Linking Process Models and Operating Data for Exploration and Visualization

Ken WENZEL [a,1], Marcel TISZTL [a]
[a] *Fraunhofer IWU, Department for Production Management, Germany*

Abstract. Data integration between different life cycle stages of a production system is a crucial requirement for the realization of continuous computer support what is often referred to as the Digital Factory. This paper presents an approach that leverages planning data within the operation phase of a production system by applying Semantic Web technologies and linked data concepts.

Keywords. production system, life cycle, process planning, semantic web, sensor networks

Introduction

When considering the design and operation of a production system, one can observe that there exists an information gap between those life cycle phases [1]. This usually results in a complete redesign of models for observation and control of such systems instead of reusing structural and functional knowledge that was already created during the design phase. We believe that this is due to the wide range of tools and complex data formats making it hard to transfer knowledge from one phase to another. Our approach transforms existing process models of production systems into a linked data representation and combines them with ontologies for Semantic Sensor Networks and units of measurement to enable the linkage with collected operating data. The development of our concepts is based on a usage scenario from the automotive industry where media consumption data for a body-in-white assembly line is collected and should be linked to processes, products and resources for the purpose of visualization and analysis.

Section 1 outlines the related ontologies and applications used in our approach. Section 2 introduces a simple ontology for the description of manufacturing processes. Based on this ontology an approach for exploration and visualization of operating data is introduced by Section 3. Finally, a summary and outlook is given by Section 5.

1. Overview on related ontologies and software systems

Figure 1 gives an overview on ontologies which were created for our approach to represent and integrate process models with operating data. These encompass the manu-

[1] Corresponding Author: Ken Wenzel, Fraunhofer-Institute for Machine Tools and Forming Technology IWU, Department for Production Management, Reichenhainer Str. 88, 09126 Chemnitz, Germany; E-mail: ken.wenzel@iwu.fraunhofer.de.

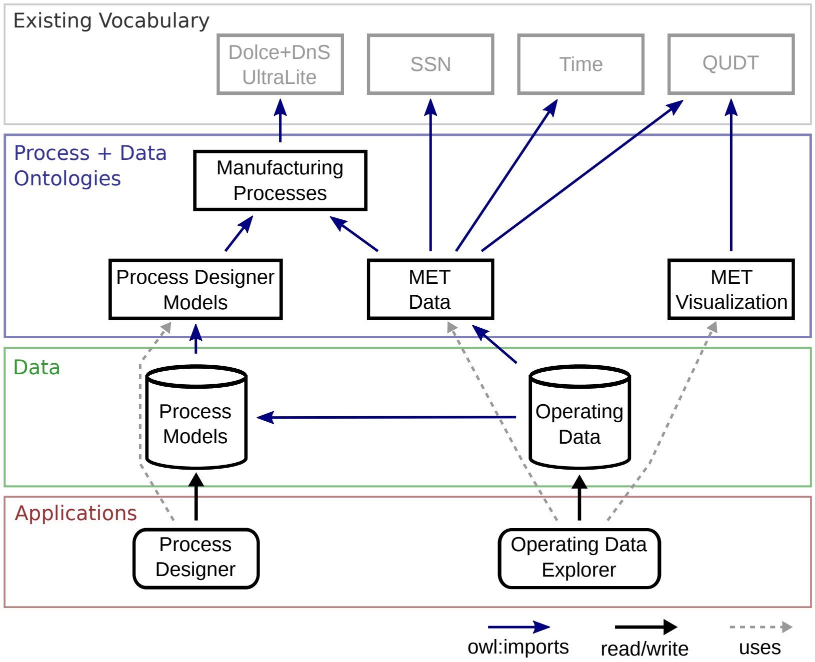

Figure 1. Ontologies and software systems used by our approach.

facturing process ontology (Section 2) for the description of process models, the MET data ontology for the representation of measurement data (Section 3) as well as the MET visualization ontology for the mapping of measured properties to suitable graphical charts (Section 4). The latter two are combined within our Measurements and Evaluation Toolkit (MET) – a collection of vocabularies and software tools to acquire, manage and visualize measurement data. The existing vocabulary that was reused for the creation of our ontologies is shown at the top of Figure 1.

