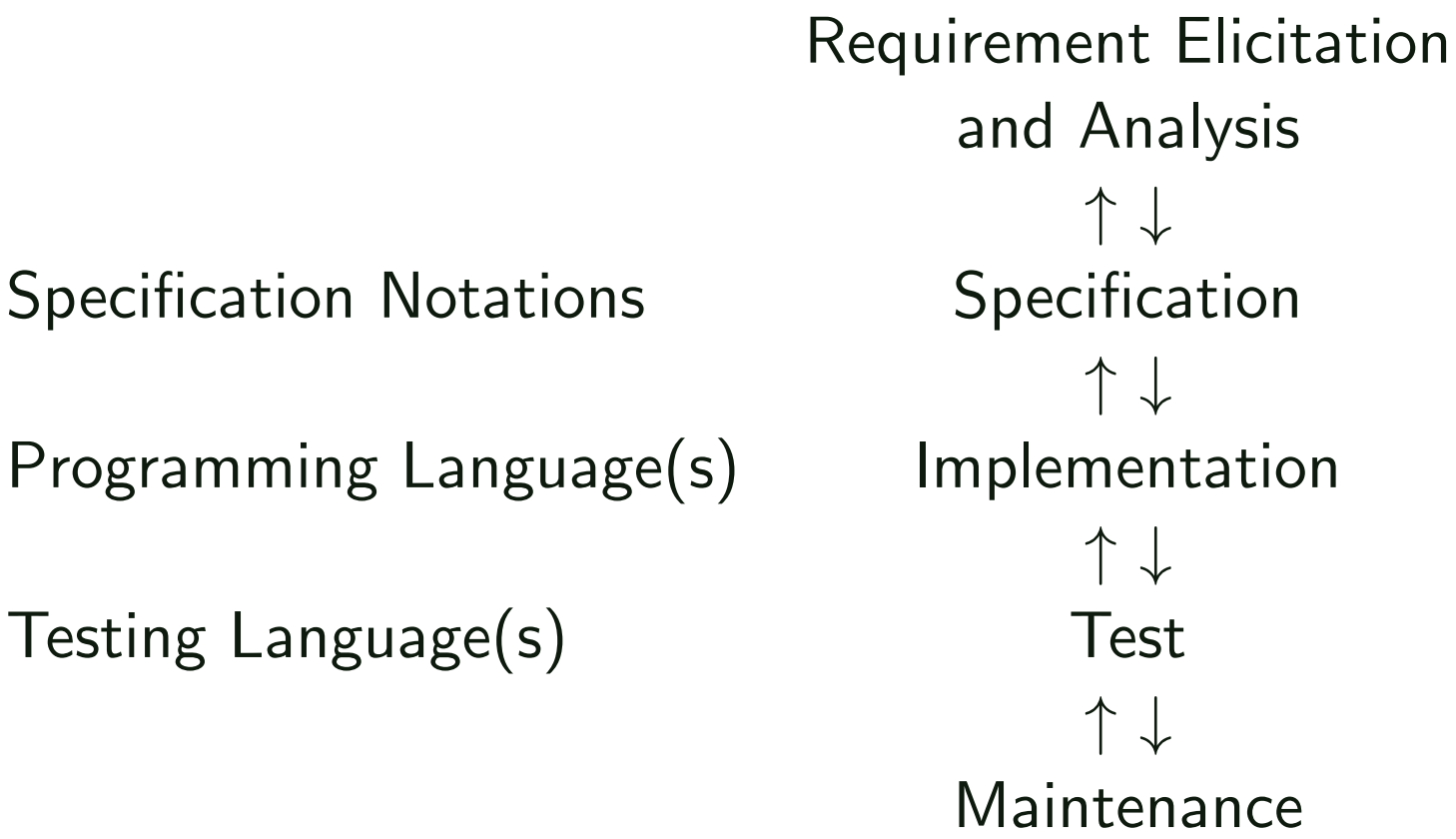
Professional Issues II: Unit 8: Software Engineering *Waterfall, Spiral, and V model*

Markus Roggenbach

April 2018

SDM I: Waterfall model

Phases



Described 1970 by Winston W. Royce.

