https://ntrs.nasa.gov/search.jsp?R=20180005402 2019-04-29T09:26:47+00:00Z

Hybridized Agile Software Development of Flight Control Team Tools for International Space Station's Payload Operations Integration Center

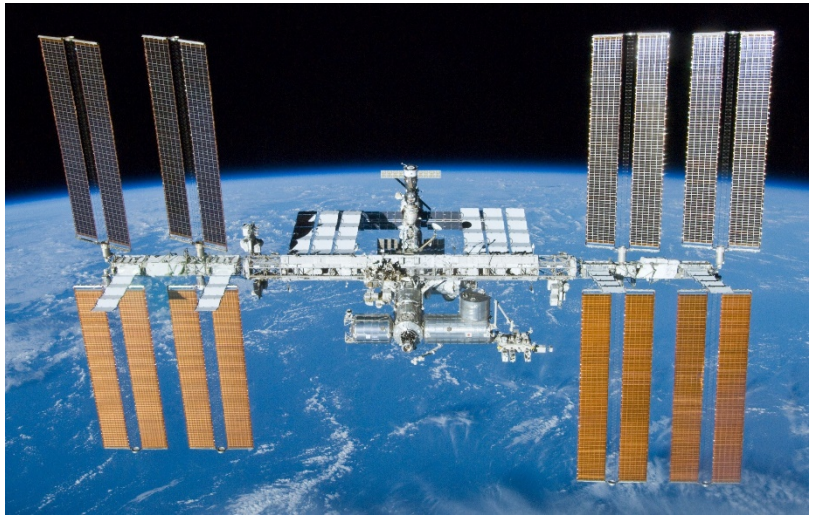
Cerese M. Albers, Ph.D., PMI-ACP

Software Development
Build Coordinator & Agile Consultant
COLSA Corporation, Huntsville, AL, USA
NASA MSFC HP27

Human Exploration Development & Operations Office

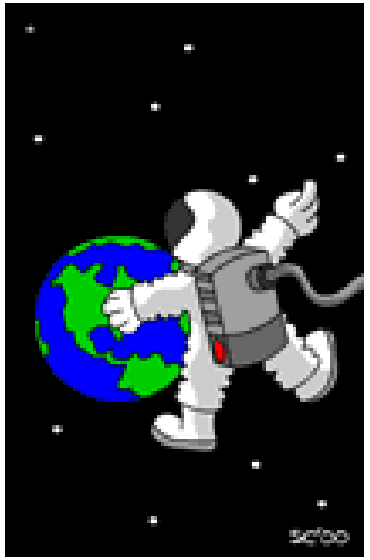
POIC @ MSFC

Supports Payload Science on ISS

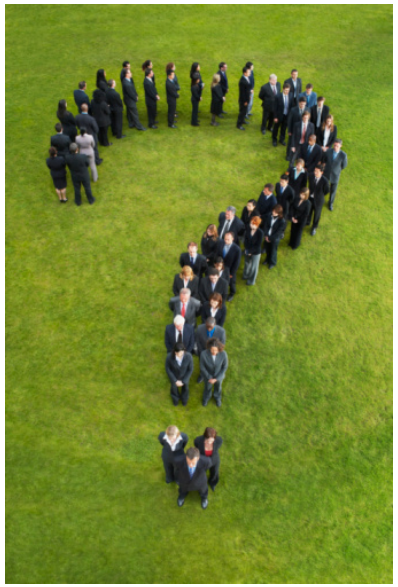


+

4th Crew Member



=



Need Tools so Flight Control Team Can Support More

Need for Hybridized ASD for FCT at ISS POIC

The Driver:

In mid-2014, ground systems support personnel at NASA MSFC's Mission Operations Laboratory (MOL) identified the need to support an anticipated large increase in payload science experimentation time due to the upcoming ability to staff an additional crew member aboard ISS.

