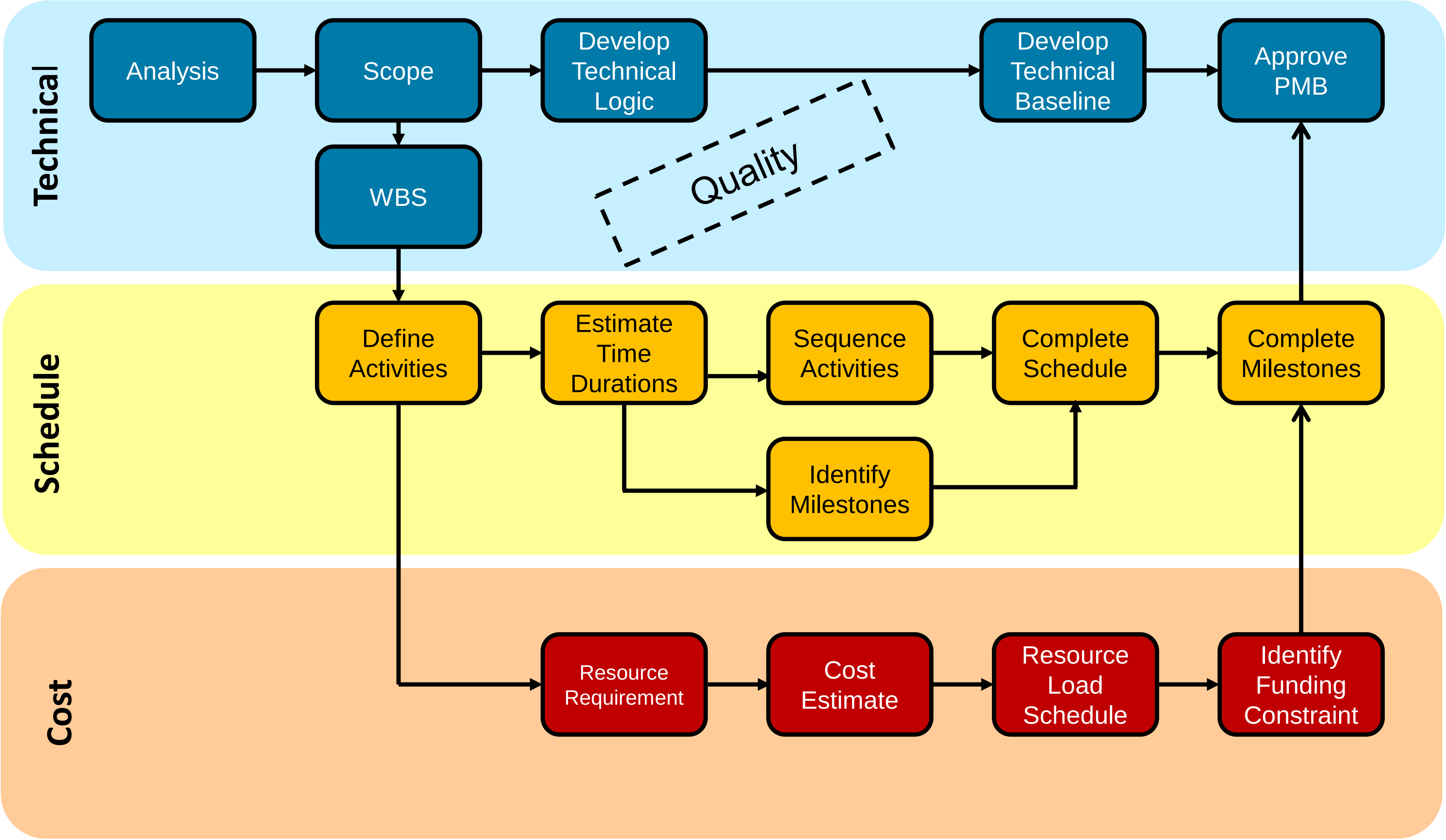
Agile Performance Measurement Baseline

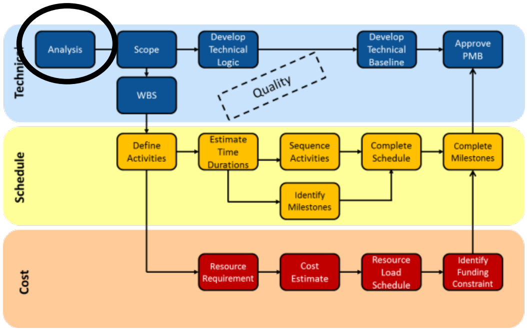
2020

Robin Yeman
LM Senior Fellow

The PMB is actually 3 baselines



Analysis



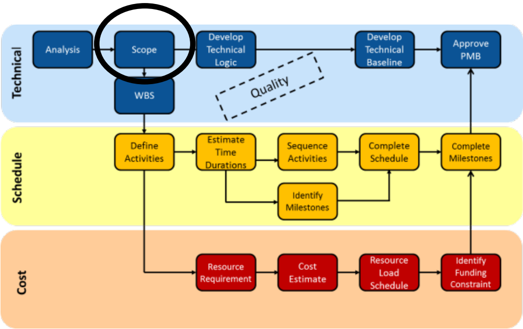
Identify the Requirements

Requirements Repository

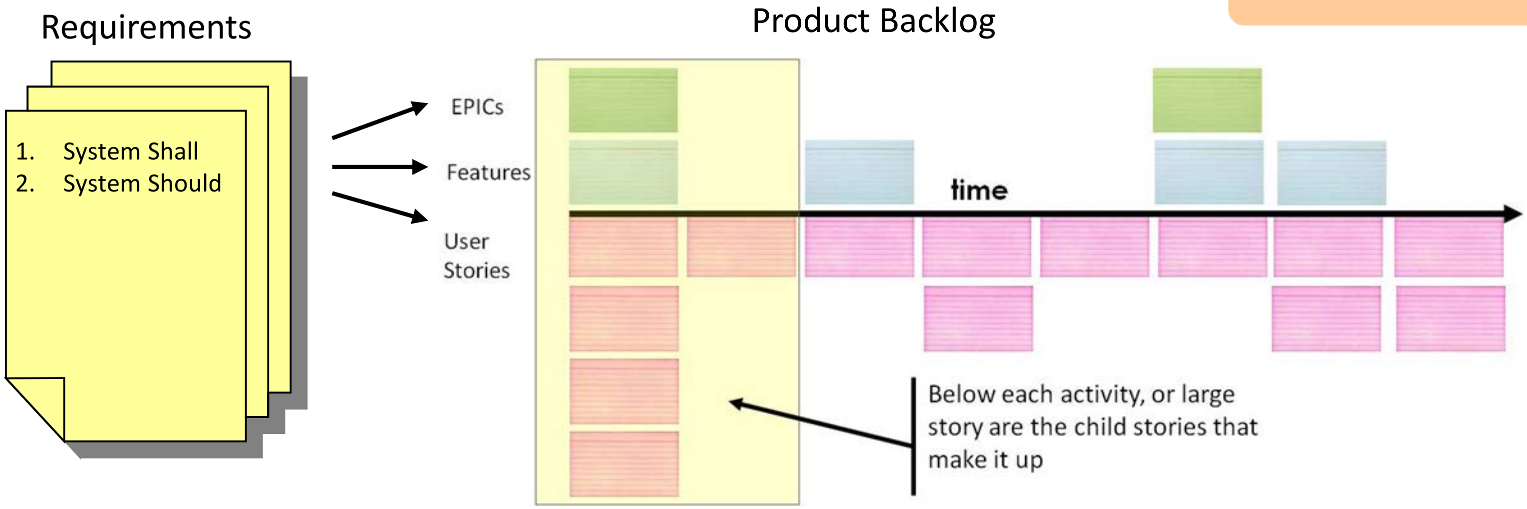
Product Backlog

Waterfall	Agile
System Design – A Spec	Epic
Component Design B-Spec	Sub Epic
Software/ Interface Requirements	Features
Detail Requirements	User Stories

Scope



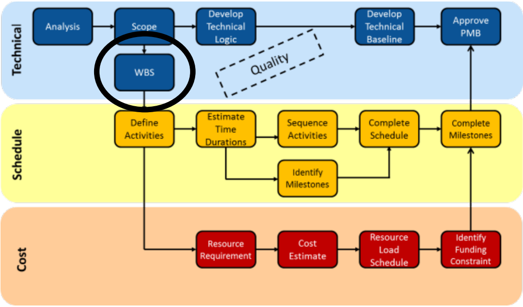
Based on the analysis of the requirements define the scope of the work. For Agile we place requirements in the form of user stories in a Product backlog.



Below each activity, or large story are the child stories that make it up

- ✓ Description
- ✓ Acceptance Criteria
- ✓ Deliverables
