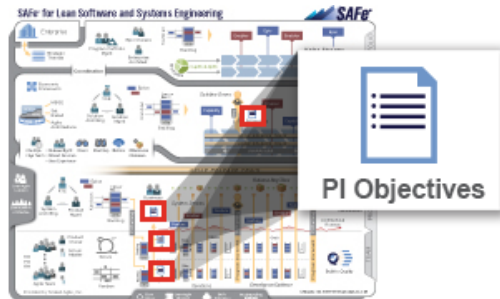


[Twitter \(https://twitter.com/ScaledAgile\)](https://twitter.com/ScaledAgile)
[LinkedIn \(https://www.linkedin.com/grps/Scaled-Agile-Framework-4189072?\)](https://www.linkedin.com/grps/Scaled-Agile-Framework-4189072?)

[YouTube \(https://www.youtube.com/user/scaledagile\)](https://www.youtube.com/user/scaledagile)
[Slideshare \(http://www.slideshare.net/ScaledAgile\)](https://www.slideshare.net/ScaledAgile)



(/)

|| *Making and meeting small commitments builds trust. Ba must be energized with its own intentions . . .*

—Nonaka and Takeuchi, *The Knowledge-Creating Company*

PI Objectives Abstract

PI Objectives are a summarized description of the specific business and technical goals that an Agile Team, ART, or Value Stream intends to achieve in the upcoming Program Increment. Their main purpose is to validate understanding of business and technical intent, focus alignment on outcomes rather than process or tactical concerns, and summarize data into meaningful information that enhances alignment and provides visibility for all.

PI objectives are formulated during PI Planning and, where applicable, Post-PI Planning. During PI planning, each team reviews the Vision and any other input objectives, defines initial Stories, and plans them into Iterations until their capacity is full. Teams then reflect on the iteration plans and synthesize and summarize the specific technical and business objectives for their team for that PI. The aggregation of each team's objectives becomes the *Program PI Objectives*, which are approved by the Business Owners. If Value Stream Level PI planning is needed, then the programs' PI objectives are synthesized and aggregated and become the *Value Stream PI Objectives*.

In this way, all stakeholders know what to expect from each team, each ART, and the value stream as a whole; all WIP is visible. Each team executes against a current, known, aligned, and definitive set of feasible, agreed-upon objectives based on plans that are created by, *not for*, the teams.

At the end of the PI, each team, each ART, and the value stream assess their results against their objectives and use this input to continuously improve performance.

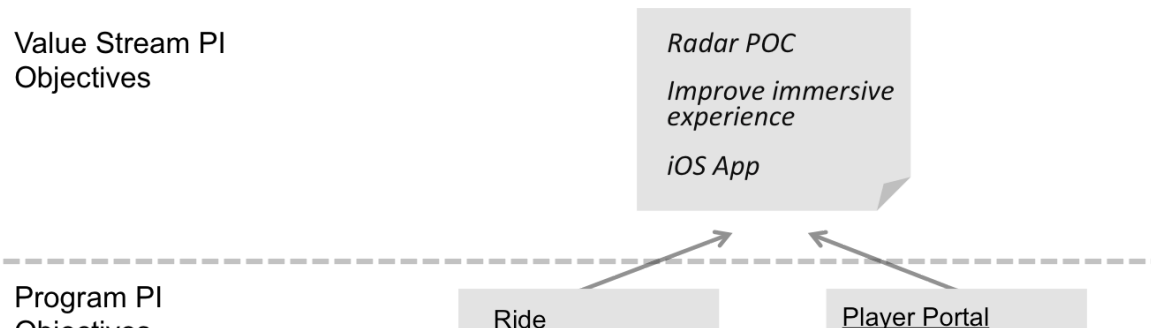
(Note: For more on this important topic, see the associated Guidance (/the-role-of-pi-objectives/) article.)

Details

SAFe is commitment based, in that it relies on a rolling series of short-term commitments from the Agile Teams (/agile-teams/), Agile Release Trains (ARTs) (/agile-release-train/), and Value Streams (/value-streams/) to assist with meaningful business planning and outcomes. This is a key element of trust that must exist between development and the business stakeholders. However, this shouldn't be confused with committing to a set of fixed, long-term, waterfall-like deliverables (as explained in the Solution Intent (/solution-intent/) article). But in order for the business to do any meaningful planning, it depends on Solution (/solution/) builders for some amount of reliable, predictable forecasting. Too little, and it's "those ARTs can't commit to anything useful." Too much, and it's "those ARTs never do what they say they will." Neither is optimum, as both increase the distrust between business and development. That significantly hinders ultimate business success, not to mention the joy of work.

