

Professional Development Day (PDD)

Agile Story Telling Requirements Refinement

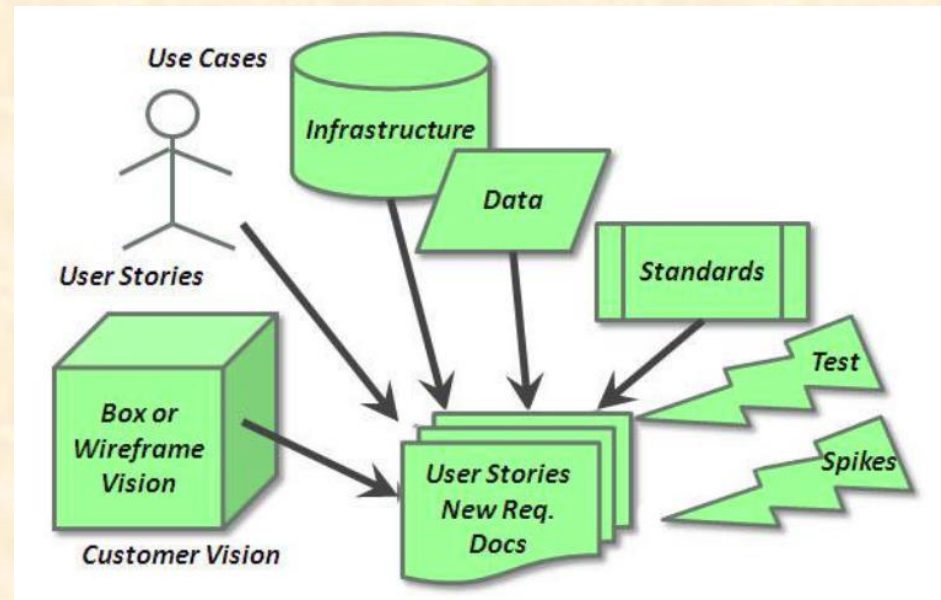
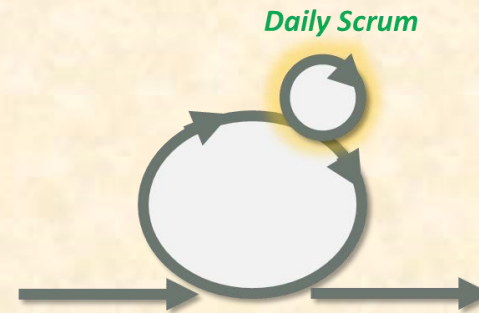
Getting Stories Ready for the Development
(Cross-functional Dev & Test) Teams
Looking at XP, Scrum, Kanban or Lean



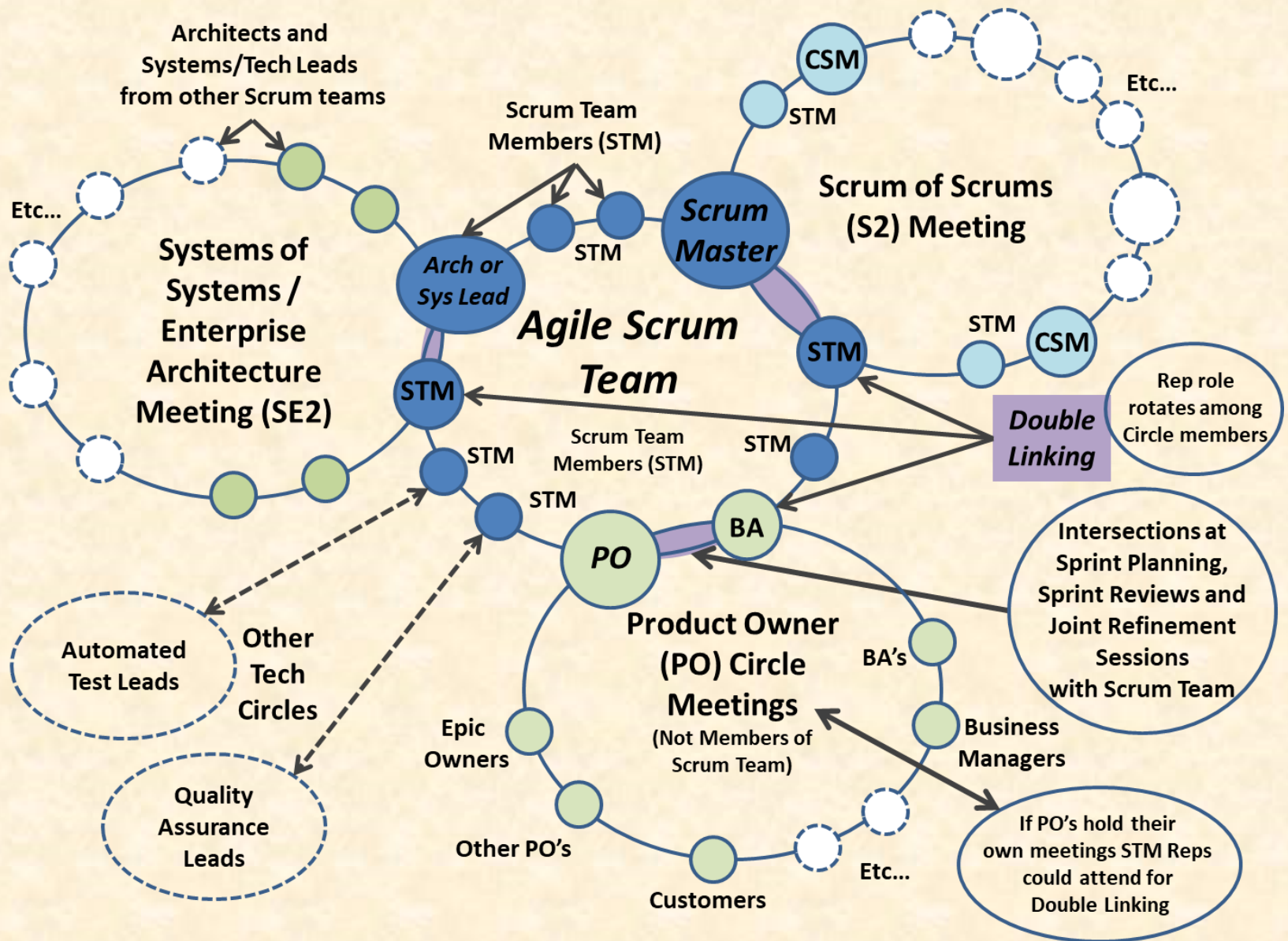
www.GregMester.com

Agenda

- Agile Story Telling
 - Gathering Requirements
 - Information Circles
 - Change
 - Right Sizing of Stories
 - Value / Prioritize
 - Feature Decomposition
 - Time Box Consideration
 - Refinement of Stories
 - Refinement Flow
 - Work Flow
 - Health of the Backlog
 - Agile Extras
- Appendix:
 - Different Agile Environments
 - XP, Scrum, Kanban, Lean