- ✓ Constraints

WBS Options



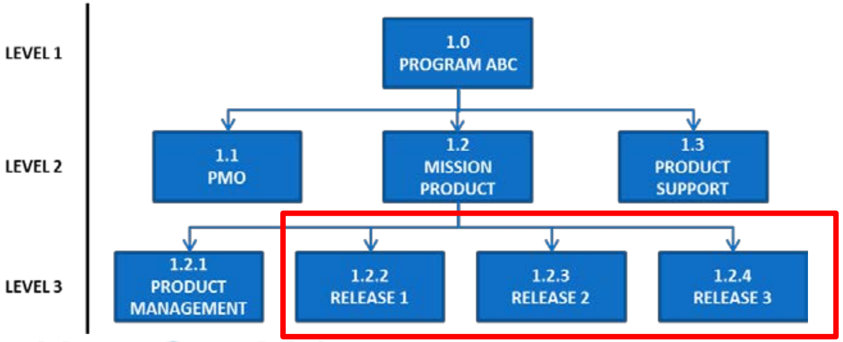
RELEASE CENTRIC

The customer views the product in terms of release. An example of this might be a large satellite ground system where the releases are based around major system events such as launch support, initial calibration, initial operations, and full system operations.

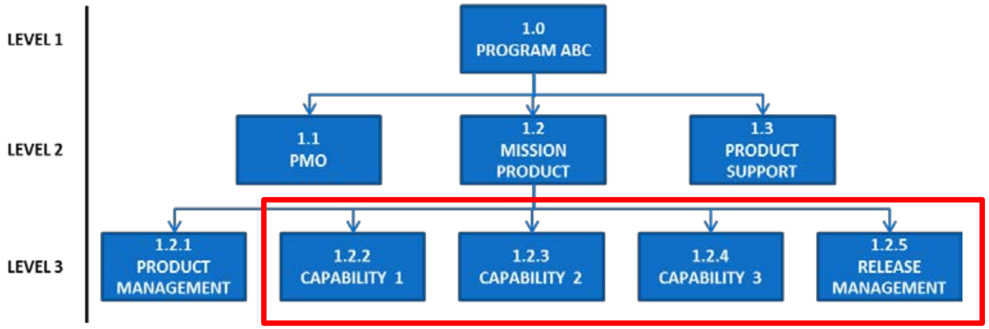
CAPABILITY CENTRIC

The customer views the product in terms of a set of discrete capabilities, where the releases are primarily viewed as time boxes for the ongoing and sustained delivery of Features. The release content may change greatly over time based upon changing priorities

WBS – RELEASE CENTRIC



WBS – CAPABILITY CENTRIC



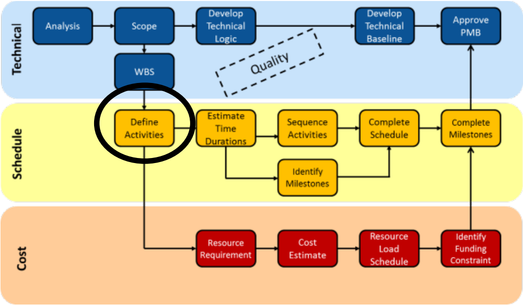
Poll #1

What type of Work Breakdown Structure do you currently have?

