

Coordinating dependencies in global system development projects – The use of dependency diagrams

Lars Taxén and Joakim Lilliesköld

Linköping University Post Print



N.B.: When citing this work, cite the original article.

©2004 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Lars Taxén and Joakim Lilliesköld, Coordinating dependencies in global system development projects – The use of dependency diagrams, 2004, IEEE International Engineering Management Conference (IEMC 2004), Singapore , October 19 -21.

Postprint available at: Linköping University Electronic Press

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-111357>

Coordinating dependencies in global system development projects – The use of dependency diagrams

J. Lilliesköld¹, L. Taxén²

¹Department of Electrical Engineering, The Royal Institute of Technology

²Department of Science and Technology, Linköping University

Abstract—This paper addresses the problems that arise in the coordination of complex system development projects. The empirical setting is based on studies at Ericsson and ABB, and the total project manager’s instrument to manage and coordinate these projects. ABB used what they called dependency diagrams. Ericsson developed a method they call the anatomy concept. The approaches are used as a complement to the traditional work breakdown structure.

The paper evaluates the ability of these notations to address the needs of the total project management. The studies have shown that it is crucial to create compact high-level pictures of the resulting product and its projects in order to make the dependencies obvious to everyone involved (organizations as well as persons). The paper stresses that when developing complex system solutions the traditional diagrams easily become complex and unreadable. Thus, there is a need for supplementing approaches.

Keywords—System development projects, system anatomy, dependency diagram, total project management

I. INTRODUCTION

This paper addresses the problems that arise in the coordination of system development projects. The developmental efforts are usually organized in several projects involving different organizational units of the company spread globally; coordinated by a total project manager. System development implies uncertainties that require a shared understanding of the dependencies between parts of the system. In practice, these uncertainties imply frequent re-planning throughout the project. However, many of the project management books and tools of today suggest an approach based on the assumption that projects can be thoroughly planned and that changes of any kind are something you should avoid. This approach is characterized by an assumption that complexity can be reduced by breaking it down into pieces, and uncertainty can be reduced by rigorous planning. Therefore, the approach emphasized in many textbooks and tools does not work on the total project management level of these development efforts.

The paper evaluates tools like dependency diagrams as a complement to the Milestone diagram and Gantt chart; these approaches have been used successfully in companies like ABB and Ericsson when developing complex systems. The notations are evaluated with respect to their expressiveness and ease of use [1] i.e. to what extent they are able to present relevant information

in a simple manner. The information (questions) we seek the answer to have been found to be important in discussions with a total project manager; and the formulated set of questions have been iterated with other total project managers to secure that they are relevant when managing these development efforts.

II. DESCRIPTION OF NOTATIONS

In this article, we describe five notations. Work Breakdown Structure (WBS) and Gantt charts serve as examples of state-of-the-practice. In addition, the anatomy concept (the anatomy diagram and the integration plan) and the dependency diagram used at Ericsson and ABB [5, 7] are examined. Below, brief descriptions of these notations are given.

A. Work Breakdown Structure (WBS)

A work breakdown structure (WBS) is often performed as the first step in the planning process. The WBS is performed top-down, and is a product-oriented family tree subdivision of the hardware, service and data required to produce the end product [2, 3]. By breaking the work down into smaller elements there is a belief that risk and uncertainties are reduced since each level provides a greater probability that every major or minor activity will be accounted for.

Although a variety of work breakdown structures exist, the most common according to Kerzner [2] is a six-level indented structure. The top three levels are called the Managerial levels and they are: 1) Total program/project 2) Project 3) Task. The following levels are referred to as Technical level and they are: 4) Subtask 5) Work package 6) Level of effort.

According to Kerzner [2] the WBS is the single most important element in the traditional project management approach because it provides a common framework from which: the total project can be described as a summation of the subdivided elements; planning can be performed; time, cost and performance can be tracked; schedules and status-reporting can be established; etc.

