Innovative Development of a Cross-Center Timeline Planning Tool

Ramon W. Pedoto,[1] Cerese M. Albers[2] Ph.D., PMI-ACP, David N. Benjamin[3], and James F. Reynolds[4]

[1]COLSA Corporation, Huntsville, AL, 35806, USA, ramon.w.pedoto@nasa.gov
[2]COLSA Corporation, Huntsville, AL, 35806, USA, cerese.m.albers@nasa.gov
[3]NASA/MSFC, Huntsville, AL, 35812, USA, david.n.benjamin@nasa.gov
[4]NASA/MSFC, Huntsville, AL, 35812, USA, james.reynolds@nasa.gov

The Payload Operations Integration Center (POIC) at Marshall Space Flight Center (MSFC) supports planning, coordination and scheduling of science activities for the International Space Station (ISS) in coordination with other NASA centers, international partners, and payload developers. The ability to efficiently plan and re-plan in response to change is critical to the flight planning teams. With the achievement of supporting a fourth crew member aboard the ISS and an increasing amount of payload science activities, came the need for a dynamic, more efficient way of building timeline planning reports that could be readily updated as fast as payload science plans could change. This paper addresses software architecture considerations in the successful cross-center development of an automated planning tool with multiple data sources. It also discusses the practical implementation of a time-boxed, hybrid Agile Software Development (ASD) approach to deliver customer-driven value despite changing requirements with respect to low-Earth orbit operational planning activities. The goal of this paper is to open discussion with members of the international community and trade effective strategies for cross-center architectural and customer-developer driven collaborations, to support increasing utilization of planning and conducting science activities in space.

I. Introduction

The Payload Operations Integration Center (POIC) at Marshall Space Flight Center (MSFC) is the United States focal point to support operations controllers and payload developers conducting payload science operations for the National Aeronautics and Space Administration (NASA) aboard the International Space Station (ISS). Some of the key functions are planning, coordination and scheduling of science activities. This effort occurs in coordination with other NASA centers, international partners, and payload developers. The need to efficiently plan and re-plan, in response to various changes, is critical to the flight planning teams.

NASA has recently increased its ability to perform payload science operations aboard the ISS with the addition of a fourth crew member. In the summer of 2015, the POIC and NASA MSFC recognized a need to quickly develop an automated tool to support the additional planning and scheduling needs brought about by this increase of payload science activities. In the short time that followed, planning team members created a new concept for operations that was shared with the POIC software development team. The vision for the Timeline Information Planning Summary (TIPS) [1] tool was to answer that request for automation via a collaborative effort between the developers and customers.

[1] Software Architect, MSFC HP27
[2] Software Build Coordinator and Agile Consultant, MSFC HP27
[3] POI Timeline Change Officer / Payload Planning Manager, MSFC HP23
[4] POI Timeline Change Officer, MSFC HP23

II. The Necessity of Automation

In the past, Payload Planning Managers (PPMs), in the Short Term Planning (STP) Room, would initialize documents that summarized the planning and scheduling information obtained from the Johnson Space Center (JSC) Operations Planning Timeline Integration System (OPTIMIS) [2]. Members of the Product Readiness and Enhancement for Payloads (PREP) Team would then maintain and update these documents, including adding discipline-specific notes to assist in the efficient execution and support of planned operations. This information is crucial for the Flight Control Team (FCT) in the Payload Control Area (PCA-1) to understand why activities are planned in a specific order and the detailed constraints between activities, during real-time execution.

This lengthy process of updating static documents while planning and re-planning was very inconvenient, introduced human error, and was inflexible to last minute changes. Therefore, there was a need for a dynamic, more efficient, less erroneous, and more concise way of building a report that could be readily updated as fast as payload science plans change.

A. Planning Information Flow

To efficiently create a consistent and effective daily planning report framework, the TIPS tool provides a collaborative automated solution for users to create and maintain summarized information for operations execution reference. In the past, Payload Planning Managers (PPMs) initialized reports with planning and scheduling information from scratch. The PREP Team maintains and updates this scheduling information, as well as adds discipline-specific knowledge for the efficient execution and support of planned operations. This information handover is required to pass notes from the Short Term Planning (STP) Room to the PREP Room, and finally to the real-time execution team in the Payload Control Area (PCA-1). This information is crucial for the FCT to understand why activities are planned in a specific order and the detailed constraints between activities.

