Methodologies for Project Management and Software Development

Lecture 3 by Professor Vladimir Geroimenko

Module "Software Project Management"

18 October 2023 - Teaching Week 4

Textbook reference: Chapter 4 and Appendix A

Lecture Outline

Part 1 : PM methodologies vs. SD methodologies

Part 2: Project Management (PM) methodologies:

- PMBOK (pronounced *pimbok*)
- PRINCE2
- STEP WISE (already covered in Lecture 2)

Part 3: Software Development (SD) methodologies:

- Structured Methodologies
  - Waterfall
  - V-process
- RAD Methodologies
  - Incremental delivery
  - Evolutionary development
- Agile Methodologies
  - DSDM
  - XP
  - Scrum

Part 1:

PM Methodologies
vs.
SD Methodologies

What is a Methodology?

- A **methodology** is a collection of methods.

- A **method** is a general way of carrying out a specific task that could be applicable to any project needing to do that task.

- A **model** is a representation of a system which implements a particular method or methodology and abstracts certain features but ignores others.

- A **technique** is a more specific approach that tend to involve the application of scientific, mathematical or logical principals to resolve a particular kind of problem and that often requires the practice of particular skills (e.g. software design).

Project Management Methodologies

- Provide a high-level project framework, such as:

    - project initiation, prioritization, project planning,

        status reporting, issue/decision/change management,

        quality management, risk management, etc.

- Managing projects is a process that should be

    performed consistently regardless of the SDLC

    used.

Software Development Methodologies

- Provides the detail on system design and development.

- Address specific steps, processes and deliverables associated with the type of projects being implemented.

- For example, a traditional systems project might utilize a Waterfall approach; a prototyping project might utilize a Rapid Applications Development (RAD) approach; a web-based application might use the Agile approach.

Part 2:

Project Management Methodologies

Most Known SPM Methodologies

- PMBOK: Project Management Body Of Knowledge

- PRINCE2: PRojects IN Controlled Environments

- SPMM: Stepwise Project Management Method (Described in detail in Lecture 2)

PMBOK: Project Management Body of Knowledge – 1 of 2

- https://www.pmi.org/pmbok-guide-standards

- As an internationally recognised standard (IEEE Std 1490-2003)

- The PM methodology for **North American** projects

- Five distinct process groups: **initiating, planning, executing, controlling, closing**

- Project knowledge areas: **integration management, scope management, time management, cost management, quality management.**

Knowledge Areas / Process Groups	Initiating	Planning	Executing	Controlling	Closing
Project Integration Management		Project Plan Development	Project Plan Execution	Integrated Change Control	
Project Scope Mangament	Initiation Scope Definition	Scope Planning	Scope Change Control	Scope Verification	
Project Time Management		Activity Definition Activity Sequencing Activity Duration Estimating Schedule Development		Schedule Control	
Project Cost Management		Resource Planning Cost Estimating Cost Budgeting		Cost Control	
Project Quality Management		Quality Planning	Quality Assurance	Quality Control	
Project Human Resource Management		Organization Planning Staff Acquisition	Team Development		
Project Communications Management		Communications Planning	Information Distribution	Performance Reporting	Administrative Closure
Risk Project Management		Risk Management Planning Risk Identification Qualitative Risk Analysis Quantitative Risk Analysis Risk Response Planning		Risk Monitoring and Control	
Project Procurement Management		Procurement Planning Solicitation Planning	Solicitation Source Selection Contract Administration		Contract Closeout

Table 1: Mapping of Project Management Processes to Process Groups and Knowledge Areas. [PMBOK-PMI].

PRINCE2:
PRojects IN Controlled Environments

- https://www.prince2.com/uk/what-is-prince2

- The **UK standard** for project management developed

- Sponsored by the Government and used in both the public and private sectors

- Focuses more on procedural at the expense of techniques

- Divide projects into several manageable Stages (subset of project activities)

