Requirements Management and Functional Requirements for NGAO

KECK ADAPTIVE OPTICS NOTE 573

Christopher Neyman, Viswa Velur, and Erik Johansson
March 25, 2008

ABSTRACT

This note outlines the requirements management process for the NGAO project. It also discusses the interrelationship between the various NGAO requirements documents and the overall design and devolvement process. The text of the functional requirements is contained as an appendix.

1. Overview of NGAO Development

A standard systems engineering tool for visualizing the development process of a project is the V-diagram. The NGAO V-diagram[1] is shown in Figure 1. The diagram shows the progress of the project from initial concept to completion, with the time axis increasing along the bottom of the diagram. Going down the V, the design of the project becomes progressively more and more defined. The design process proceeds down to a level where the subsystems are ready to be fabricated or purchased at the bottom of the V. The design processes starts with high-level requirements and continues through the system architecture definition. This is followed by the subsystem architecture definition. Next, the design process continues with increasing detail through system design, preliminary design, and ends with detailed design.

As subsystems are completed, they are tested to verify performance. Proceeding up the other side of the V, the subsystems are integrated into a full system progressing towards final testing as a complete scientific facility. Results of the performance verification of each succeeding step are documented to verify compliance with the requirements and as a baseline for determining system health throughout is operational lifetime.

In addition to the stages of project design and construction, Figure 1 also contains the associated documentation that captures the NGAO design. These requirements documents are the focus of the next section. Also shown in Figure 1 are the testing plans associated with each level in the NGAO construction. These documents contain the tests used to verify that the required performance is achieved.

2. Hierarchy of Requirements Documents for NGAO

There are three main NGAO requirements documents:

- The Science Case Requirements Document (SCRD): KAON 455 [1].
- The System Requirements Document (SRD): KAON 456 [2].
- The Functional Requirements Document (FRD): KAON 573 and the NGAO requirements database.

There is a fourth requirements document that is referenced by the SRD. This is the Instrument Baseline Requirements Document (IBRD) KAON 572 [3] which contains Keck Observatory constraints and requirements for any new instrument.

[1] Although the V-diagram is a standard systems engineering tool, ours was greatly influenced by those made for the TMT. See article at: http://www.tmt.org/newsletter/focus-0710.html.

The requirements development process can be described as follows. The initial NGAO scientific concept was used to generate the NGAO Science Case Requirements Document (SCRD). The SCRD delineates the anticipated scientific gains of NGAO and motivates its constructions.

