
Game Developers Conference®
February 28 - March 4, 2011
Moscone Center, San Francisco
www.GDConf.com
You ARE the Support, Son!
Supporting your team on the “road to ship”.
By Chris Mielke – Producer, Epic Games
UBM
GDC

About me

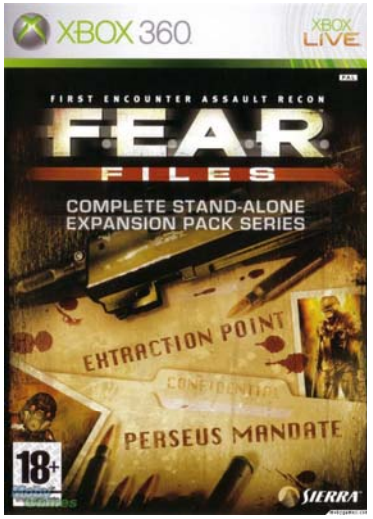
- **Started in the game industry at Day 1 Studios in 2003**
  - MechAssault 2: Lone Wolf – Content Manager

About me

- **Started in the game industry at Day 1 Studios in 2003**
  - MechAssault 2: Lone Wolf – Content Manager
  - F.E.A.R.– Associate Producer

About me

- **Started in the game industry at Day 1 Studios in 2003**
  - MechAssault 2: Lone Wolf – Content Manager
  - F.E.A.R.– Associate Producer
  - F.E.A.R. Files – Associate Producer



About me

- **Started in the game industry at Day 1 Studios in 2003**
  - MechAssault 2: Lone Wolf – Content Manager
  - F.E.A.R.– Associate Producer
  - F.E.A.R. Files – Associate Producer
- **Went to Epic Games in 2007**
  - Gears of War 2 – Art Production Manager

About me

- **Started in the game industry at Day 1 Studios in 2003**
  - MechAssault 2: Lone Wolf – Content Manager
  - F.E.A.R.– Associate Producer
  - F.E.A.R. Files – Associate Producer
- **Went to Epic Games in 2007**
  - Gears of War 2 – Art Production Manager
  - Shadow Complex – Production Manager

About me

- **Started in the game industry at Day 1 Studios in 2003**
  - MechAssault 2: Lone Wolf – Content Manager
  - F.E.A.R.– Associate Producer
  - F.E.A.R. Files – Associate Producer
- **Went to Epic Games in 2007**
  - Gears of War 2 – Art Production Manager
  - Shadow Complex – Production Manager
  - Gears of War 3 - Producer

Introduction

The ultimate Producer interview question:

-How do you close down a project?

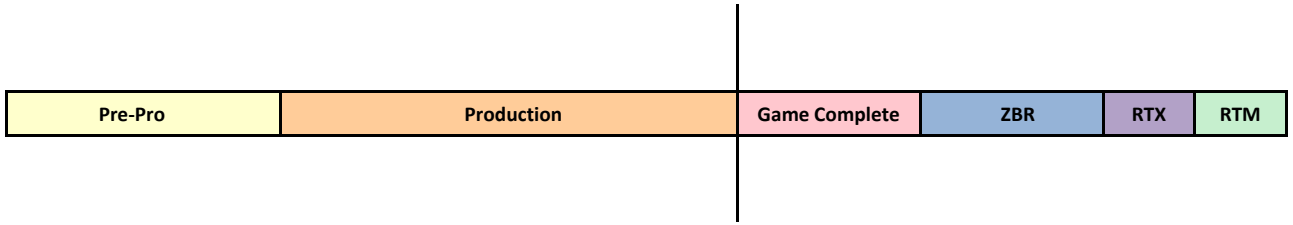
• It may seem basic, but even the easiest things are often hard to do correctly when under deadlines

• Practice makes perfect – a new project means another chance to implement more strategy

