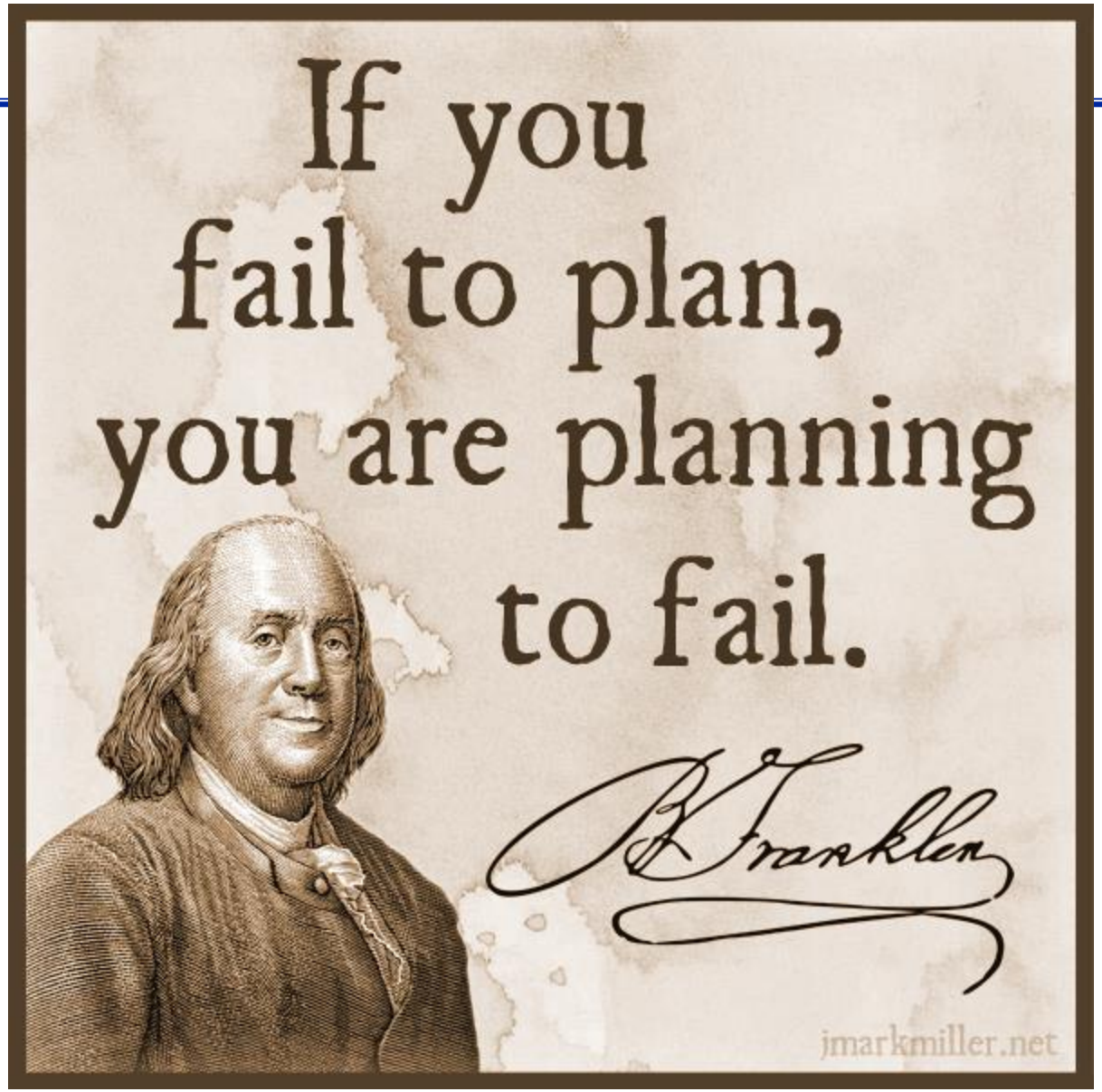
Week Two Lecture

Agenda: (50 slides in 50 minutes)

