View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Digitale Bibliothek Thüringen

Progress Measurement in Planning Processes on the Base of Process Models

W. Huhnt, Technical University Berlin, Germany
(huhnt@tu-berlin.de)

Summary

This paper presents a specific modeling technique that is focused on preparing planning processes in civil engineering. Planning processes in civil engineering are characterized by some peculiarities so that the sequence of planning tasks needs to be determined for each planning project. Neither the use of optimized partial processes nor the use of lower detailed and optimized processes guarantee an optimal overall planning process. The modeling technique considers these peculiarities. In a first step, it is focused on the logic of the planning process. Algorithms based on the graph theory determine that logic. This approach ensures consistency and logical correctness of the description of a planning process at the early beginning in its preparation phase. Sets of data – the products of engineers like technical drawings, technical models, reports, or specifications – form the core of the presented modeling technique. The production of these sets of data requires time and money. This is expressed by a specific weighting of each set of data in the presented modeling technique. The introduction of these weights allows an efficient progress measurement and controlling of a planning project. For this purpose, a link between the modeling technique used in the preparation phase and the execution phase is necessary so that target and actual values are available for controlling purposes. The present paper covers the description of this link. An example is given to illustrate the use of the modeling technique for planning processes in civil engineering projects.

1 Introduction

The question how process models can be used in a beneficial way during the execution of projects in construction companies is still very much in dispute. A lot of investigations have been done in other fields of engineering tasks. For instance, process models are used in the software industries to monitor information systems or in the motor industries to analyze, optimize and simulate industrial processes. However, the existing experiences in the use of process models cannot be adapted directly to the construction industries.

Construction projects are characterized by some specific peculiarities. In general, a project is executed only once, mass production is not applicable. This is specifically true to planning processes where clients and project partners vary from project to project, contracts and responsibilities change, locations and buildings are different, task formulations and aims are specific, budgets and time frames are individual, etc. The complexity and the variation possibilities do not allow for specifying an overall process model to be adopted for each planning process in civil engineering. Also the use of predefined and optimized partial processes does not guarantee an overall optimized process.

Concatenating optimized partial processes do not guarantee an optimized overall process because of the interdependencies between tasks of different partial processes. These interdependencies need to be considered in the overall process, and these interdependencies result in an overall process where optimized partial processes are interrupted by tasks or sequences of tasks from other partial processes. This is illustrated in figure 1.

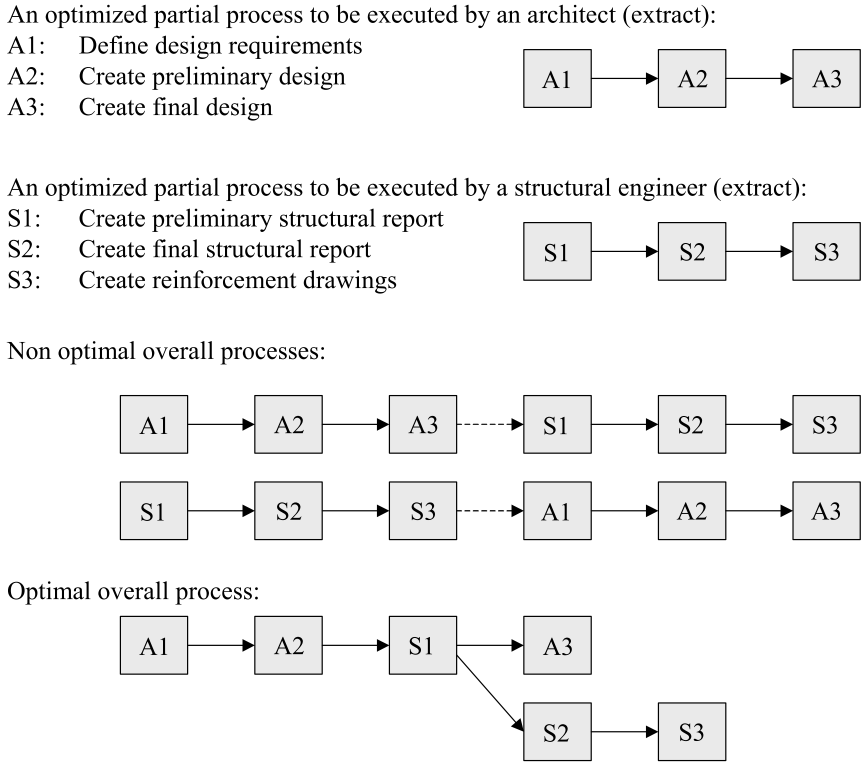


Figure 1: Optimized Partial Processes and an optimized Overall Process

Another problem results from interdependencies between different levels of details. It is not possible to derivate an optimized more detailed process from an optimized less detailed process. Figure 2 shows an example where an optimized process is modeled on a low detailed level. Each task of the process is subdivided into three tasks, e.g. as a consequence of the specific shape and geometry of a building. The original relations can be detailed as well. In addition, further relations between subdivided tasks can be necessary, e.g. based on capacity bottlenecks. The example in figure 2 shows that detailing results in an optimal process that is totally different from its origin. Detailed tasks of different original tasks need to be executed in parallel to achieve an overall optimum.

