Reorganization in Multi-Agent Architectures: An Active Graph Grammar Approach

Markus Schatten

University of Zagreb, Faculty of Organization and Informatics, Varaždin, Croatia

Abstract

Background: Organizational architecture is a holistic approach to design of humane organizations and studies an organization from five perspectives: structure, culture, processes, strategy and individuals. In this paper, the concept of organizational architecture is firstly formalized using the fractal principle and then applied to multi-agent systems' (MAS) organizations. **Objectives**: Providing a holistic framework for modeling all aspects of MAS reorganization. **Methods/Approach**: MAS organizations are formalized using graph theory and a new active graph rewriting formalism inspired by active database theory is intoduced. **Results**: The newly developed framework is graphical, event-driven and applicable in a distributed MAS environment. **Conclusions**: By defining organizational units, processes, strategies and cultural artefacts in a recursive way, it is shown that labeled graphs and hypergraphs can be used to model various levels of organizational architecture while active graph grammars allow one to model reorganization of each of the architectural perspectives.

Keywords: multiagent systems, organizational dynamics, active graph grammars, organizational architecture, graphical framework
JEL classification: C0, C61, C88
Paper type: Research article

Received: 15, February, 2013
Revised: 27, February, 2013
Accepted: 8, March, 2013

Citation: Schatten, M. (2013). "Reorganization in Multi-Agent Architectures: An Active Graph Grammar Approach", Business Systems Research,Vol. 4, No. 1, pp. 14–20.
DOI: 10.2478/bsrj-2013-0002

Acknowledgement: This article is an extension of the previously published conference paper: Schatten, Markus. Active Graph Rewriting Rules for Modeling Multi-Agent Organizational Dynamics // Proceedings of the IBC 2012, 1st International Internet & Business Conference / Ivković, Miodrag ; Pejić Bach, Mirjana ; Šimičević, Vanja (ur.). Rovinj : BIT Society, 2012. 180-185. The article extends the conference paper with additional definitions for modeling processes, strategy, and culture, since the previous paper only dealt with structure. The new article now provides a complete framework for modeling reorganization of MAS architecture.

Introduction

Multi-agent systems (MAS) are a well established abstraction for modeling distributed computing systems. MAS consist of a number of interacting agents, which act according to the goals and motives of their users. In order to achieve their goals, agents need the abilities of cooperation, negotiation as well as coordination. An important issue in MAS design is the development of agents, which are able to interact with other agents, in order to achieve their goals. This is the field of MAS organizational design.

In the following we will focus on this problem by using a well established formalism, namely graph grammars (GGs) which allow us to capture and program structural regularities (Nagl, 1979). The idea of applying GGs to MAS is not a new one. For example Nagendra Prasad et al. (1996) present a GG based task structure specification language for the TÆMS agent models. They went on and extended GGs with stochastic and attributed productions. Nagendra Prasad and Lesser (1999) later on used this tool to model the topological relationships occurring in task structures, as did Lesser et al. (1999) to implement a MAS for managing an intelligent environment as part of the intelligent home project (IHome). Giese et al. (2003) employ story diagrams introduced by Fischer et al. (2000), whereby story patterns in fact illustrate graph grammar formulae, to model safety-critical MAS macro- and micro-architectures in UML. Later Becker et al. (2006) extend the approach to develop a verification technique for arbitrarily large multi-agent systems in mechatronics. Smith et al. (2009) use embedded graph grammars to deploy and coordinate robots (agents) in various (physical) formations. They introduce an agreement protocol for agents to agree mutually before applying a production rule.

While these studies provide important particular insights into the matter, herein we will introduce a more holistic approach. Firstly, we shall define MAS organizations in terms of organizational architecture as presented in (Žugaj and Schatten, 2005). Thus MAS are not only collections of agents structured in a certain way, but an agent organization consisting of organizational structure, organizational culture, strategy, processes and individual agents (human resources in humane organizations). These five perspectives represent important and different views of the same (agent or humane) organization:

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 etc.

Strategy defines the overall objectives of an organization 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.

All of these perspectives are subject to changes due to changes in the environment of the organization. Organizational change is probably one of the most important aspects of successful organizations. Since the modeling of individual agents is a very well studied field of research, in the following we will focus on organizational dynamics of the other perspectives of organizational architecture.