1. Project Management

2. Methodologies

3. Project Management Tools


If you
fail to plan,
you are planning
to fail.
B Franklin
jmarkmiller.net

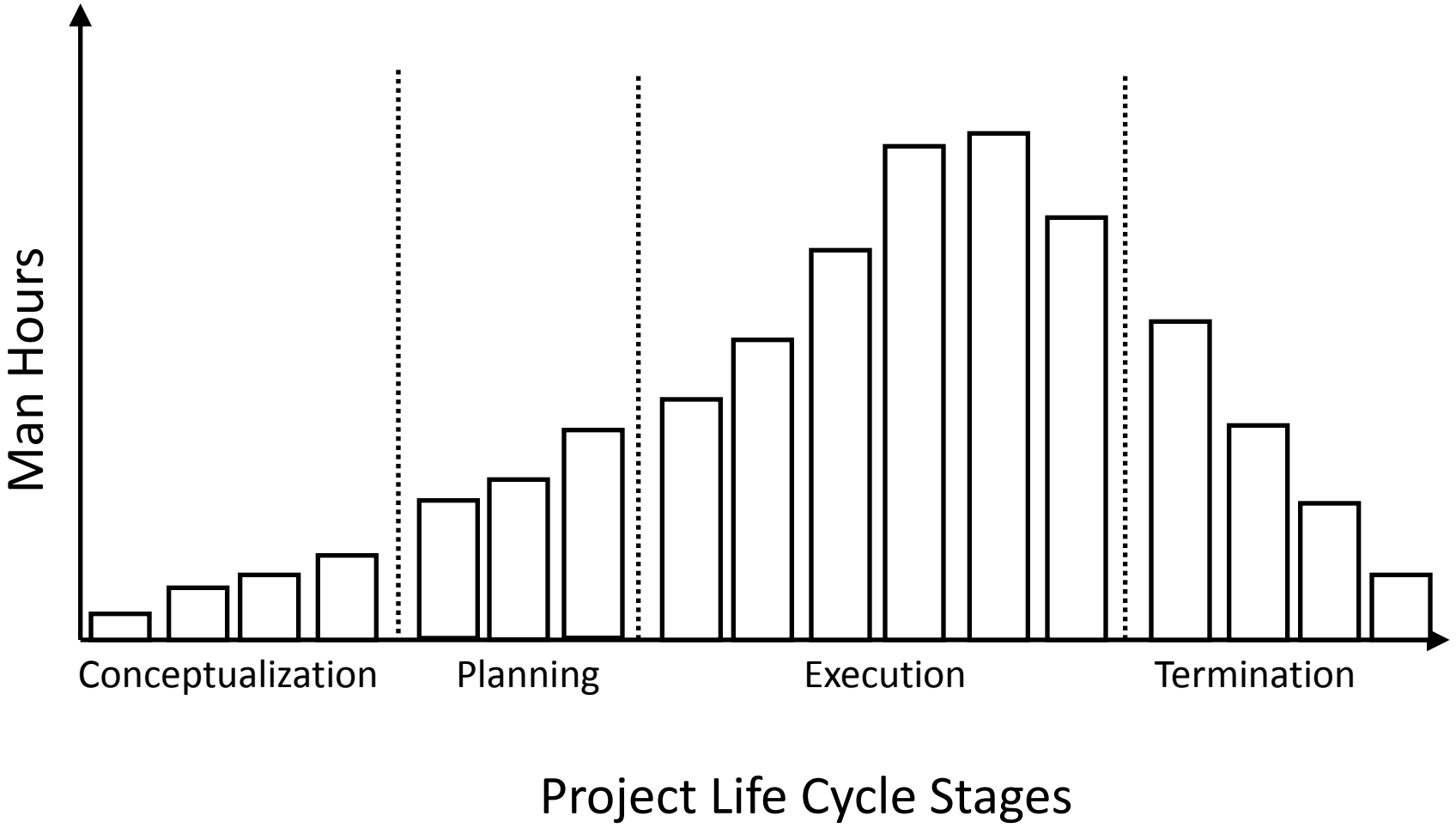
Project Management

- **How do we as organizations advance and grow?**
  - Through PROJECTS
- **IT Project Failure Rates?**
  - 50%, 65%, 70% ?
  - Statistics
- **The role of Project Manager**
  - A relatively recent addition to the workforce
    - Outside civil engineering and construction
    - Jobs
  - A structured, scientific discipline
    - PMI Certification is a big PLUS
    - Requires solid people skills

Project Management

- **What is a project?**
  - A series of **tasks**
  - Has a **distinct beginning** and a **distinct end**
  - Goal oriented
  - Bound by **time**, **cost** and **quality**
  - Requires resources (human and material)
  - Solves problems; satisfies needs
  - Stems from a larger strategy/vision
  - Has a **customer** and/or **sponsor** ($$)
  - Distinct from a process

- **The project life cycle?**
  - Typical Phases:
    - Inception/Conceptualization
    - Planning
    - Execution
    - Termination


Man Hours
Conceptualization
Planning
Execution
Termination
Project Life Cycle Stages

Project Management

- **How do we define success**
  - On time
  - On budget
  - Meets objectives
- **Triple Constraint Model**
  - Time, Cost, Scope
- **Quadruple Constraint Model**
  - Add Customer Satisfaction

- Projects must be delivered within cost
- Projects must be delivered on time
- Projects must meet the agreed scope – no more, no less
- Projects must also meet customer quality requirements

Figure 1. The Triple Constraint

- **Scope**
  - Easily understood as a sum of REQUIREMENTS

"If it can't do……., I don't want it."

"It must be able to ……."

"I want a system that will ……."

