Does Systems Engineering in Space Projects Pay?

R.J. Hamann
Delft University of Technology
Aerospace Engineering
Space Systems Engineering
P.O. Box 5058, 2600 GB Delft
The Netherlands
R.J.Hamann@tudelft.nl

B.T.C. Zandbergen
Delft University of Technology
Aerospace Engineering
Space Systems Engineering
P.O. Box 5058, 2600 GB Delft
The Netherlands
B.T.C.Zandbergen@tudelft.nl

P.J. Zijdemans
Dutch Space
P.O. Box 32070, 2303 DB Leiden
The Netherlands
F.Zijdemans@dutchspace.nl

Abstract. This paper attempts to answer the question whether it "pays" to apply Systems Engineering methods, tools and techniques within a space project. To this purpose a possible correlation has been investigated between the Systems Engineering effort applied within a number of space projects and the project result in terms of technical quality, cost and schedule.

Use has been made of historical data derived from the results of Systems Engineering audits of projects, some recent audits performed along the same lines and assessments of project results in terms of technical quality, cost and schedule by the systems engineers involved in the projects. Basis for the audits is a checklist addressing 93 different aspects of Systems Engineering in the field of requirements, concept design, design & development, verification and technical management. In total nine data sets related to six projects in the industrial and the academic world were used.

Although the data obtained are rather "noisy" there appears to be a clear positive correlation between the SE effort applied and the project result. It appears also that the positive effects mainly show up in the cost and schedule results of the project, the technical quality of the project result being generally of a rather satisfactory level.

Examining the results in detail the Systems Engineering effort in the field of requirements, design & development and technical management has the strongest correlation with the project result. The effort in the field of (concept) design and verification shows a less strong correlation.

The data have been "refined" by deleting the projects that were most strongly influencing the correlation in a positive sense. The overall results remained, however, the same.

An overview is given of those aspects generally receiving little attention in the Systems Engineering effort. Further analysis of these results will be the subject of a follow-on study.

Introduction

Motivation for the research. The formal motivation for doing Systems Engineering (SE) is to obtain oversight of and control on a relatively complex project in order to produce a product fulfilling the needs of the user and/or customer and to do so within the constraints of available resources and in an explicit and structured way. However, in view of overall cost efficiency, one needs to know in addition whether it is worth the additional effort of defining and applying all those tools, methods and procedures, in short generating a Systems Engineering Management Plan and applying it during the project.

This research has been performed to investigate whether objective proof could be produced that this is the case, in other words: Does the application of Systems Engineering in a project contribute to the project success in terms of technical quality, cost and schedule. Honour and Mar (2002) and Honour (2004) investigated this same question showing generally a positive correlation between the Systems Engineering effort and the project result. In that study, however, the quality of the Systems Engineering in a project has been measured with a subjective evaluation on a scale of 0 to 10 and the relative cost of the time spent on SE tasks. Contrary to this study we will attempt to show in this paper which Systems Engineering tasks and activities contribute most to the project result. The study does not give information on which specific activities the money has been spent. Impact on schedule and cost however has been assessed into greater detail than in this current study.

Some studies (Boehm, 2003) have aimed to model the Systems Engineering process to attempt to optimize the amount of Systems Engineering to be applied in a project. Although undoubtedly useful for the good case (when applied, it certainly will yield a better project result), they again do not provide direct insight of the effectiveness of specific elements of Systems Engineering. Also, these studies tend to be biased towards software intensive systems (Boehm, 2008)

An inherent problem with research in the effectiveness of Systems Engineering is also that nobody ever performed two identical projects, one with and one without Systems Engineering. And nobody ever did execute a project purposely banning Systems Engineering tasks from its execution. The only thing to be found is individual case studies showing the improvement of project performance after Systems Engineering had been introduced (Hole, 2005). So the best we can do is to assess projects based on the SE effort performed and to assume that project constraints and characteristics are comparable.

Methodology

General. The method selected was to use the results of questionnaires used for Systems Engineering audits performed for six space projects (1995-2005, confidential company documentation; 2008, pers.dis.) executed in an industrial or academic environment as a measure for the extent Systems Engineering tasks have been performed in these projects and to obtain an assessment of the project result asking key project personnel to complete a small questionnaire. Based on these data it has been investigated, whether a correlation between these two parameters could be established. Part of the material came from historical files; part has been generated specifically for this research. In the last case more than one person has been interviewed to prevent undesired personal bias in the results. In one case a second questionnaire has been completed to get an impression about the stability of the assessment in time.

Projects Selected. The projects selected are listed in table 1. None of these projects were considered to be a failure, although some certainly have not been as profitable as desired. The table also specifies the time the Systems Engineering audit has been performed or the questionnaire (Systems Engineering checklist) has been completed, the phase the project was is at that moment and the date of completion of the project. For Delfi-C^3 both Systems Engineering activities and project results have been evaluated by two persons and included in the data as two individual cases. The same applies for the Huygens project. For the Herschel-Planck project two interviews on the Systems Engineering activities have been conducted, one at the start and one at the end of the C/D phase of the project. So in total 9 data points are included in the study. Figures 1 and 2 show a picture of these projects.

