



The Game Development Process

Postmortems

“Those who do not learn from history are doomed to repeat it.”
- *George Santayana*



Introduction

- When starting new project reflect very critically on past projects (the Postmortem)
 - What went right
 - What went wrong and could have been done better
- Come up with a plan of attack for the new project
- “Companies that do not conduct some form of postmortem are doomed to repeat the same mistakes.”
 - Team Management, Concept, Development, Business Aspects





Sources of Postmortems

- Game Developer Magazine
 - Best articles, often
- Gamasutra
 - www.gamasutra.com



Topics to Critique (1 of 2)

- Team Management
 - Gather individual's post mortem's
 - Review anonymously (without recrimination)
 - Look for patterns, repeats
- Concept
 - Surely sound or why building?
 - But many lame ideas ... (just look at the Bargain bin)
 - Climate
 - Is the world ready?
 - May change in two years. (Blink and the weather changes. Snooze and you get an ice age.)
 - Accessibility
 - Could you get the point to them?
 - Includes marketing, and player-gameplay balance

Based on Chapter 23 of *Game Architecture and Design*, by Rollings and Morris





Topics to Critique (2 of 2)

- Development
 - Usually more here than in earlier phases
 - Longer, more intense, more complex, more people → More to go wrong
 - Software planning
 - Mistakes here, 200 fold more expensive to fix later
 - *Feature creep* often to blame "Wouldn't it be cool if ..."
 - Coding
 - Most errors here. Actually typing code small, tho
 - Testing
 - Done early enough?
 - Testing all configurations on PCs tough
- Business Aspects (financially managed?)

Based on Chapter 23 of *Game Architecture and Design*, by Rollings and Morris



Outline

- Introduction
- Summary of Postmortems (next)
- Common Patterns
- Notably Absent
- What it all Means



Summary of Postmortems

- Attempt to extract common patterns in recent (2002 - 2004) postmortems
 - Not comprehensive, just patterns "Wrong"
- More comprehensive study of earlier postmortems
 - [Gamasutra.com Postmortems](#) by Simon Larsen
 - Up to September 2002
- This article took postmortems from from Game Developer Magazine
 - October 2002 to April 2004
 - Excluded subsets of the project (like tools, animation systems, sound systems, etc)
- Selected 13 (see next page)
 - *Prince of Persia, Neverwinter Nights, Gotham Racing ...*

Based on *Postmortems: Looking Back, Looking Ahead*, by Noel Llopis
<http://www.gamesfromwithin.com/articles/0404/000019.html>



Selection of 13 Postmortems

- "*Aggressive Inline*" (Z-Axis)
- "*Neverwinter Nights*" (BioWare)
- "*No One Lives Forever 2*" (Monolith)
- "*Battle Engine Aquilid*" (Lost Toys)
- "*Ratchet & Clank*" (Insomniac Games)
- "*Rise of Nations*" (Big Huge Games)
- "*Amplitude*" (Harmonix)
- "*TRON 2.0*" (Monolith)
- "*Homeworld 2*" (Relic Entertainment)
- "*Jak II*" (Naughty Dog)
- "*Secret Weapons over Normandy*" (Totally Games)
- "*Project Gotham Racing 2*" (Bizarre Creations)
- "*Prince of Persia: The Sands of Time*" (Ubisoft)

Based on *Postmortems: Looking Back, Looking Ahead*, by Noel Llopis
<http://www.gamesfromwithin.com/articles/0404/000019.html>





Warnings

- Not big enough sample
- Self-selected group of projects
 - Choose to write a public postmortem
 - All managed to ship a game (relatively successful)
- What authors felt could write in public
 - Extremely cautious about what to say and how to say it
 - Some important problems not mentioned
 - Ex: "our publisher had no clue what it was doing,"
 - Ex: "my boss is completely incompetent but we shipped the game in spite of him."
- Still, incomplete data better than no data

Based on *Postmortems: Looking Back, Looking Ahead*, by Noel Llopis
<http://www.gamesfromwithin.com/articles/0404/000019.html>



Outline

- Introduction
- Summary of Postmortems
- Common Patterns (next)
- Notably Absent
- What it all Means





Lack of Resources/Time

- All have very hard deadlines
- Commit to shipping by a particular date
 - Christmas or Thanksgiving weekend favorites
- Number one problem listed in all postmortems was some features not started until too late
- About every aspect of game: technology, tools, design, art, etc.
- Some cases, features fell short
- Most cases, severely affected other parts
- Flip side: not enough resources
- Seems like when managing project, three variables to play with: *time*, *resources*, and *features*
 - Pick any two
 - Pick all three and deliver in none of them instead

