Explaining Meaning

Interpreters in MPS

Dr ir Gert Veldhuijzen van Zanten
Dutch Tax and Customs Administration

Problems

- Software-development is
  - Too Expensive
  - Time consuming
  - Complicated
- System cannot be easily changed
- Little reuse of business logic

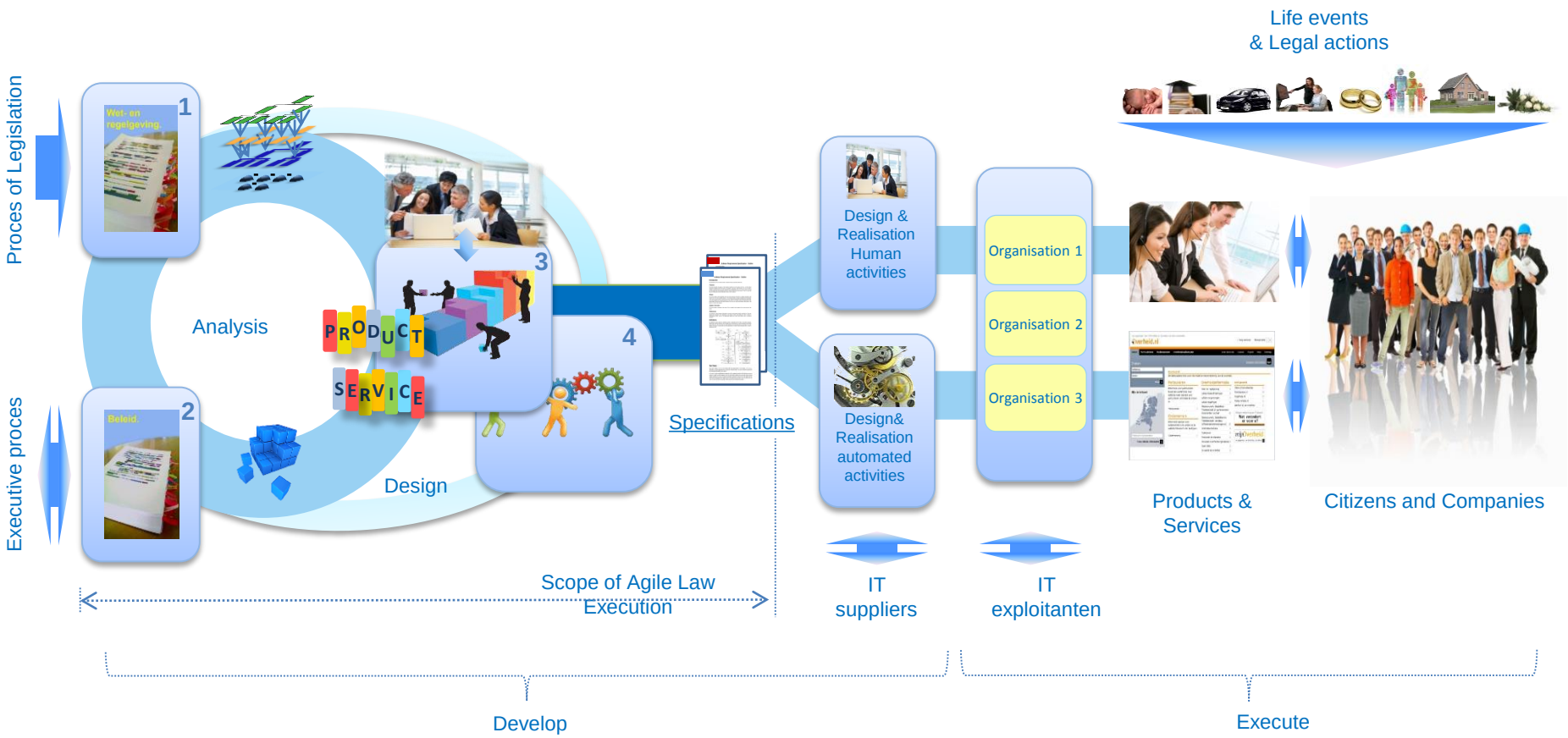
Causes

- Complexity
  - Inherent
  - Accidental
- Business – IT gap
- Unclear requirements
- Formulating requirements is an *art*
  - No tool support

