

Defying Expectations: 7 Lessons from The Sims FreePlay

Mavis Chan
Lead Product Manager
Electronic Arts Firemonkeys



About Me

- Lead Product Manager at Firemonkeys
 - 5+ years with EA
 - 3+ years on The Sims FreePlay
- Previously from Ubisoft Singapore
 - Started as Monetisation Analyst
- Business background
- 8 years spent working on F2P titles on PC and Mobile



Agenda

- Some History
- Product Life Cycle
- Stages + Lessons
- Takeaways

Some History

- Firemonkeys, an EA mobile studio in Melbourne, Australia
- Merger between IronMonkey Studios and Firemint in 2012
- Games – The Sims FreePlay, Real Racing 3, Need for Speed: No Limits, Flight Control, Spy Mouse



What did TSFP start out being?

Premium



F2P

iPad Only



Mobile +
Tablet

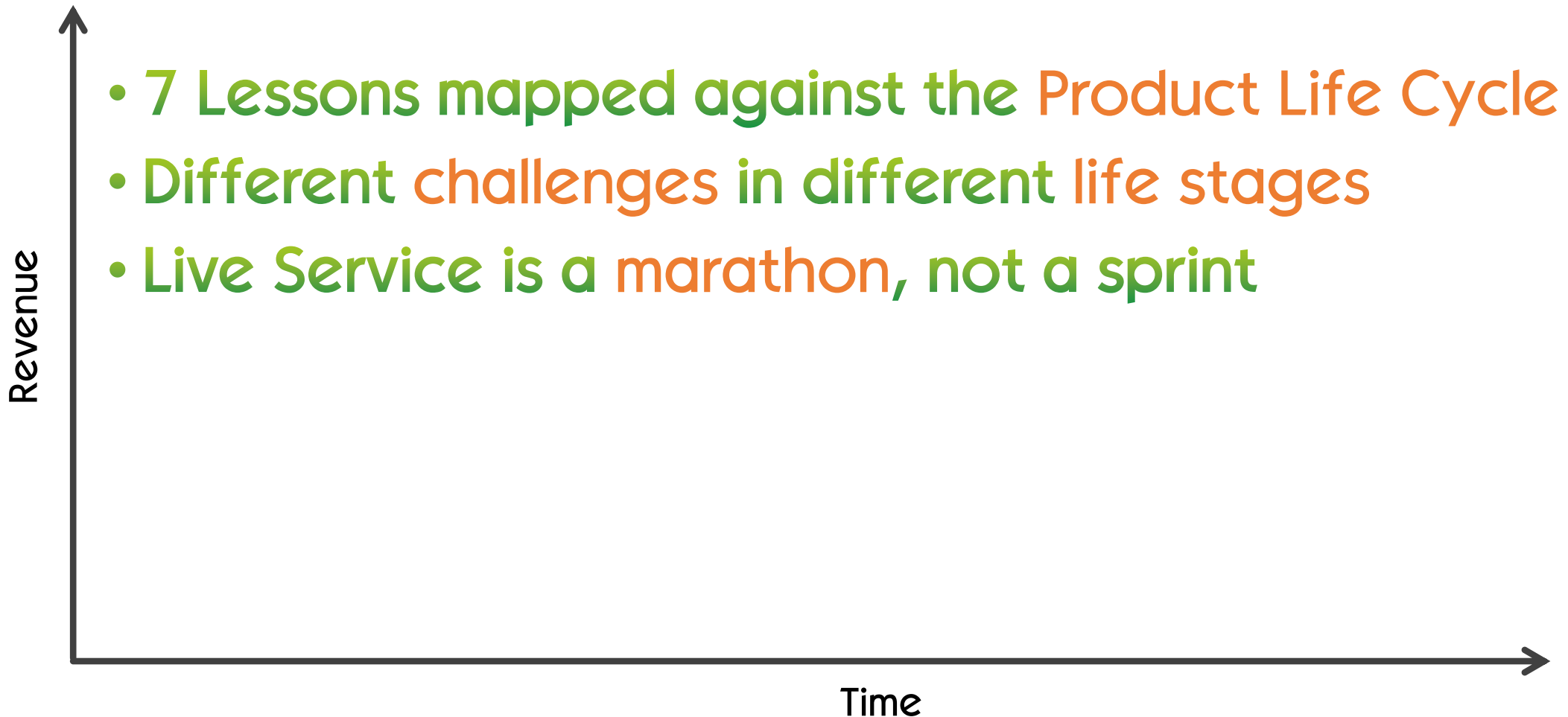
1 to 2 Year
