

Flexible Multi-Dimensional Visualization of Process Enactment Data

Thomas Heer, Christoph Außem, and René Würzberger

RWTH Aachen University, 52074 Aachen, Germany,
{heer, aussem, woerzberger}@se.rwth-aachen.de,
WWW home page: <http://se.rwth-aachen.de>

Abstract. The management of development processes is a challenging task and needs adequate tool support. In the course of a development project, many different engineering and management processes are enacted and have to be controlled. The management data has an inherently multidimensional character. Most project and process management systems fail to present large multidimensional datasets in an adequate way. This paper describes a flexible approach, which leverages OLAP technology for the processing and visualization of multidimensional project management data in the plant engineering domain. The management data includes the execution traces and the progress measures of all workflows in an engineering project. The aggregation and visualization of this data facilitates the analysis of a huge number of process instances which is a prerequisite for process improvement.

Key words: Business process visualization, Data warehousing, Monitoring of business processes

1 Introduction

The four key responsibilities of *process management* are the definition, measurement, control and improvement of processes [1]. These activities form a *continual improvement cycle*, which applies in particular to the management of development processes. In a development project, many different, *flexible, collaborative processes* are executed and the project as a whole constitutes one instance of a development process. *Process management* is a means to improve the efficiency and performance of projects in an organization. The prerequisite for *process improvement* is the *measurement and analysis* of the enacted processes [1]. In large projects, software tool-support is needed for this task.

Plant engineering projects are a prominent example for large and complex development projects [2, 3]. They comprise a huge number of tasks and subprocesses, many involved process participants with different functional roles, and a highly complex product. The project management data is inherently *multi-dimensional*, where dimensions are e.g. the engineering phases, the functional roles and the different parts of the chemical plant.

For that reason, *software-tools* for the monitoring and analysis of development projects in plant engineering need to be able to handle and to *visualize*

multi-dimensional project data in an adequate way. Business Intelligence technologies like Online Analytical Processing (OLAP) [4] together with appropriate visualization techniques [5, 6] can be applied for this purpose.

In this paper we present a novel approach for *process status analysis* which is the foundation for process improvement [7]. This paper shows how the necessary enactment data of process instances can be collected, processed and analyzed. An overview over the approach is given in Fig. 1. *Measure values* are calculated for the process management data and they are exported to a *data warehouse*. Inside the data warehouse the measure values are arranged along several dimensions of a *hyper cube*. OLAP operations on the hyper cube lead to projections, which are visualized in a flexibly configurable view. The view for process status analysis is *coupled* with the different editing views for process management.

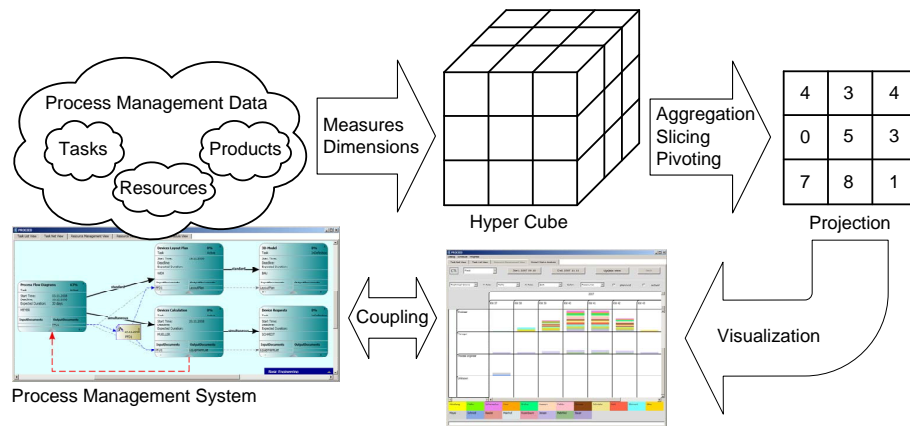


Fig. 1. Overview of the approach.

The described approach is implemented in the prototype PROCEED¹ which is developed at the Department for Computer Science 3 of RWTH Aachen University. PROCEED is an extension to the commercial computer aided engineering tool ComosTM, a product of Comos Industry Solutions GmbH [8]. The PROCEED prototype comprises project management functionality and a workflow engine for the management of recurring engineering tasks.