Much of the information required for a TIPS report exists in planning software, payload overviews, operations Guidelines and Constraints documents, activity requirements collection and documentation, console logs, and corporate knowledge applied by discipline-specific experts. TIPS allows users to define parameters for the report and store that preference; identify and retrieve the associated information from OPTIMIS Plan Repository; retrieve and display changes made to Plan Repository data through a change request web service; allow users to supplement the findings with additional information; filter all the results in a selectable format (e.g. daily, weekly); and provide the data in a consistent, expandable/collapsible framework. STP and PREP teams continually add and update the information to the ISS Onboard Short Term Plan (OSTP) and other operations activity information. TIPS consolidates the dynamic data into a single, consistently formatted summary report, as opposed to the manual process that was used before. These reports aid in timeline reviews and provide important handover information to the FCT.

B. TIPS Application Features

The TIPS application provides the user with a method to create/view a report of planning activities for a specified period. There are three main views in the application: the Calendar view, a Daily Report view and a Weekly Report view. Certain editing capabilities are available in each view depending on the user's permissions.

In the TIPS Calendar view, the user is presented with a month-long interactive calendar. For each day of the month, a privileged user can create one or more reports. Non-privileged users can only select existing reports to view. Simple navigation to previous or future months is available. The calendar view also allows the user to bring up the weekly view by clicking the "Weekly" hyperlink to the left of each week row. The primary benefit of the Calendar view to the user is the interactive ability to quickly see which reports have been created for an entire month, with custom report names, or to create new ones as needed from the same GUI.

The TIPS Weekly Report view shows a summary of the daily TIPS reports for seven days. Figure 1 provides an example for a week beginning Monday, March 12, 2018. TIPS activities, and containers of activities, are colored in accordance with the planning repository information that is ingested, and has meaning for the sequencing and execution status of payload science activities. The weekly view also permits the user the option to toggle between multiple reports created for the same day, of different origin. The weekly view presents quick links to access other TIPS software capabilities of interest to the FCT, such as summations of crew time and a task list. Each time the user navigates between reports or refreshes, TIPS presents a new amalgamation of the latest information available from the combination of data sources (discussed in the Data Access part of this paper's Architecture section).

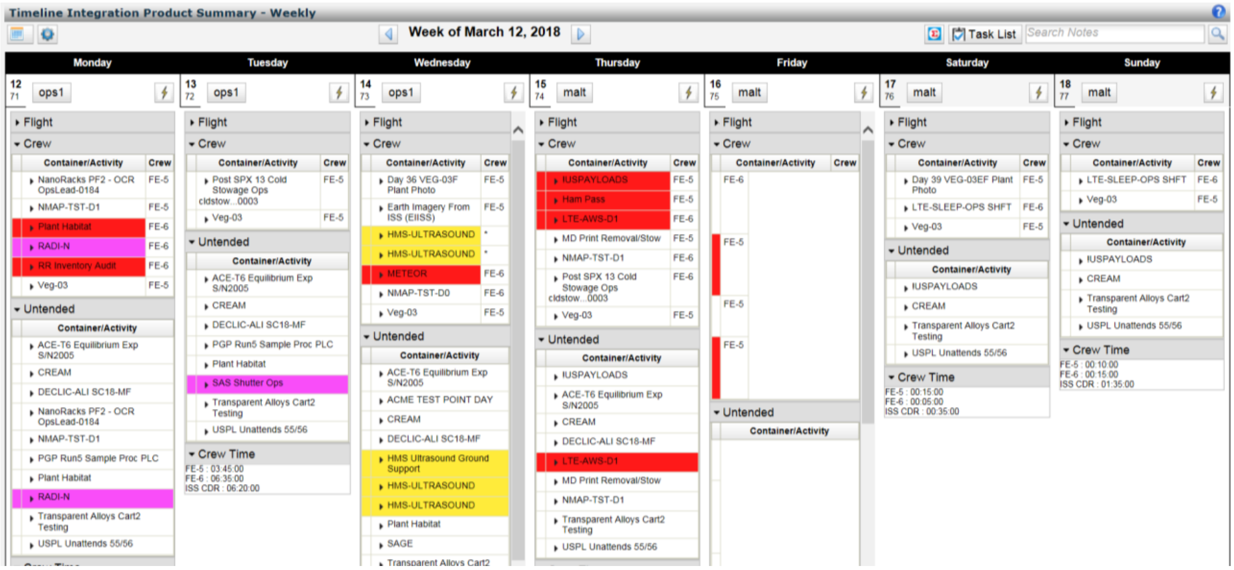

Fig. 1 The TIPS Weekly Report View.

The TIPS Daily Report view is a more detailed display of planning information. Figure 2 shows information about planning activities and any associated notes that have been added by TIPS users. Once again, planning activities are colored based on their execution status. If an activity has particular ISS crew members associated with it, then their positions are identified.