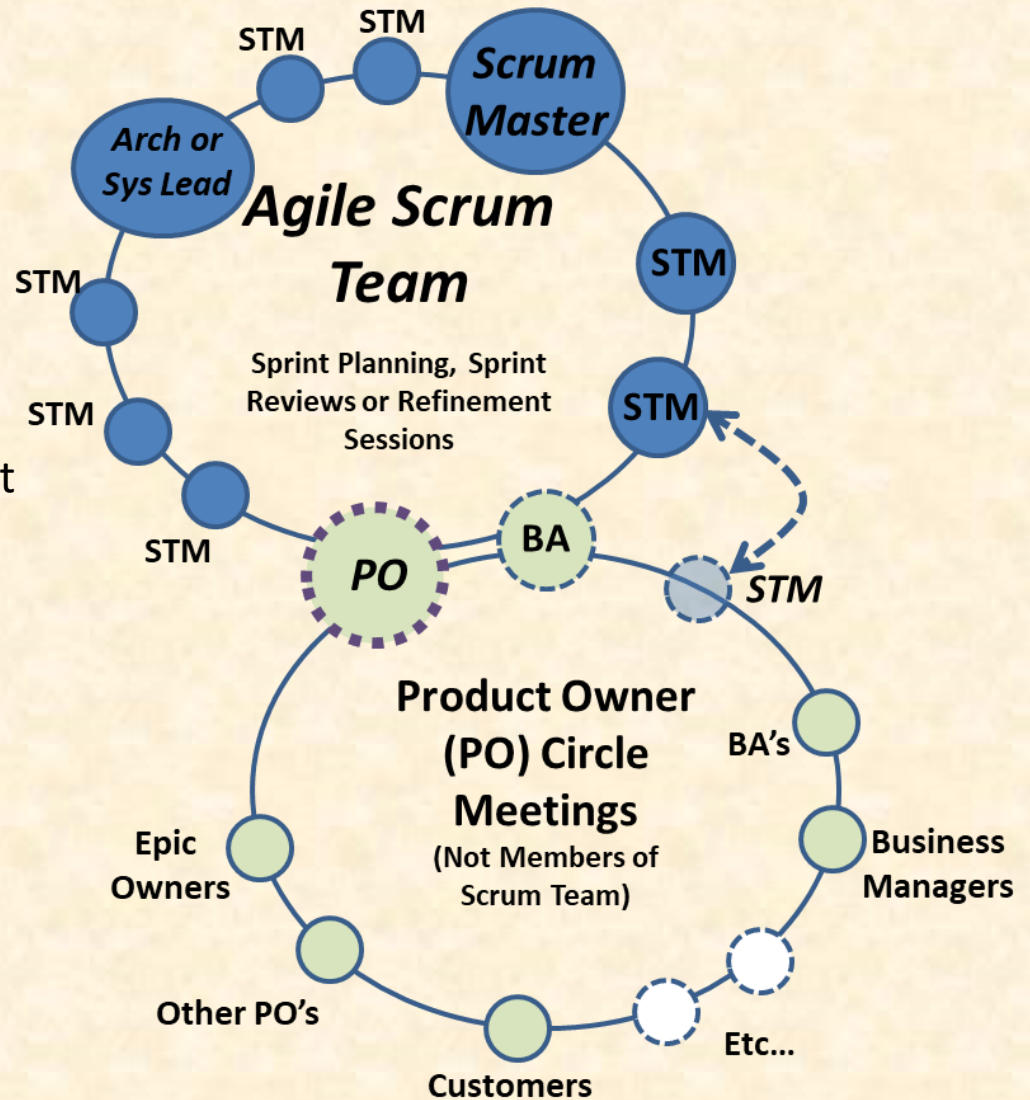


Gathering Requirements: Information Circles



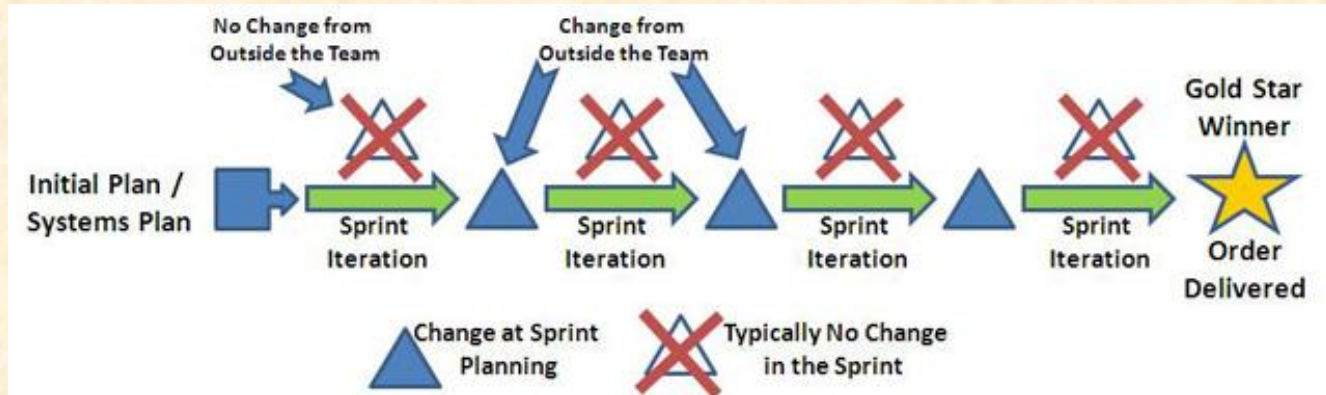
Information Circles – Building the Backlog

- Attending Demonstrations
- Information Gathering Sessions
- Product Owner / Business Meetings / Customer Meetings
 - Don't let the PO go without the BA
 - Risk Information Lost
- Vision Meetings
- Proof of Concepts or Spike Demonstrations

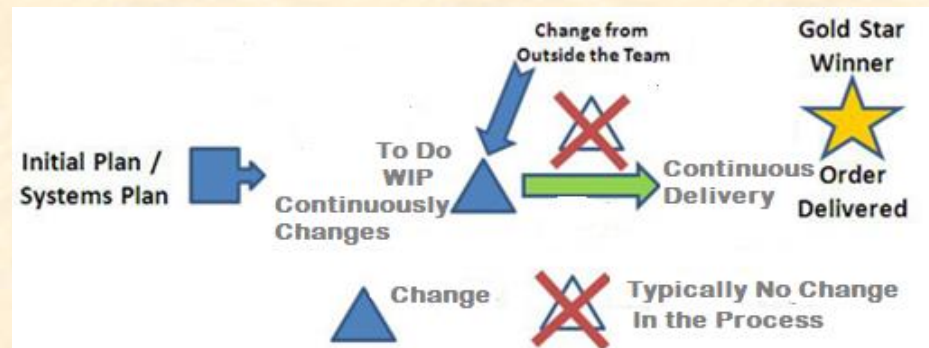


Rate of Change

- Myth: Agile Development = Requirements Change at Any Time 24x7
- Truth: Requirements Don't Change While Developing
- **Scrum Change**: Time Box Based, Change at Beginning of Each Sprint



- **Kanban Change**: To Do WIP Limit Based – Add new Story/Requirement when one is Pulled from “To Do” to “In Process”

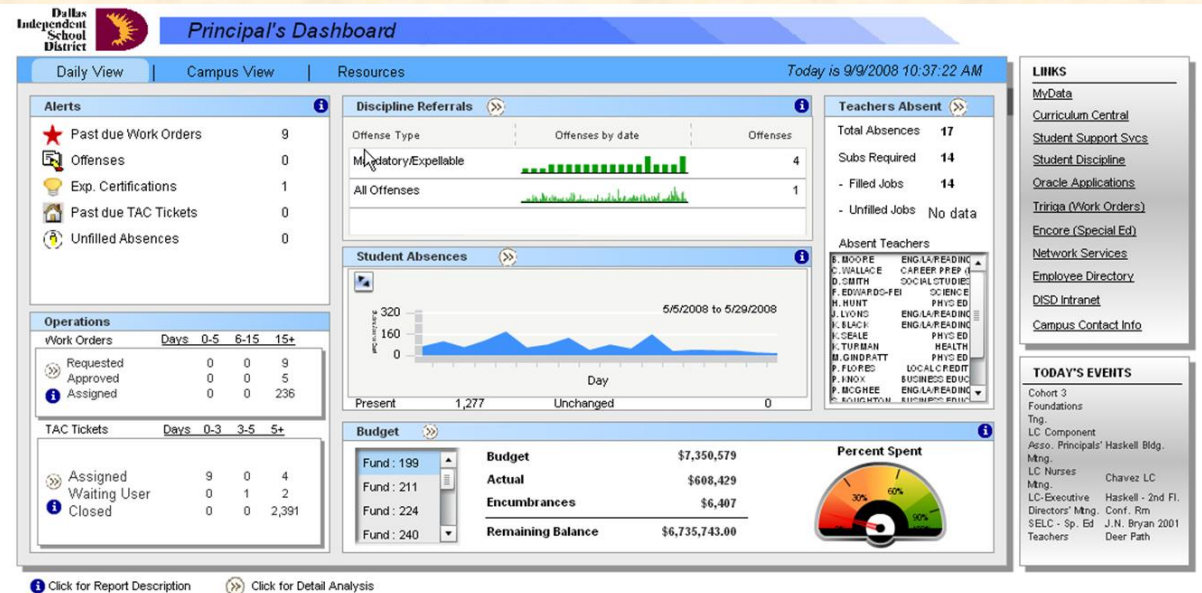


Scrum Sprint Calendars

Week	Sprint	Monday	Tuesday	Wednesday	Thursday	Friday
Wk #1	Sprint #1			Sprint #1 US/TA Plan (3.5 hrs) -- SE Ops (0.5 hrs.)		
Wk #2		Backlog Refinement Session (2 hrs)	Weekly review (1 hr)	SE Ops (0.5 hrs.)	Sprint US Priority Meeting (0.5 hrs)	
Wk #3		Sprint Review (1 hr)	Sprint Retro. (1 hr) -- Weekly review (1 hr)	Sprint #2 US/TA Plan (3.5 hrs) -- SE Ops (0.5 hrs.)		
Wk #4	Sprint #2	Backlog Refinement Session (2 hrs)	Weekly review (1 hr)	SE Ops (0.5 hrs.)	Sprint US Priority Meeting (0.5 hrs)	
Wk #5		Sprint Review (1 hr)	Sprint Retro. (1 hr) -- Weekly review (1 hr)			