As a consequence, optimal sequences of tasks need to be developed for each planning project in civil engineering. This has to be done in a project preparation phase. However, requirements on project preparation phases consist of much more, e.g. scheduling, resource planning, and cost evaluations are mayor tasks in preparation phases.

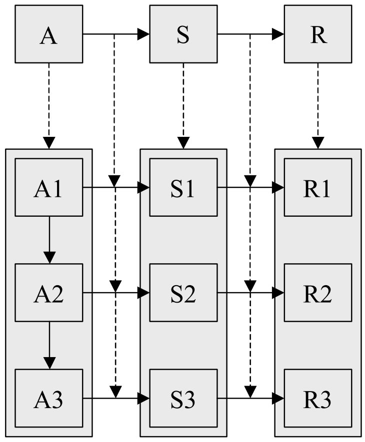
For different tasks in a preparation phase different tools are available. These tools need the sequence of the planning activities as an input. They address all necessary aspects for preparing a planning project. However, they do not support the determination of the sequence of the planning tasks in a generalized way, and they do not allow for consistency between the addressed aspects.

This paper describes a specific modeling technique for planning processes including the determination of consistent sequences of planning tasks, and it addresses the aspects of scheduling and cost evaluation for planning projects. As a result, target values are determined for a planning process. These target values are the base of an efficient controlling during the execution phase. The modeling technique guarantees consistency between all addressed aspects.

An optimized process (extract) :
A: create architectural design
S: create structural report
R: create reinforcement drawings

Detailing tasks and relations:
A: A1, A2, A3
S: S1, S2, S3
R: R1, R2, R3
A-S: A1-S1, A2-S2, A3-S3
S-R: S1-R1, S2-R2, S3-R3

Definition of further relations:
A: A1-A2, A2-A3



An optimal detailed process:

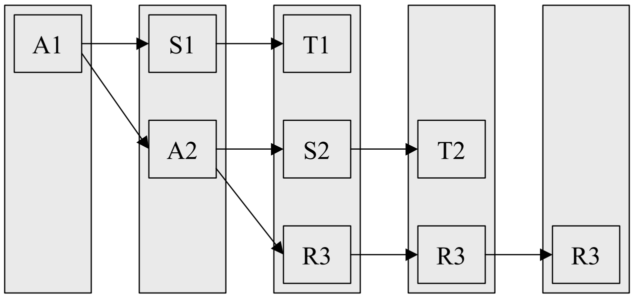


Figure 2: Processes Models with different Levels of Detail

2 Sets of Data

2.1 Significance of Sets of Data

Sets of data are the results of engineering tasks; sets of data are sold to clients. Planning processes have to be organized in such a way that the production of sets of data can take place by teams of specialists. Teams from different companies execute extensive planning processes. They produce sets of data in distributed environments.

The question of what can be seen as a set of data depends on the specific planning process and the level of detail that is necessary to prepare the planning process. Today, planning processes in civil engineering are either executed on the base of technical drawings or on the base of three-dimensional technical models. Beside technical drawings or technical models, reports, specifications etc. are worked out in planning projects, e.g. structural reports, electrical reports, or tender specifications. Independent of drawings or models, sets of data need to be named in the preparation phase. These names are used to denominate sets of data that have to be worked out during the planning process.

Up to four levels of details are sometimes used in planning project preparation phases. On the highest level of detail, each drawing, each report, each tender specification, etc. have to be named. Lower levels of details make use of summarized terms to denote sets of data. For instance, “reinforcement drawings for columns on axis A-A in the second floor of building xyz” is used as the name of a specific drawing on the highest level of detail whereas “reinforcement drawings for building xyz” might be used on a lower level of detail.

Naming of sets of data is essential in preparation phases. Extensive projects cover more than 10.000 technical drawings so that sets of data need to be structured, e.g. with respect to the disciplines by which they are produced. However, this paper is focused on the principle way of modeling planning processes to prepare and control a project. Aspects of structuring sets of data are not followed up.

2.2 Status of Sets of Data

Engineering work is characterized by three phases: distribution of work, parallel work on different aspects, and merging the results. Depending on the quality and completeness of the results, additional iteration cycles might be necessary. The phases of engineering work are shown in figure 3. In general, parallel work is executed independently. Sets of data are not necessarily consistent during this phase. Consistency is achieved by merging the results. In general, several specialists execute this task; interdependencies are discussed and an overall optimized technical solution is achieved.

