

Intelligent Tools for Policy Design



Deliverable 3.9

FUPOL CORE Platform Advanced Prototype Core Platform

Project Reference No.	287119
Deliverable No.	D 3.9
Relevant workpackage:	WP 3
Nature:	Report
Dissemination Level:	Public
Document version:	FINAL
Editor(s):	Nikolaus Rumm, Robert Thaler , Bernhard Ortner, Hakan Kagitcioglu, Peter Mairhofer, Anton Jessner, Alexander Kamenicky, Ilja Hönigschnabel
Contributors:	Peter Sonntagbauer, Herbert Löw, Susanne Sonntagbauer, Mario Neumann, Christopher Haigis
Reviewers:	Herbert Löw, Oliver Siml, Abdusalikov Bakythzan
Document description:	<p>The objective of this document is to provide a supplement to the advanced prototype release of the FUPOL Core Platform as of September 2014.</p> <p>The document covers the product on the requirements level, adds descriptions of the new features that were introduced since D3.5 and additional information on tests as long as selected architectural and design decisions.</p>

History

Version	Date	Reason	Prepared / Revised by
0.1	2014-08-22	Initial release, based on D3.5	Rumm Nikolaus
0.2	2014-09-05	Updated the features	Thaler Robert, Rumm Nikolaus
0.3	2014-09-08	Changes to the HTS integration	Thaler Robert, Ortner Bernhard, Rumm Nikolaus
0.4	2014-09-15	Update to the HTS integration – category tweaking	Ortner Bernhard, Rumm Nikolaus
0.5	2014-09-22	SEMAVIS integration - knowledge base and data browser, review results	Ortner Bernhard, Rumm Nikolaus
1.0	2014-09-26	Final release	Ortner Bernhard, Thaler Robert, Rumm Nikolaus

All rights reserved. No parts of this document may be reproduced without written permission from the FUPOL programme steering committee and/or cellent AG. This includes copying, printing, processing, duplicating and all other forms of distributing this document on any media.

Company names, product name, trademarks and brand names used in this document might be protected by law and belong to their respective owners.

We acknowledge that this document uses material from the Volere Requirements Specification Template, copyright (c) 1995-2010 the Atlantic Systems Guild Limited.

We acknowledge that this document uses material from the arc42 template by Peter Hruschka and Gernot Starke (<http://www.arc42.de>).

Table of Contents

1 INTRODUCTION AND GOALS.....	7
2 THE PURPOSE OF THE PROJECT	9
3 FEATURES	10
3.1 Product Scope of the Prototype Version	11
3.2 Business Context	13
3.3 The Hands-On Users of the Product	16
3.4 Development Team	17
4 ARCHITECTURE AND DESIGN.....	18
4.1 Architectural Overview	19
4.2 Components and Level of Completion.....	20
4.2.1 Start Screen	22
4.2.2 Client Management.....	23
4.2.3 Campaign Management.....	24
4.2.4 Opinion Maps	26
4.2.5 Text Mining, Topics and Categorization	27
4.2.6 Campaign Dashboard	33
4.2.7 Campaign Data Browser	37
4.2.8 (Social) Media Search.....	39
4.2.9 Social Media Account Pooling.....	41
4.2.10 Knowledge Base	41
4.2.11 Facts and Figures – the statistical data browser.....	42
5 DEPLOYMENT VIEW.....	44
6 TESTS AND QUALITY	45
7 USER MANUAL	46
8 HOT TOPIC SENSING API	47

8.1	Encoding and general guidelines	48
8.1.1	Literals.....	48
8.1.2	Empty values.....	48
8.1.3	Identifiers (Id).....	48
8.1.4	Support paging for resultlist on GET requests	48
8.2	Domain objects used by API methods.....	50
8.2.1	TopicEngineDocument.....	50
8.2.2	InferredDocuments	52
8.2.3	Category	54
8.2.4	Topic	55
8.2.5	TopicWord.....	57
8.3	API Methods.....	58
8.3.1	Getting categories by index	58
8.3.2	Getting document indexes	58
8.3.3	Adding Documents.....	59
8.3.4	Inferring from Existing Documents	60
8.3.5	Inferring Documents	60
8.3.6	Create Category.....	61
8.3.7	Update Category Attributes.....	62
8.3.8	Add Category Words	62
8.3.9	Remove Category Words	63
8.3.10	Add Category Documents	64
8.3.11	Remove Category Documents	64
8.3.12	Merge Category	65
8.3.13	Retrieving Categories Documents	65
8.3.14	Get Topics Proposal (with time slices).....	66
8.3.15	Get Topics Proposal on Date	67
8.3.16	Get Topics Proposal by Token	67

Management Summary

The objective of the FUPOL project is the development of a new governance model to support the policy design and implementation lifecycle. The innovations are driven by the demand of citizens and political decision makers to support policy domains in urban regions with appropriate ICT technologies. Those policy domains are very important, since more than 80% of the whole population in Europe lives in urban regions and the share will further grow in the future.

Deliverable D3.9 is the advanced prototype version of the FUPOL Core Platform as described in this document. Note that due to the nature of this deliverable almost all efforts that were necessary to produce it went into software development and that this document is only a description of what the software does and does not. So in order to benefit most from reading it we strongly suggest to use the software to get a better understanding of the features. There's a user manual available that will guide you.

The FUPOL Core Platform is a central module of the FUPOL System, providing services to the FUPOL users and to the other FUPOL modules:

- Centralized access and account management (security, user management)
- Campaign management (support for research activity) including tools like opinion maps and questionnaires
- Client management (support for multi-client operations)
- Data and knowledge management including GIS data, semantic and statistical data using semantic web technology
- Social media management including content crawling from Twitter, Facebook and other social media sites
- Operational support (services that support the reliable operations of the FUPOL System like logging, journaling)
- Integration services (messaging middleware, service coupling, ...)

An important note is that this document covers the FUPOL Core Platform, but not the complete FUPOL System. Thus all requirements mentioned in this software requirements specification, the architecture and the design focus on the core platform. Interactions with the other FUPOL modules are explained on interface level, but lack any further detail, as these have to be specified for the respective modules separately.

In order to fully understand the FUPOL Core Platform we recommend reading...

- D3.6 Revised Requirements Specification and Use Cases (which is the successor of D3.1)
- D3.2 Preliminary Software Design Description to get an understanding of the technical design that realizes the requirements from D3.1
- To a lesser extent we recommend reading D3.7 Test Reports Prototype to get an understanding of some user scenarios and workflows
- ...and finally there's our user manual that describes how to use the software product from the civic servant's point of view

This deliverable (D3.9) is based on D3.5, D3.6, D3.2 and D3.7, with parts (i.e. screenshots) taken over from the user manual, too.

There are significant dependencies between the content of D3.9 and other deliverables, like D2.6/2.19, D3.2/3.5/3.7, D4.1/4.2/4.3/4.3a/4.4, D5.1/5.2/5.3/5.4, D6.3/6.4, D7.5/7.6 and D8.5/8.9/8.12 which have been respected to design the FUPOL Core Platform based on the requirements as documented in D3.1/D3.6.

While this document's predecessor (D3.5) was based on the software release from late September 2013 this document is based on release 0.51 from the end of September 2014.

1 Introduction and Goals

This is the description of the intermediate version of the FUPOL Core Platform, as being under development by the team of work package 3 (WP3), based on the project state of late September 2014.

The FUPOL Core Platform is progressing well and it now provides features of actual real-life use for the pilot cities, and the current release is more or less feature complete. The product's quality status can be described as "near production ready". Most insufficiencies are related to non-functional requirements (better usability with some features, ...), but we had significant progress with NFRs during the last period, especially with performance.

This document shall describe what is available right now, what can be achieved when using the software and how it was done on a conceptual and technical level. The main focus is on what has been achieved and not on what will be in the future.

It's difficult to describe working software using text only, so we added some screenshots to this document, but we recommend using the product to get the whole user experience.

For an introduction to the project and the product under development we suggest to start with D3.6 Revised Software Requirements Specification and Use Cases. Deliverable D3.2 will provide details on the architecture that go beyond the scope of this deliverable, while D3.7 will enhance the reader's experience with test cases and exemplary workflows.

Finally there's the user manual which gives detailed instructions on how to use the software, following a very pragmatic and user-centric approach.

This deliverable's structure is based on its predecessor D3.5. In order to deliver a more concise document we stripped all 'generic' chapters from it (they're left as headers to preserve the chapter enumeration) and refer to D3.5 where necessary.

2 The Purpose of the Project

For an introduction to the project's purpose please refer to D3.1.

3 Features

This chapter describes the features that are implemented in release 0.51.

3.1 Product Scope of the Prototype Version

As already mentioned the prototype version is not yet the final product, but close to it and so not all of the features as listed in D3.5 (chapter 3.1) have been completed until now. Some features won't be completed at all because they could not be realised for various reasons (i.e. access to Chinese social media data) or there was no need for them any more (i.e. because we access open data online instead of importing it).

The main changes to the original scope are:

- we preferred online data access over import/store functions (statistical data, geographical data, semantic data ...)
- not all social media sources could be integrated (i.e. we were never able to access Chinese social media content for political/legal reasons), but we integrated other sources that were not on our original agenda (i.e. RSS)

For a list of all planned features please consult D3.5 (chapter 3.1). The following table lists the differences.

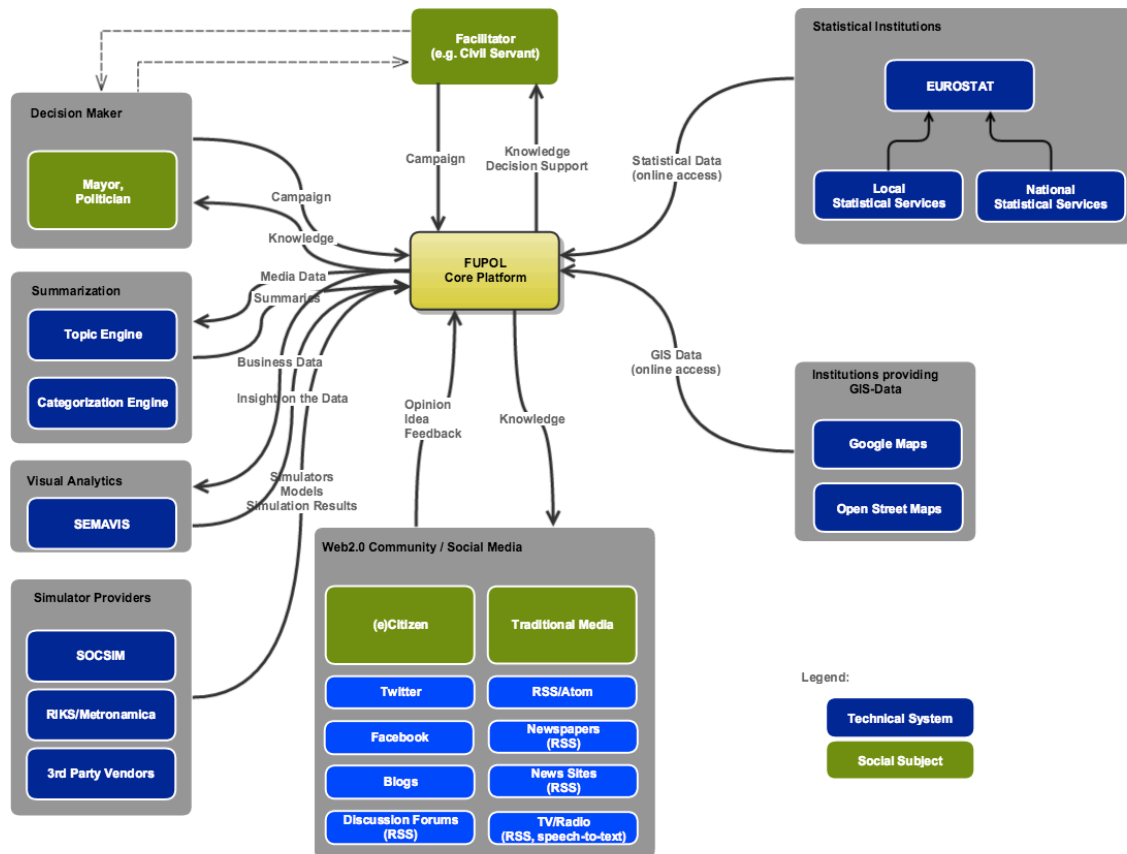
Id	Feature	Reason
WP3-338	Push campaign data back to the data base	Replaced by accessing Open Data online – no local store required
WP3-337	Pull campaign data from the data base	Replaced by accessing Open Data online – no local store required
WP3-77	Import statistical data	Replaced by accessing Open Data online – no local store required
WP3-76	Link geographical and statistical data	Replaced by accessing Open Data online – no local store required
WP3-75	Import geographical data	Replaced by accessing Open Data online (Google Maps or Open Streetmaps) – no local store

		required. Public data has advanced to a level that is sufficient for our use cases.
WP3-339	Import semantic data	Not required any more – knowledge base data is accessed online from Open Data sources
WP3-340	Browse data from the data base	Not required any more – statistical and semantic data can be browsed online using SEMAVIS
WP3-62	Publish to social media window	Not implement – regarded as spam by the social media companies and in violation of their T&Cs.
WP3-65	GIS model according to INSPIRE	Not required any more – geographical data is accessed online from Open Data sources
WP3-32	Data import from CORINE 2006 (urban atlas)	Not required – the simulators access their required data online from Open Data sources
WP3-31	LUCAS data import (land use data)	Not required – the simulators access their required data online from Open Data sources or they'll implement their own data import functions (ie.. commercial off-the-shelf simulators)
WP3-2	Support for importing and storing Eurostat data	Not required – accessed online
WP3-80	Anonymized citizen data	Not implemented because it's easy to search for the data using Google or other search engines – however we don't expose the user's data everywhere
WP3-20	Import of Linkedin data	Not implemented – not relevant for citizens
WP3-820	Import if Sina Weibo data	Not implemented for legal reasons – no access to Chinese social media data

3.2 Business Context

For a complete description of the FUPOL core platform's business context, including a representative business event list, read D3.6.