In Section 2 we describe the transition from the object-oriented data model for process management to the hyper cube of the data warehouse. Section 3 deals with the processing and the visualization of the data. Section 4 is concerned with the realization of the prototype. Related Work is discussed in Section 5, and Section 6 concludes the paper.

¹ Process Management Environment for Engineering Design Processes

2 Data Model

For PROCEED, an *object-oriented management data model* has been developed, which comprises entities and relations for process-oriented, data-driven project management in the plant engineering domain. Figure 2 shows a simplified cutout of the management data model.

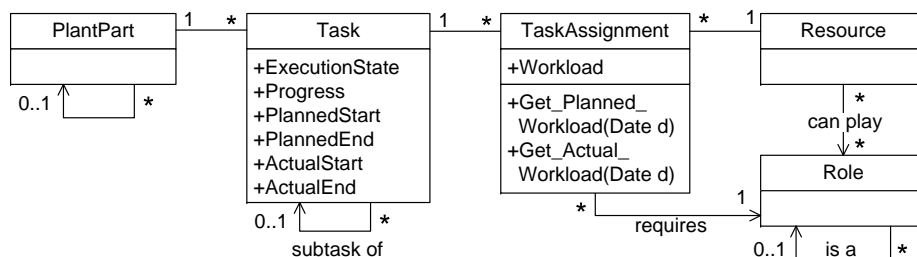


Fig. 2. Data model for project management.

The approach to *task management* implemented in PROCEED combines the main aspects of project and process management. All tasks have an *execution state* which can be modified by assigned users or the workflow engine. Tasks also comprise *scheduling data* like planned start and end dates, duration, etc. The system also keeps track of the *actual start and end dates*. Each task can be associated with a certain part of the chemical plant, which is the product of the design process.

For each task in the project, a workflow definition can be selected, so that the sub-process of the task is executed according to this definition. The task then constitutes an instance of the defined workflow type. In this way, each workflow instance is embedded in the overall dynamic task net, which defines the *context* for its execution. This connection between a project plan in the form of a dynamic task net and workflow definitions has been described in [9].

Users are assigned to tasks via explicit task assignments. A *task assignment* specifies a *required role*, which the user has to hold in order to be assigned to the task. Several users can be assigned to one task. For a task assignment, the *required workload* is defined and distributed over the duration of the task. By means of a time recording system, the *actual workload* of an assigned user is recorded for the task assignment. Roles, tasks, assignments and plant parts are defined for a specific plant engineering project.

The object-oriented data model is suitable for process management, but does not meet the requirements for project status analysis. To facilitate the latter, the management data is exported to a *data warehouse* (cf. Sec. 4) and OLAP technology is applied for data processing.

2.1 Process Measures

Process measures are used to analyze a running instance of a development process. The measure values are calculated by PROCEED and exported to the data warehouse. The defined measures have been identified together with the industry partner Comos Industry Solutions. The list of measures is extendable but would require a modification of the prototype. In PROCEED, the following measures are used:

- workload
- cost
- start date
- end date
- progress
- SPI (schedule performance index)
- CPI (cost performance index)

For a task assignment, there is the planned workload and the actual workload of the assigned resource. The unit of measurement for workload is person hours. For a task, the workload values of all task assignments and subtasks are aggregated. The cost of resources and tasks is derived from the workload and the cost rate of the respective resources. Start and end dates of tasks are planned, and the actual start and end dates are logged by PROCEED. The schedule performance index (SPI) and the cost performance index (CPI) indicate for a task, whether it is on schedule and within budget limits, respectively [10]. For each resource, the available workload per day is defined.

2.2 Dimensions and Hyper Cube

The different measures can be associated with tasks, users, roles, plant parts and dates. For this reason, the data in the data warehouse is structured along several *dimensions*. These dimensions span a *hyper cube* which stores all measure values in its cells, i.e. at its coordinates. The major dimensions of the hyper cube are:

- time
- tasks
- roles
- resources
- plant parts

The coordinates of each of these major dimensions are *structured hierarchically*, e.g. by the task-subtask relation and the specialization of roles. There may even be multiple hierarchies defined for one dimension, e.g. the tasks can also be grouped by the workflow type or the activity type of which they are an instance.

