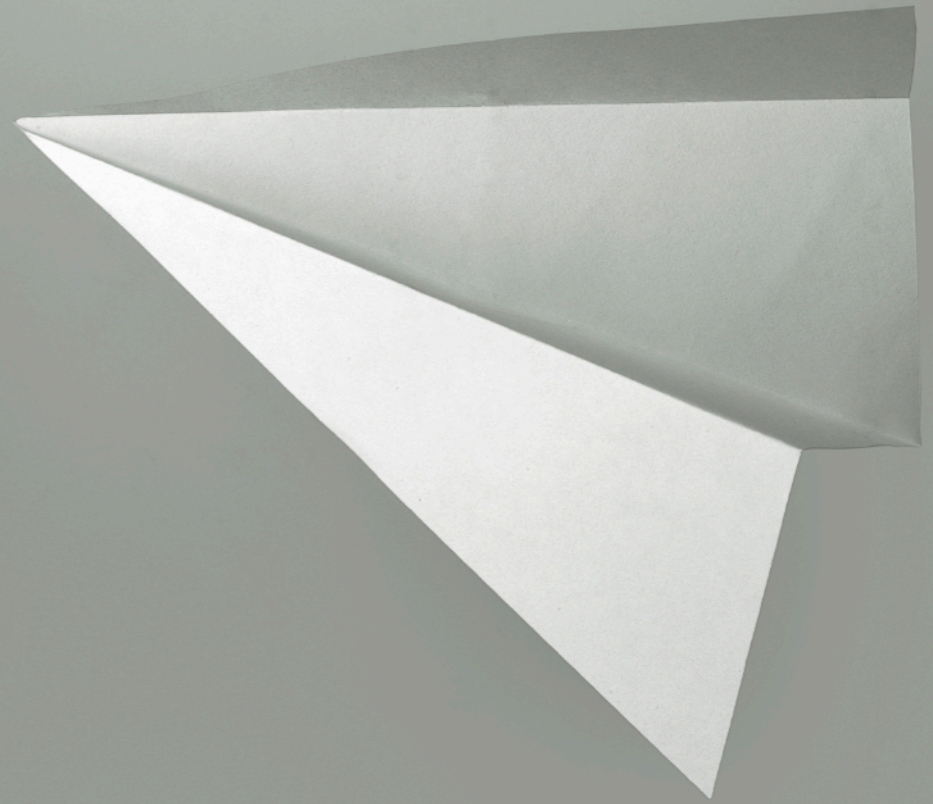


Domänenspezifische Sprachen

itemis

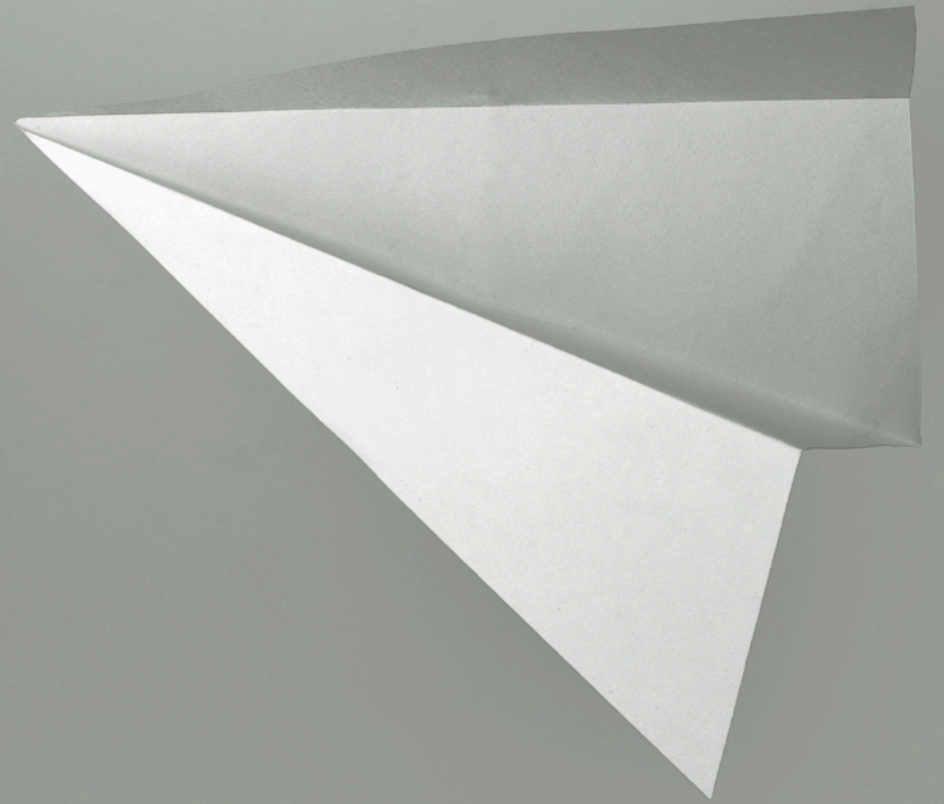


Holger Schill

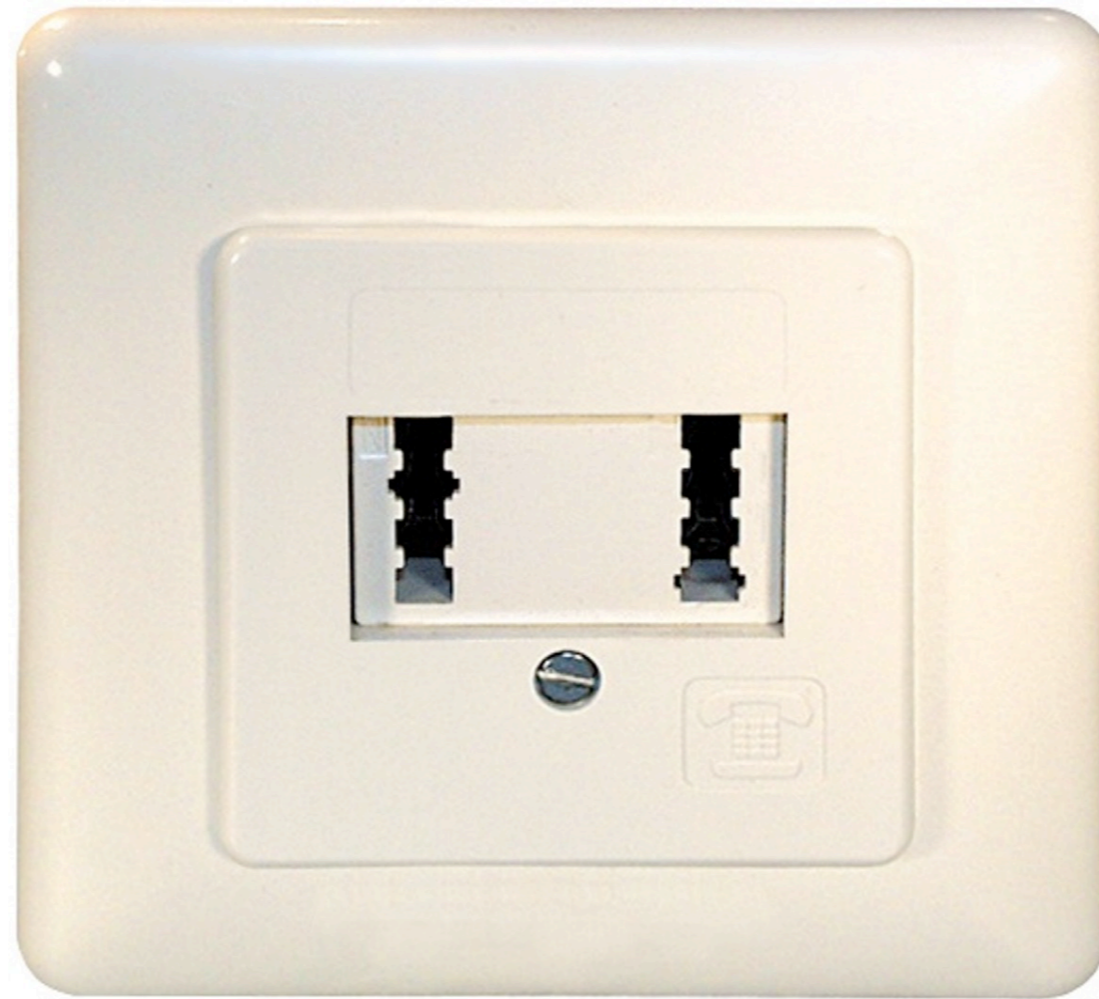


itemis

DSLs



DSL kommt aus der Wand



Programmiersprachen?

```
com.lauchenauer.lib.ui.V  
com.lauchenauer.lib.util  
public class AboutDialog extends  
protected CardLayout mLayout;  
protected JButton mCredits;  
protected JPanel mMainPanel;  
public AboutDialog(JFrame owner)  
super(owner);  
setModal(true);  
setUndecorated(true);  
initUI();  
protected void initUI() {  
setSize(440, 600);  
Container cont = getContentPane();  
JPanel p
```

Automatisierung durch Generatoren

```
com.lauchenauser.lib.ui.V  
com.lauchenauser.lib.util  
public class AboutDialog extends  
protected CardLayout mLayout;  
protected JButton mCredits;  
protected JPanel mMainPanel;  
public AboutDialog(Frame owner)  
setUndecorated(true);  
initUI();  
protected void initUI() {  
setSize(440, 600);  
Container cont = getContentPane();  
JPanel p
```

Später...!

```
com.lauchenauer.lib.util  
public class AboutDialog extends  
protected CardLayout mLayout;  
protected JButton mCredits;  
protected JPanel mMainPanel;  
public AboutDialog(JFrame owner)  
super(owner);  
setModal(true);  
setUndecorated(true);  
initUI();  
}
```

```
protected void initUI() {  
setSize(440, 600);  
Container cont = getContentPane();  
JPanel p
```

Natürliche Sprachen?



