

Scaled Professional Scrum

focused, effective, viable

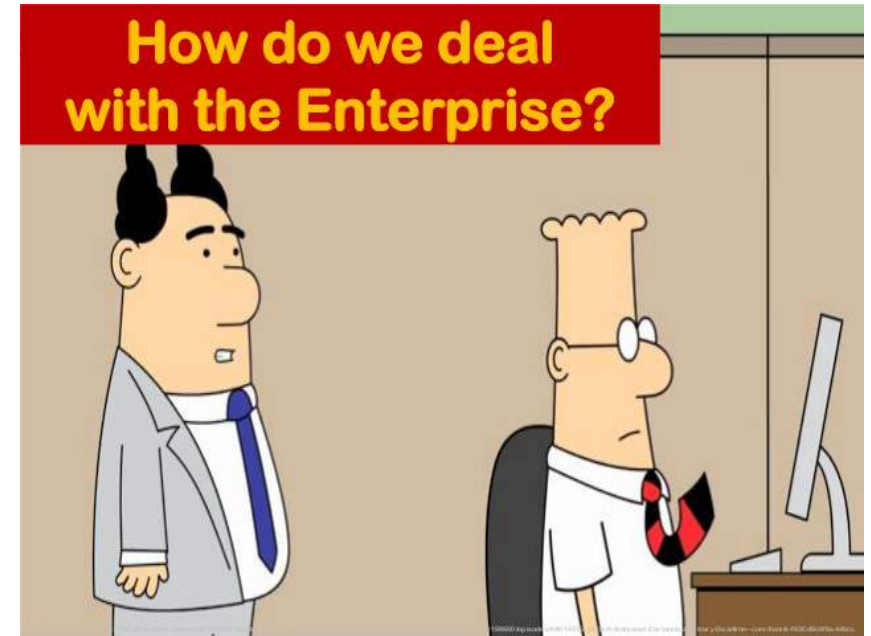
by Scrum.org – Improving the Profession of Software Development

A man who carries a cat by the tail learns something he can learn in no other way.

- Mark Twain

Organizations want to scale Scrum

- Great results with small initiatives of one to three teams
- Desire to gain similar benefits from larger initiatives
- Looking for a straightforward approach that builds on their current knowledge and skills



“The secret of getting ahead is getting started. The secret of getting started is breaking your complex overwhelming tasks into small manageable tasks, and starting on the first one.”

— Mark Twain



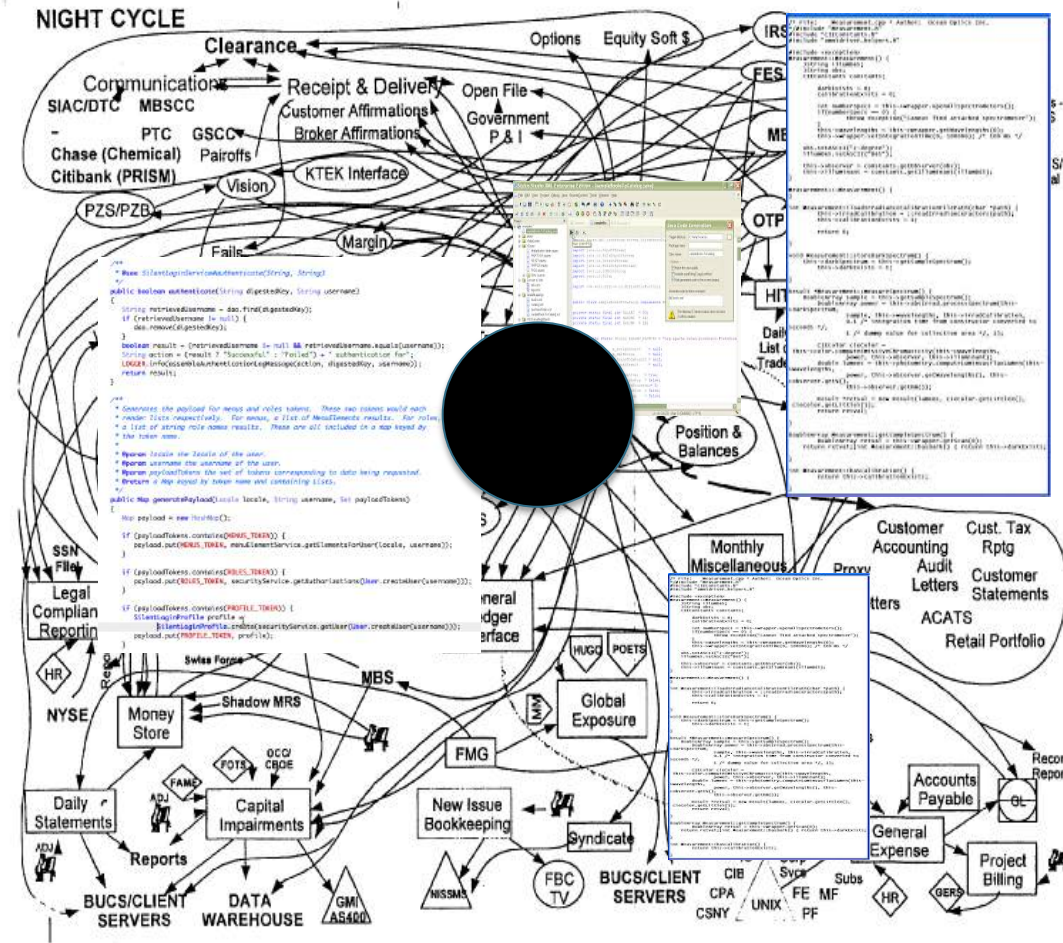
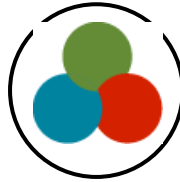
Thinking

