SCRUMstudy
Targeting success

A Guide to the

SCRUM BODY OF KNOWLEDGE

(SBOK® Guide)

12. RELEASE

A Comprehensive Guide to Implementing and Scaling Scrum, with Practical Examples

(Includes insights into how Artificial Intelligence can enhance Scrum processes)

12. RELEASE

The Release phase emphasizes delivering the Accepted Deliverables to the customer and identifying, documenting, and internalizing the lessons learned during the project.

Release, as defined in *A Guide to the Scrum Body of Knowledge* (*SBOK® Guide*), is applicable to the following:

- Portfolios, programs, and/or projects in any industry
- Products, services, or any other results to be delivered to business stakeholders
- Projects of any size or complexity

The term "product" in the *SBOK® Guide* may refer to a product, service, or other deliverable. Scrum can be applied effectively to any project in any industry—from small projects or teams with as few as six team members to large, complex projects with up to several hundred members in several teams.

To facilitate the best application of the Scrum framework, this chapter identifies inputs, tools, and outputs for each process as either "mandatory" or "optional." Inputs, tools, and outputs denoted by asterisks (*) are mandatory, or considered critical for project success, whereas those with no asterisks are optional.

It is recommended that the Scrum Team and those individuals being introduced to the Scrum framework and processes focus primarily on the mandatory inputs, tools, and outputs; while Product Owners, Scrum Masters, and other more experienced Scrum practitioners strive to attain a more thorough knowledge of the information in this entire chapter.

This chapter is written from the perspective of one Scrum Team working on one Sprint to produce potentially shippable deliverables, which could be part of a larger project, program, or portfolio. Additional information pertaining to Scaling Scrum for Large Projects is available in chapter 13. Additional information pertaining to Scaling Scrum for the Enterprise can be found in chapter 14.

The Release phase is typically conducted multiple times during a Scrum project. Although the result of each Sprint is a potentially shippable product, there is not necessarily a release after every Sprint.

Deliverables from accepted User Stories of one or more previous Sprints are released to relevant business stakeholders for acceptance and use as defined in the Release Schedule.

In addition to the release of deliverables, the Scrum Core Team and organizational business stakeholders determine ways to improve the execution of future releases in the project.

It is also important to realize that although all phases and processes are defined uniquely in the SBOK® Guide, they are not necessarily performed sequentially or separately. At times, it may be more appropriate to overlap some phases and/or processes, depending on the specific requirements of each project.

Figure 12-1 provides a high-level overview of the "release" phase in scrum. Includes key processes like shipping deliverables and conducting release retrospectives to ensure stakeholder value and continuous improvement at the product level.

12.1 Ship Deliverables—In this process, all deliverables from the accepted User Stories of previously completed Sprints are delivered or transitioned to the relevant business stakeholders. A formal Working Deliverables Agreement documents the successful completion of the release.

12.2 Retrospect Release—In this process, which completes a release, business stakeholders and Scrum Core Team members assemble to reflect on the release and identify, document, and internalize the lessons learned. Often these lessons lead to the documentation of agreed actionable improvements to be implemented in future project releases.

