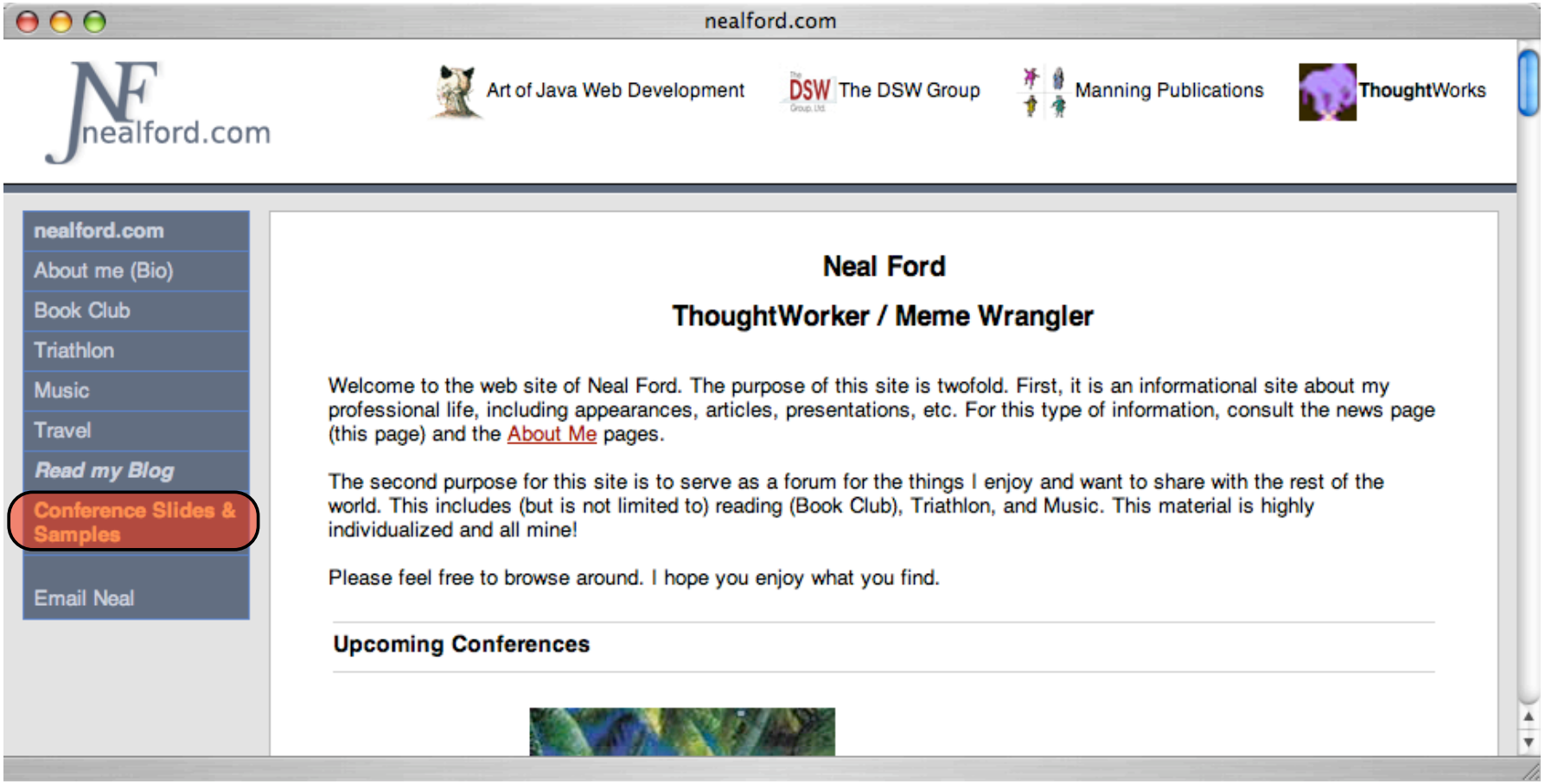
building DSLs with language workbenches

NEAL FORD software architect / meme wrangler

ThoughtWorks

nford@thoughtworks.com
3003 Summit Boulevard, Atlanta, GA 30319
www.nealford.com
www.thoughtworks.com
memeagora.blogspot.com


nealford.com
nealford.com
Art of Java Web Development
The DSW Group
Manning Publications
ThoughtWorks
nealford.com
About me (Bio)
Book Club
Triathlon
Music
Travel
Read my Blog
Conference Slides & Samples
Email Neal
Neal Ford
ThoughtWorker / Meme Wrangler
Welcome to the web site of Neal Ford. The purpose of this site is twofold. First, it is an informational site about my professional life, including appearances, articles, presentations, etc. For this type of information, consult the news page (this page) and the About Me pages.
The second purpose for this site is to serve as a forum for the things I enjoy and want to share with the rest of the world. This includes (but is not limited to) reading (Book Club), Triathlon, and Music. This material is highly individualized and all mine!
Please feel free to browse around. I hope you enjoy what you find.
Upcoming Conferences

overview

language workbenches

building grammars or pseudo-grammars

generation templates

generation

using workbench DSLs

language workbenches

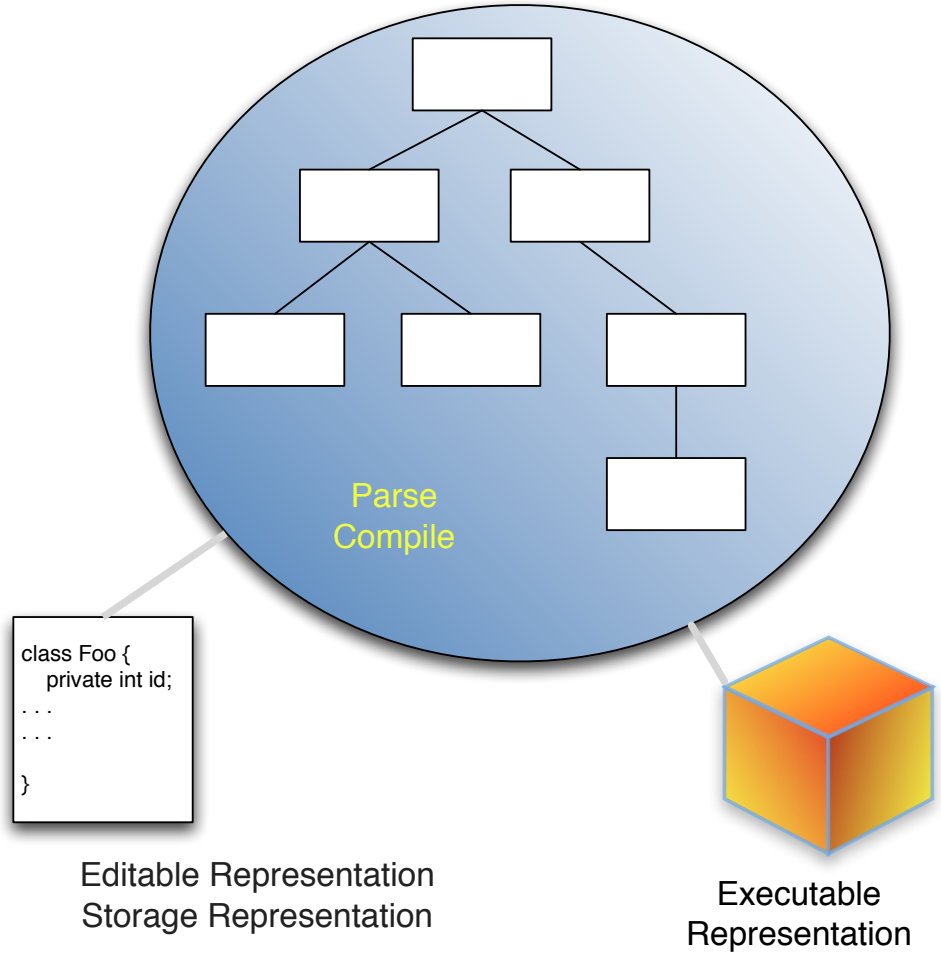
a tool that supports language oriented programming

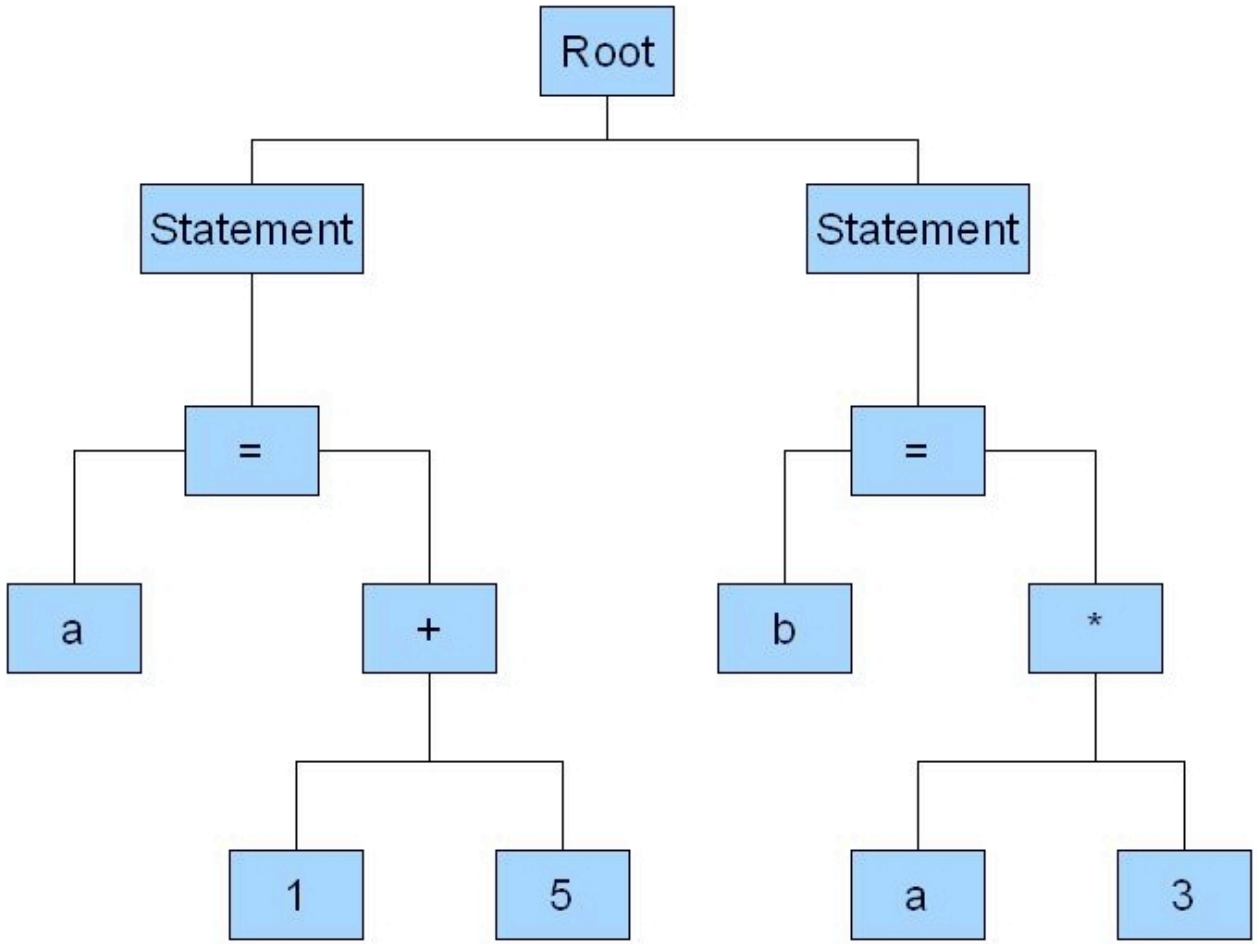
Intentional Software (Charles Simonyi)

Xtext (openArchitectureWare)