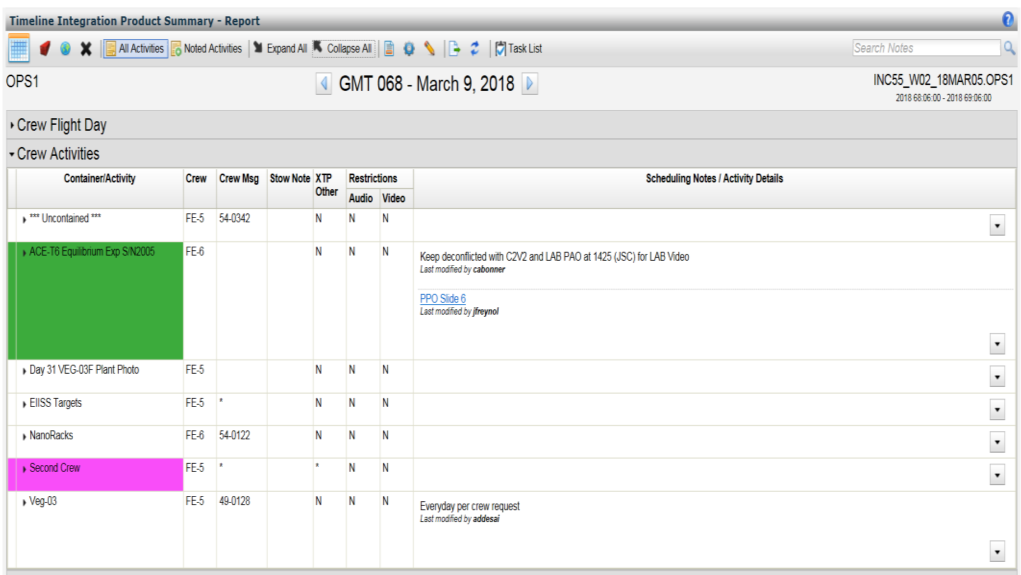

Fig. 2 The TIPS Daily Report View.

The TIPS Daily Report view provides privileged users with the ability to interact with creating custom groupings of information, to add in activities published from multiple ISS international originators (optional, if so desired), and to access other useful modes of viewing planning information. Notes have rich text format capability and display creation timestamps, and originator or modifier. Notes even have the option to be designated as “approved” by users with approval authority to speed communication between FCT members. Notes can be made on an entire Daily Report, on a container of activities, or on individual activities. All of these features aid the planning team with the ability to concisely view information from multiple sources in one location.

III. TIPS Architecture

TIPS is a rich web application that runs on most of today's modern browsers. It is hosted on a hardware environment that consists of redundant web servers, and, redundant business machines that support those web servers. The application's core capability is to generate a report summarizing the ISS planning activities. Included in this report are operational notes and change requests associated with each published plan activity. These reports are made available in various useful formats for consumption; as a daily view, as a summarized weekly view, or they may be downloaded in a printable format.

TIPS follows the classic three-tiered architecture pattern composed of data access, business logic, and presentation tiers. Figure 3 shows the data flow between the three tiers. Both the data access and business logic tiers are executed on the server-side business machines. They are written in the C# language and use the .NET Framework libraries. The presentation tier runs on the client-side web browser. It consists of a mixture of standard HTML and JavaScript using the Dojo Toolkit libraries.

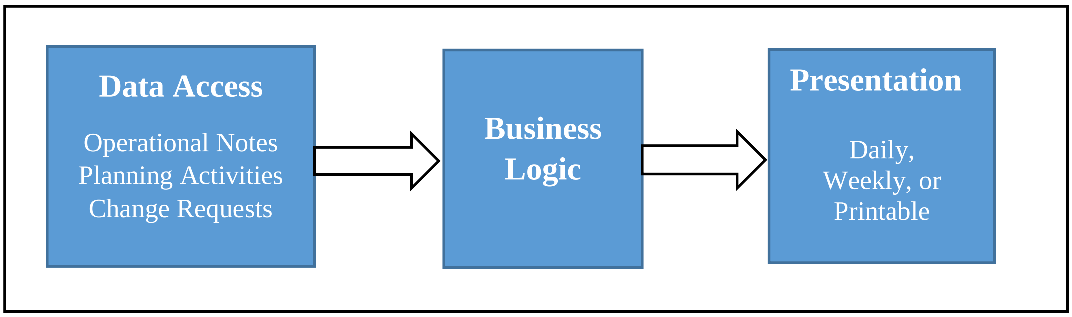

Fig. 3 The TIPS three-tiered Architecture.

