A collaborative systems engineering approach for large instrument consortia

Conference Paper

Author(s):
Glauser, Adrian Michael; Salo, Chad

Publication date:
2020-12

Permanent link:
https://doi.org/https://doi.org/10.3929/ethz-c-000462371

Originally published in:
11450, https://doi.org/10.1117/12.2576341

A Collaborative Systems Engineering Approach for Large Instrument Consortia

Adrian M. Glauser[a] and Chad Salo[b]

[a]Institute for Particle Physics and Astrophysics, ETH Zurich, 8093 Zurich, Switzerland
[b]Leiden Observatory, Leiden University, P.O. Box 9513, 2300 RA Leiden, The Netherlands

ABSTRACT

Large astronomical instruments are often built by consortia of research institutes and universities. The different locations of the various teams, the common interests and shared responsibilities of the partner organizations, and the science driven approach of these projects bring unique challenges to conduct systems engineering efficiently. In this paper we report our positive experience within the METIS consortium that is building one of the three first-generation instruments for the ESO ELT. We developed a novel and fully collaborative systems engineering approach that decentralizes the responsibilities across discipline experts and subsystem providers using a web-based software tool to engineer requirements and interfaces. We discuss the problems that forced us to develop this new approach, describe the new processes and tools, and discuss the benefits, risks, and lessons learned.

Keywords: Systems Engineering, Requirement Engineering, Collaborative Engineering, Software Tool

1. INTRODUCTION

In the area of the extremely large telescopes, the development of science instruments has reached a scale, both in complexity and cost, that exceeds the resources and capabilities of a single organisation. The three first generation instruments of the European Southern Observatory's (ESO) Extremely Large Telescope (ELT), MICADO,[1] HARMONI,[2] and METIS,[3] are being built by large consortia of universities and research institutes which are distributed across different countries. For the development of the second and third generation instruments of the ELT, similar groups of scientists and engineers will be required. While the large size and geographical diversity are common to these projects, their organisational structures and distributions of responsibilities can differ significantly. Consequently, the systems engineering approaches have to be adapted to each team's specific needs based on the distribution of responsibilities.

Building a first generation instrument for a new class of telescope inevitably comes with numerous challenges for systems engineering; changing interfaces, immature top level requirements, and unprecedented technical challenges, just to name a few. Further, with the ELT, ESO requests a higher standard in engineering processes compared to past projects but these processes are partly untested and have shifted as the project evolves. Consequently, structures and methods acquired with previous instrument developments do not necessarily guarantee success for an ELT instrument-sized project. Novel approaches need to be invented.

In the case of METIS, the various partner organisations contribute not only by developing sub-systems and components, but they also provide discipline experts that contribute to the systems engineering processes, in particular requirement engineering, interface management, and system verification. It is the motivation of every partner to develop a scientific facility that delivers ground-breaking science and, consequently, there is an intrinsic interest to build the best possible instrument by contributing to the overall development. The METIS instrument is very complex and only with a collaborative systems engineering approach among dedicated experts can the various challenges be addressed. This became evident at the beginning of the preliminary design phase when classical approaches failed and the project itself was at risk.

In this paper we will give a brief overview of METIS in Sec. 1.1, describe briefly the organisation of the technical team in Sec. 1.2, and illustrate the difficulties that we faced before the implementation of this new approach in Sec. 1.3. We then introduce the requirement engineering process in Sec. 2 that was invented in order

Further author information: Send correspondence to glauser@phys.ethz.ch

Modeling, Systems Engineering, and Project Management for Astronomy IX, edited by George Z. Angeli, Philippe Dierickx, Proc. of SPIE Vol. 11450, 1145016 · © 2020 SPIE
CCC code: 0277-786X/20/$21 · doi: 10.1117/12.2576341

to overcome the difficulties, describe in Sec. 3 how we expanded this methodology to other systems engineering processes, and discuss in Sec. 4 the advantages and disadvantages of such an approach. Finally, we conclude this paper in Sec. 5.

1.1 The METIS instrument

METIS, the Mid-infrared ELT Imager and Spectrograph, will include the following capabilities:

- Imaging in the L/M/N band (3 – 13.5 μm) over a 10.5"×10.5" Field of View (FoV) including:
  - low-resolution slit spectroscopy
  - coronagraphy for high contrast imaging (HCI)
- High resolution (R∼100,000) IFU spectroscopy in L/M over a 0.5"×1" FoV including:
  - a mode with an extended wavelength coverage of $\Delta\lambda_{instant} \sim 300$ nm
  - coronagraphy for high contrast spectroscopy (HCS)
- All observing modes work at the diffraction limit of the 39-m ELT with a Single-Conjugate AO system (SCAO); prospects for laser-supported AO are under study

METIS will combine unprecedented spatial and spectral resolution with superb sensitivity, a combination that no other ground-based or space-based instrument will achieve in this wavelength range in the foreseeable future. Phase B was successfully concluded in June 2020, one year after the Preliminary Design Review (PDR). The currently ongoing final design phase (Phase C) will culminate with the Final Design Review (FDR) in mid-2022, followed by the Manufacturing, Assembling, and Test campaign (Phase D). METIS' 1st light at the ELT is targeted for 2027.

