ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Τεχνικές ολοκλήρωσης σημασιολογικών γεωχωρικών δεδομένων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΘΩΜΑ ΜΑΡΟΥΛΗ

Επιβλέπων : Ιωάννης Βασιλείου
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούνιος 2013

Η σελίδα αυτή είναι σκόπιμα λευκή.

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Τεχνικές ολοκλήρωσης σημασιολογικών γεωχωρικών δεδομένων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΘΩΜΑ ΜΑΡΟΥΛΗ

Επιβλέπων : Ιωάννης Βασιλείου
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 8η Ιουλίου 2013.

.....	.....	.....
Ιωάννης Βασιλείου Καθηγητής Ε.Μ.Π.	Γιώργος Στάμου Καθηγητής Ε.Μ.Π.	Θοδωρής Δαλαμάγκας Ερευνητής Β' ΙΠΣΥ/Ε.Κ. "Αθηνά"

Αθήνα, Ιούνιος 2013

.....

ΘΩΜΑΣ ΜΑΡΟΥΛΗΣ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Τα πρόσφατα χρόνια, οι τεχνολογίες και μεθοδολογίες του Σημασιολογικού Ιστού έχουν ισχυροποιήσει τη θέση τους στο πεδίο της διαχείρισης δεδομένων και γνώσης. Πρότυπα για την οργάνωση και επερώτηση σημασιολογικής πληροφορίας όπως τα RDF(S) και SPARQL έχουν υιοθετηθεί από μεγάλες ακαδημαϊκές κοινότητες, ενώ εταιρικοί πάροχοι υιοθετούν σημασιολογικές τεχνολογίες για να οργανώσουν, εκθέσουν, ανταλλάξουν και ανακτήσουν τα δεδομένα τους. Παράλληλα, οι γεωγραφικές βάσεις δεδομένων είναι μερικές από τις μεγαλύτερες υπαρκτές βάσεις και έχουν μεγάλη σημασία σε ένα εύρος καθημερινών εφαρμογών. Τέτοιου είδους δεδομένα απεικονίζονται και χειραγωγούνται με χρήση Συστημάτων Γεωγραφικής Πληροφορίας – Geographic Information Systems (GIS), όμως η ολοκλήρωση εξωτερικών σετ δεδομένων σε αυτά τα συστήματα είναι χρονοβόρα και πολύπλοκη. Σε αυτό το πλαίσιο, που συντίθεται αφενός από τις αδυναμίες των υπαρχόντων GIS συστημάτων και αφετέρου από τις καλά τεκμηριωμένες δυνατότητες και οφέλη των τεχνολογιών σημασιολογικού ιστού, μία δύσκολη όσο και ενδιαφέρουσα πρόκληση είναι η αποτελεσματική ολοκλήρωση εννοιών και τεχνολογιών από τη διαχείριση γεωχωρικών δεδομένων με τον Σημασιολογικό Ιστό.

Στο πλαίσιο της παρούσας διπλωματικής αναπτύξαμε το Geosm, ένα εργαλείο για το μετασχηματισμό δεδομένων χαρτογράφησης από το OpenStreetMap σε RDF γράφους σε συμφωνία με το OGC GeoSPARQL πρότυπο. Για την ανάπτυξη του εργαλείου βασιστήκαμε στο υπάρχον εργαλείο Osmosis και στη βιβλιοθήκη Apache Jena, ενώ για τη διατήρηση της συμβατότητας με το LinkedGeoData project κάναμε χρήση των ίδιων RDF λεξιλογίων. Αναλύσαμε το θέμα της διασύνδεσης γεωχωρικών σημασιολογικών δεδομένων, εξετάζοντας τις δυνατότητες υπαρχόντων μετρικών ομοιότητας, σχεδιάζοντας νέες μετρικές προσαρμοσμένες στις απαιτήσεις των δεδομένων μας και τέλος εκτελώντας ένα ευρύ φάσμα πειραμάτων σε δεδομένα που συλλέξαμε για την ανάλυση της επίδοσης των μετρικών.

Λέξεις Κλειδιά: RDF, GeoSPARQL, γεωχωρικά δεδομένα, μετασχηματισμοί δεδομένων, διασύνδεση

Η σελίδα αυτή είναι σκόπιμα λευκή.

Abstract

In recent years, Semantic Web technologies and methodologies have strengthened their position in the field of of data and knowledge management. Standards for organising and querying semantic information such as RDF(S) and SPARQL have been adopted by large academic communities, while corporate vendors have been adopting semantic technologies to organise, expose, exchange and retrieve their data. At the same time, geographic databases haves evolved into some of the largest databases in existence and have a significant impact in a wide range of everyday applications. The data can be mapped and sometimes manipulated with the use of Geographic Information Systems (GIS), however integrating external datasets into those systems is often time consuming and complex. In this context, which formed on the one hand by the weaknesses of existing GIS systems and on the other by the well documented capabilities and benefits of semantic web technologies, a equally difficult and interesting challenge is the effective integration of concepts and technologies from the geographic data management with the Semantic Web.

In the context of this thesis we have developed Geosm, a tool for the mapping of cartographic data from OpenStreetMap to RDF graphs in compliance with the OGC GeoSPARQL standard. For the development of the tool we made use of the existing tool Osmosis and the library Apache Jena, while to maintain compatibility with the LinkedGeoData project we made use of the same RDF vocabularies. We analysed the issue of interlinking geospatial semantic data, by exploring the capabilities of existing similarity metrics, designing new metrics adapted to the requirements of our data and finally executing a wide range of tests on data we collected for the purpose of analysing the metric quality.

Keywords: RDF, GeoSPARQL, geospatial data, data transformations, interlinking

Η σελίδα αυτή είναι σκόπιμα λευκή.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον κ. Γιώργο Γιαννόπουλο, διδακτορικό υπότροφο στο ΙΠΣΥ / Ε.Κ. “Αθηνά” για την καθοδήγηση και βοήθεια του κατά την εκπόνηση της διπλωματικής. Επίσης θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Ιωάννη Βασιλείου για την δυνατότητα που μου προσέφερε να ασχοληθώ με ένα πολύ ενδιαφέρον θέμα.

Ευχαριστώ ιδιαίτερα τους γονείς μου για την αδιάλειπτη στήριξή τους κατά τη διάρκεια των σπουδών μου.

Η σελίδα αυτή είναι σκόπιμα λευκή.

Πίνακας Περιεχομένων

1

Εισαγωγή

1.1 Ολοκλήρωση σημασιολογικών γεωχωρικών δεδομένων

Τα πρόσφατα χρόνια, οι τεχνολογίες και μεθοδολογίες του Σημασιολογικού Ιστού έχουν ισχυροποιήσει τη θέση τους στο πεδίο της διαχείρισης δεδομένων και γνώσης. Πρότυπα για την οργάνωση και επερώτηση σημασιολογικής πληροφορίας όπως τα RDF(S) και SPARQL έχουν υιοθετηθεί από μεγάλες ακαδημαϊκές κοινότητες, ενώ εταιρικοί πάροχοι υιοθετούν σημασιολογικές τεχνολογίες για να οργανώσουν, εκθέσουν, ανταλλάξουν και ανακτήσουν τα δεδομένα τους. Οι αποθήκες RDF έχουν γίνει αρκετά εύρωστες ώστε να υποστηρίζουν δεδομένα με όγκο που φτάνουν τα δισεκατομμύρια εγγραφές (RDF triples), ενώ παρέχουν δυνατότητες διαχείρισης δεδομένων και λειτουργίες επερωτήσεων παρόμοιες με αυτές των παραδοσιακών σχεσιακών συστημάτων βάσεων δεδομένων (RDBMS).

Γεωχωρικά δεδομένα ή γεωγραφική πληροφορία είναι εκείνα τα δεδομένα που αναγνωρίζουν τη γεωγραφική τοποθεσία ενός φυσικού ή τεχνητού χαρακτηριστικού καθώς και συνόρων στην επιφάνεια της Γης. Οι γεωγραφικές βάσεις δεδομένων είναι μερικές από τις μεγαλύτερες υπαρκτές βάσεις και έχουν μεγάλη σημασία σε ένα εύρος καθημερινών εφαρμογών. Τέτοιου είδους δεδομένα απεικονίζονται και χειραγωγούνται με χρήση Συστημάτων Γεωγραφικής Πληροφορίας – Geographic Information Systems (GIS), όμως η ολοκλήρωση εξωτερικών σετ δεδομένων σε αυτά τα συστήματα είναι χρονοβόρα και πολύπλοκη. Σε αυτό το πλαίσιο, που συντίθεται αφενός από τις αδυναμίες των υπαρχόντων GIS συστημάτων και αφετέρου από τις καλά τεκμηριωμένες δυνατότητες και οφέλη των τεχνολογιών σημασιολογικού ιστού, μία δύσκολη όσο και ενδιαφέρουσα πρόκληση είναι η

αποτελεσματική ολοκλήρωση εννοιών και τεχνολογιών από τη διαχείριση γεωχωρικών δεδομένων με τον Σημασιολογικό Ιστό.

Η ολοκλήρωση του Σημασιολογικού Ιστού με τη διαχείριση γεωχωρικής πληροφορίας απαιτεί από την επιστημονική κοινότητα να αντιμετωπίσει δύο προκλήσεις:

i. Τον ορισμό κατάλληλων προτύπων και λεξιλογίων που περιγράφουν γεωχωρική πληροφορία σύμφωνα με τα πρωτόκολλα RDF(S) και SPARQL αφενός και αφετέρου σε συμφωνία με τις αρχές των εδραιωμένων γεωχωρικών προτύπων, όπως το πρότυπα OGC, GML, INSPIRE,
ii. Την ανάπτυξη τεχνολογιών για την αποδοτική αποθήκευση και επερώτηση σε σημασιολογικά γεωχωρικά δεδομένα.

Όσον αφορά την πρώτη πρόκληση έχουν υπάρξει αρκετές προσπάθειες από την κοινότητα του Σημασιολογικού Ιστού για την εδραίωση ενός γεωχωρικού RDF προτύπου, όπως το Basic Geo Vocabulary [GeoPos84] της W3C που επιτρέπει την αναπαράσταση σημείων σε WGS84, το GEORSS που παρείχε υποστήριξη για περισσότερα γεωχωρικά αντικείμενα (γραμμές, παραλληλόγραμμα, πολύγωνα) και το GeoOWL που αναπτύχθηκε για να παράσχει ένα περισσότερο ευέλικτο μοντέλο για γεωχωρικές έννοιες.

Πρόσφατα, το πρότυπο OGC GeoSPARQL [OGC12] έχει αναδειχθεί ως ένα πολλά υποσχόμενο πρότυπο για γεωχωρικό RDF με το στόχο να τυποποιήσει την εισαγωγή και επερώτηση γεωχωρικών RDF δεδομένων. Το GeoSPARQL παρέχει διάφορες κλάσεις συμβατότητες όσον αφορά την υλοποίηση δυνατοτήτων προχωρημένης συλλογιστικής (π.χ., ποσοτική συλλογιστική), καθώς και πολλά σετ ορολογίας για την περιγραφή τοπολογικών σχέσεων μεταξύ γεωμετριών. Επομένως, υποστηρίζονται διαφορετικές υλοποιήσεις της προδιαγραφής GeoSPARQL, σε αναλογία με το αντίστοιχο πεδίο/εφαρμογή. Επιπλέον, το GeoSPARQL ακολουθεί στενά τα υπάρχοντα πρότυπα της OGC για γεωχωρικά δεδομένα με σκοπό τη διευκόλυνση της χωρικής ευρετηρίασης από σχεσιακές βάσεις.

Σε αντίθεση με την προτυποποίηση, έχει υπάρξει πολύ λιγότερη πρόοδος όσον αφορά τη δεύτερη πρόκληση. Παρότι υπάρχει μεγάλος αριθμός αποθηκών δεδομένων RDF που μπορούν να υποστηρίξουν μεγάλους όγκους RDF δεδομένων, μόνο μερικά από αυτά, όπως τα Virtuoso, OWLIM, Parliament, uSeekM, Oracle Spatial υποστηρίζουν γεωχωρικά RDF δεδομένα. Επιπλέον, ακόμη λιγότερα υποστηρίζουν πλήρως όλα τα τυποποιημένα γεωχωρικά χαρακτηριστικά ή είναι πλήρως συμβατά με το GeoSPARQL πρότυπο. Ένα άλλο πρόβλημα είναι η αποδοτικότητα των υπαρχόντων συστημάτων διαχείρισης γεωχωρικής RDF πληροφορίας, η οποία δεν μπορεί να συγκριθεί ακόμη σε όρους επιδόσεων με τυπικά χωρικά RDBMS.

1.2 Αντικείμενο διπλωματικής

Στο πλαίσιο αυτής της διπλωματικής εξετάσαμε και προσπαθήσαμε να δώσουμε λύσεις σε δύο από τις προκλήσεις που συνθέτουν τον ευρύ στόχο που παρατίθεται ανωτέρω. Στην πρώτη φάση της διπλωματικής αναπτύχθηκε η εφαρμογή Geosm για το μετασχηματισμό δεδομένων από το OpenStreetMap σε GeoSPARQL compliant RDF. Το OpenStreeetMap ως μία από τις μεγαλύτερες πηγές ελεύθερα διαθέσιμων γεωχωρικών δεδομένων μπορεί να αποτελέσει πολύτιμη πηγή γεωχωρικής πληροφορίας για ερευνητικούς σκοπούς. Η Geosm στοχεύει να είναι μια εξίσου αποδοτική και φιλική προς το χρήστη εφαρμογή η οποία θα επιτρέπει σε οποιονδήποτε ερευνητή ή άλλο χρήστη να αποκτήσει εύκολη πρόσβαση στα δεδομένα του OSM σε συμφωνία με το GeoSPARQL πρότυπο και τις απαιτήσεις της εφαρμογής ή του πεδίου τους.

Στη δεύτερη φάση της διπλωματικής διερευνήσαμε τις δυνατότητες ανάπτυξης συνδυαστικών μετρικών ομοιότητας για το σκοπό της διασύνδεσης (data interlinking) οντοτήτων από σετ δεδομένων με γεωχωρική πληροφορία προερχόμενων από ξένες μεταξύ τους πηγές. Για το σκοπό αυτό συλλέξαμε ένα μεγάλο εύρος δεδομένων για σημεία ενδιαφέροντος (POIs) το οποίο μετατρέψαμε σε GeoSPARQL RDF γράφους. Στη συνέχεια εξετάσαμε τις δυνατότητες υπαρχόντων μετρικών διασύνδεσης σε συμβολοσειρές και γεωμετρίες και τέλος σχεδιάσαμε νέες μετρικές οι οποίες κάνουν συνδυαστική χρήστη συμβολοσειρών και γεωμετριών με στόχο τη βελτίωση της ποιότητας των παρεχόμενων συνδέσεων.

1.3 Οργάνωση κειμένου

Ο τόμος αποτελείται από 7 κεφάλαια που καλύπτουν πλήρως την ανάπτυξη της διπλωματικής καθώς και το απαιτούμενο θεωρητικό υπόβαθρο και σχετικές τεχνολογίες.

Στο **κεφάλαιο 2** παρουσιάζονται τα πρότυπα OGC GeoSPARQL και ΕΛΟΤ743, γίνεται αναφορά και περιγραφή των πηγών που χρησιμοποιήθηκαν για τη συλλογή γεωχωρικών δεδομένων και τέλος παρουσιάζονται όλες οι βιβλιοθήκες και εργαλεία που είτε έγινε άμεση χρήση τους είτε είναι σχετικά με το αντικείμενο.

Στο **κεφάλαιο 3** παρουσιάζεται πλήρως η εφαρμογή Geosm, ξεκινώντας από την προδιαγραφή απαιτήσεων και στη συνέχεια παρουσιάζοντας τη δομή και σχεδίαση της εφαρμογής, εμβαθύνοντας σε τεχνικές λεπτομέρειες της υλοποίησης όπου αυτές κρίνονται ιδιαίτερου ενδιαφέροντος. Τέλος περιγράφεται η διαδικασία ελέγχου της εφαρμογής και δίνεται το εγχειρίδιο χρήσης.

Στο **κεφαλαιο 4** παρουσιάζεται η έρευνα πάνω στις μετρικές διασύνδεσης. Παρουσιάζονται οι υπάρχουσες μετρικές που αξιοποιήθηκαν, εξηγείται η λειτουργικότητα και σκοπός των νέων μετρικών που αναπτύχθηκαν και τέλος παρουσιάζεται η πειραματική ανάλυση των μετρικών.

Στο **κεφάλαιο 5** παρουσιάζονται συγκεντρωτικά τα συμπεράσματα για το σύνολο της διπλωματικής και προτείνονται πιθανές μελλοντικές επεκτάσεις.

Στο **κεφάλαιο 6** παρουσιάζονται τα namespaces που χρησιμοποιούνται στο πλαίσιο της διπλωματικής και δίνονται τα αντίστοιχα προθέματα αυτών.

Στο **κεφάλαιο 7** δίνεται η βιβλιογραφία και οι πηγές που αξιοποιήθηκαν στο πλαίσιο της διπλωματικής.

2

Πρότυπα, Δεδομένα και Τεχνολογίες

2.1 Πρότυπα

Στα ακόλουθα τμήματα παρουσιάζουμε τα πρότυπα στα οποία έχει βασιστεί αυτή η διπλωματική.

2.1.1 OGC GeoSPARQL

Το πρόσφατο πρότυπο OGC GeoSPARQL [OGC12] υποστηρίζει αναπαράσταση γεωχωρικών δεδομένων και επερωτήσεις πάνω σε αυτά στον σημασιολογικό ιστό. Ορίζει ένα λεξιλόγιο για αναπαράσταση γεωχωρικής πληροφορίας σε RDF και ορίζει μία επέκταση της γλώσσας SPARQL για επεξεργασία γεωχωρικών δεδομένων.

Στα τμήματα που ακολουθούν θα εξετάσουμε μερικές βασικές έννοιες σχετικές με τις τυποποιήσεις του σημασιολογικού ιστού καθώς και το W3C Basic Geo Vocabulary, μία πρότερη προσέγγιση στην αναπαράσταση γεωχωρικής πληροφορίας σε σημασιολογικές αποθήκες δεδομένων. Στη συνέχεια θα παρουσιάσουμε πρότυπα αναπαράστασης γεωμετρικής πληροφορίας και τέλος τις βασικές έννοιες και αρχές του GeoSPARQL.

2.1.1.1 Βασικά πρότυπα σημασιολογικού ιστού

Στα ακόλουθα τμήματα δίνουμε μία σύντομη εισαγωγή των προτύπων του σημασιολογικού ιστού για οργάνωση και επερωτήσεις πάνω σε σημασιολογική πληροφορία.

2.1.1.1.1 RDF

Το Resource Description Framework (RDF) [RDF] είναι μία γλώσσα για την αναπαράσταση πληροφορίας σχετικά με πόρους στο διαδίκτυο. Πρόθεση του RDF είναι η οργάνωση πληροφορίας σε τυποποίηση αναγνώσιμη από μηχανή, παρέχοντας ένα κοινό πλαίσιο για την έκφραση πληροφορίας ώστε να μπορεί να ανταλλαχθεί μεταξύ εφαρμογών χωρίς απώλεια νοήματος.

Το RDF βασίζεται στην ιδέα της αναγνώρισης οντοτήτων χρησιμοποιώντας αναγνωριστικά (που ονομάζονται Internationalized Resource Identifiers, ή IRIs), και της περιγραφής πόρων (οντοτήτων) με όρους απλών ιδιοτήτων και τιμών για αυτές τις ιδιότητες. Αυτό επιτρέπει στο RDF να αναπαριστά απλές προτάσεις (RDF Statements) σχετικά με οντότητες ως ένα γράφο κόμβων και ακμών που αναπαριστά τις οντότητες, τις ιδιότητές τους και τις τιμές αυτών των ιδιοτήτων. Οι RDF προτάσεις είναι τρίπλες (υποκείμενο, κατηγόρημα, αντικείμενο) που αποτελούνται από την οντότητα που αναπαριστάται (υποκείμενο), μία ιδιότητα (κατηγόρημα) και την τιμή της ιδιότητας (αντικείμενο). Συγκεκριμένα, το υποκείμενο μπορεί να είναι ένα IRI ή ένας ανώνυμος (κενός) κόμβος. Το κατηγόρημα πρέπει να είναι ένα IRI και το αντικείμενο μπορεί να είναι ένα IRI, ένας ανώνυμος (κενός) κόμβος ή ένα RDF literals. Τα literals έχουν τύπους δεδομένων που προσδιορίζουν το εύρος των πιθανών τιμών, όπως συμβολοσειρές, αριθμοί ή ημερομηνίες. Μία συλλογή RDF προτάσεων (RDF τρίπλες) μπορούν να περιγραφούν διαισθητικά ως ένας κατευθυνόμενος επισημασμένος γράφος, στον οποίον οι πόροι αναπαριστώνται ως κόμβοι και οι προτάσεις είναι ακμές (από το υποκείμενο στο αντικείμενο) που συνδέουν τους κόμβους. Τέλος, ένα σετ RDF προτάσεων καλείται RDF Dataset ή RDF Graph.

2.1.1.1.2 RDF Schema

Το RDF παρέχει ένα τρόπο να αναπαρασταθούν απλές προτάσεις σχετικά με πόρους χρησιμοποιώντας ιδιότητες για τιμές. Όμως, οι κοινότητες χρηστών RDF χρειάζονται επίσης τη δυνατότητα να ορίσουν τα λεξιλόγια (όρους) που πρόκειται να χρησιμοποιήσουν σε αυτές τις προτάσεις, συγκεκριμένα, για να υποδείξουν ότι περιγράφουν συγκεκριμένα είδη ή κλάσεις των πόρων. Το RDF καθεαυτό δεν παρέχει κάποιο τρόπο για να οριστούν τέτοιου είδους συγκεκριμένες για την εφαρμογή κλάσεις και ιδιότητες. Αντίθετα, τέτοιες κλάσεις και ιδιότητες περιγράφονται σαν ένα RDF λεξιλόγιο, χρησιμοποιώντας επεκτάσεις που παρέχονται από το RDF Schema [RDFS].

Το RDF Schema (RDFS) [RDFS] είναι μία επέκταση του RDF σχεδιασμένη να περιγράφει, χρησιμοποιώντας ένα σετ δεσμευμένων λέξεων που καλούνται το RDFS λεξιλόγιο, πόρους ή/και σχέσεις μεταξύ πόρων. Παρέχει κατασκευές για την περιγραφή των τύπων των αντικειμένων (κλάσεις), ιεραρχίες τύπων (υποκλάσεις), κατηγορήματα που περιγράφουν χαρακτηριστικά αντικειμένων (κατηγορήματα) και ιεραρχίες κατηγορημάτων (υποκατηγορήματα).

Το λεξιλόγιο που χρησιμοποιείται στο RDF Schema είναι ένα εξειδικευμένο σετ προκαθορισμένων RDF πόρων με ειδικά νοήματα. Οι πόροι στο λεξιλόγιο του RDF Schema έχουν URIrefs με το πρόθεμα http://www.w3.org/2000/01/rdf-schema# (κατά σύμβαση αυτό το πρόθεμα συσχετίζεται με το uname πρόθεμα rdfs:). Περιγραφές λεξιλογίων (schemas) γραμμένες στη γλώσσα του RDF Schema είναι έγκυροι RDF γράφοι. Επομένως, λογισμικό RDF που δεν είναι γραμμένο για να επεξεργάζεται το επιπλέον λεξιλόγιο του RDF Schema μπορεί να ερμηνεύει ένα schema ως έναν έγκυρο RDF γράφο αποτελούμενο από διάφορους πόρους και ιδιότητες, αλλά δεν θα μπορεί να εκμεταλλεύεται το επιπλέον νόημα εγγενές στους όρους του RDF Schema.

Μία κλάση στο RDFS αντιστοιχεί στη γενική έννοια ενός τύπου ή μιας κατηγορίας και ορίζεται χρησιμοποιώντας το κατασκεύασμα *rdfs:Class*. Πόροι που ανήκουν σε μία κλάση αποκαλούνται στιγμιότυπα αυτής της κλάσης. Ένα στιγμιότυπο μιας κλάσης είναι ένας πόρος που έχει την ιδιότητα *rdf:type* της οποίας η τιμή είναι η συγκεκριμένη κλάση. Ένας πόρος μπορεί να είναι στιγμιότυπο περισσότερων των μία κλάσεων. Κλάσεις μπορούν να είναι οργανωμένες σε μία ιεραρχία χρησιμοποιώντας το κατασκεύασμα *rdf:subClassOf*. Μία ιδιότητα στο RDFS χρησιμοποιείται για να χαρακτηρίσει στιγμιότυπα μιας κλάσης ή ενός σετ κλάσεων και ορίζεται χρησιμοποιώντας το κατασκεύασμα *rdf:Property*. Το κατασκεύασμα *rdfs:domain* χρησιμοποιείται για να δείξει ότι μία συγκεκριμένη ιδιότητα μπορεί να εφαρμοστεί στην ορισμένη κλάση και το κατασκεύασμα *rdfs:range* χρησιμοποιείται για να δείξει τις πιθανές τιμές μιας ιδιότητας. Παρόμοια με τις ιεραρχίες κλάσεων, το RDFS παρέχει το κατασκεύασμα *rdfs:subPropertyOf* για να ορίσει ιεραρχίες ιδιοτήτων.

2.1.1.1.3 OWL

Η Web Ontology Language (OWL) [OWL] είναι η πρότυπη γλώσσα για τον ορισμό Web οντολογιών. Οι OWL και RDFS έχουν αρκετές ομοιότητες. Η OWL ορίζεται ως ένα λεξιλόγιο όπως το RDF, όμως η OWL έχει πλουσιότερη σημασιολογία.

Μία κλάση OWL ορίζεται χρησιμοποιώντας το κατασκεύασμα owl:Class και αντιπροσωπεύει ένα σετ ατόμων με κοινές ιδιότητες. Όλες οι κλάσεις OWL θεωρούνται υποκλάσεις της κλάσης *owl:Thing* και υπερκλάσεις της κλάσης *owl:Nothing*. Επιπλέον, η OWL παρέχει επιπλέον κατασκευαστές για τον ορισμό κλάσεων, συμπεριλαμβανομένων των βασικών τελεστών σε σετ, ένωση, τομή και συμπλήρωμα που υλοποιούνται αντίστοιχα από τα κατασκευάσματα *owl:unionOf*, *owl:intersectionOf* και *owl:complementOf* και αρκετούς άλλους κατασκευαστές όπως, για παράδειγμα, *owl:oneOf*, *owl:equivalentClass*, etc. Σχετικά με τα άτομα, η OWL επιτρέπει σε δύο άτομα να οριστούν ως ίδια ή διαφορετικά μέσω των κατασκευασμάτων *owl:sameAs* και *owl:differentFrom* αντίστοιχα. Σε αντίθεση με το RDF Schema, η OWL διαχωρίζει μία ιδιότητα της οποίας η εμβέλεια είναι ένας τύπος δεδομένων από μία της οποίας η εμβέλεια είναι ένα σετ πόρων. Τοιουτοτρόπως, οι ιδιότητες τύπων δεδομένων OWL είναι σχέσεις ανάμεσα σε άτομα κλάσεων και τύπων δεδομένων του XML schema και ορίζονται χρησιμοποιώντας το κατασκεύασμα

owl:DatatypeProperty. Οι ιδιότητες αντικειμένων OWL είναι σχέσεις μεταξύ ατόμων κλάσεων και ορίζονται χρησιμοποιώντας το κατασκεύασμα *owl:ObjectProperty*. Τέλος, δύο ιδιότητες μπορεί να δηλωθούν ως ισοδύναμες χρησιμοποιώντας το κατασκεύασμα *owl:equivalentProperty*.

2.1.1.1.4 SPARQL 1.0

Η SPARQL Protocol and RDF Query Language (SPARQL) [SPARQL1] είναι μία σύσταση της W3C και είναι σήμερα η de facto πρότυπη γλώσσα για δεδομένα RDF. Η αποτίμηση των επερωτήσεων SPARQL βασίζεται σε αναγνώριση προτύπων σε γράφο. Ένα Graph Pattern (GP) ορίζεται αναδρομικά και περιέχει πρότυπα τριπλετών (triple patterns) και τελεστές SPARQL. Οι τελεστές της SPARQL άλγεβρας που μπορούν να εφαρμοστούν σε graph patterns είναι: AND (σύζευξη), UNION (διάζευξη), OPTIONAL (επιλεκτικά πρότυπα, όπως left outer join) και FILTER (περιορισμός). Τα πρότυπα τριπλετών είναι κανονικές τριπλέτες RDF με την εξαίρεση ότι οποιαδήποτε των υποκείμενο, κατηγόρημα, αντικείμενο μπορεί να είναι μεταβλητές. Μία αλληλουχία συνδετικών προτύπων τριπλετών καλείται Basic Graph Pattern (BGP). Η συνθήκη SPARQL Where αποτελείται από ένα GP. Η συνολική δομή της γλώσσας προσομοιάζει την SQL με τρία βασικά μπλοκ:

- Μία συνθήκη WHERE, που αποτελείται από ένα graph pattern. Μιλώντας μη τυπικά, αυτή η συνθήκη δίνεται από ένα πρότυπο που αντιστοιχεί σε έναν γράφο RDF όπου κάποιοι πόροι έχουν αντικατασταθεί από μεταβλητές. Επίσης επιτρεπτές είναι περισσότερο πολύπλοκες εκφράσεις (πρότυπα) που ορίζονται χρησιμοποιώντας κάποιους αλγεβρικούς τελεστές. Αυτό το πρότυπο χρησιμοποιείται ως φίλτρο για τις τιμές του σετ δεδομένων που θα επιστραφεί.
- Μία συνθήκη FROM, που προσδιορίζει τις πηγές των σετ δεδομένων που θα επιστραφούν.
- Μία συνθήκη SELECT, που προσδιορίζει την τελική μορφή στην οποία τα αποτελέσματα επιστρέφονται στον χρήστη. Η SPARQL, σε αντίθεση με την SQL, επιτρέπει πολλαπλές μορφές στις οποίες μπορούν να επιστραφούν τα δεδομένα: ως έναν πίνακα χρησιμοποιώντας SLECT, ως ένα γράφο χρησιμοποιώντας DESCRIBE ή CONSTRUCT ή ως μία απάντηση TRUE/FALSE χρησιμοποιώντας ASK.