Vision & Value

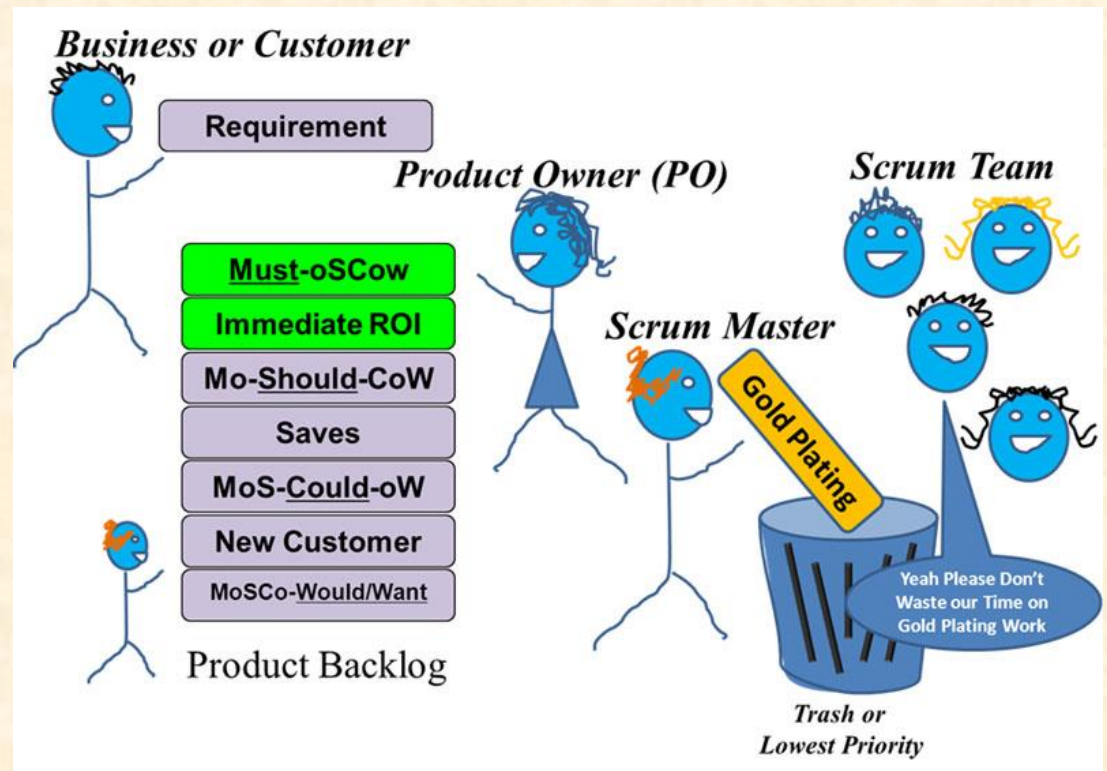
- Brand
 - User Profiles
 - Goals
 - Site Audit
 - User Needs
 - Create User Stories
 - Site Map
 - Wireframes
- Awareness
 - Engagement
 - Retention / Return
 - Referral
 - Revenue



Prioritize MoSCoW

- MoSCoW and ROI with the following prioritized order:

1. Customer MUST Have (The “M” in MoSCoW)
2. Brings in Immediate CASH INCOME – ROI
3. Customer SHOULD Have (The “S” in MoSCoW)
4. Saves the Company Money – ROI
5. Customer COULD Have
6. New Customer – Future ROI
7. Won’t Have (Would like)
8. Gold Plating – Only after the company pays bonuses and donates to charity



Prioritize – Cash Flow & Customer



Vision (Prioritizing Features and Build Pieces)

Business Value Delivery (1/11) Metrics	Speed to market By Feature Group	Average transaction processing time*	Business satisfaction score*	Business benefits delivered*	Revenue growth*	Transactions impacted (during an outage)*
	Order2Install fallout*	First call resolution (FCR)*	Order2Install time*	Average handle time (AHT)*	Transactions exceeding target processing time*	
Finance & Planning (1/10) Metrics	Spend by project and feature.	Spend on strategic applications*	Average cost per user story*	Average cost per story point*	Standard deviation for cost per user story*	
	IT spend to Business Services revenue *	Ratio of (POR + SE) to total spend*	Average cost per requirement*	Feature budget variance*		
Delivery Performance (3/14) Metrics	Speed to market.	Release health*	Release duration variance*	Requirement stability index*	Test cases passed at go-live*	Rework percentage*
	Velocity	Delivery volume*	Project schedule variance*	Story point acceptance*	Alignment to reference architecture*	QA defect to test case execution ratio*
	Acceleration	Defects reopened*				
Operational Metrics (3/9) Metrics	Mean time to resolution (MTTR)	Mean time between incidents (MTBI)	Incidents per user per month*	Post-deployment defect volume*	Average impaired minutes*	Change success percentage*
	Application availability	Emergency changes score*	Outage root cause*			













Colors correspond to Metric Business Rules Condition

Metrics without color indicate the Metric page is in-progress / Data Not Available

Click Here for Metric Pages

Feature and Story Decomposition

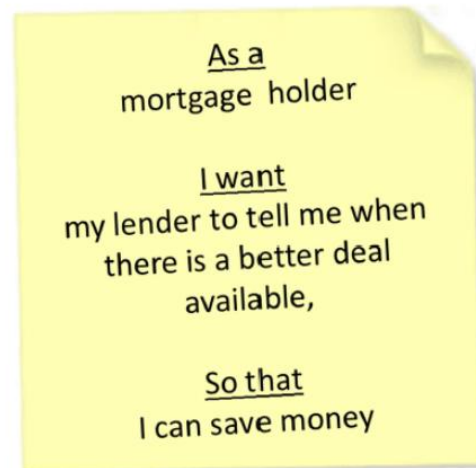
- Feature Characteristics
 - Very Large Story
 - Typically Lots of “And’s” in the description
 - Can bridge more than one Sprint
 - Made up of more than one Story/Requirement
 - Integration Testing
- Story Characteristics
 - Small Size (INVEST)
 - Very Few if any “And’s” in the descriptions
 - Complete in one Sprint
 - Ideally Complete in a day or two
 - Includes Dev & Testing
 - Take into account Automation Testing

	Sprint 1	Sprint 2	Sprint 3	Sprint 4
Feature 1	 			
Feature 2		 		
Feature 3			 	
Feature 4				 
Feature 5				
Velocity	3	3	3	3

Story Writing

- INVEST

- Independent
- Negotiable
- Valuable
- Estimable
- Small
- Testable



As a
mortgage holder

I want
my lender to tell me when
there is a better deal
available,

So that
I can save money

- Definition of Done

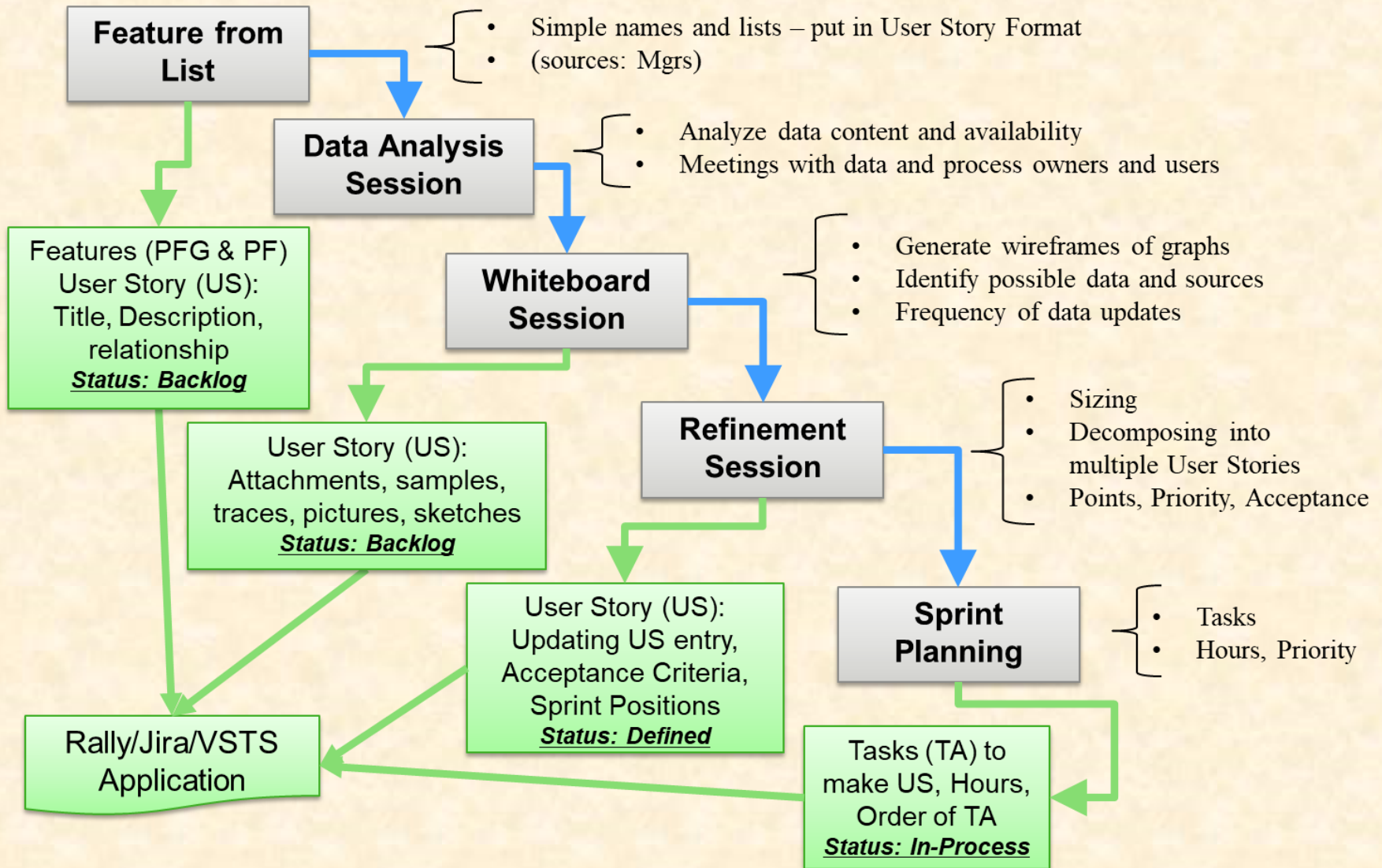
- Working SW
- Unit Tests
- Code review
- Installer
- Tests
- Functional Tests
- Successful Demo to Customer (Product Owner)
- Regression Tests
- Documentation

Exercise – 6 Word Stories (5 minutes optional)

