Modeling of Business Logic to Enhance Business Agility

Markus Völter
voelter@acm.org
www.voelter.de
@markusvoelter

Markus Völter
itemis Voelter
The Big Picture
Business Agility

Accelerate development & maintenance

Simplify expected diversification & customization
Business Agility – How?

People & Culture

IT Infrastructure & Devops

Process, Methodology & Tools
Business Agility – Process, Methodology & Tools

<table>
<thead>
<tr>
<th>Business Goals</th>
<th>Requirements</th>
<th>Design</th>
<th>Implementation</th>
</tr>
</thead>
</table>

- Business Goals
  - Process, Methodology & Tools
Business Agility – Process, Methodology & Tools

Business Goals

Requirements / Design / Impl

Impl + Non Functional

Automation
Business Agility – Process, Methodology & Tools

Business Goals

Requirements / Design / Impl

Impl + Non Functional

Automation
Business Agility – Process, Methodology & Tools

- Business Goals
- Requirements / Design / Impl
- Impl + Non Functional

Automation

SPEED
CUSTOM
Digitalization – Process, Methodology & Tools

Business Goals

Requirements / Design / Impl

Impl + Non Functional

Automation
Business Agility – DSLs

Domain-Specific Languages

- Good Abstractions and familiar Notations
- Good Abstractions, Clear Semantics
- Test Support & Good Abstractions & Notations
- Generation & CI-Server Integration
- Good IDE, with Interpreter
- Express the right kind of variability
Example DSLs
**Funktionsmodell VKzahlbtgTF**

**Formale Beschreibung**

**Funktion:** VKzahlbtgTF  
**Enthaltende Quelldatei:** wmstcfa  
**Produkt-Typ:** Produkt-Typen auswählen  
**PK-Typ:** PK-Typ auswählen  
**Status:** Status auswählen  

**Parameter-Attribute:**  
- tvk_el_ptr: tvk_el<> E Beschreibung hinzufügen  
- buzbfgr: Ganzzahl A Beschreibung hinzufügen  
- tech_ptr techptr:  

**Rückgabetype:** Kommazahl  

**Verwendete VADM-Attribute:** ...  

**Aufgerufene Funktionen:**  
- VKversartTF (tvk_el_ptr; tvk_el<>; tech_ptr techptr): VERSART

**Beschreibung**

Berechnet den Zahlbeitrag auf Vertragskomponenten-Ebene zurück

**Hilfsvariablen**

- **vkzb:** Kommazahl Beschreibung hinzufügen

**Verarbeitungen**

<table>
<thead>
<tr>
<th>PK_TYP_ID</th>
<th>Beschreibung hinzufügen</th>
<th>Bemerkung</th>
</tr>
</thead>
</table>
| PK_TYP_ID.KAPITAL_KONTO | If (:vtrk_zustand = ZUSTAND.BPFL)  
vkzb = :vtrk_zb  
End If :vtrk_zustand = ZUSTAND.BPFL | Beschreibung hinzufügen |

| PK_TYP_ID.LV_TARIF | If (:stamm_ptr <> NULL)  
If (:zustand = ZUSTAND.BPFL)  
vkzb = :vtrz  
End If  
If VKversartTF (tvk_el_ptr; tech_ptr) = VERSART.BUZB)  
buzbfgr = 0  
End If  
End If :zustand = ZUSTAND.BPFL  
End If :stamm_ptr <> NULL  
Andernfalls | Fehler (PK_TYP_NICHT_IMPLEMENTIERT ) | Beschreibung hinzufügen |

return vkzb
decision table BpScoreDecisionTable(sys: bpRange, dia: bpRange) =

<table>
<thead>
<tr>
<th>sys</th>
<th>&lt;= 90</th>
<th>[91..140]</th>
<th>[141..150]</th>
<th>[151..160]</th>
<th>[161..179]</th>
<th>&gt;= 180</th>
</tr>
</thead>
<tbody>
<tr>
<td>dia</td>
<td>&lt;= 50</td>
<td>[51..90]</td>
<td>[91..95]</td>
<td>[96..100]</td>
<td>[101..109]</td>
<td>&gt;= 110</td>
</tr>
</tbody>
</table>

decision tree DiarrheaStoolsDecisionTree(score: DiarrheaStoolsOverBaseline, patientHasAnySymptom: boolean, goToStartBrat: boolean)

patientHasAnySymptom
  score >= 7
  DiarrheaReco1

score in [4..6]
  DiarrheaReco3
  score in [4..6]
  goToStartBrat
  DiarrheaReco2

