



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΣΥΣΤΗΜΑΤΑ ΚΑΙ ΤΕΧΝΟΛΟΓΙΕΣ ΓΝΩΣΗΣ

ΕΞΑΜΗΝΙΑΙΟ ΘΕΜΑ

ΦΥΤΙΖΑ ΓΕΩΡΓΙΑ

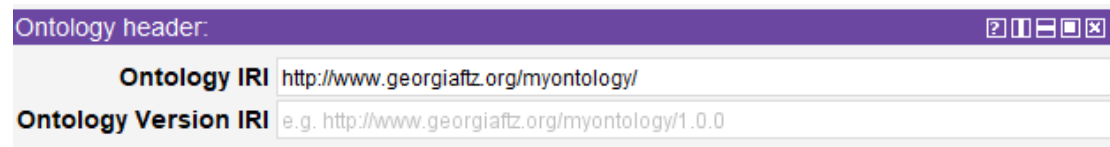
03116177

ΟΝΤΟΛΟΓΙΑ

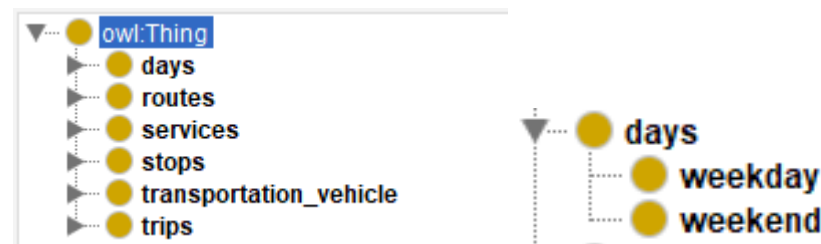
Βασιμμένοι στα δεδομένα που μας δώθηκαν από τα Δρομολόγια Αστικών Συγκοινωνιών Αθήνας κατασκευάστηκε μια οντολογία με πεδίο τα μέσα μεταφοράς. Προφανώς, το πεδίο αυτό είναι πιο ευρύ από τα δεδομένα που μας δίνονται τα οποία αφορούν αποκλειστικά τα αστικά μέσα, συνεπώς η οντολογία που κατασκευάστηκε προσπάθησε να είναι όσο πιο γενική γίνεται, ούτως ώστε να μπορεί να περιγράψει ολόκληρο το πεδίο των μέσων μεταφοράς.

Για την κατασκευή της οντολογίας, χρησιμοποιήθηκε το εργαλείο Protégé, όπως προτάθηκε και από τους διδάσκοντες, το οποίο παρέχει ένα γραφικό περιβάλλον για την κατασκευή της οντολογίας.

Αρχικά, ορίσα το βασικό URI της οντολογίας μου.

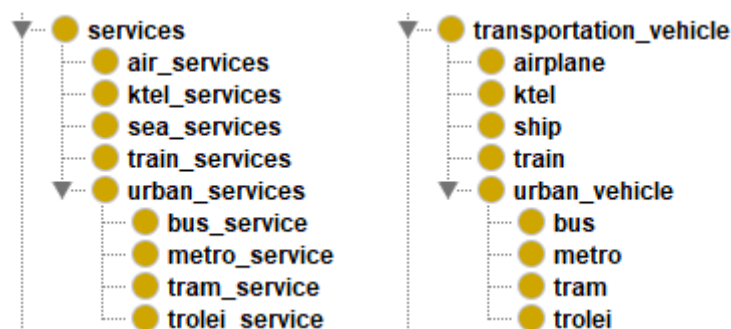


Στην συνέχεια ορίστηκαν οι απαραίτητες κλάσεις για την οντολογία μου.

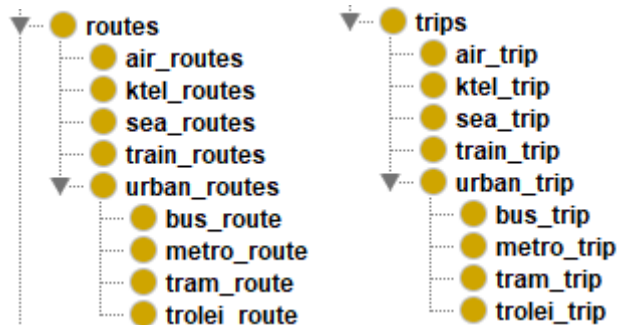


Πιο αναλυτικά, η κλάση days χωρίστηκε σε δύο υποκλάσεις τις καθημέρινες και τα Σαββατοκύριακα. Η χρησιμοτητα της είναι προφανείς καθώς μιλάμε για μέσα μεταφορών που λειτουργούν συγκεκριμένες μέρες, οι οποίες δηλώνονται κατά μοναδικό τρόπο με τα αντίστοιχα URIs.

