Bulgarian Academy of Sciences. Space Research and Technology Institute.
Aerospace Research in Bulgaria. 37, 2025, Sofia

NASA – PROJECT MANAGEMENT CONCEPTS, METHODOLOGIES, AND PROCESS MODELS (1980–2020)

Stanimir Todorov

Valencia College, USA
e-mail: stodorov@mail.valenciacollege.edu

Keywords: *NASA, Project Management, Agile, Predictive Methodologies, Waterfall Model,*

Abstract

The aim of this paper is to summarize the Project Management techniques and methodologies used by NASA during the specified period. The transition from a Predictive perspective to an Adaptive perspective, and software project types are visualized in this picture. This Research Case Study aims to demonstrate the advantages of iterative and adaptive models.

Introduction

NASA has been at the forefront of space exploration and innovation for over 60 years. The agency has struggled to oversee large-scale, high-risk projects, from space missions to spacecraft development. As missions have grown more complicated, NASA has, over the years, modified existing approaches to project management. NASA, in general, has had quite a sea change between somewhat traditional project management frameworks, such as Waterfall, to more modern, adaptive Agile frameworks. This paper describes the progressive change in NASA's project management methodologies over the past four decades. This would allow successes and pain-points in implementing Agile and hybrid approaches to be clearer. The aim of this case study research paper is to highlight how it is crucial to adopt adaptive and iterative approaches to the software engineering kernel within the scope of complex space missions. In relation to the requirements of this document, some clarifications should be stated:

1. **Agile:** Agile is a project management methodology that promotes flexibility, incremental development, small deliverable iterations, and the collaboration of cross-functional teams. Agile has many great principles like adaptive planning, evolutionary development, early delivery, and continual improvement.

2. **Waterfall Project Management Methodology:** The Waterfall model is a sequential project management methodology in which every stage needs

to be finished before the next stage begins. The model describes a series of stages, each of which has a clear output that feeds into the next stage, including requirements definition, system and software design, implementation, testing, deployment, and maintenance. (PMI, PMBOK Guide 6th edition).

3. **Hybrid Approach Example:** NASA's Artemis Program is a very good example of the use of a hybrid approach, embracing Agile for software development while using Waterfall for hardware design.

4. **Scrum:** A framework that gives options to the teams to work together and make iterative progress using well-defined roles, events, and artifacts. It breaks work down into time-boxed pieces called sprints. (PMI, PMBOK Guide 6th edn, 2017)

5. **Kanban:** A Visual Management Method that helps a team visualize work, limit work in progress, and maximize flow. It focuses on continuously delivering and improving. (PMI, PMBOK Guide 6th ed.)

1. Period I – Predictive Approaches (1980s – early 2000s)

In the 1980s, NASA firmly followed the principles of traditional project management in its approach to project management. This model —also known as Waterfall — follows a linear, sequential process where completing one project phase is a predecessor for the next. This strategy served NASA well on the straight-focus, large-scale missions it ran during this period, many of which had firm specifications and strict safety stipulations.

Key Features of the Traditional Framework:

- **Formulate Phases:** Every NASA program starts with the initial concept, then moves to planning, from arrangements to working, and finally to conveying.
- **Firm Requirements:** The requirements of the project are being set early in the project life cycle and are clear and stable.
- **Strong and detailed Documentation:** Each of the phases was strictly documented.
- **Risk management:** Traditional focus on potential risk management and attempts to foresee future trouble [NASA, 1997].

Use Cases:

This model was paramount to the success of many missions, like the Space Shuttle Program and the beginning of Mars Rover missions and was embraced by NASA. And with the development of the Space Shuttle, which was based on firm and well-defined requirements, a fixed budget, and a strict timetable, the predictive model was applicable (NASA 1981). In the same way, the Hubble Space Telescope followed a predictive model that included the specific schedules of key milestones and an integrated risk management process that allowed early warning of problems [Smith & West, 1995].

Predictive Models Are Hard:

In any case, the predictive model can be useful in managing structured and well-defined projects, but as space missions got more complex, the predictive model had serious disadvantages. As projects grew in scale and complexity, it was obvious that aspects of the work that were not predicted and changes in requirements would soon reflect in the project timeline. A native example of the problems of predictive management was the mission to the Hubble Space Telescope — Hubble's mirror was found to be defective long after the telescope was deployed. Focusing on such strategic issues required a more Agile and flexible approach, which predictive models do not really understand and support [Simmonds, 2001].

2. The Agile Foundation (Late 2000s to 2010s)

By the mid-2000s, the context for NASA's missions was changing very fast. The technology for space exploration matured, and the scale of projects grew exponentially. Agile was established to deal with the fast changes of software development and provided a new foundation for executing projects. Realizing the changing environment, NASA began to look for adaptive approaches and possibilities.