Fig. 3 shows on the left an example of a three-dimensional hyper cube which holds the actual workload in person hours. The depicted cube is a sub-cube of the complete hyper cube containing all dimensions and all measure values. It is the result of a so-called *slicing operation* [11], by which the coordinates on the

time dimension have been fixed to the timeframe 2009/09/08 to 2009/09/10. The values in the cube cells result from an *aggregation* over all coordinates of the remaining dimensions (e.g. plant parts) except for the time stamp dimension, where always the most recent value is used.

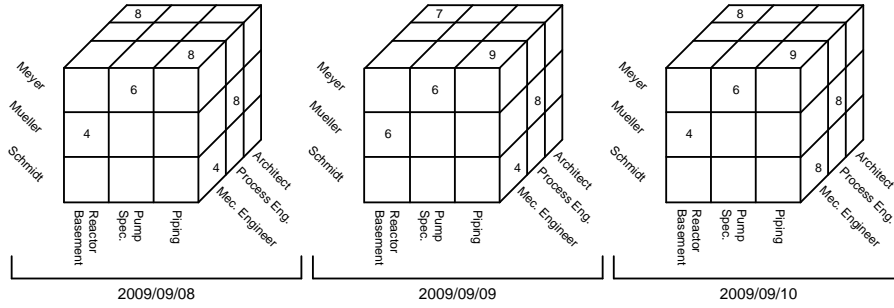


Fig. 3. Example hyper cube.

3 Configurable View for Process Status Analysis

PROCEED has been developed to support the management of dynamic development processes, where the process models may continually change during enactment. In PROCEED, tasks, resources, engineering data and their interrelations are modeled in one integrated management model

The PROCEED prototype provides several views for project planning and controlling. The *task net view* allows for the creation, execution and change of dynamic task nets [12]. A hierarchically structured *task list view* is used to give process participants an overview over their tasks. In the *resource management view*, roles can be defined for the project, the project team can be assembled and roles can be assigned to team members.

The *project status analysis view* of PROCEED has been developed for project monitoring and analysis. The development was initiated as a response to specific requirements of Comos Industry Solutions. There was a need for monitoring the huge amount of engineering and management processes in a plant engineering project. The project status analysis view comprises a *flexibly configurable pivot table* for the *multi-dimensional visualization* of different measure values. The coordinates of the pivot table show *stacked-bar charts*. The *configuration* of the pivot table is done by mapping the dimensions and measures of the data cube to the axes of the pivot table and the properties of the stacked bars. One dimension can be mapped to the color of the stack layers. The measure values are represented by the height of the stack layers.

Fig. 4 shows an example configuration of the status analysis view. The tasks dimension is mapped to the y-axis where the task-subtask relation defines the

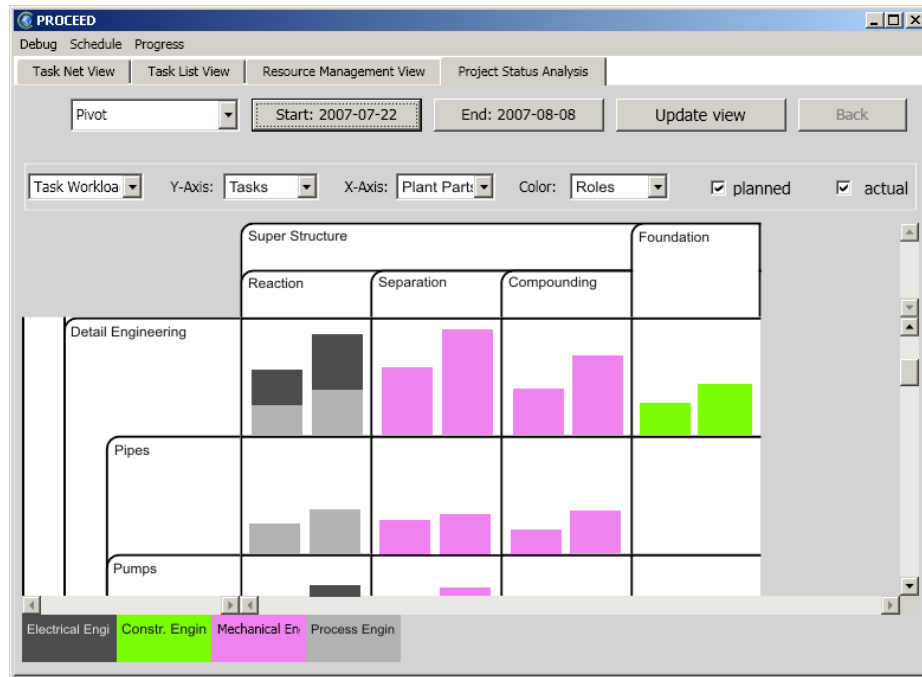


Fig. 4. Example configuration of the project status analysis view.