What's IN; What's OUT

- **What is SCOPE?**
  - Everything about a project
  - The work that must be done
  - The results that will be delivered
  - The solution provided

- **Scope must be managed**
  - Scope ➔ Requirements
  - Projects fail mostly due to unmanaged scope
    - Undefined requirements
    - Poorly defined requirements
    - Shifting requirements

- **Work Breakdown Structure ("WBS")**
  - A deliverable-oriented grouping of project elements which organizes and defines the total scope of the project
  - "Drill-Down" to tasks
  - Each descending level represents an increasingly detailed definition of a project component
  - A component may be a product or service

WBS -- Make pies for Thanksgiving Dinner

Deliverable 1	**Define Pie Menu**	
WP1	Interview Mom	1.1
WP2	Document Menu	1.2
Deliverable 2	**Groceries**	
WP1	Shopping List	2.1
WP2	Check Inventory	2.2
WP3	Purchase	2.3
Deliverable 3	**Crusts**	
WP1	Measure ingredients	3.1
WP2	Make Dough	3.2
WP3	Roll Dough	3.3
WP4	Install in pie pan	3.4

Deliverable 4	**Filling**	
WP1	Mix Pumpkin Filling	4.1
WP2	Mix Pecan Filling	4.2
Deliverable 5	**Bake**	
WP1	Place pies in oven	5.1
WP2	Set Timer	5.2
WP3	Bake	5.3
WP4	Remove from oven	5.4
Deliverable 6	**Finish**	
WP1	Set out pies to cool	6.1

- **Successful Project Management**
  - Create a thorough WBS
  - Manage scope/requirements changes
  - Have a plan

Remember:

"If you fail to plan, you plan to fail."

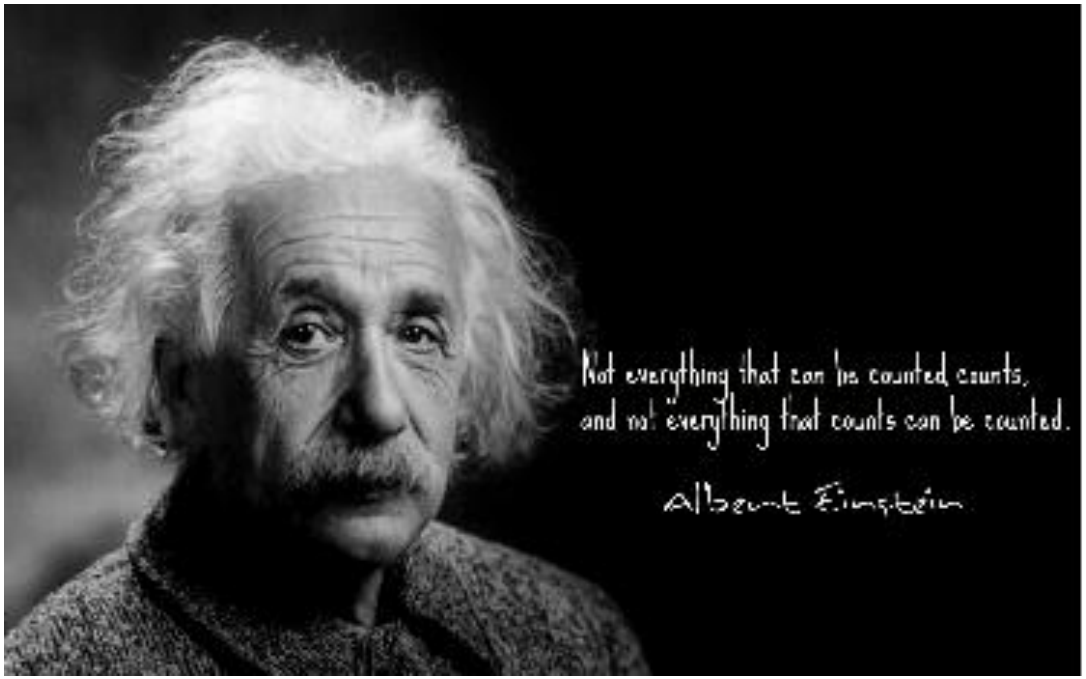
Project Management

- **Successful Project Management**
  - Measure progress
  - Communicate progress

Remember:

"You cannot manage what you do not measure"

Project Management



Project Management "Methodologies"

- "…ology" means "the study of …"
  biology, mythology, zoology, anthropology, pathology

- In this case, there are many methods of managing projects.

- If we study each method, we can describe this study as

  "project management methodology"

- However, the term has morphed into referring to ***the method*** itself, not so much the study of the method

There are many PM Methodologies

SDLC (system development life cycle)

Waterfall

PMBOK (Project Management Body of Knowledge)

Agile

Iterative

We will look at TWO

SDLC (system development life cycle)

