Beyond Document Management -
Project Management Support System

Alan Hodgkinson
SoftXS GmbH
Alpensrasse 14
CH-6300 Zug
Switzerland

Joseph J. Kaelin
Pöyry Infra Ltd.
Hardturmstr. 161
CH-8037 Zurich
Switzerland

Summary

The creation of a *project management support system* leads to considerable project efficiencies and reduction of overall project risk.

The definition of a minimum set of project management structures and their dependencies enable the creation of the project management support system.

1. Overview

Effective project collaboration requires both a social element and a structured element. If teamwork (the sharing of knowledge, learning and building of consensus) is lacking within the project team then there is no collaboration. If there is good teamwork, but no structuring of goals, work scope and work assignments then delays and cost overruns will result. The role of project management is to provide the project team with the structures necessary to complete the project work and to further teamwork to do it competently and efficiently.

Implicit in collaborative work, but paradoxically often poorly practiced, is that all project participants must have easy access to project data (including documents). Engineering and construction are by their very nature data-driven activities. This leads us to the topic of this paper, which in its essence is *data-driven collaboration*.

Collaboration, with respect to project management, is the continuous sharing of project structures and data between the individuals leading the project and with the project team.

The authors have observed a gradual loss of engineering rigor in project management. Processes and structures, most of which have a long tradition in construction projects, are often neglected. When this happens, project work quality, schedules and budgets inevitably slip. Worse, because of the lack of proper project monitoring, the project directors may not even be aware of the poor performance.

Better incorporation of project structures into project management can help to regain engineering rigor, and improve project monitoring and control. Given today's pervasive use of computer software, the project structures and associated data must be discussed in a computer science context, providing a pathway for the creation of improved project management software tools.

2. Project Management Structures

The *project management structures* described in this paper are well-known, key components of the practice of construction project management. Project management structures refer to the ordering of project managment data, which comprise the inputs required for project management processes. *Project management processes* are the methods used to set up, control and monitor the progress of the project.

Project management structures include project breakdown structures, such as a work breakdown structure (WBS), organizational chart, cost account and contract breakdown. They also include document classification and coding schemes.

The core project management processes are:

1. Scope management
2. Organizational chart
3. Contract breakdown
4. Work plan
5. Cost control

Construction projects are phased and project structures are often required both as inputs and outputs (e.g. the scope of work for the engineering consultant includes work to produce the scope of work for the contractor).

In order to effectively carry out the project work and to achieve the project deliverables, the configuration of a number of project management support processes are also required. These typically include design management, document management, risk management and interface management.

The *project baseline* is an initial output of the project work and establishes the following:

1. Site conditions
2. Design (employer's requirements)
3. Project schedule
4. Cost estimate
5. Risk register

The project baseline is updated periodically as the project progresses.

Figure 1 provides an overview of the project management methodology, beginning with the set-up of the project management structures and processes, and then applying the project processes to develop the project baseline.

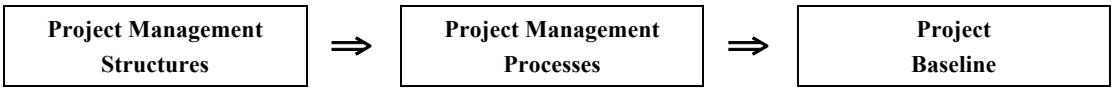

Figure 1 – Project Management Methodology

It is important to keep the project management structures and processes to the minimum, yet still provide the necessary project oversight and control. These are the methods which all project managers within a company or client organization should follow.

The basic structures required to set up project management processes, which will be referred to as the project configuration, are below shown in Table 1. These structures are the minimal required basic data that needs to be exchanged between project participants in order to run a typical project efficiently.

Structure	Description
Scope statement	summarizes all work to be done under the contract, including all work outlined in tender documents and all additional work identified during clarifications with the Client and included in the engineering services contract.
Work breakdown structure (WBS)	captures all work tasks to be completed during the project and decomposes the project into smaller and more manageable parts. The WBS is important for the work schedule, for document management, for the division of the work into contracts and for preparation of a Bill of Quantities. The WBS can in general be either functionally oriented or physically oriented.
Deliverables	identifies all required submissions as set out in the contract documents. In the design phase this includes the design reports, design drawings and tender documents. The contents and level of detail of the design reports and design drawings depends on the design phase and on the project delivery system (e.g. design-bid-build, EPC).
Organisational chart	determines how technical and project management decisions are made and also determines information flow within the project. The overall responsibilities of the project manager, lead engineer, discipline engineers and remaining project staff must be defined.

