# Building a Release Plan with Confidence Using Forecasting

Chris Shinkle - Director of Innovation
@chrisshinkle | chrisshinkle@sep.com

# 2018 Cincinnati Day of Agile & Cincy.Develop(); Sponsors

## Diamond



## Platinum



## Gold







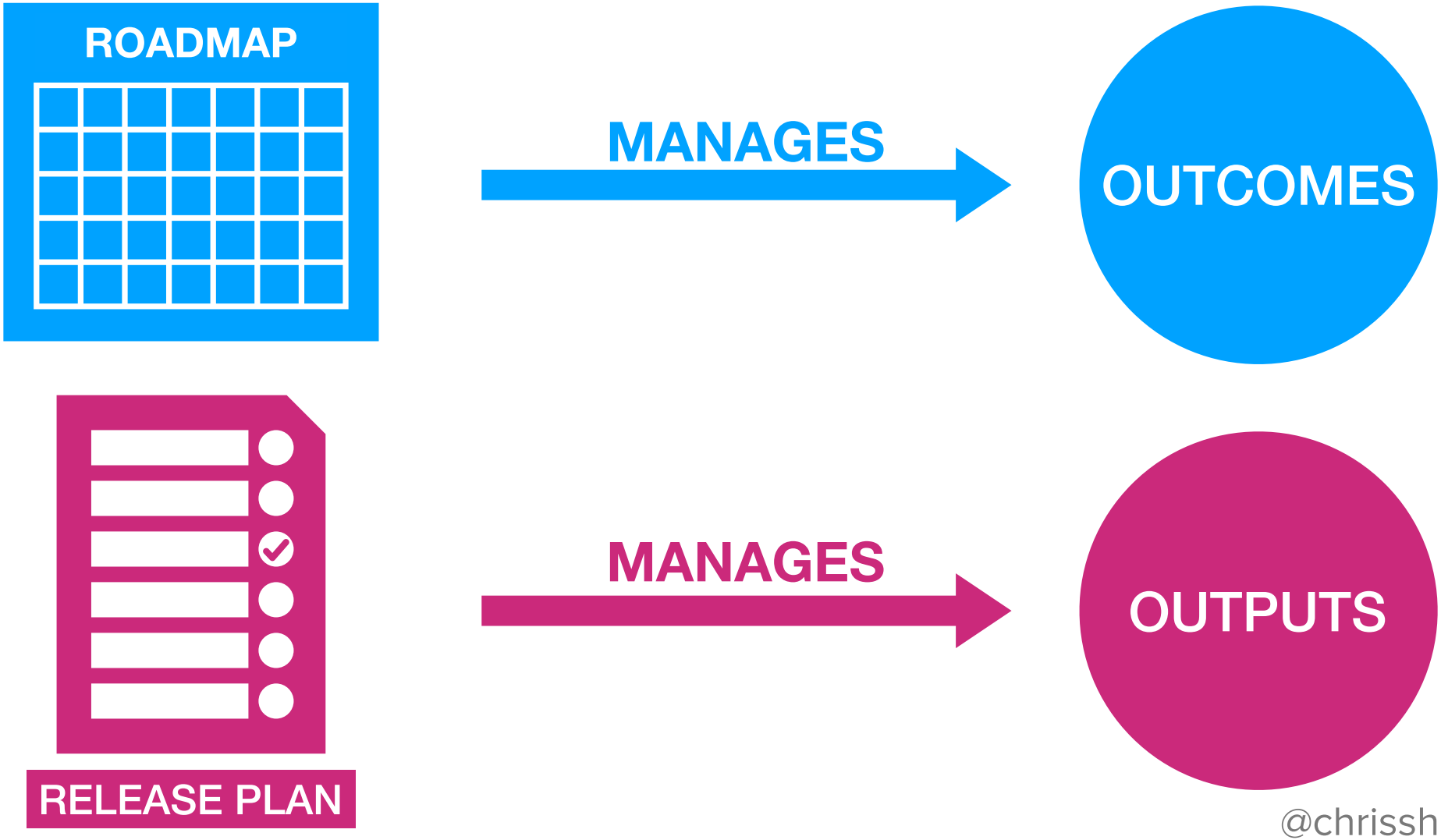

# The session has started…

1. Consider your last project or release plan…

2. Find someone you don't work with.

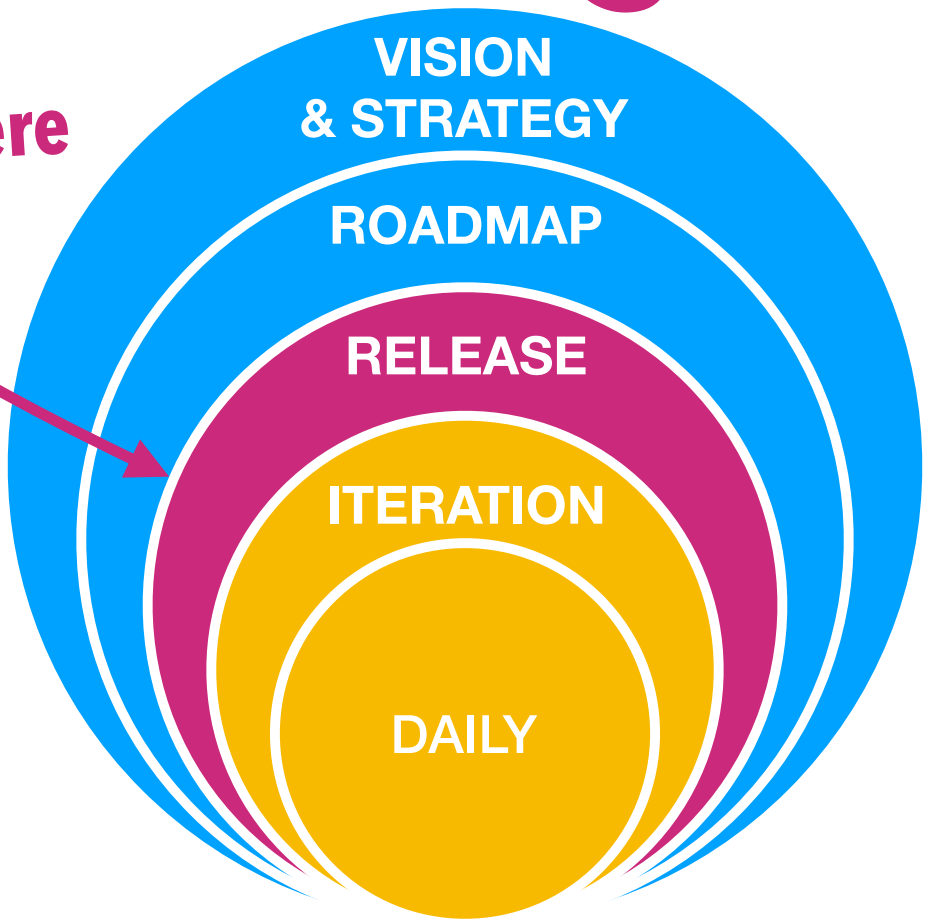
3. Share how accurate your plan was. Did you deliver everything on time?

@chrisshinkle


ROADMAP
MANAGES
OUTCOMES
RELEASE PLAN
MANAGES
OUTPUTS
@chrisshinkle

# Agile Planning Onion

We struggle here




High-level Strategic Planning

Release Planning

Short-term Planning

@chrisshinkle

# COMPONENTS OF A RELEASE PLAN

1. **WHAT** features, epics, or stories to deliver.
2. **WHEN** the Product Backlog Items will be delivered.

