

# New TSP Paths

Systems Engineering Team uses TSP

Daniel Wilson  
Wyle, Inc.  
E-2C/PSA Systems Engineer

NAVAIR Public Release 10-(882)  
Distribution: Statement A- "Approved for public release;  
distribution is unlimited"

“Organizing is what you do before you do something, so that when you do it, it's not all mixed up.”

--A.A. Milne

# Who are we?

- E-2C PSA
- PMA-231
- NAVAIR
- US NAVY



- Develop and maintain software for the E-2C Hawkeye

# E2C PSA

- Program Support Activity
  - MC : Software Engineers
    - 7 years Experience with TSP
  - ACIS : Software Engineers
    - 7 years experience with TSP

# Systems Engineering (SE) : Software Test/Evaluation

- 12 Engineers (2 on Project)
- Avg. Age : 53
- Avg. Years Experience : 28
- TSP Experience : ZERO

# Initial SE Thoughts

- “My work doesn’t have defects.”
- “I’m going to have to account for every hour?”
- “This may be fine for programming, but it isn’t going to work with what we do.”
- “Why? What’s wrong with the old way?”

# Why TSP?

- Customer wanted System Engineers to plan and track our work and he sees TSP reports from other teams.
- Constant suggestion from TSP coach Mr. Orville Starnes.
- Decision from supervisor Mr. Kerry Camm
- Pressure to do more with less

# The Old Way

- Individual effort/Single writer
- Old document template
- Always done it “this” way
- Write/test/rewrite (repeat as necessary)
- 80 percent complete!
- Finish right before formal Product Evaluation



# The Project

## Automatic Identification System (AIS)

- Combined Requirements Launch (MC, ACIS, SE)
- Software Development Launch (MC)
- Combined System Engineering and ACIS Software Development Launch. Separate projects with a combined team meeting.

# Requirements Launch

- SEs participated in ACIS/MC TSP Project.
- Authored only high level requirements.
  - SEs became familiar with inspections and the TSP process.
- Primarily Inspectors for MC/ACIS detailed requirements.
  - Built communication and trust/established relationships
  - Initiation of TSP and processes involved for SE
  - Early influence on requirements
  - Early influence of operator perspective

# System Engineering Launch

“Prediction is difficult, especially about the future.” - Yogi Berra

# System Engineering Launch

- How do we size? or What is a SLOC?
  - Pages/Steps
- What are our tangible assemblies/products?
  - Bench testing tasks
- How do we turn a document into tasks?
  - Break a single document into pieces
- Is our requirement breakdown similar to ACIS?

# First Inspection

- We don't have a review checklist
- We don't have an inspection criteria sheet.
- Why is there front matter on this test case?
- What font did you use?

# Lessons Learned

- TSP does help you plan and track your work.
- TSP also forces a team to formalize process.
- TSP aligns separate teams on a project.

# Working with ACIS

- Courtesy invitations to inspections
  - Amazing insight into development
  - Opportunity to catch problems earlier
  - Helpful inspection training from “The Experts”
  - Better alignment of current tasking across teams.

# Teams Interacting

- We argued early and often!
  - This was a good thing.
- Able to discuss minor technical issues during weekly team meetings.
- Knowledge of who to go to for answers.



# Problems

- Not enough people to fill TSP roles.
- Lack of TSP support structure (checklists, inspections etc.)
- Lack of understanding on what the collected data means.

# Team Results

- Test Cases finished prior to the first drop of software!
- Forced a standardization of products with templates, personal reviews, and inspection criteria.
- Increased individual “buy in” by giving each a portion to develop.

# Project Synergistic Results

- Increased productivity noted for ACIS in post mortem.
- Earlier problem identification
- Alignment of effort by requirements breakdown
- Shared understanding of effort through inspection attendance

# Program Success Overall

- ACIS SLOC : 15000      MC SLOC : 28000

Forty Three Thousand Lines of code

Software Trouble Reports

Since first Engineering release?

**Five**

# Product Evaluation Test

- Software trouble reports at final product evaluation test?

**ZERO**

# Post Mortem SE Thoughts

- “I gotta admit, this new way of creating software test documents is a whole lot better than the old way.”

# Lessons Learned

- Document estimate size would have been more accurate if we had used steps rather than pages.
- Formalizing the requirements breakdown to individual test case during requirements development.
- Continue using TSP!

# Advice for Others

- Build your TSP support structures ahead of time.
- Be ready to reorganize your tasks based on success.

“Insanity: doing the same thing over and over again and expecting different results.”

**Albert Einstein**

- Don't be insane, try something different.