Domänenspezifische Sprachen





**...Berechnungsvorschrift 4711
mit Riester und**



**... Berechnungsvorschrift 4711
mit Riester und**

... Bla Bla ...





Spezifikation

Formale Beschreibung

Funktion: VKzahlbtgTF
Programmquelle: vmsctfa.c
Produkt-Typ: FONDS

PK-Typ:

verwendete Attribute:	Name	Verw.	Entität
	pk_typ_id	E	
	vtzb	E	
	vtrk_zb	E	
	vtrk_zustand	E	
	buzBFR	A	RUECKGABEWERT
	vkzb	A	RUECKGABEWERT

aufgerufene Funktionen: VKversartTF

Status: 13.0

Ansprechpartner PBT	Joachim Kaiser
Ansprechpartner IT	Frank Krein
zuletzt geändert von ... (PBT oder IT) am ...	Joachim Kaiser (PBT), 21.04.2009
zuletzt qualitätsgesichert von ... (PBT oder IT) am ...	

Die Funktion liefert den Zahlbeitrag auf Vertragskomponenten-Ebene abhängig von der Produktkomponente:

pk_typ_id	vkzb, buzBFR
KAPITAL_KONTO	Falls vtrk_zustand = ZUSTAND_BPFL vkzb = vtrk_zb sonst vkzb = 0
LV_TARIF	Falls stamm_ptr <> NULL falls zustand _{stamm} = ZUSTAND_BPFL vkzb = vtzb Wenn versart = VERSART_BUZB buzBFR = 0 sonst vkzb = 0
sonst	FEHLER(„PK_TYP_NICHT_IMPLEMENTIERT“)

Code

Formale Beschreibung

Funktion: VKzahlbtgTF

Programmquelle: vmsctfa.c

Produkt-Typ: FONDS

PK-Typ:

verwendete Attribute:	Name	Verw.	Entität
	pk_typ_id	E	
	vtzb	E	
	vtrk_zb	E	
	vtrk_zustand	E	
	buzBFR	A	RUECKGAB
	vkzb	A	RUECKGAB

aufgerufene Funktionen: VKversartTF

Status: 13.0

Ansprechpartner PBT	Joachim Kaiser
Ansprechpartner IT	Frank Krein
zuletzt geändert von ... (PBT oder IT) am ...	Joachim Kaiser (PBT), 21.0
zuletzt qualitätsgesichert von ... (PBT oder IT) am ...	

Die Funktion liefert den Zahlbeitrag auf Vertragskomponenten-Ebene a Produktkomponente:

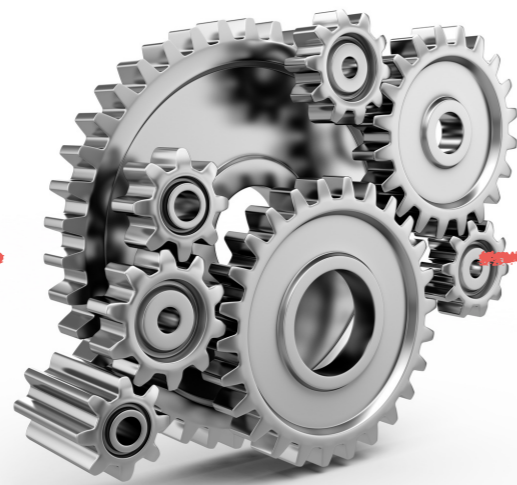
pk_typ_id	vkzb, buzBFR
KAPITAL_KONTO	Falls vtrk_zustand = ZUSTAND_BPFL vkzb = vtrk_zb sonst vkzb = 0
LV_TARIF	Falls stamm_ptr <> NULL falls zustand_{stamm} = ZUSTAND_BPFL vkzb = vtzb Wenn versart = VERSART_BUZB buzBFR = 0 sonst vkzb = 0
sonst	FEHLER(„PK_TYP_NICHT_IMPLEMENTIERT“)

```
/*
 * ===== Funktion Kopf Beginn =====
 *
 * Name           : VKzahlbtgTF()
 *
 * Beschreibung   : gibt den Zahlbeitrag auf Vertragskomponenten-Ebene zurueck
 *
 * Rueckgabewert :
 *
 * Einschränkungen : -
 *
 * ===== Funktion Kopf Ende =====
 */
double VKzahlbtgTF(TVK_EL *tvk_el_ptr, short *buzbBfr, TECHPTR *tech_ptr)
{
    double vkzb = 0.0;

    switch (tvk_el_ptr->pk_el_ptr->pk_typ_id)
    {
        case PK_TYP_ID_KAPITAL_KONTO:
            if(tvk_el_ptr->tvdvtrk_ptr->vtrk->vtrk_zustand == ZUSTAND_BPFL)
            {
                vkzb = tvk_el_ptr->tvdvtrk_ptr->vtrk->vtrk_btg.vtrk_zb;
            }
            break;
        case PK_TYP_ID_LV_TARIF:
            if(tvk_el_ptr->tvdvt_ptr->stamm_ptr != NULL)
                if(tvk_el_ptr->tvdvt_ptr->stamm_ptr->bs_ptr->zustand == ZUSTAND_BPFL)
                {
                    vkzb = tvk_el_ptr->tvdvt_ptr->vt_ptr->vtbtg.vtzb;
                    if (VKversartTF(tvk_el_ptr, tech_ptr) == VERSART_BUZB)
                    {
                        *buzbBfr = 0;
                    }
                }
            break;
        default:
            FEHLER(PK_TYP_NICHT_IMPLEMENTIERT);
            break;
    }
    return vkzb;
}
```

Spezifikation direkt in einem formalen Editor

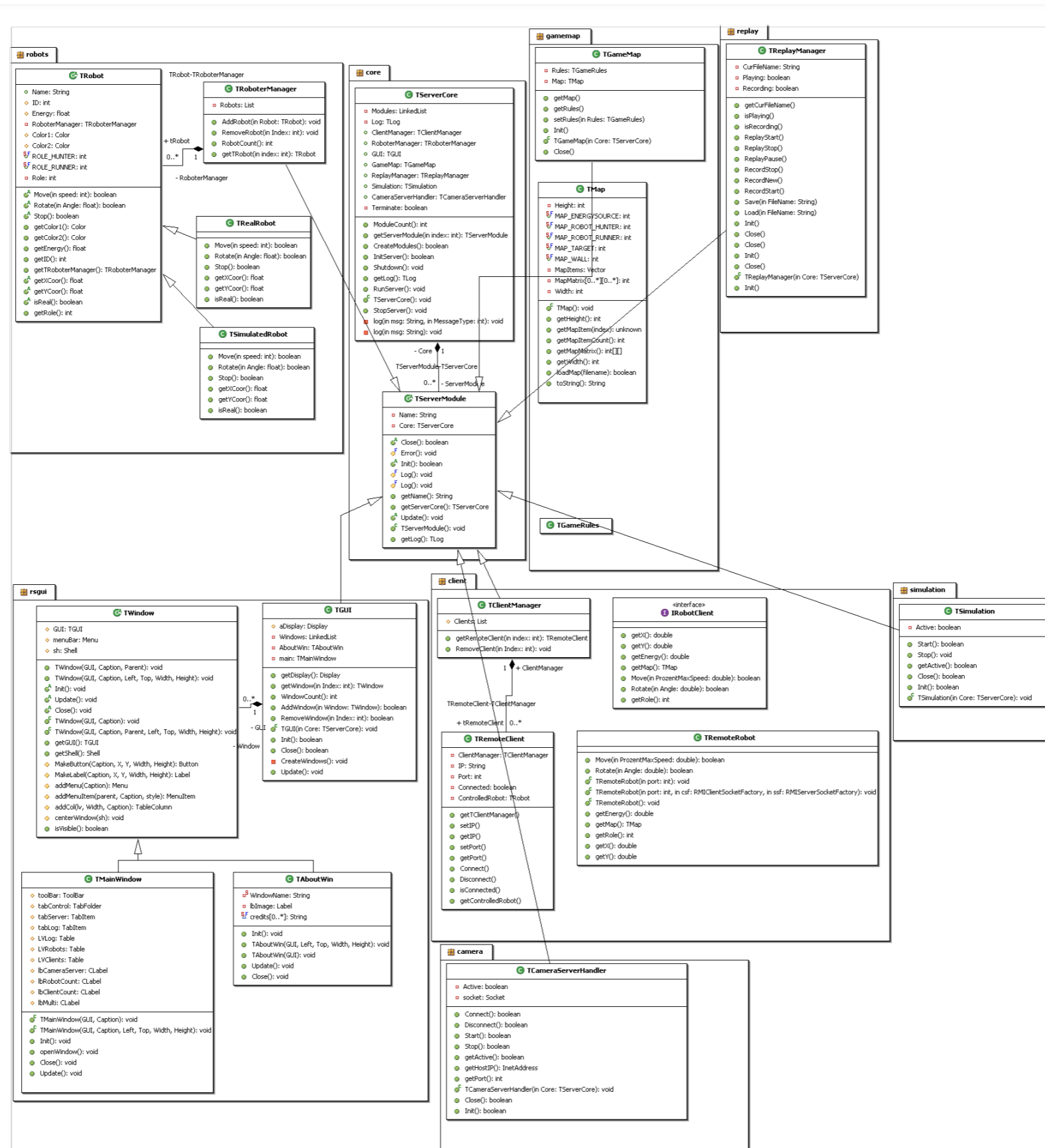
Generierung des Sourcecodes



```
template<class InputString, class OutputString>
bool unhexlify(const InputString& input, OutputString& output) {
    if (input.size() % 2 != 0) {
        return false;
    }
    output.resize(input.size() / 2);
    int j = 0;
    auto unhex = [](char c) -> int {
        return c >= '0' && c <= '9' ? c - '0' :
               c >= 'A' && c <= 'F' ? c - 'A' + 10 :
               c >= 'a' && c <= 'f' ? c - 'a' + 10 :
               -1;
    };
    for (size_t i = 0; i < input.size(); i += 2) {
        int highBits = unhex(input[i]);
        int lowBits = unhex(input[i + 1]);
        if (highBits < 0 || lowBits < 0) {
            return false;
        }
        output[j++] = (highBits << 4) + lowBits;
    }
    return true;
}
```