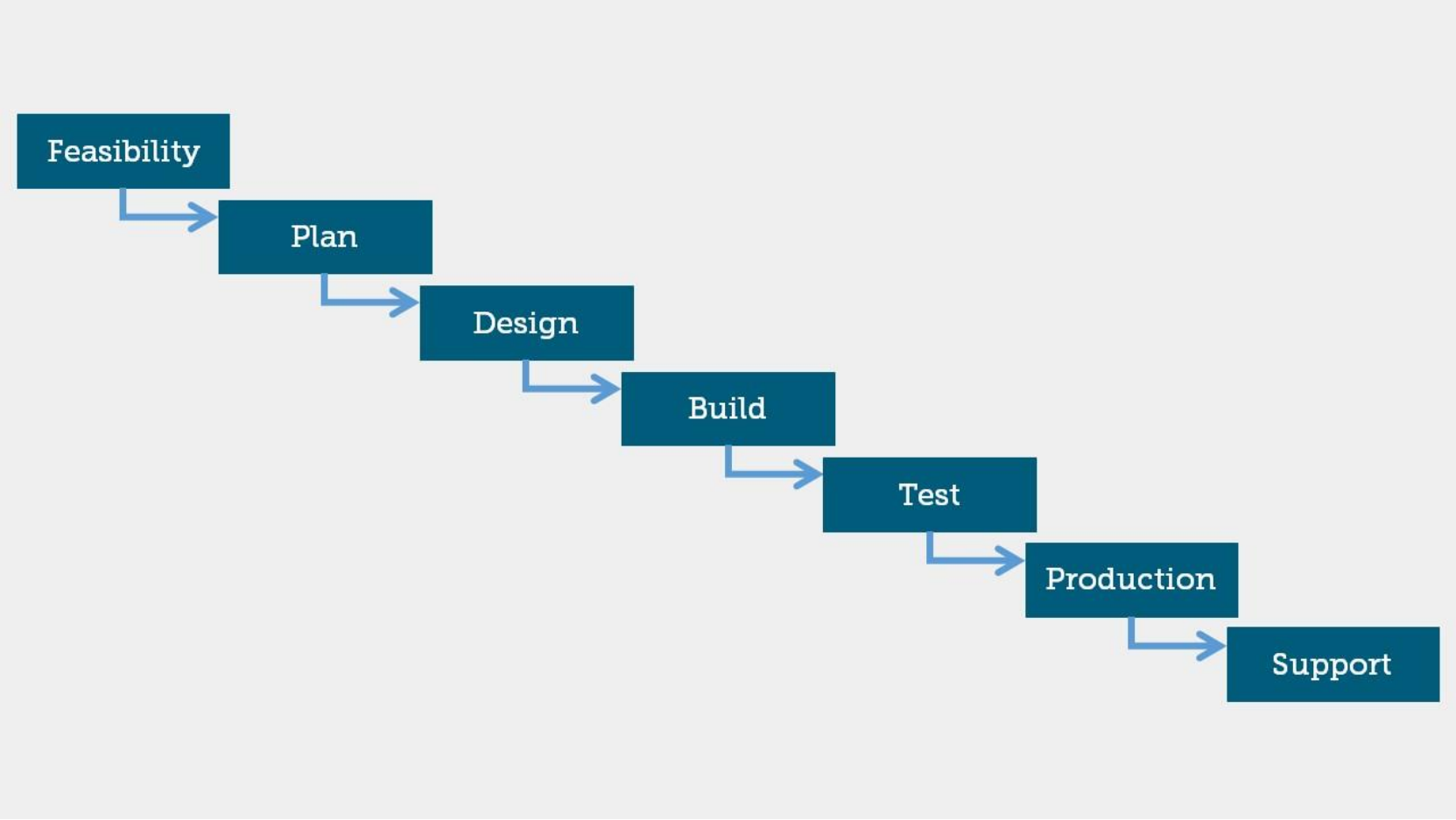
Waterfall

PMBOK (Project Management Body of Knowledge)

Agile

Iterative

Why “waterfall”?



Benefits of Waterfall

- Project sponsors get a clear picture of the complete final product
- Project sponsors get an accurate, detailed estimate of the cost of the project
- A “GO/NO GO” decision is made at the end of each phase
- Requires a detailed and accurate Work Breakdown Structure early-on in the project
- The scope of the project is locked down at that time, and should remain unchanged until solution delivery
- Any changes to scope must be strictly managed
- A decision to change scope demands renegotiating the project timeline and costs

Benefits of Waterfall

- Project sponsors get a clear picture of the complete final product
- Project sponsors get an accurate, detailed estimate of the cost of the project
- A "GO/NO GO" decision is made at the end of each phase
- Requires a detailed and accurate Work Breakdown Structure early-on in the project
- The scope of the project is locked down at that time, and should remain unchanged until solution delivery
- Any changes to scope must be strictly managed
- A decision to change scope demands renegotiating the project timeline and costs

Benefits of Waterfall

This is precisely why so many IT projects fail, and why software development project managers have embraced AGILE

- The scope of the project is locked down at the outset, and should remain unchanged until solution delivery
  - Any changes to scope must be strictly managed
  - A decision to change scope demands renegotiating the project timeline and costs

- **Waterfall Challenges**
  - A successful project demands a detailed and accurate Work Breakdown Structure early-on
  - The scope of the project is locked down at that time, and should remain unchanged until solution delivery
  - Changes to scope, even when well-managed, will extend project timeline and costs
  - Once requirements are defined, customers are much less engaged during the design and construction phases
  - Customers must wait until the project is over to enjoy any tangible benefit from the project

- **Waterfall Challenges**
  - A curate
    W
  - The scope of the project is locked down at that time, and should remain unchanged until solution delivery
  - Changes to scope, even when well-managed, will extend project timeline and costs
  - Once requirements are defined, customers are much less engaged during the design and construction phases
  - Customers must wait until the project is over to enjoy any tangible benefit from the project

What if "We don't know…"

- **Waterfall Challenges**
  - A curate W
  - Th ar
  - Changes to scope, even when well-managed, will extend project timeline and costs
  - Once requirements are defined, customers are much less engaged during the design and construction phases
  - Customers must wait until the project is over to enjoy any tangible benefit from the project

What if "We don't know…"

What if they change their minds?

