SKHMS: SAFe Adoption for Chip Development

By

Johnny Lam
Director, SKHMS
&
Dr. Sanjeev Raman
Enterprise Agile|Lean Coach, cPrime

INTRODUCTION

SK Hynix Memory Solutions (SKHMS) is a leading provider of custom system-on-chip (SOC) solutions for the solid state disk (SSD) storage market and delivers SSDs to the enterprise storage market. SKHMS storage solutions enable cloud storage, high speed online transactions and big data analytics. They are a subsidiary of one of only five companies globally developing and manufacturing high performance NAND solutions targeting enterprise storage, portable computing devices, PCs smartphones and tablets. The company was the fourth largest semiconductor company in the world in 2014.

SKHMS has been providing custom solutions for the Enterprise SSD space since 2009, with the highest-levels of system-on-chip performance coupled with the most robust firmware. Their solutions demonstrate sustained high input/output operations with ultra-low latencies, even under mixed workloads.

PROBLEM STATEMENT & ASSESSMENT

Being a leading provider of custom system-on-chip (SOC) solutions for the solid state disk (SSD) storage market, SKHMS wanted to maintain their competitive edge via relentless improvement for producing enterprise grade SSDs. Relentless improvement is part of the House of Lean, which promotes a stable process or system that is reliable, predictable, and quality driven.

Therefore, the problem statement was: *To improve program optimization, alignment, and transparency; service delivery predictability and quality, and collaboration among engineers and leaders.*

The fact that there were internal Lean-Agile Leaders in the organization made it easier to initiate a transformation pilot in the company. When the firmware team first started, there was no talk about implementing the Scaled Agile Framework. SKHMS leadership wanted to be pragmatic with the approach.

SKHMS leadership and cPrime kicked off the assessment by conducting a series of interviews with firmware & test engineers, managers, hardware group, executives, etc., to understand the areas of improvements. One of the techniques applied was the Theory of Constraints (TOC) to further understand and agree to the major impediments in the Product Delivery Life Cycle (PDLC). As the TOC idiom poetically states, “a chain is no stronger than its weakest link.”

The SKHMS leadership also gave careful attention to:

- How hardware was coordinated with firmware development.
- How testing was conducted throughout the current PDLC process.
- Departments involved in building and delivering the product.
- How often these products were released to the customer and/or to the market.
- Source code management and build deployment.
- Tooling in place to support the Agile pilot.
- The U-Curve optimization (analysis of transaction costs) for delivering work.

VALUE STREAMS

SKHMS leadership analyzed how to optimize the Value Stream [flow of information and material from concept to customer] by empowering people to self-organize around the work. This strategy moved the organization away from cost centers and functional groups, to one allocated budget per train to help deliver the Value Stream. For example, think of a new expansion NFL Football team. When putting together a new Football team with the goal to win the Superbowl, players, coaches, and domain experts are carefully selected. The goal is to put together the right mix of people based on availability, budget, skillset, team affinity, etc. Based on this strategy, they act as a cohesive unit instead of separate entities that are forced together.

In the case with SK Hynix Memory Solutions, the value stream was to deliver best of breed, enterprise class solid state drives with industry standard interfaces. As part of this value stream, SKHMS leadership also dedicated a team of experts to test the various aspects of the firmware; from emulations to hardware in the lab, to qualification.

SAFE PILOT

SKHMS leadership, with cPrime's recommendation, chose SAFe as the Agile framework because it addressed the complexity issues often associated with firmware development. For example, planning required and delivered for feature development. Therefore, the Agile Pilot kicked off with 5 Scrum teams and roughly 50 people to support the Agile Release Train. The product selected for the pilot was based on risk level, preparation work in place such as architecture, and enough features to span multiple program increments. SKHMS leadership did start with Scrum teams being component based versus cross-functional. For example, component teams for:

- Encrypted Storage
- Host
- FTL (Flash Transition Layer)
- NAND Device Driver
- Platform

Cross-functional teams were discussed, but careful consideration and analysis indicated more in-efficiencies and coordination than being productive. However, the goal is to learn from the current Scrum teams to what is the most optimal for product delivery. For the pilot, the Scrum teams were distributed across San Jose, CA, and Longmont, CO, but most of the Scrum team were collocated in one geographical location.