We need something in between, and that is a primary purpose of the *Team, Program, and Value Stream PI Objectives*. In addition to alignment, the process of feasible objective-setting is integral to reducing the excess work in process in the system.

SAFe PI objectives and plans are built bottom up, by Agile Teams who estimate and plan their own part of the solution. The team creates team PI objectives at the PI Planning (/pi-planning/) meeting, indicating what they will have ready by the end of the Program Increment (/program-increment/). The objectives created by the team are aggregated up to the Program Level (/program-level/) and then aggregated again to the Value Stream Level (/value-stream-level/), as can be seen in Figure 1.



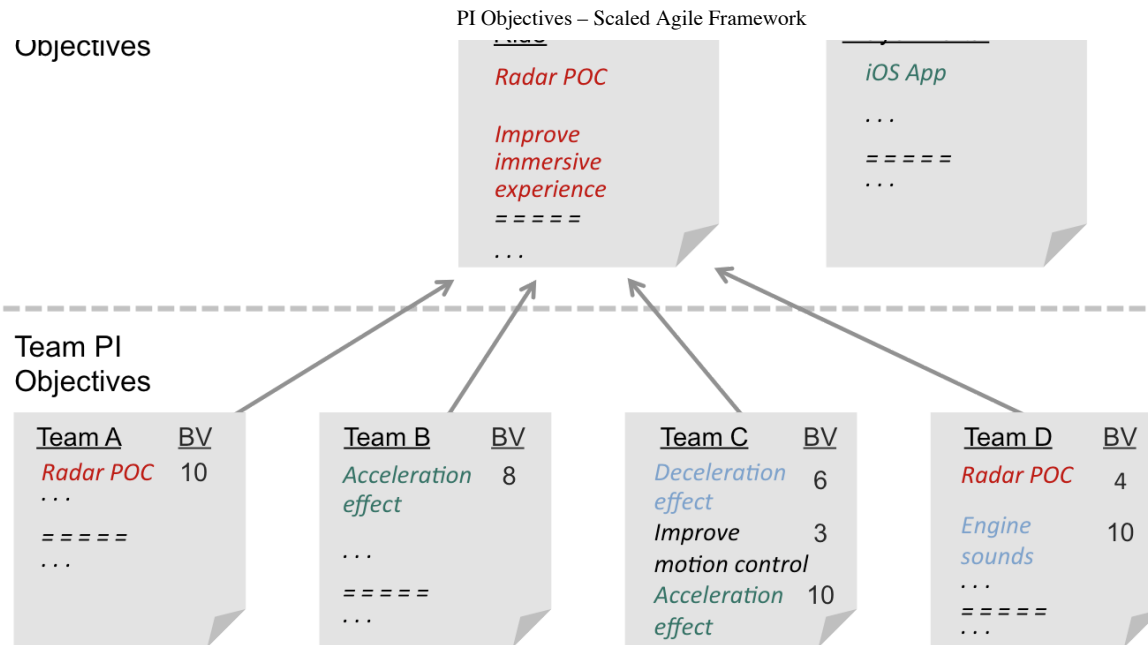


Figure 1. From team to program to value stream PI objectives

Building the Team PI Objectives

During PI planning, teams set team PI objectives, which provide many benefits:

- Provide a common language for communication between business and technology
- Align teams to each other and to a common mission
- Create the near-term vision around which teams can rally and develop as a team
- Provide an important Metric (/metrics/), the Program Predictability Measure (/metrics/#P2), that the team and Agile Release Train can use to improve performance
- Communicate with management and highlight each team's contribution to business value
- Expose dependencies between teams that must be addressed for system success

Setting team PI objectives is not a trivial thing, as it requires solid estimating and planning, a well-understood velocity, analysis of upcoming Features (/features-and-capabilities/), defining Stories (/story/) for the Team Backlog (/team-backlog/), and, finally, synthesis into simple business terms that can be understood by everyone. The net result, however, is straightforward and readily represented, as the example in Figure 2 illustrates.

Objectives For PI 4

- Decide buy/make engine noises

- Proof of concept with mock sounds
- Help with Radar POC

Figure 2. An example set of a Baja Ride team's initial PI objectives

During PI planning, the teams look at the Program Vision (/vision/) and new features and plan the stories they need to deliver. In so doing, they also identify their specific *team PI objectives*.

