

---

# Practices of a Great Product Owner

---

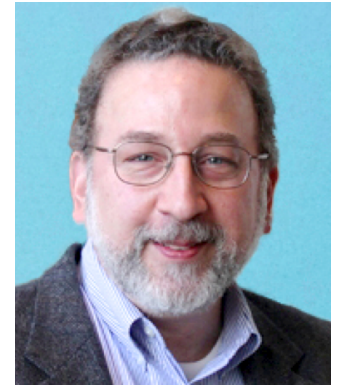


**Bob Galen**  
President & Principal Consultant  
RGCG, LLC  
[bob@rgalen.com](mailto:bob@rgalen.com)

# Introduction

## Bob Galen

- Somewhere 'north' of 30 years experience ☺
- Various lifecycles – Waterfall variants, RUP, Agile, Chaos...
- Various domains – SaaS, Medical, Financial Services, Computer & Storage Systems, eCommerce, and Telecommunications
- Developer first, then Project Management / Leadership, then Testing
- Leveraged 'pieces' of Scrum in late 90's; before 'agile' was 'Agile'
- Agility @ Lucent in 2000 – 2001 using Extreme Programming
- Formally using Scrum since 2000
- Currently an independent Agile Coach (CSC – Certified Scrum Coach, one of 50 world-wide; 20+ in North America)
  - at RGCG, LLC and Director of Agile Solutions at Zenergy Technologies
- From Cary, North Carolina
- Connect w/ me via LinkedIn and Twitter if you wish...