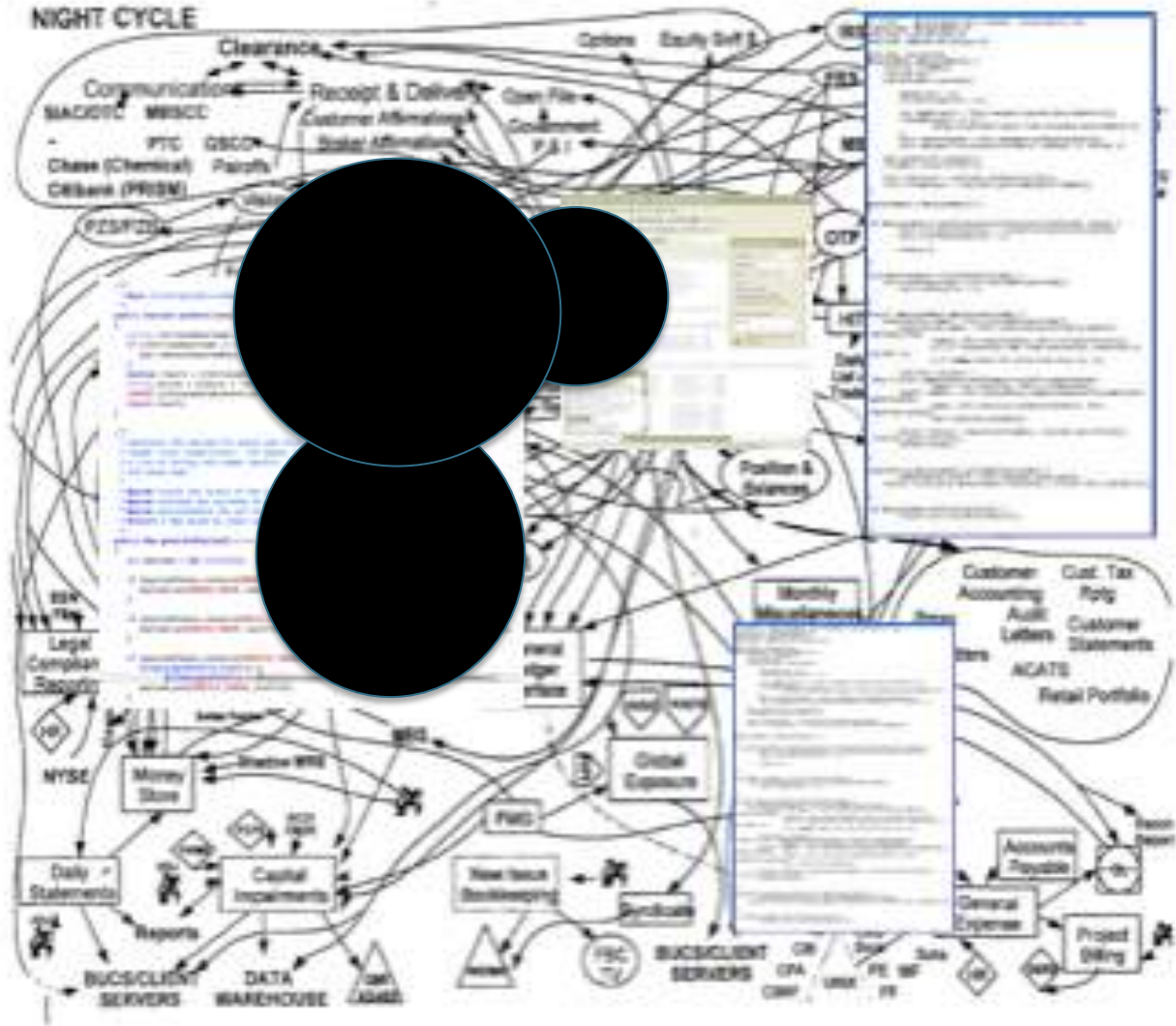
Behind our approach to scaling Scrum development

One Scrum team doing work



Product Backlog

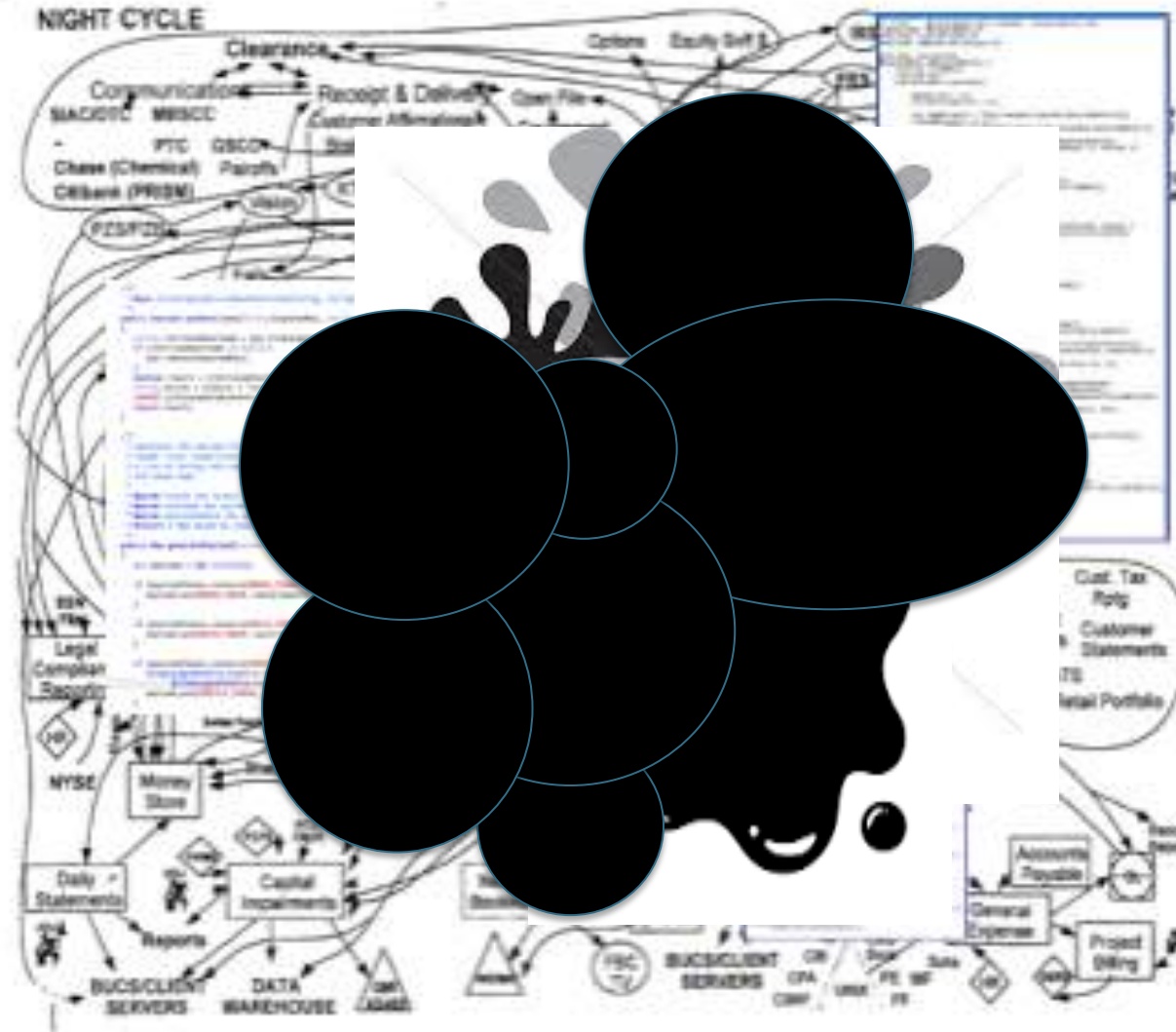
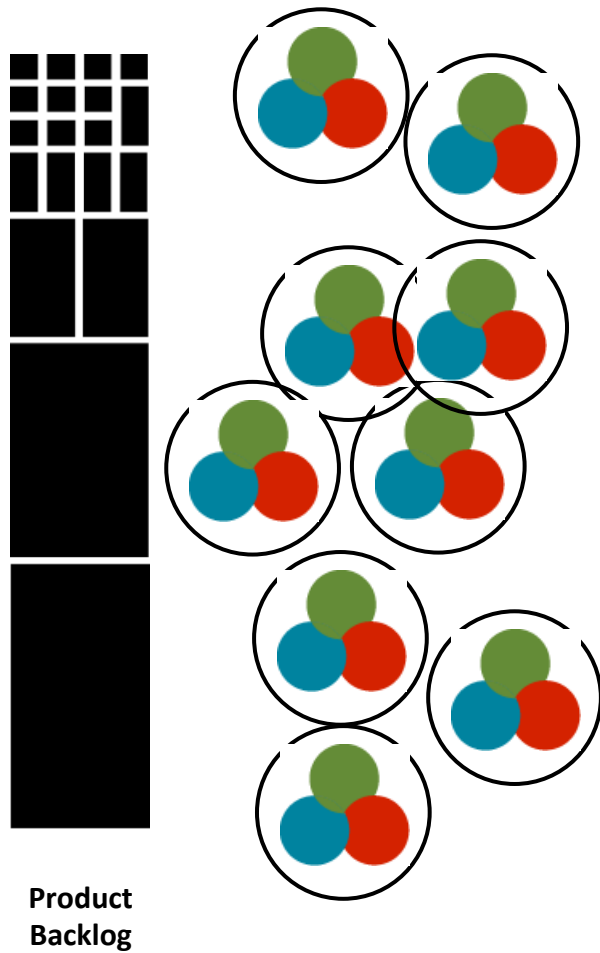




Types of Dependencies

- People
 - Domain
 - Technology
 - Software
-
- Internal vs. External

Nine Scrum teams doing work



Essence of Scaling

Identification and Resolution of Dependencies

Proactive to attempt to identify and work around dependencies

- Ongoing
- Persistent
- In all dimensions
- Prior to work occurring

Reification*

Reactive to ensure that undetected dependencies aren't corrupting the outcomes

- Frequent integration
- Acceptance testing
- Continual build and delivery
- Minimize technical debt

* **Reification:** making something real, bringing something into being, or making something concrete.

