Scaling Product Ownership Through Team Alignment and Optimization

A Story of Epic Proportions

Peter Saddington
Co-Founder and Organizational Coach
Action & Influence, Inc.
Atlanta, Georgia
peter@myai.org

Abstract— **Scaling Product Ownership isn't something that happens overnight. It is an intentional and thoughtful process. Many organizational considerations must be made and teams must be optimized and assembled correctly. The following is an experience report on how the Department of Defense and the United States Air Force scaled product ownership on a multi-million dollar program spanning multiple teams and multiple stakeholders.**

I. THE PRODUCT OWNER ROLE

The Product Owner role is an intensely important role. This individual is what I consider to be a "Value Driver" to the system. Meaning, the Product Owners are the ones who drive value into the products and projects that they are working on. The Product Owners ensure that the right product is going to be built by the team and has the right conversations around what the team needs to specifically build. They also create a prioritized backlog, or ranked list of items or requirements needing to be built by order of importance. A primary component of the Product Owner's job is to ensure this list is constantly groomed, or updated as things change.

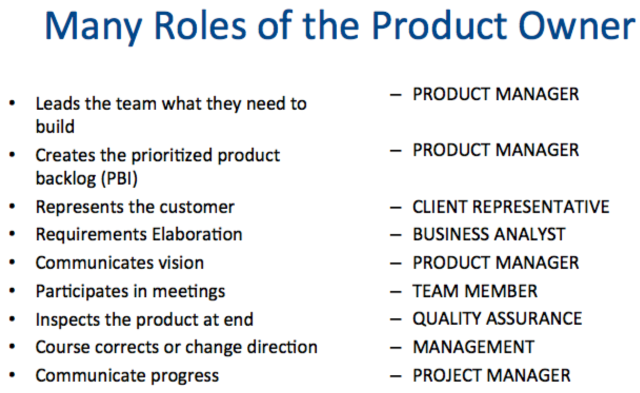
A successful Product Owner understands how they uniquely represent their end-customer, whether it be internal or external. The Product Owner should intimately understand the needs of their customers they are representing and be able to create the right amount of requirements, prepped and ready for conversations with the team to elaborate or expand on them.

Since the Product Owner is such a crucial part of driving value to the team, it is necessary for them to be actively engaged within the development teams progress. Therefore, it is important that they participate in the right amount of meetings while being available to the team at much-needed touch points to help give direction, course-correction, or more information that the team can use to help them build the right product. The consistent engagement from the Product Owner also allows him to communicate the progress of the project to the right stakeholders and interested parties in the success of the team. In a lot of ways, the Product Owner is not only an internal champion for the success of the team, but also a representative leader to the rest of the organization.

It is imperative that the Product Owner inspects the product at the end, and gives their stamp of approval on the completion of the right requirements. This most critical part of the process is where the Product Owner has the responsibility of making sure the iterative product is not only being built in the right fashion, but the trajectory of the team is moving in the right direction.

The Product Owner is not an individual in an ivory tower. They are distinctly part of the team, and as such, have a highly collaborative role to play within the team. The duties of a Product Owner are to be executed with a servant-leader attitude. Some adjectives that come to mind include: Engaged, available, informed, empowered, prepared, communicative, collaborative, flexible, adaptable, and humble.

In a project that is utilizing Agile, the Product Owner becomes one of the most critical and pivotal roles for a successful project. The Product Owner now takes on what were once duties held by traditionally different roles. While there may still be separation of roles in larger enterprises, an effective Product Owner may sometimes take on a variety of responsibilities in Figure 1 below.



Many Roles of the Product Owner

• Leads the team what they need to build	– PRODUCT MANAGER
• Creates the prioritized product backlog (PBI)	– PRODUCT MANAGER
• Represents the customer	– CLIENT REPRESENTATIVE
• Requirements Elaboration	– BUSINESS ANALYST
• Communicates vision	– PRODUCT MANAGER
• Participates in meetings	– TEAM MEMBER
• Inspects the product at end	– QUALITY ASSURANCE
• Course corrects or change direction	– MANAGEMENT
• Communicate progress	– PROJECT MANAGER

Figure 1. Various Roles of the Product Owner

II. A SIMPLE PRODUCT VS. COMPLEX PRODUCT

Before we dig into the complexities of an enterprise project, it is valuable to look at a simplified version of a standard project with minimal constraints and dependencies.

Figure 2 shows an example of a Simple Project. For a small team, within a small organization, Product Ownership is relatively straightforward as it is simple to employ. A single Product Owner within a single team helps guide and direct a single application from inception to delivery. The Product Owner interacts with the team via the backlog (total work to be done), the product reviews at the end of a sprint, as well as through grooming sessions in which the Product Owner helps provide guidance for requirements details.

