POLITECNICO DI TORINO
Repository ISTITUZIONALE

3cixty: Building comprehensive knowledge bases for city exploration

Original
3cixty: Building comprehensive knowledge bases for city exploration / Troncy, Raphaël; Rizzo, Giuseppe; Jameson, Anthony; Corcho, Oscar; Plu, Julien; Palumbo, Enrico; Ballesteros Hermida, Juan Carlos; Spirescu, Adrian; Kuhn, Kai Dominik; Barbu, Catalin; Rossi, Matteo; Celino, Irene; Agarwal, Rachit; Scanu, Christian; Valla, Massimo; Haaker, Timber. - In: JOURNAL OF WEB SEMANTICS. - ISSN 1570-8268. - 46-47:(2017), pp. 2-13. [10.1016/j.websem.2017.07.002]

Availability:
This version is available at: 11583/2679351 since: 2017-09-07T15:47:49Z

Publisher:
Elsevier

Published
DOI:10.1016/j.websem.2017.07.002

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
Elsevier postprint/Author's Accepted Manuscript

© 2017. This manuscript version is made available under the CC-BY-NC-ND 4.0 license http://creativecommons.org/licenses/by-nc-nd/4.0/.The final authenticated version is available online at: http://dx.doi.org/10.1016/j.websem.2017.07.002

(Article begins on next page)

12 October 2024

3cixty: Building Comprehensive Knowledge Bases For City Exploration

Raphaël Troncy [a] Giuseppe Rizzo [b] Anthony Jameson [d] Oscar Corcho [c] Julien Plu [a] Enrico Palumbo [b,a] Juan Carlos Ballesteros Hermida [e] Adrian Spirescu [d] Kai-Dominik Kuhn [d] Catalin Barbu [f] Matteo Rossi [g] Irene Celino [h] Rachit Agarwal [i] Christian Scanu [j] Massimo Valla [j] Timber Haaker [k]

[a] *EURECOM, Sophia Antipolis, France*
[b] *ISMB, Turin, Italy.*
[c] *Universidad Politécnica de Madrid, Madrid, Spain*
[d] *DFKI, Saarbrücken, Germany*
[e] *Localidata, Madrid, Spain*
[f] *University of Duisburg-Essen, Duisburg, Germany*
[g] *Politecnico di Milano, Milan, Italy*
[h] *CEFRIEL, Milan, Italy*
[i] *Inria, Paris, France*
[j] *Telecom Italia, Milan, Italy*
[k] *InnoValor, Amsterdam, The Netherlands*

Abstract

Planning a visit to Expo Milano 2015 or simply touring in Milan are activities that require a certain amount of *a priori* knowledge of the city. In this paper, we present the process of building such comprehensive knowledge bases that contain descriptions of events and activities, places and sights, transportation facilities as well as social activities, collected from numerous static, near- and real-time local and global data providers, including hyper local sources such as the Expo Milano 2015 official services and several social media platforms. Entities in the 3cixty KB are deduplicated, interlinked and enriched using semantic technologies. The 3cixty KB is empowering the ExplorMI 360 multi-device application, which has been officially endorsed by the E015 Technical Management Board and has gained the patronage of Expo Milano in 2015, thus has offered a unique testing scenario for the 20 million visitors along the 6 months of the exhibit. In 2016-2017, new knowledge bases have been created for the cities of London, Madeira and Singapore, as well as for the entire French Cote d'Azur area. The 3cixty KB is accessible at `https://kb.3cixty.com/sparql` while ExplorMI 360 at `https://www.3cixty.com` and in the Google Play Store and Apple App Store.

Key words: 3cixty, smart city, parallel faceted browser interface, exploratory search engine, knowledge base, instance matching

1. Introduction

Can we present a city in such a way that visitors can digitally plan a visit in advance, and explore what is happening once being in the city with

* Corresponding author. Email: raphael.troncy@eurecom.fr; Tel: +33 (0)4 - 9300 8242

the support of a digital guide? In the context of Expo Milano 2015, we have answered this question by collecting a vast amount of cultural and tourist data about the city of Milan from the E015 digital ecosystem [1] and social media platforms (such as Foursquare, Yelp, Facebook and Google). We have then strengthened the technologies and built other knowledge bases for the cities of London, Madeira and Singapore as well as the entire French Cote d'Azur area (gathering 263 cities) that also materialize regular snapshots of real-time data feeds. The resulting city knowledge base, the so-called 3cixty KB, acts as a curated data marketplace which is then used to feed the ExplorMI 360 multi-device application composed of a web application and mobile companion guides available for Android and iOS devices. The 3cixty KB contains descriptions of cultural events, sights, places and their reviews and transportation means offered in a city. As optimization, we have decomposed the geographical extent into cells and materialized travel time distances among the different geographic points. As example of real-time data, the knowledge base contains information about availability of hotel rooms at a particular instant, as well as live information about weather forecast in parts of the city.

In the case of Milan, this has offered new opportunities for Expo Milano 2015 visitors to learn what was happening during the 6 months exhibit, how to get there, and more generally, how to take advantage of the city including visiting its sights or discovering some hidden places in a completely personalized experience through the ExplorMI multi-device application. The web application enables city visitors to explore the diverse entities of the 3cixty KB in several directions in parallel, that can be saved in a personalized wishlist. The mobile application, instead, offers a companion digital guide that is mostly used by visitors once they are in the city, and want to be reminded about their favorite places and events added in their wishlist. In addition, the 3cixty KB is also intended to be used by developers to build their own apps and to extend the city digital data ecosystem. The modular design of our workflow makes the 3cixty KB extensible to any other city in the world.

Data is the new oil we repeatedly hear and are convinced of. But the typical city digital ecosystem is scattered across different data sources and with different accessing mechanisms [8]. The 3cixty KB aims to collect and harmonize descriptions of entities in one place, and to provide additional services such as reconciliation and deduplication for offering comprehensive and accurate descriptions, data analytics, data visualization and mobility optimization such as providing the time it takes to reach any place or event using public transportation from any point in the city. The 3cixty KB can empower a variety of applications and business models, as it is evident from the commercial interest from numerous companies.