**Bias Disclaimer:**  
**Agile is THE BEST Methodology for Software Development...**  
**However, NOT a Silver Bullet!**

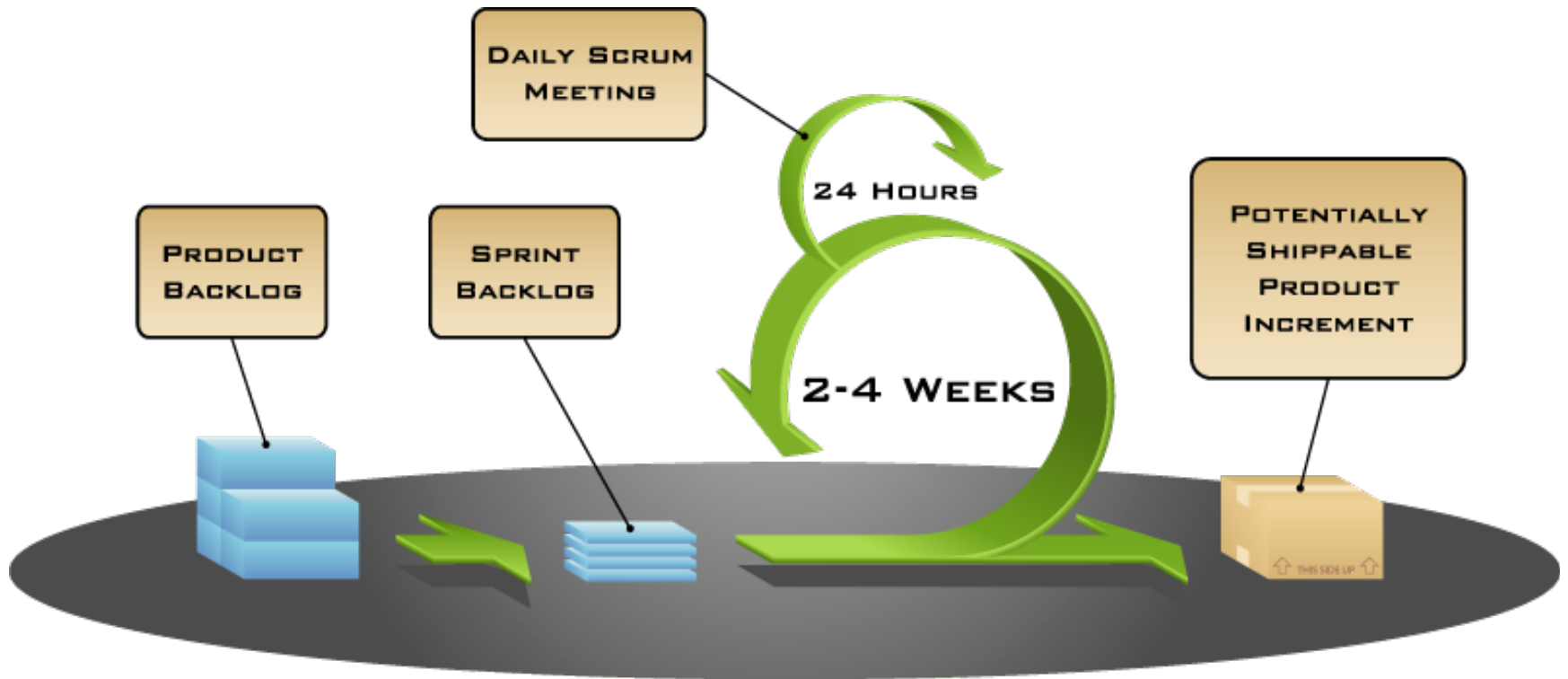


# Outline

## Introduction

- Role of the “Product Owner”
- Product Backlogs
- Sprint Dynamics
- Goals & Criteria
- Role of the Product Owner at-Scale
- Q&A

# The SCRUM Framework



COPYRIGHT © 2005, MOUNTAIN GOAT SOFTWARE

# Inside Out?

- Many Product Owners are conflicted...
  - Product Manager assigned...w/o sufficient time, too heavily customer & stakeholder facing
  - BA or others assigned...w/o PM skills
  - Team / Scrum Master tries to do it...just messy
  - Little/no PO training
  
- Serious role within Scrum and should be handled that way
  
- Premise: Product Owners first responsibility is towards their Team!
  - ✓ Backlog & work orchestration
  - ✓ Interaction & feedback
  - ✓ Goal setting & acceptance
  - ✓ Leadership & partnership

# Basics of the Role

## Part of the Challenge – Breadth



- **Part Product Manager** – traditional sense from Pragmatic Marketing
  - [www.pragmaticmarketing.com](http://www.pragmaticmarketing.com), 37 ‘activities’ as part of PM role; PO only covers 8-10 of those
- **Part Project Manager** – release planning, cross-functional & external dependencies in larger-scale environments
- **Part Leader** – partnered with the Scrum Master, influence & motivation capabilities, congruent leadership, its *your* team
- **Part Business Analyst** – creating user stories and other requirement artifacts; UI design often a ‘stretch’

# Can there be Only 1?

## Are they a Part of the Team?



- It's a complex role at best
  - Complex real-world environments, even tougher
- Scrum Yahoo Group there was discussion in the Fall of 2008 regarding whether it could be done by a small group. Schwaber stated that “there can be only 1”.
  - Respectfully I understand, but sort of disagree. What if you don't have that ‘perfect’ person?
- Oh, and are they a part of the team?
  - For example, do they attend the Sprint Retrospectives?
  - Can they perform work within the Sprint?

# Product Owner role according to Roman Pichler

- Closely collaborate with the team on an ongoing basis and to guide and direct the team:
  - By actively managing the product backlog,
  - Answering questions when they arise,
  - Providing feedback, and
  - Signing off work results.
- In simple terms, the product owner sits in the driver's seat, deciding what should be done and when the software should be shipped.

<http://www.scrumalliance.org/articles/44-being-an-effective-product-owner>



---

# Product Owner role according to Roman Pichler

- I have found three things particularly helpful for product owners—
  1. A thorough understanding of the customer needs,
  2. An active stakeholder management, and
  3. A basic knowledge of how software is developed and deployed.