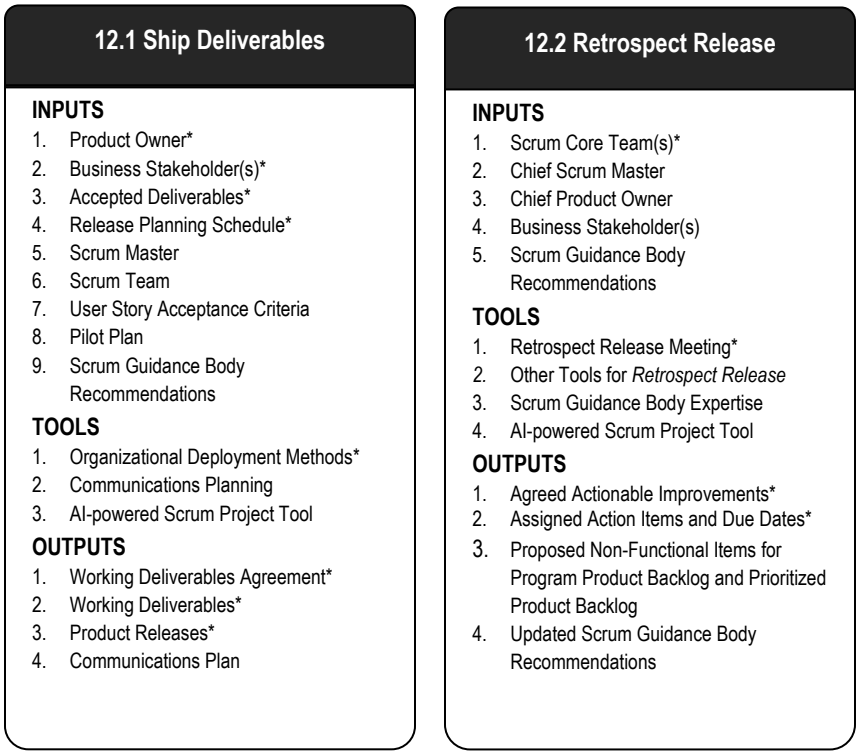

Figure 12-1: Release Overview

Note: Asterisks (*) denote a "mandatory" input, tool, or output for the corresponding process.

Figure 12-2 below shows the mandatory inputs, tools, and outputs for processes in the Release phase.

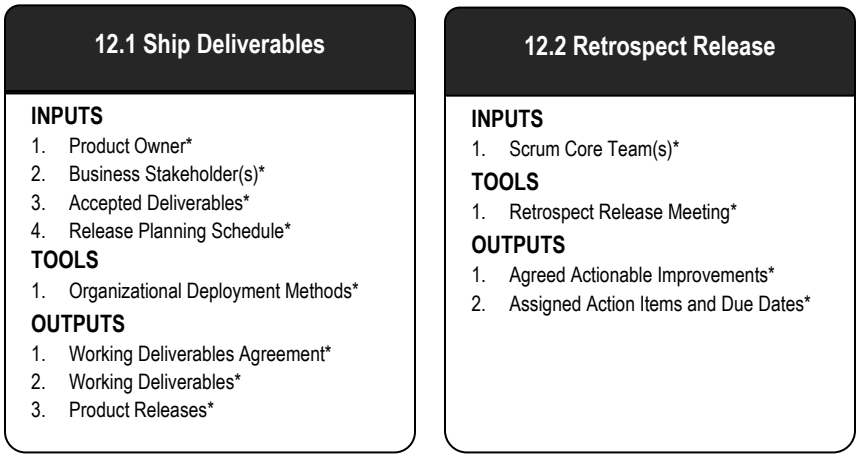

Figure 12-2: Release Overview (Essentials)

Note: Asterisks (*) denote a "mandatory" input, tool, or output for the corresponding process.

12.1 Ship Deliverables

In this process, all deliverables from the accepted User Stories of previously completed Sprints are delivered or transitioned to the relevant business stakeholders. A formal Working Deliverables Agreement documents the successful completion of the release.

Figure 12-3 shows all the inputs, tools, and outputs for the *Ship Deliverables* process.

