

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧ. ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

## **Πολυδιάστατα Ημιδομημένα Δεδομένα**

Αναπαράσταση και επερώτηση δεδομένων στον  
Παγκόσμιο Ιστό που εξαρτώνται από ερμηνευτικά  
περιβάλλοντα και παρουσιάζουν πολλαπλές εκφάνσεις

**Γιάννης Σταύρακας**

Η διατριβή αυτή υπεβλήθη στο Εθνικό Μετσόβιο Πολυτεχνείο σαν επιμέρους  
προϋπόθεση για την απονομή του τίτλου του Διδάκτορα στην Πληροφορική.

Αθήνα, Ιούνιος 2003



Οι αποδείξεις των προτάσεων συμπεριλαμβάνονται μαζί με το υπόλοιπο υλικό της διατριβής αυτής στην τεχνική αναφορά:

Yannis Stavrakas. Multidimensional Semistructured Data: Representing and Querying Context-Dependent Multifaceted Information on the Web. Technical Report, Knowledge and Database Systems Laboratory, National Technical University of Athens, Athens, Greece, 2003.  
<http://www.dblab.ntua.gr>





ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧ. ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

## Πολυδιάστατα Ημιδομημένα Δεδομένα

Αναπαράσταση και επερώτηση δεδομένων στον Παγκόσμιο Ιστό που εξαρτώνται από ερμηνευτικά περιβάλλοντα και παρουσιάζουν πολλαπλές εκφάνσεις

Διατριβή υποβληθείσα σαν επιμέρους προϋπόθεση για τον τίτλο του  
Διδάκτορα στην Πληροφορική  
από τον

**Ιωάννη Σ. Σταύρακα**

Πτυχιούχο Φυσικό, Πανεπιστήμιο Αθηνών, 1989  
M.Sc. with Distinction in Computer Science, University College London, 1992

Επιβλέπων: Τίμος Σελλής, Καθηγητής ΕΜΠ

Εγκρίθηκε από την επταμελή εξεταστική επιτροπή την 11<sup>η</sup> Ιουνίου 2003.

Τιμολέων Σελλής  
Καθηγητής ΕΜΠ

Ιωάννης Βασιλείου  
Καθηγητής ΕΜΠ

Ανδρέας Σταφυλοπάτης  
Καθηγητής ΕΜΠ

Παναγιώτης Τσανάκας  
Καθηγητής ΕΜΠ

Φώτω Αφράτη  
Καθηγήτρια ΕΜΠ

Πάνος Κωνσταντόπουλος  
Καθηγητής Παν. Κρήτης

Μανόλης Γεργατσούλης  
Επίκ. Καθ. Ιονίου Παν.



ΙΩΑΝΝΗΣ Σ. ΣΤΑΥΡΑΚΑΣ  
Διδάκτωρ Ε.Μ.Π.

© 2003 – All rights reserved





*... στους γονείς μου*



*Από την Φαντασίαν έως εις το Χαρτί. Είναι δύσκολον πέρασμα, είναι επικίνδυνος θάλασσα. Η απόστασις φαίνεται μικρά κατά πρώτην όψιν, και εν τοσούτω πόσον μακρόν ταξίδι είναι, και πόσον επιζήμιον ενίοτε δια τα πλοία τα οποία το επιχειρούν.*

*Η πρώτη ζημία προέρχεται εκ της λίαν ευθραύστου φύσεως των εμπορευμάτων τα οποία μεταφέρουν τα πλοία. Εις τας αγοράς της Φαντασίας, τα πλείστα και τα καλύτερα πράγματα είναι κατασκευασμένα από λεπτάς υάλους και κεράμους διαφανείς, και με όλην την προσοχήν του κόσμου πολλά σπάνουν εις τον δρόμον, και πολλά σπάνουν όταν τα αποβιβάζουν εις την ξηράν. Πάσα δε τοιαύτη ζημία είναι ανεπανόρθωτος, διότι είναι έξω λόγου να γυρίσει οπίσω το πλοίον και να παραλάβει πράγματα ομοιόμορφα. Δεν υπάρχει πιθανότης να ευρεθεί το ίδιον κατάστημα το οποίον τα επώλει. Αι αγοραί της Φαντασίας έχουν καταστήματα μεγάλα και πολυτελή, αλλ' όχι μακροχρονίου διαρκείας. Αι συναλλαγái των είναι βραχείαι, εκποιούν τα εμπορεύματά των ταχέως, και διαλύουν αμέσως. Είναι πολύ σπάνιον έν πλοίον επανερχόμενον να εύρη τους αυτούς εξαγωγείς με τα αυτά είδη.*

*Μία άλλη ζημία προέρχεται εκ της χωρητικότητος των πλοίων. Αναχωρούν από τους λιμένας των ευμαρών ηπείρων καταφορτωμένα, και έπειτα όταν ευρεθούν εις την ανοικτήν θάλασσαν αναγκάζονται να ρίψουν έν μέρος εκ του φορτίου δια να σώσουν το όλον. Ούτως ώστε ουδέν σχεδόν πλοίον κατορθώνει να φέρη ακεραίους τους θησαυρούς όσους παρέλαβε. Τα απορριπτόμενα είναι βεβαίως τα ολιγοτέρας αξίας είδη, αλλά κάποτε συμβαίνει οι ναύται, εν τη μεγάλη των βία, να κάμνουν λάθη και να ρίπτουν εις την θάλασσαν πολύτιμα αντικείμενα. (...)*

*Θλιβερόν, θλιβερόν είναι άλλο πράγμα. Είναι όταν περνούν κάτι πελώρια πλοία, με κοράλλινα κοσμήματα και ιστούς εξ εβένου, με αναπεπταμένας μεγάλας σημαίας λευκάς και ερυθράς, γεμάτα με θησαυρούς, τα οποία ούτε πλησιάζουν καν εις τον λιμένα είτε διότι όλα τα είδη τα οποία φέρουν είναι απηγορευμένα, είτε διότι δεν έχει ο λιμήν αρκετόν βάθος δια να τα δεχθή. Και εξακολουθούν τον δρόμον των. Ούριος άνεμος πνέει επί των μεταξωτών των ιστίων, ο ήλιος υαλίζει την δόξαν της χρυσής των πρώρας, και απομακρύνονται ηρέμως και μεγαλοπρεπώς, απομακρύνονται δια παντός από ημάς και από τον στενόχωρον λιμένα μας. (...)*

*Κ.Π. Καβάφης, «Τα πλοία».*



## Ευχαριστίες

Διάφοροι άνθρωποι με βοήθησαν να ολοκληρώσω αυτή την διατριβή. Καταρχήν, ευχαριστώ ολόψυχα τον επιβλέποντα καθηγητή μου Τίμο Σελλή, καθηγητή στο Εθνικό Μετσόβιο Πολυτεχνείο και διευθυντή του Εργαστηρίου Συστημάτων Βάσεων Γνώσεων και Δεδομένων (DBLab). Για την υποστήριξη, την καθοδήγηση, και την ενθάρρυνσή του. Για το ότι έχει θέσει υψηλά επιστημονικά πρότυπα. Για το χρόνο και την προσπάθεια που αφιέρωσε στην δουλειά μου, ιδιαίτερα στα μεταγενέστερα στάδια. Ιδίως όμως, για ένα μίγμα ισορροπίας, επαγγελματισμού, και ακεραιότητας που εκπέμπει, δημιουργώντας ένα περιβάλλον που αισθάνεται κανείς προνομιούχος να είναι μέρος του. Οι ευχαριστίες μου απευθύνονται επίσης στους Γιάννη Βασιλείου και Ανδρέα Σταφυλοπάτη, καθηγητές στο Εθνικό Μετσόβιο Πολυτεχνείο, που διετέλεσαν μέλη της τριμελούς επιτροπής της διατριβής.

Πολλά ευχαριστώ στον Μανόλη Γεργατσούλη, επίκουρο καθηγητή στο Ιόνιο Πανεπιστήμιο. Για τη συνεργασία του στη διαμόρφωση κάποιων από τις απαντήσεις, και ακόμη περισσότερων ερωτήσεων, ειδικά στα πρώτα δύσκολα βήματα. Για τα απολαυστικά ξενύχτια όταν ετοιμάζαμε εργασίες προς υποβολή. Η συνεισφορά του στην θεμελίωση της κατεύθυνσης που ακολουθεί η παρούσα διατριβή υπήρξε καθοριστική. Επιπρόσθετα, ευχαριστώ τον Πάνο Κωνσταντόπουλο, καθηγητή στο Πανεπιστήμιο Κρήτης, τον Παναγιώτη Τσανάκα, καθηγητή στο ΕΜΠ, και την Φώτω Αφράτη, καθηγήτρια στο ΕΜΠ, που διετέλεσαν μέλη της επταμελούς εξεταστικής επιτροπής της διατριβής.

Θέλω επίσης να ευχαριστήσω τον Κώστα Σπυρόπουλο, διευθυντή έρευνας στο Εργαστήριο Τεχνολογίας Γνώσεων και Λογισμικού (SKEL) του ΕΚΕΦΕ «Δημόκριτος». Για την ανοχή του στον ανώμαλο ρυθμό μου καθώς περνούσα από την ανάπτυξη έργων στην έρευνα και αντίστροφα. Αλλά περισσότερο, για το ότι πίστεψε σε μένα αυτά τα χρόνια. Θα ήθελα επίσης να ευχαριστήσω τον Πάνο Ροντογιάννη, επίκουρο καθηγητή στο Πανεπιστήμιο Αθηνών, για μια συζήτηση που επρόκειτο να με οδηγήσει στο θέμα της παρούσας διατριβής.

Στην συνέχεια, θα ήθελα να ευχαριστήσω τους φοιτητές στο Εθνικό Μετσόβιο Πολυτεχνείο των οποίων επέβλεψα τις πτυχιακές εργασίες: τους Κωστή Πριστούρη, Αντώνη Υφαντή, Χρήστο Δουλκερίδη, Βασίλη Ζαφείρη, Κώστα Καραγεώργο, Κώστα Βαλάκα, Ανδρέα Πολυράκη, Βαγγέλη Ζορμπά, Κώστα Χατζήνα, και Cristian Chereji. Οι πρώτοι τέσσερις, μαζί με τον Πάνο Γεωργαντά και τον Στέφανο Σουλδάτο, συνέβαλαν στην ανάπτυξη των εφαρμογών για MSSD που περιγράφονται σε αυτήν την διατριβή. Επιπλέον, οι Δημήτρης Καρτέρης, Αθηνά Μουζάκη, Θεόδωρος Μητάκος, και Δημήτρης Στέρπης επηρέασαν τη μορφή της MXML, και η συνεργασία μας στο ερευνητικό πρόγραμμα ΠΕΝΕΔ-99ΕΔ265 στο ΕΚΕΦΕ «Δημόκριτος» οδήγησε στην εφαρμογή “MXML Web Server”, που περιγράφεται στην διατριβή αυτή. Τους ευχαριστώ όλους. Τα μέλη του DBLab με βοήθησαν μέσα από συζητήσεις να διαμορφώσω την βάση για να αναζητήσω θέμα διατριβής. Τους ευχαριστώ θερμά, ιδιαίτερα εκείνους που βρίσκονται πιο κοντά στα ερευνητικά μου ενδιαφέροντά: οι Θεόδωρος Δαλαμάγκας, Βαγγέλης Ζορμπάς, Ντίνος Αρκουμάνης, Σπύρος Σκιαδόπουλος, Γιώργος Αδαμόπουλος, και Πάνος Ξηρός είναι μερικοί από αυτούς.

Είμαι ευγνώμων στο Ερευνητικό Πανεπιστημιακό Ινστιτούτο Συστημάτων Επικοινωνιών και Υπολογιστών (ΕΠΙΣΕΥ) για την υποτροφία κατά το πρώτο έτος της διατριβής μου.

Επιπλέον, θέλω να ευχαριστήσω τη μητέρα μου Χρυσαιγή, τον πατέρα μου Σπύρο, και τον αδελφό μου Πάνο για την υποστήριξή τους και για το ότι υπέμειναν την απουσία μου χωρίς παράπονο. Τέλος, εγκάρδιες ευχαριστίες στην Εφη Γαζή για τη συνεχή της ενθάρρυνση και υποστήριξη.



# Περιεχόμενα

<b>ΣΥΝΟΨΗ</b> .....	<b>23</b>
<b>ABSTRACT</b> .....	<b>25</b>
<b>1 ΕΙΣΑΓΩΓΗ</b> .....	<b>27</b>
1.1 ΤΟ ΠΡΟΒΛΗΜΑ, ΚΑΙ Η ΣΥΜΒΟΛΗ ΤΗΣ ΔΙΑΤΡΙΒΗΣ .....	27
1.2 ΤΑ ΚΕΦΑΛΑΙΑ ΤΗΣ ΔΙΑΤΡΙΒΗΣ .....	31
<b>2 ΠΡΟΚΑΤΑΡΚΤΙΚΑ</b> .....	<b>35</b>
2.1 ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΠΑΓΚΟΣΜΙΟΣ ΙΣΤΟΣ .....	35
2.2 ΗΜΙΔΟΜΗΜΕΝΑ ΔΕΔΟΜΕΝΑ .....	37
2.2.1 Μοντέλα δεδομένων για SSD .....	38
2.2.2 Γλώσσες επερωτήσεων για SSD .....	40
2.3 XML.....	42
2.3.1 Συντακτικό της XML και DTD .....	43
2.3.2 Τεχνολογίες συναφείς με την XML .....	45
2.3.3 Γλώσσες επερωτήσεων για XML .....	47
2.4 MSSD ΚΑΙ MXML.....	48
<b>3 ΤΥΠΟΠΟΙΗΣΗ ΕΡΜΗΝΕΥΤΙΚΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ</b> .....	<b>49</b>
3.1 ΣΧΕΤΙΚΕΣ ΕΡΓΑΣΙΕΣ .....	49
3.2 ΕΡΜΗΝΕΥΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ, ΔΙΑΣΤΑΣΕΙΣ, ΚΑΙ ΚΟΣΜΟΙ ΣΤΑ MSSD .....	52
3.2.1 Παραδοχές σχετικές με τις διαστάσεις.....	53
3.2.2 Πιθανοί κόσμοι .....	53
3.3 ΠΡΟΣΔΙΟΡΙΣΤΕΣ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΚΑΙ ΠΡΑΞΕΙΣ ΠΕΡΙΒΑΛΛΟΝΤΟΣ .....	55
3.3.1 Όροι προσδιοριστή περιβάλλοντος.....	56
3.3.1.1 Αντιστοιχία με κόσμους .....	56
<i>Επεκτεταμένη μορφή όρου</i> .....	57
<i>Ανάπτυξη όρου</i> .....	58
<i>Ανάπτυξη επεκτεταμένου όρου</i> .....	59
3.3.1.2 <i>Ισότητα όρου</i> .....	59
3.3.1.3 <i>Κενός όρος και καθολικός όρος</i> .....	61
3.3.1.4 <i>Τομή όρου</i> .....	63
<i>Αμοιβαία αποκλειόμενοι όροι</i> .....	64
3.3.1.5 <i>Οι όροι κατά την ένωση και την διαφορά</i> .....	65
3.3.1.6 <i>Υποσύνολο όρου</i> .....	65
3.3.2 Προσδιοριστές περιβάλλοντος.....	66
3.3.2.1 Αντιστοιχία με κόσμους .....	67
<i>Επεκτεταμένη μορφή περιβάλλοντος</i> .....	67
<i>Ανάπτυξη περιβάλλοντος</i> .....	68

<i>Ανάπτυξη επεκτεταμένου περιβάλλοντος</i> .....	69
3.3.2.2 <i>Ισότητα περιβάλλοντος</i> .....	69
3.3.2.3 <i>Κενό περιβάλλον και καθολικό περιβάλλον</i> .....	70
3.3.2.4 <i>Απλοποίηση προσδιοριστών περιβάλλοντος</i> .....	72
3.3.2.5 <i>Τομή περιβάλλοντος</i> .....	73
<i>Αμοιβαία αποκλειόμενα περιβάλλοντα</i> .....	74
3.3.2.6 <i>Ένωση περιβάλλοντος και ένωση όρου</i> .....	75
<i>Ένωση όρου</i> .....	75
<i>Ένωση περιβάλλοντος</i> .....	76
3.3.2.7 <i>Διαφορά περιβάλλοντος και διαφορά όρου</i> .....	77
<i>Διαφορά όρου</i> .....	77
<i>Διαφορά περιβάλλοντος</i> .....	80
3.3.2.8 <i>Υποσύνολο περιβάλλοντος</i> .....	81
3.4 <b>ΙΔΙΟΤΗΤΕΣ ΤΩΝ ΠΡΑΞΕΩΝ ΠΕΡΙΒΑΛΛΟΝΤΟΣ</b> .....	82
3.4.1 <i>Εξάρτηση από διαστάσεις</i> .....	82
3.4.2 <i>Ισότητα περιβάλλοντος και υποσύνολο περιβάλλοντος</i> .....	83
3.4.3 <i>Τομή περιβάλλοντος, ένωση περιβάλλοντος, και διαφορά περιβάλλοντος</i> .....	84
3.4.4 <i>Εκφράσεις περιβάλλοντος και συνθήκες περιβάλλοντος</i> .....	85
3.5 <b>ΠΟΛΥΠΛΟΚΟΤΗΤΑ ΤΩΝ ΠΡΑΞΕΩΝ ΠΕΡΙΒΑΛΛΟΝΤΟΣ</b> .....	85
3.6 <b>ΣΗΜΕΙΟΓΡΑΦΙΑ ΓΙΑ ΠΡΟΣΔΙΟΡΙΣΤΕΣ ΠΕΡΙΒΑΛΛΟΝΤΟΣ</b> .....	87
3.7 <b>ΠΕΡΙΛΗΨΗ</b> .....	89
<b>4 ΕΝΑ ΜΟΝΤΕΛΟ ΔΕΔΟΜΕΝΩΝ ΓΙΑ MSSD</b> .....	<b>91</b>
4.1 <b>ΠΟΛΥΔΙΑΣΤΑΤΕΣ ΟΝΤΟΤΗΤΕΣ</b> .....	92
4.2 <b>ΠΟΛΥΔΙΑΣΤΑΤΟΙ ΓΡΑΦΟΙ ΔΕΔΟΜΕΝΩΝ</b> .....	93
4.2.1 <i>Γιατί αρχίζουμε από το OEM;</i> .....	94
4.2.2 <i>Αναπαράσταση πολυδιάστατων οντοτήτων</i> .....	95
4.2.3 <i>Πολυδιάστατοι Γράφοι Δεδομένων</i> .....	97
4.2.4 <i>Το σύνολο των διαστάσεων</i> .....	99
4.3 <b>ΜΕΤΑΔΟΣΗ ΠΕΡΙΒΑΛΛΟΝΤΟΣ</b> .....	100
4.3.1 <i>Κληρονομούμενο περιβάλλον</i> .....	100
4.3.1.1 <i>Υπολογισμός του κληρονομούμενου περιβάλλοντος</i> .....	101
4.3.1.2 <i>Σημασία του κληρονομούμενου περιβάλλοντος</i> .....	104
4.3.1.3 <i>Κληρονομούμενο περιβάλλον της ρίζας</i> .....	107
4.3.2 <i>Κάλυψη περιβάλλοντος</i> .....	108
4.3.2.1 <i>Υπολογισμός της κάλυψης περιβάλλοντος</i> .....	109
4.3.2.2 <i>Σημασία της κάλυψης περιβάλλοντος</i> .....	109
4.3.2.3 <i>Κάλυψη περιβάλλοντος της ρίζας</i> .....	112
4.3.3 <i>Κληρονομούμενη κάλυψη</i> .....	112
4.3.3.1 <i>Κληρονομούμενη κάλυψη μονοπατιού</i> .....	114
4.3.3.2 <i>Σημασία της κληρονομούμενης κάλυψης</i> .....	114
4.3.3.3 <i>Περιβαλλοντική προβολή γράφου</i> .....	116
4.4 <b>ΑΝΑΓΩΓΗ</b> .....	116



4.4.1	Αναγωγή σε OEM.....	117
4.4.1.1	Γράφοι αιτιοκρατικοί ως προς το περιβάλλον .....	117
4.4.1.2	Αναγωγή σε OEM.....	119
4.4.1.3	OEM γράφοι που υφίστανται κάτω από δύο ή περισσότερους κόσμους 121	
4.4.2	Μερική αναγωγή.....	123
4.4.3	Στοιχειώδεις πράξεις της αναγωγής.....	125
4.5	ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ .....	126
4.6	ΠΟΛΥΔΙΑΣΤΑΤΟ OEM .....	130
4.7	ΕΚΦΡΑΣΕΙΣ MSSD-EXPRESSIONS .....	131
4.8	ΠΕΡΙΛΗΨΗ.....	133
<b>5</b>	<b>ΠΟΛΥΔΙΑΣΤΑΤΗ ΓΛΩΣΣΑ ΕΠΕΡΩΤΗΣΕΩΝ.....</b>	<b>135</b>
5.1	ΧΡΗΣΗ ΤΟΥ ΜΟΕΜ ΚΑΙ ΤΗΣ ΜQL.....	136
5.2	ΕΚΦΡΑΣΕΙΣ ΜΟΝΟΠΑΤΙΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ .....	138
5.2.1	Ενσωμάτωση ερμηνευτικού περιβάλλοντος σε εκφράσεις μονοπατιού 139	
5.2.1.1	Εκφράσεις μονοπατιού περιβάλλοντος και κανονική μορφή.....	139
5.2.1.2	Αποτίμηση εκφράσεων μονοπατιού περιβάλλοντος.....	141
5.2.1.3	Τμήματα οντότητας και τμήματα έκφρασης.....	142
5.2.2	Περιοριστές περιβάλλοντος.....	143
5.2.2.1	Περιοριστές περιβάλλοντος σαν προσδιοριστές περιβάλλοντος.....	144
5.2.2.2	Πρότυπα περιβάλλοντος .....	145
5.2.2.3	Περιοριστές περιβάλλοντος σαν πρότυπα περιβάλλοντος .....	147
5.2.2.4	Περιοριστές περιβάλλοντος σαν μεταβλητές περιβάλλοντος.....	147
5.2.3	Υπονοούμενοι περιοριστές περιβάλλοντος.....	148
5.2.4	Σημασιολογία των εκφράσεων μονοπατιού περιβάλλοντος.....	151
5.3	ΒΑΣΙΚΑ ΣΤΟΙΧΕΙΑ ΤΗΣ ΜQL.....	154
5.3.1	Όρος “from” .....	155
5.3.2	Όρος “select” .....	156
5.3.3	Όρος “where” .....	158
5.3.4	Όρος “within” .....	160
5.3.5	Όρος “context” .....	160
5.3.6	Διαστάσεις και πεδία ορισμού των στην ΜQL.....	162
5.4	ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ .....	162
5.4.1	Γενικευμένες εκφράσεις μονοπατιού περιβάλλοντος.....	162
5.4.1.1	Ομαλές εκφράσεις.....	163
5.4.1.2	Μπαλαντέρ (wildcards) .....	165
5.4.1.3	Μεταβλητές μονοπατιού και μεταβλητές ετικέτας .....	165
5.4.2	Εμφωλιασμένες επερωτήσεις ΜQL.....	168
5.4.3	Κατασκευή αποτελεσμάτων.....	171
5.4.3.1	Κατασκευή ακμών οντότητας και ακμών περιβάλλοντος.....	172
5.4.3.2	Εκφράσεις μονοπατιού περιβάλλοντος στο “select” .....	174

<i>Συμπερασμός των ετικετών ακμών</i> .....	175
<i>Ομαδοποίηση αποτελεσμάτων</i> .....	176
5.4.3.3 <i>Αναγωγή του αποτελέσματος γράφου</i> .....	178
5.5 ΥΛΟΠΟΙΗΣΗ ΠΑΝΩ ΣΤΟ LORE.....	179
5.5.1 Μετασχηματισμός βάσεων MOEM σε OEM .....	181
5.5.2 Μετάφραση επερωτήσεων MQL σε Lorel.....	183
5.5.2.1 <i>Από εκφράσεις μονοπατιού περιβάλλοντος σε εκφράσεις μονοπατιού</i> 184	
5.5.2.2 <i>Μεταβλητές περιβάλλοντος και ο όρος “within”</i> .....	186
5.5.3 Μετασχηματισμός αποτελεσμάτων από OEM σε Πολυδιάστατο Γράφο Δεδομένων.....	188
5.6 ΠΕΡΙΛΗΨΗ.....	189
<b>6 ΑΝΤΙΜΕΤΩΠΙΣΗ ΑΛΛΑΓΩΝ ΜΕ ΧΡΗΣΗ MSSD .....</b>	<b>191</b>
6.1 ΛΕΙΤΟΥΡΓΙΕΣ ΒΑΣΙΚΩΝ ΑΛΛΑΓΩΝ ΤΟΥ MOEM.....	191
6.2 ΑΝΑΠΑΡΑΣΤΑΣΗ ΙΣΤΟΡΙΚΟΥ ΤΟΥ OEM ΜΕ MOEM .....	193
6.2.1 Μοντελοποίηση ιστορικών του OEM με χρήση MOEM .....	193
6.2.2 Εφαρμογή ιδιοτήτων των MSSD.....	197
6.3 ΥΛΟΠΟΙΗΣΗ: “OEM HISTORY” .....	199
6.4 ΑΝΑΠΑΡΑΣΤΑΣΗ ΑΛΛΑΓΩΝ ΣΕ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ MOEM .....	202
6.5 ΕΠΕΡΩΤΗΣΕΙΣ ΑΛΛΑΓΩΝ ΜΕ ΤΗΝ MQL.....	205
6.5.1 Επερωτήσεις ιστορικών του OEM.....	205
6.5.2 Επερωτήσεις ιστορικών του MOEM .....	212
6.6 ΣΧΕΤΙΚΕΣ ΕΡΓΑΣΙΕΣ .....	215
6.7 ΠΕΡΙΛΗΨΗ.....	217
<b>7 ΠΟΛΥΔΙΑΣΤΑΤΗ XML .....</b>	<b>219</b>
7.1 ΕΝΣΩΜΑΤΩΣΗ ΕΡΜΗΝΕΥΤΙΚΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΣΤΗΝ XML.....	220
7.1.1 Γιατί επεκτείνουμε το συντακτικό της XML;.....	220
7.1.2 Συντακτικό της Πολυδιάστατης XML.....	221
7.1.3 Πως οι διαστάσεις εφαρμόζονται σε στοιχεία .....	223
7.1.4 Πως οι διαστάσεις εφαρμόζονται σε γνωρίσματα .....	224
7.2 ΠΟΛΥΔΙΑΣΤΑΤΟ DTD .....	225
7.2.1 Δήλωση διαστάσεων .....	225
7.2.2 Δήλωση στοιχείων .....	226
7.2.3 Δήλωση γνωρισμάτων .....	226
7.2.4 Παράδειγμα MDTD.....	227
7.3 ΙΔΙΟΤΗΤΕΣ ΤΗΣ MXML.....	228
7.4 ΤΟ ΠΟΛΥΔΙΑΣΤΑΤΟ ΠΑΡΑΔΕΙΓΜΑ .....	229
7.4.1 Παρουσίαση πολυδιάστατων ημιδομημένων δεδομένων .....	229
7.4.2 Ένα εκτενές παράδειγμα .....	230

7.5	ΠΡΟΤΥΠΗ ΥΛΟΠΟΙΗΣΗ .....	234
7.5.1	Επεκτείνοντας τα URL .....	234
7.5.2	Σχεδιασμός συστήματος .....	235
7.5.3	Υλοποίηση συστήματος.....	237
7.6	ΠΕΡΙΛΗΨΗ.....	239
<b>8</b>	<b>ΕΝΑ ΣΥΣΤΗΜΑ ΓΙΑ ΤΗΝ ΔΙΑΧΕΙΡΙΣΗ MSSD.....</b>	<b>241</b>
8.1	ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ .....	241
8.2	ΔΙΑΧΕΙΡΙΣΤΗΣ ΠΟΛΥΔΙΑΣΤΑΤΩΝ ΔΕΔΟΜΕΝΩΝ.....	243
8.2.1	Λειτουργίες γενικής χρήσης.....	243
8.2.2	Μονάδα συντονισμού .....	243
8.2.3	Μονάδα λειτουργιών ενημέρωσης.....	244
8.2.4	Μονάδα εισαγωγής / εξαγωγής.....	244
8.2.5	Μονάδα λειτουργιών αναγωγής.....	244
8.2.6	Υποσύστημα επερωτήσεων .....	245
8.3	MSSDESIGNER .....	245
8.4	ΠΕΡΙΛΗΨΗ.....	250
<b>9</b>	<b>ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ.....</b>	<b>251</b>
	<b>ΠΑΡΑΡΤΗΜΑ Α: ΠΡΟΔΙΑΓΡΑΦΗ ΣΥΝΤΑΚΤΙΚΟΥ ΤΗΣ MQL.....</b>	<b>257</b>
A.1	ΣΥΝΤΑΚΤΙΚΟ ΕΚΦΡΑΣΕΩΝ ΜΟΝΟΠΑΤΙΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ.....	257
A.2	ΣΥΝΤΑΚΤΙΚΟ ΤΗΣ MQL .....	261
	<b>ΑΝΑΦΟΡΕΣ.....</b>	<b>265</b>
	<b>ΓΛΩΣΣΑΡΙ / GLOSSARY.....</b>	<b>279</b>



## Κατάλογος εικόνων

ΕΙΚΟΝΑ 2.1: ΤΜΗΜΑ ΜΙΑΣ ΒΙΒΛΙΟΓΡΑΦΙΚΗΣ ΟΕΜ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ.....	39
ΕΙΚΟΝΑ 3.1: ΣΗΜΕΙΟ ΣΤΟΝ ΠΟΛΥΔΙΑΣΤΑΤΟ ΧΩΡΟ ΠΟΥ ΑΝΑΠΑΡΙΣΤΑ ΕΝΑΝ ΠΙΘΑΝΟ ΚΟΣΜΟ.....	54
ΕΙΚΟΝΑ 4.1: ΓΡΑΦΙΚΗ ΑΝΑΠΑΡΑΣΤΑΣΗ ΠΟΛΥΔΙΑΣΤΑΤΩΝ ΟΝΤΟΤΗΤΩΝ.....	95
ΕΙΚΟΝΑ 4.2: ΕΝΑΛΛΑΚΤΙΚΟΙ ΤΡΟΠΟΙ ΑΝΑΠΑΡΑΣΤΑΣΗΣ ΠΟΛΥΔΙΑΣΤΑΤΩΝ ΟΝΤΟΤΗΤΩΝ.....	96
ΕΙΚΟΝΑ 4.3: ΠΑΡΑΔΕΙΓΜΑ ΠΟΛΥΔΙΑΣΤΑΤΟΥ ΓΡΑΦΟΥ ΔΕΔΟΜΕΝΩΝ ΠΟΥ ΑΝΑΠΑΡΙΣΤΑ ΕΝΑΝ ΟΔΗΓΟ ΔΙΑΣΚΕΔΑΣΗΣ ΕΞΑΡΤΩΜΕΝΟ ΑΠΟ ΤΟ ΕΡΜΗΝΕΥΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ.....	98
ΕΙΚΟΝΑ 4.4: ΥΠΟΛΟΓΙΣΜΟΣ ΤΟΥ ΚΛΗΡΟΝΟΜΟΥΜΕΝΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ.....	102
ΕΙΚΟΝΑ 4.5: Ο ΥΠΟΓΡΑΦΟΣ ΤΟΥ (Α) ΠΟΥ ΥΦΙΣΤΑΤΑΙ ΚΑΤΩ ΑΠΟ ΤΟΝ ΚΟΣΜΟ [SCALE=LOW] ΦΑΙΝΕΤΑΙ ΣΤΟ (Β).....	106
ΕΙΚΟΝΑ 4.6: ΠΕΡΙΟΡΙΖΟΝΤΑΣ ΤΟ ΚΛΗΡΟΝΟΜΟΥΜΕΝΟ ΠΕΡΙΒΑΛΛΟΝ ΤΗΣ ΡΙΖΑΣ ΣΕ C.....	107
ΕΙΚΟΝΑ 4.7: ΚΑΛΥΨΗ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΚΑΙ Η ΣΧΕΣΗ ΤΗΣ ΜΕ ΤΗΝ ΕΞΑΓΩΓΗ ΣΥΜΒΑΤΙΚΩΝ ΟΕΜ ΓΡΑΦΩΝ.....	110
ΕΙΚΟΝΑ 4.8: ΓΡΑΦΟΣ ΕΠΙΣΗΜΕΙΩΜΕΝΟΣ ΜΕ ΤΗΝ ΚΛΗΡΟΝΟΜΟΥΜΕΝΗ ΚΑΛΥΨΗ ΤΩΝ ΑΚΜΩΝ.....	113
ΕΙΚΟΝΑ 4.9: ΑΙΤΙΟΚΡΑΤΙΚΟΙ ΚΑΙ ΜΗ ΑΙΤΙΟΚΡΑΤΙΚΟΙ ΩΣ ΠΡΟΣ ΤΟ ΠΕΡΙΒΑΛΛΟΝ ΓΡΑΦΟΙ.....	118
ΕΙΚΟΝΑ 4.10: ΤΟ ΟΕΜ ΠΟΥ ΠΑΡΑΓΕΤΑΙ ΑΠΟ ΤΗΝ ΑΝΑΓΩΓΗ ΤΟΥ ΠΟΛΥΔΙΑΣΤΑΤΟΥ ΓΡΑΦΟΥ ΔΕΔΟΜΕΝΩΝ ΤΗΣ ΕΙΚΟΝΑ 4.3 ΚΑΤΩ ΑΠΟ ΤΟΝ ΚΟΣΜΟ W.....	121
ΕΙΚΟΝΑ 4.11: ΠΡΟΒΛΗΜΑΤΑ ΣΤΗΝ ΕΞΑΓΩΓΗ ΟΕΜ ΓΡΑΦΟΥ ΠΟΥ ΥΦΙΣΤΑΤΑΙ ΚΑΤΩ ΑΠΟ ΔΥΟ ΚΟΣΜΟΥΣ.....	122
ΕΙΚΟΝΑ 4.12: Ο ΥΠΟΓΡΑΦΟΣ ΠΟΥ ΠΑΡΑΓΕΤΑΙ ΑΠΟ ΜΕΡΙΚΗ ΑΝΑΓΩΓΗ ΤΟΥ ΠΟΛΥΔΙΑΣΤΑΤΟΥ ΓΡΑΦΟΥ ΔΕΔΟΜΕΝΩΝ ΤΗΣ ΕΙΚΟΝΑ 4.3 ΓΙΑ ΤΟ ΕΡΜΗΝΕΥΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ C.....	125
ΕΙΚΟΝΑ 4.13: ΜΕΤΑΤΡΟΠΗ ΠΟΛΥΔΙΑΣΤΑΤΟΥ ΓΡΑΦΟΥ ΔΕΔΟΜΕΝΩΝ ΣΕ ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ.....	128
ΕΙΚΟΝΑ 4.14: ΤΜΗΜΑ ΤΟΥ ΓΡΑΦΟΥ ΤΗΣ ΕΙΚΟΝΑ 4.3 ΣΕ ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ.....	130
ΕΙΚΟΝΑ 5.1: ΜΙΑ ΜΟΕΜ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ ΓΙΑ ΕΝΑΝ ΟΔΗΓΟ ΔΙΑΣΚΕΔΑΣΗΣ ΕΞΑΡΤΩΜΕΝΟ ΑΠΟ ΤΟ ΕΡΜΗΝΕΥΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ.....	137
ΕΙΚΟΝΑ 5.2: ΤΑ ΜΟΝΟΠΑΤΙΑ ΔΕΔΟΜΕΝΩΝ ΕΧΟΥΝ ΣΗΜΑΣΙΑ.....	152
ΕΙΚΟΝΑ 5.3: ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΠΕΡΩΤΗΣΕΩΝ ΜΟΛ ΣΑΝ ΠΟΛΥΔΙΑΣΤΑΤΟΙ ΓΡΑΦΟΙ ΔΕΔΟΜΕΝΩΝ.....	157
ΕΙΚΟΝΑ 5.4: ΑΝΕΥΡΕΣΗ ΚΕΝΩΝ ΚΛΗΡΟΝΟΜΟΥΜΕΝΩΝ ΚΑΛΥΨΕΩΝ ΚΑΙ ΠΑΡΑΒΙΑΣΕΩΝ ΤΗΣ ΑΙΤΙΟΚΡΑΤΙΑΣ ΩΣ ΠΡΟΣ ΤΟ ΠΕΡΙΒΑΛΛΟΝ ΣΕ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΓΡΑΦΩΝ.....	167

ΕΙΚΟΝΑ 5.5: ΑΝΕΥΡΕΣΗ ΜΗ ΠΡΟΣΒΑΣΙΜΩΝ ΕΚΦΑΝΣΕΩΝ ΓΙΑ ΚΑΘΕ ΕΙΣΕΡΧΟΜΕΝΗ ΑΚΜΗ ΟΝΤΟΤΗΤΑΣ. ....	171
ΕΙΚΟΝΑ 5.6: ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΠΕΡΩΤΗΣΕΩΝ MQL. ....	173
ΕΙΚΟΝΑ 5.7: ΑΠΟΤΙΜΗΣΗ ΕΠΕΡΩΤΗΣΕΩΝ MQL ΜΕ ΧΡΗΣΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ LORE ΣΑΝ ΥΠΟΒΑΘΡΟ. ....	180
ΕΙΚΟΝΑ 5.8: ΑΝΑΠΑΡΑΣΤΑΣΗ ΠΟΛΥΔΙΑΣΤΑΤΟΥ ΓΡΑΦΟΥ ΔΕΔΟΜΕΝΩΝ ΜΕ ΧΡΗΣΗ OEM. ....	182
ΕΙΚΟΝΑ 6.1: ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΤΩΝ ΛΕΙΤΟΥΡΓΙΩΝ ΒΑΣΙΚΩΝ ΑΛΛΑΓΩΝ ΤΟΥ OEM ΜΕ ΧΡΗΣΗ ΜΟΕΜ. ....	194
ΕΙΚΟΝΑ 6.2: ΑΡΧΙΚΗ ΚΑΤΑΣΤΑΣΗ ΤΗΣ ΒΑΣΗΣ ΤΟΥ ΠΑΡΑΔΕΙΓΜΑΤΟΣ ΣΤΗΝ ΕΦΑΡΜΟΓΗ OEM HISTORY. ....	200
ΕΙΚΟΝΑ 6.3: Η ΒΑΣΗ ΤΟΥ ΠΑΡΑΔΕΙΓΜΑΤΟΣ ΜΕΤΑ ΑΠΟ ΔΥΟ ΑΚΟΛΟΥΘΙΕΣ ΒΑΣΙΚΩΝ ΑΛΛΑΓΩΝ ΠΑΝΩ ΣΤΗΝ ΑΡΧΙΚΗ ΚΑΤΑΣΤΑΣΗ ΤΗΣ ΒΑΣΗΣ. ....	201
ΕΙΚΟΝΑ 6.4: Η ΒΑΣΗ ΤΟΥ ΠΑΡΑΔΕΙΓΜΑΤΟΣ ΜΕΤΑ ΑΠΟ ΑΚΟΜΗ ΔΥΟ ΑΚΟΛΟΥΘΙΕΣ ΒΑΣΙΚΩΝ ΑΛΛΑΓΩΝ ΠΑΝΩ ΣΤΗΝ ΒΑΣΗ ΤΗΣ ΕΙΚΟΝΑ 6.3. ....	201
ΕΙΚΟΝΑ 6.5: ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΛΕΙΤΟΥΡΓΙΩΝ ΒΑΣΙΚΩΝ ΑΛΛΑΓΩΝ ΜΟΕΜ ΜΕ ΜΟΕΜ. ....	203
ΕΙΚΟΝΑ 6.6: ΑΝΑΠΑΡΑΣΤΑΣΗ ΤΟΥ ΙΣΤΟΡΙΚΟΥ ΤΟΥ ΠΟΛΥΔΙΑΣΤΑΤΟΥ MUSIC CLUB ΣΤΟ (Α) ΜΕΣΩ ΤΟΥ ΜΟΕΜ ΣΤΟ (Β). ....	204
ΕΙΚΟΝΑ 6.7: Η ΜΟΕΜ ΒΑΣΗ ΤΗΣ ΕΙΚΟΝΑ 6.4 ΣΕ ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ, ΣΥΜΠΛΗΡΩΜΕΝΗ ΜΕ ΤΗΝ ΚΛΗΡΟΝΟΜΟΥΜΕΝΗ ΚΑΛΥΨΗ ΤΩΝ ΕΔΡΩΝ. ....	206
ΕΙΚΟΝΑ 6.8: ΜΕΡΙΚΕΣ ΑΝΑΓΩΓΕΣ ΓΙΑ ΔΙΑΦΟΡΑ ΕΡΜΗΝΕΥΤΙΚΑ ΠΕΡΙΒΑΛΛΟΝΤΑ ΣΑΝ ΑΠΟΤΕΛΕΣΜΑ ΕΠΕΡΩΤΗΣΗΣ MQL. ....	211
ΕΙΚΟΝΑ 6.9: ΤΟ ΜΟΕΜ ΤΗΣ ΕΙΚΟΝΑ 6.6 (Β) ΣΕ ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ. ....	212
ΕΙΚΟΝΑ 6.10: ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΠΕΡΩΤΗΣΕΩΝ MQL ΣΤΟ ΙΣΤΟΡΙΚΟ ΜΟΕΜ ΒΑΣΗΣ. ....	214
ΕΙΚΟΝΑ 7.1: MXML, MXSL, ΚΑΙ Η ΣΧΕΣΗ ΤΟΥΣ. ....	230
ΕΙΚΟΝΑ 7.2: ΑΡΧΙΤΕΚΤΟΝΙΚΗ MXML WEB SERVER. ....	235
ΕΙΚΟΝΑ 7.3: ΟΙ ΥΠΟΜΟΝΑΔΕΣ ΤΟΥ VIEW EXTRACTOR ΕΜΦΑΝΙΖΟΝΤΑΙ ΣΤΟ (Α), ΚΑΙ ΟΙ ΥΠΟΜΟΝΑΔΕΣ ΤΟΥ CONVENTIONAL WEB SERVER ΕΜΦΑΝΙΖΟΝΤΑΙ ΣΤΟ (Β). ....	236
ΕΙΚΟΝΑ 7.4: ΕΝΑ ΣΤΙΓΜΙΟΤΥΠΟ ΟΘΟΝΗΣ ΠΟΥ ΕΜΦΑΝΙΖΕΙ ΜΙΑ ΑΠΑΝΤΗΣΗ ΑΠΟ ΤΟΝ MXML WEB SERVER. ....	238
ΕΙΚΟΝΑ 8.1: ΣΥΝΟΛΙΚΗ ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ. ....	242
ΕΙΚΟΝΑ 8.2: ΔΕΙΓΜΑ ΟΘΟΝΗΣ ΤΟΥ MSSDESIGNER. ....	246
ΕΙΚΟΝΑ 8.3: ΑΡΧΙΚΟΠΟΙΗΣΗ ΤΟΥ ΥΠΟΣΥΣΤΗΜΑΤΟΣ ΕΠΕΡΩΤΗΣΕΩΝ ΜΕ ΜΙΑ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ ΓΡΑΦΟΥ. ....	248
ΕΙΚΟΝΑ 8.4: ΑΠΟΤΙΜΗΣΗ ΜΙΑΣ ΕΠΕΡΩΤΗΣΗΣ MQL ΚΑΙ ΕΜΦΑΝΙΣΗ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ. ....	249

## ΣΥΝΟΨΗ

Ενώ στις παραδοσιακές βάσεις δεδομένων και στα αντίστοιχα πληροφοριακά συστήματα το πλήθος των χρηστών είναι λίγο πολύ γνωστό και το υπόβαθρό τους είναι σε μεγάλο βαθμό ομογενές, οι χρήστες του Παγκόσμιου Ιστού δεν έχουν κοινό υπόβαθρο ούτε όμοιες θεωρήσεις όταν επεξεργάζονται και ερμηνεύουν δεδομένα. Αυτοί οι χρήστες μπορεί να έχουν διαφορετικές οπτικές των ίδιων πληροφοριακών οντοτήτων, κάτι που θα πρέπει να ληφθεί υπόψη από τα μοντέλα δεδομένων και τις γλώσσες επερωτήσεων για τον Παγκόσμιο Ιστό. Επιπλέον, οι παροχείς πληροφορίας χρειάζεται συχνά να διαχειριστούν παραλλαγές των ίδιων ουσιαστικά δεδομένων, οι οποίες απευθύνονται σε διαφορετικές ομάδες καταναλωτών.

Σε αυτήν την διατριβή υποστηρίζουμε ότι τα δεδομένα του Ιστού θα πρέπει να μπορούν να προσαρμόζονται σε διαφορετικά *ερμηνευτικά περιβάλλοντα* [contexts], και ότι η δυνατότητα αυτή θα πρέπει να διατίθεται με τρόπο ευέλικτο και εννιαίο στο επίπεδο των συστημάτων βάσεων δεδομένων. Προτείνουμε τα *πολυδιάστατα ημιδομημένα δεδομένα* (MSSD) [multidimensional semistructured data], όπου πληροφοριακές οντότητες μπορούν να εκδηλώνουν διαφορετικές εκφάνσεις, ανάλογα με το ερμηνευτικό περιβάλλον. Καταρχήν, ορίζουμε έναν απλό και ευέλικτο τρόπο για να εκφράζουμε ερμηνευτικά περιβάλλοντα χρησιμοποιώντας *διαστάσεις* [dimensions]. Σύμφωνα με την προσέγγισή μας, το ερμηνευτικό περιβάλλον αντιπροσωπεύει μία σειρά από εναλλακτικούς *κόσμους* [worlds], σε κάθε έναν από τους οποίους οι πληροφορίες αποκτούν μια συγκεκριμένη ερμηνεία. Ορίζουμε πράξεις ανάμεσα από ερμηνευτικά περιβάλλοντα, και εξετάζουμε τις ιδιότητές τους σε βάθος. Στην συνέχεια, προτείνουμε ένα μοντέλο δεδομένων για MSSD που ονομάζεται *Πολυδιάστατος Γράφος Δεδομένων* [Multidimensional Data Graph], και ερευνούμε διεξοδικά τις ιδιότητές του. Ο Πολυδιάστατος Γράφος Δεδομένων επεκτείνει το Object Exchange Model (OEM), ένα μοντέλο γράφου για ημιδομημένα δεδομένα, προκειμένου να υποστηρίξει με άμεσο τρόπο πληροφορία που παρουσιάζει πολλαπλές εκφάνσεις. Το *Πολυδιάστατο OEM* (MOEM) [Multidimensional OEM] είναι μια ειδική περίπτωση Πολυδιάστατου Γράφου Δεδομένων, και μπορεί να θεωρηθεί σαν μια συνένωση από συμβατικούς OEM γράφους που υφίστανται κάτω από διαφορετικούς κόσμους. Οι συγκεκριμένοι OEM γράφοι είναι δυνατόν να ανακτηθούν μέσω μιας διαδικασίας που λέγεται *αναγωγή* [reduction], και η οποία μετασχηματίζει τον MOEM στον συμβατικό OEM γράφο που υφίσταται κάτω από έναν δεδομένο κόσμο. Κατόπιν, ορίζουμε την *Πολυδιάστατη Γλώσσα Επερωτήσεων* (MQL) [Multidimensional Query Language], μια γλώσσα επερωτήσεων για Πολυδιάστατους Γράφους Δεδομένων και MOEM γράφους που έχει σαν κεντρική έννοια το ερμηνευτικό περιβάλλον, και μπορεί να εκφράσει με κομψό τρόπο *επερωτήσεις οδηγούμενες από περιβάλλοντα* [context-driven queries]. Το μοντέλο MOEM μαζί με την γλώσσα επερωτήσεων MQL επιτρέπουν τη διατύπωση *δια-κοσμικών επερωτήσεων* [cross-world queries], οι οποίες δεν μπορούν να εκφραστούν με συμβατικά μοντέλα δεδομένων και συμβατικές γλώσσες επερωτήσεων. Η MQL έχει υλοποιηθεί χρησιμοποιώντας σαν βάση το σύστημα LORE, και αυτή η υλοποίηση δίνει την ευκαιρία για σύγκριση με «ισοδύναμες» επερωτήσεις στην γλώσσα Lorel.

Αξιολογήσαμε το παραπάνω πλαίσιο που αναπτύξαμε για τα MSSD εφαρμόζοντάς το σε ένα κλασικό πρόβλημα των βάσεων δεδομένων: χρησιμοποιήσαμε το μοντέλο MOEM για

να αναπαραστήσουμε το ιστορικό βάσεων δεδομένων OEM, και στην συνέχεια την MQL για να θέσουμε επερωτήσεις στο ιστορικό αυτό. Το MOEM και η MQL αποδείχθηκαν ικανά όχι μόνο για να αναπαραστήσουν και να επερωτήσουν το ιστορικό βάσεων OEM, αλλά και το ιστορικό βάσεων MOEM. Επιπλέον, σαν παράδειγμα του πώς οι έννοιες που εισαγάγαμε μπορούν να επεκταθούν σε άλλες κατευθύνσεις, παραθέτουμε την *Πολυδιάστατη XML (MXML)* [Multidimensional XML], μια εκδοχή της XML που ενσωματώνει ερμηνευτικά περιβάλλοντα. Για την παροχή και παρουσίαση πληροφορίας που ενέχει πολλαπλές εκφάνσεις προτείνουμε το *πολυδιάστατο παράδειγμα* [multidimensional paradigm], και επιδεικνύουμε μια σχετική πρότυπη υλοποίηση. Τέλος, παρουσιάζουμε ένα υλοποιημένο σύστημα για τη διαχείριση MSSD. Το σύστημα αυτό εμπεριέχει το υποσύστημα *MSSDesigner*, ένα γραφικό περιβάλλον για την εύκολη δημιουργία Πολυδιάστατων Γράφων Δεδομένων, καθώς και για την εκτέλεση επερωτήσεων διατυπωμένων στην MQL.

Εκτιμούμε ότι τα MSSD αντιμετωπίζουν ένα σημαντικό πρόβλημα των βάσεων δεδομένων στο σημερινό παγκοσμιοποιημένο περιβάλλον, και ότι έχουν τη δυνατότητα να εφαρμοστούν πρακτικά σε διάφορες περιοχές. Επιπλέον, τα MSSD ανοίγουν μια σειρά από ενδιαφέρουσες κατευθύνσεις για περαιτέρω έρευνα.



## ABSTRACT

Contrary to traditional databases and information systems where the number of users is more or less known and their background is to a great extent homogeneous, Web users do not share the same background and do not apply the same conventions when interpreting data. Such users can have different perspectives of the same information entities, a situation that should be taken into account by Web data models and query languages. Moreover, information providers often need to manage variations of essentially the same data, which are targeted to different consumer groups.

In this thesis we argue that Web data should be able to adapt to different *contexts*, and that this capability should be managed in a flexible and uniform way at the level of database systems. We propose *multidimensional semistructure data (MSSD)*, where information entities may manifest different facets, according to context. First of all, an intuitive and flexible way to express context through *dimensions* is defined. Context is viewed as representing alternative *worlds*, in which information obtains an unambiguous interpretation. Operations between contexts are introduced, and their properties are examined in detail. Then, a data model for MSSD called *Multidimensional Data Graph* is proposed, and its properties are investigated exhaustively. Multidimensional Data Graph extends the Object Exchange Model (OEM), a graph model for semistructured data, in order to explicitly support multifaceted information. *Multidimensional OEM (MOEM)* is a special case of Multidimensional Data Graph, and can be seen as a conglomeration of conventional OEMs holding under different worlds. Those OEMs can be obtained through the process of *reduction*, which transforms a Multidimensional OEM to a conventional OEM holding under a given world. Next, we define *Multidimensional Query Language (MQL)*, a query language for Multidimensional Data Graphs and MOEMs that treats context as first-class citizen, and can express *context-driven queries* in an elegant way. MOEM and MQL allow the formulation of *cross-world queries*, which have no counterpart in conventional data models and query languages. MQL has been implemented on top of LORE, and this implementation provides a comparison with “equivalent” Lorel queries.

To assess our approach, we applied the MSSD framework we have developed to a classical problem in databases: we used MOEM and MQL to represent and query histories of OEM databases. MOEM and MQL proved powerful enough to represent and query not only OEM histories, but MOEM histories as well. Moreover, as an example of how the concepts we introduce can span other directions, we discuss *Multidimensional XML (MXML)*, a version of XML that incorporates context. For manipulating and viewing multifaceted information we propose the *multidimensional paradigm*, and demonstrate a prototype implementation. Finally, an implemented system for managing MSSD is presented. This system encompasses *MSSDesigner*, a graphical user interface that supports the interactive creation, experimentation, and querying of Multidimensional Data Graphs through MQL.

We believe that MSSD address an important problem of databases in today’s global environment, and that they have the potential to be practically applied in various domains. Moreover, MSSD open a number of interesting directions for further research.



# 1 ΕΙΣΑΓΩΓΗ

Ο όρος «ημιδομημένα δεδομένα» (SSD για συντομία) εισήχθη από την ερευνητική κοινότητα των βάσεων δεδομένων για να περιγράψει πληροφορία που έχει ακανόνιστη δομή. Το μεγαλύτερο μέρος της διαθέσιμης πληροφορίας σήμερα ανήκει στην κατηγορία αυτή, συμπεριλαμβανομένων δεδομένων στον Ιστό σε μορφή HTML και XML. Μια σειρά μοντέλων και γλωσσών επερωτήσεων προτάθηκαν τα τελευταία χρόνια για την αναπαράσταση και επερώτηση των SSD. Σε αυτή την διατριβή, υποστηρίζουμε ότι η έννοια του *ερμηνευτικού περιβάλλοντος* είναι καίρια για τις βάσεις δεδομένων σε ένα παγκοσμιοποιημένο σύστημα όπως είναι ο Ιστός, και προτείνουμε ένα νέο μοντέλο δεδομένων και γλώσσα επερωτήσεων για SSD όπου το ερμηνευτικό περιβάλλον παίζει κεντρικό ρόλο.

Στο παρόν κεφάλαιο διατυπώνουμε το πρόβλημα και παρουσιάζουμε τους λόγους που μας οδήγησαν στην προσέγγισή μας. Έπειτα, αριθμούμε τις κύριες συνεισφορές αυτής της διατριβής, και αναφερόμαστε στο υλικό που έχει δημοσιευτεί μέχρι την στιγμή που γράφονται οι γραμμές αυτές. Τέλος, κλείνουμε το κεφάλαιο παρουσιάζοντας τα κεφάλαια της διατριβής.

## 1.1 ΤΟ ΠΡΟΒΛΗΜΑ, ΚΑΙ Η ΣΥΜΒΟΛΗ ΤΗΣ ΔΙΑΤΡΙΒΗΣ

Η φύση του Παγκόσμιου Ιστού έθεσε μια σειρά από νέα προβλήματα [BBC+98] στην ερευνητική κοινότητα των βάσεων δεδομένων. Ένα από αυτά είναι ότι, ενώ στις παραδοσιακές βάσεις δεδομένων και στα αντίστοιχα πληροφοριακά συστήματα το πλήθος των χρηστών είναι λίγο πολύ γνωστό και το υπόβαθρό τους είναι σε μεγάλο βαθμό ομογενές, οι χρήστες του Παγκόσμιου Ιστού δεν έχουν κοινό υπόβαθρο ούτε όμοιες θεωρήσεις όταν επεξεργάζονται και ερμηνεύουν δεδομένα. Αυτοί οι χρήστες μπορεί να έχουν διαφορετικές οπτικές των ίδιων πληροφοριακών οντοτήτων, κάτι που θα πρέπει να ληφθεί υπόψη από τα μοντέλα δεδομένων και τις γλώσσες επερωτήσεων για τον Παγκόσμιο Ιστό. Ένα συναφές ζήτημα είναι ότι συχνά οι παροχείς πληροφορίας πρέπει να διαχειριστούν διαφορετικές παραλλαγές των ίδιων ουσιαστικά δεδομένων, που απευθύνονται σε διαφορετικές ομάδες καταναλωτών. Παρόμοια προβλήματα εμφανίζονται κατά την ενοποίηση πληροφορίας από διάφορες πηγές [GPO+97], όπου η ίδια εννοιολογική οντότητα μπορεί να εμφανίζει διαφορετική δομή, ή να περιέχει αλληλοσυγκρουόμενα δεδομένα.

Τα προβλήματα αυτά καταδεικνύουν την ανάγκη για έναν τρόπο αναπαράστασης και διαχείρισης πληροφοριακών οντοτήτων που εκδηλώνουν διαφορετικές εκφάνσεις, των οποίων το περιεχόμενο μπορεί να διαφοροποιείται ως προς την δομή του και την τιμή του. Σαν ένα απλό παράδειγμα θεωρήστε ένα προϊόν (αυτοκίνητο, laptop computer, κτλ.) οι προδιαγραφές του οποίου αλλάζουν ανάλογα με την χώρα που εξάγεται. Η μια ιστοσελίδα που πρόκειται να εμφανιστεί σε συσκευές με διαφορετικές δυνατότητες, όπως κινητά τηλέφωνα, PDAs, και προσωπικούς υπολογιστές. Ένα άλλο παράδειγμα είναι μια αναφορά που πρέπει να αναπαρασταθεί σε διάφορους βαθμούς λεπτομέρειας και σε διάφορες γλώσσες.

Μία λύση θα ήταν η δημιουργία μιας διαφορετικής αναφοράς για κάθε δυνατό συνδυασμό παραλλαγών. Βεβαίως μια τέτοια προσέγγιση δεν είναι πρακτική, αφού συνεπάγεται υπερβολική επανάληψη πληροφορίας. Ακόμη πιο σημαντικό, διαφορετικές παραλλαγές δεν συνδέονται μεταξύ τους σαν μέρη της ίδιας οντότητας. Αυτό το γεγονός δεν επιτρέπει την διατύπωση ενδιαφερόντων επερωτήσεων που ενέχουν εκφάνσεις πληροφοριακών οντοτήτων, όπως για παράδειγμα «*δώσε μου σε μικρή λεπτομέρεια τις αναφορές των οποίων οι εκφάνσεις μεγάλης λεπτομέρειας ικανοποιούν την δεδομένη συνθήκη*».

Σαν ενδεικτικό παράδειγμα θεωρήστε ένα βιβλιοπωλείο στο Διαδίκτυο που εξυπηρετεί ποικιλία πελατών, οι οποίοι έχουν διαφορετικές ανάγκες και αντιμετωπίζονται διαφορετικά ανάλογα με την πολιτική του βιβλιοπωλείου. Οι φοιτητές, για παράδειγμα, θέλουν να ψάχνουν τα βιβλία με βάση τον τίτλο ή τον συγγραφέα, και χρεώνονται χαμηλότερες τιμές απ' ό,τι οι βιβλιοθήκες. Για τις βιβλιοθήκες, από την άλλη, το ISBN του βιβλίου είναι αρκετό. Το βιβλιοπωλείο θα ήθελε να παρουσιάζει πληροφορία που να είναι προσαρμοσμένη στον αντίστοιχο πελάτη που την ζήτησε:

<p><b>Book</b></p> <p>ISBN: 0-13-110362-8, Publisher: Prentice Hall, Price: 29.4</p>
<p><b>Book</b></p> <p>Title: The C programming language, Authors: Brian W. Kernighan, Dennis M. Ritchie, Publisher: Prentice Hall, Price: 13</p>

Και οι δύο εγγραφές πιο πάνω αναφέρονται στο ίδιο βιβλίο. Η πρώτη αντιστοιχεί σε αίτημα βιβλιοθήκης, ενώ η δεύτερη σε αίτημα που έγινε από φοιτητή. Παρατηρήστε ότι η κατηγορία του πελάτη (βιβλιοθήκη ή φοιτητής) επιδρά όχι μόνο στην τιμή κάποιων αντικειμένων (π.χ. στην τιμή "Price"), αλλά επίσης και στην δομή των δεδομένων: παρουσιάζεται διαφορετικό τμήμα των διαθέσιμων δεδομένων, ανάλογα με την κάθε περίπτωση. Εκτός από την κατηγορία του πελάτη, ένας άλλος παράγοντας συνδέεται με τον τρόπο που *παρουσιάζεται* η πληροφορία, και η δεύτερη εγγραφή πιο πάνω χρησιμοποιεί γράμματα μεγαλύτερου μεγέθους για την λέξη "Book"<sup>1</sup>.

Τέτοια συμπεριφορά δεν είναι ασυνήθιστη στις εφαρμογές του Ιστού. Όμως, η ικανότητα να προσαρμόζεται η πληροφορία σε χαρακτηριστικά του προσδοκώμενου πελάτη αποτελεί προς το παρόν κομμάτι της λογικής της εφαρμογής, και πρέπει να υλοποιείται εξ αρχής σε κάθε διαφορετική περίπτωση. Αυτό που οραματιζόμαστε είναι το να γίνει η ικανότητα προσαρμογής κομμάτι της ίδιας της πληροφορίας, και να την διαχειρίζονται με ενιαίο και ευέλικτο τρόπο τα συστήματα βάσεων δεδομένων. Κάτι τέτοιο θα ελαφρύνει το βάρος της ανάπτυξης ad-hoc λύσεων· αντί για αυτό, οι εφαρμογές θα θέτουν απλές επερωτήσεις, οι οποίες θα είναι σε θέση να αντιλαμβάνονται την εξάρτηση των διαφόρων εκφάνσεων της πληροφορίας από παραμέτρους όπως η «κατηγορία του πελάτη».

Παρόλο που τα υπάρχοντα μοντέλα δεδομένων για τον Ιστό όπως το OEM [PGW95, AQM+97, Suc98], και οι γλώσσες επερωτήσεων όπως η Lorel [GPQ+97, AQM+97], είναι κατά

<sup>1</sup> Στο Κεφάλαιο 7 συζητούμε λεπτομερώς το παράδειγμα αυτό, και εξηγούμε τον μηχανισμό που χρησιμοποιείται για την παραγωγή και την εμφάνιση παραλλαγών μιας πληροφοριακής οντότητας.

κύριο λόγο ικανά να αναπαραστήσουν και να επερωτήσουν οντότητες πολλαπλών εκφάνσεων, δεν αποτελούν ικανοποιητική λύση, για τους παρακάτω λόγους<sup>2</sup>.

- (a) Κρυφή σημασιολογία: επειδή δεν αντιμετωπίζουν ευθέως το ζήτημα των πολλαπλών εκφάνσεων, είναι ευθύνη της εφαρμογής να ορίσει την σημασιολογία με έναν ad-hoc τρόπο.
- (b) Άκομμη σημειογραφία: η αναπαράσταση των πολλαπλών εκφάνσεων μιας οντότητας δεν είναι δυνατόν να γίνει στα μοντέλα αυτά με κομινό τρόπο.
- (c) Επανάληψη της πληροφορίας: μια ad-hoc προσέγγιση είναι πιθανόν να οδηγήσει σε επανάληψη της πληροφορίας που είναι κοινή, πράγμα μη επιθυμητό.

Η ανάγκη για οντότητες πολλαπλών εκφάνσεων ξεπερνά τα όρια των δεδομένων στον Ιστό. Απαντάται σε ένα ευρύ φάσμα διαφόρων περιπτώσεων, που κάποιες φορές υιοθετούν ad-hoc λύσεις περιορισμένης αποτελεσματικότητας. Ένα απτό παράδειγμα μιας τέτοιας περίπτωσης που καταδεικνύει την δυνατότητα εφαρμογών πληροφορίας πολλαπλών εκφάνσεων έχουμε στο *Microsoft PowerPoint 2000*, ένα πρόγραμμα υπολογιστή για την δημιουργία και εμφάνιση παρουσιάσεων, το οποίο περιλαμβάνει ένα χαρακτηριστικό που λέγεται *custom shows*. Αυτό το χαρακτηριστικό επιτρέπει την «...δημιουργία μιας παρουσίασης μέσα σε μια παρουσίαση. Αντί να δημιουργούμε πολλαπλές, σχεδόν ταυτόσημες παρουσιάσεις για διαφορετικά ακροατήρια, μπορούμε να ομαδοποιήσουμε και να ονοματίσουμε τις διαφάνειες που διαφέρουν και κατόπιν να μεταβούμε κατευθείαν σε αυτές την ώρα της παρουσίασης»<sup>3</sup>. Το *custom shows* δεν επιτρέπει να συμπεριλάβουμε ή να αποκλείσουμε *τμήματα μίας συγκεκριμένης διαφάνειας* ανάλογα με το προοριζόμενο ακροατήριο, πράγμα που οδηγεί σε επανάληψη σε σειρά διαφανειών των τμημάτων εκείνων που απευθύνονται σε πολλά ακροατήρια. Ιδεωδώς, θα θέλαμε να είμαστε σε θέση να επισημειώνουμε οποιαδήποτε μέρη και στοιχεία μιας παρουσίασης σαν απευθυνόμενα σε συγκεκριμένο ακροατήριο ή ακροατήρια, και στην συνέχεια να καθορίζουμε ένα ακροατήριο και να προκαλούμε την αυτόματη προσαρμογή της παρουσίασης στο ακροατήριο αυτό. Επιπλέον, θα ήταν ωφέλιμο να μπορούμε να θέτουμε χρήσιμες ερωτήσεις, όπως «*δείξε μου εκείνα τα στοιχεία των διαφανειών 3 έως 8 που απευθύνονται στα ακροατήρια Α και Β, αλλά όχι στο ακροατήριο C*». Για να πάμε ακόμη μακρύτερα, εκτός από το προοριζόμενο ακροατήριο και άλλοι παράγοντες μπορούν να επηρεάσουν μια παρουσίαση: για παράδειγμα, θεωρήστε την περίπτωση όπου η διάρκεια της παρουσίασης περιορίζεται από διαφορετικά χρονικά όρια ανάλογα με την περίπτωση. Στην περίπτωση αυτή, τμήματα του υλικού της παρουσίασης θα μπορούσαν να χαρακτηριστούν με διαφορετικούς βαθμούς προτεραιότητας ή σημασίας, επιτρέποντας την αυτόματη παράλειψη της πληροφορίας χαμηλής προτεραιότητας όταν το χρονικό όριο είναι στενό.

Πληροφορία που παρουσιάζει διαφορετικές εκφάνσεις συναντάται επίσης στις *πιθανοτικές βάσεις δεδομένων*, όπου κάθε έκφανση συνδέεται με ένα μέτρο αβεβαιότητας ως προς το εάν εκφράζει τον πραγματικό κόσμο με ακρίβεια ή όχι. Μια σχετικά πρόσφατη πιθανοτική επέκταση του σχεσιακού μοντέλου υπάρχει στο [DS96], ενώ το [DS98] παρουσιάζει την αντίστοιχη επέκταση στην SQL. Τελευταία έχουν προστεθεί πιθανοτικά χαρακτηριστικά στην XML [NJ02], όμως το μέτρο αβεβαιότητας που προσδιορίζει τις εκφάνσεις έχει μια εξειδικευμένη σημασιολογία που δεν είναι αρκετά γενική για να καλύψει άλλους παράγοντες που μπορεί να χαρακτηρίζουν παραλλαγές δεδομένων στον Ιστό.

Η εργασία μας έχει επηρεαστεί από την *Intensional HTML* [WBSY98, Bro98, Yil97, Bro98a], ή *IHTML* για συντομία, μια γλώσσα συγγραφής Ιστοσελίδων που επιτρέπει σε μια Ιστοσελίδα

<sup>2</sup> Οι λόγοι αυτοί γίνονται εμφανείς στο Κεφάλαιο 5, όπου διερευνούμε την χρήση του OEM και της Lorel για την αναπαράσταση και επερωτήση πληροφορίας που παρουσιάζει πολλαπλές εκφάνσεις.

<sup>3</sup> Μεταφρασμένο απόσπασμα από την ενότητα της on-line βοήθειας του Microsoft PowerPoint 2000 με τον τίτλο “About custom shows”.

να έχει διαφορετικές παραλλαγές και να προσαρμόζεται δυναμικά σε έναν δεδομένο **κόσμο** (που ορίζεται με την ανάθεση μοναδικών τιμών σε έναν αριθμό μεταβλητών που λέγονται **διαστάσεις**). Η HTML ακολουθεί μια εγγραφο-κεντρική προσέγγιση, και χρησιμοποιεί μια συγκεκριμένη μέθοδο ονομασίας για να κωδικοποιεί κόσμους στα ονόματα των φακέλων που περιέχουν παραλλαγές ενός εγγράφου. Η δική μας κατεύθυνση είναι διαφορετική: (a) η κεντρική μας έννοια είναι το **ερμηνευτικό περιβάλλον**, που αναπαριστά ένα σύνολο πιθανών κόσμων, ενώ οι διαστάσεις που χρησιμοποιούνται για να εκφραστεί το ερμηνευτικό περιβάλλον μπορούν να λάβουν πολλές τιμές και να συνδυαστούν σε συζεύξεις και διαζεύξεις, (b) τυποποιούμε την έννοια του ερμηνευτικού περιβάλλοντος και εισάγουμε πράξεις ανάμεσα από ερμηνευτικά περιβάλλοντα, και (c) εστιάζουμε σε μια δεδομένο-κεντρική προσέγγιση, που αποσκοπεί στην ενσωμάτωση του ερμηνευτικού περιβάλλοντος σε μοντέλα δεδομένων και γλώσσες επερωτήσεων για ημιδομημένα δεδομένα.

Στην διατριβή αυτή προτείνουμε τα **πολυδιάστατα ημιδομημένα δεδομένα (MSSD** για συντομία), τα οποία είναι ημιδομημένα δεδομένα που παρουσιάζουν διαφορετικές εκδοχές κάτω από διαφορετικά **ερμηνευτικά περιβάλλοντα**. Εκφράζουμε τα ερμηνευτικά περιβάλλοντα με έναν ευέλικτο και απλό τρόπο, αναθέτοντας τιμές σε μεταβλητές που ονομάζουμε **διαστάσεις**. Η κύρια διαφορά ανάμεσα από τα συμβατικά και τα πολυδιάστατα ημιδομημένα δεδομένα είναι η εισαγωγή των **προσδιοριστών περιβάλλοντος**. Οι προσδιοριστές περιβάλλοντος είναι συντακτικές κατασκευές που χρησιμοποιούνται για να συνοδεύουν τμήματα ημιδομημένων δεδομένων και να καθορίζουν σύνολα **κόσμων** κάτω από τους οποίους υφίστανται τα τμήματα αυτά. Ο κόσμος είναι μια εκδοχή της πραγματικότητας κάτω από την οποία τα δεδομένα αποκτούν μονοσήμαντη ερμηνεία, και ορίζεται αναθέτοντας μια μοναδική τιμή σε κάθε διάσταση.

Έτσι, είναι δυνατόν να έχουμε την ίδια στιγμή παραλλαγές της ίδιας πληροφοριακής οντότητας, και κάθε μια παραλλαγή να υφίσταται κάτω από διαφορετικό σύνολο κόσμων. Μια πληροφοριακή οντότητα που περιλαμβάνει έναν αριθμό παραλλαγών ονομάζεται **πολυδιάστατη οντότητα**, και οι παραλλαγές της ονομάζονται **έκφάνσεις** της οντότητας. Οι εκφάνσεις μιας πολυδιάστατης οντότητας μπορεί να διαφέρουν σε τιμή και / ή δομή, και μπορούν με την σειρά τους να είναι πολυδιάστατες οντότητες ή συμβατικές πληροφοριακές οντότητες. Κάθε έκφανση συνδέεται με ένα ερμηνευτικό περιβάλλον που ορίζει τις συνθήκες κάτω από τις οποίες η έκφανση γίνεται η **υφιστάμενη έκφανση** της πολυδιάστατης οντότητας.

Τα κύρια σημεία της συμβολής αυτής της διατριβής είναι:

1. Ορίζουμε έναν απλό και ευέλικτο τρόπο για να εκφράζουμε ερμηνευτικά περιβάλλοντα χρησιμοποιώντας **διαστάσεις**. Σύμφωνα με την προσέγγισή μας, το ερμηνευτικό περιβάλλον αντιπροσωπεύει μια σειρά από εναλλακτικούς **κόσμους**, σε κάθε έναν από τους οποίους οι πληροφορίες αποκτούν μια συγκεκριμένη ερμηνεία. Ορίζουμε πράξεις ανάμεσα από ερμηνευτικά περιβάλλοντα, και εξετάζουμε τις ιδιότητές τους σε βάθος.
2. Προτείνουμε ένα μοντέλο δεδομένων για MSSD που ονομάζεται **Πολυδιάστατος Γράφος Δεδομένων**, και ερευνούμε διεξοδικά τις ιδιότητές του. Το **Πολυδιάστατο OEM (MOEM)** είναι μια ειδική περίπτωση Πολυδιάστατου Γράφου Δεδομένων, και μπορεί να θεωρηθεί σαν μια συνένωση από συμβατικούς OEM γράφους που υφίστανται κάτω από διαφορετικούς κόσμους. Οι συγκεκριμένοι OEM γράφοι είναι δυνατόν να ανακτηθούν μέσω μιας ευέλικτης διαδικασίας προσαρμογής της πληροφορίας που λέγεται **αναγωγή σε OEM**, και η οποία μετασχηματίζει τον MOEM στον συμβατικό OEM γράφο που υφίσταται κάτω από έναν δεδομένο κόσμο.
3. Ορίζουμε την **Πολυδιάστατη Γλώσσα Επερωτήσεων (MQL)**, μια γλώσσα επερωτήσεων για Πολυδιάστατους Γράφους Δεδομένων και MOEM γράφους που

έχει σαν κεντρική έννοια το ερμηνευτικό περιβάλλον, και μπορεί να εκφράσει με κοινό τρόπο *επερωτήσεις οδηγούμενες από περιβάλλοντα*. Το μοντέλο MOEM μαζί με την γλώσσα επερωτήσεων MQL επιτρέπουν τη διατύπωση *δια-κοσμικών επερωτήσεων*, οι οποίες δεν μπορούν να εκφραστούν με συμβατικά μοντέλα δεδομένων και συμβατικές γλώσσες επερωτήσεων. Η MQL έχει υλοποιηθεί χρησιμοποιώντας σαν βάση το σύστημα LORE, και αυτή η υλοποίηση δίνει την ευκαιρία για σύγκριση με «ισοδύναμες» επερωτήσεις στην γλώσσα Lorel.

4. Αξιολογούμε το παραπάνω πλαίσιο που αναπτύξαμε για τα MSSD εφαρμόζοντάς το σε ένα κλασσικό πρόβλημα των βάσεων δεδομένων: χρησιμοποιούμε το μοντέλο MOEM για να αναπαραστήσουμε το ιστορικό βάσεων δεδομένων OEM, και στην συνέχεια την MQL για να θέσουμε επερωτήσεις στο ιστορικό αυτό. Το MOEM και η MQL αποδείχθηκαν ικανά όχι μόνο για να αναπαραστήσουν και να επερωτήσουν το ιστορικό βάσεων OEM, αλλά και το ιστορικό βάσεων MOEM.
5. Σαν παράδειγμα του πώς οι έννοιες που εισάγουμε μπορούν να επεκταθούν σε άλλες κατευθύνσεις, παραθέτουμε την **Πολυδιάστατη XML (MXML)**, μια εκδοχή της XML που ενσωματώνει ερμηνευτικά περιβάλλοντα. Για την παροχή και παρουσίαση πληροφορίας που ενέχει πολλαπλές εκφάνσεις προτείνουμε το **πολυδιάστατο παράδειγμα**, και επιδεικνύουμε μια σχετική πρότυπη υλοποίηση.
6. Παρουσιάζουμε ένα υλοποιημένο σύστημα για τη διαχείριση MSSD. Το σύστημα αυτό εμπεριέχει το υποσύστημα *MSSDesigner*, ένα γραφικό περιβάλλον για την εύκολη δημιουργία Πολυδιάστατων Γράφων Δεδομένων, καθώς και για την εκτέλεση επερωτήσεων διατυπωμένων στην MQL.

Ο ρόλος του ερμηνευτικού περιβάλλοντος στην διαχείριση της πολυμορφίας των δεδομένων στο σημερινό παγκοσμιοποιημένο περιβάλλον προσελκύει αυξανόμενη προσοχή, ιδίως στα πλαίσια του *ubiquitous computing* και του *mobile computing* [CK00]. Πιστεύουμε ότι τα MSSD αντιμετωπίζουν ένα σημαντικό πρόβλημα των βάσεων δεδομένων στον Ιστό, και ανοίγουν ενδιαφέρουσες κατευθύνσεις για περαιτέρω έρευνα. Επιπλέον, τα MSSD έχουν την δυνατότητα να εφαρμοστούν πρακτικά σε διάφορα γνωστικά πεδία. Για παράδειγμα, η πλατφόρμα δημοσίευσης πληροφορίας στον Ιστό *OMSwE* [NP03, NP03a] βασίζεται σε ένα Αντικειμενοστρεφές Σύστημα Διαχείρισης Βάσης Δεδομένων, το οποίο έχει επεκταθεί για να υποστηρίξει ένα ευέλικτο, ανεξάρτητο από το πεδίο εφαρμογής μοντέλο για παροχή πληροφορίας, όπου το ερμηνευτικό περιβάλλον παίζει κεντρικό ρόλο. Έτσι, μια ενδιαφέρουσα κατεύθυνση είναι η χρήση εννοιών των MSSD σε προβλήματα από διαφορετικά πεδία, όπως: στην παροχή προσωποποιημένων πληροφοριών και υπηρεσιών, προσαρμόζοντάς τις στα γνωρίσματα των καταναλωτών· στην ενοποίηση πληροφορίας, για την μοντελοποίηση αντικειμένων που η τιμή ή η δομή αλλάζουν ανάλογα με την πηγή· στις ψηφιακές βιβλιοθήκες, για την αναπαράσταση μεταπληροφορίας που ακολουθεί παρόμοιους μορφότυπους· στην αναπαράσταση γεωγραφικής πληροφορίας, όπου πιθανές διαστάσεις θα μπορούσαν να είναι η *κλίμακα* και το *θέμα*.

Τμήμα της δουλειάς που παρουσιάζεται στην διατριβή αυτή έχει δημοσιευτεί ή έχει υποβληθεί για δημοσίευση στα [SG02, SPES03, SGDZ02, SGDZ03, GSK01, GSK+01, MGS101a, MGS101, GS03, SGM00, SGR00, ZDSG02, EPS03].

## 1.2 ΤΑ ΚΕΦΑΛΑΙΑ ΤΗΣ ΔΙΑΤΡΙΒΗΣ

Η διατριβή έχει την ακόλουθη δομή. Στο Κεφάλαιο 2 υπενθυμίζουμε κάποια προκαταρκτική δουλειά πάνω σε ημιδομημένα δεδομένα και σε XML, επικεντρώνοντας σε ζητήματα που αναφέρονται από τα επόμενα κεφάλαια. Διατυπώνουμε τον ορισμό του OEM,

και δίνουμε παραδείγματα επερωτήσεων σε Lorel και UnQL. Αναφερόμαστε σύντομα στην XML, και σε συναφείς τεχνολογίες όπως το DTD, το XSL, και οι γλώσσες επερωτήσεων για XML.

Στο Κεφάλαιο 3 αναπτύσσουμε μια τυποποίηση για να αναπαριστούμε ερμηνευτικά περιβάλλοντα χρησιμοποιώντας μεταβλητές που ονομάζονται **διαστάσεις**. Ορίζουμε τους **προσδιοριστές περιβάλλοντος** σαν περιορισμούς στις δυνατές τιμές που μπορούν να λάβουν οι διαστάσεις. Δίνουμε νόημα στο ερμηνευτικό περιβάλλον δείχνοντας ότι μπορεί να ιδωθεί σαν ένα σύνολο πιθανών **κόσμων**, όπου ο κόσμος είναι μια εκδοχή της πραγματικότητας κάτω από την οποία οι πληροφορίες αποκτούν μονοσήμαντη ερμηνεία. Ορίζουμε πράξεις περιβάλλοντος, και διερευνούμε τις ιδιότητές τους.

Στο Κεφάλαιο 4 εισάγουμε τον **Πολυδιάστατο Γράφο Δεδομένων**, ένα μοντέλο δεδομένων γράφου που ενσωματώνει προσδιοριστές περιβάλλοντος και αντιμετωπίζει τις οντότητες πολλαπλών εκφάνσεων σαν πολίτες πρώτης κατηγορίας. Οι οντότητες πολλαπλών εκφάνσεων δύνανται να εκδηλώνουν διαφορετικές εκφάνσεις κάτω από διαφορετικούς κόσμους, και οι προσδιοριστές περιβάλλοντος χρησιμοποιούνται για να ορίζουν τους κόσμους κάτω από τους οποίους υφίσταται κάθε έκφανση. Διερευνούμε τις ιδιότητες του Πολυδιάστατου Γράφου Δεδομένων, και ορίζουμε την διαδικασία **αναγωγής σε OEM**, που αποσπά από έναν Πολυδιάστατο Γράφο Δεδομένων έναν συμβατικό γράφο OEM που υφίσταται κάτω από δεδομένο κόσμο. Βασισμένοι στις ιδιότητες του Πολυδιάστατου Γράφου Δεδομένων, ορίζουμε το **Πολυδιάστατο OEM (MOEM)** σαν μια ειδική περίπτωση Πολυδιάστατου Γράφου Δεδομένων που προέρχεται αποκλειστικά από την συνένωση συμβατικών OEM τα οποία υφίστανται κάτω από διάφορους κόσμους.

Στο Κεφάλαιο 5 παρουσιάζουμε την **Πολυδιάστατη Γλώσσα Επερωτήσεων (MQL)**, μια γλώσσα επερωτήσεων που υποστηρίζει σαν πρώτης κατηγορίας πολίτες τα ερμηνευτικά περιβάλλοντα και τις οντότητες πολλαπλών εκφάνσεων, και μπορεί να εκφράσει *επερωτήσεις οδηγούμενες από περιβάλλον*. Οι **εκφράσεις μονοπατιού περιβάλλοντος** επεκτείνουν τις εκφράσεις μονοπατιού ενσωματώνοντας **περιοριστές περιβάλλοντος**, που χρησιμοποιούνται για να θέσουν συνθήκες περιβάλλοντος πάνω στον γράφο της βάσης. Εκτός από τις εκφράσεις μονοπατιού περιβάλλοντος, η MQL χρησιμοποιεί δύο ειδικούς όρους για τον χειρισμό του ερμηνευτικού περιβάλλοντος: ο όρος *within* εκφράζει σύνθετες συνθήκες πάνω στο ερμηνευτικό περιβάλλον, και ο όρος *context* δημιουργεί νέες μεταβλητές περιβάλλοντος για να χρησιμοποιηθούν στην κατασκευή αποτελεσμάτων. Η πρότυπη υλοποίησή μας της MQL επιδεικνύει την σχέση μιας επερωτήσης MQL με μια ισοδύναμη επερωτήση Lorel, και δείχνει το όφελος του να αντιμετωπίζονται σαν πρώτης κατηγορίας πολίτες οι οντότητες πολλαπλών εκφάνσεων και το ερμηνευτικό περιβάλλον.

Στο Κεφάλαιο 6 αξιολογούμε την προσέγγισή μας, εφαρμόζοντας το πλαίσιο για MSSD που αναπτύξαμε σε ένα κλασσικό πρόβλημα των βάσεων δεδομένων: χρησιμοποιούμε το MOEM και την MQL για να αναπαραστήσουμε και να επερωτήσουμε ιστορικά βάσεων δεδομένων OEM. Εισάγουμε τις **λειτουργίες βασικών αλλαγών** του MOEM, και προδιαγράφουμε μια διαδικασία που κωδικοποιεί σε ένα MOEM το ιστορικό μιας βάσης δεδομένων OEM. Επιπλέον, δείχνουμε ότι χρονικά στιγμιότυπα της βάσης OEM μπορούν να ανακτηθούν από το MOEM μέσω της διαδικασίας της αναγωγής σε OEM, και παρουσιάζουμε το σχετικό υλοποιημένο σύστημα *OEM History*. Δίνουμε έναν αριθμό παραδειγμάτων επερωτήσεων MQL πάνω στο ιστορικό μιας βάσης δεδομένων OEM, και μέσα από αυτά επιβεβαιώνουμε την εκφραστικότητα της MQL. Ένα ενδιαφέρον σημείο είναι ότι το MOEM και η MQL μπορούν να αναπαραστήσουν και να επερωτήσουν όχι μόνο ιστορικά βάσεων OEM, αλλά επίσης και ιστορικά βάσεων MOEM.

Το Κεφάλαιο 7 εξυπηρετεί σαν παράδειγμα του πως οι έννοιες που εισάγουμε σε προηγούμενα κεφάλαια μπορούν να επεκταθούν και σε άλλες κατευθύνσεις, και προτείνει



την **Πολυδιάστατη XML (MXML)**, μια επέκταση της XML κατάλληλη για την αναπαράσταση δεδομένων που παρουσιάζουν διαφορετικές εκδοχές κάτω από διαφορετικά ερμηνευτικά περιβάλλοντα. Επιπλέον, προτείνουμε μια επέκταση του DTD που ονομάζεται **Πολυδιάστατο DTD (MDTD)**, το οποίο περιγράφει την λογική δομή των εγγράφων MXML. Παρουσιάζουμε ένα νέο παράδειγμα που λέγεται **πολυδιάστατο παράδειγμα** και χρησιμοποιεί **πολυδιάστατα XSL (MXSL)** φύλλα διαμόρφωσης για τον χειρισμό και την εμφάνιση δεδομένων στον Παγκόσμιο Ιστό που εξαρτώνται από ερμηνευτικά περιβάλλοντα και παρουσιάζουν πολλαπλές εκφάνσεις. Περιγράφουμε ένα σύστημα που υλοποιεί την βασική λειτουργικότητα του πολυδιάστατου παραδείγματος, και επιδεικνύει πως ο χρήστης μπορεί να αλληλεπιδράσει με ένα πολυδιάστατο έγγραφο και να δει διαφορετικές παραλλαγές του κάτω από διαφορετικούς κόσμους.

Στο Κεφάλαιο 8 παρουσιάζουμε ένα σύστημα για την διαχείριση MSSD που υλοποιεί τις περισσότερες βασικές έννοιες που εισήχθησαν στην διατριβή αυτή, και το οποίο μπορεί να χρησιμοποιηθεί για την διαδραστική δημιουργία, τον πειραματισμό, και την επερώτηση Πολυδιάστατων Γράφων Δεδομένων με την MQL.

Τέλος, το Κεφάλαιο 9 κλείνει την διατριβή ανακεφαλαιώνοντας την συμβολή μας, και αναφέροντας κατευθύνσεις για μελλοντική έρευνα αλλά και πιθανά πεδία εφαρμογών των MSSD.



## 2 ΠΡΟΚΑΤΑΡΚΤΙΚΑ

Η επίδραση του Παγκόσμιου Ιστού κατά την τελευταία δεκαετία ήταν καθοριστική. Σε σύντομο διάστημα μετά την εφεύρεσή του, ο Ιστός έγινε ένα μέσο ευρέως αποδεκτό για ανταλλαγή πληροφορίας, επικοινωνία, και εμπόριο σε παγκόσμια κλίμακα. Ο Ιστός συνεχίζει να εξελίσσεται, με πολλά υποσχόμενες τεχνολογίες όπως τα web services [Pap02], και με οράματα όπως το Semantic Web [SWEB] σύμφωνα με το οποίο τα δεδομένα του Ιστού δεν θα είναι απλώς αναγνώσιμα από μηχανές, αλλά οι μηχανές θα κατανοούν την σημασία των δεδομένων.

Από την αρχή, η τεχνολογία των βάσεων δεδομένων έπαιξε σημαντικό ρόλο στον Ιστό. Ένα ουσιαστικό μέρος του περιεχομένου του Ιστού είναι αποθηκευμένο σε σχεσιακές βάσεις δεδομένων, που χρησιμοποιούνται για να κατασκευάζουν δυναμικά σελίδες HTML [HTML] και για να υποστηρίζουν εφαρμογές ηλεκτρονικού εμπορίου. Εξαιτίας της ανάγκης για υποστήριξη από βάσεις δεδομένων, γρήγορα επινοήθηκαν τρόποι για την διασύνδεση των βάσεων με τον Ιστό [Mal98, SKV98, ZSC99]. Αλλά οι βάσεις δεδομένων περισσότερο από το να είναι απλά μια τεχνολογία που δρα στο παρασκήνιο. Η ερευνητική κοινότητα των βάσεων δεδομένων αναγνώρισε ταχέως [SSU95, SZ96, SZ97] ότι ο Ιστός, σαν μια τεράστια πηγή πληροφορίας που πρέπει να την προσπελαίνουν και να την διαχειρίζονται, θα μπορούσε να ωφεληθεί από την εμπειρία που έχει συσσωρευτεί στις βάσεις δεδομένων τις τελευταίες δεκαετίες. Έτσι, μοντέλα δεδομένων, γλώσσες επερωτήσεων, βελτιστοποίηση επερωτήσεων, και τεχνικές ταυτοχρονισμού θα πρέπει να προσαρμοστούν στις νέες απαιτήσεις που βάζει ο Ιστός. Όπως αναφέρεται στην αναφορά Asilomar για την έρευνα στις βάσεις δεδομένων [BBC+98], ο μακροπρόθεσμος στόχος που προτείνεται για την έρευνα στις βάσεις δεδομένων είναι:

«Η υπηρεσία χρήσης πληροφορίας: να γίνει εύκολο για οποιονδήποτε να αποθηκεύει, οργανώνει, προσπελαύνει, και αναλύει μέσω δικτύου το μεγαλύτερο μέρος της ανθρώπινης πληροφορίας.»

Συνεπώς, οι βάσεις δεδομένων συμπληρώνουν την *εγγραφο-κεντρική* καταγωγή του Ιστού με μια *δεδομένο-κεντρική* οπτική. Ακολουθώντας, αναφερόμαστε σύντομα σε περιπτώσεις όπου η τεχνολογία των βάσεων δεδομένων έχει εφαρμοστεί [FLM98] στον Ιστό. Στην συνέχεια συζητούμε τα ημιδομημένα δεδομένα, που εμπεριέχουν μοντέλα δεδομένων και γλώσσες επερωτήσεων για τον Ιστό όπως προτάθηκαν από την κοινότητα των βάσεων δεδομένων. Τέλος, αναφερόμαστε στην XML και σε συναφείς τεχνολογίες, οι οποίες, παρά την εγγραφο-κεντρική καταγωγή τους, έχουν υιοθετηθεί από την κοινότητα των βάσεων δεδομένων και αυτή την στιγμή αποτελούν ένα πολύ δραστήριο πεδίο έρευνας στις βάσεις δεδομένων.

### 2.1 ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΠΑΓΚΟΣΜΙΟΣ ΙΣΤΟΣ

Η επιτυχία του Ιστού μπορεί να αποδοθεί στο γεγονός ότι προσέφερε έναν απλό και στάνταρ τρόπο για ανταλλαγή πληροφορίας σε παγκόσμια κλίμακα. Εν συντομία, τα στοιχεία-κλειδιά του Ιστού είναι: (α) η HTML [HTML], μια γλώσσα επισημείωσης που δομεί το κείμενο σύμφωνα με την οπτική του παρουσίαση και που περιέχει συνδέσμους προς άλλα

έγγραφα καθώς και προς άλλα μέρη του ίδιου εγγράφου, (b) το URL [URL], ένας τρόπος προσδιορισμού πόρων στο διαδίκτυο, και (c) το HTTP [HTTP], ένα απλό πρωτόκολλο που επιτρέπει σε έναν πελάτη να συνδεθεί με έναν εξυπηρετητή και να ζητήσει κάποιον πόρο.

Σε αυτή την απλοποιημένη εικόνα του Ιστού, οι μονάδες πληροφορίας είναι έγγραφα HTML που προσδιορίζονται από URLs. Σύμφωνα με το παράδειγμα αυτό, η πληροφορία θεωρείται ότι δεν έχει δομή, και οι μηχανές αναζήτησης του Ιστού επιστρατεύουν τεχνικές από την περιοχή της ανάκτησης πληροφορίας (Information Retrieval) [GRGK97] προκειμένου να ανακαλύψουν έγγραφα σχετικά με κάποιο θέμα. Παρά το γεγονός ότι οι μηχανές αναζήτησης χρησιμοποιούν τεχνολογία και από τις βάσεις δεδομένων [BMPW98, YL96], ακολουθούν κυρίως την μεθοδολογία της ανάκτησης πληροφορίας, και αντιμετωπίζουν μια επερώτηση σαν ένα σύνολο φίλτρων βασισμένων σε λέξεις-κλειδιά. Η προσέγγιση αυτή προσφέρει ευελιξία και ευκολία χρήσης, δεν διαθέτει την ακρίβεια και την πληρότητα των βάσεων δεδομένων: οι μηχανές αναζήτησης επιστρέφουν συχνά άσχετα έγγραφα, ενώ συχνά δεν αναγνωρίζουν σχετικά έγγραφα σαν τέτοια.

Από την άλλη πλευρά, η *κουλτούρα των βάσεων δεδομένων* [ABS00] προτείνει την χρήση μοντέλων δεδομένων, όπως το διάγραμμα οντοτήτων-συσχετίσεων και τα σχήματα στις σχεσιακές βάσεις δεδομένων, προκειμένου να απεικονισθεί η δομή της πληροφορίας, μαζί με την χρήση γλωσσών επερώτησης που βασίζονται στα μοντέλα δεδομένων για να προσπελάσουν την πληροφορία. Ένα κύριο χαρακτηριστικό της οπτικής των βάσεων δεδομένων είναι ο διαχωρισμός ανάμεσα στην λογική όψη, που σχετίζεται με μοντέλα και επερωτήσεις, και την φυσική όψη, που σχετίζεται με την αποθήκευση και την αποδοτικότητα. Η σημαντική αυτή διάκριση σε αφαιρετικά επίπεδα δεν αναγνωρίζεται από την εγγραφο-κεντρική θεώρηση του Ιστού. Συνεπώς, είναι λογικό η δεδομένο-κεντρική και η εγγραφο-κεντρική θεώρηση του Ιστού να συγκλίνουν και να επωφεληθούν η μία από την άλλη.

Σε μια προσπάθεια να δοθεί μια εκφραστική και ομογενής διαπροσωπεία επερωτήσεων πάνω σε ετερογενείς πηγές πληροφορίας, αναπτύχθηκε ένας αριθμός συστημάτων ενοποίησης πληροφορίας. Παραδείγματα τέτοιων συστημάτων είναι το WHIRL [Coh98, Coh98a, Coh98b, CH98], το Information Manifold [KLSS95, LMSS95, LSK95], το TSIMMIS [PGW95, GPQ+97, CGH+94, GHI+95, PGU96], και το Garlic [CHS+95, HKWY96, HMN+97, RS97]. Οι βάσεις δεδομένων προσπάθησαν να προσαρμοστούν στον Ιστό ακολουθώντας μια πολυ-διαζωματική αρχιτεκτονική αντί για την κλασσική αρχιτεκτονική πελάτη-εξυπηρετητή. Στην πολυ-διαζωματική αυτή αρχιτεκτονική ο εξυπηρετητής είναι ο χώρος αποθήκευσης δεδομένων, ο πελάτης ερμηνεύει και παρουσιάζει τα δεδομένα, και ένα *ενδιάμεσο* διάζωμα χρησιμοποιεί μεσολαβητές [Wie92] για να ενοποιήσει δεδομένα από διάφορες πηγές και για να μετατρέψει δεδομένα στην κατάλληλη μορφή. Τα συστήματα ενοποίησης πληροφορίας με μεσολαβητές μπορούν να ταξινομηθούν [Ull97, GP98] ανάλογα με το πως ορίζεται το συνολικό σχήμα: (a) σαν συνένωση όψεων που περιγράφουν μία-μία τις πηγές, ή (b) χρησιμοποιώντας συλλογές και κατηγορήματα συνολικής εμβέλειας στα οποία αντιστοιχούνται οι τοπικές πηγές.

Η προσέγγιση του TSIMMIS χρησιμοποιεί ένα δίκτυο μεσολαβητών και μεταφραστών για να εξάγει πληροφορία και να δημιουργήσει μια ενοποιημένη όψη των υποκείμενων πηγών. Το μοντέλο δεδομένων που εισήγαγε το TSIMMIS ονομάζεται Μοντέλο Ανταλλαγής Αντικειμένων (Object Exchange Model), ή OEM, και η γλώσσα που χρησιμοποιείται για επερωτήσεις λέγεται Lorel [GPQ+97, AQM+97]. Το OEM και η Lorel χρησιμοποιούνται επίσης στο Lightweight Object Repository [MAG+97], ή LORE, ένα σύστημα διαχείρισης βάσης δεδομένων που αναπτύχθηκε ειδικά για ημιδομημένα δεδομένα.

Μια άλλη ερευνητική κατεύθυνση είναι η εφαρμογή μεθοδολογιών βάσεων δεδομένων για την διαχείριση τόπων του Ιστού. Μια μεθοδολογία για τον σχεδιασμό εφαρμογών υπερμέσων παρουσιάζεται στο [ISB95]. Το STRUDEL [FFK+98, FFLS97, FFLS98] είναι ένα σύστημα για την δημιουργία και διαχείριση τόπων του Ιστού, το οποίο αντιμετωπίζει

ξεχωριστά την οργάνωση των δεδομένων σε φυσικό επίπεδο, την όψη του τόπου σε λογικό επίπεδο, και θέματα παρουσίασης της πληροφορίας. Χρησιμοποιεί το OEM σαν μοντέλο δεδομένων, και την StruQL σαν γλώσσα επερωτήσεων, η οποία ορίζει με δηλωτικό τρόπο την δομή ενός τόπου. Άλλο ένα σύστημα για την ανάπτυξη τόπων του Ιστού είναι το ARANEOUS [AMM97, AMM97a, MMAC99], που προτείνει το μοντέλο δεδομένων ADM για την περιγραφή συλλογών υπερκειμένου, την γλώσσα ULIXES για τον ορισμό όψεων, και ενσωματώνει ένα σύνολο εργαλείων που αναλαμβάνουν την κατασκευή τόπων του Ιστού.

Μια σημαντική συνεισφορά αυτής της προσέγγισης των βάσεων δεδομένων με τον Ιστό αποτελεί η εργασία στα ημιδομημένα δεδομένα και την XML. Ακολουθώντας, επισκοπούμε ένα μέρος αυτής της εργασίας που αφορά σε μοντέλα δεδομένων και γλώσσες επερωτήσης για ημιδομημένα δεδομένα και XML, εστιάζοντας σε θέματα που πρόκειται να χρησιμεύσουν στα επόμενα κεφάλαια.

## 2.2 ΗΜΙΔΟΜΗΜΕΝΑ ΔΕΔΟΜΕΝΑ

Στις παραδοσιακές βάσεις δεδομένων, η πληροφορία είναι αυστηρά δομημένη και υπακούει σε ένα σταθερό σχήμα που έχει οριστεί εκ των προτέρων. Τα συστήματα διαχείρισης βάσεων δεδομένων χρησιμοποιούν το σχήμα για να αποθηκεύουν και να ευρετηριάζουν δεδομένα, και για να διεκπεραιώνουν επερωτήσεις και ενημερώσεις. Από την άλλη, τα δεδομένα του Ιστού μπορεί να μην έχουν κανονικότητα στην δομή και να μην υπακούν σε κάποιο ξεχωριστά ιστάμενο σχήμα. Ο όρος ημιδομημένα δεδομένα [Suc98, Bun97, Suc97], ή SSD, εισήχθη για να περιγράψει πληροφορία που είναι λιγότερο δομημένη από τα σχεσιακά ή τα αντικειμενοστρεφή δεδομένα, αλλά που δεν είναι εντελώς αδόμητη. Παραδείγματα ημιδομημένων δεδομένων αποτελούν φάκελοι σε LaTeX και BibTex, καθώς και φάκελοι σε RTF, HTML, και XML.

Τα ημιδομημένα δεδομένα μπορεί να παρουσιάζουν τις εξής ιδιομορφίες.

- *Παραλλαγές στην δομή:* Η ανομοιογένεια των δεδομένων μπορεί να εκδηλωθεί σαν πεδία που λείπουν (η διεύθυνση ενός ατόμου λείπει), επαναλαμβανόμενα πεδία (δύο νούμερα τηλεφώνου για το ίδιο άτομο), ή διαφοροποιήσεις στην αναπαράσταση (κάποιες τιμές σε ευρώ ενώ άλλες σε δολάρια). Επιπλέον, τα δεδομένα μπορούν να είναι εμφωλιασμένα σε οποιοδήποτε βάθος, πράγμα που κάνει δύσκολη την χρήση του σχεσιακού μοντέλου.
- *Η δομή ίσως είναι μερική:* Τμήματα των δεδομένων μπορεί να μην έχουν καμία δομή (π.χ. εικόνες), ενώ άλλα τμήματα μπορεί να έχουν ελάχιστη δομή (π.χ. σκέτο κείμενο).
- *Οι τύποι είναι μόνο ενδεικτικοί:* Τα ημιδομημένα δεδομένα δεν υιοθετούν την αυστηρή πολιτική τύπων που συναντάμε στις παραδοσιακές βάσεις δεδομένων.
- *Ενσωματωμένο σχήμα:* Στα ημιδομημένα δεδομένα το σχήμα δεν υπάρχει ξεχωριστά από τα δεδομένα όπως συμβαίνει στις παραδοσιακές βάσεις δεδομένων. Με άλλα λόγια, ο τύπος των αντικειμένων δεν υπάρχει έξω από τα στιγμιότυπα των αντικειμένων. Για τον λόγο αυτό τα ημιδομημένα δεδομένα λέγονται *α-σχήμα* και *αυτοπεριγραφόμενα* δεδομένα, με την έννοια ότι το σχήμα υπάρχει ενσωματωμένο στην ίδια την πληροφορία.
- *Το σχήμα είναι μεγάλο και αλλάζει ταχέως:* Όσο περισσότερες οι παραλλαγές στην δομή, τόσο μεγαλύτερο γίνεται το σχήμα. Επιπρόσθετα, τα δεδομένα του Ιστού αλλάζουν συχνά και απρόβλεπτα· συνεπώς είναι πιθανό ότι και τα σχήματα εξελίσσονται με γρήγορο ρυθμό.

- *A-posteriori οδηγός δεδομένων*: Αντίθετα με τις παραδοσιακές βάσεις δεδομένων που γεμίζουν με βάση κάποιο προϋπάρχον σχήμα, το σχήμα στα ημιδομημένα δεδομένα εξάγεται εκ των υστέρων προκειμένου να διευκολυνθεί η προσπέλαση της πληροφορίας.

## 2.2.1 Μοντέλα δεδομένων για SSD

Τα μοντέλα δεδομένων για SSD βασίζονται σε κατευθυνόμενους γράφους, για να αντιμετωπιστεί η μη κανονική δομή και ο βαθύς εμφωλιασμός των ημιδομημένων δεδομένων. Ένα διαδομένο μοντέλο για SSD είναι το Μοντέλο Ανταλλαγής Αντικειμένων (Object Exchange Model), ή OEM, παραλλαγές του οποίου έχουν χρησιμοποιηθεί σε διάφορες περιπτώσεις. Το OEM εισήχθη αρχικά από το TSIMMIS [PGW95, GPQ+97] σαν ένας γράφος διασυνδεδεμένων αντικειμένων. Ένα αντικείμενο του OEM ορίζεται στο TSIMMIS σαν μια τετράδα που περιλαμβάνει το αναγνωριστικό αντικειμένου (oid), μια ετικέτα, τον τύπο, και μια τιμή. Ο τύπος μπορεί να είναι είτε *σύνολο*, είτε κάποιος ατομικός τύπος (στοιχειοσειρά, ακέραιος, κτλ.). Αντίστοιχα, η τιμή μπορεί να είναι είτε ένα σύνολο από αναγνωριστικά, ή μια ατομική τιμή. Συνεπώς, αφού οι κόμβοι στο μοντέλο γράφου αντιπροσωπεύουν αντικείμενα, ένα αντικείμενο του τύπου *σύνολο* δείχνει μέσω κατευθυνόμενων ακμών στα υπο-αντικείμενα που περιέχονται στο σύνολο που είναι η τιμή του.

Η παραλλαγή του OEM που χρησιμοποιείται στο TSIMMIS επισυνάπτει ετικέτες στους κόμβους (στα αντικείμενα). Σε άλλες παραλλαγές του OEM που χρησιμοποιούνται στο STRUDEL [FFLS97] και στο LORE [MAG+97, AQM+97] οι ετικέτες επισυνάπτονται στις ακμές και όχι στους κόμβους. Εάν οι ετικέτες επισυναφθούν σε κόμβους τότε αποτελούν ιδιότητα των αντικειμένων· από την άλλη, αν οι ετικέτες επισυναφθούν σε ακμές αποτελούν ιδιότητα της σχέσης ανάμεσα από δύο αντικείμενα. Το να επισυνάπτουμε ετικέτες σε ακμές μας επιτρέπει να «βλέπουμε» τον ίδιο κόμβο μέσα από διαφορετικά ονόματα. Μια παραλλαγή του OEM με ετικέτες στις ακμές είναι και το μοντέλο δεδομένων της UnQL [BDS95, BDHS96, BFS00], μιας γλώσσας επερωτήσεων για «αδόμητα» δεδομένα. Στην συγκεκριμένη παραλλαγή η έννοια της ισότητας μεταξύ αντικειμένων βασίζεται στην αμφιεξομοίωση αντί για τον ισομορφισμό.

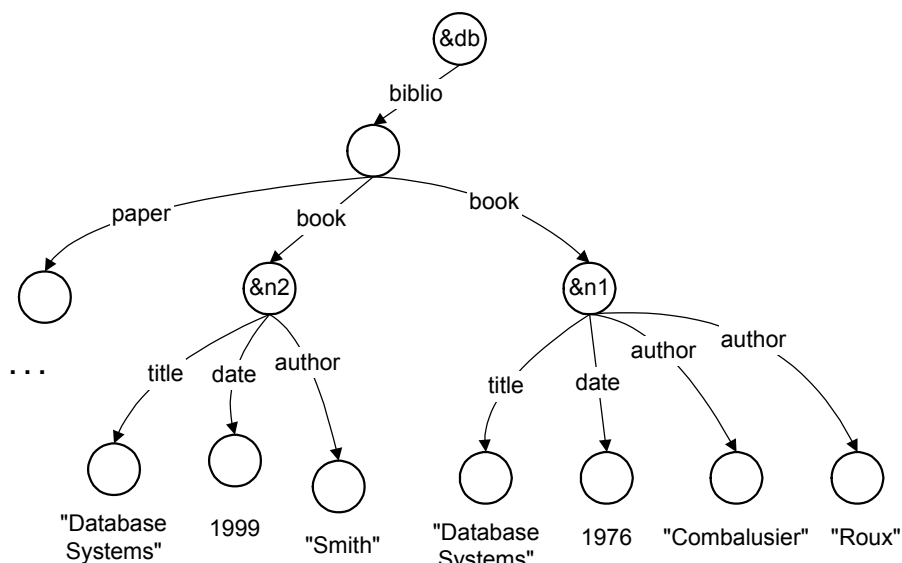
Στο πλαίσιο αυτής της διατριβής θα χρησιμοποιήσουμε την παραλλαγή του OEM με ετικέτες στις ακμές. Συγκεκριμένα, θα χρησιμοποιήσουμε τον αυστηρό ορισμό του OEM που δίνεται στο [Suc98].

### Ορισμός 2.1

*Ένας γράφος OEM είναι μια τετράδα  $(V, E, r, \nu)$ , όπου το σύνολο των κόμβων  $V$  χωρίζεται σε σύνθετους και σε ατομικούς κόμβους  $V = V_c \cup V_a$ , οι ακμές είναι το σύνολο  $E \subseteq (V_c \times L \times V)$ ,  $r \in V$  είναι η ρίζα με την ιδιότητα ότι κάθε κόμβος στο  $V$  είναι προσβάσιμος από τον  $r$ , και η συνάρτηση  $\nu : V_a \rightarrow A$  αναθέτει τιμές σε ατομικά αντικείμενα·  $A$  είναι το σύνολο όλων των ατομικών τιμών, και  $L$  είναι το σύνολο όλων των ετικετών.*

Ο ορισμός είναι ισοδύναμος με έναν λίγο διαφορετικό ορισμό για την βάση δεδομένων OEM που παρουσιάζεται στο [AQM+97]. Ένα παράδειγμα OEM από το [ABS00] δίνεται στην Εικόνα 2.1.

Εικόνα 2.1: Τμήμα μιας βιβλιογραφικής OEM βάσης δεδομένων.



Οι κόμβοι του OEM μπορούν να έχουν αναγνωριστικά αντικειμένων (oids): στην Εικόνα 2.1, αυτά είναι τα db, n1, και n2. Λόγω σύμβασης, τα αναγνωριστικά στο OEM αρχίζουν με τον χαρακτήρα &. Παρά το ότι το OEM στην Εικόνα 2.1 είναι δένδρο, στην γενική περίπτωση ένα OEM είναι γράφος. Συγκεκριμένα, το OEM είναι ένας κατευθυνόμενος πολυγράφος με ρίζα που έχει ετικέτες και φύλλα με τιμές.

Το OEM μπορεί να αναπαρασταθεί σαν κείμενο με διάφορους τρόπους [AQM+97, BFS00]. Υιοθετώντας τον μορφότυπο των εκφράσεων *ssd-expressions*, που εμφανίζονται στα [BFS00, ABS00, Suc98], το OEM στην Εικόνα 2.1 αναπαρίσταται όπως φαίνεται στο Παράδειγμα 2.1.

#### Παράδειγμα 2.1

```
&db {biblio: {book: &n1 {author: "Roux",
                        author: "Combalusier",
                        date: 1976,
                        title: "Database Systems"
                      },
            book: &n2 {author: "Smith",
                        date: 1999,
                        title: "Database Systems"
                      },
            paper: {...}
          }
}
```

◆

Κάθε αντικείμενο περικλείεται από αγκύλες, ενώ μπορεί προαιρετικά να προηγείται το αναγνωριστικό του αντικειμένου. Το συντακτικό των εκφράσεων *ssd-expressions* ορίζει αυστηρά [ABS00] ο Πίνακας 2.1 σε Extended Backus-Naur Form [EBNF], ή EBNF για συντομία. Τα σύμβολα που μπορούν να οριστούν από κάποια ομαλή έκφραση αρχίζουν με κεφαλαίο γράμμα (παράδειγμα: oid), ενώ τα σύμβολα που ορίζονται σε EBNF αρχίζουν με μικρό γράμμα (παράδειγμα: *ssd-expr*).

**Πίνακας 2.1: Σύνταξη των εκφράσεων `ssid-expression`.**

```

ssid-expr ::= value | Oid value | Oid
value ::= AtomicValue | "{" complexValue "}"
complexValue ::= LabelName ":" ssid-expr ("," complexValue)?

```

Εκτός από το OEM, έχουν προταθεί και μερικά άλλα μοντέλα γράφου για ημιδομημένα δεδομένα. Μια επέκταση του OEM που καλείται Delta OEM, ή DOEM, εισάγεται στο [CAW99], και αντιμετωπίζει το πρόβλημα της αναπαράστασης του ιστορικού των αλλαγών σε βάσεις δεδομένων OEM. Στο Κεφάλαιο 6, όπου ερευνούμε ενδελεχώς το παραπάνω πρόβλημα, συζητούμε περαιτέρω το [CAW99] και άλλες παρόμοιες προσεγγίσεις, και τις συγκρίνουμε με την δική μας προσέγγιση. Ένα μοντέλο για ημιδομημένα δεδομένα που διαφοροποιείται από το OEM έχει προταθεί στο [BDT98], όπου οι ετικέτες των ακμών είναι οι ίδιες κομμάτια ημιδομημένων δεδομένων. Στο [DBI99], προτάθηκε ένα επεκτάσιμο μοντέλο ημιδομημένων δεδομένων που χρησιμοποιεί σαν ετικέτες ακμών σύνολα ιδιοτήτων της μορφής `όνομα_ιδιότητας: τιμή_ιδιότητας`. Επισυνάπτοντας ιδιότητες κατά βούληση, ο γράφος εμπλουτίζεται με μεταπληροφορία. Διαφορετικές ιδιότητες μπορεί να έχουν άλλη σημασιολογία, πράγμα που οδηγεί σε ένα πολύ γενικό μοντέλο. Η επέκταση στο OEM που προτείνουμε εμείς παρουσιάζεται λεπτομερώς στο Κεφάλαιο 4.

## 2.2.2 Γλώσσες επερωτήσεων για SSD

Ένας αριθμός γλωσσών έχει προταθεί για την επερώτηση δεδομένων του Ιστού [FLM98, Suc98, Abi97], πριν ακόμη από τις γλώσσες επερώτησης για ημιδομημένα δεδομένα. Οι γλώσσες αυτές μπορούν να χωριστούν σε δύο κατηγορίες [FLM98]: γλώσσες επερώτησης *πρώτης γενιάς*, και γλώσσες επερώτησης *δεύτερης γενιάς*.

Το χαρακτηριστικό των γλωσσών επερώτησης πρώτης γενιάς είναι ότι συνδυάζουν επερωτήσεις βασισμένες στο περιεχόμενο με επερωτήσεις βασισμένες στην δομή. Η αναζήτηση που βασίζεται στο περιεχόμενο προϋποθέτει τον ορισμό *προτύπων κειμένου* που συγκρίνονται με το περιεχόμενο των εγγράφων, ενώ η αναζήτηση που βασίζεται στην δομή προϋποθέτει τον ορισμό *προτύπων γράφου* που περιγράφουν υπερσυνδέσμους ανάμεσα από έγγραφα. Στις γλώσσες επερώτησης Ιστού πρώτης γενιάς συμπεριλαμβάνονται η WebSQL [MMM96, MMM97], η W3QL [KS95], και η WebLog [LSS96]. Το παρακάτω παράδειγμα WebSQL βρίσκει όλα τα ζεύγη εγγράφων όπου το πρώτο βρίσκεται στον τρέχοντα τόπο και περιέχει την λέξη «database», και το δεύτερο βρίσκεται σε κάποιον απομακρυσμένο τόπο και είναι προσβάσιμο από το πρώτο μέσω ενός συνδέσμου, του οποίου η ετικέτα επιστρέφεται επίσης σαν μέρος των αποτελεσμάτων.

```

SELECT d.url, e.url, a.label
FROM Document d SUCH THAT
    "www.mysite.start" -> * d,
    d MENTIONS "database",
Document e SUCH THAT
    d => e,
Anchor a SUCH THAT
    a.base = d.url
WHERE a.href = e.url

```

Η WebSQL μοντελοποιεί τον Ιστό σαν μια σχεσιακή βάση δεδομένων με δύο εικονικές σχέσεις: *Έγγραφο* και *Σύνδεσμος*. Για κάθε έγγραφο και κάθε σύνδεσμο υπάρχει μια καταχώρηση στην αντίστοιχη σχέση. Η πλοήγηση αρχίζει από κάποιο γνωστό URL, και



χρησιμοποιείται το σύμβολο  $\rightarrow$  για να δηλωθεί πλοήγηση μέσα στον ίδιο εξυπηρετητή, ενώ το σύμβολο  $\Rightarrow$  για να δηλωθεί πλοήγηση σε κάποιον διαφορετικό εξυπηρετητή.

Οι γλώσσες επερώτησης Ιστού δεύτερης γενιάς είναι γλώσσες που χειρίζονται τα δεδομένα του Ιστού, με την έννοια ότι προσπελαύνουν την εσωτερική δομή των αντικειμένων στον Ιστό και μπορούν να δημιουργήσουν νέες σύνθετες δομές σαν το αποτέλεσμα μιας επερώτησης. Η οπτική που έχουν για τα δεδομένα του Ιστού είναι πολύ κοντά στην προσέγγιση που ακολουθούν τα μοντέλα των ημιδομημένων δεδομένων. Παραδείγματα τέτοιων γλωσσών είναι η StruQL [FFLS97, FFLS98, FFK+98], η FLORID [HLLS97], και η WebOQL [AM98]. Η StruQL είναι η γλώσσα επερώτησης του συστήματος διαχείρισης τόπων του Ιστού STRUDEL, και χρησιμοποιεί έναν όρο για να δεσμεύει μεταβλητές σε τμήματα του γράφου εισόδου που απαρτίζει την βάση δεδομένων, και έναν αριθμό επιπλέον όρων που χρησιμοποιούν συναρτήσεις Skolem για να κατασκευάσουν έναν νέο γράφο σαν αποτέλεσμα. Η FLORID είναι μια δηλωτική γλώσσα που βασίζεται στην F-logic, που είναι μια τυποποίηση για αντικειμενοστρεφή μοντέλα. Η WebOQL εστιάζει σε λειτουργίες που αλλάζουν την δομή των γράφων, και υποστηρίζει *Ιστούς* σαν ένα τύπο δεδομένων, ο οποίος επιτρέπει την μοντελοποίηση συνόλων από συσχετιζόμενα υπερδένδρα.

Οι γλώσσες ημιδομημένων δεδομένων με την μεγαλύτερη επίδραση είναι η UnQL [BFS00, BDHS96, BDS95] και η Lorel [AQM+97, GPQ+97, GMW99]. Η UnQL χρησιμοποιεί πρότυπα που μοιάζουν με τις εκφράσεις *ssd-expressions* για να δεσμεύσει μεταβλητές σε κόμβους στον όρο *where*, αλλά επίσης και για να κατασκευάσει τον γράφο του αποτελέσματος στον όρο *select*:

```
select {paper: {title: X, date: Y}}
where {biblio.paper: {title: X, date: Y}} in db,
      Y > 1995
```

Αποτιμώμενη στην βάση δεδομένων που απεικονίζεται στην Εικόνα 2.1, η παραπάνω επερώτηση επιστρέφει τον τίτλο και την ημερομηνία δημοσίευσης των εργασιών (papers) που δημοσιεύτηκαν μετά το 1995. Εκτός από τις κανονικές μεταβλητές που δεσμεύονται σε κόμβους, η UnQL χρησιμοποιεί *μεταβλητές ετικέτας* που δεσμεύονται σε ετικέτες ακμών· η χρήση των μεταβλητών ετικέτας καθιστά δυνατή την μετατροπή μεταπληροφορίας της βάσης δεδομένων σε κανονικά δεδομένα στο αποτέλεσμα. Οι επερωτήσεις σε UnQL μεταφράζονται σε *δομική αναδρομή*, μια κατασκευή που θυμίζει τα πρότυπα στις συναρτησιακές γλώσσες προγραμματισμού και που επιτρέπει την έκφραση επερωτήσεων αλλά και μετασχηματισμών με ενιαία τυποποίηση.

Η Lorel επικεντρώνει στην χρήση εκφραστικών *εκφράσεων μονοπατιού* και *εξαναγκασμών* στους τύπους, προκειμένου να διευκολύνει τον σχηματισμό επερωτήσεων και να κάνει την γλώσσα απλή στην χρήση. Η παρακάτω επερώτηση επιστρέφει τον τίτλο και τους συγγραφείς βιβλίων (books) ή εργασιών (papers) που δημοσιεύτηκαν μετά το 1972 και που περιέχουν την λέξη «Database» στον τίτλο.

```
select X.author, X.title
from db.biblio(.book | .paper) X
where X.date > 1972
and X.title grep "Database"
```

Αποτιμώμενη στην βάση δεδομένων που απεικονίζεται στην Εικόνα 2.1, η επερώτηση αυτή επιστρέφει μια *απάντηση* που ισοδυναμεί με την παρακάτω έκφραση *ssd-expression*:

```
{answer: {book: {author: "Roux",
                 author: "Combalusier",
                 title: "Database Systems"}},
}
```

```

    book: {author: "Smith",
           title: "Database Systems"
          },
    paper: {...}
  }
}

```

Το παράδειγμα αυτό επιδεικνύει έναν αριθμό χαρακτηριστικών της Lorel. Καταρχήν, η έκφραση μονοπατιού στον όρο `from` δανείζεται το συντακτικό των ομαλών εκφράσεων [Fñ97] για να επιστρέψει και βιβλία και εργασίες από την βάση δεδομένων. Οι κόμβοι που δεσμεύονται στην μεταβλητή `x` στον όρο `from` φιλτράρονται στον όρο `where`: οι εκφράσεις μονοπατιού στο `where` υπονοούν έναν υπαρξιακό προσδιορισμό της μορφής «υπάρχει στο `x` μια ημερομηνία (`date`) που είναι μεγαλύτερη από 1972». Στην συνέχεια, εκφράσεις μονοπατιού στον όρο `select` υπονοούν εμφωλιασμένες επερωτήσεις, που στην περίπτωση του `x.author` προκαλεί την εμφάνιση όλων των συγγραφέων (`author`) ενός βιβλίου κάτω από ένα και μοναδικό αντίστοιχο αντικείμενο στα αποτελέσματα. Η ετικέτα `answer` είναι η εξ ορισμού ετικέτα της Lorel για τα αποτελέσματα, ενώ οι ετικέτες `book` και `paper` αποφασίζονται κατά τον χρόνο εκτέλεσης, με βάση το μονοπάτι δεδομένων που έχει βρεθεί να ταιριάζει.

Στο [ABS00] σκιαγραφείται μια «γλώσσα πυρήνα» για ημιδομημένα δεδομένα, η οποία συνδυάζει χαρακτηριστικά και από την UnQL και από την Lorel. Μαζί με την Lorel και την UnQL, χρησιμοποιούμε αυτή την γλώσσα πυρήνα σαν βάση για να δομήσουμε την **Πολυδιάστατη Γλώσσα Επερωτήσεων** στο Κεφάλαιο 5.

## 2.3 XML

Ένας επιπλέον τρόπος για την αναπαράσταση ημιδομημένων δεδομένων είναι η eXtensible Markup Language [XML, Wa97, Cha99], ή XML. Η XML προτάθηκε από το World Wide Web Consortium [W3C], ή W3C, σαν ένα ευέλικτο και προσαρμόσιμο μέσο ανταλλαγής πληροφορίας στον Ιστό. Σε αντίθεση με την Hypertext Markup Language [HTML], ή HTML, όπου τα στοιχεία και τα γνωρίσματα είναι προκαθορισμένα και καταπιάνονται με την παρουσίαση της πληροφορίας, η XML μπορεί να ορίσει νέα στοιχεία και γνωρίσματα κατά βούληση, και έτσι να προσαρμοστεί σε διαφορετικά γνωστικά πεδία. Επιπρόσθετα, η XML εστιάζει στην δομή της πληροφορίας, αφήνοντας την αντιμετώπιση θεμάτων παρουσίασης σε επόμενα βήματα. Η XML λύνει το πρόβλημα της ad-hoc συντακτικής ανάλυσης δεδομένων στον Ιστό, και προσφέρει μια πλατφόρμα ανάπτυξης τυποποιημένων εργαλείων για την εξαγωγή από έγγραφα του Ιστού των δεδομένων που ενδιαφέρουν.

Στην πραγματικότητα η XML είναι υποσύνολο της Standard Generalized Markup Language [SGML], ή SGML, το οποίο σχεδιάστηκε για να μεταδίδεται, αν λαμβάνεται, και να τυγχάνει επεξεργασίας μέσω του Ιστού. Αυτή η εγγραφο-κεντρική οπτική, αντιλαμβάνεται την XML σαν μια αναπαράσταση σε φυσικό επίπεδο (μορφότυπος δεδομένων). Αντίθετα, η δεδομενο-κεντρική οπτική βλέπει την XML σαν μια αναπαράσταση πληροφορίας σε λογικό επίπεδο, που ενσωματώνει μεταπληροφορία (σχήμα) στα δεδομένα. Από αυτή την άποψη, η έμφαση είναι στην επερώτηση δεδομένων που αναπαριστώνται σε XML. Η XML προσφέρει μια κατάλληλη υποδομή για την σύγκλιση της εγγραφο-κεντρικής και της δεδομενο-κεντρικής οπτικής του Ιστού [ABS00].

### 2.3.1 Συντακτικό της XML και DTD

Κάθε έγγραφο XML έχει μια *λογική δομή* και μια *φυσική δομή*. Από φυσική πλευρά, το έγγραφο συντίθεται από μονάδες που καλούνται *οντότητες*. Μια οντότητα μπορεί να αναφέρεται σε άλλες οντότητες με τρόπο ώστε να προκαλεί την ενσωμάτωσή τους στο έγγραφο. Ένα έγγραφο αρχίζει με μια «ρίζα» ή αλλιώς *οντότητα εγγράφου*. Οι οντότητες περιέχουν *γραμματικά δεδομένα* ή *μη γραμματικά δεδομένα*. Τα γραμματικά δεδομένα συντίθενται από χαρακτήρες, κάποιοι από τους οποίους σχηματίζουν *δεδομένα χαρακτήρων*, και κάποιοι άλλοι σχηματίζουν *επισημειώσεις*. Οι επισημειώσεις κωδικοποιούν μια περιγραφή για την διάταξη αποθήκευσης καθώς και για την λογική δομή του εγγράφου. Η λογική δομή περιλαμβάνει *δηλώσεις, στοιχεία, σχόλια, αναφορές χαρακτήρων, και οδηγίες επεξεργασίας*. Ακολουθώντας την δεδομένο-κεντρική προσέγγιση, σε ότι ακολουθεί επικεντρωνόμαστε στην λογική δομή των εγγράφων XML, και ειδικότερα στις δηλώσεις και τα στοιχεία. Στο Παράδειγμα 2.2 φαίνεται τμήμα ενός εγγράφου XML που αναπαριστά το OEM που απεικονίζεται στην Εικόνα 2.1.

#### Παράδειγμα 2.2

```
<database id="db" >
  <biblio>
    <book id="n1" >
      <author>Roux</author>
      <author>Combalusier</author>
      <date>1976</date>
      <title>Database Systems</title>
    </book>
    <book id="n2" >
      <author>Smith</author>
      <date>1999</date>
      <title>Database Systems</title>
    </book>
    <paper>...</paper>
  </biblio>
</database>
```



Ένα έγγραφο XML περιέχει ένα ή περισσότερα *στοιχεία*, οριοθετημένα από *κάρτες-αρχής* και *κάρτες-τέλους*, ή από *κάρτες-άδειου-στοιχείου* για *άδεια στοιχεία*. Το *περιεχόμενο* ενός στοιχείου ορίζεται σαν οι χαρακτήρες ανάμεσα από την κάρτα-αρχής και την κάρτα-τέλους. Κάθε στοιχείο έχει έναν *τύπο*, που ταυτίζεται με το όνομα του στοιχείου, και ένα σύνολο από προσδιορισμούς *γνωρισμάτων*. Κάθε προσδιορισμός γνωρίσματος έχει ένα όνομα και μια τιμή. Μια *δήλωση στοιχείου* βάζει περιορισμούς στον τύπο και στο περιεχόμενο του στοιχείου. Οι προκαθορισμένοι τύποι είναι: "EMPTY", "ANY", mixed, και children. Στην περίπτωση του children, ο τύπος του στοιχείου έχει *περιεχόμενο στοιχείου*, που σημαίνει ότι περιέχονται μόνο άλλα στοιχεία και όχι ελεύθεροι χαρακτήρες. Στην περίπτωση αυτή, οι περιορισμοί περιέχουν ένα *μοντέλο περιεχομένου*, βασισμένο σε μια απλή γραμματική που υποδηλώνει τους επιτρεπόμενους τύπους των στοιχείων παιδιών και την σειρά που μπορούν αυτά να έχουν<sup>4</sup>. Η γραμματική βασίζεται σε λίστες που ορίζουν τις εναλλακτικές επιλογές καθώς και την αλληλοδιαδοχή των τμημάτων που απαρτίζουν το περιεχόμενο. Στην περίπτωση που ο τύπος του στοιχείου δηλώνει περιεχόμενο mixed, μπορούμε να θέσουμε

<sup>4</sup> Η σειρά αποτελεί μια διαφορά ανάμεσα στην XML και στα μοντέλα ημιδεδωμένων δεδομένων που είδαμε στην προηγούμενη ενότητα. Λόγω της εγγραφο-κεντρικής καταγωγής της XML, η σειρά των στοιχείων έχει σημασία. Από την άλλη, η σειρά των αντικειμένων δεν είναι σημαντική στα μοντέλα ημιδομημένων δεδομένων της Ενότητας 2.2.1.

περιορισμούς στους τύπους των στοιχείων παιδιών, αλλά όχι στην σειρά τους ή στον αριθμό των στιγμιοτύπων τους. Οι δηλώσεις λίστας-γνωρισμάτων μπορούν να χρησιμοποιηθούν για: (a) να ορίσουν ένα σύνολο γνωρισμάτων για κάποιο δεδομένο τύπο στοιχείου, (b) να εισάγουν περιορισμούς για τους τύπους αυτών των γνωρισμάτων, και (c) να παράσχουν εξ ορισμού τιμές. Οι δυνατοί τύποι γνωρισμάτων είναι: `StringType`, που σημειώνεται σαν “CDATA” και υποδηλώνει ότι η τιμή του γνωρίσματος είναι μια σειρά χαρακτήρων, `EnumeratedType`, και `TokenizedType`. Στην περίπτωση του `TokenizedType`, το γνώρισμα μπορεί να δηλωθεί μεταξύ άλλων σαν: “ID”, που υπονοεί ότι πρόκειται για ένα μοναδικό αναγνωριστικό του αντίστοιχου στοιχείου, “IDREF”, που χρησιμοποιείται για να αναφερθεί σε κάποιο άλλο στοιχείο μέσω του “ID” γνωρίσματος του τελευταίου, και “IDREFS”, που χρησιμοποιείται για να αναφερθεί σε πολλά στοιχεία μαζί μέσω των “ID” γνωρισμάτων τους. Γνώρισμα του τύπου “ID” μαζί με “IDREF” και “IDREFS” μπορούν να χρησιμοποιηθούν για να μετατρέψουν το δένδρο του εγγράφου XML σε γράφο, επιτρέποντας σε ένα στοιχείο να αναφέρεται από περισσότερα του ενός στοιχείου «πατέρα». Η εξ ορισμού τιμή ενός γνωρίσματος μπορεί να περιέχει προαιρετικά μια λέξη κλειδί και μια τιμή, όπως στα “#REQUIRED”, “#IMPLIED”, “#FIXED <εξ\_ορισμού\_τιμή>”, ή απλά “<εξ\_ορισμού\_τιμή>”.

Ένα έγγραφο XML λέγεται *ορθά-σηματισμένο* εάν:

1. Υπακούει στην παραγωγή με την ονομασία “document” που δίνεται στο [XML], η οποία ορίζει την γραμματική της XML με χρήση σημειογραφίας EBNF.
2. Οι αναφορές που υπάρχουν μέσα στο έγγραφο είναι καλά ορισμένες (δείχνουν σε στοιχείο που πράγματι υπάρχει).

Η συμμόρφωση με την παραγωγή “document” υπονοεί ότι όλα τα στοιχεία βρίσκονται το ένα μέσα στο άλλο, και ότι υπάρχει ένα μοναδικό στοιχείο που λέγεται *ρίζα* ή *στοιχείο εγγράφου*, που δεν βρίσκεται στο περιεχόμενο κανενός άλλου στοιχείου.

### Παράδειγμα 2.3

```
<!DOCTYPE database [
  <!ELEMENT biblio (book*, paper*)>
  <!ELEMENT book (author+, date, title, abstract?)>
  <!ELEMENT paper (author+, date, title, published, abstract?)>
  <!ELEMENT author #PCDATA>
  <!ELEMENT date #PCDATA>
  <!ELEMENT title #PCDATA>
  <!ELEMENT published #PCDATA>
  <!ELEMENT abstract #PCDATA>
  <!ATTLIST database id ID #REQUIRED>
  <!ATTLIST book id ID #REQUIRED>
]>
```

◆

Προκειμένου να οριστούν περιορισμοί στην λογική δομή και στην διάταξη αποθήκευσης των εγγράφων XML, μπορεί να χρησιμοποιηθεί το Document Type Definition, ή DTD. Το Παράδειγμα 2.3 παρουσιάζει ένα DTD για το έγγραφο XML στο Παράδειγμα 2.2. Ένα DTD περιέχει δηλώσεις επισημείωσης, οι οποίες μπορούν να είναι δηλώσεις τύπου στοιχείου, δηλώσεις λίστας-γνωρισμάτων, δηλώσεις οντότητας, και δηλώσεις σημειογραφίας. Οι δηλώσεις τύπου στοιχείου και οι δηλώσεις λίστας-γνωρισμάτων ορίζουν περιορισμούς στους τύπους και στο περιεχόμενο των στοιχείων και των γνωρισμάτων της XML. Ένα έγγραφο XML χαρακτηρίζεται *έγκυρο* σε σχέση με ένα DTD, εάν συνδέεται με αυτό το DTD, και εάν συμμορφώνεται με τους περιορισμούς που εκφράζονται στο DTD. Η οντότητα που περιέχει το DTD πρέπει να εμφανίζεται πριν από το πρώτο στοιχείο, και μπορεί (a) να εισάγει δηλώσεις

επισημείωσης κατευθείαν μέσα στο έγγραφο XML, (b) να δείχνει σε κάποιο εξωτερικό έγγραφο που περιέχει δηλώσεις επισημείωσης, ή (c) να συνδυάζει και τα δύο, στην οποία περίπτωση το DTD του εγγράφου αποτελείται από τα δυο υποσύνολα δηλώσεων μαζί.

## 2.3.2 Τεχνολογίες συναφείς με την XML

Το World Wide Web Consortium [W3C] έχει την ευθύνη για τον συντονισμό και την παραγωγή των θεμάτων της XML, και έχει δημοσιεύσει έναν αριθμό προδιαγραφών πάνω σε σχετικές τεχνολογίες που αξιοποιούν και προωθούν την XML. Στην ενότητα αυτή επισκοπούμε σύντομα κάποιες από τις προδιαγραφές αυτές, που φαίνεται ότι θα κατέχουν μια σημαντική θέση σε μια πλούσια πλατφόρμα τεχνολογιών για τον Ιστό.

Τα DTDs έχουν περιορισμούς στην εκφραστικότητά τους. Για παράδειγμα, δεν μπορούμε να χρησιμοποιήσουμε ένα DTD για να ορίσουμε ότι το στοιχείο `price` θα πρέπει να είναι πραγματικός αριθμός σταθερής υποδιαστολής, ή ότι ένα γνώρισμα τύπου “IDREF” θα πρέπει να δείχνει αποκλειστικά σε στοιχεία συγκεκριμένου τύπου. Για τον λόγο αυτό, προτάθηκε το XML Schema [XSC0, XSC1, XSC2], με χαρακτηριστικά που καλύπτουν τέτοιου είδους ανάγκες. Το XML Schema περιέχει έναν αριθμό προκαθορισμένων βασικών τύπων που μπορούν να χρησιμοποιηθούν για τον ορισμό καινούργιων τύπων δεδομένων. Οι τύποι δεδομένων μπορούν να είναι σύνθετοι τύποι που βασίζονται σε άλλους τύπους δεδομένων. Κάθε XML Schema είναι και το ίδιο ένα έγγραφο XML.

Όπως το DTD, έτσι και το XML Schema περιγράφει την λογική δομή δεδομένων σε μορφή XML· δεν αντιμετωπίζει θέματα σημασιολογίας της πληροφορίας. Το ζήτημα της σημασιολογίας είναι σημαντικό επειδή εάν κανείς γνωρίζει για τι πράγμα μιλάει ένα κομμάτι δεδομένων και πως σχετίζεται με άλλα δεδομένα μπορεί να κατασκευάσει καλύτερες μηχανές αναζήτησης και πιο προηγμένες εφαρμογές. Το Resource Description Framework [RDF1, RDF2], ή RDF, προσφέρει μια γενική μέθοδο που μοιάζει με το διάγραμμα οντοτήτων-συσχετίσεων (E-R diagram) και περιγράφει σημασιολογική μεταπληροφορία για έγγραφα XML. Περιγράφει *πόρους*, που προσδιορίζονται από URIs [URI], και που έχουν *ιδιότητες*. Τα *RDF statements* είναι τριάδες, αποτελούμενες από ένα *υποκείμενο* (ο πόρος), ένα *κατηγορημα* (η ιδιότητα), και ένα *αντικείμενο* (η τιμή της ιδιότητας), συνδέοντας έτσι ζεύγη ιδιότητας-τιμής με πόρους. Η τιμή μιας ιδιότητας μπορεί να είναι η τελική τιμή, ή κάποιος άλλος πόρος. Τα RDF statements εκφράζονται και τα ίδια σε XML.

Τα Namespaces [XNS] επιτρέπουν σε ένα και μόνο έγγραφο XML να περιέχει *λεξιλόγιο επισημείωσης* που ορίζεται σε έναν αριθμό διαφορετικών DTDs. Το λεξιλόγιο επισημείωσης αποτελείται από ονόματα στοιχείων και ονόματα γνωρισμάτων. Καθώς το ίδιο όνομα μπορεί να έχει δηλωθεί σε δύο διαφορετικά DTDs με διαφορετική σημασία και τύπο, η ανάμιξη αυτών των δύο DTDs ίσως καταλήξει σε σύγκρουση. Τα Namespaces επιλύουν το πρόβλημα της σύγκρουσης εισάγοντας ένα πρόθεμα που προηγείται των ονομάτων στοιχείων και γνωρισμάτων και τα διαφοροποιεί.

Το XLink [XLIN] ορίζει συνδέσμους ανάμεσα από έγγραφα XML, μέρη εγγράφων, ή πόρους γενικά. Οι σύνδεσμοι αυτοί είναι πιο εκφραστικοί από τους συνδέσμους της HTML, καθώς μπορούν να ορίσουν σχέσεις ανάμεσα από πολλούς πόρους μαζί. Επιπλέον, είναι δυνατόν να επισυναφθεί στον σύνδεσμο μεταπληροφορία, που να περιγράφει ιδιότητες του συνδέσμου. Το XLink εκφράζεται σε XML, και μπορεί να βρίσκεται είτε μέσα στα εμπλεκόμενα έγγραφα είτε ξεχωριστά από αυτά. Οι πόροι προσδιορίζονται από URIs, εκτός εάν θέλουμε να αναφερθούμε σε ένα κομμάτι στο εσωτερικό ενός εγγράφου XML, οπότε χρησιμοποιούμε την προδιαγραφή XPointer [XPTR].

Όπως έχει αναφερθεί, η XML περιορίζεται στο να περιγράφει την φυσική και λογική δομή των εγγράφων και δεν ασχολείται με θέματα παρουσίασης της πληροφορίας. Η

παρουσίαση αντιμετωπίζεται με το XML Stylesheets [XSL], ή XSL, και το XSL Transformations [XSLT], ή XSLT. Το XSL είναι μια γλώσσα που ορίζει ιδιότητες παρουσίασης εγγράφων XML, ενώ το XSLT είναι μέρος του XSL και είναι μια γλώσσα που μετατρέπει έγγραφα XML. Ένα έγγραφο XSLT είναι μια συλλογή κανόνων μετατροπής που ενεργούν πάνω σε κάποιο αρχικό έγγραφο XML για να παραχθεί ένα νέο έγγραφο. Το νέο έγγραφο μπορεί να είναι έγγραφο XML, έγγραφο HTML, ή ένα έγγραφο σε κάποιον άλλο μορφότυπο, ανάλογα με τους κανόνες μετατροπής. Κάθε κανόνας αποτελείται από ένα *πρότυπο* και έναν *οδηγό*. Αρχίζοντας από την ρίζα, τα πρότυπα συγκρίνονται αναδρομικά με το δένδρο του αρχικού εγγράφου και σε περίπτωση ομοιότητας χρησιμοποιείται ο οδηγός για να παραχθούν κομμάτια του δένδρου του αποτελέσματος. Τα τμήματα του αρχικού δένδρου που δεν επιτυγχάνουν στην σύγκριση μεταφέρονται κατευθείαν στο δένδρο του αποτελέσματος. Το XSL και το XSLT εκφράζονται σε XML.

#### Παράδειγμα 2.4

```
<xsl:template match="/" >
  <html>
    <head>
      <title>Book Titles</title>
    </head>
    <body>
      <h1>Book Titles</h1>
      <xsl:for-each select="database/biblio/book">
        <xsl:value-of select="title"/>
        <br></br>
      </xsl:for-each>
    </body>
  </html>
</xsl:template>
```

Το Παράδειγμα 2.4 δείχνει μέρος ενός εγγράφου XSLT που μετατρέπει το XML στο Παράδειγμα 2.2 σε ένα έγγραφο HTML που παραθέτει τους τίτλους των βιβλίων. Παρατηρήστε την χρήση των XML Namespaces και του προθέματος `xsl` που προηγείται όσων στοιχείων έχουν οριστεί σαν μέρος του XSLT. Το XSL και ειδικά το τμήμα του μετατροπής, το XSLT, μπορεί να θεωρηθεί σαν μια περιορισμένη γλώσσα επερώτησης για XML [ABS00, BC00].

Το XSLT στο Παράδειγμα 2.4, περιέχει το `database/biblio/book` που είναι η συντομευμένη μορφή ενός XPath [XPTH]. Το XPath είναι παρόμοιο με τις εκφράσεις μονοπατιού των SSD, αρχίζει από κάποιον κόμβο που καλείται *κόμβος περιβάλλοντος*<sup>5</sup> και συνεχίζει με μια ακολουθία *βημάτων*. Κάθε βήμα έχει έναν *άξονα*, που παρέχει οδηγίες πλοήγησης, έναν *έλεγχο κόμβου*, και μια προαιρετική λίστα *κατηγορημάτων*. Θεωρήστε το XPath `root()/descendant::book/child::author[position()=first()]`, που επιστρέφει τον πρώτο συγγραφέα για κάθε βιβλίο του XML στο Παράδειγμα 2.2. Σε αυτό το XPath, το `descendant` και το `child` αποτελούν άξονες, το `book` και το `author` είναι έλεγχοι κόμβων που συγκρίνονται με τα ονόματα των στοιχείων, και το `[position()=first()]` είναι ένα κατηγορημα που ικανοποιείται μόνο από τον πρώτο συγγραφέα κάθε βιβλίου.

<sup>5</sup> Στο πλαίσιο της παρούσης διατριβής χρησιμοποιούμε τον όρο **κόμβος περιβάλλοντος** με τελείως διαφορετική σημασία. Στο Κεφάλαιο 4 εισάγουμε τον κόμβο περιβάλλοντος σαν έναν ειδικό τύπο κόμβου που αναπαριστά εκφάνσεις **πολυδιάστατων οντοτήτων**.

### 2.3.3 Γλώσσες επερωτήσεων για XML

Όπως αναφέρθηκε στην προηγούμενη ενότητα, το XSL μπορεί να ιδωθεί σαν γλώσσα επερωτήσεων για XML. Πάντως, η εκφραστικότητα του XSL σαν γλώσσα επερωτήσεων είναι περιορισμένη [ABS00, BC00]. Η σημασία μιας ισχυρής γλώσσας επερωτήσεων για XML έγινε σύντομα αντιληπτή [Abi99], και καταρτίστηκε ένας αριθμός επιθυμητών χαρακτηριστικών που πρέπει να καλύπτει μια τέτοια γλώσσα [Mai98, DFF+99, XQR].

Λόγω της μεγάλης ομοιότητας ανάμεσα στην XML και στα μοντέλα ημιδομημένων δεδομένων, το OEM και η Lorel προσαρμόστηκαν [GMW99] στην XML χωρίς ιδιαίτερη δυσκολία. Οι κύριες αλλαγές έχουν να κάνουν με την ενσωμάτωση χαρακτηριστικών για την διαχείριση γνωρισμάτων και γνωρισμάτων αναφοράς. Μια επερώτηση Lorel μπορεί να χειριστεί γνωρίσματα αναφοράς στην XML με διττό τρόπο: είτε σαν απλά γνωρίσματα, ή σαν ακμές που δείχνουν σε «υποστοιχεία». Άλλες προσθήκες είναι οι *περιοριστές έρους* και ένας όρος *order-by* που αντιμετωπίζουν την διάταξη των στοιχείων σε έγγραφο XML, και η χρήση συναρτήσεων Skolem κατά την κατασκευή των αποτελεσμάτων για να αποφευχθεί η δημιουργία του ίδιου στοιχείου πολλές φορές.