• Everything takes time – don't get discouraged, some tips and tricks

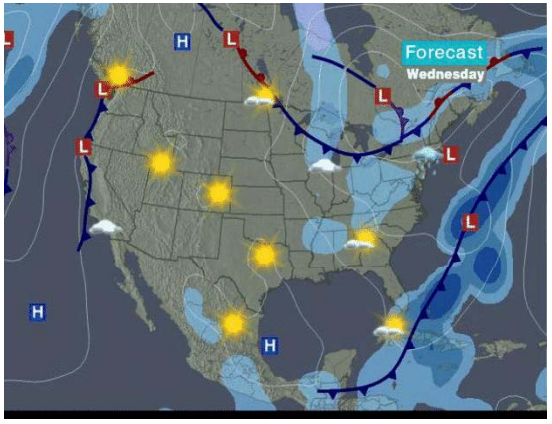
How producers support their team on the “road to ship”

What exactly does the “road to ship” encompass?



How producers support their team on the "road to ship" (cont'd)

- **Help increase predictability and efficiency through scheduling and risk mitigation**
  - Act as "weathermen" on the project – forecasting chance of success or change.



How producers support their team on the "road to ship" (cont'd)

- Facilitators of the shipping process/methodology
  - -Show the team how to get to the end product

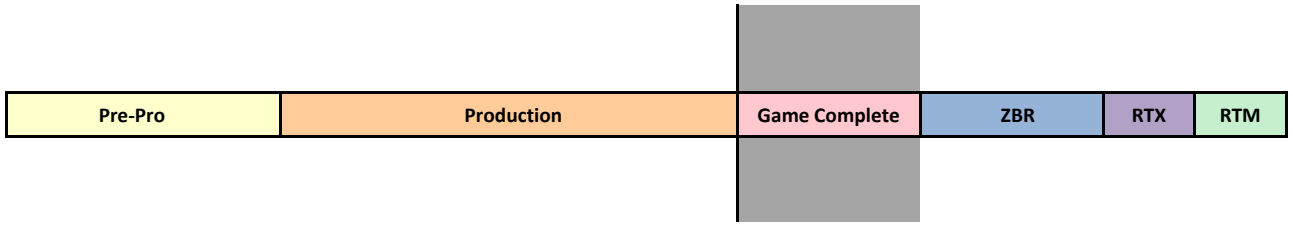
- Ultimately it's the job of the producer to do whatever is necessary to ship
  - Includes everything from resource/equipment requests to keeping stakeholders up to date and confident in the project

How producers support their team on the "road to ship" (cont'd)

- Support their team with the "softer" side of production
  - Late night coffee/food run
  - Swag for appreciation
  - Make sure people are fed/rested
- Keep the team work/home life balance

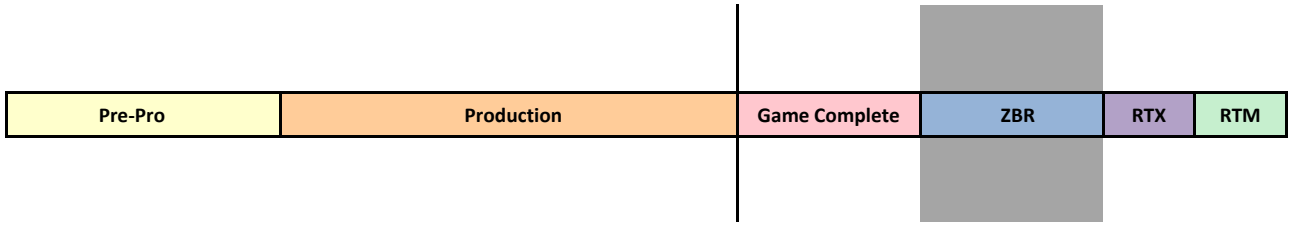
General roadmap of shipping

- **Code Complete (Alpha)**
  - Final pass of all feature work completed
  - Only feature polish and optimization work remaining
- **Content Complete (Beta)**
  - All levels content complete and polished
  - All cinematics complete (and possibly music foley)



General roadmap of shipping
(cont'd)