A detailed description of METIS is beyond the scope of this paper and can be found in Brandl et al. (2018).[3]

1.2 The Structure of the METIS Systems Team

METIS is a joint project between NOVA, the Netherlands Research School for Astronomy, the UK Astronomy Technology Centre in Edinburgh, the Max Planck Institute for Astronomy in Heidelberg, the Katholieke Universiteit in Leuven, CEA Saclay in Paris, the University of Cologne, the University of Liège, the University of Michigan, the Academia Sinica's Institute of Astronomy and Astrophysics, a consortium of Austrian as well as Portuguese Universities, and ETH Zurich. The technical development of METIS is led by the systems team that is composed of members from all partner institutes. In Fig.1 we show a generalised management structure of the Systems Team (ST). The ST is led by the Lead and Deputy Systems Engineers who are responsible for the overall systems engineering aspects and for the coordination of the ST. They are supported by discipline specialists, the Systems Leads, who are in charge of the system architecture related to their discipline. Besides the classical disciplines like optics, mechanics, etc., we also have System Leads responsible for aspects that require system overview but can not be devoted to individual subsystems. Examples include high contrast imaging and adaptive optics; these areas require harmonisation of various other disciplines together with dedicated components distributed in multiple subsystems.

The engineers are supported by the Instrument and Calibration Scientists who are also part of the ST as they play key roles in the requirement decomposition for high level science performance requirements.

Each of the 10 subsystems in METIS is led technically by a local system engineer. This person is responsible for the subsystem specifications, interface management, and verification. They also contribute directly to the system level engineering as explained in the following sections. Consequently, they are also part of the ST.

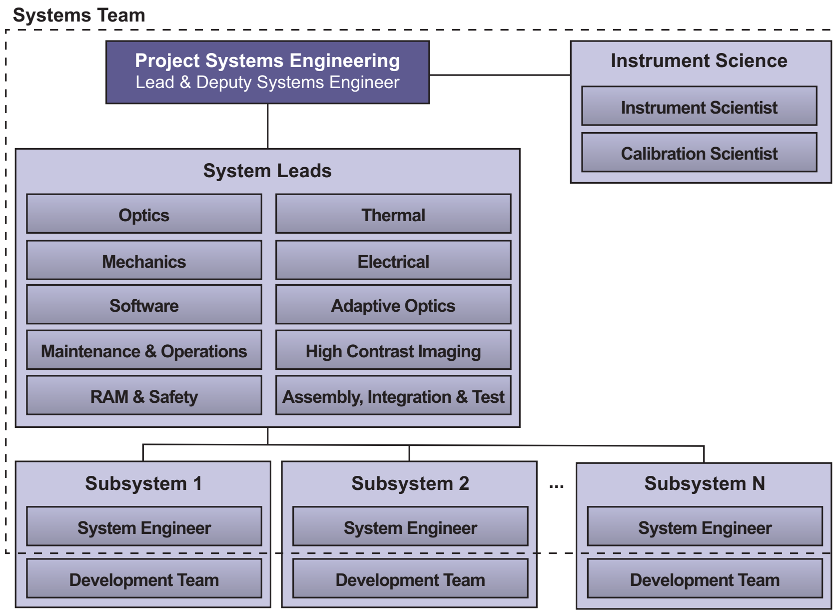

Figure 1. Generalised organisational structure of the METIS Systems Team

1.3 Problems with Classical SE Approaches for METIS

At the beginning of the design phase, the ST was focused on establishing the requirement engineering process and translating the high-level requirements to subsystem specifications. Given the distribution of responsibilities in the team, this task demands a collaborative approach to involve all areas of expertise and responsibilities, both across disciplines and including subsystem providers. Nonetheless, it is the task of systems engineering management to coordinate the analysis and distillation of requirements at system level. Lacking a good collaborative tool, the requirement breakdown was started using an MS Excel spreadsheet that was maintained by a single person (i.e. the Systems Engineer) and iterated offline with the rest of the team. This approach uncovered two issues:

1. The high-level requirements did not allow a direct translation to subsystem level (as often done for ESO projects) and the task to break them up became very difficult and non-linear. In some cases, one high-level requirement breaks in several system-level requirements that are also required to satisfy other high-level requirements. For illustration, we use the example of two high-level requirements that drive the design for METIS:

- **Imaging Contrast**: METIS shall deliver a 5σ contrast at L-band of $3 \cdot 10^{-5}$ at a distance of $5\lambda/D$ from a bright star ($L \leq 6$ mag) within one hour of on-target integration time [...].
- **Optical Quality**: Quality and alignment of the optical components within METIS shall provide diffraction limited performance (Strehl $\geq 80\%$) at $\lambda > 3\mu m$ in all modes over the entire FOV.