A. Data Access

The data access tier is responsible for gathering all of the data displayed on a TIPS report. The data comes from several sources, both internal and external to the TIPS hosting environment. The ISS planning activities are stored on the OPTIMIS system hosted at JSC. They are accessed via web services over a secure connection. The change requests are also provided by JSC via a separate set of OPTIMIS web services. Change requests are created when a planned activity needs to be modified close to execution time. The operational notes are stored at the POIC in a Microsoft SQL database. Retrieving data, whether by web service or database, involves communication delays which can add up if done sequentially in one execution thread. TIPS retrieves data in parallel, from each source, on separate execution threads. This reduces the combined request delay time significantly.

B. Business Logic

The business logic tier is responsible for the integration of the three data sources into a hierarchical data structure. Planning activities are grouped into logical containers using either information provided by planning team users, or, with custom container definitions provided by TIPS users. Both activity containers and individual activities have a unique identifier. Operational notes can be associated with a container, or an activity, via its unique identifier. Similarly, an activity may have associated change requests. The business logic tier links all these items together. It also performs some miscellaneous calculations about crew times. The crew times are computed from activities assigned to ISS crew members. The end result is a data structure that is passed on to the presentation tier for display to the user.

C. Presentation Tier

The presentation tier runs on the user's web browser. It is responsible for both requesting and rendering a report. The request consists of executing a web service call with JavaScript code. The request's reply contains the report data structure built by the business logic tier. This structure is JSON-serialized to be more compact. Finally, the relevant user interface elements on the web page are populated with the report data.

TIPS supports three different ways to view reports: daily, weekly, and printable. Part of the report request includes an indicator of the view mode. The daily view displays all the report details for a selected day. This includes operational notes and change requests. The weekly view shows the reports for seven consecutive days beginning on a

Monday. For the weekly view mode, only the published planning activities are retrieved and displayed. The last view mode is the printable view mode, which generates a printer-friendly version of a daily report.

Figure 4 shows the timeline progression of a typical TIPS report, spanning the three tiers. The Data Access tier shows the three separate data sources fetched in parallel. The Business Logic tier combines the data into a unified structure. Finally, the Presentation tier both initiates a TIPS reports request and displays its results to the user at the browser level.

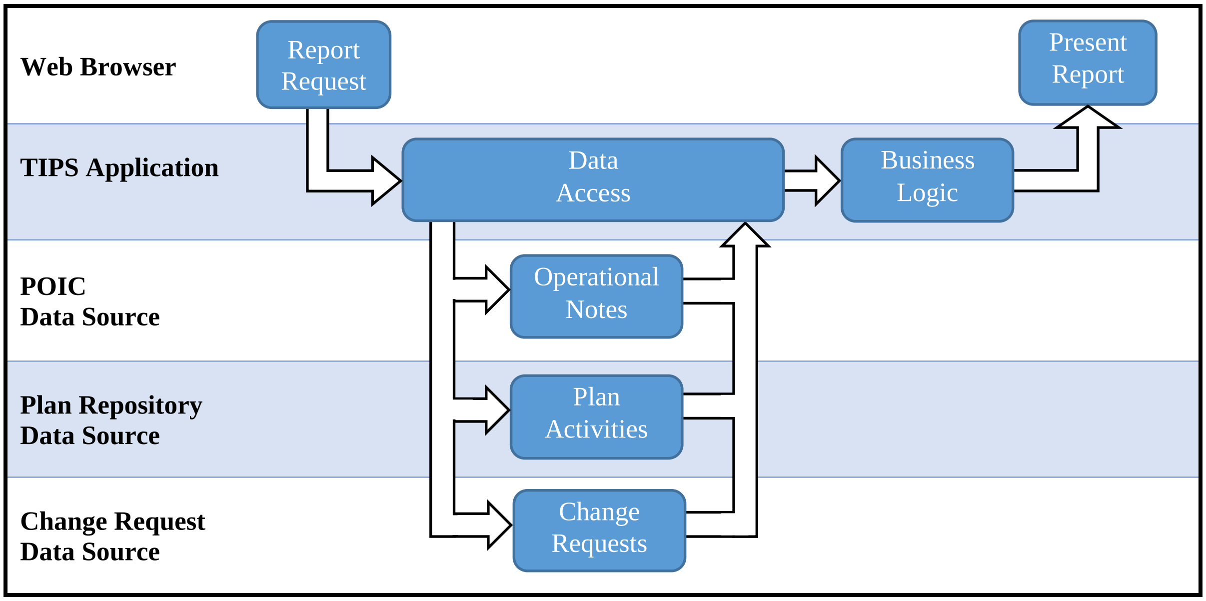

Fig. 4 Timeline of a TIPS report request.