MPS (JetBrains)

compilation since cs-101



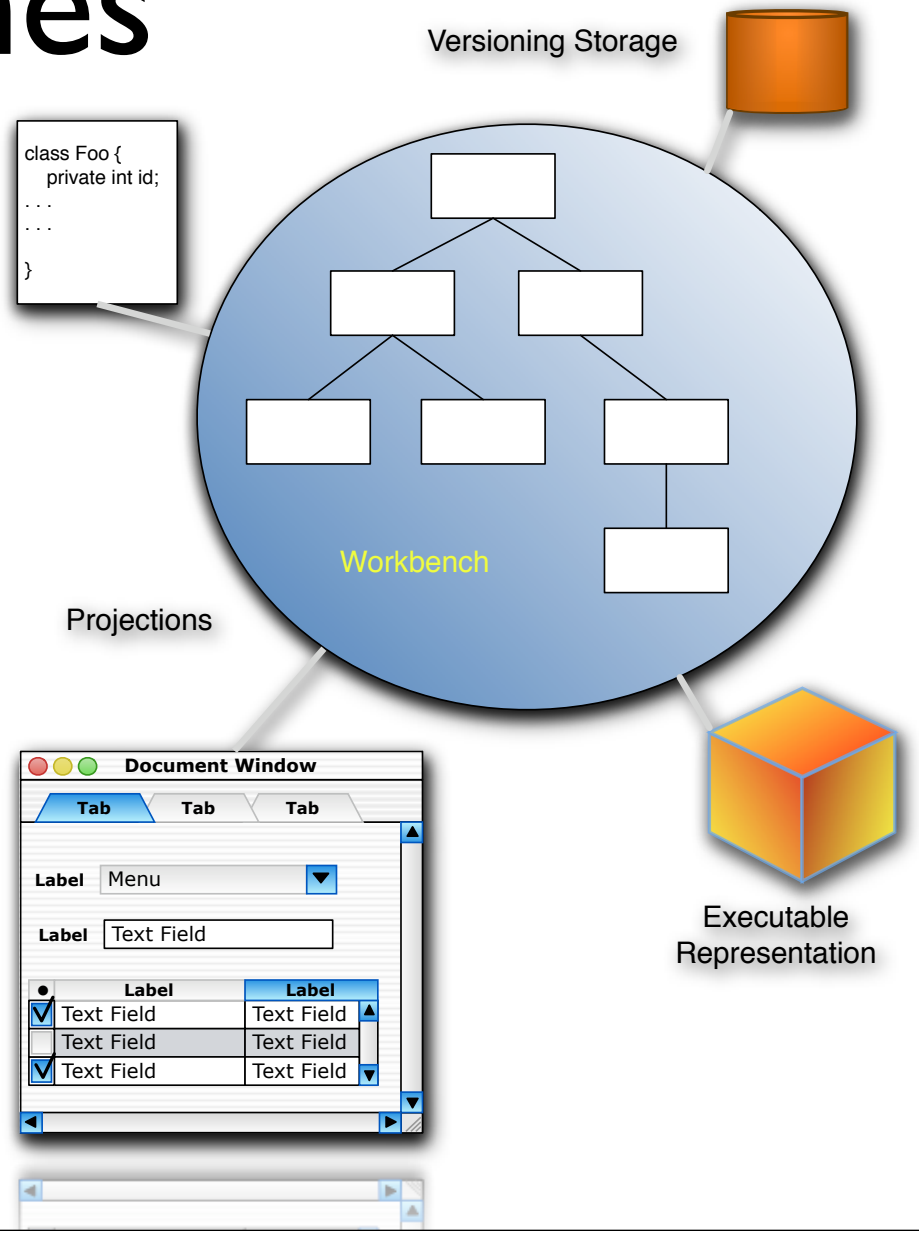

Root
Statement
Statement
=
=
a
+
b
*
1
5
a
3

“post-  ide’s”

first java ide to edit the abstract syntax directly

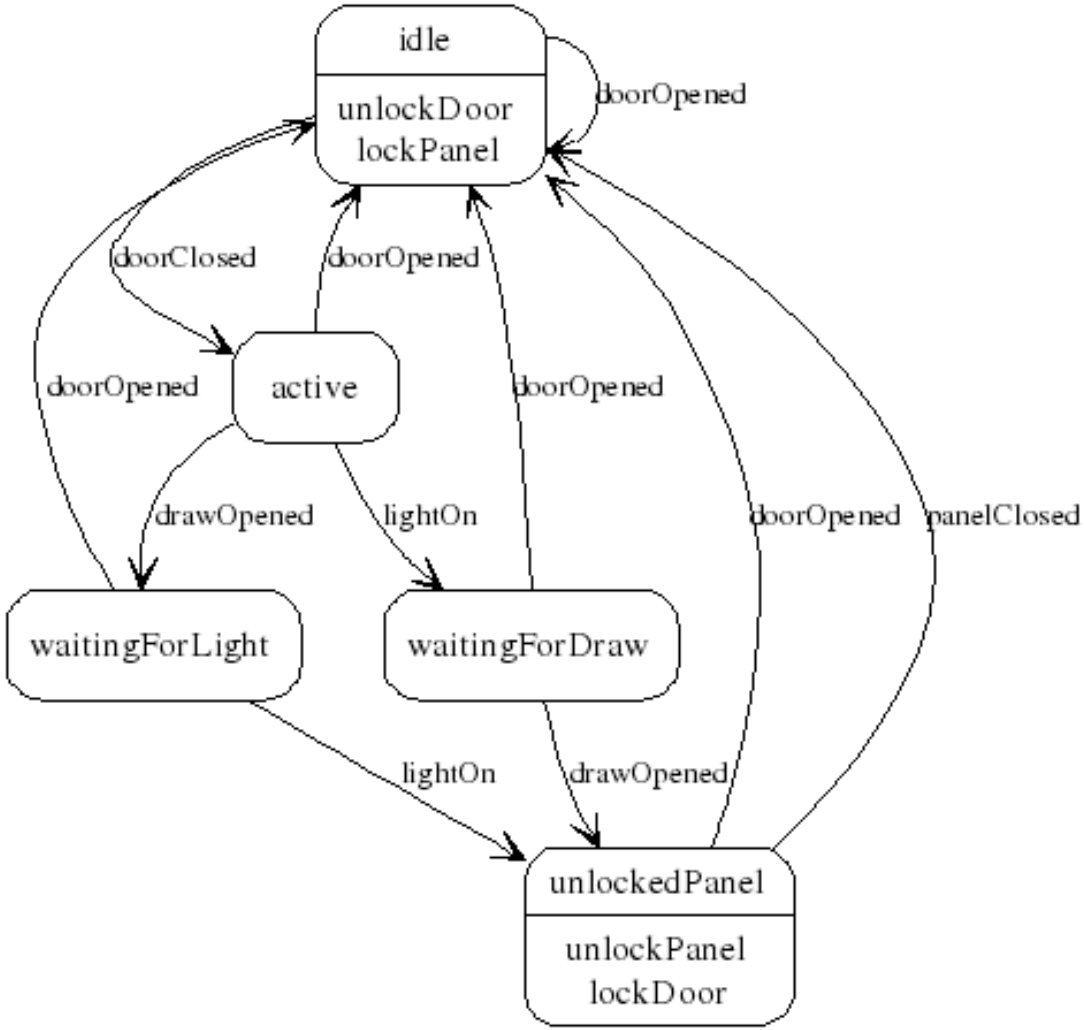
enables refactoring

workbenches



grammars

Ms. Grant's secret panel



the DSL

```
events
  doorClosed  D1CL
  drawOpened  D2OP
end

commands
  unlockPanel PNUL
  lockPanel   PNLK
end
```

the grammar

```
list        : eventList commandList;
eventList   : 'events' eventDec* 'end';
eventDec    : identifier identifier;
commandList : 'commands' commandDec* 'end';
commandDec  : identifier identifier;
```

```
events
  doorClosed  D1CL
  drawOpened  D2OP
end

commands
  unlockPanel PNUL
  lockPanel   PNLK
end
```

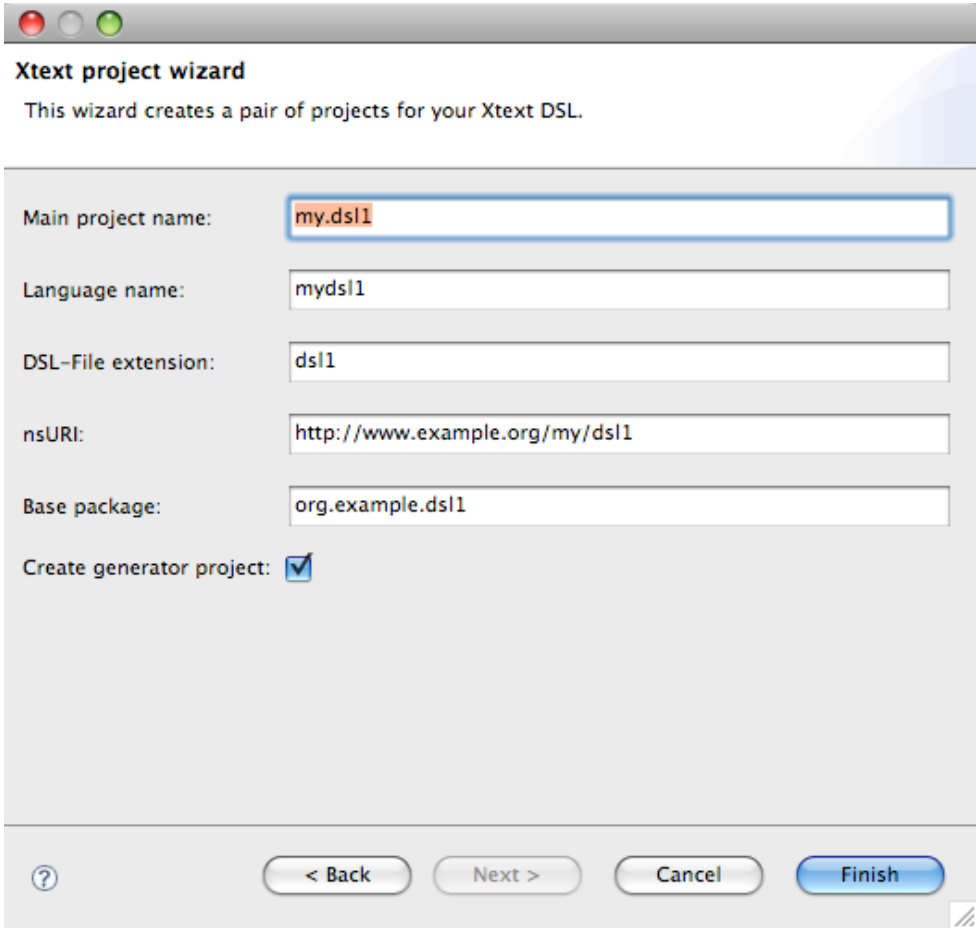
framework/tool for development of external textual DSLs