- The "fourth crew" member provided the opportunity to achieve a higher ISS return on investment.
- Kicked-off the High Operations Tempo (HOT) initiative, including developing 4th Crew Tools.

Short Time To Completion:

2017 start of this tempo increase to around 100 hours of crew-time payload science activities per week.

Why ASD:

The HOT tools had to be developed quickly, incorporate ongoing user feedback, and provide a complete and useful solution *the first time* each was delivered. There was little or no time to accommodate feedback during operational use before the tools would become highly necessary.

- Agile Software Development (ASD) was chosen as the best ideological approach to employ, but it had never been applied at the Payload Operations Integration Center (POIC) before, which was a predominantly Waterfall Software Delivery environment. There was a need for hybridization of ASD.

Top 10 Challenges to POIC HOT Tool Development

1. Product Team (PT) & Development Team (DT) not required to understand each other's work
2. Don't speak the same language- implementing vs designing/building
3. Work schedules can be quite different between the two teams
4. DT members typically work on more than one project at once
5. Infrequent Ops and DT interactions, so motivations for decisions not well-understood
6. Tool need clear, but full scope & user experience not well-understood; requirements will change
7. ConOps only reflects Ops understanding of what may be possible, and use cases may inadvertently lack information crucial to DT understanding for implementation
8. DT must design and build software to requirements with little or no Ops feedback until in use, yet new tool must fit the need on first release
9. Software release process inflexible to quick change requests: releases occur at biannual ground transitions and patches are disruptive to operations and testing schedules
10. Lack of Human Factors input means Ops use of software is potentially non-intuitive or fatiguing

Agile is first and foremost a MINDSET, not a set of prescriptive tools and processes. It's predominantly a shift in values:

The Agile Manifesto

"We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions	over	processes and tools
Working software	over	comprehensive documentation
Customer collaboration	over	contract negotiation
Responding to change	over	following a plan

That is, while there is value in the items on the right, we value the items on the left more."