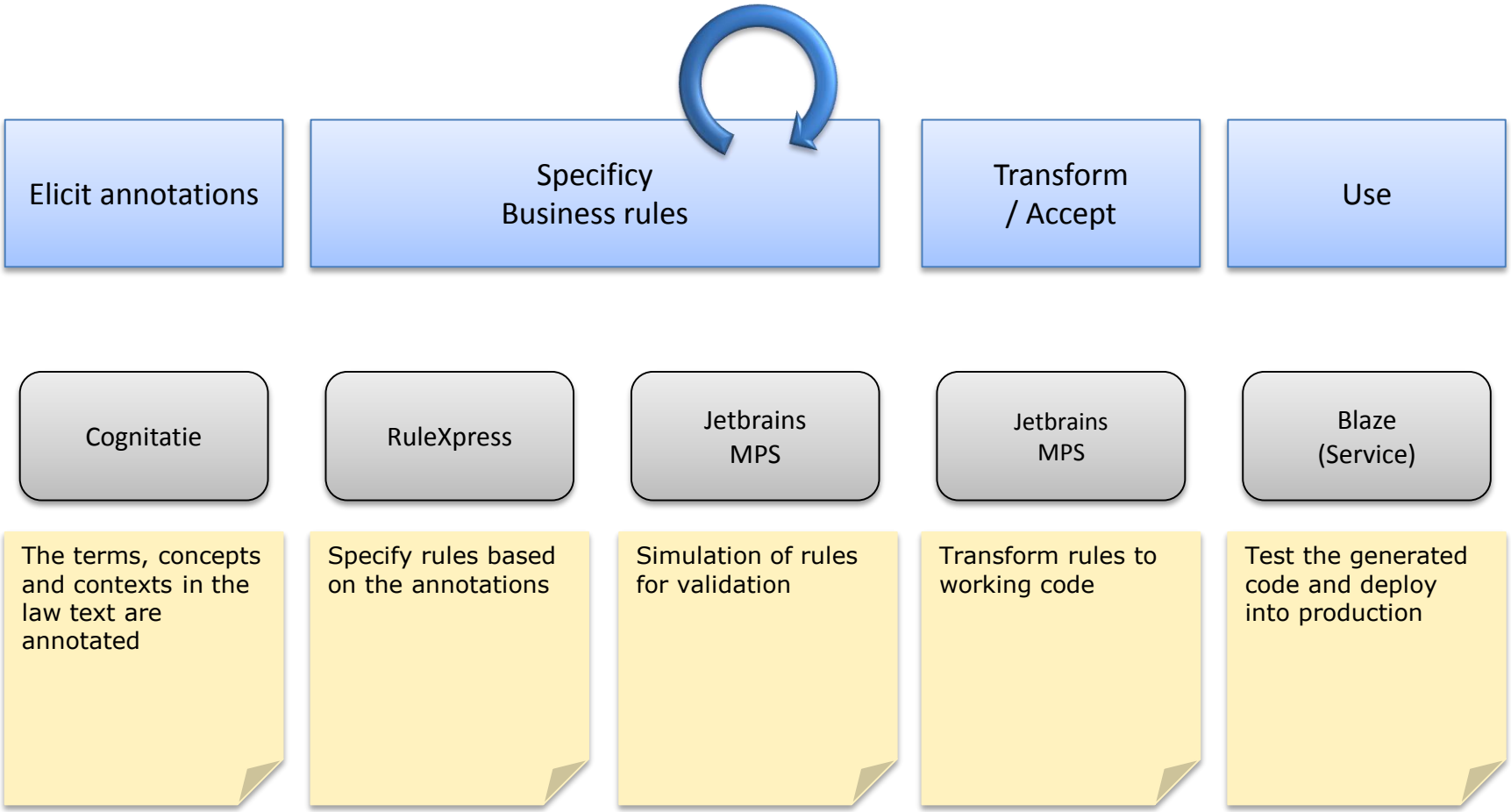
Solutions

- Separation of concerns
- Fast feedback
- Tool support
- Language that can be understood by business people
- Automate automation