Table 1 Projects examined

Project	*Time of SE audit/ SE checklist completion*		*End of Project*
	Year	*Project phase*[1]	
Crew Transfer Vehicle[2]	1994	A	1998
Delfi-C3[3]	2008	End of Project	2008
Herschel-Planck	2004	Kick-off C/D	2008
Herschel-Planck	2008	End of Project	2008
Huygens[4]	2008	End of Project	1995
MIPAS	1995	C/D	1999
Sloshsat/Flevo	1996	C	2005

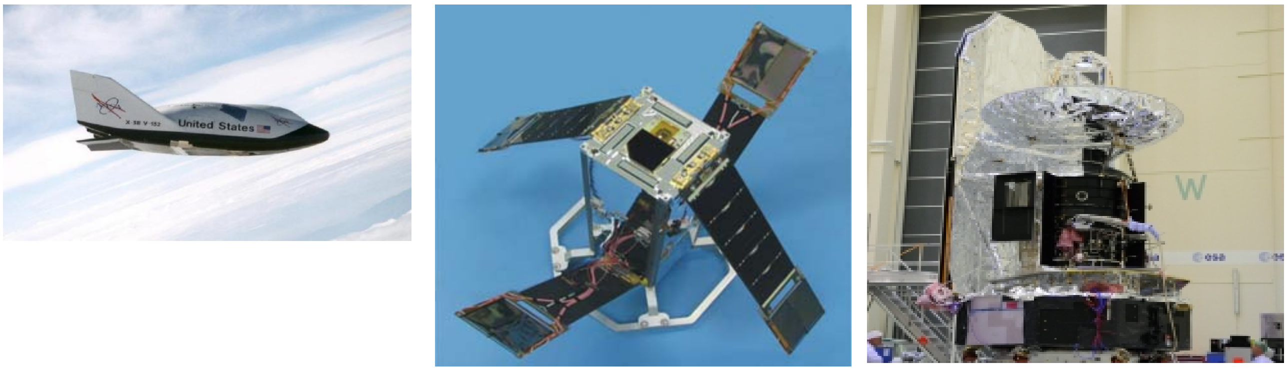

Figure 1 Left to right: the projects Crew Transfer Vehicle, Delfi-C^3 and Herschel

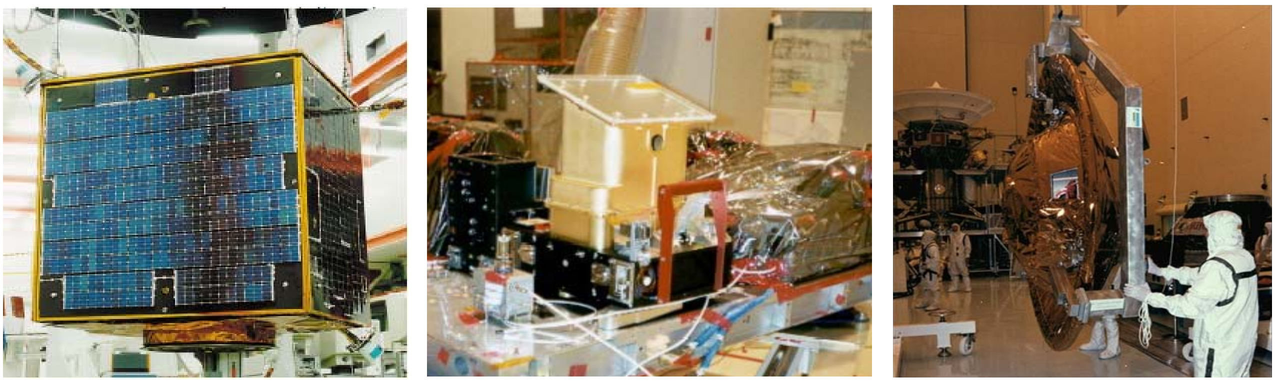

Figure 2 Left to right: the projects Sloshsat/Flevo, MIPAS and Huygens

Systems Engineering checklist. The Systems Engineering checklist (Project Handbook Standard, 2004) is a questionnaire containing a set of questions covering the full field of Systems Engineering activities. An auditor, interviewing the key project personnel in order to establish the extent Systems Engineering is applied in a project, completes the checklist. Based on the findings recommendations are made to the project to improve their performance.

The categories addressed in the checklist are: Requirements (20 items), (Concept) design (32 items), Design & development (16 items), Verification (8 items) and Technical management

[1] A, C, C/D: Project phases.
[2] Project canceled in phase C.
[3] Two Systems Engineering checklists completed.
[4] Two Systems Engineering checklists completed.

(17 items), so total 93 items. Table 2 shows typical keywords in this questionnaire.

Table 2 Systems Engineering checklist keywords

Requirements	*Concept design*	*Design & development*	*Verification*	*Technical Management*
Key requirements Requirements traceability Design maturity -ilities Risk analysis	Functional description Overview of trades and trade results Product tree Technical resource budgets Technical performance parameters Sensitivities Design & performance description Interface control documents Design-to-cost methodology	Design & development logic Design Verification Matrix Assembly, integration & verification flow Ground support equipment requirements	Verification control document Regression test methodology	Internal and external design reviews Review data package definition Data consistency control Documentation trees Document & change control Work flow diagrams
… and above all: " … documented evidence […] to enable maintenance and communication."				

