# The CitySPIN Platform: A CPSS Environment for City-Wide Infrastructures

Amr Azzam
WU Vienna
1020 Vienna, Austria
aazzam@wu.ac.at

Peb R. Aryan
ISE Institute, TU Wien
1040 Vienna, Austria
peb.aryan@tuwien.ac.at

Alessio Cecconi
WU Vienna
1020 Vienna, Austria
cecconi@ai.wu.ac.at

Claudio Di Ciccio
WU Vienna
1020 Vienna, Austria
claudio.di.ciccio@ai.wu.ac.at

Fajar J. Ekaputra
ISE Institute, TU Wien
1040 Vienna, Austria
fajar.ekaputra@tuwien.ac.at

Javier Fernández
WU Vienna
1020 Vienna, Austria
jfernand@wu.ac.at

Sotiris Karampatakis
Semantic Web Company
1070 Vienna, Austria
sotiris.karampatakis@semantic-web.com

Elmar Kiesling
ISE Institute, TU Wien
1040 Vienna, Austria
elmar.kiesling@tuwien.ac.at

Angelika Musil
ISE Institute, TU Wien
1040 Vienna, Austria
angelika.musil@tuwien.ac.at

Marta Sabou
ISE Institute, TU Wien
1040 Vienna, Austria
marta.sabou@tuwien.ac.at

Pujan Shadlau
Wiener Stadtwerke Holding AG
1030 Vienna, Austria
Pujan.Shadlau@wienerstadtwerke.at

Thomas Thurner
Semantic Web Company
1070 Vienna, Austria
t.thurner@semantic-web.at



## ABSTRACT

Cyber-physical Social System (CPSS) are complex systems that span the boundaries of the cyber, physical and social spheres. They play an important role in a variety of domains ranging from industry to smart city applications. As such, these systems necessarily need to take into account, combine and make sense of heterogeneous data sources from legacy systems, from the physical layer and also the social groups that are part of/use the system. The collection, cleansing and integration of these data sources represents a major effort not only during the operation of the system, but also during its engineering and design. Indeed, while ongoing efforts are concerned primarily with the operation of such systems, limited focus has been put on supporting the engineering phase of CPSS. To address this shortcoming, within the CitySPIN project we aim to create a platform that supports stakeholders involved in the design of these systems especially in terms of support for data management. To that end, we develop methods and techniques based on Semantic Web and Linked Data technologies for the acquisition and integration of heterogeneous data from disparate structured, semi-structured and unstructured sources, including open data and social data. In this paper we present the overall system architecturewith a core focus on data acquisition and integration.We demon-strate our approach through a prototypical implementation of an adaptive planning use case for public transportation scheduling.

*1st Workshop on Cyber-Physical Social Systems (CPSS2019), October 22, 2019, Bilbao, Spain.*



## KEYWORDS

CPSS, Linked Data, Knowledge Graphs, Publict Transport, Smart City.

## 1 INTRODUCTION

Cyber-physical Systems (CPSs) are systems that span the physical and cyber-world by linking objects and process from these spaces. A typical CPS collects data from the physical world via sensors and applies computation resources from the cyber-space to integrate and analyze this data in order to decide on optimal feedback processes that can be put in place by physical actuators. CPSs have started to diffuse into many areas, including mission-critical public transportation, energy services, and industrial production and manufacturing processes.

The results of a recent study about adaptation in CPS [16] revealed an emerging trend to add an additional social layer in a CPS architecture to address human and social factors and evolve these systems into CPSSs [21]. The resulting systems consist not only of software and raw sensing and actuating hardware, but are fundamentally grounded in the behaviour of human actors, who both generate data and make informed decisions based on data [5, 12, 22].

The CitySPIN[1] project aims to lay a foundation for the development of CPSSs in the context of Smart City infrastructure services. To this end, we develop both theoretical and conceptual foundations, as well as a set of innovative components — illustrated in Figure 1 — that support a CPSS design process in a uniform platform. This platform supports key stakeholders involved in the design process through a prototyping environment that provides a visual interface which allows them to *(i)* access a wide range of data sources from sensors, social channels, and legacy systems; *(ii)* integrate and analyze heterogeneous data; and *(iii)* visualise results. This platform is made possible by methods and tools that make use of Semantic Web and Linked Data technologies to support the collection and integration of heterogeneous data sources.

In this paper, after a brief overview of the CitySPIN arechitecture in Section 2, we focus on the two core aspects of this technology stack: the knowledge graph construction, covered in Section 3, and the prototyping environment, described in Section 4. Furthermore, we discuss the prototypical implementation and illustrate the application of the platform by means of an example use case involving Vienna's largest public transport provider in Section 5. Finally, we briefly review related work in Section 6 and conclude the paper with an outlook on future research in Section 7.

## 2 CITYSPIN ARCHITECTURE OVERVIEW

The design of cyber-physical social systems raises challenges due to high complexity introduced by social systems in terms of:

(i) *the number and heterogeneity of data sources that need to be integrated*: CPSSs involve large amounts of heterogeneous, polystructured data from a variety of sources, ranging from legacy databases to highly dynamic sensor data. To create CPSS applications and services, it is paramount to efficiently integrate not just the data produced by individual processes within the organization, but to achieve integration across processes, departments, organizational boundaries, and domains. Finally, external data, such as, for instance, social media streams, are also of pivotal importance in the context of CPSSs. Hence, a major challenge is to develop flexible data integration infrastructures that are responsive to the varying needs of CPSSs.

