

# APPLYING OF COMPONENT SYSTEM DEVELOPMENT IN OBJECT METHODOLOGY, CASE STUDY

Milan Mišovič, Oldřich Faldík

Received: June 28, 2013

## Abstract

MIŠOVIČ MILAN, FALDÍK OLDŘICH: *Applying of component system development in object methodology, case study*. Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis, 2013, LXI, No. 7, pp. 2523–2531

To create computarization target software as a component system has been a very strong requirement for the last 20 years of software developing. Finally, the architectural components are self-contained units, presenting not only partial and overall system behavior, but also cooperating with each other on the basis of their interfaces. Among others, components have allowed flexible modification of processes the behavior of which is the foundation of components behavior without changing the life of the component system. On the other hand, the component system makes it possible, at design time, to create numerous new connections between components and thus creating modified system behaviors. This all enables the company management to perform, at design time, required behavioral changes of processes in accordance with the requirements of changing production and market.

The development of software which is generally referred to as SDP (Software Development Process) contains two directions. The first one, called CBD (Component-Based Development), is dedicated to the development of component-based systems CBS (Component-based System), the second target is the development of software under the influence of SOA (Service-Oriented Architecture). Both directions are equipped with their different development methodologies. The subject of this paper is only the first direction and application of development of component-based systems in its object-oriented methodologies. The requirement of today is to carry out the development of component-based systems in the framework of developed object-oriented methodologies precisely in the way of a dominant style. In some of the known methodologies, however, this development is not completely transparent and is not even recognized as dominant. In some cases, it is corrected by the special meta-integration models of component system development into an object methodology.

This paper presents a case study applied to the process management fragment of a human resources HR (Human Resources) domain in a small manufacturing business enterprise, which confirms the success of the meta-model implementation mentioned in the contribution (Mišovič, Faldík, 2013).

object methodology, meta-model, domain HR, logical, design, physical architecture

## METHODS AND RESOURCES

### 1 Overview of relevant steps of object-oriented Methodology and meta-model “Applying of Component system Development in object Methodology”

The text of this chapter presents an outline of mixing workflows, eventually, process steps, between the object methodology and meta-model

“Applying of Component system Development in object Methodology” Thus the following chapters will present mainly resulting output meta-model artifacts.

In Chapter 2 the composition and administration of the problem domain HRM (Human Resource Management) is valid for a medium and large business manufacturing company. Given the scale of this domain, we have chosen to reduce the domain to a small business manufacturing

I: Meta-model, Object methodology – workflow and artifacts

Workflow		Output artifacts	
O M	Contextual analysis	Subsystems draft of the problem domain: <b>Subsystems of a problem domain</b>	
	The design process view	Process diagrams of subsystems	
M M	Treatment of diagram <b>Subsystems of a problem domain</b> to the <b>Logical architecture diagram</b>	Logical architecture diagram	LOGICAL ARCHITECTURE
O M	Requirements	Making <b>Use case diagrams</b> and <b>Activity diagrams</b>	
	Analysis of the requirements implementation	<b>Analytical Class diagrams</b> and <b>Analytical sequence diagrams</b>	
	Draft of implementing classes and sequence diagrams (behavior of subsystems)	Design implementation of <b>Analytical Class diagrams</b> and <b>Analytical sequence diagrams</b> . The result is called: <b>Internal behavior of subsystems</b>	
M M	To map subsystems to components, based on the <b>Logical architecture diagram</b> and the <b>Internal behavior of subsystems</b>	<b>Working scheme</b> of Component system for design architecture	DESIGN ARCHITECTURE
	Finding connections between components, based on diagrams: <b>Subsystems of problem domain</b> , <b>Logical architecture diagram</b> and <b>Process diagrams of subsystems</b>	Diagram of <b>Component system for design architecture</b>	

Legend: OM.....object methodology MM.....meta-model

company, which does not degenerate the concept of target computerization. Selected sets of processes will be described briefly; nevertheless the detailed process description will be given only for a sample set of Selection procedure.