The total score of a project is simply calculated by counting the number of SE tasks performed and dividing that by the total number of items.

To get an impression how typical checklist questions are formulated Figure 3 reproduces a sample of it.

no.	characteristic	check/notes
1	***REQUIREMENTS***	
1.1	The project has a **list of Key Requirements**, identifying	
1.1.a	the requirements which the **customer** finds of extreme importance	
1.1.b	the requirements, that **drive the design** of the concept(s) considered	
1.1.c	the requirements, for which major **non-compliances** exist or are expected	
1.1.d	the requirements, which are still **TBD or TBC** and which are suspected to be potential candidates to be included in the set listed above or which may become killer requirements (i.e. invalidate the concept(s) considered)	
1.2	The project has a **Requirements Traceability Matrix**, which identifies	
1.2.a	the relation between the requirements imposed on the contractors product and the **higher level** customer requirements	
1.2.b	the relation between the contactor's requirements and the requirements imposed on the **subcontractors**, including the rationale for them and the analysis used to derive them	
1.2.c	requirements, for which **no rationale** may be found on any level	
1.3	The project has an up-to-date and documented knowledge, in how far the requirements imposed on the contractor and his subcontractors are met and by which method(s) this is demonstrated (i.e. explicit knowledge of the **design maturity**)	

Figure 3 Systems Engineering checklist - example

The way the questions in this checklist are formulated shows that the presence of Systems Engineering **functions** is evaluated, not the application of tools, procedures or software packages. So during the audit the character, size and complexity of the project are taken into account, when formulating the answer.

Project success criteria. To assess the result of the projects success criteria are defined for each of the aspects cost, schedule and technical quality. Their definition is given in Table 3.

Table 3 Project success criteria

Score	*Cost*	*Schedule*	*Technical quality*
2	> 6% profit	No schedule problems at all	Very good technical quality
1	6% > profit > 0%	Minor schedule problems	Good quality
0	0% profit	On schedule	Sufficient quality
-1	6% > loss > 0%	Slight delays	Slight rework
-2	> 6% loss	Serious delays	Serious rework

To quantify the schedule score better the following definitions have been added for the lower three schedule scores:

- On schedule: can be repaired by incidental overtime
- Slight delays: must be repaired by structural, excessive overtime
- Serious delay: delay causes slip of total (higher level) system

The total "score" on project result has been determined by simply adding the individual scores in each of the aspects cost, schedule and technical quality, leading to a project score in the range -6 to +6.

It shall be remarked that in European institutional space projects a profit between 0 and 6% is already considered to be a good project result and that a three to four months schedule delay per year (by external and internal causes) seems to be an "accepted standard".

Evaluation. The evaluation of the results is done by plotting the percentage of the Systems Engineering tasks performed in the project on the horizontal axis and the project result on the vertical axis. A linear trend line is generated of which the slope is a measure for the amount of correlation between these two variables. Figure 4 shows such a typical correlation; the steeper this line, the more effective the SE effort is for the project result.

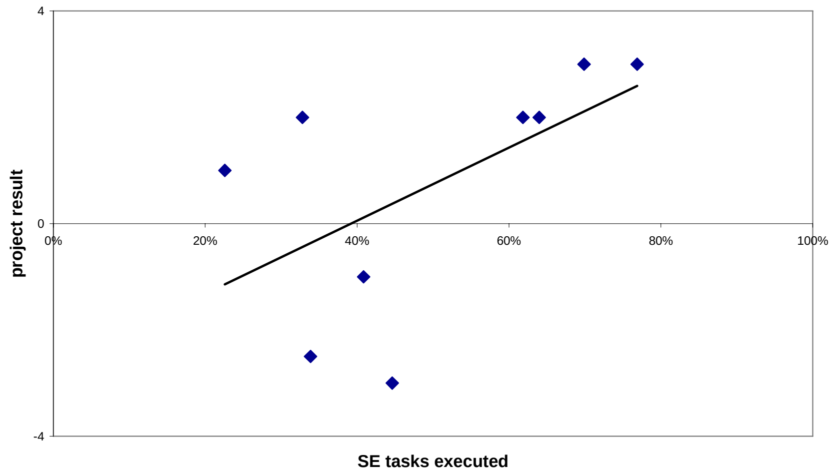

Figure 4 Correlation SE effort – project result. The steeper the trend line, the more effective the SE effort is for the project result.

Results

Raw data. The raw data are shown in Figure 5. Although project results may vary quite a lot for the same Systems Engineering effort applied, there is a clear tendency that the project result improves with increasing SE effort.