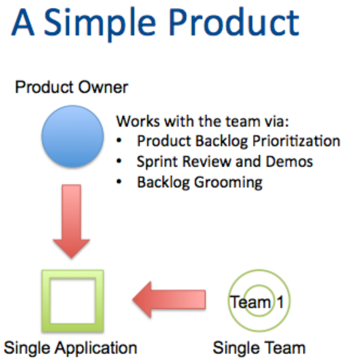


Figure 2. An example of a Simple Product

Often this is not the case. In large enterprises, one can typically find an entire program with multiple teams, organizational dependencies, team-specific constraints, multiple stakeholders, cost differentials, and more. This was no different at with the Department of Defense and the United States Air Force Program.

The desire was to unify 4 different teams with a common goal of releasing a new product that would help in the deployment of troops. This particular product had to be released quickly, with quality, while focusing on the highest-priorities of the stakeholders in mind. It was no easy task, and this was only the beginning of an entire initiative to introduce Agile across multiple programs. With millions of dollars at stake and the safety of our men and women uniform in mind, we set out to undertake and create a story of epic proportions.

III. SCALING PRODUCT OWNERSHIP CHECK LIST – VISION

The first part of beginning to scale multiple Product Owners with multiple teams was to ensure that every major stakeholder and Program Manager was aligned with each other towards a common vision for the final Product. It was absolutely crucial to nail down this first part so that everyone in the program understood where we were headed. All of the stakeholders and Program Managers were then dubbed: Product Owners.

This unified vision was to be our stake in the sand driving us towards excellence together. The execution of eliciting the vision included bringing all stakeholders and future Product Owners together to discuss not only what the vision was, but also ensure that we all had a common understanding of the purpose of the completed product was.

Agenda for Vision

- Establish the unifying need or opportunity (in the form of one sentence) for this product.
- Establish the product name
- Establish the key benefit of the product

With multiple high-ranking officers and stakeholders it was intensely frustrating for them to be part of a complex program where positioning, politicking, and pretense were the standard for the day. It was important for us to move beyond that, and unify all stakeholders on the core of what this product was all about.

IV. REQUIREMENTS OVERLOAD

After we had established a vision for the product, every single individual and stakeholder wanted to move towards the requirements of the product. Each stakeholder had come prepared with their own list of requirements that they 'needed' to include into this system. While being very empathetic towards the work that they had put into their requirements, we went through an exercise of understanding the requirements process.

Requirements Overload - ASK / REWARD / PENALIZE / BUILD Cycle:

- ASK - Often during the requirements process, stakeholders ask for everything under the sun to be thought of, considered, and included into the requirements document. Much time is spent in the building of this requirements document so that nothing is missed.
- REWARD - After all requirements and considerations are made over an often intensely long period of time, a large and robust requirements document was built. We then reward the teams with considering 'everything' and creating an awesome requirements document.
- PENALIZE - Through the process of executing on the requirements set by the requirements document, we often find that things are missing. We then go through a period of disillusionment and penalize the teams for not thinking of everything.
- BUILD - As we stick to the requirements set forth and the growing list of requirements 'not thought of' during the requirements build phase, we end up building features that are not imperative to the success of the product, and often over-build without focusing on the top priorities.

This was indeed a tough pill to swallow, but all agreed that it would be a far better use of time to come together and focus only on the top-priority goals and features needed to release a viable product on time and within budget.

V. SCALING PRODUCT OWNERSHIP CHECK LIST – BUSINESS GOALS

Through facilitation we began a strategic meeting focusing on the primary business goals and "mission critical" priorities of the product.

I began the meeting by beginning with the question "If you could only have one thing…" - Meaning that if there was only one thing that we could complete on this product, what would it be? This gave us a distinct objective that was our utmost priority. From there we began breaking down the #1 priority epic (large idea) down into features. This essentially became our first business goal for the product.

After decomposing the largest epic down into reasonable features, we discussed and chose 2 other business goals that all stakeholders could agree on as the primary viable product set for the system.

Decomposition of Epics to Stories included:

- Epics - Epics are the theme or goal, often broken out into multiple features, typically from 1-3 months in duration. These epics can span more than one team and are inclusive of all priorities that the teams decide are viable for product launch
- Features - Features were broken out from the 3 major epics, typically 2-4 weeks in duration. Ideally, features are contained within a team, and each particular Product Owner is laser focused on his or her teams feature assignments.
- User Stories - These are the smallest increment of value, typically less than a week. The team and responsible engineers broke the user stories down further before working on them.

VI. FULL UTILIZATION?

A unique opportunity arose within this program to change the way the business handled team assignments of work and capacity. Prior to our inception of an agile delivery method, there was one large team made up of 80+ members handling multiple responsibilities and applications as seen in Figure 3.

