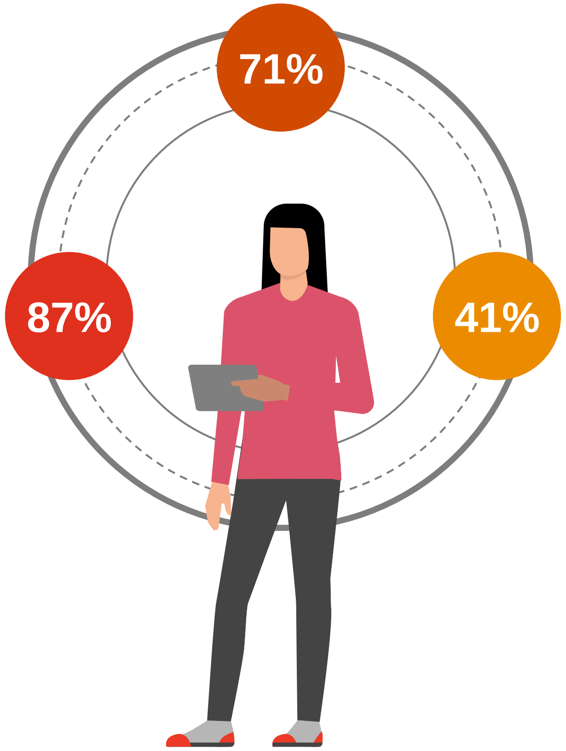
What is Agile? DevOps? How does it impact audit?

October 2019

Why is this important? Agile and DevOps methods are increasingly used by IT functions

Of respondents stated that they currently have a DevOps initiative in their organization or are planning one in the next 12 months.



Of surveyed organizations used some form of Agile practices in the past year

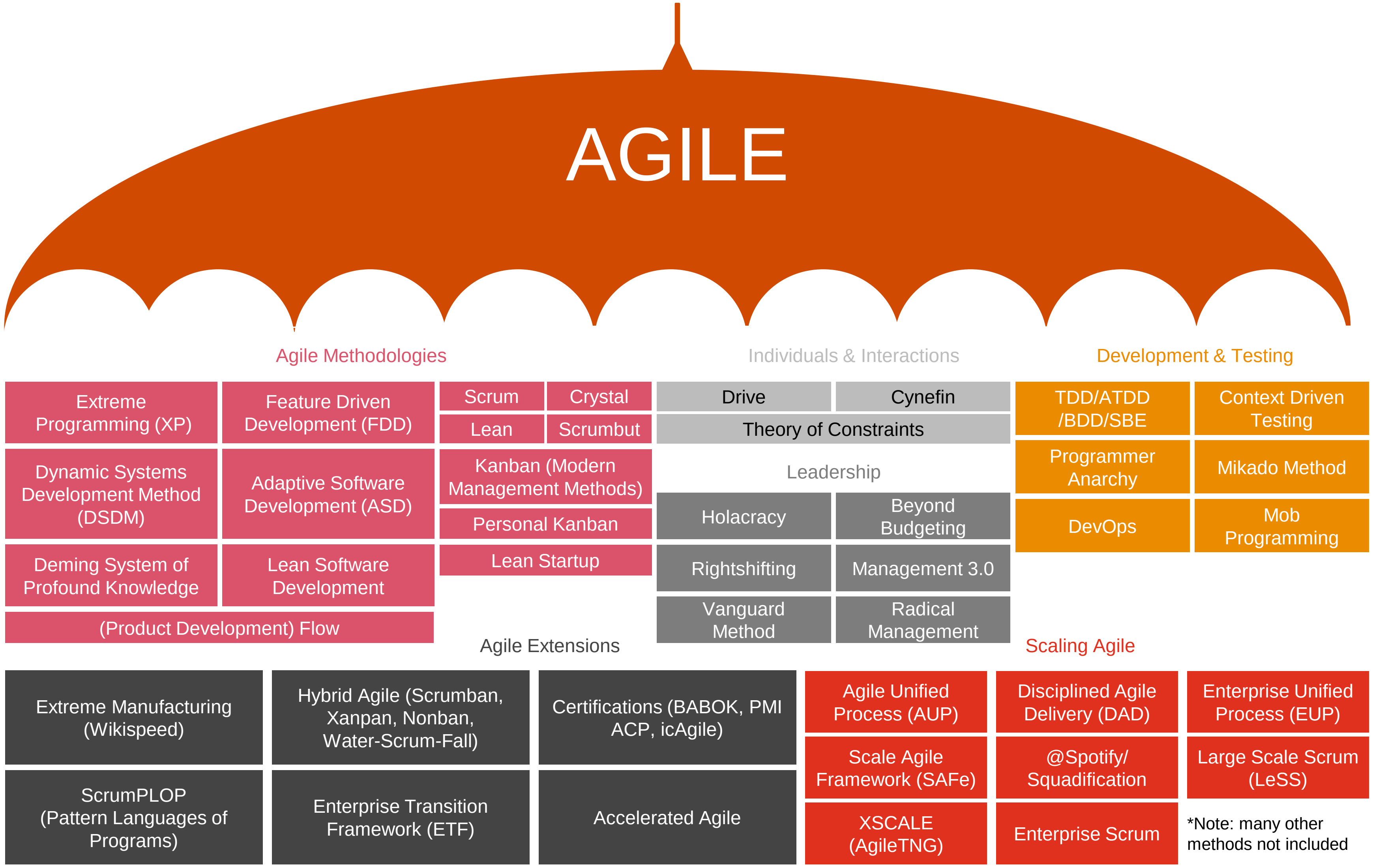
“Success in Disruptive Times,” PMI’s Pulse of the Profession, 2018

Of respondents cited reducing risk as a top reason for going DevOps, while only 17% cited ensuring compliance/governance

(Respondents could select multiple responses)

Source: Version One, 12th Annual State of Agile

Agile is an umbrella term that means many things..



