

THE NEXUS™ FRAMEWORK

FOR SCALING SCRUM



**KURT BITTNER
PATRICIA KONG
DAVE WEST**

Foreword by **KEN SCHWABER**



The Professional Scrum Series



CONTENTS

Foreword		xi
Preface		xiii
Acknowledgments		xvii
About the Authors		xix
Chapter 1	Introduction to Scaling Agile	1
	Why Agile?	2
	Why Scrum?	2
	What Is a Product?	3
	What Is Scrum?	3
	Why Nexus?	5
	Simplicity Is the Key to Scaling	6
Chapter 2	Introducing Nexus	7
	What Is Nexus?	7
	Nexus Extends Scrum	8
	The Nexus Integration Team	10
	Nexus Events	12
	Refinement	13
	Nexus Sprint Planning	13
	The Nexus Daily Scrum	15
	The Nexus Sprint Review	15

	The Nexus Sprint Retrospective	16
	Questions to Ask in Every Nexus Sprint Retrospective	17
	Nexus Artifacts	17
	Product Backlog	17
	Nexus Goal	18
	Nexus Sprint Backlog	18
	Integrated Increment	18
	Artifact Transparency	18
	Definition of “Done” in Nexus	19
	What Do You Need to Get Started with Nexus?	19
	Closing	20
Chapter 3	Forming a Nexus	21
	Evolving a Cross-functional Team	24
	Practice: Opening the Code Base	25
	Practice: Form Teams around Increments of Business Value	26
	Practice: Form Self-Organizing Teams	28
	Growing a Nexus	29
	Starting Small and Then Growing	29
	Using Pairing and “Internship” to Grow Scrum Teams	30
	Why Only Three to Nine Scrum Teams in a Nexus?	31
	Forming the Nexus Integration Team	32
	Who Is on the Nexus Integration Team?	34
	How Does a Nexus Work?	34
Chapter 4	Planning in Nexus	35
	Consolidating and Validating the Product Backlog	35
	Refining the Product Backlog	38
	Cross-Team Product Backlog Refinement	39
	Product Backlog Item Dependencies	42
	Optional Practice: Using Story Mapping to Understand Capabilities and Dependencies	43
	Optional Practice: Using a Cross-Team Refinement Board to Understand Dependencies	45
	Planning a Sprint in a Nexus	47
	Establishing the Nexus Goal	49
	Estimation and Sizing Product Backlog Items	49

	Optional Practice: Connecting Product Backlog Items to Value Delivery	51
	Building the Nexus Sprint Backlog and Scrum Team Backlogs	52
	Closing	55
Chapter 5	Running a Sprint in Nexus	57
	The Nexus Daily Scrum	58
	Providing Transparency Inside and Outside the Nexus	61
	Optional Practice: Product Backlog Treemap	62
	Optional Practice: Visualizing Product Backlog Burndown and Velocity	63
	The Nexus Sprint Review	66
	Optional Practice: Using the “Exposition” (Expo) Format for Nexus Sprint Reviews	67
	Optional Practice: Using Offline Review Techniques for Nexus Sprint Reviews	67
	Nexus Sprint Retrospective	68
	Closing	73
Chapter 6	Evolving the Nexus	75
	Optional Practice: Organizing Scrum Teams around Features	77
	Optional Practice: Managing Code Like an Open-Source Project	78
	Optional Practice: Organizing Teams around Personas	81
	Expanding the Nexus Integration Team	83
	Updating and Refining the Product Backlog	83
	Nexus Sprint Planning, Revisited	85
	The Nexus Daily Scrum, Take Two	86
	The Nexus Sprint Review, Take Two	88
	The Nexus Sprint Retrospective, Take Two	89
	Too Much Work, Not Enough Progress	90
	Growing Technical Debt	91
	Unavailable Product Owner	91
	Inadequate Build and Test Automation	92
	Forming a Plan to Improve	93
	The Challenges of Scaling Scrum	94
	Closing	96

