

A PRAGMATIC APPROACH TOWARDS LEVERAGING EMPLOYEE COMPETENCES BY USE OF SEMANTIC WEB TECHNOLOGIES

Ocker, Felix; Vogel-Heuser, Birgit

Technical University of Munich

ABSTRACT

Global competition in combination with the increasing specialization of labor requires organizations to leverage their employees' competences. The approach presented in this paper empowers organizations to do so in two ways. First, we give employers a lean tool to allocate their employees to current and upcoming projects and make informed decisions whether they should take on new projects. Secondly, we provide the means to identify potential areas for innovation by identifying blind spots of technology transfer. The approach presented relies on semantic web technologies, i.e. an ontology built in OWL and SPARQL queries. To increase usability, we realized a user interface based on a semi-formalized spreadsheet and a python script for the transformation.

Keywords: Employee Allocation, Creativity, Knowledge management, Innovation, Ontologies

Contact:

Ocker, Felix
Technical University of Munich
Institute of Automation and Information Systems
Germany
felix.ocker@tum.de

Cite this article: Ocker, F., Vogel-Heuser, B. (2019) 'A Pragmatic Approach Towards Leveraging Employee Competences by Use of Semantic Web Technologies', in *Proceedings of the 22nd International Conference on Engineering Design (ICED19)*, Delft, The Netherlands, 5-8 August 2019. DOI:10.1017/dsi.2019.110

1 MOTIVATION

The ever increasing division of labor we experience today provides our society with unprecedented productivity (Malone *et al.*, 2011). This specialization causes knowledge to become increasingly important for most organizations (Chaston, 2012). At the same time, highly specialized employees are oftentimes hard to find and costly. To stay competitive, organizations ranging from companies to universities need to exploit the competences of the highly skilled employees they have. This goes especially for knowledge intensive and multi-disciplinary domains such as engineering and production. Another trend that strongly influences the way organizations work nowadays is globalization. Tough world-wide competition requires organizations to accelerate design processes and to innovate quicker. Additionally, globalization also intensifies the war for talents. One way to cope with these challenges is to establish an organizational culture that fosters innovation. This can be achieved, together with other principles, by increasing collaboration and confrontation among the persons involved (Volpentesta and Felicetti, 2011).

The approach presented in this paper empowers organizations to make the most of their competences by targeting two research questions. Firstly, we give employers a lean tool to allocate their employees to current and upcoming projects - "*Q1: In what way should managers allocate employees to projects?*". Secondly, we provide the means to leverage potential areas for innovation by identifying blind spots of technology transfer among different domains - "*Q2: Which employees should exchange ideas to foster innovation?*". This second part is especially valuable for domains in which diverse disciplines collaborate. Semantic Web Technologies (SWT) are well suited to represent knowledge. However, usability is reduced by the technologies' complexity and degree of abstraction. We present an approach that relies on an ontology built in Web Ontology Language (OWL) combined with custom SPARQL Protocol And RDF Query Language (SPARQL) queries and increase usability by providing a simple user interface using spreadsheets.

2 STATE OF THE ART

This section presents related work from the domains of knowledge and innovation management and gives a short introduction to SWT.

Organizations from research as well as those from industry aspire to create novel ideas. There are various factors influencing the success of design teams, though. These include characteristics such as team size and interaction frequency (McComb *et al.*, 2016), but also direct and indirect learning, which strongly influence transactive memory (Singh *et al.*, 2013), as well as team familiarity, which is crucial for social learning (Singh *et al.*, 2012). These approaches do not focus on a competence-based composition of teams (*Q1*) or the support of innovation by combining previously disjoint ideas (*Q2*), though. To describe competences, the HR Open Standards Consortium (HR-OSC, 2018) distinguishes person competences from position competences and defines them as follows:

- "A specified person competency is a competency within the context of a personal record (profile, appraisal, assessment) assessed or asserted at specified level of proficiency."
- "A qualified position competency has a specified required and/or desired level of proficiency and has an explicit or implicit level of importance (weight) among sibling competencies associated with a position."

Creating a matrix is a straight-forward way of mapping the competences required by projects to the competences provided by employees to ideally allocate human resources. This mapping is known as *skills matrix* or *competence matrix*. Melosi *et al.* (2018) describe a competence mapping for a company's leather cutting department. They identify the relevant processes and divide them into various elementary tasks, thus describing required abilities. In parallel, they assess the employees' craftsman skills to check if they are appropriate for the assigned tasks. Melosi *et al.* (2018) stress the financial benefits for the company as well as the social aspect of better employment and higher satisfaction of workers, who can positively express their abilities.