Model Driven Software Development



Spezifikation

Formale Beschreibung

Funktion: VKzahlbtgTF
Programmquelle: vmsctfa.c
Produkt-Typ: FONDS

PK-Typ:

Name	Verw.	Entität
verwendete Attribute:		
pk_typ_id	E	
vtzb	E	
vtrk_zb	E	
vtrk_zustand	E	
buzBFR	A	RUECKGABEWERT
vkzb	A	RUECKGABEWERT

aufgerufene Funktionen: VKversartTF

Status: 13.0

Ansprechpartner PBT	Joachim Kaiser
Ansprechpartner IT	Frank Krein
zuletzt geändert von ... (PBT oder IT) am ...	Joachim Kaiser (PBT), 21.04.2009
zuletzt qualitätsgesichert von ... (PBT oder IT) am ...	

Die Funktion liefert den Zahlbeitrag auf Vertragskomponenten-Ebene abhängig von der Produktkomponente:

pk_typ_id	vkzb, buzBFR
KAPITAL_KONTO	Falls vtrk_zustand = ZUSTAND_BPFL vkzb = vtrk_zb sonst vkzb = 0
LV_TARIF	Falls stamm_ptr <> NULL falls zustand _{stamm} = ZUSTAND_BPFL vkzb = vtzb Wenn versart = VERSART_BUZB buzBFR = 0 sonst vkzb = 0
sonst	FEHLER(„PK_TYP_NICHT_IMPLEMENTIERT“)



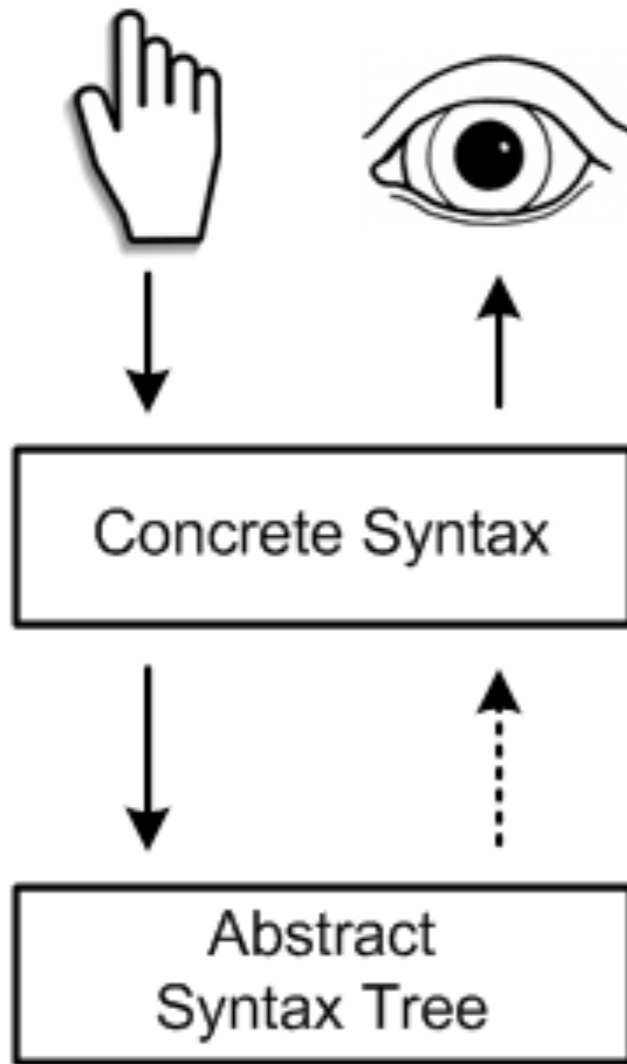
3.2 Meta Programming System

JetBRAINS



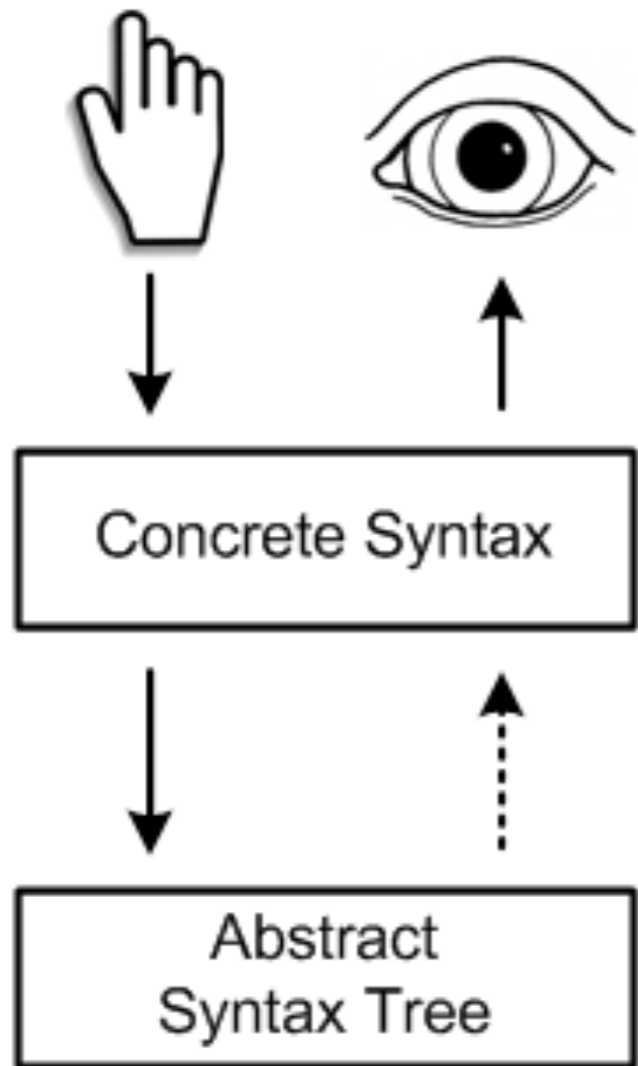
[Projectional Editing]