Σε αντίθεση με την Lorel, η XML-QL [DFF+99a, DFF+99] είναι μια γλώσσα επερωτήσεων σχεδιασμένη κατευθείαν για την XML. Το μοντέλο δεδομένων της XML-QL είναι το XML Graph, μια παραλλαγή του οικείου κατευθυνόμενου γράφου με ετικέτες στις ακμές, το οποίο έρχεται σε δύο «γεύσεις»: μια μη διατεταγμένη και μια διατεταγμένη. Όμοια με την UnQL, η XML-QL χρησιμοποιεί τον όρο *where* για να δεσμεύσει μεταβλητές μέσα σε *πρότυπα στοιχείων*, και τον όρο *construct* για να δημιουργήσει το έγγραφο XML του αποτελέσματος. Επιπλέον, η XML-QL υποστηρίζει *μεταβλητές κάρτας* που είναι παρόμοιες με τις μεταβλητές ετικέτας στην UnQL, και *ομαλές εκφράσεις μονοπατιού* και *μπαλαντέρ* για την διατύπωση εκφραστικότερων προτύπων στοιχείου.

```
<Smith_publ>
  WHERE <$p>
    <author>Smith</>
    <date> $d </>
    <title> $t </>
  </> IN "www.site.edu/biblio.xml",
  $p IN {book, paper}
  CONSTRUCT <$p>
    <title> $t </>
    <date> $d </>
  </>
</Smith_publ>
```

Το πρότυπο στοιχείου στην παραπάνω επερώτηση XML-QL ταιριάζει με τα στοιχεία *book* και *paper* του εγγράφου *biblio.xml* (δες το Παράδειγμα 2.2), που περιέχουν ένα υποστοιχείο *author* με τιμή "Smith". Η μεταβλητή κάρτας *\$p* μεταφέρει ονόματα στοιχείων (*book* ή *paper*) από το αρχικό έγγραφο στα νεοκατασκευασμένα στοιχεία. Στο έγγραφο του αποτελέσματος, τα στοιχεία που κατασκευάστηκαν είναι υποστοιχεία του στοιχείου *Smith\_publ* που είναι η ρίζα.

Άλλες γλώσσες επερωτήσεων για XML είναι η XML-GL [CCD+99], μια γραφική γλώσσα επερωτήσεων για XML, η YATL [CS00], που βασίζεται σε συναρτησιακές γλώσσες προγραμματισμού, και η XQL [XQL], που επεκτείνει το XSL. Συγκρίσεις ανάμεσα από γλώσσες επερωτήσεων για XML επιχειρούνται στα [BC00, FSW99].

Η Quilt [CRF00] είναι μια γλώσσα επερωτήσεων για XML που δανείζεται στοιχεία από πολλές άλλες γλώσσες, συγκεκριμένα από το XPath, την XQL, την XML-QL, την SQL [SQL],

την OQL [ODMG], την Lorel, και την YATL. Η δομή της Quilt βασίζεται σε εκφράσεις FLWR (προφέρεται “flower”), που απαρτίζονται από όρους for, let, where, και return. Οι μεταβλητές δεσμεύονται μέσα σε όρους for, οι όροι let επιτρέπουν να δεσμευτούν επιπλέον μεταβλητές, οι όροι where φιλτράρουν τις δεσμεύσεις πλειάδων που παράγονται από τους όρους for και let, και ο όρος return κατασκευάζει τα αποτελέσματα. Η επερώτηση Quilt που ακολουθεί παραθέτει τους συγγραφείς βιβλίων στο Παράδειγμα 2.2 σε αλφαβητική σειρά, και για κάθε συγγραφέα δίνει τα βιβλία του ή της διατεταγμένα σύμφωνα με την ημερομηνία δημοσίευσης:

```
<author_list>
  FOR $a IN distinct(document("biblio.xml")//author)
  RETURN
    <author>
      <name> $a/text() </name>
      FOR $b IN document("biblio.xml")//book[author=$a]
      RETURN
        <book>
          $b/title,
          $b/date
        </book> SORTBY date DESCENDING
    </author> SORTBY name
</author_list>
```

Η Quilt χρησιμοποιεί το XPath για την πλοήγηση σε έγγραφα XML. Για παράδειγμα, το XPath `document("biblio.xml")//book[author=$a]` ταιριάζει με όλα τα στοιχεία `book` στο `biblio.xml` που έχουν ένα υποστοιχείο `author` με τιμή ίση με την τιμή της μεταβλητής `$a`. Επιπλέον, το XPath `$b/title` προκαλεί τα υποστοιχεία `title` των βιβλίων (`books`) που είναι δεσμευμένα στο `$b` να γίνουν μέρος του αποτελέσματος. Παρατηρήστε το πως η Quilt επιτρέπει το εμφώλιασμα εκφράσεων με γενικό και ευέλικτο τρόπο. Η Quilt αποτελεί την βάση της XQuery [XQRY], μιας γλώσσας επερώτησης για XML που υποστηρίζεται από το W3C.

## 2.4 MSSD ΚΑΙ MXML

Στην διατριβή αυτή ενσωματώνουμε ερμηνευτικά περιβάλλοντα στα ημιδομημένα δεδομένα και ορίζουμε τα **πολυδιάστατα ημιδομημένα δεδομένα (MSSD)**. Συγκεκριμένα, επεκτείνουμε το OEM και προτείνουμε τον **Πολυδιάστατο Γράφο Δεδομένων** και το **Πολυδιάστατο OEM (MOEM)**, ένα μοντέλο δεδομένων για MSSD. Προκειμένου να αναπαραστήσουμε MSSD με κείμενο εισάγουμε τις εκφράσεις **mssd-expressions**, που είναι μια επέκταση των εκφράσεων `ssd-expressions`. Επίσης παρουσιάζουμε την **Πολυδιάστατη Γλώσσα Επερωτήσεων (MQL)**, η οποία ενσωματώνει το ερμηνευτικό περιβάλλον μαζί με χαρακτηριστικά της Lorel και της UnQL. Επιπλέον, επεκτείνουμε την XML, το DTD, και το XSL, και ορίζουμε την **Πολυδιάστατη XML (MXML)**, το **MDTD**, και το **MXSL**, όπου το ερμηνευτικό περιβάλλον παίζει ουσιαστικό ρόλο.

Παρά το ότι η έννοια του ερμηνευτικού περιβάλλοντος δεν υποστηρίζεται ευθέως από την XML, η σημασία του αναγνωρίζεται έμμεσα. Η XML ορίζει το ειδικό γνώρισμα `xml:lang`, που μπορεί να εισαχθεί σε έγγραφα XML για να προσδιορίσει την γλώσσα που χρησιμοποιείται για το περιεχόμενο οποιουδήποτε στοιχείου καθώς και για την τιμή των γνωρισμάτων του. Στο επόμενο κεφάλαιο επισκοπούμε προηγούμενες εργασίες σε ερμηνευτικά περιβάλλοντα (`context`), και παρουσιάζουμε την δική μας προσέγγιση.



### 3 ΤΥΠΟΠΟΙΗΣΗ ΕΡΜΗΝΕΥΤΙΚΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ

Τα πολυδιάστατα ημιδομημένα δεδομένα (MSSD για συντομία) αναπαριστούν πληροφοριακές οντότητες που παρουσιάζουν διαφορετικές εκφάνσεις, που έχουν μεταβλητό περιεχόμενο (σε τιμή και / ή δομή), κάτω από διαφορετικά ερμηνευτικά περιβάλλοντα (contexts). Αυτό το κεφάλαιο εισάγει μια τυποποίηση (formalism) για την έκφραση του ερμηνευτικού περιβάλλοντος (ή απλά περιβάλλοντος για συντομία) μέσω μεταβλητών που καλούνται **διαστάσεις**, και καθορίζει μια αντιστοιχία ανάμεσα από το ερμηνευτικό περιβάλλον και *πιθανούς κόσμους*. Επιπρόσθετα, ορίζονται πράξεις περιβάλλοντος, και εξετάζονται οι ιδιότητές τους. Τέλος, προτείνεται μια βολική σημειογραφία για εκφράσεις περιβάλλοντος και συνθήκες περιβάλλοντος.

Η προσέγγιση που παρουσιάζεται στο παρόν κεφάλαιο αντιμετωπίζει το ερμηνευτικό περιβάλλον σαν ένα σύνολο περιορισμών, που συνδυάζονται για να ορίσουν πλαίσια κάτω από τα οποία η πληροφορία αποκτά συγκεκριμένο νόημα. Ένα τέτοιο πλαίσιο λέγεται **κόσμος**, ενώ οι περιορισμοί εκφράζονται με την ανάθεση τιμών στις μεταβλητές των διαστάσεων. Το ερμηνευτικό περιβάλλον, όπως ορίζεται στο κεφάλαιο αυτό, στοχεύει στο να είναι ευέλικτο και παίρνει υπόψη τα ειδικά χαρακτηριστικά και την δυναμική φύση των δεδομένων του Ιστού. Στις εφαρμογές του Ιστού, οι παραγωγοί και οι καταναλωτές πληροφορίας που εξαρτάται από το περιβάλλον πιθανόν να μην έχουν την ίδια εντύπωση για το τι θα πρέπει να καλύπτει το ερμηνευτικό περιβάλλον. Έτσι, οι παραγωγοί της πληροφορίας δεν θα πρέπει να υποχρεώνονται στην δημιουργία περιορισμών για κάθε δυνατή διάσταση, ενώ οι καταναλωτές της πληροφορίας θα πρέπει να μπορούν να τους ερμηνεύσουν μέσα στο δικό τους πλαίσιο, πράγμα που ίσως να σημαίνει την χρήση επιπλέον διαστάσεων. Η δυνατότητα ενσωμάτωσης νέων διαστάσεων χωρίς να αλλάζει η σημασία των ήδη υπαρχουσών εκφράσεων περιβάλλοντος θα ήταν επίσης επιθυμητή για την ολοκλήρωση από διάφορες πηγές του Ιστού ετερογενούς πληροφορίας που εξαρτάται από το περιβάλλον.

#### 3.1 ΣΧΕΤΙΚΕΣ ΕΡΓΑΣΙΕΣ

Την τελευταία δεκαετία η έννοια του ερμηνευτικού περιβάλλοντος έχει εφαρμοστεί σε διάφορες περιοχές της πληροφορικής. Μια εκτεταμένη βιβλιογραφική ερευνά που απαντάται στο [The01] συζητά την χρήση του ερμηνευτικού περιβάλλοντος στην γλωσσολογία, στην μουσική, στην ανάπτυξη λογισμικού, στην μηχανική μάθηση, στα δίκτυα, στην τεχνητή νοημοσύνη, στην νοηματική κατάταξη μοντέλων, στις εμφωλιασμένες συσχετίσεις, στην κατηγοριοποίηση, στις βάσεις πληροφορίας, και στον Ιστό. Ειδικά για τον Ιστό, όπου “authors of Web pages have diverse backgrounds, knowledge, culture, and aims” και “the availability of metadata is inconsistent”, το ερμηνευτικό περιβάλλον δεν χρησιμοποιείται με τρόπο που να αναδεικνύει όλες του τις δυνατότητες και “greater use of context in web may help increase competition and diversity on the web”.

Μιλώντας γενικά, το ερμηνευτικό περιβάλλον χρησιμοποιείται σαν εργαλείο συλλογιστικής για οπτικές γωνίες και υφισταμένες δοξασίες, και σαν ένας αφαιρετικός μηχανισμός που βοηθά στην αντιμετώπιση της πολυπλοκότητας, της ετερογένειας, και της μερικής γνώσης. Σε ότι ακολουθεί, εστιάζουμε σε κάποιες προσεγγίσεις που έχουν ενδιαφέρον από την οπτική της παρούσας εργασίας.

Το ερμηνευτικό περιβάλλον εξετάζεται στο [Bun94] από γλωσσολογικής πλευράς, και εκλαμβάνεται ως εξαρτώμενο από ένα σύνολο *παραγόντων*, ομαδοποιημένων σε γενικές κατηγορίες, που επίσης καλούνται *διαστάσεις*. Γίνεται διάκριση ανάμεσα σε *συνολικές απόψεις*, που τείνουν να παραμένουν σταθερές, και *τοπικές απόψεις*, των οποίων η τιμή αλλάζει δυναμικά ανάλογα με την ανταπόκριση της εφαρμογής στην οποία χρησιμοποιείται το ερμηνευτικό περιβάλλον. Στην προσέγγισή μας ορίζουμε το περιβάλλον αναθέτοντας συγκεκριμένες τιμές σε διαστάσεις. Έτσι, παρόλο που εφαρμόζουμε το ερμηνευτικό περιβάλλον σε διαφορετικό πλαίσιο, θα μπορούσαμε να πούμε ότι η δική μας έννοια του περιβάλλοντος σχετίζεται με αυτό που περιγράφεται στο [Bun94] σαν παράγοντες συνολικής άποψης (global aspect factors).

Στο [Giu93] το περιβάλλον χρησιμοποιείται σαν μέσο για την τυποποίηση της συλλογιστικής πάνω σε ένα υποσύνολο και μόνο μιας ολικής βάσης γνώσης (global knowledge base), σε μια προσέγγιση που λέγεται *τοπικοποίηση* (*localization*). Η *Σημασιολογία Τοπικού Μοντέλου* (*Local Model Semantics*) που προτείνεται στο [GG01] τυποποιεί δυο αρχές: την *αρχή της τοπικότητας*, σύμφωνα με την οποία η συλλογιστική χρησιμοποιεί μόνον μέρος αυτού που είναι δυνητικά διαθέσιμο και αυτό το μέρος σχηματίζει το περιβάλλον, και την *αρχή της συμβατότητας*, η οποία δηλώνει ότι υπάρχει μια συμβατότητα ανάμεσα από συλλογιστικές που διατυπώνονται σε διαφορετικά περιβάλλοντα. Η σημασιολογία τοπικού μοντέλου μπορεί να εφαρμοστεί για την συλλογιστική με οπτικές γωνίες· η αρχή της τοπικότητας έχει να κάνει με το ότι έχουμε πολλαπλές όψεις της ίδιας οντότητας, ενώ η αρχή της συμβατότητας συσχετίζει αυτές τις όψεις σε συνεπείς σχηματισμούς.

Όπως αναφέρεται στο [OS99], το περιβάλλον μπορεί να χρησιμοποιηθεί σε ολικά συστήματα πληροφορίας (global information systems) για να αναπαρασταθεί η *σημασία στον πραγματικό κόσμο* ενός αντικειμένου. Επιπλέον, η χρήση του περιβάλλοντος μπορεί να έχει αρκετά πλεονεκτήματα.

- *Οικονομία αναπαράστασης*: Το περιβάλλον μπορεί να δράσει σαν μηχανισμός απομόνωσης και πρόσβασης σε τμήματα της πληροφορίας, με παρόμοιο τρόπο όπως οι όψεις των βάσεων δεδομένων (database views).
- *Οικονομία συλλογιστικής*: Η συλλογιστική μπορεί να λάβει χώρα με ένα περιβάλλον που συνδέεται με μια μοναδική πηγή πληροφορίας αντί για την ολική βάση δεδομένων.
- *Διαχείριση ασυνεπούς πληροφορίας*: Όταν ολοκληρώνουμε ανεξάρτητες πηγές πληροφορίας, μπορούμε να επιτρέψουμε ασυνέπεια στην πληροφορία εφόσον η πληροφορία παραμένει συνεπής μέσα στο περιβάλλον των επερωτήσεων του χρήστη.
- *Ευέλικτη σημασιολογία*: Η σχέση και η σημασιολογική εγγύτητα ανάμεσα από δυο αντικείμενα μπορεί να μεταβάλλεται ανάλογα με το περιβάλλον.

Η τυποποίηση της σημασιολογίας τοπικού μοντέλου [GG01] χρησιμοποιείται στο [GS98a] για να αναπαραστήσει μια ομόσπονδη βάση δεδομένων (federated database) σαν ένα σύνολο τοπικών μοντέλων που αντιστοιχούν στις διαφορετικές βάσεις του ομόσπονδου συστήματος. Η ίδια τυποποίηση εφαρμόζεται στο [GS98b] για την ολοκλήρωση πληροφορίας από έναν αριθμό αυτόνομων πρακτόρων που λειτουργούν στο πεδίο του ηλεκτρονικού εμπορίου. Στο [FDFF95] χρησιμοποιείται μια επέκταση της λογικής πρώτης τάξης που λέγεται λογική

περιβάλλοντος (context logic), για την αυξητική ολοκλήρωση ετερογενών πηγών πληροφορίας. Για κάθε πηγή, το *περιβάλλον πηγής πληροφορίας* (*information source context*) κρατά το αρχικό σχήμα, και το *περιβάλλον σημασιολογίας* (*semantic context*) εκφράζει υπονοούμενες παραδοχές για το σχήμα και περιέχει κανόνες μετάφρασης. Τέλος, το *περιβάλλον ολοκλήρωσης* (*integrating context*) συγκεντρώνει κανόνες από αρκετά περιβάλλοντα σημασιολογίας και από άλλα περιβάλλοντα ολοκλήρωσης, και παρέχει ένα ενοποιημένο μοντέλο που αντιμετωπίζει συγκεκριμένες μορφές ετερογένειας.

Στο [MM95] το περιβάλλον παρέχει έναν γενικό μηχανισμό οπτικών γωνιών που χρησιμοποιείται για την κατάτμηση μιας βάσης πληροφορίας σε διαχειρίσιμα τμήματα που περιέχουν συσχετιζόμενα αντικείμενα. Η ιδέα αυτή αναπτύσσεται παραπάνω στα [The01, TACS98], όπου ορίζεται αυστηρά το περιβάλλον και πράξεις ανάμεσα από περιβάλλοντα. Κάθε περιβάλλον συνδέεται με ένα *λεξικό*, που με την σειρά του συσχετίζει ολικά αναγνωριστικά αντικειμένων με ένα σύνολο εναλλακτικών ονομάτων μέσω των οποίων μπορούμε να αναφερθούμε στα αντικείμενα αυτά. Ένα αντικείμενο μπορεί να έχει διαφορετικά ονόματα σε διαφορετικά περιβάλλοντα. Τα αντικείμενα σε ένα περιβάλλον μπορούν να είναι και τα ίδια περιβάλλοντα, πράγμα που οδηγεί σε εμφωλιασμένες δομές οποιουδήποτε βάθους. Οι πράξεις που ορίζονται περιέχουν την *ένωση*, την *τομή*, και την *διαφορά* ανάμεσα από περιβάλλοντα. Τα *περιβάλλοντα διαμόρφωσης* (*configuration contexts*) αναπαριστούν σύνθετα αντικείμενα που αποτελούνται από συγκεκριμένες εκδοχές των συστατικών μερών τους, και μπορούν να χρησιμοποιηθούν για να συντονίσουν δουλειές συνεργασίας όπου πολλοί άνθρωποι δουλεύουν πάνω στα ίδια έγγραφα.

Στο κεφάλαιο αυτό ορίζουμε πράξεις ανάμεσα από περιβάλλοντα παρομοίως με το [The01], όπως για παράδειγμα την **ένωση περιβάλλοντος**, την **τομή περιβάλλοντος**, και την **διαφορά περιβάλλοντος**. Επιπλέον, στο Κεφάλαιο 4 ορίζουμε διαδικασίες **αναγωγής**, που αντιστοιχούν στους μηχανισμούς οπτικών γωνιών και προβάλλουν ένα τμήμα της διαθέσιμης πληροφορίας ανάλογα με το περιβάλλον. Όμως, η άποψή μας για το τι είναι το περιβάλλον και για το πώς αναπαρίσταται διαφέρει ριζικά. Στα [MM95, The01, TACS98] τα περιβάλλοντα είναι αντικείμενα τα ίδια, και συμπεριφέρονται σαν δοχεία άλλων αντικειμένων που σχετίζονται με μια συγκεκριμένη κατάσταση του πραγματικού κόσμου. Στην δική μας προσέγγιση το περιβάλλον είναι μεταπληροφορία που προσκολλάται στα αντικείμενα και στις σχέσεις ανάμεσα από τα αντικείμενα, και παρέχει μια πλατφόρμα για την ερμηνεία των δεδομένων.

Σε αυτή την ερμηνευτική πλατφόρμα, το περιβάλλον αντιπροσωπεύει ένα σύνολο *πιθανών κόσμων*. Στις λογικές Modal Logics [Fi93] οι πιθανοί κόσμοι προσδιορίζουν τις τιμές αληθείας των προτάσεων. Για παράδειγμα, η πρόταση «Βλέπω ότι βρέχει» είναι ψευδής στον κόσμο όπου είμαι στην Αθηνά και όπου η ημερομηνία είναι 3 Μαρτίου 2003, αλλά μπορεί να είναι αληθής σε κάποιον κόσμο όπου βρίσκομαι σε άλλο μέρος σε άλλον χρόνο. Αναλογικά, οι κόσμοι στα MSSD προσδιορίζουν πληροφοριακές οντότητες, και ορίζουν ποιες παραλλαγές μιας πληροφοριακής οντότητας υφίσταται κάτω από κάποιες δεδομένες συνθήκες και πώς αυτές οι παραλλαγές θα πρέπει να ερμηνευτούν.

Οι κόσμοι στα MSSD εκφράζονται με έναν απλό τρόπο, χρησιμοποιώντας ζεύγη (*όνομα, τιμή*) μεταβλητών που λέγονται διαστάσεις. Παρόμοιες κατασκευές έχουν χρησιμοποιηθεί ανεξάρτητα σε δυο περιπτώσεις. Στο [NP03] χρησιμοποιούνται ζεύγη (*όνομα, τιμή*) *χαρακτηριστικών* όπως το language για να παραμετροποιηθεί η διεργασία της αυτόματης έκδοσης στον Ιστό. Επιπλέον, οι διαστάσεις έχουν χρησιμοποιηθεί στην Intensional HTML [WSY98, Bro98, Yi97, Bro98a], ή IHTML για συντομία, μια γλώσσα συγγραφής του Ιστού η οποία βασίζεται και επεκτείνει ιδέες που προτάθηκαν για ένα σύστημα εκδοχών λογισμικού [PW93]. Η IHTML επιτρέπει σε μια Ιστοσελίδα να έχει διαφορετικές παραλλαγές και να προσαρμόζεται αυτόματα σε έναν δεδομένο κόσμο. Οι κόσμοι αυτοί ορίζονται δίνοντας μια μοναδική τιμή σε κάθε διάσταση. Η IHTML ακολουθεί μια εγγραφο-κεντρική προσέγγιση,

χρησιμοποιώντας μια διαδικασία ονοματολογίας για να κωδικοποιήσει κόσμους στα ονόματα των φακέλων των εκδοχών των έγγραφων. Η δική μας εργασία επηρεάστηκε από την HTML, αλλά ακολουθεί διαφορετική κατεύθυνση: (a) το περιβάλλον αντιπροσωπεύει ένα σύνολο πιθανών κόσμων, και οι διαστάσεις που χρησιμοποιούνται για να εκφράσουν ένα περιβάλλον μπορούν να λάβουν πολλές τιμές και να συνδυαστούν σε συζεύξεις και διαζεύξεις, (b) τυποποιούμε την έννοια του περιβάλλοντος και εισάγουμε πράξεις ανάμεσα από τα περιβάλλοντα, και (c) εστιάζουμε σε μια δεδομένο-κεντρική προσέγγιση, που στοχεύει στο να ενσωματώσει το ερμηνευτικό περιβάλλον σε μοντέλα δεδομένων και γλώσσες επερωτήσεων για ημιδομημένα δεδομένα.

## 3.2 ΕΡΜΗΝΕΥΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ, ΔΙΑΣΤΑΣΕΙΣ, ΚΑΙ ΚΟΣΜΟΙ ΣΤΑ MSSD

Η κύρια διαφορά μεταξύ πολυδιάστατων και συμβατικών ημιδομημένων δεδομένων έγκειται στην έννοια της **πολυδιάστατης οντότητας**. Η πολυδιάστατη οντότητα είναι μια πληροφοριακή οντότητα που παρουσιάζει διαφορετικές εκφάνσεις κάτω από διαφορετικές συνθήκες. Για παράδειγμα, θεωρήστε την περίπτωση της τιμής ενός βιβλίου στο Διαδίκτυο, που εκφράζεται και σε Ευρώ και σε Δολάρια: το price είναι μια πολυδιάστατη οντότητα με τιμή 25 αν το νόμισμα είναι το Ευρώ, και 22 αν το νόμισμα είναι το Δολάριο<sup>6</sup>. Οι δυο τιμές είναι δυο εκφάνσεις της ίδιας οντότητας price, ενώ το νόμισμα (currency) αντιπροσωπεύεται από μια μεταβλητή **διάστασης** που καθορίζει τις συνθήκες κάτω από τις οποίες υφίσταται κάθε έκφανση – με άλλα λόγια, το *περιβάλλον* κάτω από το οποίο υφίσταται κάθε έκφανση.

Στο παράδειγμα του price, το περιβάλλον σχηματίζεται χρησιμοποιώντας μια μόνον διάσταση, συγκεκριμένα την currency. Στην γενική περίπτωση, πάντως, οποιοσδήποτε αριθμός διαστάσεων μπορεί να συμμετέχει στον σχηματισμό ενός περιβάλλοντος. Σαν παράδειγμα, θεωρήστε την πληροφορία ενός εγγράφου που πρέπει να υπάρχει σε διάφορες γλώσσες, σε διάφορους μορφότυπους, και σε διάφορους βαθμούς λεπτομέρειας: μια διάσταση είναι το language που μπορεί να λάβει τις τιμές english, spanish, και greek· μια άλλη διάσταση είναι το format με δυνατές τιμές pdf, και ps· τέλος, η διάσταση detail μπορεί να είναι low, medium, ή high. Τα περιβάλλοντα σχηματίζονται συνδυάζοντας συνθήκες που βασίζονται στην ανάθεση τιμών σε διαστάσεις: μια έκφανση της πολυδιάστατης οντότητας document μπορεί να υφίσταται κάτω από χαμηλή ή μεσαία λεπτομέρεια, μορφότυπο pdf, και αγγλική γλώσσα, ενώ μια άλλη έκφανση μπορεί να υφίσταται κάτω από χαμηλή λεπτομέρεια, μορφότυπο pdf, και ελληνική γλώσσα.

Το περιβάλλον δεν χρησιμοποιείται μόνον για να ορίσει τις ακριβείς συνθήκες κάτω από τις οποίες υφίστανται διάφορες εκφάνσεις, αλλά και για να εκφράσει απαιτήσεις με βάση τις οποίες αποτιμάται μια πολυδιάστατη οντότητα και επιλέγονται κάποιες από τις εκφάνσεις της. Για παράδειγμα, η απαίτηση ότι η λεπτομέρεια πρέπει να είναι χαμηλή και η γλώσσα πρέπει να είναι είτε αγγλικά είτε ελληνικά, συνιστά ένα περιβάλλον το οποίο μπορεί να συγκριθεί<sup>7</sup> με τις προαναφερθείσες εκφάνσεις της πολυδιάστατης οντότητας document.

<sup>6</sup> Η ισοτιμία την στιγμή που γράφονταν αυτό το παράδειγμα ήταν 1 Ευρώ για 0.88 Δολάρια Αμερικής.

<sup>7</sup> Επεξηγείται λεπτομερώς στα επόμενα κεφάλαια.

### 3.2.1 Παραδοχές σχετικές με τις διαστάσεις

Για το υπόλοιπο αυτού του κεφαλαίου κάνουμε μια σειρά παραδοχών για τις διαστάσεις και για τα πεδία τιμών των διαστάσεων:

1. Η εμβέλεια (scope) των ονομάτων των διαστάσεων και των τιμών των διαστάσεων στα αντίστοιχα πεδία ορισμού είναι ολική (global), και η σημασία τους είναι η ίδια για όλους.
2. Οι διαστάσεις<sup>8</sup> είναι *ορθογώνιες*, με άλλα λόγια, η τιμή της μιας δεν επηρεάζει τις τιμές των άλλων.
3. Θεωρούμε μόνον σύνολα διαστάσεων  $\mathbf{D}$  που είναι μη κενά και πεπερασμένα.
4. Τα πεδία ορισμού των διαστάσεων θεωρούνται επίσης μη κενά και πεπερασμένα, και θα πρέπει να είναι δυνατόν να περιγραφούν με αναγραφή των στοιχείων τους.

Άλλοι τρόποι αναπαράστασης, καθώς και άπειρα πεδία ορισμού, είναι ίσως χρήσιμα και δεν αποκλείονται, είναι πάντως έξω από τα όρια της παρούσας εργασίας. Το ίδιο ισχύει και για διαστάσεις που δεν είναι ορθογώνιες αλλά έχουν αλληλεξαρτώμενα πεδία ορισμού.

Το ερμηνευτικό περιβάλλον ερμηνεύεται πάντοτε σε σχέση με κάποιο σύνολο διαστάσεων  $\mathbf{D}$  και με τα αντίστοιχα πεδία ορισμού των διαστάσεων. Επιπρόσθετα, προκειμένου να εκτελέσουμε μερικές πράξεις περιβάλλοντος πρέπει να πάρουμε υπόψη τις διαστάσεις στο  $\mathbf{D}$  και τα πεδία ορισμού,  $\mathbf{V}_d$ ,  $\forall d \in \mathbf{D}$ . Η εξάρτηση από το  $\mathbf{D}$  θα εξετάζεται κατά τον ορισμό μιας σχετικής πράξης, και το σύμβολο  $\mathbf{D}$  θα χρησιμοποιείται δίπλα στο σύμβολο της πράξης εάν η πράξη εξαρτάται από το  $\mathbf{D}$ . Σε ότι ακολουθεί, η φράση «σε σχέση με το  $\mathbf{D}$ » («σ.σ.μ. το  $\mathbf{D}$ » για συντομία) δηλώνει εξάρτηση από ένα σύνολο διαστάσεων  $\mathbf{D}$ .

Ένα σημαντικό σημείο είναι ότι το σύνολο διαστάσεων  $\mathbf{D}$  που θεωρούμε για ένα δεδομένο πρόβλημα μπορεί να διαφοροποιείται από περίπτωση σε περίπτωση. Αντίθετα, το πεδίο ορισμού  $\mathbf{V}_d$  μιας διάστασης  $d$  θεωρείται σε οποιαδήποτε περίπτωση σταθερό και αμετάβλητο.

Στα πλαίσια ενός μεμονωμένου προβλήματος, το σύνολο διαστάσεων  $\mathbf{D}$  πρέπει να περιέχει τουλάχιστον τις διαστάσεις εκείνες που συμμετέχουν στα εμπλεκόμενα περιβάλλοντα. Χρησιμοποιούμε το  $\mathbf{D}_{\min}$  για να συμβολίσουμε το σύνολο που περιέχει όλες τις διαστάσεις που εμφανίζονται στα περιβάλλοντα ενός δεδομένου προβλήματος και μόνον αυτές.

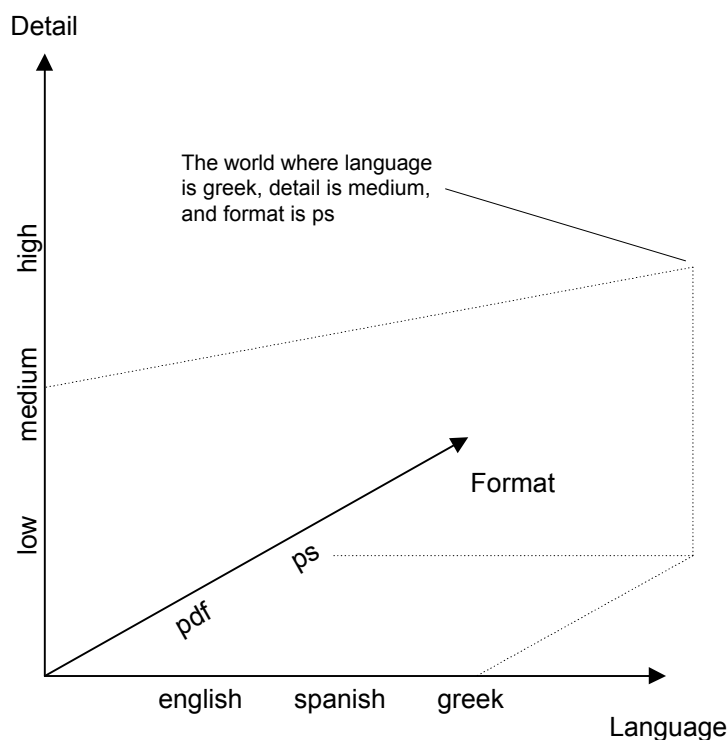
### 3.2.2 Πιθανοί κόσμοι

Ένας εναλλακτικός τρόπος για να δει κανείς το ερμηνευτικό περιβάλλον είναι μέσα από την έννοια του *κόσμου*, η οποία είναι θεμελιώδης στα MSSD. Ένας κόσμος αντιπροσωπεύει ένα περιβάλλον κάτω από το οποίο τα δεδομένα αποκτούν υπόσταση, και παρέχει ολοκληρωμένη και συγκεκριμένη καθοδήγηση για την ερμηνεία της πληροφορίας. Αν θεωρήσουμε ότι οι διαστάσεις και τα πεδία ορισμού των ορίζουν έναν πολυδιάστατο χώρο,

<sup>8</sup> Ο όρος *διάσταση* χρησιμοποιείται επίσης στις Αποθήκες Δεδομένων (Data Warehousing) [CD97, Coll96] για να περιγράψει δεδομένα που προσάπτονται σε κομμάτια πληροφορίας. Οι διαστάσεις στις Αποθήκες Δεδομένων δεν είναι πάντα ορθογώνιες και τα πεδία ορισμού τους οργανώνονται συχνά με βάση μια δενδρική δομή: θεωρήστε για παράδειγμα την διάσταση *continent* με τιμές *america*, *europa*, *asia*, *africa*, *australia*, και την διάσταση *country* με τιμές τις χώρες κάθε ηπείρου. Η διαφορά ανάμεσα στις έννοιες της διάστασης στις Αποθήκες Δεδομένων και στα MSSD είναι ότι, στις Αποθήκες Δεδομένων οι διαστάσεις χρησιμοποιούνται κυρίως για την πλοήγηση και την εκτέλεση συναθροιστικών πράξεων στα δεδομένα, ενώ στα MSSD οι διαστάσεις χρησιμοποιούνται για να δηλώσουν ένα περιβάλλον ερμηνείας των δεδομένων.

τότε κάθε σημείο αυτού του χώρου αναπαριστά έναν πιθανό κόσμο, όπως απεικονίζεται στην Εικόνα 3.1.

**Εικόνα 3.1: Σημείο στον πολυδιάστατο χώρο που αναπαριστά έναν πιθανό κόσμο.**



Ένας κόσμος ορίζεται μέσω της ανάθεσης μιας και μόνο τιμής σε κάθε διάσταση, όπως διατυπώνεται αυστηρά στον παρακάτω ορισμό.

### Ορισμός 3.1

*Ας είναι το  $\mathbf{D}$  ένα μη κενό σύνολο ονομάτων διαστάσεων, και για κάθε  $d \in \mathbf{D}$  ας είναι το  $\mathbf{V}_d$  το πεδίο ορισμού του  $d$ , με  $\mathbf{V}_d \neq \emptyset$ . Ένας **κόσμος**  $w$  σε σχέση με το  $\mathbf{D}$  είναι ένα σύνολο ζευγών  $(d, v)$ , με  $d \in \mathbf{D}$  και  $v \in \mathbf{V}_d$ , ώστε  $\forall d \in \mathbf{D}$  ακριβώς ένα  $(d, v)$  ανήκει στο  $w$ .*

Για να διαπιστώσουμε εάν ένα σύνολο ζευγών (διάσταση, τιμή) είναι κόσμος, το σύνολο των διαστάσεων  $\mathbf{D}$  πρέπει να ληφθεί υπόψη. Ένα σύνολο ζευγών που αντιπροσωπεύει κάποιον κόσμο σε σχέση με το  $\mathbf{D}$ , δεν αντιπροσωπεύει κανένα κόσμο σε σχέση με οποιοδήποτε άλλο σύνολο  $\mathbf{D}'$ .

Σαν παράδειγμα, θεωρήστε τα σύνολα  $\mathbf{D}_1 = \{\text{language, format}\}$  και  $\mathbf{D}_2 = \{\text{language, format, detail}\}$ , με πεδία ορισμού όπως φαίνονται στην Εικόνα 3.1. Το σύνολο  $\{(\text{language, greek}), (\text{format, pdf})\}$  είναι κόσμος σε σχέση με το  $\mathbf{D}_1$ , ενώ το σύνολο  $\{(\text{language, greek}), (\text{format, pdf}), (\text{detail, medium})\}$  είναι κόσμος σε σχέση με το  $\mathbf{D}_2$ .

Το σύνολο όλων των πιθανών κόσμων σε σχέση με ένα σύνολο διαστάσεων **D** καλείται **σύμπαν του D**. Για παράδειγμα, το σύμπαν του **D<sub>1</sub>** είναι το σύνολο:

$$\{ \{ (language, english), (format, pdf) \}, \\ \{ (language, english), (format, ps) \}, \{ (language, spanish), (format, pdf) \}, \\ \{ (language, spanish), (format, ps) \}, \{ (language, greek), (format, pdf) \}, \\ \{ (language, greek), (format, ps) \} \}$$

### Ορισμός 3.2

*Το σύμπαν του **D**, συμβολίζεται  $U_D$ , και είναι το σύνολο όλων των πιθανών κόσμων σε σχέση με το **D**.*

Η έννοια του κόσμου κάνει πιο κατανοητό το ερμηνευτικό περιβάλλον, που μπορεί να θεωρηθεί ότι ορίζει ένα σύνολο κόσμων με συμπαγή τρόπο. Στην ακόλουθη ενότητα εξηγήσουμε πως το περιβάλλον εκφράζεται μέσω διαστάσεων, και διάφορες πράξεις που μπορούν να εκτελεστούν στα περιβάλλοντα – συμπεριλαμβανομένων πράξεων που μετατρέπουν ένα ερμηνευτικό περιβάλλον στο σύνολο των κόσμων που αντιπροσωπεύει.

## 3.3 ΠΡΟΣΔΙΟΡΙΣΤΕΣ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΚΑΙ ΠΡΑΞΕΙΣ ΠΕΡΙΒΑΛΛΟΝΤΟΣ

Το ερμηνευτικό περιβάλλον στα MSSD εκφράζεται μέσω **προσδιοριστών περιβάλλοντος**, που είναι συντακτικές κατασκευές σχηματισμένες από διαστάσεις και τιμές διαστάσεων. Οι προσδιοριστές περιβάλλοντος προσάπτονται σε κομμάτια ημιδομημένων δεδομένων, και ορίζουν τους κόσμους κάτω από τους οποίους υφίστανται τα κομμάτια αυτά. Έτσι, οι προσδιοριστές περιβάλλοντος είναι η πιο σημαντική διαφορά ανάμεσα στα πολυδιάστατα και τα συμβατικά ημιδομημένα δεδομένα.

Καθώς ένας προσδιοριστής περιβάλλοντος μπορεί να ερμηνευτεί σαν ένα σύνολο κόσμων, οι βασικές πράξεις ανάμεσα από προσδιοριστές περιβάλλοντος είναι παρόμοιες με τις συμβατικές πράξεις συνόλων. Στην ενότητα αυτή ορίζουμε την **τομή περιβάλλοντος**, **ένωση περιβάλλοντος**, **διαφορά περιβάλλοντος**, **ισότητα περιβάλλοντος**, και **υπερσύνολο / υποσύνολο περιβάλλοντος**. Επιπλέον, οι πράξεις της **επέκτασης περιβάλλοντος** και **ανάπτυξης περιβάλλοντος** χρησιμοποιούνται για να πάρουμε το σύνολο των κόσμων που αναπαριστά ένας προσδιοριστής περιβάλλοντος. Επιπρόσθετες πράξεις των προσδιοριστών περιβάλλοντος, όπως για παράδειγμα το Καρτεσιανό γινόμενο δυο προσδιοριστών περιβάλλοντος, θα μπορούσαν επίσης να οριστούν. Στην ενότητα αυτή όμως, δεν εισάγουμε κάθε δυνατή πράξη περιβάλλοντος, αλλά εστιάζουμε στις πράξεις που θα φανούν χρήσιμες στα επόμενα κεφάλαια. Επιπρόσθετες πράξεις όπως το Καρτεσιανό γινόμενο δεν αποκλείονται, πάντως δεν ανέκυψε ανάγκη για τέτοιες πράξεις στα πλαίσια της παρούσας διατριβής.

Ο προσδιοριστής περιβάλλοντος είναι μια σύνθετη δομή που βασίζεται σε μια απλούστερη, η οποία ονομάζεται **όρος προσδιοριστή περιβάλλοντος**. Συζητούμε πρώτα τους όρους προσδιοριστή περιβάλλοντος και τις πράξεις τους, και έπειτα συνεχίζουμε με την εξέταση των προσδιοριστών περιβάλλοντος και των δικών τους πράξεων.

### 3.3.1 Όροι προσδιοριστή περιβάλλοντος

Ο όρος **προσδιοριστή περιβάλλοντος**<sup>9</sup> (ή απλά **όρος**) βασίζεται σε μια δομή που λέγεται **προσδιοριστής διάστασης**. Ένας προσδιοριστής διάστασης εκφράζει περιορισμούς στην τιμή μιας διάστασης. Για παράδειγμα, θεωρήστε την πρόταση «η τιμή της διάστασης language είναι english ή spanish»· ο αντίστοιχος προσδιοριστής διάστασης είναι  $(\text{language}, \{\text{english}, \text{spanish}\})$ . Η σύζευξη τέτοιων περιορισμών, όπου καθένας αντιστοιχεί σε διαφορετική διάσταση, σχηματίζει έναν όρο προσδιοριστή περιβάλλοντος. Ο όρος προσδιοριστή περιβάλλοντος  $\{(\text{language}, \{\text{greek}\}), (\text{detail}, \{\text{low}, \text{medium}\})\}$  ορίζει τους κόσμους στους οποίους η γλώσσα είναι Ελληνικά και η λεπτομέρεια είναι είτε χαμηλή ή μεσαία.

#### Ορισμός 3.3

*Ας είναι το  $\mathbf{D}$  ένα μη κενό σύνολο ονομάτων διαστάσεων, και για κάθε  $d \in \mathbf{D}$  ας είναι το  $\mathbf{V}_d$  το πεδίο ορισμού του  $d$ , με  $\mathbf{V}_d \neq \emptyset$ . Ο προσδιοριστής διάστασης  $s$  για μια διάσταση  $d$  είναι ένα ζεύγος  $(d, V)$  όπου  $d \in \mathbf{D}$  και  $V \in 2^{\mathbf{V}_d}$ , το δυναμοσύνολο του  $\mathbf{V}_d$ . Ο όρος προσδιοριστή περιβάλλοντος  $c^{cl}$  είναι ένα σύνολο προσδιοριστών διάστασης, τέτοιο ώστε για κάθε διάσταση  $d \in \mathbf{D}$  υπάρχει το πολύ ένας προσδιοριστής διάστασης  $(d, V)$  στο  $c^{cl}$ .*

Κάθε αριθμός διαστάσεων μπορεί να χρησιμοποιηθεί για να σχηματίσει έναν όρο προσδιοριστή περιβάλλοντος· με άλλα λόγια, δεν είναι απαραίτητο για έναν όρο προσδιοριστή περιβάλλοντος να περιέχει ένα στοιχείο (προσδιοριστή διάστασης) για κάθε δυνατή διάσταση. Η παράλειψη διαστάσεων αποτελεί μια βολική συντομογραφία της οποίας το νόημα θα αποσαφηνιστεί όταν δείξουμε πως ένας όρος μετατρέπεται σε σύνολο κόσμων.

Στα ακόλουθα, το σύμβολο  $^{cl}$ , που υπονοεί την λέξη *όρος* (*clause*), θα χρησιμοποιηθεί για να σηματοδοτεί μεταβλητές και πράξεις όρων προσδιοριστή περιβάλλοντος. Η σύμβαση αυτή θα είναι χρήσιμη αργότερα για την διάκριση μεταβλητών και πράξεων όρων προσδιοριστή περιβάλλοντος από μεταβλητές και πράξεις προσδιοριστών περιβάλλοντος.

#### 3.3.1.1 Αντιστοιχία με κόσμους

Η αντιστοιχία ανάμεσα από έναν όρο προσδιοριστή περιβάλλοντος και το σύνολο των κόσμων που αντιπροσωπεύει ορίζεται ακολούθως.

#### Ορισμός 3.4

*Ας είναι το  $\mathbf{D}$  ένα μη κενό σύνολο ονομάτων διαστάσεων, και για κάθε  $d \in \mathbf{D}$  ας είναι το  $\mathbf{V}_d$  το πεδίο ορισμού του  $d$ , με  $\mathbf{V}_d \neq \emptyset$ . Ας είναι το  $c^{cl}$  ένας όρος προσδιοριστή περιβάλλοντος, και το  $w$  ένας κόσμος σε σχέση με το  $\mathbf{D}$ . Τότε, το  $w$  αντιπροσωπεύεται από το  $c^{cl}$  εάν και μόνον εάν για κάθε ζεύγος  $(d, v) \in w$ , είτε  $(d, V) \in c^{cl}$  με  $v \in V$ , ή δεν υπάρχει ζεύγος  $(d, V)$  στο  $c^{cl}$ .*

Το σύνολο κόσμων που αντιπροσωπεύει ένας όρος προσδιοριστή περιβάλλοντος μπορεί να αποκτηθεί μέσω των πράξεων της επέκτασης όρου και ανάπτυξης όρου, που ορίζονται στην συνέχεια.

<sup>9</sup> Η δομή αυτή καλείται *όρος προσδιοριστή περιβάλλοντος* επειδή μοιάζει με τον Όρο Horn [Ull88, Kel97]:  $p_1 \wedge \dots \wedge p_n \rightarrow q$ , όπου  $p_i$  αντιστοιχεί σε προσδιοριστές διάστασης και  $q$  στην αντίστοιχη έκφραση της πολυδιάστατης οντότητας.



### Επεκτεταμένη μορφή όρου

Η **επέκταση όρου**  $\Theta_D^{cl}$  είναι μια πράξη που δέχεται έναν όρο προσδιοριστή περιβάλλοντος  $c^{cl}_1$  και επιστρέφει έναν άλλο όρο προσδιοριστή περιβάλλοντος  $c^{cl}_2$ . Η επέκταση όρου εκτελείται σε σχέση με ένα σύνολο διαστάσεων  $\mathbf{D}$  και με τα σύνολα πεδίων ορισμού των διαστάσεων  $\mathbf{V}_d$  για κάθε  $d$  στο  $\mathbf{D}$ .

$\Theta_D^{cl} : C^{cl} \rightarrow C^{cl}$ , όπου  $C^{cl}$  είναι το σύνολο όλων των όρων

Το αποτέλεσμα  $c^{cl}_2$  λέγεται **επεκτεταμένη μορφή** του όρου  $c^{cl}_1$  σε σχέση με το  $\mathbf{D}$ , και ορίζεται στην συνέχεια:

#### Ορισμός 3.5

Ας είναι το  $\mathbf{D}$  ένα μη κενό σύνολο ονομάτων διαστάσεων, και για κάθε  $d \in \mathbf{D}$  ας είναι το  $\mathbf{V}_d$  το πεδίο ορισμού του  $d$ , με  $\mathbf{V}_d \neq \emptyset$ . Ας είναι το  $c^{cl}$  ένας όρος προσδιοριστή περιβάλλοντος. Τότε η **επέκταση όρου** του  $c^{cl}$  σε σχέση με το  $\mathbf{D}$  (λέγεται επίσης και **επεκτεταμένη μορφή** του  $c^{cl}$  σ.σ.μ. το  $\mathbf{D}$ ), συμβολίζεται  $\Theta_D^{cl} c^{cl}$ , και είναι ένας όρος προσδιοριστή περιβάλλοντος που περιέχει στοιχεία ως ακολούθως: (a) εάν  $(d, V) \in c^{cl}$  τότε  $(d, V) \in \Theta_D^{cl} c^{cl}$ , και (b) εάν  $d \in \mathbf{D}$  και δεν υπάρχει ζεύγος  $(d, V)$  τέτοιο ώστε  $(d, V) \in c^{cl}$  τότε  $(d, \mathbf{V}_d) \in \Theta_D^{cl} c^{cl}$ .

Η επεκτεταμένη μορφή ενός όρου προσδιοριστή περιβάλλοντος είναι **ισοδύναμη** με τον όρο αυτό σε σχέση με το  $\mathbf{D}$ , με την έννοια ότι αντιπροσωπεύουν και οι δυο το ίδιο σύνολο κόσμων σε σχέση με το  $\mathbf{D}$ .

#### Πρόταση 3.1

Ένας όρος προσδιοριστή περιβάλλοντος και η επεκτεταμένη μορφή του σε σχέση με το  $\mathbf{D}$  αντιπροσωπεύουν το ίδιο σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$ , για οποιοδήποτε σύνολο διαστάσεων  $\mathbf{D}$ .

Ο ρόλος των παραλειπομένων διαστάσεων είναι πλέον εμφανής: για κάθε διάσταση του  $\mathbf{D}$  που δεν έχει αντίστοιχο προσδιοριστή διάστασης σε έναν όρο, υπονοείται η ύπαρξη στον όρο αυτό ενός προσδιοριστή διάστασης που περιέχει όλες τις τιμές του πεδίου ορισμού της διάστασης. Οι προσδιοριστές διάστασης στην επεκτεταμένη μορφή ενός όρου έχουν πάντοτε ένα-προς-ένα αντιστοιχία<sup>10</sup> με τις διαστάσεις του  $\mathbf{D}$ . Προφανώς ο ίδιος όρος προσδιοριστή περιβάλλοντος θα έχει διαφορετικές επεκτεταμένες μορφές σε σχέση με διαφορετικά σύνολα διαστάσεων.

#### Παράδειγμα 3.1

Θεωρήστε ότι:

$\mathbf{D} = \{\text{lang, detail, format}\}$

$\mathbf{V}_{\text{lang}} = \{\text{en, gr, sp}\}$

$\mathbf{V}_{\text{detail}} = \{\text{low, med, high}\}$

$\mathbf{V}_{\text{format}} = \{\text{pdf, ps}\}$

$c^{cl}_1 = \{(\text{lang}, \{\text{en, sp}\}), (\text{format}, \{\text{pdf}\})\}$

$c^{cl}_2 = \{(\text{lang}, \{\text{en}\}), (\text{detail}, \{\text{med, high}\}), (\text{format}, \{\text{ps}\})\}$

$c^{cl}_3 = \{(\text{lang}, \{\text{gr}\}), (\text{format}, \{\})\}$

$c^{cl}_4 = \{\}$

Τότε, οι επεκτεταμένες μορφές σε σχέση με το  $\mathbf{D}$  είναι:

<sup>10</sup> Όπως αναφέρει ο Ορισμός 3.3, αν  $(d, V)$  ανήκει σε έναν όρο προσδιοριστή περιβάλλοντος, τότε το  $d$  πρέπει να ανήκει στο σύνολο διαστάσεων  $\mathbf{D}$ .

$$\begin{aligned}
\odot_{\mathbf{D}}^{c^1} c^{c^1_1} &= \{(\text{lang}, \{\text{en}, \text{sp}\}), (\text{format}, \{\text{pdf}\}), (\text{detail}, \{\text{low}, \text{med}, \text{high}\})\} \\
\odot_{\mathbf{D}}^{c^1} c^{c^1_2} &= \{(\text{lang}, \{\text{en}\}), (\text{detail}, \{\text{med}, \text{high}\}), (\text{format}, \{\text{ps}\})\} \\
\odot_{\mathbf{D}}^{c^1} c^{c^1_3} &= \{(\text{lang}, \{\text{gr}\}), (\text{format}, \{\}), (\text{detail}, \{\text{low}, \text{med}, \text{high}\})\} \\
\odot_{\mathbf{D}}^{c^1} c^{c^1_4} &= \\
&\{(\text{lang}, \{\text{en}, \text{gr}, \text{sp}\}), (\text{detail}, \{\text{low}, \text{med}, \text{high}\}), (\text{format}, \{\text{pdf}, \text{ps}\})\}
\end{aligned}$$

◆

Στο Παράδειγμα 3.1, το  $\odot_{\mathbf{D}}^{c^1} c^{c^1_2}$  είναι το ίδιο όπως το  $c^{c^1_2}$ , αφότου ο όρος αυτός περιέχει προσδιοριστές διάστασης για κάθε διάσταση στο  $\mathbf{D}$ . Το κενό σύνολο είναι ένας έγκυρος όρος προσδιοριστή περιβάλλοντος, και η επεκτεταμένη μορφή του καλύπτει ολόκληρα τα πεδία ορισμού όλων των διαστάσεων στο  $\mathbf{D}$ , όπως φαίνεται από το  $c^{c^1_4}$  στο Παράδειγμα 3.1. Παρατηρήστε ότι η επεκτεταμένη μορφή ενός όρου δεν μπορεί ποτέ να είναι το κενό σύνολο, μια και η επέκταση θα προσθέσει προσδιοριστές διάστασης για κάθε  $d$  στο  $\mathbf{D}$ .

### Ανάπτυξη όρου

Η **ανάπτυξη όρου**  $\times^{c^1}$  είναι μια πράξη που δέχεται έναν όρο προσδιοριστή περιβάλλοντος  $c^{c^1}$ , με  $c^{c^1} \neq \emptyset$ , και επιστρέφει ένα σύνολο από σύνολα ζευγών (διάσταση, τιμή).

$$\times^{c^1} : (C^{c^1} - \emptyset) \rightarrow S, \text{ όπου } C^{c^1} \text{ είναι το σύνολο όλων των όρων, και το } S \text{ περιέχει σύνολα συνόλων ζευγών } (d, v)$$

Η ανάπτυξη όρου ορίζεται στην συνέχεια.

#### Ορισμός 3.6

Η **ανάπτυξη όρου** ενός όρου προσδιοριστή περιβάλλοντος  $c^{c^1} = \{(d_1, V_1), (d_2, V_2), \dots, (d_n, V_n)\}$ , με  $n \geq 1$ , συμβολίζεται  $\times^{c^1} c^{c^1}$ , και είναι ένα σύνολο που ορίζεται ως ακολούθως: εάν  $(d, \emptyset) \in c^{c^1}$ , τότε  $\times^{c^1} c^{c^1} = \emptyset$ , αλλιώς  $\times^{c^1} c^{c^1} = \{(d_1, v_1), (d_2, v_2), \dots, (d_n, v_n)\} \mid v_i \in V_i, \text{ με } 1 \leq i \leq n\}$ .

Συνεπώς, η ανάπτυξη ενός όρου  $c^{c^1}$  είναι ένα σύνολο συνόλων ζευγών (διάσταση, τιμή). Για να αποφασίσουμε εάν ένα τέτοιο σύνολο ζευγών (διάσταση, τιμή) αντιπροσωπεύει έναν κόσμο, όπως διευκρινίζει ο Ορισμός 3.1 πρέπει να πάρουμε υπόψη το σύνολο των διαστάσεων  $\mathbf{D}$ . Εύκολα διαπιστώνουμε ότι, αν οι προσδιοριστές διάστασης  $(d, V)$  στο  $c^{c^1}$  έχουν ένα-προς-ένα αντιστοιχία με τις διαστάσεις του  $\mathbf{D}$ , τότε το  $\times^{c^1} c^{c^1}$  δίνει το σύνολο των κόσμων σ.σ.μ. το  $\mathbf{D}$  που αντιπροσωπεύει το  $c^{c^1}$ . Όμως, η επέκταση όρου εγγυάται αυτή την συνθήκη: το γεγονός αυτό, ο Ορισμός 3.4, και η Πρόταση 3.1 οδηγούν στο εξής:

#### Πρόταση 3.2

Το σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$  που αντιπροσωπεύει ένας όρος προσδιοριστή περιβάλλοντος  $c^{c^1}$ , συμβολίζεται  $W_{\mathbf{D}}(c^{c^1})$ , και δίνεται από την ανάπτυξη όρου της επεκτεταμένης μορφής σ.σ.μ. το  $\mathbf{D}$  του  $c^{c^1}$ .

$$W_{\mathbf{D}}(c^{c^1}) = \times^{c^1} (\odot_{\mathbf{D}}^{c^1} c^{c^1})$$

**Τύπος 3.1**

#### Παράδειγμα 3.2

Θεωρήστε ότι:

$$\begin{aligned}
\mathbf{D} &= \{\text{lang}, \text{detail}\} \\
V_{\text{lang}} &= \{\text{en}, \text{gr}, \text{sp}\} \\
V_{\text{detail}} &= \{\text{low}, \text{high}\}
\end{aligned}$$

$$\begin{aligned} c^{cl}_1 &= \{ (\text{lang}, \{ \text{en}, \text{gr} \}) \} \\ c^{cl}_2 &= \{ (\text{lang}, \{ \text{en} \}), (\text{detail}, \{ \text{low}, \text{high} \}) \} \\ c^{cl}_3 &= \{ (\text{detail}, \{ \}) \} \end{aligned}$$

Τότε, ισχύουν τα ακόλουθα:

1.  $\times^{cl} c^{cl}_1 = \{ \{ (\text{lang}, \text{en}) \}, \{ (\text{lang}, \text{gr}) \} \}$   
 $\times^{cl} (\otimes^{cl}_D c^{cl}_1) =$   
 $\{ \{ (\text{lang}, \text{en}), (\text{detail}, \text{low}) \}, \{ (\text{lang}, \text{en}), (\text{detail}, \text{high}) \},$   
 $\{ (\text{lang}, \text{gr}), (\text{detail}, \text{low}) \}, \{ (\text{lang}, \text{gr}), (\text{detail}, \text{high}) \} \}$
  2.  $\times^{cl} c^{cl}_2 = \times^{cl} (\otimes^{cl}_D c^{cl}_2) =$   
 $\{ \{ (\text{lang}, \text{en}), (\text{detail}, \text{low}) \}, \{ (\text{lang}, \text{en}), (\text{detail}, \text{high}) \} \}$
  3.  $\times^{cl} c^{cl}_3 = \times^{cl} (\otimes^{cl}_D c^{cl}_3) = \emptyset$
- ◆

Σε αντίθεση με την περίπτωση (1) στο Παράδειγμα 3.2, όπου η εισαγωγή της επέκτασης άλλαξε τα αποτελέσματα της ανάπτυξης όρου, στην περίπτωση (2) η ανάπτυξη του  $c^{cl}_2$  είναι ίση με την ανάπτυξη της επεκτεταμένης μορφής του  $c^{cl}_2$ . Στην περίπτωση (1) στο Παράδειγμα 3.2, η ανάπτυξη  $\times^{cl} c^{cl}_1$  δεν δίνει κόσμους σ.σ.μ. το  $\mathbf{D}$ , επειδή δεν υπάρχουν ζεύγη για την διάσταση *detail*. Παρατηρήστε πάντως, ότι αν το σύνολο διαστάσεων  $\mathbf{D}$  ήταν  $\{ \text{lang} \}$ , τότε το  $\times^{cl} c^{cl}_1$  θα έδινε κόσμους σ.σ.μ. το  $\mathbf{D}$ . Η παρατήρηση αυτή μπορεί να γενικευθεί: *ένας όρος προσδιοριστή περιβάλλοντος που αναπαριστά έναν και μόνο κόσμο σ.σ.μ. το  $\mathbf{D}$ , αναπαριστά ένα (μη κενό) σύνολο κόσμων σ.σ.μ. οποιοδήποτε  $\mathbf{D}' \supset \mathbf{D}$ .*

### Ανάπτυξη επεκτεταμένου όρου

Η επέκταση και η ανάπτυξη χρησιμοποιούνται συχνά μαζί για να δώσουν το σύνολο των κόσμων σ.σ.μ. το  $\mathbf{D}$  που αναπαριστά ένας όρος. Παρακάτω εισάγεται μια πράξη που συνδυάζει την επέκταση και την ανάπτυξη, σαν συντομογραφία.

Η **ανάπτυξη επεκτεταμένου όρου**  $\otimes^{cl}_D$  είναι μια πράξη που δέχεται έναν όρο προσδιοριστή περιβάλλοντος  $c^{cl}$  και επιστρέφει ένα σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$ . Η ανάπτυξη επεκτεταμένου όρου εκτελείται σε σχέση με ένα σύνολο διαστάσεων  $\mathbf{D}$  και με τα πεδία ορισμού των διαστάσεων  $\mathbf{V}_d$  για κάθε  $d$  στο  $\mathbf{D}$ .

$$\otimes^{cl}_D : C^{cl} \rightarrow S_W, \text{ όπου } C^{cl} \text{ είναι το σύνολο όλων των όρων, and}$$

$$\text{το } S_W \text{ περιέχει τα σύνολα κόσμων σ.σ.μ. το } \mathbf{D}$$

Η ανάπτυξη επεκτεταμένου όρου ορίζεται ακολούθως.

#### Ορισμός 3.7

*Η ανάπτυξη επεκτεταμένου όρου ενός όρου προσδιοριστή περιβάλλοντος  $c^{cl}$ , συμβολίζεται  $\otimes^{cl}_D c^{cl}$ , και ορίζεται ως εξής:  $\otimes^{cl}_D c^{cl} = \times^{cl} (\otimes^{cl}_D c^{cl})$ .*

Με βάση τα παραπάνω, ο Τύπος 3.1 μπορεί επίσης να πάρει την μορφή:

$$W_D(c^{cl}) = \otimes^{cl}_D c^{cl}$$

### 3.3.1.2 Ισότητα όρου

Η **ισότητα όρου**  $=^{cl}$  είναι μια πράξη που δέχεται δυο όρους προσδιοριστή περιβάλλοντος  $c^{cl}_1, c^{cl}_2$  και επιστρέφει μια τιμή αληθείας (Boolean).

$=^{cl} : (C^{cl} \times C^{cl}) \rightarrow \text{Boolean}$ , όπου  $C^{cl}$  το σύνολο όλων των όρων

Η ισότητα όρου και η ανισότητα όρου ορίζονται ως εξής:

### Ορισμός 3.8

Ας είναι το  $\mathbf{D}$  ένα μη κενό σύνολο ονομάτων διαστάσεων, και για κάθε  $d \in \mathbf{D}$  ας είναι το  $\mathbf{V}_d$  το πεδίο ορισμού του  $d$ , με  $\mathbf{V}_d \neq \emptyset$ . Ας είναι τα  $c^{cl}_1, c^{cl}_2$  όροι προσδιοριστή περιβάλλοντος. Τότε τα  $c^{cl}_1$  και  $c^{cl}_2$  είναι **ίσοι-όροι**, που συμβολίζεται  $c^{cl}_1 =^{cl} c^{cl}_2$ , εάν και μόνον εάν  $W_D(c^{cl}_1) = W_D(c^{cl}_2)$ , όπου  $W_D(c^{cl})$  είναι το σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$  που αναπαριστά ένας όρος  $c^{cl}$ . Εάν  $W_D(c^{cl}_1) \neq W_D(c^{cl}_2)$ , τότε και μόνο τα  $c^{cl}_1$  και  $c^{cl}_2$  είναι **άνισοι-όροι**, που συμβολίζεται  $c^{cl}_1 \neq^{cl} c^{cl}_2$ .

Παρόλο που η ισότητα όρου ορίζεται με αναφορά σε ένα σύνολο διαστάσεων  $\mathbf{D}$ , στην πραγματικότητα δεν ορίζεται σε σχέση με κάποιο συγκεκριμένο  $\mathbf{D}^{11}$ . Η ακόλουθη Πρόταση 3.3 δείχνει ότι αν δυο όροι είναι ίσοι-όροι σε σχέση με ένα σύνολο διαστάσεων  $\mathbf{D}$ , είναι επίσης ίσοι-όροι σε σχέση με οποιοδήποτε άλλο  $\mathbf{D}'$ . Το σύμβολο  $=^{cl}_D$  χρησιμοποιείται στην Πρόταση 3.3 για να εκφράσει ισότητα όρου σε σχέση με το  $\mathbf{D}$ .

### Πρόταση 3.3

Ας είναι τα  $c^{cl}_1$  και  $c^{cl}_2$  όροι προσδιοριστή περιβάλλοντος και τα  $\mathbf{D}, \mathbf{D}'$  σύνολα διαστάσεων. Ισχύουν τα παρακάτω: (a) αν  $c^{cl}_1 =^{cl}_D c^{cl}_2$  (σε σχέση με το  $\mathbf{D}$ ), τότε  $c^{cl}_1 =^{cl}_{D'} c^{cl}_2$  (σε σχέση με το  $\mathbf{D}'$ ), και (b) αν  $c^{cl}_1 \neq^{cl}_D c^{cl}_2$  (σε σχέση με το  $\mathbf{D}$ ), τότε  $c^{cl}_1 \neq^{cl}_{D'} c^{cl}_2$  (σε σχέση με το  $\mathbf{D}'$ ).

Η Πρόταση 3.3 βασίζεται στην παραδοχή, που αναφέρεται στην Ενότητα 3.2.1, ότι θεωρούμε σύνολα διαστάσεων που πάντοτε περιέχουν (τουλάχιστον) εκείνες τις διαστάσεις που απαντώνται στα εμπλεκόμενα περιβάλλοντα. Έτσι, αν το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται στα  $c^{cl}_1$  και  $c^{cl}_2$  και καμία άλλη<sup>12</sup>,  $c^{cl}_1 =^{cl}_D c^{cl}_2 \Leftrightarrow c^{cl}_1 =^{cl}_{D'} c^{cl}_2$  και  $c^{cl}_1 \neq^{cl}_D c^{cl}_2 \Leftrightarrow c^{cl}_1 \neq^{cl}_{D'} c^{cl}_2$ ,  $\forall \mathbf{D}$  και  $\mathbf{D}'$  με  $\mathbf{D}_{\min} \subseteq \mathbf{D}$  και  $\mathbf{D}_{\min} \subseteq \mathbf{D}'$ .

Η συμβατική ισότητα συνόλων δεν είναι αρκετή για την απευθείας σύγκριση δυο όρων προσδιοριστή περιβάλλοντος, επειδή είναι δυνατόν δυο όροι με διαφορετική μορφή να αναπαριστούν το ίδιο σύνολο κόσμων σε σχέση με ένα σύνολο διαστάσεων  $\mathbf{D}$ . Δυο όροι είναι **ίσοι-όροι** αν αναπαριστούν το ίδιο σύνολο κόσμων σ.σ.μ. κάποιο  $\mathbf{D}$ , άσχετα με την μορφή που έχουν.

### Παράδειγμα 3.3

Θεωρήστε ότι:

$\mathbf{D} = \{\text{lang, detail, format}\}$

$\mathbf{V}_{\text{lang}} = \{\text{en, gr, sp}\}$

$\mathbf{V}_{\text{detail}} = \{\text{low, high}\}$

$\mathbf{V}_{\text{format}} = \{\text{ps, pdf}\}$

$c^{cl}_1 = \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{high}\}), (\text{format}, \{\text{ps, pdf}\})\}$

$c^{cl}_2 = \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{high}\})\}$

$c^{cl}_3 = \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{high}\}), (\text{format}, \{\})\}$

$c^{cl}_4 = \{(\text{lang}, \{\}), (\text{detail}, \{\text{low, high}\})\}$

<sup>11</sup> Πάντως, ο έλεγχος δυο όρων για ισότητα όρου ίσως χρειαστεί να λάβει υπόψη τα πεδία ορισμού των διαστάσεων που εμφανίζονται στους όρους αυτούς.

<sup>12</sup> Στην οριακή περίπτωση όπου τα  $c^{cl}_1$  και  $c^{cl}_2$  δεν περιέχουν προσδιοριστές διάστασης (π.χ.  $c^{cl}_1 = c^{cl}_2 = \emptyset$ ), το  $\mathbf{D}_{\min}$  δεν μπορεί να οριστεί καθώς πρέπει να είναι μη κενό σύνολο. Στην περίπτωση αυτή, η ισότητα όρου ισχύει σ.σ.μ. κάθε δυνατό σύνολο διαστάσεων.

$$c^{cl}_5 = \{ (\text{detail}, \{ \text{low}, \text{high} \}) \}$$

$$c^{cl}_6 = \{ (\text{format}, \{ \text{ps}, \text{pdf} \}) \}$$

Τότε, ισχύουν τα παρακάτω:

$$c^{cl}_1 =^{cl} c^{cl}_2$$

$$c^{cl}_2 \neq^{cl} c^{cl}_3$$

$$c^{cl}_3 =^{cl} c^{cl}_4$$

$$c^{cl}_5 =^{cl} c^{cl}_6$$

◆

Στο Παράδειγμα 3.3, τα  $c^{cl}_1$  και  $c^{cl}_2$  αναπαριστούν το ίδιο σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$ , παρότι έχουν διαφορετική μορφή. Επίσης, όπως δείχνει ο Τύπος 3.1, τα  $c^{cl}_3$  και  $c^{cl}_4$  δεν αναπαριστούν κανέναν κόσμο σ.σ.μ. το  $\mathbf{D}$ , και έτσι είναι ίσοι-όροι. Οι όροι  $c^{cl}_5$  και  $c^{cl}_6$  είναι και πάλι ίσοι-όροι, καθώς αναπαριστούν κάθε πιθανό κόσμο σ.σ.μ. το  $\mathbf{D}$ .

### 3.3.1.3 Κενός όρος και καθολικός όρος

Ο **κενός όρος**  $\mathcal{E}^{cl}$  είναι ένα κοινό σύμβολο για του όρους που δεν αναπαριστούν κανέναν κόσμο σ.σ.μ. κάποιο σύνολο διαστάσεων  $\mathbf{D}$ . Από την άλλη, ο **καθολικός όρος**  $\mathcal{U}^{cl}$  είναι ένα κοινό σύμβολο για τους όρους που στην πράξη δεν περικλείουν κάποιον περιορισμό, και που αναπαριστούν κάθε πιθανό κόσμο σ.σ.μ. κάποιο σύνολο διαστάσεων  $\mathbf{D}$ .

#### Ορισμός 3.9

*Ας είναι το  $\mathbf{D}$  ένα μη κενό σύνολο ονομάτων διαστάσεων, και για κάθε  $d \in \mathbf{D}$  ας είναι το  $\mathbf{V}_d$  το πεδίο ορισμού του  $d$ , με  $\mathbf{V}_d \neq \emptyset$ . Ας είναι το  $c^{cl}$  ένας όρος προσδιοριστή περιβάλλοντος, και το  $W_D(c^{cl})$  το σύνολο κόσμων σε σχέση με το  $\mathbf{D}$  που αναπαριστά το  $c^{cl}$ . Εάν  $W_D(c^{cl}) = \emptyset$ , τότε και μόνον το  $c^{cl}$  καλείται **κενός όρος**, και συμβολίζεται  $\mathcal{E}^{cl}$ . Εάν  $W_D(c^{cl}) = U_D$ , όπου  $U_D$  είναι το σύμπαν του  $\mathbf{D}$ , τότε και μόνον το  $c^{cl}$  καλείται **καθολικός όρος**, και συμβολίζεται  $\mathcal{U}^{cl}$ .*

Παρόλο που οι έννοιες του κενού όρου και του καθολικού όρου ορίζονται με αναφορά σε ένα σύνολο διαστάσεων  $\mathbf{D}$ , στην πραγματικότητα δεν ορίζονται σε σχέση με κάποιο συγκεκριμένο  $\mathbf{D}$ <sup>13</sup>. Στην Ενότητα 3.2.1, μιλήσαμε για την απαίτηση ότι ένα σύνολο διαστάσεων  $\mathbf{D}$  πρέπει να περιέχει (τουλάχιστον) εκείνες τις διαστάσεις που απαντώνται στα εμπλεκόμενα περιβάλλοντα. Ας θεωρήσουμε ότι το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται σε έναν όρο  $c^{cl}$  και καμία άλλη. Εύκολα διαπιστώνει κανείς ότι αν ένας όρος  $c^{cl}$  είναι κενός όρος σ.σ.μ. το  $\mathbf{D}$ , τότε το  $c^{cl}$  είναι επίσης κενός όρος σ.σ.μ. κάθε άλλο  $\mathbf{D}'$ , όπου  $\mathbf{D}_{\min} \subseteq \mathbf{D}$  και  $\mathbf{D}_{\min} \subseteq \mathbf{D}'$ . Παρομοίως, αν το  $c^{cl}$  είναι ένας καθολικός όρος σ.σ.μ. το  $\mathbf{D}$ , τότε το  $c^{cl}$  είναι επίσης ένας καθολικός όρος σ.σ.μ. κάθε άλλο  $\mathbf{D}'$ , όπου  $\mathbf{D}_{\min} \subseteq \mathbf{D}$  και  $\mathbf{D}_{\min} \subseteq \mathbf{D}'$ <sup>14</sup>.

Ο Τύπος 3.1 και ο Ορισμός 3.9 δίνουν:

$$W_D(\mathcal{E}^{cl}) = \times^{cl} (\odot^{cl}_D \mathcal{E}^{cl}) = \emptyset$$

$$W_D(\mathcal{U}^{cl}) = \times^{cl} (\odot^{cl}_D \mathcal{U}^{cl}) = U_D$$

**Τύπος 3.2**

**Τύπος 3.3**

Ο Ορισμός 3.7 δείχνει ότι ο Τύπος 3.2 και ο Τύπος 3.3 μπορούν να πάρουν την μορφή:

<sup>13</sup> Πάντως, ο έλεγχος για το αν ένας όρος είναι κενός όρος ή καθολικός όρος χρειάζεται να λάβει υπόψη τα πεδία ορισμού των διαστάσεων που εμφανίζονται στον όρο αυτόν.

<sup>14</sup> Στην οριακή περίπτωση όπου  $c^{cl} = \emptyset$ , το  $\mathbf{D}_{\min}$  δεν μπορεί να οριστεί, καθώς πρέπει να είναι μη κενό σύνολο. Στην περίπτωση αυτή, το  $c^{cl}$  είναι καθολικός όρος σ.σ.μ. κάθε δυνατό σύνολο διαστάσεων.

$$\begin{aligned} W_D(\mathcal{E}^{cl}) &= \otimes_{\mathbf{D}}^{cl} \mathcal{E}^{cl} = \emptyset \\ W_D(\mathcal{U}^{cl}) &= \otimes_{\mathbf{D}}^{cl} \mathcal{U}^{cl} = U_D \end{aligned}$$

Ο Τύπος 3.3 και οι ορισμοί της ανάπτυξης όρου και επέκτασης όρου δείχνουν ότι, αν ένας όρος  $c^{cl}$  περιέχει έναν προσδιοριστή διάστασης της μορφής  $(d, \emptyset)$ , τότε η επεκτεταμένη μορφή του θα περιέχει επίσης το  $(d, \emptyset)$ , και η ανάπτυξη του  $c^{cl}$  θα δώσει το κενό σύνολο κόσμων. Επιπρόσθετα, για να αναπαριστά ένας όρος το κενό σύνολο κόσμων, πρέπει να περιέχει τουλάχιστον ένα προσδιοριστή διάστασης της μορφής  $(d, \emptyset)$ . Συνεπώς, το  $\mathcal{E}^{cl}$  συμβολίζει κάθε όρο  $c^{cl}$ , τέτοιον ώστε  $(d, \emptyset) \in c^{cl}$ . Οι όροι αυτοί είναι μεταξύ τους ίσοι-όροι καθώς αναπαριστούν όλοι ένα κενό σύνολο κόσμων σ.σ.μ. κάποιο  $\mathbf{D}$ .

Παρομοίως, αν ένας όρος  $c^{cl}$  περιέχει αποκλειστικά προσδιοριστές διάστασης της μορφής  $(d, \mathbf{V}_d)$  ή αν  $c^{cl} = \emptyset$ , τότε η επεκτεταμένη μορφή του θα περιέχει αποκλειστικά προσδιοριστές διάστασης της μορφής  $(d, \mathbf{V}_d)$ , και η ανάπτυξη του  $c^{cl}$  θα δώσει το σύμπαν του  $\mathbf{D}$ , το  $U_D$ . Επιπρόσθετα, για να αναπαριστά ένας όρος το  $U_D$ , πρέπει να περιέχει αποκλειστικά διαστάσεις της μορφής  $(d, \mathbf{V}_d)$  ή πρέπει να είναι το κενό σύνολο. Συνεπώς, το  $\mathcal{U}^{cl}$  συμβολίζει κάθε όρο  $c^{cl}$ , τέτοιον ώστε είτε το  $c^{cl}$  περιέχει αποκλειστικά προσδιοριστές διάστασης της μορφής  $(d, \mathbf{V}_d)$ , ή αλλιώς  $c^{cl} = \emptyset$ . Αυτοί οι όροι είναι μεταξύ τους ίσοι-όροι καθώς αναπαριστούν όλοι το σύμπαν  $U_D$  κάποιου  $\mathbf{D}$ .

Με βάση τα παραπάνω, εξυπακούεται ότι το  $\mathcal{E}^{cl}$  στην πραγματικότητα συμβολίζει κάθε στοιχείο ενός συνόλου ίσων-όρων, που αναπαριστούν το κενό σύνολο κόσμων σ.σ.μ. κάποιο  $\mathbf{D}$ . Ομοίως, το  $\mathcal{U}^{cl}$  συμβολίζει κάθε στοιχείο ενός συνόλου ίσων-όρων, που αναπαριστούν το σύμπαν κάποιου  $\mathbf{D}$ .

Σημειώστε ότι η επεκτεταμένη μορφή σ.σ.μ. το  $\mathbf{D}$  ενός κενού όρου είναι και πάλι ένας κενός όρος, όπως φαίνεται στο Παράδειγμα 3.1 με το  $c^{cl}_3$  και το  $\odot_{\mathbf{D}}^{cl} c^{cl}_3$ . Η περίπτωση του  $c^{cl}_4$  στο Παράδειγμα 3.1 δείχνει ότι η επεκτεταμένη μορφή σ.σ.μ. το  $\mathbf{D}$  ενός καθολικού όρου είναι και πάλι ένας καθολικός όρος, καθώς καλύπτει ολόκληρα τα πεδία ορισμού όλων των διαστάσεων στο  $\mathbf{D}$ .

Ένα ενδιαφέρον σημείο είναι ότι ο όρος  $\emptyset$  είναι καθολικός όρος για κάθε δυνατό σύνολο διαστάσεων. Ακολουθώντας, εισάγουμε ένα ειδικό σύμβολο<sup>15</sup> που αντιπροσωπεύει έναν όρο ο οποίος είναι κενός όρος για κάθε δυνατό σύνολο διαστάσεων.

### Ορισμός 3.10

*Το σύμβολο  $\{-\}$  συμβολίζει έναν όρο που περιέχει προσδιοριστές διάστασης της μορφής  $(d, \emptyset) \forall d \in \mathbf{D}$ .*

Έτσι, αν  $\mathbf{D} = \{\text{lang}\}$  τότε  $\{-\} = \{(\text{lang}, \emptyset)\}$ , ενώ αν  $\mathbf{D} = \{\text{detail}, \text{format}\}$  τότε  $\{-\} = \{(\text{detail}, \emptyset), (\text{format}, \emptyset)\}$ . Προφανώς,  $\odot_{\mathbf{D}}^{cl} \{-\} = \{-\}$  και  $W_D(\{-\}) = \emptyset$  για κάθε δυνατό σύνολο διαστάσεων.

Στα ακόλουθα, το σύμβολο της ισότητας σε εκφράσεις της μορφής  $c^{cl} = \mathcal{E}^{cl}$  υπονοεί ότι το  $c^{cl}$  είναι κάποιος από τους όρους προσδιοριστή περιβάλλοντος που αντιπροσωπεύει το  $\mathcal{E}^{cl}$ . Η παραπάνω ισότητα μπορεί επίσης να εκφραστεί χρησιμοποιώντας την ισότητα όρου  $c^{cl} =^{cl} \{-\}$ . Παρόμοια, το σύμβολο της ισότητας σε εκφράσεις της μορφής  $c^{cl} = \mathcal{U}^{cl}$  υπονοεί ότι το  $c^{cl}$

<sup>15</sup> Δεν είναι προφανές πως να εκφράσουμε έναν όρο που είναι κενός όρος σ.σ.μ. κάθε δυνατό σύνολο διαστάσεων. Για να γίνει αυτό κατανοητό, θεωρήστε την έκφραση  $\{(\text{lang}, \emptyset)\}$ . Η έκφραση αυτή είναι ένας κενός όρος σ.σ.μ.  $\mathbf{D} = \{\text{lang}\}$ , αλλά δεν είναι καν όρος σ.σ.μ.  $\mathbf{D}' = \{\text{detail}\}$ , επειδή, όπως αναφέραμε στην Ενότητα 3.2.1, όλες οι διαστάσεις που εμφανίζονται στα περιβάλλοντα πρέπει να περιέχονται στο σύνολο διαστάσεων.

είναι ένας από τους όρους προσδιοριστή περιβάλλοντος που αντιπροσωπεύει το  $\mathcal{Z}^{cl}$ . Αυτή η ισότητα μπορεί επίσης να εκφραστεί χρησιμοποιώντας την ισότητα όρου  $c^{cl} =^{cl} \emptyset$ .

### 3.3.1.4 Τομή όρου

Η **τομή όρου**  $\cap^{cl}$  είναι μια πράξη που δέχεται δυο όρους προσδιοριστή περιβάλλοντος  $c^{cl}_1, c^{cl}_2$  και επιστρέφει έναν όρο προσδιοριστή περιβάλλοντος  $c^{cl}_3$ .

$\cap^{cl} : (c^{cl} \times c^{cl}) \rightarrow c^{cl}$ , όπου  $c^{cl}$  είναι το σύνολο όλων των όρων

Η τομή όρου ορίζεται παρακάτω.

#### Ορισμός 3.11

Ας είναι  $c^{cl}_1, c^{cl}_2$  δυο όροι προσδιοριστή περιβάλλοντος. Η **τομή όρου**  $c^{cl}_1 \cap^{cl} c^{cl}_2$  είναι ένας όρος προσδιοριστή περιβάλλοντος  $c^{cl}_3 = \{(d, V) \mid (d, V) \in c^{cl}_1 \text{ και δεν υπάρχει ζεύγος } (d, V') \text{ τέτοιο ώστε } (d, V') \in c^{cl}_2\} \cup \{(d, V) \mid (d, V) \in c^{cl}_2 \text{ και δεν υπάρχει ζεύγος } (d, V') \text{ τέτοιο ώστε } (d, V') \in c^{cl}_1\} \cup \{(d, V) \mid (d, V_1) \in c^{cl}_1 \text{ και } (d, V_2) \in c^{cl}_2 \text{ και } V = V_1 \cap V_2\}$ .

Ο Ορισμός 3.11 δείχνει ότι, αν τα  $c^{cl}_1, c^{cl}_2$  δεν περιέχουν καθόλου προσδιοριστές διάστασης, τότε το αποτέλεσμα θα είναι και πάλι το κενό σύνολο (που σημαίνει κάθε πιθανό κόσμος):  $\emptyset \cap^{cl} \emptyset = \emptyset$ . Επιπλέον,  $c^{cl} \cap^{cl} \{-\} = \{-\}$  για κάθε σύνολο διαστάσεων.

#### Παράδειγμα 3.4

Θεωρήστε τα ακόλουθα:

$$\begin{aligned} c^{cl}_1 &= \{(\text{lang}, \{\text{sp}\}), (\text{format}, \{\text{pdf}\})\} \\ c^{cl}_2 &= \{(\text{lang}, \{\text{en}, \text{gr}\}), (\text{detail}, \{\text{med}, \text{high}\})\} \\ c^{cl}_3 &= \{(\text{lang}, \{\text{gr}, \text{en}, \text{sp}\})\} \\ c^{cl}_4 &= \{(\text{currency}, \{\}), (\text{lang}, \{\text{sp}, \text{gr}\})\} \end{aligned}$$

$$\begin{aligned} c^{cl}_1 \cap^{cl} c^{cl}_2 &= \{(\text{lang}, \{\}), (\text{format}, \{\text{pdf}\}), (\text{detail}, \{\text{med}, \text{high}\})\} \\ c^{cl}_2 \cap^{cl} c^{cl}_3 &= \{(\text{lang}, \{\text{en}, \text{gr}\}), (\text{detail}, \{\text{med}, \text{high}\})\} \\ c^{cl}_3 \cap^{cl} c^{cl}_4 &= \{(\text{currency}, \{\}), (\text{lang}, \{\text{sp}, \text{gr}\})\} \end{aligned}$$

◆

Ο Ορισμός 3.11 δείχνει γενικότερα ότι, για κάθε όρο προσδιοριστή περιβάλλοντος  $c^{cl}$ , η τομή όρου με τον καθολικό όρο  $\mathcal{Z}^{cl}$  δίνει πάντα έναν όρο που είναι ίσος-όρος με το  $c^{cl}$ , ενώ η τομή όρου με τον κενό όρο  $\mathcal{E}^{cl}$  δίνει πάντα τον κενό όρο.

$$\begin{aligned} c^{cl} \cap^{cl} \mathcal{Z}^{cl} &=^{cl} c^{cl} \\ c^{cl} \cap^{cl} \mathcal{E}^{cl} &= \mathcal{E}^{cl} \end{aligned}$$

**Τύπος 3.4**

**Τύπος 3.5**

Σημειώστε ότι ο Τύπος 3.4 χρησιμοποιεί ισότητα όρου αντί για συμβατική ισότητα. Συνεπώς, το αποτέλεσμα της τομής όρου που δίνει ο Τύπος 3.4 δεν είναι κατ' ανάγκη το  $c^{cl}$ , αλλά ένας όρος που είναι ίσος-όρος με το  $c^{cl}$ .

Η τομή όρου δυο όρων αναπαριστά τους κόσμους που είναι κοινοί και στους δυο όρους, όπως δείχνεται στην ακόλουθη πρόταση.

#### Πρόταση 3.4

Αν  $c^{cl}_1, c^{cl}_2$  είναι όροι προσδιοριστή περιβάλλοντος και  $W_D(c^{cl}_1), W_D(c^{cl}_2)$  είναι τα σύνολα κόσμων σ.σ.μ. το  $\mathbf{D}$  που αναπαριστούν, τότε  $W_D(c^{cl}_1 \cap^{cl} c^{cl}_2) = W_D(c^{cl}_1) \cap W_D(c^{cl}_2)$ .

Σημειώστε ότι η τομή όρου εκτελείται χωρίς να πάρουμε υπόψη το σύνολο διαστάσεων. Με άλλα λόγια, αν  $c_1^{cl} \cap^{cl} c_2^{cl} = c_3^{cl}$  σ.σ.μ. το  $\mathbf{D}$  και  $c_1^{cl} \cap^{cl} c_2^{cl} = c_4^{cl}$  σ.σ.μ. το  $\mathbf{D}'$ , τότε το  $c_3^{cl}$  και το  $c_4^{cl}$  είναι ταυτόσημα. Επιπρόσθετα, όπως δείχνουμε στην Πρόταση 3.4, το  $c_3^{cl}$  αναπαριστά τους κόσμους σ.σ.μ. το  $\mathbf{D}$  που είναι κοινοί στο  $c_1^{cl}$  και στο  $c_2^{cl}$ , και το  $c_4^{cl}$  αναπαριστά τους κόσμους σ.σ.μ. το  $\mathbf{D}'$  που είναι κοινοί στο  $c_1^{cl}$  και στο  $c_2^{cl}$ . Συνεπώς, αν το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται στα  $c_1^{cl}$  και  $c_2^{cl}$  και μόνον αυτές<sup>16</sup>, το αποτέλεσμα της τομής όρου μπορεί να ερμηνευτεί ορθά σ.σ.μ. κάθε  $\mathbf{D}$ , όπου  $\mathbf{D}_{\min} \subseteq \mathbf{D}$ .

### Αμοιβαία αποκλειόμενοι όροι

Μια ενδιαφέρουσα περίπτωση είναι όταν δυο όροι αναπαριστούν ξένα μεταξύ τους σύνολα κόσμων. Τέτοιοι όροι καλούνται *αμοιβαία αποκλειόμενοι*.

#### Ορισμός 3.12

Δυο όροι προσδιοριστή περιβάλλοντος  $c_1^{cl}$ ,  $c_2^{cl}$  είναι *αμοιβαία αποκλειόμενοι*, εάν και μόνον εάν  $W_D(c_1^{cl}) \cap W_D(c_2^{cl}) = \emptyset$ , όπου  $W_D(c^{cl})$  είναι το σύνολο κόσμων που αναπαριστά το  $c^{cl}$  σ.σ.μ. κάποιο σύνολο διαστάσεων  $\mathbf{D}$ .

Από την Πρόταση 3.4 βλέπουμε ότι δυο όροι είναι αμοιβαία αποκλειόμενοι αν και μόνο αν  $c_1^{cl} \cap^{cl} c_2^{cl} = \mathcal{E}^{cl}$ . Συνεπώς, για να είναι δυο όροι αμοιβαία αποκλειόμενοι, το αποτέλεσμα της τομής όρου των πρέπει να είναι ένας κενός όρος.

Παρότι ο αμοιβαίος αποκλεισμός ορίζεται με αναφορά σε ένα σύνολο διαστάσεων  $\mathbf{D}$ , στην πραγματικότητα δεν είναι ορισμένος σε σχέση με κάποιο  $\mathbf{D}$ . Αφότου η τομή όρου δεν εξαρτάται από το  $\mathbf{D}$ , εξυπακούεται ότι αν δυο όροι είναι αμοιβαία αποκλειόμενοι σ.σ.μ. κάποιο  $\mathbf{D}$ , είναι επίσης αμοιβαία αποκλειόμενοι σ.σ.μ. οποιοδήποτε άλλο  $\mathbf{D}'$ , όπου  $\mathbf{D}'$  και  $\mathbf{D}$  είναι υπερσύνολα του  $\mathbf{D}_{\min}$ .

Ο Τύπος 3.4 δείχνει ότι ένας καθολικός όρος  $\mathcal{U}^{cl}$  είναι αμοιβαία αποκλειόμενος μόνον με τον κενό όρο. Ο Τύπος 3.5 δείχνει ότι ένας κενός όρος  $\mathcal{E}^{cl}$  είναι αμοιβαία αποκλειόμενος με οποιονδήποτε άλλο όρο, συμπεριλαμβανομένων και των κενών όρων.

#### Παράδειγμα 3.5

Θεωρήστε τα ακόλουθα:

$$\begin{aligned} \mathbf{D} &= \{\text{lang}, \text{detail}, \text{format}\} \\ \mathbf{V}_{\text{lang}} &= \{\text{en}, \text{gr}, \text{sp}\} \\ \mathbf{V}_{\text{detail}} &= \{\text{low}, \text{high}\} \\ \mathbf{V}_{\text{format}} &= \{\text{ps}, \text{pdf}\} \end{aligned}$$

$$\begin{aligned} c_1^{cl} &= \{(\text{lang}, \{\text{sp}\}), (\text{format}, \{\text{pdf}\})\} \\ c_2^{cl} &= \{(\text{lang}, \{\text{en}, \text{gr}\}), (\text{detail}, \{\text{low}, \text{high}\})\} \\ c_3^{cl} &= \{(\text{detail}, \{\text{low}, \text{high}\})\} \end{aligned}$$

◆

Στο Παράδειγμα 3.5 οι όροι  $c_1^{cl}$  και  $c_2^{cl}$  είναι αμοιβαία αποκλειόμενοι. Αντίθετα, οι  $c_1^{cl}$  και  $c_3^{cl}$  δεν είναι αμοιβαία αποκλειόμενοι, μια και οι κόσμοι  $\{(\text{lang}, \text{sp}), (\text{format}, \text{pdf}), (\text{detail}, \text{low})\}$  και  $\{(\text{lang}, \text{sp}), (\text{format}, \text{pdf}), (\text{detail}, \text{high})\}$  καλύπτονται και από τον  $c_1^{cl}$  και από τον  $c_3^{cl}$ .

<sup>16</sup> Στην οριακή περίπτωση όπου  $c_1^{cl}$  και  $c_2^{cl}$  δεν περιέχουν καθόλου προσδιοριστές διάστασης, το  $\mathbf{D}_{\min}$  δεν μπορεί να οριστεί καθώς πρέπει να είναι μη κενό σύνολο. Στην περίπτωση αυτή, το αποτέλεσμα της τομής όρου ισχύει σ.σ.μ. κάθε δυνατό σύνολο διαστάσεων.



### 3.3.1.5 Οι όροι κατά την ένωση και την διαφορά

Οι όροι προσδιοριστή περιβάλλοντος είναι *κλειστοί* κάτω από την πράξη της τομής όρου, όπως φαίνεται εύκολα από τον ορισμό της. Όμως, δεν είναι δυνατόν να οριστούν πράξεις όρων που να αντιστοιχούν στην ένωση και διαφορά συνόλων και που να επιδεικνύουν την ιδιότητα αυτή. Επιπλέον, υπάρχουν σύνολα κόσμων που δεν μπορούν να εκφραστούν μέσω ενός και μόνου όρου προσδιοριστή περιβάλλοντος: για παράδειγμα, θεωρήστε το σύνολο κόσμων  $\{ \{ (lang, en), (detail, low) \}, \{ (lang, gr), (detail, high) \} \}$ . Αυτά τα ζητήματα οδηγούν στην εισαγωγή στην Ενότητα 3.3.2 των **προσδιοριστών περιβάλλοντος**, που στην πράξη είναι διαζεύξεις όρων προσδιοριστών περιβάλλοντος. Στις Ενότητες 3.3.2.6 και 3.3.2.7, που εισάγονται η ένωση και η διαφορά για προσδιοριστές περιβάλλοντος, ορίζουμε επίσης την ένωση και την διαφορά για όρους. Στις ενότητες αυτές θα διαφανεί ότι, στην γενική περίπτωση, η ένωση όρου και η διαφορά όρου δίνουν προσδιοριστές περιβάλλοντος, και όχι όρους προσδιοριστή περιβάλλοντος.

### 3.3.1.6 Υποσύνολο όρου

Το **υποσύνολο όρου**  $\subseteq^{cl}$ , και το **γνήσιο υποσύνολο όρου**  $\subset^{cl}$  είναι πράξεις που δέχονται δυο όρους προσδιοριστή περιβάλλοντος  $c^{cl_1}$ ,  $c^{cl_2}$  και επιστρέφουν μια τιμή αληθείας (Boolean).

$\subseteq^{cl} : (C^{cl} \times C^{cl}) \rightarrow \text{Boolean}$ , όπου  $C^{cl}$  το σύνολο όλων των όρων  
 $\subset^{cl} : (C^{cl} \times C^{cl}) \rightarrow \text{Boolean}$ , όπου  $C^{cl}$  το σύνολο όλων των όρων

Το υποσύνολο / υπερσύνολο όρου και το γνήσιο υποσύνολο / υπερσύνολο όρου ορίζονται ως εξής:

#### Ορισμός 3.13

Ας είναι το  $\mathbf{D}$  ένα μη κενό σύνολο ονομάτων διαστάσεων, και για κάθε  $d \in \mathbf{D}$  ας είναι το  $\mathbf{V}_d$  το πεδίο ορισμού του  $d$ , με  $\mathbf{V}_d \neq \emptyset$ . Ας είναι τα  $c^{cl_1}$ ,  $c^{cl_2}$  όροι προσδιοριστή περιβάλλοντος. Τότε το  $c^{cl_1}$  είναι **υποσύνολο όρου** του  $c^{cl_2}$ , και συμβολίζεται  $c^{cl_1} \subseteq^{cl} c^{cl_2}$ , ή ισοδύναμα, το  $c^{cl_2}$  είναι **υπερσύνολο όρου** του  $c^{cl_1}$ , και συμβολίζεται  $c^{cl_2} \supseteq^{cl} c^{cl_1}$ , εάν και μόνον εάν  $W_D(c^{cl_1}) \subseteq W_D(c^{cl_2})$ , όπου  $W_D(c^{cl})$  είναι το σύνολο των κόσμων σ.σ.μ. το  $\mathbf{D}$  που αναπαριστά ένας όρος  $c^{cl}$ .

Στην περίπτωση όπου  $W_D(c^{cl_1}) \subset W_D(c^{cl_2})$ , τότε το  $c^{cl_1}$  είναι **γνήσιο υποσύνολο όρου** του  $c^{cl_2}$ , και συμβολίζεται  $c^{cl_1} \subset^{cl} c^{cl_2}$ , ή ισοδύναμα, το  $c^{cl_2}$  είναι **γνήσιο υπερσύνολο όρου** του  $c^{cl_1}$ , και συμβολίζεται  $c^{cl_2} \supset^{cl} c^{cl_1}$ .

Η παρακάτω πρόταση δείχνει ότι το υποσύνολο όρου μπορεί να εκφραστεί μέσω της τομής όρου και της ισότητας όρου, ακριβώς όπως συμβαίνει με τις συμβατικές πράξεις συνόλων.

#### Πρόταση 3.5

Για δυο όρους προσδιοριστή περιβάλλοντος  $c^{cl_1}$ ,  $c^{cl_2}$  ισχύουν τα ακόλουθα: (α)  $c^{cl_1} \subseteq^{cl} c^{cl_2} \Leftrightarrow c^{cl_1} \cap^{cl} c^{cl_2} =^{cl} c^{cl_1}$ , και (β)  $c^{cl_1} \subset^{cl} c^{cl_2} \Leftrightarrow (c^{cl_1} \subseteq^{cl} c^{cl_2} \text{ ΚΑΙ } c^{cl_1} \neq^{cl} c^{cl_2})$ .

Παρότι οι πράξεις του (γνήσιου) υποσυνόλου / υπερσυνόλου όρου ορίστηκαν με αναφορά σε κάποιο σύνολο διαστάσεων  $\mathbf{D}$ , στην πραγματικότητα δεν ορίστηκαν σε σχέση με κάποιο

$\mathbf{D}^{17}$ . Από την Πρόταση 3.5, και επειδή η τομή όρου και η ισότητα όρου δεν εξαρτώνται από το  $\mathbf{D}$ , βλέπει κανείς εύκολα ότι, αν το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται στα  $c_1^{cl}$  και  $c_2^{cl}$  και καμία άλλη<sup>18</sup>, τότε  $c_1^{cl} \subseteq_{\mathbf{D}}^{cl} c_2^{cl} \Leftrightarrow c_1^{cl} \subseteq_{\mathbf{D}'}^{cl} c_2^{cl}$  και  $c_1^{cl} \subset_{\mathbf{D}}^{cl} c_2^{cl} \Leftrightarrow c_1^{cl} \subset_{\mathbf{D}'}^{cl} c_2^{cl}$ ,  $\forall \mathbf{D}$  και  $\mathbf{D}'$  με  $\mathbf{D}_{\min} \subseteq \mathbf{D}$  και  $\mathbf{D}_{\min} \subseteq \mathbf{D}'$ .

Αναφορικά με τον κενό όρο και τον καθολικό όρο, είναι προφανές ότι  $\mathcal{E}^{cl} \subseteq^{cl} c^{cl}$  και  $c^{cl} \subseteq^{cl} \mathcal{U}^{cl}$ , για κάθε όρο προσδιοριστή περιβάλλοντος  $c^{cl}$ .

### Παράδειγμα 3.6

Θεωρήστε ότι:

$\mathbf{D} = \{\text{lang, detail, format}\}$

$\mathbf{V}_{\text{lang}} = \{\text{en, gr, sp}\}$

$\mathbf{V}_{\text{detail}} = \{\text{low, high}\}$

$\mathbf{V}_{\text{format}} = \{\text{ps, pdf}\}$

$c_1^{cl} = \{(\text{detail}, \{\text{low, high}\})\}$

$c_2^{cl} = \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{low, high}\}), (\text{format}, \{\text{ps}\})\}$

$c_3^{cl} = \{(\text{lang}, \{\text{gr}\}), (\text{detail}, \{\text{high}\}), (\text{format}, \{\text{ps}\})\}$

$c_4^{cl} = \{(\text{lang}, \{\text{gr, sp}\}), (\text{detail}, \{\text{low}\})\}$

Τότε, ισχύουν τα ακόλουθα:

$c_3^{cl} \subset^{cl} c_2^{cl} \subset^{cl} c_1^{cl}$

$c_4^{cl} \subset^{cl} c_1^{cl}$

◆

## 3.3.2 Προσδιοριστές περιβάλλοντος

Στην ενότητα αυτή εισάγουμε τους **προσδιοριστές περιβάλλοντος** (περιβάλλοντα για συντομία), και ορίζουμε πράξεις πάνω στους προσδιοριστές περιβάλλοντος (**πράξεις περιβάλλοντος**) που βασίζονται στις ήδη ορισμένες πράξεις για τους όρους προσδιοριστή περιβάλλοντος (**πράξεις όρου**). Όπως είπαμε στην Ενότητα 3.3.1.5, ένας όρος προσδιοριστή περιβάλλοντος δεν μπορεί να εκφράσει κάθε δυνατό σύνολο κόσμων. Οι προσδιοριστές περιβάλλοντος είναι δομές ικανές να αναπαραστήσουν οποιοδήποτε σύνολο κόσμων.

### Ορισμός 3.14

*Ένας προσδιοριστής περιβάλλοντος  $c$  είναι ένα μη κενό σύνολο από όρους προσδιοριστή περιβάλλοντος,  $c = \{c_1^{cl}, c_2^{cl}, \dots, c_n^{cl}\}$  με  $n \geq 1$ .*

Όπως έχει ήδη λεχθεί, το σύμβολο  $^{cl}$ , που σημαίνει *όρος* (clause), χρησιμοποιείται για την ταυτοποίηση μεταβλητών όρων και πράξεων όρων. Δεν θα χρησιμοποιήσουμε κάποιο ειδικό σύμβολο για τις μεταβλητές προσδιοριστών περιβάλλοντος. Θα χρησιμοποιήσουμε το σύμβολο  $^c$ , που σημαίνει *περιβάλλον* (context), προκειμένου να διαχωρίζουμε, όποτε είναι αναγκαίο, τις πράξεις περιβάλλοντος από την μια, και τις πράξεις όρου και τις συμβατικές πράξεις συνόλων από την άλλη.

<sup>17</sup> Πάντως, ο έλεγχος δυο όρων για (γνήσιο) υποσύνολο / υπερσύνολο όρου ίσως χρειαστεί να πάρει υπόψη τα πεδία ορισμού των διαστάσεων που απαντώνται στους όρους αυτούς.

<sup>18</sup> Στην οριακή περίπτωση όπου τα  $c_1^{cl}$  και  $c_2^{cl}$  δεν περιέχουν καθόλου προσδιοριστές διάστασης, το  $\mathbf{D}_{\min}$  δεν μπορεί να οριστεί αφού πρέπει να είναι μη κενό σύνολο. Στην περίπτωση αυτή, το αποτέλεσμα του (γνήσιου) υποσυνόλου / υπερσυνόλου όρου ισχύει σ.σ.μ. κάθε δυνατό σύνολο διαστάσεων.

### 3.3.2.1 Αντιστοιχία με κόσμους

Στην ουσία, ένας προσδιοριστής περιβάλλοντος είναι μια διάζευξη όρων προσδιοριστή περιβάλλοντος. Η σχέση των προσδιοριστών περιβάλλοντος με τους πιθανούς κόσμους εξηγείται στον ακόλουθο ορισμό.

#### Ορισμός 3.15

Το σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$  που αναπαριστά ένας προσδιοριστής περιβάλλοντος δίνεται από την ένωση των συνόλων κόσμων σ.σ.μ. το  $\mathbf{D}$  που αναπαριστούν οι όροι ο οποίοι περιέχονται σε αυτόν τον προσδιοριστή περιβάλλοντος.

$$W_D(c) = W_D(c^{c_1}) \cup W_D(c^{c_2}) \cup \dots \cup W_D(c^{c_n}),$$

$$\text{όπου } c = \{c^{c_1}, c^{c_2}, \dots, c^{c_n}\}, \quad n \geq 1$$

#### Τύπος 3.6

Η παρακάτω πρόταση δείχνει ότι κάθε δυνατό σύνολο κόσμων μπορεί να εκφραστεί σαν ένας προσδιοριστής περιβάλλοντος.

#### Πρόταση 3.6

Για κάθε σύνολο κόσμων  $S = \{w_1, w_2, \dots, w_n\}$  σ.σ.μ. το  $\mathbf{D}$  υπάρχει τουλάχιστον ένας προσδιοριστής περιβάλλοντος  $c$ , τέτοιος ώστε  $W_D(c) = S$ .

Συνεπώς, μέσα στα πλαίσια ενός συνόλου διαστάσεων  $\mathbf{D}$ , οι όροι περιβάλλον, προσδιοριστής περιβάλλοντος, και σύνολο κόσμων μπορούν να χρησιμοποιούνται χωρίς διάκριση.

Η Πρόταση 3.6 (και η απόδειξή της) δείχνουν ότι αν το  $\mathbf{D}$  είναι πεπερασμένο και το  $\mathbf{V}_d$  είναι επίσης πεπερασμένο για κάθε  $d$  στο  $\mathbf{D}$ , τότε υπάρχει ένας πεπερασμένος προσδιοριστής περιβάλλοντος που αναπαριστά κάθε δυνατό σύνολο κόσμων. Σημειώστε πάντως, ότι ένα σύνολο κόσμων μπορεί να αναπαρασταθεί από έναν αριθμό διαφορετικών μεταξύ τους προσδιοριστών περιβάλλοντος. Αφού οι όροι σε έναν προσδιοριστή περιβάλλοντος μπορούν να αντιστοιχούν σε επικαλυπτόμενα σύνολα κόσμων, ένας προσδιοριστής περιβάλλοντος μπορεί να περιέχει άπειρο αριθμό όρων ενώ ταυτόχρονα να αναπαριστά πεπερασμένο αριθμό κόσμων.

Το σύνολο των κόσμων που αναπαριστά ένας προσδιοριστής περιβάλλοντος μπορεί να το δώσει ο Τύπος 3.6, χρησιμοποιώντας τις πράξεις της επέκτασης όρου και ανάπτυξης όρου που ορίστηκαν στην Ενότητα 3.3.1.1. Εναλλακτικά, είναι δυνατόν να χρησιμοποιηθούν απευθείας οι πράξεις της επέκτασης περιβάλλοντος και της ανάπτυξης περιβάλλοντος, που ορίζονται παρακάτω.

#### Επεκτεταμένη μορφή περιβάλλοντος

Η **επέκταση περιβάλλοντος**  $\odot_D$  είναι μια πράξη που δέχεται έναν προσδιοριστή περιβάλλοντος  $c_1$  και επιστρέφει έναν προσδιοριστή περιβάλλοντος  $c_2$ . Η επέκταση περιβάλλοντος εκτελείται σε σχέση με ένα σύνολο διαστάσεων  $\mathbf{D}$  και με τα σύνολα των πεδίων ορισμού των διαστάσεων  $\mathbf{V}_d$  για κάθε  $d$  στο  $\mathbf{D}$ .

$$\odot_D : C \rightarrow C, \text{ όπου } C \text{ το σύνολο όλων των προσδιοριστών περιβάλλοντος}$$

Το περιβάλλον του αποτελέσματος  $c_2$  καλείται **επεκτεταμένη μορφή** του περιβάλλοντος  $c_1$  σε σχέση με το  $\mathbf{D}$ , και ορίζεται ως εξής:

**Ορισμός 3.16**

Ας είναι το  $\mathbf{D}$  ένα σύνολο από ονόματα διαστάσεων, και το  $c = \{c^1_1, c^1_2, \dots, c^1_n\}$ ,  $n \geq 1$ , ένας προσδιοριστής περιβάλλοντος. Τότε η **επέκταση περιβάλλοντος** του  $c$  σε σχέση με το  $\mathbf{D}$  (λέγεται επίσης και **επεκτεταμένη μορφή** του  $c$  σ.σ.μ. το  $\mathbf{D}$ ), συμβολίζεται  $\Theta_{\mathbf{D}} c$ , και είναι ο προσδιοριστής περιβάλλοντος  $\{\Theta^1_{\mathbf{D}} c^1_1, \Theta^1_{\mathbf{D}} c^1_2, \dots, \Theta^1_{\mathbf{D}} c^1_n\}$ .

Οι όροι της επεκτεταμένης μορφής ενός προσδιοριστή περιβάλλοντος περιέχουν προσδιοριστές διάστασης που αντιστοιχούν επακριβώς στις διαστάσεις του  $\mathbf{D}$ . Η επεκτεταμένη μορφή ενός προσδιοριστή περιβάλλοντος είναι ισοδύναμη με τον συγκεκριμένο προσδιοριστή περιβάλλοντος σε σχέση με το  $\mathbf{D}$ , με την έννοια ότι και οι δυο αναπαριστούν το ίδιο σύνολο κόσμων σε σχέση με το  $\mathbf{D}$ :

**Πρόταση 3.7**

Ένας προσδιοριστής περιβάλλοντος και η επεκτεταμένη μορφή του σε σχέση με το  $\mathbf{D}$  αναπαριστούν το ίδιο σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$ , για κάθε σύνολο διαστάσεων  $\mathbf{D}$ .

Παρατηρήστε ότι η επεκτεταμένη μορφή ενός περιβάλλοντος δεν μπορεί ποτέ να περιέχει το κενό σύνολο, μια και η επέκταση όρου δεν επιστρέφει ποτέ το κενό σύνολο.

**Ανάπτυξη περιβάλλοντος**

Η **ανάπτυξη περιβάλλοντος**  $\times$  είναι μια πράξη που λαμβάνει έναν προσδιοριστή περιβάλλοντος  $c$ , με  $\emptyset \notin c$ , και επιστρέφει ένα σύνολο συνόλων από ζεύγη (διάσταση, τιμή).

$\times : (C - C_{\emptyset}) \rightarrow S$ , όπου  $C$  το σύνολο των προσδιοριστών περιβάλλοντος, και  $C_{\emptyset}$  το σύνολο περιβαλλόντων  $c_{\emptyset}$  με  $\emptyset \in c_{\emptyset}$

Η ανάπτυξη περιβάλλοντος ορίζεται ως εξής.

**Ορισμός 3.17**

Η **ανάπτυξη περιβάλλοντος** ενός προσδιοριστή περιβάλλοντος  $c = \{c^1_1, c^1_2, \dots, c^1_n\}$ ,  $n \geq 1$ , συμβολίζεται  $\times c$ , και είναι το σύνολο  $\{\times^1 c^1_1\} \cup \{\times^1 c^1_2\} \cup \dots \cup \{\times^1 c^1_n\}$ .

Κάθε σύνολο στο αποτέλεσμα μιας ανάπτυξης όρου περιέχει ζεύγη (διάσταση, τιμή) για τις ίδιες ακριβώς διαστάσεις. Σε αντίθεση με την ανάπτυξη όρου, το αποτέλεσμα της ανάπτυξης περιβάλλοντος μπορεί να περιέχει σύνολα με διαφορετικό αριθμό στοιχείων που αποτελούνται από ζεύγη τα οποία αντιστοιχούν σε διαφορετικές διαστάσεις. Πάντως, η ανάπτυξη περιβάλλοντος της επεκτεταμένης μορφής σ.σ.μ. το  $\mathbf{D}$  ενός προσδιοριστή περιβάλλοντος θα δώσει ένα σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$ , ακριβώς όπως συμβαίνει με τους όρους.

**Πρόταση 3.8**

Οι κόσμοι σ.σ.μ. το  $\mathbf{D}$  που αναπαριστά ένας προσδιοριστής περιβάλλοντος  $c$  δίνονται από την ανάπτυξη περιβάλλοντος της επεκτεταμένης μορφής σ.σ.μ. το  $\mathbf{D}$  του  $c$ .

$$W_{\mathbf{D}}(c) = \times (\Theta_{\mathbf{D}} c)$$

**Τύπος 3.7**

### Ανάπτυξη επεκτεταμένου περιβάλλοντος

Η επέκταση περιβάλλοντος και η ανάπτυξη περιβάλλοντος συχνά χρησιμοποιούνται μαζί για να δώσουν το σύνολο των κόσμων σ.σ.μ. το  $\mathbf{D}$  που αναπαριστά ένας προσδιοριστής περιβάλλοντος. Παρακάτω εισάγουμε μια πράξη που συνδυάζει την επέκταση περιβάλλοντος και την ανάπτυξη περιβάλλοντος, σαν συντομογραφία.

Η **ανάπτυξη επεκτεταμένου περιβάλλοντος**  $\otimes_{\mathbf{D}}$  είναι μια πράξη που λαμβάνει έναν προσδιοριστή περιβάλλοντος  $c$  και επιστρέφει ένα σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$ . Η ανάπτυξη επεκτεταμένου περιβάλλοντος εκτελείται σε σχέση με ένα σύνολο διαστάσεων  $\mathbf{D}$  και με τα σύνολα των πεδίων ορισμού των διαστάσεων  $\mathbf{V}_d$  για κάθε  $d$  στο  $\mathbf{D}$ .

$\otimes_{\mathbf{D}} : \mathcal{C} \rightarrow S_W$ , όπου  $\mathcal{C}$  το σύνολο των προσδιοριστών περιβάλλοντος, και  $S_W$  περιέχει σύνολα κόσμων σ.σ.μ. το  $\mathbf{D}$

Η ανάπτυξη επεκτεταμένου περιβάλλοντος ορίζεται ακολούθως.

#### Ορισμός 3.18

*Η ανάπτυξη επεκτεταμένου περιβάλλοντος σε σχέση με το  $\mathbf{D}$  ενός προσδιοριστή περιβάλλοντος  $c$ , συμβολίζεται  $\otimes_{\mathbf{D}} c$ , και ορίζεται ως εξής:  $\otimes_{\mathbf{D}} c = \times (\otimes_{\mathbf{D}} c)$ .*

Με βάση τα παραπάνω, ο Τύπος 3.7 μπορεί επίσης να πάρει την μορφή:

$$W_{\mathbf{D}}(c) = \otimes_{\mathbf{D}} c$$

### 3.3.2.2 Ισότητα περιβάλλοντος

Η **ισότητα περιβάλλοντος**  $=^c$  είναι μια πράξη που λαμβάνει δυο προσδιοριστές περιβάλλοντος  $c_1, c_2$  και επιστρέφει μια τιμή αληθείας (Boolean).

$=^c : (\mathcal{C} \times \mathcal{C}) \rightarrow \text{Boolean}$ , όπου  $\mathcal{C}$  το σύνολο όλων των περιβαλλόντων

Η ισότητα περιβάλλοντος και η ανισότητα περιβάλλοντος ορίζονται ακολούθως:

#### Ορισμός 3.19

*Ας είναι το  $\mathbf{D}$  ένα μη κενό σύνολο από ονόματα διαστάσεων, και τα  $c_1, c_2$  προσδιοριστές περιβάλλοντος. Τότε τα  $c_1$  και  $c_2$  είναι **ίσα-περιβάλλοντα**, και συμβολίζεται  $c_1 =^c c_2$ , εάν και μόνον  $W_{\mathbf{D}}(c_1) = W_{\mathbf{D}}(c_2)$ , όπου  $W_{\mathbf{D}}(c)$  είναι το σύνολο των κόσμων σ.σ.μ. το  $\mathbf{D}$  που αναπαριστά ένας προσδιοριστής περιβάλλοντος  $c$ . Εάν και μόνον  $W_{\mathbf{D}}(c_1) \neq W_{\mathbf{D}}(c_2)$ , τότε τα  $c_1$  και  $c_2$  είναι **άνισα-περιβάλλοντα**, και συμβολίζεται  $c_1 \neq^c c_2$ .*

Παρόλο που η ισότητα περιβάλλοντος ορίζεται με αναφορά σε ένα σύνολο διαστάσεων  $\mathbf{D}$ , στην πραγματικότητα δεν ορίζεται σε σχέση με κάποιο  $\mathbf{D}$ <sup>19</sup>. Η ακόλουθη Πρόταση 3.9 δείχνει ότι αν δυο προσδιοριστές περιβάλλοντος είναι ίσα-περιβάλλοντα σ.σ.μ. ένα σύνολο διαστάσεων  $\mathbf{D}$ , είναι επίσης ίσα-περιβάλλοντα σ.σ.μ. κάθε άλλο  $\mathbf{D}'$ . Το σύμβολο  $=^c_{\mathbf{D}}$  χρησιμοποιείται στην Πρόταση 3.9 για να εκφράσει ισότητα περιβάλλοντος σ.σ.μ. το  $\mathbf{D}$ .

<sup>19</sup> Πάντως, ο έλεγχος δυο προσδιοριστών περιβάλλοντος για ισότητα περιβάλλοντος ίσως χρειαστεί να πάρει υπόψη τα πεδία ορισμού των διαστάσεων που απαντώνται σε αυτούς τους προσδιοριστές περιβάλλοντος.

**Πρόταση 3.9**

Ας είναι τα  $c_1$  και  $c_2$  προσδιοριστές περιβάλλοντος και τα  $\mathbf{D}$ ,  $\mathbf{D}'$  σύνολα διαστάσεων. Ισχύουν τα ακόλουθα: (a) αν  $c_1 \stackrel{c}{=}_{\mathbf{D}} c_2$  (σε σχέση με το  $\mathbf{D}$ ), τότε  $c_1 \stackrel{c}{=}_{\mathbf{D}'} c_2$  (σε σχέση με το  $\mathbf{D}'$ ), και (b) αν  $c_1 \neq^c_{\mathbf{D}} c_2$  (σε σχέση με το  $\mathbf{D}$ ), τότε  $c_1 \neq^c_{\mathbf{D}'} c_2$  (σε σχέση με το  $\mathbf{D}'$ ).

Η Πρόταση 3.9 βασίζεται στην παραδοχή, που αναφέρεται στην Ενότητα 3.2.1, ότι θεωρούμε ένα σύνολο διαστάσεων που περιέχει πάντα (τουλάχιστον) τις διαστάσεις εκείνες που απαντώνται στα εμπλεκόμενα περιβάλλοντα. Έτσι, αν το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται στα  $c_1$  και  $c_2$  και καμία άλλη,  $c_1 \stackrel{c}{=}_{\mathbf{D}} c_2 \Leftrightarrow c_1 \stackrel{c}{=}_{\mathbf{D}'} c_2$  και  $c_1 \neq^c_{\mathbf{D}} c_2 \Leftrightarrow c_1 \neq^c_{\mathbf{D}'} c_2$ ,  $\forall \mathbf{D}$  και  $\mathbf{D}'$  με  $\mathbf{D}_{\min} \subseteq \mathbf{D}$  και  $\mathbf{D}_{\min} \subseteq \mathbf{D}'$ .

Η συμβατική ισότητα συνόλων δεν είναι επαρκής για την απευθείας σύγκριση προσδιοριστών περιβάλλοντος, επειδή είναι δυνατόν δυο προσδιοριστές περιβάλλοντος με διαφορετική μορφή να αναπαριστούν το ίδιο σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$ . Δυο προσδιοριστές περιβάλλοντος είναι *ίσα-περιβάλλοντα* αν αναπαριστούν το ίδιο σύνολο κόσμων σ.σ.μ. κάποιο  $\mathbf{D}$ , ανεξάρτητα από την μορφή που έχουν.

**Παράδειγμα 3.7**

Θεωρήστε ότι:

$$\mathbf{D} = \{\text{lang, detail, format}\}$$

$$\mathbf{V}_{\text{lang}} = \{\text{en, gr, sp}\}$$

$$\mathbf{V}_{\text{detail}} = \{\text{low, high}\}$$

$$\mathbf{V}_{\text{format}} = \{\text{ps, pdf}\}$$

$$c_1 = \{ \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{high}\})\}, \{(\text{lang}, \{\text{gr}\}), (\text{detail}, \{\text{low}\})\} \}$$

$$c_2 = \{ \{(\text{lang}, \{\text{en}\}), (\text{detail}, \{\text{high}\})\}, \{(\text{lang}, \{\text{gr}\}), (\text{detail}, \{\text{low, high}\})\} \}$$

$$c_3 = \{ \{(\text{detail}, \{\text{low, high}\}), (\text{format}, \{\text{ps, pdf}\})\} \}$$

$$c_4 = \{ \{(\text{lang}, \{\text{gr, sp}\}), (\text{detail}, \{\text{low, high}\})\}, \{(\text{lang}, \{\text{en}\}), (\text{detail}, \{\text{low, high}\})\} \}$$

Τότε, ισχύουν τα ακόλουθα:

$$c_1 \stackrel{c}{=} c_2$$

$$c_2 \neq^c c_3$$

$$c_3 \stackrel{c}{=} c_4$$

◆

Στο Παράδειγμα 3.7, τα περιβάλλοντα  $c_1$  και  $c_2$  έχουν διαφορετική μορφή αλλά αναπαριστούν το ίδιο σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$ . Το περιβάλλον  $c_3$  περιέχει έναν καθολικό όρο  $\mathcal{U}^1$  που αναπαριστά κάθε πιθανό κόσμο σ.σ.μ. το  $\mathbf{D}$ , και δεν είναι ίσο-περιβάλλον με το  $c_1$  και το  $c_2$ . Ίδια είναι η περίπτωση του  $c_4$ , που περιέχει δυο όρους που μαζί καλύπτουν εντελώς το  $U_{\mathbf{D}}$ .

**3.3.2.3 Κενό περιβάλλον και καθολικό περιβάλλον**

Όπως αντίστοιχα και με τους όρους, το **κενό περιβάλλον**  $\mathcal{E}$  αποτελεί ένα κοινό σύμβολο για εκείνα τα περιβάλλοντα που δεν αναπαριστούν κανέναν κόσμο σ.σ.μ. κάποιο σύνολο διαστάσεων  $\mathbf{D}$ . Το **καθολικό περιβάλλον**  $\mathcal{U}$  αποτελεί ένα κοινό σύμβολο για εκείνα τα περιβάλλοντα που αναπαριστούν το σύμπαν  $U_{\mathbf{D}}$  κάποιου  $\mathbf{D}$ .

**Ορισμός 3.20**

Ας είναι το  $\mathbf{D}$  ένα μη κενό σύνολο ονομάτων διαστάσεων, και για κάθε  $d \in \mathbf{D}$  ας είναι το  $\mathbf{V}_d$  το πεδίο ορισμού του  $d$ , με  $\mathbf{V}_d \neq \emptyset$ . Ας είναι το  $c$  ένας προσδιοριστής περιβάλλοντος, και το  $W_D(c)$  το σύνολο των κόσμων που αυτός αναπαριστά σε σχέση με το  $\mathbf{D}$ . Εάν και μόνον  $W_D(c) = \emptyset$ , τότε το  $c$  καλείται **κενό περιβάλλον**, και συμβολίζεται επίσης  $\mathcal{E}$ . Εάν και μόνον  $W_D(c) = U_D$ , όπου  $U_D$  είναι το σύμπαν του  $\mathbf{D}$ , τότε το  $c$  καλείται **καθολικό περιβάλλον**, και συμβολίζεται επίσης  $\mathcal{U}$ .

Παρόλο που το κενό περιβάλλον και το καθολικό περιβάλλον ορίζονται με αναφορά σε ένα σύνολο διαστάσεων  $\mathbf{D}$ , στην πραγματικότητα δεν ορίζονται σε σχέση με κάποιο  $\mathbf{D}$ <sup>20</sup>. Στην Ενότητα 3.2.1, αναφέραμε την προϋπόθεση ότι ένα σύνολο διαστάσεων  $\mathbf{D}$  πρέπει να περιέχει (τουλάχιστον) τις διαστάσεις που απαντώνται στα εμπλεκόμενα περιβάλλοντα. Ας περιέχει το  $\mathbf{D}_{\min}$  όλες τις διαστάσεις που απαντώνται σε έναν προσδιοριστή περιβάλλοντος  $c$  και καμία άλλη. Βλέπουμε εύκολα ότι αν το  $c$  είναι κενό περιβάλλον σ.σ.μ. το  $\mathbf{D}$ , τότε το  $c$  είναι επίσης κενό περιβάλλον σ.σ.μ. κάθε άλλο  $\mathbf{D}'$ , όπου  $\mathbf{D}_{\min} \subseteq \mathbf{D}$  και  $\mathbf{D}_{\min} \subseteq \mathbf{D}'$ . Ομοίως, αν το  $c$  είναι καθολικό περιβάλλον σ.σ.μ. το  $\mathbf{D}$ , τότε το  $c$  είναι επίσης καθολικό περιβάλλον σ.σ.μ. κάθε άλλο  $\mathbf{D}'$ , όπου  $\mathbf{D}_{\min} \subseteq \mathbf{D}$  και  $\mathbf{D}_{\min} \subseteq \mathbf{D}'$ .

Ο Τύπος 3.7 και ο Ορισμός 3.20 δίνουν:

$$\begin{aligned} W_D(\mathcal{E}) &= \times (\odot_D \mathcal{E}) = \emptyset \\ W_D(\mathcal{U}) &= \times (\odot_D \mathcal{U}) = U_D \end{aligned}$$

**Τύπος 3.8**

**Τύπος 3.9**

Ο Ορισμός 3.18 δείχνει ότι, ο Τύπος 3.8 και ο Τύπος 3.9 μπορούν επίσης να πάρουν την μορφή:

$$\begin{aligned} W_D(\mathcal{E}) &= \otimes_D \mathcal{E} = \emptyset \\ W_D(\mathcal{U}) &= \otimes_D \mathcal{U} = U_D \end{aligned}$$

Ο Τύπος 3.6 μας δείχνει ότι ένα κενό περιβάλλον είναι δυνατόν να περιέχει αποκλειστικά κενούς όρους. Έτσι το  $\mathcal{E}$  συμβολίζει κάθε προσδιοριστή περιβάλλοντος της μορφής  $\{\mathcal{E}^{cl}, \dots, \mathcal{E}^{cl}, \dots\}$ . Όλα αυτά τα περιβάλλοντα είναι μεταξύ τους ίσα-περιβάλλοντα, μια και αναπαριστούν το κενό σύνολο κόσμων σ.σ.μ. κάποιο  $\mathbf{D}$ .

Ομοίως, κάθε προσδιοριστής περιβάλλοντος της μορφής  $\{c^{cl}_1, \dots, \mathcal{U}^{cl}, \dots, c^{cl}_m\}$  που περιέχει τουλάχιστον έναν καθολικό όρο, αντιπροσωπεύεται από το καθολικό περιβάλλον  $\mathcal{U}$ . Εκτός από προσδιοριστές περιβάλλοντος που περιέχουν έναν καθολικό όρο, το καθολικό περιβάλλον  $\mathcal{U}$  καλύπτει άλλο ένα είδος προσδιοριστών περιβάλλοντος, όπως φαίνεται στην περίπτωση του  $c_3$  στο Παράδειγμα 3.8.

**Παράδειγμα 3.8**

Θεωρήστε ότι:

$$\begin{aligned} \mathbf{D} &= \{\text{lang}, \text{detail}\} \\ \mathbf{V}_{\text{lang}} &= \{\text{en}, \text{gr}, \text{sp}\} \\ \mathbf{V}_{\text{detail}} &= \{\text{low}, \text{high}\} \end{aligned}$$

$$\begin{aligned} c_1 &= \{ \{(\text{detail}, \{\text{low}, \text{high}\})\} \} \\ c_2 &= \{ \{(\text{detail}, \{\text{low}\}), (\text{lang}, \{\text{en}, \text{sp}\})\}, \{ \} \} \end{aligned}$$

<sup>20</sup> Πάντως, ο έλεγχος για το εάν ένας προσδιοριστής περιβάλλοντος είναι κενό περιβάλλον ή καθολικό περιβάλλον πρέπει να λάβει υπόψη τα πεδία ορισμού των διαστάσεων που απαντώνται σε αυτόν τον προσδιοριστή περιβάλλοντος.

$$c_3 = \{ \{ (\text{lang}, \{ \text{gr}, \text{sp} \}), (\text{detail}, \{ \text{low}, \text{high} \}) \}, \\ \{ (\text{lang}, \{ \text{en} \}), (\text{detail}, \{ \text{low}, \text{high} \}) \} \}$$

◆

Στο Παράδειγμα 3.8, οι προσδιοριστές περιβάλλοντος  $c_1$ ,  $c_2$ , και  $c_3$  είναι καθολικά περιβάλλοντα. Καθένα από τα περιβάλλοντα  $c_1$  και  $c_2$  περιέχει έναν καθολικό όρο. Το περιβάλλον  $c_3$  από την άλλη, δεν περιέχει καθολικούς όρους. Όμως, οι δυο όροι του  $c_3$  αν παρθούν μαζί αναπαριστούν όλους τους πιθανούς κόσμους σ.σ.μ. το **D**. Όλα τα περιβάλλοντα που αναπαριστά το  $\mathcal{U}$  είναι μεταξύ τους ίσα-περιβάλλοντα, αφότου αντιπροσωπεύουν το σύμπαν  $U_D$  ενός συνόλου διαστάσεων **D**.

Από τα παραπάνω, είναι προφανές ότι το  $\mathcal{E}$  στην πραγματικότητα συμβολίζει κάθε μέλος ενός συνόλου με προσδιοριστές περιβάλλοντος που είναι ίσα-περιβάλλοντα, και που αναπαριστούν το κενό σύνολο κόσμων σ.σ.μ. κάποιο **D**. Όμοια, το  $\mathcal{U}$  συμβολίζει κάθε μέλος ενός συνόλου με προσδιοριστές περιβάλλοντος που είναι ίσα-περιβάλλοντα, και που αντιπροσωπεύουν το σύμπαν κάποιου **D**.

Σημειώστε ότι η επεκτεταμένη μορφή σ.σ.μ. το **D** ενός κενού περιβάλλοντος  $\mathcal{E}$  είναι και πάλι ένα κενό περιβάλλον, ενώ η επεκτεταμένη μορφή σ.σ.μ. το **D** ενός καθολικού περιβάλλοντος  $\mathcal{U}$  είναι και πάλι ένα καθολικό περιβάλλον.

Ένα ενδιαφέρον σημείο είναι ότι το περιβάλλον  $\{\emptyset\}$  είναι καθολικό περιβάλλον σε σχέση με κάθε δυνατό σύνολο διαστάσεων, και το περιβάλλον  $\{\{-\}\}$  είναι κενό περιβάλλον σε σχέση με κάθε δυνατό σύνολο διαστάσεων.

Στα ακόλουθα, το σύμβολο της ισότητας σε εκφράσεις της μορφής  $c = \mathcal{E}$  υπονοεί ότι το  $c$  είναι οποιοδήποτε από τα ίσα-περιβάλλοντα που αναπαριστά το  $\mathcal{E}$ . Η παραπάνω ισότητα μπορεί επίσης να εκφραστεί χρησιμοποιώντας την ισότητα περιβάλλοντος  $c =^c \{\{-\}\}$ . Ομοίως, το σύμβολο της ισότητας σε εκφράσεις της μορφής  $c = \mathcal{U}$  υπονοεί ότι το  $c$  είναι οποιοδήποτε από τα ίσα-περιβάλλοντα που αναπαριστά το  $\mathcal{U}$ . Αυτή η ισότητα μπορεί επίσης να εκφραστεί χρησιμοποιώντας την ισότητα περιβάλλοντος  $c =^c \{\emptyset\}$ .

### 3.3.2.4 Απλοποίηση προσδιοριστών περιβάλλοντος

Όπως δείξαμε στην Πρόταση 3.6, οι προσδιοριστές περιβάλλοντος μπορούν να αναπαραστήσουν οποιοδήποτε σύνολο κόσμων με την ενσωμάτωση διάφορων όρων προσδιοριστή περιβάλλοντος. Πρέπει πάντως να τονιστεί ότι οι όροι ενός προσδιοριστή περιβάλλοντος είναι δυνατόν να αναπαριστούν επικαλυπτόμενα σύνολα κόσμων, σύνολα που είναι υποσύνολα άλλων, κτλ. Παρόλο που αυτός ο πλεονασμός δεν επηρεάζει τις πράξεις στους προσδιοριστές περιβάλλοντος *per se*, μπορεί εντούτοις να σημαίνει αυξημένη πολυπλοκότητα για τις πράξεις περιβάλλοντος. Έτσι είναι χρήσιμο να εκφράσουμε τους προσδιοριστές περιβάλλοντος με οικονομικό τρόπο απαλείφοντας πλεονάζοντα στοιχεία.

#### Παράδειγμα 3.9

Το παράδειγμα αυτό επιδεικνύει τις πιθανές περιπτώσεις απλοποίησης. Αρχίζει με έναν αριθμό όρων και συνεχίζει με συνδυασμούς αυτών των όρων σε προσδιοριστές περιβάλλοντος.

$$\begin{aligned} c^{cl}_1 &= \{ (\text{lang}, \{ \text{en}, \text{gr} \}) \} \\ c^{cl}_2 &= \{ (\text{lang}, \{ \text{en}, \text{gr} \}), (\text{detail}, \{ \text{low}, \text{high} \}) \} \\ c^{cl}_3 &= \{ (\text{lang}, \{ \text{en}, \text{gr} \}), (\text{detail}, \{ \text{low} \}) \} \\ c^{cl}_4 &= \{ (\text{lang}, \{ \text{sp}, \text{gr} \}), (\text{detail}, \{ \text{low} \}) \} \\ \\ c_1 &= \{ c^{cl}_a, \mathcal{U}^{cl}, c^{cl}_b, \dots \} =^c \{ \mathcal{U}^{cl} \} = \mathcal{U} \\ c_2 &= \{ c^{cl}_a, \mathcal{E}^{cl}, c^{cl}_b, \dots \} =^c \{ c^{cl}_a, c^{cl}_b, \dots \} \end{aligned}$$



$$\begin{aligned}
c_3 &= \{ \mathcal{E}^{c_1}, \mathcal{E}^{c_1}, \dots, \mathcal{E}^{c_1} \} =^c \{ \mathcal{E}^{c_1} \} = \mathcal{E} \\
c_4 &= \{ c^{c_1}_1, c^{c_1}_2 \} =^c \{ c^{c_1}_1 \} \\
c_5 &= \{ c^{c_1}_2, c^{c_1}_3 \} =^c \{ c^{c_1}_2 \} \\
c_6 &= \{ c^{c_1}_3, c^{c_1}_4 \} =^c \{ ( \text{lang}, \{ \text{en}, \text{gr}, \text{sp} \} ), (\text{detail}, \{ \text{low} \}) \} \\
c_7 &= \{ c^{c_1}_2, c^{c_1}_3, c^{c_1}_4 \} =^c \{ c^{c_1}_2, c^{c_1}_4 \} =^c \\
&\quad \{ c^{c_1}_2, \{ ( \text{lang}, \{ \text{en}, \text{gr}, \text{sp} \} ), (\text{detail}, \{ \text{low} \}) \} \}
\end{aligned}$$

♦

Σημειώστε ότι στο Παράδειγμα 3.9,  $c^{c_1}_3 \subseteq^{cl} c^{c_1}_2 \subseteq^{cl} c^{c_1}_1$ . Αυτό εξηγεί τις πιθανές εναλλακτικές μορφές των περιβαλλόντων  $c_4$  και  $c_5$ , μια και οι όροι που είναι υποσύνολα όρου δεν συνεισφέρουν επιπλέον κόσμους σε ένα περιβάλλον. Το ίδιο ισχύει για τα  $c_1$ ,  $c_2$  και  $c_3$ , μια και  $\mathcal{E}^{c_1} \subseteq^{cl} \mathcal{E}^{c_1} \subseteq^{cl} c^{c_1} \subseteq^{cl} \mathcal{U}^{c_1}$  για κάθε όρο  $c^{c_1}$ . Έτσι, μια πιθανή απλοποίηση ενός προσδιοριστή περιβάλλοντος συνίσταται στην απαλοιφή των όρων που είναι υποσύνολα άλλων όρων σε αυτόν τον προσδιοριστή περιβάλλοντος.

Αν ένας προσδιοριστής περιβάλλοντος περιέχει κάποιον καθολικό όρο  $\mathcal{U}^{c_1}$ , όλοι οι άλλοι όροι του προσδιοριστή περιβάλλοντος μπορούν να παραλειφθούν, μια και ο καθολικός όρος ήδη αναπαριστά κάθε πιθανό κόσμο. Αυτή είναι η περίπτωση του  $c_1$  στο Παράδειγμα 3.9.

Σε έναν προσδιοριστή περιβάλλοντος, κενοί όροι  $\mathcal{E}^{c_1}$  δεν συνεισφέρουν κόσμους στο τελικό αποτέλεσμα. Συνεπώς, αν αυτοί οι όροι παραλειφθούν από τον προσδιοριστή περιβάλλοντος, το σύνολο των κόσμων που αναπαριστά αυτός ο προσδιοριστής περιβάλλοντος δεν θα αλλάξει. Σημειώστε, πάντως, ότι δεν είναι δυνατόν να παραλειφθεί ένας όρος σε περίπτωση που είναι ο μόνος όρος ενός προσδιοριστή περιβάλλοντος, επειδή ένας προσδιοριστής περιβάλλοντος έχει οριστεί σαν μη κενό σύνολο. Οι περιπτώσεις αυτές αντιστοιχούν στα  $c_2$  και  $c_3$  στο Παράδειγμα 3.9.

Το περιβάλλον  $c_6$  δείχνει ότι δυο ή περισσότεροι όροι μπορούν να συνδυαστούν σε ένα μόνο όρο. Αυτό είναι δυνατόν εφόσον οι όροι διαφέρουν ακριβώς σε έναν προσδιοριστή διάστασης που αφορά την ίδια διάσταση. Τότε ο όρος του αποτελέσματος θα συνίσταται από όλους τους υπόλοιπους προσδιοριστές διάστασης (που πρέπει να είναι κοινοί και στους δυο όρους) συν τον προσδιοριστή διάστασης  $(d, V \cup V')$ , όπου  $(d, V)$  ανήκει στον πρώτο όρο και  $(d, V')$  ανήκει στον δεύτερο. Αυτό μπορεί να θεωρηθεί σαν μια περίπτωση ένωσης όρου που επιστρέφει έναν όρο, όπως εξηγήθηκε στην Ενότητα 3.3.2.6, η οποία πάντως δεν είναι πάντα δυνατή όπως επιδεικνύει το Παράδειγμα 3.11.

Η διαδικασία της απλοποίησης δεν οδηγεί πάντοτε σε έναν προσδιοριστή περιβάλλοντος της ίδιας μορφής. Με άλλα λόγια, αν δυο περιβάλλοντα είναι ίσα-περιβάλλοντα, οι απλοποιημένες μορφές τους πιθανώς να μην είναι ίσες με την συμβατική έννοια της ισότητας συνόλου (παρόλο που θα εξακολουθήσουν να είναι ίσα-περιβάλλοντα). Αυτό φαίνεται στο Παράδειγμα 3.9 με τον προσδιοριστή περιβάλλοντος  $c_7$ , ο οποίος μπορεί να μετασχηματιστεί σε δυο προσδιοριστές περιβάλλοντος με διαφορετική μορφή, και για τους οποίους δεν είναι δυνατή περαιτέρω απλοποίηση.

Τα πεδία ορισμού των διαστάσεων πρέπει να ληφθούν υπόψη προκειμένου να αναγνωρισθούν τα καθολικά περιβάλλοντα, και για να αντιμετωπισθούν προσδιοριστές διάστασης της μορφής  $(d, \mathbf{V}_d)$  που μπορεί να οδηγήσουν σε περαιτέρω απλοποίηση των όρων.

### 3.3.2.5 Τομή περιβάλλοντος

Η **τομή περιβάλλοντος**  $\cap^c$  είναι μια πράξη που λαμβάνει δυο προσδιοριστές περιβάλλοντος  $c_1$ ,  $c_2$  και επιστρέφει έναν προσδιοριστή περιβάλλοντος  $c_3$ .

$$\cap^c : (C \times C) \rightarrow C, \text{ όπου } C \text{ το σύνολο των προσδιοριστών περιβάλλοντος}$$

Η τομή περιβάλλοντος ορίζεται ως εξής:

**Ορισμός 3.21**

Ας είναι  $c_1, c_2$  δυο προσδιοριστές περιβάλλοντος. Η **τομή περιβάλλοντος**  $c_1 \cap^c c_2$  είναι ένας προσδιοριστής περιβάλλοντος  $c_3 = \{c_n^{cl} \cap^{cl} c_m^{cl} \mid c_n^{cl} \in c_1, c_m^{cl} \in c_2\}$ .

Η τομή περιβάλλοντος ενέχει την τομή όρου για κάθε ζεύγος όρων των προσδιοριστών περιβάλλοντος.

**Παράδειγμα 3.10**

Θεωρήστε τα ακόλουθα:

$$\begin{aligned} c_1 &= \{ \{ (\text{lang}, \{ \text{en}, \text{gr} \}), (\text{detail}, \{ \text{high} \}) \} \} \\ c_2 &= \{ \{ (\text{lang}, \{ \text{en} \}), (\text{detail}, \{ \text{low} \}) \}, \{ (\text{lang}, \{ \text{gr} \}) \} \} \\ c_3 &= \{ \mathcal{E}^{c1}, \{ (\text{format}, \{ \text{ps}, \text{pdf} \}) \} \} \end{aligned}$$

$$c_1 \cap^c c_2 = \{ \mathcal{E}^{c1}, \{ (\text{lang}, \{ \text{gr} \}), (\text{detail}, \{ \text{high} \}) \} \}$$

$$c_1 \cap^c c_3 = \{ \mathcal{E}^{c1}, \{ (\text{lang}, \{ \text{en}, \text{gr} \}), (\text{detail}, \{ \text{high} \}), (\text{format}, \{ \text{ps}, \text{pdf} \}) \} \}$$

◆

Η τομή περιβάλλοντος δυο προσδιοριστών περιβάλλοντος αναπαριστά τους κόσμους που είναι κοινói και στους δυο προσδιοριστές περιβάλλοντος, όπως δείχνει η ακόλουθη πρόταση.

**Πρόταση 3.10**

Αν  $c_1, c_2$  είναι προσδιοριστές περιβάλλοντος και  $W_D(c_1), W_D(c_2)$  είναι τα σύνολα κόσμων σ.σ.μ. το  $\mathbf{D}$  που αναπαριστούν, τότε  $W_D(c_1 \cap^c c_2) = W_D(c_1) \cap W_D(c_2)$ .

Από την Πρόταση 3.10 προκύπτει ότι η τομή περιβάλλοντος οποιουδήποτε προσδιοριστή περιβάλλοντος με το κενό περιβάλλον  $\mathcal{E}$  δίνει πάντα το κενό περιβάλλον  $\mathcal{E}$ . Επιπλέον, για κάθε προσδιοριστή περιβάλλοντος  $c$ , η τομή περιβάλλοντος με το καθολικό περιβάλλον  $\mathcal{U}$  δίνει πάντα έναν προσδιοριστή περιβάλλοντος που αναπαριστά τους ίδιους κόσμους με το  $c$ , σ.σ.μ. οποιοδήποτε σύνολο διαστάσεων  $\mathbf{D}$ .

$$\begin{aligned} c \cap^c \mathcal{E} &= \mathcal{E} \\ c \cap^c \mathcal{U} &=^c c \end{aligned}$$

**Τύπος 3.10**

**Τύπος 3.11**

Σημειώστε ότι ο Τύπος 3.11 χρησιμοποιεί ισότητα περιβάλλοντος αντί για συμβατική ισότητα. Συνεπώς, το αποτέλεσμα της τομής περιβάλλοντος που εκφράζει ο Τύπος 3.11 δεν είναι κατ' ανάγκη το  $c$ , αλλά ένας προσδιοριστής περιβάλλοντος που είναι ίσο-περιβάλλον με το  $c$ .

Όπως και η τομή όρου, η τομή περιβάλλοντος εκτελείται χωρίς να παίρνει υπόψη κάποιο σύνολο διαστάσεων. Με άλλα λόγια, αν  $c_1 \cap^c c_2 = c_3$  σ.σ.μ. το  $\mathbf{D}$  και  $c_1 \cap^c c_2 = c_4$  σ.σ.μ. το  $\mathbf{D}'$ , τότε το  $c_3$  και το  $c_4$  είναι ταυτόσημα. Επιπλέον, όπως δείχνει η Πρόταση 3.10, το  $c_3$  αναπαριστά τους κόσμους σ.σ.μ. το  $\mathbf{D}$  που είναι κοινói στο  $c_1$  και στο  $c_2$ , και το  $c_4$  αναπαριστά τους κόσμους σ.σ.μ. το  $\mathbf{D}'$  που είναι κοινói στο  $c_1$  και στο  $c_2$ . Συνεπώς, αν το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται στα  $c_1$  και  $c_2$  και καμία άλλη, το αποτέλεσμα της τομής περιβάλλοντος μπορεί να ερμηνευθεί ορθά σ.σ.μ. οποιοδήποτε  $\mathbf{D}$ , όπου  $\mathbf{D}_{\min} \subseteq \mathbf{D}$ .

**Αμοιβαία αποκλειόμενα περιβάλλοντα**

Δυο προσδιοριστές περιβάλλοντος είναι αμοιβαία αποκλειόμενοι αν αναπαριστούν ξένα σύνολα κόσμων.

**Ορισμός 3.22**

Δυο προσδιοριστές περιβάλλοντος  $c_1, c_2$  είναι **αμοιβαία αποκλειόμενοι**, εάν και μόνον  $W_D(c_1) \cap W_D(c_2) = \emptyset$ , όπου  $W_D(c)$  είναι το σύνολο των κόσμων που αναπαριστά το  $c$  σ.σ.μ. κάποιο σύνολο διαστάσεων  $\mathbf{D}$ .

Από την Πρόταση 3.10 βλέπουμε ότι δυο περιβάλλοντα είναι αμοιβαία αποκλειόμενα αν και μόνον αν  $c_1 \cap^c c_2 = \mathcal{E}$ . Συνεπώς, για να είναι δυο περιβάλλοντα αμοιβαία αποκλειόμενα, το αποτέλεσμα της τομής περιβάλλοντος ανάμεσά τους πρέπει να είναι ένα κενό περιβάλλον  $\mathcal{E}$ .

Παρόλο που ο αμοιβαίος αποκλεισμός περιβάλλοντος ορίζεται με αναφορά σε ένα σύνολο διαστάσεων  $\mathbf{D}$ , στην πραγματικότητα δεν ορίζεται σε σχέση με κάποιο  $\mathbf{D}$ . Αφότου η τομή περιβάλλοντος δεν εξαρτάται από το  $\mathbf{D}$ , εξυπακούεται ότι αν δυο περιβάλλοντα είναι αμοιβαία αποκλειόμενα σ.σ.μ. το  $\mathbf{D}$ , είναι επίσης αμοιβαία αποκλειόμενα σ.σ.μ. οποιοδήποτε  $\mathbf{D}'$ , όπου  $\mathbf{D}'$  και  $\mathbf{D}$  είναι υπερσύνολα του  $\mathbf{D}_{\min}$ .