IV. Hybridized Agile Software Development of TIPS

The requirements for the TIPS tool were loosely defined and anticipated to change across the software development lifecycle, thus an Agile Software Development (ASD) methodology was deemed most appropriate to support the need for ongoing user feedback in a short delivery timeframe. This contrasted with the existing, traditional "waterfall" mode of building software at the POIC. Additional information about this new paradigm is described in another SpaceOps paper by one of this paper's co-authors, Dr. Albers, entitled "Hybridized Agile Software Development of Flight Control Team Tools for International Space Station's Payload Operations Integration Center." TIPS became the POIC's first ASD project, and needed to strike the balance between responsiveness to deadlines inherent to any operational environment, while at the same time being flexible to changing customer needs. For the first time, flight operations support personnel desired to collaborate closely with the developers building the software, and that necessitated new avenues to successfully realize that goal.

A. How the POIC Hybridized Agile Implementation Worked for TIPS

TIPS became the proving ground for the new Scrum-based Agile methodology proposed to aid the development of the tools needed to support the HOT initiative. The object of this choice was to provide the opportunity for regular user interaction with the software as it was being developed, and at timely intervals that would still permit changing requirements and reasonable new customer requests. Figure 5 details how the TIPS Agile Development Lifecycle accomplished this. At the top of the diagram, the typical Agile phases of Envision, Speculate, Explore, Adapt, and Close are shown, and the specific activities and outcomes of each phase are listed below. Exploration during software development occurred in sprints. Each sprint concluded with a user evaluation where the PT discussed feedback and changes with the DT, or approved content on-the-spot. Adaptations were made quickly, efficiently, and were discussed collaboratively to ensure that providing the customer with a valuable, releasable software package at the end of the product development phase was always the ultimate goal. From the ConOps discussions, resource estimations and sprint planning, to the execution of the work and the adaptations made as a result of feedback obtained during the end-of-sprint user evaluations, ASD provided the TIPS team with needed flexibility. It created the ability to continuously explore the best outcomes for the tool, and permission to identify and make necessary changes along the way.