AGILE

Agile Methodologies

- Extreme Programming (XP)
- Feature Driven Development (FDD)
- Scrum
- Crystal
- Lean
- Scrumbut
- Dynamic Systems Development Method (DSDM)
- Adaptive Software Development (ASD)
- Kanban (Modern Management Methods)
- Personal Kanban
- Deming System of Profound Knowledge
- Lean Software Development
- Lean Startup
- (Product Development) Flow

Individuals & Interactions

- Drive
- Cynefin
- Theory of Constraints

Leadership

- Holacracy
- Beyond Budgeting
- Rightshifting
- Management 3.0
- Vanguard Method
- Radical Management

Development & Testing

- TDD/ATDD /BDD/SBE
- Context Driven Testing
- Programmer Anarchy
- Mikado Method
- DevOps
- Mob Programming

Agile Extensions

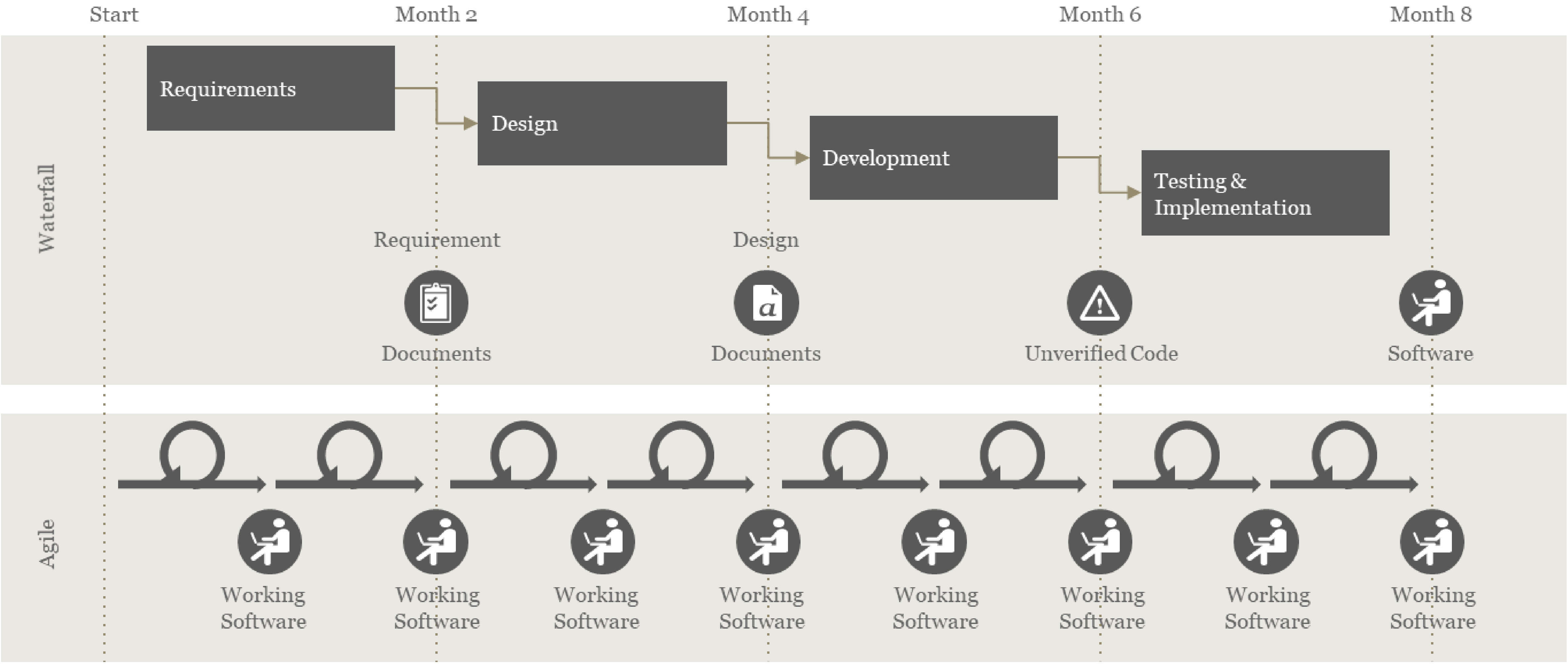
- Extreme Manufacturing (Wikispeed)
- Hybrid Agile (Scrumban, Xanpan, Nonban, Water-Scrum-Fall)
- Certifications (BABOK, PMI ACP, icAgile)
- ScrumPLOP (Pattern Languages of Programs)
- Enterprise Transition Framework (ETF)
- Accelerated Agile

Scaling Agile

- Agile Unified Process (AUP)
- Disciplined Agile Delivery (DAD)
- Enterprise Unified Process (EUP)
- Scale Agile Framework (SAFe)
- @Spotify/ Squadification
- Large Scale Scrum (LeSS)
- XSCALE (AgileTNG)
- Enterprise Scrum

*Note: many other methods not included

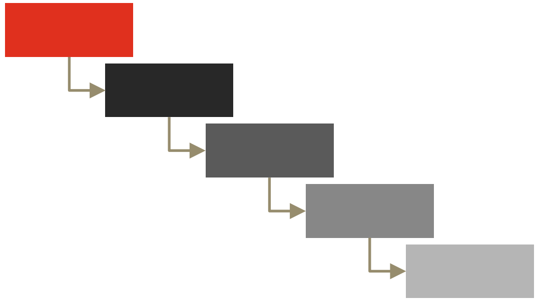
How does traditional compare with Agile?



