Getting The *Right* Product

Not iterating your way into oblivion

Mike Bria
@mbria / bria526@gmail.com
InfoQ
Industrial Logic, Inc.

2 Common Pitfalls

- *"The Scatterbrain"*: The project has little direction, is plagued with chaos, churning, and misdirection. Who knows what product you'll end up with...Frankenstein!
- *"The Robot"*: The backlog is little more than a static checklist to be executed upon. The product you end up with is exactly what you originally planned, but unfortunately nothing like what you ***need***.

The Trifecta Fix

- *Project Chartering*: Clarify Your Destination
- *Story Mapping*: Organize Your Journey
- *Planning To Iterate*: Embrace Feedback

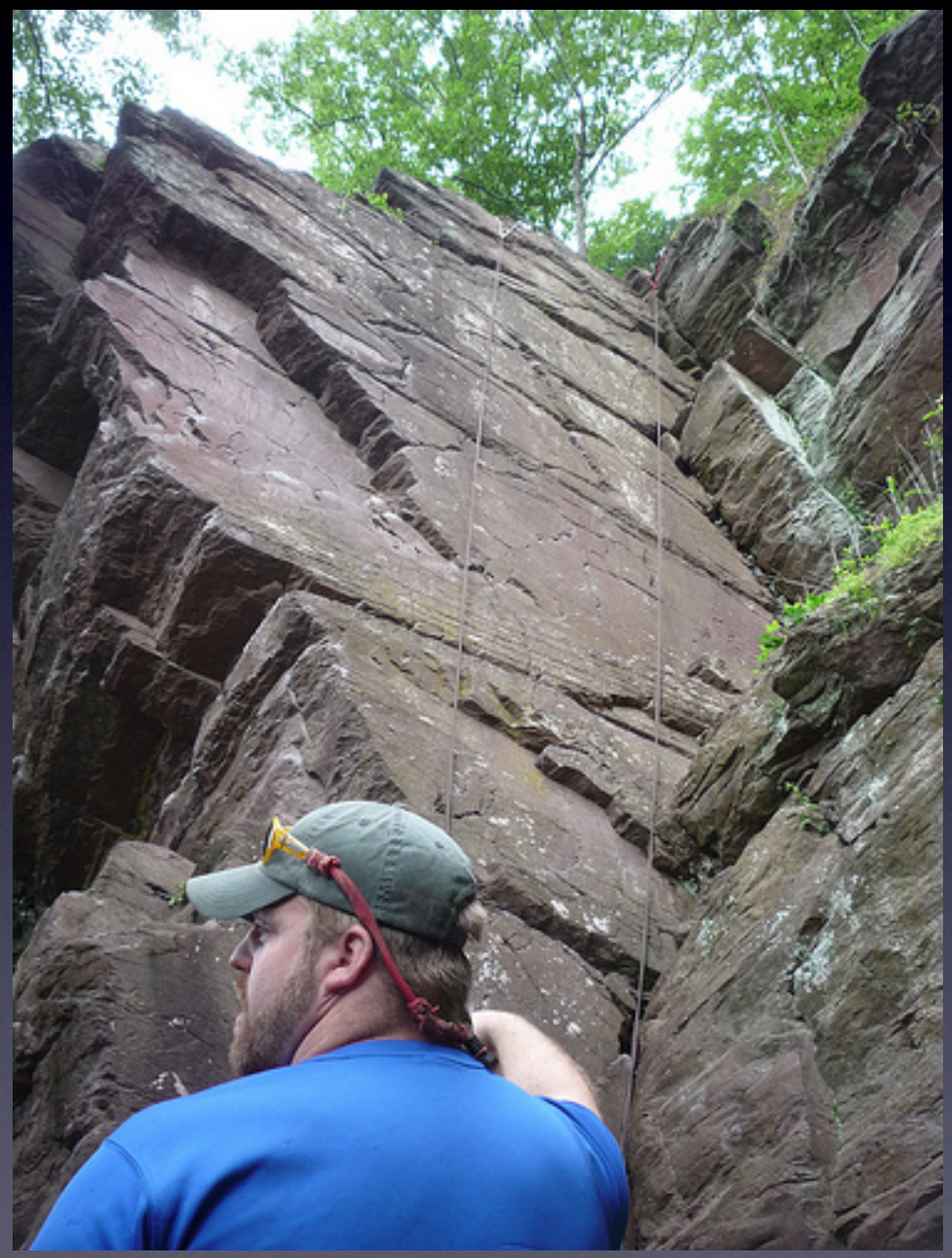
ONE:
Project Chartering

Joshua Kerievsky:

“Discover and maintain a Project Community's clarity of purpose”

The Vision

- What it is the ultimate purpose of the project?
- In business terms, “Strategy”
- Inspirational



The Vision

- What it is the ultimate purpose of the project?
- In business terms, “Strategy”
- Inspirational

The Vision

- What it is the ultimate purpose of the project?
- In business terms, “Strategy”
- Inspirational

The Vision

- What it is the ultimate purpose of the project?
- In business terms, “Strategy”
- Inspirational
- The “Guiding Light”

Your Mission

(, should you choose to accept it...)

- How will we go about achieving the vision?
- In business terms, “Tactical”
- High level operational objectives

Your Mission

(, should you choose to accept it...)

- Get a solid roping system in place

Your Mission

(, should you choose to accept it...)

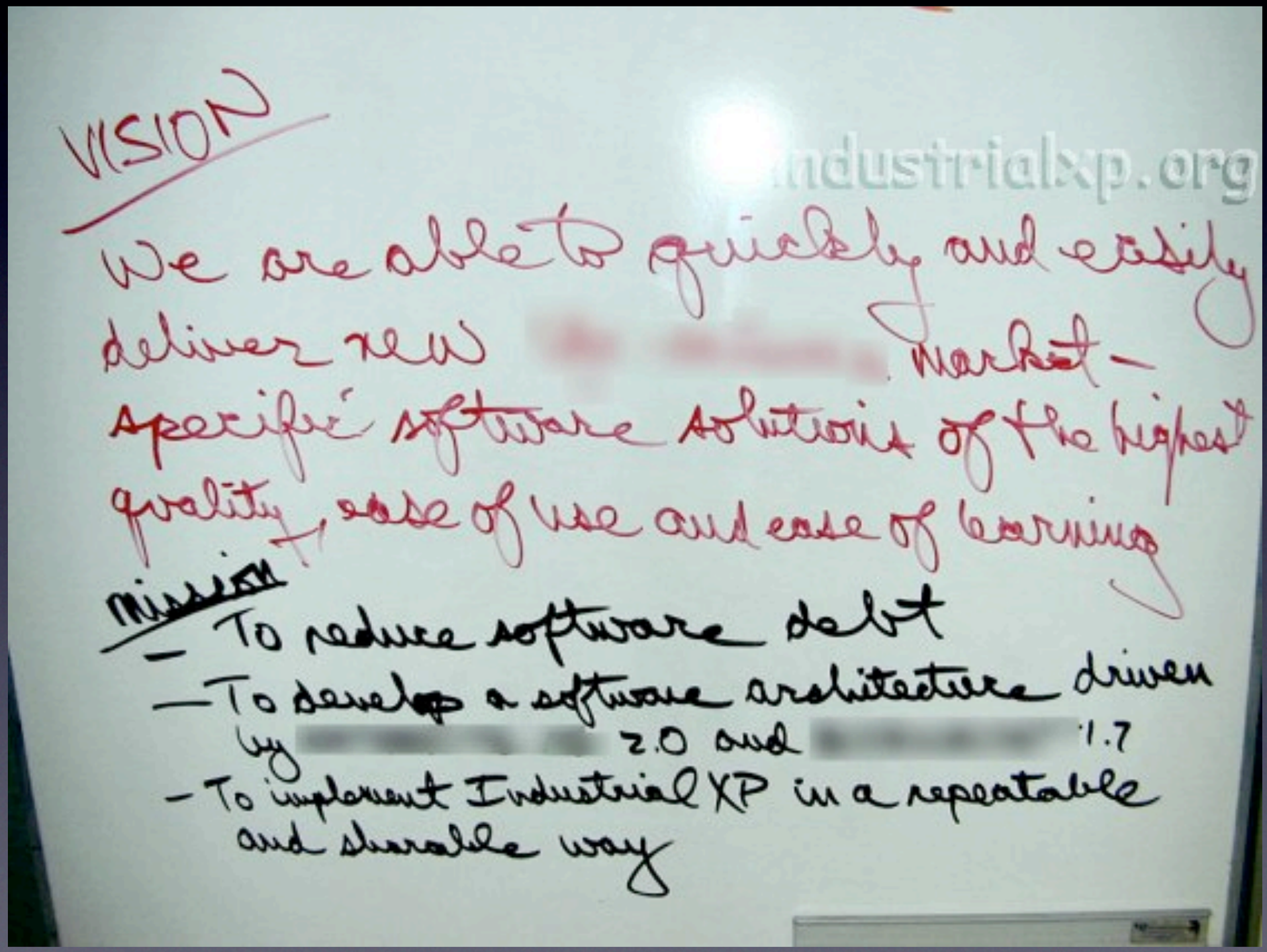
- Get a solid roping system in place
- Learn the art of knotting

Your Mission

(, should you choose to accept it...)

- Get a solid roping system in place
- Learn the art of knotting
- ***Climb rocks!***

Simple...And Visible!