Draganidis *et al.* (2006) identify two underlying needs that result in the application of competence management approaches. For one, competence management approaches create a basis to make informed decisions in current and future personnel selection. Additionally, they help to minimize the gap between available competences and those required by a project or a job role. Draganidis *et al.* (2006) also found, that SWT are well suited for competence management.

SWT are summarized in the Semantic Web Stack (W3C, 2007). They are based on ontologies, which Gruber (1993) defines as “an explicit specification of a conceptualization”. These graph-based knowledge bases can be expressed using the Web Ontology Language (OWL) (W3C, 2012) and information can be extracted by use of SPARQL queries. To increase understandability and reusability of ontologies, competency questions can be used as suggested by Noy and McGuinness (2001). Further details on how to design an ontology can, e.g., be found in the research of Arp *et al.* (2015).

Some competence management approaches have already been developed that rely on SWT. Draganidis *et al.* (2008) presents a competence ontology that aims to serve two purposes. The ontology is supposed to enable current and future personnel selection as well as individual and group development plans. The authors identify best fit employees for projects based on a comparison of the competencies an employee has and the ones required by a project. A skill gap report is created that can be used to upskill employees if they are not sufficiently qualified. However, the employees’ capacity is not taken into account. To support scientific research in new domains, Rogushina and Gladun (2012) map research domain ontologies with information available on the web. They focus on the integration of competence information, including proficiency, from various sources. These include publications as well as social network information. They store the collected information in two ontologies, one dedicated to the competencies of specialists and one describing the problems those specialists solve. Khobreh *et al.* (2016) have the goal to bridge the gap between requirements of labor markets and education systems but also assess qualification levels of job applicants. They present a framework for building a job-knowledge ontology describing knowledge, skills, and abilities as well as the relations between these concepts. The paper also gives a detailed overview of existing ontologies from the job knowledge domain. The proficiency levels of competences included in such ontologies may be assessed by employees themselves. However, to increase objectivity, organizations should rely on more comparable approaches such as the one presented by Kukk *et al.* (2015). Gaeta *et al.* (2012) describe a theoretical methodology using SWT to improve assessment processes of employee competences. This might be combined with the approach if Volpentesta and Felicetti (2011), which supports the creation of new knowledge tailored explicitly to research communities. To do so, they focus on extracting and classifying competences from research papers. Concluding, the approaches presented to address *Q1* using SWT still have deficits and do not consider *Q2*.

Morphological analysis is a creativity technique well known in the engineering domain that can be applied to support innovation (*Q2*). Originally presented by Zwicky (1967), many more refinements followed, e.g., by Ritchey (2002). As already stated by Zwicky (1967), morphological methods are applicable to human problems in general. The morphological box, e.g., can be used to map solution features, which should be independent from each other, to a problem. New ideas for a product can be generated by combining several solution features.

3 CONCEPT

This section introduces the concept developed, discusses the assumptions we make, describes the information our ontology captures, and delineates how we leverage this information. To answer the two research questions, we need formalized knowledge from both the managers and employees. Only then can we provide advice regarding the allocation of employees to projects and targeted exchange of ideas among employees. To ease the formalization of the knowledge required, we developed a semi-formalized spreadsheet that is transformed into an ontology that can be queried. Figure 1 summarizes the concept graphically and indicates the workflow.

We distinguish two underlying scenarios when a potential new project comes up. On the one hand, the manager should start the new project if his organization has sufficiently skilled and available employees. On the other hand, there are three potential courses of action if the institution does not have appropriately skilled employees with sufficient capacity. First, the manager can decline to take on the new project. Secondly, the manager can try to gather additional resources or upskill his current employees (Draganidis *et al.*, 2008) to realize the project. However, newly hired employees may reduce the institution’s overall performance at first, due to adjustment needs. Thirdly, managers may decide to act opportunistically by accepting a new project despite not having sufficient capacity. This will most probably result in non-satisfactory performance or the need to drop another project, typically with a lower priority. When making such a decision, we encourage managers to keep work ethics in mind and to consider potential