(ii) *privacy concerns associated with the processing of sensitive social data*: Adequate privacy protection is a fundamental requirement in the context of CPSSs, which often make use of and integrate sensitive information from various sources. Additionally, the new EU General Data Protection regulation imposes new demands in terms of transparency of data processing and also in terms of allowing data subjects to revoke or change their consent in parts, which calls for more flexible and dynamic compliance checking. This represents a significant barrier towards the development and provision of integrated smart city services and hinders product and process innovation.

(iii) *uncertainty due to social dynamics*: CPSS designers need a better understanding of the social dynamics of the groups involved in the CPSS, both at the design time and the run-time of the system (e.g., for on-the-fly adaptation).

All these challenges are amply reflected in a CitySPIN use case that aims to improve the daily schedule planning for the Viennese public transport network. In particular, this use case aims to support planners in their work by allowing them to treat the transportation system as a CPSS and accounting for the dynamics of the involved travelers (especially during large-scale events). This requires, amongst others, the integration of data from various sources including data internal to the organization (e.g., historic data about event attendance), open data (e.g., expected events), as well as real-time data from mobility operators. Some of these data sources can raise privacy concerns (e.g., when harvested by apps installed on individual mobiles) and therefore user consent about the use of this data needs to be appropriately captured and considered during data processing. Finally, network planners would like to understand recurring social behaviors and patterns – for example, the typical routes followed by participants of an event.

CitySPIN tackles these challenges in the design and prototyping phases of a CPSS and aims to offer support to key stakeholders involved in these stages including decision makers, project managers, software architects, and software engineers, as depicted in Figure 1. These stakeholders are provided with a *CPSS Prototyping Environment* that adopts a mashup-based paradigm to allow them to easily acquire, explore, combine and visualise a variety of data sources (e.g., legacy data, streaming data, social media data, open data). The CPSS Prototyping Environment relies on and is made possible by three key components, as described next.

*Scalable Linked Data Integration.* We adopt Linked Data technologies to address the integration of multiple, heterogeneous data sources. To this end, we developed dedicated components for the acquisition and semantic enrichment of data as well as the integration into a *CPSS Knowledge Graph.* The next sections of this paper will focus on CitySPIN's data integration architecture primarily.

*Secure Data Access and Privacy.* To deal with privacy concerns typically associated with social data, we develop components for capturing user consent and making use of this consent during the entire data integration chain.

*Process Mining on Linked Data.* Finally, to support stakeholders in gaining a better insight into group dynamics, we develop a Process Mining & Analytics component that can be used to analyze behavioral patterns and make predictions based on the CPSS knowledge graph. Process mining is the discipline connecting data science and business process management that aims at discovering, checking, and enhancing business processes based on data logged by information systems [20]. In the context of this project, we resort in particular on declarative process mining to cater for the flexibility of the processes considered in this project [15]. Declarative process models specify dynamic systems through temporal-logic-based rules that establish the constraints with which the execution must comply. Therefore, we resort on the expression of those constraints as queries over the CPSS knowledge graph to monitor and analyse the behavior of ongoing processes [9]. The query answers are thus routed to the CPSS Mashup Platform to allow for further complex analytics and refinements.