http://agilemanifesto.org/principles.html

Agile Triangle

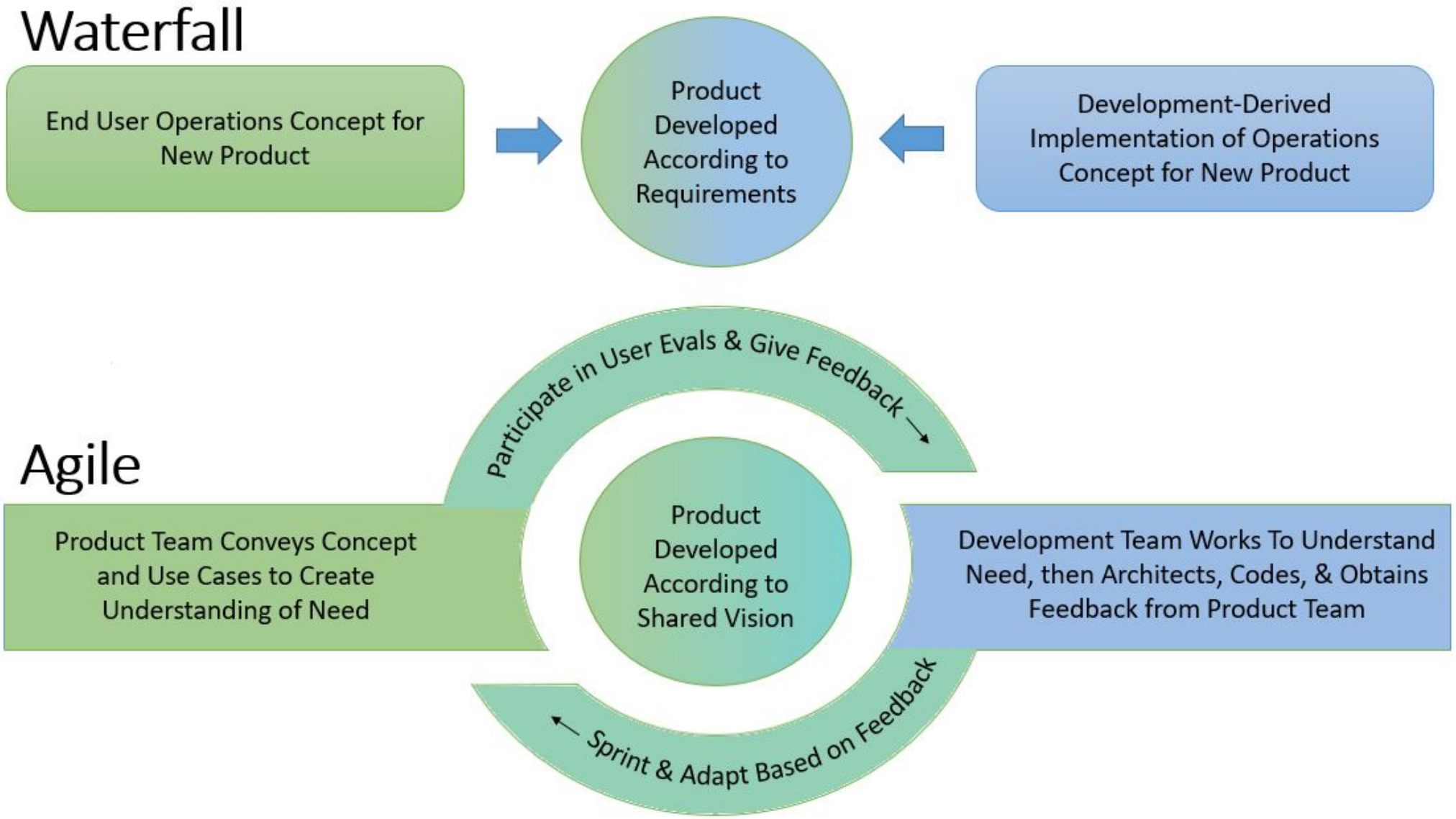
Value

http://agilecodex.com

Quality

Constraints

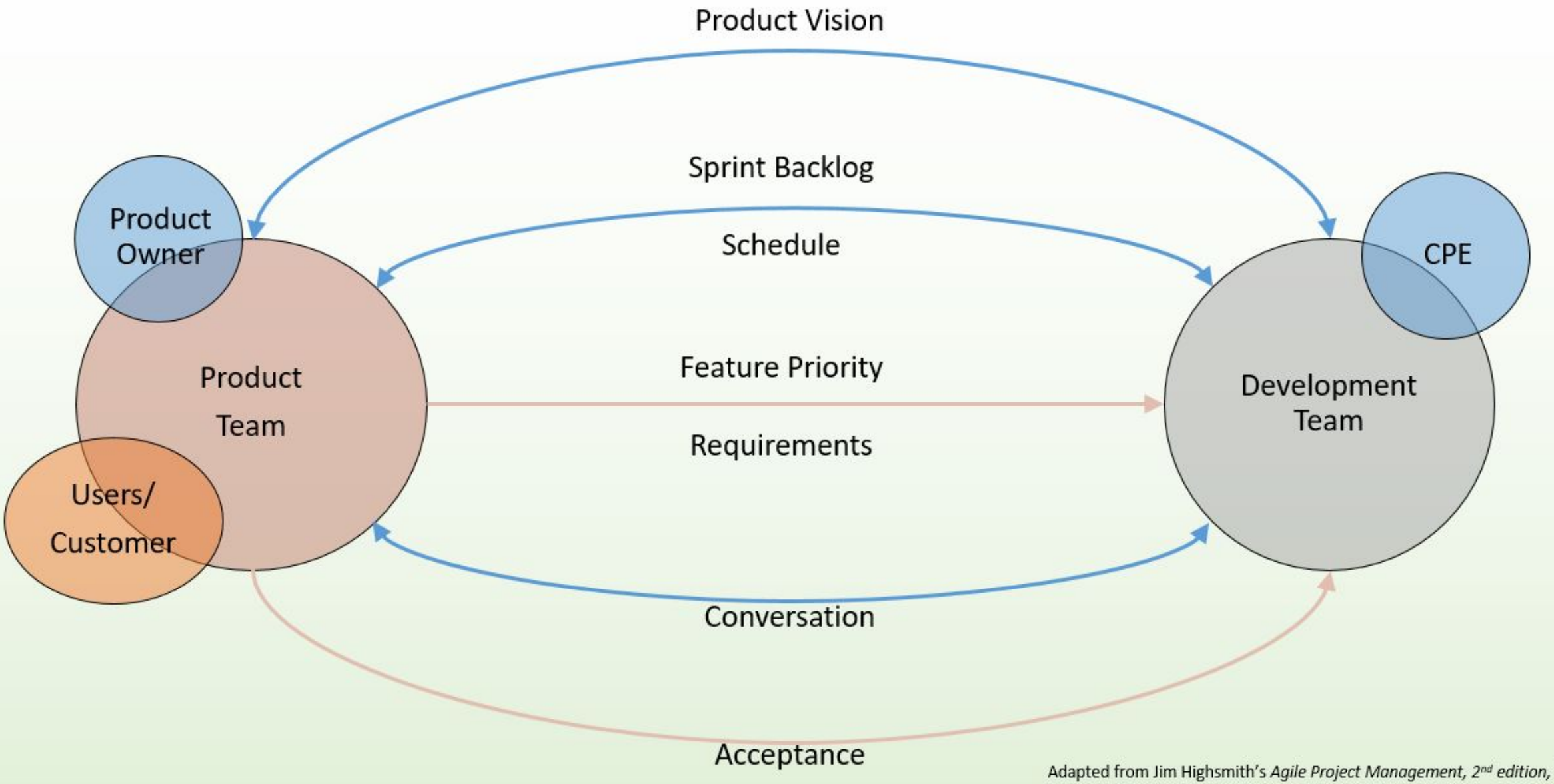
Product Benefits from an Agile Approach



How ASD Overcame the Top 10 Challenges

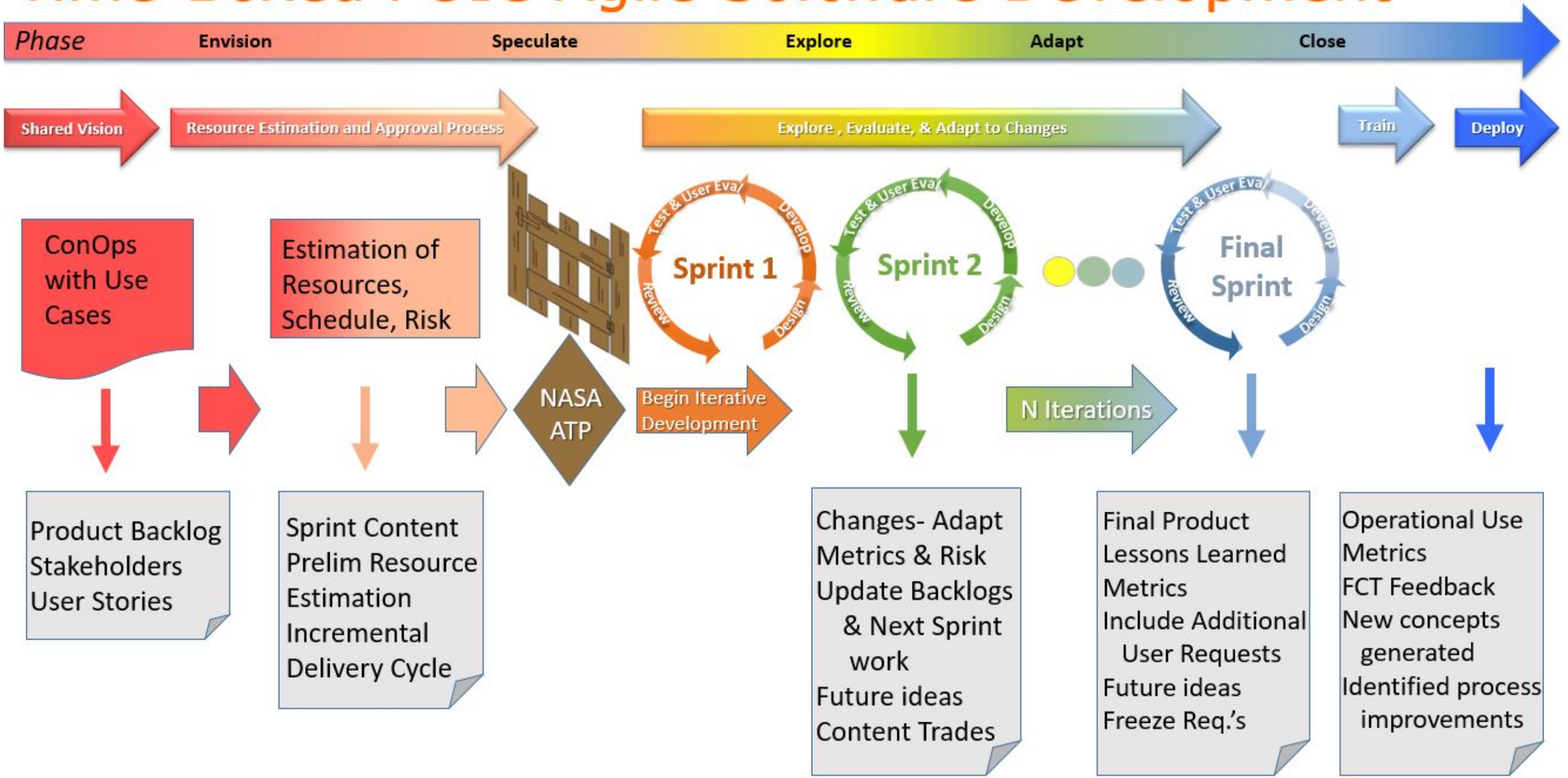
1. Product Team (PT) & Development Team (DT)- **Interact frequently to create shared product vision.**
2. Start speaking the same language- **Cross imaginary organizational borders and have developers visit ops environment, and involve users in early tag-ups to pre-evaluate software as its being built.**
3. Work schedules quite different- **Willingness to meet at odd times of day, maximize meeting time.**
4. DT members typically work on more than one project at once- **Work with team leads to free up resources for a defined period of time for the project. Consider code deliveries outside of normal schedules.**
5. Infrequent Ops and DT interactions- **Increase collaborative work and social activities and hold sprint retrospectives in a relaxed and fun atmosphere.**
6. Full scope & user experience not well-understood- **Embrace that requirements will change!**
7. ConOps only reflects Ops understanding- **Host a Sprint Zero and create a shared vision and prototypes.**
8. Little Ops feedback until in use- **User Evaluations to get feedback early and often during development.**
9. Software release process inflexible: **ASD is responsive to requirements changes and customer requests.**
10. Lack of Human Factors input- **Involve HF reps early in tool development phase.**

