

A Concept of Configuring Flexible Application Systems for Business and Administration Processes

Tobias Grollius, Erich Ortner

Department of Business Information Systems
Technische Universität Darmstadt
Hochschulstr. 1, D-64289 Darmstadt
{grollius, ortner}@winf.tu-darmstadt.de

Abstract: Business processes change with time. Component-based application systems offer the flexibility to effectively support frequently changing processes as they can be easily assembled of component variants most suitable to accomplish a task in a certain moment. This paper introduces two extreme types of configuring flexible application systems: pre-configured application systems and just-in-time configured application systems. As a prerequisite to component-based applications, a component repository and its meta schema are presented. Workflow Management Systems are proposed to control – beside the business processes – the construction processes.

1 Introduction

Reengineering and modifying business processes implies changes of the process-supporting software applications at the same time. Component-based software construction is a promising approach to build flexible and dynamic applications. Instead of customizing an application or adapting processes to standard software, applications are built of components most suitable for the purposes of the application and the user requirements at the time of the application usage. Adapting software is replaced by selecting variants of components, provided that a high number of variants is available. If there is the need to change the business processes and therefore the supporting applications, components can be easily exchanged with more suitable variants, added to the application to fulfill a new task in a process, or removed because it is obsolete in a changed business process. In the following we present prerequisites, a method and tools as part of this approach.

2 Types of Configuring Component-Based Application Systems

Flexibility is inherent to applications built of loosely-coupled components. But applications are more or less flexible depending on the point in time, when the components are assembled and finally used to execute tasks in a business process. In the following, two extremes are presented in order to define the boundaries of the field of configuring flexible application systems.

2.1 Pre-configured Application Systems

Commonly, a component-based application is assembled entirely before its first usage. Based on a specified composition plan, suitable components are searched in a component repository and, in a second step, are selected to be included in the application. Missing components have to be implemented or bought on component-marketplaces and added to the component repository. When all necessary components are gathered, they are assembled to the application. After the completion of the task of the application, it is deconstructed.

The conclusive characteristic of this type of configuring is that all components are known explicitly and composed to the application before usage-time. Because of that, the resulting application system is called pre-configured.

2.2 Just-in-time Configured Application Systems

The notion of a just-in-time configured system denotes a set of components, each of which isn't actually selected until the time of its use. A component configuration is assembled depending on the situation in the moment of use by selecting suitable components from the component-repository and connecting them to an application system to accomplish a certain task. Furthermore, the application system never exists in its entirety. Only the necessary components that are supporting or executing an activity of the business process in a certain moment are known and selected. As soon as a component has completed its (sub-)task, it is removed from the application system.

As the application system is only partly assembled as far as needed during the execution of a business process, changes of the latter are followed immediately by the supporting application system. Therefore it offers a significant higher degree of flexibility in comparison to the classical type of configuring discussed before.

3 Managing the Software Resource

To handle the paradigm shift in the fields of software that is due to component orientation, there are further tools, such as organizing trade in software components in electronic market places, or developing comprehensive repository structures for the life-cycle management of component-based application systems.

In the process of building up such repositories, it is important to ensure the constant description of the components through all life-cycle phases, from the specification of requirements, through their design and configuration and through operation to removal or replacement by other components.

For integration reasons it is additionally necessary to describe in a repository [Or05a, Or05b] the development languages that are used as well as the components and application systems developed with these languages.

Fig. 1 shows the meta schema (documentation structure) of such a universal repository in a simplified way as it is discussed in detail in [Or05b].