Additionally, a domain-specific ontology was developed as an extension of the manufacturing process ontology to (partially) represent *Tecnomatix Process Designer* models as linked data. These process models are automatically imported from their proprietary XML serialization and transformed into RDF data.

Finally, the presented approach is implemented within the *Operating Data Explorer* using the Knowledge Modeling and Management Architecture (KOMMA) [2].

2. A lightweight ontology for manufacturing processes

Diverse engineering disciplines tried to adopt ontologies for describing their respective domains. For example, the research projects COGents and IMPROVE [3] demonstrated that ontologies are a well-suited means to describe models of chemical process systems. Besides these, there are also some efforts towards creating base ontologies for the manufacturing domain. MASON [5] defines an upper ontology for manufacturing and gives

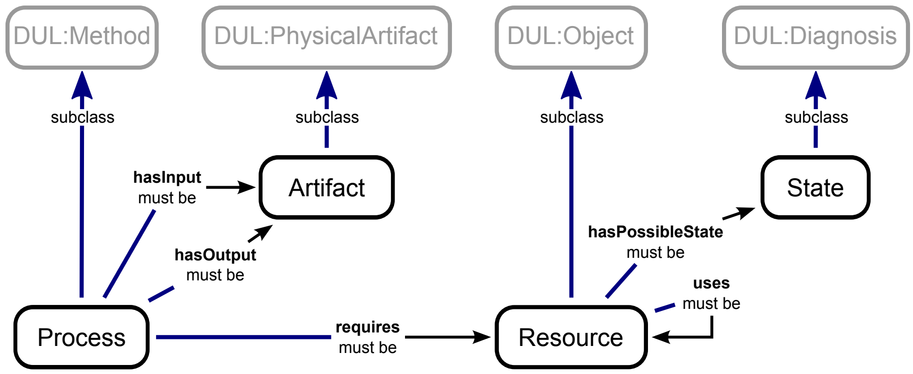

Figure 2. Manufacturing process ontology.

examples of its usage within the fields of cost estimation and production control. The ADACOR ontology [4] contains concepts to describe manufacturing systems for the purpose of scheduling and control. Both ontologies capture some common but also some distinct parts of the manufacturing domain.

Following the principle of minimal ontological commitment as proposed by Gruber [6], we argue that separation of concerns into even more lightweight ontologies simplifies their reuse and accelerates their adoption. Hence, we have created the simple ontology depicted by Figure 2 for the description of manufacturing operations with their related products and resources. This ontology is also aligned to DOLCE+DnS UltraLite (DUL)[2], an enhanced version of the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [7].

The ontology is loosely based on the core idea of the Structured Analysis and Design Technique (SADT) [8] that is able to describe systems as a hierarchy of functions. In our case those functions are manufacturing *Processes* which may have one or more *Artifacts* as inputs and outputs and may require one or more *Resources* for operation. The ontology introduces also the concept *State*, allowing a more specific description of resources, especially in regard to operating data.

The proposed ontology does intentionally not provide any concepts for the description of raw materials, tools, workers and other application specific entities. Also deeper classification of artifacts or processes as done within MASON is omitted.

The alignment to DUL allows easy extension of the ontology for various use cases. *Process* is considered a subclass of *DUL:Method* to express that this concept should be used to describe manufacturing operations in the sense of process planning. *DUL:PhysicalArtifact* is chosen as a superclass of *Artifact* since inputs and outputs of manufacturing operations are usually physical objects with an intended function. Our model unifies SADT's notion of control and mechanisms into one *Resource* concept that is a subclass of *DUL:Object*. *Resources* can be further described by using *States* representing a *DUL:Diagnosis* of these systems.

