Towards Flexible Information Architecture for Fractal Information Systems

Marite Kirikova  
Department of Systems Theory and Design  
Riga Technical University  
Riga, Latvia  
marite.kirikova@cs.rtu.lv

Abstract— In information systems development and usage the correspondence between concepts used at the business level and data structures processed at the computer systems level is to be established and maintained. The complexity of this task depends on the enterprise and information systems architectures that underlie a particular information system as well as on dynamic properties of information systems environment. In cases of heterogeneous domain ontologies and complex enterprise and information systems architectures, the fractal approach to information systems design can support balanced change propagation from business domain to computer systems domain that, in turn, allows to define and maintain flexible information architecture that supports business goals and processes. Fractal approach allows applying various systems development methods at different administrative levels of the business organization and thus can accelerate the IS development process and change management during the systems maintenance.

Keywords - information architecture; fractal systems; change propagation; systems development methods

I. INTRODUCTION

Ability to adjust rapidly to changing business needs has become one of the essential requirements for organizational information systems (IS) supporting different types of enterprises. IS development and change implementation during the maintenance period consist of particular interrelated administrative, requirements identification, software development and hardware acquisition procedures. There are several IS development methods that aim at shortening the time needed for the above-mentioned procedures, e.g., agile systems development methods [1], end-user systems development methods [2], and intelligent software systems [3]. Each of these methods has its boundaries of applicability that are characterized by intellectual resources required for the use of methods and available tools for their implementation.

As regards the organizational context one of the essential problems is that, on the one hand, centralized IS solutions are needed to ensure integrity of the enterprise, but, on the other hand, centralized solutions (1) usually are not flexible enough, (2) they require considerable administrative procedures that accompany changes in the IS and (3) new developments and changes in the IS require careful consideration of security issues in the change affected parts of the IS. In this paper, we propose an approach for information systems design and maintenance that can tolerate various developments and changes in the information system without losing its integrity and causing security threats to the centralized data resources. The approach allows applying various, most appropriate in each development/change case, systems development methods at different administrative levels of the business organization and thus accelerates the IS development process and change management during the systems maintenance. The approach suggests viewing an information system as a fractal enterprise. A fractal enterprise is an enterprise that organizes itself towards well negotiated goals allowing different development approaches in different parts of the system [4]. To achieve this, a flexible information architecture is needed that, on the one hand, ensures integrity of the system and, on the other hand, provides means for tolerating a considerable variety of information substructures and information processing sequences.

The paper is structured as follows. Section 2 describes why and how a fractal paradigm can be applied in the field of IS development and presents a definition of information architecture of fractal IS. Section 3 characterizes knowledge types related to fractal IS. Section 4 presents a variety of tolerance and change propagation approach in information architecture of fractal IS as well as includes brief comparison of the fractal systems development paradigm to object-oriented and aspect-oriented ones. Section 5 concludes the paper by a discussion of current IS implementations using the fractal paradigm and pointers to future work. Theoretical and practical issues in the paper are illustrated by examples of university IS domain.

II. “FRACITAL” IS

Flexibility is one of the systems’ features that enable their adjusting to changing environmental needs. There are different definitions of flexibility; most of them focus on the ability to respond to external changes in an appropriate period of time using a reasonable amount of resources [5, 6]. However, it is essential that flexibility possesses some degree of stability because whenever a part of the system is made flexible, some other part is made inflexible [5]. From the point of view of a fractal paradigm [4] the stable part is
perceived as a pattern inside the system that is replicated at different scales. For instance, Fig. 1 shows replication of the pattern “Educational institution” in structural units of different scale at a university. Thus, each structural unit of the university reflected in Fig. 1 has to exhibit features of an educational institution, which may include different types of teaching, research and administrative processes that are to be supported by corresponding processes of the information system. In other words, we may regard the domain system as fractal if at least one abstract system can be defined so that the whole domain system and its parts at different scales belong to the type of the defined abstract system.

It depends on the university’s policy and traditions whether the processes at different structural units are carried out in the same way or differently as well as whether databases for information storage have the same or different data structures. Regardless of those similarities or differences, there are information structures that are produced in smaller scale structural units and have to be understood by higher level structural units. This understandability is to be supported by information architecture of a fractal system.