Η SPARQL επιτρέπει τέσσερις μορφές επερωτήσεων: Select, Ask, Constuct και Describe. Η επερώτηση Select επιστρέφει μία ακολουθία λύσης, δηλαδή μία ακολουθία μεταβλητών και των συσχετίσεων τους. Η επερώτηση Ask επιστρέφει μία Boolean τιμή (yes or no), που δείχνει αν μία επερώτηση προτύπου ικανοποιείται ή όχι. Η επερώτηση Construct επιστρέφει έναν RDF γράφο δομημένο σύμφωνα με το σχεδιότυπο γράφου της επερώτησης. Τέλος, η επερώτηση Describe επιστρέφει έναν RDF γράφο που παρέχει μία “περιγραφή” των πόρων που ικανοποιούν τις συνθήκες. Τοιουτοτρόπως, με βάση αυτές τις μορφές επερωτήσεων, οι απαντήσεις μίας επερώτησης SPARQL μπορούν να είναι RDF γράφοι, SPARQL ακολουθίες λύσεων και Boolean τιμές.

Ας θεωρήσουμε την ακόλουθη επερώτηση: “Δώσε μου το όνομα και το γραμματοκιβώτιο για κάθε άτομο που έχει γραμματοκιβώτιο με τομέα '.cl'”. Αυτή η επερώτηση μπορεί να γραφτεί σε SPARQL ως εξής:

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX ex: <http://example.com/ns#>
SELECT ?name ?mbox
FROM <myDataSource.rdf>
WHERE {
      ?x foaf:name ?name .
      ?x foaf:mbox ?mbox .
      ?mbox ex:doman “.cl”
}
```

Οι πρώτες δύο γραμμές σε αυτό το παράδειγμα συνθέτουν τον πρόλογο της επερώτησης, ο οποίος προσδιορίζει τα namespaces που θα χρησιμοποιηθούν. Σε αυτή την περίπτωση, το ένα είναι η γνωστή οντολογία FOAF και το άλλο είναι ένα παραδειγματικό namespace. Οι λέξεις κλειδιά foaf και ex είναι συντομεύσεις για τα namespaces που χρησιμοποιούνται στο κυρίως σώμα της επερώτησης. Η λέξη κλειδί SELECT δείχνει ότι η επερώτηση επιστρέφει έναν πίνακα με δύο στήλες που αντιστοιχούν στις τιμές που αποκτώνται από το ταίριασμα των μεταβλητών ?name και ?mbox στον γράφο που δείχνει η συνθήκη FROM (myDataSource.rdf) και σύμφωνα με το πρότυπο που ορίζεται από την συνθήκη WHERE. Αξίζει να επισημανθεί ότι μία συμβολοσειρά που αρχίζει με το σύμβολο ? υποδηλώνει μία μεταβλητή στην SPARQL. Στην ανωτέρω επερώτηση, η συνθήκη WHERE συντίθεται από ένα πρότυπο με τρεις τρίπλες: *?x foaf:name ?name*, *?x foaf:mbox ?mbox* και *?mbox ex:domain “.cl”*, όπου το “.cl” είναι ένα literal. Αυτό το πρότυπο υποδηλώνει ότι ψάχνουμε για τα στοιχεία ?x, ?name και ?mbox στον RDF γράφο myDataSource.rdf έτσι ώστε το *foaf:name* του ?x να είναι το ?name, το *foaf:mbox* του ?x να είναι το ?mbox και το *ex:domain* του ?mbox να είναι το “.cl”. Τοιουτοτρόπως, μία έκφραση στη μορφή {A . B} στη SPARQL υποδηλώνει τη σύζευξη των A και B, καθώς αυτή η έκφραση ισχύει αν και τα δύο εκ των A, B ισχύουν.

Η SPARQL παρέχει διάφορους τροποποιητές ακολουθιών λύσεων που μπορούν να εφαρμοστούν στην αρχική ακολουθία λύσης ώστε να δημιουργήσουν μία διαφορετική ακολουθία σύμφωνα με τις επιθυμίες του χρήστη. Οι υποστηριζόμενοι τροποποιητές ακολουθιών λύσεων της SPARQL είναι: Distinct, Reduced, Limit, Offset και Order By. Ενώ ο τροποποιητής Distinct διασφαλίζει ότι διπλότυπες λύσεις θα διαγραφούν από το σετ της απάντησης, ο τροποποιητής Reduced επιτρέπει να διαγραφούν χωρίς όμως να το επιβάλει. Ο τροποποιητής Limit θέτει ένα άνω όριο στον αριθμό των λύσεων που επιστρέφονται. Περαιτέρω, ο τροποποιητής Offset επιστρέφει τις λύσεις που ξεκινούν μετά από τον ορισμένο αριθμό λύσεων. Τέλος, ο τροποποιητής Order By καθορίζει τη σειρά της ακολουθίας λύσεων.

2.1.1.1.5 SPARQL1.1

Η SPARQL 1.1 [SPARQL2] είναι το αποτέλεσμα της ομάδας εργασίας W3C SPARQL πάνω στην επέκταση της γλώσσας επερωτήσεων SPARQL. Η SPARQL 1.1 περιέχει τις ακόλουθες συνιστώσες: SPARQL 1.1 Query, Update, Protocol, Service Description, Property Paths, Entailment Regimes, Uniform HTTP Protocol for Managing RDF Graphs και Federation Extensions.

Η SPARQL 1.1 προσπαθεί να εξαλείψει τους βασικούς περιορισμούς της SPARQL 1.0 έκδοσης όπως συναθροιστικές λειτουργίες, φωλιασμένες επερωτήσεις, πράξεις ενημέρωσης, μονοπάτια και διάφορα άλλα ζητήματα. Περαιτέρω, η SPARQL 1.1 υποστηρίζει τις ακόλουθες συναθροιστικές λειτουργίες: COUNT, SUM, MIN/MAX, AVG, GROUP_CONCAT και SAMPLE. Επιπλέον, φωλιασμένες επερωτήσεις που είναι πολύ σημαντικές σε περιπτώσεις όπου το αποτέλεσμα μιας επερώτησης χρησιμοποιείται ως είσοδος σε άλλη επερώτηση υποστηρίζονται επίσης. Εκτός τούτου, με σκοπό να υλοποιηθεί η άρνηση, η SPARQL 1.1 έχει υιοθετήσει τους νέους τελεστές NOT EXISTS και MINUS.

Στην έκδοση SPARQL 1.0 μία επερώτηση SELECT μπορεί να προβάλει (project) μόνο μεταβλητές. Η SPARQL 1.1 επιτρέπει σε επερωτήσεις SELECT να προβάλουν οποιαδήποτε SPARQL έκφραση. Χρησιμοποιώντας τη λέξη κλειδί AS στην συνθήκη SELECT, το αποτέλεσμα μίας SPARQL έκφρασης συσχετίζεται με τη νέα μεταβλητή που προσδιορίζεται από την AS και αυτή η νέα μεταβλητή προβάλλεται. Σε αρκετές περιπτώσεις, για να μπορεί να βρεθεί ένα κόμβος, απαιτείται να εκφραστούν επερωτήσεις που χρησιμοποιούν σταθερού μήκους μονοπάτια στη διάσχιση του RDF γράφου. Η συνιστώσα SPARQL 1.1 Property Paths επεκτείνει τα υπάρχοντα βασικά πρότυπα γράφων με σκοπό να υποστηρίξει την έκφραση προτύπων διαδρομών (path patterns).

Η SPARQL 1.0 μπορεί να χρησιμοποιηθεί μόνο ως γλώσσα επερωτήσεων για ανάκτηση, αφού δεν υποστηρίζει τελεστές διαχείρισης δεδομένων. Η συνιστώσα SPARQL Update 1.1 περιλαμβάνει αρκετά χαρακτηριστικά για διαχείριση γράφων. Γίνεται εκμετάλλευση των τελεστών INSERT και INSERT DATA για να εισαχθούν νέες τρίπλες σε RDF γράφους. Επιπλέον, οι τελεστές DELECT και DELETE DATA χρησιμοποιούνται για να διαγραφούν τρίπλες από έναν RDF γράφο. Οι τελεστές LOAD και CLEAR εκτελούν μια ομάδα από διεργασίες ενημέρωσης ως μία μόνο ενέργεια. Συγκεκριμένα, ο τελεστής LOAD αντιγράφει όλες τις τρίπλες ενός γράφου στον γράφο στόχο, ενώ ο τελεστής CLEAR διαγράφει όλες τις προτάσεις από τον δοσμένο γράφο. Τέλος, για να δημιουργηθεί ένας νέος RDF γράφος σε μία αποθήκη RDF γράφων ή για να διαγραφεί ένας RDF γράφος από μία αποθήκη RDF γράφων έχουν εισαχθεί οι τελεστές CREATE και DROP.

2.1.1.2 Πρότερη εργασία σε Γεωχωρικό RDF – W3C Basic Geo Vocabulary

Το W3C Basic Geo Vocabulary [GeoPos84] παρέχει ένα namespace για την αναπαράσταση γεωγραφικού πλάτους (lat), γεωγραφικού μήκους (long) και άλλες πληροφορίες σχετικές με χωρικά αντικείμενα, χρησιμοποιώντας το WGS84 ως τυπική έδρα αναφοράς.

Το W3C Basic Geo λεξιλόγιο διερεύνησε τις δυνατότητες αναπαράστασης χαρτογραφικών/χωρικών δεδομένων σε RDF, οπότε δεν σχεδιάστηκε με πρόθεση να επιλύσει όλα τα θέματα που καλύπτονται από το OGC. Αντίθετα, σχεδιάστηκε για να παρέχει μόνο μερικούς βασικούς όρους που να μπορούν να χρησιμοποιηθούν σε RDF (π.χ., RSS 1.0 ή έγγραφα FOAF) ώστε να περιγράφει γεωγραφικά πλάτη και μήκη. Κίνητρο πίσω από τη χρήση του RDF ως φορέα για lat/long πληροφορία είναι οι δυνατότητες του RDF για διατομεακό (cross-domain) συγκερασμό δεδομένων. Ως εκ τούτου, οποιαδήποτε σχετικά RDF λεξιλόγια θα μπορούσαν να χρησιμοποιηθούν, χωρίς την ανάγκη για ακριβό προ συντονισμό ή για αλλαγές σε ένα κεντρικά διατηρημένο schema. Αυτός ο σχεδιασμός επιτρέπει ώστε βασική πληροφορία για σημεία να περιγράφεται σε RDF/XML και να αυξάνεται με περισσότερο εξελιγμένη ή συγκεκριμένη για την εφαρμογή μεταπληροφορία.

Ένα μινιμαλιστικό RDF λεξιλόγιο (namespace pos:) ορίζεται μόνο για να περιγράφει σημεία με ιδιότητες γεωγραφικού πλάτους, μήκους και υψόμετρου. Ορίζει μία κλάση *pos:Point,* τις οποίας τα μέλη είναι επίσης Points. Σημεία μπορούν να περιγραφούν χρησιμοποιώντας τις ιδιότητες *pos:lat, pos:long* και *pos:alt,* καθώς και άλλες RDF ιδιότητες που ορίζονται αλλού. Οι ιδιότητες *pos:lat* και *pos:long* δέχονται literals σε δεκαδικές μοίρες και η ιδιότητα *pos:alt* εκφράζεται σε δεκαδικά μέτρα σε σχέση με το τοπικό ελλειψοειδές αναφοράς. Το λεξιλόγιο έχει δει σημαντική χρήση και μέσα σε RDF έγγραφα, αλλά και ως namespace μέσα σε μη-RDF XML έγγραφα, όπως το RSS 2.0.

2.1.1.3 Γεωμετρικές αναπαραστάσεις

Στο παρών κεφάλαιο παρουσιάζουμε δύο μορφότυπους από πρότυπα της OGC για την αναπαράσταση γεωγραφικών literals.

2.1.1.3.1 Well-Known Text (WKT)

Η Well-Known Text αναπαράσταση παρέχει έναν τυποποιημένο τρόπο αναπαράστασης σε κείμενο όχι μόνο για γεωμετρικά αντικείμενα, αλλά και για τα χωρικά συστήματα αναφοράς τους, τους μετασχηματισμούς τους, κτλ. Το WKT είναι ένα ευρέως χρησιμοποιούμενο πρότυπο του OGC για την αναπαράσταση γεωμετρικών, όπως περιγράφεται στην προδιαγραφή “OpenGIS Simple Features Access – Part 1: Common Architecture” [OGC11]. Έχει γίνει επίσης το πρότυπο ISO 19125-1, που ασχολείται με την αναπαράσταση και τον χειρισμό απλών χαρακτηριστικών (simple features). Ένα απλό χαρακτηριστικό περιορίζεται στους 2-, 3- και 4- διαστατικούς χώρους και τα χωρικά ιδιοχαρακτηριστικά του περιγράφονται τμηματικά από γεωμετρίες στο WKT. Ως εκ τούτου η

προδιαγραφή WKT απασχολείται κυρίως με 2-διαστατικές γεωμετρίες με x,y συντεταγμένες, αλλά επιτρέπει να έχουν και συσχετισμένες τιμές z και m και σε λειτουργίες να έχουν πρόσβαση σε αυτές. Η ερμηνεία των συντεταγμένων για μία γεωμετρία εξαρτάται από το χωρικό σύστημα αναφοράς που είναι πάντα συσχετισμένο με κάθε γεωμετρία. Σύμφωνα με την προδιαγραφή WKT, το χωρικό σύστημα αναφοράς δεν είναι ενσωματωμένο στην αναπαράσταση του αντικειμένου, αλλά προσαρτάται χωριστά χρησιμοποιώντας κατάλληλο συμβολισμό.

2.1.1.3.2 Geography Markup Language (GML)

Η Geometry Markup Langage είναι μία XML γραμματική γραμμένη σε XML Schema για την περιγραφή των σχημάτων εφαρμογών, καθώς και την ανταλλαγή και αποθήκευση γεωγραφικής πληροφορίας. Οι βασικές έννοιες που χρησιμοποιούνται από το GML για να μοντελοποιηθεί ο κόσμος έχουν εξαχθεί από τη ISO 19100 series of International Standards και το OGC Abstract Specification. Το πρότυπο GML [OGC07] δηλώνει έναν μεγάλο αριθμό στοιχείων (elements) και ιδιοχαρακτηριστικών (attributes) XML με σκοπό να υποστηρίξει ένα μεγάλο εύρος δυνατοτήτων για γεωχωρικά δεδομένα. Για παράδειγμα, το πρότυπο GML μπορεί να κωδικοποιήσει δυναμικά χαρακτηριστικά, χωρικές και χρονικές τοπολογίες, πολύπλοκους τύπους γεωμετρικών ιδιοτήτων, συστήματα συντεταγμένων αναφοράς και καλύψεις.

Με ένα τέτοιο μεγάλο εύρος, διαλειτουργικότητα μπορεί να επιτευχθεί μόνο μέσω του ορισμού κατατομών (profile) του GML που απασχολούνται με ένα περιορισμένο υποσύνολο των δυνατοτήτων του GML. Οι κατατομές του GML (δηλαδή, λογικοί περιορισμοί του GML σε συγκεκριμένες εφαρμογές) μπορούν να προσδιοριστούν μέσω ενός εγγράφου XML, μέσω ενός σχήματος XML ή και με τους δύο τρόπους. Υπάρχουν πολλές διαφορετικές κατατομές διαθέσιμες, όπως για παράδειγμα για Points, Simple Features, RSS, κτλ. Τέτοιες κατατομές GML διαφέρουν από σχήματα εφαρμογών, γιατί είναι μέρη των GML namespaces (OGC GML) και ορίζουν περιορισμένα υποσύνολα της GML. Σε αντίθεση, τα σχήματα εφαρμογών είναι λεξιλόγια XML που είναι συγκεκριμένα για την εφαρμογή και είναι έγκυρα μόνο μέσα στο πλαίσιο των συγκεκριμένων για την εφαρμογή namespaces.

Το GML Simple Features Profile [OGC10a] και το Simple Features for SQL [OGC10b] περιγράφουν τις ίδιες γεωμετρίες. Όμως, το GML Simple Features Profile μπορεί να εφαρμοστεί και σε γεωμετρίες σε τρεις διαστάσεις, ενώ το Simple Features for SQL περιορίζεται σε μόνο δύο διαστάσεις. Στο GML Simple Features Profile, ένα χαρακτηριστικό μπορεί να έχει οποιονδήποτε αριθμό γεωμετρικών ιδιοτήτων και κάθε γεωμετρία πρέπει να αναφέρεται σε ένα σύστημα αναφοράς συντεταγμένων που έχει 1, 2 ή 3 διαστάσεις.

2.1.1.4 OGC GeoSPARQL

Στο παρών κεφάλαιο παρουσιάζουμε πρώτον το σχεδιαστικό κίνητρο του προτύπου OGC GeoSPARQL και στη συνέχεια δίνουμε τις βασικές συνιστώσες (κλάσεις συμβατότητας) που το αποτελούν.

2.1.1.4.1 Σχεδιαστικό κίνητρο

Η προτυποποίηση του GeoSPARQL είναι ανάμεσα στους στόχους του OGC με σκοπό να επιτευχθεί μία συνεπής αναπαράσταση γεωχωρικών σημασιολογικών δεδομένων στον Παγκόσμιο Ιστό, και ως εκ τούτου να επιτραπεί και στους παρόχους και στους χρήστες των δεδομένων και εφαρμογών να έχουν ομοιόμορφη πρόσβαση σε γεωχωρικά RDF δεδομένα.

Ο γεωχωρικός συλλογισμός είναι κρίσιμος για ένα μεγάλο εύρος πεδίων εφαρμογών (σχεδίαση συγκοινωνιών, υδρολογία, χρήση γης, κτλ.). Χάρη στην δυνατότητά τους για συναγωγή και ζεύξη δεδομένων, οι αποθήκες τριπλετών έχουν αποδειχθεί ελκυστικές για γεωχωρική επεξεργασία και συλλογιστική. Η θεμελιώδης τεχνολογία βάσεων δεδομένων παρέχει συνέπεια και ακεραιότητα για κλιμακούμενα μεγέθη πληροφορίας, καθώς και σημαντικά εργαλεία για κατασκευή ευρετηρίων, βελτιστοποίηση επερωτήσεων, κτλ. Από την άλλη μεριά, οι αποθήκες τριπλετών είναι πολύ καλύτερα εξοπλισμένες από σχεσιακές βάσεις δεδομένων ώστε να μπορούν να αντιμετωπίζουν διάφορα θέματα, όπως πολλαπλές ενώσεις μεταξύ οντοτήτων, επερωτήσεις με μεταβλητές ιδιότητες ή οντολογική συναγωγή σε σετ δεδομένων. Η γεωχωρική πληροφορία συχνά περιέχει πολύπλοκες ιεραρχίες τύπων που δεν μπορούν να εκφραστούν πλήρως ή να γίνει εκμετάλλευσή τους σε τυπικές GIS πλατφόρμες. Για παράδειγμα, ένα ποτάμι μπορεί να είναι πορθμός, υδατική αρτηρία και διοικητικό όριο. Επίσης, η διασύνδεση πηγών δεδομένων από τον παγκόσμιο ιστό είναι μείζονος σημασίας. Για παράδειγμα, σημεία ενδιαφέροντος συνδυασμένα με τιμές ξενοδοχείων, προτάσεις χρηστών και κατευθύνσεις οδήγησης θα μπορούσαν να προσφέρουν πολύ περισσότερο εκλεπτυσμένες και εξατομικευμένες ταξιδιωτικές υπηρεσίες. Επομένως, ένα συνδυασμός της ώριμης τεχνολογίας βάσεων δεδομένων με τη γνώση και τις δυνατότητες συλλογιστικής των αποθηκών RDF θα ήταν εξαιρετικά υποσχόμενη, και για τον Σημασιολογικό Ιστό και για το Γεωχωρικό πεδίο.

2.1.1.4.2 Συνιστώσες

Το πρότυπο GeoSPARQL υποστηρίζει την αναπαράσταση και επερώτηση πάνω σε γεωχωρικά δεδομένα στον Σημασιολογικό Ιστό. Το GeoSPARQL ορίζει ένα λεξιλόγιο για την αναπαράσταση γεωχωρικών δεδομένων σε RDF, καθώς και μία επέκταση στην γλώσσα επερωτήσεων SPARQL για την επεξεργασία γεωχωρικών δεδομένων. Ορίζει ένα θεμελιώδες σετ κλάσεων, ιδιοτήτων και τύπων δεδομένων που μπορούν να χρησιμοποιηθούν για την κατασκευή προτύπων επερωτήσεων, έτσι ώστε να γίνουν δυνατές οι πολλές χρήσιμες επεκτάσεις σε αυτό το λεξιλόγιο. Το πρόσφατα δημοσιευθέν

πρότυπο [OGC12] εδραιώνει αρκετές κλάσεις απαιτήσεων και αντίστοιχες κλάσεις συμβατότητας για το GeoSPARQL. Οποιαδήποτε υλοποίηση αξιώνει συμβατότητα με μία από τις κλάσεις συμβατότητας πρέπει να περνάει όλα τα τεστ σε ένα αφηρημένο πακέτο ελέγχου σύμφωνα με τις απαιτήσεις και τα URI των τεστ συμβατότητας που ορίζονται στο [OGC12].

Το GeoSPARQL πρότυπο ακολουθεί μία τμηματική (modular) αρχιτεκτονική και περιέχει αρκετές συνιστώσες. Οι υλοποιήσεις μπορούν να παρέχουν διάφορα επίπεδα λειτουργικότητας επιλέγοντας ποιες κλάσεις συμβατότητας να υποστηρίξουν. Για παράδειγμα, ένα σύστημα που βασίζεται αποκλειστικά σε ποιοτική χωρική συλλογιστική μπορεί να υποστηρίζει μόνο τις συνιστώσες core και topological vocabulary. Κάθε μία από αυτές τις συνιστώσες συνθέτει μία κλάση απαιτήσεων του GeoSPARQL:

- Η συνιστώσα *core* ορίζει υψηλού επιπέδου RDFS/OWL κλάσεις για την αναπαράσταση χωρικών αντικειμένων. Το προκύπτον λεξιλόγιο μπορεί να χρησιμοποιηθεί για την κατασκευή SPARQL προτύπων γράφων για επερωτήσεις σε κατάλληλα μοντελοποιημένα γεωχωρικά δεδομένα. Τα λεξιλόγια RDFS και OWL έχουν χρησιμοποιηθεί ώστε το λεξιλόγιο να μπορεί να γίνεται κατανοητό από συστήματα που υποστηρίζουν μόνο RDFS συνεπαγωγή και από συστήματα που υποστηρίζουν βασισμένη σε OWL συλλογιστική. Οι υλοποιήσεις πρέπει να υποστηρίζουν την SPARQL Query Language for RDF, το SPARQL Protocol for RDF και το SPARQL Query Results XML μορφότυπο. Οι RDFS κλάσεις *geo:SpatialObject* και *geo:Feature* πρέπει να χρησιμοποιούνται σε SPARQL πρότυπα γράφων για την αναπαράσταση γεωχωρικών χαρακτηριστικών.
- Η συνιστώσα *topology vocabulary* ορίζει RDF ιδιότητες για ισχυρισμούς και επερωτήσεις σε τοπολογικές σχέσεις πάνω σε χωρικά αντικείμενα. Η κλάση είναι παραμετρική ώστε να μπορούν να χρησιμοποιηθούν διαφορετικές οικογένειες τοπολογικών σχέσεων, π.χ. RCC8 [RCC92], Egenhofer [EF91]. Αυτές οι σχέσεις είναι γενικές ώστε να μπορούν να συνδέσουν χαρακτηριστικά (features) όπως και γεωμετρίες. Ένα πρότυπο DE9IM [CFO93, CSE94] χρησιμοποιείται για να περιγράψει κάθε χωρική σχέση, προσδιορίζοντας τη χωρική συντεταγμένη των τομών μεταξύ των εσωτερικών, συνόρων και εξωτερικών ενός ζεύγους γεωμετρικών αντικειμένων.

  Οι υλοποιήσεις για την OGC Simple Features οικογένεια πρέπει να επιτρέπουν τις ιδιότητες *geo:sfEquals, geo:sfDisjoint, geo:sfIntersects, geo:sfTouches, geo:sfCrosses, geo:sfWithin, geo:sfContains, geo:sfOverlaps* να χρησιμοποιούνται σε SPARQL πρότυπα γράφων. Παρόμοιες ιδιότητες ορίζονται για τις τοπολογικές οικογένειες που βασίζονται στα μοντέλα των Egenhofer (π.χ., *geo:ehMeet)* και RCC8 (π.χ., *geo:rcc8tppi)* . Το πρότυπο GeoSPARQL δεν επιβάλει την χρήση της ίδιας ορολογίας από όλες τις υλοποιήσεις, αλλά το αφήνει ανοικτό για κάθε υλοποίηση να επιλέξει ποια υλοποίηση θα υποστηρίξει.

- Η συνιστώσα *geometry component* ορίζει RDFS τύπους δεδομένων για τη σειριοποίηση γεωμετρικών δεδομένων, σχετικών με γεωμετρία RDF ιδιοτήτων και μη-τοπολογικές χωρικών επερωτήσεων λειτουργίες για γεωμετρικά αντικείμενα. Μία κλάση γεωμετρίας μονής ρίζας ορίζεται με το *geo:Geometry*. Στα SPARQL πρότυπα γράφων, η ιδιότητα *geo:hasGeometry* χρησιμοποιείται για να συνδέσει ένα χαρακτηριστικό με μία γεωμετρία που περιγράφει τη χωρική έκτασή του. Ένα δοσμένο χαρακτηριστικό μπορεί να έχει πολλαπλές συσχετισμένες με αυτό γεωμετρίες, π.χ., ένας σταθμός μπορεί να αναπαριστάται ως ένα σημείο, ένα γραμμικό σύνορο ή ένα πολύγωνο. Περαιτέρω, η ιδιότητα *geo:defaultGeometry* χρησιμοποιείται για να αναπαραστήσει ένα χαρακτηριστικό με την προτερότιμη (default) γεωμετρία του. Η προτερότιμη γεωμετρία είναι η γεωμετρία που πρέπει να χρησιμοποιηθεί για χωρικούς υπολογισμούς απουσία αίτησης για κάποια συγκεκριμένη γεωμετρία (π.χ., στην περίπτωση query rewrite). Είναι πιθανό για ένα χαρακτηριστικό να έχει περισσότερες της μία διακριτές προτερότιμες γεωμετρίες ή καμία. Μία κυριολεκτική (literal) αναπαράσταση μίας γεωμετρίας απαιτείται ώστε οι γεωμετρικές τιμές να μπορούν να αντιμετωπιστούν ως μία οντότητα και παρέχει κλάσεις για πολλούς διαφορετικούς γεωμετρικούς τύπους όπως σημείο, πολύγωνο, καμπύλη, τόξο, κτλ. Μία τέτοια κωδικοποίηση επιτρέπει σε γεωμετρίες να περαστούν σε εξωτερικές λειτουργίες για υπολογισμούς και να επιστρέφονται από επερωτήσεις. Η GeoSPARQL παρέχει δύο τρόπους για την αναπαράσταση γεωμετρικών literals και τις συσχετισμένες αυτών ιεραρχίες τύπων, τα WKT και GML, επιτρέποντας σε υλοποιήσεις να υποστηρίζουν οποιοδήποτε από ή και τα δύο. Η επιλεγμένη σειριοποίηση επηρεάζει έντονα την εννοιοποίηση της γεωμετρίας. Η σειριοποίηση WKT ευθυγραμμίζει τους γεωμετρικούς τύπους με το ISO 19125 Simple Features και η σειριοποίηση GML με το ISO 19107 Spatial Schema. Οι ιδιότητες *geo:asWKT* και *geo:asGML* συνδέουν ένα χαρακτηριστικό με αναπαράστασή του ως γεωμετρικό literal. Οι τιμές για αυτές τις ιδιότητες χρησιμοποιούν τους τύπους δεδομένων *geo:WKTLiteral* και *geo:GMLLiteral* αντίστοιχα. Το WGS84 υποτίθεται ως το σύστημα συντεταγμένων αναφοράς για κάθε *geo:wktLiteral* που δεν παρέχει URI συστήματος συστήματος συντεταγμένων αναφοράς. Οι υλοποιήσεις πρέπει να επιτρέπουν βασικές χωρικές μεθόδους πάνω σε γεωμετρίες με τις πρότυπες ιδιότητες *geo:dimension, geo:coordinateDimension, geo:spatialDimension, geo:isEmpty, geo:isSimple, geo:hasSerialization* να χρησιμοποιούνται σε SPARQL πρότυπα γράφων. Επιπλέον, υποστήριξη πρέπει να παρέχεται για χωρική ανάλυση με τα *geof:distance, geof:buffer, geof:convexHull, geof:intersection, geof:union, geof:difference, geof:symDifference, geof:envelope* και *geof:boundary* ως λειτουργίες επέκτασης SPARQL, σε άμεση συσχέτιση με τους ορισμούς στο OGC Simple Features (ISO 19125-1).
- Η συνιστώσα *geometry topology* ορίζει λειτουργίες τοπολογικών επερωτήσεων που λειτουργούν ως φίλτρα πάνω σε γεωμετρικά literals και επιστρέφουν boolean τιμές. Η κλάση

είναι παραμετρική για να παρέχει ευελιξία ως προς τις οικογένειες τοπολογικών σχέσεων και ως προς τις γεωμετρικές σειριοποιήσεις.. Οι υλοποιήσεις πρέπει να υποστηρίζουν το *geof:relate* ως λειτουργία επέκτασης SPARQL χρησιμοποιώντας το Dimensionally Extended 9-Intersection Model (DE9IM), σε πλήρη συνέπεια με τον τελεστή *relate* όπως ορίζεται στο Simple Features (ISO 19125-1). Υποστήριξη παρέχεται επίσης για τις λειτουργίες επέκτασης SPARQL *geof:sfEquals*, *geof:sfDisjoint*, *geof:sfIntersects*, *geof:sfTouches*, *geof:sfCrosses*, *geof:sfWithin*, *geof:sfContains*, *geof:sfOverlaps* για το μοντέλο Simple Features καθώς για άλλες τοπολογικές οικογένειες όπως τις RCC8 ή Egenhofer. Επισημαίνουμε ότι μόνο ποσοτικής ανάλυσης εφαρμογές μπορούν να κάνουν χρήση αυτών των τοπολογικών λειτουργιών, ενώ οι δυαδικές τοπολογικές ιδιότητες συσχετίζουν οντότητες *geo:Geometry* και *geo:Feature*, οπότε μπορούν να χρησιμοποιηθούν και σε ποιοτικής και σε ποσοτικής ανάλυσης εφαρμογές.

- Η συνιστώσα *RDFS entailment* ορίζει έναν μηχανισμό για υπονοούμενο (implicit) ταίριασμα RDF τριπλετών που προκύπτουν με βάση RDF και RDFS σημασιολογία. Η κλάση είναι παραμετρική ώστε να δίνει στις υλοποιήσεις ευελιξία ως προς τις οικογένειες τοπολογικών σχέσεων και ως προς τους γεωμετρικούς τύπους που θα επιλέξουν να υποστηρίξουν. Οι υλοποιήσεις πρέπει να υποστηρίζουν πρότυπα γράφων που περιλαμβάνουν όρους από μία RDF/OWL ιεραρχία κλάσεων γεωμετρικών τύπων συμβατή με αυτή στο OGC Simple Features (ISO 19125-1) ή στο GML schema (OGC 07-036).
- Η συνιστώσα *query rewrite* ορίζει κανόνες RIF [OGC12] για το μετασχηματισμό ενός απλού προτύπου τρίπλας που τεστάρει μία τοπολογική σχέση μεταξύ δύο χαρακτηριστικών σε μία ισοδύναμη επερώτηση που περιλαμβάνει συγκεκριμένες γεωμετρίες και λειτουργίες τοπολογικών επερωτήσεων. Τοπολογικές σχέσεις χαρακτηριστικού με χαρακτηριστικό και χαρακτηριστικού με γεωμετρία επιτυγχάνονται με το συνδυασμό της ιδιότητας *geo:defaultGeometry* και των κανόνων query rewrite. Μία πιθανή στρατηγική υλοποίησης είναι να μετασχηματιστεί η δοσμένη επερώτηση με την επέκταση του προτύπου τριπλετών χρησιμοποιώντας ένα άμεσο χωρικό κατηγόρημα σε μία σειρά προτύπων τριπλετών και κλήση της αντίστοιχης λειτουργίας επέκτασης όπως αυτές ορίζονται στους RIF κανόνες. Αν το *geo:Feature* είναι το υποκείμενο ή αντικείμενο κάποιας τοπολογικής σχέσης η επερώτηση ξαναγράφεται αυτόματα ώστε να συγκρίνει την *geo:Geometry* που είναι συνδεδεμένη ως προτερότιμη, με αυτόν τον τρόπο αφαιρώντας το abstraction προ της επεξεργασίας. Το ταίριασμα βασικών προτύπων γράφων πρέπει να χρησιμοποιεί τη σημασιολογία που ορίζεται από την W3C με το RIF Core Entailment Regime για τους RIF κανόνες *geor:sfEquals*, *geor:sfDisjoint*, *geor:sfIntersects*, *geor:sfTouches*, *geor:sfCrosses*, *geor:sfWithin*, *geor:sfContains*, *geor:sfOverlaps*. Παρόμοιοι κανόνες μετασχηματισμών ορίζονται για άλλα μοντέλα όπως το Egenhofer (π.χ., *geor:ehMeet)* και το RCC8 (π.χ., *geor:rcc8tppi)* . Η

υλοποίηση αυτής τη συνιστώσας είναι επιλεκτική, οπότε αποθήκες τριπλετών που δεν υποστηρίζουν query rewriting εξακολουθούν θεωρούνται συμβατές με το GeoSPARQL.

2.1.2 ΕΛΟΤ 743

Το Ελληνικό Πρότυπο ΕΛΟΤ 743 [ΕΛΟΤ743] καθιερώνει ένα σύστημα για το μεταγραμματισμό ή/και μεταγραφή των Ελληνικών χαρακτήρων σε Λατινικούς Χαρακτήρες. Αποτελεί την 2η έκδοση του Προτύπου ΕΛΟΤ 743:1982 το οποίο και αντικαθιστά και είναι ταυτόσημο με το Διεθνές Πρότυπο ISO 843:1997 όπως αυτό διορθώθηκε και επανεκτυπώθηκε την 1999-05-01.

Το ΕΛΟΤ 743 παρέχει δύο σύνολα κανόνων, καθένα από τα οποία συνιστά έναν τύπο μετατροπής.

- Τύπος 1, μεταγραμματισμός των Ελληνικών χαρακτήρων σε Λατινικούς χαρακτήρες.
- Τύπος 2, μεταγραφή των Ελληνικών χαρακτήρων σε Λατινικούς χαρακτήρες.

Το πρότυπο αυτό εφαρμόζεται για τους χαρακτήρες της Ελληνικής γραφής, ανεξάρτητα από τη χρονική περίοδο κατά την οποία αυτή χρησιμοποιείται ή χρησιμοποιήθηκε, δηλ. εφαρμόζεται σε μονοτονικά και πολυτονικά κείμενα από όλες τις περιόδους της Κλασσικής ή Μοντέρνας Ελληνικής καθώς και σε κάθε άλλη μορφή γραψίματος που χρησιμοποιεί την Ελληνική γραφή.

Το πρότυπο δεν ορίζει αυστηρά ποιος τύπος μετατροπής θα πρέπει να χρησιμοποιηθεί σε κάθε εφαρμογή. Μία εφαρμογή μπορεί να επιλέξει ένα, και μόνο ένα, από τους παραπάνω τύπους για κάποιο συγκεκριμένο σκοπό. Η εφαρμογή πρέπει να δηλώνει ρητά τον τύπο μετατροπής που υιοθετεί.

Το πρότυπο, εντούτοις, συνιστά ένα προτιμητέο τρόπο χρήσης των δύο τύπων μετατροπής:

- Ο Τύπος 1 (μεταγραμματισμός) μπορεί να χρησιμοποιηθεί στην περίπτωση που απαιτείται μονοσήμαντη μετατροπή του μεταγραμματισμένου κειμένου στην αρχική του μορφή, όπως π.χ., από κάποια μηχανή.
- Ο Τύπος 2 (μετατροπή) μπορεί να χρησιμοποιηθεί στην περίπτωση όπου η σωστή προφορά της Ελληνικής λέξης προέχει της ανάγκης επαναφοράς της αρχικής μορφής, όπως π.χ., σε διαβατήρια, οδικά σήματα, κτλ.

Table 1: Τύπος 1 (Μεταγραμματισμός)

	Ελληνικοί Χαρακτήρες	Λατινικοί Χαρακτήρες		Ελληνικοί Χαρακτήρες	Λατινικοί Χαρακτήρες
1	Α, α	A, a	13	Ν, ν	N, n
2	Β, β	V, v	14	Ξ, ξ	X, x
3	Γ, γ	G, g	15	Ο, ο	O, o
4	Δ, δ	D, d	16	Π, π	P, p
5	Ε, ε	E, e	17	Ρ, ρ	R, r

6	Z, ζ	Z, z	18	Σ, σ, ς	S, s, s
7	Η, η	Ī, ī	19	Τ, τ	T, t
8	Θ, θ	TH, th	20	Υ, υ	Y, y
9	Ι, ι	I, i	21	Φ, φ	F, f
10	Κ, κ	K, k	22	Χ, χ	CH, ch
11	Λ, λ	L, l	23	Ψ, ψ	PS, ps
12	Μ, μ	M, m	24	Ω, ω	Ō, ō

Στα πλαίσια αυτής της διπλωματικής όπου αναφέρεται μεταγραμματισμός Ελληνικών χαρακτήρων με Λατινικούς χαρακτήρες εννοείται ότι έχει γίνει χρήση του Τύπου 1 (μεταγραμματισμός) με τις εξής αλλαγές.

1. Στο μεταγραμματισμό των γραμμάτων “Η, η” χρησιμοποιούνται οι λατινικοί χαρακτήρες “I, i” αντί των “Ī, ī” που ορίζονται στον τύπο.
2. Στο μεταγραμματισμό των γραμμάτων “Ω, ω” χρησιμοποιούνται οι λατινικοί χαρακτήρες “O, o” αντί των “Ō, ō” που ορίζονται στον τύπο.

Σκοπός αυτών των αλλαγών είναι η προκύπτουσα συμβολοσειρά να μπορεί να εκφραστεί με κωδικοποίηση ASCII για την περίπτωση εφαρμογών που δεν μπορούν να επεξεργαστούν άλλες κωδικοποιήσεις. Η απώλεια δυνατότητας επαναφοράς της αρχικής συμβολοσειράς δεν μας επηρεάζει γιατί σε κάθε περίπτωση αυτή διατηρείται μαζί με την μεταγραμματισμένη συμβολοσειρά.

2.2 Δεδομένα

Για τους σκοπούς της παρούσας διπλωματικής κάναμε χρήση δεδομένων χαρτογράφησης από ένα εύρος διαδικτυακών πηγών. Στο παρών κεφάλαιο δίνουμε μία συνοπτική παρουσίαση αυτών των πηγών και των αντίστοιχων δεδομένων.

2.2.1 OpenStreetMap

Το OpenStreetMap (OSM) [OSM] είναι μια πρωτοβουλία για να παραχθούν και να διατεθούν δωρεάν γεωγραφικά δεδομένα, όπως οδικοί χάρτες, σε οποιονδήποτε. Το OpenStreetMap Foundation είναι ένας διεθνής μη-κερδοσκοπικός οργανισμός που υποστηρίζει, αλλά δεν ελέγχει το OpenStreetMap Project. Κινητήριος δύναμη πίσω από τη δημιουργία του OSM ήταν οι περιορισμοί στη χρήση ή στη διαθεσιμότητα γεωγραφικών δεδομένων για τα περισσότερα μέρη του κόσμου, καθώς και η έλευση οικονομικών συσκευών δορυφορικής πλοήγησης. Από την ίδρυση του από τον Steve Coast στο Ηνωμένο Βασίλειο το 2004 έχει φτάσει να έχει σήμερα περί τους τριακόσιες χιλιάδες συνεισφέροντες. Τα δεδομένα που συλλέγονται γίνονται διαθέσιμα υπό την άδεια Open Data Commons Open Database License (OdbL).

Εσωτερικά το OSM αποθηκεύει τα γεωγραφικά δεδομένα χρησιμοποιώντας τρία βασικά αρχέγονα (data primitives): Nodes, που αναπαριστούν σημεία στο χώρο, ways, που αναπαριστούν γραμμικά χαρακτηριστικά και περιοχές, και relations που χρησιμοποιούνται για να δείξουν σχέσεις μεταξύ των άλλων αρχέγονων. Όλα τα ανωτέρω μπορούν να έχουν ένα ή περισσότερα tags συσχετισμένα μαζί τους. Tags είναι ζευγάρια key, value, που γράφονται ως key=value και χρησιμοποιούνται για να περιγράψουν nodes, ways και relations, όπως για παράδειγμα name=*, highway=residential, κτλ.. Η γεωχωρική πληροφορία αποθηκεύεται στο σύστημα συντεταγμένων WGS-84.

Εκτός του web interface το OSM παρέχει εναλλακτικούς τρόπους πρόσβασης στα δεδομένα του μέσω ενός API και μέσω data dumps. Τα δεδομένα παρέχονται σε τέσσερα βασικά format, OSM XML, ένας XML μορφότυπος που παρέχεται από το API, PBF, ένας συμπιεσμένος, βελτιστοποιημένος μορφότυπος που σταδιακά αντικαθιστά το OSM XML, o5m, που σχεδιάστηκε ως συμβιβασμός μεταξύ των OSM XML και PBF και OSM JSON, που είναι παραλλαγή του OSM XML σε JSON. Dumps των δεδομένων του OSM, είτε για όλο τον πλανήτη είτε για συγκεκριμένες περιοχές είναι διαθέσιμα από έναν μεγάλο αριθμό επίσημων και 3rd party mirrors σε συμπιεσμένα OSM XML ή PBF αρχεία.

2.2.2 WikiMapia

Το WikiMapia [WikiMapia] είναι ένα ανοικτού περιεχομένου (open content) συνεργατικό έργο χαρτογράφησης που ξεκίνησε το 2006 από τους Alexandre Koriakine και Evgeniy Saveliev. Δηλωμένος στόχος του έργου είναι να δημιουργηθεί ένας δωρεάν, πλήρης, πολύγλωσσος και επίκαιρος χάρτης για όλο τον κόσμο με την συλλογή πληροφορίας για γεωγραφικά χαρακτηριστικά και τη διάθεσή της στον δημόσιο τομέα (public domain). Από το Μάιο του 2012 το WikiMapia ανακοίνωσε ότι όλα τα δεδομένα του θα είναι διαθέσιμα υπό την άδεια Creative Commons License Attribution-ShareAlike (CC BY-SA).

Εκτός του web interface το WikiMapia δίνει πρόσβαση στα δεδομένα μέσω ενός API που επιστρέφει αρχεία με format xml , kml, json ή jsonp.

2.2.3 geodata.gov.gr

Το geodata.gov.gr [GeodataGov] σχεδιάστηκε, αναπτύχθηκε και συντηρείται από το Ινστιτούτο Πληροφοριακών Συστημάτων του Ερευνητικού Κέντρου «Αθηνά» με σκοπό να αποτελέσει ένα κεντρικό σημείο συλλογής, αναζήτησης, διάθεσης και απεικόνισης της ανοικτής δημόσιας γεωχωρικής πληροφορίας.

Το geodata.gov.gr παρέχει μια μεγάλη ποικιλία δημοσίων δεδομένων οργανωμένων ανά φορέα και ανά θεματική κατηγορία. Τα δεδομένα παρέχονται σε τρεις ισοδύναμους μορφότυπους SHP, GML και KML, ώστε να εξασφαλισθεί συμβατότητα με όλες τις εφαρμογές GIS. Η γεωχωρική

πληροφορία παρέχεται στα εξής συστήματα συντεταγμένων ανά μορφότυπο: SHP: ΕΓΣΑ87, GML: ΕΓΣΑ87, KML:WGS84 και υπό άδεια που ποικίλει ανάλογα με την προέλευση των δεδομένων.

2.2.4 POIs.gr

Το POIs.gr [POIs] είναι ένας χώρος που δημιουργήθηκε από μία ομάδα ενδιαφερομένων με σκοπό τη συλλογή σημείων ενδιαφέροντος (POIs) που θα μπορούσαν στη συνέχεια να φορτωθούν σε συσκευές δορυφορικής πλοήγησης. Στα πλαίσια της διπλωματικής η ιστοσελίδα εξετάστηκε ως πιθανή πηγή δεδομένων για πειραματικές δοκιμές των μετρικών διασύνδεσης, αλλά τελικά απορρίφθηκε λόγο του μικρού μεγέθους των παρεχόμενων datasets.

2.3 Σχετικά εργαλεία και βιβλιοθήκες

Στο παρών κεφάλαιο παρουσιάζουμε εκείνα τα εργαλεία και βιβλιοθήκες των οποίων κάναμε χρήση κατά την εκπόνηση της διπλωματικής.

2.3.1 Osmosis

Το Osmosis [Osmosis] είναι μία εφαρμογή περιβάλλοντος γραμμής εντολών γραμμένη σε Java για την επεξεργασία δεδομένων του OSM. Το εργαλείο αποτελείται από ένα πλήθος συνιστωσών που μπορούν να συνδεθούν αλυσιδωτά για να εκτελέσουν μεγαλύτερες λειτουργίες. Για παράδειγμα η εφαρμογή έχει συνιστώσες για ανάγνωση OSM data dumps σε μορφότυπους OSM XML ή PBF, για ανάγνωση και εγγραφή των δεδομένων σε βάση, για εξαγωγή change sets από ή για την εφαρμογή τους πάνω σε δεδομένα, για εξαγωγή δεδομένων με βάση bounding box (bbox), κτλ. Η εφαρμογή μπορεί να ενσωματωθεί ως βιβλιοθήκη σε άλλες εφαρμογές Java.

Για παράδειγμα, για να διαβαστούν τα δεδομένα από ένα OSM XML αρχείο planet.osm και στη συνέχεια να φορτωθούν σε μία τοπική βάση PostGIS χρησιμοποιώντας το OSM snapshot schema θα πρέπει να συνδέσουμε σε μία αλυσίδα τα components *–read-xml* και *–write-pgsql* ως εξής:

```
osmosis --read-xml file="planet.osm" --write-pgsql host="x" database="x"
user="x" password="x"
```

Στην παρούσα διπλωματική κάναμε χρήση της δυνατότητας του εργαλείου Osmosis να ενσωματωθεί σε μία εφαρμογή Java ως βιβλιοθήκη για να δώσουμε στο εργαλείο Geosm δυνατότητες ανάγνωσης των OSM data dumps και εγγραφής τους σε βάση δεδομένων. Παράλληλα έγινε γενικότερη χρήση του osmosis ως ανεξάρτητου εργαλείου για τη διαχείριση των OSM data dumps.

2.3.2 Apache Jena

Το Apache Jena [Jena] είναι ένα Java Framework ανοικτού κώδικα για την ανάπτυξη εφαρμογών σημασιολογικού ιστού. Το Jena παρέχει μια συλλογή εργαλείων και βιβλιοθηκών Java για να

βοηθήσει στην ανάπτυξη εφαρμογών, εργαλείων και servers σημασιολογικού ιστού και διασυνδεδεμένων δεδομένων. Είναι top-level project του Apache Foundation.

Το Framework περιέχει:

- Ένα API για την ανάγνωση, επεξεργασία και εγγραφή RDF δεδομένων στους μορφότυπους XML, N-Triples και Turtle,
- Ένα API οντολογίας για τον χειρισμό οντολογιών OWL και RDFS,
- μια βασισμένη σε κανόνες μηχανή συναγωγής για συλλογιστική με δεδομένα RDF και OWL,
- αποθήκες για να μπορούν να αποθηκευτούν αποδοτικά μεγάλοι αριθμοί RDF triples στο δίσκο,
- μηχανή επερωτήσεων συμβατή με την τελευταία προδιαγραφή SPARQL,
- servers που επιτρέπουν σε δεδομένα RDF να δημοσιεύονται σε άλλες εφαρμογές χρησιμοποιώντας μία ποικιλία πρωτοκόλλων, συμπεριλαμβανομένου του SPARQL.

Η βιβλιοθήκη Jena χρησιμοποιείται στην εφαρμογή Geosm για την παρασκευή και εγγραφή σε αρχείό RDF γράφων.

2.3.3 LinkedGeoData

Το LinkedGeoData (LGD) [LGD] είναι μία προσπάθεια να δοθεί γεωχωρική διάσταση στον Σημασιολογικό Ιστό. Το LGD χρησιμοποιεί τα δεδομένα που έχουν συλλεχθεί από το OpenStreetMap project και τα κάνει διαθέσιμα ως βάση γνώσης RDF. Κύριο συστατικό αυτής της προσπάθειας είναι η απεικόνιση δεδομένων του OSM (metadata, γεωχωρική πληροφορία, tags, κτλ.) σε RDF ιδιότητες. Για να επιτύχει αυτό το στόχο το LGD ορίζει τις RDF οντολογίες LGD/Ontology (lgdo:), LGD/Triplify (lgdt:) και LGD/Geometry (lgd:) και ορίζει ένα εκτεταμένο σύνολο απεικονίσεων από attributes του OSM σε RDF triples. Συγκεκριμένα:

- Metadata που είναι κοινά σε όλα τα αρχέγονα του OSM απεικονίζονται με βάση ένα καθορισμένο σετ απεικονίσεων από χαρακτηριστικά του OSM σε RDF predicates.
- Η γεωχωρική πληροφορία αναπαριστάται χρησιμοποιώντας το W3C – Basic Geo Vocabulary και το GeoVocab.
- Tags που είναι συσχετισμένα με αρχέγονα του OSM απεικονίζονται σε RDF με τη χρήση ~1900 απεικονίσεων χωρισμένων σε έξι κατηγορίες (datatype, literal, resource key, resource key=value, property, resource prefix). Αναλυτικά:
  - Τα datatype mappings ορίζουν τον τύπο δεδομένων για κάποια tags,
  - Τα literal mappings ορίζουν απεικονίσεις σε κατηγόρημα με αντικείμενο ένα literal (π.χ., <name:el=*> απεικονίζεται σε <rdfs:label> “*”@el),

- Τα resource key και resource key=value mappings ορίζουν απεικονίσεις σε κατηγόρημα με αντικείμενο ένα IRI (π.χ., <boundary=political> απεικονίζεται σε <rdfs:type> <http://linkedgeodata.org/ontology/PoliticalBoundary>). Οι απεικονίσεις resource key=value έχουν μεγαλύτερη προτεραιότητα των resource key.
- Τα property mappings ορίζουν απεικονίσεις σε κατηγόρημα χωρίς να επηρεάζουν το αντικείμενο (π.χ. <addr:postcode=*> απεικονίζεται σε <http://linkedgeodata.org/ontology/addr/postcode> *).
- Τα prefix mappings ορίζουν απεικονίσεις σε κατηγόρημα με αντικείμενο που υφίσταται προεπεξεργασία (π.χ., για να εξασφαλισθεί ότι το αντικείμενο είναι έγκυρο URL ή email address).

Στα πλαίσια της παρούσης διπλωματικής κάναμε χρήση των λεξιλογίων του LGD γενικότερα και των tag mappings ειδικότερα ώστε να εξασφαλισθεί συμβατότητα των παραγόμενων από το Geosm γράφων με τους γράφους που παράγει το LGD.

2.3.3.1 Sparqlify

Το Sparqlify είναι ένα SPARQL->SQL query rewriter εργαλείο που επιτρέπει τον ορισμό RDF views πάνω σε σχεσιακές βάσεις δεδομένων και χρησιμοποιείται στο πλαίσιο του LinkedGeoData project. Σε αντίθεση με τα περισσότερα σύγχρονα εργαλεία που χρησιμοποιούν RDF και XML για να αναπαραστήσουν τις πληροφορίες απεικόνισης, το Sparqlify τις εκφράζει ως view definitions βασισμένες στην SPARQL γραμματική, που έχει επεκταθεί με custom κανόνες παραγωγής. Με αυτόν τον τρόπο ένας χρήστης SPARQL θα είναι ήδη εξοικειωμένος με τα βασικά συντακτικά στοιχεία του Sparqlify.

Το Sparqlify αποτελείται από τρία βασικά υποσυστήματα, την Sparqlify engine που αποτελεί τον πυρήνα του εργαλείου, είναι γραμμένη σε Java και είναι υπεύθυνη για το SPARQL->SQL rewritting, τον Sparqlify server που παρέχει HTTP web interface και το Sparqlify platform που είναι ένα integration project και συνδέει τον server με το Linked Data interface Pubby και το SPARQL web front end Snorql.

2.3.4 Silk Link Discovery Framework

Το Silk Link Discovery Framework (Silk) [Silk] είναι ένα εργαλείο για την εύρεση συνδέσμων μεταξύ δεδομένων από διαφορετικά datasets. Χρησιμοποιώντας την δηλωτική γλώσσα Silk - Link Specification Language (Silk-LSL) μπορούν να προσδιοριστούν ποιοι σύνδεσμοι είναι επιθυμητό να βρεθούν ανάμεσα σε δύο datasets και ποιες συνθήκες πρέπει να τηρούνται ώστε δύο δεδομένα (κόμβοι) να διασυνδεθούν. Αυτές οι συνθήκες μπορούν να συνδυάσουν πολλαπλές μετρικές και

έχουν τη δυνατότητα να λαμβάνουν υπόψη τον γράφο γύρω από τους κόμβους που εξετάζονται. Η πρόσβαση στα datasources υπό εξέταση γίνεται μέσω του πρωτοκόλλου SPARQL.

Το Silk παρέχεται σε τρεις παραλλαγές. Την Silk-Single Machine, που τρέχει σε ένα μηχάνημα και έχει τη δυνατότητα να εξετάσει είτε τοπικά datasets, είτε datasets ευρισκόμενα σε απομακρυσμένα μηχανήματα μέσω του πρωτοκόλλου SPARQL. Χρησιμοποιεί multithreading και caching, καθώς και τον αλγόριθμο MultiBlock για τη βελτίωση των επιδόσεων. Την Silk MapReduce, που είναι σχεδιασμένη να τρέχει σε μία συστάδα μηχανημάτων και βασίζεται στο Hadoop. Την Silk Server, που χρησιμοποιείται ως συνιστώσα υπερανάλυσης ταυτότητας (identity resolution component) στο εσωτερικό εφαρμογών που καταναλώνουν συνδεδεμένα δεδομένα από τον Παγκόσμιο Ιστό. Επίσης παρέχεται ο Silk Workbench, μία εφαρμογή ιστού που καθοδηγεί τον χρήστη στη διαδικασία εύρεσης συνδέσμων μεταξύ datasources μέσω ενός γραφικού περιβάλλοντος και δίνοντας τη δυνατότητα γρήγορης εξέτασης των παραγόμενων συνδέσμων.

Το Silk Link Discovery Framework χρησιμοποιήθηκε για την υλοποίηση και πειραματική δοκιμή των μετρικών διασύνδεσης που περιγράφουμε στο κεφάλαιο 4.

2.3.5 Java Topology Suite

Η JTS Topology Suite (JTS) [JTS] είναι μία βιβλιοθήκη Java ανοικτού κώδικα που παρέχει ένα μοντέλο για γεωμετρικές λειτουργίες σε ένα γραμμικό, Ευκλείδειο επίπεδο. Πρωταρχικός σκοπός της βιβλιοθήκης είναι να χρησιμοποιείται ως θεμελιώδες συστατικό σε βασιζόμενο σε διανύσματα γεωχωρικό λογισμικό, όπως συστήματα GIS. Μπορεί επίσης να χρησιμοποιηθεί ως γενικής χρήσεως βιβλιοθήκη σε εφαρμογές υπολογιστικής γεωμετρίας. Η JTS στοχεύει να είναι πλήρως συμβατή με το Simple Features Specification for SQL του OGC [OGC10b].

Η βιβλιοθήκη JTS χρησιμοποιήθηκε κατά την υλοποίηση των γεωγραφικών μετρικών διασύνδεσης στο Silk.

2.3.6 PostGIS for PostgreSQL

Η επέκταση FLOSS PostGIS [PostGIS] προσθέτει υποστήριξη για γεωγραφικά αντικείμενα στην αντικειμενοστρεφή βάση δεδομένων PostgreSQL [PostgreSQL]. Η postGIS αναπτύσσεται από ομάδα συντελεστών καθοδηγούμενη από την PostGIS Project Steering Commitee (PSC). Η πρώτη έκδοση δημοσιεύτηκε το 2001 από την Refractions Research υπό την άδεια GNU General Public License. Η σταθερή έκδοση 1.0 δημοσιεύτηκε το 2005, ενώ το 2006 η PostGIS ξεκίνησε την υλοποίηση του προτύπου OGC Simple Features for SQL (μία προηγούμενη έκδοση του προτύπου στο [OGC10b]). Η έκδοση 2.0.3 είναι διαθέσιμη από τον Απρίλιο του 2013, αλλά επί του παρόντος δεν είναι επίσημα πιστοποιημένη ως OGC compliant παρότι υποστηρίζει ή υπερβαίνει τις περισσότερες προδιαγραφές του OGC.

Τα χωρικά διανύσματα αποθηκεύονται σε στήλη ειδικού τύπου GEOMETRY ή GEOGRAPHY ενός πίνακα PostgreSQL. Από την έκδοση 0.9, η PostGIS υποστηρίζει όλα τα αντικείμενα και λειτουργίες στο πρότυπο OGC Simple Features for SQL [OGC10b]. Και τα δύο των WKT, WKB περιέχουν πληροφορία σχετικά με τον τύπο του αντικειμένου και τις συντεταγμένες που αποτελούν το αντικείμενο. Στην εσωτερική αποθήκευση απαιτείται όλα τα χωρικά αντικείμενα να περιλαμβάνουν αναγνωριστικό χωρικού συστήματος αντιστοίχισης (SRID). Η PostGIS επεκτείνει αυτό το πρότυπο στην μορφή EWKB/EWKT for GEOMETRY με υποστήριξη για 3D (συντεταγμένες με υψόμετρο ή linear referencing) καθώς και 4D με ενσωματωμένη πληροφορία SRID. Επί του παρόντος αυτές η τυποποιήσεις είναι υπερσύνολο αυτών που έχουν υιοθετηθεί από το OGC (κάθε έγκυρο WKB/WKT είναι και έγκυρο EWKB/EWKT), αλλά αυτό μπορεί να αλλάξει στο μέλλον.

Βάση PostGIS χρησιμοποιείται στα πλαίσια του εργαλείου Geosm σε συνδυασμό με το osmosis για την φόρτωση των OSM data dumpds και στην συνέχεια για την εκτέλεση επερωτήσεων πάνω σε αυτά.

2.3.7 Google Refine (Open Refine)

Το Google Refine (Open Refine από την έκδοση 2.6) [GRefine] είναι ένα εργαλείο ανοικτού κώδικα για τον καθαρισμό, επεξεργασία και μετατροπή δεδομένων, τον εμπλουτισμό τους με χρήση υπηρεσιών διαδικτύου και τη σύνδεση τους με βάσεις δεδομένων όπως η Freebase. Το Google Refine λειτουργεί σε γραμμές δεδομένων με την πληροφορία οργανωμένη σε κελιά κάτω από στήλες. Δεδομένα μπορούν να εισαχθούν σε ένα Google Refine Project από τους μορφότυπους TSV, CSV, κείμενο διαχωρισμένο με κενά ή ειδικούς χαρακτήρες, Excel, XML, JSON, RDF (RDF XML ή N3) ή Google Spreadsheets.

Αφού έχουν εισαχθεί στο εργαλείο, παρέχεται τη δυνατότητα στον χρήστη να φιλτράρει τα δεδομένα ανά στήλη χρησιμοποιώντας όψεις (facets), ώστε να μπορεί να εξάγει συμπεράσματα όπως για παράδειγμα να μπορεί να δει την κατανομή τιμών για μια στήλη που περιέχει αριθμητικές τιμές ή το πλήθος των κελιών των οποίων η τιμή ταιριάζει με κάποιο regex για στήλες που περιέχουν συμβολοσειρές.

Στο τμήμα της επεξεργασίας, μπορούν να γίνει μία ποικιλία μετατροπών πάνω στα δεδομένα χρησιμοποιώντας συναρτήσεις γραμμένες στην ιδιόκτητη (proprietary) γλώσσα Google Refine Expression Language (GREL) ή τις γλώσσες Jython ή Clojure. Επίσης, μπορεί να γίνει σύμμειξη (reconciliation) των δεδομένων με κάποια βάση δεδομένων, όπως η Freebase ή μέσω των APIs από web services. Για παράδειγμα, μπορεί να γίνει χρήση του Google Geocoding API ώστε από μία στήλη που περιέχει διευθύνσεις να παραχθούν δύο άλλες στήλες που θα περιέχουν το γεωγραφικό πλάτος και μήκος που τους αντιστοιχούν.

Το Google Refine μπορεί να εξάγει τα επεξεργασμένα δεδομένα στους μορφότυπους TSV, CSV, Excel και HTML tables, ενώ τα βήματα επεξεργασίας μπορούν να εξαχθούν σε μορφότυπο JSON και

αντίστοιχα να εισαχθούν σε κάποιο άλλο Google Refine Project ώστε να επαναληφθούν σε καινούργια δεδομένα.

Τέλος, με χρήση του του RDF Refine extension για το Google Refine [RDF Refine] προστίθενται οι δυνατότητες εξαγωγής των επεξεργασμένων δεδομένων σε RDF γράφο σε μορφότυπο RDF/XML ή TURTLE και η δυνατότητα σύμμειξης τους με κάποιο SPARQL endpoint.

Το Google Refine με το RDF Refine extension χρησιμοποιήθηκε για την μετατροπή δεδομένων των πηγών WikiMapia, geodata.gov.gr και POIs.gr σε RDF στην πορεία των πειραματικών δοκιμών των μετρικών διασύνδεσης.

Η σελίδα αυτή είναι σκόπιμα λευκή.

3

Geosm - Εφαρμογή για μετασχηματισμό δεδομένων του OSM σε RDF

3.1 Ανάλυση απαιτήσεων συστήματος

Σχεδιαστικός στόχος της εφαρμογής geosm είναι η μετατροπή δεδομένων του OpenStreetMap project σε RDF γράφο με πρωταρχική έμφαση στην συμβατότητα του παραγόμενου γράφου με το OGC GeoSPARQL [OGC12] πρότυπο όσον αφορά τη γεωχωρική πληροφορία.

Το βασικό σενάριο χρήσης της εφαρμογής περιλαμβάνει την παροχή από τον χρήστη ενός τοπικού αντιγράφου των data dumps του OSM Project σε PBF ή OSM XML μορφότυπο και τη μετατροπή των δεδομένων σε N-TRIPLES ή N-QUADS με τρόπο διάφανο προς το χρήστη όσον αφορά τις απαιτούμενες ενδιάμεσες επεξεργασίες.

Κατά την ανάπτυξη της εφαρμογής έχει δοθεί ιδιαίτερη έμφαση στους εξής στόχους:

- Ευκολία χρήσης, ώστε ένας καινούργιος χρήστης να μπορεί να επωφεληθεί από την εφαρμογή χωρίς να απαιτούνται πολύπλοκες διαδικασίες εγκατάστασης και εκτεταμένη καμπύλη εκμάθησης,
- Επιδόσεις, ώστε η εφαρμογή να μπορεί να επεξεργαστεί μεγάλους όγκους δεδομένων σε συστήματα τυπικών δυνατοτήτων,
- Επεκτασιμότητα, ώστε να είναι εύκολο να αναπτυχθούν και να προστεθούν επιπλέον βήματα προεπεξεργασίας των δεδομένων προσαρμοσμένα στις απαιτήσεις των χρηστών,

- Ευρωστία, μέσω της υλοποίησης εκτεταμένου logging που ενημερώνει τον χρήστη για πιθανά σφάλματα και χρήση fail fast τεχνικών, ώστε να αποφευχθεί ο κίνδυνος παρείσφρησης προβληματικών δεδομένων στο output εν αγνοία του χρήστη.
- Συμβατότητα με το LinkedGeoData project όσον αφορά τα χρησιμοποιούμενα RDF λεξιλόγια μέσω της χρήσης των ίδιων mappings.

3.1.1 Συνεισφορά και σύγκριση με τα υπάρχοντα εργαλεία

Συγκρίνοντας την εφαρμογή Geosm με το Sparqlify γενικά και ειδικότερα με το LinkedGeoData Project που αναφέρθηκαν στο κεφάλαιο 2.3.3 παρατηρούμε ότι καλύπτουν αντίστοιχες περιπτώσεις χρήσης, δηλαδή έχουν στόχο την μετατροπή δεδομένων του OpenStreetMap project σε RDF. Μάλιστα κατά τη σχεδίαση του Geosm έγινε προσπάθεια να εξασφαλισθεί συμβατότητα με το LGD όσον αφορά τα λεξιλόγια που χρησιμοποιούνται για την αναπαράσταση των οντοτήτων του OSM. Για αυτό το λόγο κρίνεται σκόπιμο να συγκρίνουμε τα δύο εργαλεία και να επισημάνουμε τις διαφορές τους.

Καταρχήν, περιγράφουμε συνοπτικά τη διαδικασία που ακολουθούν τα δύο εργαλεία για την παραγωγή RDF.

Ως SPARQL->RDF rewriter το Sparqlify διαρθρώνεται με μία βάση δεδομένων και ένα σετ ορισμού όψεων RDB-to-RDF. Με βάση αυτή τη διάρθρωση το Sparqlify μεταφράζει τις μετέπειτα SPARQL επερωτήσεις σε δύο σχετιζόμενα artifacts: Μια SQL επερώτηση και μία δέσμευση (binding) των SPARQL μεταβλητών σε εκφράσεις πάνω στο result set της SQL επερώτησης. Η SQL επερώτηση εκτελείται πάνω στη σχεσιακή βάση δεδομένων και στην συνέχεια χρησιμοποιώντας τις ορισμένες απεικονίσεις το SQL result set μετατρέπεται σε SPARQL result set.

Αντίθετα, το Geosm ως εφαρμογή μετατροπής δεδομένων χρησιμοποιεί την ανάστροφη λογική. Ο πυρήνας της εφαρμογής διαρθρώνεται με ένα σύνολο SQL επερωτήσεων και ένα σύνολο απεικονίσεων πάνω στα SQL result sets. Οι SQL επερωτήσεις, που είναι σχεδιασμένες ώστε συνολικά να καλύπτουν πλήρως την πληροφορία που περιέχεται στην βάση δεδομένων, εκτελούνται πάνω στη βάση και στη συνέχεια με βάση τις ορισμένες απεικονίσεις παράγονται και γράφονται οι RDF triples.

Συγκριτικά τα πλεονεκτήματα του Geosm είναι:

- Ολοκλήρωση όλων των απαραίτητων βημάτων επεξεργασίας στα δεδομένα του OpenStreetMap με τρόπο διάφανο ως προς το χρήστη,
- Ευκολία στην εγκατάσταση σε έναν οποιοδήποτε προσωπικό υπολογιστή ή server χωρίς την απαίτηση εκτεταμένης υποδομής,
- Ευκολία στην χρήση,

- Ευκολία στην επέκταση με νέες δυνατότητες χάρη στο μικρό μέγεθος και την απλή δομή της εφαρμογής.

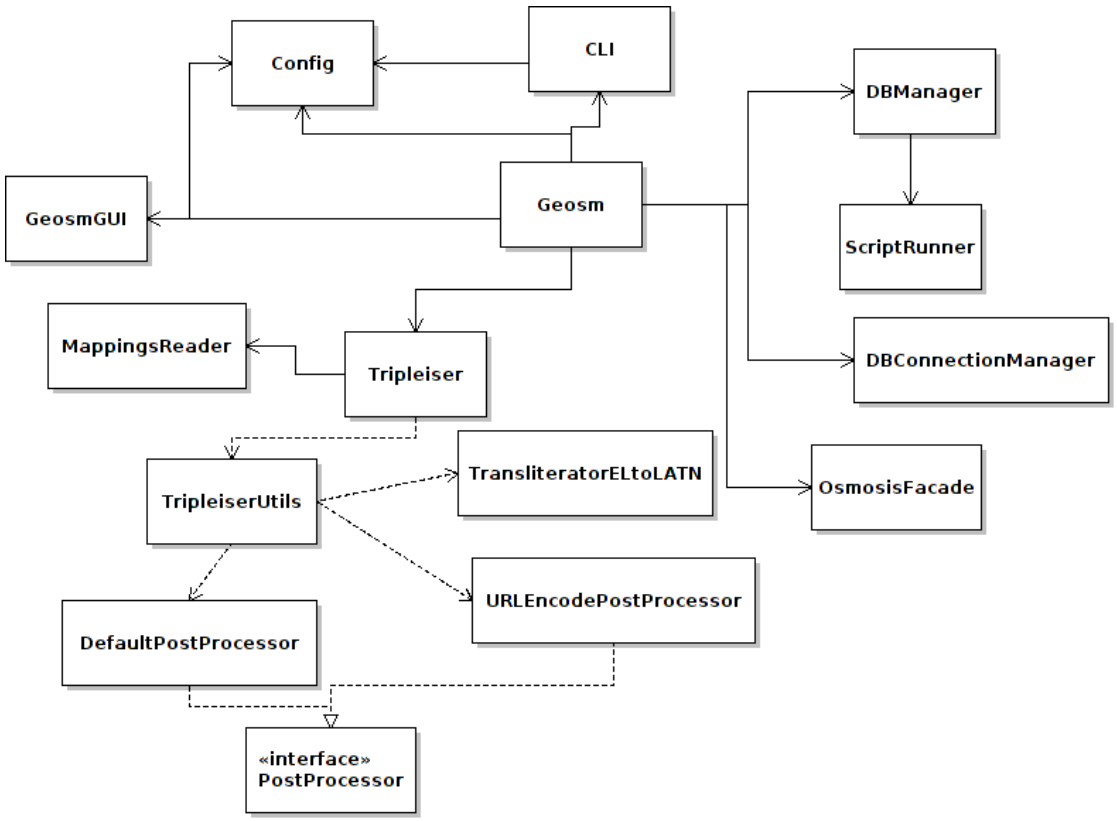
Αντίθετα, τα μειονεκτήματά του είναι:

- Υψηλή προσαρμογή στα συγκεκριμένα δεδομένα του OpenStreetMap, το οποίο αυξάνει την ευχρηστία αλλά δυσκολεύει την επεκτασιμότητα σε άλλες πηγές δεδομένων,
- Τα δεδομένα εξόδου παρέχονται μόνο σε αρχείο με μορφότυπο N-Triples ή N-Quads. Δεν υπάρχει δυνατότητα για παροχή SPARQL endpoint.

3.2 Σχεδίαση συστήματος

Στο παρόν κεφάλαιο περιγράφουμε την σχεδίαση της εφαρμογής σε συμφωνία με τις απαιτήσεις συστήματος που προδιαγράφονται στο κεφάλαιο 3.1. Ξεκινώντας από την γενικότερη αρχιτεκτονική της εφαρμογής, παρουσιάζουμε τα συστατικά της μέρη και τις αντίστοιχες κλάσεις αυτών. Στη συνέχεια εξηγούμε τη λειτουργία κάθε κλάσης και τέλος παρουσιάζουμε το σχήμα της χρησιμοποιούμενης βάσης δεδομένων.

3.2.1 Αρχιτεκτονική



Σχήμα 3.2.1 (i) Geosm class diagram

Η εφαρμογή υλοποιείται από τα ακόλουθα κύρια υποσυστήματα:

- Core: Λειτουργεί ως σημείο εισόδου στην εφαρμογή και ως διαχειριστής όλων των υπολοίπων υποσυστημάτων. Είναι υπεύθυνο για τον ομαλό τερματισμό της εφαρμογής και την απελευθέρωση όλων των δεσμευμένων πόρων κατά τη διάρκεια τόσο ομαλών όσο και ανώμαλων σεναρίων διακοπής λειτουργίας. Αποτελείται από την κλάση *Geosm*.
- UX (User eXperience) Component: Αναλαμβάνει τη διαδραστικότητα με τον χρήστη μέσω περιβάλλοντος γραμμής εντολών ή γραφικού περιβάλλοντος και παρέχει τη δυνατότητα παραμετροποίησης της εφαρμογής. Αποτελείται από τις κλάσεις *Config*, *CLI* και *GeosmGUI*.
- Database Component: Αναλαμβάνει τη κατασκευή, αρχικοποίηση και καταστροφή της βάσης δεδομένων που χρησιμοποιείται εσωτερικά από την εφαρμογή, καθώς και για την σύνδεση και αποσύνδεση από αυτή. Αποτελείται από τις κλάσεις *DBManager*, *DBConnectionManager* και *ScriptRunner*.
- Osmosis Component: Αναλαμβάνει την διαχείριση της βιβλιοθήκης Osmosis, που χρησιμοποιείται για την ανάγνωση των OSM data dumps και την εγγραφή τους στη βάση δεδομένων. Αποτελείται από την κλάση *OsmosisFacade*.
- Model Component: Αναλαμβάνει την παραγωγή των RDF triples εξόδου σε N-TRIPLES ή N-QUADS μορφότυπο από τα δεδομένα που είναι αποθηκευμένα στη βάση δεδομένων. Για το σκοπό αυτό το component εκτελεί μία σειρά SQL επερωτήσεων πάνω στη βάση και στη συνέχεια εφαρμόζει ένα σύνολο απεικονίσεων πάνω στα result set αυτών ώστε να παραχθούν οι RDF triples. Αποτελείται από τις κλάσεις και interfaces *Tripleiser*, *TripleiserUtils*, *MappingsReader*, *TransliteratorELtoLATN*, *DefaultPostProcessor*, *URLEncodePostProcessor* και *PostProcessor*.
- *Vocabulary Component:* Αυτό το υποσύστημα που δεν εμφανίζεται στο ανωτέρω διάγραμμα κλάσεων αναλαμβάνει την υλοποίηση λεξιλογίων και τύπων δεδομένων RDF σε συμφωνία με την υλοποίηση των υπαρχόντων λεξιλογίων και τύπων δεδομένων στη βιβλιοθήκη Apache Jena. Συνδέεται με και αποτελεί λογικό τμήμα του Model Component και περιέχει τις εξής κλάσεις που υλοποιούν λεξιλόγια *GEO*, *LGD*, *IMIS*, *SF* και τις κλάσεις που υλοποιούν τύπους δεδομένων *GMLLiteral* και *WKTLiteral*.