The proposal of building a knowledge base for a city by combining open and private data, including data verification, reconciliation and validation, is not new. The KM4City (Knowledge Model for City) project has conducted an expert study over smart city datasets that has resulted in a new comprehensive ontology that aims to become a standard for smart cities [1]. The CitySDK project [2] offers a suite of uniform APIs to turn cities' data ready to be consumed by web developers. The project focuses on three dimensions: citizen participation, mobility and tourism. The STAR-CITY project focuses on traffic analytics for the city of Dublin, integrating numerous sensor data using semantic web technologies for accurately predicting road traffic conditions [6]. We finally observe that the publication of structured information about culture and entertainment events is often left to private companies which are largely incomplete and inaccurate in terms of coverage [4]. In contrast, the distinguishing features of the 3cixty KB are: data-centric view of the platform, built on top of well-known ontologies, and exposed as a curated data marketplace to developers via semantic web technologies. We collected evidences that the sum of information collected from all sources is vastly superior in terms of coverage that any single source: for example, we collected more local businesses in Nice that what the chamber of commerce was aware of.

We conclude that aggregating data to provide new services is hard, due to the heterogeneity of the data and the variety of the access methods, in particular for services that could have an impact for improving the citizens' quality of life. In this paper, we summarize the technical challenges we address for building the 3cixty KB and how it is used to offer services to end-users. The remainder of this paper is organized as follows: Section 2 describes the entire process for building knowledge bases for cities including the data model and the data collectors.

[1] `http://www.e015.expo2015.org`

[2] `http://www.citysdk.eu`

Section 3 presents a novel approach for reconciling the data. Section 4 outlines the various mechanisms (applications and API) for accessing the knowledge base while we detail the lessons learned and the evolution of the main application in Section 5. Finally, we conclude and propose some future work in Section 6.

2. Knowledge Base Generation

In this section, we describe the data model used in a 3cixty KB as well as the workflow for defining a geographic coverage, selecting the appropriate data sources and collecting the data which is being enriched using semantic web technologies.

2.1. *Data Modeling*

Our ontology design principle has focused on optimizing the coverage of the terminology in the context of city exploration. For each entity to model, we looked for existing ontologies by keyword searching in LOV and Smart City catalogues. We selected terms based on popularity measured through linked data usage and favoring schema.org when suitable. Two domain experts analyzed the ontologies that resulted from the search, and, once consensus was reached, ontologies were taken and added to the 3cixty data model, which, therefore, consists of a network of existing ontologies.

In details, we re-used some concepts and properties from the following ontologies: dul [3], schema [4], dc [5], lode [6], locn [7], geo [8], transit [9], ma-ont [10] and topo [11] (Figure 1). We defined additional classes and properties [12], in particular to decompose any spatial extent into squared cells that belong to administrative units, and for time-travel distances since we materialized the distance, travel time using public transportation, nearest metro or bike station for any places described in the knowledge base. This materialization is an optimization for complementing GeoSPARQL queries and route planner mechanism.

[3] http://ontologydesignpatterns.org/ont/dul/DUL.owl#
[4] http://schema.org
[5] http://purl.org/dc/elements/1.1/
[6] http://linkedevents.org/ontology/
[7] http://www.w3.org/ns/locn#
[8] http://www.w3.org/2003/01/geo/wgs84_pos#
[9] http://vocab.org/transit/terms/
[10] http://www.w3.org/ns/ma-ont#
[11] http://data.ign.fr/def/topo#
[12] http://3cixty.eurecom.fr/ontology

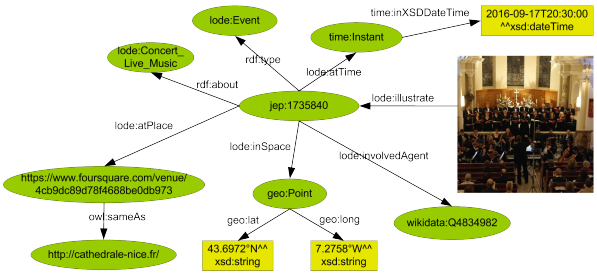

Figure 1. The event *Liberté !* scheduled during the European Heritage Days in Nice described in the 3cixty KB.

2.2. *Geographic Grid Data Sampling*

The data fetching process uses, depending on the data source, both a real-time or a batch procedure. The starting point is a grid covering a geographical area. In the case of the city of Milan, we re-used the Milan grid proposed in the 2014 BigData Challenge [13] and composed of 10,000 squared cells of 235x235 meters with a focal point in the center of the city. In the case of the territory of the French Riviera, we considered the administrative boundaries of 263 cities that touch each other, from Monaco to Toulon covering the entire Alpes Maritimes and a large part of the Var departments, yielding a total of 126,588 squared cells of 235x235 meters (Figure 2). The resulting grid is itself represented in RDF, each cell being identified by a unique URI and described by a geometry and a centroid point, while the touches relationships between cells is represented using the GeoSPARQL vocabulary.

Figure 2. The grid covering the French Riviera territory is decomposed into 126,588 squared cells of 235x235 meters.

All objects in the 3cixty KB are either described by WGS84 points (e.g. places) or are related to other objects described as points (e.g. events that happen in places). In addition, every object belongs to a

[13] https://dandelion.eu/datamine/open-big-data

cell of the grid which enables us to optimize both the reconciliation process and the query evaluation time when computing travel distance queries such as listing all restaurants that are within 10 minutes walk from a particular hotel.