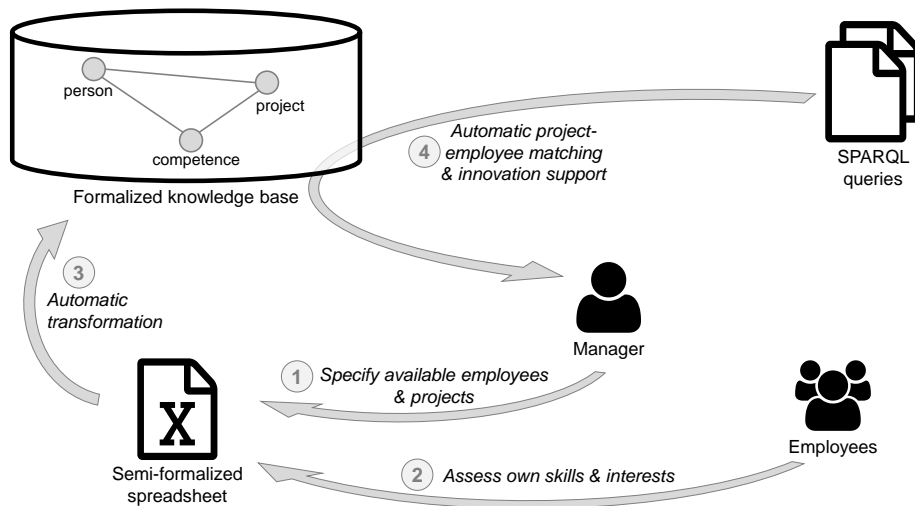


Figure 1. Overview of the concept proposed

repercussions in the form of losing a client’s trust. On a general note, we advise managers to not continuously try to utilize 100 % of their employees capacity because queuing theory shows that this likely leads to queues and eventually failure regarding project completion.

3.1 Axioms and restrictions

Within the following, we make several design choices regarding the ontology and the queries that are based on various axioms. To avoid misconceptions, we delineate these underlying assumptions and the resulting restrictions. On an implementation level, we want to design the ontology to be minimalist. This means we only include information that is relevant for *Q1* and *Q2*. This has two advantages. Apart from reducing the computational effort, employees and managers are not unnecessarily burdened with specifying knowledge that is not even put to use. Also, we assume that every project can be subdivided into “atomic” subprojects, whereof each requires exactly one competence. Subsequently, exactly one employee is assigned to each atomic project. This assumption can be lifted as demonstrated by Dragani-dis *et al.* (2008). To model that several employees work on the same project, the project may be split up into several sub projects. We suggest to do so only if the project’s effort exceeds a certain threshold to avoid the fragmentation of projects, which would lead to inefficient project execution. Another strong assumption is that employees work continuously on projects. Even though this may not hold for short periods of time, we expect this assumption to be valid for long-term planning. From a psychological perspective, employees are expected to be motivated more if they are interested in the projects they are assigned to, cp., e.g., (Melosi *et al.*, 2018). Additionally, the success rate for cooperation among individuals strongly depends on individuals and their interpersonal relations. Another assumption is that innovation can be fostered by transferring ideas from one domain to another. One famous example for this concept is biomimicry. In consequence, employees from different disciplines should be encouraged to cooperate so that they will generate new ideas. This kind of exchange also happens naturally whenever two persons from different domains collaborate.

3.2 Knowledge base

Q1 and *Q2* can be considered the ontology’s competency questions. Throughout the ontology’s design, we made several design choices. Such choices were often driven by the intention to keep the information to be specified by managers and employees to a minimum, but to still be able to reliably answer *Q1* and *Q2*. We included the necessary universals to model the following information. *Organizations* work on *projects* and employ *persons*. To successfully complete projects, *competences* of a certain *proficiency* are required by *projects*. The employees not only *provide* such *competences*, but they are usually also *interested in* them. To answer *Q1*, both the employee’s *capacity* in hours per week and the project’s effort in hours should be considered. Projects are further specified by their *priority*, *start date*, and *due date*. To describe *organization* structures in detail, the *part* relation can be used. Analogously, *projects*

can be divided into sub-projects. While one employee may be *assigned to* several *projects*, an atomic *project* always has exactly one *person* associated to it. This design choice greatly facilitates querying the knowledge base and pays respect to the axiom described earlier. Relations among employees are manifold and oftentimes complicated. Within the scope of this paper, we only model if persons *like* each other, currently *cooperate with* each other, and have *cooperated with* each other in earlier projects. These universals and relations are commonly accepted and can be found in other knowledge bases of the competence management domain, cp., e.g., Section 2. Hence, we generally support compatibility with existing ontologies. Regarding competences, we deem it important to distinguish between *domain knowledge*, e.g., *mechanical engineering*, *organizational skills*, e.g., *project management*, and *technology skills*, e.g., *semantic web technologies*. Hereby, the application of *technology skills* to various application domains is assisted by *organizational skills*. The ontology's *TBox* is shown in Figure 2.

