A UML Profile for Dynamic Execution Persistence with Monitoring Purposes

Eladio Domínguez Murillo, **Beatriz Pérez Valle**, María Antonia Zapata Abad

University of Zaragoza
(Spain)

University of La Rioja
(Spain)

University of Zaragoza
(Spain)

MiSE'13

May 19th, San Francisco, CA

Table of contents

1. Introduction and background
2. Development of the DFMS Framework
3. The SEP Profile
4. Application and contributions of our approach
5. Conclusions and future work

Table of contents

1. Introduction and background
2. Development of the DFMS Framework
3. The SEP Profile
4. Application and contributions of our approach
5. Conclusions and future work

1. Introduction and background

Context

- In order to **provide** increasingly **better services**, **best practice guides** and **standards** for information technology guide organizations to **follow a continuous process improvement**.

- **Organizations** must be **compliant** with **legislation**.

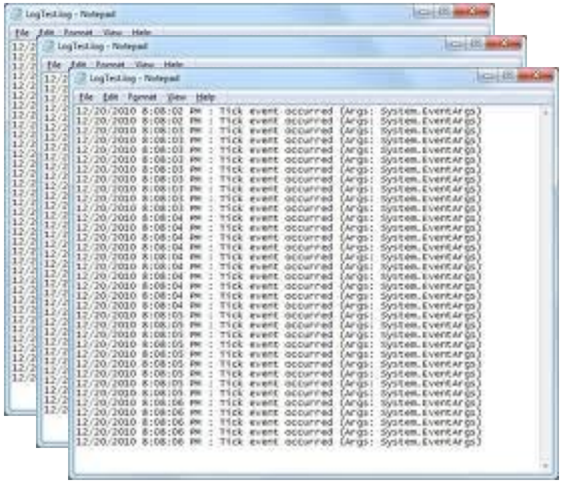
- **System monitoring** facilitates the trace of processes, and helps to achieve processes improvement.

1. Introduction and background

Problems

System monitoring

Logs



```
12/20/2010 8:08:02 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:02 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:03 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:03 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:03 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:03 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:03 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:03 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:03 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:03 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:03 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:04 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:04 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:04 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:04 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:04 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:04 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:04 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:04 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:04 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:04 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:05 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:05 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:05 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:05 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:05 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:05 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:05 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:05 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:05 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:06 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:06 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:06 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:06 PM : Tick event occurred (Args: System.EventArgs)
12/20/2010 8:08:06 PM : Tick event occurred (Args: System.EventArgs)
```

event

timestamp

performer

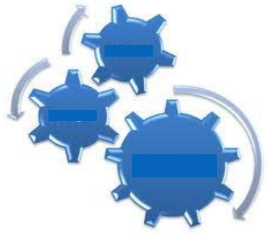
…

Complex Data mining processes

1. Introduction and background

Our approach

- Defining **more complete persistence structures** that facilitate the subsequent processing of the system trace.
- **Our particular context**:



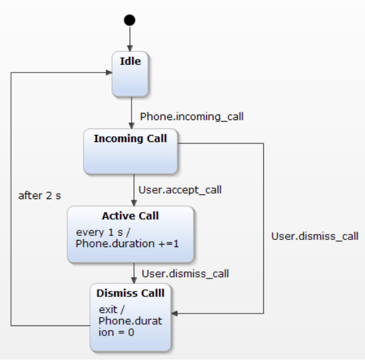
System behaviour

System trace data

- ✓ **Easing audit and process improvement**
- ✓ **Ensuring accuracy and consistency of data**
- ✓ **Guiding the transformation process of the storage structures**