The resulting component system will be a two-layer system. The first layer is its fragment (components and their interconnections) based on all five selected sets of processes for a small business. The second layer should comprise five fragments of components based on processes of selected sets. Workflows of the meta-model, for the fragments design of the second component layer, are exemplarily used only on a set of Selection procedure. The previous table illustrates a summary of relevant workflows of the meta-model and object methodology.

For simplicity, we focus only on the formation of following output artifacts:

- *Subsystems of a problem domain.*
- *Process diagrams of subsystems.*
- *Logical architecture diagram.*
- *Component System for design architecture.*

So, we skip many diagrams for *Internal design behavior of subsystems of a problem domain* and we are going to devote ourselves only to the *Selection procedure* subsystem.

## 2 Description of Human resource management

One of the primary duties of each company is management of HR. Scope of processes in this area is different not only in small, medium and

large enterprises, but also in different types of organizations.

The previous figure shows by (Sod – Klč, 2010) just thirteen relevant groups-subsets of processes in HR, the main characteristic of which is common to all groups living in mutual responsibility.

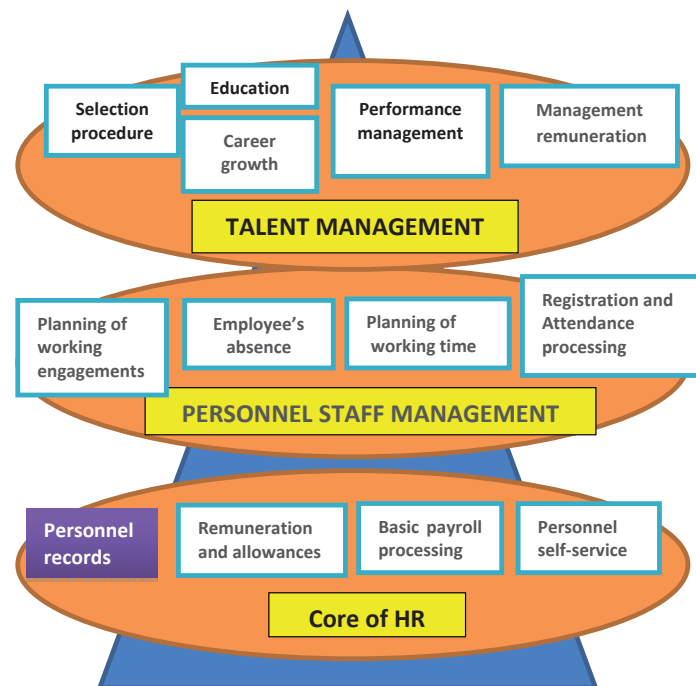
For example, small and medium enterprises with business manufacturing character, are only oriented on the basic computerization of payroll processing, simple personnel records, supplemented by calculation of missions. That is sufficient for them. On the other hand, businesses that consider systematic work with human resources to be necessary activity to maintain their competitiveness pass to the highest degree of computerization of all HR activities.

These sets are grouped into three sections: Talent Management, Personnel staff Management and Core of HR. It is understandable that at the advanced company all groups are used and individual processes are thoroughly described in the company legislative documents and the responsibility for their management is determined. It is clear that systematic personnel work in the company is slowly and systematically enforced. There is practical knowledge that its fulfillment brings the enterprise high employees' loyalty and consensus with the strategic objectives of the company.

## 3 HR process logic in a small business enterprise

It is thought that for HR management for small manufacturing business enterprises following personnel processes are sufficient:

- *Personnel records.*



1: Overview of HR sets of processes. Taken from (Sod – Klč, 2010) and modified

- Basic payroll processing.
- Employee's absence.
- Selection procedure.
- Planning of working engagements.

These sets that are forming a separate fragment, will be modeled and an information model will be created by applying the object-oriented methodology and meta-model for the development of component-based integration system for the target software.

Let's explain briefly the activities in the various sets of processes.

#### **Personnel records:**

Of course, without perfect personnel records there is not possible to perform monitoring the values of all attributes that are required for a HRM system in a comprehensive way.

#### **Basic payroll processing:**

Standard solutions of these processes present a payroll processing of all types, including the calculation of taxes and social and health insurance payments. The calculation is performed either individually or in groups of employees who have the same parameters for calculating wages.

