SCRUM

Module 2: The Framework

Overview

- This module explores the Scrum framework
  - Roles, artifacts, ceremonies and artifacts
  - The role that each of these plays in a development process
- The strategic and tactical goals of Scrum
  - Effectiveness: Making better software (we define what “better” means)
  - Efficiency: Optimal use of resources while creating software
- Using Scrum to improve the organization’s software development process

Class Discussions

- From last week
  - I want you to start providing me with inputs to narrow the class cone of uncertainty
  - The rest of the the course should focus on high value content and activities
  - Over the next several weeks, we will be doing a deep dive into Scrum
  - But we also want to focus on how you are using Scrum, make it relate to your projects
- To do this, I need to know something about:
  - Specific issues you may be having with Scrum
  - Any places we need to review how Scrum is being applied.
  - Where we need to focus our efforts in class

Observations

Software is not released, it is allowed to escape.

Anonymous

If you can't describe what you are doing as a process,
you don't know what you're doing

W. Edwards Demming

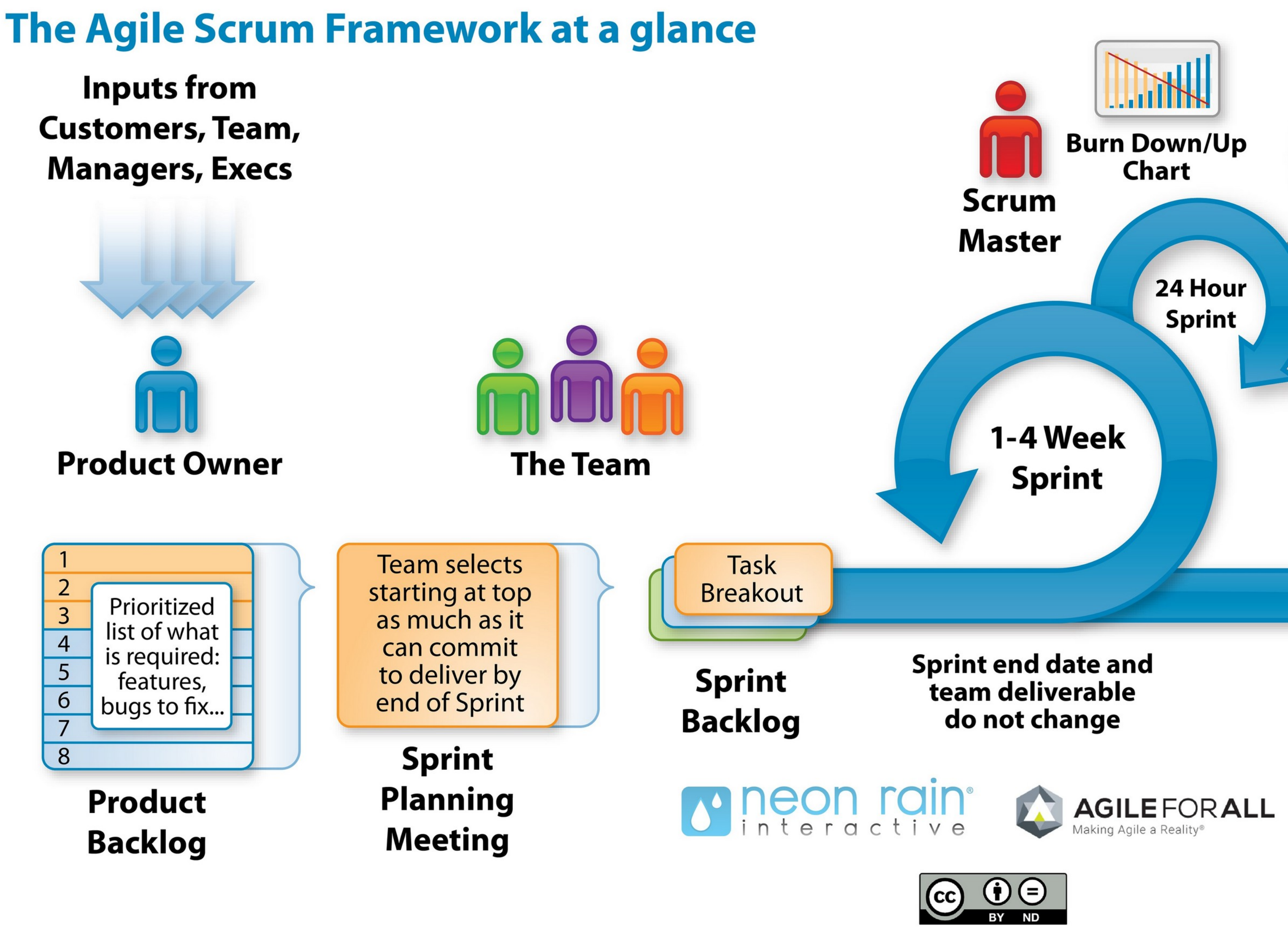
It is not a question of how well each process works,
the question is how well they all work together.