2.3. *Data Source Selection*

A systematic procedure has been followed for selecting the data sources to be included in the 3cixty KB, with the goal of maximizing the spatial and temporal coverage of data, its adequacy in terms of semantics with the main tourism objects, its real-time nature and, more importantly, its usefulness for city visitors. An agile process has been established for continuously monitoring and updating the list of potentially relevant sources thanks to an appointed board member of the E015 digital ecosystem who was constantly alerting about the data sources that became available. Three types of data sources are generally used for building a 3cixty KB: global web sources (e.g. Foursquare, Yelp, Google, Facebook, Eventful, Evensi, Eventbrite, OpenAgenda), hyper local sources (e.g. E015 data services in Milan, Tourism Office in Nice or Cannes) and specialized sources (e.g. EXPO feeds, Cannes Palais des Festivals feeds, or data curated by an editorial team).

In the case of Milan, the E015 digital ecosystem aims at sharing data relevant for Expo Milano 2015 visitors through services offered mostly by local providers. We combined such a wealth of information with content publicly available on social media platforms, the rationale being to increase the coverage of the points of interest and to complement the description of existing events with user-generated activities. In the case of Nice, the Tourism Office maintains a number of high quality feeds describing all local businesses and daily cultural events. We combined this information with data coming from web sources that have a worldwide coverage in order to densify the knowledge base. Finally, for any city knowledge base, we recognize the need for curating some important data such as the top sights offered in a city, or, in the case of Milan, the up-to-date list of events that were organized every day in the various pavilions of the exhibit. Table 1 provides a breakdown of the contribution of each source in the knowledge base of Milan while Table 2 indicates the number of places, events, reviews and media in the knowledge base of Milan as well as the broad French Riviera area.

	Milan	French Cote d'Azur
Event	35.587	13.281
Place	219.893	316.156
Media	190.472	237.757
Review	70.409	110.982

Table 2
Number of distinct entities in the Milan and French Cote d'Azur knowledge bases.

Dedicated collectors have been developed for each social media platform using their respective APIs. Google Places is the database for businesses that Google maintains, and which shows up jointly in Google Maps or when performing directly searches of businesses in the Google search form. The data in this database is generated by business owners, and the quality of the data is generally high since there is also a process of data validation done through postal mail. The Google Places data model for places is described at `https://developers.google.com/places/web-service/details`. This is the data source that provides the largest number of places in all the knowledge bases that we have built, and hence it is usually the one that generates more places in the reconciled graph. Yelp is also a database that is specially focused on businesses. However, in comparison with Google Places, Yelp focuses mostly on those businesses that are open to the general public (food, nightlife, shopping, arts and entertainment, automotive, beauty, education, event planning, health and medical, home services and local services) rather than providing information for businesses that refer to companies. The Yelp data model for places is described at `https://www.yelp.com/developers/documentation/v2/business`. Foursquare is a data source that provides a similar number of places to those provided by Yelp, and has a very similar focus. The Foursquare data model for places is described at `https://developer.foursquare.com/docs/responses/venue`. We observed that Facebook is the data source that has been less useful in the development of our KBs, since the coverage of places in a given area is relatively low. This is related to the fact that the core objective of the Facebook graph is not to provide data about all businesses. The Facebook data model for places is described at `https://developers.facebook.com/docs/places`.

In terms of events, Eventful is one global source that provides information about major events as well as information about places that host those

Entity	Local Sources (E015)	Global Sources	Editorial
Event	Expo in Città (19.51%), Teatri per Milano (0.52%), Fiera Milano (0.25%), Fondazione Arnaldo Pomodoro (0.02%), Leonardo Ambrosiana (0.01%)	Evensi (68.02), Eventful (5.64%), Lastfm (1.78%)	3cixty (4.24%)
Place	Dati Comune Milano* (2.18%), BikeMI (0.01%) Vaxita (0.29%), Expo in Città (0.24%), Fondazione Arnaldo Pomodoro (0.01%), Isnart (0.23%), Teatri per Milano (0.01%)	Yelp (32.17%), Google Places (30.69%), Facebook (19.58%), Foursquare (14.24%), Expedia (0.13%), Evensi (0.06%), Eventful (0.01%)	3cixty (0.16%)
Transport	Dati Comune Milano* (52.46%) Vaxita (7.05%), BikeMI (0.29%), Isnart (0.29%)	Foursquare (17.42%), Facebook (15.67%), Yelp (3.11%), Expedia (3.02%), Google Places (0.65%)	
Media	Expo in Città (3.4%), Vaxita (0.68%), Teatri per Milano (0.06%) Fondazione Arnaldo Pomodoro (0.01%), Leonardo Ambrosiana (0.01%)	Facebook (30.73%), Forsquare (24.09%), Evensi (13.51%), Google Places (12.77%), Yelp (11.42%), Lastfm (1.2%), Expedia (0.53%), Flickr (0.33%), Eventful (0.25%)	3cixty (1.09%)
Review		Foursquare (63.15%), Google Places (29.92%) Yelp (10.93%)	

Table 1
Breakdown figures per entity and data source type. The percentage indicates the contribution of the data source with respect to the total number of unique entities (after entity reconciliation and resolution) per type in the 3cixty KB (* is part of the public data offered by the Milan municipality).

events. The data model is described at `http://api.eventful.com/docs/events/search`. Evensi is a search engine for events that aggregates data from multiple sources but primarily Facebook events. The data model is described at `https://api.evensi.com/v1.1/docs/`. OpenAgenda is a service that enables to broadcast events and agendas. The data model is described at `https://openagenda.zendesk.com/`. We observed that the stream of events is largely incomplete when considering a single source and it must be complemented by hyper local data sources such as the E015 data services and the EXPO feeds in Milan, or the Tourism Office feed in Nice, Cannes and other cities.