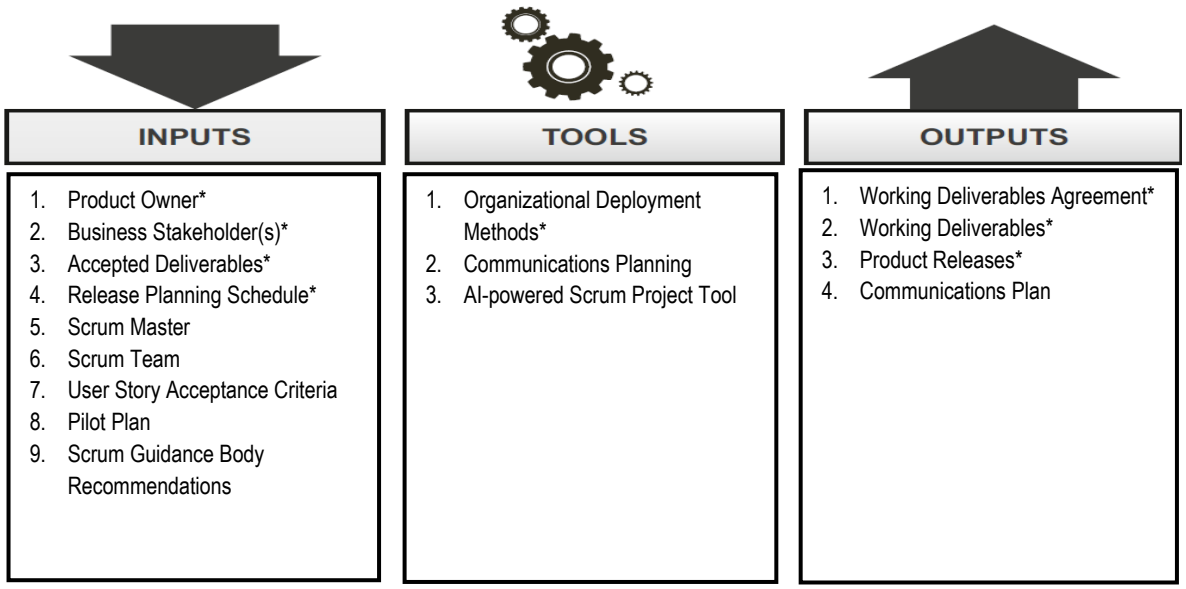

Figure 12-3: Ship Deliverables—Inputs, Tools, and Outputs

Note: Asterisks (*) denote a "mandatory" input, tool, or output for the corresponding process.

Figure 12-4 is a data flow diagram for the “ship deliverables” process. Shows how deliverables and planning data flow into deployment tools to generate product releases and communication plans.

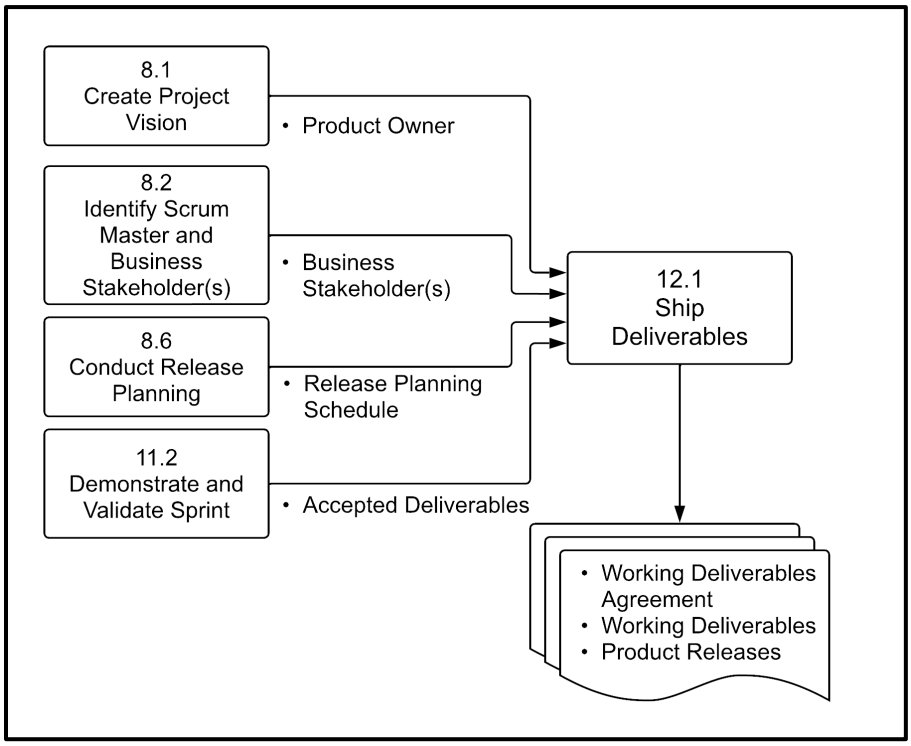

Figure 12-4: Ship Deliverables—Data Flow Diagram

12.1.1 Inputs

12.1.1.1 Product Owner*

Described in section 8.1.3.1.

12.1.1.2 Business Stakeholder(s)*

Described in section 8.2.3.2.

12.1.1.3 Accepted Deliverables*

These are the deliverables that are created by the Scrum Team and associated with the Accepted User Stories which are formally approved by the Product Owner.