You can scale as long as you continuously...

- Identify and remove dependencies
- Integrate work across all level
- Create and inspect integrated increments regularly
- Provide adequate tooling and skills
- Inspect and adapt frequently

Continuous
integration,
build, test and
delivery test
adequacy of
effort

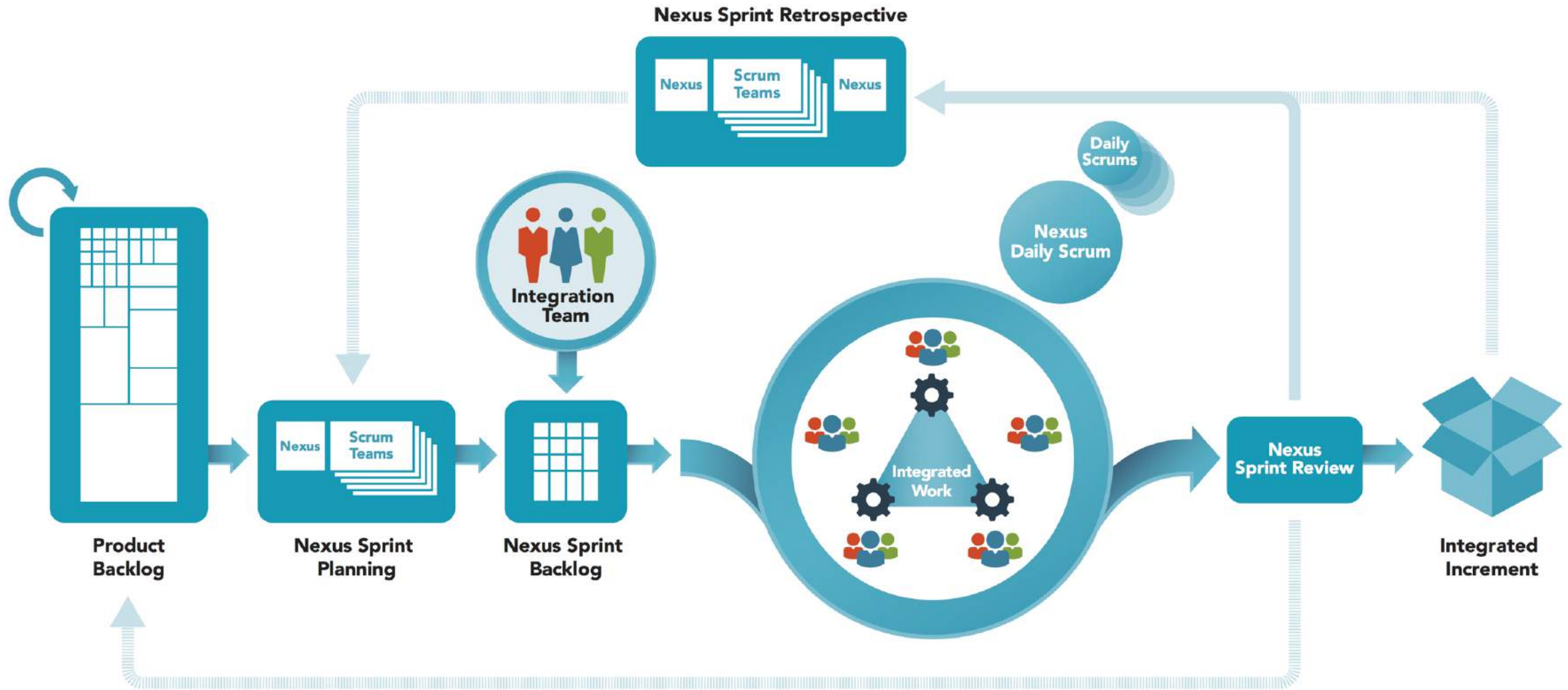
“It takes two to tango.”
– Al Hoffman

Nexus , an Introduction

Scaled Professional Scrum



The Nexus™ Framework – an Exoskeleton



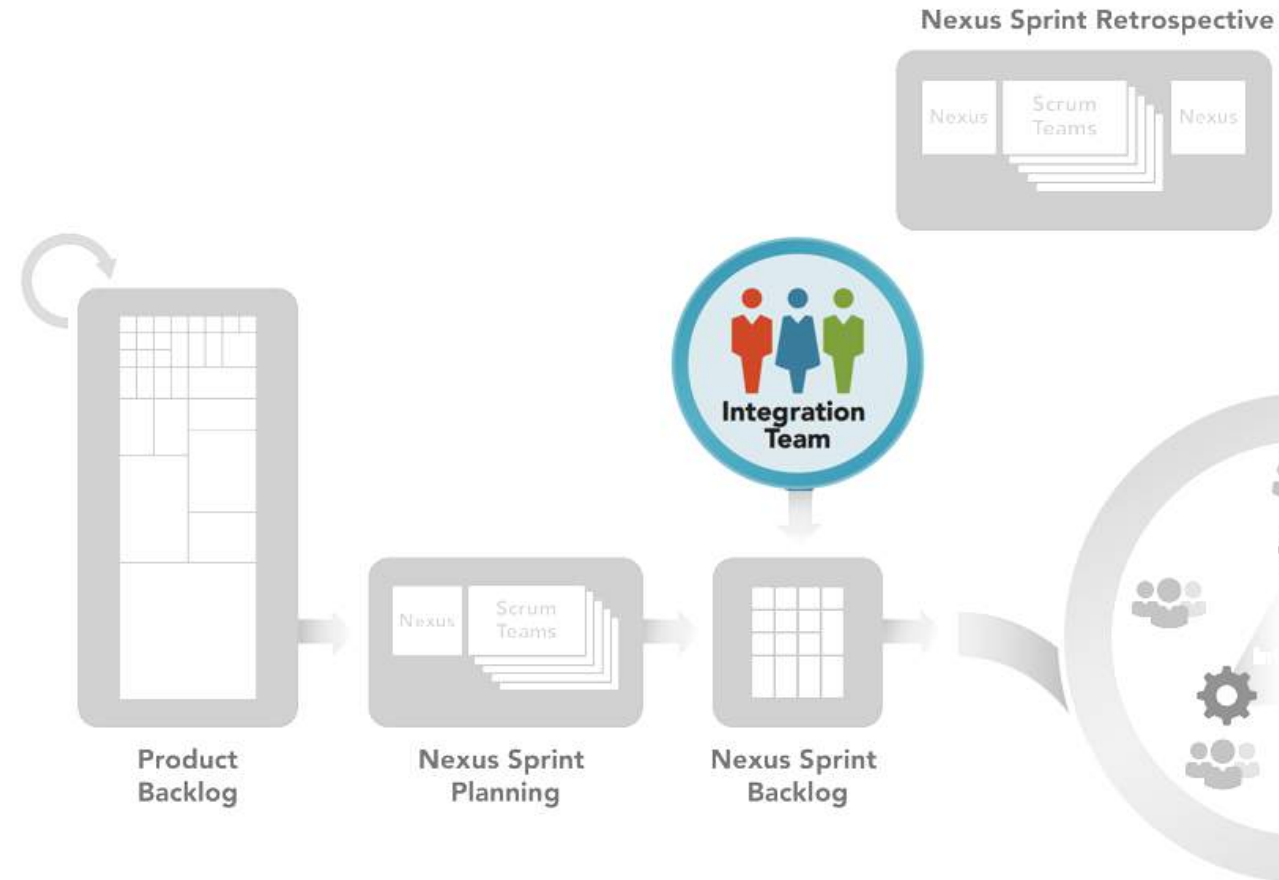
Scaling Solution in Nexus™, the Scrum Exoskeleton

