

Management for Professionals

Mario Vanhoucke

# Integrated Project Management and Control

First Comes the Theory,  
then the Practice

 Springer

# Management for Professionals

For further volumes:

<http://www.springer.com/series/10101>



Mario Vanhoucke

# Integrated Project Management and Control

First Comes the Theory, then the Practice

 Springer

Mario Vanhoucke  
Faculty of Economics and Business  
Administration  
University of Ghent  
Gent, Belgium

ISSN 2192-8096 ISSN 2192-810X (electronic)  
ISBN 978-3-319-04330-2 ISBN 978-3-319-04331-9 (eBook)  
DOI 10.1007/978-3-319-04331-9  
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2014933586

© Springer International Publishing Switzerland 2014

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Printed on acid-free paper

Springer is part of Springer Science+Business Media ([www.springer.com](http://www.springer.com))

# Preface

Completing a project on time and within budget is not an easy task. Project monitoring and controlling systems should consist of processes that are performed to observe project progress in such a way that potential problems can be identified in a timely manner. When necessary, corrective actions can be taken to exploit project opportunities or to bring projects in danger back on track. The requisite is that project performance is observed and measured regularly to identify deviations from the project baseline schedule. Therefore, monitoring the progress and performance of projects in progress requires a set of tools and techniques that should ideally be integrated into a single decision support system. The understanding of the basic elements and concepts is a requisite to successfully use and implement the various project control concepts in an integrated project management and control system.

The topic of this book is the management of projects through the integration of baseline scheduling, schedule risk analysis and project control. *Schedule Risk Analysis* (SRA) is a technique that relies on Monte Carlo simulations to generate activity duration and cost deviations from their baseline values to assess the impact of these variations on the time and cost objectives of the project. *Project control* is done using Earned Value Management (EVM) and Earned Schedule (ES) techniques and are further referred to as EVM/ES systems. Both methods can be used to monitor the performance of projects in progress and to trigger corrective actions in case the project objectives are jeopardised. The *baseline schedule* acts as a central point of reference for both methods and is a crucial element for proper use of the EVM/ES and SRA systems. The integration between EVM/ES and SRA using baseline scheduling is referred to as *dynamic scheduling* or *integrated project management and control*, and is the topic of this book.

The “Integrated Project Management and Control” theme entered my life somewhere in the beginning of 2001 and has, ever since, never left my professional life. Through collaborative work with academics (PhD students, professors at other universities and/or business schools), consultants at different national and international companies and colleagues and friends from various Project Management (PM) organisations, the research and work on integrated project management and control has quickly moved forward leading to articles, books, conference speeches and

collaborations throughout the world. An overview of the many people with whom I collaborated is beyond the scope of this introduction chapter. In Vanhoucke (2013a), I have written how the collaborations have led to numerous results, including consultancy projects, the development of software tools and business games and many projects with project management professionals as well as academics.

One of the most visible results of the fruitful collaborations with friends and colleagues from the field is the writing of the two books, *Measuring Time* and *Dynamic Scheduling* published by Springer ([www.springer.com](http://www.springer.com)). The book *Measuring Time* (Vanhoucke 2010a) is a research book written for people interested in PM research and all technical details of the research study discussed in the book. It contains a detailed description of the methodology used in the research study and (too) many results are obtained by different computational experiments. Therefore, the content of this book requires some technical skills and knowledge on the project management and control topic and is mainly relevant for researchers and/or professionals with a lot of experience in managing and controlling projects.

The book *Dynamic Scheduling* (Vanhoucke 2012c) has a completely different focus. Rather than giving a detailed description of a research study, it gives an overview of the existing methods of dynamic scheduling in the literature. It explains its three dimensions, baseline scheduling, risk analysis and project control, in detail and serves very well as an introduction to this PM theme. The book is mainly intended to be a student handbook and is used in PM courses at universities and PM programmes at business schools. It should be perfectly understandable to people with little to no knowledge or experience in the project management and control theme. It only requires a great interest and deep passion for Project Management and the willingness and guts to walk through the different algorithmic aspects, case studies and formulas available in the PM literature.

And now, dear reader, you have a third book in your hands on exactly the same topic, and you might ask yourself the question whether it is really necessary to walk through the chapters of yet another book on the same topic. I asked myself that question before starting to write the different chapters of this book, but I believe the answer is clearly a “yes”. I truly believe that the book you are now holding in your hands is fundamentally different from my two previous books, and holds the middle ground between the methodological details of a research book and the general overview descriptions of a student handbook. Despite the huge amount of excellent articles and other books available in literature, I believe the contribution of this book is threefold as briefly illustrated along the following lines:

- Technical scope: Rather than a research book with methodological details on EVM/ES and SRA systems or a detailed overview book on the three dimensions of dynamic scheduling, this book serves as a more technical book containing formulas and mathematical details on the calculations of the EVM/ES and SRA metrics. In a first part of this book (*First comes the theory*), all mathematical formulas are shown and illustrated on three fictitious project examples. Moreover, summary reports are given to illustrate the use of metrics and concepts for project performance reporting purposes.

- Ready for use: The EVM/ES and SRA concepts presented in this book should be used in a project management and control setting by students, professionals and researchers. In the second part of this book (*Then comes the practice*), both a commercial software tool ProTrack and a research tool P2 Engine are presented and an overview of their basic features is given. The integration of these tools with spreadsheets should allow the readers to quickly access all the underlying mathematical details leading to a better understanding.
- Illustrative examples: All metrics, mathematical calculations and underlying concepts are illustrated on three simple example projects. These illustrative projects are small in size and are fundamentally different in structure in order to illustrate the full potential of the use of EVM/ES and SRA methods. No references to case studies that show the potential of the concepts in a real business environments are given, and the reader is referred to many other excellent books available in literature.

Therefore, this book on *integrated project management and control* should be useful for anyone who wants to understand the theory first, and then wants to translate the theory into practice using software tools. It is written for students, professionals and academics with an interest and/or experience in running projects as well as for newcomers in the area of project control with limited knowledge of the EVM/ES and SRA concepts. Honestly, it is mainly written upon request of many of my students – the young and enthusiastic students at universities, the MBA students at business schools and/or professionals at commercial trainings – who struggle with the use of the formulas in EVM/ES and SRA systems and need an overview of the mathematics and their use in one single document. Therefore, the book is split in three main parts. First comes the theory. The focus of this first part lies on the presentation of the metrics and calculations used in EVM/ES and SRA and on their use in reporting progress performance reports for controlling projects. General guidelines on their correct use in various business settings and hints for proper implementation in business software systems are however not within the scope of this part of the book. Then comes the practice. In this second part, the use of the EVM/ES and SRA methods is illustrated using two software tools. ProTrack is a commercial software tool that is freely available to students to get acquainted with the integrated project management and control theme using real project data or case studies. The P2 Engine tool is a more sophisticated version of ProTrack and is intended to be used by master students and PhD researchers and requires some basic scripting skills. Finally, a conclusion is reached that summarises the main contributions of this book and takes a look at the needs for further research in this interesting domain. Through the use of the three small illustrative example projects, the overview of all mathematical details (Part I) and the availability of two software tools (Part II), the reader should be able to recalculate almost every little detail presented in the book. Where necessary or appropriate, references to other books or articles are given in order to stimulate further reading, but no effort has been made to give a complete literature overview.



So, dear reader, why should you read this book? Or in case you think about reading all three of them, which one should you read first? I believe that in case you want a complete and detailed overview on the three dimensions of dynamic scheduling, you should definitely read the book on “Dynamic Scheduling” (Vanhoucke 2012c). In case you have quite some experience in the PM topics and you are interested in a more fundamental and intricate view on scheduling, risk and control, then go for “Measuring Time” (Vanhoucke 2010a). In case you have (little or a lot of) knowledge on the integrated project management and control theme and you need an overview of the available techniques for controlling your projects, then I suggest you should start reading the book that you are now holding in your hands! I am not claiming that you really have to read this book chapter per chapter. Instead, you better use it as a support tool when managing and controlling projects using your spreadsheet models or software tools. Whatever your level of expertise is, in case you need an overview and/or a quick refresh of the formulas and underlying mathematical details, this book might be a helpful tool for you. A lot of the material presented in this book has been presented elsewhere, in my previous books, in articles written by researchers and professionals and in the many excellent books available in literature. Much of the work presented in this book has also been made available online on one of our websites, blogs, online learning tools, or other sources. The main reason why we share a lot of our work with the outside world is because one of the main targets of teaching is to get students involved in the exciting world of project management. Putting material online is a way to communicate with the outside world and to bring students and professionals together. In this respect, this book does not contain a lot of new concepts but rather brings all the old material together in a single document. We strongly advise readers to stay up to date with our latest developments, research and business outcomes, for which more information can be found at the following websites:

- OR-AS: Most of the work bridging the gap between academics (done at Ghent University (Belgium) and Vlerick Business School (Belgium)) and the Project Management professionals is carried out in collaboration with OR-AS. Visit our OR-AS website and browse your way to the right and relevant content on [www.or-as.be](http://www.or-as.be).
- EVM Europe: The European organisation that brings professionals and researchers together to share new ideas, to stimulate innovative research and to advance the state of the art and best practices on integrated project management and control, welcomes any interested person on the yearly conference at [www.evm-europe.eu](http://www.evm-europe.eu).
- PM Knowledge Center (PMKC): PMKC is a free and online learning tool to stimulate interaction between researchers, students and practitioners in the field of project management. It contains articles and reference resources to inform and to improve the practice of dynamic scheduling and integrated project management and control. You can browse the content at [www.pmknowledgecenter.com](http://www.pmknowledgecenter.com).
- ProTrack: ProTrack is the software tool described in the second part of this book and is completely integrated with PM Knowledge Center. Thanks to its strong

focus on the integration of baseline scheduling, risk analysis and project control, it is the ideal learning tool to stimulate interaction between researchers, students and practitioners in the field of project management and dynamic scheduling. More information is available at [www.protrack.be](http://www.protrack.be).

- P2 Engine: P2 Engine is a command line utility tool based on the LUA scripting language to generate gigabytes of project data. It generates project baseline scheduling data and risk analysis metrics as well as dynamic project progress data that can be used for testing and validating novel research ideas. Tutorials and explanations on how to use the scripting language of P2 Engine can be found at [www.p2engine.com](http://www.p2engine.com).
- PSG: The Project Scheduling Game (PSG) is an IT-supported simulation game to teach young project management professionals the basic concepts of baseline scheduling, risk management and project control. It is used at universities, business schools and commercial trainings and is useful for people with an interest in project scheduling and control. The game can be ordered at [www.protrack.be/psg](http://www.protrack.be/psg).
- OR&S: A summary of the research done at the Operations Research and Scheduling (OR&S) Research group of the Faculty of Economics and Business Administration at Ghent University (Belgium) can be found at [www.projectmanagement.ugent.be](http://www.projectmanagement.ugent.be). Most of the activities are done in collaboration with Vlerick Business School and other international universities.

It is worth noting that all this information has been integrated in the ORASTalks app available in the Apple App Store and Google Play Store and can be downloaded via the links mentioned on the [www.or-as.be/orastalks](http://www.or-as.be/orastalks) website. Although this app has been created as a communication tool for students in Project Management and Operations Research, it also contains links to blogs, software tools and interesting websites that might be relevant to researchers and professionals too.

I would like to thank the people who have contributed, directly or indirectly, to this book. Thank you to Tom Van Acker and Stephan Vandevoorde for the collaborative work with OR-AS and EVM Europe, respectively. Without them, there certainly would not have been a new book. Thanks to Walt Lipke and Kym Henderson for sharing their thoughts on this interesting topic with me. A special thank you goes to (in alphabetical order) Jordy Batselier, Jeroen Colin, Louis-Philippe Kerkhove, Pieter Leyman, Len Vandenheede, Mathieu Wauters and Laura Willems for reading the draft version of this book. A million thankful words go to Gaëtane Beernaert for editing and motivating me along the writing process, and to Joyce and Thierry. To them, I owe more than I can express. And finally ... a thank you to anyone else I have met and forgotten to mention.

Finally, for the passionate people who are really, yes really, interested in Project Management and that hard collaborative work done throughout the years, a document is available online that provides a summary of the work done on PM. More precisely, in the book *The Art of Project Management: A Story about Work and Passion*, which is available at [www.or-as.be/books/wp](http://www.or-as.be/books/wp), an overview is given about the endeavours done in the past and the ideas that will be explored in the

future. It gives the reader a quick look behind the results. It tells about the products and ideas in Project Management and gives a brief overview of the most important people who inspired the OR-AS group for the work that has been done in the past. It does not look at the Project Management work from merely a research and teaching point of view, but also from a commercial point of view. It tells about work, and the passion that has led to the results of the hard work. It is not a scientific book. It is not a managerial book either. It is just a story . . . about work and passion.

Gent, Belgium

Mario Vanhoucke

# Contents

## Part I First Comes the Theory

<b>1</b>	<b>Project Data</b> .....	3
1.1	Fictitious Projects .....	3
1.1.1	Project 1: Cost Information System .....	6
1.1.2	Project 2: Patient Transport System .....	9
1.1.3	Project 3: Old Carriage House Renovation .....	11
1.2	Real Projects .....	12
1.3	Book Overview .....	13
1.4	Summary .....	15
<b>2</b>	<b>Earned Value Management</b> .....	17
2.1	The Key Parameters .....	18
2.1.1	Planned Value (PV) .....	18
2.1.2	Actual Cost (AC) .....	20
2.1.3	Earned Value (EV) .....	20
2.2	Performance Measurement .....	22
2.2.1	Time Performance .....	22
2.2.2	Cost Performance .....	26
2.3	Conclusions .....	27
	Appendix: EVM Performance Reports .....	28
<b>3</b>	<b>Earned Schedule</b> .....	33
3.1	What Is Wrong with Earned Value? .....	33
3.1.1	Time Is Money .....	34
3.1.2	Time Is Quirky .....	35
3.2	Earned Schedule .....	35
3.3	Performance Measurement .....	38
3.4	Conclusions .....	39

<b>4</b>	<b>Schedule Adherence</b> .....	41
4.1	Schedule Adherence .....	41
4.1.1	p-Factor .....	43
4.1.2	Graphical Illustration .....	46
4.2	Rework .....	47
4.2.1	Effective Earned Value .....	47
4.2.2	Schedule Deviations .....	49
4.3	Conclusions .....	50
	Appendix: ES Performance Reports .....	51
<b>5</b>	<b>Forecasting</b> .....	57
5.1	General Principle .....	57
5.1.1	General Formulae .....	58
5.1.2	Performance Factor .....	59
5.2	Forecasting Methods .....	60
5.2.1	Time Forecasting Methods .....	60
5.2.2	Cost Forecasting Methods .....	63
5.3	Forecasting Quality .....	65
5.3.1	Accuracy .....	66
5.3.2	Stability .....	68
5.4	Conclusions .....	70
	Appendix: Forecasting Reports .....	71
<b>6</b>	<b>Schedule Risk</b> .....	77
6.1	Measuring Risk .....	77
6.2	Schedule Risk Analysis .....	78
6.2.1	Baseline Schedule .....	79
6.2.2	Define Uncertainty .....	79
6.2.3	Simulate Progress .....	80
6.2.4	Sensitivity Analysis .....	80
6.3	Sensitivity Metrics .....	81
6.3.1	Time Sensitivity .....	82
6.3.2	Time/Cost Sensitivity .....	82
6.4	Conclusions .....	84
	Appendix: Schedule Risk Reports .....	85
<b>7</b>	<b>Schedule Control</b> .....	91
7.1	Level of Control .....	91
7.2	Top-Down Project Control .....	93
7.2.1	Management by Intuition .....	94
7.2.2	Management by Statistics .....	94
7.3	Bottom-Up Project Control .....	95
7.3.1	A Black-and-White View .....	96
7.3.2	Different Shades of Grey .....	96
7.4	Control Efficiency .....	99
7.5	Conclusions .....	101

**Part II Then Comes the Practice**

**8 ProTrack** ..... 105

8.1 Dynamic Scheduling on Your Desktop ..... 105

8.1.1 Baseline Scheduling ..... 106

8.1.2 Schedule Risk Analysis ..... 107

8.1.3 Project Control ..... 109

8.2 Project Scan ..... 111

8.3 Project Classification ..... 112

8.4 Conclusions ..... 113

**9 P2 Engine** ..... 115

9.1 Advancing the State of the Art ..... 115

9.1.1 Generating Data ..... 116

9.1.2 Project Characteristics ..... 118

9.1.3 Integrated Project Management and Control ..... 119

9.2 Chapter Reports in LUA ..... 120

9.2.1 General Functions ..... 120

9.2.2 Project Data ..... 121

9.2.3 Baseline Schedules ..... 122

9.2.4 EVM/ES Reports ..... 122

9.2.5 Forecasting Reports ..... 124

9.2.6 Risk Reports ..... 124

9.2.7 Download and ... Go! ..... 126

9.3 Conclusions ..... 126

**Part III To Reach a Conclusion**

**10 Conclusions** ..... 131

10.1 Summary ..... 131

10.2 Future Work ..... 133

10.2.1 Stability Versus Accuracy ..... 133

10.2.2 Improved Forecasting ..... 134

10.3 Concerted Research Action ..... 134

10.3.1 Static Integration ..... 135

10.3.2 Dynamic Integration ..... 136

10.3.3 Statistical Project Control ..... 136

**Bibliography** ..... 139

**Part I**  
**First Comes the Theory**

# Chapter 1

## Project Data

Monitoring the progress and performance of projects in progress using integrated project management and control methods requires a set of tools and techniques that should ideally be integrated into a single decision support system. In order to understand the different control techniques that will be presented throughout the various chapters of this book, all techniques and methods are numerically illustrated on fictitious and real project data. In this chapter, the project data are presented and the project key characteristics relevant for the subsequent book chapters are discussed.

In Sect. 1.1, three illustrative project examples are presented that will be used throughout most chapters of this book. The projects are fundamentally different in their network structure and have a known baseline schedule and project progress state. Section 1.2 discusses the presence of empirical data to validate the concepts and techniques discussed in this book. It should be clearly noted that empirical data are only used for illustrative purposes and are not considered a crucial element in this book. The practical use and relevance of most of the concepts have been discussed elsewhere in literature and will not be repeated here. Section 1.3 presents an overview of the content of all chapters of this book and displays a summary picture that will be used throughout most chapters. Section 1.4 concludes this chapter with a summary of the key characteristics of the project data and draws general conclusions.

### 1.1 Fictitious Projects

Throughout the various chapters of this book, three fictitious example projects will be used to illustrate all calculations. The three projects are based on empirical project data but are simplified and reduced in size to facilitate the calculations. For each project, the network data, the schedule data and the project progress are shown.



These three important aspects are given for each project in a standardised way as described along the following lines.

**Project network** It is assumed that each project has 17 activities, each with a duration expressed in weeks and a cost expressed in euro. Relations between activities are added to express precedences between activities that are inherent to each project. The number of relations and the way they are connected to activities result in a project network with a certain topological structure. Since it has been shown in [Vanhoucke \(2010a\)](#) that the topological structure of a project network plays a crucial role in the accuracy and efficiency of project control methods, three fundamentally different project networks will be used throughout this book. The topology of each project network is measured by the so-called Serial/Parallel indicator ([Vanhoucke et al. 2008](#)), further abbreviated as the SP indicator, and defined as

$$SP = \begin{cases} 1 & \text{if } n = 1 \\ \frac{m-1}{n-1} & \text{if } n > 1 \end{cases} \quad (1.1)$$

with  $m$  the number of activities on the longest path in the project network and  $n$  the number of activities in the network. The indicator  $SP \in [0, 1]$  measures the closeness of a network to a serial or parallel network. More precisely, when  $SP = 0$  then all activities are in parallel, and when  $SP = 1$  then the project network is completely serial. Between these two extreme values, project networks can have a structure that lies closer to either a serial or a parallel network. Hence, the SP indicator determines the maximal number of levels of the network, defined as the longest path (in terms of the number of serial activities) in the network. The topological structure of the three networks is varied from a relatively parallel network to a more serial network. More precisely, the SP values are equal to 0.25, 0.50 and 0.75 for Projects 1, 2 and 3, respectively.

The project network is graphically displayed by an activity-on-the-node (AoN) network showing the activities and the relations between these activities that define the topological structure. This AoN network format displays activities by nodes and precedence relations by arcs. It is assumed that all precedence relations are Finish-to-Start (FS) relations with a minimal time-lag of zero ( $FS^{min} = 0$ ). This implies that any activity can start immediately after the finish of its predecessor activities. Extensions to the three other types of relations, to positive or negative time-lags or to maximum time-lags are not taken into account. Using these extended types of relations would not lead to fundamental differences in the concepts presented in the other chapters of this book, but would lead to more calculations. [Table 1.1](#) displays a summary of these possible extensions, and a detailed overview can be found in [Vanhoucke \(2012c\)](#).

**Project schedule** The construction of a project baseline schedule using the network data involves the creation of a timetable with activity start and finish times and, as a result, defines the total duration of the project. This baseline schedule is a crucial step during project control since it will act as a point of reference for most, if not all calculations used throughout the remaining chapters.

**Table 1.1** Types of precedence relations

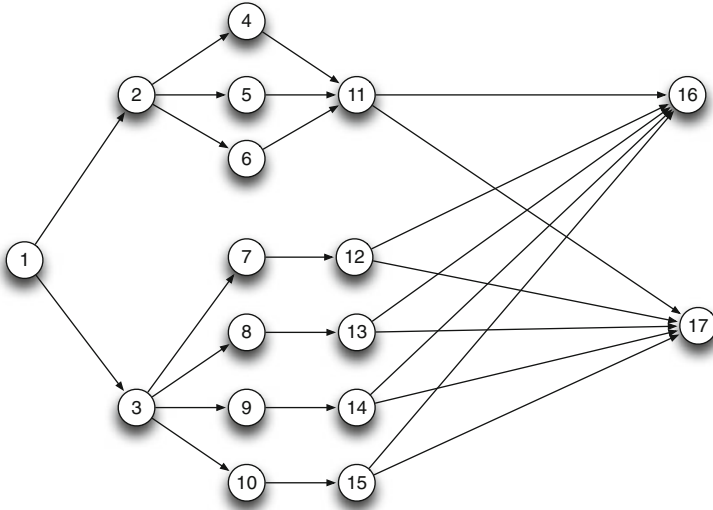
Type of relation	Time-lag			
	Zero		Nonzero	
	Min	Max	Min	Max
Start-to-start	$SS^{min} = 0$	$SS^{max} = 0$	$SS^{min} \neq 0$	$SS^{max} \neq 0$
Start-to-finish	$SF^{min} = 0$	$SF^{max} = 0$	$SF^{min} \neq 0$	$SF^{max} \neq 0$
Finish-to-start	$FS^{min} = 0$	$FS^{max} = 0$	$FS^{min} \neq 0$	$FS^{max} \neq 0$
Finish-to-finish	$FF^{min} = 0$	$FF^{max} = 0$	$FF^{min} \neq 0$	$FF^{max} \neq 0$

The project baseline schedule is visualised by a Gantt chart that consists of a list of activities and their corresponding start and finish times. The Gantt chart is displayed as an Earliest Start Schedule (ESS) obtained by using simple critical path principles in which each activity is scheduled as soon as possible within its degrees of freedom defined by the precedence relations.

Obviously, the construction of a baseline schedule can be the result of a much more complex scheduling approach, in which activities are scheduled later than their earliest start time using their slack. This can be useful when there is a limited availability of resources, or when activity splitting or activity constraints are used while constructing the baseline schedule. The list of possible extensions is long and diverse and is, without loss of generality, not taken into account. Instead, in this book, the construction of a project baseline schedule is considered as a necessary premise for sound and efficient project control, and detailed scheduling principles are therefore not discussed. For more details on the diversity of scheduling approaches, the reader is (again) referred to [Vanhoucke \(2012c\)](#).

**Project progress** Monitoring and controlling project progress is an important task for every project manager. Knowing the current status of a project relative to its baseline schedule and knowing when and how to intervene in a failing project is critical to its overall success. The purpose of this book is to present an overview of two alternative project control methods that act as triggers for these corrective interventions that should bring failing projects back on track before they become real disasters. To that purpose, the three fictitious projects are assumed to be finished and their project progress data are assumed to be known. All calculations in the remaining chapters will therefore be illustrated based on this known project outcome. The project progress reporting is assumed to be done on a weekly basis to show the progress of the project and its performance with respect to time and cost. The weekly progress data are completely artificial but will be used to show how project control indicators are calculated. For the sake of completeness, the first project is chosen to finish late, the second project finishes on time and the third project finishes earlier than expected. For all projects, the real cost exceeds the budgeted cost.

In the next paragraphs, each project will be discussed in detail along the project network, project schedule and progress dimension.



**Fig. 1.1** Project 1. Cost information system project ( $SP = 0.25$ )

### 1.1.1 Project 1: Cost Information System

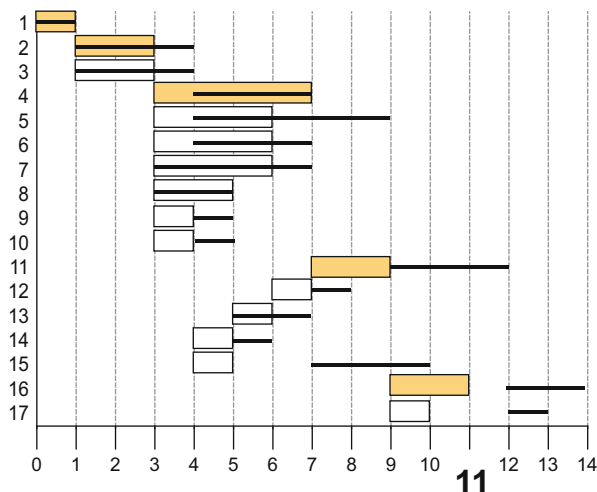
**Project network** Figure 1.1 displays the project network of a cost information system project at a producer of electromechanical components for car manufacturing. The scope of the project is to reduce customer complaints by better monitoring the condition of the company's molds (improper mold maintenance often resulted in downtimes). This is achieved by implementing a new database system that tracks the repair history, condition and scheduled maintenance for these molds. This system should also be able to suggest the optimal maintenance schedule and the time at which keeping a mold is no longer recommended. The project consists of three phases: the preparatory works, the implementation and the evaluation including possible rework. During the preparatory works the key objective is to determine the exact scope of the system to be developed. This was accomplished during simultaneous meetings with all end users. During the development phase the seven separate modules (databases and user interfaces) of the software are developed in parallel. Then in the final phase feedback is gathered from all groups of end users separately and the final product is completed. The project has started in June 2012 and the data given in this section are based on real-life data but have been adapted in order to get to 17 activities. The project schedule has originally been developed by students of Ghent University and was delivered as a ProTrack (see Chap. 8) file in December, 2012. The longest path consists of 5 activities ( $m = 5$ ) and hence  $SP = 4/16 = 0.25$ . The low value for the SP indicator shows that a large fraction of the activities can be executed in parallel. This is largely due to the fact that feedback

**Table 1.2** Project baseline and progress data of Project 1

Activity		Baseline			Progress		
ID	Name	Dur	Cost	Start	Dur	Cost	Start
1	Kick-off meeting	1	1,000	0	1	1,000	0
2	Define database specifications	2	15,000	1	3	22,500	1
3	Define user interface specifications	2	15,000	1	3	22,500	1
4	Develop maintenance cost database	4	6,000	3	3	4,500	4
5	Develop improvement cost database	3	3,900	3	5	6,500	4
6	Develop production time database	3	3,900	3	3	3,900	4
7	Develop IT department user interface	3	2,700	3	4	3,600	3
8	Develop accounting department user interface	2	2,000	3	2	2,000	3
9	Develop management user interface	1	1,500	3	1	1,500	4
10	Develop factory floor user interface	1	1,500	3	1	1,500	4
11	Database integration	2	6,000	7	3	9,000	9
12	End user feedback IT department	1	1,000	6	1	1,000	7
13	End user feedback accounting department	1	1,000	5	2	2,000	5
14	End user feedback management	1	1,000	4	1	1,000	5
15	End user feedback factory floor	1	1,000	4	3	3,000	7
16	Full product integration	2	4,000	9	2	4,000	12
17	Project evaluation	1	5,000	9	1	5,000	12

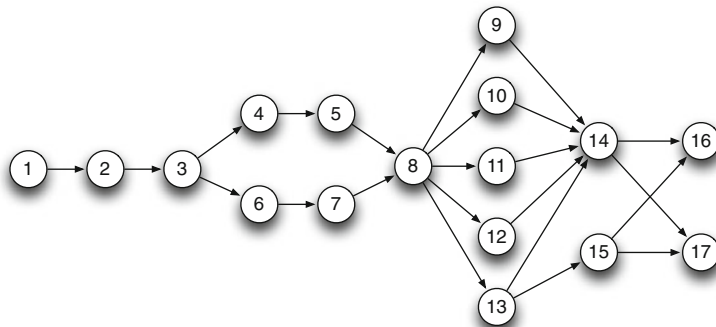
is requested from multiple user groups and the fact that the software itself consists of multiple components which can be developed simultaneously.

**Project schedule** Table 1.2 displays the data for the first project, split up in the baseline data and the data obtained during project progress. The information captured in the baseline schedule has a planned duration (PD) of 11 weeks and a budget at completion (BAC) = €71,500. The total cost of the project is the sum of all activity baseline costs given in the table. The baseline schedule Gantt chart is displayed in Fig. 1.2 by the bars at each activity row. The critical path is highlighted and consists of activities 1, 2, 4, 11, 16, resulting in the planned duration of 11 weeks.



**Fig. 1.2** Project 1. Gantt chart of baseline schedule and project progress

**Project progress** During project progress, some activities have a longer duration than originally planned (activities 2, 3, 5, 7, 11, 13 and 15) while others have a shorter duration (activity 4). Some precedence relations are not respected during project progress leading to activity overlaps that were not allowed in the baseline schedule (e.g. the relations 3–7 and 3–8 are violated). Moreover, some activities start later than their earliest start schedule. Activity 15 can start at week 5, i.e. immediately after the finish of activity 10, but only starts 2 weeks later. The real project progress is also displayed on the Gantt chart of Fig. 1.2 by the black lines at each activity row. The real duration of the project is equal to 14 weeks, denoting a 3 week delay, with a total real cost of €94,500. The real cost is equal to the sum of all real activity costs, and it is, without loss of generality, assumed that the activity cost follows a linear increase in time. More precisely, the weekly budgeted activity cost is calculated as the total activity baseline cost divided by its baseline duration. In case the activity duration deviates from the baseline duration, the increase or decrease in the real activity cost follows a linear trend with its real duration. The real cost is therefore calculated as the real activity duration multiplied by its weekly unit cost, and hence, shorter activity durations lead to lower costs and longer activity durations lead to higher costs. A possible example for which such a linear time/cost behaviour occurs is a project where the cost is expressed in € per man-hour and hence, deviations in activity duration have an immediate effect on the cost, due to an increase or a decrease in the total amount of man-hours to finish the particular activity. Although this reflects many real life situations in project management, one can consider other settings where the cost deviation has a completely different relation with the duration of an activity. Although these extended time/cost relations might reflect more realistic situations, it would not lead to fundamental differences in the project control concepts and calculations discussed in the remaining chapters.



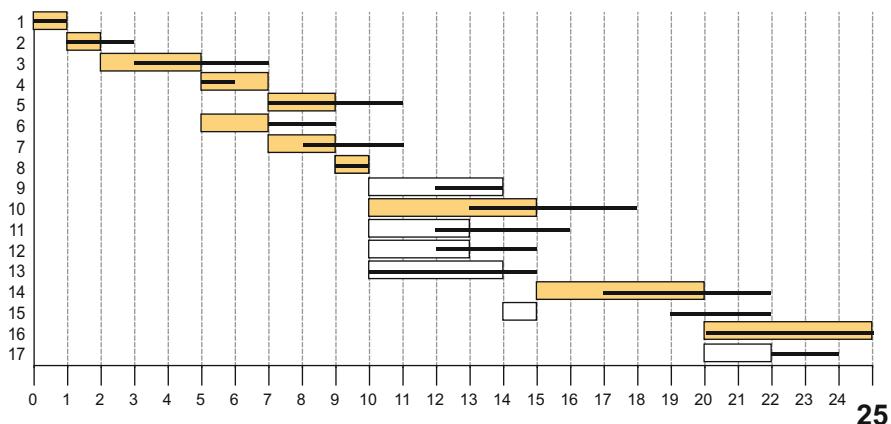
**Fig. 1.3** Project 2. Patient transport system project ( $SP = 0.5$ )

### 1.1.2 Project 2: Patient Transport System

**Project network** Figure 1.3 displays the project network for a patient transport system project in a hospital in Bruges (Belgium). The hospital was in need of a better way of organising patient transport, both in terms of hardware and in terms of software management tools. The project consisted of the installation of a new transport system for patients, which comprised both the physical transport system and the ICT infrastructure to track patient movement through the hospital. The project starts with the specification of the end user requirements and ends with the testing of the final product. This new software system should allow users to quickly locate key people when needed, greatly reduce administrative overhead as well as eliminate potential risk areas for medical mistakes. The project was started in January 2011 and the data given in this section are based on real-life data but have been adapted in order to get to 17 activities. The project schedule has originally been developed by students of Ghent University and was delivered as a ProTrack file in December, 2012. The longest path consists of 9 activities and hence,  $m = 9$  and  $SP = 8/16 = 0.50$ . The value of the SP indicator reveals that the project is situated precisely in the middle between a serial and parallel project. This is largely due to the fact that the activities for the physical transportation system and the ICT system can happen mostly in parallel, while many other activities must be executed in series.

**Project schedule** The information captured in the baseline schedule has a planned duration (PD) of 25 weeks and a budget at completion (BAC) = €137,000. The baseline information is displayed by the bars of each activity row of Fig. 1.4 and details are given in Table 1.3. The project has two critical paths as highlighted in the Gantt chart consisting of activities 1, 2, 3, 4, 5, 8, 10, 14, 16 for the first path and 1, 2, 3, 6, 7, 8, 10, 14, 16 for the second path.

**Project progress** The project progress is displayed by the black lines at each activity row of the Gantt chart in Fig. 1.4 and shows that the project finishes exactly on time (25 weeks). Despite this positive project outcome, the intermediate progress

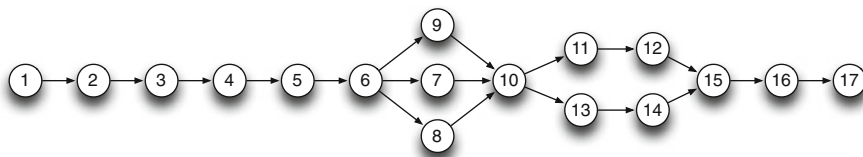


**Fig. 1.4** Project 2. Gantt chart of baseline schedule and project progress

**Table 1.3** Project baseline and progress data of Project 2

Activity		Baseline			Progress		
ID	Name	Dur	Cost	Start	Dur	Cost	Start
1	Staff needs survey	1	5,000	0	1	5,000	0
2	ICT committee meeting	1	2,500	1	2	5,000	1
3	Requirements study	3	2,100	2	4	2,800	3
4	Request transport system supplier proposals	2	3,000	5	1	1,500	5
5	Negotiate best transport system offer	2	10,000	7	4	20,000	7
6	Request database supplier proposals	2	3,000	5	2	3,000	7
7	Negotiate best database offer	2	10,000	7	3	15,000	8
8	Kick-off meeting with staff	1	6,500	9	1	6,500	9
9	Installation of transport system	4	24,000	10	2	12,000	12
10	Installation of server	5	2,500	10	5	2,500	13
11	PC training for nursing staff	3	3,300	10	4	4,400	12
12	Code list creation	3	5,700	10	3	5,700	12
13	Filling the database	4	2,400	10	5	3,000	10
14	Practical training for nursing staff	5	45,000	15	5	45,000	17
15	Final check	1	1,500	14	3	4,500	19
16	Live pilot period	5	500	20	5	500	20
17	Project evaluation	2	10,000	20	2	10,000	22

is not according to the baseline schedule. Two activity durations are shorter than planned (activities 4 and 9) while most of the activities have a longer duration (activities 2, 3, 5, 7, 11, 13 and 15). Despite these delays, the project is completed



**Fig. 1.5** Project 3. Old carriage house renovation project (SP = 0.75)

within the predefined 25 weeks horizon since most precedence relations are violated leading to many activity overlaps during progress (relations 3–4, 5–8, 6–7, 7–8, 10–14, 14–16 and 15–16). The total real cost of the project is equal to €146,400, and it is again assumed that the activity costs deviate from the baseline cost in a linear relation to its baseline duration (see the description of the previous project).

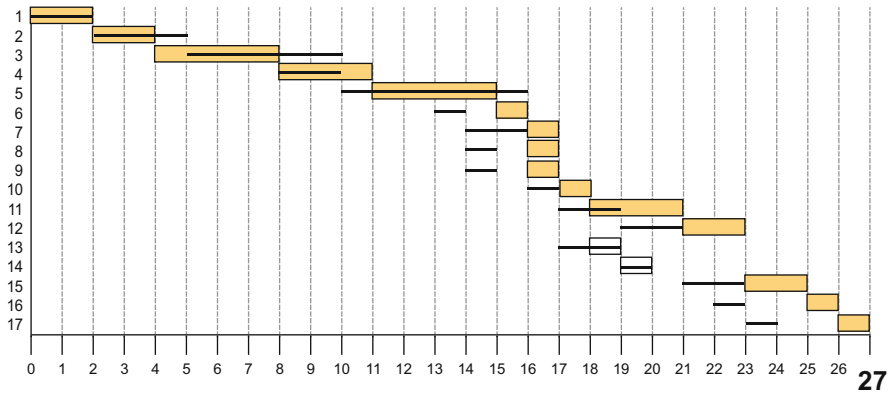
### 1.1.3 Project 3: Old Carriage House Renovation

**Project network** Figure 1.5 displays the network of a renovation project. The project is a renovation of an old carriage house that is currently used as bicycle storage, transforming it into a day care center accommodating 14 children (age 0–3). The project is owned by Ghent University (Belgium) and fits within the context of enabling employees – especially female PhD students – to combine their career with a family. Since it concerns a day care, many regulations about child safety have to be met. Also, the building is of high architectural value, thus it is requested to renovate carefully and to preserve as much of the original structure as possible. The technical specifications of the project can be briefly summarised as a one-storey building with a total floor area to be renovated of 236 m<sup>2</sup>, i.e. 131 m<sup>2</sup> on the ground floor (of which 21 m<sup>2</sup> new building) and 105 m<sup>2</sup> on the first floor. The project data given in this section are based on real-life data but have been adapted in order to get to 17 activities. The project schedule has originally been developed by students of Ghent University and was delivered as a ProTrack file on December 12, 2012. The longest path consists of 13 activities and hence,  $m = 13$  and  $SP = 12/16 = 0.75$ .