- Please note: PRINCE2 is described in Appendix A of our textbook

PRINCE2
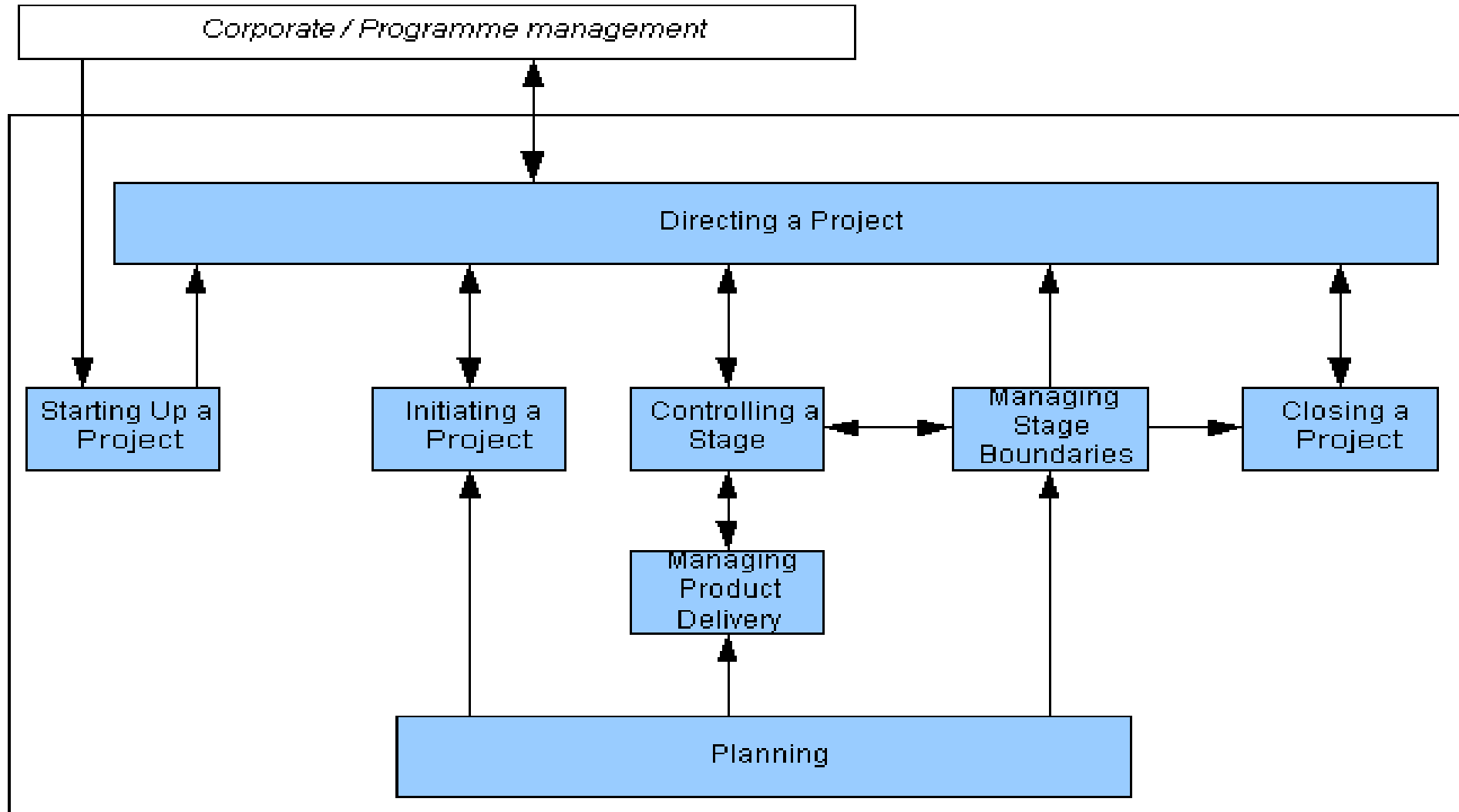
1. Starting up a Project (SP)

- To make sure that the project has a very clear beginning, this process occurs even **before** the project has actually started.

- All decision-making bodies discuss how the project is to be carried out.

- All this information will be put together in a **Project Brief.**

2. Initiating a Project (IP)

- Carefully planning for the project to ensure that it meets its objectives.

- Detailed estimations of costs, needed time and other resources have to be made.

- Generate the **Project Initiation Document (PID)** for approval by the **Project Board (PB).**

3. Controlling a Stage

- One of the advantages of PRINCE2 is that divided a project into manageable **Stages**.

- Each project stage must be **completed before** the next stage can be started.

- Each new stage is **planned in the stage** proceeding it.

- **Stage Plans** will be approved by the PB to help ensure that the project remains within budget and delivers its objectives.

4. Managing Stage Boundaries

- Preparing for the next stage and reviewing the current stage.

- The Project Manager makes suggestions to the PB about the project achievements and any changes in the business case, project plan, risks and issues.

5. Managing Product Delivery

- The goal of a PRINCE2 project is to deliver **products**.

- Everything produced in PRINCE2, even a document, is called **a product.**

- It is the Project Manager's responsibility to ensure that the 'supplier' produces the correct products at the right time.

6. Closing a Project

- At the end of the project, after its products have been delivered, the project is closed down with approval of the PB.

- The Project Manager plans the **Post Project Review** (PPR).

- Any lessons learned are recorded, resources are released and the **Post Project Review Plan** (PPRP) is created.