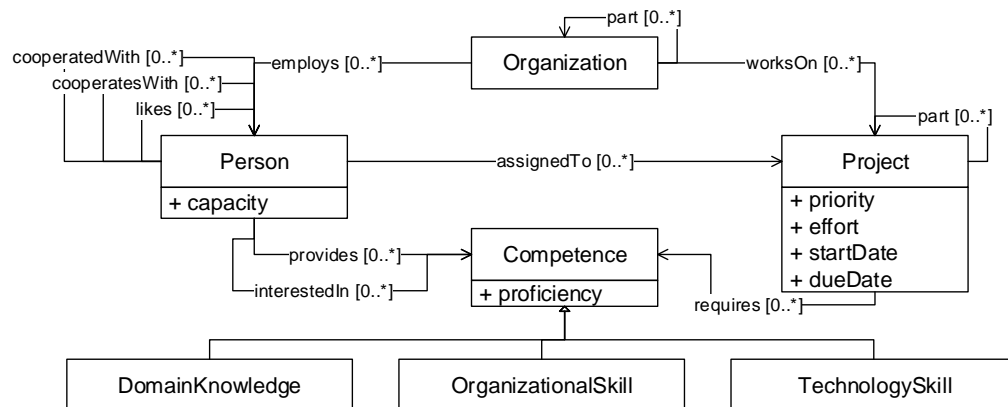


Figure 2. *TBox* of the ontology developed

3.3 Ontology creation

SWT may be a field that is out of scope for most managers and employees. We acknowledge this and provide a pragmatic approach to creating the *ABox* of the ontology using a semi-formalized spreadsheet. An overview of the workflow is given in Figure 3.

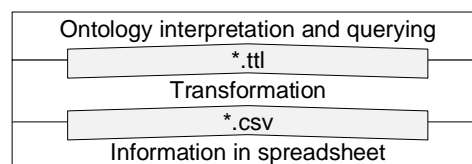


Figure 3. Transformation from the spreadsheet to an ontology

The spreadsheet consists of several types of sheets to easily specify prefixes, taxonomies, as well as the objects and relations of the *TBox* and the *ABox*. We choose a representation in the form of a matrix to facilitate the specification of relations that are used often, i.a. the *cooperatesWith* relation. This approach is generically applicable. For the ontology presented in this paper, we specified the *TBox* with its universals and properties as well as meta information such as prefixes. Such templates can also be reused, e.g., for standardizing the kind of information collected within one organization.

Based on this spreadsheet, users only need to add information for the *ABox*, i.e. the instance level, which includes instances for the classes defined in Figure 2, the relations among them, and object properties as well as datatype properties. As a feasibility study, we use Microsoft Excel and a simple macro for exporting all sheets as individual csv files. Using csv files as an intermediate format has the advantage of being less dependent on commercial software, which allows organizations to use, e.g., LibreOffice Calc instead of Microsoft Excel. Subsequently, we transform the csv files into an ontology using a python script. We choose the turtle serialization as an output format due to its good readability.

In a final step, the ontology can be interpreted and queried using reasoners such as stardog and it can be made accessible to humans, e.g., using the ontology editor Protégé.

3.4 Leveraging the ontology created

This section describes how we leverage the ontology developed to answer $Q1$ and $Q2$. The implementation using SPARQL queries is presented in the appendix, cp. Listing 2 and Listing 3.

To answer $Q1$ (*In what way should managers allocate employees to projects?*) we compare the competences required by the new projects with those provided by employees. Apart from checking which employees have the required competences, we also check whether they have sufficient proficiency and interest in the area of expertise required by the project. Additionally, we check if those employees, who have appropriate competences and interests, also have the capacity to realize the project in question, cp. Equation (1).

$$capacity_{available} \geq capacity_{required} \quad (1)$$

We calculate the available capacity as the employee's weekly capacity multiplied with the new project's duration, cp. Equation (2). This is to be compared with the required capacity, which consists of the efforts of ongoing projects the employee is assigned to and the total effort of the new project, cp. Equation (3).

$$capacity_{available} = capacity_{weekly} \times duration_{new-project} \quad (2)$$

$$capacity_{required} = \sum_i effort_{weekly_i} \times overlap_i + effort_{new-project} \quad (3)$$

Calculating the utilization of employee capacity through ongoing projects is slightly more complicated because subtracting the entire effort of ongoing projects would return false negatives. Instead, we consider only the overlap of the ongoing project (op) and the new project (np). The four kinds of overlaps are depicted in Figure 4.