The dependencies of the two requirement might be obvious at first sight but, if one derives the resulting performance needs and how they relate to each other, the situation becomes more complex as shown in Fig. 2. One common element of the two high-level requirements is a wavefront error (WFE) budget that allocates WFE to each subsystem. This WFE budget is derived by translating the optical quality needs into a WFE, but also by defining a maximum Non Common Path Aberration (NCPA) that is required to achieve the needed contrast. However, there is a third component shown which is derived by the need to control and remove quasi-static speckles. This is achieved by reading the science-detector data of the nulled star and by conducting Phase Sorting Interferometry[4] to derive the residual NCPA that is used by the Single Conjugate Adaptive Optics[5] to

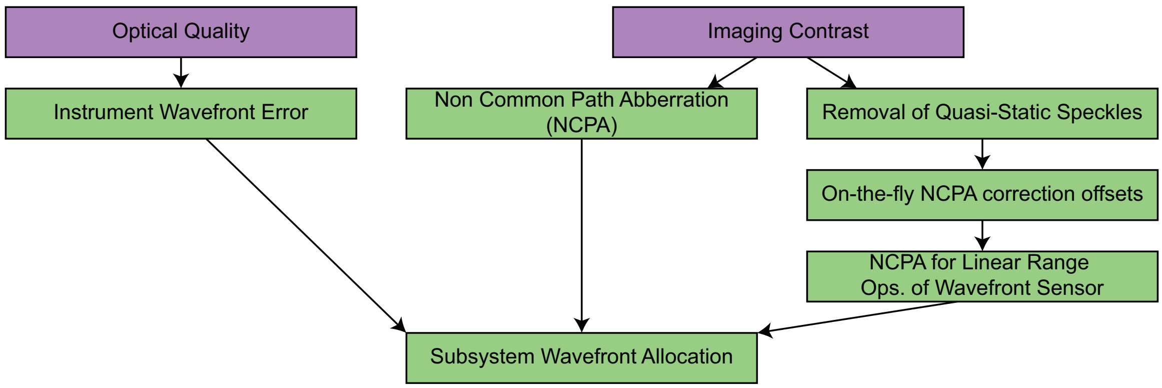

Figure 2. Illustration of dependencies of system requirements (green) to high level (purple) requirements. The arrow illustrates the direction of break-down and how a parent requirement is satisfied if its child requirement is compliant (note that there are many more requirements dependent upon the shown examples).

compensate it using the deformable mirror M4 of the telescope. In order operate within the required performance, the wavefront sensor has to remain within its linear range thus creating another constraint for the WFE budget.

This example illustrates that the WFE budget can only be derived when several high-level requirements are properly broken down and translated. In reality, the budget has many more such dependencies and a direct translation from high-level to subsystem is impossible. Further, it is also impossible to work in a linear structure and to find dependencies across different high-level requirements and disciplines, especially if different experts should be involved in the requirement engineering process. Lastly, establishing and maintaining the traceability between parent and child requirements (indicated by the arrows in Fig.2) becomes a very demanding, if not impossible, task if one works with spreadsheets or simple databases.

2. The example provided illustrates that the WFE allocation to subsystems can only be derived when at least three disciplines work together, represented by the optical, the high contrast imaging, and the AO experts. Given the substantial complexity of the topic, such a requirement engineering process is a product of many iterations and discussions. If the requirement engineering process is managed though the iteration of a single spreadsheet, the induced overhead for the systems engineer to maintain the spreadsheet is enormous if the number of complex topics is as high as it is in METIS. Further, it slows down the iteration speed making the general engineering work inefficient.

For these two reasons, the METIS project faced a serious challenge just a few months into the design phase as the requirement engineering process began to stall. The high volume of requirements, the many loose ends of the requirement discussions, and the not-yet known cross-dependencies between the high-level requirements made it impossible to continue with the original processes.

2. COLLABORATIVE REQUIREMENT ENGINEERING

2.1 Requirement Engineering Processes and the Resulting Needs for a Software Tool

The derivation of the system requirements is a product of many systems engineering processes, as illustrated in Fig. 3: 1. Analysis of the high-level requirements and the resulting break-down of system requirements such that they can be applied to the subsystem level, 2. the logical decomposition of the system functions and the structural architecture 3. the identification of requirement needs driven by design questions from the subsystems, 4. the management of external and internal interfaces, 5. the design loop via subsystem design and analysis efforts in combination with system level design and analysis activities, and 6. additional constraints driven by the planning of the various stages of the product life cycle such as the Manufacturing, Assembling, Integration, and Verification (MAIV), maintenance, operations, instrument calibration, and finally commissioning.