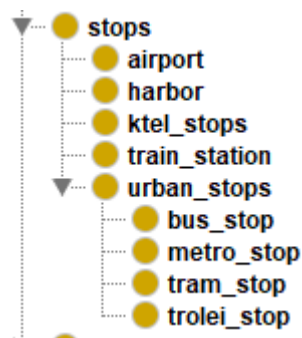
Κατόπιν, ορίζουμε τις υπηρεσίες που παρέχονται από τα αντίστοιχα μέσα. Οι υπηρεσίες διαχωρίζονται όπως φαίνεται και παρακάτω σε υποκλάσεις σύμφωνα με τον τύπο των μέσων που έχουν για στιγμιότυπα. Η ίδια λογική ισχύει και για τα οχήματα/μέσα μεταφοράς.



Οι άλλες δύο κλάσεις που δεν πρέπει να λείπουν από την οντολογία μας είναι οι διαδρομές και τα δρομολόγια, που έχουν ως στιγμιότυπα URIs που αντιπροσωπεύουν τα αντίστοιχα `route_id`, `trip_id` που υπάρχουν στις βάσεις δεδομένων των μέσων μεταφοράς και τα χαρακτηρίζουν μοναδικά. Και εδώ είναι απαραίτητο να γίνει χωρισμός υποκλάσεων καθώς το πεδίο μας είναι αρκετά μεγάλο και θα πρέπει να έχουμε την δυνατότητα διαχωρισμού των επιμέρους πεδίων.

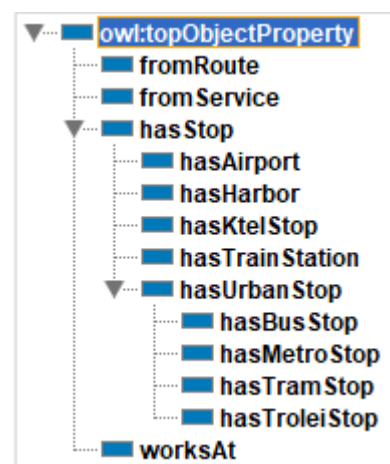


Τέλος, ορίζουμε σαν ξεχωριστή κλάση τις στάσεις. Είναι μια λογική προσθήκη καθώς επίσης λογικός μοιάζει ο διαχωρισμός της σε επιμέρους κλάσεις όπως `bus_stop`, `train_station`, `airport` κ.ο.κ.



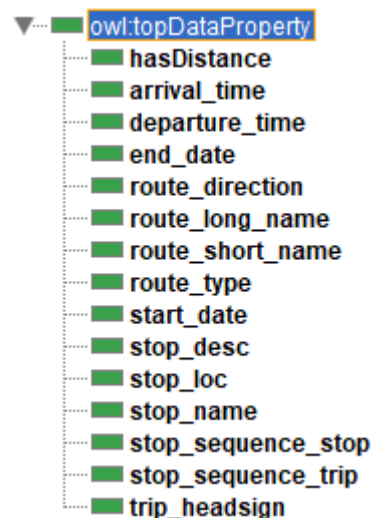
Σημείωση: εφόσον ορίστηκαν οι κλάσεις, ορίστηκαν και τα απαραίτητα αξιώματα, όπως το ποια κλάση είναι υποκλάση ποιας άλλης κλάσης, καθώς και ότι όλες οι κλάσεις είναι ξένες μεταξύ τους.

Εφόσον ορίστηκαν οι κλάσεις, ακολούθησε ορισμός ιδιοτήτων μεταξύ των κλασεων.



Οι δύο πρώτες ιδιότητες συνδέουν ένα δρομολόγιο με μια διαδρομή και με μια υπηρεασία, όπως φανερώνεται και από τα ονοματά τους. Η ιδιότητα `hasStop` ενώνει ένα δρομολόγιο με μια στάση, και προφανώς δηλώνει ότι ένα δρομολόγιο έχει στάση με ένα συγκεκριμένο `stop_id` – URI. Τέλος, η `worksAt` συνδέει ένα `service` με μια ημέρα, και δηλώνει ποιες μέρες παρέχεται, ποια υπηρεασία.