SKHMS leadership designed the Program Increments to be 3 months long (once a quarter) and followed a two week iteration cycle. Before starting the first PI Planning, there was a list of preparation work that needed to get done. This work was folded into Iteration 0. From cPrime's experience, Iteration 0 needed roughly 1 week for a 3 month initiative. Since this was a 1 year pilot, SKHMS leadership agreed to one month to get prepared for launching the first Agile Release Train (ART).

SKHMS leadership also decoupled the Hardware group from the Firmware Agile Release Train because their work was not conducive to two week iterations with the Scrum Teams. Instead the Hardware group worked in a Kanban like fashion with SLAs on their work based on the Backlog prioritization. For example, knowing what features were coming down the pipe, they were able to prioritize their own work and in some cases, put out proto-hardware for testing purposes during the Program Increment. This coordination was possible because representatives from the Hardware group attended critical program level meetings as stakeholders and because they were part of the value stream for delivering the product.

CHECKLIST FOR GETTING PREPARED

1. Train Leadership on Lean Thinking, Agile, and SAFe (Leading SAFe).
2. Train Scrum Teams on Agile development & collaboration (ScrumXP).
3. Train RTE/Scrum Masters.
4. Train Product Management/Product Owners.
5. Document Definition of Done at the Team, Integration, and Release levels.
6. Create Initial Program backlog of Epics and Features.
7. Create Initial Team backlog of Stories for the first sprint or two for each Scrum team.
8. Create Initial Architectural Runway for Sprint 1 User and Technical Stories.
9. Sprint Design (when teams were going to meet for their Sprint Ceremonies).
10. Organize teams to make them as collocated as possible and reduced team interdependencies.
11. JIRA Agile setup and training for all people participating in the pilot. The Jira Agile setup included dashboards, scrum boards, and backlogs.
12. Agree to and setup CI infrastructure and process.
13. Agree to and setup metrics to track the teams and PI progress.
14. Agree to and setup KPIs to measure the Pilot success.

PROGRAM BACKLOG PRIORITIZATION

The Product Management in conjunction with the Firmware Architect came up with the initial list of Program Epics and Features for the Program Backlog. These Epic/Features were prioritized using Lead Time and prioritization. To be more specific we looked at the Cost of Delay and job duration associated with each work item in the backlog. In particular, looked at the attributes on the next page for Cost of Delay and assigned points accordingly using the modified Fibonacci scale.