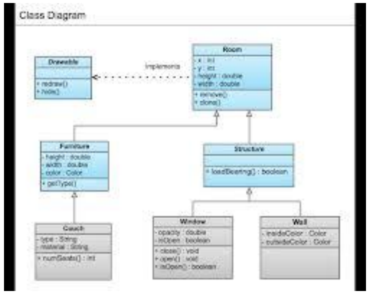
Represented

Represents

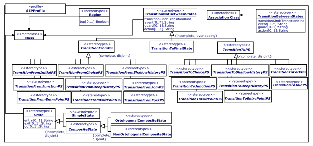


Statechart

Automatic generation



Class diagram



SEP Profile

Table of contents

1. Introduction and background
2. Development of the DFMS Framework
3. The SEP Profile
4. Application and Contributions of our Approach
5. Conclusions and future work

2. Development of the DFMS Framework

Characteristics

Decision Facts Management System (DFMS)

- Verifies clinical guidelines **against quality properties.**

- **Develops** guideline-based decision support systems **(GBDSSs).**

2. Development of the DFMS Framework

GBDSS for a guideline

GBDSS for a guideline

➢ To guide the physicians during the application of a guideline in a very specific way in order to help in their decision making.

➢ **Traceability**: To automatically record the history of the application **of the guideline** to patients.

Can be used:

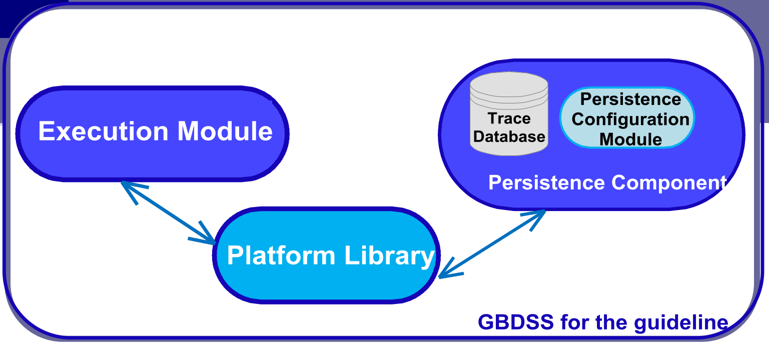
- as **key evidence** in a medical **negligence case.**

- as **a reference** in **future encounters** with the patient.

- as a **resource** for **ongoing changes** in the definition of guidelines over time.

2. Development of the DFMS Framework

GBDSS for a guideline



- **Execution module**: corresponds to the Java implementation of the statechart.

- **Persistence component**: constitutes the main module which guarantees the persistence of the guideline application.
  - **Persistence configuration component**: defined to achieve the persistent task.
  - **Trace database**: physically stores the data generated during the guideline application and whose instances come from the running of the execution module.

- **Platform library**: provides standard services related to the implementation of the presentation and the data layers, serving as “glue” between the execution module and the persistence component.