Error: type number[32.55][39.99]{4} is not a subtype of number[36|42]{1}

val T_measured: measuredTemp = 42.22
val T_calibrated: temperature = T_measured * 0.93
**D : Kommutationswerte**

<table>
<thead>
<tr>
<th>Ergebnistyp:</th>
<th>Laufvariable:</th>
<th>Parameter:</th>
</tr>
</thead>
<tbody>
<tr>
<td>number[3]</td>
<td>$x$</td>
<td>$i$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>geschlecht</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$q$</td>
</tr>
</tbody>
</table>

\[
D_x := l_x \times \frac{1}{(1 + i)^x}
\]

**l : Lebende im Jahr $x$**

<table>
<thead>
<tr>
<th>Ergebnistyp:</th>
<th>Laufvariable:</th>
<th>Parameter:</th>
</tr>
</thead>
<tbody>
<tr>
<td>number[0]</td>
<td>$x$</td>
<td>geschlecht</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$q$</td>
</tr>
</tbody>
</table>

\[
l_0 := \text{startwertLebende}
\]

\[
l_x := l_{x-1} \times (1 - q \cdot \text{lookup}(x, \text{geschlecht}))
\]
```java
state machine Game (initialPlayers: list<party>) {
  var players = initialPlayers
  var pendings = nopendings
  var bids = nobids
  var lastWinner: opt<party> = none

  event requestAccess
  event startVoting
  event voteCandidate(candidate: party, vote: boolean)

  event terminatePendings
  event play
  event offerBid(money: bid)
  event endGame

  fun clearGame/M() {
    lastWinner := none
    bids := nobids
  }
}

state playing [senderIs(players)] {
  entry: clearGame/M(EMPTY )
  state bidding [takeTurns(players|ordered|after 500 boot out)] (initial) {
    on offerBid(money): bids := bids.put(sender->money)
    if [timeInState > 2000] -> finished
  }
  state finished {
    entry: {
      val maxOfferedValue = bids.values.max
      lastWinner := bids.keys.findFirst(bids[it] == maxOfferedValue)
    }
    on endGame -> setup
  }
  exit: clearGame/M(EMPTY )
}
```
Unterhaltsvorschuss

Zeitangabe: laufend
Häufigkeit: monatlich einmal
Leistungskontext:
Leistungsart: Leer
Zählart: uvg
Anspruch Beginn: Anfang – Unbegrenzt: junger Mensch.geburtsdatum
Anspruch Ende: 01.01.1800  – 31.12.9999  : min(junger Mensch.geburtsdatum + 12 Jahre ,
                      datum + 72 Monate – Anzahl Monate mit uvg)

Zeitraum für Berechnung: Anfang – Unbegrenzt: {standardzeitraum, standardzeitraum}
zweckgebundene Leistung: □
dem Grunde nach: □

Zeitraumbbezogene Daten
nullwerte Anzeigen : boolean = 01.01.1800  – 31.05.2016  : true
                      01.06.2016  – Unbegrenzt  : false
berechnungsart : berechnungsarttyp = 01.01.1800  – 31.12.9999  : dreißigstel

