SAFe Practitioner Certification Study Notes

Code: SP

Understanding ARTs

Agile Release Train (ART)

The primary value delivery mechanism in SAFe

What is an ART?

Aspect	Description
Definition	A long-lived team of Agile teams that develops and delivers solutions
Size	Typically 50-125 people (5-12 Agile teams)
Purpose	Deliver a continuous flow of value to customers
Alignment	Common vision, program backlog, and roadmap
Cadence	Operates on fixed PI cadence (8-12 weeks)

ART Characteristics

- **Long-lived:** Teams stay together across multiple PIs
- **Self-organizing:** Teams determine how to accomplish work
- **Cross-functional:** Has all capabilities needed to deliver value
- **Aligned:** All teams work toward common PI Objectives
- **Synchronized:** All teams on same iteration boundaries

Exam Focus Areas

- ART is the primary value delivery construct in SAFe
- ARTs are virtual organizations that span functional boundaries
- Every ART has a clear mission and value stream alignment
- ART success is measured by Business Value delivered

Key ART Roles

The people who make the ART work

ART Leadership Roles

Role	Responsibility
Release Train Engineer (RTE)	Servant leader who facilitates ART events and processes
Product Management	Owns the Program Backlog and drives PI content
System Architect	Defines overall technical and architectural vision
Business Owners	Key stakeholders who evaluate PI Objectives and accept value

Team-Level Roles

Product Owner

Owns Team Backlog, defines stories, accepts completed work

Scrum Master

Facilitates team events, removes impediments, coaches team

Team Members

Cross-functional developers who build and deliver value

Shared Services

- **System Team:** Provides infrastructure, CI/CD, and integration support
- **UX/Design:** Ensures consistent user experience across features
- **Architecture:** Maintains technical integrity and enablers

⚠ Exam Focus Areas

- RTE is a servant leader, not a command-and-control manager
- Product Management ≠ Product Owner (different scope)
- Business Owners assign Business Value to PI Objectives
- Teams are cross-functional with all skills needed to deliver

ART-Level Events

Ceremonies that keep the ART aligned and synchronized

Key ART Events

Event	Frequency	Purpose
PI Planning	Start of each PI	Align all teams on PI Objectives and plan iterations
Scrum of Scrums	2-3 times/week	Coordinate across teams, surface dependencies and impediments
PO Sync	Weekly	Sync Product Owners on backlog, priorities, and scope

Event	Frequency	Purpose
ART Sync	Weekly	Review progress toward PI Objectives, address blockers
System Demo	End of each iteration	Demonstrate integrated increment to stakeholders
Inspect & Adapt	End of each PI	Reflect on PI, identify improvements, solve problems

Event Cadence

- **PI Planning:** 2 days at start of every PI
- **System Demo:** Once per iteration (every 2 weeks)
- **Inspect & Adapt:** Half-day at end of each PI
- **Sync events:** As needed to maintain coordination

> ⚠ **Exam Focus Areas**
>
> - PI Planning is the 'heartbeat' of the ART
> - System Demo shows integrated work from ALL teams
> - I&A includes quantitative review + problem-solving workshop
> - Scrum of Scrums focuses on dependencies and impediments

Executing Iterations

Iteration Planning

Planning work for the upcoming iteration

Purpose and Timing

When

First day of each iteration (typically 2-4 hours)

Who

Entire Agile team, PO required, stakeholders welcome

Output

Iteration Backlog with committed stories

Goal

Align team on what will be delivered

Iteration Planning Steps

- **1. Review PI Objectives:** Understand the bigger picture goals
- **2. Discuss capacity:** Account for PTO, meetings, and other commitments
- **3. Review and refine stories:** Ensure stories meet Definition of Ready
- **4. Estimate and commit:** Team determines what they can deliver
- **5. Identify dependencies:** Surface cross-team dependencies early
- **6. Final commitment:** Team commits to the iteration backlog