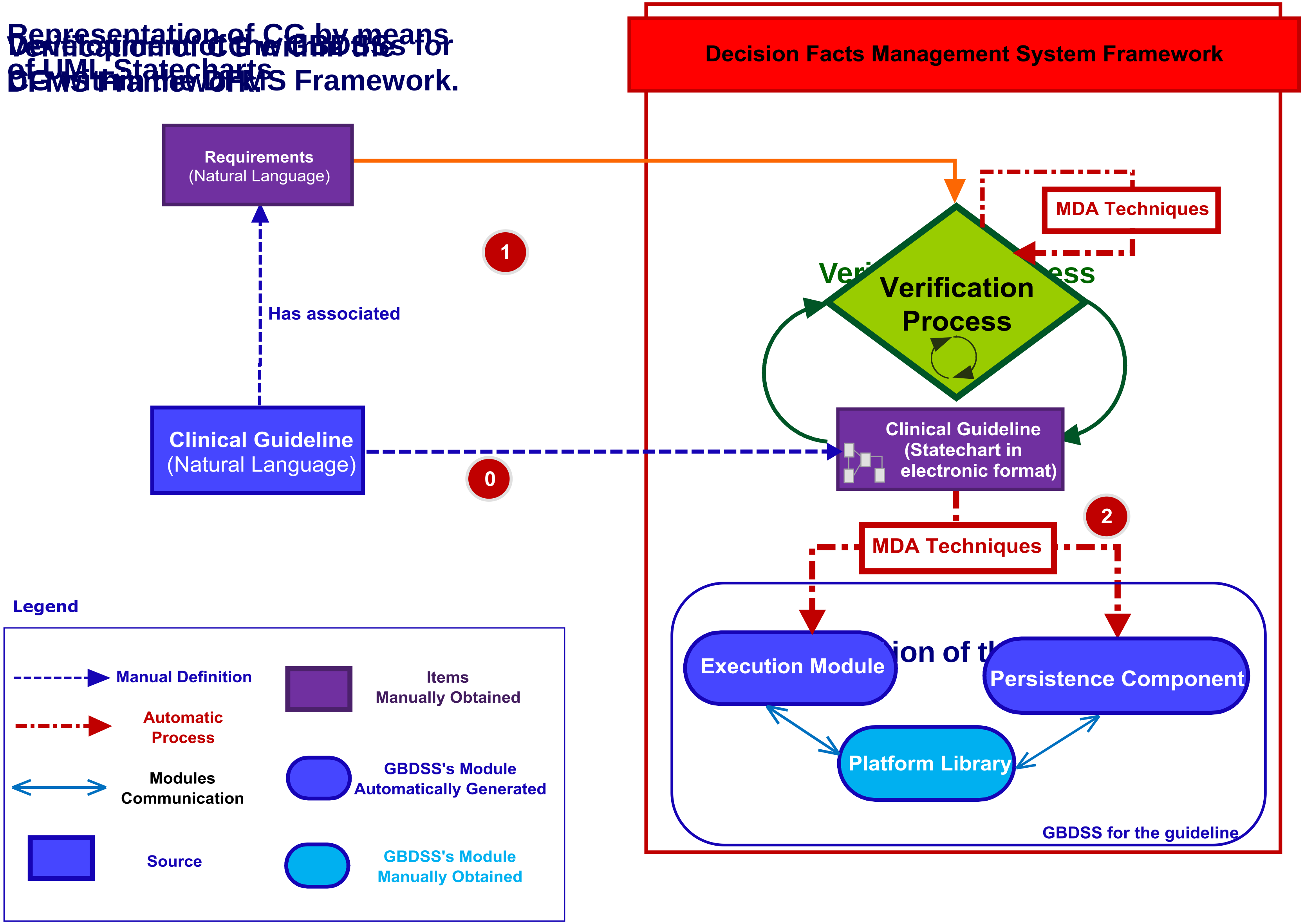
2. Development of the DFMS Framework

Decision Facts Management System

2 Representation of CG by means of UML Statecharts