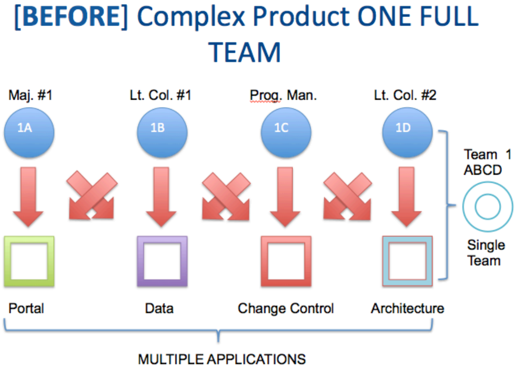


Figure 3. Multiple applications spread over one team

What we found, previously, was that managing the enterprise backlog was an intense and excruciating exercise in human resource control while managing dependencies and constraints between teams blocking progress. Often teams sat waiting while other teams completed crucial parts to integrate systems together. This was not only a waste of time, but did not allow learning to happen across teams as seen in Figure 4.

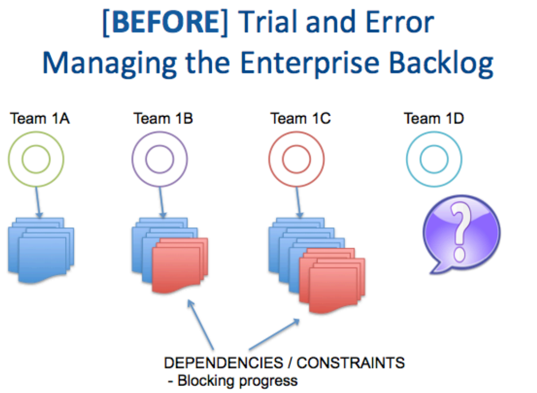


Figure 4. Trial and Error – Managing the Enterprise Backlog

Prior to engaging in an agile method to manage the program, what we found was the teams worked in tangent to each other on similar feature sets. When working capacity was low, the stakeholders injected more project work or 'work-ahead' tasks to prep for next-phase plans within the requirements document.

What was incurred was filling up each team's gaps of time with work that wasn't helping any of the teams deliver on primary objectives, and while each team was at full capacity, each team became bogged down with work that did not accelerate the backlog of items, constraints, or dependencies on the prior work needed to be complete as seen in Figure 5.

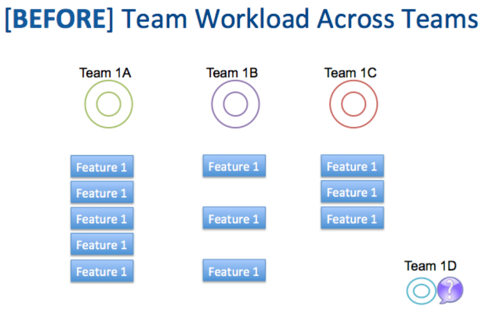
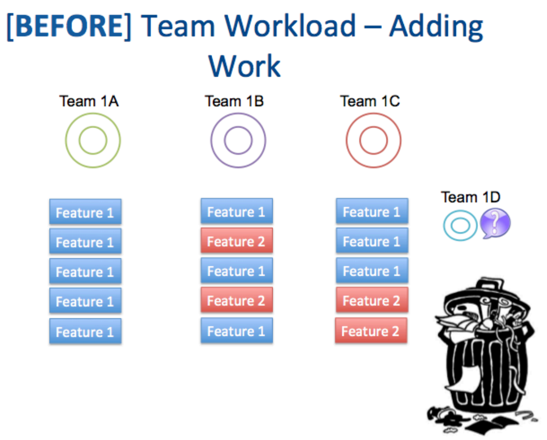


Figure 5. The Beginning Process of Filling Capacity

Because the teams have interdependencies, the teams have times they are blocked. See Team 1B for blocked time in the middle of Feature 1, and Team 1C at the end of Feature 1. Team 1D did not have any pre-defined work and played 'backup' for non-critical-path work.

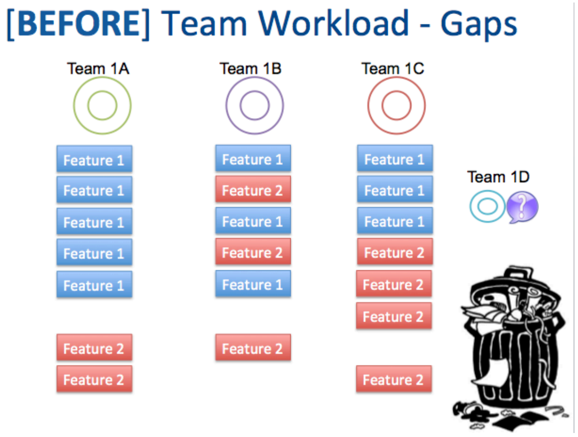


[BEFORE] Team Workload – Adding Work

Team 1A	Team 1B	Team 1C
Feature 1	Feature 1	Feature 1
Feature 1	Feature 2	Feature 1
Feature 1	Feature 1	Feature 1
Feature 1	Feature 2	Feature 2
Feature 1	Feature 1	Feature 2

Figure 6. Adding Extra Work to Fill Capacity

To remedy the blockages, Management added Feature 2 to the work queue for Teams 1B and 1C. Again, Team 1D spent time waiting for work to come to them.

