

THE ART OF SOFTWARE ESTIMATION



TODAY'S SPEAKERS



Ben Friedberg
Software Architect



Luke McConnell
Product Owner

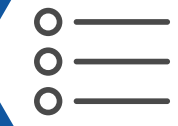
ABOUT US

We help industry-leading organizations boost speed, quality, and predictability of mission-critical software development.

-  **Web Development**
-  **App Development**
-  **Platform Development**
-  **Product Strategy**
-  **Agile Coaching**



AGENDA



Step 1

Build a Product Backlog



Step 2

Estimate the Backlog



Step 3

Estimate Velocity



Step 4

Create a Forecast

ART VS SCIENCE

ESTIMATION IS ANSWERING QUESTIONS

Product Owner: Can I have this feature done this quarter?

Customer: When will I get the new features?

Stakeholder: I just had a new idea!

Development

Flexibility

Respond to Change

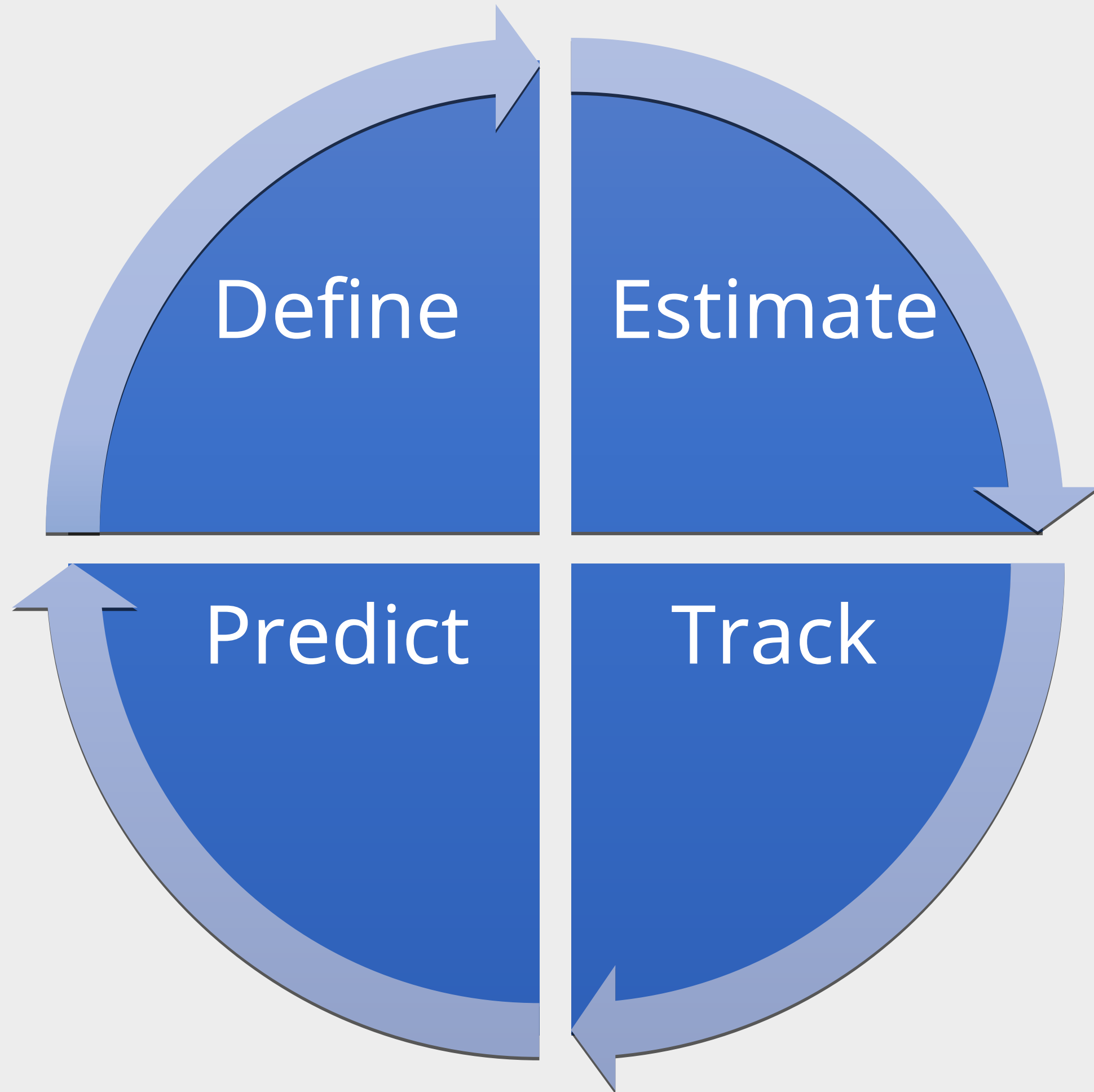
Release When Ready

Strategy

Predictability

Ability to Plan Ahead

Release on a Schedule



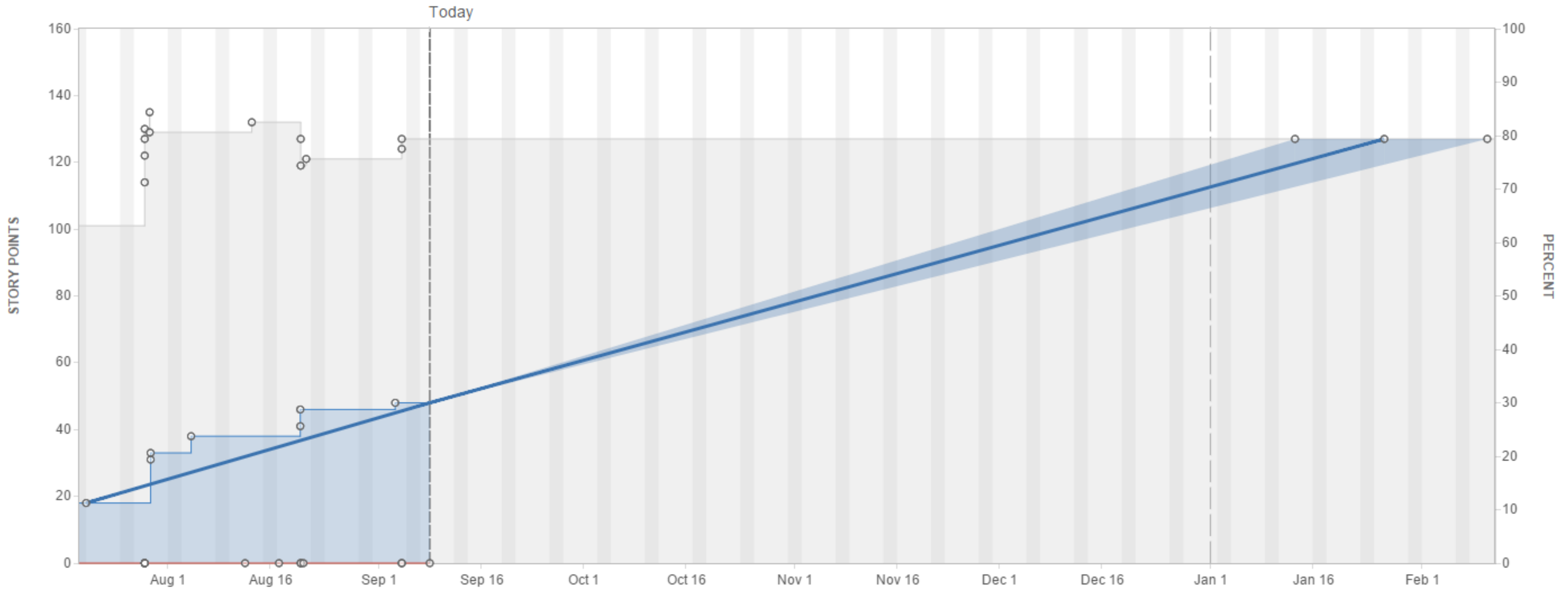
User Stories

Story Points

Velocity

Forecast

Result



Step 1

Build a Product Backlog

User Stories

Compact “requirements”

Quick and easy to write

Shared vision

Prioritized by business value

The Sentence

Who

What

As a **Shopper** I want a **gift registry** so I can share what I want with friends and family.