RELEASE PLAN COMPONENTS

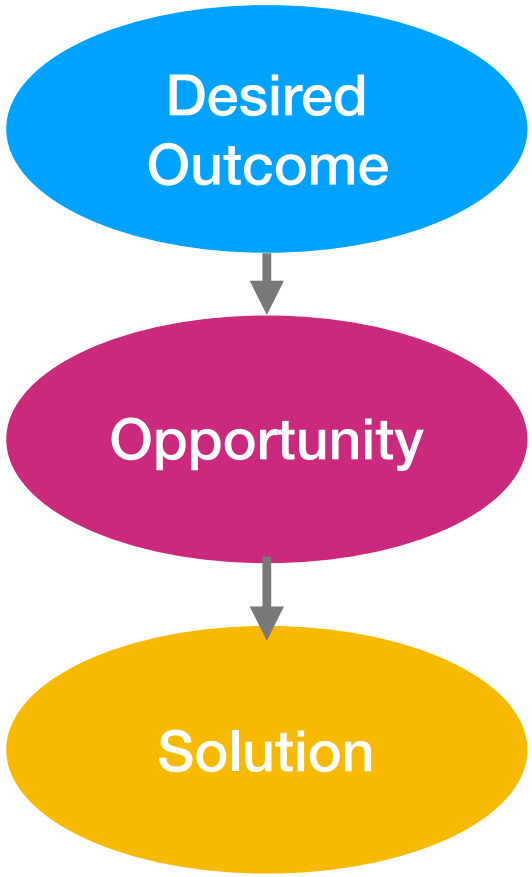
# WHAT TO DELIVER

TO PRIORITIZE FEATURES YOU

# ALIGN WITH YOUR OUTCOMES

@chrisshinkle

# OPPORTUNITY SOLUTION TREE




**Business Objective or Key Result**

**Product Outcomes**

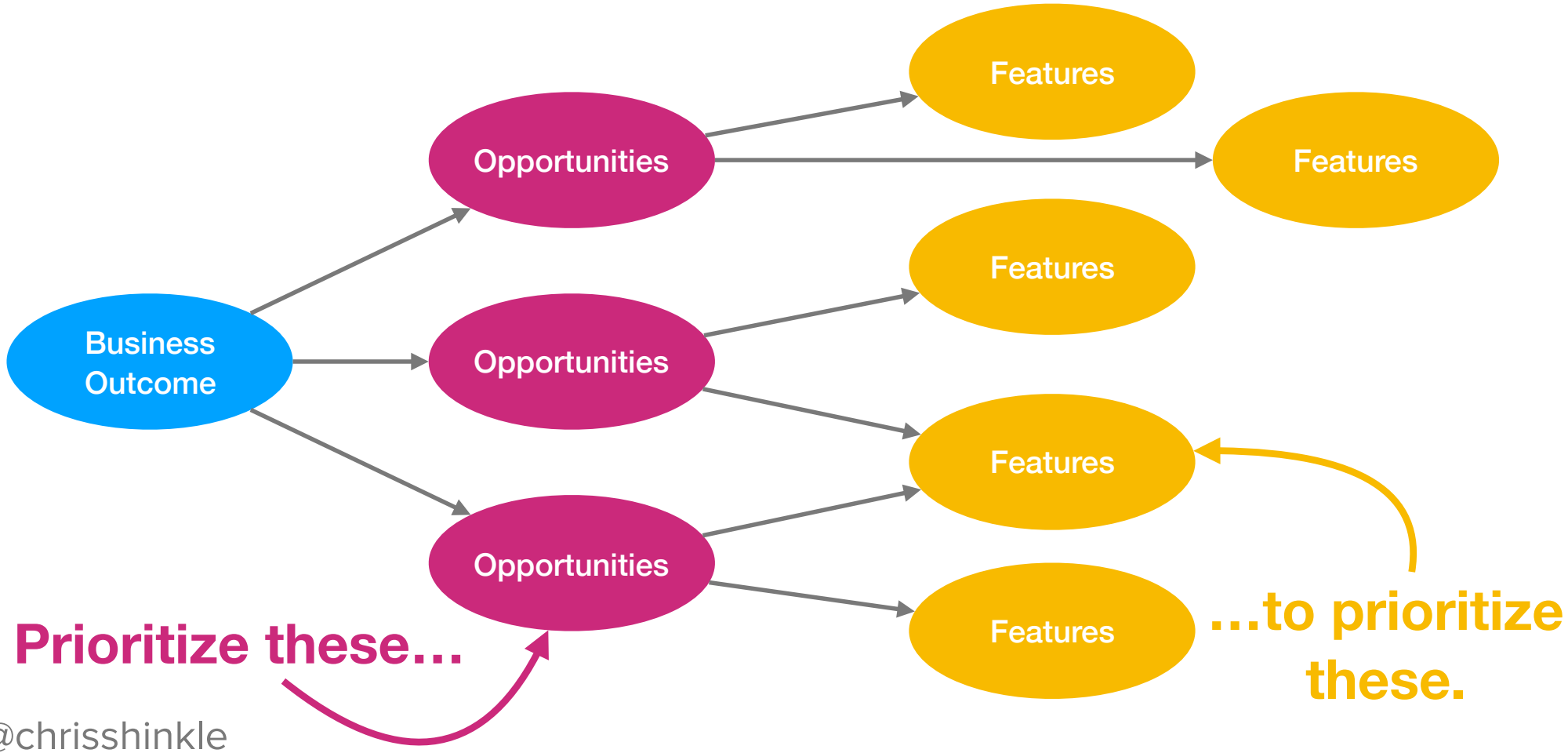
Expressed as customer needs/ problems

**Features**

Solutions to customer needs

- Teresa Torres, ProductTalk.org

# OPPORTUNITY SOLUTION TREE




@chrisshinkle

REGARDLESS OF CONTENT...

# DATES OFTEN DRIVE DECISIONS

RELEASE PLAN COMPONENTS

# WHEN CAN WE DELIVER

@chrisshinkle

$$\text{How long} = \frac{\text{Size}}{\text{Velocity}}$$

distance and pace model

How many weeks? = $\frac{300 \text{ story pts}}{15 \text{ pts per sprint}}$

Sprint = 2 weeks

How many weeks? = $\frac{300 \text{ story pts}}{[10\text{-}20] \text{ pts per sprint}}$

Sprint = 2 weeks

How many weeks? = $\frac{\text{[280-320] story pts}}{\text{[10-20] pts per sprint}}$

Sprint = 2 weeks

$$\text{How many weeks?} = \frac{[280\text{-}320]}{[10\text{-}20]} * [1 - 2]$$

Sprint = 2 weeks

$$\text{How many weeks?} = \frac{[194\text{-}251]}{[13\text{-}29]} * [1\ \text{-}1.75] * \ldots$$