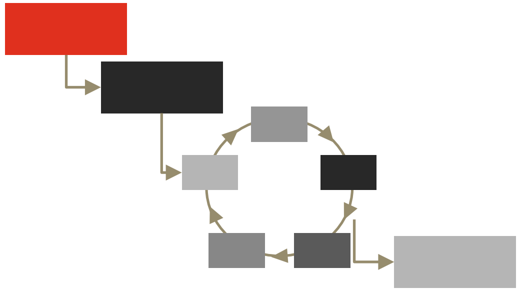
Agile could also mean this…

Deliberately overly simplified

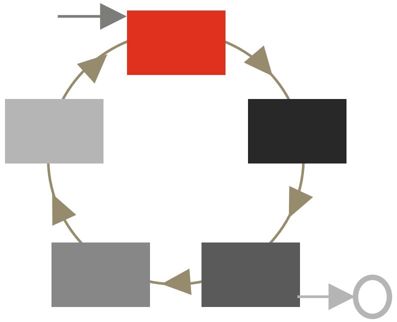
Waterfall



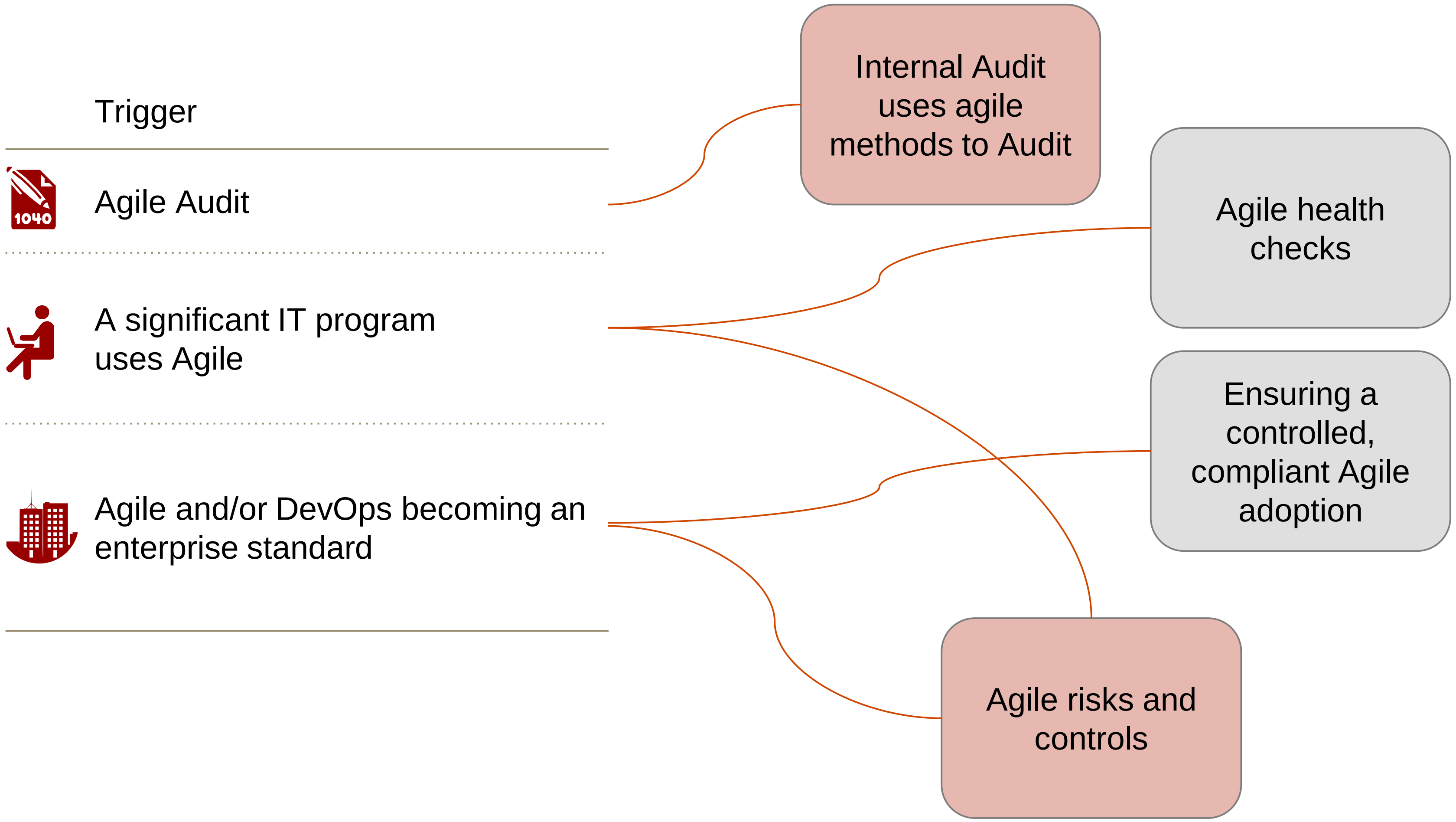
Hybrid



Agile



Impact to Audit?