[2] http://ontologydesignpatterns.org/wiki/Ontology:DOLCE+DnS_UltraLite

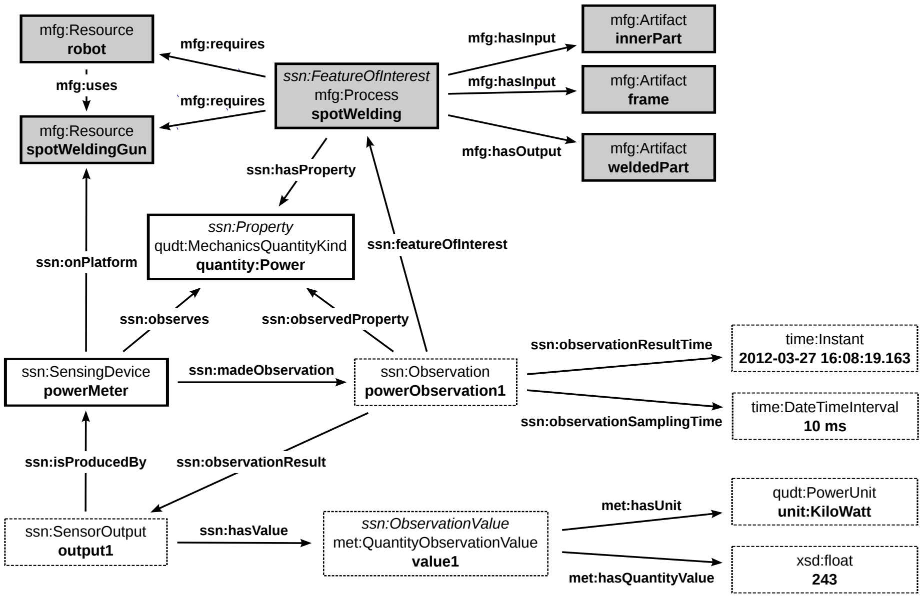

Figure 3. Example of an electric power observation.

3. Representation of operating data

The W3C Semantic Sensor Networks Incubator Group (SSN-XG)[3] has built an extensive ontology for describing sensor networks and related observations. For this reason, their work encompasses an in-depth review of existing sensor and observation ontologies where selected concepts were used to create the Semantic Sensor Network (SSN) ontology. This ontology intentionally excludes descriptions of domain concepts like time or location information to enable its applicability for multiple use cases and domains.

We are mostly using concepts of the SSN ontology that were developed for the use case of data discovery and linking. These concepts allow to describe *Observations* made by a *SensingDevice* regarding some *Properties* of a certain *FeatureOfInterest*.

In the case of operating data the *FeatureOfInterest* is usually either the process itself or the involved resources and product artifacts. When combining the SSN ontology with the ontology for manufacturing processes introduced in Section 2 the feature of interest of an observation may be an instance from one of the classes *Process*, *Resource*, *State* or *Artifact*. More complex use cases can be covered by using instances of *DUL:Event* as feature of interest.

Figure 3 denotes an example of an electric power observation expressed by applying the SSN ontology in combination with the manufacturing process ontology. The example covers a body-in-white assembly process with spot welding where the actual power consumed by the spot welding gun is observed. For the representation of observable properties and their associated units the QUDT[4] (Quantities, Units, Dimensions and Data Types

[3]http://www.w3.org/2005/Incubator/ssn/
[4]http://qudt.org/

in OWL and XML) ontology is used. An alternate ontology with comparable coverage is the Measurement Units Ontology[5] (MUO). Both ontologies provide a formal framework for defining measurable properties (quantities in QUDT, qualities in MUO) with corresponding base units and their derived forms. Additionally, both ontologies provide a set of predefined properties and unit systems like SI or CGS. MUO's data is extracted from UCUM[6], the Unified Code for Units of Measure, while QUDT uses its own definitions.

Time points and intervals are expressed by using the Time Ontology in OWL[7] (OWL-Time).

For representing observation result values a special concept *QuantityObservationValue* as subclass of *ssn:ObservationValue* is introduced by our MET data ontology. It allows to describe measured quantities (*met:hasQuantityValue*) and their associated units (*met:hasUnit*).

4. Exploration and visualization

The semantic linking of operating data, aligned with the SSN, to the manufacturing process ontology, like we have shown above, can support an analyst in exploring and examining the collected data, for example, to identify specific resource usage patterns. Therefore, we want to demonstrate, how visualization and navigation on the described model can be achieved. The starting point is the process model of the production system that can be explored within the process, resource and artifact dimensions. Each element from each dimension may have a hierarchical and topological structure.

Moreover, we want to support the visualization of a *Process*, a *Resource* or an *Artifact* by generating a chart of its properties and related observations. Figure 4 exemplifies this compound visualization for a selected *Resource*.