Management Tests

- Test-Driven Management (ala Test-Driven Development)
- Project success criteria expressed as TESTS
- SMART: Specific, Measurable, Achievable, Relevant and Time-Based
- They set a destination, but don't specify how to get there

Management Tests

- Test-Driven Management (ala Test-Driven Development)
- Project success criteria expressed as TESTS
- SMART: Specific, Measurable, Achievable, Relevant and Time-Based
- They set a destination, but don't specify how to get there

Apple's iTunes:

Management Tests

- Test-Driven Management (ala Test-Driven Development)
- Project success criteria expressed as TESTS
- SMART: Specific, Measurable, Achievable, Relevant and Time-Based
- They set a destination, but don't specify how to get there

Apple's iTunes: ***Our new service will register at least 1 million song downloads during its first month in production.***

Management Tests

- By 9:00am, a double rope, pulley-based climbing system will be setup
- By 9:30am, we each will have successfully tied 3 triple-lundy knots
- By 11am, we will have reached the summit 3 times each
- By 11:30am, we will have a photo of one of us at the summit

Project GREEN-BAR!

TWO:
Story Mapping

The [User's] World Is ***Not*** *Flat.*

(And neither should our backlogs be!)

The Forest For The Trees

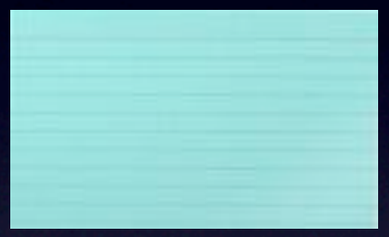
Jeff Patton: *"That's what a flat backlog is to me. A bag of* ***context-free mulch.***"

User “Stories” ?

“User Stories” are not all created equal!

User “Stories” ?

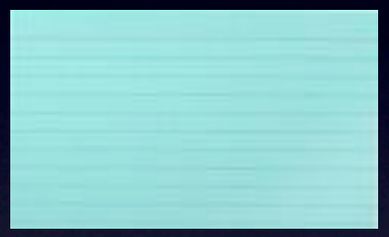
“User Stories” are not all created equal!



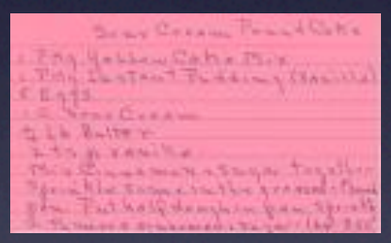
- User ***Activities***
- “I wanna manage my email so I can ...“

User “Stories” ?

“User Stories” are not all created equal!



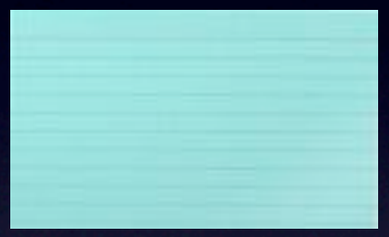
- User ***Activities***
- “I wanna manage my email so I can ...“



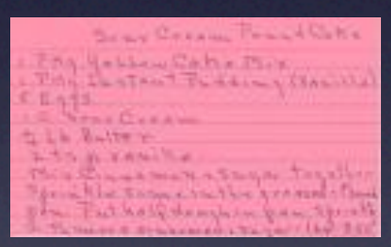
- User ***Tasks***
- "send message," "read message," "delete message," "mark message as spam"

User “Stories” ?

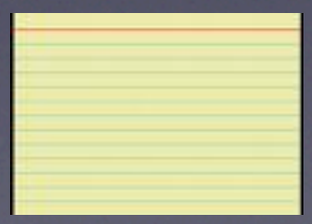
“User Stories” are not all created equal!



- User ***Activities***
- “I wanna manage my email so I can ...“



- User ***Tasks***
- "send message," "read message," "delete message," "mark message as spam"



- Sub-Tasks (Details)-> Software **features**
- “change font..”,“sort by sender”,...

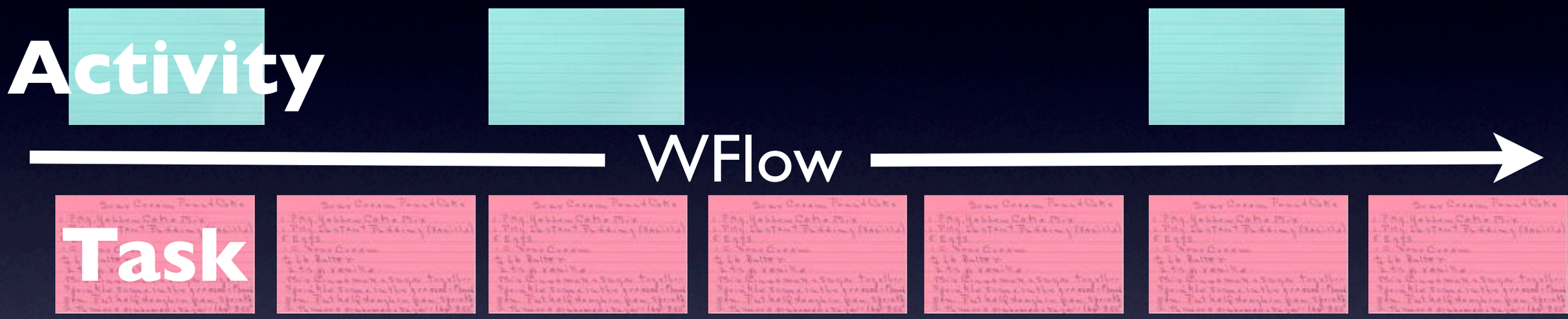
Order from Chaos:
A *Map*

Order from Chaos:
A *Map*

WFlow

Order from Chaos:
A *Map*

Order from Chaos:
A *Map*



Order from Chaos:
A *Map*

Activity

WFlow

Task

Sub-Tasks
("features")

Order from Chaos:
A *Map*

Activity

WFlow

Task

Sub-Tasks
("features")

Order from Chaos:
A *Map*

Activity

WFlow

Task

Sub-Tasks
("features")

Order from Chaos:
A *Map*

Activity

WFlow

Task

Sub-Tasks
("features")

Order from Chaos:
A *Map*

Activity “Backbone” (Rock Solid)

WFlow

Task

Sub-Tasks (“features”)

Order from Chaos:
A *Map*

Activity “Backbone” (Rock Solid)

WFlow

“ Walking Skeleton ” (Stiff, but flexible)

Sub-Tasks (“features”)

Order from Chaos:
A *Map*

Activity “Backbone” (Rock Solid)

WFlow

“ Walking Skeleton ” (Stiff, but flexible)

Sub-Tasks (“features”)

(Very transient)

Order from Chaos:
A *Map*

Activity “Backbone” (Rock Solid)

WFlow

“ Walking Skeleton ” (Stiff, but flexible)

Sub-Tasks (“features”)

(Very transient)

Necessity

Delivering The Whole Story

Activity

WFlow

Task

Sub-Tasks (“features”)

Necessity

Delivering The Whole Story

Activity

WFlow

Task

Release #1

Sub-Tasks ("features")

Necessity

Recall, your **Charter**

- You may notice that your Vision and Mission help limit/focus what User's you care about and when
- You may notice that your Management Tests help limit/focus what Activities you care about and when
- You MUST notice that your Charter helps you decide what "yellow stickies" stay, go, and/or change during development

Which brings us to...

THREE:
Planning To Iterate

Iterate: ORIGIN mid 16th cent. : from Latin ***iterat- 'repeated,'*** from the verb ***iterare***, from ***iterum 'again.'***

Increment: ORIGIN late Middle English : from Latin incrementum, from the stem of increscere 'grow' (see increase).

ITERATE = "to revise", INCREMENT = "add to"

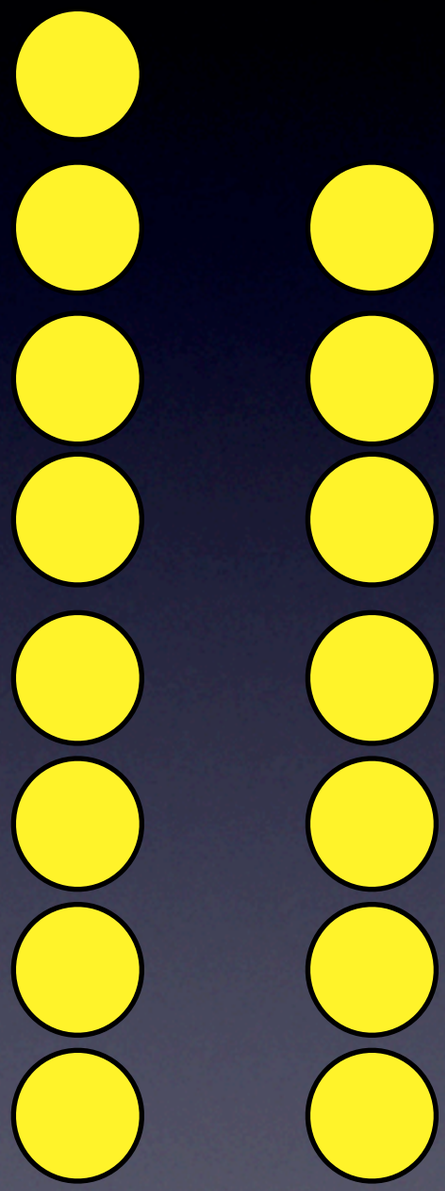
Feedback. Huh?