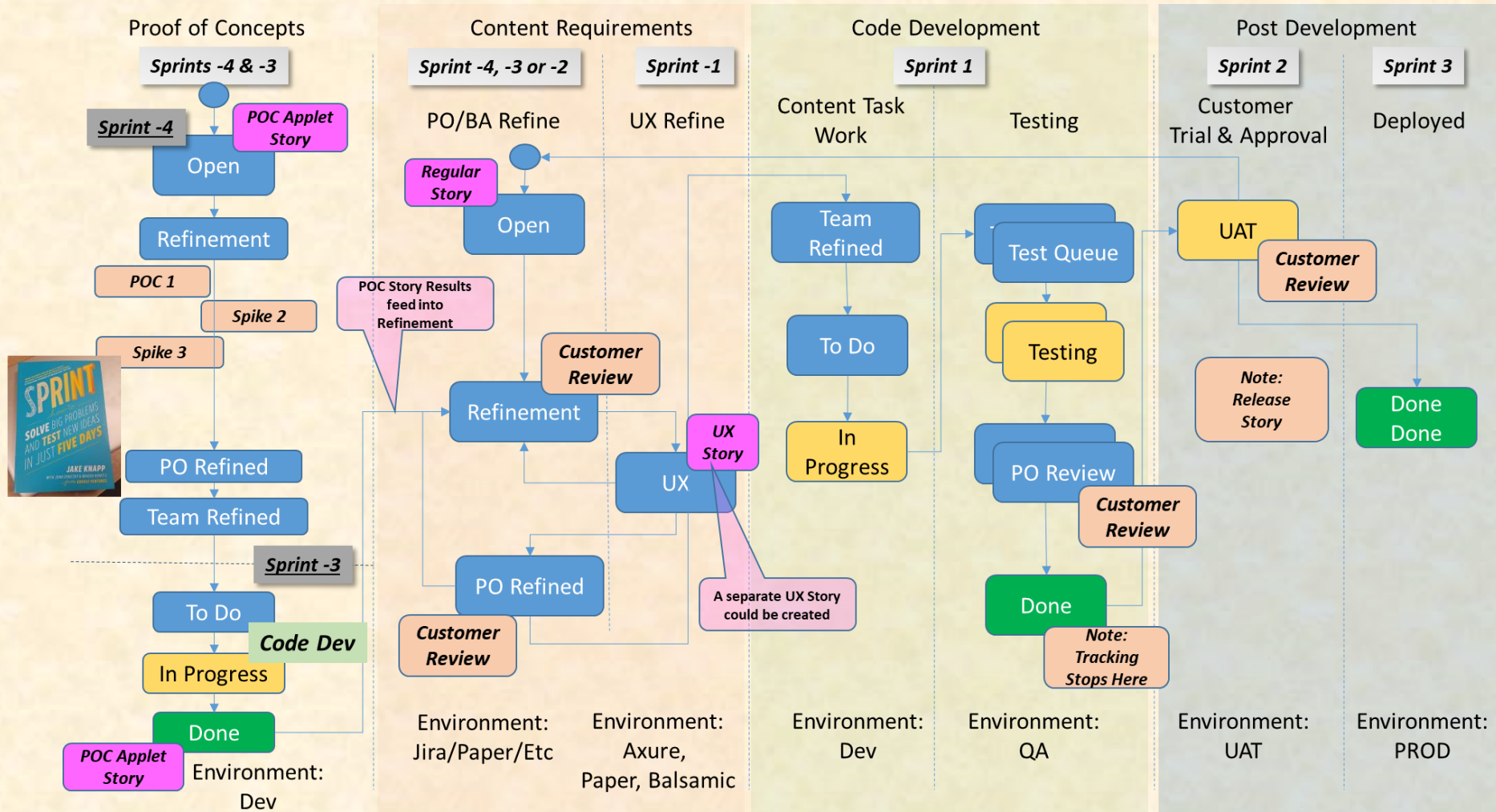
- There are 6 elements to Business Stories
 - State who the Story is about
 - Describe what is being done
 - Timeframe, when of the story
 - Define Why the Actions are Occurring
 - Explain how the actions in the story is being taken
 - Provide Verifiable quantitative evidence of assertion & performance
 - Attempt to write a 6 word Title to Describe your Story
 - Example:
 - As an trouble shooter (TS), I want to move a Email from Triage to the next flow process as New, Bug, Enhancement or Reject within 10 minutes to ensure submission email is properly worked in a timely manner.
 - As an TS, I want to our Triage Email to the next flow process as New, Bug, Enhancement or Reject within 10 minutes to ensure email is properly worked in a timely manner.
 - TS Triage Email to next stage after review (10 Mins)
-
- Step 1: 1 minute
- Step 2: 1 minute
- Step 3: 1 minute

From lecture and class with:
Thomas Cagley - Storytelling

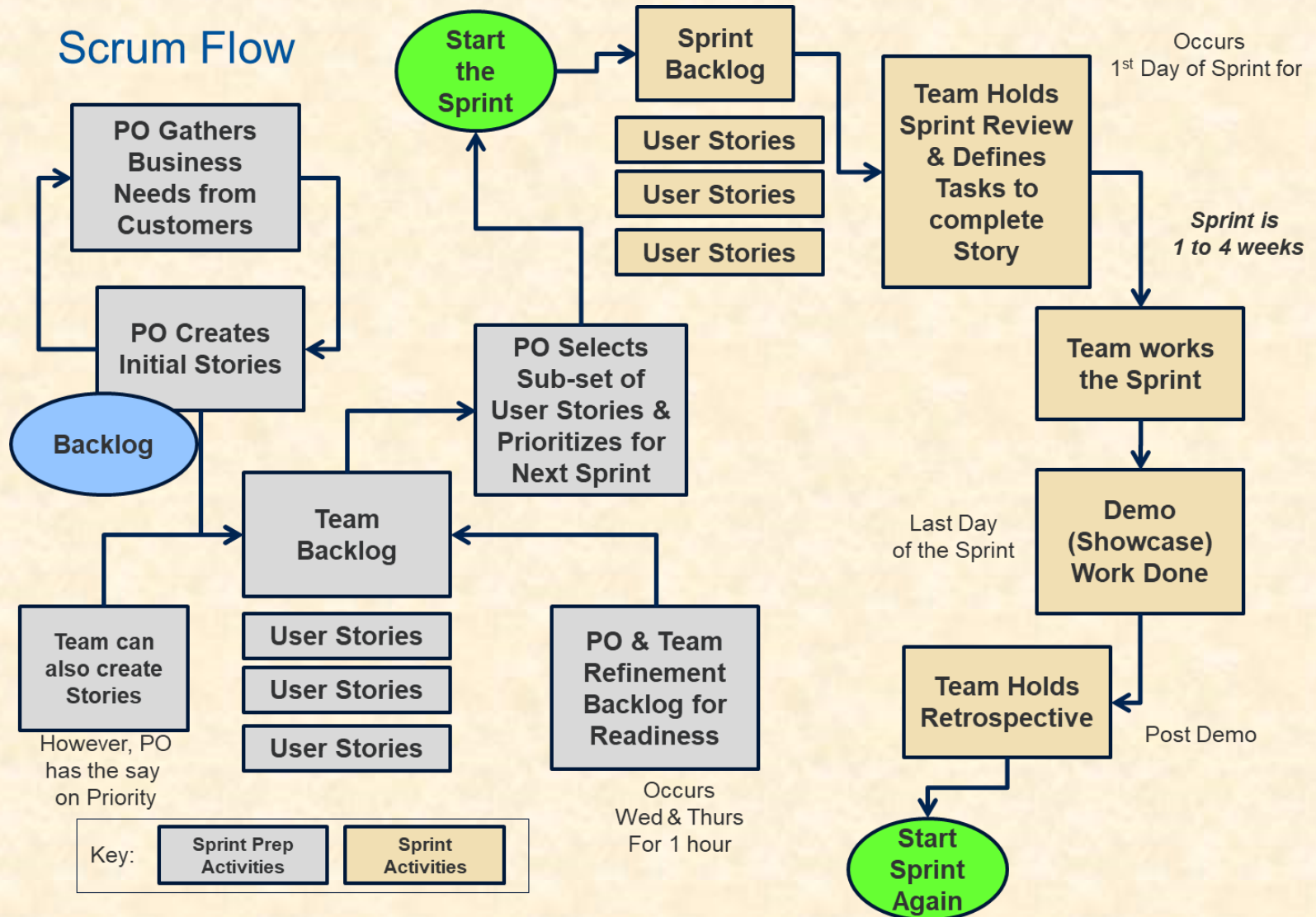
Story Refinement Flow



Measuring Story Refinement Flow (Proof of Concepts, Spikes & UX Proofs)

