Domain Specific Languages and Requirements (Engineering)

What are Requirements?
... a requirement is a singular documented need of what a particular product or service should be or perform.

Wikipedia

... specifies a verifiable constraint on an implementation that it shall undeniably meet or (a) be deemed unacceptable, or (b) result in implementation failure, or (c) result in system failure.

Wiktionary
... what a system should do, and with which quality attributes, without presupposing a specific implementation.

Cohesive
Complete
Consistent
Atomic
Traceable
Current
Feasible
Unambiguous
Mandatory
Verifiable
**Formal** requirements specify what a system should do from a domain perspective, and with which quality attributes, without presupposing a specific software implementation, but processable by tools.
A DSL is a **focussed, processable language** for describing a specific **concern** when building a system in a specific **domain**. The **abstractions** and **notations** used are natural/suitable for the **stakeholders** who specify that particular concern.
domain specific

tailor made
effective++
specialized, limited
used by experts
together with other specialized tools
map

DSL Program
(aka Model)

map

automated!

GPL Program
map

Generation
Transformation
Compilation

Interpretation

Example 1:

Pension Fund Specification
Calculation Rules and Tests

Intentional Software’s Domain Workbench
Example 3:  
Radar Systems Engineering

Component Definition
Component Behavior Specification

Resulting System Behaviour
Analysis

Eclipse Modeling
Eclipse Xtext
Wolfram Mathematica
Mathematica Workbench
Example 2:
Fountains
Hardware Structure

```plaintext
feature BasicOnePump
  pump compartment ccl
  static compressor c1

feature AtLeastOneZone extends BasicOnePump
  water compartment compl
  pumped by c1
  compartment levelsensor ct_f1
  light l_f1

feature[f] SuperPowerCompartment
  water compartment adds to f
  superPowerMode

feature WithAlarm
  level alarm a1

fountain StdFountain extends AtLeastOneZone
```

Behaviour

```plaintext
pumping program F1 for AtLeastOneZone + WithAlarm +
  SuperPowerCompartment[f=compl1] {
  parameter defaultWaterLevel : int
  parameter superWaterLevel: int
  event superPowerTimeout

  init {
    set compl1->targetHeight = defaultWaterLevel
  }

  start:
  on (compl1->needsPower == true) && !(compl1->isPumping) {
    do compl1->pumpOn
  }
  on compl1->enough {
    do compl1->pumpOff
  }
  on compl1.superPumping->turnedOn {
    set compl1->targetHeight = superWaterLevel
    raise event superPowerTimeout after 20
  }
  on compl1.superPumping->turnedOff or superPowerTimeout {
    set compl1->targetHeight = defaultWaterLevel
  }
}  ```
Plus:

In-IDE Simulator
Unit Test Support

Eclipse Modeling
Eclipse Xtext
What if I don’t yet have a language?

Actually, this is the normal case! **Domain Specific Language**
Building Languages

As you understand the domain...
...develop a language to express it!

Language resembles domain concepts
Then express the design with the language.

Clear understanding of the domain from building the language
Iterate!

Understand Domain
Domain Expert
Language Engineer

Define Language
Language Engineer

Use Language
Domain Expert
Domain User

Iterate!

Understand Domain
Domain Expert
Language Engineer

Define Language
FORMAL!

Use Language
FORMAL!
Like Analysis

... but Executable!

DSL Engineering Tools
Open Source
Eclipse Public License

Large wold wide community
graphical, textual and form-based DSLs

Developed by JetBrains
Open Source
Apache 2.0
Projectional Editor
all kinds of notations, mainly textual

Commercial Product
Projectional Editor
Most flexible notations
Textual Requirements?

Still necessary.
Tracing

```c
trace Cyclic

 This is the cyclic task that is called every 1ms to do the actual control of the
 task run_cyclic prio = 1 every = 2

 trace Communicate

 state running

 int8 bump = 0;

 bump = accrob_1_get_touch_event(SENSOR_PORT_I::NEXT_PORT_3);

 if (bump == 1) {

 send linefollower: bumped

 terminate

 } trace Talk

 int32 light = 0;

 light = accrob_1_get_light_event(SENSOR_PORT_I::NEXT_PORT_1);

 if (light < 4 \&\& write = BLACK) 1 / 2 {

 send ConsistentSetting;

 updateMotorSettings(SLOW, FAST)

 } else {

 send ConsistentSetting;

 updateMotorSettings(FAST, SLOW)

 }

 state crash

 updateMotorSettings(0, 0);

 default

 } <loop>;
```
Tracing

```c

taxon Cyclis
task run cy

<table>
<thead>
<tr>
<th>Trace ID</th>
<th>Event</th>
<th>Event ID</th>
<th>duration</th>
<th>log</th>
<th>status</th>
<th>comment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2011-03-28</td>
<td>00000000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Summary

- Image of two airplanes
Cohesive Complete Consistent Atomic Traceable Current Feasible Unambiguous Mandatory Verifiable Validation Checking Simulation

Cohesive Complete Consistent Atomic Traceable Current Feasible Unambiguous Mandatory Verifiable Everything is a model tracing links simple Wikipedia
Cohesive
Complete
Consistent
Atomic
Traceable
Current
Feasible
**Unambiguous**
Described Formally
Mandatory
Verifiable

Domain Expert involved in Definition and Review

Wikipedia
Cohesive
Complete
Consistent
Atomic
Traceable
Current
Feasible
Unambiguous
Mandatory
Verifiable

**Executable** Automatic Refinement downstream, Code Gen.

---

Cohesive
Complete
**Consistent**
Atomic
Traceable
Current
Feasible
Unambiguous
Mandatory
Verifiable
Executable

**Reward** for the additional effort of formalization!
And Developers???

... Languages
... Technology Evaluation
... Generators
... Testing
... Operations

... what they want to do anyway!

Brain [Domain Person]

DSL

Brain [Developer]

Code
Getting Started

1. Familiarize yourself with some of the tools to better understand their capabilities
Getting Started

2 Identify a limited, but meaningful aspect of your system and build a prototype

Getting Started

3 Evaluate lessons learned (technical)
Getting Started

4 Think about „more strategic“ places where DSLs can be used, and consider process consequences

Getting Started

5 Repeat forever :-)

THE END.

coordinates
web www.voelter.de
email voelter@acm.org
skype schoggid
xing http://www.xing.com/profile/Markus_Voelter
linkedin http://www.linkedin.com/pub/0/377/a31