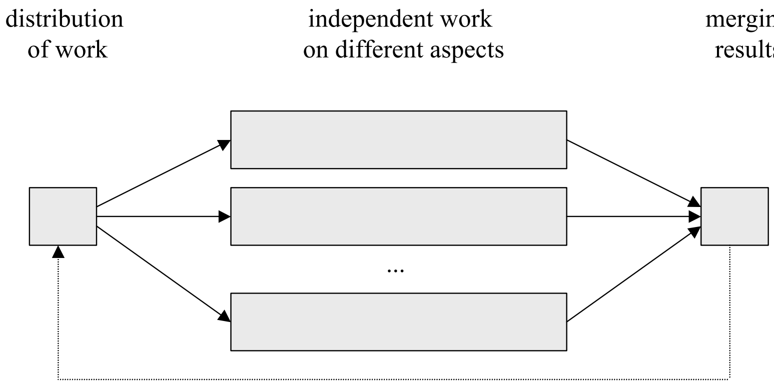


Figure 3: Phases of Engineering Work

During the planning process, the quality of sets of data and their completeness increases. This is expressed by a status attribute that is assigned to each set of data, e.g. preliminary, engineered, checked, final. In technical models, the status attribute is assigned to each component and a value has to be specified after engineering work has been executed on that set of data. This allows an automatic evaluation of a model so that an overview on the status can be generated at each time during the planning processes. Working on technical drawings requires a status attribute for each drawing. This status is assigned to each technical drawing after a specific task on that drawing has been executed.

The concept of status is not restricted to technical models and technical drawings. It is generalized applicable for each set of data that is produced or modified during a planning process, e.g. for technical reports, tender specifications or delivery notes.

2.3 Milestone Class

It is advisable to introduce milestone classes to a planning process. Sets of data that are produced or modified during a planning process are of a different type, e.g. technical drawings, technical reports, tender specification, or delivery notes. A milestone class covers the status attributes for all sets of data that are of a specific type.

Milestone classes are typically set up based on an overall list of status values. The sequence of the status values in the overall list is important because the quality and completeness of sets of data increase with respect to that sequence. Each milestone class is a subset of the overall list of status values.

It is possible to assign a rate to each status value of a milestone class. This rate describes the completeness of work on a set of data if it reaches that specific status value. Per definition, the sum of all rates in a milestone class is 1.

Figure 4 shows a simple example of a list of status values and milestone classes. The overall list describing status values consists of the four entries “assumed”, “preliminary”, “engineered”, and “checked”. A subset of these values is chosen for each milestone class, e.g. engineering drawings can be assumed, engineered, or checked. 20 % of the work on an engineering drawing is planned to be executed if it achieved the status “assumed”, 75 % if the status moves from “assumed” to “engineered”, and 5 % if the status moves from “engineered” to “checked”.

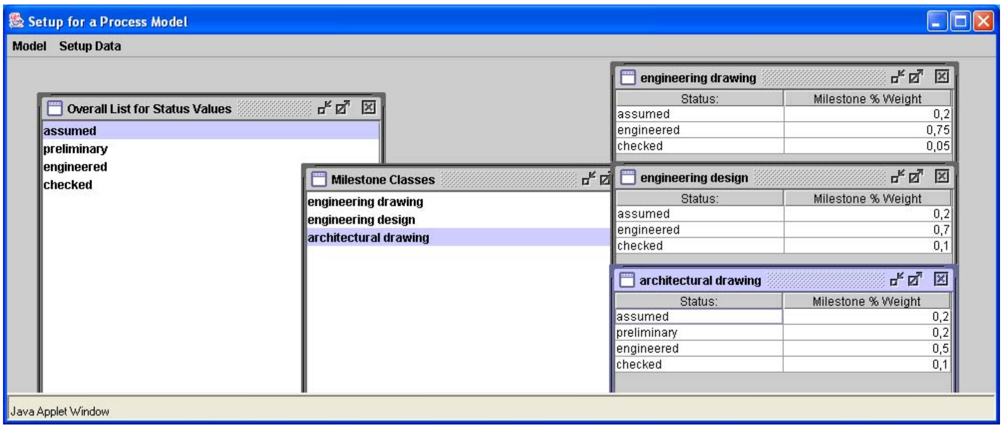


Figure 4: Status of Sets of Data and Milestone Classes

2.4 Weighting of Sets of Data

Each set of data can be weighted. This weight describes either the duration that is necessary to achieve the highest status for that set of data or the money that needs to be spent on that set of data until the highest status is achieved. It is a design decision whether the weights are specified in hours or in a currency. In general, both values can be converted by simply multiplying the duration by the money that needs to be spent for an hour of engineering work.