Bezugobjekte:
Attribute: bemerkung : string wird validiert
            antragsdatum : Datum
rule TimingPattern_07_ObjectsAcrossRows_Modified_TandA_897_1Iteration

description: 1381R3.ARCHD.0609.TimingPatternLanguage_TestScenarioMap.xlsx

parameters: [TimePeriodObjectTypA]

patterns:

table
<table>
<thead>
<tr>
<th>scenario</th>
<th>TimePeriodObjectTypA</th>
<th>TimePeriodObjectTypA6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>□</td>
<td>□</td>
</tr>
</tbody>
</table>

< TimePeriodSpecifier2::Duration = 24 Hours >

< TimePeriodSpecifier3::Duration = 15 Minutes >

\[ \triangle \text{TimeSpikeObjectTypA5} \]

database databaseOneElementAcrossRows

<table>
<thead>
<tr>
<th>Type</th>
<th>Begin</th>
<th>End</th>
<th>Duration</th>
<th>Occurence</th>
</tr>
</thead>
<tbody>
<tr>
<td>eTimePeriodObjectTypA</td>
<td>500</td>
<td>550</td>
<td>50</td>
<td></td>
</tr>
</tbody>
</table>

database databaseOneAndMoreIterationsHappy

<table>
<thead>
<tr>
<th>Type</th>
<th>Begin</th>
<th>End</th>
<th>Duration</th>
<th>Occurence</th>
</tr>
</thead>
<tbody>
<tr>
<td>eTimePeriodObjectTypA</td>
<td>50</td>
<td>100</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>eTimeSpikeObjectTypA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>eTimePeriodObjectTypA</td>
<td>86020</td>
<td>86030</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Damage potentials</td>
<td>Required attack potentials</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Very low</td>
<td>Low risk</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Low</td>
<td>Low risk</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Moderate</td>
<td>Moderate risk</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>High</td>
<td>High risk</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Very high</td>
<td>Very high risk</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Risks Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Title</th>
<th>Security goal classes</th>
<th>SUD Elements</th>
<th>Criteria</th>
<th>Depends on</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2.16</td>
<td>Confidentiality UpdateRequest</td>
<td>Confidentiality</td>
<td>UpdateRequest</td>
<td>API</td>
<td>API, OAPI</td>
</tr>
<tr>
<td>S2.17</td>
<td>Availability UpdateRequest</td>
<td>Availability</td>
<td>UpdateRequest</td>
<td>NSG</td>
<td>NSG, API</td>
</tr>
<tr>
<td>S2.18</td>
<td>Authentication UpdateRequest</td>
<td>Authentication</td>
<td>UpdateRequest</td>
<td>API</td>
<td>API, OAPI</td>
</tr>
<tr>
<td>S2.19</td>
<td>Confidentiality UpdateContainer</td>
<td>Confidentiality</td>
<td>UpdateContainer</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.20</td>
<td>Availability UpdateContainer</td>
<td>Availability</td>
<td>UpdateContainer</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.21</td>
<td>Authentication UpdateContainer</td>
<td>Authentication</td>
<td>UpdateContainer</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.22</td>
<td>Confidentiality ConfigurationProfiles</td>
<td>Confidentiality</td>
<td>ConfigurationProfiles</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.23</td>
<td>Availability ConfigurationProfiles</td>
<td>Availability</td>
<td>ConfigurationProfiles</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.24</td>
<td>Authentication ConfigurationProfiles</td>
<td>Authentication</td>
<td>ConfigurationProfiles</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.25</td>
<td>Confidentiality Dagg</td>
<td>Confidentiality</td>
<td>Dagg</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.27</td>
<td>Authentication Profileselection</td>
<td>Authentication</td>
<td>Profileselection</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.28</td>
<td>Confidentiality WebsiteData</td>
<td>Confidentiality</td>
<td>WebsiteData</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.29</td>
<td>Availability WebsiteData</td>
<td>Availability</td>
<td>WebsiteData</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.30</td>
<td>Authentication WebsiteData</td>
<td>Authentication</td>
<td>WebsiteData</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.31</td>
<td>Confidentiality GUU</td>
<td>Confidentiality</td>
<td>GUU</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.32</td>
<td>Availability GUU</td>
<td>Availability</td>
<td>GUU</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
<tr>
<td>S2.33</td>
<td>Integrity GUU</td>
<td>Integrity</td>
<td>GUU</td>
<td>OAPI</td>
<td>OAPI</td>
</tr>
</tbody>
</table>

