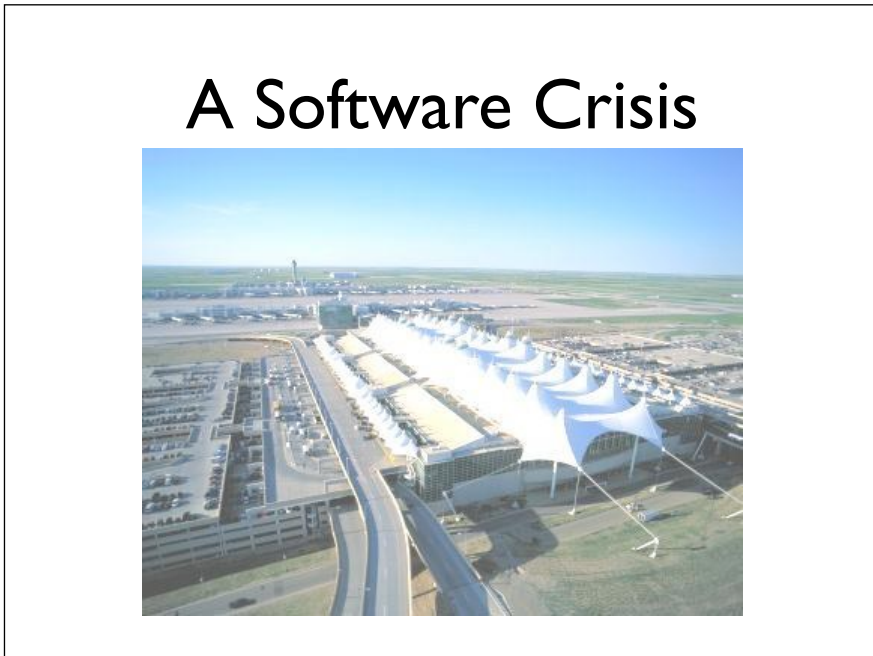


1

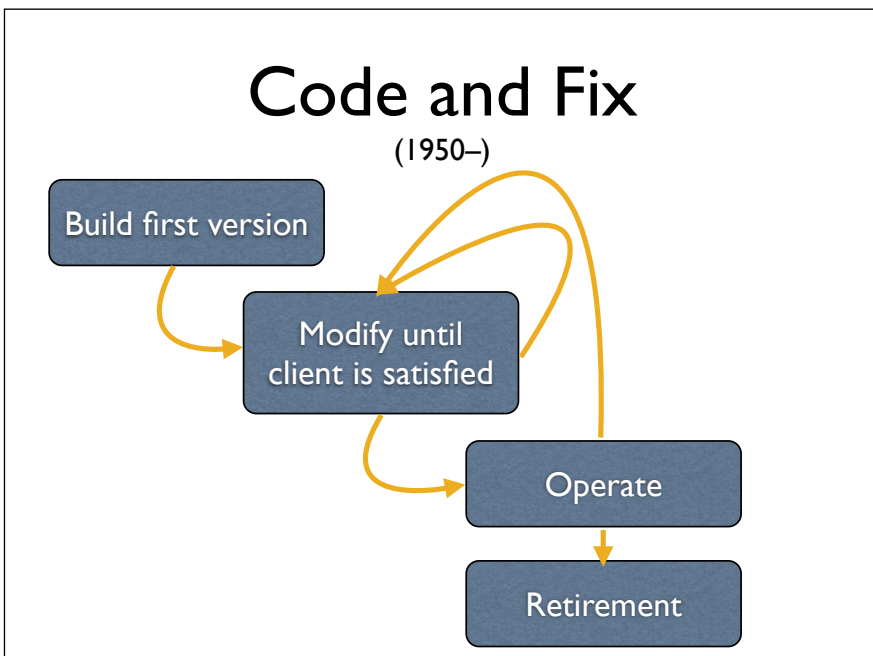


A Software Crisis

Denver International Airport (DIA)

Construction started in 1989 • 53 sq miles
• Planned: 1.7 bio USD costs, opening 1993