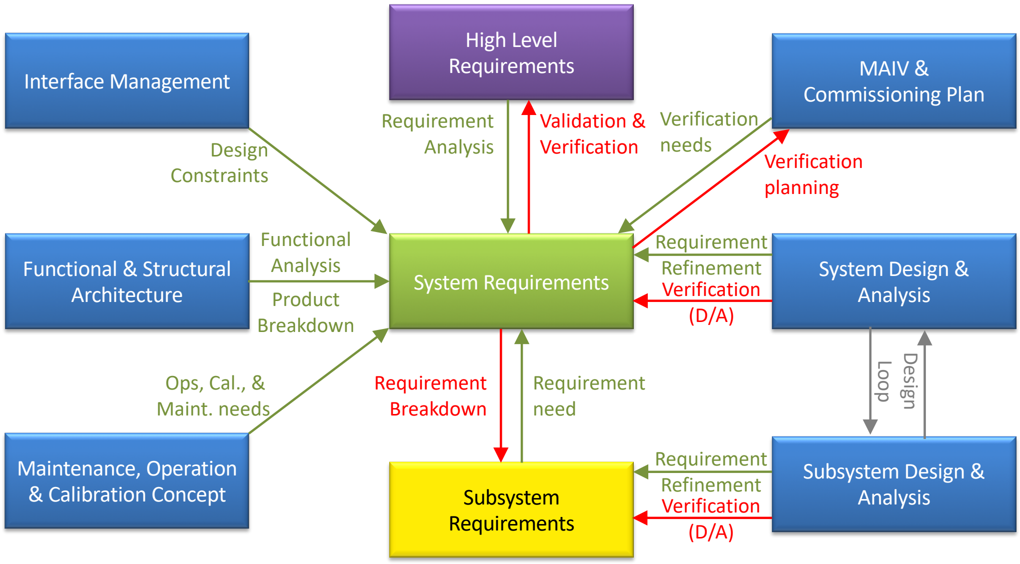

Figure 3. Engineering processes that are directly connected to system requirement engineering.

During the design phase, all of these processes run in parallel. Various experts are involved and the subsystem providers actively participate in the identification of requirement needs and contribute to the design loop iteration with the system that ultimately leads to updates of budgets and system requirements. Consequently, a large team actively contributes to the requirement engineering (in the following refereed to as the ST, but it could be generalised to the Requirement Engineering Team). In order to facilitate an efficient process for requirement engineering, a tool and a workflow are required satisfying the following criteria:

1. The tool shall contain a database for requirements and its auxiliary meta-information
2. The tool shall provide simultaneous access for all ST members
3. The tool shall contain information for traceability and requirement dependencies
4. The tool shall generate requirement documents
5. The tool shall support the implementation of a workflow as specified below:
   (a) The workflow shall enable any ST member to propose new requirements or to request changes of existing requirements
   (b) The workflow shall enable the Systems Engineer (SE) to keep full control over the addition and implementation of all proposed system requirements
   (c) The workflow shall enable any requirement stakeholder to review and approve the system requirements

2.2 Workflow for System Requirement Engineering

The solution for the criteria concerning the workflow is illustrated in Fig.4. Any ST member identifies the need for a new requirement based on their activity and field of responsibility. Most likely, they identify a small group of people who discuss the issue and work together on the derivation and formulation of the requirement. Once this group of experts is satisfied with the requirement, one of them submits the requirement to be checked by the SE. During this check, the SE ensures that the requirement is well written and not in conflict with any other requirements already specified, that all formalities are correct (such as the identification of applicable subsystems, the dependencies to other requirements, etc.), and that the provided rationale is consistent with

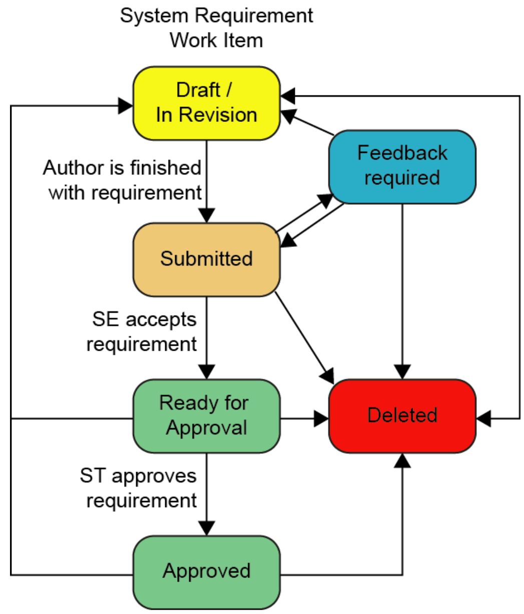

Figure 4. Workflow for collaborative requirement engineering

the system architecture. If any further iteration with the author team is needed, the SE requests an update or clarification and sends the requirement back to the authors for further improvement. Once the SE accepts the requirement, a formal review by all stakeholders (identified by the authors and SE) is conducted. During this review, a requirement can be rejected and sent back to the authors for further improvements. If all stakeholders within the ST agree with and approve the system requirement, the requirement enters the configuration controlled baseline of the system and is officially adopted.

During any time in the project, this workflow can be repeated should a system requirement need to be changed. The formal approval process provides an intrinsic change control.