Accepted User Stories is described in section to 11.1.3.1.

12.1.1.4 Release Planning Schedule*

Described in section 8.6.3.1.

12.1.1.5 Scrum Master

Described in section 8.2.3.1.

12.1.1.6 Scrum Team

Described in section 8.3.3.1.

12.1.1.7 User Story Acceptance Criteria

Described in section 9.1.3.2.

12.1.1.8 Pilot Plan

A Pilot Plan can be used to map out a pilot deployment in detail. The scope and objectives of the deployment, the target deployment user base, a deployment schedule, a transition plan, required user preparation, evaluation criteria for the deployment, and other key elements related to the deployment are specified in the Pilot Plan and shared with business stakeholders.

12.1.1.9 Scrum Guidance Body Recommendations

In the *Ship Deliverables* process, the Scrum Guidance Body can provide recommendations and guidelines regarding the deployment of products. Best practices should be considered when deploying a product to the customer in order to maximize the value delivered. For more information on Scrum Guidance Body Recommendations, see section 8.1.1.7.

12.1.2 Tools

12.1.2.1 Organizational Deployment Methods*

The deployment mechanisms of each organization tend to be different based on the industry, target users, and positioning of the product. Depending on the product being delivered, deployment can take place remotely or may involve the physical shipping or transition of the item(s). Because deployment often involves an important level of risk, organizations normally have well-defined and established deployment mechanisms, with detailed processes in place to ensure compliance with any applicable standards and quality assurance factors. These might include signoffs by specific management representatives, user approval mechanisms, and guidelines regarding the minimum functionality needed for a release.

12.1.2.2 Communications Planning

Communications planning is used to create the project's Communications Plan. This plan specifies the records that must be created and maintained throughout the project. A variety of methods are used to convey important project information to business stakeholders. As the User Story deliverables are tested, the status of the testing activities is communicated as per the Communications Plan as determined by the Product Owner and sponsor.

12.1.2.3 AI-powered Scrum Project Tool

Described in section 2.4.4

Figure 12-5 is a Vabro interface used for release management in scrum. Displays how ai-powered tools assist with packaging, approval tracking, and version release documentation for stakeholder delivery.

