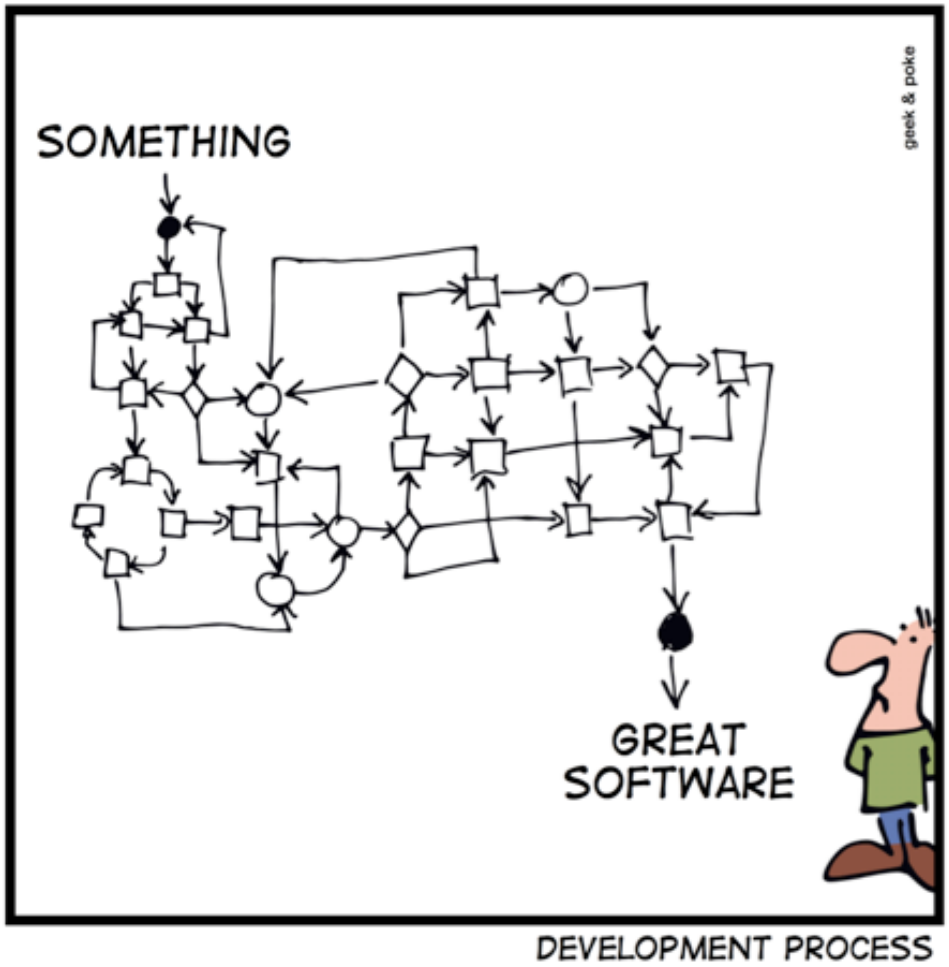
http://home.hit.no/~hansha/?page=software_development



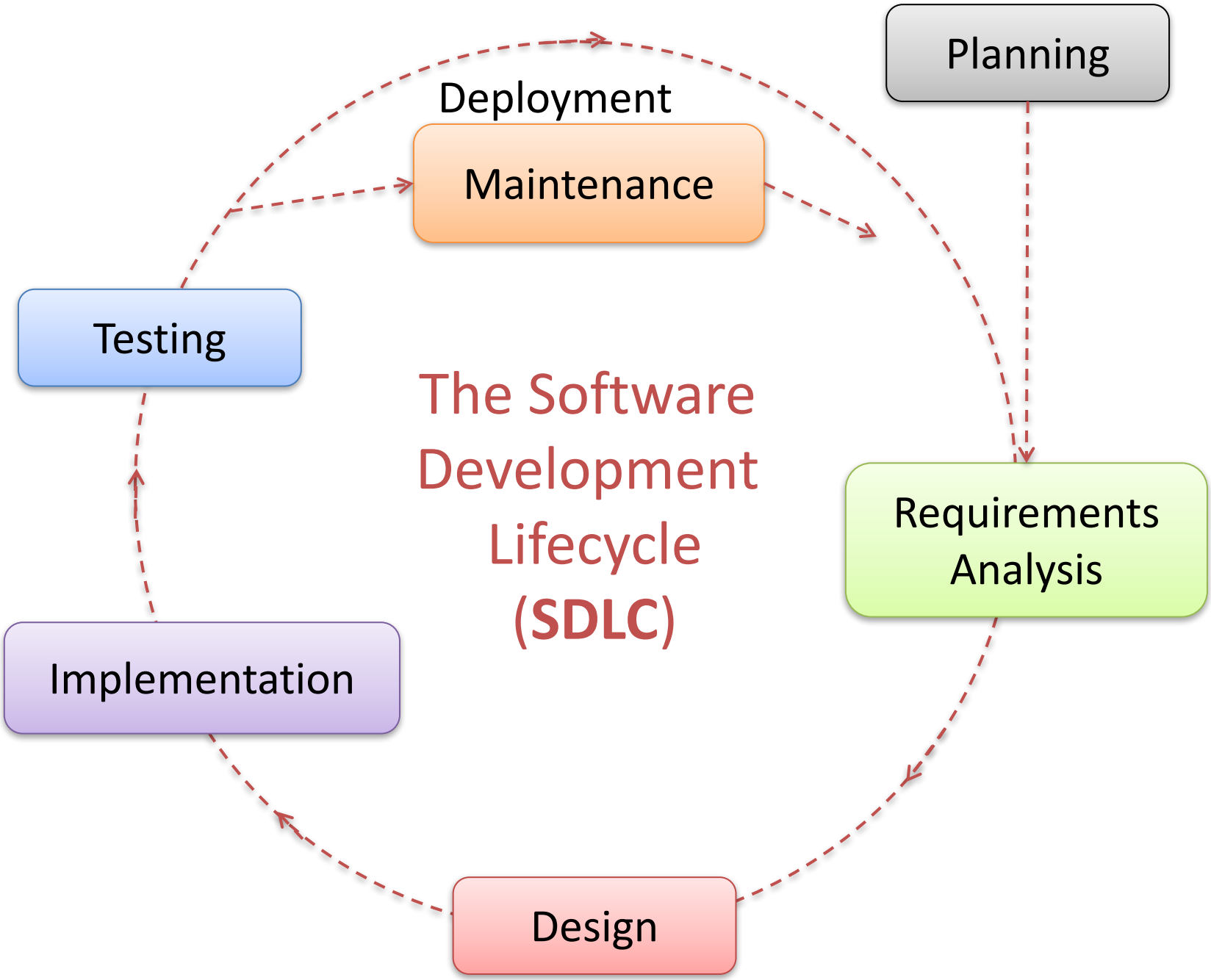
O. Widder. (2013). *geek&poke*. Available: http://geek-and-poke.com

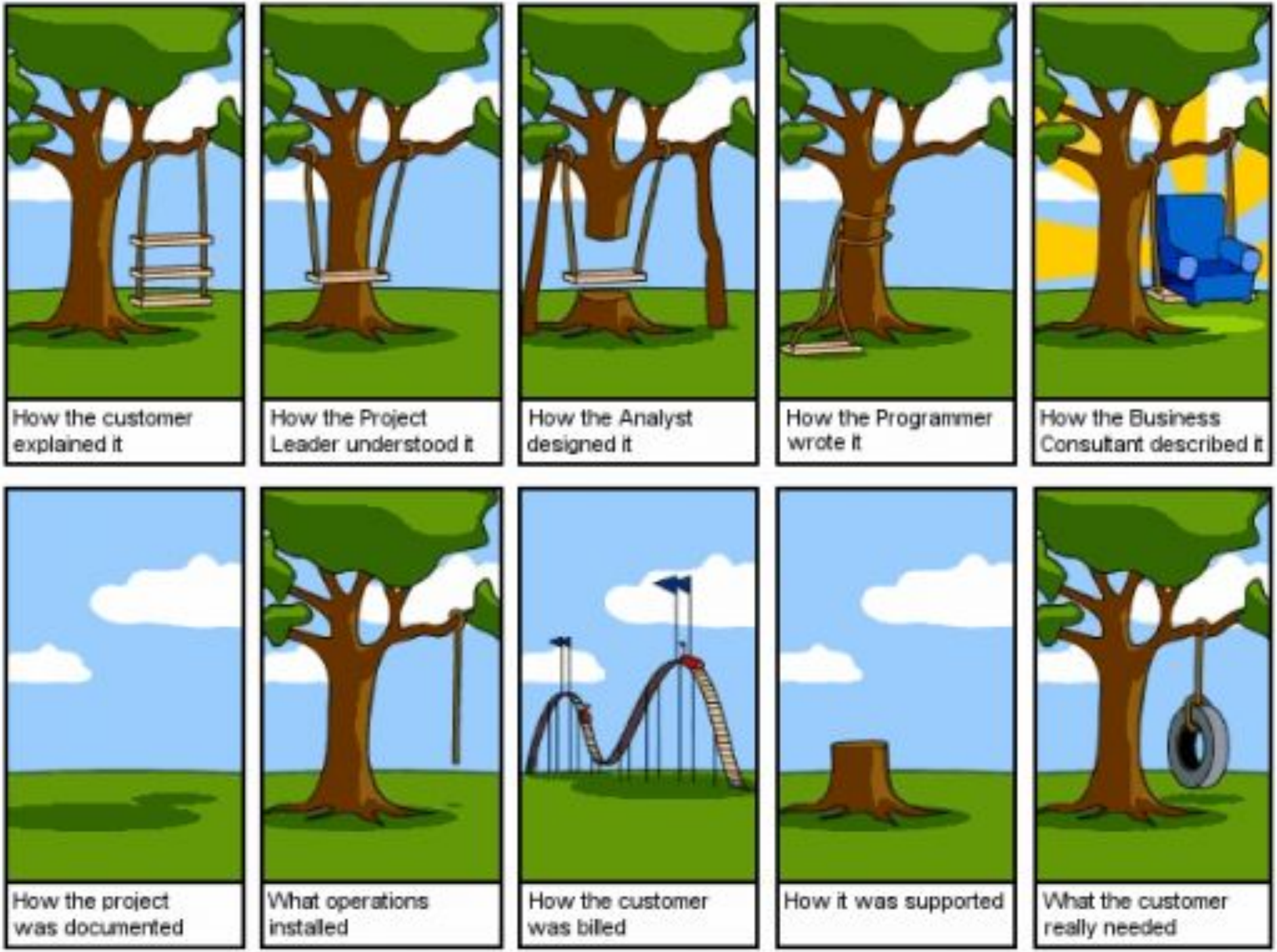
The Software Development Process (SDLC)

Hans-Petter Halvorsen

IT System

B. Lund. (2013). *Lunch*. Available: http://www.lunchstriper.no, http://www.dagbladet.no/tegneserie/lunch/


Planning
Deployment
Maintenance
Testing
The Software Development Lifecycle (SDLC)
Requirements Analysis
Implementation
Design


How the customer explained it
How the Project Leader understood it
How the Analyst designed it
How the Programmer wrote it
How the Business Consultant described it
How the project was documented
What operations installed
How the customer was billed
How it was supported
What the customer really needed

Systems Development Life Cycle (SDLC)
Life-Cycle Phases

Initiation

Begins when a sponsor identifies a need or an opportunity. Concept Proposal is created

System Concept Development

Defines the scope or boundary of the concepts. Includes Systems Boundary Document. Cost Benefit Analysis. Risk Management Plan and Feasibility Study.

Planning

Develops a Project Management Plan and other planning documents. Provides the basis for acquiring the resources needed to achieve a soulution.

Requirements Analysis

Analyses user needs and develops user requirements. Create a detailed Functional Requirements Document.

Design

Transforms detailed requirements into complete, detailed Systems Design Document Focuses on how to deliver the required functionality

Development

Converts a design into a complete information system Includes acquiring and installing systems environment; creating and testing databases preparing test case procedures; preparing test files, coding, compiling, refining programs; performing test readiness review and procurement activities.

Integration and Test

Demonstrates that developed system conforms to requirements as specified in the Functional Requirements Document. Conducted by Quality Assurance staff and users. Produces Test Analysis Reports.

