

# Modeling Organizational Competencies for Business Process Description and Alignment

João Pombinho, Technical University of Lisbon, Portugal, and Organizational Engineering Center, INESC-INOV, Lisbon, Portugal; E-mail: jpombinho@acm.org

Artur Caetano, Technical University of Lisbon, Portugal, and Organizational Engineering Center, INESC-INOV, Lisbon, Portugal

Marielba Zacarias, Technical University of Lisbon, Portugal, and Organizational Engineering Center, INESC-INOV, Lisbon, Portugal

José Tribolet, Technical University of Lisbon, Portugal, and Organizational Engineering Center, INESC-INOV, Lisbon, Portugal

## ABSTRACT

*Business activities are performed by human or automated actors. For the organization to adapt to changes it must be able to understand how and why actors are related to and assigned to processes. This requires a consistent representation of the services required by the organization's processes and those provided by its actors. This paper focuses on defining the concepts that allow to structurally align human actors and business processes through the description of the organizational competencies required to perform processes' activities. These structures are used within a marketplace-based model, supporting the management of actors and activities according to the supply and demand of competencies.*

## 1. INTRODUCTION

Competency-based management is becoming an object of growing interest as its importance is recognized from a strategic perspective. It aims leveraging the competitive advantage of knowledge-based organizations by observing individuals as strategic assets and explicitly integrating them into the organization's business strategy and processes. This approach has led to the development of several methods and systems (Woodruffe, 1993; Stone, 1995; Hamel & Prahalad, 1990; Fletcher, 1995).

Competency-based management provides individuals with a set of task definitions and objectives, linking organizational and personal objectives. It facilitates process analysis and standardization inside and outside the organization's boundaries, making recruitment and compensation systems fairer and more open. However, competency related information is prone to become obsolete due to organizational changes, so it can become expensive and time consuming to keep this information updated (Chartered Institute of Personnel and Development UK, 1995). Competency management support systems play a role in different organizational activities, such as expert finding, personnel recruitment and project management. These activities customarily involve human judgment to classify the skills held by workers, to evaluate the degree of competency and keeping up-to-date profiles. To facilitate the management of competencies and to minimize the subjectivity of human evaluation, several approaches put forward the use of information technologies to facilitate tasks such as contextualizing the communication between actors (Yimam, 2000), managing skills and activities within teams (Gronau & Uslar, 1994) and using groupware to support information sharing (Johansen et al, 1991).

Nevertheless, these solutions focus on supporting the operational phases of a business process and few assist the identification and selection of actors before the actual commitment to carry out its activities. This operational focus also means that skill management systems often do not relate the skill information pertaining to the actors to the activities' requirements as derived from the organization's process models. Consequently, information on the organization's competencies cannot be directly traced to its business processes. This hinders skill management from a process perspective and promotes the existence of mismatches within the enterprise architecture description.

This paper outlines a framework to overcome these issues by extending previous work on business process and actor modeling (Caetano, Zacarias, Rito Silva, &

Tribolet, 2005; Neves, Caetano, Vasconcelos, & Tribolet, 2001; Spencer, Spencer, & Signe, 1993; Zacarias, Caetano, Pinto, & Tribolet, 2005) with a set of concepts that allow representing and evaluating actors and competencies in the context of a business process. The remainder of this paper is structured as follows: section 3 describes our proposal on competency modeling, section 4 describes how to integrate this model into an organization using the concept of competency marketplace and section 5 sets out the conclusions and describes future work.

## 2. RELATED WORK

Despite the research on competency-based management, representation of competencies is not fully addressed in nearly all business process modeling standards. IDEF0 (Scheer, 1999) is a method of modeling organizational decisions and activities through functions, inputs and outputs but it does not provide the means to represent how activities are performed by actors. BPMN (Business Process Management Initiative, 2004) is a notation that focuses on describing business process flows. Participation in a process is represented through swimlanes that relate activities to its performer, thus mapping responsibilities at a high level of abstraction. There is no means to specify the requirements an actor must fulfill to execute an activity. Other approaches, such as IDEF3 (Mayer et al, 1995) and RADs (Ould, 1995) focus on describing process flow but overlook the specification of actor competencies as well as activity requirements.