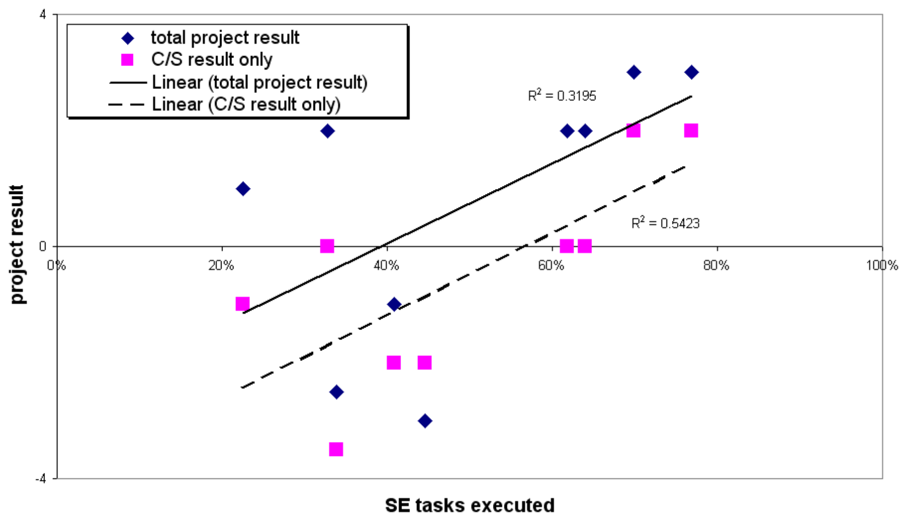

Figure 5 Correlation for the nine project assessments

The figure contains also the correlation between the project result in terms of cost and schedule only ("**C/S result only**" in Figure 5). We observe that the slope of the trend line remains about the same, suggesting that the major impact of the Systems Engineering effort is on these two project aspects, and not that much on technical quality. Figure 6, expressing the project result in terms of technical quality only confirms this (technical quality scores an average of 1.4 on a scale of –2 to +2). This result is not surprising; space (scientific) projects are performance driven; that is the technical result shall always be good at the expense of additional cost and schedule slip.

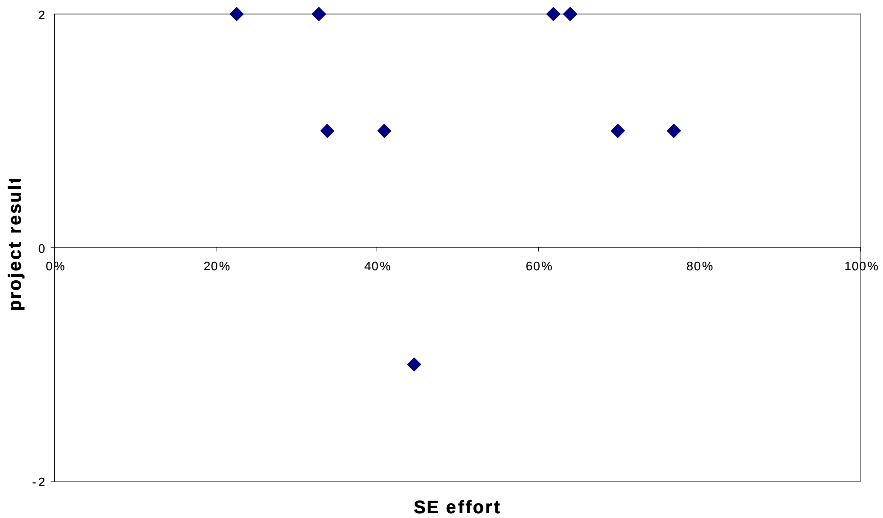

Figure 6 Correlation of technical project result and SE effort

Cleaning up the data. Delfi-C^3 is not really comparable with the other projects. It is a nano-satellite of 2.2 kg mass and with a power of 2.8 W, and only limited "space" standards and rules have been applied. Above all, it is a student and university project, where educational objectives were the most important and have precedence over schedule and technical quality, and, to a certain extent, cost. As a consequence one of the learning objectives was to let the students themselves experience how important Systems Engineering is for the project result. And learning by making mistakes is most effective…..

So we may consider removing Delfi-C^3 to obtain a better picture of the correlation. We remove it, as Delfi-C3 scores mainly in the bottom left corner of the graph for almost all aspects, so its elimination will decrease the slope of the trend line, hence decrease the positive correlation. The result is shown in Figure 7.