In the preparation phase, each set of data has to be weighted. Figure 5 shows sets of data of a simple planning process that covers some technical drawings and some reports. An intermediate level of detail is chosen so that for instance technical drawings are not named individually. They are named as architectural drawings, electrical drawings, foundation drawings, and reinforcement drawings. A milestone class and a weight are assigned to each set of data. Terms are used to structure the sets of data with respect to their data type. A work breakdown structure is not applied for that project.

3 Tasks and Sets of Data

3.1 Specification of Tasks

Tasks need to be specified for a planning process. Figure 6 shows tasks for a simple planning process. The project covers some tasks that need to be executed by an architect, some tasks require a structural engineer, some tasks require an electrical engineer, and some tasks need to be executed by a draftsman.

The tasks in figure 6 are ordered alphabetically. The process modeling technique presented in this paper does not require any interdependencies that have to be specified between tasks. Interdependencies between tasks are calculated on the base of relations between tasks and sets of data as described in section 4. However, relations between tasks and sets of data need to be specified. This is described in the next section.

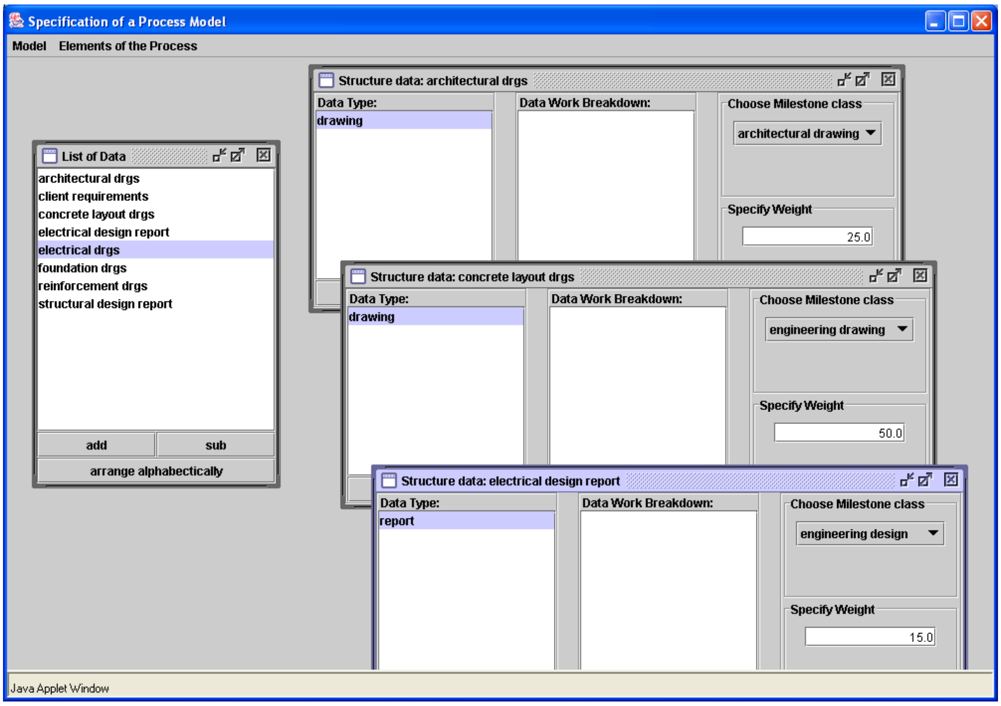


Figure 5: Weights of Sets of Data

check architectural drgs
check concrete layout drgs
check electrical design
check electrical drgs
check foundation drgs
check reinforcement drgs
check structural design
create architectural design
create concrete layout drgs
create electrical drgs
create foundation drgs
create reinforcement drgs
finalize architectural drgs
finalize concrete layout drgs
finalize electrical design
finalize electrical drgs
finalize foundation drgs
finalize reinforment drgs
finalize structural design
preliminary electrical design
preliminary structural design
review architectural design

Figure 6: Tasks of a Planning Project

3.2 Relations between Tasks and Sets of Data

Sets of data are produced during the execution of planning tasks. Three different types of access to sets of data are introduced: read, modify, and create. The deletion of sets of data is not considered in the present modeling technique. Sets of data are not deleted during planning processes. For documentation purpose, each set of data is stored independently of whether it is used or it is not used any more.

The three types of relations between tasks and sets of data can be grouped as shown in figure 7. Sets of data that are either read or modified during the execution of a specific task are input for that task; sets of data that are either modified or created during the execution of a specific task are output of that tasks. Input requires read access to the associated sets of data; output requires write access.