There is a large amount of research on competency modeling outside the scope of business process modeling. The "competency movement", credited to McClelland (1973), uses the concept of competency to classify human actors and to relate them with the ability of performing a task. Spencer (1993) defines competency as a characteristic of an individual that is causally related to effective and/or superior performance in a situation. *Underlying characteristic* means the competency is part of a person and can be used to deduce behavior, being part of what Davenport (1997) classifies as tacit knowledge. *Causally related* means that a competency causes or predicts behavior. *Criterion referenced* means it is possible to quantify the performance of a competency and implies a causal relation between purpose and result.

Defining the granularity of competency representation is equally important as a high-level representation will not provide enough information, while if it is too detailed, the entire representation process may become compromised, as it is effort and time-consuming (CIPD UK, 2005). Competencies are usually represented as hierarchical structures or competency trees. An example is the "body of knowledge", in which competencies are classified according to specific areas of knowledge, such as IEEE's Software Engineering Body of Knowledge (Abran & Moore, 2004). Lang and Pigneur (1999) also propose using a hierarchical structure to represent competencies since it simplifies processing when compared to the mining of textual descriptions. It is also easier for people to identify competencies, the expectations of the organization and possible gaps. Pigneur proposes four competency categories as starting points: Enabling Technologies, Field Experience, Knowledge and Personal Traits. However, the semantics associated with these hierarchies is static by design, relying on a specific functional domain classification to describe the competencies.

A pragmatic approach is found at CommOnCV (Harzallah & Lecrère, 2002), which models *curricula vitae* during the recruitment process. Competencies are represented as annotations derived from each *curriculum* and are represented using RDF (Brickley et al, 2000) or DAML+OIL (Euzenat, 2002). The annotations are based on a particular competency model, created within a specific knowledge area.

Competency trees present problems related to flexibility and reusability. These trees are usually deep, with the functional categorization specified in its upper levels; the description of a competency is not easily decomposable, leading to the dispersion of concepts between nodes. As a result, inference is limited as the concepts are not structured and connected to each competency as individual components. Reusing competencies in different contexts is hindered as functional categorization is embedded at the tree's upper levels.

### 3. COMPETENCY MODELING

The approach relies on defining the structure of competencies so they can be aggregated or composed at design time. This structure can be instantiated at run time enabling inference and analysis. These three phases separate the definition of individual competencies from their hierarchy as well as from the runtime information that will instantiate the structures.

The rationale behind separating the definition of a competency from its hierarchy is that it is not possible to anticipate the needs of an organization in terms of its competency categorization. If the definition and the corresponding hierarchy were entwined, then the competency would be defined in the scope of a single organizational context, making difficult specifying the same competency across multiple contexts. This is often the case where different organizational units use different classification schemes for the same competency. Thus, this separation aims at maximizing the reusability and flexibility of the structure.

#### 3.1 Competency Definition

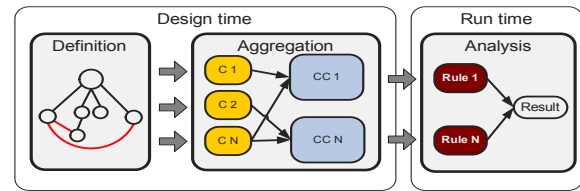
Defining a competency requires specifying or reusing the concepts that are necessary for its description and, second, specifying the structure that will hold these concepts and their relationships. Creating a structure to relate competencies is complex since it is not straightforward to model a potential large set of disparate individual features that influence the performance of a given task, such as knowledge, practical experience, psychological and social factors, context and motivation.