hierarchy. The plant parts dimension is mapped to the x-axis. Each cell of the pivot table holds one stacked bar for the planned and for the actual workload respectively. The roles are mapped to the colors of the stack layers. The timeframe is set from the beginning of the project to the current date. By means of this view configuration, the project responsible can inspect, how much workload was planned for the tasks in the project distributed over the different plant parts, and how much effort was actually spent on the respective tasks. When this view configuration is used after the end of the project or the end of a certain project phase, it can reveal that the planned effort defined in the process model was unrealistic, and that the *model has to be improved*.

The project status analysis view allows virtually any *combination* of dimensions and measures and thereby provides many *different viewpoints* on the project data. A view configuration can be *manually assembled* by selecting the dimensions and measures from drop down menus. However, some configurations are more useful than others for project status analysis. The most *common configurations* have been named and can be directly selected by the user. On the one hand, common views provided by conventional project management systems (PMS) like Microsoft Project can be configured: *Gantt Chart*, *Task Usage*, *Resource Graph*, and *Resource Usage*. On the other hand, there are several view configurations, which are not provided by conventional PMS. The configuration of Fig. 4 is called *Task Workload*. The view configuration *Technical Crews* fo-

cuses on the functional roles in the project and their planned workload in the upcoming weeks. The view configuration *Plants Progress* provides an overview over the progress made in the main engineering tasks with respect to the different plant parts.

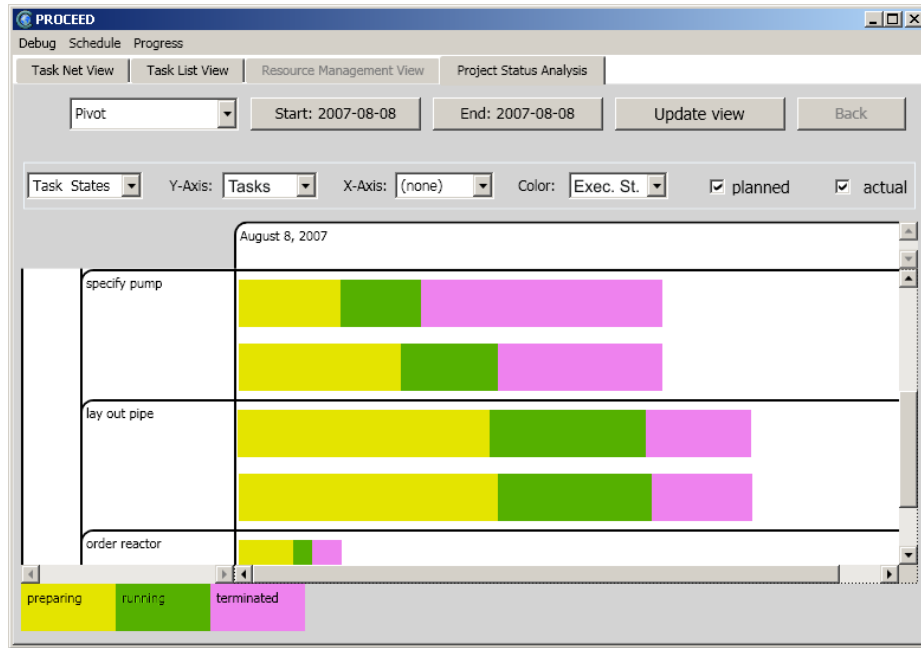


Fig. 5. Task States configuration of the project status analysis view.

A different viewpoint on the tasks in the project is provided by the view configuration *Task States*, which gives insight into the *current execution states* of the tasks in the project, which may differ from the planned states. This configuration is depicted in Fig. 5. The timeframe is set to the current date. Again, the tasks dimension is mapped onto the y-axis of the pivot table. However, this time the tasks are grouped by the workflow type of which they are an instance (cf. Sec. 2). The colors of the stacked bars indicate the execution state of the tasks, i.e. workflow instances. The measure is the planned overall workload of the tasks. By means of this view configuration, the project responsible can inspect, how much work is already successfully completed (terminated), how many tasks – as measured by workload – are currently processed (running), and how much effort still remains (preparing). These values are compared with the planned values which are derived from the planned start and end dates of the tasks. This view configuration is useful for *project controlling*, to analyze the overall performance of all workflow instances of a certain type during the course of the project.