The following diagram illustrates the business context of the FUPOL core platform. Note that this is not the context diagram of the FUPOL system, but just the part of it covering the important aspects of the FUPOL Core Platform.



The Business Context shows other systems and units which are connected to the FUPOL core platform. In the following the interaction between the FUPOL-system and the surrounding systems is described.

External System	Connection to FUPOL
Decision Maker	The decision maker is one of the most important stakeholders. He is the one that is responsible for the policy and uses FUPOL to integrate the eCitizen into the policy making process.
Facilitator	The facilitator uses FUPOL for every-day-business. He accesses the FUPOL core platform with a web-client and executes campaigns initiated by the decision maker. Facilitators are power-users.
Statistical Institutions	Statistical institutions are collecting and processing statistical data (i.e. population per region). Examples are eurostat and the municipality's local agencies. They provide their data in SDMX format or – more recently – although in RDF. The system is capable of connecting to Open Data sources using RDF.
Institutions providing GIS data	<p>GIS-Data-Providers deliver spatial data (i.e. maps, thematic map layers, ...). This data will be used for visualization, simulation and for georeferencing statistical data.</p> <p>As public sources provide quite sophisticated data these days we decided to access them online and on-the-fly instead of storing the data locally.</p>
Web 2.0 Community and Media	<p>The web 2.0/3.0 community of eCitizens and citizen organizations is a group of (possibly organized) citizens that will be integrated into the policy making process either by active eParticipation (the eCitizen provides opinions upon request by the policy maker) or by passive eParticipation (the eCitizen provides opinions without an explicit trigger from the policy maker).</p> <p>Additional input for hot topic sensing is available from traditional media (newspapers, TV, radio, ...). Spoken language is supported by using a third party software (eMedia Monitor) that is capable of transcribing news broadcasts to text.</p> <p>The community of eCitizens uses various social media sites for expressing their opinions (i.e. Facebook, Twitter, Blogs, ...).</p> <p>Using Web2.0-technology and its tools e-citizens have the possibility to take part in governmental processes; for example they can participate in online-polls. If available e-citizens can also use city-websites to leave messages, as well as they can leave their opinion in blogs.</p>
Simulator Provider	A simulator provider offers a toolset for simulation purposes. This includes the simulation software and the simulation models. Some of them will be developed as part of the FUPOL project (WP2/WP4) while others will be off-the-shelf software from 3rd parties. Note that building those models and configuring them for a municipality's specific situation

	is a very complex task that is performed by the simulation modeler, a role that is usually staffed with an external consultant.
Visual Analytics	<p>Visualization tools are used by the facilitator and by the domain expert to get insights on relevant data. FUPOL will use SEMAVIS, developed by Fraunhofer IGD (WP5).</p> <p>Visualization is an important tool for understanding trends and correlations in all kinds of data (semantic data, statistical data, geo data, ...). Despite our original plan we changed the system to support online Open Data which better suits the idea of the semantic web.</p>
Summarization	<p>Summarization tools (topic analysis, categorization) use sophisticated algorithms to extract various aspects from the (social) media data that the FUPOL core platform collects from social media sites and the web 2.0/3.0 community.</p> <p>The HTS module has been reworked by WP6 and thus the API and the required procedures to interact with it have changed.</p>

WP3 acts as a middleware, connecting the modules from WP2/WP4, WP5 and WP6 to form a common service to the user.

3.3 The Hands-On Users of the Product

For a description of the hands-on users of the product please read D3.4.

3.4 Development Team

The WP3 core development team is formed of the following people. All members are working on-site in the FUPOL Project Office in Vienna at cellent. Note that as planned the team has been reduced at the start of year three.

Team Member	Role	Relevant Skills	FUPOL Participant
Bernhard Ortner	Team Member	<ul style="list-style-type: none"> Developer, Tester 	<ul style="list-style-type: none"> Cellent
Nikolaus Rumm	WP Manager	<ul style="list-style-type: none"> Architect Requirements engineer 	<ul style="list-style-type: none"> Cellent
Robert Thaler	Team Member	<ul style="list-style-type: none"> Developer 	<ul style="list-style-type: none"> Cellent
Anton Jessner (left 2013)	Team Member	<ul style="list-style-type: none"> Developer Scrum master *) 	<ul style="list-style-type: none"> Qualysoft
Ilja Hönigschnabel (left 2013)	Team Member	<ul style="list-style-type: none"> Developer GIS Expert 	<ul style="list-style-type: none"> Cellent
Hakan Kagitcioglu (left 2013)	Team Member	<ul style="list-style-type: none"> Developer GIS expert Test automation Manual testing 	<ul style="list-style-type: none"> Cellent
Alexander Kamenicky (left 2013)	Team Member	<ul style="list-style-type: none"> Test manager Tester Requirements engineer 	<ul style="list-style-type: none"> Qualysoft
Peter Mairhofer (left 2013)	Team Member	<ul style="list-style-type: none"> Designer Developer 	<ul style="list-style-type: none"> Active Solution

4 Architecture and Design

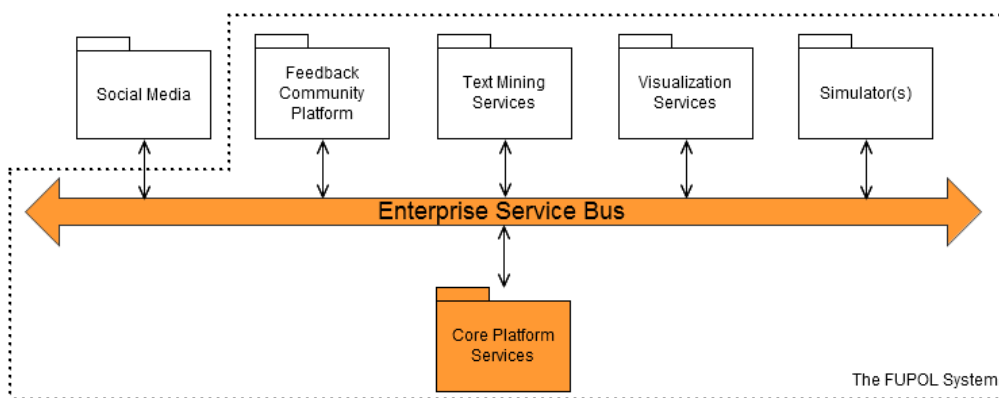
For an introduction to the architecture we recommend reading D3.5 (chapter 4).

A more detailed description can be found in D3.2.

4.1 Architectural Overview

The actual architecture is based on an enterprise service bus (ESB). Note that most connections between the modules will be (logically) point-to-point, but technically all communication will be done through the ESB.

FUPOL Architecture, Level 1



The orange modules/systems are part of the core platform

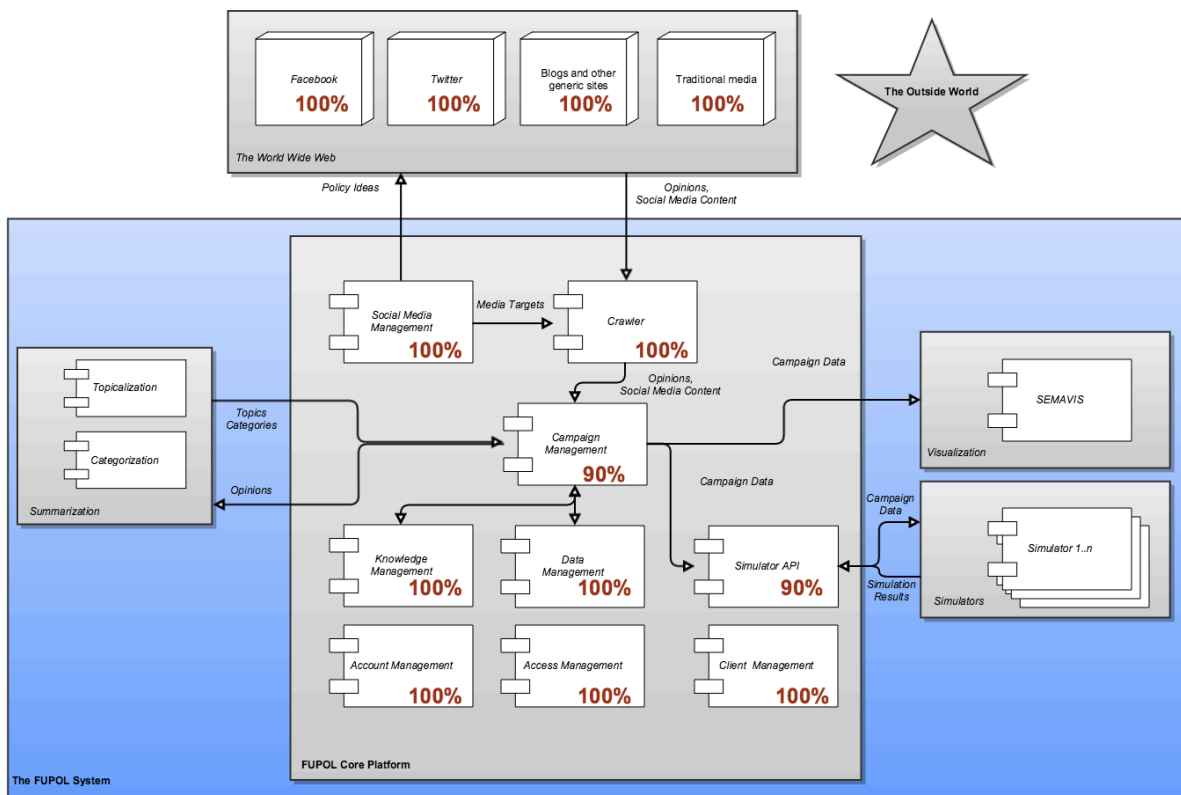
A description of the modules can be found in D3.2.

As planned (and remarked in last year's review) the Enterprise Service Bus has been re-added to the system, especially to decouple slower modules from the core platform (i.e. the text mining services).

4.2 Components and Level of Completion

The following diagram provides a bird's eye view on the logical components and the main data flows in the FUPOL system. Please note that this view doesn't represent the chosen architecture but it's here for understanding the relation/interaction between the various FUPOL modules and the data flows between them. The actual design doesn't use point-to-point-connections but instead of that it's based on a SOA architecture using an enterprise service bus.

The numbers indicate an estimation of the feature completeness of these modules (numbers on the interfaces represent the level of the current technical integration) based on release 0.51 from September 2014.



As illustrated in the diagram the core platform is now nearly feature complete and we'll focus on improving existing functionality in year four.