### Manipulation/Tampering...

<table>
<thead>
<tr>
<th>S2.42</th>
<th>Integrity WebService</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAP:</td>
<td>Very low</td>
</tr>
</tbody>
</table>

### S2.30 Authenticity WebsiteData

<table>
<thead>
<tr>
<th>S2.30</th>
<th>Confidentiality WebsiteData</th>
<th>Confidentiality</th>
<th>WebsiteData</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Confidentiality WebsiteData</td>
<td>Confidentiality</td>
<td>WebsiteData</td>
</tr>
<tr>
<td></td>
<td>Confidentiality WebsiteData</td>
<td>Confidentiality</td>
<td>WebsiteData</td>
</tr>
<tr>
<td></td>
<td>Confidentiality WebsiteData</td>
<td>Confidentiality</td>
<td>WebsiteData</td>
</tr>
<tr>
<td></td>
<td>Confidentiality WebsiteData</td>
<td>Confidentiality</td>
<td>WebsiteData</td>
</tr>
</tbody>
</table>

### B.16 (Identity) Spoofing of...

<table>
<thead>
<tr>
<th>S2.42</th>
<th>Integrity WebService</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAP:</td>
<td>Beyond high</td>
</tr>
</tbody>
</table>

### B.13 Man in the Middle Attack...

<table>
<thead>
<tr>
<th>S2.42</th>
<th>Integrity WebService</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAP:</td>
<td>Beyond high</td>
</tr>
</tbody>
</table>
public functional component DriveTrain {
    produces SpeedFromEngine
    produces EngineStatus
    produces Gear where it < gearsCount
    consumes RoadConditions
    param int gearsCount
    consumes DrivingCommands
}

FourCylEngine

DrivingCommands_in

RoadConditions_in

DriveTrainController

Gearbox

SpeedFromEngine_out

EngineStatus_out

Gear_out
```cpp
vector<int16, 3> aVector = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \times 512;

vector<int16, 3> resultOfCrossProduct = aVector \times aVector;

matrix<int16, 2x3> aMatrix = \begin{bmatrix} 1 + 2 & 2 \times 7 & 42 \\ 3 & 51 & 24 \end{bmatrix};

matrix<int16, 3x2> transposedMatrix = aMatrix^T;

int32 averageIntArray(int32[] arr, int32 size) {
    \sum_{i = 0}^{size} arr[i]
    return \frac{\sum_{i = 0}^{size} arr[i]}{size};
} averageIntArray (function)
```
val salary : TT[currency] = TT | /2017 01 01/ => 5.000 EUR |
| /2017 05 01/ => 6.000 EUR |

\[
\begin{align*}
a + s &= \begin{pmatrix} a_1 + s \\ a_2 + s \\ a_3 + s \end{pmatrix} \\
\end{align*}
\]

\[
\begin{align*}
a + b &= \begin{pmatrix} a_1 + b_1 \\ a_2 + b_1 \\ a_3 + b_2 \end{pmatrix} \\
\end{align*}
\]
calculation for Tax
depends Salary as s
valid from /2017 01 01/
calculate [monthly] {
  ...
}

result data [monthly] Salary {
  employment -> Employment
  amount : currency
}

calculation for Tax
depends Salary as s
SomeOtherThing as t
valid from /2017 07 01/
if ctx.employment.person.homeAddress.state == BW
calculate [monthly] {
  ..
}

result data [monthly] Salary from /2017 10 01/ {
  employment -> Employment
  amount : currency
  taxFree : boolean
}
Rabbit Hole

Modeling vs. Programming
A personal note.

A personal note.

MPS with full code generation.

Mixed notations, multiple integrated languages, type checking and inferencing, data flow checking, refactoring, version control, diff/merge, non-trivial transformations, languages for non-programmers.
A personal note.