Before detailing how competencies are defined, it is important to disambiguate some fundamental concepts. A *skill* is the ability to perform a particular activity, while *knowledge* is related to the information needed for enabling the skill. As an example, while there are skills needed to carry out network troubleshooting, there must be knowledge about physical media and network protocols for that skill to be available. The improvement of skills and knowledge through experience, training or education leads to *occupational competency* (Jones & Bearly 2006). This means a single competency may encompass multiple skills. A *competency* relates to the behavior that individuals must perform as input into a situation while achieving some level of performance. It represents the association of knowledge and action. This means, on the one hand, that competencies are based on knowledge, and, on the other, that for a competency to be recognized as such, it is necessary to demonstrate the capability of giving use to that knowledge through an action that brings value to the task.

A competency is an expression defined through the aggregation of elementary *nouns* and *verbs* (e.g. "coding a search algorithm in Lisp", "coding a web service in C#"). The concepts may be related or dependent (e.g. Lisp and C# are programming languages with different programming paradigms). To deal with the representation of such features, we define a set of layered hierarchic structures that specify the competency's concepts regardless of their usage context. This type of structure allows concepts to be reused while defining isolated competencies.

Actors and activities relate to competencies whose meaning depends on their usage context. These competencies may also be structurally related. To cope with this, competencies are aggregated as coherent groups comprising individual competencies. Separating the aggregation of competencies from their definition enables their rearrangement according to usage context without disrupting the corresponding definition. Groups of aggregated competencies can be bound to actors and activities, specifying supply and demand of competencies.

Figure 1. Design-time and run-time modeling phases



#### 3.2 Structuring Competencies

Competencies are often structured hierarchically. As previously discussed, such an approach offsets representation simplicity and expressiveness. However, most approaches rely on static classification schemes that prove difficult to adjust to the organization dynamics and its environment. Furthermore, hierarchical representations are defined in the scope of some functional context, so it is not straightforward to detach a structure from its context and keep its semantics. Therefore, as it is not reasonable to assume that functional contexts are shared throughout the organization, the representation structure must be able to separate competency definition from its categorization.

To overcome these issues and those identified in section 2, we propose using a multi-dimensional tree structure, where each dimension relates the element with its usage context. This allows multiple views on the same object while keeping its uniqueness. Such structure can be perceived as a regular hierarchical tree where a number of contextual layers can be superimposed.

Nodes in the same dimension or layer are connected using intra-level links allowing relating elementary concepts. Inter-level links associate nodes from different dimensions allowing the specification of complex concepts. The semantics of this type of link is defined by ontological relations, such as "belong-to", "uses" or "enables".

The diagram in Fig. 2 depicts three layers (A, B, C), each defining elementary concepts in a given context. Using inter-level links (depicted as dotted arcs) to associate a number of elementary concept nodes in different contexts enables a competency defined as  $\{ \langle A1, B1 \rangle, \langle A1, C1 \rangle, \langle B2, C1 \rangle \}$  to be specified.

As an example, Fig. 3 shows a number of elementary competencies related to software development.

Fig. 3 specifies the following set of competencies:

- Develop Software in the Banking Business Domain
- Software Design using both Imperative and Object Oriented Paradigms
- Code Software both in C# and Lisp Programming Languages
- Know Java Programming Language
- Administer Linux Operating System
- Use any Operating System
- Use the Visual Studio IDE

#### 3.3 Competency Aggregation

Competency aggregation allows specifying multiple contexts through the specification of viewpoints. After being defined, competencies can be composed so they can be handled as a unit. The meaning of the competency "programming" can change according to the perspective. One definition could be the knowledge of a given programming language. Nevertheless, it can also mean the knowledge

Figure 2. Elementary concept structure (left). Competency definition (right)