In the configuration of Fig. 5 the numbers (bar stack widths) indicate that for the workflow type *specify pump* less instances have been completed than

planned, mainly because several workflows have not been started as planned. At the same time, the enactment of the workflow instances of type *lay out pipe* goes nearly as planned. The project responsible could gain even more insight into where the bottleneck is by mapping the resource dimension to the x-axis. This would reveal, if certain resources are responsible for the delay at the specification of pumps.

The information presented to the user in the *Task States* configuration is also displayed in the workflow management environment. A workflow definition is augmented by the information about all instances of this workflow type. Fig. 6 shows a cutout of the workflow definition *specify pump* as it is presented to the user. For the whole workflow type, the number of running (300), terminated (300) and created (700) workflow instances are given. For each activity, the percentage of workflow instances, which have completed the activity, is depicted. Furthermore, the average execution time of all completed activities is shown. The augmentation of workflow definitions is one aspect of the coupling between the process management system and the project status analysis view.

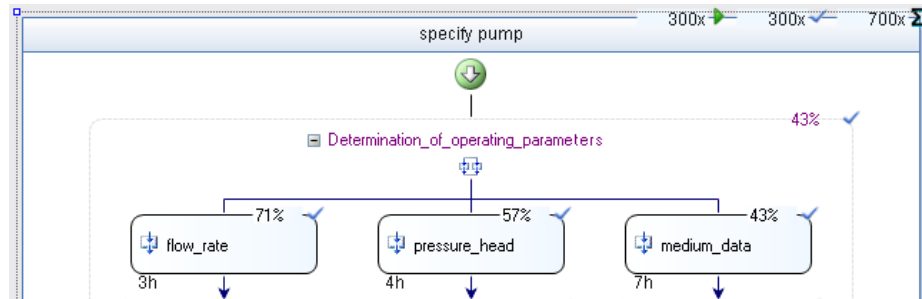


Fig. 6. Augmented workflow definition in workflow designer.

The project status analysis view supports all common operations on a multi-dimensional dataset [13]: Drill-down, roll-up, pivot, slicing and aggregation. With *drill-down* and *roll-up*, a dimension is added or removed from the visualization respectively, e.g. no dimension is mapped to the x-axis or the colors of the stacked bars. Drill-down and roll-up also refer to going downwards or upwards in the hierarchy of a dimension. The *pivot* operation changes the mapping of dimensions to axes while the number of displayed dimensions stays the same. A *slicing* operation fixes the coordinate on one or more dimensions, e.g. it sets a fixed date for which the management data should be displayed. The values in the hyper cube are *aggregated* along all dimensions which are not displayed, and which were not subject to a slicing operation.

The views for project and process management are *coupled* with the project status analysis view by navigation functionality. To take immediate action based on his analysis of the running process, the project responsible can navigate from the project status analysis view to the task net or task list view. Vice versa, the project manager often needs additional information when he is replanning the

project. For example, if he is looking for an additional resource for a task, he can *navigate* from a management view to the adequately configured analysis view, which shows him the utilization of the resources that can play the required role. General purpose multi-dimensional visualization tools fail to provide this tight coupling with project management views.

In general, the monitoring view can be used for two different purposes: for *process controlling* and for *process analysis* (cf. examples in Fig. 5 and Fig. 4 respectively). While the former may lead to *corrective actions* during the project, the latter can be used for *process improvement*. During the course of a development project, the different view configurations can reveal, if certain tasks exceed their time limits, or if the count of completed process instances of a certain type lies below the planned number. After the completion of an engineering project, the actual workload for activities and process instances can be compared with the required workload defined in the process definitions. The latter can be adapted if necessary.

Based on the time stamp that is assigned to each record in the data warehouse (cf. Sec. 4), it is furthermore possible to visualize how the planned values have changed over time. This allows for a very detailed analysis of the planning process itself as it is for example possible to see if the plan had to be adapted several times in the course of the project.

4 Realization of the Prototype

This section deals with the *realization* of the *project status analysis view*. PROCEED has been implemented as an extension to the computer aided engineering tool Comos. Comos is used in plant engineering projects to create and manage the engineering data, e.g. documents like technical drawings, device specifications and equipment lists.