Expected
Lifetime



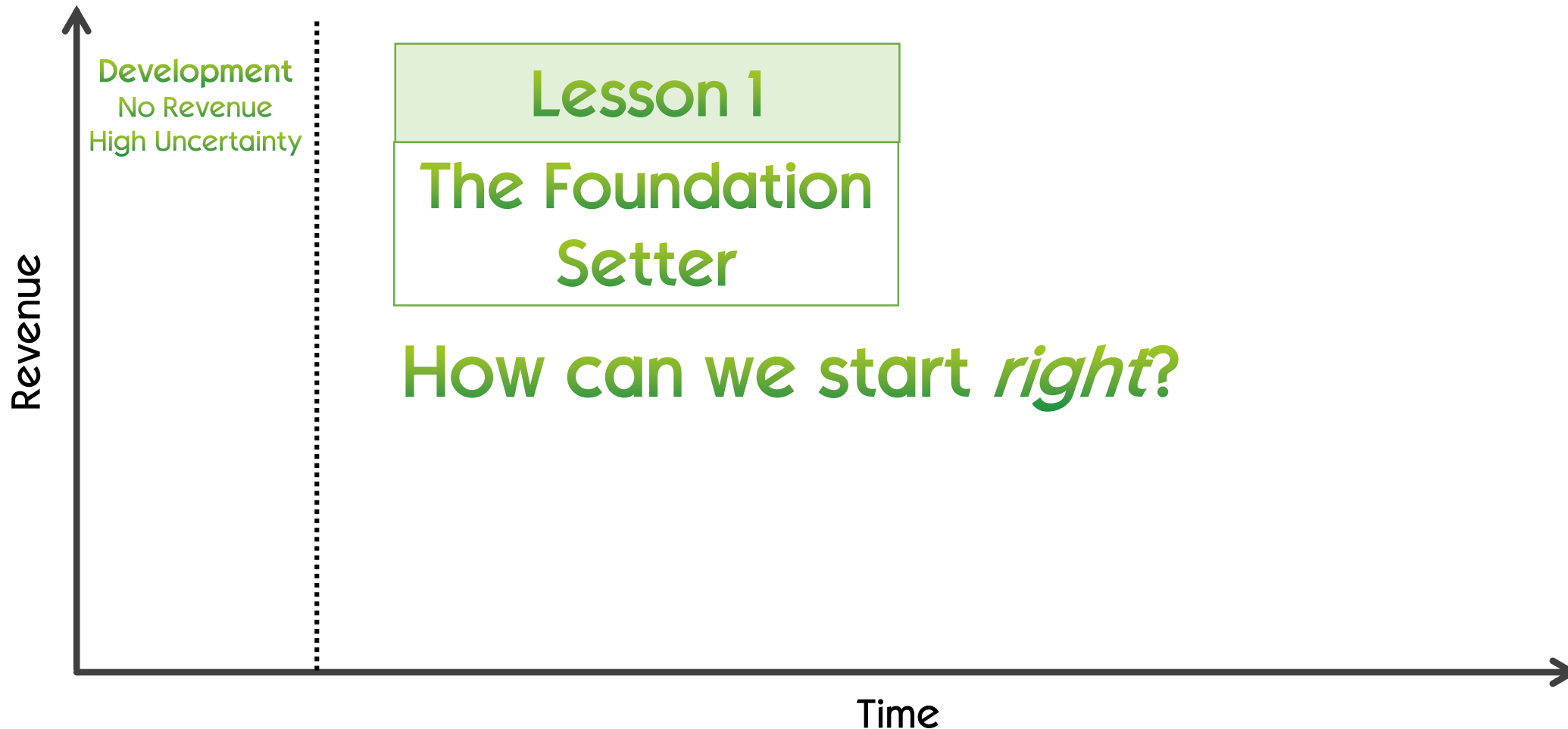
7+ Year Live
Service



Product Life Cycle for F2P Games

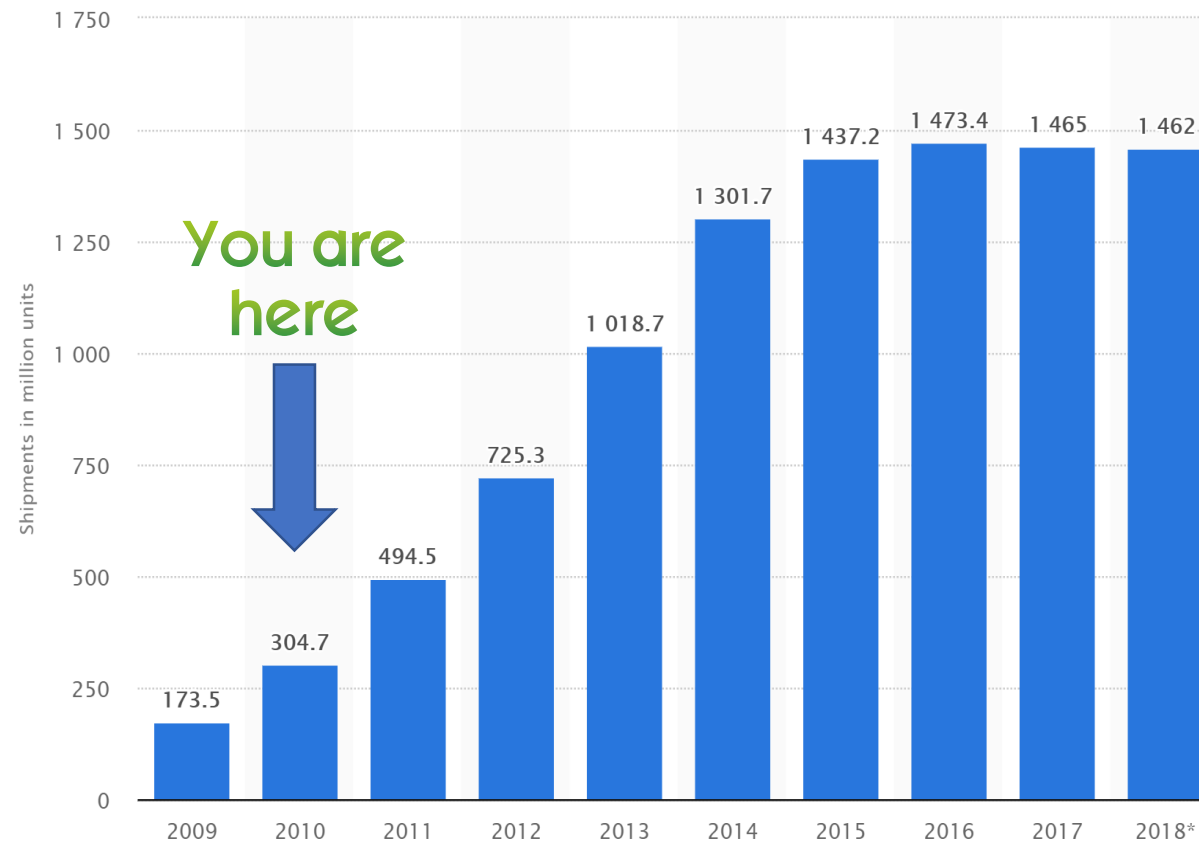


Stage 1: Development



Development = Risky Business

Global Smartphone Shipment Units (in M)



Development = Lots of Questions

Where do we start?

What do you build?

What should the
gameplay be?

What should the Meta be?

How are we going to attain, retain, and monetise?

How can we make good
decisions while lacking data?

Leverage Strengths and Manage Risks



Leverage Knowns

- Mobile studio
- Similar Team
- Shipped premium Sims titles on mobile

Limit Unknowns

- F2P Business Model

Mitigate Unknowns

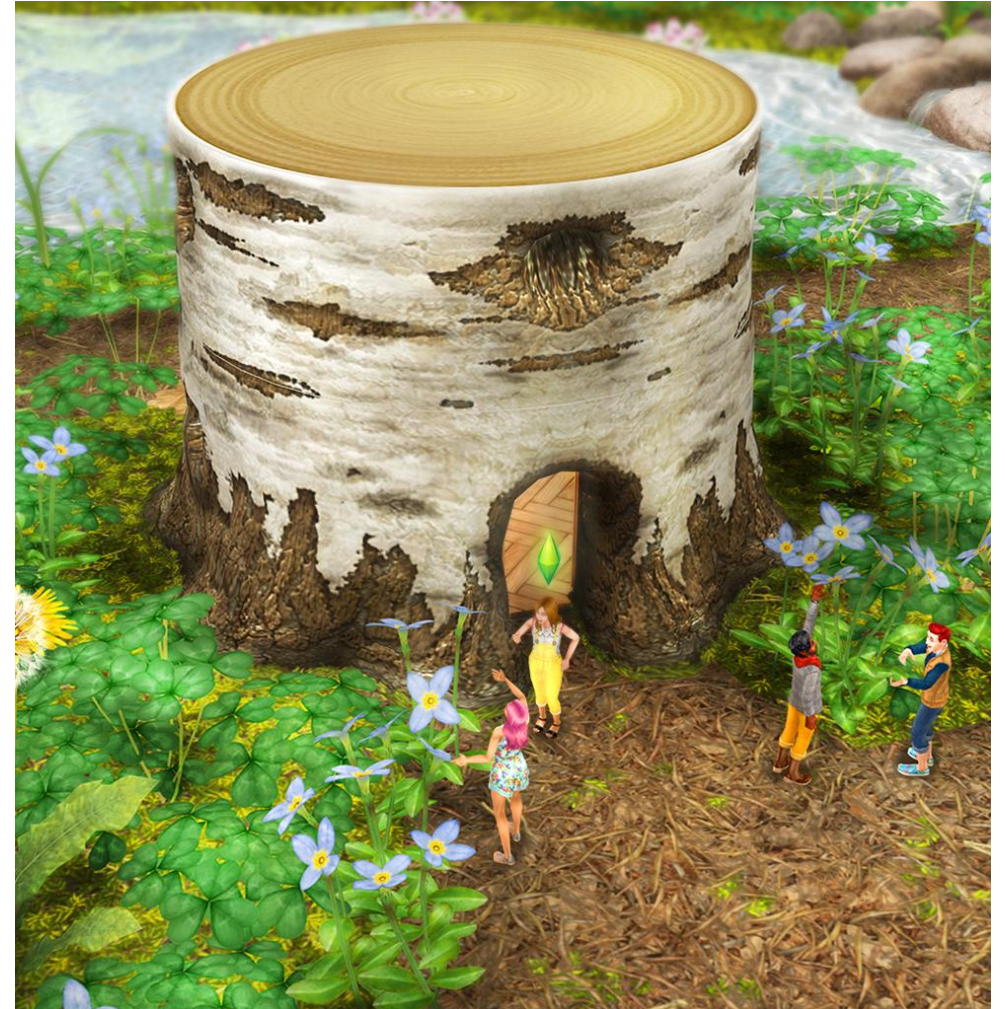
- Study successful competitors in the market
- Timeboxed development

[illegible]