Implementation

Includes implementation preparation, implementation of the system into a production environment, and resolution of problems identified in the Integration and Test Phases

Operations & Maintenance

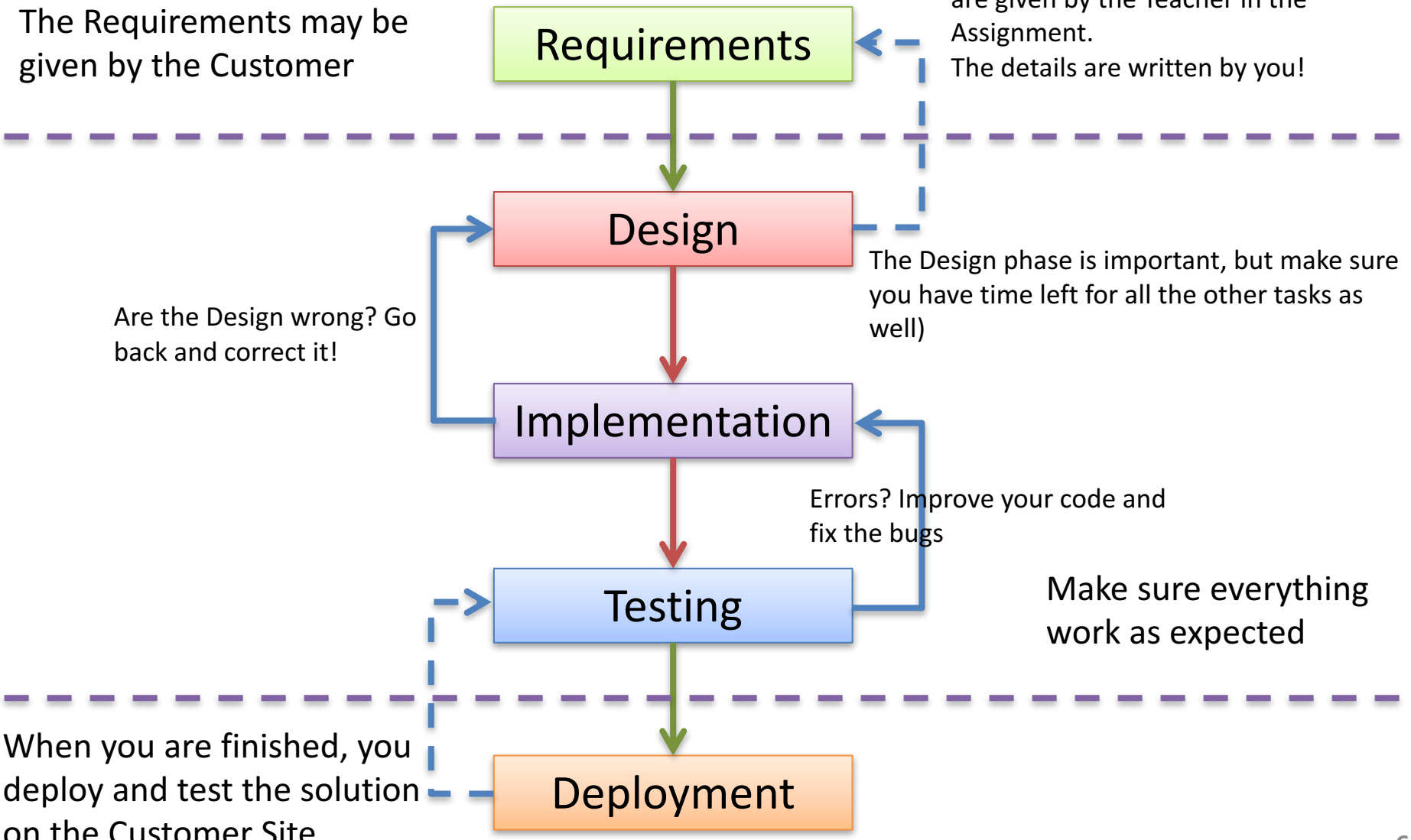
Describes tasks to operate and maintain information systems in a production environment. includes Post-Implementation and In-Process Reviews.

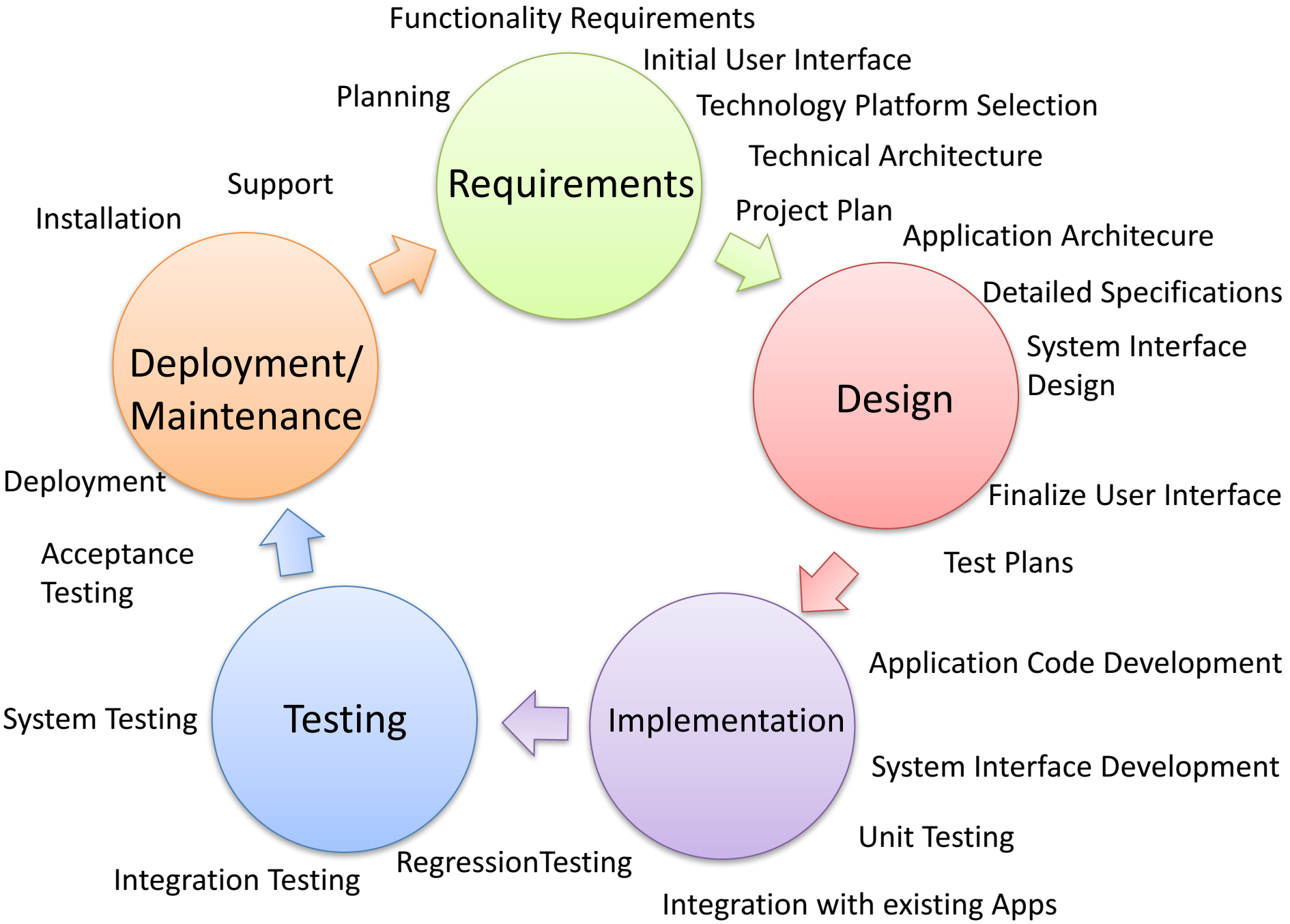
Disposition

Describes end-of-system activities, emphasis is given to proper preparation of data.

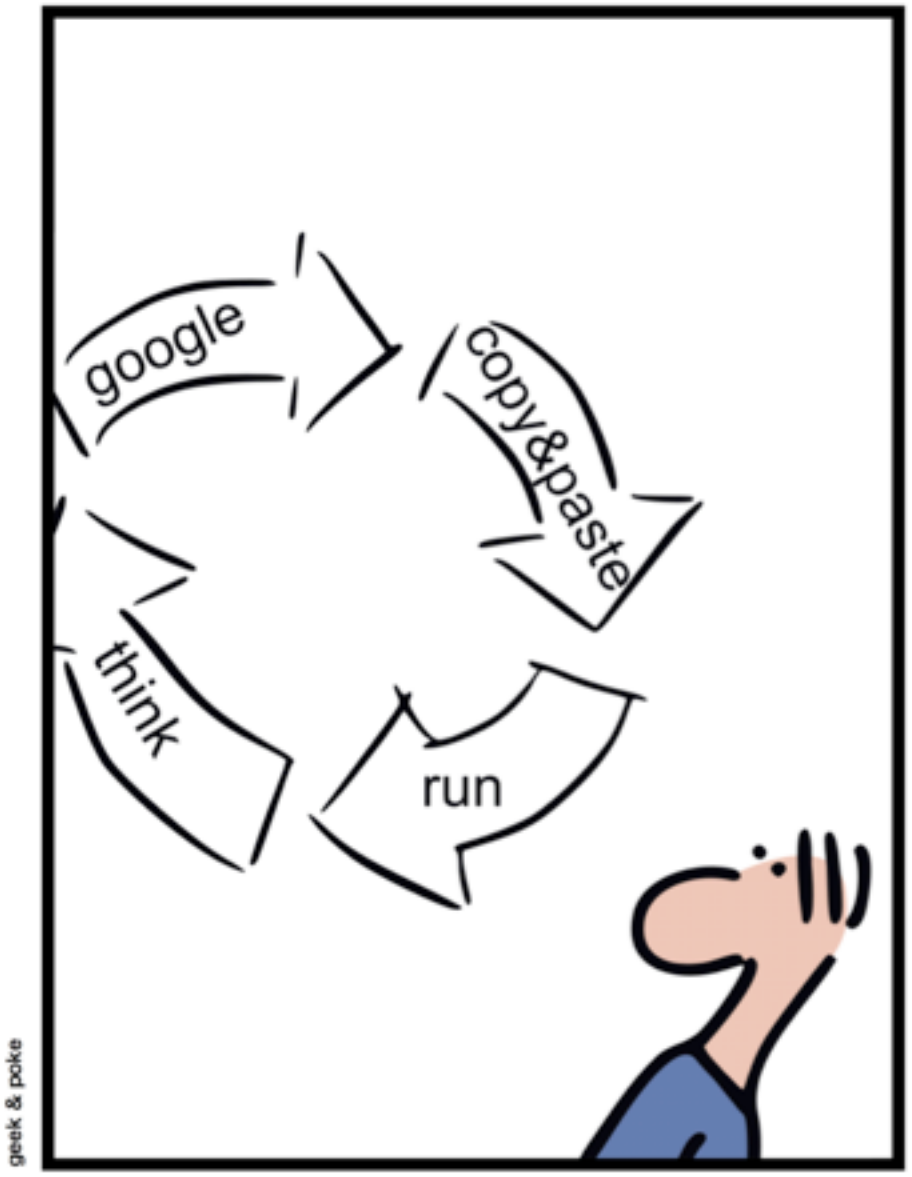
The Development Process

The Development Process involves different phases, e.g.:




Functionality Requirements
Initial User Interface
Technology Platform Selection
Technical Architecture
Project Plan
Application Architecure
Detailed Specifications
System Interface Design
Finalize User Interface
Test Plans
Application Code Development
System Interface Development
Unit Testing
Integration with existing Apps
RegressionTesting
Integration Testing
System Testing
Acceptance Testing
Deployment
Installation
Support
Planning
Requirements
Design
Implementation
Testing
Deployment/ Maintenance

