Agile: Lessons Learned

GLOBAL PRODUCT DATA
INTEROPERABILITY
SUMMIT
2017

ELYSIUM   Parker Aerospace   NORTHROP GRUMMAN   BOEING
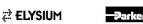
Progress and issues during our second year of developing for PLM Software in an Agile Scrum environment

Introduction

- **Sean Webster**
- **Northrop Grumman**
- **Software Development Analyst**
- **18 years in IT**
- **4 years working with PLM Software**
- **2nd year using Agile Scrum methodology**

- **Agile Scrum Methodology**

- **Continuation of "Getting Agile" (presented last year)**
  - **Project Goal**
  - **Differences between 2016 and 2017**
  - **Progress and Issues**
    - **Development Teams**
    - **Tools**
    - **Agile Scrum Process**
      - Backlog
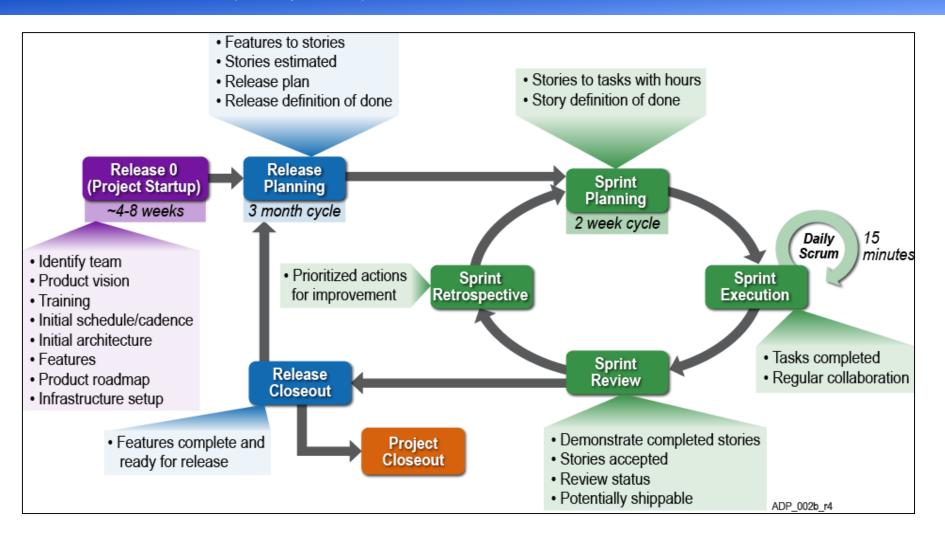   - Sprints
      - Management, Stakeholders and Vendors

Agile Scrum Methodology

Agile Scrum Methodology

Global Product Data Interoperability Summit | **2017**

Business / Management

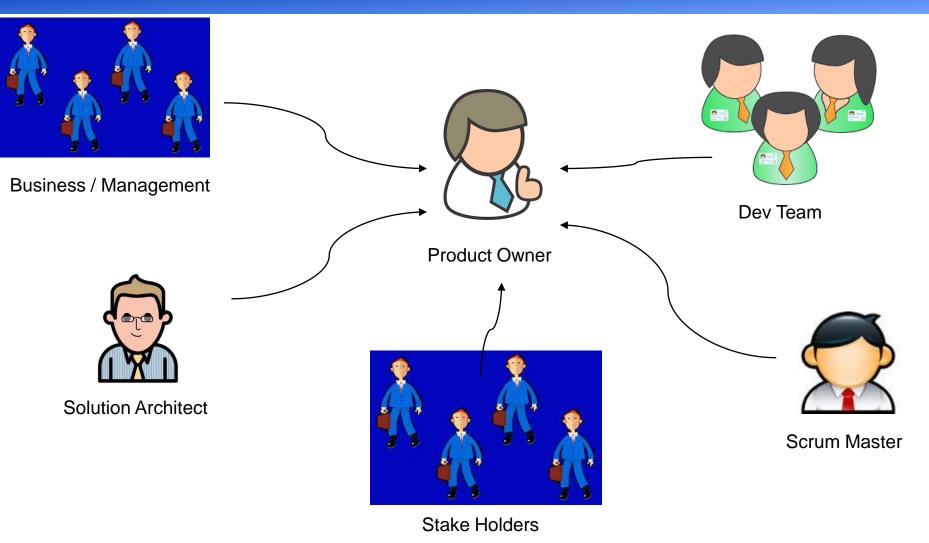Product Owner

Dev Team

Solution Architect

Stake Holders

Scrum Master

Project Goal

- **Support all existing functionality for Manufacturing and CAD integration in new system.**