Αφού οριστήκαν και οι Object Properties, ακολούθησαν οι Data Properties που ενώνουν τα στιγμιότυπα μια κλάσεις με λεκτικά. Όπως ειπώθηκε και στο μαθήμα δεν χρειάζεται στα πλαίσια της εργασίας να ορισθούν με ακρίβεια όλα τα πεδία τιμών όλων των ιδιοτήτων. Συνεπώς, θα περιγραφθούν λίγο πιο αφαιρετικά οι συγκεκριμένες ιδιότητες.



Η πρώτη ιδιότητα ενώνει μια διαδρομή με έναν ακέραιο και δηλώνει πόσα χιλιόμετρα είναι η συγκεκριμένη διαδρομή.

Οι `start_date`, `end_date` ενώνουν υπηρεασία με έναν 8ψήφιο, ο οποίος αναπαριστά μια ημερομηνία (πχ. 20160812) και ερμηνεύονται ως οι ημερομηνίες που ξεκινά να παρέχεται μια υπηρεασία και η ημερομηνίες που σταματάει.

Η ιδιότητα `route_direction` συνδέει ένα στιγμιότυπο της κλάσεις `trip` με έναν ακέραιο (0 ή 1) ο οποίος δηλώνει την κατεύθυνση του αντίστοιχου δρομολογίου.

Στην συνέχεια, οι ιδιότητες `route_long_name` και `route_short_name` ενώνουν μια διαδρομή με έναν `string` το οποίο στην πρώτη περίπτωση είναι το μεγάλο όνομα της διαδρομής και στην δεύτερη περίπτωση το σύντομο όνομα. Επίσης, έχει οριστεί και η ιδιότητα `route_type` που ενώνει μια διαδρομή με έναν ακέραιο ο οποίος δηλώνει το είδος της διαδρομής. Για παράδειγμα αν ο ακέραιος είναι 3 θεωρούμε ότι αναφερόμαστε με μια διαδρομή λεωφορίου, ενώ αν είναι 0 θεωρούμε ότι αναφερόμαστε σε διαδρομή μετρό. Τέλος, έχουμε την `trip_headsign` η οποία συνδέει την διαδρομή με τον τίτλο της, εννοώντας με την επικεφαλίδα που αναγράφεται και είναι γνωστή στους επιβάτες.

Τέλος έχουμε ορίσει αντίστοιχες ιδιότητες και για τις στάσεις. Αυτές οι ιδιότητες είναι η `stop_desc` που αναφέρεται σε ένα `string` που περιέχει μια περιγραφή στάσης, η `stop_name` που αναφέρεται επίσης σε `string` που περιέχει το όνομα της στάσης και η `stop_loc` που αναφέρεται στις γεωγραφικές συντεταγμένες μιας στάσης σε μορφή "POINT(12.3830858

51.3465518)“^<http://www.openlinksw.com/schemas/virtrdf#Geometry> , όπως προτείνεται από τους διδάσκοντες.

Πιο αναλυτικά φαίνεται η οντολογία στο αρχείο myontology.owl που υπάρχει στο ζιπαρμένο παραδοτέο.

ΜΕΤΑΤΡΟΠΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΚΩΔΙΚΑΣ

Για την μετατροπή των αρχείων των δεδομένων σε RDF 3άδες χρησιμοποιήθηκε η γλώσσα προγραμματισμού Java. Οι τριάδες που παράχθηκαν είναι σε γλώσσα Turtle, όπως προτάθηκε και στην εκφώνηση, έτσι ώστε να μπορούν να περαστούν στην συνέχεια στην αποθήκη τριάδων που μας παρέχει το Virtuso.

Πιο αναλυτικά για τα προγράμματα σε Java που γραφτηκάν χρησιμοποιήθηκε ως editor ο IntelliJ, ως Build tool maven και ως βιβλιοθήκες η openscn και η Apache Jena.

