12TH INTERNATIONAL DEPENDENCY AND STRUCTURE MODELLING CONFERENCE, DSM'10
22 – 23 JULY 2010, CAMBRIDGE, UK

APPROACH TOWARDS A MORE FLEXIBLE HANDLING OF DOMAINS IN COMPLEX SYSTEMS

Andreas Kohn and Udo Lindemann
Institute of Product Development, Technische Universität München

Keywords: flexible attributes, system modelling, system visualisation

1 INTRODUCTION

Design Structure Matrix (DSM), Domain Mapping Matrix (DMM) and Multiple Domain Matrix (MDM) methodologies offer a wide range for modelling and analysing complex systems (Lindemann et al., 2009). All these methodologies model the system by initially assigning the contained elements to domains. The later analysis of the system's structure are then limited to this domain oriented order. This paper presents an approach to handle the domains of a complex system in a more flexible way with the purpose to apply multi-domain algorithms and visualisation techniques in the field of structural complexity management more flexible. Therewith, alternative views on the system's structure are offered. The presented approach aims at a new way of thinking about the modelling of complex systems as well as their visualisation and analysis. The developed tool SysViz is finally used to illustrate and demonstrate the presented approach.

2 BACKGROUND OF THE APPROACH

Providing a deeper insight into the background of this approach, an overview of the current handling of domains in the DSM community, the use of attributes in DSMs and MDMs, and also promising findings in the field of ontology development is given.

2.1 Handling domains in the DSM community

In its beginnings, the DSM was used to analyse the structure of the system design process (Steward, 1981). Afterwards, the DSM methodology for modelling and analysing system structures was applied in a multitude of different projects in which elements of different domains were focused (for an overview see (Browning, 2001)). Danilovic introduced a matrix containing relations of elements of two types – he called it Multiple Domain Matrix (DMM) (Danilovic, 2001). Similar to the development of the DSM, the DMM was soon used not only for the first proposed pair of domains (product architecture and organization). Lindeman and Maurer developed a methodology combining both DSM and DMM methodologies under the framework of Structural Complexity Management (SCM) (Lindemann et al., 2009). The SCM methodology, in general, supports the handling of multiple-domain systems and provides a five-step procedure that supports users in system definition, information acquisition, deduction of indirect dependencies, structure analysis, and the application on the product design. For the deduction of indirect dependencies and structure analysis, algorithms for calculating DSMs from DMMs are used. The analyses are computed in the Multiple Domain Matrix (MDM) which consists of at least two, but theoretically up to an infinite number of domains. The domains (and therewith the granularity of the model) are chosen either according to the intended results of the later analysis or according to the existing information sources (Lindemann et al., 2009).

2.2 Use of attributes in DSMs and MDMs

Attributes were used in the DSM community so far for offering additional information about elements of a system. Biedermann et al. used attributes to consider cost estimations of a product (Biedermann et al., 2007) and Braun and Deubzer for new variant management (Braun and Deubzer, 2007). Eppinger gives an overview of the use of attributes by showing examples of different systems modelled considering attributes of elements in a DSM (Eppinger, 2009). For indicating the value of an attribute, the elements of the analysed systems are grouped by colouring the cells of the matrix depending to the value of their attributes.

2.3. Ontology development and benefits for DSM methodologies

Achievements in the field of ontology development (and the application of semantic technology in general) seem to offer very promising input for the DSM community. Ontologies are generally used to represent the knowledge of a certain field of interest. Different formalized languages exist for modelling the knowledge stored in an ontology (DAML, RDF, OWL, and so forth) (Corcho et al., 2003). Syldatke et al. claim that semantic technologies can facilitate and enhance analysis in the scope of classic DSM/DMM approaches (Syldatke et al., 2008). He compares the life cycle of an ontology to the life cycle of a MDM and finds several similarities. According to him, advantages of semantic technologies compared to DSM/DMM approaches lie in the formalized standards, reasoning capabilities, defining rules for querying the ontology and the reuse of existing knowledge.

In addition to the advantages identified by Syldatke et al., the authors of this paper emphasize the promising meaning of multi-inheritance in ontologies. In ontology development, a concept can be attached to more than one superconcept. This multi-inheritance offers a larger flexibility in modelling a system by overcoming the strict hierarchy-based modelling of other approaches.

