WP15

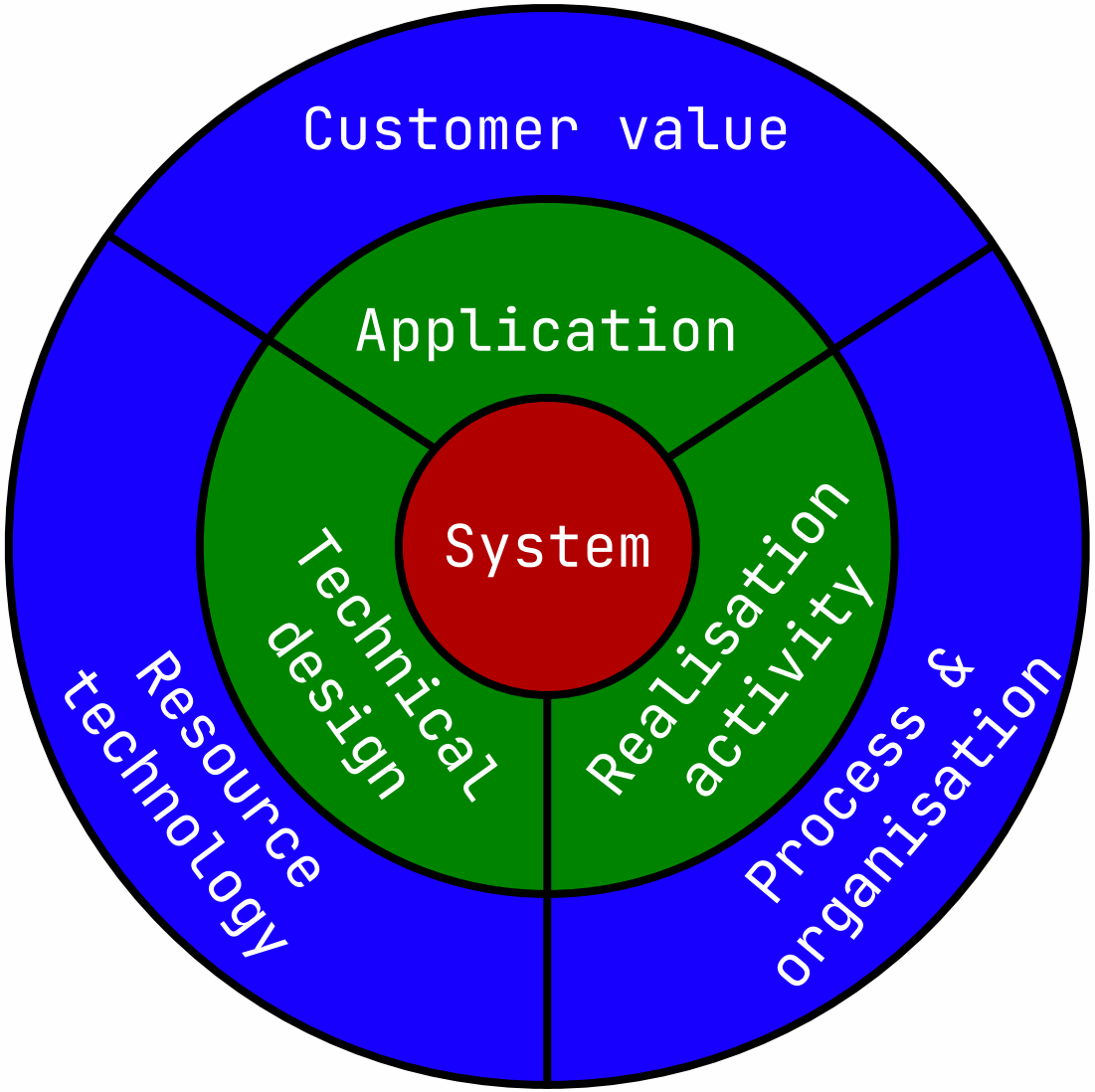
Realisation

Design of Software Architectures

