CS354P

DR SARAH ABRAHAM

AGILE OVERVIEW

TRIPLE A GAME DEVELOPMENT TIMELINES

- Notoriously difficult to predict or control
- Release dates often determined by publisher based on other game/movie releases and seasons
- Fixed timelines means pivots (however necessary) are risky and expensive
- Note: indie and mid-sized studios have different constraints, as does mobile, but similar problems arise

AGILE DEVELOPMENT

- Catch-all term for iterative software development
- Testing tightly coupled with design and execution
- Allows for a project to deviate from its original conception as new information shapes its direction
- Needs to balance long-term goals with short-term accomplishments

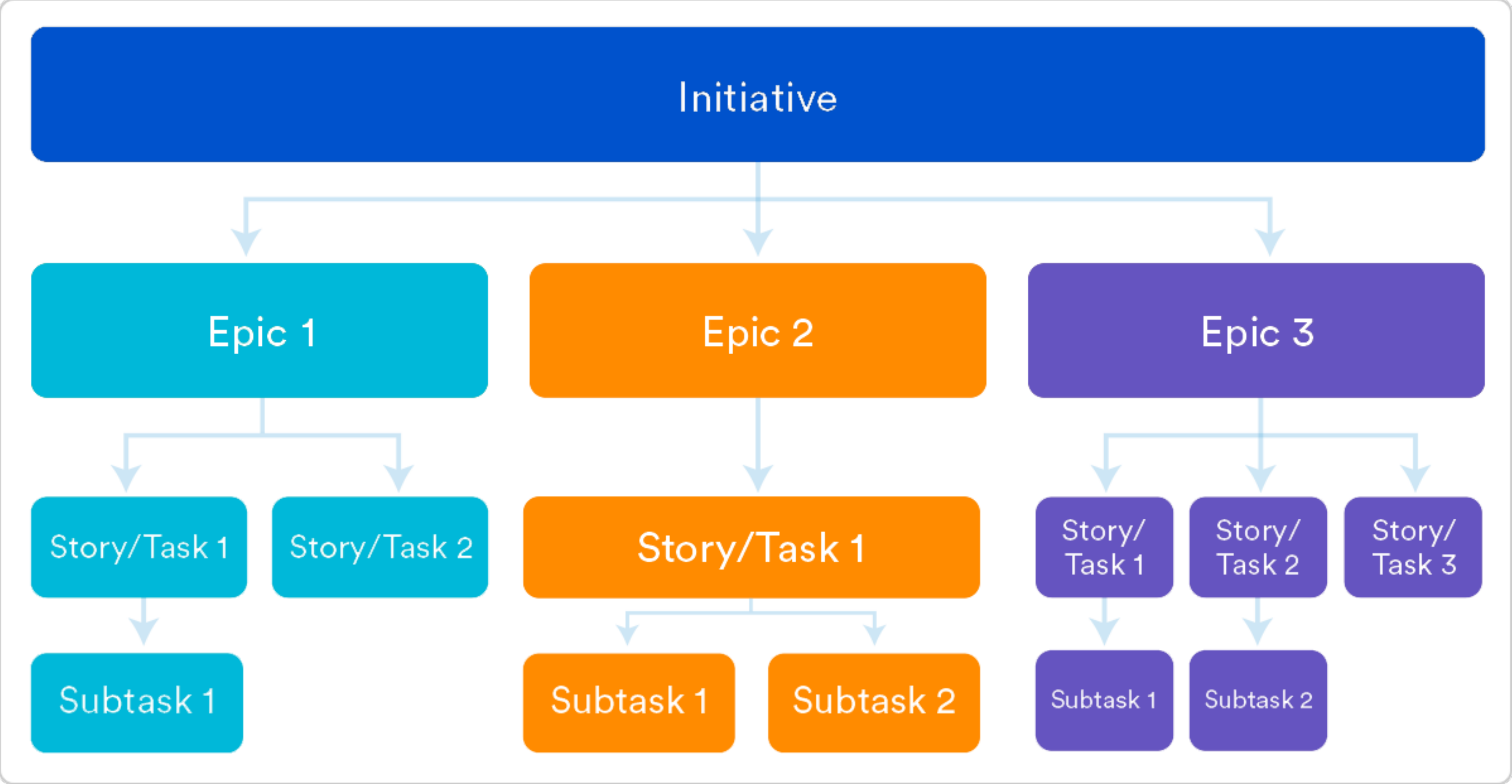
AGILE VERSUS WHAT?

- Agile as a concept mostly makes sense in the context of other methodologies such as Waterfall
- Waterfall requires planning out each stage fully before executing on it
- Works well in highly structured design cases
  - Traditional architecture
  - Hardware architecture
  - etc..

AGILE TIMEFRAMES

- Epics
  - Large body of work containing “user stories”
- Stories
  - Functionality must be accomplished to provide a user experience
- Tasks
  - Units of work that make up stories
- Sprints
  - Period of time teams have to complete tasks and stories

TIMELINE OVERVIEW



SPRINTS

- A short (usually two to four week) period of time to complete a set amount of work
- Why two to four weeks?
  - How does this affect the task size versus iterative nature of development?