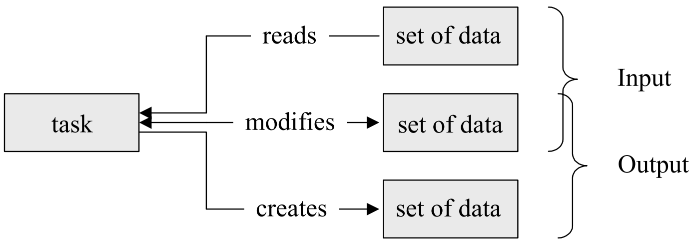


Figure 7: Types of Relations between Tasks and Sets of Data

The relations between tasks and sets of data need to be modeled. The value of the status has to be specified for the output of each task. This status is a target value that has to be achieved at the end of the execution of that task. Figure 8 shows three examples of tasks and relations to sets of data. The relations shown in figure 8 are based on the list of tasks as shown in figure 6 and the list of sets of data as shown in figure 5.

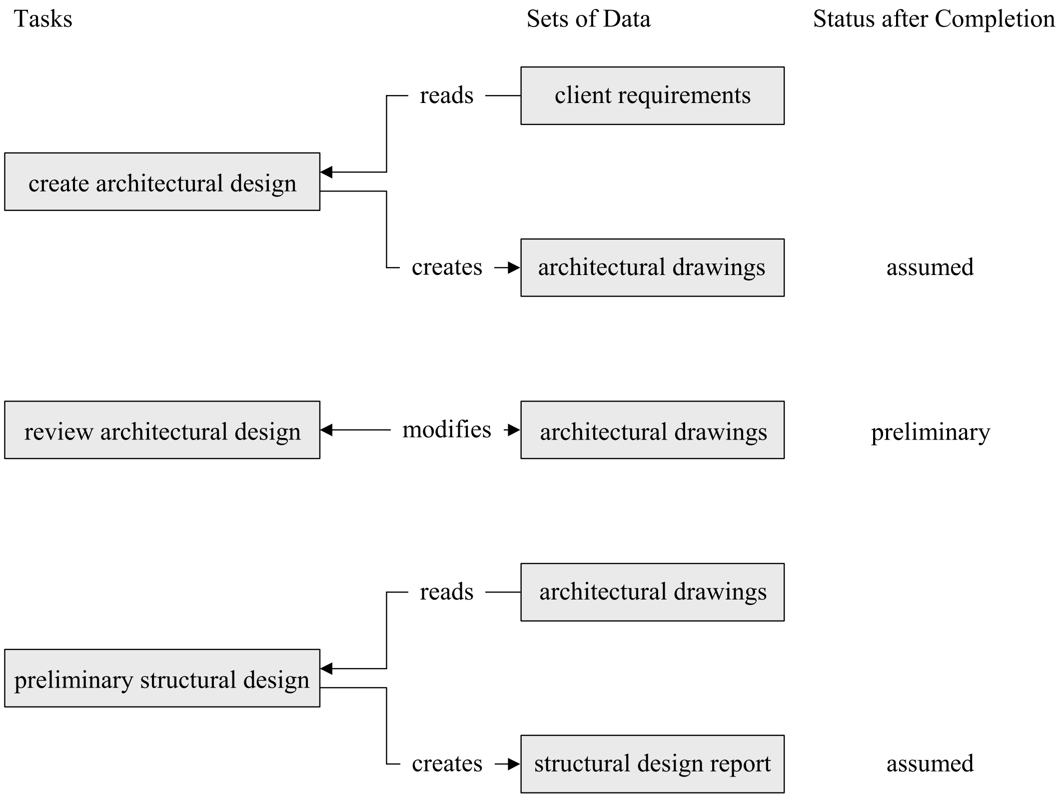


Figure 8: Relations between Tasks and Sets of Data

4 Sequence of Tasks

4.1 Determination of Interdependencies between Tasks

Interdependencies between tasks are determined on the base of relations between tasks and sets of data. Three rules are evaluated to determine these interdependencies.

First rule: “Sets of data can only be read and modified after they have been created”. This rule can be used to calculate relations between tasks. All tasks that read or modify a specific set of data have to be executed after the task that creates that set of data. Figure 9 shows the relations between tasks that are calculated on the base of this rule for the relations between tasks and sets of data as shown in figure 8.

Second rule: “The quality and completeness of sets of data have to increase if sets of data are modified”. This rule allows the calculation of relations between a task that modifies a specific set of data and all other tasks that have that set of data as an output: All that other tasks have to be executed before the actual task if they set the status of that specific set of data less the actual task. Figure 10 shows the result of the evaluation of this rule for the relations between tasks and sets of data as shown in figure 8.

Third rule: “The output of a specific task can not be better than its input”. Exceptions of this rule are all sets of data that are modified (rule 2). The highest status of the output of a specific task has to be determined. Than the input of that task is focused. All tasks that set the status of this input less or equal to the highest output status of the actual task have to be executed before the actual task. Figure 11 shows the result of the evaluation of this rule for relations between tasks and sets of data as shown in figure 8.

