Managing a Product Development Team: Part I

Larry Barrett • Ken Lehtonen

Orbiting 380 miles above the earth, NASA's Hubble Space Telescope (HST) has returned a wealth of scientific data about our universe and galaxies beyond, highlighted by spectacular images of the birth and death of stars, colliding galaxies, and other extra-terrestrial events.

CCS Integration Environment ("The Triangle"). NASA CCS photograph

Despite its tremendous success for almost two decades, the HST ground support system experienced down-to-earth problems—namely budgetary ones—prior to the turn of the 21st century. To keep HST operating efficiently to 2012 and beyond, the Vision 2000 project was conceived with the primary goal of substantially reducing the costs of operating and maintaining the spacecraft ground systems. Taking advantage of this atypical management opportunity, a set of product development teams (PDTs) was established and given the charter to re-engineer the ground system and by so doing, reduce the remaining life-of-mission operating and maintenance costs while providing improved reliability and increased capabilities.

One of those PDTs, the Control Center System (CCS) PDT, was charged with developing and deploying the system that is still responsible for the overall health and safety of the HST vehicle by sending commands to the HST vehicle for scientific data acquisition, acquiring real-time engineering telemetry data, and providing accurate and timely problem diagnosis. This article discusses the overall management of the CCS PDT as it struggled to embrace a brave new world of leading-edge technologies and to successfully advance a new management culture, and it focuses on several of the more successful techniques and strategies that ultimately ensured the success of this team.

Establishing Technical Goals

The major technical goals established for the PDT were:

- To challenge the old ways of doing business and apply new technologies where appropriate
- To build a system that, within a distributed but scalable architecture, combined reused legacy applications (e.g., HST-specific algorithms), commercial off-the-shelf (COTS) products, government off-the-shelf (GOTS) products; and leveraged evolving technologies
- To design an evolutionary and maintainable system
- To execute a development strategy for incremental releases to ensure that the HST operations staff and systems engineers could gain early operations experience, thus giving the development staff time to clarify requirements early in the process
- To become an innovative leader in developing control center systems for NASA-Goddard.

Those goals became the major guideposts for evaluating how the PDT was grown, how it was managed, and

__Barrett__ is the chief systems engineer for the HST control center system. He is primarily responsible for system architecture oversight and technical risk management. __Lehtonen__ has over 35 years of experience in software engineering. His experience includes designing, implementing, testing, and managing a wide variety of mission software applications for NASA.

how the technical decision-making process, which is always required during the course of a system development project, was optimized.

Looking for Challenges

The HST project team, located at Goddard Space Flight Center in Greenbelt, Md., was responsible for funding the Vision 2000 initiative. The team made it clear from the outset that they were looking for "new and better ways of doing business," even if this meant re-engineering the existing ground system and totally replacing it with a new system based on advanced and potentially unproven technologies.

A second challenge was the aggressive schedule dictated for this project. It was essential that the new Hubble control center system be fully operational at least one year before the third HST servicing mission scheduled for December 1999. In addition, senior HST management wanted the first major release of the CCS to "shadow" the second HST servicing mission scheduled for February 1997! So beginning in December 1995 with an intense proof of concept demonstration and culminating with the CCS architecture specification in April 1996, it would be necessary to deliver a fully operational system within three years and the shadow CCS within 10 months. (The existing ground system had taken over five years to develop.) This objective drove the delivery of six major software releases within the first year (Figure 1).

The goal was to eliminate the sources of inefficiency on the project by building a culture that fostered an atmosphere of cooperation and that was success-oriented.

An additional challenge to the management of the PDT was the decision to use—initially—the legacy software maintenance team who, although highly trained and well versed in structured development methodologies, were not as skilled in more current system design and development technologies. In fact, the PDT was front loaded (in terms of numbers) with a technical staff that normally wouldn't be required until after a traditional preliminary design review. An attrition rate that approached 30 percent showed the new management team that traditional approaches weren't going to work. One management guideline that was actively employed with good results was to utilize those management techniques that had been successfully applied to small teams or were currently being used successfully in similar re-engineering projects. The goal was to eliminate the sources of inefficiency on the project by building a culture that fostered an atmosphere of cooperation and that was success-oriented.

Flattening the Organization

One of the management team's first actions, which gradually paid big dividends, was to flatten the project organization. We established a minimal project management support staff consisting of two release managers, two quality assurance personnel, a resource scheduler, and a single administrative assistant. The remainder of the organizational structure consisted of a set of core technical teams, each with a technical lead "supervisor." There was a significant amount of initial resistance to this "radical" approach because the traditional hierarchical management structure ("command and control") from the legacy organization was firmly entrenched. Fortunately, the key stakeholder for the project was very supportive of this approach, and the staff quickly accepted a structure that imposed less bureaucracy.

Each of the empowered team leads was held responsible for implementing a specific subsystem within the Hubble control center. For example, there was a core team to develop the spacecraft command functions, another team to develop the graphical user interface (GUI), a middleware team, a data management team, and so on. The leads were also tasked with ensuring that their staffs were the right size, embodied the appropriate skill mix, and were properly trained. The technical decision-making process was pushed down as far as possible in order to streamline the overall development effort—remember we had very aggressive schedules to meet. To complete

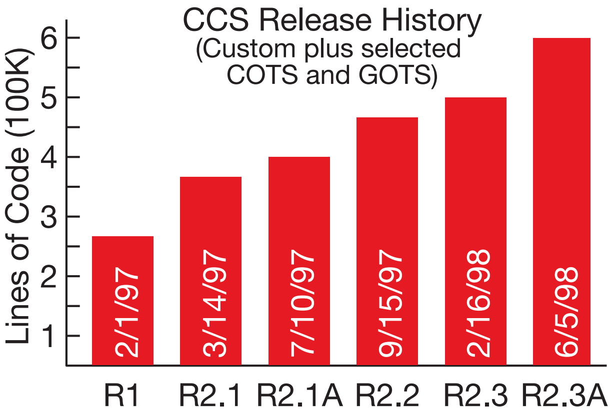

FIGURE 1: **Summary of CCS Productivity**

the picture, the release managers, who were ultimately responsible for delivering the next scheduled version of the Hubble control center system, were charged with identifying the resources they needed to meet their delivery schedule. Thus, they were required to negotiate with each of the core team leads to borrow personnel to establish a release team with the right skill mix. Only if a conflict arose did the project lead intervene to clarify the priorities and/or to reallocate the resources.