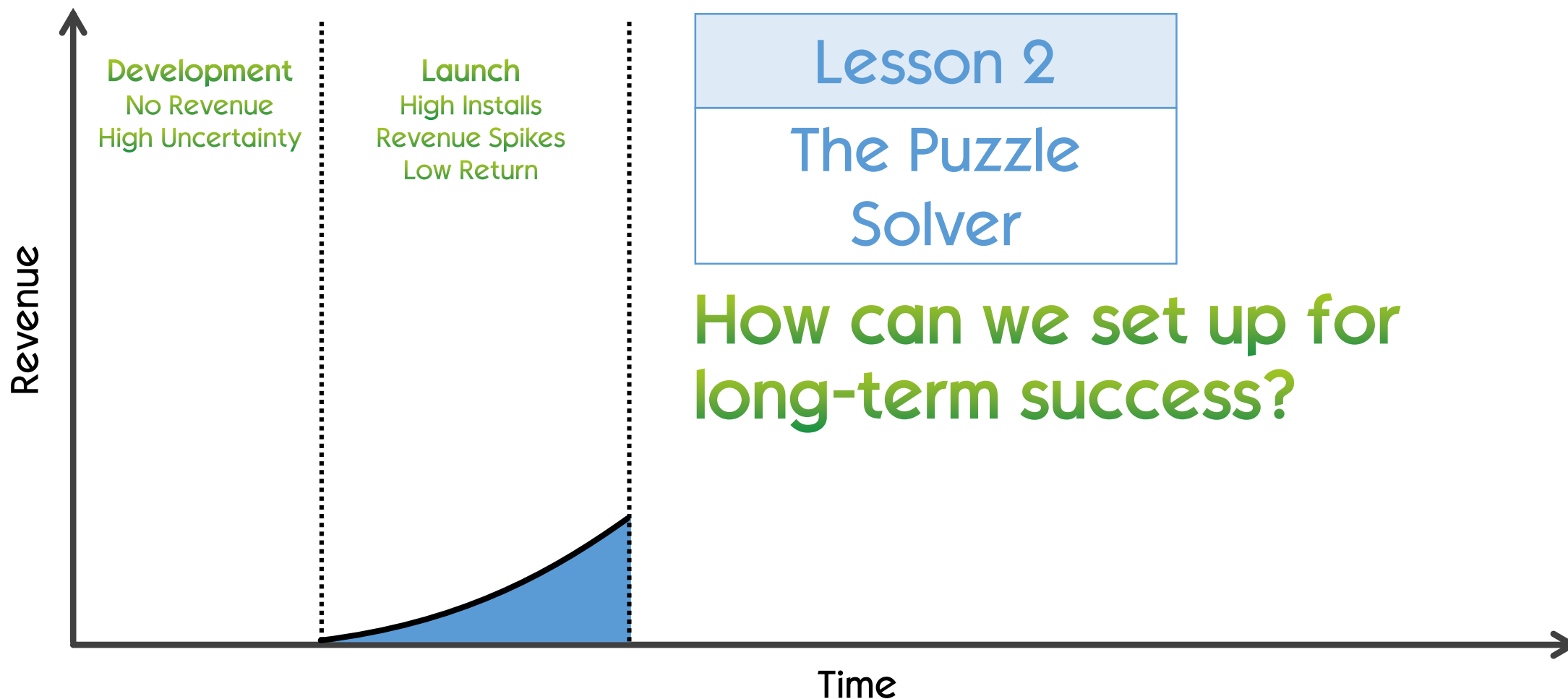
Lesson 1 – The Foundation Setter



- Be honest – Leverage what you know, and mitigate what you don't
- Resist the temptation to innovate on everything
- There is no recipe for success