Ο Τύπος 3.10 δίνει το συμπέρασμα ότι ένα κενό περιβάλλον  $\mathcal{E}$  είναι αμοιβαία αποκλειόμενο με όλα τα άλλα περιβάλλοντα, συμπεριλαμβανομένων των κενών περιβαλλόντων. Ο Τύπος 3.11 δείχνει ότι ένα καθολικό περιβάλλον  $\mathcal{U}$  δεν είναι αμοιβαία αποκλειόμενο με κανένα περιβάλλον εκτός από το κενό περιβάλλον.

**3.3.2.6 Ένωση περιβάλλοντος και ένωση όρου**

Στην Ενότητα 3.3.1.5 αναφέραμε ότι οι όροι προσδιοριστή περιβάλλοντος δεν είναι κλειστοί κάτω από τις πράξεις της ένωσης και της διαφοράς. Παρόλο που σε κάποιες περιπτώσεις η ένωση δυο όρων μπορεί να αναπαρασταθεί σαν ένας όρος, στην γενική περίπτωση μπορεί να αναπαρασταθεί μόνον σαν προσδιοριστής περιβάλλοντος. Οι προσδιοριστές περιβάλλοντος, από την άλλη, είναι κλειστοί κάτω από την πράξη της ένωσης περιβάλλοντος. Πριν ορίσουμε την ένωση περιβάλλοντος, θα ορίσουμε την ένωση όρου. Η ένωση όρου δεν χρησιμοποιείται στον ορισμό της ένωσης περιβάλλοντος και συμπεριλαμβάνεται εδώ για λόγους πληρότητας.

**Ένωση όρου**

Η **ένωση όρου**  $\cup^{cl}$  είναι μια πράξη που λαμβάνει δυο όρους προσδιοριστή περιβάλλοντος  $c^{cl}_1, c^{cl}_2$  και επιστρέφει έναν προσδιοριστή περιβάλλοντος  $c$ .

$\cup^{cl} : (C^{cl} \times C^{cl}) \rightarrow C$ , όπου  $C^{cl}$  το σύνολο όλων των όρων, και  $C$  το σύνολο των προσδιοριστών περιβάλλοντος

Η ένωση όρου ορίζεται ως εξής:

**Ορισμός 3.23**

Ας είναι  $c^{cl}_1, c^{cl}_2$  δυο όροι προσδιοριστή περιβάλλοντος. Η **ένωση όρου**  $c^{cl}_1 \cup^{cl} c^{cl}_2$  είναι ο προσδιοριστής περιβάλλοντος  $\{c^{cl}_1, c^{cl}_2\}$ .

**Παράδειγμα 3.11**

Θεωρήστε τα παρακάτω:

$c^{cl}_1 = \{(\text{lang}, \{\text{en}, \text{gr}\}), (\text{detail}, \{\text{high}\})\}$   
 $c^{cl}_2 = \{(\text{lang}, \{\text{en}, \text{gr}\}), (\text{detail}, \{\text{low}\})\}$   
 $c^{cl}_3 = \{(\text{lang}, \{\text{sp}\}), (\text{detail}, \{\text{low}\})\}$

$c^{cl}_1 \cup^{cl} c^{cl}_2 =$   
 $\{ \{(\text{lang}, \{\text{en}, \text{gr}\}), (\text{detail}, \{\text{high}\})\},$   
 $\{(\text{lang}, \{\text{en}, \text{gr}\}), (\text{detail}, \{\text{low}\})\} \}$

$$\begin{aligned}
&=^c \{ \{ (\text{lang}, \{ \text{en}, \text{gr} \}), (\text{detail}, \{ \text{high}, \text{low} \}) \} \} \\
c^{cl_2} \cup^{cl} c^{cl_3} &= \\
&\{ \{ (\text{lang}, \{ \text{en}, \text{gr} \}), (\text{detail}, \{ \text{low} \}) \}, \\
&\quad \{ (\text{lang}, \{ \text{sp} \}), (\text{detail}, \{ \text{low} \}) \} \} \\
&=^c \{ \{ (\text{lang}, \{ \text{en}, \text{gr}, \text{sp} \}), (\text{detail}, \{ \text{low} \}) \} \} \\
c^{cl_1} \cup^{cl} c^{cl_3} &= \\
&\{ \{ (\text{lang}, \{ \text{en}, \text{gr} \}), (\text{detail}, \{ \text{high} \}) \}, \\
&\quad \{ (\text{lang}, \{ \text{sp} \}), (\text{detail}, \{ \text{low} \}) \} \}
\end{aligned}$$

◆

Είναι ενδιαφέρον να σημειώσει κανείς ότι στην περίπτωση των  $c^{cl_1} \cup^{cl} c^{cl_2}$  και  $c^{cl_2} \cup^{cl} c^{cl_3}$  το αποτέλεσμα είναι ίσο-περιβάλλον με ένα περιβάλλον που περιέχει μόνον έναν όρο· συνεπώς στις περιπτώσεις αυτές η ένωση δυο όρων μπορεί να εκφραστεί πιο συμπαγώς σαν ένας και μόνον όρος. Στην περίπτωση όμως του  $c^{cl_1} \cup^{cl} c^{cl_3}$ , μια τέτοια «παραγοντοποίηση» δεν είναι δυνατή, και το αποτέλεσμα μπορεί να εκφραστεί μόνο σαν προσδιοριστής περιβάλλοντος.

### Πρόταση 3.11

*Αν τα  $c^{cl_1}$ ,  $c^{cl_2}$  είναι όροι προσδιοριστή περιβάλλοντος και τα  $W_D(c^{cl_1})$ ,  $W_D(c^{cl_2})$  είναι τα σύνολα των κόσμων σ.σ.μ. κάποιο  $\mathbf{D}$  που αυτοί αναπαριστούν, τότε  $W_D(c^{cl_1} \cup^{cl} c^{cl_2}) = W_D(c^{cl_1}) \cup W_D(c^{cl_2})$ .*

Σημειώστε ότι η ένωση όρου εκτελείται χωρίς να παίρνει υπόψη κάποιο σύνολο διαστάσεων. Συνεπώς, αν το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται στα  $c_1$  και  $c_2$  και καμία άλλη, το αποτέλεσμα της ένωσης όρου μπορεί να ερμηνευτεί ορθά σ.σ.μ. οποιοδήποτε  $\mathbf{D}$ , όπου  $\mathbf{D}_{\min} \subseteq \mathbf{D}$ .

### Ένωση περιβάλλοντος

Η **ένωση περιβάλλοντος**  $\cup^c$  είναι μια πράξη που λαμβάνει δυο προσδιοριστές περιβάλλοντος  $c_1$ ,  $c_2$  και επιστρέφει έναν προσδιοριστή περιβάλλοντος  $c_3$ .

$\cup^c : (C \times C) \rightarrow C$ , όπου  $C$  το σύνολο των προσδιοριστών περιβάλλοντος

Η ένωση περιβάλλοντος ορίζεται ως ακολούθως:

### Ορισμός 3.24

*Ας είναι  $c_1$ ,  $c_2$  δυο προσδιοριστές περιβάλλοντος. Η ένωση περιβάλλοντος  $c_1 \cup^c c_2$  είναι ο προσδιοριστής περιβάλλοντος  $c_1 \cup c_2$ .*

Συνεπώς, η ένωση περιβάλλοντος δυο προσδιοριστών περιβάλλοντος περιέχει τους όρους και των δυο προσδιοριστών περιβάλλοντος. Η ένωση περιβάλλοντος δεν είναι τίποτε άλλο παρά η συμβατική ένωση συνόλων, και εισάγεται εδώ για λόγους ομοιομορφίας του συμβολισμού.

### Παράδειγμα 3.12

Θεωρήστε τα παρακάτω:

$$\begin{aligned}
c_1 &= \{ \{ (\text{lang}, \{ \text{en}, \text{gr} \}), (\text{detail}, \{ \text{high} \}) \} \} \\
c_2 &= \{ \{ (\text{lang}, \{ \text{en} \}), (\text{detail}, \{ \text{low} \}) \}, \{ (\text{lang}, \{ \text{gr} \}) \} \} \\
c_1 \cup^c c_2 &= c_1 \cup c_2 = \\
&\{ \{ (\text{lang}, \{ \text{en}, \text{gr} \}), (\text{detail}, \{ \text{high} \}) \}, \\
&\quad \{ (\text{lang}, \{ \text{en} \}), (\text{detail}, \{ \text{low} \}) \}, \{ (\text{lang}, \{ \text{gr} \}) \} \}
\end{aligned}$$

◆

Η ένωση περιβάλλοντος δυο προσδιοριστών περιβάλλοντος αναπαριστά την ένωση των κόσμων που ορίζουν τα δυο περιβάλλοντα, όπως δείχνει η ακόλουθη πρόταση.

**Πρόταση 3.12**

*Αν τα  $c_1, c_2$  είναι προσδιοριστές περιβάλλοντος και τα  $W_D(c_1), W_D(c_2)$  είναι τα σύνολα κόσμων σ.σ.μ. το  $\mathbf{D}$  που αυτοί αναπαριστούν, τότε  $W_D(c_1 \cup c_2) = W_D(c_1) \cup W_D(c_2)$ .*

Από την Πρόταση 3.12, η ένωση περιβάλλοντος οποιουδήποτε προσδιοριστή περιβάλλοντος με ένα καθολικό περιβάλλον  $\mathcal{U}$  δίνει πάντοτε ένα καθολικό περιβάλλον  $\mathcal{U}$ . Για οποιοδήποτε προσδιοριστή περιβάλλοντος  $c$ , η ένωση περιβάλλοντος με ένα κενό περιβάλλον  $\mathcal{E}$  δίνει πάντοτε έναν προσδιοριστή περιβάλλοντος που αναπαριστά τους ίδιους κόσμους με το  $c$ .

$$\begin{aligned} c \cup^c \mathcal{U} &= \mathcal{U} \\ c \cup^c \mathcal{E} &=^c c \end{aligned}$$

**Τύπος 3.12**

**Τύπος 3.13**

Σημειώστε ότι ο Τύπος 3.13 χρησιμοποιεί ισότητα περιβάλλοντος αντί για συμβατική ισότητα. Συνεπώς, το αποτέλεσμα της ένωσης περιβάλλοντος που περιγράφει ο Τύπος 3.13 δεν είναι απαραίτητα το  $c$ , αλλά ένας προσδιοριστής περιβάλλοντος που είναι ίσο-περιβάλλον με το  $c$ .

Όπως και η τομή περιβάλλοντος, η ένωση περιβάλλοντος εκτελείται χωρίς να πάρει υπόψη κάποιο σύνολο διαστάσεων. Με άλλα λόγια, αν  $c_1 \cup^c c_2 = c_3$  σ.σ.μ. το  $\mathbf{D}$  και  $c_1 \cup^c c_2 = c_4$  σ.σ.μ. το  $\mathbf{D}'$ , τότε τα  $c_3$  και  $c_4$  είναι ταυτόσημα. Επιπλέον, όπως δείχνει η Πρόταση 3.12, το  $c_3$  αναπαριστά τους κόσμους σ.σ.μ. το  $\mathbf{D}$  του  $c_1$  και του  $c_2$ , και το  $c_4$  αναπαριστά τους κόσμους σ.σ.μ. το  $\mathbf{D}'$  του  $c_1$  και του  $c_2$ . Συνεπώς, αν το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται στα  $c_1$  και  $c_2$  και καμία άλλη, το αποτέλεσμα της ένωσης περιβάλλοντος μπορεί να ερμηνευτεί ορθά σ.σ.μ. οποιοδήποτε  $\mathbf{D}$ , όπου  $\mathbf{D}_{\min} \subseteq \mathbf{D}$ .

### 3.3.2.7 Διαφορά περιβάλλοντος και διαφορά όρου

Στην Ενότητα 3.3.1.5 αναφέραμε ότι οι όροι προσδιοριστή περιβάλλοντος δεν είναι κλειστοί κάτω από την πράξη της αφαίρεσης. Ακριβώς όπως είδαμε να συμβαίνει με την ένωση όρου, υπάρχουν περιπτώσεις όπου η διαφορά δυο όρων μπορεί να αναπαρασταθεί σαν ένας όρος, στην γενική περίπτωση πάντως μπορεί να αναπαρασταθεί μόνον σαν προσδιοριστής περιβάλλοντος. Οι προσδιοριστές περιβάλλοντος είναι κλειστοί κάτω από την πράξη της διαφοράς περιβάλλοντος. Προτού ορίσουμε την διαφορά περιβάλλοντος, θα συζητήσουμε την διαφορά όρου.

#### Διαφορά όρου

Η **διαφορά όρου**  $-^{cl}$  είναι μια πράξη που δέχεται δυο όρους προσδιοριστή περιβάλλοντος  $c^{cl}_1, c^{cl}_2$  και επιστρέφει έναν προσδιοριστή περιβάλλοντος  $c$ .

$$-^{cl} : (C^{cl} \times C^{cl}) \rightarrow C, \text{ όπου } C^{cl} \text{ το σύνολο όλων των όρων, και } C \text{ το σύνολο των προσδιοριστών περιβάλλοντος}$$

Η διαφορά όρου ορίζεται παρακάτω.

**Ορισμός 3.25**

*Ας είναι  $c^{cl}_1, c^{cl}_2$  δυο όροι προσδιοριστή περιβάλλοντος. Η διαφορά όρου των  $c^{cl}_1$  και  $c^{cl}_2$ , συμβολίζεται  $c^{cl}_1 -^{cl} c^{cl}_2$ , και είναι ένας προσδιοριστής περιβάλλοντος  $c$  που ορίζεται ως εξής:*

1. Αν  $(d, \emptyset) \in c^{cl}_2$ , τότε  $c = \{c^{cl}_1\}$ .
2. Αν  $c^{cl}_1 = \{-\}$  ή αν  $c^{cl}_2 = \emptyset$ , τότε  $c = \{\{-\}\}$ .
3. Αν  $(d, \emptyset) \notin c^{cl}_2$  και  $c^{cl}_2 \neq \emptyset$  και  $c^{cl}_1 \neq \{-\}$ , τότε το  $c$  περιέχει τους ακόλουθους όρους: (α) για κάθε προσδιοριστή διάστασης  $(d, V') \in c^{cl}_2$  τέτοιον ώστε  $(d, V) \in c^{cl}_1$ , ο όρος  $\{(d, V-V')\} \cup \{(d_i, V_i) \mid (d_i, V_i) \in c^{cl}_1 \text{ με } d_i \neq d\}$  ανήκει στο  $c$ , και (β) για κάθε προσδιοριστή διάστασης  $(d, V') \in c^{cl}_2$  τέτοιον ώστε  $(d, V) \notin c^{cl}_1$ , ο όρος  $\{(d, \mathbf{V}_d - V')\} \cup \{(d_i, V_i) \mid (d_i, V_i) \in c^{cl}_1\}$  ανήκει στο  $c$ .

Σημειώστε ότι εάν  $c^{cl}_2 = \{-\}$  τότε, σύμφωνα με ότι λέει ο Ορισμός 3.10,  $(d, \emptyset) \in c^{cl}_2$  για κάθε δυνατό σύνολο διαστάσεων. Η διαφορά όρου μπορεί να γίνει καλύτερα αντιληπτή μέσα από το παρακάτω παράδειγμα.

### Παράδειγμα 3.13

Στην συνέχεια παρατίθεται ένας αριθμός περιπτώσεων διαφοράς όρου. Υπενθύμιση: το αποτέλεσμα της διαφοράς όρου είναι ένας προσδιοριστής περιβάλλοντος.

$\mathbf{V}_{\text{lang}} = \{\text{en, gr, sp}\}$

$\mathbf{V}_{\text{detail}} = \{\text{low, medium, high}\}$

$\mathbf{V}_{\text{format}} = \{\text{ps, pdf, html, doc}\}$

$\mathbf{V}_{\text{currency}} = \{\text{Euro, USD}\}$

- 1.

$c^{cl}_1 = \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{low, high}\})\}$

$c^{cl}_2 = \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{low}\})\}$

$c^{cl}_1 \text{ } ^{-cl} \text{ } c^{cl}_2 = \{$   
 $\{(\text{lang}, \{\}), (\text{detail}, \{\text{low, high}\})\},$   
 $\{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{high}\})\} \}$

- 2.

$c^{cl}_1 = \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{low, high}\}), (\text{format}, \{\text{pdf}\})\}$

$c^{cl}_2 = \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{low}\})\}$

$c^{cl}_1 \text{ } ^{-cl} \text{ } c^{cl}_2 = \{$   
 $\{(\text{lang}, \{\}), (\text{detail}, \{\text{low, high}\}), (\text{format}, \{\text{pdf}\})\},$   
 $\{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{high}\}), (\text{format}, \{\text{pdf}\})\} \}$

- 3.

$c^{cl}_1 = \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{low}\})\}$

$c^{cl}_2 = \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{low, high}\})\}$

$c^{cl}_1 \text{ } ^{-cl} \text{ } c^{cl}_2 = \{$   
 $\{(\text{lang}, \{\}), (\text{detail}, \{\text{low}\})\},$   
 $\{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\})\} \}$

- 4.

$c^{cl}_1 = \{(\text{lang}, \{\text{en}\}), (\text{detail}, \{\text{low, medium, high}\}), (\text{format}, \{\text{pdf, ps}\})\}$

$c^{cl}_2 = \{(\text{lang}, \{\text{en}\}), (\text{detail}, \{\text{low}\}), (\text{format}, \{\text{pdf, html, doc}\})\}$

$c^{cl}_1 \text{ } ^{-cl} \text{ } c^{cl}_2 = \{$   
 $\{(\text{lang}, \{\}), (\text{detail}, \{\text{low, medium, high}\}), (\text{format}, \{\text{pdf, ps}\})\},$   
 $\{(\text{lang}, \{\text{en}\}), (\text{detail}, \{\text{medium, high}\}), (\text{format}, \{\text{pdf, ps}\})\},$   
 $\{(\text{lang}, \{\text{en}\}), (\text{detail}, \{\text{low, medium, high}\}), (\text{format}, \{\text{ps}\})\} \}$

- 5.

$c^{cl}_1 = \{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{low, high}\})\}$

$c^{cl}_2 = \{(\text{lang}, \{\text{en}\}), (\text{format}, \{\text{ps, pdf}\}), (\text{currency}, \{\text{USD}\})\}$

$c^{cl}_1 \text{ } ^{-cl} \text{ } c^{cl}_2 = \{$   
 $\{(\text{lang}, \{\text{gr}\}), (\text{detail}, \{\text{low, high}\})\},$   
 $\{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{low, high}\}), (\text{format}, \{\text{html, doc}\})\},$   
 $\{(\text{lang}, \{\text{en, gr}\}), (\text{detail}, \{\text{low, high}\}), (\text{currency}, \{\text{Euro}\})\} \}$

◆

Για τις περιπτώσεις που καλύπτει ο Ορισμός 3.25-(3), το αποτέλεσμα της διαφοράς όρου έχει τόσους όρους όσοι προσδιοριστές διάστασης ( $d, V'$ ) υπάρχουν στο  $c^{cl}_2$ . Στις περιπτώσεις (1) και (2) στο Παράδειγμα 3.13, οι πρώτοι όροι της διαφοράς όρου δεν αναπαριστούν κανένα κόσμο, και κάθε όρος του αποτελέσματος περιέχει προσδιοριστές διάστασης που αντιστοιχούν επακριβώς στους προσδιοριστές διάστασης του  $c^{cl}_1$ . Και στις δυο περιπτώσεις (1) και (2), η τιμή της διάστασης `detail` στον αντίστοιχο (δευτερο) όρο του αποτελέσματος είναι η συμβατική διαφορά συνόλων των αντιστοίχων τιμών στο  $c^{cl}_1$  και στο  $c^{cl}_2$ . Σημειώστε ότι οι διαστάσεις που υπάρχουν στο  $c^{cl}_1$  αλλά όχι στο  $c^{cl}_2$  απλά μεταφέρονται στο αποτέλεσμα, όπως το `format` στην περίπτωση (2). Η περίπτωση (5) επιδεικνύει την περίπτωση όπου το  $c^{cl}_2$  περιέχει διαστάσεις που δεν υπάρχουν στο  $c^{cl}_1$ . Για κάθε τέτοια διάσταση  $d$ , προστίθεται ένας όρος στο αποτέλεσμα ο οποίος περιέχει την διαφορά από το πεδίο ορισμού της διάστασης  $d$ ,  $V_d$ .

Ο Ορισμός 3.25 δείχνει ότι ισχύουν τα ακόλουθα για τον καθολικό όρο  $\mathcal{U}^{cl}$  και για τον κενό όρο  $\mathcal{E}^{cl}$ :

$$c^{cl} \text{ } ^{-cl} \mathcal{U}^{cl} = \mathcal{E}$$

**Τύπος 3.14**

$$c^{cl} \text{ } ^{-cl} \mathcal{E}^{cl} = \{c^{cl}\}$$

**Τύπος 3.15**

$$\mathcal{E}^{cl} \text{ } ^{-cl} c^{cl} = \mathcal{E}$$

**Τύπος 3.16**

Όπως και οι υπόλοιπες πράξεις όρου, η διαφορά όρου ορίζεται με τέτοιο τρόπο ώστε να υπάρχει μια άμεση αντιστοιχία με τα σύνολα των κόσμων που αναπαριστούν οι όροι· πιο συγκεκριμένα, αντιπροσωπεύει την διαφορά των συνόλων κόσμων που ορίζουν οι εμπλεκόμενοι όροι, καθώς δείχνει η παρακάτω πρόταση.

### Πρόταση 3.13

*Αν τα  $c^{cl}_1, c^{cl}_2$  είναι όροι προσδιοριστή περιβάλλοντος και τα  $W_D(c^{cl}_1), W_D(c^{cl}_2)$  είναι τα σύνολα κόσμων σ.σ.μ. το  $\mathbf{D}$  που αυτοί αναπαριστούν, τότε  $W_D(c^{cl}_1 \text{ } ^{-cl} c^{cl}_2) = W_D(c^{cl}_1) - W_D(c^{cl}_2)$ .*

Σημειώστε ότι για τις περιπτώσεις από (1) μέχρι (4) στο Παράδειγμα 3.13, η διαφορά όρου δεν παίρνει υπόψη τα πεδία ορισμού των διαστάσεων, όπως ακριβώς συμβαίνει με την τομή όρου και την ένωση όρου. Στην περίπτωση (5), πάντως, βλέπουμε ότι εάν το  $c^{cl}_2$  περιέχει έναν προσδιοριστή διάστασης για κάποια διάσταση που δεν περιέχεται στο  $c^{cl}_1$ , τότε το  $V_d$ , το πεδίο ορισμού της διάστασης  $d$ , πρέπει να ληφθεί υπόψη. Προκειμένου να κρατήσουμε το  $V_d$  έξω από την εικόνα, θα μπορούσαμε να είχαμε εισάγει μια ειδική σύνταξη που να συμβολίζει το συμπλήρωμα του  $V_d$ , όπως για παράδειγμα (`currency, not in {USD}`), και (`format, not in {ps, pdf}`). Η προσέγγιση αυτή θα είχε απλά μεταφέρει το πρόβλημα στον ορισμό άλλων πράξεων, όπως η τομή όρου. Στην Ενότητα 3.6 παρουσιάζουμε μια βολική σημειογραφία για το περιβάλλον που εμπεριέχει ένα παρόμοιο συντακτικό για το συμπλήρωμα προσδιοριστή διάστασης. Αυτό πάντως, εισάγεται μόνο σαν συντακτική συντομογραφία, επειδή τα πεδία ορισμού των διαστάσεων θεωρούνται πεπερασμένα και διακριτά, άρα είναι πάντοτε δυνατή μια αντικατάσταση και τα συμπληρώματα μπορούν πάντοτε να απαλειφθούν προτού εκτελεστούν πράξεις περιβάλλοντος. Συνεπώς, η διαφορά όρου ίσως χρειαστεί σε κάποιες περιπτώσεις να λάβει υπόψη τα πεδία ορισμού των διαστάσεων (που θεωρούνται αμετάβλητα, όπως διευκρινίζεται στην Ενότητα 3.2.1).

Όπως η τομή όρου και η ένωση όρου, η διαφορά όρου εκτελείται χωρίς να λαμβάνει υπόψη κάποιο σύνολο διαστάσεων. Με άλλα λόγια, αν  $c^{cl}_1 \text{ } ^{-cl} c^{cl}_2 = c$  σ.σ.μ. το  $\mathbf{D}$  και  $c^{cl}_1 \text{ } ^{-cl} c^{cl}_2 = c'$  σ.σ.μ. το  $\mathbf{D}'$ , τότε το  $c$  και το  $c'$  είναι ταυτόσημα. Επιπλέον, όπως δείχνει η Πρόταση 3.13, το

ε αναπαριστά τους κόσμους σ.σ.μ. το  $\mathbf{D}$  που καλύπτει το  $c_1^{cl}$  αλλά όχι το  $c_2^{cl}$ , και το  $c'$  αναπαριστά τους κόσμους σ.σ.μ. το  $\mathbf{D}'$  που καλύπτει το  $c_1^{cl}$  αλλά όχι το  $c_2^{cl}$ . Συνεπώς, αν το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται στα  $c_1^{cl}$  και  $c_2^{cl}$  και καμία άλλη, τα αποτελέσματα της διαφοράς όρου μπορούν να ερμηνευθούν ορθά σ.σ.μ. οποιοδήποτε  $\mathbf{D}$ , όπου  $\mathbf{D}_{\min} \subseteq \mathbf{D}$ .

### Διαφορά περιβάλλοντος

Η **διαφορά περιβάλλοντος**  $-^c$  είναι μια πράξη που δέχεται δυο προσδιοριστές περιβάλλοντος  $c_1, c_2$  και επιστρέφει έναν προσδιοριστή περιβάλλοντος  $c_3$ .

$-^c : (C \times C) \rightarrow C$ , όπου  $C$  το σύνολο των προσδιοριστών περιβάλλοντος

Η διαφορά περιβάλλοντος ορίζεται παρακάτω με βάση την διαφορά όρου, την τομή περιβάλλοντος, και την ένωση περιβάλλοντος.

#### Ορισμός 3.26

Ας είναι τα  $c_1 = \{c_{11}^{cl}, c_{12}^{cl}, \dots, c_{1n}^{cl}\}$ ,  $c_2 = \{c_{21}^{cl}, c_{22}^{cl}, \dots, c_{2m}^{cl}\}$  προσδιοριστές περιβάλλοντος, με  $n, m \geq 1$ . Η **διαφορά περιβάλλοντος** του  $c_1$  και του  $c_2$ , συμβολίζεται  $c_1 -^c c_2$ , και είναι ένας προσδιοριστής περιβάλλοντος  $c_3$  που ορίζεται σαν η ένωση περιβάλλοντος όλων των προσδιοριστών περιβάλλοντος  $[(c_{1k}^{cl} -^c c_{2l}^{cl}) \cap^c (c_{1k}^{cl} -^c c_{22}^{cl}) \cap^c \dots \cap^c (c_{1k}^{cl} -^c c_{2m}^{cl})]$  όπου  $c_{1k}^{cl} \in c_1, n \geq k \geq 1$ .

Ο προσδιοριστής περιβάλλοντος που δίνεται από το  $\{\emptyset\} -^c c$  καλείται **συμπλήρωμα περιβάλλοντος** του  $c$ .

Το σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$  που αναπαριστά η διαφορά περιβάλλοντος δυο περιβαλλόντων ισούται με την διαφορά των συνόλων κόσμων που αναπαριστούν τα περιβάλλοντα:

#### Πρόταση 3.14

Για δυο προσδιοριστές περιβάλλοντος  $c_1, c_2$  ισχύει το εξής:  $W_D(c_1 -^c c_2) = W_D(c_1) - W_D(c_2)$ .

Από την Πρόταση 3.14, προκύπτουν τα ακόλουθα για τα καθολικά περιβάλλοντα  $\mathcal{U}$  και για τα κενά περιβάλλοντα  $\mathcal{E}$ :

$$c -^c \mathcal{U} = \mathcal{E}$$

Τύπος 3.17

$$c -^c \mathcal{E} = c$$

Τύπος 3.18

$$\mathcal{E} -^c c = \mathcal{E}$$

Τύπος 3.19

Σημειώστε ότι, ο Τύπος 3.18 χρησιμοποιεί συμβατική ισότητα αντί για ισότητα περιβάλλοντος που υπονοείται από την Πρόταση 3.14. Ο Τύπος 3.18 είναι παράγωγο που μας δίνει ο Ορισμός 3.26 και ο Τύπος 3.15.

Ο Ορισμός 3.26 δείχνει ότι μια διαφορά περιβάλλοντος  $c_1 -^c c_2$  ορίζεται με βάση την τομή περιβάλλοντος, την ένωση περιβάλλοντος, και την διαφορά όρου, οι οποίες εκτελούνται χωρίς αν λάβουν υπόψη κάποιο σύνολο διαστάσεων. Συνεπώς, αν το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται στα  $c_1$  και  $c_2$  και καμία άλλη, τα αποτελέσματα της διαφοράς περιβάλλοντος μπορούν να ερμηνευθούν ορθά σ.σ.μ. οποιοδήποτε  $\mathbf{D}$ , όπου  $\mathbf{D}_{\min} \subseteq \mathbf{D}$ .



### 3.3.2.8 Υποσύνολο περιβάλλοντος

Το υποσύνολο περιβάλλοντος  $\subseteq^c$ , και το γνήσιο υποσύνολο περιβάλλοντος  $\subset^c$  είναι πράξεις που λαμβάνουν δυο προσδιοριστές περιβάλλοντος  $c_1, c_2$  και επιστρέφουν μια τιμή αληθείας (Boolean).

$\subseteq^c : (C \times C) \rightarrow \text{Boolean}$ , όπου  $C$  το σύνολο των περιβαλλόντων  
 $\subset^c : (C \times C) \rightarrow \text{Boolean}$ , όπου  $C$  το σύνολο των περιβαλλόντων

Το υποσύνολο / υπερσύνολο περιβάλλοντος και το γνήσιο υποσύνολο / υπερσύνολο περιβάλλοντος ορίζονται ως εξής:

#### Ορισμός 3.27

Ας είναι το  $\mathbf{D}$  ένα μη κενό σύνολο ονομάτων διαστάσεων, και για κάθε  $d \in \mathbf{D}$  ας είναι το  $\mathbf{V}_d$  το πεδίο ορισμού του  $d$ , με  $\mathbf{V}_d \neq \emptyset$ . Ας είναι τα  $c_1, c_2$  προσδιοριστές περιβάλλοντος. Τότε το  $c_1$  είναι υποσύνολο περιβάλλοντος του  $c_2$ , και συμβολίζεται  $c_1 \subseteq^c c_2$ , ή ισοδύναμα το  $c_2$  είναι υπερσύνολο περιβάλλοντος του  $c_1$ , και συμβολίζεται  $c_2 \supseteq^c c_1$ , εάν και μόνον  $W_D(c_1) \subseteq W_D(c_2)$ , όπου  $W_D(c)$  είναι το σύνολο κόσμων σ.σ.μ. το  $\mathbf{D}$  που αναπαριστά το  $c$ .

Εάν  $W_D(c_1) \subset W_D(c_2)$ , τότε το  $c_1$  είναι γνήσιο υποσύνολο περιβάλλοντος του  $c_2$ , και συμβολίζεται  $c_1 \subset^c c_2$ , ή ισοδύναμα το  $c_2$  είναι γνήσιο υπερσύνολο περιβάλλοντος του  $c_1$ , και συμβολίζεται  $c_2 \supset^c c_1$ .

Ο Ορισμός 3.27, δείχνει ότι είναι δυνατόν να αποφανθεί κανείς εάν ένας προσδιοριστής περιβάλλοντος είναι (γνήσιο) υποσύνολο περιβάλλοντος σ.σ.μ. το  $\mathbf{D}$  κάποιου άλλου προσδιοριστή περιβάλλοντος, συγκρίνοντας τα αντίστοιχα σύνολα κόσμων σ.σ.μ. το  $\mathbf{D}$ . Επιπρόσθετα, το υποσύνολο περιβάλλοντος μπορεί να εκφραστεί με βάση την τομή περιβάλλοντος και την ισότητα περιβάλλοντος, ακριβώς όπως συμβαίνει με τα συμβατικά σύνολα.

#### Πρόταση 3.15

Για δυο προσδιοριστές περιβάλλοντος  $c_1, c_2$  ισχύουν τα ακόλουθα: (a)  $c_1 \subseteq^c c_2 \Leftrightarrow c_1 \cap^c c_2 =^c c_1$ , και (b)  $c_1 \subset^c c_2 \Leftrightarrow (c_1 \cap^c c_2 =^c c_1 \text{ KAI } c_1 \neq^c c_2)$ .

Παρόλο που το (γνήσιο) υποσύνολο / υπερσύνολο περιβάλλοντος ορίζεται με αναφορά σε ένα σύνολο διαστάσεων  $\mathbf{D}$ , στην πραγματικότητα δεν ορίζεται σε σχέση με κάποιο  $\mathbf{D}$ <sup>21</sup>. Από την Πρόταση 3.15, και επειδή η τομή περιβάλλοντος και η ισότητα περιβάλλοντος δεν εξαρτώνται από το  $\mathbf{D}$ , είναι εύκολο να δει κανείς ότι, αν το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται στα  $c_1$  και  $c_2$  και καμία άλλη, τότε  $c_1 \subseteq^c_{\mathbf{D}} c_2 \Leftrightarrow c_1 \subseteq^c_{\mathbf{D}'} c_2$  και  $c_1 \subset^c_{\mathbf{D}} c_2 \Leftrightarrow c_1 \subset^c_{\mathbf{D}'} c_2$ ,  $\forall \mathbf{D}$  και  $\mathbf{D}'$  όπου  $\mathbf{D}_{\min} \subseteq \mathbf{D}$  και  $\mathbf{D}_{\min} \subseteq \mathbf{D}'$ .

Αναφορικά με το κενό περιβάλλον και το καθολικό περιβάλλον, είναι προφανές ότι  $\mathcal{E} \subseteq^c c$  και  $c \subseteq^c \mathcal{U}$ , για κάθε προσδιοριστή περιβάλλοντος  $c$ .

<sup>21</sup> Πάντως, ο έλεγχος δυο προσδιοριστών περιβάλλοντος για (γνήσιο) υποσύνολο / υπερσύνολο περιβάλλοντος ίσως χρειαστεί να λάβει υπόψη τα πεδία ορισμού των διαστάσεων που απαντώνται στα περιβάλλοντα αυτά.

### 3.4 ΙΔΙΟΤΗΤΕΣ ΤΩΝ ΠΡΑΞΕΩΝ ΠΕΡΙΒΑΛΛΟΝΤΟΣ

Στην Ενότητα 3.3 εισήγαμε έναν αριθμό πράξεων για τους όρους προσδιοριστή περιβάλλοντος και για τους προσδιοριστές περιβάλλοντος, και διασαφηνίσαμε την σχέση ανάμεσα από περιβάλλοντα και σύνολα κόσμων. Αναφέραμε επίσης κάποιες από τις ιδιότητες των πράξεων περιβάλλοντος, συγκεκριμένα αυτές που αφορούν τα κενά περιβάλλοντα  $\mathcal{E}$  και τα καθολικά περιβάλλοντα  $\mathcal{U}$ . Στην ενότητα αυτή ερευνούμε περισσότερο τις ιδιότητες των πράξεων περιβάλλοντος.

Οι προσδιοριστές περιβάλλοντος έχουν μια σημαντική ιδιότητα που δεν έχουν οι όροι, δηλ. είναι *κλειστοί* κάτω από τις πράξεις της τομής περιβάλλοντος, της ένωσης περιβάλλοντος, και της διαφοράς περιβάλλοντος: από τους ορισμούς αυτών των πράξεων προκύπτει ότι το αποτέλεσμά τους είναι πάντοτε προσδιοριστής περιβάλλοντος. Αυτή είναι και η περίπτωση για την επεκτεταμένη μορφή ενός προσδιοριστή περιβάλλοντος, η οποία είναι επίσης προσδιοριστής περιβάλλοντος. Επιπρόσθετα, δεδομένων κάποιων πεπερασμένων  $\mathbf{D}$  και  $\mathbf{V}_d$  για κάθε  $d$  στο  $\mathbf{D}$ , αν οι εμπλεκόμενοι προσδιοριστές περιβάλλοντος είναι πεπερασμένοι, τότε το αποτέλεσμα των πράξεων περιβάλλοντος (συμπεριλαμβανομένης της ανάπτυξης περιβάλλοντος) θα είναι επίσης πεπερασμένο. Στα ακόλουθα θεωρούμε πεπερασμένους προσδιοριστές περιβάλλοντος.

#### 3.4.1 Εξάρτηση από διαστάσεις

Χωρίζουμε τις πράξεις περιβάλλοντος σε δυο ομάδες. Η πρώτη ομάδα αποτελείται από πράξεις που εκτελούνται χωρίς να παίρνουν υπόψη κάποιο σύνολο διαστάσεων, και αυτές είναι η ανάπτυξη περιβάλλοντος, η τομή περιβάλλοντος, η ένωση περιβάλλοντος, η διαφορά περιβάλλοντος, η ισότητα / ανισότητα περιβάλλοντος, και το (γνήσιο) υποσύνολο / υπερσύνολο περιβάλλοντος. Η δεύτερη ομάδα αποτελείται από τις πράξεις που εκτελούνται σε σχέση με ένα σύνολο διαστάσεων, πράγμα που συμβολίζεται από το  $\mathbf{D}$  κάτω από το σύμβολο της πράξης, και οι πράξεις αυτές είναι η επέκταση περιβάλλοντος, και η ανάπτυξη επεκτεταμένου περιβάλλοντος.

Στα ακόλουθα, αφήνουμε κατά μέρος τις πράξεις που χρησιμοποιούνται για την μετατροπή περιβαλλόντων σε σύνολα κόσμων, δηλαδή την ανάπτυξη περιβάλλοντος, την επέκταση περιβάλλοντος, και την ανάπτυξη επεκτεταμένου περιβάλλοντος. Εστιάζουμε στην ισότητα / ανισότητα περιβάλλοντος, στο (γνήσιο) υποσύνολο / υπερσύνολο περιβάλλοντος, στην τομή περιβάλλοντος, στην ένωση περιβάλλοντος, και στην διαφορά περιβάλλοντος, οι οποίες χρησιμοποιούνται για τον σχηματισμό **εκφράσεων περιβάλλοντος** και **συνθηκών περιβάλλοντος** (που εξηγούνται στην Ενότητα 3.4.4). Οι πράξεις αυτές δίνουν τα ίδια αποτελέσματα σε σχέση με διαφορετικά σύνολα διαστάσεων. Ειδικότερα, είδαμε ότι μπορούν να εκτελεστούν σε σχέση με το  $\mathbf{D}_{\min}$ , το οποίο περιέχει μόνον τις διαστάσεις εκείνες που απαντώνται στα περιβάλλοντα τα οποία εμπλέκονται στην πράξη. Τότε το αποτέλεσμα μπορεί να ερμηνευθεί ορθά σ.σ.μ. οποιοδήποτε  $\mathbf{D}$  που είναι υπερσύνολο του  $\mathbf{D}_{\min}$ .

Αυτό το γεγονός οδηγεί σε μια σειρά από χρήσιμα συμπεράσματα για την ισότητα / ανισότητα περιβάλλοντος, το (γνήσιο) υποσύνολο / υπερσύνολο περιβάλλοντος, την τομή περιβάλλοντος, την ένωση περιβάλλοντος, και την διαφορά περιβάλλοντος:

- (a) Η προσθήκη διαστάσεων σε κάποιο δεδομένο  $\mathbf{D}$  δεν επηρεάζει τα αποτελέσματα των πράξεων που εκτελέστηκαν σ.σ.μ. το  $\mathbf{D}$ .
- (b) Είναι δυνατόν να εκτελέσει κανείς πράξεις περιβάλλοντος χωρίς να γνωρίζει την συγκεκριμένη στιγμή το σύνολο διαστάσεων  $\mathbf{D}$ .

- (c) Η ισότητα / ανισότητα περιβάλλοντος και το (γνήσιο) υποσύνολο / υπερσύνολο περιβάλλοντος μπορούν να θεωρήσουν το  $\mathbf{D}_{\min}$  στη θέση του πραγματικού  $\mathbf{D}$  προκειμένου να βελτιώσουν την αποδοτικότητα τους.

Η τομή περιβάλλοντος και η ένωση περιβάλλοντος μπορούν να θεωρήσουν ότι το πεδίο ορισμού κάθε διάστασης  $d$  είναι το  $\mathbf{V}_{\min-d}$ , το σύνολο όλων των τιμών της  $d$  που απαντώνται στα περιβάλλοντα που εμπλέκονται στην πράξη. Εύκολα βλέπει κανείς ότι τα αποτελέσματα των πράξεων αυτών θα ισχύουν επίσης για κάθε  $\mathbf{V}_d$ , όπου το  $\mathbf{V}_d$  είναι υπερσύνολο του  $\mathbf{V}_{\min-d}$ . Συνεπώς, δεν είναι αναγκαίο να γνωρίζουμε τα πεδία ορισμού των διαστάσεων για να εκτελέσουμε τομή περιβάλλοντος και ένωση περιβάλλοντος. Αντίθετα, η διαφορά περιβάλλοντος είναι μια πράξη που πρέπει να λάβει υπόψη τα πραγματικά πεδία ορισμού των διαστάσεων, μια και το αποτέλεσμά της μπορεί<sup>22</sup> να εξαρτάται από αυτά τα πεδία ορισμού. Για τον ίδιο λόγο<sup>23</sup>, η ισότητα / ανισότητα περιβάλλοντος και το (γνήσιο) υποσύνολο / υπερσύνολο περιβάλλοντος πρέπει επίσης να λάβουν υπόψη τα πραγματικά πεδία ορισμού των διαστάσεων.

### 3.4.2 Ισότητα περιβάλλοντος και υποσύνολο περιβάλλοντος

Στην Ενότητα 3.3 χρησιμοποιήσαμε δυο είδη ισότητας με τους προσδιοριστές περιβάλλοντος, συγκεκριμένα την συμβατική ισότητα συνόλων και την ισότητα περιβάλλοντος. Η συμβατική ισότητα συνόλων δεν είναι πολύ χρήσιμη, γιατί συγκρίνει την μορφή δυο περιβαλλόντων και όχι τους κόσμους που αναπαριστούν. Η ισότητα περιβάλλοντος από την άλλη ορίστηκε με βάση τα σύνολα κόσμων που αναπαριστούν τα περιβάλλοντα. Η συμβατική ισότητα είναι πιο περιοριστική από την ισότητα περιβάλλοντος. Με άλλα λόγια, ισχύει το ακόλουθο:

$$c_1 = c_2 \Rightarrow c_1 =^c c_2$$

Σημειώστε ότι η αντίθετη συνεπαγωγή (προς τα αριστερά) δεν ισχύει.

Λόγω του ορισμού της είναι προφανές ότι η ισότητα περιβάλλοντος είναι:

- (a) Ανακλαστική,  $c_1 =^c c_1$ .
- (b) Συμμετρική,  $c_1 =^c c_2 \Rightarrow c_2 =^c c_1$ .
- (c) Μεταβατική,  $(c_1 =^c c_2, c_2 =^c c_3) \Rightarrow c_1 =^c c_3$ .

Εξαιτίας των ιδιοτήτων (a), (b), και (c), η ισότητα περιβάλλοντος είναι σχέση ισοδυναμίας [SR86], και ορίζει κλάσεις ισοδυναμίας πάνω στο σύνολο όλων των προσδιοριστών περιβάλλοντος. Συγκεκριμένα, για κάθε προσδιοριστή περιβάλλοντος  $c$ , η κλάση ισοδυναμίας του  $c$  σχετικά με την ισότητα περιβάλλοντος είναι το σύνολο των προσδιοριστών περιβάλλοντος  $c'$  τέτοιων ώστε  $c =^c c'$ . Κάθε τέτοια κλάση ισοδυναμίας περιέχει όλα τα περιβάλλοντα που αναπαριστούν το ίδιο σύνολο κόσμων σ.σ.μ. κάποιο (οποιοδήποτε)  $\mathbf{D}$ .

<sup>22</sup> Όπως εξηγήθηκε στην 3.3.2.7, αν ένας όρος στο  $c_2$  περιέχει τον προσδιοριστή διάστασης  $(d, V_2)$  και κάποιος όρος στο  $c_1$  δεν περιέχει έναν αντίστοιχο προσδιοριστή διάστασης  $(d, V_1)$ , τότε το  $c_1 =^c c_2$  θα πρέπει να λάβει υπόψη το πραγματικό πεδίο ορισμού  $\mathbf{V}_d$  της διάστασης  $d$ .

<sup>23</sup> Το αποτέλεσμα της ισότητας / ανισότητας περιβάλλοντος εξαρτάται από τα πεδία ορισμού των διαστάσεων: το  $\{\{\text{lang}, \{\text{gr}, \text{en}\}\}\}$  είναι ίσο-περιβάλλον με το  $\{\{\text{detail}, \{\text{low}, \text{high}\}\}\}$  αν  $\mathbf{V}_{\text{lang}} = \{\text{gr}, \text{en}\}$  και  $\mathbf{V}_{\text{detail}} = \{\text{low}, \text{high}\}$ , αλλά δεν είναι ίσο-περιβάλλον σε καμία άλλη περίπτωση. Ομοίως και για το (γνήσιο) υποσύνολο / υπερσύνολο περιβάλλοντος, το οποίο μπορεί να εκφραστεί με βάση την τομή περιβάλλοντος και την ισότητα / ανισότητας περιβάλλοντος, όπως δείξαμε στην Ενότητα 3.3.2.8.

Ομοίως, το (γνήσιο) υποσύνολο / υπερσύνολο περιβάλλοντος έχει ιδιότητες ανάλογες με αυτές του (γνήσιου) υποσυνόλου / υπερσυνόλου των συμβατικών συνόλων.

### 3.4.3 Τομή περιβάλλοντος, ένωση περιβάλλοντος, και διαφορά περιβάλλοντος

Όπως έχουμε δείξει, οι πράξεις αυτές αντιστοιχούν στις συμβατικές πράξεις της τομής, ένωσης, και διαφοράς συνόλων εφόσον δούμε τους προσδιοριστές περιβάλλοντος σαν τα σύνολα κόσμων που αυτοί αναπαριστούν. Εύκολα αποδεικνύεται ότι ιδιότητες των συμβατικών πράξεων συνόλων για την τομή, ένωση, και διαφορά ισχύουν και για τις αντίστοιχες πράξεις περιβάλλοντος. Σαν παράδειγμα, δίνουμε την απόδειξη της προσεταιριστικότητας για την τομή περιβάλλοντος. Σημειώστε ότι χρησιμοποιείται η ισότητα περιβάλλοντος αντί για την συμβατική ισότητα συνόλων.

#### Πρόταση 3.16

Η τομή περιβάλλοντος είναι προσεταιριστική:  $c_1 \cap^c (c_2 \cap^c c_3) =^c (c_1 \cap^c c_2) \cap^c c_3$

*Απόδειξη:* Και το αριστερό και το δεξί μέρος δίνουν προσδιοριστές περιβάλλοντος. Για να είναι αυτοί ίσα-περιβάλλοντα, θα πρέπει να αναπαριστούν το ίδιο σύνολο κόσμων σ.σ.μ. κάποιο **D**. Σημειώστε όμως ότι σύμφωνα με την Πρόταση 3.10,  $W_D[c_1 \cap^c (c_2 \cap^c c_3)] = W_D(c_1) \cap [W_D(c_2) \cap W_D(c_3)]$ . Τώρα λόγω της προσεταιριστικής ιδιότητας της συμβατικής τομής συνόλου, η τελευταία έκφραση είναι ίση με  $W_D[(c_1 \cap^c c_2) \cap^c c_3]$ .

Παρακάτω δίνεται ένας ενδεικτικός κατάλογος των βασικών ιδιοτήτων για την τομή περιβάλλοντος και για την ένωση περιβάλλοντος.

- Αυτοπαθής:

$$c \cap^c c =^c c$$

$$c \cup^c c =^c c$$

- Προσεταιριστική:

$$c_1 \cap^c (c_2 \cap^c c_3) =^c (c_1 \cap^c c_2) \cap^c c_3$$

$$c_1 \cup^c (c_2 \cup^c c_3) =^c (c_1 \cup^c c_2) \cup^c c_3$$

- Αντιμεταθετική:

$$c_1 \cap^c c_2 =^c c_2 \cap^c c_1$$

$$c_1 \cup^c c_2 =^c c_2 \cup^c c_1$$

- Διαλυτική:

$$c_1 \cap^c (c_2 \cup^c c_3) =^c (c_1 \cap^c c_2) \cup^c (c_1 \cap^c c_3)$$

$$c_1 \cup^c (c_2 \cap^c c_3) =^c (c_1 \cup^c c_2) \cap^c (c_1 \cup^c c_3)$$

Επιπλέον, εύκολα βλέπει κανείς ότι, όπως και η συμβατική τομή και ένωση, η τομή περιβάλλοντος και η ένωση περιβάλλοντος είναι πράξεις *μονότονες*: αν  $c_1 \subseteq^c c_1'$  και  $c_2 \subseteq^c c_2'$ , τότε  $(c_1 \cap^c c_2) \subseteq^c (c_1' \cap^c c_2')$ , και  $(c_1 \cup^c c_2) \subseteq^c (c_1' \cup^c c_2')$ .

Παρόμοια, η διαφορά περιβάλλοντος έχει ιδιότητες ανάλογες με τις ιδιότητες της συμβατικής διαφοράς συνόλων.

### 3.4.4 Εκφράσεις περιβάλλοντος και συνθήκες περιβάλλοντος

Η **έκφραση περιβάλλοντος** είναι μια έκφραση που σχηματίζεται από προσδιοριστές περιβάλλοντος, που τους συνδέουν οι πράξεις της τομής περιβάλλοντος, της ένωσης περιβάλλοντος, ή της διαφοράς περιβάλλοντος, με παρενθέσεις που ορίζουν την σειρά εκτέλεσης των πράξεων περιβάλλοντος.

Η **συνθήκη περιβάλλοντος** είναι μια συνθήκη που σχηματίζεται από εκφράσεις περιβάλλοντος που συγκρίνονται μεταξύ τους μέσω της ισότητας / ανισότητας περιβάλλοντος, ή του (γνήσιου) υποσυνόλου / υπερσυνόλου περιβάλλοντος, που συνδυάζονται με «and» ή «or», και που μπορεί να προηγείται το «not», όπου οι παρενθέσεις ορίζουν την σειρά εκτέλεσης.

Η αντικατάσταση ενός προσδιοριστή περιβάλλοντος με ένα άλλο περιβάλλον που είναι ίσο-περιβάλλον δίνει μια ισοδύναμη έκφραση περιβάλλοντος (μια έκφραση που αναπαριστά το ίδιο σύνολο κόσμων όπως η αρχική), ή συνθήκη περιβάλλοντος. Η επεκτεταμένη μορφή περιβάλλοντος και οι απλοποιημένες μορφές (όπως συζητήθηκαν στην Ενότητα 3.3.2.4) ενός προσδιοριστή περιβάλλοντος  $c$  είναι ίσα-περιβάλλοντα με το  $c$ , και μπορούν να αντικαταστήσουν το  $c$  σε εκφράσεις περιβάλλοντος και σε συνθήκες περιβάλλοντος.

## 3.5 ΠΟΛΥΠΛΟΚΟΤΗΤΑ ΤΩΝ ΠΡΑΞΕΩΝ ΠΕΡΙΒΑΛΛΟΝΤΟΣ

Στην ενότητα αυτή συζητούμε την πολυπλοκότητα ως προς τον χρόνο των πράξεων περιβάλλοντος, με βάση τον αριθμό των συγκρίσεων που πραγματοποιούνται. Θεωρούμε ότι το μέγεθος του προβλήματος αντιπροσωπεύεται από τον αριθμό των όρων  $k$  των προσδιοριστών περιβάλλοντος. Ο αριθμός αυτός μπορεί να είναι μεγαλύτερος από τον αριθμό των κόσμων, μια και επιτρέπονται όροι που επικαλύπτονται. Επιπλέον, πράξεις όπως η τομή περιβάλλοντος, η ένωση περιβάλλοντος, και η διαφορά περιβάλλοντος αυξάνουν τον αριθμό των όρων που περιέχονται στους προσδιοριστές περιβάλλοντος<sup>24</sup>.

Για τα ακόλουθα θεωρούμε ένα σύνολο διαστάσεων  $\mathbf{D}$  με αριθμό στοιχείων  $n$ , και θεωρούμε ότι ο μέσος αριθμός των στοιχείων των πεδίων ορισμού των διαστάσεων είναι  $m$ . Τότε, ο αριθμός των κόσμων που ορίζονται για το  $\mathbf{D}$  είναι  $m^n$ . Θεωρούμε επίσης ότι οι προσδιοριστές περιβάλλοντος είναι διατεταγμένοι με τον εξής τρόπο: (a) οι τιμές των διαστάσεων στους προσδιοριστές διάστασης είναι διατεταγμένες, σύμφωνα με την αριθμητική τους τιμή ή λεξικογραφικά, και (b) οι προσδιοριστές διάστασης είναι διατεταγμένοι μέσα στους όρους σύμφωνα με την λεξικογραφική σειρά των ονομάτων των διαστάσεων.

*Τομή περιβάλλοντος,  $c_1 \cap^c c_2$ :* Θεωρούμε ότι ο μέσος αριθμός των προσδιοριστών διάστασης  $(d, V)$  στους όρους του  $c_1$  και του  $c_2$  είναι  $p$  ( $p \leq n$ ), και ότι ο μέσος αριθμός των τιμών των διαστάσεων στο  $V$  είναι  $v$  ( $v \leq m$ ). Η τομή όρου μεταξύ δυο όρων θα συγκρίνει πρώτα τους προσδιοριστές διάστασης και αν η διάσταση  $d$  είναι η ίδια, θα συγκριθούν τα σύνολα των τιμών της διάστασης. Συνεπώς, θα εκτελεστούν  $p+p*v = p*(v+1)$  συγκρίσεις στην χειρότερη περίπτωση. Αν το  $c_1$  περιέχει  $k$  όρους και το  $c_2$  περιέχει  $l$  όρους, τότε ο συνολικός αριθμός συγκρίσεων για την χειρότερη περίπτωση εκτιμάται σε  $k * l * p*(v+1)$ . Πάντως, το  $p$  και το  $v$  είναι άνω φραγμένα, μια και στα πλαίσια ενός δεδομένου προβλήματος το σύνολο των διαστάσεων και τα αντίστοιχα πεδία ορισμού είναι πεπερασμένα. Αν

<sup>24</sup> Στην Ενότητα 3.3.2.4, δείξαμε πως ο αριθμός των όρων μπορεί να μειωθεί απλοποιώντας τους προσδιοριστές περιβάλλοντος.

θεωρήσουμε ότι το  $k$  είναι ο μέσος αριθμός όρων στα  $c_1$  και  $c_2$ , τότε η τάξη μεγέθους της χρονικής πολυπλοκότητας για την χειρότερη περίπτωση είναι  $O(k^2)$ .

*Ένωση περιβάλλοντος*,  $c_1 \cup^c c_2$ : Η ένωση περιβάλλοντος δεν εκτελεί συγκρίσεις, και η τάξη μεγέθους της χρονικής πολυπλοκότητας είναι  $O(1)$ .

*Ανάπτυξη επεκτεταμένου περιβάλλοντος*,  $\otimes_D c$ : Αν το  $c$  περιέχει  $k$  όρους τότε η επέκταση θα πραγματοποιήσει  $n*k$  συγκρίσεις στην χειρότερη περίπτωση. Το στάδιο της ανάπτυξης δεν θα πραγματοποιήσει καθόλου συγκρίσεις, συνεπώς η τάξη μεγέθους της χρονικής πολυπλοκότητας για την χειρότερη περίπτωση είναι  $O(k)$ .

Η απαλοιφή διπλότυπων κόσμων θα αυξήσει τον αριθμό των συγκρίσεων. Ειδικότερα, η απαλοιφή διπλότυπων κόσμων απαιτεί το πολύ  $n*w$  συγκρίσεις για κάθε ζεύγος όρων στο  $c$ , όπου  $w$  είναι ο μέσος αριθμός κόσμων που αναπαριστούν οι όροι του  $c$ , με  $w \leq m^n$ . Ο συνολικός αριθμός συγκρίσεων για την ανάπτυξη επεκτεταμένου περιβάλλοντος με απαλοιφή διπλότυπων θα είναι για την χειρότερη περίπτωση  $n*k+n*w*(1+2+\dots+k-1)$ , συνεπώς<sup>25</sup> η τάξη μεγέθους της χρονικής πολυπλοκότητας είναι  $O(k+(k-1)^2)$ , ή απλούστερα  $O(k^2)$ .

Ο παράγοντας  $(k-1)^2$  που αντιστοιχεί στο επιπλέον κόστος της απαλοιφής διπλότυπων κόσμων μπορεί να αποφευχθεί, αν το  $c$  περιέχει όρους που είναι αμοιβαία αποκλειόμενοι. Η τομή περιβάλλοντος διατηρεί την ιδιότητα αυτή, με άλλα λόγια αν τα  $c_1$  και  $c_2$  περιέχουν μόνον αμοιβαία αποκλειόμενους όρους, τότε το  $c = c_1 \cap^c c_2$  περιέχει επίσης μόνον αμοιβαία αποκλειόμενους όρους. Η διατήρηση των περιβαλλόντων σε μια τέτοια μορφή απαιτεί τον κατάλληλο μετασχηματισμό του  $c$ , εφόσον το  $c$  είναι το αποτέλεσμα μιας έκφρασης περιβάλλοντος που ενέχει ενώσεις περιβάλλοντος και / ή διαφορές περιβάλλοντος.

*Ισότητα περιβάλλοντος*,  $c_1 =^c c_2$ : Θεωρούμε ότι ο ελάχιστος αριθμός κόσμων που αναπαριστά το  $c_1$  και το  $c_2$  είναι  $w$ ,  $w \leq m^n$ . Τότε, οι συγκρίσεις στην χειρότερη περίπτωση είναι  $n*w$ , συν τις συγκρίσεις που χρειάζονται για την ανάπτυξη επεκτεταμένου περιβάλλοντος για τα  $c_1$  και  $c_2$ . Συνεπώς, η τάξη μεγέθους της χρονικής πολυπλοκότητας για την χειρότερη περίπτωση είναι η ίδια όπως στην ανάπτυξη επεκτεταμένου περιβάλλοντος:  $O(k^2)$  αν τα  $c_1, c_2$  περιέχουν επικαλυπτόμενους όρους, όπου  $k$  είναι ο μέσος αριθμός όρων στα  $c_1$  και  $c_2$ . Αν οι όροι σε κάθε ένα από τα  $c_1, c_2$  είναι αμοιβαία αποκλειόμενοι, η τάξη μεγέθους της χρονικής πολυπλοκότητας για την ισότητα περιβάλλοντος και για την χειρότερη περίπτωση είναι  $O(k)$ .

*Υποσύνολο περιβάλλοντος*,  $c_1 \subseteq^c c_2$ : Όπως η ισότητα περιβάλλοντος.

Η παραπάνω συζήτηση μας οδηγεί να θεωρήσουμε προσδιοριστές περιβάλλοντος οι οποίοι περιέχουν όρους που είναι αμοιβαία αποκλειόμενοι. Στην περίπτωση αυτή, ένα περιβάλλον δεν θα περιέχει παραπάνω όρους από τον αριθμό των πιθανών κόσμων. Αν  $w = m^n$  είναι ο συνολικός αριθμός των πιθανών κόσμων, και θεωρώντας προσδιοριστές περιβάλλοντος που αποτελούνται από αμοιβαία αποκλειόμενους όρους, η τάξη μεγέθους της χρονικής πολυπλοκότητας για την χειρότερη περίπτωση είναι ως εξής.

*Τομή περιβάλλοντος*: παραμένει στο  $O(w^2)$ .

*Ένωση περιβάλλοντος*:  $O(w^2)$ .

*Ανάπτυξη επεκτεταμένου περιβάλλοντος*, *ισότητα περιβάλλοντος* και *υποσύνολο περιβάλλοντος*:  $O(w)$ .

Παρατηρείστε ότι το κόστος της διατήρησης των προσδιοριστών περιβάλλοντος σε μια κατάσταση όπου οι όροι είναι αμοιβαία αποκλειόμενοι βαραίνει την ένωση περιβάλλοντος

<sup>25</sup> Με χρήση του  $1+2+\dots+k = O(k^2)$ .

και την διαφορά περιβάλλοντος, οι οποίες πρέπει να συγκρίνουν και να επαναδιευθετούν όρους.

Όπως αναφέρθηκε στην Ενότητα 3.4.1, η αποδοτικότητα της ισότητας / ανισότητας περιβάλλοντος και του (γνήσιου) υποσυνόλου / υπερσυνόλου περιβάλλοντος μπορεί να βελτιωθεί αν θεωρήσουμε σαν σύνολο διαστάσεων το  $\mathbf{D}_{\min}$ , όπου το  $\mathbf{D}_{\min}$  περιέχει όλες τις διαστάσεις που απαντώνται στα περιβάλλοντα τα οποία εμπλέκονται στην πράξη και καμία άλλη διάσταση.

### 3.6 ΣΗΜΕΙΟΓΡΑΦΙΑ ΓΙΑ ΠΡΟΣΔΙΟΡΙΣΤΕΣ ΠΕΡΙΒΑΛΛΟΝΤΟΣ

Σε προηγούμενες ενότητες ορίσαμε αυστηρά τους προσδιοριστές περιβάλλοντος και τις πράξεις περιβάλλοντος. Στην ενότητα αυτή εισάγουμε ένα απλούστερο και πιο βολικό συντακτικό για προσδιοριστές περιβάλλοντος και πράξεις περιβάλλοντος.

#### Παράδειγμα 3.14

Τα παρακάτω είναι παραδείγματα προσδιοριστών περιβάλλοντος:

1. [time=07:45]
2. [lang=greek, detail in {low,medium}]
3. [lang not in {greek,spanish}, detail!=high]
4. [season=summer | season in {fall,spring}, daytime=noon]
5. [time in {08:00..13:30,17:00..20:30}]
6. [format=pdf | | - | lang!=spanish]
7. []
8. [-]



Στο Παράδειγμα 3.14, ο προσδιοριστής περιβάλλοντος (1) αποτελείται από έναν όρο που περιέχει μόνον ένα προσδιοριστή διάστασης, και ο αυστηρός του τύπος είναι  $\{(time, \{07:45\})\}$ . Ο προσδιοριστής περιβάλλοντος (2) αποτελείται από έναν όρο που περιέχει δυο προσδιοριστές διάστασης, έναν για την διάσταση lang και έναν για την διάσταση detail. Ο προσδιοριστής περιβάλλοντος (3) χρησιμοποιεί το «not in» και το «!=» (διάφορο), τα οποία δεν έχουν κάποιο αντίστοιχο στον αυστηρό ορισμό των προσδιοριστών διάστασης. Όμως, αυτά παρουσιάζονται μόνο και μόνο σαν συντακτική συντομογραφία, επειδή θεωρούμε ότι τα πεδία ορισμού των διαστάσεων είναι γνωστά την στιγμή που χρησιμοποιούνται τα περιβάλλοντα. Επιπλέον, όπως αναφέρεται στην Ενότητα 3.2.1, θεωρούμε ότι τα πεδία ορισμού των διαστάσεων είναι πεπερασμένα, και ότι μπορούν να περιγραφούν με απαρίθμηση των στοιχείων τους. Είναι, συνεπώς, πάντοτε δυνατό να βρεθεί το συμπλήρωμα συνόλου που υπονοείται από τους τελεστές στο (3), και να εκφραστούν τα περιβάλλοντα με τρόπο που να είναι άμεσα αντίστοιχος με τους αυστηρούς ορισμούς. Ο προσδιοριστής περιβάλλοντος (4) περιέχει δυο όρους, και αναπαριστά τους κόσμους που είτε είναι καλοκαίρι είτε είναι μεσημέρι άνοιξης / φθινοπώρου. Ο προσδιοριστής περιβάλλοντος (5) παρουσιάζει μια συντομογραφία για την αναπαράσταση διαστημάτων πάνω σε κάποιο φραγμένο, διακριτό, και πλήρως διατεταγμένο πεδίο ορισμού. Ο προσδιοριστής περιβάλλοντος (6) αποτελείται από τέσσερις όρους· ο δεύτερος όρος δεν περιέχει τίποτε και αντιστοιχεί στον καθολικό όρο  $\emptyset$ , ενώ ο τρίτος όρος είναι το σύμβολο « $\rightarrow$ » και αντιστοιχεί στον κενό όρο  $\{-\}$ . Τέλος, το αυστηρό αντίστοιχο του προσδιοριστή περιβάλλοντος (7) είναι το  $\{\emptyset\}$ , το καθολικό περιβάλλον για οποιοδήποτε σύνολο διαστάσεων, και το αυστηρό αντίστοιχο του προσδιοριστή περιβάλλοντος (8) είναι το  $\{-\}$ , το κενό περιβάλλον για οποιοδήποτε σύνολο διαστάσεων.

Την γραμματική για την σύνταξη των προσδιοριστών περιβάλλοντος την δίνει ο παρακάτω Πίνακας 3.1 σε Extended Backus-Naur Form [EBNF], ή EBNF για συντομία. Τα σύμβολα που μπορούν να οριστούν μέσω κάποιας ομαλής έκφρασης (regular expression) αρχίζουν με κεφάλαιο γράμμα (παράδειγμα: DimValue), ενώ τα σύμβολα που ορίζονται σε EBNF αρχίζουν με μικρό γράμμα (παράδειγμα: cxtSpec).

**Πίνακας 3.1: Σύνταξη των προσδιοριστών περιβάλλοντος (context specifiers).**

```

cxtSpec ::= "[" cxtSpecClause ("|" cxtSpecClause)* "]"
cxtSpecClause ::= "" | "-" | dimList
dimList ::= dimSpec ("," dimSpec)*
dimSpec ::= dimName (atomicOp DimValue |
                    setOp ("{" setDimValue? "}" | "ALL"))
atomicOp ::= "=" | "!="
setOp ::= "in" | "not in"
setDimValue ::= dimSpan ("," dimSpan)*
dimSpan ::= DimValue | DimValue ".." DimValue

```

Όπως θα περίμενε κανείς, ένας προσδιοριστής περιβάλλοντος επιτρέπεται να περιέχει κενά σύνολα, όπως στο [lang=english, detail in {}] και στο [lang=english, detail not in {}]. Στην τελευταία περίπτωση, υπονοείται ότι το detail κυμαίνεται σε ολόκληρο το εύρος του πεδίου ορισμού του. Επιπρόσθετα, η λέξη-κλειδί ALL είναι συντομογραφία για ολόκληρο το πεδίο ορισμού μιας διάστασης, όπως για παράδειγμα στο [time in ALL]<sup>26</sup>. Για άλλη μια φορά, οι προσδιοριστές περιβάλλοντος που περιέχουν «ALL», «not in», or «!=» αντικαθίστανται κάποια στιγμή από προσδιοριστές περιβάλλοντος που είναι ίσα-περιβάλλοντα και που δεν περιέχουν τους τελεστές αυτούς. Συνεπώς, σε τέτοιες περιπτώσεις τα πεδία ορισμού των διαστάσεων πρέπει να λαμβάνονται υπόψη προτού εκτελεστεί κάποια πράξη<sup>27</sup>.

Την σημειογραφία για τις πράξεις περιβάλλοντος την δίνει ο ακόλουθος Πίνακας 3.2.

**Πίνακας 3.2: Σημειογραφία για τις πράξεις περιβάλλοντος.**

Operation	Symbol	Notation
Context equality	$=^c$	=
Context inequality	$\neq^c$	!=
Context subset, superset	$\subseteq^c, \supseteq^c$	<= , >=
Proper context subset, superset	$\subset^c, \supset^c$	< , >
Context intersection	$\cap^c$	*
Context union	$\cup^c$	+
Context difference	$-^c$	-

Χρησιμοποιώντας την παραπάνω σημειογραφία, είναι δυνατόν να σχηματίσουμε εκφράσεις περιβάλλοντος και συνθήκες περιβάλλοντος σαν τις ακόλουθες:

<sup>26</sup> Προφανώς, ένας προσδιοριστής διάστασης της μορφής “d in ALL” μπορεί να παραλειφθεί εντελώς από έναν όρο. Παρόλα αυτά, η λέξη-κλειδί ALL είναι χρήσιμη, όπως θα διαπιστώσουμε στα ακόλουθα κεφάλαια όπου συζητούμε τα **πρότυπα περιβάλλοντος**.

<sup>27</sup> Αυτό αφορά και την τομή περιβάλλοντος και την ένωση περιβάλλοντος, οι οποίες σε άλλη περίπτωση δεν εξαρτώνται από τα πεδία ορισμού των διαστάσεων.



- [lang=greek, detail in {low,medium}] <= [lang=greek]
- [format=pdf | detail!=high] \* [lang=english, format=pdf] = [lang=english, format=pdf]

### 3.7 ΠΕΡΙΛΗΨΗ

Αντίθετα με τις παραδοσιακές βάσεις δεδομένων, οι πληροφορίες στον Ιστό συνδέονται συχνά με κάποιο υπονοούμενο *ερμηνευτικό περιβάλλον*. Στο κεφάλαιο αυτό αναπτύξαμε μια τυποποίηση για να αναπαριστούμε ερμηνευτικά περιβάλλοντα με έναν ευέλικτο και απλό τρόπο, χρησιμοποιώντας μεταβλητές που ονομάζονται **διαστάσεις**. Οι **προσδιοριστές περιβάλλοντος** ορίζουν ερμηνευτικά περιβάλλοντα περιορίζοντας τις δυνατές τιμές των διαστάσεων, και συνδυάζοντας τέτοιους περιορισμούς σε διαζεύξεις και συζεύξεις.

Δώσαμε νόημα στο ερμηνευτικό περιβάλλον δείχνοντας ότι μπορεί να ιδωθεί σαν ένα σύνολο πιθανών **κόσμων**, όπου ο κόσμος είναι μια εκδοχή της πραγματικότητας κάτω από την οποία οι πληροφορίες αποκτούν μονοσήμαντη ερμηνεία. Ορίσαμε πράξεις περιβάλλοντος που συνδυάζουν και συγκρίνουν ερμηνευτικά περιβάλλοντα, και αποδείξαμε ότι οι πράξεις αυτές τηρούν μια αναλογία με αντίστοιχες πράξεις σε συμβατικά σύνολα πιθανών κόσμων.

Επιπλέον, διερευνήσαμε τις ιδιότητες των πράξεων περιβάλλοντος, και μιλήσαμε για την πολυπλοκότητά τους ως προς τον χρόνο. Ένα σημαντικό πλεονέκτημα της προσέγγισής μας είναι ότι οι περισσότερες πράξεις περιβάλλοντος μπορούν να εκτελεστούν ακόμη και αν δεν έχουμε πλήρη γνώση των διαστάσεων που εμπλέκονται· στην περίπτωση αυτή η αντιστοίχιση ερμηνευτικών περιβαλλόντων και συνόλων κόσμων είναι δυνατόν να αναβληθεί μέχρι να έχουμε πλήρη γνώση των διαστάσεων. Τέλος, προτείναμε μια βολική σημειογραφία για να εκφράζουμε ερμηνευτικά περιβάλλοντα και πράξεις περιβάλλοντος.



## 4 ΕΝΑ ΜΟΝΤΕΛΟ ΔΕΔΟΜΕΝΩΝ ΓΙΑ MSSD

Στο προηγούμενο κεφάλαιο εισήγαμε τους προσδιοριστές περιβάλλοντος σαν έναν τρόπο αναπαράστασης του ερμηνευτικού περιβάλλοντος με την χρήση πολλαπλών διαστάσεων, εξηγήσαμε την σημασία τους μέσω της αντιστοίχισης σε κόσμους, και ορίσαμε έναν αριθμό πράξεων περιβάλλοντος που χρησιμοποιούνται στον σχηματισμό εκφράσεων περιβάλλοντος και συνθηκών περιβάλλοντος. Οι προσδιοριστές περιβάλλοντος είναι η διαφορά-κλειδί ανάμεσα στα συμβατικά δεδομένα και στα δεδομένα που εξαρτώνται από το ερμηνευτικό περιβάλλον και παρουσιάζουν πολλαπλές εκφάνσεις, και που λέγονται επίσης στην προσέγγισή μας **πολυδιάστατα δεδομένα**. Έτσι, οι προσδιοριστές περιβάλλοντος παίζουν κεντρικό ρόλο στα μοντέλα δεδομένων και στις γλώσσες επερωτήσεων για πολυδιάστατα δεδομένα.

Στο κεφάλαιο αυτό προτείνουμε ένα μοντέλο δεδομένων για πολυδιάστατα ημιδομημένα δεδομένα (MSSD), που λέγεται **Πολυδιάστατο Μοντέλο Ανταλλαγής Αντικειμένων** (Multidimensional Object Exchange Model), ή αλλιώς **Πολυδιάστατο OEM**, ή **MOEM** για συντομία. Το MOEM είναι μια υποπερίπτωση ενός μοντέλου δεδομένων βασισμένου σε γράφους που καλείται **Πολυδιάστατος Γράφος Δεδομένων**, ο οποίος ενσωματώνει προσδιοριστές περιβάλλοντος και χρησιμοποιεί ειδικού τύπου κόμβους και ακμές για να αναπαραστήσει **πολυδιάστατες οντότητες**. Μένοντας πιστοί στην προσέγγιση των ημιδομημένων δεδομένων σύμφωνα με την οποία το σχήμα υπάρχει σαν μέρος των δεδομένων, εμφυτεύουμε πληροφορία για το ερμηνευτικό περιβάλλον στο ίδιο το μοντέλο γράφου. Το MOEM και ο Πολυδιάστατος Γράφος Δεδομένων επεκτείνουν το *Μοντέλο Ανταλλαγής Αντικειμένων* (Object Exchange Model) [PGW95, AQM+97, Suc98], ή *OEM*, ένα από τα κυριότερα μοντέλα γράφου για ημιδομημένα δεδομένα, που σχεδιάστηκε αρχικά στο Πανεπιστήμιο Stanford στα πλαίσια του έργου TSIMMIS [PGW95, GPQ+97, CGH+94, GHI+95, PGU96]. Όπως θα δείξουμε, το MOEM συνενώνει έναν αριθμό από διαφορετικά OEMs που υφίστανται κάτω από διαφορετικούς κόσμους.

Η ανάγκη για μια επέκταση του OEM αναγνωρίζεται επίσης και στο [CAW99], που ασχολείται με το πρόβλημα της αναπαράστασης ιστορικών αλλαγών σε βάσεις δεδομένων OEM. Στο Κεφάλαιο 6 συζητούμε για το [CAW99] περαιτέρω, και διερευνούμε ενδελεχώς το προαναφερθέν πρόβλημα. Ένα μοντέλο για ημιδομημένα δεδομένα που αποκλίνει από το OEM έχει προταθεί στο [BDT98], όπου οι ετικέτες των ακμών είναι και οι ίδιες κομμάτια ημιδομημένων δεδομένων. Το μοντέλο που προτείνουμε αντιμετωπίζει τις ετικέτες σαν μεταπληροφορία μάλλον παρά σαν περιέχουσες δεδομένα. Στο [DBJ99], προτείνεται ένα επεκτάσιμο μοντέλο ημιδομημένων δεδομένων που χρησιμοποιεί σύνολα ιδιοτήτων της μορφής `όνομα_ιδιότητας: τιμή_ιδιότητας` σαν ετικέτες ακμών. Επισυνάπτοντας ιδιότητες κατά βούληση, ο γράφος γίνεται πλούσιος σε μεταπληροφορία. Διαφορετικές ιδιότητες μπορεί να έχουν διαφορετική σημασία, πράγμα που οδηγεί σε ένα μοντέλο αυξημένης γενικότητας, από την άλλη όμως περιπλέκει τον σχηματισμό επερωτήσεων. Ο στόχος της προσέγγισής μας είναι η αναπαράσταση πληροφορίας που παρουσιάζει διαφορετικές εκφάνσεις. Αυτό οδηγεί σε κοινή σημασία για την μεταπληροφορία, που χρησιμοποιείται μόνο για την ενσωμάτωση του ερμηνευτικού περιβάλλοντος. Επιπρόσθετα, το μοντέλο μας διατηρεί τις ακμές με ετικέτες του OEM και επισυνάπτει την

μεταπληροφορία για το ερμηνευτικό περιβάλλον σε έναν νέο τύπο ακμών· μοντέλα γράφου όπως το OEM γίνονται ειδική περίπτωση του μοντέλου που προτείνουμε.

Στα ακόλουθα, ορίζουμε την έννοια της πολυδιάστατης οντότητας, και εξηγούμε πως οι προσδιοριστές περιβάλλοντος χρησιμοποιούνται στα πλαίσια των πολυδιάστατων οντοτήτων. Έπειτα συνεχίζουμε με τον Πολυδιάστατο Γράφο Δεδομένων, και διερευνούμε τις ιδιότητές του: μετάδοση περιβάλλοντος, αναγωγή, αιτιοκρατία ως προς το περιβάλλον, και κανονική μορφή. Με βάση τις ιδιότητες αυτές, ορίζουμε το Πολυδιάστατο OEM σαν ειδική περίπτωση του Πολυδιάστατου Γράφου Δεδομένων. Τέλος, εισάγουμε τις εκφράσεις «mssd-expressions», που προσφέρουν έναν τρόπο αναπαράστασης Πολυδιάστατων Γράφων Δεδομένων και MOEMs σε μορφή κειμένου.

## 4.1 ΠΟΛΥΔΙΑΣΤΑΤΕΣ ΟΝΤΟΤΗΤΕΣ

Στα πολυδιάστατα δεδομένα, το ερμηνευτικό περιβάλλον χρησιμοποιείται για να χαρακτηρίσει **εκφάνσεις** της ίδιας πληροφοριακής οντότητας, δηλώνοντας τις συνθήκες κάτω από τις οποίες υφίσταται κάθε έκφανση και, ταυτόχρονα, θέτοντας το πλαίσιο για την ερμηνεία της πληροφορίας με μονοσήμαντο τρόπο. Μια πληροφοριακή οντότητα που αποτελείται από έναν αριθμό εκφάνσεων, όπου κάθε μια υφίσταται κάτω από έναν αριθμό κόσμων, καλείται **πολυδιάστατη οντότητα**.

### Ορισμός 4.1

*Μια πολυδιάστατη οντότητα είναι μια πληροφοριακή οντότητα που περιλαμβάνει μηδέν ή περισσότερες εκφάνσεις μαζί με τα αντίστοιχα περιβάλλοντα που καθορίζουν τους κόσμους κάτω από τους οποίους δύναται να υφίσταται κάθε έκφανση. Οι εκφάνσεις μιας πολυδιάστατης οντότητας μπορούν οι ίδιες να είναι πολυδιάστατες οντότητες, ή συμβατικές<sup>28</sup> πληροφοριακές οντότητες, ή οποιοσδήποτε συνδυασμός των δυο.*

Οι πολυδιάστατες οντότητες είναι ο ακρογωνιαίος λίθος των πολυδιάστατων δεδομένων, μια και (a) ενσωματώνουν προσδιοριστές περιβάλλοντος, που μπορούν να εμφανίζονται μόνον μέσα σε πολυδιάστατες οντότητες, και (b) συγκρατούν μαζί πληροφοριακές οντότητες που αποτελούν εκφάνσεις της ίδιας αφηρημένης οντότητας. Μια πολυδιάστατη οντότητα μπορεί να έχει οποιονδήποτε αριθμό εκφάνσεων (καμία έκφανση στην οριακή περίπτωση), και οι εκφάνσεις που είναι συμβατικές πληροφοριακές οντότητες μπορούν να έχουν διαφορετικές τιμές<sup>29</sup>. Σημειώστε ότι οι συμβατικές πληροφοριακές οντότητες μπορούν να θεωρηθούν σαν ειδική περίπτωση πολυδιάστατων οντοτήτων, όπου υπάρχει μόνον μια έκφανση η οποία υφίσταται κάτω από κάθε πιθανό κόσμο (το περιβάλλον της έκφανσης είναι ένα καθολικό περιβάλλον  $\mathcal{W}$ ).

Δεν υπάρχουν περιορισμοί στο τι μπορεί να αποτελεί το αντίστοιχο περιβάλλον μιας έκφανσης: εκφάνσεις μιας πολυδιάστατης οντότητας μπορούν να χαρακτηρίζονται από περιβάλλοντα που επικαλύπτονται (ορίζουν κοινούς κόσμους), ή ακόμα και από κενά περιβάλλοντα στην οριακή περίπτωση. Έτσι, μια πολυδιάστατη οντότητα δεν είναι υποχρεωμένη να καλύπτει κάθε πιθανό κόσμο με τις εκφάνσεις τις: δεδομένου ενός κόσμου  $w$ , καμία, μια, ή περισσότερες εκφάνσεις μιας πολυδιάστατης οντότητας μπορούν να υφίστανται κάτω από το  $w$ .

<sup>28</sup> Με το «συμβατικός» εννοούμε μη εξαρτώμενος από το ερμηνευτικό περιβάλλον.

<sup>29</sup> Όπως θα δούμε στην συνέχεια της ενότητας, στην περίπτωση των πολυδιάστατων ημιδομημένων δεδομένων, οι εκφάνσεις μιας πολυδιάστατης οντότητας μπορεί να έχουν και διαφορετικούς τύπους (δομή).

Δεδομένου ενός συνόλου κόσμων  $S_w$ , υπάρχουν δυο κύριοι τρόποι που μπορούμε να επιλέξουμε εκφάνσεις μιας πολυδιάστατης οντότητας<sup>30</sup>. Ο πρώτος είναι να επιλέξουμε μια έκφανση μόνον εφόσον υφίσταται κάτω από κάθε κόσμο στο  $S_w$ , συνεπώς να υποθέσουμε ότι υπονοείται μια σύζευξη για τους κόσμους στο  $S_w$ . Ο δεύτερος είναι να θεωρήσουμε μια διάζευξη, και να επιλέξουμε μια έκφανση εφόσον υφίσταται κάτω από τουλάχιστον έναν κόσμο που ανήκει στο  $S_w$ . Εάν μια έκφανση υφίσταται κάτω από κάθε κόσμο που αναπαριστά ένα περιβάλλον  $c$ , τότε λέμε ότι η έκφανση αυτή **υφίσταται κάτω από** το  $c$ .

#### Ορισμός 4.2

*Ας είναι το  $c$  ένας προσδιοριστής περιβάλλοντος, και ας είναι το  $e$  μια πολυδιάστατη οντότητα. Ας είναι το  $f_i$  μια έκφανση της  $e$ , και ας είναι το  $c_i$  ένα περιβάλλον που προσδιορίζει τους κόσμους κάτω από τους οποίους υφίσταται η  $f_i$ . Τότε λέμε ότι η  $f_i$  **υφίσταται κάτω από** το  $c$ , εάν και μόνον  $c \subseteq c_i$ .*

Δεδομένων ενός περιβάλλοντος  $c$  και μιας πολυδιάστατης οντότητας  $e$ , αν  $f_k, f_1, \dots, f_m$  είναι οι εκφάνσεις της  $e$  που υφίστανται κάτω από το  $c$ , τότε λέμε ότι η  $e$  **αποτιμάται σε**  $f_k, f_1, \dots, f_m$  **κάτω από** το  $c$ . Δεδομένου ενός περιβάλλοντος, μια πολυδιάστατη οντότητα μπορεί να αποτιμάται σε καμία, μια, ή περισσότερες εκφάνσεις. Σημειώστε ότι στην οριακή περίπτωση που η πολυδιάστατη οντότητα δεν έχει καθόλου εκφάνσεις, δεν αποτιμάται ποτέ σε κάποια έκφανση.

Όπως αναφέρθηκε ήδη, οι εκφάνσεις μιας πολυδιάστατης οντότητας μπορούν να έχουν διαφορετικές τιμές. Συγκεκριμένα στα MSSD, οι εκφάνσεις μιας πολυδιάστατης οντότητας μπορούν να έχουν επίσης και διαφορετικούς τύπους (με άλλα λόγια, διαφορετική δομή). Αυτό είναι σύμφωνο με το χαλαρό σύστημα τύπων των (συμβατικών) ημιδομημένων δεδομένων, όπου στιγμιότυπα της ίδιας κλάσης αντικειμένων είναι δυνατόν να παρουσιάζουν διαφοροποιημένη δομή.

## 4.2 ΠΟΛΥΔΙΑΣΤΑΤΟΙ ΓΡΑΦΟΙ ΔΕΔΟΜΕΝΩΝ

Στην ενότητα αυτή παρουσιάζουμε τον *Πολυδιάστατο Γράφο Δεδομένων*, ένα μοντέλο δεδομένων που επεκτείνει το OEM ενσωματώνοντας προσδιοριστές περιβάλλοντος. Παρόλο που το OEM και παρόμοια μοντέλα δεδομένων του Ιστού που βασίζονται σε γράφους είναι κατά κύριο λόγο ικανά να αναπαραστήσουν πολυδιάστατες οντότητες, δεν αποτελούν ικανοποιητική λύση για μια σειρά λόγων.

- (a) Κρυφή σημασιολογία: επειδή δεν αντιμετωπίζουν ευθέως το ζήτημα των πολλαπλών εκφάνσεων, είναι ευθύνη της εφαρμογής να ορίσει την σημασιολογία με έναν ad-hoc τρόπο.
- (b) Άκομμη σημειογραφία: η αναπαράσταση των πολλαπλών εκφάνσεων μιας οντότητας δεν είναι δυνατόν να γίνει στα μοντέλα αυτά με κομψό τρόπο.
- (c) Επανάληψη της πληροφορίας: μια ad-hoc προσέγγιση είναι πιθανόν να οδηγήσει σε επανάληψη της πληροφορίας που είναι κοινή, πράγμα μη επιθυμητό.

Οι Πολυδιάστατοι Γράφοι Δεδομένων αποφεύγουν τα προβλήματα αυτά *αντιμετωπίζοντας τις πολυδιάστατες οντότητες σαν πολίτες πρώτης κατηγορίας*. Χρησιμοποιούνται ειδικοί κόμβοι και ακμές για να μοντελοποιήσουν πολυδιάστατες οντότητες και για να τις διαφοροποιήσουν από τις συμβατικές. Επιπλέον, η σημειογραφία για

<sup>30</sup> Οι τρόποι αυτοί χρησιμοποιούνται στην Ενότητα 4.4 για τον ορισμό της **αναγωγής σε OEM** και της **μερικής αναγωγής**, αντίστοιχα.

το περιβάλλον που εισήγαμε στο προηγούμενο κεφάλαιο χρησιμοποιείται για να χαρακτηριστούν οι εκφάνσεις των πολυδιάστατων οντοτήτων.

Όπως είδαμε στο Κεφάλαιο 2, το OEM είναι ένας *κατευθυνόμενος πολυγράφος με ρίζα που έχει ετικέτες και φύλλα με τιμές*, αρκετά ευέλικτος για να ανέχεται την έλλειψη κανονικότητας στην δομή των ημιδομημένων δεδομένων. Έχουν προταθεί διάφορες εκδόσεις του OEM που επισυνάπτουν ετικέτες είτε στους κόμβους [PGW95, GPQ+97] είτε στις ακμές [MAG+97, AQM+97, FFLS97, BFS00]. Η ερώτηση εάν οι ετικέτες θα πρέπει να επισυνάπτονται στους κόμβους ή στις ακμές επανέρχεται συχνά στα μοντέλα γράφου για ημιδομημένα δεδομένα και XML, και κάθε προσέγγιση δίνει την δική της απάντηση. Αν οι ετικέτες επισυναφθούν στους κόμβους, τότε η ετικέτα γίνεται ιδιότητα ενός αντικείμενου (αφότου οι κόμβοι αναπαριστούν αντικείμενα), ενώ αν οι ετικέτες επισυναφθούν στις ακμές, τότε η ετικέτα γίνεται ιδιότητα της σχέσης ανάμεσα από δυο αντικείμενα. Το να επισυνάπτουμε ετικέτες στις ακμές επιτρέπει στους κόμβους να «βλέπουν» τον ίδιο κόμβο μέσω διαφορετικών ονομασιών. Στα ακόλουθα, όταν αναφερόμαστε στο OEM θεωρούμε ότι οι ετικέτες επισυνάπτονται στις ακμές, προσέγγιση η οποία ακολουθείται και από τους Πολυδιάστατους Γράφους Δεδομένων. Σαν αυστηρό ορισμό του OEM, υιοθετούμε αυτόν που δίνεται στο [Suc98] (βλέπε Κεφάλαιο 2).

#### 4.2.1 Γιατί αρχίζουμε από το OEM;

Ένα ουσιαστικό κομμάτι κάθε μοντέλου MSSD πρέπει να είναι οι προσδιοριστές περιβάλλοντος. Οι προσδιοριστές περιβάλλοντος είναι εντελώς ανεξάρτητοι από το συγκεκριμένο μοντέλο δεδομένων, συνεπώς κάθε υπάρχον μοντέλο δεδομένων για ημιδομημένα δεδομένα μπορεί να θεωρηθεί σαν η υποδομή για να ενσωματωθούν προσδιοριστές περιβάλλοντος. Όμως, προκειμένου να διερευνηθούν οι ιδιότητες των MSSD σε ένα αφηρημένο επίπεδο, πρέπει να επιλεγεί ένα μοντέλο που είναι εκφραστικό και απλό. Εκφραστικό, καθώς πρέπει να συλλαμβάνει όλα τα ουσιαστικά στοιχεία των ημιδομημένων δεδομένων. Απλό, καθώς πρέπει να χρησιμοποιεί τις ελάχιστες δυνατές πρωταρχικές έννοιες, έτσι ώστε η επέκτασή του με νέες έννοιες να μην συνεπάγεται περιττές επιπλοκές.

Η XML [XML, Wal97, Cha99] έχει κερδίσει ευρεία αποδοχή, και σαν μορφότυπος ανταλλαγής και σαν ένα λογικό μοντέλο για τα δεδομένα του Ιστού· έτσι η XML είναι ισχυρός υποψήφιος για την ενσωμάτωση προσδιοριστών περιβάλλοντος. Όμως, η XML έχει έναν αριθμό συντακτικών ιδιαιτεροτήτων που περιπλέκουν την ενσωμάτωση προσδιοριστών περιβάλλοντος. Συγκεκριμένα, ένα στοιχείο σε ένα έγγραφο XML μπορεί να συσχετίζεται με κάποιο άλλο στοιχείο με δυο διαφορετικούς τρόπους: (a) μπορεί να περιέχει το στοιχείο απευθείας, ή (b) μπορεί να παραπέμπει σε αυτό χρησιμοποιώντας αναγνωριστικά μέσω γνωρισμάτων που έχουν τους ειδικούς τύπους «IDREF» και «IDREFS». Οι δυο αυτοί τρόποι είναι συντακτικά διαφορετικοί, και η διαφορά αυτή συχνά ανακλάται στα μοντέλα δεδομένων γράφου της XML. Πάντως, και οι δυο τρόποι βασικά περιγράφουν μια σχέση στοιχείου-υποστοιχείου. Ο λόγος που υπάρχουν δυο τρόποι για να περιγραφεί το ίδιο στην ουσία πράγμα είναι ότι σε ένα έγγραφο XML ένα στοιχείο δεν μπορεί να περιέχεται από περισσότερα του ενός στοιχεία-πατέρες. Έτσι, ο συντακτικός μηχανισμός των γνωρισμάτων «IDREF» χρειάζεται για να μετατραπεί η δενδρική δομή ενός εγγράφου XML στην πιο ευέλικτη δομή του γράφου, και να μπορούν πολλά στοιχεία-πατέρες να μοιράζονται το ίδιο στοιχείο.

Θα ήταν επιθυμητό να επιλεγεί ένα μοντέλο απαλλαγμένο από τέτοιου είδους περιπλοκές, για να σχηματίσει την βάση της διερεύνησης των ιδιοτήτων των MSSD. Οι λόγοι για την επιλογή του OEM σαν σημείο αφετηρίας είναι ότι αποτελεί ένα μοντέλο εκφραστικό και απλό, και ότι αναπαριστά πληροφορία με τρόπο ομοιογενή και συνεπή. Αυτή η ομοιογένεια κάνει ευκολότερη την ενσωμάτωση νέων χαρακτηριστικών όπως οι προσδιοριστές

περιβάλλοντος, επειδή μπορεί κανείς να συγκεντρωθεί στο πως τα νέα χαρακτηριστικά μεταβάλλουν έναν μικρό αριθμό βασικών εννοιών, αντί να ανησυχεί για δευτερεύοντα ζητήματα συντακτικής φύσης. Στο Κεφάλαιο 7 εισάγουμε την *Πολυδιάστατη XML (MXML* για συντομία) που ενσωματώνει το ερμηνευτικό περιβάλλον και τις πολυδιάστατες οντότητες στην XML.

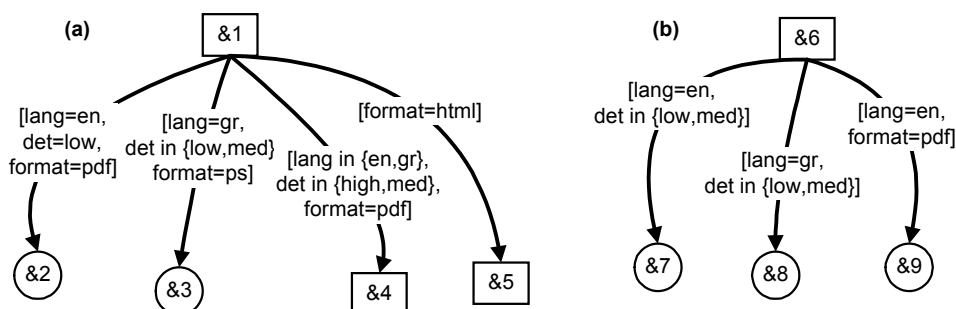
## 4.2.2 Αναπαράσταση πολυδιάστατων οντοτήτων

Για να μοντελοποιήσουμε πολυδιάστατες οντότητες, εισάγουμε δυο νέα βασικά στοιχεία γράφου:

- **Πολυδιάστατους κόμβους:** ένας πολυδιάστατος κόμβος αναπαριστά μια πολυδιάστατη οντότητα, και χρησιμοποιείται για να ομαδοποιήσει τους κόμβους που αποτελούν τις εκφάνσεις της οντότητας αυτής. Οι πολυδιάστατοι κόμβοι έχουν παραλληλόγραμμο σχήμα για να διαφοροποιούνται από τους συμβατικούς κυκλικούς κόμβους.
- **Ακμές περιβάλλοντος:** οι ακμές περιβάλλοντος είναι κατευθυνόμενες ακμές με ετικέτα που συνδέουν έναν πολυδιάστατο κόμβο με τις εκφάνσεις του. Η ετικέτα μιας ακμής περιβάλλοντος που δείχνει σε μια έκφανση  $f$  είναι ένας προσδιοριστής περιβάλλοντος που καθορίζει το σύνολο κόσμων κάτω από τους οποίους υφίσταται η  $f$ . Οι ακμές περιβάλλοντος σχεδιάζονται σαν χονδρές ή διπλές γραμμές, για να ξεχωρίζουν από τις συμβατικές ακμές που σχεδιάζονται σαν απλές λεπτές γραμμές.

Οι εκφάνσεις που δεν είναι οι ίδιες πολυδιάστατες οντότητες αναπαρίστανται σαν συμβατικοί, τύπου OEM κυκλικό κόμβοι, που λέγονται **κόμβοι περιβάλλοντος**. Όλοι οι κόμβοι στον Πολυδιάστατο Γράφο Δεδομένων, περιβάλλοντος ή πολυδιάστατοι, θεωρούνται αντικείμενα και έχουν ένα μοναδικό *αναγνωριστικό αντικείμενου* (*oid* για συντομία). Όπως στο OEM, τα oids από σύμβαση αρχίζουν με τον χαρακτήρα &. Στα ακόλουθα, οι όροι **κόμβος** και **αντικείμενο** θα χρησιμοποιούνται ισοδύναμα.

Εικόνα 4.1: Γραφική αναπαράσταση πολυδιάστατων οντοτήτων.



Η Εικόνα 4.1 απεικονίζει<sup>31</sup> δυο πολυδιάστατες οντότητες που σημειώνονται (a) και (b). Ο πολυδιάστατος κόμβος &1 στην (a) αναπαριστά μια πολυδιάστατη οντότητα της οποίας οι εκφάνσεις περιλαμβάνουν δυο κόμβους περιβάλλοντος (&2 και &3) και δυο πολυδιάστατους

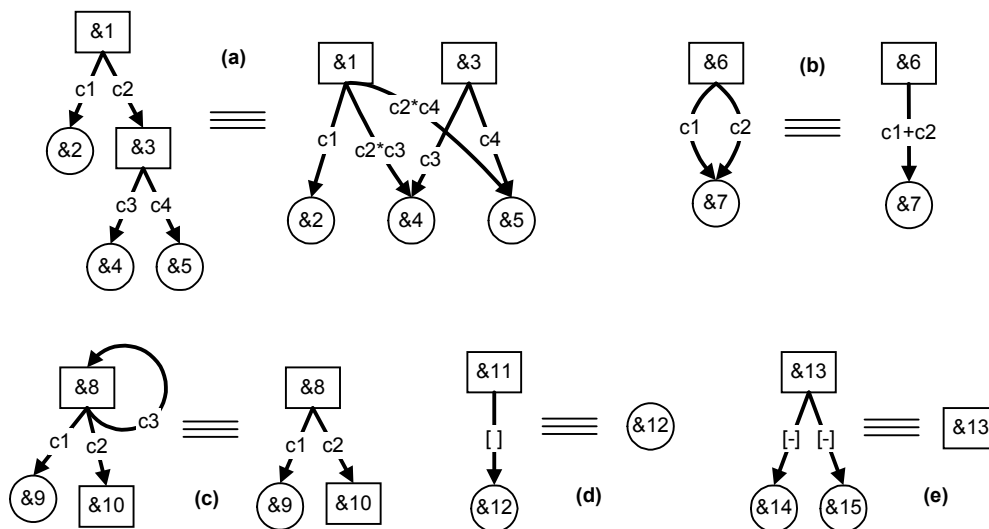
<sup>31</sup> Σημειώστε ότι στα παραδείγματα χρησιμοποιείται η σημειογραφία για τους προσδιοριστές περιβάλλοντος που εισήχθη στο τέλος του προηγούμενου κεφαλαίου, ενώ στους ορισμούς συνεχίζει να χρησιμοποιείται η αυστηρή (formal) σημειογραφία.

κόμβους (&4 και &5). Οι εκφάνσεις αυτές υφίστανται κάτω από ξένα σύνολα κόσμων, με άλλα λόγια, δεδομένου οποιουδήποτε κόσμου  $w$  το πολύ μια έκφανση υφίσταται κάτω από τον  $w$ . Από την άλλη, οι εκφάνσεις της πολυδιάστατης οντότητας στην (b) περιλαμβάνουν τους κόμβους &7 και &9, οι οποίοι υφίστανται κάτω από κοινούς κόσμους.

Δεδομένου του προσδιοριστή περιβάλλοντος  $c_1 = [\text{lang=en, det in \{med, high\}, format=pdf}]$ , η πολυδιάστατη οντότητα στην Εικόνα 4.1 (a) αποτιμάται στον κόμβο &4 κάτω από το  $c_1$ , και ο κόμβος &4 είναι και η μόνη έκφανση που υφίσταται κάτω από κάθε κόσμο του  $c_1$ . Δεδομένου του προσδιοριστή περιβάλλοντος  $c_2 = [\text{lang=en, det in \{med, high\}, format=pdf | lang=gr}]$ , που είναι υπερσύνολο περιβάλλοντος του  $c_1$ , η πολυδιάστατη οντότητα στο (a) δεν αποτιμάται σε κανένα κόμβο κάτω από το  $c_2$ , ενώ οι εκφάνσεις &3, &4, &5 υφίστανται κάτω από κόσμους που καλύπτει το  $c_2$ .

Όπως έχουμε δει στο προηγούμενο κεφάλαιο, οι προσδιοριστές περιβάλλοντος μπορούν να έχουν διαφορετικές μορφές ενώ αναπαριστούν το ίδιο σύνολο κόσμων. Συνεπώς, δυο πολυδιάστατες οντότητες μπορεί να εμπεριέχουν διαφορετικούς προσδιοριστές περιβάλλοντος, αλλά στην ουσία να είναι ίδιες από την στιγμή που οι αντίστοιχοι προσδιοριστές περιβάλλοντος είναι ίσα-περιβάλλοντα. Επιπλέον, οι πολυδιάστατες οντότητες μπορούν να αναπαρασταθούν γραφικά με διάφορους εναλλακτικούς τρόπους, και ταυτόχρονα να διατηρούν τους κόσμους κάτω από τους οποίους υφίστανται οι εκφάνσεις τους.

**Εικόνα 4.2: Εναλλακτικοί τρόποι αναπαράστασης πολυδιάστατων οντοτήτων.**



Στην Εικόνα 4.2 βλέπουμε μερικά ζευγάρια «ισοδύναμων»<sup>32</sup> πολυδιάστατων οντοτήτων. Το ζευγάρι (a) δείχνει ότι *εμφωλιασμένες* πολυδιάστατες οντότητες μπορούν να εκφραστούν σαν *επίπεδες* πολυδιάστατες οντότητες, όπου όλες οι εκφάνσεις είναι κόμβοι περιβάλλοντος, συνδυάζοντας τα περιβάλλοντα μέσω της τομής περιβάλλοντος. Το ζευγάρι (b) δείχνει πως ακμές περιβάλλοντος που έχουν την ίδια αφετηρία και προορισμό μπορούν να αντικατασταθούν από μια και μόνη ακμή περιβάλλοντος, χρησιμοποιώντας την ένωση περιβάλλοντος. Με την χρήση των αρχών που επιδεικνύονται στα ζευγάρια (a) και (b), είναι

<sup>32</sup> Όπως θα φανεί καθαρά στις επόμενες ενότητες, οι Πολυδιάστατοι Γράφοι Δεδομένων που συντίθενται από «ισοδύναμες» πολυδιάστατες οντότητες, *ανάγονται* σε ταυτόσημους OEM γράφους κάτω από κάθε πιθανό κόσμο.



εύκολο να δει κανείς ότι κυκλικά μονοπάτια που συνίστανται από ακμές περιβάλλοντος δεν συνεισφέρουν και μπορούν να αγνοηθούν, όπως απεικονίζεται στο ζευγάρι (c)<sup>33</sup>. Έτσι, το περιβάλλον c3 στο παράδειγμα (c) αγνοείται, επειδή δεν αναπαρίστα κανέναν κόσμο κάτω από τον οποίο να υφίσταται το &9 ή το &10 που να μην αναπαρίσταται ήδη από το c1 ή το c2, αντίστοιχα. Το ζευγάρι (d) δείχνει ότι μια συμβατική οντότητα (που αναπαρίσταται από τον κόμβο περιβάλλοντος &12 στα δεξιά) μπορεί να θεωρηθεί σαν μια πολυδιάστατη οντότητα που έχει μόνο μια έκφραση η οποία υφίσταται κάτω από κάθε κόσμο. Τέλος, το ζευγάρι (e) δείχνει ότι η οριακή περίπτωση μιας πολυδιάστατης οντότητας χωρίς καθόλου εκφάνσεις είναι ισοδύναμη με μια οντότητα που έχει εκφάνσεις οι οποίες δεν υφίστανται κάτω από κανέναν κόσμο.

### 4.2.3 Πολυδιάστατοι Γράφοι Δεδομένων

Σε έναν Πολυδιάστατο Γράφο Δεδομένων πολυδιάστατες οντότητες και συμβατικές οντότητες αλληλοσυνδέονται μέσω συμβατικών ακμών τύπου OEM, που εμφανίζονται σαν μονές λεπτές γραμμές. Αυτές οι γραμμές λέγονται **ακμές οντότητας** και ορίζουν συσχετίσεις μεταξύ αντικειμένων (όπως εξηγήθηκε στην Ενότητα 4.2.2, οι όροι **κόμβος** και **αντικείμενο** χρησιμοποιούνται ισοδύναμα). Οι ακμές οντότητας ξεκινούν από κόμβους περιβάλλοντος και δείχνουν σε κόμβους περιβάλλοντος ή σε πολυδιάστατους κόμβους, μια και ένας πολυδιάστατος κόμβος δρα σαν υποκατάστατο των εκφάνσεών του.

Με την ύπαρξη δυο ειδών κόμβων και δυο ειδών ακμών ανακύπτει το ερώτημα ποιοι συνδυασμοί κόμβου – ακμής είναι αυτοί που έχουν νόημα. Οι ακόλουθοι δυο περιορισμοί είναι οι μόνοι στην μορφολογία ενός Πολυδιάστατου Γράφου Δεδομένων:

- Μια ακμή περιβάλλοντος δεν μπορεί να ξεκινά από κόμβο περιβάλλοντος.
- Μια ακμή οντότητας δεν μπορεί να ξεκινά από πολυδιάστατο κόμβο.

Ένα **μονοπάτι** σε έναν Πολυδιάστατο Γράφο Δεδομένων μπορεί να αποτελείται από ακμές περιβάλλοντος και ακμές οντότητας σε οποιαδήποτε σειρά.

Όπως στο OEM, οι κόμβοι περιβάλλοντος χωρίζονται σε **σύνθετους** και **ατομικούς**. Τα ατομικά αντικείμενα έχουν τιμές από κάποιον βασικό τύπο, π.χ. ακέραιος, πραγματικός, στοιχειοσειρά, κτλ. Η τιμή ενός σύνθετου αντικειμένου είναι ένα σύνολο αναφορών σε αντικείμενα, που αναπαρίστανται από ακμές οντότητας. Όμοια, η τιμή ενός πολυδιάστατου αντικειμένου είναι και πάλι ένα σύνολο αναφορών σε αντικείμενα, που αναπαρίστανται από ακμές περιβάλλοντος. Ισοδύναμα, οι ακμές μπορούν να μην αντιμετωπιστούν σαν ενσωματωμένες στην τιμή των αντικειμένων, αλλά να θεωρηθούν αυτοδύναμες. Στην περίπτωση αυτή σαν τιμές των σύνθετων αντικειμένων και των πολυδιάστατων αντικειμένων θεωρούνται οι δεσμευμένες τιμές C και M αντίστοιχα.

Στην συνέχεια ορίζουμε αυστηρά τον Πολυδιάστατο Γράφο Δεδομένων. Όπως εξηγείται στο βήμα 4, η συνάρτηση  $\nu$  χρησιμοποιεί τις δεσμευμένες τιμές C και M για να κατατάξει τους κόμβους σε πολυδιάστατους, σύνθετους (περιβάλλοντος), και ατομικούς (περιβάλλοντος).

#### Ορισμός 4.3

*Ας είναι το CS το σύνολο όλων των προσδιοριστών περιβάλλοντος, το L το σύνολο όλων των ετικετών, και το A το σύνολο όλων των ατομικών τιμών. Ένας*

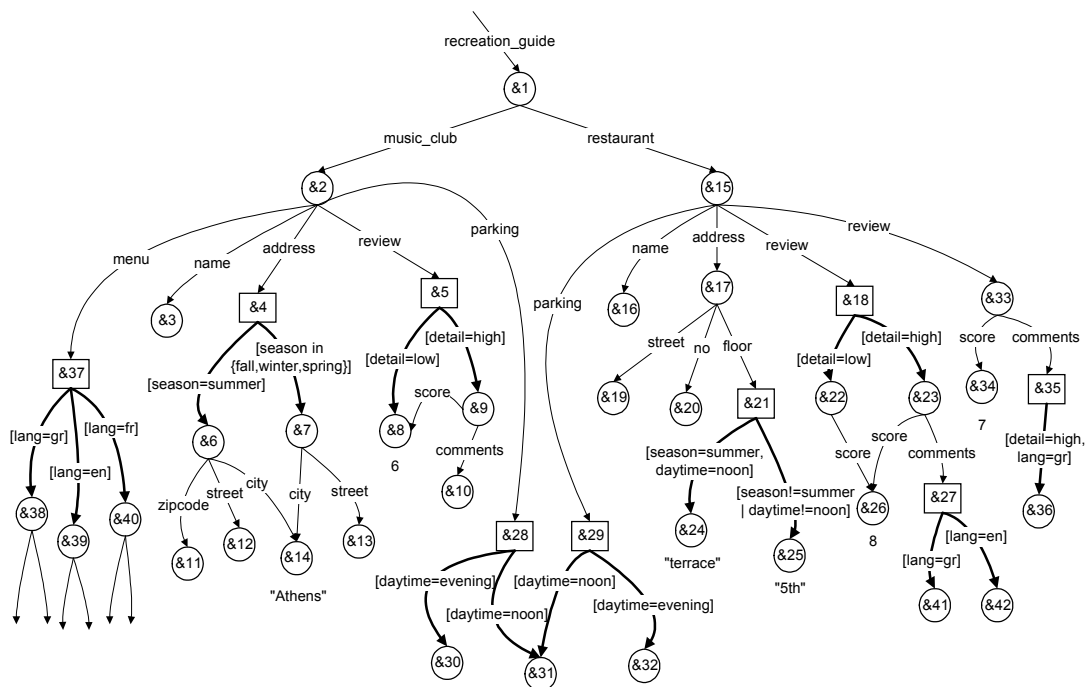
<sup>33</sup> Αυτό δεν σημαίνει ότι ακμές περιβάλλοντος που σχηματίζουν κύκλο είναι οι ίδιες άχρηστες, αλλά ότι το κυκλικό μονοπάτι μπορεί να αγνοηθεί κατά την αποτίμηση μιας πολυδιάστατης οντότητας. Στην ειδική περίπτωση όπου το κυκλικό μονοπάτι αποτελείται από μια μόνο ακμή περιβάλλοντος, όπως στην Εικόνα 4.2 (c), η ακμή είναι πράγματι άχρηστη.

**Πολυδιάστατος Γράφος Δεδομένων**  $G$  είναι ένας πεπερασμένος κατευθυνόμενος πολυγράφος με ετικέτες στις ακμές  $G = (V_{mld}, V_{cxt}, E_{cxt}, E_{etb}, r, v)$ , όπου:

1. Το σύνολο των κόμβων  $V$  αποτελείται από **πολυδιάστατους κόμβους** και **κόμβους περιβάλλοντος**,  $V = V_{mld} \cup V_{cxt}$ . Οι κόμβοι περιβάλλοντος χωρίζονται σε **σύνθετους κόμβους** και **ατομικούς κόμβους**,  $V_{cxt} = V_c \cup V_a$ .
2. Το σύνολο των ακμών  $E$  αποτελείται από **ακμές περιβάλλοντος** και **ακμές οντότητας**,  $E = E_{cxt} \cup E_{etb}$  τέτοιες ώστε  $E_{cxt} \subseteq (V_{mld} \times CS \times V)$  και  $E_{etb} \subseteq (V_c \times L \times V)$ .
3. Το  $r \in V$  είναι η **ρίζα**, με την ιδιότητα ότι υπάρχει μονοπάτι από το  $r$  προς κάθε άλλο κόμβο στο  $V$ .
4. Το  $v$  είναι μια συνάρτηση που αναθέτει τιμές σε κόμβους, έτσι ώστε:  $v(x) = M$  αν  $x \in V_{mld}$ ,  $v(x) = C$  αν  $x \in V_c$ , και  $v(x) = v'(x)$  αν  $x \in V_a$ , όπου  $M$  και  $C$  είναι δεσμευμένες τιμές, και  $v'$  είναι μια συνάρτηση τιμών  $v' : V_a \rightarrow A$  η οποία αναθέτει τιμές σε ατομικούς κόμβους.

Σημειώστε ότι οι ατομικοί κόμβοι σε έναν Πολυδιάστατο Γράφο Δεδομένων πρέπει να είναι φύλλα, όμως τα φύλλα δεν είναι αναγκαστικά ατομικοί κόμβοι και μπορούν να είναι οποιοδήποτε είδος κόμβου, ακόμη και σύνθετοι ή πολυδιάστατοι κόμβοι. Επιπλέον, η ρίζα ενός Πολυδιάστατου Γράφου Δεδομένων μπορεί να είναι κόμβος περιβάλλοντος ή πολυδιάστατος κόμβος. Φαίνεται εύκολα ότι το OEM είναι ειδική περίπτωση του Πολυδιάστατου Γράφου Δεδομένων, όπου δεν υπάρχουν πολυδιάστατοι κόμβοι και ακμές περιβάλλοντος, και όπου τα φύλλα περιορίζονται σε ατομικούς κόμβους.

**Εικόνα 4.3: Παράδειγμα Πολυδιάστατου Γράφου Δεδομένων που αναπαριστά έναν οδηγό διασκέδασης εξαρτώμενο από το ερμηνευτικό περιβάλλον.**



Ο Πολυδιάστατος Γράφος Δεδομένων στην Εικόνα 4.3 είναι ένα παράδειγμα ενός οδηγού διασκέδασης που εξαρτάται από το ερμηνευτικό περιβάλλον<sup>34</sup>. Για απλότητα, ο γράφος δεν είναι πλήρως αναπτυγμένος και κάποια από τα ατομικά αντικείμενα δεν έχουν τιμές προσαρτημένες. Οι διαστάσεις και τα αντίστοιχα πεδία ορισμού των στην Εικόνα 4.3 έχουν ως ακολούθως: `season` που κυμαίνεται στο `{summer, fall, winter, spring}`, `daytime` που κυμαίνεται στο `{noon, evening}`, `detail` που κυμαίνεται στο `{high, low}`, και `lang` που κυμαίνεται στο `{en, fr, gr}`. Το `restaurant` με `oid &15` κανονικά λειτουργεί στον πέμπτο όροφο, αλλά τα μεσημέρια του καλοκαιριού λειτουργεί στην ταράτσα. Έτσι, το `floor` με `oid &21` είναι ένα πολυδιάστατο αντικείμενο του οποίου η (ατομική) τιμή εξαρτάται από τις διαστάσεις `season` και `daytime`.

Εκτός από το να έχουν διαφορετική τιμή, τα αντικείμενα περιβάλλοντος μπορούν να έχουν και διαφορετική δομή, όπως είναι η περίπτωση των `&6` και `&7` τα οποία είναι εκφάνσεις του πολυδιάστατου αντικειμένου `address` με `oid &4`. Στην περίπτωση αυτή, το `music_club` με `oid &2` λειτουργεί σε διαφορετική διεύθυνση το καλοκαίρι απ' ότι τον υπόλοιπο χρόνο (στην Αθηνά δεν είναι ασύνηθες για τα κέντρα διασκέδασης να μεταφέρονται νότια κοντά στην θάλασσα την καλοκαιρινή περίοδο, και βόρεια προς το κέντρο της πόλης την υπόλοιπη χρονιά). Το `menu` του `club` είναι διαθέσιμο σε τρεις γλώσσες, δηλαδή Αγγλικά, Γαλλικά και Ελληνικά. Το εστιατόριο και το `club` έχουν έναν αριθμό `reviews` που μπορούν να είναι λεπτομερείς ή σύντομες, ανάλογα με την διάσταση `detail`. Επιπλέον, καθένα έχει δυο εναλλακτικά μέρη `parking`, ανάλογα με την ώρα της ημέρας όπως αυτή εκφράζεται από την διάσταση `daytime`.

## 4.2.4 Το σύνολο των διαστάσεων

Οι προσδιοριστές περιβάλλοντος σε έναν Πολυδιάστατο Γράφο Δεδομένων ερμηνεύονται σε σχέση με ένα σύνολο διαστάσεων **D**, το οποίο θεωρείται ότι συνοδεύει τον γράφο. Πάντως είναι σημαντικό να συνειδητοποιήσει κανείς ότι, όπως δείξαμε στο προηγούμενο κεφάλαιο, το **D** είναι απαραίτητο μόνον για να μετατρέψουμε έναν προσδιοριστή περιβάλλοντος σε σύνολο κόσμων μέσω της επέκτασης περιβάλλοντος ή της ανάπτυξης επεκτεταμένου περιβάλλοντος· για τις υπόλοιπες πράξεις περιβάλλοντος, δεν είναι απαραίτητο να ληφθεί υπόψη το πραγματικό **D**.

Για τον λόγο αυτό, για τα ακόλουθα θα θεωρήσουμε ότι το σύνολο διαστάσεων **D** για έναν Πολυδιάστατο Γράφο Δεδομένων **G** αποτελείται από τις διαστάσεις που απαντώνται στον **G** και από καμία άλλη, εκτός εάν δηλώνεται ρητά κάτι διαφορετικό.

Τα πεδία ορισμού των διαστάσεων χρειάζονται για όλες τις πράξεις περιβάλλοντος εκτός από την τομή περιβάλλοντος, την ένωση περιβάλλοντος, και την ανάπτυξη περιβάλλοντος. Για απλότητα, το πεδίο ορισμού μιας διάστασης θεωρείται ότι αποτελείται από τις τιμές αυτής της διάστασης που απαντώνται σε έναν Πολυδιάστατο Γράφο Δεδομένων και από καμία άλλη, εκτός εάν δηλώνεται ρητά κάτι διαφορετικό.

Υπάρχει μια σημαντική διαφορά ανάμεσα στις δυο αυτές παραδοχές. Η τελευταία παραδοχή που αφορά στα πεδία ορισμού των διαστάσεων είναι απλά μια σύμβαση για την απλοποίηση των παραδειγμάτων από εδώ και εμπρός· σε μια πραγματική εφαρμογή, η χρήση των πραγματικών πεδίων ορισμού θα δώσει πιθανότατα διαφορετικά αποτελέσματα. Η πρώτη παραδοχή που αφορά στο σύνολο των διαστάσεων μπορεί να ιδωθεί περισσότερο σαν

<sup>34</sup> Όπως και στο OEM, χρησιμοποιούμε μια επιπλέον ακμή με ετικέτα που δείχνει στην ρίζα προκειμένου να δώσουμε ένα όνομα στην ρίζα, και κατ' επέκταση στον γράφο και στην αντίστοιχη ημιδομημένη βάση.

μια βελτιστοποίηση· στην πραγματικότητα δεν έχει σημασία πιο είναι το σύνολο των διαστάσεων, εκτός αν πρέπει να εκτελεστεί επέκταση περιβάλλοντος ή ανάπτυξη επεκτεταμένου περιβάλλοντος.

### 4.3 ΜΕΤΑΛΛΟΣΗ ΠΕΡΙΒΑΛΛΟΝΤΟΣ

Οι προσδιοριστές περιβάλλοντος που επισυνάπτονται στις ακμές περιβάλλοντος φανερώνουν τα *επιδιώκόμενα* περιβάλλοντα αλλά όχι κατ' ανάγκη και τα *πραγματικά* περιβάλλοντα των αντιστοίχων εκφάνσεων. Για να καταλάβουμε τον λόγο, θεωρήστε μια πολυδιάστατη οντότητα που αποτιμάται σε μια έκφανση  $f$  κάτω από έναν κόσμο  $w$ , και που δείχνεται από ένα αντικείμενο το οποίο δεν υφίσταται κάτω από το  $w$ . Τότε το  $f$  δεν επιβιώνει κάτω από το  $w$ , επειδή ο πατέρας του δεν υπάρχει κάτω από το  $w$ . Η παρατήρηση αυτή οδηγεί στον παρακάτω ορισμό του **ρητού περιβάλλοντος**, και στην έννοια του **κληρονομούμενου περιβάλλοντος**, που είναι το θέμα της επόμενης ενότητας.

#### Ορισμός 4.4

Ας είναι το  $G = (V_{mld}, V_{cxb}, E_{cxb}, E_{etb}, r, v)$  ένας Πολυδιάστατος Γράφος Δεδομένων. Το **ρητό περιβάλλον** μιας ακμής  $h = (p, k, q)$  είναι  $k$  αν  $h \in E_{cxb}$ , και  $\{\emptyset\}$  αν  $h \in E_{etb}$ .

Συνεπώς, ο προσδιοριστής περιβάλλοντος που είναι συνημμένος σε μια ακμή περιβάλλοντος αποτελεί το ρητό περιβάλλον της ακμής αυτής, ενώ το ρητό περιβάλλον μιας ακμής οντότητας είναι το καθολικό περιβάλλον  $\{\emptyset\}$  (που σημειώνεται επίσης και  $[]$  όπως έχουμε δει), πράγμα που υπονοεί ότι οι ακμές οντότητας επιδιώκεται να υφίστανται κάτω από κάθε κόσμο. Οι κόμβοι δεν έχουν ρητό περιβάλλον.

Ακολουθώς, ορίζουμε το ρητό περιβάλλον ενός μονοπατιού.

#### Ορισμός 4.5

Ας είναι το  $G = (V_{mld}, V_{cxb}, E_{cxb}, E_{etb}, r, v)$  ένας Πολυδιάστατος Γράφος Δεδομένων, και τα  $g_1, g_2, \dots, g_n$  ακμές στο  $E$  που ορίζουν ένα μονοπάτι  $p$  στο  $G$ . Ας είναι τα  $ec_1, ec_2, \dots, ec_n$  τα αντίστοιχα ρητά περιβάλλοντα των  $g_1, g_2, \dots, g_n$ . Τότε, το **ρητό περιβάλλον μονοπατιού** του  $p$  είναι  $ec_p = ec_1 \cap^c ec_2 \cap^c \dots \cap^c ec_n$ .

Το ρητό περιβάλλον μονοπατιού δίνει τους κόσμους κάτω από τους οποίους ολόκληρο το μονοπάτι επιδιώκεται να υφίσταται, όπως αυτό επιβάλλεται από τα ρητά περιβάλλοντα των ακμών του μονοπατιού.

#### 4.3.1 Κληρονομούμενο περιβάλλον

Το ρητό περιβάλλον μπορεί να θεωρηθεί σαν το «πραγματικό» περιβάλλον μόνον μέσα στα όρια μιας απομονωμένης πολυδιάστατης οντότητας. Από την στιγμή που οντότητες συνδέονται μεταξύ τους σε έναν πολυδιάστατο γράφο, το ρητό περιβάλλον μιας ακμής δεν είναι το «πραγματικό» περιβάλλον, με την έννοια ότι δεν προσδιορίζει από μόνο του τους κόσμους κάτω από τους οποίους υφίσταται ο κόμβος προορισμού. Ο λόγος για αυτό είναι ότι, όταν μια οντότητα  $e_2$  αποτελεί μέρος (δείχνεται από κάποια ακμή) μιας άλλης οντότητας  $e_1$ , τότε η  $e_2$  μπορεί να έχει υπόσταση μόνο κάτω από τους κόσμους που έχει υπόσταση η  $e_1$ . Αυτό μπορεί να γίνει αντιληπτό σαν το περιβάλλον κάτω από το οποίο υφίσταται το  $e_1$  να κληρονομείται στο  $e_2$ . Το περιβάλλον που μεταδίδεται με αυτό τον τρόπο συνδυάζεται με (περιορίζεται από) το ρητό περιβάλλον κάθε ακμής για να δώσει το **κληρονομούμενο**

**περιβάλλον** της ακμής αυτής. Αντίθετα με τις ακμές, οι κόμβοι δεν έχουν ρητό περιβάλλον· όπως οι ακμές, πάντως, οι κόμβοι έχουν και αυτοί κληρονομούμενο περιβάλλον.

Είναι σημαντικό να τονιστεί ότι, από εδώ και στο εξής<sup>35</sup> θεωρούμε τα κληρονομούμενα περιβάλλοντα αντί για τα ρητά περιβάλλοντα προκειμένου να αποφασίσουμε τους κόσμους κάτω από τους οποίους υφίσταται ένας κόμβος ή μια ακμή. Συγκεκριμένα, ένας κόμβος ή μια ακμή υφίσταται κάτω από έναν κόσμο  $w$  αν το κληρονομούμενο περιβάλλον αυτού του κόμβου ή της ακμής καλύπτει το  $w$ . Εξυπακούεται ότι ένας κόμβος ή μια ακμή υφίσταται κάτω από ένα περιβάλλον  $c$  εφόσον το κληρονομούμενο περιβάλλον του είναι υπερσύνολο περιβάλλοντος του  $c$ .

#### Ορισμός 4.6

*Ας είναι ο  $G = (V_{mld}, V_{ext}, E_{ext}, E_{int}, r, v)$  ένας Πολυδιάστατος Γράφος Δεδομένων, το  $ic_r$  το κληρονομούμενο περιβάλλον του  $r$ , και το  $p$  ένας κόμβος στο  $V$  με  $p \neq r$ . Το κληρονομούμενο περιβάλλον του κόμβου  $p$  είναι  $ic_p = ic_1 \cup ic_2 \cup \dots \cup ic_n$ , με  $n \geq 1$ , όπου τα  $ic_1, ic_2, \dots, ic_n$  είναι τα κληρονομούμενα περιβάλλοντα των ακμών στο  $E$  οι οποίες καταλήγουν στο  $p$ . Ας είναι το  $q$  ένας κόμβος στο  $V$ , το  $ic_q$  το κληρονομούμενο περιβάλλον του κόμβου  $q$ , το  $h$  μια ακμή στο  $E$  που ξεκινά από το  $q$ , και το  $ec_h$  το ρητό περιβάλλον του  $h$ . Το κληρονομούμενο περιβάλλον της ακμής  $h$  είναι ένας προσδιοριστής περιβάλλοντος  $ic_h$ , τέτοιος ώστε: (a)  $ic_h = ic_q \cap ec_h$  και (b) το  $ic_h$  αναπαριστά το ελάχιστο σύνολο κόσμων.*

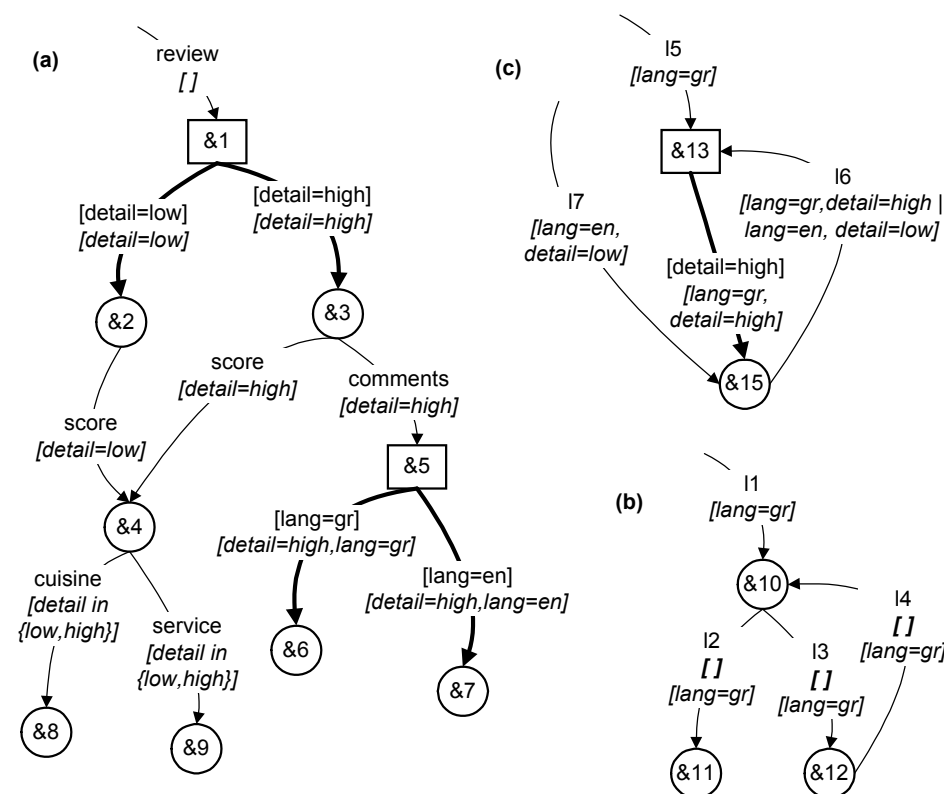
Η συνθήκη (b) στον ορισμό του κληρονομούμενου περιβάλλοντος μιας ακμής συζητείται περαιτέρω στην ενότητα που ακολουθεί.

#### 4.3.1.1 Υπολογισμός του κληρονομούμενου περιβάλλοντος

Αν το κληρονομούμενο περιβάλλον της ρίζας είναι γνωστό, μπορούμε να υπολογίσουμε το κληρονομούμενο περιβάλλον κάθε κόμβου και ακμής διασχίζοντας τον γράφο κατά πλάτος. Στην περίπτωση που ο γράφος δεν περιέχει κύκλους, μια απλή κατά πλάτος διάσχιση θα είναι αρκετή: για κάθε κόμβο που επισκεπτόμαστε, το κληρονομούμενο περιβάλλον των γονέων του έχει ήδη υπολογιστεί, και έτσι είναι δυνατόν να βρούμε το κληρονομούμενο περιβάλλον του τρέχοντος κόμβου υπολογίζοντας τα κληρονομούμενα περιβάλλοντα των εισερχόμενων ακμών. Η περίπτωση αυτή απεικονίζεται στην Εικόνα 4.4 (a), που δείχνει ένα τροποποιημένο τμήμα του γράφου στην Εικόνα 4.3. Το κληρονομούμενο περιβάλλον των ακμών αναγράφεται με πλάγιους χαρακτήρες, κάτω από τα ρητά περιβάλλοντα / ετικέτες των ακμών περιβάλλοντος / ακμών οντότητας αντίστοιχα. Το κληρονομούμενο περιβάλλον ενός κόμβου είναι η ένωση περιβάλλοντος των κληρονομούμενων περιβαλλόντων των εισερχόμενων ακμών. Για παράδειγμα, το κληρονομούμενο περιβάλλον του κόμβου &4 είναι [detail=low | detail=high] που είναι ίσο-περιβάλλον με το [detail in {low,high}].

<sup>35</sup> ...από εδώ και στο εξής και μέχρι να εισάγουμε την κληρονομούμενη κάλυψη, που θα πάρει την θέση του κληρονομούμενου περιβάλλοντος στο να αποφασίζουμε κάτω από ποιους κόσμους υφίσταται στην πραγματικότητα ένας κόμβος ή μια ακμή.

Εικόνα 4.4: Υπολογισμός του κληρονομούμενου περιβάλλοντος.



Τα πράγματα είναι πιο πολύπλοκα στην περίπτωση που ο γράφος περιέχει κύκλους. Το κληρονομούμενο περιβάλλον ενός κόμβου ή μιας ακμής που αποτελεί μέρος ενός κύκλου καθορίζεται εν τέλει ως προς τον εαυτό του. Θα δείξουμε ότι σε αυτές τις περιπτώσεις υπάρχει ένα *ελάχιστο σταθερό σημείο* (least fixed point) [Ull88], που δίνει το κληρονομούμενο περιβάλλον του κόμβου ή της ακμής. Στην Εικόνα 4.4 (b) το κληρονομούμενο περιβάλλον του κόμβου &10 εξαρτάται από αυτό της ακμής I4, το οποίο τελικά εξαρτάται και πάλι από το κληρονομούμενο περιβάλλον του κόμβου &10. Ας υποθέσουμε ότι η ακμή I4 έχει σαν κληρονομούμενο περιβάλλον το [], που σημειώνεται με έντονους χαρακτήρες στην Εικόνα 4.4 (b). Τότε, το κληρονομούμενο περιβάλλον των κόμβων &10, &11, &12 και των εδρών I2, I3 θα είναι επίσης το [], αποτέλεσμα που συμμορφώνεται πλήρως με τον ορισμό του κληρονομούμενου περιβάλλοντος σε ότι αφορά τις ενώσεις περιβάλλοντος και τις τομές περιβάλλοντος. Το αποτέλεσμα αυτό πάντως δεν συμμορφώνεται με την συνθήκη που αναφέρει ο Ορισμός 4.6 και που απαιτεί το κληρονομούμενο περιβάλλον μιας ακμής να αναπαριστά το ελάχιστο σύνολο κόσμων. Το σωστό κληρονομούμενο περιβάλλον των κόμβων &10, &11, &12 και των ακμών I2, I3, I4 είναι [lang=gr]. Μια παρόμοια περίπτωση επιδεικνύεται στην Εικόνα 4.4 (c), όπου τα πραγματικά κληρονομούμενα περιβάλλοντα τα οποία αναπαριστούν το ελάχιστο σύνολο κόσμων αναγράφονται στις ακμές με πλάγιους χαρακτήρες. Για να βρούμε το κληρονομούμενο περιβάλλον του κόμβου &13, αρχίζουμε με την αρχική παραδοχή ότι είναι [lang=gr], όπως το κληρονομούμενο περιβάλλον της ακμής I5. Με βάση αυτό, υπολογίζουμε το κληρονομούμενο περιβάλλον της ακμής περιβάλλοντος, έπειτα του κόμβου &15 και έπειτα της ακμής I4. Στην συνέχεια, υπολογίζουμε και πάλι το κληρονομούμενο περιβάλλον του &13, που αυτή την φορά δίνει διαφορετικό αποτέλεσμα: [lang=gr | lang=en, detail=low]. Τα βήματα αυτά επαναλαμβάνονται μέχρις ότου δεν λαμβάνουν πια χώρα άλλες αλλαγές στα

κληρονομούμενα περιβάλλοντα των κόμβων και των ακμών του κύκλου, και τα αποτελέσματα είναι ίσα-περιβάλλοντα με αυτά της προηγούμενης επανάληψης.

Η διαδικασία που περιγράφηκε πιο πάνω μπορεί να εφαρμοστεί σε οποιοδήποτε ισχυρά συνδεδεμένο τμήμα<sup>36</sup> (strongly connected component) σε έναν Πολυδιάστατο Γράφο Δεδομένων, και είναι ένας απλοϊκός τρόπος για την αποτίμηση του ελάχιστου σταθερού σημείου. Όπως φαίνεται υπάρχει πάντοτε ένα ελάχιστο σταθερό σημείο που δίνει το κληρονομούμενο περιβάλλον, επειδή:

- (a) Η τομή περιβάλλοντος και η ένωση περιβάλλοντος είναι *μονότονες*, όπως έχουμε αναφέρει στο προηγούμενο κεφάλαιο. Επιπλέον, η σύνθεση μονότονων πράξεων είναι επίσης μονότονη.
- (b) Ο αριθμός των κόσμων που μπορούν να αναπαρασταθούν από οποιοδήποτε προσδιοριστή περιβάλλοντος είναι άνω φραγμένος, μια και το σύνολο των διαστάσεων και τα πεδία ορισμού των διαστάσεων θεωρούνται πεπερασμένα.

Οι παραπάνω συνθήκες εξασφαλίζουν [UII88] ότι, δεδομένου ενός συνόλου εξισώσεων περιβάλλοντος που ορίζουν τα κληρονομούμενα περιβάλλοντα σε έναν γράφο, υπάρχει πάντοτε μια λύση που συμμορφώνεται με ότι διατυπώνει ο Ορισμός 4.6. Η λύση αυτή είναι το ελάχιστο σταθερό σημείο των εξισώσεων περιβάλλοντος. Συνεπώς, το πρόβλημα του υπολογισμού των κληρονομούμενων περιβαλλόντων σε έναν γράφο με κύκλους μπορεί να μεταφραστεί στην αποτίμηση των ελάχιστων σταθερών σημείων. Μια απλοϊκή μέθοδος αποτίμησης είναι η επανάληψη του υπολογισμού των κληρονομούμενων περιβαλλόντων, χρησιμοποιώντας τα αποτελέσματα κάθε γύρου σαν είσοδο στον επόμενο, μέχρις ότου να μην παρατηρηθούν διαφορές ανάμεσα στα αποτελέσματα δυο διαδοχικών υπολογιστικών γύρων.

Το κληρονομούμενο περιβάλλον μπορεί να οριστεί με έναν εναλλακτικό τρόπο, χρησιμοποιώντας την έννοια του **κληρονομούμενου περιβάλλοντος μονοπατιού**.

#### Ορισμός 4.7

*Ας είναι το  $G = (V_{mla}, V_{cxb}, E_{cxb}, E_{etb}, r, \nu)$  ένας Πολυδιάστατος Γράφος Δεδομένων, και τα  $g_1, g_2, \dots, g_n$  ακμές στο  $E$  που ορίζουν ένα μονοπάτι  $p$  στο  $G$ . Ας είναι τα  $ic_1, ic_2, \dots, ic_n$  τα αντίστοιχα κληρονομούμενα περιβάλλοντα των  $g_1, g_2, \dots, g_n$ . Τότε, το κληρονομούμενο περιβάλλον μονοπατιού του  $p$  είναι  $ic_p = ic_1 \cap ic_2 \cap \dots \cap ic_n$ .*

Το κληρονομούμενο περιβάλλον μονοπατιού αναπαριστά τους κόσμους κάτω από τους οποίους υφίστανται όλες οι ακμές ενός μονοπατιού. Σημειώστε ότι, εξαιτίας όσων δηλώνει ο Ορισμός 4.6, οι κόμβοι του μονοπατιού (συμπεριλαμβανόμενων του πρώτου και του τελευταίου κόμβου) υφίστανται και αυτοί κάτω από τους ίδιους κόσμους.

Η ακόλουθη πρόταση δείχνει ότι το κληρονομούμενο περιβάλλον ενός κόμβου μπορεί να οριστεί ως προς το κληρονομούμενο περιβάλλον μονοπατιού.

#### Πρόταση 4.1

*Το κληρονομούμενο περιβάλλον ενός κόμβου  $q$  δίνεται από την ένωση περιβάλλοντος των κληρονομούμενων περιβαλλόντων μονοπατιού όλων των μονοπατιών που ξεκινούν από την ρίζα και καταλήγουν στο  $q$ .*

<sup>36</sup> Ένα ισχυρά συνδεδεμένο τμήμα ενός κατευθυνόμενου γράφου είναι κάποιος υπογράφος που περιέχει ξεχωριστούς κόμβους  $q$ ,  $u$  τέτοιους ώστε υπάρχει μονοπάτι από το  $q$  στο  $u$  και από το  $u$  στο  $q$ .

Η ακόλουθη πρόταση δείχνει πως τα ρητά περιβάλλοντα των ακμών μπορούν να χρησιμοποιηθούν αντί των κληρονομούμενων περιβαλλόντων για τον υπολογισμό του κληρονομούμενου περιβάλλοντος μονοπατιού.

#### Πρόταση 4.2

*Ας είναι το  $G = (V_{mlb}, V_{cxb}, E_{cxb}, E_{etb}, r, \nu)$  ένας Πολυδιάστατος Γράφος Δεδομένων, το  $q$  ένας κόμβος στο  $V$  με κληρονομούμενο περιβάλλον  $ic_q$ , και το  $p$  ένα μονοπάτι στο  $G$  που ξεκινά από το  $q$ . Ας είναι το  $ec_p$  το ρητό περιβάλλον μονοπατιού του  $p$ . Τότε, το κληρονομούμενο περιβάλλον μονοπατιού  $ic_p$  του  $p$  δίνεται από το:  $ic_p = ic_q \cap ec_p$ .*

Χρησιμοποιώντας την Πρόταση 4.1 και την Πρόταση 4.2 μπορούμε να εκφράσουμε το κληρονομούμενο περιβάλλον ενός κόμβου ή μιας ακμής με βάση το κληρονομούμενο περιβάλλον της ρίζας και τα ρητά περιβάλλοντα των ακμών, αποφεύγοντας την αναδρομή που εισάγει ο Ορισμός 4.6:

#### Πρόταση 4.3

*Το κληρονομούμενο περιβάλλον ενός κόμβου είναι η ένωση περιβάλλοντος των κληρονομούμενων περιβαλλόντων μονοπατιού για κάθε μονοπάτι που οδηγεί από την ρίζα στον κόμβο αυτό, όπου το κληρονομούμενο περιβάλλον μονοπατιού δίνεται από την τομή περιβάλλοντος όλων των ρητών περιβαλλόντων των ακμών στο μονοπάτι με το κληρονομούμενο περιβάλλον της ρίζας. Ομοίως, για το κληρονομούμενο περιβάλλον μιας ακμής θεωρούμε όλα τα μονοπάτια που ξεκινούν από την ρίζα και καταλήγουν στην ακμή αυτή.*

Σημειώστε ότι, εφόσον ένας γράφος περιέχει κύκλους είναι δυνατόν να γεννηθούν άπειρα μονοπάτια, ακόμη και αν ο γράφος είναι πεπερασμένος. Για να αποφύγουμε την δημιουργία άπειρων μονοπατιών θα πρέπει να προσέξουμε ώστε να μην διασχίσουμε πολλές φορές την ίδια διαδρομή. Έτσι, ο εναλλακτικός ορισμός του κληρονομούμενου περιβάλλοντος που δίνεται στην Πρόταση 4.3 στην πράξη μεταφέρει την αποτίμηση των ελάχιστων σταθερών σημείων από προσδιοριστές περιβάλλοντος και κόσμους, σε μονοπάτια.

Ένα ενδιαφέρον σημείο είναι ότι δεν είναι απαραίτητο να λάβουμε υπόψη το σύνολο των διαστάσεων  $\mathbf{D}$  ή των πεδίων ορισμού των διαστάσεων για τον υπολογισμό των κληρονομούμενων περιβαλλόντων ενός Πολυδιάστατου Γράφου Δεδομένων, επειδή, όπως δείξαμε στο προηγούμενο κεφάλαιο, η τομή περιβάλλοντος και η ένωση περιβάλλοντος δεν εξαρτώνται από το  $\mathbf{D}$  ή από τα πεδία ορισμού των διαστάσεων<sup>37</sup>.

#### 4.3.1.2 Σημασία του κληρονομούμενου περιβάλλοντος

Αρχίζοντας από την ρίζα, οι κόμβοι και οι ακμές ενός Πολυδιάστατου Γράφου Δεδομένων  $G$  που υφίστανται κάτω από έναν κόσμο  $w$  σχηματίζουν έναν υπογράφο του  $G$  που υφίσταται κάτω από το  $w$ <sup>38</sup>. Ένας Πολυδιάστατος Γράφος Δεδομένων  $G$  μπορεί να ενσωματώνει έναν αριθμό γράφων που υφίστανται κάτω από διάφορα περιβάλλοντα, καθώς

<sup>37</sup> Στο παράδειγμα στην Εικόνα 4.4 (c) χρησιμοποιήσαμε ισότητα περιβάλλοντος, που εξαρτάται από τα πεδία ορισμού των διαστάσεων. Όμως, όπως δείχνει η Πρόταση 4.3, ακόμη και όταν υπάρχουν κύκλοι όπως στην περίπτωση του παραδείγματος στην Εικόνα 4.4 (c), είναι δυνατόν να υπολογίσουμε τα κληρονομούμενα περιβάλλοντα χωρίς την χρήση της ισότητας περιβάλλοντος. Τα πεδία ορισμού των διαστάσεων είναι πάντως απαραίτητα, από την στιγμή που προσδιοριστές περιβάλλοντος στον γράφο περιέχουν συντακτικά στοιχεία όπως «!=» ή «not in».

<sup>38</sup> Εκτός εάν αναφέρεται ρητά κάτι άλλο, αναφερόμαστε πάντοτε στον μέγιστο υπογράφο που υφίσταται κάτω από το  $w$ .



διαφορετικοί υπογράφοι του  $G$  είναι πιθανόν να έχουν υπόσταση κάτω από διαφορετικούς κόσμους.