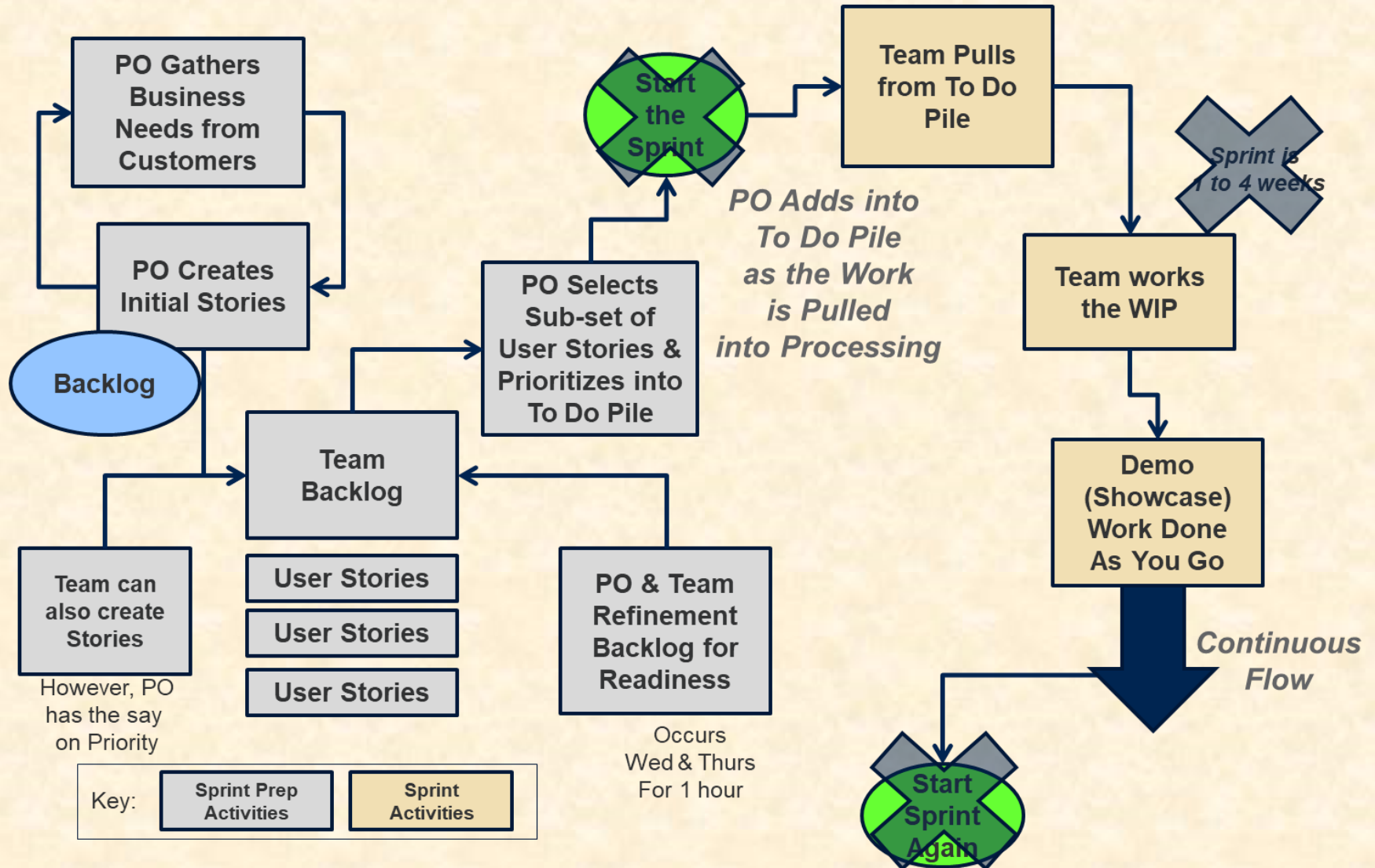


Scrum Flow

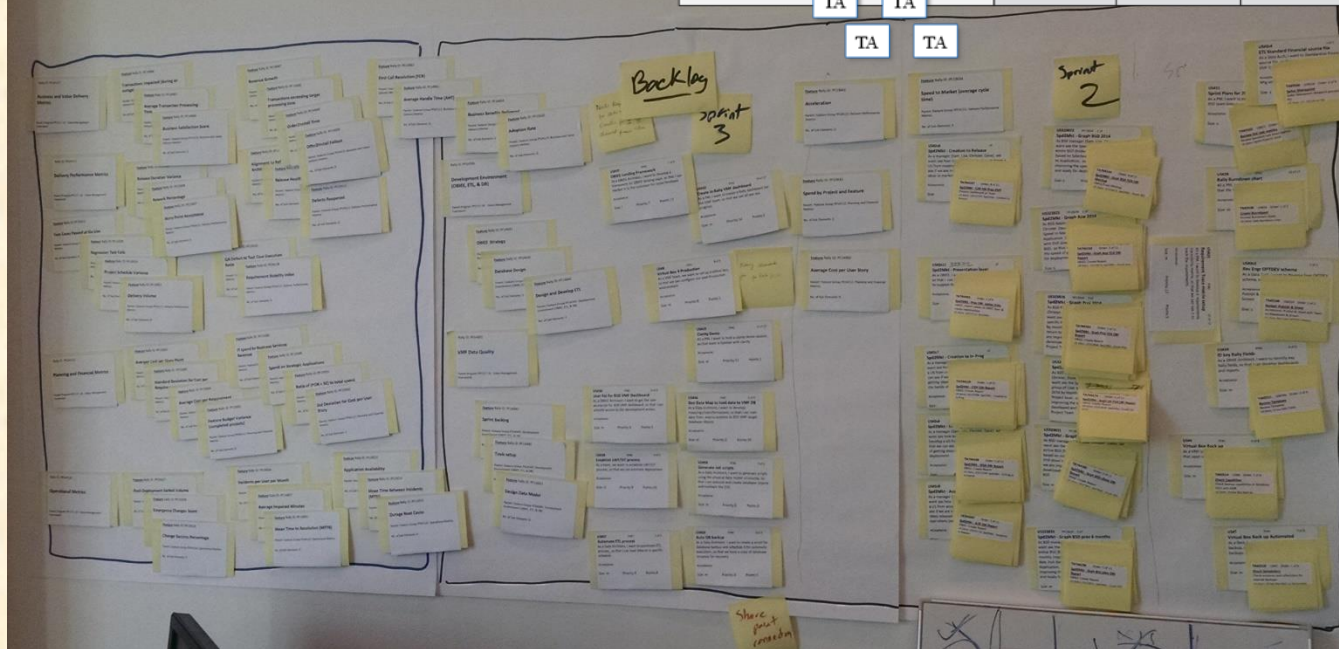
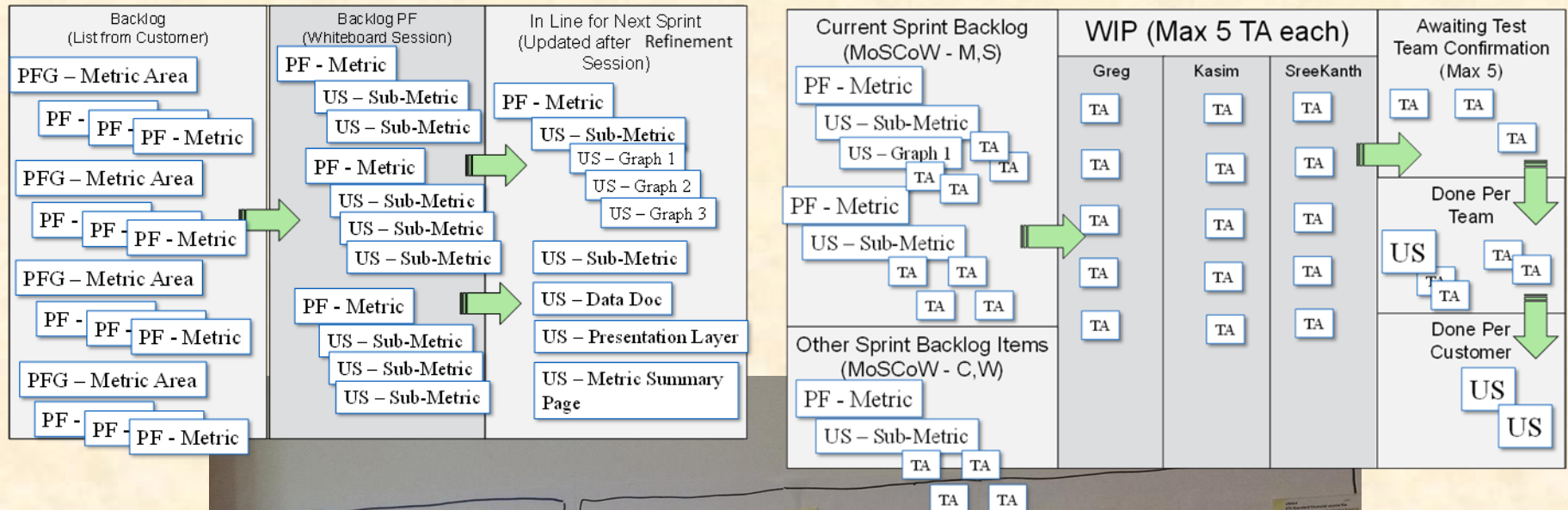


Kanban Flow

Kanban Flow

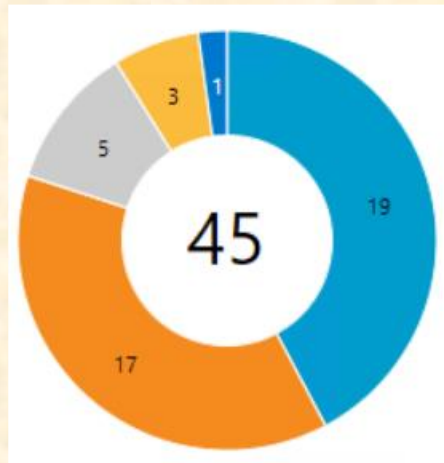


Backlog Refinement



Measuring Story Refinement Health of Backlog (Simple Pie Charts)