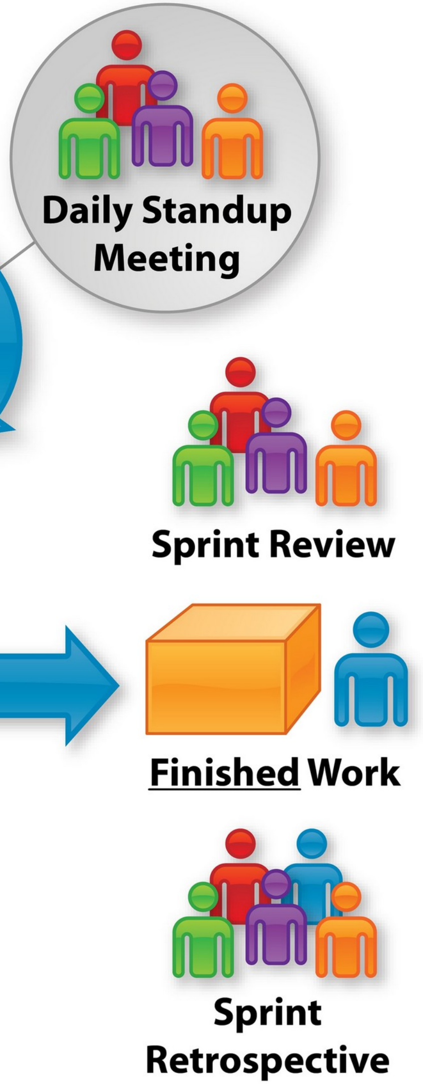
Lloyd Dobens and Clare CrawfordMason

All human error is impatience, a premature renunciation of method.

Franz Kafka

The Scrum Framework


The Agile Scrum Framework at a glance
Inputs from Customers, Team, Managers, Execs
Product Owner
The Team
1
2
3
4
5
6
7
8
Prioritized list of what is required: features, bugs to fix...
Product Backlog
Team selects starting at top as much as it can commit to deliver by end of Sprint
Sprint Planning Meeting
Task Breakout
Sprint Backlog
Scrum Master
Burn Down/Up Chart
24 Hour Sprint
1-4 Week Sprint
Sprint end date and team deliverable do not change
neon rain interactive
AGILE FOR ALL
Making Agile a Reality
cc BY ND



Standard Roles First Look

- Product Owner
  - Represents the customer and business perspective,
  - Explains how the completed Product Backlog items met or did not meet the requirements
  - Plans future priorities from the customer or business perspective
- Stakeholders
  - These may include customers, investors, and other departments affected by the product.
  - Their feedback is essential to help the Product Owner guide the product's direction.
  - We will discuss the stakeholder interaction with Product Owner in the deep dive on the product Owner role

Standard Roles First Look

- The PO is the primary point of contact between the Scrum team and the stakeholders
  - Defines the project vision from the perspective of the stakeholders
  - Responsible for ensuring the expected value is delivered by the product
  - Specifically, aligns with the project vision and stakeholder acceptance criteria
  - Should not be the Scrum Master or a member of the Development Team
- Most Agile methodologies emphasize face to face meetings with customer
  - One of the problems is identifying exactly what should be done in those meetings
  - This often requires BA type skills that developers don’t have
- The PO can be though of as in interface between the stakeholders and development team
  - The deep dive into the PO in following weeks will explore exactly how this is done.

Standard Roles First Look

- Scrum Master
  - Facilitator, mentor, and coach for the Development Team and PO.
  - Lead, facilitate, and support the entire team.
  - Primary objective is to build the team and ensure that it follows processes effectively
  - Goal is helping the team to become more efficient and effective
  - The SM is not a boss or supervisor
  - The SM is a change agent whose focus is on the team's efficiency and effectiveness
  - Facilitates the Scrum events
- The SM is generally proficient in a number of areas
  - Project management, various soft skills, product or industry knowledge
  - In a future class, we will do a deep dive into the SM role

Standard Roles First Look

- The Development Team
  - Diverse, cross-functional, self-organizing, and empowered group
  - Responsible for delivering the product or finished product increments.
  - They are guided and mentored by the SM and oriented by the Product Owner.
- The DT does not have traditional hierarchical roles
  - There are no defined team roles like “developer” or “QA tester” or “UX engineer”
  - There are no reporting roles like “team lead”
  - Allows teams to self organize depending on the a Sprint’s specific requirements
- A cross-functional team
  - Does have different skill sets like UX engineer, tester, data engineer, developer, etc
  - But these skill sets are not codified into formal roles.