Dr. Vadim Zaytsev aka @grammarware, 12 October 2022

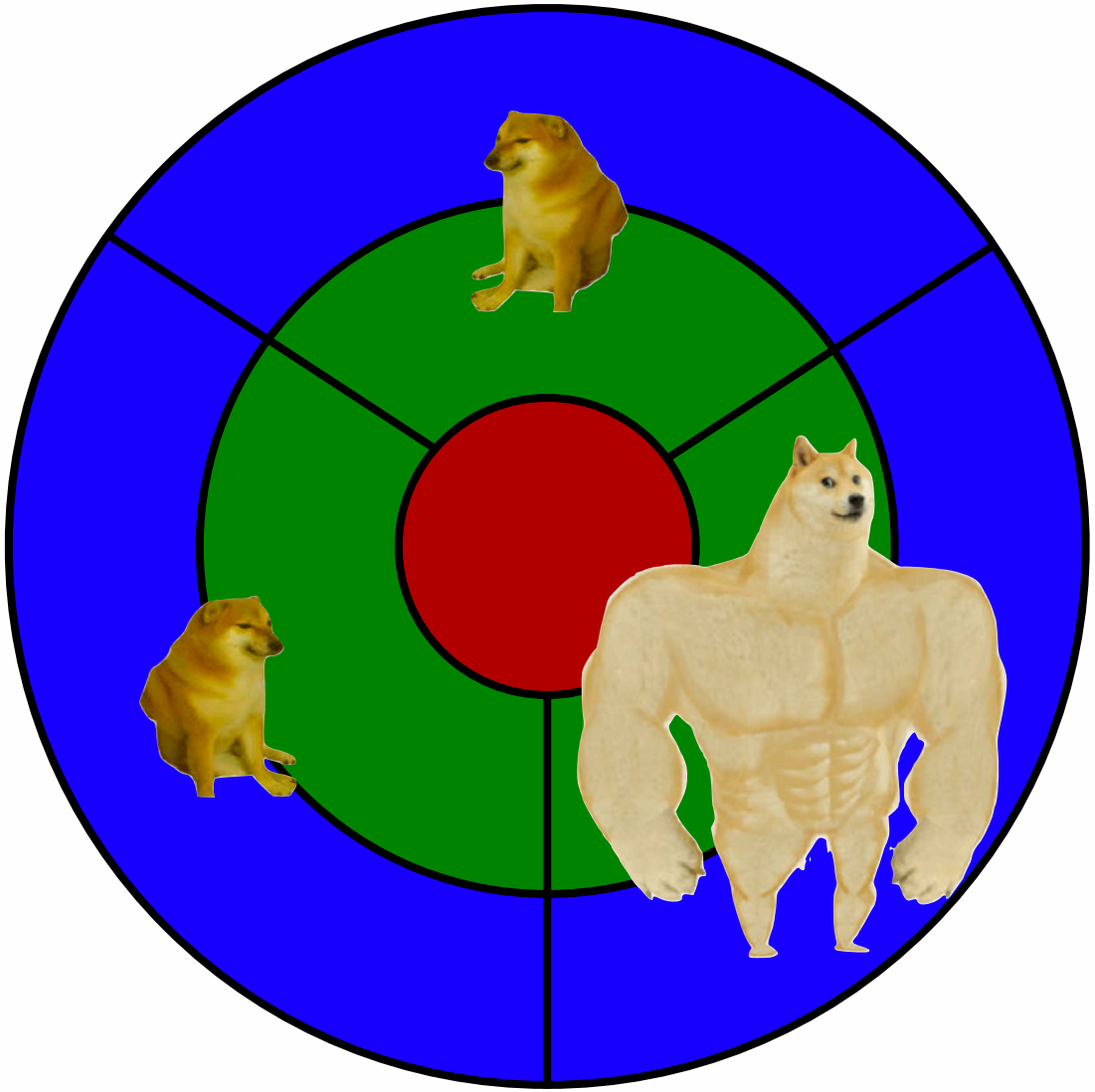
UNIVERSITY OF TWENTE.