#### **Employee's absence:**

Registration and attendance processing includes complete oversight of employee attendance in the company and all the planning that is closely related with attendance. The attendance system is especially designed to fulfill the employer's duties under the Labor Code and Timesheets with regard to various time forms (overtime, night work, work readiness, holiday, business trip, etc.). Attendance Terminal, which provides the information, is directly linked to the human resources management system.

#### **Selection procedure:**

Indeed, organization of *Selected procedure* may be specific, but what is important are the necessary forms: Program selection procedure, Public access management, Application form and a CV (curriculum vitae). By these questionnaires evaluation, the enterprise can get all the information to make a selection. Therefore, the relevant processes are considered only as follows:

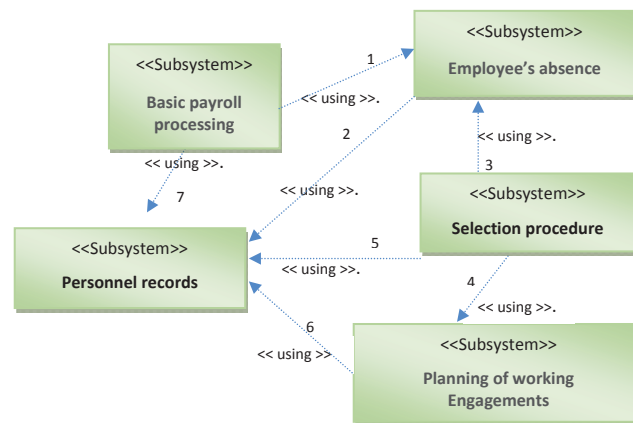
- Analysis of the usage status of a job vacancy
- Creation of a working job vacancy
- Creating documents for applicants
- Recruiting of applicants
- Selection of candidates
- Adoption of a new employee.

#### **Planning of working engagements:**

A thorough job description is a prerequisite for a later possible staff review by measuring their performance. Therefore the activity description, definition of requirements as for the employee's characteristics, experience and education of those who should hold the positions are concerned. Connecting to Personnel records is necessary. The description structure of an employee's position may enable automatic assessment of potential applicants.

Let us regard each set of processes such as a HR subsystem domain. We can assemble – from the above descriptions of the process set – the very first idea of process subsystems of any small business, further dependencies between them on the basis of relation <<using>> or another relation.

Since the concept of a relationship <<using >> is clear enough for its using in HR of a small business, we will not describe individual links between the



2: Subsystems of a problem domain and their illustrative relationships

subsystems. However, we can introduce another relation, but it is absolutely not necessary.

Figs. 3, 4 illustrate a basis of process diagrams for the first and second layers of a HR domain. Both charts will be used in the creation of content (internal behavior) for each subsystem of the domain HR. The creation of that content is specifically a task of an object methodology, the result being the subsystems content denoted as *Internal design behavior of subsystems*. The process view can be carried out using such methods as Eriksson–Penker business Extensions.

Using the method mentioned earlier, we firstly illustrate the process relationship of all five subsystems. Modeling gives results on the first layer of a double-layer diagram. Then we should continue by the construction of sub-process diagrams and get the second layer of the process diagram of the entire domain. In total we should get six sub-diagrams, one for the first layer and five for the second layer. The process of making the second layer will be reduced only to the subsystem *Selection process*, as an example model. For other subsystems, the making procedure is similar.

In the next text we gradually prepare the final *Component System for design architecture* by implementation of workflows meta-model “Applying of component system development in object methodology”. Of course, we have to start by Logic architecture modeling.

#### 4 Logic architecture

We start from the diagram *Subsystems of a problem domain*. We complete this diagram to specify the elements of the logical architecture (logically necessary components – management of summary reports, management of communication with clients, data base icon and icon for computer clients, control component, ...) which increase the transparency of the logical view. The resulting diagram *Logical architecture diagram* is already sufficiently transparent for the management of any small enterprise and software programmers of the software production company.