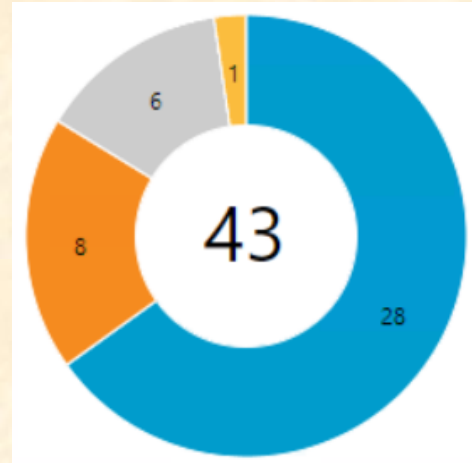
Current Sprint



■ Ready for T... ■ Dev ■ Ready for P...
■ Ready for D... ■ Closed

Development Team
Task
Concentration

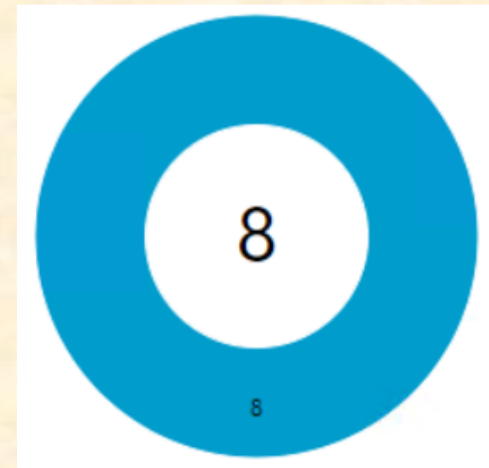
Sprint +1



■ Ready for D... ■ New ■ Removed ■ Closed

PO / BSA
Story
Concentration

Sprint +2

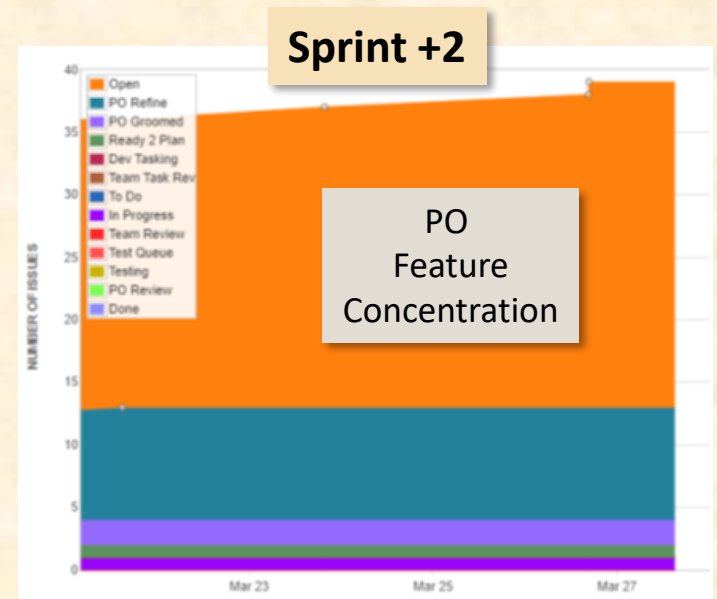
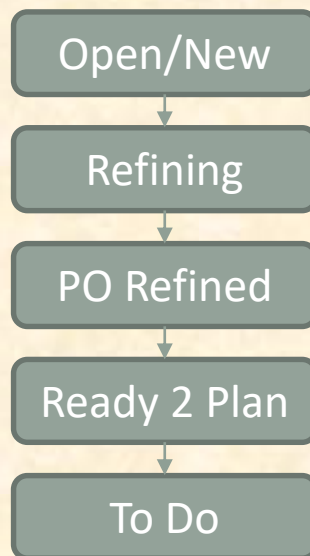
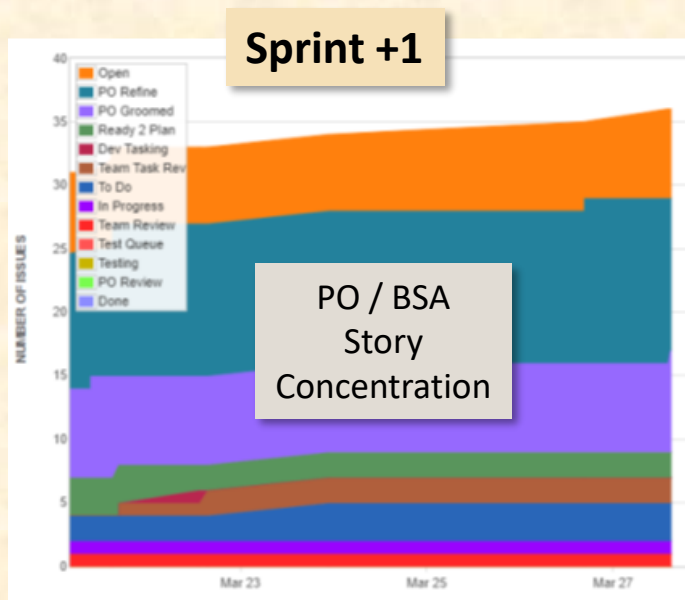
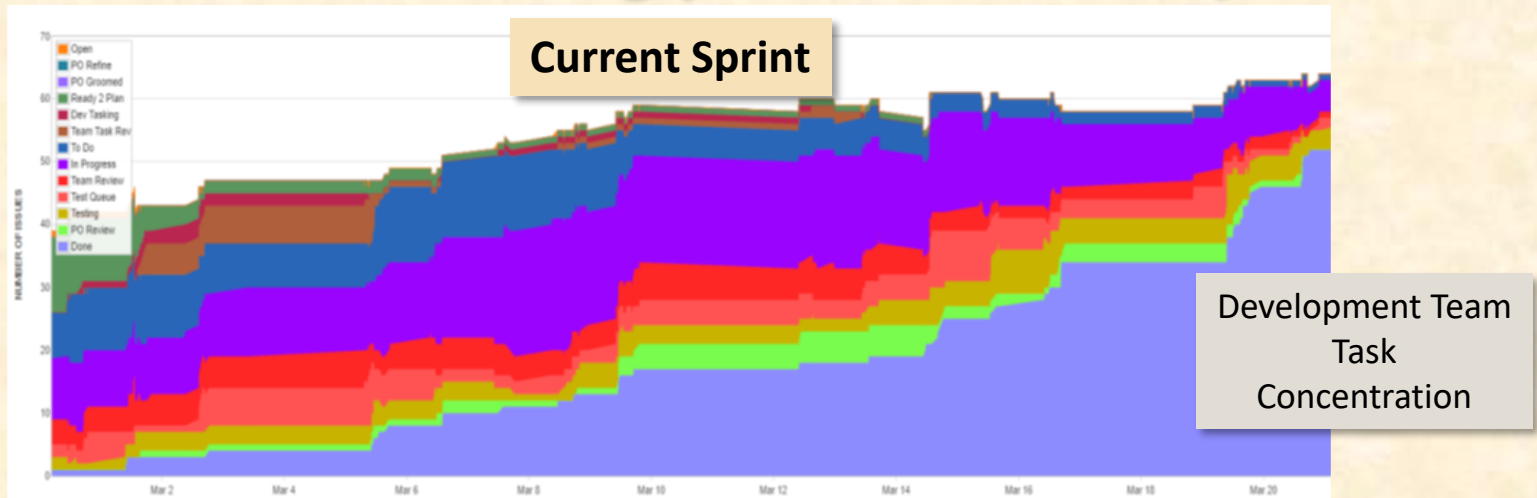


■ New

PO
Feature
Concentration

Measuring Story Refinement

Health of Backlog (Cumulative Flow)



Agile Extras

- Mixing it all
 - 3 Amigos (quasi XP) – Used in Scrum & Kanban
 - Where PO/BA, Dev and Tester get together at the beginning of Sprint or Before picking the Story from the To Do pile
 - The three work together to determine Acceptance Criteria
 - PO/BA answers questions from Dev & Tester
 - Maybe Code in Cucumber and even a little TDD
 - Lean (Refinement)
 - Only Refine a Sprint or Two of Stories
 - Any more is a waste of time and not Lean Thinking
 - Over Committing is **Not** Lean Thinking (Watch Velocity)
 - Don't waste time Refining Stories that are not going to get "Done" in Next Sprints
 - What if the Priority Changes?
 - 20% Over-Commit is 30min to 2 hours of wasted time per Sprint x Number of people on the Team



Closing: Reference Materials



Thank You and Happy Scrumming

- **Contact Information:**
 - Would you like an Agile Mentor?
 - Agile Training
 - Find out what other Agile Groups are in the Philadelphia area

www.GregMester.com

Email: greg.mester@gregmester.com

Twitter: @gemphilly

- Twitter List to follow Agilist around the country
 - ***<http://www.gregmester.com/twitter-agilist/>***
 - ***<https://twitter.com/gemphilly/lists/agilists>***



Appendix Information

- The following slides are a few extra slides that briefly talk to the various Agile Methodologies