For processing editorial data generally entered in spreadsheets, we made use of OpenRefine [14]. For all data sources, we use the `dc:publisher` property to attribute the descriptions of the real world objects in the 3cixty model. We generate timestamped dumps for each data source that are serialized in the Turtle syntax and updated nightly. To minimize the cost of the collection process, we only harvest the delta composed of the new instances available from each source. For handling real-time data (e.g. live positioning of the bus and metro, rooms availability in hotels), we rely on publish-subscribe mechanism implemented in the general purpose complex event processing engine T-Rex [3]. We use a set of SPARQL construct queries to materialize frequent computations such as planned time-travel distances among places and events. Each dump is then loaded in a dedicated named graph for easing data management. Our triple store is the Virtuoso open source version.

3. Data Reconciliation

As we have seen, a 3cixty KB is made out of numerous sources that necessarily overlap in terms of instances (e.g. the same hotel exists in multiple data sources). In a previous work [8], we perform pairwise reconciliation of the data sources including self-reconciliation, which is frequent within social media sources, using SILK [11]. On two uniform randomly sampled datasets of 100 places and 100 events, the algorithm had an accuracy of 90.24% for places and 99.6% for events [8]. However, we ob-

[14] `http://openrefine.org`

served worst performance on the entire dataset, in particular when reconciling places.

We developed a deduplication engine named STEM that stands for Stacking Threshold-based Entity Matching classifiers and based on our previous work [7]. STEM implements an ensemble of threshold-based entity matching classifiers that are combined with a supervised learning classifier, which is trained on a set of labelled examples (Figure 3). Let us define e_i the entities to reconcile and

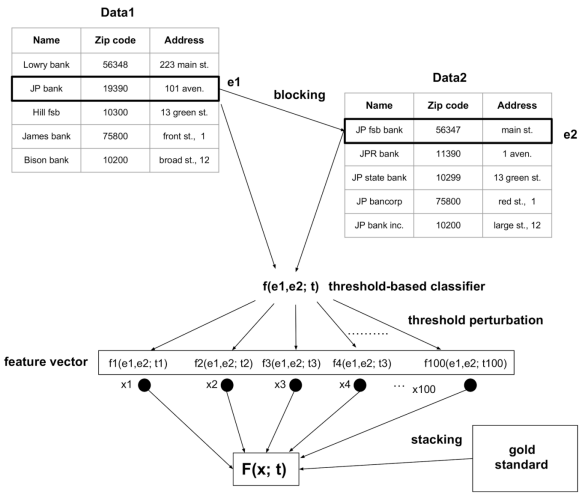

Figure 3. STEM architecture for deduplicating items in the 3cixty KB.

$f^*(e_1, e_2, t)$ a specific linkage rule obtained by the confidence function $f(e_1, e_2)$ and the threshold t. Given a set G, gold standard, of manually created `owl:sameAs` links, STEM works as follows:

(i) Start from a linkage rule $f^*(e_1, e_2, t)$ using a threshold-based classifier;
(ii) Generate an ensemble of N linkage rules $f^*(e_1, e_2, t_i)$ where t_i are linearly spaced values in the interval $[t - a_2, t + a_2]$;
(iii) Use the predictions $x_i = f^*(e_1, e_2, t_i)$ as features for a supervised learner $F(x, w)$ where w are parameters that are determined by the learning algorithm;
(iv) Train the supervised learner $F(x, w)$ on the gold standard G, determining the parameters w^*;
(v) Generate the final prediction $F(e_1, e_2, w^*)$.

The threshold-based classifier used is Duke. We used a SVM classifier on top of the different instances of the threshold-based classifiers, where the parameters w^* are generated using a statistical approach launching 100 different runs of different w^* and selecting the ones that minimize the errors (Table 3 and Table 4).

Deduplication engine	t	a	N	P [%]	R [%]	F1 [%]
Duke	0.7	-	-	50.90	86.84	64.18
STEM	0.7	0.2	10	55.18	82.46	65.94
STEM	0.7	0.4	10	71.48	68.42	69.83
STEM	0.7	0.4	100	71.08	70.61	70.77
STEM	0.7	0.5	100	71.08	70.61	70.77
Duke	0.908	-	-	76.02	65.35	70.28
STEM	0.908	0.2	5	89.68	92.11	89.68
STEM	0.908	0.2	7	79.80	70.61	81.23
STEM	0.908	0.2	10	76.24	80.70	77.95
STEM	0.908	0.2	20	78.81	79.39	78.98

Table 3
STEM performance indicators (P stands for precision, R stands for recall, and F1 as f-measure) over the gold standard of events.

Deduplication engine	t	a	N	P [%]	R [%]	F1 [%]
Duke	0.88	-	-	1	94.68	97.27
STEM	0.88	0.2	5	74.77	88.27	80.91

Table 4
STEM performance indicators (P stands for precision, R stands for recall, and F1 as f-measure) over the gold standard of events.

We can observe from the Table 4 a degradation of performance while using STEM inverting the trend that can be observed in Table 3. The analyses we have conducted showed a low quality of the gold standard concerning the set of events that affects negatively the performance of STEM. Such an outcome has triggered the revision of the gold standard for the events, which is an ongoing activity.

Both places and events are categorized according to the schema used by the data source providers [15] and the mapping established with two pivots schemes used to enable the parallel exploration of the data: the Foursquare taxonomy [16] that organizes places into a three-level hierarchy, and another taxonomy for events described in [5]. The category alignment has been supervised by two domain experts, using both the category label and description as well as the set of instances that belong to

[15] For example, Google Places use the set of categories described at `https://developers.google.com/places/supported_types`
[16] `https://developer.foursquare.com/categorytree`

those categories. The result is represented using `skos:closeMatch` and `skos:broadMatch` axioms.

For the entity resolution, we adopt a conservative approach where we add only one element when multiple sameAs links were identified by the reconciliation procedure. Such sets are thus obtained as the distinct union of the outputs of the reconciliation process with all instances that do not hold any sameAs link. They are loaded in separate graphs, namely *events* and *places* graphs of reconciled entities, which are used by ExplorMI 360 and other client applications.