Maybe this sounds familiar...

- "Cool, we're gonna do XP. This is gonna rock!"

Feedback. Huh?

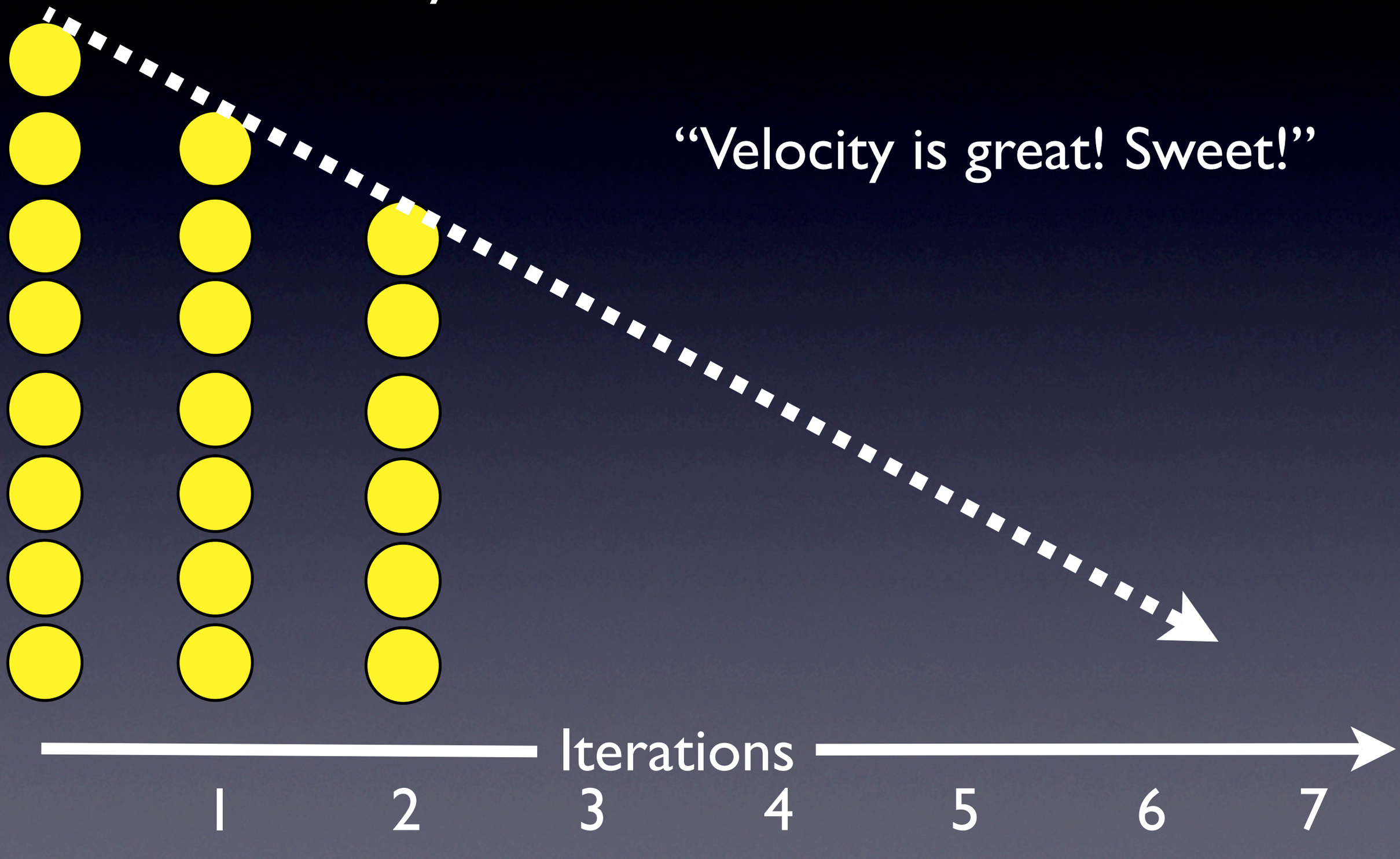
Maybe this sounds familiar...



"Looks Good!"

Feedback. Huh?

Maybe this sounds familiar...

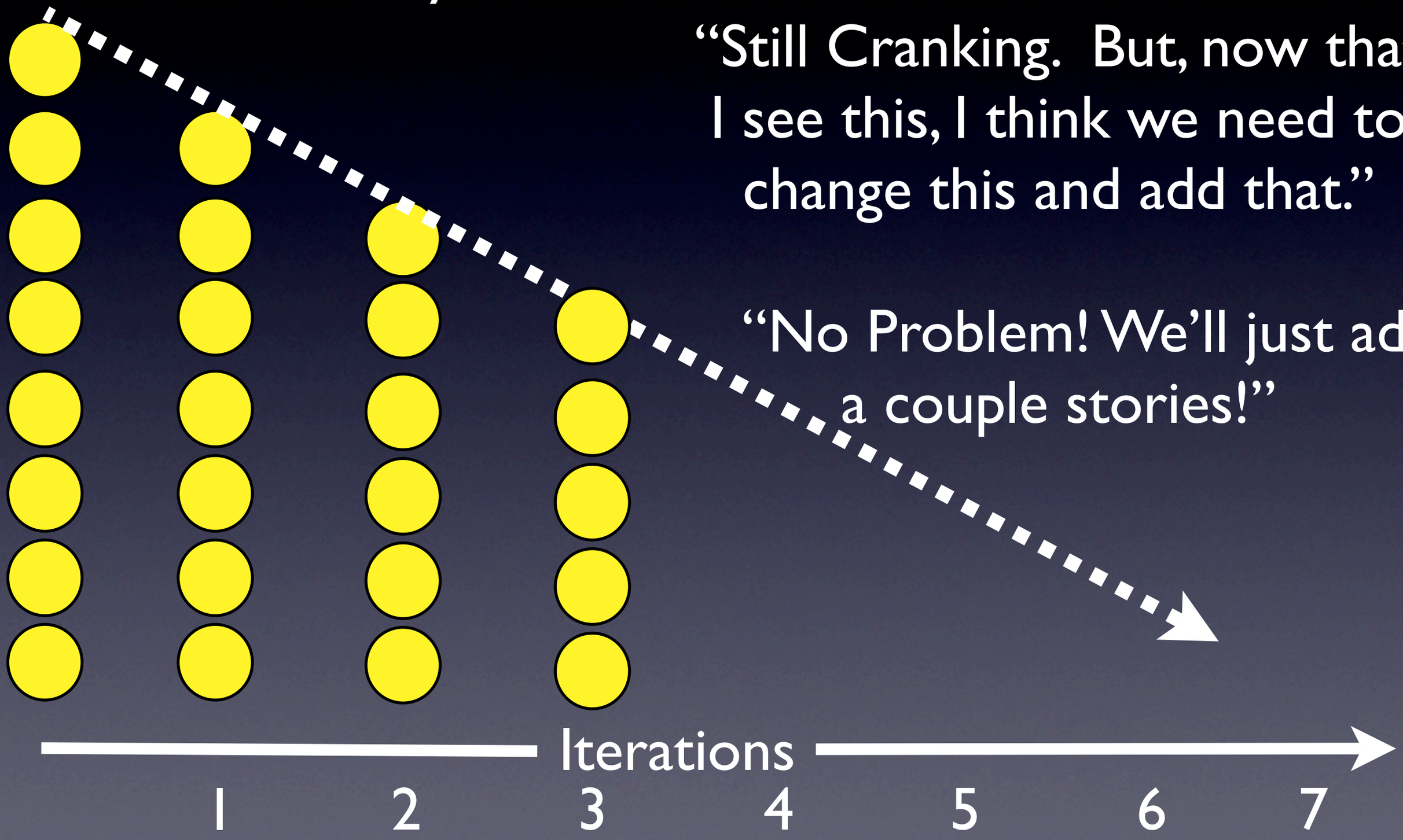


Feedback. Huh?

Maybe this sounds familiar...

"Still Cranking. But, now that I see this, I think we need to change this and add that."

"No Problem! We'll just add a couple stories!"