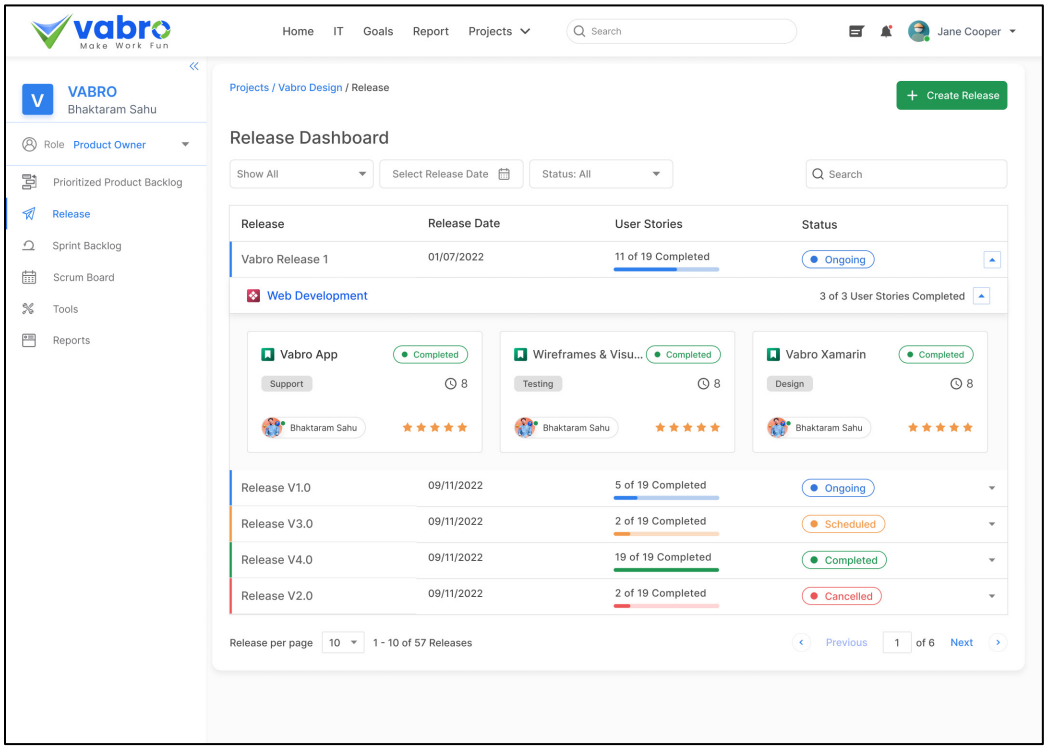

Figure 12-5: Release Management in Scrum (Source: Vabro)

Figure 12-6 is a Jira interface for managing releases. Includes visualization of release dates, deployment checklists, and linked backlog items to ensure completeness and traceability.

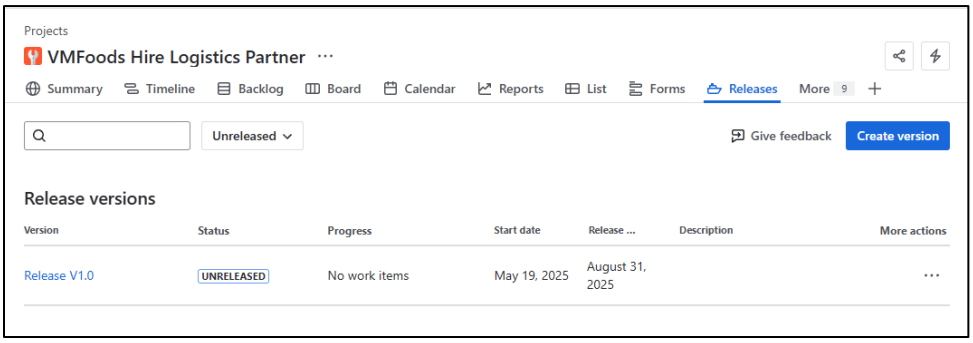

Figure 12-6: Release Management in Scrum (Source: Jira)

12.1.3 Outputs

12.1.3.1 Working Deliverables Agreement*

Deliverables that are accepted receive formal business sign-off and approval from the customer and/or sponsor. Obtaining formal acceptance from the client for each working deliverable is essential for revenue recognition, the acceptance of the overall project results, and the fulfillment of the project objectives.

12.1.3.2 Working Deliverables*

Working deliverables are the final completed, tested, and approved and shippable deliverables for which the project was sanctioned. As new product increments are created, they are continually integrated into prior increments, so there will always be a potentially shippable product available at all times throughout the project.

12.1.3.3 Product Releases*

Product releases should include the following:

- Release Content—This consists of essential information about the deliverables that can assist the customer support team in their use.
- Release Notes—Release notes include external or market-facing shipping criteria for the product to be delivered.

12.1.3.4 Communications Plan

The Communications Plan defines the methods to be used to convey important project information to business stakeholders, as well as who is responsible for various communication activities. A common communication mechanism is a visual display depicting essential information in an easy-to-interpret format, posted in an accessible location, and kept up to date with the most current information.

12.2 Retrospect Release

In this process which completes a release, business stakeholders and Scrum Core Team members assemble to reflect on the release and identify, document, and internalize the lessons learned. Often these lessons lead to the documentation of agreed actionable improvements to be implemented in future project releases.

Figure 12-7 shows all the inputs, tools, and outputs for the *Retrospect Release* process.