Social media connectivity is now complete. Content from the following sites can be crawled:

- Facebook (public posts)
- Facebook groups (wall board including comments)
- Twitter advanced search
- Blogspot.com
- RSS/Atom (including http basic authentication)

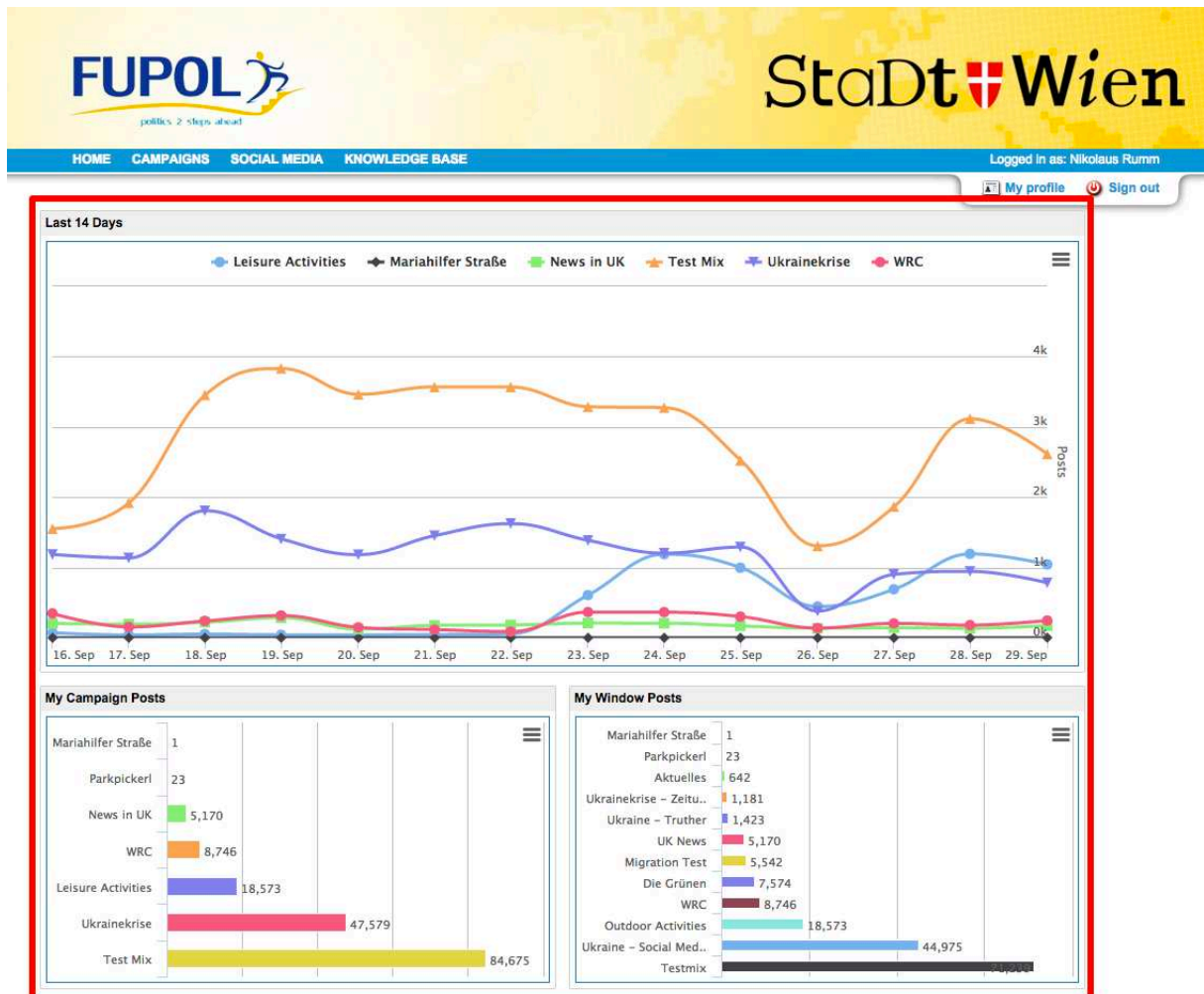
As already mentioned support for Sina Weibo has been dropped after several attempts to get access to the data were unsuccessful.

The following chapters show some screenshots of D3.9. We focused on those parts that are either new (as compared to D3.5) or that had some significant changes.

4.2.1 Start Screen

The start screen – which was empty in the previous releases – now shows some basic information about the currently ongoing campaigns.

This information supports the user in understanding the big picture, especially related to media analysis.



The diagrams show the following indicators:

- campaign buzz („last 14 days“, line chart) over time
- overall number of posts per campaign („my campaign posts“, bar chart)
- overall number of posts per social media window („my window posts“ bar chart)

The current implementation is just a teaser – more diagrams can be added as the pilot cities request them.

4.2.2 Client Management

Client management is about managing pilot cities (each pilot city is a client). This task is performed by system operators and not up to the city. Therefore we developed the FUPOL System Operator Console, a web application that is only to be used internally.

The following screenshot shows a pilot city's detail page including its geographical bounding box:

View Client

Name Zagreb
Language English
Opinion maps Use Open Streetmap

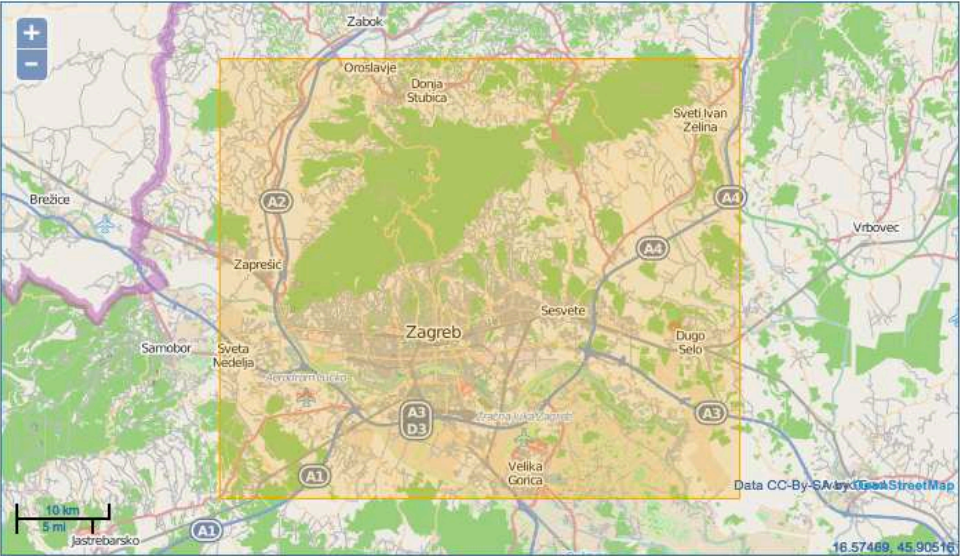
- Enable Campaign Dashboard
- Enable Categories

Only name, default language and Enable Hot Topic Sensing can be changed.

Bounding Box

NW corner

Lat 46.00200 Long 15.76500



SE corner

Lat 45.69700 Long 16.28300

Preview

The screenshot shows a map of Zagreb, Croatia, with a yellow bounding box around the city. The map includes labels for various locations such as Zabok, Oroslavje, Dornja Stubica, Sveti Ivan Zelina, Vrbovec, Brežice, Zaprešić, Sesvete, Dugo Selo, Samobor, Sveta Nedelja, Zagreb, Velika Gorica, and Jastrebarsko. Major roads are marked with A1, A2, A3, A4, and D3. A scale bar indicates 10 km and 5 miles. The map data is attributed to OpenStreetMap.

Delete client Lock all users

Edit Back

What's new is ...

- the function to enable/disable the campaign dashboard per client
- the function to enable/disable the category engine per client

The reason for these changes is that during changes in the HTS engine we turn this feature off for selected clients. Furthermore it might be required as part of our exploitation strategy (feature based pricing).

4.2.3 Campaign Management

Campaign managing is about creating, working with and closing campaigns. Campaigns can be seen as "policy making projects".

The following screenshot shows a pilot city's campaigns...

View Campaign: Ukrainekrise



Field	Value
Title	Ukrainekrise
Description	Analyse von Medieninhalten, die Ukrainekrise 2013/2014 betreffend
Start date	7/22/14
End date	9/1/15
Facilitator	Nikolaus Rumm
Initiator	Nikolaus Rumm
Team members	Peter Sonntagbauer
Goals	<ul style="list-style-type: none"> • Wir kennen die wichtigsten Themen

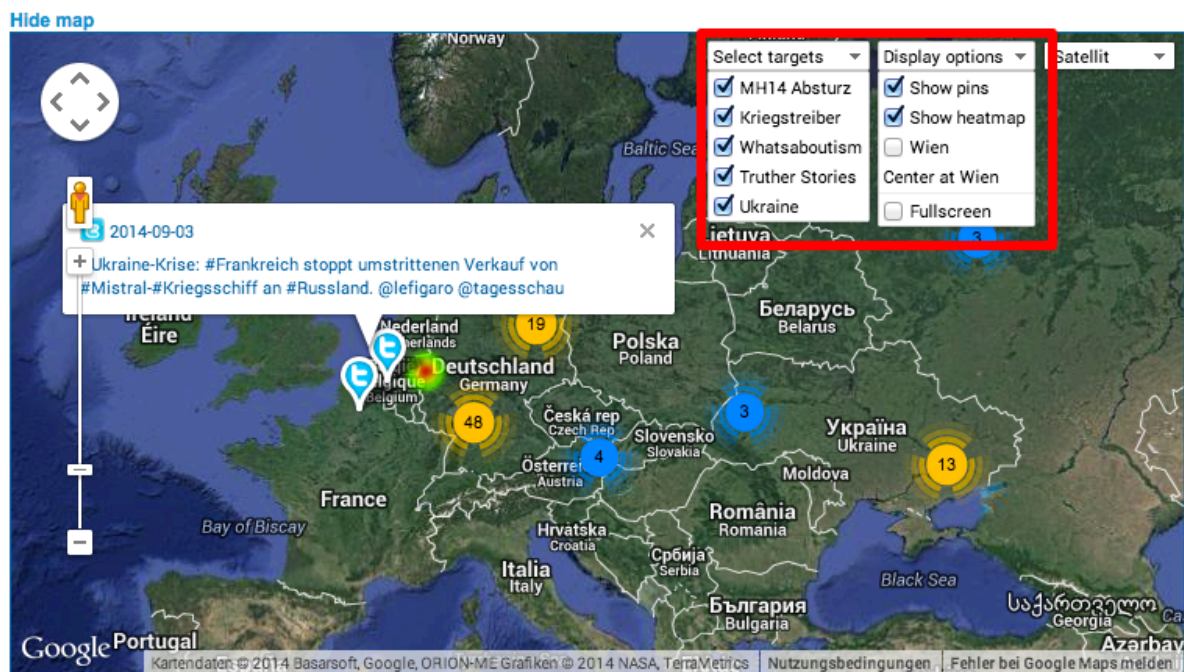
[Show map](#)
[Close campaign](#) [Delete campaign](#)
[Edit](#)

...and a campaign's detail page. Note that we've added two more sections to it:

- Dashboard – a collection of figures that illustrate various aspects of the campaign's data in a fast and convenient way
- Campaign Data – a search facility for finding posts based on various criteria

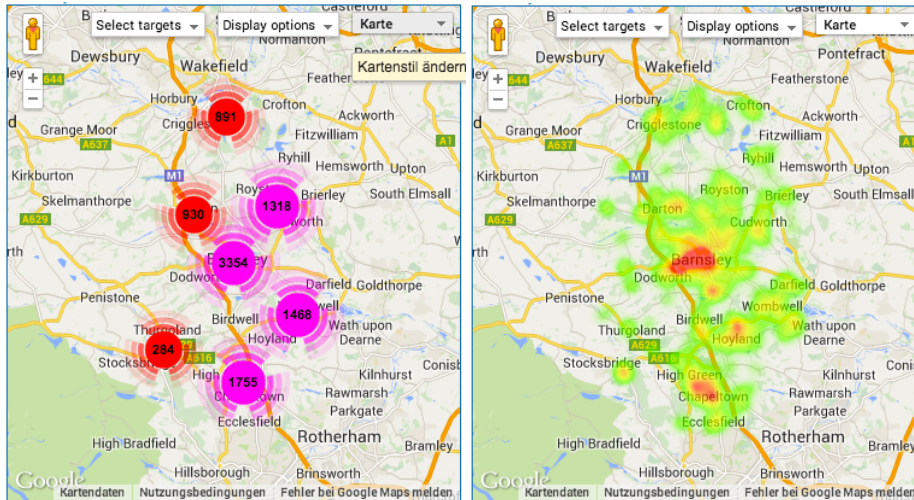
The campaign detail page's geovisualization („campaign map“) has been enhanced in the following way:

- the map allows to filter posts by their category (previously: by social media target) – so we're now able to draw i.e. heatmaps of media content according to the category learned by the hot topic sensing engine
- additional controls allow the user to zoom to their city, to open the map in fullscreen mode and to turn on/off the text on the map
- we reworked the pin detail popup to show the full text (previously it was cut after a few lines) and fixed some issues with the heatmap



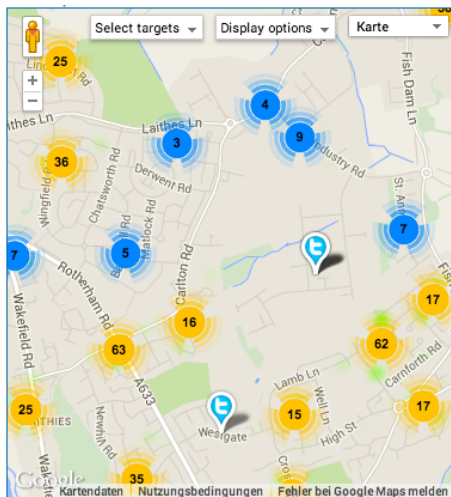
[Close campaign](#) [Delete campaign](#)

In order to get a better understanding of the geographical distribution of the posts we added a clustering mechanism (in JavaScript) that collects posts based on their proximity. This feature is a nice alternative in-between heatmaps and pins:



The screenshots above show the clusterer in action (left) and the heatmap of the same situation. While the heatmap allows for a finer resolution of the distribution, the clusterer provides absolute numbers. Clicking on a cluster sets the zoom level to focus on its data.