[1] http://cityspin.net

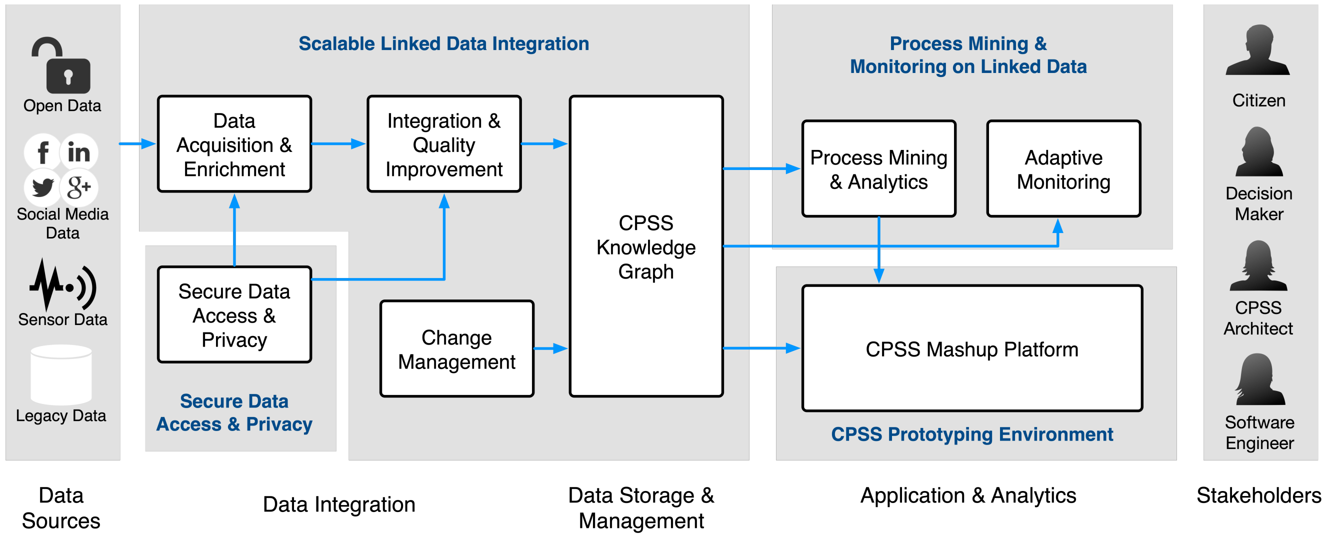


**Figure 1: CitySPIN architecture components and stakeholders.**

# 3 CPSS KNOWLEDGE GRAPH CONSTRUCTION

The broad scope of CPSSs and the large variety of technical infrastructure and data involved in them give rise to unique interoperability challenges when it comes to acquiring, enriching, integrating, managing, and processing data from various sources pertaining to the social, physical, and cyber dimensions of CPSSs. In CitySPIN a CPSS knowledge graph acts as an integration hub for all this heterogeneous data. In this section we describe the technologies used to construct this knowledge graph.

We rely on UnifiedViews[2], developed at Semantic Web Company (SWC), as a core building block for the knowledge graph construction. Specifically, data sources are aggregated and transformed using so-called Data Processing Units (DPU)s, which are assembled into data integration pipelines[3]. All input data is available in a structured format for further processing by subsequent elements of the pipeline. The pipelines transform data from various source formats and lift them into Resource Description Framework (RDF) format, a semantically explicit format standardized by the World Wide Web Consortium (W3C). This results in a knowledge graph that expresses the data using common standard vocabularies as well as vocabularies tailored to the use cases. Table 1 provides an overview of the key vocabularies used for the semantic alignment of the various datasets which underlie the public transportation planning use case used as an illustrative example in this paper.

The knowledge graph is stored into an RDF triple store – specifically Ontotext GraphDB[4]. Using the standard RDF query language SPARQL[5], input data is queried and further transformed or aggregated as needed by other components of the CitySPIN platform. The following paragraphs discuss the concepts applied here in more detail.

*Data Integration Lifecycle.* Heterogeneous sources such as social media data, sensor data and business intelligence data have to be made available to the CPSS for further processing. Connectors to the source systems hook into APIs, CSV repositories or direct database calls (data acquisition). Various steps follow to remove outliers and noise from data (data cleansing) as well as to refine their structure and align their content (data preparation). Finally data are merged, transformed and saved into pre-processable formats (data storage). The consolidated data are then available for subsequent analysis and reuse. Therefore, those data are fed back to the acquisition stage and the integration cycle restarts.

*Data Acquisition and Enrichment.* As we consistently follow an ontology-based data integration approach, we extract data according to a CPSS-wide ontology and transform the data into RDF. The RDF is, in turn, an interchange format which is used as the canonical one for further processing. By following the W3C standards for the Semantic Web, our approach ensures compatibility with a wide range of tools used in the CPSS stack.

*Semantic Alignment for Data Integration.* Aligning contents and data alongside an ontology enables the CPSS to access enriched contextual knowledge. This additional information forms a critical part of an integrated view on CPSS data and is essential for realizing the integrated user interface presenting the planning dashboard.

*Data Cleansing.* All gathered data is integrated and enriched by a processing pipeline, which lies at the functional core of the CPSS (e.g.: prediction, analysis, decision). Based on domain knowledge

[2] https://unifiedviews.eu
[3] cf. https://help.poolparty.biz/display/UDDOC/Basic+Concepts+for+DPU+developers for an introduction to the core concepts
[4] http://graphdb.ontotext.com
[5] https://www.w3.org/TR/sparql11-query/

| Name | Prefix | Namespace | Documentation | Purpose |
|---|---|---|---|---|
| Time Ontology | `time` | https://www.w3.org/2006/time# | [7] | modeling time |
| Geovocab geometry | `geom` | http://geovocab.org/geometry# | [2, 17] | desribing geographical regions |
| Geovocab spatial | `spatial` | http://geovocab.org/spatial# | [3, 17] | topological relations between features |
| wgs84 | `wgs` | http://www.w3.org/2003/01/geo/wgs84_pos# | [1] | lat(itude), long(itude) about spatially-located things |
| Event | `event` | http://w3id.org/cityspin/ns/event# | http://rebrand.ly/dmkrk0 | city event data (location, participants etc.) |
| Cellular data | `mobile` | http://w3id.org/cityspin/ns/mobile# | http://rebrand.ly/pkmq2j | Cellular location data |
| SPECIAL-CPSS | `special-cpss` | http://w3id.org/cityspin/ns/special-cpss# | http://rebrand.ly/5m33q2 | CPSS usage policy and consent specification |
| Transport | `tp` | http://w3id.org/cityspin/ns/transport# | http://rebrand.ly/ee83mf | structuring and annotating public transport data (e.g., stops, routes, schedules). |
| Disruption | `td` | http://purl.org/td/transportdisruption# | [6] | modelling travel and transport related events that have a disruptive impact on an agent's planned travel |
| Data Cube | `qb` | http://purl.org/linked-data/cube# | [8] | multi-dimensional data (e.g., district heating network statistics) |

**Table 1: Key vocabularies for semantic alignment**

and process knowledge, data are consolidated and made available for extraction and further processing by actuators, visualization and re-feeds into the learning pipeline.