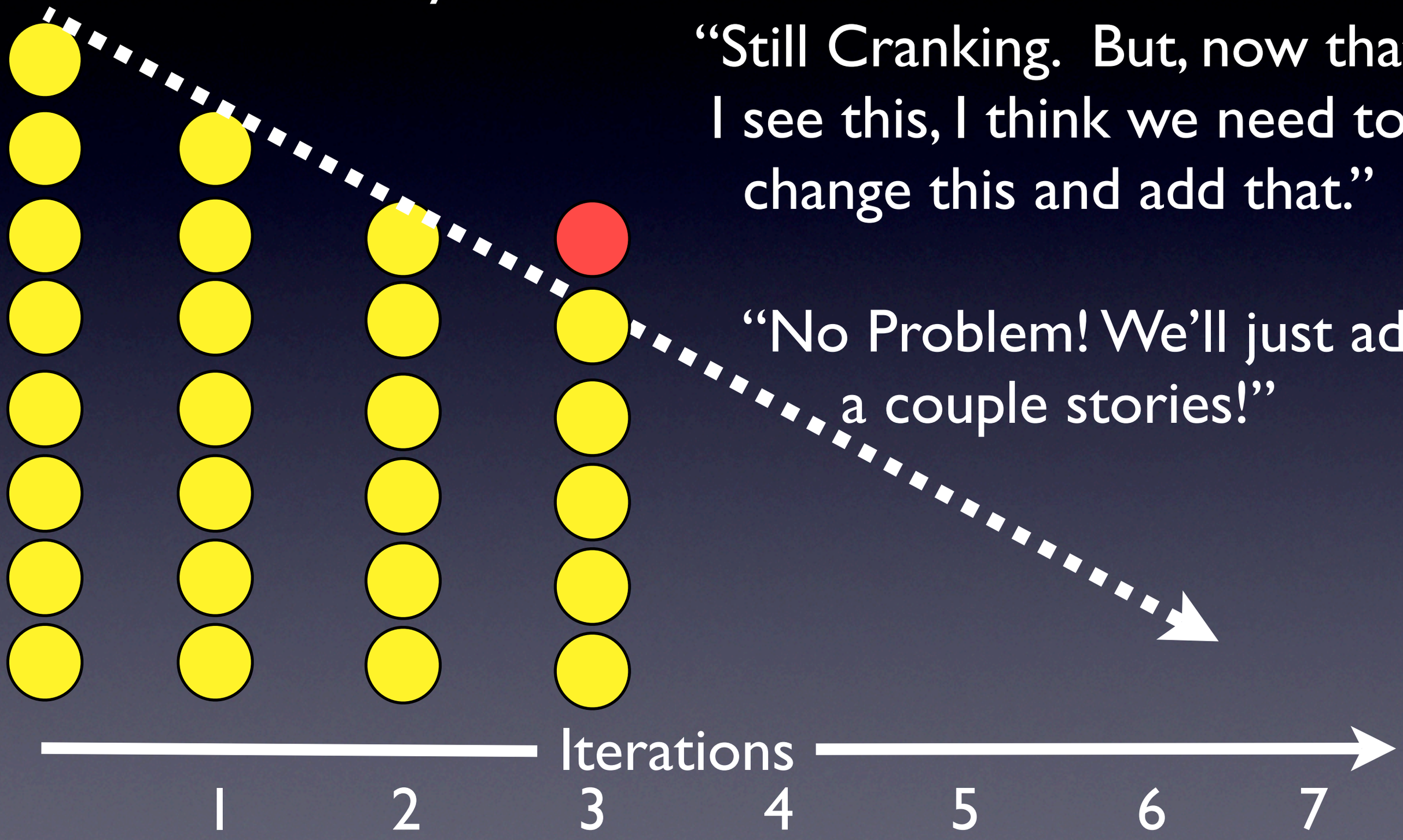


Feedback. Huh?

Maybe this sounds familiar...

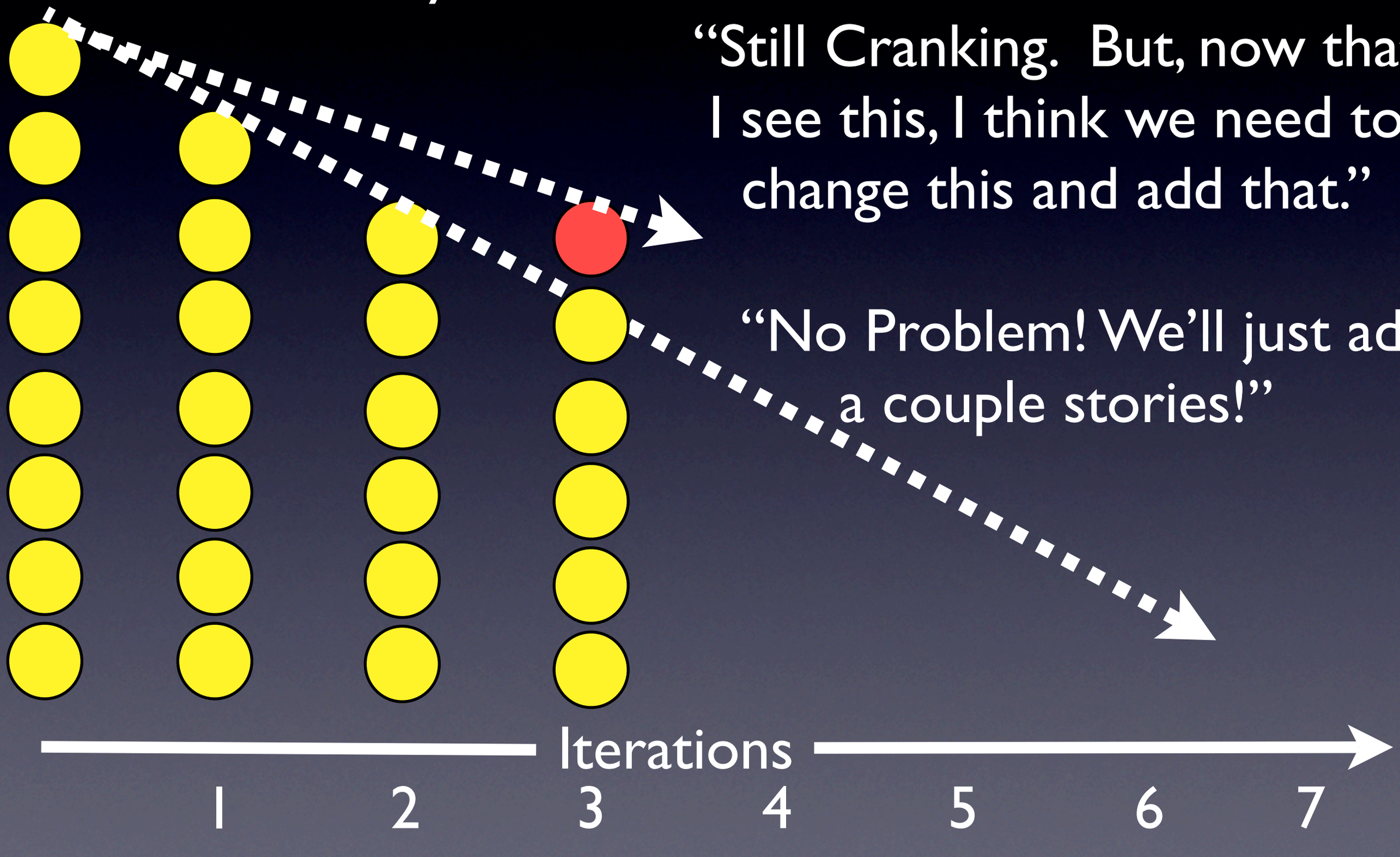
"Still Cranking. But, now that I see this, I think we need to change this and add that."

"No Problem! We'll just add a couple stories!"



Feedback. Huh?

Maybe this sounds familiar...

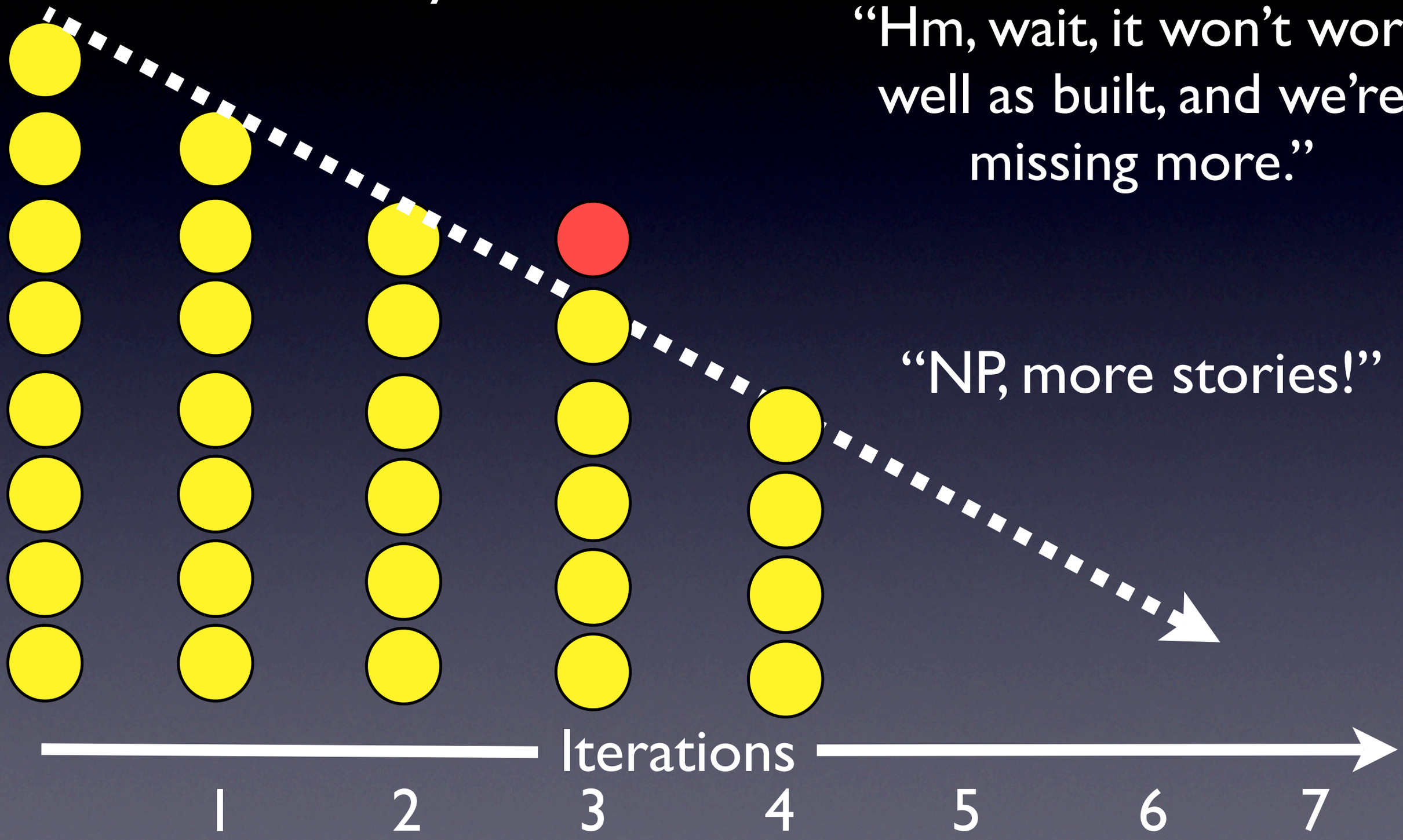


Feedback. Huh?