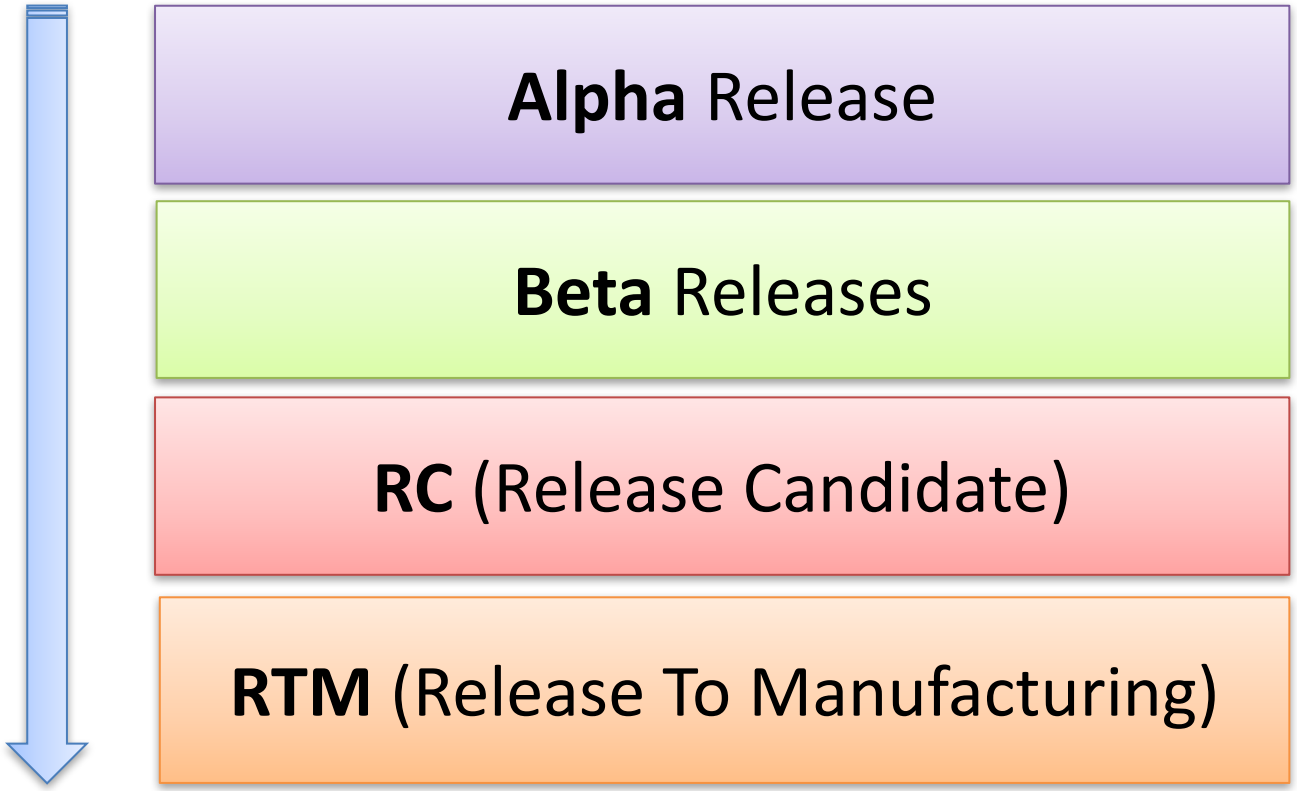
SIMPLY EXPLAINED



O. Widder. (2013). *geek&poke*. Available: http://geek-and-poke.com

Software Releases

Start



Finished

Note! other terms are also used

Software Releases

Before the software is released

- **Alpha** Release(s) (Internal release, not public)
- **Beta** Release(s) (“Developer Preview”, “Consumer Preview”)
- **RC** - Release Candidate(s) (“Release Preview”)
- **RTM** – Release To Manufactoring

Maintenance (after the software is released)

- **Patches** (small fixes)
- **SP** - Service Packs (lots of small fixes and pathes bundle together), SP1, SP2, R1, R2, ..

…

Start Planning next Release

Teams and Roles

Customer/Stakeholders

- Customer/Stakeholders
- Project Manager
- Software Architect
- Software Designer
- Developer
- Tester
- etc.

Software Architect

Collaboration!

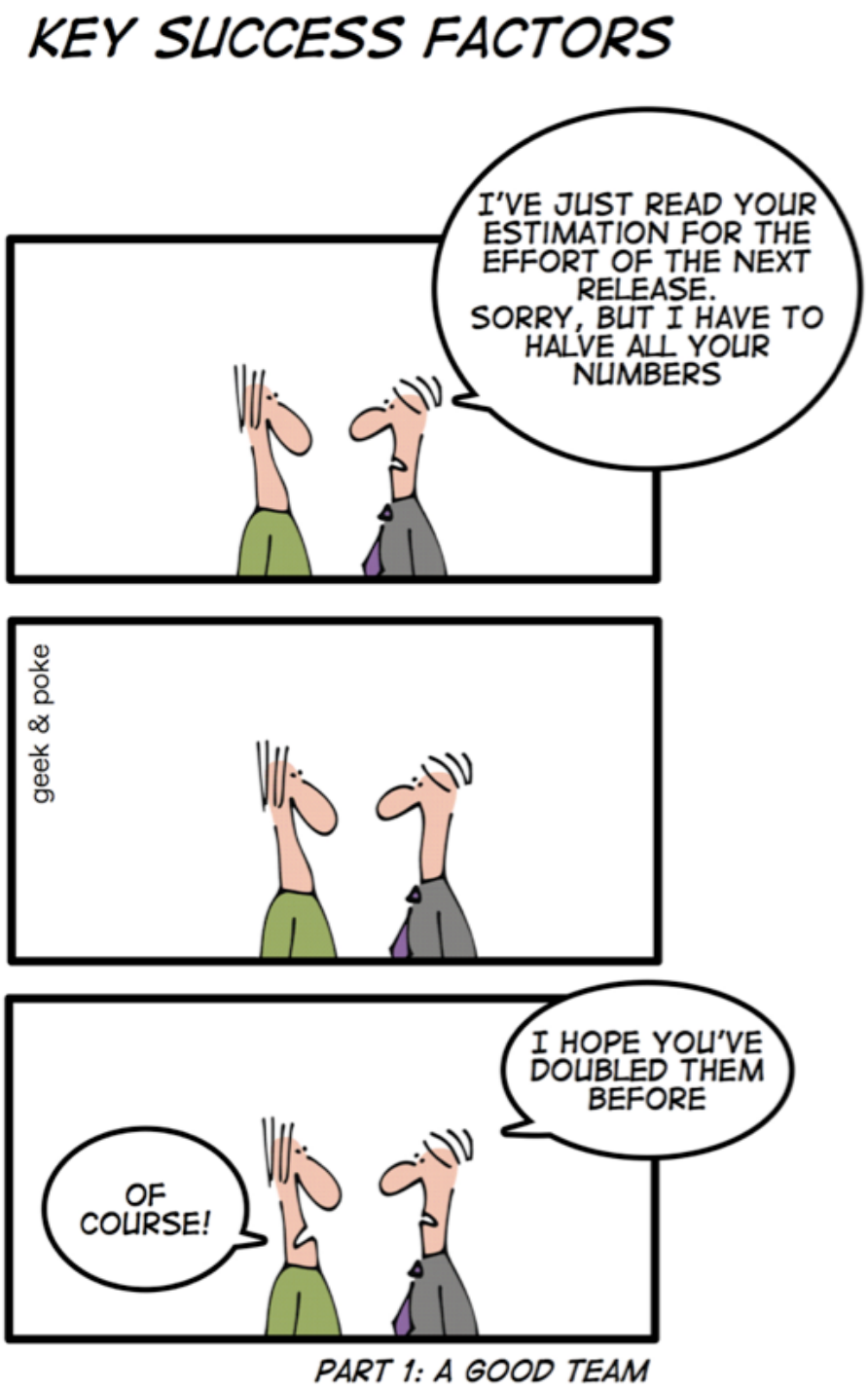
Software Tester

Project Manager

Software Designer

Programmer

Software Teams



Software Team

Stakeholders

Project Manager

Software Tester

UX Designer

System Engineer

Programmer

Software Architect

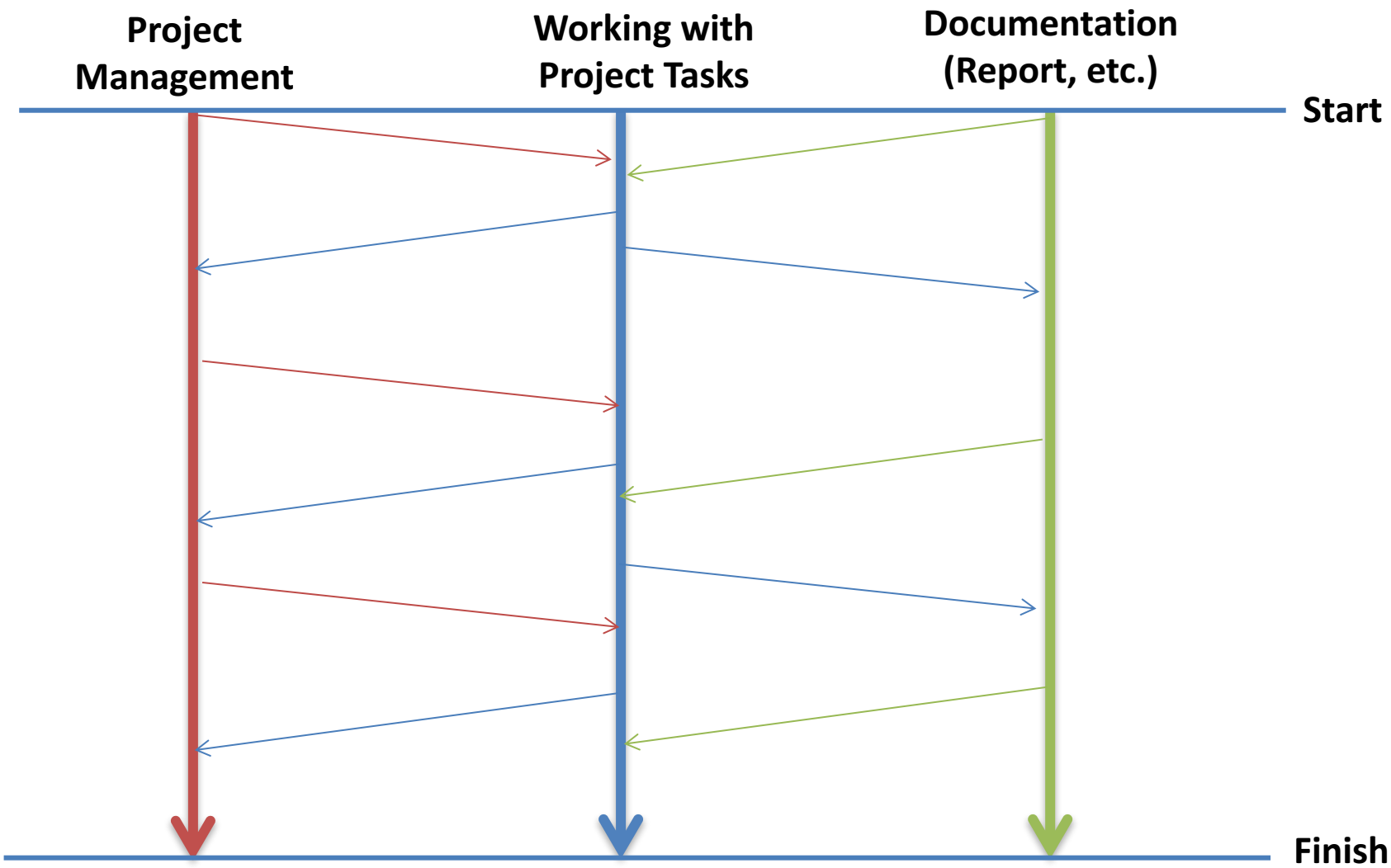
A System Engineer is a general person that could be a Programmer, Architect, Designer, Tester in different phases in the project, or he could be a tester in one project and a programmer in another project – all in one person. That is usually the case in small companies, while in larger companies these roles (designer, tester, programmer) could be a full-time job.

Project Planning and Management

Project Planning Tools: **Gantt Chart**, Backlog, Task Board, Burn Down Chart, etc.

B. Lund. (2013). *Lunch*. Available: http://www.lunchstriper.no, http://www.dagbladet.no/tegneserie/lunch/

How to work in the project period

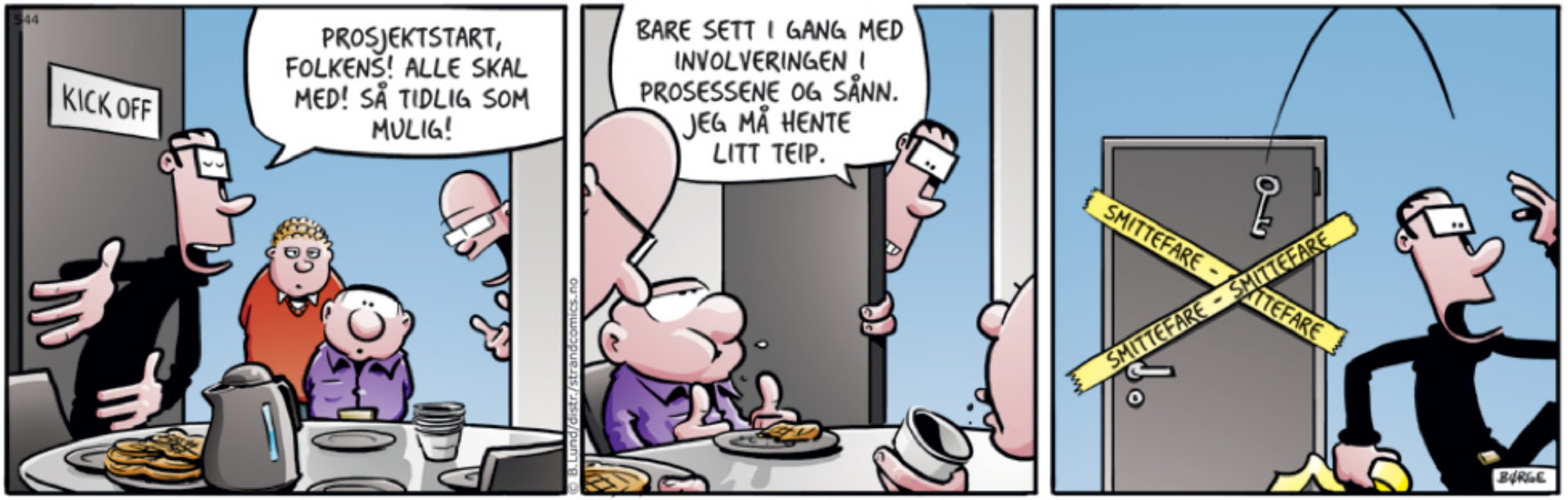


Important: Work with these activities in parallel!!!