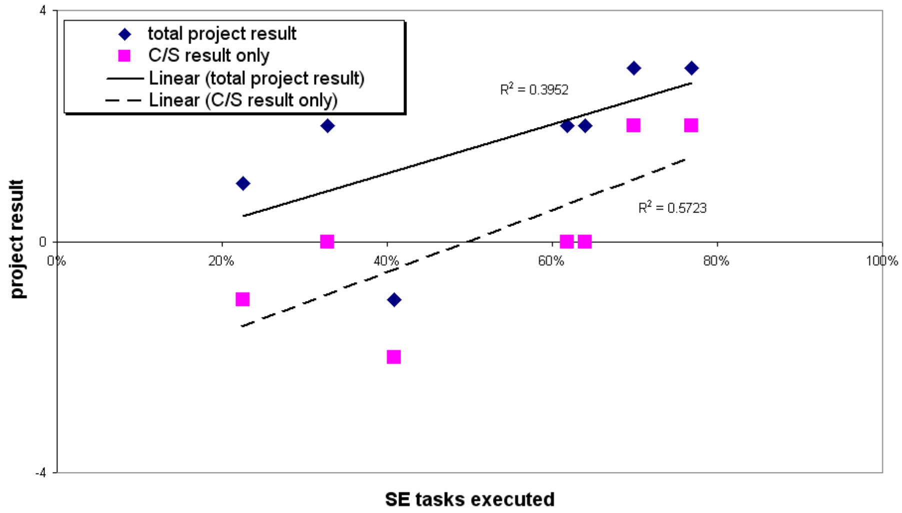

Figure 7 Correlation of project result and SE effort – without Delfi-C3

Contributing factors. To obtain a better picture of the Systems Engineering components which contribute most (or least) to the project result the scores of the projects for each of the categories Requirements (REQ), (Concept) Design (CD), Design & Development (D&D), Verification (VER) and Technical Management (TM) have been examined. Results are compiled in figure 8.

It appears that the project result benefits most from SE effort in the categories Requirements, Design & Development and Technical management; SE activities in the categories (Concept) Design and (especially) Verification contribute to a lesser extent to the project result. Another observation to be made is that increasing Systems Engineering effort contributes most on cost and schedule; these trend lines are steeper. This is consistent with the observation made earlier that the SE effort applied has not much effect on technical quality (related to the projects being performance driven).

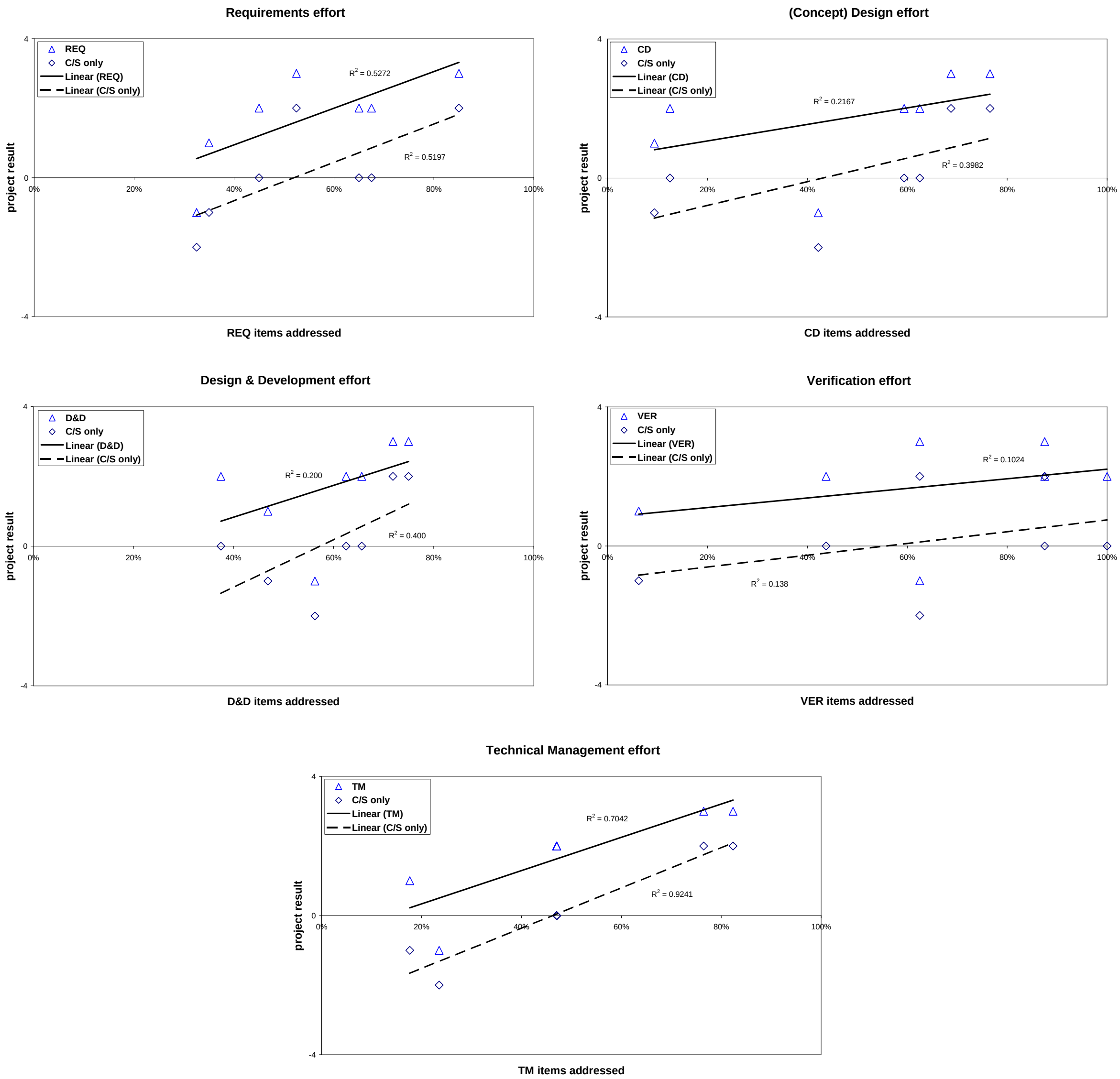

Figure 8 Correlation of Systems Engineering categories and project result.