PRINCE2 Documents

- PRINCE2 maintains several documents throughout the projects, which may be formal paper documents, word processor files, data in specialised PRINCE2 software, or informal notes by managers.

- "Registers" are intended to be more formal than "logs".

PRINCE2 Documents

- **Project Brief**
  - A short explanation of the need for the project, the management team, the structure and goals.

- **Business Case**
  - A detailed description of the need for the project and its expected benefits (to all stakeholders in the project, including its owner, users and suppliers and sometimes the general public)

- **Risk register**

- **Quality register**

- **Issues register**
  - A set of notes about problems, complaints and concerns sent by all project members.

- **Lessons log**
  - A set of notes of lessons learned (often the hard way) which may be useful to future projects.

- **Daily log**
  - A diary about the project written by the project manager, like a ship's log.

Planning

Each project plan, stage plan and team plan must consider key planning aspects:

- What products to produce
- The activities required to produce these products
- Estimated resources
- Scheduling the activities
- Analyzing risks

Directing a Project

- Approving the **Project Brief** by the **Project Board (PB).**
- During the rest of the project this PB has the overall responsibility for the success of the project.
- The **Project Manager** has the day-to-day responsibility.
- The Project Manager will inform the PB about the project's progress.

PRINCE2 Advantages

- Providing a standard approach for project management.

- Widely recognized and understood.

- PRINCE2 enables projects to have:
  - A controlled and organized start, middle and end
  - Regular reviews and control of progress against plan
  - Flexible decision points
  - Good communication channels between the project board, project management, and the rest of the organization

PRINCE2 Disadvantages

- Every person who works on a PRINCE2 project should be quite **familiar with every aspect** of PRINCE2 to know how to play the game

- Using PRINCE2 means that **a lot of documents** and lists have to be written and approved by both Project Manager and Project Board

- **Splitting up** a PRINCE2 project often results in a lack of knowledge of the project by responsible persons like the Project Manager

- Take into consideration what is **popular in your geographic area**.

- Many project management methodologies address the management of a single project, without appreciating that **many other projects** in a company compete for the very same resources and attention.

- **Consistent** project management methodology is mandatory for effective reporting and analysis.

How To Choose a SPM Methodology? 2 of 2

- There is **no one-size-fits-all** methodology

- You **cannot** simply use a methodology exactly as it stands.

- **You needed to modify** and tailor whichever methodology you select to suit your own project needs.

Part 3:

Software Development Methodologies

Software / Systems Development Methodologies

I. Structured Methodologies

- Waterfall
- V-process

II. RAD Methodologies

- Incremental delivery
- Evolutionary development
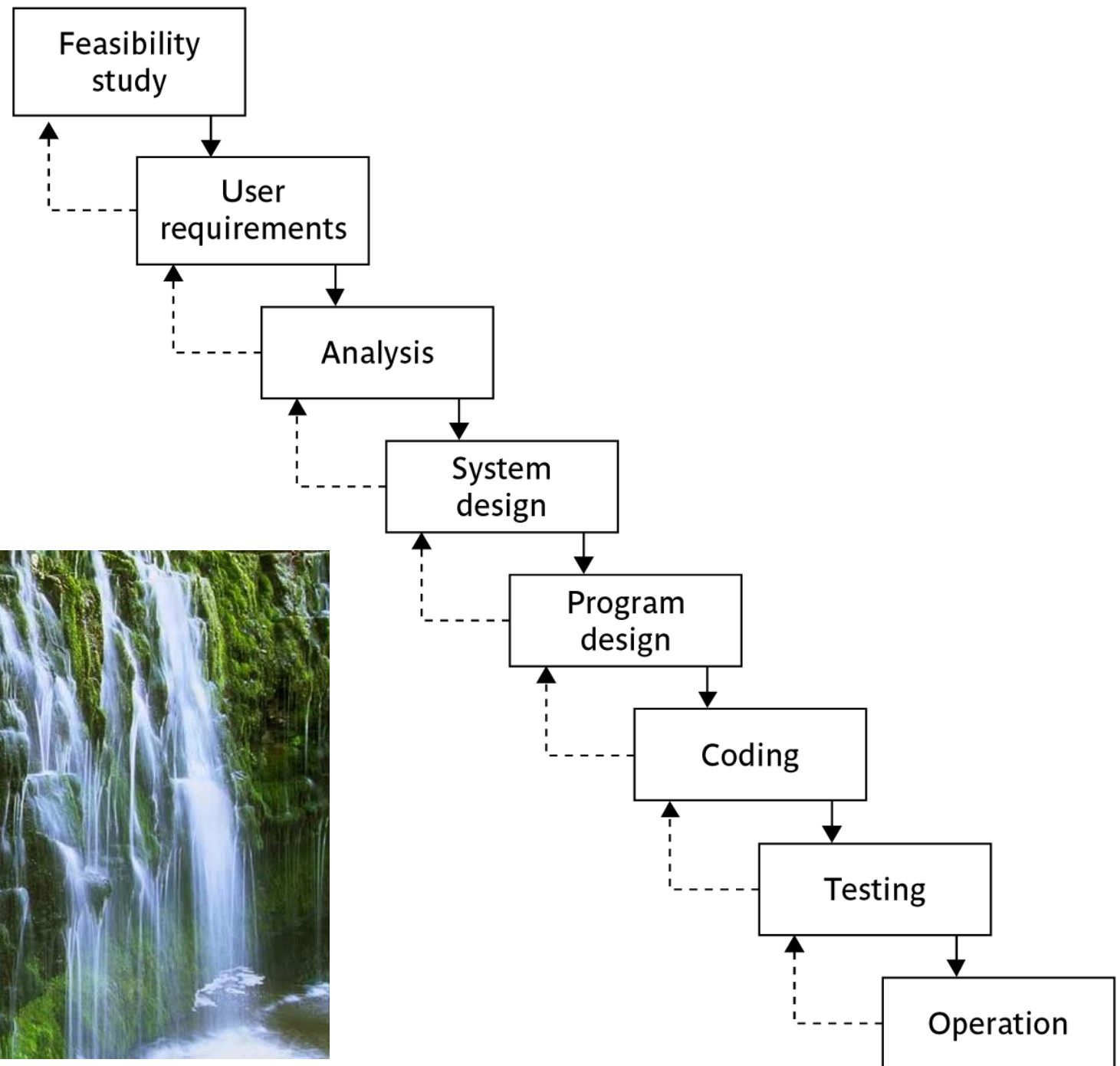
III. Agile Methodologies