3. OFFERING MORE FLEXIBILITY FOR THE HANDLING OF COMPLEX SYSTEMS

3.1 Flexible way of handling domains

By combining domains and attributes in the field of DSM, DMM, and primarily MDM methodologies with advantages of common modelling techniques in ontology development, a more flexible way of handling domains can be deduced: First of all, the present predominance of "thinking in domains" has to be judged critically. Basically, domains are no more than the values of an additional attribute attached to a DSM. By admitting this idea of the equality of domains and attributes, the second idea comes quasi along the way: As one attribute can have several values, an element can also belong to several domains. Regarding the current use of domains in the SCM methodology, the difference of the attribute "domain" in comparison to a normal attribute lies in the special role of this attribute. By assigning elements to different domains, these elements will be interpreted differently in the five steps of the SCM methodology. Thus, allowing multiple domains for one element offers a greater flexibility in analysing the system und interpreting the system structure.

By the following simple example the main idea will be explained. The system contains elements of the two domains *person* and *document*. A person *reports* to another person and can *write* a document. Elements of the domain *person* have the attribute *gender* A1 which can be either *female* or *male*. Documents have a certain *folder location* (attribute A2). In the upper part of Figure 1 the scheme for this system is shown and exemplarily filled with two persons (Bob and Eve) and two documents (protocol and bill of material) is shown.

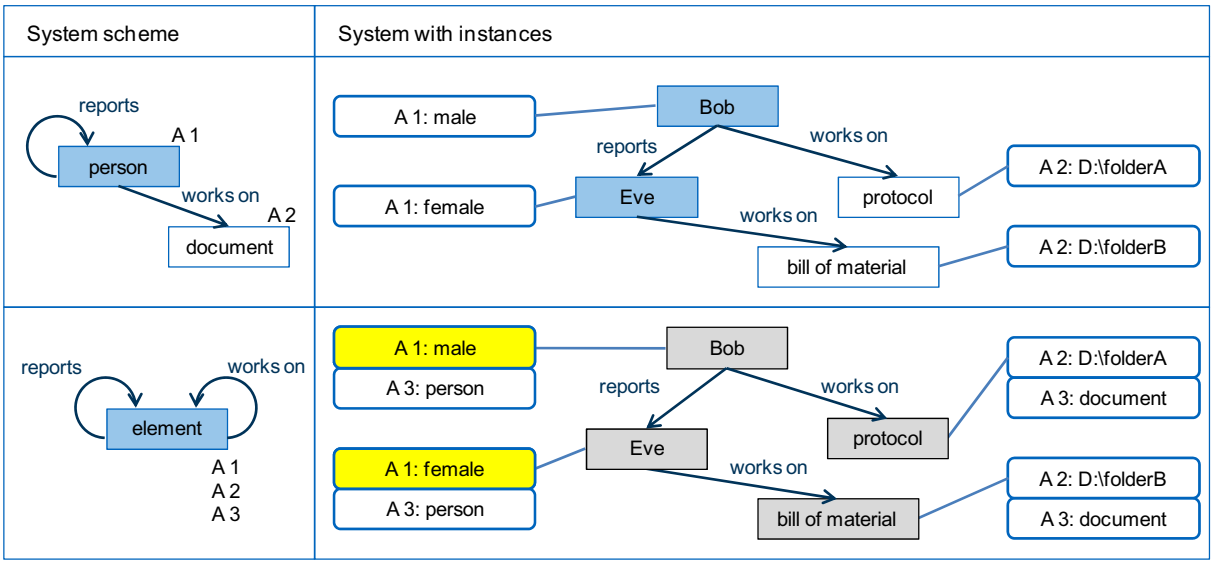


Figure 1. Interpreting domains (upper part: present way, lower part: domains as attributes)

Interpreting the domains as an additional attribute A3 with possible values *person* or *document*, the system changes like shown in the lower part of Figure 1. The former elements of the domains person and document change into a kind of "neutral format". This new interpretation offers the possibility to filter the domains either by attribute A3, or alternatively by attribute A1.

Two main restrictions of this alternative approach can be pointed out by this example. First of all, the interpretation of domains as attributes can lead to false relations. An element that has the attribute A3 = person cannot work on another element that has also the attribute A3 = person, as this would be senseless. Secondly, an element that has the attribute A3 = person cannot have the attribute A2, as people are not stored in a certain folder location. This means, the flexible handling of domains as attributes only works if steadily bearing in mind the respective constraints between 1) attributes and relations and 2) attributes and attributes.