Filtered results. One of the data points (projects) lies isolated from the others and scores systematically worse, so it may influence the conclusions disproportionally. As it is generally in the bottom left corner of the graphs, we may remove it; it will not cause the trend lines to be steeper.

There are also other, more objective reasons to exclude this project. It suffered from long delays due to a forced change of launcher, the funding had to be ensured by yearly negotiations and the concept changed from a man-tended instrument to a more autonomous mission. These circumstances necessarily led to a disproportional schedule slip and cost increase.

The results after removal of this project are shown in Figures 9 and 10.

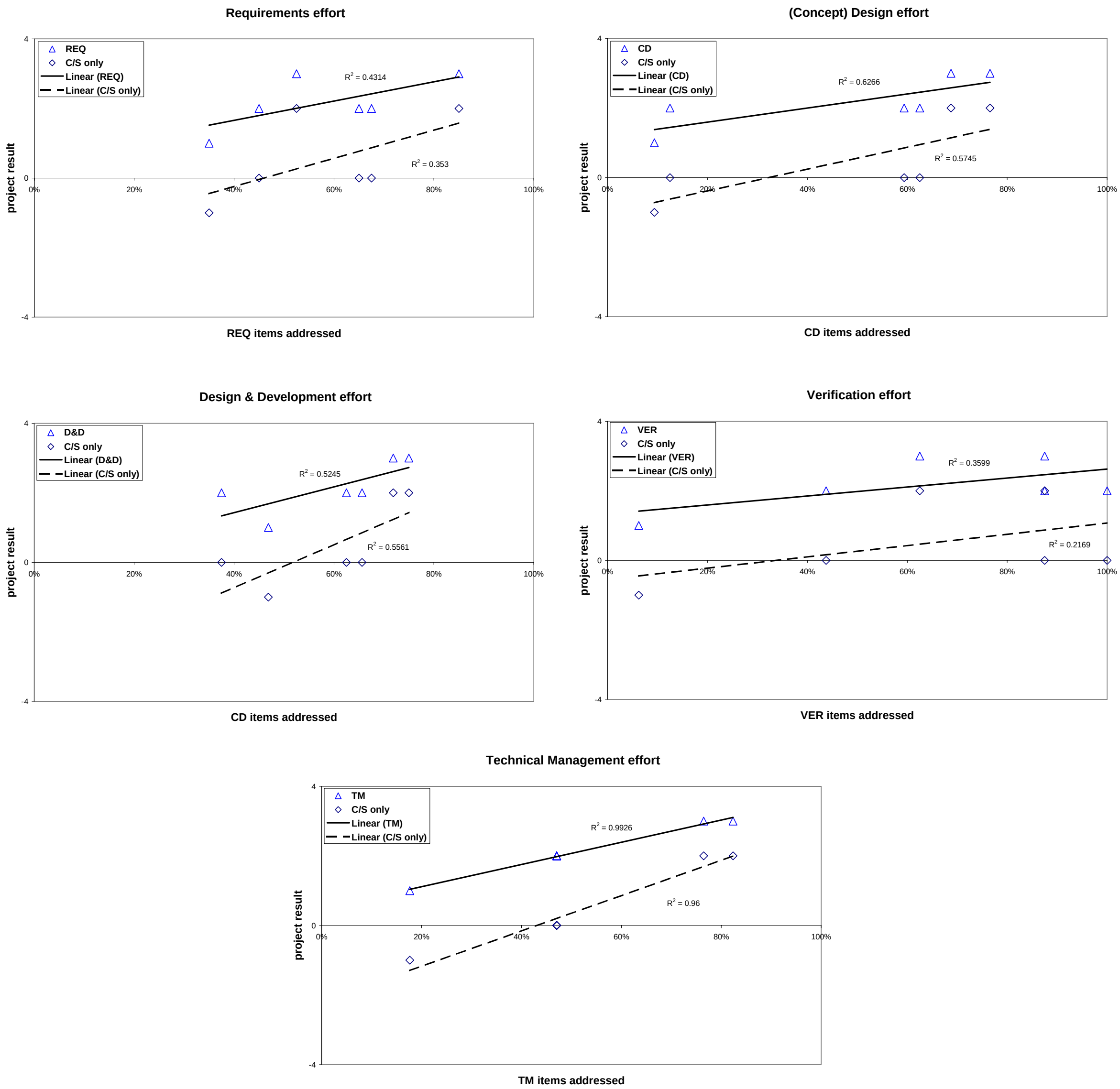

Figure 9 Correlation of Systems Engineering categories and project result – filtered results

