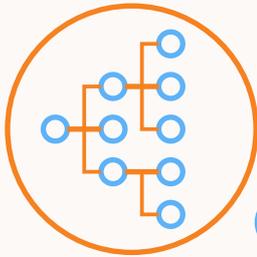


A Brief History of Software Development Methodologies



STRUCTURED PROGRAMMING

1950s

IMPROVES

- Development time
- Clarity
- Quality

BY EXTENSIVE USE OF

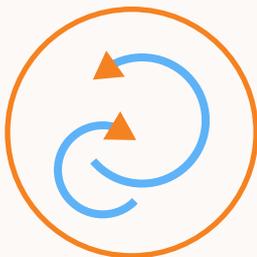
- Block Structures
- Subroutine
- FOR and WHILE loops

1960s

WATERFALL



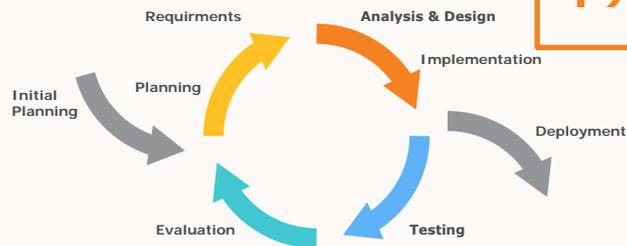
is a sequential (noniterative) process which is seen as flowing steadily downwards through the phases



ITERATIVE & INCREMENTAL

1970s

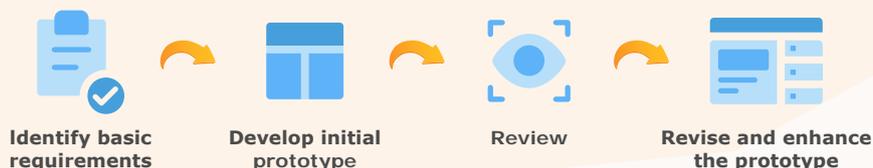
idea is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental), allowing to gain knowledge during development of earlier versions



early 1980s

PROTOTYPING

is the activity of creating prototypes of software applications, i.e., incomplete versions for users to evaluate the design



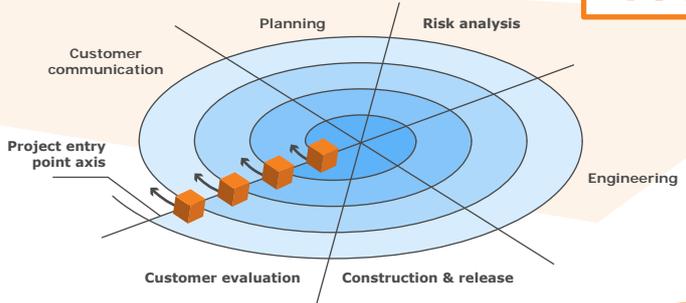


SPIRAL

is a risk-driven process model generator for software projects

late 1980s

- Product maintenance project
- Product enhancement project
- New product development project
- Concept development project



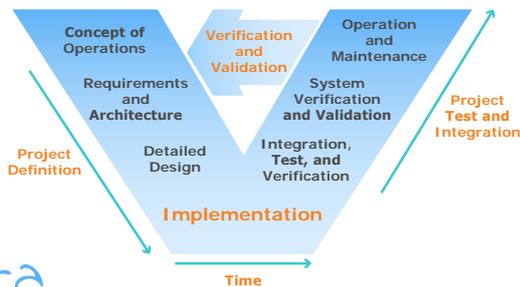
late 1980s

V-MODEL



is an extension of the waterfall model.

The process steps are bent upwards after the coding phase, to form the typical V shape



waterfall era ENDS

ADAPTIVE

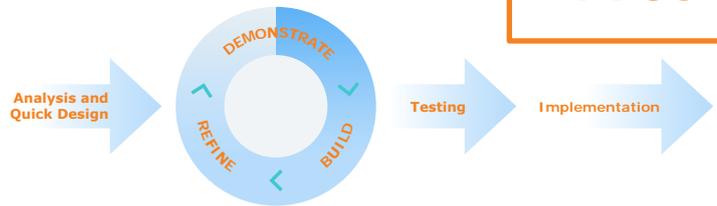
start of AGILE era



RAPID APPLICATION DEV

put less emphasis on planning and more emphasis on process, adaptability and the necessity of adjusting requirements

1990s



1990-2000s

AGILE METHODS RISE

They implement adaptive planning, evolutionary development, early delivery, and continuous improvement, and it encourages rapid and flexible response to change



