Iterative MMO Development at Scale

With Producers Amy Liu and Sonya Wolfram

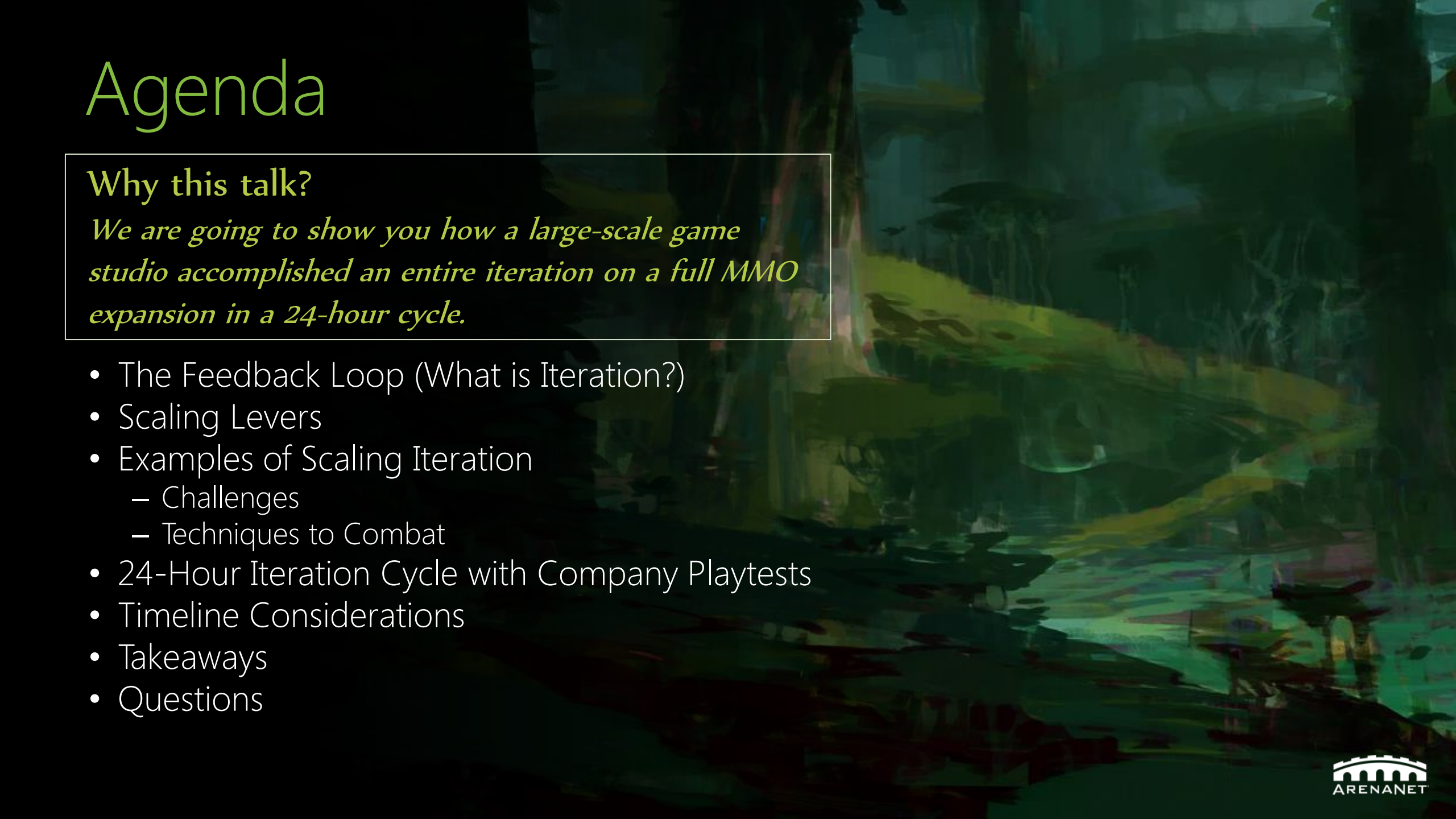
ARENANET

Agenda

Why this talk?

We are going to show you how a large-scale game studio accomplished an entire iteration on a full MMO expansion in a 24-hour cycle.

- The Feedback Loop (What is Iteration?)
- Scaling Levers
- Examples of Scaling Iteration
  - Challenges
  - Techniques to Combat
- 24-Hour Iteration Cycle with Company Playtests
- Timeline Considerations
- Takeaways
- Questions



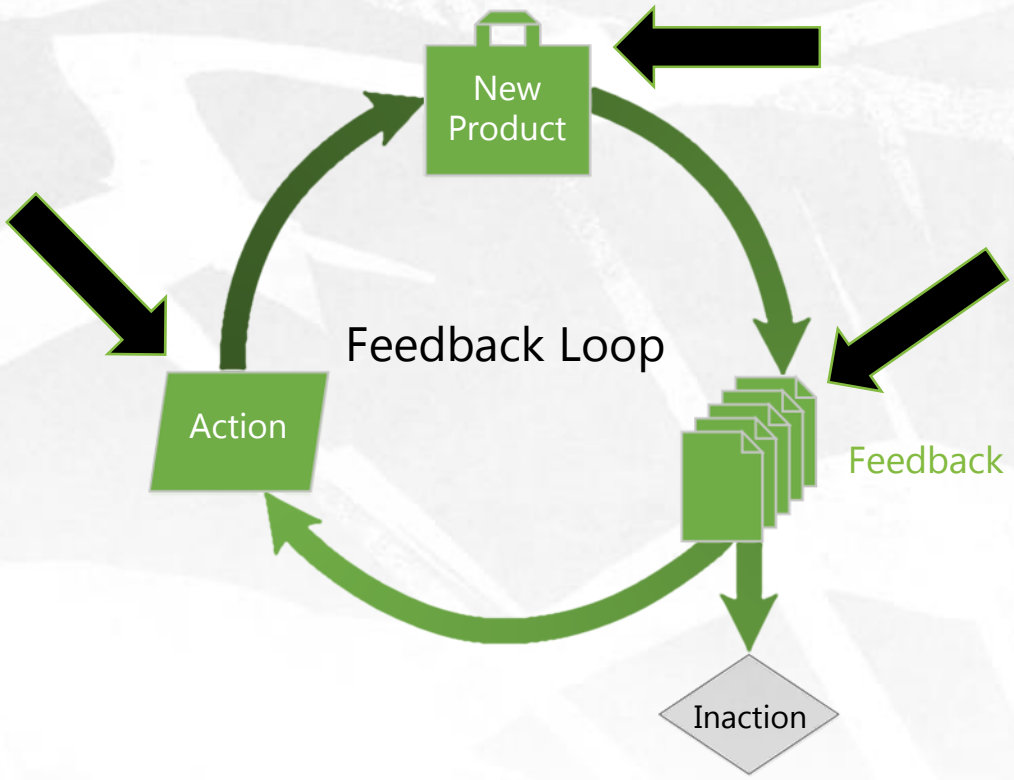
Quick Stats: ArenaNet

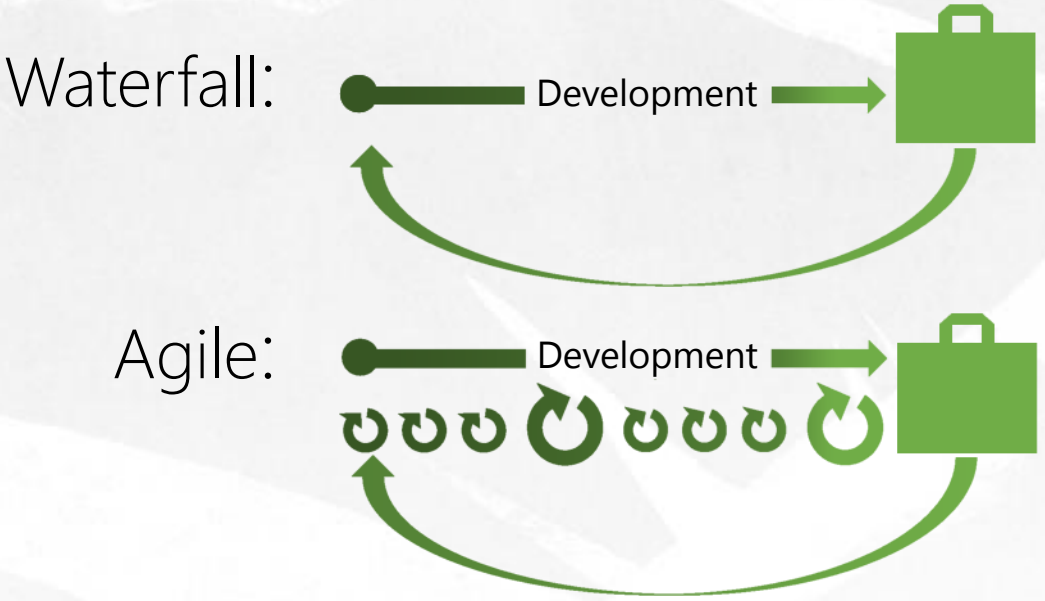
GW2

#2 Ranked MMO by MMORPG.COM, 5 million copies of GW2 sold

ARENANET

What is Iteration?