<http://www.scrumalliance.org/articles/44-being-an-effective-product-owner>

# Product Backlogs

# Product Backlog

## Simple list, or something else?

- All work vs. feature work? One list vs. many?
  - Features, Technical, Quality, Release, etc.
  - Excel Product Backlog Items (PBIs) vs. User Stories vs. something else altogether? Connecting to other artifacts?
  - *Think in terms of broad nuance; with multiple 'threads' weaving their way through the Backlog*
  
- How do you orchestrate or influence – Emergent Practices?
  - Architecture, feature sets, usability, design, etc.
  - Balancing look-ahead?

# Creating a 'New' Backlog

- **User Story Writing Workshops**
  - Collaborative & fast
  - A sense of where you're going
  - Drive them using Roles
  - I prefer User Stories, but the premise is the same...
- **Partitioning the Backlog - workflow**
  - **Opening Moves** – emergent understanding, beginning well
  - **Middle Game** – stabilization, value, mass
  - **End Game** – integration, quality, customer delivery

# Backlog ‘Tension’ Points

- How many items? Size – does it matter? Are they all the same?
- Prioritization & Valuation methods
- *FutureCast* – painting a compelling picture of the “Tactical Now” vs. the “Strategic Later” w/o scaring everyone to death...

## *Granularity heuristic*

*Use the 20/30/50 rule. 20% proper stories ready to roll. 30% are epics - bigger stories that will eventually be split out into smaller fine grained ones (only as needed). The last 50% are themes - vague ideas about long term product direction and I never put much effort here because it's almost always wrong.*

# Active Backlog Grooming

- Bring goals & stories to the table; but be open to change
- Listen actively
- Don't predetermine size nor complexity; trust your team
- Don't negotiate...collaborate
- Organic explorations of scope and options as you get closer to execution
- Explore execution dynamics – architecture & design, testing, non-functional, deployment, and risk
- Apply pressure on – value flow, quality & sustainable pace

# ‘Grooming’ your Backlogs

## Approaches?

- The Product Backlog is Organic and needs care & feeding
  - Reserve time for collaborative, team-based grooming meetings
  - ‘Assign’ individual stories to team members for vetting
  - Some combination of these two
- Keeping it interesting, grooming at 3 levels:
  - 20%: what’s right around the corner – are we ready?
  - 30%: what’s 3-4 sprints away – are we getting clearer?
  - 50%: what’s the future looking like? Break those things down...



# 'Grooming' your Backlogs

## Approaches?

- Judicious use of Research Spikes to gain understanding
- In some cases, run Sprint #0 types of iterations to get your feet under you
- Always looking for new 'contributions'
  - Technical, quality, release workflow, etc.
  - Rarely do they seem to be forthcoming; encourage them from the team
  - It's the Teams Backlog!



# A Tapestry that Includes Threads for...

Things to do...

- Features
- Value increments
- Architecture
- Design
- Process
- Quality
- Testing

In a Context-Based fashion...



- Deployment
- Regulatory
- Dependency
- Risk
- Feedback
- Customer timing
- Tempo

...Guiding us towards customer value

# Sprinting

# Sprint Dynamics

## Planning



- Always be ready; strategize with your Scrum Master
  - No surprises for the team!
  - You're part of the team, stay engaged in the entire process
  - Drive everything with 'goals'

### **Point of Sprint Planning**

1. To share and gain the teams' commitment toward the Sprint Goal
2. To identify the set of User Stories that align with and are feasible to deliver within the Sprint
3. To identify the Tasks associated with delivering those User Stories

***In that priority order and leading to goal-driven work***

# Sprint Dynamics

## Execution



- Stay engaged
  - Attend daily Scrum
  - Observe the trending; consider adjustments as the sprint evolves
  - Actively participate...perhaps in testing; certainly in grooming
  
- Looking-ahead
  - Grooming the Backlog in ALL dimensions
  - Collaborating with Customers & Stakeholders
  
- Observe & understand your team dynamics
  - Strengths, capabilities, weaknesses, etc.