**Project schedule** The baseline schedule is displayed in Fig. 1.6 and has a total planned duration of 27 weeks and a budget at completion of €443,000. Due to the highly serial structure of the project network, most activities are critical. Only activities 13 and 14 have a positive slack value and do not lie on the critical path. Detailed information is given in Table 1.4.

**Project progress** The black lines at each activity row of the Gantt chart in Fig. 1.6 show that the project finishes 3 weeks earlier than expected (24 weeks). Activities 2, 3, 5, 7 and 13 have a longer duration than planned while activities 4 and 11 have a shorter duration than expected. Thanks to the activity overlaps between activities 3–4, 5–6 and 15–16, the project finishes earlier than originally stipulated in the baseline schedule. The total project cost is equal to €529,000.





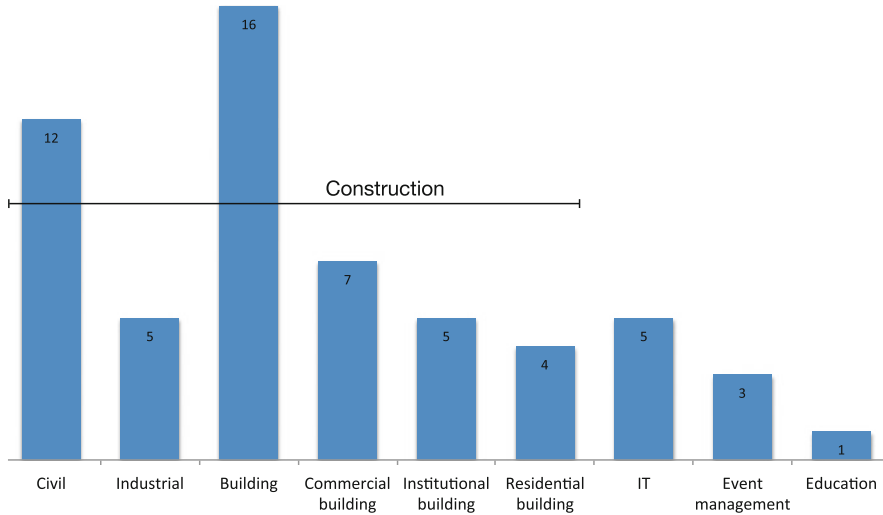
**Fig. 1.6** Project 3. Gantt chart of baseline schedule and project progress

**Table 1.4** Project baseline and progress data of Project 3

Activity		Baseline			Progress		
ID	Name	Dur	Cost	Start	Dur	Cost	Start
1	Demolition phase 1: roof & utilities	2	20,000	0	2	20,000	0
2	Demolition phase 2: masonry & floor	2	20,000	2	3	30,000	2
3	Structural work & sewage	4	120,000	4	5	150,000	5
4	Facade renovation	3	30,000	8	2	20,000	8
5	Roofing	4	80,000	11	6	120,000	10
6	Exterior joinery	1	25,000	15	1	25,000	13
7	Electricity	1	10,000	16	2	20,000	14
8	Plumbing	1	15,000	16	1	15,000	14
9	Heating & ventilation	1	25,000	16	1	25,000	14
10	Floor insulation & screed	1	5,000	17	1	5,000	16
11	Plasterboard	3	12,000	18	2	8,000	17
12	Devices & connections	2	20,000	21	2	20,000	19
13	Suspended ceiling	1	10,000	18	2	20,000	17
14	Interior joinery	1	15,000	19	1	15,000	19
15	Painting	2	6,000	23	2	6,000	21
16	Linoleum flooring	1	10,000	25	1	10,000	22
17	Furniture & cleanup	1	20,000	26	1	20,000	23

## 1.2 Real Projects

Although all metrics used in this book are shown and illustrated on the three fictitious project examples, the reader should not turn a blind eye to the practical relevance of the integrated project management and control techniques. For this very reason, the software tools of Chaps. 8 and 9 are used in research studies in



**Fig. 1.7** Empirical project database (status in 2013)

which empirical project data are used from various sectors to validate the use and relevance of existing and newly developed project control techniques.

Figure 1.7 displays the total number of projects of the current project research database consisting of 42 projects with SP values ranging between 0.01 and 0.82. This project database has been created in 2013 and will continuously be updated with new projects during the coming years. Currently, the projects mainly find their origin in the construction industry, but projects from IT, event management and education are part of the database as well. In this book, no special attention will be allocated to the practical validation of the discussed topics using this empirical project database. The reader is referred to Sect. 8.3 for a short discussion on the classification of the projects in the database.

### 1.3 Book Overview

Figure 1.8 gives a visualisation of the content of all chapters of the first part of this book. The figure displays the different methods of an integrated project management and control system and will be further discussed in the upcoming chapters of Part I of this book.

The central block of the picture is labelled “Integrated Project Management and Control” which is the central theme of the book. Each block of the picture is labelled with a number which is a reference to the chapter ID where the specific topic is discussed. The reference to Chap. 7 displayed in the central block refers to the two

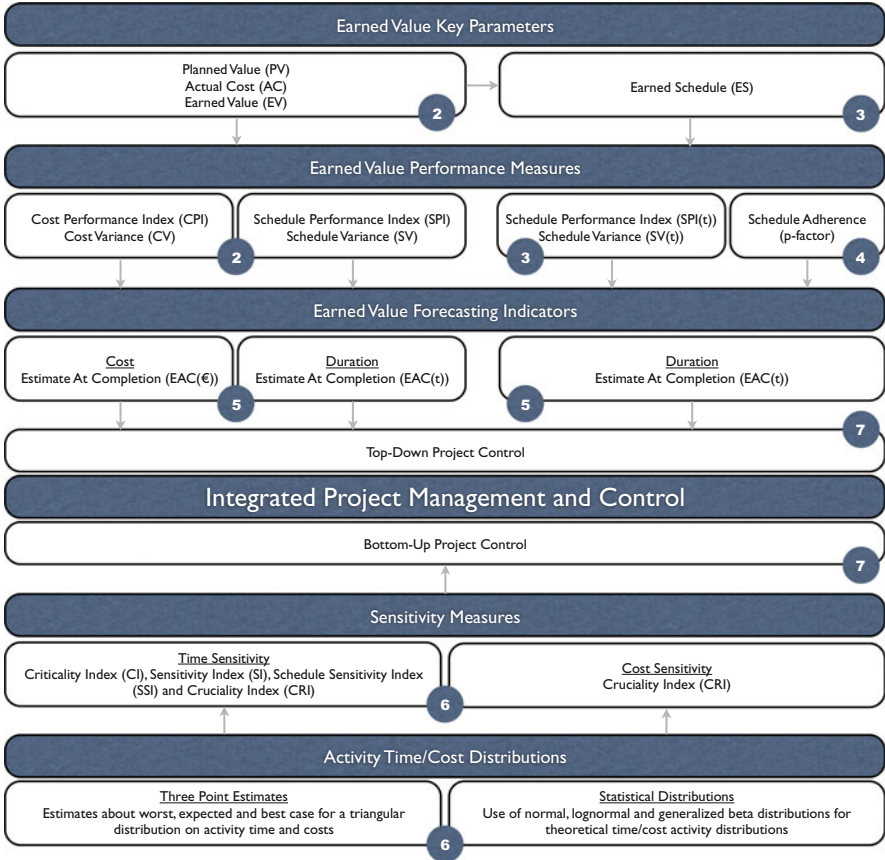


Fig. 1.8 Overview of Part I of this book

alternative project control methods that will be presented using all topics discussed in Chaps. 2–6.

All blocks displayed above this central block will be discussed in Chaps. 2–5. These chapters will make use of a well-known project control technique, known as Earned Value Management (EVM), extended with the more recently developed Earned Schedule (ES) method. Chapter 2 gives an overview of the key metrics used in an EVM system and their use for calculating the time and cost performance of projects in progress. Chapter 3 extends the three key metrics with a fourth ES key metric, and its use to calculate an alternative time performance metric will be shown. Unlike the time performance measurement, no alternative for cost performance measurement is presented. In Chap. 4, the ES metric will be used to calculate the adherence of the project baseline schedule, and a concept known as the p-factor approach is presented. Chapter 5 shows how the EVM and ES performance metrics can be used to forecast the final duration and cost of a project.

The EVM performance metrics can be used to predict the expected duration and cost of a project in progress, while the ES performance measures can only be used to calculate an alternative prediction of the expected duration of the project. Due to the close relation and similarities between the EVM and ES methods, they are often referred to as the EVM/ES method. This EVM/ES method will be used in the so-called *top-down project control* approach of Chap. 7.

The part below the central block will be discussed in Chap. 6. The so-called Schedule Risk Analysis (SRA) technique is used to measure and assess the impact of variation in activity time and cost estimates on the project objectives. The various metrics and measures discussed in this chapter will then be used in the so-called *bottom-up project control* approach of Chap. 7.

In order to clarify the structure at each chapter of this book, parts of the overview picture will be copied and displayed at the beginning of each chapter. Therefore, this picture must be regarded as a puzzle in which every piece will be explained in detail throughout the upcoming chapters, finally ending at the “Integrated Project Management and Control” theme in the final chapter of Part I of this book. In Part II, all blocks of Fig. 1.8 will be used by two software systems, ProTrack and P2 Engine.

## 1.4 Summary

In this chapter, the project data that will be used throughout the remaining chapters of this book are presented. The database consists of three fictitious diverse projects and a continuously growing empirical dataset with projects taken from various sectors. The fictitious projects take a central place in this book and will be used to illustrate all concepts and calculations of the next chapters. The empirical database only serves as a validation and illustration tool to put the concepts, tools and techniques into the right perspective.

For the sake of simplicity, the use of the fictitious project dataset is subject to assumptions and simplifications. However, straightforward extensions to more general project data could have been applied, as summarised along the following lines:

- Network data: The precedence relations between activities are finish-to-start relations with a minimal time-lag of zero, and therefore, more generalised precedence relations are not taken into account.
- Baseline schedule: The baseline schedule is constructed using simple critical path calculations resulting in an earliest start schedule. No extensions to resource levelling techniques and extended scheduling features such as activity splitting and the presence of time constraints are used.
- Progress: The real activity costs are assumed to be linearly dependent on their real durations and hence a linear trade-off between the activity duration and cost is assumed.

Obviously, these assumptions and simplifications are not taken into consideration for the empirical project data since all empirical data are gathered from real project

progress where the baseline and progress data are gathered from the project planning tool ProTrack.

Throughout the description of the project data in this chapter, a number of fundamental concepts have already been introduced that will take a central place throughout the remaining chapters. More precisely, it has been shown that the construction of a baseline schedule leads to a total project duration and total budgeted cost. These concepts are known in project control as:

- **Planned Duration (PD):** The duration of the baseline schedule given by the finish time of the latest activity.
- **Planned Cost, better known as Budget at Completion (BAC):** The total cost of the project's baseline schedule given by the sum of all activity costs.

During project progress, activities are subject to deviations from the original baseline schedule estimates leading to lower or higher durations, lower or higher costs and changes in their start and finish times, as follows:

- **Real Duration (RD):** The real duration of the project at its completion. The project can finish ahead of schedule ( $RD < PD$ ), on time ( $RD = PD$ ) or late ( $RD > PD$ ).
- **Real Cost (RC):** The real total cost of the project at its completion, which can be below ( $RC < BAC$ ), on ( $RC = BAC$ ) or above ( $RC > BAC$ ) the original budget at completion of the baseline schedule.

While the PD and BAC are known before the start of the project, the RD and RC are only known upon project completion. They can however, be estimated during the life of the project using the techniques that will be discussed in Chap. 5.

The project data given in this chapter are crucial to the other chapters since they will be used to illustrate the various control concepts. Therefore, the three fictitious projects will be used to calculate the various metrics and measures and summaries will be given in project management and control reports at the end of most chapters. The empirical project data will not be used for calculations but rather for illustrative purposes in order to put general conclusions in the right perspective.

## Chapter 2

# Earned Value Management

Earned Value Management (EVM) is a methodology used to measure and communicate the real physical progress of a project and to integrate time, cost and scope into a single project management and control system. It takes into account the work completed, the time taken and the costs incurred to complete the project and it helps to evaluate and control project risks by measuring project progress in monetary terms. The basic principles and the use in practice have been comprehensively described in many sources (Fleming and Koppelman 2005). An EVM system consists of a set of metrics and formulas that can be divided into three categories. Three *key metrics* will be used to assess the progress of the project as calculated by the *performance measures* and eventually *forecasting methods* will then translate these performance measures into predictions about the expected final time or cost objectives of the project.

A graphical overview of the three levels of an EVM system is given in the top part of Fig. 1.8. However, this chapter is restricted to the first two levels of an EVM system, i.e. only the key metrics and their use for performance measurement will be shown. The use of forecasting methods is not within the scope of this chapter but will be discussed in Chap. 5. Figure 2.1 displays the relevant parts of Fig. 1.8 that will be discussed in this chapter. In Sect. 2.1, the three key metrics of EVM, known as the Planned Value (PV), Actual Cost (AC) and Earned Value (EV) are discussed and illustrated on the example project data. Section 2.2 makes use of the key metrics to analyse the time and cost performance of projects in progress at periodic review periods. Section 2.3 gives a summary of the main topics of this chapter and draws general conclusions. The appendix of this chapter shows the summary performance reports of the three example projects introduced in the previous chapter.

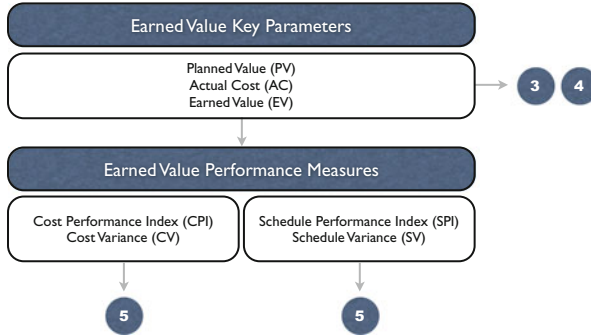


Fig. 2.1 EVM piece of the jigsaw puzzle of Fig. 1.8

## 2.1 The Key Parameters

The input parameters for an EVM system consist of three key metrics that represent different cost values for each project activity and that can be summed up to three values expressed in monetary units at the project level. These key metrics will be used to calculate the performance of a project in progress and to make time and cost predictions about the future. Their values must be collected at regular time intervals or review periods at which the status of the project in progress is monitored and controlled. In the remainder of this book, it is assumed that the project progress is measured each week along the time horizon of the project between its start and finish, as previously explained in Chap. 1 on the three example projects.

### 2.1.1 Planned Value (PV)

The *Planned Value* (PV) is the time-phased budget baseline as an immediate translation of the baseline schedule constructed from the project network. It shows a cumulative increase in the total budgeted activity cost given the start and finish times stipulated in the baseline schedule. The planned value is often called *budgeted cost of work scheduled* (BCWS).

The planned value line has a one-to-one link with the baseline schedule since it shows the cumulative cost increase as proposed by this schedule. Consequently, the construction of this schedule is key to the success of controlling the project in progress, since it acts as a point of reference during all performance measurements and forecasting calculations. Ideally, this baseline schedule should not be changed throughout the life of the project in progress in order to keep the point of reference fixed. However, when the project performance deviates dramatically from the original baseline schedule (due to e.g. changes in the project scope), it might

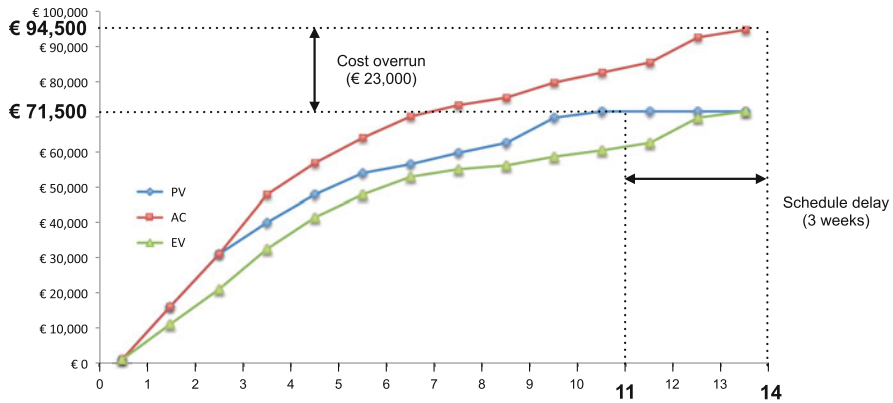


Fig. 2.2 S curve with the three EVM key metrics (Project 1)

be worth the consideration to re-baseline and hence to reset the point of reference for future project control.

Although most concepts of a baseline schedule have been introduced earlier in Chap. 1, its key outputs are summarised along the following lines:

- PD           Planned Duration of the project  
→ known from the baseline schedule  
= SAC (Schedule At Completion)
- BAC         Budget At Completion  
→ known from the baseline schedule  
= PC (Planned Cost)
- PV           Planned Value  
→ known from the baseline schedule  
= S-curve between point (0,0) and (PD,BAC) on Fig. 2.2

In Fig. 2.2, the planned value curve for Project 1 is shown as a result of the construction of the baseline schedule displayed in Fig. 1.2. The planned duration (PD) is equal to 11 weeks with a budget at completion (BAC) of €71,500. The importance of this planned value line is that it will act as a crucial point of reference throughout the performance measurements when the project is in progress.

It has been mentioned in Chap. 1 that the increase in the activity costs is assumed to be a linear increase per activity time unit. The planned value line is a direct consequence of this assumption since it displays the cumulative total cost increase of all activities along the planned life of the project. The Gantt chart shown at the top of Fig. 2.3 displays this linear increase in planned activity costs for each activity. The values in the bars show the weekly activity cost, which is equally divided over the weeks of the activity duration. The total project cost is equal to the sum of all activity costs for every week of the planned project duration.



### 2.1.2 Actual Cost (AC)

The *Actual Cost (AC)* is often referred to as the *actual cost of work performed (ACWP)* and is the cumulative actual cost spent at a given point AT in time. This AC does not necessarily share a relation with the planned cost increase shown in the PV curve but instead is a measure for the real expenditures made for the work done at the current review period. The following abbreviations will be used throughout all chapters:

AT	Actual Time → current review period = number of time periods the project is in progress
AC	Actual Cost → known at each review period = total activity cost made at the current period AT
RD	Real Duration of the project → known at project completion = indicates a project finish ahead, on or behind schedule
RC	Real Cost of the project → known at project completion = indicates a project finish under, on or over budget

In Fig. 2.2, the actual cost curve for Project 1 is shown at each weekly review period from the start of the project until the project finish at week RD = 14 (i.e. 3 weeks delay). The cumulative increase in costs illustrates that the real costs (RC) is equal to €94,500, indicating a project overrun of €23,500. As mentioned in Chap. 1, it is assumed, without loss of generality, that the real cost for an activity is calculated as the real activity duration multiplied by its weekly unit cost, and hence, shorter activity durations lead to lower costs and longer activity durations lead to higher costs. The Gantt chart shown in the middle of Fig. 2.3 displays this linear increase in real activity costs for each activity, since all cost values are equally divided along the real duration of the activity. The total project cost is equal to the sum of all real activity costs for every week of the real project duration.

### 2.1.3 Earned Value (EV)

The *Earned Value (EV)* represents the amount budgeted for performing the work that was accomplished at a given point AT in time. It is often called the *budgeted cost of work performed (BCWP)* and equals the total activity (or project) budget at completion multiplied by the percentage activity (or project) completion (PC) at the particular point in time (= PC \* BAC). The following abbreviations will be used throughout all chapters:

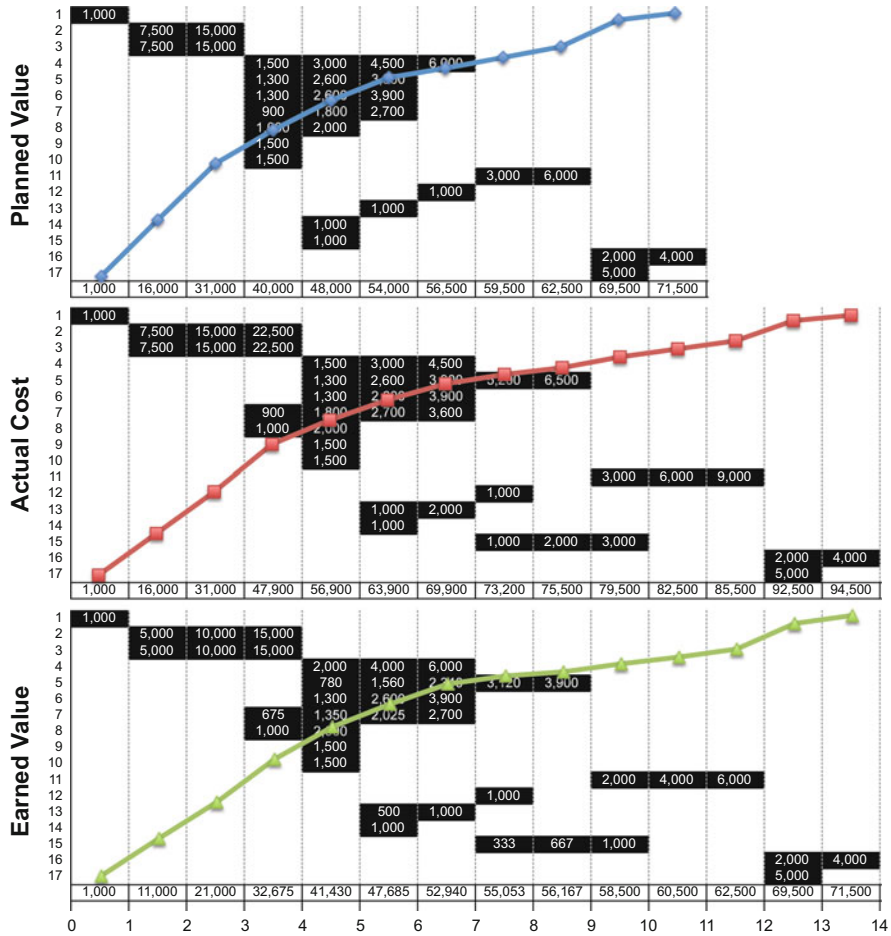


Fig. 2.3 Project 1. The three EVM key metrics (PV, AC and EV)

- PC            Percentage Completion of an activity or the project  
               → estimated at each review period  
               = Indication of physical project progress
- EV            Earned Value  
               → calculated at each review period as  $EV = PC * BAC$   
               = fraction of the PV earned at AT for the work done

In Fig. 2.2, the earned value curve for Project 1 is shown at each weekly review period from the start of the project until the project finish at week RD = 14. It is again assumed that the estimated percentage completion for each activity follows a linear increase in time, i.e. the  $PC = 33\%$  for an activity in progress for 1 week and with an estimated real duration of 3 weeks. The detailed increase in EV for

each activity and for the total project is shown in the bottom picture of Fig. 2.3. Obviously, the maximum value for EV is equal to BAC (i.e. at PC = 100 %) and is reached at the project finish RD = 14 weeks.

The PV, AC and EV metrics are the key metrics of a project in progress and are used to measure the performance at each review period as well as to forecast the real duration and cost of the project.

As mentioned, Fig. 2.2 displays these three key metrics for Project 1 and details are given in Fig. 2.3. The detailed calculations for Project 2 and Project 3 are shown in Figs. 2.4 and 2.5. The reader should be able to calculate every number in these pictures with ease when using the project data given in Chap. 1. Small differences might occur due to rounding operations.

## 2.2 Performance Measurement

Measuring the performance of a project in progress is key to the success of the monitoring process since it provides information on the current time and cost objectives compared to the baseline schedule and it might act as an early warning signal and trigger to take corrective actions. Consequently, EVM makes use of a set of performance indices to validate the performance of the project at a current moment in time.

### 2.2.1 Time Performance

The time performance of a project in progress is measured by two performance metrics, known as the schedule variance (SV) and the schedule performance index (SPI). Both metrics measure the deviation between the value of the work done (EV) at the current moment AT and the value of the work planned (PV) at that moment. The schedule variance is measured in absolute monetary units while the schedule performance index expresses the time performance as a percentage of the expected baseline schedule performance, as summarised along the following lines:

SV	Schedule Variance → calculated at each review period as $SV = EV - PV$ <ul style="list-style-type: none"> <li>&lt; 0: project behind schedule</li> <li>= 0: project on schedule</li> <li>&gt; 0: project ahead of schedule</li> </ul>
SPI	Schedule Performance Index → calculated at each review period as $SPI = EV/PV$ <ul style="list-style-type: none"> <li>&lt; 1: project behind schedule</li> <li>= 1: project on schedule</li> <li>&gt; 1: project ahead of schedule</li> </ul>

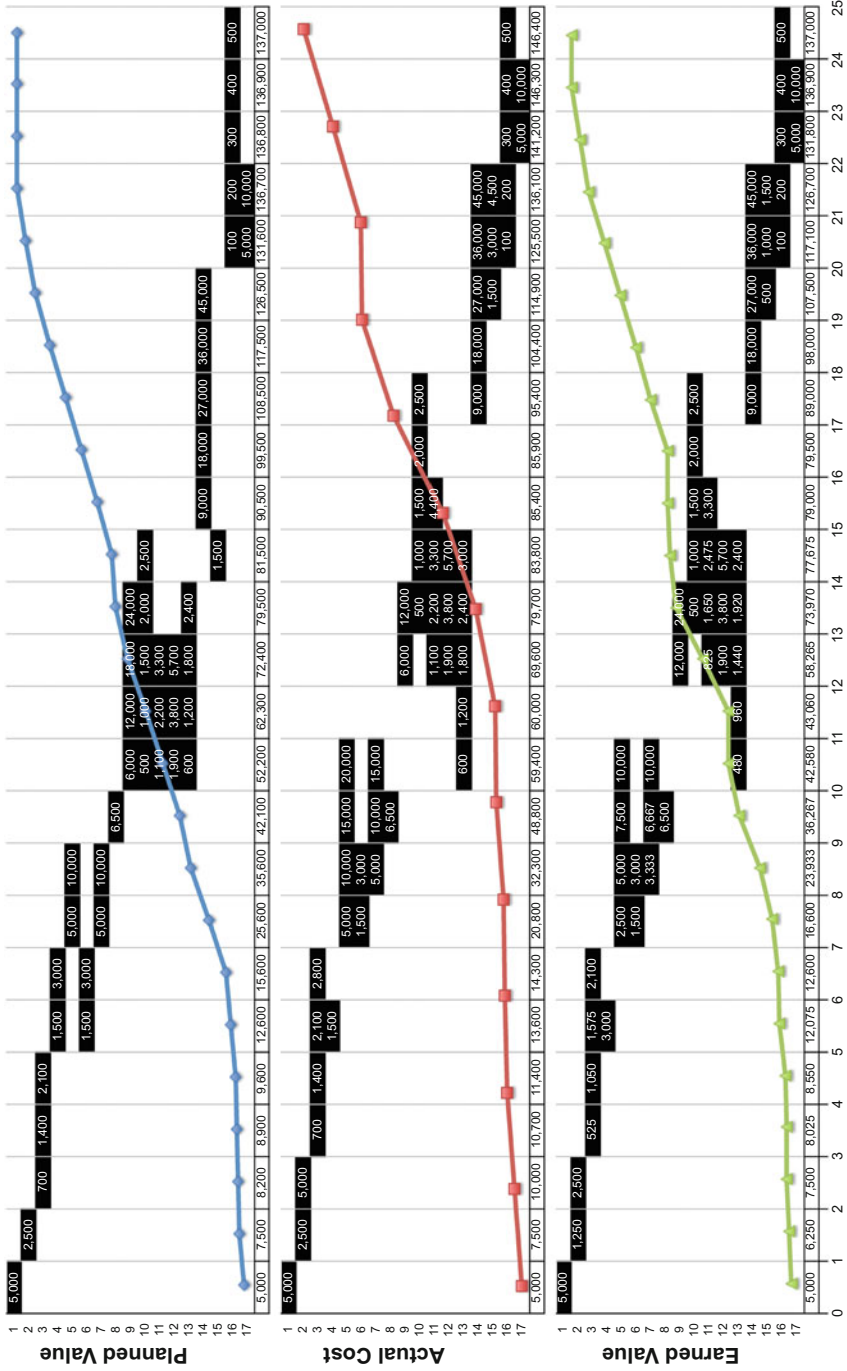


Fig. 2.4 Project 2. The three EVM key metrics (PV, AC and EV)

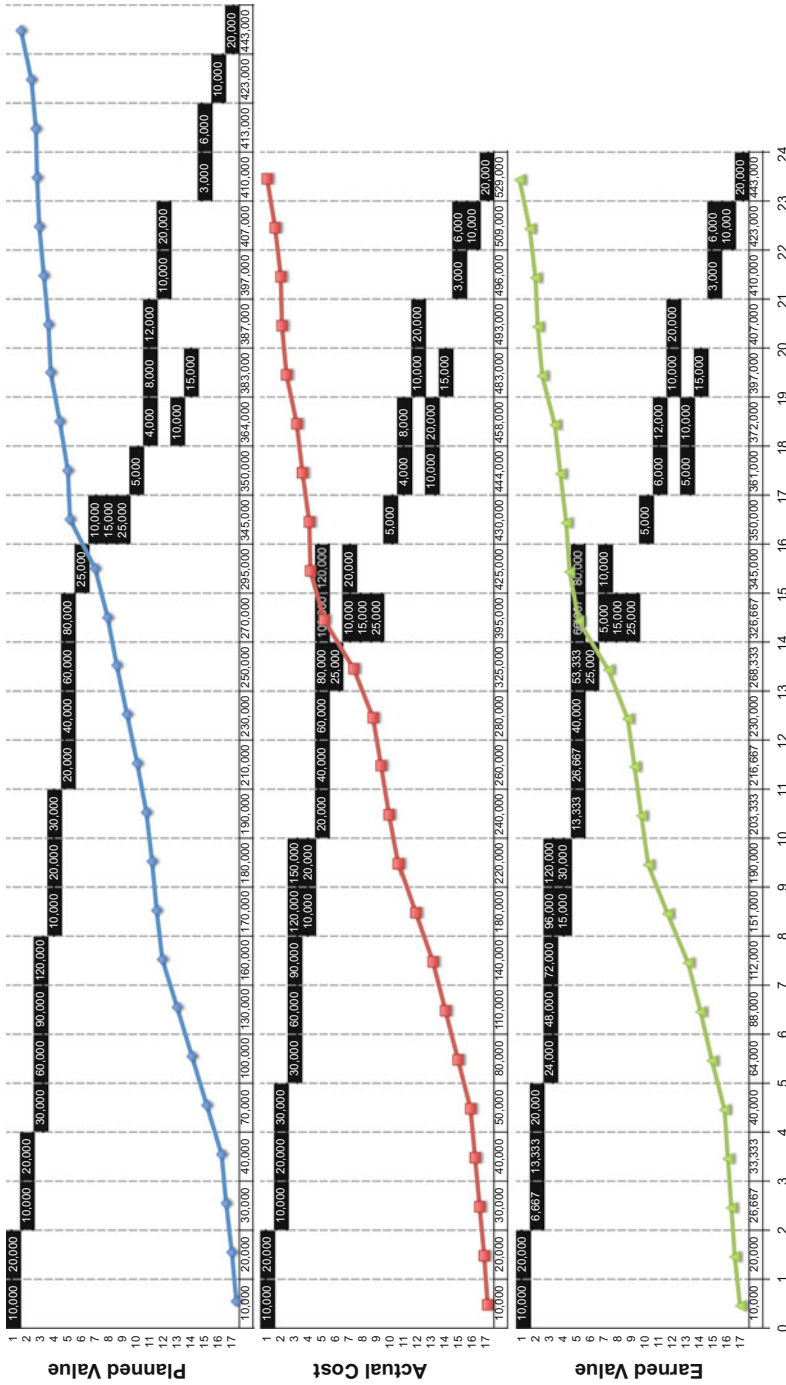
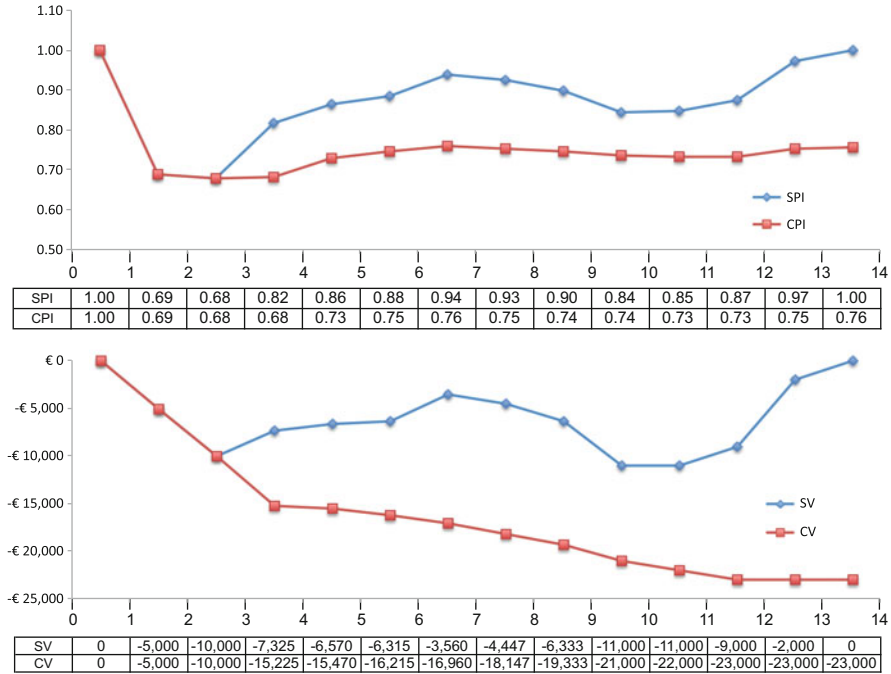


Fig. 2.5 Project 3. The three EVM key metrics (PV, AC and EV)



**Fig. 2.6** Project 1. The time and cost performance indicators SPI and CPI

Despite the simplicity of the time performance metrics SV and SPI, their interpretation often leads to confusion and misinterpretation. Two main reasons for these misinterpretations are discussed in the next paragraphs using the time performance data for Project 1, as shown in Fig. 2.6. The figure displays the weekly time performance using the SPI line at the top graph and using the SV line at the bottom graph, with the following characteristics:

**Time is money:** The deviations between the metrics EV and PV do not incorporate real costs but instead measure the time difference between the planned work and the work done. However, this difference is still expressed in a monetary unit (€) and might give the false impression that the time metrics can be used for cost performance reporting instead of time performance reporting. The PV reports the planned value at a certain moment in time, while the EV reports the value of the work done at a moment in time. Since this EV is expressed as a percentage of the planned value and does not necessarily have a relation with real costs, it just shows what the ideal planned timing would have been for the current value earned. Consequently, the difference between earned value and planned value at moment AT shows the time performance of the project. As an example, the PV of Project 1 at week 5 is equal to €48,000, while the EV is equal to €41,430 (see Fig. 2.3). Since  $EV < PV$ , this EV should normally have been earned earlier than week 5 and hence it is concluded that the project is late ( $SV < 0$  and  $SPI < 1$ ). The current

time performance is only 86 % (SPI) of the expected performance, or otherwise stated, the planned percentage completion is equal to  $48,000/71,500 = 67\%$  while the real percentage completion is only  $41,430/71,500 = 58\%$  and hence the project is currently running behind schedule.

**Time is quirky:** The time performance graphs in Fig. 2.6 show that the project is behind schedule ( $SV < 0$  and  $SPI < 1$ ) along most of the review periods of the life of the project. Only at the project finish, the  $SV = 0$  and  $SPI = 1$ , denoting a project finishing on time. However, it is known that Project 1 has a 3 week delay ending with a real duration  $RD = 14$  weeks, compared to its baseline planned duration  $PD = 11$  weeks. The reason for this quirky behaviour at the project finish lies in the  $SV$  and  $SPI$  formulas. The  $EV$  is measured as  $PC * BAC$  and expresses the fraction of the  $BAC$  that is earned for the work done. At the project finish,  $PC = 100\%$  and  $EV = BAC$ . Even a project that finishes later than its  $PD$  has a  $PV = BAC$  (in the time window between the  $PD$  and the  $RD$ ), and consequently,  $SV = EV - PV = 0$  and  $SPI = EV/PV = 1 = 100\%$ . It has been shown by Vanhoucke (2010a) that this unreliable trend towards 1 already starts at 50–60 % project completion. In order to overcome this unreliable behaviour, Lipke (2003) has proposed an alternative time performance system, known as Earned Schedule Management, which is the topic of Chap. 3. It should be noted that this unreliable trend towards 1 in the  $SPI$  will always occur for projects ending late. Since Project 2 finishes on time and Project 3 finishes early, this quirky behaviour cannot be observed.

### 2.2.2 Cost Performance

The cost performance of a project in progress is measured by two performance metrics, known as the cost variance ( $CV$ ) and the cost performance index  $CPI$ . Both metrics measure the deviation between the value of the work done ( $EV$ ) at the current moment  $AT$  and the actual cost ( $AC$ ) made at that moment. Similar to the time performance metrics, the cost variance is measured in absolute monetary units while the cost performance index expresses the cost performance as a percentage of the expected baseline cost performance, as summarised along the following lines:

$CV$	Cost Variance → calculated at each review period as $CV = EV - AC$ <ul style="list-style-type: none"> <li>&lt; 0: project cost over budget</li> <li>= 0: project cost on budget</li> <li>&gt; 0: project cost below budget</li> </ul>
$CPI$	Cost Performance Index → calculated at each review period as $CPI = EV/AC$ <ul style="list-style-type: none"> <li>&lt; 1: project cost over budget</li> <li>= 1: project cost on budget</li> <li>&gt; 1: project cost under budget</li> </ul>

Unlike the time performance metrics, the cost performance metrics now include real costs in comparison to the planned costs that should have been made for the work done (EV). These metrics are often much easier to understand since they measure the cost deviation in monetary units (€), which is the unit of costs. Moreover, no strange and unreliable behaviour can be observed for the cost performance measures CV and CPI since they report the real reliable cost performance from the start until the very end of the project.

## 2.3 Conclusions

In this chapter, an overview is given of the most commonly known metrics used in Earned Value Management (EVM) systems. An EVM system consists of a set of metrics to measure and assess the quality of the actual progress of projects relative to the expected progress as stipulated in the baseline schedule. These EVM metrics can be visually categorised into three classes as presented by the three levels in Fig. 1.8.