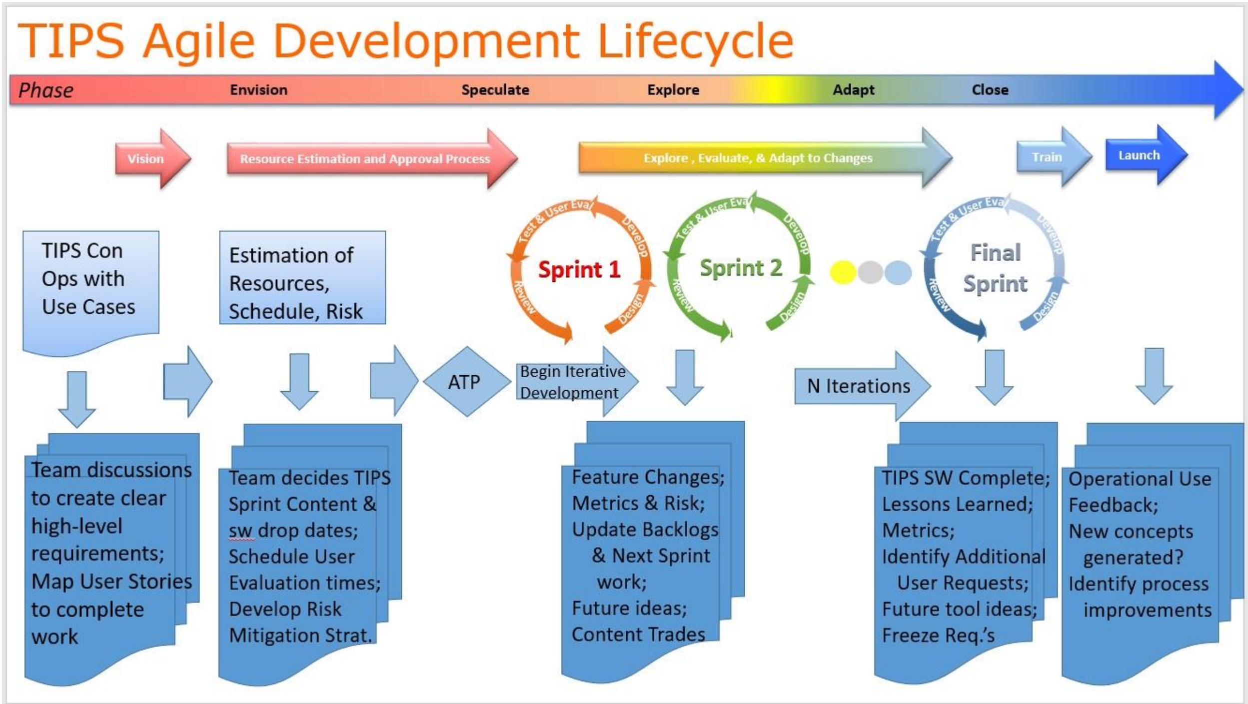

Fig 5. POIC Agile Software Development Lifecycle

B. Aiming at High Quality

The POIC hybridized Agile approach improved the TIPS software quality through a number of avenues. One such intrinsic quality improvement step involved advance testing. This allowed the Independent Verification and Validation (IV&V) test team to identify software bugs early instead of creating a late backlog of issues to be fixed once developers were already working on coding the next sprint's features. Additionally, intrinsic quality was verified by the users as they interacted with the tool's capabilities each sprint during the user evaluations. Extrinsic quality was largely captured during user evaluations on the TIPS tool as well. This provided additional verification that the POIC was properly implementing capabilities that met user expectations for the look and feel, and user experience of the tool.

Only value-adding activities were conducted during the user evaluations and sprints. Documentation was kept at a minimum and delivered just-in-time in order to minimize rework due to changes along the way. This left more time for the developers and testers to focus on providing high-quality software for evaluation. The software for each sprint was also given a quick check-out by the test team before going in front of the users for evaluation, to ensure no time was wasted by the PT.

C. Building a Self-Empowered TIPS Team the Agile Way

The TIPS team benefited from a lot of close collaboration across the development lifecycle. From initial ConOps discussions to define high-level minimum success criteria, to regular mid-sprint tag-ups, and team retrospectives after each sprint, frequent interactions between the DT and PT resulted in a common understanding of how the tool should function, look, and defined what the user experience should be.

ASD places value on the interactions between people, and moves beyond simply facilitating the work to be accomplished. The TIPS project benefited from the ASD focus on creating a thriving culture of collaborative innovation that fosters new ideas and breaks down barriers to effective communication. The TIPS project achieved this through holding recurring project and sprint-specific retrospectives, team-building activities, and socials. Thus, the TIPS team didn't only meet to work hard, they also met to celebrate accomplishments and milestones. Team-building activities were kept fun and light, and increased excitement about the next phases of work. Sometimes the DT would surprise the PT with a quick-look demo of advance work on features in the next sprint, or announce a new idea to provide an answer to an outstanding implementation question. Sometimes the PT would bring in special food to tag-ups, send along an email to the DT with high-praise from management, or pass along the excitement of the FCT about the upcoming release of the tool. These additional activities were not required; they were inspired. End-ofmeeting interactions grew more relaxed and less formal with time, and light-hearted conversations were possible, while still accomplishing the work. Long-standing relationships between two previously isolated teams developed, and discussions beyond the TIPS tool began to take shape, giving rise to new ideas for additional HOT tools. In short, the entire TIPS team enjoyed working on the project more than any other project at that time.

In effect, ASD brought the TIPS team together to pioneer a new mode of doing business between operations and software development at the POIC. It facilitated building trust and understanding between two very different functional environments. The retrospectives and less formal meetings were typically characterized by lessons learned conversations, which helped ascertain how well the team was operating and how it could improve. These events also helped identify synergies with other projects and potential new opportunities to leverage TIPS in the future.

V. The Benefits of TIPS and Future Endeavors

A. Dramatic Time Savings

One of the driving reasons behind the concept of TIPS was to provide time savings. Figure 6 shows a comparison between the labor hours expended before and after the usage of TIPS for a 25-day period. The conclusion is that the generation of manual reports was far more time-consuming than the automated generation of TIPS reports; in fact it is 60 times faster. The average time was reduced from forty *hours* to forty *minutes*, a dramatic time savings for the planning team.