4.1. Generic diagrams for observed properties

Data visualization of *FeatureOfInterests* is a complex matter, because their observed properties, that should be visualized, may differ in some aspects. For example, some properties like power consumption and temperature have numeric values for individual points in time and are suitably visualized by XY charts. Other properties like the current system state or the modified artifacts represent some kind of state that is valid over a time interval (e.g. the duration of a manufacturing operation) and can be suitably visualized by gantt or bar charts. To support a variety of visualizations, the MET-VIZ (Measurements and Evaluation Toolkit - Visualization) ontology introduces concepts to formally describe the associations between visualizable properties and suitable charts. An example is depicted in Figure 5.

The ontology defines the classes *Chart* and *ChartDescription*. Each chart supported by our visualization framework is represented as an instance of *Chart* and observed properties are associated to these charts by using *ChartDescriptions*.

[5] http://idi.fundacionctic.org/muo/muo-vocab.html
[6] http://unitsofmeasure.org/
[7] http://www.w3.org/TR/owl-time/

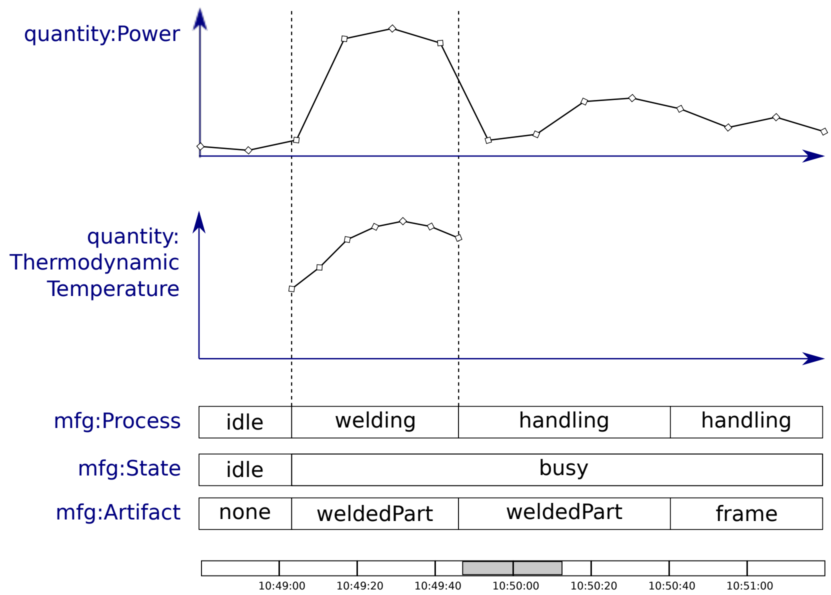

Figure 4. Visualization of observed property values for a given *Resource*.

4.2. Visualization framework based on data-driven documents

Like Figure 4 denotes, each observed property of a given *FeatureOfInterest* is visualized by a specific chart. These charts are combined into a composite chart with a timeline, where the user can select the global time window for all contained charts. To support this functionality, we have developed a visualization framework that implements versatile charts based on data-driven documents (d3) [10] and their corresponding Javascript implementation D3.js[8]. D3 allows to use powerful declarative transformations between data and SVG representations. This eases the creation of generic charts and highly simplifies the development of domain-specific visualizations for various use cases. D3's transformation capabilities are leveraged to define basic generic diagrams which can be composed with each other to create compound visualizations of multiple properties.

These compositions are supported by two design rationales of our framework. The first one is that the chart's data access is controlled by a dedicated data provider interface.

[8]http://mbostock.github.com/d3/

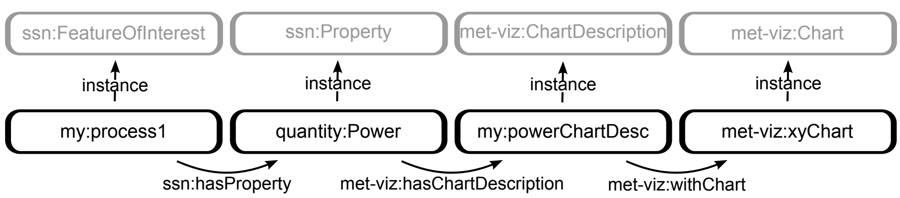

Figure 5. Associate properties with suitable charts.