“Review architectural design” reads “architectural drawings” that are created by “create architectural design”.

“Preliminary structural design” reads “architectural drawings” that are created by “create architectural design”.

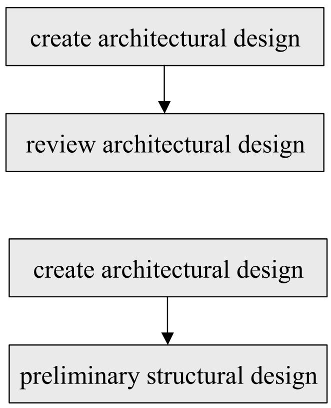


Figure 9: Relations between Tasks based on Rule 1

“Review architectural design” modifies “architectural drawings” to “preliminary” where “create architectural design” sets status of “architectural drawings” to “assumed”.

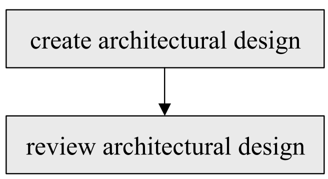


Figure 10: Relations between Tasks based on Rule 2

"Preliminary structural design" has the "structural design report" as an output and sets the status to "assumed" so that its input has to be at least "assumed" which is achieved by "create architectural design".

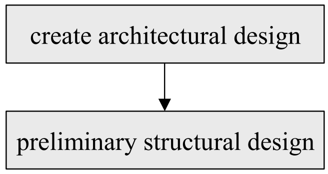


Figure 11: Relations between Tasks based on Rule 3

4.2 Topological Sort

The relations between tasks can be sorted if they do not cover a cycle. Cycles can occur if a task creates sets of data that are read by a second task that creates sets of data that are input for the first task. Such a situation is shown in figure 12. These conflicts can be solved if the affected tasks are executed in parallel. The affected tasks that have to be executed in parallel can be replaced by a major task so that cycles can be avoided.

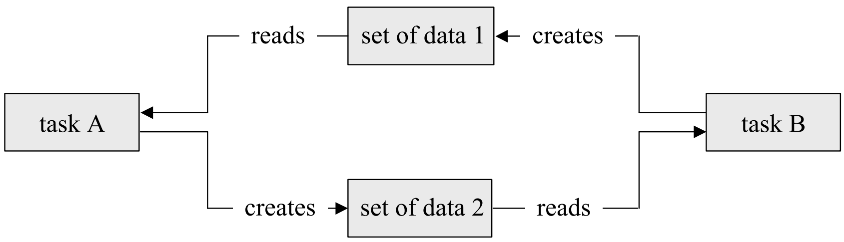
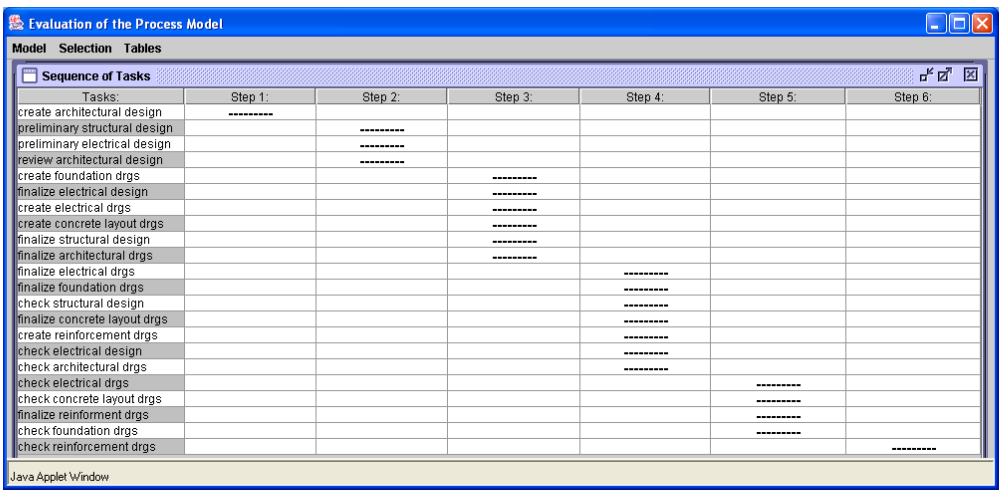


Figure 12: Cycle in a Process Model

A topological sort algorithm based of the breadth-first-search is chosen for the determination of the sequence of tasks. This algorithm guarantees the lowest number of logical steps, and each task is inserted at the first logical step when it can be executed (Pahl and Damrath 2001) (Turau 1996). Figure 13 shows the result of the topological sort algorithm for a simple planning process that covers the relations as shown in figure 8. As a result, task "create architectural design" has to be executed before task "preliminary structural design" and task "review architectural design".