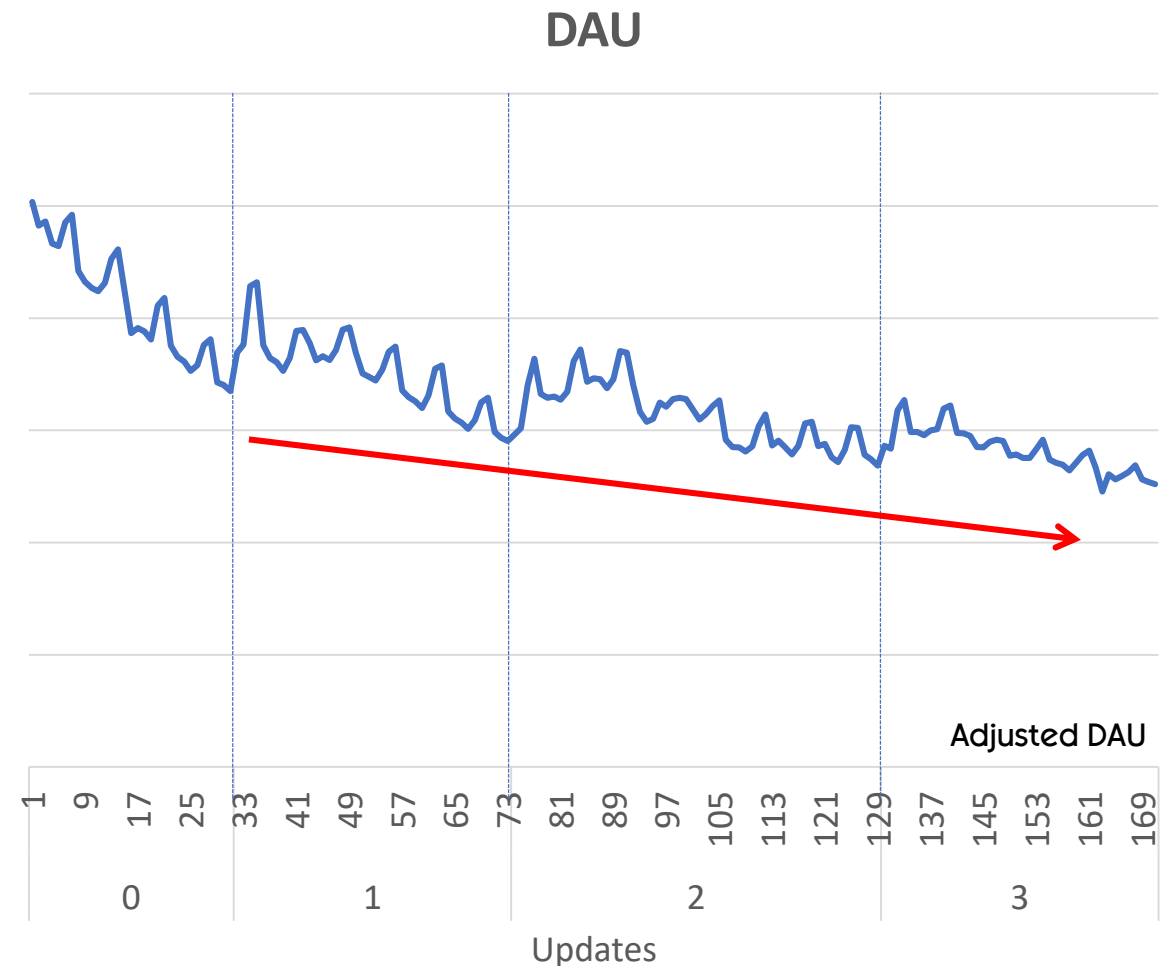


Stage 2: Launch



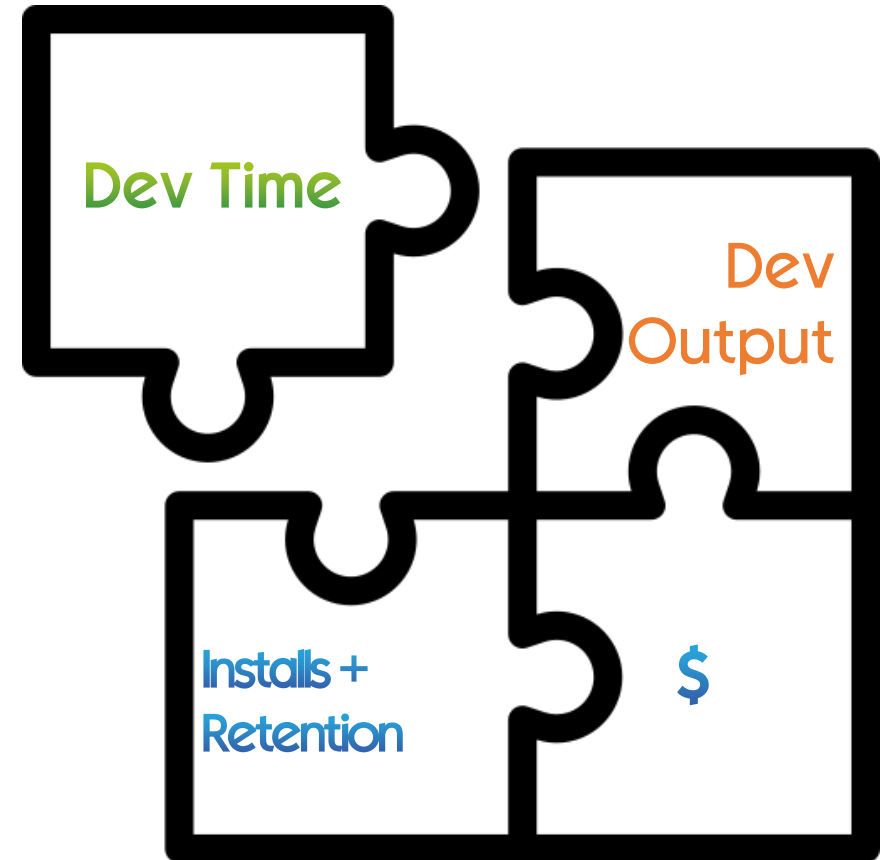
TSFP at Launch

- Featuring and Strong IP = Strong installs
- But....our initial live service plan wasn't quite working out
- Despite good installs, DAU kept sliding



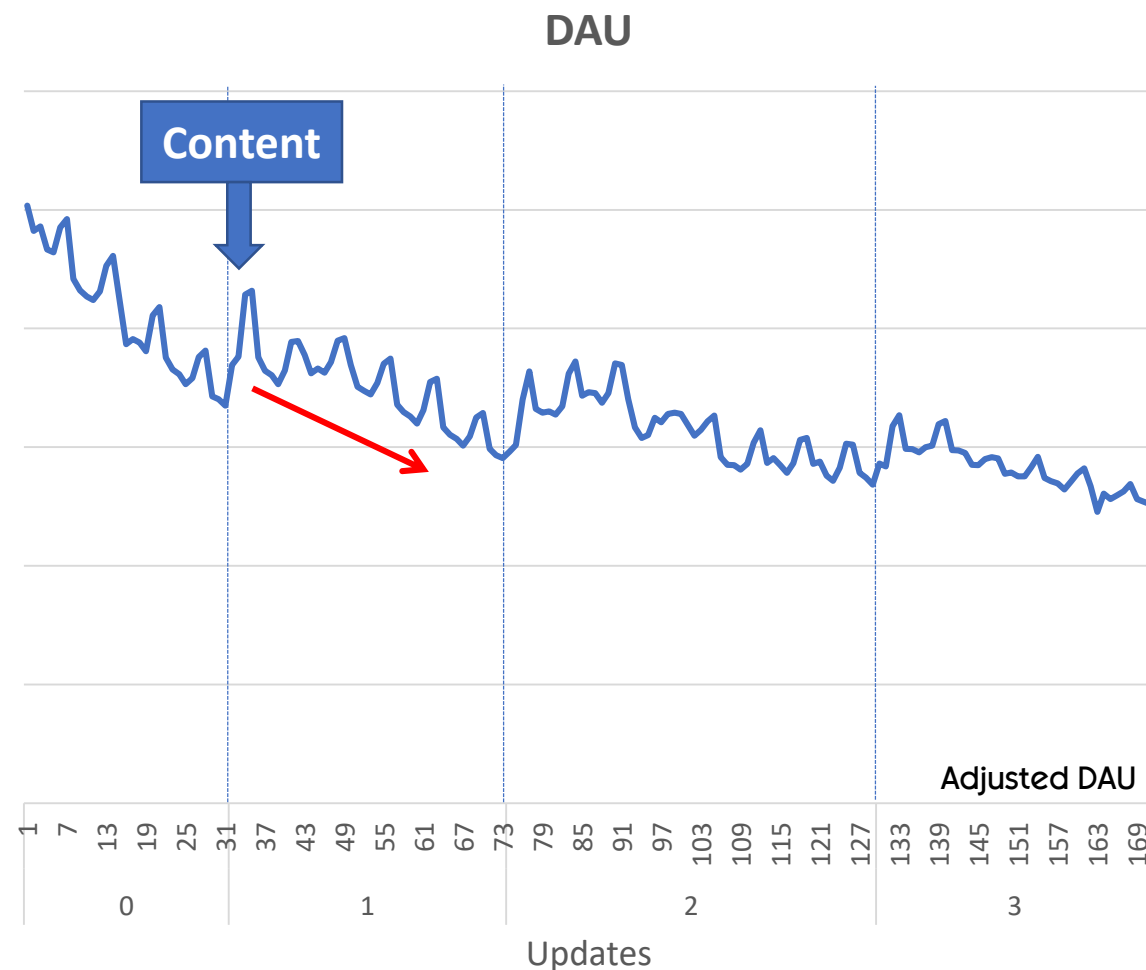
Live Service is a Puzzle...under Time Pressure

- Using Development Time to get the best return on Player Engagement and KPIs
- Development Time is split into maintenance, events, content, new features
- Manage the input and output most effectively to get the best results and achieve growth



Finding the Problem

- Through updates, we put content **directly** into the build mode store
- Failed to **sustain** DAU as players could **consume** the **content** immediately
- We needed to **retain** players for longer