Brainstorming/Kick Off

A Project should always start with a Brainstorming

- **Involve** all in the group
- Discuss what you are going to do in the project
- How are you going to solve the project?
- etc.



Deadlines

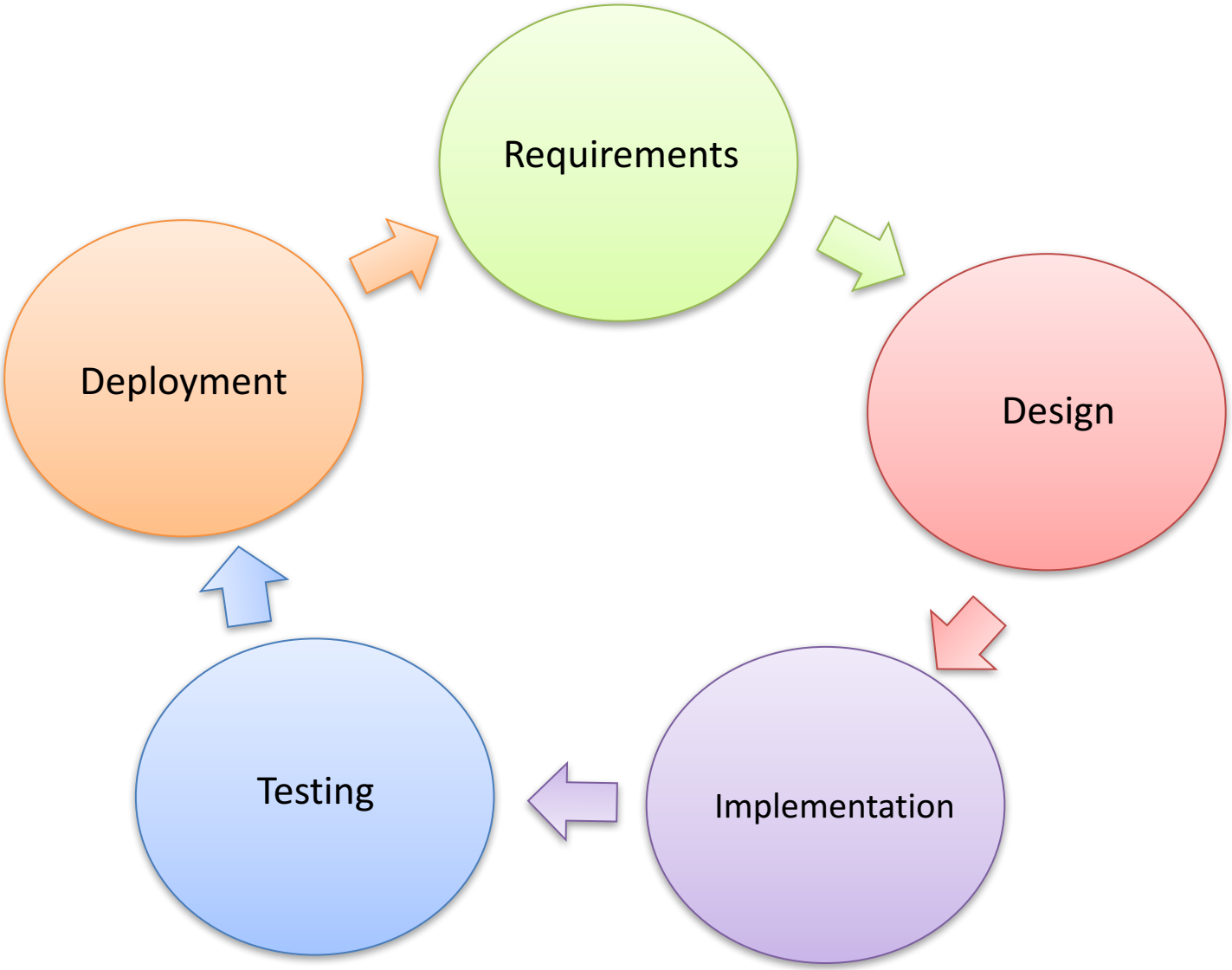
As a software developer you need to deal with lots of deadlines during the software project!

Proper Project Planning and Management makes it easier to fulfill these deadlines

Software Phases

Hans-Petter Halvorsen

Software Phases



Software Development Process – **Requirements & Design**

- The Requirements is normally given by the Customer
- SRS – Software Requirements Specification document

Software Requirements & Design

Requirements (WHAT):

- **WHAT** the system should do
- Describe what the system should do with Words and Figures,etc.
- **SRS** – Software Requirements Specification

Software Design (HOW):

- **HOW** it should do it
- Examples: GUI Design, UML, ER diagram, CAD, etc.
- **SDD** – Software Design Document

Many dont separate SRS and SDD documents, but include everything in a Requirements document.

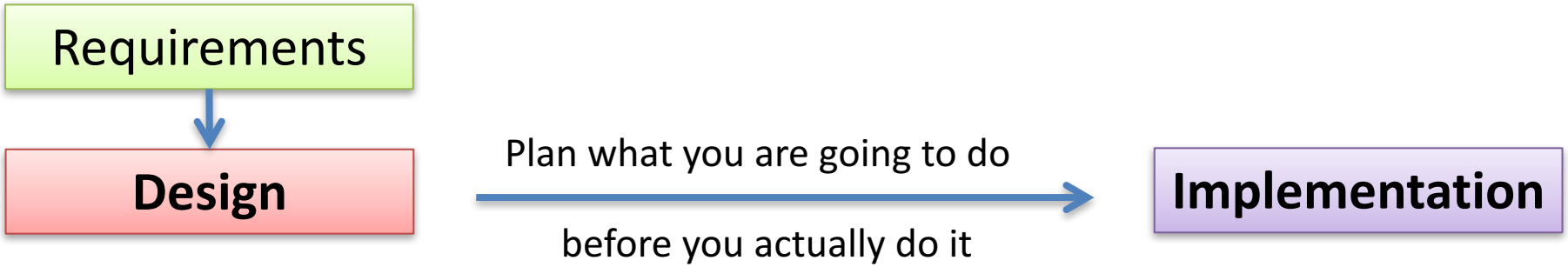
In practice, requirements and design are inseparable.

Software Development Process - **Design**

How the Software will work

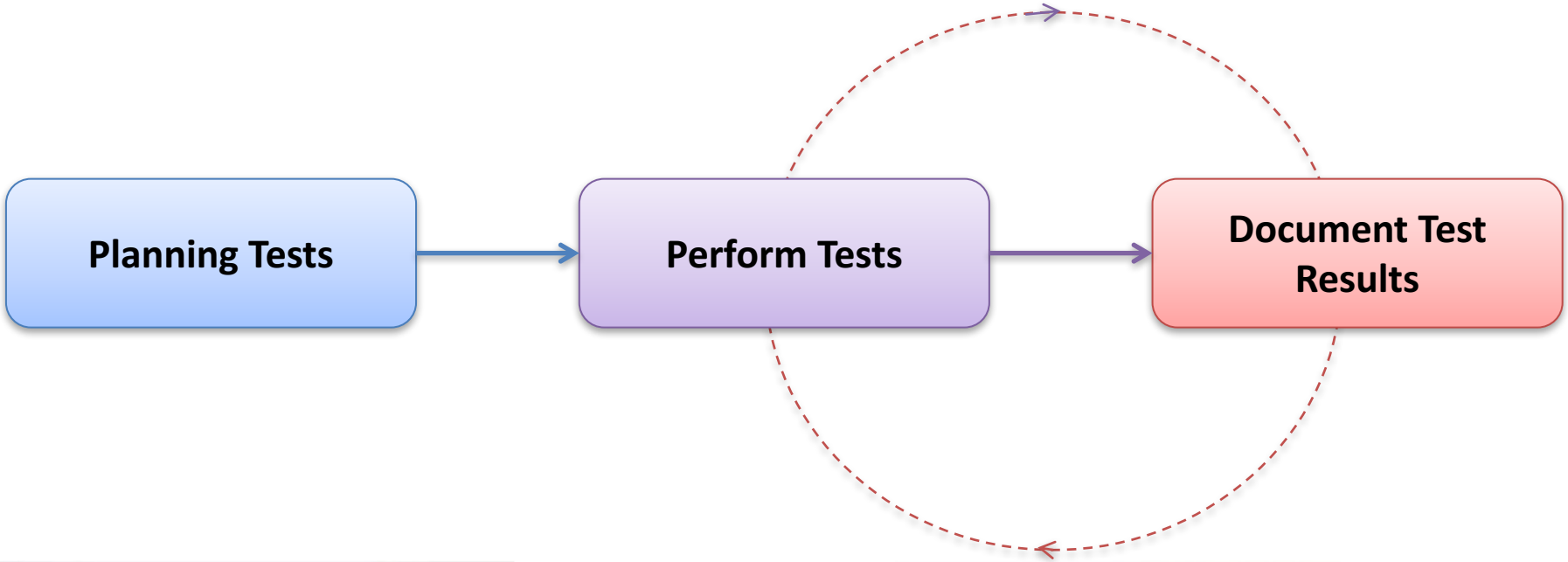
- Technical Design (Platform, Architecture, etc.)
- UX Design (User eXperience, GUI/HMI)

Software Development Process



- Make sure to have the Design ready before you start the Implementation (Coding)
- Flow Charts, UML, Database Modelling, etc.
- Create Detailed Requirements
- GUI/HMI – Start by designing your GUI on the "paper"

Software Development Process - **Testing**



Software Development Process - **Testing**

Always test your application!

- Test it yourself
- Test it on other computers and environments
- Make sure others test your application
- Eat Your Own Dog Food

Software Development Process - **Deployment**

Software deployment is all of the activities that make a software system available for use.

- Make sure the Software is well tested
- The Software should be easy to install

Software Development Methods/Processes

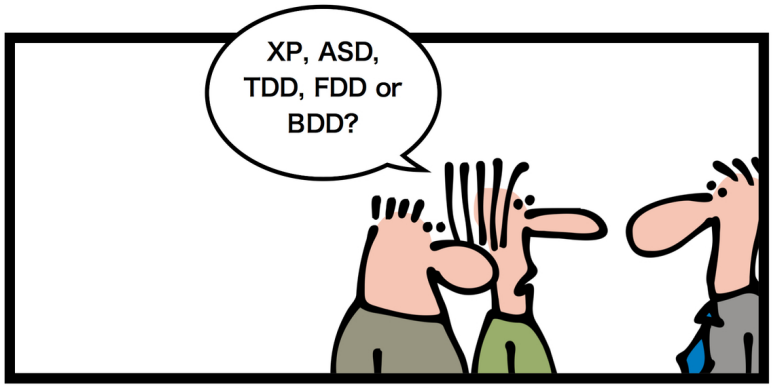
How we approach the different phases in software development (Requirements, Design, Implementation, Testing, Deployment) differ depending on what kind of Software Development method we choose.

A short overview of the most used/known methods are given on the next slides.

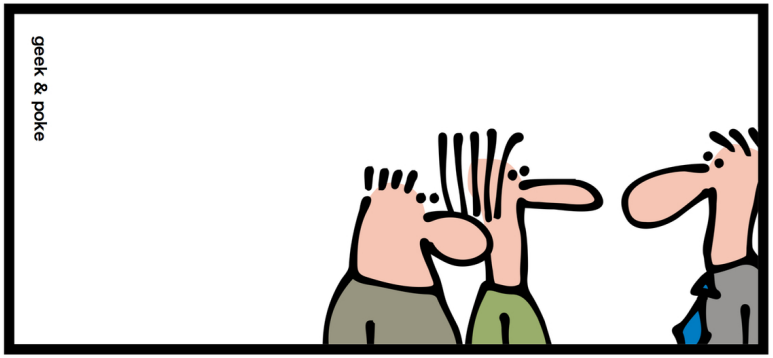
Hans-Petter Halvorsen

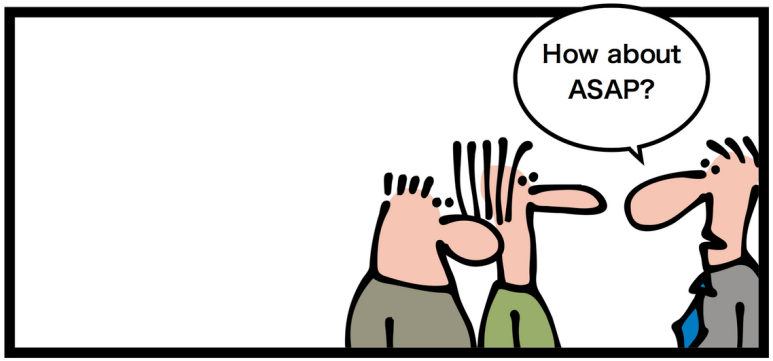
Software Development Methods

- **Waterfall model**
- Spiral model
- V model
- Iterative and Incremental development
- Rapid application development (RAD)
- Test Driven Development (TDD)
- **Agile development**
  - eXtreme Programming (XP)
  - **Scrum** (the most popular Agile method)
- Lean software development
- Kanban
- etc.