4. Knowledge Base Exploration and Usage

The 3cixty KB can be visualized through the ExplorMI 360 multi-device application or directly accessed via a SPARQL interface and a dedicated API. Furthermore, a number of official city web portals start to be automatically populated from the 3cixty KB. We describe those different applications in the next sections.

4.1. *ExplorMI 360*

ExplorMI 360 is intended to offer a 360° view of the city in different stages of the visit: a web application for preparing a visit to the city and a mobile one as a digital companion guide for visitors. The web application implements a Parallel Exploration paradigm which extends Parallel Faceted Browsing [2] by enabling the display of multiple interrelated queries and their results on the screen at the same time. It also allows users to filter places or events in some novel ways such as *(a)* specifying a word or phrase that must appear somewhere in the texts associated with the item; *(b)* manually selecting items from a result list; or *(c)* specifying that a place or event must be within a certain distance of a specified location. It also allows pivoting using relationships among entities, such as discovering all hotels that are located within a certain distance of at least one park.

Parallel Exploration is a generalization of the almost ubiquitously employed interaction paradigm of faceted search/browsing, which is used in both web and non-web user interfaces for exploring the contents of large databases such as those for product catalogs (e.g., amazon.com) and hotel rooms (e.g., booking.com). In this paradigm, the user applies a succession of filters (e.g., filtering by category) to create ever more specific queries that yield smaller, usually more relevant sets of results. A limitation of this paradigm is that only one query and result set are displayed at a time. In particular, if the user wants to see the results of a previously constructed query, he has to go back and reconstruct it by respecifying one or more filters. This limitation tends to render interaction inefficient and sometimes frustrating when users are not sure in advance what filters they want to apply or when they are looking for multiple interrelated items at same time (e.g., a museum for the parents and a playground for the children). Parallel Exploration overcomes this limitation by essentially keeping track of the queries that the user has constructed in an exploration tree. This tree of queries and associated result sets is typically built up one query at a time. At any given moment, the user has access to all of the queries and results in the exploration tree.

These advanced features help to exploit fully the power of the 3cixty KB by enabling users to deal with multiple types of entities and multiple result lists at the same time. But this power is associated with a richness and complexity of the user interface that can be daunting initially to new users. For this reason, we have developed features that make it easier for new users to become accustomed gradually to the power of Parallel Exploration: *(a)* Zoom View: the unconventional inverted tree presentation of result lists on the screen is replaced with a more familiar set of result lists arranged in parallel columns; *(b)* Quick Start Apps enable users to create trees of interrelated queries simply by clicking on natural language questions within a menu. For example, if the user clicks on the question "What sorts of concerts are taking place during the coming week and on the weekend?", the same tree of results is created and shown as if the user had constructed it step by step himself. The tree is typically accompanied by several tooltips that explain to the user how he can explore the tree of results and also how he can change the details of the queries to obtain results that are more relevant to him (e.g., replacing "500 meters" by "200 meters"). Quick Start Apps are technically simple since they just associate a set of questions with a set of bookmarks that correspond to their answers. Hence it is possible even for non-technical users to create Quick Start Apps using a point-and-click interface. Finally, the web application includes a developer mode within which a developer can view the SPARQL queries generated by the user interface, so as to be able to reuse these queries (or variants of

them) in other contexts.

4.2. *ExplorMI 360 Mobile Guide: a Digital Companion for Milan Visitors*

The mobile application is available for Android and iOS. It guides users visiting and exploring the city by showing relevant and nearby information about events, points of interest, and transportation. Users can synchronize and modify their wishlist created beforehand in the web application, visualizing details about events and places and directions for reaching them. They can also receive notification alerts while walking around the city and getting closed to favorite items. The *mobility profile* feature allows users to authorize the application to track their movements and to generate a heatmap of their movements, suggesting them to discover new areas they have not yet visited. They can explore and discover new events and places, and add them into their wishlist. Visitors can also benefit from the *Thematic Tours* feature that helps plan visits at Expo Milano 2015 and optimises time. For example, by selecting a list of tags (`art and craft`, `music and show`, `family`) and the time available (e.g. night, day) users can get a list of suggested pavilions to visit or events to attend, taking into account the walking time between places and the average visit duration. The first user tests at the Expo indicate that the digital guide powered by the 3cixty KB offers a valuable and unique way to plan a visit and explore a city.

4.3. *Cote d'Azur En Fêtes: an Official Tourism Event Portal for French Riviera Visitors*

Similar to ExplorMI 360, a number of simpler web portals are now developed on top of the 3cixty knowledge bases. For example, Côte d'Azur en Fêtes[17] is the official tourism portal from the French Riviera. This web portal directly pulls up-to-date local event information from the 3cixty knowledge bases and offers a number of basic filters such as a drop down menu enabling to select one of the 263 cities, or a date picker for filtering the events that will only take place during a particular date range. By default, the events occurring today and the immediate following days are shown first. The knowledge base is automatically updated nightly at 4:00am by a cron job that collects events from global and hyper local sources and performs a live reconciliation of the data.

[17] `http://www.cotedazur-en-fetes.com/fr/`

4.4. *SPARQL and API: a Developers' Data Marketplace.*

3cixty offers new opportunities to developers to build apps that can either access a SPARQL endpoint or use a dedicated API. In a previous version, the API offered automatic integration of public and personal data for an enhanced and personalized user experience such as augmentation/re-ranking results based on items selected in the wishlist of friends or reviewed by users who are friends. Since this API provided user-personalized results, it was secured and only accessible via an API key. A new public API has been developed based on ELDA, a Java open-source implementation by Epimorphics of the Linked Data API specification. ELDA provides a configurable way to access RDF data using simple RESTful URLs that are translated into queries to a SPARQL endpoint. The API is available at `http://aplicaciones.localidata.com/apidocs/`.