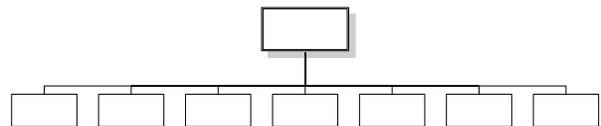


Fig. 1. Work breakdown structure [1]

Project managers normally manage at the top three levels of the WBS and also provide status reports at these levels [2].

B. The Gantt diagram

The Gantt (or bar) chart is one of the most frequently used project management notations [2, 4], and all major project management software tools provide support for its use. The main purpose of the diagram is to display activities against calendar time (or occasionally, money). Gantt charts exist in several variants, allowing the user to model slightly different phenomena.

Figure 2 constitutes an example of a Gantt chart. The diagram contains activities (also referred to as tasks) and their interdependencies. The activities are characterized by a name and duration, as well as a start and a finish date. The dependencies show that some activities can be performed in parallel while some necessitate sequential ordering. It can be noted that events (e.g. milestones) are not explicitly depicted in Gantt charts. However, since events are almost always planned to be the ending of a specific activity, they can, at least theoretically, be identified. As indicated earlier, Gantt charts have some additional features intended for, for example, project tracking. However, these features are not considered in the present exposé.

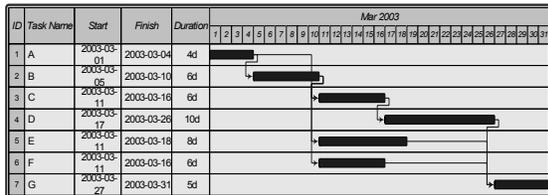


Fig. 2. Gantt chart

C. The System Anatomy

The anatomy construct is an attempt to model a system in a way which is more apt for complex system development tasks. In Figure 3, an anatomy of the same Otto engine as in Figure 1 is shown. The anatomy should be read from the bottom up. It shows the logical order and dependencies between functions when you start up the product, i.e. how to 'breath-life-in-a-system'. In order to start the engine, the start key must be pressed. This closes the circuitry between the battery and the start motor which begins to revolve. The crankshaft begins to turn which causes the ignition system to come alive and the oil to be distributed in the engine block. A mixture of air and fuel is ignited and the engine reaches the idling state [5].

The anatomy should show all of the major functional components of a product viewed from an integration and testability perspective. This is achieved with the focus on the necessary functions and their dependencies to start the product; the implementation of each function is subdued.

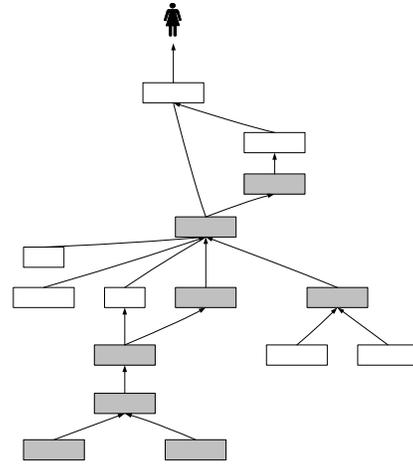


Fig. 3. System Anatomy [6]

D. The Integration Plan

The integration plan is based on the anatomy. When it is created, resources are assigned and dates for deliveries of the increments are settled. For each increment, traditional time and resource plans should be made as well.

The integration plan describes what is delivered, from whom, and when. It also clarifies the receiver for each internal delivery. Thus, it focuses on the dependencies and deliveries between subprojects. The focus on the dependencies and deliveries between every subproject clearly shows the impact of a delay at a delivery within the project since all the internal deliveries are somewhat related to the delivery of the final system to the customer [5].

In order to reduce complexity when many increments are involved, resources may be subdued in the integration plan. Often, this plan is 'tilted' compared to the anatomy in order to be aligned with a time line [5]

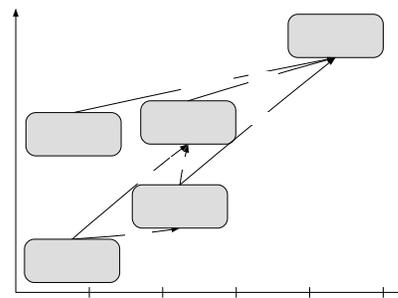


Fig. 4. Integration Plan

The anatomy is a product document that, in principle, might be the same from project to project even if different projects emphasize different aspects of it. Nevertheless, the anatomy is crucial when the integration plan is created or when discussions of the impact of changes are discussed. Thus, the integration plan is not useful if the anatomy does not exist, they interact in symbiosis.