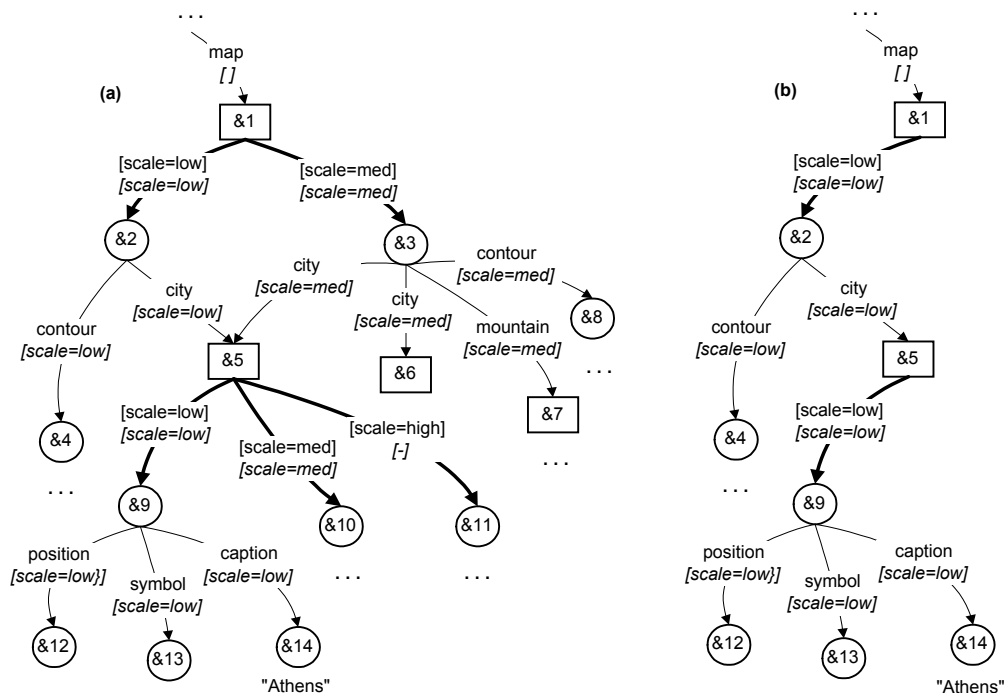
*Το κληρονομούμενο περιβάλλον ενός κόμβου ή μιας ακμής αναπαριστά τους κόσμους κάτω από τους οποίους ο κόμβος ή η ακμή «επιβιώνει» σαν τμήμα τέτοιων υπογράφων. Ο λόγος είναι ότι το κληρονομούμενο περιβάλλον στην πραγματικότητα αποτελεί μια μετάδοση περιορισμών από την ρίζα προς τα φύλλα, συνεπώς εάν το κληρονομούμενο περιβάλλον ενός κόμβου ή μιας ακμής καλύπτει έναν κόσμο  $w$ , τότε δεν υπάρχει κάποιος περιορισμός που να απαγορεύει στον κόμβο ή στην ακμή να υφίσταται κάτω από το  $w$ <sup>39</sup>.*

Δεδομένου ενός Πολυδιάστατου Γράφου Δεδομένων  $G$ , οι υπογράφοι του  $G$  που υφίστανται κάτω από διάφορους πιθανούς κόσμους θα είναι και αυτοί Πολυδιάστατοι Γράφοι Δεδομένων και θα έχουν την ίδια ρίζα όπως ο  $G$ . Να γιατί: Εξαιτίας του ορισμού του κληρονομούμενου περιβάλλοντος, αν το κληρονομούμενο περιβάλλον μιας ακμής καλύπτει έναν κόσμο  $w$ , τότε το  $w$  καλύπτεται επίσης και από τα κληρονομούμενα περιβάλλοντα των δυο κόμβων που συνδέει η ακμή αυτή. Συνεπώς, εάν μια ακμή υφίσταται κάτω από έναν κόσμο  $w$ , τότε οι αντίστοιχοι κόμβοι αφηρησίας και άφιξης υφίστανται επίσης κάτω από το  $w$ : με άλλα λόγια, οι ακμές που επιβιώνουν πάντοτε καταλήγουν σε κόμβους που επιβιώνουν. Επιπλέον, ένα σημαντικό συμπέρασμα που προκύπτει από την Πρόταση 4.1 είναι ότι, αν το κληρονομούμενο περιβάλλον ενός κόμβου ή μιας ακμής καλύπτει έναν κόσμο  $w$ , τότε αυτός ο κόμβος ή η ακμή πρέπει να είναι μέρος ενός μονοπατιού το οποίο αρχίζει από την ρίζα και του οποίου το κληρονομούμενο περιβάλλον μονοπατιού καλύπτει επίσης το  $w$ . Με άλλα λόγια, οι κόμβοι και οι ακμές ενός Πολυδιάστατου Γράφου Δεδομένων που υφίστανται κάτω από έναν κόσμο  $w$  είναι πάντοτε προσβάσιμοι από την ρίζα μέσω κάποιου μονοπατιού που υφίσταται κάτω από το  $w$ .

---

<sup>39</sup> Αυτό δεν είναι εντελώς αληθινό, όπως θα δούμε στην Ενότητα 4.3.2, όπου εισάγουμε την **κάλυψη περιβάλλοντος**.

Εικόνα 4.5: Ο υπογράφος του (a) που υφίσταται κάτω από τον κόσμο [scale=low] φαίνεται στο (b).



Σαν ένα απλό παράδειγμα, θεωρήστε τον γράφο που απεικονίζεται στην Εικόνα 4.5 (a) που αναπαριστά χαρτογραφική πληροφορία για έναν χάρτη που μπορεί να παρουσιαστεί σε διαφορετικές κλίμακες (μεγαλύτερη κλίμακα σημαίνει ότι δείχνονται στον χάρτη περισσότερα χαρακτηριστικά). Όταν η κλίμακα είναι χαμηλή, ο χάρτης (oid &2) αποτελείται από ένα γενικό περίγραμμα (oid &4) της περιοχής, και την πιο σημαντική πόλη (oid &5). Όταν η κλίμακα είναι μέτρια, ο χάρτης (oid &3) συμπεριλαμβάνει ένα πιο λεπτομερές περίγραμμα (oid &8), ένα βουνό (oid &7), και δυο πόλεις (oids &5 και &6). Τα χαρακτηριστικά του χάρτη μπορούν και αυτά με την σειρά τους να έχουν διαφορετικές εκφάνσεις που εξαρτώνται από την κλίμακα, και που αποφασίζουν τις λεπτομέρειες παρουσίασης και τις ιδιότητες αυτών των χαρακτηριστικών. Για παράδειγμα, η πόλη με oid &5 έχει τρεις εκφάνσεις, για χαμηλή (oid &9), μέτρια (oid &10), και υψηλή (oid &11) κλίμακα. Η έκφραση χαμηλής κλίμακας περιέχει μόνον ένα μικρό σύμβολο (oid &13) που σημειώνει την πόλη στον χάρτη, την θέση (oid &12) του συμβόλου, και το όνομα (oid &14) της πόλης, όπως θα εμφανιστεί στον χάρτη. Η έκφραση υψηλής κλίμακας μπορεί να είναι ένας λεπτομερής χάρτης της ίδιας της πόλης, που να περιέχει πληροφορία για τα σχεδιαγράμματα των δρόμων και των κτιρίων.

Το κληρονομούμενο περιβάλλον των ακμών στην Εικόνα 4.5 αναγράφεται στην δεύτερη γραμμή των ετικετών στις ακμές, με πλάγιους χαρακτήρες. Χρησιμοποιώντας το κληρονομούμενο περιβάλλον των κόμβων και των ακμών είναι εύκολο να διαπιστώσουμε εάν ένας κόμβος ή ακμή υφίσταται κάτω από κάποιον κόσμο. Για παράδειγμα, θεωρώντας τον κόσμο  $w = [scale=low]$ , ο κόμβος με oid &5 υφίσταται κάτω από το  $w$  επειδή το κληρονομούμενο περιβάλλον του είναι  $[scale \text{ in } \{low, med\}]$  και καλύπτει το  $w$ . Ο υπογράφος από την Εικόνα 4.5 (a) που υφίσταται κάτω από το  $w$  απεικονίζεται στην Εικόνα 4.5 (b).

Στην Εικόνα 4.5 (a) παρατηρήστε ότι η ακμή ( $&5$ , [ $scale=high$ ],  $&11$ ) και ο κόμβος  $&11$  (που είναι η έκφανση υψηλής κλίμακας της πόλης με  $oid$   $&5$ ) έχουν σαν κληρονομούμενο περιβάλλον το κενό περιβάλλον. Τα τμήματα του γράφου των οποίων το κληρονομούμενο περιβάλλον είναι το κενό περιβάλλον δεν υφίστανται κάτω από κανέναν κόσμο και δεν περιέχονται σε κανέναν υπογράφο που να υφίσταται κάτω από κάποιον κόσμο. Πρέπει να τονιστεί ότι *τέτοια τμήματα εξακολουθούν να είναι σημαντικά*, καθώς μπορεί να περιέχουν χρήσιμες πληροφορίες. Για παράδειγμα, θεωρήστε την ακόλουθη επερώτηση πάνω στον γράφο που φαίνεται στην Εικόνα 4.5 (a): «δώσε μου στην υψηλότερη λεπτομέρεια ένα σχέδιο της πόλης της οποίας η λεζάντα στον χάρτη χαμηλής κλίμακας είναι Αθήνα». Η απάντηση στην επερώτηση αυτή θα έχει να κάνει με τον κόμβο  $&11$ , το κληρονομούμενο περιβάλλον του οποίου είναι το κενό περιβάλλον.

#### 4.3.1.3 Κληρονομούμενο περιβάλλον της ρίζας

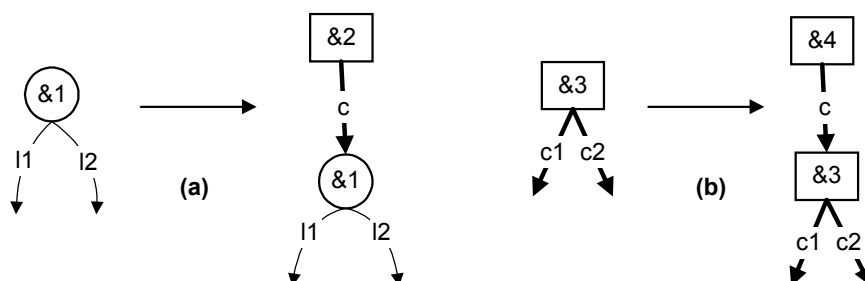
Από τον ορισμό του κληρονομούμενου περιβάλλοντος, εξυπακούεται ότι το κληρονομούμενο περιβάλλον της ρίζας θέτει έναν περιορισμό σε ολόκληρο τον γράφο. Διαισθητικά, περιορίζοντας το κληρονομούμενο περιβάλλον της ρίζας, μπορούμε να «προβάλλουμε» ολόκληρο τον γράφο: αν η ρίζα δεν υφίσταται κάτω από κάποιον κόσμο, τότε κανένας απόγονός της δεν μπορεί να υφίσταται κάτω από τον κόσμο αυτόν. Συνεπώς, δεδομένου ενός Πολυδιάστατου Γράφου Δεδομένων  $G$ , κανένας υπογράφος του  $G$  δεν μπορεί να υφίσταται κάτω από έναν κόσμο που δεν καλύπτεται από το κληρονομούμενο περιβάλλον της ρίζας. Εύκολα βλέπουμε ότι αν περιορίσουμε το κληρονομούμενο περιβάλλον της ρίζας, τότε θα περιοριστούν αντίστοιχα και τα κληρονομούμενα περιβάλλοντα όλων των κόμβων και των ακμών στον γράφο.

##### Πρόταση 4.4

*Αν το κληρονομούμενο περιβάλλον της ρίζας περιοριστεί από  $ic_r$  σε  $ic_r' \subsetneq ic_r$ , τότε το νέο κληρονομούμενο περιβάλλον  $ic_{new}$  οποιουδήποτε κόμβου ή ακμής δίνεται από το  $ic_{new} = ic_{old} \cap ic_r'$ , όπου  $ic_{old}$  είναι το προηγούμενο κληρονομούμενο περιβάλλον αυτού του κόμβου ή της ακμής.*

Αν η ρίζα υφίσταται κάτω από κάθε πιθανό κόσμο, τότε το κληρονομούμενο περιβάλλον της ρίζας είναι το καθολικό περιβάλλον. Είναι βολικό να δεχθούμε την παραδοχή ότι το κληρονομούμενο περιβάλλον της ρίζας είναι πάντοτε το καθολικό περιβάλλον. Η Εικόνα 4.6 δείχνει πως μπορούμε να περιορίσουμε τα κληρονομούμενα περιβάλλοντα σε έναν γράφο, ενώ παράλληλα το κληρονομούμενο περιβάλλον της ρίζας να παραμένει το καθολικό περιβάλλον.

Εικόνα 4.6: Περιορίζοντας το κληρονομούμενο περιβάλλον της ρίζας σε  $c$ .



Στην Εικόνα 4.6 (a) το κληρονομούμενο περιβάλλον της ρίζας με oid &1 είναι [], και θέλουμε να το αλλάξουμε σε c. Προστίθεται ένας νέος πολυδιάστατος κόμβος με oid &2, που δείχνει στον &1 μέσω μιας ακμής περιβάλλοντος η οποία έχει το c σαν ρητό περιβάλλον. Ο νέος κόμβος γίνεται η νέα ρίζα, με κληρονομούμενο περιβάλλον []. Το κληρονομούμενο περιβάλλον της παλαιάς ρίζας (oid &1) είναι τώρα το c. Η Εικόνα 4.6 (b) επιδεικνύει την ίδια μέθοδο στην περίπτωση που η αρχική ρίζα είναι πολυδιάστατος κόμβος.

Από τώρα και στο εξής, θα θεωρούμε ότι το κληρονομούμενο περιβάλλον της ρίζας ενός Πολυδιάστατου Γράφου Δεδομένων είναι το καθολικό περιβάλλον [], εκτός αν δηλώνεται ρητά κάτι διαφορετικό.

### 4.3.2 Κάλυψη περιβάλλοντος

Όπως εξηγήθηκε στην προηγούμενη ενότητα, ένας Πολυδιάστατος Γράφος Δεδομένων μπορεί να εμπεριέχει έναν αριθμό υπογράφων, που είναι και οι ίδιοι Πολυδιάστατοι Γράφοι Δεδομένων, και που καθένας τους υφίσταται κάτω από κάποιον ή κάποιους κόσμους. Στο σημείο αυτό ας θυμηθούμε ότι τα φύλλα σε έναν Πολυδιάστατο Γράφο Δεδομένων μπορούν να είναι κάθε είδος κόμβων, ατομικοί, σύνθετοι, ή πολυδιάστατοι<sup>40</sup>, ενώ τα φύλλα στο OEM είναι αποκλειστικά ατομικοί κόμβοι. Και στα δυο είδη γράφων, οι ατομικοί κόμβοι είναι αυτοί που τελικά φέρουν δεδομένα, συνεπώς η ερώτηση που τίθεται είναι: *κάτω από ποιες συνθήκες είναι τα φύλλα ενός υπογράφου που υφίσταται κάτω από κάποιον κόσμο ατομικοί κόμβοι;*

Προκειμένου να ορίσουμε σωστά μια σχέση ανάμεσα στον Πολυδιάστατο Γράφο Δεδομένων και στο OEM, χρειαζόμαστε κάποιον τρόπο για να αποφασίζουμε εάν ένας κόμβος στον Πολυδιάστατο Γράφο Δεδομένων έχει πρόσβαση σε ατομικούς κόμβους κάτω από κάποιον κόσμο. Αυτό επιτυγχάνεται με την εισαγωγή της έννοιας της **κάλυψης περιβάλλοντος**. Η κάλυψη περιβάλλοντος των κόμβων και των ακμών σχηματίζεται από την μετάδοση των ρητών περιβαλλόντων από τα φύλλα προς την ρίζα του γράφου.

#### Ορισμός 4.8

*Ας είναι το  $G = (V_{mld}, V_{cxb}, E_{cxb}, E_{etb}, r, v)$  ένας Πολυδιάστατος Γράφος Δεδομένων, και το  $p$  ένας κόμβος στο  $V$  που είναι φύλλο. Η κάλυψη περιβάλλοντος ενός φύλλου  $p$  είναι ο προσδιοριστής περιβάλλοντος  $\{-\}$  αν  $p \in V_{mld} \cup V_c$ , και ο  $\{\emptyset\}$  αν  $p \in V_a$ . Ας είναι το  $q$  ένας κόμβος στο  $V$  που δεν είναι φύλλο. Η κάλυψη περιβάλλοντος ενός εσωτερικού κόμβου  $q$  είναι  $cn_q = cn_1 \cup cn_2 \cup \dots \cup cn_n$ , με  $n \geq 1$ , όπου τα  $cn_1, cn_2, \dots, cn_n$  δίνουν την κάλυψη περιβάλλοντος των ακμών στο  $E$  που ξεκινούν από το  $q$ . Ας είναι το  $u$  ένας κόμβος στο  $V$ , το  $cn_u$  η κάλυψη περιβάλλοντος του  $u$ , το  $h$  μια ακμή στο  $E$  που δείχνει στο  $u$ , και το  $ec_h$  το ρητό περιβάλλον του  $h$ . Η κάλυψη περιβάλλοντος της ακμής  $h$  είναι ένας προσδιοριστής περιβάλλοντος  $cn_h$ , τέτοιος ώστε: (a)  $cn_h = cn_u \cap^c ec_h$  και (b) το  $cn_h$  αναπαριστά το ελάχιστο σύνολο κόσμων.*

Η κάλυψη περιβάλλοντος είναι παρόμοια με το κληρονομούμενο περιβάλλον, με τα ρητά περιβάλλοντα να μεταδίδονται όμως προς την αντίθετη κατεύθυνση.

<sup>40</sup> Ακόμη και αν απαιτούσαμε ο Πολυδιάστατος Γράφος Δεδομένων να έχει ατομικούς κόμβους για φύλλα, τίποτε δεν εγγυάται ότι αυτά τα φύλλα θα είναι προσβάσιμα κάτω από κάποιον – οποιονδήποτε – κόσμο. Η προσέγγισή μας είναι να εισάγουμε τον Πολυδιάστατο Γράφο Δεδομένων σαν μια γενική δομή που καλύπτει κάθε περίπτωση, και στην συνέχεια να ορίσουμε το MOEM σαν έναν πιο περιορισμένης μορφής γράφο που έχει πρόσθετες ιδιότητες.

### 4.3.2.1 Υπολογισμός της κάλυψης περιβάλλοντος

Ο υπολογισμός της κάλυψης περιβάλλοντος είναι στην ουσία ο ίδιος με τον υπολογισμό του κληρονομούμενου περιβάλλοντος, που συζητήθηκε στην Ενότητα 4.3.1.1. Οι ακόλουθες προτάσεις σχετικά με την κάλυψη περιβάλλοντος αντιστοιχούν σε αυτές στην Ενότητα 4.3.1.1 σχετικά με το κληρονομούμενο περιβάλλον.

#### Ορισμός 4.9

Ας είναι το  $G = (V_{mld}, V_{cxb}, E_{cxb}, E_{etb}, r, v)$  ένας Πολυδιάστατος Γράφος Δεδομένων, και τα  $g_1, g_2, \dots, g_n$  ακμές στο  $E$  που ορίζουν ένα μονοπάτι  $p$  στο  $G$ . Ας είναι  $cn_1, cn_2, \dots, cn_n$  οι αντίστοιχες καλύψεις περιβάλλοντος των  $g_1, g_2, \dots, g_n$ . Τότε, η **κάλυψη περιβάλλοντος μονοπατιού** του  $p$  είναι  $cn_p = cn_1 \cap cn_2 \cap \dots \cap cn_n$ .

#### Πρόταση 4.5

Η κάλυψη περιβάλλοντος ενός κόμβου  $q$  δίνεται από την ένωση περιβάλλοντος των καλύψεων περιβάλλοντος μονοπατιού όλων των μονοπατιών που ξεκινούν από το  $q$  και καταλήγουν σε φύλλο.

Η Πρόταση 4.5 μπορεί να επαναδιατυπωθεί με ισοδύναμο τρόπο θεωρώντας μόνον τα μονοπάτια που οδηγούν σε ατομικούς κόμβους, μια και τα μονοπάτια τα οποία οδηγούν σε φύλλα που είναι σύνθετοι ή πολυδιάστατοι κόμβοι δεν συνεισφέρουν κανέναν κόσμο στην κάλυψη περιβάλλοντος του  $q$ .

#### Πρόταση 4.6

Ας είναι το  $G = (V_{mld}, V_{cxb}, E_{cxb}, E_{etb}, r, v)$  ένας Πολυδιάστατος Γράφος Δεδομένων, το  $q$  ένας κόμβος στο  $V$  με κάλυψη περιβάλλοντος  $cn_q$ , και το  $p$  ένα μονοπάτι στο  $G$  που οδηγεί στο  $q$ . Ας είναι το  $ec_p$  το ρητό περιβάλλον μονοπατιού του  $p$ . Τότε, η κάλυψη περιβάλλοντος μονοπατιού  $cn_p$  του  $p$  δίνεται από το:  $cn_p = cn_q \cap ec_p$ .

#### Πρόταση 4.7

Η κάλυψη περιβάλλοντος ενός κόμβου είναι η ένωση περιβάλλοντος των καλύψεων περιβάλλοντος μονοπατιού για κάθε μονοπάτι που οδηγεί από τον κόμβο αυτόν σε κάποιον ατομικό κόμβο, όπου η κάλυψη περιβάλλοντος μονοπατιού δίνεται από την τομή περιβάλλοντος όλων των ρητών περιβαλλόντων των ακμών στο μονοπάτι. Ομοίως, για την κάλυψη περιβάλλοντος μιας ακμής θεωρούμε όλα τα μονοπάτια που αρχίζουν με την ακμή αυτή και τελειώνουν σε κάποιον ατομικό κόμβο.

Προφανώς, οι τεχνικές υπολογισμού του κληρονομούμενου περιβάλλοντος μπορούν να επιστρατευτούν και για τον υπολογισμό της κάλυψης περιβάλλοντος. Ένας απλός τρόπος εφαρμογής αυτής της αρχής είναι να αλλάξουμε την κατεύθυνση κάθε ακμής στον γράφο, και να υπολογίσουμε το κληρονομούμενο περιβάλλον για κάθε υπογράφο του οποίου ρίζα είναι κάποιος ατομικός κόμβος του αρχικού γράφου. Με βάση την Πρόταση 4.1 και την Πρόταση 4.5, η ένωση περιβάλλοντος όλων των κληρονομούμενων περιβαλλόντων (ένα για κάθε ατομικό κόμβο) για έναν κόμβο ή μια ακμή θα δώσει την κάλυψη περιβάλλοντος αυτού του κόμβου ή της ακμής.

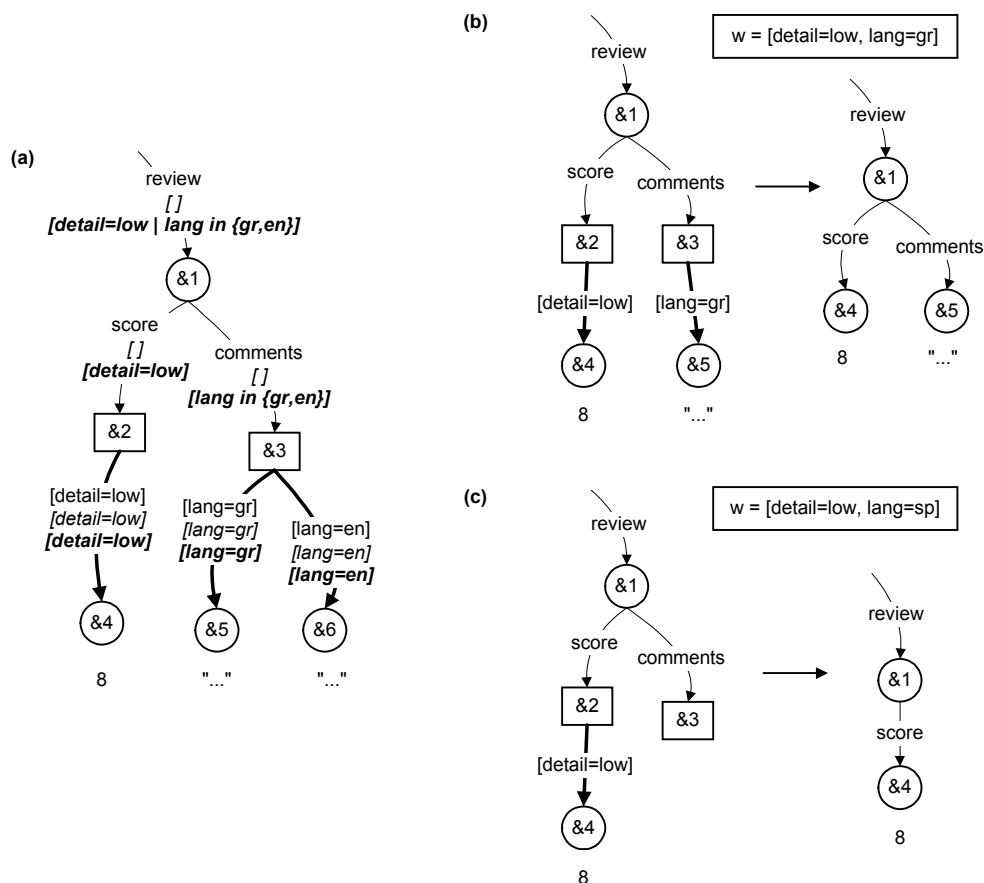
### 4.3.2.2 Σημασία της κάλυψης περιβάλλοντος

Η κάλυψη περιβάλλοντος των φύλλων που είναι σύνθετοι κόμβοι ή πολυδιάστατοι κόμβοι ορίζεται να είναι το κενό περιβάλλον  $\{-\}$  (γράφεται επίσης και  $[-]$ ), ενώ η κάλυψη περιβάλλοντος των φύλλων που είναι ατομικοί κόμβοι ορίζεται να είναι το καθολικό

περιβάλλον  $\{\emptyset\}$  (γράφεται επίσης και  $[\ ]$ ). Όπως είναι προφανές, αυτό κωδικοποιεί τον κανόνα ότι οι ατομικοί κόμβοι μπορούν να είναι φύλλα κάτω από κάθε πιθανό κόσμο, ενώ οι σύνθετοι και οι πολυδιάστατοι κόμβοι δεν μπορούν να είναι φύλλα κάτω από κανέναν κόσμο.

Η Πρόταση 4.7 δείχνει ότι η κάλυψη περιβάλλοντος ενός εσωτερικού κόμβου  $q$  αναπαριστά τους κόσμους κάτω από τους οποίους υπάρχει μονοπάτι από το  $q$  σε κάποιον ατομικό κόμβο. Για να διευκρινισθούν οι λόγοι που αυτό είναι σημαντικό, το παρακάτω παράδειγμα δίνει διαισθητικά περιπτώσεις όπου είναι δυνατόν να «αποσπάσει» κανείς από έναν Πολυδιάστατο Γράφο Δεδομένων ένα συμβατικό OEM που υφίσταται κάτω από κάποιον συγκεκριμένο κόσμο.

**Εικόνα 4.7: Κάλυψη περιβάλλοντος και η σχέση της με την εξαγωγή συμβατικών OEM γράφων.**



Η Εικόνα 4.7 (a) δείχνει μέρος ενός OEM που αναπαριστά την κριτική (review) ενός εστιατορίου. Το αντικείμενο review (oid &1) είναι ένα σύνθετο αντικείμενο που αποτελείται από δυο πολυδιάστατα υποαντικείμενα, τα score (oid &2) και comments (oid &3). Το αντικείμενο score έχει μόνον μια έκφραση (oid &4) που υφίσταται κάτω από χαμηλή λεπτομέρεια, ενώ το comments έχει δυο εκφάνσεις (oid &5 και oid &6) που υφίστανται κάτω από τους κόσμους όπου η γλώσσα είναι Ελληνικά και Αγγλικά, αντίστοιχα. Τα κληρονομούμενα περιβάλλοντα των ακμών αναγράφονται με πλάγιους χαρακτήρες στην

δεύτερη γραμμή των ετικετών, και οι καλύψεις περιβάλλοντός των αναγράφονται με έντονους πλάγιους χαρακτήρες στην τρίτη γραμμή των ετικετών.

Ο Πολυδιάστατος Γράφος Δεδομένων στο αριστερό μέρος στην Εικόνα 4.7 (b) είναι ο υπογράφος του (a) που υφίσταται κάτω από τον κόσμο  $w = [\text{detail}=\text{low}, \text{lang}=\text{gr}]$ , όπως συζητήθηκε στην Ενότητα 4.3.1.2: οι κόμβοι και οι ακμές του γράφου στο (a) των οποίων το κληρονομούμενο περιβάλλον καλύπτει το  $w$  απαρτίζουν τον υπογράφο στο (b). Εύκολα βλέπει κανείς ότι από αυτόν τον υπογράφο μπορούμε να παραλείψουμε τους πολυδιάστατους κόμβους και τις ακμές περιβάλλοντος και να πάρουμε ένα συμβατικό OEM που υφίσταται κάτω από το  $w$ , το οποίο απεικονίζεται στο δεξί μέρος στην Εικόνα 4.7 (b). Μια παρόμοια περίπτωση έχουμε αν θεωρήσουμε τον κόσμο όπου η λεπτομέρεια είναι ξανά χαμηλή, και η γλώσσα είναι Αγγλικά αντί για Ελληνικά. Όμως, αν τα Ισπανικά είναι μια από τις πιθανές γλώσσες και θεωρήσουμε τον κόσμο  $w' = [\text{detail}=\text{high}, \text{lang}=\text{sp}]$ , δεν είναι δυνατόν να εξάγουμε ένα OEM που να υφίσταται κάτω από το  $w'$ . Ο λόγος είναι ότι όλα τα φύλλα του Πολυδιάστατου (υπο-) Γράφου Δεδομένων που υφίστανται κάτω από το  $w'$  είναι πολυδιάστατοι κόμβοι, και αν οι κόμβοι αυτοί παραλειφθούν στην διαδικασία εξαγωγής ενός OEM το σύνθετο αντικείμενο `review` με `oid &1` θα μείνει χωρίς παιδιά. Στο σημείο αυτό, παρατηρήστε ότι η κάλυψη περιβάλλοντος του αντικειμένου `review` στον αρχικό γράφο στην Εικόνα 4.7 (a) περιέχει τους κόσμους κάτω από τους οποίους είναι δυνατόν να εξαχθεί κάποιο OEM, αλλά δεν περιέχει τον κόσμο  $w'$ , κάτω από τον οποίο δεν είναι δυνατόν να εξαχθεί κανένα OEM.

Μια ενδιαφέρουσα περίπτωση φαίνεται στην Εικόνα 4.7 (c). Κάτω από τον κόσμο  $w = [\text{detail}=\text{low}, \text{lang}=\text{sp}]$  τα φύλλα του γράφου στα αριστερά είναι ανάμικτα: το ένα είναι ατομικός κόμβος (`oid &4`) ενώ το άλλο είναι πολυδιάστατος κόμβος (`oid &3`)<sup>41</sup>. Μπορούμε να αντιμετωπίσουμε την περίπτωση αυτή με δυο τρόπους:

- (a) *Προσέγγιση αυστηρών τύπων*: Απαιτεί το αντικείμενο `review` να έχει την ίδια δομή (τύπο) κάτω από κάθε πιθανό κόσμο. Με βάση αυτή την συνθήκη ο γράφος στα αριστερά στην Εικόνα 4.7 (c) δεν μπορεί να δώσει κάποιο OEM.
- (b) *Προσέγγιση μεταβλητής δομής*: Χαλαρώνει τους περιορισμούς των αυστηρών τύπων, και επιτρέπει στο αντικείμενο `review` να έχει διαφορετική δομή κάτω από διαφορετικούς κόσμους. Στην περίπτωση αυτή είναι δυνατόν να εξαχθεί ένα OEM που να υφίσταται κάτω από το  $w$ , το οποίο και απεικονίζεται στο δεξί μέρος στην Εικόνα 4.7 (c).

Η προσέγγιση που υιοθετήσαμε είναι η προσέγγιση μεταβλητής δομής, επειδή προσιδιάζει στην μεταβλητή δομή που επιδεικνύουν τα ημιδομημένα δεδομένα. Θυμηθείτε ότι εκφάνσεις της ίδιας πολυδιάστατης οντότητας μπορεί να έχουν διαφορετική δομή· συνεπώς ένα πολυδιάστατο αντικείμενο μπορεί ήδη να έχει διαφορετικούς τύπους κάτω από διαφορετικούς κόσμους. Επιπρόσθετα, η προσέγγιση που υιοθετήσαμε συσχετίζει την δομή ενός αντικειμένου κάτω από έναν κόσμο με τα πολυδιάστατα υποαντικείμενά του, και παρέχει έναν εναλλακτικό, πιο οικονομικό<sup>42</sup> τρόπο για να μεταβάλλουμε την δομή κάτω από

<sup>41</sup> Είναι δυνατόν να έχουμε σύνθετους κόμβους σαν φύλλα σε έναν υπογράφο που υφίσταται κάτω από κάποιον κόσμο, αλλά μόνον αν αυτοί οι σύνθετοι κόμβοι εμφανίζονται σαν φύλλα και στον αρχικό Πολυδιάστατο Γράφο Δεδομένων. Λόγω του ορισμού του κληρονομούμενου περιβάλλοντος, ένας σύνθετος κόμβος που είναι εσωτερικός στον αρχικό Πολυδιάστατο Γράφο Δεδομένων δεν μπορεί να εμφανιστεί σαν φύλλο σε έναν υπογράφο που υφίσταται κάτω από κάποιον κόσμο. Οι σύνθετοι κόμβοι που εμφανίζονται σαν φύλλα σε υπογράφους που υφίστανται κάτω από κάποιον κόσμο αντιμετωπίζονται με τον ίδιο τρόπο όπως οι πολυδιάστατοι κόμβοι που εμφανίζονται σαν φύλλα (για παράδειγμα, δείτε τον κόμβο `&3` στην Εικόνα 4.7 (c)).

<sup>42</sup> Στον γράφο στην Εικόνα 4.7 (a), αντί να αλλάξουμε την δομή του `review` μέσω των πολυδιάστατων υποαντικειμένων του, θα μπορούσαμε να έχουμε κάνει το `review` ένα πολυδιάστατο

διαφορετικούς κόσμους. Οι κόσμοι κάτω από τους οποίους ένας κόμβος περιβάλλοντος  $u$  δεν μπορεί να περιλάβει στην δομή του ένα κόμβο-παιδί  $q$  λόγω μη πρόσβασης του  $q$  σε ατομικούς κόμβους, δίνονται από το  $ic_u -^c cn_q$ , όπου το  $ic_u$  είναι το κληρονομούμενο περιβάλλον του  $u$ , και το  $cn_q$  είναι η κάλυψη περιβάλλοντος του  $q$ .

Παρόλο που η προσέγγιση αυστηρών τύπων πιθανώς δεν είναι η πιο κατάλληλη για τα MSSD, είναι μια απολύτως έγκυρη προσέγγιση, και η πιο φυσική να ακολουθηθεί κανείς για (εντελώς) δομημένα δεδομένα. Προκειμένου να συμμορφωθεί με την προσέγγιση αυστηρών τύπων, ο Ορισμός 4.8 της κάλυψης περιβάλλοντος θα πρέπει να μετατραπεί ως εξής: η κάλυψη περιβάλλοντος  $cn_q$  ενός εσωτερικού σύνθετου κόμβου  $q$  θα πρέπει να δίνεται από το  $cn_1 \cap^c cn_2 \cap^c \dots \cap^c cn_n$  αντί από το  $cn_1 \cup^c cn_2 \cup^c \dots \cup^c cn_n$ , και συνεπώς να χρησιμοποιεί την τομή περιβάλλοντος και όχι την ένωση περιβάλλοντος της κάλυψης περιβάλλοντος των απερχόμενων ακμών οντότητας. Η προσέγγιση αυστηρών τύπων αξίζει περισσότερη προσοχή, όμως στα πλαίσια των MSSD θα εστιάσουμε στην προσέγγιση μεταβλητής δομής.

*Συμπερασματικά, η κάλυψη περιβάλλοντος ενός κόμβου περιβάλλοντος ή μιας ακμής οντότητας αναπαριστά το σύνολο κόσμων κάτω από τους οποίους αυτός ο κόμβος ή η ακμή μπορεί να συμμετέχει σε εξαχθέντα OEMs, όπως επιβάλλεται από την προσβασιμότητα σε ατομικούς κόμβους.*

### 4.3.2.3 Κάλυψη περιβάλλοντος της ρίζας

Η ακόλουθη πρόταση δίνει την κάλυψη περιβάλλοντος της ρίζας απευθείας από το κληρονομούμενο περιβάλλον των ατομικών κόμβων.

#### Πρόταση 4.8

*Ας είναι το  $G$  ένας Πολυδιάστατος Γράφος Δεδομένων. Η κάλυψη περιβάλλοντος της ρίζας του  $G$  δίνεται από την ένωση περιβάλλοντος των κληρονομούμενων περιβαλλόντων των ατομικών κόμβων στο  $G$ .*

Για έναν κόσμο  $w$  που περιέχεται στην κάλυψη περιβάλλοντος της ρίζας, υπάρχουν ατομικοί κόμβοι που είναι προσβάσιμοι κάτω από το  $w$ , και είναι δυνατόν να εξάγουμε ένα OEM που υφίσταται κάτω από το  $w$ . Από την άλλη, εάν το  $w$  δεν περιέχεται στην κάλυψη περιβάλλοντος της ρίζας, ο υπογράφος που υφίσταται κάτω από το  $w$  δεν μπορεί να δώσει ένα σωστό OEM, μια και η ρίζα δεν θα έχει πρόσβαση σε κανέναν ατομικό κόμβο. Συνεπώς, η κάλυψη περιβάλλοντος της ρίζας ενός Πολυδιάστατου Γράφου Δεδομένων  $G$  αναπαριστά τους κόσμους κάτω από τους οποίους είναι δυνατόν να εξάγουμε OEMs από το  $G$ .

Μια παραδοχή που υπονοείται στην παραπάνω πρόταση είναι ότι, όπως εξηγήθηκε στην Ενότητα 4.3.1.3, το κληρονομούμενο περιβάλλον της ρίζας είναι το καθολικό περιβάλλον. Αν τα πράγματα δεν είναι έτσι, οι κόσμοι κάτω από τους οποίους είναι δυνατόν να εξαχθούν OEMs δίνονται από το  $ic_r \cap^c cn_r$ , όπου  $ic_r$  είναι το κληρονομούμενο περιβάλλον της ρίζας και  $cn_r$  είναι η κάλυψη περιβάλλοντος της ρίζας. Η έκφραση αυτή, όπως θα δούμε στην επόμενη ενότητα, καλείται **κληρονομούμενη κάλυψη** και έχει να κάνει όχι μόνο με την ρίζα, αλλά και με κάθε κόμβο και ακμή σε έναν Πολυδιάστατο Γράφο Δεδομένων.

### 4.3.3 Κληρονομούμενη κάλυψη

Όπως έχουμε δει, το κληρονομούμενο περιβάλλον ενός κόμβου ή ακμής εκφράζει συσσωρευμένους περιορισμούς όπως αυτοί επιβάλλονται από τους προγόνους στον γράφο,

---

αντικείμενο με εκφάνσεις που να έχουν διαφορετικές δομές και που να δείχνουν κατευθείαν στους κατάλληλους κόμβους περιβάλλοντος.



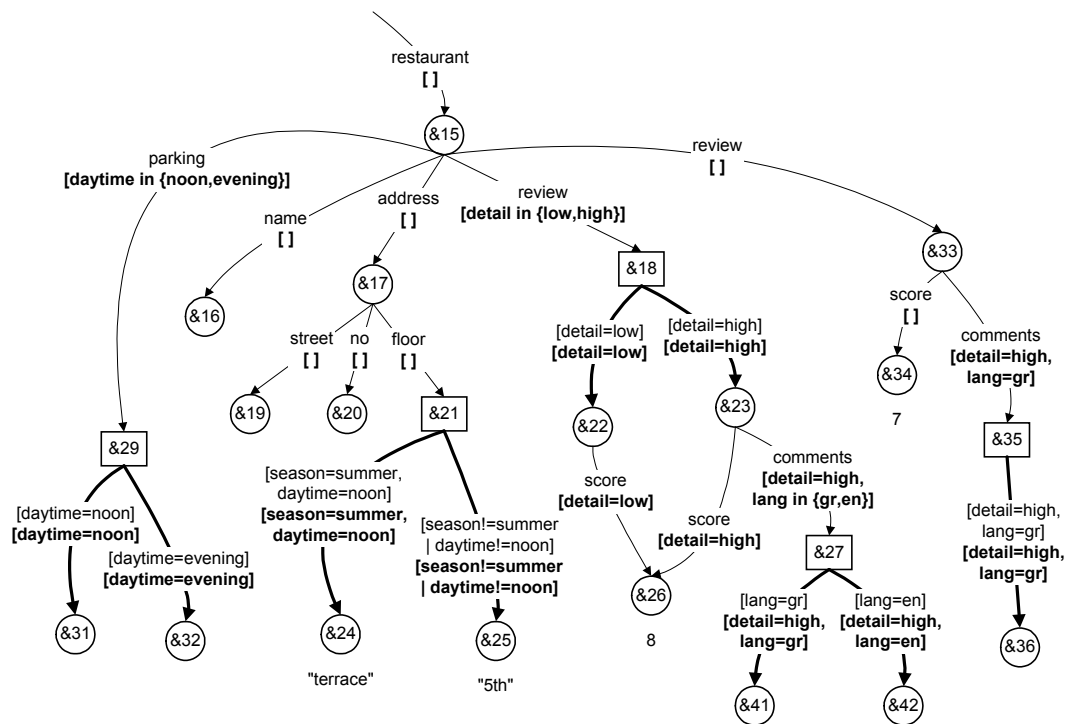
ενώ η κάλυψη περιβάλλοντος εκφράζει συσσωρευμένους περιορισμούς όπως αυτοί επιβάλλονται από τους απογόνους. Η τομή περιβάλλοντος των δυο δίνει την κληρονομούμενη κάλυψη ενός κόμβου ή μιας ακμής.

#### Ορισμός 4.10

*Η κληρονομούμε κάλυψη ενός κόμβου ή ακμής σε έναν Πολυδιάστατο Γράφο Δεδομένων είναι  $icv = ic \cap^c cn$ , όπου  $ic$  είναι το κληρονομούμενο περιβάλλον και  $cn$  είναι η κάλυψη περιβάλλοντος αυτού του κόμβου ή της ακμής.*

Συνεπώς, η κληρονομούμενη κάλυψη ενός κόμβου ή μιας ακμής αναπαριστά τους κόσμους κάτω από τους οποίους έχει νόημα να υπάρχει αυτός ο κόμβος ή η ακμή. Κάτω από κάθε τέτοιο κόσμο, υφίσταται ένα μονοπάτι που ξεκινά από την ρίζα, καταλήγει σε κάποιον ατομικό κόμβο, και περιλαμβάνει αυτόν τον κόμβο ή την ακμή.

Εικόνα 4.8: Γράφος επισημειωμένος με την κληρονομούμενη κάλυψη των ακμών.



Η Εικόνα 4.8 δείχνει τμήμα του Πολυδιάστατου Γράφου Δεδομένων που απεικονίζεται στην Εικόνα 4.3 επισημειωμένο με τις κληρονομούμενες καλύψεις των ακμών, γραμμένες με έντονους χαρακτήρες κάτω από την κανονική ετικέτα. Σημειώστε ότι όλα τα φύλλα στην Εικόνα 4.8 θεωρούνται ατομικοί κόμβοι, παρόλο που μερικές από τις τιμές τους έχουν παραλειφθεί για απλότητα.

Η προσβασιμότητα σε κάποιον ατομικό κόμβο είναι απαραίτητη προϋπόθεση προκειμένου ένας κόμβος ή μια ακμή να έχει νόημα κάτω από έναν κόσμο. Συνεπώς, στα ακόλουθα θα θεωρούμε την κληρονομούμενη κάλυψη αντί για το κληρονομούμενο περιβάλλον για να αποφασίζουμε τους κόσμους κάτω από τους οποίους υφίσταται πραγματικά ένας κόμβος ή μια ακμή. Συγκεκριμένα, ένας κόμβος ή μια ακμή υφίσταται κάτω από έναν κόσμο  $w$  αν η κληρονομούμενη κάλυψη αυτού του κόμβου ή της ακμής περιέχει τον  $w$ . Συνάγεται ότι ένας

κόμβος ή μια ακμή υφίσταται κάτω από ένα περιβάλλον  $c$  αν η κληρονομούμενη κάλυψη του είναι υπερσύνολο περιβάλλοντος του  $c$ .

Σημειώστε ότι η κληρονομούμενη κάλυψη είναι υποσύνολο περιβάλλοντος του κληρονομούμενου περιβάλλοντος, συνεπώς οι κόσμοι κάτω από τους οποίους υφίσταται ένας κόμβος ή μια ακμή σε σχέση με την κληρονομούμενη κάλυψη του αποτελούν υποσύνολο των κόσμων κάτω από τους οποίους υφίσταται αυτός ο κόμβος ή η ακμή σε σχέση με το κληρονομούμενο περιβάλλον του.

#### 4.3.3.1 Κληρονομούμενη κάλυψη μονοπατιού

Σε αναλογία με το κληρονομούμενο περιβάλλον μονοπατιού και με την κάλυψη περιβάλλοντος μονοπατιού, εισάγουμε ακολούθως την **κληρονομούμενη κάλυψη μονοπατιού**.

##### Ορισμός 4.11

*Η κληρονομούμενη κάλυψη μονοπατιού ενός μονοπατιού  $p$  σε έναν Πολυδιάστατο Γράφο Δεδομένων είναι  $icv_p = ic_p \cap^c cv_p$ , όπου  $ic_p$  είναι το κληρονομούμενο περιβάλλον μονοπατιού του  $p$  και  $cv_p$  είναι η κάλυψη περιβάλλοντος μονοπατιού του  $p$ .*

Οι παρακάτω προτάσεις δίνουν εναλλακτικούς τρόπους υπολογισμού της κληρονομούμενης κάλυψης μονοπατιού.

##### Πρόταση 4.9

*Ας είναι το  $G = (V_{mlb}, V_{cxb}, E_{cxb}, E_{etb}, r, v)$  ένας Πολυδιάστατος Γράφος Δεδομένων, και τα  $g_1, g_2, \dots, g_n$  ακμές στο  $E$  που ορίζουν ένα μονοπάτι  $p$  στο  $G$ . Ας είναι  $icv_1, icv_2, \dots, icv_n$  οι αντίστοιχες κληρονομούμενες καλύψεις των  $g_1, g_2, \dots, g_n$ . Τότε, η κληρονομούμενη κάλυψη μονοπατιού του  $p$  δίνεται από το  $icv_p = icv_1 \cap^c icv_2 \cap^c \dots \cap^c icv_n$ .*

##### Πρόταση 4.10

*Ας είναι το  $G = (V_{mlb}, V_{cxb}, E_{cxb}, E_{etb}, r, v)$  ένας Πολυδιάστατος Γράφος Δεδομένων, τα  $q$  και  $u$  κόμβοι στο  $V$ , και το  $p$  ένα μονοπάτι στο  $G$  που αρχίζει από το  $q$  και καταλήγει στο  $u$ . Ας είναι  $ic_q$  το κληρονομούμενο περιβάλλον του  $q$ ,  $cv_u$  η κάλυψη περιβάλλοντος του  $u$ , και  $ec_p$  το ρητό περιβάλλον μονοπατιού του  $p$ . Τότε, η κληρονομούμενη κάλυψη μονοπατιού του  $p$  δίνεται από το  $icv_p = ic_q \cap^c cv_u \cap^c ec_p$ .*

Παρατηρήστε ότι δεδομένου ενός μονοπατιού  $p$  με κληρονομούμενη κάλυψη μονοπατιού  $icv_p$ , λόγω των όσων περιέχει ο Ορισμός 4.6 και ο Ορισμός 4.8, η κληρονομούμενη κάλυψη κάθε κόμβου στο  $p$  (του πρώτου και του τελευταίου κόμβου συμπεριλαμβανομένων) είναι υπερσύνολο περιβάλλοντος του  $icv_p$ . Συνεπώς, η κληρονομούμενη κάλυψη μονοπατιού αναπαριστά τους κόσμους κάτω από τους οποίους υφίστανται πραγματικά όλοι οι κόμβοι και οι ακμές στο μονοπάτι.

#### 4.3.3.2 Σημασία της κληρονομούμενης κάλυψης

Όπως δείξαμε στην Ενότητα 4.3.1.2, οι κόμβοι και οι ακμές ενός Πολυδιάστατου Γράφου Δεδομένων  $G$  που υφίστανται κάτω από έναν κόσμο  $w$  σε σχέση με τα κληρονομούμενα περιβάλλοντά τους, σχηματίζουν έναν υπογράφο του  $G$  με την ίδια ρίζα όπως το  $G$  ο οποίος υφίσταται κάτω από το  $w$ . Προφανώς, αυτός ο υπογράφος συρρικνώνεται περισσότερο αν πάρουμε τους κόμβους και τις ακμές που υφίστανται στην πραγματικότητα κάτω από το  $w$  (σε

σχέση με τις κληρονομούμενες καλύψεις τους). Ας συζητήσουμε τις ιδιότητες αυτού του συρρικνωμένου υπογράφου  $g$ .

- (a) Αν ένας κόμβος  $q$  ανήκει στο  $g$ , τότε η ρίζα  $r$  του  $G$  ανήκει επίσης στο  $g$ . Επιπλέον, υπάρχει ένα μονοπάτι στο  $g$  που οδηγεί από το  $r$  στο  $q$ . Να γιατί: Αν το  $q$  ανήκει στο  $g$ , τότε το κληρονομούμενο περιβάλλον του  $ic_q$  και η κάλυψη περιβάλλοντος του  $cn_q$  περιέχουν το  $w$ . Από την Πρόταση 4.3, εφόσον το  $ic_q$  περιέχει το  $w$ , υπάρχει ένα μονοπάτι  $p$  στο  $G$  από την ρίζα στο  $q$  τέτοιο ώστε το  $ic_r \cap^c ec_1 \cap^c ec_2 \cap^c \dots \cap^c ec_n$  να περιέχει το  $w$ , όπου  $ec_1, ec_2, \dots, ec_n$  είναι τα ρητά περιβάλλοντα των ακμών στο μονοπάτι. Τότε, το  $w$  περιέχεται επίσης στο  $cn_q \cap^c ic_r \cap^c ec_1 \cap^c ec_2 \cap^c \dots \cap^c ec_n$ , που σύμφωνα με την Πρόταση 4.10 είναι η κληρονομούμενη κάλυψη μονοπατιού του  $p$ . Σύμφωνα με την Πρόταση 4.9, η κληρονομούμενη κάλυψη κάθε ακμής στο  $p$  περιέχει το  $w$ , συνεπώς κάθε ακμή και κόμβος στο  $p$  είναι μέρος του  $g$ . Εξυπακούεται ότι η ρίζα  $r$  του  $G$  είναι και η ρίζα του  $g$ .
- (b) Αν μια ακμή  $(q, l, u)$  ανήκει στο  $g$ , τότε οι κόμβοι  $q$  και  $u$  θα ανήκουν και αυτοί στο  $g$ . Αυτό προκύπτει από όσα αναφέρει ο Ορισμός 4.8 της κάλυψης περιβάλλοντος.
- (c) Τα φύλλα στο  $g$  είναι αποκλειστικά ατομικοί κόμβοι. Συγκεκριμένα, είναι οι ατομικοί κόμβοι του  $G$  των οποίων το κληρονομούμενο περιβάλλον περιέχει το  $w$ . Για να γίνει κατανοητός ο λόγος, ας θυμηθούμε ότι οι πολυδιάστατοι κόμβοι και οι σύνθετοι κόμβοι που είναι φύλλα στο  $G$  δεν μπορούν να είναι μέρος του  $g$ , αφότου οι καλύψεις περιβάλλοντός τους δεν περιέχουν κανέναν κόσμο. Ας θεωρήσουμε ότι ένας πολυδιάστατος κόμβος  $q$  που δεν είναι φύλλο στο  $G$  εμφανίζεται σαν φύλλο στο  $g$ . Μια και το  $q$  είναι μέρος του  $g$ , το κληρονομούμενο περιβάλλον του  $q$  περιέχει το  $w$ , συνεπώς τα κληρονομούμενα περιβάλλοντα των ακμών που ξεκινούν από το  $q$  περιέχουν επίσης το  $w$ . Συνεπώς, ο λόγος για τον οποίο οι ακμές που ξεκινούν από το  $q$  δεν είναι μέρος του  $g$  είναι ότι οι καλύψεις περιβάλλοντός τους δεν περιέχουν το  $w$ . Στην περίπτωση αυτή όμως, ούτε η κάλυψη περιβάλλοντος του  $q$  θα περιείχε το  $w$ , και το  $q$  δεν θα ήταν μέρος του  $g$ . Για τους ίδιους λόγους, ένας σύνθετος κόμβος που είναι εσωτερικός στο  $G$  δεν μπορεί να εμφανιστεί σαν φύλλο στο  $g$ .

Συμπερασματικά, τα στοιχεία ενός Πολυδιάστατου Γράφου Δεδομένων  $G$  που υφίστανται κάτω από έναν κόσμο  $w$  (σε σχέση με την κληρονομούμενη κάλυψή τους), σχηματίζουν έναν υπογράφο, ο οποίος είναι και ο ίδιος Πολυδιάστατος Γράφος Δεδομένων που έχει την ίδια ρίζα όπως το  $G$  και του οποίου τα φύλλα είναι αποκλειστικά ατομικοί κόμβοι<sup>43</sup>. Στην Ενότητα 4.4.1, όπου συζητούμε την διεργασία της **αναγωγής σε OEM**, θα δούμε ότι είναι πάντοτε δυνατόν να εξάγουμε από έναν τέτοιο υπογράφο ένα συμβατικό OEM που υφίσταται κάτω από το  $w$ .

Είναι εύκολο να δει κανείς ότι το παραπάνω δεν γενικεύεται για περιβάλλοντα που αναπαριστούν περισσότερους από έναν κόσμους. Συγκεκριμένα, οι ιδιότητες (a) και (c) που αποδείχθηκαν πιο πάνω για έναν κόσμο δεν ισχύουν όταν θεωρήσουμε περιβάλλοντα που αναπαριστούν περισσότερους από έναν κόσμους. Η περίπτωση της αναγωγής σε OEM κάτω από περισσότερους του ενός κόσμων συζητείται λεπτομερώς στην Ενότητα 4.4.1.3.

Το νόημα της κληρονομούμενης κάλυψης είναι πλέον ξεκάθαρο. *Η κληρονομούμενη κάλυψη ενός κόμβου ή μιας ακμής σε έναν Πολυδιάστατο Γράφο Δεδομένων  $G$  αναπαριστά τους κόσμους κάτω από τους οποίους ο κόμβος ή η ακμή λαμβάνεται υπόψη κατά την εξαγωγή συμβατικών OEMs από το  $G$ .* Συγκεκριμένα, η κληρονομούμενη κάλυψη ενός κόμβου

<sup>43</sup> Η τελευταία αυτή ιδιότητα κάνει όλη την διαφορά από τους υπογράφους που είδαμε στην Ενότητα 4.3.1.2, οι οποίοι σχηματίζονται με βάση το κληρονομούμενο περιβάλλον αντί για την κληρονομούμενη κάλυψη.

περιβάλλοντος δίνει τους κόσμους κάτω από τους οποίους ο κόμβος αποτελεί μέρος ενός συμβατικού OEM. Να γιατί: ένας κόμβος  $q$  που δεν έχει πρόσβαση σε κανέναν ατομικό κόμβο κάτω από έναν κόσμο  $w$  δεν μπορεί να είναι μέρος ενός OEM που υφίσταται κάτω από το  $w$ . Έτσι, μπορούν να εξεταστούν μόνον κόσμοι που περιέχονται στην κληρονομούμενη κάλυψή του. Επιπλέον, όπως δείξαμε στην παρατήρηση (a) πιο πάνω, για κάθε τέτοιο κόσμο  $w$ , υπάρχει ένα μονοπάτι από την ρίζα στο  $q$  που υφίσταται κάτω από το  $w$ , και η κληρονομούμενη κάλυψη της ρίζας περιέχει επίσης το  $w$ . Ομοίως, η κληρονομούμενη κάλυψη μιας ακμής οντότητας δίνει τους κόσμους κάτω από τους οποίους η ακμή αποτελεί μέρος ενός συμβατικού OEM.

### 4.3.3.3 Περιβαλλοντική προβολή γράφου

Στην Πρόταση 4.4 δείξαμε ότι εάν περιορίσουμε το κληρονομούμενο περιβάλλον της ρίζας, τότε περιορίζονται με αντίστοιχο τρόπο τα κληρονομούμενα περιβάλλοντα όλων των κόμβων και των ακμών στον γράφο. Οι καλύψεις περιβάλλοντος των κόμβων και των ακμών δεν επηρεάζονται στην περίπτωση αυτή, ενώ οι κληρονομούμενες καλύψεις των κόμβων και των ακμών περιορίζονται αντίστοιχα, όπως ακριβώς τα κληρονομούμενα περιβάλλοντα.

#### Πρόταση 4.11

*Αν το κληρονομούμενο περιβάλλον της ρίζας περιοριστεί από  $ic_r$  σε  $ic_r' \subsetneq ic_r$ , τότε η νέα κληρονομούμενη κάλυψη  $icv_{new}$  κάθε κόμβου ή ακμής δίνεται από το  $icv_{new} = icv_{old} \cap ic_r'$ , όπου  $icv_{old}$  είναι η προηγούμενη κληρονομούμενη κάλυψη αυτού του κόμβου ή της ακμής.*

Συνεπώς, περιορίζοντας το κληρονομούμενο περιβάλλον της ρίζας όπως απεικονίζει η Εικόνα 4.6 στην πραγματικότητα περιορίζουμε τους κόσμους κάτω από τους οποίους υφίστανται οι κόμβοι και οι ακμές, και για τους οποίους είναι δυνατόν να εξαχθούν OEMs. Άρα τελικά, *προβάλλουμε τον γράφο* στους κόσμους που αναπαριστά το νέο κληρονομούμενο περιβάλλον της ρίζας. Έτσι, η **περιβαλλοντική προβολή γράφου** ενός Πολυδιάστατου Γράφου Δεδομένων σε ένα περιβάλλον  $c$  περιλαμβάνει δυο βήματα:

- Περιορισμός του κληρονομούμενου περιβάλλοντος της (αρχικής) ρίζας σε  $c$ , όπως δείχνει η Εικόνα 4.6.
- Προσαρμογή των κληρονομούμενων καλύψεων των κόμβων και των ακμών στον γράφο, όπως υποδεικνύει η Πρόταση 4.11.

Η περιβαλλοντική προβολή γράφου είναι παρόμοια με την διεργασία της **μερικής αναγωγής**, την οποία συζητούμε στην Ενότητα 4.4.2. Η μερική αναγωγή πηγαίνει μακρύτερα και απομακρύνει από τον γράφο της προβολής κόμβους και ακμές που δεν υφίστανται κάτω από κανέναν κόσμο.

## 4.4 ΑΝΑΓΩΓΗ

Είναι προφανές ότι ένας Πολυδιάστατος Γράφος Δεδομένων μπορεί να συμπεριλαμβάνει έναν αριθμό συμβατικών OEMs που υφίστανται κάτω από διαφορετικούς κόσμους. Σε προηγούμενες ενότητες συζητήσαμε τις συνθήκες που είναι *αναγκαίες* και *ικανές* προκειμένου να είναι δυνατή η εξαγωγή OEM που υφίσταται κάτω από κάποιον κόσμο. Συγκεκριμένα, δείξαμε ότι μπορούμε να εξάγουμε OEMs μόνον για τους κόσμους που αναπαριστά η κληρονομούμενη κάλυψη της ρίζας. Επιπρόσθετα, δείξαμε ότι η κληρονομούμενη κάλυψη ενός κόμβου περιβάλλοντος ή μιας ακμής οντότητας αναπαριστά

τους κόσμους κάτω από τους οποίους αυτός ο κόμβος ή η ακμή αποτελεί τμήμα κάποιου OEM που είναι ενσωματωμένο στον γράφο.

Η αναγωγή είναι μια διεργασία που απομακρύνει κάποιους από τους κόμβους και τις ακμές ενός Πολυδιάστατου Γράφου Δεδομένων, δίνοντας έναν υπογράφο του οποίου τα στοιχεία ικανοποιούν κάποια κριτήρια. Τα κριτήρια αυτά έχουν να κάνουν με την κληρονομούμενη κάλυψη, συνεπώς όταν αναφερόμαστε σε έναν Πολυδιάστατο Γράφο Δεδομένων θεωρούμε ότι οι κληρονομούμενες καλύψεις των κόμβων και των ακμών του έχουν υπολογιστεί και είναι γνωστές. Οι κληρονομούμενες καλύψεις πρέπει να επαναυπολογίζονται μόνον όταν ο γράφος έχει υποστεί αλλαγές, ενώ οι αναγωγές μπορούν να λάβουν χώρα οποιαδήποτε στιγμή, για παράδειγμα μετά από μια αίτηση κάποιου χρήστη. Στην ενότητα αυτή, προδιαγράφουμε έναν αλγόριθμο που εκτελεί **αναγωγή σε OEM**, το οποίο υφίσταται κάτω από δεδομένο κόσμο, ενός Πολυδιάστατου Γράφου Δεδομένων. Εξηγούμε επίσης την **μερική αναγωγή**, η οποία ανάγει έναν Πολυδιάστατο Γράφο Δεδομένων σε έναν νέο Πολυδιάστατο υπο-Γράφο Δεδομένων με ενδιαφέρουσες ιδιότητες. Στην συνέχεια δείχνουμε ότι και οι δυο τύποι αναγωγής μπορούν να αποσυντεθούν στις ίδιες στοιχειώδεις πράξεις.

#### 4.4.1 Αναγωγή σε OEM

Για να ανάγουμε έναν Πολυδιάστατο Γράφο Δεδομένων  $G$  σε ένα OEM που υφίσταται κάτω από κάποιον κόσμο, είναι σημαντικό να γνωρίζουμε εάν το  $G$  είναι ένας **αιτιοκρατικός ως προς το περιβάλλον γράφος**. Η επόμενη ενότητα ορίζει την έννοια αυτή: η ενότητα που ακολουθεί παρουσιάζει την **αναγωγή σε OEM** κάτω από κάποιον κόσμο, ενώ η τελευταία ενότητα συζητά OEMs που υφίστανται κάτω από περισσότερους του ενός κόσμων.

##### 4.4.1.1 Γράφοι αιτιοκρατικοί ως προς το περιβάλλον

Όπως έχουμε δει, είναι δυνατόν για μια πολυδιάστατη οντότητα να ενσωματώνει περισσότερες από μια εκφάνσεις που να υφίστανται κάτω από τον ίδιο κόσμο. Η έννοια της αιτιοκρατίας ως προς το περιβάλλον θέτει κάποιους περιορισμούς στο ζήτημα αυτό. Συγκεκριμένα, στους αιτιοκρατικούς ως προς το περιβάλλον Πολυδιάστατους Γράφους Δεδομένων οι εκφάνσεις της ίδιας πολυδιάστατης οντότητας που είναι συμβατικές πληροφοριακές οντότητες (με άλλα λόγια, κόμβοι περιβάλλοντος) πρέπει να είναι προσβάσιμες κάτω από ξένα μεταξύ τους σύνολα κόσμων. Αυτό, πάντως, εφαρμόζεται άμεσα μόνον σε κάποιες από τις πολυδιάστατες οντότητες στον γράφο.

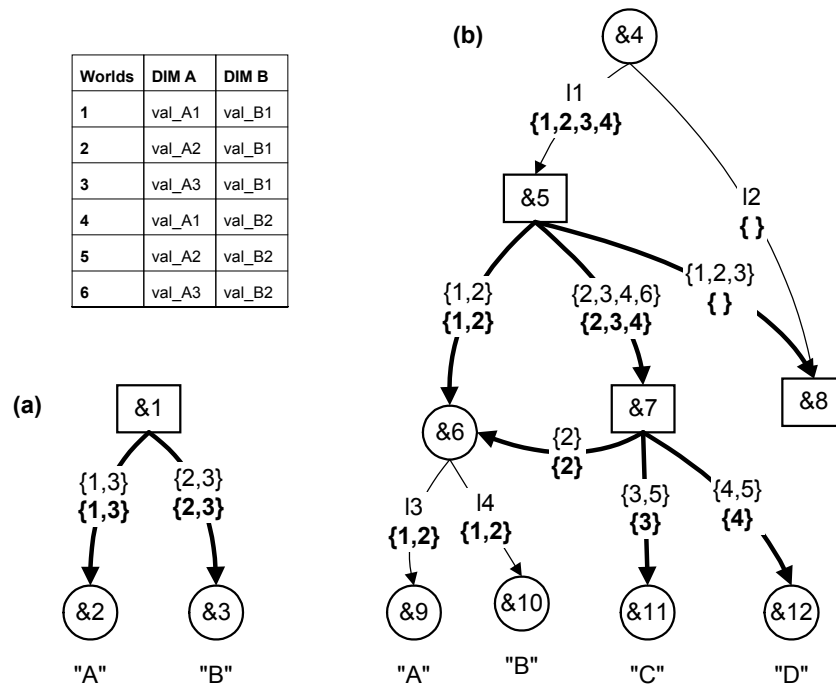
##### Ορισμός 4.12

*Ας είναι το  $G = (V_{mld}, V_{cxb}, E_{cxb}, E_{etb}, r, v)$  ένας Πολυδιάστατος Γράφος Δεδομένων. Λέμε ότι το  $G$  είναι **αιτιοκρατικό ως προς το περιβάλλον** εάν και μόνον για κάθε κόσμο  $w$  και πολυδιάστατο κόμβο  $q$ , ο οποίος είναι είτε η ρίζα είτε δείχνεται από κάποια ακμή οντότητας, ισχύει το εξής: αν  $p_1, p_2, \dots, p_n$  είναι όλα τα μονοπάτια που ξεκινούν από το  $q$ , καταλήγουν σε κόμβους περιβάλλοντος, και αποτελούνται μόνον από ακμές περιβάλλοντος η κληρονομούμενη κάλυψη των οποίων περιέχει το  $w$ , τότε όλα αυτά τα μονοπάτια  $p_1, p_2, \dots, p_n$  καταλήγουν στον ίδιο κόμβο περιβάλλοντος.*

Προφανώς, αν ένα περιβάλλον  $c$  αναπαριστά δυο ή περισσότερους κόσμους και τα  $p_k, p_j$  υφίστανται κάτω από το  $c$  και οδηγούν σε διαφορετικούς κόμβους περιβάλλοντος, τότε ο γράφος δεν είναι αιτιοκρατικός ως προς το περιβάλλον. Έτσι, σε έναν αιτιοκρατικό ως προς το περιβάλλον γράφο, δεδομένου ενός περιβάλλοντος  $c$  που αναπαριστά τουλάχιστον έναν κόσμο, αν αρχίσουμε από έναν πολυδιάστατο κόμβο που δείχνεται από μια ακμή οντότητας,

μπορούμε να φθάσουμε το πολύ σε έναν κόμβο περιβάλλοντος μέσω συνεχόμενων ακμών περιβάλλοντος που υφίστανται κάτω από το  $c$ . Με άλλα λόγια, σε έναν αιτιοκρατικό ως προς το περιβάλλον γράφο κάθε ακμή οντότητας που δείχνει σε έναν πολυδιάστατο κόμβο μπορεί να συσχετιστεί με έναν το πολύ κόμβο περιβάλλοντος κάτω από οποιοδήποτε μη κενό περιβάλλον  $c$ .

Εικόνα 4.9: Αιτιοκρατικοί και μη αιτιοκρατικοί ως προς το περιβάλλον γράφοι.



Θεωρήστε το παράδειγμα στην Εικόνα 4.9. Προκειμένου να απλοποιήσουμε τους γράφους και να τους κάνουμε πιο εύληπτους, οι προσδιοριστές περιβάλλοντος στην Εικόνα 4.9 εμφανίζονται σαν τα σύνολα των κόσμων που αναπαριστούν. Η αντιστοιχία δίνεται στον πίνακα στην πάνω αριστερή γωνία, όπου θεωρούμε δυο διαστάσεις A και B με πεδία ορισμού  $\{val\_A1, val\_A2, val\_A3\}$  και  $\{val\_B1, val\_B2\}$  αντίστοιχα. Για κάθε ακμή η ετικέτα της ή το ρητό περιβάλλον της εμφανίζεται στην πρώτη γραμμή, ενώ η δεύτερη γραμμή δείχνει την κληρονομούμενη κάλυψη με έντονους χαρακτήρες. Ο Πολυδιάστατος Γράφος Δεδομένων στην Εικόνα 4.9 (a) είναι μη αιτιοκρατικός ως προς το περιβάλλον, επειδή ο πολυδιάστατος κόμβος &1 είναι η ρίζα και συνδέεται με περισσότερους από έναν κόμβους περιβάλλοντος μέσω ακμών περιβάλλοντος που υφίστανται κάτω από τον ίδιο κόσμο, συγκεκριμένα τον κόσμο 3. Από την άλλη, ο γράφος στην Εικόνα 4.9 (b) είναι αιτιοκρατικός ως προς το περιβάλλον. Και να γιατί. Υπάρχουν δυο πολυδιάστατοι κόμβοι ενδιαφέροντος, ο κόμβος &5 και ο &8. Ο κόμβος &7 δεν είναι σχετικός, επειδή ούτε είναι η ρίζα, ούτε δείχνεται από κάποια ακμή οντότητας. Ο κόμβος &8 είναι φύλλο, και έτσι δεν είναι συνδεδεμένος με κανένα κόμβο περιβάλλοντος. Ο κόμβος &5 συνδέεται με τους κόμβους περιβάλλοντος &6, &11, και &12 μέσω μονοπατιών που αποτελούνται αποκλειστικά από ακμές περιβάλλοντος. Τα μονοπάτια  $\langle (&5, \{1,2\}, &6) \rangle$ ,  $\langle (&5, \{2,3,4,6\}, &7) \rangle$ ,  $\langle (&7, \{3,5\}, &11) \rangle$ ,  $\langle (&5, \{2,3,4,6\}, &7) \rangle$ ,  $\langle (&7, \{4,5\}, &12) \rangle$  υφίστανται κάτω από ξένα σύνολα κόσμων, ή ισοδύναμα, οι κληρονομούμενες καλύψεις μονοπατιού τους ( $\{1,2\}$ ,  $\{3\}$ , και  $\{4\}$  αντίστοιχα) είναι αμοιβαία αποκλειόμενες. Έτσι,

δεδομένου ενός κόσμου  $w$ , το πολύ ένα από τα μονοπάτια αυτά μπορεί να υφίσταται κάτω από το  $w$ . Το μόνο άλλο μονοπάτι  $\langle (\&5, \{2, 3, 4, 6\}, \&7) \cdot (\&7, \{2\}, \&6) \rangle$  υφίσταται κάτω από το  $\{2\}$  μαζί με το μονοπάτι  $\langle (\&5, \{1, 2\}, \&6) \rangle$  που υφίσταται κάτω από τα  $\{1, 2\}$ . Όμως και τα δυο μονοπάτια οδηγούν στον ίδιο κόμβο περιβάλλοντος (oid  $\&6$ ), και έτσι ο γράφος είναι αιτιοκρατικός ως προς το περιβάλλον.

Εναλλακτικά ο Ορισμός 4.12 μπορεί να διατυπωθεί ως εξής: ένας γράφος είναι αιτιοκρατικός ως προς το περιβάλλον εάν και μόνον για κάθε πολυδιάστατο κόμβο  $q$  που είναι είτε η ρίζα είτε δείχνεται από μια ακμή οντότητας, ισχύει το ακόλουθο: ας είναι  $icv_p, icv_u, \dots, icv_n$  οι αντίστοιχες ενώσεις περιβάλλοντος των κληρονομούμενων καλύψεων μονοπατιού όλων των μονοπατιών συνεχόμενων ακμών περιβάλλοντος που οδηγούν από το  $q$  στους κόμβους περιβάλλοντος  $p, u, \dots, n$  τότε τα  $icv_p, icv_u, \dots, icv_n$  είναι αμοιβαία αποκλειόμενα μεταξύ τους.

Στην Ενότητα 4.5 θα δούμε ότι η **κανονική μορφή** ενός Πολυδιάστατου Γράφου Δεδομένων επιτρέπει να ελέγξουμε εάν ένας γράφος είναι αιτιοκρατικός ως προς το περιβάλλον ή όχι με έναν απλούστερο και πιο διαισθητικό τρόπο.

#### 4.4.1.2 Αναγωγή σε OEM

Δεδομένου ενός αιτιοκρατικού ως προς το περιβάλλον Πολυδιάστατου Γράφου Δεδομένων  $G = (V_{mld}, V_{ext}, E_{ext}, E_{ett}, r, v)$  και ενός προσδιοριστή περιβάλλοντος  $w$  που αναπαριστά έναν μόνο κόσμο<sup>44</sup>, η ακόλουθη διεργασία εξάγει από το  $G$  ένα συμβατικό OEM  $O_w$  που υφίσταται κάτω από το  $w$ .

**$O_w \leftarrow \text{reduceToOEM}(G, w)$  is:**

1. Απομακρύνουμε κάθε κόμβο από το  $V$  και κάθε ακμή από το  $E$  των οποίων η κληρονομούμενη κάλυψη δεν είναι υπερσύνολο περιβάλλοντος του  $w$ .
2. Για κάθε  $(q, l, p) \in E_{ett}$  τέτοιο ώστε  $p \in V_{mld}$ , απομακρύνουμε το  $(q, l, p)$  από το  $E_{ett}$ , και προσθέτουμε το  $(q, l, u)$  στο  $E_{ett}$  εάν δεν υπάρχει ήδη, όπου  $u$  είναι ένας κόμβος περιβάλλοντος τέτοιος ώστε ένα μονοπάτι από ακμές περιβάλλοντος οδηγεί από το  $p$  στο  $u$ . Επιπλέον, αν η ρίζα  $r$  είναι ένας κόμβος  $p \in V_{mld}$ , τότε θέτουμε το  $r$  να είναι ο κόμβος  $u$ , όπου  $u$  είναι ένας κόμβος περιβάλλοντος τέτοιος ώστε ένα μονοπάτι από ακμές περιβάλλοντος οδηγεί από το  $p$  στο  $u$ .
3. Επιστρέφουμε το  $O_w = (V_{ext}, E_{ett}, r, v)$ .

Το OEM  $O_w$  είναι τότε η έκφανση του Πολυδιάστατου Γράφου Δεδομένων  $G$  κάτω από το  $w$ .

Ας συζητήσουμε την διεργασία **reduceToOEM** πιο λεπτομερώς. Όπως έχουμε δει στην Ενότητα 4.3.3.2, μετά το Βήμα 1 της διεργασίας οι εναπομείναντες κόμβοι και ακμές του  $G$  έχουν τις ακόλουθες ιδιότητες: (a) υφίστανται όλοι κάτω από το  $w$ , (b) σχηματίζουν έναν υπογράφο που είναι και αυτός Πολυδιάστατος Γράφος Δεδομένων και που έχει την ίδια ρίζα με το  $G$ , και (c) τα φύλλα είναι αποκλειστικά ατομικοί κόμβοι. Στην Ενότητα 4.3.3.2

<sup>44</sup> Σε σχέση με κάποιο σύνολο διαστάσεων  $D$ , όπου  $D \supseteq D_{min}$ , και το  $D_{min}$  περιέχει ακριβώς τις διαστάσεις εκείνες που εμφανίζονται στο  $G$ , και καμία άλλη. Στην οριακή περίπτωση που καμία διάσταση δεν εμφανίζεται στο  $G$  (οι προσδιοριστές περιβάλλοντος είναι αποκλειστικά της μορφής  $[\ ]$  και  $[-]$ ), για την αναγωγή σε OEM μπορεί να θεωρηθεί ένας κόσμος σε σχέση με κάποιο τυχαίο  $D$ : το αποτέλεσμα θα είναι πανομοιότυπο κάτω από οποιονδήποτε πιθανό κόσμο και σύνολο διαστάσεων.

ισχυρισθήκαμε ότι είναι πάντοτε δυνατόν να εξάγουμε ένα OEM από κάποιον τέτοιο υπογράφο. Η κατασκευή του OEM στην ουσία εκτελείται στο Βήμα 2 της διεργασίας. Το Βήμα 2 παίρνει κάθε ακμή οντότητας που δείχνει σε έναν πολυδιάστατο κόμβο  $p$  και την κάνει να δείχνει σε έναν κόμβο περιβάλλοντος  $u$ . Όπως έχουμε δει στην Ενότητα 4.4.1.1, ο κόμβος  $u$  είναι η μόνη έκφανση του  $p$  που είναι προσβάσιμη από το  $p$  κάτω από το  $w$ , επειδή ο  $G$  είναι αιτιοκρατικός ως προς το περιβάλλον. Επιπλέον, ο κόμβος  $u$  πρέπει να υπαρχει διαφορετικά το  $p$  δεν θα είχε πρόσβαση σε κανέναν ατομικό κόμβο κάτω από το  $w$ , και θα είχε απομακρυνθεί στο Βήμα 1. Συνεπώς, κάθε ακμή οντότητας που δείχνει σε κάποιον πολυδιάστατο κόμβο μπορεί να κατευθυνθεί εκ νέου ώστε να δείχνει στον σωστό κόμβο περιβάλλοντος, αντικαθιστώντας ένα μονοπάτι ακμών περιβάλλοντος. Σημειώστε ότι αν υπάρχει ήδη κάποια ακμή ταυτόσημη με την επανακατευθυνθείσα, μια από τις δυο ταυτόσημες ακμές πρέπει να απομακρυνθεί. Στην συνέχεια μια παρόμοια διεργασία εφαρμόζεται στην ρίζα, ώστε να διασφαλιστεί ότι η ρίζα είναι κόμβος περιβάλλοντος. Στο τέλος του Βήματος 2 κάθε ακμή οντότητας δείχνει σε κάποιον κόμβο περιβάλλοντος, και κάθε κόμβος περιβάλλοντος (εκτός ίσως της ρίζας) δείχνεται από κάποια ακμή οντότητας. Τέλος, το Βήμα 3 απομακρύνει όλους τους εναπομείναντες πολυδιάστατους κόμβους και ακμές περιβάλλοντος. Η συνάρτηση  $v$ , που αναθέτει τιμές σε κόμβους του αρχικού γράφου  $G$ , μπορεί να χρησιμοποιηθεί για να αναθέσει τιμές και στους κόμβους του  $O_w$ .

Όπως εξηγήθηκε στην Ενότητα 4.3.2.3, αν ένας κόσμος  $w$  δεν καλύπτεται από την κληρονομούμενη κάλυψη της ρίζας  $r$  του  $G$ , δεν είναι δυνατόν να εξάγουμε OEM από το  $G$  που να υφίσταται κάτω από το  $w$ . Στην περίπτωση αυτή, μετά το Βήμα 1 της διεργασίας δεν θα υπάρχει κανένας κόμβος ή ακμή για να σχηματιστεί κάποιος υπογράφος του  $G$ , και θα επιστραφεί ένας άδειος γράφος.

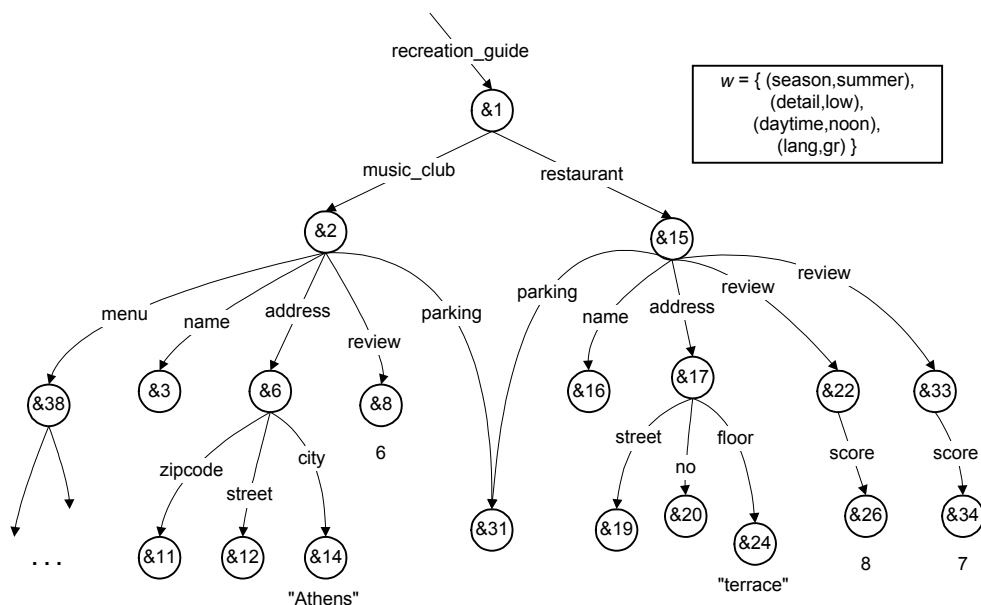
Στην διεργασία **reduceToOEM** υποθέσαμε έναν γράφο αιτιοκρατικό ως προς το περιβάλλον. Για κάποιον γράφο  $G$  που είναι μη αιτιοκρατικός ως προς το περιβάλλον, είναι δυνατόν κάτω από έναν κόσμο  $w$  να αντιστοιχούν περισσότεροι από έναν κόμβοι περιβάλλοντος σε μια και μόνη ακμή οντότητας κατά το Βήμα 2. Επιπλέον, μπορεί να ανακαλύψουμε ότι υπάρχουν πάνω από μια υποψήφια ρίζες. Σε τέτοιες περιπτώσεις, είναι δυνατόν να κατασκευάσουμε περισσότερα του ενός OEM που να υφίστανται κάτω από τον ίδιο κόσμο, παίρνοντας όλους τους πιθανούς συνδυασμούς των υποψήφιων κόμβων και δημιουργώντας ένα OEM για κάθε τέτοιο συνδυασμό. Μια άλλη δυνατότητα είναι να κατασκευάσουμε ένα και μόνο OEM που να περιέχει όλες τις υφισταμένες εκφάνσεις μιας οντότητας σαν διαφορετικά αντικείμενα. *Οι αιτιοκρατικοί ως προς το περιβάλλον γράφοι εξασφαλίζουν ότι για κάθε κόσμο υπάρχει το πολύ ένα OEM που υφίσταται κάτω από τον κόσμο αυτό.* Στα πλαίσια αυτής της εργασίας ενδιαφερόμαστε κυρίως για αιτιοκρατικούς ως προς το περιβάλλον Πολυδιάστατους Γράφους Δεδομένων.

Σαν παράδειγμα αναγωγής σε OEM, θεωρήστε τον Πολυδιάστατο Γράφο Δεδομένων στην Εικόνα 4.3. Εφαρμόζοντας την διεργασία **reduceToOEM** στον γράφο αυτό για τον κόσμο<sup>45</sup> [season=summer, detail=low, daytime=noon, lang=gr], παίρνουμε τον OEM γράφο που απεικονίζεται στην Εικόνα 4.10. Όλα τα φύλλα είναι ατομικοί κόμβοι, πάντως για απλότητα έχουμε παραλείψει κάποιες από τις τιμές τους. Παρατηρήστε πως οι δυο πολυδιάστατες οντότητες parking με oids &28 και &29 αναπαρίστανται από το αντικείμενο με oid &31. Επίσης, παρατηρήστε πως το αντικείμενο comment με oid &35 δεν έχει συμπεριληφθεί στο τελικό OEM.

<sup>45</sup> Για απλότητα, χρησιμοποιούμε τον όρο **κόσμος** για να αναφερθούμε σε έναν προσδιοριστή περιβάλλοντος που αναπαριστά ακριβώς έναν κόσμο.



Εικόνα 4.10: Το OEM που παράγεται από την αναγωγή του Πολυδιάστατου Γράφου Δεδομένων της Εικόνα 4.3 κάτω από τον κόσμο  $w$ .



Η χρονική πολυπλοκότητα της διεργασίας **reduceToOEM** εκτιμάται ως ακολούθως. Το Βήμα 1 περιέχει μια πράξη υποσυνόλου περιβάλλοντος για κάθε κόμβο και κάθε ακμή στον γράφο, συνεπώς υποθέτοντας ότι υπάρχουν  $n$  κόμβοι και  $e$  ακμές, το κόστος του Βήματος 1 είναι  $(n + e) * \langle \text{κόστος υποσυνόλου περιβάλλοντος} \rangle$ . Το Βήμα 2 ελέγχει όλες τις ακμές οντότητας για να βρει αυτές που δείχνουν σε πολυδιάστατους κόμβους: έτσι αν ο αριθμός των ακμών οντότητας είναι  $ett$ , έχουμε  $ett$  συγκρίσεις. Αν ο μέσος αριθμός των συνεχόμενων ακμών περιβάλλοντος που οδηγούν σε κάποιον κόμβο περιβάλλοντος είναι  $m$ , το κόστος δεν μπορεί να υπερβεί τις  $ett + ett * m$  συγκρίσεις. Η ίδια διεργασία εφαρμόζεται και στην ρίζα, έτσι το συνολικό κόστος του Βήματος 2 είναι το πολύ  $(ett + 1) + (ett + 1) * m$  συγκρίσεις. Το συνολικό κόστος της διεργασίας δεν θα υπερβαίνει τις  $(n + e) * \langle \text{κόστος υποσυνόλου περιβάλλοντος} \rangle + (ett + 1) + (ett + 1) * m$  συγκρίσεις.

Θεωρούμε τον αριθμό  $e$  των ακμών σαν αντιπροσωπευτικό του μεγέθους του προβλήματος, και υποθέτουμε ότι το κόστος του υποσυνόλου περιβάλλοντος είναι άνω φραγμένο. Οι κόμβοι δεν μπορούν να είναι περισσότεροι από  $e + 1$  (κάθε κόμβος εκτός από την ρίζα έχει τουλάχιστον μια εισερχόμενη ακμή), και τα  $ett$ ,  $m$  δεν μπορούν να είναι μεγαλύτερα από το  $e$ . Συνεπώς, η τάξη μεγέθους της χρονικής πολυπλοκότητας είναι  $O(e^2)$ .

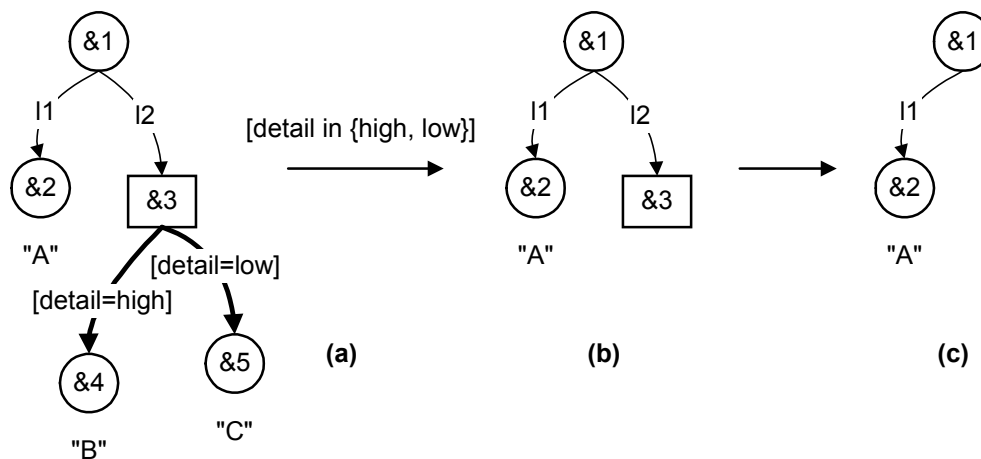
#### 4.4.1.3 OEM γράφοι που υφίστανται κάτω από δύο ή περισσότερους κόσμους

Στην προηγούμενη ενότητα ορίσαμε την διεργασία **reduceToOEM** που εξάγει από έναν Πολυδιάστατο Γράφο Δεδομένων ένα OEM που υφίσταται κάτω από έναν κόσμο  $w$ . Η ερώτηση που ανακύπτει είναι εάν η διεργασία **reduceToOEM** μπορεί να γενικευθεί για περιβάλλοντα που αναπαριστούν περισσότερους του ενός κόσμων<sup>46</sup>. Η γενικευμένη

<sup>46</sup> Οι κόσμοι στην ενότητα αυτή θεωρούνται σε σχέση με το  $\mathbf{D}_{\min}$ , που περιέχει ακριβώς τις διαστάσεις εκείνες οι οποίες εμφανίζονται στους προσδιοριστές περιβάλλοντος του Πολυδιάστατου Γράφου Δεδομένων, και καμία άλλη. Η προσθήκη επιπλέον διαστάσεων δεν επηρεάζει το πρόβλημα που συζητούμε στην ενότητα αυτή.

διεργασία θα μπορούσε να παίρνει σαν είσοδο έναν Πολυδιάστατο Γράφο Δεδομένων  $G$  και οποιοδήποτε μη κενό<sup>47</sup> περιβάλλον  $c$ , και να παράγει ένα OEM  $O_c$  που να υφίσταται κάτω από το  $c$ .

**Εικόνα 4.11: Προβλήματα στην εξαγωγή OEM γράφου που υφίσταται κάτω από δύο κόσμους.**



Η Εικόνα 4.11 δείχνει μέσω ενός παραδείγματος πως θα δούλευε μια τέτοια διεργασία. Ο γράφος  $G$  στην Εικόνα 4.11 (a) πρόκειται να αναχθεί σε ένα OEM που υφίσταται κάτω από το περιβάλλον  $c = [\text{detail in } \{\text{high}, \text{low}\}]$ , το οποίο αναπαριστά δυο κόσμους. Οι κληρονομούμενες καλύψεις των ακμών στο  $G$  έχουν ως ακολούθως: η  $I1$  έχει κληρονομούμενη κάλυψη  $[\ ]$ , η  $I2$  έχει κληρονομούμενη κάλυψη  $[\text{detail in } \{\text{high}, \text{low}\}]$ , και οι κληρονομούμενες καλύψεις των δυο ακμών περιβάλλοντος είναι οι ίδιες όπως τα αντίστοιχα ρητά τους περιβάλλοντα. Ο υπογράφος του  $G$  που υφίσταται κάτω από το  $c$  φαίνεται στην Εικόνα 4.11 (b). Παρατηρήστε ότι ο κόμβος &3 στον υπογράφο αυτόν δεν είναι ατομικός κόμβος, παρόλο που είναι φύλλο. Μια τέτοια περίπτωση δεν θα μπορούσε να παρουσιαστεί αν το  $c$  αναπαριστούσε έναν και μόνο κόσμο, λόγω της ιδιότητας (c) στην Ενότητα 4.3.3.2, που εγγυάται ότι τα φύλλα στον υπογράφο είναι αποκλειστικά ατομικοί κόμβοι του αρχικού γράφου. Η ιδιότητα αυτή, όμως, δεν ισχύει στην εξαγωγή υπογράφων κάτω από περισσότερους του ενός κόσμων.

Η απομάκρυνση τέτοιων παράνομων φύλλων θα μπορούσε να λύσει το πρόβλημα, όπως φαίνεται στον OEM γράφο στην Εικόνα 4.11 (c). Σημειώστε ότι η απομάκρυνση αυτή είναι δυνατόν να μεταδοθεί προς τα πάνω: αν υποθέσουμε ότι η ακμή  $I1$  και ο κόμβος &2 δεν υπήρχαν· τότε η απομάκρυνση του  $I2$  και του &3 θα άφηνε το σύνθετο αντικείμενο &1 χωρίς καθόλου παιδιά, και η αναγωγή σε OEM θα έπρεπε να επιστρέψει έναν κενό γράφο.

Όπως η ιδιότητα (c) στην Ενότητα 4.3.3.2, έτσι και η ιδιότητα (a) της ίδιας ενότητας δεν ισχύει για την εξαγωγή υπογράφων κάτω από περισσότερους του ενός κόσμων. Η συνέπεια αυτού είναι ότι τα στοιχεία του γράφου που υφίστανται κάτω από κάποιο περιβάλλον είναι δυνατόν να μην σχηματίζουν έναν Πολυδιάστατο (υπο-) Γράφο Δεδομένων, αν το περιβάλλον αναπαριστά περισσότερους από έναν κόσμο. Για παράδειγμα, υποθέστε ότι οι κόμβοι &4 και &5 στην Εικόνα 4.11 (a) έδειχναν σε έναν νέο κόμβο &6. Τότε, ο κόμβος &6

<sup>47</sup> Δεν έχει κανένα νόημα να εξάγουμε ένα OEM που δεν υφίσταται κάτω από κανέναν κόσμο.

θα υφίστατο κάτω από το περιβάλλον  $[detail\ in\ \{high, low\}]$ , αλλά κανένα μονοπάτι από την ρίζα στον κόμβο  $\&6$  δεν θα υφίστατο κάτω από το περιβάλλον αυτό.

Τα παραπάνω παραδείγματα δείχνουν ότι η διεργασία **reduceToOEM** στην τρέχουσα μορφή της δεν δουλεύει για περιβάλλοντα που αναπαριστούν περισσότερους του ενός κόσμους. Μια γενικευμένη διεργασία θα έπρεπε να πάρει υπόψη την πιθανότητα να υπάρχουν παράνομα φύλλα και κόμβοι που δεν είναι προσβάσιμοι από την ρίζα. Ένα OEM  $O_c$  που υφίσταται κάτω από το  $c$  θα αποτελείται από στοιχεία γράφου που είναι κοινά σε όλα τα OEMs  $O_w$ , για κάθε κόσμο  $w$  που καλύπτει το  $c$ . Πάντως, το  $O_c$  ίσως να μην εμπεριέχει κάθε τέτοιο κοινό στοιχείο γράφου, μια και κάποια πιθανόν να εισάγουν παράνομα φύλλα, ενώ αλλά πιθανόν να μην είναι προσβάσιμα από την ρίζα μέσω ενός μονοπατιού που να υφίσταται κάτω από το  $c$ .

Το συμπέρασμα είναι ότι η αρχική μας προσέγγιση να εξάγουμε το  $O_c$  με παρόμοιο τρόπο όπως εξάγουμε το  $O_w$  μας οδήγησε σε ένα OEM που δεν έχει ενδιαφέρουσες ιδιότητες ούτε κάποια ιδιαίτερη σημασία. Το πρόβλημα προκύπτει από την παραδοχή μας ότι το  $O_c$  σχηματίζεται χρησιμοποιώντας εκείνα τα στοιχεία γράφου που υφίστανται κάτω από το  $c$ , όπως συμβαίνει με τα περιβάλλοντα που αναπαριστούν έναν μόνο κόσμο. Κάνοντας ένα βήμα προς τα πίσω, παρουσιάζουμε στην συνέχεια μια πιο ουσιαστική προσέγγιση για το πότε ένα OEM υφίσταται κάτω από ένα μη κενό περιβάλλον:

*Δεδομένων ενός Πολυδιάστατου Γράφου Δεδομένων  $G$  και ενός περιβάλλοντος  $c$ , με  $c \neq \mathcal{E}$ , λέμε ότι το  $G$  μπορεί να αναχθεί σε ένα OEM  $O_c$  που υφίσταται κάτω από το  $c$  εάν και μόνον το  $G$  ανάγεται στο OEM  $O_c$  κάτω από κάθε κόσμο που καλύπτεται από το  $c$ .*

Με βάση τα παραπάνω, η διεργασία της αναγωγής του  $G$  σε ένα OEM που να υφίσταται κάτω από κάποιο μη κενό περιβάλλον  $c$  περιλαμβάνει τα ακόλουθα βήματα: (a) εφαρμόζουμε την διεργασία **reduceToOEM** στο  $G$ , για να πάρουμε το OEM  $O_w$  που υφίσταται κάτω από κάποιον κόσμο  $w$  ο οποίος καλύπτεται από το  $c$ , και (b) ελέγχουμε εάν η αναγωγή κάτω από όλους τους άλλους κόσμους στο  $c$  θα δώσει το ίδιο OEM  $O_w$ . Η αναγωγή σε OEM κάτω από ένα περιβάλλον είναι προφανώς πιο περιοριστική από την αναγωγή σε OEM κάτω από έναν και μόνο κόσμο. Σύμφωνα με αυτόν τον τελευταίο ορισμό, η αναγωγή σε OEM κάτω από το περιβάλλον  $[detail\ in\ \{high, low\}]$  του γράφου στην Εικόνα 4.11 (a) δεν είναι δυνατή<sup>48</sup>.

Ένα είδος αναγωγής που χρησιμοποιεί στην είσοδο περιβάλλον αντί για έναν και μόνο κόσμο είναι η **μερική αναγωγή**, η οποία είναι το θέμα της επόμενης ενότητας.

## 4.4.2 Μερική αναγωγή

Δεδομένων ενός Πολυδιάστατου Γράφου Δεδομένων  $G = (V_{mld}, V_{ext}, E_{ext}, E_{eff}, r, v)$  και ενός περιβάλλοντος  $c$ , η **μερική αναγωγή** εξάγει έναν υπογράφο του  $G$ , που αποτελείται από κόμβους και ακμές που υφίστανται κάτω από έναν τουλάχιστον από τους κόσμους που αναπαριστά το  $c$ . Έτσι, σε αντίθεση με την αναγωγή σε OEM, το αποτέλεσμα της μερικής αναγωγής είναι και πάλι ένας Πολυδιάστατος Γράφος Δεδομένων. Η ακόλουθη διεργασία εκτελεί μερική αναγωγή.

<sup>48</sup> Στην περίπτωση αυτή, το αποτέλεσμα της αναγωγής δεν είναι ο κενός γράφος επειδή αυτό θα σήμαινε ότι η αναγωγή σε OEM του αρχικού γράφου κάτω από κάθε έναν από τους δυο κόσμους στο  $[detail\ in\ \{high, low\}]$  δίνει σαν αποτέλεσμα τον κενό γράφο, πράγμα που δεν ισχύει.

**$G_c \leftarrow \text{reducePartially}(G, c)$  is:**

1. Απομακρύνουμε κάθε κόμβο από το  $V$  και κάθε ακμή από το  $E$  με κληρονομούμενη κάλυψη  $icv$ , τέτοια ώστε  $icv \cap c = \mathcal{E}$ .
2. Επιστρέφουμε  $G_c = (V_{mld}, V_{ext}, E_{ext}, E_{ett}, r, v)$ .

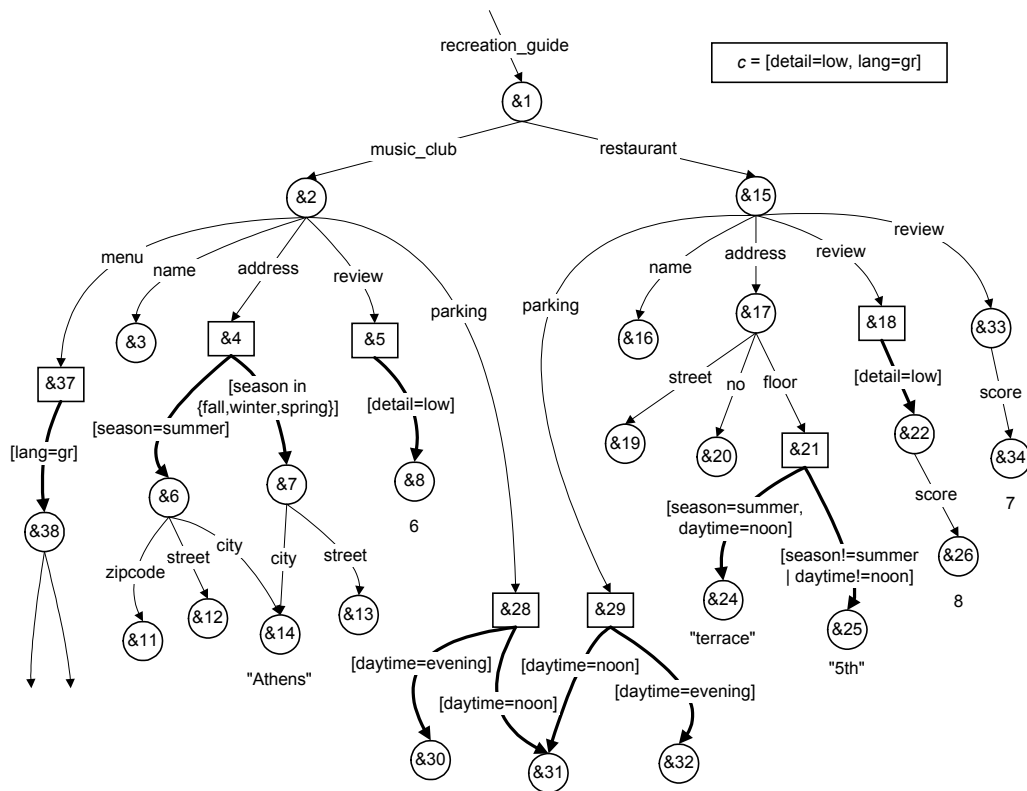
Στην πράξη, το Βήμα 1 της διεργασίας **reducePartially** απομακρύνει κόμβους και ακμές του  $G$  που δεν υφίστανται κάτω από κανέναν από τους κόσμους που καλύπτει το  $c$ . Οι εναπομείναντες κόμβοι και ακμές υφίστανται κάτω από έναν τουλάχιστον από τους κόσμους στο  $c$ . Ο γράφος  $G_c$  που επιστρέφεται στο Βήμα 2 είναι ένας υπογράφος του  $G$ , που είναι επίσης Πολυδιάστατος Γράφος Δεδομένων και έχει την ίδια ρίζα με το  $G$ . Επιπλέον, τα φύλλα στο  $G_c$  είναι αποκλειστικά ατομικοί κόμβοι. Για να γίνει κατανοητό το γιατί, θεωρήστε το ακόλουθο: κάθε κόμβος ή ακμή στο  $G_c$  υφίσταται κάτω από κάποιον τουλάχιστο κόσμο  $w$ , συνεπώς είναι μέρος ενός υπογράφου  $G_w$  του  $G$  που υφίσταται κάτω από το  $w$ . Όπως δείχνουμε στην Ενότητα 4.3.3.2, το  $G_w$  είναι και το ίδιο ένας Πολυδιάστατος Γράφος Δεδομένων με την ίδια ρίζα όπως το  $G$  και με ατομικούς κόμβους σαν φύλλα. Προφανώς, όλοι οι κόμβοι και οι ακμές του  $G_w$  ανήκουν και στο  $G_c$ . Συνεπώς, το  $G_c$  αποτελεί στην πραγματικότητα μια συνάθροιση υπογράφων όπως ο  $G_w$  για κάθε  $w$  που αναπαριστά το  $c$ . Εξυπακούεται ότι το  $G_c$  είναι ένας Πολυδιάστατος Γράφος Δεδομένων με την ίδια ρίζα όπως το  $G$ , του οποίου τα φύλλα είναι ατομικοί κόμβοι.

Είναι προφανές ότι αν ένα OEM που υφίσταται κάτω από το  $w$  μπορεί να εξαχθεί από το  $G$ , μπορεί επίσης να εξαχθεί και από το  $G_c$  με την προϋπόθεση ότι  $w \subseteq c$ . Συνεπώς, το  $G_c$  «περιέχει» τα OEMs του  $G$  που υφίστανται κάτω από κάθε έναν κόσμο που περιέχεται στο  $c$ .

Σημειώστε ότι αν το  $c$  είναι ένα κενό περιβάλλον, τότε η διεργασία **reducePartially** θα επιστρέψει έναν κενό γράφο. Αν το  $c$  είναι ένα καθολικό περιβάλλον, τότε τα τμήματα του  $G$  που θα αποκοπούν θα είναι εκείνα που δεν υφίστανται κάτω από κανέναν κόσμο.

Ο γράφος στην Εικόνα 4.12 είναι το αποτέλεσμα της εφαρμογής της διεργασίας **reducePartially** στον Πολυδιάστατο Γράφο Δεδομένων που φαίνεται στην Εικόνα 4.3, για το περιβάλλον `[detail=low, lang=gr]`. Αναφέρουμε και πάλι ότι, όλα τα φύλλα του γράφου στην Εικόνα 4.12 είναι ατομικοί κόμβοι, αλλά για απλότητα έχουμε παραλείψει μερικές τιμές τους. Παρατηρήστε ότι ο γράφος OEM που απεικονίζεται στην Εικόνα 4.10, ο οποίος είναι το αποτέλεσμα της αναγωγής σε OEM του γράφου στην Εικόνα 4.3, μπορεί επίσης να ανακτηθεί ανάγοντας σε OEM τον γράφο στην Εικόνα 4.12. Αυτό είναι έτσι επειδή `[season=summer, detail=low, daytime=noon, lang=gr] <= [detail=low, lang=gr]`.

Εικόνα 4.12: Ο υπογράφος που παράγεται από μερική αναγωγή του Πολυδιάστατου Γράφου Δεδομένων της Εικόνα 4.3 για το ερμηνευτικό περιβάλλον  $c$ .



Η χρονική πολυπλοκότητα της διεργασίας **reducePartially** εκτιμάται ως ακολούθως. Το Βήμα 1 περιέχει μια πράξη τομής περιβάλλοντος και έναν έλεγχο για ισότητα περιβάλλοντος για κάθε κόμβο και κάθε ακμή στον γράφο. Υποθέτοντας ότι υπάρχουν  $n$  κόμβοι και  $e$  ακμές, το συνολικό κόστος της διεργασίας είναι  $(n + e) * (<\text{κόστος τομής περιβάλλοντος}> + <\text{κόστος ισότητας περιβάλλοντος}>)$ .

Θεωρούμε τον αριθμό  $e$  των ακμών σαν αντιπροσωπευτικό του μεγέθους του προβλήματος, και δεχόμαστε ότι τα κόστη της τομής περιβάλλοντος και της ισότητας περιβάλλοντος είναι άνω φραγμένα. Οι κόμβοι δεν μπορούν να είναι περισσότεροι από  $e + 1$  (κάθε κόμβος εκτός από την ρίζα έχει τουλάχιστον μια εισερχόμενη ακμή). Συνεπώς, η τάξη μεγέθους της χρονικής πολυπλοκότητας είναι  $O(e)$ .

#### 4.4.3 Στοιχειώδεις πράξεις της αναγωγής

Μια ενδιαφέρουσα παρατήρηση είναι ότι εάν το περιβάλλον  $c$  αναπαριστά έναν μόνο κόσμο<sup>49</sup>, το Βήμα 1 της διεργασίας **reducePartially** στην Ενότητα 4.4.2 δίνει το ίδιο αποτέλεσμα με το Βήμα 1 της διεργασίας **reduceToOEM** στην Ενότητα 4.4.1.2.

Αυτό μας οδηγεί στο να εισάγουμε έναν αριθμό στοιχειωδών πράξεων και για τις δυο διεργασίες αναγωγής, που αναγράφονται παρακάτω:

<sup>49</sup> Θεωρούμενος σε σχέση με το  $\mathbf{D}_{\min}$ . Επιπλέον διαστάσεις δεν παίζουν σημαντικό ρόλο στην περίπτωση αυτή.

- **Περιβαλλοντική προβολή γράφου** που συζητήθηκε λεπτομερώς στην Ενότητα 4.3.3.3.
- **Περιβαλλοντική περικοπή γράφου** που απομακρύνει κάθε κόμβο και ακμή με κληρονομούμενη κάλυψη  $icv = \mathcal{E}$ . Μπορεί να επιστρέφει σαν αποτέλεσμα έναν κενό Πολυδιάστατο Γράφο Δεδομένων.
- **Απο-περιβαλλοντοποίηση γράφου** που εφαρμόζεται σε έναν Πολυδιάστατο Γράφο Δεδομένων  $G$ , και επιστρέφει έναν γράφο απαλλαγμένο από πολυδιάστατους κόμβους και ακμές περιβάλλοντος. Κάθε ακμή οντότητας στο  $G$  επανακατευθύνεται ώστε να δείχνει σε έναν κόμβο περιβάλλοντος, αν δεν δείχνει ήδη σε κάποιον, όπως περιγράφηκε στο Βήμα 2 της διεργασίας **reduceToOEM** στην Ενότητα 4.4.1.2: όταν όλες οι ακμές οντότητας δείχνουν σε κόμβους περιβάλλοντος, οι πολυδιάστατοι κόμβοι και οι ακμές περιβάλλοντος απομακρύνονται από τον γράφο.

Είναι προφανές ότι η διεργασία **reducePartially** που εισήχθη στην Ενότητα 4.4.2 είναι ισοδύναμη με την περιβαλλοντική προβολή γράφου, και την περιβαλλοντική περικοπή γράφου με την σειρά αυτή. Η μόνη διαφορά είναι ότι η περιβαλλοντική προβολή γράφου εισάγει έναν επιπλέον πολυδιάστατο κόμβο σαν την νέα ρίζα, και μια αντιστοιχη ακμή περιβάλλοντος που έχει σαν ετικέτα το περιβάλλον της αναγωγής.

Η διεργασία **reduceToOEM** που περιγράφηκε στην Ενότητα 4.4.1.2 είναι ισοδύναμη με την περιβαλλοντική προβολή γράφου σε έναν μόνο κόσμο, την περιβαλλοντική περικοπή γράφου, και την απο-περιβαλλοντοποίηση γράφου<sup>50</sup> με την σειρά αυτή. Συνεπώς, η αναγωγή σε OEM είναι η ίδια σαν να απο-περιβαλλοντοποιούμε το αποτέλεσμα της μερικής αναγωγής για έναν μόνο κόσμο.

## 4.5 ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ

Στην Ενότητα 4.2.2 είδαμε ότι οι πολυδιάστατες οντότητες μπορούν να αναπαρασταθούν με έναν αριθμό ισοδύναμων τρόπων, και δώσαμε κάποια παραδείγματα στην Εικόνα 4.2. Εξυπακούεται ότι αν δυο Πολυδιάστατοι Γράφοι Δεδομένων περιέχουν διαφορετικές αλλά ισοδύναμες αναπαραστάσεις πολυδιάστατων οντοτήτων, τότε θα είναι μεν διαφορετικοί αλλά ουσιαστικά θα εκφράζουν την ίδια πληροφορία. Η παρατήρηση αυτή οδηγεί στον ορισμό της **κανονικής μορφής (CF για συντομία)** ενός Πολυδιάστατου Γράφου Δεδομένων.

### Ορισμός 4.13

*Ένας Πολυδιάστατος Γράφος Δεδομένων βρίσκεται σε **κανονική μορφή** εάν και μόνον η ρίζα είναι πολυδιάστατος κόμβος, κάθε ακμή οντότητας δείχνει σε πολυδιάστατο κόμβο, και κάθε ακμή περιβάλλοντος δείχνει σε κόμβο περιβάλλοντος.*

<sup>50</sup> Για να δουλέψει σωστά η απο-περιβαλλοντοποίηση κάθε ακμή οντότητας πρέπει να αντιστοιχεί ακριβώς σε έναν κόμβο περιβάλλοντος. Όπως έχουμε δείξει, την αντιστοιχία αυτή την εγγυάται (a) ένας αρχικός γράφος που να είναι ατιοκρατικός ως προς το περιβάλλον, μαζί με την (b) περιβαλλοντική προβολή γράφου σε έναν μόνο κόσμο και την περιβαλλοντική περικοπή γράφου. Το αποτέλεσμα της απο-περιβαλλοντοποίησης είναι πιθανόν να μην είναι ένας γράφος OEM: αν ο γράφος εισόδου έχει φύλλα που δεν είναι ατομικοί κόμβοι, ο απο-περιβαλλοντοποιημένος γράφος θα διατηρήσει τα φύλλα αυτά. Η περιβαλλοντική προβολή γράφου και η περιβαλλοντική περικοπή γράφου διασφαλίζουν ότι δεν υπάρχουν τέτοια φύλλα στον γράφο εισόδου.

Αν ένας γράφος είναι σε κανονική μορφή, κάθε κόμβος περιβάλλοντος είναι η έκφραση κάποιου πολυδιάστατου κόμβου, και οι εκφάνσεις των πολυδιάστατων κόμβων είναι αποκλειστικά κόμβοι περιβάλλοντος (συμβατικές πληροφοριακές οντότητες).

Δεδομένου ενός Πολυδιάστατου Γράφου Δεδομένων  $G = (V_{mld}, V_{ext}, E_{ext}, E_{ett}, r, v)$  μπορούμε να μετασχηματίσουμε το  $G$  σε κανονική μορφή  $G_{CF}$  με την ακόλουθη διεργασία.

#### $G_{CF} \leftarrow CF(G)$ is:

1. Αν η ρίζα είναι ένας κόμβος περιβάλλοντος  $q$ , προσθέτουμε έναν νέο πολυδιάστατο κόμβο  $p$  και μια ακμή περιβάλλοντος  $(p, [ ], q)$ , και κάνουμε το  $p$  την νέα ρίζα.
2. Για κάθε κόμβο περιβάλλοντος  $q$  που δείχνεται από κάποια ακμή (ή ακμές) οντότητας της μορφής  $(u, l, q)$ , προσθέτουμε έναν νέο πολυδιάστατο κόμβο  $p$  και μια ακμή περιβάλλοντος  $(p, [ ], q)$ , και αντικαθιστούμε κάθε εισερχόμενη ακμή οντότητας  $(u, l, q)$  με μια ακμή οντότητας  $(u, l, p)$ .
3. Για κάθε πολυδιάστατο κόμβο που είτε είναι η ρίζα είτε δείχνεται από κάποια ακμή οντότητας, κάνουμε τα παρακάτω: για κάθε κόμβο περιβάλλοντος  $p$  προσβάσιμο από το  $q$  μέσω ακυκλικών μονοπατιών  $h_1, h_2, \dots, h_n$  που απαρτίζονται από συνεχόμενες ακμές περιβάλλοντος, απομακρύνουμε όποιες ακμές περιβάλλοντος  $(q, c, p)$  εάν υπάρχουν, και προσθέτουμε μια νέα ακμή περιβάλλοντος  $(q, c', p)$  όπου  $c'$  είναι η ένωση περιβάλλοντος των ρητών περιβαλλόντων μονοπατιού των  $h_1, h_2, \dots, h_n$ .
4. Απομακρύνουμε όλες τις ακμές περιβάλλοντος που δείχνουν σε πολυδιάστατους κόμβους. Στην συνέχεια, απομακρύνουμε όλους τους πολυδιάστατους κόμβους που δεν είναι η ρίζα και που δεν δείχνονται από κάποια ακμή οντότητας, και όλες τις ακμές περιβάλλοντος που ξεκινούν από τους κόμβους αυτούς.
5. Επιστρέφουμε τον νέο γράφο  $G_{CF} = (V_{mld}, V_{ext}, E_{ext}, E_{ett}, r, v)$ .

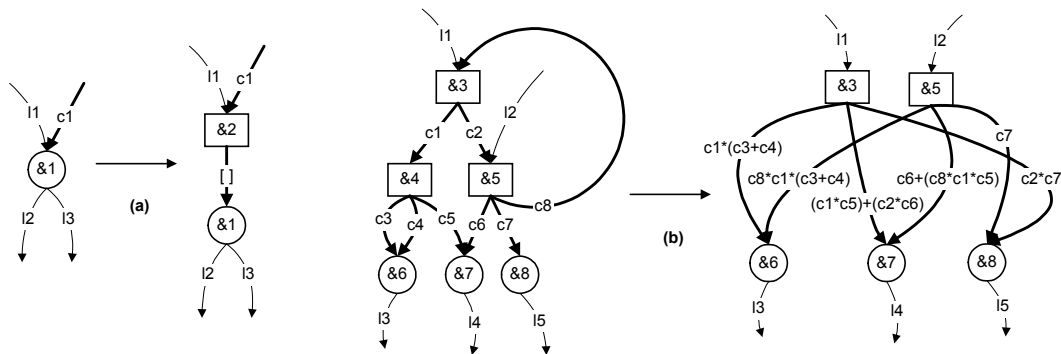
Διασθητικά το Βήμα 1 και το Βήμα 2 της διεργασίας  $CF$  δίνονται στην Εικόνα 4.13 (a), ενώ το Βήμα 3 και το Βήμα 4 στην Εικόνα 4.13 (b). Η διεργασία δεν προσθέτει ούτε αφαιρεί κόμβους περιβάλλοντος και ακμές οντότητας, παρόλο που κάποιες ακμές οντότητας μπορεί να επανακατευθυνθούν αλλάζοντας προορισμό όπως απεικονίζεται στην Εικόνα 4.13 (a) για την ακμή  $l1$ . Πολυδιαστατοι κόμβοι και ακμές περιβάλλοντος μπορούν να προστεθούν ή να αφαιρεθούν, αλλά με τρόπο που δεν αλλοιώνει τις συνθήκες οι οποίες αποφασίζουν πότε υφίσταται ένας κόμβος περιβάλλοντος.

Για παράδειγμα, θεωρήστε την ακμή οντότητας  $l1$  και τον κόμβο  $\&3$  στην Εικόνα 4.13 (b), ο οποίος μπορεί να προσπελάσει τον κόμβο περιβάλλοντος  $\&8$  μέσω των ακόλουθων μονοπατιών από συνεχόμενες ακμές περιβάλλοντος:  $\langle (\&3, c2, \&5) . (\&5, c7, \&8) \rangle$  και  $\langle (\&3, c2, \&5) . (\&5, c8, \&3) . (\&3, c2, \&5) . (\&5, c7, \&8) \rangle$ . Στην αντίστοιχη κανονική μορφή, η ακμή οντότητας  $l1$  και ο κόμβος  $\&3$  προσπελαίνουν τον κόμβο περιβάλλοντος  $\&8$  μέσω μιας και μόνο ακμής περιβάλλοντος, της οποίας το ρητό περιβάλλον συνδυάζει τους περιορισμούς που εκφράζουν τα δυο μονοπάτια:  $(c2 * c7) + (c2 * c8 * c2 * c7)$ . Όμως, σημειώστε ότι μια και το δεύτερο μονοπάτι περιέχει έναν κύκλο και λόγω των επιπλέον περιορισμών που εκφράζει το  $c8$ , το δεύτερο μονοπάτι δεν μπορεί παρά να αναπαριστά ένα υποσύνολο των κόσμων του πρώτου μονοπατιού. Συνεπώς, το Βήμα 3 εξαιρεί τα κυκλικά μονοπάτια, και το ρητό περιβάλλον της αντίστοιχης ακμής στην κανονική μορφή γίνεται  $c2 * c7$ .

Ένα άλλο σημείο που αξίζει να προσέξει κανείς είναι ότι ο κόμβος  $\&4$ , που δεν δείχνεται από καμία ακμή οντότητας, δεν υπάρχει στην κανονική μορφή, και οι συνθήκες που

εκφράζονται από τα αντίστοιχα ρητά του περιβάλλοντα έχουν ενσωματωθεί στα ρητά περιβάλλοντα των κόμβων &3 και &5.

**Εικόνα 4.13: Μετατροπή Πολυδιάστατου Γράφου Δεδομένων σε κανονική μορφή.**



Από την Εικόνα 4.13 (a) βλέπουμε ότι το Βήμα 1 και το Βήμα 2 του μετασχηματισμού δεν επηρεάζουν το κληρονομούμενο περιβάλλον και την κάλυψη περιβάλλοντος των ακμών οντότητας  $l_1$ ,  $l_2$ ,  $l_3$  και του κόμβου &1. Το ίδιο ισχύει για το Βήμα 3 και το Βήμα 4, όπως μπορούμε να δούμε στην Εικόνα 4.13 (b) για τις ακμές οντότητας  $l_1$ ,  $l_2$ ,  $l_3$ ,  $l_4$ ,  $l_5$  και για τους κόμβους &6, &7, και &8. Θυμηθείτε ότι η Πρόταση 4.3 εκφράζει το κληρονομούμενο περιβάλλον σαν την ένωση περιβάλλοντος των ρητών περιβαλλόντων μονοπατιού, και έτσι η συνεισφορά (σε κόσμους) της ακμής  $l_1$  στο κληρονομούμενο περιβάλλον του κόμβου &7, για παράδειγμα, παραμένει η ίδια και μετά τον μετασχηματισμό σε κανονική μορφή. Παρόμοια είναι η περίπτωση με την Πρόταση 4.7, σχετικά με την συνεισφορά της ακμής  $l_4$ , για παράδειγμα, στην κάλυψη περιβάλλοντος του κόμβου &3. Συμπερασματικά, η κανονική μορφή διατηρεί το κληρονομούμενο περιβάλλον και την κάλυψη περιβάλλοντος των κόμβων περιβάλλοντος και των ακμών οντότητας του αρχικού γράφου.

Είναι εύκολο να δει κανείς στην Εικόνα 4.13 ότι αν ένας πολυδιάστατος κόμβος στον αρχικό γράφο έχει πρόσβαση κάτω από έναν κόσμο σε περισσότερους από έναν κόμβους περιβάλλοντος μέσω συνεχόμενων ακμών περιβάλλοντος (Ορισμός 4.12), το ίδιο εξακολουθεί να συμβαίνει και στην κανονική μορφή αυτού του γράφου, και αντίστροφα. Έτσι, η κανονική μορφή διατηρεί την αιτιοκρατία ως προς το περιβάλλον:

#### Πρόταση 4.12

*Αν ένας γράφος είναι αιτιοκρατικός ως προς το περιβάλλον τότε έτσι είναι και η κανονική μορφή του, ενώ αν ο γράφος είναι μη αιτιοκρατικός ως προς το περιβάλλον τότε η κανονική μορφή του είναι και αυτή μη αιτιοκρατική ως προς το περιβάλλον.*

Στην πραγματικότητα, όταν ένας γράφος είναι σε κανονική μορφή είναι απλό να διαπιστώσουμε εάν ο γράφος είναι αιτιοκρατικός ως προς το περιβάλλον:

#### Πρόταση 4.13

*Ένας γράφος σε κανονική μορφή είναι αιτιοκρατικός ως προς το περιβάλλον εάν και μόνον για κάθε πολυδιάστατο κόμβο  $q$  οι κληρονομούμενες καλύψεις των ακμών που ξεκινούν από το  $q$  είναι αμοιβαία αποκλειόμενες.*



Προφανώς, αν τα ρητά περιβάλλοντα των ακμών αυτών είναι αμοιβαία αποκλειόμενα, ο γράφος είναι αιτιοκρατικός ως προς το περιβάλλον, μια και η κληρονομούμενη κάλυψη μιας ακμής είναι υποσύνολο περιβάλλοντος του ρητού της περιβάλλοντος. Το αντίστροφο δεν ισχύει: τα ρητά περιβάλλοντα μπορούν να μην είναι αμοιβαία αποκλειόμενα, αλλά ο γράφος να είναι αιτιοκρατικός ως προς το περιβάλλον. Παρατηρείστε ότι είναι επιτρεπτό για έναν πολυδιάστατο κόμβο ενός αιτιοκρατικού ως προς το περιβάλλον γράφου να έχει δυο ή περισσότερες ακμές περιβάλλοντος με κληρονομούμενες καλύψεις  $\mathcal{E}$ , μια και τα κενά περιβάλλοντα είναι αμοιβαία αποκλειόμενα με οποιοδήποτε άλλο περιβάλλον, ακόμη και με άλλα κενά περιβάλλοντα.

Συμπερασματικά, ο γράφος  $G_{CF}$  που επιστρέφεται από την διεργασία **CF** είναι ένας Πολυδιάστατος Γράφος Δεδομένων με τους ίδιους κόμβους περιβάλλοντος όπως ο αρχικός γράφος  $G$ , τον ίδιο αριθμό ακμών οντότητας οι οποίες συνδέουν αντίστοιχες (αν και όχι ταυτόσημες μορφολογικά) οντότητες, και ισοδύναμους περιορισμούς που εκφράζουν τις συνθήκες κάτω από τις οποίες υφίστανται οι εκφάνσεις. Συνεπώς, σε κάθε Πολυδιάστατο Γράφο Δεδομένων  $G$  αντιστοιχεί ένας Πολυδιάστατος Γράφος Δεδομένων  $G_{CF}$  που βρίσκεται σε κανονική μορφή, και που στην ουσία περιέχει την ίδια πληροφορία με τον  $G$ .

Μια σημαντική συνέπεια που προκύπτει από την παραπάνω συζήτηση διατυπώνεται στην ακόλουθη πρόταση:

#### Πρόταση 4.14

*Ας είναι το  $G$  ένας Πολυδιάστατος Γράφος Δεδομένων, και το  $G_{CF}$  η κανονική μορφή του. Τότε, για κάθε κόσμο  $w$ , αν  $O_w \leftarrow \text{reduceToOEM}(G, w)$  και  $O_w' \leftarrow \text{reduceToOEM}(G_{CF}, w)$ , τότε οι  $O_w$  και  $O_w'$  είναι ταυτόσημοι γράφοι OEM.*

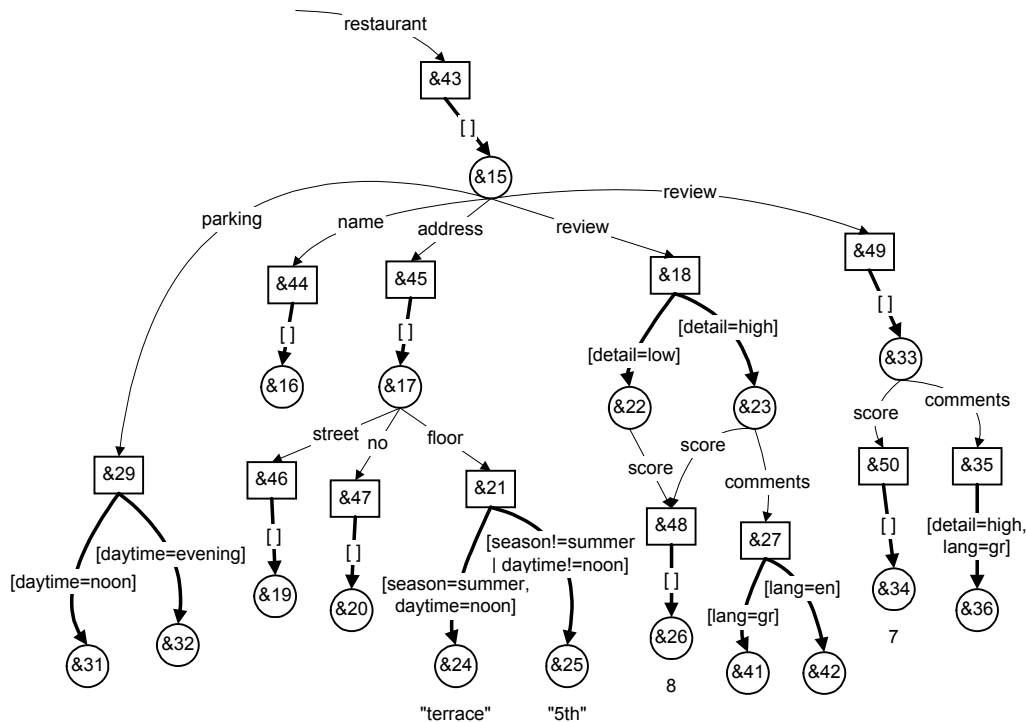
Η κανονική μορφή  $G_{CF}$  ενός Πολυδιάστατου Γράφου Δεδομένων δεν περιέχει περιττές εξαρτήσεις μεταξύ πολυδιάστατων οντοτήτων, και έτσι αποφεύγει μια σειρά *ανωμαλιών* που μπορεί να προκύψουν στο  $G$ . Στα ακόλουθα ο γράφος στο αριστερό μέρος στην Εικόνα 4.13 (b) αναφέρεται σαν  $G$ :

- (a) Θεωρήστε ότι το ρητό περιβάλλον της ακμής περιβάλλοντος (&5, c7, &8) του  $G$  αλλάζει μέσω κάποιας ενημέρωσης σε c7'. Αυτό δεν θα επηρεάσει μόνον την πολυδιάστατη οντότητα που αναπαριστά ο κόμβος &5, αλλά και την οντότητα που αναπαριστά ο &3. Η παρενέργεια αυτή είναι μια *ανωμαλία ενημέρωσης*.
- (b) Αν μια νέα έκφραση προστεθεί στην πολυδιάστατη οντότητα που αναπαριστά ο κόμβος &3 του  $G$ , εξαιτίας της ακμής (&5, c8, &3) η έκφραση αυτή εμμέσως θα προστεθεί και στην πολυδιάστατη οντότητα που αναπαριστά ο κόμβος &5. Η παρενέργεια αυτή είναι μια *ανωμαλία εισαγωγής*.
- (c) Αν η ακμή (&3, c1, &4) απομακρυνθεί από το  $G$ , τότε η οντότητα που αναπαριστά ο κόμβος &4 δεν θα είναι πια έκφραση του &3. Αυτό όμως θα κάνει τον κόμβο &5 να χάσει την πρόσβασή του στον κόμβο &6. Η παρενέργεια αυτή είναι μια *ανωμαλία διαγραφής*.

Οι ανωμαλίες αυτές αποφεύγονται χρησιμοποιώντας την κανονική μορφή  $G_{CF}$  του  $G$ , που για τα παραδείγματά μας είναι ο γράφος στο δεξί μέρος στην Εικόνα 4.13 (b).

Μια άλλη ιδιότητα της κανονικής μορφής είναι ότι *κάθε δυνατό μονοπάτι σχηματίζεται από μια επαναλαμβανόμενη αλληλοδιάδοχη μιας ακμής περιβάλλοντος και μιας ακμής οντότητας*. Δεν μπορούν να υπάρξουν περισσότερες από μια συνεχόμενες ακμές περιβάλλοντος ή συνεχόμενες ακμές οντότητας σε έναν γράφο που βρίσκεται σε κανονική μορφή. Η ιδιότητα αυτή χρησιμοποιείται για την διαμόρφωση και την αποτίμηση **εκφράσεων μονοπατιού περιβάλλοντος** σε επερωτήσεις, που είναι το αντικείμενο του επόμενου κεφαλαίου.

Εικόνα 4.14: Τμήμα του γράφου της Εικόνα 4.3 σε κανονική μορφή.



Σαν παράδειγμα θεωρήστε την Εικόνα 4.14, που δείχνει την κανονική μορφή του αντικειμένου restaurant από τον γράφο στην Εικόνα 4.3.

## 4.6 ΠΟΛΥΔΙΑΣΤΑΤΟ ΟΕΜ

Έχοντας διερευνήσει τις ιδιότητες των Πολυδιάστατων Γράφων Δεδομένων, μπορούμε πλέον να ορίσουμε μια ειδική κατηγορία Πολυδιάστατου Γράφου Δεδομένων που καλούμε **Πολυδιάστατο Μοντέλο Ανταλλαγής Αντικειμένων** (Multidimensional Object Exchange Model), ή **Πολυδιάστατο ΟΕΜ**, ή **ΜΟΕΜ** για συντομία.

### Ορισμός 4.14

*Το Πολυδιάστατο Μοντέλο Ανταλλαγής Αντικειμένων (ΜΟΕΜ) είναι ένας αιτιοκρατικός ως προς το περιβάλλον Πολυδιάστατος Γράφος Δεδομένων του οποίου κάθε κόμβος και ακμή έχει κληρονομούμενη κάλυψη  $icv \neq \mathcal{E}$ .*

Σαν παράδειγμα ΜΟΕΜ θεωρήστε τον Πολυδιάστατο Γράφο Δεδομένων που απεικονίζεται στην Εικόνα 4.3. Αντίθετα, ο γράφος στην Εικόνα 4.5 (a) δεν είναι ΜΟΕΜ, μια και το κληρονομούμενο περιβάλλον και συνεπώς η κληρονομούμενη κάλυψη της ακμής (&5, [scale=high], &11) και του κόμβου &11 είναι το κενό περιβάλλον [-].

Κάθε κόμβος και ακμή ενός ΜΟΕΜ υφίσταται κάτω από έναν τουλάχιστον κόσμο. Αυτό ισχύει και για την ρίζα, συνεπώς η κληρονομούμενη κάλυψη της ρίζας περιέχει τουλάχιστον έναν κόσμο: έτσι ένα ΜΟΕΜ εγκλείει τουλάχιστον ένα συμβατικό ΟΕΜ που υφίσταται κάτω

από κάποιον κόσμο. Επιπρόσθετα, τα φύλλα ενός MOEM πρέπει να υφίστανται κάτω από κάποιον κόσμο, συνεπώς τα φύλλα είναι αποκλειστικά ατομικοί κόμβοι. Κάθε κόμβος περιβάλλοντος και ακμή οντότητας του MOEM συμμετέχει σε τουλάχιστον ένα συμβατικό OEM στο οποίο είναι δυνατόν να αναχθεί το MOEM. Προφανώς, το OEM είναι ειδική περίπτωση του Πολυδιάστατου OEM, όπου δεν υπάρχουν πολυδιάστατοι κόμβοι και ακμές περιβάλλοντος.

Ο έλεγχος για το εάν ένας Πολυδιάστατος Γράφος Δεδομένων G είναι MOEM αποτελείται από τα ακόλουθα βήματα: (a) υπολογίζουμε τις κληρονομούμενες καλύψεις των κόμβων και των ακμών στο G, (b) ελέγχουμε ότι καμία από αυτές δεν είναι κενό περιβάλλον, και (c) εξετάζουμε το G για αιτιοκρατία ως προς το περιβάλλον.

Εύκολα φαίνεται ότι η κανονική μορφή ενός MOEM είναι και πάλι MOEM<sup>51</sup>. Όπως ήδη έχουμε αναφέρει, ο μετασχηματισμός σε κανονική μορφή διατηρεί την αιτιοκρατία ως προς το περιβάλλον, όπως και τις κληρονομούμενες καλύψεις των κόμβων περιβάλλοντος και των ακμών οντότητας. Σχετικά με τις κληρονομούμενες καλύψεις των πολυδιάστατων κόμβων και των ακμών περιβάλλοντος, δεν είναι δυνατόν να γίνουν το κενό περιβάλλον σαν αποτέλεσμα του μετασχηματισμού. Επίσης, όπως έχει λεχθεί, η κανονική μορφή  $M_{CF}$  ενός MOEM M περιέχει στην ουσία την ίδια πληροφορία όπως το M, και ανάγεται στα ίδια ακριβώς OEMs όπως το M.

Δεδομένου ενός αιτιοκρατικού ως προς το περιβάλλον Πολυδιάστατου Γράφου Δεδομένων G, μπορούμε να πάρουμε τον μέγιστο υπογράφο του G που είναι MOEM εφαρμόζοντας μερική αναγωγή στο G για το καθολικό περιβάλλον []. Η μερική αναγωγή θα απομακρύνει τους κόμβους και τις ακμές που δεν υφίστανται κάτω από κανέναν κόσμο, αφήνοντας αυτούς με κληρονομούμενη κάλυψη  $icv \neq [-]$ , εφόσον υπάρχουν τέτοιοι κόμβοι και ακμές στο G. Όπως είδαμε στην Ενότητα 4.3.3.2, όλοι αυτοί οι κόμβοι και οι ακμές που επιβιώνουν συνδέονται με την ρίζα μέσω μονοπατιών που επίσης επιβιώνουν.

## 4.7 ΕΚΦΡΑΣΕΙΣ MSSD-EXPRESSIONS

Όπως αναφέραμε στο Κεφάλαιο 2, η συντακτική βάση για την διατύπωση των ημιδομημένων δεδομένων είναι οι εκφράσεις *ssd-expression*. Ορίζουμε την έκφραση **mssd-expression** επεκτείνοντας την *ssd-expression* ώστε να ενσωματώσει προσδιοριστές περιβάλλοντος. Την γραμματική της *mssd-expression* την δίνει ο Πίνακας 4.1 σε Extended Backus-Naur Form [EBNF], ή EBNF για συντομία. Τα σύμβολα που μπορούν να οριστούν από μια ομαλή έκφραση (regular expression) αρχίζουν με κεφαλαίο γράμμα (παράδειγμα: `AtomicValue`), ενώ τα σύμβολα που ορίζονται σε EBNF αρχίζουν με μικρό γράμμα (παράδειγμα: `complexValue`).

<sup>51</sup> Το αντίθετο δεν ισχύει αναγκαστικά: ένας Πολυδιάστατος Γράφος Δεδομένων που δεν είναι MOEM, μπορεί να έχει κανονική μορφή που να είναι MOEM. Για παράδειγμα, θεωρήστε έναν Πολυδιάστατο Γράφο Δεδομένων G ο οποίος περιέχει έναν πολυδιάστατο κόμβο p που είναι φύλλο και που δεν δείχνεται από καμία ακμή οντότητας. Τότε το G δεν είναι MOEM, επειδή ο p έχει κενή κληρονομούμενη κάλυψη. Όμως, η κανονική μορφή  $G_{CF}$  του G δεν περιέχει το p, το οποίο απομακρύνθηκε από την διεργασία μετασχηματισμού, και το  $G_{CF}$  είναι δυνατόν να είναι MOEM.

**Πίνακας 4.1: Σύνταξη εκφράσεων mssd-expression.**

```

mssd-expr ::= value | Oid value | Oid
value ::= AtomicValue | "{" complexValue "}"
           | "<" multidimValue ">"
complexValue ::= LabelName ":" mssd-expr ("," complexValue)?
multidimValue ::= cxtSpec ":" mssd-expr ("," multidimValue)?

```

Είναι προφανές ότι το `multidimValue` αντιστοιχεί στους πολυδιάστατους κόμβους, ενώ τα `AtomicValue` και `complexValue` αντιστοιχούν στους κόμβους περιβάλλοντος. Οι συνθήκες [ABS00] που αφορούν στην σύνταξη των ετικετών, και στα αναγνωριστικά αντικειμένων στις `ssd-expressions`, ισχύουν επίσης και στις `mssd-expressions`. Η γραμματική για τους προσδιοριστές περιβάλλοντος, ή `cxtSpec`, έχει δοθεί στο προηγούμενο κεφάλαιο.

Η `mssd-expression` στο Παράδειγμα 4.1 περιγράφει το `music_club` με `oid &2` στην Εικόνα 4.3.

**Παράδειγμα 4.1**

```

&2 {menu: &37 <[lang=gr]: &38 {...},
      [lang=en]: &39 {...},
      [lang=fr]: &40 {...}>,
    name: &3,
    address: &4 <[season=summer]: &6 {zipcode: &11, street: &12,
      city: &14 "Athens"},
      [season in {fall,winter,spring}]:
        &7 {city: &14, street: &13}>,
    review: &5 <[detail=low]: &8 6,
      [detail=high]: &9 {score: &8, comments: &10}>,
    parking: &28 <[daytime=evening]: &30,
      [daytime=noon]: &31>
  }

```

◆

Για να εκφράσει σωστά ένα MOEM, η αντίστοιχη `mssd-expression` θα πρέπει να είναι **συνεπής**. Η συνέπεια των `mssd-expressions` είναι παρόμοια με την συνέπεια [ABS00] των `ssd-expressions`. Για να είναι συνεπής μια `mssd-expression s`, θα πρέπει να ισχύουν τα ακόλουθα:

- (a) Κάθε αναγνωριστικό αντικειμένου  $o$  ορίζεται το πολύ μια φορά στο  $s$ .
- (b) Αν ένα αναγνωριστικό αντικειμένου  $o$  χρησιμοποιείται στο  $s$ , θα πρέπει να έχει οριστεί στο  $s$ .

Ένα αναγνωριστικό αντικειμένου θεωρείται *ορισμένο* αν του έχει ανατεθεί μια τιμή. Στην πράξη, η συνέπεια μιας `mssd-expression` απαιτεί ότι σε κάθε φύλλο έχει ανατεθεί ακριβώς μια τιμή.

Ενώ το MOEM αντιστοιχεί πάντοτε σε μια συνεπή `mssd-expression`, δεν ισχύει το ίδιο για τον Πολυδιάστατο Γράφο Δεδομένων. Στην γενική περίπτωση, ένας Πολυδιάστατος Γράφος Δεδομένων μπορεί να έχει σύνθετα ή πολυδιάστατα αντικείμενα σαν φύλλα, τα οποία δεν έχουν ατομικές τιμές. Η χρήση μιας μη συνεπούς `mssd-expression` για την αναπαράσταση ενός τέτοιου γράφου είναι δυνατόν να οδηγήσει σε αμφισημία: ένα αντικείμενο σε μια `mssd-expression` που απαρτίζεται μόνον από ένα `oid` χωρίς εξερχόμενες ακμές και χωρίς ατομική τιμή μπορεί να αντιστοιχεί σε ένα φύλλο του Πολυδιάστατου Γράφου Δεδομένων που είναι είτε κόμβος περιβάλλοντος είτε πολυδιάστατος κόμβος! Η αμφισημία αίρεται εάν γνωρίζουμε ότι η αμφίσημη `mssd-expression` αναπαριστά την κανονική μορφή του Πολυδιάστατου Γράφου Δεδομένων, επειδή ο τύπος του κόμβου μπορεί να προσδιοριστεί

από τον τύπο των εισερχόμενων ακμών. Επιπλέον, ο τύπος ενός κόμβου μπορεί να κωδικοποιηθεί στο oid του κόμβου αυτού, κάνοντας το πρόβλημα πιο πολύ ένα ζήτημα υλοποίησης. Σε κάθε περίπτωση, ένας Πολυδιάστατος Γράφος Δεδομένων του οποίου τα φύλλα είναι ατομικά αντικείμενα μπορεί να αναπαρασταθεί από mssd-expressions χωρίς την ανάγκη μέτρων για την άρση αμφισημιών.

## 4.8 ΠΕΡΙΛΗΨΗ

Στο κεφάλαιο αυτό ενσωματώσαμε προσδιοριστές περιβάλλοντος σε ένα μοντέλο δεδομένων γράφου για ημιδομημένα δεδομένα, και ορίσαμε τον **Πολυδιάστατο Γράφο Δεδομένων**. Στον Πολυδιάστατο Γράφο Δεδομένων, οι συμβατικές ακμές με ετικέτα (**ακμές οντότητας**) ορίζουν συσχετίσεις αναμεταξύ πολυδιάστατων οντοτήτων, οι οποίες είναι πληροφοριακές οντότητες που δύνανται να εκδηλώνουν διαφορετικές εκφάνσεις κάτω από διαφορετικούς κόσμους. Οι εκφάνσεις αυτές συνοδεύονται από προσδιοριστές περιβάλλοντος, που ορίζουν τους κόσμους κάτω από τους οποίους υφίσταται κάθε έκφραση.

Τα ερμηνευτικά περιβάλλοντα που είναι προσαρτημένα στο μοντέλο γράφου αναφέρονται σε κάποια συγκεκριμένη πολυδιάστατη οντότητα, πράγμα το οποίο διευκολύνει την ανάπτυξη και συντήρηση του γράφου. Τα πραγματικά ερμηνευτικά περιβάλλοντα μπορούν να υπολογισθούν αυτόματα μέσω μιας διαδικασίας που διατρέχει τον γράφο και παίρνει υπόψη όλες τις πολυδιάστατες οντότητες. Διερευνώντας τις ιδιότητες του Πολυδιάστατου Γράφου Δεδομένων, συζητήσαμε το πώς τα ερμηνευτικά περιβάλλοντα διαδίδονται μέσα στον γράφο, και ορίσαμε τα εξής:

- **Ρητό περιβάλλον**, που είναι το ερμηνευτικό περιβάλλον που συνοδεύει εκφάνσεις. Το ρητό περιβάλλον εκφράζει τους κόσμους κάτω από τους οποίους μια έκφραση υφίσταται *δυναμικά*, και έχει νόημα μόνο μέσα στα όρια μιας και μόνης πολυδιάστατης οντότητας.
- **Κληρονομούμενο περιβάλλον**, που πηγάζει από την απαίτηση ότι ένα αντικείμενο μπορεί να υφίσταται μόνο κάτω από τους κόσμους που υφίσταται κάποιος «πατέρας» του. Το κληρονομούμενο περιβάλλον εκφράζει τους κόσμους κάτω από τους οποίους υφίστανται στοιχεία του γράφου, όπως αυτοί καθορίζονται από τα ρητά περιβάλλοντα που συσσωρεύονται από την ρίζα προς τα φύλλα.
- **Κάλυψη περιβάλλοντος**, που βασίζεται στην απαίτηση ότι ένα αντικείμενο μπορεί να υφίσταται μόνο κάτω από τους κόσμους στους οποίους έχει πρόσβαση σε κάποιον ατομικό κόμβο. Η κάλυψη περιβάλλοντος εκφράζει τους κόσμους κάτω από τους οποίους υφίστανται στοιχεία του γράφου, όπως αυτοί καθορίζονται από τα ρητά περιβάλλοντα που συσσωρεύουν περιορισμούς από τα φύλλα προς την ρίζα.
- **Κληρονομούμενη κάλυψη**, που για κάθε στοιχείο του γράφου αντιπροσωπεύει τους κόσμους που είναι κοινοί και στο κληρονομούμενο περιβάλλον και στην κάλυψη περιβάλλοντος του στοιχείου. Η κληρονομούμενη κάλυψη ενός στοιχείου του γράφου υποδηλώνει τους κόσμους κάτω από τους οποίους το στοιχείο πράγματι υφίσταται.

Βασισμένοι στην διάδοση του ερμηνευτικού περιβάλλοντος, δείξαμε ότι από έναν Πολυδιάστατο Γράφο Δεδομένων μπορούν να εξαχθούν συμβατικά OEM που υφίστανται κάτω από κάποιον κόσμο, και δώσαμε τις συνθήκες κάτω από τις οποίες μπορούν να λάβουν χώρα τέτοιου είδους εξαγωγές. Σαν κομμάτι αυτών των συνθηκών συζητήσαμε για τους **αιτιοκρατικούς ως προς το περιβάλλον** γράφους, και παρουσιάσαμε μια διαδικασία που **ανάγει σε OEM** έναν αιτιοκρατικό ως προς το περιβάλλον Πολυδιάστατο Γράφο Δεδομένων κάτω από έναν κόσμο. Επιπρόσθετα, περιγράψαμε την **μερική αναγωγή** ενός

Πολυδιάστατου Γράφου Δεδομένων για κάποιο ερμηνευτικό περιβάλλον. Οι αναγωγές είναι λειτουργίες που μπορεί να λάβουν χώρα οποιαδήποτε στιγμή, ενώ οι κληρονομούμενες καλύψεις πρέπει να υπολογίζονται μόνον όταν ο γράφος έχει υποστεί αλλαγές.

Περιττές αλληλεξαρτήσεις ανάμεσα από πολυδιάστατες οντότητες μπορεί να προκαλέσουν ανωμαλίες κατά την συντήρηση ενός Πολυδιάστατου Γράφου Δεδομένων. Η **κανονική μορφή** αποφεύγει τις ανωμαλίες αυτές μετατρέποντας τον αρχικό γράφο σε έναν καλύτερα δομημένο γράφο.