# Sprint Review

## A Defining Moment

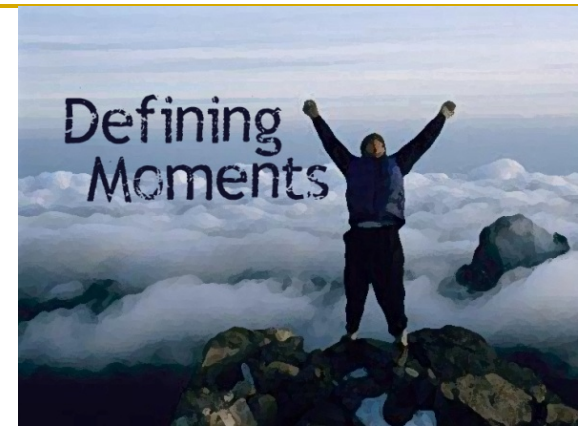


- Take ownership of attendance
  - Ensure key stakeholders are going to attend; If not, ask them to send someone
  - Make it compelling to them; sell the opportunity
  - Same 'ceremony' every Sprint?
  
- Help the team prepare
  - Have a 'script'; don't over-prepare, but DO NOT wing it
  - Product Owners should "Set the Stage" for the Sprint
  - It's not simply about features
    - Other artifacts, test automation, prototypes, etc.
  - Whole team approach

# Sprint Review

## A Defining Moment

- Serve as the M/C of the review
  - Ensure clarity of communication
  - Pace & transitions
- Call it! At a Story level and at a Sprint level...
  - Pass or Fail?
- Connect the dots
  - Relative to challenges in the sprint
  - Relative to release goals
  - Relative to customer expectations



# Sprint Retrospective



- And as a member of the team...
  - Attend the retrospective
  - Actively participate
  - Bring in an outside “business perspective”
  - It’s ok to share pressures
  - Quality impressions
  - Continuous Improvement impressions

# Goals



# Establishing Goals & Criteria

## Why it's Crucial?

- Agile teams are essentially self-directed, so plans don't drive behavior or success...
- People do and Goals drive the Team!
- The team then swarm around the goal(s), using their creativity and teamwork to figure out:
  - What's most important
  - How to achieve it
  - Always looking for simple & creative—20% solutions



# Notions of Done-ness

- Need to define “Done” from team members perspectives
- If you’re a developer, what does “I’m done with that story” mean?
  - ✓ Code complete
  - ✓ Code reviewed (paired)
  - ✓ Checked in – build successful
  - ✓ Unit tests developed – passed
  - ✓ Integration
  - ✓ QA collaboration
  - ✓ Run by the Product Owner
- Every type of task should have a definition of done-ness!  
How else could you estimate the work?



# Story Acceptance

- Each User Story should have acceptance criteria as part of the card
- They should focus on the verifiable behavior, business logic, for the story
- Typically, they are crafted by the Product Owner
  - Leveraging skills of Business Analysts and Testers
- Story acceptance tests are normally automated and run as part of feature acceptance AND regression

# Iteration / Goal Acceptance

- Each Scrum Sprint has a Product Owner determined Sprint Goal
- Usually sprint success is not determined by the exact number of completed stories or tasks
- Instead, what most important is meeting the spirit of the goal

***Deliver a 6 minute demonstration of the software that demonstrates our most compelling value features and achieves venture capital investment interest***

# Release Criteria

- Goals and objectives for the entire project release
- Usually they are multi-faceted, defining a broad set of conditions
  - Required artifacts
  - Testing activities or coverage levels
  - Quality or allowed defect levels
  - Results or performance metrics achievement levels
  - Collaboration with other groups
  - Compliance levels
- That **IF MET** would mean the release could occur.

# Levels of Criteria

Activity	Criteria	Example
Basic Team Work Products	Done'ness criteria	Pairing or pair inspections of code prior to check-in; or development, execution and passing of unit tests.
User Story or Theme Level	Acceptance Tests	Development of FitNesse based acceptance tests with the customer AND their successful execution and passing. Developed toward individual stories and/or themes for sets of stories.
Sprint or Iteration Level	Done'ness criteria	Defining a Sprint Goal that clarifies the feature development and all external dependencies associated with a sprint.
Release Level	Release criteria	Defining a broad set of conditions (artifacts, testing activities or coverage levels, results/metrics, collaboration with other groups, meeting compliance levels, etc.) that IF MET would mean the release could occur.