Trigger

Agile Audit

A significant IT program uses Agile

Agile and/or DevOps becoming an enterprise standard

Internal Audit uses agile methods to Audit

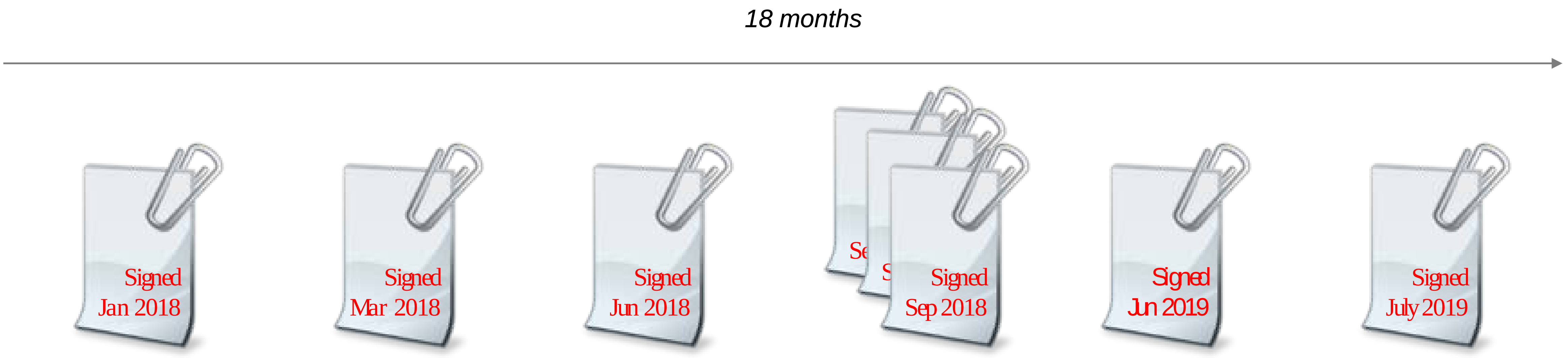
Agile health checks

Ensuring a controlled, compliant Agile adoption

Agile risks and controls

What does traditional waterfall “look like”?

Deliberately overly simplified



Project business case
(lengthy, detailed scope, requirements)

Requirements Specification
(lengthy, all requirements locked down, approved)

Design Document

Test Strategy, Plans, Requirements Traceability

Test Results

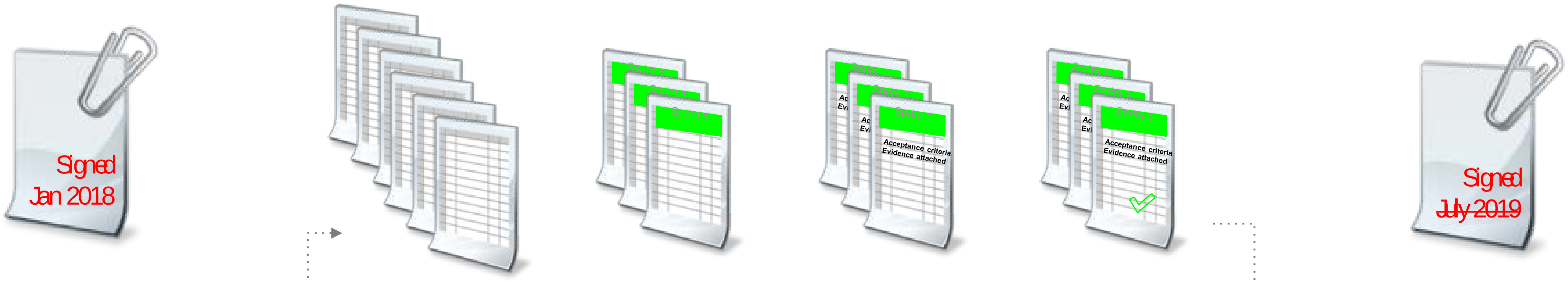
Production Change Approval

So, what does a common agile project “look like”?

Deliberately overly simplified

~~18 months~~

Every 2-4 weeks



Lightweight business case ~~(lengthy, detailed scope, requirements)~~

A backlog of **User Stories** (analogous to ‘requirements’)

The team commits to complete certain stories in an increment (e.g. “release”, “PI”, “sprint”)

Evidence is captured that each story’s **Acceptance Criteria** (~‘test plan’) was met

Stories are **marked Done** - i.e. completed to the satisfaction of the Product Owner