Iteration

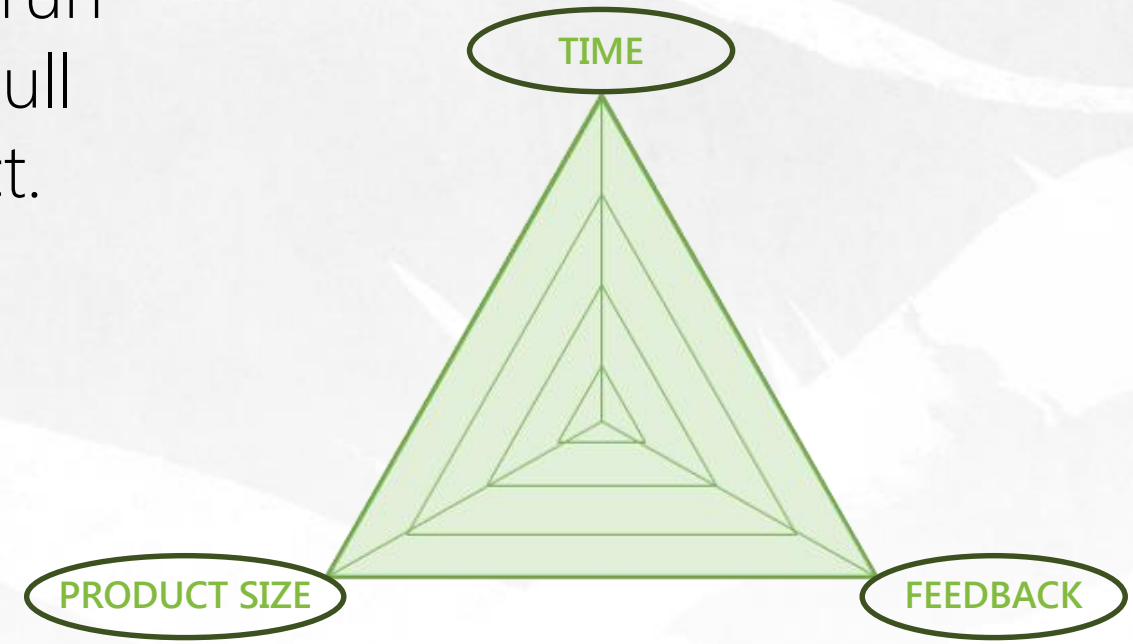
Any change of content due to feedback, bugs, or its relationship with other content shipping with or after it.

ARENANET

Levers

When Scaling this feedback loop, you run into 3 levers that you can potentially pull to maximize the value for your product.

- Product Size, Content Size, or Scope
- Feedback Sources
- Iteration Time or Cycle Length



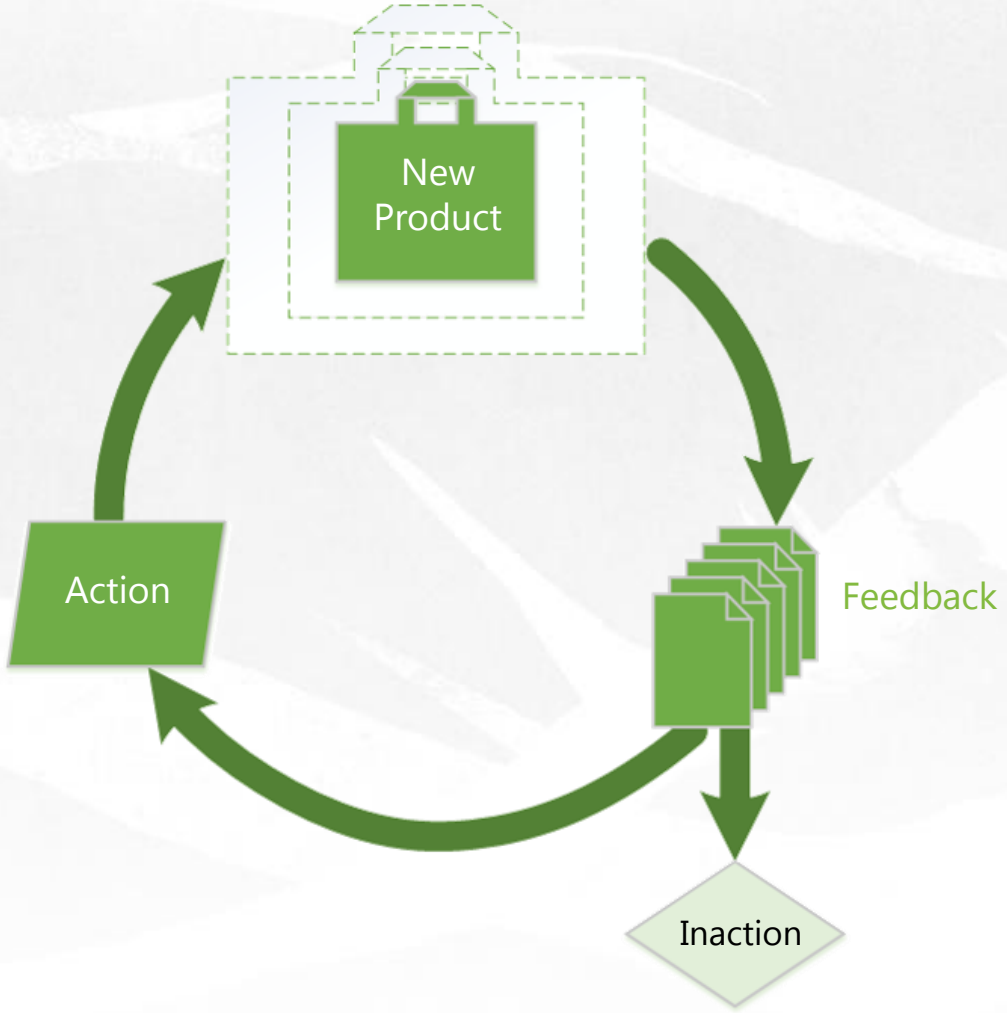
ARENANET

Lever: Product Size

Product Size: The amount of content you are putting into your next version of the product

Examples of Scaling:

- One code change
- Small feature
- Gameplay event
- Expansion
- Franchise



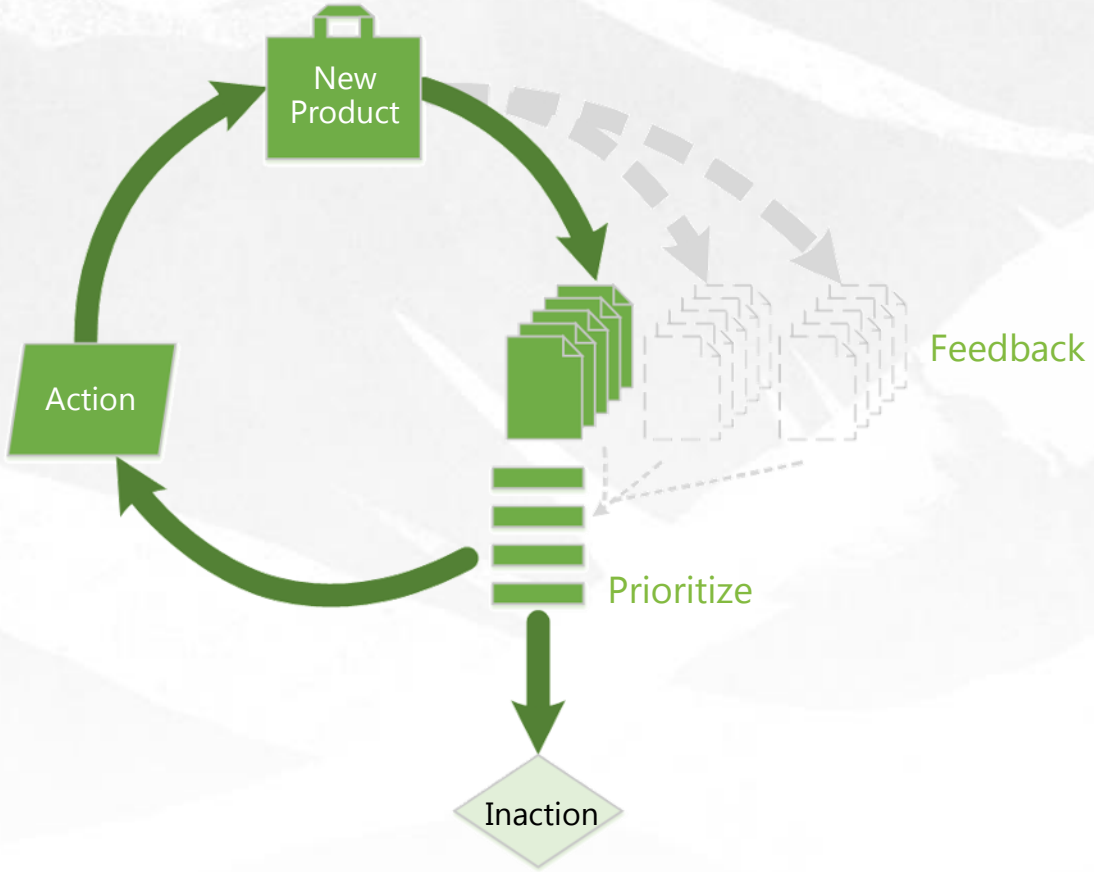
ARENANET

Lever: Feedback Sources

Feedback Sources: The diversity of feedback sources you are getting, or the number of people giving feedback

Examples of Scaling:

- Developer is only source (self)
- Team, Developer
- Stakeholders, Team, Developer
- Usability Testers, Stakeholders, Team, Developer
- Players, Usability Testers, Stakeholders, Team, Developer