Τέλος, ορίσαμε το **Πολυδιάστατο OEM (MOEM)** σαν μια ειδική περίπτωση Πολυδιάστατου Γράφου Δεδομένων ο οποίος προέρχεται αποκλειστικά από την συνένωση συμβατικών OEM που υφίστανται κάτω από διάφορους κόσμους. Ορίσαμε επίσης τις εκφράσεις **mssd-expression**, που παρέχουν ένα συντακτικό για την έκφραση Πολυδιάστατων Γράφων Δεδομένων και MOEM σε μορφή κειμένου.

## 5 ΠΟΛΥΔΙΑΣΤΑΤΗ ΓΛΩΣΣΑ ΕΠΕΡΩΤΗΣΕΩΝ

Στο προηγούμενο κεφάλαιο προτείναμε το **Πολυδιάστατο OEM**, ή **MOEM** για συντομία, ένα μοντέλο δεδομένων για δεδομένα που εξαρτώνται από το ερμηνευτικό περιβάλλον, και παρουσιάζουν πολλαπλές εκφάνσεις. Το MOEM είναι ειδική περίπτωση ενός μοντέλου δεδομένων γράφου που καλούμε **Πολυδιάστατο Γράφο Δεδομένων**. Εξετάσαμε λεπτομερώς τις ιδιότητες του Πολυδιάστατου Γράφου Δεδομένων, και δείξαμε τον τρόπο που το MOEM σχετίζεται με το OEM, το μοντέλο δεδομένων γράφου για συμβατικά<sup>52</sup> ημιδομημένα δεδομένα.

Στο κεφάλαιο αυτό εισάγουμε την **Πολυδιάστατη Γλώσσα Επερωτήσεων**, ή **MQL** για συντομία, μια γλώσσα επερωτήσεων για πολυδιάστατα ημιδομημένα δεδομένα. Η MQL αντιμετωπίζει το ερμηνευτικό περιβάλλον σαν πολίτη πρώτης κατηγορίας, και είναι συνεπώς κατάλληλη για την επερώτηση γράφων MOEM. Παρόλο που η MQL μπορεί να χρησιμοποιηθεί για την επερώτηση Πολυδιάστατων Γράφων Δεδομένων στην γενική τους περίπτωση<sup>53</sup>, το ενδιαφέρον μας στρέφεται περισσότερο προς τα MOEMs<sup>54</sup>. Όμοια με την βάση δεδομένων OEM [AQM+97, Suc98], ορίζουμε μια **βάση δεδομένων MOEM** σαν μια βάση δεδομένων της οποίας το μοντέλο είναι ένας γράφος MOEM. Στα ακόλουθα θεωρούμε ότι οι επερωτήσεις MQL τίθενται σε βάσεις δεδομένων MOEM, εκτός αν δηλώνεται ρητά διαφορετικά.

Στα πλαίσια της παρούσας εργασίας δεν αντιμετωπίζουμε το ζήτημα των ενημερώσεων της βάσης δεδομένων, αλλά επικεντρώνουμε στην ανάκτηση των δεδομένων. Ο στόχος μας είναι να δείξουμε με ποιο τρόπο συμβατικές γλώσσες επερωτήσεων μπορούν να επεκταθούν ώστε να ενσωματώσουν και να χρησιμοποιήσουν το ερμηνευτικό περιβάλλον σε κεντρικό ρόλο. Οι ίδιες αρχές που χρησιμοποιούνται για την διαμόρφωση του τμήματος ανάκτησης δεδομένων της MQL μπορούν επίσης να χρησιμοποιηθούν και για τον σχεδιασμό ενός τμήματος ενημέρωσης δεδομένων.

Στα ακόλουθα, αρχίζουμε εξηγώντας τους λόγους για την προσέγγισή μας. Στην συνέχεια συζητούμε τις **εκφράσεις μονοπατιού περιβάλλοντος**, οι οποίες είναι ένα σημείο-κλειδί της MQL. Συνεχίζουμε με μια σύντομη εισαγωγή στην ίδια την MQL, που συμπληρώνουμε με μια συζήτηση πάνω σε πιο προχωρημένα θέματα. Τέλος κλείνουμε το κεφάλαιο περιγράφοντας την πρότυπη υλοποίησή μας της MQL. Ολόκληρο το συντακτικό των εκφράσεων μονοπατιού περιβάλλοντος και της MQL δίνεται στο Παράρτημα A.

---

<sup>52</sup> Με τον όρο «συμβατικά» εννοούμε μη εξαρτώμενα από το ερμηνευτικό περιβάλλον.

<sup>53</sup> Η MQL δεν απαιτεί από τον υφιστάμενο γράφο να είναι αιτιοκρατικός ως προς το περιβάλλον, ή να έχει αποκλειστικά μη κενές κληρονομούμενες καλύψεις. Παρατηρήστε ότι αν ο υφιστάμενος γράφος έχει σαν φύλλα πολυδιάστατους ή σύνθετους κόμβους (πράγμα που επιτρέπεται σε έναν Πολυδιάστατο Γράφο Δεδομένων), τότε το αποτέλεσμα μιας επερώτησης MQL μπορεί να είναι πάλι ένας Πολυδιάστατος (υπο-) Γράφος Δεδομένων με φύλλα πολυδιάστατους ή σύνθετους κόμβους.

<sup>54</sup> Για μια συζήτηση σχετικά με την ποιοτική διαφορά των επερωτήσεων σε MOEM και σε Πολυδιάστατο Γράφο Δεδομένων, δείτε την Ενότητα 5.2.4.

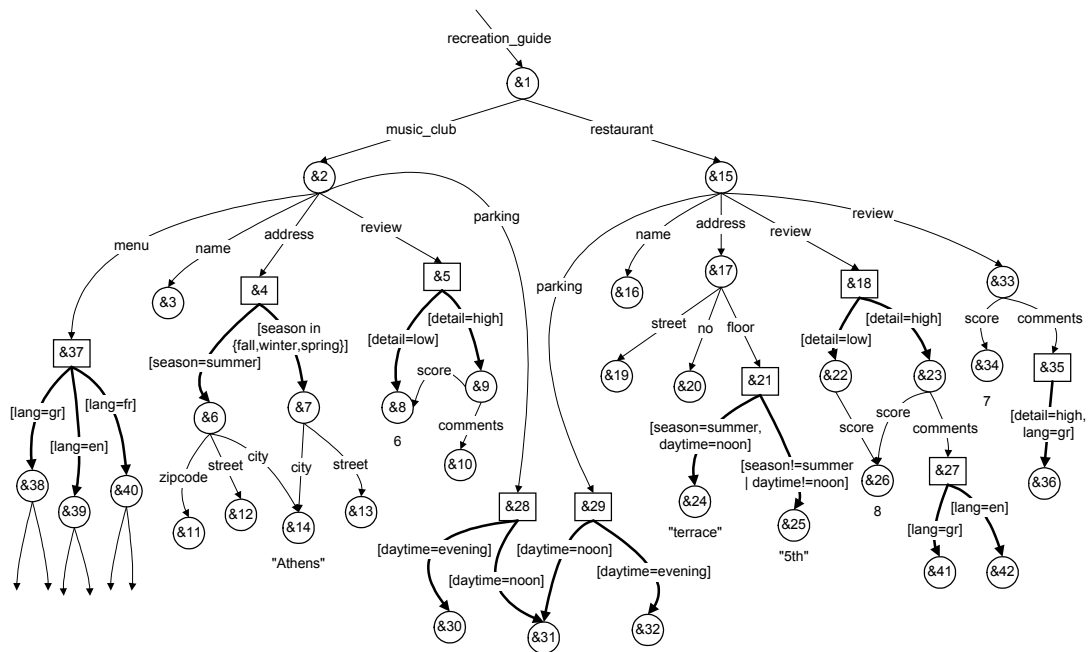
## 5.1 ΧΡΗΣΗ ΤΟΥ ΜΟΕΜ ΚΑΙ ΤΗΣ ΜQL

Στην ενότητα αυτή επιχειρηματολογούμε υπέρ της πρότασής μας για το ΜΟΕΜ και την ΜQL. Με απλά λόγια, μια **επερώτηση οδηγούμενη από το περιβάλλον** είναι μια επερώτηση όπου το ερμηνευτικό περιβάλλον είναι σημαντικό για την επιλογή των σωστών δεδομένων. Παρουσιάζουμε απλά παραδείγματα σε φυσική γλώσσα για να δώσουμε μια πρώτη εντύπωση επερωτήσεων οδηγούμενων από το περιβάλλον, και της διαφοράς τους από τις συμβατικές επερωτήσεις. Ισχυριζόμαστε ότι *σε ότι αφορά επερωτήσεις οδηγούμενες από το περιβάλλον η ΜQL είναι ισχυρότερη και πιο βολική στην χρήση από τις συμβατικές γλώσσες επερωτήσεων, ενώ ταυτόχρονα γίνεται το ίδιο απλή με αυτές όταν το ερμηνευτικό περιβάλλον δεν παίζει κάποιο ρόλο.* Ένα σημαντικό σημείο είναι ότι η ΜQL χρησιμοποιεί τα επιπλέον στοιχεία γράφου που ενσωματώνει το ΜΟΕΜ προκειμένου να απαντήσει σε επερωτήσεις οδηγούμενες από το περιβάλλον.

Για τα παραδείγματά μας θα χρησιμοποιήσουμε τον εξαρτώμενο από το ερμηνευτικό περιβάλλον οδηγό διασκέδασης που εξεξηγήθηκε στο προηγούμενο κεφάλαιο, ο οποίος απεικονίζεται και πάλι στην Εικόνα 5.1. Υποθέστε ότι μια βάση δεδομένων ΜΟΕΜ  $M$  ανάγεται σε μια βάση δεδομένων ΟΕΜ  $O_w$  κάτω από τον κόσμο  $w$ . Τότε, μια απλή επερώτηση ΜQL  $q = (q_w, w)$  στο  $M$  μπορεί να εκφραστεί σαν μια επερώτηση  $q_w$  στο  $O_w$ . Για παράδειγμα, ας είναι το  $M$  το ΜΟΕΜ στην Εικόνα 5.1, και ας θεωρήσουμε την επερώτηση  $q$  στο  $M$  «*δώσε μου τις διευθύνσεις των εστιατορίων τα μεσημέρια του καλοκαιριού σε χαμηλή λεπτομέρεια στα Ελληνικά*». Τότε η  $q$  είναι ισοδύναμη με την επερώτηση  $q_w$  «*δώσε μου τις διευθύνσεις των εστιατορίων*» πάνω στην ΟΕΜ έκφραση  $O_w$  του  $M$  η οποία υφίσταται κάτω από τον κόσμο  $w = \{(season, summer), (detail, low), (daytime, noon), (lang, gr)\}$ . Πάντως, η αναγωγή του  $M$  στο  $O_w$  δεν αποτελεί απαραίτητο βήμα για την αποτίμηση της  $q$ : η επεξεργασία της  $q$  μπορεί να λάβει χώρα απευθείας πάνω στο  $M$ , πράγμα που ίσως είναι προτιμότερο για λόγους απόδοσης. Διαισθητικά, η  $q_w$  μπορεί να χρησιμοποιηθεί για την πλοήγηση στις ακμές οντότητας του ΜΟΕΜ, ενώ το  $w$  μπορεί να καθοδηγήσει την πλοήγηση στις ακμές περιβάλλοντος του ΜΟΕΜ.



Εικόνα 5.1: Μια MOEM βάση δεδομένων για έναν οδηγό διασκέδασης εξαρτώμενο από το ερμηνευτικό περιβάλλον.



Εκτός από τέτοιου τύπου επερωτήσεις, το γεγονός ότι ο Πολυδιάστατος Γράφος Δεδομένων ομαδοποιεί μαζί τις εκφάνσεις των οντοτήτων επιτρέπει ένα διαφορετικό, «**διακοσμικό**» είδος επερωτήσεων οδηγούμενων από το περιβάλλον. Μια επερώτηση  $q$  αυτού του είδους δεν μπορεί να εκφραστεί με την μορφή  $(q_w, w)$  επειδή η πληροφορία που χρειάζεται για να απαντηθεί η  $q$  δεν περιέχεται σε ένα και μόνο OEM. Για παράδειγμα, θεωρήστε το αντικείμενο `music_club` στο  $M$ , και την ακόλουθη επερώτηση: «*δώσε μου το όνομα και την διεύθυνση τον χειμώνα ενός club για το οποίο είναι διαθέσιμη η διεύθυνσή του το καλοκαίρι*». Ή ακόμη, μια άλλη παραλλαγή: «*δώσε μου τα clubs που έχουν την ίδια διεύθυνση κατά την διάρκεια όλου του χρόνου*». Η τελευταία αυτή επερώτηση κοιτάζει εάν η διεύθυνση υφίσταται κάτω από ένα περιβάλλον που καλύπτει όλες τις δυνατές τιμές της διάστασης `season`. Συνεπώς, η MQL εκμεταλλεύεται τους πολυδιάστατους κόμβους για να θέσει επερωτήσεις που συσχετίζουν εκφάνσεις μιας και μόνο πολυδιάστατης οντότητας.

Η MQL βασίζεται σε και επεκτείνει ιδέες από γλώσσες επερωτήσεων που έχουν προταθεί για συμβατικά ημιδομημένα δεδομένα, κυρίως από την Lorel [AQM+97] και την UnQL [BFS00], των οποίων το μοντέλο δεδομένων είναι το OEM. Ιδιαίτερος η Lorel έχει και μια πρόσθετη XML παραλλαγή [GMW99], και είναι επίσης ισχυρή [BC00] και εύκολη στην χρήση, χαρακτηριστικά που είναι πολύ ελκυστικά. Όπως και στο Πολυδιάστατο OEM, το ερμηνευτικό περιβάλλον έχει κυρίαρχο ρόλο στην MQL: οι εκφράσεις μονοπατιού συνδυάζονται με πληροφορία περιβάλλοντος και δίνουν τις **εκφράσεις μονοπατιού περιβάλλοντος**, ενώ παράλληλα οι συνθήκες στην επερώτηση καθώς και η κατασκευή των αποτελεσμάτων μπορούν επίσης να εμπλέκουν ερμηνευτικά περιβάλλοντα. Σαν αποτέλεσμα, επερωτήσεις οδηγούμενες από το περιβάλλον που θα ήταν ιδιαίτερα δύσκολο ή αδύνατο να

σχηματιστούν σε μια γλώσσα επερωτήσεων για ημιδομημένα δεδομένα, μπορούν να εκφραστούν με οικονομία και κομψότητα χρησιμοποιώντας την MQL<sup>55</sup>.

Φυσικά, είναι δυνατόν να κωδικοποιήσει κανείς πληροφορία που εξαρτάται από το ερμηνευτικό περιβάλλον χρησιμοποιώντας ένα συμβατικό μοντέλο γράφου, όπως το OEM<sup>56</sup>, και να θέτει επερωτήσεις που περιέχουν πληροφορία σχετική με το περιβάλλον χρησιμοποιώντας μια γλώσσα σαν την Lorel. Το πλεονέκτημα της MQL είναι διττό.

- (a) *Συντακτικό*: στην MQL οι επερωτήσεις που οδηγούνται από το περιβάλλον είναι συνήθως πολύ πιο σύντομες και εύκολα αναγνώσιμες από τις ισοδύναμες επερωτήσεις σε Lorel, και συνεπώς πιο εύκολες στην σύνταξη και λιγότερο επιρρεπείς σε λάθη.
- (b) *Σημασιολογικό*: η MQL και το MOEM αναγνωρίζουν το ερμηνευτικό περιβάλλον σαν τέτοιο και επιτρέπουν την έκφραση σύνθετων συνθηκών που χρησιμοποιούν πράξεις περιβάλλοντος, κάτι που δεν υποστηρίζεται από την Lorel και το OEM.

Οι βελτιωμένες ικανότητες επερωτήσεων της MQL οφείλονται εν μέρει στις επεκτάσεις στο OEM που ενσωματώνει ο Πολυδιάστατος Γράφος Δεδομένων, δηλαδή στις ακμές περιβάλλοντος και στους πολυδιάστατους κόμβους. Κατά την γνώμη μας, οι επερωτήσεις που μπορεί να απαντήσει το Πολυδιάστατο OEM μέσω της MQL δικαιώνουν τις επεκτάσεις στο OEM που εισήχθησαν στο προηγούμενο κεφάλαιο. Επιπλέον, η δυνατότητα για «διακοσμικές» επερωτήσεις δείχνει ότι ένας γράφος MOEM είναι κάτι περισσότερο από το σύνολο των OEMs στα οποία μπορεί να αποσυντεθεί.

## 5.2 ΕΚΦΡΑΣΕΙΣ ΜΟΝΟΠΑΤΙΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ

Οι γλώσσες επερωτήσεων για ημιδομημένα δεδομένα χρησιμοποιούν εκφράσεις μονοπατιού προκειμένου να φθάσουν σε οποιοδήποτε βάθος στον γράφο δεδομένων [ABS00]. Παρόμοια, η MQL ενσωματώνει **εκφράσεις μονοπατιού περιβάλλοντος** οι οποίες ορίζουν πρότυπα πλοήγησης στους Πολυδιάστατους Γράφους Δεδομένων. Οι εκφράσεις μονοπατιού περιβάλλοντος είναι ο θεμέλιος λίθος της MQL.

Η Lorel κάνει την διάκριση [AQM+97] ανάμεσα στις απλές εκφράσεις μονοπατιού που επιτρέπουν την ανάκτηση αντικειμένων προδιαγράφοντας μια σειρά ετικετών στον γράφο, και στις γενικευμένες εκφράσεις μονοπατιού (general path expressions), μια ισχυρότερη μορφή με προχωρημένα χαρακτηριστικά όπως μπαλαντέρ (wildcards) και ομαλές εκφράσεις (regular expressions). Για λόγους σαφήνειας στην παρουσίαση, ακολουθούμε μια ανάλογη διάκριση ανάμεσα σε **απλές εκφράσεις μονοπατιού περιβάλλοντος** που εξηγούνται στην ενότητα αυτή, και σε **γενικευμένες εκφράσεις μονοπατιού περιβάλλοντος** που περιγράφονται στην Ενότητα 5.4.1.

Στην ενότητα αυτή αρχίζουμε παρουσιάζοντας τις εκφράσεις μονοπατιού περιβάλλοντος και εξηγώντας πως αποτιμούνται σε κόμβους. Στην συνέχεια εισάγουμε τους **περιοριστές περιβάλλοντος**, που χρησιμοποιούνται στις εκφράσεις μονοπατιού περιβάλλοντος για να δηλώνουν συνθήκες πάνω στο ερμηνευτικό περιβάλλον. Έπειτα, περιγράφουμε το πως είναι δυνατόν να υπονοούνται συγκεκριμένα τμήματα των εκφράσεων μονοπατιού περιβάλλοντος, οδηγώντας σε μορφές που είναι πιο ευανάγνωστες και πιο κοντινές στις συμβατικές

<sup>55</sup> Για μια σύγκριση ανάμεσα σε επερωτήσεις MQL και Lorel μπορεί κανείς να δει την Ενότητα 5.5.2, όπου συζητούμε την μετάφραση των επερωτήσεων MQL σε ισοδύναμες επερωτήσεις Lorel.

<sup>56</sup> Ένα παράδειγμα μιας τέτοιας κωδικοποίησης μπορεί κανείς να δει στην Ενότητα 5.5.1, όπου χρησιμοποιούμε το OEM προκειμένου να αναπαραστήσουμε το Πολυδιάστατο OEM.

εκφράσεις μονοπατιού. Κλείνουμε την ενότητα αυτή με μια συζήτηση πάνω στην σημασιολογία των εκφράσεων μονοπατιού περιβάλλοντος.

## 5.2.1 Ενσωμάτωση ερμηνευτικού περιβάλλοντος σε εκφράσεις μονοπατιού

Η απλή έκφραση μονοπατιού στην Lorel είναι μια ακολουθία  $x.l_1.l_2 \dots l_n$ , όπου τα  $l_1, l_2, \dots, l_n$  είναι ετικέτες, και το  $x$  είναι κάποιο αναγνωριστικό αντικείμενου ή μια μεταβλητή που δηλώνει κάποιο αντικείμενο. Παρόμοια, οι **εκφράσεις μονοπατιού περιβάλλοντος** αρχίζουν με ένα αναγνωριστικό αντικείμενο, ή με μια μεταβλητή που δηλώνει αντικείμενο<sup>57</sup>. Στο MOEM τα αναγνωριστικά αντικείμενων χωρίζονται στα δυο ξένα μεταξύ τους σύνολα των κόμβων περιβάλλοντος και των πολυδιάστατων κόμβων<sup>58</sup>, πράγμα που οδηγεί σε δυο ξεχωριστούς τύπους μεταβλητών για τα αντικείμενα: η μεταβλητή  $x$  υποδηλώνει έναν τύπο αντικείμενου περιβάλλοντος, ενώ η μεταβλητή  $\langle x \rangle$  υποδηλώνει έναν τύπο πολυδιάστατου αντικείμενου.

Οι εκφράσεις μονοπατιού περιβάλλοντος χρησιμοποιούν δυο είδη συστατικών στοιχείων για να επιτύχουν την πλοήγηση στις ακμές οντότητας και στις ακμές περιβάλλοντος, το **τμήμα οντότητας** και το **τμήμα έκφανσης**. Τα τμήματα οντότητας αρχίζουν με μια τελεία («.») και ορίζουν πρότυπα πλοήγησης πάνω σε ακμές οντότητας, ενώ τα τμήματα έκφανσης αρχίζουν με διπλή άνω και κάτω τελεία («.:») και ορίζουν πρότυπα πλοήγησης πάνω σε ακμές περιβάλλοντος. Θυμηθείτε ότι οι ακμές οντότητας ξεκινούν αποκλειστικά από κόμβους περιβάλλοντος, και οι ακμές περιβάλλοντος ξεκινούν αποκλειστικά από πολυδιάστατους κόμβους: συνεπώς, οι τελείες μπορεί να θεωρηθεί ότι αντιστοιχούν σε κόμβους περιβάλλοντος, ενώ οι διπλές άνω και κάτω τελείες ότι αντιστοιχούν σε πολυδιάστατους κόμβους.

### 5.2.1.1 Εκφράσεις μονοπατιού περιβάλλοντος και κανονική μορφή

Τα μονοπάτια σε έναν Πολυδιάστατο Γράφο Δεδομένων αποτελούνται από έναν αριθμό ακμών οντότητας και ακμών περιβάλλοντος *σε οποιαδήποτε σειρά*. Έτσι, εάν τα τμήματα οντότητας και τα τμήματα έκφανσης έπρεπε να έχουν μια αυστηρή αντιστοιχία με τις «συνταιριάζουσες» ακμές, θα ήταν κάποιος αναγκασμένος να γνωρίζει την ακριβή δομή του γράφου έτσι ώστε να σχηματίσει μια έκφραση μονοπατιού περιβάλλοντος με τα τμήματα οντότητας και τα τμήματα έκφανσης στην σωστή σειρά. Αυτό θα αποτελούσε πρόβλημα, μια και θα θέλαμε να είμαστε σε θέση να δημιουργούμε εκφράσεις μονοπατιού περιβάλλοντος χωρίς να έχουμε ακριβή γνώση της αλληλοδιάδοξης των ακμών οντότητας και των ακμών περιβάλλοντος σε έναν γράφο. Επιπλέον, όπως εξηγήθηκε στο προηγούμενο κεφάλαιο, διαφορετικές δομές γράφου μπορούν να είναι ισοδύναμες ως προς την πληροφορία που αναπαριστούν. Για παράδειγμα, συνεχόμενες ακμές περιβάλλοντος σε ένα μονοπάτι μπορούν να αντικατασταθούν από μια και μόνο ακμή περιβάλλοντος με κατάλληλο ρητό περιβάλλον<sup>59</sup>. Θα επιθυμούσαμε μια έκφραση μονοπατιού περιβάλλοντος να ταιριάζει με όλες τις ισοδύναμες δομές γράφου, χωρίς να χρειάζεται να περιγράψουμε εξαντλητικά όλες τις δυνατές αλληλοδιάδοχες των ακμών.

Για να επιτύχουμε τους σκοπούς αυτούς *χτίζουμε τις εκφράσεις μονοπατιού περιβάλλοντος γύρω από την κανονική μορφή ενός Πολυδιάστατου Γράφου Δεδομένων*. Παραθέτουμε από το

<sup>57</sup> Στην Ενότητα 5.2.1.1 εξηγήσαμε ότι μια έκφραση μονοπατιού περιβάλλοντος είναι επίσης δυνατόν να αρχίζει με ένα **τμήμα οντότητας** που να αντιστοιχεί στην ρίζα του γράφου.

<sup>58</sup> Όπως αναφέρθηκε στο προηγούμενο κεφάλαιο, οι όροι **κόμβος** και **αντικείμενο** χρησιμοποιούνται χωρίς διάκριση μεταξύ τους.

<sup>59</sup> Όπως εξηγήθηκε στο προηγούμενο κεφάλαιο, αυτό το ρητό περιβάλλον δίνεται από την τομή περιβάλλοντος των ρητών περιβαλλόντων των συνεχόμενων ακμών περιβάλλοντος.

προηγούμενο κεφάλαιο ότι, αν ένας Πολυδιάστατος Γράφος Δεδομένων βρίσκεται σε κανονική μορφή, «...κάθε δυνατό μονοπάτι σχηματίζεται από μια επαναλαμβανόμενη αλληλοδιάδοχη μιας ακμής περιβάλλοντος και μιας ακμής οντότητας. Δεν μπορούν να υπάρξουν περισσότερες από μια συνεχόμενες ακμές περιβάλλοντος ή συνεχόμενες ακμές οντότητας σε έναν γράφο που βρίσκεται σε κανονική μορφή». Ο σχηματισμός των εκφράσεων μονοπατιού περιβάλλοντος γύρω από την κανονική μορφή σημαίνει ότι έχουμε μια σταθερή σειρά ακμών οντότητας και περιβάλλοντος, ενάντια στην οποία να συνταιριάζουμε τμήματα οντότητας και έκφρασης. Ακόμη, σημαίνει ότι μια έκφραση μονοπατιού περιβάλλοντος έχει *ταυτόσημη αποτίμηση* για όλους τους γράφους που έχουν την ίδια κανονική μορφή.

Συνεπώς, η απλή έκφραση μονοπατιού περιβάλλοντος αποτελείται από ένα αναγνωριστικό αντικειμένου ή μεταβλητή αντικειμένου, που ακολουθείται από έναν αριθμό τμημάτων οντότητας και τμημάτων έκφρασης που διαδέχονται το ένα το άλλο. Μια μεταβλητή ή αναγνωριστικό αντικειμένου *περιβάλλοντος* θα πρέπει αμέσως να ακολουθείται από ένα τμήμα οντότητας, ενώ μια μεταβλητή ή αναγνωριστικό *πολυδιάστατου* αντικειμένου θα πρέπει να ακολουθείται από κάποιο τμήμα έκφρασης.

### Παράδειγμα 5.1

Τα ακόλουθα p1 μέχρι p5 είναι δυνατές μορφές απλών εκφράσεων μονοπατιού περιβάλλοντος, όπου  $.e_i$  δηλώνει τμήματα οντότητας και  $:f_i$  δηλώνει τμήματα έκφρασης.

p1 →  $o_{cxt} . e_1 :: f_1 . e_2 :: f_2 \dots e_n :: f_n$   
 p2 →  $X . e_1 :: f_1 . e_2 :: f_2 \dots e_n$   
 p3 →  $o_{mld} :: f_1 . e_2 :: f_2 . e_3 :: f_3 \dots e_n :: f_n$   
 p4 →  $\langle X \rangle :: f_1 . e_2 :: f_2 . e_3 :: f_3 \dots e_{n-1} :: f_{n-1} . e_n$   
 p5 →  $e_{root} :: f_1 . e_2 :: f_2 . e_3 :: f_3 \dots e_n :: f_n$

◆

Όπως φαίνεται στις εκφράσεις μονοπατιού περιβάλλοντος στο Παράδειγμα 5.1, τα τμήματα οντότητας και τα τμήματα έκφρασης εναλλάσσονται διαδοχικά. Η έκφραση μονοπατιού περιβάλλοντος p1 αρχίζει με το αναγνωριστικό αντικειμένου περιβάλλοντος  $o_{cxt}$ , που ακολουθείται άμεσα από το τμήμα οντότητας  $.e_1$  το οποίο συγκρίνεται με τις ακμές οντότητας που ξεκινούν από το  $o_{cxt}$ . Η έκφραση μονοπατιού περιβάλλοντος p2 αρχίζει με την μεταβλητή αντικειμένου περιβάλλοντος  $X$  αντί για κάποιο αναγνωριστικό. Παρατηρήστε ότι έχουμε αριθμήσει τα τμήματα οντότητας και έκφρασης σαν ζεύγη (ζεύγος 1, ζεύγος 2, κτλ.), αρχίζοντας από το τμήμα οντότητας. Διαισθητικά κάθε τέτοιο ζευγάρι «αντιστοιχεί» σε μια και μόνο ετικέτα  $.l_i$  των συμβατικών εκφράσεων μονοπατιού. Με τρόπο ανάλογο του p1, το παράδειγμα p3 αρχίζει με το αναγνωριστικό πολυδιάστατου αντικειμένου  $o_{mld}$ , που ακολουθείται από το τμήμα έκφρασης  $:f_1$  το οποίο συγκρίνεται με τις ακμές περιβάλλοντος που ξεκινούν από το  $o_{mld}$ , ενώ το p4 αρχίζει με την μεταβλητή πολυδιάστατου αντικειμένου  $\langle X \rangle$  στην θέση κάποιου αναγνωριστικού. Μια έκφραση μονοπατιού περιβάλλοντος τελειώνει είτε με ένα τμήμα οντότητας είτε με ένα τμήμα έκφρασης.

Η περίπτωση του p5 επιδεικνύει μια βολική σύμβαση και για το OEM και για το MOEM: η ρίζα του γράφου αναπαρίσταται από μια μοναδική ακμή που δείχνει στην ρίζα αλλά δεν ξεκινά από κανέναν κόμβο. Η ακμή αυτή έχει σαν ετικέτα το όνομα της συγκεκριμένης βάσης δεδομένων, όπως στην Εικόνα 5.1 όπου η ρίζα δείχνεται από μια ακμή με την ετικέτα *recreation\_guide*. Η ρίζα της κανονικής μορφής ενός Πολυδιάστατου Γράφου Δεδομένων είναι πάντοτε πολυδιάστατος κόμβος, συνεπώς η ακμή για την οποία μιλούμε είναι πάντοτε ακμή οντότητας, όπως υπονοείται από το τμήμα οντότητας  $e_{root}$  στο p5. Σημειώστε ότι στις εκφράσεις μονοπατιού περιβάλλοντος το  $e_{root}$  είναι ένα ειδικό τμήμα

οντότητας που δεν περιέχει «.», ενώ σε αντιδιαστολή στις συμβατικές εκφράσεις μονοπατιού η «ετικέτα» της ρίζας θεωρείται αναγνωριστικό αντικειμένου.

### 5.2.1.2 Αποτίμηση εκφράσεων μονοπατιού περιβάλλοντος

Μια έκφραση μονοπατιού περιβάλλοντος αποτιμάται είτε σε ένα σύνολο αντικειμένων περιβάλλοντος είτε σε ένα σύνολο πολυδιάστατων αντικειμένων. Αυτό εξαρτάται από το τελευταίο τμήμα της έκφρασης μονοπατιού περιβάλλοντος: αν το τελευταίο τμήμα είναι τμήμα έκφρασης, όπως στα  $p_1$ ,  $p_3$ , και  $p_5$  στο Παράδειγμα 5.1, τότε η έκφραση μονοπατιού περιβάλλοντος αποτιμάται σε ένα σύνολο κόμβων περιβάλλοντος: από την άλλη, αν το τελευταίο τμήμα είναι τμήμα οντότητας, όπως στα  $p_2$  και  $p_4$ , τότε αποτιμάται σε ένα σύνολο πολυδιάστατων κόμβων.

Στο OEM, ένα μονοπάτι δεδομένων (data path) [AQM+97] είναι μια ακολουθία  $o_0, l_1, o_1, l_2, o_2, \dots, l_n, o_n$ , τέτοια ώστε υπάρχει μια ακμή  $l_k$  από το αντικείμενο  $o_{k-1}$  στο αντικείμενο  $o_k$ . Δεδομένου ενός OEM γράφου, είναι δυνατόν να υπάρχουν κανένα, ένα, ή αρκετά τέτοια μονοπάτια που να ταιριάζουν με την απλή έκφραση μονοπατιού  $X.l_1.l_2 \dots l_n$  για  $X = o_0$ . Ανάλογα με τα μονοπάτια δεδομένων, ένα **μονοπάτι δεδομένων περιβάλλοντος** στον Πολυδιάστατο Γράφο Δεδομένων είναι μια ακολουθία  $o_0, [icv_1]l_1, o_1, [icv_2]l_2, o_2, \dots, [icv_n]l_n, o_n$ , τέτοια ώστε υπάρχει μια ακμή με ετικέτα  $l_k$  και με κληρονομούμενη κάλυψη  $[icv_k]$  από το αντικείμενο  $o_{k-1}$  στο αντικείμενο  $o_k$ . Σημειώστε ότι αν η ακμή είναι ακμή περιβάλλοντος, τότε η ετικέτα είναι το ρητό της περιβάλλον. Παρόλο που οι κληρονομούμενες καλύψεις δεν είναι απαραίτητες για την ταυτοποίηση των ακμών, είναι χρήσιμες για την σύγκριση των μονοπατιών δεδομένων περιβάλλοντος με μια έκφραση μονοπατιού περιβάλλοντος, όπως θα εξηγηθεί αργότερα. Δεδομένου ενός Πολυδιάστατου Γράφου Δεδομένων, μπορούν να υπάρχουν κανένα, ένα, ή αρκετά μονοπάτια δεδομένων περιβάλλοντος που να ταιριάζουν με μια έκφραση μονοπατιού περιβάλλοντος. Οι τελευταίοι κόμβοι των συνταιριαζόντων μονοπατιών δεδομένων περιβάλλοντος αποτελούν το σύνολο του αποτελέσματος της έκφρασης μονοπατιού περιβάλλοντος.

Παρόλο που θεωρήσαμε ότι οι εκφράσεις μονοπατιού περιβάλλοντος στοχεύουν βάσεις δεδομένων MOEM που βρίσκονται σε κανονική μορφή, είναι σημαντικό να συνειδητοποιήσει κανείς ότι δεν είναι απαραίτητο να μετασχηματίσουμε έναν γράφο σε κανονική μορφή προκειμένου να αποτιμήσουμε μια έκφραση μονοπατιού περιβάλλοντος. Όταν ο γράφος είναι σε κανονική μορφή, τα μονοπάτια δεδομένων περιβάλλοντος επιδεικνύουν την ίδια κανονικότητα πως οι εκφράσεις μονοπατιού περιβάλλοντος, με μια ακμή οντότητας να διαδέχεται μια ακμή περιβάλλοντος συνεχώς. Στην περίπτωση αυτή υπάρχει μια ευθεία αντιστοιχία ανάμεσα από τα διακεκριμένα στοιχεία των μονοπατιών δεδομένων περιβάλλοντος και των εκφράσεων μονοπατιού περιβάλλοντος, η οποία διευκολύνει την διαδικασία της σύγκρισης. Στην γενική περίπτωση πάντως όπου ένας γράφος δεν είναι σε κανονική μορφή, η διαδικασία σύγκρισης μπορεί να προσαρμόσει κατάλληλα την αντιστοιχία ανάμεσα από τα διακεκριμένα στοιχεία των μονοπατιών δεδομένων περιβάλλοντος και των εκφράσεων μονοπατιού περιβάλλοντος ώστε να «προσομοιάσει» μια κανονική μορφή<sup>60</sup>. Όπως θα δούμε στην Ενότητα 5.2.3, συγκεκριμένα τμήματα των εκφράσεων μονοπατιού περιβάλλοντος μπορούν να παραλειφθούν, πράγμα που περιπλέκει

<sup>60</sup> Για παράδειγμα, ένα τμήμα έκφρασης μπορεί να συγκριθεί με μια ακολουθία συνεχόμενων ακμών περιβάλλοντος ενός γράφου που δεν είναι σε κανονική μορφή. Ένα λεπτό σημείο είναι ότι η κανονική μορφή μπορεί να περιέχει πρόσθετους πολυδιάστατους κόμβους, οι οποίοι μπορεί να αποτελούν τον στόχο εκφράσεων μονοπατιού περιβάλλοντος. Μια διεργασία αποτίμησης που θεωρεί ότι ο γράφος δεν είναι σε κανονική μορφή θα πρέπει να υλοποιήσει ένα συνεπές oid σχήμα για τέτοιους «υπονοούμενους» πολυδιάστατους κόμβους.

περαιτέρω το ζήτημα. Για να κρατήσουμε τα πράγματα απλά, σε ότι ακολουθεί θεωρούμε ότι οι βάσεις δεδομένων MOEM βρίσκονται σε κανονική μορφή.

### 5.2.1.3 Τμήματα οντότητας και τμήματα έκφρασης

Όπως τονίσαμε, για να καθορίσουμε αν ένα μονοπάτι δεδομένων περιβάλλοντος ταιριάζει με μια έκφραση μονοπατιού περιβάλλοντος, οι ακμές οντότητας συγκρίνονται με τμήματα οντότητας και οι ακμές περιβάλλοντος συγκρίνονται με τμήματα έκφρασης στην σειρά. Ένα **τμήμα οντότητας** έχει είτε την μορφή `.1` ή την μορφή `.[cqicv]1`, όπου 1 είναι μια ετικέτα ακμής οντότητας και `[cqicv]` είναι ένας **περιοριστής κληρονομούμενης κάλυψης**. Ένα **τμήμα έκφρασης** έχει είτε την μορφή `::[cqec]` ή την μορφή `::[cqicv][cqec]`, όπου `[cqec]` είναι ένας **περιοριστής ρητού περιβάλλοντος** και `[cqicv]` είναι και πάλι ένας περιοριστής κληρονομούμενης κάλυψης. Οι περιοριστές ρητού περιβάλλοντος αποτελούν απαραίτητο στοιχείο των τμημάτων έκφρασης, και συγκρίνονται με το ρητό περιβάλλον των αντιστοίχων ακμών περιβάλλοντος. Οι περιοριστές κληρονομούμενης κάλυψης είναι προαιρετικοί και για τα τμήματα οντότητας και για τα τμήματα έκφρασης, και συγκρίνονται με την κληρονομούμενη κάλυψη ενός μονοπατιού, δηλαδή με την *κληρονομούμενη κάλυψη μονοπατιού* (που ορίστηκε στο προηγούμενο κεφάλαιο). Οι περιοριστές κληρονομούμενης κάλυψης και οι περιοριστές ρητού περιβάλλοντος έχουν την ίδια σύνταξη, και είναι στιγμιότυπα των **περιοριστών περιβάλλοντος**. Οι περιοριστές περιβάλλοντος είναι συντακτικές κατασκευές που χρησιμοποιούνται προκειμένου να εκφράσουν συνθήκες πάνω στο ερμηνευτικό περιβάλλον. Στην απλούστερη περίπτωση, ένας περιοριστής περιβάλλοντος δεν είναι παρά ένας προσδιοριστής περιβάλλοντος που δηλώνει τους κόσμους τους οποίους θα πρέπει να καλύπτει ένα περιβάλλον για να ταιριάζει.

#### Παράδειγμα 5.2

Οι ακόλουθες εκφράσεις μονοπατιού περιβάλλοντος αναφέρονται στην κανονική μορφή του γράφου στην Εικόνα 5.1.

```
p1→ [lang=gr]recreation_guide::[-].music_club::[-].menu::[-]
p2→ [-]recreation_guide::[-].music_club::[-].menu::[lang=gr]
p3→ [-]recreation_guide::[-].music_club::[-].menu::[-]
p4→ []recreation_guide::[-].music_club::[-].address
p5→ []recreation_guide::[-].music_club::[-]
      .review::[-][detail=high]
```

◆

Στην έκφραση μονοπατιού περιβάλλοντος `p1` στο Παράδειγμα 5.2, ο περιοριστής περιβάλλοντος `[lang=gr]` συγκρίνεται με την κληρονομούμενη κάλυψη μονοπατιού των μονοπατιών που ξεκινούν από την ρίζα και καταλήγουν σε κάποια έκφραση του `menu`. Ειδικότερα, ταιριάζει με εκείνες τις κληρονομούμενες καλύψεις μονοπατιού που είναι *υπερσύνολο περιβάλλοντος* του `[lang=gr]`. Οι υπόλοιποι περιοριστές περιβάλλοντος στο `p1` είναι περιοριστές ρητού περιβάλλοντος, και ταιριάζουν με κάθε ακμή περιβάλλοντος, επειδή κάθε περιβάλλον είναι υπερσύνολο περιβάλλοντος του κενού περιβάλλοντος `[-]`. Στην βάση MOEM στην Εικόνα 5.1, το `p1` αποτιμάται στο σύνολο `{&38}`. Στην επόμενη περίπτωση, το `p2` τυχαίνει επίσης να αποτιμάται στο `{&38}`: ο περιοριστής κληρονομούμενης κάλυψης είναι τώρα `[-]`, που ταιριάζει με κάθε κληρονομούμενη κάλυψη μονοπατιού, ενώ ο τελευταίος περιοριστής ρητού περιβάλλοντος είναι `[lang=gr]`, που ταιριάζει με ρητά περιβάλλοντα που είναι *υπερσύνολο περιβάλλοντος* του `[lang=gr]`. Παρόλο που τα `p1` και `p2` τυχαίνει να αποτιμώνται στο ίδιο αποτέλεσμα για την συγκεκριμένη βάση MOEM, έχουν διαφορετικό νόημα: το `p1` επιστρέφει τις εκφάνσεις του `menu facets` που είναι προσβάσιμες από την ρίζα μέσω ενός μονοπατιού το οποίο υφίσταται (τουλάχιστο) κάτω από το

[lang=gr]<sup>61</sup>, ενώ το p2 επιστρέφει τις εκφάνσεις του menu με αντίστοιχο ρητό περιβάλλον (τουλάχιστο) [lang=gr] άσχετα με το εάν ή όχι οι εκφάνσεις αυτές υφίστανται στην πραγματικότητα κάτω από κάποιον κόσμο. Στην περίπτωση του p3 όλοι οι περιοριστές περιβάλλοντος είναι κενά περιβάλλοντα, συνεπώς κάθε έκφραση του μενού ταιριάζει με την έκφραση μονοπατιού περιβάλλοντος, και το αποτέλεσμα είναι {&38, &39, &40}. Σημειώστε ότι δεν λαμβάνει χώρα κάποια ομαδοποίηση των αποτελεσμάτων: αν το music club στην Εικόνα 5.1 περιείχε περισσότερα από ένα πολυδιάστατα αντικείμενα menu, οι εκφάνσεις του menu που θα ταιρίαζαν θα επιστρέφονταν σε ένα επίπεδο σύνολο αποτελεσμάτων, χωρίς καμία ένδειξη σχετικά με το ποια αντικείμενα του συνόλου αποτελούν εκφάνσεις του ίδιου μενού.

Η έκφραση μονοπατιού περιβάλλοντος p4 στο Παράδειγμα 5.2 επιστρέφει ένα σύνολο *πολυδιάστατων* κόμβων, συγκεκριμένα το σύνολο {&4}. Ο περιοριστής κληρονομούμενης κάλυψης [] στην αρχή του p4 απαιτεί ότι ένα μονοπάτι που οδηγεί από την ρίζα σε κάποιον πολυδιάστατο κόμβο address θα πρέπει να υφίσταται κάτω από κάθε πιθανό κόσμο. Η περίπτωση του p5 δείχνει ότι μια έκφραση μονοπατιού περιβάλλοντος μπορεί να περιέχει πολλούς περιοριστές κληρονομούμενης κάλυψης. Ψάχνει για αντικείμενα review που είναι προσβάσιμα μέσω ενός μονοπατιού το οποίο υφίσταται κάτω από κάθε πιθανό κόσμο, και στην συνέχεια επιστρέφει τις εκφάνσεις του review που έχουν σαν αντίστοιχο ρητό περιβάλλον (τουλάχιστον) το [detail=high] άσχετα με τους κόσμους κάτω από τους οποίους υφίστανται στην πραγματικότητα. Ο πρώτος περιοριστής περιβάλλοντος [] του p5 είναι περιοριστής κληρονομούμενης κάλυψης, και συγκρίνεται με την κληρονομούμενη κάλυψη μονοπατιού των μονοπατιών που καταλήγουν σε πολυδιάστατους κόμβους review. Ο δεύτερος περιοριστής κληρονομούμενης κάλυψης είναι [-], και αφορά στις ακμές περιβάλλοντος που ξεκινούν από πολυδιάστατους κόμβους review. Στην περίπτωση αυτή το μονοπάτι αποτελείται από μια και μόνον ακμή περιβάλλοντος. Το νόημα αυτού του δεύτερου περιοριστή κληρονομούμενης κάλυψης στο p5 είναι ότι οι ακμές περιβάλλοντος που δείχνουν σε εκφάνσεις του review μπορούν στην πραγματικότητα να υφίστανται κάτω από οποιοδήποτε σύνολο κόσμων. Έτσι, *ένας περιοριστής κληρονομούμενης κάλυψης σημειώνει την αρχή του μονοπατιού στο οποίο αφορά και το τέλος του μονοπατιού για τον προηγούμενο περιοριστή κληρονομούμενης κάλυψης* στην ίδια έκφραση μονοπατιού περιβάλλοντος.

Οι περιοριστές περιβάλλοντος κάνουν τις εκφράσεις μονοπατιού περιβάλλοντος κάπως φλύαρες και δυσκολονόητες, όπως όμως θα δούμε στην Ενότητα 5.2.3 στις περισσότερες περιπτώσεις μπορούμε να τους υπονοήσουμε, πράγμα που οδηγεί σε πιο εύκολες και πιο κατανοητές μορφές των εκφράσεων μονοπατιού περιβάλλοντος. Προτού εξηγήσουμε σε ποιες περιπτώσεις είναι δυνατόν να παραλειφθούν οι περιοριστές περιβάλλοντος, παρουσιάζουμε λεπτομερώς τους περιοριστές περιβάλλοντος.

## 5.2.2 Περιοριστές περιβάλλοντος

Οι **περιοριστές περιβάλλοντος** χρησιμοποιούνται στις εκφράσεις μονοπατιού περιβάλλοντος για να εκφράσουν συνθήκες τις οποίες θα πρέπει να ικανοποιούν ερμηνευτικά περιβάλλοντα που σχετίζονται με ακμές ή με μονοπάτια του γράφου. Ένας περιοριστής περιβάλλοντος μπορεί να έχει την μορφή ενός προσδιοριστή περιβάλλοντος, ενός **προτύπου περιβάλλοντος**, ή μιας **μεταβλητής περιβάλλοντος**.

<sup>61</sup> Με άλλα λόγια, το p1 επιστρέφει κόμβους menu που υπάρχουν σε κάθε συμβατικό OEM που υφίσταται κάτω από κάποιον κόσμο που ανήκει στο [lang=gr]. Αυτό εξηγηθείτε σε μεγαλύτερη λεπτομέρεια στην Ενότητα 5.2.4.

### 5.2.2.1 Περιοριστές περιβάλλοντος σαν προσδιοριστές περιβάλλοντος

Ένας περιοριστής περιβάλλοντος που έχει την μορφή ενός προσδιοριστή περιβάλλοντος  $c_{cq}$  εκφράζει μια συνθήκη απέναντι σε κάποιο περιβάλλον  $c$ , η οποία δηλώνει ότι προκειμένου να ταιριάζουν το  $c_{cq}$  και το  $c$ , το  $c$  πρέπει να είναι *υπερσύνολο περιβάλλοντος* του  $c_{cq}$ . Με άλλα λόγια, όλοι οι κόσμοι που αναπαριστά ο περιοριστής περιβάλλοντος πρέπει να καλύπτονται από το περιβάλλον για να ταιριάζει<sup>62</sup>. Σύμφωνα με αυτό, η έκφραση μονοπατιού περιβάλλοντος  $o.[c_1]1b::[c_2][c_3]$  ταιριάζει με τα μονοπάτια που, ξεκινώντας από το  $o$ , αποτελούνται από μια ακμή οντότητας με ετικέτα  $1b$  και με κληρονομούμενη κάλυψη<sup>63</sup>  $icv_{ett} \geq [c_1]$ , ακολουθούμενη από μια ακμή περιβάλλοντος με κληρονομούμενη κάλυψη  $icv_{cxt} \geq [c_2]$  και ρητό περιβάλλον  $ec_{cxt} \geq [c_3]$ .

Μια και ο κενός προσδιοριστής περιβάλλοντος  $[-]$  είναι υποσύνολο οποιουδήποτε περιβάλλοντος, όταν χρησιμοποιείται σαν περιοριστής περιβάλλοντος παίζει τον ρόλο ενός **μπαλαντέρ περιβάλλοντος** που ταιριάζει με κάθε περιβάλλον ακμής. Χρησιμοποιώντας το κενό περιβάλλον με αυτόν τον τρόπο στην ουσία «απενεργοποιούμε» τις συνθήκες πάνω στο περιβάλλον. Για παράδειγμα, στην έκφραση μονοπατιού περιβάλλοντος  $x.[-]1b::[c] [-]$  το τμήμα οντότητας  $[-]1b$  θα ταιριάζει με όλες τις ακμές οντότητας που έχουν ετικέτα  $1b$  ανεξάρτητα από την κληρονομούμενη κάλυψή τους, ενώ το τμήμα έκφρασης  $::[c] [-]$  θα ταιριάζει με όλες τις ακμές περιβάλλοντος με κληρονομούμενη κάλυψη υπερσύνολο του  $[c]$  ανεξάρτητα από το ρητό περιβάλλον τους. Σαν άλλο ένα παράδειγμα θεωρήστε την έκφραση  $\langle x \rangle :: [-] [-].1b$ , όπου επιτρέπει σε μια ακμή περιβάλλοντος να έχει οποιοδήποτε ρητό περιβάλλον, και μαζί με μια ακμή οντότητας  $1b$  να έχουν οποιαδήποτε κληρονομούμενη κάλυψη μονοπατιού. Σύμφωνα με τα παραπάνω, το τμήμα έκφρασης  $::[-] [-]$  ταιριάζει με οποιαδήποτε ακμή περιβάλλοντος.

Όπως έχει αναφερθεί, οι περιοριστές κληρονομούμενης κάλυψης αφορούν μονοπάτια που αποτελούνται από μια ή περισσότερες ακμές, συνεπώς μπορεί να επηρεάζουν τα επόμενα τμήματα οντότητας και έκφρασης. Για παράδειγμα, στην έκφραση μονοπατιού περιβάλλοντος  $x.[c_1]1b::[c_2]$  ο περιοριστής περιβάλλοντος  $[c_1]$  δεν συγκρίνεται με την κληρονομούμενη κάλυψη του  $1b$ , αλλά με την κληρονομούμενη κάλυψη μονοπατιού ενός μονοπατιού που απαρτίζεται από: (a) μια ακμή οντότητας με ετικέτα  $1b$ , και (b) μια ακμή περιβάλλοντος με ρητό περιβάλλον που καλύπτει το  $[c_2]$ . Πάντως είναι ενδιαφέρον να παρατηρήσει κανείς ότι, αν ένας περιοριστής κληρονομούμενης κάλυψης έχει την μορφή προσδιοριστή περιβάλλοντος, μπορεί να επαναληφθεί κατά μήκος ολόκληρου του αντίστοιχου μονοπατιού χωρίς να αλλάξει η αποτίμηση της έκφρασης μονοπατιού περιβάλλοντος. Με αυτή την βάση, αν  $[c_1]$  και  $[c_3]$  είναι προσδιοριστές περιβάλλοντος, η έκφραση μονοπατιού περιβάλλοντος  $x.[c_1]a::[c_2].b::[c_3][c_4].d$  είναι ισοδύναμη με την έκφραση μονοπατιού περιβάλλοντος  $x.[c_1]a::[c_1][c_2].[c_1]b::[c_3][c_4].[c_3]d$ , στην οποία κάθε περιοριστής κληρονομούμενης κάλυψης συγκρίνεται με την

<sup>62</sup> Αυτό είναι πιο περιοριστικό από το να απαιτήσουμε δυο περιβάλλοντα να περιέχουν τουλάχιστον έναν κοινό κόσμο. Εξαιρούμενης της οριακής περίπτωσης των κενών περιβαλλόντων, στην κανονική μορφή ενός *αιτιοκρατικού ως προς το περιβάλλον* γράφου, η συνθήκη ότι η κληρονομούμενη κάλυψη μιας ακμής περιβάλλοντος πρέπει να είναι υπερσύνολο περιβάλλοντος κάποιου περιβάλλοντος μπορεί να ικανοποιηθεί από μια το πολύ ακμή περιβάλλοντος μέσα στα πλαίσια οποιασδήποτε πολυδιάστατης οντότητας. Έτσι, στην περίπτωση των αιτιοκρατικών ως προς το περιβάλλον γράφων (και υποθέτοντας μη κενά περιβάλλοντα) το πολύ μια ακμή περιβάλλοντος μπορεί να ταιριάζει με έναν περιοριστή κληρονομούμενης κάλυψης. Σαν αποτέλεσμα, αυτή η «πιο περιοριστική» προσέγγιση επιτρέπει στους περιοριστές περιβάλλοντος να προσδιορίζουν με μοναδικό τρόπο εκφάνσεις των πολυδιάστατων οντοτήτων.

<sup>63</sup> Όταν ένα μονοπάτι αποτελείται από μια και μόνον ακμή, η κληρονομούμενη κάλυψη μονοπατιού του είναι η κληρονομούμενη κάλυψη της ακμής.



κληρονομούμενη κάλυψη μιας και μόνον ακμής. Αυτό αποδεικνύεται στην ακόλουθη πρόταση:

### Πρόταση 5.1

*Ας είναι  $p$  ένας Πολυδιάστατος Γράφος Δεδομένων, και  $c$  ένας περιοριστής περιβάλλοντος που έχει την μορφή προσδιοριστή περιβάλλοντος. Τότε, το  $c$  ταιριάζει με την κληρονομούμενη κάλυψη μονοπατιού του  $p$  εάν και μόνον το  $c$  ταιριάζει με τις κληρονομούμενες καλύψεις όλων των ακμών στο  $p$ .*

Πρέπει να τονιστεί ότι η ιδιότητα αυτή των περιοριστών κληρονομούμενης κάλυψης ισχύει μόνον εφόσον έχουν την μορφή προσδιοριστών περιβάλλοντος. Περιοριστές κληρονομούμενης κάλυψης που είναι πρότυπα περιβάλλοντος ή μεταβλητές περιβάλλοντος δεν δίνουν ισοδύναμες εκφράσεις μονοπατιού περιβάλλοντος όταν επαναληφθούν κατά μήκος του μονοπατιού που αφορούν.

### 5.2.2.2 Πρότυπα περιβάλλοντος

Τα **πρότυπα περιβάλλοντος** μας επιτρέπουν να εκφράσουμε περιβάλλοντα καθορίζοντας μόνο κάποιες από τις διαστάσεις, ενώ οι υπόλοιπες μπορούν να λάβουν οποιαδήποτε τιμή. Η σύνταξη των προτύπων περιβάλλοντος είναι πανομοιότυπη με αυτή των προσδιοριστών περιβάλλοντος, με την διαφορά ότι: (a) ο χαρακτήρας «%» ακολουθεί το αριστερό τετράγωνο άγκιστρο «[» σηματοδοτώντας ότι πρόκειται για πρότυπο περιβάλλοντος, και (b) ένας όρος στο πρότυπο περιβάλλοντος πρέπει να περιέχει τουλάχιστον έναν προσδιοριστή διάστασης, και δεν μπορεί να είναι «-» (κενός όρος) ή «» (καθολικός όρος)<sup>64</sup>. Το πρότυπο δεν λειτουργεί σε συντακτικό επίπεδο, αλλά στο επίπεδο των κόσμων, όπως επιδεικνύει το ακόλουθο παράδειγμα.

### Παράδειγμα 5.3

Θεωρήστε ότι:

$V_{\text{lang}} = \{en, gr, sp\}$

$V_{\text{detail}} = \{low, high\}$

$V_{\text{format}} = \{ps, pdf, html\}$

$c_1 = [detail=low, lang \text{ in } \{en,gr\}, format=pdf]$

$c_2 = [detail=low, format=pdf \mid lang=en, format=ps \mid lang \text{ in } \{gr,sp\}]$

Τότε, τα ακόλουθα είναι συνθήκες περιβάλλοντος που χρησιμοποιούν πρότυπα περιβάλλοντος:

$cond_1: [\% lang=gr] \leq c_1$  (αληθής)

$cond_2: [\% lang \text{ in } \{gr,en\}, detail=low] = c_1$  (αληθής)

$cond_3: [\% lang \text{ in } \{gr,en\}, detail=low] = c_2$  (ψευδής)

$cond_4: [\% lang=en, detail=high \mid lang=gr, detail=low] \leq c_2$  (αληθ.)

◆

Στο Παράδειγμα 5.3, η συνθήκη  $cond_1$  ισχύει. Απαιτεί ότι το περιβάλλον  $c_1$  καλύπτει κάποιον κόσμο όπου η γλώσσα είναι Ελληνικά, ανεξάρτητα από την τιμή των άλλων διαστάσεων. Το σύμβολο  $\leq$  του υποσυνόλου περιβάλλοντος δηλώνει ότι το  $lang$  μπορεί να πάρει τιμές διαφορετικές από το  $gr$  σε άλλους κόσμους που καλύπτει το  $c_1$ , από την στιγμή

<sup>64</sup> Ένα πρότυπο περιβάλλοντος μπορεί να περιέχει κενούς όρους και καθολικούς όρους. Για παράδειγμα, προσδιοριστές διάστασης όπως το  $d \text{ in } \{ \}$  και το  $d \text{ in ALL}$  μπορούν να χρησιμοποιηθούν για τις αντίστοιχες περιπτώσεις.

που σε κάποιον κόσμο του  $c_1$  η τιμή του  $\text{lang}$  είναι  $\text{gr}$ . Η συνθήκη  $\text{cond}_2$  είναι επίσης αληθής. Δηλώνει ότι το  $c_1$  πρέπει να καλύπτει: (a) κόσμους που έχουν γλώσσα Ελληνικά και χαμηλή λεπτομέρεια, μαζί με (b) κόσμους που έχουν γλώσσα Αγγλικά και χαμηλή λεπτομέρεια, ανεξάρτητα από την τιμή των άλλων διαστάσεων. Το σύμβολο  $=$  της ισότητας περιβάλλοντος υπονοεί ότι *κάθε* κόσμος στο  $c_1$  πρέπει να πέφτει σε μια από τις δυο αυτές κατηγορίες. Η συνθήκη  $\text{cond}_3$  είναι ψευδής, επειδή το περιβάλλον  $c_2$  καλύπτει κόσμους που δεν ταιριάζουν με το πρότυπο, όπως για παράδειγμα ο κόσμος  $\{(\text{lang}, \text{en}), (\text{detail}, \text{high}), (\text{format}, \text{ps})\}$ . Η συνθήκη  $\text{cond}_4$  είναι αληθής. Χρησιμοποιεί ένα πρότυπο περιβάλλοντος με δυο όρους, και απαιτεί το  $c_2$  να καλύπτει κάποιον κόσμο όπου η γλώσσα είναι Αγγλικά και η λεπτομέρεια είναι υψηλή, *αλλά επίσης* και κάποιον άλλο κόσμο όπου η γλώσσα είναι Ελληνικά και η λεπτομέρεια χαμηλή<sup>65</sup>. Σε μια συνθήκη περιβάλλοντος, ένα πρότυπο περιβάλλοντος μπορεί να συγκριθεί με έναν προσδιοριστή περιβάλλοντος για ισότητα / ανισότητα περιβάλλοντος, ή (γνήσιο) υποσύνολο / υπερσύνολο περιβάλλοντος.

Ένας άμεσος τρόπος για να αποτιμήσουμε συνθήκες περιβάλλοντος που περιέχουν πρότυπα περιβάλλοντος είναι να τις μετασχηματίσουμε σε ισοδύναμες συνθήκες περιβάλλοντος που δεν περιέχουν πρότυπα περιβάλλοντος. Μια μη αυστηρή περιγραφή είναι ότι, απομακρύνουμε τον χαρακτήρα «%» από το πρότυπο περιβάλλοντος, και παράλληλα απομακρύνουμε όλους τους προσδιοριστές διάστασης από το άλλο μέρος (τον προσδιοριστή περιβάλλοντος) που δεν αντιστοιχούν σε κάποια διάσταση στο πρότυπο περιβάλλοντος<sup>66</sup>. Για παράδειγμα, το  $\text{cond}_3$  στο Παράδειγμα 5.3 θα δώσει  $[\text{lang in } \{\text{gr}, \text{en}\}, \text{detail}=\text{low}] = [\text{detail}=\text{low} \mid \text{lang}=\text{en} \mid \text{lang in } \{\text{gr}, \text{sp}\}]$ , που είναι μια συνθήκη περιβάλλοντος που περιέχει μόνο προσδιοριστές περιβάλλοντος. Αυτή η ισότητα περιβάλλοντος δεν είναι αληθής, συνεπώς και η αρχική συνθήκη που περιέχει το πρότυπο περιβάλλοντος δεν ισχύει.

Τα πρότυπα περιβάλλοντος μπορούν να συμμετέχουν και στην τομή περιβάλλοντος, στην ένωση περιβάλλοντος, και στην διαφορά περιβάλλοντος. Στην περίπτωση αυτή, *τα πρότυπα περιβάλλοντος χρησιμοποιούνται για να «φιλτράρουν» προσδιοριστές περιβάλλοντος*, απομακρύνοντας διαστάσεις και επηρεάζοντας τις τιμές των διαστάσεων όπως δείχνει το παρακάτω παράδειγμα.

#### Παράδειγμα 5.4

Θεωρήστε ότι:

$$\begin{aligned} \mathbf{V}_{\text{lang}} &= \{\text{en}, \text{gr}, \text{sp}\} \\ \mathbf{V}_{\text{detail}} &= \{\text{low}, \text{high}\} \\ \mathbf{V}_{\text{format}} &= \{\text{ps}, \text{pdf}, \text{html}\} \end{aligned}$$

Τα ακόλουθα είναι παραδείγματα πράξεων περιβάλλοντος στις οποίες συμμετέχουν πρότυπα περιβάλλοντος:

1.  $[\% \text{ lang in } \{\text{gr}, \text{sp}\}] * [\text{lang in } \{\text{en}, \text{gr}\}, \text{detail}=\text{low}] = [\text{lang}=\text{gr}]$
2.  $[\% \text{ lang}=\text{gr}, \text{detail}=\text{low}] + [\text{lang}=\text{en}, \text{detail}=\text{low}, \text{format}=\text{html}] =$   
 $[\text{lang in } \{\text{gr}, \text{en}\}, \text{detail}=\text{low}]$
3.  $[\text{format in } \{\text{pdf}, \text{ps}\}, \text{lang in } \{\text{en}, \text{sp}\}] - [\% \text{ lang}=\text{sp}] = [\text{lang}=\text{en}]$
4.  $[\% \text{ format in } \{\text{pdf}, \text{ps}\}] - [\text{detail}=\text{low}, \text{format}=\text{ps}] = [\text{format}=\text{pdf}]$

◆

<sup>65</sup> Παρατηρήστε ότι τα πρότυπα περιβάλλοντος με πολλούς όρους, όπως αυτό στο  $\text{cond}_4$ , δεν διαφέρουν από τα πρότυπα που έχουν μόνο έναν όρο. Όπως και στους προσδιοριστές περιβάλλοντος, οι πολλοί όροι χρησιμοποιούνται όταν ένα σύνολο κόσμων δεν είναι δυνατόν να εκφραστεί από έναν και μόνο όρο.

<sup>66</sup> Θα πρέπει να δοθεί προσοχή ώστε να μην απομακρυνθούν προσδιοριστές διάστασης της μορφής (διάσταση,  $\emptyset$ ) η παρουσία των οποίων κάνει τον όρο έναν κενό όρο, αλλά να αντικατασταθεί ολόκληρος ο όρος με τον κενό όρο  $\{-\}$ .

Όπως στις συνθήκες περιβάλλοντος, στις πράξεις περιβάλλοντος στο Παράδειγμα 5.4 απομακρύνουμε τον χαρακτήρα «%» από το πρότυπο περιβάλλοντος, και παράλληλα απομακρύνουμε από τον προσδιοριστή περιβάλλοντος όλους τους προσδιοριστές διάστασης που δεν αντιστοιχούν σε κάποια διάσταση του προτύπου περιβάλλοντος<sup>67</sup>. Στην συνέχεια, εκτελούμε την πράξη περιβάλλοντος σαν ανάμεσα από δυο προσδιοριστές περιβάλλοντος.

### 5.2.2.3 Περιοριστές περιβάλλοντος σαν πρότυπα περιβάλλοντος

Όταν ένα πρότυπο περιβάλλοντος χρησιμοποιείται σαν περιοριστής περιβάλλοντος, επιβάλλει την συνθήκη ότι το περιβάλλον της αντίστοιχης ακμής ή μονοπατιού πρέπει να είναι υπερσύνολο περιβάλλοντος του προτύπου περιβάλλοντος.

Για παράδειγμα, η έκφραση μονοπατιού περιβάλλοντος `o.[%c1]lb::[c2][%c3]` ταιριάζει με ένα μονοπάτι που, ξεκινώντας από το `o`, αποτελείται από μια ακμή οντότητας με ετικέτα `lb` και κληρονομούμενη κάλυψη `icvett >= [% c1]`, και από μια ακμή περιβάλλοντος με κληρονομούμενη κάλυψη `icvext >= [c2]` και ρητό περιβάλλον `ecext >= [% c3]`. Σαν άλλο ένα παράδειγμα θεωρήστε το `o::[%c1][c2].lb::[%c3]`, όπου `[%c1]` είναι ο μόνος περιοριστής κληρονομούμενης κάλυψης και συγκρίνεται με την κληρονομούμενη κάλυψη μονοπατιού ολόκληρου του μονοπατιού<sup>68</sup>.

Αναφερόμενοι στην βάση δεδομένων MOEM στην Εικόνα 5.1, η έκφραση μονοπατιού περιβάλλοντος

```
[% daytime=noon]recreation_guide::[-].restaurant::[-]
                                     .address::[-].floor::[-]
```

ψάχνει τις εκφάνσεις του `floor` όπου, κάποιες φορές, λειτουργούν εστιατόρια τα μεσημέρια, και αποτιμάται σε `{&24, &25}`. Ο κόμβος `&24` συμπεριλαμβάνεται επειδή το εστιατόριο λειτουργεί εκεί τα μεσημέρια του καλοκαιριού, ενώ ο κόμβος `&25` συμπεριλαμβάνεται λόγω των μεσημεριών του φθινοπώρου, χειμώνα, και άνοιξης. Παρατηρείστε ότι εάν είχε χρησιμοποιηθεί ο προσδιοριστής περιβάλλοντος `[daytime=noon]` αντί για το πρότυπο περιβάλλοντος, το αποτέλεσμα θα ήταν το κενό σύνολο.

### 5.2.2.4 Περιοριστές περιβάλλοντος σαν μεταβλητές περιβάλλοντος

Εκτός από τις μεταβλητές αντικειμένου περιβάλλοντος, που δηλώνονται με `x`, και από τις μεταβλητές πολυδιάστατου αντικειμένου, που δηλώνονται με `<x>`, οι εκφράσεις μονοπατιού περιβάλλοντος χρησιμοποιούν **μεταβλητές περιβάλλοντος**<sup>69</sup>, που δεσμεύονται σε προσδιοριστές περιβάλλοντος και δηλώνονται με `[x]`. Μια μεταβλητή περιβάλλοντος που

<sup>67</sup> Και πάλι, θα πρέπει να δοθεί προσοχή ώστε να μην απομακρυνθούν προσδιοριστές διάστασης της μορφής (διάσταση,  $\emptyset$ ) η παρουσία των οποίων κάνει τον όρο έναν κενό όρο, αλλά να αντικατασταθεί ολόκληρος ο όρος με τον κενό όρο.

<sup>68</sup> Όπως έχει αναφερθεί, σε αντίθεση με τους προσδιοριστές περιβάλλοντος, αν το `[%c1]` επαναληφθεί κατά μήκος του μονοπατιού στο οποίο αναφέρεται (δηλαδή μπροστά από το `lb` και το `[%c3]`) η έκφραση μονοπατιού περιβάλλοντος που θα προκύψει δεν θα είναι ισοδύναμη με την αρχική. Για να δούμε γιατί, ας θεωρήσουμε ότι το `[%c1]` είναι `[% lang=gr]` και ότι οι κληρονομούμενες καλύψεις δυο ακμών του μονοπατιού είναι `[lang=gr, detail=low]` και `[lang=gr, detail=high]`. Τότε η κληρονομούμενη κάλυψη μονοπατιού θα είναι ένα κενό περιβάλλον που δεν ταιριάζει με το δεδομένο πρότυπο περιβάλλοντος, παρά το γεγονός ότι και οι δυο κληρονομούμενες καλύψεις των ξεχωριστών ακμών ταιριάζουν με το πρότυπο περιβάλλοντος.

<sup>69</sup> Παρατηρήστε την διαφορά ανάμεσα στα **μεταβλητή περιβάλλοντος** και **μεταβλητή αντικειμένου περιβάλλοντος**.

εμφανίζεται στην θέση ενός περιοριστή κληρονομούμενης κάλυψης ή ενός περιοριστή ρητού περιβάλλοντος δεσμεύεται στην ανάλογη κληρονομούμενη κάλυψη μονοπατιού ή στο ανάλογο ρητό περιβάλλον, αντίστοιχα.

Για παράδειγμα, ας θεωρήσουμε την έκφραση μονοπατιού περιβάλλοντος  $\circ.[X]1b::[c][Y]$ , που περιέχει τις μεταβλητές περιβάλλοντος  $[X]$  και  $[Y]$ . Αυτή η έκφραση μονοπατιού περιβάλλοντος ταιριάζει με τα μονοπάτια που, αρχίζοντας από το αντικείμενο  $\circ$ , αποτελούνται από μια ακμή οντότητας με ετικέτα  $1b$  η οποία ακολουθείται από μια ακμή περιβάλλοντος με κληρονομούμενη κάλυψη που είναι υπερσύνολο περιβάλλοντος του  $[c]$ . Ας θεωρήσουμε ότι η έκφραση μονοπατιού περιβάλλοντος ταιριάζει με τα ακόλουθα μονοπάτια δεδομένων περιβάλλοντος: το  $cdp_1$  που οδηγεί στον κόμβο  $n_1$ , και το  $cdp_2$  που οδηγεί στον κόμβο  $n_2$ . Τότε, το αποτέλεσμα της έκφρασης μονοπατιού περιβάλλοντος θα είναι το σύνολο τριάδων  $\{(x_1, y_1, n_1), (x_2, y_2, n_2)\}$ , όπου  $x_i$  είναι η τιμή του  $[X]$  και  $y_i$  είναι η τιμή του  $[Y]$ . Συγκεκριμένα, το  $x_i$  είναι η κληρονομούμενη κάλυψη της ακμής με ετικέτα  $1b$  στο μονοπάτι δεδομένων περιβάλλοντος  $cdp_i$ , και το  $y_i$  είναι το ρητό περιβάλλον της ακμής περιβάλλοντος που ακολουθεί.

Ένα παράδειγμα ακόμη είναι το  $\circ.[X]a::[c_1].b::[c_2][c_3]$ , όπου η μεταβλητή περιβάλλοντος  $[X]$  δεσμεύεται στην κληρονομούμενη κάλυψη περιβάλλοντος των μονοπατιών που ξεκινούν από το  $\circ$  και περιλαμβάνουν με την σειρά: μια ακμή οντότητας με ετικέτα  $a$ , μια ακμή περιβάλλοντος με ρητό περιβάλλον που καλύπτει το  $[c_1]$ , και μια ακμή οντότητας με ετικέτα  $b$ . Συνεπώς, με έναν τρόπο, οι μεταβλητές περιβάλλοντος μοιάζουν με τις μεταβλητές ετικέτας της UnQL [BFS00], με την διαφορά ότι δεσμεύονται σε περιβάλλοντα αντί για ετικέτες. Οι μεταβλητές περιβάλλοντος χρησιμοποιούνται στην MQL για να ορίσουν σύνθετες συνθήκες πάνω σε περιβάλλοντα, και μπορούν επίσης να λάβουν μέρος στην κατασκευή νέων γράφων οι οποίοι επιστρέφονται σαν το αποτέλεσμα επερωτήσεων.

### 5.2.3 Υπονοούμενοι περιοριστές περιβάλλοντος

Προκειμένου να απλοποιήσουμε τις εκφράσεις μονοπατιού περιβάλλοντος και να τις καταστήσουμε πιο ευανάγνωστες, οι περιοριστές περιβάλλοντος μπορούν να παραλειφθούν σε συγκεκριμένες περιπτώσεις. Όταν ένας περιοριστής περιβάλλοντος λείπει, υπονοείται μια εξ ορισμού τιμή. Η *εξ ορισμού τιμή των περιοριστών περιβάλλοντος* είναι το κενό περιβάλλον  $[-]$ , το οποίο δεν επιβάλλει κανέναν περιορισμό στα περιβάλλοντα των ακμών ή των μονοπατιών.

Οι υπονοούμενοι περιοριστές περιβάλλοντος επεξηγούνται μέσω του ακόλουθου παραδείγματος.

#### Παράδειγμα 5.5

Σε κάθε ζεύγος, οι δυο εκφράσεις μονοπατιού περιβάλλοντος είναι ισοδύναμες: στην πρώτη υπονοούνται κάποιοι περιοριστές περιβάλλοντος, ενώ στην δεύτερη οι υπονοούμενοι περιοριστές περιβάλλοντος εμφανίζονται ρητά. Οι περιοριστές περιβάλλοντος  $[c_1]$  που εμφανίζονται πιο κάτω μπορούν να είναι προσδιοριστές περιβάλλοντος, πρότυπα περιβάλλοντος, και μεταβλητές περιβάλλοντος.

1.  $X.a::[c_1].b::[c_2][c_3].d$   
 $X.[-]a::[c_1].b::[c_2][c_3].d$
2.  $X.[c_1]a.b.d$   
 $X.[c_1]a::[-].b::[-].d$
3.  $\langle X \rangle.[c_1]a.b::[c_2][-].d$   
 $\langle X \rangle::[-][-].[c_1]a::[-].b::[c_2][-].d$



Στο Παράδειγμα 5.5 (1), το πρώτο στιγμιότυπο ενός περιοριστή κληρονομούμενης κάλυψης είναι  $[c_2]$  και εμφανίζεται στο τμήμα έκφανσης  $:: [c_2] [c_3]$ . Στην περίπτωση αυτή, υπονοείται ο περιοριστής κληρονομούμενης κάλυψης για το κομμάτι του μονοπατιού που δεν καλύπτεται ( $.a :: [c_1] .b$ ), και αυτός είναι ο εξ ορισμού περιοριστής περιβάλλοντος  $[-]$ . Το ζεύγος (2) δείχνει μια έκφραση μονοπατιού περιβάλλοντος όπου το τμήμα έκφανσης που θα έπρεπε να υπάρχει ανάμεσα από το  $. [c_1] a$  και το  $.b$  λείπει εντελώς. Το τμήμα έκφανσης που θα έπρεπε να υπάρχει ανάμεσα από το  $.b$  και το  $.d$  λείπει και αυτό. Στην περίπτωση αυτή, ο εξ ορισμού περιοριστής περιβάλλοντος  $[-]$  υπονοείται σαν ο περιοριστής ρητού περιβάλλοντος των τμημάτων έκφανσης που λείπουν. Στο ζεύγος (3), οι εκφράσεις μονοπατιού περιβάλλοντος αρχίζουν με  $\langle X \rangle$ , το οποίο είναι μια μεταβλητή πολυδιάστατου αντικειμένου και θα έπρεπε να ακολουθείται από κάποιο τμήμα έκφανσης. Το τμήμα αυτό έκφανσης λείπει, υπονοώντας το τμήμα έκφανσης  $:: [-] [-]$ , που ταιριάζει με οποιαδήποτε ακμή περιβάλλοντος. Συνεπώς, το ζεύγος (3) συνδυάζει τις περιπτώσεις που δείχνουν τα ζεύγη (1) και (2). Ένα άλλο ενδιαφέρον σημείο στο ζεύγος (3) είναι το τμήμα έκφανσης  $:: [c_2] [-]$ . Παρόλο που ο περιοριστής ρητού περιβάλλοντος είναι  $[-]$ , δεν μπορεί να παραληφθεί, επειδή το τμήμα έκφανσης εισάγει τον περιοριστή κληρονομούμενης κάλυψης  $[c_2]$  στην θέση αυτή. Αν το  $[-]$  έλειπε, τότε το  $[c_2]$  θα εκλαμβάνονταν σαν περιοριστής ρητού περιβάλλοντος.

Ενώ τα τμήματα έκφανσης μπορούν να υπονοηθούν, *τα τμήματα οντότητας δεν μπορούν να υπονοηθούν*. Συνεπώς, τα ακόλουθα είναι μη επιτρεπόμενες εκφράσεις μονοπατιού περιβάλλοντος:

- $X :: [c_1] [c_2]$
- $\langle X \rangle :: [c_1] [c_2] :: [c_3] [c_4]$

Παρόμοια με τις γλώσσες επερωτήσεων για ημιδομημένα δεδομένα, στην MQL μια έκφραση μονοπατιού περιβάλλοντος μπορεί να ακολουθείται από μια μεταβλητή αντικειμένου που δεσμεύεται στο σύνολο των κόμβων του αποτελέσματος. Το γεγονός ότι κάποια τμήματα έκφανσης είναι δυνατόν να λείπουν γεννά εκ νέου το ζήτημα που συζητήθηκε στην Ενότητα 5.2.1 δηλαδή του τύπου των κόμβων στους οποίους αποτιμάται μια έκφραση μονοπατιού περιβάλλοντος. Όπως αναφέρθηκε στην Ενότητα 5.2.1, αν το τελευταίο τμήμα είναι τμήμα έκφανσης τότε η έκφραση μονοπατιού περιβάλλοντος αποτιμάται σε ένα σύνολο κόμβων περιβάλλοντος, ενώ αν το τελευταίο τμήμα είναι τμήμα οντότητας τότε αποτιμάται σε ένα σύνολο πολυδιάστατων κόμβων. Έτσι, ένα τμήμα έκφανσης μπορεί να υπάρχει στο τέλος μιας έκφρασης μονοπατιού περιβάλλοντος μόνο και μόνο για να δηλώνει ότι το αποτέλεσμα θα πρέπει να είναι ένα σύνολο κόμβων περιβάλλοντος, πράγμα που δεν είναι ιδιαίτερα κομψό. Ευτυχώς, και πάλι μπορούμε να παραλείψουμε αυτό το τμήμα έκφανσης, και να χρησιμοποιήσουμε μια μεταβλητή αντικειμένου και καταναγκασμό (coercion) προκειμένου να επιτύχουμε το ίδιο αποτέλεσμα, όπως επιδεικνύει το παράδειγμα που ακολουθεί.

#### Παράδειγμα 5.6

Κάθε ζεύγος αποτελείται από δυο εκφράσεις μονοπατιού περιβάλλοντος: η πρώτη ακολουθείται από μια μεταβλητή αντικειμένου που επιβάλει έναν καταναγκασμό στον τύπο των κόμβων του αποτελέσματος, ενώ η δεύτερη είναι μια ισοδύναμη έκφραση μονοπατιού περιβάλλοντος όπου δεν λαμβάνει χώρα ο καταναγκασμός. Η έλλειψη της μεταβλητής αντικειμένου  $Y$  στην πρώτη έκφραση μονοπατιού περιβάλλοντος ενός ζεύγους αλλάζει την αποτίμησή της σε πολυδιάστατους κόμβους.

1.  $X. [c_1] a :: [c_2] .b \quad Y$   
 $X. [c_1] a :: [c_2] .b :: [-] \quad Y$
2.  $X.a.b.d \quad Y$

X. [-] a :: [-] . b :: [-] . d :: [-] Y



Το Παράδειγμα 5.6 δείχνει ότι μια έκφραση μονοπατιού περιβάλλοντος που ακολουθείται από μια μεταβλητή αντικειμένου περιβάλλοντος αποτιμάται πάντοτε σε αντικείμενα περιβάλλοντος, ανεξάρτητα από το εάν ή όχι τελειώνει σε τμήμα οντότητας ή έκφρασης. Πιο συγκεκριμένα, μια μεταβλητή αντικειμένου περιβάλλοντος μετά από μια έκφραση μονοπατιού περιβάλλοντος που καταλήγει σε τμήμα οντότητας, υπονοεί την προσθήκη του τμήματος έκφρασης :: [-] στο τέλος της έκφρασης μονοπατιού περιβάλλοντος. Συνεπώς, μια μεταβλητή αντικειμένου περιβάλλοντος μπορεί να προκαλέσει εξαναγκασμό στο επίπεδο της έκφρασης μονοπατιού περιβάλλοντος.

Δεδομένου ότι τα τμήματα οντότητας δεν μπορούν να υπονοηθούν, οι μεταβλητές πολυδιάστατου αντικειμένου δεν μπορούν να προκαλέσουν εξαναγκασμό όπως κάνουν οι μεταβλητές αντικειμένου περιβάλλοντος. Εκφράσεις όπου μια μεταβλητή πολυδιάστατου αντικειμένου ακολουθεί ένα τμήμα έκφρασης, όπως για παράδειγμα στο X. [c<sub>1</sub>] a :: [c<sub>2</sub>] [c<sub>3</sub>] <Y>, είναι μη επιτρεπόμενες εκφράσεις, μια και δεν είναι δυνατός ο εξαναγκασμός της έκφρασης μονοπατιού περιβάλλοντος. Πάντως, όπως έχουμε δει στο Παράδειγμα 5.5 (3), μια μεταβλητή πολυδιάστατου αντικειμένου μπορεί να υπονοήσει ένα τμήμα έκφρασης όταν βρίσκεται στην αρχή μιας έκφρασης μονοπατιού περιβάλλοντος.

Είναι ενδιαφέρον να σημειώσει κανείς ότι η έκφραση μονοπατιού περιβάλλοντος στο Παράδειγμα 5.6 (2) έχει την μορφή μιας συμβατικής έκφρασης μονοπατιού: όλοι οι περιοριστές περιβάλλοντος και τα τμήματα έκφρασης λείπουν. Αυτό σημαίνει ότι οι συμβατικές εκφράσεις μονοπατιού είναι μια ειδική περίπτωση των εκφράσεων μονοπατιού περιβάλλοντος, όπου οι συνθήκες περιβάλλοντος έχουν «απενεργοποιηθεί»<sup>70</sup>.

### Παράδειγμα 5.7

Οι πιο κάτω εκφράσεις μονοπατιού περιβάλλοντος ακολουθούνται από μια μεταβλητή αντικειμένου, και αναφέρονται στην βάση δεδομένων MOEM στην Εικόνα 5.1.

```
p1→ [lang=gr]recreation_guide.music_club.menu X
p2→ recreation_guide.music_club.menu::[lang=gr] X
p3→ recreation_guide.music_club.menu X
p4→ []recreation_guide.music_club.address <X>
p5→ []recreation_guide.music_club.review::-[-][detail=high] X
p6→ [% daytime=noon]recreation_guide.restaurant.address.floor X
p7→ recreation_guide.restaurant.[X]parking Y
p8→ recreation_guide.restaurant.parking::-[X] [-] Y
```



Στο Παράδειγμα 5.7, οι περιπτώσεις p1 έως p5 είναι ισοδύναμες με τις αντίστοιχες περιπτώσεις p1 έως p5 στο Παράδειγμα 5.2, με την διαφορά ότι οι περιοριστές περιβάλλοντος που είναι δυνατόν να υπονοηθούν λείπουν. Η εξήγηση που ακολουθεί το Παράδειγμα 5.2 εξακολουθεί να είναι σχετική και εδώ, αλλά μια σύγκριση δείχνει αμέσως ότι οι εκφράσεις μονοπατιού περιβάλλοντος στο Παράδειγμα 5.7 είναι πολύ πιο εύκολες στην κατανόηση και στον σχηματισμό τους. Η περίπτωση p6 έχει επεξηγηθεί στην Ενότητα 5.2.2.3, και η p7 αποτιμάται στο σύνολο {[daytime=noon], &31}, ([daytime=evening], &32)}, όπου η πρώτη τιμή κάθε πλειάδας αντιστοιχεί στην

<sup>70</sup> Στο προηγούμενο κεφάλαιο αναφέραμε ότι το OEM είναι μια ειδική περίπτωση Πολυδιάστατου Γράφου Δεδομένων. Αν το O είναι ένα συμβατικό OEM με M το πολυδιάστατο αντίστοιχό του, και το p<sub>O</sub> είναι μια συμβατική έκφραση μονοπατιού με p<sub>M</sub> το πολυδιάστατο αντίστοιχό της, τότε είναι εύκολο να δει κανείς ότι η αποτίμηση του p<sub>O</sub> στο O δίνει τα ίδια αποτελέσματα με την αποτίμηση του p<sub>M</sub> στο M.

μεταβλητή  $[X]$  και η δεύτερη τιμή στην μεταβλητή  $Y$ . Η περίπτωση  $p_8$  τυχαίνει να δίνει τα ίδια αποτελέσματα όπως η  $p_7$ , αλλά στην  $p_7$  το μονοπάτι όπου δεσμεύεται η  $[X]$  αποτελείται από μια ακμή οντότητας *parking* και μια ακόλουθη ακμή περιβάλλοντος, ενώ στην  $p_8$  το μονοπάτι αποτελείται μόνον από την ακμή περιβάλλοντος. Παρατηρήστε ότι, παρόλο που η καλή πρακτική απαιτεί την χρήση διαφορετικών ονομάτων για τις μεταβλητές, οι δυο μεταβλητές  $[X]$  και  $Y$  στις περιπτώσεις  $p_7$  και  $p_8$  είναι διαφορετικού τύπου και θα μπορούσαν να χρησιμοποιούν το ίδιο αναγνωριστικό,  $[X]$  και  $x$  αντίστοιχα, χωρίς να εισάγουν κάποια δισημία.

Καταλήγοντας, τα ακόλουθα απλά βήματα μπορούν να χρησιμοποιηθούν για να συμπληρώσουν τους απόντες περιοριστές περιβάλλοντος σε μια έκφραση μονοπατιού περιβάλλοντος.

1. *Περιοριστές ρητού περιβάλλοντος*: για κάθε τμήμα έκφρασης που υπονοείται (όπως προκύπτει από την σωστή αλληλοδιάδοχη των τμημάτων οντότητας και έκφρασης και από τον εξαναγκασμό σε μεταβλητές αντικειμένου περιβάλλοντος), θεωρούμε την εξ ορισμού τιμή  $[-]$ .
2. *Περιοριστές κληρονομούμενης κάλυψης*: αν το πρώτο τμήμα μιας έκφρασης μονοπατιού περιβάλλοντος, είτε τμήμα οντότητας είτε τμήμα έκφρασης, δεν περιέχει περιοριστή κληρονομούμενης κάλυψης τότε θεωρούμε την εξ ορισμού τιμή  $[-]$ .

Η ακόλουθη ενότητα στοχεύει στο να κάνει πιο κατανοητή την σημασία των εκφράσεων μονοπατιού περιβάλλοντος επιχειρώντας μια ποιοτική ανάλυση του νοήματός των.

## 5.2.4 Σημασιολογία των εκφράσεων μονοπατιού περιβάλλοντος

Στην ενότητα αυτή ερμηνεύουμε τις εκφράσεις μονοπατιού περιβάλλοντος με βάση τις συμβατικές εκφράσεις μονοπατιού και τα OEMs, και εξηγούμε συγκριτικά τις δυνατότητες των περιοριστών κληρονομούμενης κάλυψης και των περιοριστών ρητού περιβάλλοντος όταν αυτοί χρησιμοποιούνται σε MOEMs και σε Πολυδιάστατους Γράφους Δεδομένων.

Αρχίζουμε με την ακόλουθη έκφραση μονοπατιού περιβάλλοντος *cpe*:

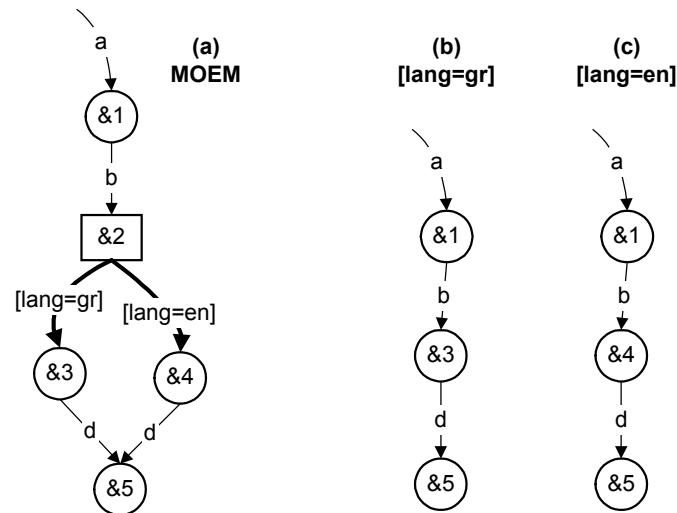
$[c]a.b.d$

Ας υποθέσουμε ότι το *cpe* αποτιμάται σε έναν κόμβο  $q$  πάνω σε κάποιο Πολυδιάστατο Γράφο Δεδομένων  $M$ . Θεωρούμε επίσης ότι το  $[c]$  είναι ένας προσδιοριστής περιβάλλοντος που αναπαριστά τους κόσμους  $\{w_1, w_2, \dots, w_n\}$ , και ότι η αναγωγή του  $M$  κάτω από κάθε έναν από τους κόσμους αυτούς δίνει τα OEMs  $O_1, O_2, \dots, O_n$  αντίστοιχα. Τότε, ο κόμβος  $q$  επιστρέφεται επίσης και από την (συμβατική) έκφραση μονοπατιού  $a.b.d$  αποτιμώμενη πάνω σε κάθε ένα από τα  $O_1, O_2, \dots, O_n$ . Να γιατί: λόγω των ιδιοτήτων της κληρονομούμενης κάλυψης μονοπατιού, αν ένας κόμβος  $q$  επιστρέφεται από το *cpe*, τότε οι ακμές  $a, b$ , και  $d$  και οι σχετικοί κόμβοι περιβάλλοντος του αντιστοίχου μονοπατιού δεδομένων περιβάλλοντος σχηματίζουν ένα μονοπάτι δεδομένων που επιβιώνει σε κάθε  $O_1, O_2, \dots, O_n$ . Έτσι μπορούμε να πούμε ότι μια έκφραση μονοπατιού περιβάλλοντος *cpe* της μορφής  $[c]a.b.d$  έχει μια αντίστοιχη συμβατική έκφραση μονοπατιού  $a.b.d$ , της οποίας η αποτίμηση σε κάθε  $O_1, O_2, \dots, O_n$  είναι υπερσύνολο της αποτίμησης του *cpe* στο  $M$ .

Για να δούμε ποιοι από τους κόμβους στους οποίους αποτιμάται το  $a.b.d$  ανήκουν και στην αποτίμηση του *cpe*, πρέπει να εξετάσουμε τα αντίστοιχα μονοπάτια δεδομένων σε κάθε  $O_1, O_2, \dots, O_n$ . Αν ένα μονοπάτι δεδομένων που οδηγεί στο  $q$  υπάρχει σε κάθε  $O_1, O_2, \dots, O_n$ , τότε υπάρχει και ένα αντίστοιχο μονοπάτι δεδομένων περιβάλλοντος στο  $M$  και το  $q$  επιστρέφεται και από το *cpe*. Ένα σημείο που απαιτεί προσοχή είναι ότι το ίδιο μονοπάτι

δεδομένων πρέπει να επιβιώνει στα  $O_1, O_2, \dots, O_n$ . Η Εικόνα 5.2 επιδεικνύει το σημείο αυτό μέσω ενός παραδείγματος.

Εικόνα 5.2: Τα μονοπάτια δεδομένων έχουν σημασία.



Η Εικόνα 5.2 (a) απεικονίζει τον Πολυδιάστατο Γράφο Δεδομένων  $M$ , που τυχαίνει να είναι ένα MOEM, ενώ η (b) και η (c) απεικονίζουν τα OEMs που υφίστανται κάτω από τους κόσμους  $\{[lang, gr]\}$  και  $\{[lang, en]\}$  αντίστοιχα. Η μόνη διαφορά ανάμεσα από το (b) και το (c) είναι το oid του μεσαίου κόμβου, το οποίο είναι &3 στο (b) και &4 στο (c). Η έκφραση μονοπατιού  $a.b.d$  επιστρέφει  $\{&5\}$  όταν αποτιμηθεί είτε πάνω στο (b) ή πάνω στο (c), αλλά η έκφραση μονοπατιού περιβάλλοντος  $[lang \text{ in } \{gr, en\}]a.b.d$  επιστρέφει ένα κενό σύνολο όταν αποτιμηθεί πάνω στο (a). Συνεπώς, το  $[c]a.b.d$  δεν αποτιμάται στους κόμβους που απλά είναι κοινί στα σύνολα αποτελεσμάτων του  $a.b.d$  έναντι των  $O_1, O_2, \dots, O_n$ : τα αντίστοιχα μονοπάτια δεδομένων (που συμπεριλαμβάνουν και τα oids των κόμβων που απαντώνται) θα πρέπει επίσης να είναι ταυτόσημα.

Ας θεωρήσουμε τώρα την έκφραση μονοπατιού περιβάλλοντος  $cpe_1$ :

$$[c_1]a.b.[c_2]d$$

Αυτή η έκφραση μονοπατιού περιβάλλοντος μπορεί να γίνει κατανοητή αν θεωρήσουμε τις επόμενες δυο εκφράσεις μονοπατιού περιβάλλοντος,  $cpe_2$  και  $cpe_3$  αντίστοιχα:

$$\begin{array}{l} [c_1]a.b \quad X \\ X.[c_2]d \quad Y \end{array}$$

Το  $cpe_2$  και το  $cpe_3$  είναι και τα δυο παρόμοια με το  $cpe$ , και η συζήτηση σχετικά με το  $cpe$  ισχύει και για τα  $cpe_2$  και  $cpe_3$ . Η μεταβλητή αντικειμένου  $x$  δεσμεύεται σε κόμβους που ταιριάζουν με το  $cpe_2$ . Κάθε κόμβος από αυτούς αντιμετωπίζεται σαν ρίζα από το  $cpe_3$ , που επιστρέφει τους κόμβους που είναι προσβάσιμοι μέσω της ακμής  $d$  κάτω από όλους τους



κόσμους του  $[c_2]$ . Προφανώς το  $cpe_1$  είναι ισοδύναμο με το ζευγάρι των  $cpe_2$  και  $cpe_3$ , με την έννοια ότι το  $cpe_1$  αποτιμάται στο ίδιο σύνολο κόμβων όπου δεσμεύεται το  $Y$ <sup>71</sup>.

Εκφράσεις μονοπατιού περιβάλλοντος όπως το  $cpe_1$  μπορούν να αντιμετωπιστούν σαν να ορίζουν ένα μονοπάτι όχι μόνο μεταξύ κόμβων, αλλά και μεταξύ συνόλων κόσμων, «πηδώντας» σε διαφορετικές αναγωγές OEM. Στην πραγματικότητα, οι εκφράσεις μονοπατιού περιβάλλοντος που συζητήσαμε στην ενότητα αυτή μπορούν να ιδωθούν σαν συνδέσεις (joins) μεταξύ κόμβων OEM και μεταξύ μονοπατιών δεδομένων των OEM που υφίστανται κάτω από διαφορετικούς κόσμους.

Ένα σχετικό ζήτημα είναι η θέση ενός περιοριστή περιβάλλοντος σε μια έκφραση μονοπατιού περιβάλλοντος. Θεωρήστε μια βάση δεδομένων υπαλλήλων με ερμηνευτικό περιβάλλον που αναπαριστά έγκυρο χρόνο, και τις εκφράσεις μονοπατιού περιβάλλοντος  $cpe_1$ ,  $cpe_2$ , και  $cpe_3$  αντίστοιχα:

```
[validtime=30]db.company.employee.salary X
db.company.employee.[validtime=30]salary X
db.company.employee.salary::[validtime=30] [-] X
```

Η έκφραση μονοπατιού περιβάλλοντος  $cpe_1$  αποτιμάται σε αντικείμενα περιβάλλοντος salary για τα οποία υπάρχει ένα μονοπάτι db.company.employee.salary στο συμβατικό OEM που υφίσταται κάτω από την χρονική στιγμή 30. Η έκφραση μονοπατιού περιβάλλοντος  $cpe_2$  είναι λιγότερο περιοριστική από το  $cpe_1$ , και επιτρέπει στο μονοπάτι που οδηγεί σε εκφάνσεις υπαλλήλων να υφίσταται κάτω από οποιονδήποτε κόσμο<sup>72</sup>. Ειδικότερα, το  $cpe_2$  απαιτεί να υπάρχει ένα μονοπάτι db.company.employee σε κάποιο OEM που υφίσταται κάτω από οποιαδήποτε χρονική στιγμή, και να υπάρχουν ακμές salary στο OEM το οποίο υφίσταται κάτω από την χρονική στιγμή 30. Η σύνδεση εκτελείται στις εκφάνσεις των υπαλλήλων από τις οποίες ξεκινούν ακμές salary. Ένας κόμβος  $q$  που επιστρέφεται από το  $cpe_2$  υπάρχει στο OEM που υφίσταται κάτω από την χρονική στιγμή 30, όπως επίσης και η ακμή salary που δείχνει σε αυτόν και ο κόμβος από τον οποίο ξεκινά η ακμή αυτή, αλλά όμως το μονοπάτι από τον τελευταίο αυτό κόμβο μέχρι την ρίζα του συγκεκριμένου OEM δεν είναι ορισμένο. Εκφράσεις μονοπατιού περιβάλλοντος όπως το  $cpe_2$  εστιάζουν σε συγκεκριμένα τμήματα ενός MOEM, και περιορίζουν μέσω κάποιου περιβάλλοντος μόνον μερικά από τα αντικείμενα (και τις σχέσεις τους) σε ένα μονοπάτι.

Σημειώστε ότι αν η κληρονομούμενη κάλυψη μονοπατιού κάθε στιγμιότυπου στην βάση του μονοπατιού db.company.employee::[-] είναι υπερσύνολο περιβάλλοντος του [validtime=30], τότε τα  $cpe_1$  και  $cpe_2$  είναι ισοδύναμα. Η εξήγηση για αυτό είναι ότι, στην περίπτωση αυτή, για κάθε κόμβο  $q$  που επιστρέφεται από το  $cpe_2$ , το μονοπάτι δεδομένων που αντιστοιχεί στο db.company.employee και υπάρχει σε κάποιο OEM θα υπάρχει επίσης και στο OEM που υφίσταται κάτω από την χρονική στιγμή 30, και το  $q$  θα επιστραφεί και από το  $cpe_1$ . Εξαιτίας αυτού μπορεί να χρησιμοποιούμε την μορφή του  $cpe_2$

<sup>71</sup> Σημειώστε ότι είναι επίσης δυνατό να γράψουμε το  $cpe_1$  σαν

```
[c1] a Z
Z. [c1] b X
X. [c2] d Y
```

ή να το σπάσουμε ακόμη περισσότερο χρησιμοποιώντας τμήματα έκφρασης και μεταβλητές πολυδιάστατου αντικειμένου, αλλά μόνον εάν τα  $[c_1]$  και  $[c_2]$  είναι προσδιοριστές περιβάλλοντος: όπως δείχνει η Πρόταση 5.1, ένας περιοριστής κληρονομούμενης κάλυψης που είναι προσδιοριστής περιβάλλοντος μπορεί να επαναληφθεί κατά μήκος του μονοπατιού που αφορά. Όμως, αν το  $[c_1]$  είναι πρότυπο περιβάλλοντος ή μεταβλητή περιβάλλοντος, το αντίστοιχο μονοπάτι  $a::[-].b::[-]$  θα πρέπει να περιοριστεί σαν ολότητα. Όμοια, αν το  $[c_2]$  είναι πρότυπο περιβάλλοντος ή μεταβλητή περιβάλλοντος, το αντίστοιχο μονοπάτι  $d::[-]$  πρέπει να περιοριστεί σαν ολότητα.

<sup>72</sup> Αν η βάση δεδομένων είναι ένα MOEM, το μονοπάτι θα υφίσταται κάτω από κάποιον κόσμο.

στις επερωτήσεις αντί για το  $cpe_1$ , συνήθως όταν η κληρονομούμενη κάλυψη των πρώτων ακμών στην βάση δεδομένων είναι το καθολικό περιβάλλον· τέτοιες ακμές μπορεί να αφεθούν χωρίς περιοριστή κληρονομούμενης κάλυψης.

Η παραπάνω συζήτηση ισχύει και για το  $cpe_3$ . Η διαφορά από το  $cpe_2$  είναι ότι το  $cpe_3$  απαιτεί να υπάρχει στο OEM που υφίσταται κάτω από την χρονική στιγμή 30 μόνο η έκφραση του μισθού, και ο κόμβος αυτός να είναι προσβάσιμος μέσω ενός μονοπατιού `db.company.employee.salary` σε οποιοδήποτε OEM που υφίσταται κάτω από κάποια χρονική στιγμή. Από την άλλη, το  $cpe_2$  απαιτεί ότι η έκφραση του μισθού και η ακμή `salary` και η αντίστοιχη έκφραση του υπαλλήλου υπάρχουν στο OEM που υφίσταται κάτω από την χρονική στιγμή 30.

Άλλο ένα ενδιαφέρον ζήτημα είναι η συγκριτική εκφραστικότητα μη κενών περιοριστών περιβάλλοντος σε MOEMs και σε Πολυδιάστατους Γράφους Δεδομένων. Οι περιοριστές κληρονομούμενης κάλυψης είναι ισχυροί με τα MOEMs, όπου οι κληρονομούμενες καλύψεις δεν είναι ποτέ κενά περιβάλλοντα. Από την άλλη, στους Πολυδιάστατους Γράφους Δεδομένων μπορεί να υπάρχουν υπογράφοι με κενή κληρονομούμενη κάλυψη (δεν υφίστανται κάτω από κανέναν κόσμο), πράγμα που στην πράξη μειώνει τις δυνατότητες πλοήγησης των εκφράσεων μονοπατιού περιβάλλοντος. Χρησιμοποιώντας αποκλειστικά μη κενούς περιοριστές κληρονομούμενης κάλυψης, μια έκφραση μονοπατιού περιβάλλοντος μπορεί να προσπελάσει μόνον το MOEM υπο-τιμήμα ενός Πολυδιάστατου Γράφου Δεδομένων.

Όμως, όπως αναφέρθηκε στο προηγούμενο κεφάλαιο, υπάρχουν ενδιαφέρουσες επερωτήσεις στους Πολυδιάστατους Γράφους Δεδομένων που εμπλέκουν εκφάνσεις οι οποίες δεν υφίστανται κάτω από κανέναν κόσμο (αλλά που διατηρούν την δυνατότητα να υπάρξουν κάτω από κάποιον κόσμο, μια και έχουν μη κενό ρητό περιβάλλον). Τέτοιες επερωτήσεις μπορούν να σχηματιστούν χρησιμοποιώντας περιοριστές ρητού περιβάλλοντος, οι οποίοι δίνουν ικανότητες πλοήγησης σε «χαμηλό επίπεδο» στις εκφράσεις μονοπατιού περιβάλλοντος.

Οι εκφράσεις μονοπατιού περιβάλλοντος που εξετάσαμε στην ενότητα αυτή αποτιμούνται μόνον σε αντικείμενα περιβάλλοντος. Ο λόγος είναι ότι δεν είναι δυνατόν να εκφράσουμε την σημασιολογία ενός πολυδιάστατου αντικειμένου χρησιμοποιώντας συμβατικά OEMs.

## 5.3 ΒΑΣΙΚΑ ΣΤΟΙΧΕΙΑ ΤΗΣ MQL

Η **πολυδιάστατη γλώσσα επερωτήσεων** (Multidimensional Query Language ή **MQL** για συντομία) βασίζεται σε γλώσσες επερωτήσεων για ημιδομημένα δεδομένα, συγκεκριμένα στην Lorel [AQM+97], την οποία επεκτείνει χρησιμοποιώντας εκφράσεις μονοπατιού περιβάλλοντος και πρόσθετους όρους που διαχειρίζονται το ερμηνευτικό περιβάλλον. Προκειμένου να εστιάσουμε σε ζητήματα σχετικά με το ερμηνευτικό περιβάλλον, χρησιμοποιήσαμε σαν σημείο αφετηρίας μια βασική «γλώσσα πυρήνα» που περιγράφεται στο [ABS00] και η οποία ενσωματώνει ουσιαστικά χαρακτηριστικά από την Lorel και την UnQL [BFS00], ενώ ταυτόχρονα είναι απλή και εύκολα επεκτάσιμη. Η MQL κρατάει τα κύρια χαρακτηριστικά της Lorel, δηλαδή ισχυρές εκφράσεις μονοπατιού και εξαναγκασμούς τύπων. Μια και ο κύριος στόχος της MQL είναι ο σχηματισμός επερωτήσεων οδηγούμενων από το περιβάλλον, δεν επικεντρωνόμαστε εδώ σε χαρακτηριστικά που έχουν επεξηγηθεί σε βάθος αλλού, όπως για παράδειγμα οι κανόνες εξαναγκασμού. Η σημασιολογία της MQL ανάγεται στην σημασιολογία της Lorel (και τελικά στο ODMG [ODMG])· στην Ενότητα 5.5 υλοποιούμε την MQL πάνω από το LORE, και ορίζουμε πως οι επερωτήσεις MQL μπορούν να μεταφραστούν σε ισοδύναμες επερωτήσεις Lorel.

Η MQL περιέχει τους ακόλουθους όρους:

```
select <οδηγός αποτελεσμάτων>
context <ορισμός μεταβλητών περιβάλλοντος>
from <εκφράσεις μονοπατιού περιβάλλοντος>
where <κατηγορήμα>
within <κατηγορήμα περιβάλλοντος>
```

Ο όρος `select` πρέπει να υπάρχει σε μια ερώτηση MQL, ενώ οι υπόλοιποι όροι είναι προαιρετικοί. Σε γενικές γραμμές, ο όρος `from` δίνει σαν αποτέλεσμα δεσμεύσεις μεταβλητών και πρέπει να αντιμετωπίζεται πρώτος· οι όροι `where` και `within` φιλτράρουν τις δεσμευμένες τιμές και αντιμετωπίζονται στην συνέχεια· ο όρος `context` ορίζει νέες μεταβλητές περιβάλλοντος που χρησιμοποιούνται στον όρο `select`· τέλος, αντιμετωπίζεται ο όρος `select`, που κατασκευάζει τον γράφο του αποτελέσματος. Ακολούθως, περιγράφουμε τους όρους έναν προς έναν.

### 5.3.1 Όρος “from”

Ο ρόλος του όρου `from` είναι να εισάγει μεταβλητές προσδεσμένες σε εκφράσεις μονοπατιού περιβάλλοντος. Η έξοδος ενός όρου `from` είναι ένα σύνολο πλειάδων, όπου κάθε πλειάδα περιέχει δεσμεύσεις μεταβλητών για κάποιο μονοπάτι δεδομένων περιβάλλοντος που έχει ταιριάζει.

```
from recreation_guide.restaurant X,
    X.[season=winter]address.floor Y,
    X.[season=summer, daytime=noon]address.floor Z
```

Αποτιμώμενος πάνω στην βάση MOEM που φαίνεται στην Εικόνα 5.1, αυτός ο όρος `from` δίνει ένα σύνολο μιας και μόνης πλειάδας  $\{(\&15, \&25, \&24)\}$ , όπου οι τιμές της πλειάδας αντιστοιχούν στις μεταβλητές  $(x, y, z)$ .

Ο ίδιος όρος `from` μπορεί να γράφει σαν:

```
from recreation_guide.restaurant {X}. [season=winter]address.floor Y,
    X.[season=summer, daytime=noon]address.floor Z
```

Όπως στην Lorel, οι αγκύλες `{ }` χρησιμοποιούνται για να εισάγουμε μεταβλητές αντικειμένου στην μέση μεγάλων μονοπατιών. Οι μεταβλητές στις αγκύλες μπορούν να είναι αντικειμένου περιβάλλοντος  $\{x\}$ , ή πολυδιάστατου αντικειμένου  $\{<x>\}$ . Οι μεταβλητές πολυδιάστατου αντικειμένου δεν μπορούν να εμφανιστούν αμέσως μετά από τμήμα έκφρασης, όπως εξηγήθηκε στην Ενότητα 5.2.3.

Ένα σημαντικό σημείο είναι ότι, αν ο περιοριστής κληρονομούμενης κάλυψης  $[c]$  είναι προσδιοριστής περιβάλλοντος, χρησιμοποιώντας την Πρόταση 5.1 ο όρος `from`

```
from [c]l1.l2 X
```

είναι ισοδύναμος με τον

```
from [c]l1 <V1>,
    <V1>:: [c] [-] V2,
    V2.[c]l2 <V3>,
    <V3>:: [c] [-] X
```

Όμως, αν το  $[c]$  είναι μεταβλητή περιβάλλοντος ή πρότυπο περιβάλλοντος, οι δυο όροι `from` πιο πάνω δεν είναι ισοδύναμοι, επειδή το  $[c]$  αναφέρεται στην κληρονομούμενη

κάλυψη μονοπατιού του 11::[-].12::[-], και όχι σε ξεχωριστές ακμές<sup>73</sup>. Η χρήση των αγκύλων είναι ένας βολικός τρόπος για να εισάγουμε μεταβλητές αντικειμένου στην μέση εκφράσεων μονοπατιού περιβάλλοντος και ταυτόχρονα επιτρέπει στις μεταβλητές αντικειμένου και στα πρότυπα περιβάλλοντος να αναφέρονται σε μεγάλα μονοπάτια σαν μια ολότητα.

Προσέξτε ότι στην ακόλουθη περίπτωση

```
from [c]11.12 X,
      X.13.14 Y
```

Ο περιοριστής περιβάλλοντος [c] που περιορίζει το πρώτο μονοπάτι δεν περιορίζει και το δεύτερο, όπου υπονοείται ο εξ ορισμού περιοριστής: X. [-]13.14 Y. Σε περίπτωση που χρησιμοποιούμε μεταβλητές αντικειμένου ενσωματωμένες στην έκφραση μονοπατιού περιβάλλοντος

```
from [c]11.12{X}. [-]13.14 Y
```

θα πρέπει να εισάγουμε τον περιοριστή κληρονομούμενης κάλυψης [-] πριν από το 13, διαφορετικά το [c] θα αναφέρεται σε ολόκληρο το μονοπάτι.

### 5.3.2 Όρος “select”

Ο όρος `select` κατασκευάζει τα αποτελέσματα. Το αποτέλεσμα μιας επερώτησης MQL είναι ένας Πολυδιάστατος Γράφος Δεδομένων στην μορφή μιας έκφρασης `mssd-expression`, η οποία κατασκευάζεται με βάση έναν οδηγό (template) που παρέχει ο όρος `select`. Όταν η επερώτηση

```
select restaurant: {name: <Z>, winter_floor: Y}
from recreation_guide.restaurant X,
      X.[season=winter]address.floor Y,
      X.name <Z>
```

αποτιμάται στην βάση δεδομένων στην Εικόνα 5.1, επιστρέφει την `mssd-expression`:

```
{restaurant: {name: <[]: &16 "...">, winter_floor: &25 "5th"}}
```

Η Εικόνα 5.3 (a) δίνει τα αποτελέσματα της επερώτησης γραφικά. Για απλότητα ο γράφος στην Εικόνα 5.1 δεν εμφανίζει τιμή για το φύλλο &16 που αναπαριστά το όνομα του εστιατορίου, και έτσι γράφουμε "... " αντί για την τιμή αυτή. Παρατηρήστε την χρήση της μεταβλητής πολυδιάστατου αντικειμένου <Z>, που δεσμεύεται σε έναν υπονοούμενο πολυδιάστατο κόμβο ο οποίος εισάγεται από τον μετασχηματισμό σε κανονική μορφή του MOEM στην Εικόνα 5.1. Αυτό εξηγεί και το καθολικό περιβάλλον [] στα αποτελέσματα.

Για να διευκολύνουμε τις συγκρίσεις, αναγράφουμε στα αποτελέσματα της επερώτησης τα `oids` των αντικειμένων όπως εμφανίζονται στην βάση δεδομένων στην Εικόνα 5.1. Ας έχουμε όμως κατά νου, ότι τα `oids` στα αποτελέσματα των επερωτήσεων δεν είναι απαραίτητως τα ίδια όπως τα `oids` των αντιστοιχών αντικειμένων στην βάση δεδομένων.

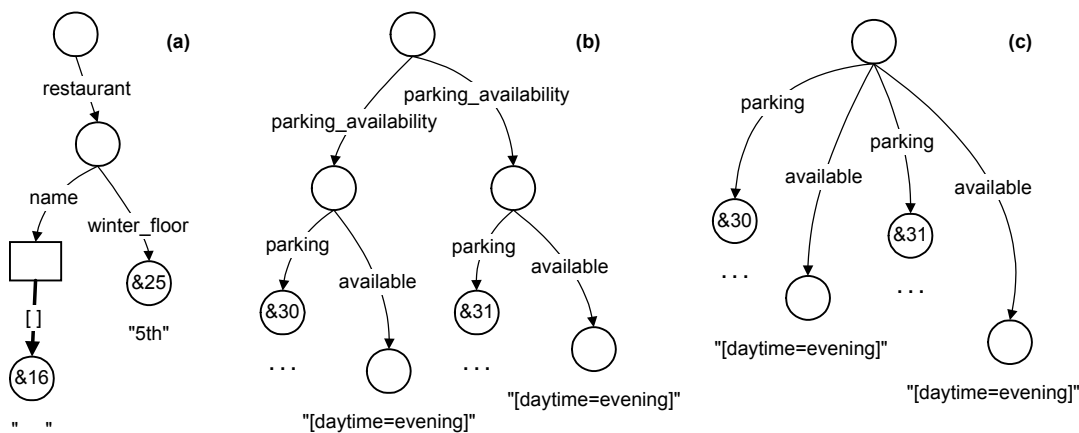
<sup>73</sup> Αν τα [V], [V1], [V2], [V3], [V4] είναι μεταβλητές περιβάλλοντος, τότε η έκφραση

```
from [V]a.b X
within κατηγορημα([V])
```

είναι ισοδύναμη με την έκφραση

```
from [V1]a <X1>, <X1>:::[V2] [-] X2,
      X2.[V3]b <X3>, <X3>:::[V4] [-] X
within κατηγορημα([V1]*[V2]*[V3]*[V4])
```

**Εικόνα 5.3: Αποτελέσματα ερωτήσεων MQL σαν Πολυδιάστατοι Γράφοι Δεδομένων.**



Οι μεταβλητές που εμφανίζονται στο `select` «προβάλλονται» από τις δεσμεύσεις των μεταβλητών που έρχονται από τον όρο `from`, δίνοντας ένα σάκο (bag) νέων πλειάδων, ή ένα σύνολο αν χρησιμοποιηθεί η λέξη-κλειδί `distinct`. Παραδείγματος χάριν, στην περίπτωση που δυο διαφορετικά αντικείμενα `restaurant` (διαφορετικές τιμές για το `x`) δείχνουν στα ίδια αντικείμενα `name` και `floor`, τότε θα έχουμε δυο πανομοιότυπες ( $\langle z \rangle, Y$ ) πλειάδες στα αποτελέσματα, εκτός εάν η λέξη-κλειδί `distinct` ακολουθεί την `select`. Η απομάκρυνση διπλοτύπων για τα αντικείμενα βασίζεται στο `oid`.

Όπως έχουμε δει, ο οδηγός `mssd-expression` στον όρο `select` μπορεί να περιέχει μεταβλητές πολυδιάστατου αντικειμένου και μεταβλητές αντικειμένου περιβάλλοντος στην θέση των τιμών αντικειμένων. Επιπλέον, μπορεί να περιέχει μεταβλητές περιβάλλοντος, όπως φαίνεται στην ακόλουθη ερώτηση:

```
select parking_availability: {parking: X, available: [Y]}
from recreation_guide.music_club.parking:: [Y] X
```

Η μεταβλητή περιβάλλοντος `[Y]` δεσμεύεται στα ρητά περιβάλλοντα που αντιστοιχούν στις εκφάνσεις του `parking`, και το αποτέλεσμα της ερώτησης είναι η `mssd-expression`

```
{parking_availability:
  {parking: &30 {...}, available: "[daytime=evening]"},
 parking_availability:
  {parking: &31 {...}, available: "[daytime=noon]"}}
```

Τα αποτελέσματα<sup>74</sup> δείχνονται επίσης και σαν γράφος στην Εικόνα 5.3 (b). Προσέξτε ότι τιμές της μεταβλητής περιβάλλοντος `[Y]` γίνονται τιμές τύπου στοιχειοσειράς σε αντικείμενα του αποτελέσματος<sup>75</sup>. Ακολουθεί μια παρόμοια ερώτηση:

<sup>74</sup> Για συντομία, ο γράφος στην Εικόνα 5.1 δεν είναι ολοκληρωμένος. Εδώ υποθέτουμε ότι οι εκφάνσεις του `parking` είναι σύνθετα αντικείμενα. Αν αντίθετα θεωρήσουμε ότι είναι ατομικά, το `{...}` θα πρέπει να αντικατασταθεί από το `"..."`.

<sup>75</sup> Μια παρόμοια περίπτωση στην Lorel είναι όταν χρησιμοποιούμε μεταβλητές ετικέτας ή μεταβλητές μονοπατιού για να μετατρέψουμε μεταπληροφορία (ονόματα ετικετών) σε δεδομένα (τιμές τύπου στοιχειοσειράς στα αντικείμενα του αποτελέσματος).

```
select parking: X, available: [Y]
from recreation_guide.music_club.parking:: [Y] X
```

Το αποτέλεσμα της επερώτησης είναι:

```
{parking: &30 {...},
 available: "[daytime=evening]",
 parking: &31 {...},
 available: "[daytime=noon]" }
```

Η Εικόνα 5.3 (c) απεικονίζει τα αποτελέσματα της επερώτησης σαν γράφο. Παρατηρήστε ότι εάν το `parking_availability` παραληφθεί, τα αντικείμενα `parking` και `available` δεν είναι πια ομαδοποιημένα σε ζευγάρια. Ο λόγος είναι ότι οι μεταβλητές `X` και `[Y]` δεσμεύονται σαν ζευγάρια, αλλά αν δεν χρησιμοποιηθεί κάποιο αντικείμενο πατέρας για να αντικατοπτρίσει το γεγονός αυτό στα αποτελέσματα, η πληροφορία σχετικά με το ποια τιμή του `X` αντιστοιχεί σε ποια τιμή του `[Y]` θα χαθεί.

Οι μεταβλητές περιβάλλοντος μπορούν επίσης να χρησιμοποιηθούν στην θέση των ετικετών των ακμών περιβάλλοντος στις εκφράσεις `mssd-expression` του οδηγού στο `select`, ορίζοντας το ρητό περιβάλλον των ακμών περιβάλλοντος στα αποτελέσματα. Η Ενότητα 5.4.3 συζητά κάποια προχωρημένα χαρακτηριστικά του `select`, συμπεριλαμβανομένων της παράλειψης ετικετών και της χρήσης εκφράσεων μονοπατιού περιβάλλοντος, και εξετάζει πιο σύνθετες περιπτώσεις κατασκευής αποτελεσμάτων της MQL.

### 5.3.3 Όρος “where”

Ο όρος `where` δεν διαφέρει από τον αντίστοιχο της `Lorel`, και φιλτράρει τα αποτελέσματα ακριβώς με τον ίδιο τρόπο:

```
select restaurant: {name: P, winter_floor: Y}
from recreation_guide.restaurant X,
     X.[season=winter]address.floor Y,
     X.[season=summer, daytime=noon]address.floor Z
     X.name P
where Z="terrace"
```

Η παραπάνω επερώτηση επιστρέφει το όνομα ενός εστιατορίου και τον όροφο όπου λειτουργεί τον χειμώνα, για τα εστιατόρια που λειτουργούν στην ταράτσα τα μεσημέρια του καλοκαιριού. Είναι σημαντικό να καταλάβει κανείς ότι, όταν ένα εστιατόριο δείχνει σε ένα πολυδιάστατο αντικείμενο `name` που περιλαμβάνει δυο εκφάνσεις, το εστιατόριο θα συμμετέχει στο αποτέλεσμα της επερώτησης σαν δυο διαφορετικά αντικείμενα `restaurant`, ένα για κάθε έκφανση του `name`. Αυτό συμβαίνει επειδή η MQL δεν ομαδοποιεί αυτόματα αντικείμενα, αλλά δημιουργεί αντικείμενα στα αποτελέσματα σύμφωνα με τις πλειάδες των δεσμεύσεων των μεταβλητών. Η ομαδοποίηση των αποτελεσμάτων συζητιέται στην Ενότητα 5.4.2.

Άλλο ένα ενδιαφέρον σημείο είναι η χρήση του εξαναγκασμού στην σύγκριση `Z="terrace"`, ακριβώς όπως στην `Lorel`. Η μεταβλητή `Z` δεσμεύεται σε ένα αντικείμενο, αλλά συγκρίνεται με μια στοιχειοσειρά. Φυσιολογικά η σύγκριση αυτή θα έδινε πάντοτε `false` λόγω της ασυμφωνίας των τύπων, αλλά λόγω του εξαναγκασμού επιστρέφει `true` αν η τιμή του αντικειμένου που είναι τρεχόντως δεσμευμένο στο `Z` ισούται με την

στοιχειοσειρά<sup>76</sup>. Επιπλέον, μια διεύθυνση που δεν περιέχει κάποια έκφραση του ορόφου που να υφίσταται κάτω από μεσημέρια του καλοκαιριού δεν θα προκαλέσει σφάλμα τύπου (type error), αλλά απλά θα αποτύχει στην σύγκριση<sup>77</sup>. Ο εξαναγκασμός διευκολύνει τον σχηματισμό επερωτήσεων σε δεδομένα με μεταβλητή δομή (ακανόνιστοι τύποι και πεδία που λείπουν). Ο εξαναγκασμός χρησιμοποιείται στην MQL με τον ίδιο τρόπο όπως στην Lorel: (a) μετριάζει την ανάγκη να γνωρίζει κάποιος αν μια τιμή είναι αριθμητική ή στοιχειοσειρά και επιτρέπει την σύγκριση μιας τιμής με μια αναφορά αντικειμένου, (b) επιτρέπει να συγκρίνουμε ένα αντικείμενο με ένα σύνολο αντικειμένων, υπονοώντας έναν υπαρξιακό τελεστή. Η ακόλουθη επερώτηση δείχνει ένα παράδειγμα αυτής της περίπτωσης:

```
select name: Y
from recreation_guide.restaurant{X}.name Y
where X.review.score::[-] >= 8
```

Η έκφραση `X.review.score::[-]` τελειώνει με ένα τμήμα έκφρασης και αποτιμάται σε ένα σύνολο αντικειμένων περιβάλλοντος, το οποίο συγκρίνεται με την αριθμητική τιμή 8. Το τμήμα έκφρασης `::[-]` δεν είναι απαραίτητο, επειδή η MQL προκαλεί την αποτίμηση σε αντικείμενα περιβάλλοντος όποτε εκφράσεις μονοπατιού περιβάλλοντος συγκρίνονται με στοιχειοσειρές ή με αριθμητικές τιμές. Η επερώτηση ερμηνεύεται σαν:

```
select name: Y
from recreation_guide.restaurant{X}.name Y
where exists Z in X.review.score (Z >= 8)
```

Ένα τμήμα έκφρασης `::[-]` στο τέλος της έκφρασης μονοπατιού περιβάλλοντος στο `where` είναι και πάλι προαιρετικό, επειδή η μεταβλητή αντικειμένου περιβάλλοντος `Z` προκαλεί αποτίμηση σε αντικείμενα περιβάλλοντος. Η παραπάνω επερώτηση δεν χρησιμοποιεί περιβάλλοντα: οι υπονοούμενοι περιοριστές περιβάλλοντος έχουν την εξ ορισμού τιμή `[-]`, και δεν επιβάλλουν συνθήκες στα περιβάλλοντα. Σημειώστε ότι μια επερώτηση Lorel με το ίδιο νόημα θα ήταν συντακτικά πολύ παρόμοια με την συγκεκριμένη επερώτηση MQL: *όταν τα χαρακτηριστικά περιβάλλοντος της MQL δεν χρησιμοποιούνται, οι επερωτήσεις σε MQL σχηματίζονται το ίδιο εύκολα όπως οι επερωτήσεις σε Lorel.*

Στην πραγματικότητα, οι εκφράσεις μονοπατιού περιβάλλοντος στο `where` υπονοούν (τμήματα από) έναν όρο `from`. Σαν παράδειγμα θεωρήστε το:

```
select club_with_menu: {X.name, <Y>}
where exists recreation_guide.-{X}.menu{<Y>}
```

Η επερώτηση χρησιμοποιεί μια έκφραση μονοπατιού περιβάλλοντος στο `select`, και κάποιες ετικέτες έχουν παραλειφθεί από τον οδηγό `mssd-expression`. Και τα δυο αυτά ζητήματα συζητούνται στην Ενότητα 5.4.3.2. Ο μπαλαντέρ - ταιριάζει με οποιαδήποτε ετικέτα ακμής οντότητας, και συζητείται στην Ενότητα 5.4.1.2. Μεταβλητές που εισάγονται σαν μέρος εκφράσεων μονοπατιού περιβάλλοντος στο `where` μπορούν να χρησιμοποιηθούν και σε άλλους όρους, όπως για παράδειγμα το `X` και το `<Y>` στο `select`. Η έκφραση μονοπατιού περιβάλλοντος στο `where` υπονοεί έναν όρο `from`:

<sup>76</sup> Όταν χρησιμοποιείται το `=` μεταξύ δυο αντικειμένων, ο εξαναγκασμός δεν λαμβάνει χώρα και η ισότητα αποφασίζεται συγκρίνοντας `oids`. Όπως στην Lorel, ο τελεστής `==` μπορεί να χρησιμοποιηθεί για να προκαλέσει εξαναγκασμό και σύγκριση τιμών αντί για σύγκριση `oids`.

<sup>77</sup> Όπως στην Lorel, η ειδική τιμή `nil` υποδεικνύει ότι δεν υπάρχει δέσμευση μιας μεταβλητής σε κάποιο αντικείμενο, μια συνηθισμένη περίπτωση στα ημιδομημένα δεδομένα όπου η δομή είναι μεταβαλλόμενη. Η παρουσία του `nil` σε μια συνθήκη κάνει πάντα την συνθήκη ψευδή. Επιπρόσθετα, μια τιμή `nil` δεν προκαλεί την δημιουργία ενός νέου αντικειμένου στον γράφο του αποτελέσματος.

```
select club_with_menu: {X.name, <Y>}
from recreation_guide.-{X}.menu{<Y>}
where exists <Y>
```

Όταν αποτιμάται στην βάση δεδομένων στην Εικόνα 5.1, ο όρος `form` δεσμεύει τα ( $x$ ,  $\langle Y \rangle$ ) στις πλειάδες  $\{(\&2, \&37), (\&15, nil)\}$ . Για την δέσμευση του  $x$  στο  $\&15$  που αντιστοιχεί στο `restaurant` στην Εικόνα 5.1, η μεταβλητή  $\langle Y \rangle$  δεσμεύεται στην ειδική τιμή `nil`, επειδή το αντικείμενο `restaurant` δεν περιέχει `menu`. Συνεπώς, η δεύτερη πλειάδα αποτυγχάνει στην συνθήκη του `where`, και το `select` επιστρέφει ένα αντικείμενο `club_with_menu`. Αν η συνθήκη στο `where` δεν υπήρχε, θα συμπεριλαμβάνονταν στα αποτελέσματα και ένα δεύτερο αντικείμενο `club_with_menu` που θα περιείχε μόνο το `name` του αντικειμένου  $\&15$ .

### 5.3.4 Όρος “within”

Ο όρος `within` παίζει τον ρόλο του όρου `where` αλλά για ερμηνευτικά περιβάλλοντα: εκφράζει συνθήκες περιβάλλοντος πάνω σε μεταβλητές περιβάλλοντος που έχουν εισαχθεί σε άλλους όρους. Το κατηγορήμα περιβάλλοντος θα πρέπει να το ικανοποιήσει μια πλειάδα των δεσμεύσεων των μεταβλητών, διαφορετικά η πλειάδα απορρίπτεται.

```
select row: {context: [X], menu: Y}
from recreation_guide.music_club.menu:: [X] Y
within [X] * [lang in {gr,en}] != [-]
```

Η επερώτηση χρησιμοποιεί την μεταβλητή περιβάλλοντος  $[X]$  για να μετασχηματίσει μεταπληροφορία περιβάλλοντος στην βάση δεδομένων σε κανονικά δεδομένα στον γράφο του αποτελέσματος, αλλά και για να φιλτράρει κόμβους στον όρο `within`. Επιστρέφει τα μενού των `music clubs` στα Ελληνικά ή στα Αγγλικά, μαζί με το αντίστοιχο ρητό τους περιβάλλον.

Όπως και στον όρο `where`, το κατηγορήμα περιβάλλοντος στον όρο `within` μπορεί να περιέχει μεταβλητές περιβάλλοντος, προσδιοριστές περιβάλλοντος, και πρότυπα περιβάλλοντος, μαζί με τα σύμβολα πράξεων περιβάλλοντος `+`, `*`, `-`, `<`, `>`, `<=`, `>=`, `=`, `!=`, παρενθέσεις, και τις λέξεις-κλειδιά `and`, `or`, και `not`. Μια πλήρης προδιαγραφή της σύνταξης του όρου `within` δίνεται στο Παράρτημα Α.

### 5.3.5 Όρος “context”

Ο όρος `context` ορίζει νέες μεταβλητές περιβάλλοντος που μπορούν στην συνέχεια να χρησιμοποιηθούν στον όρο `select` για την κατασκευή του γράφου του αποτελέσματος. Οι τιμές των νεοεισαχθέντων μεταβλητών περιβάλλοντος υπολογίζονται ξεχωριστά για κάθε πλειάδα δεσμεύσεων μεταβλητών που έχει περάσει από τους όρους `select` και `within`, και τα αποτελέσματα εισάγονται στην αντίστοιχη πλειάδα και την επαυξάνουν.

```
select menu_worlds: [Z]
context [Z] as union([Y])
from recreation_guide.music_club{X}.menu:: [Y] [-]
where oid(X) = "&2"
```

Η παραπάνω επερώτηση χρησιμοποιεί την **συναθροιστική συνάρτηση περιβάλλοντος** `union([c])`, για να υπολογίσει τους κόσμους κάτω από τους οποίους υπάρχει μια έκφραση του `menu` για το `music club` με `oid &2`. Η συνάρτηση `oid(OBJVAR)` επιστρέφει το `oid` ενός



αντικειμένου σαν στοιχειοσειρά. Όταν αποτιμηθεί στο MOEM στην Εικόνα 5.1, το αποτέλεσμα είναι η *mssd-expression*:

```
{menu_worlds: "[lang in {gr,en,fr}]"} }
```

Σαν άλλο ένα παράδειγμα, θεωρήστε την επερώτηση

```
select comments: {language: [Z], contents: X}
context [Z] as [% lang in ALL] * [Y]
from recreation_guide. #.comments:: [Y] [-] X
```

που επιστρέφει για κάθε έκφραση του *comments*, τα ίδια τα σχόλια και την γλώσσα τους. Το σύμβολο *.#* είναι ένας μπαλαντέρ που ταιριάζει με κάθε μονοπάτι, και επεξηγείτε στην Ενότητα 5.4.1.2. Η λέξη-κλειδί *ALL* είναι συντομογραφία για ολόκληρο το πεδίο ορισμού μιας διάστασης. Η επερώτηση δεσμεύει την μεταβλητή περιβάλλοντος *[Y]* στις κληρονομούμενες καλύψεις των ακμών περιβάλλοντος που οδηγούν σε εκφάνσεις του *comments*. Η κληρονομούμενη κάλυψη μπορεί να περιέχει διαστάσεις διαφορετικές από το *lang*, όπως για παράδειγμα την διάσταση *detail*. Οι διαστάσεις αυτές απομακρύνονται από τον όρο *context*, χρησιμοποιώντας τομή περιβάλλοντος με το πρότυπο περιβάλλοντος *[% lang in ALL]*. Με τον τρόπο αυτό, το *[Z]* περιέχει μόνο τις γλώσσες κάτω από τις οποίες υφίσταται μια έκφραση, παραλείποντας άλλες διαστάσεις. Τα πρότυπα περιβάλλοντος χρησιμοποιούνται στην τομή περιβάλλοντος, στην ένωση περιβάλλοντος, και στην διαφορά περιβάλλοντος για να φιλτράρουν προσδιοριστές περιβάλλοντος, όπως περιγράφεται στην Ενότητα 5.2.2.2.

Ο όρος *context* μπορεί να ορίζει οποιονδήποτε αριθμό μεταβλητών περιβάλλοντος οι οποίες χωρίζονται με κόμμα. Μια μεταβλητή περιβάλλοντος που ορίζεται στον όρο *context* δεν μπορεί να εμφανιστεί στο δεξί μέρος της λέξης-κλειδιού *as* οποιουδήποτε ορισμού μεταβλητής που βρίσκεται στον ίδιο όρο. Στον ορισμό μιας μεταβλητής, το δεξί μέρος μπορεί να περιέχει μεταβλητές περιβάλλοντος, προσδιοριστές περιβάλλοντος, πρότυπα περιβάλλοντος, και σύμβολα πράξεων περιβάλλοντος *+*, *\**, *-*, παρενθέσεις, και τις συναθροιστικές συναρτήσεις περιβάλλοντος *union([c])*, και *intersect([c])*<sup>78</sup>.

Επιπλέον, μπορεί να χρησιμοποιηθεί η συνάρτηση *extension([c])*, η οποία μετατρέπει ένα περιβάλλον σε «κόσμους». Στην πραγματικότητα, οι προσδιοριστές περιβάλλοντος που επιστρέφονται από την *extension([c])* μπορεί να μην αναπαριστούν κόσμους με την αυστηρή έννοια, επειδή η συνάρτηση θεωρεί ότι το σύνολο των διαστάσεων *D* περιέχει μόνο τις διαστάσεις που εμφανίζονται στο τρέχον της όρισμα περιβάλλοντος.

Ειδικότερα, η συνάρτηση *extension([c])* παράγει ένα νέο σύνολο προσδιοριστών περιβάλλοντος για κάθε περιβάλλον δεσμευμένο στην μεταβλητή περιβάλλοντος *[c]* ως ακολούθως: (a) αν η τιμή της παραμέτρου είναι *nil* ή αν περιέχει το καθολικό περιβάλλον  $\emptyset$ , επιστρέφεται ένα σύνολο που περιέχει μόνο το *nil*, (b) αν η τιμή της παραμέτρου είναι ένα κενό περιβάλλον, επιστρέφεται ένα κενό σύνολο, και (c) σε οποιαδήποτε άλλη περίπτωση, επιστρέφεται ένα σύνολο προσδιοριστών περιβάλλοντος που αντιστοιχεί στην *ανάπτυξη περιβάλλοντος* της τιμής της παραμέτρου. Το ακόλουθο παράδειγμα δείχνει την χρήση της *extension* με ένα περιβάλλον σαν όρισμα:

```
extension([lang in {gr,en}, detail=low]) = { [lang=gr, detail=low],
                                             [lang=en, detail=low] }
```

<sup>78</sup> Η παράμετρος *[c]* μπορεί να είναι μια δεσμευμένη μεταβλητή περιβάλλοντος ή ένας προσδιοριστής περιβάλλοντος. Στην περίπτωση που το *[c]* είναι προσδιοριστής περιβάλλοντος, το *union([c])* και το *intersect([c])* επιστρέφουν *[c]*.

Ας θεωρήσουμε ότι το  $[c]$  είναι μια μεταβλητή περιβάλλοντος, και ας είναι  $F_i$  το σύνολο του αποτελέσματος της  $\text{extension}([c])$  για την τιμή  $v_i$  του  $[c]$ ,  $1 \leq i \leq n$ . Ας είναι  $S_i$  το υποσύνολο πλειάδων των δεσμεύσεων των μεταβλητών πριν κληθεί η  $\text{extension}([c])$ , για το οποίο το  $[c]$  έχει την τιμή  $v_i$ . Τότε, το νέο σύνολο πλειάδων των δεσμεύσεων των μεταβλητών μετά την κλήση της  $\text{extension}([c])$  είναι  $\sum_{i=1}^n (F_i \times S_i)$ . Το σύμβολο  $\times$  αντιπροσωπεύει το Καρτεσιανό γινόμενο, και το  $\sum_{i=1}^n$  την ένωση συνόλου για όλα τα Καρτεσιανά γινόμενα από  $i = 1$  έως  $i = n$ .

Στην Ενότητα 5.4.3.3 επιδεικνύουμε την χρήση της  $\text{extension}([c])$  μέσω ενός παραδείγματος επερωτήσεως. Οι πλήρεις προδιαγραφές της σύνταξης του όρου `context` δίνονται στο Παράρτημα Α.

### 5.3.6 Διαστάσεις και πεδία ορισμού των στην MQL

Όπως δείξαμε σε προηγούμενα κεφαλαία, δεν είναι απαραίτητο να γνωρίζουμε το πραγματικό σύνολο διαστάσεων  $\mathbf{D}$  προκειμένου να επιτελέσουμε τις πράξεις περιβάλλοντος που χρησιμοποιεί η MQL. Συνεπώς, η MQL δεν χρειάζεται να γνωρίζει το σύνολο διαστάσεων  $\mathbf{D}$  για μια βάση δεδομένων, αλλά εκτελεί τις πράξεις περιβάλλοντος με βάση τους τρέχοντες προσδιοριστές περιβάλλοντος.

Όμως, έχουμε δει ότι για συγκεκριμένες πράξεις περιβάλλοντος είναι απαραίτητο να γνωρίζουμε τα πεδία ορισμού των εμπλεκόμενων διαστάσεων. Θεωρούμε λοιπόν ότι, δεδομένης μιας διάστασης, η MQL έχει πλήρη πληροφορία (ή πρόσβαση σε πληροφορία) που αφορά στο πεδίο ορισμού της διάστασης αυτής.

## 5.4 ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ

Κάποια χαρακτηριστικά των γλωσσών επερωτήσεων για ημιδομημένα δεδομένα είναι ορθογώνια με τις οδηγούμενες από το περιβάλλον επερωτήσεις και μπορούν εύκολα να ενσωματωθούν στην MQL, ενώ άλλα απαιτούν μεγαλύτερη προσοχή. Μια και η πρόθεσή μας είναι να δείξουμε πως το ερμηνευτικό περιβάλλον μπορεί να ενσωματωθεί σε μια γλώσσα επερωτήσεων, εστιάζουμε στην δεύτερη κατηγορία. Στην ενότητα αυτή συνεχίζουμε την παρουσίαση της MQL με μια σειρά σημαντικών ζητημάτων: γενικευμένες εκφράσεις μονοπατιού περιβάλλοντος, εμφωλιασμένες επερωτήσεις MQL, και κατασκευή των αποτελεσμάτων.

### 5.4.1 Γενικευμένες εκφράσεις μονοπατιού περιβάλλοντος

Οι γενικευμένες εκφράσεις μονοπατιού περιβάλλοντος επεκτείνουν τις απλές εκφράσεις μονοπατιού περιβάλλοντος με ένα πιο ισχυρό συντακτικό που δίνει στην MQL την ευελιξία να θέτει επερωτήσεις χωρίς ακριβή γνώση της δομής της βάσης δεδομένων. Οι γενικευμένες εκφράσεις μονοπατιού περιβάλλοντος χαρακτηρίζονται από **ομαλές εκφράσεις** (regular expressions) and **μπαλαντέρ** (wildcards).

Στην ενότητα αυτή εισάγουμε επίσης τις **μεταβλητές ετικέτας** και τις **μεταβλητές μονοπατιού** της MQL, οι οποίες παρά το ότι δεν αποτελούν μέρος των γενικευμένων εκφράσεων μονοπατιού περιβάλλοντος έχουν στενή σχέση με αυτές.

### 5.4.1.1 Ομαλές εκφράσεις

Όπως η «γλώσσα πυρήνα» που παρουσιάζεται στο [ABS00], η MQL χρησιμοποιεί **ομαλές εκφράσεις** (regular expressions) [Fri97] σε δυο επίπεδα: στο επίπεδο των ετικετών των ακμών οντότητας, και στο επίπεδο των τμημάτων οντότητας και των τμημάτων έκφανσης<sup>79</sup>. Το παράδειγμα που ακολουθεί ψάχνει για εκφάνσεις του review αντικειμένων το όνομα των οποίων περιέχει το «music» ή το «Music»:

```
X. [% detail=high] ".*[Mm]usic.*".review::[-]
```

Η χρήση εισαγωγικών σημαίνει ότι οι ομαλές εκφράσεις εφαρμόζονται στο επίπεδο των ετικετών μάλλον παρά στο επίπεδο των τμημάτων οντότητας και των τμημάτων έκφανσης. Έτσι, το `.*[Mm]usic.*` ταιριάζει με ετικέτες όπως: `music_club`, `MusicRestaurant`, `live-music-hall`, κτλ. (το `.` σημαίνει οποιαδήποτε ακολουθία από μηδέν ή περισσότερους χαρακτήρες). Η χρήση του προτύπου περιβάλλοντος `[% detail=high]` υποδηλώνει ότι η αντίστοιχη κάλυψη μονοπατιού περιβάλλοντος πρέπει να περιέχει τουλάχιστον έναν κόσμο όπου η λεπτομέρεια είναι υψηλή.

Το επόμενο παράδειγμα χρησιμοποιεί ομαλές εκφράσεις στο επίπεδο των τμημάτων οντότητας και των τμημάτων έκφανσης, και ψάχνει για εκφάνσεις του menu στα Ελληνικά των music clubs και των εστιατορίων:

```
[% lang=gr]recreation_guide(.music_club | .restaurant).menu::[-]
```

Οι ομαλές εκφράσεις στα δυο επίπεδα μπορούν να συνδυαστούν, όπως δείχνει το παρακάτω παράδειγμα:

```
X.theater(".zip.*" | .[% season=summer]address.(code)|(zip.*"))
```

Σε αυτή την έκφραση μονοπατιού περιβάλλοντος, αν η ετικέτα μιας ακμής οντότητας που ξεκινά από κάποια έκφανση ενός θεάτρου<sup>80</sup> αρχίζει με «zip», τότε ο περιοριστής κληρονομούμενης κάλυψης υπονοείται και έχει την εξ ορισμού τιμή `[-]` για ολόκληρο το μονοπάτι. Εναλλακτικά, η έκφανση ενός θεάτρου μπορεί να οδηγεί σε μια ακμή `address` και η αντίστοιχη έκφανση της διεύθυνσης (`address`) να ακολουθείται από μια ακμή `code`, ή από μια ακμή που η ετικέτα της αρχίζει από «zip». Στην περίπτωση αυτή, ο περιοριστής κληρονομούμενης κάλυψης του μονοπατιού που αρχίζει με το `address` είναι `[% season=summer]`.

Η χρήση ομαλών εκφράσεων στο επίπεδο των ετικετών των ακμών οντότητας είναι ξεκάθαρη: από την άλλη, το επίπεδο των τμημάτων οντότητας και των τμημάτων έκφανσης απαιτεί περισσότερη προσοχή. Αν τα  $s_1$  και  $s_2$  είναι επιτρεπόμενες ακολουθίες τμημάτων οντότητας και τμημάτων έκφανσης, ένα **γενικευμένο στοιχείο έκφρασης μονοπατιού περιβάλλοντος** (general context path expression component ή για συντομία **gcpe\_component**) μπορεί να έχει τις μορφές:

$$s_1s_2 \quad s_1 \mid s_2 \quad (s_1) \quad (s_1)? \quad (s_1)+ \quad (s_1)^*$$

<sup>79</sup> Ένα τρίτο επίπεδο είναι αυτό των περιβαλλόντων, όπου θα θέλαμε να εκφράσουμε περιορισμούς πάνω σε συγκεκριμένες διαστάσεις χωρίς να ορίζουμε πλήρως έναν προσδιοριστή περιβάλλοντος. Αυτό καλύπτεται από τα **πρότυπα περιβάλλοντος**, που παρουσιάστηκαν στην Ενότητα 5.2.2.2.

<sup>80</sup> Θυμηθείτε ότι, παρόλο που αυτή η έκφραση μονοπατιού περιβάλλοντος δεν περιέχει καθόλου τμήματα έκφανσης, τα τμήματα έκφανσης υπονοούνται. Η ακμή οντότητας `theater` ακολουθείται από κάποια ακμή περιβάλλοντος που οδηγεί σε μια έκφανση του θεάτρου.