- 40 Practices
- New Role
 - Nexus Integration Team
- New Events
 - Nexus Sprint Planning
 - Nexus Daily Scrum
 - Nexus Sprint Review
 - Nexus Sprint Retrospective
- New Artifacts
 - Nexus Sprint Backlog
 - Integrated Increment



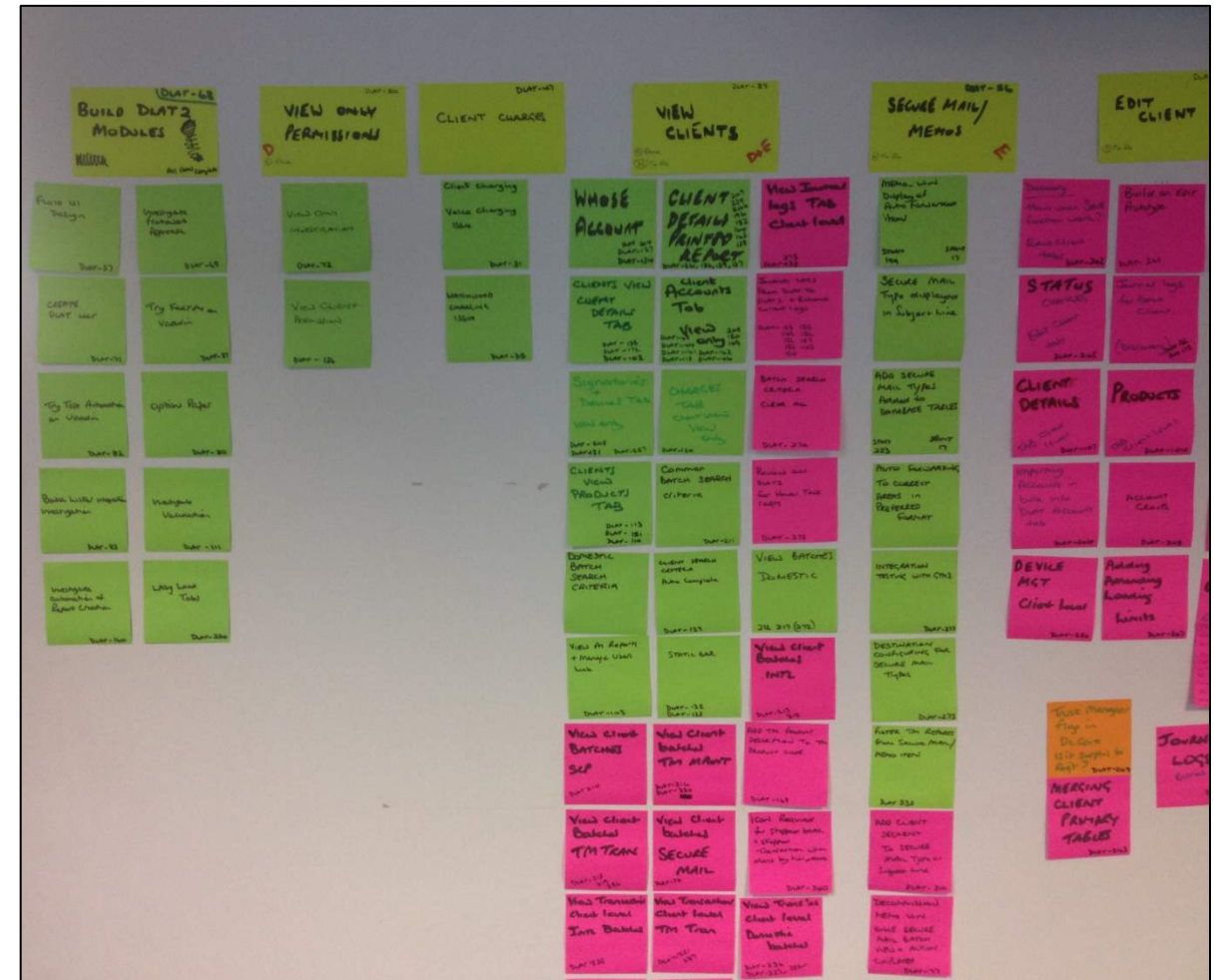
The Nexus Integration Team

- The Nexus Integration Team is a Scrum Team
 - All the events, roles and artifacts
- Members may be full or part time
 - **Membership in the Nexus Integration Team takes precedence**
- Composition may change between Sprints