Κατά τη φυσιολογική ροή του προγράμματος εκτελούνται σε επίπεδο component τα ακόλουθα βήματα:

1. Η εκτέλεση εκκινεί στο Core Component,
2. Καλείται το UX Component ώστε να συλλεχθούν οι επιθυμητές παράμετροι της εκτέλεσης,
3. Καλείται το DB Component το οποίο κατασκευάζει και αρχικοποιεί την βάση δεδομένων, και στη συνέχεια παρέχει σύνδεση προς αυτή,

4. Καλείται το Osmosis Component το οποίο διαβάζει το OSM data dump από το αρχείο εισόδου και το φορτώνει στη βάση δεδομένων,
5. Καλείται το Model Component το οποίο διαβάζει τα δεδομένα από την βάση, κατασκευάζει triples και τις στέλνει στο αρχείο εξόδου,
6. Τέλος εκτελείται η μέθοδος clean() που καλεί τα components Model και DB για να απελευθερωθούν ανοικτοί πόροι και να καταστραφεί η βάση δεδομένων.

3.2.2 Περιγραφή κλάσεων

Σε αυτό το κεφάλαιο θα παρουσιάσουμε και θα περιγράψουμε τη λειτουργία των κλάσεων του συστήματος οργανωμένες ανά υποσύστημα.

3.2.2.1 Core Component

3.2.2.1.1 Geosm

Η Geosm αποτελεί την μόνη κλάση του υποσυστήματος Core και ως εκ τούτου είναι υπεύθυνη για όλες τις λειτουργίες του. Αποτελεί το σημείο εισόδου στην εφαρμογή και είναι υπεύθυνη για για την ομαλή έναρξη και τον ομαλό τερματισμό αυτής κατά τη διάρκεια των ομαλών και ανώμαλων σεναρίων λειτουργίας. Η κλάση Geosm συνδέεται με όλα τα υπόλοιπα υποσυστήματα και είναι υπεύθυνη για την αρχικοποίησή τους και την επίκλησή τους με τα κατάλληλα δεδομένα στις διάφορες φάσεις εκτέλεσης.

3.2.2.2 UX Component

3.2.2.2.1 Config

Η κλάση Config λειτουργεί σαν κεντρική αποθήκη όλων των δεδομένων που είναι παραμετροποιήσιμα από τον χρήστη. Παρέχει μεθόδους για πρόσβαση και αλλαγές σε αυτές τις παραμέτρους και είναι επίσης υπεύθυνη για την ανάθεση προτερότιμων (default) τιμών σε όποιες παραμέτρους δεν έχουν τεθεί από το χρήστη. Αλλαγές των παραμέτρων γίνονται από από τις κλάσεις CLI και GeosmGUI για είσοδο από τον χρήστη μέσω του περιβάλλοντος γραμμής εντολών και του γραφικού περιβάλλοντος αντίστοιχα και πρόσβαση σε αυτές γίνεται μόνο μέσω της κλάσης Geosm για να ελαχιστοποιηθεί το coupling μεταξύ υποσυστημάτων.

3.2.2.2.2 CLI

Η κλάση CLI (Command Line Interface) είναι υπεύθυνη για το parsing των παραμέτρων γραμμής εντολών που παρέχονται στο πρόγραμμα κατά την εκκίνησή του. Αναλυτική περιγραφή των

παραμέτρων γραμμής εντολών που δέχεται το πρόγραμμα και της χρήσης τους μπορεί να βρεθεί στο κεφάλαιο 3.5.1. Η κλάση κάνει χρήση της βιβλιοθήκης Commons-CLI [CommonsCLI].

3.2.2.2.3 GeosmGUI

Η κλάση GeosmGUI είναι υπεύθυνη για την κατασκευή του γραφικού περιβάλλοντος εντολών και για την διαχείριση της εισόδου από τον χρήστη μέσω αυτού. Αναλυτική περιγραφή του τρόπου χρήσης του GUI μπορεί να βρεθεί στο κεφάλαιο 3.5.2.

3.2.2.3 Database Component

3.2.2.3.1 DBManager

Για την εκτέλεση του προγράμματος απαιτείται μία προσωρινή βάση δεδομένων PostgreSQL στην οποία θα μπορεί το Osmosis Component να φορτώσει τα δεδομένα του OSM. Αυτή η βάση θα πρέπει να έχει τα PostGIS και hstore extensions εγκατεστημένα και τα pgsnapshot v0.6 και pgsnapshot linestring v0.6 schemata φορτωμένα σε αυτή. Η κλάση DBManager παρέχει τρεις βασικές μεθόδους για την κατασκευή, αρχικοποίηση με τα προαναφερθέντα schemata και καταστροφή της απαιτούμενης βάσης δεδομένων. Για το σκοπό αυτό κάνει χρήση του PostgreSQL JDBC driver [PostgreSQLJDBC] και όσον αφορά την αρχικοποίηση της κλάσης ScriptRunner.

3.2.2.3.2 DBConnectionManager

Η κλάση DBConnectionManager είναι υπεύθυνη για τη σύνδεση στη και αποσύνδεση από τη βάση δεδομένων και παρέχει μεθόδους για τους σκοπούς αυτούς καθώς και για την επιστροφή αντικειμένου java.sql.Connection στη βάση. Κάνει χρήση του PostgreSQL JDBC driver [PostgreSQLJDBC].

3.2.2.3.3 ScriptRunner

Η μέθοδος ScriptRunner προέρχεται από την κλάση org.ibatis.jdbc.ScriptRunner από το MyBatis project [MyBatis]. Σκοπός της κλάσης είναι το parsing αρχείων .sql και η εκτέλεσή τους πάνω σε μία βάση. Στο πλαίσιο της εφαρμογής Geosm χρησιμοποιείται για την εφαρμογή των pgsnapshot v0.6 και pgsnapshot linestring v0.6 schemata πάνω στη βάση. Καλείται από την DBManager στη φάση της αρχικοποίησης.

3.2.2.4 Osmosis Component

3.2.2.4.1 OsmosisFacade

Η κλάση OsmosisFacade λειτουργεί ως facade για την βιβλιοθήκη Osmosis παρέχοντας στο Core Component ένα απλοποιημένο interface προς αυτή. Συγκεκριμένα παρέχει μία μέθοδο για την ανάγνωση των δεδομένων του OSM σε PBF ή OSM XML μορφότυπους και την εγγραφή τους στη βάση. Η βιβλιοθήκη Osmosis περιγράφεται αναλυτικά στο κεφάλαιο 2.3.1.

3.2.2.5 Model Component

3.2.2.5.1 Tripleiser

Η μέθοδος Tripleiser είναι το μόνο σημείο διεπαφής του Model Component με το Core Component και είναι υπεύθυνη για τη λειτουργία του Model Component συνολικά. Για την κατασκευή των N-TRIPLES και N-QUADS ακολουθούνται τα ακόλουθα βήματα:

Για κάθε έναν πίνακα στο pgsnapshot v0.6 schema (Nodes, Ways, Relations, Users, Way_Nodes, Relation_Members) εκτελούνται:

- Ένα SQL query πάνω στη βάση που επιστρέφει όλα τα δεδομένα του αντίστοιχου πίνακα, συμπεριλαμβανομένης της γεωχωρικής πληροφορίας σε δύο μορφότυπους WKT και GML.
- Καλείται η αντίστοιχη για τον πίνακα μέθοδος από την κλάση TripleiserUtils με παράμετρο το ResultSet που επέστρεψε η SQL query στο πρώτο βήμα. Αυτή η μέθοδος είναι υπεύθυνη για την παρασκευή όλων των σχετικών με τον πίνακα triples με βάση μια σειρά απεικονίσεων πάνω στα queries.

Εκτός αυτών η κλάση είναι υπεύθυνη για την αρχικοποίηση των TripleiserUtils και MappingsReader, καθώς και για την καταγραφή στο log της πορείας προόδου της επεξεργασίας συναρτήσει του συνολικού αριθμού εγγραφών στη βάση.

3.2.2.5.2 TripleiserUtils

Η κλάση TripleiserUtils παρέχει ένα πλήθος utilities (static methods) που χρησιμοποιούνται για την παρασκευή των N-TRIPLES ή N-QUADS. Αναλυτικά παρέχονται οι εξής βασικές λειτουργίες:

- Μεθόδους για τη δημιουργία και καταστροφή ενός sink προς το αρχείο εξόδου που θα δέχεται και θα εγγράφει ατομικά (individually) κάθε triple μετά την παρασκευή της,
- Μία βοηθητική μέθοδο για την παρασκευή και αποστολή στο sink μιας triple με παραμέτρους υποκείμενο, κατηγόρημα και αντικείμενο,

- Βοηθητικές μεθόδους για την παρασκευή ομάδων triples που είναι κοινές μεταξύ διαφορετικών αρχετύπων του OSM.
    - addGeometry: Η μέθοδος αυτή αναλαμβάνει την παρασκευή triples σχετικών με τη γεωμετρική πληροφορία. Συγκεκριμένα παράγει δύο ανώνυμους κόμβους με κλάσεις geo:Geometry που αναπαριστούν τις γεωμετρίες, με σειριοποιήσεις geo:asGML και geo:asWKT αντίστοιχα. Η γεωμετρία με την GML αναπαράσταση συνδέεται με το αρχέγονο με κατηγόρημα geo:defaultGeometry, ενώ η γεωμετρία με την WKT αναπαράσταση συνδέεται με κατηγόρημα geo:hasGeometry. Η επιλογή της GML αναπαράστασης ως default έχει γίνει λόγω της μεγαλύτερης ακρίβειας σε δεκαδικά ψηφία που προσφέρει. Τέλος και για τις δύο γεωμετρίες παράγονται rdf:type triples με αντικείμενα κλάσεις επιλεγμένες σύμφωνα με το RDFS Entailment Compatibility specification του OGC GeoSPARQL.
    - addUniversalMetadata: Η μέθοδος αυτή αναλαμβάνει την παρασκευή triples σχετικών με metadata που συνοδεύουν τα OSM αρχέγονα, όπως π.χ., τελευταία ημερομηνία επεξεργασίας,
    - addTags: Η μέθοδος αυτή αναλαμβάνει την παρασκευή triples από tags. Για το σκοπό αυτό κάνει χρήση ενός χάρτη απεικονίσεων που έχει παραχθεί από την κλάση MappingsReader. Επίσης, γίνεται χρήση των κλάσεων που υλοποιούν το PostProcessor interface όπου οι απεικονίσεις απαιτούν ειδική επεξεργασία. Τέλος, γίνεται χρήση της κλάσης TransliterateELtoLATN για την προσθήκη μιας rdfs:label[@lang='el-latn'] triple με μεταγραμματισμένο το ελληνικό όνομα σε όποιο αρχέγονο υπάρχει tag <name:el=*>. Περαιτέρω λεπτομέρειες σχετικές με την απεικόνιση των tags σε triples μπορούν να βρεθούν στο κεφάλαιο 2.3.3.
- Μία μέθοδο για κάθε πίνακα στο pgsnapshot v0.6 schema που αναλαμβάνει την παραγωγή όλων των triples σχετικών με τις στήλες αυτού του πίνακα. Αυτές είναι οι μέθοδοι που καλούνται από την κλάση Tripleiser για κάθε επερώτηση στη βάση.

Για την παρασκευή των IRIs που απαιτούνται για τις triples οι μέθοδοι της TripleiserUtils κάνουν χρήση των κλάσεων του Vocabulary Component.

3.2.2.5.3 MappingsReader

Η κλάση MappingsReader αναλαμβάνει την ανάγνωση των απεικονίσεων tags σε triples από το αρχείο Mappings και την εγγραφή τους σε έναν Map. Το αρχείο Mappings έχει παραχθεί από το αρχείο Mappings.sql του LinkedGeoData project. Περαιτέρω λεπτομέρειες σχετικές με τη δομή του μπορούν να βρεθούν στο κεφάλαιο 2.3.3.

3.2.2.5.4 TransliteratorELtoLATN

Η κλάση TransliteratorELtoLATN αναλαμβάνει την κατασκευή ενός χάρτη μεταγραμματισμού με βάση το αρχείο TransliterationSetELtoLATN και επίσης παρέχει μέθοδο που δέχεται κάποια συμβολοσειρά με Ελληνικούς χαρακτήρες και τη μεταγραμματίζει με χρήση Λατινικών. Περαιτέρω λεπτομέρειες για το μεταγραμματισμό μπορούν να βρεθούν στο κεφάλαιο 2.1.2.

3.2.2.5.5 PostProcessor

Η PostProcessor ορίζει ένα interface για την επεξεργασία ζευγών συμβολοσειρών prefix, value.

3.2.2.5.6 DefaultPostProcessor

Η κλάση DefaultPostProcessor παρέχει μέθοδο που ενώνει τις παραμέτρους prefix, value και επιστρέφει το αποτέλεσμα χωρίς περαιτέρω επεξεργασία.

3.2.2.5.7 URLEncodePostProcessor

Η κλάση URLEncodePostProcessor παρέχει μέθοδο που ενώνει τις παραμέτρους prefix, value από κωδικοποιήσει την τιμή value ως URL και επιστρέφει το αποτέλεσμα. Το prefix υποτίθεται ότι είναι έγκυρο URL, επομένως οι τιμή που επιστρέφεται είναι έγκυρο URL.

3.2.2.6 Vocabulary Component

3.2.2.6.1 GEO

Υλοποιεί το λεξιλόγιο geo: με uri <"http://www.opengis.net/ont/geosparql#">.

3.2.2.6.2 LGD

Υλοποιεί τα λεξιλόγια lgdt: και lgdo: με uri <"http://linkedgeodata.org/triplify/”> και <"http://linkedgeodata.org/ontology”>

3.2.2.6.3 IMIS

Υλοποιεί το λεξιλόγιο imis: με uri <"http://www.imis.athena-innovation.gr/ontology#">.

3.2.2.6.4 SF

Υλοποιεί το λεξιλόγιο sf: με uri <"http://www.opengis.net/ont/sf#">.

3.2.2.6.5 GMLLiteral

Υλοποιεί τον τύπο δεδομένων geo:gmlLiteral.

3.2.2.6.6 WKTLiteral

Υλοποιεί τον τύπο δεδομένων geo:wktLiteral.

3.2.3 Βάση δεδομένων

Η εφαρμογή απαιτεί την ύπαρξη βάσης δεδομένων PostgreSQL με τα PostGI και hstore extensions. Η βάση χρησιμοποιείται αρχικά για να φορτωθούν τα δεδομένα των data dumps του OSM από το Osmosis Component και στη συνέχεια για να γίνουν επερωτήσεις πάνω σε αυτά τα δεδομένα από το Model Component κατά την κατασκευή του RDF γράφου.

Για το σκοπό αυτό απαιτείται η βάση να έχει φορτωμένα τα schemata του OSM pgsnapshot v0.6 και pgsnapshot linestring v0.6 τα οποία διανέμονται με τη βιβλιοθήκη osmosis. Παρουσιάζουμε το μοντέλο οντοτήτων-συσχετίσεων για αυτά τα schemata.

3.2.3.1 Μοντέλο οντοτήτων συσχετίσεων

3.2.3.1.1 Οντότητες

- **schema_info:** Αντιπροσωπεύει την έκδοση του schema που χρησιμοποιείται. Περιέχει μία μόνο οντότητα. Έχει το ακόλουθο πεδίο:
  - **version** (int, primary key): Έκδοση του schema που χρησιμοποιείται.
- **users:** Αντιπροσωπεύει τους χρήστες του OSM. Έχει τα ακόλουθα πεδία:
  - **id** (int, primary key): Αναγνωριστικό χρήστη.
  - **name** (text): Όνομα χρήστη.
- **nodes:** Αντιπροσωπεύει τα nodes του OSM. Έχει τα ακόλουθα πεδία:
  - **id** (bigint, primary key): Αναγνωριστικό node.
  - **version** (int): Έκδοση node.
  - **user_id** (int): Κωδικός χρήστη υπεύθυνου για την τελευταία επεξεργασία του node.
  - **tstamp** (timestamp without time zone): Σφραγίδα χρόνου τελευταίας επεξεργασίας.
  - **changeset_id** (bigint): Κωδικός changeset τελευταίας επεξεργασίας.
  - **tags** (hstore): Πεδίο hstore που περιέχει συσχετισμένα με το node tags ως ζεύγη key=>value.

- **geom** (geometry): Πεδίο PostGIS για αποθήκευση ενός point με τη θέση του node. SRID 4326. Οι πίνακας έχει GiST ευρετήριο σε αυτό το πεδίο και οργανώνεται σε χωρικά clusters με βάση τη γεωγραφική θέση.

- **ways:** Αντιπροσωπεύει τα ways του OSM. Έχει τα ακόλουθα πεδία:
  - **id** (bigint, primary key): Αναγνωριστικό way.
  - **version** (int): Έκδοση way.
  - **user_id** (int): Κωδικός χρήστη υπεύθυνου για την τελευταία επεξεργασία του way.
  - **tstamp** (timestamp without time zone): Σφραγίδα χρόνου τελευταίας επεξεργασίας.
  - **changeset_id** (bigint): Κωδικός changeset τελευταίας επεξεργασίας.
  - **tags** (hstore): Πεδίο hstore που περιέχει συσχετισμένα με το way tags ως ζεύγη key=>value.
  - **linestring** (geometry): Πεδίο PostGIS για αποθήκευση ενός linestring με τη γεωμετρία του way. SRID 4326. Οι πίνακας έχει GiST ευρετήριο σε αυτό το πεδίο και οργανώνεται σε χωρικά clusters με βάση τη γεωγραφική θέση.
- **relations:** Αντιπροσωπεύει τα relations του OSM. Έχει τα ακόλουθα πεδία:
  - **id** (bigint, primary key): Αναγνωριστικό relation.
  - **version** (int): Έκδοση relation.
  - **user_id** (int): Κωδικός χρήστη υπεύθυνου για την τελευταία επεξεργασία του relation.
  - **tstamp** (timestamp without time zone): Σφραγίδα χρόνου τελευταίας επεξεργασίας.
  - **changeset_id** (bigint): Κωδικός changeset τελευταίας επεξεργασίας.
  - **tags** (hstore): Πεδίο hstore που περιέχει συσχετισμένα με το relation tags ως ζεύγη key=>value.
- **way_nodes:** Αντιπροσωπεύει τις σχέσεις μεταξύ nodes και ways. Έχει τα ακόλουθα πεδία:
  - **way_id** (bigint, primary key): Αναγνωριστικό way.
  - **node_id** (bigint): Αναγνωριστικό node μέλους του way με αναγνωριστικό way_id. Ο πίνακας έχει B-tree ευρετήριο σε αυτό το πεδίο.
  - **sequence_id** (int, primary key): Θέση του node με αναγνωριστικό node_id στην αλληλουχία nodes που συνθέτει το way με αναγνωριστικό way_id.
- **relation_members:** Αντιπροσωπεύει τις σχέσεις μεταξύ nodes/ways και relations. Έχει τα ακόλουθα πεδία:
  - **relation_id** (bigint, primary key): Αναγνωριστικό relation.

- **member_id** (bigint): Αναγνωριστικό μέλους (node, way ή relation). Ο πίνακας έχει B-tree ευρετήριο σε αυτό το πεδία σε συνδυασμό με το member_type πεδίο.
- **member_type** (character(1)): Τύπος μέλους.
  - 'N': για μέλος τύπου node
  - 'W': για μέλος τύπου way
  - 'R': για μέλος τύπου relation

    Ο πίνακας έχει B-tree ευρετήριο σε αυτό το πεδίο σε συνδυασμό με το member_id πεδίο.
- **member_role** (text): Ρόλος μέλους.
- **sequence_id** (int, primary key): Θέση μέλους με αναγνωριστικό member_id και τύπο member_type στην αλληλουχία που συνθέτει το relation με αναγνωριστικό relation_id.

3.2.3.1.2 Συσχετίσεις

- **last_edited_by:** συνδέει τις οντότητες nodes, ways, relations στο πεδίο user_id με την οντότητα users στο πεδίο id. Δείχνει τον τελευταίο χρήστη που επεξεργάστηκε την αντίστοιχη οντότητα.
- **node_member_of_way:** συνδέει τις οντότητες nodes και ways μέσω της οντότητας way_nodes. Δείχνει ότι μία οντότητα node είναι μέλος με συγκεκριμένη θέση στην αλληλουχία οντοτήτων nodes που συνθέτουν ένα way.
- **is_member_of_relation:** συνδέει τις οντότητες nodes, ways, relations με την οντότητα relations μέσω την οντότητας relation_members. Δείχνει ότι μία οντότητα node, way, relation είναι μέλος με συγκεκριμένη θέση και ρόλο στην αλληλουχία οντοτήτων που συνθέτουν ένα relation.

3.2.4 Κωδικοποίηση αρχείων

Από την εφαρμογή χρησιμοποιούνται οι ακόλουθοι μορφότυποι αρχείων.

3.2.4.1 OSM XML (.osm)

Ο μορφότυπος OSM XML [OSM/XML] είναι ένας εκ των δύο μορφοτύπων για τα data dumps του OSM που μπορεί να επεξεργαστεί η εφαρμογή Geosm. Ο μορφότυπος OSM XML είναι ένα έγκυρο αρχείο XML που ακολουθεί το OSM XML Schema.

Η δομή του είναι η ακόλουθη:

- Ένα XML suffix που δηλώνει την UTF-8 κωδικοποίηση του αρχείου.

- Ένα στοιχείο OSM που περιέχει την έκδοση του API και την εφαρμογή που παρήγαγε το αρχείο.
    - Ένα node block που περιέχει πληροφορίες για τα nodes συμπεριλαμβανόμενης της θέσης τους σε WGS84.
        - Τα tags για κάθε node.
    - Ένα way block που περιέχει πληροφορίες για τα ways.
        - Αναφορές προς τα nodes που το συνθέτουν για κάθε way.
        - Τα tags για κάθε way.
    - Ένα relation block που περιέχει πληροφορίες για relations.
        - Αναφορές προς τα μέλη που το συνθέτουν για κάθε relation.
        - Τα tags για κάθε relation.

Ενδεικτικά παρουσιάζουμε μια συντομευμένη έκδοση ενός αρχείου δείγματος σε μορφότυπο OSM XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<osm version="0.6" generator="CGImap 0.0.2">
 <bounds minlat="54.0889580" minlon="12.2487570" maxlat="54.0913900"
maxlon="12.2524800"/>
 <node id="298884269" lat="54.0901746" lon="12.2482632" user="SvenHRO" uid="46882"
visible="true" version="1" changeset="676636" timestamp="2008-09-21T21:37:45Z"/>
 <node id="261728686" lat="54.0906309" lon="12.2441924" user="PikoWinter"
uid="36744" visible="true" version="1" changeset="323878" timestamp="2008-05-
03T13:39:23Z"/>
 <node id="1831881213" version="1" changeset="12370172" lat="54.0900666"
lon="12.2539381" user="lafkor" uid="75625" visible="true" timestamp="2012-07-
20T09:43:19Z">
  <tag k="name" v="Neu Broderstorf"/>
  <tag k="traffic_sign" v="city_limit"/>
 </node>
 ...
 <node id="298884272" lat="54.0901447" lon="12.2516513" user="SvenHRO" uid="46882"
visible="true" version="1" changeset="676636" timestamp="2008-09-21T21:37:45Z"/>
 <way id="26659127" user="Masch" uid="55988" visible="true" version="5"
changeset="4142606" timestamp="2010-03-16T11:47:08Z">
  <nd ref="292403538"/>
  <nd ref="298884289"/>
  ...
  <nd ref="261728686"/>
```

```
  <tag k="highway" v="unclassified"/>
  <tag k="name" v="Pastower Straße"/>
 </way>
 <relation id="56688" user="kmvar" uid="56190" visible="true" version="28"
changeset="6947637" timestamp="2011-01-12T14:23:49Z">
  <member type="node" ref="294942404" role=""/>
  ...
  <member type="node" ref="364933006" role=""/>
  <member type="way" ref="4579143" role=""/>
  ...
  <member type="node" ref="249673494" role=""/>
  <tag k="name" v="Küstenbus Linie 123"/>
  <tag k="network" v="VVW"/>
  <tag k="operator" v="Regionalverkehr Küste"/>
  <tag k="ref" v="123"/>
  <tag k="route" v="bus"/>
  <tag k="type" v="route"/>
 </relation>
 ...
</osm>
```