Contributed to Chapter 20 of Lisa Crispin & Janet Gregory's new 2009 *Agile Testing* book

# Who Decides on Quality?

- Of Course, Quality isn't a simple pattern, it's a façade
- Jim Coplien responding to a point on Scrum Alliance leadership list (*paraphrased*)...

*Value doesn't matter when examining technical debt. Rather, that cleaning up after yourself transcended the normal determination of business value and was simply an inherent part of delivering software. That it is our responsibility and is non-negotiable. The decision-making wasn't FOR the business-side, but instead resides within the team.*

- Listen to your team!
- Ask the 'Right' questions!

# Technical Debt

- Intentional vs.
  - We don't have the time, you're simply gold-plating, I don't trust you or your overreacting
- Unintentional
  - Mistakes, M&A activity, poor design choices, skill gaps, etc.
- Warning signs & terms
  - Hacking, Crufty Code, Over complexity
  - Fear Factor (FUD), Estimate Inflation





# Role of the Product Owner At Scale

# Product Owners

## Their Evolving Role

- Within each Scrum, the PO is typically narrowly focused on crafting the Backlog, engaging in progress, and reviewing sprint results
- However, as Scrum scales, the PO team needs to become more focused on:
  - **Product Line Evolution** – Meta Backlog and coordination across Sprint teams, strategy development & execution, resource load-balancing, and budgeting
  - **Cross-Team Planning** – SoS coordination, linked goals and backlog work, delivery integration, and staged (forward-thinking) planning
  - **Delivery Dynamics** – timing, marketing, packaging, interrelationships, customer feedback, and achieving production quality

All of course with the team(s) delivering the product

# Chartering

## Chartering Components

- ❑ Project visioning
- ❑ Shared stakeholder expectations
- ❑ Goals & Success criteria
- ❑ Approaches, Process, Methods
- ❑ Team, Communication, Metrics
- ❑ Scope & Budget
- ❑ Risk handling
- ❑ Sign-off

## Agile Practices

- ❑ Road-mapping
- ❑ Backlog grooming, Story writing
- ❑ Collaborative release planning
- ❑ Crystal – Blitz Planning
- ❑ Patton – Story-Mapping
- ❑ XP – Planning Game

# Product Management

- Facilitating high level visioning
  - Competitive landscape
  - Technology and corporate direction
- Chartering of new projects
- Product Road-maps & release orchestration
- Story development
  - Epic -> Feature (MMF) -> Story stream
  - Priority & value
  - Technical clarity (quality, architecture, technical debt)