2



3

Waterfall Model (1968)

```
...
//output_upper_byte then upper byte
OUTPUT_BYTE(h<<BYTE_SIZE);
OUTPUT_BYTE(l);
OUTPUT_BYTE(w<<BYTE_SIZE);
OUTPUT_BYTE(w);

order1 = order;
order2 = order;

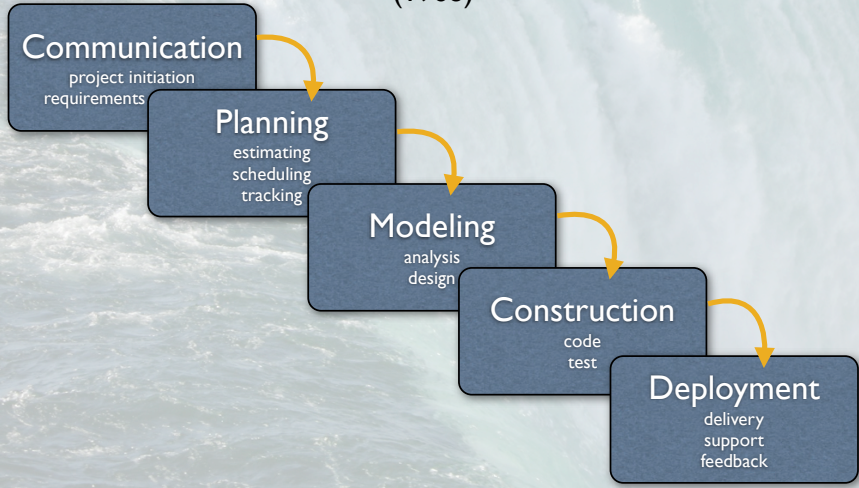
#ifdef TRACE
if (!fpm)
{
    fprintf
    exit(2)
}
#endif

/* allocate 'order+1' elements
... is used to stop
... the f
```

Construction
code
test

13

Waterfall Model (1968)



14

Deployment

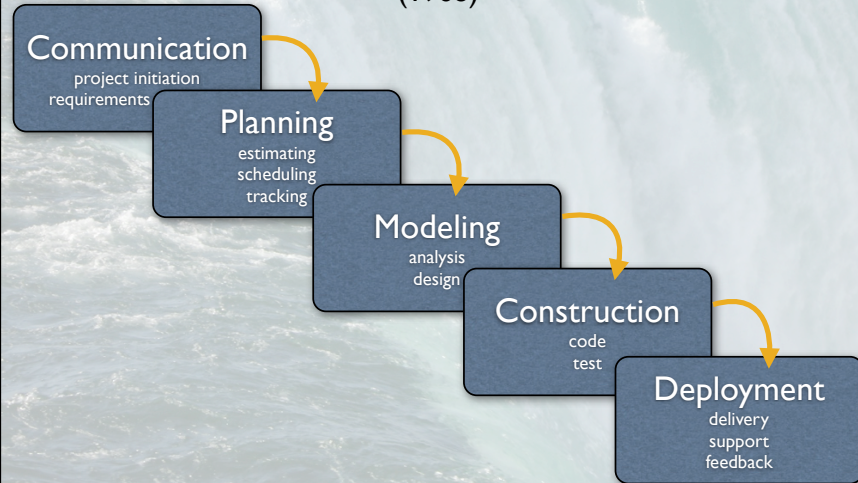


Deployment
delivery
support
feedback

15

Waterfall Model

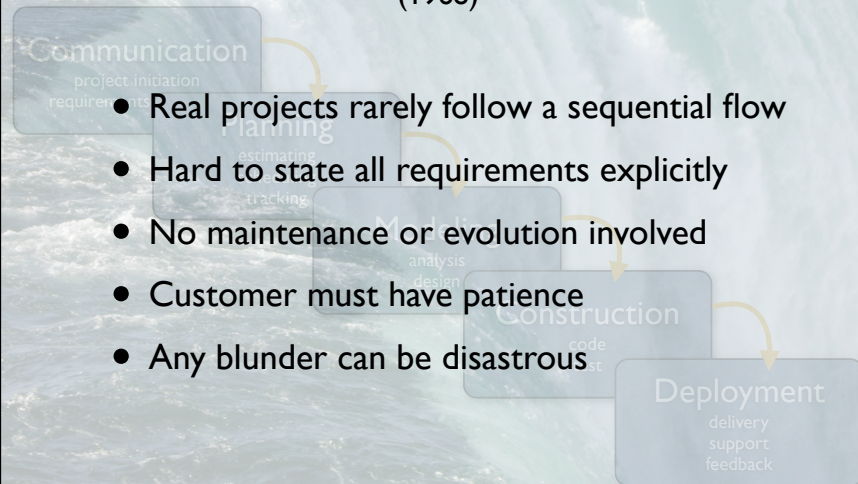
(1968)