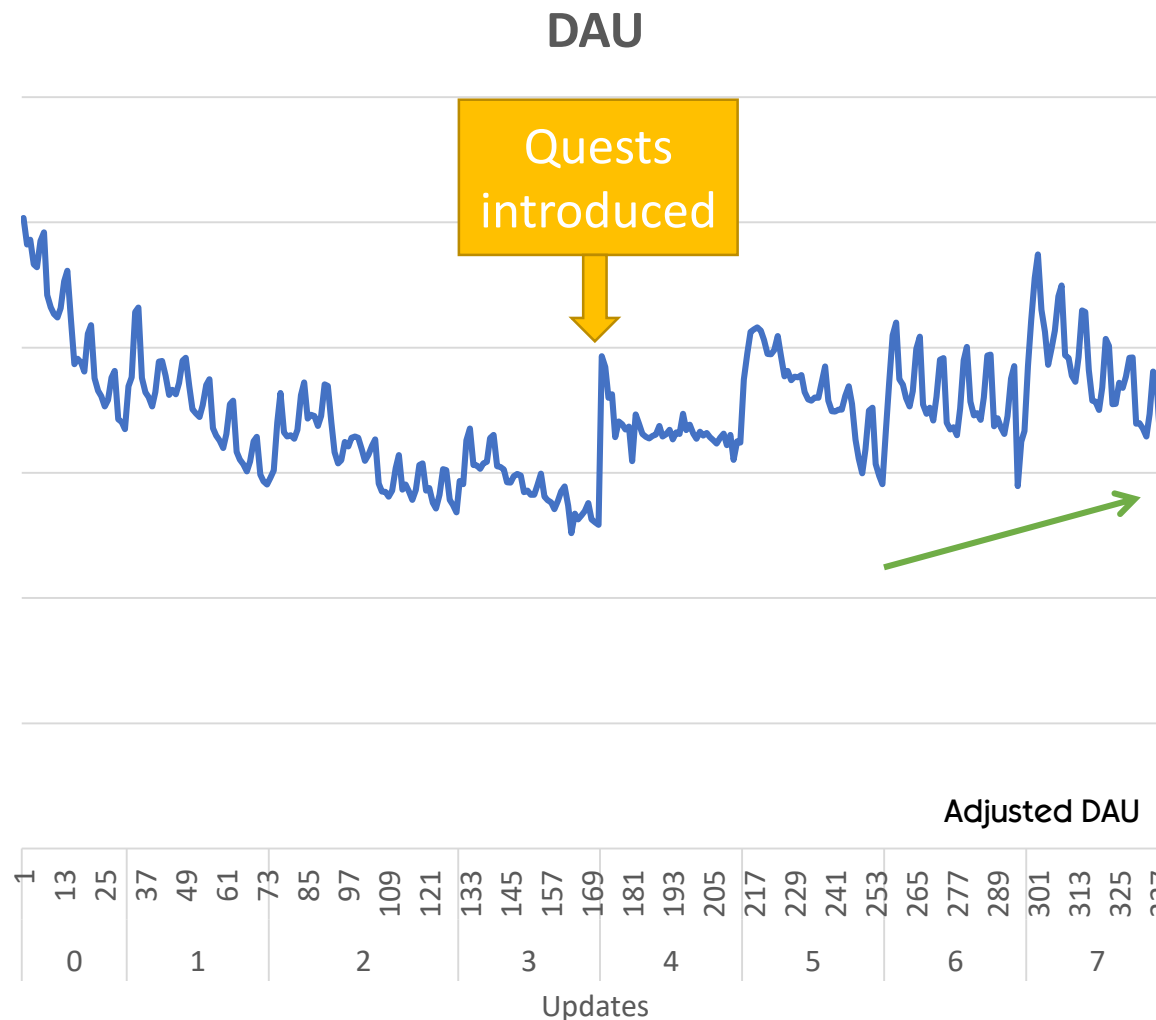
Solving the Puzzle

- Experimented with putting content behind **Quests**
- Players had to **engage** in a quest to win **content** at the end
- That worked to **extend the content lifespan** solving the puzzle



DAU Turnaround

- Finding the problem and solving the puzzle
- DAU started to grow gradually over updates



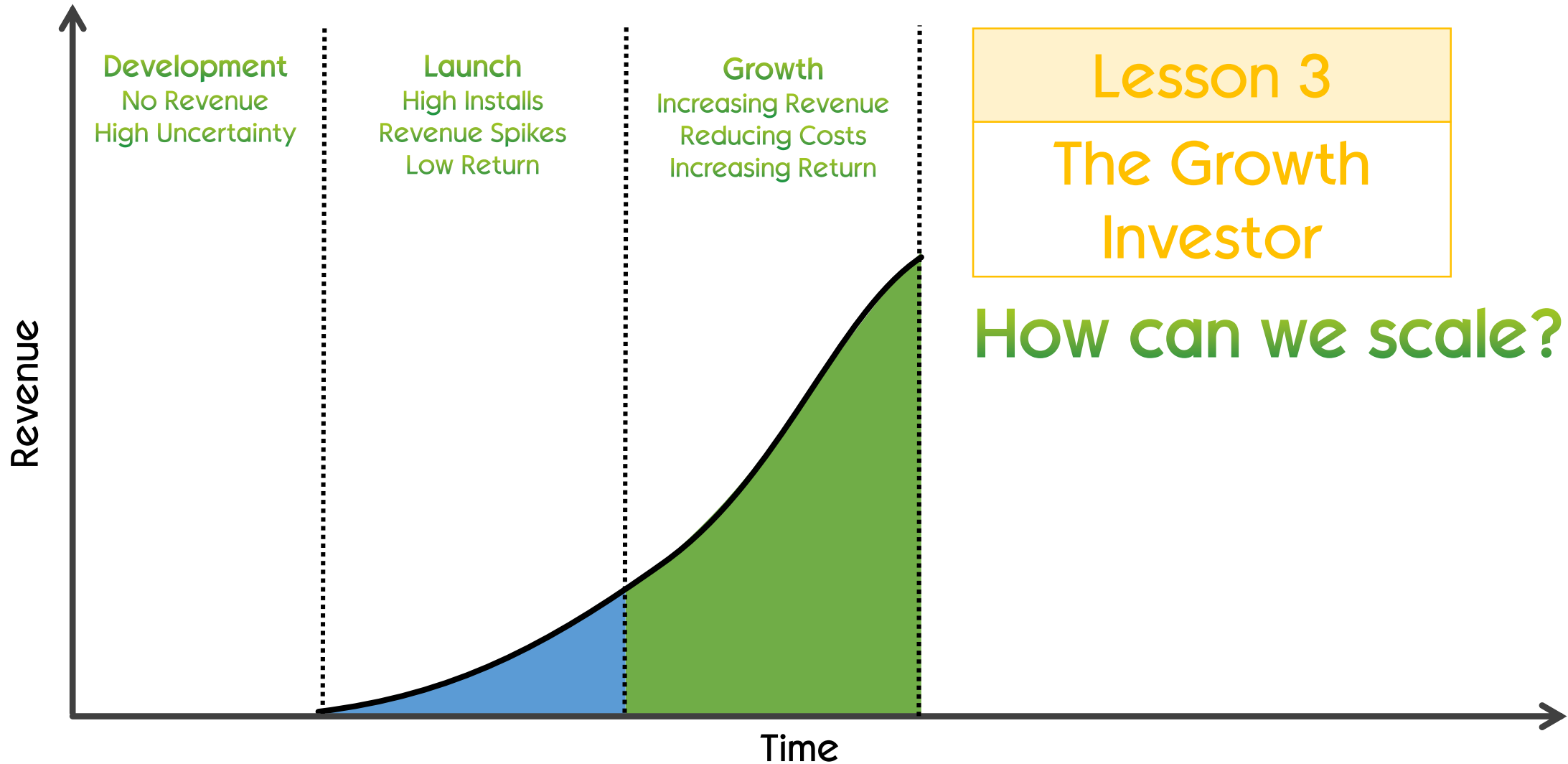
Lesson 2 – The Puzzle Solver



- 1) Identify **key** problems at launch
- 2) Live Service is **puzzle** of moving pieces
- 3) Solving the Puzzle = **Viable Product** through Live Service



Stage 3: Growth



Increase Content Production

Since we knew how to use Content to drive KPIs, we invested further by,

- 1) Grew internal art team
- 2) Worked on outsourcing art
- 3) We have steadily been adding assets every update

Added benefit of making it harder for new titles to compete in the Simulation space



Reduce Costs through Tools

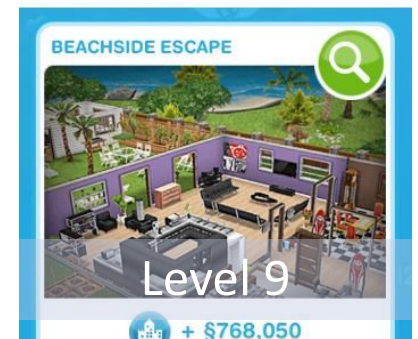
- Tools to reduce engineering time and empower other disciplines
 - Data Tool for Tuning
 - Quest Template Sheets
 - Sales Tool
 - Scheduler Tool
- Economies of Scale – The more we do it, the better we get



Feature Optimisation

Build on top of what we knew worked

- Chained the quests together and locked new features behind them
- That helped to strengthen the baseline DAU further as now players had more things to work on between updates