- **Load Parts and Documents from legacy system and keep them in sync. The primary tool used for this function is a Migration and Integration solution tool.**

Differences between 2016 and 2017

	2016	2017
Development Teams	2 Teams Agile rookies	3 Teams - Added a team to handle requirements outside our expertise Increased Agile knowledge
Tools	Utilize software to: • Manage Agile process (Backlog,Sprints) • Git	Added the ability to auto build and deploy to multiple servers
Backlog	Little effort	Added 2 week long workshops for Backlog refinement Added weekly grooming sessions to our Sprints
Sprints	3 Week	2 Week 4 Week Teams split Sprint duration based on statement of work

- # Team members took an Agile 101 class
  - ## Are we properly following Agile process
  - ## Get insight from Scrum Masters helping other teams
  - ## Suggest development team do this together

- # Experience using Agile daily
  - ## Better at point estimation
  - ## Better at requirement definition
  - ## Added Definition of Done

- # New Team
  - ## Handles backlog items not in Development Team's expertise
  - ## Follows a "relaxed" Agile Scrum process

Development Teams - Issues

- **Recording progress in Agile software**
  - **Hours remaining**
  - **Hours to complete**

- **Testing**
  - **Test driven development**
  - **Including testers sooner in the Sprint cycle**

- **Definitive acceptance criteria**
  - **Stakeholder and developer have different interpretation of "done"**
  - **Additional Backlog effort required**

- **Automate Build and Deploy**
  - **Builds are triggered on commit**
  - **Notifications of results and options to deploy**
  - **Configured for multiple servers (independent configurations)**

➢ **More details provided by Jim Brooks in his presentation**

- **No major issues to report**

Backlog - Progress

- **Workshops for Backlog refinement**
  - **1 week, January and June**
  - **Everyone in attendance, in person**
  - **HUGE help for Development Team**

- **Weekly grooming sessions**
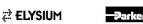 - **Part of Agile Scrum process we weren't doing**

- **Still do not spend enough time on Backlog**
  - **Agile recommends 4-8 weeks to work on backlog**
  - **Perform Backlog refinement**

- **Stakeholders**
  - **Make sure you have the correct users defining requirements**
  - **Delayed requirements**

- **Flexibility**
  - **Team A moved to a 2 week sprint**
    - **More experienced**
    - **Better backlog**
    - **Considered 1 week sprint, not yet**

  - **Team C moved to a 4 week sprint**
    - **Stories not able to be completed in 3 weeks**
    - **Hard to get tasks under 8 hours**
    - **2 "daily" scrums a week**

- ## **Definition of Done**
  - ### **Part of Agile Scrum process**
  - ### **Everyone knows when a story is complete**
  - ### **Can change with every Sprint**

Sprints - Issues

- **Stories complete but NOT**
  - **Complete quickly in next sprint (Ex.: testing remains)**

  - **Carry Story forward**
    - **Keep story points in initial sprint**

  - **Plan for Stories with outside dependencies**
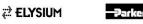   - **2 weeks can be a short turn around**
    - **Split the Story**

- **Culture - Waterfall World**
  - **Working with QA, Master Scheduling and Program**
    - **How to track progress**
    - **Quarterly release / 4 Sprints / …**

  - **Keeping points in initial sprint**
    - **Helps tracking progress on each Sprint**
    - **Carry over can provide incorrect velocity**

  - **How to handle additional stories?**
    - **Requirements change**

- ## Have the correct Stakeholders

- ## Stakeholder Buy In
  - ### Make sure they understand the Agile process

  - ### Demo partial product

Vendors – Progress/Issues

- **3rd Party Software**
  - **Software not working as expected**
  - **Bugs**
  - **Longer response times on issues**
  - **Key to work hand in hand or closely coupled with Vendor**

Overall

- **Development efforts have improved**
- **Velocity has leveled, showing efficiency and experience**
- **Agile Scrum process was well received by the Teams**

- **My $.02**
  - **Enjoy the process**
  - **Get quick wins or success**
  - **Get quick knowledge of issues**

Questions?