# Creating Burn Ups: Perfection vs. Progress

ABCD Agile Working Group – February 8, 2024

Why Consider using a Burn Up Chart?

- Project management requires balancing **three constraints** to deliver a quality project/product.

- We knew we would have the **most control over scope**.

- Burn ups help measure progress and incomplete scope in an easy-to-read manner and **facilitate communication with stakeholders.**
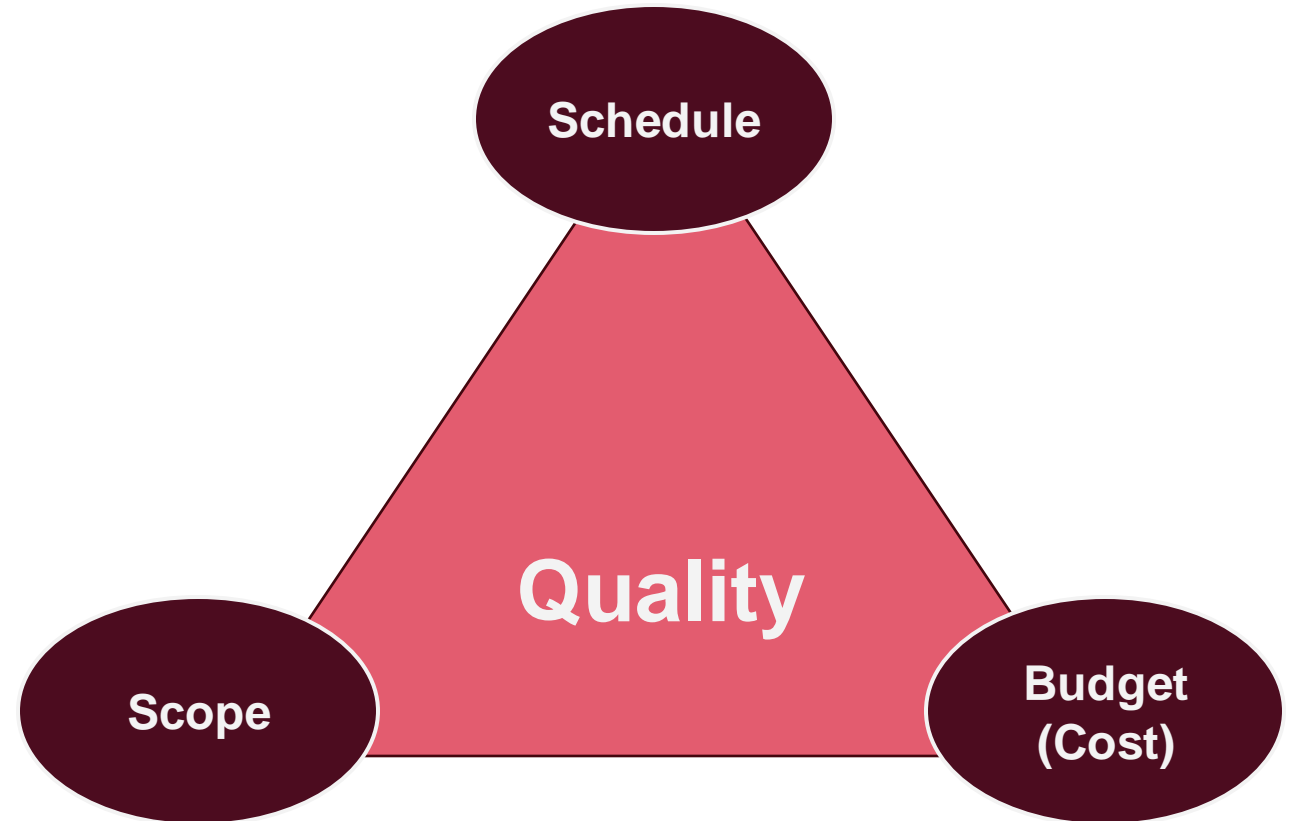
Understanding the Elements