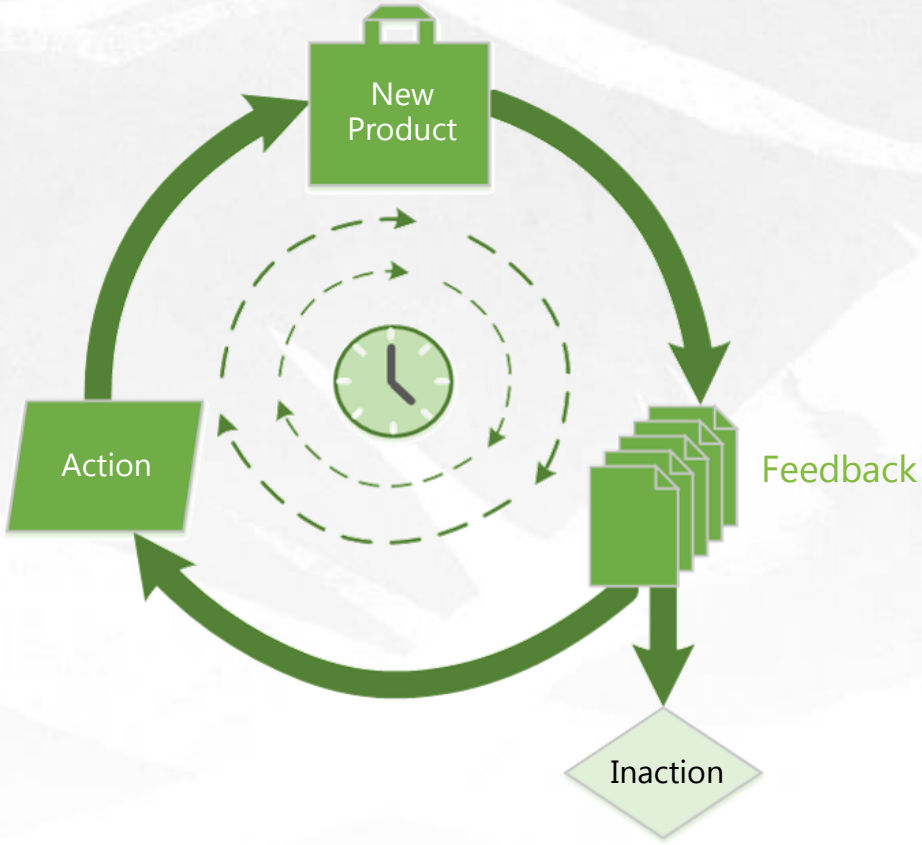


ARENANET

Levers: Cycle Length

Cycle Length or Iteration Time: The amount of time it takes to get from one version to the next.

As a result of pulling the other levers, the Cycle Length will naturally go up.



ARENANET

EXAMPLE 1:

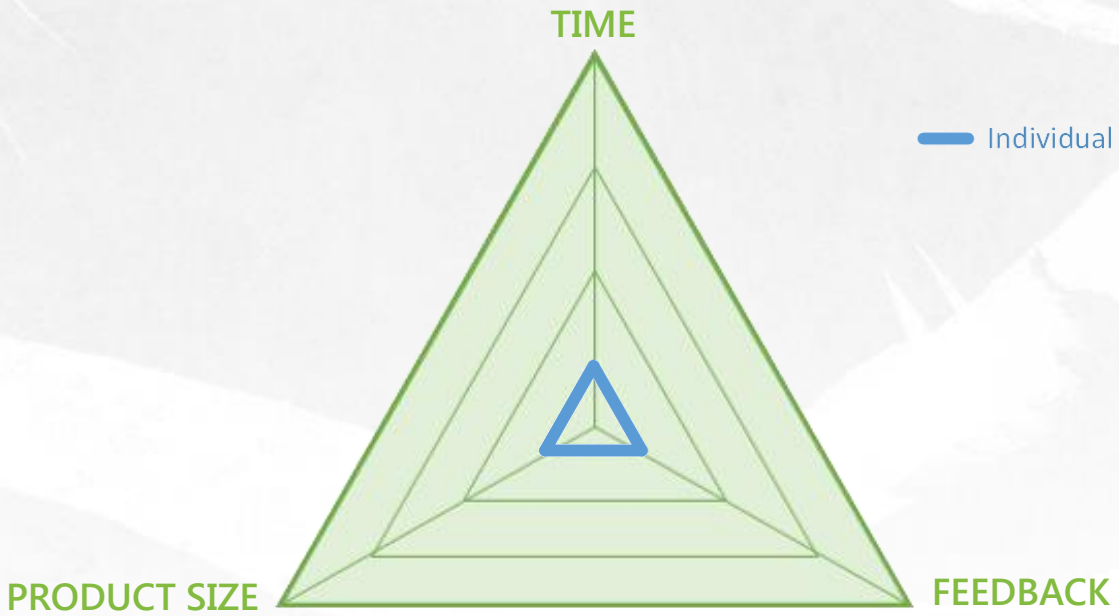
Individual Playtest

ARENANET

Example 1: Individual Playtests

- "Individual" or "Informal" Iteration
- One person implements
- Very little process

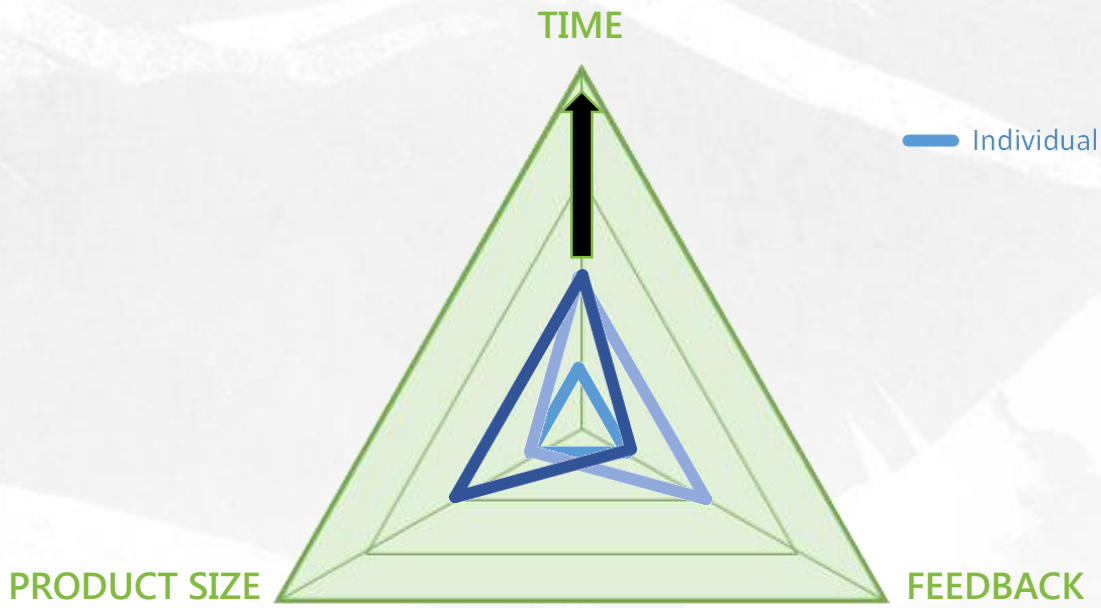
Levers:
Product Size: Small
Feedback Sources: Small
Time: Small



ARENANET

Individual Playtests

- **Challenge #1:** Bottleneck of fixing feedback
- **Techniques:**
  - Dev can see work before builds
  - Product Owner prioritizing the most important items
  - Email/Lists for Feedback, Task Tracking



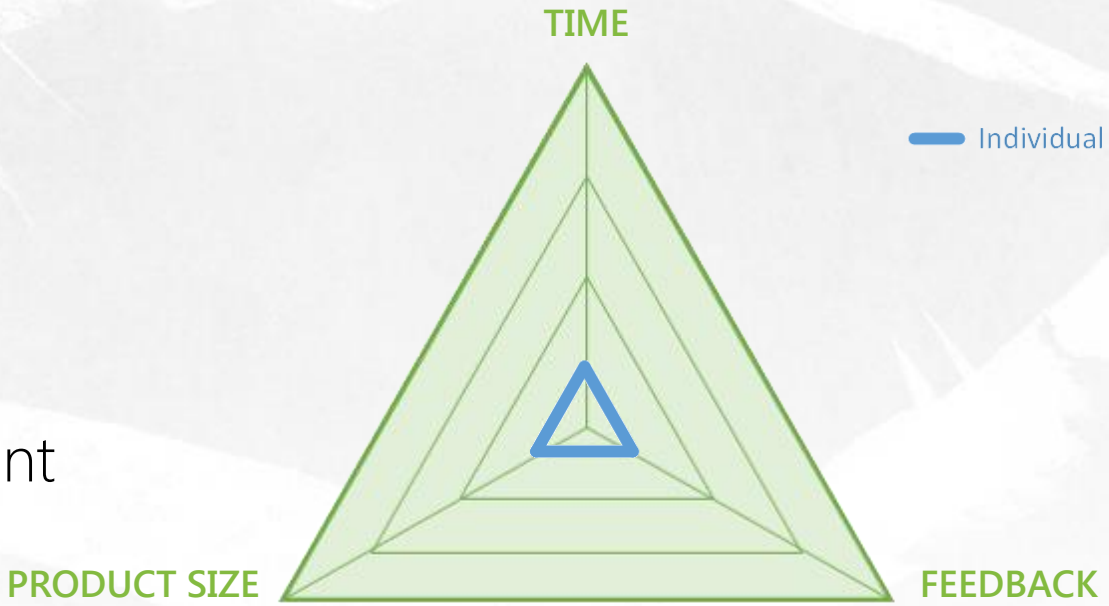
Note

When increasing size and feedback levels, the amount of time it takes to iterate will increase.