5. Evaluation of the 3cixty Exploration Application

We conducted a continuous evaluation of the "multi-device" application ExplorMI 360, using field surveys and questionnaires as well as observations of how the application was used. The conception was found by many users to be too complex and unfamiliar: they wanted to be able to do everything on a single device, often preferring their mobile device for this purpose so as not to have to plan ahead. Consequently, we developed a new version of the Parallel Exploration user interface that fulfilled the following conditions:

- It should be attractive and easy to use even for unsophisticated end users who are accustomed to using faceted search and browsing.
- It should be usable even on small mobile devices such as smartphones, while still taking advantage of larger screen sizes when available.

We summarize below the advances made in terms of successively improved versions of the user interface [10].

5.1. *Showing Only One Result Set at a Time*

This version of the Parallel Exploration user interface was based on the notion that a user ought to be able to operate the user interface at first just as if it were a normal user interface for faceted browsing: the exploration tree is not explicitly constructed by the user but rather built up automatically from the sequence of queries that the user has issued. If the user wishes to view the results of a previously issued query, he can navigate back to it quickly and without losing any of the more recently constructed queries and result sets. Figure 4 shows a simple example of this version. The user has searched for well-rated restaurants in Nice and then for well-rated bars. If, after seeing the bars, he decides that restaurants are more interesting after all, he can return to the restaurants by clicking on the word "restaurant" in the fourth of the blue horizontal bars. The same principle is applicable to arbitrarily complex exploration trees.

Figure 4. Example of a use of Parallel Exploration that closely resembles normal faceted browsing.

For clarity and ease of navigation, the specifications of the queries that have been issued are represented not through a linear set of "breadcrumbs", as is typical in faceted browsing systems, but rather through a set of horizontal "breadcrumb bars", each of which represents one level of refinement of a query. This solution has also advantages in terms of allowing the use of verbose natural language formulations, which make complex queries easier to understand and which makes it possible for the entire Parallel Exploration user interface to be translated automatically via Google Translate. If these breadcrumb bars begin to take up too much space on the screen, the user can have them rolled up like Venetian blinds by clicking on an icon near the top of the screen. This version actually already fulfills the two main requirements formulated above:

- Because the interaction is very similar to that of normal faceted browsing until the user begins navigating backward to other parts of the exploration tree, virtually no initial learning is required for a new user until he or she actually wants to make use of the novel functionality that is available.
- Since at any given time only one result set is displayed on the screen, there is no reason why the user interface cannot be employed on small mobile devices.

5.2. *Showing Result Sets in Parallel*

The previous version, however, lose one of the most popular features of Parallel Exploration: the possibility of viewing two or more result sets side-by-side, which makes it easier to compare result sets than if the user had to click back and forth between them. In this version, this possibility is reintroduced as follows: whenever a node in the exploration tree has at least two result sets below it, a "Show in Parallel" icon appears on the corresponding breadcrumb bar with which the user can cause the entire subtree under that node to be displayed on the screen. Only the result sets at the bottom of the tree are displayed, but the breadcrumb bars for the higher-level nodes are also displayed, so that the content of the result sets can be understood. This solution enables the user to view any subtree of the exploration tree, including possibly the entire exploration tree.

If, after having caused a subtree to be displayed, the user wants to focus on just a subtree (possibly containing just a single result set) of that subtree, he can click on the "Zoom In" icon on the breadcrumb bar for the smaller subtree. To see a result set which is not at the bottom of the tree, the user can click on the breadcrumb bar corresponding to that result set; the more specific result sets are then temporarily pushed "below the fold" until the user has once again clicked on the same breadcrumb bar or performed

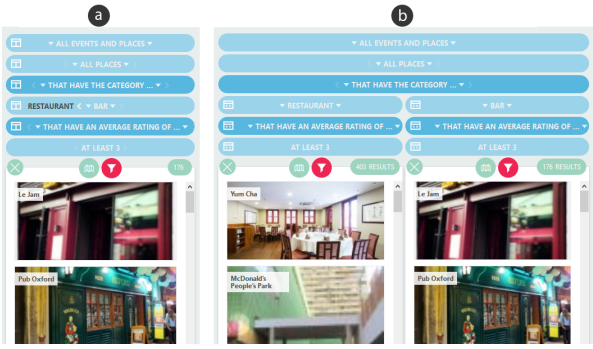

Figure 5. Example of display of result sets next to each other (b), in contrast to the previous version (a).

any other operation.

5.3. *Automatic Adaptation to Small Windows*

The previous version has the limitation that it does not work well on small screens (or, more generally, in small browser windows). If on such a screen, for example, the user requests that a subtree with 5 terminal nodes should be displayed, it is possible that the screen will not be large enough to display all of these terminal result sets even with the minimal width that is required for each result set.

This problem was resolved in a new version: whenever this problem would arise, the user interface automatically combines some of the terminal nodes in question into a single displayed box in such a way that only one result set is displayed in the box and the sibling result sets can be reached through clicking on the arrows in the breadcrumb bar above the box. It is desirable to apply this method to as few nodes as possible, so that as many as possible of the result sets of interest can still be displayed. Hence the system applies a fairly sophisticated algorithm to search through the set of all possible ways in which nodes can be combined in this way, and it chooses the way that allows the largest number of result sets to be displayed.

5.4. *Column View*

Even the last version proved not simple enough for new users to grasp quickly, largely because of an unnecessary complication that had been inherited from earlier versions of Parallel Exploration: In these earlier versions, there was a distinction between

- the display of a result set comprising any number of results of a query (though, with very large result sets, only the first 100 results are initially shown in the others are loaded on demand); and
- the display of an individual result in more detail.

This same distinction is found in typical systems for faceted browsing, where the user clicks on a displayed item in a result set to see a separate page with more details about that item. In connection with Parallel Exploration, this distinction has a drawback: the individual nodes need to be fit into the exploration tree as a separate type of node.