If a chart is part of a composite then the composite assigns a specific data provider to this chart. This enables the composite to simply pass through data requests and responses from and to its children or intercept the communication to implement data conversion or caching (e.g. within a time window).

The second building block is the interconnection of multiple chart axes. Thus a composite can define the axis of a contained chart as parent of an axis of another child. If a parent axis is changed in its dimensions or selection then it propagates these changes to the respective child axes.

4.3. Composite visualization of observed properties

For the creation of a composite visualization, the database is queried for all observed properties of a given *FeatureOfInterest*. This is exemplified by the following SPARQL[9] query.

```
PREFIX ssn:<http://purloclc.org/NET/ssnx/ssn#>
PREFIX met-viz:<http://enilink.net/vocab/met/visualization#>

SELECT DISTINCT ?property ?chart WHERE {
  [ a ssn:FeatureOfInterest; ssn:hasProperty ?property ] .
  ?property met-viz:hasChartDescription ?description .
  ?description met-viz:withChart ?chart
}
```

Needless to say, the user can narrow down the search results to constrain the visualization to certain properties. For each of these selected properties an instance of the corresponding chart is added to the composite.

A timeline within the top-level composite enables the user to select a specific time window whose corresponding data should be visualized. Based on this selection all charts representing observed properties are updated to show the correct values within the selected time interval. Our visualization framework supports this use case by interconnecting the charts and their axes like described above. After selecting the time window, each chart individually retrieves its data by using a data provider that itself queries the database.

The following SPARQL query retrieves measurement data for a given time window and an observed property. This query selects all observations for a given *Property* of a *FeatureOfInterest* and retrieves the associated data values by filtering the results using the selected time window with start time and end time points.

```
PREFIX dul:<http://www.loa-cnr.it/ontologies/DUL.owl#>
PREFIX time:<http://www.w3.org/2006/time#>
PREFIX ssn:<http://purloclc.org/NET/ssnx/ssn#>

SELECT ?value ?rtime WHERE {
  ?o a ssn:Observation;
    ssn:observedProperty ?property; ssn:featureOfInterest ?foi;
```

[9] http://www.w3.org/TR/rdf-sparql-query/

```
    ssn:observationResultTime [
      time:inXSDDateTime ?rtime
    ];
    ssn:observationResult [
      ssn:hasValue [ dul:hasRegionDataValue ?value ]
    ] .
  FILTER (?rtime > ?startTime && ?rtime < ?endTime)
}
```

4.4. Navigation using the manufacturing process ontology

We have shown how observed properties of a particular *FeatureOfInterest* (a *Process*, a *Resource* or an *Artifact* of the manufacturing process ontology) can be visualized. However, it would be helpful to have some kind of visualization about their relations to each other and also to directly use these relations to switch between different *FeatureOfInterests*. Hence, additionally to the visualization of the properties of a *FeatureOfInterest* the following information can be displayed:

- If the visualized *FeatureOfInterest* is a *Resource* then we can visualize all the *Processes* that were active on that *Resource* in a gantt chart.
- If the visualized *FeatureOfInterest* is a *Process* then we can visualize the states of its required *Resources* in a gantt chart and set a reference to its input and output *Artifacts*.
- If the visualized *FeatureOfInterest* is an *Artifact* then we can visualize all *Processes* that caused changes and their corresponding *Resources* in a gantt chart.

This can be done by querying the process model (expressed using our manufacturing process ontology) for this information using SPARQL and by visualizing the results accordingly. The following query exemplifies the retrieval of active processes belonging to a resource. It assumes that the current state of processes (e.g. active or inactive) is also observed. For example, an observed property *ex:ProcessState* of a process may have the values *ex:ActiveState* or *ex:InactiveState* where the latter may be instances of *mfg:State*. Now the query retrieves all processes with their corresponding start times and durations that require the given *Resource* and were active at some point in time.