a. Functional
b. Product
c. Unsure

Define Activities

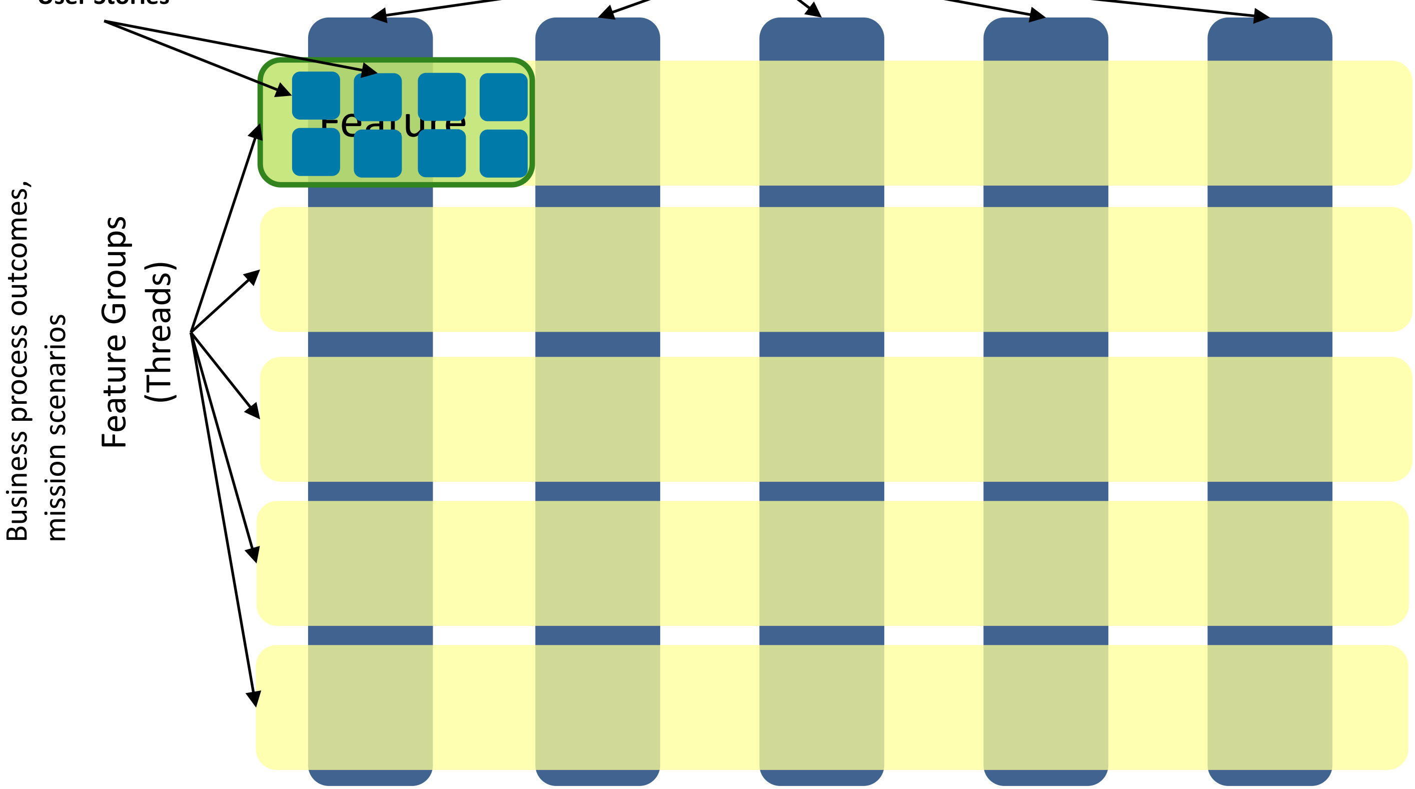
Define Activities for backlog, start with Epics and Feature groups and iteratively decompose into features and user stories



Epics

Cross cutting system capabilities, persistent through life-cycle

User Stories

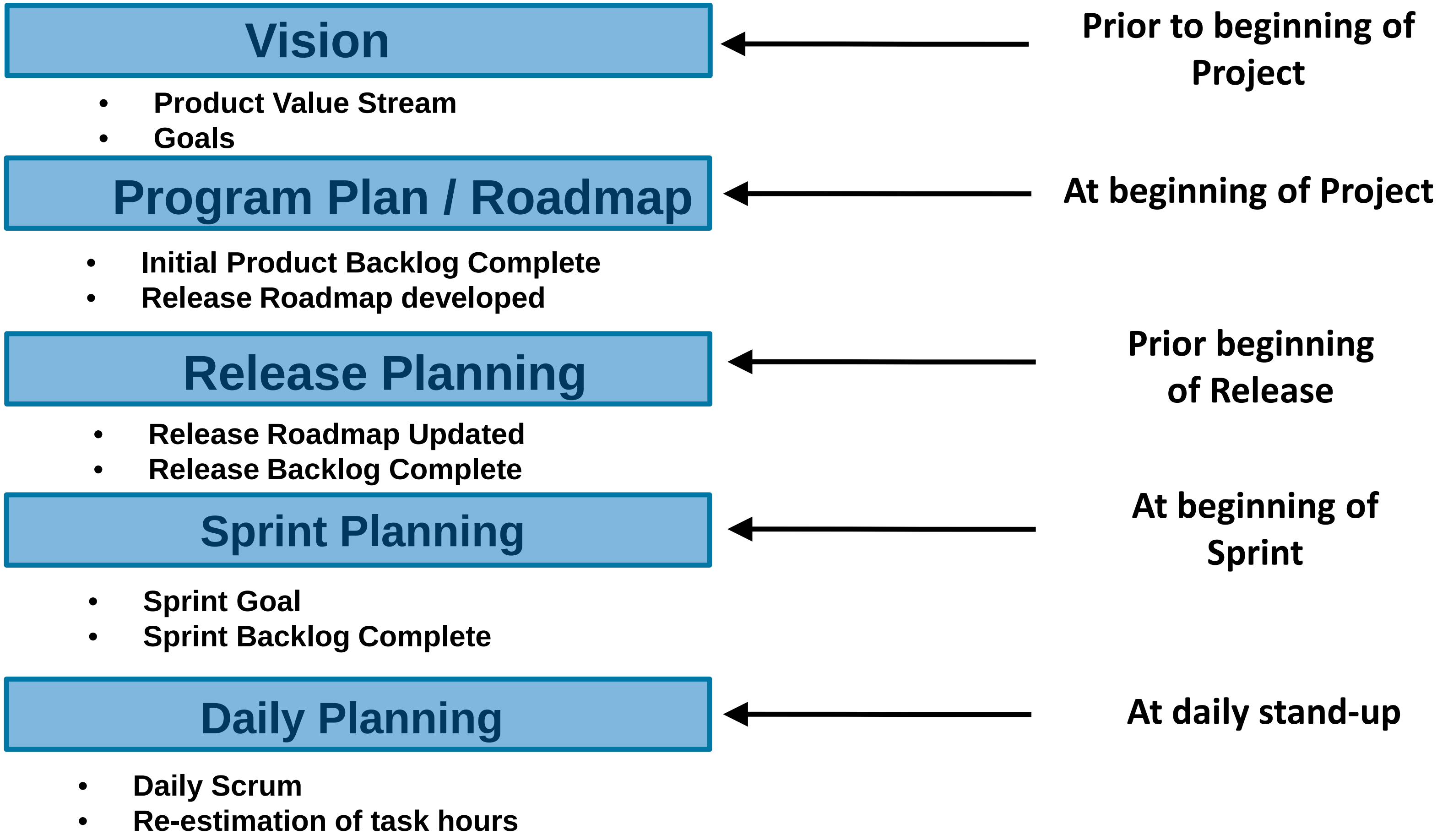


Feature

Business process outcomes, mission scenarios

Feature Groups (Threads)