Maybe this sounds familiar...

"Hm, wait, it won't work well as built, and we're missing more."

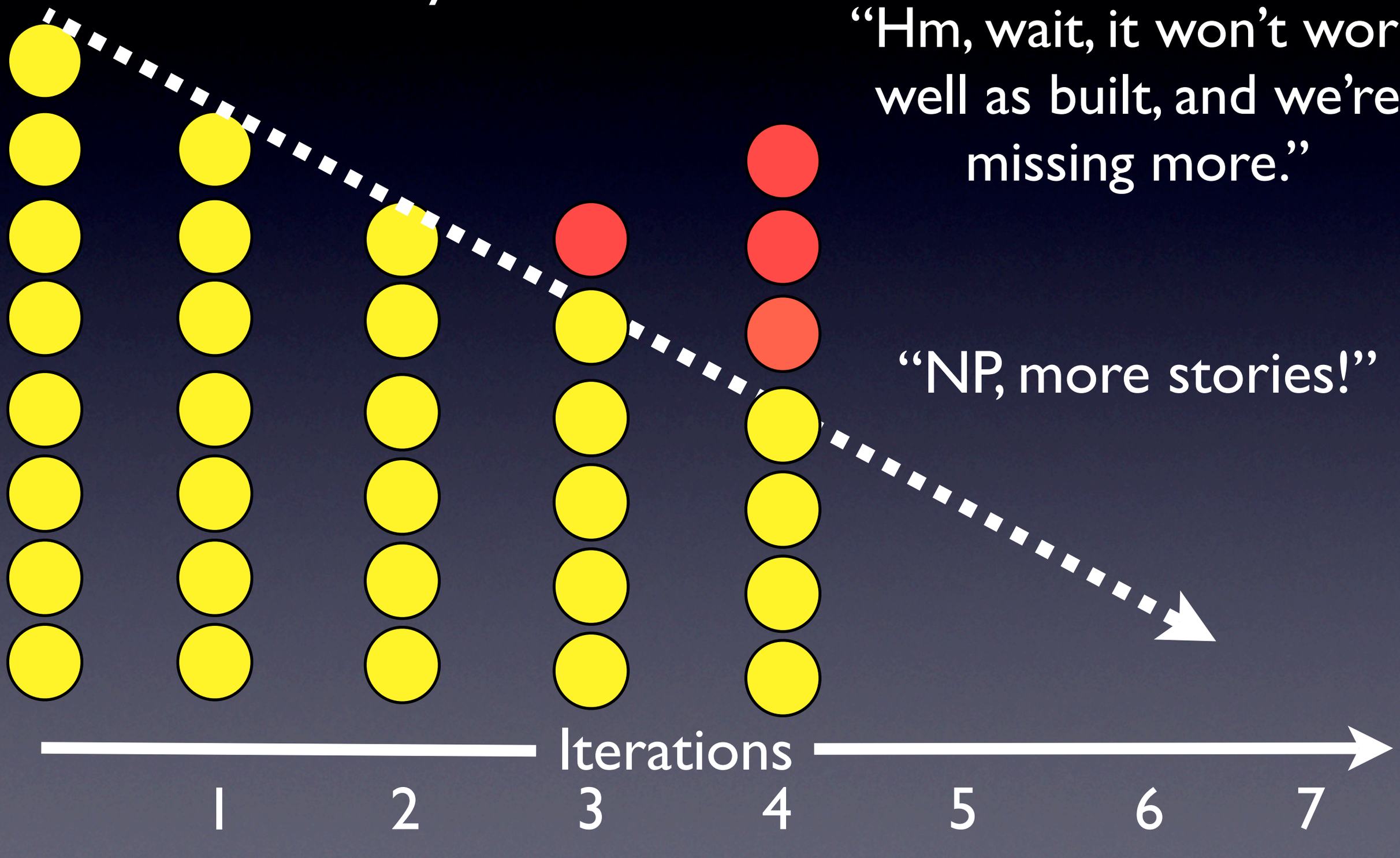
"NP, more stories!"



Feedback. Huh?

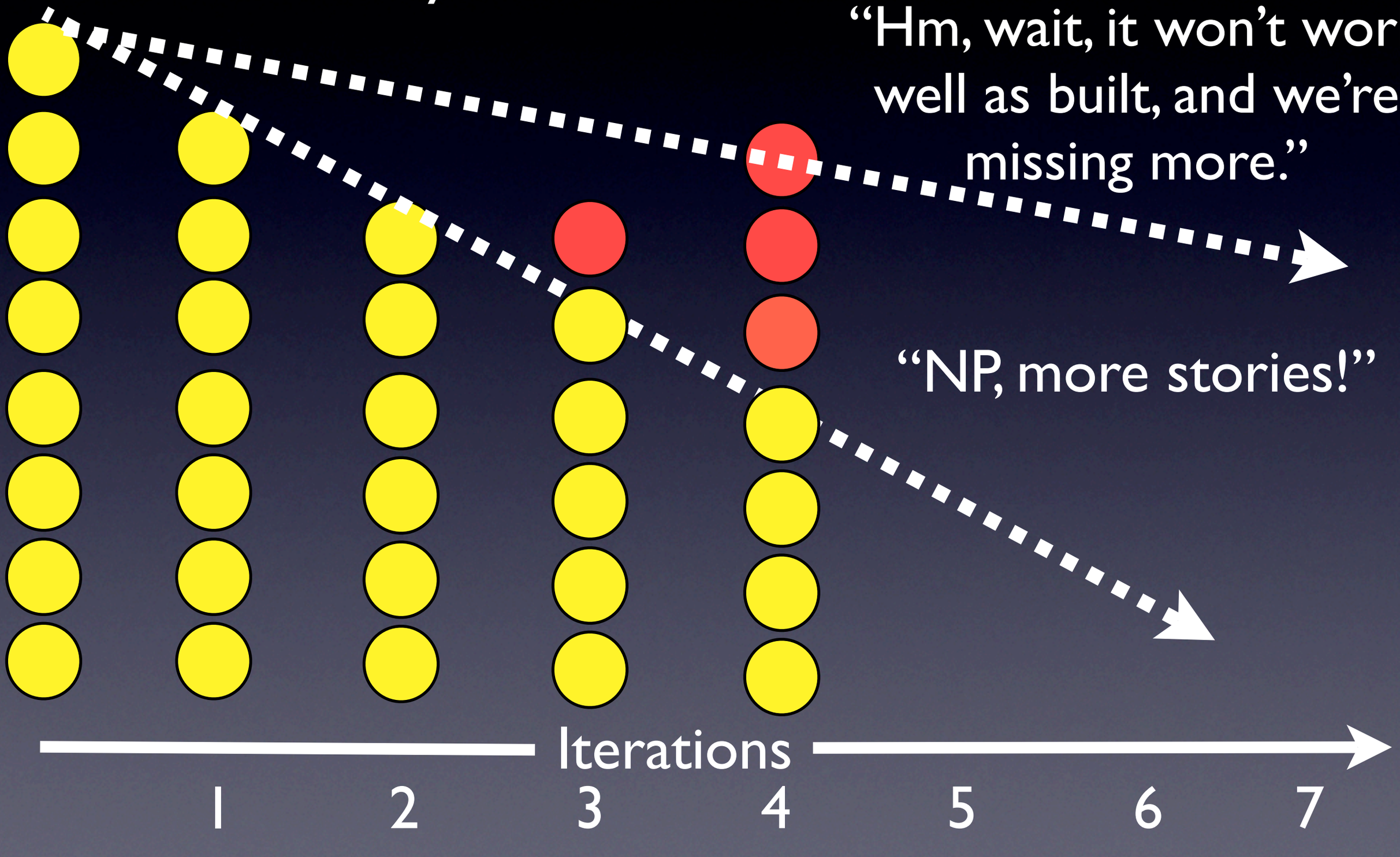
Maybe this sounds familiar...

"Hm, wait, it won't work well as built, and we're missing more."