Agile Law Execution

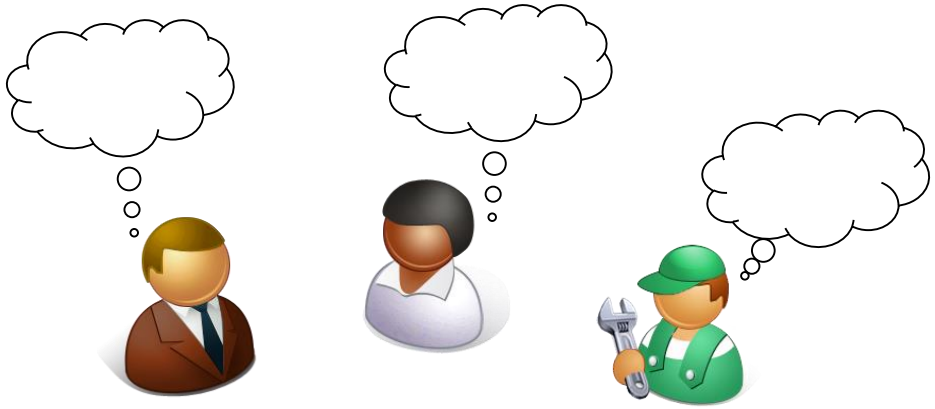


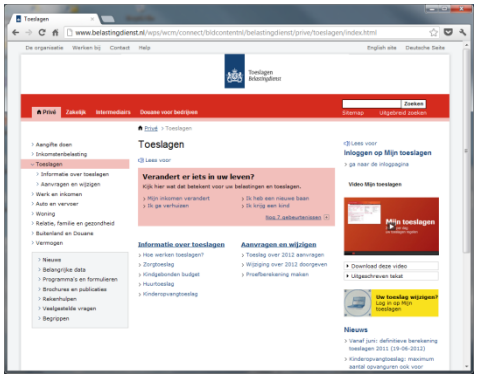
Specification of rules



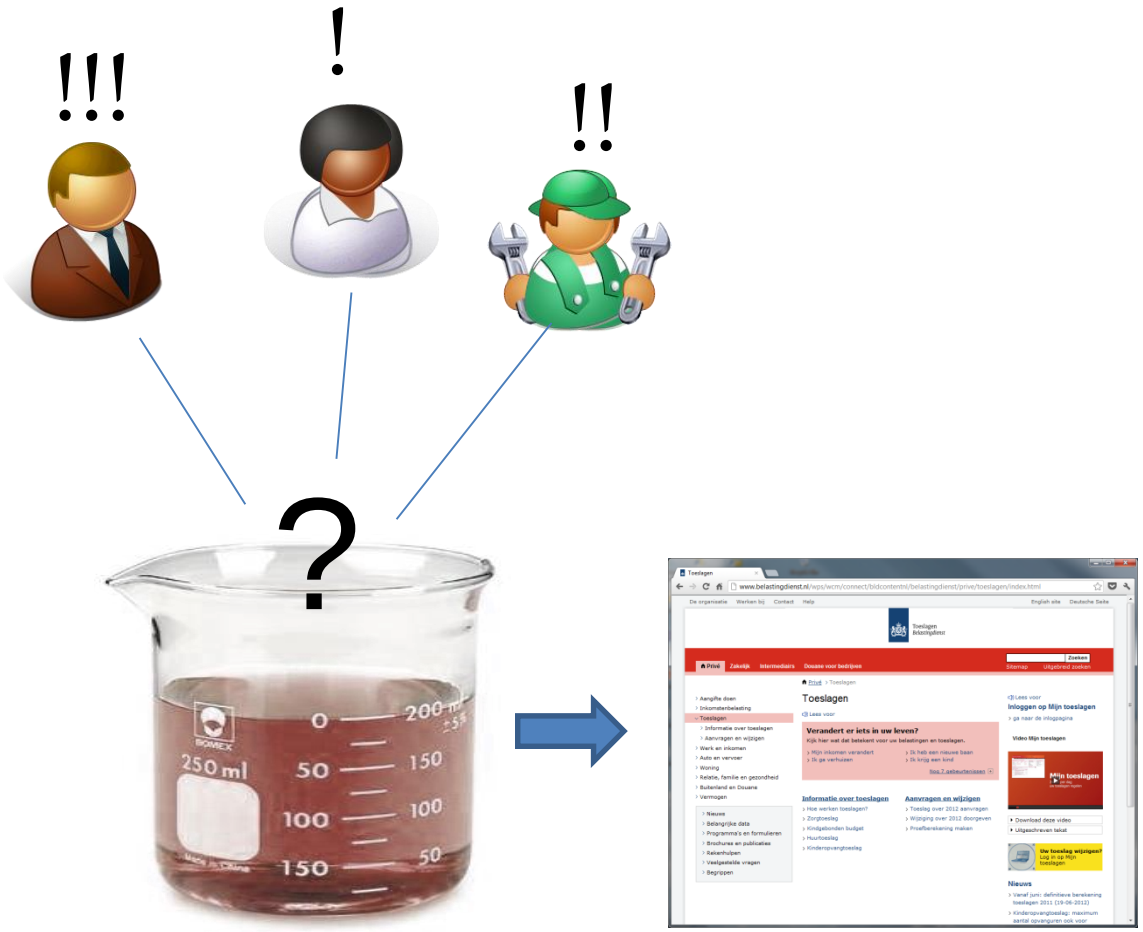
Various concerns