As can be seen in the next screenshot, the clusterer won't add isolated posts to one of the adjacent clusters – this depends on the zoom level (the clusters are calculated in the client side using JavaScript).



4.2.4 Opinion Maps

As requested by the pilot cities we've added an "anonymous mode" to the opinion maps so that they can be used without authentication (using OAuth). Some citizens

felt unsafe when they had to allow the map (or OAuth) to access their Facebook or Twitter accounts.

View Opinion Map

Settings

Name Sportski i rekreativni objekti

Description

Dragi građani, molimo dajte Vaše mišljenje u vezi proširenja i adaptacije postojećih sportskih i rekreativnih objekata te izgradnje novih u Zagrebu. Mišljenje možete dati desnim klikom miša na dio mape koja prikazuje naselje u kojem želite izgradnju ili proširenje navedenih objekata te unosom teksta u otvoreni prostor za tekst.

Creation date 2013-04-03 17:03

Locking state

A locked map prevents users from adding, editing or deleting opinions.

Authentication

Authentication required/nor required.

Visibility

Users can see all opinions, only their own opinions or no opinions at all.

Availability

Online maps are visible to all users, while offline maps are invisible

User	Read all opinions	Read my opinions	Add/Edit/delete my opinion
Anonymous	✔	✔	✔
Logged in user	✔	✔	✔

Besides that we've changed the way how the core platform stores opinions that the citizens add to the map: they're now a special kind of posts (georeferenced of course) which allows us to analyze them using FUPOL's text mining facilities (topics, categories).

4.2.5 Text Mining, Topics and Categorization

The social media tools have been heavily modified, especially with the integration of the functionalities that are developed by WP6 ("hot topic sensing").

Due to a change of the algorithm and a new approach (categories in addition to topics) the workflow between the core platform and the HTS module has changed. This required us to implement a new API for hot topic sensing (including the ESB orchestration). Nevertheless the user interface is now fully integrated, as can be seen in the following screenshots:

View Campaign: Ukrainekrise

[Summary](#) | [Tools](#) | [Social Media](#) | [Data Dictionary](#) | [Dashboard](#) | [Campaign Data](#)

Social media windows

Visualize data [Open Semavis](#)

Categories		
Name	Description	Words
Absturz MH17		777 MH17 absturzort absturzstelle airlines beweise blackboxen experten flugzeug internationalen kurs leichen luftabwehrsysteme luftraum luhansk malaysia malaysischen maschine obama opfer ostukraine osze passagiermaschine propaganda prorussischen rakete raketen rebellen regierungschef sondersitzung tschurkin ukrainische ukrainischen untersuchung video zugang
Kriegstreiber		afghanistan aggression alle amerika amerikanische blutigen damit donezk europa faschisten frage juli konzerne krieg krim land landes menschen nato obama online osten putin regierung russische schuld spiegel ukrainier verteidigung washington washingtons welt weltkrieg zukunft
MH14 Absturz		777 abschuss absturz airlines amerika berichten beschuldigungen besteht beweise blackboxen boeing deutsche flugzeug heute iran jetzt junta kamen kann konsequenzen maidan medien mh14 obama passagierflugzeugs raketen rebellen regierung russische staaten tschurkin ukrainische vereinigten washington westen zeigt zivilen zivilisten
Truther Stories		CIA amerikanische blutigen boeing damit deswegen faschisten flugzeug gemacht genau jazenjuk menschen niemand obama partei regierung russisch schuld soldaten welt
Völkermord		beschossen donbass donbassagaintnazi donezk donezker einheiten flüchtlinge kiev lugansk lugansker nazi neurussland novorossia savedonbasspeople stopukrainianarmy ukraineviolatedoeasefire ukrainiancrisis volksrepublik völkermord
Whatsaboutism		amerikanische amerikanischen amerikanischer aviv besetzung china demokratie deutschen deutschland dominikanischen februar flughafen gaza haus hilfe honduras irak iran israel israelischen krieg kuba land logistische medien menschen mexiko nachrichten nato nicaragua propaganda raketen republik september staaten trujillo vereinigten washington washingtons welt weltkrieg
Wirtschaft		achmetow aktuellen ausschuss deutsche deutschen deutschland dollar donezk entwicklung euro exporte janukowitsch kollegen krise mann millionen oligarch oligarchen parlament partei politischen prozent sanktionen ukrainier unternehmen verhandlungen vorsitzende wirtschaftliche wirtschaftssanktionen

[Create Category](#)

The screenshot above shows the user-defined categories and the words that they're based on. New categories can be created by the facilitator by pressing the "Create Category" button, which opens the following popup screen:

Edit Category

Category Title

Category Description

Category Words No words in category

Topic Words

Add 136.40	Topic 35	afrikanische	berichterstattung	brigade	explodieren	explodiert
		flugzeugmotoren	gasturbinen	gefechten	größte	hersteller
		konzern	luftbeweglichen	motor	netzfrauen	tschad
		ukrainische	verkaufte	welt	zahl	zusammengefasst
Add 133.36	Topic 10	berufssoldaten	bundeswehr	dazu	deutsche	deutschen
		deutschlands	diesen	djibouti	euch	für
		gebührt	gegen	leben	leisten	missbraucht
		russland	sanktionsspirale	schlachtfeldern	sudan	verbluten
Add 127.83	Topic 1	china	cia	dominikanische	erster	florida
		gebiet	haiti	honduras	insel	inseln
		japan	kolumbien	korea	kuba	mexiko
		nicaragua				

Sort Words by Value Probability

Add Category Word

This screen is used to define a new category based on either...

- topics ("topic words") that were learned from analyzing the campaign's content (posts) – note that all sources are considered (social media posts, opinion map posts, ...). The topics words are listed, along with the topic's accuracy.
- manual words ("add category word") can be added to a category in order to adapt it.

In most cases the category will be based on one or more topics and the manual addition of single words is just used to improve its accuracy.

The following screenshot shows a new category that's based on the "Topic 10" (the second one in the list):

Edit Category

Category Title

Category Description

Category Words

berufssoldaten x bundeswehr x dazu x deutsche x deutschen x deutschlands x
diesen x djibouti x euch x für x gebührt x gegen x leben x leisten x
missbraucht x rußland x sanktionsspirale x schlachtfeldern x sudan x verbluten x

Topic Words

Add	136.40	Topic 35	afrikanische	berichterstattung	brigade	explodieren	explodiert	flugzeugmotoren	gasturbinen	gefechten	größte	hersteller	konzern	luftbeweglichen	motor	netzfrauen	tschad	ukrainische	verkaufte	welt	zahl	zusammengefasst
Add	133.36	Topic 10	berufssoldaten	bundeswehr	dazu	deutsche	deutschen	deutschlands	diesen	djibouti	euch	für	gebührt	gegen	leben	leisten	missbraucht	rußland	sanktionsspirale	schlachtfeldern	sudan	verbluten
Add	127.83	Topic 1	china	cia	dominikanische	erster	florida	gebiet	haiti	honduras	insel	inseln	japan	kolumbien	korea	kuba	mexiko	nicaragua				

Sort Words by Value Probability

Add Category Word

Note that the words from "Topic 10" were added to the category's definition. The user can now remove words (by clicking the "x" next to each word) that she thinks don't fit the category (i.e. noise words like "für"; german for "for"). Or she can add additional words to it.

The user might want to combine topics in one category by adding them to the category's definition.

After that the user assigns a name and description to the category and saves it. The category's definition will then be handed over to the hot topic sensing engine using the HTS API and future posts will be categorized using the existing category definitions.

Another option with categories is that the user can manually overwrite the categorization. The override will be signaled to the hot topic sensing module which will use this information to adapt its category definition (the matrix).

Overwriting category definitions is done on the social media search page. Again, the user clicks the “x” next to a category in the tag cloud to unassign the category from the posting or she manually assigns a category to the posting by selecting it from the dropdown box:



In the example above clicking the “x” right next to the category “Truther Stories” will unassign the category “Truther Stories” from the first tweet, while selecting “Whatsaboutism” would manually assign the category “Whatsaboutism” to the post.

Categories are assigned to each post that’s stored in the campaign’s datastore automatically. This information can be used to draw category maps, to calculate some indicators (i.e. how the category evolves over time) or to find similar content.

Furthermore the category’s detail page was enhanced in order to show the available data – including the changes to the category’s buzz over time and the geographical distribution of its related posts. This screen can be accessed by clicking on a category’s name in a tag cloud:

Category

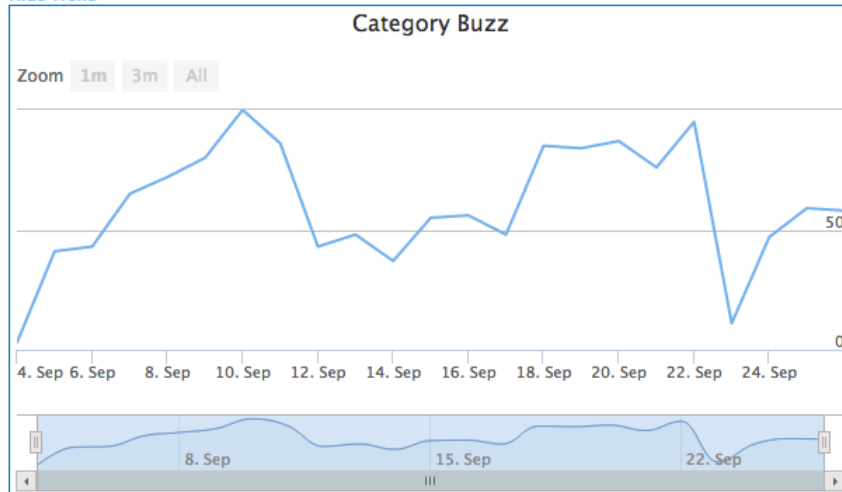
Label Truther Stories (Ukrainekrise)

Description

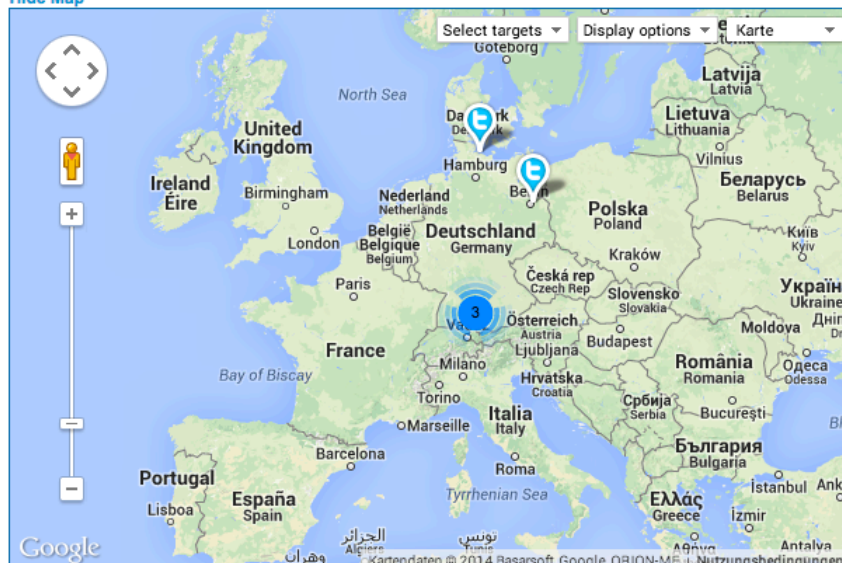
Words CIA amerikanische blutigen boeing damit deswegen faschisten flugzeug gemacht genau jazenjuk menschen niemand obama partei regierung russisch schuld soldaten welt

Content 5913 posts

[Hide Trend](#)



Hotspots [Hide Map](#)



Furthermore the campaign’s social media section provides a link to open the SEMAVIS client in a new browser tab. SEMAVIS will be launched with the campaign’s data, so the user can immediately start to analyze the campaign’s content.

We could’ve embedded SEMAVIS on the same page (i.e. in an iframe), but we felt that some users might want to use their available screen estate in a more efficient way (full screen) for analyzing content.

4.2.6 Campaign Dashboard

The campaign dashboard provides an overview about the campaign's data and some selected details. Its purpose is to give the user information about the campaign's history and some insights.

Unlike SEMAVIS, which is a visual analytics tool, the campaign dashboard does not enable the user to...