Parsing

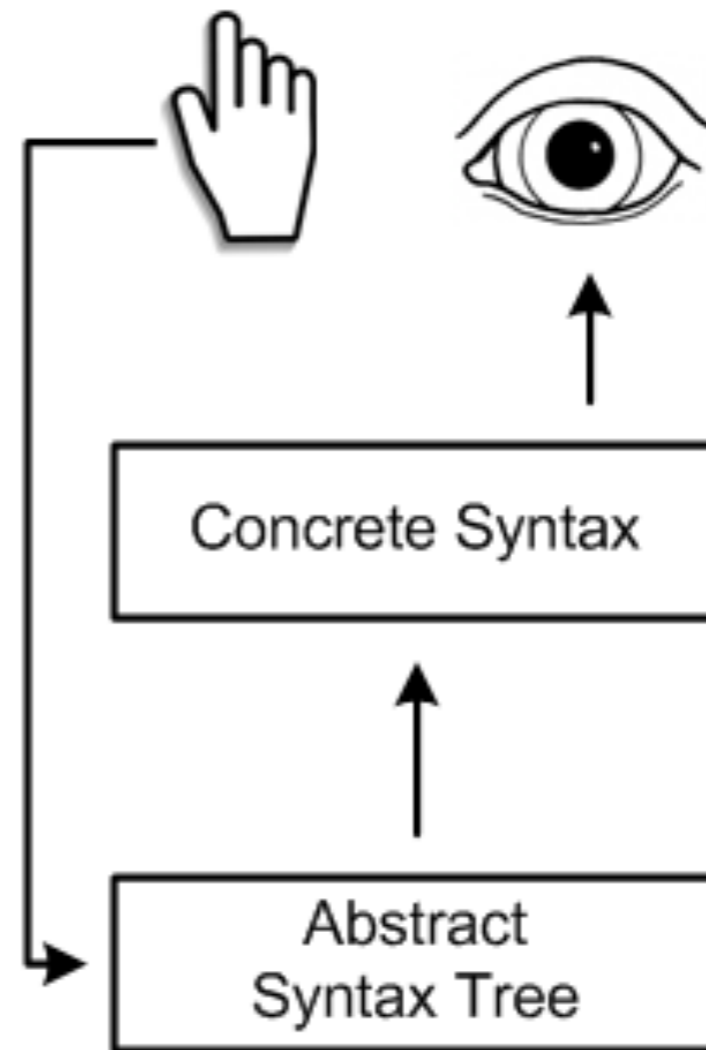


[Projectional Editing]

Parsing



Projectional Editing



[Projectional Editing]

Syntactic Flexibility

Regular Code/Text



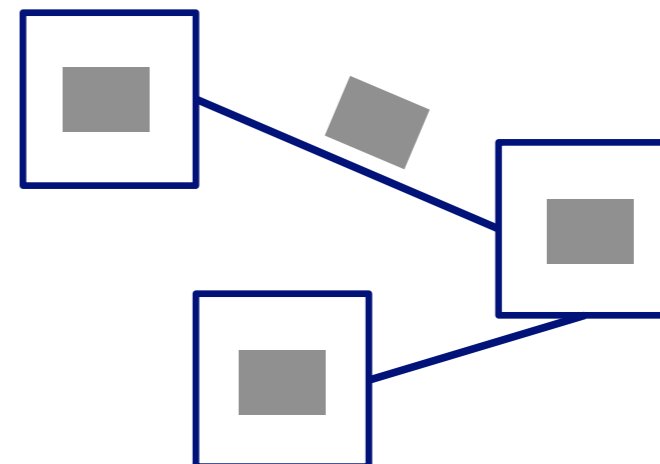
Mathematical



Tables

█	█	█
█		
█	█	
█		

Graphical



[Projectional Editing]

Syntactic Flexibility

Regular Code/Text

```
// [ A documentation comment with references ]  
// [ to @arg(data) and @arg(dataLen) ]  
void aSummingFunction(int8[] data, int8 dataLen) {  
    int16 sum;  
    for (int8 i = 0; i < dataLen; i++) {  
        sum += data[i];  
    } for  
} aSummingFunction (function)
```

Tables

```
int16 decide(int8 spd, int8 alt) {  
    return 

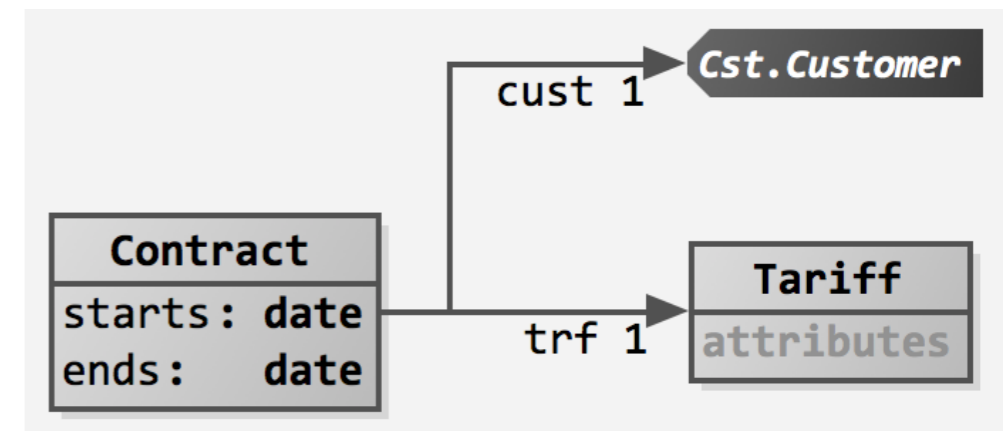
|           | spd > 0 | spd > 100 |
|-----------|---------|-----------|
| alt < 0   | 1       | 1         |
| alt == 0  | 10      | 20        |
| alt > 0   | 30      | 40        |
| alt > 100 | 50      | 60        |

 otherwise 0;  
} decide (function)
```

Mathematical

```
double midnight2(int32 a, int32 b, int32 c) {  
    return 
$$\frac{-b + \sqrt{b^2 - \sum_{i=1}^4 a * c}}{2 * a};$$
  
} midnight2 (function)
```

Graphical



Modell

Funktionsmodell VKzahlbtgTF

Formale Beschreibung

Funktion: VKzahlbtgTF
Enthaltende Quelldatei: vmsctfa.c
Produkt-Typ: Produkt-Typen auswählen
PK-Typ: PK-Typ auswählen
Status: Status auswählen

Parameter-Attribute

tvk_el_ptr: tvk_el <> E Beschreibung hinzufügen
buzbBfr: Ganzzahl A Beschreibung hinzufügen
tech_ptr techptr

Rückgabe-Typ

Kommazahl

Verwendete VADM-Attribute

pk_el.pk_typ_id E
tvdvt.stamm_ptr E
vt.bs.zustand E
vt.vtbtg.vtzb E
vtrk.vtrk_btg.vtrk_zb E
vtrk.vtrk_zustand E

aufgerufene Funktionen

VKversartTF (tvk_ptr: tvk_el <> E; tech_ptr: techptr E): Ganzzahl

Beschreibung

```
* ==== Funktion Kopf Beginn =====  
*  
* Name      : VKzahlbtgTF()  
*  
* Beschreibung : gibt den Zahlbeitrag auf Vertragskomponenten-Ebene zurueck  
*  
* Rueckgabewert :  
*  
* Einschränkungen : -  
*  
* ==== Funktion Kopf Ende =====
```

Hilfsvariablen

vkzb: Kommazahl Beschreibung hinzufügen

Verarbeitungen

:pk_typ_id	tvk_el_ptr.tvdvtrk_ptr.vtrk.vtrk_zustand Beschreibung hinzufügen	Bemerkung
PK_TYP_ID.KAPITAL_KONTO	Falls :vtrk_zustand = ZUSTAND.BPFL vkzb = :vtrk_zb Ende Falls :vtrk_zustand = ZUSTAND.BPFL	Beschreibung hinzufügen
PK_TYP_ID.LV_TARIF	Falls :stamm_ptr <> NULL Falls :zustand = ZUSTAND.BPFL vkzb = :vtzb Falls VKversartTF (tvk_el_ptr; tech_ptr) = VERSART.BUZZ buzbBfr = 0 Ende Falls VKversartTF(tvk_el_ptr; tech_ptr) = VERSART.BUZZ Ende Falls :zustand = ZUSTAND.BPFL Ende Falls :stamm_ptr <> NULL	Beschreibung hinzufügen
Andernfalls	Fehler (PK_TYP_NICHT_IMPLEMENTIERT)	Beschreibung hinzufügen