Το σύμβολο  $|$  σημαίνει διάζευξη, το  $?$  σημαίνει 0 ή 1 εμφανίσεις, το  $+$  σημαίνει 1 ή περισσότερες εμφανίσεις, και το  $*$  σημαίνει 0 ή περισσότερες εμφανίσεις.

Για να αποφύγουμε τον σχηματισμό μη επιτρεπόμενων εκφράσεων μονοπατιού περιβάλλοντος, όπως είναι το `x.label (:: [c]) +` όπου τμήματα έκφανσης συσσωρεύονται το ένα μετά το άλλο, διακρίνουμε τέσσερις κατηγορίες των `gere_components` ανάλογα με τον τύπο του αρχικού και του τελικού τους τμήματος:

- $e-e$ , που αρχίζει με τμήμα οντότητας και τελειώνει με τμήμα οντότητας.
- $e-f$ , που αρχίζει με τμήμα οντότητας και τελειώνει με τμήμα έκφανσης.
- $f-f$ , που αρχίζει με τμήμα έκφανσης και τελειώνει με τμήμα έκφανσης.
- $f-e$ , που αρχίζει με τμήμα έκφανσης και τελειώνει με τμήμα οντότητας.

Τότε, το  $s_1s_2$  είναι επιτρεπόμενο μόνον εάν το αρχικό τμήμα του  $s_2$  επιτρέπεται να ακολουθεί το τελικό τμήμα του  $s_1$ , ενώ η διάζευξη  $|$  μπορεί να χρησιμοποιηθεί μόνο ανάμεσα από `gere_components` της ίδιας κατηγορίας. Τα σύμβολα  $?$ ,  $+$ , και  $*$  μπορούν να χρησιμοποιηθούν με `gere_components`  $e-f$  και  $f-e$ , ενώ για τα  $e-e$  και  $f-f$  επιτρέπονται μόνο τα  $(e-e) +$  και  $(f-f) ?$ .

Γενικά, δεν είναι δυνατόν να εισαχθούν μεταβλητές στα `gere_components`. Ο μόνος τύπος μεταβλητής που μπορεί να εμφανιστεί σε ένα `gere_component` είναι μεταβλητές περιβάλλοντος στην θέση περιοριστή κληρονομούμενης κάλυψης (δεν μπορούν να εμφανιστούν μεταβλητές περιβάλλοντος σαν περιοριστές ρητού περιβάλλοντος στα `gere_components`). Στην περίπτωση που μια μεταβλητή περιβάλλοντος εμφανίζεται σε μια διάζευξη, όπως στο `(.[X]a | .[Y]b)`, και ανάλογα με το μονοπάτι δεδομένων περιβάλλοντος που ταιριάζει, η τιμή της μπορεί να είναι η ειδική τιμή *nil*<sup>81</sup>.

Σχετικά με τα σύμβολα  $+$  και  $*$ , οι περιοριστές κληρονομούμενης κάλυψης σε ένα `gere_component` χρησιμοποιούνται μόνον στην πρώτη εμφάνιση και δεν επαναλαμβάνονται στις ακόλουθες εμφανίσεις<sup>82</sup>. Για παράδειγμα, το `(.[% quality=high]part) +` είναι ένα `gere_component` που δηλώνει εκφράσεις σαν `.[% quality=high]part.part...part`. Παρατηρήστε ότι ο περιοριστής κληρονομούμενης κάλυψης `[% quality=high]` αναφέρεται σε ολόκληρο το μονοπάτι και δεν επαναλαμβάνεται για κάθε ξεχωριστή ακμή οντότητας `part`.

Το σύμβολο  $*$  του Kleene closure εισάγει ένα πρόβλημα όταν χρησιμοποιείται σε βάσεις δεδομένων γράφου που περιέχουν κύκλους: μια ομαλή έκφραση που τελειώνει σε  $*$  ή σε  $+$  μπορεί να ταιριάζει με έναν άπειρο αριθμό μονοπατιών δεδομένων. Υπάρχει ένας αριθμός τρόπων για να αντιμετωπιστεί αυτό το ζήτημα, ένας από τους οποίους είναι να μην

<sup>81</sup> Όπως και το *nil* για μεταβλητές αντικειμένου, η παρουσία του *nil* σε μια συνθήκη περιβάλλοντος κάνει πάντα την συνθήκη περιβάλλοντος ψευδή. Επιπλέον, μια τιμή *nil* μιας μεταβλητής περιβάλλοντος δεν προκαλεί την δημιουργία ενός νέου αντικειμένου στον γράφο του αποτελέσματος. Πράξεις περιβάλλοντος με το *nil* αποτιμούνται σε *nil*, ενώ η παρουσία του *nil* σε μια συναθροιστική συνάρτηση περιβάλλοντος αγνοείται. Όπως ήδη αναφέρθηκε, το `extension(nil)` επιστρέφει ένα σύνολο που περιέχει μόνο *nil*.

<sup>82</sup> Όπως δείξαμε στην Ενότητα 5.2.2.1, η επανάληψη ενός περιοριστή κληρονομούμενης κάλυψης κατά μήκος του μονοπατιού που περιορίζει δίνει μια ισοδύναμη έκφραση μονοπατιού περιβάλλοντος μόνον εάν ο περιοριστής έχει την μορφή ενός προσδιοριστή περιβάλλοντος. Αν ο περιοριστής είναι πρότυπο περιβάλλοντος ή μεταβλητή περιβάλλοντος, τότε η νέα έκφραση μονοπατιού περιβάλλοντος δεν είναι ισοδύναμη με την αρχική. Χρησιμοποιώντας περιοριστές κληρονομούμενης κάλυψης μόνο στην πρώτη εμφάνιση, επιτρέπουμε στα πρότυπα περιβάλλοντος και στις μεταβλητές περιβάλλοντος στις ομαλές εκφράσεις να προσάπτονται σε μονοπάτια αντί για ξεχωριστές ακμές.

επιτρέπουμε την διάσχιση του ίδιου αντικειμένου δεύτερη φορά όταν έχουμε σύγκριση με μια ομαλή έκφραση αυτού του είδους [AQM+97].

### 5.4.1.2 Μπαλαντέρ (wildcards)

Οι **μπαλαντέρ** (wildcards) είναι χρήσιμοι όταν κάποιες από τις ετικέτες σε ένα μονοπάτι δεν είναι γνωστές, ή η σχετική τους θέση σε ένα μονοπάτι δεν είναι γνωστή. Έχουμε ήδη αναφέρει ότι ο κενός περιοριστής περιβάλλοντος [-] συμπεριφέρεται σαν ένας **μπαλαντέρ περιβάλλοντος**, ταιριάζοντας με οποιοδήποτε περιβάλλον. Ανάλογα, ο μπαλαντέρ - ταιριάζει με κάθε ετικέτα ακμής οντότητας. Για παράδειγμα θεωρήστε την ακόλουθη έκφραση μονοπατιού περιβάλλοντος, η οποία χρησιμοποιεί τον μπαλαντέρ - σε μια ομαλή έκφραση, και ταιριάζει με κάθε έκφραση χαμηλής λεπτομέρειας του review στην βάση δεδομένων, οσοδήποτε βαθιά και αν βρίσκεται αυτή:

```
recreation_guide(.-::[-])*review::[% detail=low] [-]
```

Ο μπαλαντέρ # χρησιμοποιείται για να υποδηλώσει ένα μονοπάτι που αποτελείται από μηδέν ή περισσότερα ζεύγη ακμής οντότητας / ακμής περιβάλλοντος. Συγκεκριμένα, το .# είναι συντομογραφία για το (.-::[-])\* , και είναι ένα e-f gpre\_component που σημαίνει εκφράσεις της μορφής .-::[-].-::[-].-...-::[-]. Από την άλλη, το ::# είναι συντομογραφία για το (::[-].-)\* , και είναι ένα f-e gpre\_component που σημαίνει εκφράσεις της μορφής ::[-].-::[-].-...::[-].<sup>83</sup> Χρησιμοποιώντας το #, το παραπάνω παράδειγμα μπορεί να γράφει σαν:

```
recreation_guide.#review::[% detail=low] [-]
```

Ο μπαλαντέρ # μπορεί να ακολουθεί έναν περιοριστή κληρονομούμενης κάλυψης: το e-f gpre\_component .[c]# είναι συντομογραφία για το (.[c]-::[-])\* , και σημαίνει εκφράσεις της μορφής .[c]-::[-].-::[-].-...-::[-], ενώ το f-e gpre\_component ::[c]# είναι συντομογραφία για το (::[c] [-].-)\* , και σημαίνει εκφράσεις της μορφής ::[c] [-].-::[-].-...::[-] :- που έχουν 0 ή περισσότερα ζεύγη τμήματος έκφρασης / τμήματος περιβάλλοντος.

### 5.4.1.3 Μεταβλητές μονοπατιού και μεταβλητές ετικέτας

Η MQL ενσωματώνει μεταβλητές μονοπατιού όπως απαντώνται στην Lorel, και μεταβλητές ετικέτας όπως απαντώνται στην UnQL.

Οι **μεταβλητές μονοπατιού** έχουν την μορφή @X και δεσμεύονται σε **μονοπάτια δεδομένων περιβάλλοντος** (που ορίστηκαν στην Ενότητα 5.2.1.2). Μπορούν να χρησιμοποιηθούν στον όρο where για να ελέγξουν μονοπάτια δεδομένων περιβάλλοντος για ισότητα, ή στον όρο select για να εμφανίσουν μονοπάτια δεδομένων περιβάλλοντος στα αποτελέσματα της επερώτησης. Μια δεσμευμένη μεταβλητή μονοπατιού @X μπορεί επίσης να παρουσιαστεί στην συνάρτηση pathof(@X)<sup>84</sup>, η οποία απογυμνώνει ένα μονοπάτι δεδομένων περιβάλλοντος από τα oids και τις κληρονομούμενες καλύψεις, και το μετατρέπει σε μια ακολουθία ετικετών ακμών οντότητας και ετικετών ακμών περιβάλλοντος της μορφής «...a::[ec<sub>1</sub>].b::[ec<sub>2</sub>]...». Αν μια μεταβλητή μονοπατιού εμφανίζεται στο select,

<sup>83</sup> Μια και το υπονοούμενο τελικό τμήμα του ::# είναι τμήμα οντότητας, η έκφραση ::#::f είναι επιτρεπόμενη (όπου το ::f είναι τμήμα έκφρασης). Παρόμοια, επειδή το υπονοούμενο τελικό τμήμα του .# είναι τμήμα έκφρασης, η έκφραση .#::f είναι μη επιτρεπόμενη.

<sup>84</sup> Για μια μεταβλητή μονοπατιού @X, η ανάλογη συνάρτηση στην Lorel είναι η pathof(X), που δεν επαναλαμβάνει τον χαρακτήρα @. Σε αντίθεση με την Lorel, η MQL θεωρεί το @ σαν μέρος του αναγνωριστικού μιας μεταβλητής μονοπατιού.

δημιουργείται ένας νέος κόμβος στον γράφο του αποτελέσματος για κάθε δεσμευμένο μονοπάτι, έχοντας το μονοπάτι σαν τιμή τύπου στοιχειοσειράς.

```
select distinct with_menu_in_greek: pathof(@P)
from recreation_guide(.#)@P.menu::[% lang=gr]
```

Η παραπάνω επερώτηση επιστρέφει τα μονοπάτια που οδηγούν σε πολυδιάστατα αντικείμενα menu που περιλαμβάνουν μια Ελληνική έκφραση. Το μέρος της έκφρασης μονοπατιού περιβάλλοντος που αντιστοιχεί σε μια μεταβλητή μονοπατιού θα πρέπει να εγκλειστεί σε παρενθέσεις, όπως στο (.#)@P. Η μεταβλητή μονοπατιού @P δεσμεύεται στα μονοπάτια δεδομένων περιβάλλοντος που ταιριάζουν, τα οποία αρχίζουν από μια έκφραση της ρίζας και τελειώνουν σε έναν κόμβο περιβάλλοντος από τον οποίο ξεκινά μια ακμή menu. Επειδή πολλά διαφορετικά μονοπάτια δεδομένων περιβάλλοντος για διαφορετικά clubs μπορούν να δώσουν το ίδιο μονοπάτι (ακολουθία ετικετών των εδρών), χρησιμοποιούμε το distinct με το pathof(@P) για την απαλοιφή διπλοτύπων. Για το ΜΟΕΜ στην Εικόνα 5.1 επιστρέφεται μόνον ένα μονοπάτι, που αποτελείται από μια ακμή οντότητας και μια ακμή περιβάλλοντος.

```
{with_menu_in_greek: "music_club::["]}
```

Η ακόλουθη επερώτηση χρησιμοποιεί μπαλαντέρ και μεταβλητές μονοπατιού προκειμένου να εντοπίσει ακμές στην βάση δεδομένων οι οποίες δεν υφίστανται κάτω από κανέναν κόσμο.

```
select icv_violation: @P
from root::#(:[X] [-])@P
within [X]=[-]
union
select icv_violation: @P
from root.#(. [X] -)@P
within [X]=[-]
```

Η επερώτηση είναι η ένωση δυο ξεχωριστών επερωτήσεων. Η πρώτη επερώτηση χρησιμοποιεί μια μεταβλητή μονοπατιού που δεσμεύεται σε κάθε ακμή περιβάλλοντος με κενή κληρονομούμενη κάλυψη, ενώ η δεύτερη χρησιμοποιεί μια μεταβλητή μονοπατιού που δεσμεύεται σε κάθε ακμή οντότητας με κενή κληρονομούμενη κάλυψη. Η λέξη-κλειδί union συγχωνεύει τις ρίζες των δυο γράφων του αποτελέσματος σε μια κοινή ρίζα, και στην συνέχεια απαλείφει διπλότυπες ακμές<sup>85</sup> που ξεκινούν από την νέα ρίζα. Η μορφή των τελικών αποτελεσμάτων απεικονίζεται στην Εικόνα 5.4 (a). Η τιμή του πρώτου αντικείμενου αντιστοιχεί σε μια ακμή περιβάλλοντος, ενώ η τιμή του δεύτερου σε μια ακμή οντότητας: το [icv] είναι η κενή κληρονομούμενη κάλυψη, το [ec] υποδηλώνει κάποιο ρητό περιβάλλον, και το label υποδηλώνει κάποια ετικέτα ακμής οντότητας.

Σε έναν Πολυδιάστατο Γράφο Δεδομένων, αν ένας κόμβος έχει κενή κληρονομούμενη κάλυψη τότε οι εισερχόμενες και οι εξερχόμενες ακμές έχουν και αυτές κενή κληρονομούμενη κάλυψη. Συνεπώς, αν η παραπάνω επερώτηση δεν επιστρέψει κανένα αντικείμενο icv\_violation, η βάση δεδομένων δεν περιέχει καμία ακμή ή κόμβο με κενή κληρονομούμενη κάλυψη. Για να διασφαλίσουμε ότι ο γράφος της βάσης δεδομένων είναι στην πραγματικότητα ΜΟΕΜ, πρέπει επίσης να ελέγξουμε εάν είναι αιτιοκρατικός ως προς

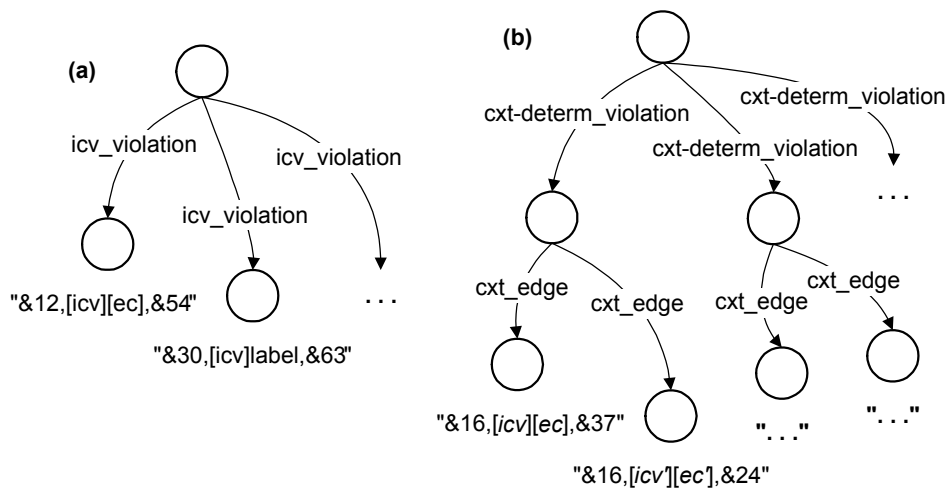
<sup>85</sup> Στην περίπτωση του union, όπως επίσης και στο intersect και στο except που εκτελούν τις ανάλογες πράξεις μεταξύ επερωτήσεων, σαν διπλότυπες ακμές θεωρούνται οι ακμές που έχουν την ίδια ετικέτα και που οδηγούν σε αντικείμενα που αντιστοιχούν στο ίδιο αντικείμενο της βάσης δεδομένων.

το περιβάλλον. Η ακόλουθη επερώτηση επιστρέφει ζευγάρια ακμών περιβάλλοντος που παραβιάζουν την αιτιοκρατία ως προς το περιβάλλον, αν υπάρχουν:

```
select cxt-determ_violation: {cxt_edge: @P, cxt_edge: @Q}
from root::# <X>,
    <X> (:: [I] [-]) @P Y,
    <X> (:: [J] [-]) @Q Z
where Y != Z
within [I] * [J] != [-]
```

Το αποτέλεσμα της επερώτησης έχει την μορφή που απεικονίζει η Εικόνα 5.4 (b)<sup>86</sup>.

**Εικόνα 5.4:** Ανεύρεση κενών κληρονομούμενων καλύψεων και παραβιάσεων της αιτιοκρατίας ως προς το περιβάλλον σε βάσεις δεδομένων γράφων.



Οι **μεταβλητές ετικέτας** έχουν την μορφή  $\$X$ , και δεσμεύονται σε ετικέτες ακμών οντότητας. Μια μεταβλητή ετικέτας δεν μπορεί να δεσμευτεί σε ετικέτες ακμών περιβάλλοντος: στην περίπτωση αυτή, οι ετικέτες είναι ρητά περιβάλλοντα και πρέπει να χρησιμοποιηθούν μεταβλητές περιβάλλοντος. Οι μεταβλητές ετικέτας μπορούν να ιδωθούν σαν μια ειδική περίπτωση των μεταβλητών μονοπατιού: το  $\$X$  θα μπορούσε να εκφραστεί σαν `pathof(@X)`, με το `@X` να δεσμεύεται σε ένα μονοπάτι που αποτελείται από μια μόνο ακμή οντότητας. Το πλεονέκτημα του να έχει κανείς μεταβλητές ετικέτας σαν έναν ξεχωριστό τύπο μεταβλητών είναι ότι μπορούν να χρησιμοποιηθούν στην θέση των ετικετών ακμών οντότητας κατά την κατασκευή των αποτελεσμάτων<sup>87</sup>. Επιπλέον, όπως αναφέρεται στο [ABS00], οι μεταβλητές ετικέτας μπορούν να χρησιμοποιηθούν για να «κρύψουν» συγκεκριμένα υποαντικείμενα ενός αντικειμένου<sup>88</sup> στα αποτελέσματα.

<sup>86</sup> Σημειώστε ότι για κάθε μια παραβίαση θα υπάρχουν δυο αντικείμενα `cxt-determ_violation` στον γράφο του αποτελέσματος, τα οποία αντιστοιχούν σε αντικατοπτρικές δεσμεύσεις των  $Y$  και  $Z$ .

<sup>87</sup> Μια μεταβλητή μονοπατιού δεν μπορεί να χρησιμοποιηθεί για την κατασκευή ακμών στα αποτελέσματα, επειδή η στοιχειοσειρά του μονοπατιού στην γενική περίπτωση δεν αποτελεί επιτρεπόμενη ετικέτα για μια ακμή.

<sup>88</sup> Η Ενότητα 5.4.3.1 παρουσιάζει ένα παράδειγμα επερώτησης που χρησιμοποιεί μια μεταβλητή περιβάλλοντος προκειμένου να «κρύψει» εκφάνσεις αντικειμένων ανάλογα με τα ρητά περιβάλλοντα (ετικέτες των ακμών περιβάλλοντος). Οι μεταβλητές ετικέτας μπορούν να χρησιμοποιηθούν με

Θεωρήστε την επερώτηση:

```
select $L: <X>
from recreation_guide.-$L.name <X>
```

Στην παραπάνω επερώτηση, η μεταβλητή ετικέτας \$L δεσμεύεται στις ετικέτες των ακμών οντότητας που ξεκινούν από εκφάνσεις της *ρίζας* και οδηγούν μέσω μιας ακμής περιβάλλοντος σε πολυδιάστατα αντικείμενα name. Η μεταβλητή δεσμεύεται σε κάθε ετικέτα ακμής οντότητας, επειδή ο μπαλαντέρ - ταιριάζει με κάθε ετικέτα ακμής οντότητας. Η επερώτηση επιστρέφει πολυδιάστατα αντικείμενα name, που τα δείχνουν ακμές οντότητας που υποδηλώνουν το είδος του αντιστοίχου μαγαζιού (music club, εστιατόριο, κτλ.).

Οι μεταβλητές ετικέτας μπορούν επίσης να εμφανιστούν στην θέση της τιμής αντικειμένων στο select, μετατρέποντας μεταπληροφορία της βάσης δεδομένων σε δεδομένα του αποτελέσματος. Η ακόλουθη επερώτηση MQL

```
select distinct restaurant_attr: $L
from recreation_guide.restaurant.-$L
```

επιστρέφει όλα τα διαφορετικά γνωρίσματα (attributes) των εστιατορίων στην βάση δεδομένων στην Εικόνα 5.1, σαν τιμές αντικειμένων:

```
{restaurant_attr: "name",
 restaurant_attr: "address",
 restaurant_attr: "review",
 restaurant_attr: "parking" }
```

Προσέξτε ότι η MQL περικλείει τις τιμές των μεταβλητών ετικέτας σε εισαγωγικά όταν χρησιμοποιούνται σαν τιμές αντικειμένων στον οδηγό του όρου select.

Συνοψίζοντας, οι μεταβλητές ετικέτας δεσμεύονται σε ετικέτες ακμών οντότητας, και οι μεταβλητές περιβάλλοντος όταν χρησιμοποιούνται σαν περιοριστές ρητού περιβάλλοντος δεσμεύονται σε ετικέτες ακμών περιβάλλοντος (ρητά περιβάλλοντα). Από την άλλη, μια μεταβλητή μονοπατιού μπορεί να δεσμευτεί σε ακμές οντότητας, ακμές περιβάλλοντος, ή οποιαδήποτε ακολουθία των δυο, αλλά δεν μπορεί να χρησιμοποιηθεί σαν ετικέτα ακμής στον οδηγό του select για την κατασκευή των αποτελεσμάτων.

Οι μεταβλητές ετικέτας και οι μεταβλητές μονοπατιού επισυνάπτονται συνήθως σε μπαλαντέρ και σε gcre\_components, και δεσμεύονται στα μέρη του γράφου που ταιριάζουν. Όμως, ένα gcre\_component δεν επιτρέπεται το ίδιο να περιέχει μεταβλητές αντικειμένου (περικλεισμένες σε {}), μεταβλητές μονοπατιού, μεταβλητές ετικέτας, ή μεταβλητές περιβάλλοντος που χρησιμοποιούνται σαν περιοριστές ρητού περιβάλλοντος. Οι εκφράσεις μονοπατιού περιβάλλοντος που περιέχουν τις παραπάνω μεταβλητές καλούνται **επαυξημένες γενικευμένες εκφράσεις μονοπατιού περιβάλλοντος**, και η σύνταξή τους προσδιορίζεται αυστηρά στο Παράρτημα A.

## 5.4.2 Εμφωλιασμένες επερωτήσεις MQL

Οι επερωτήσεις MQL μπορούν να είναι εμφωλιασμένες. Μια υποεπερώτηση μπορεί να εμφανίζεται στον όρο select μιας επερώτησης, και οι νέοι κόμβοι που κατασκευάζονται από την εσωτερική επερώτηση ενσωματώνονται στα αποτελέσματα της εξωτερικής

---

παρόμοιο τρόπο για να «κρύψουν» πολυδιάστατα αντικείμενα ανάλογα με τις ετικέτες των ακμών οντότητας.



επερώτησης. Οι εσωτερικές επερωτήσεις μπορούν να χρησιμοποιήσουν μεταβλητές που έχουν δεσμευτεί στις εξωτερικές επερωτήσεις, όπως στο:

```
select club_comments: (select gr_comments: Y
                        from <X>.#.[% lang=gr] comments Y)
from recreation_guide.- <X>
```

Στην παραπάνω επερώτηση, η μεταβλητή <X> δεσμεύεται σε πολυδιάστατα αντικείμενα `music_club` και `restaurant`. Στην συνέχεια, για κάθε τέτοια δέσμευση, η εσωτερική επερώτηση βρίσκει τις εκφάνσεις των `comments` στα Ελληνικά, ανεξάρτητα από το πόσο βαθιά βρίσκονται στον γράφο της βάσης δεδομένων. Σαν αποτέλεσμα η εξωτερική επερώτηση ομαδοποιεί τις εκφάνσεις των `comments` που έχουν ταιριάζει ανάλογα με το εστιατόριο ή το `music club` στο οποίο ανήκουν (πράγμα που, όπως αναφέραμε στην Ενότητα 5.3.2, δεν συμβαίνει αυτόματα στην MQL). Όταν αποτιμάται πάνω στον γράφο στην Εικόνα 5.1, η επερώτηση επιστρέφει:

```
{club_comments: {gr_comments: &10 "..."},
 club_comments: {gr_comments: &41 "...", gr_comments: &36 "..."}
}
```

Η πρώτη γραμμή αντιστοιχεί στο `music_club`, ενώ το δεύτερο `club_comments` περιέχει τα σχόλια (`comments`) για το `restaurant`. Σημειώστε ότι αν ο περιοριστής κληρονομούμενης κάλυψης [% lang=gr] δεν υπήρχε στην επερώτηση, τότε η δεύτερη γραμμή των αποτελεσμάτων θα περιείχε επίσης και τον κόμβο &42. Άλλο ένα ενδιαφέρον σημείο είναι η χρήση της μεταβλητής πολυδιάστατου αντικειμένου <x> αντί για την μεταβλητή αντικειμένου περιβάλλοντος x. Ο λόγος για αυτό είναι ότι θέλουμε να ομαδοποιήσουμε τα σχόλια σύμφωνα με την πολυδιάστατη οντότητα στην οποία αναφέρονται, και όχι σύμφωνα με μια συγκεκριμένη έκφανση κάποιου club. Στην Εικόνα 5.1 η πολυδιάστατη οντότητα `music_club` αποτελείται από μια και μόνη έκφανση, αλλά στην γενική περίπτωση θα μπορούσε να περιλαμβάνει αρκετές εκφάνσεις, με κάθε μια να οδηγεί σε έναν αριθμό κόμβων `comments`.

Χάριν συγκρίσεως, να μια παρόμοια επερώτηση που δεν ομαδοποιεί τα σχόλια:

```
select gr_comments: X
from recreation_guide.#.[% lang=gr] comments X
```

Η επερώτηση δίνει:

```
{gr_comments: &10 "...",
 gr_comments: &41 "...",
 gr_comments: &36 "..."}
}
```

Στην Ενότητα 5.3.5 αναφέραμε ότι οι συναθροιστικές συναρτήσεις όπως η `union([c])` μπορούν να χρησιμοποιηθούν στον όρο `context`: το σχετικό παράδειγμα στην Ενότητα 5.3.5 επέστρεφε τους κόσμους κάτω από τους οποίους υπάρχει μια έκφανση του `menu` για ένα `music club` με συγκεκριμένο `oid`. Το ακόλουθο παράδειγμα προχωρά μακρύτερα, και επιστρέφει τις γλώσσες στις οποίες είναι διαθέσιμο το μενού για κάθε εστιατόριο και `music club` στην βάση δεδομένων.

```
select place: (select place_name: N, menu_langs: [Z]
               context [Z] as union([Y]) * [% lang in ALL]
               from <X>.menu::[Y] [-],
                  <X>.name N)
from recreation_guide.- <X>
```

Το παραπάνω παράδειγμα χρησιμοποιεί μια εμφωλιασμένη επερώτηση για να κάνει «group by» τα περιβάλλοντα προτού να εκτελεστεί η συναθροιστική συνάρτηση. Συγκεκριμένα, η εξωτερική επερώτηση δεσμεύει την μεταβλητή <X> σε music clubs και εστιατόρια. Στην συνέχεια, για κάθε πολυδιάστατο αντικείμενο music\_club ή restaurant, η εσωτερική επερώτηση δεσμεύει την μεταβλητή N στις εκφάνσεις του name, και την μεταβλητή περιβάλλοντος [Y] στην κληρονομούμενη κάλυψη των ακμών περιβάλλοντος που οδηγούν σε εκφάνσεις του menu. Η ένωση περιβάλλοντος αυτών των κληρονομούμενων καλύψεων υπολογίζεται στο όρο context για το τρέχον music club ή εστιατόριο, χρησιμοποιώντας την συναθροιστική συνάρτηση union([c]). Το αποτέλεσμα της συναθροιστικής συνάρτησης είναι ένας προσδιοριστής περιβάλλοντος που αναπαριστά τους κόσμους κάτω από τους οποίους υφίστανται εκφάνσεις του μενού για το συγκεκριμένο μέρος (music club ή εστιατόριο). Αυτός ο προσδιοριστής περιβάλλοντος είναι δυνατόν να περιέχει διαστάσεις διαφορετικές του lang, όπως για παράδειγμα τις detail και season· για να περιορίσουμε το αποτέλεσμα στην lang, τέμνουμε τον προσδιοριστή περιβάλλοντος με το πρότυπο περιβάλλοντος [% lang in ALL], το οποίο απομακρύνει τους προσδιοριστές διάστασης που δεν αναφέρονται στην lang. Όπως έχουμε αναφέρει, το ALL είναι συντομογραφία για ολόκληρο το πεδίο ορισμού μιας διάστασης. Έτσι, το [Z] υποδηλώνει τις γλώσσες για τις οποίες υπάρχει κάποιος κόσμος κάτω από τον οποίο υφίσταται μια έκφανση του μενού. Όταν αποτιμηθεί στον γράφο στην Εικόνα 5.1, η επερώτηση επιστρέφει την mssd-expression:

```
{place: {place_name: &3 "...", menu_langs: [lang in {gr,en,fr}]},
 place: {place_name: &16 "..."} }
```

Το πρώτο αντικείμενο place αντιστοιχεί στο music club στην Εικόνα 5.1. Το δεύτερο αντικείμενο place αντιστοιχεί στο restaurant στην Εικόνα 5.1, και δεν περιέχει κάποιο αντικείμενο menu\_langs επειδή η μεταβλητή [Y] είναι δεσμευμένη στο nil. Προκειμένου να περιορίσουμε τα αποτελέσματα σε αντικείμενα που πράγματι περιέχουν ένα μενού, θα μπορούσαμε να προσθέσουμε τον όρο «where exists <X>.menu::[-]» στην επερώτηση.

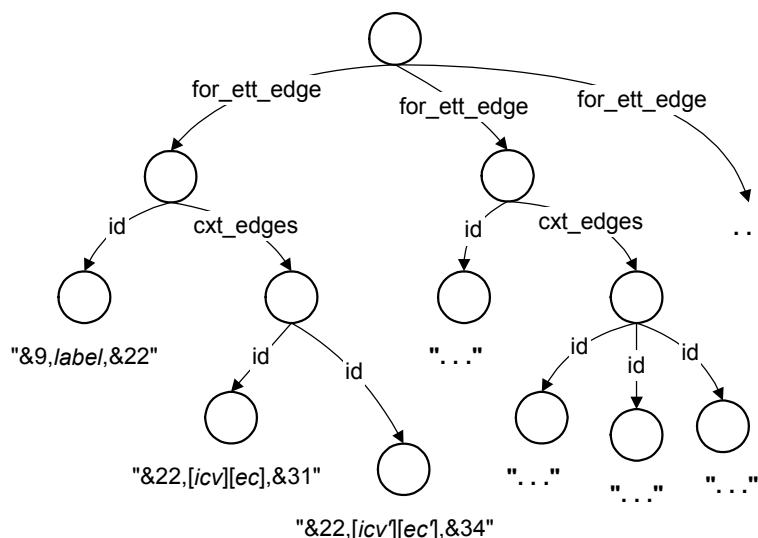
Παρατηρήστε ότι αν ένα πολυδιάστατο αντικείμενο name ενός music club ή εστιατορίου έχει πολλές εκφάνσεις, το αντίστοιχο αντικείμενο place θα περιέχει ισάριθμα ζευγάρια (place\_name, menu\_langs). Χρησιμοποιώντας μια μεταβλητή πολυδιάστατου αντικειμένου <N> αντί για την N, μπορούμε να αποφύγουμε την επανάληψη αυτή· όμως, τα ζευγάρια θα εξακολουθούν να εμφανίζονται εάν ένα music club ή εστιατόριο περιέχει πολλά πολυδιάστατα αντικείμενα name. Η ομαδοποίηση των (υποθετικά πολλών) αντικειμένων name ενός μαγαζιού θα απαιτούσε μια πρόσθετη εμφωλιασμένη επερώτηση.

Στο MOEM, κάποιες εκφάνσεις μιας πολυδιάστατης οντότητας είναι προσβάσιμες μέσω μιας εισερχόμενης ακμής οντότητας, ενώ άλλες είναι πιθανόν να μην υφίστανται κάτω από κανέναν κοινό κόσμο με την εισερχόμενη ακμή. Συνεπώς, ανάλογα με το όνομα (ετικέτα ακμής οντότητας) που χρησιμοποιούμε για μια πολυδιάστατη οντότητα, διαφορετικές εκφάνσεις μπορεί να είναι προσβάσιμες κάτω από τους αντίστοιχους κόσμους. Η ακόλουθη επερώτηση βρίσκει τις εκφάνσεις που δεν είναι προσβάσιμες μέσω εισερχόμενων ακμών οντότητας, για κάθε ακμή οντότητας στην βάση δεδομένων.

```
select for_ett_edge: {id: @P, cxt_edges:
                    (select id: @Q
                     from <X> (:: [J] [-]) @Q
                     within [I] * [J] = [-]) }
from root.# (. [I] -) @P <X>
```

Το αποτέλεσμα της επερώτησης έχει την μορφή του γράφου στην Εικόνα 5.5.

**Εικόνα 5.5:** Ανεύρεση μη προσβάσιμων εκφάνσεων για κάθε εισερχόμενη ακμή οντότητας.



Για κάθε ακμή οντότητας, το αντικείμενο `id` καθορίζει την ακμή, και το αντικείμενο `cxt_edges` περιέχει τα `ids` των μη προσβάσιμων ακμών περιβάλλοντος (που οδηγούν σε μη προσβάσιμα αντικείμενα περιβάλλοντος). Αν όλες οι εκφάνσεις της πολυδιάστατης οντότητας είναι προσβάσιμες μέσω κάποιας ακμής οντότητας, το αντίστοιχο `cxt_edges` δεν έχει καθόλου `id` παιδιά<sup>89</sup>. Παρατηρήστε πως η μεταβλητή περιβάλλοντος `[I]`, που εισάγεται στην εξωτερική επερώτηση, χρησιμοποιείται στην εσωτερική επερώτηση για να προσδιορίσει εάν μια ακμή οντότητας και μια ακμή περιβάλλοντος δεν υφίστανται κάτω από κανέναν κοινό κόσμο.

### 5.4.3 Κατασκευή αποτελεσμάτων

Το αποτέλεσμα μιας επερώτησης MQL είναι ένας Πολυδιάστατος Γράφος Δεδομένων στην μορφή μιας έκφρασης `mssd-expression`. Η MQL κατασκευάζει τα αποτελέσματα σύμφωνα με έναν οδηγό που ορίζεται στον όρο `select`. Λόγω του ότι ο Πολυδιάστατος Γράφος Δεδομένων έχει δυο τύπους κόμβων και δυο τύπους ακμών, η MQL θα πρέπει να καλύψει περισσότερες εκδοχές από την περίπτωση που τα αποτελέσματα θα ήταν συμβατικά ημιδομημένα δεδομένα.

Στα ακόλουθα, αρχίζουμε εξηγώντας πως η MQL χειρίζεται τους δυο διαφορετικούς τύπους κόμβων και ακμών όταν κατασκευάζει τα αποτελέσματα. Στην συνέχεια συζητούμε την χρήση των εκφράσεων μονοπατιού περιβάλλοντος στον όρο `select`. Τέλος, συζητούμε το πως η MQL μπορεί να **ανάγει μερικώς** τον γράφο του αποτελέσματος, επιστρέφοντας τελικά έναν υπογράφο του οποίου κάθε κόμβος και ακμή υφίσταται κάτω από κάποιον κόσμο.

<sup>89</sup> Στην περίπτωση αυτή, ο γράφος του αποτελέσματος θα περιέχει σύνθετα αντικείμενα σαν φύλλα. Αυτό είναι επιτρεπόμενο στην MQL, που το αποτέλεσμα είναι ένας Πολυδιάστατος Γράφος Δεδομένων. Στην Ενότητα 5.4.3.3 εξηγούμε την χρήση της λέξης-κλειδιού `holding` η οποία απομακρύνει τέτοιους κόμβους.

### 5.4.3.1 Κατασκευή ακμών οντότητας και ακμών περιβάλλοντος

Ενώ οι εκφράσεις μονοπατιού περιβάλλοντος είναι χτισμένες γύρω από την κανονική μορφή ενός Πολυδιάστατου Γράφου Δεδομένων, ο γράφος του αποτελέσματος μιας επερωτήσης MQL δεν είναι απαραίτητο να βρίσκεται σε κανονική μορφή. Συνεπώς, όταν κατασκευάζει τα αποτελέσματα, η MQL πρέπει μόνο και μόνο να διασφαλίζει ότι οι ακμές που ξεκινούν από έναν κόμβο έχουν τον σωστό τύπο: ακμές οντότητας πρέπει να ξεκινούν από κόμβους περιβάλλοντος, και ακμές περιβάλλοντος πρέπει να ξεκινούν από πολυδιάστατους κόμβους. Στην MQL, τα αποτελέσματα κατασκευάζονται με βάση τον οδηγό εκφράσεων `mssd-expression` στον όρο `select`, που ορίζει την σειρά των κόμβων περιβάλλοντος και των πολυδιάστατων κόμβων στα αποτελέσματα. Στον οδηγό αυτό, οι αγκύλες `{...}` δηλώνουν αντικείμενα περιβάλλοντος, και τα άγκιστρα `<...>` δηλώνουν πολυδιάστατα αντικείμενα.

Ο οδηγός μπορεί να περιέχει τα ακόλουθα στην θέση μιας ετικέτας *ακμής οντότητας*:

- Μια στοιχειοσειρά που συμμορφώνεται με το συντακτικό των ετικετών των ακμών οντότητας. Η στοιχειοσειρά αυτή γίνεται η ετικέτα των ακμών οντότητας στα αποτελέσματα.
- Μια μεταβλητή ετικέτας. Στα αποτελέσματα δημιουργούνται ακμές οντότητας που έχουν σαν ετικέτες τις τιμές που είναι δεσμευμένες σε αυτή την μεταβλητή ετικέτας.

Ο οδηγός μπορεί να περιέχει τα ακόλουθα στην θέση μιας ετικέτας *ακμής περιβάλλοντος*:

- Μια στοιχειοσειρά που συμμορφώνεται με το συντακτικό των προσδιοριστών περιβάλλοντος. Στα αποτελέσματα δημιουργούνται ακμές περιβάλλοντος με την στοιχειοσειρά αυτή σαν ετικέτα.
- Μια μεταβλητή περιβάλλοντος. Στα αποτελέσματα δημιουργούνται ακμές περιβάλλοντος που έχουν σαν ετικέτες τους προσδιοριστές περιβάλλοντος που είναι δεσμευμένοι σε αυτή την μεταβλητή περιβάλλοντος.

Συνοψίζοντας, ο οδηγός `mssd-expression` μπορεί να περιέχει σαν ετικέτες ακμών: (a) στοιχειοσειρές και μεταβλητές ετικέτας μέσα σε αγκύλες (αντικείμενα περιβάλλοντος), και (b) προσδιοριστές περιβάλλοντος και μεταβλητές περιβάλλοντος μέσα σε άγκιστρα (πολυδιάστατα αντικείμενα). Ο οδηγός μπορεί να περιέχει μεταβλητές κάθε τύπου σαν τιμές αντικειμένων: μεταβλητές αντικειμένου περιβάλλοντος, μεταβλητές πολυδιάστατου αντικειμένου, μεταβλητές περιβάλλοντος, μεταβλητές ετικέτας, ή μεταβλητές μονοπατιού.

Ειδικά για την ρίζα του γράφου του αποτελέσματος, οι αγκύλες ή τα άγκιστρα μπορούν να παραλειφθούν από τον οδηγό. Ο τύπος του κόμβου της ρίζας μπορεί να υπονοηθεί από τον τύπο των ετικετών<sup>90</sup>. Θεωρήστε τις δυο ισοδύναμες επερωτήσεις:

```
select <[Y]: X>
from recreation_guide.#.comments:: [Y] [-] X
```

```
select [Y]: X
from recreation_guide.#.comments:: [Y] [-] X
```

Το αποτέλεσμα των επερωτήσεων αυτών είναι η «αντιγραφή» όλων των ακμών περιβάλλοντος που δείχνουν σε κάποια έκφανση του `comments` από την βάση δεδομένων στην Εικόνα 5.1 στα αποτελέσματα, με την διαφορά ότι για κάθε ακμή περιβάλλοντος η κληρονομούμενη κάλυψη στην βάση δεδομένων μεταφέρεται σαν ρητό περιβάλλον στα

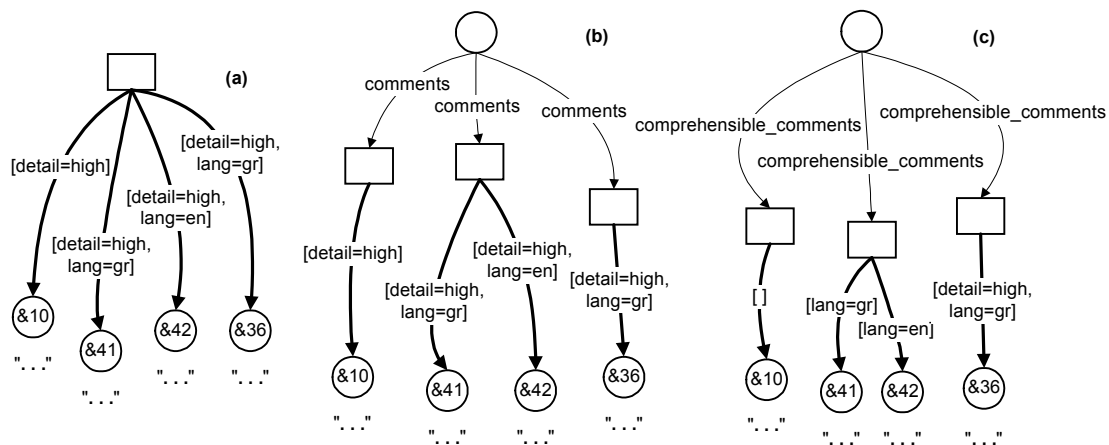
<sup>90</sup> Προφανώς, αν ο οδηγός ορίσει περισσότερες από μια ακμές που να ξεκινούν από την ρίζα, πρέπει όλες να συμφωνούν σχετικά με τον τύπο της ρίζας (περιβάλλοντος ή πολυδιάστατη).

αποτελέσματα. Όλες οι εκφάνσεις του `comments` συγκεντρώνονται κάτω από ένα πολυδιάστατο αντικείμενο, που είναι η ρίζα του γράφου του αποτελέσματος, και από την οποία ξεκινούν οι ακμές περιβάλλοντος:

```
<[detail=high]: &10 "...",
  [detail=high,lang=gr]: &41 "...",
  [detail=high,lang=en]: &42 "...",
  [detail=high,lang=gr]: &36 "...">
```

Το γεγονός ότι η ρίζα είναι πολυδιάστατο αντικείμενο δηλώνεται ρητά στην πρώτη επερώτηση μέσω των δυο αγκίστρων `<` και `>` τα οποία περικλείουν τον οδηγό `msd-expression` στον όρο `select`. Στην δεύτερη επερώτηση, το ίδιο γεγονός συνάγεται από την μεταβλητή περιβάλλοντος `[Y]` που εμφανίζεται στην θέση της ετικέτας της πρώτης ακμής. Τα αποτελέσματα απεικονίζονται σαν γράφος στην Εικόνα 5.6 (a).

Εικόνα 5.6: Αποτελέσματα ερωτήσεων MQL.



Η ακόλουθη εμφωλιασμένη επερώτηση βελτιώνει το παραπάνω παράδειγμα και διατηρεί την αρχική ομαδοποίηση των εκφάνσεων του `comments` κάτω από τα αντίστοιχά τους πολυδιάστατα αντικείμενα. Οι αγκύλες στην εξωτερική επερώτηση και τα άγκιστρα στην εσωτερική επερώτηση που αντιστοιχούν στην ρίζα μπορούν να παραλειφθούν. Η Εικόνα 5.6 (b) εμφανίζει τα αποτελέσματα.

```
select {comments: (select <[Y]: Z>
                    from <X>:::[Y] [-] Z)}
from recreation_guide.#.comments <X>
```

Το αποτέλεσμα της επερώτησης σε μορφή κειμένου είναι:

```
{comments: <[detail=high]: &10 "...",
  comments: <[detail=high,lang=gr]: &41 "...",
             [detail=high,lang=en]: &42 "...",
  comments: <[detail=high,lang=gr]: &36 "...">}
```

Σαν άλλο ένα παράδειγμα θεωρήστε την ακόλουθη επερώτηση, που χρησιμοποιεί μια μεταβλητή περιβάλλοντος προκειμένου να «κρύψει» από τα αποτελέσματα όλες τις εκφάνσεις του `comments` που δεν είναι σε μια από τις τέσσερις γλώσσες που καταλαβαίνει ο χρήστης. Το φιλτράρισμα των ακατανόητων σχολίων (`comments`) λαμβάνει χώρα στον όρο

within. Μια λεπτομέρεια είναι ότι στην περίπτωση αυτή η μεταβλητή περιβάλλοντος [Y] δεσμεύεται σε ρητά περιβάλλοντα, αντί για κληρονομούμενες καλύψεις.

```
select comprehensible_comments:
    (select [Y]: Z
     from <X>::[Y] Z
     within [Y] * [lang in {gr,en,fr,sp}] != [-])
from recreation_guide.#.comments <X>
```

Ο γράφος των αποτελεσμάτων δίνεται στην Εικόνα 5.6 (c). Για κάθε πολυδιάστατο αντικείμενο comments που δεσμεύεται στο <X>, κατασκευάζεται μια ακμή οντότητας με ετικέτα comprehensible\_comments που δείχνει σε έναν πολυδιάστατο κόμβο. Ο πολυδιάστατος κόμβος είναι η ρίζα της εσωτερικής επερώτησης, και αντιστοιχεί στον κόμβο που είναι η τρέχουσα τιμή του <X>, με την διαφορά ότι στα αποτελέσματα αντιγράφονται από την βάση δεδομένων μόνο εκφάνσεις σχολίων στα Ελληνικά, Αγγλικά, Γαλλικά, και Ισπανικά.

#### 5.4.3.2 Εκφράσεις μονοπατιού περιβάλλοντος στο “select”

Οι εκφράσεις μονοπατιού περιβάλλοντος μπορούν να εμφανιστούν στον όρο select. Η επερώτηση

```
select gr_comments: recreation_guide.#. [% lang=gr] comments:: [-]
```

είναι ισοδύναμη με την επερώτηση

```
select gr_comments: X
from recreation_guide.#. [% lang=gr] comments X
```

η οποία, όπως έχουμε δει στην Ενότητα 5.4.2, επιστρέφει όλες τις εκφάνσεις στα Ελληνικά του comments από την βάση δεδομένων:

```
{gr_comments: &10 "...",
 gr_comments: &41 "...",
 gr_comments: &36 "..."}
```

Όπως φαίνεται από τις δυο ισοδύναμες επερωτήσεις, αν μια έκφραση μονοπατιού περιβάλλοντος εμφανιστεί στον όρο select τότε μπορεί να συναχθεί ένας όρος from, ακριβώς όπως στην Lorel [AQM+97].

Μια έκφραση μονοπατιού περιβάλλοντος στο select πρέπει να τελειώνει με ένα τμήμα έκφρασης (:: [-] στην περίπτωση της παραπάνω επερώτησης) για να προκαλέσει αποτίμηση σε αντικείμενα περιβάλλοντος. Αν η έκφραση μονοπατιού περιβάλλοντος τελειώνει με ένα τμήμα οντότητας (. [% lang=gr] comments στην επερώτηση που ακολουθεί), τότε τα αποτελέσματα της επερώτησης θα περιέχουν πολυδιάστατα αντικείμενα, ωσάν να είχε χρησιμοποιηθεί μια μεταβλητή πολυδιάστατου αντικειμένου:

```
select comments: recreation_guide.#. [% lang=gr] comments
```

```
select comments: <X>
from recreation_guide.#. [% lang=gr] comments <X>
```

```
{comments: &100 <...>,
 comments: &27 <...>,
 comments: &35 <...>}
```

Το `oid &100` χρησιμοποιείται παραπάνω για να δηλώσει τον πολυδιάστατο κόμβο η μόνη έκφραση του οποίου είναι ο κόμβος `&10`, και που εισάγεται από τον μετασχηματισμό του γράφου στην Εικόνα 5.1 σε κανονική μορφή.

### Συμπερασμός των ετικετών ακμών

Οι ετικέτες των ακμών μπορούν να παραλειφθούν από τον οδηγό `mssd-expression` στο `select`. Στην περίπτωση αυτή, ανακύπτουν δυο ζητήματα: (a) πως να συνάγουμε στα αποτελέσματα τις ετικέτες ακμών οι οποίες λείπουν, και (b) πως να καθορίσουμε τον τύπο της ρίζας. Σχετικά με το ζήτημα (b), όπως έχουμε αναφέρει στην Ενότητα 5.4.3.1, αν δεν υπάρχουν αγκύλες ή άγκιστρα για να καθορίσουν τον τύπο της ρίζας, η MQL κοιτάζει τον τύπο της ετικέτας της πρώτης ακμής. Έτσι, δεν επιτρέπεται να παραλείπει κανείς αγκύλες ή άγκιστρα και ετικέτες ακμών την ίδια στιγμή, και αν έχουν παραλειφθεί οι ετικέτες θα πρέπει να υπάρχουν οι αγκύλες ή τα άγκιστρα. Συνεχίζουμε με το ζήτημα (a).

Οι ακόλουθες δυο επερωτήσεις είναι πολύ παρόμοιες με τις δυο προηγούμενες επερωτήσεις, αλλά οι ετικέτες των ακμών λείπουν από τον οδηγό `mssd-expression`. Η πρώτη επερωτηση επιστρέφει εκφάνσεις του `comments` στα Ελληνικά, ενώ η δεύτερη επιστρέφει πολυδιάστατα αντικείμενα `comments` που περιλαμβάνουν κάποια έκφραση στα Ελληνικά:

```
select {recreation_guide.#. [% lang=gr] comments:: [-] }
select {recreation_guide.#. [% lang=gr] comments }
```

Και στις δυο επερωτήσεις, οι εκφράσεις μονοπατιού περιβάλλοντος πρέπει να εγκλειστούν σε αγκύλες, για να δηλωθεί ότι η ρίζα του γράφου του αποτελέσματος είναι αντικείμενο περιβάλλοντος. Οι αγκύλες προκαλούν την δημιουργία ακμών *οντότητας*, που έχουν σαν ετικέτα αυτήν της τελευταίας ακμής οντότητας στο συνταιριάζον μονοπάτι δεδομένων περιβάλλοντος που οδηγεί στον αντίστοιχο κόμβο της βάσης δεδομένων. Στην συγκεκριμένη περίπτωση, αυτή η ακμή οντότητας έχει πάντα την ετικέτα `comments`, όμως αν μια έκφραση μονοπατιού περιβάλλοντος περιέχει μπαλαντέρ ή ομαλές εκφράσεις, τότε η ετικέτα στα αποτελέσματα θα είναι η ίδια με την ετικέτα στην βάση δεδομένων της ακμής εκείνης που οδηγεί στο αντίστοιχο αντικείμενο. Αυτό μπορεί να ιδωθεί σαν να έχει χρησιμοποιηθεί μια μεταβλητή ετικέτας:

```
select $L: OBJ_VAR
from cxt_path_expr$L OBJ_VAR
```

Αν μια ετικέτα ακμής οντότητας η οποία λείπει δεν είναι δυνατόν να συναχθεί<sup>91</sup>, τότε χρησιμοποιείται η εξ ορισμού ετικέτα `answer` κατά την κατασκευή του γράφου του αποτελέσματος.

Είναι δυνατόν η ρίζα των αποτελεσμάτων να είναι πολυδιάστατο αντικείμενο. Αυτό προκαλεί την δημιουργία ακμών περιβάλλοντος, με ετικέτα αυτήν της τελικής ακμής περιβάλλοντος στο συνταιριάζον μονοπάτι δεδομένων περιβάλλοντος που οδηγεί στον αντίστοιχο κόμβο της βάσης δεδομένων. Στην πράξη, αυτό σημαίνει ότι τα ρητά περιβάλλοντα «αντιγράφονται» στα αποτελέσματα μαζί με τις εκφάνσεις τις οποίες προσδιορίζουν στην βάση δεδομένων:

```
select <recreation_guide.#. [% lang=gr] comments:: [-] >
```

Η επερωτηση αυτή είναι ισοδύναμη με την:

<sup>91</sup> Για παράδειγμα, μια ετικέτα ακμής οντότητας δεν μπορεί να συναχθεί στην περίπτωση που είναι η ετικέτα ενός αντικειμένου ορισμένου απο τον χρήστη στον οδηγό του όρου `select`, το οποίο αντικείμενο μπορεί να περιέχει περισσότερες απο μια μεταβλητές.

```
select [Y]: X
from recreation_guide.#. [% lang=gr] comments:: [Y] X
```

Αποτιμώμενη στον γράφο στην Εικόνα 5.1 η παραπάνω επερώτηση επιστρέφει:

```
<[]: &10 "...",
 [lang=gr]: &41 "...",
 [detail=high,lang=gr]: &36 "...">
```

Προκειμένου να είμαστε σε θέση να συνάγουμε ετικέτες ακμών περιβάλλοντος η τελική ακμή του αντιστοίχου μονοπατιού δεδομένων περιβάλλοντος *θα πρέπει* να είναι ακμή περιβάλλοντος. Σε περίπτωση που δεν είναι δυνατόν να συναχθεί η ετικέτα μιας ακμής περιβάλλοντος, τότε χρησιμοποιείται η εξ ορισμού τιμή που είναι το κενό περιβάλλον [-], και που χρησιμοποιείται σαν το ρητό περιβάλλον των ακμών περιβάλλοντος στα αποτελέσματα.

Παρόλο που η ενότητα αυτή εισάγει την παράλειψη των ετικετών των ακμών κάτω από το πρόσχημα των εκφράσεων μονοπατιού περιβάλλοντος στο `select`, οι ετικέτες των ακμών μπορούν να παραλείπονται ανεξάρτητα από το εάν ή όχι υπάρχουν εκφράσεις μονοπατιού περιβάλλοντος στο `select`: οι κανόνες συμπερασμού των ετικετών που λείπουν είναι σε κάθε περίπτωση οι ίδιοι.

### **Ομαδοποίηση αποτελεσμάτων**

Όπως έχουμε δει σε προηγούμενες ενότητες, οι εμφωλιασμένες επερωτήσεις μπορούν να χρησιμοποιηθούν για να ομαδοποιήσουν τα αποτελέσματα. Ένας εναλλακτικός τρόπος ομαδοποίησης των αποτελεσμάτων είναι η χρήση εκφράσεων μονοπατιού περιβάλλοντος στον όρο `select` οι οποίες είναι «δεσμευμένες» σε έναν όρο `from`:

```
select club_comments:
      {gr_comments: <X>.#. [% lang=gr] comments:: [-]}
from recreation_guide.- <X>

{club_comments: {gr_comments: &10 "..."},
 club_comments: {gr_comments: &41 "...", gr_comments: &36 "..."} }
```

Η έκφραση μονοπατιού περιβάλλοντος στο `select` μπορεί να ερμηνευτεί σαν ένας δεύτερος όρος `from`, ο οποίος ανήκει σε μια εμφωλιασμένη επερώτηση που υπονοείται. Η πιο πάνω επερώτηση είναι ισοδύναμη με το πρώτο παράδειγμα εμφωλιασμένης επερώτησης στην Ενότητα 5.4.2, και επιστρέφει εκφάνσεις σχολίων στα Ελληνικά ομαδοποιημένες ανά μαγαζί (music club ή εστιατόριο).

Είναι δυνατόν να παραλείψουμε τις ετικέτες μπροστά από τις εκφράσεις μονοπατιού περιβάλλοντος, στην οποία περίπτωση αυτές συνάγονται με τον τρόπο που έχει περιγραφεί στην προηγούμενη ενότητα:

```
select club_comments: {<X>.#. [% lang=gr] comments:: [-]}
from recreation_guide.- <X>
```

Μια ισοδύναμη εμφωλιασμένη επερώτηση είναι η:

```
select club_comments: (select {Y}
                      from <X>.#. [% lang=gr] comments Y)
from recreation_guide.- <X>
```

Τα αποτελέσματα είναι τα ίδια όπως προηγουμένως, με την διαφορά ότι χρησιμοποιείται η συναγόμενη ετικέτα `comments` αντί της ετικέτας `gr_comments`. Παρατηρήστε ότι οι δυο ισοδύναμες επερωτήσεις πιο πάνω χρησιμοποιούν εκφράσεις μονοπατιού περιβάλλοντος που



καταλήγουν σε διαφορετικά τμήματα. Όταν χρησιμοποιούνται εκφράσεις μονοπατιού περιβάλλοντος στο `select`, δεν μπορούν να χρησιμοποιηθούν μεταβλητές αντικείμενου περιβάλλοντος για να προκαλέσουν αποτίμηση σε αντικείμενα περιβάλλοντος. Έτσι, οι εκφράσεις μονοπατιού περιβάλλοντος στο `select` πρέπει να καταλήγουν στο κατάλληλο τμήμα: τμήμα οντότητας για αποτίμηση σε πολυδιάστατα αντικείμενα, και τμήμα έκφρασης για αποτίμηση σε αντικείμενα περιβάλλοντος.

Μια έκφραση μονοπατιού περιβάλλοντος στο `select` δεν είναι πάντοτε δυνατόν να ιδωθεί σαν εμφωλιασμένη επερώτηση. Συγκρίνετε τις ακόλουθες δυο επερωτήσεις και τα αποτελέσματά τους.

```
select {{restaur_name: <X>.name::[-], winter_floor: Y}}
from recreation_guide.restaurant <X>,
      <X>.[season=winter]address.floor Y

{answer: {restaur_name: ...,
          restaur_name: ...,
          winter_floor: ...},
  answer: ...
}

select {{restaur_name: {<X>.name::[-]}, winter_floor: Y}}
from recreation_guide.restaurant <X>,
      <X>.[season=winter]address.floor Y

{answer: {restaur_name: {name: ...,
                        name: ...},
          winter_floor: ...},
  answer: ...
}
```

Οι επερωτήσεις επιστρέφουν για κάθε εστιατόριο τις εκφάνσεις του ονόματός του και τον όροφο στον οποίο λειτουργεί τον χειμώνα. Οι εξωτερικές αγκύλες υπονοούν μια ετικέτα ακμής οντότητας που λείπει (η οποία παίρνει την εξ ορισμού τιμή `answer`), και συγκρατούν μαζί τα ονόματα του εστιατορίου με το `winter_floor` κάθε εστιατορίου κάτω από ένα κοινό αντικείμενο πατέρα. Η μόνη διαφορά είναι ότι στην πρώτη επερώτηση η έκφραση μονοπατιού περιβάλλοντος `<X>.name::[-]` δεν περιβάλλεται από αγκύλες, ενώ στην δεύτερη επερώτηση περιβάλλεται από αγκύλες.

Στην πρώτη επερώτηση, η έκφραση μονοπατιού περιβάλλοντος δεν είναι ισοδύναμη με μια εμφωλιασμένη επερώτηση, με την έννοια ότι δεν επιστρέφει έναν γράφο με μια και μόνο ρίζα για κάθε δέσμευση του `<X>`. Αντί για αυτό, δυο ακμές `restaur_name` δείχνουν σε δυο εκφάνσεις ονόματος ενός εστιατορίου. Η δεύτερη επερώτηση περικλείει την έκφραση μονοπατιού περιβάλλοντος σε αγκύλες, πράγμα που μπορεί να ιδωθεί σαν να δηλώνει τον τύπο της ρίζας μιας ισοδύναμης εσωτερικής επερώτησης (αγκύλες για περιβάλλοντος, άγκιστρα για πολυδιάστατη). Σαν αποτέλεσμα, για κάθε εστιατόριο μια ακμή `restaur_name` δείχνει σε ένα αντικείμενο που μπορεί να θεωρηθεί η ρίζα μιας υπονοούμενης εσωτερικής επερώτησης, και που είναι ο πατέρας των εκφάνσεων του `name`.

Ένα σημαντικό σημείο είναι ότι μια έκφραση μονοπατιού περιβάλλοντος στο `select` αντιμετωπίζεται από την MQL πάντοτε με τον ίδιο τρόπο: το εάν ή όχι το ίδιο αποτέλεσμα θα μπορούσε να έχει επιτευχθεί με μια εσωτερική επερώτηση, εξαρτάται από την ομαδοποίηση των αγκύλων και των άγκιστρων στον οδηγό που βρίσκεται στον όρο `select`. Η σημασιολογία μιας έκφρασης μονοπατιού περιβάλλοντος στο `select` εξηγείται ακολούθως:

```
select ... label: X.cpe2
from cpe1 X
```

```
select ... label: Yx
from cpe1 X, X.cpe2 Yx
```

Η μεταβλητή  $Y_x$  αποτιμάται σε ένα σύνολο για κάθε τιμή του  $x$ . Αν το `label` δεν είναι προσδεδεμένο στην ρίζα των αποτελεσμάτων, η MQL δημιουργεί έναν κόμβο για κάθε τιμή του  $x$ . Στην συνέχεια, ακμές `label` ξεκινούν από κάθε τέτοιο κόμβο, και οδηγούν στις τιμές του αντίστοιχου  $Y_x$ . Από την άλλη, αν το `label` είναι προσδεδεμένο στην ρίζα των αποτελεσμάτων, δεν είναι δυνατόν να δημιουργηθούν ξεχωριστοί κόμβοι για τις τιμές του  $x$  και οι τιμές όλων των συνόλων που είναι δεσμευμένα στο  $Y_x$  ομαδοποιούνται κάτω από την ρίζα.

### 5.4.3.3 Αναγωγή του αποτελέσματος γράφου

Ανεξάρτητα από το εάν ή όχι η βάση δεδομένων είναι MOEM, ο γράφος που επιστρέφεται από μια επρώτηση MQL είναι δυνατόν να περιέχει κόμβους και ακμές που δεν υφίστανται κάτω από κανέναν κόσμο. Συχνά επιθυμούμε να εξασφαλίσουμε ότι κάθε κόμβος και ακμή στα αποτελέσματα υφίσταται κάτω από έναν τουλάχιστον κόμβο. Η λέξη-κλειδί `holding` ακολουθεί την λέξη-κλειδί `select` στον όρο `select`, και προκαλεί την απομάκρυνση όλων των κόμβων και των ακμών που έχουν κενή κληρονομούμενη κάλυψη. Όπως δείξαμε στο προηγούμενο κεφάλαιο, το αποτέλεσμα αυτής της διεργασίας είναι ένας Πολυδιάστατος Γράφος Δεδομένων τα φύλλα του οποίου είναι αποκλειστικά ατομικοί κόμβοι, και ο οποίος περιέχει κόμβους και ακμές που «επιβιώνουν» σαν μέρος ενός συμβατικού OEM που υφίσταται κάτω από κάποιον κόσμο. Στην περίπτωση που ο γράφος του αποτελέσματος μιας επρώτησης MQL είναι αιτιοκρατικός ως προς το περιβάλλον, η χρήση του `holding` εξασφαλίζει ότι η επρώτηση θα επιστρέψει ένα MOEM.

Η διεργασία αυτή είναι η τελευταία πράξη που εκτελείται πριν την επιστροφή των αποτελεσμάτων μιας επρώτησης, και εφαρμόζεται σε ολόκληρο τον γράφο που δημιουργείται με βάση τον οδηγό `mssd-expression` στο `select`, αφού οι μεταβλητές του οδηγού έχουν αντικατασταθεί από τις δεσμευμένες τιμές και οι εσωτερικές επερωτήσεις έχουν αποτιμηθεί. Προκειμένου να απομακρυνθούν οι κόμβοι και οι ακμές που δεν υφίστανται κάτω από κανέναν κόσμο, θα πρέπει να υπολογιστούν οι κληρονομούμενες καλύψεις του γράφου του αποτελέσματος.

Στην πραγματικότητα, το `holding` προκαλεί αυτό που περιγράψαμε στο προηγούμενο κεφάλαιο σαν **περιβαλλοντική περικοπή γράφου**. Όπως είδαμε στο προηγούμενο κεφάλαιο, ο συνδυασμός της **περιβαλλοντικής προβολής γράφου** σε ένα περιβάλλον `c` μαζί με την περιβαλλοντική περικοπή γράφου, είναι ισοδύναμος με την **μερική αναγωγή** για το περιβάλλον `c`. Θεωρήστε την επρώτηση:

```
select holding guide_facets:
      <[detail=low]: <X>, [detail=high]: <X>>
from recreation_guide <X>
```

Η μεταβλητή `<X>` δεσμεύεται στην ρίζα της βάσης δεδομένων, και στον οδηγό δείχνεται από δυο ακμές περιβάλλοντος με ρητά περιβάλλοντα `[detail=low]` και `[detail=high]`. Αυτές οι ακμές περιβάλλοντος στην πράξη προβάλλουν την βάση δεδομένων στα αντίστοιχα περιβάλλοντά τους: τότε η λέξη-κλειδί `holding` προκαλεί την απομάκρυνση των κόμβων και των ακμών με κενή κληρονομούμενη κάλυψη. Το αποτέλεσμα είναι ένας γράφος που περιλαμβάνει δυο μερικές αναγωγές της βάσης δεδομένων, μια για το περιβάλλον `[detail=low]` και μια άλλη για το περιβάλλον `[detail=high]`.

Η ακόλουθη επερώτηση επιστρέφει έναν γράφο που έχει την εξής δομή: η ρίζα είναι πολυδιάστατος κόμβος, και οι εξερχόμενες ακμές περιβάλλοντος οδηγούν σε όλες τις δυνατές εκφάνσεις της βάσης δεδομένων που υφίστανται κάτω από κάποιον κόσμο.

```
select holding <[Z]: <X>>
context [Z] as extension([Y])
from [Y]recreation_guide <X>
```

Η μεταβλητή <x> δεσμεύεται στην ρίζα της βάσης δεδομένων, και η μεταβλητή περιβάλλοντος [Y] δεσμεύεται στην κληρονομούμενη κάλυψη της ρίζας. Θυμηθείτε ότι η κληρονομούμενη κάλυψη της ρίζας αναπαριστά ακριβώς τους κόσμους κάτω από τους οποίους ο γράφος μπορεί να αναχθεί σε OEMs. Η μεταβλητή περιβάλλοντος [Z] δεσμεύεται στους κόσμους αυτούς μέσω της συνάρτησης `extension([c])`, η οποία, όπως αναφέρθηκε στην Ενότητα 5.3.5, μετατρέπει ένα περιβάλλον στους κόσμους που αυτό αναπαριστά. Το αποτέλεσμα της χρήσης της μεταβλητής περιβάλλοντος [Z] στο `select` είναι η δημιουργία μιας νέας ακμής περιβάλλοντος για κάθε κόσμο κάτω από τον οποίο η βάση δεδομένων μπορεί να αναχθεί σε OEM<sup>92</sup>. Οι ακμές περιβάλλοντος έχουν ρητά περιβάλλοντα που αντιστοιχούν στους κόσμους αυτούς, προβάλλοντας έτσι την βάση δεδομένων ξεχωριστά σε κάθε έναν από αυτούς τους κόσμους.

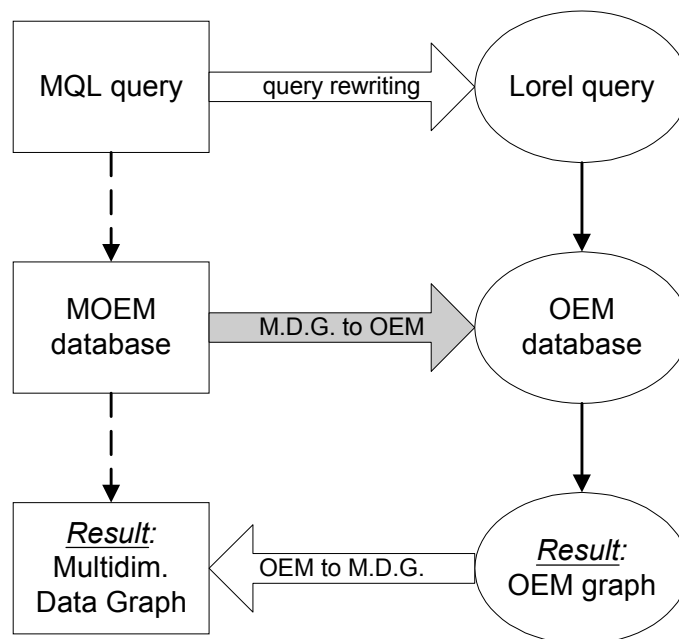
Από το προηγούμενο κεφάλαιο, η **αναγωγή σε OEM** κάτω από έναν κόσμο  $w$  είναι ισοδύναμη με την περιβαλλοντική προβολή γράφου σε ένα περιβάλλον που αναπαριστά μόνον το  $w$ , την περιβαλλοντική περικοπή γράφου, και την **απο-περιβαλλοντοποίηση γράφου**. Συνεπώς, το αποτέλεσμα της παραπάνω επερώτησης είναι ένας γράφος που περιλαμβάνει όλες τις ξεχωριστές εκφάνσεις OEM της βάσης δεδομένων, με την διαφορά ότι αυτές δεν έχουν απο-περιβαλλοντοποιηθεί. Αυτό σημαίνει ότι οι πολυδιάστατοι κόμβοι και οι ακμές περιβάλλοντος, παρόλο που δεν έχουν πια κανένα νόημα, δεν έχουν απομακρυνθεί, αφήνοντας τις εκφάνσεις OEM σε μια MOEM μορφή.

## 5.5 ΥΛΟΠΟΙΗΣΗ ΠΑΝΩ ΣΤΟ LORE

Ανάλογα με την Lorel, η οποία έχει υλοποιηθεί πάνω σε μια αντικειμενοστρεφή βάση δεδομένων [AQM+97], υλοποιήσαμε την MQL πάνω στο LORE [MAG+97]. Η πρότυπη υλοποίησή μας της MQL αποτελείται από μια εφαρμογή που αρχικοποιείται με μια βάση δεδομένων MOEM, δέχεται επερωτήσεις MQL, και επιστρέφει Πολυδιάστατους Γράφους Δεδομένων που είναι τα αποτελέσματα των επερωτήσεων αυτών. Η εφαρμογή διασυνδέεται με το LORE, που χρησιμοποιείται σαν «back-end». Επιλέξαμε το LORE σαν βάση για την υλοποίηση της MQL, επειδή είναι ενδιαφέρον να δει κανείς: (a) πως μια επερώτηση MQL συγκρίνεται με μια «ισοδύναμη» επερώτηση Lorel, και (b) πως ένα MOEM μπορεί να εκφραστεί μέσω ενός συμβατικού OEM.

<sup>92</sup> Για να έχει το αποτέλεσμα αυτό η επερώτηση, η κληρονομούμενη κάλυψη της ρίζας δεν πρέπει να περιέχει τον καθολικό όρο  $\emptyset$ . Όπως έχει εξηγηθεί, η **ανάπτυξη περιβάλλοντος** ενός περιβάλλοντος [c] δεν είναι ορισμένη αν το [c] περιέχει τον καθολικό όρο  $\emptyset$ , και η `extension([c])` επιστρέφει *nil*. Σε αυτή την περίπτωση δεν δημιουργούνται ακμές περιβάλλοντος, και το αποτέλεσμα προτού εφαρμοστεί το `holding` αποτελείται μόνον από έναν πολυδιάστατο κόμβο της ρίζας.

**Εικόνα 5.7:** Αποτίμηση επερωτήσεων MQL με χρήση του συστήματος LORE σαν υπόβαθρο.



Η συνολική αρχιτεκτονική φαίνεται στην Εικόνα 5.7. Η διεργασία που θέλουμε να υλοποιήσουμε απεικονίζεται με μια διακεκομμένη γραμμή, που αρχίζει από μια επερώτηση MQL, περνάει από μια βάση δεδομένων MOEM, και καταλήγει σε έναν Πολυδιάστατο Γράφο Δεδομένων που είναι το αποτέλεσμα της επερώτησης.

Η διεργασία που λαμβάνει χώρα στην πραγματικότητα απεικονίζεται σαν μια κανονική γραμμή, και δείχνει μια επερώτηση Lorel που αποτιμάται πάνω σε μια βάση δεδομένων OEM και επιστρέφει έναν γράφο OEM σαν αποτέλεσμα. Η γραμμή αυτή μαζί με τα ελλειψοειδή κουτιά αποτελούν μέρος του LORE, το οποίο καθοδηγείται από την εφαρμογή μας μέσω της προγραμματιστικής διεπαφής που παρέχει.

Το κύριο ζήτημα είναι να ορίσουμε έναν μετασχηματισμό  $T$  από Πολυδιάστατους Γράφους Δεδομένων  $M$  σε OEMs  $O = T(M)$ , με τις ακόλουθες ιδιότητες: (a) υπάρχει ο αντίστροφος μετασχηματισμός  $T^{-1}$ , και αν δίδεται το  $O$  τότε το  $M = T^{-1}(O)$  είναι δυνατόν να ανακτηθεί, και (b) είναι δυνατόν να μεταφράσουμε μια επερώτηση MQL  $q_M$  σε μια «ισοδύναμη» επερώτηση Lorel  $q_L$ . Με τον όρο ισοδύναμη εννοούμε ότι αν η  $q_M$  αποτιμώμενη στο  $M$  επιστρέφει  $M'$  και η  $q_L$  αποτιμώμενη στο  $O$  επιστρέφει  $O'$ , τότε  $T(M') = O'$ . Τότε, η απάντηση στην  $q_M$  μπορεί να υπολογιστεί αποτιμώντας την  $q_L$  στο  $T(M)$ , και εφαρμόζοντας τον αντίστροφο μετασχηματισμό  $T^{-1}(O')$  στα αποτελέσματα της  $q_L$ .

Οι μετασχηματισμοί αυτοί και η μετάφραση μιας επερώτησης MQL απεικονίζονται στην Εικόνα 5.7 σαν παχιά οριζόντια βέλη. Η εφαρμογή, μεταξύ άλλων, υλοποιεί τα βέλη αυτά και εκτελεί τα ακόλουθα σημαντικά βήματα:

1. Μετατρέπει μια βάση δεδομένων MOEM σε μια βάση δεδομένων OEM, που γίνεται η βάση δεδομένων του LORE.
2. Μεταφράζει μια επερώτηση MQL σε μια επερώτηση Lorel, που αποστέλλεται στο LORE προκειμένου να αποτιμηθεί πάνω στην βάση δεδομένων OEM.

3. Παίρνει τα αποτελέσματα από το LORE, και τα μετατρέπει από OEM πίσω σε Πολυδιάστατο Γράφο Δεδομένων.

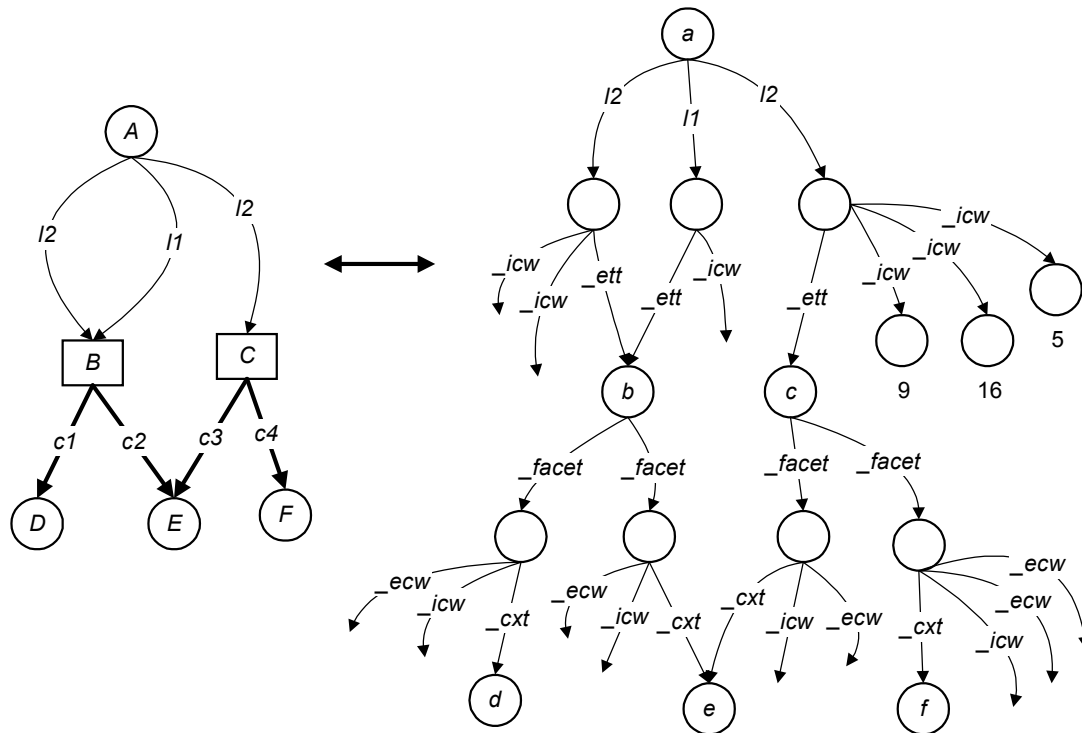
Ο Πολυδιάστατος Γράφος Δεδομένων του βήματος 3 είναι το αποτέλεσμα της επερώτησης MQL του βήματος 2 αποτιμώμενης πάνω στην βάση δεδομένων MOEM του βήματος 1. Το βήμα 1 αρχικοποιεί την βάση δεδομένων και αντιστοιχεί στο γκρι οριζόντιο βέλος στην Εικόνα 5.7, ενώ τα βήματα 2 και 3 εκτελούνται κάθε φορά που υποβάλλεται μια επερώτηση MQL.

Στα ακόλουθα, εστιάζουμε στο να προδιαγράψουμε τα τρία σημαντικά βήματα που αναφέραμε παραπάνω. Η τελική εφαρμογή που τα υλοποιεί και που αποτιμά επερωτήσεις MQL είναι μέρος ενός πιο εκτεταμένου συστήματος για MSSD, το οποίο παρουσιάζεται στο Κεφάλαιο 8.

### 5.5.1 Μετασηματισμός βάσεων MOEM σε OEM

Προκειμένου να μετασηματίσουμε ένα MOEM σε OEM, πρέπει να χρησιμοποιήσουμε στοιχεία του OEM για να αναπαραστήσουμε τα στοιχεία του MOEM που δεν έχουν κάποιο ανάλογο στο OEM, δηλαδή τις ακμές περιβάλλοντος και τους πολυδιάστατους κόμβους: οι ακμές περιβάλλοντος μπορούν να αναπαρασταθούν από ακμές του OEM που έχουν κάποια ειδική ετικέτα, ενώ οι πολυδιάστατοι κόμβοι αντιστοιχούν σε κόμβους του OEM από τους οποίους ξεκινούν οι ειδικές αυτές ακμές. Θα πρέπει επίσης να βρούμε έναν τρόπο για να κωδικοποιήσουμε το ερμηνευτικό περιβάλλον στο OEM. Το περιβάλλον δεν μπορεί απλά να εμφανίζεται σαν στοιχειοσειρά η τιμή της οποίας είναι ένας προσδιοριστής περιβάλλοντος. Ο λόγος είναι ότι η Lorel δεν καταλαβαίνει την έννοια του ερμηνευτικού περιβάλλοντος, συνεπώς αν οι οδηγούμενες από το περιβάλλον επερωτήσεις πρόκειται να μεταφραστούν σε επερωτήσεις Lorel, το περιβάλλον πρέπει να κωδικοποιηθεί με τρόπο που η Lorel να καταλαβαίνει και να μπορεί να χειριστεί. Στα περιβάλλοντα που πρέπει να κωδικοποιηθούν συμπεριλαμβάνονται τα ρητά περιβάλλοντα, αλλά και οι κληρονομούμενες καλύψεις των ακμών, οι οποίες παίζουν σημαντικό ρόλο στις εκφράσεις μονοπατιού περιβάλλοντος και στις επερωτήσεις MQL.

Εικόνα 5.8: Αναπαράσταση Πολυδιάστατου Γράφου Δεδομένων με χρήση OEM.



Η Εικόνα 5.8 δίνει μια διαίσθηση για τον μετασχηματισμό μας. Παρουσιάζει έναν απλό Πολυδιάστατο Γράφο Δεδομένων  $M$  και το αντίστοιχό του OEM  $O$ . Οι κόμβοι με κεφαλαίο γράμμα στο  $M$  αντιστοιχούν σε κόμβους με το ανάλογο μικρό γράμμα στο  $O$ . Παρατηρείστε ότι για κάθε ακμή στο  $M$  υπάρχει ένας επιπλέον κόμβος στο  $O$ , που χωρίζει την ακμή σε δυο ακμές του OEM. Ο ρόλος αυτού του κόμβου είναι να ομαδοποιήσει το κωδικοποιημένο περιβάλλον της αντίστοιχης ακμής του Πολυδιάστατου Γράφου Δεδομένων. Χρησιμοποιείται επίσης και ένας αριθμός δεσμευμένων ετικετών με ειδική σημασία. Όλες οι δεσμευμένες ετικέτες αρχίζουν με τον χαρακτήρα «\_», και είναι οι: `_ett`, `_facet`, `_cxt`, `_icw`, και `_ecw`. Μια ακμή οντότητας αναπαρίσταται από μια ακμή με την ίδια ετικέτα και μια ακόλουθη ακμή με ετικέτα `_ett`. Μια ακμή περιβάλλοντος αναπαρίσταται από μια ακμή με ετικέτα `_facet` και μια ακόλουθη ακμή με ετικέτα `_cxt`. Τα ρητά περιβάλλοντα και οι κληρονομούμενες καλύψεις των ακμών μετατρέπονται στους κόσμους που αναπαριστούν, και όλοι οι πιθανοί κόσμοι αντιστοιχούνται σε ακεραίους<sup>93</sup>. Οι ακμές με ετικέτα `_icw` δείχνουν σε αριθμημένους κόσμους (ακεραίους) που ανήκουν σε κληρονομούμενες καλύψεις, ενώ οι ακμές με ετικέτα `_ecw` δείχνουν σε αριθμημένους κόσμους (ακεραίους) που ανήκουν σε ρητά περιβάλλοντα.

Η διεργασία μετασχηματισμού ενός Πολυδιάστατου Γράφου Δεδομένων  $M = (V_{mld}, V_{cxt}, E_{cxt}, E_{ett}, r, v)$  σε ένα OEM  $O$  δίνεται παρακάτω. Η διεργασία θεωρεί ότι υπάρχει μια αντιστοιχία όλων των πιθανών κόσμων σε ακεραίους αριθμούς.

<sup>93</sup> Συνεπώς, αυτή η κωδικοποίηση του ερμηνευτικού περιβάλλοντος απαιτεί το σύνολο των διαστάσεων  $D$  να είναι γνωστό εκ των προτέρων, σε αντίθεση με το MOEM και την MQL που δεν χρησιμοποιούν το  $D$  προκειμένου να αναπαραστήσουν και να επερωτήσουν MSSD.

**O ← MDGToOEM (M) is:**

1. Για κάθε κόσμο, προσθέτουμε έναν νέο ατομικό κόμβο  $w$  στο  $V_{\text{cxt}}$  που αντιστοιχεί στον κόσμο αυτό, και που έχει σαν τιμή τον αντίστοιχο ακέραιο.
2. Μετακινούμε όλους τους κόμβους από το  $V_{\text{mld}}$  στο  $V_{\text{cxt}}$  σαν σύνθετους κόμβους.
3. Για κάθε ακμή  $h = (q, l, p) \in E_{\text{ett}}$  προσθέτουμε έναν νέο σύνθετο κόμβο  $u$  στο  $V_{\text{cxt}}$ . Στην συνέχεια, αντικαθιστούμε το  $h$  με τις νέες ακμές  $(q, l, u)$  και  $(u, \_ett, p)$  στο  $E_{\text{ett}}$ . Έπειτα, για κάθε κόσμο που ανήκει στην κληρονομούμενη κάλυψη του  $h$ , προσθέτουμε μια ακμή  $(u, \_icw, w)$  στο  $E_{\text{ett}}$ , όπου το  $w$  αντιστοιχεί στον κόσμο αυτό.
4. Για κάθε ακμή  $h = (q, c, p) \in E_{\text{cxt}}$  προσθέτουμε έναν νέο σύνθετο κόμβο  $u$  στο  $V_{\text{cxt}}$ . Στην συνέχεια, απομακρύνουμε το  $h$  από το  $E_{\text{cxt}}$ , και προσθέτουμε τις ακμές  $(q, \_facet, u)$  και  $(u, \_cxt, p)$  στο  $E_{\text{ett}}$ . Έπειτα, για κάθε κόσμο που ανήκει στην κληρονομούμενη κάλυψη του  $h$ , προσθέτουμε μια ακμή  $(u, \_icw, w)$  στο  $E_{\text{ett}}$ , όπου το  $w$  αντιστοιχεί στον κόσμο αυτό. Επιπλέον, για κάθε κόσμο που ανήκει στο ρητό περιβάλλον του  $h$ , προσθέτουμε μια ακμή  $(u, \_ecw, w)$  στο  $E_{\text{ett}}$ , όπου το  $w$  αντιστοιχεί στον κόσμο αυτό.
5. Επιστρέφουμε το  $O = (V_{\text{cxt}}, E_{\text{ett}}, r, v)$ .

Μερικοί ατομικοί κόμβοι που αντιστοιχούν σε κόμβους είναι δυνατόν να μην δείχνονται από καμία ακμή, και να μην είναι προσβάσιμοι από την ρίζα. Αυτό μπορεί να αποφευχθεί εύκολα με το να προσθέσουμε αυτούς τους κόμβους στο  $V_{\text{cxt}}$  την στιγμή που πρόκειται να χρησιμοποιηθούν στα βήματα 3 και 4 αντί για το βήμα 1. Εναλλακτικά, οι μη προσβάσιμοι κόμβοι μπορούν να απομακρυνθούν στο τέλος της διεργασίας. Αφήνουμε το ζήτημα αυτό στην υλοποίηση. Επιπλέον, η συνάρτηση  $v$  θα πρέπει να ενημερωθεί ώστε να περικλείει και τις τιμές για τους νέους κόμβους.

Προσέξτε ότι, αν ο Πολυδιάστατος Γράφος Δεδομένων έχει σύνθετους κόμβους ή πολυδιάστατους κόμβους σαν φύλλα, τότε το OEM του αποτελέσματος θα έχει και αυτό σύνθετους κόμβους σαν φύλλα. Στην πράξη, στο βήμα 2 θα μπορούσαμε να αναπαραστήσουμε τα σύνθετα φύλλα του Πολυδιάστατου Γράφου Δεδομένων σαν ατομικούς κόμβους του OEM που έχουν την δεσμευμένη τιμή «\_C», και τα πολυδιάστατα φύλλα σαν ατομικούς κόμβους του OEM που έχουν την δεσμευμένη τιμή «\_M». Αυτό θα διασφάλιζε ότι οποιοσδήποτε Πολυδιάστατος Γράφος Δεδομένων μπορεί να μετασχηματιστεί σε ένα έγκυρο OEM, και ότι στον αντίστροφο μετασχηματισμό θα είναι δυνατόν να αναγνωρίσουμε τα φύλλα σαν τέτοια. Πάντως, κυρίως ενδιαφερόμαστε να μετασχηματίσουμε βάσεις δεδομένων MOEM, όπου τα φύλλα είναι αποκλειστικά ατομικοί κόμβοι.

## 5.5.2 Μετάφραση επερωτήσεων MQL σε Lorel

Οι επερωτήσεις MQL μπορούν να μεταφραστούν σε «ισοδύναμες» επερωτήσεις Lorel, οι οποίες αποτιμώνται στο OEM που δίνει ο μετασχηματισμός που ορίζεται στην προηγούμενη ενότητα. Για να γίνει σωστά η μετάφραση αυτή, η βάση δεδομένων MOEM *πρέπει να βρίσκεται σε κανονική μορφή* όταν λαμβάνει χώρα ο μετασχηματισμός σε OEM. Αυτό επιτρέπει στις εκφράσεις μονοπατιού περιβάλλοντος, οι οποίες είναι χτισμένες γύρω από την κανονική μορφή, να μεταφραστούν σε «ισοδύναμες» συμβατικές εκφράσεις μονοπατιού.

Στην ενότητα αυτή ορίζουμε μια τέτοια μετάφραση που υποστηρίζει τα κυριότερα χαρακτηριστικά της MQL, έτσι ώστε να πάρουμε μια εντύπωση για το πως μοιάζει μια ισοδύναμη επερωτήση σε Lorel. Δεν αντιμετωπίζουμε κάποια χαρακτηριστικά της MQL που

φάνηκαν δύσκολο ή αδύνατο να μεταφραστούν, όπως οι ομαλές εκφράσεις σε εκφράσεις μονοπατιού περιβάλλοντος (γενικευμένες εκφράσεις μονοπατιού περιβάλλοντος). Επίσης αφήσαμε έξω από την υλοποίηση χαρακτηριστικά της MQL που δεν φάνηκαν κρίσιμα για τον στόχο μας, όπως για παράδειγμα ο όρος `context` και τα πρότυπα περιβάλλοντος.

Επιπλέον, δεν ορίζουμε κάποια μετάφραση για τον όρο `select` της MQL, επειδή οι όροι `select` στην MQL και στην Lorel είναι σημασιολογικά πολύ παρόμοιοι. Παρόλη την ομοιότητά τους, οι όροι `select` των δυο γλωσσών διαφέρουν στον ορισμό του γράφου από τον οποίο κρέμονται δεσμεύσεις μεταβλητών. Συνεπώς, δεδομένου ότι ο όρος `select` της MQL δεν μεταφράζεται, ο επιστρεφόμενος γράφος θα είναι κατασκευασμένος σύμφωνα με την ερμηνεία της Lorel για τον όρο `select` της MQL. Αυτό, πάντως, δεν είναι απαγορευτικό για απλές επερωτήσεις επιδείξεως.

### 5.5.2.1 Από εκφράσεις μονοπατιού περιβάλλοντος σε εκφράσεις μονοπατιού

Για να διευκολύνουμε την κατανόηση των μεταφρασμένων επερωτήσεων Lorel, χρησιμοποιούμε το E-HUB σαν αναγνωριστικό για κόμβους από τους οποίους ξεκινά μια ακμή `_ett`, το MLD για κόμβους από τους οποίους ξεκινά μια ακμή `_facet`, και το C-HUB για κόμβους από τους οποίους ξεκινά μια ακμή `_cxt`. Επιπρόσθετα, χρησιμοποιούμε τα  $w_1, w_2, \dots$  για να δηλώσουμε τους ακέραιους που έχουν αντιστοιχηθεί σε κόσμους.

Αρχίζουμε με τον πολύ απλό όρο `from` της MQL:

```
from X. [c] label Y
```

Θεωρούμε ότι το `[c]` είναι ένας προσδιοριστής περιβάλλοντος που αναπαριστά τους κόσμους που αντιστοιχούν στα  $w_4, w_7$ , και  $w_9$ . Από την Πρόταση 5.1 και επειδή το `[c]` είναι προσδιοριστής περιβάλλοντος, μπορεί να επαναληφθεί κατά μήκος του μονοπατιού που αφορά, και ο όρος μπορεί να γραφτεί σαν:

```
from X. [c] label :: [c] [-] Y
```

Αυτός ο όρος `from` μπορεί επίσης να γραφτεί στην MQL σαν:

```
from X. [c] label <V>,
      <V> :: [c] [-] Y
```

Η ισοδύναμη έκφραση Lorel είναι:

```
from X.label E-HUB,
      E-HUB._ett MLD,
      MLD._facet C-HUB,
      C-HUB._cxt Y
where   E-HUB._icw{W4} = w4
        and E-HUB._icw{W7} = w7
        and E-HUB._icw{W9} = w9
        and C-HUB._icw{W4} = w4
        and C-HUB._icw{W7} = w7
        and C-HUB._icw{W9} = w9
```

Ο όρος `where` δηλώνει ότι οι κληρονομούμενες καλύψεις των δυο ακμών του MOEM πρέπει να περιέχουν όλους τους κόσμους που ορίζονται από το `[c]`. Παρατηρήστε την χρήση των μεταβλητών  $w_4, w_7$ , και  $w_9$ , που δηλώνουν ότι δεν πρόκειται για τον *ίδιο* κόμβο που



πρέπει να είναι ίσος με  $w_4$ , με  $w_7$ , και με  $w_9$  (διαφορετικά η συνθήκη θα ήταν πάντοτε ψευδής)<sup>94</sup>.

Θεωρούμε τώρα μια ολοκληρωμένη επερώτηση MQL:

```
select restaurant: {name: P, winter_floor: Y}
from recreation_guide.restaurant X,
     X.[season=winter]address.floor Y,
     X.[season=summer, daytime=noon]address.floor Z
     X.name P
where Z="terrace"
```

Για συντομία, χρησιμοποιούμε το [c1] για να υποδηλώσουμε τον προσδιοριστή περιβάλλοντος [season=winter], και το [c2] για να υποδηλώσουμε τον [season=summer, daytime=noon]. Η επερώτηση MQL μπορεί τώρα να γραφτεί σαν:

```
select restaurant: {name: P, winter_floor: Y}
from [-]recreation_guide <V1>, <V1>::[-] [-] V2,
     V2.[-]restaurant <V3>, <V3>::[-] [-] X,
     X.[c1]address <V4>, <V4>::[c1] [-] V5,
     V5.[c1]floor <V6>, <V6>::[c1] [-] Y,
     X.[c2]address <V7>, <V7>::[c2] [-] V8,
     V8.[c2]floor <V9>, <V9>::[c2] [-] Z,
     X.[-]name <V10>, <V10>::[-] [-] P
where Z="terrace"
```

Η ισοδύναμη επερώτηση Lorel είναι:

```
select restaurant: {name: P, winter_floor: Y}
from
     recreation_guide E-HUB1, E-HUB1._ett V1,
     V1._facet C-HUB1, C-HUB1._cxt V2,

     V2.restaurant E-HUB2, E-HUB2._ett V3,
     V3._facet C-HUB2, C-HUB2._cxt X,

     X.address E-HUB3, E-HUB3._ett V4,
     V4._facet C-HUB3, C-HUB3._cxt V5,

     V5.floor E-HUB4, E-HUB4._ett V6,
     V6._facet C-HUB4, C-HUB4._cxt Y,

     X.address E-HUB5, E-HUB5._ett V7,
     V7._facet C-HUB5, C-HUB5._cxt V8,

     V8.floor E-HUB6, E-HUB6._ett V9,
     V9._facet C-HUB6, C-HUB6._cxt Z,

     X.name E-HUB7, E-HUB7._ett V10,
     V10._facet C-HUB7, C-HUB7._cxt P
where
     Z="terrace"
     and predicate(E-HUB3)
     and predicate(C-HUB3)
     and predicate(E-HUB4)
     and predicate(C-HUB4)
```

<sup>94</sup> Όταν χρησιμοποιούνται μεταβλητές με αυτόν τον τρόπο, η Lorel υπονοεί την συνθήκη: and  $w_4 <> w_7$  and  $w_4 <> w_9$  and  $w_7 <> w_9$

```

and predicate(E-HUB5)
and predicate(C-HUB5)
and predicate(E-HUB6)
and predicate(C-HUB6)

```

Οι εκφράσεις  $predicate(VAR)$  διασφαλίζουν ότι οι αντίστοιχες ακμές έχουν την κατάλληλη κληρονομούμενη κάλυψη. Έτσι, κάθε έκφραση  $predicate(VAR)$  θα πρέπει να αντικατασταθεί με

```

VAR._icw{W1} = w1
and VAR._icw{W2} = w2
and VAR._icw{W3} = w3
and ...

```

όπου τα  $w_1, w_2, w_3, \dots$  είναι οι ακέραιοι που αντιστοιχούν σε κόσμους του ανάλογου περιοριστή κληρονομούμενης κάλυψης: για τα E-HUB3, C-HUB3, E-HUB4, και C-HUB4 ο περιοριστής κληρονομούμενης κάλυψης είναι [season=winter], ενώ για τα E-HUB5, C-HUB5, E-HUB6, και C-HUB6 ο περιοριστής κληρονομούμενης κάλυψης είναι [season=summer, daytime=noon]. Οι ακμές που αντιστοιχούν στις μεταβλητές E-HUB1, C-HUB1, E-HUB2, C-HUB2, E-HUB7, και C-HUB7 μπορούν να έχουν οποιαδήποτε κληρονομούμενη κάλυψη επειδή ο περιοριστής κληρονομούμενης κάλυψής τους είναι το κενό περιβάλλον [-], και έτσι δεν συμπεριλαμβάνονται στο where.

Χρησιμοποιώντας το παραπάνω πλαίσιο, είναι απλό να μεταφράσουμε επερωτήσεις MQL που περιέχουν μεταβλητές πολυδιάστατου αντικειμένου. Στην πραγματικότητα, η αναλυτική μορφή του παραδείγματός μας της επερώτησης MQL περιέχει τις μεταβλητές πολυδιάστατου αντικειμένου <V1>, <V3>, <V4>, <V6>, <V7>, <V9>, και <V10>, οι οποίες αντιστοιχούν στις μεταβλητές V1, V3, V4, V6, V7, V9, και V10 της ισοδύναμης επερώτησης Lorel. Επιπλέον, είναι εύκολο να υποστηρίξουμε περιοριστές ρητού περιβάλλοντος. Ένα τμήμα έκφρασης :: [C<sub>I</sub>] [C<sub>E</sub>] θα δώσει ένα κατηγορημα (predicate) της μορφής:

```

VAR._icw{W1} = w1
and VAR._icw{W2} = w2
and VAR._icw{W3} = w3
and ...
and VAR._ecw{W2} = w2
and VAR._ecw{W6} = w6
and ...

```

όπου τα  $w_1, w_2, w_3, \dots$  αντιστοιχούν στους κόσμους του περιοριστή κληρονομούμενης κάλυψης [C<sub>I</sub>], και τα  $w_2, w_6, \dots$  αντιστοιχούν στους κόσμους του περιοριστή ρητού περιβάλλοντος [C<sub>E</sub>].

Παρότι είναι δυνατόν να μεταφράσουμε τον μπαλαντέρ – της MQL στον μπαλαντέρ % της Lorel, δεν είναι πάντοτε δυνατόν να μεταφράσουμε επερωτήσεις MQL που περιέχουν τον μπαλαντέρ #. Συγκεκριμένα, αν ο περιοριστής κληρονομούμενης κάλυψης ενός μονοπατιού δεν είναι το κενό περιβάλλον, δεν είναι δυνατόν να μεταφράσουμε επερωτήσεις MQL με ομαλές εκφράσεις που χρησιμοποιούν το + ή το \*. Ο λόγος είναι ότι δεν μπορούμε να εισάγουμε μεταβλητές για ολόκληρο το (άγνωστο) μονοπάτι ώστε να εκφράσουμε συνθήκες για τις αντίστοιχες κληρονομούμενες καλύψεις στο where.

### 5.5.2.2 Μεταβλητές περιβάλλοντος και ο όρος “within”

Θεωρήστε την επερώτηση MQL

```
select comments: Y
from recreation_guide.restaurant.[X]review.comments Y
within [X] * [detail=high] <= [lang=gr]
```

η οποία χρησιμοποιεί μια μεταβλητή περιβάλλοντος για να βρει τις εκφάνσεις των σχολίων (comments) στα Ελληνικά σε υψηλή λεπτομέρεια<sup>95</sup>. Το πρώτο βήμα είναι να εκφράσουμε τους προσδιοριστές περιβάλλοντος σαν επερωτήσεις Lorel. Υποθέστε ότι το [detail=high] αναπαριστά τους κόσμους που αντιστοιχούν στα  $w_1$ ,  $w_2$ , και  $w_3$ . Η ακόλουθη επερώτηση Lorel μπορεί να χρησιμοποιηθεί για να εκφράσει το [detail=high]:

```
select W
from recreation_guide.#._icw W
where W=w1 or W=w2 or W=w3
```

Ας χρησιμοποιήσουμε το  $L_{[detail=high]}$  για να αναφερόμαστε στην επερώτηση αυτή, και το  $L_{[lang=gr]}$  για να αναφερόμαστε σε μια παρόμοια επερώτηση Lorel που εκφράζει το [lang=gr]. Επιπρόσθετα, χρησιμοποιούμε το σύμβολο  $L_{CXT\_VAR}$  για να αναφερόμαστε σε μια επερώτηση Lorel που εκφράζει την κληρονομούμενη κάλυψη μονοπατιού του μονοπατιού review::[-].comments::[-], η οποία είναι η τιμή της μεταβλητής περιβάλλοντος [X]. Αυτή η επερώτηση Lorel είναι η:

```
E-HUB3._icw intersect C-HUB3._icw intersect
E-HUB4._icw intersect C-HUB4._icw
```

Η επερώτηση αποτιμάται στους «κόσμους» κάτω από τους οποίους υφίστανται όλες οι ακμές του μονοπατιού. Τώρα που έχουμε εκφράσει όλα τα περιβάλλοντα σαν επερωτήσεις οι οποίες αποτιμούνται σε σύνολα κόμβων που αναπαριστούν κόσμους, μπορούμε να εκφράσουμε το υποσύνολο περιβάλλοντος σαν μια σχέση ανάμεσα από τις επερωτήσεις. Θεωρώντας ότι το *query1* εκφράζει ένα περιβάλλον [c1] και το *query2* ένα περιβάλλον [c2], η συνθήκη [c1] <= [c2] ([c1] υποσύνολο περιβάλλοντος του [c2]) υλοποιείται από το κατηγορήμα

```
for all LEFT in (query1):
exists RIGHT in (query2): LEFT = RIGHT
```

όπου το LEFT και το RIGHT είναι μεταβλητές που κυμαίνονται πάνω στους «κόσμους» στην αριστερή και στην δεξιά πλευρά του σύμβολου <=, αντίστοιχα.

Η επερώτηση MQL μπορεί τώρα να μεταφραστεί στην ακόλουθη επερώτηση Lorel:

```
select comments: Y
from
  recreation_guide E-HUB1, E-HUB1._ett V1,
  V1._facet C-HUB1, C-HUB1._cxt V2,

  V2.restaurant E-HUB2, E-HUB2._ett V3,
  V3._facet C-HUB2, C-HUB2._cxt V4,

  V4.review E-HUB3, E-HUB3._ett V5,
  V5._facet C-HUB3, C-HUB3._cxt V6,
```

<sup>95</sup> Η επερώτηση μπορεί να εκφραστεί απλούστερα, χρησιμοποιούμε πάντως την συγκεκριμένη σαν μια γενικότερη περίπτωση χάριν του παραδείγματος.

```
V6.comments E-HUB4, E-HUB4._ett V7,
V7._facet C-HUB4, C-HUB4._cxt Y
where
  for all LEFT in (LCXT_VAR intersect L[detail=high]):
    exists RIGHT in (L[lang=gr]): LEFT = RIGHT
```

Χρησιμοποιώντας επερωτήσεις Lorel με παρόμοιο τρόπο, είναι δυνατόν να υλοποιήσουμε όλες τις πράξεις περιβάλλοντος στον όρο *within*.

### 5.5.3 Μετασηματισμός αποτελεσμάτων από OEM σε Πολυδιάστατο Γράφο Δεδομένων

Το LORE επιστρέφει τα αποτελέσματα μιας επερωτήσης Lorel σαν έναν γράφο OEM<sup>96</sup>. Όπως εξηγήσαμε ήδη, αυτός ο γράφος OEM μπορεί να μετασηματιστεί σε έναν Πολυδιάστατο Γράφο Δεδομένων, που είναι το αποτέλεσμα της αρχικής επερωτήσης MQL.

Η διεργασία που μετασηματίζει ένα OEM  $O = (V, E, r, v)$  σε έναν Πολυδιάστατο Γράφο Δεδομένων δίνεται παρακάτω.

**M ← OEMToMDG (O) is:**

1. Αναπαριστούμε το  $O$  σαν έναν Πολυδιάστατο Γράφο Δεδομένων  $M = (V_{mld}, V_{cxt}, E_{cxt}, E_{ett}, r, v)$ , όπου  $V_{cxt} = V$ , και  $E_{ett} = E$ , και τα  $V_{mld}, E_{cxt}$  είναι κενά σύνολα.
2. Για κάθε ακμή  $h = (q, l, u) \in E_{ett}$  όπου το  $l$  δεν είναι δεσμευμένη ετικέτα, απομακρύνουμε το  $u$  από το  $V_{cxt}$ . Στην συνέχεια απομακρύνουμε το  $h$  και το  $(u, \_ett, p)$  από το  $E_{ett}$ , και προσθέτουμε την ακμή  $(q, l, p)$  στο  $E_{ett}$ . Απομακρύνουμε όλες τις ακμές  $(u, \_icw, w)$  από το  $E_{ett}$ .
3. Για κάθε ακμή  $h = (q, \_facet, u) \in E_{ett}$ , μεταφέρουμε το  $q$  από το  $V_{cxt}$  στο  $V_{mld}$  (αν δεν έχει ήδη μετακινηθεί), και απομακρύνουμε το  $u$  από το  $V_{cxt}$ . Για όλους τους κόμβους  $w$ , όπου  $(u, \_ecw, w) \in E_{ett}$ , εφαρμόζουμε ένωση περιβάλλοντος στους αντιστοιχούς κόσμους και παίρνουμε έναν προσδιοριστή περιβάλλοντος  $c$ . Στην συνέχεια απομακρύνουμε το  $h$  και το  $(u, \_cxt, p)$  από το  $E_{ett}$ , και προσθέτουμε την ακμή  $(q, c, p)$  στο  $E_{cxt}$ . Απομακρύνουμε όλες τις ακμές  $(u, \_ecw, w)$  και  $(u, \_icw, w)$  από το  $E_{ett}$ .
4. Απομακρύνουμε από το  $V_{cxt}$  όλους τους κόμβους που αντιστοιχούν σε κόσμους.
5. Επιστρέφουμε το  $M = (V_{mld}, V_{cxt}, E_{cxt}, E_{ett}, r, v)$ .

Το βήμα 4 απομακρύνει όλους τους ατομικούς κόμβους με τιμές ακεραίων που αντιστοιχούν σε πιθανούς κόσμους. Οι κόμβοι αυτοί μπορούν εύκολα να αναγνωριστούν επειδή εκείνη την στιγμή είναι μη προσβάσιμοι από την ρίζα  $r$ . Στο τέλος της διεργασίας, η συνάρτηση  $v$  είναι δυνατόν να περικλείει πιο πολλές αναθέσεις τιμών σε κόμβους από όσες υπάρχουν πραγματικά. Παρατηρήστε ότι, για να ανακατασκευάσει τους προσδιοριστές περιβάλλοντος, το βήμα 3 χρειάζεται την ίδια αντιστοιχία κόσμων σε ακεραίους με αυτή που χρησιμοποιήθηκε κατά τον μετασηματισμό της βάσης δεδομένων MOEM στο OEM.

<sup>96</sup> Κωδικοποιημένο σε XML.

## 5.6 ΠΕΡΙΛΗΨΗ

Στο κεφάλαιο αυτό παρουσιάσαμε την **Πολυδιάστατη Γλώσσα Επερωτήσεων**, ή **MQL** για συντομία, μια γλώσσα επερωτήσεων για πολυδιάστατα ημιδομημένα δεδομένα. Η έμφαση της MQL βρίσκεται στις *οδηγούμενες από περιβάλλον επερωτήσεις*: επερωτήσεις όπου το ερμηνευτικό περιβάλλον πρέπει να ληφθεί υπόψη προκειμένου να επιλεγούν τα σωστά δεδομένα. Η MQL βασίζεται ευρέως στην Lorel και διατηρεί τις βασικές της συνεισφορές, δηλαδή τις εκφράσεις μονοπατιού και τον εξαναγκασμό. Επιπρόσθετα, εισάγει μια σειρά από νέα χαρακτηριστικά προκειμένου:

- *Να χρησιμοποιήσει το ερμηνευτικό περιβάλλον σε επερωτήσεις.* Το ερμηνευτικό περιβάλλον στην MQL αναγνωρίζεται σαν τέτοιο και αντιμετωπίζεται σαν πολίτης πρώτης τάξης. Για την απευθείας υποστήριξη συνθηκών περιβάλλοντος και πράξεων περιβάλλοντος, η MQL χρησιμοποιεί επιπλέον όρους.
- *Να χειριστεί ένα πιο περίπλοκο μοντέλο δεδομένων.* Το μοντέλο δεδομένων της MQL είναι ο Πολυδιάστατος Γράφος Δεδομένων, που χρησιμοποιεί δύο τύπους κόμβων και δύο τύπους ακμών για να αναπαραστήσει πολυδιάστατες οντότητες και τις εκφάνσεις τους. Αυτοί οι γράφοι απαιτούν νέους τρόπους πλοήγησης και καθορισμού προτύπων προσπέλασης. Επιπλέον, αλλάζουν τον τρόπο που κατασκευάζεται ο γράφος των αποτελεσμάτων.

Για να αντιμετωπίσουμε αυτά τα θέματα ορίσαμε τις **εκφράσεις μονοπατιού περιβάλλοντος**, που ενσωματώνουν το ερμηνευτικό περιβάλλον στις συμβατικές εκφράσεις περιβάλλοντος, και επιστρατεύουν την κανονική μορφή του Πολυδιάστατου Γράφου Δεδομένων για να διατυπώσουν πρότυπα προσπέλασης που έχουν επίγνωση του ερμηνευτικού περιβάλλοντος. Οι εκφράσεις μονοπατιού περιβάλλοντος χρησιμοποιούν προσδιοριστές περιβάλλοντος, **μεταβλητές περιβάλλοντος**, και **πρότυπα περιβάλλοντος** για να θέσουν συνθήκες περιβάλλοντος πάνω στον γράφο της βάσης. Βελτιώσαμε τις εκφράσεις μονοπατιού περιβάλλοντος ενσωματώνοντας ομαλές εκφράσεις με τρόπο που να μην επιτρέπονται συνδυασμοί τύπων ακμής που δεν έχουν νόημα. Ένα ενδιαφέρον σημείο είναι ότι μια κατηγορία εκφράσεων μονοπατιού περιβάλλοντος μπορεί να ιδωθεί σαν ορίζουσα συνδέσεις ανάμεσα από συμβατικά OEM που υφίστανται κάτω από διαφορετικούς κόσμους.

Στην συνέχεια, εισήγαμε την MQL, με δύο ειδικούς όρους για τον χειρισμό του ερμηνευτικού περιβάλλοντος: τον όρο *within* που εκφράζει συνθήκες πάνω στο ερμηνευτικό περιβάλλον, και τον όρο *context* που δημιουργεί νέες μεταβλητές περιβάλλοντος για να χρησιμοποιηθούν στην κατασκευή αποτελεσμάτων. Εξετάσαμε τον όρο *select* λεπτομερώς: επικεντρωθήκαμε στην κατασκευή αποτελεσμάτων που αποτελούνται από δύο τύπους κόμβων και δύο τύπους ακμών, και στην χρήση εκφράσεων μονοπατιού περιβάλλοντος στο *select*. Έπειτα, εξηγήσαμε το πως η MQL ανάγει τον γράφο των αποτελεσμάτων, κάνοντας χρήση των ιδιοτήτων που παρουσιάστηκαν στο προηγούμενο κεφάλαιο. Για να συμπληρώσουμε την συζήτηση πάνω στην MQL, παρουσιάσαμε μια σειρά χαρακτηριστικών που δεν επηρεάζονται ιδιαίτερα από το ερμηνευτικό περιβάλλον, όπως οι μεταβλητές μονοπατιού, οι μεταβλητές ετικέτας, και οι εμφωλιασμένες επερωτήσεις. Στο Παράρτημα Α ορίζουμε αυστηρά το συντακτικό των εκφράσεων μονοπατιού περιβάλλοντος και της MQL.

Η πρότυπη υλοποίησή της MQL μας επιτρέπει την σύγκριση επερωτήσεων MQL με ισοδύναμες επερωτήσεις Lorel: σε επερωτήσεις οδηγούμενες από περιβάλλον, η MQL και το MOEM είναι φανερά πολύ πιο εκφραστικά και κομψά απ' ότι η Lorel και το OEM, ενώ σε συμβατικές επερωτήσεις η MQL και οι εκφράσεις μονοπατιού περιβάλλοντος γίνονται το ίδιο απλές στον σχηματισμό τους όπως η Lorel και οι συμβατικές εκφράσεις μονοπατιού.

Κατά την γνώμη μας, οι δυνατότητες των MSSD και η εκφραστικότητα της MQL δικαιώνει τις επεκτάσεις στο OEM που εισήγαμε στο προηγούμενο κεφάλαιο. Τα επιπλέον αυτά στοιχεία στον Πολυδιάστατο Γράφο Δεδομένων χρησιμοποιούνται από την MQL για την διατύπωση *δια-κοσμικών επερωτήσεων*, που δεν έχουν αντίστοιχες σε βάσεις δεδομένων χωρίς επίγνωση του ερμηνευτικού περιβάλλοντος.

## 6 ΑΝΤΙΜΕΤΩΠΙΣΗ ΑΛΛΑΓΩΝ ΜΕ ΧΡΗΣΗ MSSD

Τα πολυδιάστατα ημιδομημένα δεδομένα (MSSD) είναι ημιδομημένα δεδομένα που παρουσιάζουν διαφορετικές εκφάνσεις κάτω από διαφορετικά ερμηνευτικά περιβάλλοντα. Στα προηγούμενα κεφάλαια εξηγήθηκαν διάφορες πλευρές των MSSD. Το ερμηνευτικό περιβάλλον ορίστηκε σαν κάτι που αντιπροσωπεύει εναλλακτικούς κόσμους, και εκφράζεται με την ανάθεση τιμών σε ένα σύνολο μεταβλητών που λέγονται διαστάσεις. Η έννοια του ερμηνευτικού περιβάλλοντος ενσωματώθηκε στο OEM, δίνοντας το Πολυδιάστατο OEM (MOEM), ένα μοντέλο γράφου για MSSD. Η Πολυδιάστατη Γλώσσα Επερωτήσεων (MQL) προτάθηκε για την διατύπωση επερωτήσεων εξαρτώμενων από το ερμηνευτικό περιβάλλον στα MSSD.

Στο παρόν κεφάλαιο, χρησιμοποιούμε ότι αναπτύξαμε στα προηγούμενα κεφάλαια για να αντιμετωπίσουμε ένα συγκεκριμένο πρόβλημα, και να παρουσιάσουμε μια εφαρμογή των MSSD που να επιδεικνύει τις δυνατότητές τους. Το πρόβλημα μπορεί να διατυπωθεί ως εξής: δεδομένου ενός γράφου OEM που αποτελεί την βάση, θα θέλαμε να αναπαριστούμε δυναμικά τις αλλαγές στην βάση καθώς αυτές συμβαίνουν, κρατώντας ένα ιστορικό μεταβάσεων, ώστε να είμαστε σε θέση αργότερα να θέτουμε επερωτήσεις πάνω στις αλλαγές αυτές.

Ακολουθως, αρχίζουμε υπενθυμίζοντας τους ορισμούς των τεσσάρων λειτουργιών βασικών αλλαγών για OEM, και εισάγοντας τις δικές μας **λειτουργίες βασικών αλλαγών** για το MOEM. Στην συνέχεια ορίζουμε πως το MOEM μπορεί να χρησιμοποιηθεί για να αναπαραστήσει το ιστορικό μιας βάσης δεδομένων OEM. Έπειτα, συζητούμε πως οι ιδιότητες του Πολυδιάστατου Γράφου Δεδομένων εφαρμόζονται στην περίπτωση της αναπαράστασης ιστορικού OEM, και δείχνουμε ότι **χρονικά στιγμιότυπα OEM** μπορούν να ανακτηθούν με την αναγωγή του MOEM. Παρουσιάζουμε ένα σύστημα που υλοποιεί τα παραπάνω, και ακολουθούμε ένα αναλυτικό παράδειγμα που επιδεικνύει πως μια υφιστάμενη βάση MOEM μπορεί να φιλοξενήσει αλλαγές σε μια βάση OEM. Επιπλέον, δείχνουμε ότι το MOEM μπορεί να μοντελοποιήσει αλλαγές που συμβαίνουν όχι μόνο σε βάσεις OEM, αλλά επίσης και σε βάσεις MOEM. Στην συνέχεια, επιδεικνύουμε πως η MQL μπορεί να χρησιμοποιηθεί για να εκφράσει διάφορες επερωτήσεις πάνω στα ιστορικά βάσεων OEM and MOEM. Τέλος, συγκρίνουμε την προσέγγισή μας με προηγούμενη δουλειά στην αναπαράσταση και επερώτηση ιστορικών βάσεων ημιδομημένων δεδομένων.

### 6.1 ΛΕΙΤΟΥΡΓΙΕΣ ΒΑΣΙΚΩΝ ΑΛΛΑΓΩΝ ΤΟΥ MOEM

Ένας συμβατικός γράφος OEM ορίζεται στο [Suc98, AQM+97] σαν ένα τετράπτυχο  $O = (V, E, r, v)$ , όπου  $V$  είναι ένα σύνολο κόμβων,  $E$  είναι ένα σύνολο κατευθυνόμενων ακμών με ετικέτα  $(p, l, q)$  όπου  $p, q \in V$  και  $l$  είναι μια στοιχειοσειρά,  $r$  είναι ένας ειδικός κόμβος που λέγεται ρίζα<sup>97</sup>, και  $v$  είναι μια συνάρτηση που αντιστοιχεί κάθε κόμβο σε μια ατομική τιμή κάποιου τύπου (ακέραιος, στοιχειοσειρά, κτλ.), ή στην δεσμευμένη τιμή  $c$  που δηλώνει ένα

<sup>97</sup> Κάθε κόμβος στο  $V$  πρέπει να είναι προσβάσιμος από την ρίζα μέσω κάποιου μονοπατιού.

σύνθετο αντικείμενο. Για να τροποποιήσουμε μια βάση OEM που καλείται  $O$ , οι παρακάτω τέσσερις λειτουργίες βασικών αλλαγών έχουν αναγνωρισθεί στο [CAW99].

- $creNode(nid, val)$ : δημιουργεί έναν νέο κόμβο, όπου  $nid$  είναι ένα νέο αναγνωριστικό (oid) του νέου κόμβου ( $nid \notin V$ ), και  $val$  είναι μια ατομική τιμή ή η δεσμευμένη τιμή  $c$ .
- $updNode(nid, val)$ : αλλάζει την τιμή του υπάρχοντος αντικειμένου  $nid$  στην νέα τιμή  $val$ . Ο κόμβος  $nid$  δεν μπορεί να έχει ακμές που αναχωρούν από αυτόν (σε περίπτωση που η παλιά τιμή του είναι  $c$ , οι ακμές πρέπει να έχουν απομακρυνθεί προτού ενημερωθεί η τιμή).
- $addArc(p, l, q)$ : προσθέτει μια νέα ακμή με την ετικέτα  $l$  από το αντικείμενο  $p$  στο αντικείμενο  $q$ . Και οι δύο κόμβοι  $p$  και  $q$  πρέπει να υπάρχουν ήδη στο  $V$ , και η  $(p, l, q)$  δεν πρέπει να υπάρχει στο  $E$ .
- $remArc(p, l, q)$ : διαγράφει την υπάρχουσα ακμή  $(p, l, q)$ . Και οι δύο κόμβοι  $p$  και  $q$  πρέπει να υπάρχουν στο  $V$ .

Δεδομένης μιας βάσης MOEM που καλείται  $M = (V_{mld}, V_{cxt}, E_{cxt}, E_{eff}, r, v)$ , εισάγουμε τις ακόλουθες βασικές λειτουργίες για να τροποποιούμε το  $M$ <sup>98</sup>. Υπενθυμίζουμε ότι το  $V = V_{mld} \cup V_{cxt}$  είναι το σύνολο όλων των κόμβων στο  $M$ , και το  $E = E_{cxt} \cup E_{eff}$  είναι το σύνολο όλων των ακμών στο  $M$ .

- $createCNode(cid, val)$ : δημιουργείται ένας νέος κόμβος περιβάλλοντος. Το αναγνωριστικό  $cid$  είναι νέο και πρέπει να μην υπάρχει ήδη στο  $V_{cxt}$ . Η τιμή  $val$  μπορεί να είναι μια ατομική τιμή κάποιου τύπου, ή η δεσμευμένη τιμή  $c$ .
- $updateCNode(cid, val)$ : αλλάζει την τιμή του  $cid \in V_{cxt}$  σε  $val$ . Ο κόμβος δεν μπορεί να έχει ακμές που αναχωρούν από αυτόν.
- $createMNode(mid)$ : δημιουργείται ένας νέος πολυδιάστατος κόμβος. Το αναγνωριστικό  $mid$  είναι νέο και δεν πρέπει να υπάρχει ήδη στο  $V_{mld}$ .
- $addEEdge(cid, l, id)$ : δημιουργεί μια νέα ακμή οντότητας με ετικέτα  $l$  από τον κόμβο  $cid$  στον κόμβο  $id$ , όπου  $cid \in V_{cxt}$  και  $id \in V$ .
- $remEEdge(cid, l, id)$ : διαγράφει την ακμή οντότητας  $(cid, l, id)$  από το  $M$ . Η ακμή  $(cid, l, id)$  πρέπει να υπάρχει στο  $E_{eff}$ .
- $addCEdge(mid, cxt, id)$ : δημιουργεί μια νέα ακμή περιβάλλοντος με το περιβάλλον  $cxt$  σαν ετικέτα, από τον κόμβο  $mid$  στον κόμβο  $id$ , όπου  $mid \in V_{mld}$  και  $id \in V$ .
- $remCEdge(mid, cxt, id)$ : διαγράφει την ακμή περιβάλλοντος  $(mid, cxt, id)$  από το  $M$ . Η ακμή περιβάλλοντος  $(mid, cxt, id)$  πρέπει να υπάρχει στο  $E_{cxt}$ .

Στο OEM όπως και στο MOEM η διαγραφή αντικειμένων επιτυγχάνεται μέσω της διαγραφής ακμών, αφότου στο OEM και στο MOEM η διατήρηση ενός αντικειμένου αποφασίζεται με βάση την προσβασιμότητα του αντικειμένου από την ρίζα του γράφου.

Κάποιες φορές το αποτέλεσμα μιας βασικής λειτουργίας  $u$  οδηγεί την βάση σε ασυνεπή κατάσταση: για παράδειγμα, όταν δημιουργείται ένα νέο αντικείμενο, είναι προσωρινά μη

<sup>98</sup> Αυτές οι λειτουργίες βασικών αλλαγών μπορούν να εφαρμοστούν όχι μόνο σε MOEMs αλλά γενικά σε Πολυδιάστατους Γράφους Δεδομένων, στο παρόν κεφάλαιο πάντως επικεντρωθήκαμε σε MOEMs.



προσβάσιμο από την ρίζα. Στην πράξη πάντως, είναι τυπικό να έχουμε μια ακολουθία  $L = u_1, u_2, \dots, u_n$  βασικών λειτουργιών  $u_i$ , που αντιστοιχεί σε μια υψηλότερου επιπέδου μετατροπή στην βάση δεδομένων. Συνδέοντας τέτοιες υψηλότερου επιπέδου μετατροπές με μια χρονοσφραγίδα, ορίζουμε σαν το ιστορικό  $H$  ενός OEM την ακολουθία ζευγών  $(t, U)$ , όπου  $U$  είναι το σύνολο των λειτουργιών βασικών αλλαγών που αντιστοιχεί [CAW99] στο  $L$ , και  $t$  είναι η ανάλογη χρονοσφραγίδα. Σημειώστε ότι μέσα στα πλαίσια μιας ακολουθίας  $L$ , ένας μόλις δημιουργημένος κόμβος μπορεί να μην είναι προσβάσιμος από την ρίζα και παρόλα αυτά να μην θεωρείται διαγραμμένος. Στο τέλος κάθε ακολουθίας πάντως, μη προσβάσιμοι κόμβοι θεωρούνται διαγραμμένοι και οι μετέπειτα λειτουργίες δεν μπορούν να αναφερθούν σε αυτούς.

## 6.2 ΑΝΑΠΑΡΑΣΤΑΣΗ ΙΣΤΟΡΙΚΟΥ ΤΟΥ OEM ΜΕ ΜΟΕΜ

Στην ενότητα αυτή δίνουμε μια λεπτομερή εξήγηση του τρόπου με τον οποίο το ΜΟΕΜ μπορεί να αναπαραστήσει αλλαγές σε μια βάση δεδομένων OEM. Στην συνέχεια, συζητούμε το πως οι ιδιότητες του Πολυδιάστατου Γράφου Δεδομένων, όπως η κληρονομούμενη κάλυψη και η αναγωγή, εφαρμόζονται στην περίπτωση της αναπαράστασης του ιστορικού του OEM. Συγκεκριμένα, δείχνουμε ότι μπορούμε να χρησιμοποιήσουμε την αναγωγή σε OEM και την μερική αναγωγή για να αποκτήσουμε χρονικά στιγμιότυπα της βάσης OEM.

### 6.2.1 Μοντελοποίηση ιστορικών του OEM με χρήση ΜΟΕΜ

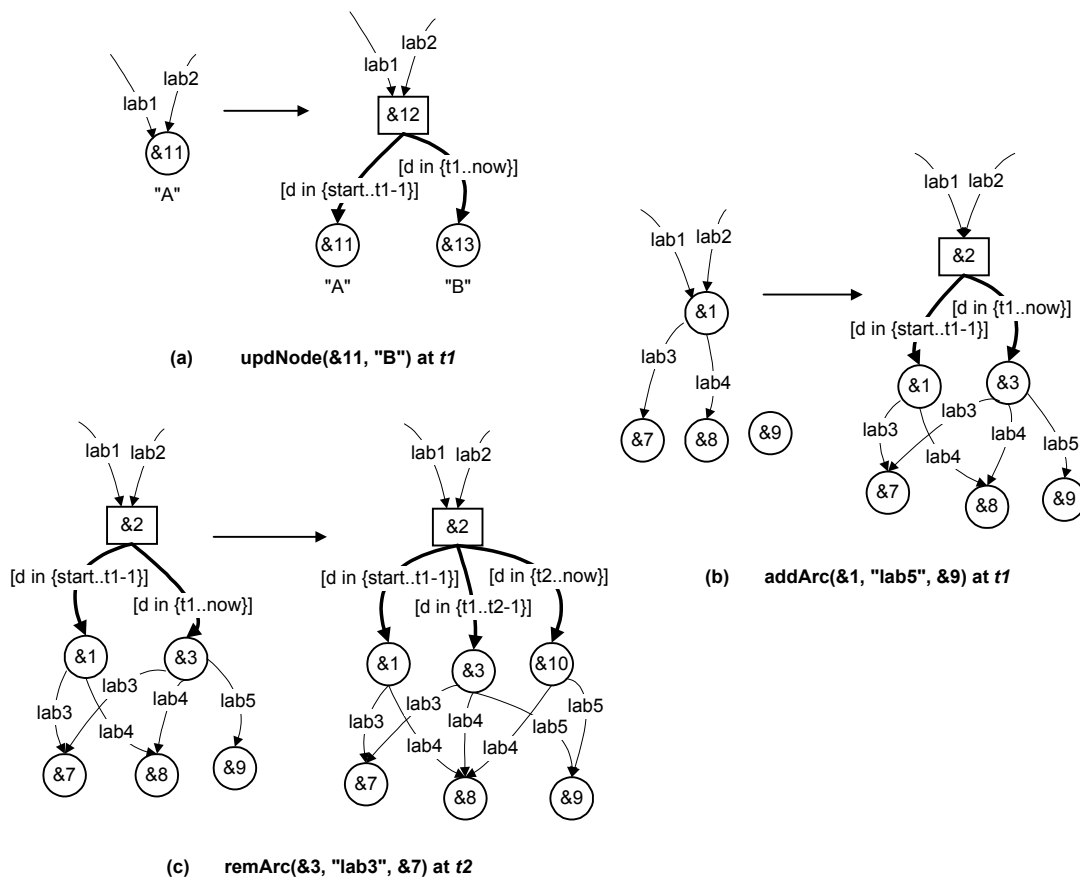
Η προσέγγισή μας είναι να αντιστοιχίσουμε τις τέσσερις λειτουργίες βασικών αλλαγών του OEM σε βασικές λειτουργίες του ΜΟΕΜ, με τέτοιον τρόπο ώστε να δημιουργούνται νέες εκφάνσεις ενός αντικειμένου κάθε φορά που συμβαίνουν αλλαγές στο αντικείμενο αυτό. Έτσι, η αρχική βάση OEM  $O$  μετατρέπεται σε έναν γράφο ΜΟΕΜ<sup>99</sup>, που χρησιμοποιεί μια διάσταση  $d$  της οποίας το πεδίο ορισμού είναι ο χρόνος για να αναπαραστήσει ένα ιστορικό  $H$  έγκυρο [CAW99] για το  $O$ . Θεωρούμε ότι το πεδίο του χρόνου  $T$  είναι συνεχές και διακριτό. Επίσης θεωρούμε:

- Μια δεσμευμένη τιμή  $start$ , τέτοια ώστε  $start < t$  για κάθε  $t \in T$ , που αναπαριστά την αρχή του χρόνου.
- Μια δεσμευμένη τιμή  $now$ , τέτοια ώστε  $t < now$  για κάθε  $t \in T$ , που αναπαριστά την τρέχουσα χρονική στιγμή.

Η χρονική περίοδος κατά την οποία ένας κόμβος περιβάλλοντος είναι ο υφιστάμενος κόμβος της αντίστοιχης πολυδιάστατης οντότητας δηλώνεται προσάπτοντας στον κόμβο περιβάλλοντος έναν προσδιοριστή περιβάλλοντος της μορφής  $[d \text{ in } \{t_1..t_n\}]$ . Ας θυμηθούμε ότι στους προσδιοριστές περιβάλλοντος η συντακτική συντομογραφία  $v_1..v_n$  για διακριτά και πλήρως διατεταγμένα πεδία ορισμού συμβολίζει όλες τις τιμές  $v_i$  τέτοιες ώστε  $v_1 \leq v_i \leq v_n$ .

<sup>99</sup> Η αρχική βάση OEM μετατρέπεται σε έναν Πολυδιάστατο Γράφο Δεδομένων, ο οποίος, όπως θα δείξουμε στην Ενότητα 6.2.2, είναι πάντα ΜΟΕΜ.

**Εικόνα 6.1: Μοντελοποίηση των λειτουργιών βασικών αλλαγών του OEM με χρήση MOEM.**



Η Εικόνα 6.1 δίνει μια διαίσθηση για την αντιστοιχία ανάμεσα στις λειτουργίες OEM και MOEM. Θεωρήστε τα σύνολα  $U_1$  and  $U_2$  λειτουργιών βασικών αλλαγών, με χρονοσφραγίδες  $t_1$  και  $t_2$  αντίστοιχα. Η Εικόνα 6.1 (a) δείχνει την αναπαράσταση σε MOEM ενός ατομικού αντικειμένου, του οποίου η τιμή "A" αλλάζει σε "B" μέσω μιας κλήσης στην λειτουργία βασικών αλλαγών `updNode` του  $U_1$ . Η Εικόνα 6.1 (b) δείχνει το αποτέλεσμα της λειτουργίας `addArc` του  $U_1$ , ενώ η Εικόνα 6.1 (c) δείχνει το αποτέλεσμα της λειτουργίας `remArc` του  $U_2$ , πάνω στην ίδια πολυδιάστατη οντότητα. Είναι ενδιαφέρον να παρατηρήσει κανείς ότι τρεις από τις τέσσερις λειτουργίες βασικών αλλαγών του OEM είναι παρόμοιες, στο ότι ενημερώνουν ένα αντικείμενο είτε ατομικό (`updNode`) είτε σύνθετο (`addArc`, `remArc`), και αντιστοιχούν και οι τρεις σε λειτουργίες του MOEM που στην πράξη ενημερώνουν μια νέα έκφανση του αρχικού αντικειμένου. Η δημιουργία ενός νέου κόμβου με την `creNode` δεν καταλήγει σε επιπλέον λειτουργίες στο MOEM: ο νέος κόμβος θα ενωθεί στην συνέχεια με τον υπόλοιπο γράφο (στα πλαίσια του ίδιου συνόλου  $U_i$ ) μέσω της λειτουργίας `addArc`, πράγμα το οποίο θα προκαλέσει την δημιουργία νέων εκφάνσεων των εμπλεκόμενων αντικειμένων. Αξίζει να σημειωθεί ότι οι αλλαγές που προκαλούνται εξαιτίας των λειτουργιών βασικών αλλαγών του OEM επηρεάζουν μόνον τοπικά τμήματα του MOEM γράφου, και δεν διαδίδονται σε όλη την έκταση του γράφου.

Όπως φαίνεται στην Εικόνα 6.1, ένα αντικείμενο OEM μπορεί να αντιστοιχεί σε έναν αριθμό MOEM αντικειμένων με διαφορετικά αναγνωριστικά (oids), πράγμα που εκλαμβάνεται σαν μια αλλαγή του αναγνωριστικού κατά την διάρκεια του ιστορικού ενός

αντικειμένου OEM. Πάντως, αυτό είναι περισσότερο θέμα υλοποίησης και δεν αποτελεί πραγματικό πρόβλημα.

Έχοντας διαγράψει την προσέγγιση, δίνουμε τώρα τις ακριβείς προδιαγραφές. Αρχικά, ορίζουμε τις παρακάτω τέσσερις βοηθητικές συναρτήσεις και διαδικασίες για το MOEM.

1.  $id1 \leftarrow md(id2)$ , με  $id1, id2 \in V$ . Επιστρέφει τον αντίστοιχο πολυδιάστατο κόμβο για έναν κόμβο περιβάλλοντος, εάν υπάρχει τέτοιος. Αν  $id2 \in V_{cxt}$  και υπάρχει στοιχείο  $(mid, cxt, id)$  στο  $E_{cxt}$  τέτοιο ώστε  $id = id2$ , τότε επιστρέφεται το  $mid$ . Αν  $id2 \in V_{cxt}$  και δεν υπάρχει αντίστοιχη ακμή περιβάλλοντος, επιστρέφεται το  $id2$ . Αν  $id2 \in V_{mld}$ , επιστρέφεται το  $id2$ . Θεωρούμε ότι υπάρχει το πολύ ένας πολυδιάστατος κόμβος που να δείχνει σε οποιονδήποτε κόμβο περιβάλλοντος, με αλλά λόγια για κάθε  $cid \in V_{cxt}$  υπάρχει το πολύ ένα  $mid$  τέτοιο ώστε  $(mid, cxt, cid) \in E_{cxt}$ . Σημειώστε πάντως, ότι αυτή είναι μια ιδιότητα του MOEM για το συγκεκριμένο πρόβλημα λόγω του ειδικού τρόπου που κατασκευάζεται ο MOEM γράφος για την αναπαράσταση ιστορικών, και δεν αφορά την γενική περίπτωση.
2.  $boolean \leftarrow withinSet(cid)$ , με  $cid \in V_{cxt}$ . Ελέγχει εάν ο κόμβος περιβάλλοντος  $cid$  δημιουργείται στα πλαίσια του τρέχοντος συνόλου λειτουργιών βασικών αλλαγών  $U$ . Η συνάρτηση αυτή χρησιμοποιείται ενώ οι λειτουργίες αλλαγών βρίσκονται σε εξέλιξη, και επιστρέφει  $true$  αν το  $cid$  δημιουργήθηκε στα πλαίσια του ίδιου συνόλου. Επιστρέφει  $false$  αν το  $cid$  δημιουργήθηκε στα πλαίσια ενός προηγούμενου συνόλου λειτουργιών.
3. Η επόμενη διαδικασία  $mEntity(id)$ , με  $id \in V_{cxt}$ , δημιουργεί έναν νέο πολυδιάστατο κόμβο  $mid$  που δείχνει στον  $id$ , και αναδιατάσσει όλες τις εισερχόμενες ακμές ώστε αντί για το  $id$  να δείχνουν στο  $mid$ . Σημειώστε ότι ενώ η διαδικασία αλλάζει τον γράφο, δεν τροποποιεί την πληροφορία που μοντελοποιεί ο γράφος: η πολυδιάστατη οντότητα που δημιουργείται από την διαδικασία έχει τον κόμβο  $id$  σαν μοναδική έκφραση που υφίσταται κάτω από κάθε κόσμο<sup>100</sup>.

```
mEntity(id) {
  createMNode(mid)
  addCEdge(mid, [d in {start..now}], id)
  for every (x, l, id) in Eett {
    addEEEdge(x, l, mid)
    remEEEdge(x, l, id)
  }
}
```

4. Στην διαδικασία  $newCxt(id1, id2, ts)$ , με  $id1, id2 \in V_{cxt}$  και  $ts \in T$ , το  $id1$  είναι τρεχόντως η πιο πρόσφατη έκφραση της πολυδιάστατης οντότητας, και το  $id2$  είναι μια νέα έκφραση που πρόκειται να γίνει η πιο πρόσφατη. Η διαδικασία μετατρέπει κατάλληλα τους προσδιοριστές περιβάλλοντος.

```
newCxt(id1, id2, ts) {
  remCEdge(md(id1), [d in {x..now}], id1)
  addCEdge(md(id1), [d in {x..ts-1}], id1)
  addCEdge(md(id1), [d in {ts..now}], id2)
}
```

Το επόμενο βήμα είναι να δείξουμε πως υλοποιείται κάθε λειτουργία βασικών αλλαγών του OEM χρησιμοποιώντας τις βασικές λειτουργίες του MOEM. Θεωρούμε ότι κάθε

<sup>100</sup> Ο γράφος που προκύπτει έχει την ίδια κανονική μορφή με τον αρχικό γράφο.

λειτουργία του OEM αποτελεί μέρος ενός συνόλου  $U$  με χρονοσφραγίδα  $t_s$ , και ότι ο κόμβος  $p$  είναι ο πιο πρόσφατος κόμβος περιβάλλοντος της αντίστοιχης πολυδιάστατης οντότητας, εάν αυτή υπάρχει. Αυτό επειδή οι αλλαγές συμβαίνουν πάντα στο τρέχον στιγμιότυπο του OEM, το οποίο αντιστοιχεί στις πιο πρόσφατες εκφάνσεις των πολυδιάστατων οντοτήτων του MOEM. Ο πιο πρόσφατος κόμβος περιβάλλοντος είναι αυτός που υφίσταται την τρέχουσα χρονική στιγμή, δηλαδή ο κόμβος που έχει προσδιοριστή περιβάλλοντος της μορφής  $[d \text{ in } \{somevalue..now\}]$ .

- `updNode(p, newval)` λειτουργία αλλαγής OEM:

Αν το  $p$  έχει δημιουργηθεί στα πλαίσια του  $U$ , η τιμή του ενημερώνεται άμεσα, και η διαδικασία τελειώνει. Διαφορετικά, αν το  $p$  δεν δείχνεται από κάποιον πολυδιάστατο κόμβο, ένας νέος πολυδιάστατος κόμβος δημιουργείται για το  $p$ , έχοντας το  $p$  σαν τον μόνο του κόμβο περιβάλλοντος με προσδιοριστή περιβάλλοντος  $[d \text{ in } \{start..now\}]$ . Μια νέα έκφανση δημιουργείται τότε με τιμή `newval`, και γίνεται η πιο πρόσφατη έκφανση τροποποιώντας τους σχετικούς προσδιοριστές περιβάλλοντος. Καθώς ένας κόμβος που ενημερώνεται από την `updNode` δεν μπορεί να έχει απερχόμενες ακμές, δεν λαμβάνει χώρα αντιγραφή ακμών σε αντίθεση με την περίπτωση της `addArc`.

```

updNode(p, newval) {
  if not withinSet(p) {
    if not exists(x, cxt, p) in Ecxt
      mEntity(p)
      createCNode(n, newval)
      newCxt(p, n, ts)
  }
  else updateCNode(p, newval)
}

```

- `addArc(p, l, q)` λειτουργία αλλαγής OEM:

Αν το  $p$  έχει δημιουργηθεί στα πλαίσια του  $U$ , χρησιμοποιείται απευθείας: προστίθεται η νέα ακμή, και η διαδικασία τερματίζει. Διαφορετικά, αν το  $p$  δεν δείχνεται ήδη από κάποιον πολυδιάστατο κόμβο, ένας νέος πολυδιάστατος κόμβος δημιουργείται για το  $p$ , έχοντας το  $p$  σαν τον μόνο του κόμβο περιβάλλοντος με προσδιοριστή περιβάλλοντος  $[d \text{ in } \{start..now\}]$ . Μια νέα έκφανση «κλώνος» δημιουργείται τότε, αντιγράφοντας όλες τις απερχόμενες ακμές του  $p$  στο  $n$ . Στην περίπτωση αυτή, οι προσδιοριστές περιβάλλοντος προσαρμόζονται έτσι ώστε να ληφθεί υπόψη το  $t_s$ , και το  $n$  γίνεται η πιο πρόσφατη έκφανση όπως απεικονίζεται στην Εικόνα 6.1 (b) για  $t_s = t_1$ . Τέλος η νέα ακμή που προσδιορίζεται από την λειτουργία βασικών αλλαγών προστίθεται στην πιο πρόσφατη έκφανση. Σημειώστε ότι, στα πλαίσια της αναπαράστασης αλλαγών, το MOEM κατασκευάζεται με τέτοιον τρόπο που μια ακμή οντότητας δεν δείχνει απευθείας σε έναν κόμβο περιβάλλοντος  $q_c$  εφόσον υπάρχει κάποια ακμή περιβάλλοντος ( $q_m, cxt, q_c$ ); αντίθετα, δείχνει πάντα στον αντίστοιχο πολυδιάστατο κόμβο  $q_m$ , αν αυτός υπάρχει. Αυτό επιτυγχάνεται με την χρήση της συνάρτησης `md(q)` σε συνδυασμό με την `mEntity(p)`.

```

addArc(p, l, q) {
  if not withinSet(p) {
    if not exists(x, cxt, p) in Ecxt
      mEntity(p)
      createCNode(n, 'C')
      newCxt(p, n, ts)
      for every (p, k, y) in Eett
        addEEdge(n, k, y)
      addEEdge(n, l, md(q))
  }
  else addEEdge(p, l, md(q))
}

```

- `remArc(p, l, q)` λειτουργία αλλαγής OEM:

Η διαδικασία είναι ουσιαστικά η ίδια με την `addArc(p, l, q)`, με την διαφορά ότι στο τέλος της διαδικασίας αφαιρείται μια ακμή, αντί να προστεθεί. Έτσι, η `remArc` είναι όπως η `addArc` εκτός από τις δυο τελευταίες κλήσεις στην `addEEdge`, που αντικαθίστανται από κλήσεις στην `remEEdge` με τα ίδια ορίσματα.

- `creNode(p, val)` λειτουργία αλλαγής OEM:

Αυτή η λειτουργία βασικών αλλαγών αντιστοιχίζεται στην `createCNode(p, val)` χωρίς περαιτέρω βήματα. Νέες εκφάνσεις θα δημιουργηθούν όταν προστεθούν νέες ακμές για να συνδέσουν τον κόμβο  $p$  με τον υπόλοιπο γράφο.

## 6.2.2 Εφαρμογή ιδιοτήτων των MSSD

Στην προηγούμενη ενότητα εξηγήσαμε πως το ιστορικό μιας βάσης δεδομένων OEM ενσωματώνεται σε έναν γράφο MOEM, χρησιμοποιώντας πολυδιάστατους κόμβους και ακμές περιβάλλοντος. Όμως, δεν έχουμε δείξει ότι ο γράφος αυτός είναι πράγματι ένας γράφος MOEM. Προφανώς είναι ένας Πολυδιάστατος Γράφος Δεδομένων, αλλά για να είναι MOEM γράφος (a) πρέπει να είναι αιτιοκρατικός ως προς το περιβάλλον, και (b) κάθε κόμβος και ακμή πρέπει να έχουν μη κενή κληρονομούμενη κάλυψη.