3.2.4.2 PBF (.osm.pbf)

Ο μορφότυπος Protocolbuffer Binary Format (PBF) [PBF] χρησιμοποιείται από το OSM, πρωτίστως ως αντικαταστάτης του OSM XML. Ένα αρχείο planet data dump σε PBF έχει περίπου το μισό μέγεθος σε σχέση με την ίδια πληροφορία σε αρχείο OSM XML συμπιεσμένο με gzip και είναι 30% μικρότερο από ένα αρχείο OSM XML συμπιεσμένο με bzip. Είναι επίσης 5x γρηγορότερο στην εγγραφή και 6x γρηγορότερο στην ανάγνωση σε σχέση με το ανωτέρω gzip αρχείο.

Ο υποκείμενος μορφότυπος αρχείου (file format) επιλέχθηκε ώστε να υποστηρίζει τυχαία πρόσβαση σε 'fileblock' κοκκιότητα (granularity). Κάθε fileblock μπορεί να αποκωδικοποιηθεί ανεξάρτητα και περιέχει ~8k οντότητες OSM στην προτερότιμη διάρθρωση (default configuration). Για την αποθήκευση χαμηλού επιπέδου (low level store) χρησιμοποιούνται Google Protocol buffers.

Ένα αρχείο περιέχει μία επικεφαλίδα ακολουθούμενη από μία αλληλουχία fileblocks. Αυτή η σχεδίαση έχει σκοπό να επιτρέπει μελλοντική τυχαία πρόσβαση στα περιεχόμενα του αρχείου και σε μία εφαρμογή να προσπερνάει δεδομένα που δεν επιθυμεί ή δεν καταλαβαίνει.

Ο μορφότυπος είναι μια επαναλαμβανόμενη αλληλουχία από:

- int4: το μήκος του μηνύματος BlobHeader σε network byte order
- σειριοποιημένο το μήνυμα BlobHeader
- σειριοποιημένο το μήνυμα Blob (το μέγεθός του δίνεται στην επικεφαλίδα)\

Ένα BlobHeader ορίζεται ως:

```
message BlobHeader {
  required string type = 1;
  optional bytes indexdata = 2;
  required int32 datasize = 3;
}
```

- **type** περιέχει τον τύπο των δεδομένων στο μήνυμα
- **indexdata** σε κάποιο αυθαίρετο blob μπορεί να περιέχει metadata για το επόμενο blob (π.χ., για δεδομένα OSM μπορεί να περιέχει ένα bounding box). Αυτό το πεδίο υπάρχει για να επιτρέψει τη μελλοντική σχεδίαση αρχείων *.osm.pbf με ευρετήρια.
- **datasize** περιέχει το σειροποιημένο μέγεθος του μηνύματος Blob που ακολουθεί

Την στιγμή της εγγραφής του παρόντος το Blob χρησιμοποιείται για να αποθηκεύσει ένα αυθαίρετο κομμάτι δεδομένων είτε ασυμπίεστα, είτε σε ένα zlib/deflate συμπιεσμένο μορφότυπο.

```
message Blob {
   optional bytes raw = 1; // No compression
   optional int32 raw_size = 2; // Only set when compressed, to the
uncompressed size
   optional bytes zlib_data = 3;
   // optional bytes lzma_data = 4; // PROPOSED.
   // optional bytes OBSOLETE_bzip2_data = 5; // Deprecated.
}
```

3.2.4.3 N-Triples / N-Quads (.nt)

Ο μορφότυπος N-Triples [NT] είναι ένας line-based, απλού κειμένου μορφότυπος σειριοποίησης για γράφους RDF και υποσύνολο του μορφότυπου TURTLE. Ο μορφότυπος N-Triples σχεδιάστηκε ως ένα απλούστερο των μορφοτύπων N3 και TURTLE και επομένως ευκολότερος για εγγραφή και ανάγνωση από λογισμικό. Λόγω της απουσίας συντομεύσεων που υπάρχουν σε άλλους μορφότυπους είναι κοπιαστικό να γραφτούν μεγάλες ποσότητες δεδομένων σε N-Triples με το χέρι και δύσκολο να διαβαστούν.

Κάθε γραμμή ενός αρχείου N-Triples αναπαριστά μία πρόταση RDF αποτελούμενη από υποκείμενο, κατηγόρημα και αντικείμενο χωρισμένα με whitespaces και τερματίζεται με μία τελεία.

Το υποκείμενο μπορεί να είναι ένα URI ή ένας κενός κόμβος, το κατηγόρημα πρέπει να είναι ένα URI και το αντικείμενο μπορεί να είναι ένα URI, ένας κενός κόμβος ή ένα literal. Τα URIs οριοθετούνται από τους χαρακτήρες 'μικρότερο από', 'μεγαλύτερο από'. Οι κενοί κόμβοι αναπαριστώνται από μία αλφαριθμητική συμβολοσειρά, με το πρόθεμα '_:'. Τα literals

αναπαριστώνται ως printable χαρακτήρες ASCII (με '\' ως escape χαρακτήρα), οροθετημένοι με "" και προαιρετικά με επίθεμα που δηλώνει τη γλώσσα ή τον τύπο δεδομένων τους. Ένα επίθεμα που προσδιορίζει τη γλώσσα αποτελείται από το σύμβολο '@' ακολουθούμενο από ένα RFC 3066 language tag. Ένα επίθεμα τύπου δεδομένων αποτελείται από το σύμβολο '^' ακολουθούμενο από ένα URI. Τα σχόλια είναι γραμμές που ξεκινούν με το σύμβολο '#'.

Ο μορφότυπος N-Quads επεκτείνει τον μορφότυπο N-Triples με την προσθήκη ενός προαιρετικού context URI μετά το υποκείμενο.

Ακολουθεί ένα παράδειγμα αρχείου σε μορφότυπο N-Triples:

```
<http://linkedgeodata.org/triplify/node162150121>
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://linkedgeodata.org/ontology/node> .
<http://linkedgeodata.org/triplify/node162150121>
<http://purl.org/dc/elements/1.1/identifier>
"162150121"^^<http://www.w3.org/2001/XMLSchema#long> .
<http://linkedgeodata.org/triplify/node162150121>
<http://linkedgeodata.org/ontology/version> "3" .
<http://linkedgeodata.org/triplify/node162150121>
<http://linkedgeodata.org/ontology/contributor> "66120" .
<http://linkedgeodata.org/triplify/node162150121>
<http://linkedgeodata.org/ontology/modificationDate> "2008-01-
27T10:40:31.0"^^<http://www.w3.org/2001/XMLSchema#dateTime> .
<http://linkedgeodata.org/triplify/node162150121>
<http://linkedgeodata.org/ontology/changeset> "705607" .
_:BX2D26832e22X3A13f6bfba711X3AX2D7fff <http://www.w3.org/1999/02/22-rdf-
syntax-ns#type> <http://www.opengis.net/ont/geosparql#Geometry> .
<http://linkedgeodata.org/triplify/node162150121>
<http://www.opengis.net/ont/geosparql#defaultGeometry>
_:BX2D26832e22X3A13f6bfba711X3AX2D7fff .
_:BX2D26832e22X3A13f6bfba711X3AX2D7fff <http://www.w3.org/1999/02/22-rdf-
syntax-ns#type> <http://www.opengis.net/ont/sf#Point> .
_:BX2D26832e22X3A13f6bfba711X3AX2D7fff
<http://www.opengis.net/ont/geosparql#asGML> "<gml:Point
srsName=\"EPSG:4326\"><gml:coordinates>20.8579419,39.610261000000001</gml:c
oordinates></gml:Point>"^^<http://www.opengis.net/ont/geosparql#gmlLiteral>
.
```

3.2.4.4 Java properties (.properties)

Ο μορφότυπος .properties είναι ένας μορφότυπος απλού κειμένου που χρησιμοποιείται για την αποθήκευση παραμέτρων μιας εφαρμογής ή για την αποθήκευση συμβολοσειρών που χρησιμοποιούνται στην διεθνοποίηση κάποιας εφαρμογής.

Κάθε παράμετρος αποθηκεύεται ως ένα ζεύγος συμβολοσειρών. Η πρώτη συμβολοσειρά καλείται key και είναι το όνομα της παραμέτρου και η άλλη καλείται value και είναι η τιμή της. Τα ζεύγη key, value μπορούν να αναπαρασταθούν με πολλαπλούς τρόπους, συμπεριλαμβανομένων 'key=value', 'key = value', 'key:value' και 'key value'. Κάθε γραμμή στο αρχείο περιέχει μία μόνο παράμετρο. Μία γραμμή που ξεκινάει με τα σύμβολα '#' ή '!' ως το πρώτο μη κενό σύμβολο της αντιμετωπίζεται ως σχόλιο. Ο χαρακτήρας '\' χρησιμοποιείται ως escape χαρακτήρας.

Ακολουθεί παράδειγμα αρχείου .properties:

```
# Input and output file paths (source must end in '.osm' or '.pbf' for
RDF/XML and PBF respectively)
input=in.osm
output=out.nt

# Database config
dbname=osm
user=psql
password=psql

max_rows_to_cache=100000

# Model context for N-Quads (if blank N-Triples will be printed)
context=http://www.example.org
```

3.3 Υλοποίηση

Στα προηγούμενα κεφάλαια παρουσιάσαμε τις προδιαγραφές του συστήματος και τη γενικότερη σχεδίασή του. Σε αυτό το τμήμα θα παρουσιάσουμε τις λεπτομέρειες της υλοποίησης, εμβαθύνοντας σε εκείνα τα σημεία του κώδικα που παρουσιάζουν το μεγαλύτερο ενδιαφέρον. Στη συνέχεια θα παρουσιάσουμε τα προγραμματιστικά εργαλεία που χρησιμοποιήθηκαν κατά την διάρκεια της ανάπτυξης, τις απαιτήσεις της εφαρμογής σε υπολογιστικούς πόρους και βιβλιοθήκες και τέλος θα παρουσιάσουμε την διαδικασία εγκατάστασης της εφαρμογής.

3.3.1 Λεπτομέρειες υλοποίησης

3.3.1.1 Database Creation & Initialisation

Όπως έχει περιγραφεί στα ανωτέρω κεφάλαια για τις ανάγκες της εφαρμογής απαιτείται μία βάση δεδομένων PostgreSQL με τα PostGIS και hstore extensions και με τα schemata pgsnapshot v0.6 και pgsnapshot linestring v0.6 εγκατεστημένα.

Η κατασκευή και αρχικοποίηση της βάσης είναι ευθύνη της κλάσης DBManager του Database Component και συγκεκριμένα γίνεται μέσω των μεθόδων DBManager#createDB() και DBManager#initialiseDB(java.sql.Connection db) αντίστοιχα. Αναλυτικά:

Στη μέθοδο DBManager#createDB() η κατασκευή της βάσης γίνεται με τη σύνδεση στο DBMS μέσω του PostgreSQL JDBC driver και την εκτέλεση των ακόλουθων εντολών.

```
DROP DATABASE IF EXISTS <dbname>
CREATE DATABASE <dbname>
```

Στη μέθοδο DBManager#initialiseDB(java.sql.Connection db) η αρχικοποίηση της βάσης γίνεται ως εξής. Αρχικά παρέχεται στη μέθοδο ένα αντικείμενο java.sql.Connection db που παρέχει τη σύνδεση στην βάση. Στη συνέχεια αρχικοποιείται ένα στιγμιότυπο της κλάσης ScriptRunner με παράμετρο το αντικείμενο db. Σκοπός της ScriptRunner είναι η ανάγνωση από αρχείο και στη συνέχεια εκτέλεση σε βάση ενός sql script. Αξίζει να σημειωθεί ότι μέσω της μεθόδου ScriptRunner#setSendFullScript(Boolean sendFullScript) ενεργοποιείται η δυνατότητα της ScriptRunner να εκτελεί το SQL script συνολικά στην βάση, αντί να στέλνει τις εντολές του script μία τη φορά. Αυτή η επιλογή έχει γίνει ώστε να αποφευχθούν προβλήματα με την αναγνώριση της συμβολοσειράς '$$' που εμφανίζεται στα script. Τέλος με χρήση της μεθόδου ScriptRunner#runScript(Reader reader) εκτελούνται τα ακόλουθα SQL scripts:

extensions.sql

```
create extension hstore;
create extension postgis;
create extension postgis_topology;
```

pgsnapshot_schema_0.6

```
-- Database creation script for the simple PostgreSQL schema.

-- Drop all tables if they exist.
DROP TABLE IF EXISTS actions;
DROP TABLE IF EXISTS users;
DROP TABLE IF EXISTS nodes;
DROP TABLE IF EXISTS ways;
DROP TABLE IF EXISTS way_nodes;
```

```
DROP TABLE IF EXISTS relations;
DROP TABLE IF EXISTS relation_members;
DROP TABLE IF EXISTS schema_info;

-- Drop all stored procedures if they exist.
DROP FUNCTION IF EXISTS osmosisUpdate();


-- Create a table which will contain a single row defining the current
schema version.
CREATE TABLE schema_info (
    version integer NOT NULL
);


-- Create a table for users.
CREATE TABLE users (
    id int NOT NULL,
    name text NOT NULL
);


-- Create a table for nodes.
CREATE TABLE nodes (
    id bigint NOT NULL,
    version int NOT NULL,
    user_id int NOT NULL,
    tstamp timestamp without time zone NOT NULL,
    changeset_id bigint NOT NULL,
    tags hstore
);
-- Add a postgis point column holding the location of the node.
SELECT AddGeometryColumn('nodes', 'geom', 4326, 'POINT', 2);


-- Create a table for ways.
CREATE TABLE ways (
    id bigint NOT NULL,
    version int NOT NULL,
    user_id int NOT NULL,
```

```
    tstamp timestamp without time zone NOT NULL,
    changeset_id bigint NOT NULL,
    tags hstore,
    nodes bigint[]
);


-- Create a table for representing way to node relationships.
CREATE TABLE way_nodes (
    way_id bigint NOT NULL,
    node_id bigint NOT NULL,
    sequence_id int NOT NULL
);


-- Create a table for relations.
CREATE TABLE relations (
    id bigint NOT NULL,
    version int NOT NULL,
    user_id int NOT NULL,
    tstamp timestamp without time zone NOT NULL,
    changeset_id bigint NOT NULL,
    tags hstore
);

-- Create a table for representing relation member relationships.
CREATE TABLE relation_members (
    relation_id bigint NOT NULL,
    member_id bigint NOT NULL,
    member_type character(1) NOT NULL,
    member_role text NOT NULL,
    sequence_id int NOT NULL
);


-- Configure the schema version.
INSERT INTO schema_info (version) VALUES (6);


-- Add primary keys to tables.
```

```
ALTER TABLE ONLY schema_info ADD CONSTRAINT pk_schema_info PRIMARY KEY
(version);

ALTER TABLE ONLY users ADD CONSTRAINT pk_users PRIMARY KEY (id);

ALTER TABLE ONLY nodes ADD CONSTRAINT pk_nodes PRIMARY KEY (id);

ALTER TABLE ONLY ways ADD CONSTRAINT pk_ways PRIMARY KEY (id);

ALTER TABLE ONLY way_nodes ADD CONSTRAINT pk_way_nodes PRIMARY KEY (way_id,
sequence_id);

ALTER TABLE ONLY relations ADD CONSTRAINT pk_relations PRIMARY KEY (id);

ALTER TABLE ONLY relation_members ADD CONSTRAINT pk_relation_members
PRIMARY KEY (relation_id, sequence_id);


-- Add indexes to tables.
CREATE INDEX idx_nodes_geom ON nodes USING gist (geom);

CREATE INDEX idx_way_nodes_node_id ON way_nodes USING btree (node_id);

CREATE INDEX idx_relation_members_member_id_and_type ON relation_members
USING btree (member_id, member_type);


-- Cluster tables by geographical location.
CLUSTER nodes USING idx_nodes_geom;


-- Create the function that provides "unnest" functionality while remaining
compatible with 8.3.
CREATE OR REPLACE FUNCTION unnest_bbox_way_nodes() RETURNS void AS $$
DECLARE
      previousId ways.id%TYPE;
      currentId ways.id%TYPE;
      result bigint[];
      wayNodeRow way_nodes%ROWTYPE;
      wayNodes ways.nodes%TYPE;
```

```
BEGIN
      FOR wayNodes IN SELECT bw.nodes FROM bbox_ways bw LOOP
            FOR i IN 1 .. array_upper(wayNodes, 1) LOOP
                  INSERT INTO bbox_way_nodes (id) VALUES (wayNodes[i]);
            END LOOP;
      END LOOP;
END;
$$ LANGUAGE plpgsql;


-- Create customisable hook function that is called within the replication
update transaction.
CREATE FUNCTION osmosisUpdate() RETURNS void AS $$
DECLARE
BEGIN
END;
$$ LANGUAGE plpgsql;
```

pgsnapshot_schema_0.6_linestring

```
-- Add a postgis GEOMETRY column to the way table for the purpose of
storing the full linestring of the way.
SELECT AddGeometryColumn('ways', 'linestring', 4326, 'GEOMETRY', 2);

-- Add an index to the bbox column.
CREATE INDEX idx_ways_linestring ON ways USING gist (linestring);

-- Cluster table by geographical location.
CLUSTER ways USING idx_ways_linestring;
```

3.3.1.2 Osmosis

Μέσω της μεθόδου OsmosisFacade#executeOsmosis() εκτελείται μία σειρά εντολών σε pipeline της Osmosis που σκοπό έχουν την ανάγνωση του αρχείου εισόδου και την εγγραφή του στη βάση. Αρχικά, γίνεται απόπειρά να αναγνωριστεί ο μορφότυπος του αρχείου μεταξύ PBF και OSM XML μέσω regex matching του file extension και αντίστοιχα επιλέγεται ο κατάλληλος reader μεταξύ των:

```
--read-xml-0.6
--read-pbf-0.6
```

Στη συνέχεια εκτελείται ο ακόλουθος κώδικας:

```
String[] parameters = new String[] {
            "-q",
            reader, inputFile,
```

```
            "--write-pgsql-0.6",
            "database=" + dbName,
            "user=" + dbUsername,
            "password=" + dbPassword };

Osmosis.run(parameters);
```

Για λόγους αναφοράς με, για παράδειγμα, αρχείο εισόδου OSM XML ο ανωτέρω κώδικας θα ήταν το ισοδύναμο της εκτέλεσης της ακόλουθης εντολής osmosis στη γραμμή εντολών:

```
omsosis -q --read-xml-0.6 <inputFile> --write-pgsql-0.6 database=<dbname>
user=<dbUsername> password=<dbPassword>
```

3.3.1.3 Triple Production

Στη βάση της διαδικασίας παραγωγής των triples στην κλάση TripleiserUtils βρίσκονται ένα αντικείμενο org.apache.jena.atlas.lib.Sink<T> sink και η μέθοδος TripleiserUtils#createAndSendSerialisation(Node subject, Node predicate, Node object).

Το αντικείμενο sink είναι στιγμιότυπο μίας εκ των κλάσεων SinkTripleOutput(OutputStream outs) και SinkQuadOutput(OutputStream outs) του package org.apache.jena.riot.out, που υλοποιούν το interface Sink<T> για παραγωγή N-Triples και N-Quads αντίστοιχα.

Η μέθοδος createAndSendSerialisation δέχεται τρία αντικείμενα com.hp.hpl.jena.graph.Node που αντιπροσωπεύουν το υποκείμενο, κατηγόρημα και αντικείμενο αντίστοιχα, παρασκευάζει την αντίστοιχη Triple η Quad και την στέλνει στο sink το οποίο αναλαμβάνει την εγγραφή της στο αρχείο εξόδου.