The Rise of Flexibility:

Despite the prediction of the future and funding it, many started to embrace the agile methodologies, to iteratively design, develop, and receive immediate feedback on software and other engineering-related solutions. Agile is for big, complex projects and breaking them down into smaller, more manageable chunks of work – known as "sprints" in SCRUM – and adaptive reviews [Beck et al., 2001].

NASA's engagement with Agile was, at first, a fantastic decision. They needed to try to balance the flexibility needed by the software development and the firm and strict technical specifications and safety standards that are paramount for space missions. The condition here is that, instead of Kubernetes Agile across the board of the whole mission, NASA adopted it one mission at a time, and explicitly where an iterative and Agile approach would make a significant difference.

Agile in Practice:

To be noted as one of the early projects to utilize Agile as a project management framework in NASA was the Curiosity Rover on the Mars Science

Laboratory (MSL). The software development process of NASA was chosen to be Agile. This flexibility gave the software developers time to overcome unforeseen problems, iterate on each deliverable, and adapt to change more quickly (Simmonds, 2007).

- **Development cycles using Scrum and Kanban: "**Development cycles were managed at NASA using Scrum and Kanban frameworks. These frameworks pushed teams for cross-functional collaboration and to stay on priorities and timelines for their projects" [Rigby, Sutherland, & Takeuchi, 2016].

- **Sprints and Iterations:** For example, in the Curiosity project, it was possible to break development down into small (a couple of weeks) sprints (SCRUM) that enabled continuous tests and integrations. This allowed NASA to iterate quickly and have quick sessions where they just sit down and remove lots of bugs in the software, which were critical for the development of the software that would need to evacuate the rover.

A lot of NASA's Agile transformation forged more responsive, adaptive software-based systems geared towards dealing with the uncertainties of space operations. For Curiosity Rover, that refers to the development team that was able to fix the rover's software to respond to the unknown environment of Mars.

3. How NASA is Leveraging the Power of Agile

NASA has been using Agile for the software work, and their experience uncovered the need for a transformation in managing missions from the ground up, while using an Agile approach to surface at that level took time, and there was no universal solution to cover all the mission components. Even though the adaptive approach helped in lots of areas of software development, NASA still decided that the traditional project management approaches were indeed guaranteed for other, less flexible elements of space missions, especially hardware related.

Why Agile Became a Software Development Standard:

"The James Webb Space Telescope was the subject of another Agile case study at NASA, as this new instrument represents yet another sign of NASA's growing use of Agile frameworks. JWST is one of the grandest of all space observatories! James Webb Space Telescope development crashed through blowback, like cost overruns and delays. In theory, this theory then underpinned an evolution in its own process to address and resolve all these compounding conflicts: to minimize, NASA has adopted Agile, to optimize certain facets of the project, particularly, its software development process and its risk management" [NASA, 2015].

- **Agile Risk Management:** "The iterative structure of this method led to better risk identification and response for project managers, on the go. The

JWST development team applied daily sprint reviews and retrospectives to identify potential problems early and solve them quickly" [Harbaugh, 2017].

- **Software Integration:** An Agile software development approach enabled the team to consistently iterate through new features in the software integration. This was vital because the James Webb Space Telescope was such a comprehensive and complicated project that essential software was meticulously tested in a single-write manner to confirm everything is operating as expected.

Integrating Agile in Hardware and Integration:

With all the other aspects of the development machine, it worked to apply Agile to software and risks, but the development of hardware and connecting the large systems required a more consistent approach. As hardware design requires detailed specifications and adherence to government and safety standards, NASA has also found that traditional methodologies like Waterfall have benefited their projects. Hence, Agile was largely limited to subsystems like software development, testing, and simulations, while the traditional model was embraced for the hardware development.

4. Hybrid Model: Agile + Waterfall (Late 2010s till Now)

NASA's response was an amalgam of adaptive and predictive frameworks for project management by the end of the 2010s. The hybrid approach allows NASA to give them, on the one hand, the flexibility of Agile while at the same time providing them with the discipline and framework of predictive models; the examples are the Artemis Program and Orion Spacecraft.

Implementing Hybrid Framework:

- **Software Agile:** "As part of the Artemis Program, NASA transitions from traditional development habits and practices to Agile for software, focusing on simulation, testing, and specific engineering subsystems. This allowed quick iteration because teams were able to build on lessons learned from earlier phases of the mission in the later ones" [NASA, 2020].
- **Predictive for the Hardware:** Predictive approaches are highlighting hardware design and systems integration. Building hardware, especially human-rated spacecraft, is, on the one hand, very expensive and, on the other, firmly regulated, and demands a much stricter, more sequential approach to check that it's going to be safe and do the job.