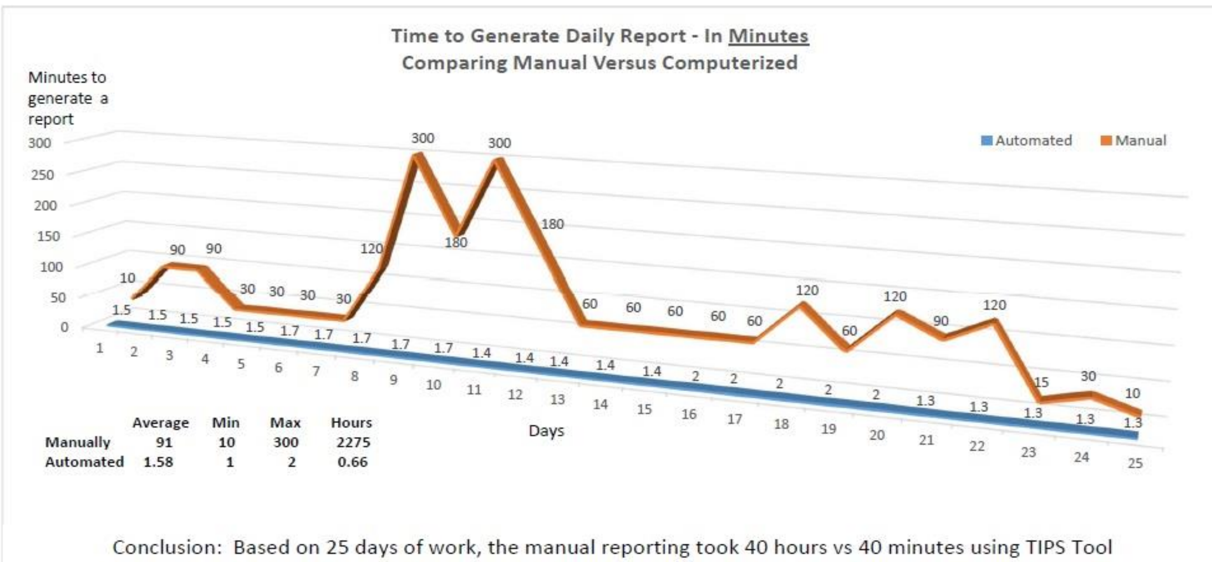

Fig 6. POIC Agile Software Development Lifecycle.

B. Product Team Perception of TIPS and Its Development

Since deployment, the implementation of TIPS in the planning process has allowed planners to focus on creating better products for both the crew and ground teams by freeing up the time that used to be spent updating documents manually. There is an increased focus on impacts to re-planning of activities now that making report updates is automated. Also, because TIPS reports display live data directly from OPTIMIS web services, comments from the FCT on timeline plan summaries have dramatically decreased, allowing more time for the team to focus on execution of the plan, rather than the plan itself. TIPS is a vital tool for MSFC to support increased payload science with the addition of a fourth crew member onboard the ISS.

The Agile Software Development of TIPS allowed MSFC planners to spend additional time with the development team than normally would have been possible, and to provide the DT with a shared understanding of the concept. The process also allowed users from all FCT positions to see early demonstrations, to evaluate the product, and to provide feedback on the tool during the development phase, which were vital to all positions having influence on the final

TIPS product. Feedback from this process allowed PT leads to reprioritize tool features and maximize utilization of the development time allotted. Throughout the development phase there were several discussions which were key to receiving the most useful end product, and without the ASD paradigm of continuous project integration, this would not have been achieved.

The ASD process provided users with hands-on time with TIPS prior to its completion. This helped the PT refine requirements for the look and feel of the tool's displays and functionalities. One example of this feedback process to the DT involved the evaluation of making display changes to the report after creation. During collaborative discussions with the DT, the PT realized they could connect the change request web service to deliver published changes directly into the report, and have the latest changes displayed as a note on each activity. Without the ASD-facilitated collaboration, this would not have been identified and brought to fruition. Also, during evaluations, users testing out the software noticed some sluggishness in retrieval of those change requests. They were able to point out that feedback, and the DT was able to fix it prior to deployment. TIPS has also fostered new ideas for note storing among the FCT and will continue to create further efficiencies with planned future development.

TIPS was initially proposed as a single ISS Increment development effort. This would have meant developing the entire scope of the project in a single phase, and delivering it as a final and complete project at one discrete time. One of the benefits of using an ASD approach was that through continuous collaboration with the customer, ideas from both the DT and PT were permitted to evolve such that concepts for several additional, valuable capabilities were identified. That gave rise a request for a second development cycle to implement more complex capabilities, as well as some of the formerly "Nice to Have" features that, when implemented, further increased the tool's usefulness.

C. The Future of TIPS

Once the first phase of TIPS was deployed into an operational state, NASA Johnson Space Center (JSC) took notice of the dramatic improvement in time-to-completion of reports, the reduction of human error, the ability to quickly re-plan, and the overall benefits of having an automated process that could easily provide the user with the latest-and-greatest information available. It wasn't long before JSC planners requested their own instance of TIPS. This request was not only a result of the quality and design of the tool, but also due to the general enthusiasm for TIPS by the planning teams. As of the time of this paper, the TIPS for JSC project is scheduled for operational deployment in the fall of 2018.