return vkzb



**Kompiliert aber
trotzdem falsch...**

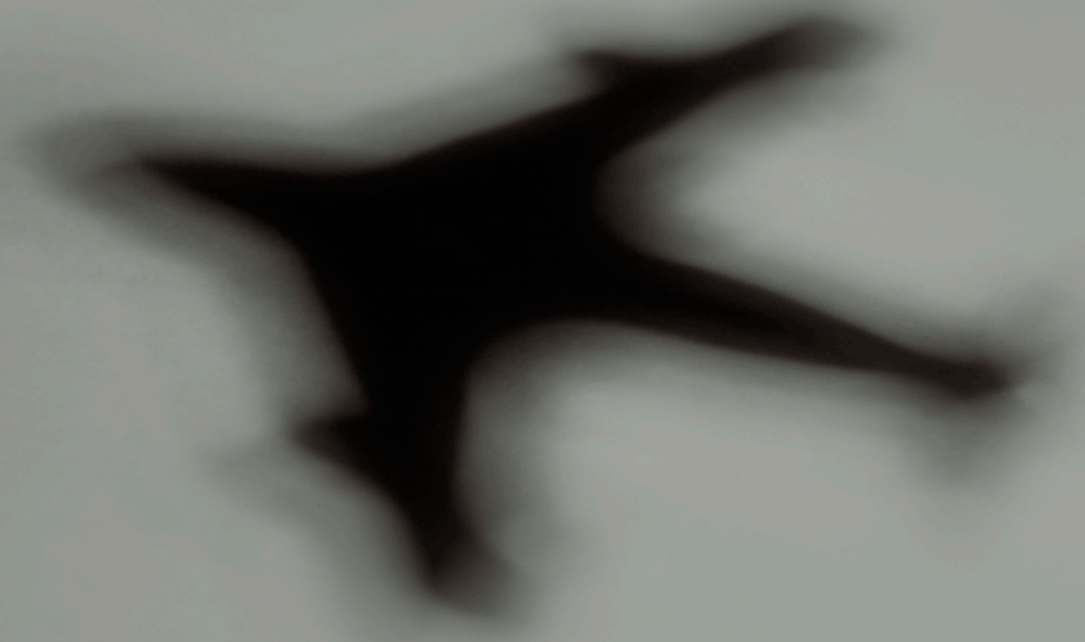
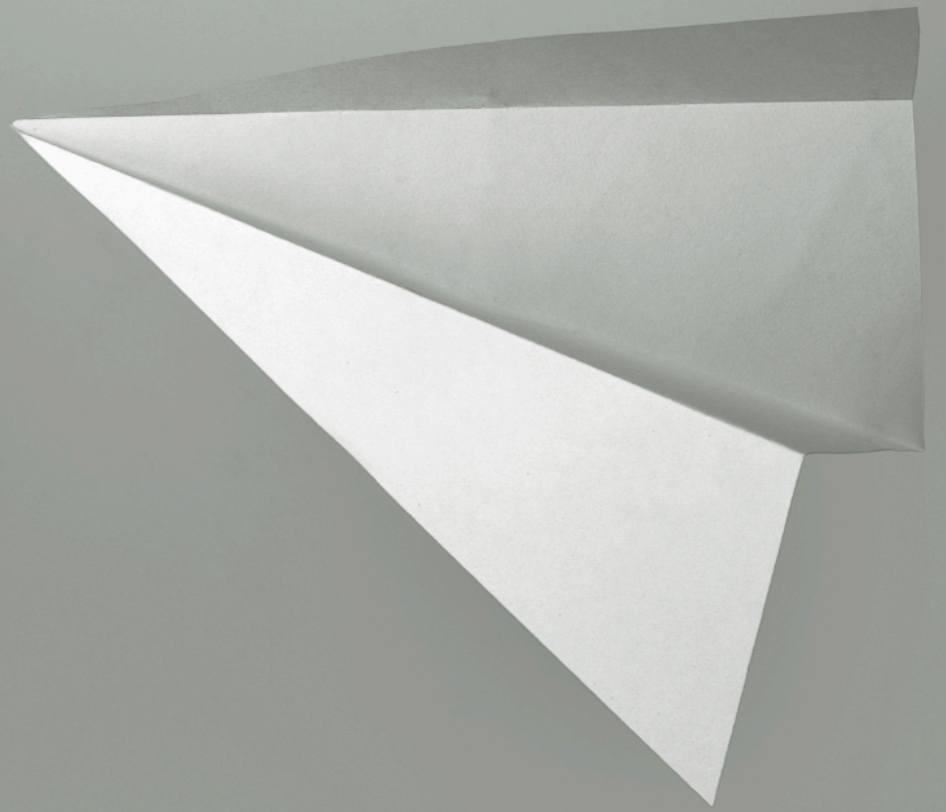


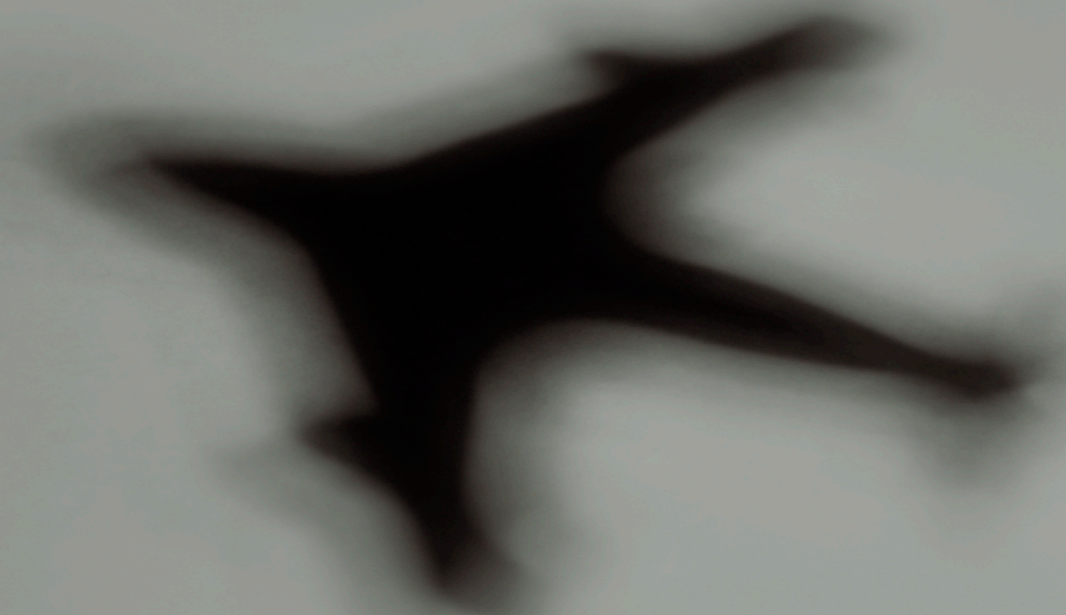
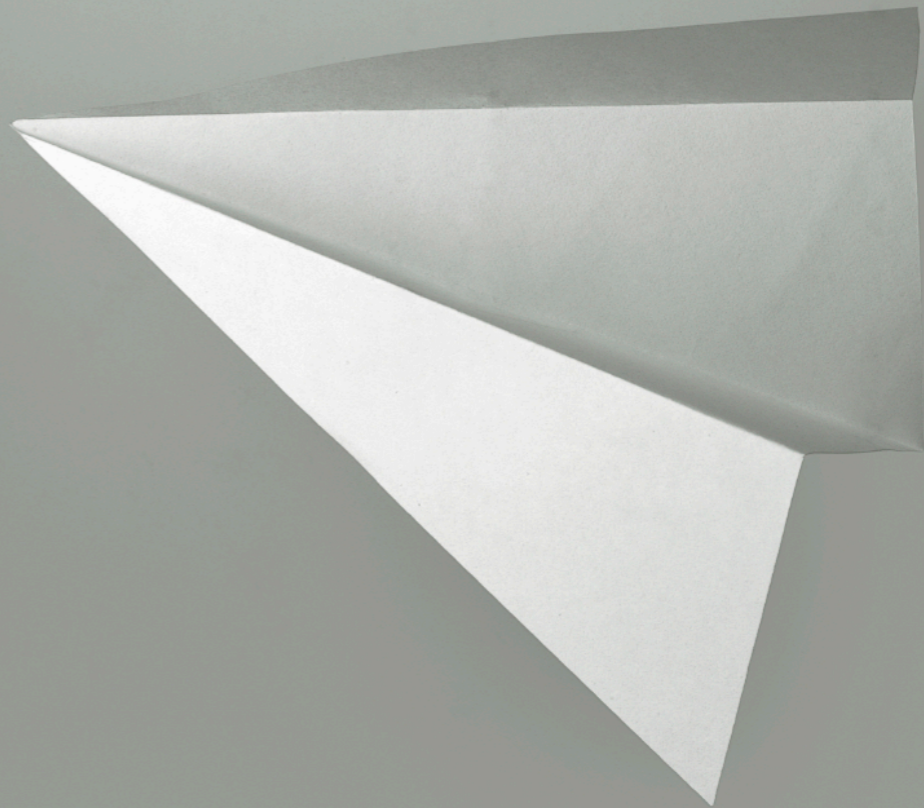
Fachliche Validierung im Model