The notion of a fractal system in this paper is used similarly to “fractal factory” introduced by H.J. Warneke [4] where the fractal is regarded as an independently acting corporate entity whose goals and performance can be precisely described. In our interpretation, the fractal system is a relatively independently acting physical or abstract system, whose goals and performance can be fully or partly described using in systems boundaries commonly understandable conceptual constructs. Those constructs form the basis for a particular information architecture utilized by fractal IS. In general, IS may be organized as a fractal system for different organizations: administratively fractal and administratively non-fractal ones. In this paper, we consider the case when the organization itself functions as a fractal system at least to some extent [7], and IS reflects the organizational fractality in the sense that it supports fulfillment of at least those goals of the fractals that are common to the whole fractal system. In-depth study of related work about the use of fractal paradigm in IS development is given in [7].

Information architecture is the term most commonly associated with websites and intranets, but it can be used in the context of any information structures or computer systems. Information architecture describes the structure of a system, i.e., the way information is grouped, navigation methods and terminology used within a system [8]. We can distinguish between visible and invisible parts of information architecture. Effective visible information architecture enables people to step logically through a system confident that they are getting closer to the information they require [8]. A well-organized invisible information architecture enables efficient information processing in software systems and in the human brain. Flexible visible and invisible information architecture enables effective information processing in heterogeneous and changing information processing environments.

In fractal information systems, information architecture has at least two orthogonal dimensions. The existence of those dimensions depends on (1) different modes of information processing (human brain, software, hardware) and (2) a particular fractal hierarchy under the consideration. Fractal hierarchy relates to the living systems theory [9] where hierarchical levels emerge for achieving higher communication efficiency. For instance, at a university, communication via fractal levels should be more efficient than direct simultaneous individual communication between all employees of the university. Both dimensions are reflected in Fig. 2, where D1 points to the dimension of fractal hierarchy and D2 points to the dimension of information processing modes. With respect to computer systems supported information transfer and processing, in fractal IS, part of the information is to be processed by software and hardware belonging to a higher fractal entity, which ensures the existence of common core information architecture and data available to all fractal entities. At the same time, each fractal entity may have independent information processing facilities. Therefore, information architecture should be flexible enough to support all information needs of all fractal entities.
management systems where lower levels are mainly information suppliers, but higher level hierarchies use the information acquired from the lower levels of hierarchy for their decision making. In fractal IS, lower levels of hierarchy are decision makers as well, and their decisions are made taking into consideration information supported by higher level fractals. This use of information is illustrated in Fig. 3. Shaded areas in the form of irregular triangles depict information acquisition space of higher level fractals. This information in a processed form is available to lower level fractal entities (shown by arrows). Taking into consideration security requirements, such information acquisition and distribution pattern requires specific data exchange and hardware access procedures, which are a matter of further research and will not be discussed in this paper.

So far we have used the term “information processing”. However, to define the model of information architecture, we need to distinguish between the terms: data, knowledge and information. There are different ways of exposing similarities and differences between those terms [10]. Nevertheless, information is commonly understood as interpreted data. To be interpreted, data should be processed by the human brain or by software. Therefore, we have chosen to model information architecture as the backbone of data processing networks.

A data processing network may be regarded as the synonym for information flow system if information flow from node A to node B is considered as data that is provided by node A and interpreted by node B. Thus, the same flow from A to B from the point of view of A may be considered as an information flow, because data that A provides is its interpretation of data received from other information processing node(s) or obtained by direct observation. The same flow may be regarded as data flow from the point of view of node B. To ensure correct interpretation (or processing) of data, both nodes (A and B) should operate in the same context and share a common ontology. According to [11, 12] the following information flows exist in fractal systems:

- information flows inside the fractal entity (for instance, information flows in Department 2.1 in Fig. 3)
- information flows between the same level fractals (for instance, information flows between Department 2.1 and Department 2.2 in Fig. 3)
- information flows between different level fractals (for instance, information flows between Department 2.1 and Faculty B in Fig. 3)
- information flows between fractal entities and external environment

Thus, information architecture in a fractal system may be presented by orthogonally decomposable sets of related elements \( \{D_i, N_i, P_i, I_i, C_{Di}, C_{Pi}\} \) where \( D \) is data acquired by knowledge processing node \( N \), \( P \) data transformation performed by \( N \) to have or to provide information \( I \). \( C \) refers to the concepts in the ontology shared by particular above-mentioned sequences such as \( C_{Di} \) is equal to \( C_{(i-1)} \). (i.e., it is shared by all upper hierarchies). Each node may be decomposed in terms of fractal hierarchy and in terms of data processing (Fig. 2 and Fig. 4). Decomposition in terms of data processing may represent both visible and invisible parts of information architecture as shown in Fig. 4.