Axes
- Sprints (Time)
- Story Points (Measure of workload)
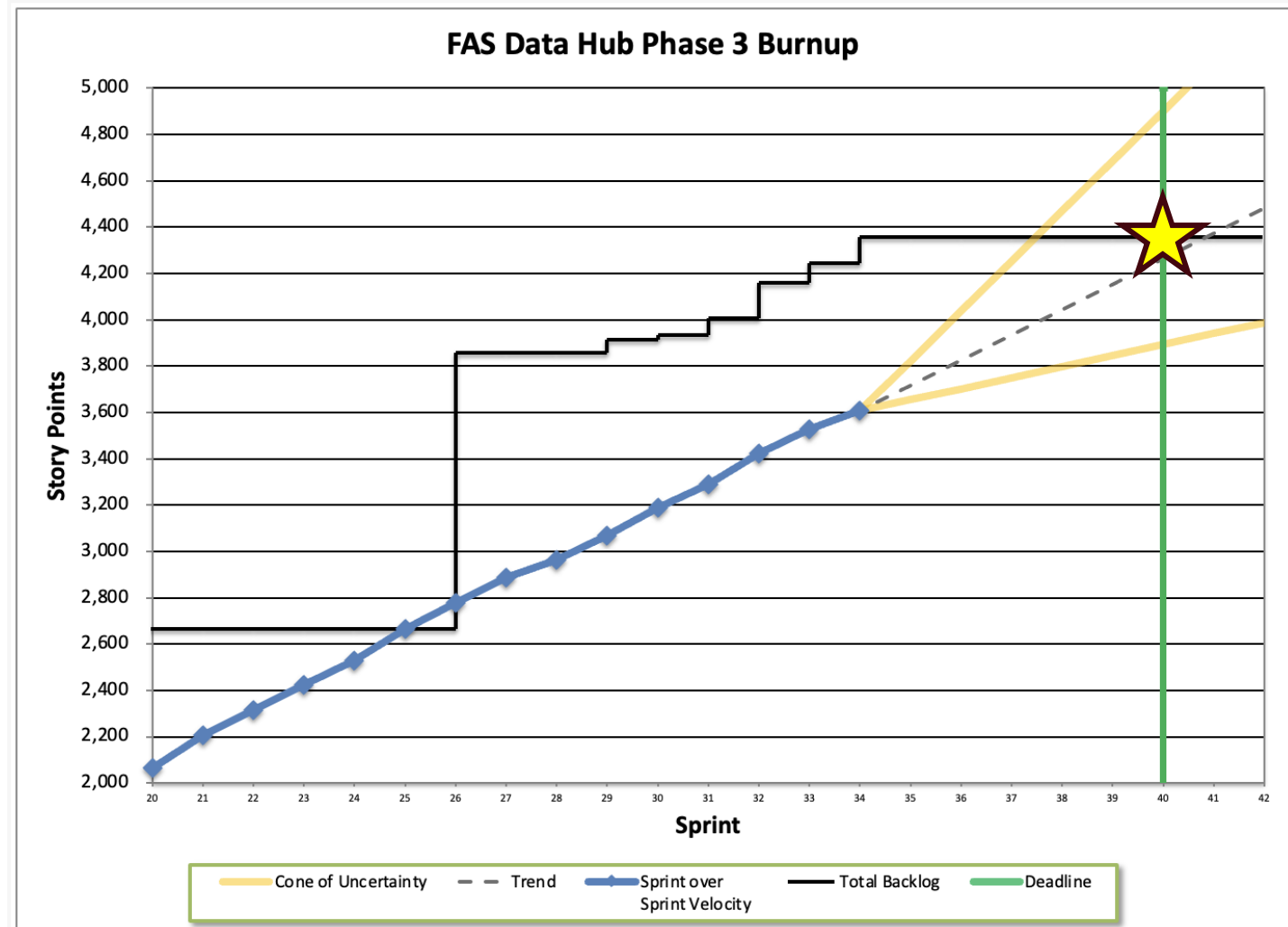
Deadline
- What we live by

Backlog
- How much work is included in the scope?

Velocity
- How much work does the team complete sprint over sprint

Trend + Cone of Uncertainty
- A prediction of how much work the team is likely to complete during future sprints

FAS Data Hub Phase 3 Burnup

But how do you get a a burn up when ...

- Our team had a little bit of agile experience, but most had not worked in traditional 2-week sprints to build one specific product

- Our backlog was empty

- We didn't know our own velocity

- We didn't know how our release cycle worked or what amount of overhead was associated with each release

- The team didn't point tickets in the same manner

- We were afraid our Executive Committee would never allow us to de-scope anything

1. Start with the Deadline

- The governance of our project and funding allocation cycle has allowed us to cleanly divide our project into phases, usually 4-6 months long.

- Our phases end with:
  - The delivery of a new product or product iteration
    - Goal is to define and deliver value to customers!
  - The end of a calendar or fiscal year
    - Allows the team to maintain sanity when going on winter break and rest
    - Facilitates financial management; we typically have a fixed resource allocation/ team size for each phase of the project and rebalance this in the next phase depending on the goals we have

UPSHOT: The deadline is mostly fixed

2. Measure the Backlog: Phase 1 – Starting from Scratch

How do you measure the backlog of a brand-new project?

- On Day 1, we had a **blank slate** in Jira and no stories in the backlog

- When we had conversations about the big features of our project, we **struggled to write manageably-sized tickets** to do the work

- We wrote tickets for the work at hand and tried to build out the backlog as time allowed

- We experimented with placeholder tickets worth many points – intending to break these into smaller tickets later – this led to ticket clutter and team confusion

Learnings

- There is **no substitute for discussion** when it comes to planning the work
  - A PM/ PO will never succeed in writing the whole backlog alone

- There is a tension between **planning the work vs. doing the work**
  - Both ultimately result in creation of new backlog items, but at different intervals

- All team members need to develop a **baseline understanding of the whole product** being built to support scoping efforts
  - Diverse perspectives spot the gaps in any technical plan

2. Measure the Backlog: Phases 2-5

How do you measure the backlog of a new project phase?

