Available online at www.sciencedirect.com

ScienceDirect

Procedia Technology 15 (2014) 577 – 586

Procedia Technology

2nd International Conference on System-Integrated Intelligence: Challenges for Product and Production Engineering

Towards a formal conceptualization of organizational design techniques for large scale multi agent systems

Markus Schatten[a,*], Petra Grd[b], Mladen Konecki[a], Robert Kudelić[a]

[a]Artificial Intelligence Laboratory, Faculty of Organization and Informatics, University of Zagreb, Pavlinska 2, 42000 Varaždin
[b]Centre for Forensics, Biometrics and Privacy, Faculty of Organization and Informatics, University of Zagreb, Pavlinska 2, 42000 Varaždin

Abstract

Computing in the Internet of Things is increasingly pervasive, with everyday items including clothes, smart-phones, cars and various household appliances gaining sophisticated communication and computing capacities. It seems to be just a matter of time before devices have to collaborate and compete mutually as well as with their users, in order to provide better services to mankind. These embedded computers are increasingly autonomous and connected, and can thus be modeled as agents within multi-agent systems (MAS). Only 30 years ago it was science fiction that over a billion of people will exchange billions of e-mails on a daily basis. Today a scenario of millions of collaborating agents embedded in gadgets and appliances, across various networks may also sound futuristic. However given the current rate of development in electronics, we will soon have to manage large scale MAS (LSMAS) where millions of agents exist, collaborate and compete with each other. While a recent study shows that there are at least 50 organizational structure, superstructure and architecture types employed in modern organizations, there is a lack of research that would apply organizational design methods to organizations composed entirely of agents and agent systems in order to achieve alignment of organizational structure, processes, and reward system with the goals and strategy of the organization. Our research is therefore aimed towards enriching formal design methods for the development of LSMAS to foster the development of self-organizing and adaptable networks of devices that will contribute towards a sustainable development of the information society. In this work-in-progress study we apply a collaborative semantic wiki approach towards formalizing organizational design techniques in order to provide a foundation for future studies of automated LSMAS development.

Peer-review under responsibility of the Organizing Committee of ICIAME 2014.

Keywords: Large scale multi-agent systems; organizational design; semantic wiki; formalization; Internet of Things

* Corresponding author. Tel.: +385-42-390-891; fax: +385-42-213-413.
E-mail address: markus.schatten@foi.hr

Peer-review under responsibility of the Organizing Committee of ICIAME 2014.
doi:10.1016/j.protcy.2014.09.018

1. Introduction

Complex organizational systems in contemporary turbulent market environments have to respect a number of prerequisites in order to have a chance of survival and eventually success. While modern organizations cope with these requirements using various organizational methods, most large-scale multi-agent (LSMAS) proposals and solutions engage in engineering techniques. LSMAS have numerous application areas often associated to the Internet of Things initiative and co-affiliated with domains of complex organizations, including but not limited to intelligent transport systems, logistics, traffic, work flow and process management, pervasive computing, catastrophe management, anticrime/terrorism actions, ecology and nature preservation systems, smart cities, intelligent energy distribution, smart Internet services, share trading, massively multi-player on-line games (MMOGs), infrastructural services for citizens and other.

Complex organizations and LSMAS have quite a number of similarities: they are decentralized to cope with scope, they are organized to cope with complexity, they have to be adaptable to cope with turbulence, they have to take advantage of all their (often heterogeneous) strengths to mitigate weaknesses, they have to be strategic to achieve long-term objectives; not to mention that they are systems of individuals (agents) which provide services, which collaborate and compete with other such systems and operate in a complex environment. Thus, the usage of organizational design methods in the development of LSMAS seems to be a natural suggestion.[1] From this perspective it is our aim to provide a higher level of abstraction for dealing with LSMAS by introducing selected concepts from organization theory into the processes planning, modeling, development, implementation and maintenance.

The authors believe that recent investigations which use organizational design methods for developing MAS including [1, 2, 3, 4, 5, 6, 7, 8, 9] and others, seems to be promising and might allow us to address complex (computational) problems in much more (human) understandable form. The objective of our efforts is to update software engineering methodology one more time with a metaphor (having the well accepted metaphors like assembly language, machine independent programming languages, sub-routines, procedures & functions, abstract data types, objects, services and agents in mind). Thus the main impact of the proposed technique would be a shift in paradigm that will allow us to develop organizations (of agents) in large-scale settings.