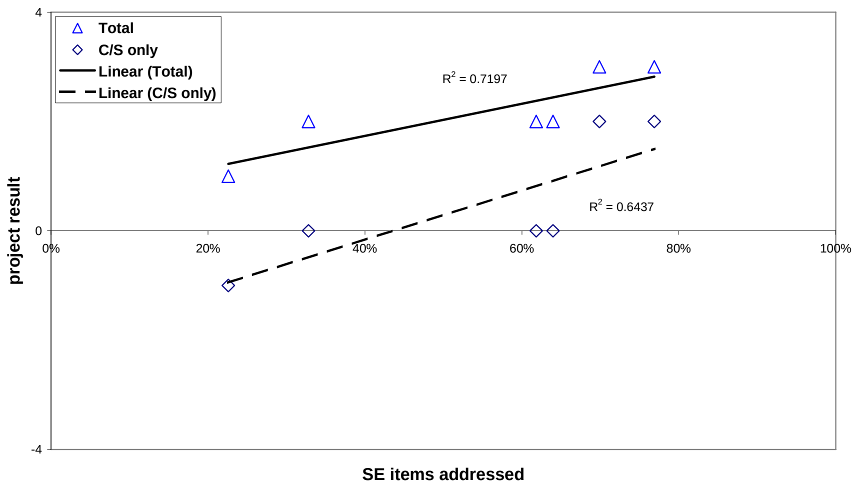

Figure 10 Overall correlation of Systems Engineering and project result – filtered results

The overall trend line (slope) does not change very much. The correlation of the data improves generally, except for the requirements related SE effort. Verification related SE effort does not seem to improve project result very much. There is a slight level change upwards, which is logical as the project that has been eliminated scored relatively bad. So we may repeat the careful conclusion that the categories Requirements, Design & Development and Technical Management contribute most to a positive project result and that their influence is strongest on cost and schedule.

Analysis

Comparison of trend line slopes. The slope of the trend lines have been collected in table 4, both for the unfiltered and for the filtered set. To allow for easier comparison the slopes have been normalized for the maximum slope in each set.

Table 4 Trend line slopes

Categories	*Absolute slope*				*Normalized slope*			
	Total–DC3	*C/S-DC3*	*Total filtered*	*C/S filtered*	*Total–DC3*	*C/S-DC3*	*Total filtered*	*C/S filtered*
Requirements	5.27	5.55	2.77	4.07	0.8	0.8	0.4	0.7
(Concept) Design	2.38	3.42	2.02	3.15	0.3	0.5	0.3	0.5
Design & Development	4.57	6.86	3.71	6.21	0.7	**1.0**	0.6	**1.0**
Verification	1.38	1.70	1.29	1.63	0.2	0.2	0.2	0.3
Technical Management	4.79	5.39	2.93	4.51	0.7	0.8	0.5	0.8
Total	**4.22**	**5.39**	**2.93**	**4.51**	**0.6**	**0.8**	**0.5**	**0.7**

The table emphasizes the finding that the Systems Engineering effort has its greatest effect on cost and schedule aspects of the project. SE activities in the category Design & Development contribute most to the project result. Apparently a decent effort in (technical) planning of the process always pays off.

Less addressed SE aspects. An inventory of Systems Engineering aspects that have only been addressed in less than 50% of the projects investigated shows the following picture.

- Identification of requirements without rationale and the establishment of design guidelines for availability, producibility and testability. The last point may indicate that established routine may be a ruling factor. This carries a certain, non-negligible risk.
- In the field of (concept) design a functional block diagram, showing relations and interactions between functions in the system considered and the higher-level system. This implies that the context of the project in its environment may not have been assessed properly.
- An overview of trade options considered, option selected, options retained as back up, options rejected and the related references. This represents a risk if the preferred option may not work in the course of a (sometimes lengthy) project.
- Generally characterization and/or standardization of interfaces. In some cases this may be explained by the prescription of such standardization by customer or prime contractor.
- The relation between design and development and assembly, integration and verification approach on the one hand and functional block diagram and product tree on the other. This potentially obscures the view on the reason why certain activities in this field are undertaken.

- Establishment of major system sensitivities. Again this may show a (over) reliance on established practices.
- Explicit iteration loops in the design and development logic. This increases the risk of "unexpected" schedule delay.
- Explicit interaction with –ility disciplines. This may jeopardize proper system synthesis.
- Records of design modifications and their sources (other than those included in standard configuration and data management systems, which include per definition always only a subset of changes).
- Compatibility of the Verification Control Document with the requirements traceability system and the Design Verification Matrix. This may, however, be justified for a small or relatively simple system.
- Planning of regression verification. This is not a very serious issue, if the system is mainly a hardware system without important software components.
- Explicit objectives and pass-fail criteria for internal & external design reviews in the field of Technical Management. It shall be noted that for external reviews the delivery and acceptance of project documentation specifies these items in general sufficiently.
- A system for a systematic check on data consistency and a fast and traceable system to record isolated analysis and design results. This may influence project efficiency in a negative way.
- Detailed work flow diagrams. Generally these are only used for large, complex projects.

Items well addressed. The good news is that the remaining tasks and activities are performed by more than 50% of the projects investigated. A list of tasks that have been performed by more than 70% of the projects is shown below.

- Requirements that are TBD or TBC, potential drivers or killers.
- (Technical) resource budgets.
- Layout sketches.
- ICD's defining interfaces to the customer.
- Milestones for reviews.
- Definition of activities as well as products in D&D logic.
- Product tree including specifics for each model.
- Verification Control Document identifying documents planned for verification.