Παρακάτω φαίνεται ένα αρχείο κώδικα μετροπής καθώς επίσης και ενδεικτικά κάποια κομμάτια ενός αρχείου εισόδου και του αντίστοιχου εξόδου ώστε να γίνει εμφανής η μετατροπή.

```
package com.example.Reader;

import ...

public class tripReader {
    private Model model;
    private static final String SAMPLE_CSV_FILE_PATH = "./trips.txt";
    OutputStream out1 = new FileOutputStream( name: "trips-data.ttl");

    public tripReader() throws FileNotFoundException {
    }

    public void writeModel() {
        model.write(System.out, s: "TTL");
        RDFDataMgr.write(out1, model, Lang.TURTLE);
    }

    public void addStatement(String s, String p, String o){
        Resource subject = model.createResource(s);
        Property predicate = model.createProperty(p);
        RDFNode object = model.createResource(o);
        Statement stmt = model.createStatement(subject, predicate, object);
        model.add(stmt);
    }
}
```

```

public void createModel(String [] nextRecord){
    model = ModelFactory.createDefaultModel();
    String ns = "http://www.georgiaftz.org/myontology/";
    String nsRDFS = "http://www.w3.org/2000/01/rdf-schema#";

    addStatement( s: ns+nextRecord[2], p: nsRDFS+"type", o: ns+"trips");
    addStatement( s: ns+nextRecord[2], p: ns+"fromService", o: ns+nextRecord[1]);
    addStatement( s: ns+nextRecord[2], p: ns+"fromRoute", o: ns+nextRecord[0]);
    Resource route = model.getResource( s: ns+nextRecord[2]);
    route.addLiteral(model.createProperty( s: ns+"trip_headsign"), model.createTypedLiteral(new String( nextRecord[3] )));
    route.addLiteral(model.createProperty( s: ns+"route_direction"), model.createTypedLiteral(new Integer( nextRecord[4] )));
}

public static void main(String[] args) throws IOException {
    BasicConfigurator.configure();
    tripReader dave = new tripReader();
    Reader reader = Files.newBufferedReader(Paths.get(SAMPLE_CSV_FILE_PATH));
    CSVReader csvReader = new CSVReader(reader);
    // Reading Records One by One in a String array
    String[] nextRecord;
    nextRecord = csvReader.readNext();
    while ((nextRecord = csvReader.readNext()) != null) {
        dave.createModel(nextRecord);
        dave.writeModel();
    }
}

```

Δεδομένα εισόδου

```

route_id,service_id,trip_id,trip_headsign,direction_id,block_id,shape_id
T3-20,THAEMA-T3-Παρασκευή-02,9803638-THAEMA-T3-Παρασκευή-02,NEO ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ,0,2184458,
T3-20,THAEMA-T3-Παρασκευή-02,9803639-THAEMA-T3-Παρασκευή-02,NEO ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ,0,2184460,
T3-20,THAEMA-T3-Παρασκευή-02,9803640-THAEMA-T3-Παρασκευή-02,NEO ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ,0,2184459,
T3-20,THAEMA-T3-Παρασκευή-02,9803641-THAEMA-T3-Παρασκευή-02,NEO ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ,0,2184461,
T3-20,THAEMA-T3-Παρασκευή-02,9803642-THAEMA-T3-Παρασκευή-02,NEO ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ,0,2184462,
T3-20,THAEMA-T3-Παρασκευή-02,9803643-THAEMA-T3-Παρασκευή-02,NEO ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ,0,2184462,
T3-20,THAEMA-T3-Παρασκευή-02,9803644-THAEMA-T3-Παρασκευή-02,NEO ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ,0,2184458,
T3-20,THAEMA-T3-Παρασκευή-02,9803645-THAEMA-T3-Παρασκευή-02,NEO ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ,0,2184460,
T3-20,THAEMA-T3-Παρασκευή-02,9803646-THAEMA-T3-Παρασκευή-02,NEO ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ,0,2184459,

```

Δεδομένα εξόδου

```

<http://www.georgiaftz.org/myontology/9803638-THAEMA-T3-Παρασκευή-02>
  <http://www.w3.org/2000/01/rdf-schema#type>
    <http://www.georgiaftz.org/myontology/trips> ;
  <http://www.georgiaftz.org/myontology/fromRoute>
    <http://www.georgiaftz.org/myontology/T3-20> ;
  <http://www.georgiaftz.org/myontology/fromService>
    <http://www.georgiaftz.org/myontology/THAEMA-T3-Παρασκευή-02> ;
  <http://www.georgiaftz.org/myontology/route_direction>
    "0"^^<http://www.w3.org/2001/XMLSchema#int> ;
  <http://www.georgiaftz.org/myontology/trip_headsign>
    "NEO ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ" .
<http://www.georgiaftz.org/myontology/9803639-THAEMA-T3-Παρασκευή-02>
  <http://www.w3.org/2000/01/rdf-schema#type>
    <http://www.georgiaftz.org/myontology/trips> ;
  <http://www.georgiaftz.org/myontology/fromRoute>
    <http://www.georgiaftz.org/myontology/T3-20> ;
  <http://www.georgiaftz.org/myontology/fromService>
    <http://www.georgiaftz.org/myontology/THAEMA-T3-Παρασκευή-02> ;
  <http://www.georgiaftz.org/myontology/route_direction>
    "0"^^<http://www.w3.org/2001/XMLSchema#int> ;
  <http://www.georgiaftz.org/myontology/trip_headsign>
    "NEO ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ" .
<http://www.georgiaftz.org/myontology/9803640-THAEMA-T3-Παρασκευή-02>