To overcome these restrictions, a combination of the actually used "thinking in domains" with the flexible handling of domains in the appropriate steps of the SCM methodology is proposed. The aim is to keep the present advantages while simultaneously gaining from the new opportunities. The existing advantages of the present "thinking in domains" and the existing limitations concerning the modelling, the information acquisition, the analysis, and the visualisation of complex systems will be discussed in the following section.

3.2 Applying flexible domain handling in SCM methodology

System definition and information acquisition

The main benefit of "thinking in domains" concerning system definition and information acquisition lies in making systems more easily understandable for the people working with the model. Dividing a system into subsets (domains) and relevant dependency types between these domains can be visualised clearly and is easily understandable. Confusions as shown in the example above do not occur. Also, for the later information acquisition it is absolutely necessary that each member of a workshop has a common understanding of the respective relations and domains. Allowing more domains for one element and therewith making the system more difficult to understand for the user, could lead to confusion and poorly acquired information. For this reason, eliminating "thinking in domains" in these first two steps of the SCM methodology would be contra productive.

However, the basis for the needed flexibility of the domains in the following steps can be achieved by additionally acquiring the information about the attributes. The selection of relevant attributes that have to be collected depends on the later intended analysis of the system. Like the present definition of domains, this will still be one of the most difficult tasks in system definition.

Deduction of indirect dependencies

The main advantage in overcoming the present "thinking in domains" lies in the steps of deduction of indirect dependencies and system analysis. Here, the new flexibility can enhance the applicable analysis on the system. More indirect dependencies can be deduced by interpreting a single domain as two flexible domains. This offers alternative insights into the system's structure as the deduction of indirect dependencies can be computed with a finer granularity.

Structure analysis

Additionally, the analysis of the complex system can be applied more differentiatedly. In the present, well established matrix-based and graph-based methods are used for analysing complex systems (clustering, tearing, calculation of active sum, criticality, and so forth; for an overview see (Lindemann et al., 2009)). All these methods base on the chosen domains. These methods can now be used with a higher flexibility by interpreting not only the "normal" domains but also the flexible domains. Therewith, deeper insights into the structure of a system can be given.

4. SYSVIZ – A TOOL FOR FLEXIBLY MODELLING AND VISUALISING COMPLEX SYSTEMS

The prototypical tool SysViz (www.sysviz.org) was developed to investigate the presented idea and make the results visually available. This tool enables the flexible modelling and visualisation of complex systems. SysViz is designed as web-based java application and can be freely accessed via a standard web browser.

Figure 2 shows as an example the system presented above with 6 persons and 4 documents visualised in SysViz. The elements are arranged automatically according to their domain dependencies. All nodes belonging to the same domain (having the same value of a certain attribute) are arranged in a circle, and the distinct domain circles are spatially separated on the canvas.

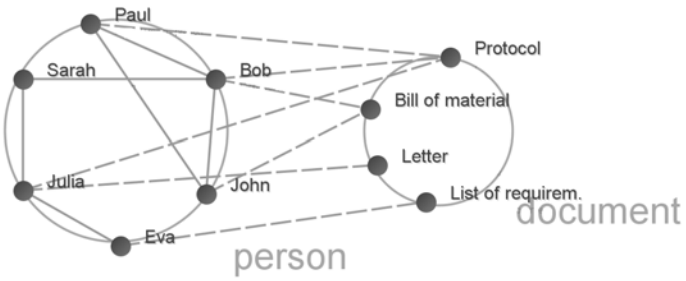


Figure 2. Visualising the modelled system with two domains

By applying the alternative view on the system, the elements so far ordered into two circles are visualised in three circles according to the new ordering (Figure 3). The members of the "new domains" *female* and *male* are now arranged in two separated circles.

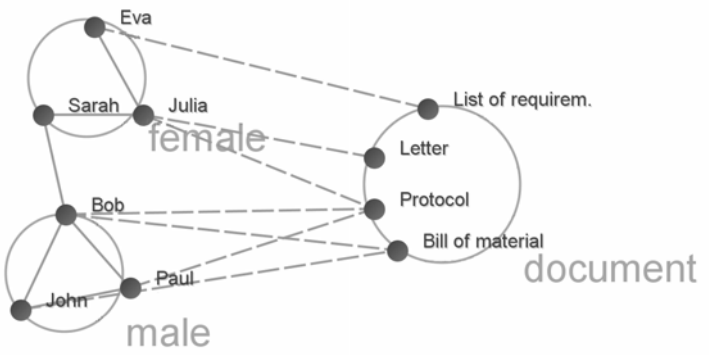


Figure 3. Visualising the modelled system with the flexible domains

The main difference between the two figures lies in the deeper insight into the systems structure via the second figure. Here, the two "new domains" *female* and *male* are visualised as spatially separated circles and as there is only one relation between these two circles, it becomes clearly visible that Sarah is the only female person in contact with one of the male persons.