Product Team-Development Team Interface

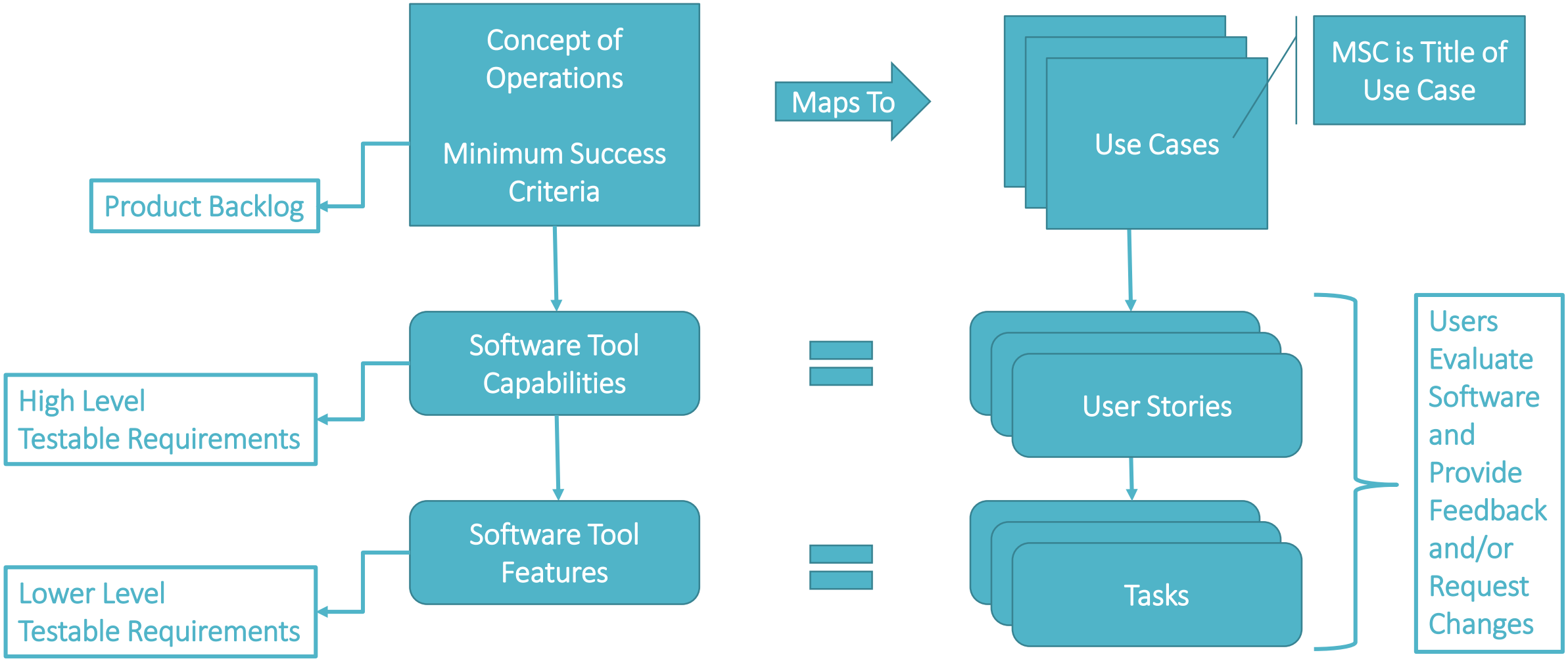


Adapted from Jim Highsmith's *Agile Project Management, 2nd edition, © 2010*, p. 119