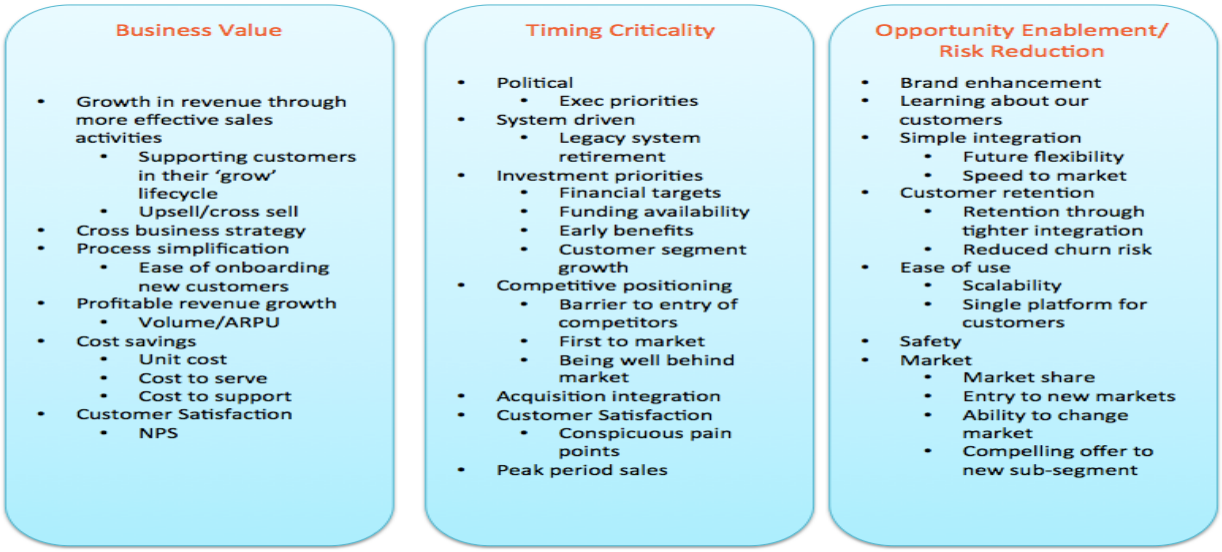


Figure 1: *Cost of Delay Definition*

FEATURE ANALYSIS & ARCHITECTURE DESIGN

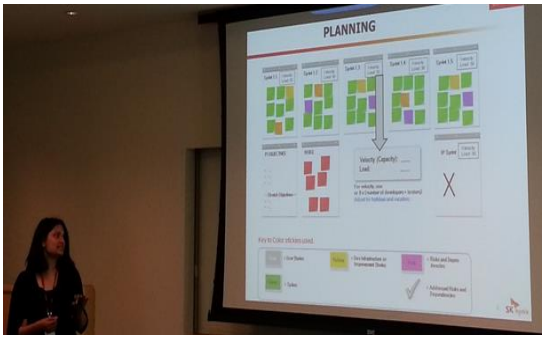
Since features and architecture design required in-depth analysis, it was very difficult to create user and technical stories. At the same time, the design analysis required lead time given the complexity and resource gap. With this challenge, the Scrum teams decided to create Spikes to represent architecture and high level design work in the team's backlogs. Based on upcoming feature prioritization and the intentional architecture strategy, the Product Owners were able to collaborate with their dev teams to create Spikes. Spikes were prioritized based on when the features had to be implemented. For example, if implementation of feature X was in Sprint 5, then the Scrum teams planned for the Spikes before Sprint 5. To further improve quality and limit costly redesigns, these spikes had a review component to it where key stakeholders of the dev team and external reviewers had to provide the necessary feedback. The feedback from the reviews were discussed during backlog grooming, which was scheduled on a bi-weekly cadence as part of the Sprint design. In some cases, the architecture runway had to be implemented before feature development. In this scenario, technical stories were implemented to create the runway before the user stories for feature development started.

PROGRAM INCREMENT (PI) PLANNING

Executive management was committed to the execution, and enabled everyone to fly to San Jose, California for the 2 day PI planning. This helped ensure that each team achieved an approved draft plan for the next 3 months' worth of work with measurable, aligned objectives.

Mr. Bumsoo Kim, SKHMS SVP of Firmware Engineering, kicked off the PI planning session by setting the business context. The context helped to align teams to a common vision for the why we were working on this product and what it meant to the organization. The context also covered who the clients and customers were, where the business was heading, economic conditions impacting the industry, and internal changes to drive competitiveness in the SSD market.

The Release Train Engineer (RTE), Shubha Kumbadakone, explained how to create the team draft plan. She further explained how the final draft plan will be transcribed to JIRA Agile to facilitate the Sprints and be the source of truth for all the work committed; and the source of agile metrics. The reason we used JIRA Agile was because it was very scalable from a cost and geography perspective, since we had teams that were remote and future Scrum Teams beyond the Pilot. Additionally, the tool was already installed and adopted corporate wide.