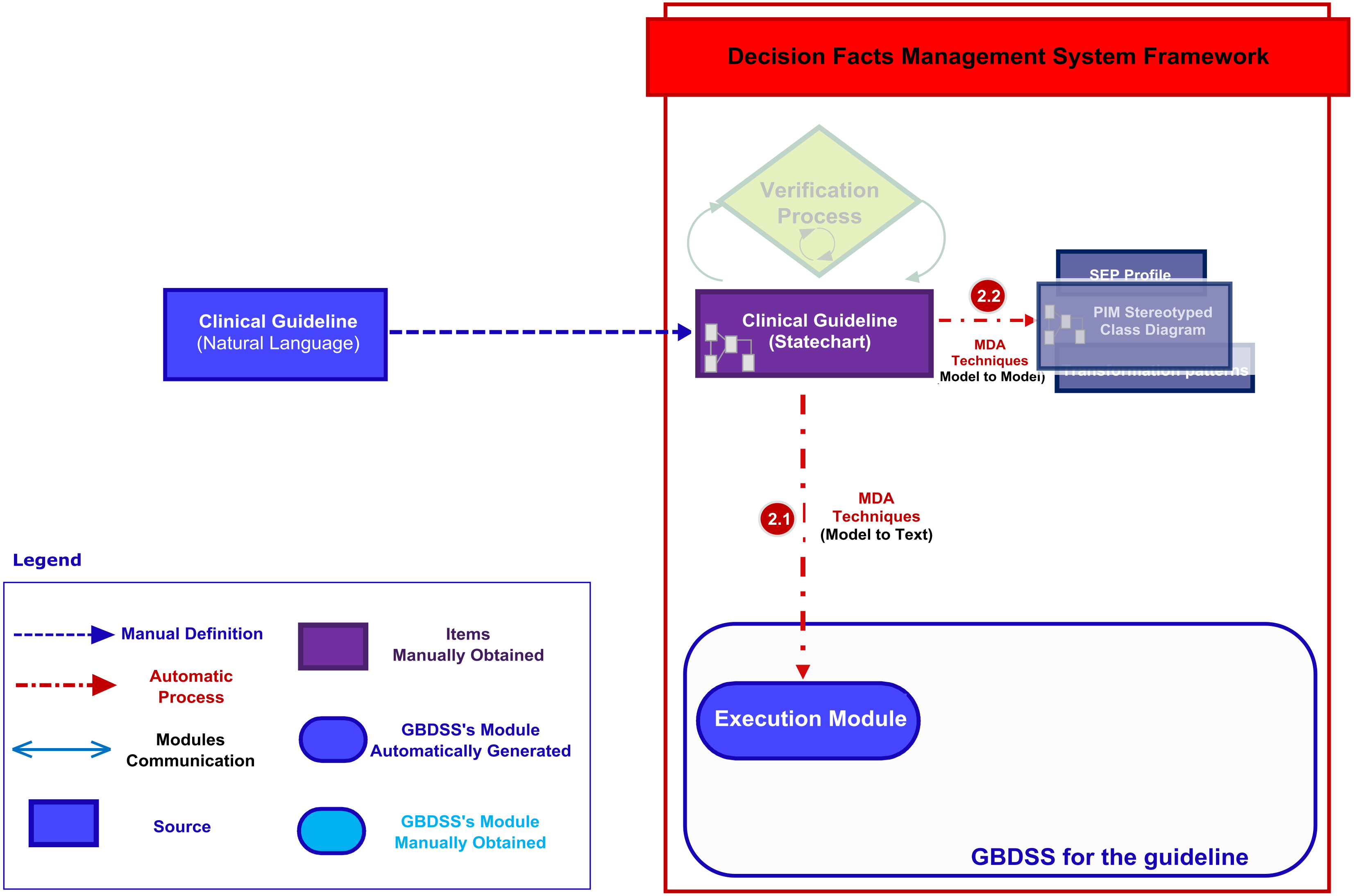
Verification of CG within the DFMS Framework.

Development of CG with GBDSS for DFMS Framework.