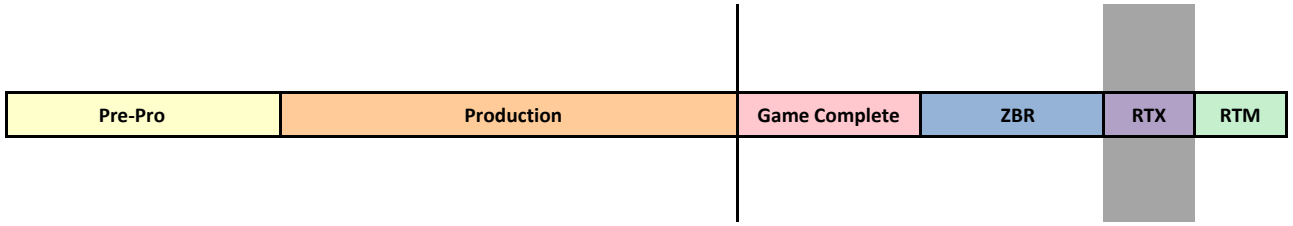
• **ZBR** (Need to hit a bug count of zero every night and have no new bugs logged for 24 hours)
  - The "bounce"
  - RC0 (RC LOL) - Priority 0 bugs become do not leave desk
  - RC0 is to practice shipping, it's an exercise in how to get your team to ship



General roadmap of shipping

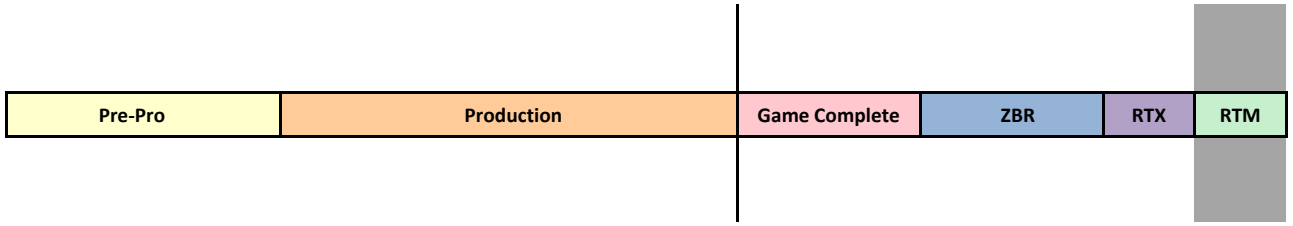
(cont'd)

- **RTX - Release to Certification**
  - Xbox (TCRs), PS3 (TRCs), PC (publisher driven criteria), etc.
  - Package up the build, submit documentation (ESRB also)
  - Four weeks allotted – allows for one "bounce" from Cert
  - Multiple regions



General roadmap of shipping
(cont'd)

- **RTM - Release to Manufacturer**
  - Going "gold"
  - 3-5 week process depending on number of regions
  - Party!



Advice and methods for supporting your team during the “endgame”

- Playing for keeps
- A spoonful of sugar helps the bug count go down
- Charting the course
- Letting your leaders lead
- One must talk little and listen much

Playing for keeps

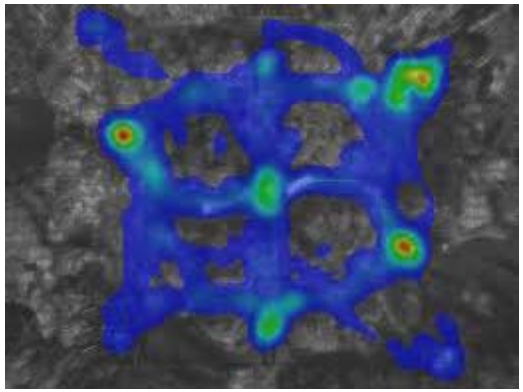
- **Playing and testing the game on a consistent basis**
  - One way to increase the quantity and quality of the bugs is go outside of the test team
  - Todd Howard at Bethesda said at D.I.C.E. in 2009: *"Great games are played not made."*
  - Playtest when you get value out of testing