based on openArchitectureWare

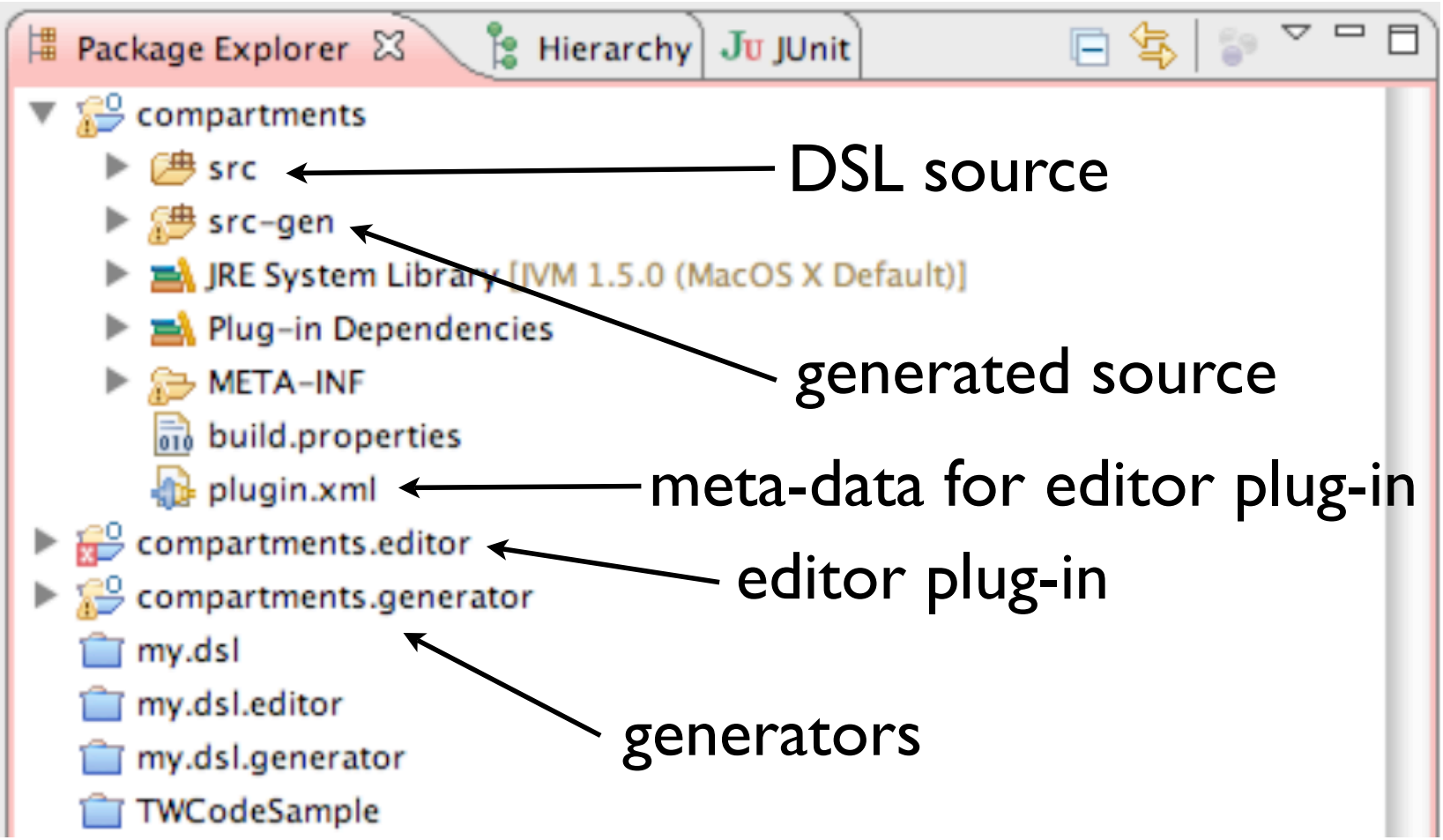
you provide a grammar in their format

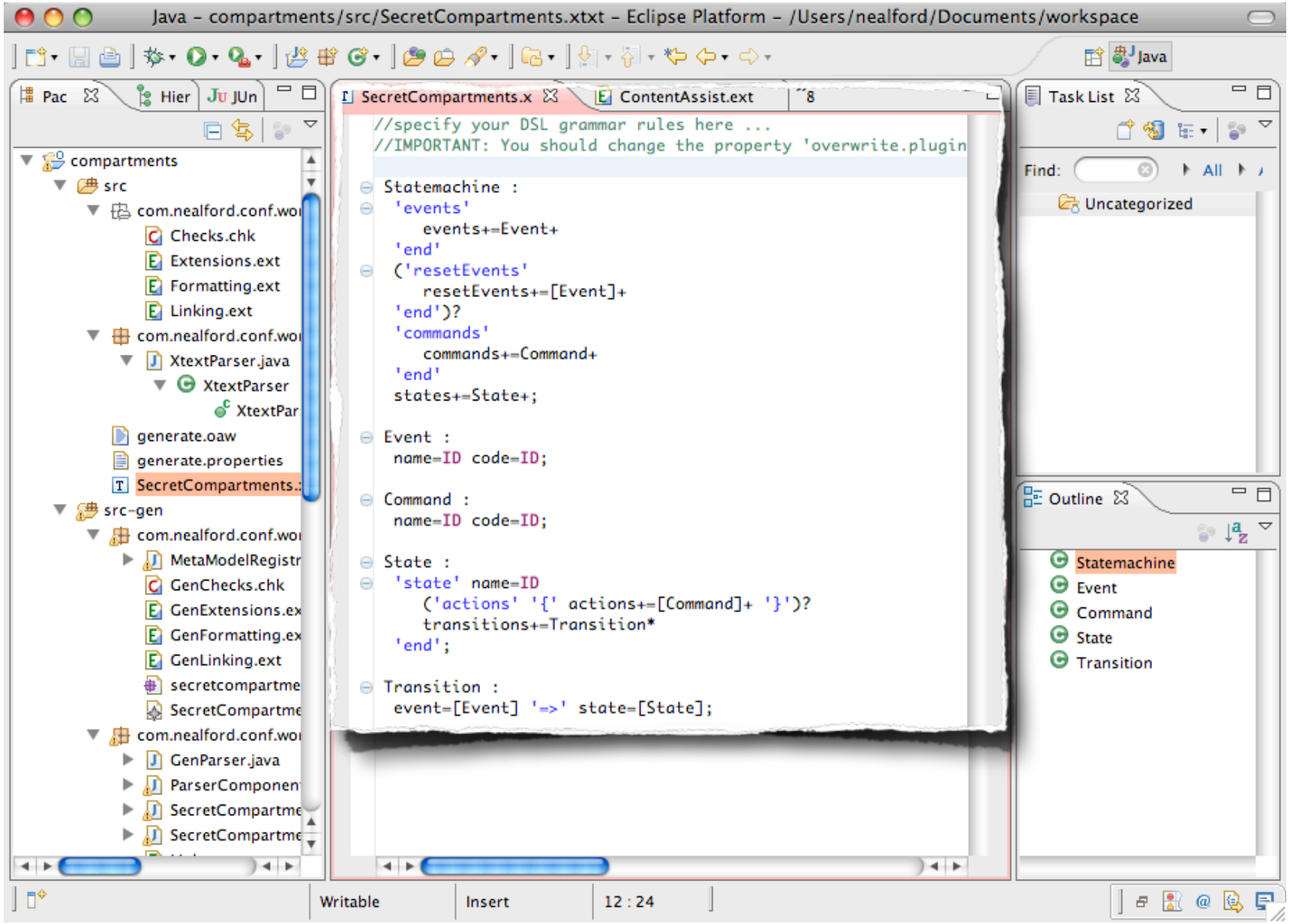
Xtext produces: ANTLR grammar, parser, abstract syntax tree meta-model, and Eclipse editor

Xtext project



Xtext artifacts



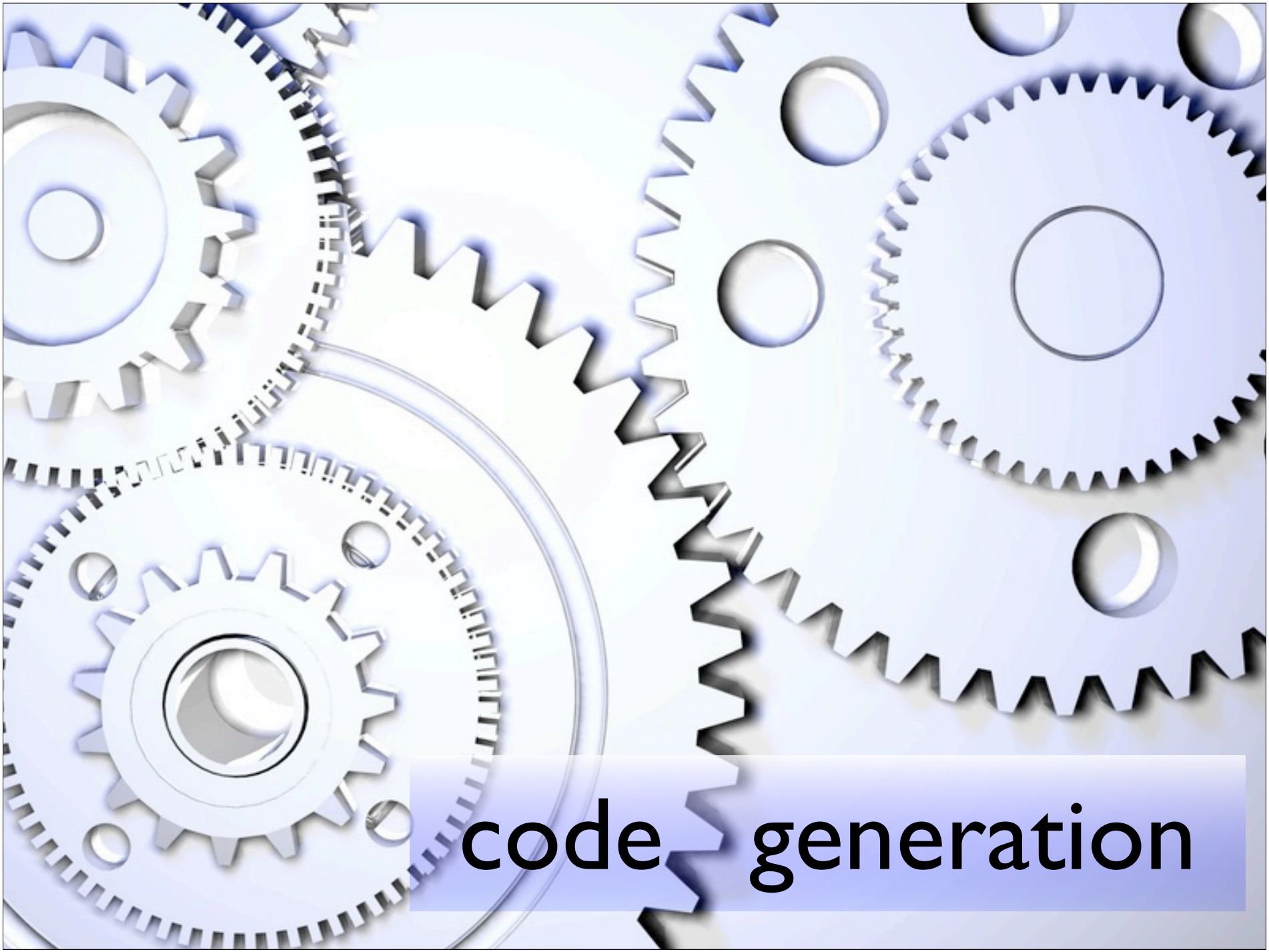

Java – compartments/src/SecretCompartments.xtxt – Eclipse Platform – /Users/nealford/Documents/workspace
Java
Pac
Hier
JUn
compartments
src
com.nealford.conf.wor
Checks.chk
Extensions.ext
Formatting.ext
Linking.ext
com.nealford.conf.wor
XtextParser.java
XtextParser
XtextPar
generate.oaw
generate.properties
SecretCompartments.
src-gen
com.nealford.conf.wor
MetaModelRegistr
GenChecks.chk
GenExtensions.ex
GenFormatting.ex
GenLinking.ext
secretcompartme
SecretCompartme
com.nealford.conf.wor
GenParser.java
ParserComponen
SecretCompartme
SecretCompartme
SecretCompartments.x
ContentAssist.ext
8
//specify your DSL grammar rules here ...
//IMPORTANT: You should change the property 'overwrite.plugin
Statemachine :
'events'
events+=Event+
'end'
('resetEvents'
resetEvents+=[Event]+
'end')?
'commands'
commands+=Command+
'end'
states+=State+;
Event :
name=ID code=ID;
Command :
name=ID code=ID;
State :
'state' name=ID
('actions' '{' actions+=[Command]+ '}')?
transitions+=Transition*
'end';
Transition :
event=[Event] '=>' state=[State];
Task List
Find:
All
Uncategorized
Outline
Statemachine
Event
Command
State
Transition
Writable
Insert
12 : 24

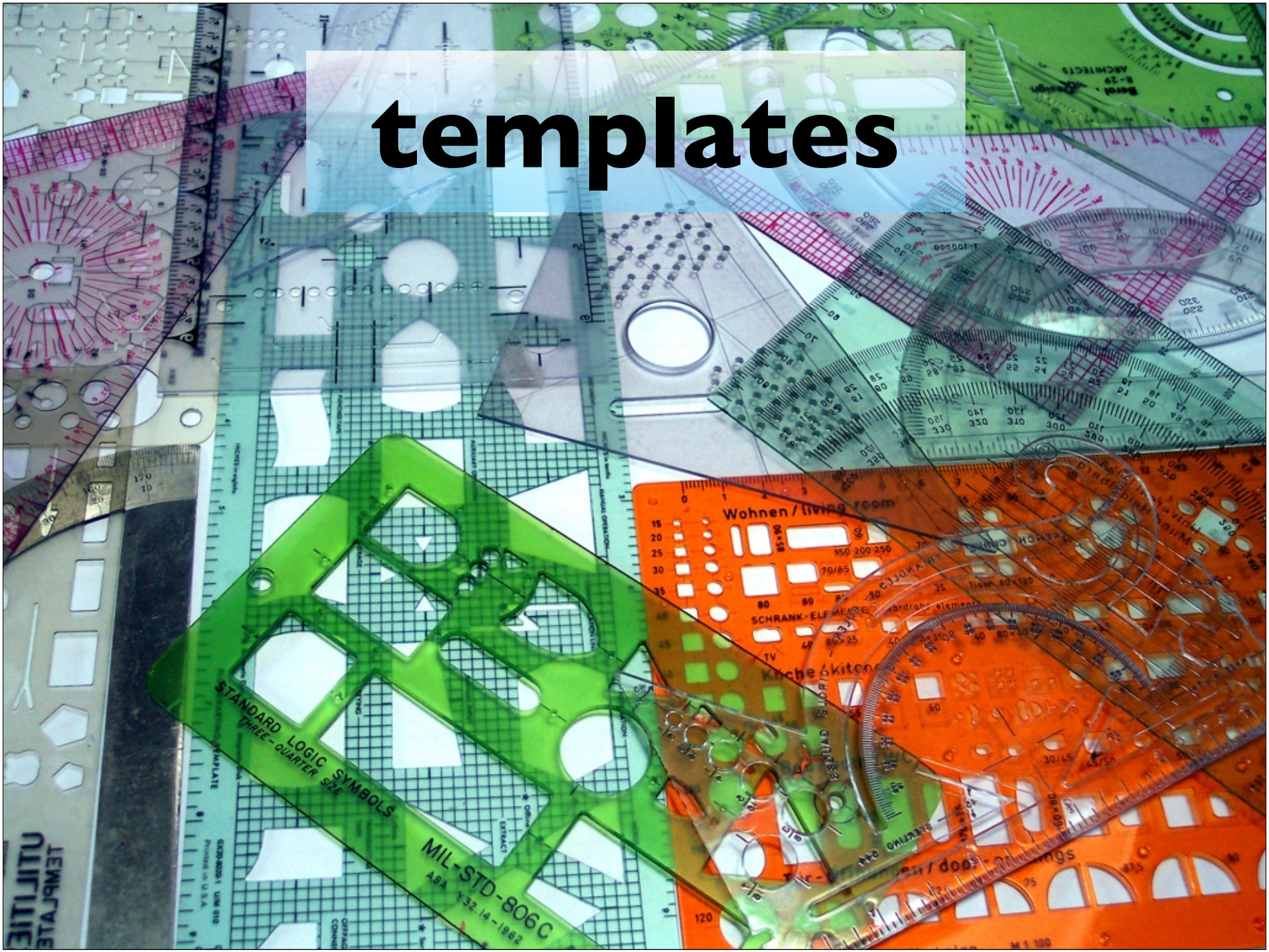
```
grammar debugMeWithAntlrWorks;
 options{backtrack=true; memoize=true;}



parse :
     result=ruleStatemachine EOF
;

ruleStatemachine  :
(('events')

(temp_events=ruleEvent )

('end')

(('resetEvents')

(temp_resetEvents=RULE_ID
)

('end')
)?

('commands')

(temp_commands=ruleCommand )

('end')

(temp_states=ruleState )
)
;
```

```
ruleEvent  :
((temp_name=RULE_ID )

(temp_code=RULE_ID )
)
;

ruleCommand  :
((temp_name=RULE_ID )

(temp_code=RULE_ID )
)
;

ruleState  :
(('state')

(temp_name=RULE_ID )

(('actions')

('{')

(temp_actions=RULE_ID
)

('}')
)?
```

