The Impact of Faster, Better, Cheaper

Jairus M. Hihn

J

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California 9110

Introduction

At NASA our mantra has become Faster, Better, Cheaper

- Shift towards increased use of rapid development and concurrent engineering
    - Decreased percentage of effort and time spent up front
        - Less documentation, more informal CM
    - Increased time spent in implementation
    - Test time should reduce due to the early testing done as part of the demos

- Impact is
    - Decreased development costs
    - Increased cost over runs
    - Not clear what is happening to quality but suspect has decreased

- Software metrics data is becoming more blurred and more difficult to collect

DSN Productivity at Completion

- 8-10 SLOC/day ($75-$100 per line) in 80s and early 90s
- Increased to 15 SLOC/day ($50 per line) for 93-95
- 20-25 SLOC/day ($25 per line) in last 4 years

The metrics above are for new and modified lines of code counting modifications as new. A line is a physical line excluding comments. During this time period a virtual subsusytem was created called Multi Use Software (MSW). This provided a set of internally developed reusable software modules. The increase in development productivity over tha past ten years is a reflection of better tools, new languages, simpliefied development process, and indirectly the impact of a growing reuasable library and interface standards.

DSN Effort Distribution
(Percentage No. of Work-Months)

Activity	Rapid Development 1995-Present	Traditional 1980-1995
Management	11	14
System Engineering	13	11
Software Engineering	59	57
Test Engineering	17	19

Life-Cycle Phase	Rapid Prototyping 1995-Present	Traditional 1980-1995	Industry 1980's
Software Requirements	8	13	8
Software Design	16	23	20
Implementation	51	44	48
Integration & Pre-AT	24	20	24

Major Software Cost Over Run Sources (Flight SW)

Risk Area	Likelihood of Occurrence	**Mission 1**	**Mission 2**	**Mission 3**	**Mission 4**	**Mission 5**	**Mission 6**	**Mission 7**	**Mission 8**
Planning	63%	**X**	**X**	**X**	**X**	**X**			
SW Requirements	75%	**X**			**X**	**X**	**X**	**X**	**X**
Testing	63%	**X**	**X**	**X**			**X**	**X**	
Software Reuse	50%			**X**	**X**	**X**	**X**		
Tools, Technology, Infrastructure	50%	**X**	**X**		**X**		**X**		
Staffing	63%	**X**	**X**	**X**	**X**		**X**		
Mgmt. AND SE	75%	**X**	**X**	**X**		**X**	**X**	**X**	
Overrun		**25%**	**50-70%**	**50-70%**	**>30%**	**100%**	**25-35%**	**33%**	**0%**

- Cost performance based on plan at time of the beginning of Phase C/D excluding reserves.
- Data as documented is based on self reports of respondents only.

Key issues are pressure for over optimistic planning, failed attempts at reuse, staffing turn over, and lack of real understanding of SW by project managers and system engineers.

Major Software Cost Over Run Sources: Details (Flight SW)

Risk Area	Mission 1	Mission 2	Mission 3	Mission 4
MGMT AND SE	• PM & Flight SW Mgr did not understand SW dev as exp based on HW	• Need SE's who are programmers and can understand system issues	• SE's lacked understanding of SW • PM's lacked understanding of SW	•
Planning	• Planning does not accommodate SW	• IP continues to underscope their work	• Grossly under estimated	• When did major redesign we were rushed and produced an overly optimistic estimate • Scheduled determined by sys and ground SW without ever talking to Flight SW • Pressure to intentionally low ball • There were no processes or standards defined.
SW Requirements & Design	• SW has no mantra • SW Changes fast • SW gets committed to code before you ever see HW • Too many SS decisions made without sensitivity to how HW and SW play together • Lack of good arch.	• Need a design that tolerates defects. IP does not understand this.	•	• Lack of HW documentation
Testing	• Tesbeds were late	• IP does not have a real testbed where can test actual HW • Lack of system level fac.	• Insufficient testbeds • Better tools for analysis of test results	• Eng models kept breaking so could not test SW
Software Reuse	•	•	• Assumed could inherit 30% from classified projs and some from mission # 2	• Mission # 7 SW did not behave as advertised and many key people were not available to help us
Tools etc.	• CM is important • Tools not well integrated	• IP does not have enough tools	• Better tools for analysis of test results	• Docs not properly CM'd
Staffing	• Always understaffed • SW guys think are done when del to ATLO	• IP is currently overwhelmed	• Ind Partner understaffed and would not hire outside	• Many people in positions for first time (Newness) • Had high turnover at 1st

Major Software Cost Over Run Sources: Details (Flight SW)

Risk Area	Mission 5	Mission 6	Mission 7
MGMT AND SE	• I am at a disadvantage like many of my peers as I grew up with flight HW	• Inadequate SE to account for Sys impacts • Lack of adequate SW SE	• SE's need to know more about SW • How do you educate PM's to new ideas?
Planning	•	•	• (Newness mad it difficult to plan up front)
SW Reqs & Design	• Problem getting a good set of SW reqs • SW job not well defined up front • SW always trails HW	• sponsor added reqs with no money • When I came on there was no reqs baseline	• We had well defined end point but had to develop our own process to capture requirements through documenting problem statements. • JPL has no process for F,B,C
Testing	•	• Could not get enough test time on Radar	• Had really good test facilities but we never tested the best case scenarios
Software Reuse	• Assumed inheritance from Mission 2 but is not happening because people not available	•	•
Tools etc.	•	• Heterogeneous platforms and OS • When I came on there was a lack of CM	•
Staffing	•	• Analysts did code with no docs and then left JPL	•