E. Dependency diagram (The Drop Plan)

In a project studied, a dependency diagram for subproject releases was developed and deployed [7]. In the project, subprojects were required to work independently from each other. The dependency diagram served as a base for managing the overall project time schedule and controlling internal deliveries. The intention was to highlight the impact of subprojects' delays on the common project schedule. Figure 5, shows one of the dependency diagrams used in the project. It demonstrates the different product development activities, on the vertical axis to the left, as part of the system solution; the circles represent events in terms of releases, each one marked with the date of delivery and whether the delivery is part of an "alpha" or "beta" release. The arrows indicate which other releases the current one depends on and its influences.

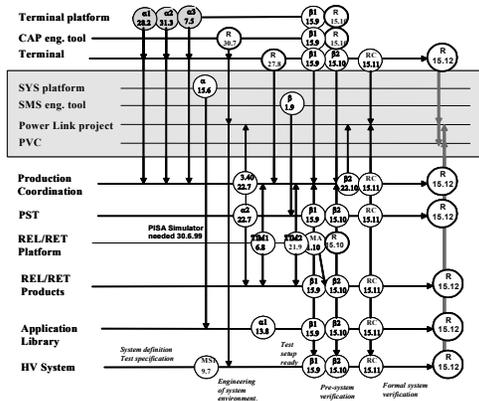


Fig. 5. Dependency diagram

A similar diagram was found in other projects as well, and was referred to as a drop plan. The drop plan however focused on the drops i.e. alpha, beta, and finally the release of the system, in an iterative develop environment. While the dependency diagram mixed products, customer project, coordination and test on the vertical axis; the drop plan focused on the different test sites and the pilot customer project who was receiving the product before it was released on the market. The deliveries were marked with an arrow; and information such as dates for delivery, what drop it belong to, and important milestones were marked as well in the picture.

III. EVALUATING THE NOTATIONS

In this section, we will present the condensed evaluation of the notations described above. In Table 1, the notations are found horizontally, and the questions that were found important to total project managers are enumerated vertically. The fairly coarse categories "*", "**", and "***" indicates whether or not the notation is able to answer the questions and how difficult it is to read out the answers; where "*" means that it is somewhat trivial, and "***" that some analysis of the notation (or in combination with other notations such as the WBS or Anatomy) has to be made in order to find the answer. The evaluation is based on Ekstedt et al [1] who elaborated on the concept of elegance. Briefly and roughly stated, elegance is the usefulness and simplicity of an abstraction. The "-" markings indicate that the notion is not capable of answering the question at all. In the table, the most suitable notation to answer a certain question is highlighted.

TABLE I - THREE POTENTIAL NOTATIONS TO BE USED BY THE TOTAL PROJECT MANAGER

	#	Question	Gantt (WBS)	Integration Plan (System Anatomy)	Dependency diagram
Time management	1	What increments and deliveries constitute the critical path?	**ii	*	-
	2	What is the lead time of the critical path?	**ii	*	-
	3	How long will increment I take?	*	*	-
	4	At what date does release R occur?	*	*	*
Dependency	5	On which tasks do increment I depend directly? On which tasks is increment I directly dependent?	*	*	***
	6	On which tasks do increment I depend indirectly? On which tasks is increment I indirectly dependent?	*	*	***
	7	On which increments or events is release R directly dependent? On which increment or events is release R indirectly dependent?	**	*	*
	8	Which increments can be performed in parallel?	***	*	***
	9	How can the dependencies between different organizations and responsibilities be clarified?	**	*	*
Delivery	10	Which supporting activities (sale & marketing material, training etc) must be performed in order to be successful with release R?	*	**	*
	11	Who is the receiver of delivery Y and who is the one delivering?	**	*	*
	12	What functionality can be delivered early?	**	*	***