ANTLR grammar

• • •


code generation


templates
STANDARD LOGIC SYMBOLS
THREE-QUARTER SIZE
MIL-STD-806C
ASA Y32.14-1962
Wohnen / living room
SCHRANK-ELEMENTE

template file

```
«IMPORT secretcompartments»

«REM»
    This is the 'main' template.
    You should replace the type 'Object' with your respective meta type.
«ENDREM»
«DEFINE main FOR Object»

«ENDDEFINE»
```

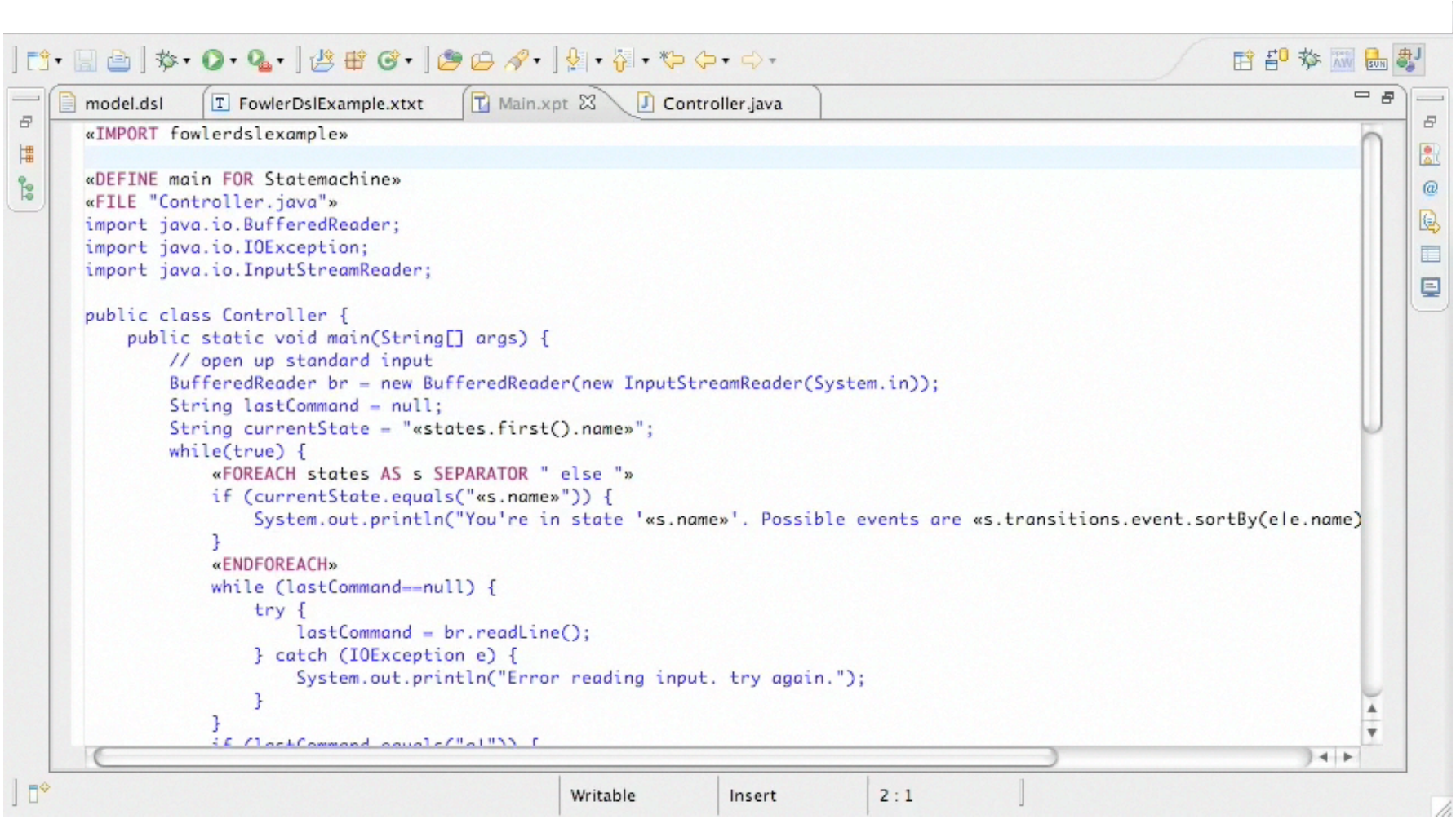
template

```
«FOREACH states AS s »
else if (currentState.equals("«s.name»")) {
    «FOREACH s.transitions AS t SEPARATOR ' else '»
    if (lastCommand.equals("«t.event.name»")) {
        currentState = "«t.state.name»";
        «FOREACH t.state.actions AS a»
        System.out.println("«a.name»!");
        «ENDFOREACH»
    }
    «ENDFOREACH»
    else {
        System.out.println("Unkown command '"+lastCommand+"'. Try again.");
    }
}
«ENDFOREACH»
lastCommand=null;
```

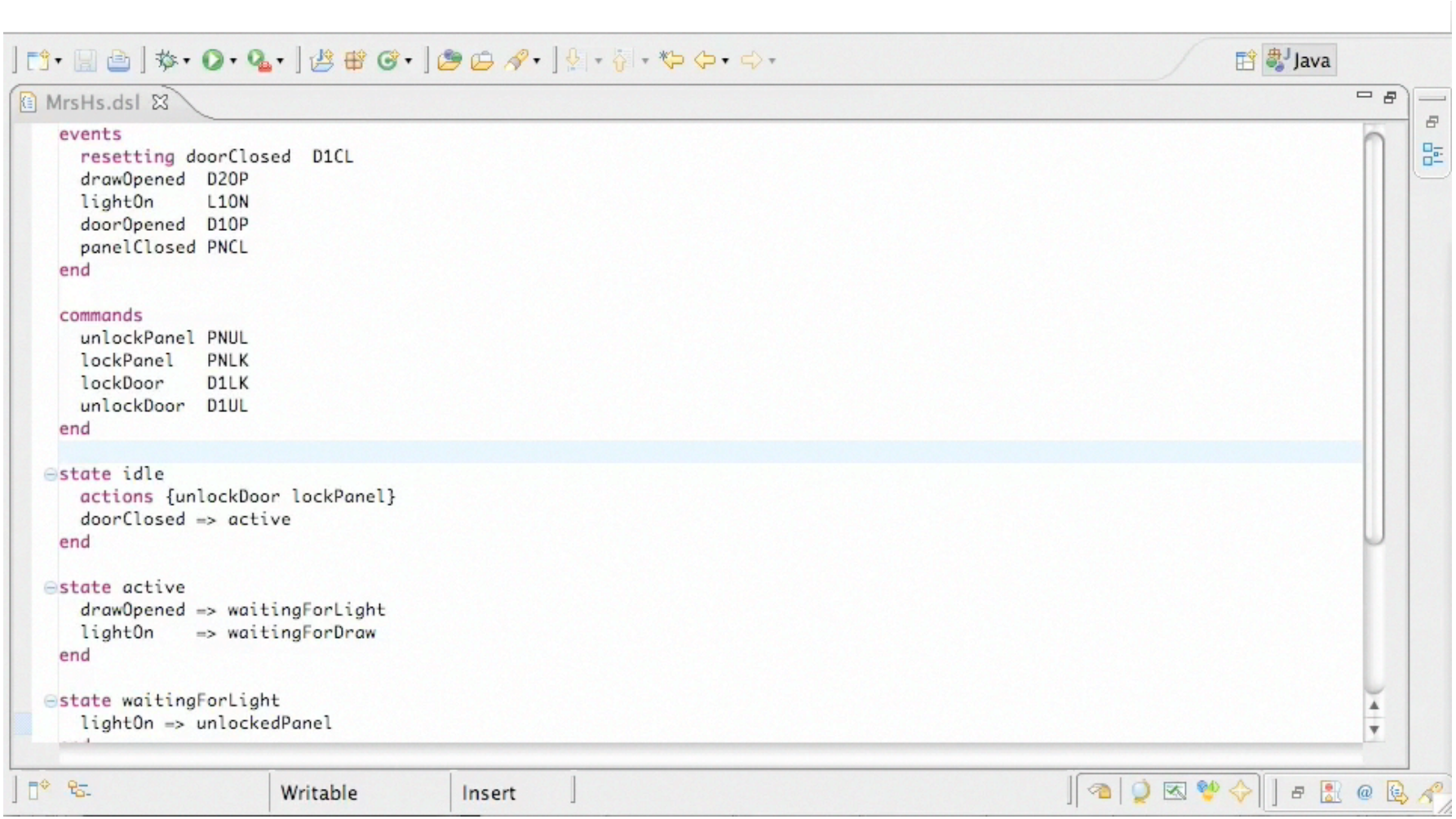
main()

```
public class Controller {
    public static void main(String[] args) {
        // open up standard input
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        String lastCommand = null;
        String currentState = "idle";
        while(true) {

            if (currentState.equals("idle")) {
                System.out.println("You're in state 'idle'. Possible events are [doorClosed]. Type 'q!' to escape.");
            }
             else
            if (currentState.equals("active")) {
                System.out.println("You're in state 'active'. Possible events are [drawOpened, lightOn]. Type 'q!' to
            }
```

templates


model.dsl
FowlerDslExample.xtxt
Main.xpt
Controller.java
«IMPORT fowlerdslexample»
«DEFINE main FOR Statemachine»
«FILE "Controller.java"»
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
public class Controller {
public static void main(String[] args) {
// open up standard input
BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
String lastCommand = null;
String currentState = "«states.first().name»";
while(true) {
«FOREACH states AS s SEPARATOR " else "»
if (currentState.equals("«s.name»")) {
System.out.println("You're in state '«s.name»'. Possible events are «s.transitions.event.sortBy(ele.name)
}
«ENDFOREACH»
while (lastCommand==null) {
try {
lastCommand = br.readLine();
} catch (IOException e) {
System.out.println("Error reading input. try again.");
}
}
Writable
Insert
2 : 1

eclipse editor



Xtext

most mature of the text-based DSL tools

still extricating itself from openArchitectureware & MDA background

heavily Eclipse based (editors, project structure, etc.)

efficient generation of DSL, grammar, code generation, & templates

nice editor projection (acts like source file)


BAKERY

bakery life

competition is brutal!

incentives to encourage repeat customers...

...but the other guys do the same thing

flexible business rules

easy to define & change

establishing profiles

```
Profile comesInRarely = new ProfileImpl();
comesInRarely
        .frequency(2)
        .monthlySpending(20);

Profile everyDay = new ProfileImpl();
everyDay
        .member()
        .frequency(25)
        .monthlySpending(500);
```

discounts

```
rules.add()
     .basedOn(comesInRarely)
     .forMembership(5.0)
     .forMonthlySpendingOf(15, 5.0)
     .forNumberOfVisits(10, 5.0);

rules.add()
     .basedOn(everyDay)
     .forMembership(10.0)
     .forMonthlySpendingOf(100, 10.0)
     .forNumberOfVisits(20, 10.0);
```

discount

```
public interface Discount {
    Discount basedOn(Profile profile);
    Discount forMembership(double discount);
    Discount forNumberOfVisits(int numberOfVisits,
            double discount);
    Discount forMonthlySpendingOf(int monthlySpending,
            double discount);

    double getDiscount();
    String toString();
}
```

discount implementation

```
public Discount forMembership(double discount) {
    _discountForMembership = discount;
    return this;
}

public DiscountImpl forNumberOfVisits(int numberOfVisits,
                                      double discount) {
    _numberOfVisits = numberOfVisits;
    _discountForVisits = discount;
    return this;
}

public Discount forMonthlySpendingOf(int monthlySpending,
                                     double discount) {
    _monthlySpending = monthlySpending;
    _discountForMonthlySpending = discount;
    return this;
}
```

getDiscount()

```
public double getDiscount() {
    double discount = 0.0;
    if (_profile.isMember())
        discount += _discountForMembership;
    if (_profile.getFrequency() > _numberOfVisits)
        discount += _discountForVisits;
    if (_profile.getMonthlySpending() > _monthlySpending)
        discount += _discountForMonthlySpending;
    return discount;
}
```