16

Waterfall Model

(1968)



17

Boehm's first law

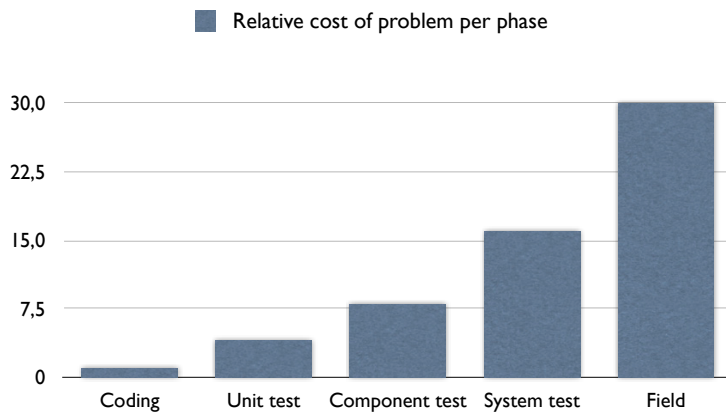
Errors are most frequent during *requirements* and *design* activities and are the more expensive the later they are removed.

This and other laws are found in Endres/Rombach: Handbook of Software and Systems Engineering. Evidence: Several studies before

18

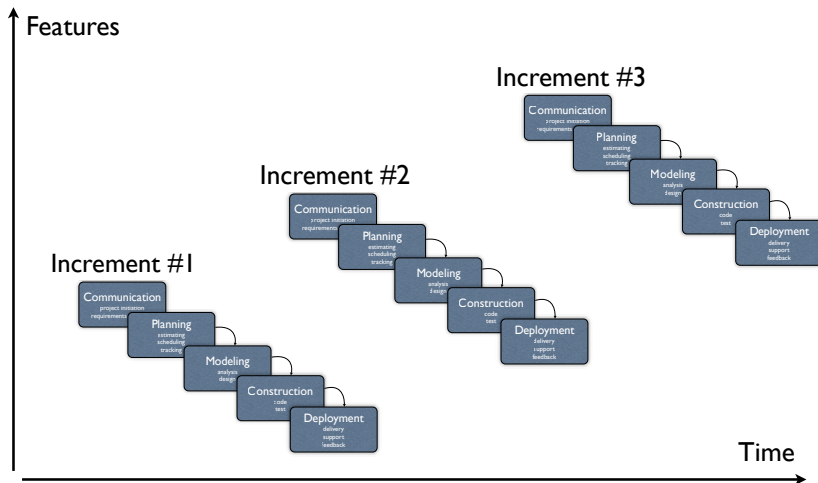
1074

Problem Cost



19

Incremental Model



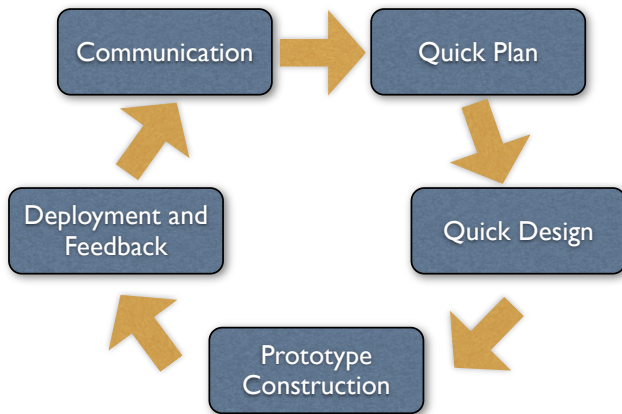
20

Incremental Model

- Each linear sequence produces a particular “increment” to the software
- First increment typically core product; more features added by later increments
- Allows flexible allocation of resources