Differentiate Between Features and Objectives

The team's PI objectives often relate directly to intended features; indeed, many are the same. However, the mapping is not always straightforward, since some features require the collaboration of multiple teams, as Figure 3 illustrates.

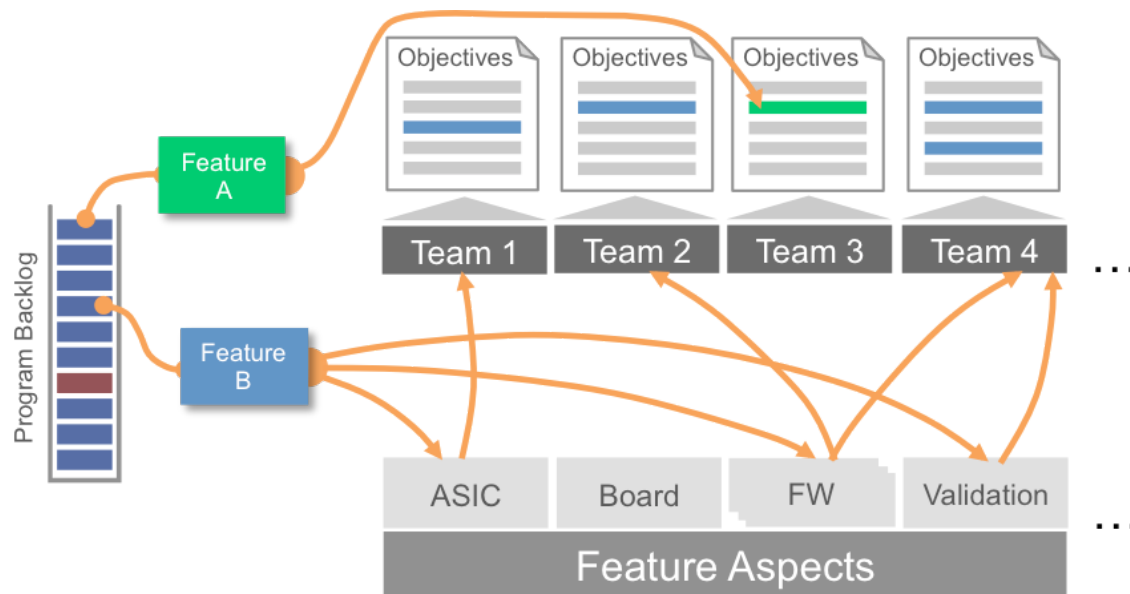


Figure 3. From features to objectives; some features will appear on more than one team's objectives

Note that some features (such as Feature A) can be delivered by individual teams; others (Feature B) require collaboration. In addition to features and inputs to features, other team objectives will appear as well. These can include technical objectives (for example, the proof of concept in Figure 2) that enable future features, enhancements to development infrastructure, Milestones (/milestones/), and more. All the results of this process are captured in the affected team's objectives.

PI objectives help teams shift focus off the feature language and onto the desired business outcomes. Features and acceptance criteria are excellent tools to help understand, capture, and collaborate around the work that needs to be done to iterate the solution

criteria are excellent tools to help understand, capture, and collaborate around the work that needs to be done to iterate the solution to the next level, but it's all too easy to get caught up in "finishing the features" and missing the overall goals hiding inside them. The core question becomes, "Is our goal to complete the listed features, or is our goal to provide the outcomes desired by those features? In other words, if we could provide the same value with half the amount of work, and without building all of the features, would this be acceptable?"

Better understanding of the intent offered by direct conversations with the Business Owners (/business-owners/) often results in the teams providing new perspectives to System Architects/Engineering (/system-and-solution-architect-engineering/) and Product Management (/product-and-solution-management/) and quickly finding ways to apply their expertise more effectively.

Use Stretch Objectives

Stretch objectives provide a way to help ensure that the delivery timebox will be met (synchronizing to a delivery cadence requires capacity margins [2]). Teams *commit to deliver their non-stretch objectives*. In addition, teams also *agree to do their best to deliver the stretch objectives*, and they *are included* in the capacity for the PI. Realistically, however, those may or may not be achieved in the timebox, due to timing or other uncertainties. As these objectives might not be finished in the PI, stakeholders plan accordingly. Stretch objectives provide a number of benefits:

- Improved economics. Without them, if the team must commit to a 100% scope in a fixed timebox, they are forced to trade off quality or build other buffers into the system. Buffers convert *uncertain* "might-have-been" scope to *certain* but "less-than-what-might-have-been" scope, resulting in less overall throughput.
- Increased reliability. Stretch objectives provide an estimating error allowance, thereby increasing confidence in delivery of the main priorities. In turn, delivering commitments (stretch objectives are not committed objectives) is the most important factor in *building trust* between the teams and the stakeholders.
- Adaptability to change. In order to reliably deliver on a cadence, stretch objectives provide the capacity needed to meet commitments yet alter priorities if necessary when fact patterns change.