Five Levels of Planning



Vision

- Product Value Stream
- Goals

Prior to beginning of Project

Program Plan / Roadmap

- Initial Product Backlog Complete
- Release Roadmap developed

At beginning of Project

Release Planning

- Release Roadmap Updated
- Release Backlog Complete

Prior beginning of Release

Sprint Planning

- Sprint Goal
- Sprint Backlog Complete

At beginning of Sprint

Daily Planning

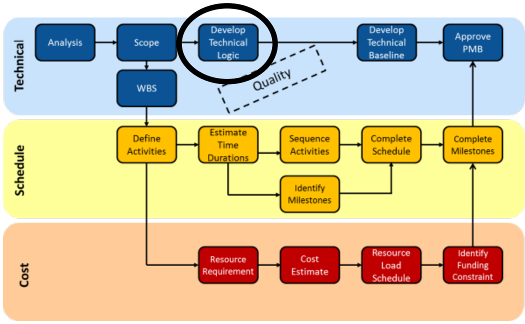
- Daily Scrum
- Re-estimation of task hours

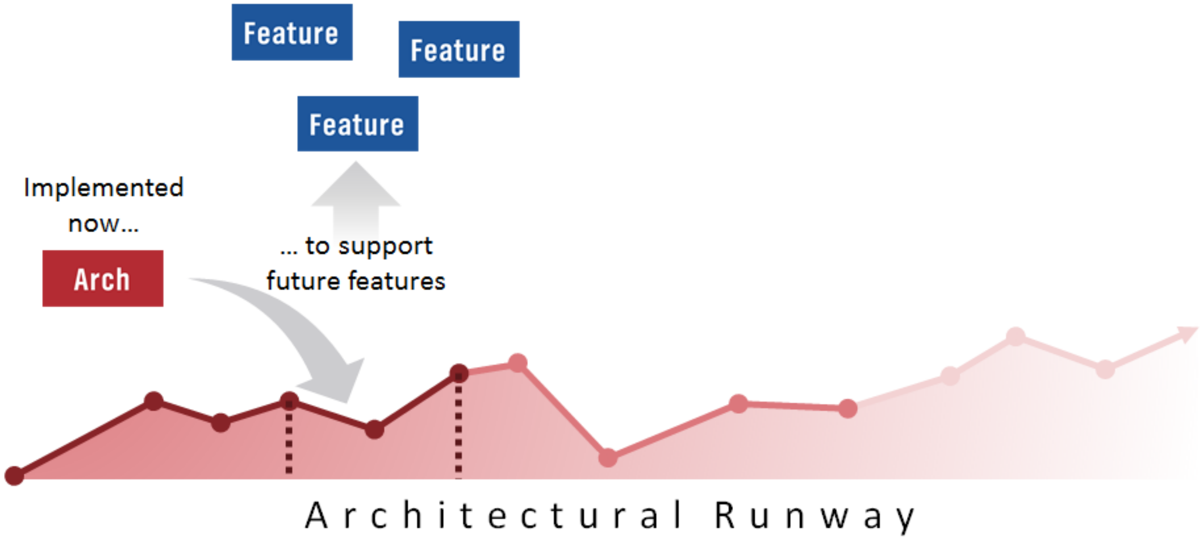
At daily stand-up

Define technical logic

- ✓ How much architectural runway
- ✓ Incremental pattern utilized
- ✓ Artifacts required
- ✓ Identify systems we interface
- ✓ Non-functional requirements
- ✓ Accreditation requirements

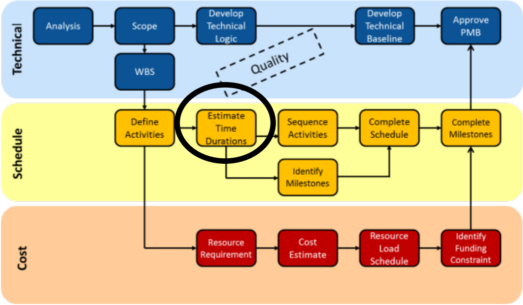
In Agile we keep this at a high level





Estimate time durations

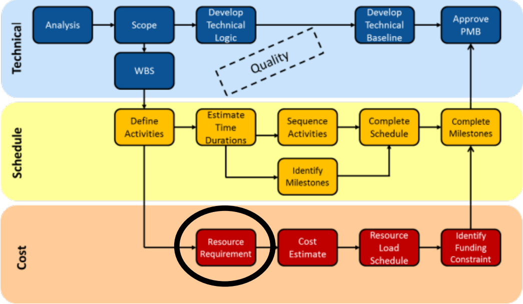
Size and duration estimates can be developed using any combination of the 4 methods below. In Agile we will estimate capabilities (Epic/Features) vs Functions (Software / Test)



Method	Description	Pro	Con
Expert Judgment	Judgment guided by subject matter experts based on historical experience	Rapid estimates based on a position of knowledge	Could miss variables and be too heavily weighted on single opinion
Analogous Estimating	Estimate parameters of project based on duration, budget, size, weight complexity. Adjusting for differences	Estimates proven on another project of similar size and complexity	Dependent on having projects of similar size and complexity
Parametric Modeling	Estimates performed based on variables such as function points or SLOC using SEER-SEM or Cocomo.	Provides an objective metric based on historical analysis of similar projects	With the 3rd and 4th generation languages, SLOC becomes less meaningful
3 point Estimates	Estimates based on a weighted average of most likely, optimistic, and pessimistic estimates	Looks at multiple points of view, and considers uncertainty and risk	Difficult to estimate large projects with.