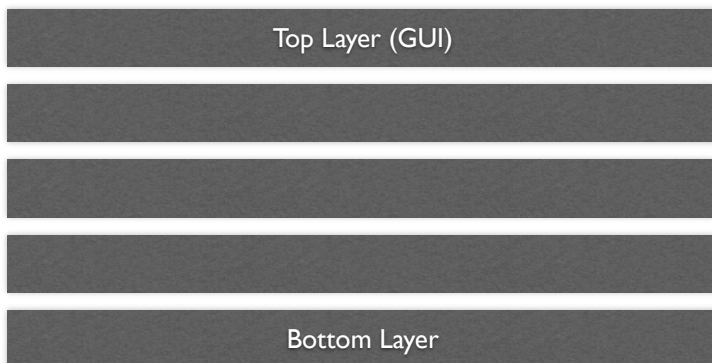
21

Prototyping



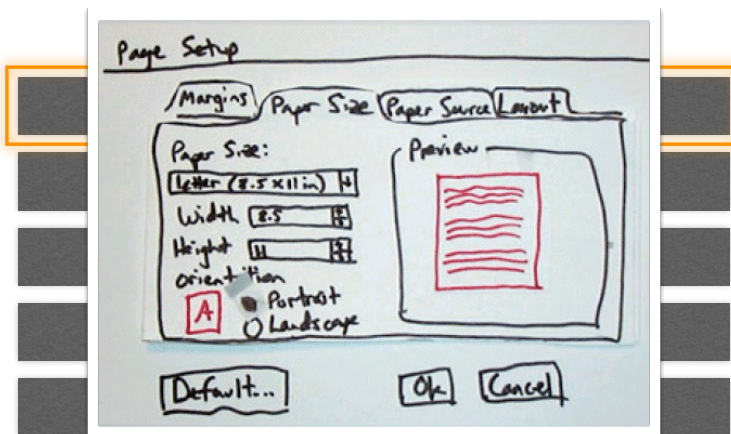
22

Prototypes



23

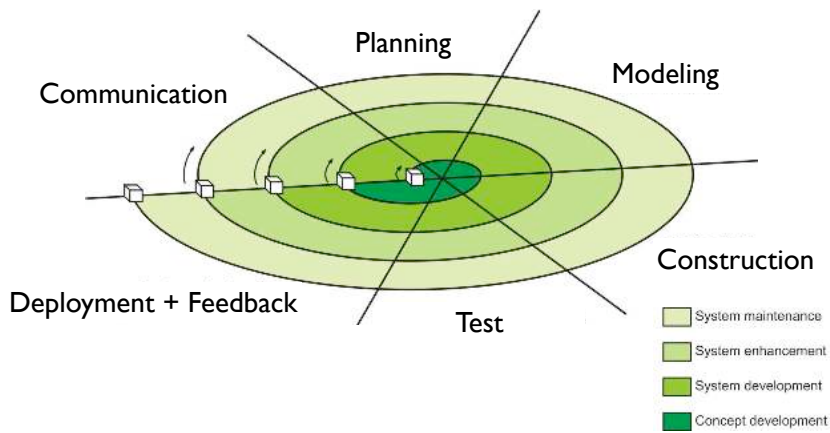
Horizontal Prototype



24

Spiral Model

(1988)



28

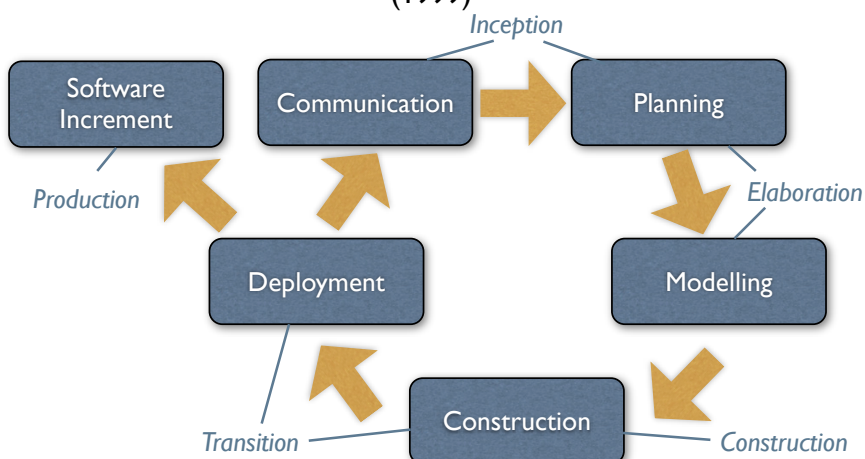
Spiral Model

- System is developed in series of evolutionary releases
- Milestones for each iteration of the spiral
- Process does not end with delivery
- Reflects iterative nature of development