Typically, the total allowance for stretch objectives is 10% to 15% of the total capacity. And one must constantly keep in mind that stretch objectives *are* used to identify what *can* be variable within the scope of a plan; stretch objectives *are not* the way for stakeholders to load the teams with more than they can possibly do.

Write SMART Objectives

Team PI objectives serve as a brief summary of a team's plan for the PI. However, the fact that the objectives are at the higher level of abstraction means that they may tend toward fuzzy and non-verifiable "chunks of intent." To address this, teams use SMART objectives. Each objective is written in such a way that it is:

- **Specific.** States the intended outcome as simply, concisely, and explicitly as possible. (Hint: Try starting with an action verb.)
- **Measurable.** It should be clear what a team needs to do to achieve the objective. The measures may be descriptive, yes/no, quantitative, or provide a range.

- **Achievable.** Achieving the objective should be within the team's control and influence.
- **Realistic.** Recognize factors that cannot be controlled. (Hint: Avoid making "happy path" assumptions.)
- **Time-bound.** The time period for achievement must be within the PI, and therefore all objectives must be scoped appropriately.

Communicate Business Value with Objectives

As objectives are finalized during PI planning, Business Owners assign business value to each of the teams' individual objectives in face-to-face conversation with the teams. The value of this particular conversation, from business-to-team and team-to-business, cannot be overstated, as it communicates the strategy and context behind these weighting decisions. Each objective is ranked by the Business Owners on a scale of 1 to 10. Business value should not be confused with any other measures, such as the associated effort, total story points associated with an objective, etc. Business value is assigned, not calculated, and serves as an input to execution considerations. Many of the team's objectives provide direct and *immediate* value to the solution. Others, such as Enablers (/enablers/), advances in infrastructure, development environments, and quality initiatives, enable the faster creation of *future* business value. All of these factors must be weighed in the final balance.

Finalize the Team PI Objectives

When objectives have been made "smarter," stretch objectives have been identified, and business value has been established, then the objectives in Figure 2 might evolve to look like those in Figure 4.

<u>Objectives For PI 4</u>	<u>BV</u>
- Proof of concept with mock sounds	10
- Help with Radar POC	4
- Decide buy/make engine noises	3
===== Stretch Objectives =====	=====
- Proof of concept with real sounds	7

Figure 4. Objective sheet with business value and stretch objectives

Commit to PI Objectives

Near the end of PI planning, once objectives have been agreed to, some scope margin has been provided by stretch objectives, and risks have been addressed, the teams hold a vote of confidence on their ability to meet the objectives. While, strictly speaking, a vote of confidence is not the same as a commitment, they are treated as about the same thing. Therefore, this commitment has to be a reasonable thing to ask for. In addition to the Agile prima facie case that a commitment has to *come from* the teams, not be *given for* the teams, a team's SAFe commitment has two parts:

the teams, a team's SAFe commitment has two parts:

1. Teams agree to do everything in their power to meet the committed objectives
2. If, during the course of the PI, facts dictate that some objectives are simply not achievable, then the teams agree to escalate *immediately* so that corrective action can be taken

In this way, all stakeholders know that either: a) the program results will be achieved as planned, or b) they will be provided sufficient notice so as to be able to mitigate and take corrective action, thereby minimizing business disruption. That's about as good as it gets, because this is, after all, *research* and development.

Creating Program and Value Stream Objectives

The result of the PI planning process will be some number of approved objectives sheets, one per team. Teams vote on the confidence level for the objectives as a set, and if confidence is high enough, the aggregate set of objectives becomes the committed ART plan. (If not, planning continues until it is.) This is all the input the Release Train Engineer (/release-train-engineer-and-value-stream-engineer/) needs, and he or she uses that team data to roll up the team objectives into the program PI objectives in a format suitable for management communication.

These objectives should be SMART, much like the team PI objectives, and have stretch objectives. Also, like the team PI objectives, these might be business Capabilities (/features-and-capabilities/) the ART is working on, enablers, or other business or technical goals.