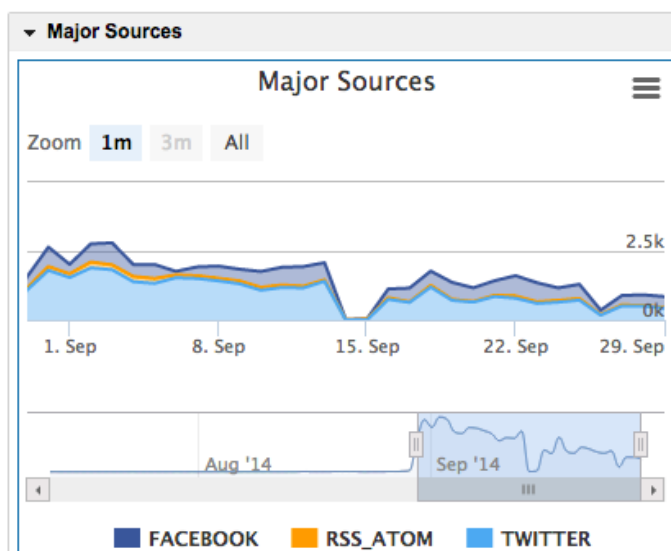
- interact with the visualization (besides setting the timeframe)
- allow to change the detail level of the visualization (drill down)
- identify single posts
- combine more data than the one that's used to render that specific diagram

However, the dashboard is very fast, as it accesses the underlying data directly instead of spooling everything to the SEMAVIS' Flash client and it provides important information for the user.

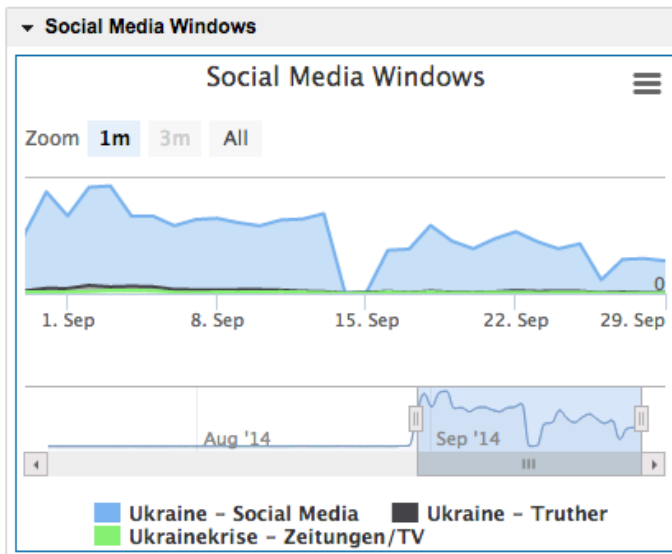
So the campaign dashboard and SEMAVIS are complementary.

The following widgets have been developed:

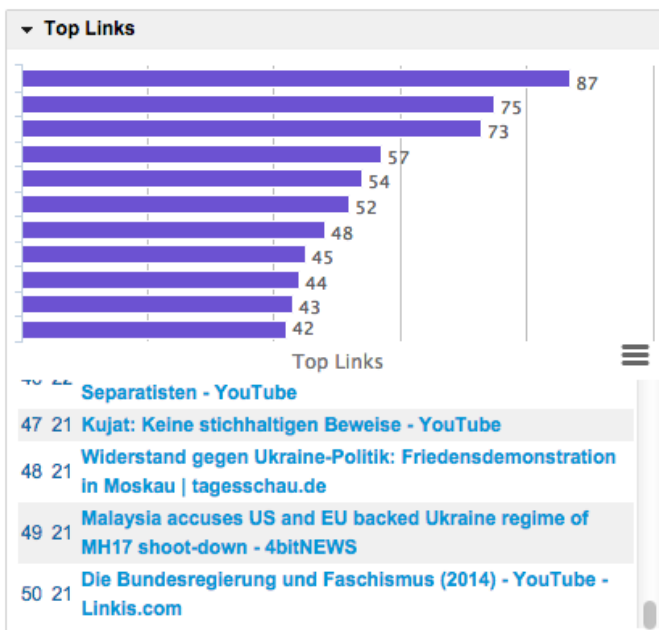
- Major Sources – shows the number of posts per media source (Facebook, Twitter, ...), independent of any social media window (it uses SIOC's site concept)



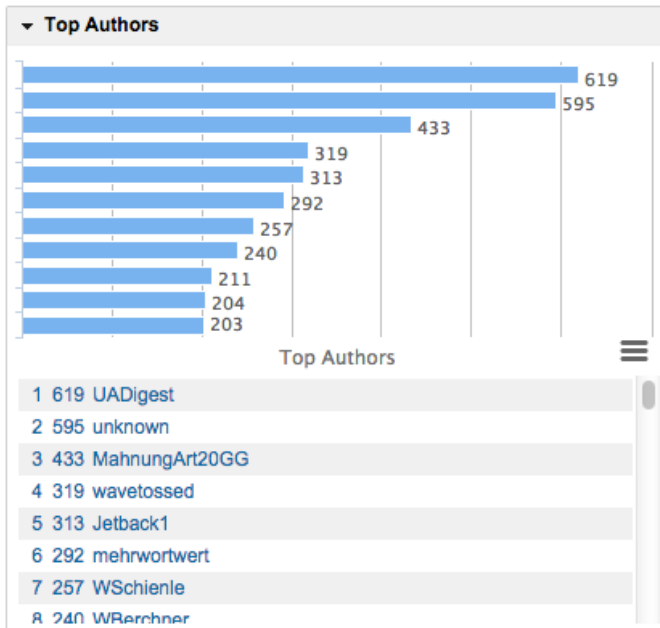
- Social Media Windows – shows the number of posts per social media window



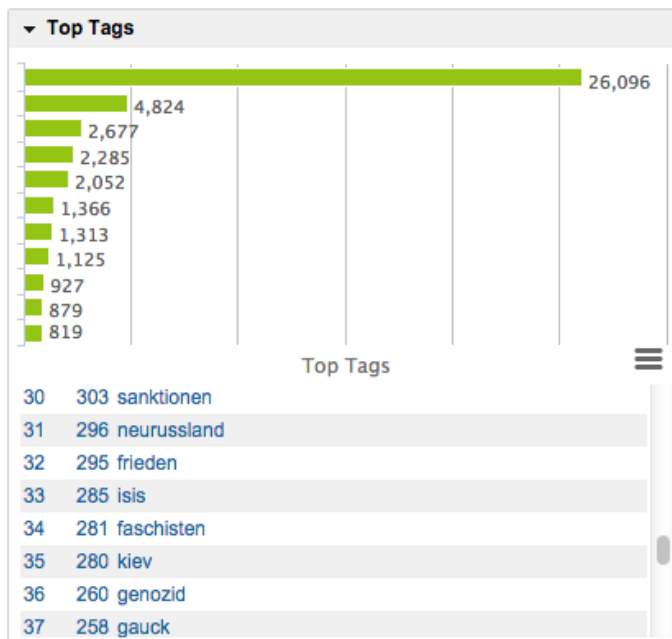
- Top Links – shows the order of the 50 most cited links that are extracted from the campaign’s posts. This information is important to identify content that citizens link to, i.e. a newspaper article, an Instagram picture, ...



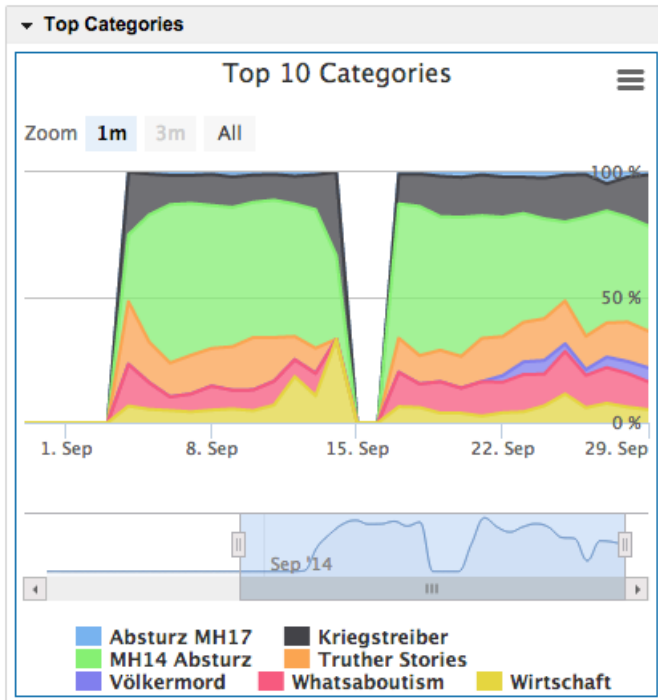
- Top Authors – shows the order of the top-contributing authors (by their number of posts) – this can be used to identify power posters. It uses SIOC’s author concept.



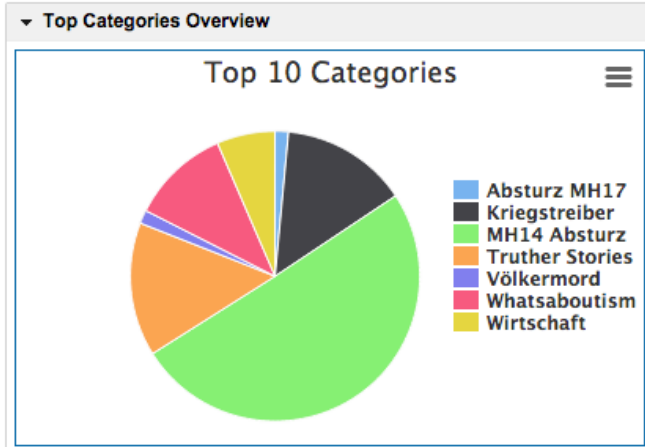
- Top Tags – shows the order of the 50 most-used hashtags that were extracted from the campaign’s data. This information can be used to understand how the citizens self-organize their content and subsequently for improving the social media targets.



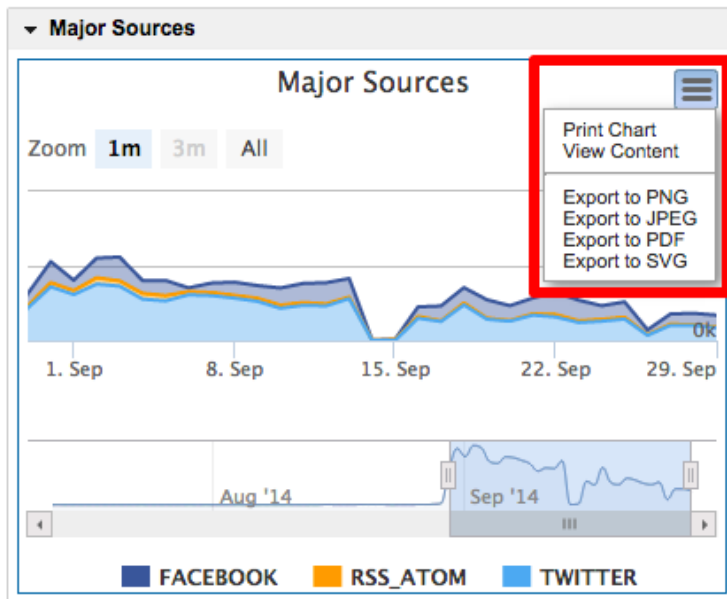
- Top categories – shows how the distribution of the defined categories evolves over time (the empty data around September 15th is caused by a system crash on that day).



- Top 10 categories – shows a pie chart of the distribution of categories over all posts in that campaign



All widgets have some additional functions that can be accessed by clicking on the menu icon on the widget's top right corner:



These functions include...

- Exporting the chart's content to PNG, JPEG, PDF and SVG
- Printing the chart
- Some widgets allow to search for the content that's shown in the chart (timeframe)

The dashboard supports drag & drop of widgets (by dragging a widget using the header line), so that the user can build his personal dashboard.