```

Στην συνέχεια τα αρχεία με της τριάδες καθώς και το .owl αρχείο μας περάστηκα στο Virtuoso με την ακόλουθη διαδικασία.

Κατέβασα το Virtuoso OpenSource 7.2 και στην συνέχεια εκκίνησα τον server. Κατόπιν, μπήκα στο <http://localhost:8890/> και συνδέθηκα ως admin με credentials (dba, dba). Τέλος, φορτώσα ένα ένα τα αρχεία που δημιουρήσαν στην σελίδα που φαίνεται παρακάτω.

The screenshot shows the Virtuoso Conductor web interface. The browser address bar displays `localhost:8890/conductor/rdf_import.vsp?sid=d0147104693f630fbae495d26a91db47&realm=virtuoso_admin`. The interface includes a sidebar with links like 'Interactive SQL (ISQL)', 'WebDAV Browser', and 'Documentation (web)'. The main menu has tabs for 'Home', 'System Admin', 'Database', 'Replication', 'Web Application Server', 'XML', 'Web Services', 'Linked Data', and 'NNTP'. The 'Linked Data' tab is active, showing a 'Quad Store Upload' dialog. In this dialog, the 'File*' option is selected with the filename 'trips-data.ttl'. The 'Named Graph IRI*' field contains 'http://localhost:8890/DAV'. There are 'Cancel' and 'Upload' buttons at the bottom right of the dialog.

SPARQL ΕΡΩΤΗΜΑΤΑ

Ένα πρώτο ενδεικτικό απλό παράδειγμα sparql ερωτήματος είναι να θέλεις να αναζητήσεις τις στάσεις ενός δρομολογίου.

The screenshot shows the Virtuoso Conductor web interface with a SPARQL query entered and its results displayed. The 'Default Graph IRI' field is set to 'http://localhost:8890/DAV'. The query is as follows:

```
SELECT ?stops {
  <http://www.georgiaftz.org/myontology/9803638-ΤΗΛΕΜΑ-Τ3-Παρασκευή-02>
    <http://www.georgiaftz.org/myontology/hasStop> ?stops .
}
```

Below the query, there are buttons for 'Execute', 'Save', 'Load', and 'Clear'. The results are displayed in a table with the following data:

stops
http://www.georgiaftz.org/myontology/080023
http://www.georgiaftz.org/myontology/080024
http://www.georgiaftz.org/myontology/080073
http://www.georgiaftz.org/myontology/080074
http://www.georgiaftz.org/myontology/080075
http://www.georgiaftz.org/myontology/160041
http://www.georgiaftz.org/myontology/160045
http://www.georgiaftz.org/myontology/160056

Πιο πρακτικό θα ήταν αν επιστρέφαμε το όνομα των στάσεων και όχι τι id τους. Αυτό φαίνεται παρακάτω.

Query

```
SELECT ?stops, ?stop_name, ?stop_desc {
<http://www.georgiaftz.org/myontology/9803638-ΤΗΛΕΜΑ-Τ3-Παρασκευή-02>
  <http://www.georgiaftz.org/myontology/hasStop> ?stops .
?stops <http://www.georgiaftz.org/myontology/stop_name> ?stop_name .
?stops <http://www.georgiaftz.org/myontology/stop_desc> ?stop_desc .
}
```