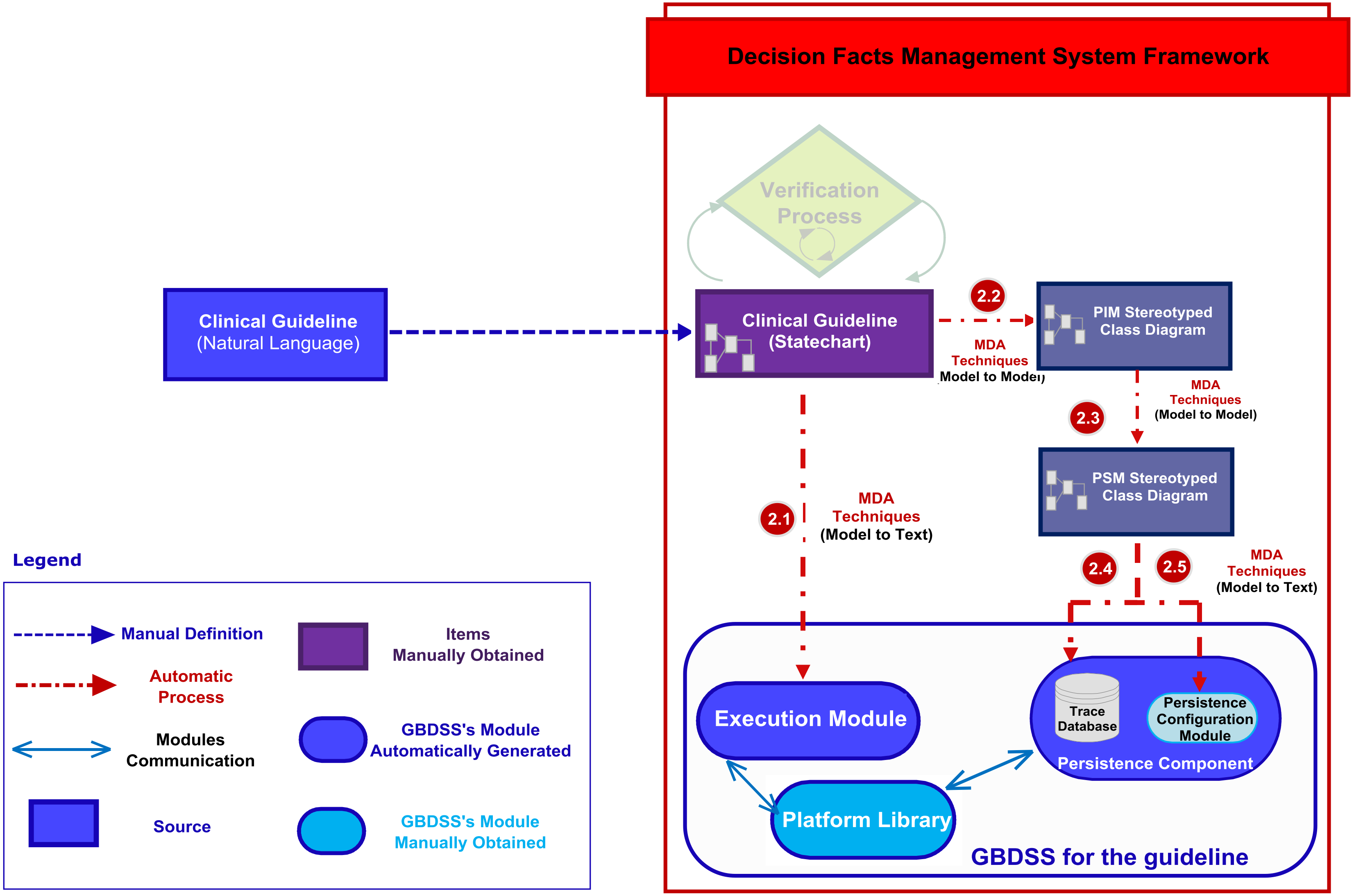
2. Development of the DFMS Framework

Decision Facts Management System



2. Development of the DFMS Framework

Decision Facts Management System



Decision Facts Management System Framework

Verification Process

Clinical Guideline (Natural Language)

Clinical Guideline (Statechart)

2.2 MDA Techniques Model to Model)

PIM Stereotyped Class Diagram

2.3 MDA Techniques (Model to Model)

PSM Stereotyped Class Diagram

2.1 MDA Techniques (Model to Text)

2.4 2.5 MDA Techniques (Model to Text)

Execution Module

Trace Database

Persistence Configuration Module

Persistence Component

Platform Library

GBDSS for the guideline

Legend

Manual Definition

Automatic Process

Modules Communication

Source

Items Manually Obtained

GBDSS's Module Automatically Generated

GBDSS's Module Manually Obtained

Table of contents

1. Introduction and background
2. Development of the DFMS Framework
3. The SEP Profile
4. Application and Contributions of our Approach
5. Conclusions and future work

3. SEP Profile

Definition

Since UML 2.x does not provide a definition strategy for profiles, but simply presents the *UML Extensibility Mechanism* package with its components…

- **we have based on a mixture of:**

  (1) The proposal given by Conallen [**Cona00**], which defines UML profiles starting with a brief description and then specifying its stereotypes, tagged values and constraints, and

  (2) the proposal given by Gogolla and Henderson [**GoHe02**], which describes stereotypes using a complete template definition.