Change mgmt	13	What is the impact of a certain change (external) to increment I or the project?	*** ⁱ	** ⁱ	*** ⁱ
	14	What is the impact of a certain change (internal) in increment I or the project?	*** ⁱ	** ⁱ	*** ⁱ
Communication	15	How can the project status effectively be communicated?	** ⁱⁱ	*	*
	16	How can the project dependencies and delivery dates be effectively communicated?	*** ⁱ	**	*
	17	How efficient is the notation when communicating with the steering group?	*** ⁱ	*	**
	18	How can the notation be used to provide a “big picture” throughout the project, i.e. a picture showing how different part fits in the system as a whole?	***	*	**
	19	How can the notation be used to provide a common picture of the project goals on the total project level?	***	*	**

- ⁱ Depending on the nature of the change, there is no correct answer to this question. The different levels refer to the experience that some of the notations are used as a support when changes are discussed.
- ⁱⁱ In the case of Total project management level, the Gantt chart becomes unreadable. In simpler project or in the sub-project level the answer would probably be another.

IV. DISCUSSION

The result shows that none of the model answers all the questions. Although, if all the questions need to be answered, the total project manager would have no choice but to deploy the anatomy concept and the dependency diagram. Of course, this is not at all surprising! No model can depict everything.

A way to refine the analysis somewhat would be to grade how important the questions are to the total project manager. It is not likely that he or she would regard all things as equally important, and would therefore probably be willing to compromise high complexity on certain questions in favor of ease of use in others. Further, it should be recognized that the questions we seeks the answers to in this evaluation are not entirely tuned in with the intentions of the creators of the different models.

When working with any kind of complex system, it is important to understand how the different parts interact. To create this understanding, we have to use different models of our problem. One single model cannot describe a project and its product from every point of view. As a matter of fact, an attempt to do so would fail, not only because the complexity of the task as such, but because any model quickly becomes unreadable due to the limited mind of most human beings. Instead, what we need are complementing views (models) of the same phenomena but with slightly different semantic content so that we are able to comprehend (some aspect of) the depicted problem. The models and their syntax can be more or less precise as long as the notation is able to express the aspect of the problem that a stakeholder may ask for. On the other hand, the syntax must be adapted to human cognition, understanding and problem solving. Without those two aspects any model will be of little or no use.

Thus, every complex project requires a set of perspectives to overview the situation, and the anatomy concept (i.e. the system anatomy and the integration plan) or the dependency diagram are useful methods to model

the reality. Other important views are still needed, for instance those creating a perspective of the system’s mechanical, physical or logical construction and different ways to configure the system; scenario pictures, descriptions of issues such as the ideas behind the project, the connection between the products market, the construction, production, expected profit, etc. Of course, many useful models already exist for software, hardware, and mechanical engineers; however, models clarifying the dependencies between those domains that can be used by project managers are not abundant.

A. Time management

The integration plan shows all of the necessary functions and increments that need to be developed in order to deliver the system to a customer at the end. It also shows important ‘collection’ points where several increments must be ready in order for the development effort to proceed further.

In the evaluation, it is not considered that with proper tool support, the difficulties of the critical path calculation in the Gantt chart may be transferred to a computer, and the questions (1 & 2) might easily be answered.

B. Dependencies

The WBS shows all the parts a product consists of, but it does not necessarily capture the crucial dependencies between the product parts, they are only implicitly controlled. In large projects, the network plans describing the dependencies become complex, and thus unforeseeable. This increases the risk of erroneous planning which may be reduced by using the anatomy as a complement to the WBS in the planning process, since the dependencies are explicit in the anatomy.

The integration plan focuses on the dependencies between subprojects. In combination with the anatomy it also shows what can be done in parallel.