Software-development



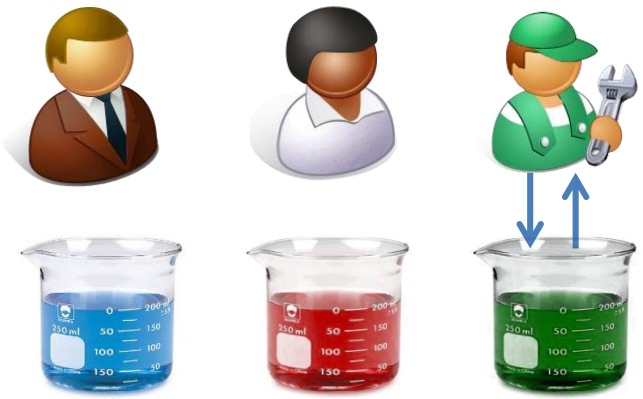


Software-development

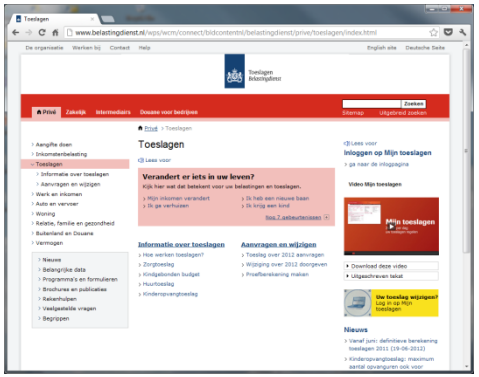


Software-development

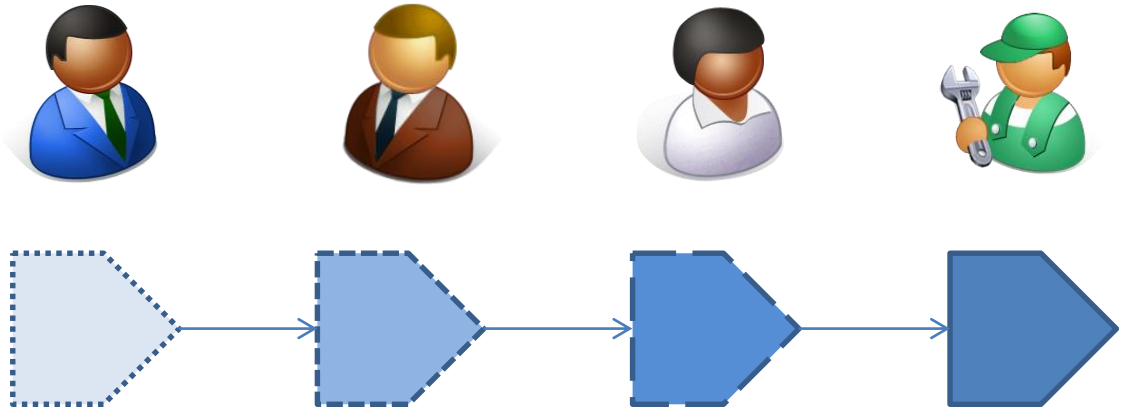
!!



