Nexus Guide Notes

Vasileios Papadopoulos

Scaled Sorfware Development

- If more than one Scrum Teams are working off the same Product Backlog and in the same codebase for a product, difficulties arise
- Communication between teams, work integration and testing of the Integrated Increment becomes a challenge
- The above problems become increasingly difficult when three or more Scrum Teams integrate their work into a single Increment

Scaled Sorfware Development - Team Dependencies

- **Requirements**
  The scope of the requirements may overlap and the manner in which they are implemented may also affect multiple teams
- **Domain knowledge**
  Knowledge of the various business and computer systems should be distributed across the Scrum Teams to ensure they have the necessary knowledge to do their work and minimize interruptions between Scrum Teams during a Sprint
- **Software and test artifacts**
  Requirements are, or will be, instantiated in Sorfware

Definition of Nexus

*Nexus (n): A **framework** for developing and sustaining scaled product and software delivery initiatives.*

Concept

- Uses Scrum as building block
- Multiple Scrum Teams work on a single Product Backlog
- The result is an *Integrated* Increment that meets a goal

Note

The difference between Nexus and Scrum is that, in Nexus, more attention is paid to dependencies and interoperation between Scrum Teams delivering at least one "Done" Integrated Increment every Sprint

Nexus Consists of

- The Nexus Integration Team
- Approximately 3 to 9 Scrum Teams

Nexus Integration Team (NIT)

- Constists of:
  - The Product Owner
  - A Scrum Master
  - *One or more* Nexus Integration Team members
- **Accountable** for ensuring that a "Done" Integrated Increment is produced *at least* once every Sprint
- Common activities include coaching, consulting and highlighting awareness of dependencies and cross-team issues
- NIT members might also perform work from the Product Backlog
- Composition of the Nexus Integration Team may change over time to reflect the current needs of a Nexus

Note

Members of the Nexus Integration Team may also be members of individual Scrum Teams. In this case, they must give priority to their work on the Nexus Integration Team

Product Owner in the NIT

- Responsible for maximizing the value of the Product
- Nexus works off a single Product Backlog
- A Product Backlog has a **single** Product Owner

Scrum Master in the NIT

- Has overall responsibility for ensuring Nexus framework is understood and enacted
- May also act as Scrum Master in one or more of the Scrum Teams

Nexus Integration Team Members

- Responsible for coaching and guiding the Scrum Teams in Nexus
- If their primary responsibility is satisfied, Nexus Integration Team members may also work as Development Team members in one or more Scrum Teams

Nexus Events

- Refinement
- Nexus Sprint Planning
- Nexus Daily Scrum
- Nexus Sprint Review
- Nexus Sprint Retrospective

Refinement

- Helps the Scrum Teams forecast which team will deliver which Product Backlog Items
- Identifies dependencies across Scrum Teams
- After Refinement Product Backlog Items should be sufficiently independent to be worked on by a single Scrum Team without excessive conflict
- Number, frequency and *attendance* of refinement is based on the dependencies and uncertainty in the Product Backlog Items
- Refinement will continue within each Scrum Team in order for Product Backlog Items to be ready for selection in a Nexus Sprint Planning event

Nexus Sprint Planning

- Coordinate the activities of all Scrum Teams in a Nexus for a single Sprint
- Product Backlog should be adequetly refined with dependencies identified and removed or minimized
- Appropriate representatives from each Scrum Team validate and make adjustments to the ordering of the work created during refinement events
- All members of Scrum Teams should participate to minimize communication issues
- Product Owner discusses the *Nexus Sprint Goal*
- Planning continues with each Scrum Team performing their own separate Sprint Planning
- Scrum teams share newly found dependencies with other Scrum Teams
- Planning is complete when each Scrum Team has finished their individual Sprint Planning events

Note

All Product Backlog Items selected for this Sprint and their dependencies should be made transparent on the *Nexus Sprint Backlog*

Nexus Sprint Goal

- Objective set for the Sprint
- Sum of all the work and Sprint Goals of the Scrum Teams

Nexus Daily Scrum

- Appropriate representatives from Development Teams
- Inspect current state of the integrated Increment
- Identify integration issues or cross-team dependencies

Nexus Daily Scrum - Topic

- Was the previous day's work successfully integrated?
- What new dependencies or impacts have been identified?
- What information needs to be shared across teams?

Note

At least every Nexus Daily Scrum the Nexus Sprint Backlog should be adjusted to reflect the work of the Scrum Teams within Nexus

Note

When the Nexus Daily Scrum is completed, individual Scrum Teams take back issues and work that were identified during the Nexus Daily Scrum for planning inside their individual Daily Scrum events

Nexus Sprint Review

- Held at the end of the Sprint
- Replaces individual Scrum Team Sprint Reviews
- May not be possible to show all completed work in detail
- Result: Revised Product Backlog

Nexus Sprint Retrospective

- Representatives from across Nexus meet and identify issues that have impacted *more than a single team*
- Each Scrum Team holds its own Retrospective
- Using issues from the first part of the Nexus Sprint Retrospective, Scrum Teams should form actions to address these issues
- Representatives from the Scrum Teams meet again and agree on how to visualize and track the identified actions

Nexus Sprint Retrospective - Subjects

- Was any work left undone?
- Did Nexus generate technical debt?
- Were all artifacts, paricularly code, frequently and successfully integrated?
- Was the sorfware successfully built, tested and developed often enough to prevent the overwhelming accummulation of unresolved dependencies?

Nexus Artifacts

- Product Backlog
- Nexus Sprint Backlog
- Integrated Increment

Note

Product Backlog Items are deemed "Ready" for the Nexus Sprint Planning when Scrum Teams can select items to be done with no or minimal dependencies with other Scrum Teams

Note

Nexus Sprint Backlog is updated daily often as part of the Nexus Daily Scrum

Integrated Increment

- Represents sum of integrated work completed by a Nexus
- Must be usable and potentially releasable
- Must meet definition of "Done"
- Inspected during Nexus Sprint Review

Definition of "Done"

- Nexus Integration Team is responsible for a Definition of "Done" that can be applied to the Integrated Increment developed each Sprint
- All Scrum Teams adhere to this Definition of "Done"
- Individual Scrum Teams may choose to apply a more stringent Definition of "Done" but cannot apply less rigorous criteria than those agreed for the Increment

Note

The Nexus Integration Team is accountable for ensuring that a "Done" Integrated Increment is produced at least once every Sprint

References

[1] Ken Schwaber. The nexus guide. `https://www.scrum.org/resources/online-nexus-guide`, January 2018.

[2] Ken Schwaber and Jeff Sutherland. The scrum guide. `https://www.scrumguides.org/`, November 2017.