5. CONCLUSION AND OUTLOOK

The approach presented in this paper proposes a new way of modelling complex systems in a more flexible way. The currently applied breakdown of complex systems is questioned and an approach to model domains similar to attributes in a DSM, thus offering more flexibility by analysing and visualising the complex system is shown. The developed tool SysViz is used to get a first idea of this approach and first tests showed promising results. Future work will cover research about the consequences of this approach, especially concerning flexible information acquisition and the resulting opportunities for the analysis of a system in combination with the necessary visualisation techniques. Also the scalability of this approach will have to be evaluated when applying it to large-scale systems

ACKNOWLEDGEMENT

The authors thank Tim Dutz for his helpful input and the implementation of the tool SysViz within his master thesis.

REFERENCES

Biedermann, W., Maurer, M., and Lindemann, U. (2007), The Multiple-Domain-Approach and Cost Attributes. In U. Lindemann, et al. (Eds.), *9th International DSM Conference*. (pp. 287-95). Aachen: Shaker.

Braun, T. and Deubzer, F. (2007), New Variant Management Using Multiple-Domain Mapping. In U. Lindemann et al. (Eds.), *9th International DSM Conference*. (pp. 363-72). Aachen: Shaker.

Browning, T. R. (2001), Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions, *IEEE Transactions on Engineering Management,* Vol. 48, No. 3, pp. 292-306.

Corcho, O., Fernández-López, M., and Gómez-Pérez, A. (2003), Methodologies, Tools and Languages for Building Ontologies. Where Is Their Meeting Point? In *Data & Knowledge Engineering,* Vol. 46, No. 1, pp. 41-64.

Danilovic, M. (2001), Supplier Integration in Product Development. In *10th International Annual IPSERA Conference* (Jönköping, Sweden), pp. 253-65.

Eppinger, Steven D. (2009), Integrating the Product, Process, and Organization Views of Complex System Development. In Matthias Kreimeyer, et al. (Eds.), *11th International DSM Conference*. Munich: Hanser.

Lindemann, U., Maurer, M., and Braun, T. (2009), *Structural Complexity Management - An Approach for the Field of Product Design*. Berlin: Springer.

Steward, D. V. (1981), The Design Structure System: A Method for Managing the Design of Complex Systems. In *IEEE Transactions on Engineering Management,* Vol. 28, No. 3, pp. 71-74.

Syldatke, T., et al. (2008), Managing Dependencies in the Product Development Process with Semantic Technologies. In M. Kreimeyer, U. Lindemann, and M. Danilovic (Eds.), *10th International DSM Conference*. (pp. 105-14). Munich: Hanser.

Contact: Andreas Kohn
Technische Universität München
Institute of Product Development
Boltzmannstraße 15
85748 Garching
Germany
Phone: +49 89 28915153
Fax: +49 89 289 15144
e-mail: andreas.kohn@pe.mw.tum.de

Approach towards a More Flexible Handling of Domains in Complex Systems

Andreas Kohn
Udo Lindemann

Technische Universität München

MANAGING COMPLEXITY
BY MODELLING DEPENDENCIES

Index

Introduction

Problem description

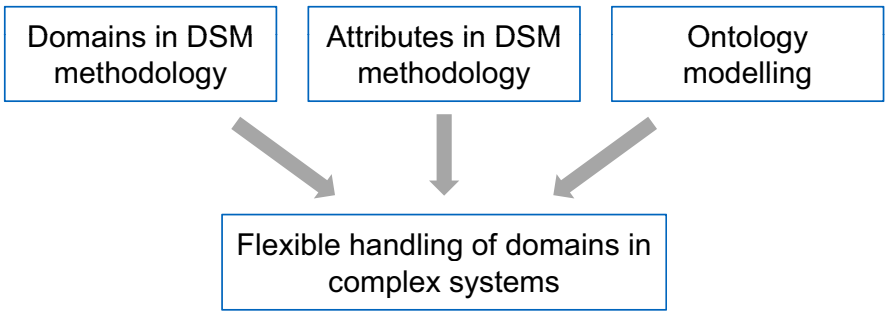
- DSM, DMM and MDM approaches group elements of a complex system into so-called domains
- Algorithms for analysing and interpreting the system structure depend on the chosen domains
- Limitation due to the domain-oriented classification and the present “thinking-in-domains”