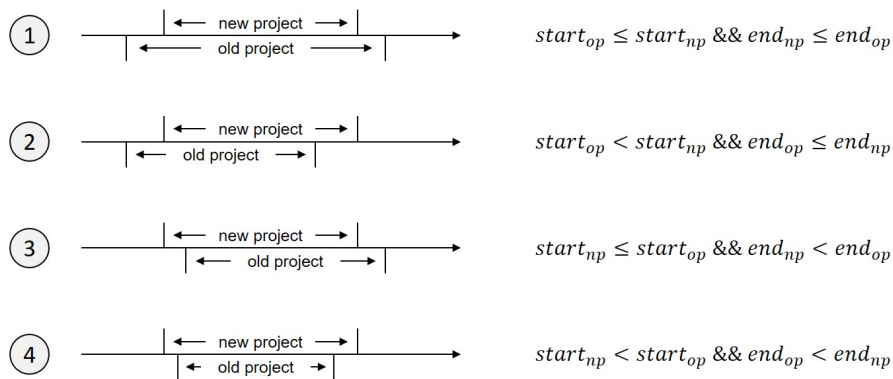


Figure 4. Possible overlaps of projects

Equation (4) shows how we calculate the duration of the overlap of an existing and a new project according to cases one through four depicted in Figure 4.

$$overlap = \begin{cases} end_{np} - start_{np} & \text{case 1} \\ end_{op} - start_{np} & \text{case 2} \\ end_{np} - start_{op} & \text{case 3} \\ end_{op} - start_{op} & \text{case 4} \end{cases} \quad (4)$$

Since we are only interested in those projects that overlap with the new project, we have to double check that there actually is an overlap. This constraint is summarized in Equation (5).

$$start_{np} \leq start_{op} \wedge end_{np} \geq start_{op} \quad \vee \quad start_{np} \leq end_{op} \wedge end_{np} \geq end_{op} \quad (5)$$

Finally, *Q1* sorts all potential new projects according to their priority to support managers in choosing from several new projects.

The second query *Q2* uses the methodology of a morphological box to combine previously unconnected areas of competences, analogously to the creation of innovative product ideas. This can be expected to result in innovative technologies and approaches, thus answering *Q2* (*Which employees should exchange ideas to foster innovation?*). First, the query identifies tuples of non-organizational competence classes the organization's employees have. Within this paper, we neglect organizational competences because they do not contribute to technical innovations as directly as technical competences. In the second step, all competence tuples are removed that are provided by the same person. Analogously, all tuples are removed that two employees have who cooperate or have cooperated in earlier projects. We assume that the exchange of ideas happens naturally in both cases. On this basis, managers can identify employees who have the competences for which there is no knowledge exchange. These employees can be asked to proactively discuss their domains of expertise to encourage the transfer of ideas.

4 FEASIBILITY STUDY

This section describes the use case chosen and gives insights into our modular implementation as well as the application of the queries developed to answer the questions *Q1* and *Q2*.

4.1 Use case

The fictional research institute *Innovation and Engineering Laboratory (IEL)* has six employees who currently work on three projects A through C depending on their competences. The current mapping between employees and subprojects is depicted in Figure 5.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	matrix	abox													
2															
3			assignedTo	Project											
4			Person		ProjectA	ProjectA1	ProjectA2	ProjectB	ProjectB1	ProjectB2	ProjectC	ProjectC1	ProjectC2	ProjectC3	newProject
5				Camile					x						
6				Felix	x										
7				Gennadiy		x							x		
8				Jane					x						
9				Mia							x				
10				Raoul									x		
11															

Figure 5. Mapping of employees and projects

When the organization receives a request to work on an additional project *newProject*, the manager needs to make an informed decision whether the organization should take the project. Using the approach presented in this paper, the manager and the employees specify competences provided and required, relations between employees, and project details as described in Figure 2 in the semi-formalized Excel file. Exemplary employee competences include, e.g., *semantic web technologies*, *model driven development*, but also knowledge of the application domains such as *electrical engineering* or *mechanical engineering*. When analyzing the new project, the manager finds that it requires *model driven development* competences with a proficiency of at least one. The customer specified the start time and due date as the first of March 2020 and the 31st of December 2020, respectively. Finally, the manager assesses the project's effort as 400 hours and assigns a priority of 2 to the potential new project.

4.2 Querying the knowledge base

The *IEL*'s manager uses the queries presented to make an informed decision whether he should accept the new project offer and which employee should work on it. All three employees *Felix*, *Gennadiy*, and *Mia* provide the competence *model driven design* with a proficiency of two. However, *Gennadiy* is already assigned to the projects *A2* and *C3*, cp. Figure 5, and does not have sufficient capacity to realize the new project. Even though *Mia* also has appropriate competences for the new project and has sufficient capacity, she has little interest in *model driven development*. Hence, the first query, cp. Listing 2, returns only *Felix* as an employee who should work on the new project. Since *Felix* has sufficient capacity, the manager can take on the new project.