Why

Step 2

Estimate the Backlog

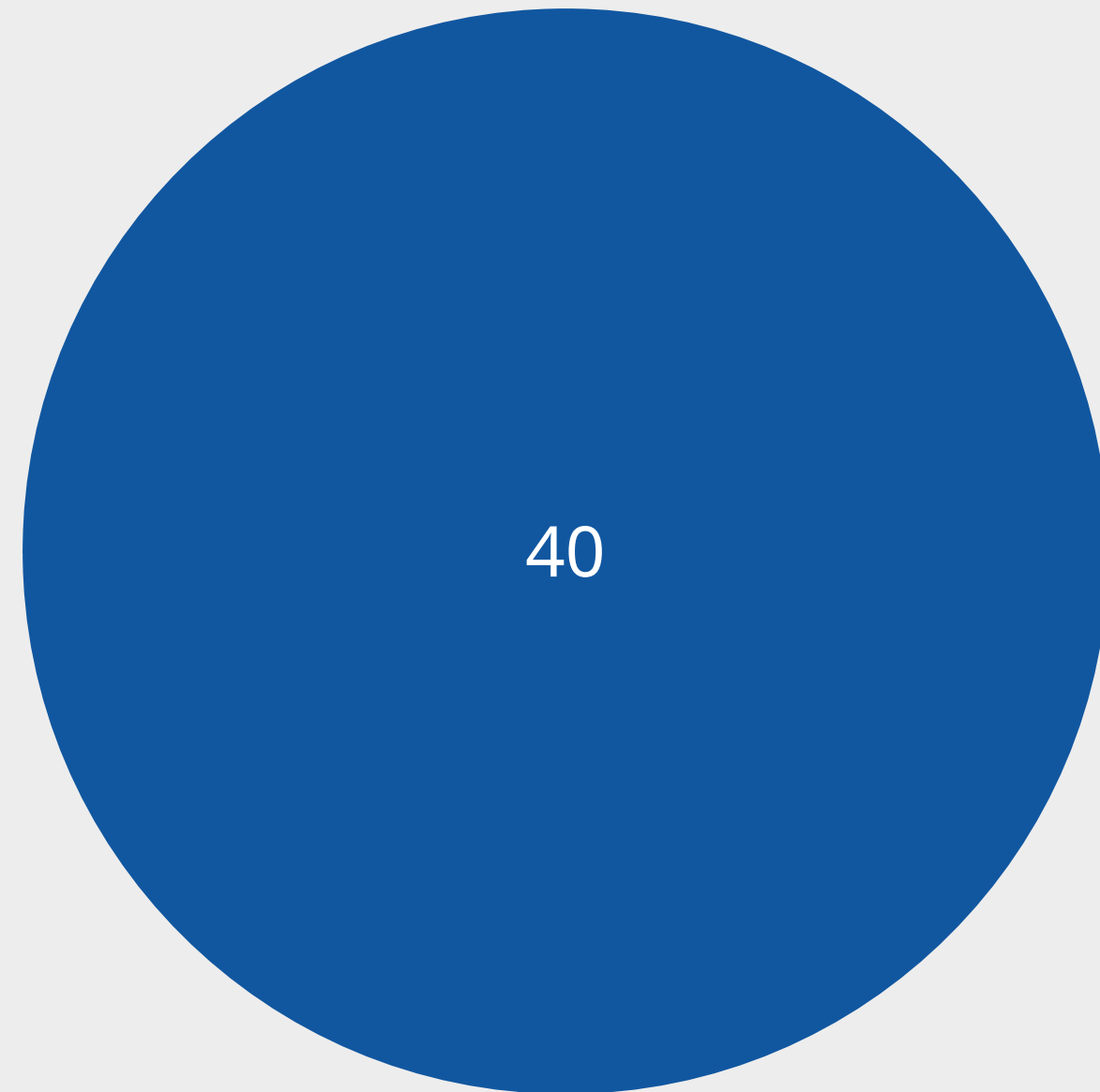
Absolute vs Relative Size

Absolute

Relative



Story Points



“Cohn Scale”