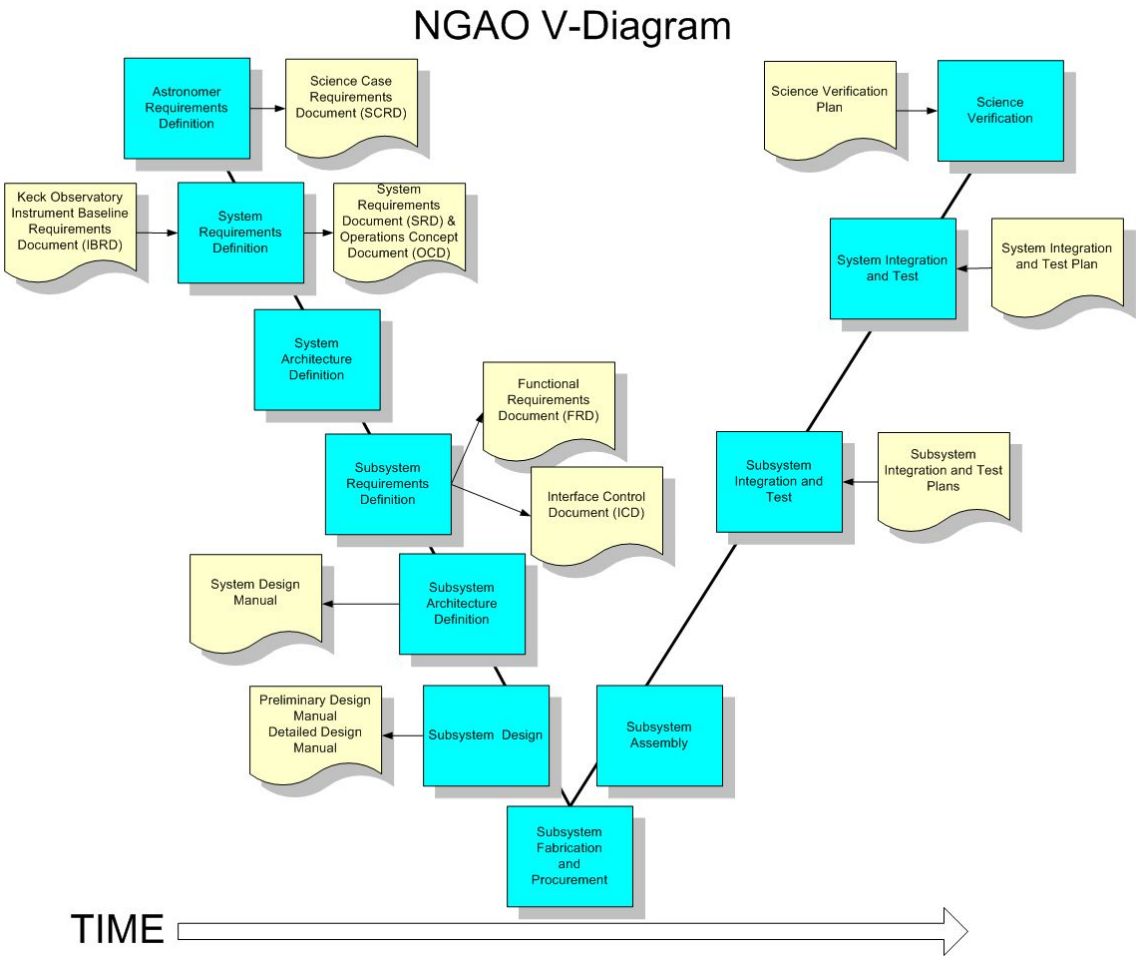

Figure 1: NGAO project V-diagram including the design and development process. Processes are shown in blue and associated documentation in tan.

The implications of these science requirements were then flowed down to requirements on the adaptive optics system and associated instruments. The constraints imposed by the Keck Observatory on new instrumentation are contained in the Instrument Baseline Requirements Document (IBRD). These constraints were also considered along with the science requirements, resulting in the NGAO System Requirements Document (SRD). The science requirements from the SCRD and additional requirements imposed by the Keck Observatory are tabulated in the Overall Requirements section of the System Requirements Document [2]. These overall requirements are then flowed down to discipline-based requirements in the SRD. The requirements document is structured from the common disciplines that make up a typical scientific instrument for observational astronomy.

In the SRD for the NGAO, the disciplines are:

- Optical
- Mechanical
- Electronic/Electrical
- Safety
- Software
- Interface
- Reliability
- Spares
- Service and Maintenance
- Documentation

The SRD, as much as possible, avoids prescribing specific design or implementation solutions as is appropriate before the system architecture is defined.