As a result of this new organizational structure, team members had both an organizational "core" identity that closely matched their own technical skills and an affiliation with the delivered system release. A better designation for this new organization was "dynamic matrix," as the free flow of information between and among teams and team members was encouraged, as was staff movement between teams when conditions warranted.

Another facet of the organization's character was management's tolerance for "failure" without retribution. After any significant problem was resolved, a post-mortem was conducted, and if necessary, process improvement was initiated and/or the team's skill mix was adjusted. This approach led to a project environment that fostered the growth of personal strengths instead of punishing weaknesses. As a result, the staff became more willing to assume responsibility for making the decisions necessary to meet the aggressive schedule. Over time, a set of informal checks and balances evolved between the teams enabling continual progress, rapid decision making, and a reduction in the magnitude of corrections required.

Communication, Communication, Communication

The CCS management team realized that a project of this size (it started out with 75 people and reached its maximum at about 150 people) and complexity required constant and effective communications, oral as well as written. We took advantage of the fact that the HST project decided to colocate the majority of the CCS team to a nearby off-site building. Under one roof we housed the systems engineers, software developers, system testers, hardware engineers, operations personnel (small subset of HST flight operations team), quality assurance, procurement, and management personnel. This arrangement turned out to be one of the major reasons for the success of this project and will be examined in greater detail in Part II.

As with any team effort, a combination of both formal and informal communications was required. We used formal communications chiefly to inform HST stakeholders and senior management of the status of the project. Core teams and the release managers provided status estimates for their areas on a weekly basis. The CCS management team then met on a scheduled basis with the key project stakeholders and presented a consolidated status in terms of schedules, percent complete estimates, and other traditional project management reporting vehicles. Periodic formal presentations to an independent audit team were also required to ensure that all of the PDTs were progressing as planned and in unison with each other.

On an informal basis, the CCS management team implemented a series of actions that proved to be highly beneficial to overall project success. First of all, the building's layout was leveraged to group each of the technical teams as physically close together as possible. This step enabled significant intra- and inter-team communications at the technical level as well as for the project team as a whole. As a result, it was very easy for anyone on the project to obtain a real-time status of the development process, and staff mingling was encouraged and supported by the management team. We also conducted daily 10-minute "stand-up" technical meetings led by the release manager to foster timely communications across groups. To further enhance communications during software release integration, we dedicated a portion of the building as an integration facility, dubbed "the Triangle." Each Hubble control center core team had its own dedicated floor space and workstations. As the CCS data flows traversed through the system during a particular test, the teams were able to communicate directly and immediately with each other and to identify firsthand any interface problems that arose. This was a significant contributing factor to our on-time software deliveries.

As the project progressed, it was necessary to increase the size of the staff, especially in those technology areas where the legacy personnel were less experienced. About that time, in order to enhance our team-building activities, we began to hold biweekly summer barbecues. In turn, the core teams assumed responsibility for the theme and management of the cookout. The rationale for these social occasions was the conviction that people who got to know each other in an informal, non-stressful setting would work together much more effectively during the stressful software system integration period. This reasoning proved to be correct: the traditional finger-pointing associated with system interface testing was virtually non-existent within the PDT.

It's also important to note that as project lead, author Lehtonen conducted frequent informal MBWA ("management by walking around") sessions. These sessions enabled him to meet firsthand all of the members of the various teams as well as to communicate the ideals for an open, inclusive project and to encourage the sharing of technical knowledge among team members. It also strengthened the goal of having an active and trusted management presence on the project.

An Electronic World

We made a decision to reduce but not entirely eliminate the need for hardcopy documentation (addressing the often-heard comment that as soon as a document is published it is out of date). To that end, a couple of internal Web sites were established for the electronic distribution of key documents. We also relied heavily on e-mail. At the core of the design process, a CASE (computer-aided software engineering) tool was established to store our Object Management Technology (OMT) design information electronically and to generate hardcopy design documents for walkthroughs. This electronic repository was not only very effective in streamlining the documentation of the development process, but it also remained a key engineering component during the transition from the purely development project to the current sustaining engineering environment.

In effect, what we accomplished was to embed an effective information management environment within the project. By carefully selecting and tailoring the right tools and processes, we were able to enhance technical communication significantly and meet the information needs of the project. Because of this tailoring, we had at our fingertips the necessary information to facilitate decision-making processes, making for quick analysis of alternatives and timely selection, which kept the team moving forward at all times.

*In Part Two of this article (*Defense AT&L *May-June), the authors explore some of the challenges of building a cohesive, synergistic team, and conclude the article with a list of "implementation strategies" that were used successfully on this project and might be helpful for readers' projects.*

Editor's note: The authors welcome comments and questions. Barrett can be reached at lbarrett@hst.nasa.gov. Lehtonen can be reached at kenneth.e.lehtonen@nasa.gov.

Office of Force Transformation Unveils Primer on

Network-Centric Warfare (Jan. 7, 2004)

To download a copy, go to the Office of Force Transformation Web site at:

http://www.oft.osd.mil/index.cfm