The second query we developed addresses *Q2*. It identifies competences among which there is no exchange, cp. Listing 3. Based on this, employees can be encouraged to exchange ideas. For this paper's use case, the second query reveals that the employee *Felix* should exchange ideas with both *Jane* and *Raoul*. This way, *Felix*, *Jane*, and *Raoul* can discuss in what way semantic web technologies and inconsistency management can be applied to mechanical engineering, modularization, and visualization.

5 SUMMARY AND OUTLOOK

In this paper, we present a pragmatic approach to allocate employees to projects (*Q1*) and to support the targeted exchange of ideas among employees to foster innovation (*Q2*). To do so, we provide a minimalist ontology that represents organizations, their projects, their employees, and the required as well as available competences. To increase usability, we implemented an interface based on a semi-formalized spreadsheet that is transformed into an ontology. Tailored to this ontology, two SPARQL queries provide guidance regarding employee allocation and innovation support. This approach can be expected to be especially useful in the knowledge intensive context of engineering and production.

When applying this approach, organizations should consider local regulations regarding privacy protection. Also, the effort estimation of projects is critical. To make the approach more applicable, it should be combined with, e.g., the work presented by [Draganidis et al. \(2008\)](#). This includes considering the combination of several required competences of various importance for each atomic project. Taking the sunk cost fallacy into account, it might be sensible for an organization to re-assign employees to other projects in some cases. This should be combined with an approach to globally optimize the organization's goals, which typically include profitability. In that case, each project's individual profitability should be included in the ontology and could be used for more advanced optimization using linear programming, which may also be applied for finding a trade-off between employee workload and innovation potentials. Another factor that should be considered when allocating employees to projects is the compatibility of the team. This is in principle supported by the ontology, as it represents interpersonal relations via the relation "likes". However, this single relation may be insufficient to model complex social behavior. Future research is required to formally describe tacit factors such as personality, mood, and shifting responsibilities.

Future work should focus on the integration of the various efforts made in the domain of competence management to formalize job knowledge. This includes using existing knowledge bases, cp. Section 2, as well as methodologies to automatically create competence sets, e.g., from publications and projects ([Volpentesta and Felicetti, 2011](#)). Additionally, explicit knowledge should be taken into account, too, e.g., in the form of best practice guidelines and checklists. Finally, organizations should continuously update the knowledge base. Exemplary, the successful completion of a project should be appreciated with an increased assessment of the employees' competence proficiency.

REFERENCES

- Arp, R., Smith, B., and Spear, A. D., (2015), *Building ontologies with basic formal ontology*, MIT Press.
- Chaston, I. (2012), "Strategy for Sustainable Competitive Advantage", Taylor & Francis Group, New York. <http://doi.org/10.1080/10599230903323698>
- Draganidis, F., and Mentzas, G. (2006). "Competency based management: a review of systems and approaches". *Information management & computer security*, Vol. 14 No. 1, pp. 51–64. <https://doi.org/10.1108/09685220610648373>
- Draganidis, F., Chamopoulou, P., and Mentzas, G. (2008), "A semantic web architecture for integrating competence management and learning paths", *Journal of knowledge management*, Vol. 12 No. 6, pp. 121–136. <https://doi.org/10.1108/13673270810913667>
- Gaeta, M., Orciuoli, F., Fenza, G., Mangione, G. R., and Ritrovato, P. (2012), "A semantic approach for improving competence assessment in organizations", *International Conference on Advanced Learning Technologies*, IEEE, pp. 85–87. <http://doi.org/10.1109/ICALT.2012.168>
- Gruber, T. R. (1993), "A translation approach to portable ontology specifications", *Knowledge Acquisition*, Vol. 5 No. 2, pp. 199–220. <http://doi.org/10.1006/knac.1993.1008>
- HR Open Standards Consortium, Inc. (2018), *HR Open Standards*, online: <https://hropenstandards.org/>