Time-Boxed POIC Agile Software Development



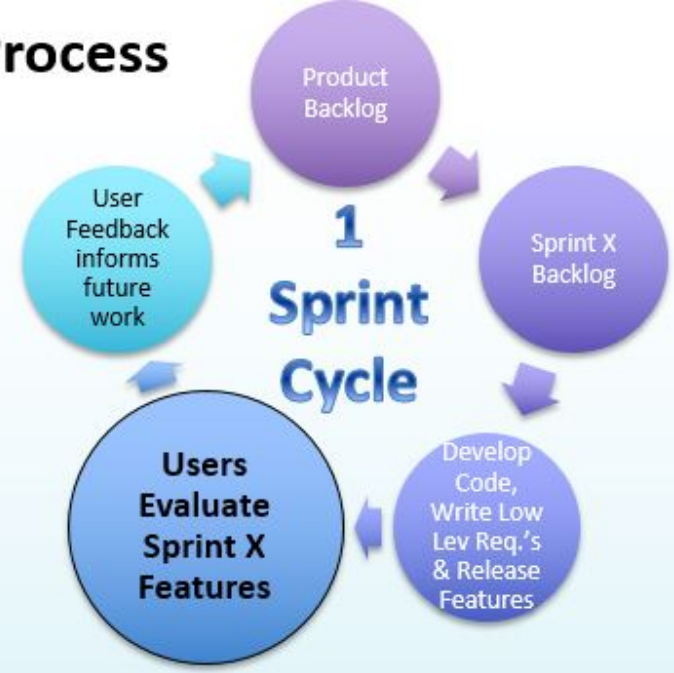
Hybridized ASD Way of Defining Requirements in a Predominantly Waterfall Delivery Paradigm

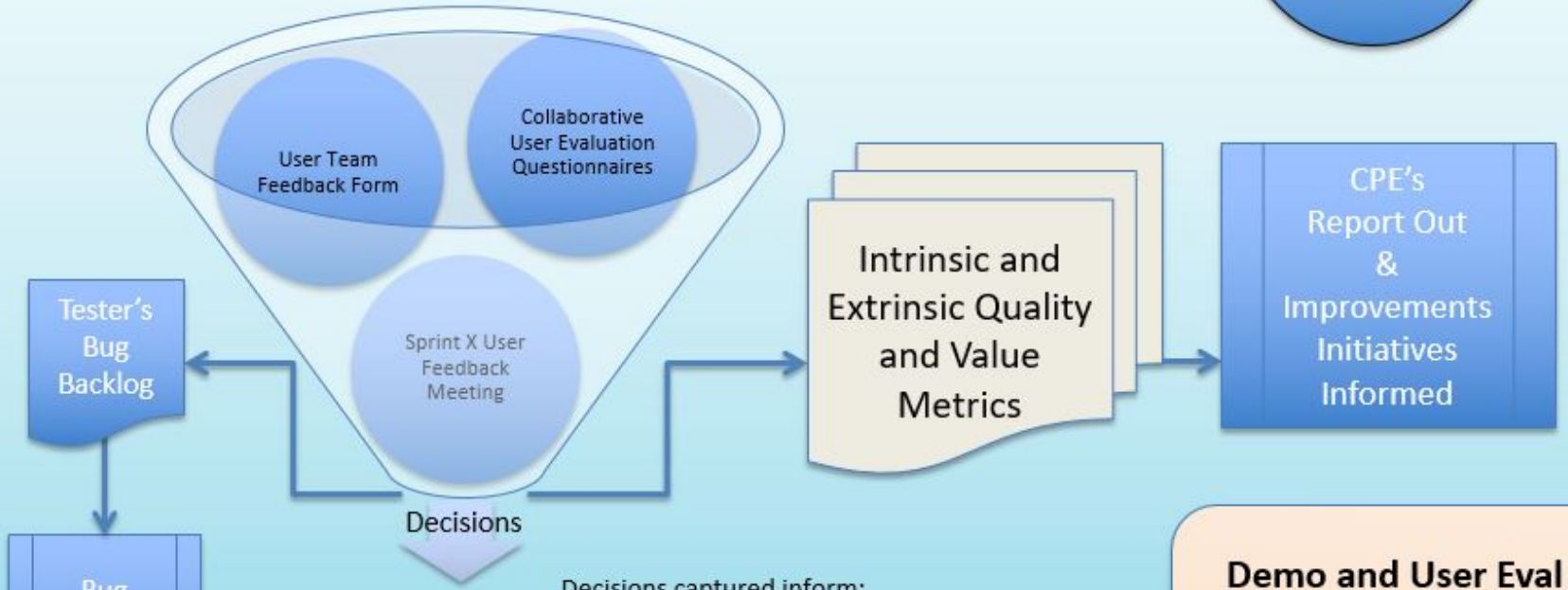


POIC Agile User Evaluation Process

User Evaluation Week of Sprint X

Demo	Users "Play" with Sprint X features in the Eval Environment and meet to prioritize feedback	User Eval Feedback Meeting
1 HR	← 5-7 Business Days →	1.5 HR