- **Waterfall Challenges**
  - A curate W
  - Th ar
  - C ex
  - O less engaged during the design and construction phases
  - Customers must wait until the project is over to enjoy any tangible benefit from the project

What if "We don't know…"

What if they change their minds?

What if they demand changes, but won't let me change the budget or timeline?

- **Waterfall Challenges**
  - A curate
    W
  - Th
    an
  - C
    ex
  - O
    le
    ph
  - C
    any tangible benefit from the project

What if "We don't know…"

What if they change their minds?

What if they demand changes, but won't let me change the budget or timeline?

What if they want to see results before we're ready?

- **The "Waterfall" model**

It's really difficult to go back upstream !!

The Agile "Manifesto"

On February 11-13, 2001, at The Lodge at Snowbird ski resort in the Wasatch mountains of Utah, seventeen people met to talk, ski, relax, and try to find common ground.

What emerged was the Agile 'Software Development' Manifesto.

http://agilemanifesto.org/

The Agile "Manifesto"

- Focused on Software Development Projects

The Agile Manifesto:

"We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

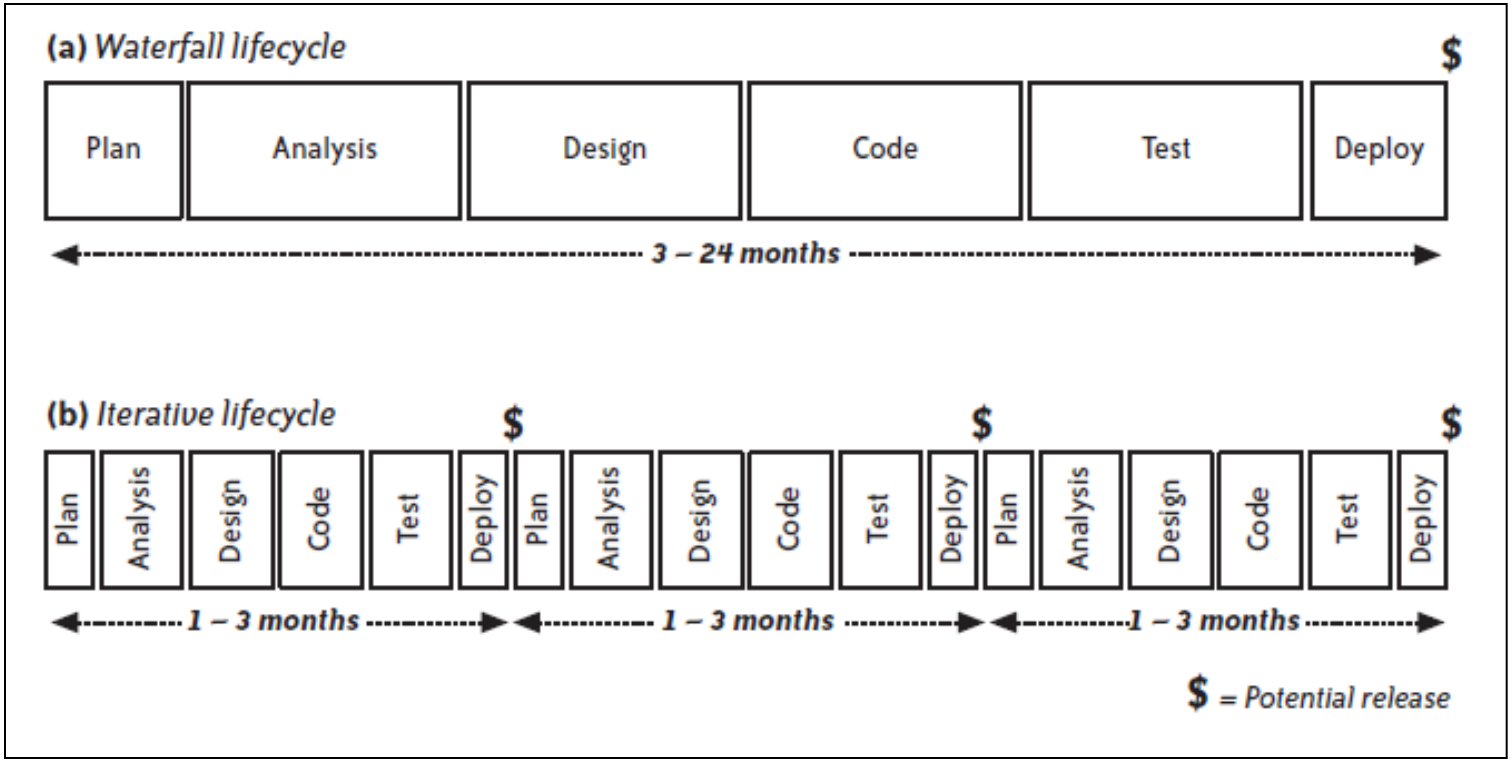
Individuals and interactions	over	processes and tools.
Working software	over	comprehensive documentation.
Customer collaboration	over	contract negotiation.
Responding to change	over	following a plan.