*Logical architecture diagram* is obtained by adjusting the diagram *Subsystems of a problem domain* by means of next steps:

1. We will take the basic links 1, 2, 3, 4, 5, 6 and 7 between the subsystems (relation <<using>>) in diagram *Subsystems of a problem domain*.
2. We add the elements of the logical architecture that will increase transparency of functionality logic and we determine the importance of other links 8, 9, 10 and 11.

The *Logical architecture diagram* introduces a certain data-driven philosophy, consisting in the following:

1. Only the subsystem *Personnel records* of employees can cooperate with a data base (DB). Therefore, other subsystems transmit data to this subsystem to write, read requests and take away the collected data supplied from DB in the agreed form.
2. Changing the type of DB (e.g. Oracle, Microsoft, ...) is carried out only in the subsystem *Personnel records* of employees. Formats of the distributed data between the subsystem *Personnel records* of employees and other subsystems are unchanged.

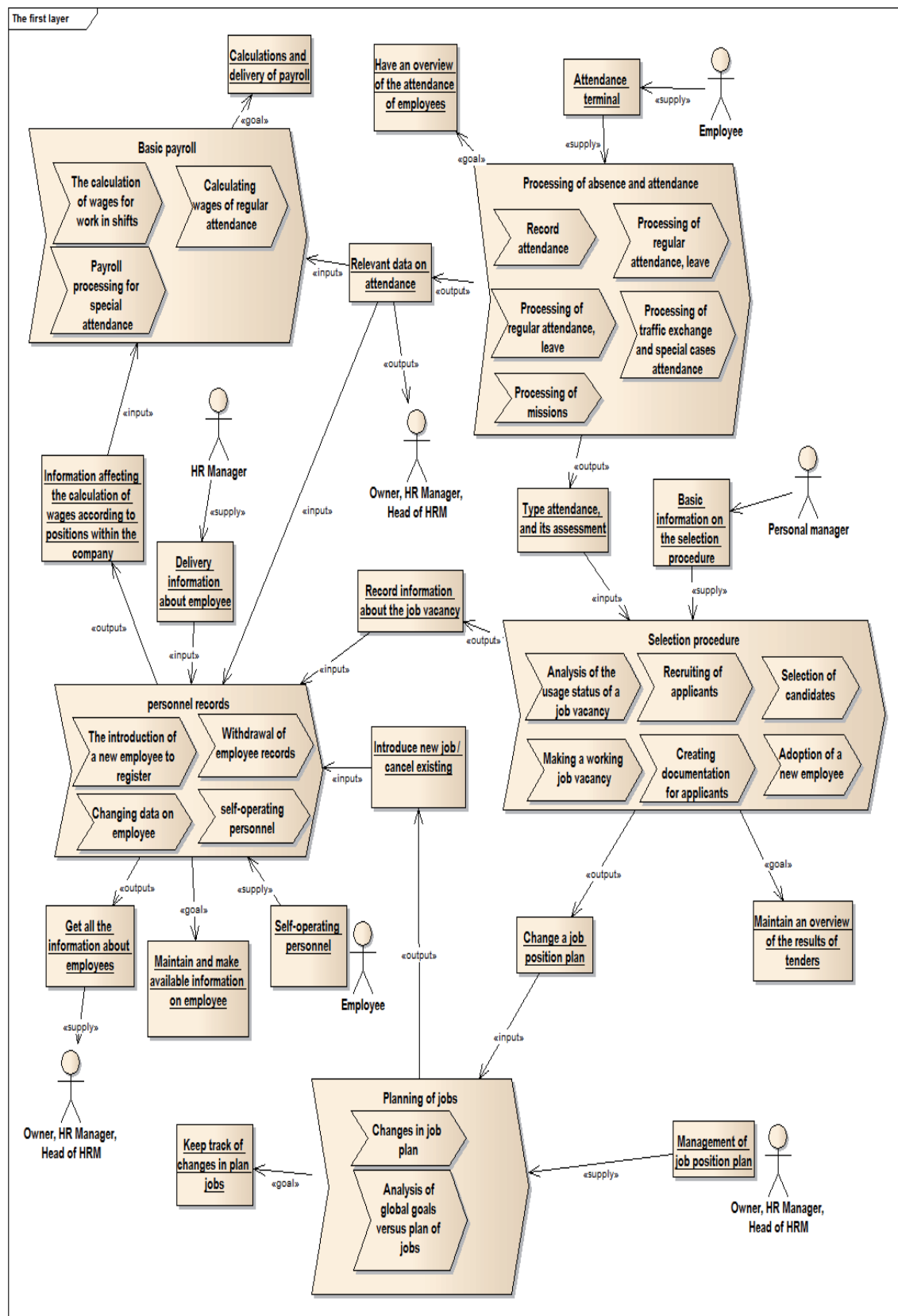
#### 5 Design architecture

The component system is made just for small business manufacturing firms, i.e. only for five subsystems (*Personnel records*, *Basic payroll processing*, *Employee's absence*, *Selection procedure*, *Planning of working engagements*) that are sufficient for the management of human resources.

Mapping of the subsystems to the components is 1:1 type. We determine a basic interconnection of components by means of interfaces according to following diagrams:

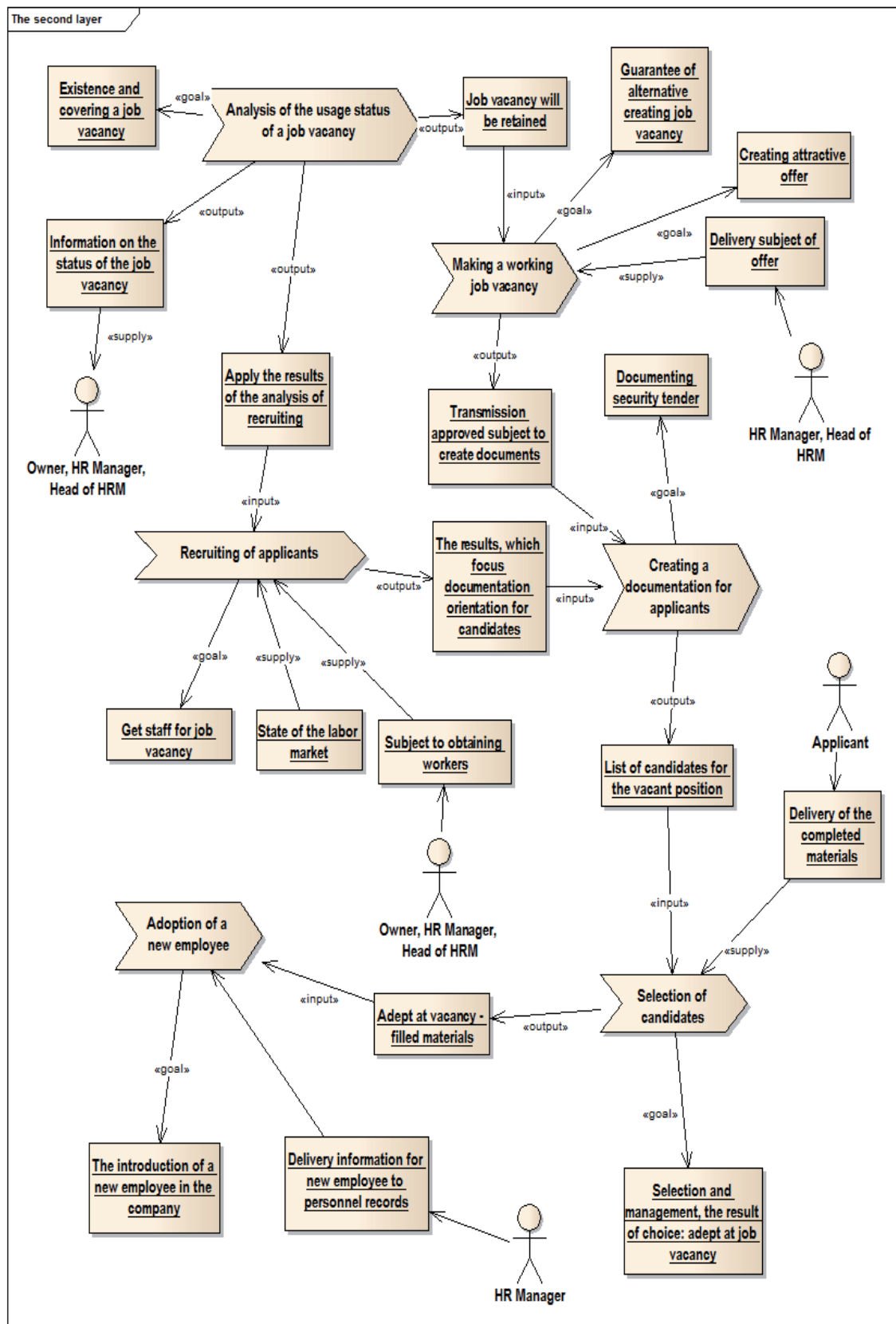
1. *Subsystems of a problem domain*
2. *Logical Architecture Diagram*
3. Process diagrams of subsystems (first layer).