Standard Roles First Look

- A cross-functional team (cont.)
  - There is no formal team lead but different members may take a lead role depending on the work being done in a Sprint.
  - For example, during a Sprint is focusing on data models and persistence, the DB engineer may take a leaderships role for that development period.
  - This is the whole idea of self organizing teams – with the SM providing guidance and feedback as to how effective their collaboration currently is.
- One of the Scrum ideas mentioned in the last class is
  - The SECI model of knowledge dimensions
  - Used to externalize knowledge for process improvement
  - And to support design thinking
  - We will cover these in more depth in the deep dive into DT processes and dynamics

Standard Artifacts First Look

- Product Backlog
  - An ordered list of everything required to develop a product, project, or service.
  - Not only includes functionality but can also include
    - *prototypes,*
    - *maintenance tasks,*
    - *marketing plans,*
    - *bugs or issues, designs,*
    - *legal planning,*
    - *components,*
    - *Integration builds, and more.*
- Owned by the PO
  - Since it contains everything to be done necessary to deliver a product to spec
  - Single source of truth as to requirements, changes and acceptance criteria

Standard Artifacts First Look

- Product Backlog
  - Contains a variety of items that may include features, functions, requirements, documentation, improvements, and fixes needed to deliver a successful product.
  - Clarifies and quantifies to do items
  - Typically includes
    - ***Description***: *A brief explanation of what needs to be done.*
    - ***Value***: *The value that the item brings to the business or user.*
    - ***Estimates***: *Work estimates, often using story points, to help with sprint planning.*
    - ***Acceptance Criteria:*** *Clear definitions of what it means for an item to be completed, ensuring that everyone on the team understands when the item is done.*
    - ***Stakeholder Feedback:*** *Input from stakeholders may lead to adjusting the priority of certain items to better align the product with market or user needs.*
- Creating the PB
  - Owned by the PO but developed in collaboration with others
  - PO ensures the inclusion of items on the PB and their ranking reflect the goals of the project

Standard Artifacts First Look

- Development Team and PB
  - The DT provides estimates of time, effort and resources required to complete the task
  - Provides best estimates of work required and costs
  - Requires a detailed enough specification of the item so that the estimate can be made.
    - *This may require additional input from the PO*
  - May make suggestions to modify the item to improve performance or efficiency
    - *Based on past experience and expertise with similar types of requested functionality or features*
- Stakeholders and the PB
  - Often provide insights and context that allow a better understanding what needs to be done
  - Often provides insights the PO cannot
    - *Regulatory issues, market challenges, industry or organizational context*
    - *Insight into key requirements, for example, the specific clinical requirements a doctor has for some feature on diagnostic software*

Standard Artifacts First Look

- PB prioritization
  - The objective is to produce a “feature baseline”
  - This is a list of items that meet the customer’s requirements
  - *And* has a best chance of success within the constraints of time and budget.
- Re-prioritization
  - Based on stakeholder feedback, changes may be made to the PB
  - This requires a re-prioritization of the PB items
  - Necessary since requirements will change or their value to the customer may change
  - Insights from the DT may also require re-prioritization
    - *Features that were originally thought to be complex or difficult may be easier or simpler*
    - *For example, part of the problem might solved with reusable components the DT did not know about initially*
    - *Or a feature may be much more difficult than originally estimated due uncovered technical issues*

Standard Artifacts First Look

- Sprint Backlog
  - The Sprint Backlog is a set of items from the PB selected for development and delivery during the sprint
  - Done during the Spring planning meeting
  - The items are chosen to produce a product increment, which is a measurable step towards the final product release.
    - *This often referred to as the Spring goal*
  - Each SB has an associated plan for delivering the product increment and achieving the sprint goal.
- Sprint Plan:
  - Specifies specific tasks, task assignments, and time estimates required to complete the selected PB items.
  - Can be thought of as a project plan for that SprintS

Standard Artifacts First Look

- Sprint Increment
  - Includes all product backlog items (PBIs) completed and tested during the sprint that meet the defined acceptance criteria.
  - Also integrates all features developed in previous Sprints that remain part of the product.
  - Ensures integration of the current increment with the newly added features, to avoid regression
- Sprint Activities
  - *Task selection:* During Spring planning, collaboratively selects PBIs which are believe can be completed,
  - *Development and Testing:* During the sprint, the team works on testing components to ensure they meet the defined acceptance criteria and do not compromise existing functionality.
  - *Continuous Integration:* Ideally, the team uses continuous integration practices to ensure that new features or components are regularly integrated with the existing product and thoroughly tested.