*Knowledge Graph Storage.* The central processing pipeline acts as an interface to other algorithms, further user-driven explorations, or visual representations of the output. The loop-back to the Data Acquisition stage of the CPSS is realized through interim storage in a central triple store and actuation of external triggers.

## 4 CITYSPIN PROTOTYPING ENVIRONMENT

For the implementation of the prototyping environment, the CitySPIN project proposes an architecture inspired by the Presentation Abstraction Control (PAC) architectural pattern [10]. In this section, we adapt the PAC architecture to the CPSS needs and integrate it with the modular approach of Linked Widgets [19] and UnifiedViews [13] to develop a CPSS prototyping environment. In the following subsections, we illustrate the longitudinal section of the software architecture to describe the associations and information flows between the main logical components at large. In line with the PAC pattern, three-layered architecture of the CitySPIN CPSS prototyping environment consists of:

- the Back-end layer, in which data are loaded, pre-processed, and aggregated (abstraction) - details on this layer are previously discussed in Section 3, together with the CPSS Knowledge Graph construction and therefore will not explained further in this Section;
- the Service layer, in which those data are queried and analyzed to infer additional knowledge and later on generate prediction models (control) - cf. Section 4.1;
- the Front-end layer (presentation), from which users can access the prediction models and data analysis reports to monitor the current status of the infrastructure, explore the historic performance, and make informed decisions on the future settings (Section 4.2).

### 4.1 Service: Querying and Prediction

There are a wide range of services required in the CPSS context due to the diversity of application domains, use cases and scenarios. In our CPSS Prototyping environment, we focused on two main services: *(i)* Querying, and *(ii)* Prediction.

To cater for the reporting and predicting needs of a CPSS, our architecture includes an intermediate layer in which data are extracted from the *Querying* component and fed to the *Prediction* component or directly to the Dashboard of the frontend layer. The Querying component of our prototyping environment relies on the data endpoint provided by the back-end module for the execution of queries. In this component, we are using the W3C-standard SPARQL query language[6] for querying the integrated data. Furthermore, we can also use SPARQL Construct queries to encode rules for inferring new knowledge.

The Prediction component is designed to allow for the application of Machine Learning (ML) techniques aimed to derive prediction models that – based on historical data – can be used as a decision support system for CPSS stakeholders to react ahead of time to predicted arising situations [4]. Example prediction results include the forecast of numerical trends of variables under analysis, the identification of changes in the classification of recently collected data to raise alerts in case of anomalies, or the recommendation on the next operation to undergo in light of the recent developments of the data under observation.

ML algorithms require learning, validation, and testing phases on historical data, prior to, or alternated with, run-time processing or reinforcement on live data. To cater for these requirements, our architecture binds the Querying and Prediction components with data-flow associations that proceed in both ways: *(i)* from Querying to Prediction for data feed, and *(ii)* from Prediction to Querying for updates on the classifications and predictions made. Notice that this architectural choice allows for the marshalling and storage of models learned from the Prediction component for further reuse. This is the basis through which ex-post data analyses conducted via process mining can be readily available for decision support and monitoring via successive queries, as suggested in [9]. Finally, we emphasize that both the Querying and Prediction components are containers for diverse ML modules that can be used alternatively in multiple use cases, e.g., as a plugin for Linked Widgets.

[6]https://www.w3.org/TR/2013/REC-sparql11-query-20130321/

## 4.2 Frontend: Visualization and Decision Support

The Frontend layer allows users to interact with, get informed about, and interactively explore the integrated knowledge acquired from the data and augmented by the Prediction component. To this end, the use of Linked Widget Platform (LWP) [19] provides the necessary high degree of flexibility and customizability for CPSS prototyping. LWP combines semantic web and mashup concepts to support non-expert users in efficiently making use of various open and non-open data sources. In particular, the platform allows users to collaboratively and interactively integrate data in an ad-hoc and distributed manner. Each stakeholder can contribute their data and computing resources to a shared data processing flow in a shared interface that allows them to orchestrate the interaction among components within a CPSS.

Depending on their needs, users can directly construct analytical data flows, fine-tune queries, ML parameters, and visualization parameters within a single graphical interface. Bi-directional information flows between Querying and Dashboard components allow users to save their preferences and potentially store the relevant facts that they may have discovered in the Data Store. This would be crucial, for instance, to enable reinforcement learning for future projects building upon the CPSS Prototyping Environment, e.g., application developments based on the prototype results.

# 5 USE CASE

In this section, we introduce one of our real-world use cases in the public transport domain (Section 5.1), discuss data exploration (Section 5.2), describe the construction of the knowledge graph for the use case in Section 5.3 and illustrate the prototypical implementation within the CitySPIN platform (Section 5.4).

## 5.1 Mobility Use Case Description

The goal of the CitySPIN project is to deliver a generic platform for CPSS development that can support a wide variety of use cases in the context of city infrastructure services. To develop and prototype this platform, we chose use cases that cover a broad spectrum of smart city services (viz. public transportation and district heating network control) while, and on the other hand, exhibiting synergies in terms of data and component requirements.