Chapter 7	The Nexus in Emergency Mode	97
	Product Backlog Refinement, Take Three	99
	Nexus Sprint Planning, Take Three	102
	Facilitating Large-scale Distributed Sprint-Planning Sessions	102
	Nexus with Mixed Hardware/Software Development	103
	Teams Working at Different Sprint Cadences	105
	Mixing Scrum and Waterfall Approaches in a Nexus	107
	The Nexus Daily Scrum, Take Three	108
	The Nexus Daily Scrum with Distributed Teams	109
	What to Do When the Nexus Starts to Struggle	110
	The Nexus Integration Team in Emergency Mode	111
	Downscaling	112
	Using a Health Check to Understand Team Sentiments	114
	Scrumbling	115
	The Nexus (Pseudo) Sprint Review and Retrospective	118
	Closing	119
Chapter 8	Retrospective on the Nexus Journey	121
	What Worked Well	122
	The Nexus Daily Scrum	122
	The Nexus Integration Team	123
	Release Frequency	124
	Productivity	125
	Self-organization	125
	Areas for Improvement	126
	Managing Technical Debt	127
	Scaling the Product Owner	128
	Skill Development	129
	Transparency and Trust	130
	What's Next?	132
	Closing	133
	Where to Learn More	134
	Glossary	135
	Index	143

INTRODUCING NEXUS

2

In this chapter, we describe the Nexus Framework in its entirety. As you will see, Nexus is a relatively small and simple extension of Scrum. As we like to say, “Scaled Scrum is still Scrum.” Scrum itself is quite simple, at least to understand. When scaling, this simplicity is a big advantage because complexity is the enemy of scaling. Nexus’ simplicity also makes it highly adaptable, as we will see in subsequent chapters.

WHAT IS NEXUS?

Nexus is a framework that enables multiple Scrum Teams to collaboratively work from a single Product Backlog to deliver at least one “Done” Integrated Increment every Sprint. “Multiple” means, typically, three to nine Scrum teams. Why not two? Because two teams can generally coordinate between one another without additional structure. Why nine? Just as Scrum recommends limiting teams to no more than nine members to improve cohesion and reduce complexity, Nexus recommends the same for the number of teams. Just as in Scrum, however, this upper limit is not absolute and slightly larger numbers may still work, depending on the circumstances. With Nexus we have

discovered that collaboration complexity and coordination between teams increases significantly beyond nine teams, and for those cases some different techniques apply.¹

Since Nexus builds on Scrum, its parts will be familiar to those who have used Scrum. The difference is that more attention is paid to dependencies and communication between Scrum Teams (see Figure 2-1).

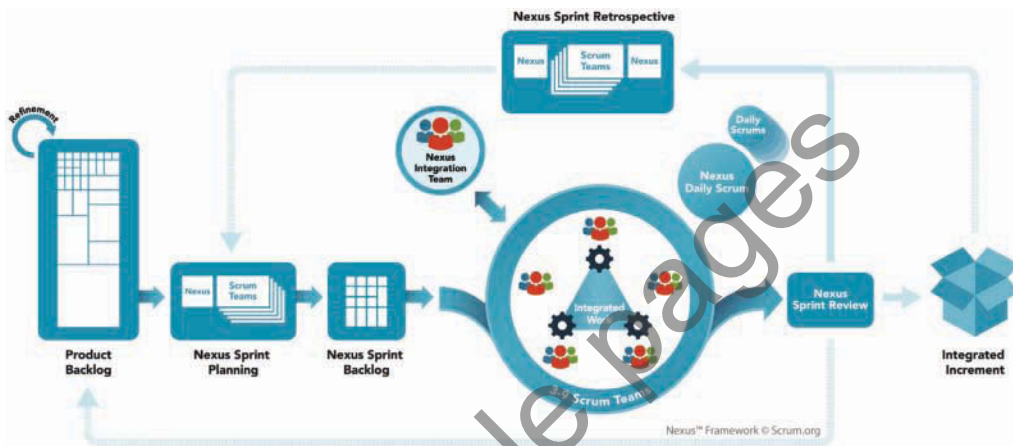


Figure 2-1 The Nexus Framework for scaling Scrum

NEXUS EXTENDS SCRUM

Nexus is Scrum, with some small additions (see Table 2-1).