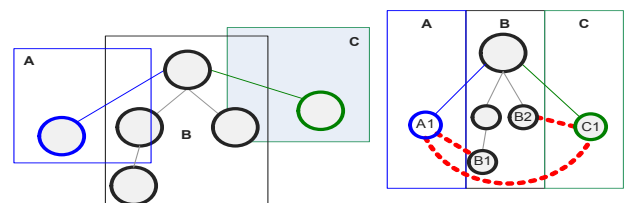
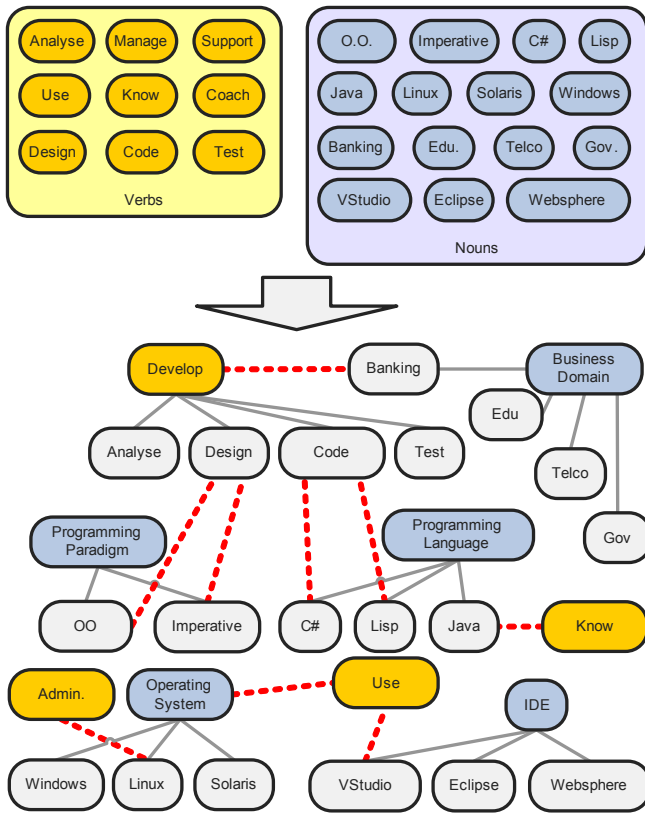


Figure 3. Elementary concepts and competency definition



in programming paradigms and programming languages. These two contexts can be specified as distinct viewpoints.

Different observations can be taken from the value of a competency element by changing the hierarchy while using the same inputs. It becomes necessary to ensure the independence between the viewpoint and the competencies themselves. The latter cannot be dependent on the categorization chosen by the organizational unit: they are part of the individual and exist regardless of context. Therefore, we propose using two layers: one featuring objective information on the competencies and other with their classification and context information. Thus, the upper level hierarchy corresponds to categorization and views, allowing the creation of structures that can reuse previously defined components. This approach is flexible since it enables the creation of a semantic level above elementary competencies by separating both and making individual's competencies context invariant.

Figure 4. Competency aggregation

**C# Developer specialized in Banking**

| Use     | Develop | C4   | Banking |
|---------|---------|------|---------|
| C1      |         |      |         |
| VStudio | Design  | Code |         |
|         | C2      | C3   |         |
|         | OO      | C#   |         |

Other aspect has to do with granularity: a monolithic structure is inadequate to representing competencies, so it is useful to create smaller trees, particularly in the case of relationships between nodes that arise from their own definition as concepts. As an example, take imperative, object-oriented and functional while programming paradigms. This classification can be considered context insensitive. These smaller trees are also known as “concept islands” (Lau & Sure, 2002), self-contained concept groups from sub-domains of a larger domain. Since these islands are interlaced, it makes sense to include them into multiple trees, which implies a complex network of node relationships. One approach is combining the concept islands into a single tree. This introduces complexity at ontology design time. Worse, it limits the expressiveness and flexibility of the structure. Therefore, we propose representing the concept island as multiple trees and keeping them separately in a repository. This approach enables these structures to be used as modular building blocks when composing larger trees. It also supports the flexible rearrangement of blocks, without breaking existing connections, i.e., maintaining the connections traceable from runtime back to the atomic elements that form a competency.