In this paper, we focus on the *CitySPIN Event-Aware Mobility Planning* (CaMP) use case, which allows planners at Wiener Linien (WL) to estimate mobility demands of large-scale events in order to tailor the mobility planning accordingly. To cater for the needs of participants of such large-scale events, WL already actively adapts its transportation network schedule. In particular, the types, capacities and frequencies of vehicles in service during such events are currently decided by planners based on historic data about the number of attendants to recurring events, which are recorded in event planning protocols saved as .pdf files.

This current approach makes it difficult to plan for new or non-recurring events for which no planning protocols exist. Additionally, the current planning process does not take into account any feedback from social sources, e.g., such as event attendant profiles.

The CitySPIN project addresses this use case with the concept of Cyber-Physical Social Systems (CPSS), where citizens are seen as parts of city-wide infrastructures. Therefore, relevant data is collected from social sensors and data sources that act as proxies for human behavior (e.g., ticket sales). The relevant data is collected from a multitude of data sources (e.g., ticket sales, open government data, mobility data). The resulting Event-Aware Mobility Planner (CaMP) system enables WL planners to inspect attendance specific information for a large number of events drawn from a variety of data sources. It allows integrated and visual access to attendance data (i) from legacy (historic) sources, (ii) open data sources and (iii) social data.

## 5.2 Data exploration

To elicit requirements and how they could be addressed with available data, several workshops were held to *(i)* review the organizational and technical context of the real-world use case, *(ii)* conduct a high-level survey of available data sources within the use case partner's organization as well as externally available data, *(iii)* prioritize available data sources and the required data acquisition methods, *(iv)* evaluate design alternatives for data acquisition and semantic enrichment, *(v)* explore architectural options for a platform environment that supports integration of large-scale batch and high-frequency data flows.

This resulted in a set of preliminary data models, vocabularies, and guidelines used in the extraction, transformation and enrichment steps of the knowledge graph construction, as described next.

## 5.3 Mobility Knowledge Graph Construction

The knowledge graph constructed for the mobility use case covers *(i)* public transportation infrastructure (e.g., agencies, lines, schedule), *(ii)* internal planning protocols from WL, and *(iii)* event information.

*Public Transportation Data.* The first part of the mobility knowledge graph covers public transportation data in Vienna. Transportation data are often available as open data in GTFS format, which is widely used by Google for their online services[7]. This data format covers transport agencies/operators, the routes and the stop locations, trip schedule, and rules to describe the operation/service.

In the context of our prototype, we rely on the existing GTFS ontology[8] and GTFS CSV converter[9] to transform the original GTFS data provided by the City of Vienna[10] to produce our GTFS transportation KG. In total, the resulted KG contains more than 20 million triples, which is now available online as a SPARQL endpoint[11] hosted in an HDT[11] server.

*Event Information.* In addition to public transportation data, we include the event information from the Wien-Ticket open data API[12] as the second part of the mobility knowledge graph. The Wien-Ticket data contains general event information in Vienna, e.g., event name, address of the event location, and performer's name.

[7] https://gtfs.org
[8] https://github.com/OpenTransport/linked-gtfs
[9] https://github.com/OpenTransport/gtfs-csv2rdf
[10] https://www.data.gv.at/katalog/dataset/wiener-linien-fahrplandaten-gtfs-wien
[11] http://triple.ai.wu.ac.at
[12] http://data.opendataportal.at/dataset/wien-ticket-vorverkauf