- **It adds one additional Artifact: the Nexus Sprint Backlog.** Nexus Sprint Backlog is the Nexus’ plan for the Sprint; it helps the Nexus understand what Scrum Teams are working on and makes any dependencies transparent that may exist between the teams during the Sprint.

1. George Miller’s oft-cited paper, “The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information,” describes limitations in the way that we process information and form memories that reinforces ad hoc experience in organizing teams: when teams grow beyond about nine people they begin to lose cohesion and their work becomes harder to manage. For more information on Miller’s article, see https://en.wikipedia.org/wiki/The_Magical_Number_Seven,_Plus_or_Minus_Two.

- **It adds five additional Events: Refinement, Nexus Sprint Planning, the Nexus Daily Scrum, the Nexus Sprint Review, and the Nexus Sprint Retrospective.** These additional events extend Scrum to ensure that work is divided and coordinated across Scrum Teams in the most effective manner possible, and to share experiences across teams in the Nexus.
- **It removes the individual Scrum Team Sprint Review, in favor of the Nexus Sprint Review.** Since Scrum Teams in a Nexus work together to produce a single Integrated Increment, that Integrated Increment should be reviewed as a whole.
- **It adds a new Role: the Nexus Integration Team.** The Nexus Integration Team (NIT) exists to promote and provide transparent accountability for integration in a Nexus. It coaches and guides the application of Nexus with the Scrum Teams as well as within the organization. The NIT consists of the Product Owner of the product, a Scrum Master, and NIT members who are usually members of the Scrum Teams in the Nexus, but may come from other functional areas in the organization such as Operations, Security, Architecture, or other specialist areas that may help the Nexus deliver an Integrated Increment. These “outside” members may be temporary members who join the NIT for as long as is necessary.

Table 2-1 Nexus Roles, Events, and Artifacts

Roles	Events	Artifacts
Development Teams	The Sprint	Product Backlog
Product Owner	<i>Nexus Sprint Planning</i> *	<i>Nexus Sprint Backlog</i> *
Scrum Master	Sprint Planning	Sprint Backlog
<i>Nexus Integration Team</i> *	<i>Nexus Daily Scrum</i> *	Integrated Increment
	Daily Scrum	
	<i>Nexus Sprint Review</i> *	
	<i>Nexus Sprint Retrospective</i> *	
	Sprint Retrospective	
	<i>Refinement</i> *	

* Nexus specific

THE NEXUS INTEGRATION TEAM

The NIT ensures that an Integrated Increment is produced at least every Sprint for the Nexus. The Scrum Teams do the work. Ultimately, the NIT is accountable for maximizing the value of the integrated Product (see Figure 2-2). Their activities may include developing tools and practices that will help with integration or serving as coaches and consultants to help with coordination.

NIT members need to have a teaching mind-set to help Scrum Teams resolve their issues whenever possible. Their role is to help to highlight issues that need to be solved and to help the Scrum Teams solve the issues. Only in emergencies does the NIT jump in and solve problems directly.

The NIT consists of:

- **The Product Owner**, *the* owner of the Product, and ultimately accountable for its success. In the context of the NIT, the Product Owner is accountable for ensuring that maximum value is delivered by the Nexus during each Sprint. The Product Owner's role does not change from Scrum; the scope of the work is simply more complex.
- **A Scrum Master**, who has overall responsibility for ensuring the Nexus framework is enacted and understood. This Scrum Master is often a Scrum Master in one or more of the other Scrum Teams in the Nexus.
- **A Development Team**, whose members are usually members of Scrum Teams in the Nexus.

Contrary to what its name may suggest, the NIT doesn't integrate the work of all Scrum Teams as it is delivered. Instead, it is accountable for ensuring that the teams are able to achieve integration themselves.