Naturally, we need to establish for each component what it requires from the other components and what it provides to the other components as available for linking. Furthermore, let us remember that each component inherits the internal design of subsystem behavior, which was mapped to it.

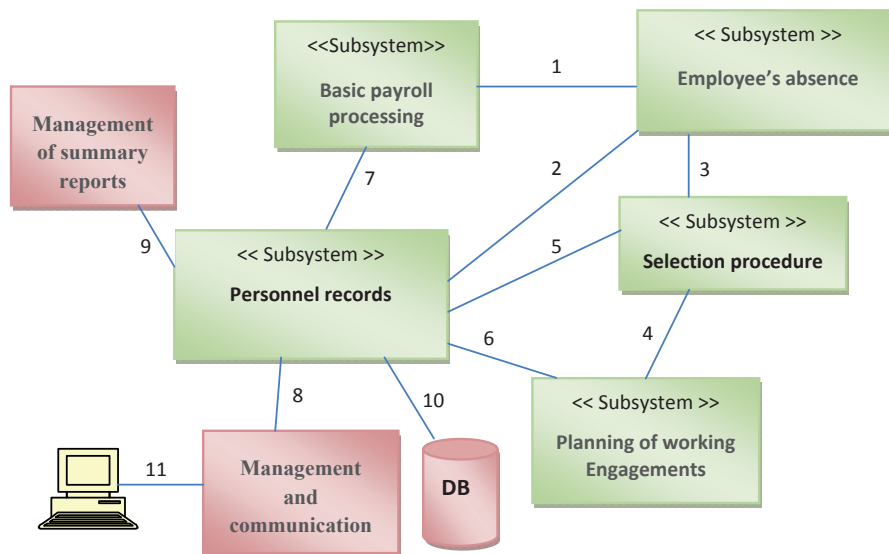