Sprint = 2 weeks

$$\text{Time} = \frac{\text{Size} * \text{Growth}}{\text{Pace}}$$

# WHAT ABOUT JUST USING AVERAGES?

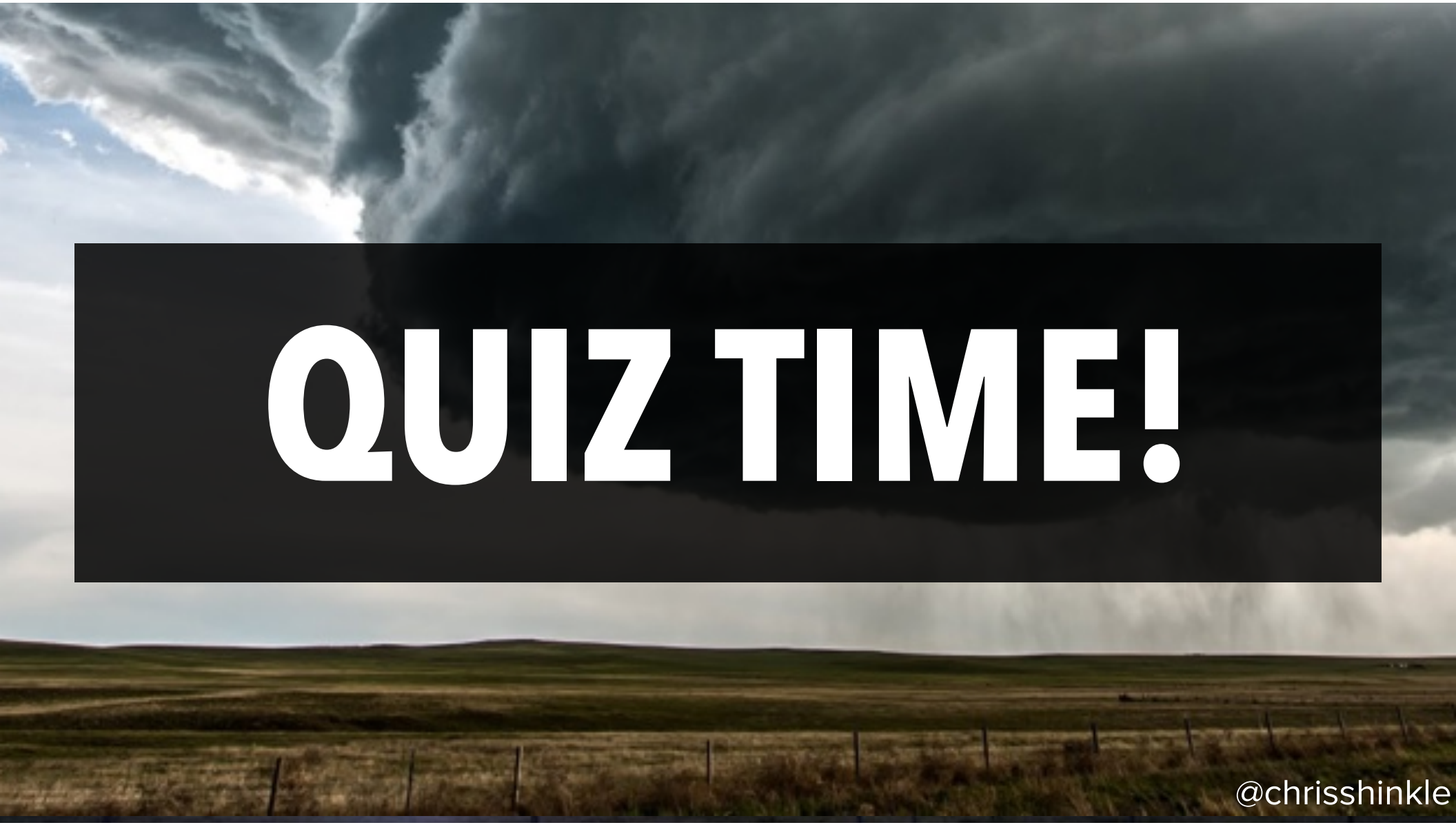
@chrisshinkle


FORECASTING

# Forecasts Must Contain 3 Things:

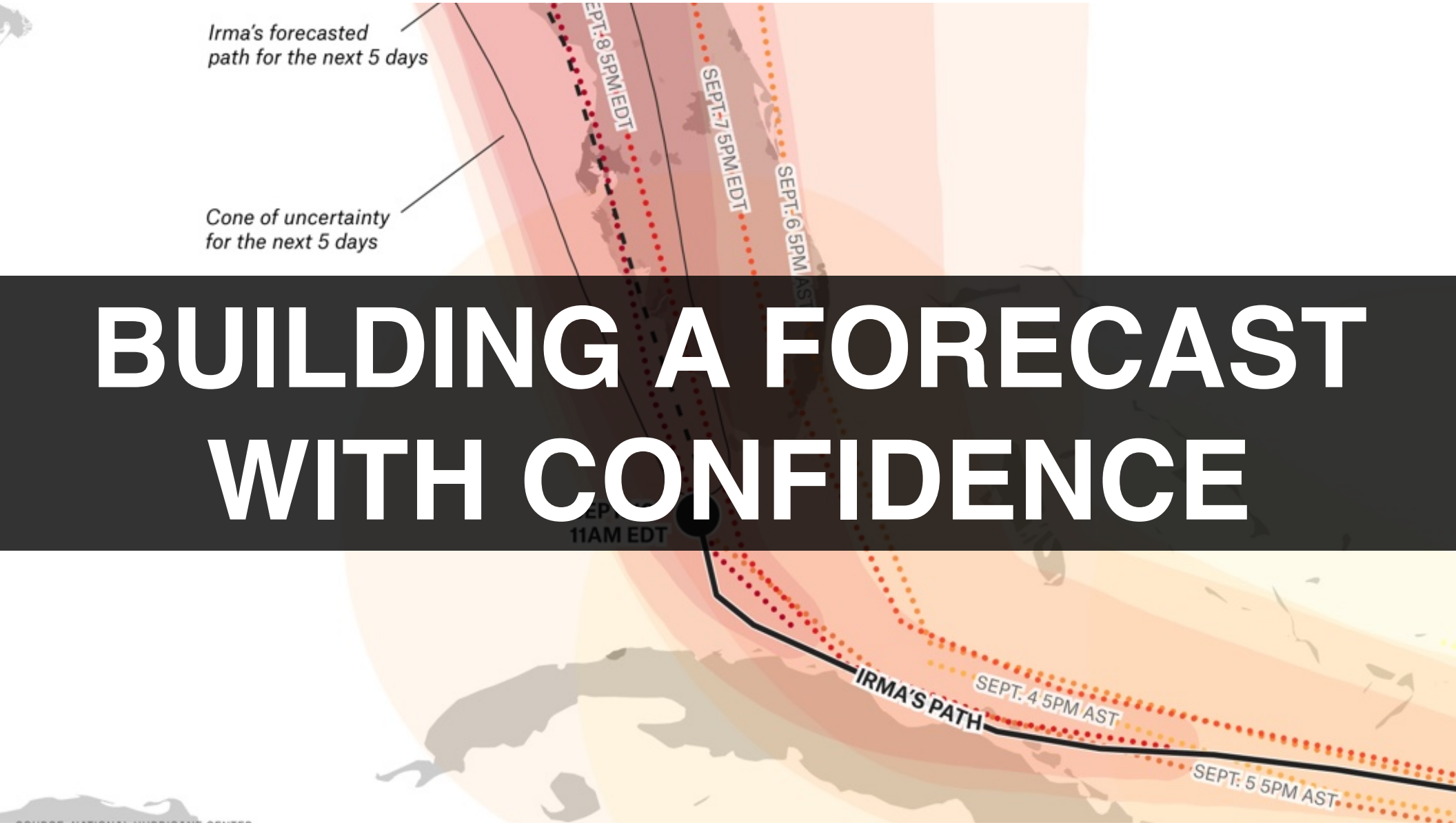
1. Statement about some future outcome or unknown event
2. Statement about the level of uncertainty, but not guaranteed
3. Way of eventually testing the actual outcome against the forecast

1. Question: “Will it snow tomorrow?”
2. Uncertainty: “10% chance of snow.”
3. Feedback: “Did it snow?” No? Why did we get it wrong? Are we wrong the appropriate amount of expected times?


QUIZ TIME!
@chrisshinkle

# A minimum set of assumptions should cover these aspects:

1. A measure of the ability to start delivery
2. A measure of initial scope and size.
3. A measure of expected rework and scope growth.
4. A measure of expected progress of scope delivery over time.
5. A measure of acceptable quality to be able to deliver to customers.
6. A measure of the ability to deliver to customers (environments, process, logistics).