Στην ενότητα αυτή εξετάζουμε το πως ιδιότητες των MSSD όπως η αιτιοκρατία ως προς το περιβάλλον, η κληρονομούμενη κάλυψη, και η αναγωγή εφαρμόζονται στην ιδιαίτερη περίπτωση της αναπαράστασης ιστορικών του OEM. Δείχνουμε ότι ένας Πολυδιάστατος Γράφος Δεδομένων κατασκευασμένος όπως ορίστηκε στην Ενότητα 6.2.1 είναι πάντα MOEM, και ακόμη περισσότερο, ένα MOEM που έχει ειδικά χαρακτηριστικά που δεν συναντώνται γενικά στα MOEMs.

Ας θεωρήσουμε ότι ο  $G$  είναι ένας Πολυδιάστατος Γράφος Δεδομένων που δημιουργήθηκε από την διεργασία που ορίζεται στην Ενότητα 6.2.1, ότι το  $e$  είναι μια πολυδιάστατη οντότητα στον  $G$ , με πολυδιάστατο κόμβο  $m$  και εκφάνσεις  $e_1, e_2, \dots, e_n$ , και ότι οι  $c_1, c_2, \dots, c_n$  είναι προσδιοριστές περιβάλλοντος των αντίστοιχων ακμών περιβάλλοντος. Τότε, η διεργασία στην Ενότητα 6.2.1 εγγυάται τις ακόλουθες ειδικές ιδιότητες για τον  $G$ :

- Οι ακμές περιβάλλοντος δείχνουν πάντα σε κόμβους περιβάλλοντος, συνεπώς οι  $e_1, e_2, \dots, e_n$  είναι αποκλειστικά κόμβοι περιβάλλοντος.
- Ένας κόμβος περιβάλλοντος  $e_i$  αποτελεί έκφραση το πολύ ενός πολυδιάστατου κόμβου  $m$ , συνεπώς το πολύ μια ακμή περιβάλλοντος μπορεί να δείχνει στον  $e_i$ .

- (c) Κάθε πολυδιάστατος κόμβος  $m$  (εκτός πιθανώς από την ρίζα) δείχνεται από μια το λιγότερο ακμή οντότητας.
- (d) Μια χρονική στιγμή είναι κόσμος για τον  $G$ .
- (e) Τα φύλλα του  $G$  είναι αποκλειστικά ατομικοί κόμβοι<sup>101</sup>.

Λόγω της διαδικασίας `newCxt`, που κατασκευάζει τις ακμές περιβάλλοντος, για κάθε πολυδιάστατη οντότητα  $e$  στον  $G$  τα ρητά περιβάλλοντα  $c_1, c_2, \dots, c_n$  ορίζουν πάντοτε ξένα σύνολα κόσμων. Συνεπώς, οι κληρονομούμενες καλύψεις των ακμών περιβάλλοντος είναι αμοιβαία αποκλειόμενες, αφού είναι υποσύνολα περιβάλλοντος των αντίστοιχων ρητών περιβαλλόντων. Λόγω της ειδικής ιδιότητας (a) παραπάνω, δεδομένου κάποιου κόσμου  $w$ , αν ξεκινήσουμε από το  $m$  μπορούμε να καταλήξουμε το πολύ σε έναν κόμβο περιβάλλοντος μέσω ακμών περιβάλλοντος που υφίστανται κάτω από τον  $w$ , κι έτσι ο  $G$  είναι αιτιοκρατικός ως προς το περιβάλλον.

Επιπρόσθετα, από τις διαδικασίες `mEntity` και `newCxt` μπορούμε να δούμε ότι:

- (f) Το  $c_1$  έχει την μορφή  $[d \text{ in } \{\text{start}..somevalue\_I\}]$
- (g) Το  $c_n$  έχει την μορφή  $[d \text{ in } \{somevalue\_N..now\}]$
- (h) Η ένωση περιβάλλοντος των  $c_1, c_2, \dots, c_n$  είναι  $[d \text{ in } \{\text{start}..now\}]$ , για κάθε  $e$  στον  $G$ .

Παρόλο που για κάθε  $e$  στον  $G$  τα ρητά περιβάλλοντα  $c_1, c_2, \dots, c_n$  καλύπτουν ολόκληρο το χρονικό εύρος  $\{\text{start}..now\}$ , δεν ισχύει το ίδιο στην περίπτωση των αντίστοιχων κληρονομούμενων καλύψεων, οι οποίες δίνουν το πραγματικό διάστημα ζωής μιας οντότητας και των εκφάνσεών της. Καταρχήν, ας συζητήσουμε για την κάλυψη περιβάλλοντος. Λόγω των ειδικών ιδιοτήτων (c), (e), και (h) παραπάνω, η κάλυψη περιβάλλοντος όλων των κόμβων περιβάλλοντος, των πολυδιάστατων κόμβων, και των ακμών οντότητας στον  $G$  είναι  $[d \text{ in } \{\text{start}..now\}]$ , που είναι ένα καθολικό περιβάλλον. Εξάλλου, η κάλυψη περιβάλλοντος μιας ακμής περιβάλλοντος ταυτίζεται με το ρητό περιβάλλον της. Συνεπώς, στην περίπτωση της αναπαράστασης ιστορικών του OEM η κάλυψη περιβάλλοντος δεν μεταδίδει περιορισμούς, και για κάθε κόμβο και ακμή στον  $G$  η κληρονομούμενη κάλυψη συμπίπτει με το κληρονομούμενο περιβάλλον.

Ας εξετάσουμε το νόημα του κληρονομούμενου περιβάλλοντος στον  $G$ . Κάθε πολυδιάστατη οντότητα  $e$  στον  $G$  αντιστοιχεί σε έναν κόμβο που υπήρξε κάποια στιγμή στην εξέλιξη του γράφου OEM. Οι εκφάνσεις της  $e$  αντιστοιχούν σε αλλαγές του OEM που επηρέασαν τον κόμβο αυτό. Οι ακμές που δείχνουν στο  $m$  αντιστοιχούν σε ακμές που έδειχναν στον κόμβο αυτό κάποια στιγμή στην εξέλιξη του γράφου OEM. Επιπρόσθετα, το κληρονομούμενο περιβάλλον των ακμών που δείχνουν στο  $m$  θα είναι τέτοιο ώστε να επιτρέπει σε κάθε ένα από τα  $e_1, e_2, \dots, e_n$  να «επιβιώνουν» κάτω από κάποιον κόσμο. Άρα, για κάθε  $e_i$  με  $2 \leq i \leq n-1$  το ρητό περιβάλλον  $c_i$  είναι επίσης και το κληρονομούμενο περιβάλλον του κόμβου περιβάλλοντος  $e_i$ . Όπως έχουμε δει,  $c_1 = [d \text{ in } \{\text{start}..somevalue\_I\}]$ , και  $c_n = [d \text{ in } \{somevalue\_N..now\}]$ : για τις εκφάνσεις  $e_1$  και  $e_n$  οι εισερχόμενες ακμές περιορίζουν τα ρητά περιβάλλοντα, κι έτσι το κληρονομούμενο περιβάλλον του  $e_1$  μπορεί να έχει πρώτη τιμή μεγαλύτερη από το `start`, ενώ το κληρονομούμενο περιβάλλον του  $e_n$  μπορεί να έχει δεύτερη τιμή μικρότερη από το `now`.

Τώρα είναι εύκολο να δει κανείς ότι δεν μπορεί να υπάρχουν στοιχεία στον  $G$  με κενή κληρονομούμενη κάλυψη, και, εφόσον ο  $G$  είναι αιτιοκρατικός ως προς το περιβάλλον, ο  $G$  είναι ένα MOEM. Επιπλέον, η κληρονομούμενη κάλυψη της ρίζας του  $G$  είναι ένα καθολικό

<sup>101</sup> Αυτό ισχύει εφόσον όλες οι διαδοχικές καταστάσεις στο ιστορικό της βάσης δεδομένων είναι μη κενά OEMs των οποίων τα φύλλα είναι ατομικοί κόμβοι.

περιβάλλον, πράγμα που σημαίνει ότι για κάθε χρονική στιγμή  $t$  μέσα στο διάστημα  $\{start..now\}$  μπορούμε να ανάγουμε τον  $G$  σε ένα OEM που υφίσταται κάτω από το  $t$ .

Συνεπώς, δεδομένης μιας βάσης OEM  $O$  και μιας βάσης MOEM  $M$  που αναπαριστά το ιστορικό της  $O$ , είναι δυνατόν για μια οποιαδήποτε χρονική στιγμή  $t$  από το πεδίο ορισμού του χρόνου  $T$  να ανάγουμε το  $M$  σε μια OEM βάση  $O'$  που υφίσταται κάτω από το  $t$ . Τότε η  $O'$  θα είναι το στιγμιότυπο της  $O$  κατά την χρονική στιγμή  $t$ . Είναι εξίσου δυνατό να εφαρμόσουμε μερική αναγωγή στον  $M$  για ένα σύνολο χρονικών στιγμών  $[d \text{ in } \{t_1, t_2, \dots, t_n\}]$ . Στην περίπτωση αυτή θα επιστραφεί ένας MOEM υπογράφος του  $M$ , που θα ενσωματώνει όλα τα στιγμιότυπα OEM κατά τις χρονικές στιγμές  $t_1, t_2, \dots, t_n$ .

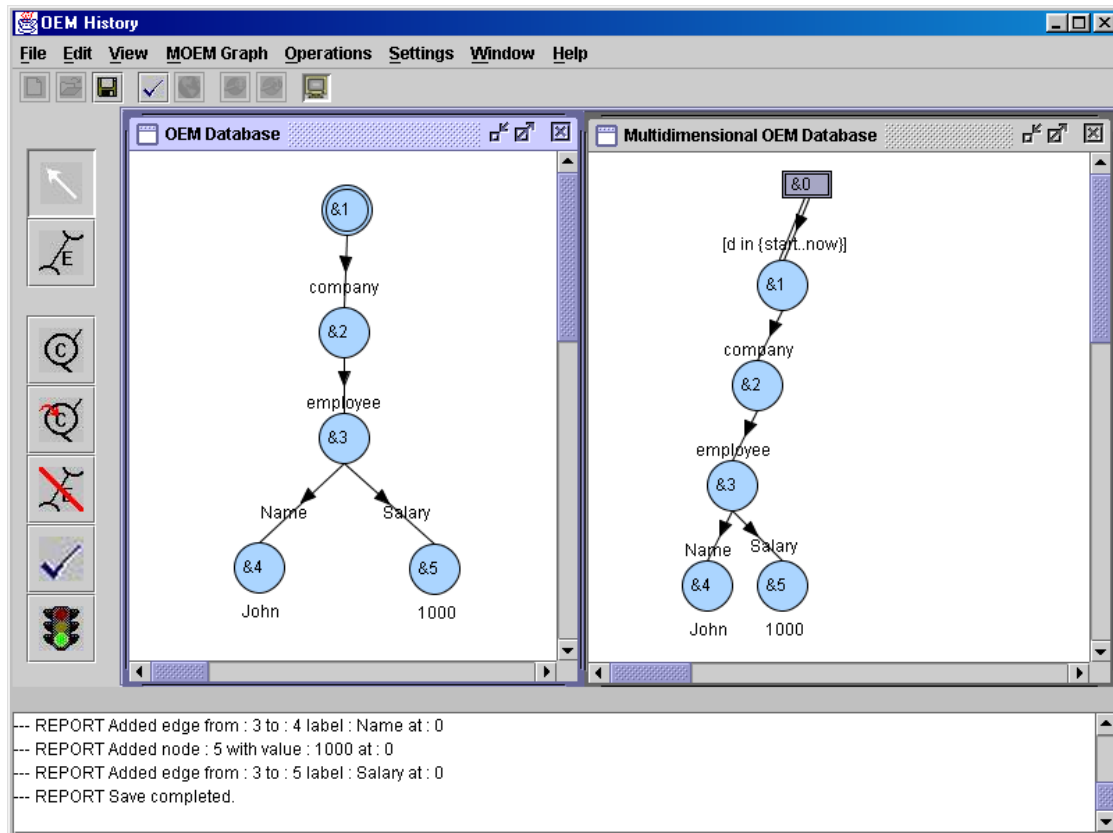
### 6.3 ΥΛΟΠΟΙΗΣΗ: “OEM HISTORY”

Η *OEM History* είναι μια εφαρμογή αναπτυγμένη σε Java [JAVA, CL97], η οποία υλοποιεί την μέθοδο που περιγράφηκε στην Ενότητα 6.2 για την αναπαράσταση ιστορικών του OEM. Όπως φαίνεται στην Εικόνα 6.2, η *OEM History* χρησιμοποιεί διαπροσωπεία πολλαπλών εγγράφων (multi-document interface – MDI) όπου κάθε εσωτερικό παράθυρο παρουσιάζει έναν γράφο δεδομένων. Υπάρχουν δυο κύρια παράθυρα: το ένα εμφανίζει έναν γράφο MOEM που αντιστοιχεί στο εσωτερικό μοντέλο της εφαρμογής, και ένα που δείχνει πάντα την τρέχουσα κατάσταση της βάσης OEM. Επιπλέον, ο χρήστης μπορεί να ζητήσει ένα στιγμιότυπο της βάσης για κάθε χρονική στιγμή στο  $T$  (το πεδίο ορισμού του χρόνου), το οποίο θα παρουσιαστεί σαν ένας γράφος OEM σε ξεχωριστό παράθυρο. Η εργαλειοθήκη στο αριστερό μέρος περιέχει κουμπιά που αντιστοιχούν στις τέσσερις λειτουργίες βασικών αλλαγών του OEM, και μπορούν να χρησιμοποιηθούν μόνον πάνω στο παράθυρο με το OEM που απεικονίζει την τρέχουσα κατάσταση της βάσης. Στην πραγματικότητα, τα κουμπιά αυτά καλούν λειτουργίες που ενημερώνουν το εσωτερικό μοντέλο δεδομένων MOEM<sup>102</sup> της εφαρμογής, που είναι και το μόνο μοντέλο που διατηρεί το *OEM History*. Η τρέχουσα βάση OEM είναι το αποτέλεσμα αναγωγής σε OEM του MOEM κάτω από τον κόσμο όπου η διάσταση  $d$  έχει την τιμή  $now$ .

Το κουμπί «tick» στην αριστερή εργαλειοθήκη απομακρύνει κόμβους που δεν είναι προσβάσιμοι από την ρίζα. Το τελευταίο κουμπί στην εργαλειοθήκη σημειώνει το τέλος μιας ακολουθίας λειτουργιών βασικών αλλαγών, και εκτελεί όλες τις αλλαγές στην βάση κάτω από μια κοινή χρονοσφραγίδα. Λειτουργίες όπως η αναγωγή σε OEM του MOEM μπορούν να κληθούν από την εργαλειοθήκη στο πάνω μέρος ή από το μενού της εφαρμογής.

<sup>102</sup> Όπως ορίστηκε στην Ενότητα 6.2.1.

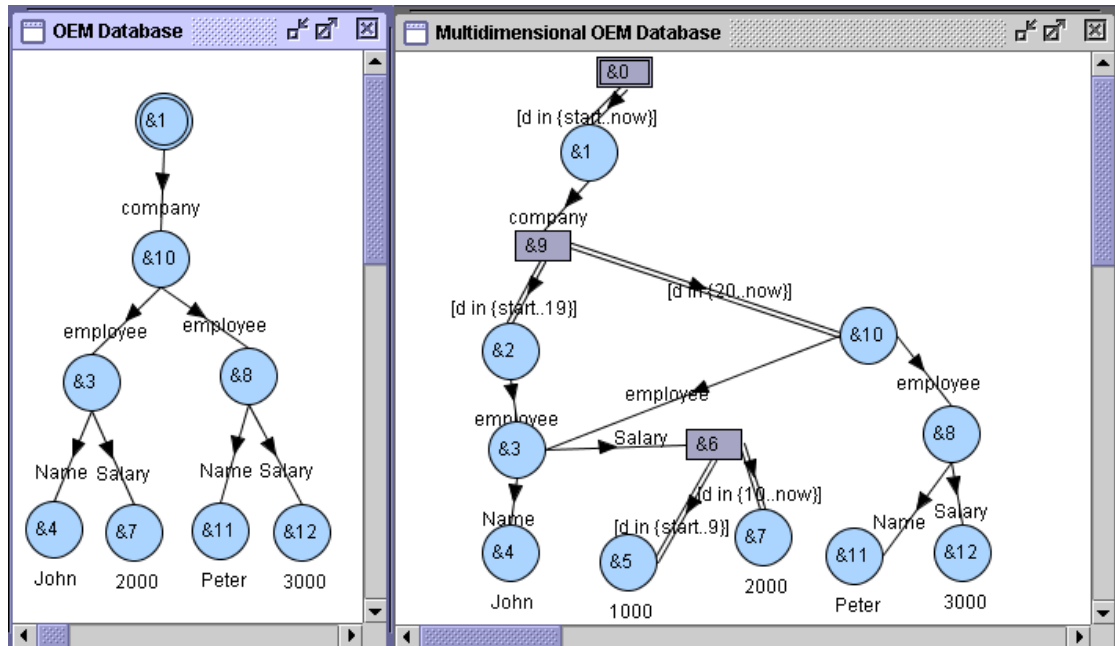
**Εικόνα 6.2:** Αρχική κατάσταση της βάσης του παραδείγματος στην εφαρμογή *OEM History*.



Στην Εικόνα 6.2 βλέπουμε την αρχική κατάσταση μιας βάσης OEM που περιέχει πληροφορία για τους υπαλλήλους μιας εταιρίας, μαζί με τον αντίστοιχο γράφο MOEM. Το δεξί παράθυρο εμφανίζει το εσωτερικό μοντέλο MOEM, ενώ το αριστερό παράθυρο εμφανίζει το αποτέλεσμα μιας αναγωγής του MOEM για  $d=now$ .

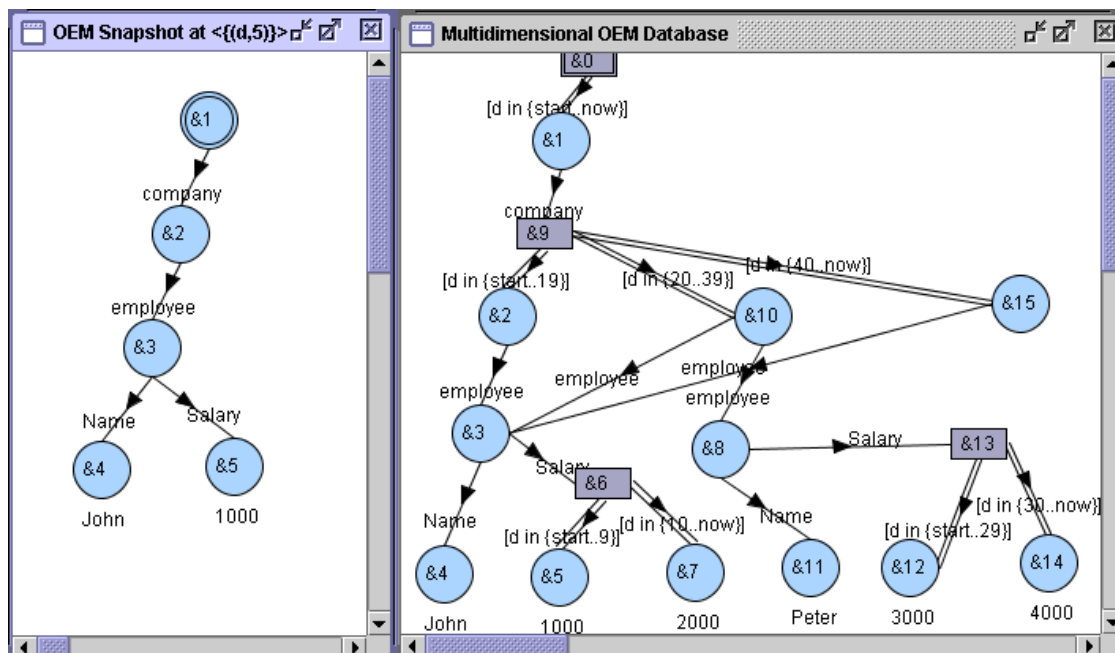


Εικόνα 6.3: Η βάση του παραδείγματος μετά από δύο ακολουθίες βασικών αλλαγών πάνω στην αρχική κατάσταση της βάσης.



Η Εικόνα 6.3 δείχνει την τρέχουσα κατάσταση της βάσης OEM και τον αντίστοιχο γράφο MOEM μετά από δυο ακολουθίες αλλαγών. Καταρχήν, την χρονική στιγμή 10 ο μισθός του John αυξήθηκε από 1000 σε 2000. Στην συνέχεια, κατά την χρονική στιγμή 20 ένας νέος υπάλληλος με το όνομα Peter προσελήφθη στην εταιρία με μισθό 3000.

Εικόνα 6.4: Η βάση του παραδείγματος μετά από ακόμη δύο ακολουθίες βασικών αλλαγών πάνω στην βάση της Εικόνα 6.3.



Στην Εικόνα 6.4 έχουν συμβεί δυο ακόμη ακολουθίες αλλαγών. Ο μισθός του Peter αυξήθηκε σε 4000 την χρονική στιγμή 30, αλλά εντελώς αγάριστα ο Peter έφυγε από την εταιρία την χρονική στιγμή 40. Σημειώστε ότι, όπως φαίνεται στην επιγραφή του παραθύρου, το αριστερό παράθυρο δεν εμφανίζει το τρέχον OEM. Αντί για αυτό απεικονίζει το στιγμιότυπο της βάσης OEM για την χρονική στιγμή 5, που αποκτάται μέσω αναγωγής σε OEM του MOEM στο δεξί παράθυρο για  $d=5$ . Αυτό το στιγμιότυπο είναι ταυτόσημο με την αρχική κατάσταση της βάσης, αφού η πρώτη αλλαγή έλαβε χώρα την χρονική στιγμή 10.

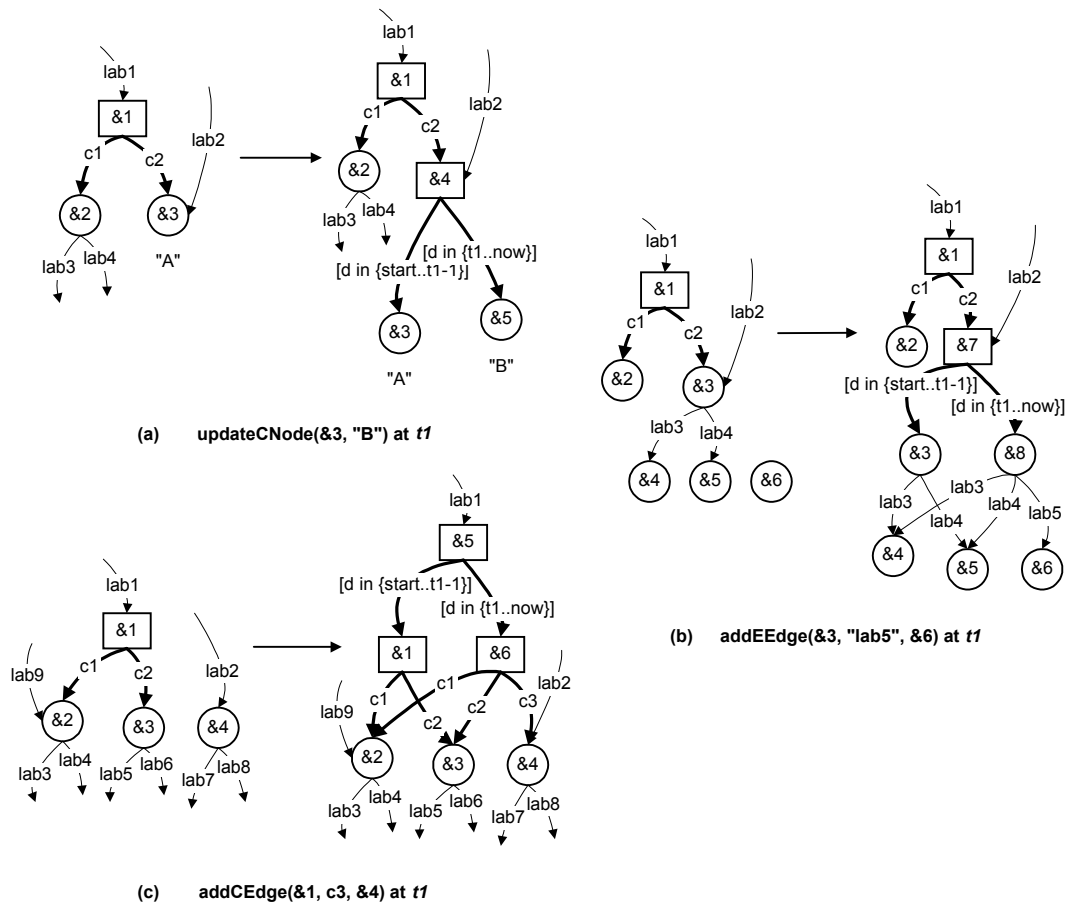
## 6.4 ΑΝΑΠΑΡΑΣΤΑΣΗ ΑΛΛΑΓΩΝ ΣΕ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ MOEM

Εκτός από την αναπαράσταση ιστορικών των βάσεων OEM, το MOEM έχει άλλη μια ενδιαφέρουσα ιδιότητα. Στην ενότητα αυτή δείχνουμε ότι το MOEM είναι αρκετά εκφραστικό για να *μοντελοποιήσει τα ίδια του τα ιστορικά*. Με άλλα λόγια, για κάθε βάση MOEM  $M$  που εξελίσσεται στον χρόνο είναι δυνατόν να διατηρούμε μια MOEM βάση  $M'$ , τέτοια ώστε η  $M'$  να αναπαριστά το ιστορικό της  $M$ <sup>103</sup>.

Η προσέγγιση είναι παρόμοια με αυτή στην Ενότητα 6.2.1· δείχνουμε ότι καθεμία από τις λειτουργίες βασικών αλλαγών του MOEM που εφαρμόζονται στο  $M$ , μπορεί να αντιστοιχιστεί σε έναν αριθμό λειτουργιών βασικών αλλαγών του MOEM στο  $M'$ , με τέτοιο τρόπο ώστε το  $M'$  να αναπαριστά το ιστορικό του  $M$ . Η Εικόνα 6.5 δίνει μια διαίσθηση αυτής της αντιστοιχίας, για τρεις βασικές λειτουργίες. Οι ετικέτες των ακμών περιβάλλοντος  $c_1, c_2, \dots, c_N$  δηλώνουν προσδιοριστές περιβάλλοντος που περιέχουν οποιονδήποτε αριθμό διαστάσεων, ενώ η διάσταση  $d$  έχει οριστεί στην Ενότητα 6.2.1. Σημειώστε ότι η χρήση της διάστασης  $d$  στο  $M'$  δεν εμποδίζει το  $M$  από το να χρησιμοποιήσει άλλες διαστάσεις που έχουν σαν πεδία ορισμού τον χρόνο. Οι λειτουργίες του MOEM που απεικονίζονται στην Εικόνα 6.5 είναι βασικές λειτουργίες που λαμβάνουν χώρα στο  $M$ , και οι αντίστοιχοι γράφοι δείχνουν πως αυτές οι λειτουργίες τροποποιούν το  $M'$ . Για απλότητα, οι γράφοι στην αριστερή πλευρά δεν περιέχουν προσδιοριστές περιβάλλοντος με την διάσταση  $d$ , και όλες οι χρονοσφραγίδες είναι  $t_1$ . Είναι πάντως εύκολο να φανταστούμε την περίπτωση όπου και η  $d$  είναι στην αριστερή πλευρά ενώ η τιμή των χρονοσφραγίδων αυξάνει σταδιακά, εάν κοιτάξουμε την Εικόνα 6.1 (b) και την Εικόνα 6.1 (c) που απεικονίζουν μια ανάλογη διαδικασία.

<sup>103</sup> Γενικά, ένας Πολυδιάστατος Γράφος Δεδομένων  $G'$  μπορεί να κατασκευαστεί έτσι ώστε να αναπαριστά τις αλλαγές που λαμβάνουν χώρα σε έναν άλλο Πολυδιάστατο Γράφο Δεδομένων  $G$ . Όπως στην Ενότητα 6.2.2 για το OEM, μπορεί να δειχθεί ότι ο  $G'$  είναι ένας MOEM εφόσον όλες οι διαδοχικές καταστάσεις του ιστορικού του  $G$  είναι μη κενά MOEMs.

Εικόνα 6.5: Μοντελοποίηση λειτουργιών βασικών αλλαγών MOEM με MOEM.

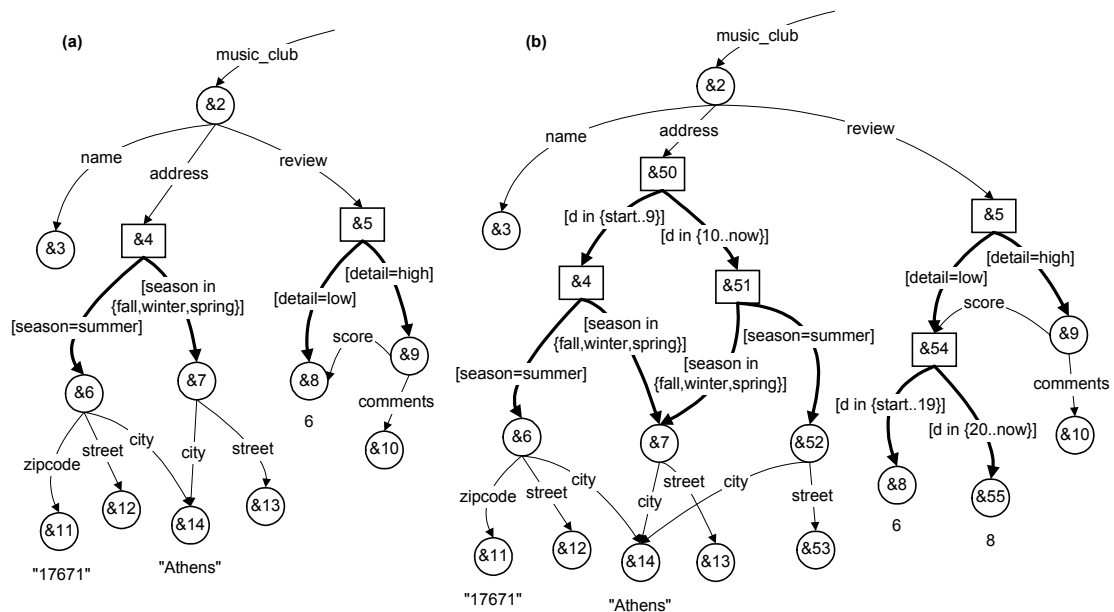


Η Εικόνα 6.5 (a) δείχνει μια έκφανση με αναγνωριστικό (oid) &3 της οποίας η τιμή αλλάζει από "A" σε "B" μέσω μιας κλήσης στην `updateCNode`. Η Εικόνα 6.5 (b) δείχνει το αποτέλεσμα της λειτουργίας `addEdge`. Τέλος, η Εικόνα 6.5 (c) απεικονίζει την βασική λειτουργία `addCEdge`. Από τις βασικές λειτουργίες του MOEM που δεν φαίνονται στην Εικόνα 6.5, η `remEdge` είναι παρόμοια με την `addEdge`: η διαφορά είναι ότι μια ακμή οντότητας απομακρύνεται από την έκφανση &8 αντί να προστεθεί. Επιπλέον, η `remCEdge` είναι παρόμοια με την `addCEdge`: αντί να προστεθεί μια ακμή περιβάλλοντος στον &6, διαγράφεται μια. Τέλος, οι βασικές λειτουργίες του MOEM `createCNode` και `createMNode` αντιστοιχίζονται στους εαυτούς τους: ο  $M'$  θα καταγράψει την αλλαγή όταν οι νέοι κόμβοι συνδεθούν με τον υπόλοιπο γράφο  $M$ , μέσω κλήσεων στην `addEdge` ή στην `addCEdge`.

Ένας γράφος MOEM  $M'$  κατασκευασμένος με την διεργασία που περιγράφηκε παραπάνω αναπαριστά το ιστορικό της MOEM βάσης  $M$ . Σε αντίθεση με την περίπτωση των ιστορικών του OEM όπου ένας κόσμος ορίζεται δίνοντας μια τιμή σε μια μοναδική διάσταση  $d$  που αντιστοιχεί στον χρόνο  $time$ , στην περίπτωση των ιστορικών του MOEM ένας κόσμος για το  $M'$  γενικά εμπλέκει περισσότερες από μια διαστάσεις, συμπεριλαμβανομένης της διάστασης χρόνου  $d$ . Συνεπώς, με το να προσδιορίσουμε με χρονική στιγμή  $t$  στην πραγματικότητα ορίζουμε το σύνολο κόσμων για τους οποίους  $d=t$ . Στο σύνολο αυτό, διαστάσεις εκτός του  $d$  μπορούν να έχουν οποιονδήποτε συνδυασμό τιμών από τα αντίστοιχα πεδία ορισμού τους.

Για να πάρουμε ένα χρονικό στιγμιότυπο του  $M$  από το  $M'$ , χρησιμοποιούμε την μερική αναγωγή. Εφαρμόζοντας την διαδικασία της μερικής αναγωγής στο  $M'$  για οποιαδήποτε χρονική στιγμή  $t \in T$ , παίρνουμε το MOEM στιγμιότυπο της βάσης  $M$  κατά την  $t$ <sup>104</sup>. Είναι επίσης δυνατό να εφαρμόσουμε μερική αναγωγή στο  $M'$  για ένα σύνολο χρονικών στιγμών  $[d \text{ in } \{t_1, t_2, \dots, t_n\}]$ . Στην περίπτωση αυτή επιστρέφεται ένας MOEM υπογράφοις του  $M'$ , ο οποίος ενσωματώνει όλα τα MOEM στιγμιότυπα κατά τις  $t_1, t_2, \dots, t_n$ .

**Εικόνα 6.6: Αναπαράσταση του ιστορικού του πολυδιάστατου music club στο (a) μέσω του MOEM στο (b).**



Σαν παράδειγμα, θεωρήστε τον γράφο MOEM στην Εικόνα 6.6 (b), που αναπαριστά το ιστορικό του πολυδιάστατου «music club» στην Εικόνα 6.6 (a). Η αρχική κατάσταση της βάσης όπως απεικονίζεται στην Εικόνα 6.6 (a) δεν περιέχει τους κόμβους &50, &51, &52, &53, &54, και &55 καθώς και τις απερχόμενες ακμές τους, και που εισάγονται στην Εικόνα 6.6 (b) εξαιτίας των λειτουργιών αλλαγών. Δυο ακολουθίες αλλαγών έχουν λάβει χώρα: (1) την χρονική στιγμή 10 η καλοκαιρινή διεύθυνση του club άλλαξε σε μια άλλη διεύθυνση με διαφορετικό street και χωρίς zipcode, και (2) την χρονική στιγμή 20 η βαθμολογία της κριτικής (review score) άλλαξε από 6 σε 8. Η πρώτη ακολουθία αλλαγών συνεπάγεται την αλλαγή κατεύθυνσης μιας ακμής περιβάλλοντος, ώστε να δείχνει στον νέο κόμβο &52 αντί για τον κόμβο &6. Η αλλαγή κατεύθυνσης επιτυγχάνεται μέσω μιας κλήσης στην `remCEdge` και μιας ακόλουθης κλήσης στην `addCEdge`, που εξηγούνται στην Εικόνα 6.5 (c). Η δεύτερη ακολουθία αλλαγών είναι στην πραγματικότητα μια κλήση στην `updateCNode` για τον κόμβο &8, που εξηγείται στην Εικόνα 6.5 (a).

<sup>104</sup> Για να είμαστε αυστηροί, ένα ακόμη βήμα πρέπει να γίνει προκειμένου να πάρουμε το χρονικό στιγμιότυπο μιας βάσης: οι ακμές περιβάλλοντος με ρητό περιβάλλον που αφορά στην διάσταση  $d$  δεν είναι πια σημαντικές και πρέπει να απομακρυνθούν. Το βήμα αυτό μπορεί να θεωρηθεί σαν παρόμοιο με την **απο-περιβαλλοντοποίηση**, που ορίζεται στο Κεφάλαιο 4.

## 6.5 ΕΠΕΡΩΤΗΣΕΙΣ ΑΛΛΑΓΩΝ ΜΕ ΤΗΝ MQL

Η προσέγγιση που παρουσιάσαμε παραπάνω διατηρεί διαφορετικές «εκδοχές» των αντικειμένων καθώς αυτά αλλάζουν, ενσωματώνοντας τα ιστορικά τους μέσα στις αντίστοιχες πολυδιάστατες οντότητες σαν μια σειρά στιγμιότυπων αντικειμένων. Είναι ενδιαφέρον να παρατηρήσει κανείς ότι μια απλούστερη αναπαράσταση θα μπορούσε να έχει επιτρέψει επίσης την αναγωγή της βάσης σε διάφορα χρονικά στιγμιότυπα: ακμές περιβάλλοντος που ξεκινούν από μια πολυδιάστατη ρίζα οδηγούν σε γράφους όλων των καταστάσεων από τις οποίες πέρασε η βάση, έχοντας σαν ετικέτα την περίοδο κατά την οποία η αντίστοιχη κατάσταση της βάσης ήταν έγκυρη. Στην πράξη, η μέθοδος αυτή κρατά ένα αντίγραφο ολόκληρης της βάσης μετά από κάθε ακολουθία λειτουργιών βασικών αλλαγών, και εμφανώς σπάταλα χώρο, αλλά επιτρέπει την αναγωγή με άμεσο τρόπο. Ένα σημαντικό πλεονέκτημα της αναπαράστασης αλλαγών στο επίπεδο των πολυδιάστατων οντοτήτων και όχι στο επίπεδο μιας συλλογής στιγμιότυπων ολόκληρης της βάσης προέρχεται από τις επερωτήσεις που μπορούμε να σχηματίσουμε για να ανακτήσουμε πληροφορία σχετικά με την εξέλιξη των οντοτήτων αυτών.

Στην ενότητα αυτή δίνουμε παραδείγματα τέτοιων επερωτήσεων, και επιδεικνύουμε πως η MQL μπορεί να χρησιμοποιηθεί στα πλαίσια του παρόντος προβλήματος. Αρχίζουμε με επερωτήσεις πάνω στο ιστορικό βάσεων OEM, και συνεχίζουμε με επερωτήσεις πάνω στο ιστορικό βάσεων MOEM.

### 6.5.1 Επερωτήσεις ιστορικών του OEM

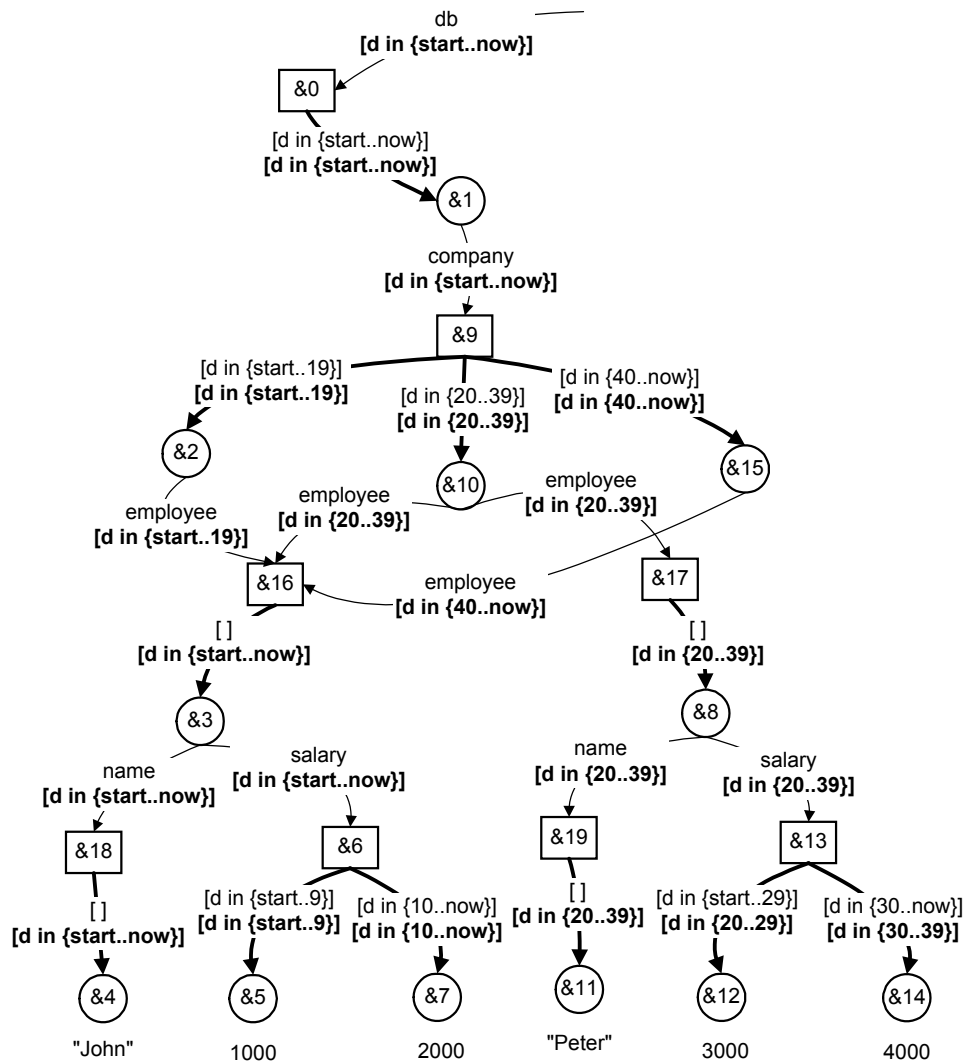
Για τα παραδείγματα των επερωτήσεών μας, θα χρησιμοποιήσουμε την MOEM βάση που απεικονίζεται στην Εικόνα 6.4. Οι εκφράσεις μονοπατιού περιβάλλοντος στις επερωτήσεις MQL αποτιμώνται με βάση την κανονική μορφή της MOEM βάσης, και χρησιμοποιούν συχνά την κληρονομούμενη κάλυψη μονοπατιού, η οποία είναι ένα ισχυρό χαρακτηριστικό των Πολυδιάστατων Γράφων Δεδομένων. Για να διευκολυνθεί η κατανόηση των επερωτήσεων, η Εικόνα 6.7 απεικονίζει την βάση στην Εικόνα 6.4 αφού έχει μετασχηματιστεί σε κανονική μορφή και συμπληρωμένη με τις κληρονομούμενες καλύψεις των ακμών, οι οποίες εμφανίζονται με έντονους χαρακτήρες. Όπως φαίνεται στην Εικόνα 6.7, ο μετασχηματισμός σε κανονική μορφή εισήγαγε τους πολυδιάστατους κόμβους &16, &17, &18, και &19.

Είναι σημαντικό να σημειώσουμε ότι ο μετασχηματισμός της υφισταμένης MOEM βάσης σε κανονική μορφή μπορεί να λάβει χώρα σε οποιοδήποτε στάδιο κατά την εξέλιξη της OEM βάσης. Η διεργασία που ορίστηκε στην Ενότητα 6.2.1 για την μοντελοποίηση ιστορικών των OEM μπορεί να εφαρμοστεί το ίδιο είτε η υφισταμένη MOEM βάση είναι σε κανονική μορφή είτε όχι, με μια μοναδική εξαίρεση: εάν το MOEM είναι σε κανονική μορφή, πρέπει να διατηρήσουμε αυτή την κανονική μορφή όταν δημιουργούνται νέοι OEM κόμβοι. Για να καλυφθεί αυτή την εξαίρεση, πρέπει να ενσωματωθεί μια απλή επέκταση: όποτε δημιουργείται ένας νέος κόμβος περιβάλλοντος σαν αποτέλεσμα της `createNode(p, val)`, δημιουργούνται επίσης ένας πολυδιάστατος κόμβος και μια ακμή περιβάλλοντος που δείχνει στον κόμβο περιβάλλοντος.

Στην πράξη, αν η βάση MOEM βρίσκεται σε κανονική μορφή, τότε η διεργασία στην Ενότητα 6.2.1 είναι σημαντικά απλούστερη. Συγκεκριμένα, δεν θα είναι πια απαραίτητο να καλούμε την διαδικασία `mEntity(id)` για να δημιουργείται ένας πολυδιάστατος κόμβος,

επειδή ο μετασχηματισμός σε κανονική μορφή θα έχει εισάγει με μιας όλους τους πολυδιάστατους κόμβους που είναι απαραίτητοι<sup>105</sup>.

**Εικόνα 6.7: Η MOEM βάση της Εικόνα 6.4 σε κανονική μορφή, συμπληρωμένη με την κληρονομούμενη κάλυψη των ακμών.**



Οι πολυδιάστατες οντότητες της βάσης δεδομένων στην Εικόνα 6.7 μοντελοποιούν αντικείμενα OEM που ίσως έχουν υποστεί αλλαγές, και οι MQL ερωτήσεις πάνω στις οντότητες αυτές μπορούν να ερμηνευτούν σαν ερωτήσεις στο ιστορικό των αντίστοιχων αντικειμένων OEM.

<sup>105</sup> Ένα έλασσον θέμα είναι ότι οι ακμές περιβάλλοντος που έχουν δημιουργηθεί από τον μετασχηματισμό σε κανονική μορφή έχουν ρητό περιβάλλον το καθολικό περιβάλλον [], ενώ οι ακμές περιβάλλοντος που δημιουργήθηκαν από την `mEntity(id)` έχουν ρητό περιβάλλον `[d in {start..now}]`, το οποίο στην περίπτωση μας είναι επίσης ένα καθολικό περιβάλλον που αντιπροσωπεύει κάθε πιθανό κόσμο.

- Είναι απλό να βρεθεί η τιμή που είχε ένα αντικείμενο OEM σε μια συγκεκριμένη χρονική στιγμή.

Η επερώτηση που ακολουθεί επιστρέφει τον μισθό του Peter κατά την χρονική στιγμή 32:

```
select salary: S
from [d=32]db.company.employee{X}.salary S
where X.name = "Peter"
```

Το αποτέλεσμα της επερώτησης είναι η mssd-expression {salary: &14 4000}. Η έκφραση μονοπατιού περιβάλλοντος [d=32]db.company.employee.salary είναι ισοδύναμη με την συμβατική έκφραση μονοπατιού db.company.employee.salary αποτιμώμενη πάνω στο συμβατικό OEM που υφίσταται κάτω από την χρονική στιγμή 32. Συνεπώς, η μεταβλητή αντικειμένου S δεσμεύεται στην έκφραση του salary που υπάρχει σε αυτό το OEM. Το ίδιο αποτέλεσμα θα είχε επιστραφεί εάν, για παράδειγμα, είχαμε χρησιμοποιήσει [d in {32..36}] αντί για [d=32], επειδή το μονοπάτι που οδηγεί στον κόμβο &14 υφίσταται κάτω από όλες τις χρονικές στιγμές ανάμεσα από 32 και 36. Πάντως, αν είχαμε χρησιμοποιήσει [d in {25..50}], δεν θα επιστρέφονταν κανένα αντικείμενο salary, επειδή καμία μοναδική έκφραση του salary δεν υφίσταται κάτω από την χρονική περίοδο 25 μέχρι 50.

Μια παρόμοια επερώτηση επιστρέφει τα ονόματα των υπαλλήλων των οποίων ο μισθός κατά την χρονική στιγμή 25 ήταν μεγαλύτερος ή ίσος με 2500:

```
select name: Y
from [d=25]db.company.employee{X}.salary S,
     X.name Y
where S >= 2500
```

Το αποτέλεσμα είναι {name: &11 "Peter"}.

- Μπορούμε να ανακτήσουμε πληροφορία για το ιστορικό ενός αντικειμένου OEM.

Η παρακάτω επερώτηση επιστρέφει τους μισθούς του Peter, μαζί με τις περιόδους κατά τις οποίες ίσχυαν:

```
select [Y]: S
from db.company.employee{X}.salary::[Y] [-] S
where X.name = "Peter"
```

Η μεταβλητή περιβάλλοντος [Y] δεσμεύεται στην κληρονομούμενη κάλυψη των ακμών περιβάλλοντος που οδηγούν σε εκφάνσεις του salary, και χρησιμοποιείται για να την μετατρέψει σε ρητό περιβάλλον στο αποτέλεσμα:

```
<[d in {20..29}]: &12 3000,
 [d in {30..39}]: &14 4000>
```

Ο γράφος του αποτελέσματος αποτελείται από μια πολυδιάστατη ρίζα από την οποία ξεκινούν δυο ακμές περιβάλλοντος, που οδηγούν στους κόμβους περιβάλλοντος &12 και &14. Παρατηρήστε ότι η κληρονομούμενη κάλυψη στην βάση δεδομένων της ακμής περιβάλλοντος που οδηγεί στον κόμβο &12 είναι [d in {20..29}], και αντικατοπτρίζει το γεγονός ότι ο Peter προσελήφθη στην εταιρία την χρονική στιγμή 20. Παρομοίως, η κληρονομούμενη κάλυψη στην βάση δεδομένων της ακμής περιβάλλοντος που οδηγεί στον κόμβο &14 αντικατοπτρίζει το γεγονός ότι ο Peter άφησε την εταιρία την χρονική στιγμή 40. Και οι δυο αυτές κληρονομούμενες καλύψεις εμφανίζονται στο αποτέλεσμα σαν ρητά περιβάλλοντα.

Η ακόλουθη επερώτηση ανακτά τις χρονικές στιγμές στις οποίες ο μισθός του Peter υπέστη κάποια αλλαγή:

```
select period_without_change: [Y]
from db.company.employee{X}.salary:: [Y] [-]
where X.name = "Peter"
```

Το αποτέλεσμα είναι:

```
{period_without_change: "[d in {20..29}]",
 period_without_change: "[d in {30..39}]"}
```

Το αποτέλεσμα δείχνει ότι την χρονική στιγμή 30 μια αλλαγή συνέβη στον μισθό του Peter. Παρατηρείστε πως η μεταβλητή περιβάλλοντος [Y] γίνεται τιμή στοιχειοσειράς νέων ατομικών κόμβων στο αποτέλεσμα.

Η μικρότερη και η μεγαλύτερη τιμή (20 και 39) του d στο παραπάνω αποτέλεσμα δείχνουν την διάρκεια ζωής της πολυδιάστατης οντότητας salary, που στην συγκεκριμένη περίπτωση τυγχάνει να είναι ένα συνεχές διάστημα. Είναι δυνατόν να ανακτήσουμε άμεσα την διάρκεια ζωής μιας πολυδιάστατης οντότητας, όπως φαίνεται στην επερώτηση που ακολουθεί:

```
select lifespan: [Z]
context [Z] as union([Y])
from db.company.employee{X}.salary:: [Y] [-]
where X.name = "Peter"
```

Ο όρος context χρησιμοποιείται για να εισάγει την νέα μεταβλητή περιβάλλοντος [Z] σαν την ένωση περιβάλλοντος όλων των τιμών που δεσμεύθηκαν στην [Y]. Το αποτέλεσμα είναι: {lifespan: "[d in {20..39}]"}

- Μπορούμε να ανακτήσουμε τις τιμές ενός αντικειμένου OEM οι οποίες ίσχυσαν οποτεδήποτε μέσα σε μια περίοδο.

Η ακόλουθη επερώτηση επιστρέφει τους μισθούς του Peter την περίοδο από 25 έως 50.

```
select [Y]: S
from [Y]db.company.employee{X}.salary S
where X.name = "Peter"
within [Y] * [d in {25..50}] != [-]
```

Για κάθε τιμή του s, η μεταβλητή περιβάλλοντος [Y] δεσμεύεται σε μια κληρονομούμενη κάλυψη μονοπατιού, που αντιπροσωπεύει τις χρονικές στιγμές κάτω από τις οποίες υφίστανται OEMs, που περιέχουν ένα μονοπάτι το οποίο οδηγεί στην τιμή αυτή του s. Τότε, η [Y] χρησιμοποιείται στον όρο within για να αποκλείσει εκφάνσεις του salary που δεν υφίστανται κάτω από καμία χρονική στιγμή ανάμεσα στο 25 και το 50. Η επερώτηση επιστρέφει:

```
<[d in {20..29}]: &12 3000,
 [d in {30..39}]: &14 4000>
```

Εάν είχαμε χρησιμοποιήσει μια χρονική στιγμή μεγαλύτερη από το 29 αντί για την χρονική στιγμή 25, μόνον ο κόμβος &14 θα είχε επιστραφεί. Για την περίοδο ανάμεσα στο 40 και το 50 δεν υπάρχουν μισθοί, επειδή την χρονική στιγμή 40 ο Peter έφυγε από την εταιρία. Σημειώστε την διαφορά με ένα προηγούμενο παράδειγμα που ψάχνει για έναν μισθό που υφίσταται κάτω από ολόκληρη την περίοδο 25 με 50.

- Μπορούμε να βρούμε πότε ένα αντικείμενο OEM είχε μια συγκεκριμένη τιμή.



Η παρακάτω επερώτηση επιστρέφει την περίοδο κατά την οποία ο μισθός του John ήταν 2000.

```
select valid_time: [Y]
from [Y]db.company.employee{X}.salary S
where X.name = "John" and S = 2000
```

Το αποτέλεσμα είναι:

```
{valid_time: "[d in {10..19}]",
 valid_time: "[d in {20..39}]",
 valid_time: "[d in {40..now}]" }
```

Η μεταβλητή περιβάλλοντος [Y] δεσμεύεται σε κληρονομούμενες καλύψεις *μονοπατιού*, και εφόσον υπάρχουν τρία μονοπάτια στην βάση που οδηγούν στον κόμβο &7, υπάρχουν τρία αντικείμενα στο αποτέλεσμα. Και τα τρία μαζί, δίνουν την περίοδο κατά τη οποία ο μισθός του John ήταν 2000. Αν θέλουμε ολόκληρη την περίοδο σαν ενιαίο αντικείμενο, μπορούμε να χρησιμοποιήσουμε έναν όρο `context` για να εισάγουμε μια νέα μεταβλητή σαν `union([Y])`. Εναλλακτικά, μπορούμε να χρησιμοποιήσουμε μιαν άλλη έκφραση *μονοπατιού περιβάλλοντος* στον όρο `from`:

```
select valid_time: [Y]
from db.company.employee{X}.salary::[Y] [-] S
where X.name = "John" and S = 2000
```

Στην περίπτωση αυτή, το [Y] αντιστοιχεί σε ένα μονοπάτι που αποτελείται από μια ακμή περιβάλλοντος η οποία οδηγεί σε μια έκφραση του `salary`, και το αποτέλεσμα θα είναι `{valid_time: "[d in {10..now}]"}`, που δίνει και πάλι την περίοδο κατά την οποία ο μισθός του John ήταν 2000.

- Είναι δυνατόν να τεθούν γενικές επερωτήσεις πάνω στο ιστορικό της βάσης OEM.

Η επερώτηση που ακολουθεί βρίσκει όλα τα αντικείμενα OEM που δεν έχουν υποστεί αλλαγές από την στιγμή της δημιουργίας τους:

```
select constant_object: {value: Z, period: [Y]}
from db.#.- <X>
where count(<X>::[Y] [-]{Z}) = 1
```

Η επερώτηση χρησιμοποιεί τον μπαλαντέρ # που ταιριάζει με μηδέν ή περισσότερα ζεύγη *ακμή οντότητας / ακμή περιβάλλοντος*, και τον μπαλαντέρ - που ταιριάζει με οποιαδήποτε ακμή οντότητας. Η μεταβλητή αντικειμένου <x> δεσμεύεται σε κάθε πολυδιάστατο αντικείμενο της βάσης, εκτός από αυτό που αντιστοιχεί στην ρίζα. Τότε, ο όρος `where` απορρίπτει τα πολυδιάστατα αντικείμενα που έχουν περισσότερες από μια εκφάνσεις. Θυμηθείτε ότι σε ένα MOEM που βρίσκεται σε κανονική μορφή, κάθε αντικείμενο περιβάλλοντος είναι η έκφραση κάποιου πολυδιάστατου αντικειμένου, και κάθε πολυδιάστατο αντικείμενο σχετίζεται με μια το λιγότερο έκφραση η οποία είναι αντικείμενο περιβάλλοντος.

Το ίδιο αποτέλεσμα επιστρέφεται από την επερώτηση

```
select constant_object: {value: Z, period: [Y]}
from db.#.-::[Y] [] Z
```

που εκμεταλλεύεται το γεγονός ότι εάν ένα αντικείμενο OEM δεν έχει υποστεί αλλαγές, τότε το αντίστοιχο αντικείμενο περιβάλλοντος του MOEM δείχνεται από μια ακμή περιβάλλοντος της οποίας το ρητό περιβάλλον είναι ένα καθολικό περιβάλλον.

Σαν επιπλέον παράδειγμα, η παρακάτω επερώτηση επιστρέφει τους υπάλληλους των οποίων ο μισθός δεν έχει αλλάξει από την χρονική στιγμή 32.

```
select employee_data: {name: Z, salary: S, valid_period: [Y]}
from db.#.employee{X}.salary::[Y] [-] S,
     X.name Z
within [Y] >= [d in {32..now}]
```

Το αποτέλεσμα είναι:

```
{employee_data: {name: &4 "John",
                  salary: &7 2000,
                  valid_period: "[d in {10..now}]"}}
}
```

Ο όρος `within` ελέγχει εάν η περίοδος κάτω από την οποία υφίσταται μια έκφραση του `salary` είναι υπερσύνολο περιβάλλοντος της περιόδου από το 32 μέχρι την παρούσα στιγμή. Σημειώστε ότι ο κόμβος `&14` δεν περιλαμβάνεται στα αποτελέσματα, επειδή η κληρονομούμενη κάλυψη υποδεικνύει σωστά το γεγονός ότι την χρονική στιγμή 40 ο μισθός έπαψε να υπάρχει.

- Μπορούμε να συνδυάσουμε τιμές και έγκυρα διαστήματα διαφορετικών αντικειμένων OEM για να σχηματίσουμε επερωτήσεις όπως «βρες τις τιμές ενός αντικειμένου κατά το διάστημα που ένα άλλο αντικείμενο είχε μια συγκεκριμένη τιμή».

Η παρακάτω επερώτηση επιστρέφει τους μισθούς του John, κατά το διάστημα που ο μισθός του Peter ήταν 3000:

```
select john_salary: S1
from db.company X,
     X.employee{Y1}.salary::[Z1] [-] S1,
     X.employee{Y2}.salary::[Z2] [-] S2
where Y1.name = "John" and Y2.name = "Peter" and S2 = 3000
within [Z1] * [Z2] != [-]
```

Το αποτέλεσμα της επερώτησης είναι `{john_salary: &7 2000}`.

- Είναι δυνατόν να «αποδομήσουμε» μια πολυδιάστατη οντότητα κατά την διάσταση `d`.

Η ακόλουθη επερώτηση δίνει τα στιγμιότυπα κατά χρόνο μιας εταιρίας με το όνομα «ACME»:

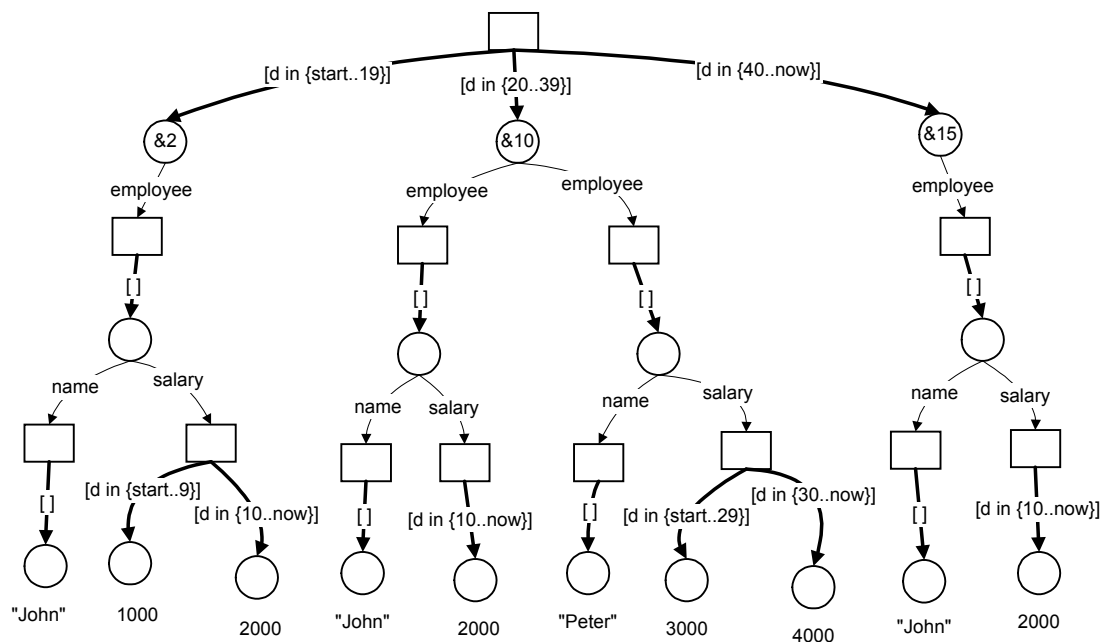
```
select holding <[Y]: Z>
from db.company{<X>}::[Y] [-] Z
where <X>.company_name = "ACME"
```

Η μεταβλητή αντικειμένου `<X>` δεσμεύεται σε ένα πολυδιάστατο αντικείμενο `company` για το οποίο υπάρχει κάποια έκφραση με το όνομα «ACME»<sup>106</sup>. Οι μεταβλητές `[Y]` και `Z` δεσμεύονται σε κληρονομούμενες καλύψεις και εκφάνσεις εταιριών αντίστοιχα, και χρησιμοποιούνται στο `select` για να κατασκευαστεί μια νέα πολυδιάστατη οντότητα, στην

<sup>106</sup> Με τον τρόπο αυτό, εκφάνσεις μιας εταιρίας που δεν δείχνουν σε ένα `company_name` με την τιμή «ACME» θα συμπεριλαμβάνονται εξίσου στα αποτελέσματα, εφόσον υπάρχει στην ίδια πολυδιάστατη οντότητα μια έκφραση που να δείχνει σε «ACME». Συγκρίνετε αυτή την περίπτωση με την περίπτωση όπου ο όρος `where` θα περιείχε: `Z . company_name = "ACME"`.

οποία οι κληρονομούμενες καλύψεις μεταβάλλονται σε ρητά περιβάλλοντα<sup>107</sup>. Η λέξη-κλειδί `holding` απομακρύνει από τον γράφο του αποτελέσματος τους κόμβους και τις ακμές που έχουν κενή κληρονομούμενη κάλυψη στον γράφο του αποτελέσματος. Στην πραγματικότητα, το `holding` προκαλεί την μερική αναγωγή των υπογράφων που είναι δεσμευμένοι στο `Z` για το αντίστοιχο περιβάλλον που είναι δεσμευμένο στο `[Y]`. Θεωρώντας ότι το όνομα της εταιρίας της βάσης στην Εικόνα 6.7 είναι «ACME», το τελικό αποτέλεσμα απεικονίζεται στην Εικόνα 6.8<sup>108</sup>.

**Εικόνα 6.8:** Μερικές αναγωγές για διάφορα ερμηνευτικά περιβάλλοντα σαν αποτέλεσμα επερώτησης MQL.



Ο γράφος στην Εικόνα 6.8 περιέχει τρεις υπογράφους, έναν για κάθε κατάσταση στην εξέλιξη της εταιρίας. Ο πρώτος υπογράφος έχει σαν ρίζα τον κόμβο &2, και αντιστοιχεί στην κατάσταση της εταιρίας από την αρχή του χρόνου μέχρι την χρονική στιγμή 19. Εκείνη την στιγμή, η εταιρία είχε μόνο έναν υπάλληλο με το όνομα «John». Ο δεύτερος υπογράφος έχει σαν ρίζα τον κόμβο &10, και αντιστοιχεί στην περίοδο από το 20 έως το 39. Κατά την διάρκεια αυτής της περιόδου, η εταιρία είχε δυο υπάλληλους, τον «John» και τον «Peter». Παρατηρήστε ότι ο «John» περιέχει μόνο έναν μισθό τώρα, μια και ο πρώτος του μισθός δεν έχει σχέση με την περίοδο αυτή. Ο τρίτος υπογράφος έχει σαν ρίζα τον κόμβο &15, και

<sup>107</sup> Στην βάση δεδομένων στην Εικόνα 6.7, οι κληρονομούμενες καλύψεις των εδρών που οδηγούν σε εκφάνσεις μιας εταιρίας τυγχάνει να είναι οι ίδιες με τα ρητά περιβάλλοντα.

<sup>108</sup> Όπως αναφέρθηκε στο προηγούμενο κεφάλαιο, για να διευκολύνουμε την κατανόηση των επερωτήσεων εμφανίζουμε στα αποτελέσματα των επερωτήσεων τα αναγνωριστικά (oids) των αντικειμένων όπως αυτά εμφανίζονται στην βάση. Ας θυμάται κανείς πάντως, ότι τα oids στα αποτελέσματα των επερωτήσεων δεν είναι αναγκαστικά τα ίδια με τα oids των αντίστοιχων αντικειμένων στην βάση δεδομένων. Αυτό φαίνεται καθαρά στην Εικόνα 6.8, όπου το ίδιο αντικείμενο της βάσης επαναλαμβάνεται περισσότερες από μια φορές στο αποτέλεσμα, και συνεπώς δεν μπορεί να έχει το ίδιο oid.

αντιστοιχεί στην περίοδο από το 40 μέχρι την τρέχουσα στιγμή. Κατά την διάρκεια αυτής της περιόδου, ο «John» είναι ο μόνος υπάλληλος της εταιρίας με μισθό που παραμένει ο ίδιος.

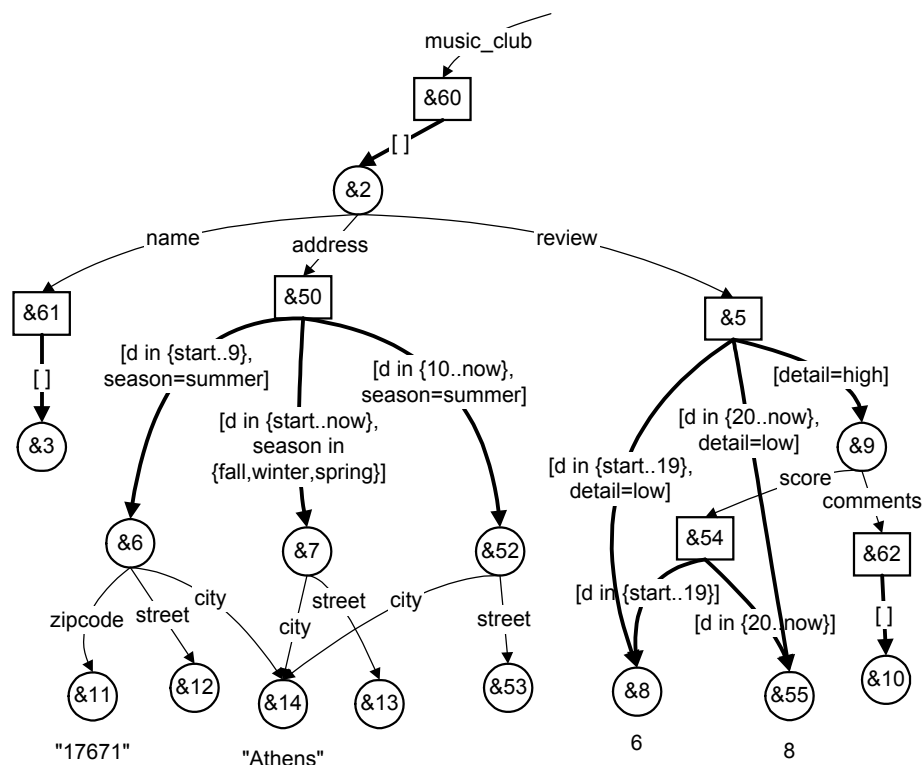
Παρατηρήστε ότι αν το περιβάλλον της αναγωγής γίνει μια χρονική στιγμή, τότε κάθε πολυδιάστατος κόμβος στον αντίστοιχο υπογράφο θα δείχνει σε έναν ακριβώς κόμβο περιβάλλοντος, και η μερική αναγωγή θα γίνει ανάλογη με την αναγωγή σε OEM κάτω από την συγκεκριμένη χρονική στιγμή.

## 6.5.2 Επερωτήσεις ιστορικών του MOEM

Εκτός από την επερώτηση ιστορικών του OEM, η MQL μπορεί να χρησιμοποιηθεί για να επερωτήσει ιστορικά των βάσεων MOEM. Στην ενότητα αυτή παρουσιάζουμε κάποια παραδείγματα επερωτήσεων πάνω σε ιστορικά του MOEM, επικεντρώνοντας στις διαφορές με την επερώτηση ιστορικών του OEM.

Για τα παραδείγματα των επερωτήσεων μας θα χρησιμοποιήσουμε το MOEM που απεικονίζεται στην Εικόνα 6.9, το οποίο στην πραγματικότητα είναι το MOEM στην Εικόνα 6.6 (b) αφού έχει μετασχηματιστεί σε κανονική μορφή. Παρατηρήστε ότι ο μετασχηματισμός σε κανονική μορφή έχει εισάγει τρεις νέους πολυδιάστατους κόμβους, τους &60, &61 και &62, και έχει απομακρύνει τους πολυδιάστατους κόμβους &4 και &51.

Εικόνα 6.9: Το MOEM της Εικόνα 6.6 (b) σε κανονική μορφή.



Η ακόλουθη «εισαγωγική» επερώτηση επιστρέφει την κριτική (review) του music club την χρονική στιγμή 15 σε χαμηλή λεπτομέρεια:

```
select review: X
from [d=15,detail=low]music_club.review X
```

Το αποτέλεσμα είναι η mssd-expression {review: &8 6}. Αυτή η απλή επερώτηση καθορίζει τιμές για όλες τις διαστάσεις που σχετίζονται με το review, και παίρνει μια έκφραση του review σαν αποτέλεσμα. Όμως, ίσως να θέλουμε να ορίσουμε περιορισμούς μόνον για μερικές από τις διαστάσεις. Όπως μπορούμε να δούμε στην Εικόνα 6.9, όταν αναπαριστούμε ιστορικά του MOEM φυσιολογικά η διάσταση d χρησιμοποιείται στους προσδιοριστές περιβάλλοντος μαζί με άλλες διαστάσεις. Συχνά χρειάζεται να «απομονώσουμε» την διάσταση αυτή στις επερωτήσεις και να θέσουμε περιορισμούς στην συγκεκριμένη διάσταση μόνο, ανεξάρτητα από τις τιμές των άλλων διαστάσεων. Ένας κομψός τρόπος για να το επιτύχουμε αυτό είναι η χρήση των **πρότυπων περιβάλλοντος**, τα οποία εισήχθησαν στο προηγούμενο κεφάλαιο. Η παρακάτω επερώτηση ανακτά τις διευθύνσεις του music club την χρονική στιγμή 15, ανεξάρτητα από τις τιμές των άλλων διαστάσεων:

```
select address: X
from [% d=15]music_club.address X
```

Το πρότυπο περιβάλλοντος [% d=15] ταιριάζει με τις κληρονομούμενες καλύψεις μονοπατιού που περιέχουν (κατ' ελάχιστο) έναν κόσμο όπου το d είναι 15, και η επερώτηση επιστρέφει τα αντικείμενα &7 και &52. Το επόμενο βήμα προχωρεί πιο πέρα την προηγούμενη επερώτηση, και επιστρέφει διευθύνσεις μαζί με τα περιβάλλοντα κάτω από τα οποία υφίστανται οι διευθύνσεις αυτές:

```
select address: <[Y]: X>
from [% d=15]music_club.address::[Y] [-] X
```

Το αποτέλεσμα είναι:

```
{address:
  <[d in {start..now}, season in {fall,winter,spring}]: &7 {...},
  [d in {10..now}, season=summer]: &52 {...}>
}
```

Η επερώτηση χρησιμοποιεί την μεταβλητή περιβάλλοντος [Y] για να επιστρέφει τα περιβάλλοντα που αντιστοιχούν σε εκφάνσεις του address. Για το επόμενο βήμα, υποθέστε ότι δεν ενδιαφερόμαστε να ανακτήσουμε την διάσταση d, την οποία έχουμε ήδη ορίσει στον όρο from. Η παρακάτω επερώτηση επιστρέφει το ίδιο αποτέλεσμα όπως πριν, αλλά με περιβάλλοντα που περιέχουν μόνον την διάσταση season:

```
select address: <[Z]: X>
context [Z] as [Y] * [% season in ALL]
from [% d=15]music_club.address::[Y] [-] X
```

Το αποτέλεσμα είναι:

```
{address:
  <[season in {fall,winter,spring}]: &7 {...},
  [season=summer]: &52 {...}>
}
```

Ο όρος context χρησιμοποιεί το πρότυπο περιβάλλοντος [% season in ALL] για να απαλείψει διαστάσεις εκτός της season από τα περιβάλλοντα που δεσμεύονται στην [Y]. Η παραπάνω επερώτηση επιδεικνύει την άξια και πολλαπλή χρησιμότητα των προτύπων περιβάλλοντος. Για να ενισχύσουμε την παρατήρηση αυτή, προσέξτε την χρήση των

προτύπων περιβάλλοντος στην ακόλουθη επερώτηση. Επιστρέφει διευθύνσεις που υφίστανται ακριβώς από την χρονική στιγμή 10 μέχρι την now, ανεξάρτητα από τις τιμές άλλων διαστάσεων:

```
select address: X
from [Y]music_club.address X
within [Y] = [% d in {10..now}]
```

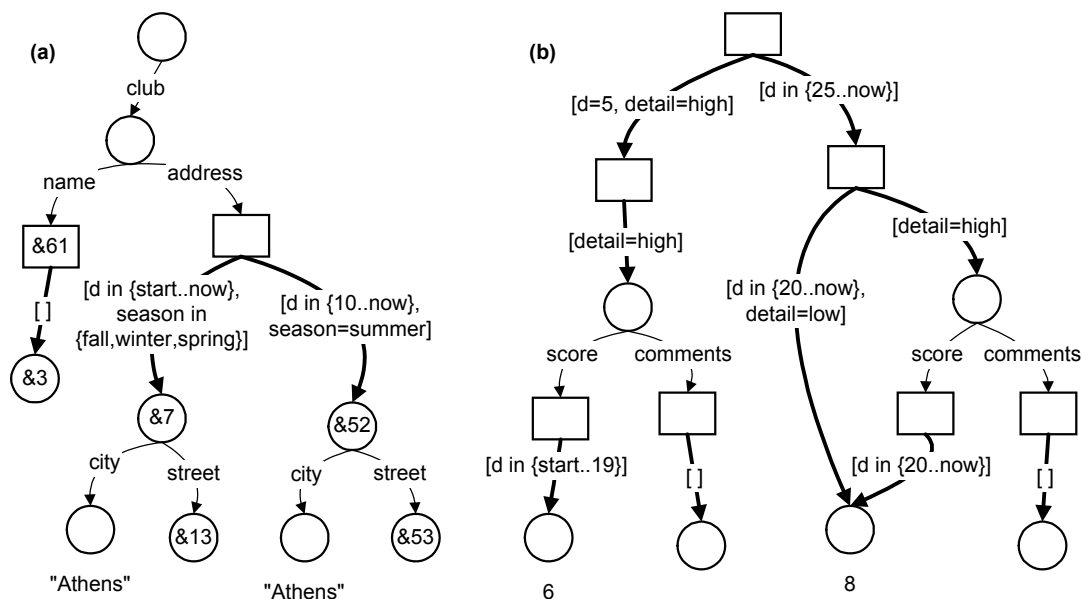
Η επερώτηση επιστρέφει το αντικείμενο &52.

Σαν άλλο ένα παράδειγμα, η παρακάτω επερώτηση ανακτά την τρέχουσα διεύθυνση και το όνομα των clubs εκείνων, τα οποία την χρονική στιγμή 6 βρίσκονταν στην Αθήνα με ταχυδρομικό κώδικα 17671:

```
select club: {name: X.name, address: <[Q]: Z>}
from music_club X,
     X. [% d=6]address Y,
     X. [% d=now]address::[Q] [-] Z
where Y.city = "Athens" and Y.zipcode = 17671
```

Το αποτέλεσμα της επερώτησης απεικονίζεται στην Εικόνα 6.10 (a), και δείχνει ότι ένα club που την χρονική στιγμή 6 βρίσκονταν στην Αθήνα με ταχυδρομικό κώδικα 17671 έχει αυτή την στιγμή δυο διευθύνσεις, μια για το καλοκαίρι και μια για τον υπόλοιπο χρόνο.

**Εικόνα 6.10: Αποτελέσματα επερωτήσεων MQL στο ιστορικό MOEM βάσης.**



Η επερώτηση που ακολουθεί επιστρέφει τις κριτικές (reviews) που υφίστανται κάτω από κάποια χρονική στιγμή ανάμεσα από την 12 και την 22, μαζί με το πραγματικό τους έγκυρο διάστημα, ανεξάρτητα από άλλες διαστάσεις:

```
select [Z]: X
context [Z] as [Y] * [% d in ALL]
from [Y]music_club.review X
within [Y] * [% d in {12..22}] != [-]
```

Το αποτέλεσμα είναι:

```
<[d in {start..19}]: &8 6,
  [d in {20..now}]: &55 8,
  [d in ALL]: &9 {score: &54 <...>, comments: &62 <...>}
>
```

Το αποτέλεσμα ενσωματώνει όλες τις εκφάνσεις του `review`, επειδή όλες υφίστανται κάτω από κάποια χρονική στιγμή ανάμεσα από την 12 και την 22. Παρατηρήστε ότι μόνον η διάσταση `d` εμφανίζεται στα περιβάλλοντα του αποτελέσματος, επειδή οι άλλες διαστάσεις – συγκεκριμένα η `detail` – έχουν απαλειφθεί από το πρότυπο περιβάλλοντος `[% d in ALL]`. Για άλλη μια φορά, η επερώτηση αυτή επιδεικνύει πως τα πρότυπα περιβάλλοντος μπορούν να χειριστούν περιβάλλοντα που περιέχουν πολλές διαστάσεις με κομψό και αποτελεσματικό τρόπο.

Σαν τελευταίο παράδειγμα, η παρακάτω επερώτηση ανάγει μερικώς το πολυδιάστατο αντικείμενο `review` για δυο περιβάλλοντα.

```
select holding <[d=5,detail=high]: <X>, [d in {25..now}]: <X>>
from music_club.review <X>
```

Η Εικόνα 6.10 (b) δείχνει τον γράφο του αποτελέσματος. Η μεταβλητή πολυδιάστατου αντικειμένου `<X>` δεσμεύεται στον κόμβο `&5`, και η λέξη-κλειδί `holding` προξενεί την απομάκρυνση σε όλη την έκταση του υπογράφου των κόμβων και των ακμών που είναι άσχετοι με το περιβάλλον της αναγωγής. Η μερική αναγωγή για `[d=5,detail=high]` δίνει τα περιεχόμενα της κριτικής σε υψηλή λεπτομέρεια κατά την χρονική στιγμή 5, ενώ η μερική αναγωγή για `[d in {25..now}]` δίνει τα περιεχόμενα της κριτικής σε οποιαδήποτε λεπτομέρεια εφόσον υφίστανται κάτω από κάποια χρονική στιγμή ανάμεσα από την 25 και την `now`.

## 6.6 ΣΧΕΤΙΚΕΣ ΕΡΓΑΣΙΕΣ

Η ικανότητα μοντελοποίησης της χρονικής διάστασης του πραγματικού κόσμου είναι σημαντική για πολλές εφαρμογές υπολογιστών. Τις δυο τελευταίες δεκαετίες υπήρξε σημαντικός όγκος ερευνάς πάνω στην επέκταση των βάσεων δεδομένων προκειμένου να υποστηρίξουν χρόνο [SA85, JCG+92, ZCF+97, OS95, TG95, BCW93, Org96]. Επιπλέον, πρόσφατη δουλειά διερευνά τρόπους για την ενσωμάτωση χρόνου στα ημιδομημένα δεδομένα [CAW99, OQT01, DOQT01, GS98] και στην XML [GS98, AYU00, GM00, MGS101a, Dyr01, Nor02, Gra02, SD02].

Το πρόβλημα της αναπαράστασης και επερώτησης αλλαγών στα ημιδομημένα δεδομένα έχει μελετηθεί στα [CAW99, CAW98], όπου προτάθηκε το Delta OEM (DOEM για συντομία). Το DOEM είναι ένα μοντέλο γράφου που επεκτείνει το OEM με επισημειώσεις που περιέχουν χρονική πληροφορία. Τέσσερις λειτουργίες βασικών αλλαγών, συγκεκριμένα οι `creNode`, `updNode`, `addArc`, και `remArc` χρησιμοποιήθηκαν από τους συγγραφείς για την μετατροπή ενός γράφου OEM. Οι λειτουργίες αυτές αντιστοιχίζονται σε τέσσερις κατηγορίες επισημειώσεων. Οι επισημειώσεις είναι ετικέτες που επικολλούνται σε έναν κόμβο ή σε μια ακμή, και που περιέχουν πληροφορία που κωδικοποιεί το ιστορικό των αλλαγών για τον συγκεκριμένο κόμβο ή την ακμή. Όταν λάβει χώρα μια λειτουργία βασικών αλλαγών του

OEM, μια νέα επισημείωση προστίθεται στον κόμβο ή την ακμή που επηρεάστηκε, διατυπώνοντας το είδος της λειτουργίας, την χρονοσφραγίδα, και στην περίπτωση της *updNode* την παλιά τιμή του αντικειμένου. Οι τροποποιήσεις που υποδεικνύονται από τις λειτουργίες βασικών αλλαγών πράγματι εκτελούνται, εκτός από την περίπτωση της διαγραφής ακμής, που έχει σαν αποτέλεσμα απλά την επικόλληση μιας επισημείωσης στην ακμή. Για την επερώτηση βάσεων DOEM, προτείνεται η γλώσσα επερωτήσεων Chorel. Η Chorel επεκτείνει την Lorel με κατασκευές που λέγονται *εκφράσεις επισημείωσης (annotation expressions)*, που τοποθετούνται στις εκφράσεις μονοπατιού και συγκρίνονται με τις επισημειώσεις στον γράφο DOEM. Οι εκφράσεις επισημείωσης μπορούν να περιέχουν *μεταβλητές χρόνου (time variables)* που επιτρέπουν την διατύπωση συνθηκών πάνω στις χρονοσφραγίδες. Η Chorel χειρίζεται ξεχωριστές χρονικές στιγμές και χρονικά διαστήματα εισάγοντας ειδικές λέξεις-κλειδιά στις εκφράσεις επισημείωσης, όπως οι *at*, *during*, και *snap*.

Η προσέγγισή μας βασίζεται σε κάποιες από τις βασικές έννοιες που παρουσιάστηκαν στο [CAW99]. Είναι όμως πολύ διαφορετική, καθώς οι αλλαγές αναπαρίστανται εισάγοντας νέες εκφάνσεις αντί για την προσθήκη επισημειώσεων. Όπως το DOEM, έτσι και το MOEM μπορεί να δώσει το στιγμιότυπο μιας βάσης OEM για οποιαδήποτε χρονική στιγμή, μέσω της αναγωγής σε OEM. Η κύρια διαφορά ανάμεσα στις γλώσσες επερωτήσεων πηγάζει, όπως είναι αναμενόμενο, από τις διαφορετικές αναπαραστάσεις του ιστορικού του OEM. Η Chorel, με τις εκφράσεις επισημείωσης, διατηρεί σε κεντρικό ρόλο την έννοια της λειτουργίας βασικών αλλαγών του OEM, η οποία λειτουργία εφαρμόζεται σε κάποιο συγκεκριμένο στοιχείο και λαμβάνει χώρα μια συγκεκριμένη χρονική στιγμή. Στην συνέχεια, αρχίζοντας από αυτή την βάση, η Chorel χτίζει έννοιες υψηλότερου επιπέδου, όπως για παράδειγμα αυτή ενός στοιχείου που υφίσταται κάτω από ένα χρονικό διάστημα, ή ένα ολόκληρο μονοπάτι που υφίσταται κάτω από μια χρονική στιγμή. Από την άλλη, η έννοια των λειτουργιών βασικών αλλαγών του OEM δεν υπάρχει στην MQL. Οι επερωτήσεις MQL απευθύνονται στα αποτελέσματα των λειτουργιών αυτών, που κρατούνται σαν εκφάνσεις πολυδιάστατων οντοτήτων. Η κεντρική έννοια στην MQL είναι αυτή του περιβάλλοντος, πράγμα που σημαίνει ότι η MQL χειρίζεται απευθείας τα διαστήματα έγκυρου χρόνου των στοιχείων της βάσης και των μονοπατιών της βάσης.

Ένας ειδικός γράφος για την μοντελοποίηση δυναμικών θεμάτων των ημιδομημένων δεδομένων, που λέγεται «*semistructured temporal graph*» προτείνεται στο [OQT01]. Στον γράφο αυτό, κάθε κόμβος και ακμή έχει μια ετικέτα που περιλαμβάνει ένα τμήμα το οποίο δηλώνει το έγκυρο διάστημα για τον κόμβο ή την ακμή. Τροποποιήσεις στον γράφο προκαλούν αλλαγές στο χρονικό τμήμα των ετικετών των κόμβων και των ακμών που επηρεάζονται.

Μια προσέγγιση για την αναπαράσταση χρονικών έγγραφων XML προτείνεται στο [AYU00], όπου οι κόμβοι που είναι φύλλα μπορούν να έχουν εναλλακτικές τιμές, με καθεμία να υφίσταται κάτω από μια χρονική περίοδο. Πάντως, το μοντέλο που προτείνεται το [AYU00] δεν επιτρέπει διαστάσεις εκτός του χρόνου, και δεν υποστηρίζει άμεσα εκφάνσεις με διαφοροποιούμενη δομή για κόμβους που δεν είναι φύλλα. Άλλες προσεγγίσεις για την αναπαράσταση χρόνου στην XML μπορούν να βρεθούν στα [GM00, MGS101a, Gra02]. Στο [Dyr01] το μοντέλο δεδομένων του XPath επεκτείνεται για να υποστηρίζει χρόνο δοσοληψίας (*transaction time*). Η γλώσσα επερωτήσεων του XPath επεκτείνεται επίσης με έναν «άξονα» για χρόνο δοσοληψίας ώστε να επιτραπεί η πρόσβαση σε παρελθούσες και μελλοντικές καταστάσεις. Προτείνονται επίσης κατασκευές που εξάγουν και συγκρίνουν χρόνους. Τέλος, στο [SD02] το μοντέλο δεδομένων και η γλώσσα επερωτήσεων του XPath επεκτείνονται ώστε να ενσωματώσουν έγκυρο χρόνο (*valid time*), και το XPath επεκτείνεται με έναν «άξονα» προκειμένου να έχει πρόσβαση στον έγκυρο χρόνο των κόμβων.



Ένα σημαντικό πλεονέκτημα των MSSD σε σχέση με άλλες προσεγγίσεις για την διαχείριση ιστορικών των SSD είναι ότι το MSSD μπορεί να εφαρμοστεί σε μια ποικιλία προβλημάτων από διαφορετικά γνωστικά πεδία: η αναπαράσταση έγκυρου χρόνου είναι μόνο μια από τις πιθανές εφαρμογές των MSSD. Το MOEM είναι κατάλληλο για την μοντελοποίηση οντοτήτων που παρουσιάζουν διαφορετικές εκφάνσεις, ένα πρόβλημα που απαντάται συχνά στον Ιστό, και η αναπαράσταση ιστορικών των ημιδομημένων βάσεων μπορεί να ιδωθεί σαν ειδική περίπτωση αυτού του προβλήματος. Οι ιδιότητες και οι διεργασίες που ορίστηκαν γενικά για τα MSSD (όπως η κληρονομούμενη κάλυψη μονοπατιού, η αναγωγή σε OEM, η μερική αναγωγή) χρησιμοποιούνται χωρίς αλλαγή και στην περίπτωση της αναπαράστασης ιστορικών των ημιδομημένων δεδομένων. Ομοίως, η MQL δεν έχει φτιαχτεί ειδικά για την επερώτηση ιστορικών των ημιδομημένων βάσεων, αλλά στοχεύει στην γενική περίπτωση των ημιδομημένων δεδομένων που εξαρτώνται από το ερμηνευτικό περιβάλλον. Συνεπώς, οι έννοιες που προτείναμε σαν μέρος του MOEM και της MQL έχουν ευρύτερη εφαρμογή, και δεν περιορίζονται από το πλαίσιο του συγκεκριμένου προβλήματος. Επιπλέον, όπως δείξαμε στις Ενότητες 6.4 και 6.5.2, το MOEM και η MQL μπορούν να αναπαραστήσουν και να επερωτήσουν τα ιστορικά του ίδιου του MOEM, χωρίς να χρειάζεται να εισαχθούν επιπλέον έννοιες.

## 6.7 ΠΕΡΙΛΗΨΗ

Στο κεφάλαιο αυτό εξηγήσαμε σε βάθος πως το πολυδιάστατο OEM, ένα μοντέλο γράφου για ημιδομημένα δεδομένα που εξαρτώνται από το ερμηνευτικό περιβάλλον, μπορεί να χρησιμοποιηθεί για να αναπαραστήσει το ιστορικό μιας βάσης δεδομένων OEM. Εισηγήσαμε τις **λειτουργίες βασικών αλλαγών** του MOEM, και προδιαγράψαμε με ψευδοκώδικα την διαδικασία που κωδικοποιεί στο MOEM το ιστορικό μιας βάσης δεδομένων OEM. Συζητήσαμε πως οι ιδιότητες του Πολυδιάστατου Γράφου Δεδομένων εφαρμόζονται στην συγκεκριμένη περίπτωση, και δείξαμε ότι χρονικά στιγμιότυπα της βάσης OEM μπορούν να ανακτηθούν από το MOEM. Παρουσιάσαμε το υλοποιημένο σύστημα *OEM History*, και επιδείξαμε μέσα από ένα παράδειγμα την διαδικασία χρήσης μιας υποκειμένης βάσης δεδομένων MOEM προκειμένου να μοντελοποιηθούν αλλαγές στο OEM. Επιπρόσθετα, δείξαμε ότι το MOEM είναι ικανό να μοντελοποιήσει αλλαγές που συμβαίνουν όχι μόνο σε βάσεις δεδομένων συμβατικών OEM, αλλά επίσης και σε βάσεις δεδομένων MOEM.

Ένα σημαντικό όφελος της προσέγγισής μας προέρχεται από τις επερωτήσεις που μπορούν να τεθούν σε ιστορικά OEM και ιστορικά MOEM. Η προσέγγισή μας διατηρεί στιγμιότυπα των διαδοχικών καταστάσεων ενός αντικειμένου σαν εκφάνσεις μιας πολυδιάστατης οντότητας. Αυτό επιτρέπει τον σχηματισμό επερωτήσεων που συσχετίζουν διαφορετικές καταστάσεις του ίδιου ή και διαφορετικών αντικειμένων σε διάφορες χρονικές στιγμές. Παρουσιάσαμε έναν αριθμό παραδειγμάτων επερωτήσεων MQL, και μέσα από αυτά επιβεβαιώσαμε την εκφραστικότητα της MQL και την αξία κάποιων χαρακτηριστικών της, όπως για παράδειγμα των προτύπων περιβάλλοντος.

Κατά την άποψή μας, αυτή η χαρακτηριστική εφαρμογή του MOEM και της MQL επιδεικνύει τις δυνατότητες των MSSD. Το εύρος εφαρμογών του MOEM και της MQL δεν εξαντλείται στην αναπαράσταση ιστορικών ημιδομημένων δεδομένων. Η σημασία δεδομένων που εξαρτώνται από το ερμηνευτικό περιβάλλον αυξάνει σε ένα παγκοσμιοποιημένο σύστημα όπως είναι ο Ιστός, και τα MSSD μπορούν να εφαρμοστούν σε προβλήματα από διαφορετικά γνωστικά πεδία, μεταξύ των οποίων: στην παροχή προσωποποιημένων πληροφοριών και υπηρεσιών· στην ενοποίηση πληροφορίας όπου αντικείμενα διαφέρουν σε τιμή και δομή ανάλογα με την πηγή· στην αναπαράσταση και

επερώτηση γεωγραφικής πληροφορίας όπου πιθανές διαστάσεις μπορούν να είναι η *κλίμακα* και το *θέμα*.

## 7 ΠΟΛΥΔΙΑΣΤΑΤΗ XML

Η eXtensible Markup Language, ή XML για συντομία, είναι μια γλώσσα επισημείωσης που έχει καθιερωθεί για την αναπαράσταση και την ανταλλαγή δεδομένων στον Ιστό [XML, Wal97, Cha99]. Η XML μοιάζει με την Hypertext Markup Language [HTML], ή HTML για συντομία, αλλά αντίθετα με την HTML επικεντρώνεται στην δομή των δεδομένων και όχι στην παρουσίασή τους. Η XML μπορεί να ιδωθεί είτε από μια εγγραφο-κεντρική, ή από μια δεδομενο-κεντρική οπτική. Η εγγραφο-κεντρική προσέγγιση προέρχεται από την SGML [SGML], την γλώσσα επισημείωσης που ενέπνευσε τον σχεδιασμό της XML, και βλέπει την XML σαν έναν τρόπο για να ενσωματωθεί σε έγγραφα του Ιστού πληροφορία σχετική με την δομή τους. Η δεδομενο-κεντρική προσέγγιση αντιλαμβάνεται την XML σαν μια γλώσσα κατάλληλη για την έκφραση ημιδομημένων δεδομένων και για την ανταλλαγή δεδομένων μέσω του Ιστού. Από την προοπτική αυτή, η έμφαση βρίσκεται στις επερωτήσεις και στην περιγραφή των συσχετίσεων ανάμεσα από τα δεδομένα, όπως εάν επρόκειτο για κάποιο σχήμα βάσης δεδομένων.

Παρά το ότι το κύριο χαρακτηριστικό της XML είναι η επεκτασιμότητά της καθώς είναι δυνατόν να ορίζονται νέα στοιχεία κατά βούληση, δεν είναι κατάλληλη για την αναπαράσταση πληροφορίας που εμφανίζει διαφορετικές εκφάνσεις κάτω από διαφορετικά ερμηνευτικά περιβάλλοντα.. Σαν ένα απλό παράδειγμα, φαντασθείτε μια αναφορά που πρέπει να αναπαρασταθεί σε διάφορους βαθμούς λεπτομέρειας, και σε διάφορες γλώσσες. Μια λύση θα ήταν να δημιουργηθεί ένα διαφορετικό έγγραφο XML για κάθε δυνατό συνδυασμό παραλλαγών, και να προσκολληθεί κάποιο είδος «ετικέτας» που να υποδεικνύει το ερμηνευτικό περιβάλλον κάθε εγγράφου. Μια τέτοια προσέγγιση δεν είναι βεβαίως πρακτική, καθώς ενέχει περιττή επανάληψη πληροφορίας σε μεγάλο βαθμό. Ακόμη πιο σημαντικό, οι διαφορετικές εκφάνσεις που βρίσκονται σε διαφορετικά έγγραφα δεν συνδέονται μεταξύ τους σαν μέρη της ίδιας οντότητας.

Κατευθύνσεις για το πως μπορεί να αντιμετωπιστεί το πρόβλημα αυτό δίνονται στα [SGR00, SGM00], όπου παρουσιάζεται η **Πολυδιάστατη XML**, ή **MXML** για συντομία. Επιπλέον, μερικές πρώτες σκέψεις για την αναπαράσταση χρόνου με την χρήση της MXML συζητούνται στα [MGS101, MGS101a]. Στο κεφάλαιο αυτό, ενοποιούμε δουλειά που δημοσιεύτηκε στα [GSK01, GSK+01]. Συγκεκριμένα, περιγράφουμε πως η MXML επεκτείνει το συντακτικό της XML, επιτρέποντας σε προσδιοριστές περιβάλλοντος να χαρακτηρίζουν στοιχεία και γνωρίσματα και να καθορίζουν τους κόσμους κάτω από τους οποίους τα συστατικά μέρη ενός εγγράφου έχουν υπόσταση. Επίσης προτείνουμε ένα **πολυδιάστατο παράδειγμα** για την εμφάνιση δεδομένων του Ιστού που εξαρτώνται από το ερμηνευτικό περιβάλλον, και παρουσιάζουμε ένα σύστημα που υλοποιεί την βασική λειτουργικότητα αυτού του πολυδιάστατου παραδείγματος.

Η MXML επηρεάστηκε από την Intensional HTML [WBSY98, Bro98, Yi197, Bro98a], ή IHTML για συντομία, μια γλώσσα συγγραφής εγγράφων του Ιστού που βασίζεται και επεκτείνει ιδέες που προτάθηκαν για ένα σύστημα διαχείρισης εκδοχών λογισμικού (software versioning system) [PW93]. Η IHTML επιτρέπει σε μια ιστοσελίδα να έχει διαφορετικές παραλλαγές και να προσαρμόζεται δυναμικά σε ένα δεδομένο ερμηνευτικό περιβάλλον. Η κύρια διαφορά ανάμεσα από την MXML και την IHTML είναι μια προβολή της κύριας διαφοράς ανάμεσα

στην XML και την HTML, δηλαδή η έμφαση βρίσκεται στην κωδικοποίηση της δομής και όχι στην παρουσίαση των δεδομένων.

Η ιδέα της ενσωμάτωσης πολλαπλών διαστάσεων σε γλώσσες δεν είναι καινούρια. Η GLU [AFJ91, AFJW95] είναι μια γλώσσα συναρτησιακού προγραμματισμού που επιτρέπει στον χρήστη να δηλώσει διαστάσεις και να ορίσει οντότητες που διαφοροποιούνται κατά μήκος αυτών των διαστάσεων. Επιπλέον, μια γλώσσα στο ίδιο πνεύμα με την GLU έχει αναπτυχθεί στο πεδίο του λογικού προγραμματισμού [OD97], ενώ η ISE [SW00] αποτελεί μια πολυδιάστατη εκδοχή της Perl [PERL].

Η δουλειά μας στα MSSD άρχισε σαν μια προσπάθεια να ενσωματώσουμε το ερμηνευτικό περιβάλλον στην XML, αλλά γρήγορα στραφήκαμε στο OEM σαν μια απλούστερη και πιο αφηρημένη βάση για να αντιμετωπίσουμε πάνω της τα προβλήματα που θα εμφανίζονταν. Στα ακόλουθα *δεν παρουσιάζουμε την MXML σαν μια ολοκληρωμένη εργασία, αλλά μάλλον σαν παράδειγμα του πως οι ιδέες που εισήχθησαν σε προηγούμενα κεφάλαια μπορούν να επεκταθούν σε διαφορετικές κατευθύνσεις.*

## 7.1 ΕΝΣΩΜΑΤΩΣΗ ΕΡΜΗΝΕΥΤΙΚΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΣΤΗΝ XML

Σε ένα έγγραφο πολυδιάστατης XML, ή έγγραφο MXML για συντομία, το ερμηνευτικό περιβάλλον μπορεί να προσδιορίζει στοιχεία και γνώρισμα. Ένα στοιχείο του οποίου το περιεχόμενο εξαρτάται από μία ή περισσότερες διαστάσεις λέγεται **πολυδιάστατο στοιχείο**, ενώ ένα γνώρισμα του οποίου η τιμή εξαρτάται από μία ή περισσότερες διαστάσεις λέγεται **πολυδιάστατο γνώρισμα**.

Στην ενότητα αυτή, εισάγουμε την MXML σαν μια επέκταση του συντακτικού της XML που ενσωματώνει προσδιοριστές περιβάλλοντος και μπορεί να αναπαραστήσει πολυδιάστατες οντότητες. Επικεντρώνουμε στο κομμάτι εκείνο των εγγράφων που αντιστοιχεί στην λογική δομή της πληροφορίας, και δεν αντιμετωπίζουμε θέματα που σχετίζονται με την φυσική δομή των εγγράφων XML. Επιπλέον, δεχόμαστε ότι η MXML χρησιμοποιεί το συντακτικό της XML για να αναφέρεται σε φακέλους DTD και XSL.

### 7.1.1 Γιατί επεκτείνουμε το συντακτικό της XML;

Κάποιος ταγμένος στην διατήρηση των στάνταρτ ίσως να βλέπει σαν μειονέκτημα το γεγονός ότι η MXML τροποποιεί το συντακτικό της XML. Η εναλλακτική προσέγγιση θα ήταν να οριστεί η MXML σαν ένα συμβατικό DTD (ένα «στιγμιότυπο» της XML), χωρίς να επεκταθεί το συντακτικό της XML, και να χρησιμοποιούνται namespaces [XNS] προκειμένου να συνδυάζεται με DTDs για συγκεκριμένες θεματικές περιοχές. Υπάρχουν τρεις λόγοι για τους οποίους επιλέξαμε να επεκτείνουμε το συντακτικό της XML:

- (a) Το ερμηνευτικό περιβάλλον μπορεί να χρησιμεύσει σε μια ευρεία γκάμα άσχετων μεταξύ τους θεματικών περιοχών, έτσι κατά την άποψή μας τα μοντέλα δεδομένων και οι γλώσσες επερώτησης του Ιστού θα πρέπει να υποστηρίζουν το ερμηνευτικό περιβάλλον σαν πολίτη πρώτης τάξης. Πιστεύουμε ότι ορίζοντας την MXML σαν ένα συμβατικό DTD υποβιβάζουμε το ερμηνευτικό περιβάλλον, και ότι ο ρόλος του και οι δυνατότητές του στον Ιστό υποτιμούνται.
- (b) Η XML πράγματι θα μπορούσε να αναπαραστήσει *στοιχεία* πολλαπλών εκφάνσεων με την βοήθεια λίγων κρατημένων συμβόλων, ορισμένων σε ένα συμβατικό DTD. Όμως, προκειμένου να εκφραστούν πολυδιάστατα *γνωρίσματα* χωρίς να επεκταθεί

το συντακτικό της XML, τα γνώρισμα θα πρέπει να μετατραπούν σε κάποιο είδος ειδικών στοιχείων XML. Το πρόβλημα με την προσέγγιση αυτή είναι ότι μια XML με «επίγνωση» του ερμηνευτικού περιβάλλοντος γίνεται ακόμη πιο φλύαρη από ότι είναι ήδη η XML, και – παρότι κατανοητή από τα μηχανήματα – ακατάλληλη για να επιδείξει σε άλλους ανθρώπους το ουσιαστικό σημείο. Το ουσιαστικό σημείο, που είναι το ερμηνευτικό περιβάλλον και η πληροφορία πολλαπλών εκφάνσεων, κρύβεται πίσω από την κοπιώδη κατανόηση ενός παράξενου συντακτικού.

- (c) Όπως εξηγούμε στο Κεφάλαιο 2, ο τρόπος που βλέπουμε την XML είναι σαν ένα λογικό μοντέλο δεδομένων παρά σαν έναν μορφότυπο στο φυσικό επίπεδο. Η άμεση υποστήριξη ερμηνευτικού περιβάλλοντος σε λογικό επίπεδο θα διευκόλυne την χρήση γλωσσών επερωτήσης για XML με «επίγνωση» ερμηνευτικού περιβάλλοντος. Είναι ευκολότερο να σχηματίσει κανείς επερωτήσεις οδηγούμενες από περιβάλλον εάν αυτές χρησιμοποιούν έννοιες που αντιστοιχούν ευθέως στο υποκείμενο μοντέλο δεδομένων.

Οι παραπάνω λόγοι μας φάνηκαν επαρκείς για να επεκτείνουμε το συντακτικό της XML. Πάντως, από την στιγμή που εδραιωθεί η εγκυρότητα των πολυδιάστατων μοντέλων δεδομένων και γλωσσών όπου έχουν επίγνωση του ερμηνευτικού περιβάλλοντος, *ίσως είναι λογικό για πρακτικούς σκοπούς να εκφράζουμε πληροφορία πολλαπλών εκφάνσεων μέσω της συμβατικής XML, αντί να χρησιμοποιούμε κάποια επέκταση της XML όπως η MXML.*

## 7.1.2 Συντακτικό της Πολυδιάστατης XML

Το συντακτικό της XML επεκτείνεται ως ακολούθως προκειμένου να ενσωματωθούν προσδιοριστές περιβάλλοντος.

- Ένα πολυδιάστατο στοιχείο έχει την μορφή:

```
<@element_name attribute_specification>
  [context_specifier_1]
    <element_name attribute_specification_1>
      element_content_1
    </element_name>
  [/]
  . . .
  [context_specifier_N]
    <element_name attribute_specification_N>
      element_content_N
    </element_name>
  [/]
</@element_name>
```

Ένα πολυδιάστατο στοιχείο χαρακτηρίζεται από το ειδικό σύμβολο @ που προηγείται του ονόματος του στοιχείου, και περικλείει ένα ή περισσότερα **στοιχεία περιβάλλοντος** που αποτελούν τις εκφάνσεις του πολυδιάστατου αυτού στοιχείου, που υφίστανται κάτω από τους κόσμους που ορίζονται από τον αντίστοιχο προσδιοριστή περιβάλλοντος. Τα στοιχεία περιβάλλοντος έχουν την ίδια μορφή σαν τα συμβατικά στοιχεία XML. Όλα τα στοιχεία περιβάλλοντος που ανήκουν στο ίδιο πολυδιάστατο στοιχείο έχουν το ίδιο όνομα, που είναι το όνομα του πολυδιάστατου στοιχείου.

- Ένα πολυδιάστατο γνώρισμα έχει την μορφή:

```
attribute_name = [context_specifier_1] attribute_value_1 [/]
                .
                .
                [context_specifier_N] attribute_value_N [/]
```

Ένα πολυδιάστατο γνώρισμα εκφράζεται σαν ένα γνώρισμα όπου η τιμή του είναι ένα σύνολο από ζεύγη ερμηνευτικό περιβάλλον / τιμή. Κάθε μία από αυτές τις τιμές που προσδιορίζονται από ερμηνευτικό περιβάλλον γίνεται η υφιστάμενη τιμή του γνωρίσματος για το αντίστοιχο ερμηνευτικό περιβάλλον.

Το Παράδειγμα 7.1 δείχνει μέρος ενός εγγράφου MXML που περιγράφει το πολυδιάστατο φανταστικό μενού ενός εστιατορίου.

**Παράδειγμα 7.1: Ένα MXML που περιγράφει το μενού εστιατορίου.**

```
<restaurant>
  <menu>
    <salad name="Chef's salad" vegetarian=[season=summer]
                                          "yes" [/]
                                          [season!=summer]
                                          "no" [/]>

    <@comment>
      [language=English, detail=low]
      <comment>A traditional salad.</comment>
      [/]
      [language=English, detail=high]
      <comment>
        A salad with a long history, which
        has roots in the tradition of the town.
      </comment>
      [/]
      [language=French, detail in {low,high}]
      <comment>Une salade regionale traditionnelle.</comment>
      [/]
    </@comment>

    <@price>
      [season=summer] <price>3 EUR</price> [/]
      [default] <price>4 EUR</price> [/]
    </@price>

    <ingredient>tomato</ingredient>

    <@ingredient>
      [season!=summer] <ingredient>bacon</ingredient> [/]
    </@ingredient>

    <ingredient>olive oil</ingredient>
```

```

<@ingredient>
  [occasion=special]
    <ingredient special_supplier =
      [season in {spring,summer}] "sp1" [/]
      [default] "sp2" [/]>
    <name>special sauce</name>
    <remarks>Must order three days in advance</remarks>
  </ingredient>
  [/]
  [default] <ingredient>normal sauce</ingredient> [/]
</@ingredient>

... other salad ingredients ...

</salad>

... other menu items ...

</menu>

<supplier scode="sp1">
  <name>John Smith</name>
  <address>234 XYZ Street</address>
</supplier>

<supplier scode="sp2">Peter Brown</supplier>

<\restaurant>
♦

```

Οι προσδιοριστές περιβάλλοντος είναι αναπόσπαστα μέρη των στοιχείων περιβάλλοντος που προσδιορίζουν. Επιπρόσθετα, ο προσδιοριστής περιβάλλοντος [default] χρησιμεύει σαν συντομογραφία για να εκφράσει εκείνους τους κόσμους που δεν καλύπτονται από τους υπόλοιπους προσδιοριστές περιβάλλοντος της ίδιας πολυδιάστατης οντότητας. Όπως στον Πολυδιάστατο Γράφο δεδομένων, η ρίζα ενός εγγράφου MXML μπορεί να είναι ένα συμβατικό στοιχείο, ή ένα πολυδιάστατο στοιχείο.

### 7.1.3 Πως οι διαστάσεις εφαρμόζονται σε στοιχεία

Τα πολυδιάστατα στοιχεία μπορούν να περιέχουν μόνο στοιχεία περιβάλλοντος<sup>109</sup>, ενώ τα στοιχεία περιβάλλοντος μπορούν να περιέχουν πολυδιάστατα στοιχεία, συμβατικά στοιχεία<sup>110</sup>, ή κάποιον συνδυασμό των δυο σε οποιοδήποτε βάθος. Τα στοιχεία περιβάλλοντος του ίδιου πολυδιάστατου στοιχείου δεν απαιτείται να έχουν το ίδιο περιεχόμενο, ούτε καν να συμμορφώνονται με τους ίδιους δομικούς περιορισμούς όπως αυτοί θα ορίζονταν σε κάποιο DTD. Συνεπώς, οι διαστάσεις μπορούν να επηρεάσουν το περιεχόμενο ενός στοιχείου από κάθε πλευρά, είτε αυτή αφορά στην δομή του είτε στην τιμή του.

<sup>109</sup> Στον Πολυδιάστατο Γράφο Δεδομένων οι πολυδιάστατοι κόμβοι μπορούν να δείχνουν σε πολυδιάστατους κόμβους. Στην MXML ένα πολυδιάστατο στοιχείο δεν μπορεί να περιέχει ένα άλλο πολυδιάστατο στοιχείο· μόνον πολυδιάστατα γνωρίσματα μπορούν να χρησιμοποιηθούν για να αναφερθούν σε πολυδιάστατα στοιχεία.

<sup>110</sup> Στην MXML τα στοιχεία περιβάλλοντος και τα συμβατικά στοιχεία έχουν την ίδια μορφή, εν τούτοις θεωρούνται σαν διαφορετικές κατηγορίες επειδή έχουν διαφορετικού είδους δηλώσεις στο MDTD, όπως θα δούμε στην Ενότητα 7.2.2.

*Η επίδραση του ερμηνευτικού περιβάλλοντος στην τιμή των στοιχείων:* Θεωρήστε το πολυδιάστατο στοιχείο `comment` στο Παράδειγμα 7.1, του οποίου η τιμή εξαρτάται από τις διαστάσεις `language` και `detail`. Ο προσδιοριστής περιβάλλοντος του τρίτου στοιχείου περιβάλλοντος του `comment` είναι `[language=French, detail in {low, high}]`. Σαν αποτέλεσμα, η τιμή του `comment` κάτω από το `[language=French, detail=low]` και το `[language=French, detail=high]` είναι “Une salade regionale traditionnelle”.

*Η επίδραση του ερμηνευτικού περιβάλλοντος στην δομή των στοιχείων:* Οι προσδιοριστές περιβάλλοντος επηρεάζουν επίσης και την δομή των στοιχείων. Σαν παράδειγμα, θεωρήστε το τέταρτο συστατικό του `salad` στο Παράδειγμα 7.1, όπου το στοιχείο `ingredient` περιέχει τα υποστοιχεία `name` και `remarks` όταν το ερμηνευτικό περιβάλλον είναι `[occasion=special]`, αλλά για όλες τις άλλες τιμές της διάστασης `occasion` (που υποδηλώνονται με την χρήση του `[default]`), δεν περιέχει καθόλου υποστοιχεία.

Παρατηρήστε ότι δεν είναι απαραίτητο για ένα πολυδιάστατο στοιχείο να έχει στοιχεία περιβάλλοντος για κάθε δυνατό κόσμο. Για παράδειγμα, το πολυδιάστατο στοιχείο

```
<@ingredient>
  [season!=summer] <ingredient>bacon</ingredient> [/]
</@ingredient>
```

στο Παράδειγμα 7.1, δεν έχει καμία έκφραση για το ερμηνευτικό περιβάλλον `[season=summer]`.

Υπάρχει μια περίπτωση όπου το πολυδιάστατο στοιχείο ή το πολυδιάστατο γνώρισμα γίνεται ισοδύναμο με ένα συμβατικό στοιχείο ή συμβατικό γνώρισμα. Για παράδειγμα, θεωρήστε το στοιχείο `ingredient` στο Παράδειγμα 7.1:

```
<ingredient>tomato</ingredient>
```

Το παραπάνω είναι στην πραγματικότητα συντομογραφία για:

```
<@ingredient>
  [] <ingredient>tomato</ingredient> [/]
</@ingredient>
```

## 7.1.4 Πως οι διαστάσεις εφαρμόζονται σε γνωρίσματα

Σε ένα έγγραφο XML, ένα δεδομένο που εκφράζεται σαν γνώρισμα μπορεί επίσης να εκφραστεί και σαν στοιχείο. Το αντίστροφο δεν ισχύει, επειδή τα γνωρίσματα δεν μπορούν να είναι σύνθετου τύπου. Στην πραγματικότητα, από την οπτική της έκφρασης πολυδιάστατων δεδομένων, η μόνη σημαντική διαφορά ανάμεσα από στοιχεία και γνωρίσματα είναι η χρήση των γνωρισμάτων για να διασυνδέουν στοιχεία μέσω των τύπων γνωρισμάτων “ID”, “IDREF”, και “IDREFS”, που μπορούν να ιδωθούν σαν αντίστοιχα της σχέσης στοιχείου – υποστοιχείου. Έτσι, τα στοιχεία και τα γνωρίσματα θα πρέπει να θεωρηθούν ισοδύναμα από νοηματική άποψη, και ότι ισχύει για τα στοιχεία θα πρέπει καταρχήν να ισχύσει και για τα γνωρίσματα.

Μέσα σε ένα πολυδιάστατο στοιχείο, κάθε στοιχείο περιβάλλοντος μπορεί να έχει τα δικά του γνωρίσματα (συμβατικά ή πολυδιάστατα), πιθανώς διαφορετικά από εκείνα άλλων στοιχείων περιβάλλοντος, ακριβώς όπως μπορεί να έχει τα δικά του στοιχεία παιδιά.

Οι προσδιοριστές περιβάλλοντος καθορίζουν ποια είναι τα γνωρίσματα ενός στοιχείου κάτω από έναν κόσμο. Παρατηρήστε ότι στο Παράδειγμα 7.1, το `ingredient` μιας σαλάτας



(salad) έχει το γνώρισμα `special_supplier` για το ερμηνευτικό περιβάλλον [`occasion=special`], αλλά για κάθε άλλη τιμή του `occasion` (όπως υποδηλώνεται από το [`default`]) το στοιχείο `ingredient` δεν έχει καθόλου γνωρίσματα.

Γνωρίσματα του τύπου “ID” μπορούν επισυναφθούν σε πολυδιάστατα στοιχεία<sup>111</sup> καθώς επίσης και σε στοιχεία περιβάλλοντος και συμβατικά στοιχεία, ενώ γνωρίσματα του τύπου “IDREF” και “IDREFS” μπορούν να επισυναφθούν μόνο σε στοιχεία περιβάλλοντος και συμβατικά στοιχεία. Έτσι, χρησιμοποιώντας γνωρίσματα τύπου “IDREF” και “IDREFS”, τα στοιχεία περιβάλλοντος και τα συμβατικά στοιχεία μπορούν να αναφέρονται (ή να «δείχνουν») σε πολυδιάστατα, συμβατικά, ή στοιχεία περιβάλλοντος. Στην πραγματικότητα, το μόνο γνώρισμα που μπορεί να επισυναφθεί σε πολυδιάστατα στοιχεία είναι τύπου “ID”. Συνεπώς, παρόλο που τα πολυδιάστατα στοιχεία μπορούν να δείχθούν από γνωρίσματα “IDREF”, δεν μπορούν τα ίδια να χρησιμοποιήσουν γνωρίσματα για να δείξουν άλλα στοιχεία<sup>112</sup>. Γνωρίσματα τύπου “ID” δεν μπορούν να είναι πολυδιάστατα γνωρίσματα.

Το γνώρισμα `special_supplier` στο τέταρτο `ingredient` στοιχείο στο Παράδειγμα 7.1, έχει τιμή “sp1” κάτω από τους κόσμους που περιέχονται στο [`season in {spring, summer}`], και τιμή “sp2” κάτω από κάθε άλλο κόσμο (όπως συμβολίζεται από το [`default`]). Παρατηρήστε ότι, επειδή η τιμή αυτού του γνωρίσματος είναι μια αναφορά που δείχνει σε ένα στοιχείο `supplier`, το γνώρισμα δείχνει σε διαφορετικά στοιχεία `supplier` ανάλογα με τις τιμές της διάστασης `season`. Σαν επιπλέον παράδειγμα, θεωρήστε το πολυδιάστατο γνώρισμα `vegetarian` του στοιχείου `salad`, που προσδιορίζει την σαλάτα σαν κατάλληλη ή όχι για χορτοφάγους ανάλογα με την διάσταση `season`.

## 7.2 ΠΟΛΥΔΙΑΣΤΑΤΟ DTD

Ένα Document Type Definition [XML] για XML, ή DTD για συντομία, χρησιμοποιείται σαν μια προδιαγραφή που ορίζει περιορισμούς πάνω στην λογική δομή εγγράφων XML. Στην ενότητα αυτή, προτείνουμε μια επέκταση του DTD που λέγεται **Πολυδιάστατο DTD**, ή **MDTD** για συντομία, που παίρνει υπόψη του το ερμηνευτικό περιβάλλον και τις διαστάσεις, και περιγράφει την λογική δομή εγγράφων MXML. Ένα έγγραφο MXML καλείται **έγκυρο** σε σχέση με ένα MDTD εάν συμμορφώνεται με τους περιορισμούς που δηλώνονται σε αυτό το MDTD.

### 7.2.1 Δήλωση διαστάσεων

Οι διαστάσεις δηλώνονται στο MDTD ως εξής:

```
<!DIMENSION dimension_name dimension_domain>
```

Χρησιμοποιώντας δηλώσεις διαστάσεων μπορούμε να δηλώσουμε μια διάσταση και να την συνδέσουμε με ένα σύνολο πιθανών τιμών. Για παράδειγμα, η δήλωση

```
<!DIMENSION language {English, French}>
```

<sup>111</sup> Γνωρίσματα ID πολυδιάστατων στοιχείων χρησιμοποιούνται μόνον για να αναφέρονται σε αυτά και δεν θα πρέπει να ενέχουν κάποια πληροφορία (όπως, για παράδειγμα, το ISBN ενός βιβλίου), επειδή δεν επιβιώνουν κάτω από κανέναν κόσμο.

<sup>112</sup> Ένα στοιχείο περιβάλλοντος στην MXML δείχνεται μόνον από το πολυδιάστατο στοιχείο που το περιέχει. Όμως, μπορεί να δείχνεται από πολλά πολυδιάστατα γνωρίσματα. Αυτό είναι παρόμοιο με το MOEM, όπου ένας κόμβος μπορεί να δείχνεται από οποιονδήποτε αριθμό πολυδιάστατων κόμβων.

σημαίνει ότι το `language` είναι ένα όνομα διάστασης, και ότι το πεδίο ορισμού της είναι το σύνολο `{English, French}`. Εναλλακτικά, είναι δυνατόν να δηλωθεί το πεδίο ορισμού μιας διάστασης χε την χρήση ενός URI, όπως στο

```
<!DIMENSION branch_no "http://company_server/MXML/branches.html">
```

## 7.2.2 Δήλωση στοιχείων

Για να δηλωθούν συμβατικά στοιχεία στο MDTD χρησιμοποιείται το συντακτικό του συμβατικού DTD. Οι δηλώσεις πολυδιάστατων στοιχείων καλύπτουν και τα πολυδιάστατα στοιχεία και τα στοιχεία περιβάλλοντος. Οι δηλώσεις πολυδιάστατων στοιχείων έχουν την μορφή:

```
<!MULTIELEMENT element_name associated_dimensions type_decl>
```

Το σύνολο των διαστάσεων από το οποίο εξαρτάται το πολυδιάστατο στοιχείο εμφανίζεται στην θέση `associated_dimensions`. Για παράδειγμα, στην παρακάτω δήλωση

```
<!MULTIELEMENT comment {language,detail} (#PCDATA)>
```

το στοιχείο `comment` ορίζεται σαν ένα πολυδιάστατο στοιχείο που εξαρτάται από τις διαστάσεις `language` και `detail`.

Χρησιμοποιώντας δηλώσεις πολυδιάστατων στοιχείων, είναι δυνατόν να ορίσει κανείς διαφορετικούς δομικούς περιορισμούς για κάθε ένα από τα στοιχεία περιβάλλοντος ενός πολυδιάστατου στοιχείου. Σαν παράδειγμα, στην παρακάτω δήλωση

```
<!MULTIELEMENT ingredient {season,occasion}
  [occasion=special] ((name,remarks?) | #PCDATA) [/]
  [default] (#PCDATA) [/]
>
```

ο τύπος του στοιχείου `ingredient` δηλώνεται σαν `(name,remarks?) | #PCDATA` όποτε η τιμή της διάστασης `occasion` είναι `special`: κάτω από οποιονδήποτε άλλο κόσμο, ο τύπος του στοιχείου `ingredient` δηλώνεται σαν `(#PCDATA)`.

## 7.2.3 Δήλωση γνωρισμάτων

Οι δηλώσεις γνωρισμάτων έχουν επεκταθεί προκειμένου να πάρουν υπόψη τους πολυδιάστατα γνωρίσματα. Για παράδειγμα, στην δήλωση

```
<!ATTLIST salad
  name CDATA #REQUIRED
  vegetarian {season} CDATA #IMPLIED
>
```

το στοιχείο `salad` δηλώνεται με δύο γνωρίσματα, τα `name` και `vegetarian`. Η τιμή του γνωρίσματος `name` δεν εξαρτάται από διαστάσεις, ενώ η τιμή του γνωρίσματος `vegetarian` εξαρτάται από την διάσταση `season`.

Οι δηλώσεις γνωρισμάτων επιτρέπουν να ορίσει κανείς ότι ένα γνώρισμα είναι παρόν κάτω από κάποια ερμηνευτικά περιβάλλοντα, ενώ παράλληλα είναι απόν κάτω από άλλα ερμηνευτικά περιβάλλοντα. Για παράδειγμα, στην δήλωση

```
<!ATTLIST ingredient
  [occasion=special] special_supplier {season} IDREF #REQUIRED [/]
>
```

το στοιχείο `ingredient` δηλώνεται να έχει το γνώρισμα `special_supplier` μόνο κάτω από τους κόσμους στους οποίους η τιμή της διάστασης `occasion` είναι `special`. Σε όλους τους άλλους κόσμους το στοιχείο `ingredient` δεν έχει γνωρίσματα.

Η δήλωση

```
<!ATTLIST @element_name attribute_name ID>
```

χρησιμοποιεί τον χαρακτήρα `@` για να δηλώσει ένα γνώρισμα τύπου “ID” για κάποιο πολυδιάστατο στοιχείο.

## 7.2.4 Παράδειγμα MDTD

Στο Παράδειγμα 7.2 παρουσιάζουμε ένα MDTD για το έγγραφο MXML στο Παράδειγμα 7.1.

### Παράδειγμα 7.2: Ένα MDTD για το MXML στο Παράδειγμα 7.1.

```
<!DOCTYPE menuDTD [
<!DIMENSION language {English,French}>
<!DIMENSION detail {low,high,exhaustive}>
<!DIMENSION season {spring,summer,fall,winter}>
<!DIMENSION occasion {special,normal}>

<!ELEMENT restaurant (menu | supplier)*>
<!ELEMENT menu (salad+, first+, maindish+, dessert+)>
<!ELEMENT salad (comment?, price, ingredient*)>
<!ATTLIST salad
  name CDATA #REQUIRED
  vegetarian {season} CDATA #IMPLIED>

<!MULTIELEMENT comment {language,detail} (#PCDATA)>
<!MULTIELEMENT price {season} (#PCDATA)>
<!MULTIELEMENT ingredient {season,occasion}
  [occasion=special] ((name, remarks?) | #PCDATA) [/]
  [default] (#PCDATA) [/]>
<!ATTLIST ingredient
  [occasion=special]
  special_supplier {season} IDREF #REQUIRED
  [/]>
<!ELEMENT name (#PCDATA)>
<!ELEMENT remarks (#PCDATA)>

<!ELEMENT supplier (name, address)>
<!ATTLIST supplier
  scode ID #REQUIRED>
]>
```

◆

Το MDTD στο Παράδειγμα 7.2 δηλώνει ότι το έγγραφο στο Παράδειγμα 7.1 εξαρτάται από τέσσερις διαστάσεις: την διάσταση `language` με πεδίο ορισμού `{English,French}`, την διάσταση `detail` με πεδίο ορισμού `{low,high,exhaustive}`, την διάσταση `season` με πεδίο ορισμού `{spring,summer,fall,winter}`, και την διάσταση `occasion` με

πεδίο ορισμού {special, normal}. Εκτός από συμβατικά στοιχεία όπως το menu και το salad, δηλώνονται και τα πολυδιάστατα στοιχεία comment, price, και ingredient.

### 7.3 ΙΔΙΟΤΗΤΕΣ ΤΗΣ MXML

Είναι φανερό ότι τα έγγραφα MXML μπορούν να μετατραπούν σε Πολυδιάστατους Γράφους Δεδομένων, αν θεωρήσουμε ότι τα πολυδιάστατα στοιχεία και τα πολυδιάστατα γνωρίσματα της MXML αντιστοιχούν σε πολυδιάστατους κόμβους. Τα γνωρίσματα της MXML με τύπο “IDREF” και “IDREFS” αναπαριστώνται στον Πολυδιάστατο Γράφο Δεδομένων σαν ακμές που δείχνουν στο αναφερόμενο αντικείμενο: ακμές οντότητας για συμβατικά γνωρίσματα, και ακμές περιβάλλοντος για πολυδιάστατα γνωρίσματα<sup>113</sup>.

Ο Πολυδιάστατος Γράφος Δεδομένων και το MOEM έχουν μεγαλύτερη ευελιξία στην αναπαράσταση των MSSD από την MXML. Λόγω των δεδομένο-κεντρικών καταβολών της XML, το συντακτικό της MXML επιβάλλει περιορισμούς που δεν συναντώνται στον Πολυδιάστατο Γράφο Δεδομένων και στο MOEM. Για παράδειγμα, δεν είναι δυνατόν να αναπαρασταθεί σε MXML ένα μονοπάτι τριών ή περισσότερων συνεχόμενων ακμών περιβάλλοντος<sup>114</sup>. Παρ’ όλα αυτά, ένας Πολυδιάστατος Γράφος Δεδομένων μπορεί πάντα να εκφραστεί σε MXML εάν θεωρήσουμε την κανονική του μορφή.

Ιδιότητες των Πολυδιάστατων Γράφων Δεδομένων ισχύουν και για την MXML. Οι έννοιες του ρητού περιβάλλοντος, του κληρονομούμενου περιβάλλοντος, της κάλυψης περιβάλλοντος, της κληρονομούμενης κάλυψης, της κληρονομούμενης κάλυψης μονοπατιού, και της αιτιοκρατίας ως προς το περιβάλλον είναι οι ίδιες στην MXML όπως στον Πολυδιάστατο Γράφο Δεδομένων. Επιπρόσθετα, η διαδικασία της αναγωγής σε OEM και της μερικής αναγωγής μπορούν να προσαρμοστούν για την MXML. Κάθε έγγραφο MXML αντιπροσωπεύει στην πραγματικότητα ένα σύνολο συμβατικών εγγράφων XML. Δεδομένου ενός κόσμου  $w$ , ένα έγγραφο MXML μπορεί να αναχθεί σε ένα συμβατικό έγγραφο XML, το οποίο είναι η έκφανση του πολυδιάστατου εγγράφου κάτω από τον  $w$ . Το Παράδειγμα 7.3 επιδεικνύει την **αναγωγή σε XML**, και δείχνει την έκφανση συμβατικής XML του MXML στο Παράδειγμα 7.1 που υφίσταται κάτω από τον κόσμο {(language, English), (detail, low), (season, summer), (occasion, special)}.

#### Παράδειγμα 7.3: Μία XML έκφανση του MXML στο Παράδειγμα 7.1.

```
<restaurant>
  <menu>
    <salad name="Chef's salad" vegetarian="yes" >
      <comment>A traditional salad.</comment>
      <price>3 EUR</price>
      <ingredient>tomato</ingredient>
      <ingredient>olive oil</ingredient>
      <ingredient special_supplier="sp1" >
        <name>special sauce</name>
        <remarks>Must order three days in advance</remarks>
      </ingredient>

    ... other salad ingredients ...
```

<sup>113</sup> Σημειώστε ότι προκειμένου να μπορέσουμε να διακρίνουμε τα γνωρίσματα “IDREF” και “IDREFS” από άλλα γνωρίσματα, πρέπει να συμβουλευτούμε το αντίστοιχο MDTD, όπου δηλώνονται οι τύποι των γνωρισμάτων.

<sup>114</sup> Δύο διαδοχικές ακμές περιβάλλοντος μπορούν να παρασταθούν σαν ένα πολυδιάστατο γνώρισμα που αναφέρεται σε κάποιο πολυδιάστατο στοιχείο.

```

</salad>

... other menu items ...

</menu>

<supplier scode="sp1">
  <name>John Smith</name>
  <address>234 XYZ Street</address>
</supplier>

<supplier scode="sp2">Peter Brown</supplier>

<\restaurant>
♦

```

Η έννοια του **ορθά-σηματισμένου** MXML είναι μια επέκταση της έννοιας του ορθά-σηματισμένου XML [XML]. Τα κριτήρια του ορθού-σηματισμού της XML ισχύουν επίσης και για τα στοιχεία και τα γνωρίσματα της MXML. Επιπλέον, ένα ορθά-σηματισμένο έγγραφο MXML πρέπει: (a) να είναι αιτιοκρατικό ως προς το περιβάλλον, και (b) να περιέχει αποκλειστικά στοιχεία και γνωρίσματα που έχουν μη κενή κληρονομούμενη κάλυψη. Συνεπώς, *ένας Πολυδιάστατος Γράφος Δεδομένων που παριστά ένα ορθά-σηματισμένο MXML είναι MOEM.*

## 7.4 ΤΟ ΠΟΛΥΔΙΑΣΤΑΤΟ ΠΑΡΑΔΕΙΓΜΑ

Οι πληροφοριακές οντότητες του Ιστού που είναι δυνατόν να παρουσιάζουν διαφορετικές εκφάνσεις κάτω από διαφορετικά ερμηνευτικά περιβάλλοντα οδηγούν σε ένα νέο παράδειγμα χειρισμού και παρουσίασης δεδομένων του Ιστού που εξαρτώνται από το ερμηνευτικό περιβάλλον. Αναφερόμαστε στο νέο παράδειγμα σαν το **πολυδιάστατο παράδειγμα**. Στην ενότητα αυτή, εξηγούμε το παράδειγμα που προτείνουμε, και χρησιμοποιούμε έγγραφα MXML και **πολυδιάστατα XSL φύλλα διαμόρφωσης** για να επιδείξουμε την εφαρμογή του μέσα από ένα εκτενές παράδειγμα.

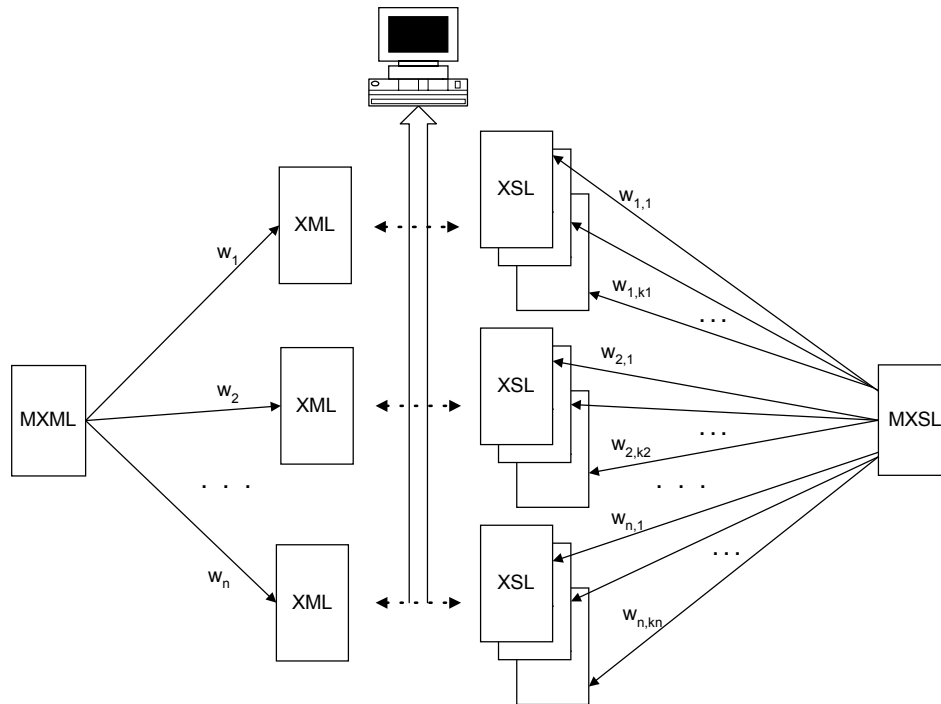
### 7.4.1 Παρουσίαση πολυδιάστατων ημιδομημένων δεδομένων

Η XML δεν αντιμετωπίζει το πως η πληροφορία παρουσιάζεται στον χρήστη. Η eXtensible Stylesheet Language [XSL], ή XSL για συντομία, δίνει λύση σε αυτό το θέμα. Ένα έγγραφο XSL περιέχει οδηγίες για το πως να παρουσιαστεί η πληροφορία εγγράφων XML. Επιπρόσθετα, είναι δυνατόν να επιλέξει κομμάτια δεδομένων από κάποιο έγγραφο XML και να τα μετατρέψει σε άλλο μορφότυπο (για παράδειγμα, HTML), χρησιμοποιώντας XSL Transformations [XSLT] (XSLT για συντομία). Είναι σημαντικό να σημειωθεί ότι τα XSL stylesheets είναι και τα ίδια έγγραφα XML, και ακολουθούν το συντακτικό της XML. Ένα φύλλο διαμόρφωσης XSL μπορεί να εφαρμοστεί πάνω σε συγκεκριμένο έγγραφο XML, και το αποτέλεσμα μπορεί να δειχθεί από έναν διαφυλλιστή του Ιστού. Στην πραγματικότητα, ένας αριθμός εγγράφων XSL μπορεί να εφαρμοστεί πάνω στο ίδιο έγγραφο XML, καταλήγοντας σε διαφορετικούς τρόπους παρουσίασης του εγγράφου ή τμημάτων αυτού.

Το πολυδιάστατο παράδειγμα επιτρέπει σε ένα έγγραφο να έχει έναν αριθμό εκφάνσεων, με κάθε μια να υφίσταται κάτω από διαφορετικούς κόσμους. Η πληροφορία σε ένα τέτοιο έγγραφο είναι κωδικοποιημένη με μια κατάλληλη γλώσσα επισήμειωσης, όπως η MXML. Από την στιγμή που καθορίζεται ένας κόσμος, το έγγραφο MXML μπορεί να αναχθεί σε ένα

συμβατικό έγγραφο XML που αποτελεί την έκφραση κάτω από τον κόσμο αυτό. Η αποσύνθεση ενός εγγράφου MXML σε έναν αριθμό συμβατικών εκφάνσεων XML απεικονίζεται στην Εικόνα 7.1, όπου οι κόσμοι υποδηλώνονται από τα  $w_1, w_2, \dots, w_n$ .

Εικόνα 7.1: MXML, MXSL, και η σχέση τους.



Αφού τα φύλλα διαμόρφωσης XSL είναι έγγραφα XML, οι ίδιες αρχές ισχύουν και για τα φύλλα διαμόρφωσης XSL. Ένα **πολυδιάστατο φύλλο διαμόρφωσης XSL**, ή **MXSL** για συντομία, κωδικοποιεί ένα σύνολο συμβατικών φύλλων διαμόρφωσης XSL, καθένα από τα οποία είναι η έκφραση του MXSL κάτω από έναν συγκεκριμένο κόσμο. Όπως και ένα συμβατικό XSL, ένα MXSL πρέπει να συνδεθεί με κάποιο έγγραφο XML ή MXML. Για κάθε πιθανό κόσμο, η υφιστάμενη έκφραση XSL εφαρμόζεται στην υφιστάμενη έκφραση XML για να εμφανίσει το έγγραφο κάτω από τον κόσμο αυτό. Η σχέση ανάμεσα στο MXML και στο MXSL δίνεται γραφικά στην Εικόνα 7.1. Σημειώστε ότι οι πιθανοί κόσμοι για ένα MXSL ίσως να μην είναι οι ίδιοι με αυτούς του αντίστοιχου MXML. Επιπλέον διαστάσεις στο MXSL μπορούν να χρησιμοποιηθούν για να ορίσουν εναλλακτικές παρουσιάσεις για το ίδιο έγγραφο XML. Από την άλλη, κάποιες από τις διαστάσεις σε ένα έγγραφο MXML μπορεί να εμφανίζονται και στο αντίστοιχο MXSL, προκειμένου να θεσπίσουν μια αντιστοιχία μεταξύ των υφισταμένων εκφάνσεων MXML και MXSL.

## 7.4.2 Ένα εκτενές παράδειγμα

Το Παράδειγμα 7.4 δείχνει ένα έγγραφο MXML σχετικά με ένα βιβλίο που υπάρχει σε δύο διαφορετικές εκδόσεις, μια αγγλική και μια ελληνική. Το στοιχείο `book` έχει έξι υποστοιχεία, που περιγράφονται ακολούθως. Το `isbn` και το `publisher` είναι πολυδιάστατα στοιχεία, και εξαρτώνται από την διάσταση `edition`. Τα στοιχεία `title` και `authors` παραμένουν τα ίδια κάτω από κάθε πιθανό κόσμο. Το στοιχείο `price` είναι ένα

πολυδιάστατο στοιχείο του οποίου η τιμή εξαρτάται από τις διαστάσεις `edition` και `customer_type`. Τέλος, το στοιχείο `translator` έχει υπόσταση μόνον κάτω από τους κόσμους όπου η `edition` έχει την τιμή `greek`.

**Παράδειγμα 7.4: Ένα MXML που περιέχει πολυδιάστατη πληροφορία για ένα βιβλίο.**

```
<book>
  <@isbn>
    [edition=greek] <isbn>0-13-110370-9</isbn> [/]
    [edition=english] <isbn>0-13-110362-8</isbn> [/]
  </@isbn>

  <title>The C programming language</title>

  <authors>
    <author>Brian W. Kernighan</author>
    <author>Dennis M. Ritchie</author>
  </authors>

  <@publisher>
    [edition=english] <publisher>Prentice Hall</publisher> [/]
    [edition=greek] <publisher>Klidarithmos</publisher> [/]
  </@publisher>

  <@translator>
    [edition=greek] <translator>Thomas Moraitis</translator> [/]
  </@translator>

  <@price>
    [edition=english,customer_type=individual]
      <price>38</price> [/]
    [edition=english,customer_type=library]
      <price>29.4</price> [/]
    [edition=english,customer_type=student]
      <price>34</price> [/]
    [edition=greek,customer_type=individual]
      <price>15</price> [/]
    [edition=greek,customer_type=library]
      <price>8.8</price> [/]
    [edition=greek,customer_type=student]
      <price>13</price> [/]
  </@price>
</book>
```



Το MXML στο ακόλουθο Παράδειγμα 7.5 είναι ένα φύλλο διαμόρφωσης MXSL για το έγγραφο MXML στο Παράδειγμα 7.4, και καθορίζει το πώς παρουσιάζονται οι διάφορες εκφάνσεις του MXML στο Παράδειγμα 7.4.

**Παράδειγμα 7.5: Ένα MXSL για το MXML στο Παράδειγμα 7.4.**

```
<xsl:template match="/">
  <DIV STYLE=[size=large] "font-size:22pt" [/]
    [size=normal] "font-size:18pt" [/] >
    Book
  </DIV>
```

```

<@SPAN>
  [customer_type=library]
    <SPAN STYLE="font-size:15pt">
      ISBN: <xsl:value-of select = "book/isbn"/>,
    </SPAN>
  [/]

  [customer_type in {individual,student}]
    <SPAN STYLE="font-size:15pt">
      Title: <xsl:value-of select="book/title"/>,
      Authors: <xsl:value-of select="book/authors"/>,
      <@wrapper>
        [edition=greek]
          <wrapper>
            Translator: <xsl:value-of select="book/translator"/>,
          </wrapper>
        [/]
      </@wrapper>
    </SPAN>
  [/]
</@SPAN>

<SPAN STYLE="font-size:15pt">
  Publisher: <xsl:value-of select="book/publisher"/>,
</SPAN>

<SPAN STYLE="font-size:15pt">
  Price: <xsl:value-of select="book/price"/>
</SPAN>
</xsl:template>

```



Το ISBN δείχνεται μόνο εάν η αίτηση για πληροφορία έχει προέλθει από κάποια βιβλιοθήκη (library), ενώ το title και το authors δείχνονται αν ο ενδεχόμενος πελάτης είναι ιδιώτης (individual) ή φοιτητής (student). Παρατηρήστε πως το στοιχείο wrapper, που είναι ορισμένο σαν μέρος του XSL<sup>115</sup>, μπορεί επίσης να χρησιμοποιηθεί με κομψό τρόπο στο MXSL για να αποκλείσει το translator στην περίπτωση που η αίτηση αφορά έκδοση του βιβλίου στην πρωτότυπη γλώσσα. Στην γενική περίπτωση, το MXSL μπορεί να περιέχει πολυδιάστατες εκδοχές στοιχείων και γνωρισμάτων ορισμένων για το συμβατικό XSL. Τέλος, τα στοιχεία publisher και price δείχνονται σε κάθε περίπτωση.

Δεν υπάρχει κανείς περιορισμός ως προς το ποιες διαστάσεις εμφανίζονται στους προσδιοριστές περιβάλλοντος ενός MXSL. Μερικές από αυτές μπορεί να εμφανίζονται και στο αντίστοιχο έγγραφο MXML, όπως στην περίπτωση των διαστάσεων customer\_type και edition στο Παράδειγμα 7.5, ή μπορεί να είναι διαφορετικές, όπως στην περίπτωση της διάστασης size στο ίδιο παράδειγμα, η οποία δεν υπάρχει στο MXML.

Κάτω από τον κόσμο  $w = \{(edition, greek), (customer\_type, student), (size, large)\}$ , το έγγραφο MXML στο Παράδειγμα 7.4 ανάγεται στο συμβατικό έγγραφο XML που φαίνεται στο Παράδειγμα 7.6. Το γεγονός ότι η διάσταση size δεν εμφανίζεται στο MXML στο Παράδειγμα 7.4 δεν παρουσιάζει κάποιο πρόβλημα, αφού ολόκληρο το πεδίο ορισμού της size υπονοείται στους προσδιοριστές περιβάλλοντος.

<sup>115</sup> Όπως ορίζεται στο XSL, το στοιχείο wrapper δεν έχει κάποια σημασιολογία αλλά δρα σαν «φορέας» ιδιοτήτων μορφοποίησης, οι οποίες εφαρμόζονται στα περιεχόμενά του.



**Παράδειγμα 7.6: Μία XML έκφραση του MXML στο Παράδειγμα 7.4.**

```

<book>
  <isbn>0-13-110370-9</isbn>
  <title>The C programming language</title>
  <authors>
    <author>Brian W. Kernighan</author>
    <author>Dennis M. Ritchie</author>
  </authors>
  <publisher>Klidarithmos</publisher>
  <translator>Thomas Moraitis</translator>
  <price>13</price>
</book>

```

◆

Κάτω από τον ίδιο κόσμο w, το φύλλο διαμόρφωσης MXSL στο Παράδειγμα 7.5 ανάγεται στο φύλλο διαμόρφωσης XSL που φαίνεται στο Παράδειγμα 7.7.