In order to apply ideas from organization theory to LSMAS, we have decided to use the concept of organizational architecture which represents a well-established paradigm for describing (human) organizations [10, 11, 12, 13, 14, 15, 16] (table 1).

Herein, we adopt the interpretation of [17] that organizational architecture is a complex organizational system which can be modeled from different but mutually intertwined perspectives: organizational structure, organizational culture, business processes, strategy and individual agents (human or artificial). Additionally we decided to add two important perspectives which are not clearly captured in organizational architecture: organizational dynamics as well as contextual and inter-organizational aspects. Thus our proposed framework shall reside on the following perspectives (defined informally):

Organizational structure defines the decision and information flows of an organization.

Organizational culture defines important intangible aspects of an organization including knowledge, social norms, reward systems, language and similar.

Strategy defines the long term objectives of an organization, action plans for their realization as well as tools on how to measure success.

Processes define the activities and procedures of an organization.

Individual agents define the most important asset of any organization - the individuals actually performing the work.

Organizational dynamics define organizational changes including reorganization of any of the above mentioned components.

[1] The other way around is possible as well, as has been shown in a number of studies dealing with computational organization theory.

Context and inter-organizational aspects define organizational behavior towards its environment including strategic alliances, joint ventures, mergers, splits, spinouts and similar.

Table 1. Various conceptualizations of organizational architecture [17, 16]

Nadler and Tushman(1997)	Merron (1995)	Galbraith (1995)	Henning (1997)	Churchill (1997)	Corporate Transitions International (2004)
	Vision, strategic goals and strategic management	Strategy	The role of the organization		Strategy
Informal organization	Organizational culture	Reward systems	Reward systems	Organizational culture	Organizational culture
Formal organization	Organizational structure	Organizational structure	Groupings	Organizational structure	Organizational structure
Business processes		Processes and lateral links	Business processes and work design		
Human resources		Human resources		Human resource development	Communication

In the following we will particularly concentrate our efforts on organizational structure, organizational dynamics (in form of organizational design methods) as well as context and inter-organizational aspects since these areas are from our perspective not well captured by contemporary LSMAS research.

An important step towards using organizational design concepts in computing is the formalization of these concepts. It is our objective to develop a formal meta-model of organizational design concepts that can be a foundation for model-driven software development tools that take advantage of these important insights from organization theory. Since a lot of organization theory literature is based on metaphors, the formalization process in non-trivial. We decided to take a three step approach towards implementing a meta-model which reflects state-of-the-art knowledge engineering practices: firstly we will develop a semantic wiki on organizational design techniques that will allow us to edit a large corpus of knowledge in a collaborative on-line environment and allow for an initial formalization of important concepts; secondly we will develop a formal semantic Web ontology based on the insights gathered during the semantic wiki development; and thirdly we will use this ontology to develop a formal meta-model as well as a knowledge base of best practices for LSMAS development. The current article gives an overview of the first step dealing with the development of a semantic wiki.

The rest of this article is structured as follows: firstly we describe our methodology in section 2. Afterwards we present the obtained results in section 3. In the end we discuss these results, draw conclusions and give guidelines for future research in section 4.

2. Methodology

In order to approach this complex corpus of organizational design related knowledge we have chosen a semantic wiki approach [18, 19, 20, 21]. Semantic wikis are wiki systems that allow their users to add semantic annotations to wiki text in order to allow for automated inference, querying and intelligent agent implementation [22] and thus represent tools which take advantage of both social applications and the semantic Web initiative. In particular, we have chosen to use the TaOPis system build upon the Ƒlora-2 reasoning engine [23]. TaOPis allows for tagging articles in an object-oriented manner, has OWL (Web Ontology Language) and Flora-2 ontology export facilities [24, 25], a dynamic querying engine [26] as well as ontology visualization [27].

After a literature review[2] of available organizational design techniques, organizational structures, superstructures and architectures (mostly based on [17]), we have firstly collaborated on structuring wiki text about our findings.