MPS with full code generation.

Mixed notations, multiple integrated languages, type checking and inferencing, data flow checking, refactoring, version control, diff/merge, non-trivial transformations, languages for non-programmers.

Modeling 2.0

I have no patience for people who still use modeling 1.0 – and then complain that it does not work.

Try 2.0!

2018
Language Workbenches
MPS Language Workbench

Language Workbench
Open Source, by JetBrains
Very Expressive
Used for years in industry
Vast Experience
MPS Language Workbench

+ Refactorings, Find Usages, Syntax Coloring, Debugging, ...
MPS: Notational Freedom

100% crucial for acceptance in domain!

Rabbit Hole

Modeling is more than diagrams!
MPS: Language Composition

Embedding

\[ L_{Host} + L_{Adapt} + L_{Emb} = \]

Extension

\[ L_{Base} + L_{Ext} = \]

Extension Composition

\[ L_{Base} + L_{Ext1} + L_{Ext2} = \]

SPLE on Language Level!
Other Language Workbenches

{S} spoofax  TU Delft
xttext        itemis/Typefox
Rascal        CWI Amsterdam
The Whole Platform  Solmi/Persiani
Metacase  RWTH Aachen
Language Workbench - Convection

Browser · Cloud · Collaborative
Multi-Notation · Language Modularity
FR2  Coffee maker operation

The coffee maker has four main operational modes. These are StartingUp, Ready, Maintenance, and StandBy.

The coffee maker is in operation as soon as the main powerswitch is switched on by the user. If the user switches off the power switch or unplugs the power supply then the machine will go out of operation immediately without changing its mechanical state.

```
state machine OperationModes {

  event done
  event maintain
  event interaction
  number timeout = 0

  initial is StartingUp

  The initial state of the coffee maker is StartingUp. It will be entered as soon as the user physically powers up the machine.

  state StartingUp {
    on done [] -> Ready
    on maintain [] -> }

  state Ready {
    on [timeout >= 120] -> StandBy
    on maintain [] -> Maintenance
  }

  state StandBy {
  }

  state Maintenance {
  }

  done
```

Graphical View

```
state machine OperationModes

StartingUp

done

interaction maintain

Ready

[timout==120]

StandBy

Maintenance
```

**Language Workbench - Convection**

### Assessment Model

<table>
<thead>
<tr>
<th>security goal class</th>
<th>Confidentiality [ CON ]</th>
<th>Availability [ AVA ]</th>
<th>Integrity [ INT ]</th>
<th>Authenticity [ AUT ]</th>
</tr>
</thead>
</table>

**damage potentials**

<table>
<thead>
<tr>
<th>Damage Level</th>
<th>RGB Color Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very low</td>
<td>VLO: # C6EFCE</td>
</tr>
<tr>
<td>Low</td>
<td>LOW: # C4D79B</td>
</tr>
<tr>
<td>Moderate</td>
<td>MOD: # FFF99</td>
</tr>
<tr>
<td>High</td>
<td>HIG: # FFB9C</td>
</tr>
<tr>
<td>Very high</td>
<td>VHI: # E6B8B7</td>
</tr>
</tbody>
</table>

**damage class**

<table>
<thead>
<tr>
<th>Damage Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Safety</td>
<td>[ SAF ]: Violation of safety-relevant aspects</td>
</tr>
<tr>
<td>Financial consequences [ FIA ]</td>
<td>[ FIA ]: Financial consequences</td>
</tr>
<tr>
<td>Laws &amp; Privacy [ LAW ]</td>
<td>[ LAW ]: Violation of laws, regulations, contracts, or impairment of the right to informational self-determination</td>
</tr>
<tr>
<td>Quality</td>
<td>[ QUA ]: Reduction of quality</td>
</tr>
</tbody>
</table>