Comos did not provide any explicit process management support until the beginning of the research project. Hence, in a first step, a *workflow management system* was implemented using the Windows Workflow Foundation [14]. The client applications of the WFMS allow for the definition and monitoring of workflows, which are enacted by the *workflow engine*. Workflow instances can be *dynamically changed at runtime*, whereby several correctness constraints apply. The developed workflow engine comprises *progress measurement* functionality. The progress of running workflow instances is automatically calculated by taking the control flow, the current execution state and experienced data for activity durations into account.

The core component of the PROCEED prototype is the *project management module*. The project management module and the WFMS are tightly integrated as described in [9]. A project plan is created in the form of a hierarchically structured *dynamic task net* [12]. The project management module checks the consistency constraints for dynamic task nets. PROCEED offers automatic *task scheduling* functionality. Furthermore, the *progress* of all tasks in the project is calculated based on different calculation methods like user estimates, document

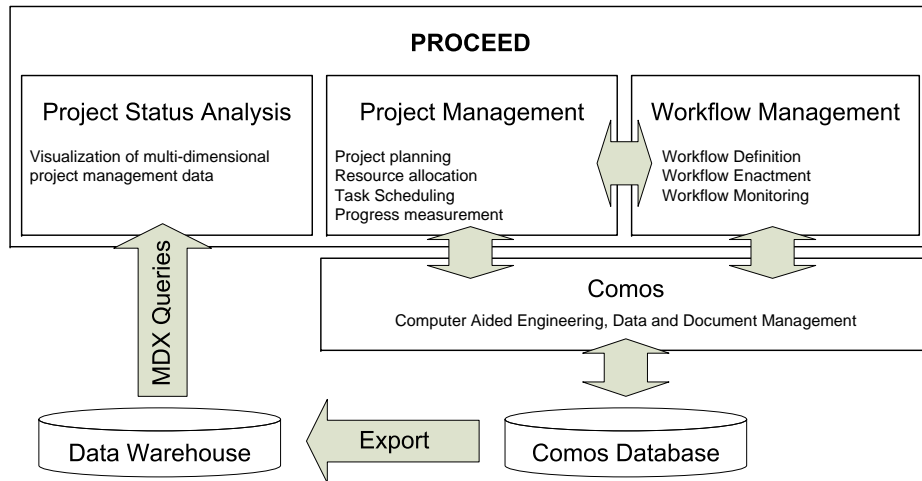


Fig. 7. General overview over the system architecture.

states or milestones. The measures used for project status analysis like start and end times, workload and progress are in large part calculated by PROCEED.

The management data of the project management component and the WFMS is stored in the Comos database, which is accessed via the Comos API. However, for the project status analysis view of PROCEED a separate *data warehouse* was required for technical reasons. In an ETL (Extract-Transform-Load) process, the management data is extracted from the Comos database and exported to the data warehouse. The export is *incremental*: Only changed measure values are exported.

The data warehouse is realized using the Microsoft SQL Server with Microsoft Analysis Services [15]. After the hyper cube has been generated, multi-dimensional data records can be retrieved from the data warehouse using the query language MDX.

For each export a new *time stamp* is created. All exported measure values are associated with this timestamp. This way, the history of plan changes is stored in the data warehouse, which can be used to assess the quality of process definitions.

Dynamic changes to the project management data are *immediately reflected* in the multi-dimensional analysis view. Whenever a change to a task, task assignment, resource or the like occurs, the changed values of the affected entities are immediately exported to the data warehouse and the hyper cube is updated. This dynamic update functionality constituted a technical challenge, since the export and processing of data is time consuming even for an incremental export, but the user should not be impeded in his work with the tool.

5 Related Work

The visualization of multi-dimensional data and especially the application of these techniques to project management have been tackled in several related research projects, some of which are described in this section. However, none of the related works covers the full cycle depicted in Fig. 1. Furthermore, domain specific measures and dimensions are neglected.

Polaris [16] has been a research project at Stanford University concerned with the visualization of multi-dimensional data from a data warehouse. The according prototype offers a configurable pivot table whose axes can be associated with the dimensions of a data cube. The entries of the pivot table can be numbers or even diagrams, and their color can also be associated with a dimension. The approach is closely related to ours. However, since Polaris is a general purpose visualization tool, it does not integrate with a process management system.