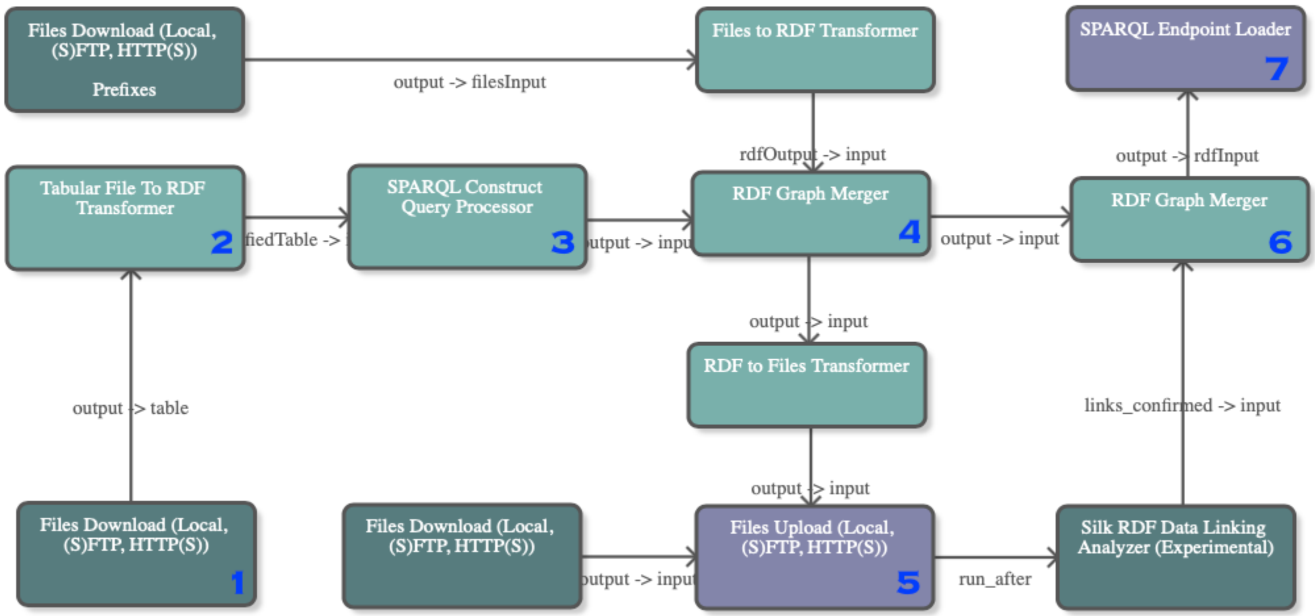


**Figure 2: Data Processing Units (DPUs) Orchestration for Wien-Ticket data**

To extract this data, we implement a data extraction workflow as a pipeline in UnifiedViews – depicted in Figure 2. The pipeline consists of three main stages: *(i)* The first step is the event data extraction from the open data API, which is originally provided in CSV format. This process downloads the original data from the API (#1), translates into RDF (#2 & #3), and merges it with a namespace graph (#4). *(ii)* After the event data is transformed into RDF format, the second step of the extraction performs the linking of the events and address dataset (#5). *(iii)* Finally, in the last step, the resulted linked graph is merged with the original event dataset (#6) and inserted into a triple store via SPARQL (#7).

As a result of this process, we extracted more than 2,2 million triples of event data. We do not yet provide the resulting data as open data due to server limitations, but we are investigating options for opening the dataset for public access in the future.

*WL Planning Protocols.* The third part of the mobility KG is transportation planning data, which originated from the internal transport planning protocols. The data is extracted from WL planning protocol documents, which are used internally to document mobility planners' measures taken in response to demand expectations, including those due to special event. Such measures include increasing the frequency of transportation lines that have stations in the vicinity of the event in a time interval covering the event's duration. The planning protocols are typically stored as Word or PDF documents, which makes automatic data extraction difficult. Parts of the challenges on this task includes dealing with various irregularities and inconsistencies in document layouts and extracting locally-used codes and abbreviations which are embedded within written comments. To address this issue, we employ a semi-automatic information extraction pipeline, using a combination of Natural Language Processing (NLP) techniques and human computation to extract the necessary information. In the end, we are able to extract information from more than 250 out of a set of 300 test planning documents, which accounts for more than 8,700 triples in total. We do not plan to make the raw information about this planning protocol public, as it may contain sensitive internal information.

## 5.4 Interactive Planning Support

To support the mobility use-case, we developed an interactive planning support tool (cf. Screenshot in Figure 3) by instantiating the CPSS prototyping environment. The intended user of this tool is the operation planning department at WL. In particular, the system is designed to support decisions on measures to optimize the transportation network in anticipation of a certain event, especially by taking into account historic records of such measures for the same type of events or for events that happened at the same or neighbouring venues.

We aim to support scenarios in which a transportation planner needs to decide traffic adjustment measures for an upcoming event. In this scenario, the planner will start by browsing a list of *upcoming events* - as shown by the top left widget in Figure 3 based on second part of the knowledge graph on event information. From this list, they then choose a focus event (e.g., "Cirque du Soleil - TOTEM") for planning adjustments. Based on the selected event, a *geo-map mashup* will visualise the location of this event as a green pin on the map.

From this point, there are several possibilities for the planner to choose as follows:

- inspect the list of events that took place in nearby locations and for which a planning protocol has been produced (*"event*