Some cautions. It shall be noted that often missing items can be very well explained from the specific project characteristics, as are project complexity, number of personnel involved, project duration and project objective. And of course it is only a limited data set of six projects.

The available data in one case showed that planned and actually performed SE tasks (4 years later) were in good agreement, which is a positive aspect. In two other cases it was found that the assessment of SE tasks performed by project personnel that had been involved from the beginning of the project, was considerably more positive than the assessment by personnel joining the project in a later phase. This may show that possibly the planned effort has not been realized, or that the intended effort has not been conveyed convincingly to the team members, who joined the team later.

Conclusions

Even with all uncertainties and "noise" in the data we may conclude from this study that Systems Engineering has a positive impact on the result of the space projects examined. This agrees well with Honour (2002). The most important categories of Systems Engineering activities are related to requirements, (planning and management of) the design & development and technical management. The other categories have less influence, but cannot be neglected. It seems that the verification related Systems Engineering effort increases overall project result only by a small amount; however, this may be caused by the fact that a proper planning in setting up the design and development effort is sufficient for the smaller and simpler projects and that no "heavy" verification planning and control tool is necessary in this last project phase.

An increasing Systems Engineering effort appears to have most effect on cost and schedule performance (at least in the domain of space science projects). Technical quality is generally quite satisfactory (+1.36 on a scale of -2 to +2). As the projects are scientific, performance driven space projects this is not surprising, as there is no possibility of recovering faulty systems.

At this stage of the study no firm guidelines can be given for the optimum scope of SE effort, although some areas of improvement are identified. It is however emphasized that other factors, as are project character and size and team composition, play an important role.

Epilogue. This bit of research has been started in the expectation that no clear relation between the extent of Systems Engineering effort and the project result would be present. That assumption was wrong; although other project characteristics play a role too, the (noisy) data suggest that SE activities contribute clearly to a better project result, specifically cost and schedule performance.

Abbreviations

A	Feasibility investigation phase
C	Product definition phase
CD	(Concept) Design
C/S	Cost / schedule
D	Product realization phase
DC3	Delfi-C^3
D&D	Design and Development
ICD	Interface Control Document
-ility	Reliability, maintainability, availability, safety, producibility, testability, operability, human factors, etc.
REQ	Requirements
ROI	Return on Investment
SE	Systems Engineering
TBC	To Be Confirmed
TBD	To Be Defined
TM	Technical Management
VER	Verification

References

Boehm, B. 2003. COSYSMO: A systems Engineering Cost Model. In Proceedings of the 1st Conference on Systems Integration, Hoboken, NJ, USA

——— 2008. The ROI of Systems Engineering: Some Quantitative Results for Software-Intensive Systems. *Systems Engineering* 11 (3): 221-234.

Hole, E. 2005. Development of the ibm.com Interactive Solution Marketplace (ISM): A

Systems Engineering Case Study. *Systems Engineering* 8 (1): 78-92.

Honour, E. and Mar, B.L. 2002. Value of Systems Engineering – SECOE Project Report, In Proceedings of the 12th INCOSE International Symposium, Las Vegas, NV, USA.

Honour, E. 2004. Understanding the Value of Systems Engineering, In Proceedings of the 14th INCOSE International Symposium ,Toulouse, France

Project Handbook Standard, Chapter 2, Engineering, Issue 2. 2004. Company confidential publication, Dutch Space BV, Leiden, the Netherlands.

Biography

Robbert Hamann received an Aerospace Engineering education at Delft University of Technology in the Netherlands and Princeton University Graduate School, USA. From 1974 to 2000 he worked at Dutch Space (formerly Fokker Space), Leiden, the Netherlands as an Engineer and Systems Engineer for many space projects. From 1990 he has been in charge of introducing, developing and maintaining the Systems Engineering methodology at Fokker Space. Since that time he has been a visiting lecturer on the subject at the Delft University of Technology, the University Twente in the Netherlands, and the Ecole des Mines de Nantes in France. Since February 2000 he is employed at the Delft University of Technology as Coordinator Space Systems Engineering and Senior Lecturer. Recently he has been project manager of the first Dutch university satellite Delfi-C^3.

Barry Zandbergen received an Aerospace Engineering degree (1986) and a Space Systems Engineering (1998) education at Delft University of Technology. From 1986 to 1987 he worked at the ministry of Defense, The Hague, the Netherlands. Since September 1987 he is employed at the Delft University of Technology as assistant professor in the field of rocket propulsion 1987-1992, advanced launchers 1992-1996, and since 1996 in the field of space systems engineering and again rocket propulsion.

Philip Zijdemans received an Aerospace Engineering education at Delft University of Technology in the Netherlands. He has worked for about 30 years in Space industry, initially as a mechanical engineer, later as a systems engineer for solar array projects, until he became department manager. As such he was involved in the introduction and teaching of Systems Engineering, both in the company he works for (Fokker Space, currently Dutch Space), and, as an active INCOSE member, in other companies in the Netherlands.