Members coach Scrum Teams and help remove dependencies. If something is preventing the Scrum Teams in the Nexus from producing an integrated Product, the NIT is accountable for making sure that those issues get resolved.



Figure 2-2 The NIT is accountable for maximizing the value of the integrated Product

Members of the NIT may also work on Scrum Teams in the Nexus, but when they do they must put their work on the NIT first for the greater benefit the whole Nexus (see Figure 2-3).

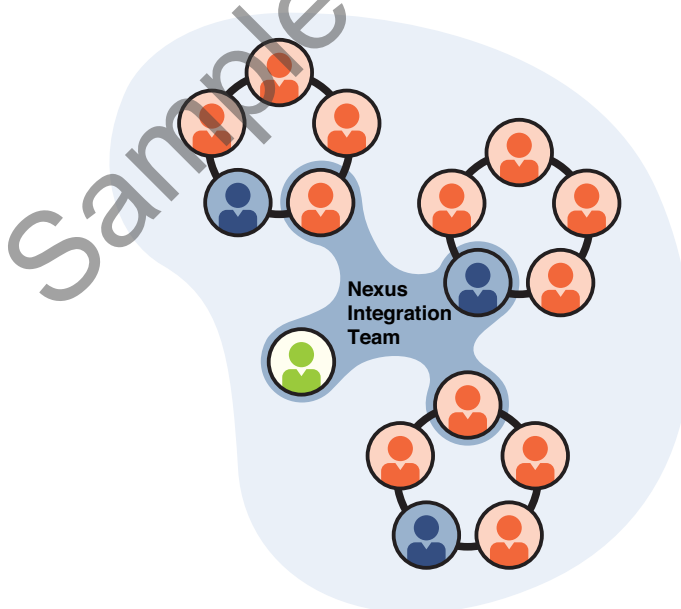


Figure 2-3 Members of the NIT are usually drawn from Scrum Teams

NIT members may come from outside the Scrum Teams; that is, from other parts of the organization. When they do, it is to provide unique expertise that the Scrum Teams lack, in areas such as Enterprise Architecture or Continuous Delivery, or in some area of specialized domain knowledge. NIT members may simply obtain their help without them actually becoming full-fledged members of the Nexus, but in some cases, when extensive support is needed, it may make sense for them to actually join the Nexus. When they do, they become accountable for Product delivery, just as are other members of the NIT.

The membership of the NIT can change over time as its needs evolve. Early in the life cycle of a Nexus, it may focus on coaching the Scrum Teams in the Nexus on scaling practices, or it may be more involved in stabilizing the shared build and test automation framework. Later in the life cycle, when the Nexus is running smoothly, the NIT may shift to raising awareness of issues from cross-team dependencies. Chapter 3, “Forming a Nexus,” describes the formation, composition, and evolution of the NIT in more detail.

NEXUS EVENTS

Nexus adds four events to Scrum, and replaces one Scrum event, to help Scrum Teams divide and coordinate work across teams in the most effective manner.² The events defined by Nexus are

- **Refinement** is a formal event for the Nexus to collaborate on the details of the Product Backlog Items (PBIs) and see that they are adequately independent, so that the teams can select and work on without excessive conflict. In the process of working out the dependencies, teams also work out which backlog items they will likely work on. The Nexus continually refines the Product Backlog, as needed, and there is no specific time box for refinement.

2. Nexus events are guided by the time boxes for their related Scrum events, meaning that they generally take a similar amount of time. As a practical matter, a Nexus event takes as much time as the Nexus needs and is over when it's over. If, after that, the Nexus thinks that it took too long, there is a good opportunity for inspection and adaptation to improve for the next time.

- **Nexus Sprint Planning** helps the teams in the Nexus to collectively agree on the Nexus Goal and how each team will contribute to it.
- **The Nexus Daily Scrum** helps the Nexus to make integration issues transparent so that the Scrum Teams can know who is responsible for fixing them. It is a daily opportunity for the teams in the Nexus to sync with one another.
- **The Nexus Sprint Review** enables the Nexus to gather feedback on the Integrated Increment. It replaces individual Scrum Team Sprint Reviews.
- **The Nexus Sprint Retrospective** helps the teams share experiences and coordinate their resolution of shared challenges.