Responsibilities-authorities matrix	relates the organizational chart to the project WBS. Technical and managerial responsibilities and authorities ("competences") are assigned for all key positions, with each element of the WBS assigned to the responsible person with the appropriate authority (e.g. designer, checker, approver, submitter).
Design checking and approvals workflow	regulates how design packages and other project work are started, carried out, checked, approved and submitted. This includes internal workflows (design team) as well as external workflows (Client or Client's representative, vendors). The required time period for all reviews and approvals must be defined.
Project phases	are defined according to the Client's overall project planning and with consideration of the project delivery system.
Milestone dates	are defined according to the Client's overall project planning with consideration of expected construction progress.
Division into contracts	is a breakdown structure of all engineering and construction contracts envisioned by the Client to complete the project work. This breakdown can only be established after completion of feasibility and planning studies and is strongly influenced by the chosen project delivery system (e.g. EPC versus design-bid-build).
Cost account structure	organizes the costs of all work to be done for the project. There may be separate account structures for the project and for participating companies (who have their own accounting requirements) and the different account structures have to be keyed. The cost account structure serves as the basis for budgeting and monitoring project costs.
Document classification and coding scheme	defines all engineering and administrative classification attributes for documents (e.g. WBS, location, work type, project phase) and a coding scheme for documents (may have differing coding schemes for different document types).

Table 1 – Project Management Structures

Clear responsibilities for defining and updating the project data structures are essential. Table 2, in the next section, shows typical responsibilities.

The remainder of this paper examines the project management structures and processes, using computer science terminology, with the aim to implement improved project management software tools.

3. Project Data Structures

This section explains the data and data flows associated with creating and managing the *project data structures*.

The Project Data Structures represent all the individual data sets that are used as inputs and outputs for the monitoring and managing of a project's progress. The project data structures range from high-level statements of what the project intends to accomplish, known at the start of the project, down to low-level tactical data structures, representing the results of management and technical activities.

Some of the project data structures, called here *fundamental project data structures*, are given from the outside (either by the employer, the site, etc.) and are largely unchanging over the life of the project. Others, called here *derived project data structures*, are created during the project based on the fundamental structures and other derived structures. The derived project structures may be subject to change as the project progresses. Most of them are the outputs of specific project processes.

In this context, a *Project Management Support System* consists of the processes that read, create and update the project data structures; and project managers are the agents that take the project data structures as inputs and produce new project data structures, or modify existing ones, as outputs. The complete set of project data structures represent, to the degree that they accurately reflect the contractual and physical reality, a complete baseline of the project at a given point in time.

In order to actually plan and manage a project, each of the project data structures must be populated by the actual data associated with the particular project. A further project management challenge is to ensure that all the project data structures remain accurate and self-consistent in the face of the inevitable surprises and normal work that take place during the life of the project.

Successful project management therefore requires a systematic methodology for creating and maintaining a

complete and consistent set of project data structures. The first steps in developing this methodology are to define the basic requirements:

1. Define the complete list of project data structures to be managed
2. Define in detail the individual data items (record structures) in each project data structure
3. Define the inputs needed to populate each project data structure and maintain it over time
4. Define the person or team responsible for maintaining each project data structure

A comprehensive list of project data structures is show in table 1. The table shows the list of major project data structures, most of which should be familiar.