Approach

- Increasing the flexibility of handling domains in complex systems
- Combining domain modelling and attribute modelling
- Integrating flexible handling of domains in current complexity management approaches

Background of this approach

- Use of domains for the handling of complex systems (DSM, DMM, MDM)
- Use of attributes in DSM methodology
- Similarities of modelling systems in the field of ontology development



Background of this approach

Domains

- Use of various domains in many industrial and scientific applications
- Domains are chosen according to either the later intended analyses or the available information

Attributes

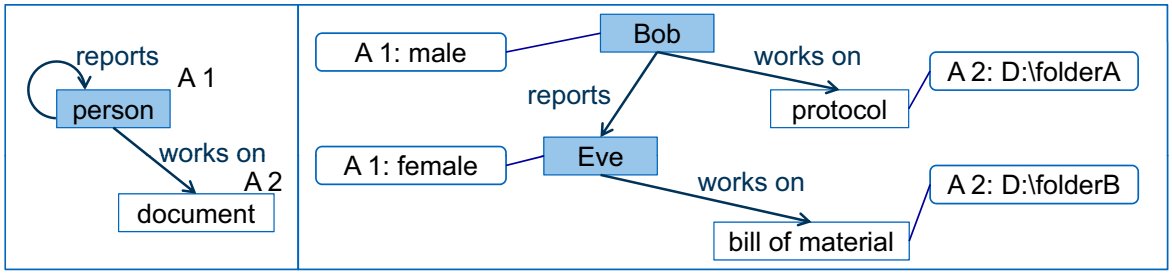
- Attributes are used for adding additional information to the elements of a system

Ontology modelling

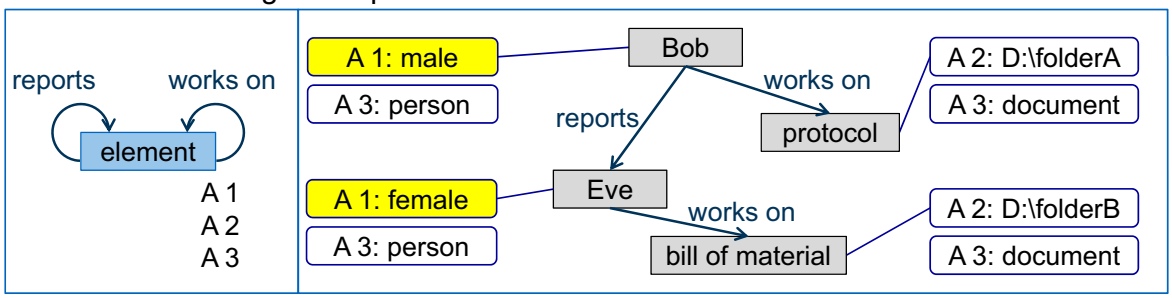
- Ontology languages offer a huge variety of artefacts for modelling system components and relations
- Similarities to the modelling of Multiple-Domain Matrices
- Advantages by overcoming the strictly hierarchical structuring of the modelled systems (e.g. by multi-inheritance of concepts)

Flexible modelling of domains

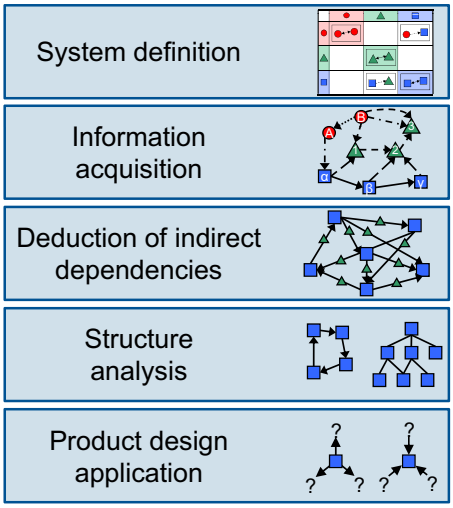
Current modelling



Flexible modelling – interpretation of the domain as additional attribute



Application in structural complexity methodology (SCM)