2.3 Software Tool

In principle, any online software for collaborative work could satisfy the first criterion and, one could argue that, online spreadsheets could satisfy criteria 1-3 (or even 4 if a spreadsheet is converted into a document at the end of the engineering task). However, criteria 5-8 are more problematic given that not all online tools provide the implementation of workflows combined with selective editing privileges depending on the roles and the status of a requirement.

Our tool of choice was the Application Lifecycle Management tool from Siemens*, Polarion© ALM™. In our evaluation for the right tool, we also considered alternative products such as IBM DOORS Next Generation but Polarion had the most flexibility for adaptation to our needs, in particular, the ability to customize workflows and the meta-data information. Polarion is a web-based, version controlled, xml database system that allows inclusion of individual database entries, so called Work Items (e.g. the individual requirements), into a live-document that can be exported and released like a classical document. The user interface is a standard web browser. The majority of the work is conducted in a form view of the work item where all relevant meta-data is visible. Different types of work item can be defined each with an individual set of fields, views, and workflows. A work item can be assigned to one or multiple users that act as authors. Each work item can have customizable relationships to other work items. Polarion also allows scripting to produce documents based on dedicated database queries. This became very handy to automatically produce verification control documents, for example (see Sect. 3.1). The revision control of Polarion provides the useful feature to view the change history of any

*https://polarion.plm.automation.siemens.com/products/polarion-alm work item. Further, comments can be added to any work item which allows a ticket-based conversation within the team.

2.4 Tool Configuration

We started with defining three different types of requirement work items, one for high-level requirements, one for system requirements, and one for subsystem requirements. At a later stage, we also defined a work item for component requirements for cases where a dedicated component specification is required. The main purpose of the different work item types is that the different types of meta-data can be defined and that access privileges can be set individually to specific roles, e.g., we did not want the ST to be able to change high-level requirements, and having only partial writing rights to system requirements. These four levels of work items are connected via a dependency, "is satisfied by", to link parent requirements to their children. This provides the required decomposition traceability of high-level to lower-level requirements. The traceability is essential for change control and to assess the impact of non-compliancies of requirements to higher levels. Fig. 5 shows the different work items and their dependencies.

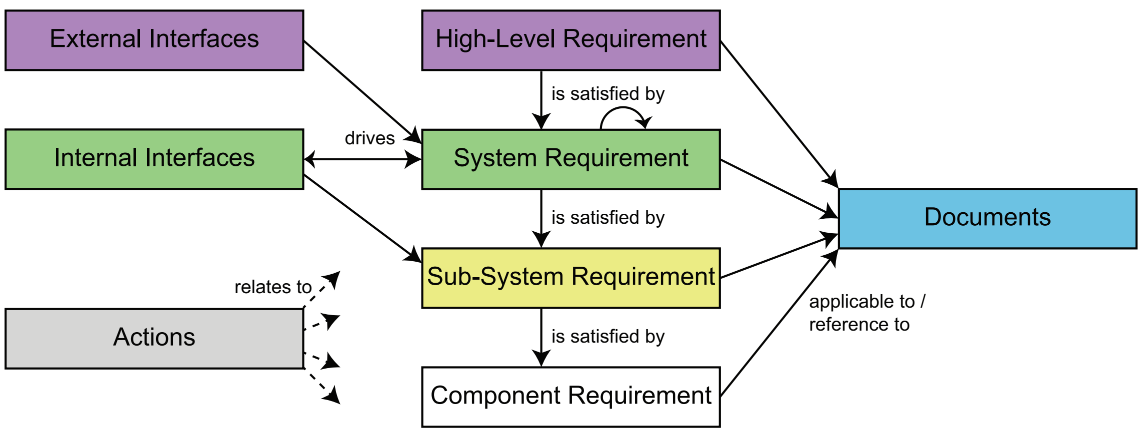

Figure 5. Work items and their dependencies for the Polarion implementation of the METIS project

We also identified the need to establish a work item describing external interfaces (i.e. to the telescope), as some are directly driving the design of the instrument. Consequently, some system requirements are impacted in case an external interface would change. Given that METIS is a first generation instrument, changing and maturing interfaces are a reality.

During the design process, internal interfaces between subsystems are being defined. In some cases, these interfaces constrain the system design and this dependency also needs to be reflected. Consequently, we established another work item type for internal interfaces and made use of the Polarion workflow capability to manage the interface approval process (see Sect. 3.3).

Common to all levels of requirements is the need to include references to other documents, either to provide more context in the requirement itself, or in its rationale. The final product of the requirement engineering process is a released specification document. As is typical for engineering documents, a list of applicable and referenced documents is provided at the beginning (or at the end) of the document in order to allow the reader finding the corresponding references. We implemented a script that generates these lists based on the dependencies of work items to documents. Therefore, we established another work item type that we call "reference item", that contains the meta-data of a document. The dependency between requirements and documents discriminates if a document is just a reference or if it is applicable to the requirement. Consequently, the reference will appear in either of the two reference tables of the specification, for applicable or referenced documents, respectively.