Playing for keeps
(cont'd)

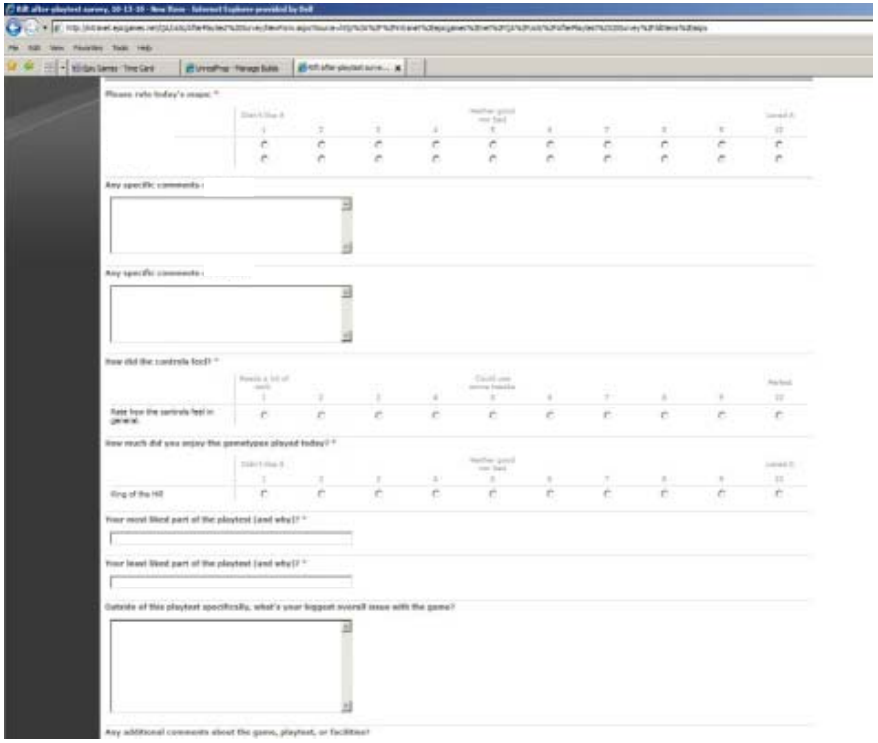
- **Playing and testing the game on a consistent basis** (cont'd)
  - Starts before the "Road to Ship" but continues through it
- Consistent daily play tests for multiplayer
  - Tester has notebook computer and takes down notes/comments in real time
  - People see their work in context in the game – so a bug may surface on usage

Playing for keeps
(cont'd)

- Reporting the results of play
  - Tester has notebook computer and takes down notes/comments in real time in the lab (feedback sent to all participants to verify with heat maps)
  - Wrap up meeting after the play test where everyone can voice their opinion
  - Survey generated for play test participants to return (people may not be comfortable voicing opinions in a group)



Playing for keeps
(cont'd)



Playing for keeps
(cont'd)

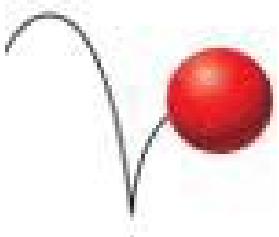
- **Testing the product is not just a job for QA and producers should help find ways to get the team involved in testing**
  - Project leads do play tests 2 times a week over lunch
  - Different modes of the game/coop through different levels
  - Evening play tests after dinner
  - Finding issues/maintaining team morale

Playing for keeps
(cont'd)