With fractal IS each enterprise has to develop and maintain core information architecture ontologically shared by all fractal entities. In terms of service-oriented paradigm [13], the core architecture may serve as “information bus”, where all fractal entities may “plug in” for negotiation and realization of the systems common goals. The successful use of information architecture depends on the knowledge possessed by a fractal system. The types of knowledge relevant to information architecture in fractal IS are discussed in the next section.
III. KNOWLEDGE TYPES IN FRACTAL IS

A fractal information system has properties of self-similarity at different scales, self organization, goal-orientation, and dynamics and vitality [7]. These properties are ensured not only by administrative and technical organization of subsystems but also by different types of human knowledge involved in continuous data processing, information acquisition and organizational knowledge development.

Self-similarity of fractal systems often is understood as the ability of systems components to produce similar outputs from similar inputs using different internal procedures and structures [4, 11]. This means that common knowledge structures should exist to be used by different fractals of the same and different scales. Those structures may be represented by tacit and explicit knowledge where data structures in data bases and data warehouses may be regarded as part of explicit knowledge (Fig. 5).

Properties of self-organization and dynamics and vitality can be supported in several ways:

- By human knowledge at business administration and performance level
- By human knowledge at IS administration and performance level
- By software that uses built in self-organization procedures
- Different combinations of above-mentioned knowledge

It is essential to understand that IS includes not only personnel that uses data provided by computer systems, but also personnel that maintains or changes the information system. Thus, changes made to the information system by the maintenance team may be regarded as self-organization activity of the IS.

All the above-mentioned knowledge types are the context for and the “raw substance” of the information architecture of fractal IS (discussed in Section 2). Changes in any type of knowledge may lead to extensions to or changes in the information architecture in use. Most commonly changes in tacit knowledge cause changes in explicit knowledge and thus also in software and hardware of the IS. If there are changes, it is necessary to decide whether they will be local or will concern several fractal entities, or all fractals as a whole. Change tolerance and propagation possibilities in fractal systems will be considered in the next section. In the discussion the notion ‘ontology’ will be used for shared conceptual interpretation of business domain, which may or may not be explicitly documented (i.e., the ontology may be ‘tacit’). While such use of the concept is a little uncommon, we have introduced this view on ontology because of the need to take into consideration not only formally expressed but also unexpressed emergent needs of data processing that may develop only gradually and in other cases may not develop in permanent externalized conceptual structures, officially used in a business organization as a whole. Those conceptual structures, which are used relatively permanently in all organizational structures of fractal nature, form core ontology of fractal IS. This ontology is supposed to be explicitly documented using arbitrary tools for the recording of concepts and their relationships. Other relevant structures are data structures that usually form invisible (for the end-user) constituents of information architecture. They also may change when new software applications are introduced.

For flexible information architecture purposes, it is essential that at the highest level of abstraction in data structure each concept is explicitly related to the core ontology. In fractal systems, those patterns of data structures that are relevant to all fractal units form the core data structure of the system. Each data flow of information architecture processed by software therefore has to be related to the core conceptual structure of the system (Fig. 6).

In Fig. 6 a fractal entity is represented as a network of internal information processing processes (manual and performed by the software) together with the plug-ins for communicating with other fractal entities and business system environment. An essential feature of the fractal system is that each fractal unit may utilize slightly different information architectures while their integrity is supported by commonly used patterns of ontological and data structure constructs. In terms of contents of data/information flows the entity is “plugged in” also in core ontology and core data structures.
IV. VARIETY TOLERANCE AND CHANGE PROPAGATION IN FRACTAL IS

In the discussion of change propagation in fractal systems, we will compare fractal IS to centralized IS solutions. In centralized IS solutions, any change may affect all users of the IS. In fractal IS, all users are affected only when the core ontology and core data structures are changed. Those changes may be made by top-down and bottom-up change management procedures. Change propagation may involve changes in software procedures as well as changes in the information architecture of the fractal system. In the fractal system, changes are suggested by a particular fractal entity where the need for change originates because of changes in the knowledge of users or developers/maintainers of the IS as described in the previous section. Flexibility of information architecture is achieved by tolerating different uses of core information architecture at different fractal entities (Fig. 7). As a result, a considerable variety of local information architectures may be maintained while using the same pattern of core ontology and data structures at every fractal unit. For instance, several projects may use the university’s Curriculum Vitae (CV), CV of the department, and CV of the researcher while different groupings of projects (research/industrial; national/international, etc.) may be utilized at different fractal units.