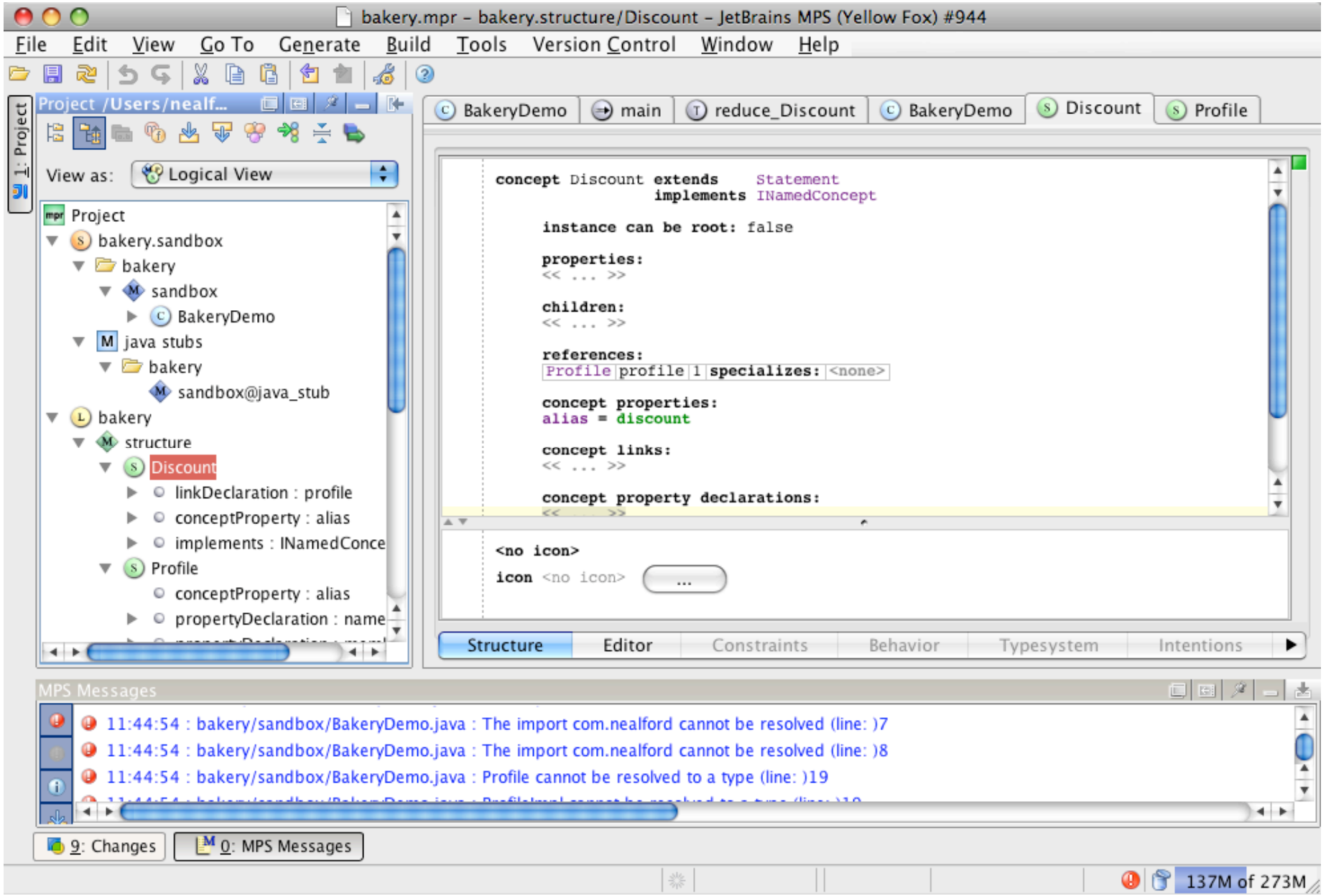
internal project by JetBrains

based on lessons learned with IntelliJ

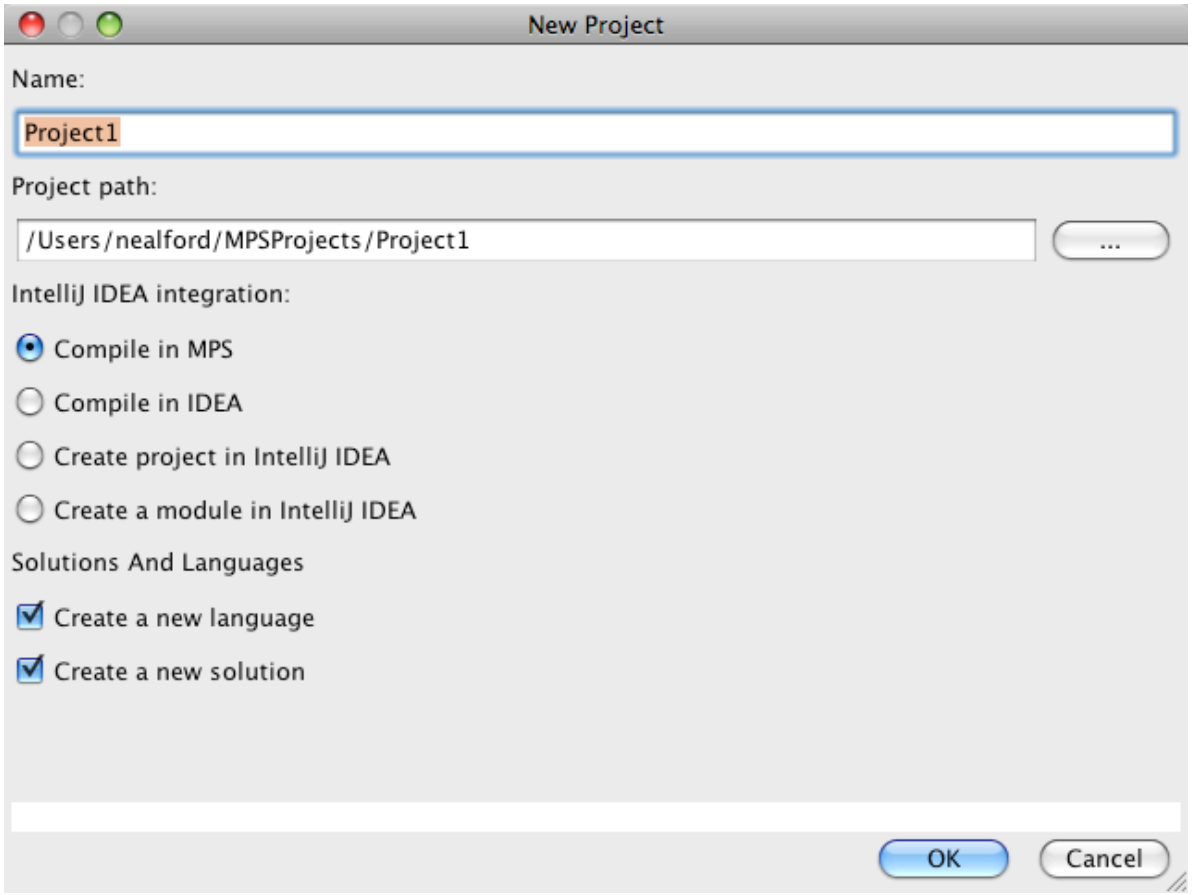
used internally to build new products

created by Sergey Dmitriev & Konstantin Solomatov

actively used now for 3+ years


bakery.mpr – bakery.structure/Discount – JetBrains MPS (Yellow Fox) #944
File Edit View Go To Generate Build Tools Version Control Window Help
1: Project
Project /Users/nealf...
View as: Logical View
Project
bakery.sandbox
bakery
sandbox
BakeryDemo
java stubs
bakery
sandbox@java_stub
bakery
structure
Discount
linkDeclaration : profile
conceptProperty : alias
implements : INamedConce
Profile
conceptProperty : alias
propertyDeclaration : name
BakeryDemo
main
reduce_Discount
BakeryDemo
Discount
Profile
concept Discount extends Statement
implements INamedConcept
instance can be root: false
properties:
<< ... >>
children:
<< ... >>
references:
Profile profile 1 specializes: <none>
concept properties:
alias = discount
concept links:
<< ... >>
concept property declarations:
<< ... >>
<no icon>
icon <no icon>
...
Structure
Editor
Constraints
Behavior
Typesystem
Intentions
MPS Messages
11:44:54 : bakery/sandbox/BakeryDemo.java : The import com.nealford cannot be resolved (line:)7
11:44:54 : bakery/sandbox/BakeryDemo.java : The import com.nealford cannot be resolved (line:)8
11:44:54 : bakery/sandbox/BakeryDemo.java : Profile cannot be resolved to a type (line:)19
9: Changes
0: MPS Messages
137M of 273M

new MPS project



concepts

fundamental building block

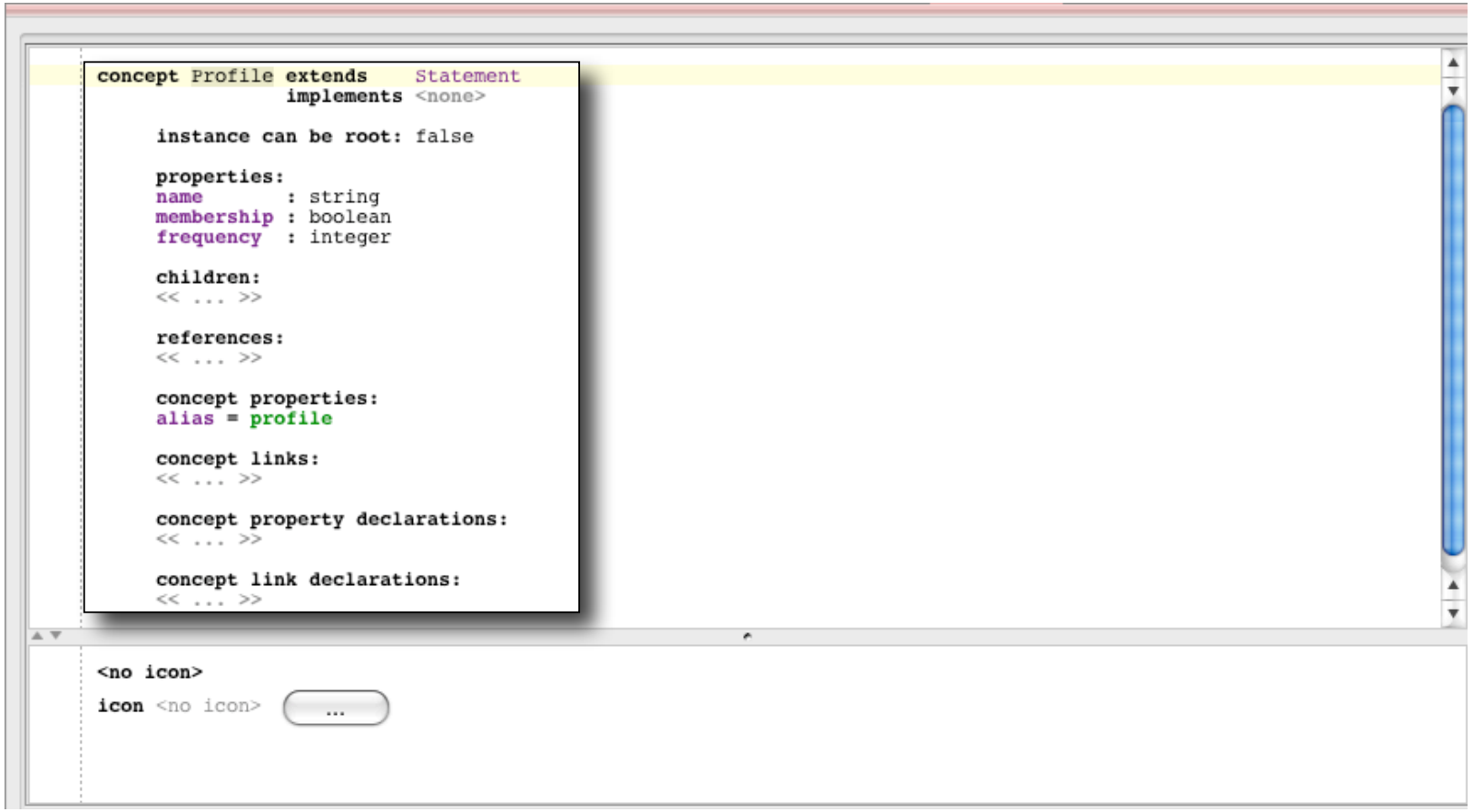
like a class: data, behavior, encapsulation