[2] The reviewed references aren't given in the bibliography section since more than 230 various references have been considered including books, articles and Websites. The interested reader is advised to visit http://ai.foi.hr/wiki/oovasis where all the references are given under the appropriate wiki pages.

More than 60 organizational design concepts have been identified, each of which has been described on an individual wiki page similarly to Wikipedia editing policies. Afterwards, these wiki pages have been semantically annotated using three particular strategies:

1. An initial taxonomy has been established by tagging content with special attribute-value tags to represent classes (concepts) and their subclasses.

2. Each identified class has been additionally annotated with attribute value tags that have been chosen to be of particular interest for LSMAS implementation. The used attributes included:

- For organizational structure related concepts: criteria of structuring (function, division, territory, product/service, client, process, etc.), type of nodes (agent, organizational unit, process, project etc.), type of edges (power, communication, collaboration, competition etc.), dynamics (dynamic, static), establishment (ad-hoc depending on environment or stable if bureaucracy), number of hierarchies (0 - heterarchy, 1 - 'normal' hierarchy, 2 - matrix, N - tensor), hierarchies overlapping (if multiple hierarchies), number of roles per node (1 in single structures, 2 or more in overlay and multiplex structures), size suitable for (small, medium, large organizations), environment (stable/deterministic, unstable/stochastic, complex), type of services (process, project, hybrid), type of agents (simple, average, complex).
- For organizational design technique related concepts: type of change (structure, culture, process, strategy, individuals), impact of change (small, medium, large), source reason for change (quality, exertion, environment, knowledge etc.), key area of influence (inter organizational adaptability, self-reorganization, robustness/fault-tolerance, strategic management, organizational memory).

3. Each concept has additionally been annotated with keywords in order to capture additional ideas which might have been missed using the previous two approaches.

Using the OWL export feature of TaOPis, the obtained taxonomy was imported into the Protégé ontology management system [28]. After the annotation process we have implemented two intelligent agents which allowed us to reason about the domain. Both agents were implemented in Flora-2 similarly to the work presented in [29]. The first agent was designed to construct a list of keywords which apply to a given class and all its subsequent subclasses. Its implementation was fairly simple, and is as follows:

```
keywords ( ?class , ?keywords ) :-
   ?keywords = collectbag { ?k | ?_: ? class [ keyword ->?k ] }.
```

The body of the rule just collects a list of all values for the keyword attribute for all objects in the given class. Note that Flora-2 automatically infers that all objects which are instances of subclasses of the given class, are instances of this class as well. Basically, this agent allowed us to create tag clouds of particular classes of organizational design concepts.

The second agent was by far more complex. Its task was to find classes which are conceptually similar in order to identify classes which (from a LSMAS perspective) can be considered one and the same thing. Basically, two concepts are similar if they have same values for the previously outlined attributes we identified as important for LSMAS implementation. Such identified concepts shall be considered identical in our future research, but it is important to stress that they might be considerably different in human organizations and that the selection of attributed we performed highly influences the end result. The used approach is similar to the one described in [30]. We will omit the full implementation of the agent since some auxiliary predicates deal with filtering and clustering the results, but show only the similarity predicate that defines when two classes are considered similar:

```
   similar ( ?c1, ?c2 , ?class ) :-
         ?i1 :? class [ title ->?c1 ],
         ?i2 :? class [ title ->?c2 ],
```

```
        ? i1 \= ?i2,
        criteria ( ?class , ?lst ),
        ?att1 = collectset {
                ?va1 |
                  ?i1 [ ?a1->?v1 ],
                  member ( ?a1 , ?lst )@_plg ( basics ),
                  ?va1 = p( ?a1 , ?v1 )
        },
        ?att2 = collectset {
                ?va2 |
                  ?i2 [ ?a2->?v2 ],
                  member ( ?a2 , ?lst )@_plg ( basics ),
                  ?va2 = p( ?a2 , ?v2 )
        },
        ?att1 \= [ ],
        ?att2 \= [ ],
        permutation ( ?att1 , ?att2 )@_plg ( lists ).
```

We firstly declare that we want to consider the title attributes of the considered objects (the object id in TaOPis is an internal symbol which is not very descriptive) as well as that these objects have to be instances of the considered class. Then we make sure that we are not dealing with one and the same object (in the case of TaOPis, the same wiki page). Afterwards we bind the ?lst variable to the list of attributes that are considered using the criteria=2 predicate which makes sure that the right list is bound for the considered class. Then we collect two lists of attribute-value pairs in the form p(attribute; value) using the built-in collectset aggregate operator. We make sure that these lists aren't empty and in the end state that the lists are permutations of each other, meaning that the sets of attributes for the two considered classes are equivalent.

