ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA011

SCALING AGILE FOR THE SQUARE KILOMETRE ARRAY

M. Bartolini*, N. Rees, M. Miccolis, J. Santander-Vela, L.R.S. Brederode,
M. Deegan, SKA Organisation, Macclesfield, United Kingdom

Abstract

The Square Kilometre Array (SKA) Organisation (SKAO) is responsible for the design and construction of the first phase (SKA1) of its vision: designing, construction, and operating telescopes with an equivalent collecting area to one square kilometre. The SKA1 project is finishing its pre-construction phase in December 2019. A bridging phase was kicked-off before construction commences during which lean-agile processes, structures, and practices are being prototyped. By the end of the bridging phase we plan to have pivoted from a document based, earned value, stage gated set of processes arranged around pre-construction consortia to a code based, value flow driven, lean-agile set of processes unified around the Scaled Agile Framework. During the bridging process we have onboarded more than 10 agile development teams. In this paper we describe the processes, the main technical and cultural challenges, and the preliminary results of adopting a lean-agile culture within the SKA Organisation.

PROJECT CONTEXT

The Square Kilometre Array Phase 1 (SKA1) project [1] is approaching a system-level Critical Design Review (CDR) in early December 2019. The design phase has been conducted after partitioning the SKA system on different subsystems or *elements*, and different consortia have developed a design for different subsystems of the telescope. Most elements have now passed their element-level CDR, increasing the confidence in the overall maturity of the design, but the process also highlighted some system-level issues [2]. A bridging phase has now been initiated, covering the period between element-level CDRs and the start of the construction phase. In this phase those issues are being tackled by adopting a system level approach, as the pre-construction consortia dissolved after their CDR.

The Need for an Agile System Approach

As indicated, the element-level design reviews highlighted several system-level issues, with emphasis on the lack of a single system-level perspective. Some inconsistencies have been identified in the interfaces between subsystems, there are some interfaces with not clearly demarcated responsibilities, and there are inconsistencies due to conflicting assumptions between the parties of an interface.

Compounding the issue, the design phase, and specially the way it has been targeted and delivered, has been strongly document-based. There have been several modelling attempts in order to provide a single source of truth for designs, but the final deliverable has always been a document. This is specially problematic for software components, as software architectures need validation by means of prototyping, with particular attention to their data exchange interfaces.

Upon reflection, it appears that the principles stated in the Agile Manifesto [3] well describe this situation:

- **Individuals and interactions** are needed in order to take a system approach and bridge the communication between different elements; currently that communication is mediated by formally issued documents such as Interface Control Documents (ICDs), but they remember difficult to interpret for many software engineers.
- **Working software** is needed in order to validate many assumptions in the design via evolutionary prototyping. This highlights the need for an integrated approach to software development.
- **Customer collaboration** is reflected in the necessity of a major interaction with the users of the telescope, starting from the preliminary design phase. This interaction is needed to drive the software development in an iterative fashion.
- **Responding to change** is essential for a project that still has many unknowns. Scientific projects are a complex endeavour and the SKA project is a complex system. In complex systems, the interactions between different parts resolve and identify patterns that cannot be traced, or even disappear, when we examine the parts in isolation. The complexity will evolve also in time with the increased understanding of the system itself.

Technical Implementation

The Agile principles do not live in isolation, and in recent years a set of technical best practices have emerged in the software world [4], trying to leverage agility through software craftsmanship in order to increase the quality of the software developed. Chief amongst these are:

Continuous Delivery [5] is the ability to get changes of all types – including new features, configuration changes, bug fixes and experiments —- into production, or into the hands of users, safely and quickly in a sustainable way. Addressing the development of a complex system incrementally and with a cohesive approach, greatly benefits from continuous integration and deployment of the different software components.

DevOps [6] is an essential practice that eliminates bottlenecks in the software development life-cycle, thereby making continuous delivery possible. DevOps promotes a process where teams can make decisions about their products by being responsible for their entire life cycle. This is essential

* m.bartolini@skatelescope.org

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA011

in a project like SKA, where different teams have a very deep knowledge of specific technical domains.

Why SAFe?

While the agile principles can be applied to a single software development team, the SKA Organisation (SKAO) needs a way to leverage those consistently across different teams and domains, while still maintaining control over the entire project development. A number of frameworks have been proposed by the industry to scale agile practices at an organisational level. The Scaled Agile Framework [7] (SAFe®) is a knowledge-base of proven, integrated principles, practices, and competencies that empowers organisations to successfully transform to Lean, Agile, and DevOps. Combining the power of Agile with Lean product development, DevOps, and systems thinking, SAFe provides synchronized alignment, collaboration, and delivery for multiple Agile teams.