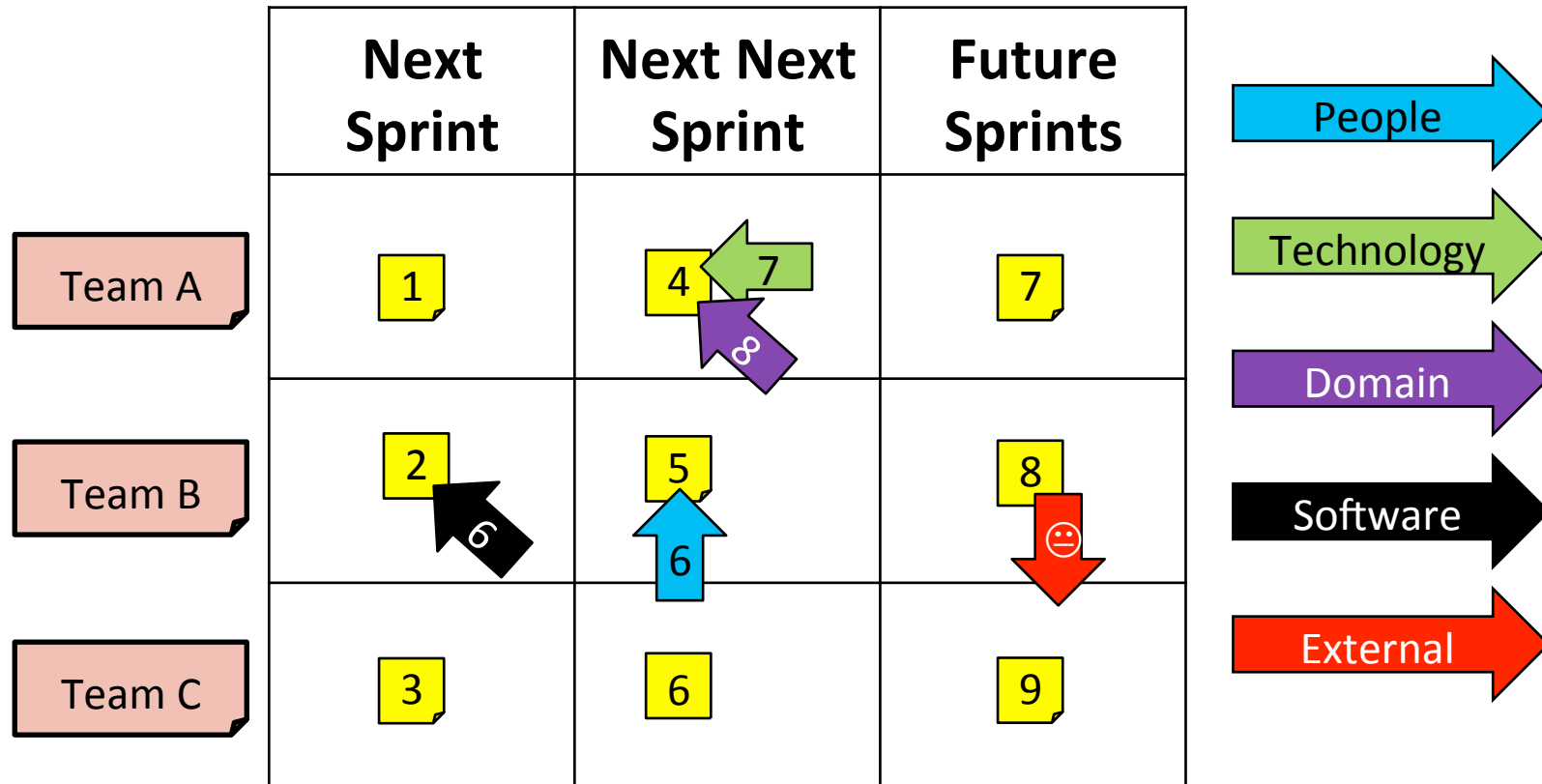


Story Mapping

- Product Backlogs are one-dimensional
 - This makes it difficult to see the workflow, relationships, and dependencies
- Story maps help visualize and plan upcoming work
 - They foster collaboration by providing different perspective



Cross-Team Refinement



Communities of Practice (CoP)

- Cut across organizational structure
- Focused on improving specialized skills
- Typically self-emergent with volunteer membership
- Leadership emerges and is situational
- Needs organizational support



Formally nurturing CoPs helps build learning organizations and increases capabilities and relevance in fast-changing environments.

“Put all your eggs in one basket and then watch that basket.”

— Mark Twain

Nexus+

Scaled Professional Scrum



Number of teams in a Nexus

YES

9 +



- System code, design, architecture, and tests
- System and domain knowledge
- Technical skills and underlying automation
- Scrum and agile practices
- Coherence and independence of work

NO

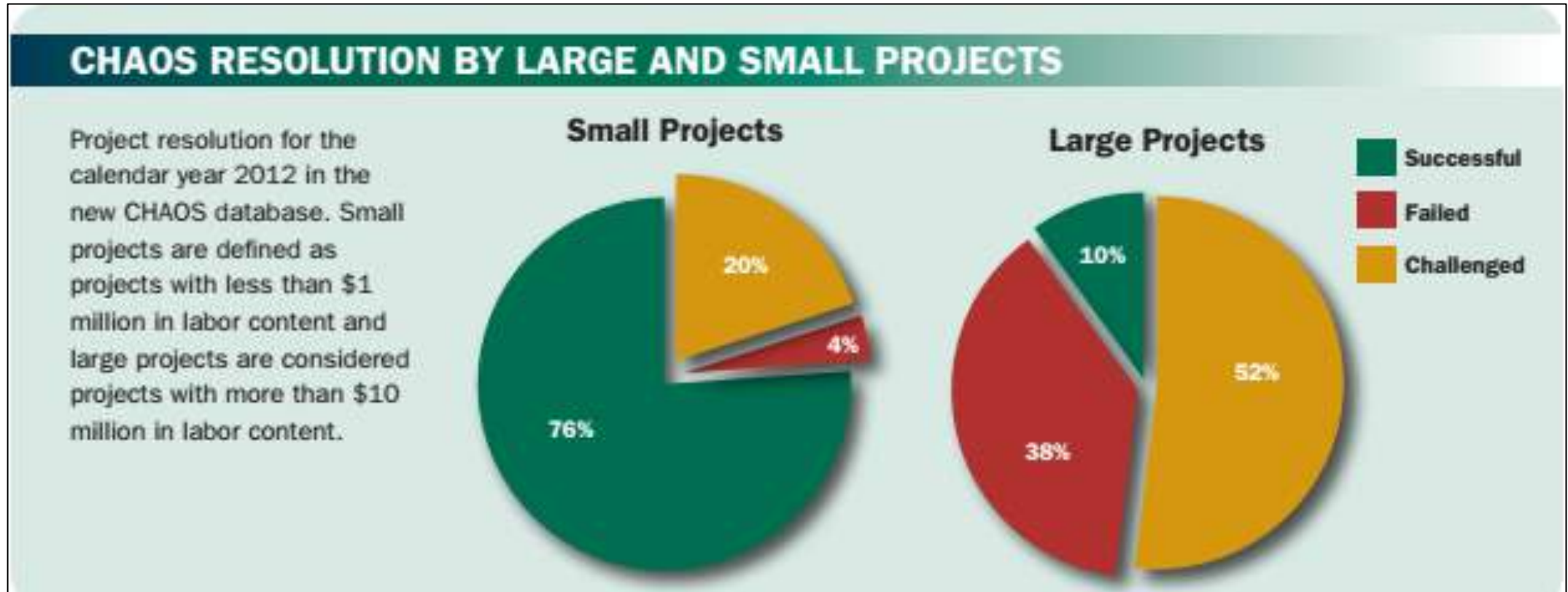
9 -



The Challenge of Large Scale Development

- We have found that the Nexus starts to fray and create confusion at around 9 teams
 - We have experienced this across a range of companies and industries
- Dependencies and integration issues are magnified and can create chaos
- Then engineering solutions are necessary, necessitating enabling, integrating architectures
- **There is no guaranteed recipe at this scale – every project is unique**

