CARL VON OSSIETZKY universität OLDENBURG
FAKULTÄT II
INFORMATIK, WIRTSCHAFTS- UND RECHTSWISSENSCHAFTEN

DEPARTMENT FÜR INFORMATIK

ABTEILUNG
WIRTSCHAFTSINFORMATIK I
VERY LARGE BUSINESS APPLICATIONS

PROF. DR. JORGE MARX GÓMEZ

TELEFONDURCHWAHL
(0441) 7 98 – 4470
Sekretariat – 4478

FAX
(0441) 7 98 – 4472

EMAIL
Jorge.Marx.Gomez@uni-oldenburg.de

GEBÄUDE A4
Uhlhornsweg 84 – Raum A4 3-318

OLDENBURG
12.05.2022

VERY LARGE BUSINESS APPLICATIONS
Carl von Ossietzky Universität Oldenburg

POSTANSCHRIFT
D-26111 Oldenburg

PAKETANSCHRIFT
Ammerländer Heerstraße 114 - 118
D-26129 Oldenburg

TELEFONZENTRALE
(0441) 7 98 – 0

BANKVERBINDUNG
Landessparkasse zu Oldenburg
Kto. Nr.: 1 988 112
BLZ: 280 501 00
BIC: BRLADE21LZO
IBAN: DE 4628 0501 0000 0198 8112

Bachelor's or Master's Thesis

A System Architecture for Managing Asset Master Data

(practical / application-focused)

Description

In the context of our MigHANA research cooperation (https://uol.de/en/vlba/projects/mighana) and together with our project partner OOWV (https://www.oowv.de/home/), we are developing innovative approaches for information management within a heterogeneous system landscape.

An important entity for a water utility are technical assets (wells, pumps, pipes, valves, the process stages of wastewater treatment plants, etc.). Assets - more specifically: attributes of these assets - play a role in many different business processes. They can be the subject of purchasing processes (procurement of a pump) or of major projects (example: construction of a wastewater treatment plant), they must be capitalized and depreciated in accounting (after acquisition or construction), they are monitored by control solutions and - if necessary - serviced in the course of maintenance processes.

Ergo, the "asset" entity is present in a variety of software solutions used by a water supplier. However, the minimum number of attributes that each of these solutions must know or the maximum scope it can handle are not disjoint, i.e., there are overlaps and redundancies in the databases. This raises questions regarding system and data architecture:

1. Which functions should be mapped in which software?
2. Which data should be stored in which software?
3. How does the data exchange between the software solutions work?

In other words: Theoretically, there is a myriad of variants for how m attributes of the "asset" entity can be distributed across n systems. In order to be able to make decisions in this regard, two preparatory steps are required:

1. There needs to be a catalog of decision criteria. What is the goal of the architecture? Should there be as few redundancies as possible? Should as few applications as possible be used? What should be minimized: the effort required for implementation, or the effort required for subsequent day-to-day operation of the system as a whole?

2. The data that must be held in individual applications is determined by two factors: on the one hand, there are certain minimum requirements defined by the application (for example, every solution involved will need the key of the "asset" entity); on the other hand, the scope of the information required in each application naturally depends on the function(s) that the respective application is supposed to perform.

Problem Statement

Depending on whether the task is designed as a bachelor's or master's thesis, different tasks arise. The work then has either a more practical or a more conceptual orientation.

When carried out as a bachelor's thesis, the focus is on developing a system and data architecture oriented alongside the "life cycle" of an asset (i.e., for example, from purchase to scrapping), considering 3-4 software solutions. The task is not to select a software but instead to decide, how functionalities and data are to be distributed over a limited and pre-defined software portfolio:

- Collect the functional and non-functional user requirements in the context of the life cycle of an asset (across departments)
- Collect the functional and non-functional technical requirements
- Define the solutions to be considered (3-4 applications)
- Familiarize yourself with the applications in question
- Select a modeling tool
- Definition system and data architecture (holistic view)
- Configure an example in the selected systems, possibly development of interfaces
- Workshop in which the result is evaluated

In the case of a master thesis, it is more a matter of developing a procedure for such a decision and answering the following questions:

- What are the relevant decision criteria for the project partner (independent of the specific entity and departments involved) and how do they relate to each other?
- How can the requirements recorded in different areas be abstracted across domains to such an extent that a uniform "requirements ontology" results?
- How are evaluations aggregated?
- What does the objective function look like?
- In addition to function-oriented considerations, business and strategic considerations must be taken into account; one keyword in this context is "TCO" (https://de.wikipedia.org/wiki/Total_Cost_of_Ownership). Low-cost solutions or solutions with low setup costs can subsequently prove to be very maintenance-intensive or entail extensive programming work to cover additional requirements.
- Can a procedure be developed that the cooperation partner can possibly use for other decisions between different solutions?

Requirement(s)

In the case of the bachelor's thesis, we are talking about a classic IT consulting task. A company uses a variety of software products for different reasons and - in view of overlaps in functions and data - must decide which parts/aspects of a business process should be covered by what solution. The focus here is not on technical but on business considerations. The student should therefore have at least a certain level of interest on such issues. It would also be helpful if the student has an interest in/previous knowledge of system and data architecture.

The master thesis is less about making one specific decision (the business scenario is meant to make sure that the work is of practical relevance) but rather about developing of a procedure model (if programming experience exists perhaps also a developing a tool) with which the project partner can make similar decisions in different areas. The prerequisite for this is a certain propensity/capacity for thinking in abstract terms.

Contact

Dipl. oec. Univ. Michael Mattern
michael.mattern@uol.de