That is, while there is value in the items on the right, we value the items on the left more."

Plan the entire project up front

VS

Incremental, iterative development cycles



Traditional	Agile
Design up front	Continuous design
Fixed scope	Flexible scope
Deliverables	Features/requirements
Freeze design as early as possible	Freeze design as late as possible
Low uncertainty	High uncertainty
Avoid change	Embrace change
Low customer interaction	High customer interaction
Conventional project teams	Self-organized project teams

Prototyping

Build it, assess it, modify it, repeat

1. Continuous integration, verification, and validation of the product.
2. Frequent demonstration of progress to increase the likelihood that the end product will satisfy customer needs.
3. Early detection of defects and problems.

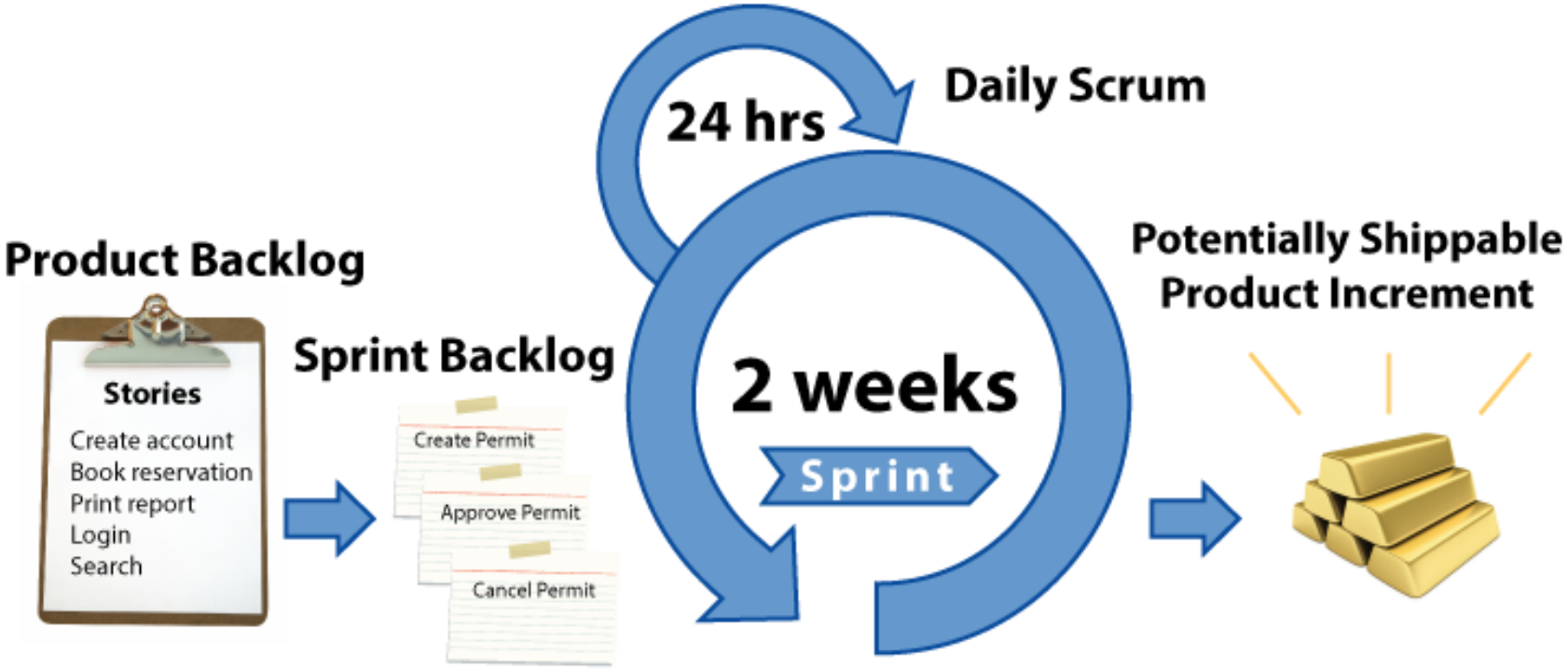
Agile Model

1. Focus on customer value
2. Iterative and incremental delivery
3. Experimentation and adaptation
4. Self-Organization
5. Continuous Improvement

The "Scrum" Method

- The product is composed of "**Features**" as described by one or more "**User Stories**"

  https://www.mountaingoatsoftware.com/agile/user-stories
- A feature delivers **functionality** to the customer according to the function described in a user story
- Features and user stories are **prioritized** by their perceived highest value
- Team tackles the highest priorities first
- Priorities of "feature backlog" are re-evaluated after each iteration
- Each "**sprint**" produces fully functional features
- Each feature involves analysis, design, build and test