Για παράδειγμα για την παραγωγή μιας triple με υποκείμενο έναν ανώνυμο κόμβο, κατηγόρημα rdf:type και υποκείμενο geo:Geometry και στη συνέχεια μιας triple από τον ίδιο κόμβο με κατηγόρημα geo:asGML και υποκείμενο ένα literal με τιμή το περιεχόμενο της μεταβλητής geom και datatype geo:GMLLiteral θα είχαμε τον ακόλουθο κώδικα:

```
Node anon = Node.createAnon();
createAndSendSerialisation(anon, Node.createURI(RDF.type.toString()),
Node.createURI(GEO.geometry));
createAndSendSerialisation(anon, Node.createURI(GEO.asGML.toString()),
Node.createLiteral(geom, GEO.gmlLiteral));
```

3.3.1.4 Mappings

Τα mappings όπως έχει αναφερθεί καθορίζουν τον τρόπο μετατροπής των tags που συνοδεύουν τα OSM αρχέγονα σε RDF triples. Σε αυτό το κεφάλαιο θα περιγράψουμε την μορφή των mappings στο αρχείο Mappings και τον τρόπο με τον οποίο εφαρμόζονται κατά τη διάρκεια της παραγωγής των triples για κάθε κατηγορία mapping ξεχωριστά.

- @datatype

  Τα mappings αυτού του τύπου έχουν πάντα δύο τιμές, το key ενός tag και τον τύπο δεδομένων που πρέπει να έχει το αντίστοιχο value. Επειδή αυτά τα mappings έχουν πάντα αντίστοιχο mapping και στην κατηγορία @property η χρήση τους θα περιγραφεί συγκεντρωτικά στο @property.

- @literal

  Τα mappings αυτού του τύπου μπορούν να έχουν δύο ή τρεις τιμές. Ένα tag key, ένα κατηγόρημα και ένα προαιρετικό language tag. Το αντίστοιχο triple που παράγεται θα έχει το δοσμένο κατηγόρημα και ως αντικείμενο ένα literal με τιμή την tag value και με ένα language tag postfix αν υπήρχε στο mapping.

- @k,@kv

  Τα mappings αυτών των τύπων έχουν τρεις τιμές (tag key, κατηγόρημα, αντικείμενο) στην περίπτωση του @k ή τέσσερις τιμές (tag key, tag value, κατηγόρημα, αντικείμενο). Σε κάθε περίπτωση το αντίστοιχο triple που παράγεται θα έχει το δοσμένο κατηγόρημα και το δοσμένο αντικείμενο. Η διαφορά έγκειται στο ότι ένα @kv mapping είναι περισσότερο συγκεκριμένο και πρέπει να χρησιμοποιηθεί αντί ενός @k mapping όπου υπάρχει.

- @property

  Τα mappings αυτού του τύπου έχουν πάντα δύο τιμές. Ένα tag key και ένα κατηγόρημα. Το παραγόμενο triple θα έχει το δοσμένο κατηγόρημα και για αντικείμενο ένα literal με τιμή την tag value. Αν έχει βρεθεί και @datatype mapping για το συγκεκριμένο tag key θα προστεθεί στο αντικείμενο ένα postfix με το αντίστοιχο datatype URI.

- @prefix

  Τα mappings αυτού του τύπου έχουν πάντα τέσσερις τιμές. Ένα tag key, ένα κατηγόρημα, ένα URI που θα αποτελέσει το πρόθεμα του αντικειμένου και μία λέξη κλειδί που καθορίζει πως θα σχηματιστεί το αντικείμενο από το πρόθεμα και την tag value. Ο σχηματισμός του αντικειμένου γίνεται από κλάσεις που υλοποιούν το interface PostProcessor και αυτή τη στιγμή υποστηρίζονται δύο post processors, ο DefaultPostProcessor που απλώς ενώνει τα prefix και tag value και ο URLEncodePostProcessor που πρώτα κάνει URL encode την tag value και στη συνέχεια την ενώνει με το prefix ώστε να εξασφαλίσει ότι το τελικό αποτέλεσμα θα είναι ένα έγκυρο URL. Αντίστοιχα με τα προηγούμενα το triple παράγεται από το δοσμένο κατηγόρημα και από το προκύπτων αντικείμενο.

Οι διεργασίες που περιγράφονται σε αυτό το κεφάλαιο γίνονται στην κλάση MappingsReader και στη μέθοδο TripleiserUtils#addTags(Node node, String tags[]). Η μεταφορά των mappings μεταξύ του MappingsReader και του TripleiserUtils γίνεται με χρήση ενός java.util.HashMap<String, String[]>.

3.3.2 Πλατφόρμες και προγραμματιστικά εργαλεία

3.3.2.1 Προγραμματιστικά εργαλεία

Η ανάπτυξη της εφαρμογής έγινε σε περιβάλλον Ubuntu 12.04 LTS, με χρήση του Java 7 OpenJDK AMD64 που αντιστοιχεί στην έκδοση 1.7 του επίσημου Sun Java JDK, της έκδοσης 9.1 του PostgreSQL DBMS και της έκδοσης 2.0 του PostGIS.

Η εφαρμογή αναπτύχθηκε ως Ant project πάνω στο Netbeans IDE v7.3.

3.3.2.2 Απαιτήσεις

Για την εγκατάσταση και σωστή λειτουργία της η εφαρμογή Geosm απαιτεί να υπάρχουν εγκατεστημένα το PostgreSQL DBMS με το PostGIS extension και το JRE v1.7 ή νεότερο. Όσων αφορά τις δυνατότητες του συστήματος η εφαρμογή δεν έχει ιδιαίτερες απαιτήσεις και μπορεί να ρυθμιστεί ως προς την διαθέσιμη RAM του συστήματος με κατάλληλη επιλογή της τιμής 'Max rows to cache'. Το bottleneck του συστήματος είναι οι ταχύτητες εγγραφής και ανάγνωσης στο δίσκο και η ταχύτητα εκτέλεσης του προγράμματος μπορεί να ποικίλει έντονα ανάλογα με το διαθέσιμο I/O.

3.3.2.3 Εγκατάσταση

Η εφαρμογή διανέμεται με τη μορφή ενός συμπιεσμένου .tar.gz αρχείου που περιέχει όλα τα απαραίτητα αρχεία. Η εγκατάσταση απαιτεί μόνο την αποσυμπίεση του αρχείου .tar.gz σε ένα περιβάλλον με τα κατάλληλα JRE, PostgreSQL και PostGIS εγκατεστημένα.

3.4 Έλεγχος

Ο έλεγχος της εφαρμογής έχει γίνει σε δύο επίπεδα, πρώτον μέσω unit testing χρησιμοποιώντας τη JUnit unit testing library, όπου ελέγχθηκε η λειτουργία επιμέρους κλάσεων της εφαρμογής και δεύτερον μέσω χρήσης της εφαρμογής πάνω σε πραγματικά δεδομένα όπου ελέγχθηκε η γενικότερη ορθότητα του output καθώς και οι επιδόσεις και κατανάλωση πόρων συστήματος της εφαρμογής σε τυπική λειτουργία.

3.5 Εγχειρίδιο χρήσης

Η εφαρμογή geosm διανέμεται με τη μορφή συμπιεσμένου αρχείου .tar.gz, το οποίο περιέχει ένα αρχείο .jar με τον μεταγλωττισμένο κώδικα της εφαρμογής, ένα φάκελο lib με τις απαιτούμενες βιβλιοθήκες, ένα αρχείο java properties, τις άδειες που ορίζουν τη χρήση του λογισμικού και των βιβλιοθηκών και ένα αρχείο readme. Η εκτέλεση γίνεται όπως σε κάθε άλλη εφαρμογή java σε jar, δίνοντας μέσω του περιβάλλοντος εντολών την εντολή:

```
java –jar geosm.jar
```

Υπάρχουν τρεις βασικοί τρόποι παραμετροποίησης της εφαρμογής.

1. Δίνοντας τις απαιτούμενες παραμέτρους μέσω γραμμής εντολών,
2. Δίνοντας μέσω της γραμμής εντολών το μονοπάτι (path) ενός αρχείου .properties και ορίζοντας τις απαιτούμενες παραμέτρους στο αρχείο,
3. Μέσω γραφικού περιβάλλοντος.

Οι μέθοδοι 1. και 2. μπορούν να συνδυαστούν, στην οποία περίπτωση παράμετροι που έχουν οριστεί απευθείας στην γραμμή εντολών έχουν προτεραιότητα έναντι παραμέτρων που έχουν οριστεί στο αρχείο .properties.

3.5.1 Περιβάλλον γραμμής εντολών

Εδώ περιγράφουμε τις μεθόδους παραμετροποίησης 1. και 2.. Καταρχήν δίνουμε συνοπτικά όλες τις παραμέτρους που δέχεται η εφαρμογή.

```
usage: geosm [-c <file>] [-d <database>] [-g] [-h] [-i <file>] [-M
<max_rows_to_cache>] [-o <file>] [-p <password>] [-u <username>] [-v] [-x
<context>]

 -c <file>                configuration file
 -d <database>            name of postgreSQL database to use for osm data
 -g                       if set will spawn a gui to set all variables
 -h,--help                print this message
 -i <file>                data file to be imported (Must end in '.osm' or
'.pbf')
 -M <max_rows_to_cache>   maximum number of rows to cache when executing
sql statements
 -o <file>                output file for RDF data
 -p <password>            database password
 -u <username>            database username
 -v                       verbose output
 -x <context>             context for N-Quads. If undefined N-Triples will
be printed instead
```

Το ανωτέρω κείμενο μπορεί να προκύψει και μέσω της εφαρμογής με εκτέλεση της εντολής:

```
java –jar geosm.jar –h
```

ή

```
java –jar geosm.jar --help
```

Αναλυτικά:

```
-i <file>
```

Το OSM data dump αρχείο προς επεξεργασία. Πρέπει να έχει την κατάληξη .pbf ή .osm για αρχεία με το μορφότυπο PBF ή OSM XML αντίστοιχα.

```
-o <file>
```

Το αρχείο εξόδου που θα γραφτούν οι N-TRIPLES ή N-QUADS. Αν το αρχείο υπάρχει ήδη θα αντιγραφεί.

```
-d <database>
```

Το όνομα της βάσης δεδομένων στην οποία θα γραφτούν τα δεδομένα του OSM data dump. Αν υπάρχει ήδη βάση με αυτό το όνομα θα γίνει dropped. Όταν τελειώσει η εκτέλεση η βάση γίνεται dropped. Η βάση πρέπει να είναι PostgreSQL με PostGIS extension.

```
-u <username>
```

Όνομα χρήστη της βάσης δεδομένων.

```
-p <password>
```

Κωδικός χρήστη της βάσης δεδομένων.

```
-M <max_rows_to_cache>
```

Μέγιστος αριθμός γραμμών που θα γίνουν cached κατά την εκτέλεση των SQL επερωτήσεων.

```
-x <context>
```

Ένα URI που θα χρησιμοποιηθεί ως context κατά την παραγωγή των N-QUADS. Αν δεν δοθεί ή δοθεί η κενή συμβολοσειρά θα παραχθούν N-TRIPLES.

```
-g
```

Εκκινεί το γραφικό περιβάλλον χρήστη. Αν υπάρχει αυτή η παράμετρος, όλες οι άλλες εντολές που έχουν δοθεί μέσω γραμμής εντολών αγνοούνται.

```
-v
```

Verbose. Αλλάζει το logging level από WARN σε INFO. Το πρόγραμμα θα εκτυπώνει ενημερωτικά μηνύματα για την πορεία της επεξεργασίας στο stdout.

```
-c <file>
```

Εδώ προσδιορίζουμε το path προς ένα αρχείο .properties που θα περιέχει τις παραμέτρους που θέλουμε να περάσουμε στο πρόγραμμα. Οποιεσδήποτε παράμετροι έχουν οριστεί μέσω των ανωτέρω εντολών έχουν προτεραιότητα έναντι των αντίστοιχων παραμέτρων στο .properties αρχείο. Το αρχείο θα πρέπει να έχει την κατάλληλη μορφή. Ενδεικτικά:

```
# Input and output file paths (source must end in '.osm' or '.pbf' for
RDF/XML and PBF respectively)
input=<input file>
output=<output file>
```

```
# Database config
dbname=<database name>
user=<database user>
password=<database user password>

max_rows_to_cache=<max rows to cache in sql queries>

# Model context for N-Quads (if blank N-Triples will be printed)
context=<context URI>
```

3.5.2 Γραφικό περιβάλλον

Η εφαρμογή μπορεί να εκτελεστεί μέσω γραφικού περιβάλλοντος με χρήση της ακόλουθης εντολής:

```
java -jar geosm.jar -g
```

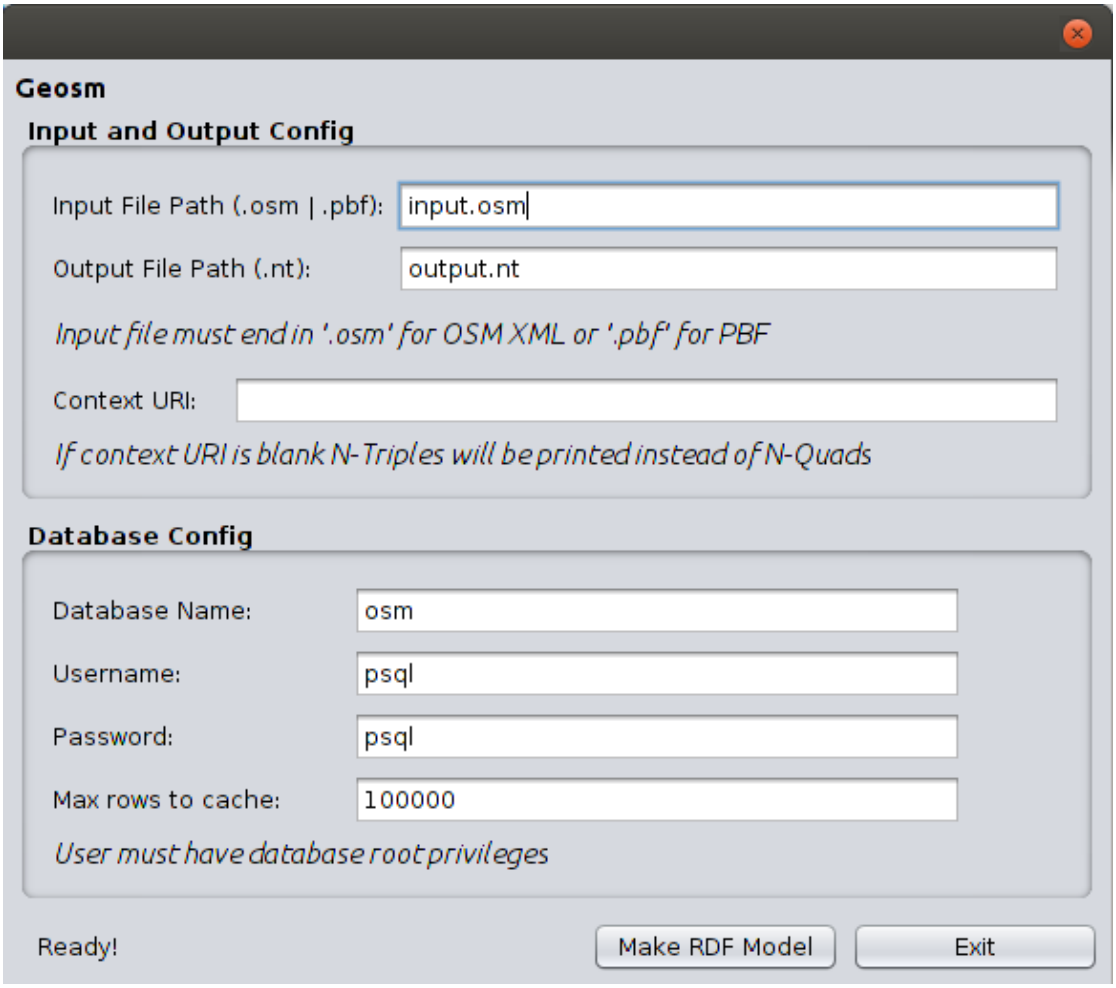


Illustration 1: Γραφικό Περιβάλλον

Θέτοντας τις ζητούμενες πληροφορίες στα πεδία και πατώντας το κουμπί “Make RDF Model” θα εκκινήσει η διαδικασία μετατροπής του αρχείου εισόδου σε RDF. Το μήνυμα “Ready!” κάτω αριστερά θα αντικατασταθεί με το μήνυμα “Working...”, όσο η επεξεργασία είναι σε εξέλιξη, και στη συνέχεια με το μήνυμα “Done!”, όταν η επεξεργασία έχει τελειώσει και η εφαρμογή είναι έτοιμη να εκκινήσει καινούργια επεξεργασία.

Η σελίδα αυτή είναι σκόπιμα λευκή.

4

Σχεδίαση μετρικών διασύνδεσης σημασιολογικών γεωχωρικών δεδομένων

4.1 Θεωρητικό υπόβαθρο

Σε αυτό το κεφάλαιο θα δώσουμε το απαραίτητο θεωρητικό υπόβαθρο για την σχεδίαση μετρικών διασύνδεσης σημασιολογικών γεωχωρικών δεδομένων, ξεκινώντας πρώτα από την παρουσίαση μερικών υπαρχόντων μετρικών ομοιότητας συμβολοσειρών στις οποίες θα βασιστούμε και στη συνέχεια παρουσιάζοντας τα μέτρα ποιότητας μετρικών precision, recall και F-measure.

4.1.1 Μετρικές Ομοιότητας Συμβολοσειρών

Ο όρος μετρικές ομοιότητας συμβολοσειρών αναφέρεται σε μετρικές που χρησιμοποιούνται στην ποσοτικοποίηση της ομοιότητας/ανομοιότητας δύο συμβολοσειρών. Τέτοιες μετρικές χρησιμοποιούνται από εφαρμογές όπως το προσεγγιστικό ταίριασμα συμβολοσειρών ή η ασαφής αναζήτηση συμβολοσειρών μέχρι το ταίριασμα γενετικών αλληλουχιών DNA και RNA.

4.1.1.1 Levenshtein distance

Η απόσταση Levenshtein αναπτύχθηκε από τον Vladimir Levenshtein το 1965 και χρησιμοποιείται για τον υπολογισμό της ομοιότητας μεταξύ δύο αλληλουχιών. Ανεπίσημα η μετρική περιγράφεται ως ο ελάχιστος αριθμός αλλαγών (εισαγωγή, διαγραφή, αντικατάσταση) που απαιτούνται για να μετατραπεί μία αλληλουχία σε μία άλλη.

Μαθηματικά, η απόσταση Levenshtein μεταξύ δύο συμβολοσειρών a, b δίνεται από τον τύπο $lev_{a,b}(|a|,|b|)$ όπου:

$$lev_{a,b}(|i|,|j|)=\begin{cases} max(i,j), if\ min(i,j)=0 \\ min\begin{cases} lev_{a,b}(i-1,j)+1 \\ lev_{a,b}(i,j-1)+1 \\ lev_{a,b}(i-1,j-1)+[a_i \neq b_j] \end{cases}, otherwise \end{cases}$$

Επισημαίνεται ότι στο ελάχιστο η πρώτη σχέση αντιστοιχεί στην διαγραφή χαρακτήρα (από το a στο b), η δεύτερη στην είσοδο χαρακτήρα και η τρίτη στο ταίριασμα ή μη των χαρακτήρων.

Αυτή η μετρική έχει αρκετά απλά άνω και κάτω όρια. Επιγραμματικά:

- Είναι τουλάχιστον η διαφορά των μηκών των δύο συμβολοσειρών,
- Είναι το πολύ το μήκος της μεγαλύτερης συμβολοσειράς,
- Είναι μηδέν αν και μόνο αν οι δύο συμβολοσειρές είναι ίσες,
- Αν οι συμβολοσειρές έχουν το ίδιο μήκος, η απόσταση Hamming είναι άνω όριο της απόστασης Levenshtein,
- Η απόσταση Levenshtein μεταξύ δύο συμβολοσειρών είναι το πολύ το άθροισμα της απόστασής τους από μία τρίτη συμβολοσειρά (τριγωνική ανισότητα).

Η απόσταση Levenshtein μπορεί να υπολογισθεί με τη χρήση δυναμικού προγραμματισμού με κόστος $\Theta(n,m)$ όπου *n* είναι το πλήθος χαρακτήρων της συμβολοσειράς *a* και *m* το πλήθος χαρακτήρων της συμβολοσειράς *b*.

4.1.1.2 Jaccard Index

Η Jaccard Index αναπτύχθηκε από τον Paul Jaccard το 1901 και είναι μία στατιστική για τον υπολογισμό της ομοιότητας μεταξύ δύο συνόλων. Ο συντελεστής Jaccard μετράει την ομοιότητα δύο συνόλων και ορίζεται ως ο λόγος του πλήθους της τομής δύο συνόλων προς την ένωσή αυτών.

$$J(A,B)=\frac{|A\cap B|}{|A\cup B|}$$

Παρόμοια η απόσταση Jaccard ορίζεται ως το συμπλήρωμα του συντελεστή Jaccard και μετράει την ανομοιότητα δύο συνόλων και υπολογίζεται από την ακόλουθη σχέση:

$$J_\delta(A,B)=1-J(A,B)=\frac{|A\cup B|-|A\cap B|}{|A\cup B|}$$

Η τεχνική min-wise independent permutations locality sensity hashing scheme (MinHash) μπορεί να χρησιμοποιηθεί για να υπολογίσει μία ακριβή εκτίμηση του συντελεστή Jaccard για δύο σύνολα τα οποία αναπαριστώνται από μία σταθερού μήκους υπογραφή που προκύπτει από τις ελάχιστες τιμές της συνάρτησης hash.

Στο πλαίσιο της σύγκρισης συμβολοσειρών η απόσταση Jaccard μπορεί να εφαρμοσθεί πάνω σε συμβολοσειρές που έχουν χωριστεί σε tokens με βάση κάποια regular expression (π.χ., "\s" για να χωριστεί στα κενά μεταξύ των λέξεων) για να δώσει μία μετρική του πλήθους των tokens στα οποία διαφέρουν.

4.1.1.3 Soft Jaccard

Η Soft Jaccard είναι μία συνδυαστική μετρική η οποία συνδυάζει τις μετρικές Jaccard και Levenshtein. Το αποτέλεσμα της μετρικής είναι όπως και στην Jaccard το πλήθος της τομής δύο συνόλων προς το πλήθος της ένωσης τους. Η διαφορά έγκειται στον τρόπο υπολογισμού της τομής όπου στην περίπτωση της Soft Jaccard δύο tokens αποτελούμενα από αλληλουχίες χαρακτήρων θεωρούνται ίδια αν η απόστασή Levenshtein τους είναι ίση ή μικρότερη από μία τιμή T που επιλέγεται αυθαίρετα ως όριο.

Όσον αφορά τη σύγκριση συμβολοσειρών αποτελούμενων από tokens (π.χ., λέξεις σε μία πρόταση) η μετρική Soft Jaccard παρουσιάζει το συγκριτικό πλεονέκτημα ότι επιτρέπει μεγαλύτερο βαθμό κοκκιότητας (granularity) έναντι της Levenshtein με τη δυνατότητα προσδιορισμού ανοχής σε αποστάσεις ανά token αντί συνολικά για ολόκληρες τις συμβολοσειρές, ενώ δεν έχει τη δυαδική (δύο token ελέγχονται αν είναι ακριβώς ίδια ή όχι) "αυστηρότητα" της Jaccard.

4.1.2 Ανάκτηση πληροφορίας

Στα πλαίσια της ανάκτησης πληροφορίας απαιτούνται μέτρα το οποία να επιτρέπουν την ποσοτικοποίηση της ποιότητας της παρεχόμενης πληροφορίας με σκοπό την αξιολόγηση και αντικειμενική σύγκρισή των χρησιμοποιούμενων τεχνικών ώστε να γίνεται δυνατή η επιλογή της βέλτιστης για τις ανάγκες κάθε εφαρμογής. Τρία τέτοια μέτρα τα οποία θα χρησιμοποιηθούν στο πλαίσιο αυτής της διπλωματικής για την αξιολόγηση των μετρικών που έχουμε αναπτύξει παρουσιάζονται στα ακόλουθα κεφάλαια.

4.1.2.1 Precision

Ως precision ορίζεται το ποσοστό των ανακτηθέντων, επιθυμητών οντοτήτων επί του συνόλου των ανακτηθέντων οντοτήτων. Μαθηματικά ορίζεται ως:

$$precision = \frac{|\{relevant\ entities\} \cap \{retrieved\ entities\}|}{|\{retrieved\ entities\}|}$$

Εναλλακτικά, χρησιμοποιώντας τους όρους true positive (tp), true negative (tn), false positive (fp) και false negative (fn) όπου οι όροι positive, negative αναφέρονται στην ανάκτηση ή μη κάποιας οντότητας και οι όροι true, false στην ορθότητα της ανάκτησης ή μη μπορούμε να το ορίσουμε μέσω του τύπου:

$$precision = \frac{tp}{tp + fp}$$

Επιπλέον του μέτρου precision (*p*) που λαμβάνει υπόψιν όλες τις ανακτηθέντες οντότητες ορίζεται και το μέτρο precision@n (*p@n*) που λαμβάνει υπόψιν μόνο τις πρώτες *n* ανακτηθέντες οντότητες (π.χ., τις πρώτες 10 σελίδες αποτελεσμάτων σε μία μηχανή αναζήτησης στον παγκόσμιο ιστό).

4.1.2.2 Recall

Ως recall ορίζεται το ποσοστό των ανακτηθέντων, επιθυμητών οντοτήτων επί του συνόλου των επιθυμητών οντοτήτων. Μαθηματικά ορίζεται ως:

$$recall = \frac{|\{relevant\ entities\} \cap \{retrieved\ entities\}|}{|\{relevant\ entities\}|}$$

Όμοια με πριν χρησιμοποιώντας τους όρους true positive (tp), true negative (tn), false positive (fp) και false negative (fn) μπορούμε να το ορίσουμε μέσω του τύπου:

$$recall = \frac{tp}{tp + fn}$$

4.1.2.3 F-Measure

Το μέτρο F-Measure (traditional F-Measure ή balanced F-Measure) χρησιμοποιείται για να συνδυάσει τα μέτρα precision και recall και ορίζεται ως ο αρμονικός μέσος όρος τους. Μαθηματικά:

$$F = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

Αυτό το μέτρο είναι επίσης γνωστό ως το F_1 μέτρο γιατί τα precision και recall έχουν το ίδιο βάρος. Ορίζεται και το F_β μέτρο για μη αρνητικές, πραγματικές τιμές του *β*, όπου το *β* είναι ο συντελεστής βάρους του precision ως προς το recall.

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{\beta^2 \cdot precision + recall}$$

Στα πλαίσια της διπλωματικής όπου αναφέρεται το F-Measure (F) θα εννοείται το F_1 .

4.2 Σχεδίαση μετρικών διασύνδεσης

Κίνητρο αυτού του τμήματος της διπλωματικής ήταν η σχεδίαση νέων μετρικών που θα μπορούσαν να χρησιμοποιηθούν στην διασύνδεση σημασιολογικών γεωχωρικών δεδομένων. Ελπίζεται ότι η διασύνδεση με συνδυαστική χρήση των αναγνωριστικών συμβολοσειρών και τις γεωχωρικής πληροφορίας για κάθε οντότητα θα επιφέρει ποιοτικά καλύτερα αποτελέσματα από ότι

μπορούσε να επιτευχθεί μέσω της διασύνδεσης πάνω στις συμβολοσειρές μόνο. Για το σκοπό αυτό δόθηκε έμφαση σε τρεις σχετιζόμενες μεταξύ τους ερευνητικές προσπάθειες.

- Στην ανάπτυξη μετρικών για τη σύγκριση συμβολοσειρών,
- Στην ανάπτυξη μετρικών για τη σύγκριση γεωχωρικής πληροφορίας και
- Στην ανάπτυξη συνδυαστικών μετρικών που θα λαμβάνουν υπόψιν και τις δυο ανωτέρω μετρικές.

Η υλοποίηση και δοκιμή των μετρικών έγινε πάνω στο εργαλείο Silk [Silk] με χρήση της γλώσσας Scala και της βιβλιοθήκης JTS [JTS].

4.2.1 Σχεδίαση νέων γεωχωρικών μετρικών

Σε αυτό το κεφάλαιο περιγράφουμε τη σχεδίαση μιας μετρικής ομοιότητας γεωχωρικών δεδομένων σε συμφωνία με το σενάριο χρήσης σύγκρισης δύο γεωμετριών εκ των οποίων:

- Η μία περιγράφεται ως σημείο (WKT POINT),
- Η δεύτερη περιγράφεται ως κλειστή γραμμή (WKT LINESTRING) ή πολύγωνο (WKT POLYGON).

Κίνητρο του σεναρίου χρήσης είναι η διασύνδεση σημείων ενδιαφέροντος (POIs) από διαφορετικά datasets, όπου στο ένα η οντότητα μπορεί να αναπαρίσταται ως ένα σημείο (π.χ., προσεγγιστική αναπαράσταση της θέσης ενός κτηρίου με ένα σημείο) ενώ στο άλλο ως πολύγωνο (π.χ., το περίγραμμα ενός κτηρίου).

Αυτή η μετρική αποτελείται από το συνδυασμό των δύο μετρικών που περιγράφονται στο υποκεφάλαια που ακολουθούν.

Σημειώνεται ότι ο όρος πολύγωνο θα χρησιμοποιηθεί για να περιγράφει τη γεωμετρία πολύγωνο όσο και την γεωμετρία γραμμή, όπου η αρχή και το τέλος της γραμμής συμπίπτουν (κλειστή).

4.2.1.1 Γεωμετρική απόσταση

Σκοπός αυτής της μετρικής είναι να επεκτείνουμε την έννοια της απόστασης δύο σημείων ώστε να ορίζεται η απόσταση σημείου με πολύγωνο.