A final work item was introduced that is linked to any other type of work item: Actions. With the capability of Polarion to define customized workflows and the possibility to assign work items to users, Polarion became the natural tool for action management. Having the links to requirements, interfaces, or even documents became very handy during the requirement engineering process. Of course, the workflow for actions is different from requirements (open to closed) but it allows to maintain a direct relationship between actions and their related work items.

3. COLLABORATIVE SYSTEMS ENGINEERING

While we introduced Polarion primarily for requirement engineering, the team's experience with this collaborative working approach via an online tool was very positive. We decided to extend the functionalities of our tool to include other processes.

3.1 Verification Control

For each requirement work item type, we implemented additional fields to define and control the verification activities. Once the requirements were defined, we added the verification method(s) based either on the needs given via the high-level requirements, or by an assessment by the Systems Engineers. We also defined the verification level and, in some cases, provided a more detailed verification plan. We defined a verification status and assigned team members to take on the responsibility for the verification documentation.

Once this preparation was completed, the individual team members had the full functionality of the Polarion database at hand to maintain an overview of the verification tasks they needed to complete. During the documentation of the verification, a link to the corresponding verification evidence had to be provided (e.g. by pointing to an external report) and the compliance status was indicated.

We manufactured script based documents that automatically generated the verification control matrices and documents for the various specifications based on the provided meta-data in the work items. Using the traceability that was established during the requirement engineering process, it was then very easy to flag partial or non-compliance of system and high-level requirements in case a subsystem reported an issue.

For the MAIV phase, we plan to extend the functionality to add work items for specific test activities that will also be included in test plan and test report documents.

3.2 Configuration Management

As explained in Sec. 2.4, we introduced reference items to point to documents from within requirements. Due to the lack of a proper document management system, we decided to also use Polarion for the document management process, in particular, to control the revision and approval process. The individual reference items contain a hyperlink that links to an external server where the documents are stored. The Polarion database contains all the required meta-data such as version, document number, authors, and approvers. A dedicated workflow was invented to ensure that the documents were reviewed and released in a controlled way. Thanks to the revision control of the Polarion database, at any given moment in the project life cycle, we are able to extract the approved configuration of the system thus providing consistency between revisions of requirements and documents. This eliminates the need for manually maintaining Configured Item Data Lists (CIDLs) as each reference item also points to the corresponding product number making it easy to extract product-specific CIDLs.

3.3 Interface Management

Finally, we also implemented the Interface Control Documents (ICDs) in Polarion for internal, subsystem to subsystem, interfaces. Each interface characteristic is described in a dedicated work item. Once the engineering process is completed, each work item gets approved via a dedicated workflow by the two Subsystem Engineers and the System Lead responsible for that type of interface characteristic (e.g. mechanical, optical, etc.). The completed ICDs can then be exported as pdfs and will be approved at the managerial level.

4. DISCUSSION

Implementing a collaborative online tool for requirement engineering in particular, and for many other system engineering processes in general, proved to increase the quality of work and to enhance the efficiency of a large team that has to work together for a complex instrument like METIS. The key advantages compared with classical approaches are summarised in the following.

4.1 Availability of Information

During the early development phase, the functions and the conceptual design might not be well defined. Many fundamental aspects of the instrument can change and are in continuous development. If the instrument is complex, subsystems but also disciplines depend strongly on each other and the work can not be executed independently. If the information within the large team is not constantly exchanged, the team members risk to create loose ends and might miss a development that happened somewhere else in the project.

Polarion provided us with a central repository of knowledge where everyone had simultaneous access at any time. The team had to get accustomed to the idea that "what is not in the database is not a part of the instrument". Once this concept was established, it became very efficient to store and administrate the team's knowledge about the instrument. Inconsistencies that existed before the implementation of Polarion were uncovered and resolved. Combining the requirement work with action management allowed documenting all loose ends and to systematically remove them.

4.2 Simplified Communication

To state the obvious, it is very important that communication is sound and inclusive. However, this comes with significant overhead for a large team as many meetings are required, especially during the phase when many aspects are not yet documented. Written information is often lost in lengthy email threads where either too many or too few people are involved.

In addition to being a central knowledge repository, each work item includes comments that can be used to document a discussion, to provide feedback, and to replace iterations otherwise done via email. In this way, the communication is stored at the very location where it is relevant and it is accessible to all team members for the entire duration of the project.

Additionally, the implementation of the review and approval processes in the workflows reduced the number of review meetings required as most of the review was done remotely at the convenience of the individual team members. We required face-to-face meetings only for complex aspects that can be addressed best in a group discussion.

One downside of the online work is that people need to be made aware when an action is expected from them. For example, ST members need to be informed if a requirement is ready for their review, authors need to be informed if a requirement is rejected by the SE for further improvements, etc.. Users that access the Polarion server very frequently have customized their personal home screen to indicate statistics of the pending work. Other users do not login on a daily basis, hence, a notification email is helpful to make them aware of pending work. Such notification emails have a tendency to be ignored if they come in at a too high frequency, an unavoidable situation shortly before major releases of the requirement documents. Consequently, other communication channels, such as regular video calls, are needed during peak activity times in order to ensure that the team is sufficiently aware of the remaining work.