- DSDM
- XP
- Scrum

I. Structured Methodologies

- Structured design methodologies adopt a formal step-by-step approach to the SDLC (Software Development Life Cycle) that moves logically from one phase to the next.

The Waterfall Model – 1 of 4

The Waterfall Model – 2 of 4

With waterfall-based methodologies, the analysts and users proceed sequentially from one phase to the next.

The two key advantages of waterfall-based methodologies are:

1. The system requirements are identified long before programming begins.

2. Changes to the requirements are minimized as the project proceeds.

The Waterfall Model – 3 of 4

The key disadvantages of waterfall development-based methodologies are:

- The design must be **completely specified** before programming begins.

- A **long time** elapses between the completion of the system proposal in the analysis phase and the delivery of the system.

- Limited scope of **iteration.**

The Waterfall Model – 4 of 4

- The 'classical' model
- Imposes structure on the project
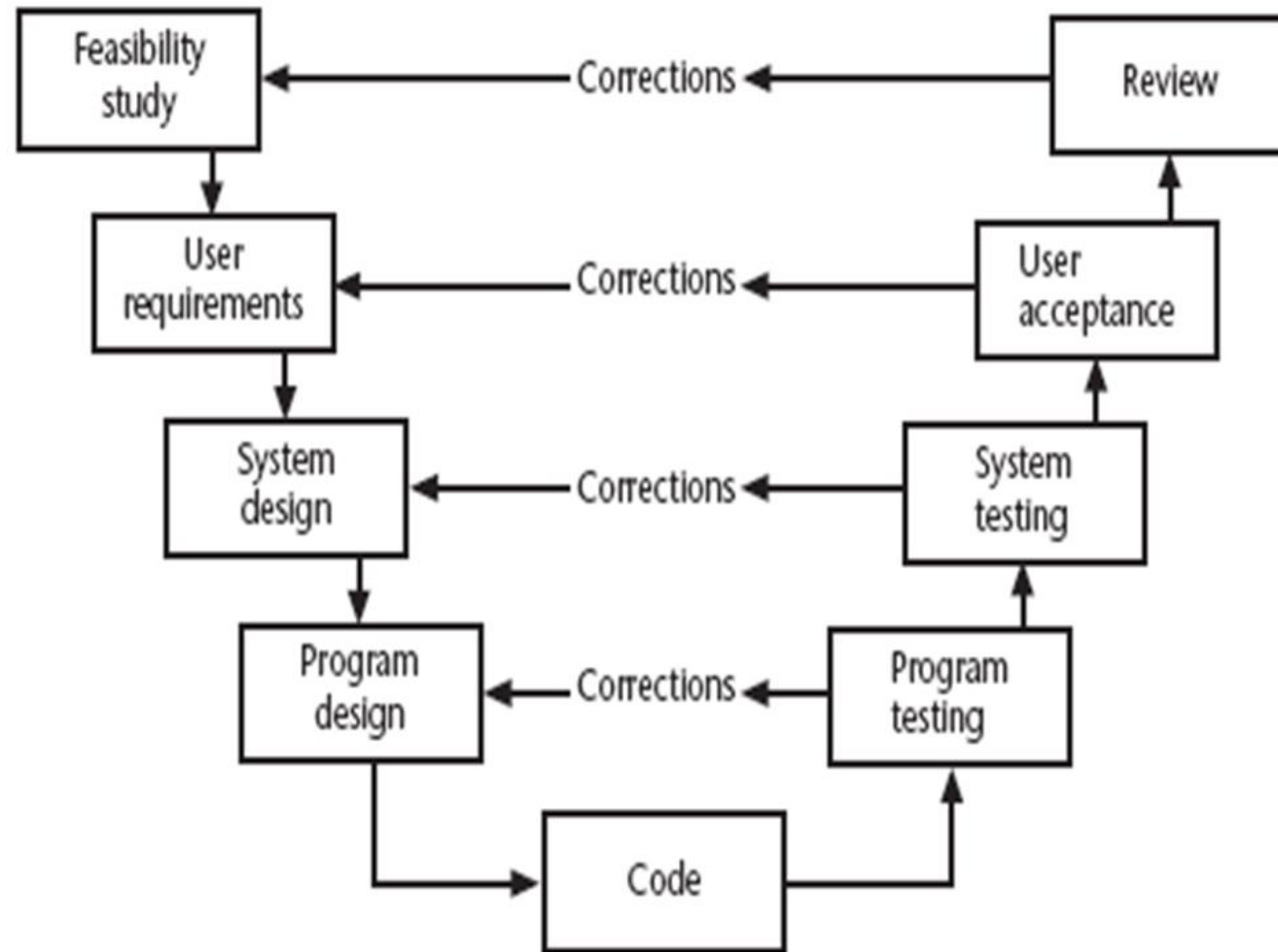- Every stage needs to be checked and signed off