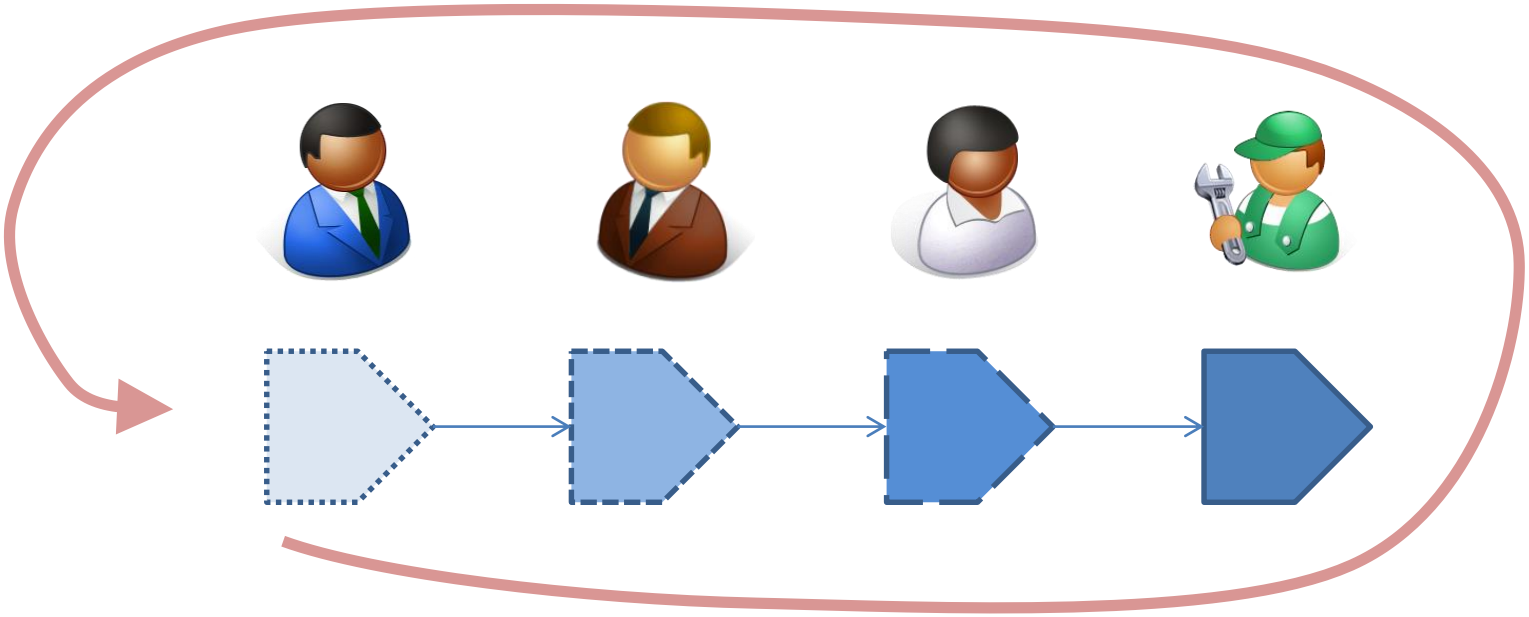
automatic
transformation



Development proces

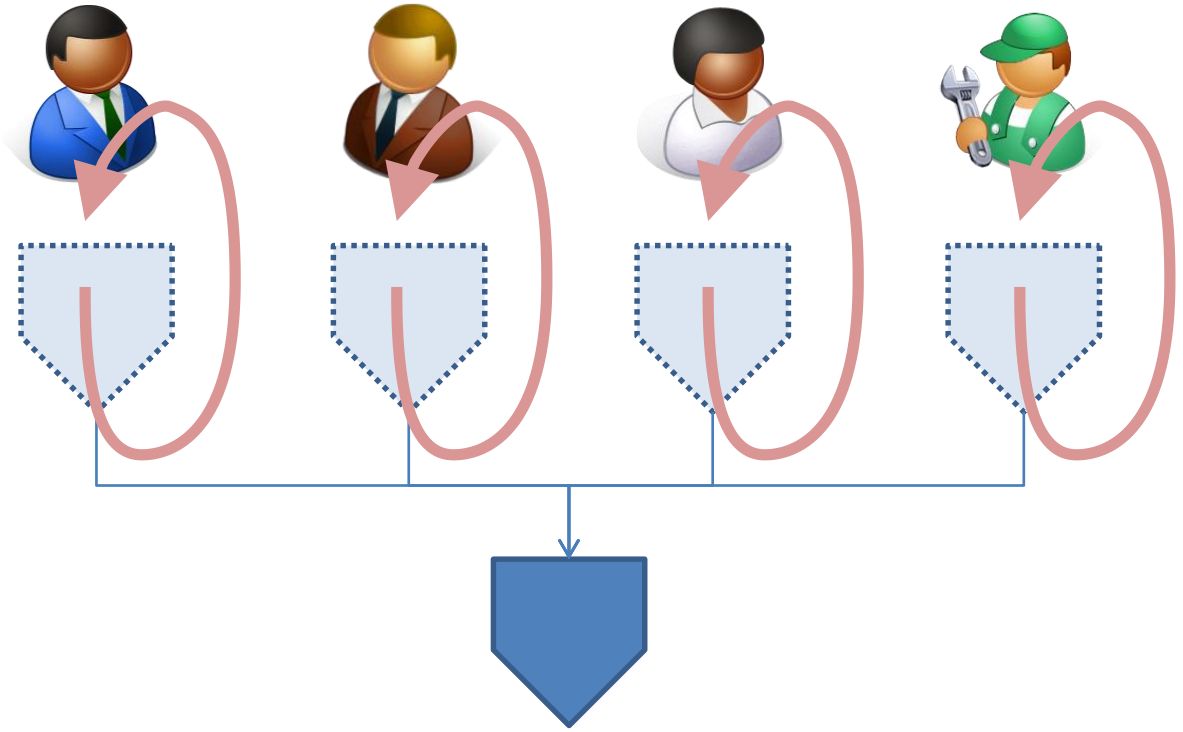


Development proces



A slow and lengthy feedback loop
in which all aspects are involved

Separation of concerns



Multiple fast feedback loops
one per aspect/concern

Model driven software engineering

- Model the essense
  - That which is likely to change
  - Model it in only one place
- Generate the code
  - Using proven programming-/technology patterns
- Automate automation

Domain Specific Languages

- Law
- Execution/Policies
  - How do we gather/deal with information
- Conceptual models
  - Facts vs assertions vs data
  - Time aspects
  - Sources
  - Reliability
- Accountability
  - Traceable rule applications
- Technology
  - Databases, services, network latency, scalability, performance

Language Design and Decomposition

- Abstraction
- What if:
  - We have infinitely fast computers with infinite memory?
  - We would know all there is to know about citizens/companies?
  - Time aspects would play no role?

Example:
current design document

Number of days ZVW (U1):

If [startdate obligation ZVW] (H1) = [empty]
[number of days ZVW] (U1) = 0
Else
If [startdate military service] (1)
and
[enddate military service] present
Then
[number of days ZVW] (U1) =
(month of [enddate obligation ZVW] (H2) minus
month of [startdate obligation ZVW] (H2)) times
30 plus
(day of [enddate obligation ZVW] (H2) minus
day of [startdate obligation ZVW] (H2)) minus
(month of [enddate military service] (f) minus
month of [startdate military service] (e)) times
30 minus
(day of [enddate military service] (f) minus
day of [startdate military service] (e))

In other words

Nr of days per month ZVW = nr of *days* in each *month*
in which premium liable ZVW

Exception:

Nr of days per month ZVW = **30**
in case that premium liable ZVW is valid for the whole *month*