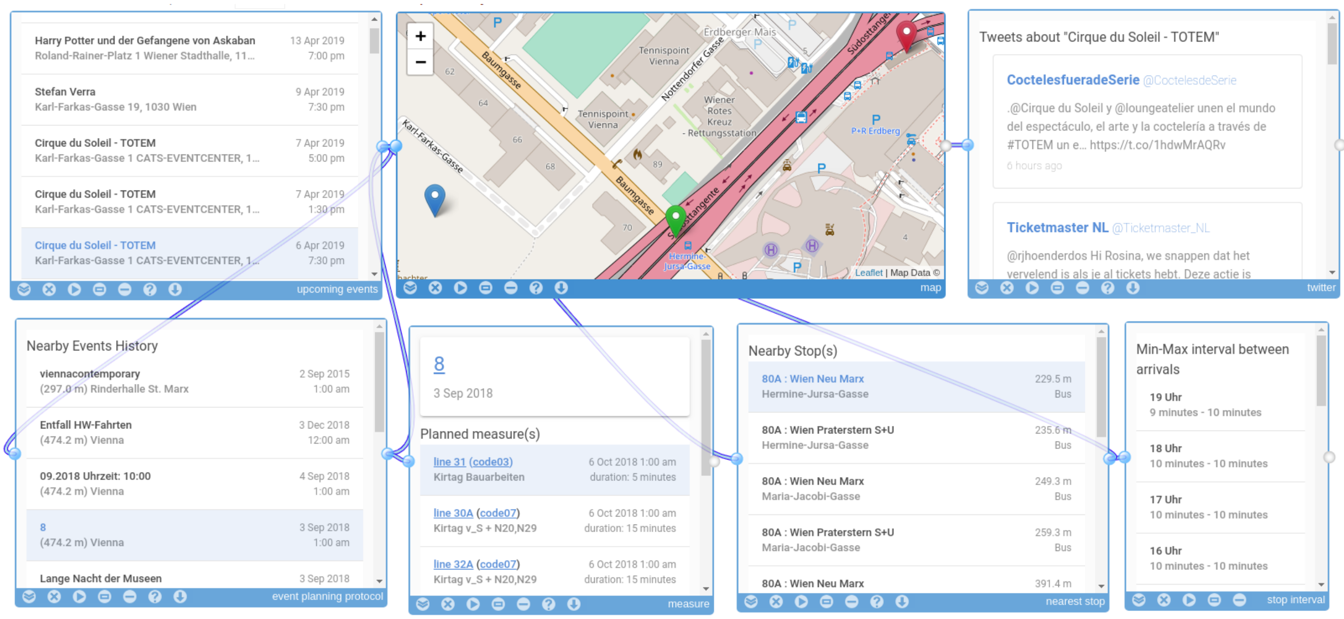


**Figure 3: Events-Public transportation planning prototype**

*planning protocol"* widget). This widget draws on the data extracted from historic planning protocols –which is the third part of the knowledge graph on event information– and allows planners to easily access the decisions taken for the nearby events (e.g., for event "8" the frequency of 3 lines has been set to 5, 15, 15 minutes respectively).

- identify the public transportation stops in the immediate geographic vicinity of the focus event's location (*"Nearby Stops"* widget) based on first part of the knowledge graph on GTFS public transportation data. In our example scenario, *Hermine-Jursa-Gasse* and *Maria-Jacobi-Gasse* are two nearest stops to the event location.
- browse social media messages related to the event in order to identify any additional information from social signals (e.g., general satisfaction with the transportation support etc.)

These functionalities for the CaMP use case are made available by the underlying infrastructures which *(i)* ensures that data from various data sources is loaded and semantically integrated so that it can be *(ii)* visualised using a visual widget-based platform where various widget types can be combined into mashups in order to support the exploration of relevant planning information by the transportation planners.

We plan to continue the development of the current CaMP prototype[13] with the integration of additional social data, in particular data from mobile operators and results from the process mining components that should also allow planners to get a better understanding of the social aspect of the CPSS.

## 6 RELATED WORK

A number of *vision papers* explored the applicability of CPSS in given domains. In the *military* domain, the CPSS concept fits naturally by spanning the boundaries of and connecting physical networks, the cyberspace, mental space and social networks that are the main components of command and control systems [14]. By integrating these spaces, CPSS bring benefits such as synchronization across the spaces, self-adaptation and "chaotic control" as an alternative to precise control in order to deal with inherent uncertainties in the domain. In *manufacturing* [23], a new industrial revolution is emerging enabled by socio-cyber-physical system (SCPS) which combine social elements with smart manufacturing thanks to the four technical pillars of Internet of Things (IoT at the physical layer), Internet of Knowledge (IoK) and Internet of Services (IoS) at the cyber level, and Internet of People (IoP). A vision of Physical-Cyber-Social computing enabled by knowledge technologies and illustrated with an application in the medical domain is discussed in [18]. *Smart City* applications inherently subscribe to the concept of CPSS [5] as we also demonstrate in our own project with a transportation and a sustainable energy related use case.

Common to CPSS efforts in all domains is that they primarily focus on describing concrete systems, and how they function. In CitySPIN, on the contrary, we aim to support the *engineering* phase of these systems. A particular focus is on the ETL and data integration process which takes up considerable effort. Similarly to our projects, the QROWD project[14] also develops semantics based data integration approaches. However, these do not support privacy-aware data integration as has been done in CitySPIN.

[13] http://rebrand.ly/mobility-mashup

[14] http://qrowd-project.eu/

## 7 CONCLUSION AND OUTLOOK

In this paper, we provided an overview of the CitySPIN CPSSs platform and development approach focusing mainly on a data engineering perspective. Using multiple use cases developed with stakeholders in a city-scale context as a lense to explore challenges of heterogeneity, privacy, and process dynamics, we motivated the design of the CitySPIN architecture described in this paper. We illustrated the prototypical implementation of this architecture by means of a real-world use case in public transportation planning.

In future work, we will investigate the integration of more real-time sensing and actuation components into the platform, which will enable CPSS developers to integrate additional social components into the CPSS loop. In the long term, this could facilitate the implementation of adaptive strategies in various use cases in the mobility and energy domains.

## ACKNOWLEDGMENTS

This work was funded by the Austrian Research Promotion Agency FFG under grant 861213 (CitySPIN).

