How to keep good quality without increasing cost and time for development

Seiko Shirasaka
The Graduate School of System Design and Management, KEIO University
Self Introduction

• Academic background
  – Masters degree on Astronautics from University of Tokyo
  – Doctoral degree on Systems Engineering from KEIO University

• Working experiences
  – Work for Mitsubishi Electric Corporation as Space systems engineer for 15 years
  – Project 1: (ETS-VII) Engineering Test Satellite VII
  – Project 2: H-II Transfer Vehicle (HTV)
Self Introduction

– Work for Astrium GmbH (@Friedrichshafen) as Exchange Engineer (May. 2000 to January 2002)

– Project 3: QZSS
  Quasi Zenith Satellite System

– Associate Professor at KEIO SDM from April 2010
  • Member of “Hodoyoshi” project and “UNIFORM” project
1. Background

Reasonably Reliable SE

Performance-cost Curve

Cost Explosion:
- Complicated dual/triple redundancy, additional paper work, additional tests, additional human resource, expensive space rated parts, etc.

Governmental satellites

Nano-satellite design point

Less interfaces, proven technology

Standardization/process innovation

Actual reliability = Designed reliability x Probability that the system can behave as designed

- Modeling of various expertise and experiential knowledge
- Design methodology
- Application to different areas

Rather flat area where performance improvement can be achieved without much additional cost/workload

Performance/reliability

2012.10.11
Keywords

– Revisit (Japanese) traditional development style with the experience of the current systematic development style
  • Sense atmosphere
  • Understand each other with accountability
  • Follow rules and heart

– Design Based on Experience
Keywords

– Revisit (Japanese) traditional development style with the experience of the current systematic development style
  • Sense atmosphere
  • Understand each other with accountability
  • Follow rules and heart

→ Process Approach

– Design Based on Experience

→ Product Approach
3. Process Approach

• Revisit (Japanese) traditional development style with the experience of the current systematic development style
  • Sense atmosphere
  • Understand each other with accountability
  • Follow rules and heart

Sense atmosphere  Understand each other  Follow rules and heart

“空気を読む”  “場を読む”  “お作法と心”

“あうんの呼吸”
Sense atmosphere (“Ba”: Context)

- Sense and understand the context and decide what we have to do and what we don’t have to do.

Different Context
- Customer
- Contract
- Developer
...
Understand each other (A-Un)

• Understand what your partners (contractors, colleagues and even customers) do and coordinate who will do what. (usually implicit)

• Utilize visualization technique for accountability (“visualize what you or your partners think)
Follow rules and heart

- Rules have objectives (Heart). Always think about objectives and decide whether we have to follow the rules or not, and how we follow. (We can skip a step if it meets objectives.)

Ex. Skip formal design reviews if it can be covered by daily check.
4. Product Approach

• Design Based on Experience
Design Based on Experience

• Low failure rate of COTS parts and components
  – Utilize COTS parts

• Low “parts/component” failure rate compared with “Systemic failure” (design errors)
  – Simple design to reduce systemic failure
  – Maximize software utilization (we can fix it on board and we can make “exactly” same one.)

• Normal redundancy can cover just “random failure”. (Only 20% onboard failure is covered (Saito, 2010))
  – Use “power cycle (off-on)” to reset failed components.
5. Conclusion
Keywords

– Revisit Japanese traditional development style based on western systematic development style
  • Sense atmosphere
  • Understand each other with accountability
  • Follow rules and heart

– Design Based on Experience
Thank you for your attention
What new?

Current Approach
- Standard Process Approach (ISO, IEEE, ECSS etc.)
- Modular Design or integral design

New Approach
- Standard Meta-Process Approach
- Integral Product Assurance
  - Integral: 擦り合わせ
Standard Meta-Process Approach

- Each Process has objectives.
- Decide which process we perform and which do not and how we perform it in accordance with objectives and system and context. ("Ba")
- A set of objectives (or meta level process) is “Standard Meta-Process”.

![Diagram showing the relationship between objectives and processes]

- Standard Meta Process
  - Objectives 1
    - Process 1
  - Objectives 2
    - Process 2
  - Objectives 3
    - Process 3

Standard Process
Standard Meta-Process Approach

Example: Design Review

• **Current**: Just do Design Review as Usual

• **Our Approach**
  
  • Consider
    
    • What is the objectives of this design review?
    
    • Who is designer, how about his or her experiences?
    
    • Repeated design or new design?
    
    • Customer care the documents?
    
    Etc
  
  • And decide / clarify
    
    • Whether we should do that review, why we do, why we do not?
    
    • Who should be reviewers?
    
    • How should we do?
Integral Product Assurance Approach (A-Un)

Project Level Approach

• Decide what we do and what don’t in accordance with the coordination with stakeholders (customers, systems engineers, subsystems engineers and component engineers.)

Program Level Approach

• Coordinate between several projects to perform on-orbit verification. (TRL)
Example: Current

3. Process Approach

Example of Normal Process

Integration
- STR/EPS
- COM/DH
- AOCS

System Functional Test

Vibration Test
Acoustic Test
Thermal Vacuum Test
Long Duration Test
Burn-in
Example: New

Only functions which are not verified at system test will be tested at this level.

Example of Integral Process
• **Current Reliability**
  
  Calculated from Random Failure of Parts

However, several research results show that random failure is not dominant and many of onboard failures are temporal failures not permanent failures.

• **Introduce new indicator : Probability of Mission Continuity**
Product Approach: Categorization of Failure

- Permanent Failure
  - Random Failure
  - Systemic Failure

- Temporal Failure
  - Random Failure
  - Systemic Failure
### Product Approach: Counter measures for Failure

#### Categorization of Failures

<table>
<thead>
<tr>
<th></th>
<th>Counter measure</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Normal Redundancy</td>
</tr>
<tr>
<td>Permanent Failure</td>
<td></td>
</tr>
<tr>
<td>Random Failure</td>
<td>O</td>
</tr>
<tr>
<td>Systemic Failure</td>
<td>X</td>
</tr>
<tr>
<td>Temporal Failure</td>
<td></td>
</tr>
<tr>
<td>Random Failure</td>
<td>O</td>
</tr>
<tr>
<td>Systemic Failure</td>
<td>X</td>
</tr>
</tbody>
</table>

#### Notes:
- **O**: Indicates the counter measure is effective.
- **X**: Indicates the counter measure is not effective.
- **O/X**: Indicates mixed effectiveness, depending on the specific failure type.
• Current Reliability
  Calculated from Random Failure of Parts

• Probability of Mission Continuity
  Calculated from the combination of Temporal Failure and Permanent Failure

Change Viewpoint
Probability of Mission Continuity

- Probability of Mission Continuity

\[ R = \exp(-\lambda_1) \cdot \exp(-(1-a)\lambda_2) \]

- Example: Increase current reliability by redundancy

- Example: Probability of Mission Continuity

\[ a : \text{Recovery Rate, } \lambda_1 : \text{Permanent Failure Rate, } \lambda_2 : \text{Temporal Failure Rate} \]
Probability of Mission Continuity

When temporal failure rate is small, the redundancy makes probability of mission continuity increase. But when temporal failure rate is high, the recovery function makes probability of mission continuity increase.
System Architecture design with the consideration of recovery function

Architecture for recovery

Communication

Data Handling

Power

Architecture with the recovery function

Attitude Control

Payload

Others