Nr of days per year ZVW = **sum** nr of days per month ZVW
for each *month year*

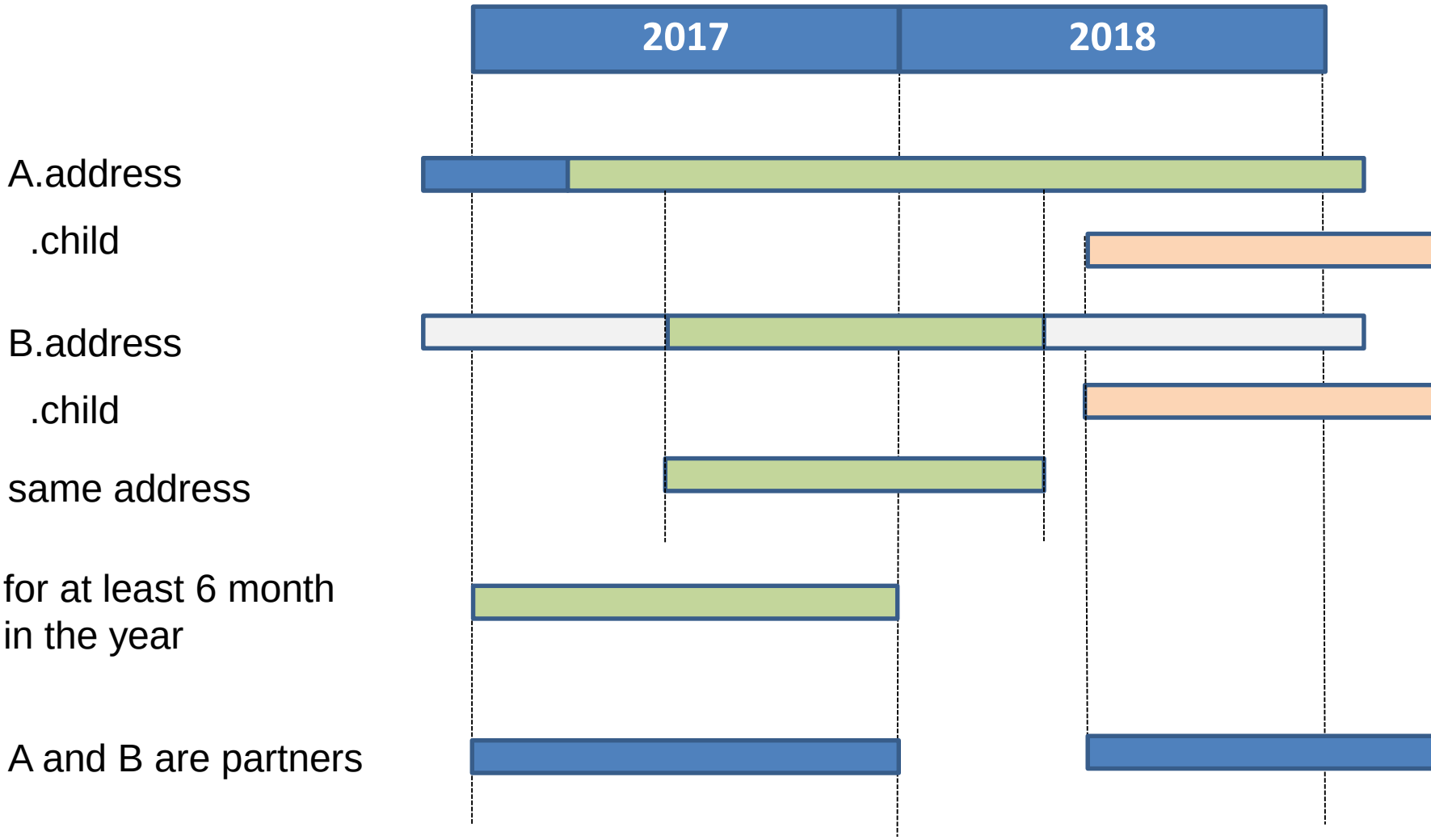
premium liable ZVW = liable ZVW and not military service

Example: time aspects

- Conceptmodel contains information about time aspect
  - Attribute has history
  - Validity granularity: day, month, or year
  - Rounding to month boundaries
- Specificaties contain time operator

```
A is partner of B
   in case that A and B live on the same address for more
          that 6 months in a year
or
   in case that A and B have or once had a child together
```