Story Points and Velocity

Concept	Description
Story Points	Relative estimate of effort, complexity, and risk
Velocity	Average story points completed per iteration
Capacity	Available time considering holidays, PTO, etc.
Load Factor	Percentage of capacity spent on planned work

Exam Focus Areas

- Teams commit to stories, not story points
- Velocity should stabilize over 3-4 iterations
- Never compare velocity across teams

- Include buffer for unplanned work and emergencies

Daily Execution

How teams work during the iteration

Daily Stand-up (DSU)

Aspect	Details
Duration	15 minutes maximum, timeboxed
Frequency	Every day, same time and place
Format	Three questions or walk-the-board
Focus	Coordination, not status reporting

Three Questions Format

- **What did I do yesterday** that helped the team meet the iteration goal?
- **What will I do today** to help the team meet the iteration goal?
- **Are there any impediments** blocking me or the team?

Work Flow Practices

Pull System

Team members pull work when ready, not pushed

WIP Limits

Limit work-in-progress to improve flow

Swarming

Multiple team members collaborate on blocked items

Pairing

Two people work together for quality and learning

Exam Focus Areas

- DSU is for the team, not for management
- If discussion needed, take it offline (parking lot)
- Scrum Master facilitates but doesn't lead
- Walk-the-board focuses on work items, not people

Iteration Review & Retrospective

End-of-iteration ceremonies

Iteration Review (Team Demo)

Aspect	Details
Purpose	Demonstrate completed work and gather feedback
Duration	1-2 hours for a 2-week iteration
Attendees	Team, PO, stakeholders, other teams
Focus	Working software, not slides or reports

Iteration Retrospective

- **Purpose:** Inspect and adapt team process and practices
- **Duration:** 1-2 hours, immediately after review
- **Attendees:** Team members and Scrum Master only
- **Output:** 1-2 improvement items for next iteration

Retrospective Formats

Start/Stop/Continue

Mad/Sad/Glad

What to start doing, stop doing, continue doing

Emotional reflection on the iteration

4Ls

Liked, Learned, Lacked, Longed for

Sailboat

Wind (helps), Anchors (slows), Rocks (risks)

Exam Focus Areas

- Retros are safe spaces - no blame, focus on improvement
- Limit improvement items to 1-2 per iteration
- Track improvement items in the backlog
- Vary retro format to keep it fresh and engaging

PI Planning

PI Planning Event

The heartbeat of the Agile Release Train

PI Planning Basics

Aspect	Details
Duration	2 full days (8 hours each)
Frequency	Every PI (8-12 weeks)
Attendees	All ART members (50-125 people)
Location	Face-to-face preferred, co-located or distributed

Aspect	Details
Facilitated by	Release Train Engineer (RTE)

PI Planning Inputs

- **Business Context:** Strategic themes, business goals, market conditions
- **Product Vision:** Product roadmap and upcoming features
- **Architecture Vision:** Technical enablers and architecture runway
- **Top Features:** Prioritized features from Program Backlog
- **Team Capacity:** Available capacity considering holidays/PTO

PI Planning Outputs

PI Objectives

Business and technical goals for each team

Program Board

Visual representation of dependencies

ROAM Board

Risks categorized and assigned

Iteration Plans

High-level plan for each iteration

⚠ Exam Focus Areas

- PI Planning is essential and cannot be skipped
- All teams must participate - no representatives
- Face-to-face planning is strongly preferred
- Business Owners assign Business Value to objectives

PI Planning Day 1

Setting context and team planning

Day 1 Agenda

Time	Activity	Who
1 hour	Business Context	Business Owners, Executives
30 min	Product/Solution Vision	Product Management
30 min	Architecture Vision & Development Practices	System Architect
30 min	Planning Context & Lunch	RTE
3-4 hours	Team Breakouts #1	Teams
1 hour	Draft Plan Review	All ART members
30 min	Management Review & Problem Solving	Leadership

Team Breakout Activities

- **Review features:** Understand assigned features and enablers
- **Write stories:** Break features into user stories
- **Identify dependencies:** Find and negotiate cross-team dependencies
- **Draft objectives:** Create draft PI Objectives
- **Plan iterations:** Assign stories to iterations