Recall WP03



by Robert Deckers, based on ISO 42010 & DYA

Recall WP03



by Robert Deckers, based on ISO 42010 & DYA

Recall WP08: realisation aspects

- Waterfall ⟺ Iterative
- Manual ⟺ Automated
- Buy ⟺ Reuse
- Adapt ⟺ Remake
- Feature first ⟺ Market first
- Develop ⟺ Outsource
- Assign ⟺ Hire
- Test for release ⟺ Test for integration

Why should you architect?

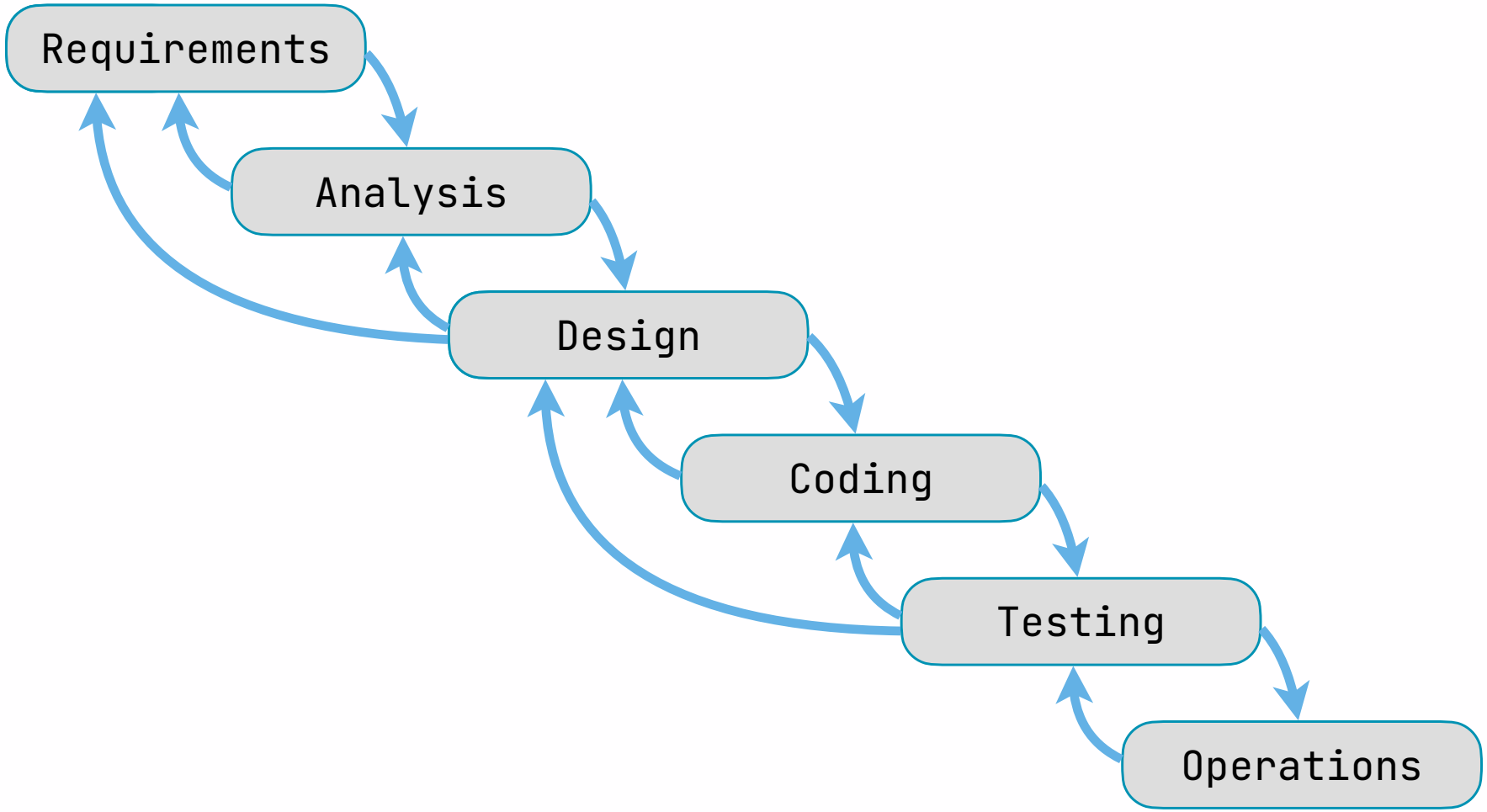
(why provide directions for complicated properties)

- because like different parties have different ideas and they do their own things
- consistency
- to make the project successful
- to define goals
- to connect domains
- providing prioritisation
- taking decisions and weighing the options

What happens if you don’t?