Organizational Architecture in MAS

The previously outlined studies, deal mostly with individual agents as part of an agent organization, whereby each agent takes a certain role. Herein our focus will be an actual organizational unit to which some agent might belong. This allows us to use the fractal organization principle (Warnecke, 1992; Parr Rud, 2011) which allows us to define organizational units recursively.

Definition 1 *An organizational unit is defined as follows:*

- *Any agent is an organizational unit.*
- *If $O = \{o_1, o_2, ..., o_n\}$ is a set of organizational units which collaborate with a common objective, then O is an organizational unit.*

This definition is very subtle since it allows us to deal with agents, groups/teams of agents, organizations of agents, networks of organizations of agents as well as virtual organizations of agents (in the sense of (Barnatt, 1995) virtual organizations as overlay structures) in the same way. Note that the term objective here is arbitrary and could easily be replaced with function, goal, mission, unit name etc. Also note that the fixed point in this definition is the individual agent. The implications of this are twofold: (1) it allows us to connect the perspective of structure with the perspective of individuals, and (2) it allows us to model structure as a network of agents, which is defined over the relation of a common goal. This further implies connections to the strategy perspective which in turn deals with various goals and objectives.

The fractal organization principle, allows us further to define other perspectives. For instance the processes perspective, might be modeled using the following definition.

Definition 2 *An organizational process is defined as follows:*

- *Any activity performed by some individual agent is an organizational process.*
- *If $P = \{p_1, p_2, ..., p_n\}$ is a set of organizational processes which are mutually influenced, then P is an organizational process.*

This definition allows us to model organizations as a network of processes which might be defined in a number of ways (Hernaus et al., 2012). For instance, the relation between processes might be that one process uses inputs from another or that two processes are using the same resources, or even further that two processes are performed by the same organizational unit or that they are crutial for the same organizational goal. Note that the fixed point in this defintion is an atomic activity performed by some individual agent. From this reasoning follows that we are able to connect processes with all the other perspectives of organizational architecture.

By using the fractal principle further, we can define the strategy perspective as follows:

Definition 3 *An organizational strategy is defined as follows:*

- *Any measurable objective that can be achieved by an atomic activity is a strategy.*
- *If $S = \{s_1, s_2, ..., s_n\}$ is a set of strategies which are mutually intertwined, then S is an organizational strategy.*

Similar to the previously outlined, this definition allows us to study agent organizations as networks of objectives (Obradović et al., 2007). This network might again be defined in a number of ways, including but not limited to influence (the outcome of one strategy influences another), responsibility (two strategies are under the responsibility of one and the same organizational unit), achieveability (two strategies can be achieved by one and the same organizational process), etc. Note again that the fixed point is an atomic objective that can be achieved by an atomic activity, which in turn is performed by an individual agent. Again, there is a mutual corespondence between the strategy perspective and all other organizational perspectives.

The cultural perspective needs further explanation. Organizational culture in human organizations deals with the intangible aspects of organizations which nevertheless can have huge impact on organizational performance. These aspects might include language, symbols, rituals, customs, methods of problem solving, implicit knowledge etc. (Brčić , 2002). Organizational culture is inherently a complex cybernetic system, and probably very hard to formalize in humane organizations. Thus, to keep thing simple but still allow for complex cultural situations to be modeled in multi-agent organizations, we will focus on the concept of cultural artifact that are (more or less tangible) representations of the aforementioned intangible aspects. In this regard, cultural artifacts might be represented as knowledge in some agents knowledge base, written norms of behaviour like language protocols or processes through which agents acquire (cultural) knowledge. From this perspective we can define culture recursively:

Definition 4 *An organizational culture is defined as follows:*

- *Any cultural artifact is considered an organizational culture.*
- *If $C = \{c_1, c_2, ..., c_n\}$ is a set of organizational cultures which are mutually intertwined, then C is an organizational culture.*

Thus, again organizational culture can be viewed as a network, this time as a network of cultural artifacts or sets of cultural artifacts. Since these artifacts can be a number of different things, including agent knowledge, norms, procedures and knowledge acquisition processes, it is evident that there is a connection between this and the other perspectives of organizational architecture. The fixed point in this definition are the actual cultural artifacts which might be connected in various ways like influence (one artifact influences another), or across organizational boundaries like organizational units (two artifacts apply to both considered units), processes (two artifacts influence the same organizational process) or strategy (two artifacts both have influence on decision making regarding some particular strategic objective).