REFINEMENT

In Scrum, Product Backlog refinement is not a mandatory event, but it is a strongly recommended practice. In Nexus, refinement is essential; it helps Scrum Teams work together to determine which team will deliver specific PBIs and to identify cross-dependencies across teams. Refinement is a cross-team event, with as many Scrum Team members present as is necessary to understand and decompose the PBIs.

Refinement results in a Product Backlog that is granular enough for Scrum Teams to pull without creating unmanageable dependencies. During Refinement, the Scrum Teams should focus on these questions.

- What work will each team pull?
- In what order does that work need to be done to deliver the greatest business value earliest, while minimizing risk and complexity?

NEXUS SPRINT PLANNING

The Nexus takes the refined Product Backlog as input for the Nexus Sprint Planning event (see Figure 2-4). Nexus Sprint Planning helps to synchronize the activities of the Scrum Teams in a Nexus for a single Sprint.

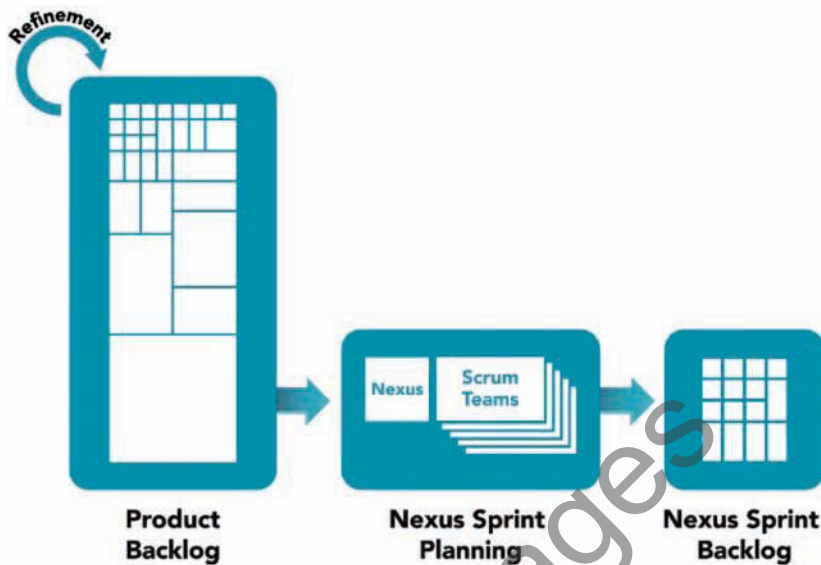


Figure 2-4 Nexus Sprint Planning

Nexus Sprint Planning consists of:

- **Validating the Product Backlog.** The Scrum Teams review the PBIs and make any necessary adjustments needed to the work from the Refinement event. All of the Scrum Teams should participate and contribute to minimize communication issues; however, only the appropriate representatives (those who feel that they can make a contribution to refining the PBIs) from each of the Scrum Teams need to attend.
- **Formulating the Nexus Goal.** The Nexus Goal is a Sprint objective that is met through the implementation of PBIs by multiple teams.
- **Scrum Team Sprint Planning.** Once the Nexus Goal for the Sprint is understood, each Scrum Team will conduct its individual Sprint Planning events in which the members create their own Sprint Backlogs. As they identify dependencies with other teams, they work with those teams to minimize or eliminate the dependencies.

In some cases, this will mean that the sequence of work across teams may have to be adjusted to let one team finish its work before another starts.

This could be accomplished by breaking apart dependent work so that some parts can be worked independently, or by one team choosing non-dependent PBIs to work on, to avoid waste resulting from unresolved cross-team dependencies. Teams may also work together to shift work from one team to another to better balance the work. The NIT will help to make sure that dependencies are communicated and visualized on the Nexus Sprint Backlog.