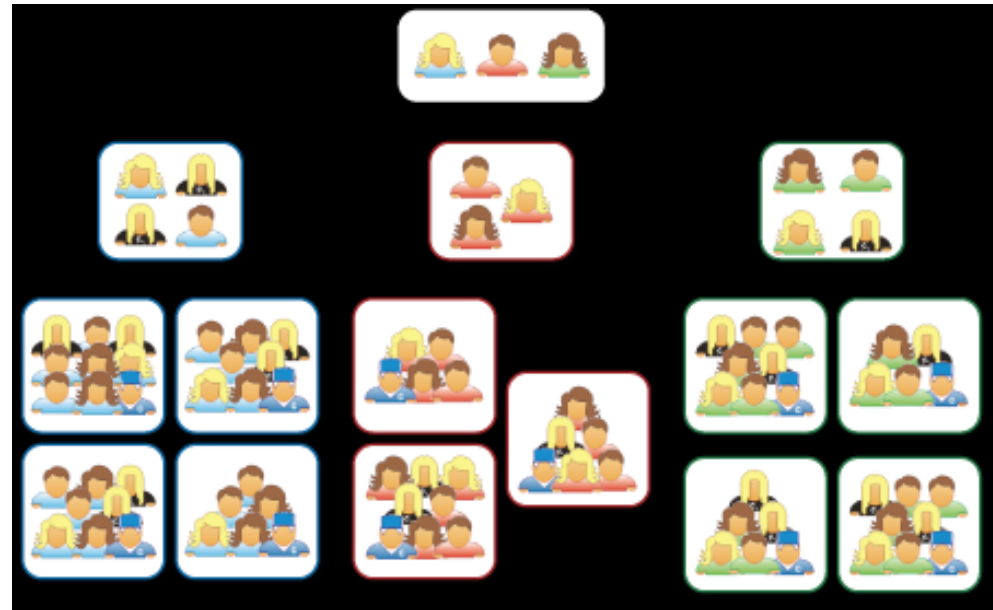
# Portfolio Management

- Something similar to the Lean Dog – Big Visible Room for executives to
  - <http://www.slideshare.net/LeanDog/agile-from-the-top-down>
- Instead of an electronic dashboard,
  - Vision, Portfolio, Assignments, Value, ROI
  - Release plans, application funnel
  - Capability and competency
  - Agile practices / team alignment



# Scrum of Scrums

- Periodic meeting – similar to daily stand-up
- Focus:
  - X-team interactions, dependencies, and blocks
  - Release planning & communication
  - Frequency dictated by state
  - Information radiators: release burndown, impediments, etc.
- Key reference
  - <http://www.scrumalliance.org/articles/46-advice-on-conducting-the-scrum-of-scrums-meeting>



# Scrum of Scrums board

## Story + Status (across teams)



Source: <http://www.xqa.com.ar/visualmanagement/2009/08/scrum-of-scrums-making-it-visual/>