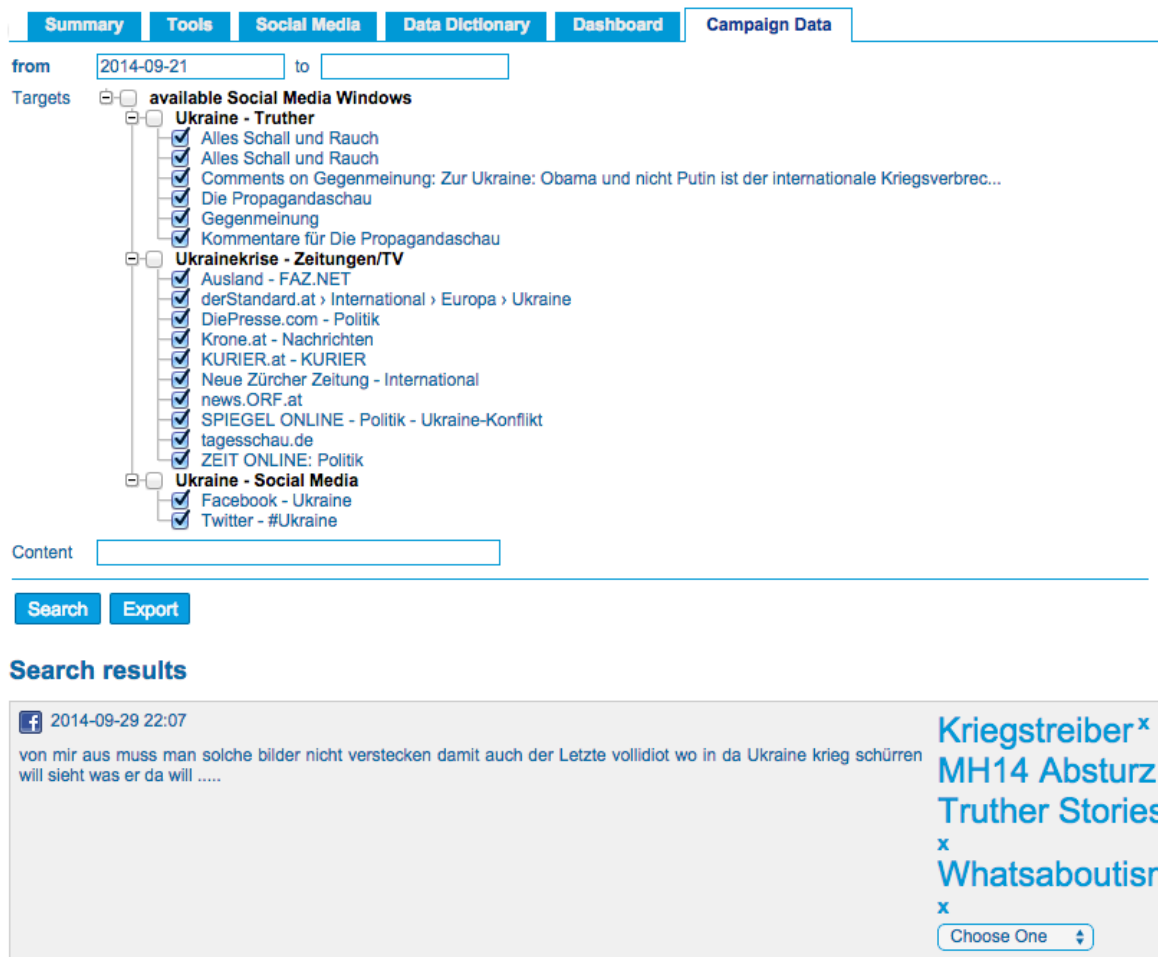
4.2.7 Campaign Data Browser

The new campaign data browser enables the user to search for content based on specific criteria. The search criteria include:

- text (keywords from the posts' content)
- timeframe (timestamp of the post within „from date“ and „to date“)
- social media target (selectable in a tree control with the social media windows on top)

- category (selectable like the social media targets) – this feature is still under development and will be available at the time of the review (the screenshot doesn't show it yet)

View Campaign: Ukrainekrise



The screenshot above shows the „Campaign Data“ screen, including the search criteria on top (timeframe, targets, content) and the result's first post, including the categories that it refers to. Note that the user can manually assign/unassign categories here.

As requested by the pilot cities we added an export function that can be used to export the search result to an Excel-file. The following screenshot illustrates an exported result as shown in Excel:

	A	B	C	D	E
	Link to original content	Created	User	Medium	Content
2	https://www.facebook.com/10000353518696/posts/311536479034800	29.09.14 22:36	Katrin Müller	Facebook	#Ukraine: Wir wissen nicht mehr, was und wem wir noch glauben sollen. Befinden wir u
3	http://derstandard.at/2000000205144/Moskau-leitet-Verfahren-gegen-Kiew-wegen-Voelkermordes-ein?ml=rss	29.09.14 22:36	redaktion@derStandard.at (derStandard.at Redaktion)	RSS/Atom	Alles über Onlinewerbung, Stellenanzeigen und Immobilieninserate Moskau leitet Verfa
4	https://www.facebook.com/100003013226672/posts/63827382751000	29.09.14 22:36	Bettina Tschirpig	Facebook	www.terschirp.de
5	https://twitter.com/318294428/status/516687611206066177	29.09.14 22:35	Youngh	Twitter	RT @tagesspiegel: #russland hat ein Strafverfahren wegen Völkermordes in der Ost-#U
6	https://www.facebook.com/1091594141/posts/10204277248266846	29.09.14 22:35	alexandersbert	Twitter	RT @Kachelmann: Immer spannende Blogbeiträge zu den wichtigen Themen zwischer
7	https://www.facebook.com/238703495/status/51668745204834624	29.09.14 22:34	Daniel M. Porcedda	Facebook	www.zeit.de
8	https://twitter.com/102911440/status/516687252353952768	29.09.14 22:34	Kndr. Wagner	Twitter	Berüchtigte Freiwilligen-Verbände in der #Ukraine: "Die Anführer und viele Mitglieder si
9	https://www.facebook.com/102911440/status/51668822353952768	29.09.14 22:34	Kachelmann	Twitter	Immer spannende Blogbeiträge zu den wichtigen Themen zwischen #mollath und #ukr
10	https://www.facebook.com/287137584802091/posts/306410192874830	29.09.14 22:31	uainform.de	Facebook	Gestern Hitler und Danzig, heute Putin und Donezk. http://www.welt.de/leitartikel/
11	https://twitter.com/37888945/status/516688365911401192	29.09.14 22:30	InnoMeX	Twitter	#Ukraine #Russland Verfahren wg Verdacht auf Völkermord "verschärft Konflikt" http://t
12	https://twitter.com/258379797/status/516688362259759104	29.09.14 22:30	igimiau	Twitter	RT @Mister_Ka: Immer mehr Beweise für von #Kiew angeordneten und zugelassene N
13	https://www.facebook.com/683717827057340/posts/679228168839639	29.09.14 22:29	Euroimedia Press auf Deutsch	Facebook	"Laut Moskau will die ukrainische Führung die russischsprachigen Bewohner in der Ost
14	https://www.facebook.com/13848620964759/posts/74233359111048	29.09.14 22:29	Berliner Osturopa-Experten	Facebook	"Laut Moskau will die ukrainische Führung die russischsprachigen Bewohner in der Ost
15	https://www.facebook.com/1380876518872311/posts/1473838768222953	29.09.14 22:28	Br-West News	Facebook	www.zeit.de
16	https://www.facebook.com/100007090541027/posts/1510816846851139	29.09.14 22:28	Ben Ramstein	Facebook	www.zeit.de
17	https://twitter.com/2781953751/status/5166881178669316	29.09.14 22:28	KID_MIS_Support	Twitter	#Ukraine-Konflikt: #Russland leitet Strafverfahren wegen Völkermordes in Ostukraine e
18	https://twitter.com/55331478/status/516688531764776981	29.09.14 22:28	keylog	Twitter	RT @PawelMVP: Metro in Kharkov in der Gewalt von Ultras. #ukraine. Eine 2-Millionen
19	https://www.facebook.com/10000078614826/posts/673226809302998	29.09.14 22:27	Annette Cornelia Eckert	Facebook	das glaube ich sofort.
20	https://twitter.com/18307269/status/51668876589530024	29.09.14 22:27	Korn	Twitter	RT @Wolfgang_H: #Russland leitet Strafverfahren wegen Völkermordes in Ost-#Ukraine
21	https://twitter.com/1381489394/status/516688539249842176	29.09.14 22:27	AgnesdeBerlino	Twitter	RT @Mister_Ka: Immer mehr Massengräber werden auf von vormalis von #Kiew's Toed
22	https://twitter.com/110774802/status/516688540549484545	29.09.14 22:26	NeryGerdYMan	Twitter	RT @Andena_: Haben die vom "Westen" finanzierten Faschisten in der #Ukraine ein V
23	https://twitter.com/97108142/status/51668851902091192	29.09.14 22:26	KnutlesewaSer	Twitter	RT @LinkInfraktion: Sewim Dagegen: Völkertrecker deutscher Politik. - http://t.co/MLSDy4
24	https://twitter.com/1693645183/status/516688494548333696	29.09.14 22:25	ThierrySordello	Twitter	RT @SZ: #Ukraine: Russland ermittelt wegen angeleglichen "Völkermordes" in der Ost#U
25	https://twitter.com/376888945/status/516688299818458624	29.09.14 22:25	InnoMeX	Twitter	RT @Gegenstrom: #USA kampf; #EU unterstützen #Ukraine - #Russland's
26	https://twitter.com/23393161/status/51668829180155523	29.09.14 22:25	skm_he	Twitter	US-Demokratie #Strafver. Deutschlands neue Außenpolitik ist schamlos http://t.co/zyNk
27	https://www.facebook.com/100001837857425/posts/743358622402083	29.09.14 22:25	Philip Kirmier	Facebook	netzfrauen.org
28	https://www.facebook.com/761714600527644/posts/688226803205586	29.09.14 22:25	RADAR	Facebook	www.zeit.de
29	https://twitter.com/2283295750/status/5166849852293776	29.09.14 22:24	Berni_Quart	Twitter	RT @Andena_: Haben die vom "Westen" finanzierten Faschisten in der #Ukraine ein V
30	https://twitter.com/1381489394/status/516684907092140032	29.09.14 22:24	AgnesdeBerlino	Twitter	RT @tagesspiegel: #russland hat ein Strafverfahren wegen Völkermordes in der Ost-#U
31	https://www.facebook.com/1000003341024815/posts/31372795481919	29.09.14 22:23	Heiko Heinenmann	Facebook	Syrien nach amerikanischem Verständnis für Demokratie - diese Bilder findet man auc
32	https://www.facebook.com/1000007897961/posts/13812693321050	29.09.14 22:23	Madlen Geißler	Facebook	www.terschirp.de
33	https://twitter.com/159146257/status/516684591587233024	29.09.14 22:23	MeditiantMart	Twitter	RT @Wolfgang_H: #Russland leitet Strafverfahren wegen Völkermordes in Ost-#Ukraine
34	https://twitter.com/2207739974/status/516684425251217152	29.09.14 22:22	Tachonka	Twitter	RT @Mister_Ka: Immer mehr Beweise für von #Kiew angeordneten und zugelassene N
35	https://www.facebook.com/100002810190389/posts/1553774881508937	29.09.14 22:22	Frank Gottschlich	Facebook	Die russische Antwort auf eine doppelte Kriegserklärung: TEIL 1 (aus Übersetzungstet
36	https://twitter.com/101691018/status/51668432904297600	29.09.14 22:21	Andrena	Twitter	RT @Mister_Ka: Immer mehr Beweise für von #Kiew angeordneten und zugelassene N

4.2.8 (Social) Media Search

Social media search has been improved as requested by the pilot cities. Unfortunately we were not able to fulfill all request, as we're limited in the API functions that the social media vendors provide.

All targets must be named – this name is shown in the various filter options (i.e. for content search – „campaign data“ tab)

- Facebook – additional (post-processing) filter options were added to the Facebook search. The user can filter the content that will be stored by adding keywords that must, should or must not be contained in the content.

Social Media Window

Choose a social media site select new target

Add social media target

Target Name

All these words


Any of these words

None of these words

- Twitter – Twitter search was improved by adding more search criteria. We’re now using almost all criteria that Twitter’s API provides, including geofencing (the user can click the „house icon“ to set the geofence to his city’s bounding box). Note that the sentiment criterium is processed by Twitter (actually in a rather simple way by counting the smilies in the content), as FUPOL doesn’t perform sentiment analysis.

Social Media Window

Choose a social media site select new target

 **Add social media target**

Target Name

All these words

Exact phrase

Any of these words

None of these words


These hashtags

Written in

From these accounts

To these accounts



Mentioning these accounts

Near this place 

latitude,longitude,radius (ie. 48.2082,16.3702,5km)

Sentiment Positive :)
 Negative :(
 Question ?

Use expert query


 **add to target list**  **cancel**

- RSS/Atom – we’ve added a post-processing filter that enables the user to filter posts by their content (keyword based). This feature was requested by the pilot cities in order to filter newspaper articles, as the RSS streams that many newspapers provide are rather generic. For example an RSS stream might contain all newspaper articles that are related to the UK, to politics or to economy, but the facilitator is only interested in articles that cover the next election. The user can now filter the content by any or none of the provided


words. Note that this filter post-processes the content (so it filters after we've crawled the content – it just prevents the system from storing it).

Social Media Window

Choose a social media site select new target

 **Add social media target**

Target Name

Url  Check



RSS example: <http://derstandard.at/?page=rss>

Filter result

Any of these words

None of these words

Use authentication

 add to target list  cancel

4.2.9 Social Media Account Pooling

In order to allow the pilot cities to collect more content than usual, and to distribute their searches over more than one requester, we've added a social media account pooling mechanism.

While this mechanism is questionable with regard to the T&C of some social media sites, it supports the city in obfuscating its requests and to some extent in circumventing the sites' rate limits.

The mechanism basically allows the administrator to set up more than one social media accounts for every site and the system will use all of them (based on a randomized round robin mechanism) when accessing the site.