# BUILDING A FORECAST WITH CONFIDENCE

## KEY INPUTS

# HOW BIG (size)?

Understanding the size of a feature or project with less effort

# HOW MUCH MORE (growth)?

Understanding how much the original scope might grow

# HOW FAST (pace)?

Understanding at what rate we will complete work

## FORECASTING TOTAL STORY COUNT

**QUESTION**: How can I estimate the size of a feature or project without analyzing every piece of work?

**THEORY**: The "size" patterns of randomly sample features, will persist through all other features. Analyze a few and compute for the many.

## FORECASTING TOOLS: HOW BIG?

# STORY COUNT FORECASTER

bit.ly/storyforecaster

@chrisshinkle

## FORECASTING GROWTH

**QUESTION**: Any time we use a distance and pace based model for forecasting, we must consider what might increase the distance or modify the pace? Not doing so means poor forecasts.

**THEORY**: There are 4 areas of possible growth to consider. Make sure you address each one in your forecast.

## TYPES OF GROWTH TO CONSIDER

**Time based** - The longer we go the more alterations to original scope get added.

**Rate based** - The more work we complete the more we learn about what we need to do to deliver. (e.g. defects and rework)

**Scale based** - Often work items are split as the team understands the feature story in more detail.

**Event based** - Feedback or things that go wrong in the approval to release process.



- Troy Magennis, Forecasting using data, Chapter 8

# TIME BASED GROWTH

| Release frequency to customers | Technically easy | Technically hard |
|---|---|---|
| Continuous to every 2 weeks | 1x | 1.25x |
| 3 weeks to every 6 weeks | 1.25x (+25%) | 1.5x |
| 7 weeks to every 12 weeks | 1.5x (+50%) | 1.75x |
| 13 weeks to every 26 weeks | 1.75x (+75%) | 2x |
| 26 + weeks | 2x (+100%) | 4x |



- Troy Magennis, Forecasting using data, Chapter 8

# RATE BASED GROWTH

| Growth vector | Occurrence Estimate | Story Count Estimate |
|---|---|---|
| **Defects** | Every story | 1 to 3 |
| **Localization** | U.I. stories, 20-30% total | 3 to 4 (string translation, adding to deployed resources, testing) |
| … | … | … |
| … | … | … |



- Troy Magennis, Forecasting using data, Chapter 8

## SCALE BASED GROWTH (unit correction)

- This is the most overlooked growth of work.
- It's caused when the pace of delivered items is assumed to be the pace remaining backlog items will be completed.
- A good starting scale correction estimate is 1 to 3 times.



- Troy Magennis, Forecasting using data, Chapter 8

# EVENT BASED GROWTH

| Risk | Probability Estimate | Story Count Estimate (if this risk happens) |
|---|---|---|
| **Performance under load < 1 second page load time** | 50-75%<br>More likely than not | 20 to 30 stories to add an indexing server |
| **Browser compatibility. Major errors in Safari, Chrome or IE 10+ browsers** | 20-40%<br>Less likely due to early QA using virtual machine images. Beta testers identified. | 6-15 stories<br>Assumes, 2 to 5 stories per browser. |
| … | … | … |
| … | … | … |



- Troy Magennis, Forecasting using data, Chapter 8

## FORECASTING DURATION (and delivery date)

**QUESTION**: How can I estimate the amount of time it will take to deliver a feature or project?

**THEORY**: Using a range estimate or actual team delivery rate, calculate how many time periods to complete delivery

## FORECASTING TOOLS: HOW LONG?

# THROUGHPUT FORECASTER

bit.ly/throughputforecaster

@chrisshinkle

## TOP 3 REASONS YOUR FORECASTS MAY FAIL

# START DATE INCORRECT

Teams almost never fully available on day 1.

# BACKLOG VS COMPLETION RATE

We may under-forecast if split-rate isn't properly accounted.

# IGNORING RISKS

Work that "might" need to be done but we don't know yet

@chrisshinkle

# TAKEAWAYS

- Use product outcomes for selecting what goes into the release plan
- Use forecasting to “guess” the backlog size and project duration
- Any statement about duration should include a range and probability

# download slides and worksheets at:

# info.sep.com/cincydayofagile2018