At the first level, three key metrics are used as inputs for all further calculations discussed in the next chapters. These key metrics are measured at the activity level, i.e. the lowest level of the work breakdown structure, and serve as inputs for the second (performance measurement) and third (forecasting) level of EVM systems, and are summarised along the following lines:

- **Planned Value:** The cumulative increase in the value of the project over time as a direct result of the baseline schedule cost values of project activities.
- **Actual Cost:** The cumulative increase in real activity costs summed up at the project level to know the actual spendings at a given moment in time.
- **Earned Value:** The cumulative increase of the value of the work done at a certain moment in time by connecting the physical percentage completion with the expected baseline project value.

Based on these three key input metrics used in EVM, a fourth metric can be calculated known as the Earned Schedule (ES) metric which is the topic of Chap. 3.

The second level makes use of the three key metrics to measure and assess the performance of projects in progress. Since a project in progress might have unexpected deviations from the baseline schedule estimates, measuring its performance is crucial for efficient project control. These EVM performance measures are used to give a general idea of the project's health, both for the time and cost objectives, at higher levels of the Work Breakdown Structure (WBS), and can be summarised as follows:

- **Time performance:** Time deviations between the current project progress and the baseline schedule estimates are measured by the Schedule Variance (SV), expressed in monetary units or by the Schedule Performance Index (SPI) expressed as a percentage of the expected performance.



- **Cost performance:** Cost deviations between the current project progress and the baseline schedule estimates are measured by the Cost Variance (CV), expressed in monetary units or by the Cost Performance Index (CPI) expressed as a percentage of the expected performance.

The performance of a project in progress at a certain moment in time plays a crucial role in the decision making process during project control to determine when actions are necessary to bring a project in trouble back on track or to exploit project opportunities in case the performance is better than expected. The use of these measures in a project control setting is the topic of Chap. 7. The current performance metrics will be used as inputs for the third level of an EVM system in order to predict the expected final project objectives. The forecasting methods predict the expected project time and cost and are based on the currently known and expected performances in the future. These EVM forecasting methods will be discussed in Chap. 5.

## Appendix: EVM Performance Reports

This appendix to Chap. 2 displays three summary reports for the three example projects. A summary report contains the relevant information for an EVM performance analysis and is built up as follows:

Each report shows the three key metrics along the project life in the top left pane, for which a short conclusion and discussion is reported next to these graphs. The bottom pane contains the time and cost performance measures with a short conclusion at the bottom right pane of each report.

Figure 2.7 displays a performance report for Project 1 at its project finish as a simple and easy to understand bird's eye view on the project performance in a one-page summary. Using such a report during project progress facilitates decision making since it gives a quick overview of the general health of the project as displayed by the periodic EVM key performance metrics. Obviously, these reports should be made at regular review periods during which the project is still in progress and the EV and AC lines are not yet known completely. However, it is assumed that the project is finished and the real time and cost status is known from start until end.

Figure 2.8 displays the EVM performance report for Project 2. Since the project finishes exactly on time, the  $SV = 0$  and  $SPI = 1$ . The costs are higher than planned and hence  $CV < 0$  and  $CPI < 1$  throughout the project.

Similarly, Fig. 2.9 displays the EVM performance report for Project 3, showing a project finishing ahead of schedule ( $SV > 0$  and  $SPI > 1$ ) with a cost overrun ( $CV < 0$  and  $CPI < 1$ ).

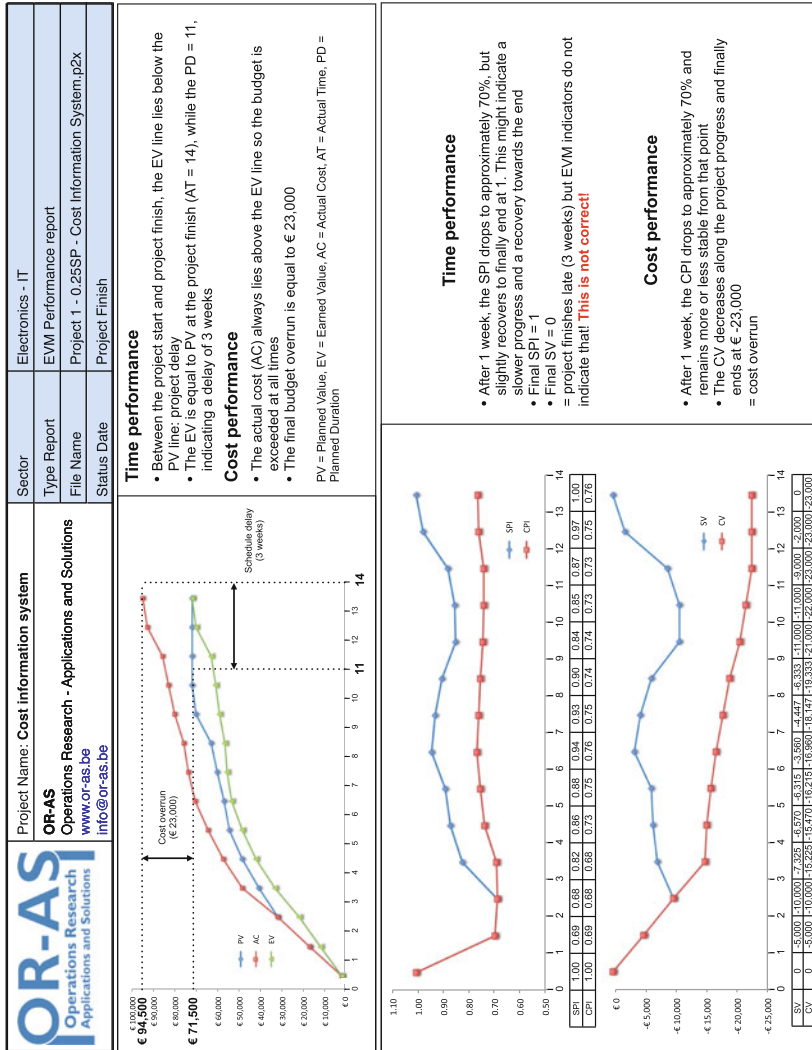


Fig. 2.7 Project 1. An EVM performance report

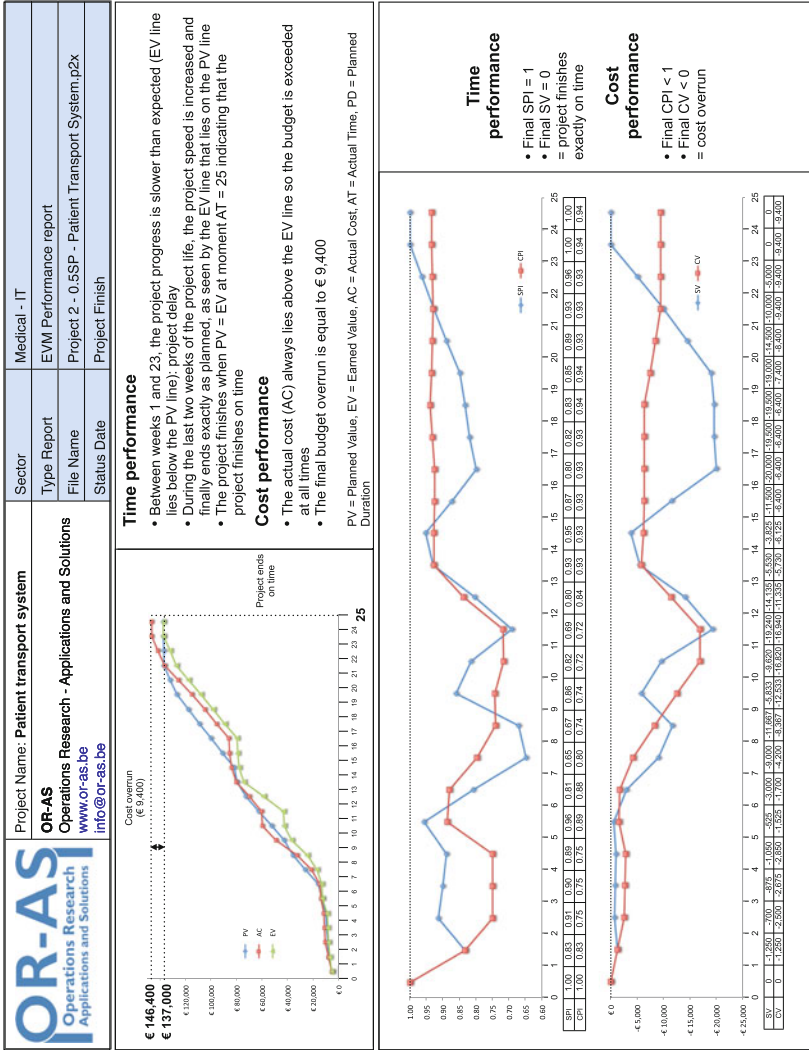


Fig. 2.8 Project 2. An EVM performance report

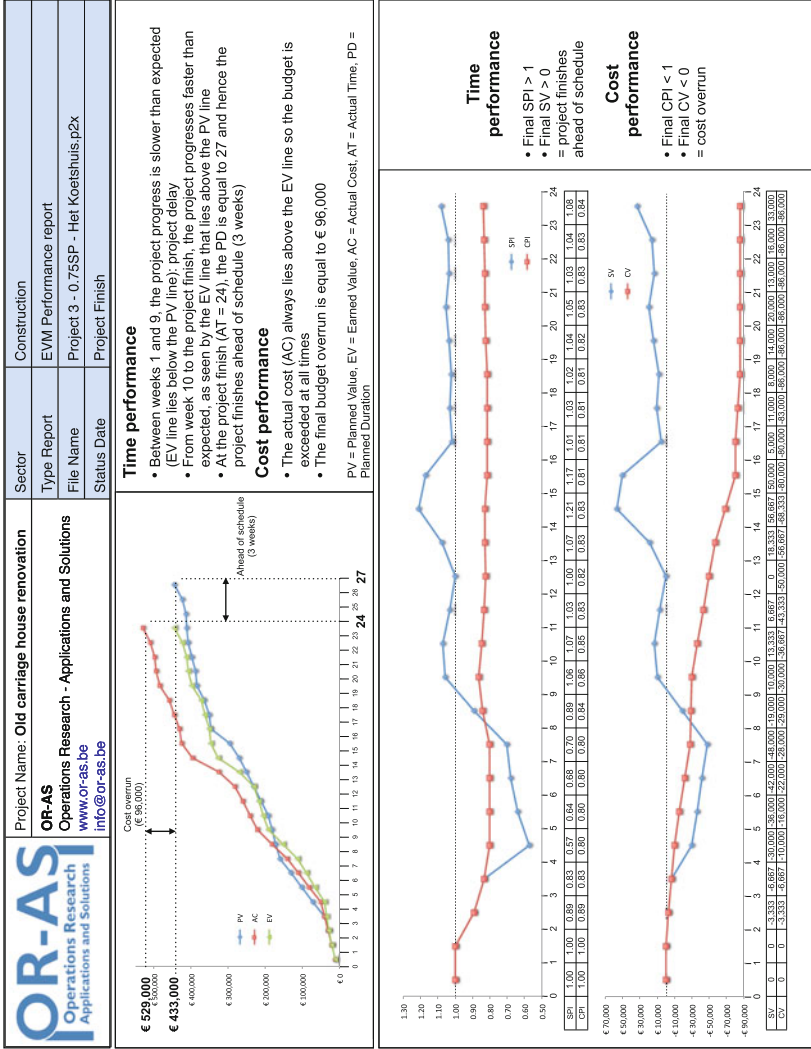


Fig. 2.9 Project 3. An EVM performance report

# Chapter 3

## Earned Schedule

The two time performance metrics presented in the previous chapter, known as the Schedule Variance (SV) and the Schedule Performance Index (SPI), have known shortcomings and exhibit unreliable trends from the middle to late project stages. These shortcomings have led to the development of a simple alternative, known as the Earned Schedule (ES) method. This method makes use of the traditional EVM key metrics, and can therefore be seen as an extension and/or improvement to the current traditional time performance metrics rather than a newly designed technique.

In this chapter, this ES method is discussed and its advantages are shown in comparison with the traditional time performance metrics of EVM. Figure 3.1 displays the relevant parts of Fig. 1.8 that will be discussed in this chapter and shows that the three key metrics discussed in the previous chapter serve as input metrics for the ES method. In Sect. 3.1, the shortcomings of the EVM-based time performance metrics are briefly reviewed since they were the main reasons for the development of the alternative Earned Schedule method. Section 3.2 gives a detailed introduction to the Earned Schedule (ES) metric as a time-based alternative for the Earned Value (EV) metric. In Sect. 3.3, the newly proposed time performance metrics SV(t) and SPI(t) are introduced as an improved alternative for the SV and SPI metrics. Finally, Sect. 3.4 gives a summary of the main topics of this chapter and draws general conclusions.

### 3.1 What Is Wrong with Earned Value?

In the previous chapter (Sect. 2.2.1), the two time performance metrics SV (schedule variance) and SPI (schedule performance index) have been introduced and their shortcomings have been discussed. In this section, these two shortcomings are briefly repeated since they are necessary to understand the use of the ES method that is a simple yet effective solution to these known problems. More precisely, in Sect. 3.1.1, the SV is criticised for its use of monetary units to express variances in time, while in Sect. 3.1.2, the SPI is shown to have an unreliable behaviour from

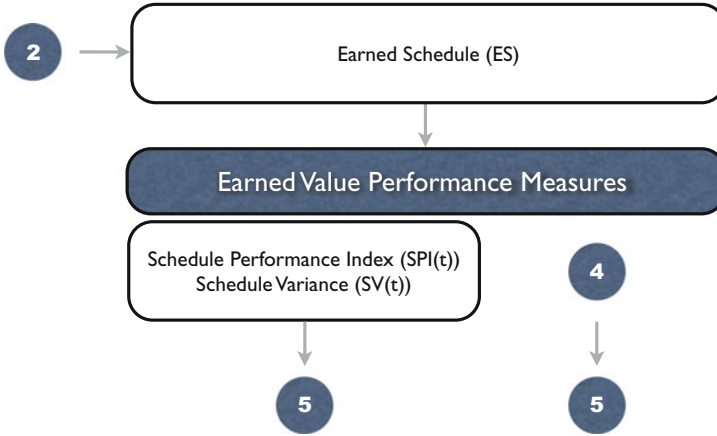


Fig. 3.1 ES piece of the jigsaw puzzle of Fig. 1.8

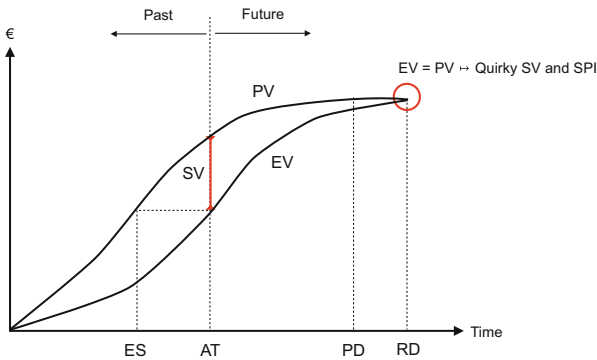


Fig. 3.2 Schedule variance to measure time expressed in euro

the middle to late stages of the project progress. Figure 3.2 displays the cumulative increase of the Planned Value (PV) and Earned Value (EV) of an example project and will be used to illustrate the two flaws of the earned value method in an easy way.

### 3.1.1 Time Is Money

One of the reasons why ES is preferred above the EV metric to monitor the time performance of a project is that time should be expressed in a time unit and not in a monetary unit. The SV measured at the current review period, i.e. at actual time (AT), is shown as the vertical difference between the PV and EV and is expressed in euro. However, this variance should better be expressed in a time unit, since it

is used to measure the difference between planned work and earned work in time. This is exactly what the Earned Schedule metric does: it reports the moment in time where the current EV should have been earned according to the baseline schedule. As an example, the current EV at time AT should have been earned earlier in time as can be seen in Fig. 3.2, and hence, the ES is the time where the EV should have been earned according to the baseline schedule. This ES point in time can simply be found by the projection of the EV onto the PV curve and its calculation will be outlined in Sect. 3.2.

While expressing time variances and performance metrics in a time unit instead of a monetary unit makes them easier to comprehend, this was certainly not the only reason why the ES metric has been added as an extension to the EVM methodology. Instead, their unreliability to measure the correct time performance along the life of the project was the main reason for this simple yet effective EVM add-on, as discussed in the next section.

### 3.1.2 Time Is Quirky

The main reason why the ES metric can be seen as a worthwhile alternative to the EVM based time performance measures lies in the quirky and unreliable behaviour of the SV and SPI metrics towards the later stages of the project progress. Figure 3.2 clearly shows that the vertical distance between PV and EV becomes more narrow towards the project end and ultimately will be equal to zero at the project finish. This observation can be explained by the formula of EV which is equal to  $PC * BAC$ . The percentage of project completion (PC) varies between 0% (at the project start) and 100% (at the finish) and hence EV varies between 0 and BAC, just like the PV curve.

This decreasing deviation between the PV and EV values leads to  $SV = 0$  and  $SPI = 1$  at the project finish, indicating a perfect on-time project schedule, even when the project is unacceptably late. This unreliable trend has been observed in early EVM studies but has been formally described by Lipke (2003) in his seminal paper “*Schedule is different*”. His reference to the so-called quirky behaviour of the SV and SPI measures has been the start of a new stream of papers in the academic and popular literature promoting, criticising and validating the newly proposed Earned Schedule method, which is the topic of this chapter. In the research study of Vanhoucke and Vandevoorde (2007b), it is shown that the quirky behaviour results in an unreliable trend towards 0 (SV) or 1 (SPI) that starts from 50 to 60% project completion onwards.

## 3.2 Earned Schedule

In this section, the calculations to obtain periodic values for the Earned Schedule (ES) metric are discussed, and the metric’s relevance to monitor the time performance of a project in progress is shown. The ES metric is similar to the EV metric

and connects the periodic EV measured during project progress to the PV curve in order to obtain a periodic ES value. Consequently, the periodic calculation of the ES metric does not require extra input other than the three key metrics PV, EV and AC presented in Chap. 2 and can therefore easily be used by any project manager familiar with traditional EVM.

The one-to-one link between EV and ES can easily be verified along the following lines. While the EV line displays the earned increase of the project value, expressed in monetary units (e.g. €) at a certain review point, the ES line shows a similar increase of the project value earned at a certain review point expressed in a time unit (such as days or weeks). Therefore, the ES is not more than an extension of the EV metric and can easily be calculated by linking the EV metric with the PV curve.

ES                      Earned Schedule  
                               → calculated at each review period based on current EV  
                               = time point where the EV is equal to the PV

The time point where the EV at the current review period (actual time) is equal to the PV value on the PV curve can easily be computed as follows:

$$\text{Find } t \text{ such that } EV \geq PV_t \text{ and } EV < PV_{t+\Delta t}$$

$$ES = t + \frac{EV - PV_t}{PV_{t+\Delta t} - PV_t} \Delta t$$

with

ES                      Earned Schedule  
 EV                      Earned Value at AT  
 PV<sub>t</sub>                      Planned Value at time instance t

The cumulative value for the ES is found by using the EV to identify in which time increment  $t$  of PV the cost value for EV occurs. ES is then equal to the cumulative time  $t$  to the beginning of that time increment, plus a fraction  $\frac{EV - PV_t}{PV_{t+\Delta t} - PV_t} \Delta t$  of it. The fraction equals the portion of EV extending into the incomplete time increment divided by the total PV planned for that same time period, which is simply calculated as a linear interpolation between the timespan of time increment  $t$  and  $t + \Delta t$ . Note that the formula description is not completely mathematically correct in case  $EV = PV_t = PV_{t+\Delta t}$ . In this case, the ES is equal to the earliest period  $t$  for which  $EV = PV_t$ . Figure 3.3 shows a graphical fictitious example of the linear interpolation of the planned values between the review periods at time  $t$  and time  $t + \Delta t$ . In the remainder of this book, it is assumed without loss of generality that  $\Delta t = 1$  week.

Figure 3.4 displays the cumulative ES value along the life of Project 1 with a planned duration  $PD = 11$  weeks and a real duration  $RD = 14$  weeks (3 weeks delay). The x-axis shows the real timing of the project progress from start till end (14 weeks). The y-axis shows a similar time axis to display the value of the ES,



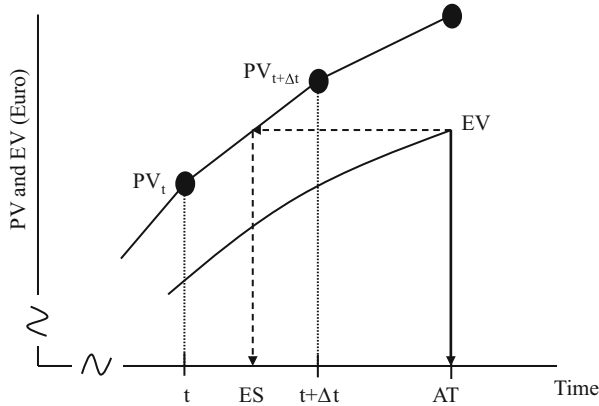


Fig. 3.3 Linear interpolation between  $PV_t$  and  $PV_{t+\Delta t}$

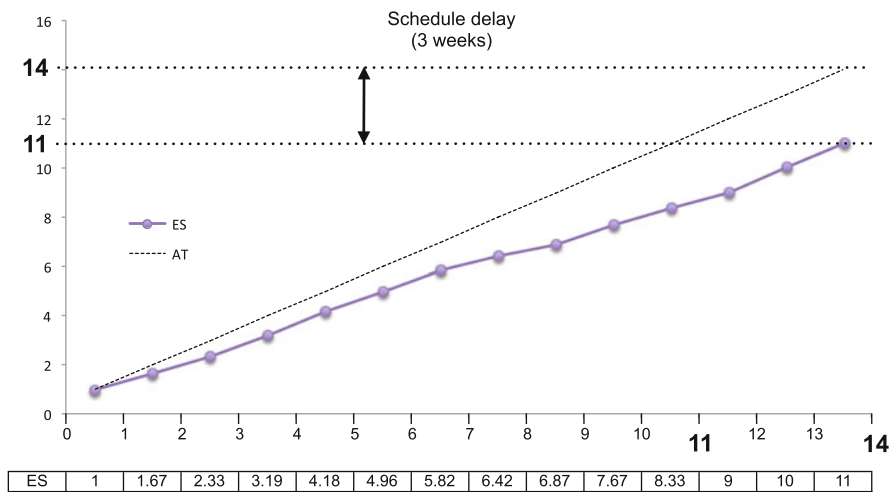


Fig. 3.4 Project 1. ES curve

which is expressed in time units. The dotted line is the actual time (AT) line which is a 45° slope line since both axes express one and the same time unit. This AT line is used as the reference point to display normal project progress. The ES line always lies below this normal progress line denoting a slower project progress. Similar to the EV metric that has a maximum value equal to BAC at the end of the project, the ES metric has a maximum value of PD at its finish. Indeed, at the project finish  $EV = BAC$  and since ES expresses the time point where the EV value should have been obtained, this moment lies at the expected planned duration given by the baseline schedule.

As an example, at week 5, the  $EV = \text{€}41,430$  as shown in Fig. 2.3. This value lies between the PV values of weeks 4 and 5, i.e.  $PV_4 = \text{€}40,000$  and  $PV_5 = \text{€}48,000$ .

Consequently,  $ES = 4 + \frac{41,430-40,000}{48,000-40,000} (5 - 4) = 4.18$ , as shown in Figs. 3.4 and 4.4. Note that similar graphs have been drawn for Projects 2 and 3, showing that early project progress results in an ES line that lies above the AT line at some moments in time. These graphs can be found in the ES summary reports of Figs. 4.5 and 4.6.

### 3.3 Performance Measurement

Since the ES method is an alternative method to measure the time performance of a project in progress, alternative performance metrics for the schedule variance (SV) and the schedule performance index (SPI) as presented in Sect. 2.2.1, can be calculated along the following lines:

SV(t)	Schedule Variance → calculated at each review period as $SV(t) = ES - AT$ < 0: project behind schedule = 0: project on schedule > 0: project ahead of schedule
SPI(t)	Schedule Performance Index → calculated at each review period as $SPI(t) = ES/AT$ < 1: project behind schedule = 1: project on schedule > 1: project ahead of schedule

It has been shown earlier that the ES metric has a one-to-one link with the EV metric, and it is therefore easy to see that the SPI(t) formula is also similar to the traditional SPI formula. Therefore, the SPI(t) calculation at a certain review period AT is completely analogous to its SPI calculation.

For a project in progress at the current moment AT, the following holds:

EV = value earned at AT for the work done expressed in a monetary unit

ES = time earned at AT for the work done expressed in a time unit

and

PV = work planned at AT expressed in a monetary unit

AT = the amount of time the project is in progress

Based on the similarities between EV and ES and PV and AT, respectively, the calculation of  $SPI = EV/PV$  can be replaced by its time based alternative SPI(t) calculated as  $ES/AT$ .

Figure 3.5 displays the SV(t) and SPI(t) curves along the life of Project 1. Both graphs show a project delay from the second week onwards and show that the time performance metrics SV(t) and SPI(t) display a reliable behaviour from the very start until the project end. At the project finish,  $SV(t) = -3$  denoting a 3 weeks delay, or alternatively, the  $SPI(t) = 79\%$  denoting a time performance lower than initially planned. Unlike the SV and SPI metrics, there is no unreliable zone (from

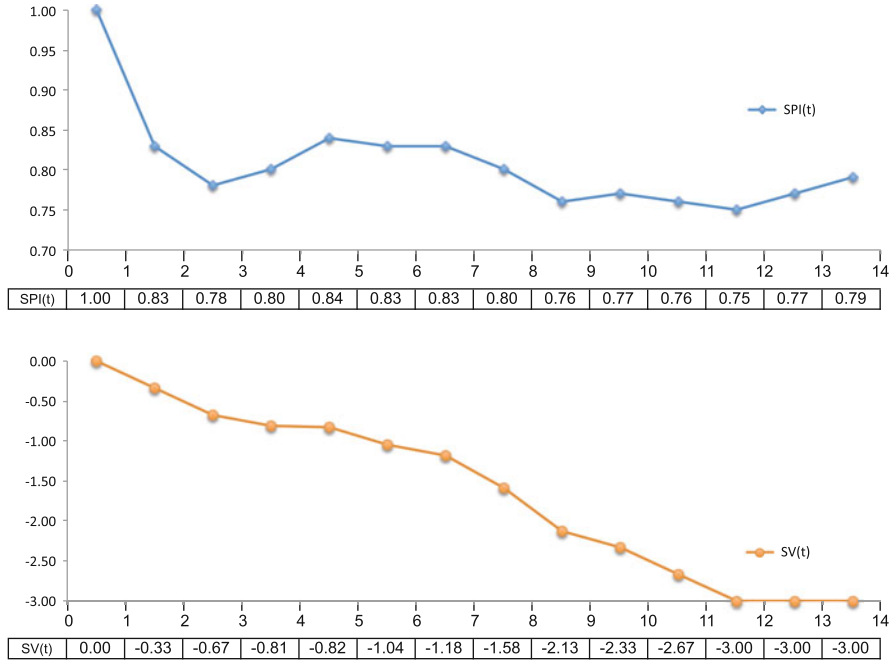


Fig. 3.5 Project 1. Time performance indicators SV(t) and SPI(t) curve

the 50 to 60 % project completion onwards) where the metrics tend to go to zero (SV) or one (SPI) and hence there is no quirky behaviour.

### 3.4 Conclusions

In this chapter, the Earned Schedule (ES) technique as originally developed in the seminal article “*Schedule is different*” by Lipke (2003) has been presented as a simple and straightforward alternative to the EVM time metrics. The metric is an extension of the EV metric and can be calculated based on the current value for the EV metric and the known PV line. Consequently, no extra input parameters are needed per review period to calculate the value of the project ES. The Earned Schedule metric can be defined as the translation of EV expressed in monetary units into a metric expressed in a time unit. Its value is obtained by projecting the EV at the actual time AT onto the PV curve. Values of ES lower than AT denote a project delay while ES values bigger than AT refer to a project running ahead of schedule.

Similar to the time performance measures of an EVM system, the ES metric is used as input for measuring and assessing the time performance of a project in progress, and results in two performance measures summarised along the following lines:

- The Schedule Variance ( $SV(t)$ ) is a time-based alternative for the  $SV$  measure and expresses the time variance in a time unit rather than in a monetary unit.
- The schedule performance index ( $SPI(t)$ ) is, similar to the  $SPI$ , a time performance index expressed as a percentage of the expected baseline performance.

The advantage of the  $SV(t)$  and  $SPI(t)$  measures lies in their reliability to measure the correct time performance from project start until finish. While the  $SPI$  is known to exhibit a quirky and unreliable trend towards seemingly perfect performance, even for projects ending dramatically late, the  $SPI(t)$  does not show this misleading trend and is therefore much more reliable.

Similar to the EVM metrics presented in the previous chapter, the ES performance indicator  $SPI(t)$  will be used to predict the total duration of the project in progress. Ever since the introduction of Earned Schedule in 2003, a lot of research and practical experience has been accumulated. In the research study of [Vanhoucke \(2010a\)](#), it was shown that the ES metrics outperform the EVM metrics when used for duration forecasting. The EVM and ES forecasting methods and all corresponding calculations will be discussed in Chap. 5.

# Chapter 4

## Schedule Adherence

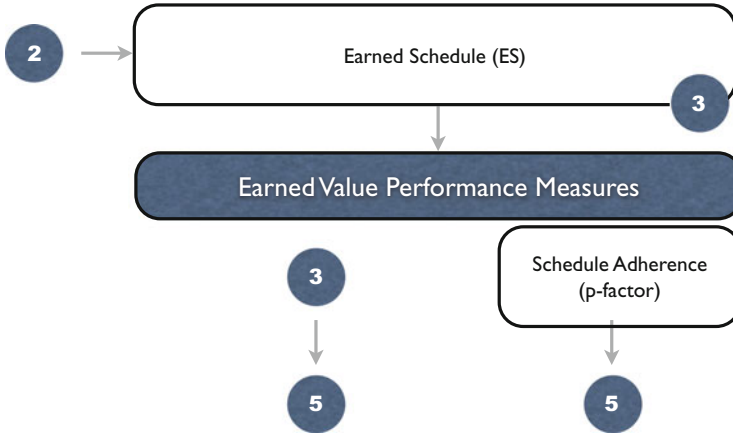
Measuring schedule adherence is a relatively unexplored concept in Earned Value Management and has been proposed as an extension to the Earned Schedule methodology described in the previous chapter. The purpose of schedule adherence is to connect the logic of the baseline schedule with the dynamics of project progress. It is formalised in the so-called p-factor approach and measures the portion of the Earned Value (EV) accrued in congruence with the baseline schedule. The approach is used to adapt the current EV to an effective earned value (EV(e)) taking the risk of rework into account.

In this chapter, this p-factor approach is discussed in detail and illustrated on the artificial project data. Figure 4.1 displays the relevant parts of Fig. 1.8 that will be discussed in this chapter.

Section 4.1 describes the p-factor approach to measure the adherence to the baseline schedule and illustrates its calculations on one of the fictitious projects. In Sect. 4.2, the p-factor is used to calculate the effective earned value (EV(e)) that takes the risk of rework due to a certain lack of schedule adherence into account. The appendix of this chapter displays three summary reports for the three projects showing the ES performance and schedule adherence analyses. Section 4.3 gives a summary of the main topics of this chapter and draws general conclusions.

### 4.1 Schedule Adherence

Throughout the book, the integrated project management and control approach is based on the principles of *dynamic scheduling*, which implicitly assume that the relevance and use of a project baseline schedule is rather limited. Rather than constructing a baseline schedule that should act as a roadmap and realistic scenario for the expected project progress, it only acts as a point of reference in the project life cycle. In this respect, a project baseline schedule should especially be considered as nothing more than a predictive model that can be used for resource



**Fig. 4.1** Schedule adherence piece of the jigsaw puzzle of Fig. 1.8

efficiency calculations, time and cost risk analyses, project control and performance measurement. It is a necessary step in the life of a project to allow efficient project control using EVM (Chap. 2), ES (Chap. 3) or SRA (Chap. 6). Controlling projects from this point of view also implies that following the logic and timing of the baseline schedule is not a goal in itself, but rather a benchmarked reference point that acts as a trigger for corrective actions. The word “dynamic” is therefore added to refer to the difference between *static* scheduling where the baseline schedule is considered to hold the holy truth and deviations are seen as unexpected changes, and *dynamic* scheduling where these deviations are seen as inevitable and where the baseline schedule is used to detect them as soon as possible in order to take corrective actions.

The concept of *schedule adherence* discussed in this chapter goes one step further in order to integrate the philosophies of static and dynamic scheduling. It tries to strengthen the connection between the static project baseline schedule and the dynamic project progress. It assumes that not only the timing of the baseline schedule, but also the logic of the project network should be respected, and deviations from that network logic can lead to unexpected negative outcomes, regardless of the performance of the project in progress. Consequently, the subtle added value of the schedule adherence concept lies in the importance it puts on the logic of the project network and the baseline schedule. This is in contrast with traditional EVM/ES performance measurement metrics which consider this baseline schedule merely as a point of reference to calibrate all measurements.

The schedule adherence approach is formalised in the so-called p-factor concept which is an extension to the EVM and ES methods. This new concept has only been introduced as an idea in 2004 and has not yet passed the test of logic, however it certainly deserves more research attention to test its merits in a dynamic project control environment. Therefore, in the remaining sections of this chapter, the

p-factor approach is discussed and its relevance is sketched without elaborating on further possible improvements and/or shortcomings that could lead to an even more integrated project control approach.

### 4.1.1 *p*-Factor

The rationale behind the p-factor approach lies in the observation that performing work not according to the baseline schedule logic often indicates activity impediments or is likely a cause of rework. Consequently, the premise is that, whenever impediments occur (activities that are performed relatively less efficiently compared to the normal project progress defined in the baseline logic), resources are shifted from these constrained activities to other activities where they could gain earned value. However, this results in a project execution that deviates from the original baseline schedule logic which might involve a certain degree of risk. Indeed, the latter activities are performed without the necessary inputs, and might result in a certain portion of rework. Based on these observations, the p-factor has been introduced by Lipke (2004) as a measure to provide the connection between project output and EVM/ES.

Although the PV and EV metrics have been defined in the previous chapters, they are redefined below on a more detailed level. The main difference with the EV and PV abbreviations of previous chapters lies in the use of the subscript  $i$  to refer to the activity level instead of the project level and subscript  $t$  to refer to the point in time. Consequently,  $PV_{i,t}$  and  $EV_{i,t}$  are used to refer to the planned and earned value of activity  $i$  at time  $t$ , respectively. The metrics, necessary to display the mathematical formula of the p-factor, are explained along the following lines:

$N$	Set of activities in the project → known from the project network
$EV_{i,AT}$	Earned value of activity $i$ at the actual time AT → calculated at each review period = fraction of the activity PV earned at AT for the work done
$PV_{i,ES}$	Planned value of activity $i$ at time instance ES → PV is known from the baseline schedule → ES calculated at each review period
$p$	p-factor → calculated at each review period based on the current PV and EV = fraction of the total EV accrued before ES = 1: perfect schedule adherence < 1: lack of schedule adherence

The p-factor measures the portion of earned value accrued in congruence with the baseline schedule, i.e. the activities that ought to be either completed or in progress, as follows:

$$p = \frac{\sum_{i \in N} \min(PV_{i,ES}, EV_{i,AT})}{\sum_{i \in N} PV_{i,ES}}$$

The p-factor is the ratio of the earned value corresponding to the baseline schedule divided by the total planned value at time instance ES. Since the numerator takes the minimum of the planned value at time unit ES and the earned value accrued at the actual time, the p-factor obviously always lies between zero and one, inclusive. Hence, the p-factor measures to what degree the earned value is accrued according to the baseline schedule (100 % means a perfect schedule adherence).

Figure 4.2 displays a comparison between the p-factors for the three projects along their life. Project 1 shows a lack of perfect schedule adherence in the middle stages of the project. The p-factor of Project 2 is characterised by relatively deep drops below 1 denoting a significant lack of perfect schedule adherence. Project 3 has a relatively stable and almost perfect p-factor except for a few periods of the project progress. A summary of ES project performances, including p-factor evolutions, is given for each of the three example projects in the appendix of this chapter.

Table 4.1 shows the illustrative p-factor calculations for Project 1 at period AT = 4. The earned schedule at AT = 4 is ES = 3.186<sup>1</sup> and the p-factor is equal to 31,353.61/32,675 = 0.96 and is calculated as the sum of the last column divided by the sum of the second column. The values of the table can easily be found or calculated from the PV and EV graphs in Fig. 2.3. Some of the calculations for the second column (PV<sub>i,ES</sub>) and third column (EV<sub>i,AT</sub>) are illustrated along the following lines:

- PV<sub>i,ES</sub>: The values for the PV for each activity at time moment ES = 3.19 can be calculated based on the planned values given at the top picture of Fig. 2.3. The PV for activities 1, 2 and 3 are equal to the total planned activity costs 1,000, 15,000 and 15,000 since the activities are planned to be finished at time 3.19. The PV for activities 11–17 is equal to 0 since they all have a planned start after 3.19. For all other activities, the PV must be calculated as the planned value in the baseline schedule at time ES = 3.19. As an example, activity 4 has a planned start time at week 3 with an expected planned duration of 4 weeks and an expected total cost of €6,000 (or alternatively, €1,500 per week). Consequently, at time moment 3.186, the planned percentage completion is equal to  $\frac{3.186-3}{4} = 4.65\%$ . Consequently, the planned value is equal to PV<sub>4,ES</sub> = 0.0465 \* 6,000 = €279.17 (small rounding errors might occur again).
- EV<sub>i,AT</sub>: The values for the EV for each activity at time moment AT = 4 can simply be copied from the bottom Gantt chart of Fig. 2.3.

<sup>1</sup>The reader should be careful with manual calculations since rounding errors might occur. The ES at AT = 4 is equal to  $3 + \frac{32,675-31,000}{40,000-31,000} = 3.1861111$  and small rounding errors might lead to differences in the calculated results.