- Khobreh, M., Ansari, F., Fathi, M., Vas, R., Mol, S. T., Berkers, H. A., and Varga, K. (2016), “An ontology-based approach for the semantic representation of job knowledge”, *IEEE Transactions on Emerging Topics in Computing*, Vol. 4 No. 3, pp. 462–473. <http://doi.org/10.1109/TETC.2015.2449662>
- Kukk, V., Umbleja, K., and Jaanus, M. (2015), “Two-dimensional knowledge model for learning control and competence mapping”. *International Workshop on Learning Technology for Education in Cloud*, Springer, Maribor, Slovenia, 2015, pp. 16–27. https://doi.org/10.1007/978-3-319-22629-3_2
- Malone, T. W., Laubacher, R., and Johns, T. (2011), “The Big Idea: The Age of Hyperspecialization”, *Harvard Business Review*, online: <https://hbr.org/2011/07/the-big-idea-the-age-of-hyperspecialization>
- McComb, C., Cagan, J., and Kotovsky, K. (2016), “Linking Properties of Design Problems to Optimal Team Characteristics”, *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. <https://doi.org/10.1115/DETC2016-59333>
- Melosi, F., Campana, G., and Cimatti, B. (2018), “Competences Mapping as a Tool to increase Sustainability of Manufacturing Enterprises”, *Procedia Manufacturing*, Vol. 21, pp. 806–813. <https://doi.org/10.1016/j.promfg.2018.02.187>
- Noy, N. F., and McGuinness, D. L. (2001), *Ontology development 101: A guide to creating your first ontology*.
- Ritchey, T. (2002), “General Morphological Analysis”, *Wicked Problems - Social Messes. Risk, Governance and Society*, Springer, Berlin, pp. 7–18. <http://doi.org/10.1007/978-3-642-19653-9>
- Rogushina, J., and Gladun, A. (2012), “Ontology-based competency analyses in new research domains”, *Journal of Computing and Information Technology*, Vol. 20 No. 4, pp. 277–291. <https://doi.org/10.2498/cit.1002034>
- Singh, V., Dong, A., and Gero, J. S. (2012), “Computational studies to understand the role of social learning in team familiarity and its effects on team performance”, *CoDesign*, Vol. 8 No. 1, pp. 25–41. <https://doi.org/10.1080/15710882.2011.633088>
- Singh, V., Dong, A., and Gero, J. S. (2013), “Social learning in design teams: The importance of direct and indirect communications”, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 27, pp. 167–182. <https://doi.org/10.1017/S0890060413000061>
- Volpentesta, A. P., and Felicetti, A. M. (2011), “Competence mapping through analysing research papers of a scientific community”, *Doctoral Conference on Computing, Electrical and Industrial Systems*, Springer, Berlin, Heidelberg, pp. 33–44. http://doi.org/10.1007/978-3-642-19170-1_4
- World Wide Web Consortium (2007), “Semantic Web Stack”. Online, last retrieved 2019-06-20: <https://www.w3.org/2007/03/layerCake.png>
- World Wide Web Consortium (2012), “OWL 2 Web Ontology Language”. Online, last retrieved 2019-06-20: <https://www.w3.org/TR/owl2-overview/>
- Zwicky, F. (1967), “The Morphological Approach to Discovery, Invention, Research and Construction”, In: Zwicky F., Wilson A.G. (eds), *New Methods of Thought and Procedure*, Springer, Berlin, pp. 273–297. http://doi.org/10.1007/978-3-642-87617-2_14

ACKNOWLEDGMENTS

We thank the German Research Foundation (DFG) for funding the project CRC 768.

APPENDIX

The appendix includes a listing for the prefixes used in the queries as well as both queries *Q1* and *Q2*.

Listing 1 Prefixes used for the following SPARQL queries

```

1 prefix : <http://innomgmt.com#>
2 prefix owl: <http://www.w3.org/2002/07/owl#>
3 prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
4 prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
5 prefix xsd: <http://www.w3.org/2001/XMLSchema#>

```

Listing 2 SPARQL query for allocating employees to new projects (Q1)

```

1 SELECT ?newproject ?priority ?competencetype ?availableemployee ?npeffort ?overlap ?capacity ?
   npduration ?capacityavailable WHERE {
2 # identify new project and its required competence
3 ?newproject a :project .
4 ?newproject :priority ?priority .
5 FILTER NOT EXISTS {
6   ?employee :assignedTo ?newproject .
7 }
8 FILTER NOT EXISTS {
9   ?newproject :part ?subproject .
10 }
11 ?newproject :requires ?rcompetence .