Draft Plan Review

- **Each team presents:** Draft objectives and risks (2-3 min each)
- **Dependencies shared:** Teams announce dependencies on program board
- **Risks identified:** Surfaced risks added to ROAM board
- **Feedback gathered:** Questions and concerns noted

Exam Focus Areas

- Business context sets the 'why' for the PI
- Teams must be empowered to plan their own work
- Dependencies should be visible on the program board
- Management problem-solving happens after Day 1

PI Planning Day 2

Final planning, commitment, and confidence vote

Day 2 Agenda

Time	Activity	Who
30 min	Planning Adjustments	RTE
3 hours	Team Breakouts #2	Teams
1 hour	Final Plan Review & Lunch	All ART members
30 min	ART Risks	RTE, Teams
30 min	Confidence Vote	All team members
30 min	Plan Rework (if needed)	Teams
30 min	Planning Retrospective	All ART members

Confidence Vote

Vote	Meaning	Action
5 fingers	High confidence, no concerns	Ready to commit
4 fingers	Confident with minor concerns	Ready to commit
3 fingers	Some concerns but committed	Commit with reservations
2 fingers	Significant concerns	Discussion needed
1 finger	Cannot commit	Must discuss and rework

ROAM Board

Resolved

Risk is addressed, no longer a concern

Owned

Someone takes responsibility for the risk

Accepted

Risk is acknowledged and will be managed

Mitigated

Actions taken to reduce impact/likelihood

⚠ Exam Focus Areas

- Average confidence should be 3 or higher to proceed
- Low votes (1-2) must be discussed and addressed
- Commitment is to objectives, not specific features
- Planning retro improves the next PI Planning event

Cross-Team Collaboration

Cross-Team Coordination

Keeping multiple teams aligned and synchronized

Scrum of Scrums (SoS)

Aspect	Details
Purpose	Coordinate dependencies and impediments across teams
Frequency	2-3 times per week (or as needed)
Attendees	Scrum Masters or team representatives
Duration	15-30 minutes
Facilitated by	Release Train Engineer

SoS Discussion Topics

- **Progress:** What did our team accomplish since last SoS?
- **Plans:** What will our team do before next SoS?
- **Impediments:** What blockers need help from other teams?
- **Dependencies:** Are dependencies on track? Any risks?

PO Sync

Purpose

Align Product Owners on scope, priorities, and dependencies

Frequency

Weekly or as needed

Attendees

Product Owners, Product Management

Topics

Backlog changes, scope adjustments, acceptance criteria

Exam Focus Areas

- SoS is about coordination, not status reporting
- RTE facilitates but teams drive the discussion
- Keep sync events short and focused
- Escalate blockers that can't be resolved at SoS

System Demo

Demonstrating integrated value at the ART level

System Demo Overview

Aspect	Details
Purpose	Show integrated increment from all teams
Frequency	End of every iteration
Duration	1-2 hours
Attendees	All ART members, Business Owners, stakeholders
Environment	Staging or integration environment

System Demo vs Team Demo

Team Demo	System Demo
Shows team's completed work	Shows integrated work from ALL teams
Team-level stakeholders	ART-level stakeholders and Business Owners

Team Demo	System Demo
Any team environment	Integration/staging environment
Held during iteration	Held at end of iteration

System Demo Best Practices

- **Prepare:** Ensure integration is complete before demo
- **Focus:** Demo working software, not slides
- **Feedback:** Actively seek stakeholder input
- **Time-box:** Respect scheduled duration
- **Celebrate:** Recognize team accomplishments

Exam Focus Areas

- System Demo is the primary measure of ART progress
- Demo must show integrated, working software
- Stakeholder feedback drives backlog refinement
- Failed integration = no System Demo (red flag)

Managing Dependencies

Identifying and resolving cross-team dependencies

Dependency Types

Type	Description	Example
Intra-ART	Between teams in same ART	Team A needs API from Team B