Partnership

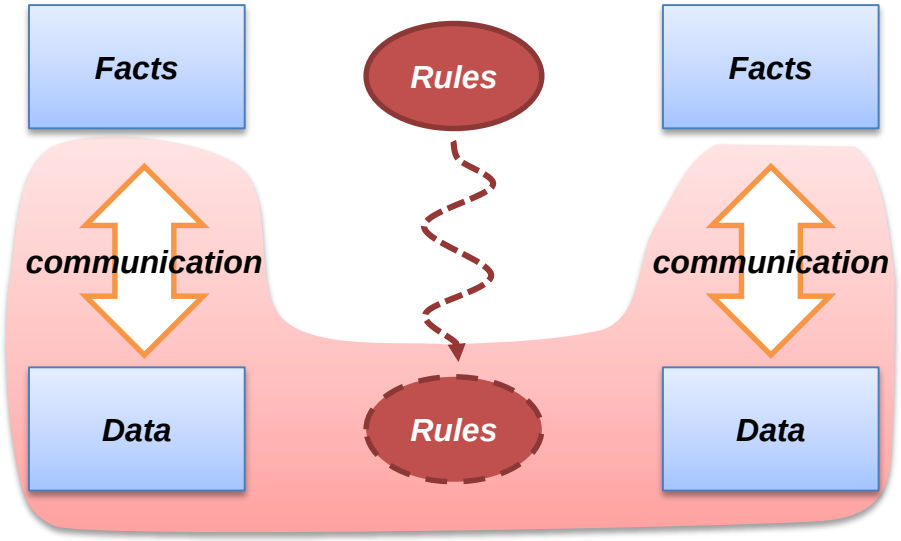


Law-speak

- Use gecontrolled natural language
  - Combines intuïtive and formal meaning
- Scope is limited to factual situations
- Use examples to validate formal meaning
- Explain results
  - Which rules have been applied, to which values

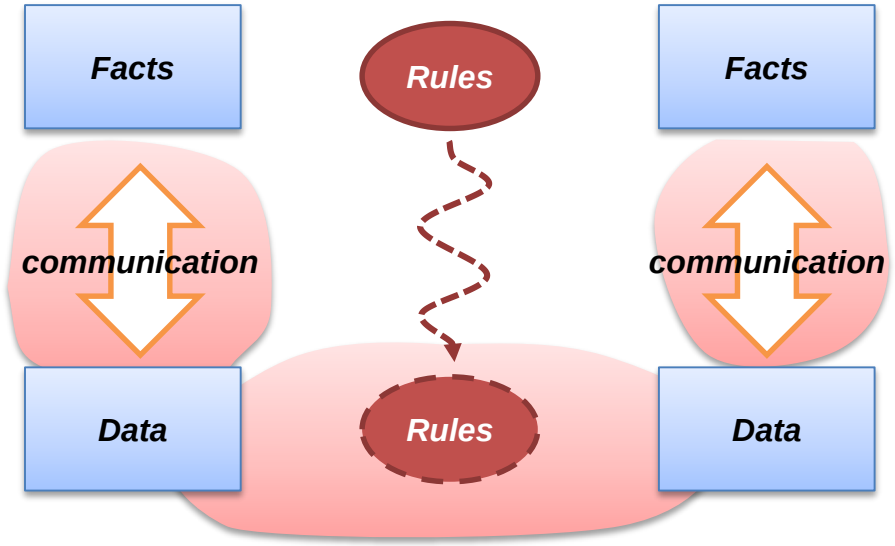
Separation of concerns

- The law deals with factual circumstances and events
- Execution uses data *about* circumstances and events
- Facts and data can only be correlated in terms of communication (assertions…)



Separation of concerns

- The law deals with factual circumstances and events
- Execution uses data *about* circumstances and events
- Facts and data can only be correlated in terms of communication (assertions…)



Feedback

- Model checks
  - Type checking
  - Scope rules
  - Static analysis
- Interpreter
  - Compare results with expectations
  - Accountability (debug traces)

Rules

- One-to-one traceability to law articles
- Exception rules
- Increment / Decrement rules
- Rounding rules
- Time operators

Income tax service

- Shadow run
- 12 million calculations
- Differences with production system ABS
  - 0 for definitive assesments
  - 20 for preliminary assesments
- Nightly build runs all production cases of the whole year
- Dashboard

Demo