Καταρχήν δίνουμε τον τρόπο υπολογισμού της απόστασης σημείου προς σημείο σε σφαίρα. Αυτή η απόσταση ονομάζεται απόσταση μεγάλου κύκλου ή ορθοδρομική απόσταση και ορίζεται ως η συντομότερη διαδρομή μεταξύ δύο σημείων όπου κάθε σημείο της διαδρομής βρίσκεται στην επιφάνεια του κύκλου. Για κάθε δύο σημεία στην επιφάνεια ενός κύκλου τα οποία δεν βρίσκονται σε αντίθετα άκρα της ίδιας διαμέτρου (αντίποδας) ορίζεται ένας μοναδικός μεγάλος κύκλος (γεωδεσικός). Το δύο σημεία χωρίζουν τον κύκλο σε δύο τόξα. Το μήκος του μικρότερου τόξου είναι

η απόσταση μεγάλου κύκλου των δύο σημείων. Ένας μεγάλος κύκλος που έχει μια τέτοια μετρική απόστασης ονομάζεται Riemannian κύκλος.

Για να τον υπολογισμό αυτής της απόστασης *d* μεταξύ δύο σημείων με latitude $\varphi_{1,}\varphi_{2}$ και longitude $\lambda_{1,}\lambda_{2}$ αντίστοιχα σε κύκλο ακτίνας *r* μπορεί να χρησιμοποιηθεί η συνάρτηση haversin ως εξής:

$$d = r \cdot haversin^{-1}(h) = 2\mathrm{r}\arcsin(\sqrt{h}),$$
$$\text{όπου } h = haversin(d/r) = haversin(\varphi_2 - \varphi_1) + \cos(\varphi_1)\cdot\cos(\varphi_2)\cdot haversin(\lambda_2 - \lambda_1),$$
$$\text{και } haversin(\theta) = \sin^2(\frac{\theta}{2})$$

Συνοπτικά:

$$d = 2\mathrm{r}\cdot\arcsin(\sqrt{\sin^2(\frac{\varphi_2-\varphi_1}{2}) + \cos(\varphi_1)\cdot\cos(\varphi_2)\cdot\sin^2(\frac{\lambda_2-\lambda_1}{2})})$$

Αντίστοιχος υπολογισμός της απόστασης μπορεί να γίνει χρησιμοποιώντας τον σφαιρικό νόμο των συνημίτονων, αλλά για μικρές αποστάσεις (<1km) λόγω της χρήσης συνημίτονων προκύπτουν μεγάλα σφάλματα στρογγυλοποιήσεων. Η συνάρτηση haversin αποφεύγει αυτό το πρόβλημα με τη χρήση ημίτονων.

Επισημαίνεται ότι η συνάρτηση haversin υποθέτει ιδανική σφαίρα με αποτέλεσμα σφάλματα όταν χρησιμοποιείται για υπολογισμούς πάνω στην επιφάνεια της Γης. Η ακτίνα της Γης ποικίλει σε μήκος από 6356.752 km στους πόλους σε 6378.137 km στον ισημερινό. Επίσης η ακτίνα καμπύλης για μία γραμμή με διεύθυνση Βορρά-Νότο πάνω στην επιφάνεια της Γης είναι 1% μεγαλύτερη στους πόλους από ότι στον ισημερινό. Αυτό έχει σαν αποτέλεσμα την εισαγωγή σφάλματος στον υπολογισμό της απόστασης, το οποίο μπορεί να φτάνει τα ~2 km για απόσταση ~20,000 km. Συγκριτικά, οι εξισώσεις του Vincenty παρέχουν ακρίβεια 0.5mm για σημεία στην επιφάνεια της Γης. Στα πλαίσια αυτού του σεναρίου χρήσης απαιτείται ακρίβεια σε υπολογισμούς αποστάσεων μέχρι μερικά χιλιόμετρα και επομένως ένα σφάλμα ~0.1% κρίθηκε αποδεκτό.

Σε δεύτερη φάση θέλουμε να ορίσουμε τη σημασία της απόστασης μεταξύ σημείου και πολυγώνου. Για το σκοπό αυτό κάνουμε χρήση της έννοιας του βαρυκέντρου. Ως βαρύκεντρο μιας επίπεδης επιφάνειας ορίζεται ο αριθμητικός μέσος όρος όλων των σημείων αυτής. Ο ορισμός επεκτείνεται σε ν-διάστατες γεωμετρίες ως ο μέσος όρος των σημείων σε όλες τις κατευθύνσεις συντεταγμένων. Στο πλαίσιο αυτής της μετρικής υπολογίζουμε το βαρύκεντρο ενός πολυγώνου και χρησιμοποιούμε το προκύπτον σημείο ως προσεγγιστική αναπαράσταση αυτού. Στη συνέχεια υπολογίζουμε την απόσταση μεγάλου κύκλου μεταξύ του βαρύκεντρου και της δεύτερης γεωμετρίας και χρησιμοποιούμε το αποτέλεσμα ως μετρική της απόστασής τους.

4.2.1.2 Γεωμετρικό containment

Σκοπός αυτής της μετρικής είναι να ελέγξουμε αν ένα σημείο περιέχεται (is contained) σε ένα πολύγωνο. Πρόκειται για δυαδική μετρική, η οποία επιστρέφει αληθές αν ένα σημείο βρίσκεται στο εσωτερικό ενός πολυγώνου και ψευδές αλλιώς. Μία γεωμετρία Α θεωρείται ότι περιέχεται στο εσωτερικό μίας γεωμετρίας Β αν κάθε σημείο της γεωμετρίας Α είναι και σημείο της γεωμετρίας Β και τα εσωτερικά των δύο γεωμετριών έχουν τουλάχιστον ένα κοινό σημείο.

4.2.2 Σχεδίαση συνδυαστικών μετρικών συμβολοσειρών

Στα προηγούμενα κεφάλαια περιγράψαμε τις υπάρχουσες μετρικές συμβολοσειρών Levenshtein, Jaccard και Soft Jaccard και δώσαμε τους λόγους για τους οποίους η Soft Jaccard υπερέχει στο σενάριο χρήσης σύγκρισης συμβολοσειρών που αποτελούνται από tokens, όπως για παράδειγμα στην σύγκριση σύνθετων ονομάτων. Σε αυτό το κεφάλαιο θα παρουσιάσουμε καταρχήν σενάρια χρήσης στα οποία η άμεση εφαρμογή της Soft Jaccard δεν δίνει βέλτιστο αποτέλεσμα και στη συνέχεια θα περιγράψουμε μία σύνθετη μετρική που επιλύει αυτά τα προβλήματα.

Ας θεωρήσουμε τα ακόλουθα ζεύγη συμβολοσειρών:

1) “Μέγαρο Νομαρχίας Ιωαννίνων” / “ΜΕΓΑΡΟ ΝΟΜΑΡΧΙΑΣ ΙΩΑΝΝΙΝΩΝ”
2) “Μέγαρο Μουσικής Αθηνών” / “Μέγαρο Μουσικής Θεσσαλονίκης”
3) “401 Γενικό Στρατιωτικό Νοσοκομείο” / “403 Γενικό Στρατιωτικό Νοσοκομείο”

Καταρχήν μεταγραμματίζουμε τις συμβολοσειρές με Λατινικούς printable ASCII χαρακτήρες. Αυτό μας επιτρέπει να χρησιμοποιούμε τους Λατινικούς χαρακτήρες ως βάση αναφοράς για τη μετέπειτα ανάλυση.

1. “Megaro Nomarchias Ioanninon” / “MEGARO NOMARCHIAS IOANNINON”
2. “Megaro Mousikis Athinon” / “Megaro Mousikis Thessalonikis”
3. 401 Geniko Stratiotiko Nosokomeio” / “403 Geniko Stratiotiko Nosokomeio”

Εφαρμόζοντας την μετρική Soft Jaccard η δυσκολία έγκειται στην κατάλληλη επιλογή ορίων (thresholds) που θα ταιριάξουν θετικά το πρώτο ζεύγος συμβολοσειρών, ενώ θα ταιριάξουν αρνητικά το δεύτερο και τρίτο ζεύγος. Θεωρούμε δύο όρια, το όριο *J*, ως το μέγιστο αριθμό λέξεων που μπορούν να διαφέρουν μεταξύ των δύο συμβολοσειρών, και το όριο *L*, ως τη μέγιστη απόσταση Levenshtein μεταξύ δύο λέξεων ώστε να θεωρηθούν ίδιες.

Αμέσως γίνεται προφανές ότι για $J \geq 1$ η μετρική θα επιστρέψει false positive για τα ζεύγη 2, 3. Αυτό μας οδηγεί στην επιλογή $J = 0$. Εξετάζοντας τώρα τις αποδεκτές τιμές για το όριο *L* διαπιστώνουμε ότι για να εξασφαλισθεί απάντηση true positive για το 1 απαιτείται $L \geq 9$. Αντίστοιχα για να εξασφαλισθεί απάντηση true negative για το 3 απαιτείται $L = 0$. Παρατηρούμε ότι δεν υπάρχει εύρος τιμών που να εξασφαλίζει ορθή απάντηση και τα τρία ζεύγη.

Για να αντιμετωπίσουμε αυτές τις δυσκολίες διαχωρίζουμε τις συμβολοσειρές ως προς αλφαβητικούς και αριθμητικούς χαρακτήρες. Οι αριθμητικοί χαρακτήρες συγκρίνονται με μία απλή μετρική ομοιότητας ώστε να εξασφαλισθεί η απαίτηση ισότητάς τους και όχι απλής ομοιότητας. Οι αλφαβητικοί χαρακτήρες μετατρέπονται σε lower case ώστε να αγνοηθούν διαφορές στις συμβολοσειρές λόγω κεφαλαιοποίησης, χωρίζονται σε tokens με το regex “\s” και στη συνέχεια συγκρίνονται με τη μετρική Soft Jaccard.

4.3 Πειράματα και αξιολόγηση

4.3.1 Οργάνωση πειραμάτων

4.3.1.1 Επιλογή δεδομένων

Για την πειραματική δοκιμή των σχεδιασθέντων μετρικών ελέγχθηκαν ως πιθανές πηγές δεδομένων τα OpenStreetMap, geodata.gov.gr, WikiMapia και POIs.gr που περιγράφονται στο κεφάλαιο 2.2 σε μια ποικιλία κατηγοριών (δημόσιες υπηρεσίες, σχολεία, νοσοκομεία, κτλ.). Βασικές απαιτήσεις για την επιλογή των πειραματικών δεδομένων είναι:

- Το ένα σετ δεδομένων να περιέχει αναπαράσταση της γεωχωρικής πληροφορίας ως σημείο και το έτερο σετ δεδομένων ως πολύγωνο ώστε να υπαγόμαστε στο σενάριο χρήσης που εξετάζουμε,
- Επαρκές μέγεθος σετ δεδομένων ώστε να μπορούν να ληφθούν ασφαλή στατιστικά συμπεράσματα,
- Σχετικά καλή ποιότητα δεδομένων όσων αφορά τα σφάλματα ώστε τα δεδομένα να είναι εντός των ρεαλιστικών δυνατοτήτων των χρησιμοποιούμενων μετρικών.

Με βάση αυτά τα κριτήρια επιλέχθηκαν για το σκοπό των ακόλουθων πειραμάτων το σετ δεδομένων που περιέχει τα σχολεία των Ελληνικών Περιφερειών από το geodata.gov.gr (~17k οντότητες) (πλέον θα αναφέρεται ως GeoData-Σχολεία) και το σετ δεδομένων της κατηγορίας Σχολεία, για την περιοχή της Αττικής από το WikiMapia (~800 οντότητες) (πλέον θα αναφέρεται ως WikiMapia-Σχολεία). Από το WikiMapia-Σχολεία θα χρησιμοποιήσουμε μόνο τις πρώτες 400 οντότητες ώστε να διευκολυνθεί η χειροκίνητη εξαγωγή συμπερασμάτων από τα πειραματικά αποτελέσματα.

4.3.1.2 Προεπεξεργασία δεδομένων

Για τις ανάγκες των πειραμάτων απαιτούνταν μετατροπή των δεδομένων GeoData-Σχολεία, WikiMapia-Σχολεία από τον μορφότυπο στον οποίο παρέχονται από τον αντίστοιχο οργανισμό σε RDF γράφο με τη γεωμετρική πληροφορία αναπαραστημένη με το μορφότυπο WKT από το Simple

Features πρότυπο. Για να εξασφαλισθεί συμβατότητα με το πρώτο τμήμα της διπλωματικής οι μετατροπές σε RDF έγιναν επιπλέον σε συμφωνία με το GeoSPARQL πρότυπο.

Όλες οι μετατροπές δεδομένων των σετ δεδομένων GeoData-Σχολεία, WikiMapia-Σχολεία έγιναν με χρήση του εργαλείου Google Refine που περιγράφεται στο κεφάλαιο 2.3.7. Τα βήματα των διαδικασιών μετατροπής έχουν αποθηκευτεί σε αρχείο JSON με σκοπό την εύκολα επανάληψή τους σε δεδομένα με ίδιο μορφότυπο.

Αναλυτικά:

- GeoData-Σχολεία:

  Το σετ δεδομένων GeoData-Σχολεία παρέχεται από το geodata.gov.gr σε μορφή αρχείου στο μορφότυπο CSV. Τα δεδομένα αναπαριστώνται σε πέντε στήλες χωρισμένες με “;”, εκ των οποίων: Η πρώτη περιέχει τον αύξοντα αριθμό, η δεύτερη το όνομα του σχολείου, η τρίτη είναι κενή, η τέταρτη το γεωγραφικό μήκος (long) και η πέμπτη το γεωγραφικό πλάτος (lat). Η διαδικασία της μετατροπής σε RDF γίνεται με απεικόνιση του ονόματος σε triple με κατηγόρημα *rdfs:label[@lang='el']*, το μεταγραμματισμό του ονόματος με Λατινικούς χαρακτήρες και την απεικόνισή του σε triple με κατηγόρημα *rdfs:label[@lang='el-latn']*, την απεικόνιση των long, lat σε triples με κατηγόρηματα wgs:long και wgs:lat αντίστοιχα και τέλος την απεικόνιση του ζεύγους long, lat σε WKT POINT σε συμφωνία με το GeoSPARQL πρότυπο σε τρεις triples, <school> geo:defaultgeometry <anonymous_node>, <anonymous_node> rdf:type sf:Point και <anonymous_node> geo:asWKT “wkt_representation”^geo:wktLiteral.

- WikiMapia-Σχολεία:

  Το σετ δεδομένων WikiMapia-Σχολεία παρέχεται μέσω του WikiMapi API σε αρχείο με μορφότυπο xml, kml, json και jsonp. Για τις ανάγκες των πειραμάτων επιλέχθηκε ο μορφότυπος kml. Τα δεδομένα στο μορφότυπο kml παρέχονται σε ένα αρχείο xml με τα δεδομένα για κάθε οντότητα να περιέχονται σε ένα xml element με όνομα Placemark που περιέχει ένα attribute “id” με τιμή ένα μοναδικό αναγνωριστικό. Ως παιδιά του Placemark περιέχονται xml elements με τα ακόλουθα δεδομένα: όνομα, περιγραφή, πληροφορίες για την αναπαράσταση της γεωμετρίας στο πρόγραμμα Google Earth, εναλλακτική αναπαράσταση της γεωμετρίας των δεδομένων σε ορθογώνιο παραλληλόγραμμο με όνομα LatLongAltBox με το βόριο, νότιο, ανατολικό και δυτικό όριό του δοσμένα χωριστά, εναλλακτική αναπαράσταση της γεωμετρίας των δεδομένων σε σημείο με συντεταγμένες long, lat, altitude και τέλος αναπαράσταση της γεωμετρίας των δεδομένων σε linestring με αλληλουχία τριάδων συντεταγμένων long, lat, altitude. Η διαδικασία της μετατροπής σε RDF έγινε με απεικόνιση του ονόματος σε triple με κατηγόρημα *rdfs:label[@lang='el']*, μεταγραμματισμός του ονόματος με Λατινικούς χαρακτήρες και απεικόνιση σε triple με κατηγόρημα *rdfs:label[@lang='el-latn']*, απεικόνιση της περιγραφής σε triple με κατηγόρημα

rdfs:comment και τέλος την απεικόνιση του linestring σε WKT LINESTRING σε συμφωνία με το GeoSPARQL πρότυπο σε τρεις triples, <school> geo:defaultgeometry <anonymous_node>, <anonymous_node> rdf:type sf:LineString και <anonymous_node> geo:asWKT “wkt_representation”^geo:wktLiteral.

Οι RDF γράφοι εξήχθησαν από το Google Refine σε αρχεία με μορφότυπο TURTLE.

4.3.1.3 Πλατφόρμα εκτέλεσης πειραμάτων Silk

Σε αυτό το κεφάλαιο περιγράφουμε πρώτα τον τρόπο μέσω του οποίου του εργαλείο Silk παράγει τις τιμές ομοιότητας με βάση τα κατώφλια που έχουν επιλεχθεί για τις μετρικές και δεύτερον κάνουμε μία παρουσίαση του τρόπου ρύθμισης του εργαλείου με χρήση της Link Specification Language.

4.3.1.3.1 Κατώφλια μετρικών και παραγωγή τιμών ομοιότητας

Οι μετρικές του Silk υπάγονται στις εξής βασικες κατηγορίες: asian (για ομοιότητα συμβολοσειρών ασιατικών χαρακτήρων), characterbased (για ομοιότητα συμβολοσειρών, π.χ. Levenshtein Distance), equality (για δυαδικές μετρικές που επιστρέφουν αληθές ή ψευδές), numeric (για αριθμητικές συγκρίσεις) και tokenbased (για συγκρίσεις πάνω σε tokens, π.χ. Jaccard). Επιστρέφουν πάντα 0 για τέλειο ταίριασμα ή τιμή μεγαλύτερη του 0 για ατελές ταίριασμα.

Η παραγωγή της τιμής ομοιότητας για κάποια μετρική χρησιμοποιεί αφενός την τιμή που επιστρέφει η μετρική (απόσταση) και αφετέρου το κατώφλι που έχει ορίσει ο χρήστης για τη συγκεκριμένη μετρική. Οι τιμή ομοιότητας ορίζεται στο εύρος $[-1,1]$, με την τιμή 1 να αναπαριστά μέγιστη βεβαιότητα και την τιμή -1 μέγιστη αβεβαιότητα και υπολογίζεται σύμφωνα με την ακόλουθη εξίσωση:

$$similarity = \begin{cases} 1.0, & for\ distance = 0 \text{ and } threshold = 0 \\ 1.0 - distance/threshold, & for\ distance \le 2 \cdot threshold \\ -1.0, & otherwise \end{cases}$$

Η ανωτέρω εξίσωση παρουσιάζεται σχηματικά στο επόμενο γράφημα. Επισημαίνουμε ότι θετικές τιμές εμπιστοσύνης (ομοιότητας) επιστρέφονται μόνο για αποστάσεις στο εύρος $[0, threshold]$ ενώ για αποστάσεις μεγαλύτερες ή ίσες του $2 \cdot threshold$ επιστρέφεται πάντα -1.

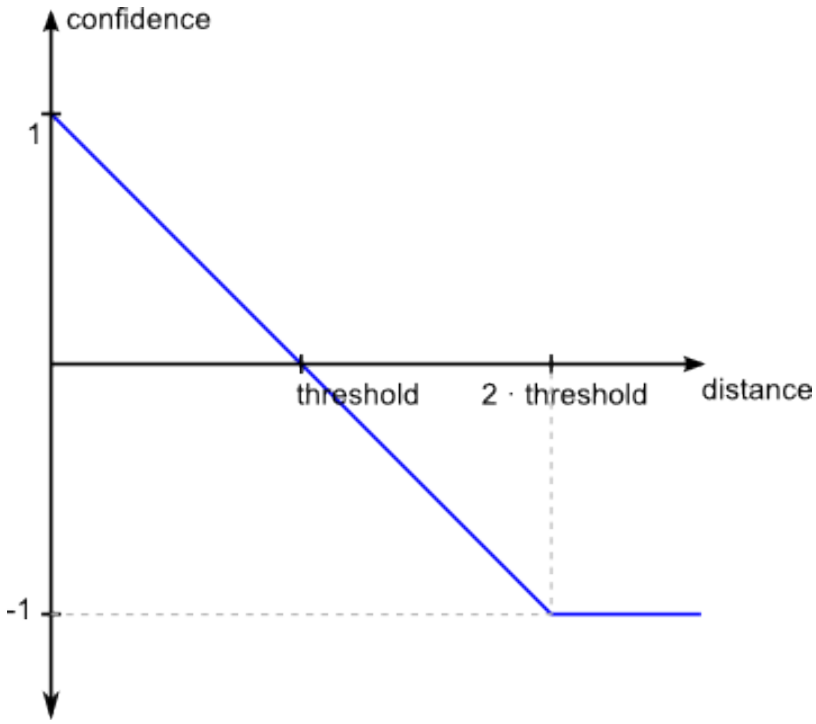


Illustration 2: Silk Confidence Computation (from Silk Website)

4.3.1.3.2 Ρυθμίσεις εργαλείου

Η ρύθμιση του εργαλείου Silk γίνεται μέσω ενός αρχείου XML σε συμφωνία με την Link Specification Language (LSL). Ο κόμβος ρίζα είναι ο <silk> και υπό τη ρίζα μπορούν να υπάρχουν τέσσερις τύποι top-level δηλώσεων:

- <Prefixes>

  Επιτρέπει την αντιστοίχιση προθεμάτων με namespaces χρησιμοποιώντας δηλώσεις της μορφής: <Prefix id="prefix id" namespace="namespace URI" />

- <DataSources>

  Επιτρέπει τον προσδιορισμό των παραμέτρων πρόσβασης σε τοπικά ή απομακρυσμένα SPARQL endpoints καθώς και σε τοπικά αρχεία RDF/XML, N-TRIPLE, TURTLE, TTL ή N3.

- <Blocking>

  Επιτρέπει τον ορισμό παραμέτρων για το χωρισμό των δεδομένων σε blocks ώστε να βελτιωθεί η επίδοση του εργαλείου ως προς το χρόνο. Αυτό έχει σαν αποτέλεσμα την αποτίμηση του καρτεσιανού γινομένου μόνο μεταξύ συσχετισμένων blocks αντί της αποτίμησης του πλήρους καρτεσιανού γινομένου για όλα τα αντικείμενα στα δεδομένα που διασυνδέονται με τη συνεπακόλουθη μείωση στον αριθμό των συγκρίσεων.

- <Interlinks>

  Οι δηλώσεις Link Specification (interlinks) ορίζουν τις συνθήκες που πρέπει να τηρούνται για παραχθεί σύνδεση μεταξύ δύο δεδομένων. Μπορεί να περιέχουν πολλαπλές μετρικές σύγκρισης, συναρτήσεις συναθροίσματος (aggregation) και συναρτήσεις μετασχηματισμών και επιτρέπει τον ορισμό κατωφλίων και βαρών. Αυτό το XML element μπορεί να έχει ως παιδιά του τα ακόλουθα:

  - <LinkType>

    Επιτρέπει τον ορισμό των links που θα παραχθούν αν τηρούνται οι ορισμένες συνθήκες (π.χ. owl:sameAs).

  - <SourceDataset>&<TargetDataset>

    Επιτρέπουν τον ορισμό του σετ των δεδομένων που θα συγκριθούν. Οι οντότητες προς διασύνδεση προσδιορίζονται μέσω περιορισμών με τη μορφή SPARQL triple patterns.

  - <LinkageRule>

    Επιτρέπει τον ορισμό των κανόνων βάση των οποίων θα γίνει η διασύνδεση των δεδομένων με χρήση input paths που ορίζουν μονοπάτια από τις οντότητες που θα διασυνδεθούν προς τις τιμές που θα χρησιμοποιηθούν για κάποια σύγκριση, μετασχηματισμών των τιμών (π.χ., lowerCase), μετρικών σύγκρισης και συναρτήσεων συνάθροισης για τη συνάθροιση πολλαπλών τιμών εμπιστοσύνης.

  - <Filter>

    Επιτρέπει τον ορισμό κανόνων φιλτραρίσματος των παραγόμενων link.

  - <Output>

    Επιτρέπει τον ορισμό προορισμού για τις παραγόμενες triples. Υποστηρίζονται αρχείο μορφότυπου N-TRIPLES ή alignment, SPARQL/Update endpoint και αρχείο μορφότυπου detailed alignment.

Όλες οι δηλώσεις εκτός των <Blocking> και <Output> απαιτούνται.

4.3.2 Διεξαγωγή πειραμάτων

Σε αυτό το κεφάλαιο θα περιγράψουμε τη διεξαγωγή πειραμάτων με διαφορετικές μετρικές, αναλύοντας παράλληλα τα παρατηρούμενα αποτελέσματα. Θα παρουσιαστούν πειραματικές δοκιμές με μετρικές ομοιότητας συμβολοσειρών μόνο, με μετρικές ομοιότητας γεωμετριών μόνο και με συνδυαστικές μετρικές ενώ τέλος θα γίνει χρήση των συμπερασμάτων που έχουν προκύψει από τα επιμέρους πειράματα για να αναλυθούν τα συγκριτικά οφέλη της συνδυαστικής προσέγγισης.

Υπενθυμίζεται ότι όλα τα πειράματα εκτελούνται σε δεδομένα μεγέθους ~400x17k.

4.3.2.1 Αξιολόγηση μετρικών ομοιότητας συμβολοσειρών

Στην αξιολόγηση μετρικών ομοιότητας συμβολοσειρών χρησιμοποιήσαμε τη συνδυαστική μετρική που περιγράφεται στο κεφάλαιο 4.2.2 και εκτελέσαμε δύο πειράματα με Levenshtein distance threshold 1 και 2 αντίστοιχα. Ως κατώφλι Jaccard στη μετρική επιλέχθηκε το 0. Τα αποτελέσματα παρουσιάζονται στον ακόλουθο γράφο:

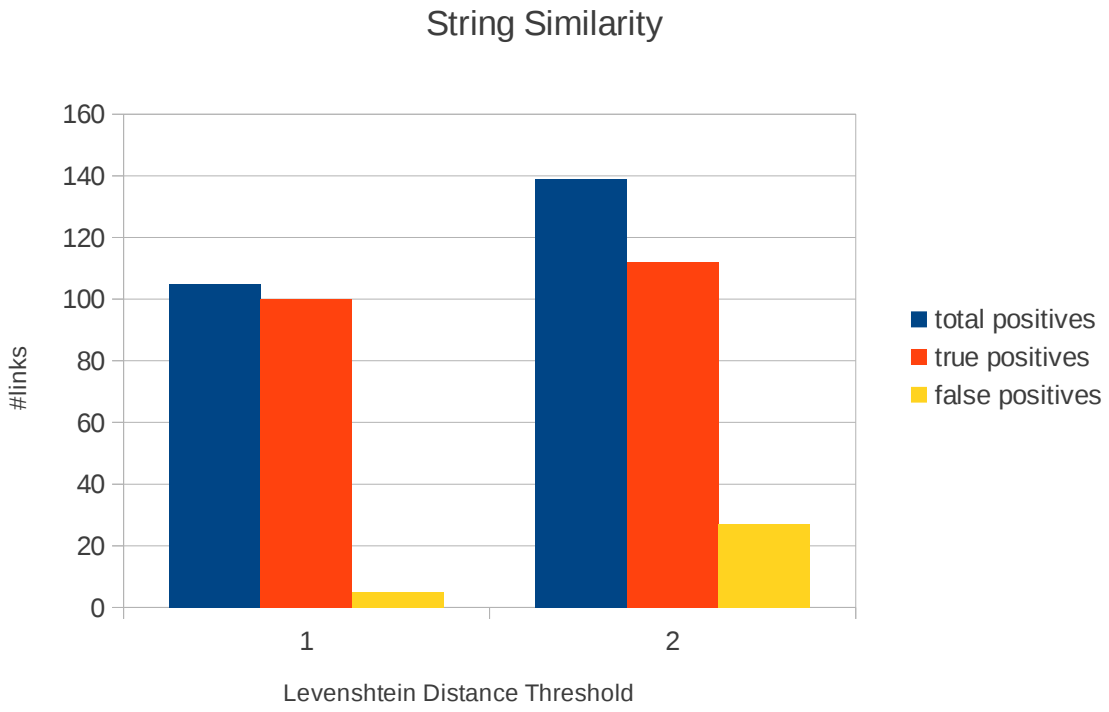


Illustration 3: Αποτελέσματα μετρικών ομοιότητας συμβολοσειρών

Αμέσως παρατηρούμε ότι αφενός με Levenshtein distance threshold 1 έχουμε όπως αναμενόταν υψηλή ακρίβεια (precision ~0.95) στα παρεχόμενα αποτελέσματα. Αφετέρου με Levenshtein distance threshold 2 έχουμε μικρή αύξηση των true positives (~12%) ενώ η ακρίβεια πέφτει στο ~0.81.

Αυτά τα αποτελέσματα μας οδηγούν να συμπεράνουμε ότι:

- Η δοκιμαζόμενη μετρική με απόσταση Levenshtein 1 μπορεί να εντοπίσει επαρκή όγκο συνδέσεων (25% του συνόλου των οντοτήτων στο WikiMapia-Σχολεία) με υψηλή ακρίβεια (~95%),
- Ο όγκος των αποτελεσμάτων δεν μπορεί να αυξηθεί εύκολα χωρίς μεγάλη απώλεια της ακρίβειας,
- Αύξηση της ακρίβειας μπορεί να γίνει μόνο με απαίτηση απόλυτης ομοιότητας των ελεγχόμενων συμβολοσειρών.