Nexus Sprint Planning is complete when each Scrum Team in the Nexus has finished its individual Sprint Planning events.

THE NEXUS DAILY SCRUM

The Nexus Daily Scrum brings together the appropriate representatives from individual Scrum Teams to inspect the current state of the Integrated Increment and to identify integration issues or newly discovered cross-team dependencies. Topics typically discussed include the following.

- Was the previous day's work successfully integrated, and if not, why?
- Have any new dependencies been identified?
- What information needs to be shared across teams in the Nexus?

During the Nexus Daily Scrum and throughout the day, the Nexus Sprint Backlog may be updated by the Scrum Teams to visualize and manage current inter-team dependencies. It is not simply an aggregation of the individual teams' Sprint Backlogs, since each team will have work for itself as well as the Product Backlog work that it needs to do. Work that is identified during the Nexus Daily Scrum is then taken back to individual Scrum Teams for planning inside their Daily Scrum events.

THE NEXUS SPRINT REVIEW

The Nexus Sprint Review replaces individual Scrum Team Sprint Reviews and is held at the end of the Sprint. Its purpose is to capture feedback from stakeholders on the entire Integrated Increment of the Nexus. The Nexus Sprint Review replaces the individual Scrum Team Sprint Reviews because individual Scrum Teams might not produce a meaningful Integrated Increment on their own when Nexus is used.

There are several benefits to having a single Sprint Review for the Nexus, including the following.

- Teams are logically each other's stakeholders, so they can provide one another with feedback that helps the Nexus improve.
- If individual Scrum Team Sprint Reviews were held, stakeholders may not be able to attend all of them, and even if they did they would not see the integrated Product.
- Some issues only become evident when the integrated Product is reviewed as a whole, especially when each team is developing one or more components. Each component may work in isolation, but they may not work together to produce an integrated Product.
- Reviewing the Integrated Increment as a whole brings all the teams in the Nexus together and reminds them that their goal is a single integrated solution.

Even when some teams may actually deliver logically separated subproducts that may be independently reviewed, shipped, and used, there is value in reviewing them in the context of the Nexus' integrated Product Increment.

All members of the Nexus participate in the Nexus Sprint Review.

THE NEXUS SPRINT RETROSPECTIVE

The Nexus Sprint Retrospective provides the means by which the Nexus enables inspection and adaptation. To conduct the Nexus Retrospective:

1. Representatives from across the Nexus meet and identify issues that have impeded more than a single team to make shared issues transparent to all Scrum Teams. The representatives consist of the NIT members, as well as anyone with an interest in sharing their perspectives on inter-team issues.
2. Each Scrum Team holds its own Sprint Retrospective, just as they would do in Scrum, but the team also considers issues raised from the first part of the Nexus Retrospective as input to its team discussions while the members determine actions to address these issues.

3. Representatives from the Scrum Teams meet once again to discuss common issues identified in the Scrum Team Retrospectives. They agree on how to visualize and track the identified actions that will enable the Nexus to learn and adapt as a whole.

QUESTIONS TO ASK IN EVERY NEXUS SPRINT RETROSPECTIVE

Nearly every Nexus encounters common scaling challenges. Questions that help teams to identify challenges include the following.

- Was any work left undone?
- Did the Nexus generate technical debt?
- Were all artifacts, particularly code, frequently (as often as every day) successfully integrated?
- Was the software successfully built, tested, and deployed often enough to prevent the overwhelming accumulation of unresolved dependencies?

When challenges are identified, ask the following:

- Why did this happen?
- How can technical debt be undone?
- How can the recurrence be prevented?

Nexus Events are described in more detail in Chapter 5, “Nexus in Action.”

NEXUS ARTIFACTS

Artifacts capture the results of work performed. They also provide transparency and opportunities for inspection and adaptation.

PRODUCT BACKLOG

There is a single Product Backlog for the entire Nexus and all of its Scrum Teams. Since a Nexus is organized a single product, it only has a single Product Owner, and that single Product Owner maintains a single Product Backlog. All teams pull work from this single artifact.