TIPS was given a very positive reception at deployment since the reports are also useful to many positions on the FCT. By including other positions in the user evaluations as well as periodic human factors assessments, the TIPS tool has become more useful and valuable than its original concept. With configurable user access permissions, the tool is designed to be a single-access nexus of timeline planning information across the flight control and preparation teams. This achievement is a direct result of the synergy facilitated by advanced collaboration under the POIC hybridized ASD implementation. In the future, the POIC plans to leverage TIPS as a central piece of an even more interconnected FCT toolset, and also plans to request the continued use of ASD to ensure the future toolset will be the direct result of a close collaborative effort focused on customer-defined value.

VI. Conclusion

TIPS represents the next generation in timeline planning, moving from a manual process to an automated one. The forward-thinking TIPS software architecture design combined with the ASD collaborative process to result in the fast completion of a tool that delivers high customer value. Not only does TIPS fulfill functional requirements, but its implementation directly reflects the evolving needs of the customer. The planning team finds the reports intuitive, easy to understand, free of human error, and reliable; TIPS permits an efficient and continuously up-to-date view into the planning timeline information, and provides console operators with a clear summary for execution. This became more complex with the addition of a fourth crew member and increased payload science activities, so there is additional value built into TIPS through its design, and ability to assist with planning judicious resource allocation. After just the first phase, efficiency metrics revealed a reduction in average time to complete the summarization of a timeline plan, from 40 hours with a manual report, to 40 minutes with a TIPS Report (60 times faster).

Through the unique collaborative Agile Software Development (ASD) process, the team developed a synergy that allowed for a free flow of ideas. This resulted in the desire for previously unidentified additional capabilities, and when some resources to support it came available, the customer requested a second phase of TIPS to forge ahead with the creation of these newly identified capabilities to make the tool even more useful. With just the first phase of TIPS in operations, NASA JSC began to take notice, and shortly after the first release, requested an instance of TIPS for JSC planning and operations. The flexible TIPS architecture is able to accommodate a JSC instance with relatively few software changes, thus the JSC version can be provided quickly and is planned for release in fall of 2018.

TIPS is more than a tool to optimize an existing way of planning and viewing timeline information, it is a next-generation automation capability that permits a highly efficient, real-time view into cross-center planning of payload science activities. The development of TIPS into such an immediately beneficial and valuable tool, within the short timeframe between concept and deployment to support the High Operations Tempo (HOT) of increasing payload science activities, was a direct result of employing hybridized Agile Software Development (ASD) methods. This pioneered the first successful collaboration between console operators and software developers during the software development phase in the history of the POIC. For the first time, customers had direct input into the tool they would be seeing in operations in less than nine months, while there was still time to make changes to implementation details and have their feedback on the look and feel of the tool incorporated quickly before release. With the amount of increasing payload science activities to plan in such a short amount of time, this was extremely beneficial in that there was no longer a need for the customer to wait for the tool to be in operations before their feedback could be gathered and passed along to possibly be addressed later on if time and resources so permitted. In short, the practical application of a time-boxed, hybridized ASD methodology permitted the delivery of customer-driven value despite changing requirements, while experiencing the benefits of synergistic collaboration and giving rise to new and innovative future ideas.

TIPS is a valuable timeline planning tool but it also represents a major milestone for the POIC space flight operations community. Bringing operations and software development teams together to achieve high-value objectives quickly has inherent challenges, when only having an initially abstract concept. The TIPS team achieved an unprecedented level of success together and paved the way for future ASD projects at the POIC. As different instances of TIPS will be deployed to operations for multiple centers by the end of 2018, the TIPS team anticipates the tool gaining a wider audience which could result in additional implementation requests. It is hoped that partners in the international community of SpaceOps will be encouraged by the TIPS success story. The desire is to trade, with other space science operations centers, effective strategies for cross-center planning architectures, employing customer-developer focused collaborations, to improve the ability to plan an increasing number of science activities in space.

Acknowledgements

Mr. Pedoto would like to thank COLSA Corporation for its generous support of the author's attendance at SpaceOps 2018. All the authors wish to thank the dedicated personnel at the POI for their hard work on the TIPS tool, their continual championing of its usefulness, and for their dedication to providing high quality, continuous operational support for ISS payload science activities.

References

[1] TIPS, Timeline Integration Product Summary, Software Package, Ver. 2.0, NASA MSFC POI, Huntsville, AL, 2017.

[2] E. Smith, Ernest & Korsmeyer, David. (2011). Intelligent Systems Technologies for Human Space Exploration Mission Operations. 169 - 176. 10.1109/SMC-IT.2011.12. URL: https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20120016015.pdf [retrieved 20 April 2018].