Standard Artifacts First Look

- Sprint QA
  - Intended to ensure Increment is potentially releasable,
- Definition of Done (DoD):
  - Before an Increment can be considered complete, it must meet the agreed Definition of Done by the team and the Product Owner.
  - This includes specific criteria or operational definition that each feature must meet to ensure quality.
  - Often expressed as a set of acceptance tests provided by the PO
- Automated and Manual Testing:
  - Both automated and manual testing during development helps identify and fix bugs before theyare fully integrated into the product.

Scrum Events First Look

- Sprint
  - Fixed period during which the team develops a specific set of features from the PB
  - Should result in a product increment.
  - Increment goal incorporates the concepts of
    - *MVP (Minimum Viable Product)*
    - *MMP (Minimum Marketable Product)*
- Sprint Length
  - Always as a fixed duration, generally ranging from one to four weeks. The duration should not be modified during the Sprint.
  - The most common duration is two weeks.
  - This consistency creates a predictable work rhythm known as cadence, helping the team to plan and deliver improvements continuously.

Scrum Events First Look

- Sprint Velocity
  - As a rule of thumb, Sprint goals should be attainable working 40 hours a week
  - If not, there is problem with the planning process
    - *Overestimating the capabilities of what the team can accomplish*
    - *Not allowing enough time to complete the work – the Sprint is too short*
    - *Poor task breakdown and work planning*
    - *Underestimating the complexity or difficulty of PBI*
    - *Incomplete specification as to what DoD should be*
- Sprint Length factors
  - More complex projects may require longer Sprints to adequately handle development and testing.
  - Products that need quick market or customer feedback may benefit from shorter Sprints.
  - The experience and efficiency of the DT based on historical data

Scrum Events First Look

- Sprint 0 or Sprint Init
  - Every project needs an initial proposal to be submitted, accepted by the internal or external client, and then the product development can begin.
  - This initial proposal is delivered in Sprint 0, also known as Sprint Init.
  - Duration of Sprint 0:
    - *The time the required to create a project or product proposal in Scrum, or to create some sort of product charter and define the scope of the project*
  - Who Participates?
    - *Everyone on the Scrum team as well as stakeholders who provide insight into project governance, portfolio management etc.*
  - Contains at a minimum
    - *Sprint length,*
    - *if there are MVPs, MMPs and their approximate costs,*
    - *estimated costs for materials and the project,*
    - *Scrum team composition and characteristics,*
    - *contract type, and billing method,*
    - *risk plan (if any), marketing (if any),*
    - *a Story Mapping or Product Backlog at a high level, commonly broken down into Themes.*

Scrum Events First Look

- Sprint Planning
  - Key ceremony in Scrum that marks the beginning of each Sprint.
  - During this meeting, the DT, PO and SM, plan the work to be done in the upcoming Sprint.
- Objectives of Sprint Planning
  - Define the Sprint goal: a short, clear description of what the team intends to achieve during the Sprint.
    - *This goal guides the team’s work and helps maintain focus on the important outcomes.*
  - Determine which PBIs will be addressed during the Sprint, based on the Product Owner's priority and the team's capacity.
  - Break down the selected PBIs items into smaller tasks and estimate the time and resources needed to complete them.