When you're thinking about new software development approaches...





··· don't ask your boss!!!

Software Development Methods

Traditional Plan-driven Methods

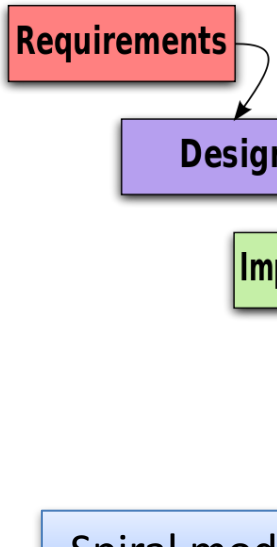
Agile Methods

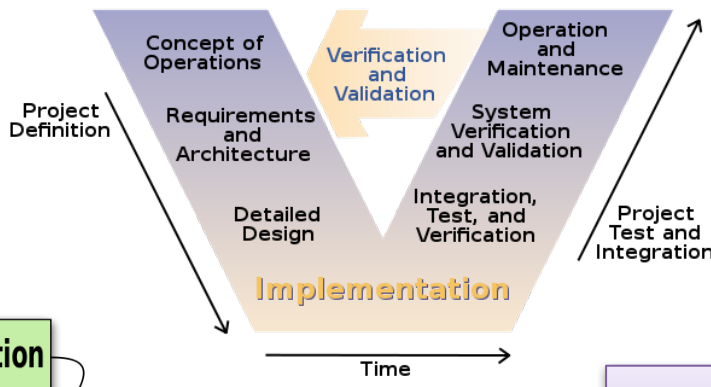
Waterfall Method

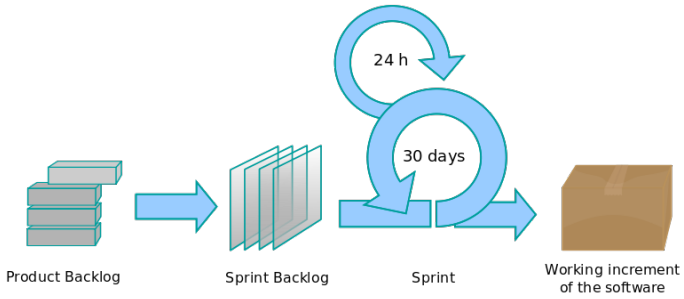
V-Model

eXtreme Programming (XP)

Scrum







Lean Software Development

Feature Driven Development (FDD)

Spiral model

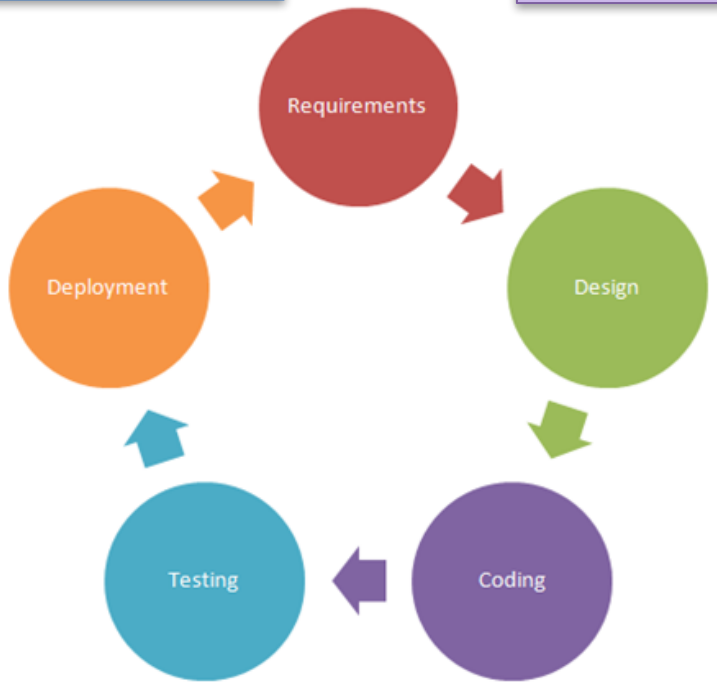
Kanban

Test Driven Development (TDD)

Software Development Methods

Traditional Plan-driven Methods

Agile Methods



Waterfall Method

eXtreme Programming (XP)

Scrum

Feature Driven Development (FDD)

V-Model

Test Driven Development (TDD)

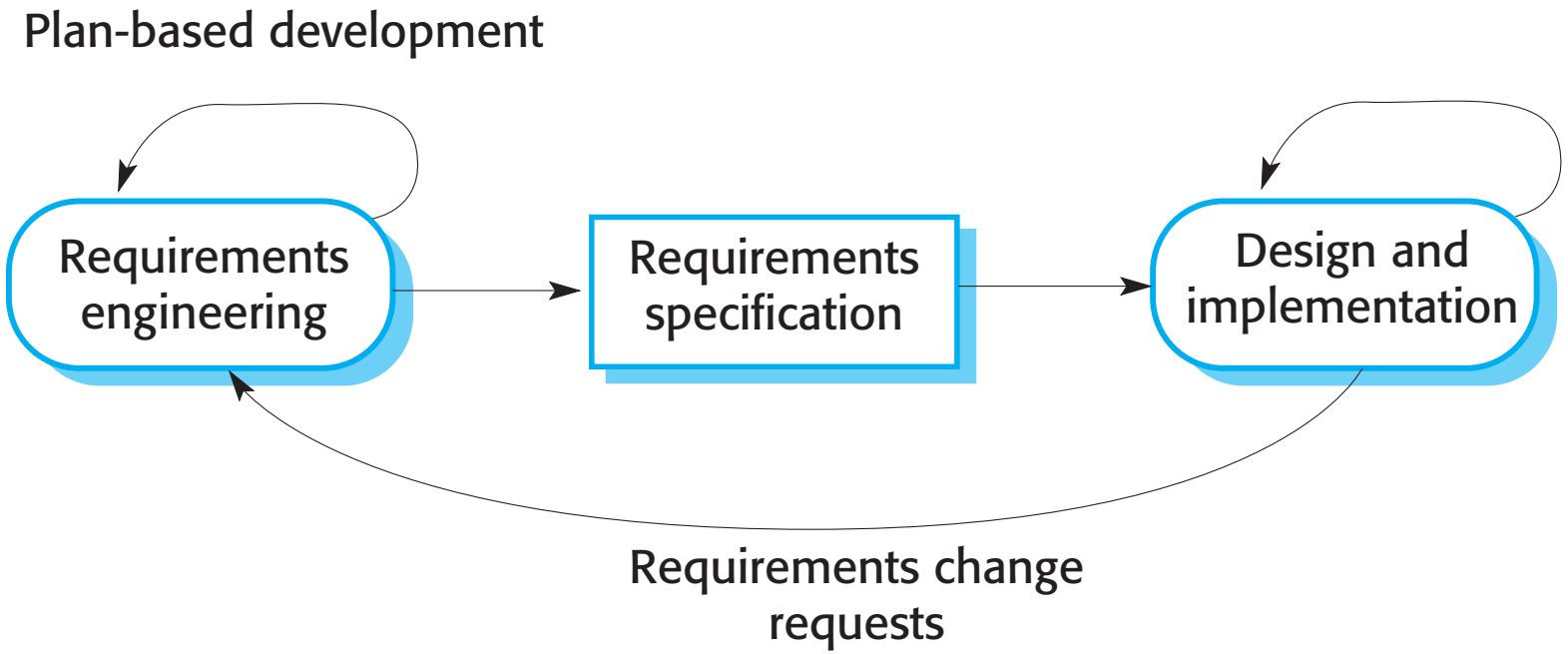
Spiral model

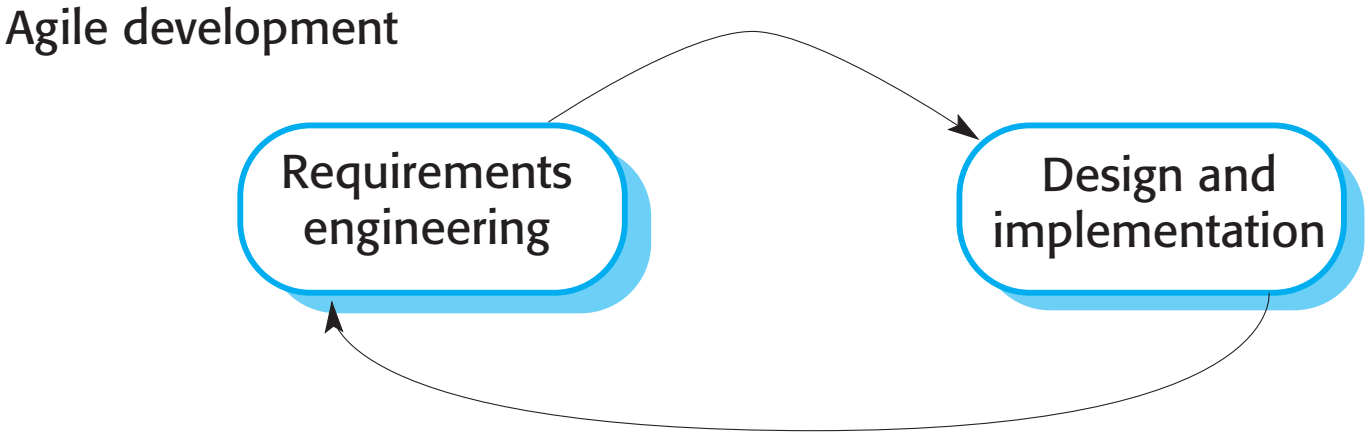
Lean Software Development

Kanban

Even if we use different software development methods, we deal with the same phases like Requirements, Design, Coding, Testing and Deployment – but they may have different priority and may be done in different manners and order, etc.