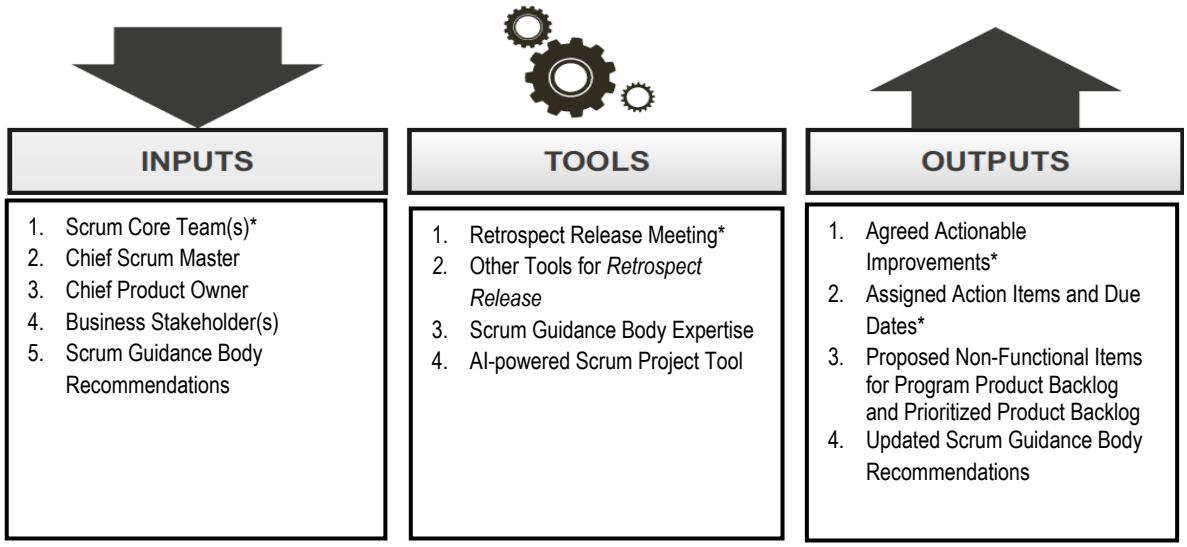

Figure 12-7: Retrospect Release—Inputs, Tools, and Outputs

Note: Asterisks (*) denote a "mandatory" input, tool, or output for the corresponding process.

Figure 12-8 is a data flow diagram for the “Retrospect Release” process. Illustrates how feedback from program or portfolio levels feeds into improvements, action plans, and updated scrum practices.

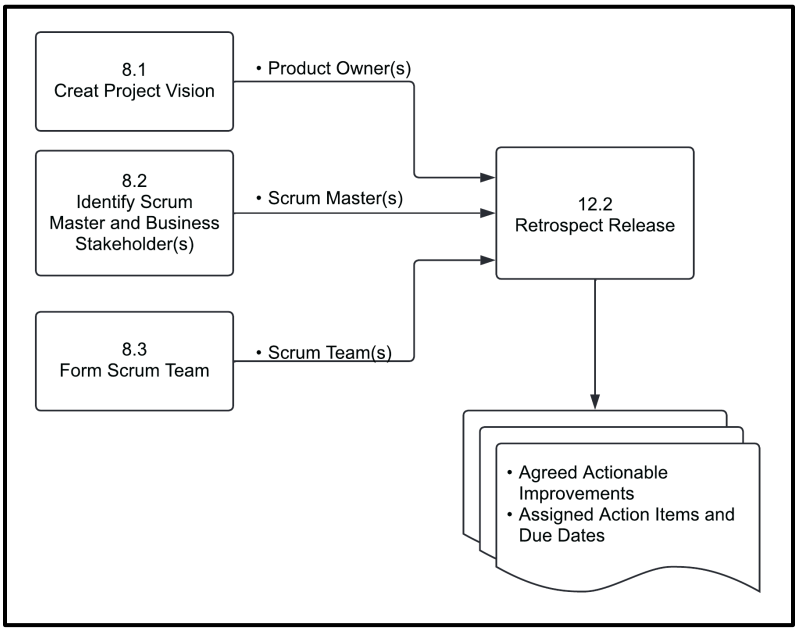

Figure 12-8: Retrospect Release—Data Flow Diagram

12.2.1 Inputs

12.2.1.1 Scrum Core Team(s)*

Described in section 8.4.1.1.

12.2.1.2 Chief Scrum Master

Described in section 3.7.2.2.

12.2.1.3 Chief Product Owner

Described in section 3.7.2.1.

12.2.1.4 Business Stakeholder(s)

Described in section 8.2.3.2.

12.2.1.5 Scrum Guidance Body Recommendations

In the *Retrospect Release* process, recommendations from the Scrum Guidance Body can include a repository of internal templates that support all projects and guidance for conducting the Retrospect Release Meeting. The guidance provided can relate to administrative procedures, audits, evaluations, and project transition criteria. Often, recommendations also include how the organization will maintain the knowledge base of lessons learned and information from all projects. For more information on Scrum Guidance Body Recommendations, see section 8.1.1.7.