Success at Large Scale

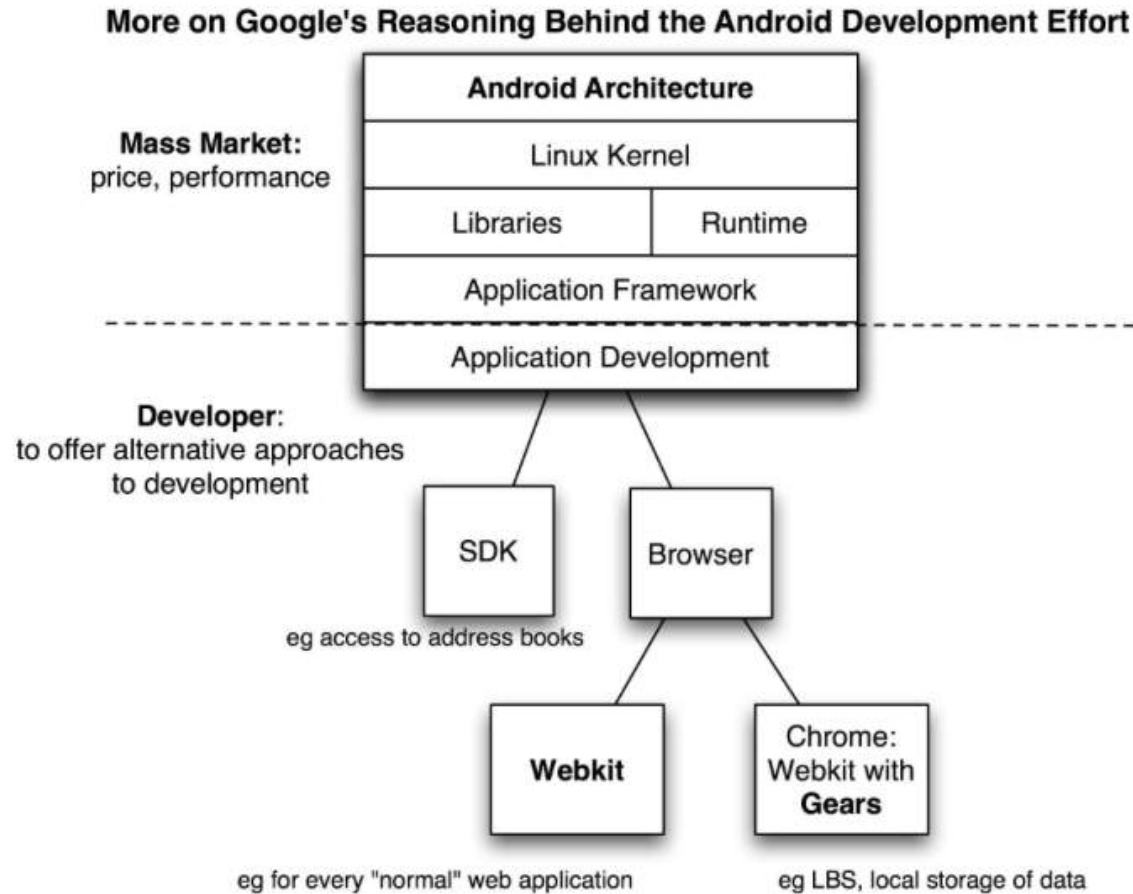


Source: 2013 CHAOS Manifesto Report, The Standish Group International

Scaling Options – Horizontal, an Architecture of Equals

- Nexus(s) integrate into horizontal platform
- Platform provides stability through integration standards and facilities
- Additional Nexus manages and develops platform
- Tightly integrated Nexuses
 - IOS/SDK
- Medium integration Nexuses
 - Microsoft COM
 - Product family architecture
 - API
- Loosely integrated Nexuses
 - Workflow
 - UI Platform

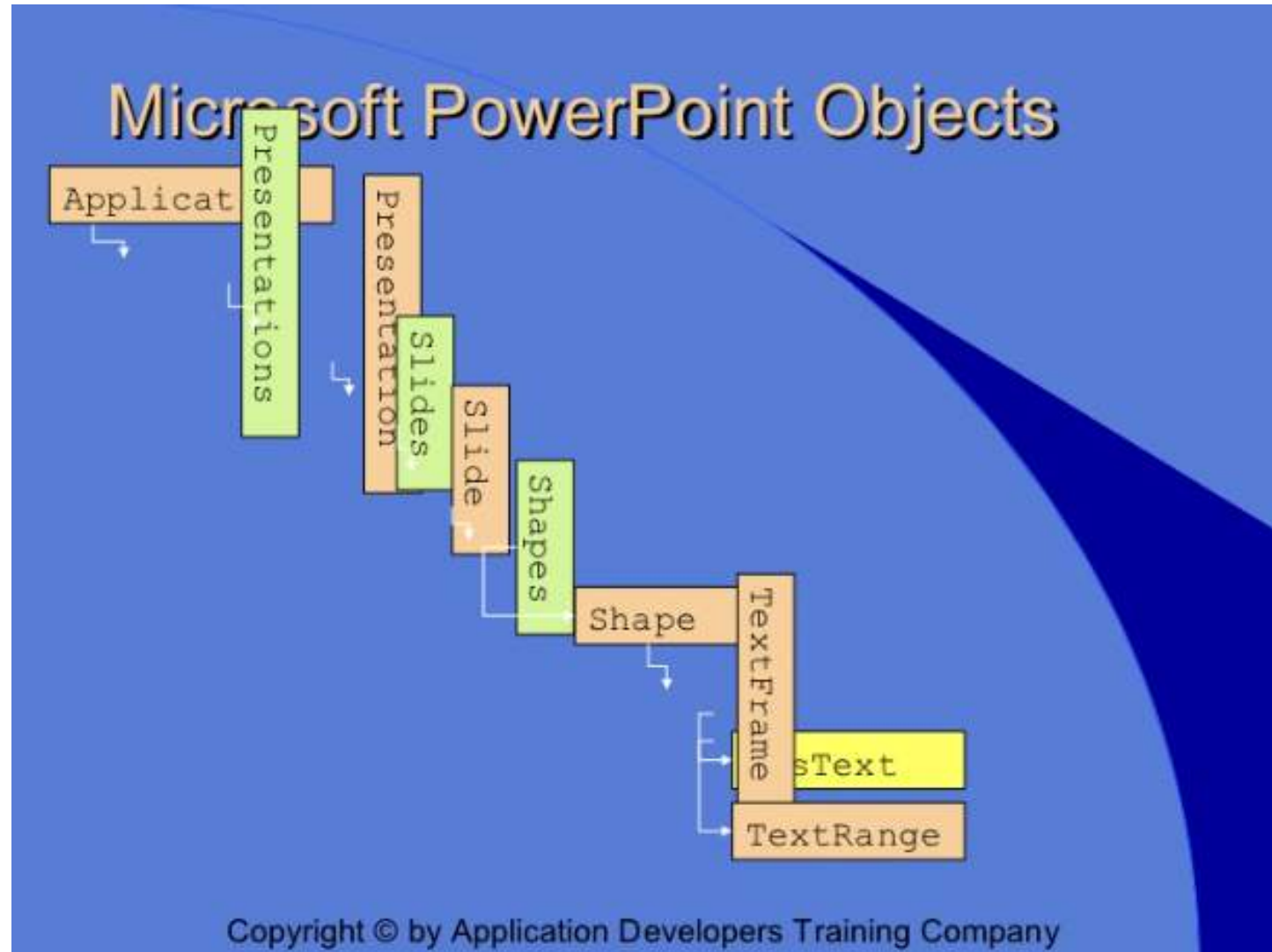
Build your own IOS and SDK



Provide the IOS and the Apps will come:

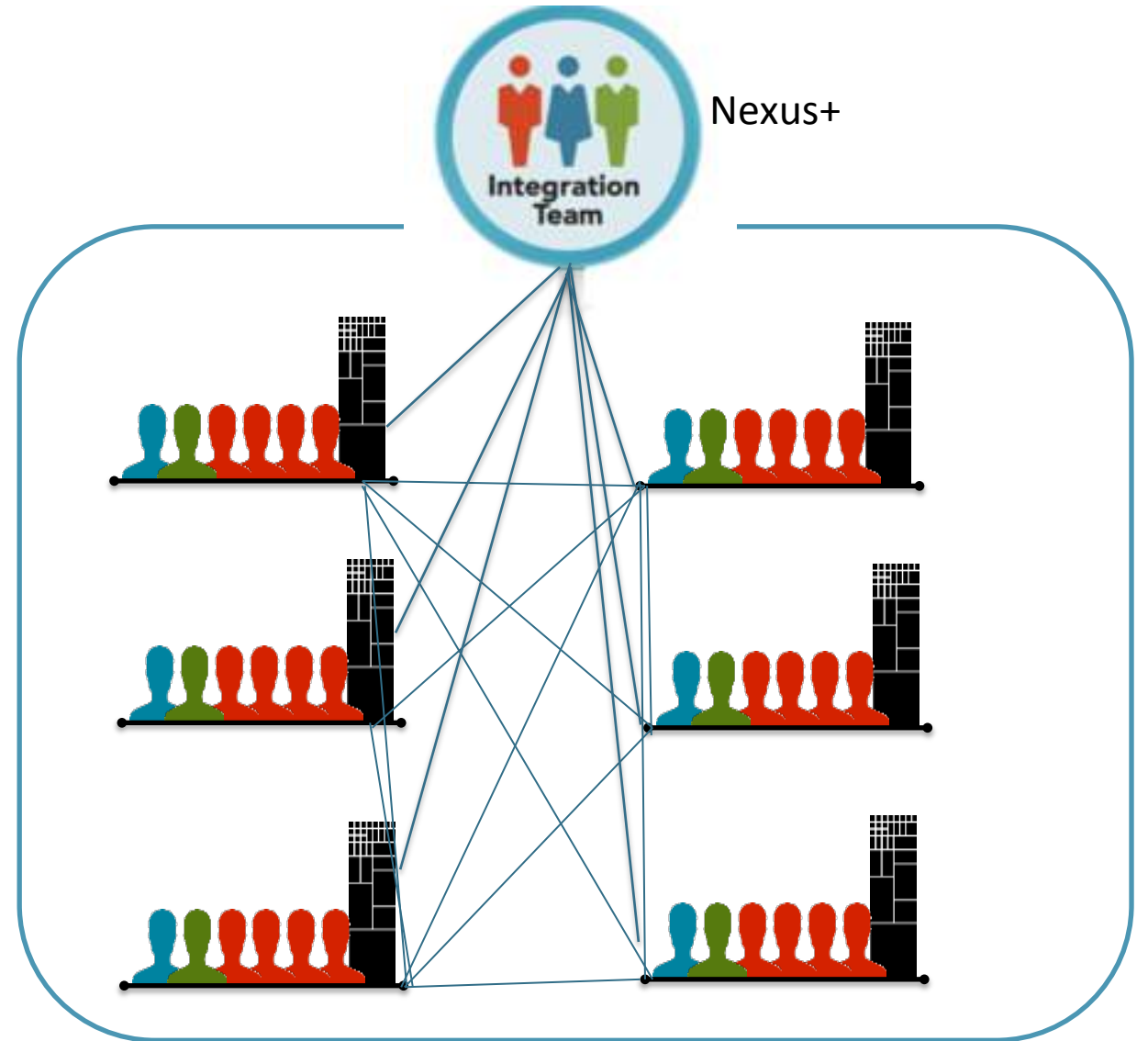
- Shell Oil
- Apple
- Google
- Tesla

Microsoft Component Object Model



Scaling Options – ad-hoc into a light architecture

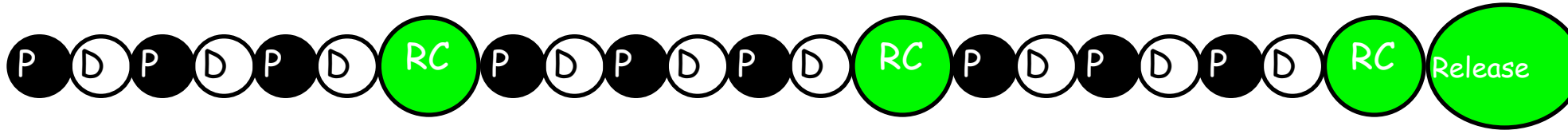
- Adding additional Nexuses to a product, each integrating their work into a single releasable Increment
- Often broken down into product area teams, value chain area teams, and other delineators that denote collaboration and unit of purpose
- Cross team / Nexus dependencies increase exponentially



Stabilization: Plan vs. Reality

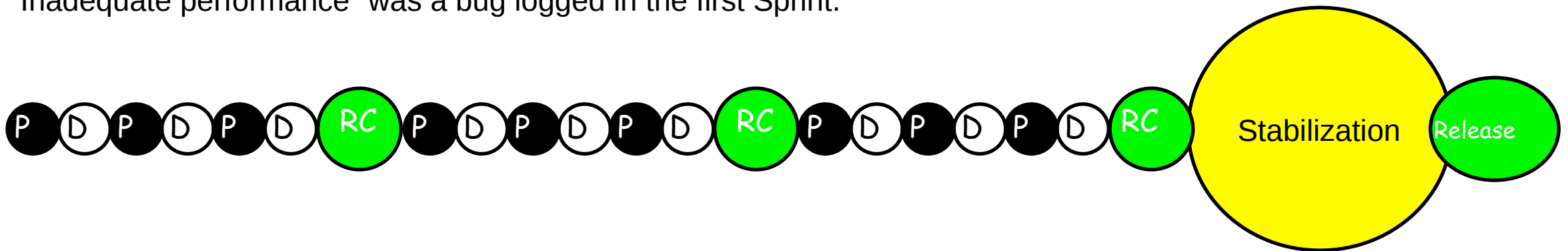
Plan

9 Sprints, 3 release candidates and then release.
800-person development organization.



Actual

The release candidates were presentation of partially working functionality from the code branch for each team. A five+ month stabilization was required prior to release.
“Inadequate performance” was a bug logged in the first Sprint.



Nexus+ Integration

You will need either:

- A full time integration team who work above the Nexus+ helping to coordinate across each Nexus
- An integration Nexus
- Architecture adequate to complexity

Google runs 4,000 builds and
60 million tests every day

“Success in management requires learning as fast as the world is changing.”

– Warren Bennis

Managing the Nexus

Scaled Professional Scrum



THE Questions of Scaling

Process

- What must be done to integrate the work?
- How frequently do you need the work integrated into shippable product?
- How do you measure and manage the work and the integration?

Cost

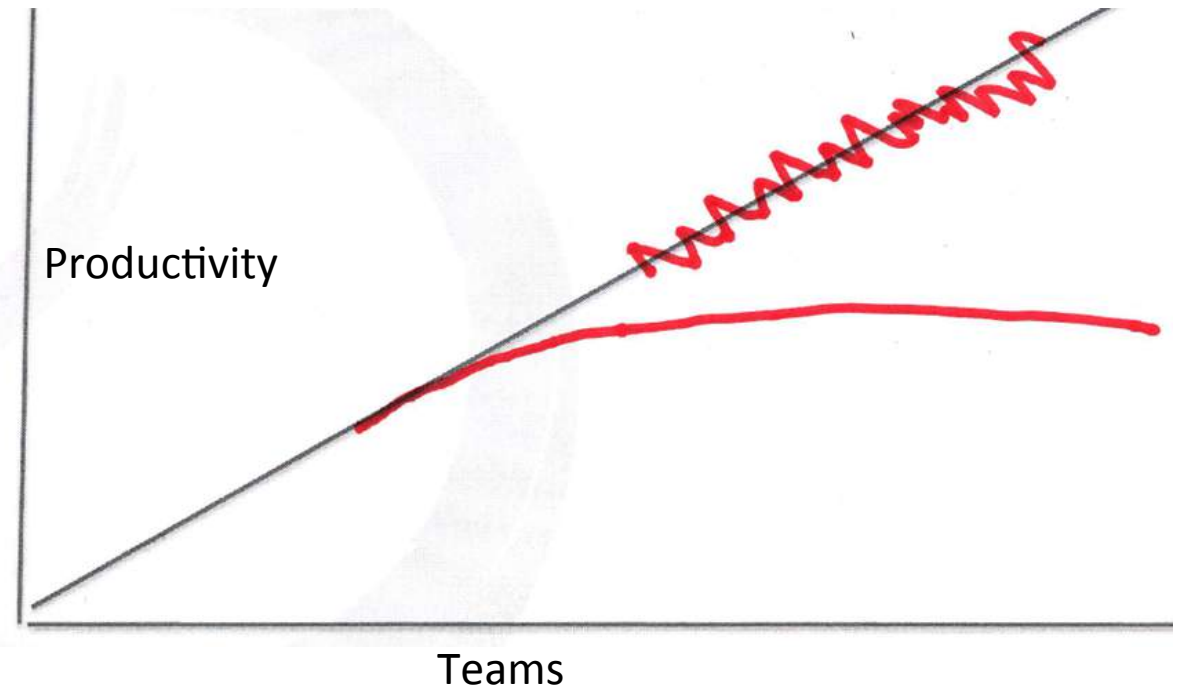
- What is the overhead of integration and delivery?
- Are you balancing cost and benefits of this overhead with value produced?
- Is the cost systematically being reduced?