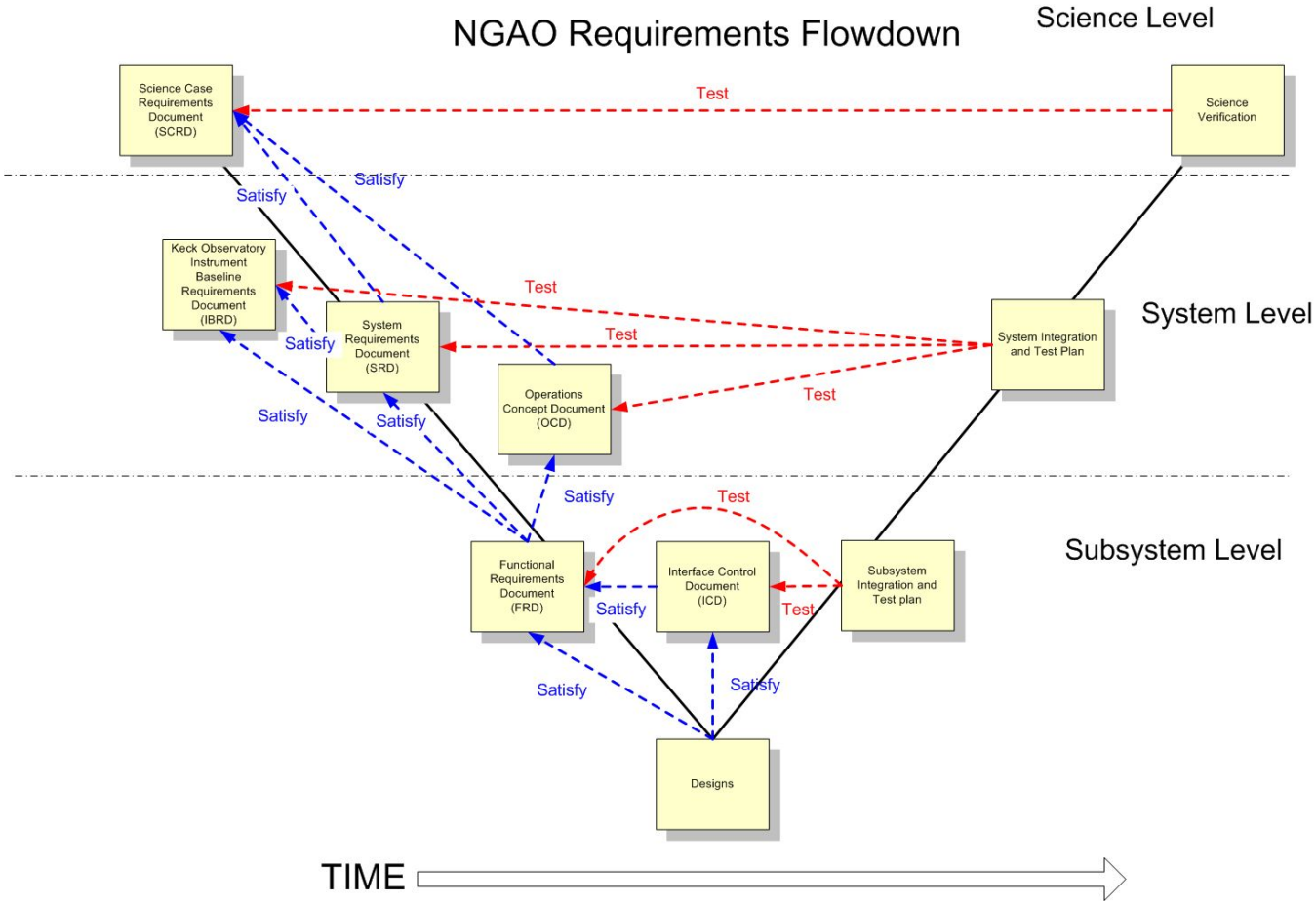

Figure 2: Modified V diagram showing the requirements flowdown and the relationship to testing.

After the system architecture is defined, the Functional Requirements Document is developed. The FRD flows down the requirements from the design-independent SRD to requirements on a few high level subsystems. Note that the flow down of the SRD to the FRD usually entails design choices that could be revisited as the design proceeds. The subsystems were chosen to divide the NGAO system into functions that would be required independent of the selected architecture. These subsystems included the AO system, laser facility, science operations facility and science instruments. The current FRD is consistent with the subdivisions contained in the NGAO System Design phase WBS [4]. For each subsystem there is a breakdown of the requirements by the same engineering disciplines as used in the SRD.

A modified V-diagram is shown in Figure 2; it contains the interrelationship between requirement levels, interface definitions, designs, and the various testing plans. Each requirement must satisfy a higher-level requirement or interface definition, while every test is designed to verify one or more requirements. The FRD provides the criteria against which the subsystems will be evaluated. The SRD provides the criteria against which the NGAO system will be evaluated. The SCRD provides criteria against which the NGAO and its instruments are evaluated.

3. NGAO Requirements Management

The NGAO system design started the configuration management of the SCRD and SRD using word processing tools and the NGAO TWiki page as a means of managing changing versions and traceability between science and system requirements. When the FRD, was started the authors of this KAON came to the realization that the total number of possible requirements and interfaces was likely to be several thousand by the time NGAO and its instruments reached detailed (final) design. Keeping all the levels of systems and subsystem requirements consistent is a large undertaking. One needs to link or relate requirements so that the overall design can be kept logically consistent. Some examples of how the links can be used include:

- Changes in the science (high level) requirements can be traced to effected systems and subsystems (low level) requirements.
- Knowing which low-level requirements satisfy which science requirements can point out over- or under-designed systems.
- Determine how a requirement will be tested.
- Find unassigned or orphaned requirements.

In order to manage the requirements, the authors of this KAON suggested using database software to maintain the requirements. Erik Johansson made a survey of existing requirements management software tools [5] and Contour by Jama Software was selected for a trial evaluation. The results of that trial evaluation are documented in Appendix A: Software Selection for Requirements Management. At present, we continue to use the database to manage the functional requirements. The content of the SCRD and SRD are maintained outside the database but requirements tables from these documents are imported into it.