4.3 Automation of Tasks

Using offline tools often requires manual work for editing and streamlining between different documents. A good example is the VCD that often requires substantial effort, especially if the information across multiple levels (i.e., subsystem, system) needs to be synchronized. Setting up the scripting feature in Polarion to generate the VCD based on the database entries also required work. However, now that this initial investment has been made, the VCDs can be generated automatically and remain consistent at any time of the project. For METIS, we extract VCDs for all major reviews, at least 4 times during the project life cycle. Consequently, this automation feature saves a lot of effort and prevents mistakes that occur from manually editing large spreadsheets.

4.4 Mapping Complex Dependencies

The relationship between requirements is classically illustrated by providing upper or lower links in tables that accompany the requirement documents. The maintenance of such tables would be a tremendous amount of work. In Polarion, we made use of another script that generates these tables automatically for the extracted documents. And for the work within the database, the individual work items are displayed with their dependent parents and children. This makes the browsing through the requirement structure very simple.

However, at the present stage, Polarion does not offer a good visualisation of entire requirement trees. For this purpose, we had to generate our own code in Python that accesses the database from the outside and generates nodal diagrams using the graphviz library.

4.5 Risks and Lessons Learned

Developing a system requirement document using a collaborative online tool within a large team imposes also some risks.

First of all, different members of a team get accustomed to new software tools and processes at different rates. During the roll-out of Polarion within the METIS team, this soon became an issue. When changing from classical office software to a web-based database, some standard processes appear to be more cumbersome and less efficient than what people have been used to and the usefulness of such a tool might be put in question. Over time, the benefit of using Polarion became clear to all team members and the willingness to overcome the initial difficulties and resistances increased. However, we identified that the benefit might be less obvious for smaller and less complex projects and a careful trade is required before implementing such a system.

Second, the distributed responsibility for system requirement engineering requires all involved team members to invest a significant fraction of their work for this activity. This might be in conflict with other tasks and, while requirements are of high priority for systems engineers, other team members might be less willing to give the same level of attention to the requirement work. Collaborative work requires cooperation and, if even one team member falls behind of their tasks, the entire process is affected. Consequently, the success of the collaboration depends on the performance of the weakest link. If a team is performing well, like in METIS, such difficulties can be overcome. But for new teams such an approach bears the risks for the overall success of systems engineering.

Finally, having many authors of system requirements comes at a price that the different authors have different writing styles and implement a different logical structure in the requirement definition and decomposition. It does require a substantial amount of work to streamline the requirements and to build up a consistent specification. METIS currently has 400 system requirements and it is fair to say that we had to make some compromises and deviated in some cases from the text-book style for requirement. As the system requirements serve primarily the consortium and are not used as a specification for industry, this compromise is acceptable as all stakeholder understand the intent and scope of the systems requirement. Consequently, we believe that the presented collaborative systems engineering approach is well suited for science driven instruments like METIS, but it might be less suited for industrial projects.

5. CONCLUSIONS

We presented a collaborative systems engineering methodology that was introduced in the METIS project to enhance the efficiency and quality of the requirement engineering of system requirements. We extended this method to other processes such as interface management, verification control, configuration and action management. We use Polarion© ALM™, the web-based Application Lifecycle Management tool from Siemens. However, the presented methodology based on a database with a controlled workflow could work in principle with any software tool that provides similar functionality.

We are convinced that for complex science driven instruments that are built by a large consortium of research institutes, such an approach significantly enhances the project success as it allows maximizing the interaction and collaboration of the various experts. However, this method does not replace the need for a single entity, i.e., the systems engineer, who maintains and establishes the overview of the system architecture and requirement.

ACKNOWLEDGMENTS

We would like to thank the METIS Systems Team and the entire consortium for the great support during and after the implementation of new tools and processes for systems engineering that were unknown to all team members before. Without the team's eagerness to embrace a new tool, the METIS systems engineering approach would not have been appraised by ESO during the Preliminary Design Review as "exceptional".