Product Backlog
Stories
Create account
Book reservation
Print report
Login
Search
Sprint Backlog
Create Permit
Approve Permit
Cancel Permit
24 hrs
Daily Scrum
2 weeks
Sprint
Potentially Shippable
Product Increment

Roles

- Product Owner
  - Acts on behalf of the customer
  - Keeper of the feature **backlog**
  - Keeps the team focused on **priority objectives**
  - Final authority on requirements
  - Decides when a feature is complete

Roles

- Team
  - Builds the feature
  - 5–9 people
  - Self-organizing
  - No roles or titles
  - Input to sprint planning

Roles

- Scrum Master
  - Facilitator
  - Resolves issues
  - Keeps team on process
  - More “coach” than “manager”

Meetings

- Built around Sprints
  - A sprint runs a maximum of 4 weeks
  - A sprint team has 5-9 people
- **Daily Scrum**
  - Same time & place,15 minutes max
  - All Stand
  - Tasks & Blocks on white board
  - Questions
    - What have you done since the last scrum?
    - What will you do before the next scrum?
    - What is blocking you?

Scrum Master

- Does NOT assign daily tasks to team
- The team decides amongst themselves
- Master of the process, not of the team
- At the end of a sprint, team reviews the product/feature
- At the end of a sprint, team reviews the process in a "**retrospective**"

Product Backlog

- Prioritized list of features/requirements
- A prioritized list of work to be accomplished
- Owned by the product owner

Sprint backlog

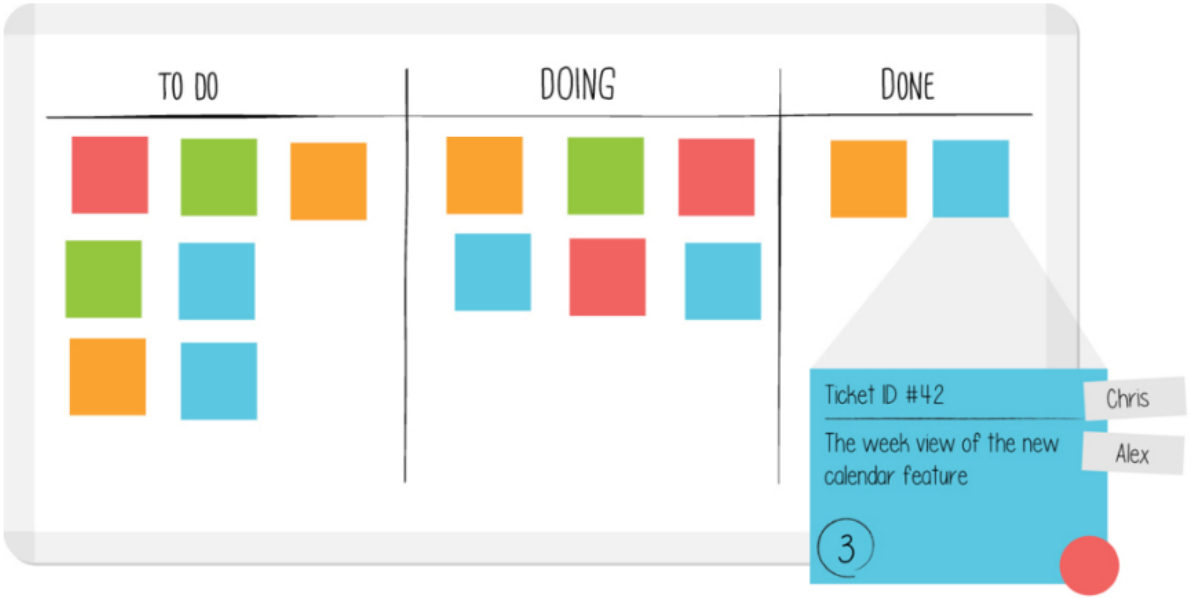
- List of individual sprints
- Tied to product backlog
- Owned by the team

Resistance to Agile

- Agile PM does not satisfy top management's desire for controlling budget, scope and schedule
- Many of the agile principles -- including self-organizing and intense collaboration – may run counter to corporate culture
- Requires a knowledgeable Scrum Master to keep teams focused on proper methods

Kanban

- A method for managing work
- Visual, on a board
- Tracking work
  - Planned
  - In-Progress
  - Complete



A physical Kanban board with a basic, three-step workflow

“The Kanban technique emerged in the late 1940s as Toyota's re-imagined approach to manufacturing and engineering.

Line-workers displayed colored kanbans — actual cards — to notify their downstream counterparts that demand existed for parts and assembly work. (Kanban is the Japanese word for "visual signal" or "card.")

The system's highly visual nature allowed teams to communicate more easily on what work needed to be done and when.

It also standardized cues and refined processes, which helped to reduce waste and maximize value.”

https://leankit.com/learn/kanban/kanban-board/

Scrum & Kanban

- Both embrace principles of agile development
- Both encourage
  - early and frequent delivery
  - self-organized teams
  - continuous improvement
  - high quality
  - prioritizing of requirements based on business value

Project Management Tools

Why a PM Tool?