The SAFe framework was selected as a basis to coordinate the software development efforts during construction. However, given the need for a culture change, it was also decided to start using it for the SKA bridging activities related to software development, allowing us to prototype the implementation of the collaboration structure for the impending construction phase.

SAFe is appealing to the SKAO not just because of the software development, but also because of the Lean Engineering principles, encompassing different engineering domains. Moreover, SAFe leverages Built-in quality, Continuous Delivery, and DevOps as first class citizens of the framework, making it easy to support those concepts within a large organisation. The maturity of the framework, and its widespread adoption within industry make it an ideal choice when it comes to adopting a robust set of practices that can ease the collaboration with industrial partners.

PREPARE THE TRANSFORMATION

The SAFe framework provides a roadmap conceived to ease the implementation and the adoption of the framework within a company.

Training Change Agents

The essential starting point is represented by a group of early adopters and supporters. This tipping point was reached by the agreement between the SKAO computing group and representatives for all the more software-centered pre-construction consortia. As the group was formed, a formal training session was held at SKA Global Headquarters (see Fig. 1). This 4-day training session was essential in shaping the future organisation, as it provided a common understanding of the framework vocabulary and core values for all stakeholders.

The training was also the perfect opportunity to highlight the main reasons why a shift towards a Lean Agile approach was necessary and to reflect about these reasons in the light of the larger SKA collaboration, addressing concerns from

Figure 1: SPC trainees at SKA Global Headquarters.

project's stakeholders and partner countries. As a result of the training 11 delegates were certified as SAFe Program Consultants (SPC), forming the change agents to train others in the adoption of the SAFe framework, essentially spreading the knowledge throughout the SKA collaboration.

Aligning to a Common Vision

As a result of the first training, it was very clear that SKAO was embarking on a major transformation. Not only technical aspects, but managerial, political, and cultural aspects are essential to this transformation. A vision statement for the bridging phase was developed as one outcome of the training event, and it was used to drive the upcoming transformation process (extract):

> The SKA Enterprise Objective over the bridging period is to successfully reach a milestone where construction can begin on a well-funded project to build the two SKA1 telescopes under the guidance of the SKAO. To do this we must pass the intermediate milestones of:
>
> - Presenting a credible design at System CDR - which serves as a solid foundation for:
> - Submitting a convincing construction proposal - which serves as the mechanism for:
> - The SKA Observatory Council releasing funds for construction.
>
> This is inherently, and unavoidably, a document-based, stage gated process. To be successful we must align this with our vision for software development, which is that: *by the end of the bridging period we will have pivoted from a document based, earned value, stage-gated set of processes arranged around pre-construction consortia to a code based, value flow driven, lean-agile set of processes unified around the Scaled Agile Framework.*

Training the Leaders

For any significant change to succeed within any organisation, it is essential that the organisation's management and

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA011

wider stakeholders' group support this change. A group of early adopters is beneficial, but it is limited in what it can achieve. A series of workshops was organised within the SKA HQ and with relevant stakeholders outside the office, presenting ideas that emerged during the first SPC training event. This step was essential, as it brought the whole transformation process a level up, exposing it to the management concerns. This process also had the benefit of exposing the SAFe framework implementation to other non-software related teams and disciplines within the project, where those interface with the software world. Capturing their concerns bootstrapped a series of conversations on aspects such as contract negotiation and procurement, schedule and project control that had not emerged in previous conversations. Setting up these sort of events is also very beneficial in communicating that the company is supporting the transformation initiative.

IMPLEMENTATION

The SAFe framework supports different configurations, allowing the framework to scale with the number of teams and departments within an organisation. The SKAO started this transformation process by targeting the adoption of what is called "Essential SAFe" [8], that is basically a structure coordinating a limited number of agile development teams contributing to a common value stream. Many practical steps were necessary to implement this sort of change, briefly summarised as:

- Identify the **Agile Release Train** (ART) that is the team-of-teams that will be working together towards a common goal.
- Prepare and prioritize a **backlog of features** that will be the focus of the first program increment.
- Identify one **Release Train Engineer** (RTE) that is responsible for acting as a process authority for the entire release train.
- Understand the set of **teams** comprising the ART and their composition. The collaboration started with 4 development teams plus one system team that is dedicated to supporting the infrastructure.
- As the framework suggests to adopt **Scrum** [9] as a process to drive the team-level work, make sure that all teams are familiar with it and properly trained. All the essential roles of Scrum Master and Product Owner must be covered.
- Define the cadence adopted by the Agile Release Train. SAFe heavily revolves around the concept of **Program Increment** (PI). A PI is a predefined series of development sprints that are planned together by all teams during the PI planning events. See more in the following section.
- Set-up a minimal **supporting infrastructure**. SAFe organises the development in terms of features and stories. A collaborative tool is needed where teams can be assigned features that get decomposed into stories in order to be implemented. The set of IT tools supporting the SAFe process is very wide, and the SKA Office evaluated a number of those via demos and interaction with vendors. Atlassian Jira and Confluence were adopted as a result of the evaluation process.