Please note that it would also be possible to model a resource's current active process as one of its observed properties (*ssn:observedProperty*).

```
PREFIX dul:<http://www.loa-cnr.it/ontologies/DUL.owl#>
PREFIX time:<http://www.w3.org/2006/time#>
PREFIX ssn:<http://purloclc.org/NET/ssnx/ssn#>
PREFIX mfg:<http://enilink.net/vocab/manufacturing#>
PREFIX ex:<http://example.org/process-data#>

SELECT ?process ?startTime ?duration WHERE {
  ?process mfg:requires ?resource .
  ?o a ssn:Observation;
    ssn:featureOfInterest ?process;
```

```
    ssn:observedProperty ex:ProcessState;
    ssn:observationResultTime [ time:inXSDDateTime ?startTime ];
    ssn:observationSamplingTime ?duration;
    ssn:observationResult [
      ssn:hasValue [ dul:hasRegionDataValue ex:ActiveState ]
    ] .
}
```

5. Conclusions and future work

We have introduced a lightweight ontology for describing manufacturing processes. Due to the alignment with DOLCE+DnS UltraLite an easy extension and combination with other vocabularies is ensured. Moreover, an approach was presented that combines our ontology with the SSN vocabulary to represent operating data of production systems and link it to elements of a process model. Based on these results we have shown the creation of mashups with charts for multiple observed properties by leveraging ontologies and advanced visualization technologies. This is supported by linking to an existing process model that improves the navigation and filtering of operating data.

Our future work will investigate how computation rules for indicators can be expressed by embedding mathematical formulas into RDF data sets [11]. This technology will allow the automatic computation of aggregated performance information based on the linked representation of process models and operating data.

Moreover, while currently data has only been collected from a running production system, the next steps may investigate how ontologies can help to integrate rules for manufacturing control into process models for using them within the operating phase. For example, we are currently working on augmentation of process and infrastructure models with additional knowledge and rules for energy-oriented control of production systems. The resulting solution should be able to operate a production system along with its infrastructure in an energy-efficient way.

Acknowledgements

The work presented in this paper is co-funded within the Cluster of Excellence Energy-Efficient Product and Process Innovation in Production Engineering (eniPROD®) by the European Union (European Regional Development Fund) and the Free State of Saxony.

References

[1] B. Denkena, M. Shpitalni, P. Kowalski, G. Molcho, and Y. Zipori, *Knowledge management in process planning*, CIRP Annals - Manufacturing Technology, vol. 56, no. 1, 2007, pp. 175–180.

[2] K. Wenzel, *Ontology-Driven Application Architectures with KOMMA*, 7th International Workshop on Semantic Web Enabled Software Engineering, Bonn, 2011. `http://iswc2011.semanticweb.org/fileadmin/iswc/Papers/Workshops/SWESE/6.pdf`

[3] J. Morbach, A. Wiesner, and W. Marquardt, *OntoCAPE – A (re)usable ontology for computer-aided process engineering*, Computers & Chemical Engineering, vol. 33, no. 10, 2009, pp. 1546–1556.

[4] S. Borgo and P. Leito, *Foundations for a Core Ontology of Manufacturing*, in: Ontologies, vol. 14, R. Sharman, R. Kishore, and R. Ramesh, Eds. Springer US, 2007, pp. 751–775.

[5] S. Lemaignan, A. Siadat, J.-Y. Dantan, and A. Semenenko, *MASON: A Proposal For An Ontology Of Manufacturing Domain*, in: IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications, 2006. DIS 2006, 2006, pp. 195–200.

[6] T. Gruber, *Toward Principles for the Design of Ontologies Used for Knowledge Sharing*, International Journal Human-Computer Studies Vol. 43, Issues 5–6, Novemer 1995, pp. 907–928.

[7] A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider, *Sweetening Ontologies with DOLCE*, in: Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management, Ontologies and the Semantic Web, London, 2002, pp. 166–181.

[8] D. Marca and C. L. McGowan, *SADT : structured analysis and design technique*, McGraw-Hill, New York 1988.

[9] J. R. Hobbs and F. Pan, *An Ontology of Time for the Semantic Web*, ACM Transactions on Asian Language Information Processing, vol. 3, 2004, pp. 66–85.

[10] M. Bostock, V. Ogievetsky, and J. Heer, *D3: Data-Driven Documents*, in: IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis), 2011.

[11] K. Wenzel and H. Reinhardt, *Mathematical Computations for Linked Data Applications with OpenMath*, 24th OpenMath Workshop, Bremen, 2012. `http://www.informatik.uni-bremen.de/cicm2012/papers/OM2012/paper_1.pdf`