BUT
- limited scope for iteration
- **V-model** approach is an **extension of waterfall** where different testing phases are identified which check the quality of different development phases

The V-Process Model:
Another way of looking at the Waterfall Model

II. Rapid Application Development (RAD)

- RAD-based methodologies adjust the SDLC phases to get some part of system developed quickly and into the hands of the users.

- Most RAD-based methodologies recommend that analysts use special techniques and computer tools to speed up the analysis, design, and implementation phases, such as CASE (computer-aided software engineering) tools.

Evolutionary delivery: prototyping

Main types

- 'throw away' prototypes

- evolutionary prototypes

What is being prototyped?

- human-computer interface

- functionality
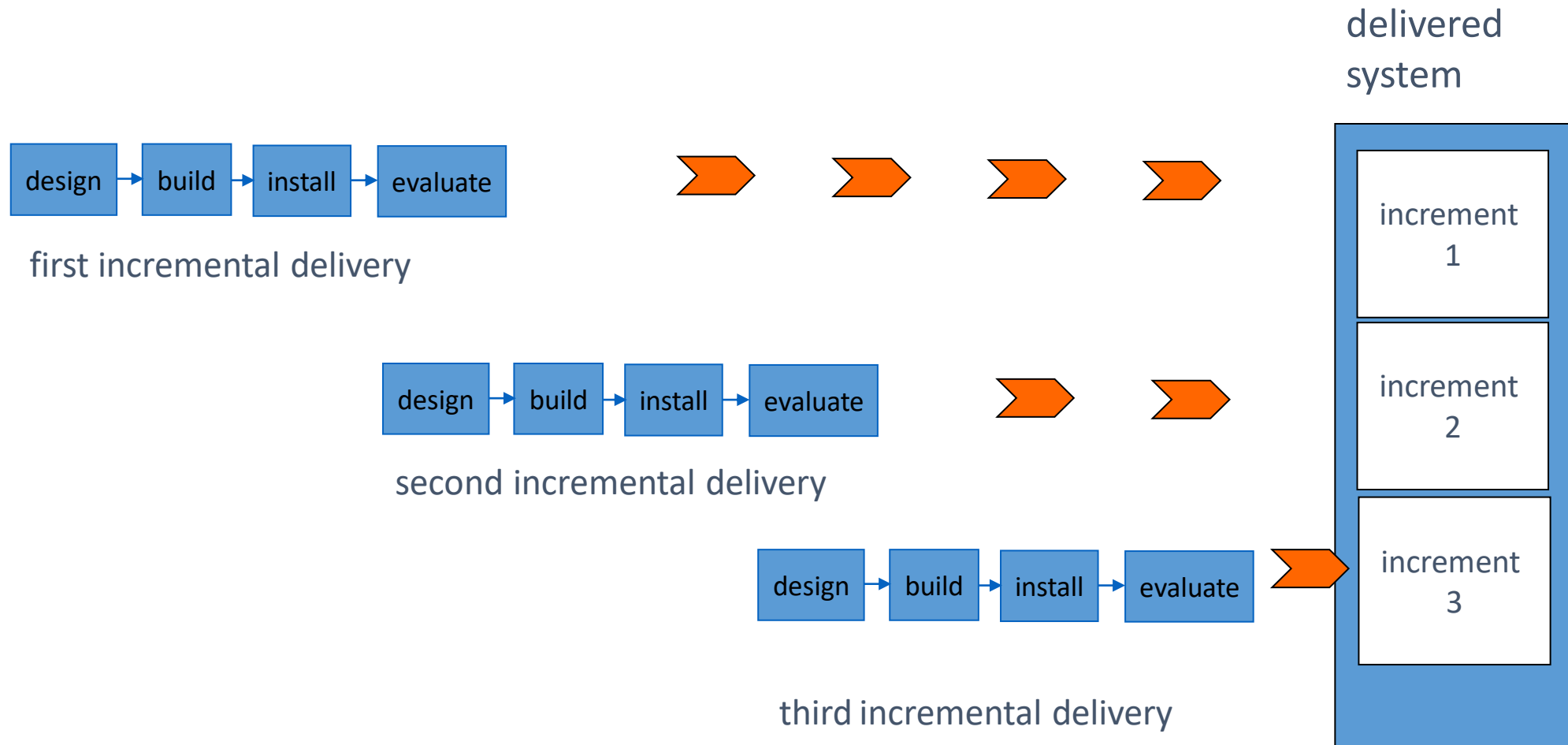
Reasons for prototyping

- learning by doing
- improved communication
- improved user involvement
- a feedback loop is established
- reduces the need for documentation
- reduces maintenance costs i.e., changes after the application goes live
- prototype can be used for producing expected results
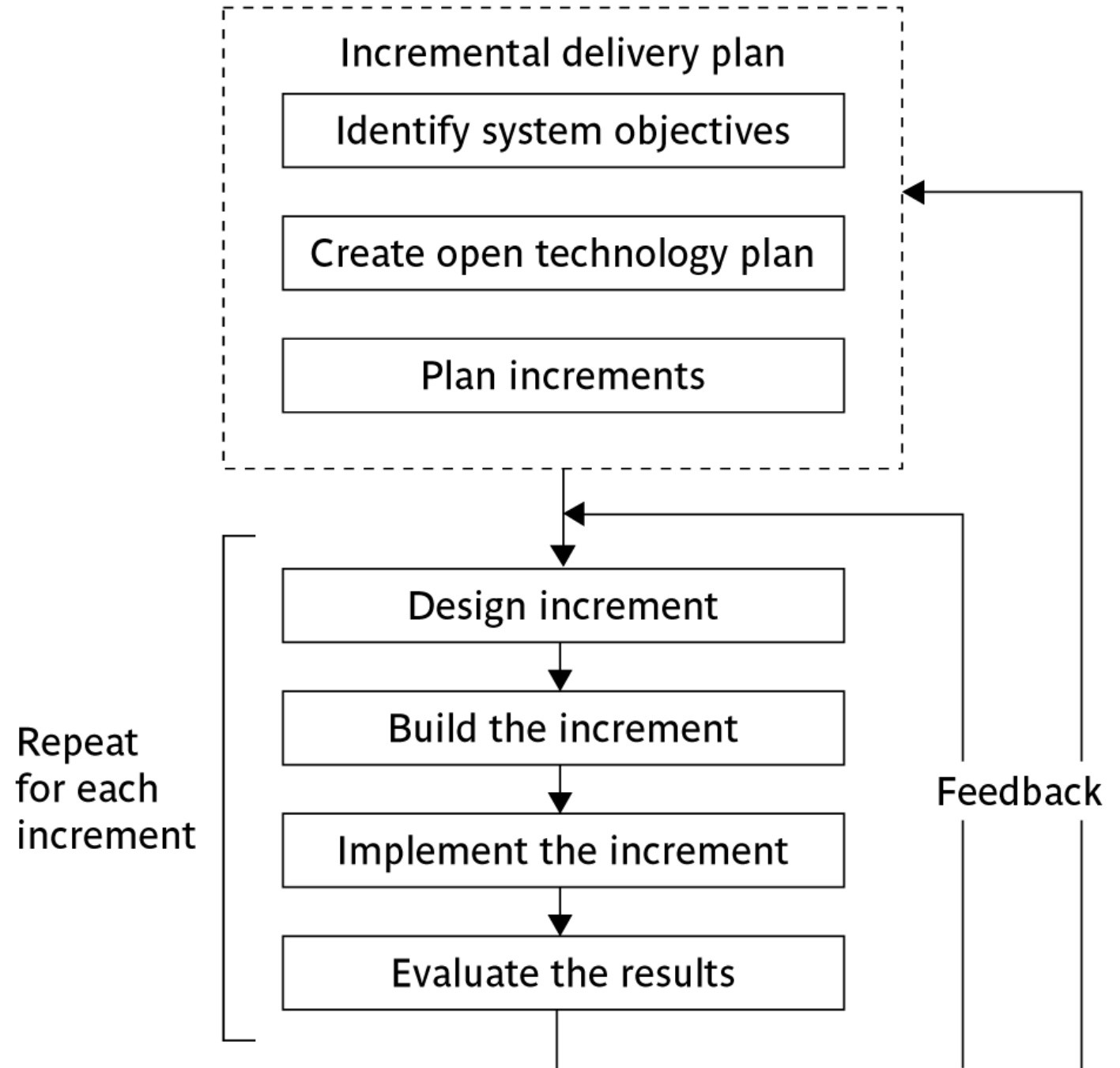
Prototyping: some dangers