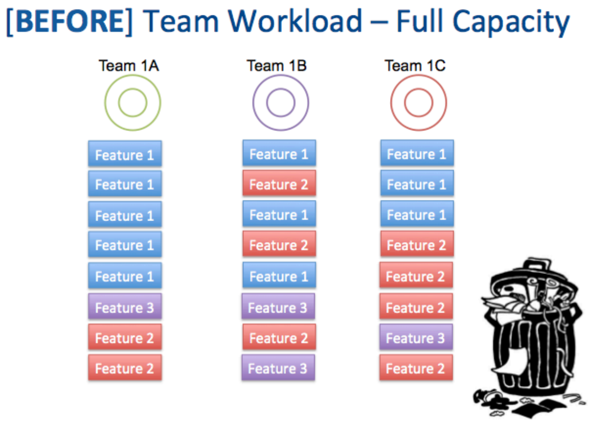


[BEFORE] Team Workload - Gaps

Team 1A	Team 1B	Team 1C
Feature 1	Feature 1	Feature 1
Feature 1	Feature 2	Feature 1
Feature 1	Feature 1	Feature 1
Feature 1	Feature 2	Feature 2
Feature 1	Feature 1	Feature 2
		Feature 2
Feature 2	Feature 2	
Feature 2		Feature 2

Figure 7. Adding More Work Increases Gaps

Adding Feature 2 seemed like a great idea. But it mostly just complicated the current work being done by the teams. As they switched between activities, lost productivity and confusion occurred.



[BEFORE] Team Workload – Full Capacity

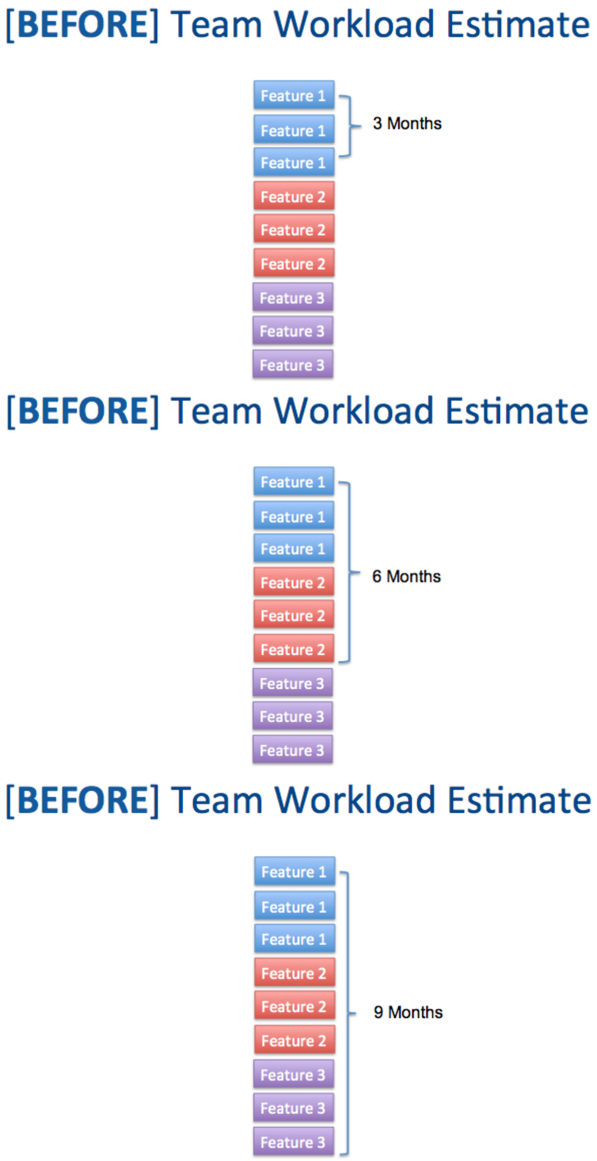
Team 1A	Team 1B	Team 1C
Feature 1	Feature 1	Feature 1
Feature 1	Feature 2	Feature 1
Feature 1	Feature 1	Feature 1
Feature 1	Feature 2	Feature 2
Feature 1	Feature 1	Feature 2
Feature 3	Feature 3	Feature 2
Feature 2	Feature 2	Feature 3
Feature 2	Feature 3	Feature 2

Figure 8. Full Capacity Reached – No Buffer for Errors

This was our team previously, at full capacity, with multiple dependencies and feature crossover. Our teams were “busy” but not focused nor as productive as they could be if they were allowed to focus!