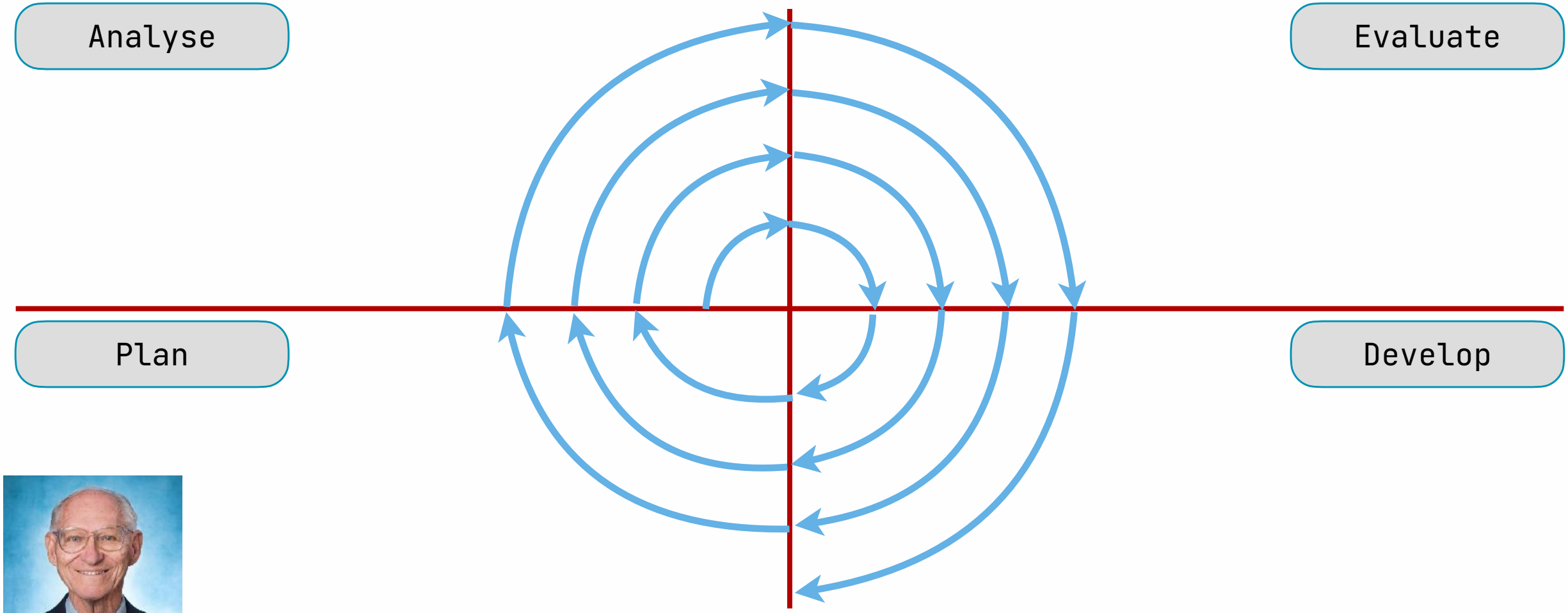
- devs will waste time on irrelevant issues
- misalignment of interests
- goals unclear
- directions unclear
- project becomes unrealisable
- unhappy stakeholders
- product will not meet the right reqs
- becomes a mess
- over budget
- becomes improperly documented