## REFERENCES

[1] 2003. Basic Geo (WGS84 lat/long) Vocabulary. (2003). https://www.w3.org/2003/01/geo/
[2] 2012. NeoGeo Geometry Ontology. (2012). http://geovocab.org/geometry
[3] 2012. NeoGeo Spatial Ontology. (2012). http://geovocab.org/spatial
[4] Ethem Alpaydin. 2009. *Introduction to machine learning*. MIT press.
[5] Christos G. Cassandras. 2016. Smart Cities as Cyber-Physical Social Systems. *Engineering* 2, 2 (2016), 156 – 158. https://doi.org/10.1016/J.ENG.2016.02.012
[6] David Corsar, Milan Markovic, Peter Edwards, and John D. Nelson. 2015. The Transport Disruption Ontology. In *The Semantic Web - ISWC 2015*, Marcelo Arenas, Oscar Corcho, Elena Simperl, Markus Strohmaier, Mathieu d'Aquin, Kavitha Srinivas, Paul Groth, Michel Dumontier, Jeff Heflin, Krishnaprasad Thirunarayan, and Steffen Staab (Eds.). Vol. 9367. Springer International Publishing, Cham, 329–336. https://doi.org/10.1007/978-3-319-25010-6_22
[7] Simon Cox, Chris Little, Jerry R. Hobbs, and Feng Pan. 2018. *Time Ontology in OWL*. W3C Recommendation. W3C. https://www.w3.org/TR/owl-time/
[8] Richard Cyganiak, Dave Reynolds, and Jeni Tennison. 2014. *The RDF data cube vocabulary*. W3C Recommendation. W3C. https://www.w3.org/TR/vocab-data-cube/.
[9] Claudio Di Ciccio, Fajar J. Ekaputra, Alessio Cecconi, Andreas Ekelhart, and Elmar Kiesling. 2019. Finding Non-compliances with Declarative Process Constraints through Semantic Technologies. In *CAiSE Forum*. Springer, 60–74. https://doi.org/10.1007/978-3-030-21297-1_6
[10] A Dix, J Finlay, GD Abowd, and R Beale. 2004. Human-computer interaction: Pearson prentice hall. *Inc, England* (2004).
[11] Javier D Fernández, Miguel A Martínez-Prieto, Claudio Gutiérrez, Axel Polleres, and Mario Arias. 2013. Binary RDF representation for publication and exchange (HDT). *Web Semantics: Science, Services and Agents on the World Wide Web* 19 (2013), 22–41.
[12] W. Guo, Y. Zhang, and L. Li. 2015. The integration of CPS, CPSS, and ITS: A focus on data. *Tsinghua Science and Technology* 20, 4 (August 2015), 327–335. https://doi.org/10.1109/TST.2015.7173449
[13] Tomas Knap, Petr Skoda, Jakub Klímek, and Martin Necaskỳ. 2015. UnifiedViews: Towards ETL Tool for Simple yet Powerfull RDF Data Management.. In *DATESO*. 111–120.
[14] Z. Liu, D. Yang, D. Wen, W. Zhang, and W. Mao. 2011. Cyber-Physical-Social Systems for Command and Control. *IEEE Intelligent Systems* 26, 4 (2011), 92–96.
[15] Fabrizio Maria Maggi, Claudio Di Ciccio, Chiara Di Francescomarino, and Taavi Kala. 2018. Parallel algorithms for the automated discovery of declarative process models. *Inf. Syst.* 74, Part 2 (2018), 136–152. https://doi.org/10.1016/j.is.2017.12.002
[16] Angelika Musil, Juergen Musil, Danny Weyns, Tomas Bures, Henry Muccini, and Mohammad Sharaf. 2017. Patterns for Self-Adaptation in Cyber-Physical Systems. In *Multi-Disciplinary Engineering for Cyber-Physical Production Systems*, Stefan Biffl, Arndt Lüder, and Detlef Gerhard (Eds.). Springer International Publishing, Chapter 13, 331–368.
[17] Barry Norton, Luis M. Vilches, Alexander De León, John Goodwin, Claus Stadler, Suchith Anand, Dominic Harries, Boris Villazón-Terrazas, and Ghislain A. Atemezing. 2012. NeoGeo Vocabulary Specification. (2012). http://geovocab.org/doc/neogeo/
[18] A. Sheth, P. Anantharam, and C. Henson. 2013. Physical-Cyber-Social Computing: An Early 21st Century Approach. *IEEE Intelligent Systems* 28, 1 (Jan 2013), 78–82. https://doi.org/10.1109/MIS.2013.20
[19] Tuan-Dat Trinh, Peter Wetz, Ba-Lam Do, Elmar Kiesling, and A Min Tjoa. 2015. Distributed mashups: a collaborative approach to data integration. *International Journal of Web Information Systems* 11, 3 (2015), 370–396.
[20] Wil M. P. van der Aalst. 2016. *Process Mining - Data Science in Action, Second Edition*. Springer. https://doi.org/10.1007/978-3-662-49851-4
[21] Y. Wang, W. Dai, B. Zhang, J. Ma, and A. V. Vasilakos. 2017. Word of Mouth Mobile Crowdsourcing: Increasing Awareness of Physical, Cyber, and Social Interactions. *IEEE MultiMedia* 24, 4 (October 2017), 26–37. https://doi.org/10.1109/MMUL.2017.4031317
[22] G. Xiong, F. Zhu, X. Liu, X. Dong, W. Huang, S. Chen, and K. Zhao. 2015. Cyber-physical-social system in intelligent transportation. *IEEE/CAA Journal of Automatica Sinica* 2, 3 (July 2015), 320–333. https://doi.org/10.1109/JAS.2015.7152667
[23] Xifan Yao and Yingzi Lin. 2016. Emerging manufacturing paradigm shifts for the incoming industrial revolution. *The International Journal of Advanced Manufacturing Technology* 85, 5 (01 Jul 2016), 1665–1676.