29

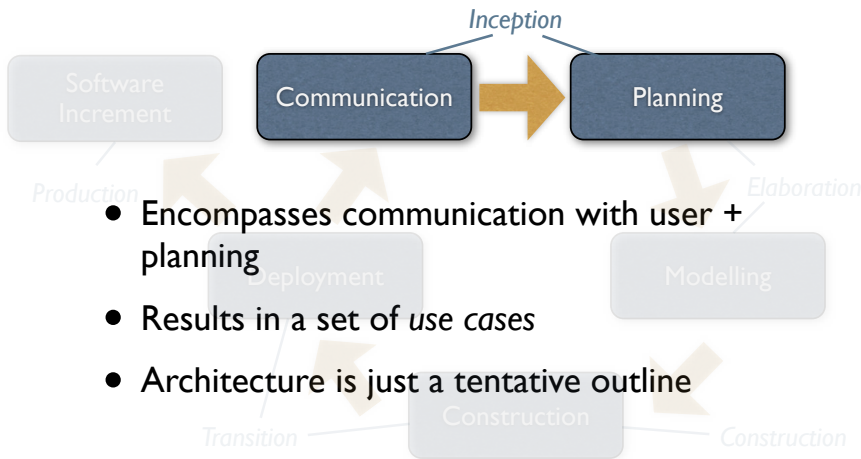
Unified Process

(1999)