Think of activities



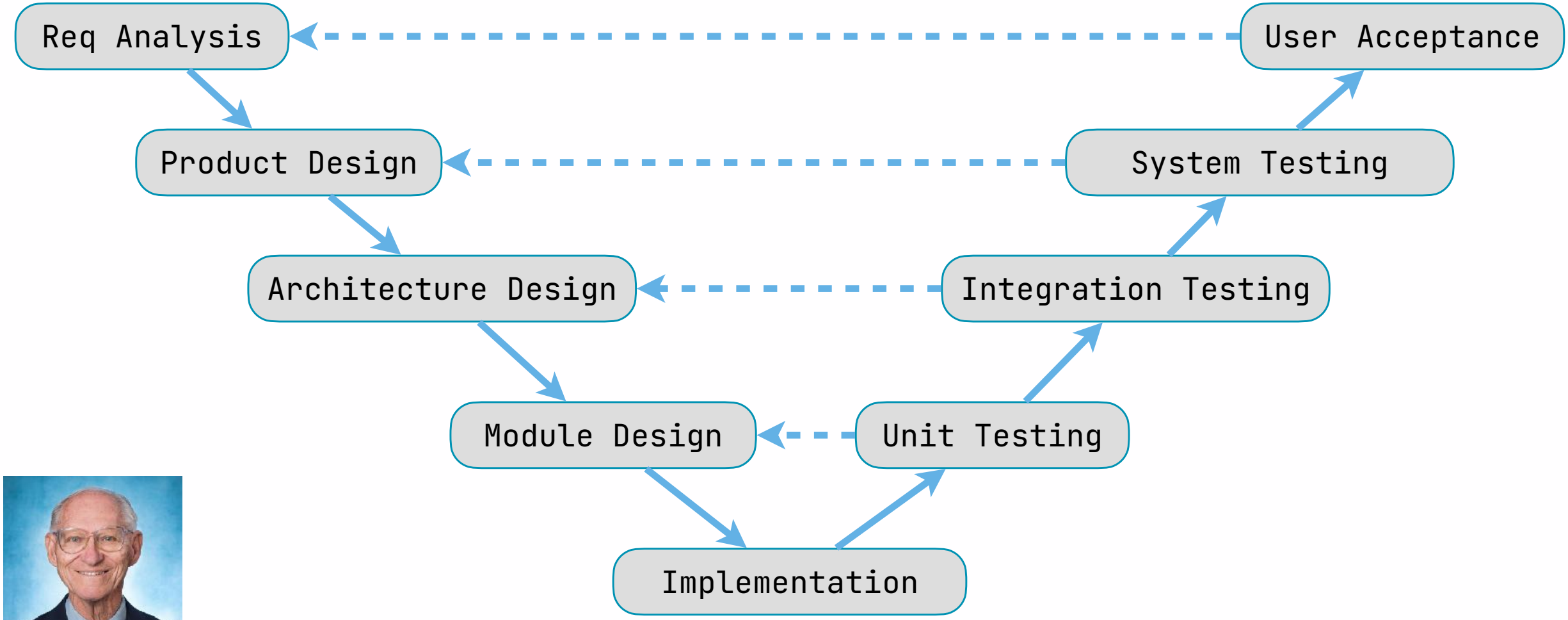
W.W.Royce, Managing the Development of Large Software Systems: Concepts and Techniques, ICSE'87.