To illustrate the competency aggregation step, Fig. 4 shows the result of competency composition. The original structure does not require revision. Aggregated competencies defined in this step may also be used to map roles to competencies.

### 3.4 Competency Analysis

The main goal of this phase is instantiating the competency structures according to the competencies provided by the organization’s human actors. While the first two phases are enacted during design time, this phase relates activities and actors at run time, allowing the assessment of competency supply and demand.

Competency analysis requires a set of propagation rules to be defined between different hierarchy levels. Both the rules and the hierarchy itself are part of the classification scheme used by an organizational unit and should not be mistaken for the definition of the competencies. The latter should be independent of the analysis phase as otherwise competencies will not be able to cross-organizational boundaries.

Propagation rules describe how a node’s information can be computed from its child nodes. They are defined bottom-up and are related to non-leaf nodes. Leaf nodes are instantiated with values that are propagated up to the root node using the rules on each intermediate node. A rule can include operations, such as logical expressions, weighted averages and threshold definitions. This enables real-time propagation of values, as opposed to static assignment where the connection to the original structure is lost from the moment that the first values are obtained.

When matching competency supply and demand there are important questions about the abstraction level or granularity used on both sides of the negotiation. If the supply is specified in more detail than the demand, simple bottom-up derivation is sufficient to match both sides. However, in the opposite case, it is necessary to request a further specification to the entity responsible for the offer. The updated specification will be included in the existing model as a supplementary layer.

While the most visible aspect of inference is performed at competency level, its application in that case is straightforward. The following example focuses on the application of rules to the relationships between elementary competency concepts. Fig. 5 depicts a scenario where a programming competency depends 60% on programming language knowledge and 40% on programming paradigms. As the programming paradigm is not a leaf node, another rule must specify how

Figure 5. Example of propagation rules

|         | Programming Paradigms | Programming Languages |
|---------|-----------------------|-----------------------|
| Average | 40%                   | 60%                   |
| Func    | Imperative            | OO                    |

to compute its value. In this case, a non-weighted average of its descendents has been used.

A fundamental aspect of the proposed framework is the traceability across the three phases. It enables the identification of a specific atomic element that composes an aggregate competency instantiated at runtime. In turn, low-level syntactic translation mechanisms create the conditions to analyze competency elements that would otherwise be obscured by their macro description. The competencies are expandable down to leaf-level, therefore representing the most objective information possible about the organization's human actors. So, competencies can become as independent as desirable from their organizational classification and activity-specific arrangements. As a result, competency movement across different units implies ontological agreement only at the lowest hierarchical level.

#### 4. THE COMPETENCY MARKETPLACE

The primary goal of representing competency supply and demand is allowing an organization to find, schedule and manage suitable teams of actors to perform activities. Our approach follows the paradigm of a marketplace transaction. This facilitates the management and tracking of competencies from an organizational perspective while promoting actors to develop their own competencies. It also facilitates correcting the gaps between the actual requirements of an activity and those specified in the process models through market rules. It is important that all parties in the marketplace share a common vocabulary and semantic model to make the negotiation possible. Such shared knowledge should be iteratively built and continuously updated. The market dynamics enables the runtime evaluation of the negotiated work, thus offering feedback for process improvement.

The implementation of a competency marketplace involves a specific process, which goes through the stages of Information, Negotiation and Settlement (Lang & Pigneur, 1999). The Information stage consists in representing actual supply and demand. This is specified at design time during the Definition and Aggregation phases of the framework. The next step is finding the best candidate that fulfills the requirement of an activity. This is accomplished by searching the supply-side representations using inference and propagation mechanisms on the hierarchical structures, instantiated with the proficiency values associated with each actor. The result is a set of actors and the corresponding evaluations computed from the similarity level between the required and provided competencies.