**Aber wir sind doch
Entwickler...**

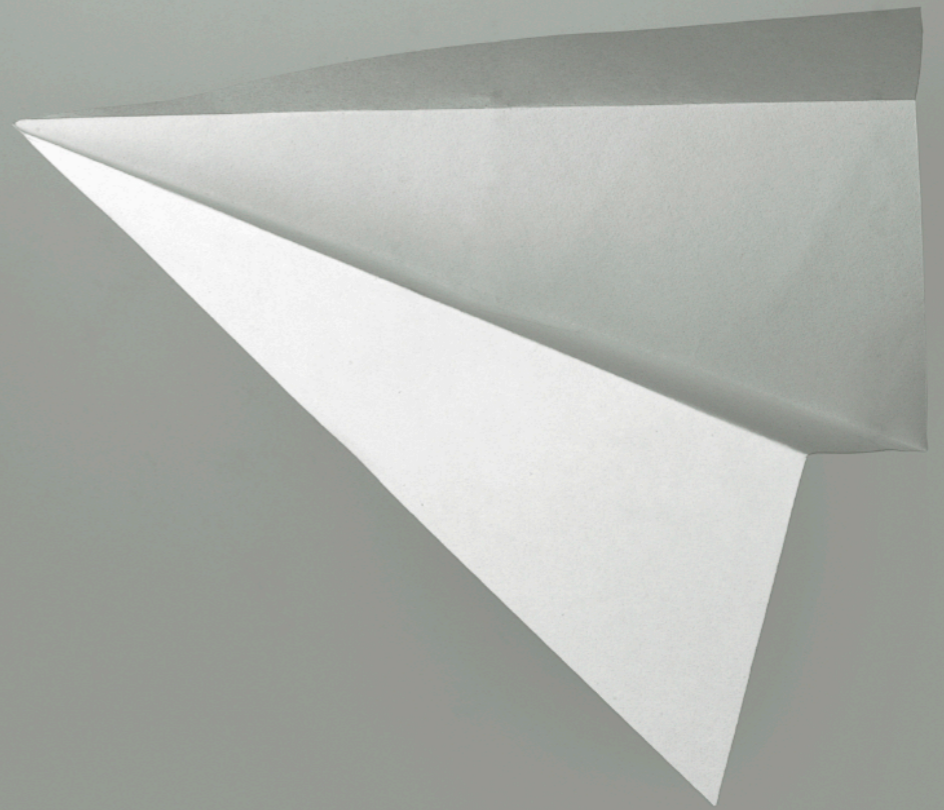






Komplexe Umsetzung im Code

Wenig Informationen



Komplexe Umsetzung im Code



```
public class Customer implements Serializable {  
  
    private String name;  
    private Address address;  
    private List<Contract> contracts;  
  
    public String getName() {  
        return name;  
    }  
    public void setName(String name) {  
        this.name = name;  
    }  
  
    public Address getAddress() {  
        return address;  
    }  
    public void setAddress(Address address) {  
        this.address = address;  
    }  
  
    public List<Contract> getContracts() {  
        return contracts;  
    }  
    public void setPhones(List<Contract> contracts) {  
        this.contracts = contracts;  
    }  
}
```

```
bean Customer {  
    name: String  
    address: Address  
    contracts: Contract*  
}
```


The Challenge

EJB

Navigationrules

FormBeans

Basis

Extras

Databinding

Properties

Apache Tobago

Projectsetup

Webservices

JSF



CONFIGURATION HELL.

A hand is shown from the top right, with the index finger pressing down on a circular green button. To the left of the green button is a white arrow pointing towards it. Below the green button is a circular red button. To the left of the red button is a white arrow pointing towards it. The word "Abstraction" is written in a large, bold, black font across the middle of the image, overlapping the hand and the buttons.

Abstraction



Entities

ScreenFlow

Widgets

Gui

Forms

Services

Datatypes

Exceptions

Mapping

Constraints





Xtext



Language Development Framework

DOMAIN-SPECIFIC LANGUAGES

The image shows two Eclipse IDE windows. The left window, titled `*FowlerDsl.xtext`, displays the DSL grammar. The right window, titled `*sample.fowlerdsl`, shows a graphical representation of the DSL code with a context menu open over the `startEvent => unknown` transition, offering options to change the target state to `emptyState`, `start`, or `stop`. The bottom of the right window shows the Problems view with two errors: `Couldn't resolve reference to State 'unknown'.` and `No transitions`.

```
grammar org.eclipse.xtext.example.FowlerDsl with org.eclipse.xtext.common.Terminals

generate fowlerdsl "http://example.xtext.org/FowlerDsl"

StateMachine :
{StateMachine}
'events'
(events+=Event)*
'end'
'commands'
(commands+=Command)*
'end'
(states+=State)*;

Event :
(resetting?='resetting')? name=ID code=ID;

Command :
name=ID code=ID;

State :
'state' name=ID
('actions' '{' (actions+=[Command])+ '}')?
(transitions+=Transition)*
'end';

Transition :
event=[Event] '=>' state=[State];
```

```
events
  startEvent START1
  stopEvent STOP2
end

commands
end

state start
  startEvent => unknown
end

state stop
  stopEvent =>
end

state emptyState
  No transitions
end
```

Problems 2 errors, 0 warnings, 0 others

Description	Resource
Errors (2 items)	
Couldn't resolve reference to State 'unknown'.	sample.fowlerdsl
No transitions	sample.fowlerdsl

PROGRAMMING LANGUAGES

The image shows a screenshot of an IDE with three main windows and a problems panel.

- app_message_pkg.sql**:

```
create or replace
package body app_message_pkg
is
-----
-- Create a new message
-----

procedure createAppMessage (
    p_mesg_type      in
    p_mesg_name      in
    p_mesg_text      in
)
is
begin
insert into app_messages
values (
exception
when others
then
    if sqlcode = -1
    then
        null;
    end if;
end createAppMessage;

end app_message_pkg;
/
show errors
```
- install.sql**:

```
start sequences
start app_messages
start app_message_pkg

exec app_message_pkg.createAppMessage('ERROR', 'NAME_NOT_UNIQUE', 'The name must be unique.');
```
- *app_messages.sql**:

```
drop table app_messages cascade constraints;

create table app_messages (
    mesg_id      number(10),
    mesg_type    varchar2(30) constraint mesg_mesg_type_nn not null,
    mesg_name    varchar2(30) constraint mesg_mesg_name_nn not null,
    mesg_text    varchar2(200) constraint mesg_mesg_text_nn not null
);

comment on table app_messages is 'Application messages';
comment on column app_messages.mesg_id is 'Primary Key';
comment on column app_messages.mesg_type is 'Type of message';
comment on column app_messages.mesg_name is 'Unique name of the message';
comment on column app_messages.mesg_text is 'Message text';

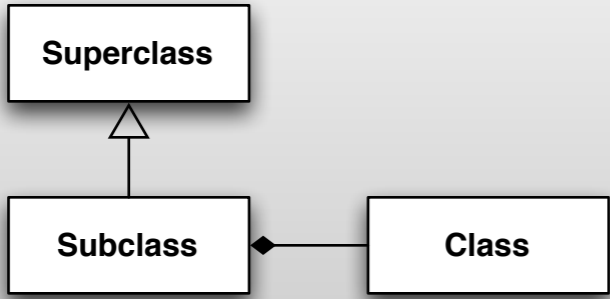
alter table app_messages
add constraint mesg_pk
primary key (mesg_id)
using index;

alter table app_messages
add constraint mesg_type_nn
unique (mesg_type)
using index;
```
- Problems Panel**:
 - 3 errors, 0 warnings, 0 others
 - Errors (3 items):
 - Guideline 25 violated: Always specify the target columns when executing an insert command.
 - Guideline 47 violated: Never handle unnamed exceptions using the when others then clause.
 - Guideline 54 violated: Always use a string variable to execute dynamic SQL.