Producing Value is More Important than Being Busy

The resulting net affect was that estimates were most often wrong and based on an artificial metric. If the previous features were averaging 3 months to complete, we therefore could also estimate likewise as Figure 9 depicts.



[BEFORE] Team Workload Estimate

Feature 1
Feature 1
Feature 1
Feature 2
Feature 2
Feature 2
Feature 3
Feature 3
Feature 3

[BEFORE] Team Workload Estimate

Feature 1
Feature 1
Feature 1
Feature 2
Feature 2
Feature 2
Feature 3
Feature 3
Feature 3

[BEFORE] Team Workload Estimate

Feature 1
Feature 1
Feature 1
Feature 2
Feature 2
Feature 2
Feature 3
Feature 3
Feature 3

Figure 9. Team Workload Estimates

The reality was that in filling capacity to the fullest was more likely to hinder overall progress to complete the total system as seen in Figure 10.

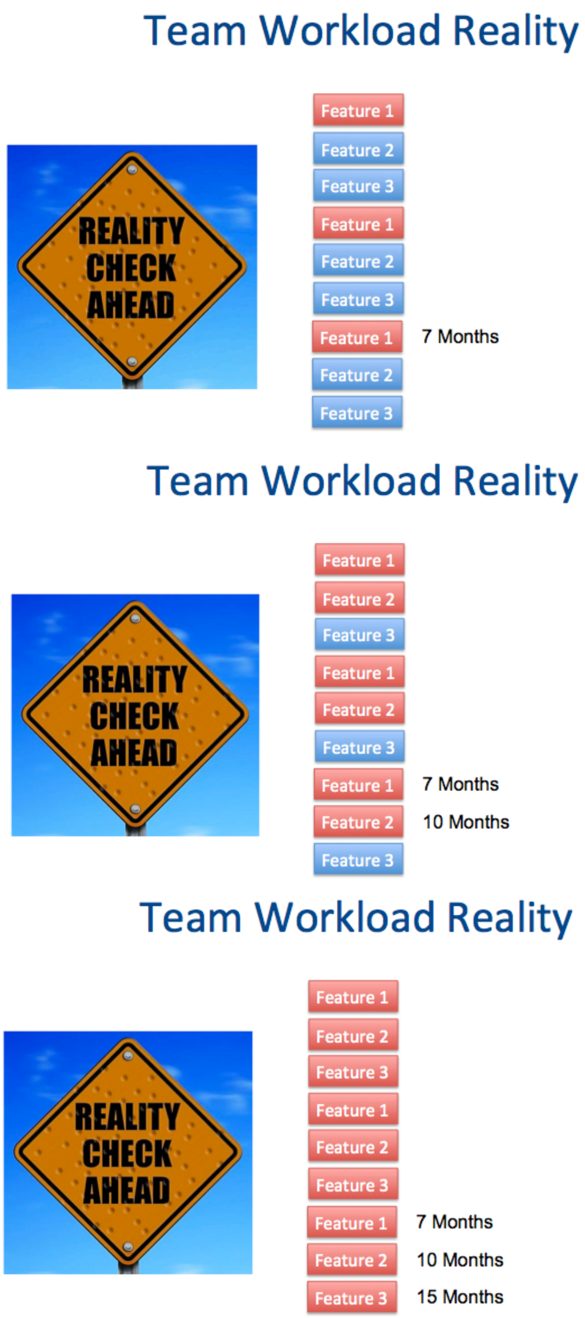


Figure 10. Team Workload Reality

Shifting the Sands

After 2 weeks of seeing the giant team in action and taking copious amounts of notes, we decided on a clear plan on how to better align each stakeholder/Product Owner with respective teams focusing on very specific deliverables. What was previously one big team with multiple inter-dependent team units, we suggested that each Product Owner take on the responsibility of a single team focusing on a particular segment of the total system working diligently on each part and build it with excellence and quality.

What we needed was that each Product Owner had a specific team that they could manage successfully while making sure that the right connections between teams were well understood and managed appropriately.

We utilized a team assessment and optimization tool called Action & Influence to better understand each team member's behaviors, skill-sets, and collaboration methods to create 4 different teams, lead by the 4 different stakeholders. This tool allowed us to uniquely craft each team according to the needed roles and responsibilities for each team as well as fit each team member into a role that they were uniquely fit for. No longer did we have to guest empirically as to which team member should be on each team. With Action & Influence, we were able to have a matrix of our entire program's human resources and allocate each team member to exactly the right role and fit for each team. This in turn created a unique team culture for each team.