- "Thinking in domains" useful for the mutual understanding of the model during system definition
- One has to take in mind the required additional information during information acquisition
- Higher granularity in the later deduction of indirect dependencies
- Use current methods for analysing and visualising the flexible domains
- Improve product design application due to deeper insights into the system structure

Deduction of indirect dependencies

Flexible domains can be used for the more detailed deduction of indirect dependencies

Domains *person* and *document*

	person	**document**
person	reports	works on
document		

Domains *female, male* and *document*

	female	**male**	**document**
female	reports	reports	works on
male	reports	reports	works on
document			

Analysis of the structure

Current matrix-based and graph-based algorithms can be applied with a finer granularity

Domains *person* and *document*

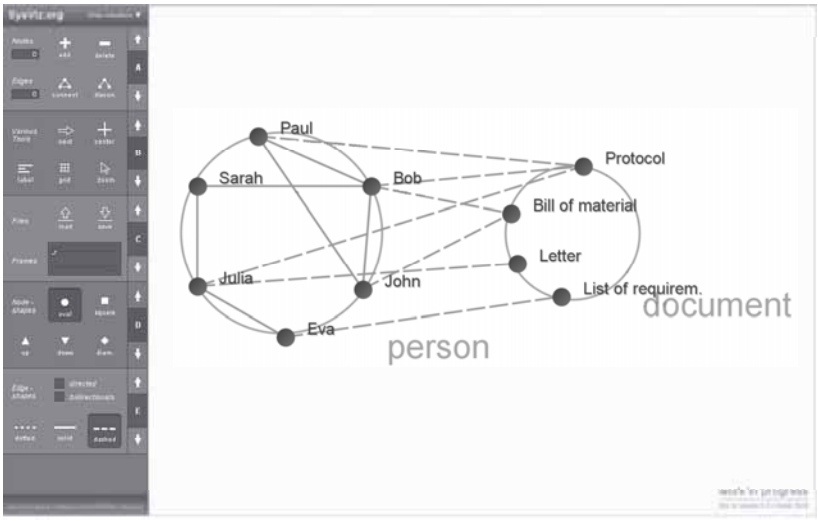
	person	**document**
person	reports	works on
document		

Domains *female, male* and *document*

	female	**male**	**document**
female	reports	reports	works on
male	reports	reports	works on
document			

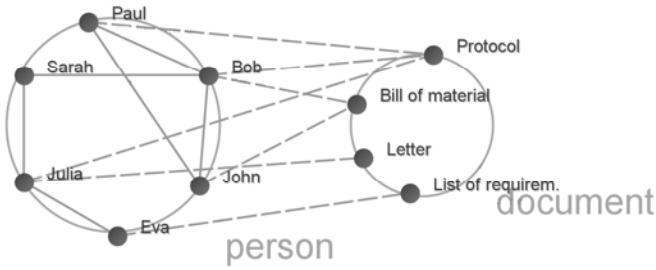
Application for visualising complex systems: SysViz

- Flexible modeling and visualization of domains in a complex systems
- Visualizing the elements belonging to a domain by the arrangement in spatially separated circles

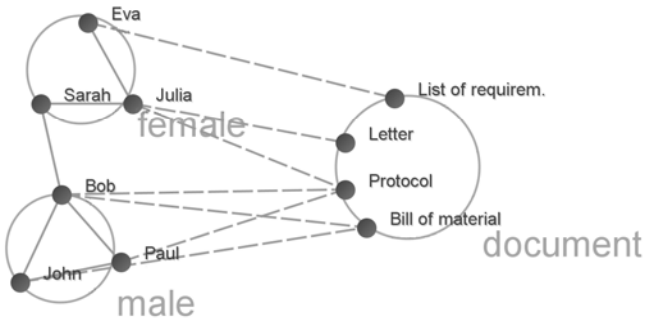


SysViz: Flexible visualisation of multiple domain systems

- Visualising domains person and document



- Visualising domains female, male and document



→ Deeper insight into the system structure.

Summary and Outlook

Summary

- Approach towards a more flexible modelling of domains by interpreting the domains as attributes
- Flexibility in analysing and visualising complex systems
- Development of a tool for modelling and visually analysing systems prototypically implemented

Outlook

- Further development of the tool to enable analysis of a system considering flexible domains
- Application and evaluation of this approach in current projects
- Evaluation of the scalability of this approach in large-scale systems