Get Ready for PI Planning

One of the main ideas that differentiates SAFe framework with respect to other scaled-agile approaches, such as LESS, or Scrum of Scrums, is the concept of Program Increment (PI), and Program Increment planning. A PI is a time-boxed period defined as a fixed number of development sprints plus one *Innovation and Planning* (IP) sprint during which the planning for the subsequent PI takes place. The SKAO adopted a somewhat unusual cadence of 13 weeks, composed of five development sprints of two weeks each, plus a three-week-long IP sprint. This cadence evenly divides the solar year in four PIs, and takes into account all of the major holidays in the states that participate in the SKA collaboration. Due to the longer IP sprint, more time is allowed for buffering capacities, and more preparatory planning can occur. At the end of each PI, all teams gather together for the PI planning [10] event (see Fig. 2). During PI planning every team plans its activity for the next PI. Each team is empowered to pull features from a prioritized program backlog, and then commit to deliver a set of related objectives during the development sprints, according to their planned work capacity.

Figure 2: The first PI planning participants at SKA HQ.

Having all the teams plan their sprints together, facilitates conversations and promotes the prompt identification of key dependencies between teams, so that the resulting planning takes those dependencies into account. The PI planning event lasts for two or three entire days, and produces a set of objectives to which every team commits, along with a Program Board (see Fig. 3), that is a the agreed schedule of the planned features to be delivered during the next PI by each team.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA011

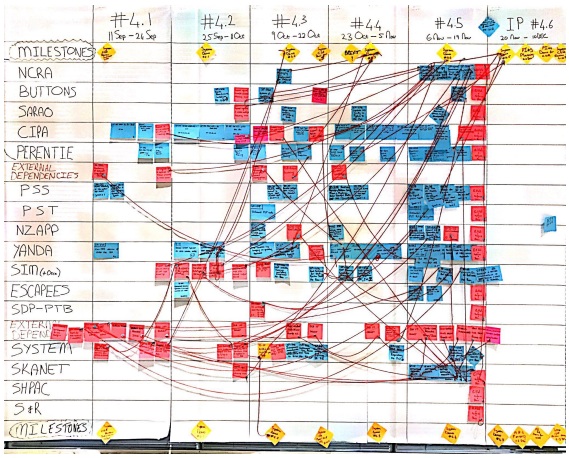

Figure 3: The program board resulting from PI 4 planning. Columns denote Sprints, and rows indicate Teams. Blue sticky notes represent the planned delivery date for a feature, pink sticky notes represent dependencies between teams, and orange sticky notes represent milestones.

The first PI planning event was held at SKA head quarters, in Jodrell Bank UK, during the first week of December 2018. Setting a fixed date well in advance created a forcing function with many positive effects. The date was used to generate a plan for the ART launch; travel plans could be made in advance, thus reducing the overall costs; and it generated expectations and enthusiasm about the upcoming event. It was decided to arrange the PI planning agenda over three days in order to accommodate remote participation from different time zones and maximize the overlap time between participants. The remaining two days of the working week have been dedicated to team meetings and training. Some teams were meeting for the first time, and not all of them were familiar with SAFe and the PI planning process, so some training was essential to achieve an effective PI planning session. The planning was set up so to have plenary sessions in a common room, and team breakouts in dedicated rooms equipped with videoconferencing facilities for each team. This setup proved to be effective, and it led to a positive participation and involvement of all participants.

From PI1 to PI4

At the time of this writing, the SKA-wide collaboration has executed four PI planning events, and it is currently executing the program increment number four (PI4). In this period of time, the number of teams involved in the SAFe program has grown from 5 to 17, and the project is evaluating the possibility to split the effort into two different ARTs to better manage the large number of people involved.

All the PI planning events were executed successfully with positive outcomes. The collaboration has grown in size and maturity during the series of program increments, and some aspects of this journey are worth noting.

Planning is a difficult activity in the software world, and it is a very approximate process, in particular for newly formed teams. Time-boxing the planning to three months helps to achieve better predictability, but all teams struggle in their early program increments. Teams improve their planning and predictability in successive PIs.

Big-room planning has many positive effects, as all teams have noted how everyone now feels more involved in the project. Transparency is greatly enhanced as well, as all team's plans are shared, and dependencies are better understood and managed. It is also important that people from different teams have a chance to meet each other, something particularly true in a collaboration such as the SKA, which involves teams across all five continents.

Teams are encouraged to take ownership of their own planning and decisions. Decisions are effectively delegated to the most appropriate level in a transparent process that *allows feedback* and *celebrates failures* as learning milestones.