Execute

Save

Load

Clear

stops	stop_name	stop_desc
http://www.georgiaftz.org/myontology/080023	"ΠΙΚΡΟΔΑΦΝΗ"	"Επί της ΛΕΩΦ.ΠΟΣΕΙΔΩΝΟΣ"
http://www.georgiaftz.org/myontology/080024	"ΜΑΡΙΝΑ ΑΛΙΜΟΥ"	"Επί της ΛΕΩΦ.ΠΟΣΕΙΔΩΝΟΣ"
http://www.georgiaftz.org/myontology/080073	"ΚΑΛΑΜΑΚΙ"	"Επί της ΛΕΩΦ.ΠΟΣΕΙΔΩΝΟΣ"
http://www.georgiaftz.org/myontology/080074	"ΖΕΦΥΡΟΣ"	"Επί της ΛΕΩΦ.ΠΟΣΕΙΔΩΝΟΣ"
http://www.georgiaftz.org/myontology/080075	"ΛΟΥΤΡΑ ΑΛΙΜΟΥ"	"Επί της ΛΕΩΦ.ΠΟΣΕΙΔΩΝΟΣ"

Τώρα θα υλοποιηθούν κάποια πιο περίπλοκα ερωτήματα, όπως η αναζήτηση των κοντινών γεωγραφικά στάσεων σε ένα σημείο. Με το παρακάτω ερώτημα βρίσκουμε ποιες στασεις απέχουν το πολύ ενά χιλιόμετρο από το σημείο (37.9946947354979,23.6611529618238).

Query

```
SELECT ?stop_id ?name ?desc
WHERE
{
  ?stop_id <http://www.georgiaftz.org/myontology/stop_loc> ?geo .
  ?stop_id <http://www.georgiaftz.org/myontology/stop_name> ?name .
  ?stop_id <http://www.georgiaftz.org/myontology/stop_desc> ?desc .
  FILTER (bif:st_intersects (?geo, bif:st_point
(37.9946947354979,23.6611529618238), 1))
}
```


stop_id	name	desc
http://www.georgiaftz.org/myontology/010001	"ΣΤΡΟΦΗ"	"Επί της ΕΛ.ΒΕΝΙΖΕΛΟΥ"
http://www.georgiaftz.org/myontology/010002	"ΚΟΛΩΝΕΣ"	"Επί της ΛΕΩΦ.ΕΛ.ΒΕΝΙΖΕΛΟΥ"
http://www.georgiaftz.org/myontology/010003	"ΑΓ.ΒΑΡΒΑΡΑ"	"Επί της ΛΕΩΦ.ΕΛ.ΒΕΝΙΖΕΛΟΥ"
http://www.georgiaftz.org/myontology/010004	"ΔΗΜΑΡΧΕΙΟ"	"Επί της ΗΠΕΙΡΟΥ"
http://www.georgiaftz.org/myontology/010005	"ΚΡΗΤΗΣ"	"Επί της ΠΑΛΑΙΩΝ ΠΟΛΕΜΙΣΤΩΝ"
http://www.georgiaftz.org/myontology/010006	"ΚΟΥΝΤΟΥΡΙΩΤΟΥ"	"Επί της ΠΑΛΑΙΩΝ ΠΟΛΕΜΙΣΤΩΝ"
http://www.georgiaftz.org/myontology/010007	"ΔΗΜΑΡΧΕΙΟ ΑΓ.ΒΑΡΒΑΡΑΣ"	"Επί της ΛΕΩΦ.ΕΛ.ΒΕΝΙΖΕΛΟΥ"
http://www.georgiaftz.org/myontology/010008	"1Η ΑΓ. ΜΑΡΙΝΑΣ"	"Επί της ΑΓ.ΜΑΡΙΝΑΣ"
http://www.georgiaftz.org/myontology/010011	"ΘΕΜΙΣΤΟΚΛΕΟΥΣ"	"Επί της ΠΑΛ.ΠΑΤΡΩΝ ΓΕΡΜΑΝΟΥ"
http://www.georgiaftz.org/myontology/010013	"ΒΕΝΕΤΙΑ"	"Επί της ΛΕΩΦ.ΜΕΓ.ΑΛΕΞΑΝΔΡΟΥ"
http://www.georgiaftz.org/myontology/010014	"ΑΦΕΤΗΡΙΑ 837"	"Επί της ΠΑΛ.ΠΑΤΡΩΝ ΓΕΡΜΑΝΟΥ"

Λίγο πιο ολοκληρωμένο ερώτημα από το παραπάνω είναι αυτό που εμφανίζει τα αποτελέσματα ταξινομημένα από την πιο κοντινή στάση στην πιο μακρινή και με περιορισμό να εμφανιστούν μόνο 20 αποτελέσματα.