Req.	Project Data Structure	Data Type	Responsible
Project Definition			
√	A.1 - Scope of Work	list	Client Proposal Manager Project Manager (for change orders)
	A.2 - Terms of Reference / Contract	associated lists (table of contents, clauses)	Client
√	A.3 - Deliverables	list	Client Proposal Manager
	A.4 - Design packages	hierarchy	Chief Designer
	A.5 - Work Breakdown Structure (WBS)	hierarchy	Scheduler
Project Structure			
√	B.1 - Organization Breakdown Structure (org chart)	hierarchy	Project Directors Project Manager
√	B.2 - Project participants	list (name, company, position)	Project Directors Project Manager
Scheduling			
√	C.1 - Work Plan	associated lists (activities, schedule dates, assignments)	Scheduler Project Manager
√	C.2 - Activity Lists	list	Scheduler Chief Designer
√	C.3 - Activity Assignments	associated lists	Project Manager Chief Designer
√	C.4 - Scheduling Logic	data records	Scheduler
√	C.5 - Milestone Dates	associated lists	Client
	C.6 - Scheduling Dates	data records	Scheduler
Costing			
√	D.1 - Cost Budget	data records	Project Manager Cost engineer
√	D.2 - Resource Availability (man-hours available from participants)	data records	Department Heads
	D.3 - Man-hour Budget	data records	Project Manager Scheduler
Contractual			
	E.1 - Contract Breakdown Structure	hierarchy	Contracts Engineer
√	E.2 - Contracts	list	Client Contracts Engineer

Notes:
1. The **Req.** column marks items that are the minimum set of project management data required.

Table 2 – List of Project Data Structures

As stated previously, the project data structures belong to the project as a whole. Effective project management requires that the data associated with each project data structure be easily available to those who need it. This is particularly important for derived project data structures, as updates to one project data structure are likely to cascade to others.

On typical projects, the inability to easily find project structure data will lead to its duplication. If the existing data necessary to complete a task cannot be found, or if it is not known that the data exists, then the data will be re-collected or re-generated. If there are differences between the resulting data sets then it is likely that additional conflicts will occur, due to working from divergent baselines. This can affect the most basic project data structures, such as the scope of work, and there will be a serious risk of expensive mistakes.

An additional issue is that of defining baselines for each project data structure. The project data structures must be placed under formal control systems, so that the cascades of changes between project data structures can be managed. This gives users of project data structures confidence that they are working with correct and up-to-date information. The authors believe that the problem of efficiently handling cascading updates (including cross checking the results) of multiple project data structures is one of the serious bottlenecks in project management.

These issues raise a number of additional requirements for the project support system:

1. The existence and location of the primary data for each project data structure, and its owners, must be defined and broadcast to the entire project team
2. The dependencies between the different project data structures must be explicitly defined and rules for resolving differences between them must also be defined
3. Revision control workflows, including check, approval and distribution steps (based on the dependencies), must be defined for all project data structures
4. The format of the individual data items in the project data structures must be well-known and it must be possible to extract them for other uses

Unfortunately, it is not easy to meet all these requirements given the software tools available. The current state of the art is a divergent set of programs and systems, which frequently store their data in proprietary formats, where it is often difficult or impossible to export the underlying information. For example: work scheduling may be done using spreadsheets, MS-project, Primavera, or a any of a number of programs, all with incompatible data formats, and none of them offering any serious methodology for managing production, check, approval and distribution workflows.

In the ideal world, all tools would be able to export and import data in standardized formats, and it would be possible to automatically cross check data over multiple project data structures. Imagine being able to automatically generate a report that lists the differences between a deliverables list, a work breakdown structure and a work schedule, without having to deal with any data export and formatting issues.

Open data formats can provide a starting point for a better exchange of data between project participants, however project management still has the responsibility to ensure that the required basic data is available and accurate, by adhering to the requirements stated above.

4. Data Dependencies

The project management processes, as discussed above, are data-driven and depend on project data structures as inputs. The project data structures, in computer science terms, are simple. They consist of lists or hierarchies of data elements (e.g. records with fields and values). The individual data items have direct meaning to construction professionals, who also have an intuitive understanding their inter-relationships. This means that the project structures are actually easy to define and check.

A great deal of project management work consists of ‘running’ the project management processes, which take project data structures as inputs and produce new, or update existing, project structures and reports as outputs. The project management processes formally define the dependencies between the project data structures. A general list of dependencies is shown below in table 3.

As a trivial example: associating the project participants with the chosen form of project organization provides the organizational chart. As a more complicated example: associating the activity list with the scheduling logic, scheduling dates, activity assignments and resource availability provides the work plan.