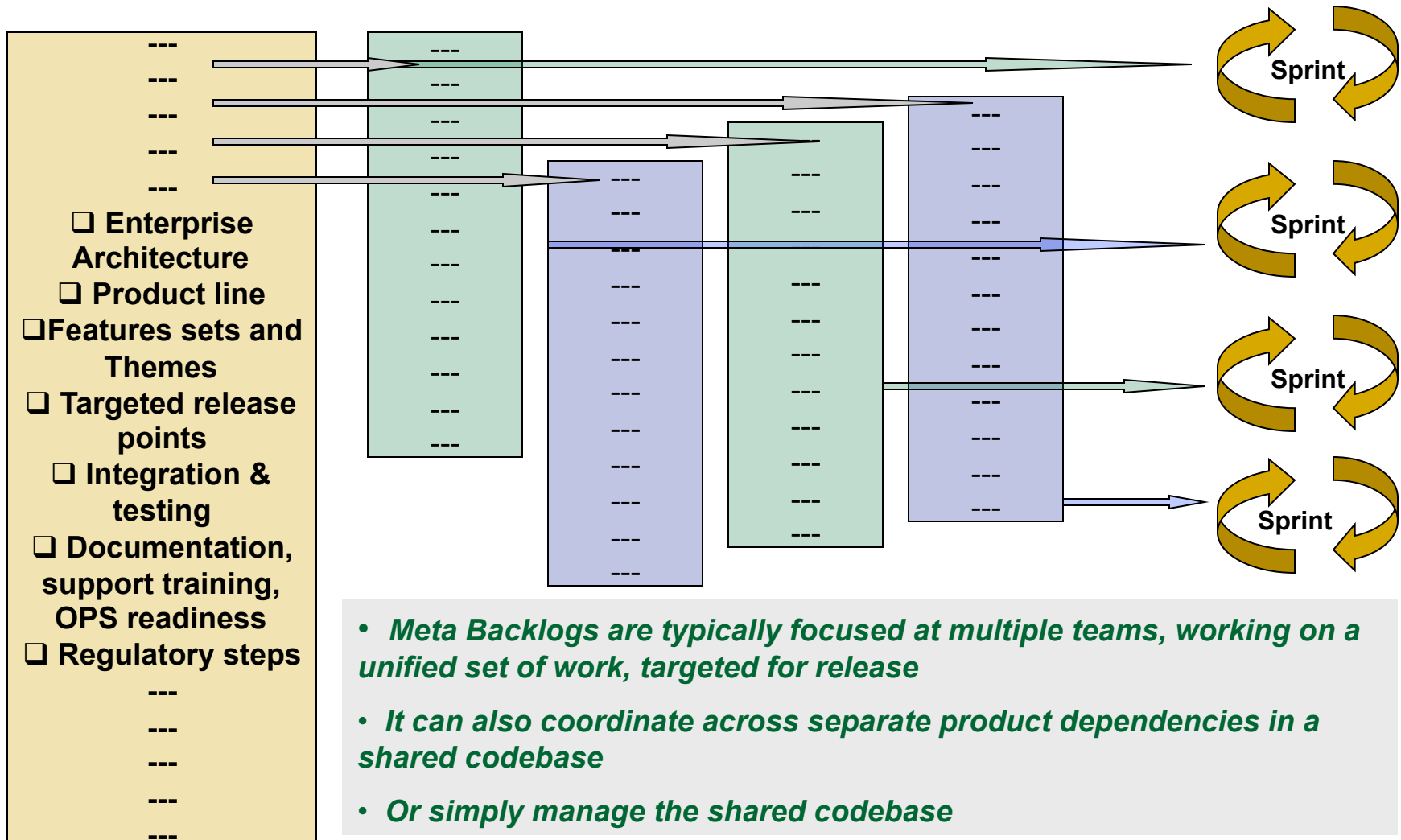
# Product Owners

## Guiding Testing

- The PO also needs to have a Quality & Testing perspective that within each Sprint focuses on –
  - Developing specific multifaceted quality release goals
  - Working with the team (Testers) to develop acceptance tests
  - Guiding daily convergence towards goals
- Across the Scrum of Scrums:
  - Developing Product & Portfolio quality meta goals that cross all Sprints
  - Integrating deliverables and qualifying the overall Product via planned Integration & Stabilization Sprints
  - Balancing automation vs. manual testing capacity and investment
  - Focusing teams towards Product release points



# Meta-Backlog – to – Backlog Translation



# Release Train Management



- Iterative model with a release target
  - Product centric
  - Focused on a production push/release
- Synchronized Sprints across teams
  - Some teams are un-synchronized, but leads to less efficient cross-team (product) interactions
- Continuous Integration is the glue
  - Including automated unit and feature tests; partial regression
- Notion of a *“Hardening Sprint”*
  - Focused more on Integration & Regression testing
  - Assumption that it’s mostly automated
  - Environment promotion
- Define a final Hardening Sprint where the product is readied for release
  - Documentation, Support, Compliance, UAT, Training