DYNAMIC SYSTEM DEVELOPMENT METHOD

UNIFIED PROCESS



CHARACTERISTICS

1. Iterable and incremental
2. Architecture-centric
3. Risk-focused



CORE TECHNICS

1. Timeboxing
2. MoSCoW
3. Prototyping
4. Testing
5. Workshop
6. Modeling
7. Configuration
8. Management

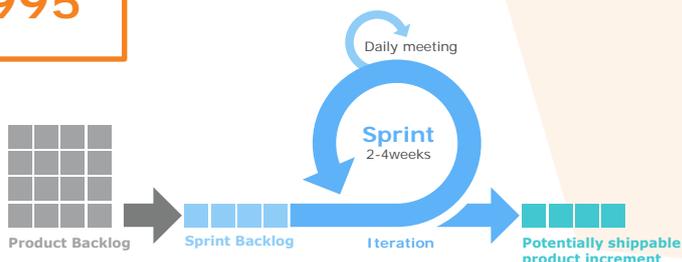
1994

1995

SCRUM



enables teams to self-organize by encouraging physical co-location or close online collaboration of all team members, as well as daily face-to-face communication



EXTREME PROGRAMING (XP)



advocates frequent "releases" in short development cycles, introducing checkpoints for requirements adoption



CRYSTAL

is lightweight, adaptable approach with specific tailored set of policies, practices, and processes based on unique characteristics

1996

1997

FEATURE DRIVEN DEVELOPMENT

FDD's purpose is to deliver tangible, working software repeatedly in a timely manner.



MANIFESTO

for Agile Software Development

2001

Through This Work We Have Come to Value



MANIFESTO!



2000s



AGILE UNIFIED PROCESS

applies agile techniques including test-driven development (TDD), agile modeling (AM), agile change management, and database refactoring to improve productivity



DISCIPLINED AGILE DELIVERY

is a process decision framework that enables simplified process decisions around incremental and iterative solution delivery

SCALED AGILE FRAMEWORK (SAFe)



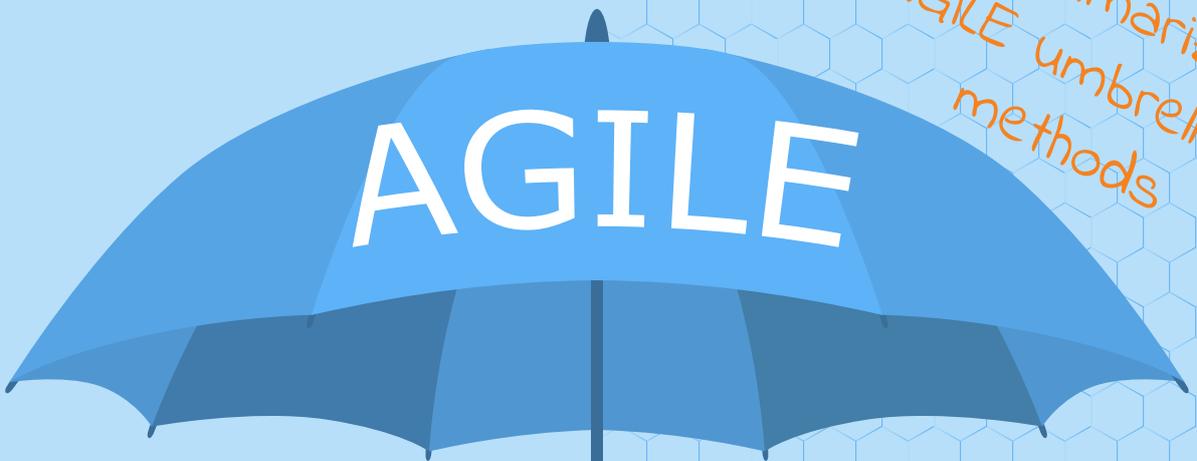
is a framework consisting of a knowledge-base of integrated patterns intended for enterprise-scale Lean-Agile development



LARGE SCALE SCRUM (LeSS)

is regular Scrum applied to large-scale development. For example, for one product group with 500 people

2010s



Summarizing
AGILE umbrella of
methods

- Scrum
- Lean Software Development
- Kanban (process + method)
- Extreme Programming (**XP**)
- Continuous Integration (**CI**)
- Continuous Delivery (**CD**)
- Feature Driven Development (**FDD**)
- Test Driven Development (**TDD**)
- Crystal Clear
- ...

- Scrum-of-Scrums
- Scrum at Scale (**Scrum@ScaIe**)
- Large-Scale Scrum (**LeSS**)
- Scaled Agile Framework (**SAFe**)
- Disciplined Agile Delivery (**DAD**)
- Dynamic Systems Development Method (**DSDM**)
- Agile Project Management (**AgilePM**)
- Agile Unified Process (**AUP**)
- Open Unified Process (**OpenUP**)
- ...

Lightweight approaches

Fuller approaches (beyond 1 team)

Predictive Software Engineering (PSE)

2014

is a framework that addresses bottlenecks of custom software products development and reconstructs the reliable approach to delivering software development services.



PSE Concepts

1. Meaningful Customer Care
2. Transparent End-to-End Control
3. Proven Productivity
4. Efficient Distributed Teams
5. Disciplined Agile Delivery Process
6. Measurable Quality Management and Technical Debt Reduction
7. Sound Human Development

2020s

TO BE CONTINUED...

Sources

- [Agile software development](#) →
- [Software development process](#) →
- www.intetics.com →