3. Results

Example images from the developed semantic wiki are shown on figure 1: subfigure (a) shows the front page of the wiki which features an outline of all articles on the system, (b) shows an example page of a particular organizational structure (the tensor structure), (c) shows an example of semantic annotation which was used to describe a given wiki page, and (d) shows an automatically generated navigation for a given annotation. [3]

The semantic annotation has yielded a social taxonomy as perceived by the authors which was exported to OWL and imported into the Protégé ontology management system. After being initially cleaned up of standard TaOPis classes and individuals, the taxonomy is shown on figure 2.

As one can see from the figure five top-level classes have been identified: (1) DesignFactors, (2) DesignMethod, (3) OrganizationalStructure, (4) SuperStructure, and (5) OrganizationalArchitecture, each with corresponding subclasses. DesignFactors represent factors which influence the design of a given organization and can be internal and external. In particular development of science and technology, human resources, institutional environment, location, market, mergers and acquisitions, organizational life-cycle, product/service, size of organization, strategic alliances, strategy, tasks and technology were identified as important organizational design factors. DesignMethod is a class which instances are common organizational design practices. The literature review revealed 11 such design methods, but also indicated that further investigation about other methods addressing other aspects of organizational architecture is necessary. The identified methods include: business process reengineering, total quality management, lean management, kaizen, six-sigma, the Taguchi method, the complex analytical method, organizational chaos theory, knowledge management, communities of practice as well as organizational memory.

On the other hand, OrganizationalStructure is a class in which all concepts dealing with various forms of structuring organizational units are captured. We decided to use two criteria for approaching various structures described in literature: (1) is it the main structure or is it an overlay or superstructure build-up over the main

[3] By clicking on any attribute-value tag on some page, TaOPis will show a list of all pages that are tagged with the same tag.

structure, and (2) is the structure hierarchical or heterarchical (network-based) in nature. Since superstructures can potentially spread across multiple organizations, we decided to introduce an additional upper-level class named SuperStructure to capture such structures. According to the second criteria, all main structures have been categorized into hierarchical, heterarchical and hybrid structures (structures which capture some aspects of both hierarchy and heterarchy). Hierarchical structures can be divisional (customer oriented, product/service oriented or territory oriented), functional, project oriented, matrix, and tensor structures. Heterarchical structures on the other hand can be dynamic and static network, fishnet structure, Hollywood structure, infinite flat hierarchy, internal markets, or even spaghetti structures. At last, hybrid structures identified in literature include the academic, the front-back, and the inverted structure.

SuperStructure, as mentioned previously, represents all structures that are formed above conventional organizational structure and thus represent inter organizational and overlay structures. They were not that easily categorized as organizational structures, due to a greater pool of possibilities. We have chosen to classify them into five (not necessarily distinct) subclasses, but need to stress that further analysis is required to formalize them in an appropriate way. The subclasses were: (1) stable superstructures - structures which are relatively stable over time including the cluster organization and the starburst organization, (2) adhocracies - structures which are formed ad-hoc depending on a particular problem or opportunity in the environment and usually disapear when the issue is resolved including team based structures, the spider's web, the amoeba organization, the virtual organization and the modular organization, (3) strategic alliances - long lasting partnerships between organizations of various forms including e-alliances, keiretsu and cheabol, (4) mergers and acquisitions - organizational integrations, and (5) fractal organizations which deal with the concept of self-similarity between organizational units on various hierarchical levels and the spreading of this similarity across them (an example can be objectives).