SCRUM

- Popular methodology for implementing Agile principles
- Basic idea of having frequent check-ins on project's current direction and progress
  - Short daily check-ins (stand-up)
  - Check-ins at the start of a sprint (sprint planning)
  - Check-ins at the end of a sprint (sprint review)
- A number of formal "ceremonies" and roles within this methodology but not strictly necessary to accomplish its purpose

STAND-UP

- Daily check-in at the beginning of the work day
  - Ideally brief (less than 15 minutes total)
- Each team member lists accomplishments, obstacles, and progress toward the sprint deadline
  - What they did yesterday
  - What they're going to do today
  - Any known blocks
- Ensures everyone knows what the rest of the team is doing and provides clear daily objectives
- Reduces number of meetings…in a perfect world!

STAND-UP ETIQUETTE

- Try to remain standing if you don't have a condition that makes standing difficult
  - The idea is to keep it short, and standing incentivizes short meetings
- This is **not** a time for discussions or problem-solving
- Anything that needs to be discussed should happen **after** the stand-up
- Stand-up should happen within relatively small teams
  - Too large of groups take too long
  - Too large of groups means most of the information is irrelevant for the individual developers

SPRINT PLANNING AND REVIEW

- Planning meeting organizes what needs to be accomplished during a sprint
  - Prioritization of tasks
  - Estimates of work time
- Review meeting shows what has been accomplished during the sprint
  - Time for feedback and critique but not the final approval process

TASKS AND TICKETS

- Tasks usually assigned at the start of a sprint
- Tickets encapsulate the features and bugs you are expected to work on (more on this next time)
- Good time estimates on tickets is essential
  - How to know how long a feature/bug fix will take?

WORKING AS A TEAM

- Needs and desires of programmers, artists, and designers are often in direct opposition!
  - Designers want more features
  - Artists want more texture memory
  - Programmers want the game to run fast
- Compromise required
- Game Design Document (and Technical Design Document) can guide this compromise process
- Clear scoping prevents miscommunications and wasted effort

WHAT IS SCOPE?

- All-encompassing assessment of project effort
- Game features require time and money
  - Reducing time or money reduces scope of game features
  - Reducing scope of game features can reduce need for time or money

SCOPING FEATURES

- Creating new engines, tools, or features increases scope more than using existing engines, tools, or features
- Being conservative in terms of features may reduce time and effort but potentially lowers game quality
- Be deliberate when choosing where to spend effort
- Pick the right people (and the right tools) for the right job

SCOPING MONEY AND TIME

- Allowing more time for a project can reduce the needed money
  - Often how indie developers work without quitting their "day job"
- Having more money for a project can reduce the amount of time a game takes
  - Fast ramp-up time or hiring on experts can make a project go more smoothly

THE MYTHICAL MAN-MONTH

- 1975 book on software productivity by Fred Brooks
- Addresses common productivity misconception that adding more people can reduce development time
  - “The bearing of a child takes nine months, no matter how many women are assigned.”
  - Game studios often guilty of this fallacy!
  - When game falls behind schedule, large number of contractors brought on to “speed up” development
- The moral is money doesn’t inherently reduce time
  - Work smarter – not harder

OVER-SCOPING AND FEATURE CREEP

- Over-scoping is the underestimation of time and effort required for a project
  - Can be mitigated with better planning and consulting experts with direct experience in that domain
- Feature creep is changes/additions of features well into the game development cycle
  - Earlier wasted effort can slow development
  - Can be mitigated by “locking in” ideas earlier
  - Note: changing direction may not lead to a better game and locking in early may not work out either!

WHAT IS CRUNCH?

- Overtime work for an extended period
- Often unpaid if you’re salaried
- Common in game development as well as other software industries

MITIGATING CRUNCH

- Crunch will probably happen at most studios but 1) it isn't an inevitability and 2) there is such a thing as reasonable crunch:
    - No more than 2-4 weeks
    - No more than 60 hours/week
    - Meets a specific goal or major deadline
    - Normal hours afterward for at least twice the duration of crunch

PRICE OF CRUNCH

- Humans unable to maintain creative work for that long!
  - Reduces cognitive function
  - Destructive toward worker’s mental and physical health
- Accomplishes less per 60+ hour week than during a normal 40 hour week!
- Leads to massive company attrition
- https://www.ign.com/articles/naughty-dog-leaders-discuss-crunch-tlou2

CRUNCH IS NOT REQUIRED TO MAKE A GOOD GAME!

- https://kotaku.com/the-secret-to-the-success-of-bastion-pyre-and-hades-1838082618
- https://screenrant.com/video-game-companies-without-crunch-time-forced-hours/

HOW TO AVOID CRUNCH CULTURE?

- Not all game studios crunch!
  - Life at a studio varies widely based on the company's values
- Look for jobs that are known for promoting work-life balance
- Good management leads to good balance
- It's easy to know if a studio crunches because everyone knows
  - Ask at game networking events
  - Check on Glassdoor
  - Ask during the interview (if a place supports crunch they usually won't try to hide it)

WHAT DO YOU LOOK FOR IN A COMPANY?

- Something to consider whether or not you want to work in games
- Many considerations beyond salary:
  - Work-life balance
  - Job satisfaction
  - The team and the management
  - The commute time
  - And many many other things…

WORK-LIFE BALANCE

- Even if you value money or productivity more than a personal life, having a personal life is essential for your mental and physical health and creativity
- Make sure you get enough sleep, eat well, and cultivate enough time for personal hobbies and personal connections