Successful utilization of new developments, the decision can be made to propagate new solutions up or down in the fractal structure. For instance, two new concepts “research projects” and “industrial projects” may be added to core ontology and data organized so that information about all projects is obtained by merging data about different types of projects. A number of various appropriate methods may be used for the change propagation. The scope of methods for change propagation depends on the extended workspace development methods. Table 1 illustrates how methods for change propagation depend on extended workspace development modes. Rows of the table correspond to the modes of the extended workspace development. Columns of the table correspond to the modes of change propagation in the fractal system. Examples of systems development methods that are suitable for different development mode and change propagation mode combinations are given in the cells of the table.

TABLE I. DEPENDABILITY OF CHANGE PROPAGATION METHODS ON THE METHODS OF DEVELOPING THE EXTENDED WORKSPACE.

<table>
<thead>
<tr>
<th>Extended workspace development mode</th>
<th>Change propagation mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual*</td>
<td>Semi-automatic</td>
</tr>
<tr>
<td>Manual</td>
<td>Plan-driven</td>
</tr>
<tr>
<td>Semi-automatic</td>
<td>Plan-driven Agile</td>
</tr>
<tr>
<td>Automatic</td>
<td>Plan-driven Agile</td>
</tr>
<tr>
<td>End-user development</td>
<td>Plan-driven</td>
</tr>
<tr>
<td>By intelligent software</td>
<td>End-user development</td>
</tr>
</tbody>
</table>

*Methods are considered “manual” if they do not use software applications belonging to the fractal system.

The use of fractal IS facilitates the usage and maintenance of flexible information architectures and provides the following benefits to IS design and maintenance:

- Extended workspaces allow to develop locally solutions that later can be propagated in the fractal IS. Thus complex changes are substituted by simpler developments: (1) the solution is well developed on small scale first and (2) when changes are introduced on a larger scale – requirements for a new version of work are well developed and, thanks to similarity of ontological and data structure patterns at all fractal

Figure 7. Extended workspace in fractal IS
levels, changes can be extended to different scales of the fractal system with relative ease.

- The above-mentioned reduction of complexity enables the usage of agile, semi-automated and automatic development methods that in centralized IS case could appear to be too unreliable or too sophisticated.

- Fractal patterns of ontology and data structure ultimately require the inclusion of common systems goal-oriented concepts and data structures that help to ensure systems integrity with respect to the support of the fulfillment of systems business goals.

- Bottom-up change propagation possibility facilitates leveraging of the best practices in the business organization [14].

Fractal approach resembles to some extent object-oriented and aspect-oriented approaches. Top down change propagation may be done in a manner similar to object-oriented systems development [15], and patterns in the fractal system may be found using methods from the scope of aspect-oriented ones [16]. However, it is necessary to note that neither object-oriented approaches, nor aspect-oriented approaches per se consider organizational goal orientation at different levels of scale, the relationship of human knowledge to ontological constructs, and the relationship between ontological constructs and data structures that are essential in fractal IS. Therefore from the methodological point of view new extensions to existing methods or entirely new methodology will be needed for serial fractal IS development.

V. CONCLUSIONS

The approach discussed in this paper uses fractal paradigm in the area of IS development. Methods described in the paper are at the stages of development or tuning. A method of IS change for the best practices utilization has been developed theoretically [14], the service oriented architecture based conceptual model of fractal knowledge management system has been developed, where some practical implementations are accommodated [17]. The announcement services for the university, faculty and departmental levels of use are developed using the fractal approach. Those developments exhibit practical applicability of the fractal paradigm in IS development in general and in the development of information architecture, in particular.

Further research concerns integrated modeling of all constituents involved in information architecture: business ontology, data structures, process breakdown in fractal entities and change propagation networks.

ACKNOWLEDGMENT

The research work described in the paper is partly supported by Riga Technical University and the Ministry of Education and Science of Republic of Latvia, project No. R7390.