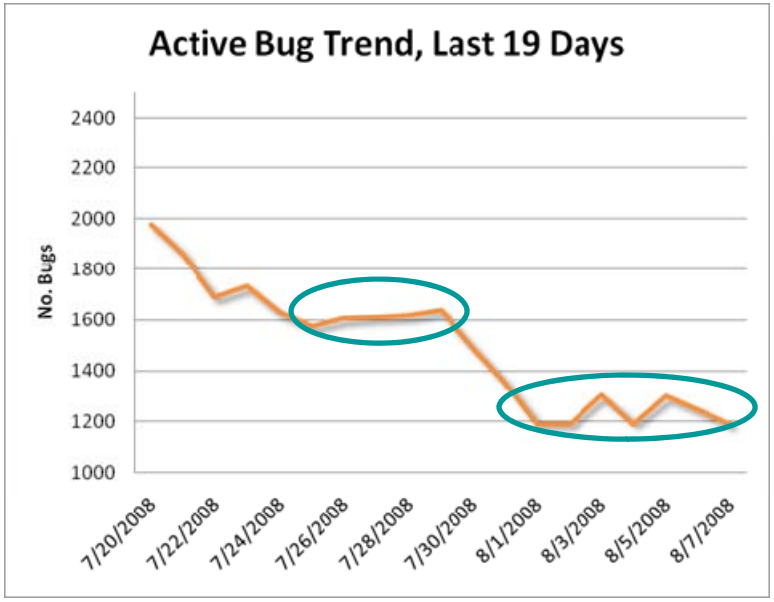
- Casting the playtest net wider
  - Friday evening play tests with food and mixed drinks ☺
  - Gets other disciplines from the office involved (finance, admin)
  - Playing the game non-stop during ZBR/RTX for team members not on "team-ship"
  - Teams doing each path (split screen, coop, solo, MP (all modes))
  - Friends and family weekends
  - Take home build over Xmas break

Charting the course
(cont'd)

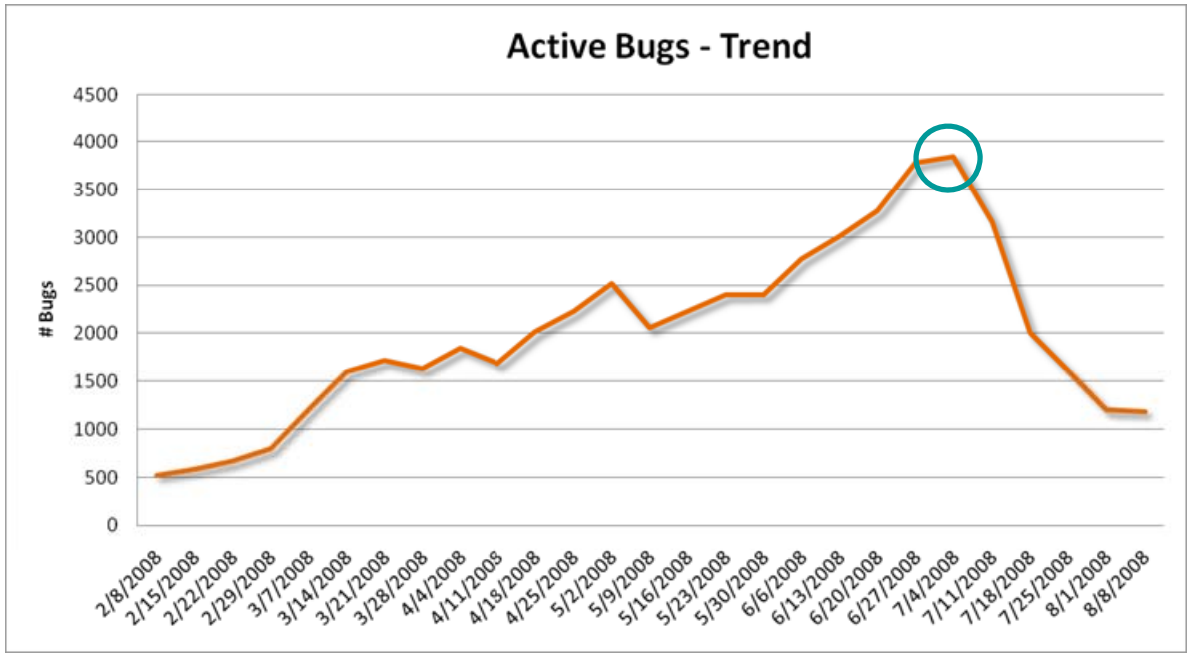
- **Create / maintain a bug slope**
- The accuracy of your bug database
  - Important in terms of triaging and making product decisions
  - Sets the pace of how many bugs need to get fixed
  - Helps estimate at the current rate of find/fix when you will get to ZBR
- The bug glide slope indicates how hard you may bounce after reaching ZBR