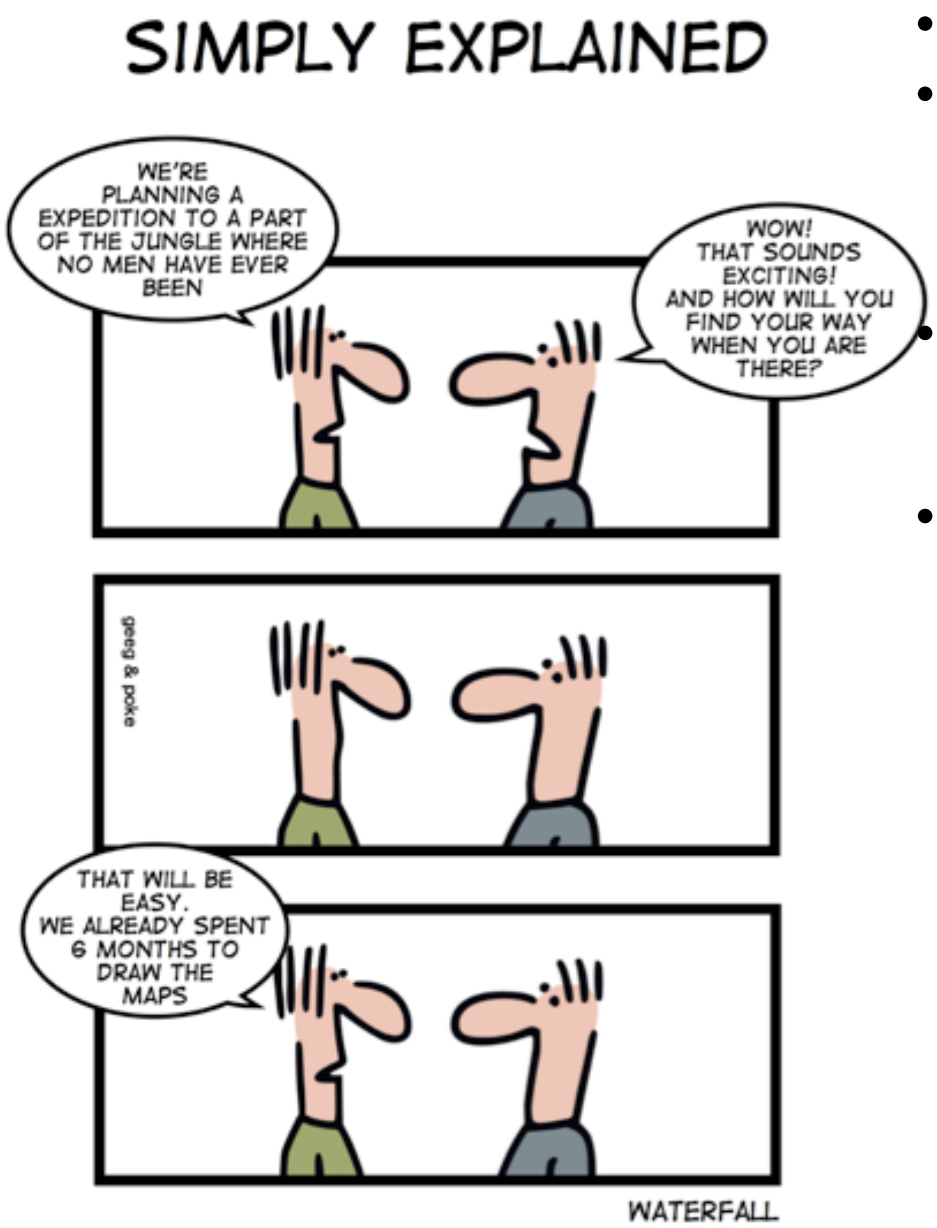
Plan-driven vs. Agile





I. Sommerville, *Software Engineering*: Pearson, 2010.

Waterfall method

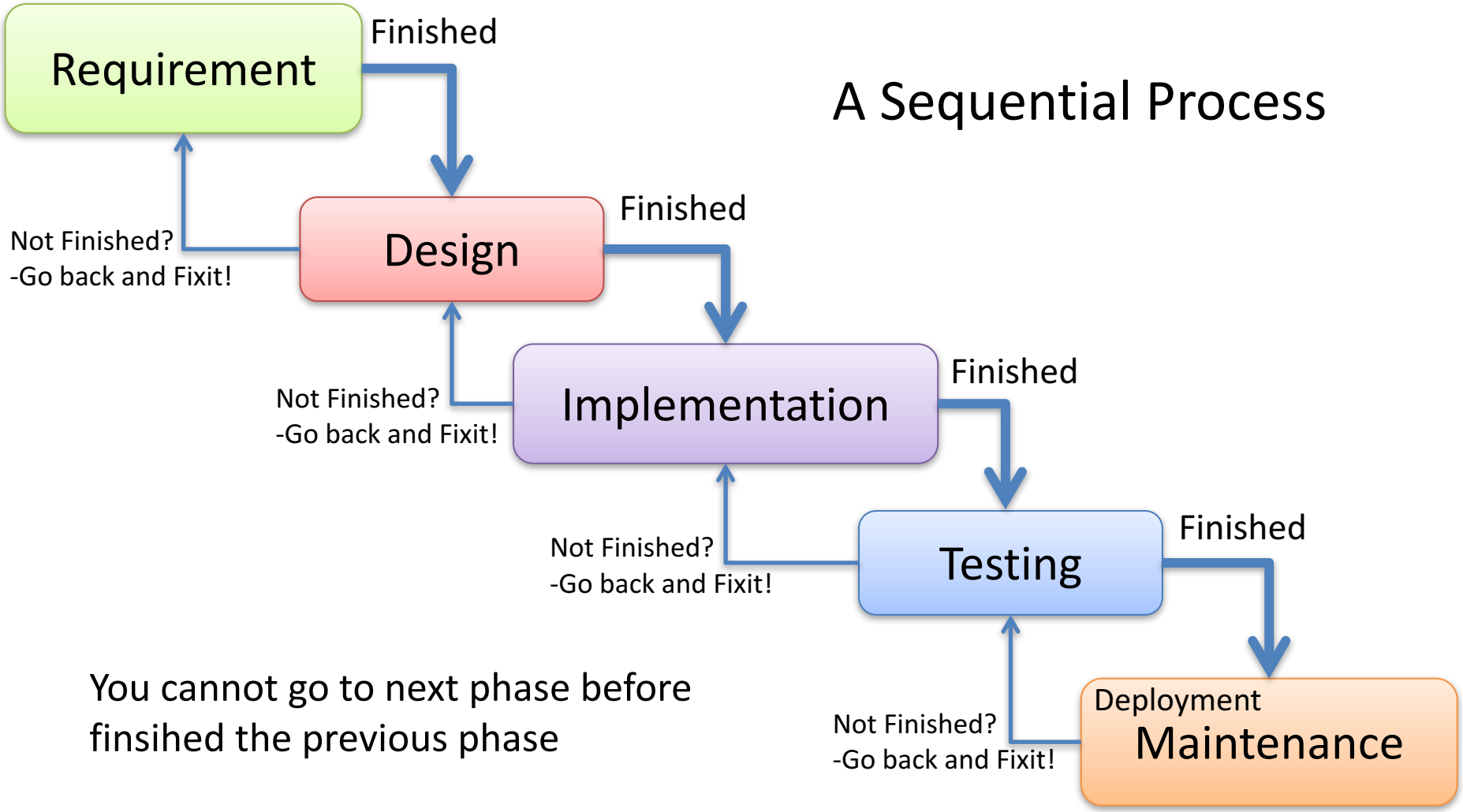


O. Widder. (2013). *geek&poke*. Available: http://geek-and-poke.com

- Sequential process
- The Phases Requirements, Design, Implementation, Testing, Deployment and Maintenance need to be followed in that order.
- In a strict Waterfall model, after each phase is finished, it proceeds to the next one.
- Cons:
  - impossible for any non-trivial project to finish a phase of a software product's lifecycle perfectly before moving to the next phases
  - For example, clients may not know exactly what requirements they need before reviewing a working prototype and commenting on it

The Waterfall Model

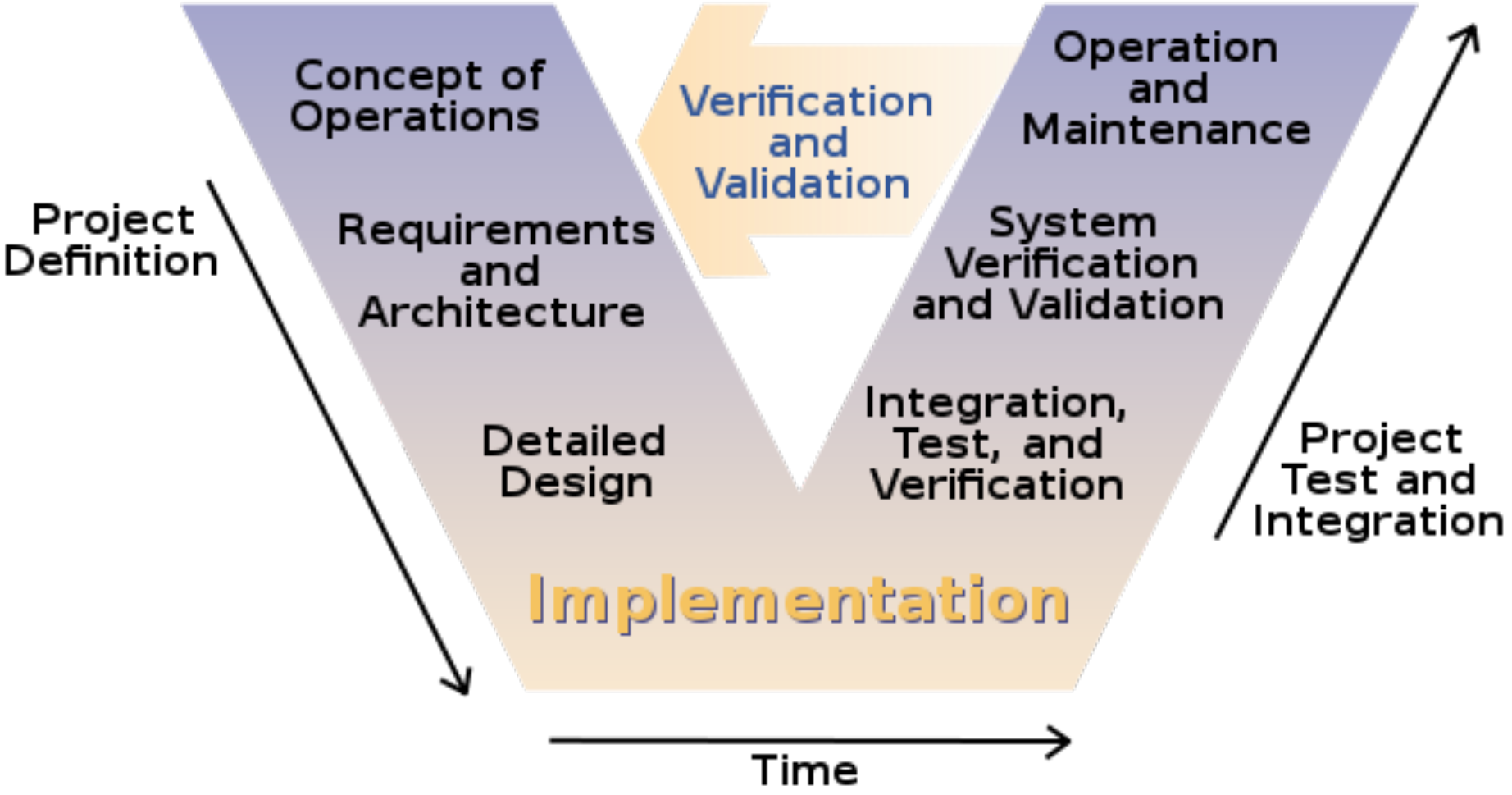
Planning to create a new Software



A Sequential Process

You cannot go to next phase before finsihed the previous phase

V-model

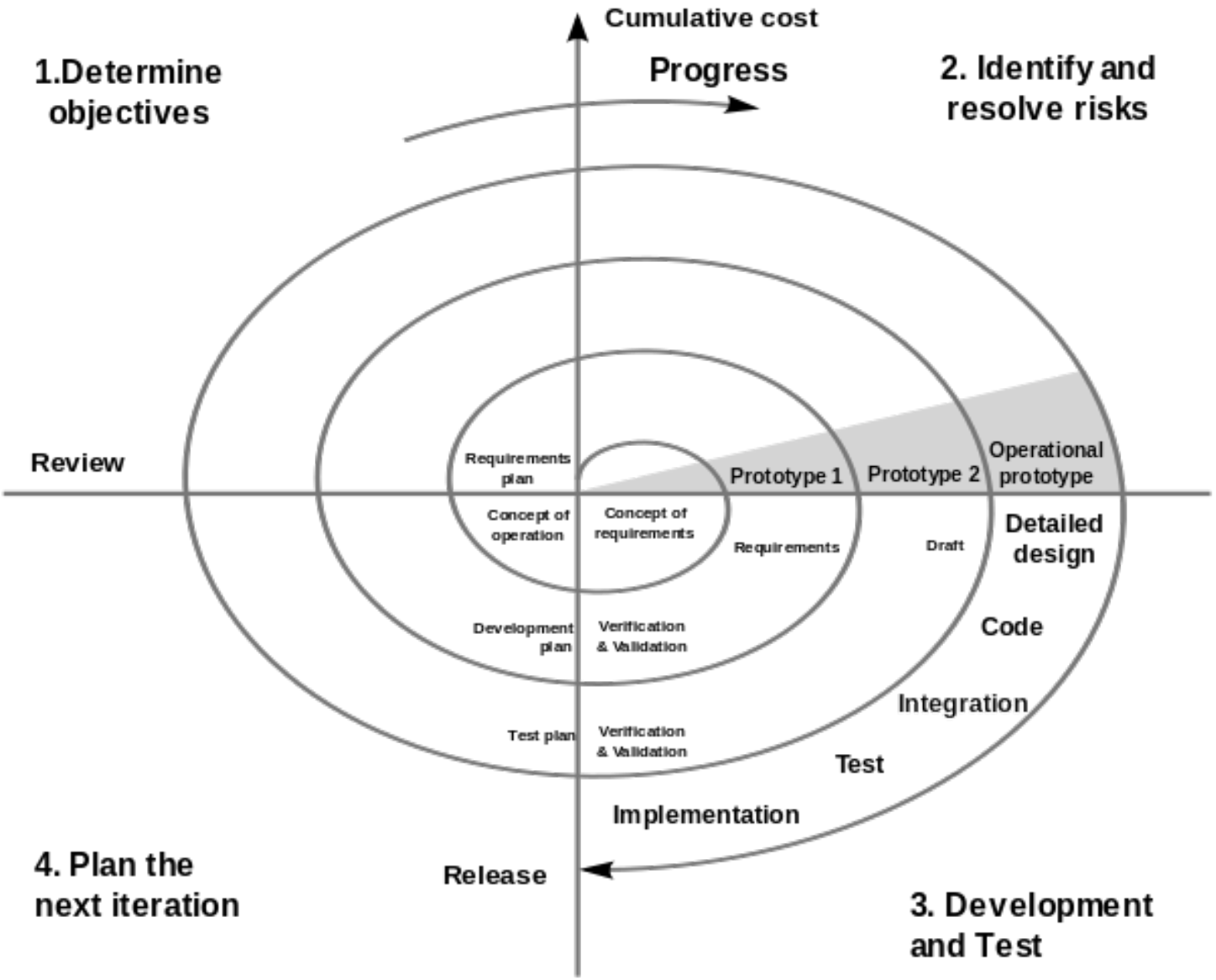

Concept of Operations
Requirements and Architecture
Detailed Design
Implementation
Integration, Test, and Verification
System Verification and Validation
Operation and Maintenance
Verification and Validation
Project Definition
Project Test and Integration
Time

V-model

- An extension of the waterfall model
- More flexible

- “The V-Model reflects a project management view of software development and fits the needs of project managers, accountants and lawyers rather than software developers or users.”