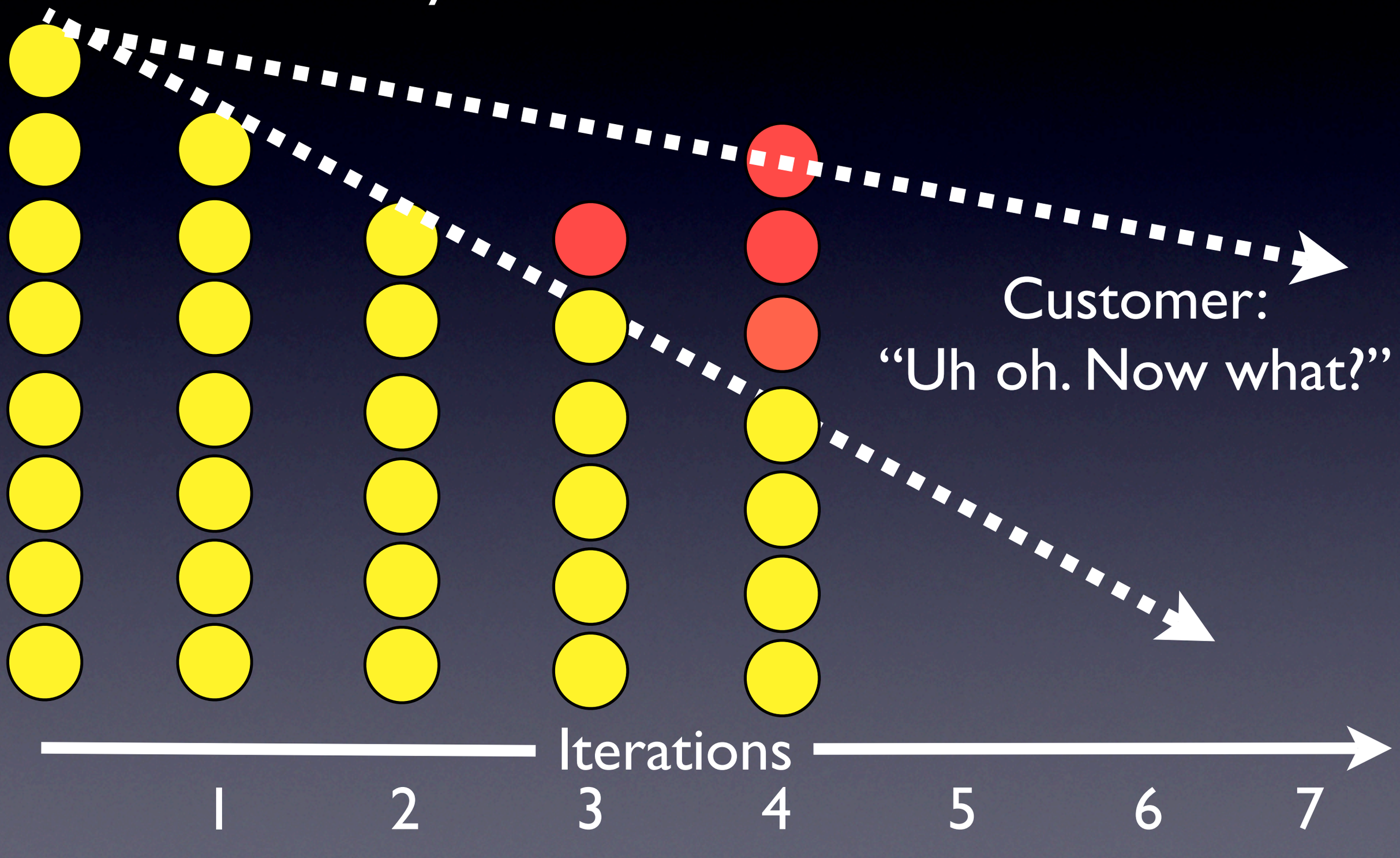
Feedback. Huh?

Maybe this sounds familiar...



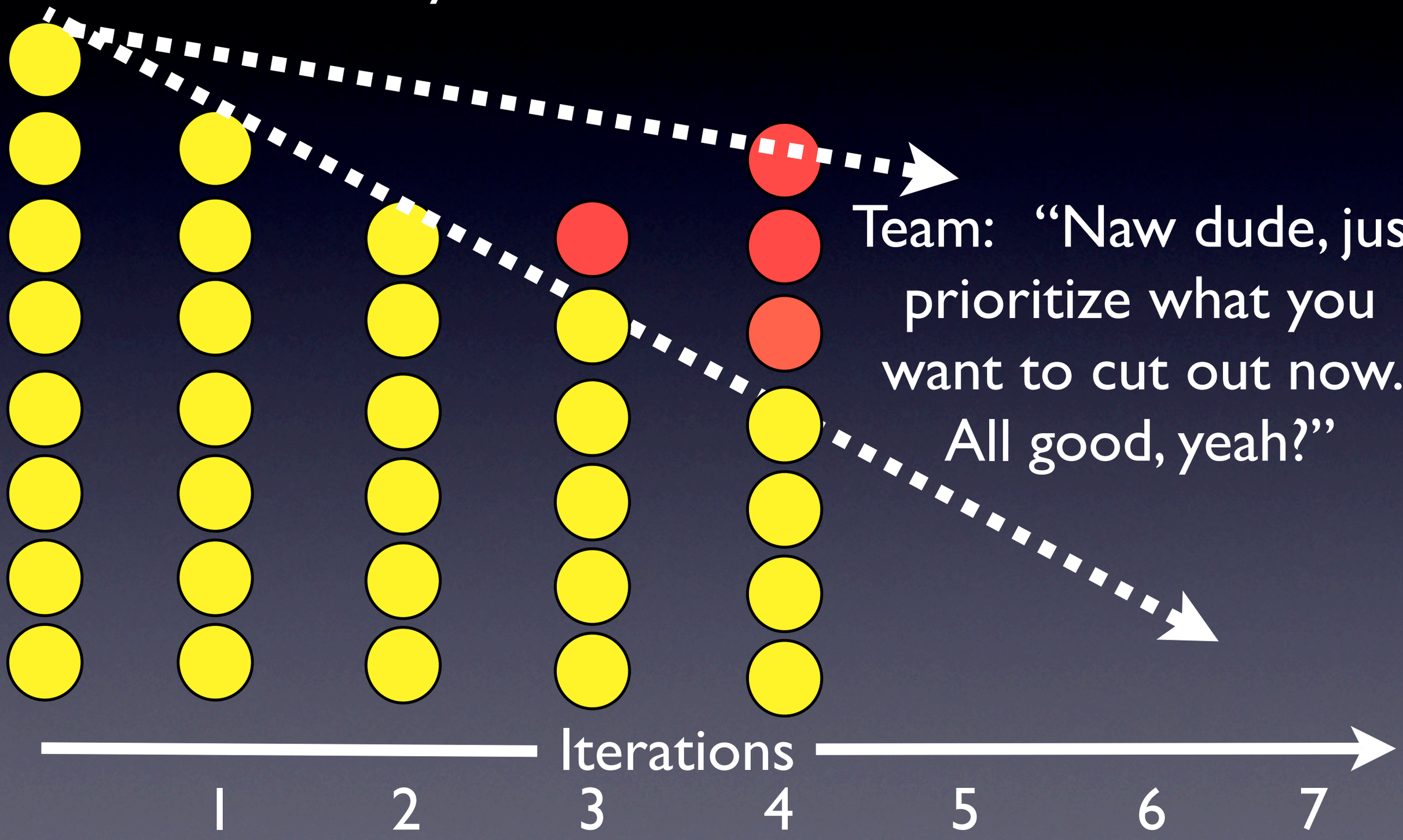
Feedback. Huh?

Maybe this sounds familiar...