A short descriptive name and the text of each functional requirement are entered into the database. In addition, the following data is associated with each requirement.

- Why the requirement exists (rationale)
- What requirements are related to it (traceability)
- What reference material supports it (references)How the requirement is checked in the test plan
- Who was the source of the requirement
- Revision history
- Approval status (draft, final, approved, pending, dropped)

4. Status of Requirements

At this time, the Science Case Requirements and System Requirements are fully documented and released [1,2]. They have also been transferred into the database. The first level and lower levels of the Functional Requirements are at various levels of development. The Functional Requirements are being modified to reflect the most recent

changes to the NGAO architecture as the design matures in spring 2008. A deliverable of the subsystem design process is revised functional requirements. Checking and modifying the linking of the various requirement levels will follow. The project plans to complete the System Design level functional requirements by May 2008.

A preliminary report of the functional requirements taken from the database is included as Appendix B: Functional Requirements.

References

1. C. Max, E. McGrath, and P. Wizinowich, "Keck Next Generation Adaptive Optics Science Case Requirements Document," Keck Adaptive Optics Note 455, release 2.1, version 5, (W. M. Keck Observatory, Kamuela, Hawaii, 2008).
2. D. Gavel, C. Max, E. McGrath, D. Le Mignant, and P. Wizinowich, "Keck Next Generation Adaptive Optics System Requirements Document," Keck Adaptive Optics Note 456, version 1.17, (W. M. Keck Observatory, Kamuela, Hawaii, 2008).
3. S. Adkins, "Instrument Baseline Requirements Documents", version 0.1, Keck Adaptive Optics Note 572, (.
4. P. Wizinowich, R. Dekany, D. Gavel, C. Max, S. Adkins, and D. Le Mignant, "Keck Next Generation Adaptive Optics: System Design Phase Systems Engineering Management Plan", Keck Adaptive Optics Note (KAON) 414, (W. M. Keck Observatory, Kamuela, Hawaii, 2006)
5. E. Johansson, "Requirements Management Software for NGAO", Keck Adaptive Optics Note 581, (W. M. Keck Observatory, Kamuela, Hawaii, 2008).

Appendix A: Software Selection for Requirements Management

This appendix was originally an NGAO inter-team memo

From: Christopher Neyman
To: NGAO EC and NGAO Team
Date: November 30, 2007

1. Summary and Recommendation

I propose that we purchase a version of Contour starting with the new v2.0 (release date December 2007). We should attempt to use it to the fullest extent possible during the remaining parts of the system design phase of the NGAO project: roughly December 1, 2007- April 1, 2008. As the system design is completed the NGAO team should re-evaluate our requirements management methods as part of planning the next phases of NGAO. At that time, the decision to use Contour should be reevaluated.

2. Current Status of Requirements Management tool Contour by Jama Software:

After evaluation of eight different software tools by Erik Johansson, Contour by Jama Software was selected for installation and a more thorough evaluation by the NGAO team. This evaluation has been ongoing over the past month. The following subset of the NGAO team spent some time reviewing Contour including: Viswa Velur (CIT), Chris Neyman (WMKO), Elizabeth McGrath (UCSC), David Le Mignant (WMKO), and Erik Johansson (WMKO). As far as I am aware other team members were issued accounts but didn't attempt significant evaluation of the software. Overall, the response to the tool was favorable. It was considered a superior way to collaborate across geographically remote locations as opposed to using MS word and email.

However, we have identified several issues with Contour which include:

- Contour is not compatible with Safari v2.0. Firefox with Mac OS X is a functional work around.
- The glossary can be neither imported nor exported until the Contour 2.0 release (expected December 2007). I created an "NGAO Glossary" report so you can at least output the glossary.

- Two people can edit the same requirement at the same time; there is no check out scheme like other version control software (CVS, Subversion). The software does track all changes and provides an easy way to view them. However, whoever last saves their edits determine the current document state. This may be fixed in later versions of Contour.
- No provision for version control and multiple links to an attached document, expected to be fixed in early 2008.
- Reports (output from the database) in MS word format generate a Java error, should be fixed in release 2.0 (December 2007). HTML format is a partial work around at the present time.

3. Survey of Practices at TMT and JPL

I have made an informal survey of what software the Thirty Meter Telescope and JPL use for requirements management and what their current methods and practices are at this time.