- users may misunderstand the role of the prototype
- lack of project control and standards possible
- additional expense of building prototype
- focus on user-friendly interface could be at expense of machine efficiency

Incremental delivery

delivered system

first incremental delivery

design → build → install → evaluate

second incremental delivery

design → build → install → evaluate

third incremental delivery

design → build → install → evaluate

increment 1

increment 2

increment 3

The incremental process

Incremental approach: benefits

- feedback from early stages used in developing latter stages

- shorter development thresholds

- user gets some benefits earlier

- project may be put aside temporarily

BUT there are some possible disadvantages

- loss of economy of scale (one large system vs a series of smaller ones)

- 'software breakage' (modifications to earlier increments)

The outline incremental plan

- steps ideally 1% to 5% of the total project
- non-computer steps should be included
- ideal if a step takes one month or less:
  - not more than three months
- each step should deliver some benefit to the user
- some steps will be physically dependent on others

Structured vs Agile Development

Structured development methods have some disadvantages:

- produce large amounts of documentation which can be largely unread
- documentation has to be kept up to date
- division into specialist groups and need to follow procedures can distress communication
- users can be excluded from decision process
- long lead times to deliver anything etc.

The answer is Agile methods

Dictionary: Agile

- ***Adjective***
  1. quick and well-coordinated in movement:
     an agile leap.

  2. active; lively: *an agile person.*

  3. marked by an ability to think quickly; mentally acute
     or aware: *She's 95 and still very agile.*

  http://www.macmillandictionary.com/pronunciation/british/agile

III. Agile Development

- This category focuses on streamlining the SDLC by eliminating much of the modelling and documentation overhead and the time spent on those tasks.

- Projects emphasize simple, iterative application development.

Dynamic System Development Method (DSDM) a.k.a. Atern

- Produced by UK-based consortium

- Is a widely-used computer application development method in the UK, where its use is often specified as a requirement for government computing projects.

Eight core DSDM principles

1. Focus on business need
2. Deliver on time
3. Collaborate
4. Never compromise quality
5. Develop iteratively
6. Build incrementally from firm foundations
7. Communicate continuously
8. Demonstrate control

DSDM: time-boxing

- *time-box* fixed deadline by which *something* has to be delivered

- typically two to six weeks

- MoSCoW priorities
    - Must have - essential
    - Should have - very important, but system could operate without
    - Could have
    - Want - but probably won't get!

Extreme programming

Extreme Programming (XP) was founded on four core values:

- **Communication**
  - Close interaction with the end users to build systems very quickly
- **Simplicity** (Simple coding)
- **Feedback** (Continuous testing)
- **Courage**

Extreme programming – 2 of 3

- increments of one to three weeks
  - customer can suggest improvement at any point
- argued that distinction between **design and building** of software are artificial
- code to be developed to meet current needs only
- frequent re-factoring to keep code structured

Extreme programming – 3 of 3

- **developers work in pairs**

- test cases and expected results devised *before* software design

- after testing of increment, test cases added to a consolidated set of test cases

Limitations of extreme programming

- Reliance on availability of **high-quality developers**
- Dependence on **personal knowledge** – after development knowledge of software may decay making future development less easy
- Rationale for decisions may be lost e.g. which test case checks a particular requirement
- **Reuse** of existing code less likely