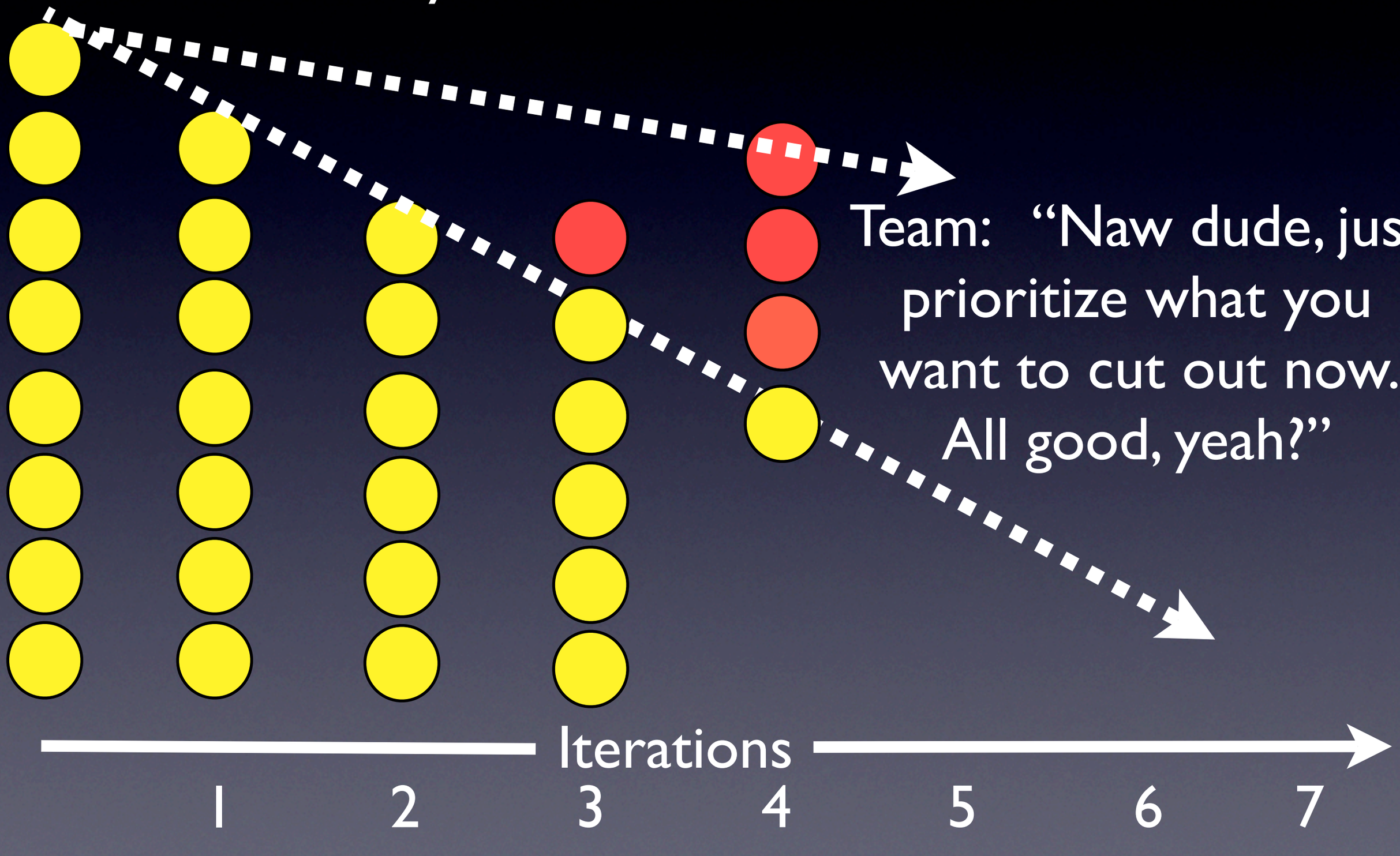
Feedback. Huh?

Maybe this sounds familiar...



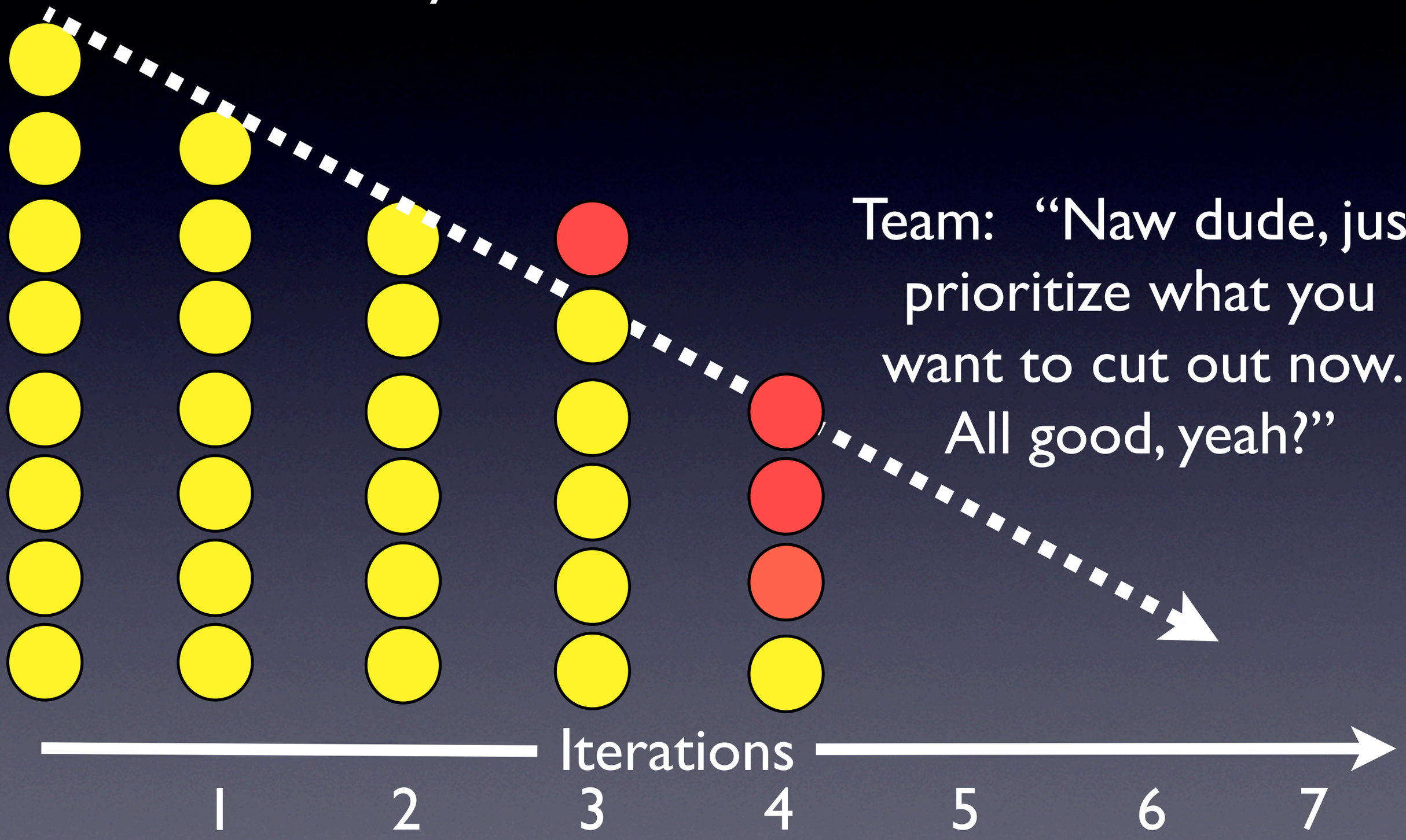
Feedback. Huh?

Maybe this sounds familiar...



Feedback. Huh?

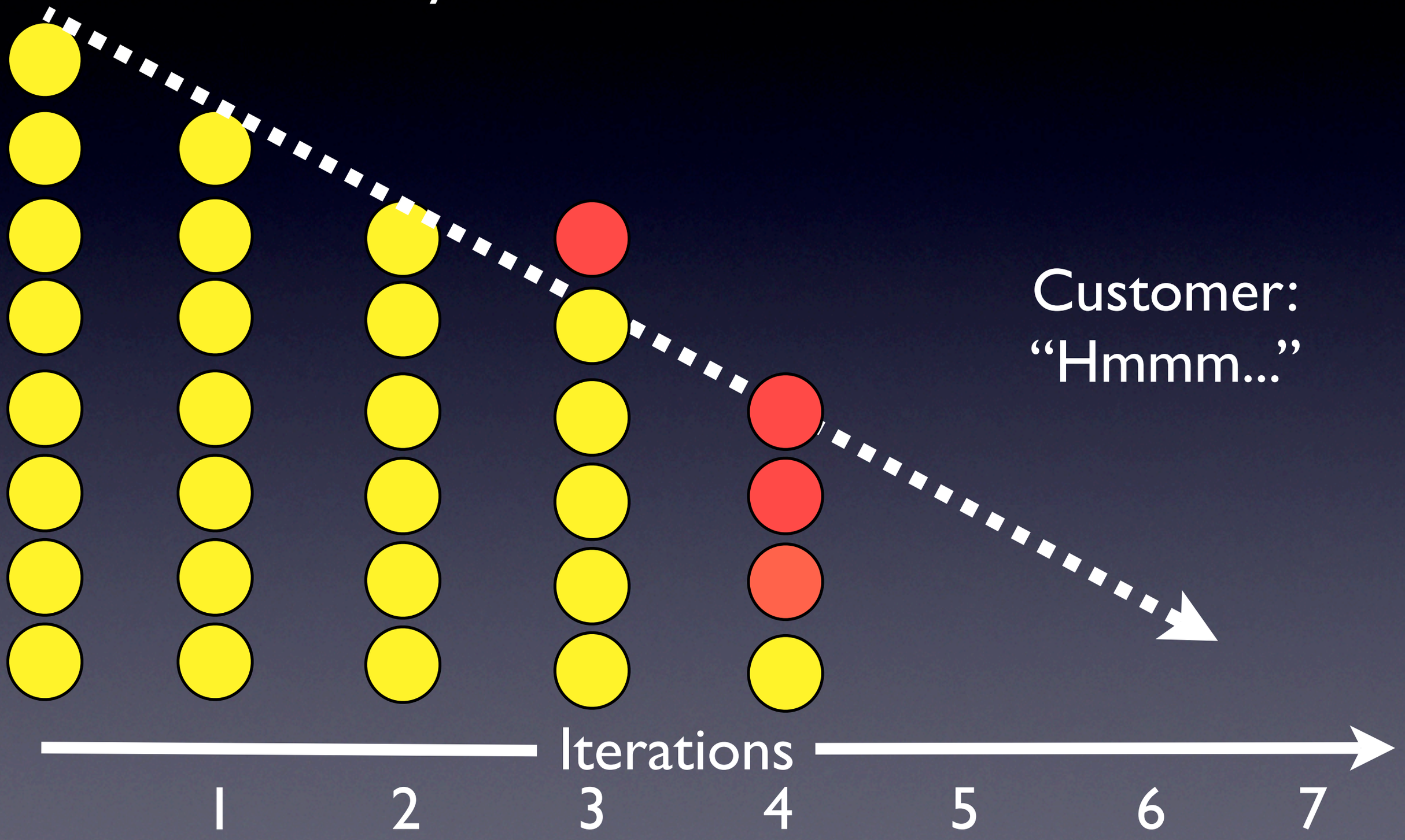
Maybe this sounds familiar...