Determine resource requirements

Determine the staffing profile of your project, based on skill sets required. The difference with Agile is that we are going to estimate the team requirements as opposed to individual functions because we do not know the actual individual performing the task



Technical
Analysis
Scope
Develop Technical Logic
Develop Technical Baseline
Approve PMB
WBS
Quality

Schedule
Define Activities
Estimate Time Durations
Sequence Activities
Complete Schedule
Complete Milestones
Identify Milestones

Cost
Resource Requirement
Cost Estimate
Resource Load Schedule
Identify Funding Constraint

Projected	Skill	Level
Tom A	Scrum Master/ Software	5
Robin D	Software Developer	4
Ian B	Software Developer	3
Scott Y	Software Developer	2
Jeff T	Requirements Analyst	3
Helen W	Test Engineer	4
Paul R	Test Engineer	3
James B	Database Engineer	4
		3.5

Develop and average labor cost across team

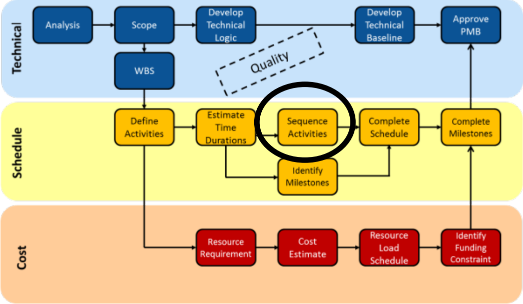
Hint:
Who you need is not necessarily who you have today

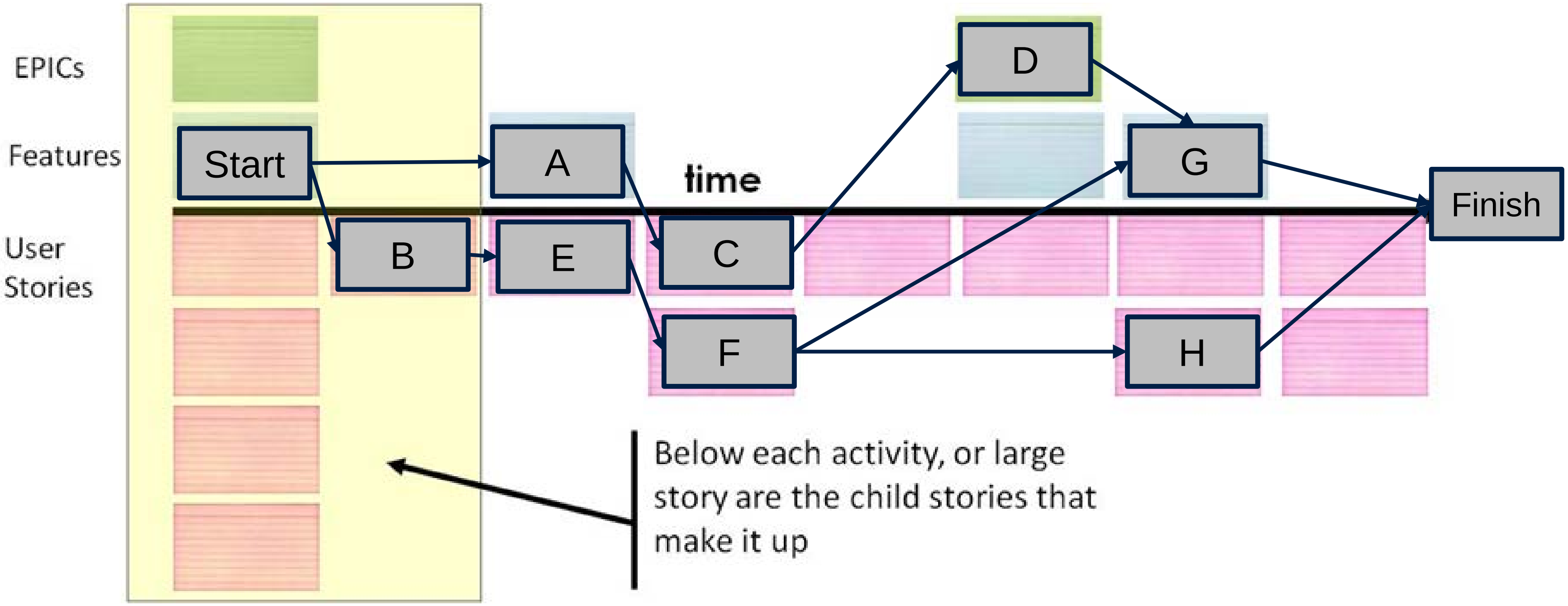
Hint:
*The best team results in a **3** to **3.5** when levels averaged across resources*

NextGenLM

Sequence activities

Program will sequence their activities. In Agile programs we refer to this as story mapping. On traditional projects sequencing activities is known as the Precedence Diagramming Method (PDM).

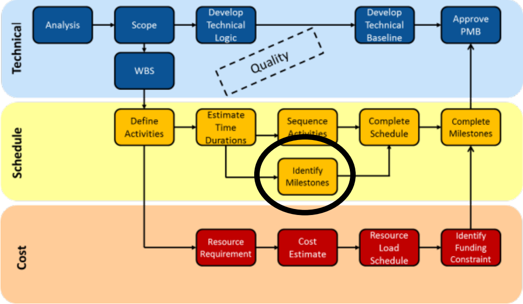




Below each activity, or large story are the child stories that make it up

Identify milestones

Identify and list their key milestones. However in Agile programs we focus on outcomes as opposed to document and design reviews to take credit.



Traditional

Milestone	Date
Initial Baseline Review (IBR)	03/04/2020
System Requirements Review (SRR)	05/27/2020
Preliminary Design Review (PDR)	08/19/2020
Critical Design Review (CDR)	11/11/2020
Test Readiness Review (TRR)	02/03/2021
Operational Readiness Review (ORR)	04/28/2021
Project Closeout Review (PCR)	07/21/2021

Agile

Milestone	Date
Product Vision Complete	03/04/2020
Product Plan / Roadmap Complete	03/18/2020
Initial Baseline Review (IBR)	04/15/2020
Release Roadmap complete	05/27/2020
Release 1 Demo (Feature 1-4 complete)	08/19/2020
Release 2 Demo (Feature 5-9 Complete)	11/11/2020
Release 3 Demo (Feature 10-13 Complete)	02/03/2021
Release N Demo (Feature 14-17 Complete)	04/28/2021
Project Closeout Review (PCR)	07/21/2021

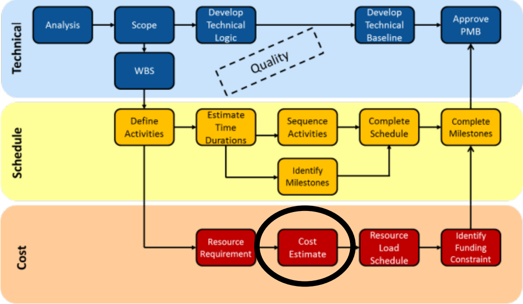
Which schedule has more risk 02/03/2021

Poll #2

Which Schedule has more risk on 02/03/21?

a. Traditional
b. Agile
c. Unsure

Cost estimate



There is very little difference in how teams estimate costs between Agile and traditional projects. We often see many efficiencies and risk reductions which enable Agile projects to be lower cost, when run properly.