- As we ruthlessly **re-scope one phase**, we start to push items we know we won't accomplish into the **next phase's backlog**

- We reevaluate the backlog tickets and try to **surface themes/ features that need to stay together**

- We incorporate feedback received from stakeholders in the prior phase and use that to guide us towards a slate of possible work for the next phase

- We have our **Executive Committee prioritize**

- We write **more tickets** as fast as we can!

Learnings

- It's still hard!

- Write tickets on the spot and save them for later

- We started using "parking lot" sprints in Jira to group backlog items together thematically so they can be prioritized against one another and/or de-scoped as a group

- Expect the backlog to grow during the first half of the phase as the work becomes more clearer

- Always be ready to de-scope something

3. Velocity: "In Daylights, In Sunsets, In Midnights, In Cups of Coffee?"

How do you measure velocity of a brand new team?

- When a new team comes together, they don't know how much they'll be able to complete in a 2 week sprint
- One developer's idea of "Easy" is another's definition of "Complicated"; ticket pointing can be all over the place
- Result:
  - Sprint planning is HARD!
  - It's hard to know whether the points completed accurately measure the work the team completed

The impact of Velocity on the burn up chart

- The backlog line and the velocity lines will both be volatile if the team doesn't point tickets in a consistent manner
- Over time, this has leveled out as developers have agreed on what counts as 1-2-3-5-8 point ticket
- Once the velocity number begins to settle, the cone of uncertainty gets smaller and you can judge more accurately whether the team can complete the remaining scope by the deadline
- Understanding average velocity leads to more reasonable sprint scope definition

... How it's Going

- (Bad) Habits we've broken
  - ☐ Adding a ticket into the sprint just b/c someone thought of it that day
  - ☐ Taking on 2x as many tickets in a sprint than we could actually complete

- Things we're still improving
  - ☐ Making backlog refinement and ongoing practice
  - ☐ Starting from high level "rocks" and turning that into tickets or groups of tickets, and then refining
  - ☐ We capture meetings with stakeholders as tickets - they are work; yet we can't predict all the meetings we will need to have

- Harvard-specific Challenges
  - ☐ Project approval and funding cycles
  - ☐ Impact of Project Governance on agile
    - ☐ We have an Executive Committee that meets quarterly; we have to be able to talk about scope and resource challenges on that cadence

Backlog Management in Jira

- Sprint "Parking Lots" help us group and sequence work that is thematically related
- Numbered sprints are the traditional 2-week sprints our team actually goes through

Data Stream-specific Progress Check

- Just because the burnup says we can do it, doesn't mean we ACTUALLY can
- Sequencing tasks is key because some developers focus on just one or several pieces of a linear-ish development process

	Requirements definition	Source Design	Data Model	ETL Spec Write Up	ETL Development	RPD Spec Write Up	RPD Development	QC (Source view vs. ETL vs. RPD)	Dashboard (File) Development	Dept Profile RTF Development	Dept Profile Excel Development
Faculty - FAR	In Progress	Completed	Completed	Completed	Completed	Needs update	Completed	In Progress	In Progress	N/A	N/A
Faculty - ARIES	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed	In Progress	Blocked
Faculty - Aurora Hiring	Completed	Completed	Completed	Completed	Completed	Not started	Completed	Not started	Not started	Blocked	Blocked
Faculty - FROPS	Completed	Not started	Not started	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked
Faculty - General	Completed	Completed	Completed	Completed	Completed	Needs update	Completed	Completed	Completed	Needs update	In Progress
Researchers	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Resmpleted	Completed	Completed
Courses	Completed	Completed	Completed	Completed	Completed	Needs update	Completed	In Progress	Blocked	Completed	Completed
Classes	In Progress	Completed	Completed	Completed	Completed	Needs update	Completed	In Progress	Blocked	In Progress	TBD
Course & Enrollment Trends	Completed	Completed	Completed	Completed	Completed	N/A	In Progress	Blocked	In Progress	Blocked	Blocked
Staff	Completed	Completed	Completed	Completed	Completed	Needs update	Completed	Completed	Completed	In Progress	In Progress
Students - HCOL	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Stompleted	Completed	Completed
Students - HCOL Double Concentrators, Secondary Fields	Completed	In Progress	In Progress	N/A	N/A	Not started	Not started	Blocked	Needs update	Blocked	Blocked
Students - GSAS	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Stompleted	Completed	Completed
Space - FAS	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed
Finance	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed	Completed
Sponsored	In Progress	In Progress	N/A	N/A	N/A	Blocked	Blocked	Blocked	Blocked	Blocked	Blocked

Questions?

Challenge

- Create a platform for FAS Deans and Sr. Leaders to **access reliable, timely, and relevant information to inform decision-making, strategic planning, and operations** of academic departments and divisions.

- Academic Planning data is **distributed across multiple data sources** (HR, finance, grants management, student information, course offerings, physical space) and leaders struggled to access and analyze data cross-functionally.

- To guide academic planning and resource allocation, the FAS sought to **centralize and automate the consolidation of these data sources** to surface insights around departmental efficiency and strategic planning.