Quick Battle Summary

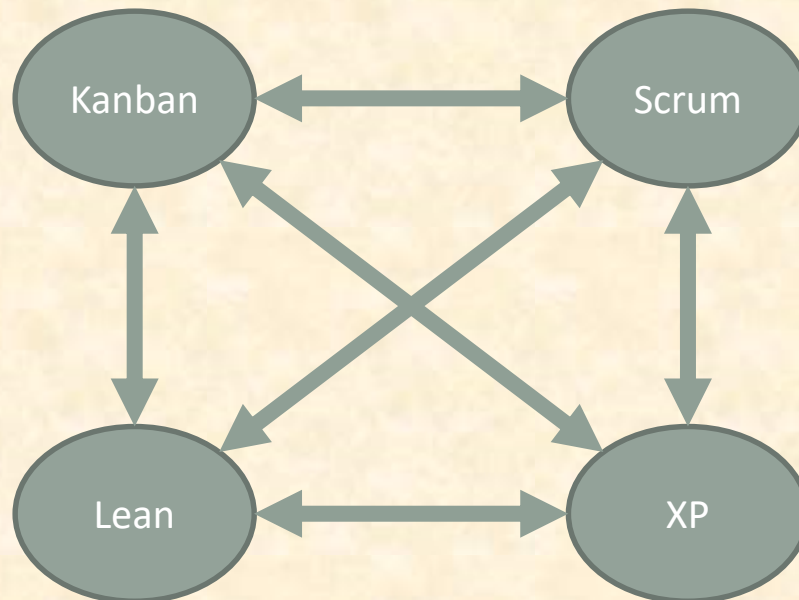
- XP
 - **Everyone Working Together** to Solve a Problem at The Same Time
 - Pair Programming, Unit Testing, Integration Testing
- Scrum
 - **Time Boxed**
 - Teaming to solve a problem, but also some me time
 - Cross Trained
 - **Team Delivery**
- Kanban
 - **Continuous Pull** System
 - More Individual Approaches
 - Work In Process or Progress (**WIP**) Limits
 - Prevents too much or too little work from entering the system
 - Quasi-Cross Trained
- Lean
 - Eliminating Waste
 - Waste in Time – Waiting
 - Maximize **Value**

Scrum ↔ Kanban



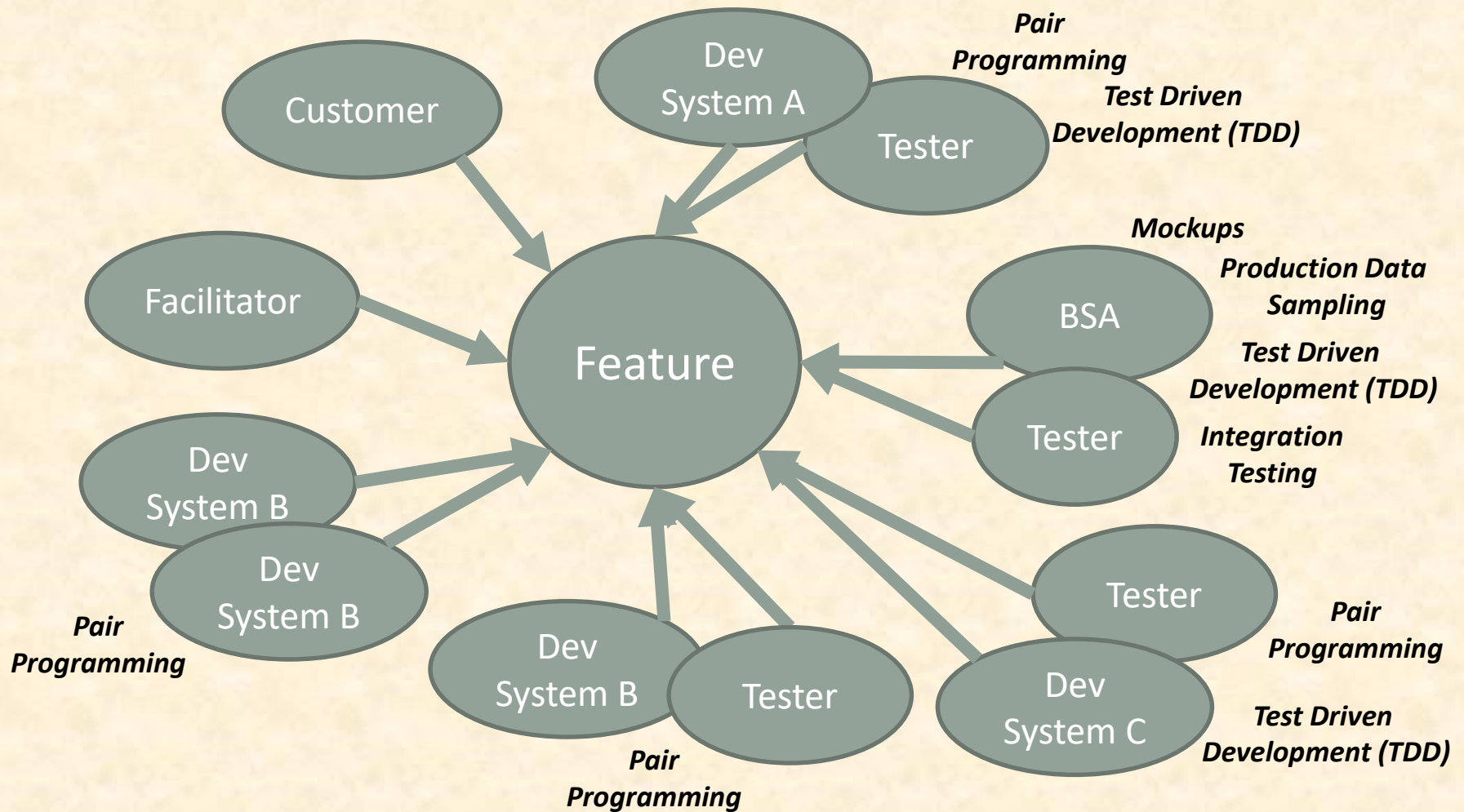
Agile Systems or Methodologies

- Agile Manifesto
 - We are uncovering better ways of developing software by doing it and helping others do it.
 - Through this work we have come to value:
 - Individuals and interactions over processes and tools
 - Working software over comprehensive documentation
 - Customer collaboration over contract negotiation
 - Responding to change over following a plan



***Business Analysis
Has to Be Agile***

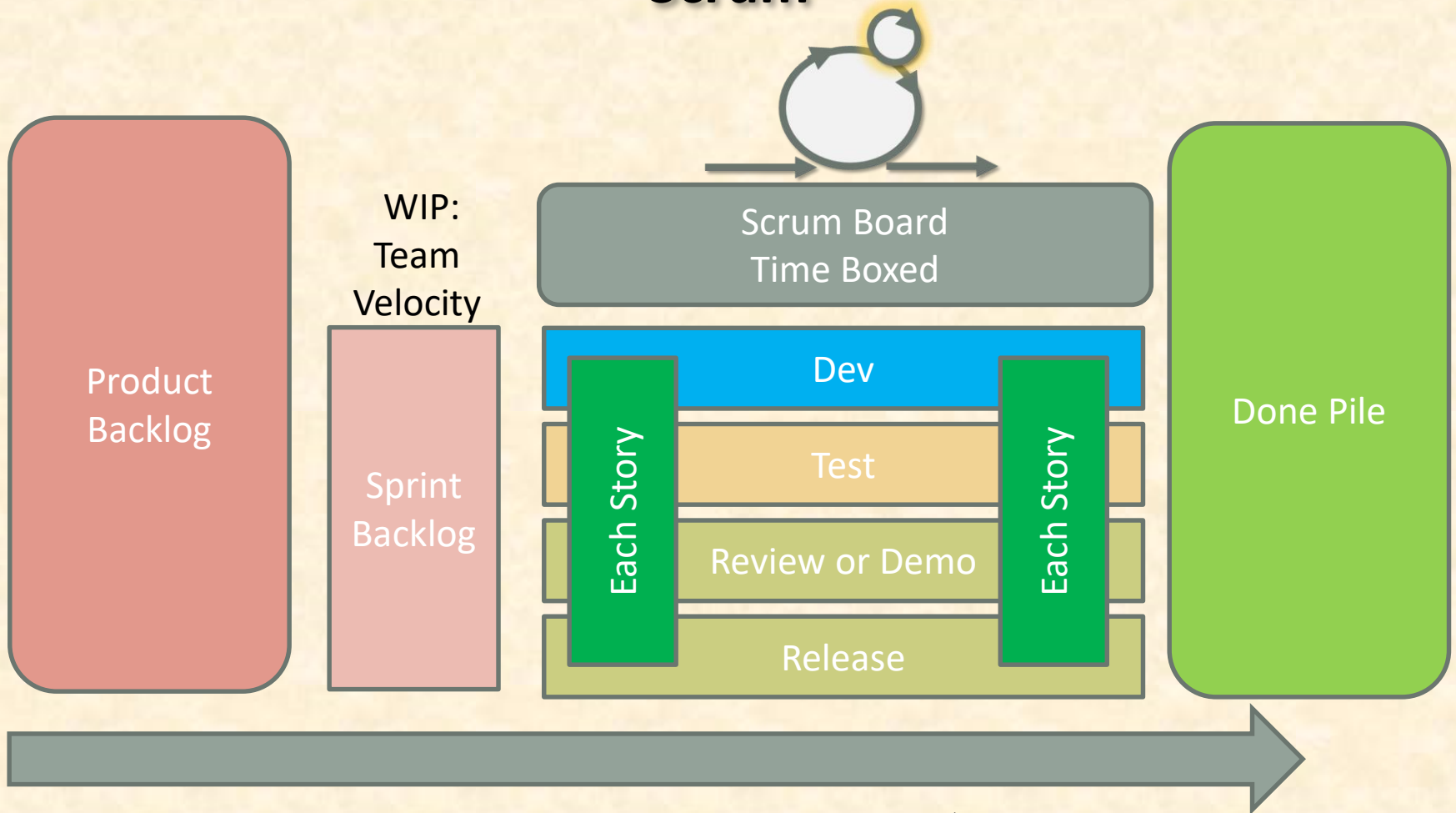
Extreme Programming (XP)



“XP is a lightweight methodology for small to medium sized teams developing software in the face of vague or rapidly changing requirements”