also editors, code generators

implemented in *base language*

properties, events, methods, and links

concept declaration

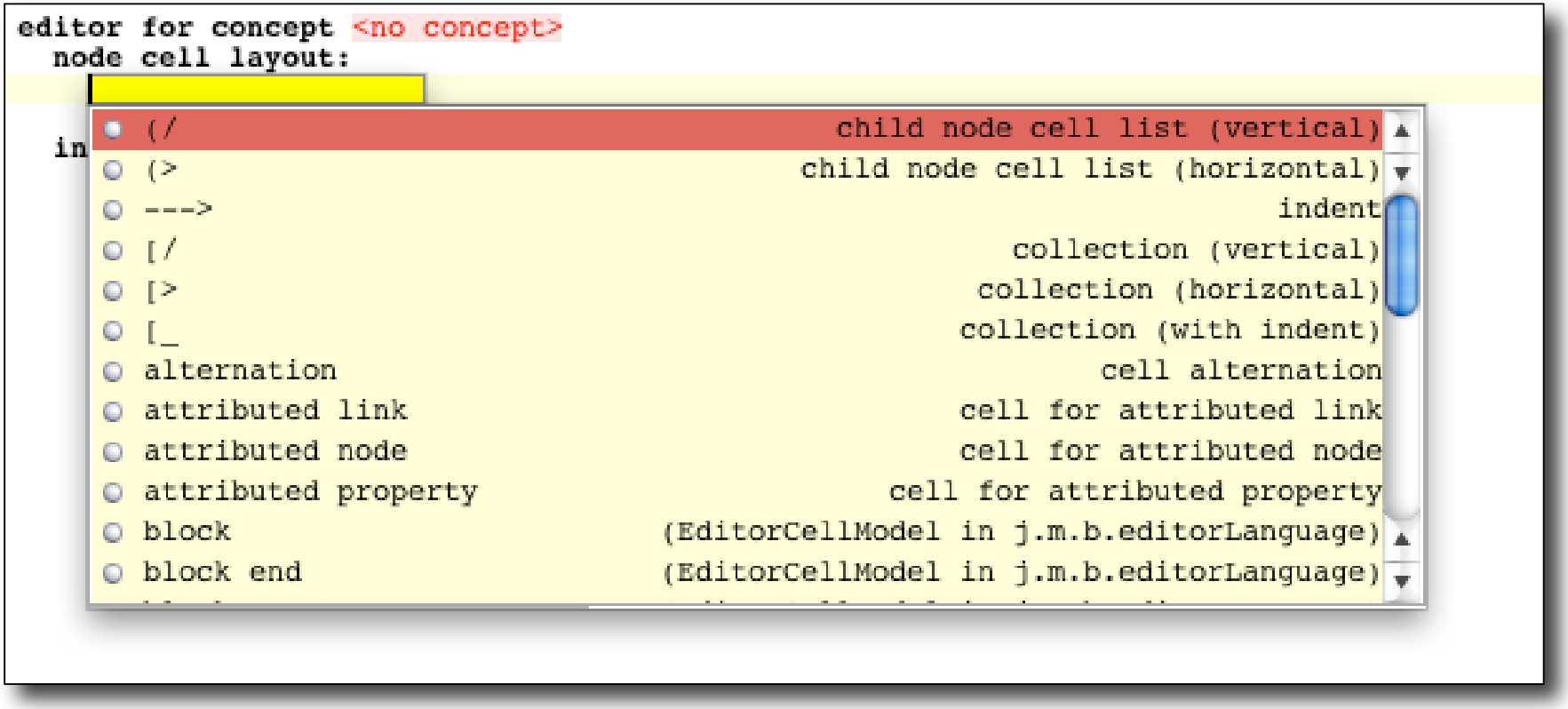


concept editor

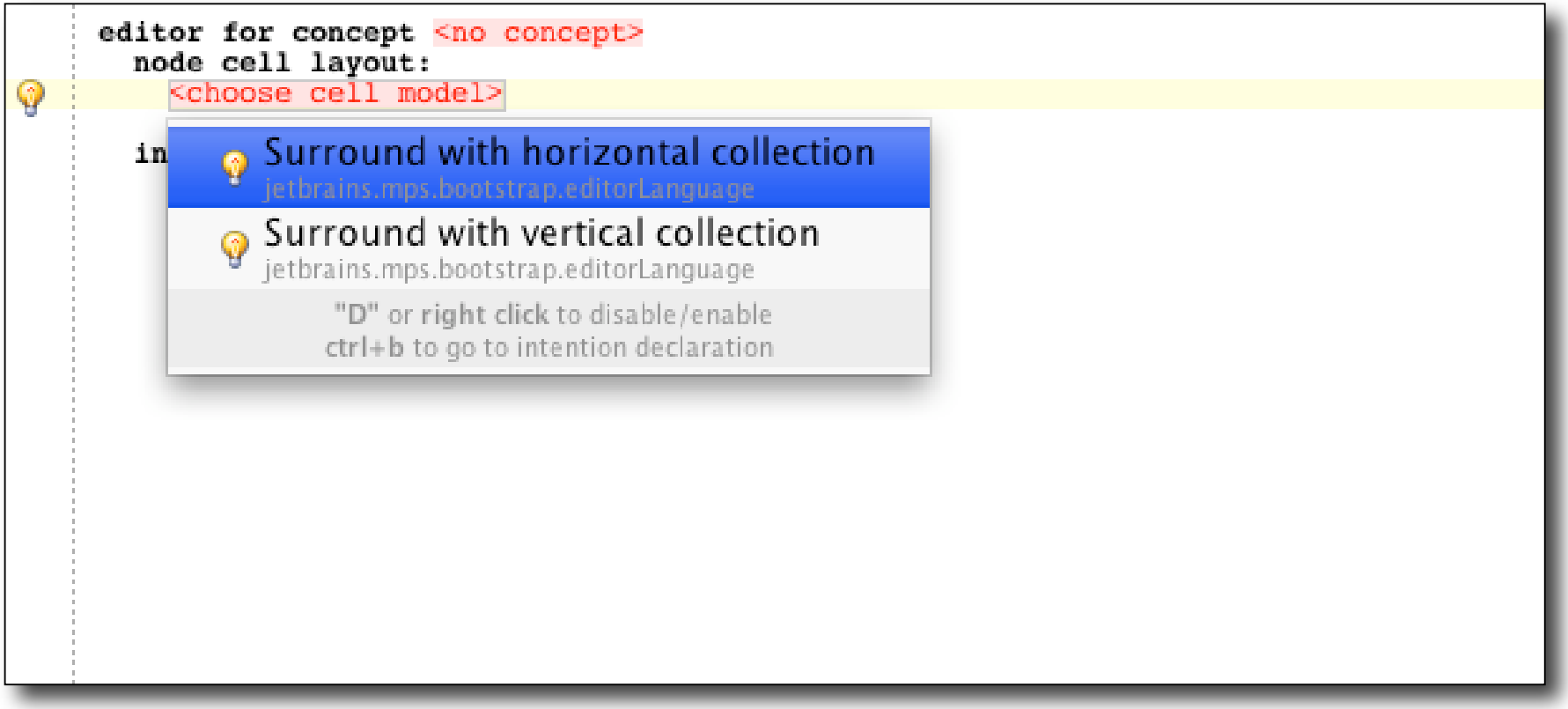
```
editor for concept Profile
  node cell layout:
    [/
      [> profile { name } <]
      [> ---> [/
                [> membership { membership } <]
                [> frequency { frequency } <]
              \]
    \]

  inspected cell layout:
    <choose cell model>
```

```
concept Profile extends     Statement
                implements  <none>

    instance can be root: false

    properties:
    name       : string
    membership : boolean
    frequency  : integer

    children:
    << ... >>

    references:
    << ... >>

    concept properties:
    alias = profile

    concept links:
    << ... >>

    concept property declarations:
    << ... >>

    concept link declarations:
    << ... >>
```

concept editor



smart help


editor for concept <no concept>
node cell layout:
<choose cell model>
in
Surround with horizontal collection
jetbrains.mps.bootstrap.editorLanguage
Surround with vertical collection
jetbrains.mps.bootstrap.editorLanguage
"D" or right click to disable/enable
ctrl+b to go to intention declaration

discount

```
concept Discount extends     Statement
                 implements  INamedConcept

    instance can be root: false

    properties:
    << ... >>

    children:
    << ... >>

    references:
    Profile | profile | 1 | specializes: | <none>

    concept properties:
    alias = discount

    concept links:
    << ... >>

    concept property declarations:
    << ... >>

    concept link declarations:
    << ... >>
```

reduction rule mapping

```
mapping configuration main
top-priority group    false

mapping labels:
  << ... >>

conditional root rules:
  << ... >>

mapping rules:
  << ... >>

weaving rules:
  << ... >>

reduction rules:
  [concept    Profile  ]--> reduce_Profile
  [inheritors false    ]
  [condition  <always> ]

  [concept    Discount ]--> reduce_Discount
  [inheritors false    ]
  [condition  <always> ]


abandon roots:
  << ... >>

pre-processing scripts:
  << ... >>

post-processing scripts:
  << ... >>
```

reduction template

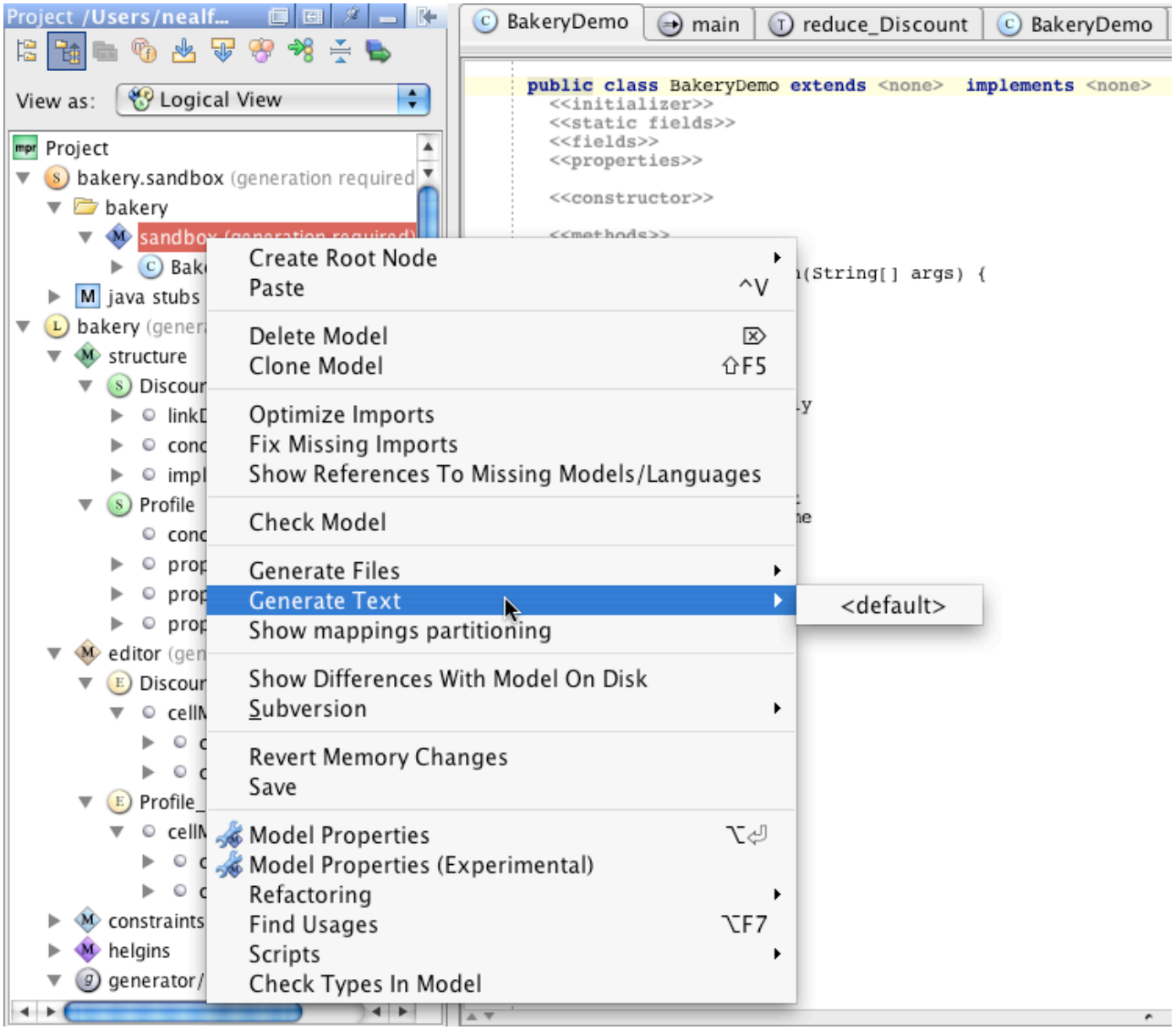
```
template reduce_Profile
source   Profile

Ctrl-Shift-F : create template fragment
Ctrl-Shift-M : create macro

content node:
<TF  [Profile$[profile] = new ProfileImpl( ) {                 ; ] TF>
                                        {
                                          $IF$[this.member( );]
                                          this.frequency($[1]);
                                        }
                                        <add members (ctrl+space)>
                                      }
```

discount template

```
template reduce_Discount
source   Discount

Ctrl-Shift-F : create template fragment
Ctrl-Shift-M : create macro

content node:
{
  Profile p;
  <TF  Discount$[name] = new DiscountImpl( ) {                ; TF>
                          { this.basedOn(->$[p]);}
                          <add members (ctrl+space)>
                       }
}
```

using the editor

```
public class BakeryDemo extends <none>  implements <none>  {
  <<initializer>>
  <<static fields>>
  <<fields>>
  <<properties>>

  <<constructor>>

  <<methods>>

  public static void main(String[] args) {
    profile rarely
      membership false
      frequency 25
    profile allTheTime
      membership true
      frequency 30
    discount basedOnRarely
      profile rarely
    profile all_the_time
      membership true
      frequency 12
  }


  <<static inner classifiers>>

}
```


Project /Users/nealf...
View as: Logical View
Project
bakery.sandbox (generation required
bakery
java stubs
structure
editor
constraints
helgins
BakeryDemo
main
reduce_Discount
BakeryDemo
public class BakeryDemo extends <none> implements <none>
<<initializer>>
<<static fields>>
<<fields>>
<<properties>>
<<constructor>>
Create Root Node
Paste ^V
Delete Model
Clone Model ⇧F5
Optimize Imports
Fix Missing Imports
Show References To Missing Models/Languages
Check Model
Generate Files
Generate Text
Show mappings partitioning
<default>
Show Differences With Model On Disk
Subversion
Revert Memory Changes
Save
Model Properties ⌥⏎
Model Properties (Experimental)
Refactoring
Find Usages ⌥F7
Scripts
Check Types In Model

```
package bakery.sandbox;

/*Generated by MPS */

import com.nealford.conf.dsl.bakery.Profile;
import com.nealford.conf.dsl.bakery.ProfileImpl;
import com.nealford.conf.dsl.bakery.Discount;
import com.nealford.conf.dsl.bakery.DiscountImpl;

public class BakeryDemo {

  public static void main(String[] args) {
    Discount basedOnRarely = new DiscountImpl() {
      {
        this.basedOn(rarely);
      }

    };
    Profile all_the_time = new ProfileImpl() {
      {
        this.member();
        this.frequency(12);
      }

    };
    Discount new_discount = new DiscountImpl() {
      {
        this.basedOn(all_the_time);
      }

    };
  }

}
```