Default Graph IRI

Query

```

SELECT ?stop_id ?name ?desc (bif:st_distance (?geo, bif:st_point
(37.9946947354979,23.6611529618238)))
WHERE
{
  ?stop_id <http://www.georgiaftz.org/myontology/stop_loc> ?geo .
  ?stop_id <http://www.georgiaftz.org/myontology/stop_name> ?name .
  ?stop_id <http://www.georgiaftz.org/myontology/stop_desc> ?desc .
  FILTER (bif:st_intersects (?geo, bif:st_point
(37.9946947354979,23.6611529618238), 1))
}
ORDER BY 4
LIMIT 20.

```

stop_id	name	desc	callret-3
http://www.georgiaftz.org/myontology/010003	"ΑΓ.ΒΑΡΒΑΡΑ"	"Επί της ΑΕΩΦ.ΕΛ.ΒΕΝΙΖΕΛΟΥ"	0.0
http://www.georgiaftz.org/myontology/010024	"ΑΓ.ΒΑΡΒΑΡΑ"	"Επί της ΑΕΩΦ.ΕΛ.ΒΕΝΙΖΕΛΟΥ"	0.123333
http://www.georgiaftz.org/myontology/010036	"ΙΚΑ"	"Επί της ΑΕΩΦ.ΕΛ.ΒΕΝΙΖΕΛΟΥ"	0.214887
http://www.georgiaftz.org/myontology/010027	"28ης ΟΚΤΩΒΡΙΟΥ"	"Επί της 28ης ΟΚΤΩΒΡΙΟΥ"	0.234493
http://www.georgiaftz.org/myontology/010031	"28ης ΟΚΤΩΒΡΙΟΥ"	"Επί της 28ης ΟΚΤΩΒΡΙΟΥ"	0.235229
http://www.georgiaftz.org/myontology/010087	"ΚΥΠΡΟΥ"	"Επί της ΧΙΟΥ"	0.239834

Ένα άλλο πρακτικό ερώτημα είναι αν μπορεί να μεταβεί κανείς από μια στάση σε μια άλλη χωρίς αλλαγή μέσου ή με αλλαγή το πολύ ενός μέσου. Αυτό μπορεί να υλοποιηθεί όπως φαίνεται ακολούθως.

Έστω ότι κάποιος θέλει να μετακινηθεί μεταξύ των στάσεων με stop_id 400045 και 380112 .

Default Graph IRI

Query

```

SELECT DISTINCT ?trip_headname ?trip1_headname ?trip2_headname ?change_stop
{
  {
    ?trip <http://www.georgiaftz.org/myontology/hasStop>
      <http://www.georgiaftz.org/myontology/400045> .
    ?trip <http://www.georgiaftz.org/myontology/hasStop>
      <http://www.georgiaftz.org/myontology/380112> .
    ?trip <http://www.georgiaftz.org/myontology/trip_headsign> ?trip_headname .
    ?trip <http://www.georgiaftz.org/myontology/fromRoute> ?route .
  }
  UNION
  {
    ?trip1 <http://www.georgiaftz.org/myontology/hasStop>
      <http://www.georgiaftz.org/myontology/400045> .
    ?trip2 <http://www.georgiaftz.org/myontology/hasStop>
      <http://www.georgiaftz.org/myontology/380112> .
    ?trip1 <http://www.georgiaftz.org/myontology/hasStop> ?change_stop .
    ?trip2 <http://www.georgiaftz.org/myontology/hasStop> ?change_stop .
    ?trip1 <http://www.georgiaftz.org/myontology/trip_headsign> ?trip1_headname .
    ?trip2 <http://www.georgiaftz.org/myontology/trip_headsign> ?trip2_headname .
  }
}