During the Post-PI Planning (/pre-and-post-pi-planning/) meeting, after all the ARTs have planned, objectives are further rolled up to the value stream level by the Value Stream Engineer (/release-train-engineer-and-value-stream-engineer/), and the value stream PI objectives are synthesized and summarized. This is the top level of PI objectives in SAFe, and they communicate to stakeholders what the value stream as a whole will deliver in the upcoming PI. Figure 1 (above) illustrates this aggregation from team to program to value stream PI objectives.

It is important to note that only team PI objectives have business value attached to them. This value is not rolled up to the other levels. When calculating the predictability measure for programs and value streams, the metric itself is rolled up to ascertain predictability at a higher level.

Shed Excess WIP with Realistic Objectives

During review of the team PI objectives, it will typically become obvious that not everything that was envisioned by the various business stakeholders will likely be achieved in the PI timebox. Therefore, in order to gain agreement, some of the current in-flight development work (WIP) will need to be reevaluated. This happens throughout the PI planning process but is crystallized in the final agreement among *all* the stakeholders to the program PI objectives. Those lower-priority work items get moved back into the Program Backlog (/program-and-value-stream-backlogs/) (a lower-cost holding pattern, also called “not doing this now”). Decreasing excess WIP reduces overhead and thrashing, and it increases productivity and velocity. The net result is a *feasible* set of objectives that are

agreed to by all business stakeholders and team members, as well as increased efficiency and a higher probability of delivery success. And that's something that most everyone should be able to commit to.

Planning at the value stream level can work similarly: The planning of the ARTs will impact each other, pushing some work back into the Value Stream Backlog (/program-and-value-stream-backlogs/) for reevaluation in a later PI.

Learn More

[1] Leffingwell, Dean. *Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise*. Addison-Wesley, 2011.

[2] Reinertsen, Donald. *The Principles of Product Development Flow: Second Generation Lean Product Development*. Celeritas Publishing, 2009.

Last update: 21 April 2016

The information on this page is © 2010-2016 Scaled Agile, Inc. and is protected by US and International copyright laws. Neither images nor text can be copied from this site without the express written permission of the copyright holder. Scaled Agile Framework and SAFe are registered trademarks of Scaled Agile, Inc. Please visit Permissions FAQs (<http://scaledagile.com/permission-faq/>) and contact us (<http://www.scaledagile.com/permissions-form/>) for permissions.

FRAMEWORK

[Downloads \(/posters/\)](#)

[Latest Updates \(/blog/\)](#)

[SAFe 3.0 \(http://v3.scaledagileframework.com\)](http://v3.scaledagileframework.com)

TRAINING

[Course Calendar \(http://www.scaledagileacademy.com/events/event_list.asp\)](http://www.scaledagileacademy.com/events/event_list.asp)

[About Certification \(http://www.scaledagileacademy.com/?page=WhichCertification\)](http://www.scaledagileacademy.com/?page=WhichCertification)

[Become a Trainer \(http://www.scaledagileacademy.com/?page=becomeatrainer\)](http://www.scaledagileacademy.com/?page=becomeatrainer)

PERMISSIONS

Permission FAQ (<http://www.scaledagile.com/permission-faq/>)

Permissions Form (<http://www.scaledagile.com/permissions-form/>)

Usage and Permissions (</usage-and-permissions/>)

PARTNER

Becoming a Partner (<http://www.scaledagile.com/become-a-partner/>)

Partner Directory (<http://www.scaledagile.com/listingcategory/directory/>)

Partner Event Calendar (<http://www.scaledagile.com/event-list/>)

GET SOCIAL

Twitter (<https://twitter.com/ScaledAgile>)

Linkedin (<https://www.linkedin.com/grps/Scaled-Agile-Framework-4189072?>)

YouTube (<https://www.youtube.com/user/scaledagile>)

SlideShare (<http://www.slideshare.net/ScaledAgile>)

RECENT POSTS

New Essential SAFe guidance article (</new-essential-safe-guidance-article/>)

Nov, 30th 2016

SAFe Website Updates and new Glossary (</safe-website-updates-and-new-glossary/>)

Nov, 29th 2016

End of life for SAFe 3.0 website and courseware 12/30/16 (</eol-safe3/>)

Nov, 28th 2016

SCALED AGILE, INC

CONTACT US

5480 Valmont Rd., Suite 100

Boulder, CO 80301 USA

Email: support@scaledagile.com (<mailto:support@scaledagile.com>)