20 teams total velocity per Sprint each team averages 10 PPS individually

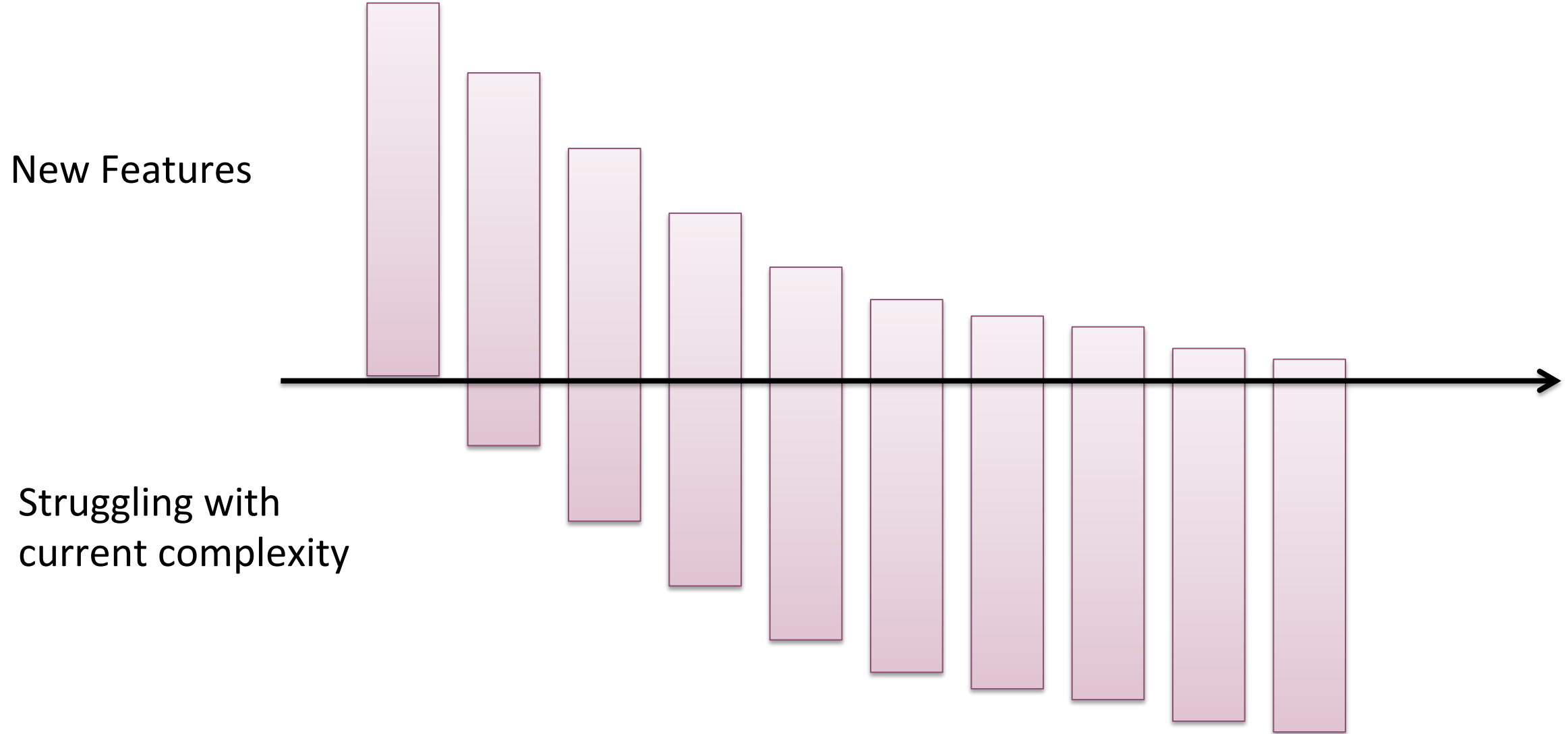
Velocity per Sprint	Description
200 PPS	Each team works independently of other teams, and increments create product with any framework
50 PPS	Management, recognizing complexity, assigns people to fixed feature and infrastructure teams
80 PPS	The people self-organize throughout the release to maximize coherence and minimize dependencies
160 PPS	Formal dependency removal and reification through use of Nexus process

Descaling

- Proceed with caution
- Add practices or tools to help
- Reduce the overall pace by reducing the number of teams to a more sustainable number (and/or velocity)
- Clean up and integrate the current software so it can be built upon in future Sprints

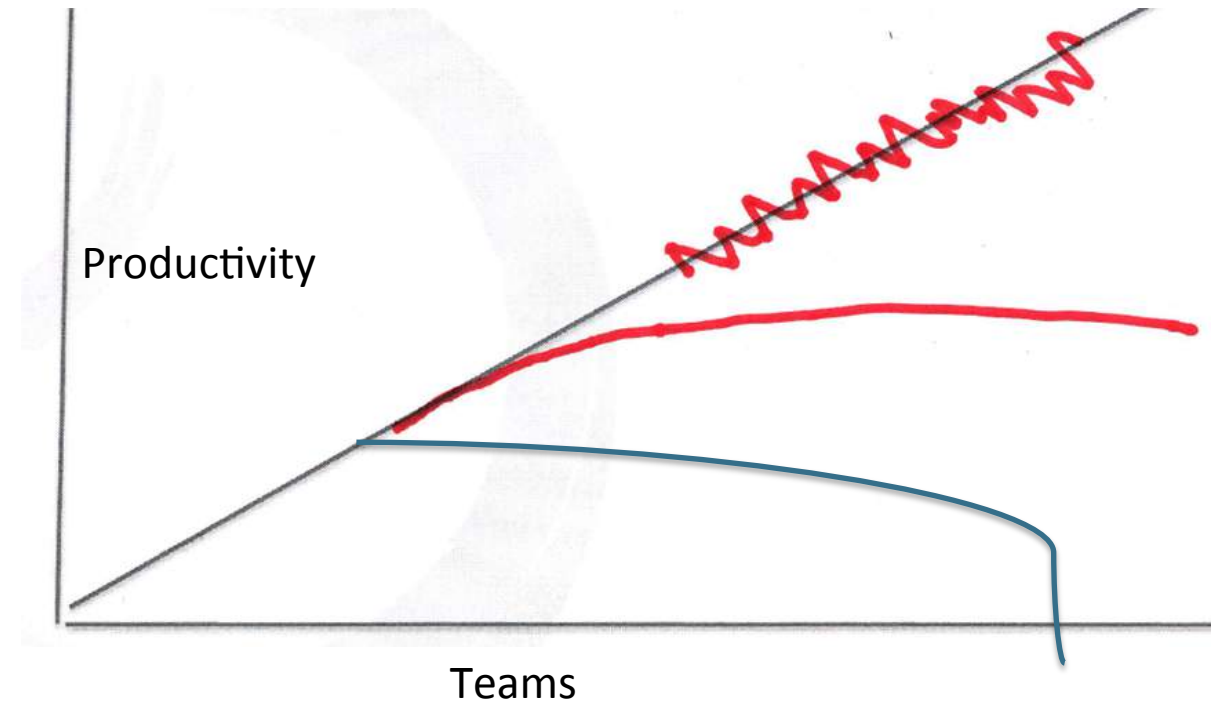


Dependency Persistence Causes Overwhelming Technical Debt



Scrumble

- When technical debt, domain knowledge, and test results overwhelm forward progress, Scrumble
- Scrumble is a period of unknown duration and staffing when work is done to allow forward progress to resume
- Staffing should be minimized and talent applied maximized



How well your scaling effort is going?



Agility Index



Scaled Professional Scrum

- SPS Workshops
- Nexus Guide
- Nexus Assessments
- Agility Index
- Agility Path

Thank You!