After identifying a potential group of actors able to perform the task, the results must be categorized according to existing work scheduling plans that limit their availability. The results also need to consider business rules that may constrain the process (e.g. actor B and C can not participate in the same activity instance) and the type of results (e.g. the result must be a team of two people). The information resulting from this step can be used to facilitate team formation.

Team formation corresponds to the negotiation between the transaction participants. Each individual actor, the team as a whole and the activity owner settle the contract details, making explicit the assignment and scheduling terms. A contract is defined after the negotiation phase is complete, specifying the details, thus binding actors to an activity through their competencies along the settlement step.

Implementing the contract means performing the activity's tasks. The execution phase can be evaluated during its performance or *a posteriori* using metrics whose goals are measuring how the contract was carried out.

This approach promotes an actor to be actively involved in the management of her own competencies. This influences her evaluation criteria and, as a result, how management observes her performance. It is in the best interest of the actor to be properly represented in the model, since an accurate description will facilitate her scheduling to the tasks included on her competency pool. It will also enable the identification of gaps inhibiting good performance, calling for specific training.

#### 5. CONCLUSIONS AND FUTURE WORK

The concept of competency is fundamental to align actors and activities within business processes. It enables an organization to understand and evaluate what is required by its processes and what is provided by its actors. It also provides a means to make competencies visible to all process stakeholders, including its performers, designers and owners, allowing mismatches to be identified and continually adjusted. Provided the different stakeholders see the benefit of accurately representing the competencies they are responsible for, the marketplace will converge to a state that represents the actual supply and demand of competencies within the organization.

To validate this approach, we have developed a web-based system that implements the concepts outlined in this paper, which is currently being evaluated in real organizations. It allows employees advertise their competencies and process owners and designers to specify activity requirements and evaluate the performance of actors. The system supports the dynamic aspects of actor scheduling, allowing the search of competencies, assisting the process of team formation and evaluating the scheduling results.

The establishment of trust relationships is crucial and the integration into the Semantic Web will enable the creation of shared repositories of services. The unification between competencies and technological services is subject of our future work. We are currently extending the framework to include the representation of the services provided by information systems and other business process support systems. This will enable defining alignment metrics between the organization's service providers and its processes.

#### REFERENCES

- Abran, A., & Moore, J. (2004). Guide to the Software Engineering Body of Knowledge, IEEE.
- Brickley, D., et al. (2000). RDFS: Resource Description Framework Schema Specification 1.0.
- Business Process Management Initiative. (2004). Business Process Modeling Notation (Version 1.0).
- Caetano, A., Zacarias, M., Rito Silva, A., & Tribolet, J. (2005). A Role-Based Framework for Business Process Modeling. In HICSS 38, IEEE.
- Chartered Institute of Personnel and Development, UK (2005). Competency and competency frameworks.
- Davenport, T., & Prusak, L. (1997). Working Knowledge: How Organizations Manage What They Know. Harvard Business School Press: Boston.
- Euzenat, J. (2002). Research challenges and perspectives of the semantic web. EU-NSF Strategic Workshop.
- Fletcher, S. (1997). Competence-based assessment techniques. Kogan Page, London.
- Gronau, N., & Uslar, M. (2004). Integrating knowledge management and human resources via skill management. In Proceedings IKNOW '04.
- Hamel, G., & Prahalad, C. (1990). The core competency of the corporation. Harvard Business Review.
- Harzallah, M., & Leclère, M. (2002). CommOnCV: Modelling the Competencies Underlying a Curriculum Vitae. SEKE'02.
- Johansen, R., et al. (1991). Leading Business teams: How Teams Can Use Technology and Group Process Tools to Enhance Performance. Addison-Wesley.
- Jones, J., & Bearley, W. (2006). Developing 360° Leadership-Feedback Instruments. Organizational Universe Systems.
- Lang, A., & Pigneur, Y. (1999). Digital Trade of Human Competencies, Proc. HICSS-32.
- Lau, T., & Sure, Y. (2002). Introducing ontology-based skills management at a large insurance company. In Proceedings of the Modellierung 2002.
- Mayer, R. et al. (1995). IDEF3 Process Description Capture Method Report. Knowledge Based Systems Inc.
- McClelland, D. (1973). Testing for competence rather than for intelligence. American Psychologist, Jan;28(1):1-14.
- Neves, J., Caetano, A., Vasconcelos, A., & Tribolet, J. (2001). Integrating Knowledge Into Business Processes. ICEIS 2001.
- Ould, M. (1995). Business Processes, Modeling and Analysis for Reengineering and Improvement. John Wiley & Sons.
- Scheer, A. W. (1999). Business Process Modeling. Springer Verlag.
- Spencer, L., Spencer, S., & Signe, M. (1993) Competency at work: models for superior performance. John Wiley & Sons, Inc, New York.
- Stone, F. (1995). New Definition of Corporate Competencies. Management Review, vol. 84, no. 6.
- Woodruffe, C. (1993). What is meant by a competency? Leadership & Organization Development Journal, vol. 14, no. 1.
- Yimam, D. (2000). Expert finding systems for organizations: Domain analysis and the memoir approach. In Beyond Knowledge Management: Sharing Expertise. MIT Press.
- Zacarias, M., Caetano, A., Pinto, S., & Tribolet, J. (2005). Modeling Contexts for Business Process Oriented Knowledge Support. In Knowledge Management for Distributed Agile Processes, Springer-Verlag.