The OrganizationalArchitecture class is a common denominator for concepts that deal with at least two aspects of organizational architecture (as opposed to OrganizationalStructure and SuperStructure which deal only with the structural aspect). We have identified 15 such architectures describe in literature which are the T-form organization, the Shamrock organization, federalism, the inverted doughnut, 4C organization, the hypertext organization, the platform organization, the strategic organization, the individualized organization, the 3I organization, the information based organization, the empowered organization, the open organization, the learning organization and bio teaming. Since a lot of semantics is lost by just creating a taxonomy and annotating concepts with predefined attributes, we decided to additionally add keywords to each identified concept, in order to provide hints for a future (more detailed) analysis that should allow us to develop the desired ontology or organizational design methods for LSMAS. By using the first agent we collected these keywords across classes we considered most important for further analysis. The results obtained by the agent are summarized in table 2. The obtained keywords do not only provide hints to concepts that have to be additionally analyzed, but also imply possible connections between concepts. For example, there seems to be a connection between design methods (especially knowledge management related ones) and the learning organization, adhocracies seem to be heterarchic in nature etc. We shall analyze these and similar issues in our further research. Since concepts in organization theory are particularly described with metaphors it is often the case that some concepts are in practice actually very similar. This is especially true for our case of LSMAS organizations, since a great number of human related issues do not apply to artificial agents. In order to find concepts which are very similar and would not add extra benefit if implemented separately, we have implemented the second agent. Using this agent we were able to identify the following clusters of conceptually similar terms:

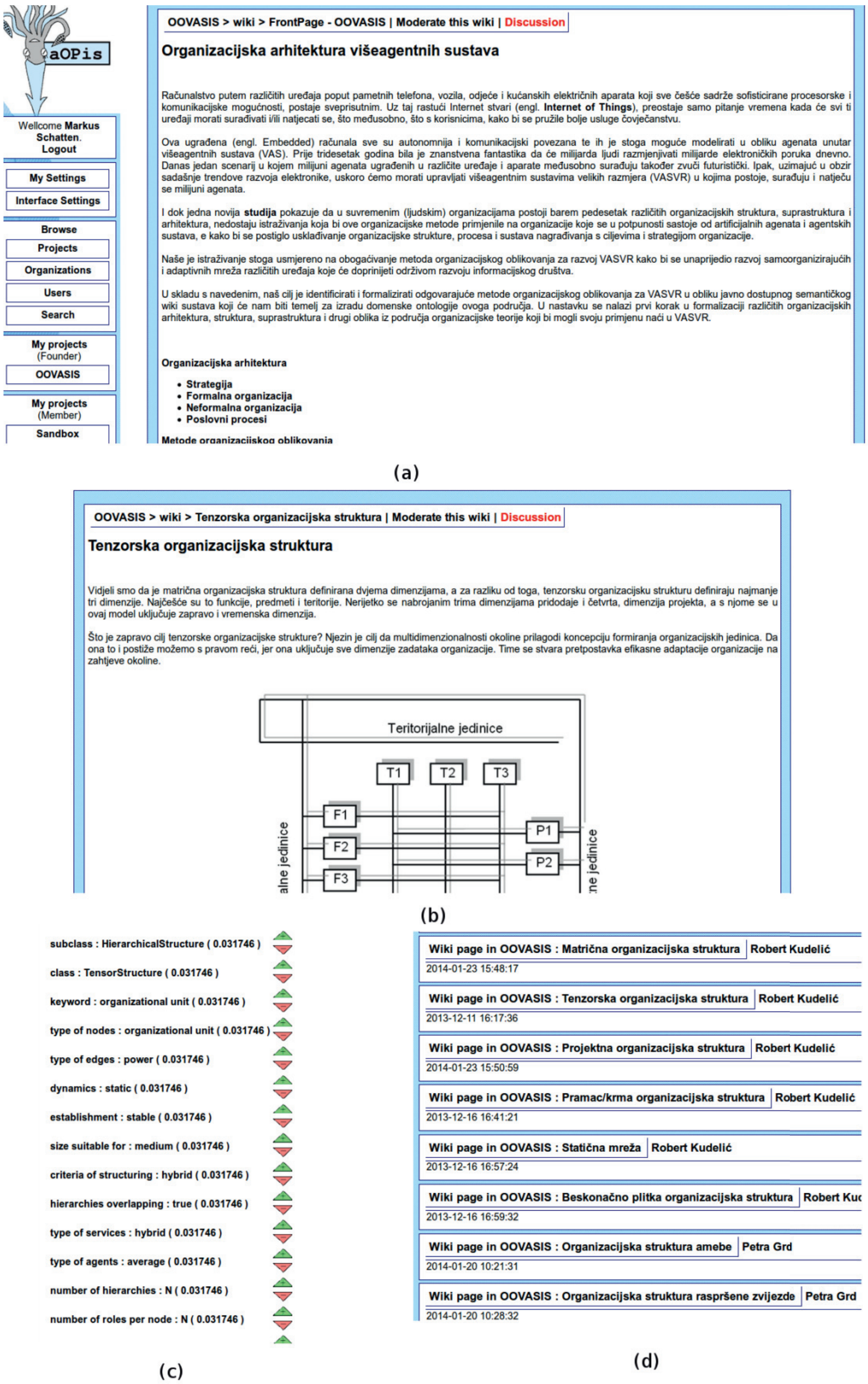

Fig. 1. Semantic wiki on organizational architecture.

- knowledge management, communities of practice, organizational memory
- lean management, kaizen
- dynamic network, spaghetti structure, Hollywood organization, networked organization
- strategic alliances, e-alliances, keiretsu
- virtual organization, modular organization, the spider's web

Table 2. Concepts and corresponding keywords

Concept	Keywords
HeterarchicalStructure	dynamic hierarchies, external, fishnet, flat hierarchy, heterarchy, internal, internal market, network, organizational unit, self-activation
HierarchicalStructure	client, division, function, matrix, organizational unit, product/service, project, tensor, territory
HybridStructure	back, front, heterarchy, hierarchy, hybrid structure, inverted hierarchy, organizational unit
Merger	organizational merging, superstructure
StableSuperStructure	cluster, organizational split, organizational unit, overlay structure, stable superstructure, starburst, strategic
Adhocracy	action team, adaptability, adhocracy, amoeba principle, creativity, experimentation, heterarchy, individual initiative, organizational split, organizational unit, overlay structure, pseudo hand, readiness, short term, team
StrategicAlliance	conglomerate, heterarchy, holding company, overlay structure, partnership, strategic
DesignMethod	DMAIC, analytics, black belt, business transformation, call center, common practice, community of practice, continuous improvement, cross functional, culture, data, document, green belt, hoshin, improvement team, incremental, individual knowledge, information, information management, information network, innovation, intellectual advantage, intellectual capital, kaizen, knowledge, knowledge bank, knowledge based system, knowledge sharing, learning organization, master black belt, metric, objective, orange belt, organigram, organizational exertion, organizational learning, organizational unit, procedures, process, process owner, quality, quality leader, reengineering, robustness, role-play system, routines, rules, seiketsu, seiri, seiso, seiton, shitsuke, strategy, value flow, virtual knowledge team, white belt, wisdom, yellow belt
DesignFactors	development of science and technology, human resources, institutional environment, location, market, mergers and acquisitions, organizational life-cycle, product/service, size of organization, strategic alliances, strategy, tasks and technology

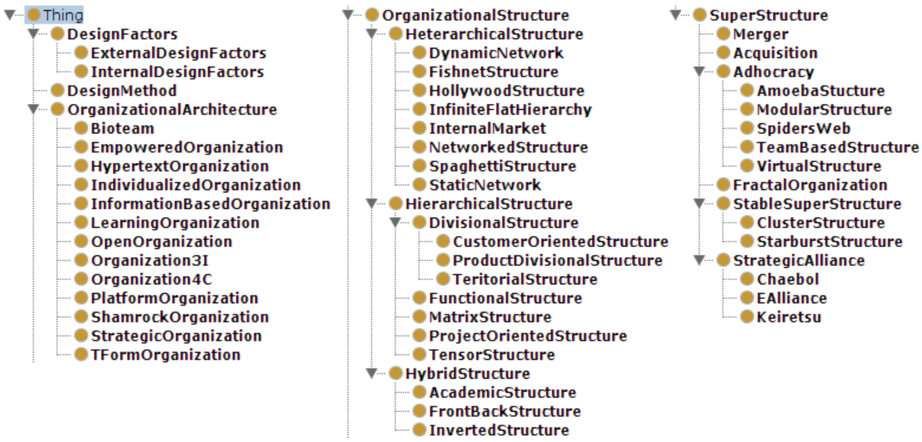

Fig. 2. Initial taxonomy visualized with Protégé.