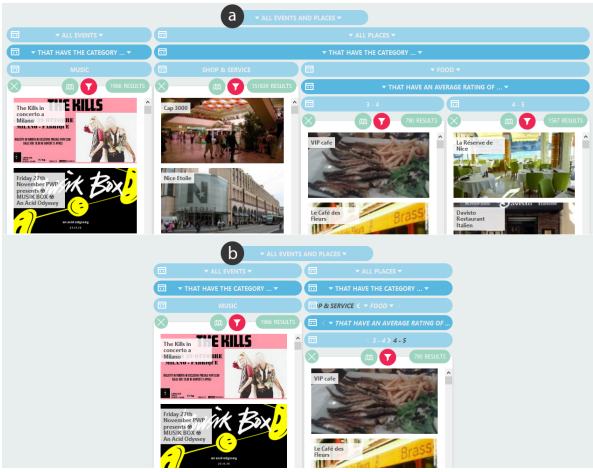

Figure 6. Example of automatic combination (b) of result sets into a single box when space is limited.

The individual nodes necessarily had different properties and a different appearance from those of the result set nodes, and these differences had to be learned by the user in addition to the other novel elements of the Parallel Exploration user interface. Accordingly, we eliminated individual nodes through the introduction of a third way of visualizing the elements of a result set in addition to the default "Thumbnail View" and the alternative "Map View"; the new view is called "Column View". In Column View, instead of displaying the items in a result set with thumbnail images accompanied by some minimal information, the user interface presents for each item in the result set an entire column of information and graphics, which can be scrolled downward if necessary (see Figure 7 and Figure 8). If the number of results in the set is too large for the columns to be shown at the same time even with their minimum width, the user can scroll to the right to see more of them. In this case, the value of Column View lies in the fact that the user can see more detail about each item than in Thumbnail View, at the cost of having to scroll more to see all of the results.

If the number of results in the set is small enough for all of the columns in Column View to be displayed at once, the width of each column is increased so that the columns together use up all of the available space in the box for the result set. In particular, if there is only one result to be displayed, this result occupies exactly the same amount of space that would previously have been occupied by an individual node. In other words, an individual node is replaced in this solution by a generously displayed result set comprising a single element.

In addition to simplifying the conceptual model of the user interface, this solution offers the advantage that the novel filter type of manual selection, which was introduced in 2015, is now more widely useful: Looking at a result set in "Column View", the user now typically has enough information about each item to be able to decide whether he wants to put it into a smaller manually selected result set comprising items that he wants to compare or examine more closely. Therefore, it is quite natural to embed the "manual selection" option into the basic method for displaying results sets so that the user can manually create a smaller result set comprising N elements with just N+1 clicks. In this way, for example, the user can progressively narrow down a set of 20 restaurants to smaller sets of 8 and then 2 restaurants that he wants to examine more closely a natural procedure for a decision maker who is "winnowing" a large set of options but one that is not at all well supported by any widely used interaction paradigm.

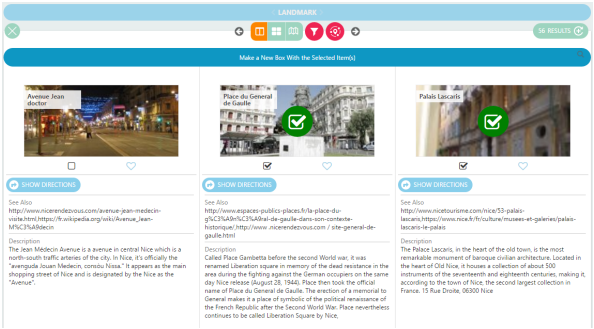

Figure 7. Display of a large result set in Column View. The user has selected two items for closer inspection (see Figure 8).

5.5. *Entry Points*

Entry Points are simple widgets on the home screen of the web application (see Figure 9) that

Figure 8. The two selected items are displayed in a larger size in a separate result list.

make it possible for a new user to take advantage of the powerful functionality of Parallel Exploration even while initially using a widget that is as simple and fast as for example, the search box that is found on the homepage of Airbnb [18]. We describe here the technical aspects of this innovation.

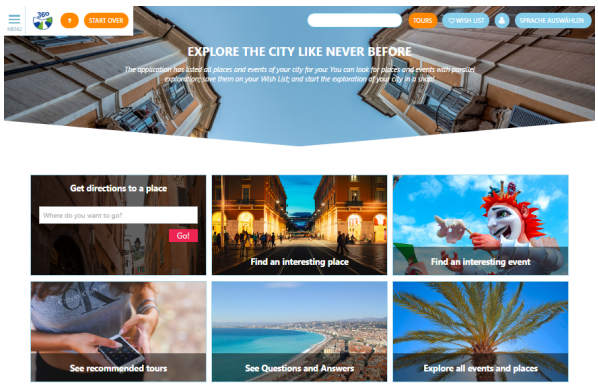

Figure 9. Entry Points on the home screen of https://nice.3cixty.com. After clicking on one of the six Entry Points, a user can specify the requested specific input (if any).

Entry Points can be seen as a generalization of the Quick Start App. Selecting a preformulated natural language question (e.g. "What hotels are next to parks?") from a Q&A list simply causes a stored bookmark to be invoked and a specific exploration tree to be displayed whereby the particular results that are shown for the exploration tree will depend on the state of the knowledge base at the time of invocation.