Scrum Events First Look

- The Daily Scrum or Daily Stand-Up
  - Objectives of the Daily Scrum
    - *Daily Synchronization: Helps keep the team aligned on the progress of the work during the Sprint.*
    - *Impediment Identification: Provides a platform for team members to highlight and discuss any obstacles that could hinder their progress.*
    - *Fostering Self-Management: Encourages the team to manage their own performance and adapt to challenges in real time.*
  - Participants
    - *Development Team: The main participants in the Daily Scrum. Each member shares their updates and discusses their plans for the day.*
    - *Scrum Master: Facilitates the meeting to ensure it stays within the allocated time and achieves its purpose. Although the Scrum Master participates in the meeting, they should not lead it.*
    - *Product Owner: May attend but is not an active participant unless quick clarifications about PBIa are needed.*
  - Importance of the Daily Scrum
    - *Transparency and Communication: Promotes transparency among team members and facilitates daily communication, which is essential for quickly identifying and resolving issues.*
    - *Adaptability and Flexibility: Allows the team to make daily adjustments to the work plan to respond effectively to changes or challenges that arise.*
    - *Commitment and Responsibility: Fosters a sense of responsibility and personal commitment, as each team member reports their progress and plans to the rest of the team.*
    - *Purpose: Enable the Development Team to synchronize activities and create a plan for the next 24 hours.*
    - *Duration: No more than 15 minutes, regardless of Sprint length.*

Scrum Events First Look

- Sprint Review
  - Held at the end of each Sprint.
  - Main purpose is to inspect the product increment and adapt the PB if necessary, based on feedback received.
  - Opportunity for the Scrum Team to present their work to stakeholders and gather their feedback for future development.
- Objectives of the Sprint Review
  - Evaluate the Product Increment: Review and demonstrate the features or increment developed during the Sprint to ensure they meet the acceptance criteria and the expectations of stakeholders.
  - Gather Feedback: Collect direct feedback from stakeholders to inform the next steps in product development.
  - Adapt the Product Backlog: Based on feedback and discussions during the Sprint Review, the PB may be adjusted to reflect new priorities, remove obsolete items, or add new requirements.

Scrum Events First Look

- Importance of the Sprint Review
  - Ensures that the product development aligns with the needs and expectations of users and the business.
  - Provides a regular opportunity to adjust the product's development direction, which is crucial in changing environments.
  - Fosters transparency and collaboration between the development team and stakeholders, strengthening trust and mutual understanding.
- We will explore the Sprint review in more detail in an upcoming class

Scrum Events First Look

- Sprint Retrospective
  - Unlike the review, focuses on the work done and the team performance
  - Objectives
    - *Analyze what went well and what could be improved in terms of processes, tools, and relationships within the team.*
    - *Identify concrete actions to improve the team’s efficiency, effectiveness, and work quality in future Sprints.*
    - *Create a safe space for the team to discuss and commit to continuous improvement, promoting self-management and responsibility.*
  - Meeting Structure: usually divided into three parts:
    - *What went well and can be improved? Identify the successes and strengths of the last Sprint.*
    - *What didn’t go well and should be improved? Discuss challenges and areas where the team encountered problems.*
    - *What was not done and should be implemented? Propose improvements in general.*
  - Importance of the Sprint Retrospective
    - *By regularly reflecting and adapting their processes, the team can continuously improve their efficiency and effectiveness.*
    - *Fosters a sense of ownership and commitment among team members, as they have the opportunity to actively contribute to how they work.*
    - *Provides a regular forum for addressing problems before they become significant obstacles.*

Scrum Events First Look

- Sprint Retrospective
  - Unlike the review, focuses on the work done and the team performance
  - Objectives
    - *Analyze what went well and what could be improved in terms of processes, tools, and relationships within the team.*
    - *Identify concrete actions to improve the team's efficiency, effectiveness, and work quality in future Sprints.*
    - *Create a safe space for the team to discuss and commit to continuous improvement, promoting self-management and responsibility.*
  - Meeting Structure: usually divided into three parts:
    - *What went well and can be improved? Identify the successes and strengths of the last Sprint.*
    - *What didn't go well and should be improved? Discuss challenges and areas where the team encountered problems.*
    - *What was not done and should be implemented? Propose improvements in general.*
  - Importance of the Sprint Retrospective
    - *By regularly reflecting and adapting their processes, the team can continuously improve their efficiency and effectiveness.*
    - *Fosters a sense of ownership and commitment among team members, as they have the opportunity to actively contribute to how they work.*
    - *Provides a regular forum for addressing problems before they become significant obstacles.*

Scrum “Values”

- Problems with how development processes are often implemented
- They are poorly defined.
  - Much of the work is ad hoc, meaning actual work technique are invented as needed, but then forgotten about once the need has passed
  - Creates “critical man” situation which risks loss of knowledge
    - *“The last time we had this database problem, Jose came up with a fix that worked, but he’s left the company to become a drummer in a rock band so we have no idea how to duplicate what he did without reverse engineering his work.”*
  - The same problem encountered in different project is handled from scratch
    - *There is no ability to apply lessons learned in previous encounters*
- No ability to improve the efficiency of the work processes
  - Metrics on tasks are often no collected or collected inconsistently
  - Even when collected, they are not used to make the processes more efficient