4. Conclusions & Future Research

In this paper we have described the first step towards the creation of an organizational design meta-model for LSMAS, which is again only the first step towards establishing an integrated development environment that would enable us to manage the LSMAS lifecycle. We believe that by using organizational design techniques (techniques that humans use to organize humans) for the modeling, specification, development, and maintenance of LSMAS (processes usually relying on complex engineering techniques), these production processes will be easier for humans to grasp and understand, similarly to other well established metaphors like agents, objects or services. Computing systems in the emerging Internet of Things are becoming distributed and ubiquitous and these trends present major challenges for product and production engineering in industry as well. Due to the large body of knowledge in organization theory, we have decided to use a number of steps before developing the final meta-model, namely: (1) a wiki, (2) a semantic wiki, (3) a semantic Web ontology.

The semantic wiki approach described in this paper has shown a number of advantages but also a number of problems have occurred. On one hand, it allowed us to work on a large corpus of knowledge in a dynamic and collaborative way. It was fairly easy to add additional ideas and concepts to the wiki as well as split and merge articles when new insights came to attention. On the other hand the formalization process has been shown to be much more complex than initially envisaged. Since there has been a great number of concepts involved, it was not always obvious how to semantically annotate them nor into which class to classify them. Especially when splits and merges of articles occurred, the annotations had to be deleted and adjusted to reflect the new organization of the taxonomy.

The obtained taxonomy showed us gaps which have to be addressed in future research. In this semantic wiki we have only formalized aspects dealing with organizational design techniques, organizational structures and architectures, but other aspects including strategy, culture, processes and individuals have not been considered thoroughly.

The implemented agents gave us hints on concepts to consider when developing a formal ontology of the domain including connections between these concepts. Another aspect that became apparent is that organizational design techniques have to be considered in more detail due to the large number of different keywords that have been used to describe them. The agents also identified concepts that are similar enough to be considered equal from the perspective of LSMAS development.

Our next steps will include the definition of a high-level conceptualization of organizational architecture based on the insights we obtained by developing this semantic wiki which shall be the foundation for a formal ontology of organizational design techniques for LSMAS development.

References

[1] Fischer K, Schillo M, Siekmann J, Holonic multiagent systems: A foundation for the organisation of multiagent systems, in: V. Marık, D. McFarlane, P. Valckenaers (Eds.), Holonic and Multi-Agent Systems for Manufacturing. Lecture Notes in Comp. Sci. 2003;2744:71-80.

[2] Blanzieri E, Giorgini P, Giunchiglia F, Zanoni C, Implicit Culture-Based personal agents for knowledge management, 2003, p. CH439. URL http://www.springerlink.com/content/hel0cf0y084wp43n.

[3] Ferber J, Gutknecht O, Michel F. From agents to organizations: An organizational view of multi-agent systems, in: P. Giorgini, J. P. Muller, J. Odell (Eds.), Agent-Oriented Software Engineering IV. Lecture Notes in Computer Science 2003;2935: 443–459.

[4] Filipe F. Information fields in organization modeling using an eda multi-agent architecture, in: L. Elst, V. Dignum, A. Abecker (Eds.), Agent-Mediated Knowledge Management. Lecture Notes in Computer Science 2004;2926:130–142.

[5] Horling B, Lesser V. A Survey of Multi-Agent Organizational Paradigms, The Knowledge Engineering Review 2005;19(4);281–316.

[6] Kolp M, Giorgini P, Mylopoulos J. Multi-agent architectures as organizational structures. Autonomous Agents and Multi-Agent Systems 2006;13(1):3–25.

[7] Argente E, Julian V, Botti V. Multi-agent system development based on organizations, Electron. Notes Theor. Comput. Sci. 2006;150(3): 55–71.

[8] Garcia E, Argente E, Giret A, Botti V. Issues for organizational multiagent systems development, in: Sixth International Workshop From Agent Theory to Agent Implementation (AT2AI-6), Citeseer, 2008. pp. 59–65

[9] Desai N, Chopra AK, Singh MP, Amoeba. A methodology for modelling and evolving cross-organizational business processes, ACM Transactions on Software Engineering and Methodology (TOSEM) 19 (2), 2009.

[10] Nadler D A, Gerstein M S, Shaw R B. Organizational Architecture, Designs for Changing Organizations. San Francisco: Jossey-Bass; 1992.