Charting the course
(cont'd)



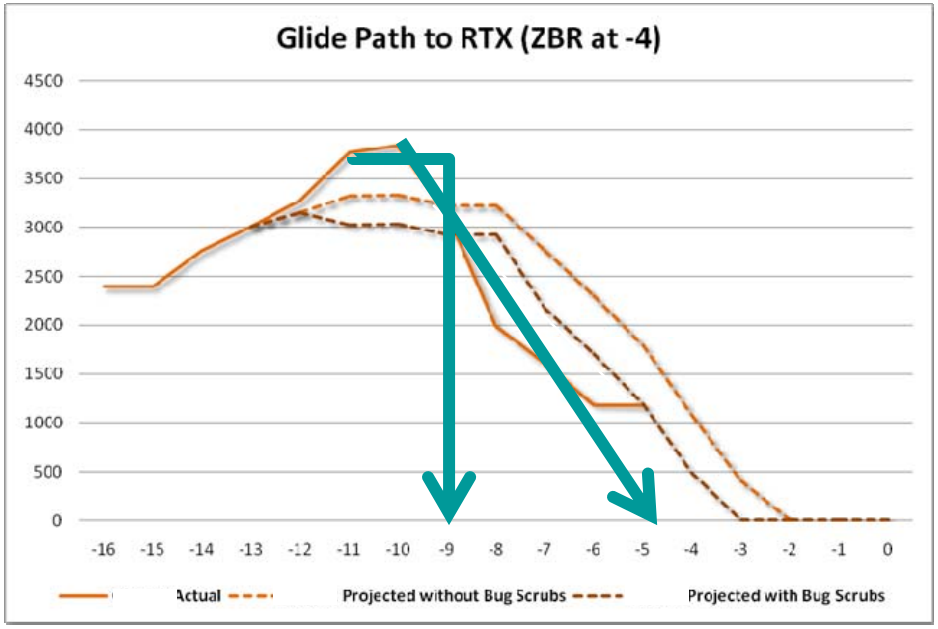
Charting the course
(cont'd)



Charting the course
(cont'd)

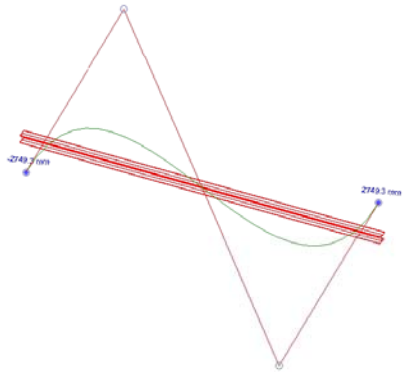
- **This "peak" shows the health of your project**
  - It's where your fix rate overtakes your find rate
  - The later this occurs in the project the longer it will take to get to ZBR
  - Watch your regression rate if it's 10% or less failure rate it's good, 50% or more means that your team may be tired

Charting the course
(cont'd)



Charting the course
(cont'd)

- **Glide slopes should be realistic**
  - Some producers try to draw a glide slope and just do variances around it
  - A typical glide slope isn't smooth since your devs don't fix bugs at constant rates