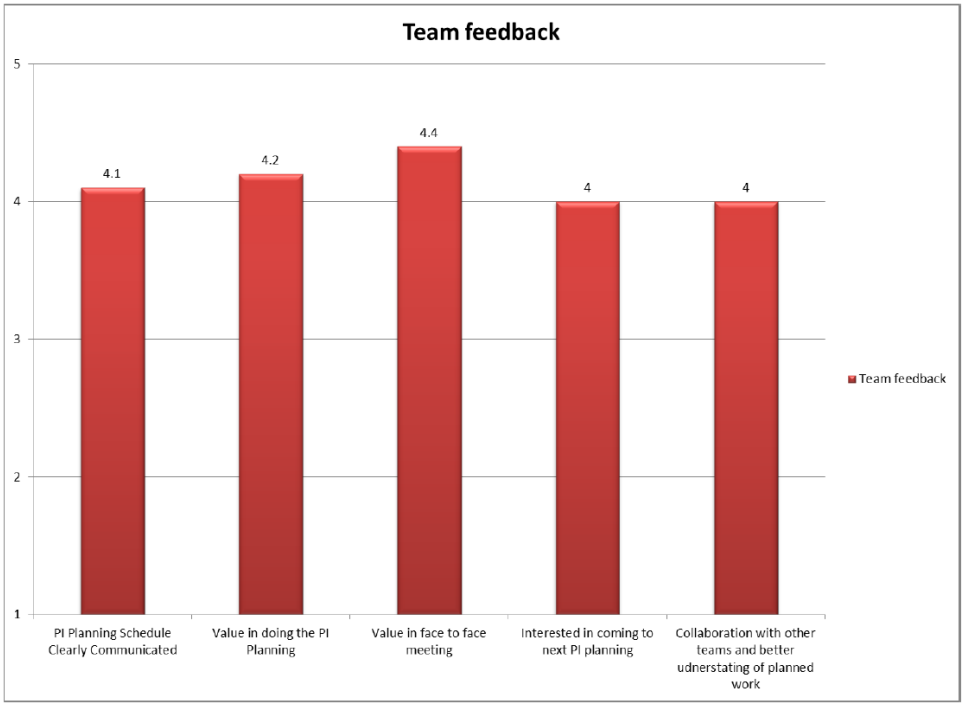
By lunch time on day 2, each team had a final draft plan to present that reflected intra and inter FW team and outside dependencies, risks, business value, and program increment objectives.

CONTINOUS INTEGRATION

Perforce is a source code management and content collaboration software. For CI Builds, Bamboo with Perforce was chosen to meet the many operational challenges inherent in component-based development, provide enterprise-class version management for distributed teams of chip designers, engineers, developers, and testers using a wide range of design and development tools.

CONCLUSION

By taking a pragmatic approach to initiating an organization change to help improve the product delivery processes, being mindful of people's roles and opinions, SKHMS leadership was able to kick off an Agile Transformation that people felt good about. Below is a survey from our first PI Planning from all the teams collectively.



To measure the Program Increment and the Agile Pilot success, the following qualitative and quantitative factors were evaluated:

I. Program Level

a. Feature completion.
b. Program Increment Objectives completion.
c. Stretch Program Increment Objectives completion.
d. Number of defects outstanding.
e. Time spent fixing continuous integration server and source code management.
f. Leadership and senior stakeholder feedback.

II. Team Level

a. Percentage of Stories accepted based on Sprint commitment.
b. Percentage of points accepted based on Sprint commitment.
c. Average cycle time of Stories (helps identify bottlenecks in the development process).
d. Velocity – consistent or improving over time.
e. Sprint Scope change.
f. Team Structure – is the team size optimal, dedicated, and collocated as much as possible.
g. Scrum Master feedback.
h. Team Self-Assessment and feedback.

Based on the above criteria, the overall consensus was the Agile Pilot was off to a solid start and teams were embracing the change, and seeing the tangible value of using the Scaled Agile Framework. The overall metrics and feedback also indicated 60% improved transparency, 55% defect reduction rate, and 50% improved service delivery predictability.

REFERENCES

1. Leffingwell, Dean, et al., *Scaled Agile Framework 3.0*, Scaled Agile, Inc. Accessed 30 June 2015. www.scaledagileframework.com