Spiral model



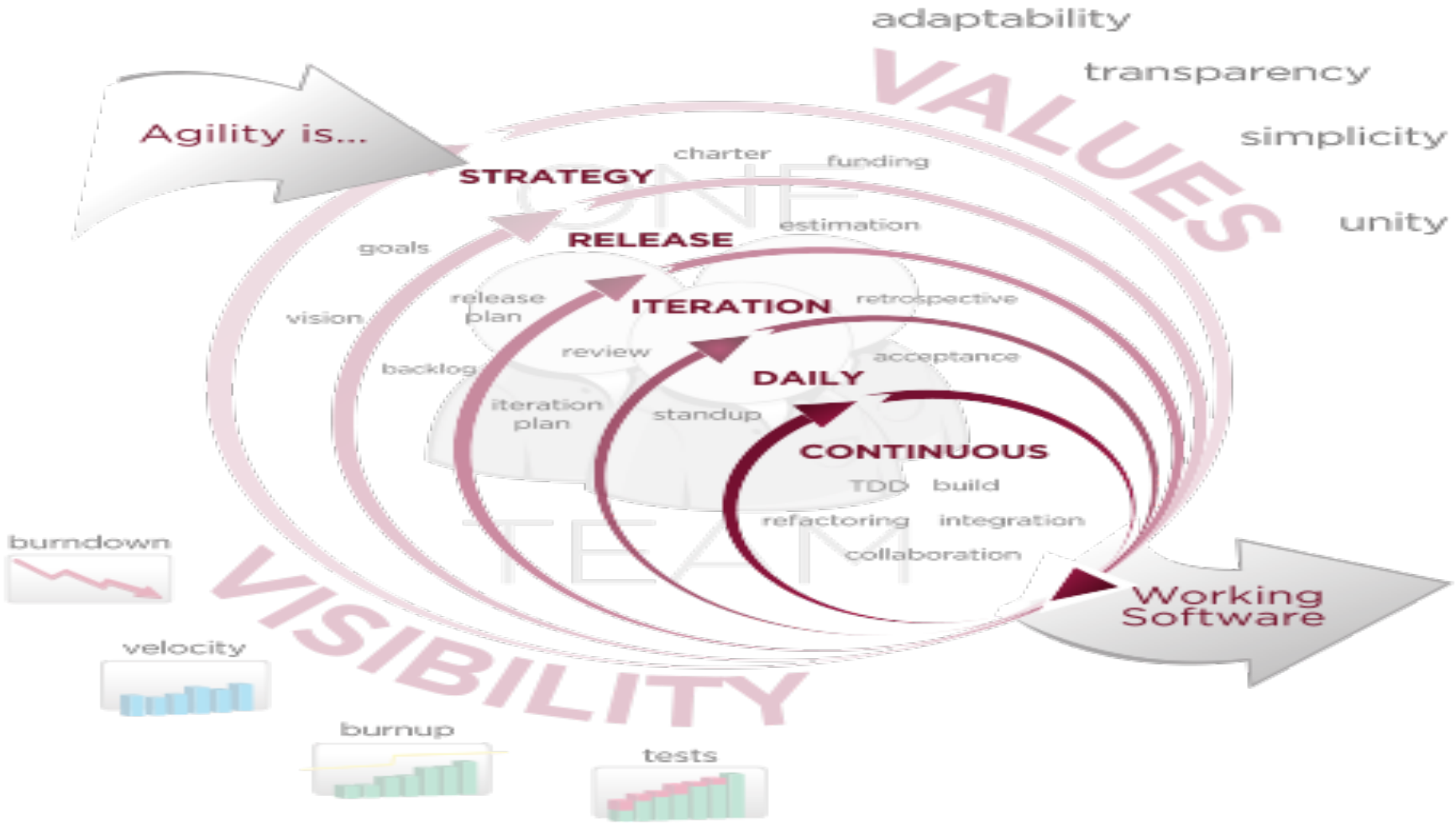
Spiral model

- Some key aspecta from Waterfall and RAD
- Iterative risk analysis
- Suitable for large-scale complex systems

Agile Software Development

Hans-Petter Halvorsen

AGILE DEVELOPMENT



ACCELERATE DELIVERY

Agile

- A group of software development methods
- Iterative approach
- Self-organizing and cross-functional Teams

Examples:

- eXtreme Programming (XP)
- Scrum
- Kanban

Agile

Focus on Programmig

Focus on Process

XP

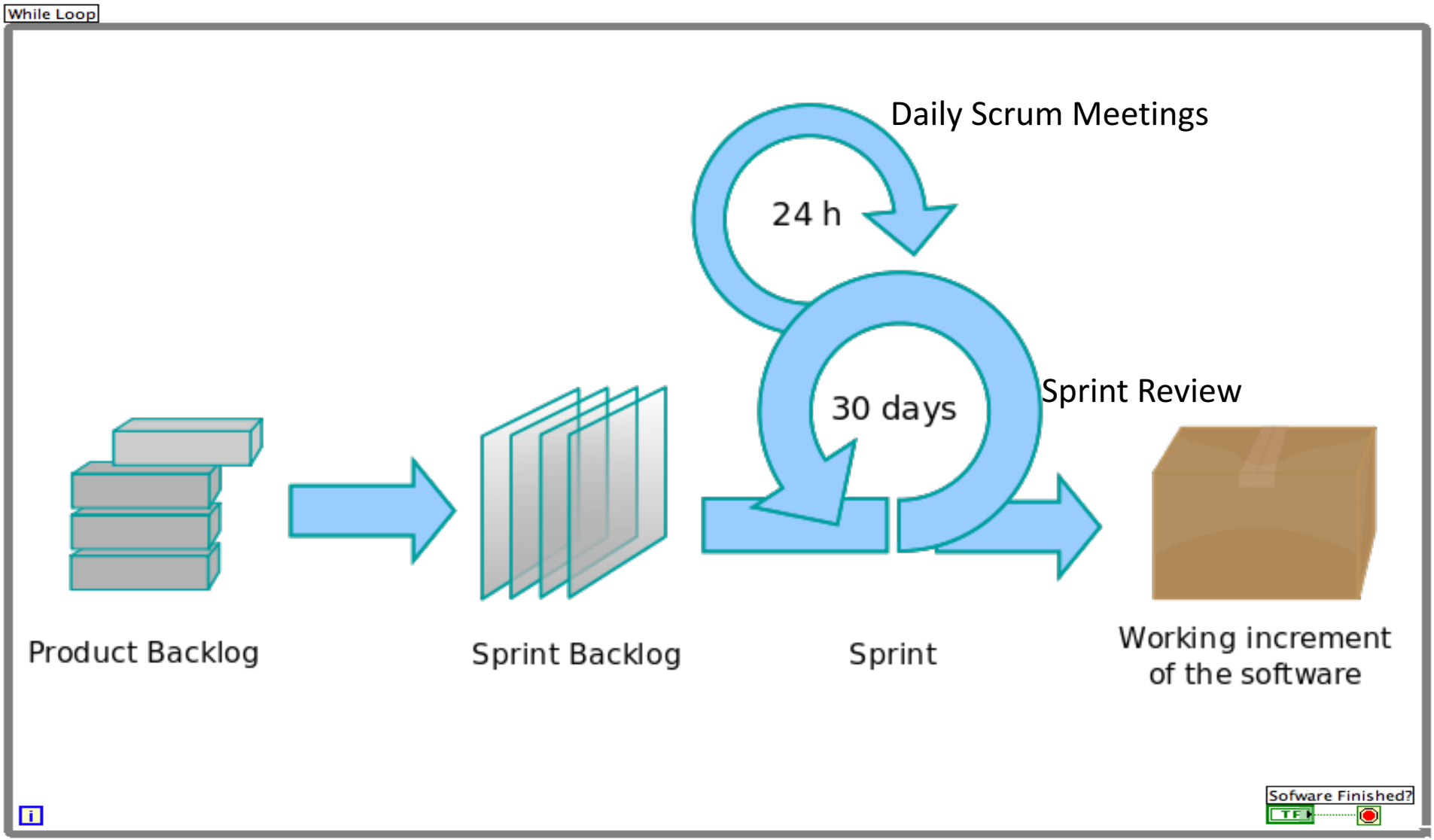
Timebox-based

Scrum

Flow-based

Kanban

Scrum


While Loop
Daily Scrum Meetings
24 h
30 days
Sprint Review
Product Backlog
Sprint Backlog
Sprint
Working increment
of the software
Sofware Finished?
TF

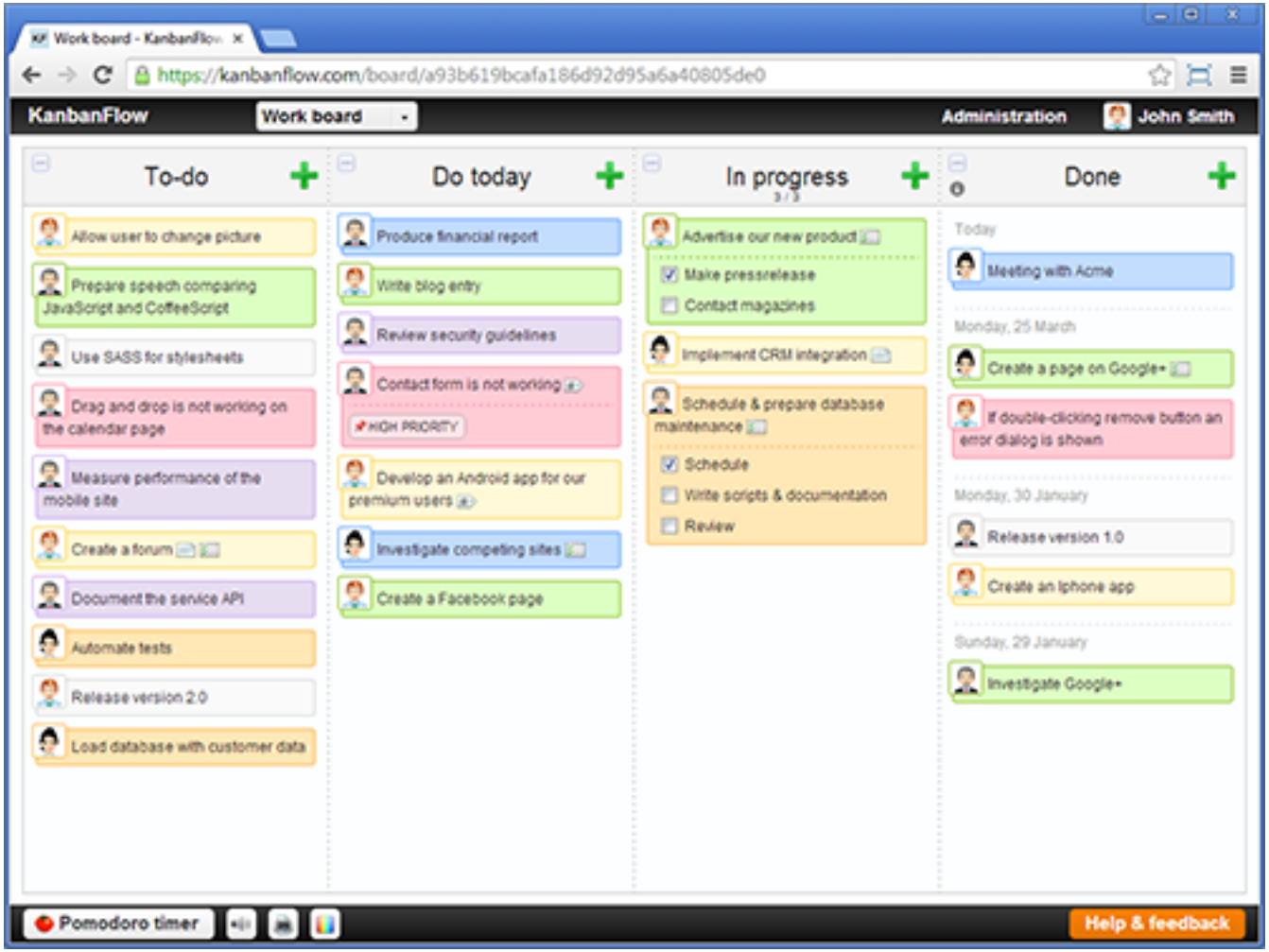
Scrum

- A Framework for Software Development
- Agile Software Development method
- Simple to understand
- Flexible
- Exremely difficult to master!
- Self-organizing Teams (3-9 persons)
- Scrum Team:
  - Product Owner
  - Scrum Master
  - Development Team