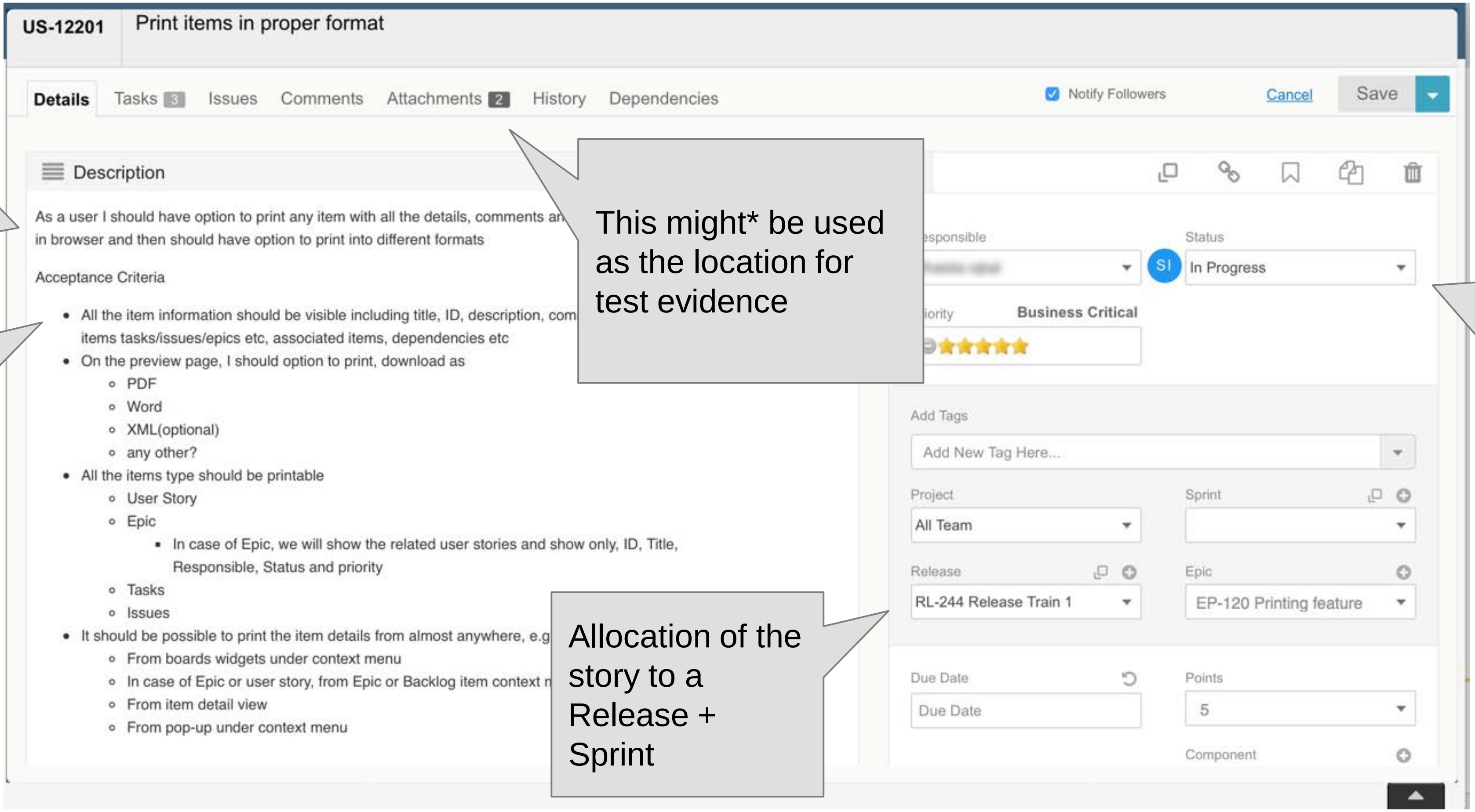
Production Change Approval

..and it could look like this in your Agile tool

Deliberately overly simplified

‘Requirement’ in the form of a user story: “As a …”

Acceptance criteria



US-12201 Print items in proper format

Description

As a user I should have option to print any item with all the details, comments an
in browser and then should have option to print into different formats

Acceptance Criteria

- All the item information should be visible including title, ID, description, com
  items tasks/issues/epics etc, associated items, dependencies etc
- On the preview page, I should option to print, download as
  - PDF
  - Word
  - XML(optional)
  - any other?
- All the items type should be printable
  - User Story
  - Epic
    - In case of Epic, we will show the related user stories and show only, ID, Title, Responsible, Status and priority
  - Tasks
  - Issues
- It should be possible to print the item details from almost anywhere, e.g
  - From boards widgets under context menu
  - In case of Epic or user story, from Epic or Backlog item context n
  - From item detail view
  - From pop-up under context menu

esponsible: [redacted]

Status: SI In Progress

iority: Business Critical ★★★★★

Add Tags: Add New Tag Here...

Project: All Team

Sprint:

Release: RL-244 Release Train 1

Epic: EP-120 Printing feature

Due Date: Due Date

Points: 5

Component

This might* be used as the location for test evidence

Allocation of the story to a Release + Sprint

This might* be used to evidence authorized business review and approval (of ‘requirements’ and ‘tested’ / ‘done’)