Team: "Naw dude, just prioritize what you want to cut out now. All good, yeah?"

Feedback. Huh?

Maybe this sounds familiar...

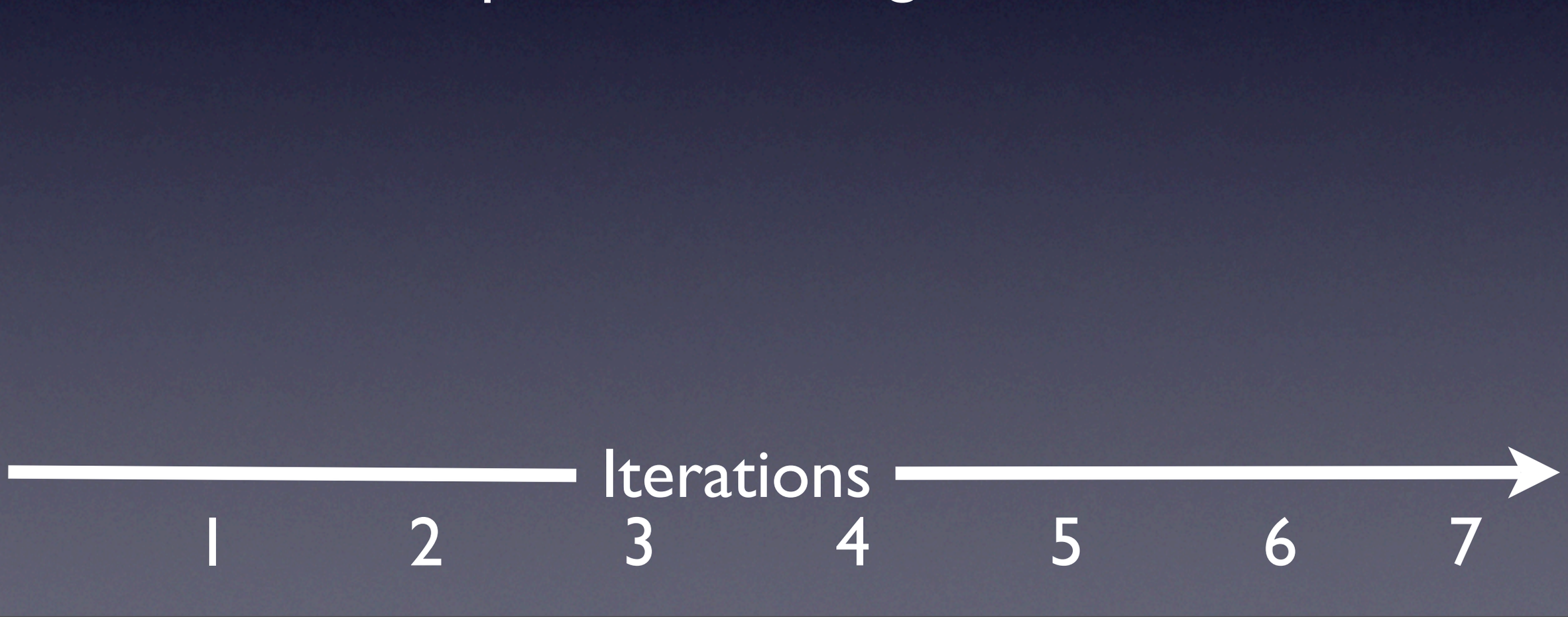


Learning Culture

(That hurt, don't do it again)

Customer:

"That sucked, next time we'll have to spend a little more time up front thinking about what we want.

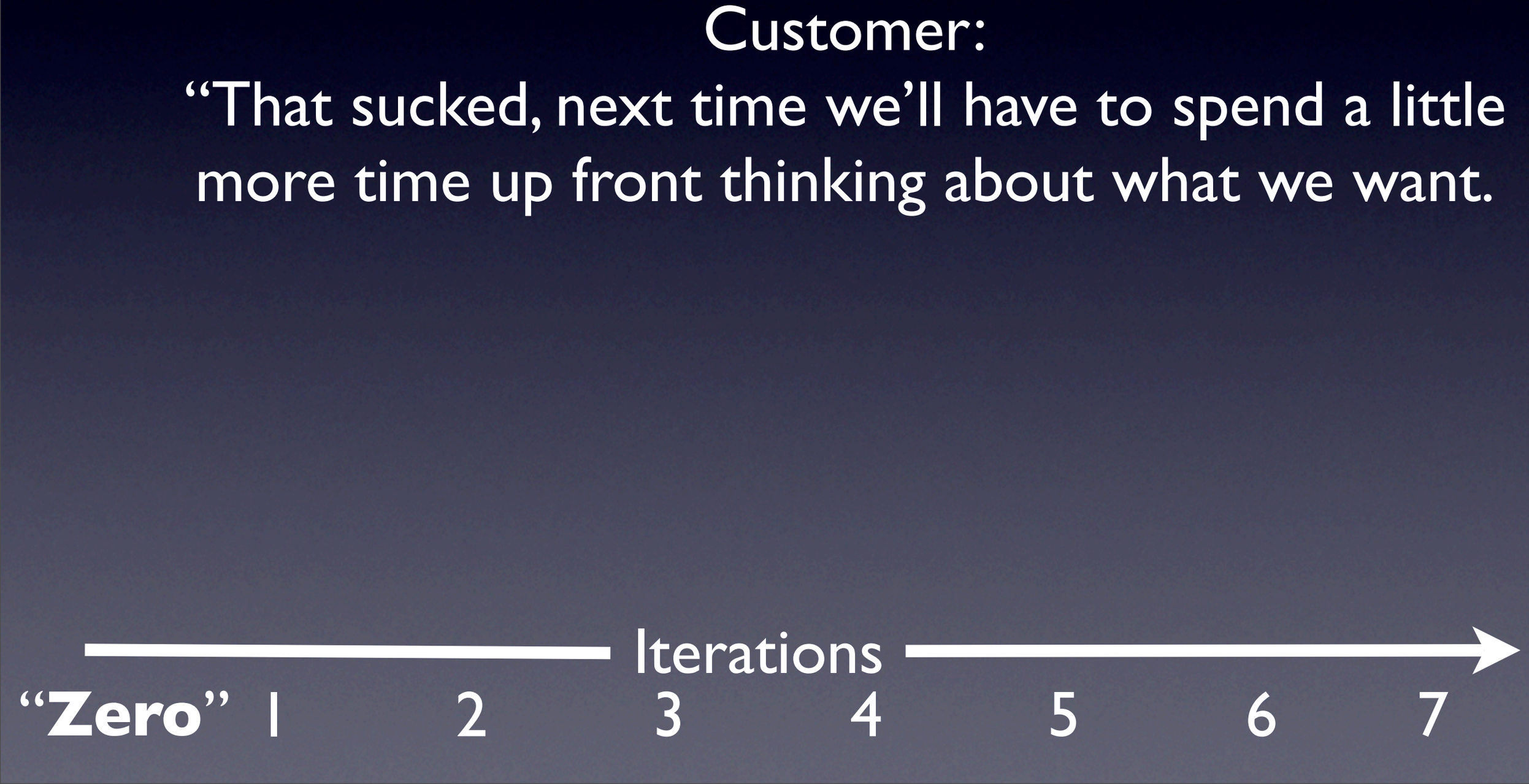


Learning Culture

(That hurt, don't do it again)

Customer:

"That sucked, next time we'll have to spend a little more time up front thinking about what we want.



Learning Culture

(That hurt, don't do it again)

Customer:

"That sucked, next time we'll have to spend a little more time up front thinking about what we want.

Iterations

"**Zero**" 1 2 3 4 5 6 7

Learning Culture

(That hurt, don't do it again)

Customer:

"That sucked, next time we'll have to spend a little more time up front thinking about what we want.

Sorry, A for effort...but not the right idea!
You're on the "Right back to Waterfall" train.

Iterations

"**Zero**" 1 2 3 4 5 6 7

What To Do About It?

Could:

- Buffer The Plan
- Add Up-Front "Revision Cards"
- Rolling Wave Planning

What To Do About It?

Should:

- Story Map!!!
- “Commit” to Management Tests, User Activities (and *maybe* some User Tasks)
- No “commitments” to anything lower

What To Do About It?

Should:

- Story Map!!!
- “Commit” to Management Tests, User Activities (and *maybe* some User Tasks)
- No “commitments” to anything lower

Most importantly, remember “iterating” is about ***REVISION****...and that is what “agile” is all about!!!*

“**Responding To Change**, over following a plan.”

Thanks & Credit to...

- Joshua Kerievsky, *Project Chartering*
- Jeff Patton, *Story Mapping*
- Jeff Patton & Alistair Cockburn, *"Iterating"*

Thanks To YOU!

Getting The *Right* Product

Not iterating your way into oblivion

Mike Bria
@mbria / bria526@gmail.com
InfoQ
Industrial Logic, Inc.