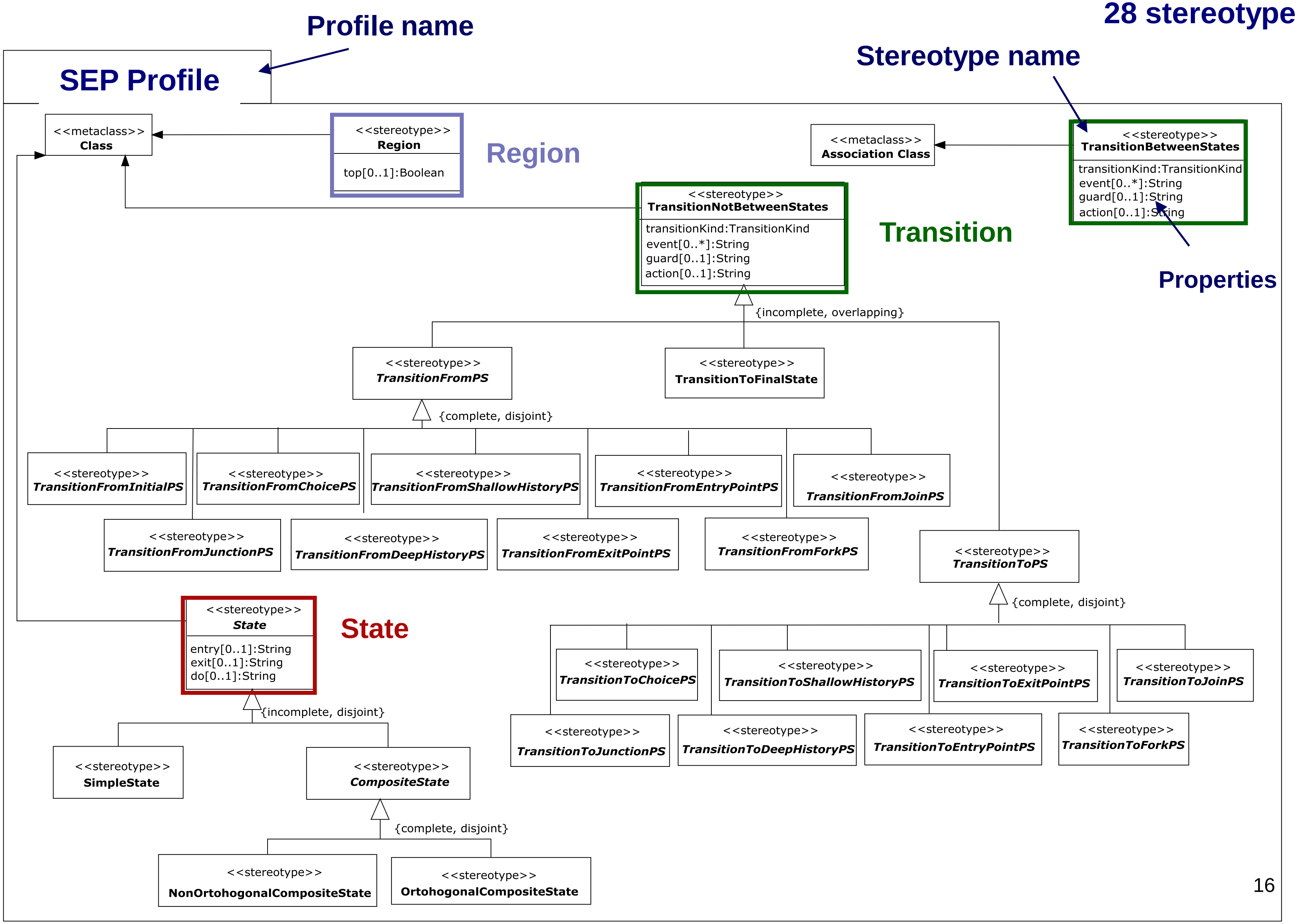
- **Additionally:**

  - We have extended [**GoHe02**] so that our profile conforms to the **UML 2.4 metamodel**.
  - the correct definition and use of our extension mechanism is assured by the **definition of the stereotypes' constraints** in both **natural language** and **OCL**.