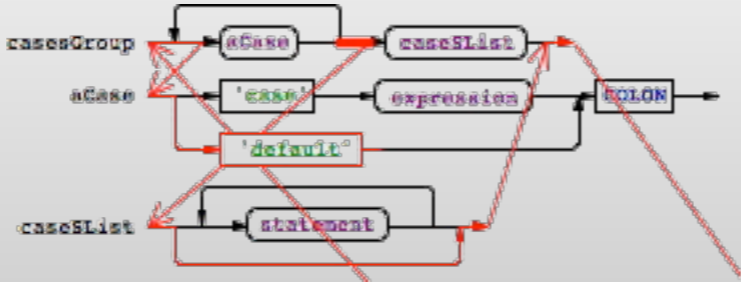
Reference Model



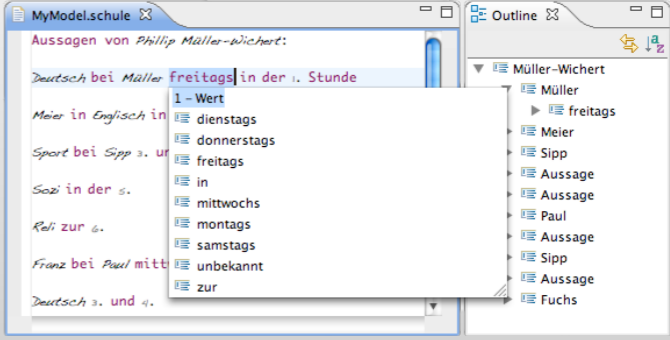
Xtext Runtime



Ecore model

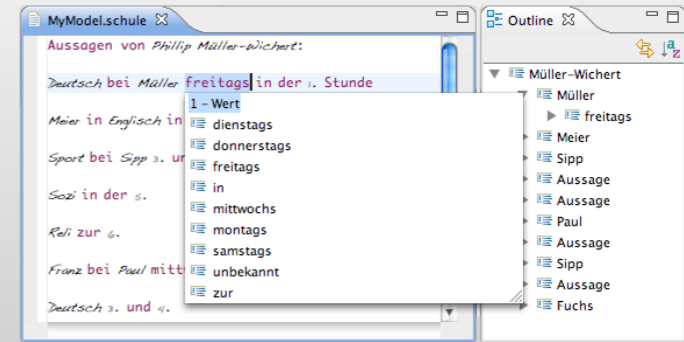
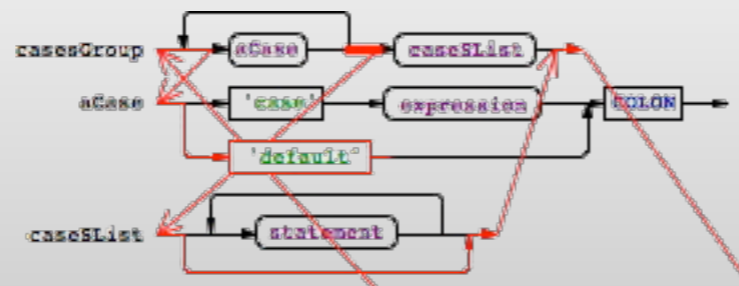
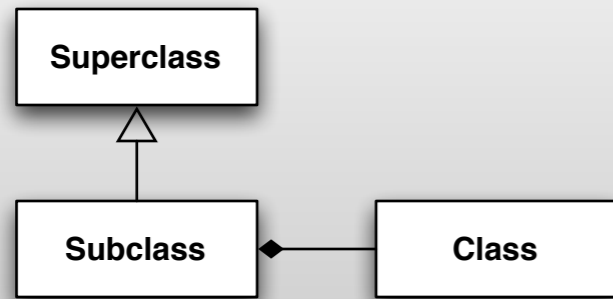


LL(*) Parser



Editor

Xtext Runtime



Ecore model

LL(*) Parser

Editor

<uses types>

<instanceof>

<instantiates>

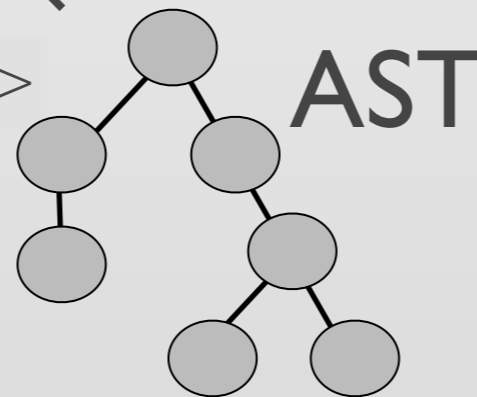
<parsed by>

<creates>

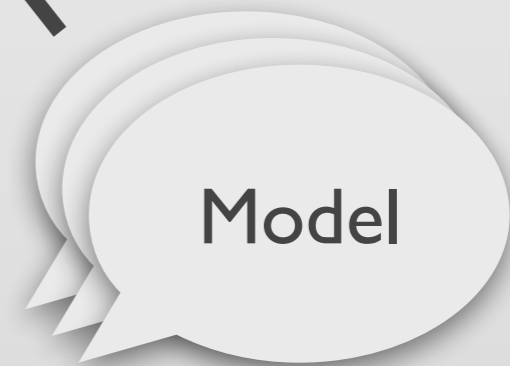
Interpreter

Code Generator

<processed by>



AST



```
package de.nordlbit.gamma;  
import de.nordlbit.gamma.po.model;
```

```
ManagedBean currentUser {  
    type Person;  
    scope session;  
}
```

```
ManagedBean mail {  
    type Mail;  
    scope request;  
}
```

```
ManagedBean folder {  
    type Folder;  
    scope request;  
}
```

```
ManagedBean adressBookPrivate {  
    type Adressbook;  
    scope session;  
}
```

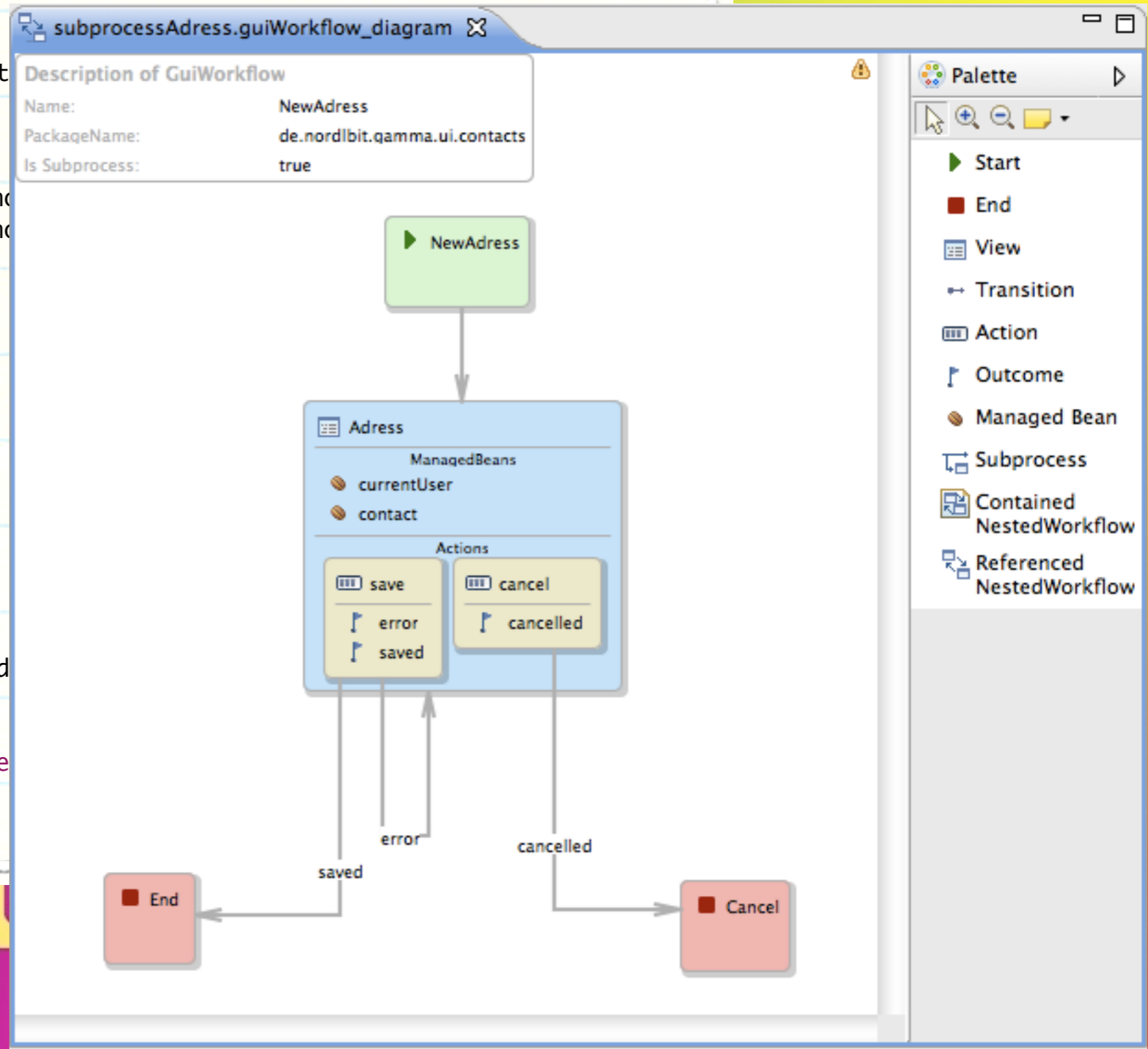
```

GuiWorkflowNode NewAdress {
  packageName de.nordlbit.gamma.ui.contacts
  subProcess;
  StartNode NewAdress;
  ViewNode Adress {
    ManagedBeanRefNode managedBean de.no
    ManagedBeanRefNode managedBean de.no
    ActionNode save {
      OutcomeNode error;
      OutcomeNode saved;
    }

    ActionNode cancel {
      OutcomeNode cancelled;
    }
  }

  EndNode Cancel;
  EndNode End;
  Transition source NewAdress -> target Ad
  Transition source Adress.save -> target
  Transition source Adress.save -> target
  Transition source Adress.cancel -> targe
}