An extended example applying the Waterfall Model

Ex: super-automatic espresso machine (i)

[De'Longhi "Magnifica S" Super Automatic Espresso Machine]

Ex: super-automatic espresso machine (ii)

Requirements:

4.11 Pump Control

4.11.1 Cup size adjustment

With SAEM-3000 machines it is easy for the user to control the volume of liquid processed by the pump.

4.11.2 Cup sizes of "basic" machines

"Basic" SAEM-3000 machines can serve 'small' and 'large' cup sizes.

4.11.2.1 Selecting cup size in "basic" machines

The user can select a 'small' or 'large' coffee by simply pressing a button.

4.11.2.1.1 Regional setting of cup size in "basic" machines

Depending on the regional code, a 'small' cup holds

- 0.15l in Europe and Asia,
- 0.2l in the Americas
- 0.16l in other regions

By selecting the 'large' option, this amount is increased by 80%.

4.11.3 Cup size of "commercial" machines

In the "Commercial" line of SAEM-3000, the cup size can be programmed by the owner to fit the individual size of cups in the establishment.

...

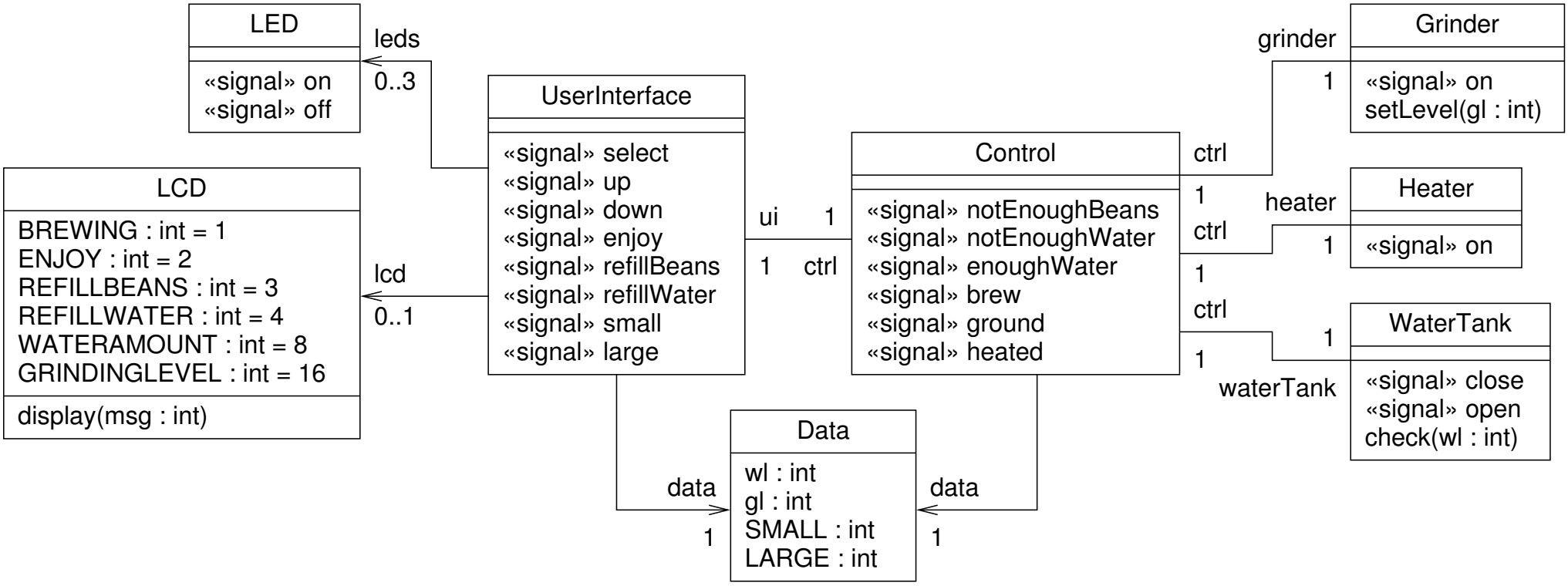
4.12.2.3.1 Tank empty during brewing

If during coffee preparation the water supply is empty, a warning is issued to the user and the machine resumes its idle state.