**These are common patterns from one example tool. It is up to management to define their standards and controls.*

“

The program team *says* they’re using Agile for the implementation

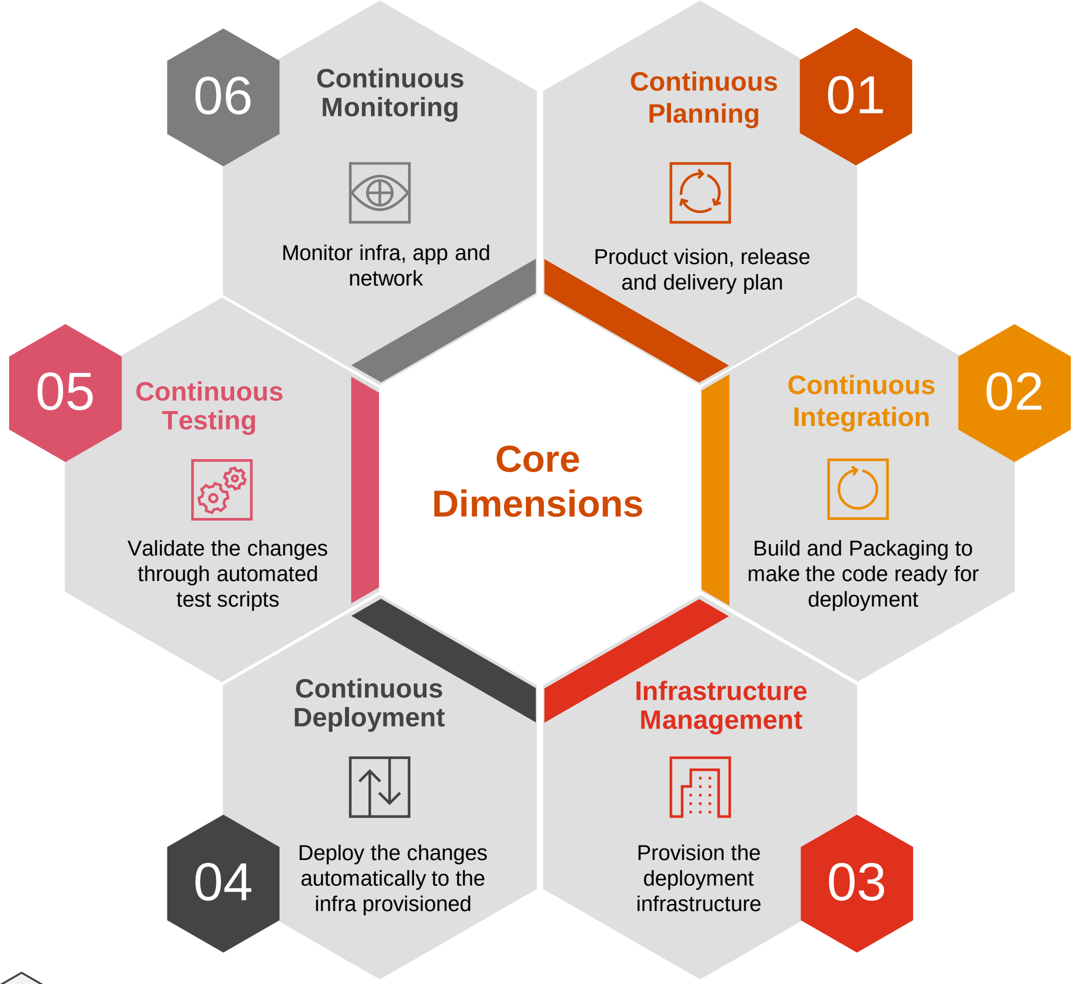
Global Bank - IA case study

What is DevOps? The contraction of “Dev” and “Ops” refers to replacing siloed Development & Operations to create multidisciplinary teams that work together with shared and efficient practices and tools. - Sam Guckenheimer

Splunk, Sumo Logic, Fluentd, Prometheus, ITRS, Moogsoft, Logstash, Nagios, Zabbix, Zenoss

FitNesse, Selenium, Gatling, Cucumber, JUnit, JMeter, TestNG, Mocha, Karma, Jasmine, Tricentis Tosca, Locust.io, Soap UI, Sauce Labs, Perfecto, MicroFocus UFT

XL Deploy, Octopus Deploy, AWS CodeDeploy, ElasticBox, UrbanCode Deploy, GoCD, ElectricCloud, CA Automic



Core Dimensions

06 Continuous Monitoring

Monitor infra, app and network

01 Continuous Planning

Product vision, release and delivery plan

02 Continuous Integration

Build and Packaging to make the code ready for deployment

03 Infrastructure Management

Provision the deployment infrastructure

04 Continuous Deployment

Deploy the changes automatically to the infra provisioned

05 Continuous Testing

Validate the changes through automated test scripts

ServiceNow, Jira, Trello, Slack, Stride, CollaboNet VersionOne, Remedy, Agile Central, OpsGenie, PagerDuty

Jenkins, Bamboo, Travis CI, Circle CI, Codeship, VSTS, TeamCity, AWS CodeBuild

Containers: Docker, Kubernetes, Mesos, Rancker, Docker Enterprise, GKE, AKS, AWS ECS, Rkt, Codefresh, Helm
Cloud: AWS, Google and Azure’s suite of tools

07 **Version Control/Source Code Management** underpins all of the dimensions and will likely be at the center of the process and control structure. Examples: Gitlab, BitBucket, Subversion

Tools displayed above are examples only and neither an exhaustive list nor endorsed.

> All production changes go through our Continuous Deployment tool which has restricted access. Isn’t that enough?

Large cloud ERP - case study

Some common controls and audit concerns with Agile and DevOps

The changing nature of risk ...