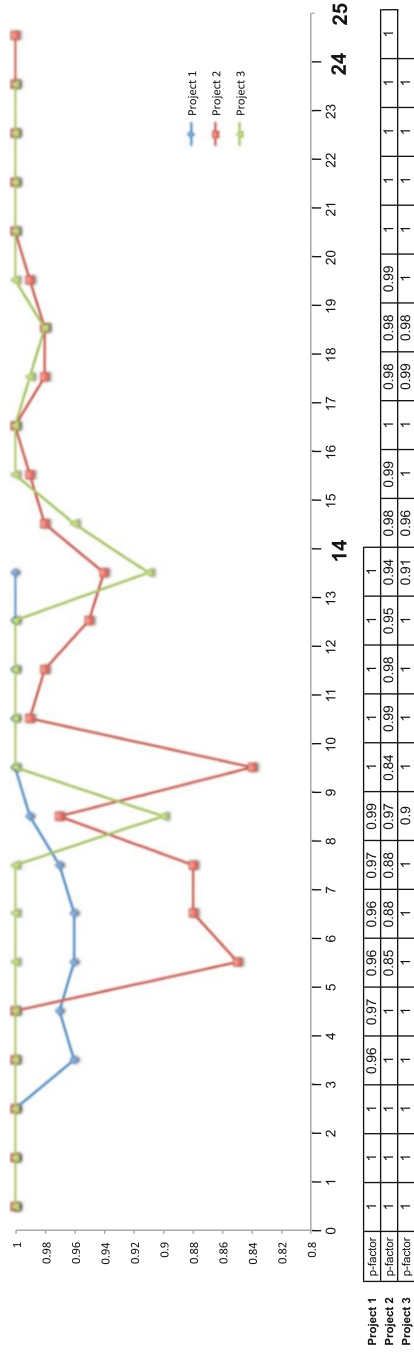
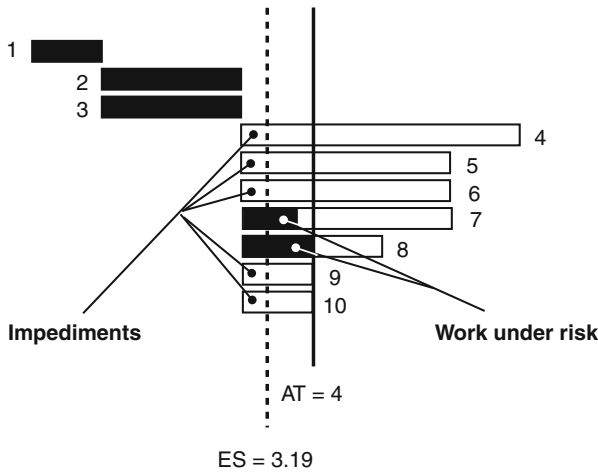


Fig. 4.2 p-factor for the three example projects

**Table 4.1** p-factor calculations at review period  $AT = 4$  for Project 1

Activity	$PV_{i,ES}$	$EV_{i,AT}$	$\min(PV_{i,ES}, EV_{i,AT})$
1	1,000	1,000	1,000
2	15,000	15,000	15,000
3	15,000	15,000	15,000
4	279.17	0	0
5	241.94	0	0
6	241.94	0	0
7	167.50	675	167.50
8	186.11	1,000	186.11
9	279.17	0	0
10	279.17	0	0
11–17	0	0	0
	32,675	32,675	31,353.61



**Fig. 4.3** The p-factor: connecting project progress to the baseline schedule

### 4.1.2 Graphical Illustration

The graphical meaning of the p-factor value can be illustrated on a Gantt chart that connects the planned activity durations with the activity progress. Figure 4.3 displays the Gantt chart of the baseline schedule of Project 1 at time moment  $AT = 4$ . The bars show the start and finish times of each activity as given by the baseline schedule displayed in the Gantt chart of Fig. 1.2. The black part of each bar is used to show the activity progress as its percentage completion, and displays the EV at the current time  $AT = 4$ . This value is known from the progress Gantt chart of Fig. 4.3. Consequently, Fig. 4.3 shows the current state of Project 1 at  $AT = 4$  relative to its baseline schedule.

Connecting the baseline schedule to the activity and project progress is exactly what the p-factor aims to accomplish. The ES of the project at  $AT = 4$  is equal to

3.19 weeks, and some black bars end before this dotted ES line (activities 1, 2 and 3) while others end later than this ES line (activities 7 and 8). The figure allows the detection of the portion of work performed in congruence with the baseline schedule. More precisely, the p-factor concept assumes that the portion of work performed before the ES line is in line with the baseline schedule while the portion of work after the ES line is performed under a certain degree of risk (see Sect. 4.2). In the example, a portion of the EV of activities 7 and 8 is considered as work under risk. In the case of perfect schedule adherence, this EV portion should have been accrued for activities 4, 5, 6, 9 and 10. Graphically, the p-factor is equal to the sum of the fractions of the earned values lying to the left of the earned schedule line divided by the total EV (i.e. sum of all black bars). The ES line is defined as the point on the baseline schedule where PV is equal to EV and consequently the ES value is the time at which the EV should have been accrued if the baseline schedule had been followed perfectly. The value of PV at time instance ES is the sum of planned values from a set of activities which are either planned to be completed or in progress at the time associated with ES and constitutes the denominator of the p-factor formula (this value is equal to the EV). The EV corresponding to these activities, not to exceed the planned value at ES, is the numerator of the p-factor formula.

## 4.2 Rework

The main contribution of the p-factor is to serve as a management tool to reveal impediments or constraints during the execution of the project and to detect the portion of work performed under risk. The fundamental idea behind the p-factor concept lies in the detection of impediments or constraints whenever resources are used in an inefficient way resulting in activity execution that lags the ES performance. This inefficiency, expressed by the p-factor, allows to correct the accrued earned value to the risk of rework, and calculates the effective earned value (EV(e)) as the risk-adapted portion of earned value that is performed within the expected baseline schedule performance, defined by the earned schedule metric. The transformation from the EV to EV(e) is discussed in Sect. 4.2.1. In Sect. 4.2.2, possible factors that might lead to p-factor values lower than 1 are briefly discussed.

### 4.2.1 *Effective Earned Value*

The p-factor assumes that lack of schedule adherence is caused by a combination of the presence of impediments or constraints and work performed under risk. The detection of both the activity impediments and the portion of work done under risk can be easily obtained using the earned schedule metric ES that splits the accrued earned value at time AT into two disjoint subsets, as follows:

- The portion of the work to the left of the ES line is assumed to be performed without risk and might reveal the presence of an impediment or project constraint.
- The portion of work to the right of the ES line indicates work that is ahead of the normal project performance and is assumed to have a certain degree of risk.

In the example at  $AT = 4$ , the EV for activities 1, 2 and 3 but also portions of the accrued EV for activities 7 and 8 lying before the ES line are done without risk. The portion of work to the right of the ES line for activities 7 and 8 indicates work that is ahead of the normal project performance and that is assumed to have a certain degree of risk.

The p-factor assumes that the degree of risk is often the result of inefficient use of resources compared to the baseline schedule logic. Normally, some work should have been done for activities 4, 5, 6, 9 and 10 at time ES, but no EV has been accrued for these activities. They are performed less efficiently relative to the normal project progress as measured by the ES line, which might indicate the presence of an impediment or project constraint. This might be an indication that resources were shifted from the constrained activities (4, 5, 6, 9 and 10) to less constrained activities (7 and 8) where the resources could gain earned value. However, since these resources often work without the necessary inputs, their work might result in a certain portion of rework (i.e. risk). In order to take this risk of rework into account, a correction to the earned value is needed to distinguish between the portion without risk and the portion of work which is likely to be performed under risk.

Consequently, the p-factor approach assumes that risk for rework is the result of inefficient use of resources that were shifted from the constrained activities to less constrained activities where the resources could gain earned value. Since these shifted resources work without the necessary inputs, the p-factor is a measure to express the portion of the EV without risk (referred to as  $EV(p)$ ), while the remaining portion is denoted as  $EV(r)$ . A project manager should realise that the remaining  $EV(r)$  portion might be subject to risk and possibly results in rework. The *effective earned value* ( $EV(e)$ ) is defined as the risk-adapted portion of earned value that is performed within the expected baseline schedule performance, taking into account that only  $R\% \in [0\%, 100\%]$  of the  $EV(r)$  will be accounted as risk-free. The metrics necessary to display the mathematical formula of the effective earned value ( $EV(e)$ ) are explained along the following lines:

EV	Total earned value
EV(p)	Risk-free earned value = $p * EV$
EV(r)	Remaining earned value portion performed under risk = $(1 - p) * EV$
EV(e)	Effective earned value = $EV(p) + R\% * EV(r)$
R%	Estimated portion of $EV(r)$ that is usable and requires no rework → User defined estimate

Since the portion of earned value under risk is likely to cause rework, the effective earned value (EV(e)) decreases the total earned value by taking the possibility of rework into account, as follows:

$$EV = p * EV + (1 - p) * EV = EV(p) + EV(r) \rightarrow EV(e) = EV(p) + R\% * EV(r)$$

In the example Project 1 at AT=4 as shown in Fig. 4.3, the value for the inefficiencies and risk of rework can be calculated as follows:

- Impediments: Activities 4, 5, 6, 9 and 10 all have a white zone before the ES line indicating an inefficiency and activity impediment. The planned value of these activities at time ES is equal to  $279.17 + 241.94 + 241.94 + 279.17 + 279.17 = \text{€}1,321.39$ .
- Risk for rework: A fraction of the work for activities 7 and 8 is done under risk, shown by the value of the black bars after ES. The EV accrued under risk is equal to  $\sum_{i=1}^N (EV_{i,AT} - PV_{i,ES}) = (675 - 167.50) + (1,000 - 186.11) = \text{€}1,321.39$ .

The value for the work done under risk is always the same as the value for the work not performed in congruence with the planned value at time ES shown by the inefficient activities. The total EV is equal to  $\text{€}32,675$  and hence,  $EV(p) = 32,675 - 1,321.39 = \text{€}31,353.61$  which is the value of the work done without risk. When it is assumed that the fraction of the work done under risk of 20% will be subject to rework, then  $R\% = 80\%$  and  $EV(e) = 31,353.61 + 0.8 * 1,321.39 = \text{€}32,410.72$  and  $\text{€}264.28$  is considered as lost as a result of the expected rework due to the lack of schedule adherence.

## 4.2.2 Schedule Deviations

Since the p-factor aims at connecting the baseline schedule with the project progress, it should be able to detect baseline schedule deviations during project progress. These schedule deviations lead to a certain lack of schedule adherence, and hence, the original logic captured in the project network (activity estimates and precedence relations) as well as the timetable of these activities should be violated. Possible reasons that might lead to a lack of schedule adherence, resulting in p-factor values lower than 1, are summarised along the following lines:

- Deviations in baseline estimates: Deviations from the original baseline activity time and cost estimates during project progress obviously lead to deviations in the expected time and cost performances, but might also lead to shifts in resources and p-factor values lower than 1.
- Early or late start times: Similar to baseline estimate deviations, early or late activity starts might lead to unexpected project performances, shifts in resources and p-factor values lower than 1.

- Overlaps between activities: Violating the original logic of the precedence relations and allowing overlaps between activities involves the execution of activities under incomplete information (since their predecessors have not been finished yet) and might involve a certain degree of risk and a probability for rework.
- Deviations in EV accrues: The accrue in EV can be different from the PV accrue assumption, leading to p-factor values lower than one. The choice of a PV accrue needs to be made before the start of the project, and can be chosen from various earned value measurement methods (e.g. the percent complete estimates, the milestones with weighted values, fixed formulas such as 25/75, 50/50 or others). When the reported EV accrue differs from the selected PV accrue system, inefficiencies will be reported by the p-factor.

All of the deviations mentioned above can lead to p-factor values lower than 1 during the project progress, but might also lead to SPI(t) values that differ from 1 denoting a deviation in the expected project performance. Consequently, the question arises whether the p-factor is fundamentally different from schedule performance metrics such as the SPI(t). At least one reason why the schedule adherence concept is not merely a replacement for the SPI(t) is that its relevance and novelty lies in the detection of impediments during project progress as well as the detection of the portion of work performed under risk. However, it is conjectured that the p-factor has much more value since it highlights certain dimensions of project performance that cannot be measured by the SPI(t). It is easy to construct an example where SPI(t) values lie close to 1 (denoting perfect performance) although p-factor values are significantly lower than 1. In the example calculations above, the project is subject to a small delay ( $ES < AT$ ) but the p-factor could have been equal to one if the inefficiencies in the activities would not have occurred. A fact is that the p-factor takes the current (good or bad) performance as a given and aims at detecting inefficiencies in normal project progress relative to the baseline logic. However, apart from occasional case studies, no clear research results have been reported in the literature, and hence, more research is certainly necessary (and on its way) to clarify the difference, if any, between project performance and schedule adherence. It should be noted that the p-factor references are restricted to the original work of [Lipke \(2004\)](#) who has proposed the p-factor concept and a limited simulation study reported in [Vanhoucke \(2013c\)](#). Book chapters with references to the p-factor can be found in [Lipke \(2012\)](#) and [Vanhoucke \(2009, 2010a, 2013b\)](#).

### 4.3 Conclusions

In this chapter, a novel concept to measure the adherence of an ongoing project to the baseline schedule is presented using the so-called p-factor. This concept aims at connecting the baseline schedule with project progress and is a simple and easy extension to the earned schedule approach. The p-factor approach can be used as a

management tool to reveal impediments or constraints during the execution of the project and to detect the portion of work performed under risk. The portion of work under risk might involve a certain degree of rework, and hence, the earned value measured at a certain point in time must be corrected for this estimated portion of rework. This changes the original EV calculations into an effective earned value calculation that splits the EV in a portion of value performed without risk of rework and a remaining portion of value that might be the subject to rework.

The meaning of two new metrics, abbreviated as the p-factor and EV(e), is briefly summarised along the following lines:

- p-factor: The p-factor is the ratio of the earned value corresponding to the baseline schedule divided by the total planned value at time instance ES and measures to what degree the earned value is accrued according to the logic captured by the baseline schedule.
- Effective earned value (EV(e)): The risk-adapted portion of earned value that is performed within the expected baseline schedule performance and without risk for rework.

The fundamental idea behind the p-factor concept lies in the detection of impediments or constraints whenever resources are used in an inefficient way resulting in an activity execution that lags the ES performance. This inefficiency, expressed by the p-factor, results in a correction of the accrued earned value by including the risk of rework, and calculates the effective earned value (EV(e)).

## Appendix: ES Performance Reports

The appendix to this chapter displays three summary reports for the three example projects. Each summary report contains the relevant information for an analysis of the project performance using the Earned Schedule metrics, including the p-factor.

Each report is constructed as explained along the following lines. The Earned Schedule metric is shown in the top left pane of each report and compared with a straight line showing the normal increase in ES, i.e. an increase of 1 week in ES when the real time increases with 1 week. When the ES line lies above this straight line, project progress is done faster than expected, while ES values below this line indicate project delays. For each report, this ES graph is analysed and discussed in the top right pane.

The graphs at the bottom of each report display the performance metrics of the projects in progress given by the SPI(t) (top graph) and SV(t) (bottom graph), as well as the schedule adherence over time, given by the p-factor value. A short discussion and/or conclusion is given next to each graph at the bottom right pane of each report.

Figure 4.4 displays the ES performance report for Project 1. Unlike the SV and SPI metrics shown in the EVM performance report of Fig. 2.7, the SV(t) and SPI(t) metrics show the real time performance from the start until the project finish at

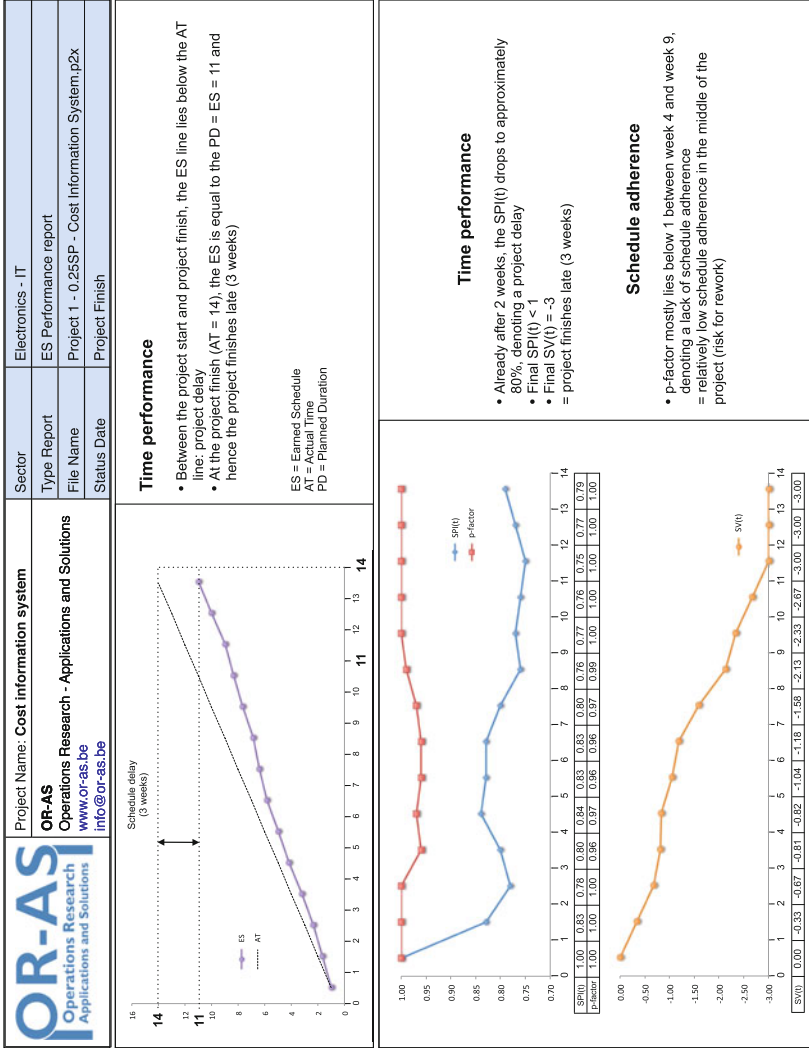


Fig. 4.4 Project 1. An ES performance report



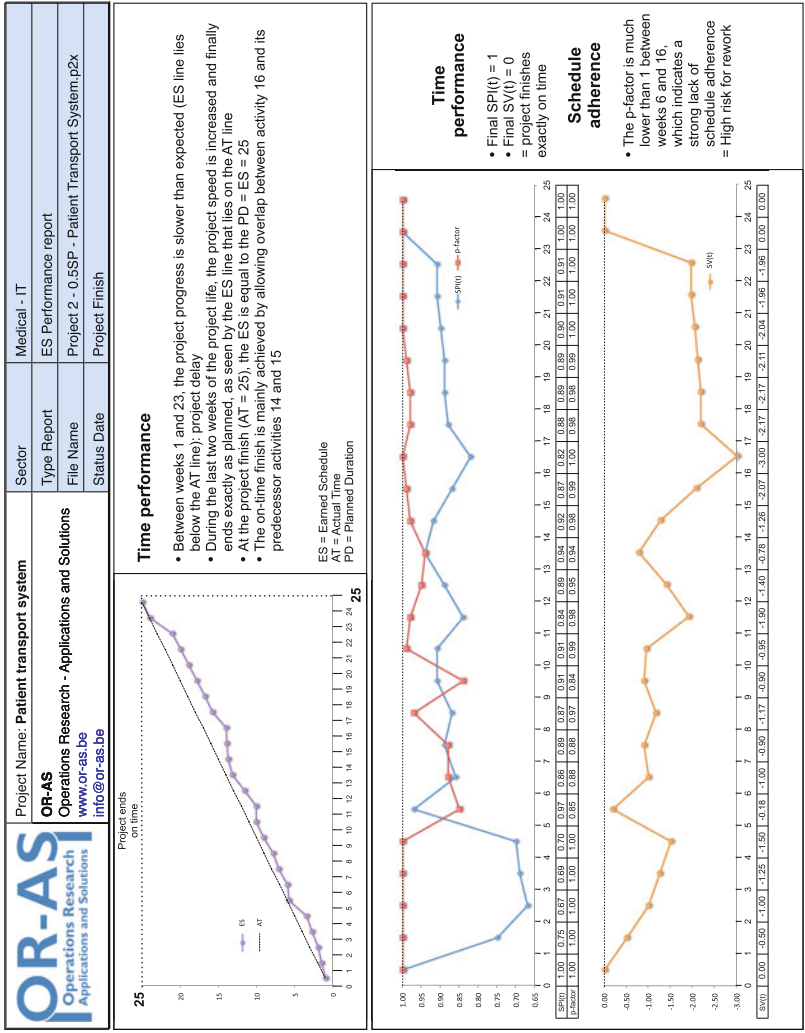


Fig. 4.5 Project 2. An ES performance report

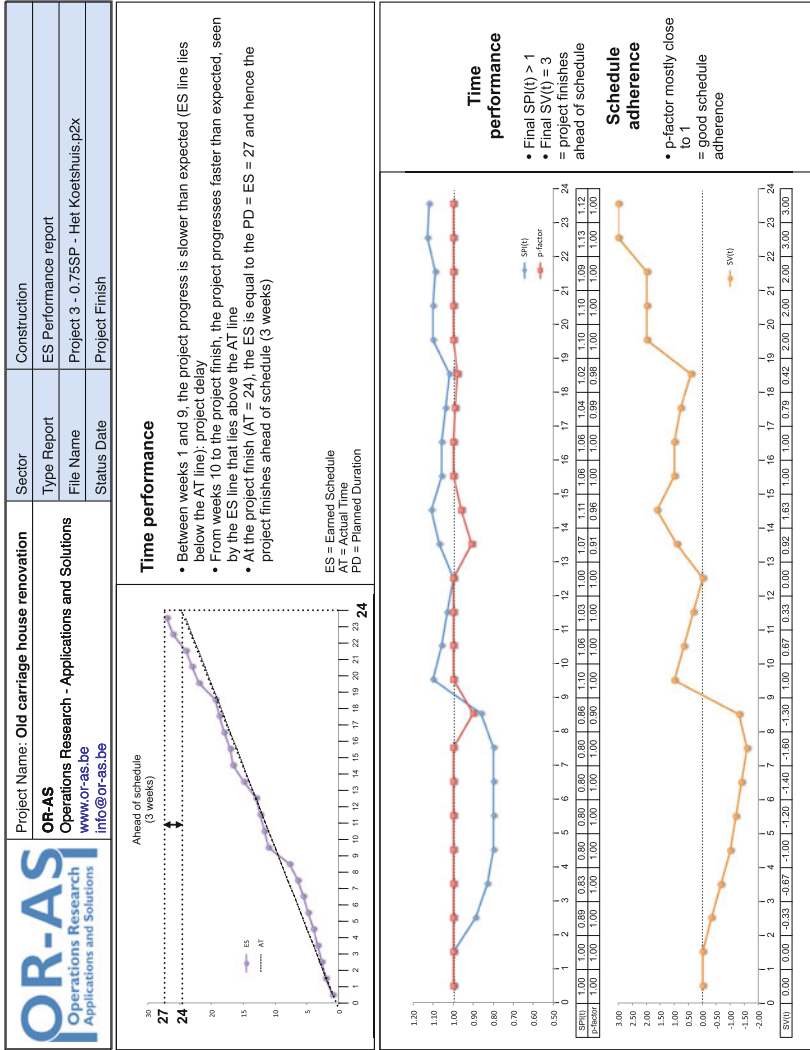


Fig. 4.6 Project 3. An ES performance report

week 14. The p-factor drops below 1 between weeks 4 and 9 due to activity overlaps and activity durations/costs not in congruence with the baseline estimates.

Figure 4.5 displays the ES performance report for Project 2. Since this project finishes exactly on time, the  $SV(t)$  and  $SPI(t)$  metrics end at 0 and 1, respectively. In between, the delays are reported by  $SV(t) < 0$  and  $SPI(t) < 1$ . Due to the large degree of activity overlaps, the p-factor shows some values that are significantly lower than its ideal value of 1 indicating a certain risk for rework and loss of earned value.

Figure 4.6 displays the ES performance report for Project 3. The ES line starts with a small delay but shows an improvement and even an ahead of schedule situation from week 9 on. The  $SV(t)$  and  $SPI(t)$  metrics demonstrate this promising project progress by values larger than 0 and 1, respectively. The p-factor is relatively stable and close to one, except for the review periods 9, 14 and 15.

# Chapter 5

## Forecasting

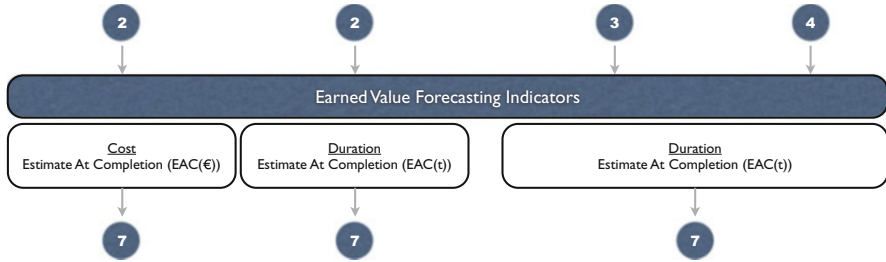
One of the primary tasks of a project manager is making decisions about the future during project progress. EVM systems are designed to follow up the performance of a project and to act as a warning signal to take corrective actions in the (near) future. Forecasting the total project cost and the time to completion is crucial to take corrective actions when problems or opportunities arise and hence, the performance measures will be mainly used as early warning signals to detect these project problems and/or opportunities.

This chapter gives an overview of the predictive EVM/ES metrics designed to forecast the expected total cost and project duration based on the actual performance up to date and the assumptions about future performance. Figure 5.1 displays the relevant parts of Fig. 1.8 that will be discussed in this chapter.

Section 5.1 presents a general formula for time and cost forecasting which uses a performance factor to express the assumption about future expected performance. In Sect. 5.2, eight cost forecasting and nine time forecasting methods are shown. Section 5.3 elaborates on the accuracy and stability of forecasting methods and presents some techniques to validate these quality measures. Section 5.4 gives a summary of the forecasting techniques of this chapter and draws general conclusions. The appendix displays three summary reports showing an overview of the forecast analyses for the example project data.

### 5.1 General Principle

In the previous chapter, it has been shown how the three EVM metrics PV, AC and EV and their extension to the fourth ES metric can be used to measure the schedule and cost performance of a project in progress at a particular point in time (i.e. actual time AT). These observed performances give an indication of the progress of the work done in the past and should ideally reflect the future performance of



**Fig. 5.1** Forecasting piece of the jigsaw puzzle of Fig. 1.8

the work yet to be done. In doing so, the manager can have a relatively good idea about the expected time and cost performance and can use the EVM/ES system as an early warning indicator when schedule and cost margins (safety time and cost above the PD and BAC, respectively) tend to be consumed completely. Fortunately, the EVM/ES metrics are designed to forecast these two important performance measures (time and cost) based on the actual performance up to date and the assumptions about future performance. Setting margins on project durations and costs using uncertainty information in activity durations can be done using Schedule Risk Analysis and will be discussed in Chap. 6. This chapter gives an overview of the existing predictors of final project duration and cost and provides some practical guidelines on how the quality of these techniques can be evaluated based on project characteristics.

### 5.1.1 General Formulae

This section presents the general formulae used to predict the final project duration and cost, while details will be given in the subsequent sections. The abbreviation AC is used for the actual costs made at the current review moment and AT is used to refer to the actual time or the time the project is currently in progress, as previously defined in Sect. 2.1.2. The time forecasts  $EAC(t)$  made at each review period can easily be calculated in the following way:

PDWR	Planned Duration of Work Remaining → Calculated at each review period = An estimate for the remaining duration of the project
$EAC(t)$	Estimated duration at completion → Calculated at each review period as $AT + PDWR$ = An estimate for the total project duration RD

Similarly, the cost forecasts  $EAC(€)$  are calculated using the following two metrics:

PCWR	Planned Cost of Work Remaining → Calculated at each review period = An estimate for the remaining cost of the project
$EAC(€)$	Estimated cost at completion → Calculated at each review period as $AC + PCWR$ = An estimate for the total project cost RC

$EAC$  is an abbreviation for Expected At Completion and is used to measure the expected time or cost at the project finish to be compared with the PD and BAC. In order to distinguish between time and cost forecasts, the € symbol is added ( $EAC(€)$ ) to refer to cost forecasting and a t between brackets is added ( $EAC(t)$ ) to refer to time forecasting.<sup>1</sup>

The time overrun  $EAC(t) - PD$  is often referred to as the project slippage and the estimated final cost overrun is equal to  $EAC(€) - BAC$ . Forecasting the total expected duration and cost of a project in progress boils down to estimating accurate values for the PDWR and PCWR. The way these estimates are made lies at the heart of EVM/ES performance management and takes all the metrics of previous chapters into account. Mathematical details of both forecasting methods are given in the next two sections.

### 5.1.2 Performance Factor

The PDWR and PCWR metrics described in the previous section are the components that have to be estimated in order to obtain a forecast on the time and/or cost of the project, and heavily depend on the specific characteristics and the current status of the project as well as on the assumptions about future project progress (Anbari 2003).

These assumptions about the expected future project performance will be taken into account when predicting the PDWR and PCWR to predict  $EAC(€)$  and  $EAC(t)$  by using a *performance factor* (PF) in all the prediction methods. This performance factor (PF) is used to adapt the future performance to the past performance (depending on the project characteristics) and is based on the work done by Vandevoorde and Vanhoucke (2006). The three PF situations to determine the PDWR/PCWR can be summarised along the following lines:

- PDWR/PCWR according to plan: Past performance is not a good predictor of future performance. Project problems or opportunities of the past will not affect the future, and the remaining work is expected to be done according to

---

<sup>1</sup>In other sources in literature,  $EAC$  is used for cost forecasting and  $EAC(t)$  for time forecasting.

the planned work of the baseline schedule. In this situation, deviations from the expected time/cost performance as predicted in the baseline schedule are assumed to be one-time deviations and have no impact on the remaining work. In this case, the performance factor (PF) is equal to 1.

- PDWR/PCWR according to current time *or* cost performance: Past performance is a good predictor of future performance. In this case, problems or opportunities of the past will affect future performance, and the remaining work will be corrected for the observed efficiencies or inefficiencies of the past. In this scenario, the performance factor is equal to SPI or SPI(t) in case of time forecasting and equal to CPI in case of cost forecasting.
- PDWR/PCWR according to current time *and* cost performance: Past cost and schedule problems are good indicators for future performance. It is assumed that the cost and schedule management are inseparable and hence the current performance is measured by a combined time and cost performance. In this case, the performance factor is equal to the Schedule Cost Index defined as  $SCI = SPI * CPI$ , or as  $SCI(t) = SPI(t) * CPI$  or as a slightly modified version of these SCI or SCI(t) formulas.

In the next section, the performance factor is used to define the various time and cost forecasting methods using the general principle described earlier.

## 5.2 Forecasting Methods

This section reviews nine time forecasting methods and eight cost forecasting methods based on the performance measures discussed previously in Chaps. 2 and 3.

### 5.2.1 Time Forecasting Methods

In this section, nine forecasting methods will be discussed. These nine methods are the result of three fundamentally different methods proposed by three authors, and each method can be calculated under the three performance factor (PF) situations.

Table 5.1 gives an overview of the nine forecasting methods. The rows under “performance factor” make a distinction between the three different project situations based on the PF classification described in the previous section. Each situation involves another assumption about the future performance of the work to finish the project. The first project situation assumes that the remaining work PDWR will be done according to the baseline schedule, the second situation assumes the future work will follow the current SPI or SPI(t) trend while the third situation assumes that the remaining work will follow the current SCI or SCI(t) trend. The columns under “forecasting method” display the three methods to forecast the final project

**Table 5.1** Time forecasting methods (EAC(t))

Performance factor	Forecasting method		
	Planned value	Earned duration	Earned schedule
PF = 1	$EAC(t)_{PV1}$	$EAC(t)_{ED1}$	$EAC(t)_{ES1}$
PF = SPI or SPI(t)	$EAC(t)_{PV2}$	$EAC(t)_{ED2}$	$EAC(t)_{ES2}$
PF = SCI or SCI(t)	$EAC(t)_{PV3}$	$EAC(t)_{ED3}$	$EAC(t)_{ES3}$

duration, known as the planned value method (Anbari 2003), the earned duration method (Jacob 2003) and the earned schedule method (Lipke 2003).

In this section, various time metrics are used to express the duration of a project. The planned duration (PD) and actual time (AT) have been defined earlier while the earned duration is new and will be used later in this section. A summary is given along the following lines:

- PD           Planned Duration of the project  
              → defined in Chap. 2
- AT           Actual Time  
              → actual duration that the project is in progress
- ED           Earned Duration of the project  
              → used in the ED forecasting method defined later in this section  
              = AT \* SPI

**Planned Value Method:** The planned value method does not directly give an estimate for the PDWR but relies on the planned value rate, which is equal to the average planned value per time period, i.e.  $PV_{rate} = \frac{BAC}{PD}$  where BAC is used to denote the budget at completion as defined in Chap. 2. This method assumes that the schedule variance can be translated into time units by dividing it by the planned value rate, resulting in the time variance TV as follows:

$$TV = \frac{SV}{PV_{Rate}} = \frac{SV * PD}{BAC} = \frac{(EV - PV) * PD}{BAC}$$

Depending on the PF method (Table 5.1), the following forecasting formulas can be used:

$$EAC(t)_{PV1} = PD - TV$$

$$EAC(t)_{PV2} = PD/SPI$$

$$EAC(t)_{PV3} = PD/SCI$$

As an example, the final duration of Project 1 at  $AT = 4$  can be predicted as follows: It is known from Fig. 2.3 that  $PV = \text{€}40,000$  and  $EV = \text{€}32,675$  at



AT = 4 and hence  $TV = \frac{(32.675-40.000)*11}{71.500} = -1.13$ . Given the SPI = 0.816875 and CPI = 0.682150 values, shown as rounded up or down values in Fig. 2.6, the EAC(t) values are equal to  $EAC(t)_{PV1} = 11 - (-1.13) = 12.13$ ,  $EAC(t)_{PV2} = \frac{11}{0.816875} = 13.47$  and  $EAC(t)_{PV3} = \frac{11}{0.816875*0.682150} = 19.74$ .

**Earned Duration Method:** The earned duration (ED) is defined as the product of the actual duration the project is in progress (AT) and the schedule performance index SPI, i.e.  $ED = AT * SPI$ . The generic earned duration forecasting formula is given by the following equation:

$$EAC(t)_{ED} = AT + \frac{\max(PD, AT) - ED}{PF}$$

In a situation where the current actual time (i.e. the actual duration that the project is in progress) is smaller than its planned duration PD (i.e.  $AT < PD$ ), the formulas for the three PF methods can be written as follows:

$$EAC(t)_{ED1} = AT + (PD - ED) = PD + AT * (1 - SPI)$$

$$EAC(t)_{ED2} = AT + \frac{PD - ED}{SPI} = \frac{PD}{SPI}$$

$$EAC(t)_{ED3} = AT + \frac{PD - ED}{SCI} = \frac{PD}{SCI} + AT * (1 - \frac{1}{CPI})$$

In a situation where the actual project duration exceeds the planned duration (i.e.  $AT > PD$ ), and the work is not yet completed, the PD will be substituted by the AT in the above mentioned formulas. In this case, the formulas are:

$$EAC(t)_{ED1} = AT + (AT - ED) = AT * (2 - SPI)$$

$$EAC(t)_{ED2} = AT + \frac{AT - ED}{SPI} = \frac{AT}{SPI}$$

$$EAC(t)_{ED3} = AT + \frac{AT - ED}{SCI} = AT * (1 - \frac{1}{CPI} + \frac{1}{SCI})$$

The final duration predictive values for Project 1 at AT = 4 can be calculated as  $EAC(t)_{ED1} = 11 + 4 * (1 - 0.816875) = 11.73$ ,  $EAC(t)_{ED2} = \frac{11}{0.816875} = 13.47$  and  $EAC(t)_{ED3} = \frac{11}{0.816875*0.682150} + 4 * (1 - \frac{1}{0.682150}) = 17.88$ .

**Earned Schedule Method:** The earned schedule metric has been discussed in Chap. 3 as a time based alternative to the traditional earned value method. The generic earned schedule duration forecasting formula is given by:

$$EAC(t)_{ES} = AT + \frac{PD - ES}{PF}$$

Using the three PF methods, the following ES formulas can be derived:

$$EAC(t)_{ES1} = AT + (PD - ES)$$

$$EAC(t)_{ES2} = AT + \frac{PD - ES}{SPI(t)}$$

$$EAC(t)_{ES3} = AT + \frac{PD - ES}{CPI * SPI(t)} = AT + \frac{PD - ES}{SCI(t)}$$

The final duration of Project 1 at  $AT = 4$  can be predicted as follows: Given an ES value equal to  $3 + \frac{32,675-31,000}{40,000-31,000} = 3.18611$ , for which a rounded up value (3.19) is given in Fig. 3.4, and an  $SPI(t)$  value of  $\frac{3.186}{4} = 0.796527$ , shown as a rounded up value (0.80) in Fig. 3.5, the  $EAC(t)$  values are equal to  $EAC(t)_{ES1} = 4 + (11 - 3.186) = 11.81$ ,  $EAC(t)_{ES2} = 4 + \frac{11-3.186}{0.796527} = 13.81$  and  $EAC(t)_{ES3} = 4 + \frac{11-3.19}{0.796527*0.68215} = 18.38$ .

Figure 5.2 displays the three forecasting methods along the life of Project 1 for the three PF methods. The top graph displays the three PF versions of the planned value method. The graph in the middle shows the earned duration forecasting methods and the graph at the bottom visualises the earned schedule forecasting methods. The horizontal dashed line lies at 14 and displays the real project duration ( $RD = 14$  weeks). All points below this line denote an underestimation while the dots above the line refer to an overestimation.