ARENANET

Individual Playtests

- Challenge #2: Tunnel Vision
- Techniques:
  - Have a Product Owner
  - Assess backlog often
  - Encourage dev to play other content
  - Might be time for more people



ARENANET

EXAMPLE 2:

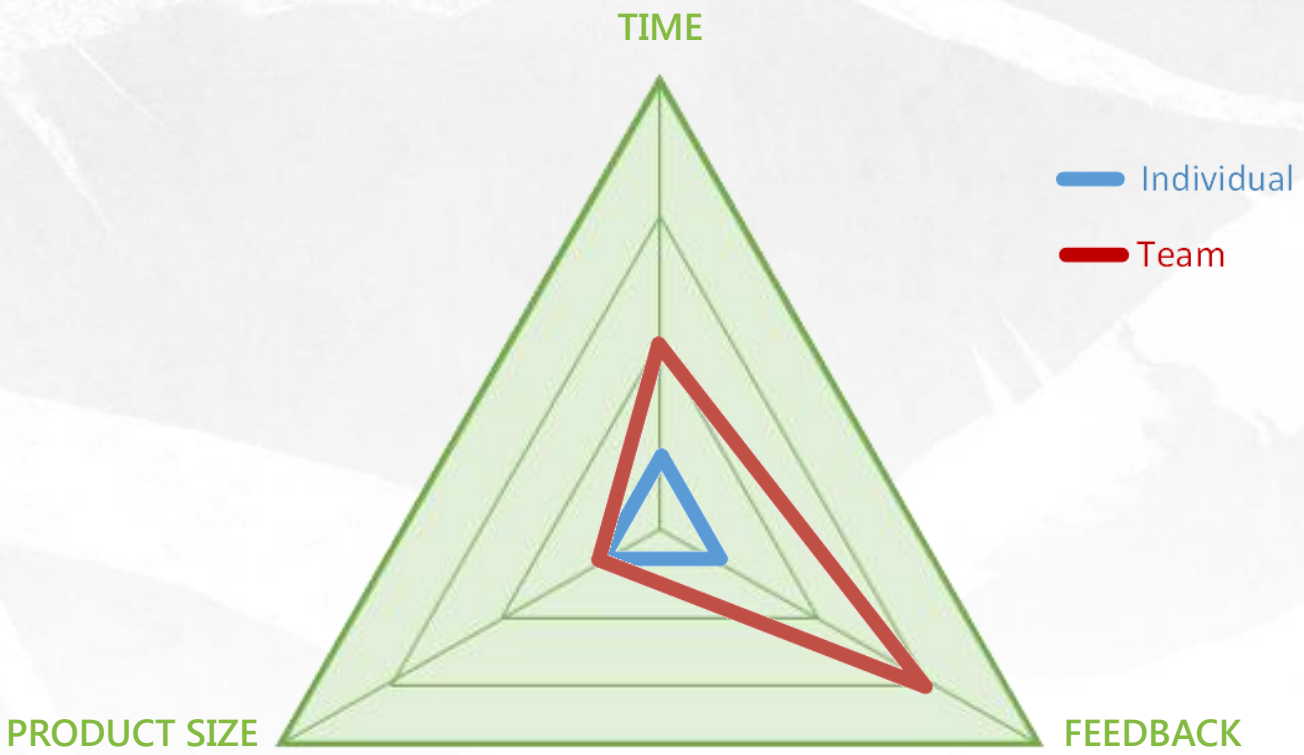
Team Playtest

ARENANET

Example 2: Team Playtests

- Playtests involve smaller versions of content
- Frequently shown to stakeholders
- Lots of feedback encouraged

Levers:
Product Size: Small
Feedback Sources: Large
Time: Medium



ARENANET

Example 2: Team Playtests

- Challenge #1: Feedback pacing or timing doesn't support development
- Techniques:
  - Managing velocity & progress
  - Constant communication about the state of the product
  - Focused Demos

Focused Demos

- Challenge #1: Feedback pacing or timing doesn't support development
- Technique: Focused Demos
  - 2x a week max; every 2 weeks minimum
  - Specific Agenda
    - Which features are polished?
    - Which features need feedback?
    - Are we looking for bugs?
    - What are the known issues?
  - Cancel or call short if agenda is a bust
  - When in doubt… get some eyes on it!
  - Immediate "Action Items" required & processed

Benefits

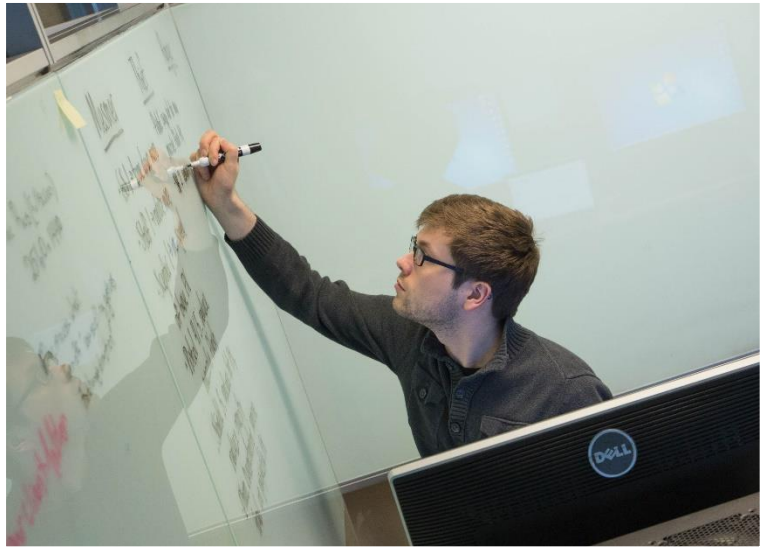
- Reveals dependencies and connections
- Scoping and Priorities become clear
- Inspiration to other developers
- Clear goals for the next demo

ARENANET

Focused Demos

Scaling for the studio

- Ensure stakeholder availability is reasonable
- Scale frequency up and down per team
- Give devs an informal option



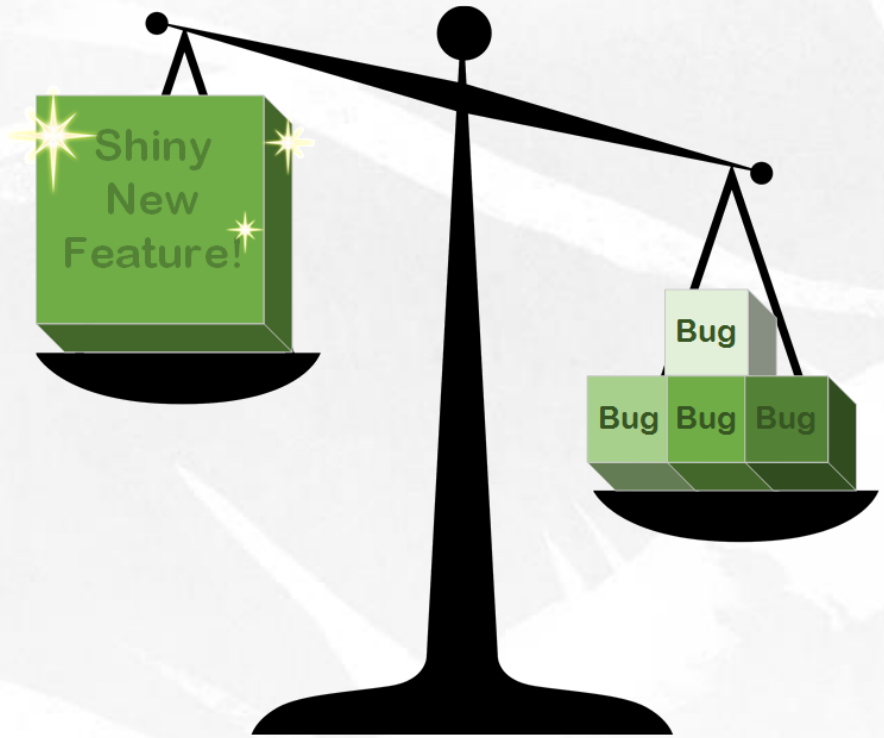
ARENANET

Example 2: Team Playtests

- Challenge #2: Scope can drastically increase with more feedback
- Techniques:
  - Stakeholders present and always negotiating
  - Shared Backlogs

Shared Backlogs

- Challenge #2: Scope can drastically increase with more feedback
- Technique: Shared Backlogs
  - Feedback and bugs are in the same backlog
  - All stakeholders meet frequently to see if dependencies are affected
  - Negotiation & trade-off discussions are encouraged



Benefits

- Quality doesn't get overshadowed by new features
- Honest conversations can happen when you see everything holistically
- Stakeholders are better able to keep scope in check