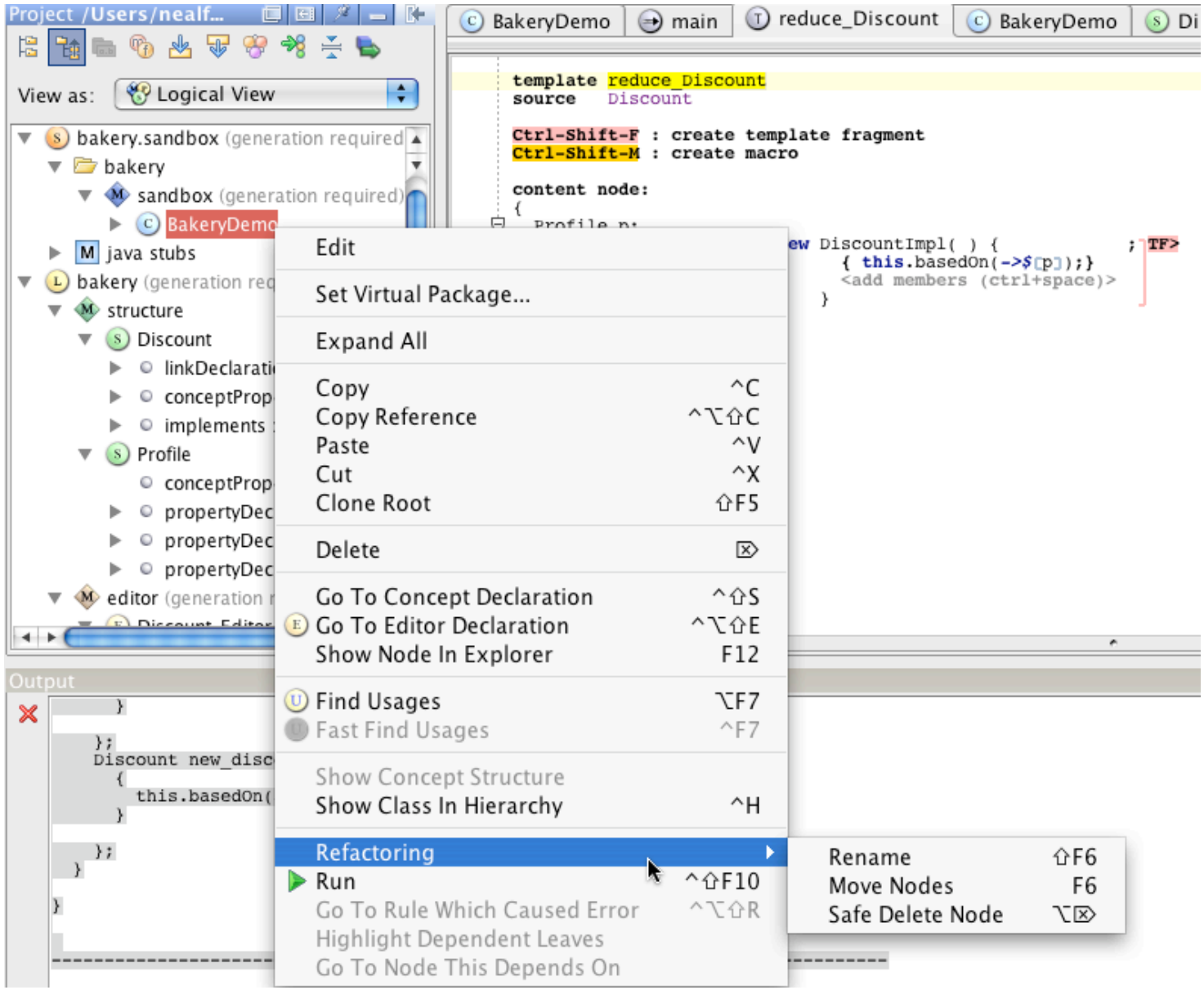
```
content node:
<TF Profile$[profile] = new ProfileImpl( ) {                    ; TF>
                          {
                            $IF$[this.member( );]
                            this.frequency($[1]);
                          }
                          <add members (ctrl+space)>
                        }
```

```
public static void main(String[] args) {
  profile rarely
    membership false
    frequency 25
  profile allTheTime
    membership true
    frequency 30
  discount basedOnRarely
    profile rarely
  profile all_the_time
    membership true
    frequency 12
  discount new_discount
    profile all_the_time
}
```

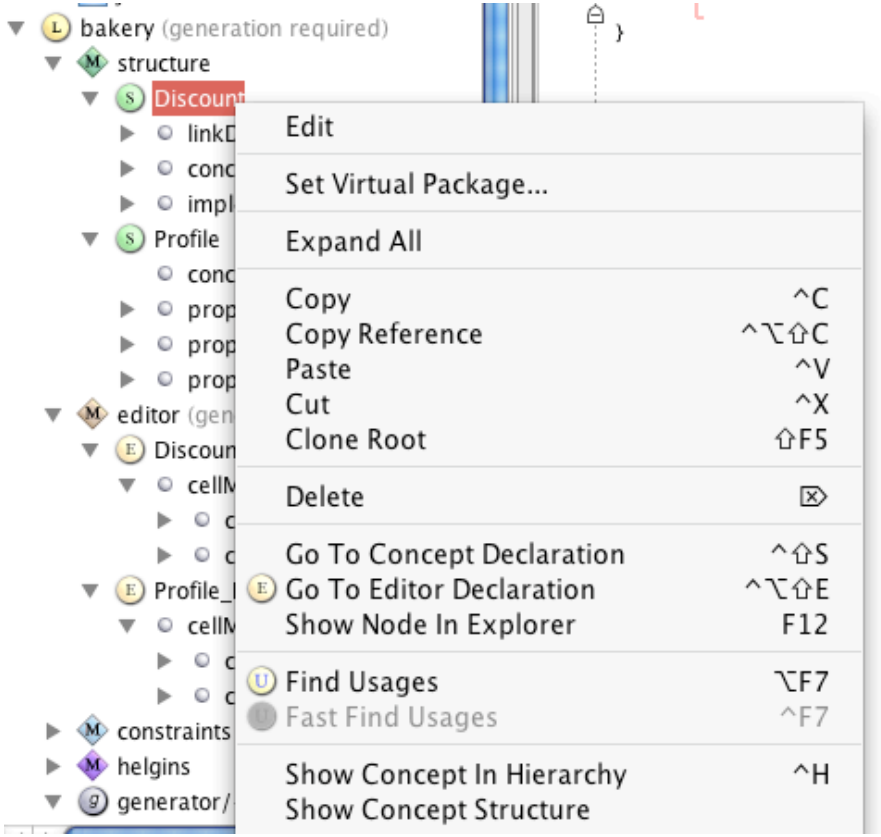
```
template reduce_Discount
source   Discount

Ctrl-Shift-F : create template fragment
Ctrl-Shift-M : create macro

content node:
{
  Profile p;
  <TF Discount$[name] = new DiscountImpl( ) {                   ; TF>
                          { this.basedOn(->$[p]);}
                          <add members (ctrl+space)>
                        }
}
```

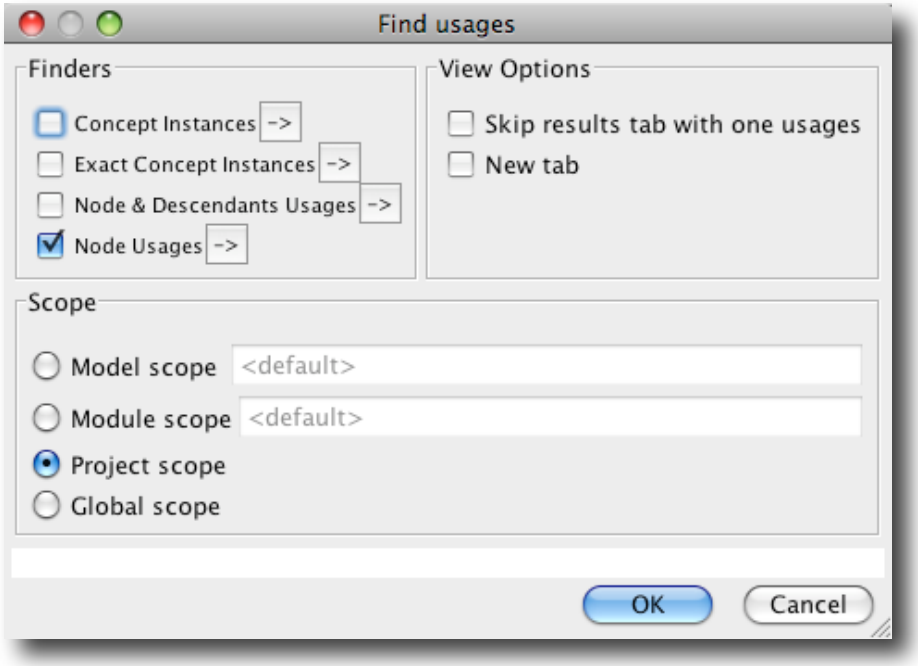
grammar?

encapsulated within

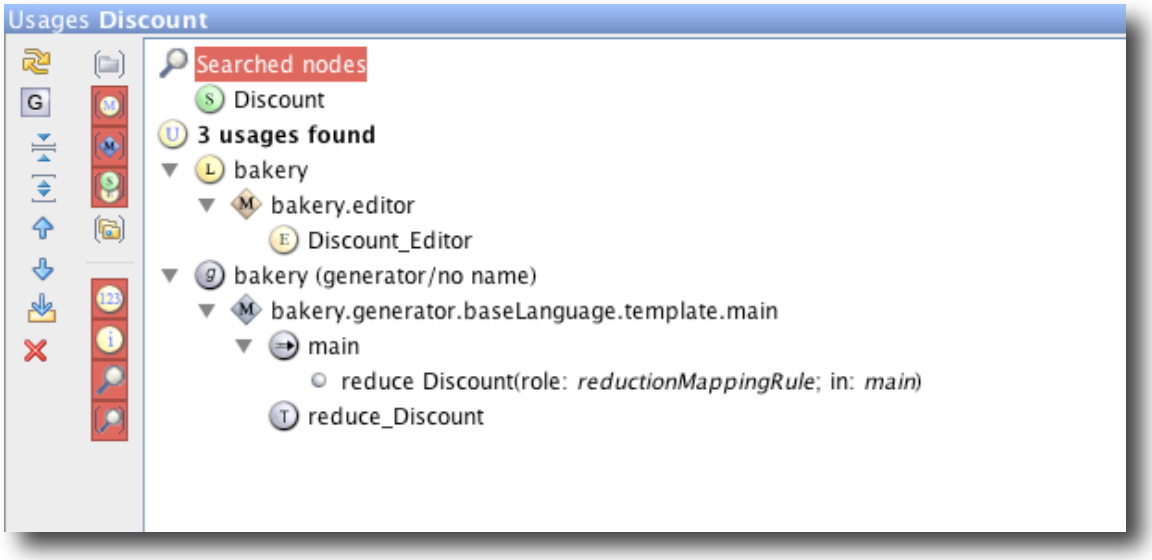

Project /Users/nealf...
View as: Logical View
bakery.sandbox (generation required)
bakery
sandbox (generation required)
BakeryDemo
java stubs
bakery (generation req
structure
Discount
linkDeclarati
conceptProp
implements
Profile
conceptProp
propertyDec
propertyDec
propertyDec
editor (generation
BakeryDemo
main
reduce_Discount
BakeryDemo
template reduce_Discount
source Discount
Ctrl-Shift-F : create template fragment
Ctrl-Shift-M : create macro
content node:
{
ew DiscountImpl( ) {
{ this.basedOn(->$[p]);}
<add members (ctrl+space)>
}
TF>
Output
Edit
Set Virtual Package...
Expand All
Copy ^C
Copy Reference ^⌥⇧C
Paste ^V
Cut ^X
Clone Root ⇧F5
Delete
Go To Concept Declaration ^⇧S
Go To Editor Declaration ^⌥⇧E
Show Node In Explorer F12
Find Usages ⌥F7
Fast Find Usages ^F7
Show Concept Structure
Show Class In Hierarchy ^H
Refactoring
Run ^⇧F10
Go To Rule Which Caused Error ^⌥⇧R
Highlight Dependent Leaves
Go To Node This Depends On
Rename ⇧F6
Move Nodes F6
Safe Delete Node ⌥⌦