3: Domain HR – Process diagram of the first layer

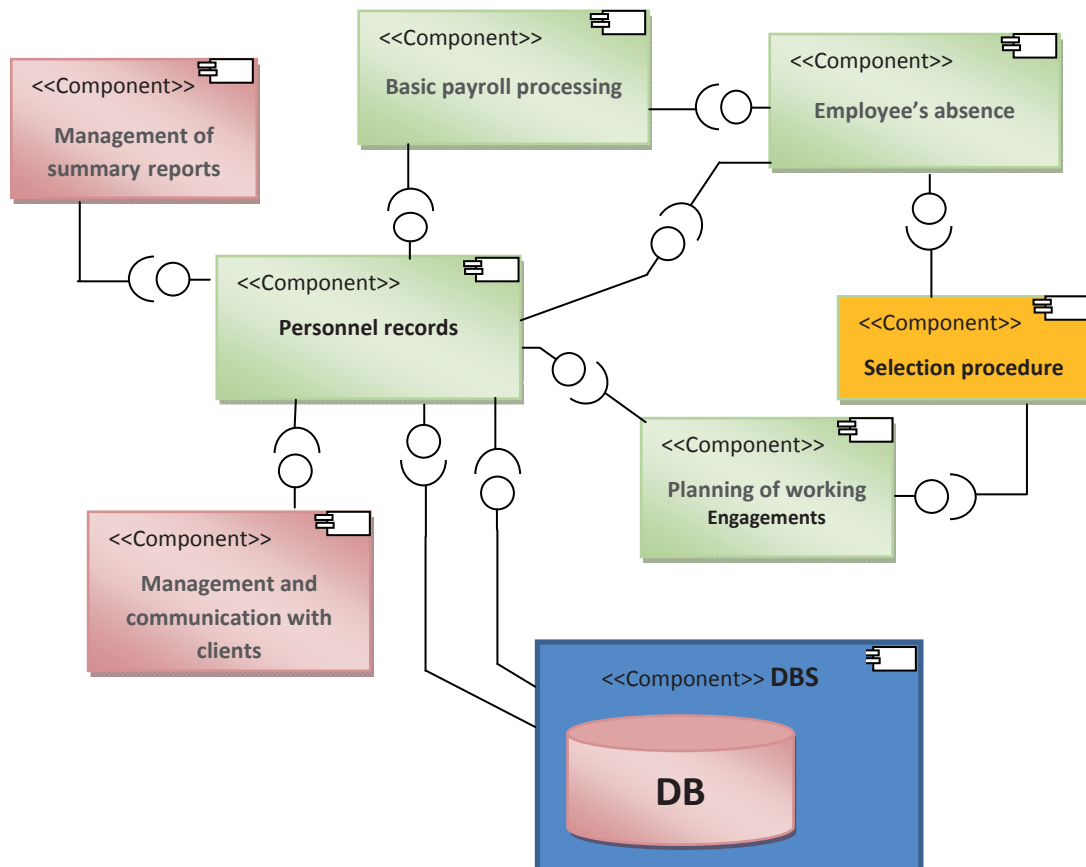




4: Domain HR (Selection procedure) – Process diagram of the second layer



5: Logical architecture diagram



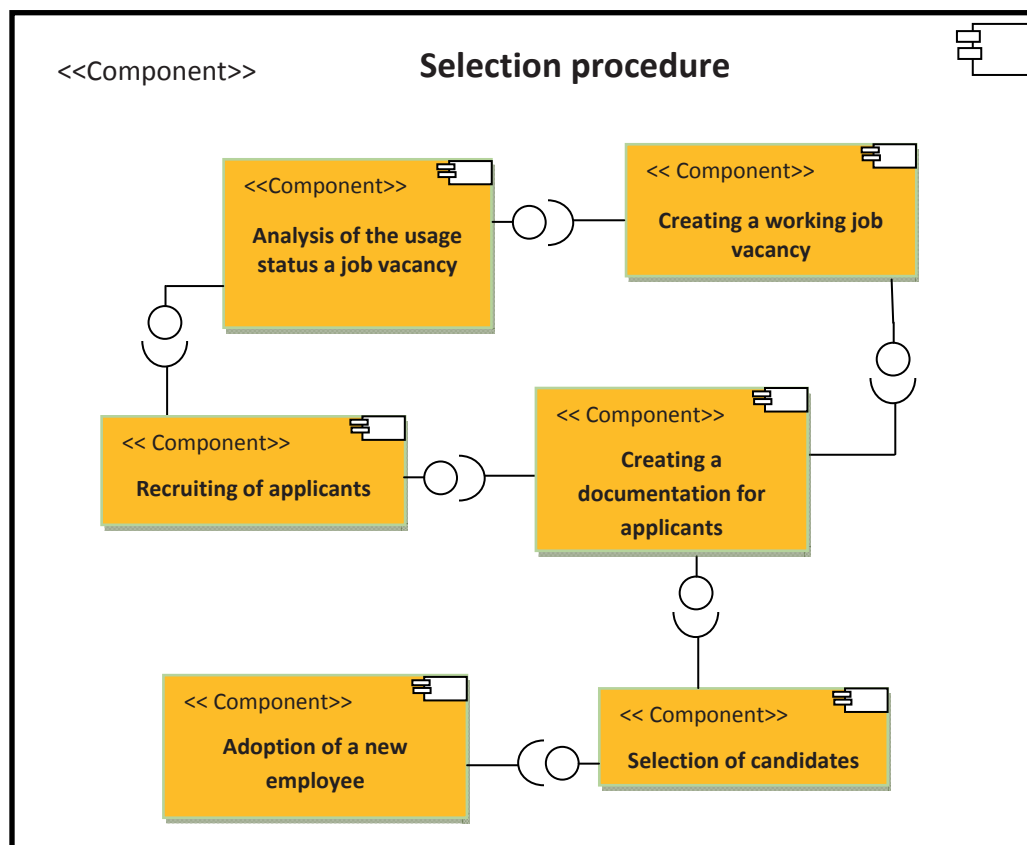
6: Component system for design architecture