ARENANET

Example 2: Team Playtests

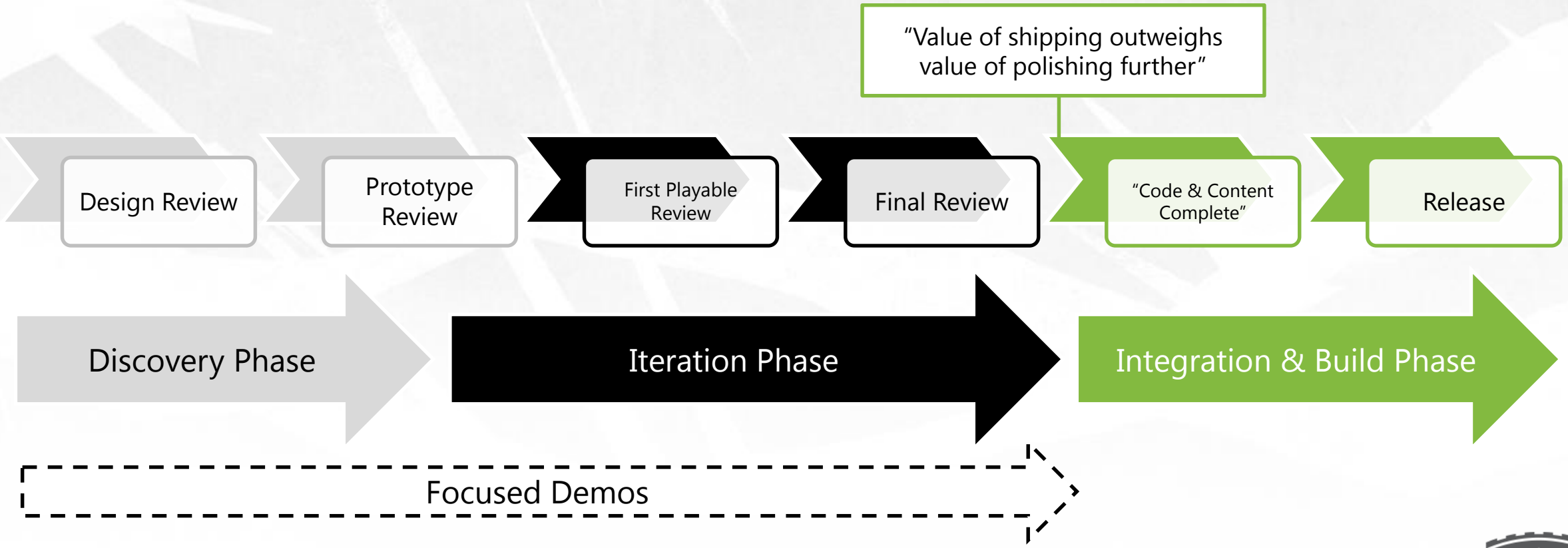
- Challenge #3: Defining the polish
- Technique: Review Gates
  - Clearly defined criteria – requires approval
  - Prototype (Optional)
  - First Playable
  - Final Review
  - "Code & Content Complete"
    - "The value of shipping now outweighs the value of polishing it further."

ARENANET

Review Gates

Dev Quote

"The power of review gates is not to add work,
but to subtract it to allow you to focus on WHAT IS MOST IMPORTANT."
–Gavian Whishaw, ArenaNet Director of Production

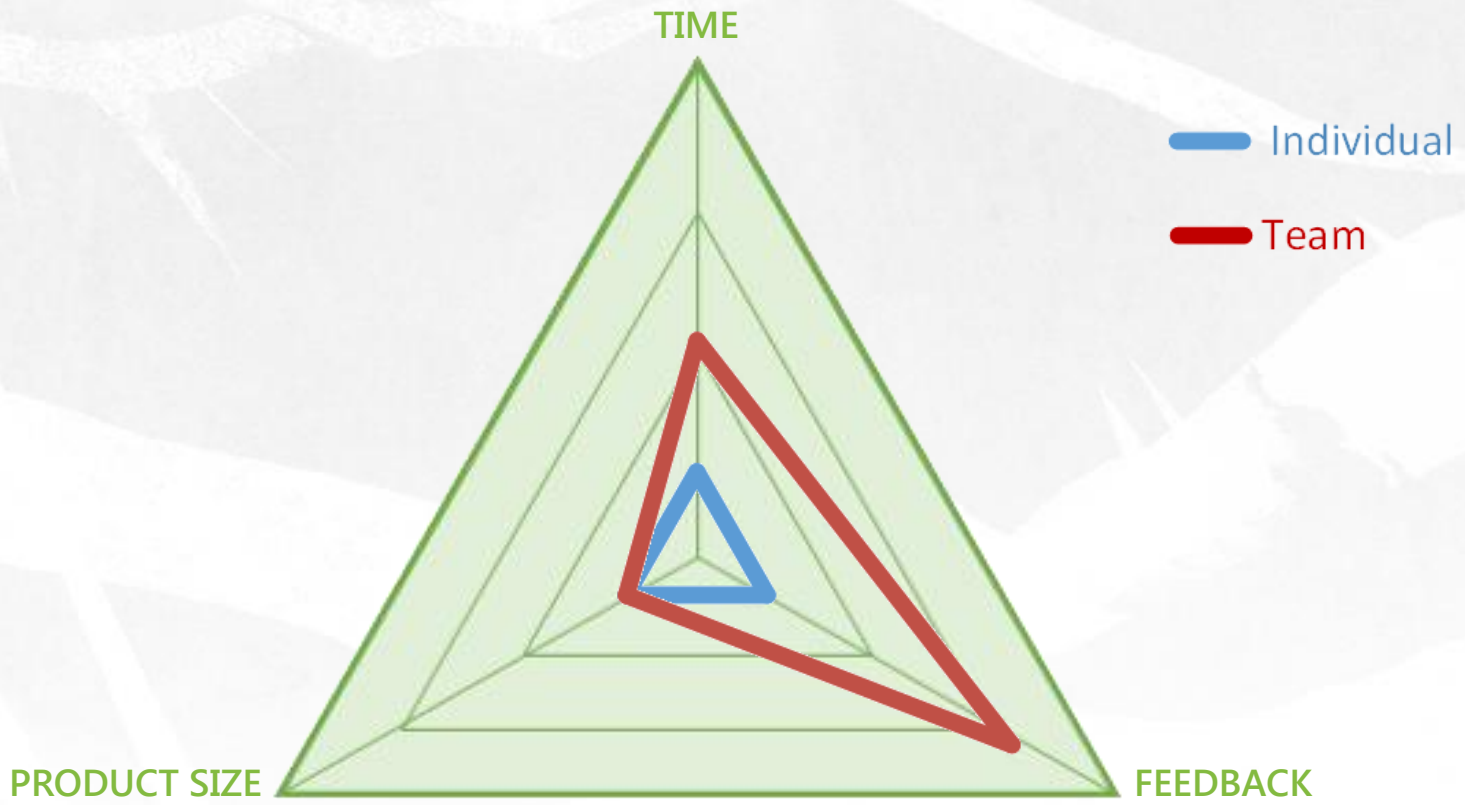


ARENANET

Recap

Individual Levers:
Product Size: Small
Feedback Sources: Small
Time: Small

Team Levers:
Product Size: Small
Feedback Sources: Large
Time: Medium



ARENANET

EXAMPLE 3:

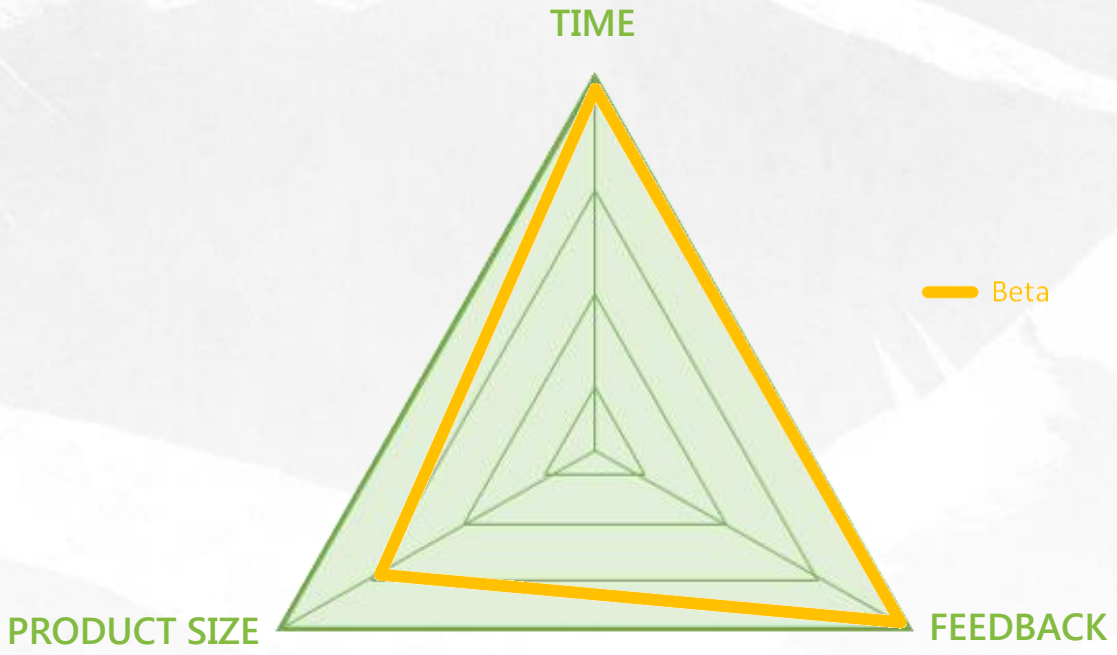
Public Playtest

ARENANET

Example 3: Public Playtests

Large-scale testing of a vertical slice of the game with external players.