12.2.2 Tools

12.2.2.1 Retrospect Release Meeting*

The Retrospect Release Meeting is a meeting used to determine ways in which team collaboration and effectiveness can be improved in future releases. Positives, negatives, and potential opportunities for improvements are also discussed. This meeting is not Time-boxed and may be conducted in person or in a virtual format. Attendees include the Scrum Core Team, Chief Scrum Master, Chief Product Owner, and business stakeholder(s). During the meeting, lessons learned are documented and participants look for opportunities to improve processes and address inefficiencies. If suggestions for improvements are outside the approval for the Scrum Core Team and/or the Scrum Guidance Body, these suggestions should be escalated to the appropriate organizational executives or others outside the project.

12.2.2.2 Other Tools for Retrospect Release

Some of the tools used in the *Retrospect Sprint* process can also be used in this process. Examples include:

- Explorer, Shopper, Vacationer, Prisoner (ESVP),
- Speed Boat, and
- Metrics and Measuring Techniques

For more information on the above tools, see sections 11.2.2.2, 11.2.2.3, and 11.2.2.4.

12.2.2.3 Scrum Guidance Body Expertise

In the *Retrospect Release* process, the primary responsibility of the Scrum Guidance Body is to ensure that the lessons learned from all projects are not lost but are embedded throughout the organization and its continuous improvement efforts. In addition to Scrum-related expertise, expertise may also be provided in various other areas (such as quality management and human resource management), that may be helpful in the *Retrospect Release* process. Also, there may be suggestions from the Scrum Guidance Body concerning how the Retrospect Release Meeting should be conducted. For more information on Scrum Guidance Body Expertise, see section 8.4.2.7.

12.2.2.4 AI-powered Scrum Project Tool

Described in section 2.4.4

12.2.3 Outputs

12.2.3.1 Agreed Actionable Improvements*

Described in section 11.2.3.1.

12.2.3.2 Assigned Action Items and Due Dates*

Described in section 11.2.3.2.

12.2.3.3 Proposed Non-Functional Items for Program Product Backlog and Prioritized Product Backlog

When the initial Program Product Backlog or Prioritized Product Backlog are developed, they are based on User Stories and required functionalities. Often, non-functional requirements may not be fully defined in the early stages of the project and can surface during the Sprint Review, Retrospect Sprint, or Retrospect Release Meetings. These items should be added to the Program Product Backlog (for the program) and the relevant Prioritized Product Backlog (for the project) as they are discovered. Some examples of non-functional requirements are response times, capacity limitations, and security-related issues.

12.2.3.4 Updated Scrum Guidance Body Recommendations

Described in section 11.2.3.6.

12.3 Release Phase Data Flow Diagram

Figure 12-9 depicts the final two processes in the release phase: "Ship Deliverables" and "Retrospect Release." It shows the sequential flow where deliverables are shipped to the customer, followed by a release retrospective to evaluate performance, gather lessons learned, and improve future delivery cycles.