JPL: source Frank Dekens

Software tools: DOORS by Telelogic.

Comments on scope:
JPL does not have a formal project size threshold, but that is probably because flight projects, which are required to use it, are by definition so large that it is needed. On SIM, for example, has 1200 requirements at just the Instrument level (L3) and the requirements run from L1 (mission science requirements) all the way to L6, which are for example optics wave-front-error requirements, etc.

Methods:
JPL has a 1-day course that teaches people how to use DOORS in the morning, and teaches standards so that the requirements are consistent in the afternoon. When a new project is started templates are made by the people that teach the class, so that makes it easy and standardized. Because of that, the overhead of using DOORS is made smaller. Without the JPL support, Frank had no basis of estimating when the overhead of a software tool like DOORS would become worth the effort.

TMT: sources Mark Sirota and Scott Roberts

Software tools: DOOR by Telelogic.
The TMT project started out using Word for requirements, but in the last couple of months has been using DOORS and finds it very helpful. The major issues that prompted converting to dedicated requirements software are tracking changes to requirements and developing traceability. DOORS is a very useful tool for both of these issues. It is set up to allow you to easily track changes to individual requirements and allows you to baseline a set of requirements at a particular time in the project so that you can always see what the requirements were at a certain milestone in the project. It also includes a relational database and good tools that allow you to add attributes to requirements and to relate them to each other. DOORS is also very good at reading requirements from and outputting to Word and Excel.

Comments on Scope:
TMT has been working hard on requirements for the project for the last year. They have a top level science requirements document which flows down into 3 foundation engineering documents: the Operations Concept Document, the Observatory Requirements Document, and the Observatory Architecture Document. These in turn flow down into subsystem requirement documents, of which they will have about 20, and into approximately 100 interface control documents. In addition to this, they will have test plan documents for each of the subsystems and the integrated system. One of the things that they have been asked to do by their external advisory panel is to show the traceability of requirements from the top level science requirements down to the engineering requirements. By doing this they will be able to show which lower level requirements are related to the top level and visa versa. They believe it will be helpful for engineers, managers, and reviewers to trace through this tree to understand why and how the requirements are related, and to understand the key drivers as to why the requirements are what they are.

Methods:
TMT project current approach with DOORS is to keep it internal to the systems engineering group and to maintain the project requirements in Word. DOORS can also create traceability reports and html pages to allow project staff to view the structure of the requirements. TMT believes that implementing DOORS for the entire staff might be useful, but it would require significant training etc.

Both Scott and Mark commented that DOORS is not a silver bullet. To quote Scott, "it is a good tool if you decide to approach requirements from a formal systems engineering approach. I think this is the crux of the issue, in that a project has to define an approach to requirements before you can proceed with any sort of implementation. In TMT it took a significant amount of effort to define our approach to requirements, and required compromise and understanding of the various viewpoints before we could make progress. Requirements engineering is not an insignificant amount of work to undertake and the approach can result in different amounts of effort being invested. Your project as a whole will need to buy into an approach for it to be successful."

4. Recommendations:

Both surveyed groups (TMT and JPL) are using DOORS by Telelogic and find it to be useful for projects the size of TMT and SIM. All people surveyed mentioned the importance of adopting a team wide process and trying to use as much standardization as possible.

Although DOORS appears to be a superior product to Contour, I do not think we have time learn to use it during the next 3 months of the NGAO project, even if it was purchased today. Therefore, our available choices are using MS word or using Contour for the next 3 months of NGAO requirements management.

In comparison to MS word, the cons for the Contour software are:

- Putting requirements in database requires extra work that was not and is not in NGAO plan
- Support costs at host site for installation and maintenance
- Team is not fully trained to use the tool; this will result in some inefficiency
- Contour is a small company, may not be around in a year
- Several issues with current software; these may be fixed during the next release, but no firm commitment from Jama Software at this time
- Export of requirements with available tools in Contour will likely require additional formatting by a person to produce acceptable final requirements documents for the NGAO system design review

The pros for Contour software are:

- Web based: remote access is essentially instantaneous
- Easier to track change than MS Word
- Links for traceability between requirements
- Useful for checking design against requirements
- Seeing impacts of changes, compliance matrices, test matrices
- Costs compared to other software choices are reasonable for NGAO project
- Support is currently very fast and friendly

I proposed to continue with Contour and then re-evaluate requirements management practices and tools as part of planning for NGAO system design management plan in March of 2008.

Appendix B: Functional Requirements