Scrum “Values”

- Best practices don’t exist at a team level
  - The team doesn’t get better although some individuals might
  - Back to the critical man dilemma
- Process descriptions are problematic
  - They may be out of date, unusable or not applicable to the current project
  - Often reflect governance and business rules rather than actual work processes
    - *“All requirements with cost estimates must be signed off by the Accounting Department before work can begin.”*
  - Eventually the process descriptions become irrelevant
    - *“Following official procedures won’t get the job done.”*
- These relate to notions of efficiency and effectiveness

Scrum “Values”

- Best practices don’t exist at a team level
  - The team doesn’t get better although some individuals might
  - Back to the critical man dilemma
- Process descriptions are problematic
  - They may be out of date, unusable or not applicable to the current project
  - Often reflect governance and business rules rather than actual work processes
    - *“All requirements with cost estimates must be signed off by the Accounting Department before work can begin.”*
  - Eventually the process descriptions become irrelevant
    - *“Following official procedures won’t get the job done.”*
- These relate to notions of efficiency and effectiveness
  - The scum values are about being able to manage and improve processes
  - We will explore this topic later when we look at how to improve Scrum

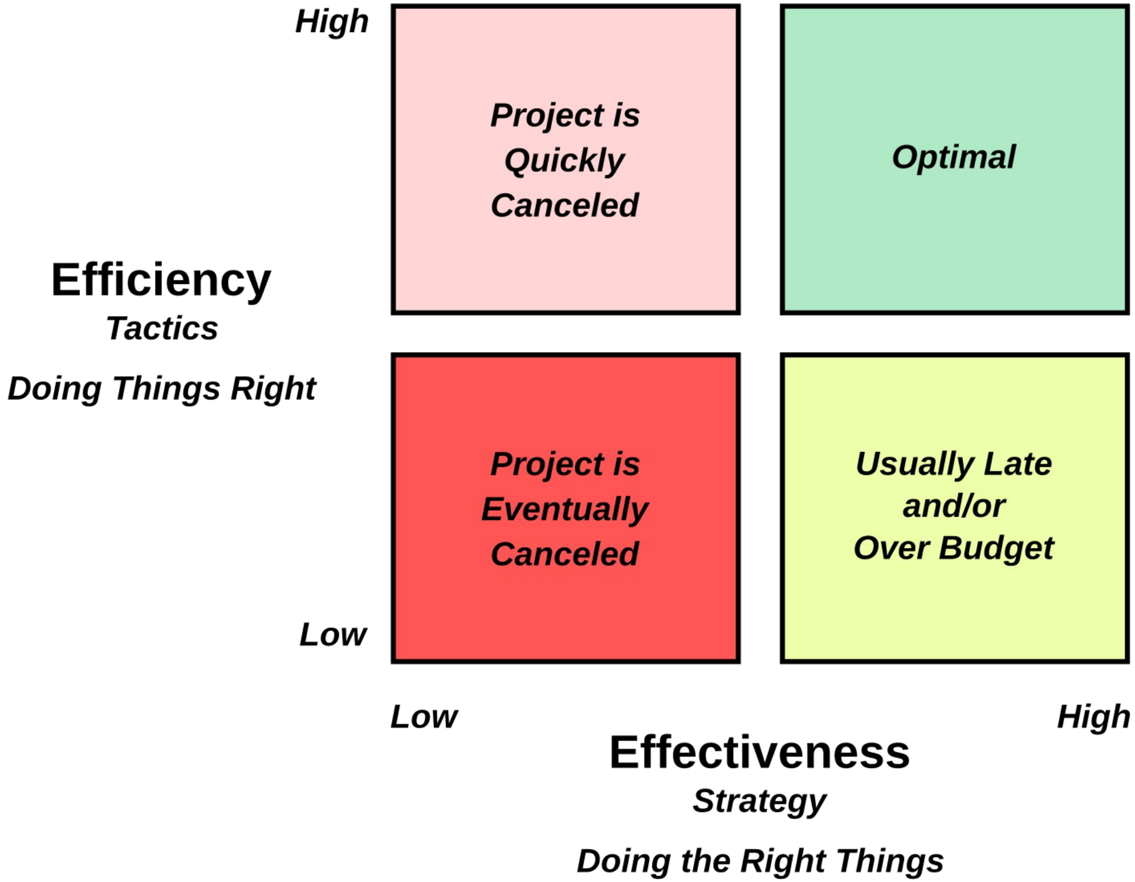
Efficiency and Effectiveness

Effectiveness:

- A process is effective when it produces the right result.
- We often call this delivering to spec
  - Delivering software that meets the stakeholder needs and solves the business problems as it was supposed to.
- Effective software development processes build the right software.