### 5.2.2 Cost Forecasting Methods

Cost performance and forecasting have been widely investigated by numerous researchers, and a detailed description is outside the scope of this chapter. For an overview, the reader is referred to Christensen (1993), who reviewed different  $EAC(\text{€})$  formulas. Most of the cost forecasting methods described below have been validated in literature and make use of the following generic cost forecasting formula:

$$EAC(\text{€}) = AC + \frac{BAC - EV}{PF}$$

Similar to the time prediction methods, each cost forecasting method relies on three possible performance factor (PF) values which make assumptions about future cost performance. The PF versions that rely on time or cost performance contain three forecasting methods, while the PF versions that rely on time and cost performance contain four methods. Moreover, it should be noted that PF methods using time performance can be done in two ways, one using the traditional Schedule Performance Index (SPI) assumptions, and another with the new Earned Schedule

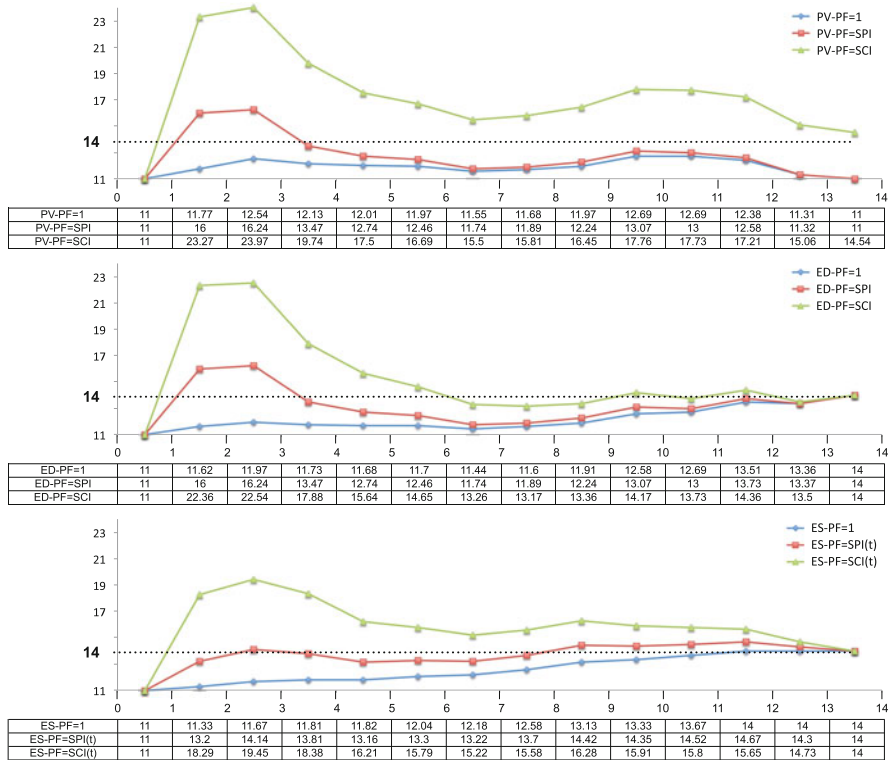


Fig. 5.2 Project 1. Nine time forecasting methods

Table 5.2 Cost forecasting methods (EAC(€))

Performance factor	Forecasting method
PF = 1	$EAC(€)_{PF=1}$
PF = SPI	$EAC(€)_{PF=SPI}$
PF = SPI(t)	$EAC(€)_{PF=SPI(t)}$
PF = CPI	$EAC(€)_{PF=CPI}$
PF = CPI and SPI	$EAC(€)_{PF=CPI*SPI}$
	$EAC(€)_{PF=CPI*SPI(t)}$
	$EAC(€)_{PF=0.8*CPI+0.2*SPI}$
	$EAC(€)_{PF=0.8*CPI+0.2*SPI(t)}$

based Schedule Performance Index (SPI(t)). All these methods provide an estimate for the total predicted cost at the end of the project, and offer a range of possibilities and hence a lower and upper bound on what the project manager expects to spend on the project. An overview of the eight methods to forecast the final expected project cost is given in Table 5.2.

Figure 5.3 displays four weekly cost forecasts along the life of Project 1 for the eight cost forecasting methods. The abbreviation  $w(x,y)$  is used to refer to the weighted method with a performance factor  $PF = 0.8 * x + 0.2 * y$  with

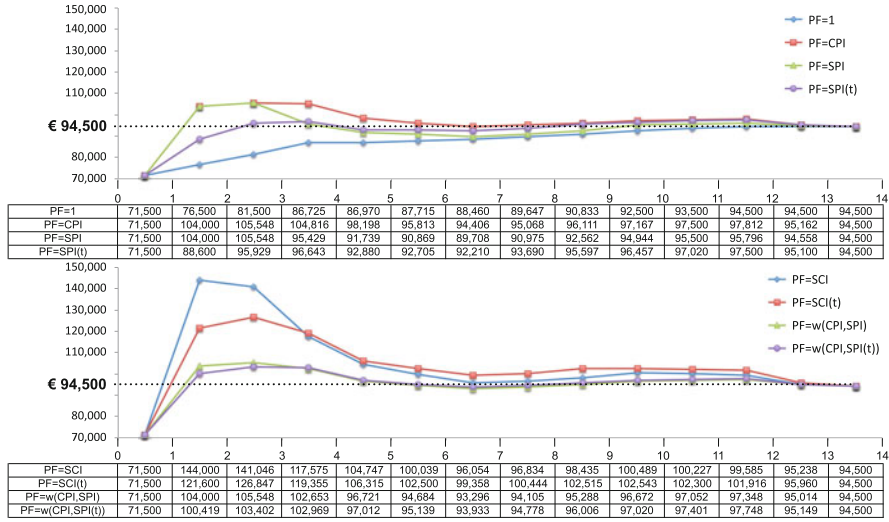


Fig. 5.3 Project 1. Eight cost forecasting methods

$x = \text{CPI}$  and  $y = \text{SPI}$  or  $\text{SPI}(t)$ . Similar to the time forecasting graphs of Fig. 5.2, dots below the horizontal line denote underestimations, while dots lying above the line denote overestimations. The horizontal line lies at €94,500 which represents the real project cost known at the project’s finish.

The calculations are illustrated for week  $AT = 4$ . Given the values for the performance measures  $\text{SPI} = 0.816875$ ,  $\text{CPI} = 0.682150$  and  $\text{SPI}(t) = 0.796527$  at week 4 (see Figs. 2.6 and 3.5 for rounded values) and an actual cost  $\text{AC} = \text{€}47,900$  (see Fig. 2.3), the cost forecasts can be calculated as  $\text{EAC}(\text{€})_{\text{PF}=1} = 47,900 + (71,500 - 32,675) = \text{€}86,725$ ,  $\text{EAC}(\text{€})_{\text{PF}=\text{CPI}} = 47,900 + \frac{71,500 - 32,675}{0.682150} = \text{€}104,816$ ,  $\text{EAC}(\text{€})_{\text{PF}=\text{SPI}} = 47,900 + \frac{71,500 - 32,675}{0.816875} = \text{€}95,429$ ,  $\text{EAC}(\text{€})_{\text{PF}=\text{SPI}(t)} = 47,900 + \frac{71,500 - 32,675}{0.796527} = \text{€}96,643$ ,  $\text{EAC}(\text{€})_{\text{PF}=\text{CPI}*\text{SPI}} = 47,900 + \frac{71,500 - 32,675}{0.682150*0.816875} = \text{€}117,575$ ,  $\text{EAC}(\text{€})_{\text{PF}=\text{CPI}*\text{SPI}(t)} = 47,900 + \frac{71,500 - 32,675}{0.682150*0.796527} = \text{€}119,355$ ,  $\text{EAC}(\text{€})_{\text{PF}=0.8*\text{CPI}+0.2*\text{SPI}} = 47,900 + \frac{71,500 - 32,675}{0.8*0.682150+0.2*0.816875} = \text{€}102,653$  and  $\text{EAC}(\text{€})_{\text{PF}=0.8*\text{CPI}+0.2*\text{SPI}(t)} = 47,900 + \frac{71,500 - 32,675}{0.8*0.682150+0.2*0.796527} = \text{€}102,969$ .

### 5.3 Forecasting Quality

The nine time forecasting methods and eight cost forecasting methods presented in the previous sections can be used to estimate the final project duration and/or cost at each review period  $r$  (at  $AT$ ) based on EVM performance data. However, given the multitude of forecasting techniques, it is often hard to decide which of these methods

provides the most reliable result for forecasting and decision making. Nevertheless, the quality of these forecasting methods is crucial to make effective decisions for projects in progress. It is therefore interesting to have a continuous validation of their quality, which depends on project characteristics such as the project network logic (e.g. the Serial/Parallel indicator (SP) of Chap. 1), the size of the project, the magnitude of the activity durations and costs, and more.

In this section, the quality of the previous time and cost forecasting methods is discussed using two different criteria, known as forecasting *accuracy* and *stability*. In order to present the accuracy and stability metrics, the index  $r$  is added to the EAC(t) and EAC(€) abbreviations used in the previous sections to refer to the forecast at review period  $r$  ( $r = 1, \dots, R$ ). A review period  $r$  is a period in which the EVM performance metrics are calculated at a certain moment in time (i.e. actual time AT) and the time and forecasting metrics are calculated. The following abbreviations will be used:

R	Number of review periods
EAC(t) <sup>r</sup>	Time forecasting at review period $r$ ( $r = 1, \dots, R$ )
EAC(€) <sup>r</sup>	Cost forecasting at period $r$ ( $r = 1, \dots, R$ )

The two criteria to evaluate the quality of the forecasting methods are inspired by the three criteria proposed by Covach et al. (1981) for evaluating the performance of EAC methods, i.e. accuracy, timeliness and stability. The calculations to evaluate the forecasting *accuracy* are given in Sect. 5.3.1. The *stability* calculations for the time and cost forecasting methods are given in Sect. 5.3.2. The *timeliness* criterium is only implicitly taken into account in this chapter. Timeliness refers to the time window in which the forecasting quality is measured in order to evaluate the accuracy and stability of forecasting methods along the completion stages of the projects. It is used to detect whether the forecasting methods produce reliable results in the early, middle and late stages of the project life cycle. In the next sections, it is assumed that all calculations are made for all EAC(t) and EAC(€) methods over all review periods R, i.e. from  $r = 1$  to R. However, smaller time intervals  $[t_1, t_2]$  can be selected to validate the accuracy and stability of the forecasting methods, with  $t_1 > 1$  and  $t_2 < R$  to vary the settings of the timeliness criterium.

### 5.3.1 Accuracy

In order to evaluate the forecast accuracy of the time and cost forecasting methods, various metrics have been developed that measure the error of the forecasts along all review periods. The forecasting error is defined as the deviation between the value of the forecast and the real observed value of the variable. Consequently, the accuracy measures must calculate the average deviations between the periodic EAC(t) and EAC(€) forecasts made during project progress and the final project outcome (real duration (RD) and real cost (RC)) known at the project finish.

**Table 5.3** Project 1. Time forecasting accuracy

PF	Planned value			Earned duration			Earned schedule		
	1	SPI	SCI	1	SPI	SCI	1	SPI(t)	SCI(t)
MAPE (t)	14.95	13.13	26.65	12.86	9.96	15.09	9.92	4.60	16.47
MPE (t)	-14.95	-8.80	23.59	-12.86	-5.64	8.99	-9.92	-2.15	13.41

In this section, two straightforward measures are presented. Statistically, the Mean Absolute Percentage Error (MAPE) is defined as the average of the absolute percentage errors over all review periods  $r = 1, \dots, R$ . Obviously, the lower the MAPE value, the more accurate the predictions. The MPE provides the average of percentage errors based on total errors regardless of the sign. MPE values lower than zero then denote an underestimation while MPE value above zero refer to overestimations. However, since the averaging process sums up positive and negative percentage errors allowing them to cancel each other out, the MPE can often display unrealistic low values.

The time forecasting accuracy calculations are given along the following lines:

$$\begin{aligned} \text{MAPE}(t) & \quad \text{Mean Absolute Percentage Error for EAC}(t) \\ & = \frac{1}{R} \sum_{r=1}^R \frac{|EAC(t)^r - RD|}{RD} * 100 \end{aligned}$$

$$\begin{aligned} \text{MPE}(t) & \quad \text{Mean Percentage Error for EAC}(t) \\ & = \frac{1}{R} \sum_{r=1}^R \frac{EAC(t)^r - RD}{RD} * 100 \end{aligned}$$

Similarly, the cost forecasting accuracy calculations are given by:

$$\begin{aligned} \text{MAPE}(\text{€}) & \quad \text{Mean Absolute Percentage Error for EAC}(\text{€}) \\ & = \frac{1}{R} \sum_{r=1}^R \frac{|EAC(\text{€})^r - RC|}{RC} * 100 \end{aligned}$$

$$\begin{aligned} \text{MPE}(\text{€}) & \quad \text{Mean Percentage Error for EAC}(\text{€}) \\ & = \frac{1}{R} \sum_{r=1}^R \frac{EAC(\text{€})^r - RC}{RC} * 100 \end{aligned}$$

Table 5.3 displays the accuracy values for MAPE and MPE for the nine time forecasting methods for Project 1 and shows that the EAC(t)<sub>ES2</sub> method (with PF = SPI(t)) performs best. Similarly, Table 5.4 shows the accuracy values for the eight cost forecasting methods, and shows that the method with performance factor (PF) equal to SPI(t) performs best, while most alternative methods also perform reasonably well.

**Table 5.4** Project 1. Cost forecasting accuracy

PF	1	CPI	SPI	SPI(t)	SCI	SCI(t)	w(CPI,SPI)	w(CPI,SPI(t))
MAPE (€)	7.08	5.35	4.83	3.64	13.85	12.90	4.88	4.62
MPE (€)	-7.08	1.86	-1.16	-1.71	10.38	9.42	1.16	1.06

A major drawback of the accuracy metrics is that their value is only known upon project finish and that such knowledge probably is of little value for the project under study and might only be relevant for future similar projects. However, the MAPE and MPE metrics can be calculated *ex ante*, before the start of the project, using predefined activity duration and cost distributions and Monte Carlo simulations. These simulations imitate the fictitious progress of the project by generating random variations in both activity durations and costs under a controlled design. These fictitious project executions result in time-based values for the actual cost (AC) and earned value (EV) which can be translated into EVM performance and forecasting metrics. After each simulation run, the project outcomes RD and RC are known and the EAC(t) and EAC(€) metrics can be evaluated using the MAPE and MPE metrics. This knowledge is then available prior to the project start and can be used as a selection tool for the most accurate forecasting method. In Chap. 6, Monte Carlo simulations are also used to analyse the impact of random variations on the project outcome.

### 5.3.2 Stability

The accuracy of the estimates is measured by averaging the error made during forecasting and gives an indication of the average quality of the predicted values for total project duration and cost. However, forecasts are made at regular periods and knowledge on the evolution of these predictions between consecutive periods is an important driver to support decisions. Stability in the forecasted values is often as important as the accuracy of these values. Hence these two forecasting quality measures should both be evaluated when selecting a forecasting method.

A forecast is considered to be perfectly stable when the forecasted value for any period equals the previous period's forecasted value. The simplest and perhaps most natural way to measure stability is the standard deviation of the forecasts over all review periods. Therefore, the time forecasting stability calculations are given along the following lines:

$$\begin{aligned} \overline{\text{EAC}}(t) & \quad \text{Average time forecast over all review periods } R \\ & = \frac{1}{R} \sum_{r=1}^R \text{EAC}(t)^r \end{aligned}$$

**Table 5.5** Project 1. Time forecasting stability

PF	Planned value			Earned duration			Earned schedule		
	1	SPI	SCI	1	SPI	SCI	1	SPI(t)	SCI(t)
$\sigma_{EAC(t)}$	0.56	1.61	3.34	0.89	1.48	3.40	1.06	0.94	2.04

**Table 5.6** Project 1. Cost forecasting stability

PF	1	CPI	SPI	SPI(t)	SCI	SCI(t)	w(CPI,SPI)	w(CPI,SPI(t))
$\sigma_{EAC(€)}$	7,001.19	8,087.66	7,786.16	6,599.66	18,783.04	13,379.18	7,930.81	7,525.24

$\sigma_{EAC(t)}$  Standard deviation of the time forecasts over all review periods R

$$= \sqrt{\frac{1}{R-1} \sum_{r=1}^R (EAC(t)^r - \overline{EAC(t)})^2}$$

Similarly, the cost forecasting stability calculations are given by:

$\overline{EAC(€)^r}$  Average cost forecast over all review period R

$$= \frac{1}{R} \sum_{r=1}^R EAC(€)^r$$

$\sigma_{EAC(€)}$  Standard deviation of cost forecasts over all review periods R

$$= \sqrt{\frac{1}{R-1} \sum_{r=1}^R (EAC(€)^r - \overline{EAC(€)})^2}$$

Table 5.5 displays the stability values for the nine time forecasting methods for Project 1 and shows that the EAC(t) methods with a performance factor (PF) of 1 perform best. Similarly, Table 5.6 shows the stability values for the eight cost forecasting methods, and shows that all methods perform more or less equally well, except for the methods with performance factors equal to SCI and SCI(t). It should however be noted that the results for the accuracy and stability values for Project 1 cannot be generalised and only serve an illustrative purpose.

The research on stability in project performance metrics dates back to the research efforts done by David Christensen (see e.g. Christensen 1993, 2004; Christensen et al. 1995; Christensen and Heise 1993; Christensen and Payne 1992; Christensen and Rees 2002; Christensen and Templin 2002) on the US DoD project dataset. Some of the results found in these studies have come to be regarded as being generally applicable to all types of projects; the main result being that the



cumulative CPI stabilises by the time the project is 20 % complete,<sup>2</sup> where stability is defined to mean that the final value does not vary by more than 0.10 from the value at 20 % complete. However, other studies (e.g. [Henderson and Zwikael 2008](#); [Lipke et al. 2009](#)) have shown that the stability results could not be generalised to all types of projects.

Unlike the accuracy metrics, the stability metrics do not require knowledge of the project outcome and therefore, these stability metrics can be calculated and updated along the project progress. This way, the stability metrics can be used as dynamic evaluation tools during project progress, and no prior analysis using Monte Carlo simulations is necessary.

## 5.4 Conclusions

In this chapter, the performance measures obtained during project progress using EVM/ES calculations have been used to predict the final project outcome. The general formulae for these forecasting methods make use of the known time and cost data from the past as well as a so-called performance factor that is used to steer the estimates about future performance. This performance factor defines the assumptions made for future performance and can be classified into the two following classes:

- According to plan: Future performance is assumed to follow the expected baseline schedule performance, regardless of the performance made in the past.
- According to current time/cost performance: The current time and cost performances are good and reliable indicators for the expected future performance to predict the total duration and cost. The current time and cost performance measures can be used separately, or in an integrated manner.

Based on these performance factor classes, different forecasting methods are available in literature for time and cost forecasting, as follows:

- Time forecasting: Nine forecasting methods, abbreviated by EAC(t) are available taking past time and/or cost performance data into account. These methods can be split into three classes known as the planned value, earned duration and earned schedule methods.
- Cost forecasting: Eight forecasting methods, abbreviated by EAC(€) are available taking past time and/or cost performance data into account.

A crucial component in time and cost forecasting is the validation of the predictions in terms of their accuracy and stability, as summarised along the following lines:

---

<sup>2</sup>Studies on the SPI were performed to a much lesser degree.

- **Accuracy:** The accuracy of a forecast is measured by its absolute or relative error which is only known at the finish of the project. Two accuracy measures, abbreviated by MAPE and MPE, are used in this chapter.
- **Stability:** The stability of a forecast is defined as the variability of the predicted values over the different review periods of the project in progress for which forecasts are made.

Based on estimated values for the forecast accuracy and stability, the reliability of the various forecasting methods can be estimated and the right forecasting method can be selected for the project in progress. Forecasting values are used to predict whether the current performance would lead to schedule and cost deviations and therefore act as triggers for corrective actions. This corrective action mechanism is discussed in Chap. 7.

## Appendix: Forecasting Reports

The appendix to Chap. 5 displays the summary reports for the three example projects. Each summary report contains the relevant information for an EVM/ES forecasting analysis and is built up as follows:

The weekly forecasts for the three best performing cost and time methods are displayed in the top left pane of each report and consist of the figures displayed earlier in this chapter. A short summary is given to the right of each graph. The accuracy and stability data are given in the graphs displayed at the bottom right part of each report. The bar charts contain the MAPE and MPE for the eight cost forecasting methods and nine time forecasting methods. The stability data are given in the pie charts and show the relative distribution of the standard deviations for each time and cost forecasting method. All accuracy and stability values have been calculated based on the weekly forecasts for each project from its start until its finish. Obviously, when projects are in progress, not all relevant data are available, and hence, the calculations of the standard deviations can only be based on partial data (i.e. forecasts done for the review periods in the past). Moreover, at the very beginning of the project, artificial project progress can be simulated using Monte Carlo simulation runs as used in Chap. 6 and the simulated MAPE and MPE values can then be used to select the expected best performing forecasting technique for the project that starts its progress. The bottom left pane of each report draws conclusions on the accuracy and stability estimates and selects the three best performing time and cost forecasting methods (based on the MAPE values).

Figure 5.4 displays the EVM/ES forecasting data of Project 1. It is mainly a summary of the previous figures displayed and discussed in this chapter. Likewise, an EVM/ES forecasting analysis is summarised in Figs. 5.5 and 5.6 for Projects 2 and 3, respectively. A summary for the three best performing forecasting methods

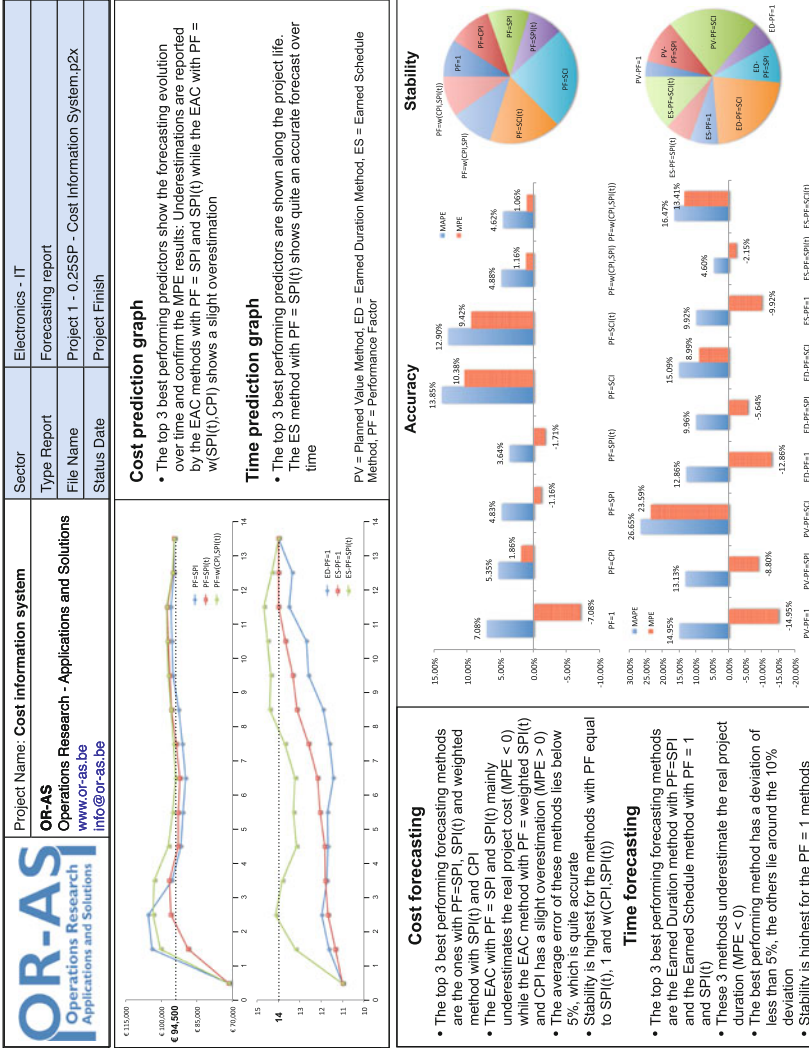


Fig. 5.4 Project 1. An EVM/ES forecasting report

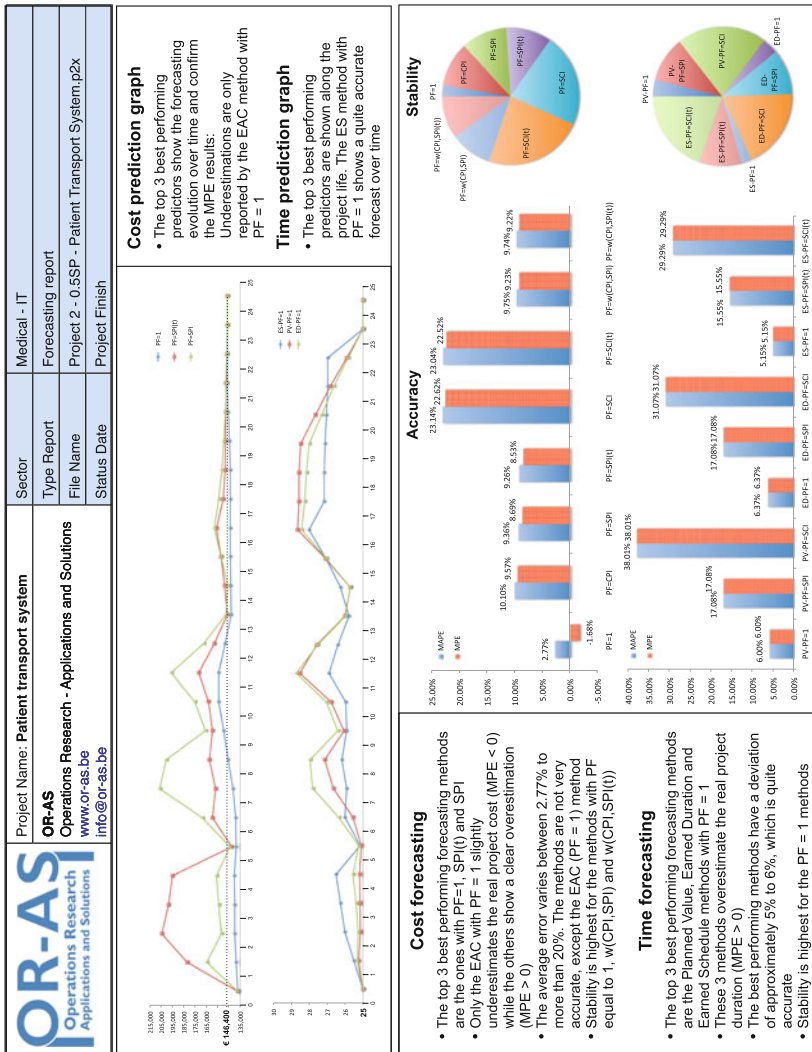


Fig. 5.5 Project 2. An EVM/ES forecasting report

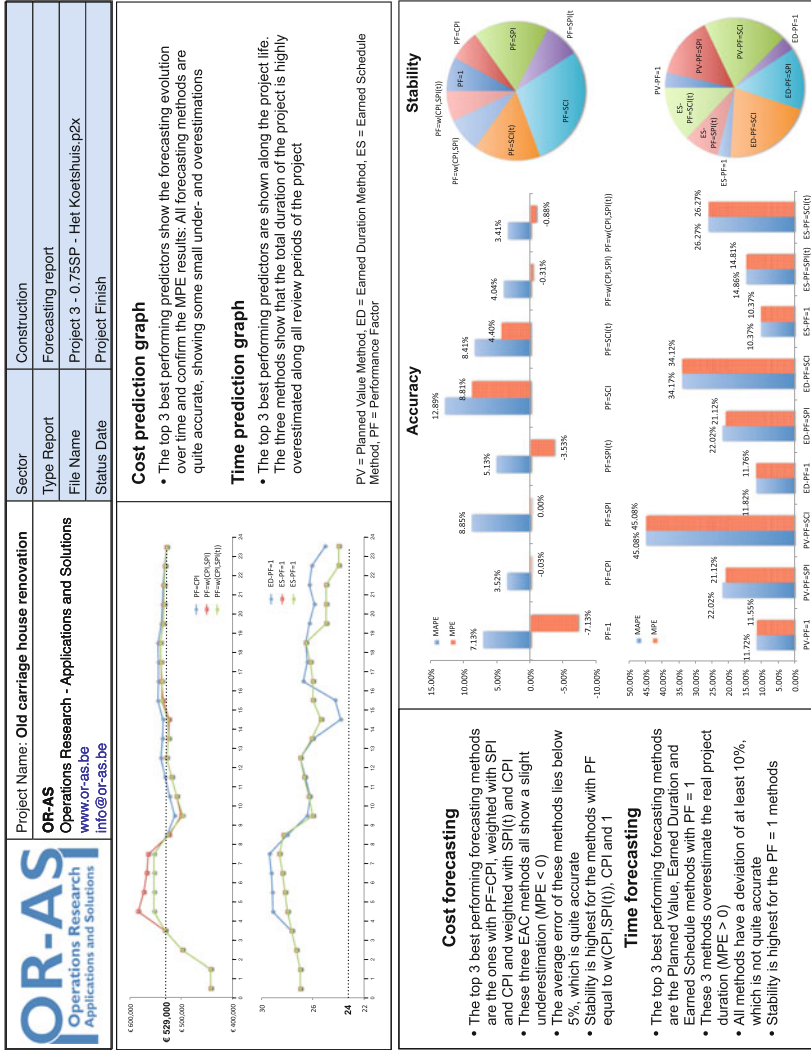


Fig. 5.6 Project 3. An EVM/ES forecasting report

**Table 5.7** The top three best performing forecasting methods for accuracy and stability

	Cost	Time
Project 1	SPI(t), w(CPI,SPI(t)), SPI	ES-SPI(t), ES-1, ED-SPI
Project 2	1, SPI(t), SPI	ES-1, PV-1, ED-1
Project 3	w(CPI,SPI(t)), CPI, w(CPI,SPI),	ES-1, PV-1, ED-1
Project 1	SPI(t), 1, w(CPI,SPI(t))	PV-1, ED-1, ES-1
Project 2	1, w(CPI,SPI), w(CPI,SPI(t))	ES-1, ED-1, PV-1
Project 3	w(CPI,SPI(t)), CPI, 1	ES-1, ED-1, PV-1

is displayed in Table 5.7 for accuracy (top, using the lowest MAPE values) and for stability (bottom). It should be noted that these results are certainly not generally true, but only serve as illustrations. For a more general conclusion on the forecasting accuracy of time predictions, the reader is referred to [Vanhoucke \(2010a\)](#).

# Chapter 6

## Schedule Risk

Projects in progress are typified by uncertainty and risk. They generate deviations from the initial estimates and bring the expected project objectives in danger. Therefore, Schedule Risk Analyses (SRA) are necessary to measure and understand the impact of these unexpected deviations from the baseline schedule on the total expected project duration and cost. They connect the risk information of project activities to the baseline schedule and provide sensitivity information of individual project activities in order to assess the potential impact of uncertainty and to refine the black-and-white focus of the traditional critical path methodology. Similar to EVM, it should, ideally, lead to better management of projects in progress and improved decision making when projects tend to run into trouble.

This chapter gives an overview of the most commonly known sensitivity metrics in SRA and the Monte Carlo technique to generate these metrics. Figure 6.1 displays the relevant parts of Fig. 1.8 that will be discussed in this chapter.

Section 6.1 briefly discusses the relevance of risk measurements on the baseline schedule to extend the concepts of the critical path method. Section 6.2 describes the four step procedure to run a schedule risk analysis. The outcome of this run is a set of sensitivity measures to validate the impact of unexpected changes on the project timing and cost, and is discussed in Sect. 6.3. Section 6.4 gives a summary of the main topics of this chapter and draws general conclusions. The appendix displays three summary reports for the schedule risk analyses done on the three example projects.

### 6.1 Measuring Risk

Analysing project risk using the SRA technique is one of the three crucial elements of integrated project management and control. It is a simple and easy to use technique to provide an answer on the shortcomings of traditional baseline scheduling. Indeed, a baseline schedule constructed by the critical path method

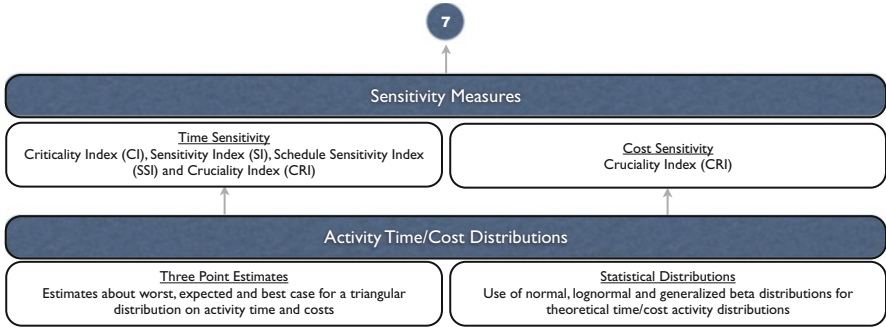


Fig. 6.1 SRA piece of the jigsaw puzzle of Fig. 1.8

(CPM) or its resource related extensions assumes that the activity time and cost estimates as well as all precedence relations between these activities are known with certainty. However, unexpected events occur during project progress, leading to schedule and cost deviations and stressing the need for monitoring systems that measure these deviations as early as possible.

Therefore, prior knowledge of the impact of these unexpected events on the project objectives is useful. It allows the project manager to focus on the project parts with a high expected impact on the total project duration and cost, and to spend less attention on the activities with a minor expected impact on these objectives. This is exactly what the critical path aims to do, but it fails in providing a complete view on the criticality of project activities. While the CPM assumes a black-and-white view on the criticality of project activities (an activity is critical or not), a schedule risk analysis calculates the activity criticality as a percentage ranging from 0 (not critical) to 100 % (highly critical). In this chapter, the SRA technique is discussed using a four step procedure and the sensitivity metrics that it produces are explained in detail.

## 6.2 Schedule Risk Analysis

In this section, the four steps to run a schedule risk analysis are discussed. The first step assumes the construction of a baseline schedule that serves as a point of reference for the other steps. In a second step, uncertainty is defined for the activity time and cost estimates. In a third step, the project is subject to Monte Carlo simulations to imitate real project progress on a computer. In the last step, various sensitivity measures are reported that can be used to analyse the impact of unexpected events on the project outcome.



### 6.2.1 *Baseline Schedule*

The construction of a baseline schedule can be done by the simple and straightforward critical path method that constructs an activity timetable, taking the network logic into account. The Gantt charts displayed in Chap. 1 make use of this technique and consider the earliest start schedule as the point of reference for all other chapters. When these CPM techniques are extended to take into account the presence of resource constraints, other objectives, extra constraints or new activity features, baseline scheduling becomes a complex task.

Similar to the Earned Value Management techniques to monitor the project performance as discussed in the previous chapters, SRA also makes use of this baseline schedule as its point of reference. During a schedule risk analysis, schedule and cost deviations are always measured relative to the baseline estimates. This limited but crucial role of the project baseline schedule is one of the central principles of *dynamic scheduling*. This dynamic scheduling point of view implicitly assumes that the relevance of a project baseline schedule is rather limited since it only acts as a point of reference in the project life cycle. It recognises that unexpected events will result in deviations from the baseline schedule, but it nevertheless relies on these initial baseline estimates to assess the impact of these unexpected events in order to take better corrective actions when the project is in danger. Consequently, dynamic scheduling says that the construction of a *baseline schedule* and the analysis of the *schedule risk* of the project should go hand in hand as a way to be better prepared for the *project control* phase during the progress of the project. The integration of these three crucial elements is called *integrated project management and control* in this book. More information and details on baseline scheduling are outside the scope of this book, and an overview can be found in [Vanhoucke \(2012c\)](#).

### 6.2.2 *Define Uncertainty*

The time and cost estimates for the project activities necessary to construct a baseline schedule are often, if not always, subject to a margin of error. Therefore, these single point estimates do not adequately address the uncertainty inherent to activities. Moreover, estimates about time and cost are predictions for the future, and human beings often tend to be optimistic about them or, on the contrary, often add some safety margin to protect themselves against unexpected events. Consequently, estimates often do not reflect reality.

In order to overcome this bias in the activity estimates, SRA suggests the use of statistical distributions on the activity estimates instead of single point estimates. People tend to feel more comfortable with a range of possible project outcomes rather than with a single point estimate to calculate the critical path length. On top

of that, SRA allows the project manager to assess the impact of uncertainty in the estimates on the project objectives as discussed in the next step. It should be noted that using distributions on activity estimates requires analytical skills and basic knowledge of statistics, which is often perceived as mathematically complex and sometimes theoretical and far from practice. However, a basic understanding of probability and distribution functions allows the project manager to better estimate the impact of unexpected events on the project outcome. The level of detail of the activity distributions can be varied according to the level of expertise in mathematics and statistics, and varies from the use of simple three-point estimates with triangular distributions (similar to a PERT analysis) to the use of predefined statistical distributions available in spreadsheets or dedicated statistical tools. A detailed description is outside the scope of this chapter, and a basic reference to SRA is found in [Hulett \(1996\)](#).

### ***6.2.3 Simulate Progress***

Monte Carlo simulation runs boil down to generating numbers from predefined distributions using a random number generator. In SRA, this Monte Carlo approach is used to generate activity durations and costs that might differ from their original baseline estimates, leading to a change in the set of critical activities and the real project outcome. At each simulation run, the real duration RD and real cost RC are compared to planned duration PD and budget at completion BAC, and the set of critical activities is saved. The number of runs that imitate fictitious project progress must be high to ensure that enough numbers are generated from each activity distribution such that unbiased results can be reported. The average effect of the changes during the runs relative to the baseline schedule estimates is measured by sensitivity metrics that are reported and discussed in Sect. 6.3.

### ***6.2.4 Sensitivity Analysis***

During each simulation run, the simulation engine has recorded all the project data (real durations and costs for all activities and for the project and all critical paths) in order to be able to measure the degree of activity sensitivity and its potential impact on the project objectives. The output of a schedule risk analysis is a set of metrics that define the degree of activity criticality and sensitivity. Each sensitivity metric gives an indication of how sensitive the activity is towards the final project duration and/or cost. The metric values are available upon completion of the simulation run and are used as triggers to focus on the risky activities, which probably require more attention in order to achieve successful project fulfilment. The specific calculations

of each sensitivity measure are discussed in the next section and are based on the work done by Williams (1992) and PMBOK (2004), and are reviewed by Vanhoucke (2010c).