- Agile tends toward a different, or less, documentation
- Increased reliance on tools to “control” the process
- Increased reliance on automated testing
- Segregation of duties between DEV and PRD difficult to obtain
- Traditional traceability from requirements to testing to release can be difficult to obtain without a detailed spreadsheet
- Demonstrating business involvement and sign-off
- Traditional status reports and other communications tools are not produced

Considerations (not an exhaustive list)

- Consider what can be extracted from tools and/or automated into the process. Look for defined minimum acceptable standards
- Restricting user access to the tools and the configuration of the automation within the tools will be key.
- Consider what is being done to ensure quality of test scripts and overall coverage
- Look at secure pipelines and/or code reviews to “cleanse” code prior to release into production
- Create linkages and traceability between the tools within the tool chain, essentially automating traceability
- Co-locate the product owner, consider automated controls within tools to demonstrate sign-off
- Educate stakeholders on automated “status” like information in tools, allowing for real time information and decision-making

Tips

Agile and DevOps will primarily impact program development and program change. However, it may not be limited to these areas so you need to understand the risk.

Tips

Performing a detailed walkthrough of the processes, tools, risks and controls is the 1st step. Take the time to build your understanding.

“

What are other typical business issues we are seeing from the adoption of Agile?

Agile and DevOps tend to come naturally to new, digitally-native technology companies with low regulatory burden. Following are other topics that we hear at our clients.

Using agile buzzwords, but in reality:

- Agile as an excuse for no documents
- Excessive overhead (satisfying old and new controls)
- Inconsistent and/or conflicting approaches
- They're 'doing agile' vs 'being agile'

Surrounding processes clash:

- Annual budgeting
- Capex/opex
- Timely involvement of cyber, risk, controls advisory, etc
- … and many more

Lack of a coherent plan for adoption of agile and DevOps (processes, culture, systems, org structures, etc)

Where can I get more guidance on establishing effective controls, and auditing them?

Internal audit: Thinking differently in an agile organization

"As companies adapt to rapid change, the risks they face are often unknown and uncharted and their potential impact on the business's future is substantial. Internal audit functions that are adapting to the changing risk landscape are involved in significant initiatives at the optimal time in the process to be able to focus on strategic risks ahead of risk occurrence."[1]

Agile is trending, having gained increased adoption in recent years. PwC's Global Portfolio and Programme Management Survey 2014 reports that the use of agile project management methodology has increased by 11% since 2012. However, many internal audit functions are struggling with how to interact with such methodologies, especially those that are wedded to traditional system development life-cycle (SDLC) controls. In addition to relying traditional internal audit approaches, teams typically don't have people with the right skills that could add value.

Before talking about internal audit's plans to interact with agile projects, it's important to understand what's meant by agile. Agile is a collective term for iterative software delivery methodologies such as scrum, extreme programming, and feature-driven development. There's no one type of agile delivery, and most organizations take a hybrid approach which draws from multiple methodologies. What the methodologies have in common is their lean and iterative approach to delivery of usable product at the time of the release—in this case, working code that meets requirements and enables business operations. The methodologies focus on perceived value-add processes, which is why agile practitioners are usually critical of the traditional SDLC controls.

The values that underpin those methodologies, as defined by the Agile Manifesto[2], specify that:

- Individuals and interactions are more valuable than processes and tools.
- Working software is a higher priority than comprehensive documentation.
- Customer collaboration is more important than contract negotiation.
- Responding to change is preferable to following a rigid plan.

Readers familiar with the traditional Sarbanes-Oxley based SDLC controls can immediately see how some of the agile values conflict with the traditional internal audit approach. Those controls are very "after the fact," and they rely heavily on documentation, neither of which works well with the agile methodologies.

For internal audit to close that gap, consider the following.

1. Get involved early, and understand what type of agile development process is being used.
2. What controls exist, what risks are being introduced and are they being mitigated?
3. Know how agile teams' define done.
4. Do you have the right skill set to perform the reviews, and are you focused on value?
5. Does your reporting framework supply the type of real-time feedback that agile teams need?

Internal audit should focus on helping the agile team and the company effectively execute their responsibilities while adequately understanding and controlling the company's risks.

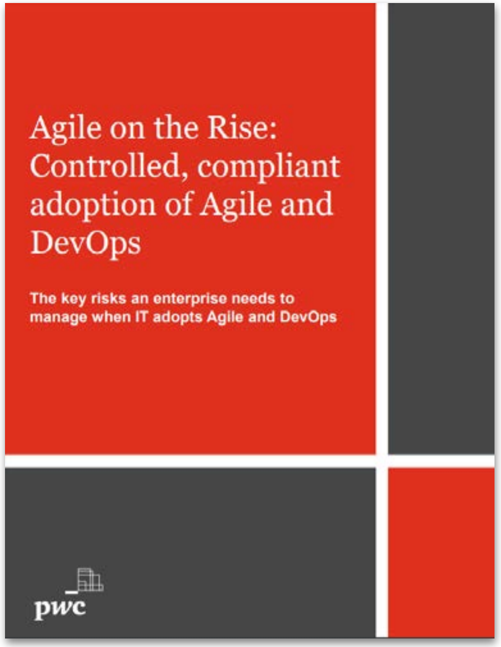
1. Get involved early, and understand what type of agile development process is being used.

The earlier that internal audit gets involved, the better. Working with the program in the early stages increases understanding of the program's life cycle and its key benefits, drivers, and objectives. That understanding enables internal audit to contribute to the program as it defines its risk management approach and strategy.

[1] 2015 State of the Internal Audit Profession Study: Finding True North in a period of rapid transformation, PwC, p. 8.
[2] Source: http://www.agilemanifesto.org.