Based on *Postmortems: Looking Back, Looking Ahead*, by Noel Llopis
<http://www.gamesfromwithin.com/articles/0404/000019.html>



Lack of Approval Process

- Second most common, lack of internal approval process
- Examples:
 - sub-par content in final game
 - technology that appeared to be finished but wasn't
 - feature creep that ruined the schedule
 - overly ambitious designs not really feasible
- Closely tied was unnecessary rework
 - Caused significant delays
- Difference between useless rework and an iterative approach

Based on *Postmortems: Looking Back, Looking Ahead*, by Noel Llopis
<http://www.gamesfromwithin.com/articles/0404/000019.html>



Inadequate Content Pipeline

- A surprise topic (at least to the author)
- Examples:
 - Not being able to deal with so many assets
 - Iteration time being too long for content creators
 - Not having fully automated system to CD burn
- Will become more important in the near future
 - Will pay for companies to explicitly define and streamline content pipeline

Based on *Postmortems: Looking Back, Looking Ahead*, by Noel Llopis
<http://www.gamesfromwithin.com/articles/0404/000019.html>



Large Team Woes

- Trouble coordinating the efforts
 - Results in unnecessary rework
- Interestingly only one mention of communication
 - Might expect to be more common
 - Maybe taboo area?
- Sizes of teams where coordination an issue:
 - *Prince of Persia*: 65 (peak, excluding testers)
 - *Project Gotham Racing 2*: 40 (core team), 102 (peak, including testers)
 - *Jak 2*: 48 (full time)
 - *Neverwinter Nights*: 75 (peak), 40 QA, 5 sound, 20 translators.
- Teams count differently, but most others smaller
- If large way of future, better get used to it

Based on *Postmortems: Looking Back, Looking Ahead*, by Noel Llopis
<http://www.gamesfromwithin.com/articles/0404/000019.html>





Crunch Time

- Surprisingly, only a few clearly identified crunch time and employee burnout as problem
- Most acknowledged crunch time
- Scary part ... in the "what went right" section!
 - "how dedicated the team was"
 - "how macho they all were" that pulled it through
- Industry should grow out of basement coder mentality, we'll continue having the same problems
 - See IGDA Quality of Life (<http://www.igda.org/qol/>)
 - From EA fallout, forum at GDC'05, too

Based on *Postmortems: Looking Back, Looking Ahead*, by Noel Llopis
<http://www.gamesfromwithin.com/articles/0404/000019.html>



Other Problems

- Localization (porting to other countries)
 - more complicated in dialogue/movie-heavy games
- QA problems (either bad testing or not enough)
- Side-tracked by demos
- No clear objective where game heading
- Too flexible engine as a negative point
 - Great feature on paper, but usually means not great in any specific way





Outline

- Introduction
- Summary of Postmortems
- Common Patterns
- Notably Absent (next)
- What it all Means



Technology/Performance

- Surprising, considering the amount of time, effort, and trouble that programmers go through
 - Only mention were when some multiplatform development
- Too proud to say?
- Or too much emphasis on performance at start?
- Maybe should concentrate on making more robust and playable and accept a slightly lower particle count





Team Problems

- Probably taboo area that public postmortems can't touch
- Only one mentioned, then only to hiring process
- Maybe everybody else had perfect team where everybody got along great and worked together in perfect harmony.
 - Yeah, right



Quality and Bugs

- No complaints about quality of code produced
- Maybe one of things game industry takes for granted?
- Good code quality leads to few bugs, little (if any) overtime, and much better overall game
 - Features tested and experimented easily until release
- Automation can help





Publisher Interference

- No unreasonable misguided demands by publishers
- Derail production schedules
 - Result in the crazy hours
 - Nasty crunch times
- The "run faster" factor
- Can result in changing requirements, feature creep, and trying to copy latest chart-topper
- Falls in category of taboo subjects
 - Developers hard enough securing funding and a publisher for their games
 - Not about to bite the hand that feeds them



What Does It All Mean?

- Step back. One problem all facing: complexity
- Used to be technical issues. Now past that.
- Game industry is "whale that has difficulty breathing due to its own weight" but don't fully admit it
- Increasingly, games require:
 - huge team
 - produces a huge amount of assets
 - needs to communicate and coordinate efforts
 - create a great game in the end
 - Oh yeah, did I mention ship in time for Christmas?