Levers:
Time: Large
Product Size: Medium to Large
Feedback Sources: Large



ARENANET

Public Playtests

WvW Playtests

PvP Playtests

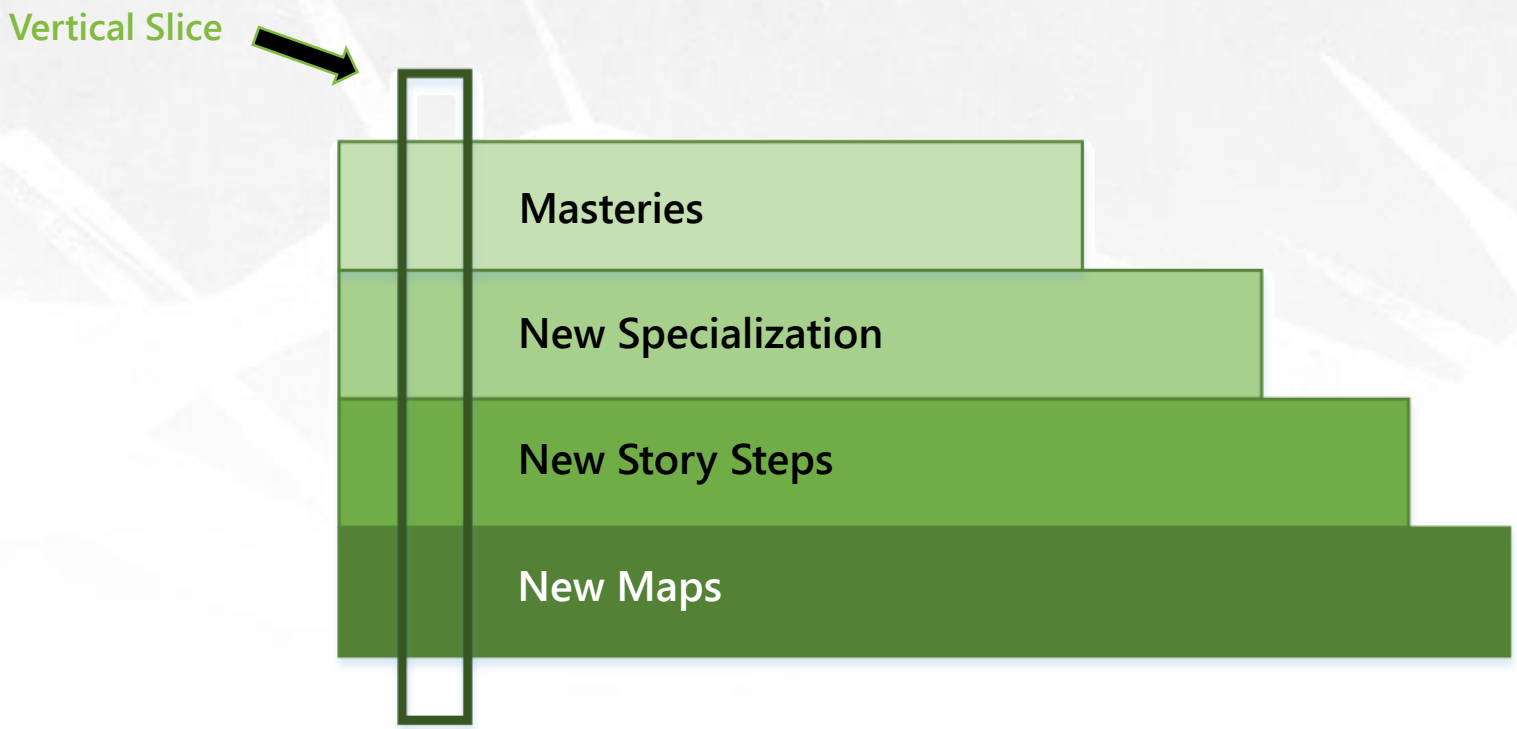
Beta Weekend Events

NOW LIVE

ARENANET

Public Playtest Challenges

- Challenge #1: Players expect a highly polished beta.
- Technique: Control quality by limiting what was in the beta build to a vertical slice.



ARENANET

Public Playtest Challenges

- Challenge #2: Betas on the Live servers – extra time, effort, and risk
  - Don't break the Live game
  - Protect ourselves from spoilers
  - Hard to change the beta plan
- Technique: Plan your betas during the prototype phase

ARENANET

EXAMPLE 4:

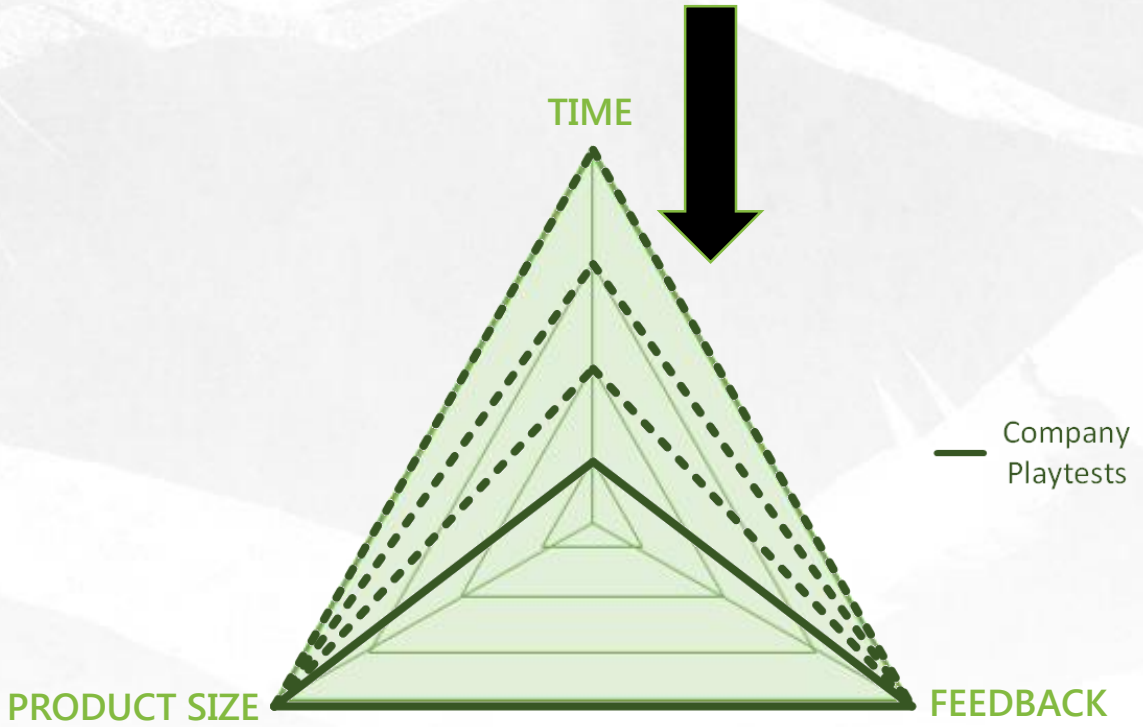
Company Playtest

ARENANET

Example 4: Company Playtests

Large-scale testing of large portion of the game with the entire company staff.

Levers:
Time: Large → Small
Product Size: Large
Feedback Sources: Large



ARENANET

Company Playtests

Goals:

- Studio plays the game as much as possible.
- Get great feedback from Developers
- Stress and performance testing
- Focus on specific features for added QA