REFERENCES

[1] Davies, R., Alves, J., Clénet, Y., Lang-Bardl, F., Nicklas, H., Pott, J. U., Ragazzoni, R., Tolstoy, E., Amico, P., Anwand-Heerwart, H., Barboza, S., Barl, L., Baudoz, P., Bender, R., Bezawada, N., Bizenberger, P., Boland, W., Bonifacio, P., Borgo, B., Buey, T., Chapron, F., Chemla, F., Cohen, M., Czoske, O., Déo, V., Disseau, K., Dreizler, S., Dupuis, O., Fabricius, M., Falomo, R., Fedou, P., Förster Schreiber, N., Garrel, V., Geis, N., Gemperlein, H., Gendron, E., Genzel, R., Gillessen, S., Glück, M., Grupp, F., Hartl, M., Häuser, M., Hess, H. J., Hofferbert, R., Hopp, U., Hörmann, V., Hubert, Z., Huby, E., Huet, J. M., Hutterer, V., Ives, D., Janssen, A., Jellema, W., Kausch, W., Kerber, F., Kravcar, H., Le Ruyet, B., Leschinski, K., Mandla, C., Manhart, M., Massari, D., Mei, S., Merlin, F., Mohr, L., Monna, A., Muench, N., Müller, F., Musters, G., Navarro, R., Neumann, U., Neumayer, N., Niebsch, J., Plattner, M., Przybilla, N., Rabien, S., Ramlau, R., Ramos, J., Ramsay, S., Rhode, P., Richter, A., Richter, J., Rix, H. W., Rodeghiero, G., Rohloff, R.-R., Rosensteiner, M., Rousset, G., Schlichter, J., Schubert, J., Sevin, A., Stuik, R., Sturm, E., Thomas, J., Tromp, N., Verdoes-Kleijn, G., Vidal, F., Wagner, R., Wegner, M., Zeilinger, W., Ziegleder, J., Ziegler, B., and Zins, G., "The MICADO first light imager for the ELT: overview, operation, simulation," in [*Proceedings of the SPIE*], Takami, H., Evans, C. J., and Simard, L., eds., 107021S, Max-Planck-Institut für extraterrestrische Physik (Germany), SPIE (July 2018).

[2] Thatte, N. A., Clarke, F., Bryson, I., Shnetler, H., Tecza, M., Fusco, T., Bacon, R. M., Richard, J., Mediavilla, E., Neichel, B., Arribas, S., Garcia-Lorenzo, B., Evans, C. J., Remillieux, A., El Madi, K., Herreros, J. M., Melotte, D., O'Brien, K., Tosh, I. A., Vernet, J., Hammersley, P., Ives, D. J., Finger, G., Houghton, R., Rigopoulou, D., Lynn, J. D., Allen, J. R., Zieleniewski, S. D., Kendrew, S., Ferraro-Wood, V., Pécontal-Rousset, A., Kosmalski, J., Laurent, F., Loupias, M., Piqueras, L., Renault, E., Blaizot, J., Daguisé, E., Migniau, J.-E., Jarno, A., Born, A., Gallie, A. M., Montgomery, D. M., Henry, D., Schwartz, N., Taylor, W., Zins, G., Rodríguez-Ramos, L. F., Cagigas, M., Battaglia, G., Rebolo López, R., Hernández Suárez, E., Gigante-Ripoll, J. V., Piqueras López, J., Villa Martin, M., Correia, C., Pascal, S., Blanco, L., Vola, P., Epinat, B., Peroux, C., Vigan, A., Dohlen, K., Sauvage, J.-F., Lee, M., Carlotti, A., Verinaud, C., Morris, T., Myers, R., Reeves, A., Swinbank, M., Calcines, A., and Larrieu, M., "The E-ELT first light spectrograph HARMONI: capabilities and modes," in [*Proceedings of the SPIE*], Evans, C. J., Simard, L., and Takami, H., eds., 99081X, Univ. of Oxford (United Kingdom), SPIE (Aug. 2016).

[3] Brandl, B. R., Absil, O., Agócs, T., Baccichet, N., Bertram, T., Bettonvil, F., van Boekel, R., Burtscher, L., van Dishoeck, E., Feldt, M., Garcia, P. J. V., Glasse, A., Glauser, A., Güdel, M., Haupt, C., Kenworthy, M. A., Labadie, L., Laun, W., Lesman, D., Pantin, E., Quanz, S. P., Snellen, I., Siebenmorgen, R., and van Winckel, H., "Status of the mid-IR ELT imager and spectrograph (METIS)," in [*Proceedings of the SPIE*], Takami, H., Evans, C. J., and Simard, L., eds., 107021U, Leiden Univ. (Netherlands), SPIE (July 2018).

[4] Kenworthy, M. A., Absil, O., Carlomagno, B., Agócs, T., Por, E. H., Bos, S., Brandl, B., and Snik, F., "A review of high contrast imaging modes for METIS," in [*Proceedings of the SPIE*], Takami, H., Evans, C. J., and Simard, L., eds., 10702A3, Leiden Observatory, Leiden Univ. (Netherlands), SPIE (July 2018).

[5] Bertram, T., Absil, O., Bizenberger, P., Brandner, W., Briegel, F., Cantalloube, F., Carlomagno, B., Cárdenas Vázquez, M. C., Feldt, M., Glauser, A. M., Henning, T., Hippler, S., Huber, A., Hurtado, N., Kenworthy, M. A., Kulas, M., Mohr, L., Naranjo, V., Neureuther, P., Obereder, A., Rohloff, R.-R., Scheithauer, S., Shatokhina, I., Stuik, R., and van Boekel, R., "Single conjugate adaptive optics for METIS," in [*Proceedings of the SPIE*], Schmidt, D., Schreiber, L., and Close, L. M., eds., 1070314, Max Planck-Institut für Astronomie (Germany), SPIE (July 2018).