Tasks:	Step 1:	Step 2:	Step 3:	Step 4:	Step 5:	Step 6:
create architectural design	---------					
preliminary structural design		---------				
preliminary electrical design		---------				
review architectural design		---------				
create foundation drgs			---------			
finalize electrical design			---------			
create electrical drgs			---------			
create concrete layout drgs			---------			
finalize structural design			---------			
finalize architectural drgs			---------			
finalize electrical drgs				---------		
finalize foundation drgs				---------		
check structural design				---------		
finalize concrete layout drgs				---------		
create reinforcement drgs				---------		
check electrical design				---------		
check architectural drgs				---------		
check electrical drgs					---------	
check concrete layout drgs					---------	
finalize reinforment drgs					---------	
check foundation drgs					---------	
check reinforcement drgs						---------

Figure 13: Sequence of Tasks

5 History of Development of Sets of Data

5.1 Quality and Completeness

Based on the sequence of tasks, the process model can be evaluated with respect to the planned development of the sets of data. This is expressed by the status that increases during the planning process. The input and output status can be visualized for each set of data in each logical project step. Figure 14 shows an example that is based on the sets of data as shown in figure 5 and the sequence of task as shown in figure 13.

Data:	1: in	1: out	2: in	2: out	3: in	3: out	4: in	4: out	5: in	5: out	6: in	6: out
architectural drgs		assumed	assumed	preliminary	preliminary	engineered	engineered	checked				
client requirements												
concrete layout drgs						assumed	assumed	engineered	engineered	checked		
electrical design report				assumed	assumed	engineered	engineered	checked				
electrical drgs						assumed	assumed	engineered	engineered	checked		
foundation drgs						assumed	assumed	engineered	engineered	checked		
reinforcement drgs								assumed	assumed	engineered	engineer...	checked
structural design report				assumed	assumed	engineered	engineered	checked				

Figure 14: Planned Quality and Completeness of Sets of Data

5.2 Time and Costs

The weights of the sets of data can be visualized in the same manner as the quality and completeness. Each set of data has got a weight that expresses either duration or money to be spent on that set of data. Figure 15 shows that values based on the weights as shown in figure 5. Absolute values are presented so that a progress curve can be drawn on the base of the summed up values. This progress curve with its typical S-form is shown in figure 16.

NAME OF DATA	Step 1:	Step 2:	Step 3:	Step 4:	Step 5:	Step 6:
architectural drgs	5,0	10,0	22,5	25,0		
client requirements						
concrete layout drgs			10,0	47,5	50,0	
electrical design report		3,0	13,5	15,0		
electrical drgs			1,4	6,6	7,0	
foundation drgs			2,0	9,5	10,0	
reinforcement drgs				2,4	11,4	12,0
structural design report		5,0	22,5	25,0		
TOTAL	5,0	18,0	71,9	131,0	143,4	144,0

Figure 15: Planned Evaluation of Weights of Sets of Data

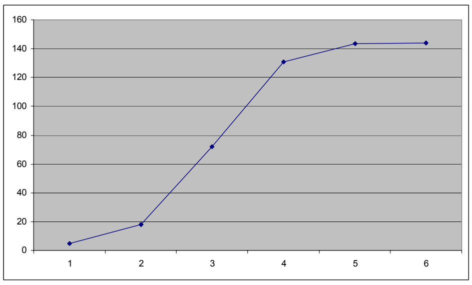
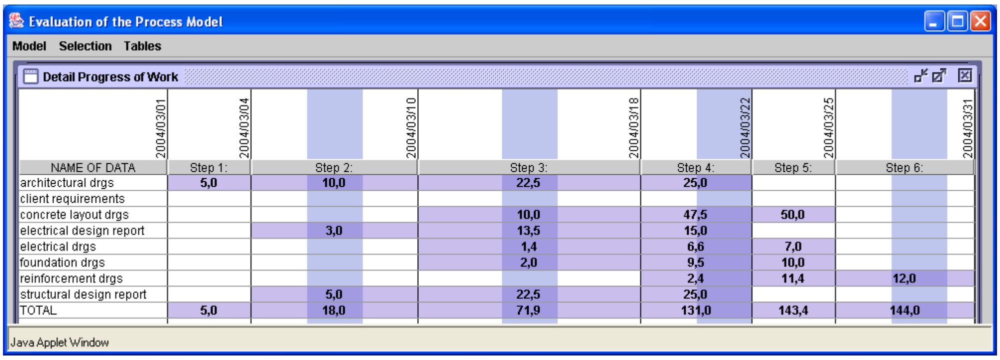


Figure 16: Planned Project Progress

5.3 Scheduling

Scheduling requires start and end dates for each logical project step. Figure 17 shows such a project schedule. The project covers a single month. Weekends are visualized as blue bars. The weights to be spent on each set of data are shown in that project schedule. These weights are target values for the execution phase.