Type	Description	Example
Inter-ART	Between different ARTs	ART 1 needs component from ART 2
External	With teams outside SAFe	Need data from legacy system team

Program Board

- **Purpose:** Visualize dependencies and milestones across teams
- **Format:** Grid with teams as rows, iterations as columns
- **Features:** Features placed in iteration when delivered
- **Dependencies:** Strings/lines connect dependent items
- **Milestones:** Key dates and external commitments marked

Resolving Dependencies

Negotiate

Teams discuss and agree on delivery timing

Reorder

Adjust iteration plans to accommodate dependencies

Escalate

RTE helps when teams cannot agree

Accept Risk

Move forward with contingency plan

⚠ Exam Focus Areas

- Dependencies are normal - minimize but don't eliminate
- Program board is updated throughout the PI
- Red string = risky dependency (needs attention)
- Best dependency is no dependency (architecture matters)

Lean-Agile Principles

SAFe Core Values

The foundational values that guide SAFe behavior

The Four Core Values

Value	Description	Behaviors
Alignment	Everyone understands and works toward common goals	Clear vision, PI Objectives, sync events
Built-in Quality	Quality is not added later, it's built from the start	TDD, CI/CD, automation, pairing
Transparency	Trust-based environment with open information	Visible work, honest metrics, open communication
Program Execution	Focus on working systems that deliver value	Iteration delivery, System Demo, I&A

Alignment in Practice

- **Strategic Themes:** Connect daily work to business strategy
- **PI Objectives:** Clear goals that teams commit to
- **Shared understanding:** Everyone knows the 'why'
- **Common cadence:** Teams synchronized on same rhythm

Exam Focus Areas

- Core Values guide behaviors and decisions
- Built-in Quality is not negotiable in SAFe
- Transparency builds trust; trust enables speed

- Program Execution = working software delivered regularly

Built-In Quality

Quality practices at every level

Quality Practices

Practice	Description
Test-Driven Development (TDD)	Write tests before code
Behavior-Driven Development (BDD)	Define behavior with examples
Continuous Integration	Integrate code frequently (daily or more)
Pair Programming	Two developers work together
Collective Code Ownership	Anyone can modify any code
Refactoring	Continuously improve code structure

Definition of Done (DoD)

- **Team-level DoD:** Criteria for story completion
- **ART-level DoD:** Criteria for feature integration
- **Solution-level DoD:** Criteria for release readiness
- **Evolving:** DoD improves over time as capability grows

Architecture Runway

Definition

Existing code, components, and infrastructure that support

Purpose

Enable fast feature delivery without major rework

upcoming features

Maintained by

Enablers in the backlog (technical and infrastructure)

Balance

Not too far ahead, not falling behind

Exam Focus Areas

- You can't scale crappy code (quality first)
- DoD is non-negotiable - not done until DoD met
- Technical debt must be addressed continuously
- Architecture runway enables business agility

Inspect & Adapt (I&A)

The ART's continuous improvement event

I&A Overview

Aspect	Details
When	End of each Program Increment
Duration	3-4 hours (half day)
Attendees	All ART members, Business Owners
Purpose	Reflect on PI, identify improvements, solve problems

I&A Components

- **1. PI System Demo:** Final integrated demo of the entire PI

- **2. Quantitative Review:** Review metrics and objectives achievement
- **3. Retrospective:** Reflect on what went well and what didn't
- **4. Problem-Solving Workshop:** Root cause analysis and improvement identification

Problem-Solving Workshop

Agree on Problem

Select 1-2 significant problems to solve

Root Cause Analysis

Use 5 Whys or fishbone diagram

Identify Solutions

Brainstorm countermeasures

Create Improvement Items

Add to backlog for next PI

⚠ Exam Focus Areas

- I&A is mandatory - the ART's biggest retrospective
- Quantitative review uses objective data, not opinions
- Problem-solving focuses on systemic issues, not symptoms
- Improvement stories go into next PI's backlog