Example



8



13



20



40

Estimating Goals

Leverage team experience

Encourage discussion

Involve everyone

Make it democratic

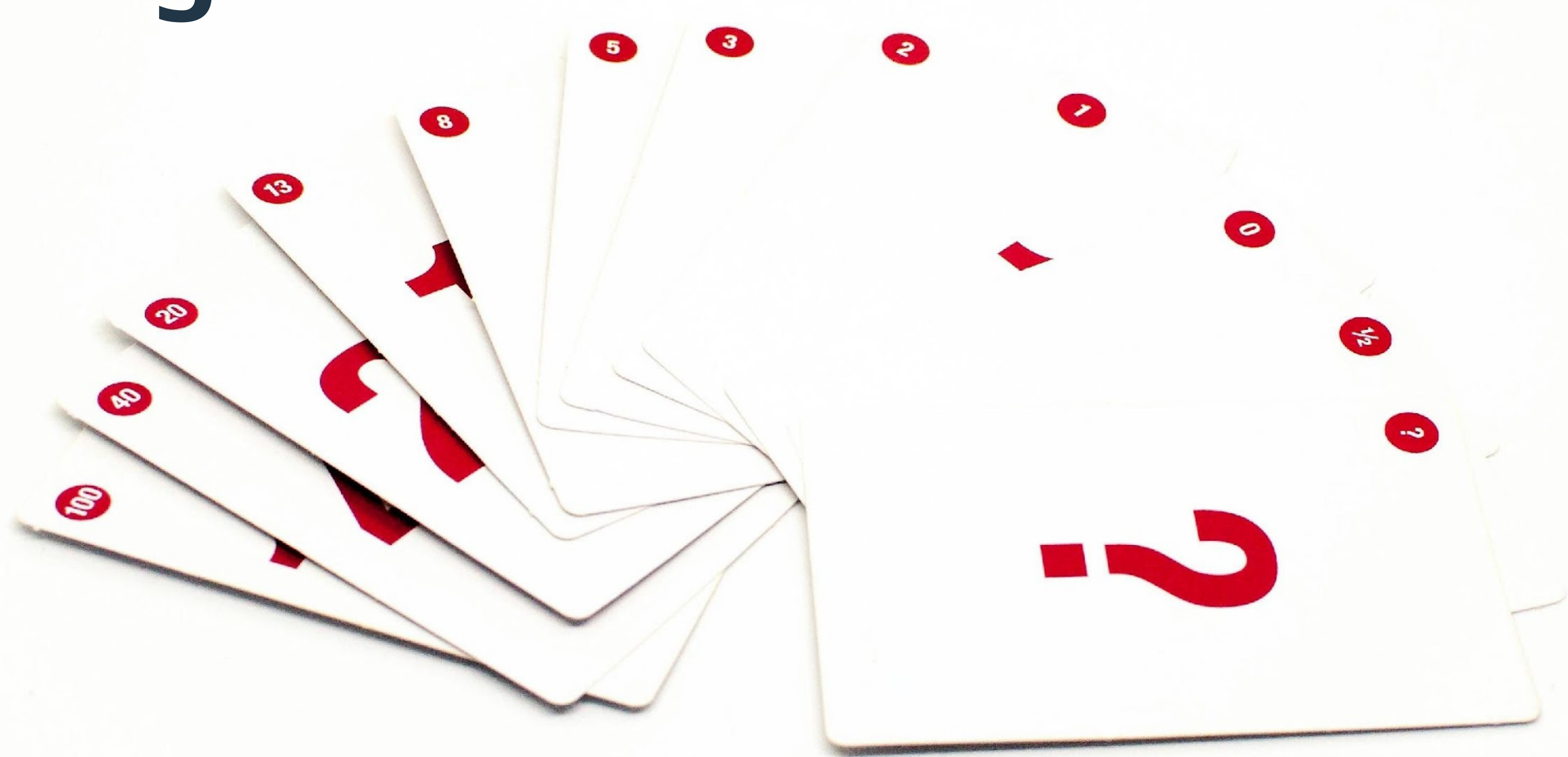
Planning Poker

Discuss

Vote

Discuss

Re-vote



Step 3

Estimate Velocity

Velocity

How quickly the team can move

Story points per sprint

Shippable user stories

**“BUT WE HAVEN’T
STARTED THE
PROJECT YET!”**



THREE OPTIONS

1. Run a
couple of
sprints

2. Compare
to a prior
project

3. Break
down some
stories

HOW MUCH TIME DO WE HAVE?

Team Capacity

Total hours per sprint
dedicated to this project

Meeting Time

Sprint planning, daily Scrums,
etc.