The most important observation regarding the given definitions is that all perspectives represent some form of network. Networks are usually studied by using graph theory, and thus we shall in the following make use of these definitions to apply a graph formalism for modeling organizational dynamics in MAS. A similar approach to organizational architecture was given by (Krackhardt and Carley, 1998) in form of the MetaMatrix model which formalizes various perspectives of organization (personnel, knowledge, resources, tasks, organizations) in form of different (graph) matrices. By using matrix algebra, one can use these matrixes to reason about various aspects of these perspectives. The approach put forward herein gives a more general definition of these networks, allows for a hierarchical representation of the modeled networks, and what is most important allows for modeling of organizational dynamics.

Active Graph Rewriting Rules

Inspired with active database theory, we will introduce active graph rewriting rules (AGRR) over labeled graphs. We will define labeled graphs in an (unusual) object-oriented way.

Definition 5 *A label is an ordered pair $a : v$ in which a is an attribute and v is a value.*

Definition 6 *Let N be a set of nodes, $E \subseteq N \times N$ a set of edges, $L = L^N \cup L^E$ a set of labels (whereby L^N is a set of node labels and L^E a set of edge labels). Let furthermore $\nu \subseteq N \times L^N$ and $\varepsilon \subseteq E \times L^E$ be two corresponding relations which map nodes and edges to their labels respectively. We denote the tuple $(N, E, L, \nu, \varepsilon)$ as a labeled graph.*

To further foster object-orientation we will extend node labels with methods.

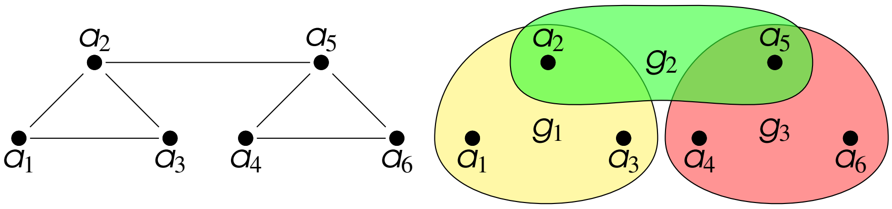
Definition 7 *A method label is the ordered pair $m/a : v$ in which m is a function of arity a and v is a return value. Usually the first argument to the function is a reference to the current node.*

In the following we will model MAS using labeled graphs, whereby fixed points as elaborated above (agents, agent activities, atomic objectives, and cultural artifacts) are denoted with nodes, and their interaction with edges. To model organizational elements of higher order, we need to introduce labeled hypergraphs.

Definition 8 *Let N be a set of nodes, $\Xi \subseteq \mathcal{P}(N) \times \mathcal{P}(N)$ be a set of hyperedges (whereby $\mathcal{P}(N)$ denotes the power set of N), and $L = L^N \cup L^\Xi$ a set of labels (whereby L^N is a set of node labels and L^Ξ a set of hyperedge labels). Let furthermore $\nu \subseteq N \times L^N$ and $\varepsilon \subseteq \Xi \times L^\Xi$ be two corresponding relations which map nodes and edges to their labels respectively. We denote the tuple $(N, \Xi, L, \nu, \varepsilon)$ as a labeled hypergraph.*

In order to model MAS organizational structure, we will introduce two special labels: (1) c being an edge label denoting that two nodes (organizational units) which are connected through such a label collaborate, and (2) g being a node label which denotes the goal of the node. Note that g could easily be a method function for various situations in which an organizational unit might be. Note also that hypergraphs allow us to express higher order relations between sets of organizational units. For example consider a MAS organization that consists of 6 agents ($a_1, ..., a_6$) as shown on figure 1.