Efficiency:

- A process is efficient when it is optimal in terms of how development resources are used.
- One of the main symptoms of a process not being efficient is how much rework is done during development.
  - Rework is where previous work has to be done over because of errors made in the previous work.
- Efficient software development processes build software right.



Efficiency

Tactics

Doing Things Right

High

Low

Project is Quickly Canceled

Optimal

Project is Eventually Canceled

Usually Late and/or Over Budget

Low

High

Effectiveness

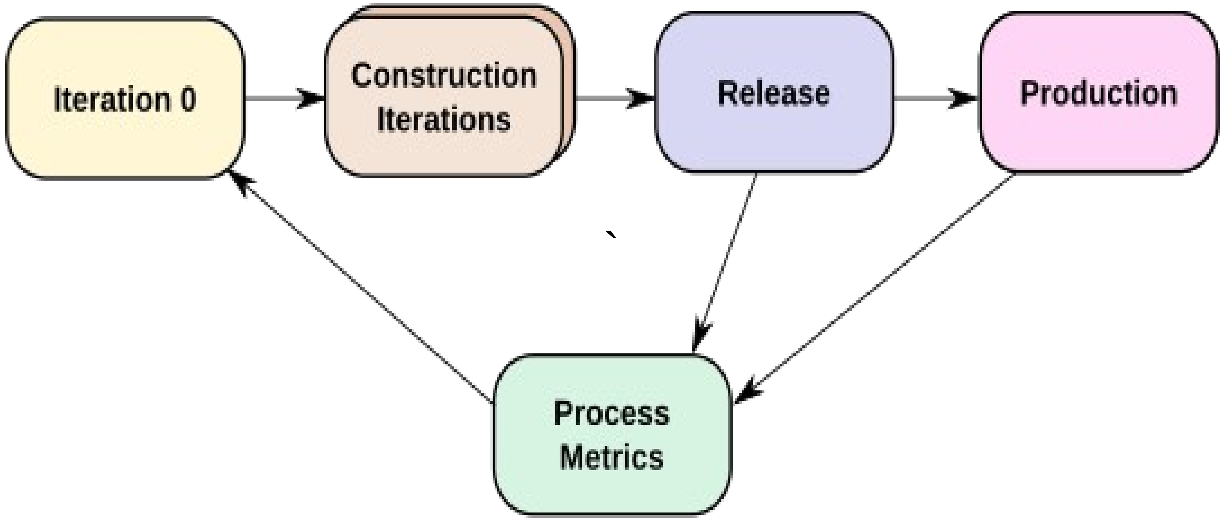
Strategy

Doing the Right Things

Efficiency and Effectiveness

- Effectiveness
  - Guided by the Product Owner with continuous feedback from the stakeholders
  - Guides the development of the Product Backlog and the Sprint Backlog
  - Relies heavily on the QA and acceptance testing done during the project and feedback from the Sprint Review
- Efficiency
  - Primary concern of the Development Team and Scrum Master
  - Reviews the work done and looks for ways to improve what they are doing
  - Relies heavily on the Sprint Retrospective
  - There may also be a Project Retrospective if the team was using a process defined by the organization

Project Retrospective



- The iteration 0 is the Scrum Init
- The Release stage is where the project retrospective is done
  - Incorporates all of the lessons learned from the Sprint Retrospectives
  - As well as any relevant feedback from the production environment
- These lessons learned are then incorporated into the Scrum Init of the next project to make the next project more efficient

Process Maturity

- Software Engineering Institute
  - Commissioned by US Government to find out how organizations built quality software.
  - They found it depended on three factors:
    - *They had a defined software production process.*
    - *They followed the process when they build software*
    - *They were always improving the process in terms of efficiency and effectiveness.*
  - The measure of how much the organization did this was process maturity