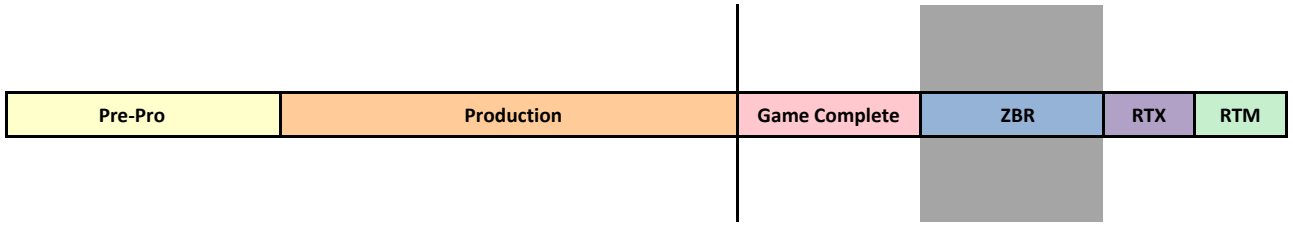


Charting the course
(cont'd)

- The dangers of bug regression
  - Just because a bug is marked as fixed, it's not off the radar and failed verifications is one of the main reasons for the bounce post ZBR

A spoonful of sugar helps the bug count go down

- **The bug database and the competitive spirit of teamwork coupled with a little "sugar" help it along**
  - Bug find/fix “leaderboards” with nightly updates
  - Co-workers helping others out to help them get ahead
- A mature development team balances fixing lots of low priority bugs over fixing high priority bugs



A spoonful of sugar…
(cont'd)

- **Prioritize your database (Every bug gets a priority and a severity)**
  - Bug scrubs are invaluable in avoiding wasted time and increases the team's efficiency
  - Priority is subjective
  - Severity is an objective measure (supplied by QA)

A spoonful of sugar…
(cont'd)

- **Levels of priority - Priority 0-5**
  - **Priority 0 -** Do Not Leave Desk - Needs to be fixed immediately - this bug is usually a showstopper or build play through blocker (better eat lunch at your desk)
  - **Priority 1 -** Immediate - Tasks or major functionality that needs to be completed/fixed today (go out for lunch)
  - **Priority 2 –** High Priority - This is a bug that's blocking some functionality, but is not major. It needs to be fixed for an upcoming milestone, but not immediately.

A spoonful of sugar…

(cont'd)

- **Levels of priority - Priority 0-5**
  - **Priority 3 –** Normal Priority (default) - Possible items that could slip to the next milestone, work on as time permits. Not something to be tested immediately
  - **Priority 4 –** Low Priority - Nice to have bugs to fix, usually reserved for polish items
  - **Priority 5 –** Lowest Priority - Low on the list for being fixed, usually cosmetic, work on only if you have time, and can be punted

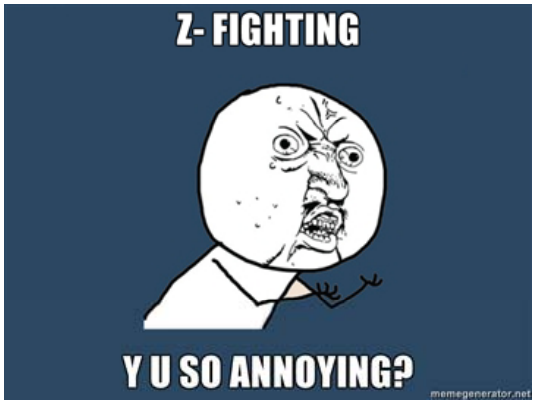
A spoonful of sugar…

(cont'd)

- **Levels of severity - Severity 1-5**
  - **Severity 1 –** A game progression or test blocking bug (i.e.; a crash that doesn't let you complete a level or map)
  - **Severity 2 –** A crashing or major functionality that's broken but DOES NOT stop you from testing or progressing through the game or has a workaround.
  - **Severity 3 –** A minor function that is broken or doesn't work but has minimal impact on the user.