find usages

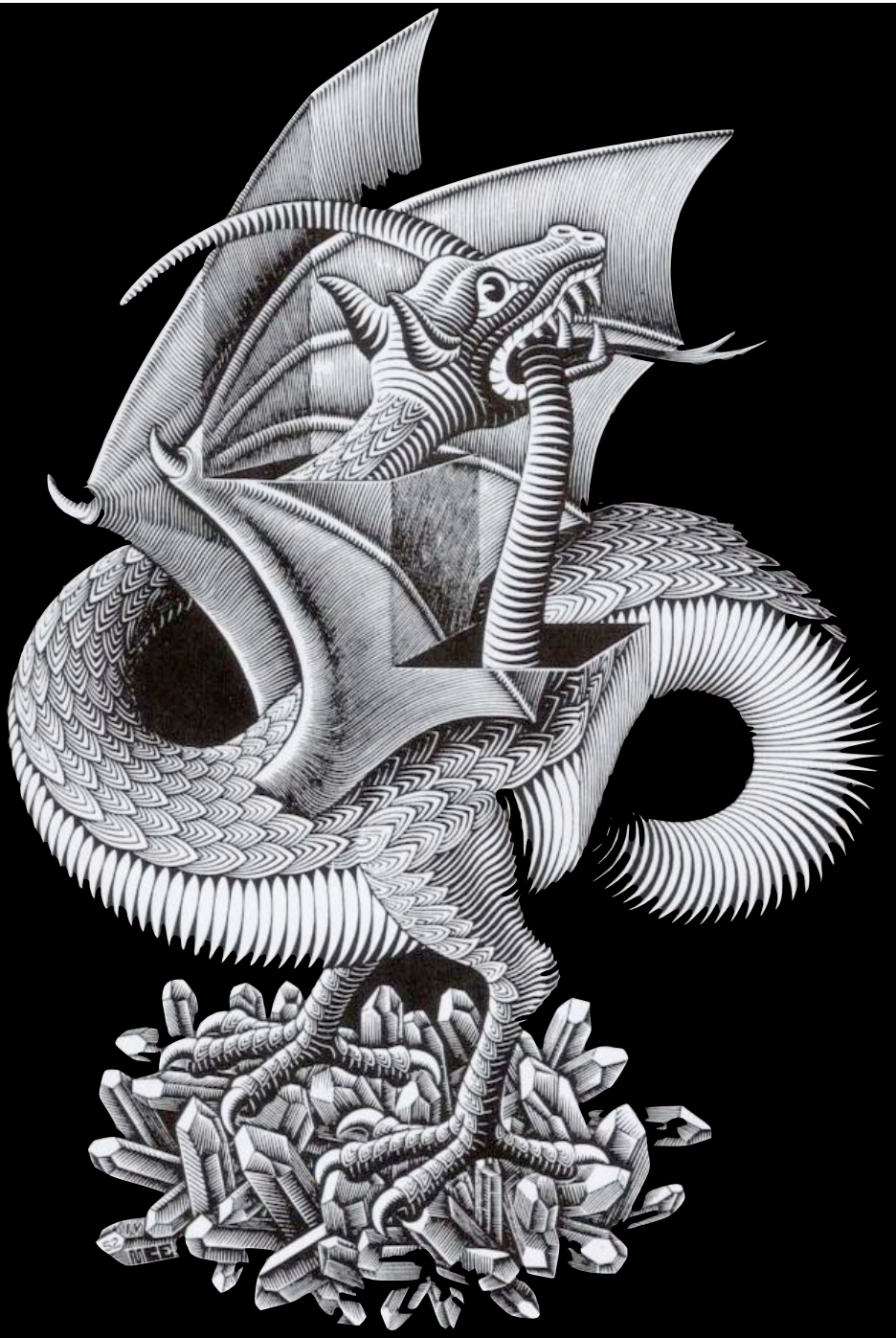

bakery (generation required)
structure
Discount
Profile
editor
constraints
helgins
generator/
Edit
Set Virtual Package...
Expand All
Copy ^C
Copy Reference ^⌥⇧C
Paste ^V
Cut ^X
Clone Root ⇧F5
Delete
Go To Concept Declaration ^⇧S
Go To Editor Declaration ^⌥⇧E
Show Node In Explorer F12
Find Usages ⌥F7
Fast Find Usages ^F7
Show Concept In Hierarchy ^H
Show Concept Structure


Find usages
Finders
Concept Instances ->
Exact Concept Instances ->
Node & Descendants Usages ->
Node Usages ->
View Options
Skip results tab with one usages
New tab
Scope
Model scope <default>
Module scope <default>
Project scope
Global scope
OK
Cancel

found usages

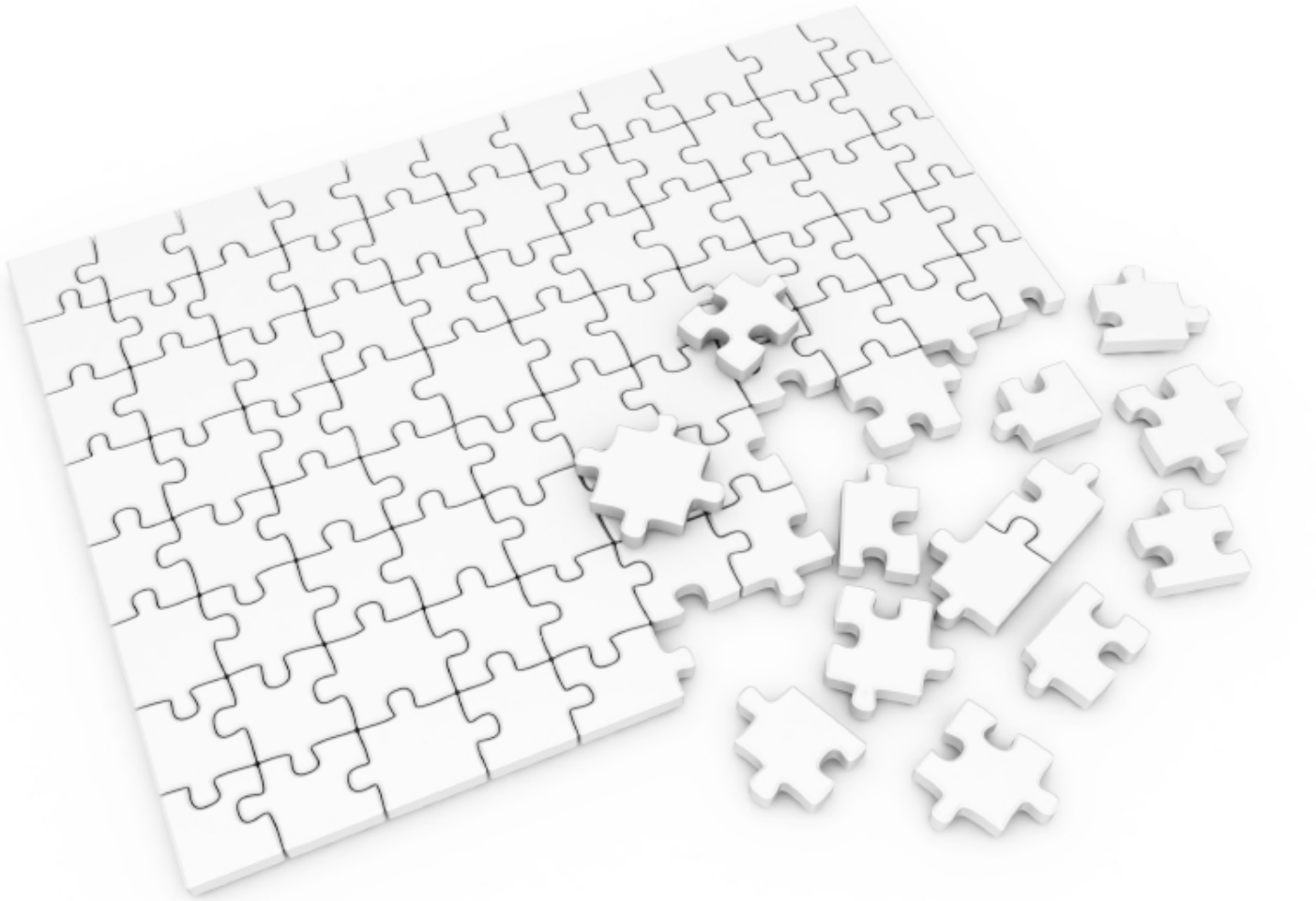


MPS is written *in* MPS

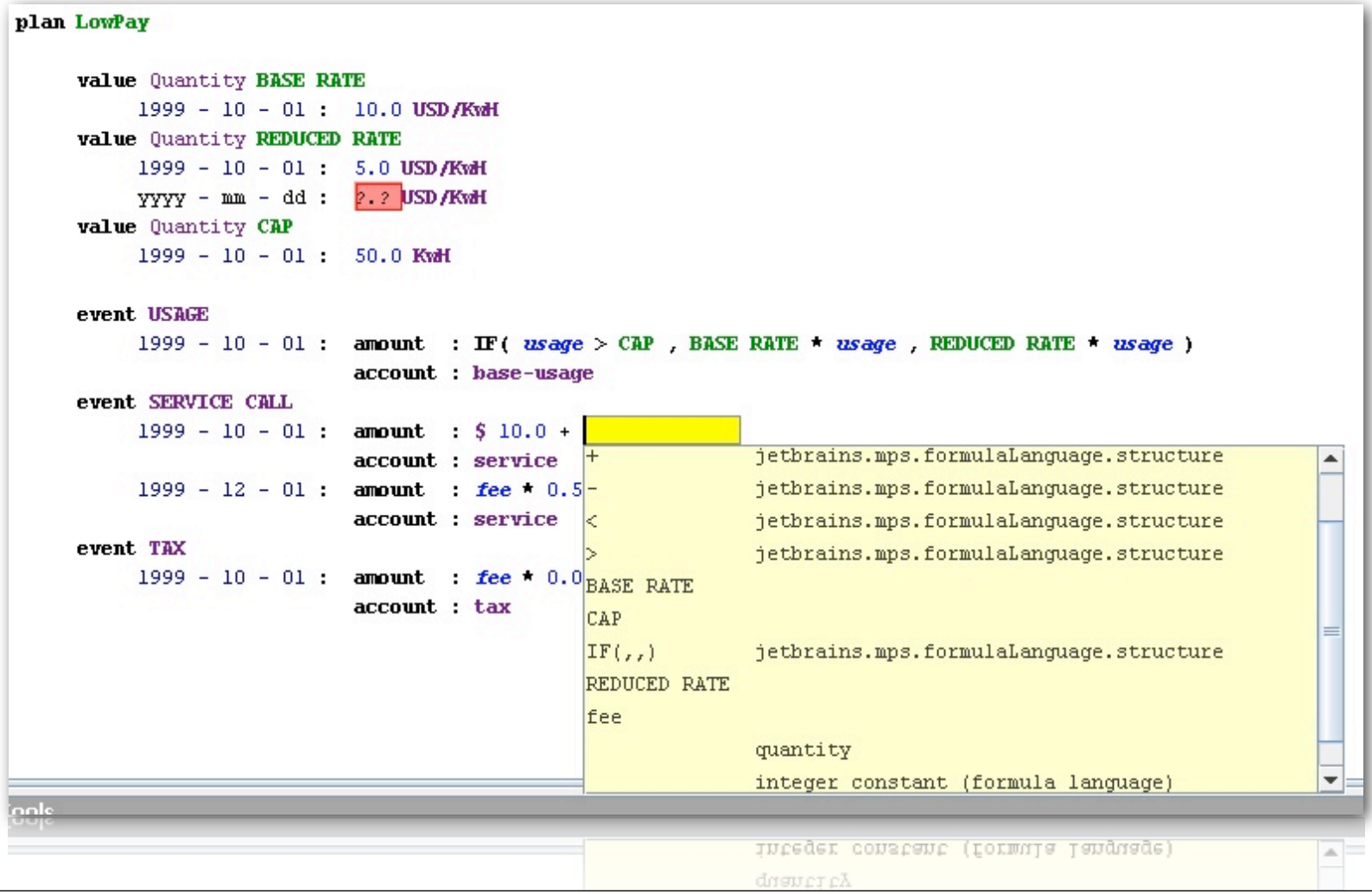


```
migration script ConvertOldDateOperations

DotExpression: Convert old date operations [migrate form b. 531]

updaters:
    description : Convert IsNotNullOperation
    for each    : IsNotNullOperation
    where       : <no condition>
    do          : (node)->void {
                    datetime fake;
                    node.replace with(<%( node.datetime)% .isNotNull >);
                  }
    ;
    description : Convert IsNullOperation
    for each    : IsNullOperation
    where       : <no condition>
    do          : (node)->void {
                    datetime fake;
                    node.replace with(<%( node.datetime)% .isNull >);
                  }
    ;
    description : Convert ToDateTimeOperation
    for each    : ToDateTimeOperation
    where       : <no condition>
    do          : (node)->void {
                    Date fake;
                    node.replace with(<%( node.expression)% .datetime>);
                  }
    ;
    description : Convert ToJavaCalendarOperation
    for each    : ToJavaCalendarOperation
    where       : <no condition>
    do          : (node)->void {
                    datetime fake;
                    node.replace with(<%( node.datetime)% .javaCalendar >);
                  }
    ;
    description : Convert ToJavaDateOperation
    for each    : ToJavaDateOperation
    where       : <no condition>
    do          : (node)->void {
                    datetime fake;
                    node.replace with(<%( node.datetime)% .javaDate >);
                  }
    ;
    description : Convert ToJodaDateTimeOperation
    for each    : ToJodaDateTimeOperation
    where       : <no condition>
    do          : (node)->void {
                    datetime fake;
                    node.replace with(<%( node.datetime)% .jodaDateTime >);
                  }
    ;
    description : Convert DateTimePropetyReference
    for each    : DateTimePropetyReference
    where       : <no condition>
    do          : (node)->void {
                    datetime fake;
                    node.replace with(<%( node.datetime)% .^( node.dateTimeProperty)^ >);
                  }
    ;
```

composition



composing DSLs



comparisons

	Xtext	MPS	Intentional
grammar	EBNF/antlr	encapsulated	?
environment	Eclipse	MPS	?
code generation	via Eclipse	via MPS or IntelliJ	internal
projections	EMF & text	text-like (for now)	rich
project ecosystem	Eclipse	MPS	?
availability	now! open source	soon open source	? commercial

questions?

please fill out the session evaluations
slides & samples available at nealford.com

NEAL FORD software architect / meme wrangler

ThoughtWorks

nford@thoughtworks.com
3003 Summit Boulevard, Atlanta, GA 30319
www.nealford.com
www.thoughtworks.com
memeagora.blogspot.com

resources

Xtext
http://wiki.eclipse.org/Xtext

MPS
http://www.jetbrains.com/mps/

Intentional Softare
http://www.intentsoft.com/

Martin Fowler's DSL Work in Progress
http://martinfowler.com/dslwip/