User Generated Content

- Houses are very expensive for us to build, but highly demanded and valued by players
- We built Architect Homes for players to submit their houses
 - Rotates every 2 weeks
 - Players get to preview and purchase
- Removes dev bottleneck



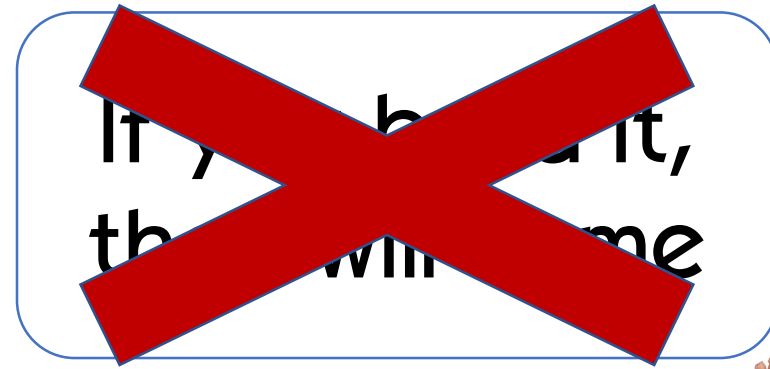
Lesson 3: The Growth Investor

- 1) Invest when you know what works
- 2) Gain efficiencies in content creation and consumption
- 3) Simulation games get stronger over time
- 4) Opportunity cost if you can't scale when the time is right



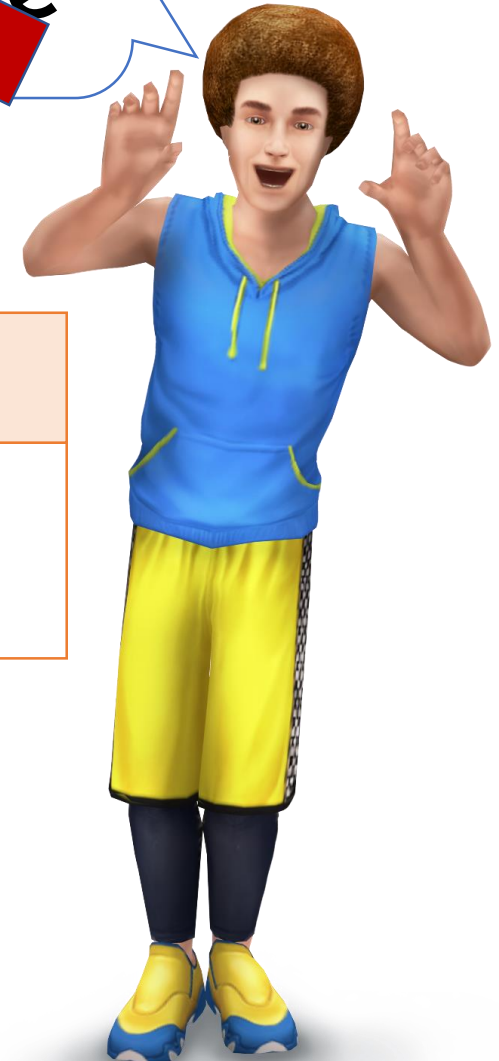
Paradox of Content

- We know we **need** to build content
- We know **how** to build
- But **what** should we build?
- We've learnt by getting it **wrong**
 - Difference between good and dud content can be **more than 50%**



Lesson 4

The Keen Listener



Surprise! Players will tell you

- TSFP audience skews **younger**
- Highly **vocal** with lots of opinions

External Channels

- Social – Facebook, Twitter
- Forums
- Website
- Customer Support

Internal Channels

- CSAT Surveys
- UXR
- Game Telemetry

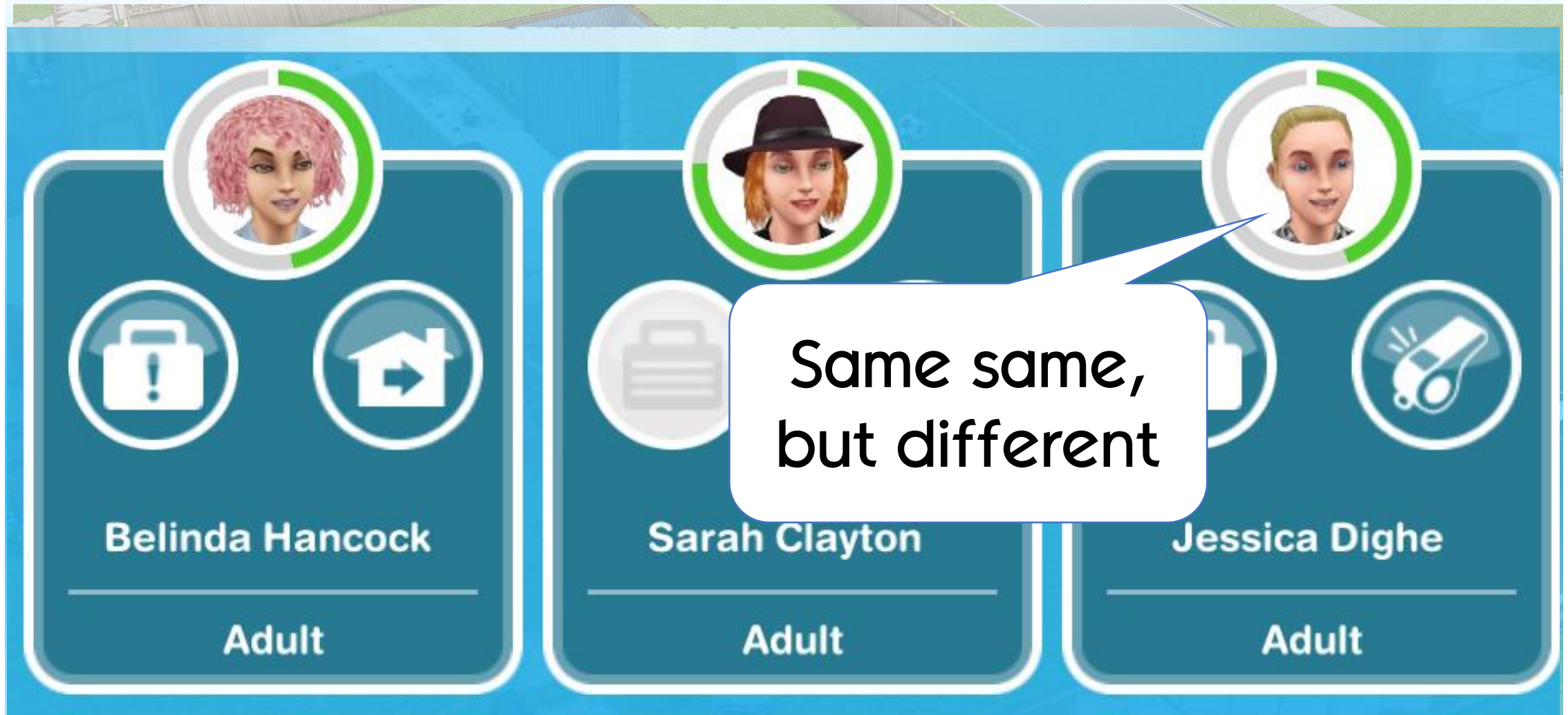


What can we learn?