```

trip_headname	trip1_headname	trip2_headname	change_stop
"ΝΕΟ ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ"			
" ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ - ΝΕΟ ΦΑΛΗΡΟ "			
"ΣΥΝΤΑΓΜΑ - ΝΕΟ ΦΑΛΗΡΟ"			
" ΝΕΟ ΦΑΛΗΡΟ - ΣΥΝΤΑΓΜΑ "			
	"ΣΥΝΤΑΓΜΑ - ΝΕΟ ΦΑΛΗΡΟ"	"ΣΥΝΤΑΓΜΑ - ΝΕΟ ΦΑΛΗΡΟ"	http://www.georgiaftz.org/myontology/060404
	" ΝΕΟ ΦΑΛΗΡΟ - ΣΥΝΤΑΓΜΑ "	"ΣΥΝΤΑΓΜΑ - ΝΕΟ ΦΑΛΗΡΟ"	http://www.georgiaftz.org/myontology/060404
	"ΣΥΝΤΑΓΜΑ - ΝΕΟ ΦΑΛΗΡΟ"	" ΝΕΟ ΦΑΛΗΡΟ - ΣΥΝΤΑΓΜΑ "	http://www.georgiaftz.org/myontology/060404
	" ΝΕΟ ΦΑΛΗΡΟ - ΣΥΝΤΑΓΜΑ "	" ΝΕΟ ΦΑΛΗΡΟ - ΣΥΝΤΑΓΜΑ "	http://www.georgiaftz.org/myontology/060404
	"ΝΕΟ ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ"	"ΝΕΟ ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ"	http://www.georgiaftz.org/myontology/080023
	" ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ - ΝΕΟ ΦΑΛΗΡΟ "	"ΝΕΟ ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ"	http://www.georgiaftz.org/myontology/080023

Όπως φαίνεται παραπάνω μπορούμε να πάμε από την μια στάση στην άλλη με ένα δρομολόγιο αλλά μπορούμε να χρησιμοποιήσουμε και 2. Έχουμε επιλέξει να εμφανίζεται στα αποτελέσματα τόσο το μονό δρομολόγιο όσο και τα διπλά. Στην δεύτερη περίπτωση εμφανίσαμε και την στάση στην οποία θα γίνει αλλαγή. Αυτό το πετυχαίνουμε μέσω τις ιδιότητας που μας δίνει η SPARQL να συνενώσουμε δύο αποτελέσματα μέσω του UNION. Παρατηρούμε ότι στο συγκεκριμένο παράδειγμα οι στάσεις που επιλέχθηκαν είναι στην ίδια γραμμή του τράμ συνεπώς δεν έχει πολύ νοημα η αλλαγή δρομολογίου. Παρόλα αυτά το ερωτημά μας που θα ανατρέξει στην βάση γνώσης θα μας επιστρέψει και αυτά τα αποτελέσματα. Ακόμη αξίζει να σημειωθεί ότι δεδομένου ότι δεν δώθηκε κατεύθυνση στην σειρά που θέλουμε να επικεφτούμε τις στάσεις είναι λογικό να περνούμε ως αποτελέσματα και το "ΝΕΟ ΦΑΛΗΡΟ - ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ" και το " ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑ - ΝΕΟ ΦΑΛΗΡΟ ". Θα μπορούσα φυσικά να καθορίσουμε την σειρά που επιθυμούμε μέσω της ιδιότητας route_direction που υπάρχει στην βάση γνώσης μας απλά σε αυτή την περίπτωση το query μας θα γινόταν λίγο πιο περίπλοκο.

ΠΡΟΒΛΗΜΑΤΑ ΚΑΤΑ ΤΗΝ ΥΛΟΠΟΙΗΣΗ

Το παρόν θέμα ήταν πάρα πολύ ενδιαφέρον καθώς μας δώθηκε η ευκαιρία να χρησιμοποιήσουμε νέα εργαλεία. Η υλοποίηση ήταν ευχάριστη καθώς για την χρησιμοποίηση των εργαλείων δεν υπήρχαν σοβαρά τεχνικά προβλήματα. Επιπλέον, οι οδηγίες που δόθηκαν τόσο στην εκφώνηση όσο και στο εργαστηρικό μάθημα ήταν κατατοπιστικές.

Η κυρία δυσκολία που αντιμετώπισα ήταν να κατασκευάσω μια οντολογία που σύμφωνα με τα δεδομένα που μου δινόντουσαν να μην χάσω γνώση. Αυτό δυστυχώς δεν μπορέσα να το καταφέρω για το αρχείο stop-times και πιο συγκεκριμένα για το stop_sequence και τα arrival_time, departure_time. Το πρόβλημα εντοπιστήκε στο γεγονός ότι πρακτικά θέλαμε να συνδέσουμε όχι ένα ζεύγος αντικειμένων με μια ιδιότητα αλλά μια τριάδα αντικειμένων (stop_id, trip_id, stop_sequence). Συνεπώς, με την παρούσα οντολογία αυτή είναι η μόνη γνώση που χάνεται.