4.3.2.2 Αξιολόγηση μετρικών ομοιότητας γεωμετριών

Στην αξιολόγηση μετρικών ομοιότητας γεωμετριών χρησιμοποιήσαμε τη μετρική γεωμετρικής απόστασης με κατώφλια απόστασης 50m, 100m, 250m, 500m, 1000m και 2000m και όριο εμπιστοσύνης το 0.5. Σκοπός αυτών των πειραμάτων είναι να αποκομίσουμε μία γενική εικόνα για την χωρική συγκέντρωση των οντοτήτων στα σετ δεδομένων μας. Τα αποτελέσματα που συλλέχθηκαν φαίνονται στο ακόλουθο διάγραμμα:

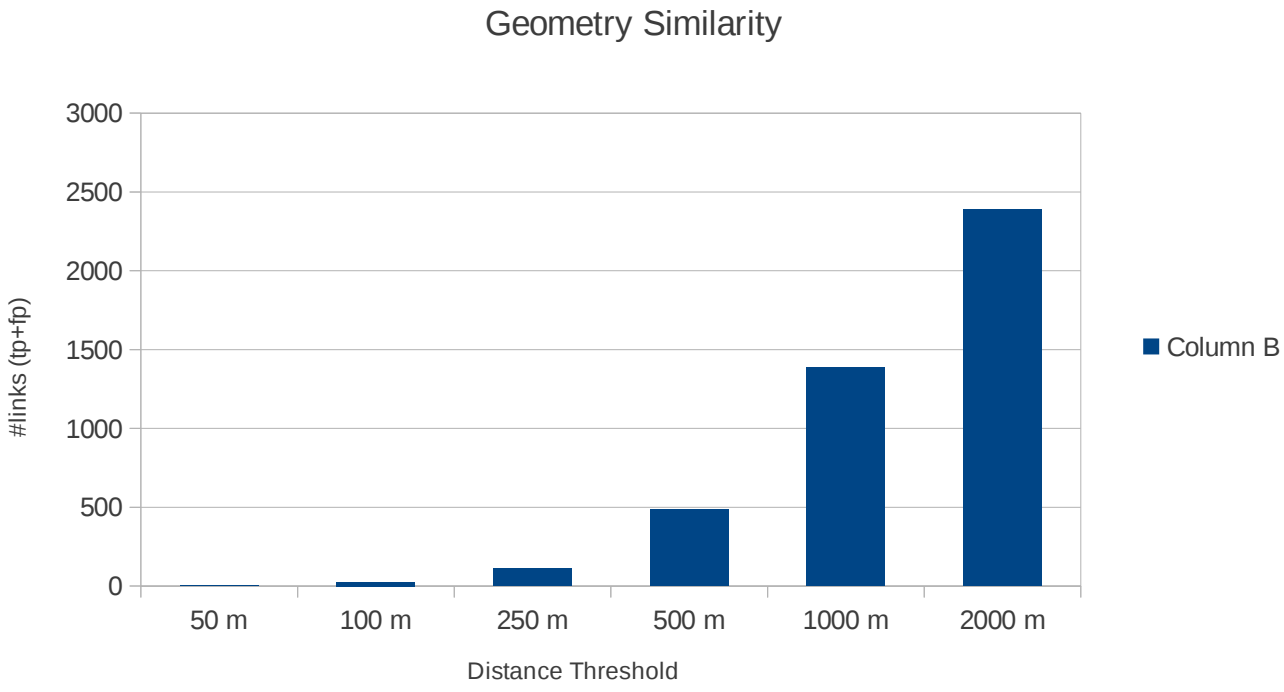


Illustration 4: Αποτελέσματα μετρικών ομοιότητας γεωμετριών

Από τα αποτελέσματα γίνεται σαφές ότι η μετρική ομοιότητας γεωμετριών μόνο δεν επαρκεί για να γίνει αποτελεσματική διάκριση μεταξύ γειτονικών σημείων ενδιαφέροντος. Αυτό το συμπέρασμα συμβαδίζει με τη φύση των οντοτήτων που εξετάζουμε καθώς, μεταξύ άλλων δυσκολιών, πολλές σχολικές εγκαταστάσεις συστεγάζονται. Ειδικά για κατώφλια μικρότερα των 500 m η χρησιμότητά τους κρίθηκε αμφίβολη με δεδομένο ότι τουλάχιστον για το WikiMapia-Σχολεία δεν έχουμε γνώση των τεχνικών συλλογής γεωχωρικής πληροφορίας που χρησιμοποιήθηκαν από τους δημιουργούς των οντοτήτων και επομένως δεν έχουμε πληροφορίες για το εύρος του σφάλματος μέτρησης.

4.3.2.3 Αξιολόγηση μετρικών συνδυαστικής ομοιότητας

Για την αξιολόγηση μετρικών συνδυαστικής ομοιότητας χρησιμοποιήσαμε τη συνδυαστική μετρική που περιγράφεται στο κεφάλαιο 4.2.2 σε συνδυασμό με τη μετρική γεωμετρικής απόστασης. Εκτελέσθηκαν δοκιμές με ένα πλήθος συνδυασμών τιμών για τις εξής παραμέτρους: κατώφλι απόστασης Levenshtein για την ομοιότητα συμβολοσειρών, κατώφλι γεωμετρικής απόστασης για την ομοιότητα γεωμετριών σε μέτρα και σχετικό βάρος μεταξύ των μετρικών ομοιότητας συμβολοσειρών και γεωμετριών.

Τα αποτελέσματα των πειραμάτων εμφανίζονται στο ακόλουθο διάγραμμα όπου στον άξονα *x* εμφανίζονται τα κατώφλια του πειράματος και στον άξονα *y* οι τιμές των precision, recall και F-Measure. Αποδεκτοί έχουν γίνει εκείνοι οι σύνδεσμοι με τιμή εμπιστοσύνης 0.5 ή μεγαλύτερη.

Σημαντική σημείωση: Για τον υπολογισμό του recall απαιτείται το πλήθος των σωστών πιθανών συνδέσμων tp+fn. Λόγω όμως της δυσκολίας να υπολογισθεί αυτή η τιμή με ακρίβεια για τα πειραματικά δεδομένα επιλέχθηκε να χρησιμοποιηθεί ως ενδεικτική αυτής η μέγιστη τιμή επιβεβαιωμένων true positives που έχει βρεθεί κατά την εκτέλεση όλων των πειραμάτων σε αυτά τα σετ δεδομένων, δηλαδή $tp+fn=128$. Αυτή η τιμή υπολογίσθηκε ως το πλήθος της ένωσης των συνόλων των επιβεβαιωμένα ορθών συνδέσεων. Ως εκ τούτου οι τιμές των recall και F-Measure δεν μπορούν να χρησιμοποιηθούν για την εξαγωγή συμπερασμάτων σε απόλυτη βάση, αλλά μόνο σε σχέση με τα άλλα αποτελέσματα του πειραματικού σετ.

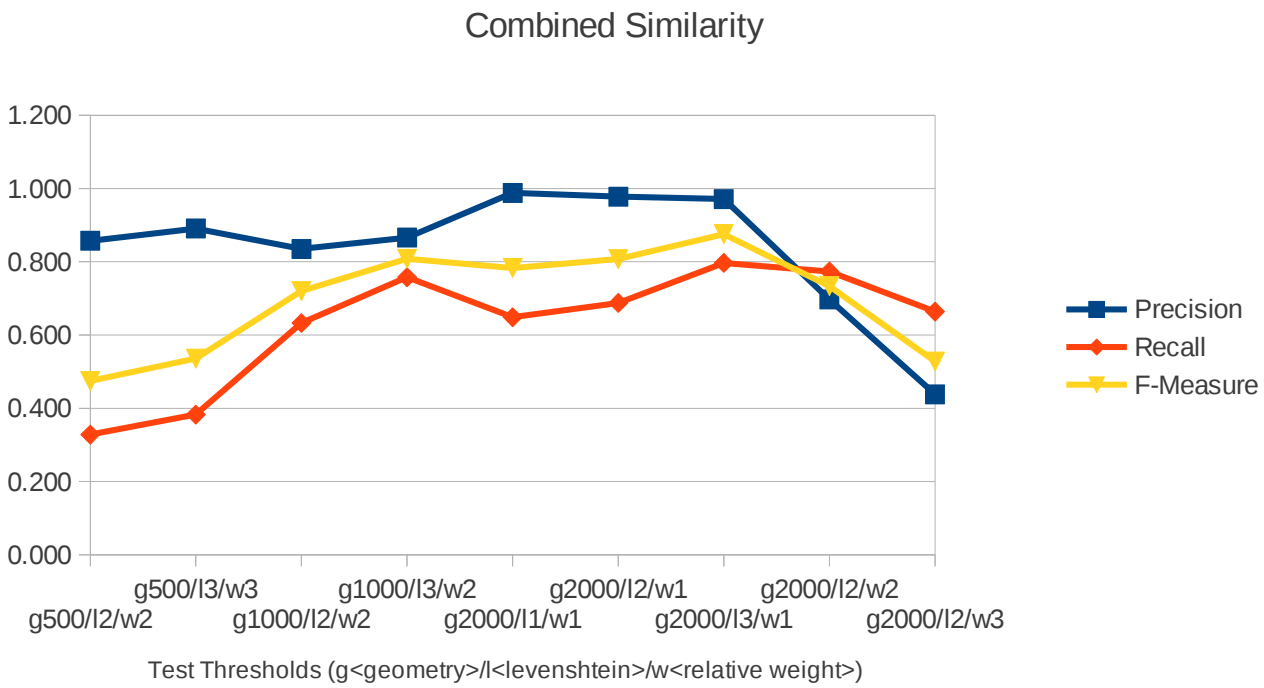


Illustration 5: Αποτελέσματα συνδυαστικών μετρικών

Από τα αποτελέσματα παρατηρούμε ότι:

- Μικρές τιμές του κατωφλίου γεωμετρικής απόστασης έχουν σημαντική αρνητική επίδραση στο recall όπως αναμενόταν από τα συμπεράσματα του 4.3.2.2,
- Το recall μπορεί να βελτιωθεί με αύξηση του σχετικού βάρους της γεωμετρικής μετρικής. Όπως φαίνεται από τα πειράματα 5-8 όμως αυτή η αύξηση προκαλεί μεγάλη απώλεια precision σε μετρικές με υψηλό κατώφλι γεωμετρικής απόστασης (2000m) που την καθιστά μη πρακτικά εφαρμόσιμη λύση. Αντίστοιχο recall, αλλά με σημαντικά καλύτερο precision επιτυγχάνεται με σχετικό βάρος 2, κατώφλι γεωμετρικής απόστασης 1000m και κατώφλι απόστασης Levenshtein 3 (πείραμα 4).
- Βέλτιστο precision επιτυγχάνεται με κατώφλι γεωμετρικής απόστασης 2000m, σχετικό βάρος 1 και κατώφλι Levenshtein στο εύρος 1-3. Συγκεκριμένα για κατώφλι Levenshtein 3 επιτυγχάνουμε ταυτόχρονα υψηλό precision και ικανοποιητικό recall.

Από τα ανωτέρω συμπεράσματα σημαντικότερο κρίνεται το τρίτο. Πριν αναλύσουμε τους λόγους αυτού παρουσιάζουμε τα precision / recall / F-Measure των συγκεντρωτικών μετρικών σε ένα καινούργιο διάγραμμα μαζί με τις αντίστοιχες τιμές για τις μετρικές συμβολοσειρών ώστε να επιτραπεί η άμεση σύγκριση.

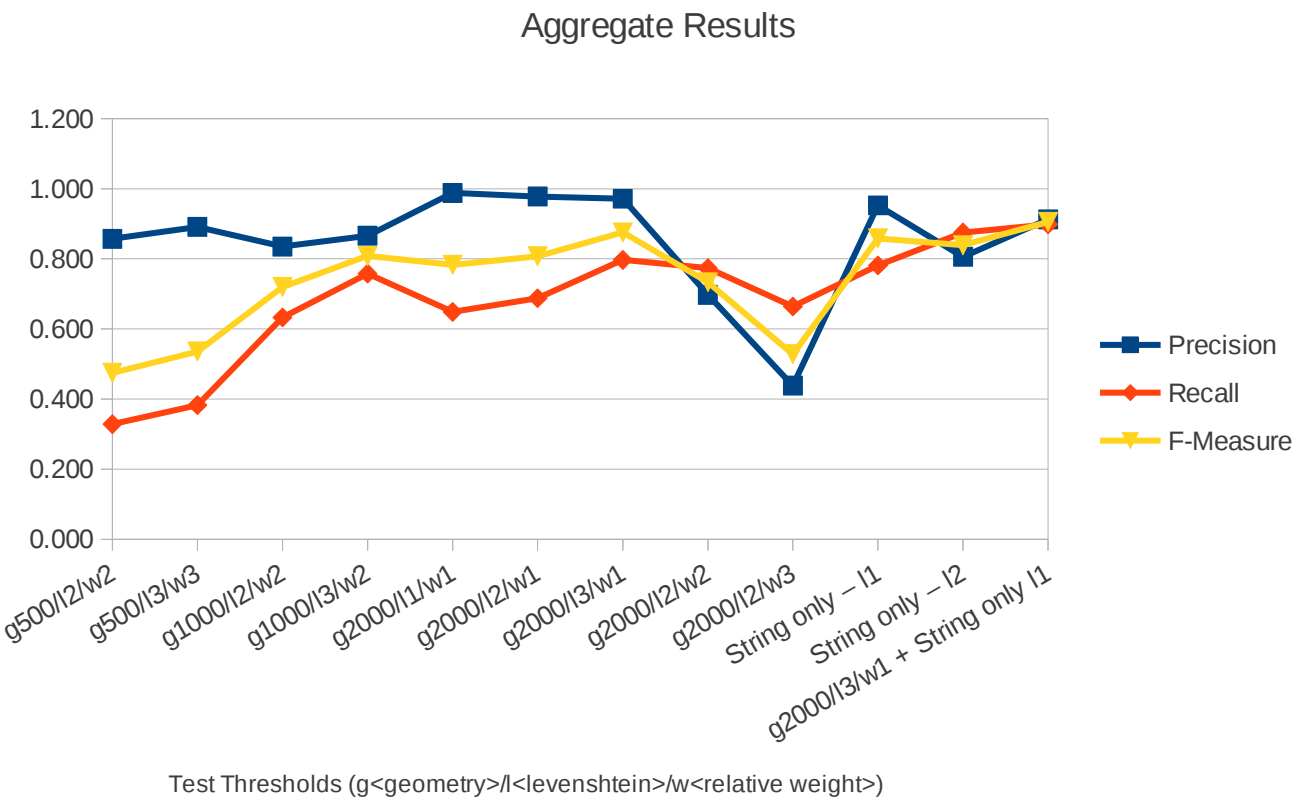


Illustration 6: Συγκεντρωτικά αποτελέσματα συνδυαστικών μετρικών και μετρικών συμβολοσειρών

Συγκρίνοντας τις στατιστικές για το πείραμα “g2000/l3/w1” (Geometry threshold 2000m, Levenshtein distance threshold 3, relative weight 1) με το πείραμα μετρικών συμβολοσειρών μόνο “String only – l1” (Levenshtein distance threshold 1) παρατηρούμε ότι το πρώτο επιτυγχάνει οριακά καλύτερα precision / recall / F-Measure παρότι το υψηλό κατώφλι απόστασης Levenshtein θα περιμέναμε από τα πειράματα των μετρικών συμβολοσειρών να προκαλεί σημαντική μείωση του precision σε συνδυασμό με αύξηση του recall. Η όχι μόνο διατήρηση, αλλά και μικρή αύξηση του precision αποδίδεται στο συνδυασμό με τη μετρική γεωμετρικής απόστασης η οποία λειτουργεί σαν φίλτρο περιορίζοντας τους πιθανούς συνδέσμους προς εύρεση σε μία ακτίνα γύρω από κάθε οντότητα και επιτρέποντας μέσω αυτού μεγαλύτερη ανοχή σε διαφορές των συμβολοσειρών.

Τέλος μέσω της ένωσης των συνδέσμων που προέκυψαν από τα πειράματα “g2000/l3/w1” και “String only – l1” ώστε να βρεθούν προκύπτει ένα νέο σετ αποτελεσμάτων που εμφανίζεται στο διάγραμμα ως “g2000/l3/w1 + String only l1”. Το προκύπτον σετ συνδέσμων εμφανίζει μικρή πτώση του precision σε σχέση με τα σετ γονιών του, αλλά παράλληλα εμφανίζει σημαντική αύξηση του recall και τελικά επιτυγχάνει τις καλύτερες τιμές για recall / F-Measure από όλα τα πειράματα που έχουμε διεξάγει. Υπενθυμίζεται ότι η τιμή F-Measure που εμφανίζεται στα διαγράμματα είναι η F1.

4.3.3 Συμπεράσματα

Από τα πειραματικά αποτελέσματα παρατηρήσαμε ότι συνδυασμός των μετρικών ομοιότητας συμβολοσειρών και μετρικών ομοιότητας γεωμετριών μπορεί μέσω κατάλληλης επιλογής των κατωφλίων να προσφέρει σημαντικές βελτιώσεις στην ακρίβεια των παρεχόμενων αποτελεσμάτων χωρίς απώλεια του recall. Η βελτίωση της ακρίβειας αποδίδεται στην λειτουργία της μετρικής γεωμετρικής απόστασης ως φίλτρο που περιορίζει τις πιθανές συνδέσεις για κάθε οντότητα σε έναν δακτύλιο γύρω από αυτή. Επιλέγοντας την ακτίνα του δακτυλίου ώστε να εξασφαλίσουμε υψηλή πιθανότητα ότι όποιες πιθανές συνδέσεις θα βρίσκονται στο εσωτερικό του και απορρίπτοντας όλες τις οντότητες που βρίσκονται στο εξωτερικό του επιτυγχάνουμε τη σημαντική μείωση της πιθανότητας λανθασμένης αναγνώρισης. Παράλληλα διαπιστώνουμε ότι οι συνδυαστικές μετρικές είναι ιδιαίτερα ευαίσθητες ως προς τιμή κατωφλίου γεωμετρικής απόστασης σε σχέση με το recall. Η σωστή επιλογή αυτής της τιμής κρίνεται κρίσιμη για την ποιότητα των αποτελεσμάτων των συνδυαστικών μετρικών και, απουσία πληροφορίας για τα όρια του σφάλματος της γεωμετρικής θέσης, απαιτούνται πειραματικές δοκιμές για τον προσδιορισμό της ως προς τα σετ δεδομένων που εξετάζονται.

Τέλος, αξιοσημείωτος είναι ο συνδυασμός των αποτελεσμάτων της “αυστηρής” μετρικής συμβολοσειρών με κατώφλι απόστασης Levenshtein 1 και της βέλτιστης συνδυαστικής μετρικής “g2000/l3/w1” (Geometry threshold 2000m, Levenshtein distance threshold 3, relative weight 1). Επιτρέποντας εμμέσως την διατήρηση ενός συνδέσμου αν αυτός ικανοποιεί τους όρους οποιασδήποτε από τις δύο μετρικές “γονείς” επιτύχαμε, με μικρή απώλεια της ακρίβειας, καλύτερες τιμές για τα recall / F-Measure από όλες τις υπόλοιπες πειραματικές δοκιμές που διεξάγαμε.

Από τα πειραματικά αποτελέσματα των γεωμετρικών μετρικών συμπεραίνουμε επίσης ότι η χρήση της μετρικής γεωμετρικού containment δεν είναι πιθανό να παράσχει αξιοποιήσιμα αποτελέσματα χωρίς γνώση του εύρος του σφάλματος μέτρησης της γεωμετρικής πληροφορίας στα σετ δεδομένων που διασυνδέονται. Η επιπλέον διακριτική ικανότητα αυτής της μετρικής θα μπορούσε να εμφανιστεί χρήσιμη σε περιπτώσεις όπου το σφάλμα μέτρησης θα επέτρεπε μείωση του κατωφλίου γεωμετρικής απόστασης σε τιμή αντίστοιχης τάξης μεγέθους με τις διαστάσεις τον πολυγώνων που εξετάζονται (π.χ., διαστάσεις μιας κτιριακής εγκατάστασης).

5

Επίλογος

5.1 Σύνοψη

Συνοψίζοντας:

Στην πρώτη φάση της διπλωματικής αναπτύξαμε μία εφαρμογή για την απεικόνιση των δεδομένων του OpenStreetMap σε RDF γράφο σε συμφωνία με το πρότυπο OGC GeoSPARQL και με χρήση των λεξιλογίων του LinkedGeoData project. Για το σκοπό αυτό κάναμε χρήση του υπάρχοντος εργαλείου Osmosis και του συστήματος διαχείρισης βάσης δεδομένων PostgreSQL με το PostGIS extension. Το προκύπτον εργαλείο Geosm επιτρέπει την γρήγορη παραγωγή RDF δεδομένων από OSM data dumps μέσω φιλικών προς το χρήστη διεπαφών και προσφέρει μία πλατφόρμα για την προσθήκη βημάτων προεπεξεργασίας για την κάλυψη των απαιτήσεων του τελικού χρήστη.

Στη δεύτερη φάση της διπλωματικής διερευνήσαμε τις δυνατότητες διασύνδεσης γεωχωρικών σημασιολογικών δεδομένων. Συλλέξαμε τα απαραίτητα για τις πειραματικές δοκιμές δεδομένα και τα μετατρέψαμε σε GeoSPARQL συμβατούς RDF γράφους για εισαγωγή στο εργαλείο Silk. Στη συνέχεια αναπτύξαμε και υλοποιήσαμε νέες μετρικές για την ομοιότητα γεωμετριών και νέες συνδυαστικές μετρικές για την ομοιότητα συμβολοσειρών και τις δοκιμάσαμε πειραματικά. Τέλος εξετάσαμε συνδυαστικές μετρικές που κάνουν ταυτόχρονη χρήση συμβολοσειρών και γεωμετριών και δώσαμε γενικά συμπεράσματα που μπορούν να εφαρμοστούν για διασύνδεση γεωχωρικής σημασιολογικής πληροφορίας σε ένα εύρος συμβατών σεναρίων χρήσης.

5.2 Μελλοντικές επεκτάσεις

Για το εργαλείο Geosm προτείνονται ως πιθανές επεκτάσεις:

- Προσθήκη δυνατότητας εξαγωγής των RDF γράφων σε άλλους ευρέως χρησιμοποιήσιμους μορφότυπους για σημασιολογικά δεδομένα πέραν των N-TRIPLES, N-QUADS όπως για παράδειγμα RDF/XML, TURTLE και N3,
- Προσθήκη δυνατότητας εξαγωγής των RDF γράφων σε αποθήκες σημασιολογικών δεδομένων μέσω του πρωτοκόλλου SPARQL,
- Παραμετροποίηση των επερωτήσεων SQL και των απεικονίσεων SQL table attributes -> RDF ώστε το εργαλείο να καταστεί ανθεκτικό σε αλλαγές του OSM pgsnapshot schema και να μπορεί να επεκταθεί για χρήση με άλλες πηγές δεδομένων.

Για το τμήμα των μετρικών διασύνδεσης προτείνονται ως πιθανές επεκτάσεις:

- Διερεύνηση της δυνατότητας επεξεργασίας συμβολοσειρών που έχουν παραχθεί με σύνθεση ονομάτων συστεγασμένων εγκαταστάσεων (π.χ., "7ο, 8ο, 12ο, 15ο Δημοτικά Σχολεία Ζωγράφου"),
- Διερεύνηση της δυνατότητας επιλογής του σχετικού βάρους μεταξύ μετρικών ομοιότητας συμβολοσειρών και μετρικών ομοιότητας γεωμετριών και του κατωφλίου γεωμετρικής απόστασης σε περιπτώσεις συναρτήσει του σφάλματος μέτρησης γεωμετρικής πληροφορίας όπου αυτό είναι γνωστό,
- Διερεύνηση της επίδρασης των κατωφλίων απόστασης στις μετρικές ομοιότητας γεωμετριών σε σετ δεδομένων προερχόμενων από άλλες πηγές,
- Διερεύνηση της δυνατότητας διόρθωσης ορθογραφικών λαθών προ της εφαρμογής μετρικών ομοιότητας συμβολοσειρών,
- Διερεύνηση της εφαρμογής των περιγραφόμενων μετρικών ομοιότητας συμβολοσειρών σε γλώσσες πέραν της Ελληνικής.

6

Παράρτημα

6.1 Namespaces

```
geo: <http://www.opengis.net/ont/geosparql#>
geof: <http://www.opengis.net/def/function/geosparql/>
geor: <http://www.opengis.net/def/rule/geosparql/>
gml: <http://www.opengis.net/ont/gml#>
lgdg: <http://linkedgeodata.org/geometry/>
lgdo: <http://linkedgeodata.org/ontology/>
lgdt: <http://linkedgeodata.org/triplify/>
ogc: <http://www.opengis.net/>
owl: <http://www.w3.org/2002/07/owl#>
pos: <http://www.w3.org/2003/01/geo/wgs84_pos#>
rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
rdfs: <http://www.w3.org/2000/01/rdf-schema#>
sf: <http://www.opengis.net/ont/sf#>
uom: <http://www.opengis.net/def/uom/OGC/1.0/>
xsd: <http://www.w3.org/2001/XMLSchema#>
```

Η σελίδα αυτή είναι σκόπιμα λευκή.

7

Βιβλιογραφία

[CFO93] E. Clementini, P. Di Felice, and P. van Oosterom. A Small Set of Formal Topological Relationships Suitable for End-user Interaction. In Proceedings of the 3rd International Symposium on Advances in Spatial user Databases (SSD ’93), pp. 277–295, London, UK, 1993.

[CommonsCLI] Apache Commons CLI Library. Available at http://commons.apache.org/proper/commons-cli/

[CSE94] E. Clementini, J. Sharma, and M.J. Egenhofer. Modelling Topological Spatial Relations: Strategies for Query Processing. Computers & Graphics, 18(6):815 – 822, 1994.

[EF91] M.J. Egenhofer and R. Franzosa. Point Set Topological Spatial Relations. International Journal of Geographical Information Systems, 5(2):161-174, 1991.

[GeodataGov] geodata.gov.gr. Available at http://geodata.gov.gr

[GeoPos84] Basic Geo (WGS84 lat/long) Vocabulary. Available at http://www.w3.org/2003/01/geo/

[GRefine] Google Refine tool for data cleanup and transformation. Available at https://code.google.com/p/google-refine/

[Jena] Apache Jena project. Available at http://jena.apache.org/

[JTS] JTS Topology Suite. Available at http://tsusiatsoftware.net/jts/main.html

[LGD] LinkedGeoData. Available at http://linkedgeodata.org/

[MyBatis] MyBatis persistence framework. Available at http://mybatis.github.io/mybatis-3/

[NT] N-Triples format. Available at http://www.w3.org/2001/sw/RDFCore/ntriples/

[OGC07] Open Geospatial Consortium Inc. OpenGIS Geography Markup Language (GML) Encoding Standard. Version 3.2.1, 27/08/2007. Available at http://portal.opengeospatial.org/files/?artifact_id=20509s

[OGC10a] Open Geospatial Consortium Inc. OpenGIS Geography Markup Language (GML) Simple Features Profile. Version 2.0, 07/10/2010. Available at **http://portal.opengeospatial.org/files/?artifact_id=39853**

[OGC10b] Open Geospatial Consortium Inc. OpenGIS Implementation Specification for Geographic information - Simple feature access Part 2: SQL option. OpenGIS Implementation Standard. Version 1.2.1, 04/08/2010. Available at **http://portal.opengeospatial.org/files/?artifact_id=25354**

[OGC11] Open Geospatial Consortium Inc. OpenGIS Implementation Standard for Geographic information - Simple Feature Access - Part 1: Common Architecture. OpenGIS Implementation Standard. Version 1.2.1, 28/05/2011. Available at http://portal.opengeospatial.org/files/?artifact_id=25355

[OGC12] Open Geospatial Consortium Inc. OGC GeoSPARQL standard - A geographic query language for RDF data. Version 1.0, 27/04/2012. Available at https://portal.opengeospatial.org/files/?artifact_id=47664

[OSM] OpenStreetMap project. Available at http://www.openstreetmap.org/

[OSM/XML] OSM XML file format. Available at http://wiki.openstreetmap.org/wiki/OSM_XML

[Osmosis] Command line utility for processing OSM data. Available at http://wiki.openstreetmap.org/wiki/Osmosis

[OWL] OWL Web Ontology Language Overview. Avaiable at http://www.w3.org/TR/owl-features/

[PBF] PBF file format. Available at http://wiki.openstreetmap.org/wiki/Pbf

[POIs] POIs.gr. Available at http://pois.gr/

[PostGIS] PostGIS – Spatial and Geographic objects for PostgresQL. Available at http://postgis.net/

[PostgreSQL] PostgreSQL Database. Available at http://www.postgresql.org/

[PostgreSQLJDBC] PostgreSQL JDBC driver. Available at http://jdbc.postgresql.org/

[RCC92] D. A. Randell, Z. Cui, and A. G. Cohn. A spatial logic based on regions and connection. In Proceedings of the 3rd International Conference on Knowledge Representation and Reasoning, pp. 165 -176, 1992.

[RDF Refine] RDF Refine. Google Refine extension for exporting RDF. Available at http://refine.deri.ie/

[RDF] Resource Description Framework Primer. Available at http://www.w3.org/TR/rdf/

[RDFS] RDF Schema. Available at http://www.w3.org/TR/rdf-schema/

[Silk] Silk Link Discovery Framework. Available at https://www.assembla.com/spaces/silk

[SPARQL1] SPARQL Query Language for RDF. Available at http://www.w3.org/TR/rdf-sparql-query/

[SPARQL2] SPARQL 1.1 Query Language. Available at http://www.w3.org/TR/sparql11-query/

[Sparqlify] SPARQL->SQL rewriter. Available at http://aksw.org/Projects/Sparqlify.html

[w3c-geo] W3C. WGS84 Geo Positioning: an RDF vocabulary Available at http://www.w3.org/2003/01/geo/wgs84_pos#

[WikiMapia] WikiMapia project. Available at http://wikimapia.org

[ΕΛΟΤ743] ΕΛΟΤ 743 Ε2. Μετατροπή των Ελληνικών χαρακτήρων με χαρακτήρες Λατινικούς. Available at https://sales.elot.gr/online/search/details.do?documentId=300010000020380