Of course the legal implications of using this must be considered by the cities before using it.

4.2.10 Knowledge Base

In order to provide a sophisticated semantic knowledge base SEMAVIS was enabled to browse data from Freebase and DBpedia online:

Knowledge Browser Fact and Figures

SEMAVIS Zagreb

Freebase DBpedia

You are not logged in.

Zagreb

Semap DBpedia x Semagraph DBpedia x

agent

place

event

University of Zagreb

OKB Drama Zagreb

M. Zagreb

AT (Croatia)

Croatian Academy of Sciences and Arts

Radio-television

Thing

Zagreb

Semap Freebase x Semagraph Freebase x SemacContent Freebase x

Location Country

Organization Location

City/Town/Village

Event Statistical region

Common Dated location

Business Administrative Division

Neighborhood

City/Town/Village

Location

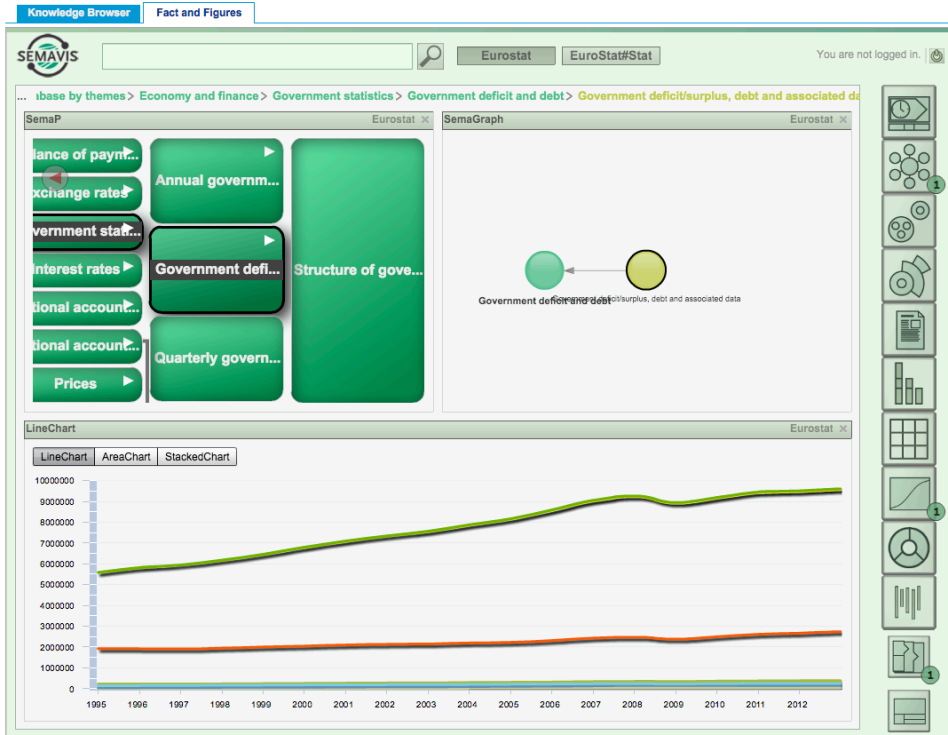
Zagreb

Zagreb

an pronunciation: [zâ greb]) is the capital and the largest city of the Republic of Croatia. It is located in the northwest of the country, along the Sava

4.2.11 Facts and Figures – the statistical data browser

Another usage of SEMAVIS within the FUPOL system is to use it as a browser for statistical data. Again, the integration point in the core platform is the knowledge base:



5 Deployment View

The current pilot system is equal to the proposed “demo system” and hosted on the FUPOL virtualized server(s). Cloud hosting is not available yet, but we experimented with Microsoft’s Azure cloud. The move from the dedicated servers into the cloud will be done in year four.

For a detailed description of the deployed system we refer to D3.2/D3.5. The overall setup has not changed (besides that the ESB is part of the implementation again).

As WP6 moved their implementation from the now ageing WP3 servers to the more powerful Xerox-servers (connected over https/JSON) we were able to split their environment into a dedicated test and another demo stage. This has allowed us to progress without affecting the pilot city operations.

All system components have been updated to newer versions over year three (i.e. the relational database).

6 Tests and Quality

For a list of test cases please refer to D3.5.

As new features were added to the system we've added additional test cases as well. Maintaining the automatic regression test suite however was quite time consuming, so some parts lack the high test coverage that we had before.

The pilot cities reported some problems with the system's stability. Almost all of them were caused by Virtuoso (crashes under heavy concurrent writing load, memory leaks in the JDBC driver, ...). We had to patch the JDBC driver which solved some cases, but in general we were forced to restart Virtuoso.

7 User Manual

In order to support the users in the pilot city (and to prevent resource drain caused by personal phone/email support) WP8 wrote a user manual.

This manual is available in electronic form (as a set of wiki pages) and as printed documentation.

8 Hot Topic Sensing API

This chapter provides a description of the API that is used to support the workflows between the core platform and the hot topic sensing API.

We decided to use a REST/JSON based http-implementation as it's network-friendly, easy to distribute and provides a good tradeoff between readability (for debugging) and efficiency.

8.1 Encoding and general guidelines

8.1.1 Literals

Timestamp: currently timestamp values are represented as Unix/Posix time values (numerical literals; number of seconds since 1970-01-01). We recommend to switch to more user-readable and exact format that includes timezone information (i.e. ISO 8601 encoding, "2014-01-01T23:00+01").

Values: Please use an uniform format. For example, if you decide to encode numeric values in long, try to stick every value to that format.

8.1.2 Empty values

Attributes that have no value assigned don't have to be included in the JSON encoded object.

8.1.3 Identifiers (Id)

Identifiers must be unique (at least within the set of objects of that domain type) and must not change at any time.

8.1.4 Support paging for resultlist on GET requests

Basically it would be good practice, if GET requests returning lists as results support paging. In fact paging support is common practice and many REST frameworks already support building pageable results out of the box.

Instead of returning a list/array as result a structured object is returned. Besides the actual data, this object contains metadata about the queryresult - query parameters allow to specify and limit the data returned. The current implementation of HTS already supports this for retrieving resources. Below a sample request and result using paging.

Request: <http://fupol-1.fupol.eu/hts/api/v1/topicengine/?limit=10>

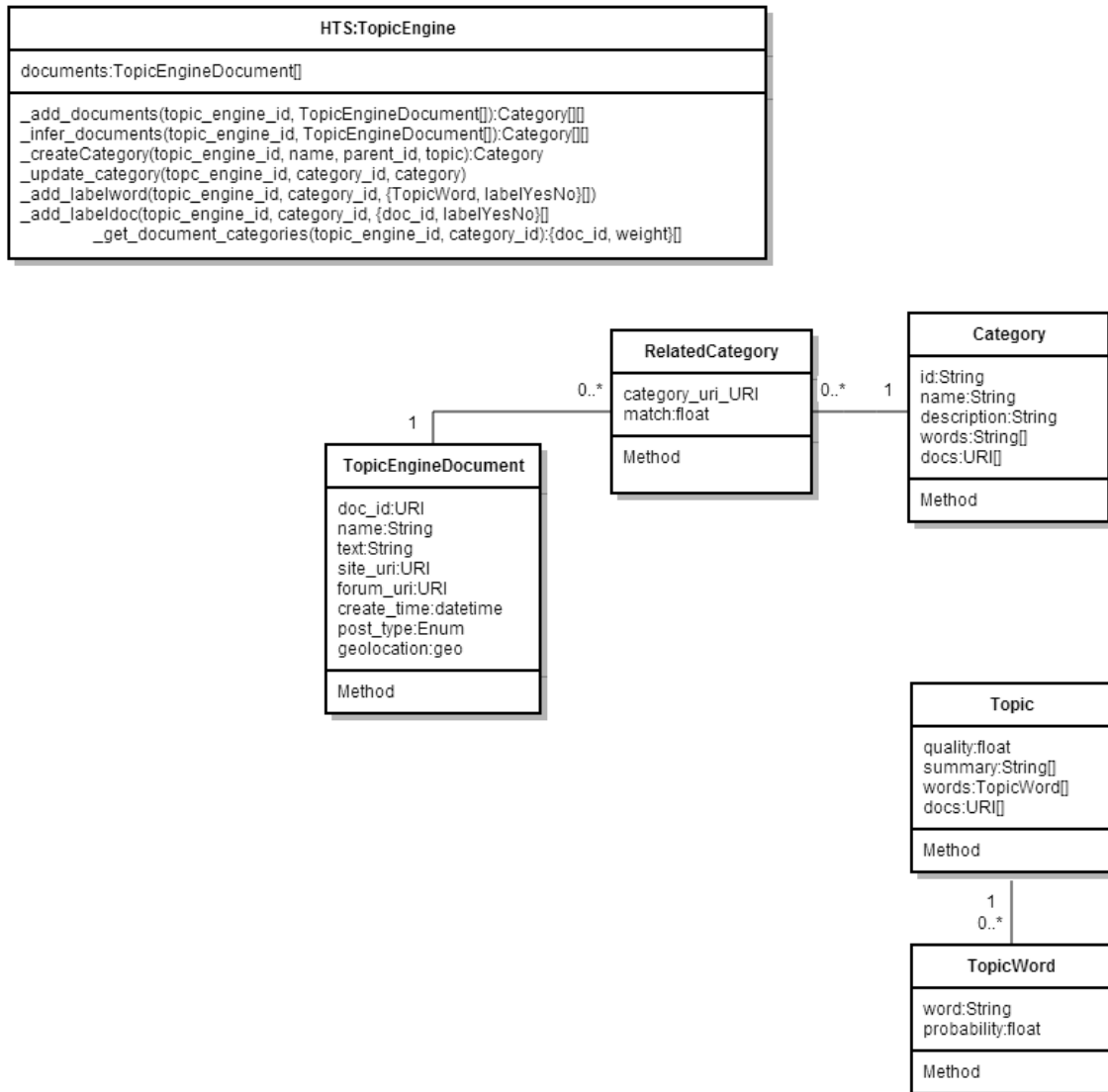
Paging Result

```
{
  "meta": {
    "limit": 10,
    "next": "/hts/api/v1/topicengine/?offset=10&limit=10",
    "offset": 0,
    "previous": null,
    "total_count": 16
  },
  "objects": [
    {
      "id": "52987acbf8e31529e08e262e",
      "name": "test language",
      ...
    },
    {
      "id": "52987acbf8e31529e08e262f/",
      "name": "London_NMFTE_20_topics",
      ...
    }
  ]
}
```

Omitting paging parameters leads to using default values, using a limit value of 0 returns the whole resultset of data.

8.2 Domain objects used by API methods

The following diagram illustrates the most important domain classes that are related to the HTS functionality:



8.2.1 TopicEngineDocument

A topic engine document is the representation of a post collected from (social) media.

Attribute	Type	Required	Description
-----------	------	----------	-------------

Attribute	Type	Required	Description
doc_id	String	yes	The document's unique identifier - usually an uri.
name	String		The name or title of the document (currently not set for any type of post)
text	String		The content of the document. A document's content should be represented in plain text without any formatting or structural information.
site_uri	String		A URI pointing to the document's origin (site, i.e. http://www.twitter.com)
forum_uri	String		A URI pointing to the document's forum (i.e. http://www.facebook.com/user/234324)
crawled_at	Timestamp	yes	Timestamp when the document has been created at the given site. In case that the create timestamp is unknown we try to guess it (usually we take the timestamp when we observed the post for the first time)
post_type	String		The type of the post that contains the content. Currently the following types are known: sioc:Post (generic post) sioc:MicroblogPost (i.e. Twitter) sioc:Comment (i.e. a Facebook comment) sioc:WeblogPost (i.e. Blogspot)
geolocation	Point		The geographical position that relates to this

Attribute	Type	Required	Description
			<p>post encoded in GeoJSON format. A specification of GeoJSON can be found here.</p> <p>Depending on the social media site only about 5% of posts will have a geolocation.</p>
index	Int		<p>The row index of the document in its corresponding TopicEngine in any future matrix returned by "Inferring Categories Documents". 1-1 relationship with doc_id</p>

Sample TopicEngineDocument

```

{
  "doc_id": "",
  "create_time": 1392108158,
  "forum_uri": "",
  "geolocation": { "type": "Point", "coordinates": [47.5678,16.7893] },
  "name": "What a good story!",
  "post_type": "sioc:Post",
  "site_uri": "",
  "text": "This story is about....",
  "index": 0
}

```

8.2.2 InferredDocuments

An inferred documents is the representation of a sparse matrix of documents X categories matching probability.

We implement it here as an array of dictionnaires, where the array is indexed by the index of the documents, and the inner dictionnaires are indexed by the index of categories.

Unfortunatly, JSON does not support integer keys for dictionnaires, so we will use the string representation of the indexes for the inner dictionnaires, it will still be much more compact than the full ids.

Only matches greater than a given threshold (for example 0.5) will be present in the inner dictionaries (sparsity), thus absent indexes of a category for a document can be interpreted as 0.