[fictious example as presented in Knapp, Roggenbach, Schlingloff '15]

Ex: super-automatic espresso machine (iii)

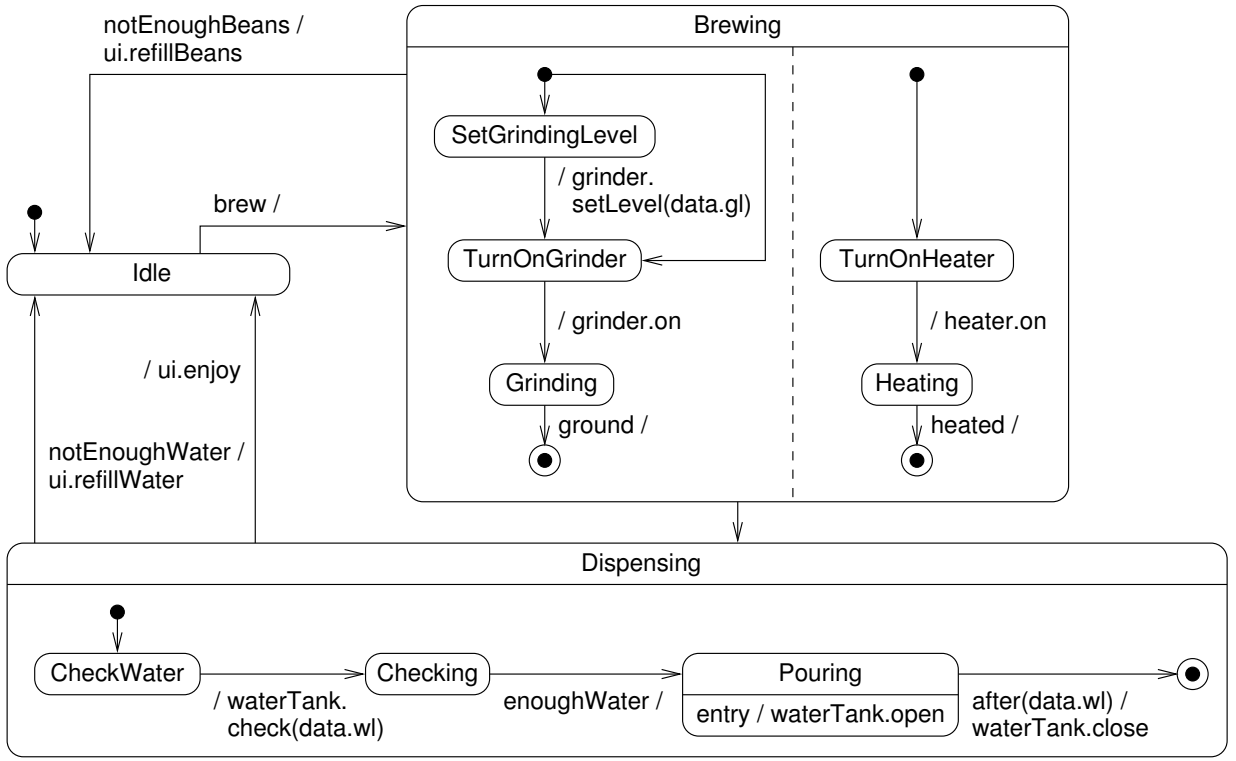
Specification: static structure (UML class diagram)



[fictious example as presented in Knapp, Roggenbach, Schlingloff '15]

Ex: super-automatic espresso machine (iv)

Specification: controller (UML state machine diagram)



[fictious example as presented in Knapp, Roggenbach, Schlingloff '15]

Ex: super-automatic espresso machine (v)

Implementation in Java: part of the static structure

```
class CoffeeMachineData {
    int wl;
    int gl;
    int SMALL;
    int LARGE;
}
```

Ex: super-automatic espresso machine (vi)