Net Capacity

Hours per sprint the team can
spend on tasks

What Does One Story Take?

As a shopper I want to review my cart so I can make adjustments prior to checkout.

- Write and review acceptance tests – 2 hours
- Create user interface views – 4 hours
- Write unit tests and code – 4 hours
- Run acceptance tests – 1 hour
- Fix bugs & make adjustments – 30 minutes
- Peer code review & make updates – 30 minutes
- **Total: 12 hours**

How Many Can We Complete?

As a shopper I want to review my cart so I can make adjustments prior to checkout.

5

As a shopper I want to view a list of products so I can select some to purchase.

8

As a fulfillment specialist I want to print packing labels so that I can ship packages

5

As a shopper I want to review my orders so I can see what I've purchased in the past.

3

**ESTIMATED
VELOCITY:
21**

Step 4

Create a Forecast

Calculating a Timeframe

$$\text{Duration} = \frac{\text{Story Points}}{\text{Velocity}} \times \text{Sprint Length}$$

Calculating a Timeframe

$$\text{Duration} = \frac{90}{21} \times 2 \text{ Weeks} = < 9 \text{ Weeks}$$

**ARE WE
DONE?**





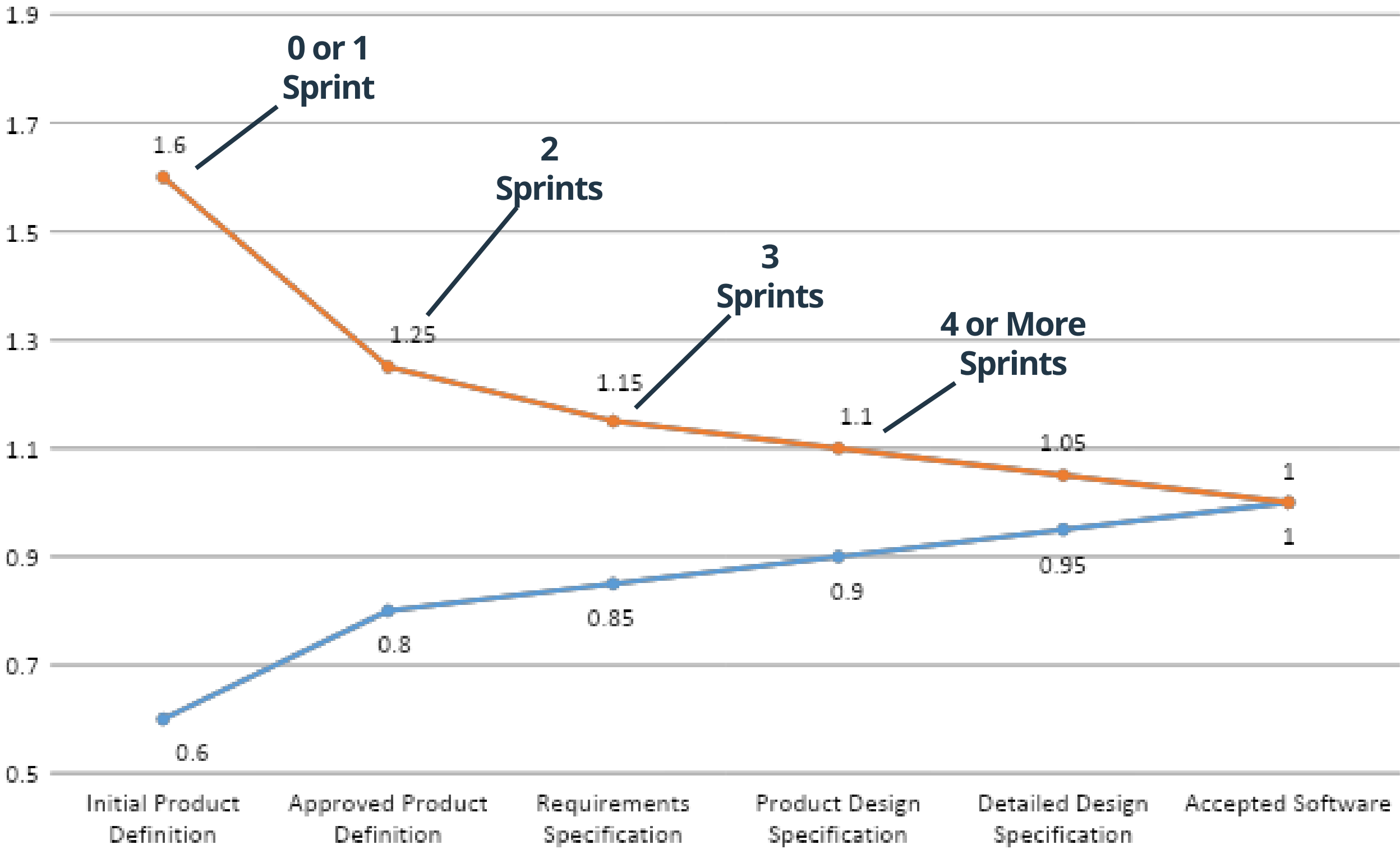
Same velocity?