NAME OF DATA	Step 1:	Step 2:	Step 3:	Step 4:	Step 5:	Step 6:
architectural drgs	5,0	10,0	22,5	25,0		
client requirements						
concrete layout drgs			10,0	47,5	50,0	
electrical design report		3,0	13,5	15,0		
electrical drgs			1,4	6,6	7,0	
foundation drgs			2,0	9,5	10,0	
reinforcement drgs				2,4	11,4	12,0
structural design report		5,0	22,5	25,0		
TOTAL	5,0	18,0	71,9	131,0	143,4	144,0

Figure 17: Planned Weights and Dates

6 Execution Phase

6.1 Operating Data Logging

Creating, modifying, and reading sets of data characterize the execution phase. Virtual project rooms in the Internet enable project participants to access to sets of data from all over the world. Upload and download functionalities are available to support planning processes in distributed environments (Haas, Ilieva and Kessoudis 2002).

Virtual project rooms allow storing and downloading of sets of data in files. Dates can be determined automatically if a person downloads or uploads a specific file. In addition, attributes can be used to store the status variable as part of a specific file.

The only information that has to be specified by project participants is the duration they work on a specific file between download and upload. This value has to be specified manually. As a design decision, “burned money” can be chosen instead of “burned time”. This depends of the general question whether time or money is chosen as the controlling variable. The difference between download and upload dates cannot be chosen instead of the duration of work; it is not guaranteed that each person works the complete duration between download and upload on that set of data.

The operating data logging consists consequently on three different types of information: upload date of a specific set of data, status of that set of data, and duration of work or money spent on that set of data. This information can be compared to the target process as described in the next section.

6.2 Comparison of Target and Actual

The comparison of upload dates and statuses of sets of data with the planned process gives a detailed view on the project progress. Tables can be drawn that combine the planned status as shown in figure 14 with the actual progress.

The comparison of upload dates, statuses of sets of data, and durations of work or money spent on sets of data with the planned process allows to control planning processes. The progress including target and actual weights are now available. These are exactly the information needed by project managers to monitor and control a planning process.

7 Outlook

A pilot-software has been implemented to set up target process models for planning processes as described above. The underlying model covers more than the described process modeling technique: data structures are available for the specification of persons and tools in order to assign persons to tasks (responsibilities) and tools to set of data (Huhnt and Lawrence 2004). Evaluation functionalities can be executed that calculate an overview on personal loadings or tool loadings. These functionalities enlarge the focus of the process modeling technique. All necessary aspects of the complete logic of planning processes can be modeled. However, functionalities for resource planning and optimization of resource loads are not addressed in the actual version. Further research has to be done to develop these functionalities into the presented modeling technique.

The presented modeling technique has been tested in real engineering planning projects. The technique does not replace existing tools for project management. It results in a coordinated and consistent base for project management and controlling. Export functionalities are implemented as part of the pilot-software so that information can be transferred to enterprise resource planning systems, scheduling systems, or engineering management systems. The advantage of using the presented technique is that all necessary information is consistent, and it can be specified quicker compared to existing tools where a lot of information has to be specified more than once and subsequent modifications need to be tracked by hand.

The link to an existing project room is under development. Some research work needs to be done so that the basic principles described in this paper can be proved in real planning processes. This is part of the next activities in enlarging the focus of the modeling technique described in this paper.

8 Acknowledgements

The considerations on detailing processes are based on personal discussions with Prof. Dr. Rudolf Damrath, Hannover, Germany, in September 2003. Aspects of dealing with statuses of sets of data and their weighting have been worked out in a common research project with Stellenbosch University, Stellenbosch, South Africa, Hatch, Johannesburg, South Africa, and Technical University Berlin, Berlin, Germany. The Hatch Group sponsors that project. The author thanks Jeff Lawrence and Kimon Cominos from Hatch as well as Bertie Olivier and Dr. Gert van Rooyen from Stellenbosch University for their contributions.

9 References

Pahl, P. J. and Damrath, R. (2001): Mathematical Foundation of Computational Engineering, Springer, Berlin, Germany

Turau, V. (1996): Algorithmische Graphentheorie, Addison-Wesley, Bonn, Germany

Hass, W., Ilieva, D. and Kessoudis, K. (2002): Projektmanagementsysteme sinnvoll einsetzen: Erfahrungsbericht aus dem Forschungsprojekt proCURE, AEC Report, Dressler Verlag, Jahrgang 2004, Heft 4, pp. 58-63, Heidelberg, Germany

Huhnt, W.; Lawrence, J. (2004): Methodik zur Modellierung konsistenter Soll-Vorgaben für komplexe Planungsprozesse, Bauingenieur, Springer-VDI, Jahrgang 79, pp. 50-56, Düsseldorf, Germany