Implementation in Java: part of the Controller

```
interface State {
    public State next();
}

enum States implements State {
    idle {
        @Override
        public State next(Input word) {
            switch(word.read()) {
                case 'brew': return Brewing;
                default: return idle;
            }
        }
    },
  Brewing{ ..
```

Ex: super-automatic espresso machine (vii)

Testing: Test cases for the Test Objective

"Upon pressing of the select/small/large button, the machine will automatically brew coffee."

$$T1 = \langle \mathsf{occ}(tester, \mathsf{small}, ui), \mathsf{disp}(tester, \mathsf{small}, ui), \mathsf{occ}(ui, \mathsf{brew}, ctrl) \rangle$$
$$T2 = \langle \mathsf{occ}(tester, \mathsf{large}, ui), \mathsf{disp}(tester, \mathsf{large}, ui), \mathsf{occ}(ui, \mathsf{brew}, ctrl) \rangle$$
$$T3 = \langle \mathsf{occ}(tester, \mathsf{select}, ui), \mathsf{disp}(tester, \mathsf{select}, ui), \mathsf{occ}(ui, \mathsf{brew}, ctrl) \rangle$$

[fictious example as presented in Knapp, Roggenbach, Schlingloff '15]

Documents required by the Waterfall Model

Documents produced in Design

- System Architecture,
- User Handbook, and
- Verification Plan.

Documents produced in Implementation and Debugging

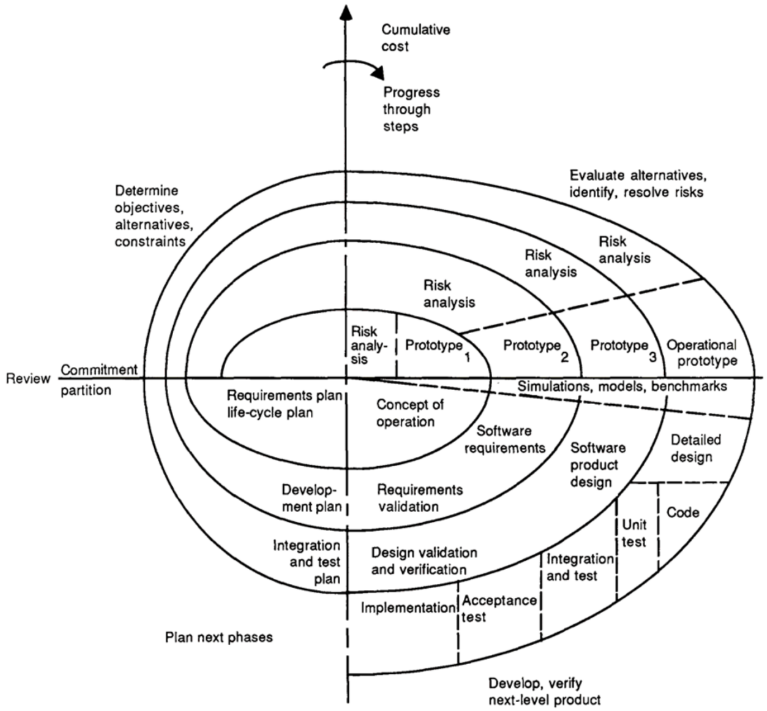
- Implemented Modules and their documentation
- Test data

Documents produced in Testing

- Running System and
- Evaluation of the software project.

Further Classical Software Devopment models

Spiral model



Defined by Barry Boehm in his 1988 article "A Spiral Model of Software Development and Enhancement".

How does the spiral ever get started?

The spiral gets started by a hypothesis that a particular operational mission (or set of missions) could be improved by a software effort.