Benefits Of the Hybrid Model:

- Agility and Rigor: By above all being Agile wherever it is practical and by remaining in predictive techniques where justified, "NASA can easily adapt to changes in growing old and mission needs; however, continue to substantiate the integrity of mission-critical components" [Cohn & Ford, 2010].

- A hybrid strategy maximizes efficiency by enhancing cross-team collaboration and quality assurance for both software and hardware. Software development proves faster and cheaper while spacecraft assembly and integration proceed with rigorous oversight.
- This method proves more adaptable when confronting unexpected issues and obstacles, ensuring focus is maintained on core goals yet ample flexibility exists for necessary innovations.

Lessons Learned and the Path Forward

- Adaptability reigns supreme, especially for software development, risk management, and external partnerships, as shown by data through October 2023.
- A balanced mix proves critical since while Agile benefits certain domains, hierarchical oversight remains vital for compliance in mission-critical construction, as NASA recognized in 2015.

The end goal:

As experience grows, NASA's refinement will yield methods enabling objective delivery of heavy-lift ventures within each directorate's minimized unwarranted interference. With new frontiers on the horizon like Moon and Mars missions, flexibility, iteration, and cross-team coordination will become ever more essential. Also, partnerships with private industry and cutting-edge commercial space require more pliancy, integrating Agile into strategic planning.

Conclusion

NASA's avoidance of rigid predefined plans, part of a wider industry shift, brought the creativity and adaptability of Agile to a fast-changing, intricate environment where traditional methods still play key reliability and success roles. Continued integration of Agile experience can enhance NASA's life cycle management. This hybrid approach exemplifies the promising future of space exploration project management by synthesizing traditional and Agile techniques.

References

1. Beck, K., et al. (2001). Manifesto for Agile Software Development. Agile Manifesto
2. Cohn, M., & Ford, D. (2010). Agile Estimating and Planning. Prentice Hall.
3. Harbaugh, R. (2017). The Challenges and Innovations in the Development of the James Webb Space Telescope. NASA Technical Reports, 2017.
4. NASA. (1981). Space Shuttle Program: Overview and Milestones. NASA Headquarters.

6. NASA. (1997). Risk Management and Project Management: The Experience of NASA. NASA History Division.
7. NASA. (2015). James Webb Space Telescope: A Case Study in Project Management and Risk Mitigation. NASA Science Mission Directorate.
8. NASA. (2020). The Artemis Program: Mission Overview and Future Goals. NASA.
9. Rigby, D. K., Sutherland, J., & Takeuchi, H. (2016). Embracing Agile: How the Most Innovative Companies Capitalize on Agile Software Development. Harvard Business Review.
10. Simmonds, S. (2001). The Hubble Space Telescope: Lessons Learned from its Early Development. Journal of Spacecraft and Rockets, 38(3), 145-151.
11. Simmonds, S. (2007). Adapting Agile for Space Exploration: The Curiosity Rover Experience. Proceedings of the International Astronautical Congress, 2007.
12. Smith, M., & West, D. (1995). The Hubble Space Telescope: The Ups and Downs of Its Development. Journal of Space Mission Management, 12(2), 35-42.
13. Project Management Institute. *PMBOK Guide | Project Management Institute*. 7th ed., Aug. 2021, https://www.pmi.org/standards/pmbok.
14. Project Management Institute. PMBOK Guide Project Management Institute 6th ed., Sep. 2017, https://www.pmi.org

НАСА – МЕТОДОЛОГИИ И ПОДХОДИ ЗА УПРАВЛЕНИЕ НА ПРОЕКТИ (1980–2020)

С. Тодоров

Резюме

В статията се описват начините, методологиите и техниките, използвани при управлението на проектите в НАСА за периода 1980–2020 г. Това е преглед на методологиите за посочения период, както промяната и прехода от традиционен към адаптивен или т. нар. гъвкав подход (Agile), като специално внимание е насочено към софтуерните проекти. Използването на хибриден модел при управлението на проектите е свързано със спецификите им. При разработването на хардуерни компоненти е следван традиционен модел с оглед регулациите, за сигурност и държавни изисквания, като при софтуерните модели е следван гъвкав модел. Целта на статията е да постави на фокус предимствата на гъвкавите подходи при управлението на проектите, съпътствано със съответните примери. Това изследване е обобщително, но не е изчерпателно, като е базирано на общодостъпна литература и официално публикувани данни.