Think of activities



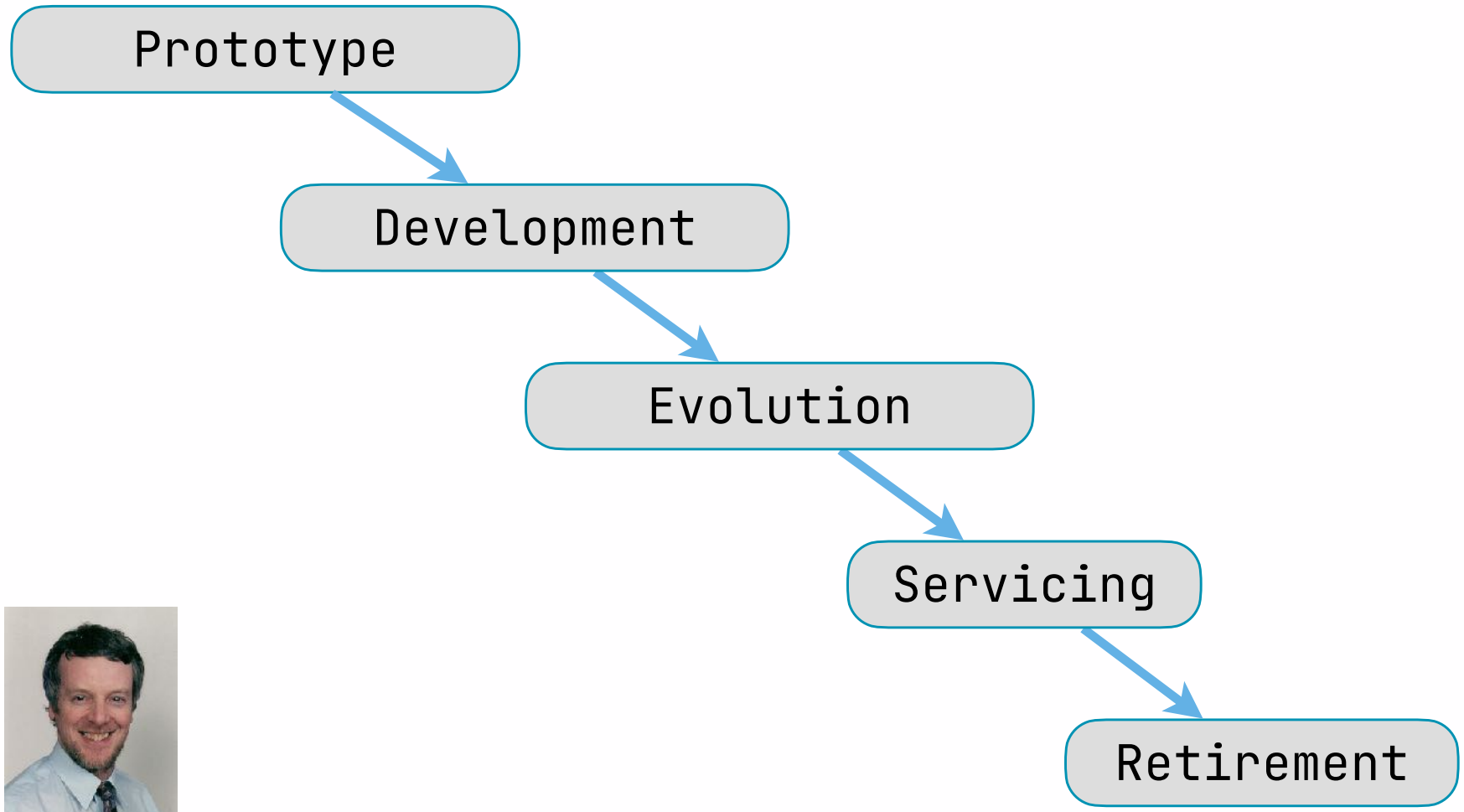
B.Boehm, *A [redacted] model of software development and enhancement*, ACM SIGSOFT SE Notes, 1986.

Think of activities



B.W.Boehm, *Guidelines for verifying and validating software requirements and design specification*, EuroIFIP, 1979.

Think of activities



V.T.Rajlich,K.H.Bennett, *A [redacted] Model for the Software Life Cycle*, Computer 33(7), 2000.

Think of artefacts

- Source code
- Executables, binaries
- Logs, traces
- Tests, models, stories
- Design, diagrams
- Documentation
- Packaging, installation
- Demonstration, MVPs, PoCs, pilots
- Progress, backlog
- Feedback on input

how to organise this?

what process to follow?

Think of constraints

- Stakeholder constraints
- User requirements
- Operations / infrastructure
- Enterprise architecture
- System architecture
- Company guidelines
- Regulations
- Laws

Think of development standards

- Coding standards
- Design patterns
- Implementation patterns
- Templates
- Components
- Platforms
- Libraries
- Tools

Think of systems and people

- Systems
  - Often operate on existing systems
  - Interacting systems
- People
  - Skills and knowledge
  - Preferences and beliefs

Map architecture onto the process

- Activities, artefacts, constraints, standards, systems, people
- What process comes after your architecture document:
  - main steps
  - involved roles (teams, stakeholders)
- How is the architecture used by the development process?
  - Which of your views is used by whom?
  - How is the architecture applied properly?
  - How can you verify if it is?