- Document “deliverables” (a.k.a. features, requirements)
- Capture the WBS as features/tasks
- Plan the work
- Schedule the work
- Assign work to teams or individuals
- Plan sprints/work packets
- Track and report progress toward completion

Are you Agile? Your tool must

  - support people over process
  - facilitate collaboration

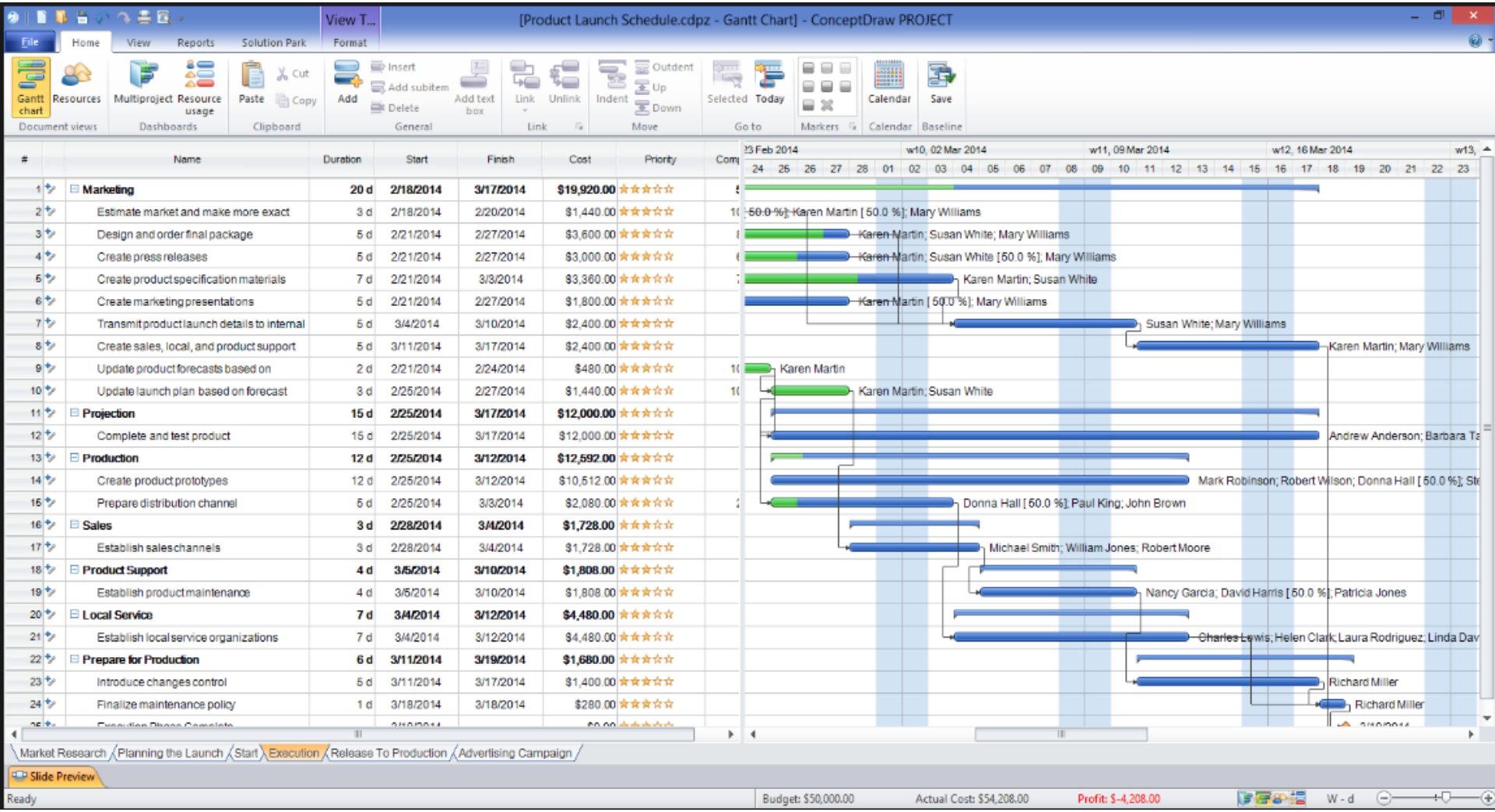
Creating a Project Plan

- GANTT chart
  - WBS
  - Predecessors/Successors
  - Resource Assignments
  - Duration
  - Calendar

Creating a GANTT chart

Henry Gantt

Project Management Tools



Project Management Tools

Some PM tools to look at:

- Asana
- Freedcamp
- Trello
- Wrike

(there are many more, but most cost $$ after a 30-day free trial)

Project Management Tools

Going Deeper:

Kanban overview:
https://www.youtube.com/watch?v=jf0tlbt9lx0

https://www.youtube.com/watch?v=N3BoLRVXoI0

Scrum overview:

https://www.youtube.com/watch?v=9TycLR0TqFA