At the start of the project, none of the project data structures exist. An important activity of project initiation is to produce the fundamental project data structures, such as the project scope, milestone dates and work breakdown structures. The project data structure production proceeds from the general to the specific, starting with the

project scope and work breakdown structures. The fundamental data structures are then used as inputs to create the, more detailed, derived project data structures, such as schedules and activity lists.

The data dependencies, e.g. the inputs, for the derived project data structures fall into two classes:

1. *Fundamental data inputs* that determine the project data structure's main content
2. *Check data inputs* that are used to check the project data structure for completeness and correctness

The process is shown below in figure 2.

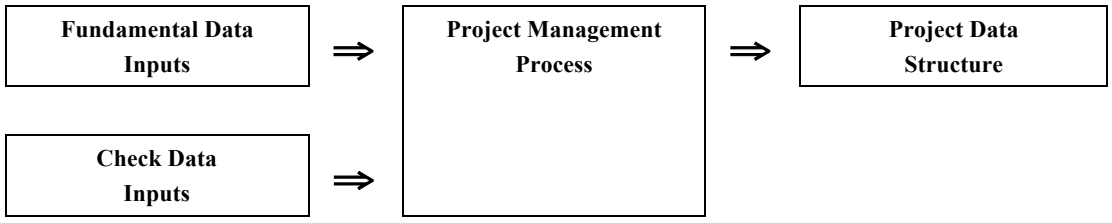

Figure 2 – Project Management Process Inputs and Outputs

The basic process for creating (or updating) a project data structure is to use the fundamental data inputs as the main guide for creating and updating records, and then using the check data inputs to verify the result. For example:

1. The organizational chart requires the list of participating companies and individuals together with the chosen form of project organization as fundamental inputs, and the resulting organizational chart as the means of checking for correctness.
2. The work plan requires the activity list and milestone dates as the fundamental input, and the scheduling dates, activity assignments, scheduling logic and resource availability as the constraints for checking for completeness and correctness.

Project Data Structure	**Type**	**Dependencies**
A.5 - Work Breakdown Structure (WBS)	F C	A.1 - Scope of Work Project schematic layout
B.1 - Organization Breakdown Structure (org chart)	F C	B.2 - Project Participants Agreed organizational form
C.1 - Work Plan	F F C C C C C	A.3 - Deliverables C.2 - Activity Lists C.3 - Activity Assignments C.4 - Scheduling Logic C.6 - Scheduling Dates C.5 - Milestone Dates D.2 - Resource Availability
C.2 - Activity Lists	F C	A.1 Scope of Work Development of detailed tasks
D.1 - Cost Budget	F F C	C.2 - Activity Lists C.3 - Activity Assignments D.3 - Man-hour Budget
E.1 - Contract Breakdown Structure	C F	A.5 - Work Breakdown Structure E.2 - Contracts
Notes: 1. The **Type** column marks the dependency type: **F** is *fundamental* and **C** is *checking*.		

Table 3 – Project Data Structure Dependencies

Over the life of the project, the project data structures are refined, as additional detail is added to them. This occurs in a top-down manner. In the early project phases only the top-level of data hierarchies are known. As work proceeds, project activities will produce additional information, which is used to fill in the lower levels of the project data structures.

Updates to a project data structures leads to the cascading updates, described above. When a project data structure is updated, the chain of dependencies must be followed and the dependent project data structures must also be updated and checked for correctness. A key part of this process is that there is only one definitive instance of each project data structure (as defined in the requirements in the previous section).

5. Computer Implementation Issues

Computer implementations of project management processes are intended to automate the management of project data structures, the checking of the dependencies and the viewing of the resulting data. Currently we know of no software systems that implement the authors' vision. This section is intended to provide guidelines for software implementers.

Large, complex project management software systems, typically requiring extensive and specialized training are not very suitable for collaboration in project management processes. They typically only implement one part of a single management process (such as scheduling) and it is difficult to get them to receive data from, and share data with, other management processes.