A spoonful of sugar…
(cont'd)

- **Levels of severity - Severity 1-5**
  - **Severity 4 –** A cosmetic or annoying issue that can be avoided and has very little impact on the user.
  - **Severity 5 –** Task

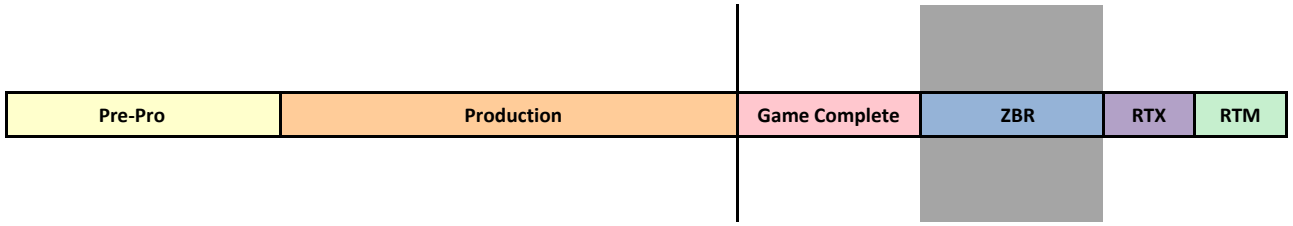


A spoonful of sugar…
(cont'd)

- Balancing Priority and Severity
  - Using these 2 establishes a workflow for the developer
  - Also deals with rare cases – say a bug is Pri 5, Sev 1 (the occasional dreaded "blue moon" crasher)

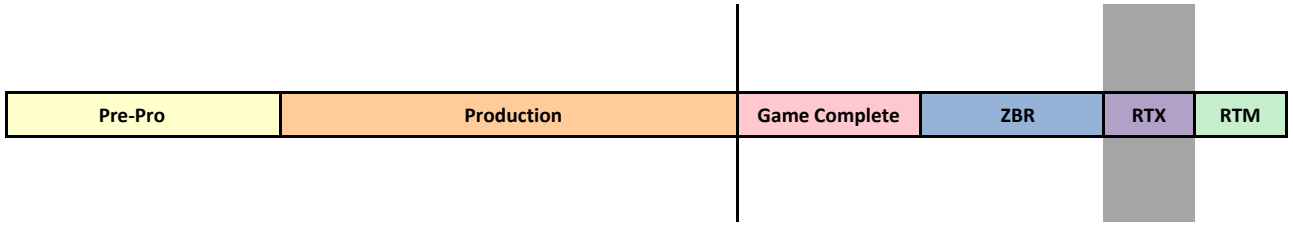
A spoonful of sugar…
(cont'd)

- Raising of the bug bar
  - Ideally very 2 weeks a priority level drops off
  - So your ZBR runs about 10 weeks to account for all the different priority of bugs



Letting your leaders lead

- **Forming of Team Ship**
  - Stripping away of the dev team
  - Numbers are reduced going into ZBR and into RTX



Letting your leaders lead
(cont'd)

- Reasoning behind Team Ship
  - Fewer hands means fewer chances of shaking of the "Jell-O"
  - Every change has the potential to create even more unwanted change
  - Or every fix is risking more bugs

Letting your leaders lead
(cont'd)

- Team leaders take responsibility for shipping the product
  - Encourages teamwork to get the job done since there are fewer people touching the product
- Producers help determine when the product is finished with consensus from the team
  - Going through the release candidate process to get to that very last bug

Letting your leaders lead
(cont'd)

• The ship cycle at Epic is an exciting time that leads to an awesome product
  - High energy, people rise to the occasion
  - "Leave it all on the field"

One must talk little and listen much

- **Creating the "Triage Committee" and it's purpose**
  - Formed to facilitate the shipping process so the committee will go through the bug database and punt or reprioritize
  - Made up of core leads/producers
  - Every bug goes to triage for review
  - Can go from monthly, weekly, daily or more
  - Producer is more of a facilitator and listens to concerns

Resources

- Game Production Handbook (Heather M. Chandler)
- The Game Producer's Handbook (Dan Irish)
- The Mythical Man-Month: Essays on Software Engineering (Frederick Brooks)
- Project Risk Management (Bruce Barkley)

Questions?


EPIC
GAMES
We Are Hiring
www.EpicGames.com/jobs

Thanks for listening!

Please fill out your survey form.