We can obtain the diagram *Component System* of the second layer (only *Selection procedure*) primarily from the process diagram of this layer. Transformation is very simple and is based on the mapping of process diagram to components (e.g. mapping 1:1) and determines component interconnection. The

component *Selection procedure* is a composite one and mapping is a matter of the analytic point of view.

## RESULTS AND DISCUSSION

The result of this paper is the controlled implementation of the meta-model "Applying



7: Selection procedure and Component system for design architecture

of Component system Development in object Methodology” in the object domain HR for a small business. The sequence of work processes, together with a sequence of input and output artifacts of the meta-model and object methodology give a sufficiently transparent overview of the component system development process.

Whereas, the HRM concept of a small business is simple, we have not consistently described a construction way to the components interconnections, based on the interfaces. Our

idea being pointed only at the integration process of meta-model into the object methodology itself, we have not addressed many details (e.g. internal design of subsystems HRM behavior, interface specifications in the two developed component systems, a work scheme of component architecture).

We have also skipped the concept of Physical and Deployment architectures, see (Mišovič, Faldík, 2013), because that does not play such an important role as logical and design architectures.

## SUMMARY AND RECOMMENDATION

The paper uses the design meta-model “Applying of Component system Development in object Methodology”, which was created in (Mišovič, Faldík, 2013). This proposal is elaborated in mixed sequence workflows of a meta-model and object methodology. Two component systems and the interconnection of components in them have been constructed on the basis of the relevant workflows of the meta-model and object-oriented methodology. The first concerning the upper layer processes in the domain of HR, the other sub-layers. It is recommended that the reader should implement development schemes for the Physical architecture and Deployment architecture, although the meta-model in this development loses any influence.

## REFERENCES

KLČOVÁ, H., FERENČÍKOVÁ, D., 2010: *KS mzdy: specialista v řízení lidských zdrojů*. Prague: IT Systems

6/2011. Czech System Integration. [Online www. SystemOnline.cz].



- KOUBEK, J., 2003: *Personální práce v malých podnicích*. Praha: Grada Publishing a. s., 248 p. ISBN 80-247-0602-4. [cit. 11.04. 2013]
- MIŠOVIČ, M., FALDÍK, O., 2013: Applying of Component System Development in object Methodology. *Acta univ. agric. et silvic. Mendel. Brunen.*, 61, 7: 2515–2522. ISSN 1211–8516.
- ROSOL, I.: 2011: *Moderní systémy pro řízení lidských zdrojů*. Prague: IT Systems 6/2011. Czech System Integration. [Online [www.SystemOnline.cz](http://www.SystemOnline.cz)].
- SODOMKA, P., KLČOVÁ, H., 2010: *Personální IS budoucnosti*. Prague: IT Systems 12/2010. [Online [www.SystemOnline.cz](http://www.SystemOnline.cz)]. [cit. 10.04. 2013].
- TOLOCH, A., KUBÁŇ, M., 2010: *Moderní řízení lidských zdrojů*. Prague: IT Systems 12/2010. [Online [www.SystemOnline.cz](http://www.SystemOnline.cz)].

## Address

prof. RNDr. Milan Mišovič, CSc., Bc. Oldřich Faldík, Department of Informatics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech republic, e-mail: [misovic@mendelu.cz](mailto:misovic@mendelu.cz), [xfaldik@node.mendelu.cz](mailto:xfaldik@node.mendelu.cz)