Previously:

- Multiple Product Owners and Stakeholder within one large team
- Multiple priorities and loss of cohesiveness between team members who were attached to different stakeholders
- Poor visibility into total overall movement of project towards strategic goals
- Lack of stakeholder alignment as to integration points and quality checks

After Re-Alignment - Multiple Teams for a Single Product (Figure 11):

- Single Product Owners managing a single functional team
- A single unified priority in which all functional teams and Product Owners focused on
- More visibility (as depicted in Section VII)
- Better alignment of teams (as depicted in Section IX)
- Better communication and collaboration with each team through the use of Action & Influence

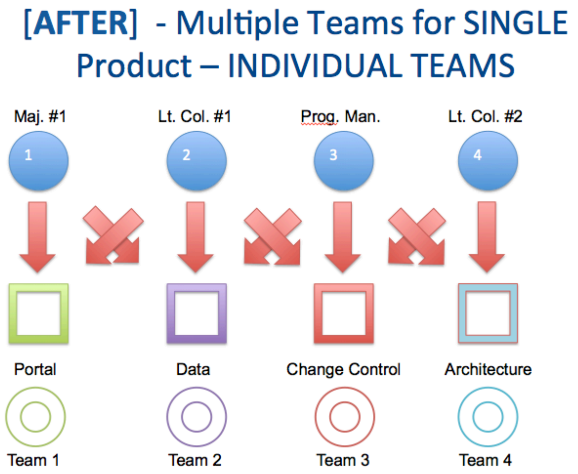


Figure 11. Multiple Teams Focused on Feature Development

Team optimization and understanding the distinct makeup for each team was absolutely essential to each team's functional roles.

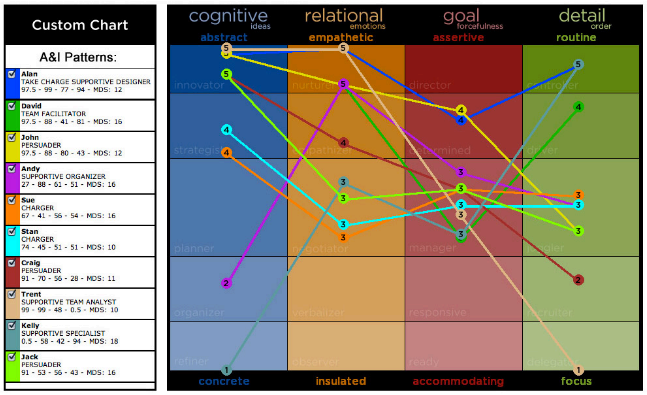


Figure 12. After Re-Alignment – Moving and shifting employees and contractors to the right teams with the right roles and responsibilities.

Utilizing Action & Influence (myai.org) we were able to create the best functional teams according to the human resources we had. This is an example of one of our teams, which we uniquely outfitted, with the right balance of developers, designers, quality engineers, and a business analyst [1].

VII. SCALING PRODUCT OWNERSHIP CHECK LIST – VISUALIZE PRIORITIES AND LIMIT TEAM WORK IN PROGRESS (WIP LIMIT)

Our next step was to enable each team to visually represent necessary work and integration points with other teams. Each team built a physical wallboard representing their commitment to each iterative build and where integration points were necessary with other teams. This was one of the biggest benefits for all teams in that within a span of 20 meters or so, one could fully see what each team was working on, the dependencies, constraints, and integration points, as well as how each teams workload and capacity was managed (See Figure 13 below).

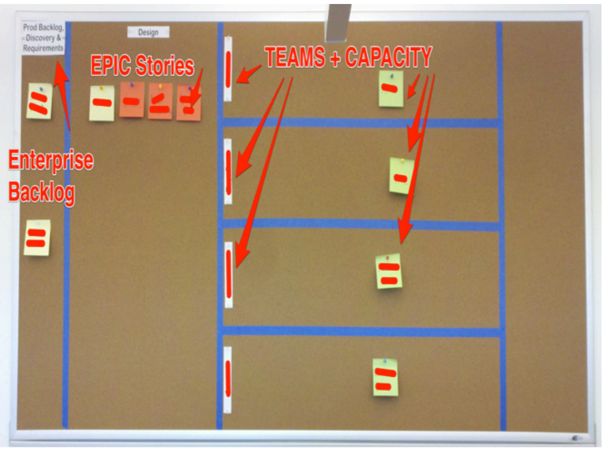


Figure 13. Wallboard Shows Team Capacity

Every day, each team's Product Owners would walk the wallboard and take notes on to-do's as well as questions they needed answering to help the teams be as effective as they could be.

VIII. SCALING PRODUCT OWNERSHIP CHECK LIST – DEFINITION OF DONE

With the visualization of work and team assignments, it was obvious that we needed to have a very clear understanding of a Definition of Done between the teams. We not only had to make sure that all the functionality was working, but technical documentation was done, service integration processes fulfilled, acceptance criteria met, UI conforms to approved templates, internal wiki was updated, security measures covered, coding standards were met, and more.