```

```

12  ?rcompetence a ?competencetype .
13  # identify available employee with appropriate competences
14  ?availableemployee a :person .
15  ?availableemployee :provides ?pcompetence .
16  ?pcompetence a ?competencetype .
17  ?rcompetence :proficiency ?rproficiency .
18  ?pcompetence :proficiency ?pproficiency .
19  FILTER ( ?pproficiency >= ?rproficiency ) .
20  # check interest fit
21  ?availableemployee :interestedIn ?icompetence .
22  ?icompetence a ?competencetype .
23  # calculate project overlap in weeks; ensure that there is an overlap
24  ?availableemployee :assignedTo ?currentproject .
25  ?currentproject :effort ?cpeffort .
26  ?currentproject :startDate ?cpstart .
27  ?currentproject :dueDate ?cpend .
28  ?newproject :effort ?npeffort .
29  ?newproject :startDate ?npstart .
30  ?newproject :dueDate ?npend .
31  FILTER ( ?npstart <= ?cpstart && ?npend >= ?cpstart || ?npstart <= ?cpend && ?npend >= ?cpend ) .
32  BIND ( IF( ?cpstart <= ?npstart, ?cpstart, ?npstart ) AS ?overlapstart ) .
33  BIND ( IF( ?cpend >= ?npend, ?cpend, ?npend ) AS ?overlapend ) .
34  BIND ( year( ?overlapend ) - year( ?overlapstart ) AS ?yeardelta ) .
35  BIND ( month( ?overlapend ) - month( ?overlapstart ) + 12 * ?yeardelta AS ?monthdelta ) .
36  BIND ( day( ?overlapend ) - day( ?overlapstart ) AS ?daydelta ) .
37  BIND ( 4.3 * ?monthdelta + ?daydelta / 7 AS ?overlap ) .
38  # calculate new duration of new project
39  BIND ( year( ?npend ) - year( ?npstart ) AS ?npyeardelta ) .
40  BIND ( month( ?npend ) - month( ?npstart ) + 12 * ?npyeardelta AS ?npmonthdelta ) .
41  BIND ( day( ?npend ) - day( ?npstart ) AS ?npdaydelta ) .
42  BIND ( 4.3 * ?npmonthdelta + ?npdaydelta / 7 AS ?npduration ) .
43  # check availability
44  ?availableemployee :capacity ?capacity .
45  # calculate work load from current projects during np as cpreleffort
46  BIND ( year( ?cpend ) - year( ?cpstart ) AS ?cpyeardelta ) .
47  BIND ( month( ?cpend ) - month( ?cpstart ) + 12 * ?cpyeardelta AS ?cpmonthdelta ) .
48  BIND ( day( ?cpend ) - day( ?cpstart ) AS ?cpdaydelta ) .
49  BIND ( 4.3 * ?cpmonthdelta + ?cpdaydelta / 7 AS ?cpduration ) .
50  BIND ( ?cpeffort / ?cpduration AS ?cpweeklyeffort ) .
51  BIND ( ( ?cpweeklyeffort * ?overlap ) AS ?cpreleffort ) .
52  BIND ( xsd:float( ?capacity * ?npduration ) AS ?capacityavailable )
53  }
54  GROUP BY ?newproject ?priority ?competencetype ?availableemployee ?npeffort ?overlap ?capacity ?
      npduration ?capacityavailable
55  HAVING ( ?capacityavailable >= SUM( ?cpreleffort ) + ?npeffort )

```

Listing 3 SPARQL query for identifying innovation potentials (Q2)

```

1  SELECT DISTINCT ?e1 ?competencetype1 ?competencetype2 ?e2 WHERE {
2  # identify non-organizational competence types
3  ?competencetype1 rdfs:subClassOf/ rdfs:subClassOf :competence .
4  MINUS { ?competencetype1 rdfs:subClassOf* :organizationalSkill } .
5  ?competencetype2 rdfs:subClassOf/ rdfs:subClassOf :competence .
6  MINUS { ?competencetype2 rdfs:subClassOf* :organizationalSkill } .
7  FILTER ( ?competencetype1 != ?competencetype2 ) .
8  # remove all competence tuples provided by one employee
9  FILTER NOT EXISTS {
10   ?employee :provides ?competence1 .
11   ?competence1 a ?competencetype1 .
12   ?employee :provides ?competence2 .
13   ?competence2 a ?competencetype2 .
14  }
15  # remove all competence tuples provided by two cooperating employees
16  FILTER NOT EXISTS {
17   ?employee1 :provides ?competence1 .
18   ?competence1 a ?competencetype1 .
19   ?employee2 :provides ?competence2 .
20   ?competence2 a ?competencetype2 .
21   {
22     ?employee1 :cooperatesWith ?employee2 .
23   } UNION {
24     ?employee1 :cooperatedWith ?employee2 .
25   }
26  }
27  # identify employees that should exchange ideas
28  ?e1 :provides ?c1 . ?c1 a ?competencetype1 .
29  ?e2 :provides ?c2 . ?c2 a ?competencetype2 .
30  }
31  ORDER BY ?e1 ?e2 ?competencetype1

```