Immature Organizations

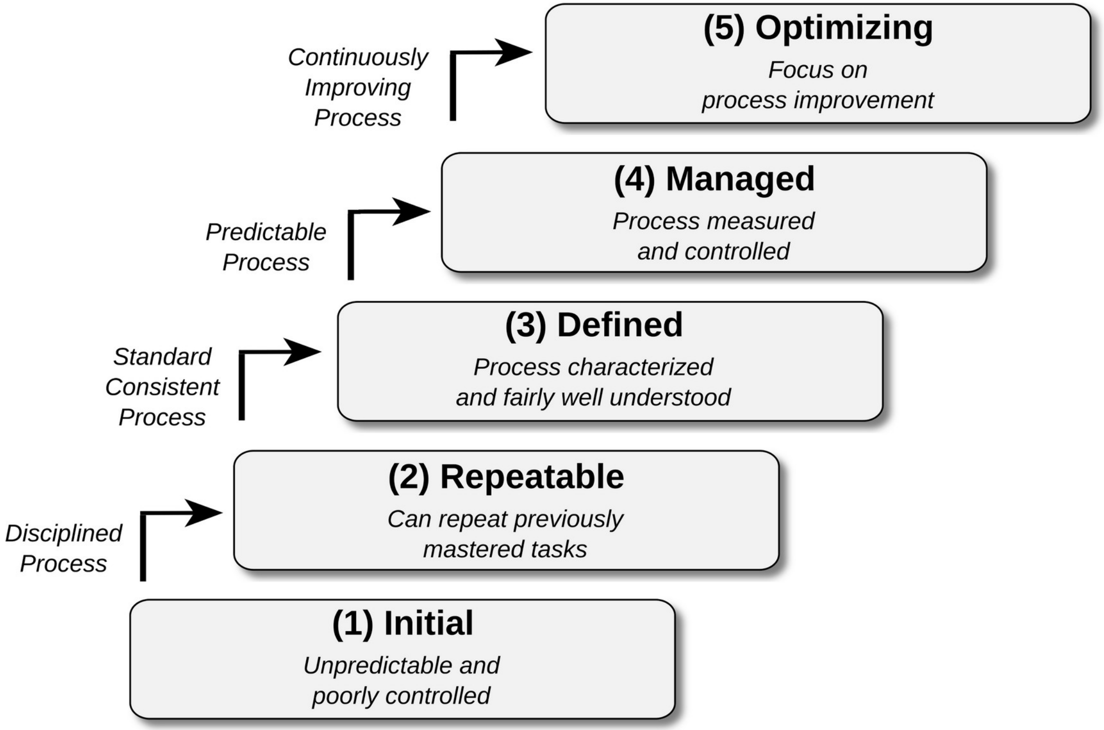
- The process is often improvised by the developers and management during the course of the project.
- Defined processes are often not followed or enforced.
- Project activities are reactionary and often focus on solving immediate crises.
- Schedules and budgets are based on unrealistic estimates and routinely exceeded.
- Bringing a project in line with deadlines or costs causes functionally and quality of the product to suffer.
- Product quality is difficult to predict and there are no objective standards to assess product quality.
- There are no standard ways for solving problems, everything is done on an “fire-fighting” basis and the same problems keep recurring with the same negative impacts.
- The defined process is not realistic and the organization is characterized by the attitude “Following the process won't get the job done.”
- The defined process is often created in part by those who are not part of the development staff and tends to be more about reporting and management activities than work tasks.

Mature Organizations

- The software process is standardized across the organization and communicated to all existing staff and new employees.
- All work activities are carried out according to the planned process.
- The processes are usable and realistic because they describe how work actually gets done.
- Defined processes are updated as necessary.
- Improvements to the process are planned and evaluated through pilot tests or cost benefit analyses.
- Responsibilities are defined clearly in both the organization and in each project.
- Management continuously monitors the quality of the products and the processes using objective and quantifiable criteria.
- There are defined processes for analyzing and solving problems with the product and process.
- Budgets and schedules are realistic because they are based on historical performance data.
- The infrastructure of the organization supports the process.
- A disciplined process is consistently followed because all of the participants understand the value of doing so and enforce following the process at a peer level.

The Levels

- **Initial**: Basically describes a start-up company – no defined processes
- **Repeatable**: Production process mastered – all projects are on spec, on time and to budget.
- **Encultured**: The process is part of the culture, it's "the way do it here" and is enforced by everyone in the organization.
- **Proactive**: A review based culture – everything is reviewed to ensure it is in alignment with best practices and quality metrics.
- **Integrated**: The processes have become self correcting and prevent non-optimal actions.



Continuously Improving Process
(5) Optimizing
Focus on process improvement
Predictable Process
(4) Managed
Process measured and controlled
Standard Consistent Process
(3) Defined
Process characterized and fairly well understood
Disciplined Process
(2) Repeatable
Can repeat previously mastered tasks
(1) Initial
Unpredictable and poorly controlled

Questions