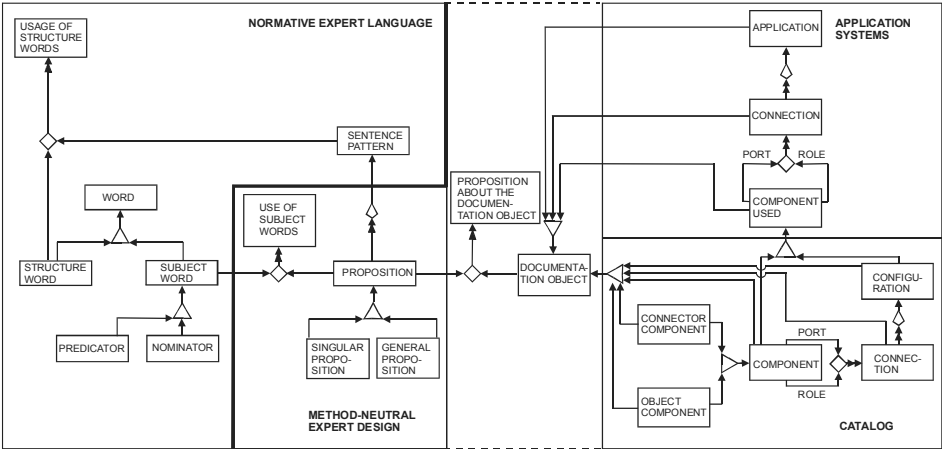


Fig. 1: Meta schema of a component repository

4 The Role of Workflow-Management-Systems

Workflow-Management-Systems (WfMS) control, support and monitor workflows as partly-automated business processes [JB96, AH02]. Their traditional area of application is office work and its automation. But they are also an essential part of a middleware architecture called E-NOgS (Electronic New Organon {Service, Server, Servant}) that consists of functionally classified and specified reusable components for the development of web applications [Or05a]. These components are differentiated in system components implementing generic functions (e.g. database management or workflow management) and expert components covering a specific functionality such as accounting components.

Besides and complementary to the control of a workflow or business process, a WfMS can also control the process of configuring flexible application systems. The configuration process should be considered a "help process" of the "main process" (Fig. 2). Regarding the operational perspective of workflow management [JB96], the WfMS controls the software application configuration process, which includes resolving corresponding variant parts lists as plans for a component configuration [Or05a]. As afore-mentioned, WfMS can be used to control the configuration of application systems that are intended to support frequently changing business processes. Are WfMS really flexible enough at all to deal with dynamically changing business processes? In [Aa99], an approach to flexible WfMS is discussed, which is inspired by variant parts lists of sub-processes. To use variant parts lists as a means to plan flexible application systems – and not only component-based software applications or processes – seems to be a promising approach.

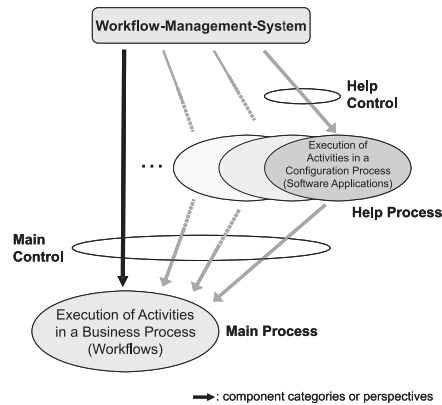


Fig. 2: WfMS used to control configuration processes

5 Future Work

Building upon an already available component repository, the presented concept of configuring flexible component-based application systems is currently being implemented prototypically in compliance with the before-mentioned E-NOgS middleware architecture. In a further stage of extension, this new provisionally named "application management system" should be able to cover not only software components but also additional component categories such as processes, knowledge, machines/hardware or employees. This long-term goal should be preceded by further investigations on suitable catalogs for these component categories as well as detailed reflections on the efficiency of component-based application systems. Furthermore, research on different types of configuring flexible application systems (in between the both mentioned in section 2) has to be done. In a frequently changing business world, the potential of flexible component-based application systems could be a valuable achievement.

Bibliography

- [Aa99] van der Aalst, W.M.P.: Flexible Workflow Management Systems – An Approach Based on Generic Process Modells. In (Bench-Capon, T.; Soda, J.; Tjoa, A.M. Eds.): Proc. 10th Int. Conf. on Database and Expert Systems Applications (DEXA'99), Florence, 1999, LNCS 1677. Springer-Verlag, Heidelberg, 1999; pp. 186-195.
- [AH02] van der Aalst, W.M.P.; van Hee, K.M.: Workflow Management – Models, Methods, and Systems. MIT Press, Cambridge, 2002.
- [JB96] Jablonski, S.; Bussler, C.: Workflow Management – Modeling Concepts, Architecture and Implementation. International Thomson Computer Press, London, 1996.
- [Or05a] Ortner, E.: Sprachbasierte Informatik. EAGLE-Verlag, Leipzig, 2005.
- [Or05b] Ortner, E.: Component-Based Application Architecture for Enterprise Information Systems. In (Härder, Th.; Lehner, W. Eds.): Issues in Modern Data Management, LNCS. Springer-Verlag, Heidelberg, 2005, to be published.