- What **content** they want
- What would they like to see **again**
- What **features** players want
 - How players feel about the features we released
- **Motivations**



Why Hair? Because Sims have the same face

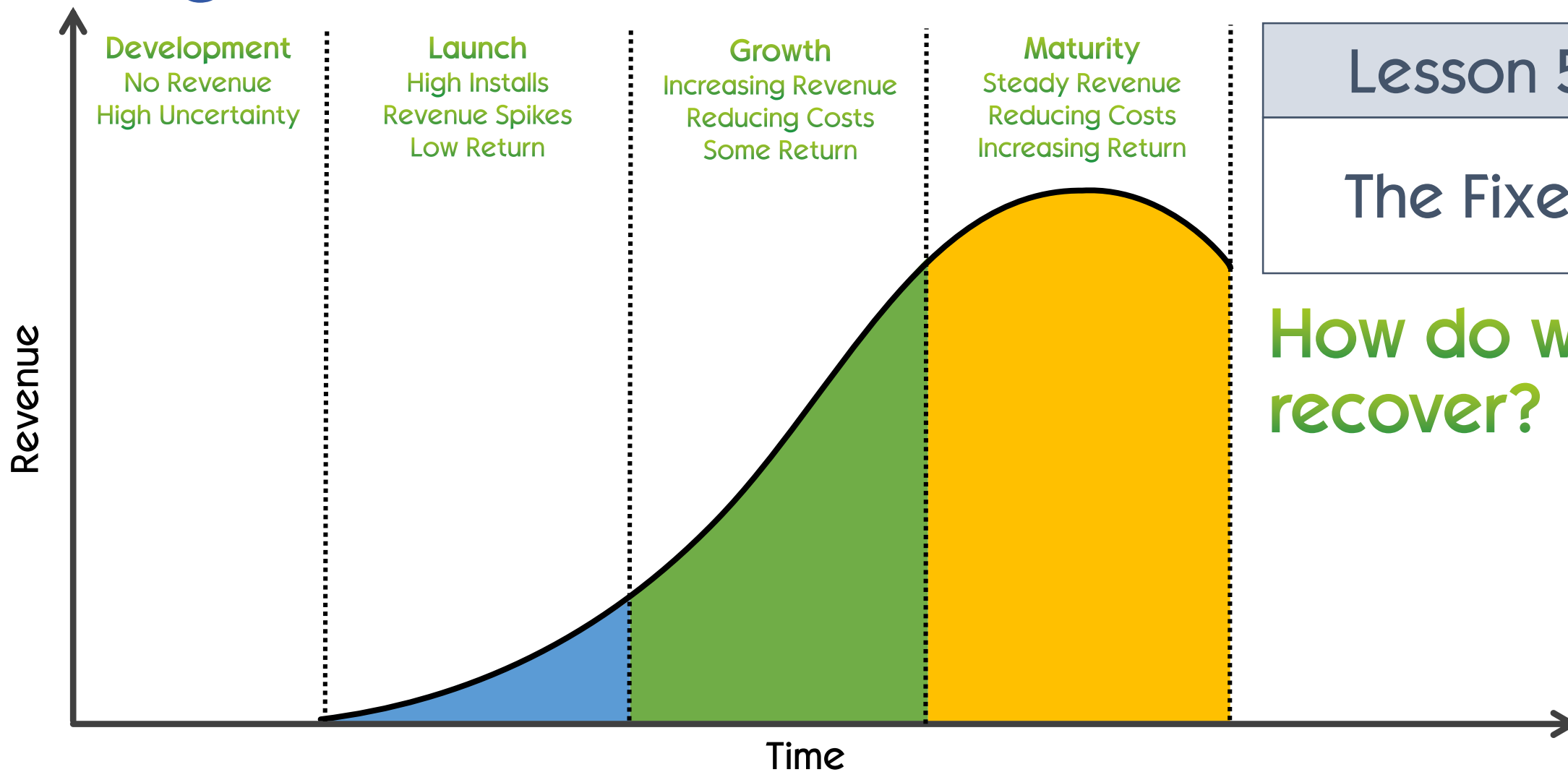


Lesson 4: The Keen Listener

- We aren't our target audience
- Build multiple channels to listen and ask questions
- Guides development and reduces risk
- Relevant all through the product lifecycle



Stage 4: Maturity



Lesson 5

The Fixer

How do we
recover?

Mistakes in Live Service



Categories

- Bugs, bugs, bugs
- Not getting the **content** right
- Not getting the **feature** right

Impacts the following,

- **Stability**
- **Sentiment** on social channels
- **Feature and Event Performance**



How to Deal

Bugs

- Triage process
- Communicate transparently

Other Mistakes

- Deal with them based on impact and return
- Consider risk to Player Sentiment and KPIs
- Easier to add vs remove



Weakening Foundations

Some challenging things to overcome in Maturity

- **Diminishing Returns** – Rewards that were initially highly valuable to players, don't stay that way
- **Expectations** – Players understand the rules and get really good at min/max-ing
- **Anchoring** – Pricing, quantity, and quality of rewards

Have to be careful and **thoughtful** in how we try to solve these problems



Lesson 5 – The Fixer

- 1) It's normal to trip up
- 2) Be prepared to fix bugs with an effective triage process
- 3) Regularly assess problems and consider solutions
- 4) Be both careful and thoughtful with weakening foundations



Live Service is about Launching + Learning



6 week cycles where we learn by
Building, Measuring, and Learning

2 Ways of testing hypotheses

- 1) Launching and measuring after
- 2) AB testing

Set up expectations and hypotheses
in both instances to sharpen
judgement

Lesson 6

The Validated
Learner

How do we
learn?



AB Testing

Test with changes where we're
not certain on impact

Couple ones we've done include,

- Optimising the FTUE Funnel
- Optimising the quest progression
- Optimising where features are unlocked

Be wary of the **cost** of testing
every small change



Storing Lessons as Memories

- Ideally, we would like to build on the lessons from past years
- Major challenge as the product ages
- Team members don't stay on forever
- Team executes better if they understand why and how we got here
- Strategies include Update Telemetry Reviews, documentation, and hand me down lessons



Lesson 6 – The Validated Learner

- 1) Every update is an opportunity to learn
- 2) Set up expectations to improve judgement
- 3) AB testing is a great way to measure impact
- 4) Have a plan to store lessons so new team members can avoid making the same mistakes