User Feedback Informs Future Work

Decisions captured inform:
1) Future Sprint work & updating backlogs & ConOps;
2) Minor changes to this Sprint's features;
3) Changes to resources requiring approval;
4) Software bugs to fix;
5) Synergies and interoperability with other tools

Demo and User Eval Attendees:
CPE, Product Owner, Dev POC, Testers, Security rep, Trainer, Human Factors, Developers, Users

Growing Pains

Obstacle	Solution
Conveying operational use cases and defining expectations	**Tabular distinction of MSC, Highly Desired, Nice-to-Have Capabilities**
Changing expectations for project approval process	**New Change Package guidelines**
Time commitment to user evaluations	**Getting the same evaluators across multiple sprints and multiple console positions**
Need dedicated eval environment separate from Dev, Test, Sim, & Ops	**EVAL- Environment for Value Assessment and Learning**
Simulating usage scenarios to really test-run a tool	**USIMs (User Evaluation-Style Simulations)**
Increasingly complex automation projects	**Introduced Sprint Zero concept**
Phasing releases, FCT ability to absorb tool via time for training around competing flight objectives	**Metrics of time investment vs time savings; cost-benefit analysis Tool list reprioritization based on ability to support ASD process**
Defining regular interactions bridging the two organizations	**SOP creation and approval**

Declaring Success: 5 Customer-Valued FCT Tools Created Using POIC ASD in 18 months

1. TIPS, Timeline Integration Product Summary
   - Automates near real-time consolidation of planning info for payload and crew activities
   - Tremendous efficiency gains: from 40 hours to 40 minutes to create updated reports
2. COMMDASH, Communications Dashboard
   - A one-stop collection of 5 comm apps with user-specific customizable views, leveraging social media concepts to declutter voice loop traffic and facilitate operational awareness
3. PD Status, Payload Developer Status application
   - Displays active and upcoming payload activities and status to console operators in order to allow for support preparation and resource allocation
4. FCTL in CoLT, Flight Control Team Log in Console Log Tool
   - Allows console operators to push their log entries to a single commonly viewable team-level log when info affects three or more positions
5. SMARTSearch
   - Allows highly customizable search of cross-center NASA internal sources for information useful to the FCT

How ASD Helped Achieve Success

HOT Tools projects successfully used the ASD paradigm because it:

- Helped develop special tools to do specific jobs despite the customer not having the advance ability to know exactly how they would specifically want to interact with the tools.
- Permitted the POIC to take advantage of uncertainty, and plan for it, by providing a process that facilitated rapid and flexible response to changing requirements.
- Changed the development and release perspective from prescriptive to adaptive.
- Provided an avenue for strategic investigation and exploration of new technologies.
- Incorporated customer feedback throughout the product development lifecycle and allowed for continuous quality improvement of each tool so that the final products were released on time as useful, efficient, and user-friendly applications of value.

Related Papers at SpaceOps 2018

Marrying Social Media Approaches and Space Flight Control – Eight Years at SpaceOps

David W. Scott, Dr. Cerese M. Albers, Hugh Cowart, Andrew J. Nichols, Rob L. Roy

30 May, 1200-1230, Estaque

In OC-04. OC – Mission Operations Concepts

Innovative Development of a Cross-Center Timeline Planning Tool

Ramon Pedoto, Cerese M. Albers, David Benjamin, James Reynolds

29 May, 1700-1730, Notre Dame

In PS-02. PS – New Techniques and Planning Software II

Thank You!

4th Crew Tools Cross-Disciplinary Team Members