C. Deliveries

The integration plan describes what is delivered, from whom and when. It also clarifies the receiver for each internal delivery. The focus on the dependencies and deliveries between every subproject in the integration plan clearly shows the impact of a delay at a delivery within the project, since all the internal deliveries are somewhat related to the delivery of the final system to the customer.

The dependency diagram clarifies the deliveries between the different development projects and the delivery project to the pilot customer; especially at the end of the project, when all the different part should be integrated and tested, this type of diagram has proven efficient.

The integration plan could show the necessary supporting activities (question 10); however, since the approach focuses on how the system comes alive, it is not obvious that these activities will be part of the integration plan.

D. Change management

When discussing changes, the anatomy concept provides a very efficient tool to communicate and discuss the effects of changes. A project manager described how the anatomy and the integration plans were laminated with plastic; and then whiteboard pens could be used in the project room when changes needed to be discussed. However, to be able to analyze the impact of a certain change, the integration plan alone does not contain all of the necessary information. It has to be used in combination with the anatomy.

When discussing changes Gantt chart easily becomes too complex. It was told in the discussions with total project managers that if the diagram was larger than one page, it was not useful by means of communication or discussions. The chart is documented to be good for static environments, and less useful in ever changing environment [8]. Nevertheless, on the level beneath the total project, i.e. projects and subprojects, the Gantt chart is preferred when planning [1] and controlling the project.

E. Communication

Different views, such as the notations described above, must evolve in cooperation within the project team. Based on these views, each project member will create their own mental model of the projects; from this model each project member acts and solves their part of the project. If the mental models of the people involved are not somewhat alike, more communication problems will be present than necessary.

In the case of a globally distributed project organization, there is a need to clarify the interactions between different organizations and the meaning of different deliveries; thus, there is a need for a common vocabulary or common understanding of milestones and internal deliveries. The dependency diagram has turned

out to be good tool for defining common language, i.e., common understanding of “alpha” and “beta” status of a product.

By means of communication, the dependency diagram and the integration plan were considered efficient when reporting the project status. In the studied cases, the notations were combined with traffic light control; i.e. colored green if the task was running according to schedule, yellow if there was a high risk of running late, and red for task running behind schedule.

V. CONCLUSION

The studies have shown that it is crucial to create compact high-level pictures of the resulting product and its projects in order to make the dependencies obvious to everyone involved (organizations as well as persons). A notation becomes unreadable when it is larger than one page.

This paper has shown that the models found in the studies when managing complex system development projects are more relevant from a total project manager perspective, than the traditional Gantt chart. Thus, they should be used as a complement to the traditional approach.

ACKNOWLEDGMENT

The authors would like to thank Mattias Nyrén and the other total project managers for their input to this paper.

REFERENCES

- [1] M. Ekstedt, P. Jonsson, J. Lilliesköld, and N. Jonsson, “Making Project Complexity Understandable – The Elegance of Notations”, *IAMOT’03 Proceedings*, 2003
- [2] H. Kerzner, “Project Management: A Systems Approach to Planning, Scheduling, and Controlling”, 7th ed., John Wiley & Sons, 2001
- [3] D. I. Cleland, L. R. Ireland, “Project Management – Strategic Design and Implementation”, 4th ed., McGraw-Hill, 2002
- [4] J. R. Meredith, S. J. Mantel, “Project Management – A Managerial Approach”, 3rd ed., John Wiley & Sons, 1995
- [5] J. Lilliesköld, L. Taxén, M. Karlsson, M. Klasson, “Managing complex development projects – using the system anatomy”, Submitted to HICSS 38
- [6] J. Anderstedt, U. Anderstedt, M. Karlsson, and M. Klasson, “Projekt och helhet – att leda projekt i praktiken” (in Swedish), Författares Bokmaskin, Stockholm, Sweden, 2002
- [7] M. Eriksson, J. Lilliesköld, N. Jonsson, and D. Novosel, “How to manage complex, multinational R&D projects successfully”, *Engineering Management Journal*, Vol. 14, No 2, pp. 53-60, 2002
- [8] H. Maylor, “Project Management”, 3rd Edition, Prentice Hall, 2002