[11] Nadler D, Tushman M. Competing by design: The power of organizational architecture. New York: Oxford University Press; 1997.

[12] Merron K. Riding the wave: Designing your organization's architecture for enduring success. Van Nostrand Reinhold; 1995.

[13] Galbraith J R. Designing organizations: An executive briefing on strategy, structure, and process. San Francisco: Jossey-Bass; 1995.

[14] Henning J P. The Future of Staff Groups: Daring to Distribute Power and Capacity. Berrett-Koehler Store; 1997.

[15] Churchill C. Managing growth: The organizational architecture of microfinance institutions, in: USAID Microenterprise Best Practices Project Paper, 1997. pp. 7–26, 81–87

[16] Miciunas G, Cre/fm organizational architecture: Structuring staff success, the Environments Group, Available at http://www.envgroup.com/browse/presentations/IFMA2002_staffsuccesspaper.pdf, 2002.

[17] Žugaj M, Schatten M. Arhitektura suvremenih organizacija, Tonimir and Faculty of Organization and Informatics, Varaždinske Toplice, Croatia, 2005.

[18] Campanini S E, Castagna P, Tazzoli R. Platypus wiki: a semantic wiki wiki web, in: SemanticWeb Applications and Perspectives (SWAP) 1st Italian Semantic Web Workshop, Ancona, Italy, 2004.

[19] Aumueller D, Auer S. Towards a semantic wiki experience - desktop integration and interactivity in wiksar, in: 1st Workshop on The Semantic Desktop, Next Generation Personal Information Management and Collaboration Infrastructure, 2005.

[20] Kim H, Choi J. A semantic web-enabled wiki system for ontology construction and sharing, Journal of KISS: Software and Applications 2006;33(8):703–717.

[21] van Elst L, Kiesel M, Schwarz S, Buscher G, Lauer A, Dengel A. Contextualized knowledge acquisition in a personal semantic wiki, in: Knowledge Engineering: Practice and Patterns. 16th International Conference, EKAW 2008, Springer-Verlag, Berlin, Germany, 2008. pp. 172–187

[22] Schaert S, Bry F, Baumeister J, Kiesel M. Semantic wiki. Informatik-Spektrum 2007;30(6):434–439.

[23] Yang G, Kifer M, Zhao C. Ƒlora-2: A rule-based knowledge representation and inference infrastructure for the semantic web, in: Second International Conference on Ontologies, Databases and Applications of Semantics (ODBASE) Proceedings, Catania, Sicily, Italy, 2003.

[24] Schatten M, Čubrilo M, Ševa J. A semantic wiki system based on f-logic, in: B. Aurer, M. Bača (Eds.), 19th Central European Conference on Information and Intelligent Systems – CECIIS2008 Conference Proceedings, Faculty of Organization and Informatics, 2008. pp. 57–61

[25] Schatten M. Programming languages for autopoiesis facilitating semantic wiki systems, Ph.D thesis. University of Zagreb, Faculty of Organization and Informatics, Varaždin, Croatia, February 2010.

[26] Schatten M, Čubrilo M, JŠ eva. Dynamic queries in semantic wiki systems, in: B. Aurer, M. Bača (Eds.), Proceedings of the 20th Central European Conference on Information and Intelligent Systems, Faculty of Organization and Informatics, 2009. p. 13–20

[27] Štritof T, Schatten M. Visual knowledge discovery on semantic wikis, in: B. Aurer, M. Bača, M. Schatten (Eds.), Proceedings of the 21st Central European Conference on Information and Intelligent Systems (CECIIS 2010), Faculty of Organization and Informatics, 2010. p. 41–45

[28] Noy N F, Crubézy M, Fergerson R W, Sintek W, Decker S, Musen M A. Creating semantic web contents with protege-2000. IEEE Intelligent Systems 2001;16(2): 60-71.

[29] Schatten M. Knowledge management in semantic social networks. Computational & Mathematical Organization Theory 2013;19(4):538-568.

[30] Schatten M, Magdalenic I, Vrdoljak B. Towards ontology alignment of e–business standards using owl and f–logic. International Journal of Metadata, Semantics and Ontologies 2011;6(3):207–218.