0 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: [www.igi-global.com/proceeding-paper/modeling-organizational-competencies-business-process/33183](http://www.igi-global.com/proceeding-paper/modeling-organizational-competencies-business-process/33183)

## Related Content

---

### Swarm Intelligence for Automatic Video Image Contrast Adjustment

RR Aparna (2016). *International Journal of Rough Sets and Data Analysis* (pp. 21-37).

[www.irma-international.org/article/swarm-intelligence-for-automatic-video-image-contrast-adjustment/156476](http://www.irma-international.org/article/swarm-intelligence-for-automatic-video-image-contrast-adjustment/156476)

### Lack of Characteristics Management Causing Biggest Projects Failure

Loredana Arana (2018). *Encyclopedia of Information Science and Technology, Fourth Edition* (pp. 5650-5659).

[www.irma-international.org/chapter/lack-of-characteristics-management-causing-biggest-projects-failure/184265](http://www.irma-international.org/chapter/lack-of-characteristics-management-causing-biggest-projects-failure/184265)

### Automated System for Monitoring and Diagnostics Pilot's Emotional State in Flight

Tetiana Shmelova, Yuliya Sikirdaand Arnold Sterenharz (2021). *International Journal of Information Technologies and Systems Approach* (pp. 1-16).

[www.irma-international.org/article/automated-system-for-monitoring-and-diagnostics-pilots-emotional-state-in-flight/272756](http://www.irma-international.org/article/automated-system-for-monitoring-and-diagnostics-pilots-emotional-state-in-flight/272756)

### Mapping the Chromosome through a Novel Use of GIS and Spatial Analysis

Jane L. Garb, D. Joseph Jerry, Mary J. Hagenand Jennifer Friderici (2015). *Encyclopedia of Information Science and Technology, Third Edition* (pp. 5573-5583).

[www.irma-international.org/chapter/mapping-the-chromosome-through-a-novel-use-of-gis-and-spatial-analysis/113011](http://www.irma-international.org/chapter/mapping-the-chromosome-through-a-novel-use-of-gis-and-spatial-analysis/113011)

### Reasoning on vague ontologies using rough set theory

( ). *International Journal of Rough Sets and Data Analysis* (pp. 0-0).

[www.irma-international.org/article//288522](http://www.irma-international.org/article//288522)