Internal Audit: Thinking differently in an agile organization

Publication available on request



Agile on the Rise: Controlled, compliant adoption of Agile and DevOps

The key risks an enterprise needs to manage when IT adopts Agile and DevOps

pwc

Our perspective on the areas of friction that emerge when organizations adopt agile

https://www.pwc.com/us/en/services/risk-assurance/library/controlled-compliant-adoption-agile-devsecops.html

Agile on the rise

Integrating effective controls into Agile environments

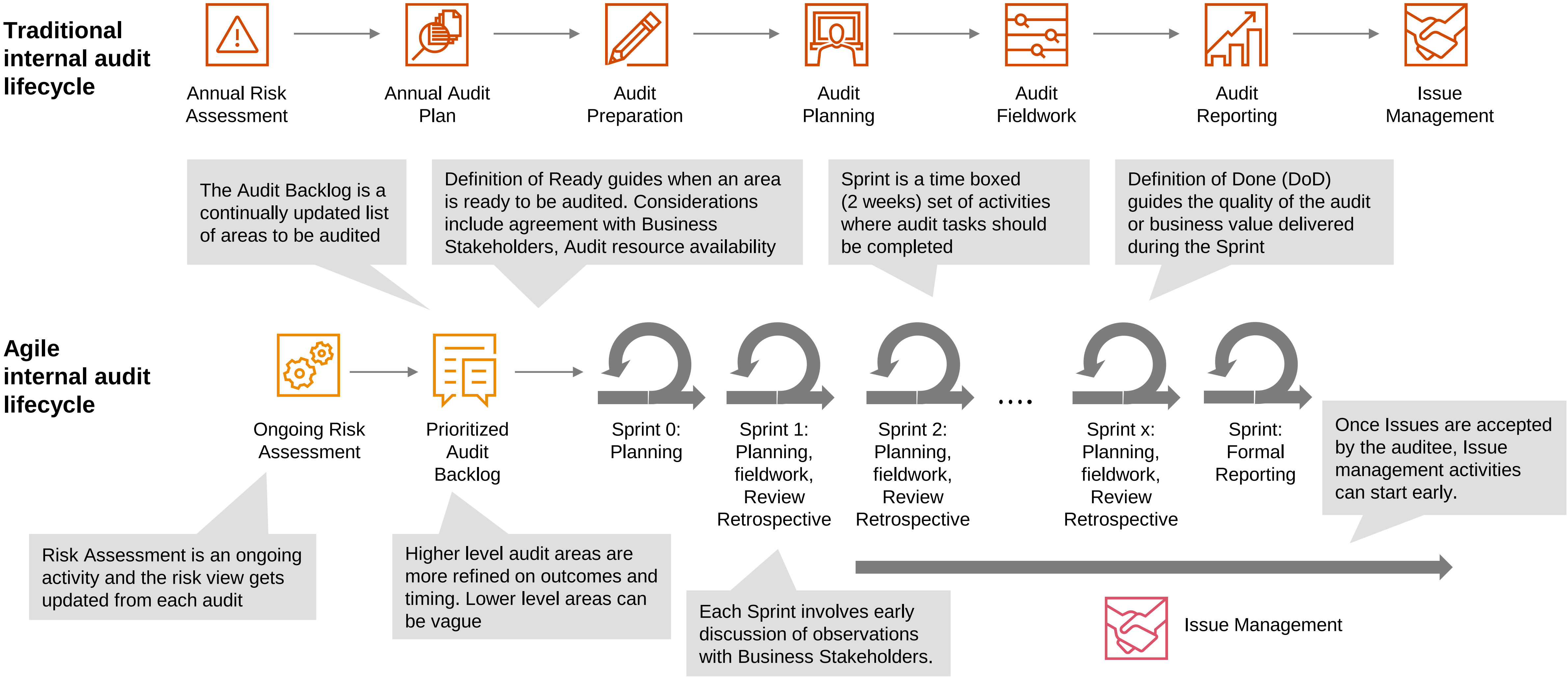
pwc

Our perspective on the audit & controls impacts when organizations adopt agile

https://www.pwc.com/us/en/services/risk-assurance/library/effective-controls-agile-environment.html

Applying Agile methods to Internal Audit Methodology

Imagine if..



Traditional internal audit lifecycle

Annual Risk Assessment → Annual Audit Plan → Audit Preparation → Audit Planning → Audit Fieldwork → Audit Reporting → Issue Management

The Audit Backlog is a continually updated list of areas to be audited

Definition of Ready guides when an area is ready to be audited. Considerations include agreement with Business Stakeholders, Audit resource availability

Sprint is a time boxed (2 weeks) set of activities where audit tasks should be completed

Definition of Done (DoD) guides the quality of the audit or business value delivered during the Sprint

Agile internal audit lifecycle

Ongoing Risk Assessment → Prioritized Audit Backlog → Sprint 0: Planning; Sprint 1: Planning, fieldwork, Review Retrospective; Sprint 2: Planning, fieldwork, Review Retrospective; …. Sprint x: Planning, fieldwork, Review Retrospective; Sprint: Formal Reporting

Once Issues are accepted by the auditee, Issue management activities can start early.

Risk Assessment is an ongoing activity and the risk view gets updated from each audit

Higher level audit areas are more refined on outcomes and timing. Lower level areas can be vague

Each Sprint involves early discussion of observations with Business Stakeholders.

Issue Management

Feedback from the trenches

“We saved 100’s of hours in risk assessment and audit planning”

“We were able to pivot to new risks as needed”

“New templates and processes reduced workload and increased speed”

“Our customers loved the overall audit experience”

Thank you

pwc.com