Scrum

Scrum is an agile software development model based on multiple small teams working in an intensive and interdependent manner.
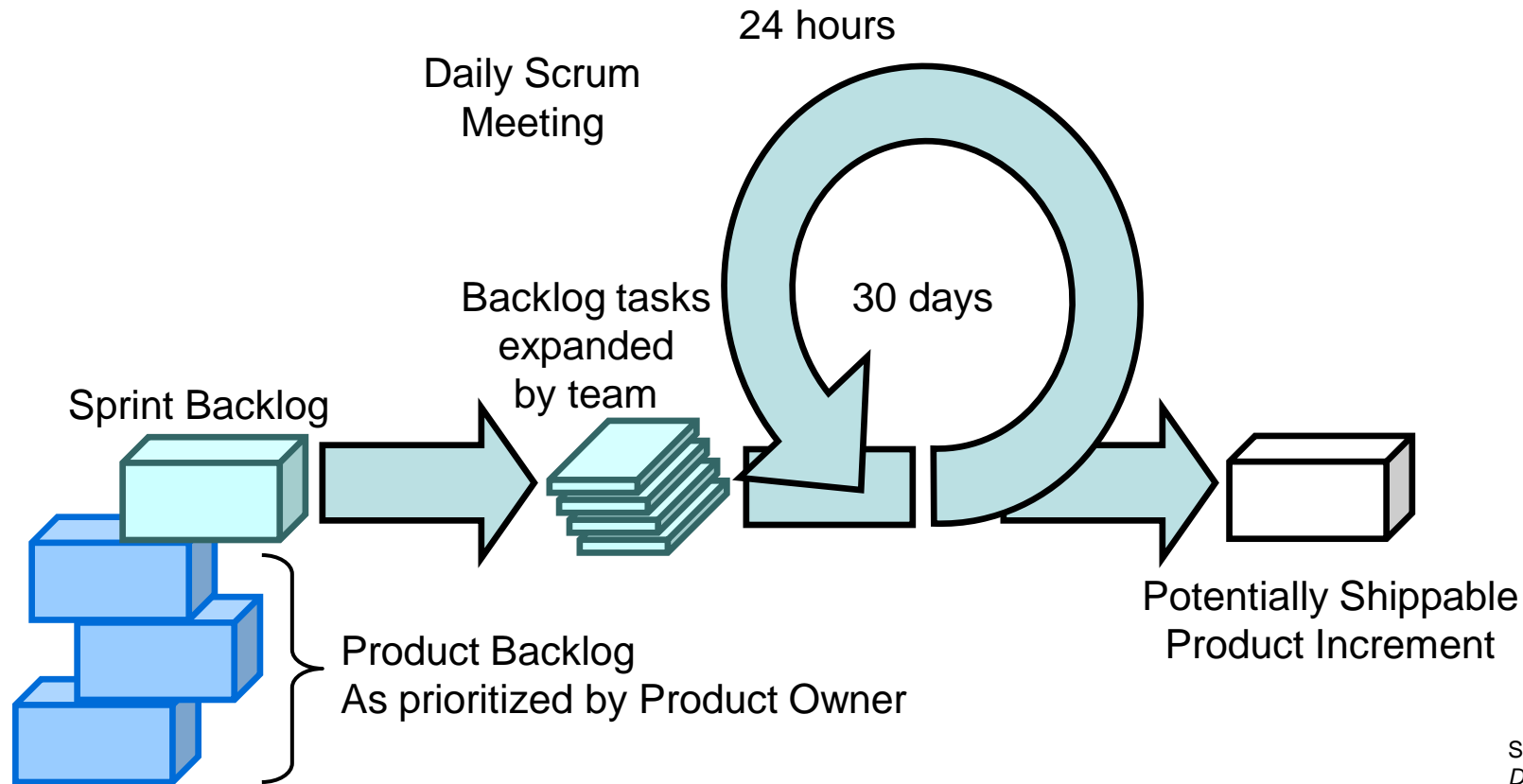

The term is named for the scrum formation in rugby, which is used to restart the game after an event that causes play to stop.

Characteristics

- Self-organizing teams
- Product progresses in a series of month-long "sprints"
- Requirements are captured as items in a list of "product backlog"
- No specific engineering practices prescribed
- Uses generative rules to create an agile environment for delivering projects
- One of the "agile processes"

How Scrum works

24 hours

Daily Scrum Meeting

Backlog tasks expanded by team

30 days

Sprint Backlog

Product Backlog
As prioritized by Product Owner

Potentially Shippable Product Increment

Source: Adapted from *Agile Software Development with Scrum* by Ken Schwaber and Mike Beedle.

'Rules of Thumb' about approach to be used

IF uncertainty is high
  THEN use evolutionary approach

IF complexity is high but uncertainty is not
  THEN use incremental approach

IF uncertainty and complexity both low
  THEN use waterfall or V-prosess

IF schedule is tight
  THEN use evolutionary or incremental

Thank you for your attention

Any questions, please?