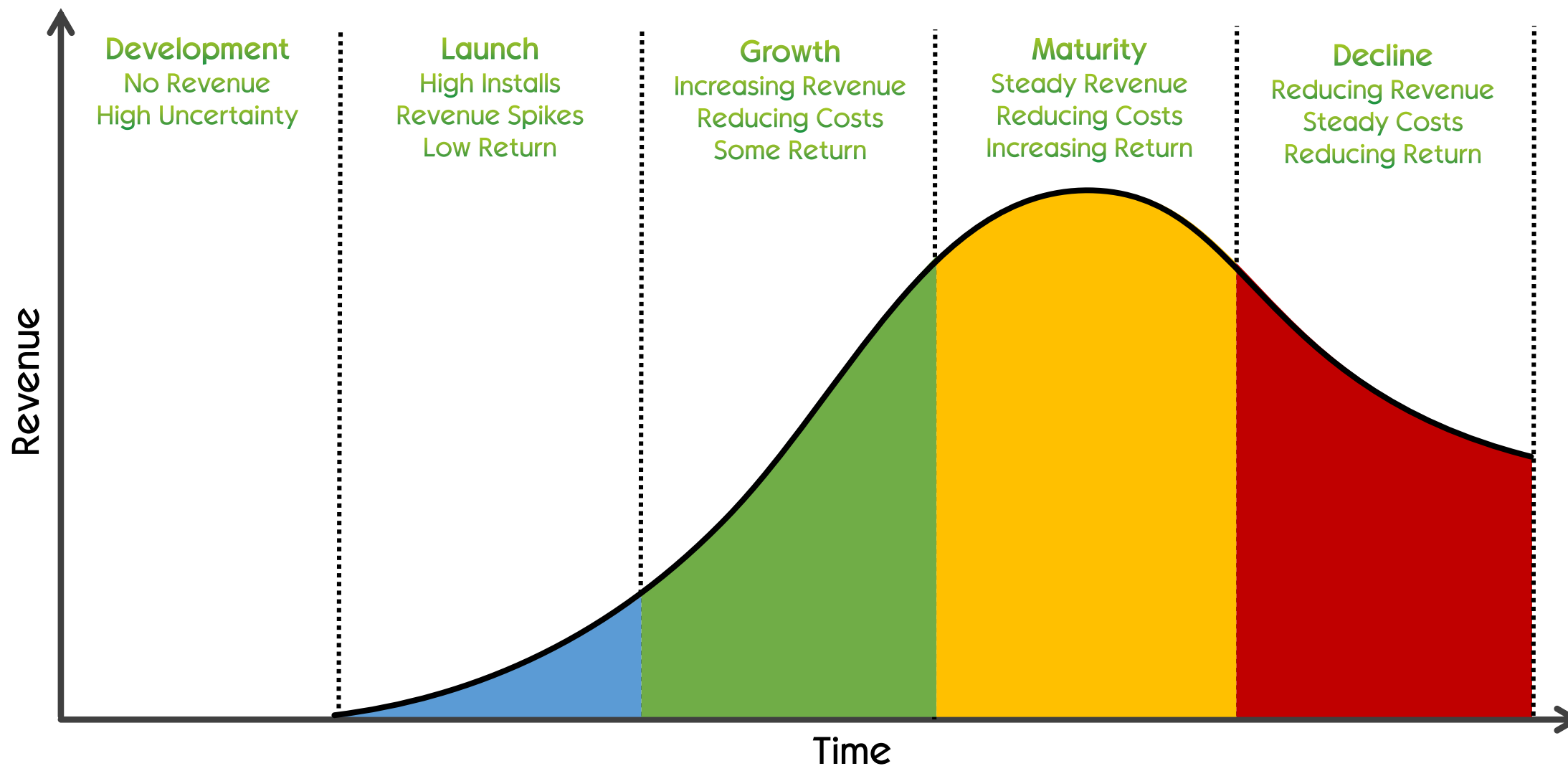


Product Life Cycle - Challenge

- The real challenge with the Product Life Cycle is **not** about how the stages work
- It's the fact that we don't know **how long** each stage lasts



Stage 5: Decline



Strategy Paradox

- Context changes
- Formula reliance is great in growth and maturity phases but...
- Ironically puts us on a path towards both player and dev fatigue
- Challenging to build for a difficult stage when you're in a comfortable one

Lesson 7

The Perpetual Challenger

How can we live forever?

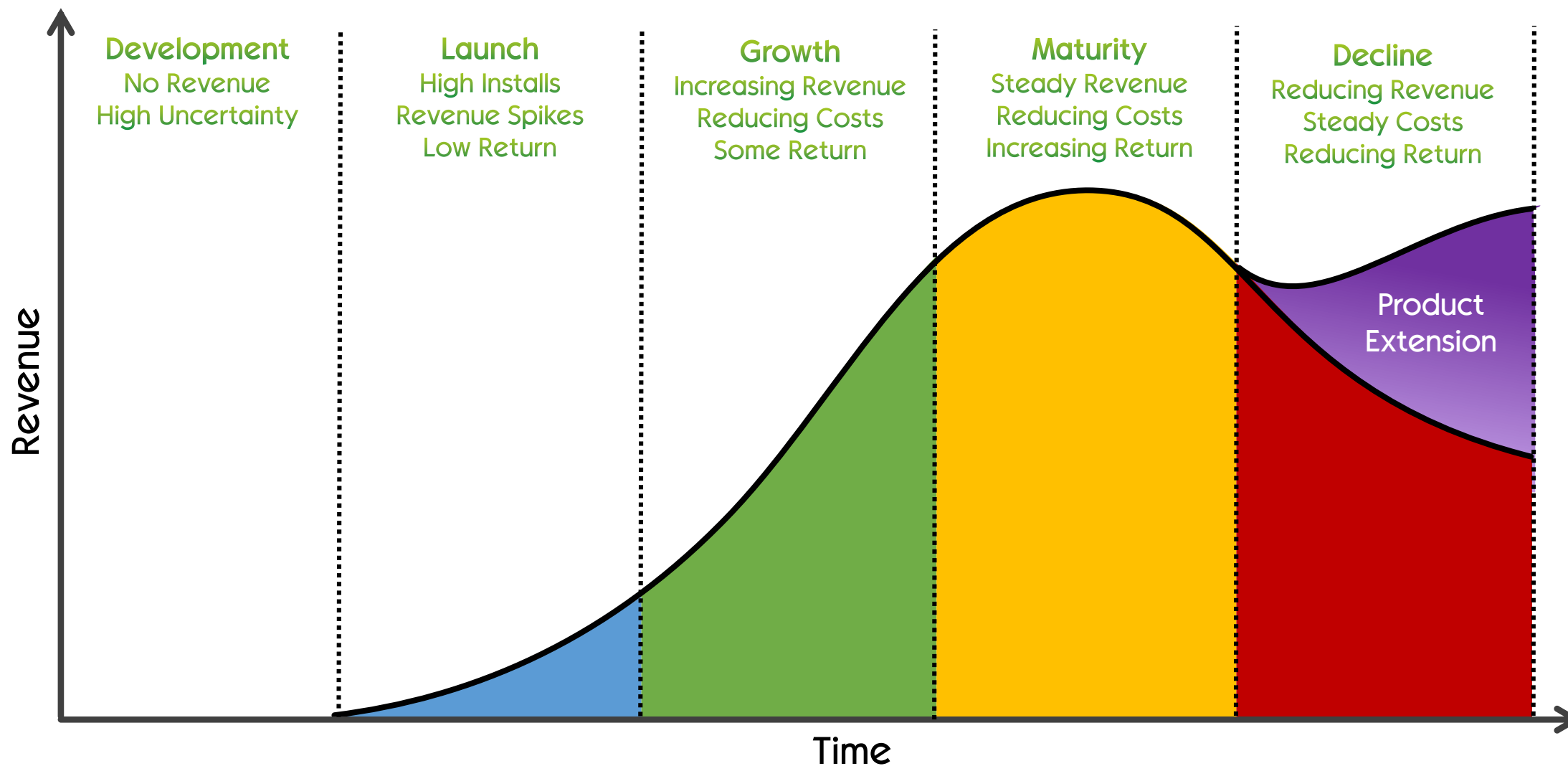


Breaking the Legacy Shackles

- Main Challenges
 - Spending dev time on things that don't work as well over time due to weakening foundations
 - Processes that were optimised for the past
- Have to overcome inertia by
 - Identifying key problems
 - Building innovative solutions



Product Extension



Embrace Reinvention

- Look at a longer horizon (2 to 4 years)
- Align team to stay ambitious and embrace change
- Develop features that satisfy business goals + what players want + fixes problems
- Features introduced over the years
 - Live Event System – Grew LiveOps Return
 - Professions – Grew baseline
 - Pregnancy – Grew baseline
 - VIP System – Added new motivation to spend



Lesson 7: The Perpetual Challenger

- 1) Too easy to **cruise** on success
- 2) Have **ambitious** goals and plans to get there
- 3) Think **long haul** – 2 to 4+ years out
- 4) Develop features that satisfy **business goals + player wants + fixes problems**
- 5) Be prepared to **face legacy** head on



Defying Expectations - 7 Lessons from The Sims FreePlay



- #1 – The Foundation Setter
- #2 – The Puzzle Solver
- #3 – The Growth Investor
- #4 – The Keen Listener
- #5 – The Fixer
- #6 – The Validated Learner
- #7 – The Perpetual Challenger



Final Takeaways

- Product Life Cycle of F2P games
 - Understand the stages and determine which one are you in
 - Be prepared for the different challenges
- Simulation games are content driven
 - Get better with over time
- F2P Live Service is hard but rewarding
 - There is no recipe for success
 - Instead, you get to learn multiple lessons



Be brave,
keep learning,
and go out there to
defy expectations





Thank You