Figure 1 Left: Collaboration Graph; Right: Corresponding Organizational Unit Hypergraph



Source: Author's drawing

Agents a_1, a_2 and a_3 collaborate together with a common (particular) goal g_1. The same holds for agents a_2 and a_5 with goal g_2, as well as agents a_4, a_5 and a_6 with goal g_3. We can model this MAS organizational structure either with the labeled graph on the left if we want to analyze the mutual connections of individual agents, or with the hypergraph on the right to analyze organizational units on a higher level. The organizational unit around objective g_2 can be interpreted in various ways, for instance as a management team or a virtual overlay unit that integrates the other units.

To model reactive organizational behavior we have to introduce changes in the structure. Changes are modeled with events. From our perspective there are four main types of events that can occur over time on a graph: (1) update events (due to change of any part of the graph), (2) temporal events (absolute, relative and periodical), (3) implicit events (any event regardless of type that meets a given condition), and (4) complex events (a combination of events constructed using the usual logic operators $\wedge$, $\vee$, $\neg$, $\Rightarrow$, $\Leftrightarrow$).

Definition 9 *With G_i we denote the state of graph G in time i (we assume a discrete time isomorphic to the set of natural numbers $\mathbb{N}$). $\Delta(G_i) = G_{i+1}$ means that G_{i+1} is the result of a changing state in G_i.*

- *An update event $u(G)$ holds in G_i iff $G_i \neq G_{i-1}$. We can classify this type of event further into various changes (insertion, deletion) of various parts of the graph (nodes, edges, labels).*

- *An absolute temporal event $\text{◷}_i(G)$ hold iff G is in state i.*
- *A relative temporal event $\text{◷}_{i+k}(G)$ holds iff G is in state l and there exists a state i such that $i+k=l$.*
- *A periodic temporal event $\text{◷}_{\%j}(G)$ holds iff G is currently in state i and it holds that $i \equiv 0\%j$*
- *An implicit event $\square_F(G)$ holds iff formula F holds (we read any event that satisfies F).*
- *If e_1 and e_2 are two events then:*
  - *$\neg e_1$ holds if e_1 does not hold.*
  - *$e_1 \wedge e_2$ holds if both e_1 and e_2 hold.*
  - *$e_1 \vee e_2$ holds if either e_1 holds or e_2 holds.*
  - *$e_1 \Rightarrow e_2$ doesn´t hold only if e_1 holds but e_2 does not.*
  - *$e_1 \Leftrightarrow e_2$ holds if both $e_1 \Rightarrow e_2$ and $e_2 \Rightarrow e_1$ hold.*

Note that the above definition of events applies both to labeled graphs and subgraphs if edges are replaced with hyperedges. We are now able to introduce AGRR. An AGRR has the ECA (Event-Condition-Action) form borrowed from active database theory, and allows us to state what change has to be done on a graph if a certain event occurs and if a certain condition is met.

Definition 10 *An active graph rewriting rule (AGRR) has the form $E \xrightarrow{C} A$ whereby E is an event, C is a condition formula, and $A = L \rightarrow R$ is a graph transformation. L is called the left-hand-side or pattern graph of the transformation, and R is called the right-hand-side or the replacement graph. An active graph rewriting or active graph grammar (AGG) is a set of AGRRs.*

Modeling Examples

In order to show an application example of AGG to MAS organization we will create a model of the amoeba organization (Daft, 1992). This organizational structure represents a biomimetic metaphor (Schatten and Žugaj, 2011) in which organizational units are so called amoebas which are autonomous and can split and merge if the number of employees is greater or smaller than given limits respectively.

For example, if such an organizational unit acquires more then 100 employees (due to employment of new personnel), the unit will split into two equal amoebas each taking part of the employees. Assume that this organizational unit has been represented by a labeled hypergraph, in which a special node *n* is responsible for tracking the need for a split and is labeled with the label $role:\mathbb{BOSS}$ (the definition of this node is arbitrary, but a token based distributed algorithm might be used).