An Entry Point is more general in that it includes some parameters that are to be specified by the user, as in "Find nearby places in one or more of the following categories". This functionality can actually

[18] `https://www.airbnb.com`

be realized quite straightforwardly as a generalization of bookmarks, since a bookmark internally consists of a representation of the sequence of operations that are required to build up the exploration tree in question. For an Entry Point, the internal definition is a sequence of steps, one or more of which depend on the specific input provided by the user. It is possible to represent such a sequence internally in a declarative form that is relatively straightforward to understand and edit without reference to any of the JavaScript code that actually executes the Entry Point. For this reason, even specific Entry Points that are of only temporary or local interest can be introduced with little effort

6. Conclusion and Future Work

3cixty is an example of a successful implementation of semantic web technologies in a real world proof-of-concept that has been tested first for the occasion of the Expo Milano 2015, where 20 million people visited the Expo site and a portion of them [19] used the 3cixty KB via the ExplorMI 360 application. We have established a nightly mechanism of data release and KB updates. A central repository is designed to welcome discussion and record issues. In addition, team members are regularly participating to W3C discussions on the topics of smart city, event modeling, spatio-temporal representation, and open annotation, as well as on more local initiatives. This brings daily discussions on the standard approaches for the data modeling.

The 3cixty KB is the output of a specific need in exploring new opportunities for Expo Milano 2015 visitors to exploit cultural, tourist, business, transport, point of interests offered by Milan in a ubiquitous and contextualized manner. The research results, data model, and engineering solutions have then been deployed in other cities such as London and the large French Cote d'Azur area which is now empowering the official tourism portal of the French Riviera region consulted by thousands of people every month. The approach of collecting cultural and tourist data from two types of data streams, such as specific to city-related data providers and global social media platforms, allows being agile in building a knowledge base specifically tailored for any other city, improving the precision of the instances collected from social media platforms by using specific data feeds provided by cities themselves.

The innovative exploration, powered on top of the 3cixty KB, offered a 360 degree view of the Milan city in different stages of the visit: a web application for preparing a visit to Milan and a mobile one as a digital companion guide for visitors. The Parallel Exploration enables the display of multiple interrelated queries and their results on the screen at the same time, while the mobile application guides users visiting and exploring the city by showing relevant and nearby information about events, points of interest, and transportation modalities. Users can synchronize and modify their wishlist created beforehand in the web application, visualizing details about events and places and directions for reaching them.

As part of future work, we are currently investigating how to mine relevant data to generate a knowledge base automatically. Mining such data becomes of an unexploited value for research and commercial initiatives. Thanks to our handmade curated mechanism of data source selection, we offer certain degrees of quality in the generation process, further validated by data instance selection and a systematic procedure of rdf-ization. To strengthen this, we are exploring the importance of performing temporal analysis over the different developments of the knowledge base in order to automatically detect anomalies and to automatically repair the 3cixty KB. A pilot research line that we are currently exploring is focused on generating automatically the geographic fingerprints of the Milan extent [9] that soon will be offered as another type of entity in the knowledge base. Another pilot research line is focusing on using association mining rules to align categories, which is currently performed by hand. 3cixty aims to provide an immersive experience by offering personalized user interfaces that are the other research line we are conducting to advance the state-of-the-art in the parallel exploration.

7. Acknowledgments

The authors would like to thank Emiliano Verga (CEFRIEL), Cong Kinh Nguyen (INRIA), Animesh Pathak (INRIA), José Luis Redondo Garcia (EURECOM) and Ahmad Assaf (EURECOM) for their technical contributions. This work was primarily supported by the innovation activity 3cixty (14523)

[19] The number of downloads in the Apple App Store and Google Play Store is in the range of thousand. The number of unique visitors to the Web Application is in the range of ten thousand according to Google Analytics.

of EIT Digital (`https://www.eitdigital.eu`).

References

[1] P. Bellini, M. Benigni, R. Billero, P. Nesi, N. Rauch, Km4City ontology building vs data harvesting and cleaning for smart-city services, Journal of Visual Languages & Computing 25 (6) (2014) 827–839.

[2] S. Buschbeck, A. Jameson, A. Spirescu, T. Schneeberger, R. Troncy, H. Khrouf, O. Suominen, E. Hyvönen, Parallel Faceted Browsing, in: International Conference on Human Factors in Computing Systems (Interactivity Track) (CHI), 2013.

[3] G. Cugola, A. Margara, Complex Event Processing with T-REX, Journal of Systems and Software 85 (8) (2012) 1709–1728.

[4] A. Fialho, R. Troncy, L. Hardman, C. Saathoff, A. Scherp, What's on this evening ? Designing user support for event-based annotation and exploration of media, in: 1st International Workshop on EVENTS: Recognising and tracking events on the Web and in real life, Athens, Greece, 2010.

[5] H. Khrouf, V. Milicic, R. Troncy, Mining events connections on the social web: Real-time instance matching and data analysis in EventMedia, Web Semantics: Science, Services and Agents on the World Wide Web 24 (1) (2014) 3–10.

[6] F. Lécué, S. Tallevi-Diotallevi, J. Hayes, R. Tucker, V. Bicer, M. L. Sbodio, P. Tommasi, STAR-CITY: Semantic Traffic Analytics and Reasoning for CITY, in: 19th International Conference on Intelligent User Interfaces (IUI), 2014.

[7] E. Palumbo, G. Rizzo, R. Troncy, An Ensemble Approach to Financial Entity Matching for the FEIII 2016 Challenge, in: $2^n d$ International Workshop on Data Science for Macro-Modeling (DSMMM), 2016.

[8] G. Rizzo, O. Corcho, R. Troncy, J. Plu, J. C. Ballesteros Hermida, A. Assaf, The 3cixty Knowledge Base for Expo Milano 2015: Enabling Visitors to Explore the City, in: 8th International Conference on Knowledge Capture (K-CAP), 2015.

[9] G. Rizzo, R. Meo, R. G. Pensa, G. Falcone, R. Troncy, Shaping City Neighborhoods Leveraging Crowd Sensors, Information Systems 64 (2017) 368–378.

[10] A. Spirescu, From parallel faceted browsing to Parallel Exploration: Strengthening of an interaction paradigm, Master's thesis, Saarland University, Department of Computer Science (2016).

[11] J. Volz, C. Bizer, M. Gaedke, G. Kobilarov, Discovering and Maintaining Links on the Web of Data, in: 8th International Semantic Web Conference (ISWC), 2009.