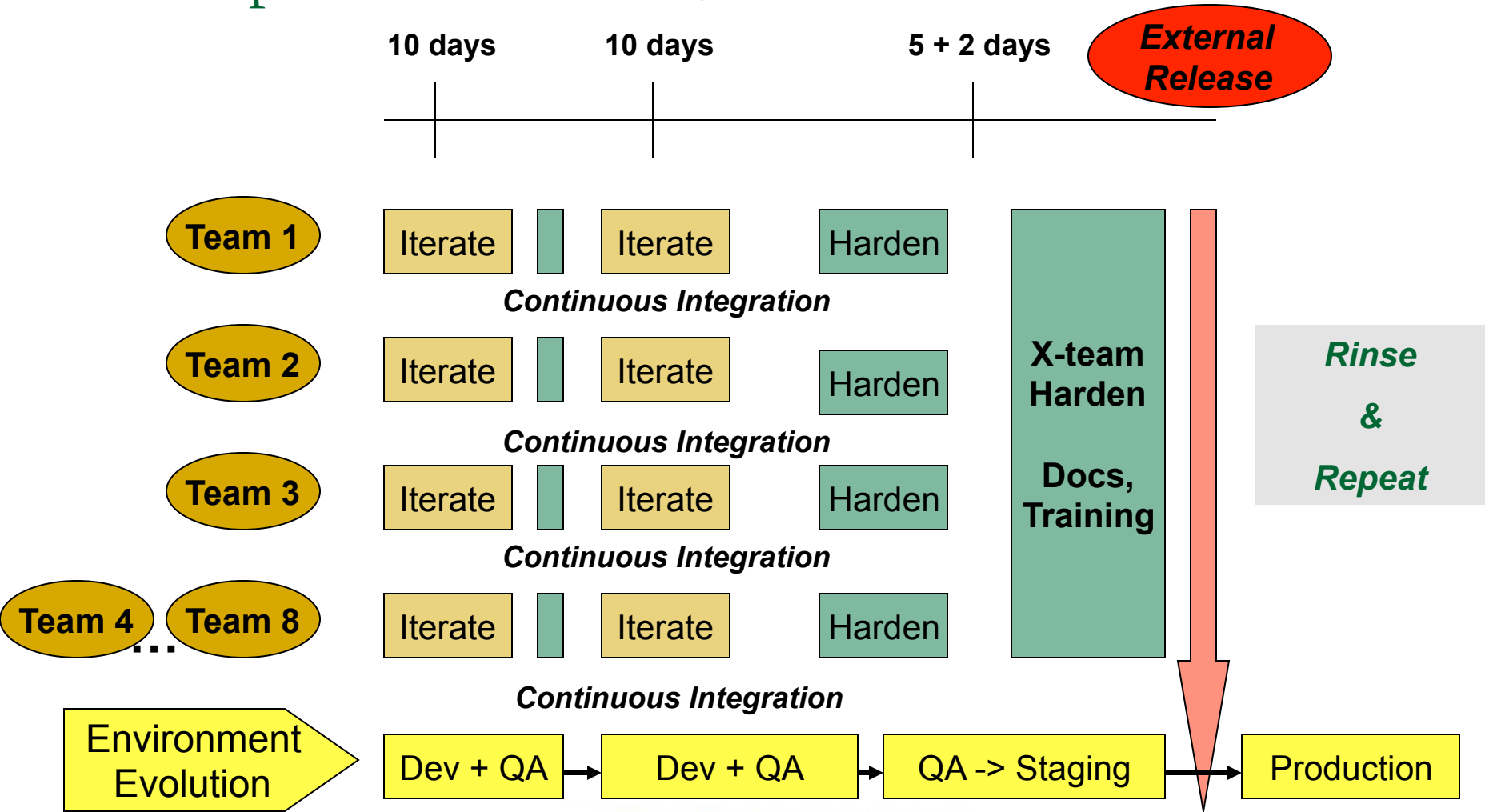
# Release Train Management “Internal” Driving Forces



- Customer’s ability to “accept” the release
- Value being delivered in the release – purely scope
- Hardening Sprint “reality”
  - Time, Complexity, Automation, Size, Compliance, and Industry
- Internal team readiness
  - Customer support
  - Sales & Marketing readiness
  - Overall documentation & training

# The Agile Release Train

## Example: eCommerce / SaaS Model



# In Closing...

- Product Ownership (by the Customer, Stakeholder, BA, Product Manager) is the most crucial role within agile teams, providing—
  - ✓ Inspirational vision
  - ✓ Clear goal – setting; quality balanced
  - ✓ Prioritized requirements – value based, workflow
  - ✓ Measured & accepted results
  - ✓ Scaled appropriately
  - ✓ Focus towards the team first

# Wrap-up

- What were the most compelling ideas, stories, or lessons?
- What adjustments will you make in your Product Ownership?
- What ideas did I miss?
- Final questions or discussion?



## Thank you!



# Contact Info

## Bob Galen

Principal Consultant,  
RGalen Consulting Group, L.L.C.

*Experience-driven agile focused training,  
coaching & consulting*

Contact: (919) 272-0719

[bob@rgalen.com](mailto:bob@rgalen.com)

[www.rgalen.com](http://www.rgalen.com)

## Blogs

Project Times - <http://www.projecttimes.com/robert-galen/>

BA Times - <http://www.batimes.com/robert-galen/>

Podcast on all things 'agile' - <http://www.meta-cast.com/>