It is easy to spot chances for early integration of software products developed by different teams. System-level integration is now encouraged as one of the most valuable activities, and not something to be relegated to the final stages of development.

Progress is demonstrated by teams during system demos. This reporting technique is very effective in capturing stakeholders feedback during the early stages of development. The frequency of system demos has intensified over the different PIs, and the collaboration is now targeting one system demo for each development sprint.

Teams developing hybrid systems composed of hardware and firmware have joined the ART. The overall structure demonstrates to be very well suited to support this kind of work, also enhancing the chances for early verification of these systems in association with the software stack.

The adoption of a well-known, industry-supported framework proved to be essential in the setup of PI planning events external to the SKA HQ. This was obvious during PI planning event number 3 held in Pune, India (see Fig. 4). The PI planning event was hosted by a private company called Persistent Systems, and the mutual understanding of the framework with associated logistics and "ceremonies" helped in setting up everything correctly with little overhead.

Figure 4: PI planning number 3, held in Pune, India.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA011

It is relevant that SAFe also offers very well structured training packages, and as new teams were onboarded, it was easy to train everyone to gain an understanding of the SAFe processes, and most importantly to adopt the underlying values and principles. Having the ability to host these training events internally thanks to the SPC certifications made them extremely cost effective.

CONTINUOUS IMPROVEMENT

The SAFe framework explicitly calls for time to be devoted to analysis and introspection in relation to the processes and their effectiveness. This happens for the teams in each sprint during the *sprint retrospective*. The ART has a dedicated *Inspect and Adapt workshop* during every IP sprint, where everyone in the collaboration can highlight aspects that are subject to improvement and participate in the definition of a possible solution. This mechanism is essential, and it has clearly produced improvements for the ART. Some improvements are related to the tooling and to the supporting infrastructure, while others are introduced additional concepts. In particular this innovation process led to the creation of a number of **Communities of Practice** (CoPs), responding to the emerging need from the teams to improve and harmonize some technical aspects of the software development activity. Among them the Testing CoP, aimed to standardise the testing practices among the teams, by providing a forum to discuss testing related topics.

CONCLUSION

Changing a collaboration model on the basis of Lean-Agile principles is a major endeavour that should not be approached lightly. Starting from a common understanding of the core values and principles is essential in driving the transformation process; those should remain the focus of the attention when transforming the organisational structure, subject to compromises and trade-offs. The SKA Organisation faced many challenges in adopting this approach, both on cultural and technical aspects, and we expect that every organisation on a similar path will face similar challenges, even if they are in particular dependent on the specific context and stakeholders. Overcoming some obstacles was only possible thanks to strong management support. Adopting SAFe proved to be very effective in improving the approach to software design and development, effectively moving the focus of all collaborators towards a system perspective, and towards a sense of shared ownership of the project. The technical practices associated with iterative software development and continuous integration are fundamental in this transformation and striving to reach high performance on these aspects is essential for a modern software development organisation [11]. This results in improved software quality processes and a higher degree of confidence in the software development activity. The trade-offs imposed by a collaboration such as SKA, where many developers are not fully dedicated to the project and teams are distributed around the globe, are particularly challenging when adopting a framework such as SAFe, that has a strong emphasis on individuals and in-person interactions. A strong emphasis on IT infrastructure and Audio Video systems for videoconferencing is necessary to enable the process in such a distributed setup.

ACKNOWLEDGEMENTS

The SKA Organisation would like to thank Ian Spence, Chief Scientist and SAFe Fellow of Ivar Jacobson International. Ian played a fundamental role in supporting the SAFe transformation and in driving the cultural change associated with it.

REFERENCES

[1] J. Santander-Vela, L. Pivetta, N. Rees, "Status of the Square Kilometre Array", ICALEPCS 2017, Barcelona, Spain, paper FRAPL01

[2] *"How do Committees Invent?"*, Conway, Melvin E. (April 1968), Datamation, 14 (5): 28–31, retrieved 2015-04-10.

[3] Manifesto for agile software development, https://agilemanifesto.org/

[4] *Extreme Programming Explained: Embrace Change*, Beck, K. (1999), Addison-Wesley, ISBN 978-0-321-27865-4.

[5] *Continuous Delivery*, J. Humble, D. Farley, Addison Wesley, 2011.

[6] *The DevOps Handbook*, G. Kim, J. Humble, P. Debois, J. Willis, IT Revolution Press, 2016.

[7] Scaled Agile Framework website, https://www.scaledagileframework.com/

[8] essential SAFe, https://www.scaledagileframework.com/essential-safe/

[9] Scrum definition, https://www.scrumguides.org/scrum-guide.html#definition

[10] PI planning, https://www.scaledagileframework.com/pi-planning/

[11] *Accellerate, State of DevOps 2019*, N. Forsgren, D. Smith, J. Humble, J. Frazelle, 2019.