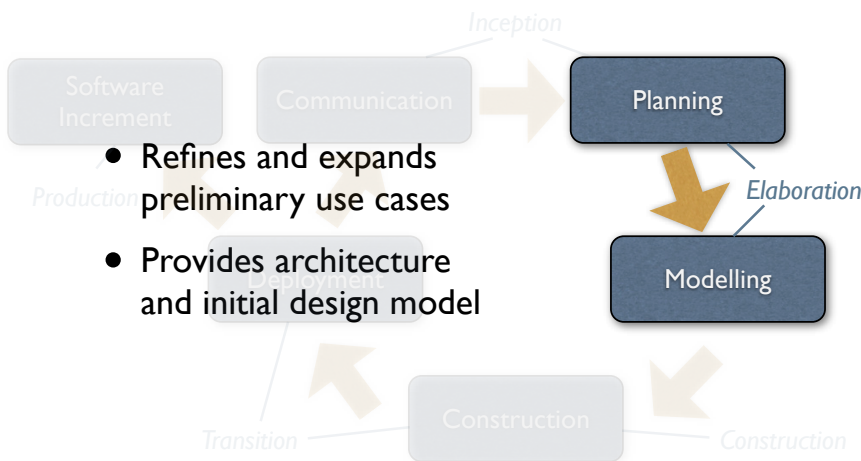
30

Inception



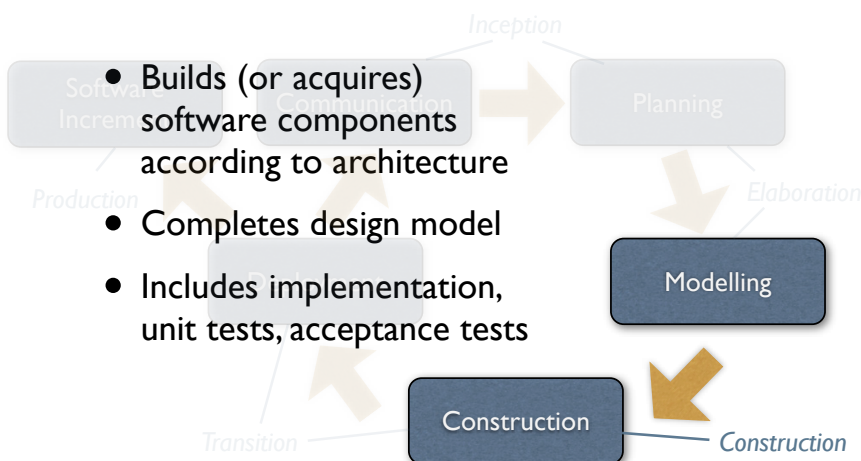
31

Elaboration



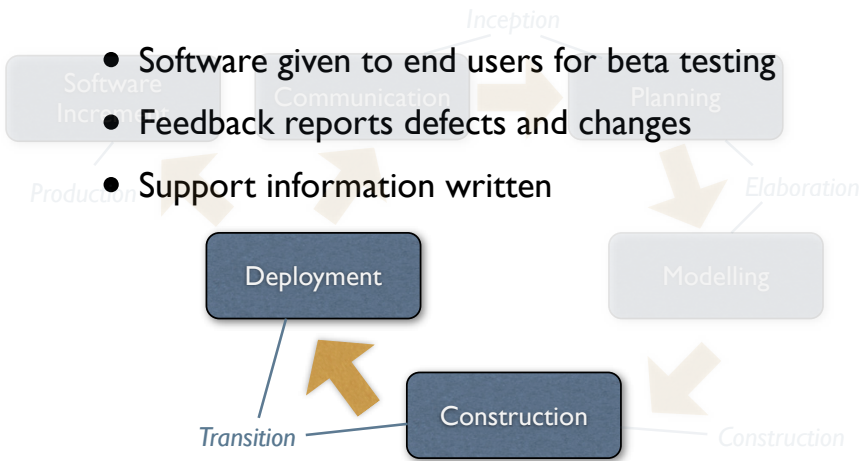
32

Construction



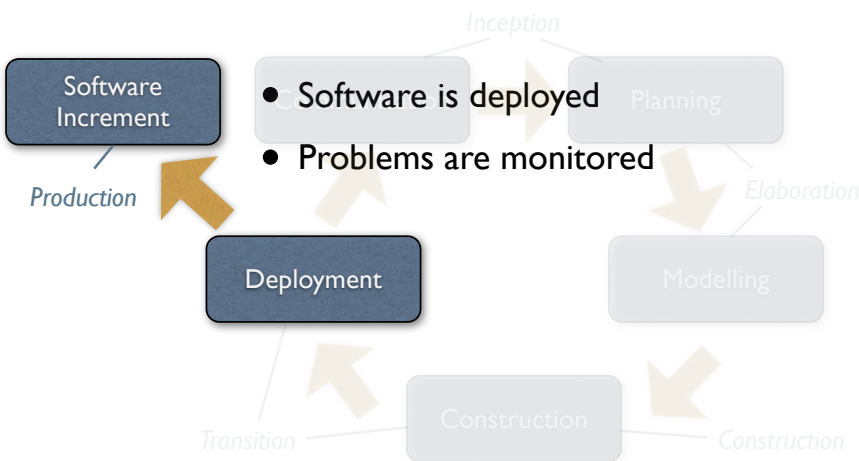
33

Transition



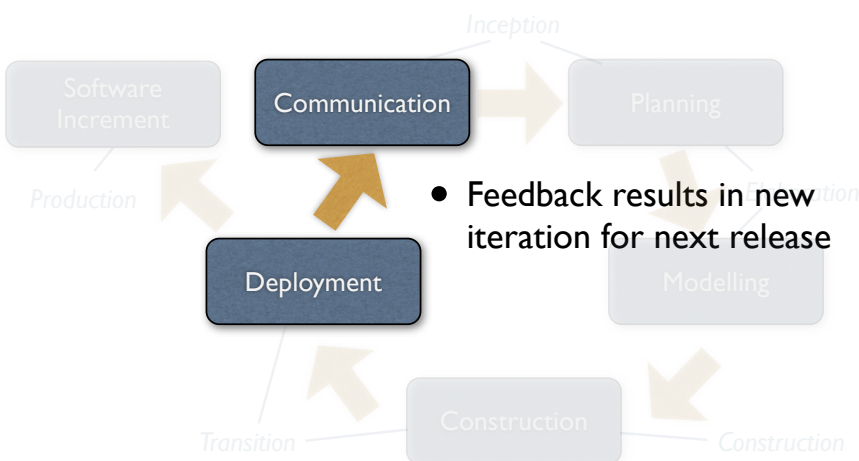
34

Production