We completed a Definition of Done with all the teams through a Mind Mapping Definition of Done Exercise in which we brought all the team leads together to create a full system definition of done at 2 levels:

- Sprint Definition of Done – 'Done' requirements after each sprint.
- Release Definition of Done – 'Done' requirements after each release, to include: integration, quality, security, documentation, etc.

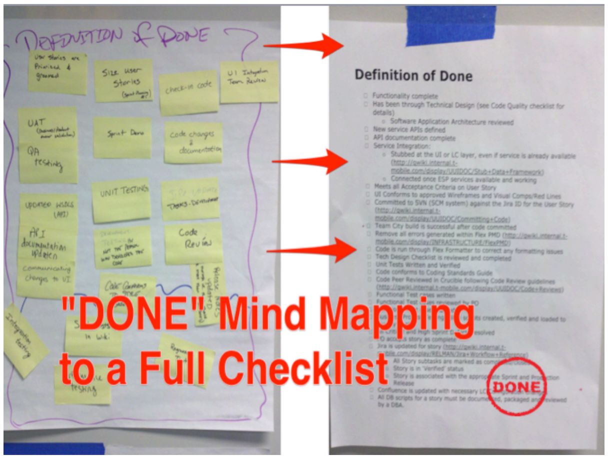


Figure 14. Our Mind Mapping to Official Definition of Done Template

The outcome of aligning all of the teams around single pieces of functionality, while creating a highly visible wallboard and artifacts, and establishing a common Definition of Done was amazing. What used to be an exercise in 'filling teams time when they had capacity' turned into a swarming effect where we spread the features across each team and worked together to complete each feature before we began work on other features. An extra infrastructure team filled in gaps where necessary as depicted in Figure 15.

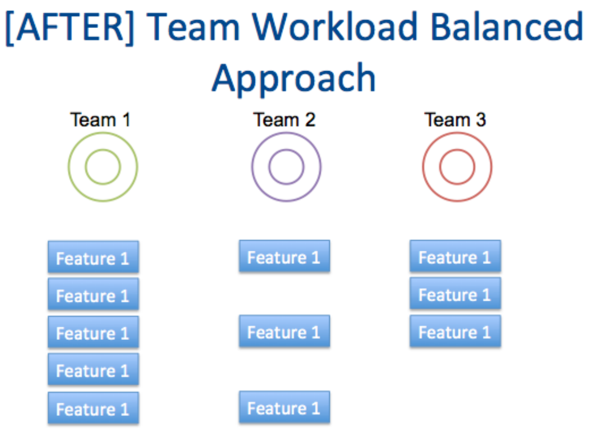
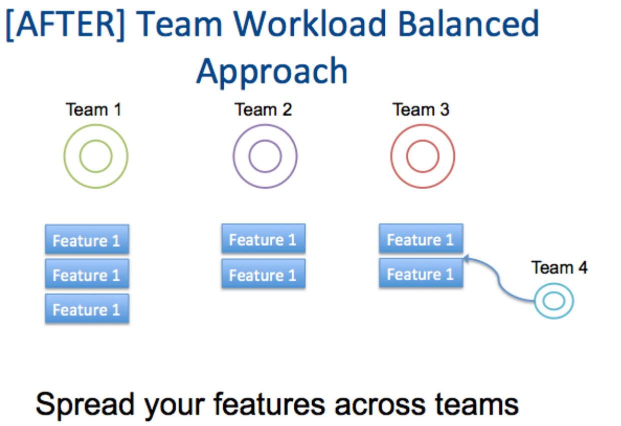

[AFTER] Team Workload Balanced Approach

Figure 15. Begin Team Workload Balanced Approach

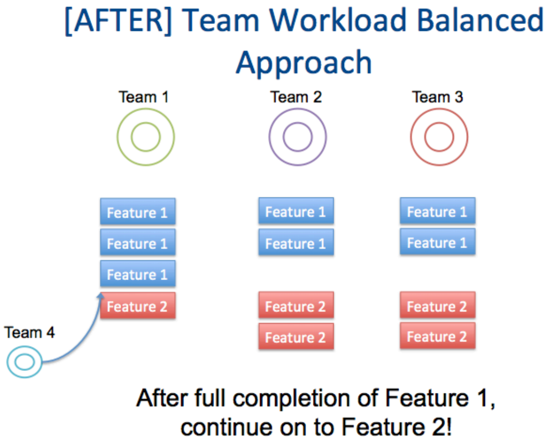
At the beginning of our new process we allowed teams to pick up their required work necessary to complete Feature1.