ARENANET

Company Playtests: Groups

Casual

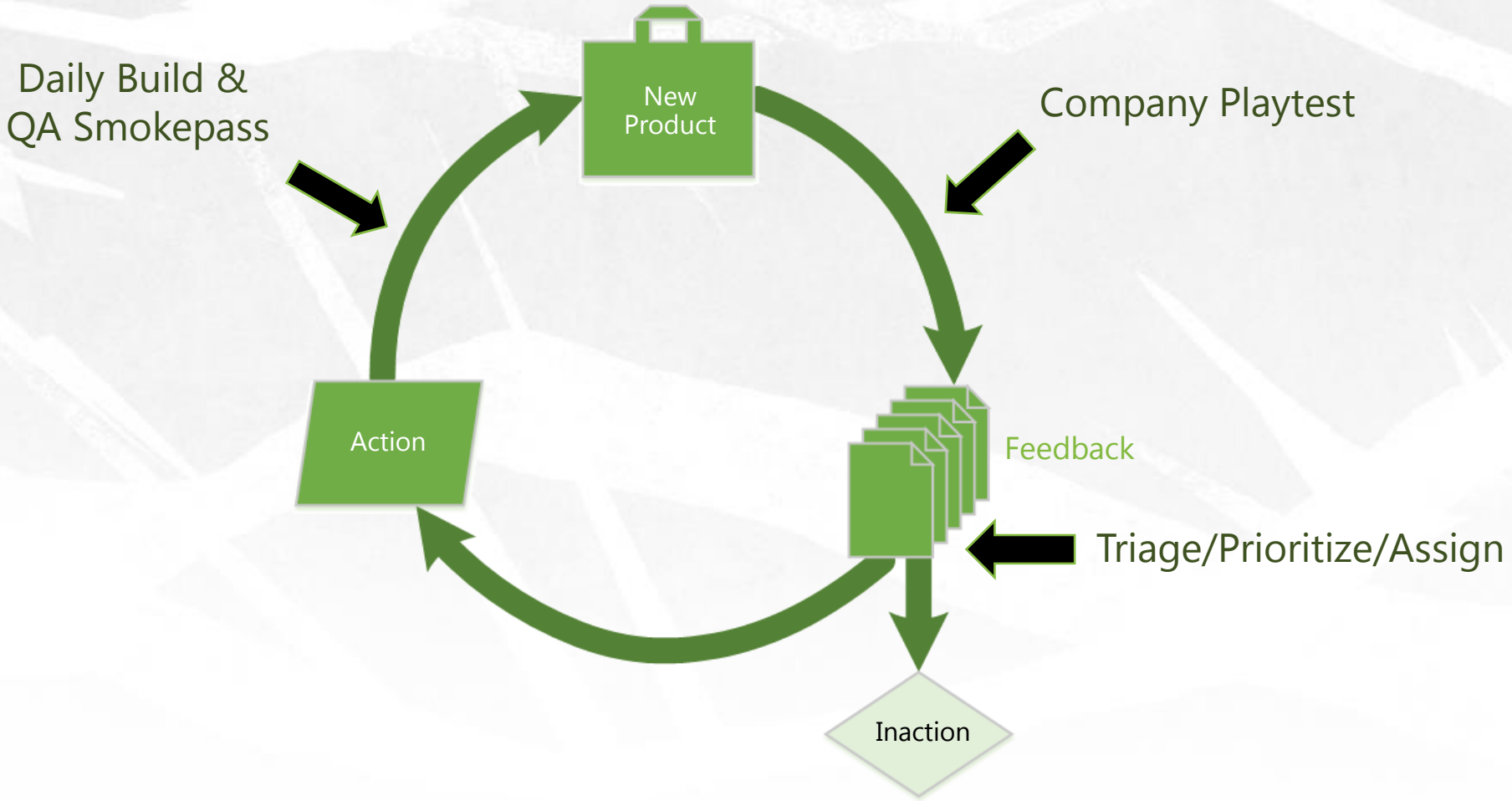
Story

PvP

WvW

ARENANET

Company Playtests: Daily Cycle



Company Playtests

- Challenge #1: Developer expectations from internal playtests
  - Content owners did not find initial feedback useful
  - Testers didn't see enough progress between iterations
- Technique: Clearly communicate the content's status and action items from the playtest

ARENANET

Company Playtests

- Challenge #2: Using company time efficiently
- Techniques: Highly coordinated playtests

Playtesting Schedule	Began 6 months before launch. Increased to 3 hours every day with the entire studio.

ARENANET

Company Playtests

	9/21	9/22	9/23	9/24	9/25
	M	**T**	**W**	**T**	**F**
Content (Producers/POs)	10-1:00PM Core Natural Progression (Fleet): Expansion Test Raids + Adventures: Expansion 11-11:30AM Sky Bosses Raids + Adventures: Expansion Designer 1	10-1PM Roots Expansion Test (All) Designer 2	10-1:00PM Army Final Review Expansion (All) Designer 2	9:30-11:30AM Gold City Defense Expansion Test (All) Designer 3 + 11:30-1:00PM Core Natural Progression (Story): Expansion Test Raids + Adventures: Expansion	10-1:00PM Core Natural Progression (Roots): Expansion Test Raids + Adventures: Expansion 11-11:30AM Sky Bosses Raids + Adventures: Expansion Designer 2
Main Usability lab	Adventures			Adventures	Adventures
Usability lab 3	Raids			Raids	Raids
Backup	Core	Core	Core	Core	Core

ARENANET

Company Playtests

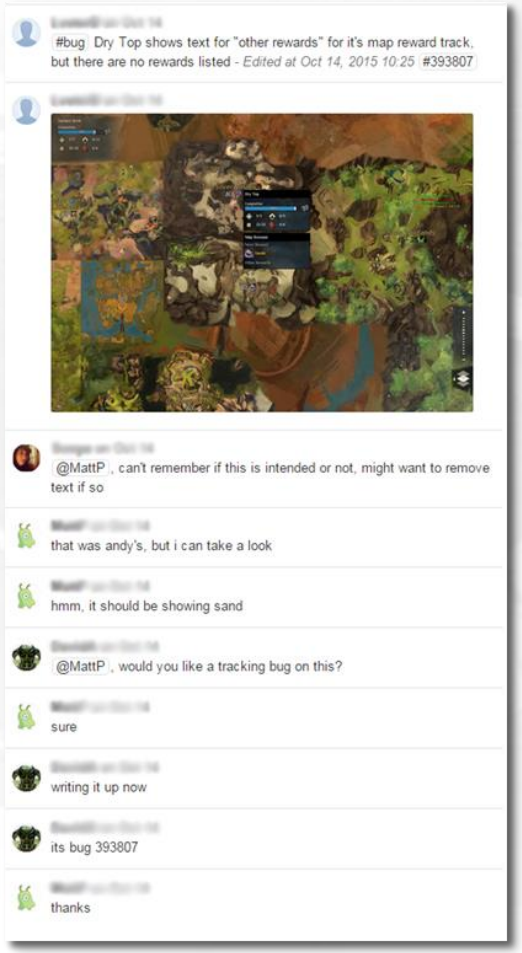
- Challenge #3: Skewed feedback through use of dev commands
- Technique: Know limitations of internal testing
  - Earning vs. Granting rewards
  - Learning vs. Cheating through natural progression
  - Experiencing vs. Skipping through content

ARENANET

Company Playtests

- Challenge #4: Making bugs and feedback useful
- Technique: Real-time chat system
  - Dedicate flows for each playtest
  - Tag relevant developers
  - Tag bugs as they come in

Dev A: Hey @DevB, I found a bug on your stuff. #Bug
Dev B: Eww…that's not good.
QA: This is bug #45234



Playtest tracking

2,560 feedback items were actioned through company playtesting

ARENANET

Timing Considerations

Questions to ask:

- Should you scale up or down your iteration types during different moments in development?
- How can you limit dependencies on other teams?
- What is your post-launch plan?

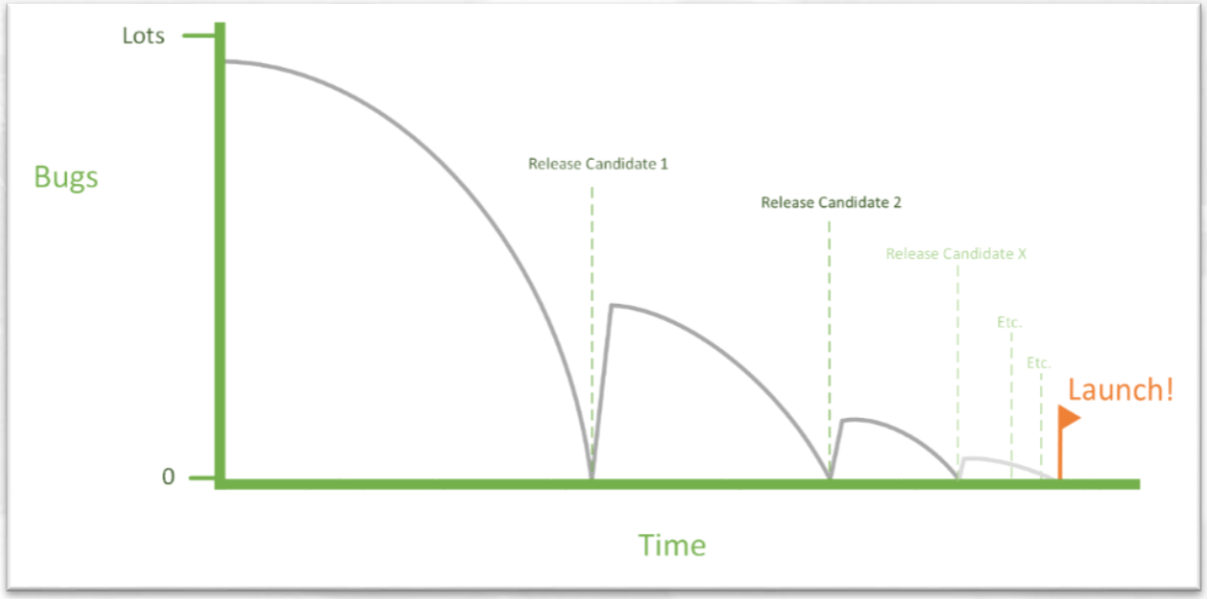
ARENANET

General Iteration Concerns

- Challenge #1 Iterating past our deadlines
- Future Technique: Limit inter-team dependencies to get a holistic view of the product.