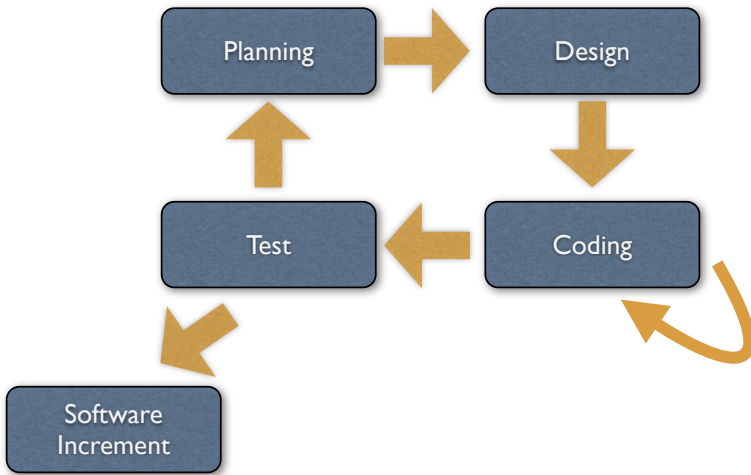
35

Re-Iteration



36

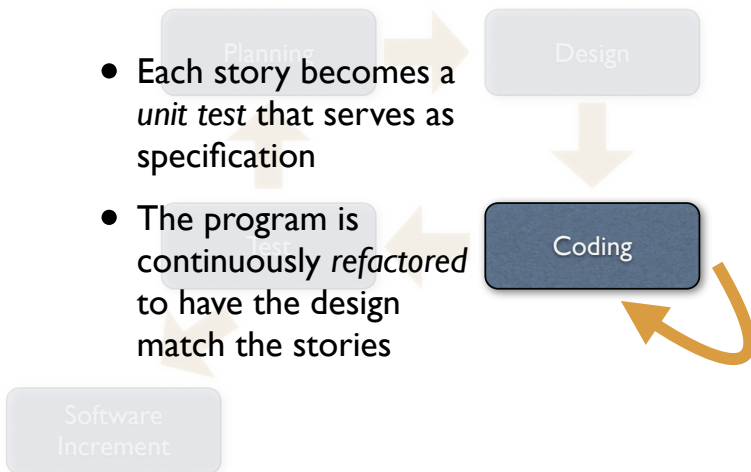
Extreme Programming



43

Coding

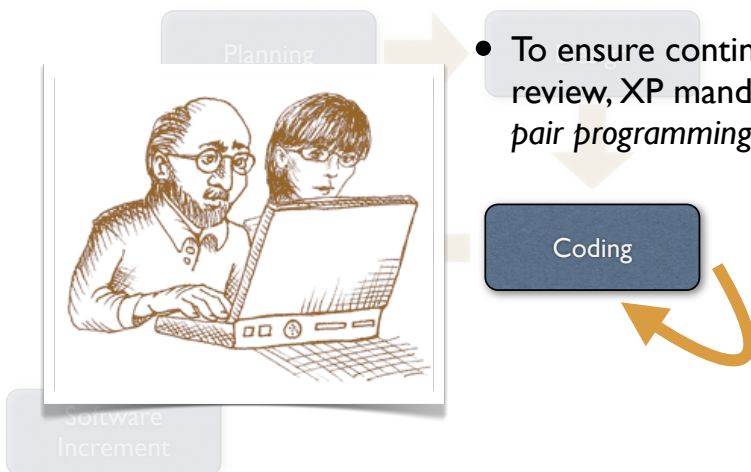
- Each story becomes a *unit test* that serves as specification
- The program is continuously *refactored* to have the design match the stories