### 6.3 Sensitivity Metrics

In order to present the sensitivity metric formulas, the following notation will be used:

$nrS$	Number of simulation runs (index $s = 1, \dots, nrS$ )
$d_i^s$	Duration of activity $i$ during simulation run $s$
$c_i^s$	Cost of activity $i$ during simulation run $s$
$s_i^s$	Slack of activity $i$ during simulation run $s$
$RD^s$	Real duration of the project under simulation run $s$
$RC^s$	Real cost of the project under simulation run $s$

On top of that, the description will make use of some mathematical operators, as follows:

$\mathbf{1}(G)$	General indicator function $\equiv \begin{cases} 1, & \text{if } G \text{ is true,} \\ 0, & \text{if } G \text{ is false.} \end{cases}$
$\delta_{dur,s}$	Difference between ranking values of $d_i^s$ and $RD^s$ of simulation run $s = rank(d_i^s) - rank(RD^s)$ for $s = 1, \dots, nrS$
$\delta_{cost,s}$	Difference between ranking values of $c_i^s$ and $RC^s$ of simulation run $s = rank(c_i^s) - rank(RC^s)$ for $s = 1, \dots, nrS$
$\bar{X}$	Average value of the variable $X$ $= \sum_{s=1}^{nrS} \frac{X^s}{nrS}$
$\sigma_X$	Standard deviation of the variable $X$ $= \sqrt{\frac{\sum_{s=1}^{nrS} (X^s - \bar{X})^2}{nrS - 1}}$

In the next sections, the variable  $X$  will be equal to the variables  $d_i$  or  $c_i$  on the activity level or RD or RC on the project level. Values for these variables will be reported for each simulation run  $s = 1, \dots, nrS$  and averages and standard deviations can be calculated. These values can also be ranked and the differences in ranking per simulation run, measured by  $\delta_{dur,s}$  and  $\delta_{cost,s}$  can then be calculated. It should be noted that all sensitivity metric formulas will be given in the next sections, but their calculations will not be illustrated on our fictitious project examples as was done for the EVM metrics in the previous chapter. The reason is that all calculations require a lot of computer simulations and extensive automatic calculations that

cannot be repeated on a single sheet of paper. Instead, the results will simply be reported as graphs and summary reports in the appendix of this section.

### 6.3.1 Time Sensitivity

Three well-known time sensitivity metrics for assessing the impact of variability in the activity time estimates are widely used in project management. They make use of the number of times an activity lies on the critical path for all simulations, as well as the slack each activity has during all simulations. These three sensitivity metrics can be calculated using the following formulas:

CI	<p>Criticality Index</p> <p>→ Probability that an activity lies on the critical path</p> $= \frac{1}{nrS} \sum_{s=1}^{nrS} \mathbf{1}(s_i^s = 0)$
SI	<p>Significance Index</p> <p>→ Measures the relative importance of an activity <math>i</math></p> $= \frac{1}{nrS} \sum_{s=1}^{nrS} \left( \frac{d_i^s}{d_i^s + s_i^s} \frac{RD^s}{RD} \right)$
SSI	<p>Schedule Sensitivity Index</p> <p>→ Measures the criticality and significance of an activity <math>i</math></p> $= \frac{\sigma_{d_i}}{\sigma_{RD}} \text{CI}$

Since risk refers to unexpected events that might occur, the probability of its occurrence can range anywhere between 0 and 100%. However, when it does occur, any manager wants to know its impact on the project objectives. Consequently, risk assessment should contain these two risk dimensions, and therefore, risk is measured by the  $risk = probability \times impact$  equation. The three metrics fit this risk equation very well and each highlights one dimension of risk assessment. The CI measures the probability while the SI measures the relative activity importance and the expected impact on the project duration. The SSI takes the product of the probability and impact into account and is therefore a more complete sensitivity measure compared to the two others. Although the  $probability \times impact$  equation is often criticised for its inability to correctly measure risk and for its lack of qualitative risk assessment, it might be considered useful when evaluating several risks to prioritise and evaluate against one another.

### 6.3.2 Time/Cost Sensitivity

A more general way to assess the impact of changes in activity estimates on the project outcome is to calculate the *correlation* between the activity estimates and

total project outcome. This correlation is measured by the cruciality index (CRI) of each activity and can be used for both time and cost project objectives. Three versions of the CRI measure the correlation between the random variables  $d_i$  and RD for time and  $c_i$  and RC for cost based on the simulation runs. These metrics reflect the relative importance of an activity in a statistical way and measure the portion of total project duration/cost uncertainty that can be explained by the uncertainty in the activity estimates. The three versions of the CRI can be calculated using the time/cost sensitivity formulas along the following lines:

$$\begin{aligned}
 \text{CRI}(r) & \quad \text{Cruciality Index} \\
 & \rightarrow \text{Pearson's product moment correlation} \\
 & = \frac{\sum_{s=1}^{nrS} (d_i^s - \bar{d}_i)(RD^s - \overline{RD})}{(nrS - 1)\sigma_{d_i}\sigma_{RD}} \text{ or } \frac{\sum_{s=1}^{nrS} (c_i^s - \bar{c}_i)(RC^s - \overline{RC})}{(nrS - 1)\sigma_{c_i}\sigma_{RC}} \\
 \text{CRI}(\rho) & \quad \text{Cruciality Index} \\
 & \rightarrow \text{Spearman's rank correlation} \\
 & = 1 - \frac{6}{nrS(nrS^2 - 1)} \sum_{s=1}^{nrS} \delta_{dur,s}^2 \text{ or } 1 - \frac{6}{nrS(nrS^2 - 1)} \sum_{s=1}^{nrS} \delta_{cost,s}^2 \\
 \text{CRI}(\tau) & \quad \text{Cruciality Index} \\
 & \rightarrow \text{Kendall's tau rank correlation} \\
 & = \left[ \frac{4}{nrS(nrS - 1)} \sum_{s=1}^{nrS-1} \sum_{\ell=s+1}^{nrS} \mathbf{1} \{ (d_i^\ell - d_i^s)(RD^\ell - RD^s) > 0 \} \right] - 1 \\
 & \text{ or } \left[ \frac{4}{nrS(nrS - 1)} \sum_{s=1}^{nrS-1} \sum_{\ell=s+1}^{nrS} \mathbf{1} \{ (c_i^\ell - c_i^s)(RC^\ell - RC^s) > 0 \} \right] - 1
 \end{aligned}$$

*Pearson's product moment correlation* metric  $\text{CRI}(r)$  is a measure of the degree of linear relationship between two variables. However, the relation between an activity estimate and the total project outcome often follows a nonlinear relation. Therefore, nonlinear correlation measures such as the Spearman rank correlation coefficient or Kendall's tau measure can be used. The *Spearman's rank correlation*  $\text{CRI}(\rho)$  assumes that the values for the variables are converted to ranks and the differences between the ranks of each observation for the two variables are then calculated. *Kendall's tau rank correlation* index  $\text{CRI}(\tau)$  measures the degree of correspondence between two rankings to obtain its correlation value.

Details of calculations require a lot of data obtained by the Monte Carlo runs and are therefore not shown. Unlike many EVM users who measure performance of projects in progress in a simple MS Excel sheet, SRA cannot be done manually using spreadsheets. Instead, the user should rely on dedicated software tools that make the connection between random Monte Carlo number generators and project scheduling software tools. Numerous excellent tools are available on the market

for a very low price, and ProTrack (discussed in Chap. 8) is without any doubt the best choice.<sup>1</sup> The reports of the appendix for the three example projects have been generated by this software tool.

## 6.4 Conclusions

In this chapter, an overview is given of a schedule risk analysis as a simple and easy to use technique to understand the impact of unexpected deviations from the baseline schedule on the total expected project duration and cost. Since projects in progress are typified by uncertainty and risk, the relevance of SRA is high and its correct use is key to understand the expected project behaviour. It allows adding a time and cost safety margin to the traditional CPM time and cost estimates based on predefined activity variation distributions.

A traditional SRA consists of four steps, and starts with the construction of the baseline schedule. Similar to the performance measurement systems of EVM and ES, this baseline schedule acts as a point of reference and all calculations are done relative to this baseline schedule. After defining distributions on the activity durations and/or costs, the Monte Carlo simulation runs generate artificial project progress and measure some important metrics such as the number of times each activity lies on the critical path, the variability in the total project duration and cost and much more.

Based on these calculations, two classes of sensitivity metrics are reported as relevant output metrics. The measures to assess the time sensitivity are given by the following three indices:

- Criticality Index: Probability of an activity to lie on the critical path.
- Significance Index: Relative importance of each activity taking slack calculations into account.
- Schedule Sensitivity Index: Combined metric taking probability and variability calculations into account.

Cost forecasting is done by calculating the correlation between the cost variation for each activity and the variation in the total project cost. Depending on the assumption of linear or nonlinear correlations, three variants can be used, as follows:

---

<sup>1</sup>This statement is based on a very personal opinion and has not been validated by computational experiments. It is a statement based on a deep passion for this software product. Any interested reader can download a book about the work and passion that accompanied the development of ProTrack at [www.or-as.be/books/wp](http://www.or-as.be/books/wp).

- Pearson's product moment correlation
- Spearman's rank correlation
- Kendall's tau rank correlation

The relevance of the output sensitivity metrics lies in their ability to efficiently control projects in progress in order to take corrective actions. This will be further discussed in Chap. 7. Furthermore, it should be noted that the Monte Carlo simulation approach can also be used to calculate entirely different output measures than the ones discussed in this chapter. As an example, the forecasting accuracy and stability metrics discussed in Chap. 5 can easily be calculated using fictitious EVM/ES performance data obtained by the Monte Carlo simulations. More information is available in [Vanhoucke \(2010a\)](#).

## Appendix: Schedule Risk Reports

The appendix to Chap. 6 displays summary reports for the three example projects. Each summary report contains the relevant information for a schedule risk analysis and is built up as follows:

The graphs displayed at the left of each report show the values of the six time (top) and three cost (bottom) sensitivity metrics for each project activity. A more detailed analysis and discussion on the sensitivity metric values are shown in the right panes of each report. The correlations between the metrics are displayed in a table for both the time and cost sensitivity metrics and might be useful to select the most reliable metric(s). Since it is known that the SSI is the most reliable metric for time sensitivity ([Vanhoucke 2010c](#)), the average values of all metrics that are highly correlated with the SSI are shown in a graph in the right pane of each report.

Figure 6.2 displays the risk summary data of Project 1. All simulation runs are based on triangular distributions set on the activity durations with minimum and maximum values below and above 20% of the baseline schedule duration estimate. Obviously, the choice of these distributions should reflect reality and must be made with care, and hence, the currently used triangular distributions are only for illustrative purposes. In literature, it has been conjectured that the use of lognormal and/or generalised beta distributions might reflect real project duration deviations ([Colin and Vanhoucke 2013b](#); [Kuhl et al. 2007](#); [Trietsch et al. 2012](#); [Vanhoucke 2010c, 2011](#)). Likewise, a summary of the risk analyses is given in Figs. 6.3 and 6.4 for Projects 2 and 3, respectively.

The reports clearly indicate a decrease in discriminative power of the time sensitivity metrics along an increase in the value for the Serial/Parallel (SP) indicator. While the discriminative power is quite high for Project 1 (SP = 0.25), it drops to dramatically low values for the CI and SI for Project 3 (SP = 0.75). As an example, most of the activities of Project 3 have a CI value equal to one, and hence, no discrimination can be made between highly sensitive activities and others that

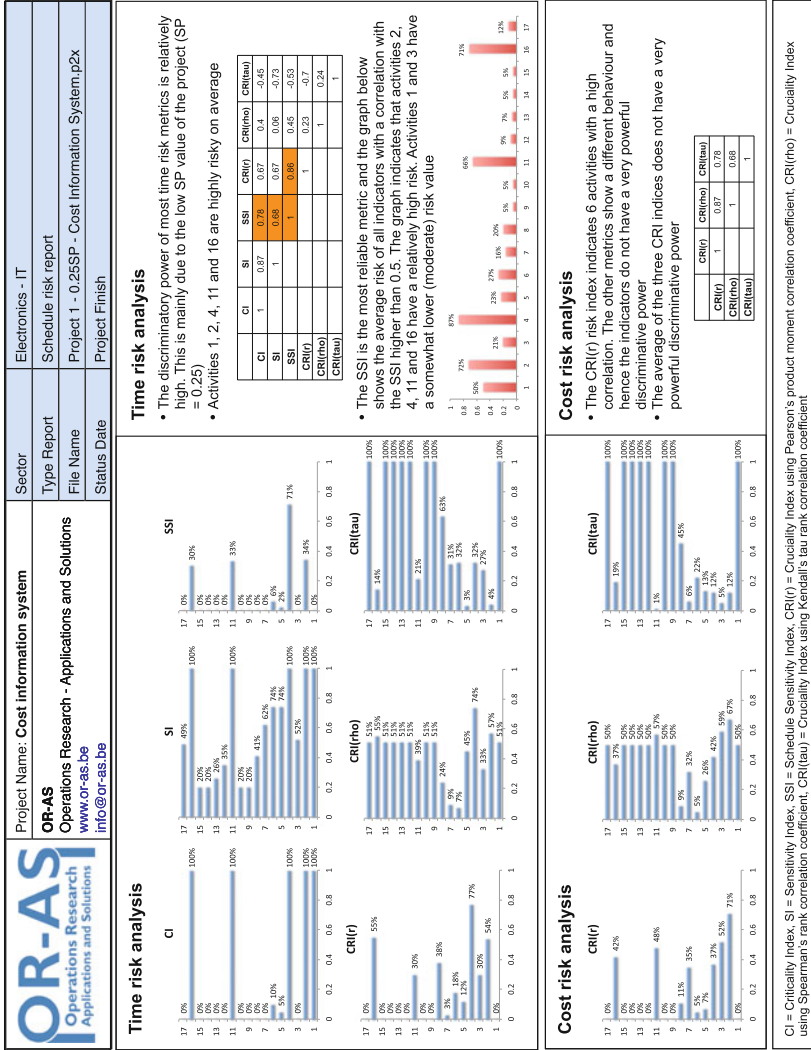


Fig. 6.2 Project 1. A risk summary report



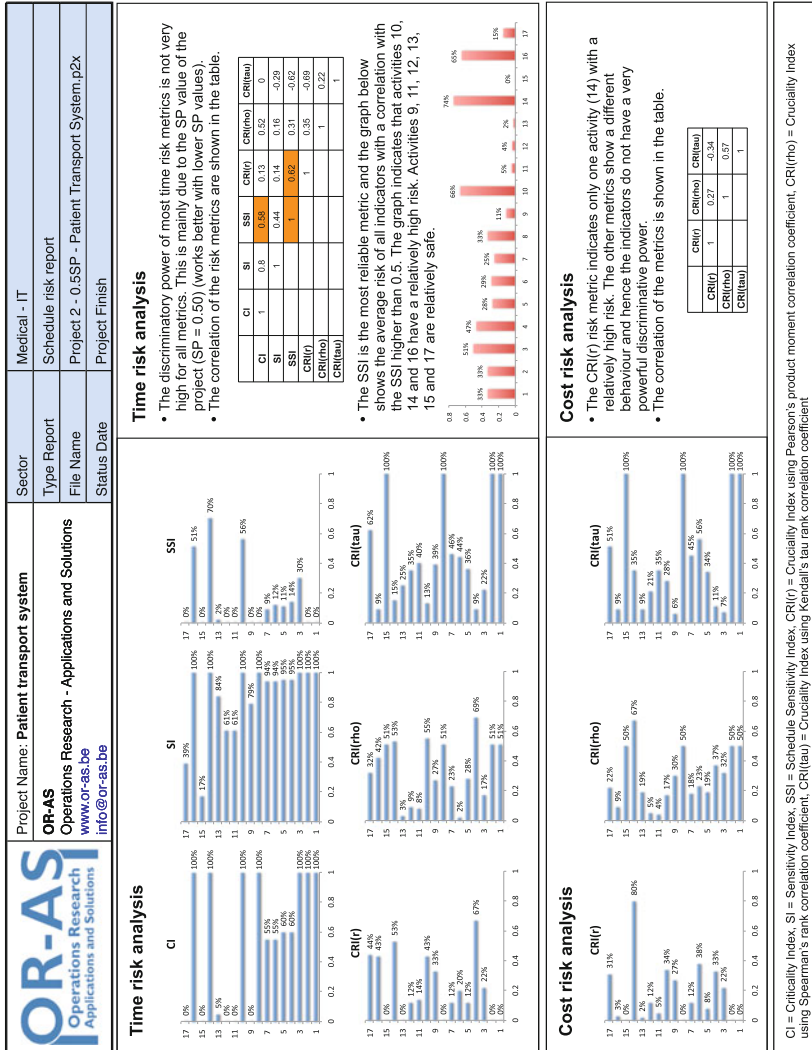


Fig. 6.3 Project 2. A risk summary report

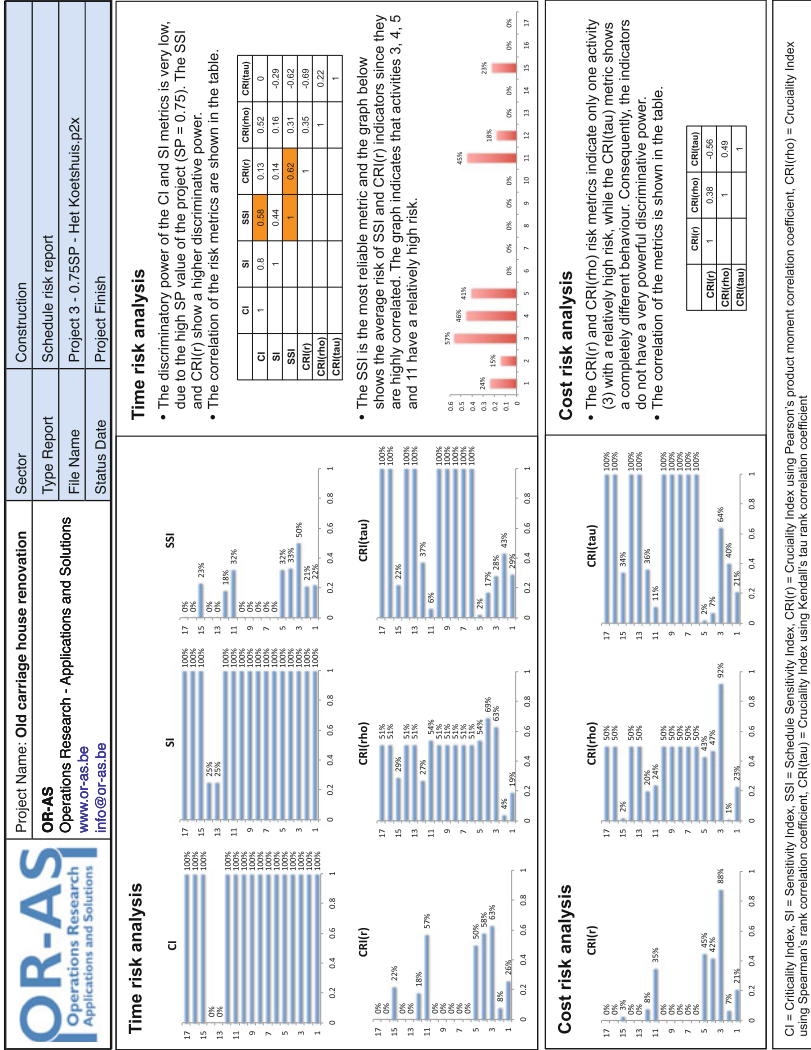


Fig. 6.4 Project 3. A summary risk report

should be classified as almost insensitive. This has also been found in the simulation studies by [Vanhoucke \(2010a,c, 2011, 2013d\)](#) which have clearly shown that the relevance of an SRA study decreases for increasing SP values. This observation will also be repeated in [Chap. 7](#) when the SRA studies are used and validated in a project control efficiency setting.

# Chapter 7

## Schedule Control

Integrating baseline scheduling, schedule risk analysis and project control lies at the heart of dynamic scheduling and is the central theme of this book. This integrated project management and control approach assumes that project scheduling and control involves a continuous stream of changes and is a never-ending process to support decisions that need to be made along the life of the project. The integration of the three key components of dynamic scheduling should allow the project manager to be better prepared for unexpected changes in order to support improved corrective action decision making when the project is in trouble.

This chapter describes how Schedule Risk Analysis and Earned Value Management can be integrated using the project network information and how this can lead to a more efficient project control approach. Figure 7.1 displays the relevant parts of Fig. 1.8 that will be discussed in this chapter.

In Sect. 7.1, the level of control is defined by showing the relevance and use of EVM and SRA in the work breakdown structure (WBS). Section 7.2 describes a top-down control approach using the EVM/ES control techniques discussed in Chaps. 2 and 3. In Sect. 7.3, an alternative bottom-up control approach is discussed using the SRA metrics of Chap. 6. Section 7.4 gives a short overview of the efficiency of these two alternative control approaches based on a research study. Section 7.5 provides a summary of the main topics of this chapter and draws general conclusions.

### 7.1 Level of Control

Efficient project control requires a management focus on the risky elements of the project and corrective actions when necessary that contribute to the overall success of the project. The optimal level of control can be defined in the work breakdown structure by defining the parts of the project that require a high degree of attention. The construction of a Work Breakdown Structure (WBS) is an important step in managing and mastering the inherent complexity of a project. It involves

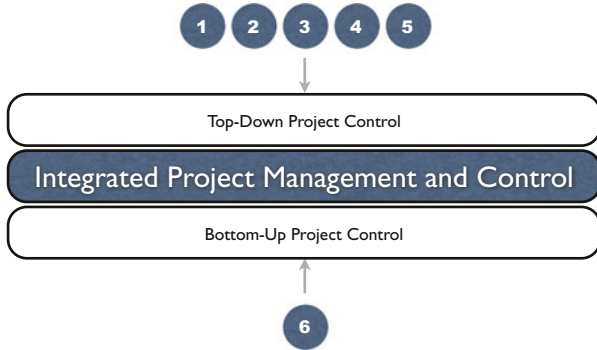


Fig. 7.1 Center piece of the jigsaw puzzle of Fig. 1.8

the decomposition of major project deliverables into smaller, more manageable components until the deliverables are defined in sufficient detail to support the development of project activities (PMBOK 2004). The WBS is a tool that defines the project and groups the project’s discrete work elements to help organise and define the total work scope of the project. It provides the necessary framework for detailed cost estimation and control along with providing guidance for schedule development and control. Each descending level of the WBS represents an increased level of detail of the project work. The WBS is often displayed graphically as a hierarchical tree and has multiple levels of detail. The top node gives a short description of the project scope. Below this root node, various levels can be defined. The second level is the level of the work items in which the project scope is broken down into manageable pieces (items) to be able to cope with the project complexity. Each work item can be further subdivided into work packages. At this level, the collection of time and cost data is important for efficient project control. The lowest WBS level is used for the project activities. At this level, time, cost and resource estimates are calculated and precedence relations are defined, resulting in the project network that forms the input for the construction of a baseline schedule.

Figure 7.2 shows two fictitious work breakdown structures to illustrate the relevance and use of the two alternative project control approaches. The top-down project control is used to refer to a downwards approach, starting from measurements at the top level of the WBS to detect problems or opportunities at the lowest activity level. The details of this project control approach are discussed in Sect. 7.2. Bottom-up project control refers to an opposite control approach in which activity information is used to control the project performance. Actions are necessary when these activities at the lowest WBS levels are in danger, since otherwise they might harm the project objectives displayed at the top WBS levels. This approach is discussed in Sect. 7.3.

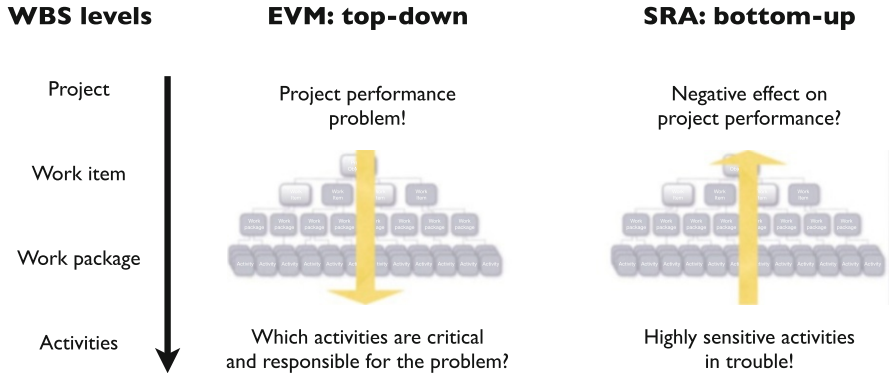


Fig. 7.2 Two alternative project control methods shown in the WBS

## 7.2 Top-Down Project Control

Project control using earned value management should not be considered as an alternative to the well-known critical path based scheduling and control tools. Instead, the EVM methodology offers the project manager a tool to perform a quick and easy sanity check on the work package level or even higher levels of the work breakdown structure. In this respect, an earned value management system is set up as an early warning signal system to detect problems and/or opportunities in an easy and efficient way, which is obviously less accurate than the detailed critical path based scheduling analysis of each individual activity. However, these early warning signals, if analysed properly, indicate the need to eventually drill down into lower WBS levels. In conjunction with the project schedule, it allows taking corrective actions on those activities that are in trouble (especially those tasks which are on the critical path). This control approach is called *top-down project control* to refer to the use of metrics at the project level (high WBS level) as a trigger for actions at the bottom level of the WBS.

The top-down project control approach using EVM provides key performance metrics such as the SPI, SPI(t) and CPI discussed in Chaps. 2 and 3. These metrics should serve as early warning signals to timely detect project problems or opportunities that define the need for actions. To that purpose, managers should set threshold values on these performance metrics that act as triggers for taking action during project progress. Ideally, these thresholds should be set at such values so that they give a certain (high) guarantee that exceeding these thresholds means that real underlying project problems or opportunities are present at the activity level. It is then up to the project manager to find out where these problems or opportunities occur in order to take the appropriate actions. However, false alarms should be excluded as much as possible since drilling down to search for problems or opportunities triggered by false warning signals results in a less efficient project

control approach. Setting these threshold values on the project level is therefore key to the success and efficiency of the top-down project control approach, and can be done as discussed along the two following sections.

### ***7.2.1 Management by Intuition***

Setting arbitrary control limits based on experience from previous projects allows the project manager to control the project in progress using the EVM performance metrics, but does not give the guarantee that real project problems (or opportunities) will be detected in a timely fashion. Indeed, controlling projects using EVM at the project level (or any other high WBS level) is not without any risk. Using the information of the EVM performance metrics measured at the project level might result in drawing biased conclusions and hence, wrong corrective actions can be taken. It is generally recognised that effects of nonperforming activities (delays) can be neutralised by well performing activities (ahead of schedule), which might result in masking potential problems. For this reason, [Book \(2006a,b\)](#), [Jacob \(2006\)](#) and [Jacob and Kane \(2004\)](#) criticised this top-down approach and argue that the EVM performance metrics are true indicators for project performance as long as they are used on the activity level, and not on higher WBS levels. However, other authors such as [Vanhoucke \(2010a\)](#) and [Lipke et al. \(2009\)](#) recognise this potential bias but argue that a top-down approach is the only reasonable approach that can be easily taken by practitioners. However, arbitrary control limits are dangerous and often lack precision and the ability to detect real problems (or opportunities).

### ***7.2.2 Management by Statistics***

Setting action thresholds by tolerance limits based on project specific data and based on user defined inputs on acceptable and unacceptable variation on the lowest WBS levels is known as *Statistical Project Control* (SPC). Unlike the intuitive approach, this statistical method should be able to better detect real deviations from the baseline schedule (both problems and opportunities) and should reduce the risk of false warning signals. Therefore, tolerance limits for the EVM performance metrics should be calculated. They act as real action thresholds such that exceeding these limits results in a WBS drilldown to search for real activity problems or opportunities. The Monte Carlo approach discussed in [Chap. 6](#) can be easily used to determine tolerance limits on the performance metrics. However, despite the similarity of using Monte Carlo simulation runs to generate random numbers, there is a fundamental difference between schedule risk analysis ([Chap. 6](#)) and setting tolerance limits on EVM key performance metrics (this chapter).

- **Inputs:** Monte Carlo simulations require distributions on activity (time and cost) estimates to enable the generation of numbers from these distributions to imitate activity variation. However, the meaning and interpretation of the activity distributions is different between the uncertainty generation used in SRA and the tolerance generation used in SPC. Indeed, while SRA relies on *risk profiles* to measure the expected variation in the activity estimates in order to test their impact on the project objectives, SPC relies on *acceptable variability tolerances* to define the distinction between acceptable variation and unacceptable variation in the activity estimates. The distinction between acceptable and unacceptable variation depends on the characteristics of the project and the experience and wishes of the project manager who is responsible for monitoring baseline scheduling deviations during project control.
- **Measurements:** Running a Monte-Carlo simulation is not fundamentally different for SRA and SPC, but the measurements taken during each simulation run are completely different. While an SRA focuses on the generation of sensitivity metrics as shown in Chap. 6 to measure the activity time and cost criticality, SPC focuses on the generation of project performance metrics such as the SPI, SPI(t) and CPI. These project performance metrics are generated for each simulation run and for all project periods and are saved on a stack for each percentage project completion between 0 and 100 %.
- **Outputs:** While SRA delivers sensitivity metrics to measure the probability of occurrence of risk or the expected impact of activity variation on the project objectives, SPC will create tolerance limits on the EVM performance metrics. All the SPC measurements collected during the simulation runs are used to calculate average expected values for the EVM performance metrics as well as their variability. These two statistics will then be used to define confidence intervals and to set tolerance limits that serve as action thresholds during project progress.

An extensive discussion of the simulation method and the SPC calculations is outside the scope of this chapter and can be found in [Colin and Vanhoucke \(2013b\)](#) for univariate statistical project control and in [Colin and Vanhoucke \(2013a\)](#) for multivariate statistical project control.

### 7.3 Bottom-Up Project Control

Schedule risk analysis provides a set of sensitivity metrics in order to predict the expected impact of unexpected changes on the project objectives. This knowledge is crucial to steer a project manager's attention towards a subset of the project activities that have a high expected impact on the overall project performance during project progress. The sensitive activities should be subject to intensive control, while other less sensitive activities require less or no attention during project execution. This approach is referred to as *bottom-up project control* to denote the upwards control



approach starting from information from the activities (at the lowest WBS levels) in order to make decisions during project progress that might have a positive impact on the overall project objectives (top WBS level).

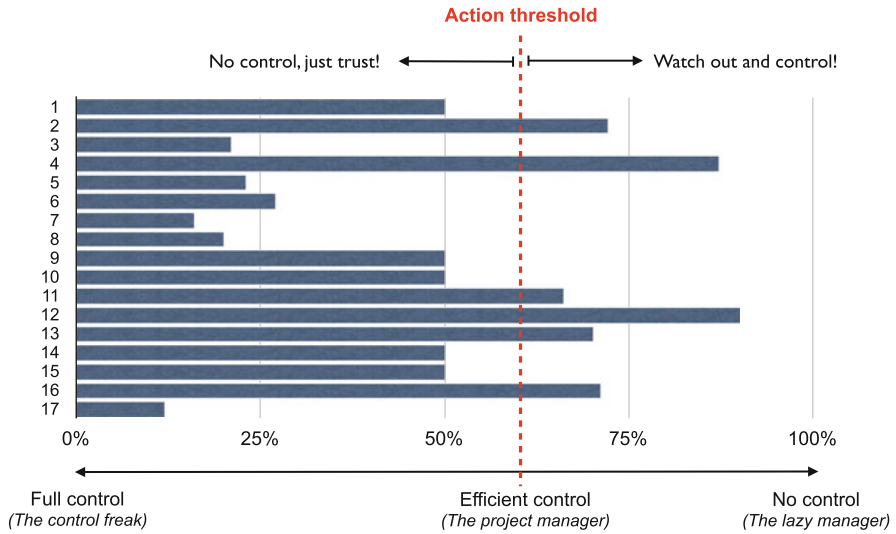
Ideally, the sensitivity metrics should be able to distinguish between risky and non-risky activities and should allow the project manager to reduce the control effort to only the risky activities, i.e. these activities with a high expected impact on the project time and cost objectives. Therefore, selecting the best performing sensitivity metric as well as setting a reasonable action threshold is key to the success and efficiency of the bottom-up project control approach.

### ***7.3.1 A Black-and-White View***

The critical path method (CPM) is probably the best-known method for bottom-up project control. Indeed, a straightforward way to control projects starting from activity information in order to take actions that have a beneficial impact on the project objectives, is to rely on the critical activities. CPM is the default method in baseline scheduling and defines the set of activities that are critical as the activities without slack. Any delay in these activities will have an immediate impact on the project duration. Using the CPM for bottom-up control has major disadvantages for at least two reasons. First, the time focus of the CPM ignores the cost objectives of the project as monitored by the CPI metric in the top-down project control approach. Secondly, the CPM assumes a black-and-white view on the criticality of project activities since activities are said to be critical or not. This narrow view on activity criticality leads to an extreme view on the importance of the various activities during project control: an activity deserves attention when it is critical or can be ignored when it is not. In reality, however, every non-critical activity has the potential to become critical and hence SRA can be considered as a refinement of the CPM, as previously discussed in Chap. 6. In the next section, the SRA method is used as an ideal tool for bottom-up control.

### ***7.3.2 Different Shades of Grey***

The extension of the CPM to SRA to overcome the narrow view on activity criticality and sensitivity was discussed earlier in Chap. 6. The availability of activity sensitivity metrics as a refinement of the critical path allows the project manager to refine his/her focus on the project in order to take appropriate corrective actions during bottom-up project control. Rather than using the critical/non-critical measure of the critical path, SRA provides degrees of criticality and therefore allows to set action thresholds that better distinguish between important and less important activities. The use of schedule risk analysis and its activity sensitivity measures to set action thresholds in bottom-up project control is discussed and illustrated using a



**Fig. 7.3** Action threshold for bottom-up project control

weighted combination of sensitivity metrics displayed in Fig. 7.3. The values of the time sensitivity metrics are copy-pasted from the graph displayed in the “Time risk analysis” pane of Fig. 6.2. This graph has been briefly discussed in the summary risk reports of the appendix of Chap. 6 and displays the average value of all indicators with a correlation with the SSI higher than 0.5.

This figure can be used in a dynamic bottom-up project control approach during project progress and shows how an action threshold can be set to trigger corrective actions in case of problems. This action threshold defines the degree of control, which can vary between no control (left of picture) and full control (right of picture), and is shown by the vertical dotted line on Fig. 7.3. All activities with a sensitivity value higher than or equal to this line are said to be highly sensitive activities that require attention during project progress and corrective actions in case of delays. In the example case of the figure, the action threshold has been set at 60% such that only the most sensitive activities 2, 4, 11, 12, 13 and 16 need to be considered during the control process. These highly sensitive activities are displayed in Fig. 7.4 at the bottom of the WBS and require full attention and action when necessary. All other activities are said to be insensitive and require less or no attention during project progress. Moving the action thresholds to the left or right increases and decreases the level of control, respectively, and possibly has an impact on the quality of the actions taken when problems are detected.

Hence, setting action thresholds is a user defined choice that depends on the desired effort the manager puts in controlling the project. The choice of the action thresholds depends on the management type as illustrated by the three types of project managers displayed at the bottom of Fig. 7.3. Each project manager

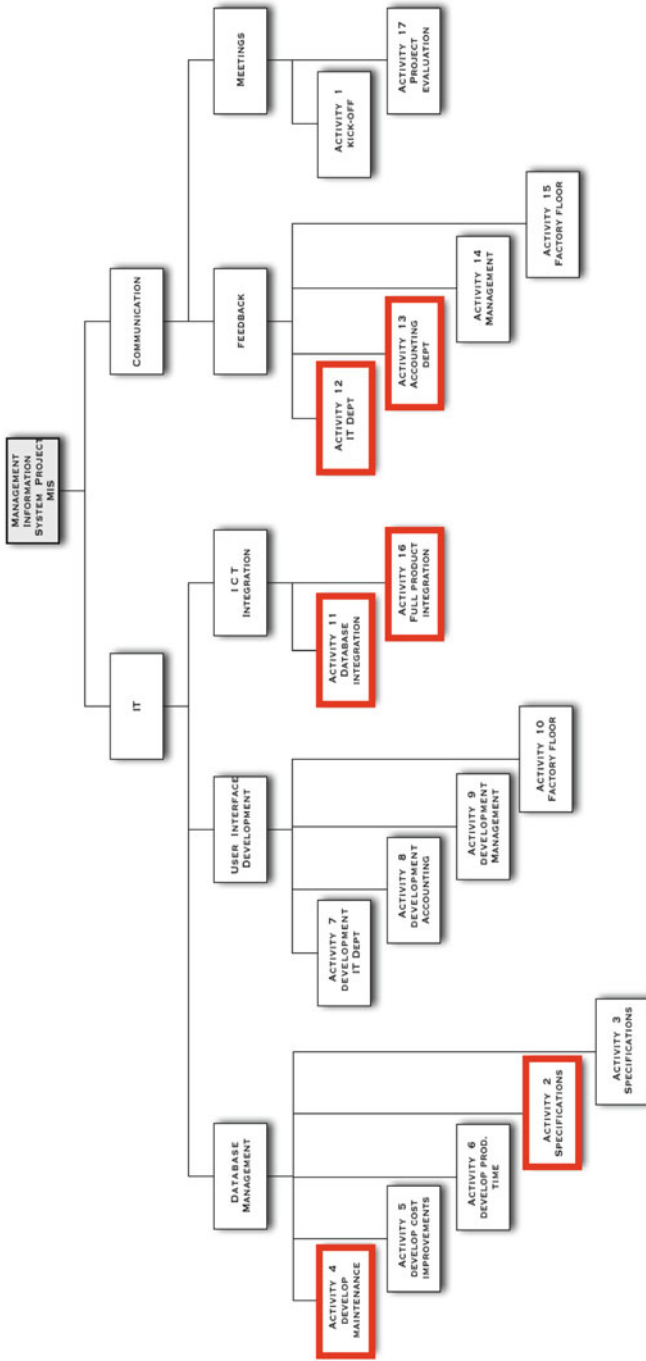


Fig. 7.4 The work breakdown structure for Project 1 (six activities under control)

wants to be successful when taking corrective actions in case of project problems. The difference lies in their desired level of control during project progress to detect activity problems, as discussed along the following lines.

- The control freak: A control freak puts the action threshold to the left of the picture and controls almost all activities along the project progress in order to avoid that problems are not timely detected. In this approach, the control effort is very high and the focus lies on detecting problems when they occur. Because of this high level of control, the probability that a problem will not be detected is very low and hence the control freak is someone who does not want to take the risk of missing any problem or opportunity.
- The lazy manager: The lazy manager is trustful and puts the action thresholds to the right in order to avoid too intensive control. Instead, only activities with a high expected impact will be controlled. In doing so, the lazy manager focuses on effort reduction and hopes/trusts that problems will be detected anyway, even with low levels of control. However, the risk that activity problems will remain undetected is quite high.
- The efficient project manager<sup>1</sup>: The efficient manager tries to balance between the level of control and the risk of not detecting problems and therefore carefully selects the action threshold somewhere between the control freak and the lazy manager. In this approach, the manager tries to reduce his/her effort of control in the hope that problems that will undoubtedly occur during project progress only have a significant impact on the project objectives for the activities that are under his/her control.