The best solution to this impasse is to define *open data formats* for exchanging data between software systems, and then modify or create software systems that use them. The open data formats should defined be based on the project data structures described previously. Ideally this should be done on an industry-wide basis. Using open data formats to exchange data between project participants enables collaboration when performing project management processes.

Implement modular software tools that *'do one thing well'*, using open data formats. These tools should read their inputs and write their outputs directly from and to the project data structures. Advantages:

1. Significantly reduces the complexity and consequently the effort required to implement, learn and operate the software tools
2. Easier to replace individual software tools with improved versions without disrupting the entire software '*ecosystem'*

The data input, validation and processing should be kept separate from reporting. Reporting tools should self-standing, reading their inputs directly from the project data structures and presenting their output in the format most appropriate for the intended use. Advantages:

1. It is easy to implement new reports as new uses for data are discovered. This can often be done by the data users themselves without involving the data producers
2. The data collection, input and validation processes can be optimized without regard to reporting. The easiest and most accurate way to collect and enter data is usually unrelated to the downstream reports

Needless to say, all software systems must be web-enabled to facilitate collaboration, regardless of the geographical location of the participants. Web-based '*Cloud'* software has matured and can be deployed reliably and quickly. Every computer already has a web browser, which allows work to start immediately. Dedicated applications are discouraged because company software policies tend to prohibit or hinder their installation on team member's computers.

Acceptance by users is a key factor in the adoption of new software tools. The most important acceptance criterion is that the tool be appropriate for its intended use. Those tools that make use of familiar procedures and terminology (provided they actually do the job) are most likely to be embraced by construction professionals. The second criterion is that the software can be rapidly learned and used to perform useful work.

The core project management processes outlined in section 1 are the core of the project management support system. They provide the most efficient means of accessing the project progress and determining corrective action. A software implementation will help project managers complete projects on time and within budget.

6. Conclusion

1. The role of project management is to provide the project team with the structures necessary to complete the project work and to further teamwork to do it competently and efficiently.
2. Successful project management requires a systematic methodology for creating and maintaining a complete and consistent set of project data structures.
3. Computer implementations of project management processes are intended to automate the management of project data structures, the checking of the dependencies and the viewing of the resulting data.
4. Implement modular software tools that *'do one thing well'*, using open data formats. All software systems must be web-enabled to facilitate collaboration, regardless of the geographical location of the participants.
5. The proposed project management support system is easy to understand and work with since it reinforces accepted best practices.
6. The authors believe that the formalization of this process will decrease project delays and better control project costs.

Bibliography

Hodgkinson, A. & Kaelin, J., "*Regaining Control: Finding the Information Needed for Effective Decision Making*", World Tunnel Conference 2008 – Underground Facilities for Better Environment and Safety, Vol. 3, pp. 1715-1725, Agra, India, 2008.

Hodgkinson, A., J., Gisiger, Kaelin, J-P., Schmuck, C., "*A Conceptual Methodology and Practical Guidelines for Managing Data and Documents on Tunneling Projects*", Hydro 2010 Conference Proceedings – Meeting Demands for a Changing World , §16.02, Lisbon, Portugal, 2010.

Hodgkinson, A., Kaelin, J. & Pike, M., "*Standing on the Shoulders of Giants: A Vision for Data Standards for Hydroelectric Project Management*", Hydro 2011 Conference Proceedings – Practical Solutions for a Sustainable Future, §28.03, Prague, Czech Republic, 2011.

Hodgkinson, A. & Kaelin, J., "*Project Configuration: Making Project Management Software More Useful*", World Tunnel Congress 2012 – Tunnelling and Underground Space for Global Society, pp. 867-868, Bangkok, Thailand, 2012.

Authors

Alan Hodgkinson graduated with ScB degrees in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology. He is the general manager of SoftXS GmbH, a company specializing in document and data management for large construction projects. Email: alan@softxs.ch. Tel: +41-41-511-9776.

Joseph J Kaelin graduated with an ScM degree in Civil Engineering from the Massachusetts Institute of Technology. He is a senior tunnel engineer and senior project manager at Pöyry Infra Ltd, and has worked on hydroelectric projects for over three decades. Email: joseph.kaelin@poyry.com. Tel: +41-76-356-2050.