To model such an organization we can use the following AGGR:

$$u(G) \xrightarrow{(n.count()>100)\ \wedge\ (n.role=\mathbb{BOSS})} L_G \rightarrow R_G$$

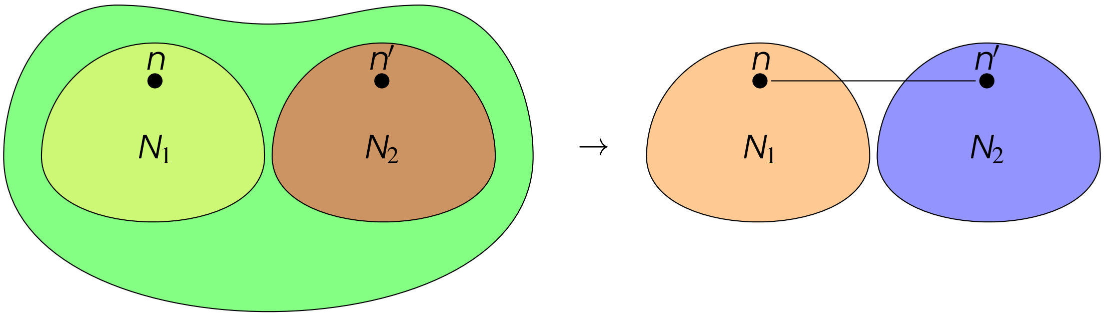
Whereby $L_G = (N_1 \cup N_2, \{\{n_i | n_i \in N_1 \cup N_2\}\}, L, \nu, \varepsilon)$ and $R_G = (N_1 \cup N_2,\ \{\{n_i | n_i \in N_1\},\ \{n_j | n_j \in N_2\}, \{n, n'\}\},\ L, \nu \cup \{(n', role:\mathbb{BOSS})\}, \varepsilon)$. We read on any update event, if a node detects that it has more than 100 collaborators (method $count()$) and if it is the node labeled with $role:\mathbb{BOSS}$, then split the organizational unit connected through the hyperedge $\{n_i | n_i \in N_1 \cup N_2\}$ into two organizational units connected through hyperedges $\{n_i | n_i \in N_1\}$ and $\{n_j | n_j \in N_2\}\}$. Also connect the responsible $role:\mathbb{BOSS}$ node in one organizational unit with the newly established node n' in the other organizational unit. The split into node sets N_1 and N_2 here is arbitrary, but could be modeled in more detail with additional constraints (to for example take into account that the newly established units have to have all the same roles defined as the original unit etc.). This AGGR can also be represented graphically as shown in figure 2.
Source: Author's drawing

On this graphical representation we introduced two arbitrary node sets (N_1 and N_2) on the left-hand-side which are transformed into hyperedges on the right-hand-side.

Conclusion & Future Research

In this paper we introduced active graph grammars for modeling organizational structure dynamics in MAS. By borrowing ideas from active database theory we were able to construct a simple graphical formalism that allows us to model MAS organization in a distributed yet (expressively and semantically) powerful way. By introducing a recursive definition of organizational units, we were able to model units and agent roles on any level by using hypergraphs. It has been shown how the usual graph grammars can be extended to account for hypergraphs as well as to work in a dynamic (event-driven) environment.

Figure 2 AGRR for an Amoeba Organization



As opposed to previous studies, the approach outlined herein is holistic since all aspects of change in organizational architecture can be modeled with it: organizational structure, culture, strategy and processes; it naturally corresponds to object-oriented frameworks due to its object-oriented extensions; and allows for specifying change on higher levels of abstraction. Due to its object-oriented extensions it can be easily implemented in object-oriented logic frameworks like frame logic or description logic based systems. On the other hand, since the formalism is highly expressive, its implementation might suffer from combinatoric explosion. Still, since each AGG is local to an agent, there likely won't be complex grammars defined, but this of course has to be tested.

We believe that organizational design of MAS will benefit from a holistic approach to formalizing organizational change. Our future research will target a proof-of-concept implementation of the system in an adequate MAS development environment.

References

1. Barnatt, C. (1995), "Office space, cyberspace & virtual organization", Journal of General Management, Vol. 20 No. 4, pp. 78–91.

2. Becker, B., Beyer, D., Giese, H., Klein, F., and Schilling, D. (2006), "Symbolic invariant verification for systems with dynamic structural adaptation". In *Proceedings of the 28th international conference on Software engineering*, ICSE '06, pp. 72–81.

3. Brčić, B. (2002), "Organizacijska kultura u funkciji djelotvornosti upravne organizacije", Ekonomski pregled, Vol. 53 No. 11–12, pp. 1048–1069

4. Daft, R. L. (1992), Organization Theory and Design. 4th ed. Saint Paul etc., West Publishing Company.

5. Fischer, T., Niere, J., Torunski, L., and Zündorf, A. (2000), "Story Diagrams: A New Graph Rewrite Language Based on the Unified Modeling Language and Java". Lecture Notes in Computer Science, Vol. 1764 Chap. 21, pp. 157–167.