In [17], an application with different multi-dimensional views for project status control of construction projects is presented. The analysis is limited to budget data, and there is no pivot table among the visualizations. The focus lied rather on a comparative study about the utility of different diagrams than on the development of a UI concept for a project management tool.

Another approach which applies OLAP technology to project status analysis can be found in [18]. A multi-dimensional data model for a data warehouse is developed, which comprises five dimensions and the measures person hours, actual costs and planned costs. A pivot table is used to generate different views on the project data using MDX-queries. No information about plan changes in the monitored project is stored in the data warehouse. No graphical visualization techniques are applied to present the data in the pivot table. Only the standard functionality of the SQL-server is used. The focus of [18] lied on the evaluation of the OLAP technology for project management, but not on a suitable visualization of the data.

6 Conclusions

In this paper, a novel approach was presented for the analysis of process enactment data by means of OLAP technology and the application of visualization techniques for multi-dimensional data. The approach applies technologies from the area of business intelligence to process management. The PROCEED prototype has been developed as an extension to the CAE-tool Comos. The functionalities for project status analysis meet the requirements of the industry partner Comos Industry Solutions: There was a need for monitoring the huge amount of engineering and management processes in a plant engineering project. The prototype will be evaluated by customers of Comos Industry Solutions in the near future.

7 Acknowledgements

We thank Comos Industry Solutions for the valuable input and precise requirements, and the DFG for funding the project in the context of the Transfer Center 61 following the Collaborative Research Center 476 IMPROVE.

References

1. Florac, W.A., Carleton, A.D.: Measuring the Software Process: Statistical Process Control for Software Process Improvement. Addison-Wesley, Boston, USA (1999)
2. Helmus, F.P.: Process Plant Design - Project Management from Inquiry to Acceptance. Wiley-VCH, Weinheim (2008)
3. DIN German Institute for Standardization: PAS 1059 Processing plant design – Procedural model and terminology (2006)
4. E.F Codd, S.B. Codd, C.S.: Providing OLAP to User-Analysts: An IT Mandate. (1993)
5. Keim, D.A.: Information visualization and visual data mining. IEEE Transactions on Visualization and Computer Graphics **8**(1) (2002) 1–8
6. Tufte, E.R.: The visual display of quantitative information. Graphics Press, Cheshire, CT, USA (1986)
7. Schleicher, A.: Management of Development Processes - An Evolutionary Approach. PhD thesis, RWTH Aachen University, Aachen (2002)
8. Comos Industry Solutions: Website. <http://www.comos.com> (April 2009)
9. Heer, T., Briem, C., Wörzberger, R.: Workflows in Dynamic Development Processes. In: In Proceedings of the 1st International Workshop on Process Management for Highly Dynamic and Pervasive Scenarios (PM4HDPS 08), Springer (2009)
10. Patzak, G., Rattay, G.: Projekt Management. Linde (1998)
11. Lehner, W.: Datenbanktechnologie für Data-Warehouse-Systeme : Konzepte und Methoden, dpunkt-Verlag (2003)
12. Heller, M., Jäger, D., Krapp, C.A., Nagl, M., Schleicher, A., Westfechtel, B., Wörzberger, R.: An Adaptive and Reactive Management System for Project Coordination. Volume 4970 of LNCS., Berlin, Springer (2008) 307–373
13. Jarke, M., Lenzerini, M., Vassiliou, Y., Vassiliadis, P.: Fundamentals of Data Warehouses. Springer (2000)
14. Microsoft: Windows Workflow Foundation (2008) <http://msdn.microsoft.com/en-us/netframework/aa663328.aspx>.
15. Microsoft Corporation: SQL Server 2008 Analysis Services. <http://www.microsoft.com/Sqlserver/2008/en/us/analysis-services.aspx> (2009)
16. Stolte, C., Tang, D., Hanrahan, P.: Multiscale Visualization Using Data Cubes. IEEE Transactions on Visualization and Computer Graphics **9**(2) (2003) 176–187
17. Songer A D., Hays B., N.C.: Multidimensional Visualization of Project Control Data. Journal of Computing in Civil Engineering **4** (2004) 173 – 190
18. Hao Nie, Sheryl Staub-French, T.F.: OLAP-Integrated Project Cost Control and Manpower Analysis. Journal of Computing in Civil Engineering **21** (2007) 164–174