NEXUS GOAL

During the Nexus Sprint Planning meeting, the Product Owner discusses a goal for the Sprint. This is called the Nexus Goal. It is the sum of all the work and Sprint Goals of the individual Scrum Teams within the Nexus. The Nexus should demonstrate the functionality that it developed to achieve the Nexus Goal at the Nexus Sprint Review.

NEXUS SPRINT BACKLOG

The Nexus Sprint Backlog contains the PBIs that have cross-team dependencies or potential integration issues. It does not contain PBIs that have no dependencies, nor does it contain tasks from the individual Scrum Team Sprint Backlogs. It is used to highlight dependencies and the flow of work during the Sprint. It is updated at least daily, often as part of the Nexus Daily Scrum.

INTEGRATED INCREMENT

The Integrated Increment is the integrated aggregation of all work completed by *all* the Scrum Teams in a Nexus. The Integrated Increment must be usable and potentially releasable, which means it must meet the definition of “Done” agreed to by the Development Team. The Product Owner is a key stakeholder for it and defines the quality criteria that the Product Increment must meet. The Integrated Increment is inspected at the Nexus Sprint Review.

ARTIFACT TRANSPARENCY

Just like Scrum, on which it builds, Nexus is based on transparency. The NIT works with the Scrum Teams in the Nexus, and the broader organization, to ensure that all Scrum and Nexus artifacts are visible and that the state of the Integrated Increment can be easily understood.

Decisions made based on the state of Nexus artifacts are only as effective as the level of artifact transparency. Incomplete or partial information will lead to incorrect or flawed decisions, making it difficult or impossible to guide the Nexus effectively to minimize risk and maximize value.

The greatest challenge a Nexus faces is detecting and resolving dependencies before technical debt accumulates to an unacceptable level. The test of unacceptable technical debt is when the Nexus tries to integrate the work from its Scrum Teams. When that integration fails, the unresolved dependencies remain hidden in the code and test base, lowering or negating the value of the software.

DEFINITION OF “DONE” IN NEXUS

The NIT is responsible for a definition of “Done” that can be applied to the Integrated Increment developed each Sprint. All Scrum Teams of a Nexus adhere to this definition of “Done.”

The Increment is done only when it has been determined to be usable and potentially releasable by the Product Owner. A PBI can be considered done when that functionality has been successfully added to the product and integrated into the Increment.

All Scrum Teams are responsible for developing and integrating their work into an Increment that satisfies these attributes. Individual Scrum Teams may choose to apply a more stringent definition of “Done” within their own teams, but they cannot apply less rigorous criteria than agreed for the Increment.

WHAT DO YOU NEED TO GET STARTED WITH NEXUS?

Nexus is based on the Scrum Framework and adds minimal events, roles, and artifacts to increase transparency, communication, and collaboration among teams. The new Nexus events, roles, and artifacts help ensure a successfully Integrated Increment is developed. Just like with Scrum, you don’t need much to get started with Nexus. And just like with Scrum, Nexus is simple to learn but hard to master. Here are the minimal but required prerequisites you will need to implement Nexus.

You should have

- ✓ Scrum experience.
- ✓ A single Product Backlog, and a single Product Owner, for a single Product.
- ✓ Identified teams that will be in a Nexus. They should have an overview of the Nexus Framework.
- ✓ Identified members who will make up the NIT for the Nexus.
- ✓ A definition of “Done.”
- ✓ Identified Sprint cadence.

CLOSING

Nexus is simple to understand, but mastering it takes practice and feedback. Like Scrum, its basic concepts are simple. Like Scrum, it is also not prescriptive; it says that you need to engage the Nexus in planning the Sprint, but it does not tell you how to do that because there are many techniques that you might find useful to help you to plan. In the following chapters, we will explore applying Nexus in a case study. In doing so, we will often illustrate how Nexus works by using specific complementary practices. These are not specifically part of Nexus, but they will help you understand better how Nexus works. With that, let's get started with what it takes to form a Nexus in Chapter 3.