Berry Boehm in: A Spiral Model of Software Development and Enhancement, IEEE Computer, IEEE, 21(5):61-72, May 1988. http://csse.usc.edu/TECHRPTS/1988/usccse88-500/usccse88-500.pdf

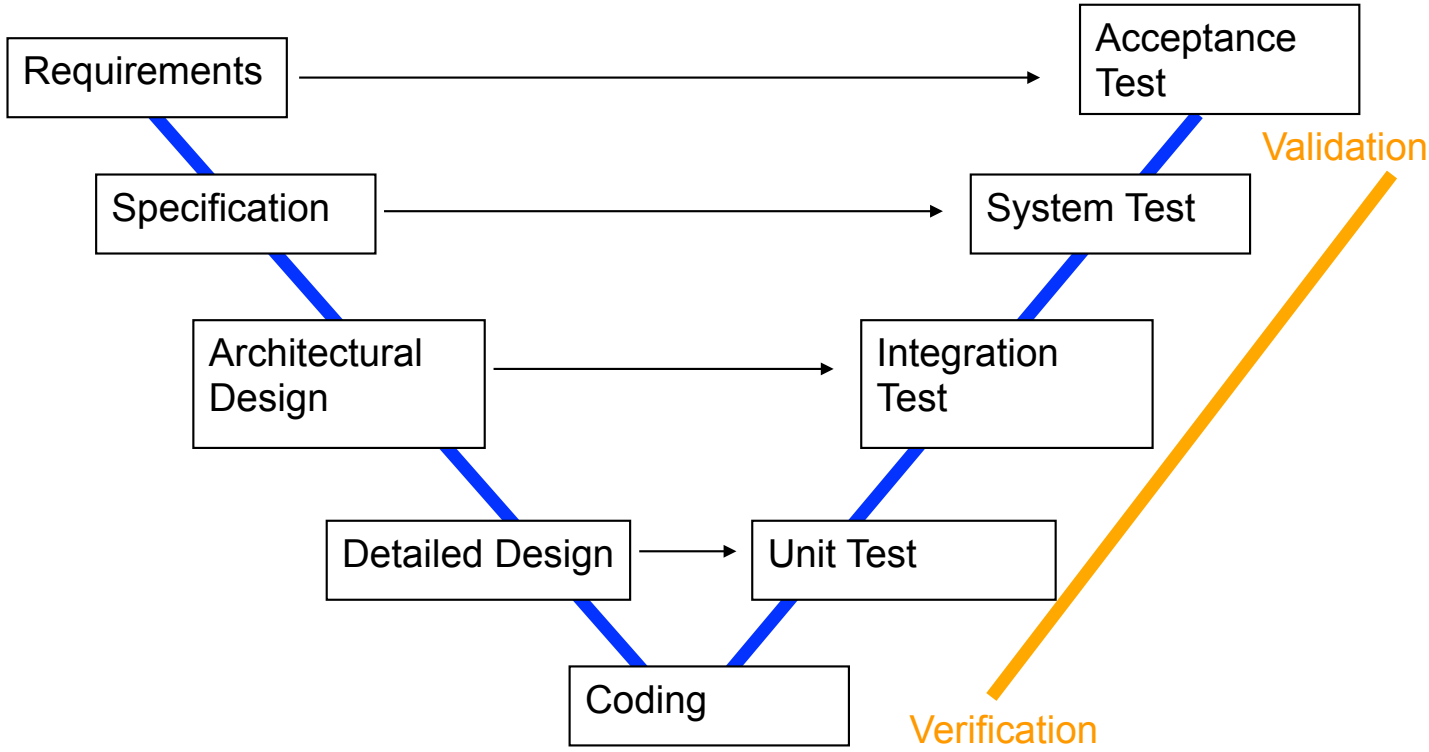
How do you get off the spiral when it is appropriate to terminate a project early?

The spiral process then involves a test of this hypothesis: at any time, if the hypothesis fails the test (for example, if delays cause a software product to miss its market window, or if a superior commercial product becomes available), the spirat is terminated. Otherwise, it terminates with the installation of new on modified soft-ware, and the hypothesis is tested by observing the effect on the operational mission.

Berry Boehm in: A Spiral Model of Software Development and Enhancement, IEEE Computer, IEEE, 21(5):61-72, May 1988.

http://csse.usc.edu/TECHRPTS/1988/usccse88-500/usccse88-500.pdf

V-Model



Current standard for German federal administration and defence projects, as well as for software developers within in the region.