44

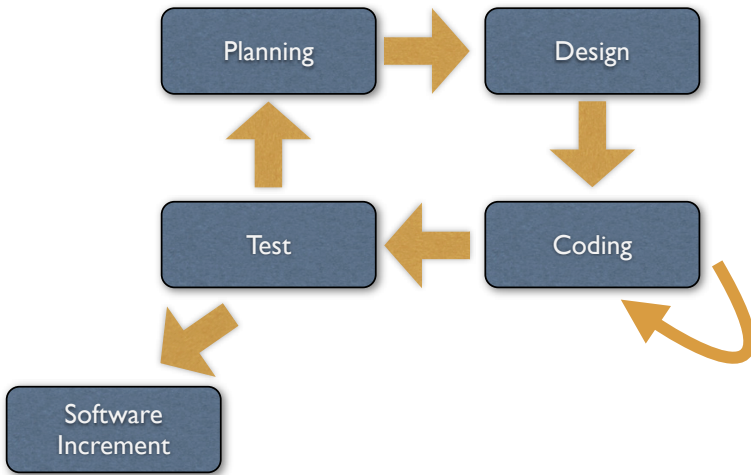
Coding

- To ensure continuous review, XP mandates *pair programming*



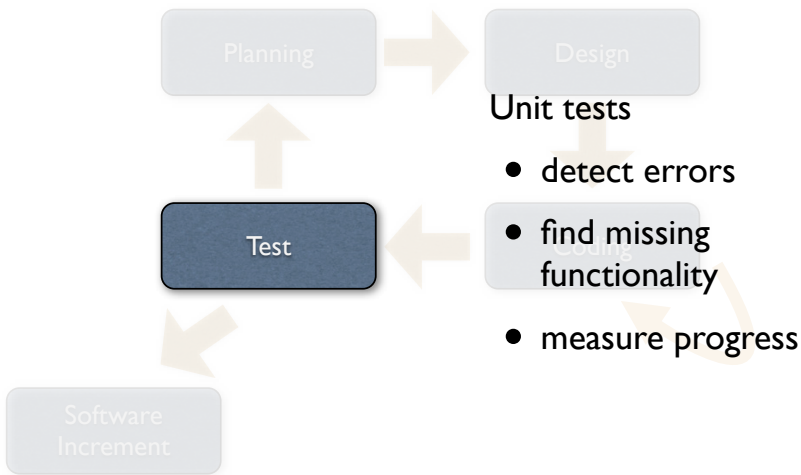
45

Extreme Programming



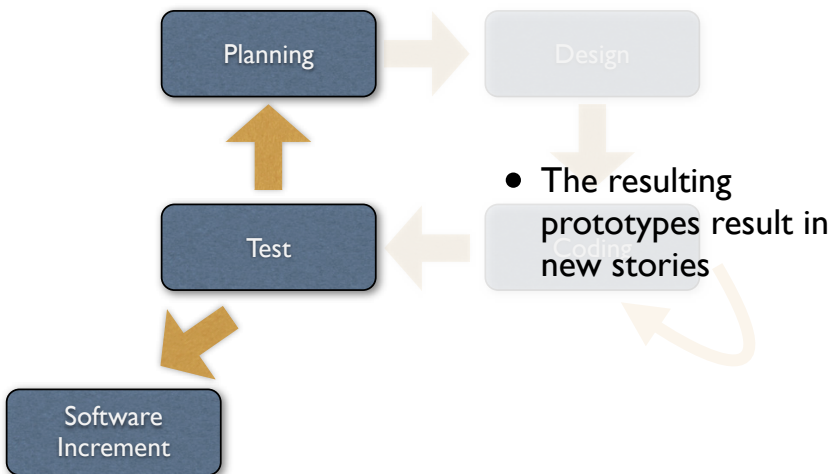
46

Testing



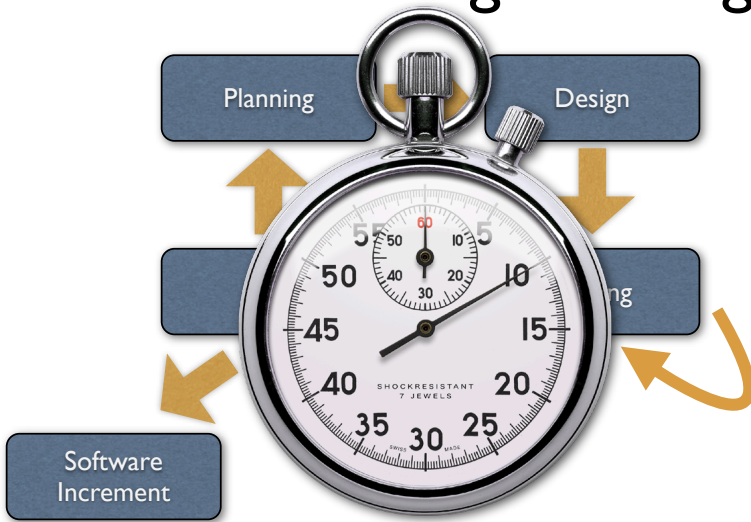
47

Extreme Programming



48

Extreme Programming

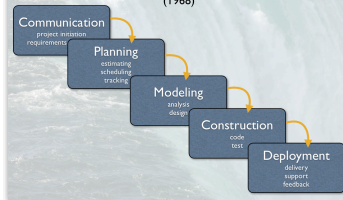


Extreme Programming is fast – with multiple deliverables per day!

Code and Fix

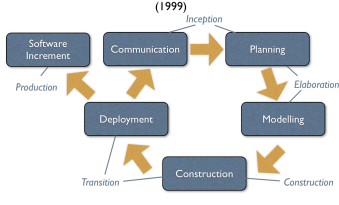


Waterfall Model (1968)



Summary

Unified Process (1999)



Extreme Programming