This will be the return of the 3 functions: "Inferring from Existing Documents" "Inferring Documents" and "Inferring Categories Document".

In the first case, only the categories of the asked documents will be present in the array, and the order will be the same than the order in which they where provided in input.

In the second case, the documents are not added to the TopicEngine, the indexes will also match with the order in which they where provided in input.

In the third case, all the documents are present and in the order of their predefined index.

If a document has no category that match higher than the threshold, at its index the inner dictionary will be empty.

Attribute	Type	Required	Description
Does not really have an object formalism, this is just an array of dictionaries.			

Sample InferredDocument

```
[
  {
    "2" : 0.95,
    "17" : 0.68,
    "29" : 0.7
  },
  {
    "0" : 0.65,
    "15" : 0.99
  },
  ...
]
```

8.2.3 Category

Attribute	Type	Required	Description
id	String	true	The unique identifier of this category. It is also the column index of the category in its corresponding TopicEngine in any future matrix returned by "Inferring Documents" or "Inferring from Existing Documents" or "Inferring Categories Documents"
resource_uri	String	true	The full path for accessing the category through the TopicEngine (finally come back, quite necessary to access all the category related functions and will be easier for you if we want to test different apis (beginning of the path changes))
name	String	true	Name of this category that is used to label the category. Would be set by the facilitator.
description	String		A brief description of the category. Empty by default the value can be changed by the user in the core platform.
words	Array	words doc_ids required	Array of strings
doc_ids	Array	words doc_ids required	Array of document ids

Sample Category

```
{
  "id": 7,
```

```

"uri": "/hts/api/v1/topicengine/527bb851f8/category/7",
"name": "economy",
"description": "This category aims to capture document talking about local econom
"words": [
  "unemployment",
  "jobs"
],
"doc_ids": [
  "",
  ""
]
}

```

8.2.4 Topic

A Topic does not have a stable Id. Topics can only be addressed by their index position in the list of topics of a TopicEngine.

Attribute	Type	Required	Description
id	String	true	Indexing by position is not relevant since the next time the engine runs, it might be another topic at this position. Finally, an id come back, but it will valid for a given time after the proposal is made (24h for example). During this time, a category can be created from this topic by providing its id in the "Create Category" function.
resource_uri	String	true	Corresponding uri
quality	Float		for later use: A score to rank topics in order to present 'the most relevant ones' to the user firsts ..

Attribute	Type	Required	Description
summary	Array		<p>An array of Strings containing sentences taken from the document. The summary must be calculated as part of the topic training process in HTS. We'd like to avoid an additional call to generate summaries and assume that once a topic is available the summary is available as well.</p> <p>Currently we expect to get up to 10 sentences per topic.</p>
words	Array	true	Array of TopicWords
doc_ids	Array	true	Array of documents ids of selected documents that are representative of this topic.

Sample Topic

```
{
  "id": a52b4fe541,
  "uri": "/hts/api/v1/topicengine/527bb851f8/topic/a54b2c4ea",
  "quality": 37.04,
  "summary": [
    { "sentence": "The arts exhibition in Paris was a great success." },
    { "sentence": "Paintings of dutch masters were in high demand during the auct
  ],
  "words" : [
    { "word" : "painting", "probability" : 0.48, },
    { "word" : "exhibition", "probability" : 0.42, }
  ]
  "doc_ids" : [
    "",
    ""
  ]
}
```


8.2.5 TopicWord

Simple object that holds the attributes of a topic's top word or with a category

Attribute	Type	Required	Description
word	String	yes	
probability	Float		

8.3 API Methods

The API is REST-based with some exceptions.

As a general design principle the calls should return as fast as possible. All of them are synchronous until we're back on the ESB, so the core platform will be blocked until the calls terminate on the HTS side. If a call triggers a long-running task then the long-running task must be executed after the (triggering) call has terminated.

8.3.1 Getting categories by index

Retrieve categories from their indexes

HTTP method	POST
Request path	/topicengine/<id>/_get_cats_by_index/
Request path parameters	id - string - identifier of the topicengine
Request representation	An array of categories indexes (integers)
Result	HTTP 200 - returns an array of the categories of the corresponding indexes HTTP 404 - if the topicengine does not exist

8.3.2 Getting document indexes

Retrieve document indexes from their doc_ids

HTTP method	POST	
Request path	/topicengine/<id>/_get_docs_indexes/	

Request path parameters	id - string - identifier of the topicengine	
Request representation	An array of doc_ids	see TopicEngineDocument
Result	HTTP 200 - returns an array of the indexes of the corresponding documents HTTP 404 - if the topicengine does not exist	

8.3.3 Adding Documents

Adds a collection of documents to the corpus and returns the related topics for each document. As a side effect this call might trigger a retraining of the topic engine.

It's important for the core platform that this method returns quickly, so depending on the time that retraining takes it might be a good idea to infer the topics based on the pre-trained topic engine and train the topic engine after this call has terminated.

HTTP method	POST	
Request path	/topicengine/<id>/_add_documents/	
Request path parameters	id - string - identifier of the topicengine	
Request representation	An array of TopicEngineDocuments	see TopicEngine Document
Result	HTTP 200 - returns an array of ints, the indexes of the added documents HTTP 404 - if the topicengine does not exist	

8.3.4 Inferring from Existing Documents

Returns the related categories for every requested doc_id. Corresponding document must have been added to the TopicEngine beforehand. This function have been created for two purposes: separate

adding documents and inferring their categories, thus giving the possibility to infer categories later and many times; and to separate with the next function that infer categories on not added documents.

HTTP method	POST	
Request path	/topicengine/<id>/_infer_doc_ids/	
Request path parameters	id - string - identifier of the topicengine	
Request representation	An array of doc_ids	see TopicEngineDocument
Result	HTTP 200 - An InferredDocuments object HTTP 404 - If the topicengine does not exist	see InferredDocument

8.3.5 Inferring Documents

Returns the related categories for every requested document. Unlike addDocuments this function doesn't add the documents to the corpus, but instead it just uses the knowledge that is contained inside the topic engine for labeling the documents.

For the sake of uniformity this API call basically has the same method signature as *_add_documents*.

HTTP method	POST	
Request path	/topicengine/<id>/_infer_documents/	
Request path parameters	id - string - identifier of the	

	topicengine	
Request representation	An array of TopicEngineDocuments	see TopicEngine Document
Result	HTTP 200 - An InferredDocuments object HTTP 404 - If the topicengine does not exist	see InferredDocument

WP3 main use cases related to the HTS engine are interactive search: the user searches for content on Facebook, Twitter etc. and we show him a list of search results, but those posts are not added to the corpus. Each post shall be labelled based on the existing topic engine. Is this possible with the proposed matrix approach, as it contains the links between the categories and the existing documents.

8.3.6 Create Category

Creates a new category with a given parent category

HTTP method	POST
Request path	/topicengine/<id>/_create_category/
Request path parameters	id - string - identifier of the topicengine
Request representation	<p>topic_id - String - Topic from which the Category is built from (see <u>Topic</u>).</p> <p>name - string - required - The name of the Category</p> <p>words - Array - Optional - The selected words by the annotator (from the topic proposal)</p> <p>doc_ids - Array - Optional - The selected documents by the annotator (from the topic proposal)</p> <pre>{ "name": "my first category", "topic_id": "a45bc2"</pre>
Result	HTTP 200 : The category object is returned with its id

	<p>and uri fields filled</p> <p>HTTP 404 - Topicengine doesn't exist, or Topic doesn't exist or is no more valid.</p>
--	---

8.3.7 Update Category Attributes

Allows the caller to modify the attributes of a topic.

HTTP method	POST
Request path	/category/<category_id>/_update_category/
Request path parameters	id - string - identifier of the topicengine category_id - string - identifier of the category
Request representation	<p>name - string - the new name for the category description - string - the new description for the category</p> <pre>{ "name": "the categories new name", "description": "the categories new description" }</pre>
Result	<p>HTTP 200 - Successfully updated the category HTTP 404 - If the topicengine or the category does not exist</p>

8.3.8 Add Category Words

Add several words to a specific category. The request might contain words that have already been added to the category. The server has to silently ignore those word - no error is expected by the client.

HTTP method	POST
Request path	/category/<category_id>/_add_words/

Request path parameters	id - string - identifier of the topicengine category_id - string - identifier of the category
Request representation	an array of strings (words) ["arts", "politics", "sports"]
Result	HTTP 200 - Success HTTP 404 - Either the topicengine or the category does not exist

8.3.9 Remove Category Words

Removes several words from a specific category. If the request contains words that are not associated with the category, the server has to silently ignore those words - no error is expected by the client.

HTTP method	POST
Request path	/category/<category_id>/_remove_words/
Request path parameters	id - string - identifier of the topicengine category_id - string - identifier of the category
Request representation	An array of strings (words) ["arts", "politics", "sports"]
Result	HTTP 200 - Success HTTP 404 - Either the topicengine or the category does not exist

8.3.10 Add Category Documents

Adds several document ids to a specific category. The request might contain ids for documents that already have been associated to the category. The server has to silently ignore those document ids without raising an error.

HTTP method	POST
Request path	/category/<category_id>/_add_docs/
Request path parameters	id - string - identifier of the topicengine category_id - string - identifier of the category
Request representation	An array of strings describing the document ids [" "]
Result	HTTP 200 - Success HTTP 404 - Either the topicengine or the category does not exist

8.3.11 Remove Category Documents

Removes several document ids from a specific category. If the request contains ids for documents that have not been associated to the category, the server has to silently ignore those document ids without raising an error.

HTTP method	POST
Request path	/category/<category_id>/_add_docs/
Request path parameters	id - string - identifier of the topicengine category_id - string - identifier of the category
Request representation	An array of strings describing the document ids

	[" "]
Result	HTTP 200 - Success HTTP 404 - Either the topicengine or the category does not exist

8.3.12 Merge Category

Merge Categories and create a root category. Envisaged later

HTTP Method	Post
Request Path	/topicengine/<id>/_merge_category/
Parameters	An Array of CategoryID=String
Result	A new category ID

8.3.13 Retrieving Categories Documents

Return the matrix Doc x Category with predictions scores for all documents in the campaign

HTTP method	GET
Request path	/topicengine/<id>/_document_categories/?limit=100&offset=0
Request path parameters	id - string - identifier of the topicengine
Request query	optional paging parameters - see paging get result (Paging)

parameters	
Result	HTTP 200 - An <code>InferredDocumentS</code> object (see new definition) HTTP 404 - If the specified topicengine does not exist

I decided to split the Get Topic Proposal function in two (actually three, but we'll get there soon) in order to match your needs and computation time reality as much as possible.

Here is the thing: we will constantly update a few topic models for each topic engine, which will be computed on the last hour, day, week and month for example.

Through the first function, Get Topics Proposal, you can retrieve topic proposals on one of these pre-specified time slices. The goal is to match with your granularity needs, like getting very specific topics on very recent documents, for example if a big car accident just happened on the beltway.

The bigger the time slice is, the broader the generated topics will be. On the last month, you're more likely to get topic proposals that encompasses all traffic aspects. This is a best effort service, it means that it will not provide topics exactly on the last hour for example, but rather on the last result on the hour time slice, which might have been computed 10 minutes ago for example (because the next one is not finished yet), and thus provide topic proposals from T-10min to T-1h10.

But if it happens you want to get topic proposals on a specific time slice, you can use the Get Topics Proposal on Date function, where you can specify this time slice, you will receive a token for the corresponding future topic proposals that will be generated, and come back later to ask if it is ready (the third function).

Here is the new spec:

8.3.14 Get Topics Proposal (with time slices)

HTTP method	GET	
-------------	-----	--

Request path	/topicengine/<id>/_get_topic_proposals/
Request path parameters	id - string - identifier of the topicengine
Request query parameter	slice - string - Can only be "hour", "day", "week" or "month" (to be defined)
Result	HTTP 200 - An array of <u>TopicS</u> HTTP 404 - If the specified topicengine does not exist

8.3.15 Get Topics Proposal on Date

HTTP method	GET
Request path	/topicengine/<id>/_get_topic_proposals_ondate/
Request path parameters	id - string - identifier of the topicengine
Request query parameter	start_time - Timestamp - Required end_time - Timestamp -Optional
Result	HTTP 200 - A string, the identifier of the future proposal HTTP 404 - If the specified topicengine does not exist

8.3.16 Get Topics Proposal by Token

HTTP method	GET
Request path	/topicengine/<id>/_get_topic_proposals_ondate/
Request path parameters	id - string - identifier of the topicengine
Request query parameter	proposal_token - string - Required - Identifier of a previously asked proposal, (have been returned by the Get Topics Porposal

	on Date function)
Result	HTTP 200 - An array of <u>TopicS</u> HTTP 202 - The topic proposal is not ready yet. HTTP 404 - If the specified topicengine does not exist or the provided proposal_token is invalid.