--- **Kent Beck**

Scrum

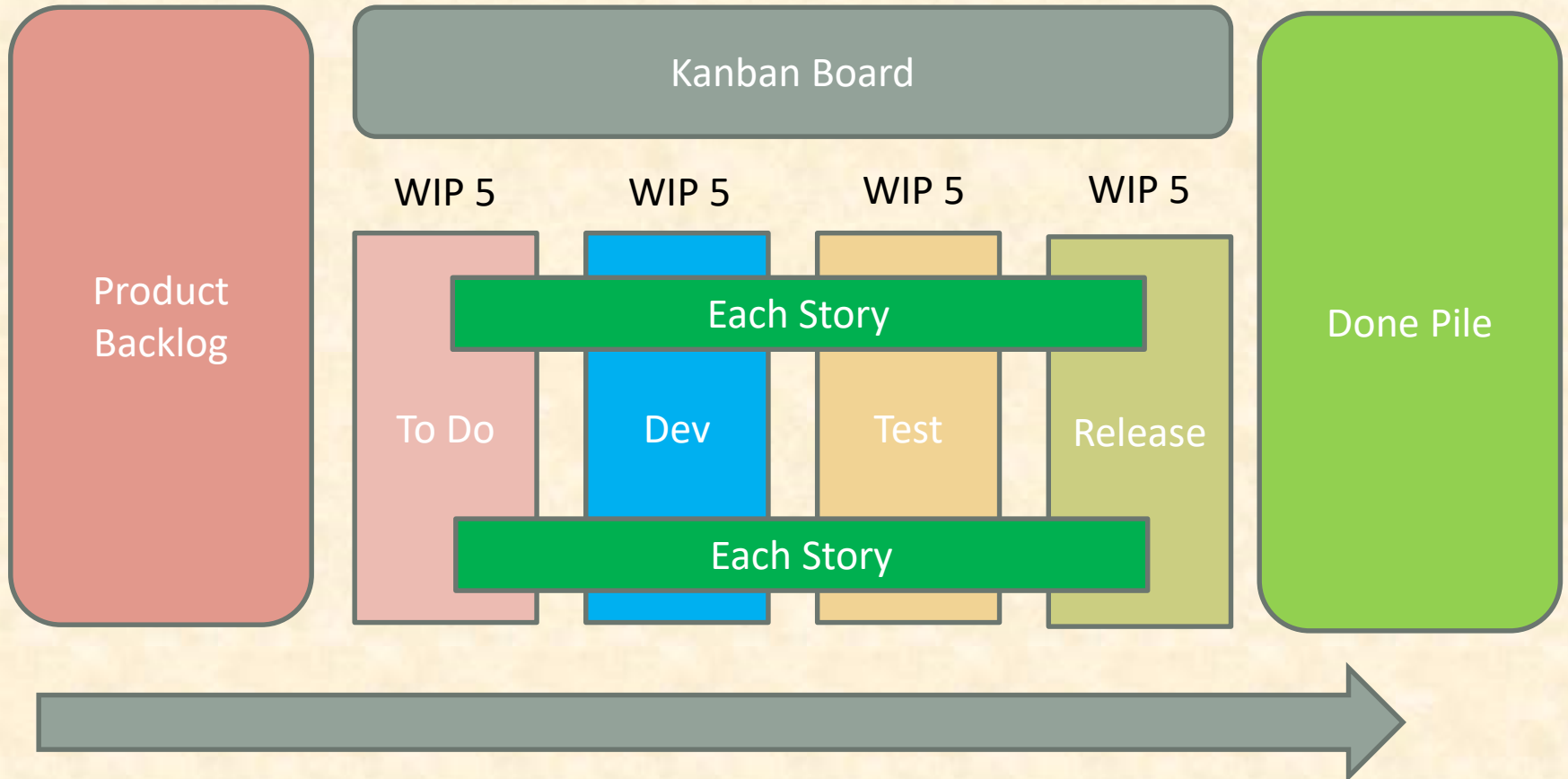


Roles:

- ✓ Product Owner
- ✓ Dev Team
- ✓ Scrum Master

- ✓ Time-Box
- ✓ Planning (Time Box and Daily)
 - × Continuous Flow
- ✓ Cross Functional As Much as Possible
- ✓ Retrospective

Kanban

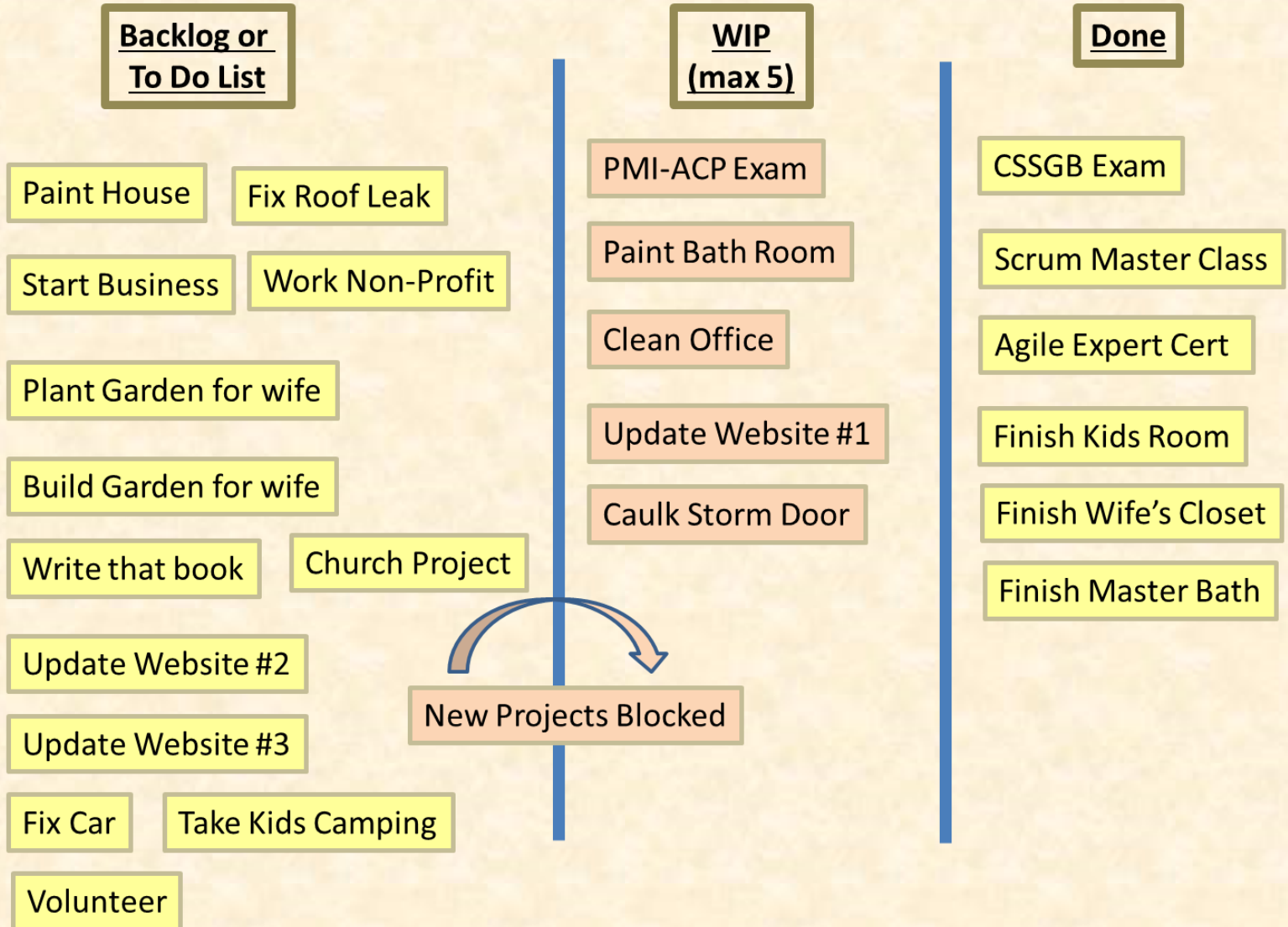


Roles

- × Product Owner
- × Dev Team
- ✓ Dev & Tester
- × Scrum Master

- × No Time-Box
- × Planning (Time Box and Daily)
- ✓ Continuous Flow
- ✓ Cross Functional When WIP is Maxed out
- × Retrospective (optional)

Personal Kanban Board



Agile Systems or Methodologies

- Mixing it all
 - Scrum-ban
 - Combining WIP Limits into Scrum Boards and # of Tasks worked at a time
 - 3 Amigos (quasi XP) – Used in Scrum & Kanban
 - Where PO/BA, Dev and Tester get together at the beginning of Sprint or Before picking the Story from the To Do pile
 - The three work together to determine Acceptance Criteria
 - PO/BA answers questions from Dev & Tester
 - Maybe Code in Cucumber and even a little TDD
 - Kanban
 - Adding Retrospectives
 - Need to get better as a Team
 - Adding Time Boxes
 - Just to clear Done Pile from the tracking systems
 - Lean (Refinement)
 - Only Refine a Sprint or Two of Stories
 - Any more is a waste of time and not Lean Thinking
 - Over Committing is **Not** Lean Thinking (Watch Velocity) – Don't waste time Refining Stories that are not going to get "Done" in Next Sprints – What if the Priority Changes?
 - 20% Over-Commit is 30min to 2 hours of wasted time per Sprint x Number of people on the Team