No changes?

No surprises?

No chance.

Cone of Uncertainty

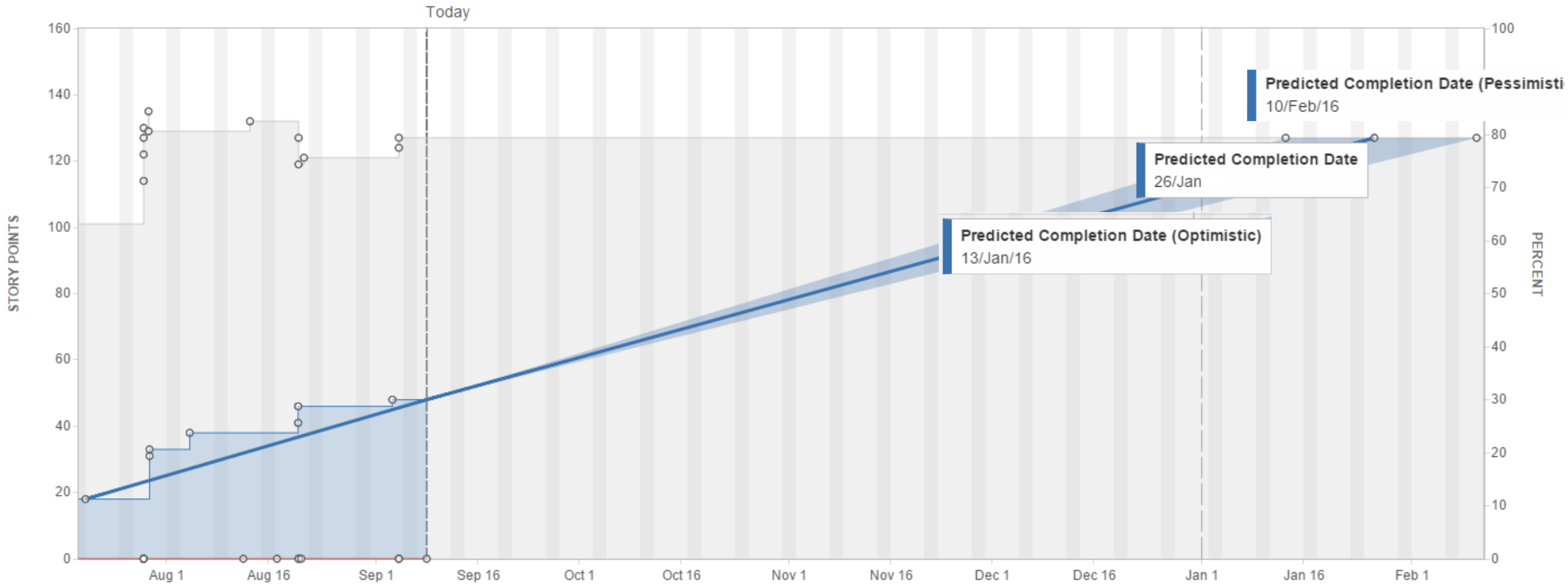


Calculating a Timeframe *Range*

$$\text{Low Duration} = \frac{90}{21 \times 0.6} \times 2 \text{ Weeks} = \sim 5 \text{ Weeks}$$

$$\text{High Duration} = \frac{90}{21 \times 1.6} \times 2 \text{ Weeks} = \sim 14 \text{ Weeks}$$

Result



QUESTIONS?