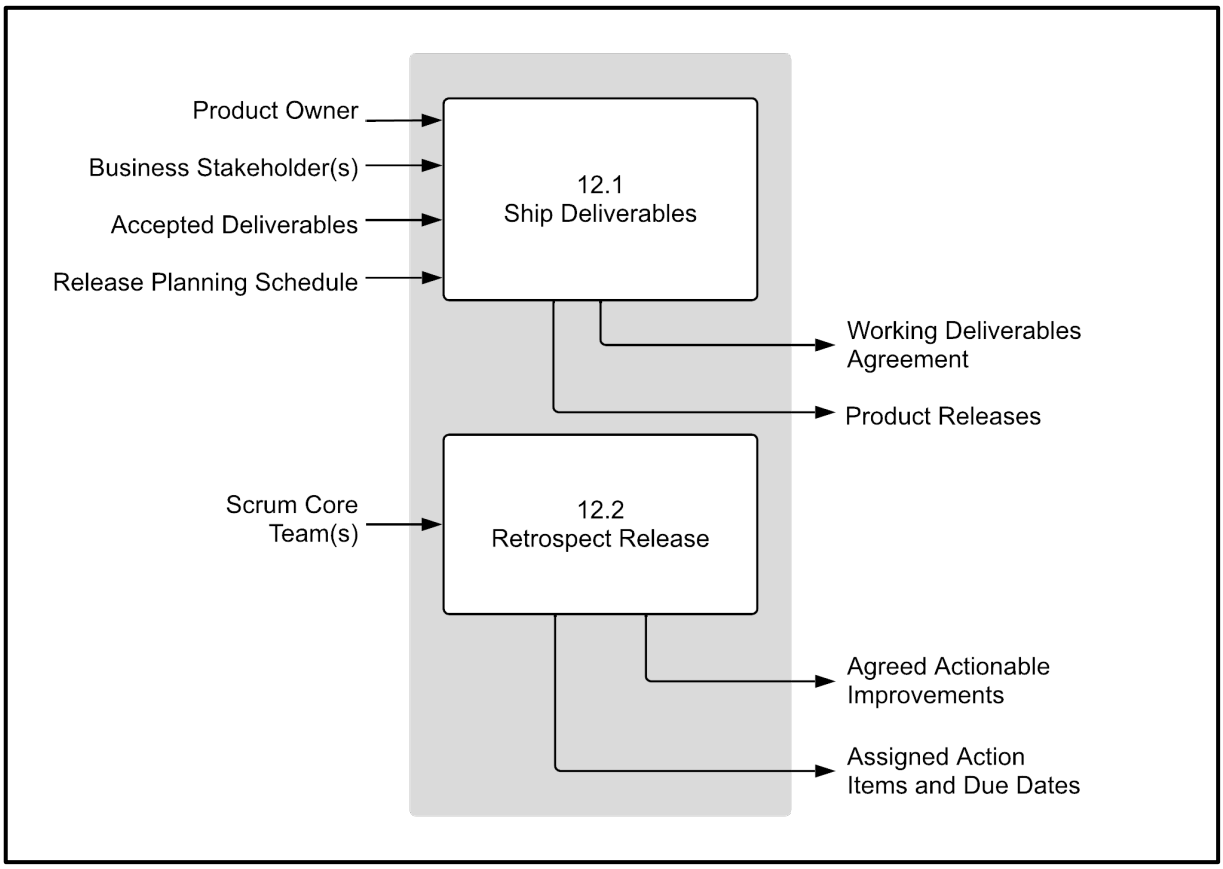

Figure 12-9: Release Phase—Data Flow Diagram

The Essential Guide to Successfully Deliver Projects using Scrum

A Guide to the Scrum Body of Knowledge (SBOK® Guide) provides comprehensive guidelines for the successful implementation of Scrum—the most popular Agile product development and project delivery approach. Defined in the SBOK® Guide as a flexible framework, Scrum can be applied to portfolios, programs, or projects of any size or complexity across industries to deliver products, services, or other results.

This Fifth Edition is based on the collective knowledge gained from thousands of projects across diverse organizations and industries. It reflects contributions from a large number of experts in Scrum and project delivery. Feedback from the global Scrum community played a vital role in shaping improvements and additions, making the SBOK® Guide a truly collaborative effort.

Unlike other Scrum references, the SBOK® Guide is available for free on Scrumstudy.com, along with free certifications, webinars, videos, and study guides. It is ideal for professionals seeking a foundational understanding of Business Analysis or exploring careers in related fields. The Guide addresses real-life challenges faced by Scrum practitioners and explains how to solve them using modern tools and Artificial Intelligence (AI).

Designed to be accessible and engaging, the SBOK® Guide follows the 80-20 rule—80% of key concepts can be learned by reading just 20% of the content, with the remainder available for deeper reference. It is more readable than most Scrum books, which are often either too simplistic or overly detailed.

The SBOK® Guide serves as a reference for both experienced practitioners and those with no prior knowledge of Scrum or project delivery methods. Organized for easy navigation, the SBOK® Guide aims to inform, support, and inspire all readers through its rich, collaborative content.

SCRUMstudy™ is a brand of VMEdu, Inc.