[AFTER] Team Workload Balanced Approach

Spread your features across teams

Figure 16. Teams Work Together to Complete Features in Unison

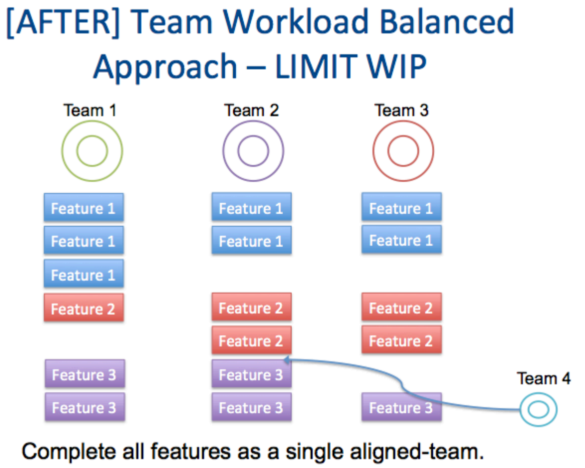
We then allowed the team to align their required work together and allowed Team 4 to help with any integration or quality assurance tasks.


[AFTER] Team Workload Balanced Approach

After full completion of Feature 1, continue on to Feature 2!

Figure 17. After Full Completion of a Feature, All Teams Align to Complete Next Feature Together

After all teams had completed Feature 1, we moved as a whole to Feature 2, while still allowing Team 4 to solidify integration, remove technical debt, and assist in any extra feature development.


[AFTER] Team Workload Balanced Approach – LIMIT WIP

Complete all features as a single aligned-team.

Figure 18. After Completing Feature 2, All Teams Work on Feature 3.

After completing Feature 2, the entire group moved on to Feature 3. Team 4 used their time to assist in any feature development.

Figure 19 (below) shows an example of the alignment of the 3 teams together using a common wallboard in a visible area. The pink cards show integration points for each team to be aware of.

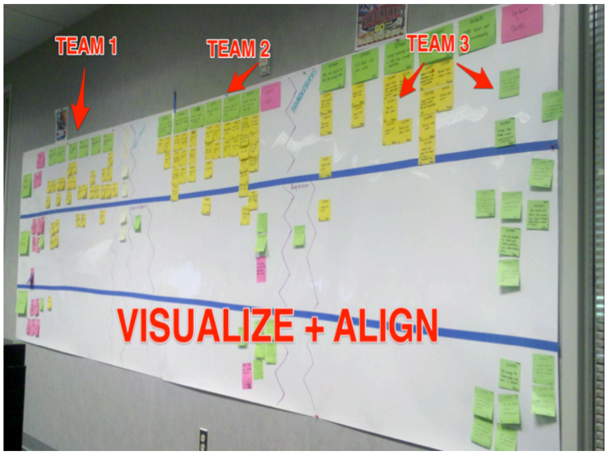


Figure 19. Our Big Visible Wallboard allowed all teams to see required work needed to complete together

IX. SCALING PRODUCT OWNERSHIP CHECK LIST – SCRUM OF SCRUMS

Lastly, we employed a Product Owner Scrum of Scrums in which weekly each Product Owner from each team would meet and discuss how to remove team impediments and negative dependencies for each sprint. This was effectively called a "Product Management Alignment Team."

Agenda for Product Management Alignment Team Scrum of Scrums:

- Review previous meeting notes and resolution to pending issues
- Team by team review of considerations
- Execution and communication plan for each tiered issue
- Close

X. SUMMARY – ALIGNMENT AND TEAM OPTIMIZATION MAKES ALL THE DIFFERENCE!

In summary, the key to our Program's success was alignment of vision, goals, teams, and workload. We ensured this to be possible through executing on the following disciplines:

- Big visible charts and team wallboards
- Team alignment daily/weekly
- Making policies explicit through working agreements and a common Definition of Done
- Product Owners need to align and know all constraints on teams
- Cultural change must happen
- Optimize, build, and re-order teams according to their skill-sets (we used the Action & Influence Solution)
- Increase communication and collaboration through team building

The final results of the program were outstanding! The entire program moved to 2-week sprints, with full Product Owner and stakeholder engagement for each team.

- 78% of total features were complete in the first 4 months
- 130% decrease in defects
- 90% of Mission Critical Features completed ahead of schedule (9 months)

Dealing with large enterprise projects can be a daunting task. A great facilitator can help quell the fears that come along with making sure every team, every moving part, and every invested stakeholder is aligned. A cultural or team audit is a great place to start, as it helps a consultant or coach better understand the team dynamics at play. No team is perfect from the start, but with time, patience, empathy, and understanding, a great coach can help move and disentangle the complexities and intricacies of a complex project or program.

Regardless, it is absolutely imperative that full program alignment happens, at the executive and team level. This is only possible through quick feedback loops, intentional points of collaboration, and balanced teams that have been reorganized or built to fit their unique functional role. In total, a daunting task it may be, but it can be a fun, rewarding, and exciting opportunity to help teams thrive, businesses be successful, and people feel fulfilled.

REFERENCES

[1] "The Action & Influence Solution," [Online] Available http://myai.org/services/assessment-tool/team-matrix/, and http://myai.org. April, 2011. Accessed January, 2012