Kanban

- Kanban is based on Lean and Toyota production priciples
  - Just-in-Time principle
- Kanban has fewer “rules” than scrum
- Kanban is flow-based, while Scrum is Timebox-based (Sprints)
- Limit WIP (Work in Progress)
- Less focus on Esitmation

Kanban board (Task board)



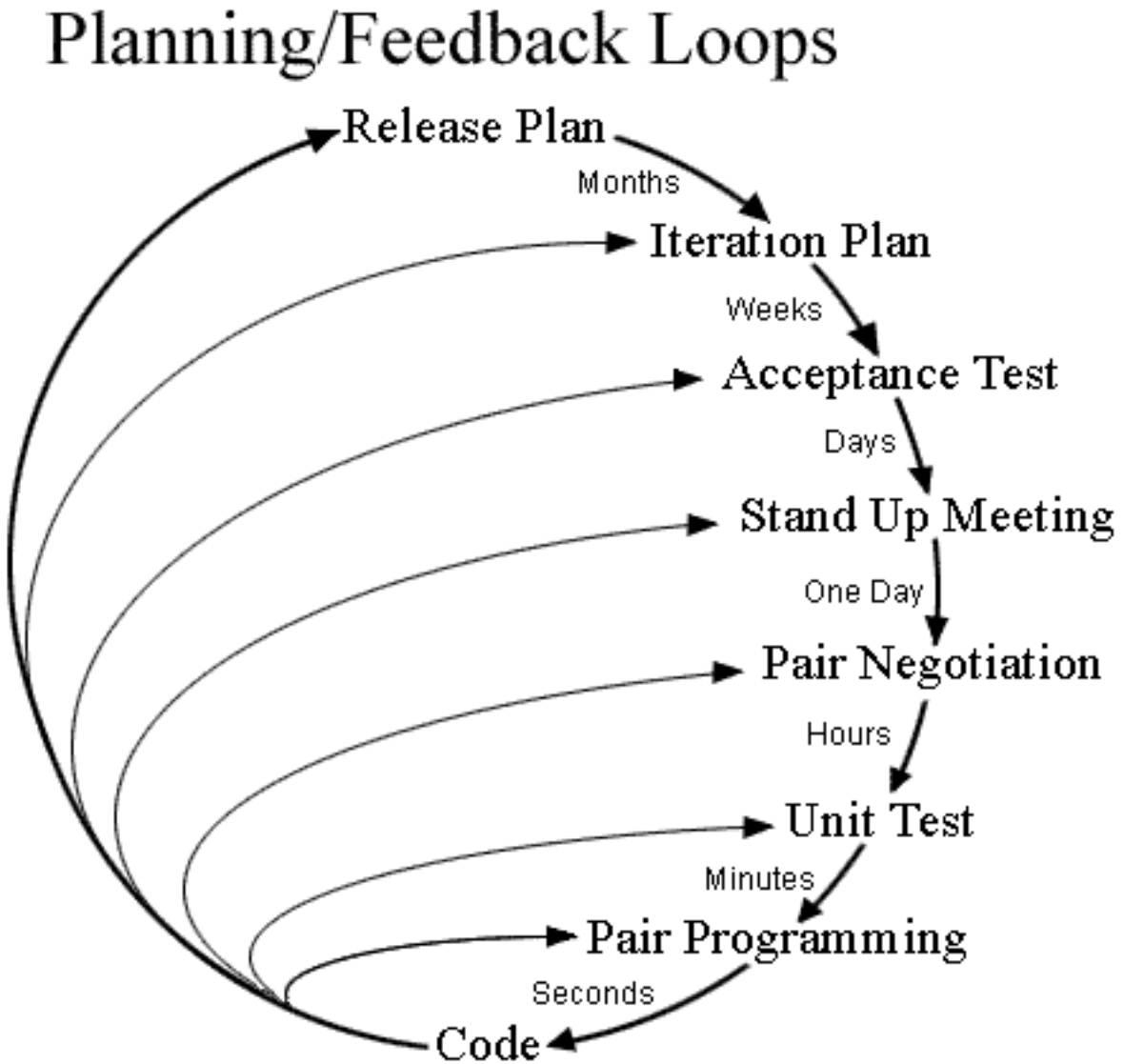
https://kanbanflow.com

Similiar to the Taskboard used in Scrum

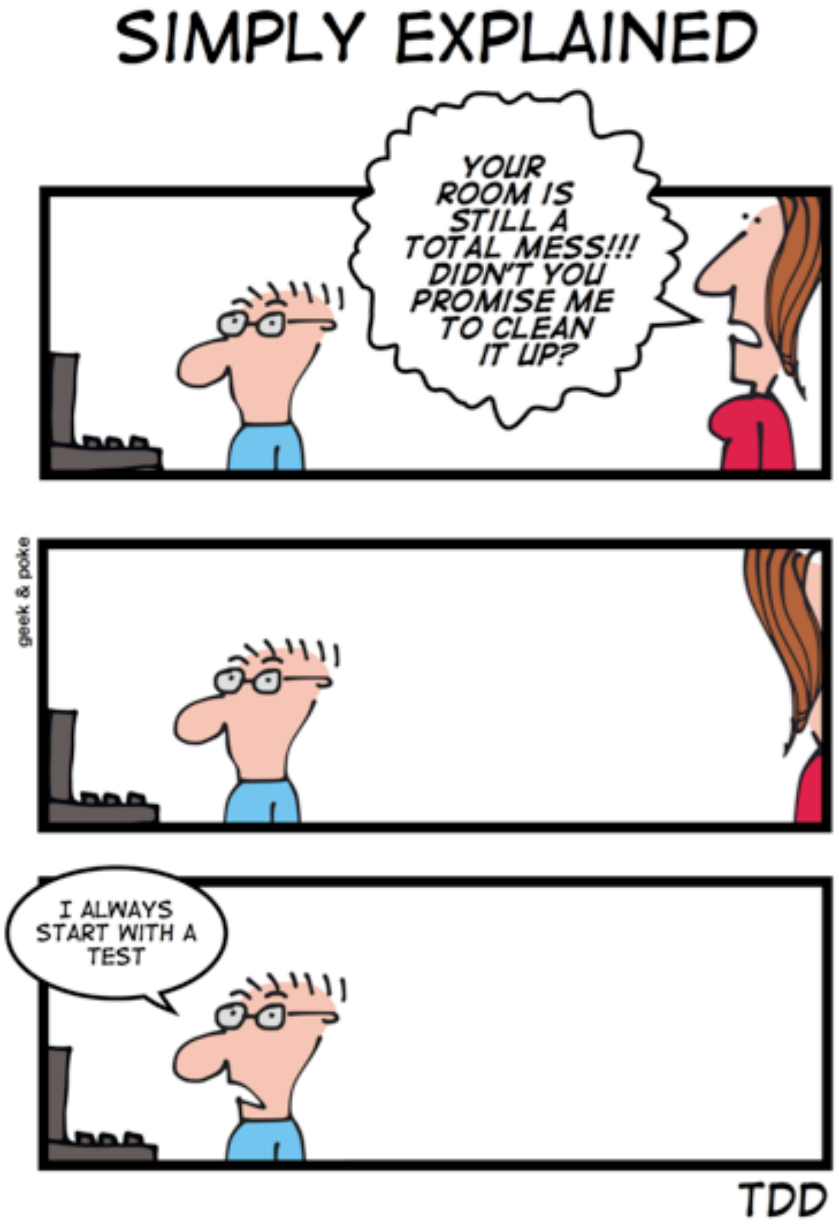
eXreme Programming (XP)

- An Agile method
- Pair Programming
- Code Reviews
- Unit Testing
- Standup Meetings

eXreme Programming (XP)



TDD

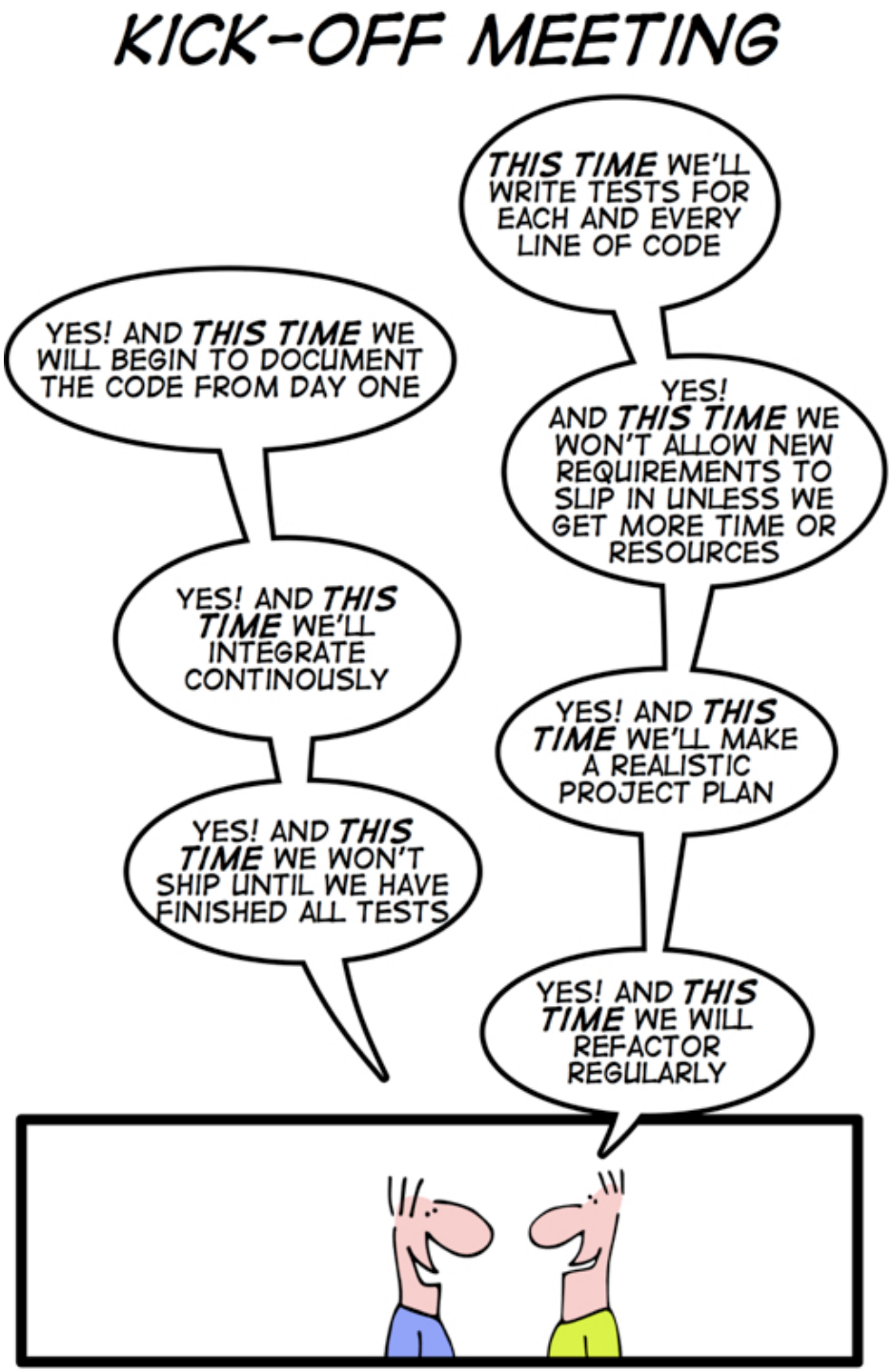

SIMPLY EXPLAINED
YOUR ROOM IS STILL A TOTAL MESS!!! DIDN'T YOU PROMISE ME TO CLEAN IT UP?
geek & poke
I ALWAYS START WITH A TEST
TDD

Finally…

"There are no right or wrong software processes" Ian Sommerville

But someone may be better than other in a given situation

It is better to use a software development metod than using none



Summary

Learn from your previous mistakes!

References

- I. Sommerville, *Software Engineering*: Pearson, 2010.
- Wikipedia. (2013). *Scrum Development*. Available: http://en.wikipedia.org/wiki/Scrum_(development)
- Wikipedia. (2013). *Agile Software Development*. Available: http://en.wikipedia.org/wiki/Agile_software_development
- CoreTrek. (2013). *Scrum i et nøtteskall*. Available: http://www.coretrek.no/scrum-i-et-noetteskall/category642.html
- S. Adams. *Dilbert*. Available: http://dilbert.com
- O. Widder. (2013). *geek&poke*. Available: http://geek-and-poke.com
- B. Lund. (2013). *Lunch*. Available: http://www.lunchstriper.no, http://www.dagbladet.no/tegneserie/lunch/

Hans-Petter Halvorsen, M.Sc.

University College of Southeast Norway

www.usn.no

E-mail: hans.p.halvorsen@hit.no

Blog: http://home.hit.no/~hansha/