6. Giese, H., Burmester, S., Klein, F., Schilling, D., and Tichy, M. (2003), "Multi-agent system design for safety-critical self-optimizing mechatronic systems with uml". In *OOPSLA 2003 - Second International Workshop on Agent-Oriented Methodologies*.

7. Hernaus, T., Bosilj Vukšić, V., Pejić Bach, M. (2012), "Influence of strategic approach to BPM on financial and non-financial performance". Baltic Journal of Management, Vol. 7 No. 4, pp. 376–396.

8. Krackhardt, D., Carley, K. M. (1998), "A PCANS model of structure in organization". In: *Proceedings of the 1998 International Symposium on Command and Control Research and Technology, Evidence Based Research, Vienna, VA*

9. Lesser, V., Atighetchi, M., Benyo, B., Horling, B., Raja, A., Vincent, R., Wagner, T., Xuan, P., and Zhang, S.X.Q. (1999), "A Multi-Agent System for Intelligent Environment Control". Computer science technical report, University of Massachusetts.

10. Nagendra Prasad, M. V. and Lesser, V. (1999), "Learning situation-specific coordination in cooperative multi-agent systems". Autonomous Agents and Multi-Agent Systems, Vol. 2 No. 2, pp. 173–207.

11. Nagendra Prasad, M. V., Decker, K., Garvey, A., and Lesser, V. (1996), "Exploring Organizational Designs with TAEMS: A case study of distributed data processing". *Proceedings of the Second International Conference on Multi-Agent Systems*, pp. 283–290.

12. Nagl, M. (1979), "A tutorial and bibliographical survey on graph grammars". In Volker Claus, Hartmut Ehrig, and Grzegorz Rozenberg, editors, Graph-Grammars and Their Application to Computer Science and Biology, Lecture Notes in Computer Science, Vol. 73, pp. 70–126.

13. Parr, Rud O. (2011), "Adaptability" *Business Systems Research*, Vol. 2 No. 2, pp. 4–12.

14. Obradović, M., Dabić, M., Pejić Bach, M. (2007), "Implementation of balanced scorecard applications". In Dražen Barković, Bodo Runzheimer, editors, Interdisciplinary Management Research III, pp. 358–373. Faculty of economics, Postgraduite study organization and management & Hohchschule Pforzheim.

15. Schatten, M., Žugaj, M. (2011), "Biomimetics in modern organizations - laws or metaphors?" *Interdisciplinary Description of Complex Systems - scientific journal*, Vol. 9 No. 1, pp. 39–55.

16. Smith, B., Howard, A., Mcnew, J.-M., Wang, J., and Egerstedt, M. (2009), "Multi-robot deployment and coordination with embedded graph grammars". Auton. Robots, Vol. 26 No. 1, pp. 79–98.

17. Warnecke, H.-J. (1992), "Die fraktale Fabrik - produzieren im Netzwerk". In GI Jahrestagung, pp. 20–33.

18. Žugaj, M., Schatten, M. (2005), Arhitektura suvremenih organizacija. Varaždinske Toplice, Tonimir and Faculty of Organization and Informatics

About the Author

Markus Schatten is an assistant professor at the Faculty of Organization and Informatics holding courses in Knowledge management, Programming, Database theory, Database management system's implementation, Logic programming and Multi-agent systems. He holds a PhD in information and communication sciences (University of Zagreb, Faculty of Organization and Informatics, thesis "Programming Languages for Autopoiesis Facilitating Semantic Wiki Systems"), MSc in information sciences (University of Zagreb, Faculty of Organization and Informatics, thesis "Developing an Open Ontology of Biometrics"), and BSc in information systems (University of Zagreb, Faculty of Organization and Informatics, thesis "Neural Network based Face Recognition using 2D Images"). He authored and co-authored more then 60 scientific and professional papers under which a monograph book on modern organizational architectures. His research interest include the semantic Web, organization theory, social network analysis, multi-agent systems and Web mining.