```



ScreenFlow

Aktuelles Konto **batman@gotham-city.us**

Adressbuch

Privat **Geschäftlich**

Name	eMail	Ort
Holger	Holger.Schill@itemis.de	Kiel
Holger	Holger.Schill@itemis.de	Kiel

adressbook.gui_diagram

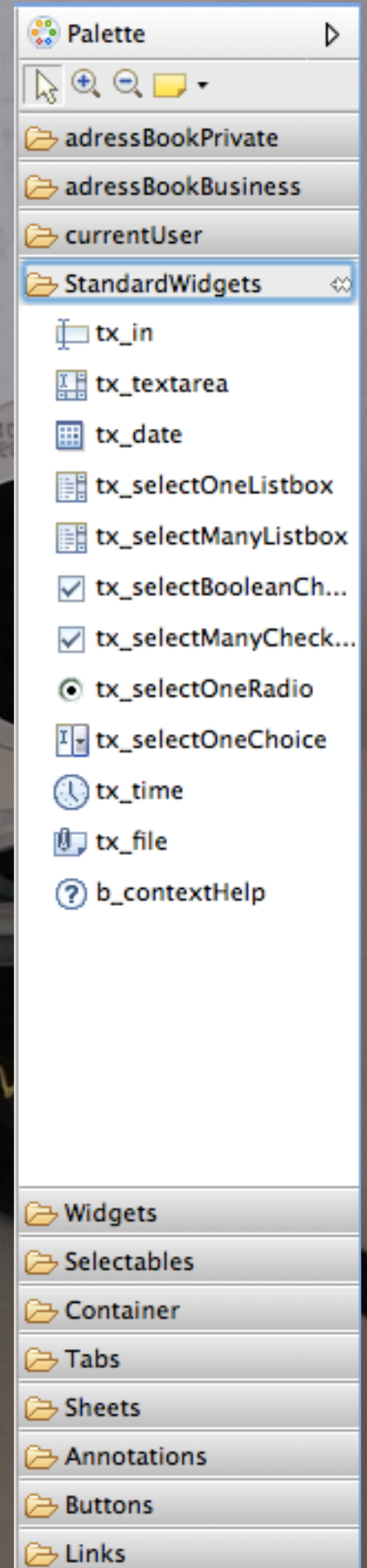


- Palette
- addressBookPrivate
- addressBookBusiness
- currentUser
- StandardWidgets
- Widgets
- Selectables
- Container
- Tabs
- Sheets
- Annotations
- Buttons
- Links

Dynamic Palette

```
package de.nordlbit.baseone;

WidgetTypeGroup StandardWidgets {
    public Widget tx_in xmlElementType "tx:in" {
        Attribute value "The current value of this component.";
        Attribute tip "Text value to display as tooltip.";
        ...
    }
    public Widget tx_date xmlElementType "tx:date" {
        Attribute value "The current value of this component.";
        ...
    }
}
```



Dynamic Palette

The image shows a GUI design tool interface. The main window displays a GUI diagram for 'adressbook.gui_diagram'. The diagram includes a text field with the placeholder 'Aktuelles Konto : \${currentUser.EMAIL}', a checkbox labeled 'Adressbuch', and two tabs labeled 'Privat' and 'Geschäftlich'. A palette on the right side of the tool is open, showing a list of widgets and containers. The 'contacts' widget is highlighted, and a tooltip 'Create a Sheet for contact:' is visible. Below the GUI diagram, a code editor shows the following code:

```
}  
PO Adress {  
  shortDescription "An adress is referenced  
  serialUID 1L;  
  Attribute FDTString type;  
  Attribute FDTString street;  
  Attribute FDTInteger zip;  
  Attribute FDTString city;  
  Attribute FDTString country;  
}
```

On the right side of the code editor, there is a snippet of code:

```
tc_label;  
tc_out;  
}
```

The palette on the right side of the tool is titled 'Palette' and contains the following items:

- adressBookPrivate
- adressBookBusiness
- adressbookName
- contacts
- owner
- currentUser
- StandardWidgets
- Widgets
- Selectables
- Container
- Tabs
- Sheets
- Annotations
- Buttons
- Links

The palette also shows a tooltip for the 'contacts' widget: 'Create a Sheet for contact:'.

Aber . . .



Migration from mainframe to Base/ONE

The screenshot shows the Eclipse IDE interface. The title bar reads "Java - migrator.web/src-model/migration/zoform.mfs - Eclipse Platform". The Package Explorer on the left shows the project structure:

- migrator.web
 - src-model
 - migration
 - zoform.mfs

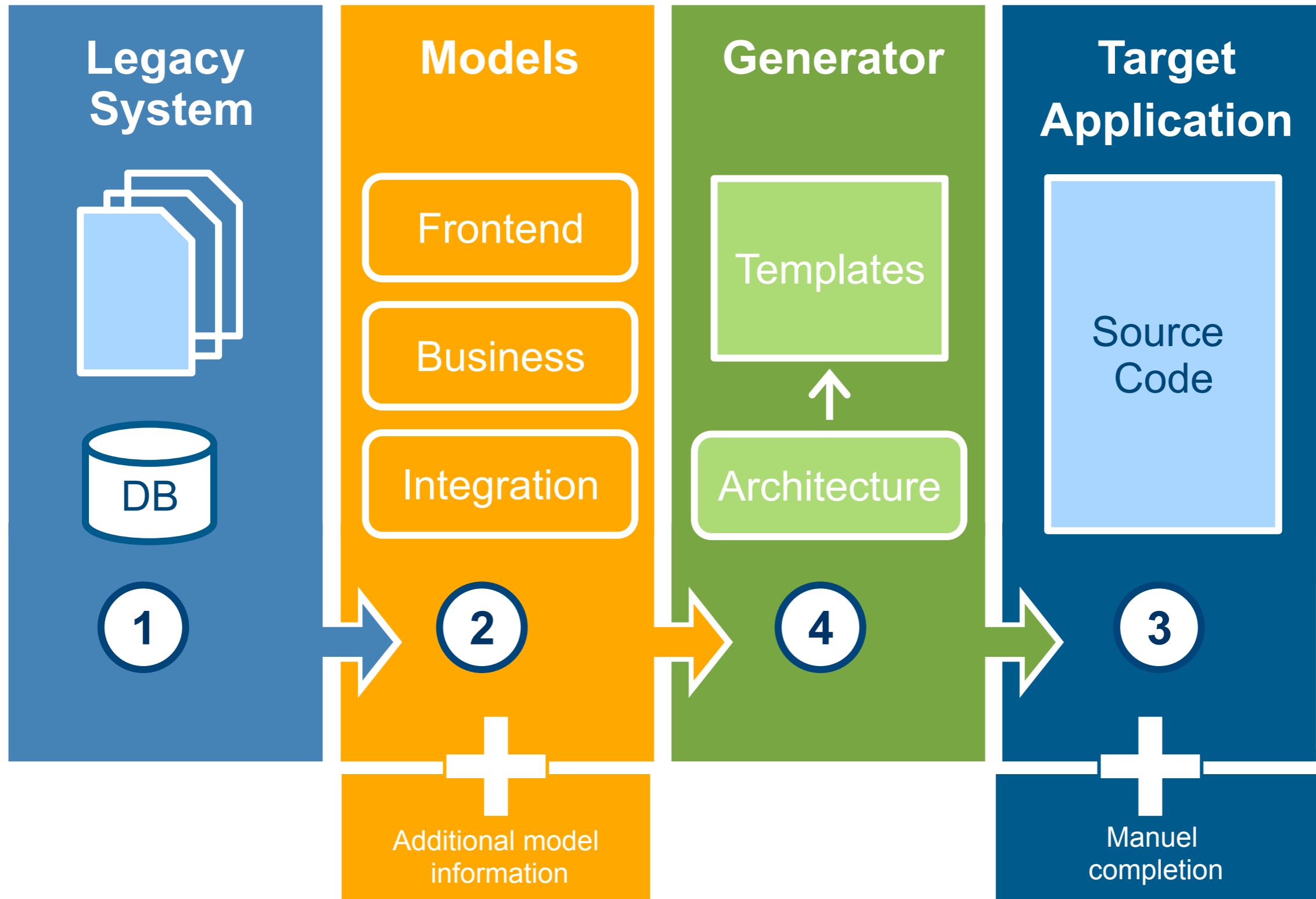
The main editor displays the content of 'zoform.mfs', which is a list of data records. Each record is a line of text with fields separated by commas. The records are grouped by a number (10, 11, 12, 13, 14) indicated by a dashed line and an asterisk. The records are as follows:

```
DFLD      '/' ,POS=(09,55,01),H
DFLD      'T-Taglich)' ,POS=(09,57,01),H
*
* -----10-----
F0101    DFLD      POS=(10,03,01),LTH=01,MHUL
DFLD      'Obligo' ,POS=(10,05,01)
F1001    DFLD      POS=(10,28,01),LTH=10,MHUL
F0106    DFLD      POS=(10,42,01),LTH=01,MHUL
*
* -----11-----
F0102    DFLD      POS=(11,03,01),LTH=01,MHUL
DFLD      'Konzern' ,POS=(11,05,01)
F0701    DFLD      POS=(11,31,01),LTH=7,MHUL
F0107    DFLD      POS=(11,42,01),LTH=01,MHUL
*
* -----12-----
F0103    DFLD      POS=(12,03,01),LTH=01,MHUL
DFLD      'Kreditnehmer' ,POS=(12,05,01)
F0111    DFLD      POS=(12,26,01),LTH=1,MHUL
F1003    DFLD      POS=(12,28,01),LTH=10,MHUL
F0108    DFLD      POS=(12,42,01),LTH=01,MHUL
*
* -----13-----
F0104    DFLD      POS=(13,03,01),LTH=01,MHUL
DFLD      'Kunde' ,POS=(13,05,01)
F0301    DFLD      POS=(13,17,01),LTH=3,MHUL
F1004    DFLD      POS=(13,28,01),LTH=10,MHUL
*
* -----14-----
F0105    DFLD      POS=(14,03,01),LTH=01,MHUL
```

Replatforming



Model-based migration (analysis & extraction)



Fragen?