Scope
(Epic/ Feature)

Schedule

Resource Plan

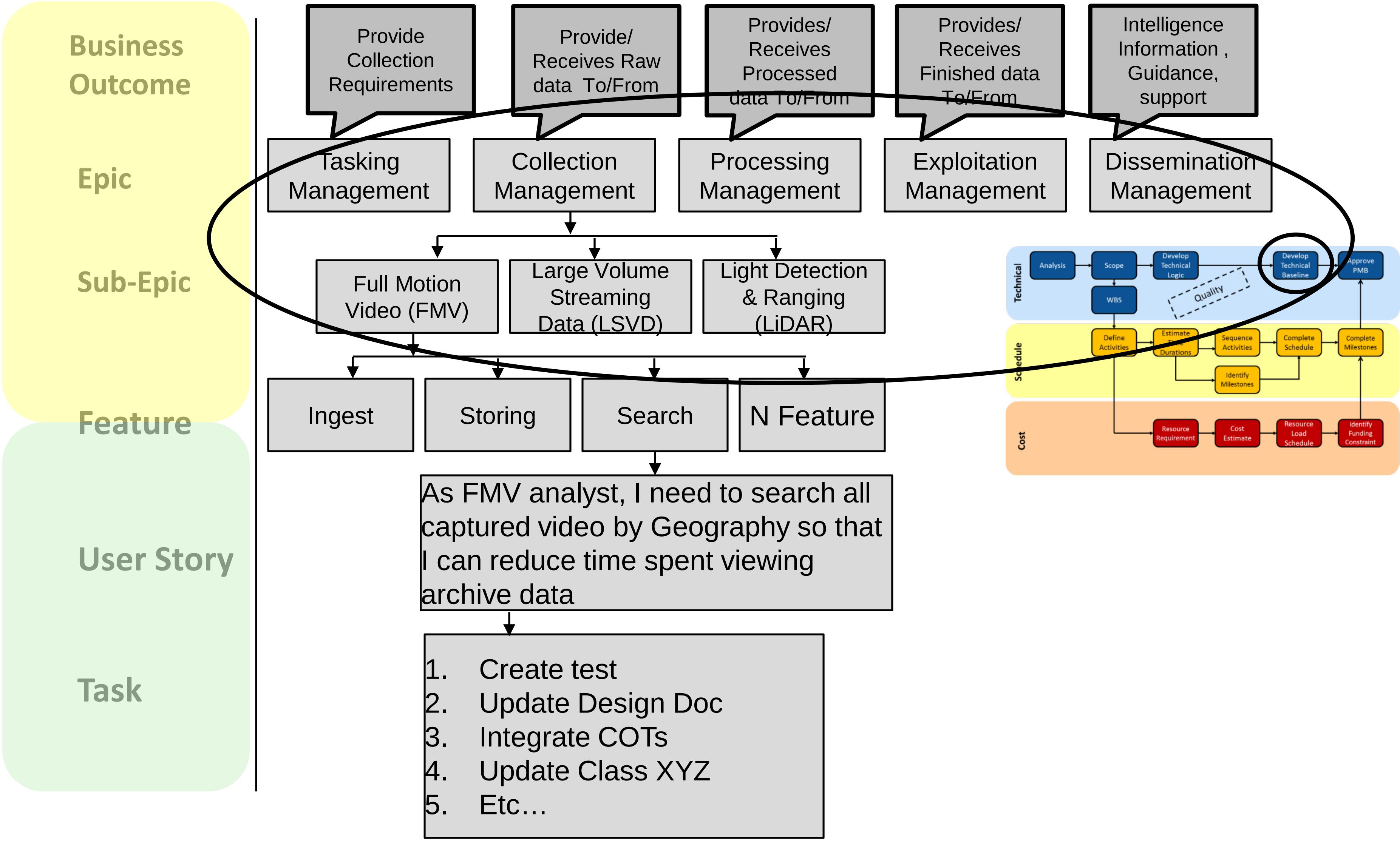
Risks

Cost Estimation

Project estimates are a range that will continually need to be reviewed and refined

NextGenLM

Develop technical baseline



Business Outcome

Epic

Sub-Epic

Feature

User Story

Task

Provide Collection Requirements

Provide/ Receives Raw data To/From

Provides/ Receives Processed data To/From

Provides/ Receives Finished data To/From

Intelligence Information , Guidance, support

Tasking Management

Collection Management

Processing Management

Exploitation Management

Dissemination Management

Full Motion Video (FMV)

Large Volume Streaming Data (LSVD)

Light Detection & Ranging (LiDAR)

Ingest

Storing

Search

N Feature

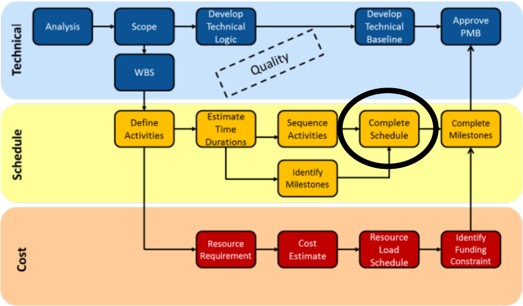
As FMV analyst, I need to search all captured video by Geography so that I can reduce time spent viewing archive data