**damage subclass**

<table>
<thead>
<tr>
<th>Financial Consequences [ FIA ]</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monetary losses (Monetary) [ LLO ] = LOW: Low</td>
<td></td>
</tr>
<tr>
<td>Undesirable financial damage. [ LMD ] = MOD: Moderate</td>
<td></td>
</tr>
<tr>
<td>Substantial financial damage, but yet not existence-threatening. [ LHI ] = HIG: High</td>
<td></td>
</tr>
<tr>
<td>Existence-threatening financial damage. [ LVH ] = VHI: Very high</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Safety with criteria</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Potentially dangerous situation. [ LIL ] = LOW: Low</td>
<td></td>
</tr>
<tr>
<td>Light and moderate injury. [ LIM ] = MOD: Moderate</td>
<td></td>
</tr>
<tr>
<td>Life-threatening injury (survival uncertain), fatal injury. [ LIF ] = VHI: Very high</td>
<td></td>
</tr>
</tbody>
</table>
A Language is not Enough
More than Language

Great IDE
- Syntax Coloring
- Code Completion
- Goto Definition
- Collaboration
- Migration

Language
- Abstractions
- Notations

GOOD

GREAT
More than Language

**GOOD**

- Language
  - Abstractions
  - Notations

- Great IDE
  - Syntax Coloring
  - Code Completion
  - Goto Definition
  - Collaboration
  - Migration

- Analyses
  - Relevant
  - Good Errors
  - Type System!

- Refactorings
  - Aligned with Processes

**GREAT**

- Testing
  - Write Tests
  - Run them in IDE
  - Report Back

- Debuggers
  - Animate Execution
  - Simulators

GOOD to GREAT: More than Language
Influences on Language Design
Language Design Influences

Domain Structure

Non Functionals
Permissions, IP, Sharing

User Skills

Sep. of Concerns
Different Views

Educate, Put results in context

Get a better tool :-)

Refactor towards Structure

Model Purpose
Analyze, Generate

Tool Capabilities
Notations, Editing, Scale

Software Engineering Practices
Language Design In flu enes

- **Domain Structure**
- **Non Functionals**
  - Permissions, IP, Stewardship
- **User Skills**
- **Model Purpose**
  - Analyze, Generate
- **Tool Capabilities**
  - Notations, Editing, Scale
- **Software Engineering Practices**

- Sep. of Concerns
- Different Views
- Refactor towards Structure
- Get a better tool :-)
- Style!

- Educate, Put results in context
Language
Modularity
Works
DSL Development

**GPL Extension**
- Reuse GPL incl. Expressions and TS
- Add/Embed DS-extensions
- Compatible notational style
- Reduce to GPL

**New Language**
- Analyze Domain to find Abstractions
- Define suitable, new notations
- Rely on existing behavioral paradigm
- Reuse standard expression language
- Interpret/Generate to one or more GPLs

**Formalization**
- Use existing notation from domain
- Clean up and formalize
- Generate/Interpret
- Often import existing “models”
Is this the next legacy system?
Today’s software is tomorrow’s legacy system.

Or is it?
Existing models become incompatible with new language

⇒ Language Versions
Migration Scripts
Runtime Tech outdated, uncool or slow

⇒ Keep Lang Technology
Keep Models
Build new Generator
Language Tech outdated, uncool

⇒ Build new Tool

Migrate Data

Feasible, because it well-defined domain semantics and free from „technology stuff“
Today’s software is tomorrow’s legacy system.

No, it is not.
„We don’t have The Skills“
Organizations do not have the necessary skills to build DSLs. True.

But...

Separation of Concerns
Most important Architectural Decision:
Separate Business Logic from Technology. Completely!
Modeling vs. Programming

Modeling ≠ Programming

Modeling:
- Performance
- Scalability
- Robustness
- Deployment

Programming:
- Formulas, Rules
- Data Structures
- Tables
- Values
The concepts in the DSL are well aligned with the domain. This makes learning the language much simpler. And there is tool support that helps create valid models.
Meaningful Validation
More Efficient Development
Integrate Domain Experts
Decouple from Technology