## 7.4 Control Efficiency

Both control approaches should be used in a project control setting to monitor the performance of the projects in progress and to take actions to bring projects back on track in case problems occur. Efficient project control is crucial to a project manager to guarantee project success. Efficiency of control is therefore defined as a combination of level of control as a proxy for the effort a manager puts in controlling projects, and the impact of corrective actions as a measure for timely detection of real activity problems. The higher the efficiency of control, the higher the expected return of actions taken with a reasonable level of control.

The level of control has been discussed in the previous sections and can be set based on the action threshold. It has been mentioned that the specific use of the thresholds depends on the used control approach. For the bottom-up control approach, only highly sensitive activities are controlled, and hence, action thresholds are set on the values for the sensitivity metrics. As an example, from the moment the SSI value is higher than 70 %, the activity is said to be highly

---

<sup>1</sup>Rumour has it that this is a project manager who uses ProTrack to manage his/her projects ©.

sensitive and it is expected that delays on this activity might have a significant impact on the total project duration. Therefore, it is better to carefully control this activity when it is in progress. Activities with a low SSI value, on the contrary, are considered to be safe and need no control during progress. The top-down project control approach is done using schedule performance information at regular points in time, given by the CPI, SPI and SPI(t). From the moment these values drop below a certain predefined threshold, say e.g. 70 %, it is an indication that some of the underlying activities at the lowest WBS level might be in danger. Therefore, the project manager has to drill down, trying to detect the problem and find out whether corrective actions are necessary to improve the current low performance.

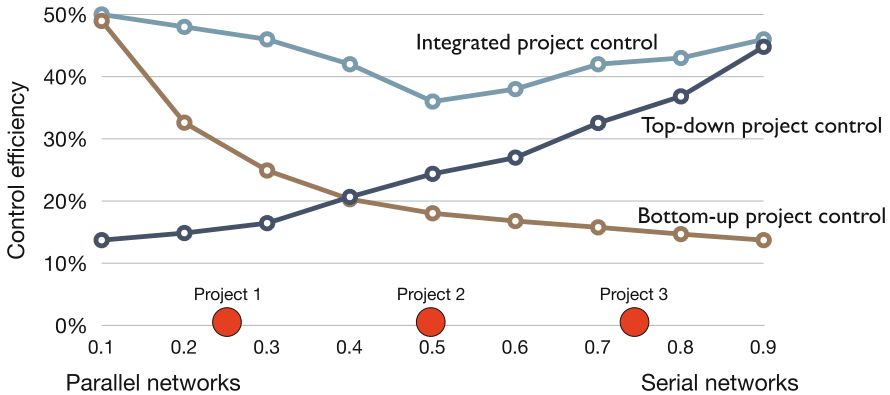
The two alternative control methods show one important difference: in the top-down approach, all EVM performance metrics are calculated at each time period, and only when thresholds are exceeded, a drilldown requires further attention in search for potential problems that might result in corrective actions. For a bottom-up approach, a subset of activities in progress, determined by the thresholds, is subject to control and might need further action in case of problems. Consequently, the selection of the set of activities that require intensive control in search of potential problems and the corresponding actions to bring problems back on track is different for the two control methods, and goes as follows:

- Top-down: At every time period, all EVM performance metrics are calculated and when thresholds are exceeded, all activities in progress will be scanned in search for potential problems (and corresponding actions). Consequently, the search is triggered by EVM metrics and once triggered, is performed on all activities in progress.
- Bottom-up: At every time period, all SRA sensitivity metric values are known for each activity as a result of the Monte-Carlo simulation runs. Only a subset of these activities, namely those that exceed the thresholds, will be further analysed in search for problems (and corresponding actions). Consequently, the search is triggered by SRA sensitivity metrics and performed on a subset of activities in progress.

Figure 7.5 shows an illustrative graph of this control efficiency for both control approaches. This graph is the result of a large simulation study published in Vanhoucke (2010a) and only evaluates the time control efficiency. To the best of my knowledge, no such results are available for cost control. The x-axis displays the closeness of each project to a completely serial or parallel network, as measured by the SP indicator discussed in Chap. 1. The dots on the x-axis display the SP values for the three example projects used throughout the book. The y-axis measures the control efficiency as follows:

$$\text{Efficiency} = \text{Return}/\text{Effort}$$

- The *effort* is measured by the number of control points which is equal to the number of times the action thresholds are exceeded. Indeed, from the moment a threshold is exceeded, the project manager must spend time to find out whether



**Fig. 7.5** The control efficiency of a bottom-up and top-down control approach

there is a problem during progress. Hence, the number of control points is used as a proxy for the effort of control and depends on the value of the actions thresholds. Obviously, the lower the effort, the higher the control efficiency and hence the effort is set in the denominator of the control efficiency output metric.

- When corrective actions are taken, their impact should bring endangered projects back on track and should therefore contribute to the overall success of the project. Therefore, the *return* of the actions is measured as the difference between the project delay without actions and the project delay with actions. Obviously, the return of the actions can be considered as a proxy for the quality of the actions and should be set in the numerator of the project control efficiency output calculation.

The graph clearly demonstrates that a top-down project-based control approach using the EVM/ES systems provides highly accurate results when the project network contains more serial activities. The bottom-up activity-based control approach using sensitivity information of activities obtained by an SRA is particularly useful when projects contain a lot of parallel activities. This bottom-up approach requires subjective estimates of probability distributions to define the activity risk profiles, but simplifies the control effort by focusing on those activities with a highly expected effect on the overall project objective. An integrated project management and control approach mixing a bottom-up and a top-down control approach using sensitivity metrics as well as EVM/ES metrics leads to an increase in the control efficiency, as displayed by the top line in Fig. 7.5.

## 7.5 Conclusions

In this chapter, all metrics presented in the previous chapters are brought together in two alternative project control methods that aim at measuring the performance of projects in progress and generating triggers for corrective actions to bring projects in

trouble back on track or to timely exploit project opportunities. The two alternative control methods differ in the fact that their measurement points are taken at different levels of the Work Breakdown Structure (WBS), as summarised along the following lines:

- Bottom-up project control starts at the lowest levels of the WBS and relies on the sensitivity metric values of a schedule risk analysis discussed in Chap. 6 to determine which of the project activities must be subject to intensive control.
- Top-down project control relies on the performance measures given by the EVM/ES systems discussed in Chaps. 2–5 that act as detectors of project problems at lower levels of the WBS.

In order to be able to efficiently control projects in progress, these two alternative control methods should be integrated.<sup>2</sup> The efficiency of project control is measured by two parameters, as follows:

- The *effort* of control is determined by the number of times the project manager has to control the project. This number of control points is based on action thresholds set on the SRA or EVM/ES metrics for the bottom-up and top-down project control approaches, respectively. These action thresholds can be set manually based on best guesses or experience or can be determined by statistical methods in order to guarantee project success.
- The *return* is measured by the quality of the corrective actions and triggered by the control metrics of the two alternative control approaches. The return is defined as the reduction in time or cost when actions are taken for projects in problems.

In this chapter, it has been shown that the two alternative control methods have a different behaviour for time control and that their efficiency depends on the seriality of the project network. More precisely, for projects closely resembling a parallel network, the bottom-up approach works best, while for projects with a more serial structure, the EVM/ES metrics used in a top-down approach lead to much better efficiency results. The closeness of a project to a serial or parallel structure can be easily measured by the SP indicator of Chap. 1. No similar efficiency results have been reported for cost performance and efficiency. However, it is conjectured that the SP has no influence on the efficiency of the two alternative control methods for cost control.

---

<sup>2</sup>The reader finally knows why I have chosen to entitle the book “Integrated Project Management and Control”.

**Part II**  
**Then Comes the Practice**



# Chapter 8

## ProTrack

ProTrack is an acronym for *Project Tracking*, which is a synonym for project control. It has been developed as an integrated project management and control software tool based on research done at the Operations Research and Scheduling (OR&S) research group of Ghent University. The integration between baseline scheduling and risk analysis to facilitate and improve project control lies at the heart of ProTrack. The integrated project management and control theme of this book has been largely inspired by numerous research and consultancy projects that have finally resulted in the development of the first ProTrack version. Ever since, new versions with extended features have been released for which a brief summary will be given in this chapter. The outline of this chapter can be summarised along the following lines. In Sect. 8.1, the integrated project management and control approach is briefly explained and illustrated using the terminology and features of ProTrack. Section 8.2 describes the use of ProTrack to evaluate the characteristics of a project. Section 8.3 briefly describes how ProTrack has been used for the classification of real project data that can and will be used in future research projects. Finally, Sect. 8.4 gives a summary of the main topics of this chapter and draws general conclusions. A lot of the work presented in this chapter is available elsewhere in promotional material or other books and articles discussing the use of ProTrack for project management and control. A detailed list of features of the latest ProTrack version is available via the free online tutorials at [www.protrack.be](http://www.protrack.be).

### 8.1 Dynamic Scheduling on Your Desktop

ProTrack 4.0 is an extended version of ProTrack 3.0. Its integration with PM Knowledge Center and its strong focus on the integration of baseline scheduling, risk analysis and project control makes it yet a stronger learning tool to stimulate interaction between researchers, students and practitioners in the field of project management and dynamic scheduling. New to ProTrack 4.0 is the inclusion of statistical project control as well as improved export functionalities. ProTrack is

useful for constructing baseline schedules, conducting risk analyses and composing control reports for complex projects with limited resource availabilities, and restricted time and budget constraints. ProTrack puts a strong focus on the integration between the three integrated project management and control dimensions used throughout the various chapters of this book:

- **Baseline Scheduling:** Schedule projects using critical path and resource allocation algorithms. The construction of a baseline schedule received little attention in this book, except for straightforward critical path calculations presented in Chap. 1. For an extended overview on the construction of a baseline schedule, the reader is referred to [Vanhoucke \(2012c\)](#).
- **Risk Analysis:** Analyse project risk using basic or advanced Monte Carlo simulation runs, as discussed in Chap. 6.
- **Project Control:** Generate project performance data and analyse EVM/ES control data, as discussed in Chaps. 2–5 and in Chap. 7.

The fourth release of ProTrack contains a new online support and help center and is now used in both academic and business trainings. ProTrack's algorithms are based on award winning research published in international books and a free online learning tool PM Knowledge Center. The aim of PM Knowledge Center ([www.pmknowledgecenter.com](http://www.pmknowledgecenter.com)) is to bring relevant articles and references on dynamic scheduling together, and its integration with ProTrack helps students and practitioners to better understand the underlying techniques. ProTrack's help and support center allows the user to interact with the OR-AS team for useful suggestions and comments, to report problems and to propose and share future development ideas. A complete tutorial of ProTrack is available since the second release of ProTrack in [Vanhoucke and Van Acker \(2010\)](#). Despite the complete redesign of ProTrack 3.0 compared to its previous version and the extended features of ProTrack 4.0, the tutorial is still relevant to get familiar with the functionality of ProTrack. Updates of tutorials will be posted at the bookstore of the OR-AS website ([www.or-as.be](http://www.or-as.be)).

In the remaining paragraphs, each of the three integrated project management and control dimensions will be briefly explained without mentioning the full functionality of ProTrack. Each paragraph will also make a brief reference to more advanced features that have been added to ProTrack as an immediate result of research projects done in the past.

### ***8.1.1 Baseline Scheduling***

In Chap. 1 of this book, the three baseline schedules used throughout most chapters have been constructed by simple earliest start calculations defined by the critical path method. However, the construction of a baseline schedule for real projects is not always straightforward, and often is a time-consuming and cumbersome task. The incorporation of renewable and non-renewable (or consumable) resources

increases the complexity of the baseline scheduling phase, and therefore requires some basic knowledge of the available tools and techniques as well as insight in the way these schedules are constructed. Nevertheless, the baseline schedule plays a central role in an integrated project management and control setting, both for analysing the schedule risk (Chap. 6) and for the EVM calculations (Chaps. 2–5). This baseline schedule acts as a point of reference in the project life cycle, and hence, a project schedule should be considered as nothing more than a predictive model that can be used for resource efficiency calculations, time and cost risk analyses, project tracking and performance measurements, and so on. Hence, care must be taken to construct a resource feasible and realistic schedule that meets all requirements of the project manager and the members of his/her team. No special attention is given in this book on how to construct a baseline schedule, and the reader is referred to the book of Vanhoucke (2012c) and the summary article published in Vanhoucke (2013f) for more advanced information. An example of recent experiences of baseline scheduling is given in Vanhoucke (2013e).

Since the construction of a baseline schedule is key in an integrated project management and control setting, many of the algorithms and procedures incorporated in ProTrack are the result of years of research, for which a summary is given in Vanhoucke (2013g).<sup>1</sup> Prior to the construction of a baseline schedule, the ProTrack user has to add project data, including a project network with activities and precedence relations, possibly extended with activity constraints, resource data and much more. Since this manual data input process is often a time-consuming task and because ProTrack is widely used in training sessions, ProTrack can automatically generate project and resource data based on a well-designed generation approach proposed in Demeulemeester et al. (2003) and Vanhoucke et al. (2008). The generation process design includes project network topology as well as resource indicators, such as the Serial/Parallel (SP) indicator proposed in Chap. 1. Once the data are available, the construction of the baseline schedule can be done by ignoring resource constraints, as is done in the traditional *critical path method* or by leveling resources to minimise the total project duration, as proposed by the well-known *resource-constrained project scheduling methods* available in literature.

Figure 8.1 displays the baseline schedule of Project 1 as an earliest start schedule Gantt chart, and is similar to the Gantt chart displayed in Fig. 1.2. The graph at the bottom of the picture displays the Planned Value curve of Fig. 2.2 which is known from the moment the baseline schedule is saved.

### 8.1.2 Schedule Risk Analysis

The Schedule Risk Analysis technique discussed in Chap. 6 connects the risk information of project activities to the baseline schedule and provides sensitivity

---

<sup>1</sup>This free online article will be updated regularly when new research results become available.

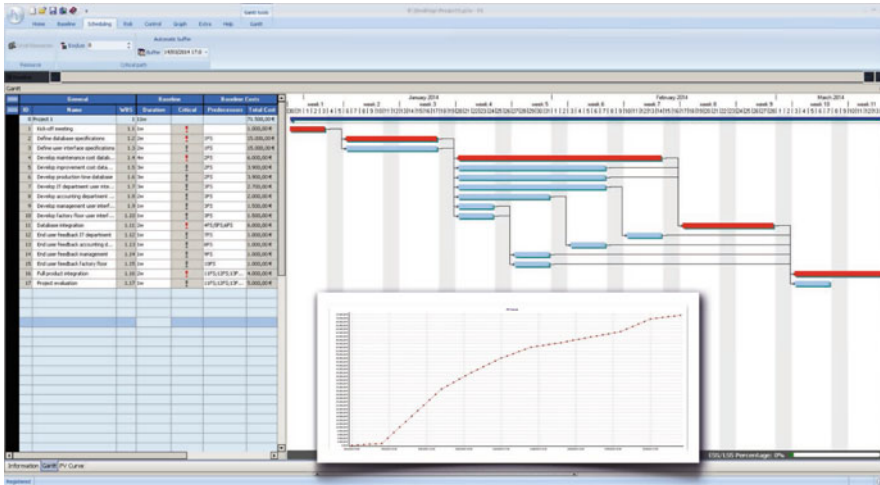


Fig. 8.1 A Gantt chart to display the baseline schedule of Project 1 in ProTrack

information of individual project activities as a way to assess the potential impact of uncertainty on the final project duration and cost. ProTrack’s schedule risk analysis makes use of two simulation engines that are hidden behind the simulation screen of the software tool. The standard simulation engine allows the user to start a quick and easy simulation without a thorough study of the risk profiles of the various project activities. This standard engine does not require data about estimated probability distributions for the activity durations, but instead makes use of nine predefined simulation scenarios. The advanced simulation engine requires a more detailed risk quantification for each project activity using predefined or user-specific activity duration distributions.

ProTrack is able to generate sensitivity measures and reports for the duration and cost of each activity as well as for the resources linked to these activities (note that resources have not been used in this book). Moreover, the two simulation engines can also be used to assess the quality of the EVM forecasts as measured by the MAPE and MPE measures discussed in Chap. 5. The integration between an SRA and a MAPE/MPE accuracy study is highly relevant for understanding the relevance of project control and illustrates the deep integration between scheduling, risk analysis and control in ProTrack, which is the central theme of this book.

Although Schedule Risk Analysis has been known for years in the academic literature (Hulett 1996), many of the ProTrack features are unique and based on sound and academic research. As an example, the basic simulation engine consists of nine well-designed scenarios, originally presented in Vanhoucke and Vandevorode (2007b) and used in several studies in Vanhoucke (2010a). A validation of the sensitivity metrics presented in Chap. 6 has been done in Vanhoucke (2010c). The bottom-up project control approach discussed in Chap. 7 has been initially proposed in Vanhoucke (2011) and used in Vanhoucke (2012a).

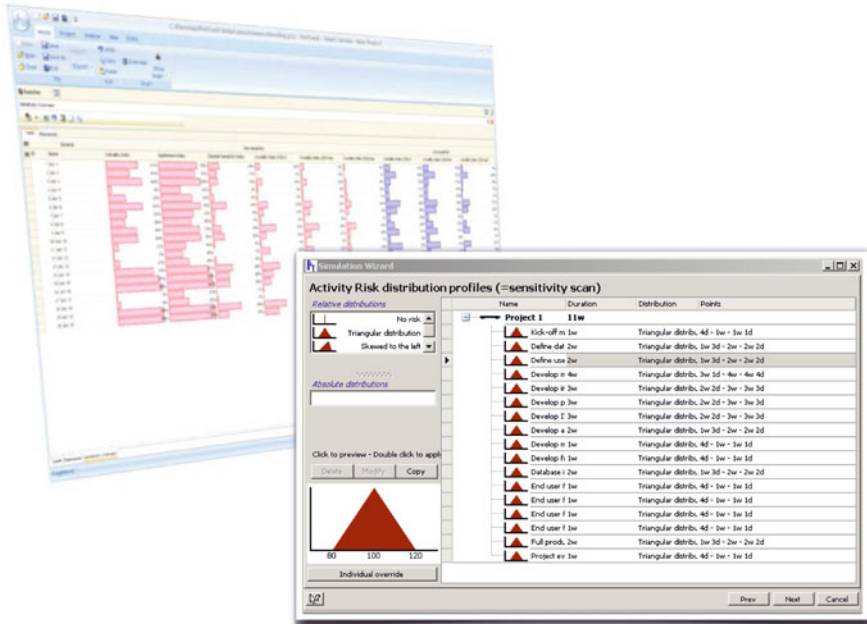


Fig. 8.2 Schedule risk analysis screens of Project 1 in ProTrack

An overview of simulation studies used in project control is given by Vanhoucke (2013d).

Figure 8.2 displays the output graphs (top left) as a result of the Monte Carlo simulations that define the inputs for a Schedule Risk Analysis. Sensitivity metrics for time and cost are available, based on user-defined distributions of the activity durations, as shown in the bottom right picture.

### 8.1.3 Project Control

Project control is the process performed to observe project execution in order to identify potential problems and/or opportunities in a timely manner such that corrective actions can be taken when necessary. The requisite is that the current project status is observed on a regular basis, enabling the calculation of the project performance variance that is equal to the gap between actual performance and the baseline schedule. Since the current project performance is measured by deviations from the project management plan, the baseline schedule plays a central and unambiguous role during the project tracking process. ProTrack employs a strict definition of a unique baseline schedule for each tracking period, such that it is always clear during the complete project execution what the active baseline

schedule is. ProTrack allows the user to define multiple tracking periods in order to get a clear view of the progress of the project over time. Each tracking period is linked to a unique status date and a corresponding baseline schedule, and measures the current progress up to the status date of the project compared to the active baseline schedule. Both the top-down and bottom-up control approaches discussed in Chap. 7 lie at the heart of the control engine of ProTrack. Unique in ProTrack is the integration between EVM and ES, including concepts such as the p-factor (Chap. 4). As mentioned in Sect. 7.2, more advanced statistical project control concepts using tolerance limits also belong to the standard control options in ProTrack.

Controlling a project can only be done in an efficient way when the construction of a baseline schedule and the analysis of its risk is deeply integrated with the techniques available for project control. ProTrack is an ideal tool for project management and control thanks to this integration, and allows an easy comparison between the periodic control data and the available schedule and risk data. However, controlling a project in progress requires intensive data collection, which is not always desirable, for instance when ProTrack is used in teaching or training sessions. Therefore, the automatic generation of project control periods, similar to the generation of project and resource data discussed earlier, is a strong tool that can be used to facilitate the teaching sessions and to avoid the periodic data collection. It allows the user to see all the formulas presented in the previous chapters on their screen for their project in progress. Moreover, most, if not all, of the formulas and techniques embedded in ProTrack are validated based on a sound and detailed research methodology. The three time forecasting methods (Planned Value, Earned Duration and Earned Schedule, see Chap. 5) have been compared in a study by Vandevoorde and Vanhoucke (2006) and tested and validated in Vanhoucke and Vandevoorde (2007b). A comparison between top-down and bottom-up control approaches discussed in Chap. 7 is done in Vanhoucke (2011), and the incorporation of more advanced statistical techniques is based on the studies by Colin and Vanhoucke (2013b) and Colin and Vanhoucke (2013a). Most of the work embedded in ProTrack comes from the research study published in Vanhoucke (2010a). Further details have been published elsewhere (Vanhoucke 2008, 2009, 2010b, 2013d; Vanhoucke and Shtub 2011; Vanhoucke and Vandevoorde 2007a), showing links with the network topology (Vanhoucke and Vandevoorde 2009), activity criticality (Vanhoucke and Vandevoorde 2008) and schedule adherence (Vanhoucke 2013b,c).

Figure 8.3 displays the artificial project progress of Project 1 with two graphs indicating the project progress performance.<sup>2</sup> The picture on the left shows the three key metrics, while the picture on the right displays the two alternative schedule performance indices.

---

<sup>2</sup>Note that the project progress differs from the progress used in the previous chapters. This picture is only for illustrative purposes.

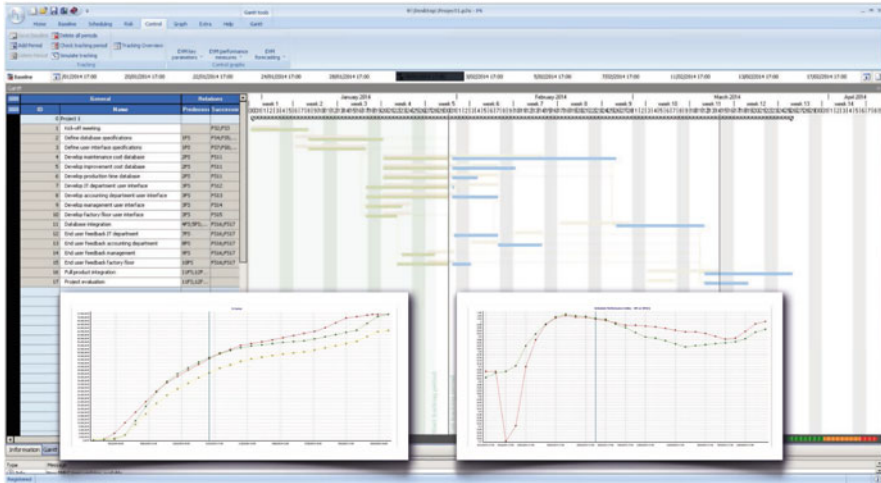


Fig. 8.3 Project control of Project 1 in ProTrack

## 8.2 Project Scan

Visualisation of the project performance at a certain moment in time is key to communicate and discuss the current project problems or opportunities with the project manager’s team, the project stakeholders and the supervising board. ProTrack is an ideal tool to visualise the static project characteristics as well as the dynamic periodic time and cost performances, the project forecasts, the risk metrics and much more. The reports shown in the appendices of Chaps. 2, 3, 5 and 6 are illustrative reports but the look and feel of a progress report obviously depends on the wishes and the needs of the audience. It should be noted that the reports of this book display the information at the end of the project, and assume that all values for the EVM and SRA metrics are known. Obviously, intermediate reports are much more valuable since they can be used to assess the current project performance and to take actions when needed.

A project scan is defined as a simple yet effective way to assess the static (i.e. before the project start) and/or dynamic (i.e. during project progress) state of the project in order to analyse and discuss the current state, report summary conclusions to the targeted audience and find improvements that could facilitate the management of projects in progress which, ultimately, should contribute to the final success. In this book, summary reports have been shown containing graphs accompanied by a short and to-the-point conclusion, for the following themes:

- EVM performance report: The time and cost performance over the complete lifespan of the project using EVM metrics.
- ES performance report: The time performance over the complete project lifespan using ES metrics.

- EVM/ES forecasting report: Time and cost forecasts including accuracy and stability measures.
- Schedule risk report: Time and cost sensitivity metrics to analyse and assess the project risk.

Visualisation of progress reports is of course of little use without interpretation. Therefore, the central idea of a detailed project scan is not only to visualise the progress but rather to understand the underlying characteristics and to map them with the best practices and research knowledge to better control your projects and improve your actions to bring projects in danger back on track. It is based on years of research and aims at benchmarking a project along the four following dimensions:

- Network scan. The characteristics of your project WBS in terms of network topology and time and cost distributions are a crucial factor in the understanding and selection of the best project control method.
- Resource scan. The efficiency of your resource allocations depends on the availability and tightness of your resources and has an impact on the schedule risk analyses and project control methods.
- Sensitivity scan. A Schedule Risk Analysis (SRA) using basic and/or advanced Monte Carlo simulation runs is a crucial analysis to validate the efficiency of a bottom-up control approach for your project.
- Control scan. A project control scan using simulated time/cost accuracy predictions is necessary for understanding the efficiency of a top-down control approach using Earned Value Management/Earned Schedule (EVM/ES) performance systems.

Although ProTrack can be easily used to analyse the properties and characteristics of a project, its focus lies on the daily use of monitoring the performance of the project and not on the intensive calculations that are required for a full project scan. Therefore, P2 Engine has been developed as a fast and efficient project calculation tool to scan the characteristics of the network and the resources, as well as to predict the accuracy of the risk and control techniques. This tool will be discussed in Chap. 9.

### 8.3 Project Classification

Most of the research work that has led to the development of ProTrack has been done on fictitious project data, generated under a well-established design. However, research results have little or no use if they cannot be translated into practical guidelines and rules-of-thumb that are of interest to the professional project manager. Therefore, validation of newly developed tools and techniques on empirical data is necessary to lift up the current knowledge on project management and control. However, it is of the author's personal belief that the empirical validation on real project data must come after the well-designed tests on fictitious projects.



Therefore, any research project carried out at the OR&S group at Ghent University will put forward a well-considered and balanced view between theoretical fictitious project data and empirical real project data, with the main research focus on fictitious project data based on a controlled and full-factorial design. In doing so, the researchers have full control of the project parameters in order to obtain and present general results that are applicable to a wide variety of projects. Afterwards, these general results can be translated into practical guidelines and rules-of-thumb that differ from project to project, company to company and sector to sector. Only at this final step can the theoretical results be validated on a diverse set of empirical project data that will be obtained through the collaboration with the companies and professionals working together with the OR&S group and OR-AS, and are then ready to be implemented in ProTrack.

These collaborations have currently led to two publications in the academic literature. [Vanhoucke \(2012b\)](#) summarises the main conclusions of various experiments performed in a large simulation study on the efficiency of project control techniques and the ability to trigger corrective actions in case of project problems. The main purpose of these simulation experiments is to understand why EVM and/or SRA work so well for some projects and fail so miserably for others. The research has therefore been carried out on fictitious projects. However, in order to illustrate its relevance and use in practice, the paper also compares the results obtained on fictitious project data with additional tests performed on a set of real-life data from eight Belgian companies from various sectors. In [Batselier and Vanhoucke \(2013\)](#), more than 40 projects from various sectors have been analysed and classified. The database of projects is used to validate the research on the stability and accuracy and other project control related topics found in various research projects on empirical data. This database is publically available at [www.or-as.be/ipmc](http://www.or-as.be/ipmc) and will continuously grow over the coming years. Unique to this research and database is the way the project data are structured in so-called project cards. A card is made for each project in the database and contains all relevant information on the three integrated dimensions discussed earlier (scheduling, risk and control). Moreover, the authenticity is also labelled as a way to assess how realistic the data are. Scheduling, risk and control authenticity are measured along various dimensions.

## 8.4 Conclusions

This chapter briefly reviews the main functionality of ProTrack, an integrated project management and control software system developed by OR-AS. ProTrack is used in various trainings on integrated project management and control. Thanks to its full integration with PM Knowledge Center, it should lead to a better understanding of how to manage and control projects. ProTrack is developed by and for academics and professionals who share a common passion to get complex projects in progress back in control. It is relevant to project managers working in both the private and public sector, and is applicable to large and small projects

with critical performance, time and budget targets. Furthermore, it is useful for constructing baseline schedules, risk analyses and control reports for easy and complex projects with limited resource availabilities, and restricted time and budget constraints. The integration between these three project management disciplines is known as dynamic scheduling and has a central focus in all the activities of OR-AS.

ProTrack puts a strong focus on the integration between the three dynamic scheduling dimensions:

- **Baseline Scheduling:** Critical path and resource scheduling
- **Risk Analysis:** Schedule risk analysis and Monte Carlo simulations
- **Project Control:** Earned Value Management and Earned Schedule control

Thanks to the use of ProTrack at university programs, the creation of a project database by selecting the best case studies received by the OR&S group at Ghent University was possible. This library of projects is available online and each project is accompanied by a project card that enables the user to learn about the various topics relevant for integrated project management and control. This set of projects is used in trainings and, more importantly, in research projects. The reader is therefore stimulated to send his/her project data to OR-AS to contribute to the growth of the project database and to the research activities on integrated project management and control.

# Chapter 9

## P2 Engine

P2 Engine is an integrated project management and control utility tool developed by OR-AS. It is a command-line based application containing all algorithms of ProTrack that can be called using the LUA scripting language in order to generate gigabytes of project data. It generates project baseline scheduling data and risk analysis metrics as well as dynamic project progress data that can be used for testing and validating novel research ideas. P2 Engine is mainly used as a research tool for master students and PhD students at universities, but also acts as a support tool in consultancy projects to advance current business processes with extended dynamic scheduling features.

In this chapter, an overview is given of the main classes of P2 Engine and their basic functionalities are briefly described. A detailed explanation of all functions of each class is outside the scope of this chapter, and can be found at [www.p2engine.com](http://www.p2engine.com). The outline of this chapter can be summarised along the following lines. Section 9.1 gives an overview of the seven classes implemented in P2 Engine. In Sect. 9.2 it is shown how project data can be analysed by P2 Engine to generate the integrated project management and control reports shown in previous chapters. The LUA code for generating all chapter reports is also given in this chapter. Section 9.3 gives a summary of the main functionalities of P2 Engine.

### 9.1 Advancing the State of the Art

P2 Engine is a command line utility tool based on the LUA scripting language to generate gigabytes of project data. It gives the user access to a wide range of complex algorithms incorporated in ProTrack. By using simple scripts, any researcher can solve difficult and critical dynamic project scheduling optimisation problems using these intelligent algorithms. P2 Engine can easily produce an enormous database of optimisation results for a wide range of project management problems, faster than ever before and advances the state-of-the-art knowledge

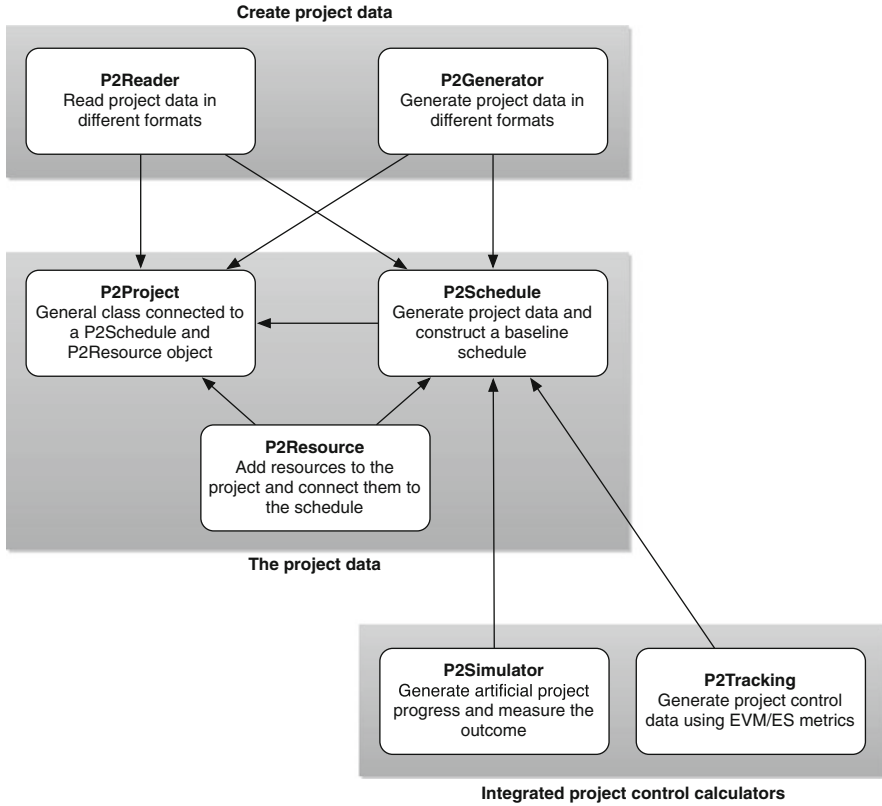


Fig. 9.1 The logic between the seven classes of P2 Engine

available today. It generates project baseline scheduling data and risk analysis metrics as well as dynamic project progress data that can be used for testing and validating novel research ideas. P2 Engine is a platform independent software tool that runs on Windows, Mac as well as on Linux. It also runs on the supercomputer of the Flemish Supercomputer Centre (VSC). This cluster of computers is the largest academic computing infrastructure in Flanders. P2 Engine can be easily accessed using the LUA scripting language available at [www.lua.org](http://www.lua.org). In the remaining subsections, the different classes of P2 Engine will be briefly discussed. This chapter has no intention to give a full overview of all technical details, but instead, provides a short summary of all currently implemented features of P2 Engine. Details on all functions can be found at [www.p2engine.com](http://www.p2engine.com) (Fig. 9.1).

### 9.1.1 Generating Data

Project data can be generated in two ways. Project data can be read from a previously saved data file or can be automatically generated using network and resource

parameters to characterise the structure of the data. The project data presented in Chap. 1, and used in almost all chapters of this book, can be easily generated using short and simple LUA scripts as shown in Sect. 9.2.

## **P2Reader**

The P2Reader class is used to read project data files in pre-defined formats that are well-known in the academic literature. These formats are mainly used for academic research and do not contain all relevant data as the project data files used in ProTrack. The project data are saved and restored in the P2Reader class in the following formats:

- **Patterson format:** The Patterson format contains data about the project network with the presence of renewable resources under a fixed resource availability.
- **PSPLib format:** The PSPLib format is similar to the Patterson format, but adds some extra information. Information can be found in [Kolisch and Sprecher \(1996\)](#).
- **P2 Engine format:** The P2 Engine format is written to exchange data between P2 Engine and ProTrack and contains both network and resource information (similar to the Patterson and PSPLib formats), as well as risk and project control data.

The details of each format are not within the scope of this book, and can be found at the P2Reader tab on the P2 Engine website. Obviously, any other user-defined data format can easily be implemented by calling all values of the parameters relevant for the project using the functions of the P2 Engine classes and defining an output printing function in LUA.

## **P2Generator**

The P2Generator class is used to generate activity-on-the-node project networks with activities, precedence relations and resource constraints. Both the topological network structure and the resource scarceness can be controlled using pre-defined parameters, as summarised along the following lines:

- **Activity parameters:** Values for activity durations and (fixed and variable) costs, as well as constraints on these activities, such as ready times or due dates can be generated under a strict design defined by the generation parameters.
- **Network parameters:** The network topology of a generated project can be controlled by setting a value for the SP indicator discussed in Chap. 1. It defines the number and structure of the precedence relations between project activities.
- **Resource parameters:** The presence of renewable and consumable resources can be generated under a strict design, as defined by the parameters to generate both the availability as well as the demand of resources linked to the activities.

The data generation process is based on a sound and well-developed algorithmic generation process originally proposed in the random network generator RanGen1 (Demeulemeester et al. 2003) and further developed in RanGen2 (Vanhoucke et al. 2008).

### **9.1.2 Project Characteristics**

The definition of a project and the construction of a schedule with or without taking resources into account can be done by making use of three classes, that will be briefly described along the following subsections. The integrative use of these three classes allows the user to collect and save project data, as well as to construct baseline schedules using simple or more advanced scheduling methods.

#### **P2Project**

The P2Project class is a general class to define a project. It does not contain a lot of functionality but creating an object from this class is necessary to work with P2 Engine due to the links with the P2Schedule and P2Resource classes.

#### **P2Schedule**

The P2Schedule class is used to define an activity-on-the-node network with activities and precedence relations and to construct a project schedule using traditional critical path scheduling techniques. Extensions to resource scheduling are possible using the P2Resource class. The different functions of the P2Schedule class can be classified as follows:

- Activity data: Data about activity durations and costs can be set in the P2Schedule object.
- Activity constraints: Setting time constraints for activities like ready times, due dates or fixed time slots enables the user to control the construction of the baseline schedule.
- Network information: The use of precedence relations with the four well-known time-lags can be set.
- Resource information: Connecting the resources of the P2Resource class to the project activities in order to quantify the demand for resources for each activity can be done in the P2Schedule class.
- Scheduling information: Constructing project baseline schedules can be done using the simple and easy critical path method (CPM), or using more advanced resource scheduling and levelling algorithms under various settings and assumptions.

The construction of a baseline schedule is not discussed in detail throughout this book. Only in Chap. 1, the simple critical path based earliest start schedule has been presented, and no links to resource-constrained project scheduling and levelling procedures have been made. However, constructing baseline schedules in the presence of limited resource availability does not fundamentally change the concepts discussed in the previous chapters. More information on resource-constrained project scheduling can be found in [Vanhoucke \(2012c\)](#).

## **P2Resource**

The P2Resource class is used to add renewable and consumable resources to a project and to link them to the project activities. The different functions of the P2Resource class can be classified as follows:

- General resource information: Adding resources to a project and defining the resource type can be done in the P2Resource class.
- Resource availability and demand: Resources have a variable availability through time and can be defined in the P2Resource class. The demand for resources by the project activities can be set at the P2Schedule class.
- Resource cost: The cost of resources as a combination between fixed/variable and per-use and per-unit cost can be defined in the P2Resource class.