**Παράδειγμα 7.7: Μία XSL έκφραση του MXSL στο Παράδειγμα 7.5.**

```

<xsl:template match="/">
  <DIV STYLE="font-size:22pt">Book</DIV>
  <SPAN STYLE="font-size:15pt">
    Title: <xsl:value-of select="book/title"/>,
    Authors: <xsl:value-of select="book/authors"/>,
    <wrapper>
      Translator: <xsl:value-of select="book/translator"/>,
    </wrapper>
  </SPAN>
  <SPAN STYLE="font-size:15pt">
    Publisher: <xsl:value-of select="book/publisher"/>,
  </SPAN>
  <SPAN STYLE="font-size:15pt">
    Price: <xsl:value-of select="book/price"/>
  </SPAN>
</xsl:template>

```

◆

Το αποτέλεσμα της εφαρμογής του φύλλου διαμόρφωσης από το Παράδειγμα 7.7 στο έγγραφο XML από το παράδειγμα Παράδειγμα 7.6 μοιάζει ως εξής:

## Book

Title: The C programming language, Authors: Brian W. Kernighan, Dennis M. Ritchie, Translator: Thomas Moraitis, Publisher: Klidarithmos, Price: 13

Τέλος, το αποτέλεσμα μιας ανάλογης διεργασίας για τον κόσμο { (edition, english), (customer\_type, library), (size, normal) } θα μοιάζει ως εξής:

## Book

ISBN: 0-13-110362-8, Publisher: Prentice Hall, Price: 29.4

### 7.5 ΠΡΟΤΥΠΗ ΥΛΟΠΟΙΗΣΗ

Στην ενότητα αυτή, περιγράφουμε ένα πρότυπο σύστημα που επιδεικνύει τις βασικές αρχές του πολυδιάστατου παραδείγματος που προτείναμε. Το σύστημα που υλοποιήσαμε λέγεται *MXML Web Server*, και είναι ένας εξυπηρετητής Ιστού ικανός να χειριστεί πολυδιάστατα δεδομένα MXML και MXSL.

Σε ένα τυπικό σενάριο, ο χρήστης ζητάει ένα έγγραφο MXML μέσω ενός συμβατικού διαφυλλιστή Ιστού, και ερωτάται να επιλέξει τιμές για κάθε μια από τις διαστάσεις που σχετίζονται με το ζητούμενο έγγραφο. Αφού ο χρήστης ορίσει έναν κόσμο, ο εξυπηρετητής στέλνει τις αντίστοιχες εκφάνσεις XML και XSL για να εμφανιστούν από τον διαφυλλιστή. Ο χρήστης μπορεί να αλλάξει τις τιμές των διαστάσεων, και να παρατηρήσει πως διαφορετικοί κόσμοι συσχετίζονται με διαφορετικές όψεις του ίδιου πολυδιάστατου εγγράφου.

Ένα σημαντικό σημείο είναι ότι, στο πολυδιάστατο παράδειγμα η διεργασία αναγωγής των MXML και MXSL μπορεί να λάβει χώρα στον εξυπηρετητή, στον πελάτη, ή και στους δύο. Το σύστημά μας υλοποιεί την λειτουργία αυτή στην μεριά του εξυπηρετητή κυρίως για λόγους συμβατότητας με τους υπάρχοντες διαφυλλιστές του Ιστού. Στην γενική περίπτωση πάντως, κάποιες από τις διαστάσεις μπορούν να ληφθούν υπόψη στην μεριά του εξυπηρετητή με την χρήση μερικής αναγωγής, για να απαλειφθούν μη σχετικά δεδομένα και να μειωθεί το μέγεθος της απάντησης, ενώ οι υπόλοιπες διαστάσεις θα μπορούσαν να τύχουν χειρισμού στην μεριά του πελάτη, για λόγους μυστικότητας (privacy) ή για την ελαχιστοποίηση του αριθμού επόμενων αιτήσεων για παρεμφερή πληροφορία. Αυτή η ευελιξία θα μπορούσε να αξιοποιηθεί με την χρήση της MXML σε πεδία όπως αυτά του ηλεκτρονικού εμπορίου και της μοντελοποίησης χρηστών.

#### 7.5.1 Επεκτείνοντας τα URL

Το URL [URL], που είναι συντομογραφία για το Uniform Resource Locator, είναι ο καθιερωμένος τρόπος για να προσδιοριστεί ένας πόρος που είναι διαθέσιμος στο Διαδίκτυο. Επεκτείναμε το συντακτικό του URL ώστε να είναι σε θέση να χειριστεί πολυδιάστατη πληροφορία. Το επεκτεταμένο URL έχει την μορφή:

```
http://<host>:<port>/<path><context>?<search>
```

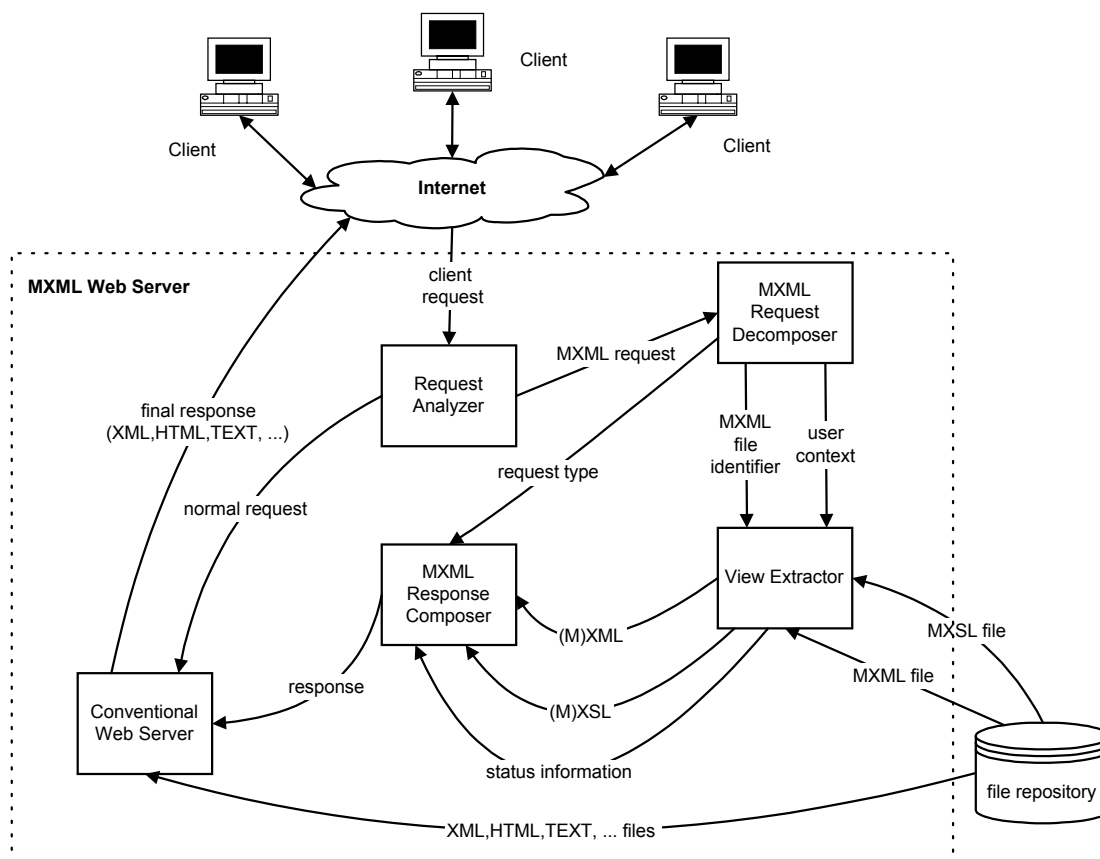
Η μόνη διαφορά από το συμβατικό URL είναι η προσθήκη του συμβόλου <context>. Το ακόλουθο παράδειγμα επεκτεταμένου URL είναι ένα αίτημα για την έκφανση του εγγράφου books.mxml, για την οποία το edition είναι english και το customer\_type είναι student.

```
http://myserver/books.mxml [edition=english, customer_type=student]
```

## 7.5.2 Σχεδιασμός συστήματος

Το σύστημα εμπεριέχει τις μονάδες λογισμικού που παρουσιάζονται στην Εικόνα 7.2. Οι μονάδες αυτές είναι: *Request Analyzer*, *MXML Request Decomposer*, *View Extractor*, *MXML Response Composer*, και *Conventional Web Server*. Στις επόμενες παραγράφους θα περιγράψουμε σύντομα την λειτουργικότητα καθεμιάς μονάδας, καθώς και τις υπομονάδες τους, και θα συζητήσουμε τον ρόλο τους στην συνολική λειτουργία του συστήματος.

Εικόνα 7.2: Αρχιτεκτονική *MXML Web Server*.



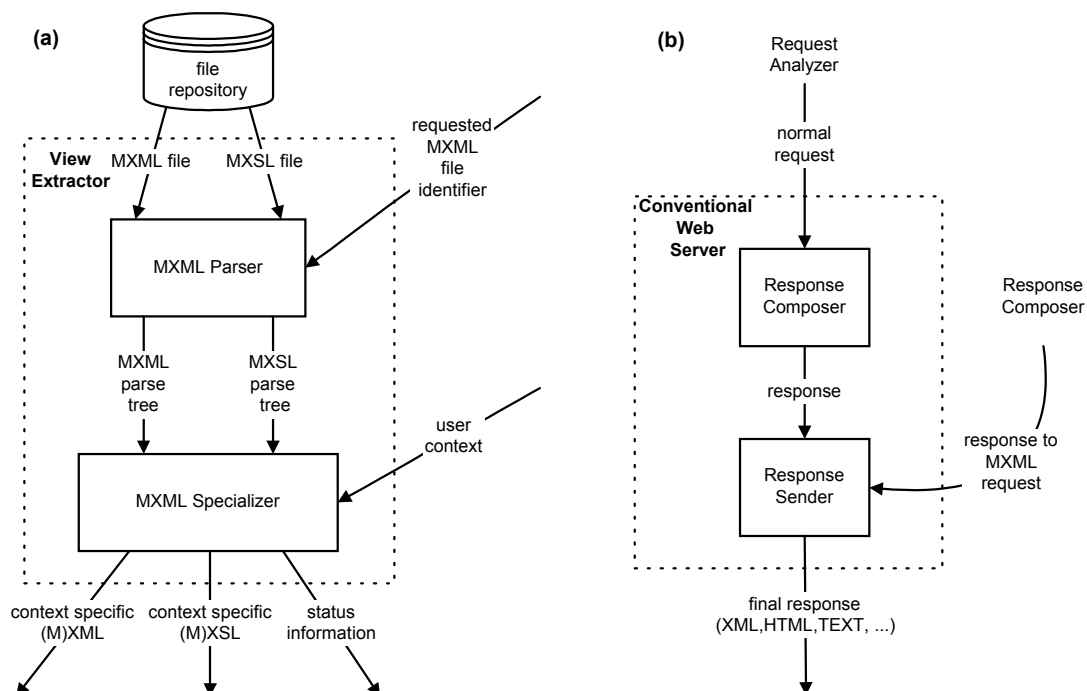
*Request Analyzer*: Η μονάδα *Request Analyzer* είναι υπεύθυνη για να αποφασίζει τον τύπο των αιτημάτων του πελάτη. Υπάρχουν δύο τύποι αιτημάτων: *normal requests*, που αναφέρονται σε HTML, XML, φακέλους κειμένου κτλ., και *MXML-requests*, που αναφέρονται σε φακέλους MXML. Αν το αίτημα είναι τύπου *normal*, τότε το χειρίζεται η μονάδα *Conventional Web Server*. Διαφορετικά, αν το αίτημα αφορά κάποιο έγγραφο MXML, τότε καλείται η μονάδα *MXML Request Decomposer*.

*MXML Request Decomposer*: Η μονάδα αυτή διασπά αιτήματα τύπου MXML σε ενότητες. Οι τρεις ενότητες που συνθέτουν ένα αίτημα τύπου MXML είναι: το *MXML File Identifier*, το *User Context*, και το *Request Type*. Οι δύο πρώτες ενότητες αποστέλλονται στην μονάδα *View Extractor*, ενώ η Τρίτη στέλνεται κατευθείαν στην μονάδα *MXML Response Composer*.

*View Extractor*: Αυτή η μονάδα συνδυάζει την πληροφορία που παρέχεται από την μονάδα MXML Request Decomposer με τον αντίστοιχο φάκελο MXML ή MXSL, προκειμένου να εξυπηρετήσει το αίτημα. Η μονάδα View Extractor αποτελείται από δύο υπομονάδες, που ονομάζονται *MXML Parser* και *MXML Specializer*, όπως φαίνεται στην Εικόνα 7.3 (a).

- *MXML Parser*: Η υπομονάδα MXML Parser χρησιμοποιεί την ενότητα MXML File Identifier του αιτήματος, που παρέχεται από το MXML Request Decomposer, για να προσπελάσει τον αντίστοιχο φάκελο MXML ή MXSL στο File Repository. Ο ρόλος της είναι να αναλύσει συντακτικά το περιεχόμενο και να δημιουργήσει το ανάλογο συντακτικό δένδρο MXML ή MXSL.
- *MXML Specializer*: Η υπομονάδα MXML Specializer χρησιμοποιεί το συντακτικό δένδρο που δημιουργείται από το MXML Parser, μαζί με το User Context, που παρέχεται από το MXML Request Decomposer, για να φτιάξει έναν φάκελο (M)XML ή (M)XSL με προσδιορισμένο ερμηνευτικό περιβάλλον. Εφόσον το User Context αντιπροσωπεύει έναν μόνο κόσμο χρησιμοποιείται αναγωγή σε XML, ενώ αν το User Context αντιπροσωπεύει περισσότερους από έναν κόσμους χρησιμοποιείται μερική αναγωγή. Ο τύπος του παραγόμενου φακέλου περινέται στο *MXML Response Composer* μέσω του *status information*.

**Εικόνα 7.3:** Οι υπομονάδες του *View Extractor* εμφανίζονται στο (a), και οι υπομονάδες του *Conventional Web Server* εμφανίζονται στο (b).



*MXML Response Composer*: Η μονάδα MXML Response Composer κατασκευάζει την απάντηση και την στέλνει στο *Response Sender*, που διακομίζει την απάντηση στον πελάτη.

*Conventional Web Server*: Η μονάδα Conventional Web Server υλοποιεί πολλά από τα σημαντικά χαρακτηριστικά ενός συμβατικού εξυπηρετητή Ιστού. Αποτελείται από δύο

υπομονάδες, που φαίνονται στην Εικόνα 7.3 (b). Σε περίπτωση που ο τύπος του αιτήματος είναι *normal*, την απάντηση κατασκευάζει η υπομονάδα Response Composer της μονάδας Conventional Web Server, ενώ σε περίπτωση που ο τύπος του αιτήματος είναι *MXML-request*, η απάντηση κατασκευάζεται στο MXML Response Composer και αποστέλλεται στην υπομονάδα *Response Sender* της μονάδας Conventional Web Server.

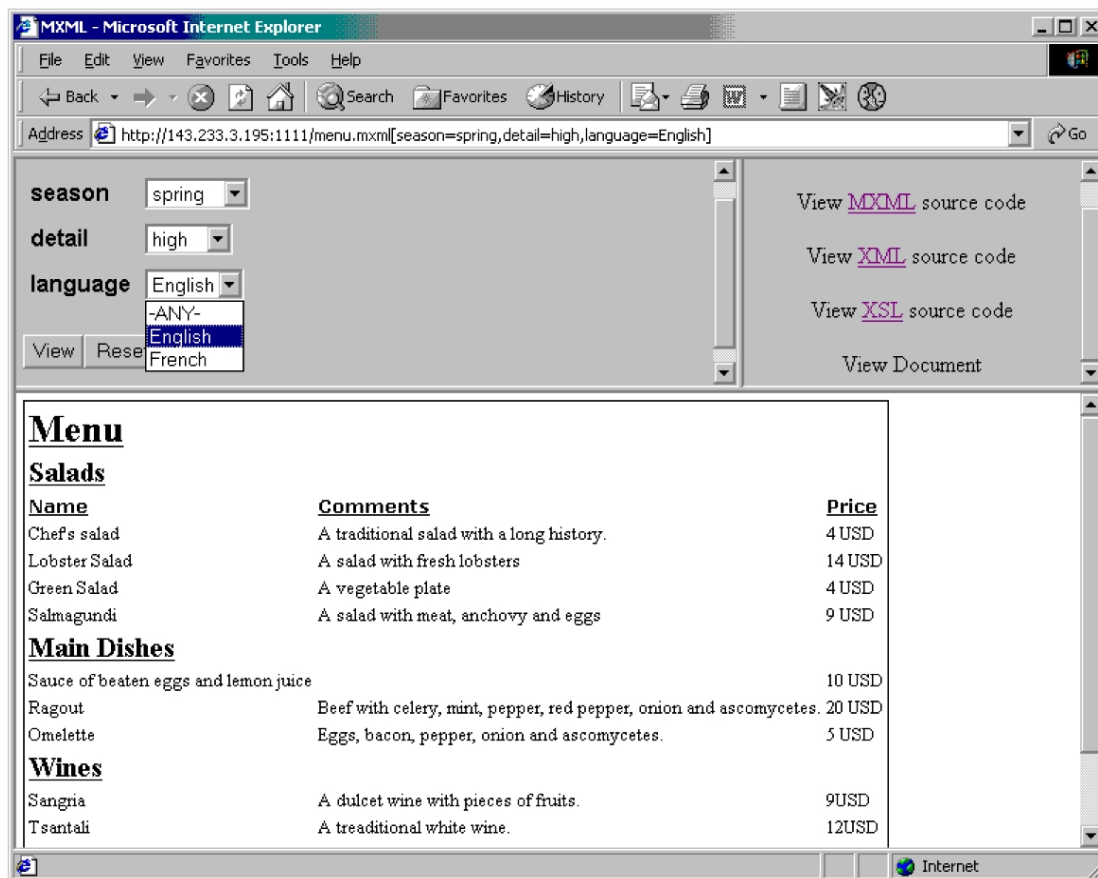
- *Response Composer*: Η υπομονάδα Response Composer αναλύει αιτήματα τύπου *normal*, και κατασκευάζει τις απαντήσεις που αποστέλλονται στον πελάτη. Όπως απεικονίζεται στην Εικόνα 7.3 (b), αυτή η υπομονάδα εμπλέκεται μόνο αν το αίτημα είναι τύπου *normal*.
- *Response Sender*: Η υπομονάδα Response Sender είναι υπεύθυνη για την αποστολή της απάντησης στον πελάτη. Όπως αναφέρθηκε πιο πάνω, η απάντηση μπορεί να προέλθει είτε από το Response Composer, ή από το MXML Response Composer.

### 7.53 Υλοποίηση συστήματος

Τα τμήματα του συστήματος που περιγράφηκαν στην προηγούμενη ενότητα έχουν στην πραγματικότητα υλοποιηθεί σαν δύο χωριστά προγράμματα. Το MXML Web Server, εκτός από το View Extractor, αναπτύχθηκε σε Java [JAVA, CL97]. Το View Extractor έχει αναπτυχθεί σε C [KR88].

Όταν υπάρξει ένα αίτημα για κάποιο έγγραφο MXML (όπως, για παράδειγμα, το έγγραφο `menu.mxml`), το MXML Web Server προσδιορίζει τις διαστάσεις που εμπλέκονται και τα αντίστοιχα πεδία ορισμού των ελέγχοντας το σχετικό MDTD, εάν αυτό υπάρχει, ή διατρέχοντας το ίδιο το έγγραφο MXML, και προσθέτοντας πιθανές επιπλέον διαστάσεις που εμφανίζονται στο έγγραφο MXSL. Έπειτα, το MXML Web Server παρουσιάζει στον χρήστη λίστες για την ανάθεση τιμών στις διαστάσεις, όπως απεικονίζεται στο *dimension frame*, το πάνω αριστερό πλαίσιο στην Εικόνα 7.4.

Εικόνα 7.4: Ένα στιγμιότυπο οθόνης που εμφανίζει μια απάντηση από τον MXML Web Server.



Η Εικόνα 7.4 είναι ένα στιγμιότυπο οθόνης του MXML Web Server καθώς απαντάει στο αίτημα:

```
http://143.233.3.195:1111/menu.mxml [season=spring,
                                     detail=high, language=English]
```

Ο προσδιοριστής περιβάλλοντος σε αυτό το επεκτεταμένο URL συντίθεται δυναμικά από κώδικα JavaScript [JSCR], με βάση τις τιμές που έχει επιλέξει ο χρήστης για την λίστα των διαστάσεων στο πάνω αριστερό *dimension frame*. Παρατηρήστε ότι για κάθε διάσταση υπάρχει μια τιμή “ANY”. Η σημασία του “ANY” είναι ότι δεν έχει προσδιοριστεί καμιά τιμή για την συγκεκριμένη διάσταση. Συνεπώς, εάν ένας προσδιοριστής περιβάλλοντος περιέχει ένα ή περισσότερα “ANY”, καθορίζει περισσότερους από έναν κόσμους. Εάν δεν περιέχει καθόλου “ANY”, καθορίζει ακριβώς έναν κόσμο.

Το πάνω δεξιά *action frame* επιτρέπει στον χρήστη να δει τον πηγαίο κώδικα του MXML, καθώς και του MXSL. Αν το επιλεγμένο ερμηνευτικό περιβάλλον καθορίζει ακριβώς έναν κόσμο, τότε ο χρήστης μπορεί επιπλέον να δει τον κώδικα για τα XML και XSL των αντίστοιχων εκφάνσεων κάτω από τον κόσμο αυτόν, συν το τελικό έγγραφο του αποτελέσματος. Ο πηγαίος κώδικας και το τελικό έγγραφο εμφανίζονται στο πλαίσιο του κάτω μέρους της οθόνης που λέγεται *output frame*, και που στην Εικόνα 7.4 δείχνει το πολυδιάστατο μενού ενός εστιατορίου κάτω από τον κόσμο  $\{(season, spring), (detail, high), (language, English)\}$ . Αν το ερμηνευτικό περιβάλλον που ορίστηκε

από τον χρήστη αντιπροσωπεύει περισσότερους από έναν κόσμους, τότε το *output frame* δείχνει τα MXML και MXSL που παρήχθησαν μετά από μερική αναγωγή για αυτό το ερμηνευτικό περιβάλλον<sup>116</sup>. Στην περίπτωση αυτή, το *output frame* εμφανίζει αντί για το τελικό έγγραφο ένα συγκεκριμένο μήνυμα, που δηλώνει ότι προκειμένου να εμφανιστεί το τελικό έγγραφο ο χρήστης πρέπει να δώσει τιμή διάφορη του “ANY” σε κάθε διάσταση.

## 7.6 ΠΕΡΙΛΗΨΗ

Στο κεφάλαιο αυτό προτείναμε την **Πολυδιάστατη XML**, ή **MXML** για συντομία, μια επέκταση της XML κατάλληλη για την αναπαράσταση δεδομένων που παρουσιάζουν εκφάνσεις με διαφορετική τιμή ή δομή κάτω από διαφορετικά ερμηνευτικά περιβάλλοντα. Στην MXML, στοιχεία και γνωρίσματα είναι δυνατόν να εξαρτώνται από έναν αριθμό διαστάσεων, οι οποίες ορίζουν τους κόσμους κάτω από τους οποίους υφίστανται οι εκφάνσεις των στοιχείων και των γνωρισμάτων αυτών. Επιπλέον, προτείναμε μια επέκταση του DTD, που ονομάζεται **Πολυδιάστατο DTD** ή **MDTD** για συντομία, το οποίο περιγράφει την λογική δομή των εγγράφων MXML. Συζητήσαμε τις ιδιότητες της MXML και τις σχέσεις της με τον Πολυδιάστατο Γράφο Δεδομένων και το MOEM, και δώσαμε παραδείγματα του πως ένα έγγραφο MXML μπορεί να αναχθεί σε ένα συμβατικό έγγραφο XML κάτω από έναν δεδομένο κόσμο.

Επιπρόσθετα, παρουσιάσαμε ένα νέο παράδειγμα για τον χειρισμό δεδομένων στον Παγκόσμιο Ιστό που εξαρτώνται από ερμηνευτικά περιβάλλοντα και παρουσιάζουν πολλαπλές εκφάνσεις. Το νέο παράδειγμα λέγεται **πολυδιάστατο παράδειγμα**, και περιλαμβάνει ζητήματα αναπαράστασης, διαχείρισης, και παρουσίασης. Συζητήσαμε την παρουσίαση εγγράφων MXML με την βοήθεια **πολυδιάστατων XSL (MXSL)** φύλλων διαμόρφωσης. Περιγράψαμε τον σχεδιασμό ενός συστήματος που υλοποιεί την βασική λειτουργικότητα του πολυδιάστατου παραδείγματος, και επιδεικνύει πως ο χρήστης μπορεί να αλληλεπιδράσει με ένα πολυδιάστατο έγγραφο και να δει διαφορετικές παραλλαγές του κάτω από διαφορετικούς κόσμους.

<sup>116</sup> Δεν είναι δυνατόν για τον χρήστη να καθορίσει *οποιοδήποτε* πιθανό ερμηνευτικό περιβάλλον. Για παράδειγμα, το ερμηνευτικό περιβάλλον [season in {spring,winter} | season=summer, detail=low] δεν μπορεί να σχηματιστεί με το *dimension frame* στην Εικόνα 7.4. Ο κυρίως στόχος του υλοποιημένου συστήματος είναι η αναγωγή σε XML κάτω από έναν συγκεκριμένο κόσμο, ενώ η μερική αναγωγή για ένα σύνολο κόσμων υποστηρίζεται μόνο οριακά.





## 8 ΕΝΑ ΣΥΣΤΗΜΑ ΓΙΑ ΤΗΝ ΔΙΑΧΕΙΡΙΣΗ MSSD

Στο κεφάλαιο αυτό, παρουσιάζουμε ένα σύστημα που υλοποιεί ουσιαστικές έννοιες από προηγούμενα κεφάλαια, και υποστηρίζει την διαχείριση πολυδιάστατων ημιδομημένων δεδομένων. Συγκεκριμένα, μέσω των λειτουργιών βασικών αλλαγών του MOEM που ορίστηκαν στο Κεφάλαιο 6, το σύστημα επιτρέπει την ανάπτυξη και συντήρηση Πολυδιάστατων Γράφων Δεδομένων, οι οποίοι εμφανίζονται σε ένα γραφικό διαδραστικό περιβάλλον. Υλοποιεί πράξεις περιβάλλοντος όπως αυτές εισήχθησαν στο Κεφάλαιο 3, και υπολογίζει το κληρονομούμενο περιβάλλον των γράφων, όπως ορίστηκε στο Κεφάλαιο 4. Εκτελεί αναγωγή σε OEM, μερική αναγωγή, και μετατροπή σε κανονική μορφή· τα αποτελέσματα αυτών των λειτουργιών εμφανίζονται σαν νέοι γράφοι στο ίδιο διαδραστικό περιβάλλον, και μπορούν να αποτελέσουν το αντικείμενο περαιτέρω επεξεργασίας όπως ακριβώς και οι αρχικοί γράφοι. Το σύστημα μπορεί να φορτώσει και να σώσει MOEMs σε διάφορους μορφότυπους, συμπεριλαμβανομένων των εκφράσεων mssd-expressions και της MXML. Επιπλέον, συμπεριλαμβάνει ένα υποσύστημα επερωτήσεων που αποτιμά επερωτήσεις MQL, όπως περιγράφεται στο Κεφάλαιο 5.

Το σύστημα έχει υλοποιηθεί σε Java [JAVA, CL97], και μπορεί να λειτουργήσει σαν υποδομή για την ανάπτυξη νέων εφαρμογών και εργαλείων για MSSD που χρειάζεται να χρησιμοποιήσουν τις υποστηριζόμενες λειτουργίες. Μια τέτοια εφαρμογή είναι το *OEM History*, που παρουσιάστηκε στο Κεφάλαιο 6, και που βασίζεται στο παρόν σύστημα για την εκτέλεση λειτουργιών χαμηλού επιπέδου πάνω στα MSSD. Το *MSSDesigner* είναι ένα γραφικό περιβάλλον για τον χειρισμό και την επερώτηση Πολυδιάστατων Γράφων Επερωτήσεων, και προσφέρει έναν διαδραστικό τρόπο χρήσης του συστήματος.

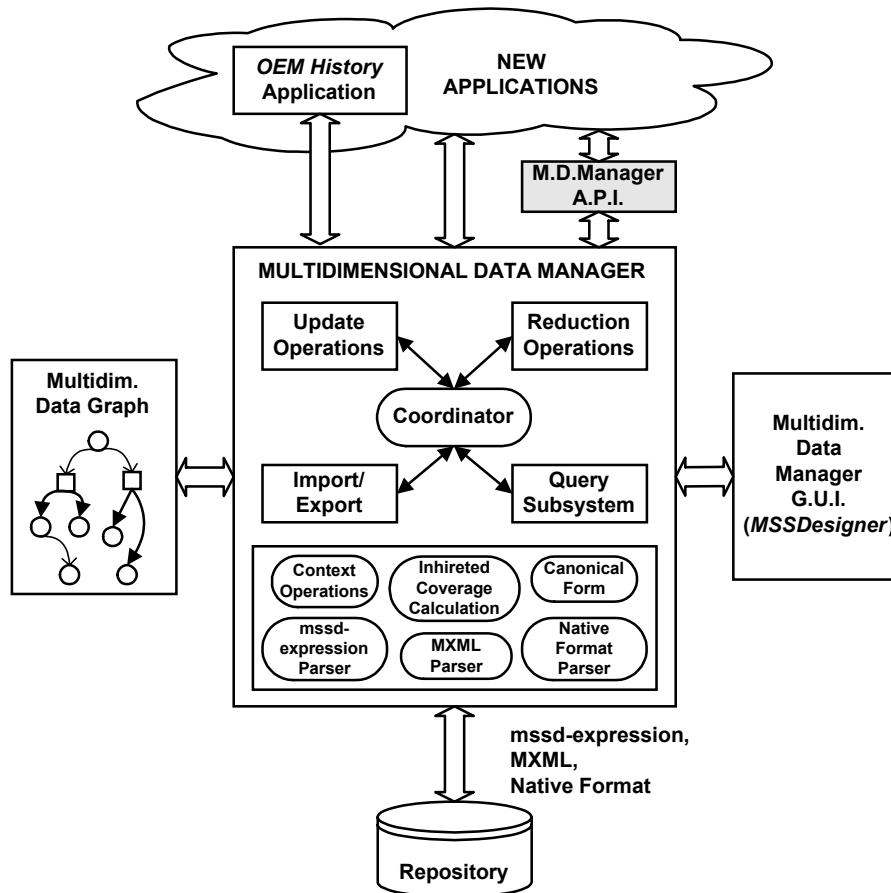
Στα ακόλουθα, δίνουμε μια συνολική περιγραφή του συστήματος, και συζητούμε τα βασικά του τμήματα ένα προς ένα.

### 8.1 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ

Το υλοποιημένο σύστημα αποτελεί μια πρότυπη υποδομή για MSSD που συμπεριλαμβάνει ένα σύνολο εργαλείων και διεργασιών για την δημιουργία, διαχείριση, και επερώτηση Πολυδιάστατων Γράφων Δεδομένων. Το σύστημα μπορεί να χρησιμοποιηθεί κατευθείαν μέσω του γραφικού περιβάλλοντος, ή μπορεί να παράσχει υποστήριξη σε εφαρμογές που χρειάζονται μια πλατφόρμα MSSD.

Η συνολική αρχιτεκτονική του συστήματος φαίνεται στην Εικόνα 8.1. Ακολούθως συζητούμε εν συντομία τα διάφορα τμήματά της.

Εικόνα 8.1: Συνολική αρχιτεκτονική του συστήματος.



*Πολυδιάστατος Γράφος Δεδομένων (Multidimensional Data Graph):* Αυτό το τμήμα αποτελείται από τις δομές δεδομένων της κύριας μνήμης που αποθηκεύουν αναπαραστάσεις MSSD γράφων.

*Αποθηκευτικός χώρος (Repository):* Είναι το φυσικό μέσο αποθήκευσης για το φόρτωμα και το σώσιμο MSSD από και προς φακέλους. Υποστηρίζεται ένας αριθμός μορφώσεων, συγκεκριμένα εκφράσεις mssd-expressions, MXML, και Native Format.

*Διαχειριστής πολυδιάστατων δεδομένων (Multidimensional Data Manager), ή MDM για συντομία:* Το MDM είναι υπεύθυνο για την διαχείριση Πολυδιάστατων Γράφων Δεδομένων και γράφων MOEM. Αποτελείται από ένα σύνολο μονάδων που υποστηρίζουν την δημιουργία, συντήρηση, και επερώτηση πολυδιάστατων ημιδομημένων δεδομένων. Το MDM μετατρέπει έναν γράφο από μια αναπαράσταση σε άλλους μορφώσιμους προκειμένου να σώσει MSSD σε φακέλους, και φορτώνει γράφους από φακέλους που περιέχουν MSSD σε διάφορους μορφώσιμους. Συζητούμε περαιτέρω για το MDM στην Ενότητα 8.2.

*MSSDesigner:* Το MSSDesigner είναι ένα γραφικό περιβάλλον για το MDM. Προσφέρει εύκολη, διαδραστική πρόσβαση σε διάφορες λειτουργίες του MDM για την δημιουργία γράφων, την συντήρηση, και την επερώτησή τους. Το MSSDesigner παρουσιάζεται λεπτομερώς στην Ενότητα 8.3.

*Προγραμματιστική διεπαφή εφαρμογών του MDM (MDM Application Programming Interface):* Το MDM API είναι μια βιβλιοθήκη εντολών που δίνει πρόσβαση στην

λειτουργικότητα του συστήματος. Νέες εφαρμογές θα είναι σε θέση να χρησιμοποιούν την υπάρχουσα πλατφόρμα για MSSD κωδικοποιώντας εντολές στην μορφή μιας script γλώσσας. Το τμήμα αυτό δεν έχει υλοποιηθεί ακόμη.

*Εφαρμογές (Applications):* Νέες εφαρμογές θα είναι δυνατόν να χτιστούν πάνω από το σύστημα, χρησιμοποιώντας τις λειτουργίες του μέσω του MDM API. Επιπρόσθετα, οι εφαρμογές μπορούν να χρησιμοποιούν το σύστημα απευθείας, όπως στην περίπτωση του *OEM History* που περιγράφεται λεπτομερώς στο Κεφάλαιο 6.

Το MDM και το MSSDesigner είναι προφανώς τα πιο σημαντικά τμήματα, και αποτελούν το αντικείμενο των ενοτήτων που ακολουθούν.

## 8.2 ΔΙΑΧΕΙΡΙΣΤΗΣ ΠΟΛΥΔΙΑΣΤΑΤΩΝ ΔΕΔΟΜΕΝΩΝ

Οι μονάδες που περιέχονται στο MDM φαίνονται στην Εικόνα 8.1. Αρχίζουμε με μια σειρά λειτουργιών γενικής χρήσης, που στην Εικόνα 8.1 εμφανίζονται στο κουτί στο κάτω μέρος του MDM, και που είναι άμεσα προσβάσιμες από κάθε άλλη μονάδα του MDM.

### 8.2.1 Λειτουργίες γενικής χρήσης

Η λειτουργία *πράξεων περιβάλλοντος (Context Operations)* υλοποιεί πράξεις πάνω σε προσδιοριστές περιβάλλοντος όπως ορίστηκαν στο Κεφάλαιο 3, συμπεριλαμβανομένων της τομής περιβάλλοντος, της ένωσης περιβάλλοντος, και της απλοποίησης προσδιοριστών περιβάλλοντος.

Η λειτουργία *υπολογισμού κληρονομούμενης κάλυψης (Inherited Coverage Calculator)* χρησιμοποιείται για τον υπολογισμό των κληρονομούμενων καλύψεων σε Πολυδιάστατους Γράφους Δεδομένων. Προς το παρόν υπολογίζεται μόνο το κληρονομούμενο περιβάλλον, οι ίδιες τεχνικές πάντως μπορούν να εφαρμοστούν για τον υπολογισμό της κάλυψης περιβάλλοντος και της κληρονομούμενης κάλυψης, όπως εξηγήθηκε στο Κεφάλαιο 4. Για να υπολογιστεί το κληρονομούμενο περιβάλλον, ο γράφος διασχίζεται κατά πλάτος, αρχίζοντας από την ρίζα. Υλοποιούμε έναν απλό αλγόριθμο σταθερού σημείου για την αντιμετώπιση της ύπαρξης κύκλων στον γράφο.

Η λειτουργία *κανονικής μορφής (Canonical Form)* μετατρέπει έναν γράφο στην κανονική του μορφή.

Οι υπόλοιπες λειτουργίες εκτελούν συντακτική ανάλυση εκφράσεων σε διάφορους μορφότυπους, και χρησιμοποιούνται για να σώσουν κάποιον γράφο σε έναν φάκελο, ή για να φορτώσουν κάποιον γράφο από έναν φάκελο. Οι λειτουργίες αυτές είναι η *ανάλυση mssd-expression (mssd-expression Parser)*, η *ανάλυση MXML (MXML Parser)*, και η *ανάλυση Native Format (Native Format Parser)*, οι οποίες εκτελούν συντακτική ανάλυση εκφράσεων mssd-expressions, MXML, και εκφράσεων “Native Format”, αντιστοίχως.

### 8.2.2 Μονάδα συντονισμού

Εξωτερικά τμήματα επικοινωνούν με το MDM μέσω της μονάδας *συντονισμού (Coordinator)*. Η δουλειά της είναι να αναλύει τα εισερχόμενα αιτήματα σε μια σειρά βασικών λειτουργιών, και να τις αναθέτει στις κατάλληλες μονάδες του MDM.

Για παράδειγμα, θεωρήστε την περίπτωση που ένας χρήστης διαγράφει μια ακμή από έναν γράφο μέσω του γραφικού περιβάλλοντος MDM GUI. Η μονάδα συντονισμού δέχεται

το ανάλογο αίτημα από το MDM GUI, και το προωθεί στην μονάδα *λειτουργιών ενημέρωσης*, η οποία είναι υπεύθυνη για την διεκπεραίωση τέτοιων αλλαγών. Στην συνέχεια, η μονάδα συντονισμού ειδοποιεί το MDM GUI ώστε να εμφανίσει τον ανανεωμένο γράφο.

### 8.2.3 Μονάδα λειτουργιών ενημέρωσης

Η μονάδα *λειτουργιών ενημέρωσης (Update Operations)* διεκπεραιώνει αλλαγές σε έναν Πολυδιάστατο Γράφο Δεδομένων. Υλοποιεί τις λειτουργίες βασικών αλλαγών που εισήχθησαν στο Κεφάλαιο 6, δηλαδή τις *createCNode*, *updateCNode*, *createMNode*, *addEEdge*, *remEEdge*, *addCEdge*, και *remCEdge*.

Όπως και στο συμβατικό OEM, στον Πολυδιάστατο Γράφο Δεδομένων η διαγραφή αντικειμένων επιτυγχάνεται μέσω της αφαίρεσης ακμών, επειδή η διατήρηση της ύπαρξης ενός αντικειμένου εξαρτάται από το εάν η ρίζα του γράφου έχει πρόσβαση στο αντικείμενο αυτό. Η μονάδα μπορεί να ελέγξει τον γράφο για μη προσβάσιμους κόμβους, οι οποίοι τότε διαγράφονται αυτόματα.

### 8.2.4 Μονάδα εισαγωγής / εξαγωγής

Ο Πολυδιάστατος Γράφος Δεδομένων μπορεί να κωδικοποιηθεί και να αποθηκευτεί σε κάποιον φάκελο με μια σειρά διαφορετικών μορφότυπων. Προς το παρόν, το σύστημα υποστηρίζει τρεις διαφορετικούς μορφότυπους: εκφράσεις *mssd-expressions* που ορίζονται στο Κεφάλαιο 4, πολυδιάστατη XML που ορίζεται στο Κεφάλαιο 7, και εκφράσεις *Native Format*. Η μονάδα *εισαγωγής / εξαγωγής (Import / Export)* χειρίζεται την διαδικασία αποθήκευσης και φορτώματος ενός γράφου στους μορφότυπους αυτούς.

Αντίθετα με άλλους μορφότυπους, οι εκφράσεις *Native Format* διατηρούν τους κόμβους και τις ακμές ενός γράφου στην σωστή θέση τους στην οθόνη. Κάθε κόμβος περιγράφεται με μια γραμμή της μορφής

```
[nodeId, nodeType, xPos, yPos, value, isRoot, bHung]
```

όπου *nodeId* είναι το αναγνωριστικό του κόμβου, *nodeType* είναι ο τύπος του κόμβου (ατομικός, σύνθετος, ή πολυδιάστατος), *xPos* και *yPos* είναι οι συντεταγμένες του κόμβου στην επιφάνεια της οθόνης, και *value* είναι η τιμή του κόμβου. Επιπρόσθετα, το *isRoot* σημειώνεται ως αληθές εάν ο συγκεκριμένος κόμβος είναι η ρίζα του γράφου, και το *bHung* είναι αληθές εάν ο κόμβος δεν είναι προσβάσιμος από την ρίζα.

Κάθε ακμή περιγράφεται με μια γραμμή της μορφής

```
(fromNode; toNode; edgeType; value)
```

όπου *fromNode* είναι ο κόμβος αναχώρησης, *toNode* είναι ο κόμβος προορισμού, το *edgeType* καθορίζει εάν η ακμή είναι ακμή περιβάλλοντος ή ακμή οντοτητας, και το *value* είναι η ετικέτα της ακμής: προσδιοριστής περιβάλλοντος (ρητό περιβάλλον) αν η ακμή είναι ακμή περιβάλλοντος, και συμβατική ετικέτα στοιχειοσειράς αν η ακμή είναι ακμή οντότητας.

### 8.2.5 Μονάδα λειτουργιών αναγωγής

Η μονάδα αυτή υλοποιεί την αναγωγή σε OEM κάτω από κάποιον κόσμο, και την μερική αναγωγή για δεδομένο ερμηνευτικό περιβάλλον. Και οι δύο διαδικασίες ορίζονται στο Κεφάλαιο 4. Αυτή την στιγμή το σύστημα χρησιμοποιεί το κληρονομούμενο περιβάλλον για την υλοποίηση της αναγωγής, αντί για την κληρονομούμενη κάλυψη όπως περιγράφεται στο

Κεφάλαιο 4. Ο λόγος είναι ότι, όπως αναφέρεται στην Ενότητα 8.2.1, το σύστημα προς το παρόν δεν υπολογίζει την κάλυψη περιβάλλοντος. Σαν συνεπακόλουθο, σε ορισμένες περιπτώσεις είναι δυνατόν η αναγωγή σε OEM να επιστρέψει γράφο με σύνθετους κόμβους για φύλλα.

## 8.2.6 Υποσύστημα επερωτήσεων

Το υποσύστημα επερωτήσεων (*Query Subsystem*) υλοποιεί την MQL πάνω από το LORE [MAG+97], όπως περιγράφεται λεπτομερώς στο Κεφάλαιο 5. Είναι υπεύθυνο για την εκτέλεση MQL επερωτήσεων πάνω σε κάποιον Πολυδιάστατο Γράφο Δεδομένων, και για την παραγωγή νέων Πολυδιάστατων Γράφων Δεδομένων σαν αποτέλεσμα των επερωτήσεων αυτών. Όπως σημειώνουμε στην Ενότητα 8.2.1, αυτή την στιγμή το σύστημά μας δεν υπολογίζει την κάλυψη περιβάλλοντος. Γι' αυτό, όταν αποτιμούμε επερωτήσεις MQL χρησιμοποιούμε το κληρονομούμενο περιβάλλον του γράφου αντί για την κληρονομούμενη κάλυψη.

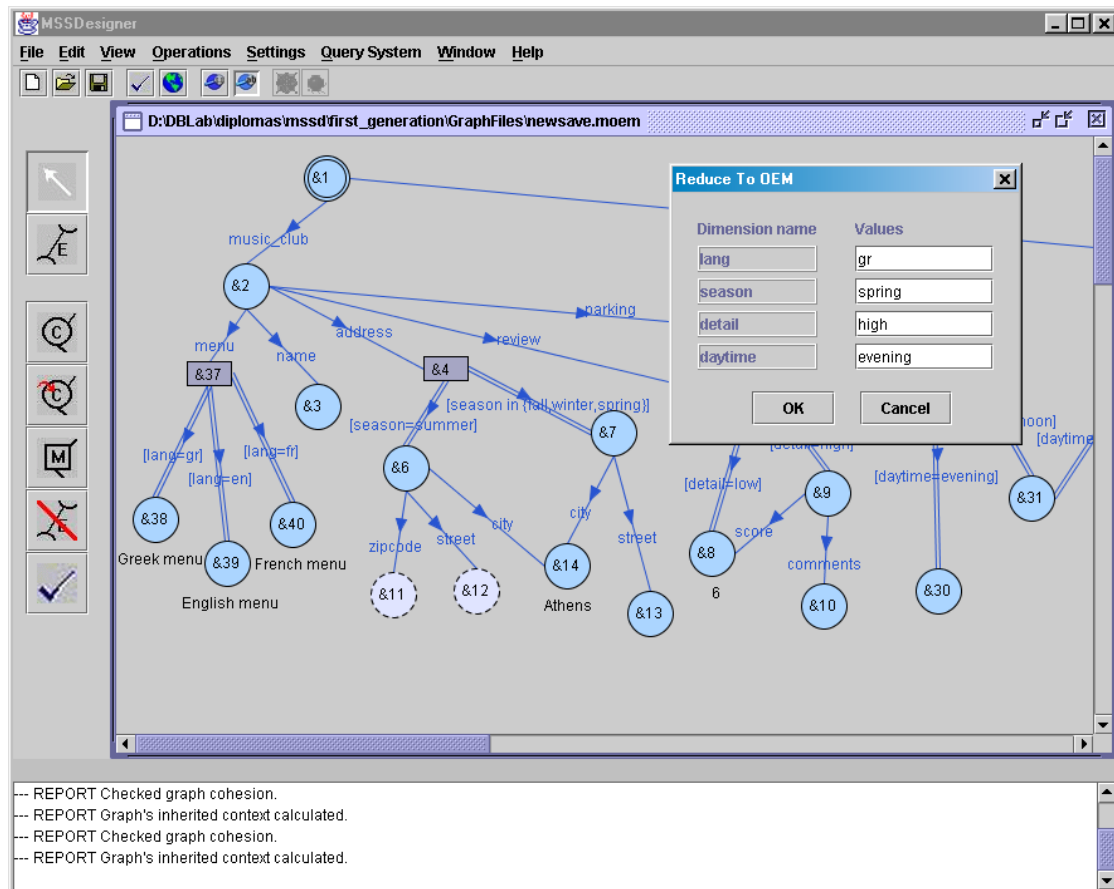
Το σύστημά μας έχει υλοποιηθεί σε Java, και τρέχει σε κάθε λειτουργικό σύστημα που υποστηρίζεται από την Java. Από την άλλη, το LORE τρέχει σε LINUX [LIN], γεγονός που βάζει έναν περιορισμό στο σύστημά μας από την στιγμή που θέλουμε να λειτουργεί το υποσύστημα επερωτήσεων. Σαν λύση, υλοποιήσαμε σε Java ένα πρόγραμμα εξυπηρετητή που δρα σαν ενδιάμεσος ανάμεσα στο υποσύστημα επερωτήσεων και στο LORE, και ανταλλάσσει δεδομένα και με τα δύο συστήματα χρησιμοποιώντας ένα ειδικό πρωτόκολλο. Αυτό το πρόγραμμα εξυπηρετητή λειτουργεί στο ίδιο μηχάνημα LINUX όπως το LORE, και επιτρέπει στο σύστημά μας να χρησιμοποιεί το LORE από κάποιο άλλο μηχάνημα.

Η ακόλουθη ενότητα περιγράφει το MSSDesigner, και δίνει επίσης μια συγκεκριμένη εικόνα του πως μπορεί να χρησιμοποιηθεί το υποσύστημα επερωτήσεων.

## 8.3 MSSDESIGNER

Το *MSSDesigner* είναι ένα γραφικό περιβάλλον που δίνει πρόσβαση στις λειτουργίες του διαχειριστή πολυδιάστατων δεδομένων (Multidimensional Data Manager). Στην Εικόνα 8.2 φαίνεται ένα ενδεικτικό στιγμιότυπο οθόνης του MSSDesigner, που εμφανίζει ένα γράφο για το παράδειγμα του πολυδιάστατου οδηγού διασκέδασης που παρουσιάστηκε σε προηγούμενα κεφάλαια. Στο δεξί μέρος ένα κουτί διαλόγου ζητάει από τον χρήστη να ορίσει έναν κόσμο, ώστε να ανάγει τον γράφο σε OEM.

Εικόνα 8.2: Δείγμα οθόνης του MSSDesigner.



Το MSSDesigner χρησιμοποιεί πολλαπλά παράθυρα (MDI), που μπορούν να είναι ανοιχτά ταυτόχρονα, και καθένα να εμφανίζει διαφορετικό γράφο. Όλες οι λειτουργίες εκτελούνται από τις επιλογές του μενού και τα κουμπιά της εργαλειοθήκης, και επιδρούν στον γράφο του οποίου το παράθυρο είναι την στιγμή εκείνη εστιασμένο. Τα κουμπιά στο MSSDesigner είναι τοποθετημένα σε δύο εργαλειοθήκες, μία στο πάνω και μία στο αριστερό μέρος του κυρίως παραθύρου. Η εργαλειοθήκη του αριστερού μέρους περιέχει κουμπιά που αντιστοιχούν στις λειτουργίες βασικών αλλαγών οι οποίες μεταβάλλουν τον γράφο. Αρχίζοντας από την κορυφή της εργαλειοθήκης, το κουμπί με το βέλος επιτρέπει την επιλογή πολλών κόμβων μαζί και την μετάθεσή τους στην επιφάνεια της οθόνης. Τα πέντε επόμενα κουμπιά αντιστοιχούν σε λειτουργίες για την προσθήκη και ενημέρωση κόμβων, και για την προσθήκη και διαγραφή ακμών. Το τελευταίο κουμπί εκτελεί έλεγχο συνέπειας του γράφου, και αφαιρεί κόμβους που δεν είναι προσβάσιμοι από την ρίζα.

Η εργαλειοθήκη στο πάνω μέρος περιέχει εννέα κουμπιά. Το πρώτο ανοίγει ένα άδειο παράθυρο για τον σχεδιασμό νέου γράφου. Το δεύτερο κουμπί δημιουργεί έναν γράφο διαβάζοντας εκφράσεις Native Format, ενώ το επόμενο κουμπί εξάγει έναν υπάρχοντα γράφο δημιουργώντας εκφράσεις Native Format. Το κουμπί με το σύμβολο “tick” ελέγχει την εγκυρότητα του γράφου<sup>117</sup>, ενώ το επόμενο κουμπί χρησιμεύει για την αναγωγή σε OEM

<sup>117</sup> Η έννοια της **εγκυρότητας (validity)** ορίζεται στο [SG02], σε κάθε περίπτωση όμως η *κάλυψη περιβάλλοντος* είναι μια έννοια πιο ισχυρή, και η εισαγωγή της έθεσε την ιδέα της εγκυρότητας στο περιθώριο.

ενός γράφου κάτω από δεδομένο κόσμο. Η μερική αναγωγή μπορεί να εκτελεστεί μέσα από το μενού της εφαρμογής. Με το επόμενο κουμπί είναι δυνατόν να εμφανίσει κανείς ή να αποκρύψει τα ρητά περιβάλλοντα και τις ετικέτες των ακμών. Το κουμπί που ακολουθεί καλεί την λειτουργία υπολογισμού κληρονομούμενης κάλυψης, και προκαλεί τα κληρονομούμενα περιβάλλοντα<sup>118</sup> να εμφανιστούν στην οθόνη. Τα δύο τελευταία κουμπιά ανοίγουν το παράθυρο επερωτήσεων *MQL Query*, και ένα παράθυρο που δείχνει τους ακέραιους που χρησιμοποιούνται για την κωδικοποίηση κόσμων (όπως εξηγείται στο Κεφάλαιο 5). Όλες οι λειτουργίες των κουμπιών μπορούν επίσης να εκτελεστούν μέσω του μενού του MSSDesigner.

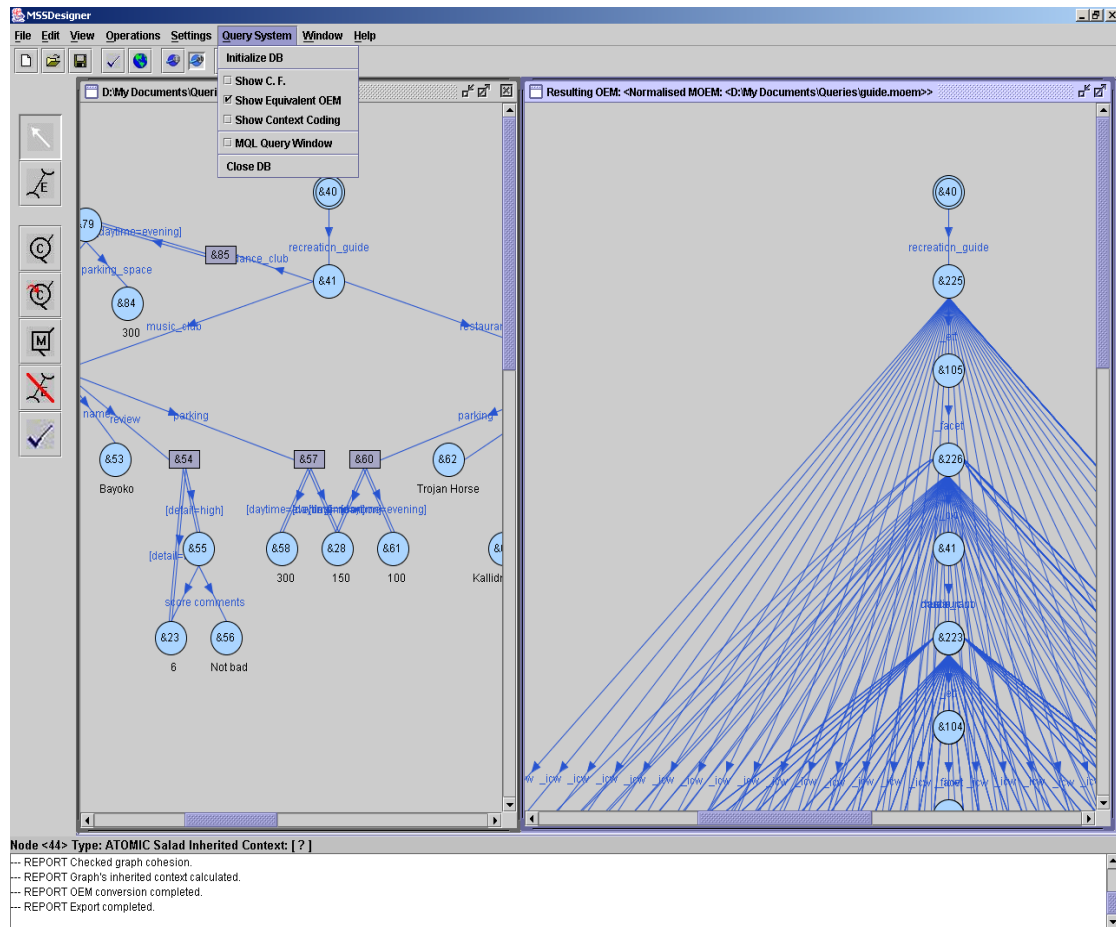
Με το MSSDesigner μπορούμε να δημιουργήσουμε γράφους από εκφράσεις *mssd-expression* ή από έγγραφα *MXML*, και να εξάγουμε υπάρχοντες γράφους σε αυτούς τους μορφότυπους. Οι επιλογές *File Import / Export* του μενού αντιστοιχούν στις λειτουργίες αυτές. Οι εκφράσεις *Native Format* αποθηκεύουν επίσης την θέση των κόμβων στην οθόνη, και είναι χρήσιμες για το σώσιμο και το φόρτωμα ημιτελών γράφων, επειδή η διεργασία του σώσιματος και φορτώματος σε *Native Format* δεν εκτελεί έλεγχο συνέπειας, και έτσι οι μη προσβάσιμοι κόμβοι / υπογράφοι δεν διαγράφονται. Από την άλλη, η συνέπεια του γράφου ελέγχεται πάντα όταν εισάγουμε ή εξάγουμε εκφράσεις *mssd-expressions* ή έγγραφα *MXML*.

Η Εικόνα 8.3 δείχνει ένα στιγμιότυπο οθόνης του MSSDesigner μετά την αρχικοποίηση του υποσυστήματος επερωτήσεων. Για την αρχικοποίηση του υποσυστήματος επερωτήσεων εκτελούνται τα παρακάτω βήματα: (1) ο γράφος *G* στο εστιασμένο παράθυρο μετατρέπεται σε έναν νέο γράφο *G<sub>CF</sub>* που βρίσκεται σε κανονική μορφή, (2) οι πιθανοί κόσμοι για τον *G* κωδικοποιούνται σαν ακέραιοι αριθμοί, (3) ο γράφος *G<sub>CF</sub>* μετατρέπεται σε έναν ισοδύναμο *OEM* γράφο *O* μέσω της διαδικασίας **MDGToOEM** που ορίστηκε στο Κεφάλαιο 5, και (4) το αποτέλεσμα της διεργασίας, ο *OEM* γράφος *O*, παραδίδεται στο *LORE* και γίνεται η τρέχουσα βάση δεδομένων.

---

<sup>118</sup> Προς το παρόν, υπολογίζονται και εμφανίζονται τα κληρονομούμενα περιβάλλοντα αντί για τις κληρονομούμενες καλύψεις.

**Εικόνα 8.3:** Αρχικοποίηση του Υποσυστήματος Επερωτήσεων με μια βάση δεδομένων γράφου.



Όπως φαίνεται στην Εικόνα 8.3, με την επιλογή “Initialize DB” από το μενού, ο χρήστης μπορεί να ανοίξει ένα νέο παράθυρο με την βάση δεδομένων γράφου σε κανονική μορφή, ή με τον ισοδύναμο γράφο OEM που παραδόθηκε στο LORE. Το δεξί παράθυρο στην Εικόνα 8.3 εμφανίζει τον ισοδύναμο γράφο OEM για την βάση δεδομένων γράφου που φαίνεται στο αριστερό παράθυρο. Επιπρόσθετα, ο χρήστης μπορεί να ανοίξει ένα κουτί διαλόγου που περιέχει τον κατάλογο των ακέραιων μαζί με τους κόσμους που κωδικοποιούν.

Από την στιγμή που το υποσύστημα επερωτήσεων έχει αρχικοποιηθεί, είναι δυνατόν να εμφανιστεί το παράθυρο επερώτησης “MQL Query”. Στην Εικόνα 8.4, το παράθυρο MQL Query προσδιορίζεται από ένα παχύ μαύρο πλαίσιο. Η επερώτηση MQL εισάγεται στο πάνω μέρος του παραθύρου, είτε πληκτρολογώντας είτε φορτώνοντας το κείμενο από κάποιον φάκελο. Τότε πατώντας το κουμπί “Lorel equivalent”, η ισοδύναμη<sup>119</sup> επερώτηση Lorel εμφανίζεται στο κάτω μέρος του παραθύρου MQL Query. Καθώς η ισοδύναμη επερώτηση Lorel μπορεί να καταλαμβάνει αρκετές γραμμές, είναι δυνατόν να την εμφανίσουμε σε ξεχωριστό παράθυρο με επικεφαλίδα “Equivalent Lorel Query”, το οποίο φαίνεται στο κάτω αριστερό μέρος στην Εικόνα 8.4.

<sup>119</sup> Όπως ορίζεται στο Κεφάλαιο 5.



Εικόνα 8.4: Αποτίμηση μιας επερώτησης MQL και εμφάνιση των αποτελεσμάτων.

The screenshot displays the MSSDesigner application with several windows open:

- MQL Query:** Contains an MQL query for selecting restaurant information based on terrace and season conditions.
- Equivalent Lorel Query:** Shows the query translated into a Lorel format with variables and set operations.
- Context Coding:** Lists various context codes such as [daytime=evening, detail=high, season=spring, lang=en].
- Graph Visualization:** A complex graph with nodes (e.g., &41, &46, &52, &53, &56, &59) and edges representing relationships between different data elements like 'address', 'name', 'review', and 'booking'.
- REPORT:** A status bar at the bottom indicating the completion of the report generation process.

Το κουμπί “Submit” στο παράθυρο MQL Query υποβάλλει την ισοδύναμη επερώτηση Lorel στο LORE, και λαμβάνει το αποτέλεσμα στην μορφή ενός εγγράφου XML. Πατώντας το κουμπί “Show Result” που δείχνει το αποτέλεσμα, εκτελούνται τα ακόλουθα βήματα: (1) το έγγραφο XML με το αποτέλεσμα μετατρέπεται σε έναν γράφο OEM, (2) ο γράφος OEM μετατρέπεται σε έναν Πολυδιάστατο Γράφο Δεδομένων μέσω της διαδικασίας **OEMToMDG** που ορίστηκε στο Κεφάλαιο 5, και (3) ο Πολυδιάστατος Γράφος Δεδομένων εμφανίζεται σε ένα νέο παράθυρο, σαν το τελικό αποτέλεσμα της επερώτησης MQL.

Στην Εικόνα 8.4, το μικρό παράθυρο στο κάτω μέρος με τον γράφο δείχνει το αποτέλεσμα της επερώτησης MQL<sup>120</sup> που εμφανίζεται στο πάνω αριστερό μέρος. Παρόλο που ο γράφος έχει την μορφή συμβατικού OEM, στην πραγματικότητα είναι ένας Πολυδιάστατος Γράφος Δεδομένων που τυχαίνει να περιέχει μόνο κόμβους περιβάλλοντος και ακμές οντότητας. Οι ακέραιοι που χρησιμοποιήθηκαν για την κωδικοποίηση κόσμων μετατρέπονται πάλι στα αποτελέσματα σε προσδιοριστές περιβάλλοντος. Το μικρό παράθυρο πάνω δεξιά στην Εικόνα 8.4 εμφανίζει τον κατάλογο των ακεραίων μαζί με τους κόσμους που κωδικοποιούν.

<sup>120</sup> Προσεκτική εξέταση αποκαλύπτει κάποιες μικρές διαφορές ανάμεσα στην συγκεκριμένη επερώτηση MQL και στο συντακτικό της MQL όπως αυτό ορίστηκε στο Κεφάλαιο 5. Ο λόγος είναι ότι το υποσύστημα επερωτήσεων υλοποιεί την πρώτη έκδοση της MQL, που έχει μερικές διαφορές στο συντακτικό από την τρέχουσα έκδοση, η οποία ορίζεται στο Κεφάλαιο 5.

Επερωτήσεις MQL μπορούν να υποβληθούν στην ίδια βάση δεδομένων MOEM η μία μετά την άλλη· όμως, για να αλλάξει η βάση δεδομένων σε κάποιον διαφορετικό γράφο, το υποσύστημα επερωτήσεων πρέπει να αρχικοποιηθεί ξανά μέσω την επιλογής του μενού “Initialize DB”. Η επιλογή του μενού “Close DB” που κλείνει την βάση δεδομένων μπορεί να χρησιμοποιηθεί για να τερματίσει την σύνδεση με το LORE, και για να εκτελέσει λειτουργίες «καθαριότητας» στον διαχειριστή πολυδιάστατων δεδομένων και στο LORE.

## 8.4 ΠΕΡΙΛΗΨΗ

Στο κεφάλαιο αυτό παρουσιάσαμε ένα σύστημα για τη διαχείριση MSSD που εφαρμόζει τις περισσότερες από τις βασικές έννοιες που εισήχθησαν στα προηγούμενα κεφάλαια, και το οποίο μπορεί να χρησιμοποιηθεί σαν υποδομή για την ανάπτυξη νέων εργαλείων και εφαρμογών MSSD. Συζητήσαμε την αρχιτεκτονική του συστήματος, τις μονάδες του, και την λειτουργία του μέσα από το *MSSDesigner*, ένα γραφικό περιβάλλον που υποστηρίζει τη διαδραστική δημιουργία, τον πειραματισμό, και την επερώτηση Πολυδιάστατων Γράφων Δεδομένων με την MQL.

## 9 ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ

Ενώ στις παραδοσιακές βάσεις δεδομένων και στα αντίστοιχα πληροφοριακά συστήματα το πλήθος των χρηστών είναι λίγο πολύ γνωστό και το υπόβαθρό τους είναι σε μεγάλο βαθμό ομογενές, οι χρήστες του Παγκόσμιου Ιστού δεν έχουν κοινό υπόβαθρο ούτε όμοιες θεωρήσεις όταν επεξεργάζονται και ερμηνεύουν δεδομένα. Αυτοί οι χρήστες μπορεί να έχουν διαφορετικές οπτικές των ίδιων πληροφοριακών οντοτήτων, κάτι που θα πρέπει να ληφθεί υπόψη από τα μοντέλα δεδομένων και τις γλώσσες επερωτήσεων για τον Παγκόσμιο Ιστό. Επιπλέον, οι παροχείς πληροφορίας πρέπει συχνά να διαχειριστούν διαφορετικές παραλλαγές των ίδιων ουσιαστικά δεδομένων, που απευθύνονται σε διαφορετικές ομάδες καταναλωτών. Σε αυτήν την διατριβή προτείνουμε τα **πολυδιάστατα ημιδομημένα δεδομένα (MSSD)**, όπου πληροφοριακές οντότητες μπορούν να εκδηλώνουν διαφορετικές εκφάνσεις, ανάλογα με το *ερμηνευτικό περιβάλλον*. Στα παρακάτω συνοψίζουμε την προσέγγισή μας και την συμβολή αυτής της διατριβής.

Αναπτύξαμε μια τυποποίηση για να αναπαριστούμε ερμηνευτικά περιβάλλοντα με έναν ευέλικτο και απλό τρόπο, χρησιμοποιώντας μεταβλητές που ονομάζονται **διαστάσεις**. Οι **προσδιοριστές περιβάλλοντος** ορίζουν ερμηνευτικά περιβάλλοντα περιορίζοντας τις δυνατές τιμές των διαστάσεων, και συνδυάζοντας τέτοιους περιορισμούς σε διαζεύξεις και συζεύξεις. Δώσαμε νόημα στο ερμηνευτικό περιβάλλον δείχνοντας ότι μπορεί να ιδωθεί σαν ένα σύνολο πιθανών **κόσμων**, όπου ο κόσμος είναι μια εκδοχή της πραγματικότητας κάτω από την οποία οι πληροφορίες αποκτούν μονοσήμαντη ερμηνεία. Ορίσαμε πράξεις περιβάλλοντος που συνδυάζουν και συγκρίνουν ερμηνευτικά περιβάλλοντα, και διερευνήσαμε τις ιδιότητές τους και την πολυπλοκότητά τους ως προς τον χρόνο. Ένα σημαντικό πλεονέκτημα της προσέγγισής μας είναι ότι οι περισσότερες πράξεις περιβάλλοντος μπορούν να εκτελεστούν ακόμη και αν δεν έχουμε πλήρη γνώση των διαστάσεων που εμπλέκονται: στην περίπτωση η αντιστοιχία ερμηνευτικών περιβαλλόντων σε σύνολα κόσμων είναι δυνατόν να αναβληθεί μέχρι να έχουμε πλήρη γνώση των διαστάσεων.

Προτείναμε τον **Πολυδιάστατο Γράφο Δεδομένων**, ένα μοντέλο δεδομένων γράφου για MSSD που ενσωματώνει προσδιοριστές περιβάλλοντος. Στον Πολυδιάστατο Γράφο Δεδομένων, οι ακμές με ετικέτα ορίζουν συσχετίσεις αναμεταξύ πολυδιάστατων οντοτήτων, οι οποίες είναι πληροφοριακές οντότητες που δύνανται να εκδηλώνουν διαφορετικές εκφάνσεις κάτω από διαφορετικούς κόσμους. Οι εκφάνσεις αυτές συνοδεύονται από προσδιοριστές περιβάλλοντος, που ορίζουν τους κόσμους κάτω από τους οποίους υφίσταται κάθε έκφανση. Για να πάρουμε όμως τους πραγματικούς κόσμους κάτω από τους οποίους υφίστανται τα στοιχεία του γράφου, πρέπει να λάβουμε υπόψη τους συσσωρευμένους περιορισμούς καθώς αυτοί μεταδίδονται από την ρίζα στα φύλλα και από τα φύλλα στην ρίζα. Η μετάδοση αυτή διασφαλίζει ότι ένας κόμβος-παιδί υφίσταται μόνον κάτω από τους κόσμους που υφίσταται κάποιος πατέρας του, και ότι ένας κόμβος υφίσταται μόνον κάτω από τους κόσμους όπου έχει πρόσβαση σε κάποιον ατομικό κόμβο. Με βάση αυτές τις αρχές, ορίσαμε πως υπολογίζεται στα πλαίσια ενός γράφου το *πραγματικό* ερμηνευτικό περιβάλλον ενός κόμβου ή μιας ακμής. Αυτό το ερμηνευτικό περιβάλλον καλείται **κληρονομούμενη**

**κάλυψη**, ενώ η **κληρονομούμενη κάλυψη μονοπατιού** αντιπροσωπεύει τους κόσμους κάτω από τους οποίους υφίσταται ένα ολόκληρο μονοπάτι.

Δείξαμε ότι από έναν Πολυδιάστατο Γράφο Δεδομένων μπορούν να εξαχθούν συμβατικά OEM που υφίστανται κάτω από κάποιον κόσμο, και δώσαμε τις συνθήκες κάτω από τις οποίες μπορούν να λάβουν χώρα τέτοιου είδους εξαγωγές. Σαν κομμάτι αυτών των συνθηκών συζητήσαμε για τους **αιτιοκρατικούς ως προς το περιβάλλον** γράφους, και παρουσιάσαμε μια διαδικασία που **ανάγει σε OEM** έναν αιτιοκρατικό ως προς το περιβάλλον Πολυδιάστατο Γράφο Δεδομένων κάτω από έναν κόσμο. Επιπρόσθετα, περιγράψαμε την **μερική αναγωγή** ενός Πολυδιάστατου Γράφου Δεδομένων για κάποιο ερμηνευτικό περιβάλλον. Επίσης ορίσαμε μια διαδικασία που μετατρέπει έναν Πολυδιάστατο Γράφο Δεδομένων σε **κανονική μορφή**. Περιττές αλληλεξαρτήσεις ανάμεσα από πολυδιάστατες οντότητες μπορεί να προκαλέσουν ανωμαλίες κατά την συντήρηση ενός Πολυδιάστατου Γράφου Δεδομένων, και η κανονική μορφή αποφεύγει τις ανωμαλίες αυτές μετατρέποντας τον αρχικό γράφο σε έναν καλύτερα δομημένο γράφο. Μια σημαντική ιδιότητα ενός γράφου που είναι σε κανονική μορφή είναι ότι οποιοδήποτε μονοπάτι έχει προβλέψιμη δομή, πράγμα πολύ χρήσιμο για την διατύπωση επερωτήσεων. Έχοντας διερευνήσει λεπτομερώς τις ιδιότητες του Πολυδιάστατου Γράφου Δεδομένων, ορίσαμε το **Πολυδιάστατο OEM (MOEM)** σαν μια ειδική περίπτωση Πολυδιάστατου Γράφου Δεδομένων που προέρχεται αποκλειστικά από την συνένωση συμβατικών OEM τα οποία υφίστανται κάτω από διάφορους κόσμους. Ορίσαμε επίσης τις εκφράσεις **mssd-expression**, που παρέχουν ένα συντακτικό για την έκφραση Πολυδιάστατων Γράφων Δεδομένων και MOEM σε μορφή κειμένου.

Παρουσιάσαμε την **Πολυδιάστατη Γλώσσα Επερωτήσεων (MQL)**, μια γλώσσα επερωτήσεων για πολυδιάστατα ημιδομημένα δεδομένα. Η MQL υποστηρίζει σαν πρώτη κατηγορία πολίτες τα ερμηνευτικά περιβάλλοντα και τις οντότητες πολλαπλών εκφάνσεων, και μπορεί να εκφράσει *επερωτήσεις οδηγούμενες από περιβάλλον*. Η MQL βασίζεται ευρέως στην Lorel και διατηρεί τις βασικές της συνεισφορές, δηλαδή τις εκφράσεις μονοπατιού και τον εξαναγκασμό. Ορίσαμε τις **εκφράσεις μονοπατιού περιβάλλοντος**, που επεκτείνουν τις εκφράσεις μονοπατιού ενσωματώνοντας **περιοριστές περιβάλλοντος**. Οι περιοριστές περιβάλλοντος μπορούν να είναι προσδιοριστές περιβάλλοντος, **μεταβλητές περιβάλλοντος**, και **πρότυπα περιβάλλοντος**, και χρησιμοποιούνται για να θέσουν συνθήκες περιβάλλοντος πάνω στον γράφο της βάσης. Δείξαμε ότι μια κατηγορία εκφράσεων μονοπατιού περιβάλλοντος μπορεί να ιδωθεί σαν ορίζουσα συνδέσεις (joins) ανάμεσα από συμβατικά OEM που υφίστανται κάτω από διαφορετικούς κόσμους. Η MQL χρησιμοποιεί δύο ειδικούς όρους για τον χειρισμό του ερμηνευτικού περιβάλλοντος: τον όρο *within* που εκφράζει συνθήκες πάνω στο ερμηνευτικό περιβάλλον, και τον όρο *context* που δημιουργεί νέες μεταβλητές περιβάλλοντος για να χρησιμοποιηθούν στην κατασκευή αποτελεσμάτων. Επιπλέον, η δεσμευμένη λέξη *holding* μπορεί να χρησιμοποιηθεί για να εφαρμόσουμε μερική αναγωγή στον γράφο των αποτελεσμάτων. Η πρότυπη υλοποίησή της MQL μας επέτρεψε να συγκρίνουμε επερωτήσεις MQL με ισοδύναμες επερωτήσεις Lorel: σε επερωτήσεις οδηγούμενες από περιβάλλον, η MQL και το MOEM είναι φανερά πολύ πιο εκφραστικά και κομψά απ' ότι η Lorel και το OEM, ενώ σε συμβατικές επερωτήσεις η MQL και οι εκφράσεις μονοπατιού περιβάλλοντος γίνονται τόσο απλές στον σχηματισμό τους όσο η Lorel και οι συμβατικές εκφράσεις μονοπατιού. Κατά την γνώμη μας, οι δυνατότητες των MSSD και η εκφραστικότητα της MQL δικαιώνουν τις επεκτάσεις στο OEM που εισήγαμε στον Πολυδιάστατο Γράφο Δεδομένων. Τα επιπλέον αυτά στοιχεία του Πολυδιάστατου Γράφου Δεδομένων χρησιμοποιούνται από την MQL για την διατύπωση *δια-κοσμικών επερωτήσεων*, που δεν έχουν αντίστοιχες σε βάσεις δεδομένων χωρίς επίγνωση του ερμηνευτικού περιβάλλοντος.

Για να αξιολογήσουμε την προσέγγισή μας, εφαρμόσαμε το πλαίσιο για MSSD που αναπτύξαμε σε ένα κλασσικό πρόβλημα των βάσεων δεδομένων: χρησιμοποιήσαμε το MOEM και την MQL για να αναπαραστήσουμε και να επερωτήσουμε ιστορικά βάσεων δεδομένων OEM. Εξηγήσαμε σε βάθος πως το MOEM μπορεί να χρησιμοποιηθεί για να αναπαραστήσει το ιστορικό μιας βάσης δεδομένων OEM. Εισήγαμε τις **λειτουργίες βασικών αλλαγών** του MOEM, και προδιαγράψαμε με ψευδοκώδικα την διαδικασία που κωδικοποιεί στο MOEM το ιστορικό μιας βάσης δεδομένων OEM. Συζητήσαμε πως οι ιδιότητες του Πολυδιάστατου Γράφου Δεδομένων εφαρμόζονται στην συγκεκριμένη περίπτωση, και δείξαμε ότι χρονικά στιγμιότυπα OEM μπορούν να ανακτηθούν από το MOEM. Παρουσιάσαμε το υλοποιημένο σύστημα *OEM History*, και επιδείξαμε μέσα από ένα παράδειγμα την διαδικασία χρήσης μιας υποκείμενης βάσης δεδομένων MOEM προκειμένου να μοντελοποιηθούν αλλαγές στο OEM. Επιπρόσθετα, δείξαμε ότι το MOEM είναι ικανό να μοντελοποιήσει αλλαγές που συμβαίνουν όχι μόνο σε βάσεις δεδομένων συμβατικών OEM, αλλά επίσης και σε βάσεις δεδομένων MOEM. Η προσέγγισή μας διατηρεί στιγμιότυπα των διαδοχικών καταστάσεων ενός αντικειμένου, σαν εκφάνσεις μιας πολυδιάστατης οντότητας. Αυτό επιτρέπει τον σχηματισμό επερωτήσεων που συσχετίζουν διαφορετικές καταστάσεις του ίδιου ή και διαφορετικών αντικειμένων σε διάφορες χρονικές στιγμές. Παρουσιάσαμε έναν αριθμό παραδειγμάτων επερωτήσεων MQL, και μέσα από αυτά επιβεβαιώσαμε την εκφραστικότητα της MQL και την αξία κάποιων χαρακτηριστικών της, όπως για παράδειγμα των προτύπων περιβάλλοντος.

Σαν παράδειγμα του πως οι έννοιες που προτείναμε μπορούν να επεκταθούν και σε άλλες κατευθύνσεις, συζητήσαμε την **Πολυδιάστατη XML (MXML)**, μια επέκταση της XML κατάλληλη για την αναπαράσταση δεδομένων που παρουσιάζουν διαφορετική τιμή ή δομή κάτω από διαφορετικά ερμηνευτικά περιβάλλοντα. Στην MXML, στοιχεία και γνωρίσματα είναι δυνατόν να εξαρτώνται από έναν αριθμό διαστάσεων, οι οποίες ορίζουν τους κόσμους κάτω από τους οποίους υφίστανται οι εκφάνσεις των στοιχείων και των γνωρισμάτων αυτών. Επιπλέον, προτείναμε μια επέκταση του DTD, που ονομάζεται **Πολυδιάστατο DTD (MDTD)**, το οποίο περιγράφει την λογική δομή των εγγράφων MXML. Συζητήσαμε τις ιδιότητες της MXML, και δώσαμε παραδείγματα του πως ένα έγγραφο MXML μπορεί να αναχθεί σε ένα συμβατικό έγγραφο XML κάτω από έναν δεδομένο κόσμο. Επιπρόσθετα, παρουσιάσαμε ένα νέο παράδειγμα για τον χειρισμό δεδομένων στον Παγκόσμιο Ιστό που εξαρτώνται από ερμηνευτικά περιβάλλοντα και παρουσιάζουν πολλαπλές εκφάνσεις. Το νέο παράδειγμα λέγεται **πολυδιάστατο παράδειγμα**, και περιλαμβάνει ζητήματα αναπαράστασης, διαχείρισης, και παρουσίασης. Συζητήσαμε την παρουσίαση εγγράφων MXML με την βοήθεια **πολυδιάστατων XSL (MXSL)** φύλλων διαμόρφωσης. Περιγράψαμε τον σχεδιασμό ενός συστήματος που υλοποιεί την βασική λειτουργικότητα του πολυδιάστατου παραδείγματος, και επιδεικνύει πως ο χρήστης μπορεί να αλληλεπιδράσει με ένα πολυδιάστατο έγγραφο και να δει διαφορετικές παραλλαγές του κάτω από διαφορετικούς κόσμους.

Τέλος, παρουσιάσαμε ένα σύστημα για τη διαχείριση MSSD που υλοποιεί τις περισσότερες βασικές έννοιες που εισήχθησαν στην διατριβή αυτή, και το οποίο μπορεί να χρησιμοποιηθεί σαν υποδομή για την ανάπτυξη νέων εργαλείων και εφαρμογών MSSD. Συζητήσαμε την αρχιτεκτονική του συστήματος, τις μονάδες του, και την λειτουργία του μέσα από το *MSSD Designer*, ένα γραφικό περιβάλλον που υποστηρίζει τη διαδραστική δημιουργία, τον πειραματισμό, και την επερωτήση Πολυδιάστατων Γράφων Δεδομένων με την MQL.

Η παρούσα εργασία ανοίγει μια σειρά από ενδιαφέρουσες κατευθύνσεις για περαιτέρω έρευνα, τις οποίες περιγράφουμε εν συντομία στην συνέχεια.

- *Αναπαράσταση ερμηνευτικού περιβάλλοντος*: Δεχθήκαμε ότι οι διαστάσεις είναι ορθογώνιες, και ότι τα πεδία ορισμού τους είναι επίπεδα σύνολα που απαριθμούν

στοιχεία. Εντούτοις, δεν συμμορφώνονται όλες οι πραγματικές εφαρμογές με αυτές τις προϋποθέσεις. Θα ήταν ενδιαφέρον να υποστηριχθούν διαστάσεις που δεν είναι ορθογώνιες, ή των οποίων τα πεδία ορισμού έχουν ιεραρχική δομή, και να ερευνηθούν οι επιπτώσεις τέτοιων διαστάσεων στην έννοια του ερμηνευτικού περιβάλλοντος και των πράξεων περιβάλλοντος.

- *Μοντέλα δεδομένων που παίρνουν υπόψη το ερμηνευτικό περιβάλλον*: Ένα ανοικτό ζήτημα είναι η ανεύρεση αποδοτικών αλγόριθμων για τον υπολογισμό των κληρονομούμενων καλύψεων ενός Πολυδιάστατου Γράφου Δεδομένων. Επιπρόσθετα, πολύ χρήσιμα για επερωτήσεις θα ήταν σχήματα για βάσεις δεδομένων MOEM τα οποία περιλαμβάνουν και διαστάσεις. Επίσης, οι μη αιτιοκρατικοί ως προς το περιβάλλον γράφοι θα πρέπει να εξεταστούν με μεγαλύτερη προσοχή. Ένα άλλο ενδιαφέρον ζήτημα είναι μέθοδοι που να χρησιμοποιούν το ερμηνευτικό περιβάλλον για να απεικονίσουν με κατάλληλο τρόπο μοντέλα δεδομένων ώστε να διευκολύνουν τις ενημερώσεις και τη συντήρηση της πληροφορίας. Ακόμη σημαντικότερο, έννοιες και ιδιότητες του Πολυδιάστατου Γράφου Δεδομένων μπορούν να μεταφερθούν σε άλλους τύπους μοντέλων δεδομένων, προκειμένου να αναπτυχθούν αντικειμενοστρεφείς ή σχεσιακές βάσεις δεδομένων που να υποστηρίζουν το ερμηνευτικό περιβάλλον σαν πολίτη πρώτης τάξης.
- *MQL*: Η αποτίμηση επερωτήσεων MQL εμπεριέχει έναν αριθμό ανοιχτών κατευθύνσεων για έρευνα, όπως η αποθήκευση Πολυδιάστατων Γράφων Δεδομένων, μέθοδοι προσπέλασης δεδομένων, και βελτιστοποίηση επερωτήσεων οδηγούμενων από περιβάλλοντα. Επιπρόσθετα, η MQL θα μπορούσε να επεκταθεί ώστε να καλύπτει και αλλαγές στους Πολυδιάστατους Γράφους Δεδομένων.
- *Αναπαράσταση ιστορικού βάσεων δεδομένων OEM*: Κατά την χρήση του MOEM για την αναπαράσταση του ιστορικού βάσεων δεδομένων υποθέσαμε ότι ο χρόνος είναι γραμμικός. Η υποστήριξη διακλαδιζόμενου χρόνου αποτελεί μια ενδιαφέρουσα κατεύθυνση και θα χρησίμευε σε αρκετές πρακτικές εφαρμογές. Σε μια τέτοια περίπτωση, οι αλλαγές θα μπορούσαν να συμβούν σε οποιοδήποτε χρονικό στιγμιότυπο της βάσης, και όχι μόνο στο τρέχον στιγμιότυπο. Η υποστήριξη διαστάσεων με ιεραρχικά πεδία ορισμού είναι ίσως ένα πρώτο βήμα προς αυτή την κατεύθυνση.
- *MXML*: Η XML και οι συναφείς τεχνολογίες αποτελούν μια ιδιαίτερα ενεργή ερευνητική περιοχή. Απαιτείται μια πιο αυστηρή διερεύνηση του πως το ερμηνευτικό περιβάλλον μπορεί να ενσωματωθεί στις τεχνολογίες αυτές, συμπεριλαμβανομένων και των γλωσσών επερωτήσεων για XML. Επιπλέον, θα ήταν ενδιαφέρον να προσαρμόσουμε στην XML την προσέγγισή μας για την αναπαράσταση ιστορικών των βάσεων δεδομένων OEM, έτσι ώστε να είναι δυνατόν να αναπαρασταθούν ιστορικά δεδομένων XML.

Ο ρόλος του ερμηνευτικού περιβάλλοντος στην διαχείριση της πολυμορφίας των δεδομένων του Ιστού προσελκύει αυξανόμενη προσοχή. Εκτιμούμε ότι τα MSSD αντιμετωπίζουν ένα σημαντικό πρόβλημα των βάσεων δεδομένων στο σημερινό παγκοσμιοποιημένο περιβάλλον, και ότι έχουν τη δυνατότητα να εφαρμοστούν πρακτικά σε διάφορες περιοχές. Έτσι, μια ακόμη ενδιαφέρουσα κατεύθυνση είναι η χρήση εννοιών των MSSD σε προβλήματα από διαφορετικά πεδία, όπως:

- Στην παροχή εξατομικευμένων πληροφοριών και υπηρεσιών, μέσω της προσαρμογής τους στα προφίλ των καταναλωτών.
- Στην ενοποίηση πληροφορίας, για την μοντελοποίηση αντικειμένων των οποίων η τιμή ή η δομή ποικίλει ανάλογα με την πηγή.

- Στις ψηφιακές βιβλιοθήκες, για την αναπαράσταση μεταπληροφορίας που ακολουθεί παρόμοιους αλλά όχι ταυτόσημους μορφότυπους, ανάλογα με την πηγή.
- Στην αναπαράσταση γεωγραφικής πληροφορίας, όπου πιθανές διαστάσεις θα μπορούσαν να είναι οι κλίμακα και το θέμα.





# ΠΑΡΑΡΤΗΜΑ Α: ΠΡΟΔΙΑΓΡΑΦΗ ΣΥΝΤΑΚΤΙΚΟΥ ΤΗΣ MQL

Ακολούθως προδιαγράφουμε το συντακτικό των εκφράσεων μονοπατιού περιβάλλοντος και της MQL σε Extended Backus-Naur Form [EBNF], ή αλλιώς EBNF για συντομία.

Σύμβολα που αρχίζουν με κεφαλαίο γράμμα μπορούν να ορισθούν μέσω μιας ομαλής έκφρασης, όπως για παράδειγμα:

```
LabelName ::= [A-Za-z0-9_]+
```

Αυτά τα σύμβολα δεν αναλύονται περαιτέρω εδώ. Σε αυτά τα σύμβολα περιλαμβάνονται τα: `Identifier`, που δηλώνει στοιχειοσειρές οι οποίες ταυτοποιούν μεταβλητές, `CxtOid` και `MldOid`, που υποδηλώνουν αναγνωριστικά (oids) αντικειμένων περιβάλλοντος και πολυδιάστατων αντικειμένων αντίστοιχα, `LabelName`, που δηλώνουν στοιχειοσειρές οι οποίες είναι ετικέτες ακμών οντότητας, και `QuotedStringLiteral`, `IntLiteral`, `RealLiteral`, που είναι αυτό που δηλώνει το όνομά τους.

Σύμβολα που αρχίζουν με μικρό γράμμα ορίζονται από παραγωγές EBNF, που περιλαμβάνονται παρακάτω. Δεν περιλαμβάνονται παραγωγές για το `labelNamePattern` και το `stringPattern`, που συμβολίζουν ομαλές εκφράσεις πάνω σε ετικέτες ακμών οντότητας και σε στοιχειοσειρές αντίστοιχα. Τέλος, η παραγωγή EBNF για το `cxtSpec` και το `dimList` δίνονται στο Κεφάλαιο 3.

## A.1 ΣΥΝΤΑΚΤΙΚΟ ΕΚΦΡΑΣΕΩΝ ΜΟΝΟΠΑΤΙΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ

Εδώ ορίζουμε τις εκφράσεις μονοπατιού περιβάλλοντος, χτίζοντας παραγωγές από τις ειδικότερες προς τις γενικότερες.

Ο Πίνακας A.1 δίνει παραγωγές για τους τύπους μεταβλητών στην MQL.

**Πίνακας A.1: Μεταβλητές (variables).**

```
cxtObjVar ::= Identifier
mldObjVar ::= "<" Identifier ">"
cxtVar    ::= "[" Identifier "]"
labelVar  ::= "$" Identifier
pathVar   ::= "@" Identifier
```

Ο Πίνακας A.2 ορίζει το συντακτικό των προτύπων περιβάλλοντος και των περιοριστών περιβάλλοντος.

**Πίνακας A.2: Πρότυπο περιβάλλοντος (context pattern) και περιοριστής περιβάλλοντος (context qualifier).**

```
cxtPattern ::= "[% dimList ("|" dimList)* "]"
cxtQualifier ::= cxtSpec | cxtPattern | cxtVar
```

Ο Πίνακας A.3 ορίζει το συντακτικό των τμημάτων οντότητας και των τμημάτων έκφρασης. Τμήματα οντότητας που πιθανώς ακολουθούνται από μια μεταβλητή ετικέτας, και τμήματα έκφρασης που πιθανώς περιλαμβάνουν μια μεταβλητή περιβάλλοντος σαν περιοριστή ρητού περιβάλλοντος ονομάζονται **επαυξημένα τμήματα**. Το τμήμα ρίζας ορίζεται σαν ένα ειδικό τμήμα οντότητας το οποίο δεν αρχίζει με τελεία.

**Πίνακας A.3: Τμήμα οντότητας (entity part) και τμήμα έκφρασης (facet part).**

```
ettLabelExpr ::= "-" | LabelName | "\"" labelNamePattern "\""
ettPart ::= "." cxtQualifier? ettLabelExpr
ettPartAugment ::= ettPart labelVar?
facetPart ::= "::" cxtQualifier? (cxtSpec | cxtPattern)
facetPartAugment ::= "::" cxtQualifier? cxtQualifier
rootPart ::= cxtQualifier? ettLabelExpr
rootPartAugment ::= rootPart labelVar?
| rootPartAugment "{" cxtObjVar "}"
| rootPartAugment "{" mldObjVar "}"
```

Ο Πίνακας A.4 δίνει τους συντακτικούς κανόνες για τα γενικευμένα στοιχεία εκφράσεων μονοπατιού περιβάλλοντος (gpre\_components). Ορίζονται τέσσερις τύποι στοιχείων, ανάλογα με τους τύπους του πρώτου και του τελευταίου τμήματος. Οι παραγωγές δεν επιτρέπουν σε ένα τμήμα έκφρασης να ακολουθείται από ένα άλλο τμήμα έκφρασης.

**Πίνακας A.4: Γενικευμένα στοιχεία έκφρασης μονοπατιού περιβάλλοντος (general context path expression components).**

```

gcpeComponEtt-Ett ::= ettPart
    | gcpeComponEtt-Ett gcpeComponEtt-Ett
    | gcpeComponEtt-Ett gcpeComponFacet-Ett
    | gcpeComponEtt-Facet gcpeComponEtt-Ett
    | "(" gcpeComponEtt-Ett "|" gcpeComponEtt-Ett ")"
    | "(" gcpeComponEtt-Ett ")" "+"

gcpeComponEtt-Facet ::= "." cxtQualifier? "#"
    | gcpeComponEtt-Ett gcpeComponEtt-Facet
    | gcpeComponEtt-Ett gcpeComponFacet-Facet
    | gcpeComponEtt-Facet gcpeComponEtt-Facet
    | "(" gcpeComponEtt-Facet "|" gcpeComponEtt-Facet ")"
    | "(" gcpeComponEtt-Facet ")" ("*" | "?" | "+")

gcpeComponFacet-Facet ::= facetPart
    | gcpeComponFacet-Ett gcpeComponEtt-Facet
    | gcpeComponFacet-Ett gcpeComponFacet-Facet
    | gcpeComponFacet-Facet gcpeComponEtt-Facet
    | "(" gcpeComponFacet-Facet "|" gcpeComponFacet-Facet ")"
    | "(" gcpeComponFacet-Facet ")" "?"

gcpeComponFacet-Ett ::= ":" cxtQualifier? "#"
    | gcpeComponFacet-Ett gcpeComponEtt-Ett
    | gcpeComponFacet-Ett gcpeComponFacet-Ett
    | gcpeComponFacet-Facet gcpeComponEtt-Ett
    | "(" gcpeComponFacet-Ett "|" gcpeComponFacet-Ett ")"
    | "(" gcpeComponFacet-Ett ")" ("*" | "?" | "+")

```

Ο Πίνακας A.5 επανξάνει τα `gcpe_components` με μεταβλητές ετικέτας, μονοπατιού, και αντικειμένων, οι οποίες δεν προβλέπονται μέσα στα `gcpe_components`. Επιπλέον, σε ένα **επανξημένο `gcpe_component`** μια μεταβλητή περιβάλλοντος μπορεί να χρησιμοποιηθεί σαν περιοριστής ρητού περιβάλλοντος (ένα είδος μεταβλητής ετικέτας για ακμές περιβάλλοντος), το οποίο δεν είναι επίσης επιτρεπτό μέσα σε ένα `gcpe_component`<sup>121</sup>.

<sup>121</sup> Τα επανξημένα `gcpe_components` αντιστοιχούν στα *qualified gcpe\_components* της Lorel [AQM+97].

**Πίνακας A.5: Επαυξημένα γενικευμένα στοιχεία έκφρασης μονοπατιού περιβάλλοντος (augmented general context path expression components).**

```

gcpeAugmentEtt-Ett ::= ettPartAugment
    | gcpeComponEtt-Ett
    | gcpeAugmentEtt-Ett gcpeAugmentEtt-Ett
    | gcpeAugmentEtt-Ett gcpeAugmentFacet-Ett
    | gcpeAugmentEtt-Facet gcpeAugmentEtt-Ett
    | "(" gcpeAugmentEtt-Ett ")" pathVar
    | gcpeAugmentEtt-Ett "{" cxtObjVar}"
    | gcpeAugmentEtt-Ett "{" mldObjVar}"

gcpeAugmentEtt-Facet ::=
    gcpeComponEtt-Facet
    | gcpeAugmentEtt-Ett gcpeAugmentEtt-Facet
    | gcpeAugmentEtt-Ett gcpeAugmentFacet-Facet
    | gcpeAugmentEtt-Facet gcpeAugmentEtt-Facet
    | "(" gcpeAugmentEtt-Facet ")" pathVar
    | gcpeAugmentEtt-Facet "{" cxtObjVar}"

gcpeAugmentFacet-Facet ::= facetPartAugment
    | gcpeComponFacet-Facet
    | gcpeAugmentFacet-Ett gcpeAugmentEtt-Facet
    | gcpeAugmentFacet-Ett gcpeAugmentFacet-Facet
    | gcpeAugmentFacet-Facet gcpeAugmentEtt-Facet
    | "(" gcpeAugmentFacet-Facet ")" pathVar
    | gcpeAugmentFacet-Facet "{" cxtObjVar}"

gcpeAugmentFacet-Ett ::=
    gcpeComponFacet-Ett
    | gcpeAugmentFacet-Ett gcpeAugmentEtt-Ett
    | gcpeAugmentFacet-Ett gcpeAugmentFacet-Ett
    | gcpeAugmentFacet-Facet gcpeAugmentEtt-Ett
    | "(" gcpeAugmentFacet-Ett ")" pathVar
    | gcpeAugmentFacet-Ett "{" cxtObjVar}"
    | gcpeAugmentFacet-Ett "{" mldObjVar}"

```

Ο Πίνακας A.6 ορίζει τις εκφράσεις μονοπατιού περιβάλλοντος σαν ένα αναγνωριστικό κάποιου τύπου, που ακολουθείται από ένα επαυξημένο gcpe\_component συμβατού τύπου (ένα επαυξημένο gcpe\_component μπορεί να εμπεριέχει οποιονδήποτε αριθμό από επαυξημένα gcpe\_components). Διακρίνονται δύο τύποι εκφράσεων μονοπατιού περιβάλλοντος, ανάλογα με τον τύπο του τελευταίου τμήματος.

**Πίνακας A.6: Εκφράσεις μονοπατιού περιβάλλοντος (context path expressions).**

```

cxtPathExpr-Ett ::=
    (CxtOid | cxtObjVar) gcpeAugmentEtt-Ett
    | (MldOid | mldObjVar | rootPartAugment) (gcpeAugmentFacet-Ett
    | gcpeAugmentEtt-Ett)

cxtPathExpr-Facet ::=
    (CxtOid | cxtObjVar) gcpeAugmentEtt-Facet
    | (MldOid | mldObjVar | rootPartAugment) (gcpeAugmentFacet-Facet
    | gcpeAugmentEtt-Facet)

cxtPathExpr ::= cxtPathExpr-Ett | cxtPathExpr-Facet

```

Έχοντας ορίσει τις εκφράσεις μονοπατιού περιβάλλοντος, συνεχίζουμε με την καθεαυτή σύνταξη της MQL.

## A.2 ΣΥΝΤΑΚΤΙΚΟ ΤΗΣ MQL

Οι συντακτικοί κανόνες της MQL καταγράφονται πιο κάτω αρχίζοντας από τους γενικότερους προς τους ειδικότερους.

Ο Πίνακας A.7 δίνει την γενική δομή μιας επερώτησης MQL. Διακρίνουμε δυο κατηγορίες επερωτήσεων, ανάλογα με τον τύπο των ακμών που ξεκινούν από την ρίζα του αποτελέσματος, όπως αυτός προσδιορίζεται στον όρο `select`. Μόνο επερωτήσεις του ίδιου τύπου μπορούν να συνδυάσουν τα αποτελέσματά τους χρησιμοποιώντας τα `union`, `intersect`, και `except`.

**Πίνακας A.7: Γενική δομή επερωτήσεων MQL.**

```

query      ::= queryEtt | queryCxt

queryEtt ::= clauseQueryEtt
          | queryEtt "union" queryEtt
          | queryEtt "intersect" queryEtt
          | queryEtt "except" queryEtt
          | "(" queryEtt ")"

queryCxt ::= clauseQueryCxt
          | queryCxt "union" queryCxt
          | queryCxt "intersect" queryCxt
          | queryCxt "except" queryCxt
          | "(" queryCxt ")"

clauseQueryEtt ::=
    "select" "holding"? "distinct"?
    (labeledTemplateEtt | "{" templateEtt "}")
    queryBody
clauseQueryCxt ::=
    "select" "holding"? "distinct"?
    (labeledTemplateCxt | "<" templateCxt ">")
    queryBody

queryBody ::= ("context" contextExpr ("," contextExpr)*)?
            ("from" fromExpr ("," fromExpr)*)?
            ("where" predicate)?
            ("within" cxtPredicate)?

```

Ο Πίνακας A.8 ορίζει την σύνταξη του οδηγού (template) για τις εκφράσεις `mssd-expression` στο `select`. Η πρώτη ετικέτα στον οδηγό πρέπει να υπάρχει υποχρεωτικά εάν λείπουν οι εξωτερικές αγκύλες που καθορίζουν τον τύπο της ρίζας.

**Πίνακας A.8: Οδηγός (template) αποτελεσμάτων στον όρο “select”.**

```

labeledTemplateEtt ::=
    templObjLabelEtt ":" templObjValue ("," templateEtt)?
labeledTemplateCxt ::=
    templObjLabelCxt ":" templObjValue ("," templateCxt)?

templateEtt ::=
    (templObjLabelEtt ":")? templObjValue ("," templateEtt)?
templateCxt ::=
    (templObjLabelCxt ":")? templObjValue ("," templateCxt)?

templObjLabelEtt ::= LabelName | labelVar
templObjLabelCxt ::= cxtSpec | cxtVar

templObjValue ::= "{" templateEtt "}" | "<" templateCxt ">"
                | atomicValue | cxtPathExpr | safeQuery

safeQuery ::= "(" query ")"

```

Ο Πίνακας A.9 ορίζει τις ατομικές τιμές που μπορούν να πάρουν την θέση αντικειμένων στον οδηγό του select, ή να χρησιμοποιηθούν στον όρο where για συγκρίσεις.

**Πίνακας A.9: Τιμές αντικειμένων στον οδηγό του όρου “select”.**

```

atomicValue ::= variable | constant | numValue
              | "oid" "(" (cxtObjVar | mldObjVar) ")"
              | "pathof" "(" pathVar ")"

variable ::= cxtObjVar | mldObjVar | cxtVar | labelVar | pathVar

constant ::= QuotedStringLiteral | numConstant
numConstant ::= IntLiteral | RealLiteral

numValue ::= cxtObjVar | numConstant
            | numValue arithOp numValue
            | "(" numValue ")"
            | "-" numValue
            | "abs" "(" numValue ")"
            | numAggrFunct "(" (cxtObjVar | cxtPathExpr-Facet) ")"
            | "count" "(" (variable | cxtPathExpr) ")"

arithOp ::= "+" | "-" | "*" | "/" | "mod"
numAggrFunct ::= "min" | "max" | "sum" | "avg"

```

Ο Πίνακας A.10 προδιαγράφει τον ορισμό μιας νέας μεταβλητής περιβάλλοντος στον όρο context.

**Πίνακας A.10: Στοιχεία του όρου “context”.**

```

contextExpr ::= cxtVar "as" (cxtValue
                        | "extension" "(" cxtValue ")")
cxtValue    ::= cxtSpec | cxtVar
                | cxtValue cxtOp cxtValue
                | cxtValue cxtOp cxtPattern
                | cxtPattern cxtOp cxtValue
                | "(" cxtValue ")"
                | "union" "(" cxtValue ")"
                | "intersect" "(" cxtValue ")"

cxtOp       ::= "+" | "-" | "*"

```

Ο Πίνακας A.11 δίνει το συντακτικό ενός στοιχείου του όρου `from`, που αποτελείται από μια έκφραση μονοπατιού περιβάλλοντος και (ίσως) από μια μεταβλητή αντικειμένου συμβατού τύπου.

**Πίνακας A.11: Στοιχεία του όρου “from”.**

```

fromExpr ::= cxtPathExpr-Facet ("as"? cxtObjVar)?
            | cxtPathExpr-Ett ("as"? (cxtObjVar | mldObjVar))?
            | rootPartAugment ("as"? (cxtObjVar | mldObjVar))?

```

Ο Πίνακας A.12 ορίζει το συντακτικό του όρου `where`.

**Πίνακας A.12: Δομή του όρου “where”.**

```

predicate ::= "not" predicate
            | predicate "and" predicate
            | predicate "or" predicate
            | "(" predicate ")"
            | boolConstant
            | (atomicValue | cxtPathExpr) "like" "" stringPattern ""
            | (atomicValue | cxtPathExpr) compOp compQuantifier?
              (atomicValue | cxtPathExpr | safeQuery)
            | "exists" (variable | cxtPathExpr)
            | "exists" cxtObjVar "in" cxtPathExpr-Facet "(" predicate ")"
            | "exists" (cxtObjVar | mldObjVar) "in"
              (cxtPathExpr-Ett | safeQuery) "(" predicate ")"
            | "for all" cxtObjVar "in" cxtPathExpr-Facet "(" predicate ")"
            | "for all" (cxtObjVar | mldObjVar) "in"
              (cxtPathExpr-Ett | safeQuery) "(" predicate ")"

boolConstant ::= "true" | "false"
compOp       ::= "<" | "<=" | "=" | "==" | "!=" | ">=" | ">"
compQuantifier ::= "any" | "all"

```

Ο Πίνακας A.13 ορίζει το συντακτικό του όρου `within`.

**Πίνακας A.13: Δομή του όρου “within”.**

```
cxtPredicate ::= "not" cxtPredicate
              | cxtPredicate "and" cxtPredicate
              | cxtPredicate "or" cxtPredicate
              | "(" cxtPredicate ")"
              | boolConstant
              | cxtValue cxtCompOp cxtValue
              | cxtValue cxtCompOp cxtPattern
              | cxtPattern cxtCompOp cxtValue

cxtCompOp    ::= "<" | "<=" | "=" | "!=" | ">=" | ">"
```

Εδώ ολοκληρώνεται η προδιαγραφή του συντακτικού της MQL.



## ΑΝΑΦΟΡΕΣ

- [Abi97] Serge Abiteboul. Querying Semi-Structured Data. In *Proceedings of the 6th International Conference on Database Theory (ICDT'97)*, Delphi, Greece, January 1997.
- [Abi99] Serge Abiteboul. On Views and XML. In *Proceedings of the 18th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS'99)*, May 1999.
- [ABS00] Serge Abiteboul, Peter Buneman, and Dan Suciu. *Data on the Web: From Relations to Semistructured Data and XML*. Morgan Kaufmann, 2000.
- [AFJ91] E. A. Ashcroft, A. A. Faustini, and R. Jagannathan. An Intensional Language for Parallel Applications Programming. *Parallel Functional Languages and Compilers*, pages 11-49. ACM Press, 1991.
- [AFJW95] E. A. Ashcroft, A. A. Faustini, R. Jagannathan, and W. W. Wadge. *Multidimensional Programming*. Oxford University Press, 1995.
- [AM98] Gustavo O. Arocena, and Alberto O. Mendelzon. WebOQL: Restructuring Documents, Databases and Webs. In *Proceedings of the International Conference on Data Engineering (ICDE'98)*, Orlando, Florida, 1998.
- [AMM97] Paolo Atzeni, Giansalvatore Mecca, and Paolo Merialdo. To Weave the Web. In *Proceedings of the 23rd International Conference on Very Large DataBases (VLDB'97)*, Athens, Greece, August 1997.
- [AMM97a] Paolo Atzeni, Giansalvatore Mecca, and Paolo Merialdo. Semistructured and Structured Data in the Web: Going Back and Forth. *SIGMOD Record*, 26(4):16-23, December 1997.
- [AQM+97] Serge Abiteboul, Dallan Quass, Jason McHugh, Jennifer Widom, and Jannet L. Wiener. The Lorel Query Language for Semistructured Data. *International Journal on Digital Libraries*, 1(1): 68-88, April 1997.
- [AYU00] Toshiyuki Amagasa, Masatoshi Yoshikawa, and Shunsuke Uemura. A Data Model for Temporal XML Documents. In *Proceedings of the 11th International Conference on Database and Expert Systems Applications (DEXA'00)*, London, September 2000. Lecture Notes in Computer Science (LNCS) 1873, pages 334-344, Springer-Verlag, 2000.

- [BBC+98] Phil Bernstein, Michael Brodie, Stefano Ceri, David DeWitt, Mike Franklin, Hector Garcia-Molina, Jim Gray, Jerry Held, Joe Hellerstein, H. V. Jagadish, Michael Lesk, Dave Maier, Jeff Naughton, Hamid Pirahesh, Mike Stonebraker, and Jeff Ullman. The Asilomar Report on Database Research. *SIGMOD Record*, 27(4): 74-80, 1998.
- [BC00] Angela Bonifati, and Stefano Ceri. Comparative Analysis of Five XML Query Languages. *SIGMOD Record*, 29(1), March 2000.
- [BCW93] M. Baudinet, J. Chomicki, and P. Wolper. Temporal Deductive Databases. *Temporal Databases: Theory, Design, and Implementation*, pages 294-320. The Benjamin / Cummings Publishing Company, 1993.
- [BDHS96] Peter Buneman, Susan Davidson, Gerd Hillebrand, and Dan Suciu. A Query Language and Optimization Techniques for Unstructured Data. In *Proceedings of the ACM International Conference on Management of Data*, 1996.
- [BDS95] Peter Buneman, Susan Davidson, and Dan Suciu. Programming Constructs for Unstructured Data. In *Proceedings of the Workshop on Database Programming Languages*, Italy, 1995.
- [BDT98] Peter Buneman, Alin Deutsch, and Wang-Chiew Tan. A Deterministic Model for Semistructured Data. In *Workshop on Query Processing for Semistructured Data and Non-Standard Data Formats*, 1998.
- [BFS00] Peter Buneman, Mary Fernandez, and Dan Suciu. UnQL: A Query Language and Algebra for Semistructured Data Based on Structural Recursion. *The VLBD Journal*, 9(1): 76-110, 2000.
- [BMPW98] Sergey Brin, Rajeev Motwani, Lawrence Page, and Terry Winograd. What Can You Do with a Web in Your Pocket? *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 21(2): 37-47, June 1998.
- [Bro98] Gord Brown. IHTML 2: Design and Implementation. In *Proceedings of the 11th International Symposium on Languages for Intensional Programming*, pages 1-13, 1998.
- [Bro98a] Gord Brown. Intensional HTML 2: A Practical Approach. Master's Thesis, Department of Computer Science, University of Victoria, Canada, 1998.
- [Bun94] Harry Bunt. Context and Dialog Control. *THINK Quarterly*, 3(1): 19-31, 1994.
- [Bun97] Peter Buneman. Semistructured Data (Tutorial). In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS'97)*, pages 117-121, Tucson, Arizona, 1997.
- [CAW98] Sudarshan Chawathe, Serge Abiteboul, and Jennifer Widom. Representing and Querying Changes in Semistructured Data. In *Proceedings of the 14th International Conference on Data Engineering*, Orlando, Florida, February 1998.

- [CAW99] Sudarshan Chawathe, Serge Abiteboul, and Jennifer Widom. Managing Historical Semistructured Data. *Theory and Practice of Object Systems*, 24(4): 1-20, 1999.
- [CCD+99] Stefano Ceri, Sara Comai, Ernesto Damiani, Piero Fraternali, Stefano Paraboschi, and Letizia Tanca. XML-GL: a Graphical Language for Querying and Restructuring XML Documents. In *Systemi Evaluti per Busi di Dati (SEBD'99)*, pages 151-165, Como, Italy, 1999.
- [CD97] S. Chaudhuri, and U. Dayal. An Overview of Data Warehousing and OLAP Technology. *SIGMOD Record*, 26(1), March 1997.
- [CGH+94] Sudarshan Chawathe, Hector Garcia-Molina, Joachim Hammer, Kelly Ireland, Yannis Papakonstantinou, Jeffrey Ullman, and Jennifer Widom. The TSIMMIS Project: Integration of Heterogeneous Information Sources. In *Proceedings of the Conference on Information Processing Society of Japan (IPSJ'94)*, pages 7-18, Tokyo, Japan, October 1994.
- [CH98] William W. Cohen, and Haym Hirsh. Joins that Generalize: Text Classification Using WHIRL. In *Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining*, 1998.
- [Cha99] Sudarshan S. Chawathe. Describing and Manipulating XML Data. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 22(3): 3-9, September 1999.
- [CHS+95] M. J. Carey, L. M. Haas, P. M. Schwarz, M. Arya, W. F. Cody, R. Fagin, M. Flickner, A. W. Luniewski, W. Niblack, D. Petkovic, J. Thomas, J. H. Williams, and E. L. Wimmers. Towards Heterogeneous Multimedia Information Systems: The Garlic Approach. In *Proceedings of the International Workshop on Research Issues in Data Engineering*, March 1995.
- [CK00] Guanling Chen, and David Kotz. A Survey of Context-Aware Mobile Computing Research. Dartmouth College Computer Science Technical Report TR2000-381, November 2000. <ftp://ftp.cs.dartmouth.edu/TR/TR2000-381.ps.Z>
- [CL97] Patrick Chan, and Rosanna Lee. *The Java Class Libraries: An Annotated Reference*. Addison-Wesley, 1997.
- [Coh98] William W. Cohen. Integration of Heterogeneous Databases Without Common Domains Using Queries Based on textual Similarity. *ACM SIGMOD Conference*, June 1998.
- [Coh98a] William W. Cohen. The WHIRL Approach to Integration: An Overview. *AAAI Workshop on AI and Information Integration*, 1998.
- [Coh98b] William W. Cohen. A Web-based Information System that Reasons with Structured Collections of Text. In *Proceedings of the 2nd International Conference on Autonomous Agents (Agents'98)*, Minneapolis, May 1998.
- [Coll96] G. Colliat. OLAP, Relational, and Multidimensional Database Systems. *SIGMOD Record*, 25(3), September 1996.

- [CRF00] Don Chamberlin, Jonathan Robie, and Daniela Florescu. Quilt: An XML Query Language for Heterogeneous Data Sources. In *Proceedings of the 3rd International ACM SIGMOD Workshop on the Web and Databases (WebDB 2000)*, Dallas, Texas, May 2000.
- [CS00] Sophie Cluet, and Jérôme Siméon. YATL: a Functional and Declarative Language for XML. Draft Manuscript, May 2000. <http://www-db.research.bell-labs.com/user/simeon/icfp.ps>
- [DBJ99] C. Dyreson, M. Böhlen, and C. Jensen. Capturing and Querying Multiple Aspects of Semistructured Data. In *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB'99)*, pages 290-301, 1999.
- [DFF+99] Alin Deutsch, Mary Fernandez, Daniela Florescu, Alon Levy, David Maier, and Dan Suciu. Querying XML Data. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 22(3): 10-18, September 1999.
- [DFF+99a] Alin Deutsch, Mary Fernandez, Daniela Florescu, Alon Levy, and Dan Suciu. A Query Language for XML. In *Proceedings of the 8th International World Wide Web Conference (WWW8)*, Toronto, Canada, May 1999.
- [DOQT01] E. Damiani, B. Oliboni, E. Quintarelli, and L. Tanca. Modeling Users' Navigation History. In *Proceedings of the Workshop on Intelligent Techniques for Web Personalization*, Seattle, USA, August 2001.
- [DS96] Debabrata Dey, and Sumit Sarkar. A Probabilistic Relational Model and Algebra. In *ACM Transactions on Database Systems (TODS'96)*, 21(3): 339-369, September 1996.
- [DS98] Debabrata Dey, and Sumit Sarkar. PSQL: A Query Language for Probabilistic Relational Data. In *Data & Knowledge Engineering (DKE'98)*, 28(1): 107-120, 1998.
- [Dyr01] Curtis E. Dyreson. Observing Transaction-Time Semantics with TTXPath. In *Proceedings of the 2nd International Conference on Web Information Systems Engineering (WISE'01)*, Kyoto, Japan, December 2001. IEEE Computer Press, pages 193-202, 2001.
- [EBNF] ISO (International Organization for Standardization). *ISO / IEC 14977: 1996 (E)*. Extended Backus-Naur Form (EBNF), 1996.
- [EPS03] Antonis Efandis, Kostis Pristouris, and Yannis Stavrakas. A Query Evaluation System for Multidimensional Semistructured Data. To appear in *Proceedings of the 1st Balkan Conference on Informatics (BCI 2003)*, Thessaloniki, Greece, November 2003.
- [FDFP95] Adam Farquhar, Angela Dappert, Richard Fikes, and Wanda Pratt. Integrating Information Sources Using Context Logic. In *Proceedings of the AAAI Spring Symposium on Information Gathering from Distributed Heterogeneous Environments*, 1995.

- [FFK+98] M. Fernandez, D. Florescu, J. Kang, A. Levy, and D. Suciu. Catching the Boat with Strudel: Experiences with a Web-Site Management System. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD'98)*, Seattle, 1998.
- [FFLS97] Mary Fernandez, Daniela Florescu, Alon Levy, and Dan Suciu. A Query Language for a Web-Site Management System. *SIGMOD Record*, 26(3): 4-11, September 1997.
- [FFLS98] Mary Fernandez, Daniela Florescu, Alon Levy, and Dan Suciu. Web-Site Management: The Strudel Approach. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 21(2): 14-20, June 1998.
- [Fit93] Melvin Fitting. Basic Modal Logic. *Handbook of Logic in Artificial Intelligence and Logic Programming*, Volume I. Oxford University Press, 1993.
- [FLM98] Daniela Florescu, Alon Levy, and Alberto Mendelzon. Database Techniques for the World-Wide Web: A Survey. *SIGMOD Record*, 27(3): 59-74, September 1998.
- [Fri97] Jeffrey E.F. Friedl. *Mastering Regular Expressions*. O'Reilly, 1997.
- [FSW99] Mary Fernandez, Jérôme Siméon, and Philip Wadler. XML Query Languages: Experiences and Exemplars. Draft Manuscript, Communication to the XML Query W3C Working Group, September 1999. <http://www-db.research.bell-labs.com/user/simeon/xquery.html>
- [GG01] Chiara Ghidini, and Fausto Giunchiglia. Local Model Semantics, or contextual reasoning = locality + compatibility. *Artificial Intelligence* 127, pages 221-259, 2001.
- [GHI+95] Hector Garcia-Molina, Joachim Hammer, Kelly Ireland, Yannis Papakonstantinou, Jeffrey Ullman, and Jennifer Widom. Integrating and Accessing Heterogeneous Information Sources in TSIMMIS. In *Proceedings of the AAI Symposium on Information Gathering*, Stanford, California, March 1995.
- [Giu93] Fausto Giunchiglia. Contextual Reasoning. *Epistemologia* (Special Issue on I Linguaggi e le Macchine) XVI, pages 345-364, 1993.
- [GM00] F. Grandi, and F. Mandreoli. The Valid Web: an XML/XSL Infrastructure for Temporal Management of Web Documents. In *Proceedings of the 1st International Conference on Advances in Information Systems (ADVIS'02)*, pages 294-303, Izmir, Turkey, October 2000.
- [GMW99] Roy Goldman, Jason McHugh, and Jennifer Widom. From Semistructured Data to XML: Migrating the Lore Data Model and Query Language. In *Proceedings of the 2nd International Workshop on the Web and Databases (WebDB'99)*, Philadelphia, Pennsylvania, June 1999.
- [GP98] Luis Gravano, and Yannis Papakonstantinou. Mediating and Metasearching on the Internet. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 21(2): 28-36, June 1998.

- [GPQ+97] Hector Garcia-Molina, Yannis Papakonstantinou, Dallon Quass, Anand Rajaraman, Yehoshua Sagiv, Jeffrey Ullman, Vasilis Vassalos, and Jennifer Widom. The TSIMMIS Approach to Mediation: Data Models and Languages. *Journal of Intelligent Information Systems*, 8(2): 117-132, March-April 1997.
- [Gra02] Fabio Grandi. XML Representation and Management of Temporal Information for the Web-Based Cultural Heritage Applications. *Data Science Journal*, 1(1): 68-83, 2002.
- [GRGK97] Venkat N. Gudivada, Vijay V. Raghavan, William I. Grosky, and Rajesh Kananagottu. Information Retrieval on the World Wide Web. *IEEE Internet Computing*, September-October 1997.
- [GS03] Manolis Gergatsoulis, and Yannis Stavarakas. Representing Changes in XML Documents Using Dimensions. To appear in *Proceedings of the XML Database Symposium (XSym 2003)*, in Conjunction with VLDB 2003, Berlin, Germany, September 2003.
- [GS98] Fabio Grandi, and Maria Rita Scalas. Extending Temporal Database Concepts to the World Wide Web. In *Proceedings of SEBD'98, National Conference on Advanced Database Systems*, pages 53-70, Italy, June 1998.
- [GS98a] C. Ghidini, and L. Serafini. Model Theoretic Semantics for Information Integration. In *Proceedings of the 8th International Conference on Artificial Intelligence, Methodology, Systems, and Applications (AIMSA '98)*, Sozopol, Bulgaria, 1998.
- [GS98b] Ghiara Ghidini, and Luciano Serafini. Information Integration for Electronic Commerce. In *Proceedings of the Workshop on Agent Mediated Electronic Trading (AMET'98)*, Minneapolis, 1998.
- [GSK+01] Manolis Gergatsoulis, Yannis Stavarakas, Dimitris Karteris, Athina Mouzaki, and Dimitris Sterpis. A Web-based System for Handling Multidimensional Information through MXML. In *Proceedings of the 5th East European Conference on Advances in Databases and Information Systems (ADBIS'01)*, Vilnius, Lithuania, September 2001. Lecture Notes in Computer Science (LNCS) 2151, pages 352-365, Springer-Verlag, 2001.
- [GSK01] Manolis Gergatsoulis, Yannis Stavarakas, and Dimitris Karteris. Incorporating Dimensions to XML and DTD. In *Proceedings of the 12th International Conference on Database and Expert Systems Applications (DEXA'01)*, Munich, Germany, September 2001. Lecture Notes in Computer Science (LNCS) 2113, pages 646-656, Springer-Verlag, 2001.
- [HKWY96] Laura M. Haas, Donald Kossmann, Edward L. Wimmers, and Jun Yang. An Optimizer for Heterogenous Systems with NonStandard Data and Search Capabilities. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 1996.

- [HLLS97] Rainer Himmeröder, Georg Lausen, Bertram Ludäscher, and Christian Schleppehorst. On a Declarative Semantics for Web Queries. In *Proceedings of the 5th International Conference on Deductive and Object-Oriented Databases (DOOD'97)*, Montreux, Switzerland, December 1997.
- [HMN+97] L. M. Haas, R. J. Miller, B. Niswonger, M. Tork Roth, P. M. Schwarz, and E. L. Wimmers. Transforming Heterogeneous Data with Database Middleware: Beyond Integration. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 1997.
- [HTML] The World Wide Web Consortium (W3C). HTML 4.01 Specification, 1999. <http://www.w3.org/TR/html4/>
- [HTTP] R. Fielding, J. Gettys, J. Mogul, H. Frystyk Nielsen, and T. Berners-Lee. HTTP Version 1.1, 1997. <http://www.ietf.org/rfc/rfc2068.txt>
- [ISB95] Tomás Isakowitz, Edward A. Stohr, and P. Balasubramanian. RMM: A Methodology for Structured Hypermedia Design. *Communications of the ACM*, August 1995.
- [JAVA] Sun Microsystems, Java, 2003. <http://java.sun.com>
- [JCG+92] C. S. Jensen, J. Clifford, S. K. Gadia, A. Segev, and R. T. Snodgrass. A Glossary of Temporal Database Concepts. *SIGMOD Record*, 21(3): 35-43, September 1992.
- [JSCR] Netscape Communications Corporation, Javascript, 2003. <http://www.netscape.com/eng/mozilla/3.0/handbook/javascript/>
- [Kel97] John Kelly. *The Essence of Logic*. Prentice Hall, 1997.
- [KLSS95] Thomas Kirk, Alon Y. Levy, Yehoshua Sagiv, and Divesh Srivastava. The Information Manifold. In *Proceedings of the AAAI Spring Symposium on Information Gathering in Distributed Heterogeneous Environments*, Stanford, California, March 1995.
- [KR88] Brian Kernighan, and Dennis Ritchie. *The C Programming Language*. Prentice Hall, 1988.
- [KS95] David Konopnicki, and Oded Shmueli. W3QS: A Query System for the World-Wide Web. In *Proceedings of the 21st International Conference on Very Large DataBases (VLDB'95)*, Zurich, Switzerland, 1995.
- [LIN] LINUX Online, 2003. <http://www.linux.org/>
- [LMSS95] Alon Y. Levy, Alberto O. Mendelzon, Yehoshua Sagiv, and Divesh Srivastava. Answering Queries Using Views. In *Proceedings of the 14th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, San Jose, California, 1995.

- [LSK95] Alon Y. Levy, Divesh Srivastava, and Thomas Kirk. Data Model and Query Evaluation in Global Information Systems. *Journal of Intelligent Information Systems*, Special Issue on Networked Information Discovery and Retrieval, 1995.
- [LSS96] Laks V. S. Lakshmanan, Fereidoon Sadri, and Iyer N. Subramanian. A Declarative Language for Querying and Restructuring the Web. In *Proceedings of the 6th International Workshop on Research Issues in Data Engineering (RIDE'96)*, New Orleans, February 1996.
- [MAG+97] J. McHugh, S. Abiteboul, R. Goldman, D. Quass, and J. Widom. LORE: A Database Management System for Semistructured Data. *SIGMOD Record*, 26(3): 54-66, September 1997.
- [Mai98] David Maier. Database Desiderata for an XML Query Language, 1998. <http://www.w3.org/TandS/QL/QL98/pp/maier.html>
- [Mal98] Susan Malaika. Resistance is Futile: The Web Will Assimilate Your Database. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 21(2): 4-13, June 1998.
- [MGSI01] Theodoros Mitakos, Manolis Gergatsoulis, Yannis Stavarakas, and Efstathios V. Ioannidis. Representing Time-Dependent Information in Multidimensional XML. In *Proceedings of the 23rd International Conference on Information Technology Interfaces (ITI'01)*, Pula, Croatia, June 2001.
- [MGSI01a] Theodoros Mitakos, Manolis Gergatsoulis, Yannis Stavarakas, and Efstathios V. Ioannidis. Representing Time-Dependent Information in Multidimensional XML. *Journal of Computing and Information Technology*, 9(3): 233-238, 2001.
- [MM95] John Mylopoulos, and Renate Motschnig-Pitrik. Partitioning Information Bases with Contexts. In *Proceedings of the 3rd International Conference on Cooperative Information Systems (CoopIS'95)*, pages 44-55, Vienna, Austria, May 1995.
- [MMAC99] Giansalvatore Mecca, Paolo Merialdo, Paolo Atzeni, and Valter Crescenzi. The (Short) ARANEUS Guide to Web-Site Development. In *Proceedings of the ACM SIGMOD Workshop on the Web and Databases (WebDB'99)*, Philadelphia, Pennsylvania, June 1999.
- [MMM96] Alberto O. Mendelzon, George A. Mihaila, and Tova Milo. Querying the World Wide Web. In *Proceedings of the 1st International Conference on Parallel and Distributed Information Systems (PDIS'96)*, 1996.
- [MMM97] Alberto O. Mendelzon, George A. Mihaila, and Tova Milo. Querying the World Wide Web. *International Journal of Digital Libraries*, 1(1): 54-67, April 1997.
- [NJ02] Andrew Nierman, and H. V. Jagadish. ProTDB: Probabilistic Data in XML. In *Proceedings of the 28th International Conference on Very Large DataBases (VLDB'02)*, Hong Kong, China, 2002.



- [Nor02] Kjetil Nørnvåg. Algorithms for Temporal Query Operators in XML Databases. In *Proceedings of EDBT Workshops*. Lecture Notes in Computer Science (LNCS) 2490, pages 169-183, Springer-Verlag, 2002.
- [NP03] Moira C. Norrie, and Alexios Palinginis. Empowering Databases for Context-Dependent Information Delivery. To appear in *Proceedings of the CAiSE'03 Workshop on Ubiquitous Mobile Information and Collaboration Systems (UMICS'03)*, Austria, June 2003.
- [NP03a] M. C. Norrie, and A. Palinginis. From State to Structure: an XML Web Publishing Framework. To appear in *Proceedings of the 15th International Conference on Advanced Information Systems Engineering (CAiSE'03)*, Austria, June 2003.
- [OD97] M. A. Orgun, and W. Du. Multi-dimensional Logic Programming: Theoretical Foundations. *Theoretical Computer Science*, 158(2): 319-345, 1997.
- [ODMG] Rick Cattell et al. *The Object Database Standard: ODMG-93, Release 1.2*. Morgan Kaufmann Publishers, 1996.
- [OQT01] Barbara Oliboni, Elisa Quintarelli, and Letizia Tanca. Temporal Aspects of Semistructured Data. In *Proceedings of the 8th International Symposium on Temporal Representation and Reasoning (TIME'01)*, pages 119-127, 2001.
- [Org96] M. A. Orgun. On Temporal Deductive Databases. *Computational Intelligence*, 12(2): 235-259, 1996.
- [OS95] Gultekin Özsoyoğlu, and Richard T. Snodgrass. Temporal and Real-Time Databases: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 7(4): 513-532, August 1995.
- [OS99] Aris M. Ouksel, and Amit Sheth. Semantic Interoperability in Global Information Systems: a brief introduction to the research area and the special section. *SIGMOD Record*, 28(1): 5-12, 1999.
- [Pap02] Michael P. Papazoglou. The Web-Services Phenomenon: Concepts, Technologies, Current Trends, and Research Directions. Tutorial in the 14th International Conference on Advanced Information Systems Engineering (CAiSE'02), Toronto, Canada, May 2002.
- [PERL] Perl programming language, 2003. <http://www.perl.org>
- [PGU96] Yannis Papakonstantinou, Hector Garcia-Molina, and Jeffrey Ullman. MedMaker: A Mediation System Based on Declarative Specifications. In *Proceedings of the International Conference on Data Engineering*, pages 132-141, New Orleans, February 1996.
- [PGW95] Yannis Papakonstantinou, Hector Garcia-Molina, and Jennifer Widom. Object Exchange Across Heterogeneous Information Sources. In *Proceedings of the International Conference on Data Engineering*, pages 251-260, Taipei, Taiwan, 1995.

- [PW93] J. Plaice, and W. Wadge. A New Approach to Version Control. *IEEE Transactions on Software Engineering*, 19(3): 268-276, 1993.
- [RDF1] The World Wide Web Consortium (W3C). Resource Description Framework (RDF) Schema Specification, 1999. <http://www.w3.org/TR/PR-rdf-schema>
- [RDF2] The World Wide Web Consortium (W3C). Resource Description Framework (RDF) Model and Syntax Specification, 1999. <http://www.w3.org/TR/REC-rdf-syntax>
- [RS97] Mary Tork Roth, and Peter Schwarz. Don't Scrap It, Wrap It! A Wrapper Architecture for Legacy Data Sources. In *Proceedings of the 23rd International Conference on Very Large DataBases (VLDB'97)*, Athens, Greece, August 1997.
- [SA85] R. Snodgrass, and Ilsoo Ahn. A Taxonomy of Time in Databases. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 236-246, 1985.
- [SD02] S. Shang, and C. Dyreson. Adding Valid Time to XPath. In *Proceedings of Database and Network Information Systems (DNIS'02)*. Lecture Notes in Computer Science (LNCS) 2544, pages 29-42, Springer-Verlag, 2002.
- [SG02] Yannis Stavarakas, and Manolis Gergatsoulis. Multidimensional Semistructured Data: Representing Context-Dependent Information on the Web. In *Proceedings of the 14th International Conference on Advanced Information Systems Engineering (CAiSE'02)*, Toronto, Canada, May 2002. Lecture Notes in Computer Science (LNCS), Vol. 2348, pages 183-199, Springer-Verlag, 2002.
- [SGDZ02] Yannis Stavarakas, Manolis Gergatsoulis, Christos Doukeridis, and Vassilis Zafeiris. Accommodating Changes in Semistructured Databases Using Multidimensional OEM. In *Proceedings of the 6th East European Conference on Advances in Databases and Information Systems (ADBIS'02)*, Bratislava, Slovakia, September 2002. Lecture Notes in Computer Science (LNCS) 2435, pages 360-373, Springer-Verlag, 2002.
- [SGDZ03] Yannis Stavarakas, Manolis Gergatsoulis, Christos Doukeridis, and Vassilis Zafeiris. Representing and Querying Histories of Semistructured Databases Using Multidimensional OEM. To appear in *Information Systems journal*, 2003.
- [SGM00] Yannis Stavarakas, Manolis Gergatsoulis, and Theodoros Mitakos. Representing Context-Dependent Information Using Multidimensional XML. In *Proceedings of the 4th European Conference on Research and Advanced Technology for Digital Libraries (ECDL'00)*, Lisbon, September 2000. Lecture Notes in Computer Science (LNCS) 1923, pages 368-371, Springer-Verlag, 2000.
- [SGML] ISO (International Organization for Standardization). *ISO 8879: 1986 (E). Information processing – Text and Office Systems – Standard Generalized Markup Language (SGML)*. First edition, 1986.

- [SGR00] Yannis Stavarakas, Manolis Gergatsoulis, and Panos Rondogiannis. Multidimensional XML. In *Proceedings of the 3rd International Workshop on Distributed Communities on the Web (DCW'00)*, Quebec City, Canada, June 2000. Lecture Notes in Computer Science (LNCS) 1830, pages 100-109, Springer-Verlag, 2000.
- [SKV98] Yannis Stavarakas, Nikos Karagiannidis, and Panos Vasiliadis. Direct Database Access through WWW Browsers (in Greek). In *Proceedings of the Panhellenic Conference on New Information Technology*, Athens, Greece, 1998.
- [SPES03] Yannis Stavarakas, Kostis Pristouris, Antonis Efandis, and Timos Sellis. Implementing a Query Language for Context-dependent Semistructured Data. Submitted for publication, 2003.
- [SQL] International Organization for Standardization (ISO). *Information Technology – Database Language SQL*. Standard No. ISO/IEC 9075:1999.
- [SR86] Romualdas Skvarcius, and William B. Robinson. *Discrete Mathematics with Computer Science Applications*. The Benjamin / Cummings Publishing Company, 1986.
- [SSU95] Avi Silberschatz, Mike Stonebraker, and Jeff Ullman (editors). Database Research: Achievements and Opportunities Into the 21st Century. *Report of an NSF Workshop on the Future of Database Systems Research*, May 1995.
- [Suc97] Dan Suciu. Management of Semistructured Data. *SIGMOD Record*, 26(4): 4-7, December 1997.
- [Suc98] Dan Suciu. An Overview of Semistructured Data. *SIGACT News*, 29(4): 28-38, December 1998.
- [SW00] P. Swoboda, and W. W. Wadge. Vmake and ISE General Tools for the Intensionalization of Software Systems. *Intensional Programming II*, pages 310-320. World Scientific, 2000.
- [SWEB] The World Wide Web Consortium (W3C). Semantic Web, 2001. <http://www.w3.org/2001/sw/>
- [SZ96] Avi Silberschatz, and Stan Zdonik. Strategic Directions in Database Systems – Breaking Out of the Box. *ACM Computer Surveys*, 28(4): 764-778, December 1996.
- [SZ97] Avi Silberschatz, and Stan Zdonik. Database Systems – Breaking Out of the Box. *SIGMOD Record*, 26(3): 36-50, September 1997.
- [TACS98] Manos Theodorakis, Anastasia Analyti, Panos Constantopoulos, and Nikos Spyrtatos. Context in Information Bases. In *Proceedings of the 3rd International Conference on Cooperative Information Systems (CoopIS'98)*, New York City, 1998.
- [TG95] V. Tsotras, and B. Gopinath. Efficient Management of Time-Evolving Databases. *IEEE Transactions on Knowledge and Data Engineering*, 7(4): 591-607, August 1995.

- [The01] Manos Theodorakis. Contextualization: an Abstraction Mechanism for Information Modeling. PhD Thesis, University of Crete, Greece, 2001.
- [Ull88] Jeffrey D. Ullman. *Principles of Database and Knowledge-Base Systems*, Vol. 1. Computer Science Press, 1988.
- [Ull97] Jeffrey D. Ullman. Information Integration Using Logical Views. In *Proceedings of the 6th International Conference on Database Theory (ICDT'97)*, Delphi, Greece, January 1997.
- [URI] T. Berners-Lee, R. Fielding, U. C. Irvine, and L. Masinter. Uniform Resource Identifiers (URI): Generic Syntax, 1998. <http://www.ietf.org/rfc/rfc2396.txt>
- [URL] T. Berners-Lee, L. Masinter, and M. McCahill. Uniform Resource Locators (URL), 1994. <http://www.ietf.org/rfc/rfc1738.txt>
- [W3C] The World Wide Web Consortium (W3C), 2003. <http://www.w3.org/>
- [Wal97] Norman Walsb. A Guide to XML. *World Wide Web Journal "XML: Principles, Tools, and Techniques"*, 2(4): 97-107, 1997.
- [WBSY98] W. Wadge, G. Brown, M. Schraefel, and T. Yildirim. Intensional HTML. In *Proceedings of the 4th International Workshop on Principles of Digital Document Processing (PODDP'98)*, March 1998. Lecture Notes in Computer Science (LNCS) 1481, pages 128-139, Springer-Verlag, 1998.
- [Wie92] Gio Wiederhold. Mediators in the Architecture of Future Information Systems. *The IEEE Computer Magazine*, 25: 38-49, March 1992.
- [XLIN] The World Wide Web Consortium (W3C). XML Linking Language (Xlink), 1999. <http://www.w3.org/TR/xlink>
- [XML] The World Wide Web Consortium (W3C). eXtensible Markup Language (XML) 1.0 (Second edition), 2000. <http://www.w3.org/TR/2000/REC-xml-20001006>
- [XNS] The World Wide Web Consortium (W3C). Namespaces in XML, 1999. <http://www.w3.org/TR/REC-xml-names>
- [XPTH] The World Wide Web Consortium (W3C). XML Path Language (XPath), 1999. <http://www.w3.org/TR/xpath>
- [XPTR] The World Wide Web Consortium (W3C). XML Pointer Language (XPointer), 1999. <http://www.w3.org/TR/xptr>
- [XQL] The World Wide Web Consortium (W3C). XML Query Language (XQL), 1998. <http://www.w3.org/TandS/QL/QL98/pp/xql.html>
- [XQR] The World Wide Web Consortium (W3C). XML Query Requirements, 2000. <http://www.w3.org/TR/xmlquery-req>
- [XQRY] The World Wide Web Consortium (W3C). XQuery 1.0: An XML Query Language, 2002. <http://www.w3.org/TR/xquery>

- [XSC0] The World Wide Web Consortium (W3C). XML Schema Part 0: Primer, 2001. <http://www.w3.org/TR/xmlschema-0/>
- [XSC1] The World Wide Web Consortium (W3C). XML Schema Part 1: Structures, 1999. <http://www.w3.org/TR/xmlschema-1/>
- [XSC2] The World Wide Web Consortium (W3C). XML Schema Part 2: Datatypes, 1999. <http://www.w3.org/TR/xmlschema-2/>
- [XSL] The World Wide Web Consortium (W3C). eXtensible Stylesheet Language (XSL) Version 1.0, 2001. <http://www.w3.org/TR/xsl>
- [XSLT] The World Wide Web Consortium (W3C). XSL Transformations (XSLT) Version 1.0, 1999. <http://www.w3.org/TR/xslt>
- [Yil97] Taner Yildirim. Intensional HTML. Master's Thesis, Department of Computer Science, University of Victoria, Canada, 1997.
- [YL96] Budi Yuwono, and Dik Lun Lee. WISE: A World Wide Web Resource Database System. *IEEE Transactions on Knowledge and Data Engineering*, 8(4): 548-554, August 1996.
- [ZCF+97] C. Zaniolo, S. Ceri, C. Faloutsos, R. Snodgrass, V. Subrahmanian, and R. Zicari. *Advanced Database Systems*. Morgan Kaufmann Publishers, San Francisco, California, 1997.
- [ZDSG02] Vassilis Zafeiris, Christos Doukeridis, Yannis Stavarakas, and Manolis Gergatsoulis. An Infrastructure for Manipulating Multidimensional Semistructured Data. In *Proceedings of the 1st Hellenic Data Management Symposium (HDMS'02)*, Athens, Greece, July 2002.
- [ZSC99] Vagelis Zorbas, Yannis Stavarakas, and Kostas Chatzinas. Specification of a Mechanism for Direct Access to Databases through WWW. In *Proceedings of the 7th Hellenic Conference on Informatics*, Ioannina, Greece, 1999.



## ΓΛΩΣΣΑΡΙΑ / GLOSSARY

Άγκιστρα	Angle brackets < >
Αγκύλες	Curly brackets { }
Αιτιοκρατικός ως προς το περιβάλλον	Context-deterministic
Ακμή οντότητας	Entity edge
Ακμή περιβάλλοντος	Context edge
Αλγόριθμος σταθερού σημείου	Fixed-point algorithm
Αμφιεξομοίωση	Bisimulation
Αναγνωριστικό αντικειμένου	Object identifier (oid)
Αναγωγή σε OEM	Reduction to OEM
Ανάπτυξη επεκτεταμένου όρου	Clause expanded extension
Ανάπτυξη επεκτεταμένου περιβάλλοντος	Context expanded extension
Ανάπτυξη όρου	Clause extension
Απο-περιβαλλοντοποίηση	De-contextualization
Αποπεριβαλλοντοποίηση γράφου	Graph de-contextualization
Ατομικός κόμβος	Atomic node
Βάση δεδομένων γράφου	Graph database
Γενικευμένη έκφραση μονοπατιού	General path expression
Γλώσσα επισημείωσης	Mark-up language
Γλώσσα πυρήνα	Core language
Γραμματικά δεδομένα της XML	XML parsed data
Γραφικό περιβάλλον	Graphical interface
Δεδομένο-κεντρικός	Data-centric
Διαδραστικός	Interactive
Διάζευξη	Disjunction
Διακλαδιζόμενος χρόνος	Branching time
Δια-κοσμική επερώτηση	Cross-world query
Διάσταση	Dimension

Διαφυλλιστής	Browser
Διεπαφή	Interface
Διεργασία	Process
Έγκυρος χρόνος	Valid time
Εγκυρότητα	Validity
Έκφραση	Facet
Έκφραση μονοπατιού περιβάλλοντος	Context path expression
Ελάχιστο σταθερό σημείο	Least fixed point
Εμβέλεια	Scope
Ενδιάμεσο διάζωμα	Middleware tier
Εννοιολογικός	Conceptual
Ενοποίηση πληροφορίας	Information integration
Εξ ορισμού	Default
Εξαναγκασμός	Coercion
Εξυπηρετητής Ιστού	Web server
Επέκταση όρου	Clause expansion
Επεκτεταμένη μορφή	Expanded form
Επερώτηση οδηγούμενη από περιβάλλον	Context-driven query
Επιτρεπόμενος	Legal
Ερμηνευτικό περιβάλλον	Context
Ευρετήριο	Index
Ίσο-περιβάλλον	Context equal
Ιστορικό μιας βάσης δεδομένων OEM	History of an OEM database
Ιστός	Web
Ισχυρά συνδεδεμένο τμήμα	Strongly connected component
Καθολικό περιβάλλον	Universal context
Καθολικός όρος	Universal clause
Κάλυψη περιβάλλοντος	Context coverage
Κανονική μορφή	Canonical form
Κάρτα-αρχής	Start-tag
Κατηγορημα	Predicate
Κενό περιβάλλον	Empty context
Κενός όρος	Empty clause



Κληρονομούμενη κάλυψη	Inherited coverage
Κληρονομούμενη κάλυψη μονοπατιού	Path inherited coverage
Κληρονομούμενο περιβάλλον	Inherited context
Κόμβος περιβάλλοντος	Context node
Λειτουργίες βασικών αλλαγών	Basic change operations
Λέξη-κλειδί	Keyword
Μέθοδος προσπέλασης δεδομένων	Data access method
Μερική αναγωγή	Partial reduction
Μεσολαβητής	Mediator
Μέσω δικτύου	Online
Μεταβλητή περιβάλλοντος	Context variable
Μεταπληροφορία	Metadata
Μεταφραστής	Wrapper
Μονάδα	Module
Μονοπάτι δεδομένων περιβάλλοντος	Context data path
Μοντέλο γράφου	Graph model
Μορφότυπος	Format
Μπαλαντέρ περιβάλλοντος	Context wildcard
Οδηγός	Template
Ολικός	Global
Ομαλές εκφράσεις	Regular expressions
Οντότητα πολλαπλών εκφάνσεων	Multifaceted entity
Ορθά-σχηματισμένος	Well-formed
Όρος	Clause
Όρος περιβάλλοντος	Context clause
Όρος προσδιοριστή περιβάλλοντος	Context specifier clause
Παγκόσμιος Ιστός	World Wide Web
Πεδίο ορισμού, γνωστικό πεδίο	Domain
Περιβαλλοντική περικοπή γράφου	Graph context pruning
Περιβαλλοντική προβολή γράφου	Graph context projection
Περιοριστής περιβάλλοντος	Context qualifier
Πιθανοί κόσμοι	Possible worlds
Πιθανοτικός	Probabilistic

Πλατφόρμα	Framework
Πληροφοριακή οντότητα	Information entity
Πολυ-διαζωματικός	Multi-tier
Πολυδιάστατα ημιδομημένα δεδομένα	Multidimensional semistructured data
Πολυδιάστατη XML / MXML	Multidimensional XML / MXML
Πολυδιάστατη Γλώσσα Επερωτήσεων / MQL	Multidimensional Query Language / MQL
Πολυδιάστατη οντότητα	Multidimensional entity
Πολυδιάστατο DTD / MDTD	Multidimensional DTD / MDTD
Πολυδιάστατο XSL / MXSL	Multidimensional XSL / MXSL
Πολυδιάστατο γνώρισμα	Multidimensional attribute
Πολυδιάστατο OEM / MOEM	Multidimensional OEM / MOEM
Πολυδιάστατο παράδειγμα	Multidimensional paradigm
Πολυδιάστατο στοιχείο	Multidimensional element
Πολυδιάστατος Γράφος Δεδομένων	Multidimensional Data Graph
Πολυδιάστατος κόμβος	Multidimensional node
Προσδιοριστής διάστασης	Dimension specifier
Προσδιοριστής περιβάλλοντος	Context specifier
Πρότυπο περιβάλλοντος	Context pattern
Ρητό περιβάλλον	Explicit context
Ρητό περιβάλλον μονοπατιού	Path explicit context
Σάκος	Bag
Σε σχέση με	With respect to
Σημειογραφία	Notation
Στοιχείο παιδί	Child element
Στοιχείο περιβάλλοντος	Context element
Στοιχειοσειρά	String
Συζευξη	Conjunction
Συλλογιστική	Reasoning
Συναθροιστική συνάρτηση περιβάλλοντος	Context aggregate function
Σύνδεση	Join
Σύνδεσμος	Link
Σύνθετος κόμβος	Complex node
Ταυτοχρονισμός	Concurrency

Τετράγωνα άγκιστρα	Square brackets [ ]
Τμήμα έκφανσης	Facet part
Τμήμα οντότητας	Entity part
Τόπος του Ιστού	Web site
Τυποποίηση	Formalism
Υποδομή	Infrastructure
Υποκείμενος	Underlying
Υφιστάμενη έκφανση	Holding facet
Υφίσταται κάτω από	Holds under
Φύλλο διαμόρφωσης	Stylesheet
Χρονικό στιγμιότυπο OEM	Temporal OEM snapshot
Χρόνος εκτέλεσης	Run-time
Χρονοσφραγίδα	Timestamp