General Iteration Concerns

- Challenge #2 : Knowing when to stop working.
- Technique: Ramp down development using release candidates

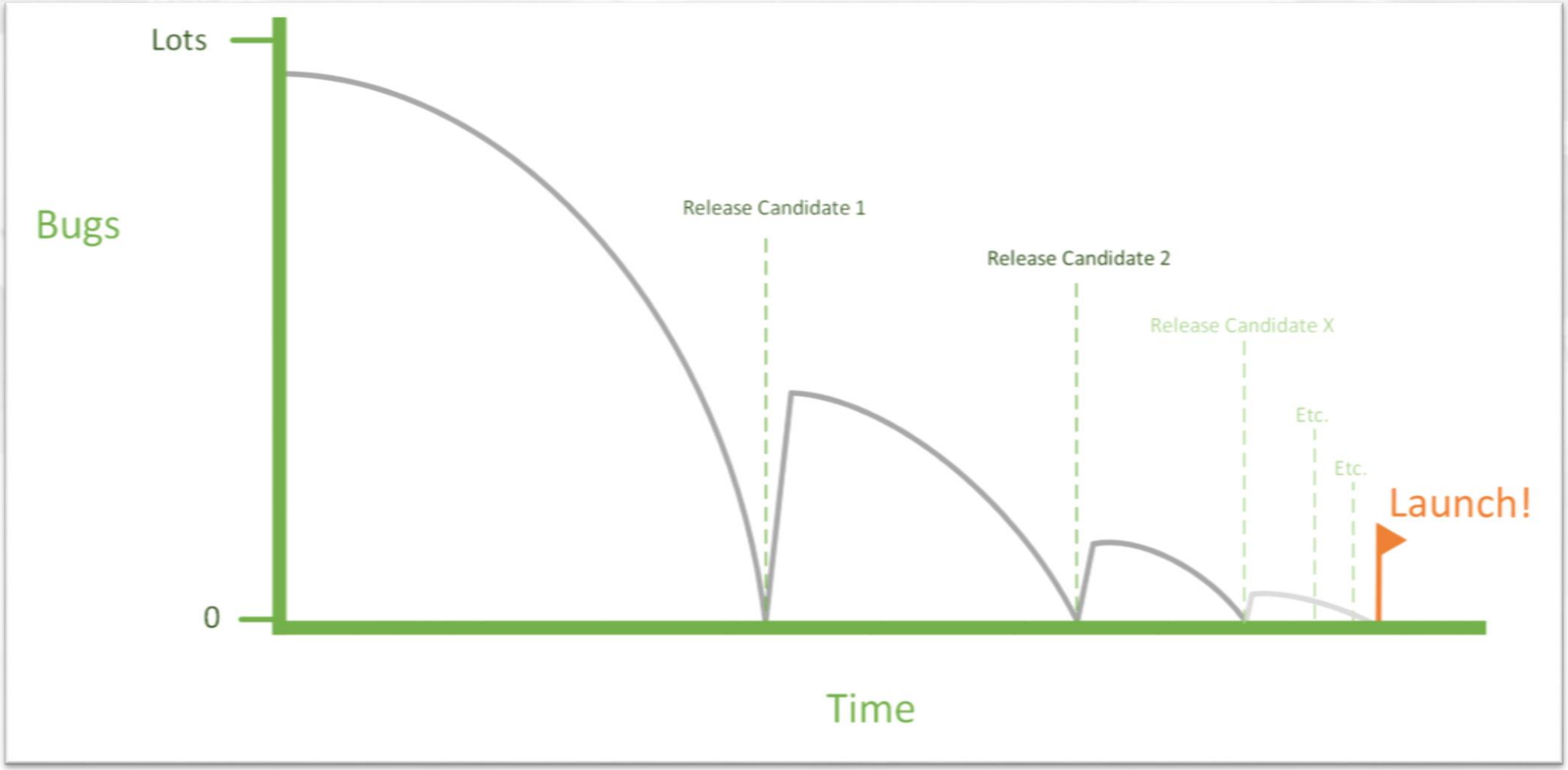


Dev Quote

"Our game would be so stable if we just stopped touching it."
~Susan Thayer, Server Lead

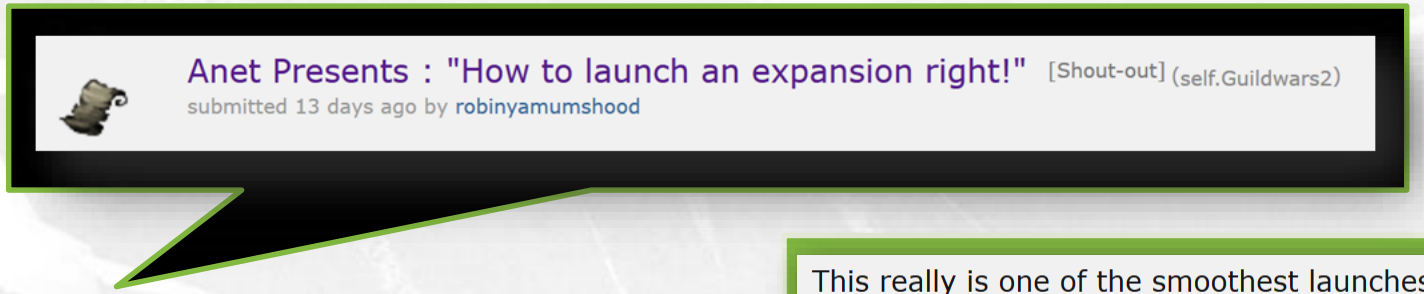
ARENANET

Release Candidates



ARENANET

Launch



This really is one of the smoothest launches I have ever heard of for an MMO/expansion.

ANet, you've really outdone yourselves on the technical side. Incredible job!

the game's subReddit in order to iterate on the launch build of *Heart of Thorns*. I believe that the speed of iteration and corrective patching is in no small part facilitated by how smooth the expansion launch was: Relatively few major bugs marred the launch, allowing the team to quickly get to our concerns and make amends.

Heart of Thorns Recognition:

- MMORPG : "Best MMO Expansion for 2015"
- TenTonHammer: "MMO of the Year for 2015"
- MassivelyOP: "Best MMO Update or Expansion of 2015"
- MMOReporter: "Game of the Year for 2015"

ARENANET

Takeaways from HoT Iteration

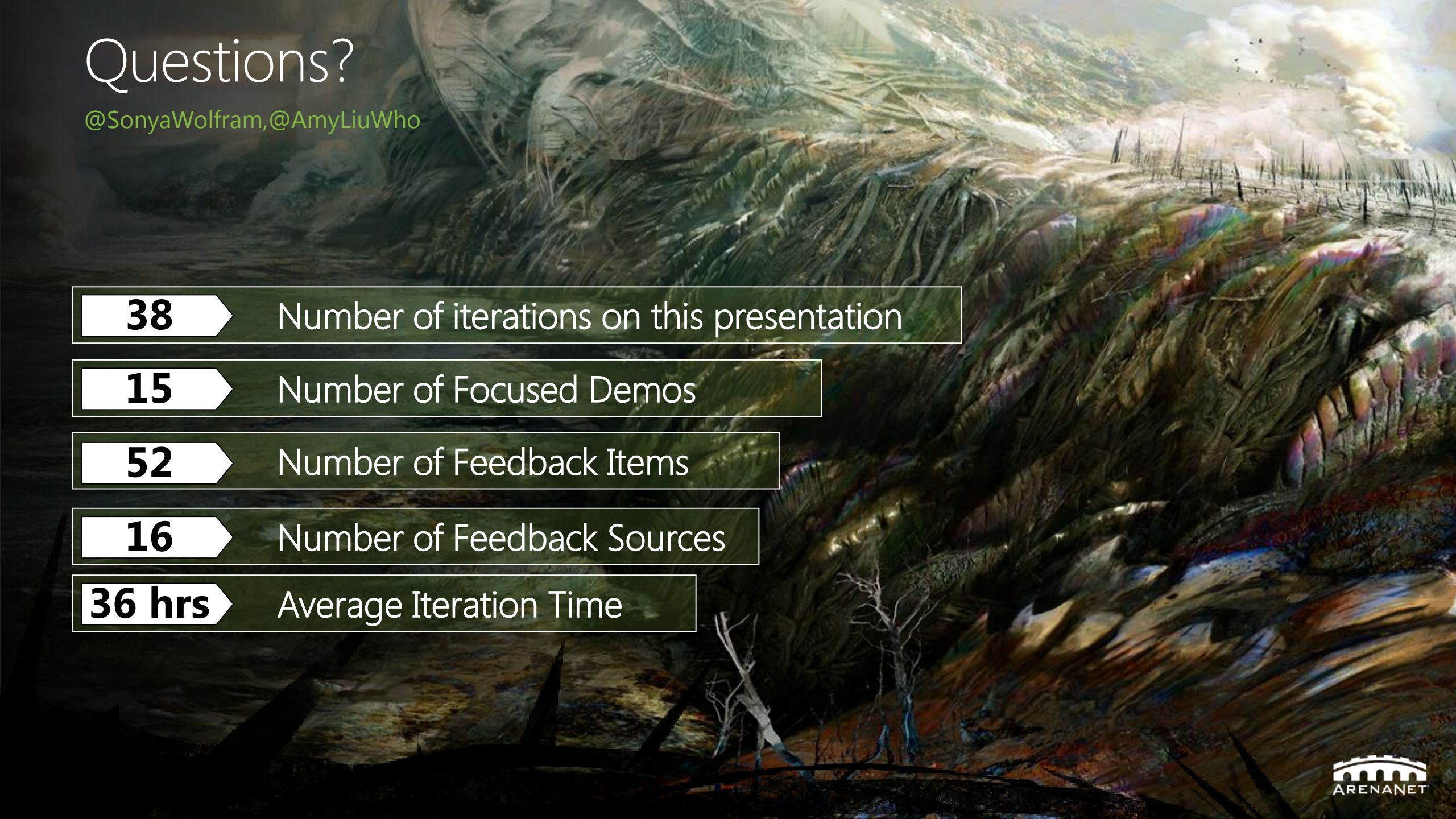
- Have a variety of iteration playtest types.
- Require a beta plan during pre-production
- Limit inter-team dependencies and weigh the costs of remaining work daily

Takeaways from HoT Iteration

- Use techniques to reduce iteration time to combat the expense of company playtesting
- Have clear goals and specific agenda each time that you ask for feedback.
- Encourage playing the game!

Questions?

@SonyaWolfram,@AmyLiuWho



38	Number of iterations on this presentation
15	Number of Focused Demos
52	Number of Feedback Items
16	Number of Feedback Sources
36 hrs	Average Iteration Time

ARENANET