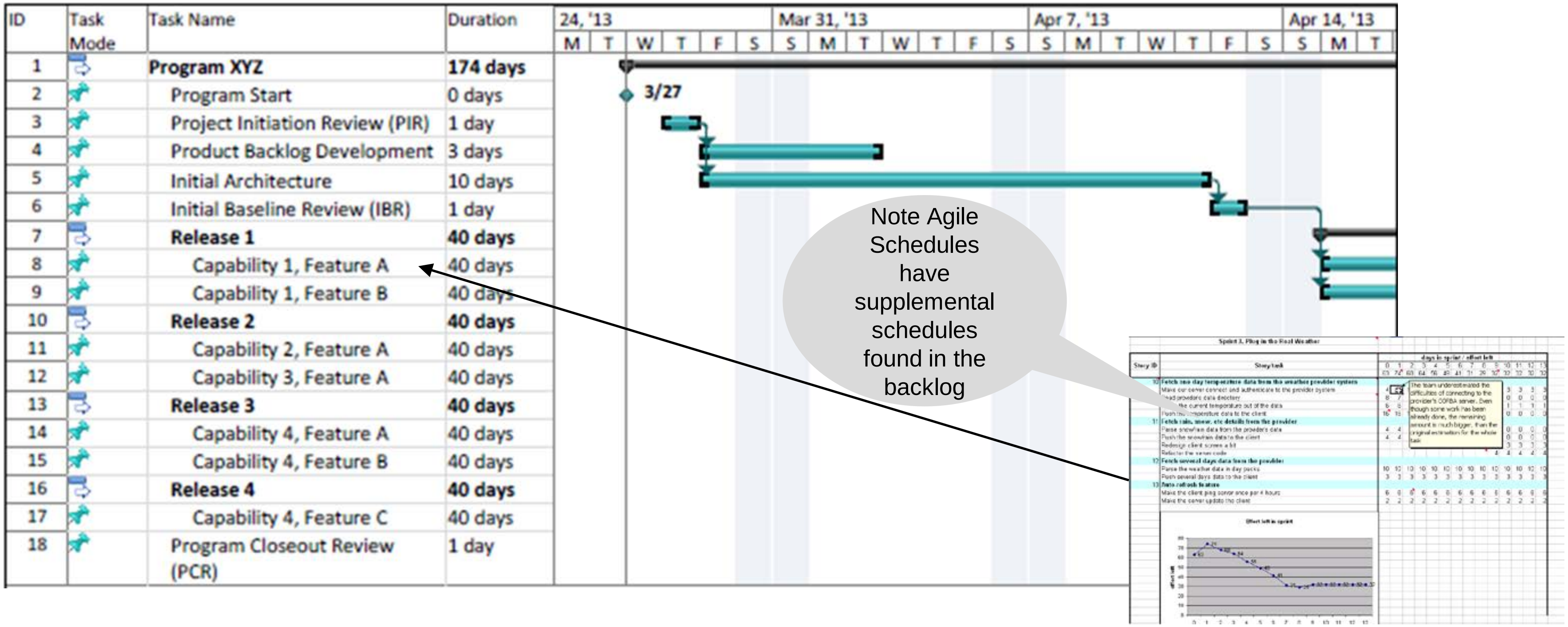
1. Create test
2. Update Design Doc
3. Integrate COTs
4. Update Class XYZ
5. Etc…

NextGenLM

Baseline current schedule

After we have created a high level schedule, we will baseline the schedule. With Agile programs schedules will continuously be revisited and monitored to ensure they are still accurate



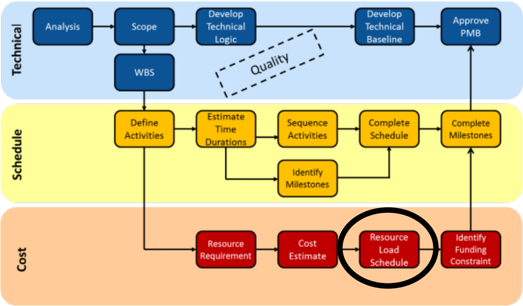


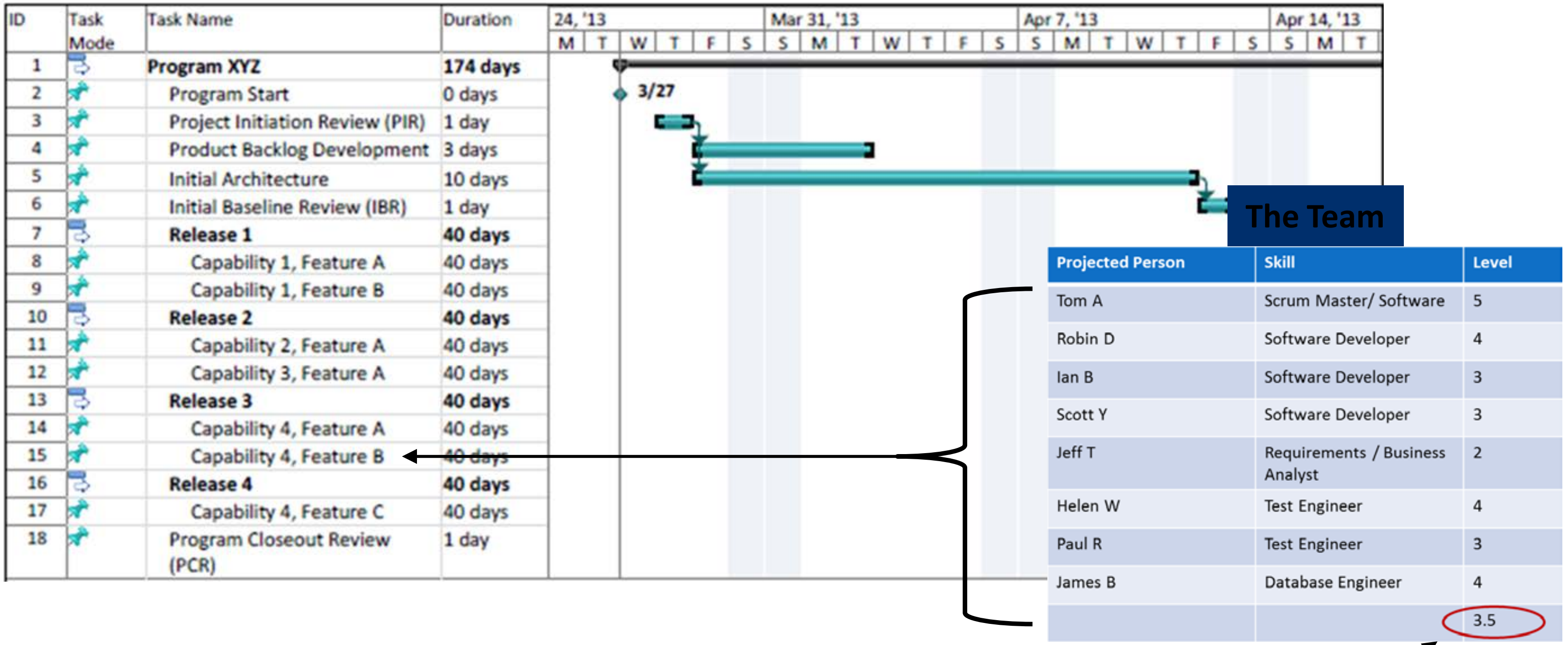
ID	Task Mode	Task Name	Duration
1		Program XYZ	174 days
2		Program Start	0 days
3		Project Initiation Review (PIR)	1 day
4		Product Backlog Development	3 days
5		Initial Architecture	10 days
6		Initial Baseline Review (IBR)	1 day
7		Release 1	40 days
8		Capability 1, Feature A	40 days
9		Capability 1, Feature B	40 days
10		Release 2	40 days
11		Capability 2, Feature A	40 days
12		Capability 3, Feature A	40 days
13		Release 3	40 days
14		Capability 4, Feature A	40 days
15		Capability 4, Feature B	40 days
16		Release 4	40 days
17		Capability 4, Feature C	40 days
18		Program Closeout Review (PCR)	1 day