### ***9.1.3 Integrated Project Management and Control***

The two remaining classes are the heart and soul of integrated project management and control and focus on the use of EVM/ES systems as well as SRA studies to perform top-down and/or bottom-up control as explained in Chap. 7.

## **P2Simulator**

The P2Simulator class is used to run Monte Carlo simulations on a project as presented in Chap. 6. However, these simulation runs can be used for various purposes, and are therefore not restricted to a schedule risk analysis. The simulations can be used for project control to generate periodic EVM/ES reports as shown in the previous chapters, including forecasting reports, schedule adherence metrics and much more. The ability to imitate artificial project progress marks P2 Engine as the ideal research tool to test novel ideas. The different functions of the P2Simulator class can be classified as follows:

- Simulation settings: Defining the number of runs, the use of distributions and more advanced settings.
- SRA simulation: The generation of all time and cost sensitivity measures presented in Chap. 6.

- EVM/ES simulation: All EVM/ES metrics presented in Chaps. 2–5 can be simulated by imitating artificial project progress.
- SPC simulation: The automatic action thresholds using statistical project control, as briefly discussed in Chap. 7, can be generated by means of simulation runs.

## P2Tracking

The P2Tracking class is used to construct periodic project performance reports of projects in progress. These reports contain current project performance measures, schedule and cost metrics and EVM forecasting reports as well as ways to update the remaining work of the project to better respond to future project needs. The EVM/ES metrics presented in Chaps. 2–5 and in Chap. 7 can be used in this class. The different functions of the P2Tracking class can be used to periodically update the data of the project in progress and can be classified as follows:

- Control input data: Adding periodic control data can be done using artificial project progress by the P2Simulator class or by manual input using the P2Tracking class.
- EVM/ES output data: All EVM/ES metrics presented earlier, including the p-factor and all forecasting methods, can be generated periodically in the P2Tracking class.

## 9.2 Chapter Reports in LUA

In this section, example LUA scripts are displayed and discussed that have been used for all calculations to generate the reports in the previous chapters. Each script contains a basic functionality and can be extended to more advanced P2 Engine scripts. The scripts can be freely downloaded from [www.p2engine.com/examples](http://www.p2engine.com/examples).

### 9.2.1 General Functions

All scripts start with an object from P2Project and a corresponding schedule object from P2Schedule to construct a project baseline schedule without resources. Project control can be done by an object from the P2Tracking class, as defined along the three following functions:

```
function create_project_object()  
    MyProject=P2Project:new()  
end
```



```

function create_schedule_object()
    MySchedule=P2Schedule:new(MyProject)
end

function create_tracking_object()
    MyTracking=(MyTracking or P2Tracking:new(MySchedule))
end

```

Note that in the example functions above, the name of each object starts with “My” followed by the name of the class, excluding the P2 (e.g. MyProject is an object from the P2Project class). Obviously, any other name can be chosen to generate an object of each class. After its use, an object can be deleted by the nil operator, as follows:

```

function delete_objects()
    MyProject=nil
    MySchedule=nil
    MyTracking=nil
end

```

## 9.2.2 *Project Data*

The basic logic of the network structure, including activity durations but no costs, is available in a Patterson file that can be downloaded from the P2 Engine website for the projects used in this book. These files can be added to P2 Engine by calling the P2Reader functions.

```

function read_a_patterson_file(filename)
    MyReader=P2Reader(MyProject,MySchedule)
    MyReader:read_patterson_file(filename)
    MyReader=nil
end

```

The Patterson files assume finish-start relations between activities with a time-lag of zero. Moreover, since Patterson files only contain activity durations and do not contain data about activity costs, the cost values must be added. This can be done by generating costs or by manually adding the costs as given in Chap. 1, using the following function:

```

function add_costs(projectID)
    - [Add weekly cost, see table 1.2]
    cost_per_time_unit={1000,7500,...,5000}
    for act,cost in ipairs(cost_per_time_unit) do
        MySchedule:set_variable_cost(act,cost)
    end
end

```

Additional project data such as the values for four network topology measures, known and abbreviated as the SP, AD, LA and TF (Vanhoucke 2010a), can easily

be calculated using an object of the P2Generator class. The SP measure is discussed in Chap. 1 and measures the closeness of a project network to a serial or parallel network.

```
function measure_network_topology()
  MyGenerator=P2Generator:new(MyProject,MySchedule)
  SP,AD,LA,TF=MyGenerator:calculate_topological_structure()
  MyGenerator=nil
end
```

### 9.2.3 Baseline Schedules

The construction of a project baseline schedule as given in Chap. 1 boils down to the generation of an earliest start schedule (ESS) using the critical path method, resulting in a planned duration PD and budget at completion BAC, and can be done by the following function:

```
function construct_baseline_schedule()
  MySchedule:earliest_start_schedule_with_float_factor(0)
  PD=MySchedule:get_project_pd()
  BAC=MySchedule:get_project_bac()
end
```

### 9.2.4 EVM/ES Reports

Project progress can be simulated to generate EVM/ES reports using the data presented in Chap. 1. The generation of the EVM/ES reports is done in three steps, as shown in the LUA scripts along the following lines.

**Step 1.** Set the real durations, costs and start times as shown in the Gantt charts of Figs. 1.2, 1.4 and 1.6. The real costs are assumed to change linearly in time according to the baseline unit costs set during baseline scheduling.

```
function set_progress_data(projectID)
  if projectID==1 then
    RealStart={0,1,1,4,4,4,3,3,4,4,9,7,5,5,7,12,12}
    RealDuration={1,3,3,3,5,3,4,2,1,1,3,1,2,1,3,2,1}
  elseif projectID==2 then
    RealStart={0,1,3,5,7,7,8,9,12,13,12,12,10,17,19,20,22}
    RealDuration={1,2,4,1,4,2,3,1,2,5,4,3,5,5,3,5,2}
  else
    RealStart={0,2,5,8,10,13,14,14,14,16,17,19,17,19,21,22,23}
    RealDuration={2,3,5,2,6,1,2,1,1,1,2,2,2,1,2,1,1}
  end
  RealCost={}
  nract=MySchedule:get_number_of_activities()
  for act=1,nract do
    RealCost[act]=MySchedule:get_variable_cost(act)*RealDuration[act]
  end
end
```

**Step 2.** Project progress is imitated using the real activity durations/costs and start times set in the previous step, as given in the following lines:

```
function simulate_progress_manually(RealStart,RealDuration)
  MyTracking:reset_tracking_object()
  - [Construct period tracking periods]
  local timePtr=1
  local projectEnded=false
  while projectEnded==false do
    nrFinish=0
    periodID=MyTracking:add_tracking_period(timePtr)
    nract=MySchedule:get_number_of_activities()
    for act=1,nract do
      if timePtr<RealStart[act] then
        - [Activity: not started yet]
        PC=0
        AD=0
        AS=99999999
      elseif timePtr>=RealStart[act]+RealDuration[act] then
        - [Activity is finished]
        PC=1
        AD=RealDuration[act]
        AS=RealStart[act]
        nrFinish=nrFinish+1
      else
        - [Activity is busy]
        PC=(timePtr-RealStart[act])/RealDuration[act]
        AD=timePtr-RealStart[act]
        AS=RealStart[act]
      end
      MyTracking:set_actual_start(act,AS,periodID)
      MyTracking:set_percentage_completed(act,PC,periodID)
      MyTracking:set_actual_duration(act,AD,periodID)
      AC=MyTracking:get_planned_actual_cost(act,periodID)
      MyTracking:set_actual_cost(act,AC,periodID)
    end
    - [Construct an EVM report for this tracking period]
    MyTracking:construct_evm_report(periodID)
    timePtr=timePtr+1
    if nrFinish==nract then projectEnded=true end
  end
end
```

Project progress can be randomly simulated using predefined distributions, or can be generated using user-defined inputs. In this book, the second option has been chosen in order to influence the outcome in such a way that the reports shown in the previous chapters fully illustrate the use of all EVM, ES and SRA metrics. However, when P2 Engine is used in a research setting, manual artificial project progress can be replaced by a large number of simulation runs under a controlled design in order to test relevant research hypotheses, as for example done in [Vanhoucke \(2010a, 2013d\)](#).

**Step 3.1.** For all the project tracking periods generated in step 2, the EVM and ES metrics can be automatically calculated using the following script:

```
function performance_measurement()
  nract=MySchedule:get_number_of_activities()
  nrtrackper=MyTracking:get_number_of_tracking_periods()
```

```

RC=MyTracking:get_real_project_cost()
RD=MyTracking:get_real_project_duration()
- [The EVM key metrics]
for period=1,nrtrackper do
    PV=MyTracking:get_project_pv(period)
    EV=MyTracking:get_project_ev(period)
    AC=MyTracking:get_project_ac(period)
    ES=MyTracking:get_project_es(period)
end
- [The EVM performance measures]
for period=1,nrtrackper do
    SV=MyTracking:get_project_sv(period)
    CV=MyTracking:get_project_cv(period)
    SPI=MyTracking:get_project_spi(period)
    CPI=MyTracking:get_project_cpi(period)
    SVT=MyTracking:get_project_svt(period)
    SPIT=MyTracking:get_project_spit(period)
    p_factor=MyTracking:get_project_p_factor(period)
end
end
end

```

## 9.2.5 Forecasting Reports

**Step 3.2.** Similar to step 3.1 of the EVM/ES report generation script, the forecasting metrics can be generated for all tracking periods generated in step 2 using the following script:

```

function forecasting()
for period=1,nrtrackper do
    PF1, PFCPI, PFSPI, PFSPIT, PFSCI, PFSCIT,
    PFwSPICPI, PFwSPITCPI=MyTracking:get_eac_cost(period)
    PV1, PV2, PV3, ED1, ED2, ED3, ES1, ES2, ES3=MyTracking:get_eac_time(period)
end
end
end

```

## 9.2.6 Risk Reports

The generation of risk reports to display the time/cost sensitivity metrics using schedule risk analysis can be done using simple or more advanced simulation runs. In the following three scripts, a straightforward SRA study is described using simple symmetrical triangular distributions on the activities of the three projects.

**Step 1.** Create a P2Simulator object

```

function create_simulator_object()
    MySimulator=P2Simulator:new(MySchedule)
end

```

**Step 2.** Run an SRA. In order to generate activity durations from the predefined triangular distributions, all activity durations have been expressed in days by

multiplying their baseline durations by 5 (it is assumed that the durations are expressed in weeks and a week consists of 5 work days). The upper and lower limits of the symmetrical triangular distribution are then set at 20% below and above the baseline duration and the simulations are run using time unit jumps of 1 week (i.e. 5 days). These steps are made to assure a better precision. The number of runs is set as an input parameter (e.g. 100 runs). The parameters of the `run_simulation` procedure need to be set carefully and will not be explained in this chapter. Likewise, the use of the `sim_callback` procedure, which is one of the most important procedures for researchers, is not discussed in this chapter. It is a function that enables P2 Engine to write all intermediate data of each run in the simulation to an output file, which is widely used for research purposes. The interested reader is referred to the P2 Engine website for more details.

```
function run_a_schedule_risk_analysis(nrruns)
  MySimulator:reset_simulation_object(MySchedule)
  MySimulator:create_monte_carlo_object()
  - [Increase precision]
  nract=MySchedule:get_number_of_activities()
  for act=1,nract do
    dur=MySchedule:get_duration(act)
    MySchedule:change_duration(act,dur*5)
  end
  - [Assign triangular distributions to the project activities]
  prob={0,100,0}
  lower=20
  upper=20
  for act=1,MySchedule:get_number_of_activities() do
    MySimulator:set_number_advanced_distributions(act,3)
    a=math.floor((1-(lower/100))*MySchedule:get_duration(act))
    m=math.floor(MySchedule:get_duration(act))
    b=math.ceil((1+(upper/100))*MySchedule:get_duration(act))
    val={a,m,b}
    for x=1,3 do
      MySimulator:set_advanced_distributions(act,pnt,val[x],prob[x])
    end
  end
  end
  time_unit=5
  MySimulator:set_periodic_time_unit(time_unit)
  MySimulator:set_callback("sim_callback")
  MySimulator:run_simulation(nrruns,11,1,1,0,0)
end
```

**Step 3.** The time and cost sensitivity measures can be obtained using predefined P2 Engine functions and are stored in the sensitivity variables, as shown in the following script:

```
function get_sensitivity_measures()
  for act=1,MySchedule:get_number_of_activities() do
    CI,SI,SSI,r,rho,tau=MySimulator:get_activity_time_sensitivity(act)
    c_r,c_rho,c_tau=MySimulator:get_activity_cost_sensitivity(act)
  end
end
```

### 9.2.7 *Download and ... Go!*

All scripts and input files are available on the P2 Engine website and can be used to recalculate all tables displayed in this chapter. It should be noted however that the scripts discussed in this chapter are very basic and do not show the full power of P2 Engine. Rather than generating data by inputting hard values for networks, durations and costs, P2 Engine is mainly used to generate data based on a full factorial research design, allowing the researcher to test any novel idea on its merits to improve the current knowledge on integrated project management and control.

## 9.3 Conclusions

In this chapter, a basic summary of the functionality of P2 Engine is given by briefly describing the seven main classes and illustrating all calculations using example scripts in LUA. The seven main classes are categorised into three groups: data reading or generation, project data definition and performing EVM/ES and SRA calculations. The objects generated from these classes can be integrated and combined in a variety of ways, making P2 Engine an excellent research tool for testing novel, yet unproven ideas.

The chapter has no intention of giving a full overview of the details of the different functions of each class and the reader is referred to the P2 Engine website for a recent update. However, the illustration of simple and straightforward scripts that have been used to construct all reports of the previous chapters are displayed to give the reader a glance of the way P2 Engine must be used. However, the example scripts only show the basic functionality of P2 Engine and a lot more can be done.

Since P2 Engine has initially been developed by and for researchers, its ultimate power and goal lies in its use for purposes for which it was initially not intended. Due to the great flexibility of the seven classes and its integration with the LUA scripting language, P2 Engine can be extended to other project scheduling and control settings. A non-exhaustive summary of recent extensions<sup>1</sup> is given along the following lines:

- **Multi-project planning:** Although P2 Engine and ProTrack have been developed for single project planning, extensions to multi-project planning and resource allocation to a project portfolio can easily be done in P2 Engine using a clever combination of classes and objects. This research project is under construction with the working title *mP2 Engine*, with *m* referring to its multi-project extension.

---

<sup>1</sup>At the time of reading this book, some of these extensions may be implemented as P2 Engine extensions, while others might have been cancelled due to underperformance or failure to improve the current set of available methods.

- The use of statistical project control to construct an automatic threshold on EVM/ES performance metrics, as briefly discussed in Sect. 7.2, has been developed in P2 Engine and test results have been analysed with R ([www.r-project.org](http://www.r-project.org)). Due to the promising test results in [Colin and Vanhoucke \(2013a,b\)](#), SPC functions have been added to ProTrack and P2 Engine, and more extensions are definitely on their way.
- Extensions of the simulation study of [Vanhoucke \(2010a\)](#) to more advanced project control systems are currently on the future research agenda. The extensions from P2 Engine to a project portfolio planning calculator, currently under development under the working title P3 Engine is only one of the future projects of the P2 Engine team.

**Part III**  
**To Reach a Conclusion**



# Chapter 10

## Conclusions

This book reviewed the most commonly used metrics in Earned Value Management (EVM) and Schedule Risk Analysis (SRA) to efficiently control projects in progress. The integrated view on project management and control, seen as a combination of baseline scheduling and schedule risk analysis within a project control framework, has led to the development of new control methods, research results and two software systems discussed in this book. The intention of this book was not to provide a full summary of the literature or a description of the best practices within project control, but rather to provide a short and easy-to-use overview of the currently existing methods and techniques.

In this conclusive section, a short overview and forward look is given, as summarised in the following sections. In Sect. 10.1, the chapters of the book are briefly reviewed and their main topics are highlighted. Section 10.2 gives a small non-complete overview of research studies that will be carried out in the near future and that are linked with the individual topics discussed in this book. In Sect. 10.3, a large research study of over 1 million euro is discussed in order to illustrate the future research paths that the research group at Ghent University will take.

### 10.1 Summary

In Part I of this book, most of the metrics used for two alternative control methods, known as top-down project control and bottom-up project control, have been reviewed, as explained along the following lines. In Chaps. 2–5, the most commonly used EVM metrics are discussed and illustrated on fictitious project examples, while Chap. 6 is written to give an overview of the most commonly known sensitivity metrics used in an SRA. In Chap. 7, the EVM and SRA metrics are used to make a distinction between a top-down and bottom-up project control approach.

EVM consists of three key metrics, known as Planned Value (PV), Actual Cost (AC) and Earned Value (EV) that are periodically measured on low levels of the

Work Breakdown Structure (WBS) and that are transformed into general project based performance measures on higher WBS levels. The cost performance measures are known as the Cost Variance (CV) and the Cost Performance Index (CPI), while the time performance measures are known as the Schedule Variance (SV) and the Schedule Performance Index (SPI).

Due to an unreliability in the schedule performance measures, the three key metrics have been extended to a fourth key metric, known as the Earned Schedule (ES) metric, which has resulted in two alternative time performance measures, known as the Schedule Variance (SV(t)) and the Schedule Performance Index (SPI(t)).

Thanks to the success of the Earned Schedule method to measure and monitor the time dimension of a project in progress, a relatively new concept, known as the schedule adherence concept and formalised in the p-factor approach, has been developed.

Both the traditional EVM performance measures and the newly developed ES methods, thanks to their strong relation further abbreviated as the EVM/ES metrics, are used for forecasting the expected time and cost of a project in progress. In this book, nine duration and eight cost forecasting methods have been presented, and the link with their accuracy and stability calculations has been shown.

While all EVM/ES metrics are used in a top-down project control approach that provides performance information on high WBS levels to trigger the need for drilling down to lower levels in case of project problems and/or opportunities, an alternative bottom-up project control method exists. In a Schedule Risk Analysis (SRA), sensitivity metrics for time and cost are provided that measure the potential impact of time and cost changes in the project activities, defined at the lowest level of the WBS, on the overall project objective. Unlike the EVM/ES metrics, the SRA sensitivity metrics can therefore be used in a bottom-up project control approach.

In Part II of this book, two software systems have been proposed. They incorporate all EVM/ES and SRA metrics discussed in this book and have been developed by OR-AS. ProTrack ([www.protrack.be](http://www.protrack.be)) is the commercial software tool for integrated project management and control, including advanced features for baseline scheduling, schedule risk and project control, and is inspired by the research study of Vanhoucke (2010a). It is used in the classroom for Project Management courses as well as in company trainings to illustrate best practices on efficient and effective project management and control. Since ProTrack is intensively used by students and professionals that collaborate with the team of OR-AS and OR&S, the empirical data obtained for our research studies come from the ProTrack files received from various sources.

P2 Engine ([www.p2engine.com](http://www.p2engine.com)) is nothing more than an advanced version of ProTrack. Having removed all GUI parts and having embedded the source code in a scripting language to enable users to access all procedures, it has become a tool for and by researchers to move up the integrated project management and control research and to bring it closer to the professionals. P2 Engine has been developed under the belief that research drives innovation. P2 Engine is and will be a driving force for all the research endeavours of the research group at Ghent University, for

which some of the potential research projects will be discussed in the following sections.

## 10.2 Future Work

It has been mentioned in the introduction that this book has no intention to give a full summary of the excellent research that has been done during the last decades. However, although this book is not a research book, most, if not all work discussed in the previous chapters is the result of hard and intensive research by academics and professionals throughout the world. Therefore, it goes without any doubt that future research is definitely indispensable for advancing the Project Management discipline. A complete flash forward to interesting research avenues is outside the scope of this conclusive chapter, but instead, three project control research lines that will be carried out in the near future at the Operations Research and Scheduling group (OR&S, [www.projectmanagement.ugent.be](http://www.projectmanagement.ugent.be)) in collaboration with OR-AS ([www.or-as.be](http://www.or-as.be)) are briefly reviewed. An overview of the current research projects and preliminary results of research and software developments will be continuously updated in the online overview article of [Vanhoucke \(2013g\)](#) and the online book of [Vanhoucke \(2013a\)](#).

### 10.2.1 *Stability Versus Accuracy*

In Sect. 5.3, the quality of EVM/ES forecasting methods has been evaluated by measures of forecasting accuracy and stability. While accuracy is measured by the relative or absolute error a forecasting method makes, stability is defined as a measure for the variation in this error. Research on forecasting accuracy of EVM/ES methods has been published in literature and a summary of research is given in [Vanhoucke \(2010a\)](#). Stability has always been a central point in EVM research, primarily with a focus on cost control, and is widespread in the popular and – to a lesser extent – academic literature. However, research on the integration between accuracy and stability, and the possible trade-off between the two quality measures, is to the best of my knowledge completely void. More than a decade after the introduction of the Earned Schedule concept proposed by [Lipke \(2003\)](#) and discussed in Chap. 3, researchers and practitioners have investigated, validated, criticised and modified this novel concept and successfully used it in research simulation studies, software developments and practical project control settings. Only 1 year later, [Lipke \(2004\)](#) has proposed an extension of his Earned Schedule concept to measure the influence of rework due to the lack of scheduling adherence on the earned value. This concept is known as the p-factor approach and has been discussed in Chap. 4. Unlike the ES concept, the p-factor approach has remained in the background, and little adoption by both researchers and practitioners has been

detected. Only a few papers have been written (Lipke 2004; Vanhoucke 2013b,c) and it is therefore hoped that new research using this promising concept is under development. It is believed and conjectured that this p-factor approach, and similar extensions, can potentially help to fill the void in the integrative accuracy/stability research need.

### 10.2.2 Improved Forecasting

Next to investigating the relation and the potential trade-off between forecasting accuracy and stability, improvement on the forecasting methods could also lead to insightful and relevant research results. As an example, the eight cost forecasting methods and the nine time forecasting methods of Chap. 5 can be further improved by combining them in a clever way. The rationale supporting these so-called *composite forecasts* is the fact that every method uses some kind of information other methods do not use. Hence, even if a forecasting method does not perform well in and by itself, it may contain useful information that can lead to improved forecast accuracy. The goal is to find the best weight for each forecasting method such that forecasting accuracy, measured by the MAPE and MPE evaluation measures of Sect. 5.3.1, is maximised. Other similar extensions, such as *probabilistic forecasting* or extended integration between SRA and EVM to improve the forecasting accuracy of project duration and total cost are interesting research directions to further improve the ability to predict the final project outcome. An example study that integrates the concepts of Chaps. 3 and 6 to improve the prediction of a project's duration using earned schedule method is given in Elshaer (2013).

## 10.3 Concerted Research Action

Although the initial research efforts in project management stem from the early 1950s, and despite numerous and continuous research efforts in the project management literature, there is still a wide gap between the academic literature and the needs and requirements formulated in practice (Hartmann and Briskorn 2010), and hence fundamental research should put a clear and strong focus on topics that contribute to the academic state-of-the-art knowledge as well as on the practical relevance. The main reasons for the discrepancy between academic research and practical use are given as follows:

- A main focus is put on the cost objective of projects, while the time and scope dimensions have received relatively less attention.
- The main focus lies on the development of static solution procedures, while the attention to a more dynamic oriented approach is growing but still relatively new.

- There is little integration of different interlinked phases during the whole project management process.
- There is little research done to reveal the actual drivers that determine a project's success.

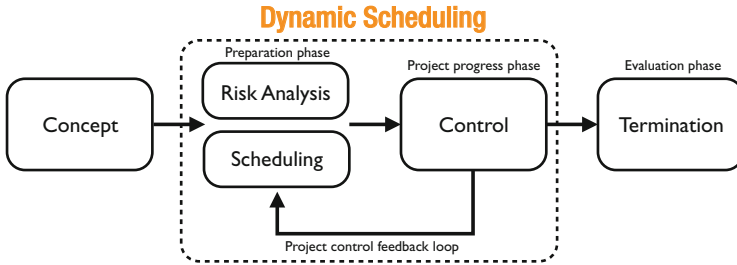
Inspired by these shortcomings of academic research, a new research project proposal has been recently submitted and approved and has resulted in one of the largest research fundings known in Project Management academic research. The research is known under the name *Concerted Research Action* funded by Ghent University and aims at formulating answers on five research questions summarised in three clear and interlinked research lines to tighten the pre-described discrepancy. These three research lines are based on research experience from the academic literature, as well as on expertise gained from the project management field. The research proposal aims at providing a profound and critical answer to the following research questions:

- What is the impact of a higher baseline schedule realism and a higher schedule risk accuracy on the required intensity and focus of the project control phase?
- What is the impact of various sources of resource disruptions/instabilities during project progress on the project performance characteristics and the quality and effectiveness of the project control phase?
- What is the relation between the degree of these resource disruptions/instabilities during project progress and the need for highly realistic project schedules and highly accurate risk analyses?
- What are the main drivers to determine the optimal project management and contingency reserves?
- Can statistical project control predictions be used to control the project baseline schedule performance and its corresponding project reserves?

Each of these research questions will be very briefly outlined in the following paragraphs and relies on a three-phased stepwise approach. Ideally, it should contribute to the search for static and dynamic drivers that determine the success or failure of a project in order to tighten the discrepancy between theory and practice.

### ***10.3.1 Static Integration***

The first research line of the project aims at the integration of three fundamental phases of the project life cycle, known as baseline scheduling, risk analysis and project control into a single static modelling approach. Figure 10.1 displays the static integration between the three project life cycle phases and the necessary feedback loop and data sharing between the dynamic project control phase and the static scheduling and risk analysis phases. The research aims at investigating the impact of a higher baseline schedule realism and a higher schedule risk accuracy on the required intensity and focus of the project control phase. Since both scheduling



**Fig. 10.1** Research study 1. Static integrated project control

and risk analysis happens prior to the start of the project, the research focus is said to be static.

### 10.3.2 *Dynamic Integration*

While the first research line primarily assumes a static focus on the integration between three crucial project life cycle phases, the second research line aims at tightening the gap between the academic static scheduling procedures and the dynamic needs in practice. More precisely, the second research line of the project aims at the dynamic incorporation of project disruptions in the project control phase during the project's progress. The results from this study should lead to a higher degree of baseline schedule realism and an improved schedule risk accuracy. The research is said to be dynamic due to its focus on dynamic resource disruptions and/or instabilities during project progress. It wishes to measure the impact of various sources of resource disruptions/instabilities during project progress on the project performance characteristics and the quality and effectiveness of the project control phase. A conceptual graphical overview of the second research line is displayed in Fig. 10.2.

### 10.3.3 *Statistical Project Control*

The third research line of the project proposal aims at the integration of the static (Fig. 10.1) and dynamic (Fig. 10.2) information into a single project control decision support system. Moreover, it aims at the development of a statistical project control method that incorporates the static and dynamic information in order to better support corrective actions in case the project objectives are in danger. Given the simplicity and speed of generating data using Monte Carlo simulations, the use of statistical techniques for controlling projects has been briefly touched upon in Sect. 7.2. Despite the simplicity of most methods, these techniques could potentially

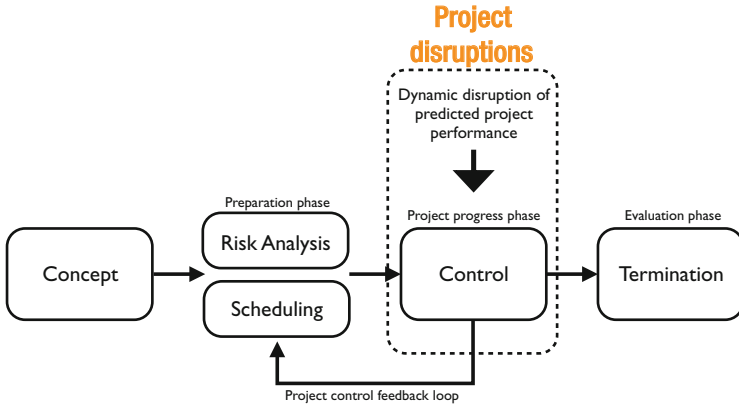


Fig. 10.2 Research study 2. Dynamic integrated project control

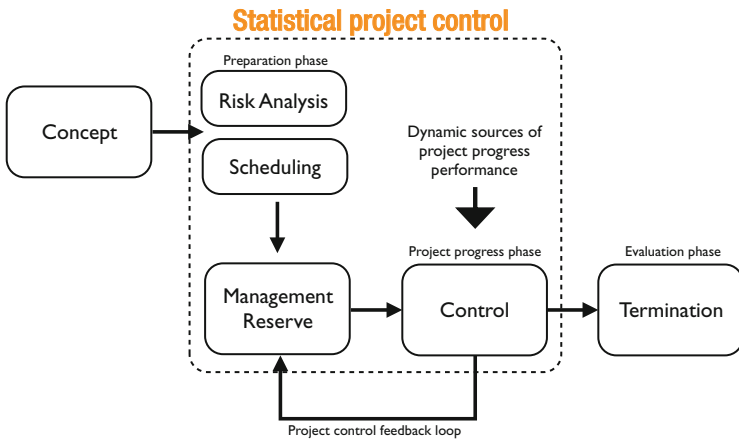


Fig. 10.3 Research study 3. Statistical project control

lead to major improvements in terms of control efficiency. It is therefore hoped and believed that the research endeavours published in [Colin and Vanhoucke \(2013a,b\)](#), which can be considered as statistical extensions of the top-down project control approach of [Vanhoucke \(2011\)](#) are only the tip of the iceberg. Figure 10.3 displays the statistical project control research line that aims at improving the efficiency of project control systems in order to better monitor the performance of projects in progress as well as to better predict the main drivers that determine the optimal project management and contingency reserves.

# Bibliography

- Anbari, F. (2003). Earned value project management method and extensions. *Project Management Journal*, 34(4), 12–23.
- Batselier, J., & Vanhoucke, M. (2013). *Performing risk analysis and project control on real-life projects: An evaluation of established and novel metrics and techniques*. Working paper, Ghent University.
- Book, S. (2006a, Fall). Correction note: “Earned schedule” and its possible unreliability as an indicator. *The Measurable News*, 22–24.
- Book, S. (2006b, Spring). “Earned schedule” and its possible unreliability as an indicator. *The Measurable News*, 24–30.
- Christensen, D. (1993). The estimate at completion problem: A review of three studies. *Project Management Journal*, 24, 37–42.
- Christensen, D. (2004, Winter). Is the cumulative SCI-based EAC an upper bound to the final cost of post-A12 defense contracts? *Journal of Cost Analysis and Management*, 6, 1–10.
- Christensen, D., Antolini, R., & McKinney, J. (1995, Spring). A review of estimate at completion (EAC) research. *Journal of Cost Analysis and Management*, 41–62.
- Christensen, D., & Heise, S. (1993). Cost performance index stability. *National Contract Management Journal*, 25, 7–15.
- Christensen, D., & Payne, K. (1992). Cost performance index stability – Fact or fiction? *Journal of Parametrics*, 10, 27–40.
- Christensen, D., & Rees, D. (2002, Winter). Is the CPI-based EAC a lower bound to the final cost of post A-12 contracts? *Journal of Cost Analysis and Management*, 4, 55–65.
- Christensen, D., & Templin, C. (2002). EAC evaluation methods: Do they still work? *Acquisition Review Quarterly*, 9, 105–116.
- Colin, J., & Vanhoucke, M. (2013a). *A multivariate approach to statistical project control using earned value management*. Working paper at Ghent University, currently under journal submission (Check [www.projectmanagement.ugent.be](http://www.projectmanagement.ugent.be) for an update).
- Colin, J., & Vanhoucke, M. (2013b). *Setting tolerance limits for statistical project control using earned value management*. Working paper at Ghent University, currently under journal submission (Check [www.projectmanagement.ugent.be](http://www.projectmanagement.ugent.be) for an update).
- Covach, J., Haydon, J., & Riether, R. (1981). *A study to determine indicators and methods to compute estimate at completion (EAC)*. Virginia: ManTech International Corporation.
- Demeulemeester, E., Vanhoucke, M., & Herroelen, W. (2003). Rangen: A random network generator for activity-on-the-node networks. *Journal of Scheduling*, 6, 17–38.
- Elshaer, R. (2013). Impact of sensitivity information on the prediction of project’s duration using earned schedule method. *International Journal of Project Management*, 31, 579–588.



- Fleming, Q., & Koppelman, J. (2005). *Earned value project management* (3rd ed.). Newtown Square: Project Management Institute.
- Hartmann, S., & Briskorn, D. (2010). A survey of variants and extensions of the resource-constrained project scheduling problem. *European Journal of Operational Research*, 207, 1–15.
- Henderson, K., & Zwikael, O. (2008). Does project performance stability exist? A re-examination of CPI and evaluation of SPI(t) stability? *Crosstalk – The Journal of Defense Software Engineering*, 21, 7–13.
- Hulett, D. (1996). Schedule risk analysis simplified. *Project Management Network*, 10, 23–30.
- Jacob, D. (2003, March). Forecasting project schedule completion with earned value metrics. *The Measurable News*, 1, 7–9.
- Jacob, D. (2006, Fall). Is “earned schedule” an unreliable indicator? *The Measurable News*, 15–21.
- Jacob, D., & Kane, M. (2004, Summer). Forecasting schedule completion using earned value metrics? Revisited. *The Measurable News*, 1, 11–17.
- Kolisch, R., & Sprecher, A. (1996). PSPLIB – A project scheduling problem library. *European Journal of Operational Research*, 96, 205–216.
- Kuhl, M. E., Lada, E. K., Steiger, N. M., Wagner, M. A., & Wilson, J. R. (2007). Introduction to modeling and generating probabilistic input processes for simulation. In S. Henderson, B. Biller, M. Hsieh, J. Shortle, J. Tew, & R. Barton (Eds.), *Proceedings of the 2007 winter simulation conference*, Washington, DC (pp. 63–76). New Jersey: Institute of Electrical and Electronics Engineers.
- Lipke, W. (2003, Summer). Schedule is different. *The Measurable News*, 31–34.
- Lipke, W. (2004, Winter). Connecting earned value to the schedule. *The Measurable News*, 1, 6–16.
- Lipke, W. (2012). *Earned schedule*. lulu.com.
- Lipke, W., Zwikael, O., Henderson, K., & Anbari, F. (2009). Prediction of project outcome: The application of statistical methods to earned value management and earned schedule performance indexes. *International Journal of Project Management*, 27, 400–407.
- PMBOK. (2004). *A guide to the project management body of knowledge* (3rd ed.). Newtown Square: Project Management Institute, Inc.
- Trietsch, D., Mazmanyan, L., Govergyan, L., & Baker, K. R. (2012). Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation. *European Journal of Operational Research*, 216, 386–396.
- Vandevoorde, S., & Vanhoucke, M. (2006). A comparison of different project duration forecasting methods using earned value metrics. *International Journal of Project Management*, 24, 289–302.
- Vanhoucke, M. (2008, August). Project tracking and control: Can we measure the time? *Projects and Profits*, 35–40.
- Vanhoucke, M. (2009). Static and dynamic determinants of earned value based time forecast accuracy. In T. Kidd (Ed.), *Handbook of research on technology project management, planning, and operations* (pp. 361–374). Hershey: Information Science Reference.
- Vanhoucke, M. (2010a). *Measuring time – Improving project performance using earned value management* (Volume 136 of International series in operations research and management science). Dordrecht: Springer.
- Vanhoucke, M. (2010b). Measuring time: An earned value performance management study. *The Measurable News*, 1, 10–14.
- Vanhoucke, M. (2010c). Using activity sensitivity and network topology information to monitor project time performance. *Omega – The International Journal of Management Science*, 38, 359–370.
- Vanhoucke, M. (2011). On the dynamic use of project performance and schedule risk information during project tracking. *Omega – The International Journal of Management Science*, 39, 416–426.
- Vanhoucke, M. (2012a). Dynamic scheduling: Integrating schedule risk analysis with earned value management. *The Measurable News*, 2, 11–13.

- Vanhoucke, M. (2012b). Measuring the efficiency of project control using fictitious and empirical project data. *International Journal of Project Management*, 30, 252–263.
- Vanhoucke, M. (2012c). *Project management with dynamic scheduling: Baseline scheduling, risk analysis and project control* (Vol. XVIII). Berlin/New York: Springer.
- Vanhoucke, M. (2013a). *The art of project management: A story about work and passion*. Available online at [www.or-as.be](http://www.or-as.be).
- Vanhoucke, M. (2013b). The impact of project schedule adherence and rework on the duration forecast accuracy of earned value metrics. In E. C. Hoffmann (Ed.), *Project management: Practices, challenges and developments*. Hauppauge: NOVA Publishers.
- Vanhoucke, M. (2013c). Measuring schedule adherence. *The Measurable News*, 4, 21–26.
- Vanhoucke, M. (2013d). An overview of recent research results and future research avenues using simulation studies in project management. *ISRN Computational Mathematics*, Article ID:513549, 1–19.
- Vanhoucke, M. (2013e). Project baseline scheduling: An overview of past experiences. *Journal of Modern Project Management*, 1(2), 18–27.
- Vanhoucke, M. (2013f). Project management using dynamic scheduling: Baseline scheduling, risk analysis and project control. *The Measurable News*, 2, 45–50.
- Vanhoucke, M. (2013g). *Welcome to OR&S! Where students, academics and professionals come together* (pp. 1–15). Available online at [www.or-as.be/books/research](http://www.or-as.be/books/research).
- Vanhoucke, M., Coelho, J., Debels, D., Maenhout, B., & Tavares, L. (2008). An evaluation of the adequacy of project network generators with systematically sampled networks. *European Journal of Operational Research*, 187, 511–524.
- Vanhoucke, M., & Shtub, A. (2011). Adding value to earned value analysis. *PM World Today*, XIII(1).
- Vanhoucke, M., & Van Acker, T. (2010). *Dynamic scheduling on your desktop*. Available online at [www.or-as.be/books](http://www.or-as.be/books).
- Vanhoucke, M., & Vandevoorde, S. (2007a, Winter). Measuring the accuracy of earned value/earned schedule forecasting predictors. *The Measurable News*, 26–30.
- Vanhoucke, M., & Vandevoorde, S. (2007b). A simulation and evaluation of earned value metrics to forecast the project duration. *Journal of the Operational Research Society*, 58, 1361–1374.
- Vanhoucke, M., & Vandevoorde, S. (2008, Summer). Earned value forecast accuracy and activity criticality. *The Measurable News*, 13–16.
- Vanhoucke, M., & Vandevoorde, S. (2009, Spring). Forecasting a project's duration under various topological structures. *The Measurable News*, 26–30.
- Williams, T. (1992). Criticality in stochastic networks. *Journal of the Operational Research Society*, 43, 353–357.