Note Agile Schedules have supplemental schedules found in the backlog

Resource load

All programs need to understand their resource allocation in order to understand whether they can successfully complete the project. Agile programs load *teams* against the schedule as opposed to *individuals*. The team is responsible for completing all work needed to complete the project.





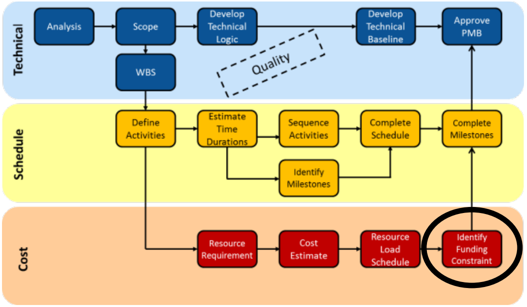
ID	Task Mode	Task Name	Duration
1	⇨	**Program XYZ**	**174 days**
2	📌	Program Start	0 days
3	📌	Project Initiation Review (PIR)	1 day
4	📌	Product Backlog Development	3 days
5	📌	Initial Architecture	10 days
6	📌	Initial Baseline Review (IBR)	1 day
7	⇨	**Release 1**	**40 days**
8	📌	Capability 1, Feature A	40 days
9	📌	Capability 1, Feature B	40 days
10	⇨	**Release 2**	**40 days**
11	📌	Capability 2, Feature A	40 days
12	📌	Capability 3, Feature A	40 days
13	⇨	**Release 3**	**40 days**
14	📌	Capability 4, Feature A	40 days
15	📌	Capability 4, Feature B	40 days
16	⇨	**Release 4**	**40 days**
17	📌	Capability 4, Feature C	40 days
18	📌	Program Closeout Review (PCR)	1 day

The Team

Projected Person	Skill	Level
Tom A	Scrum Master/ Software	5
Robin D	Software Developer	4
Ian B	Software Developer	3
Scott Y	Software Developer	3
Jeff T	Requirements / Business Analyst	2
Helen W	Test Engineer	4
Paul R	Test Engineer	3
James B	Database Engineer	4
		3.5

Fictional Average = $100 hour

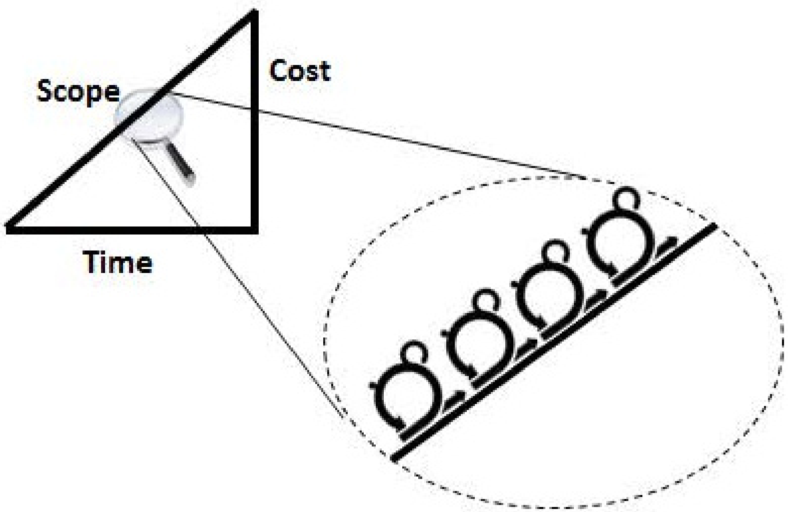
Identify any funding constraints



Before teams can complete their performance measurement baseline, they need identify and analyze any potential funding Constraints they may have.

- ✓ Contract terms and conditions
- ✓ Appropriation of funds
- ✓ Budget profiles

The benefit of Agile, is that programs can get started even if a customers funding profile does not cover the entire scope of work. Customers can purchase incremental features, with regular feedback cycles to prioritize.



Incrementally complete milestones

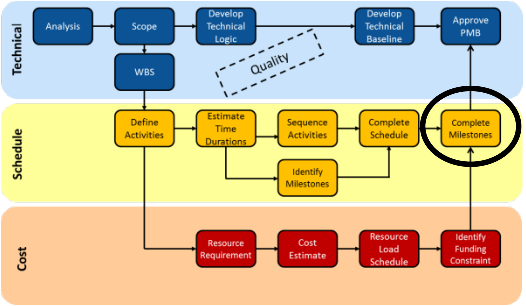
Once the baseline is instantiated, teams can begin to complete milestones incrementally, allowing us to be responsive to stakeholder’s changing needs.

Ensure we begin with the end in mind, Clear acceptance criteria

What

Measurable success Criteria

- New Biometrics System
- Automates 85% of workflows
- 80% of external systems integrated
- 80% of users satisfied with interface
- Single Sign on
- 100% secure



How

ID	Feature
1	Implement Log-in
2	Automated Export to excel
3	Integrate with system x
4	Integrate with system y
….	….
180	User Preferences
181	Security Feature

Focus on the What, this aligns to business value

QUESTIONS
&
ANSWERS