3. SEP Profile

Definition. Stereotypes

28 stereotypes



Profile name
Stereotype name
SEP Profile
<<metaclass>> Class
<<stereotype>> Region
top[0..1]:Boolean
Region
<<metaclass>> Association Class
<<stereotype>> TransitionBetweenStates
transitionKind:TransitionKind
event[0..*]:String
guard[0..1]:String
action[0..1]:String
<<stereotype>> TransitionNotBetweenStates
transitionKind:TransitionKind
event[0..*]:String
guard[0..1]:String
action[0..1]:String
Transition
Properties
{incomplete, overlapping}
<<stereotype>> TransitionFromPS
<<stereotype>> TransitionToFinalState
{complete, disjoint}
<<stereotype>> TransitionFromInitialPS
<<stereotype>> TransitionFromChoicePS
<<stereotype>> TransitionFromShallowHistoryPS
<<stereotype>> TransitionFromEntryPointPS
<<stereotype>> TransitionFromJoinPS
<<stereotype>> TransitionFromJunctionPS
<<stereotype>> TransitionFromDeepHistoryPS
<<stereotype>> TransitionFromExitPointPS
<<stereotype>> TransitionFromForkPS
<<stereotype>> TransitionToPS
{complete, disjoint}
<<stereotype>> TransitionToChoicePS
<<stereotype>> TransitionToShallowHistoryPS
<<stereotype>> TransitionToExitPointPS
<<stereotype>> TransitionToJoinPS
<<stereotype>> TransitionToJunctionPS
<<stereotype>> TransitionToDeepHistoryPS
<<stereotype>> TransitionToEntryPointPS
<<stereotype>> TransitionToForkPS
<<stereotype>> State
entry[0..1]:String
exit[0..1]:String
do[0..1]:String
State
{incomplete, disjoint}
<<stereotype>> SimpleState
<<stereotype>> CompositeState
{complete, disjoint}
<<stereotype>> NonOrtohogonalCompositeState
<<stereotype>> OrtohogonalCompositeState

3. SEP Profile

Definition. Stereotypes

Name

Base Class

Description

Tagged Values
or
Attributes

Constraints

Name	*State*
Base class	
InfrastructureLibrary::Core::Constructs::Class	
Description	
This stereotype is defined as an abstract stereotype and it is the base for all the concrete stereotypes representing states in the statechart. It has two direct derived stereotypes: SimpleState and CompositeState.	
Attributes	
The State stereotype has three properties: (1) *entry* [0..1]: String, which refers to the optional behavior that is executed whenever the state, from which the stereotyped class comes, is entered (2) *exit* [0..1]:String, which specifies the optional behavior that is executed whenever the state is exited, (3) *do* [0..1]:String, that refers to the optional behavior that is executed while being in the state.	
Constraints	
Context InfrastructureLibrary::Core::Constructs::Class **inv**: self. isStereotyped ("State") **implies** **let** generalizationClass: self.generalization.general **and** **let** associatedClasses: self.ownedAttribute.association.memberEnd.class → asSet() → excluding(self) **in** generalizationClass → forAll(e| e.oclIsKindOf(Class) and e.isStereotyped("Region")) generalizationClass → size()=1 associatedClasses → forAll(c: Class | (c.isStereoKinded("TransitionNotBetweenStates") or c.isStereotyped ("TransitionBetweenStates") or c.isStereotyped ("Region")) and not (c.isStereoKinded("State"))	

3. SEP Profile

Definition. Stereotype

Wellformedness rules

Name

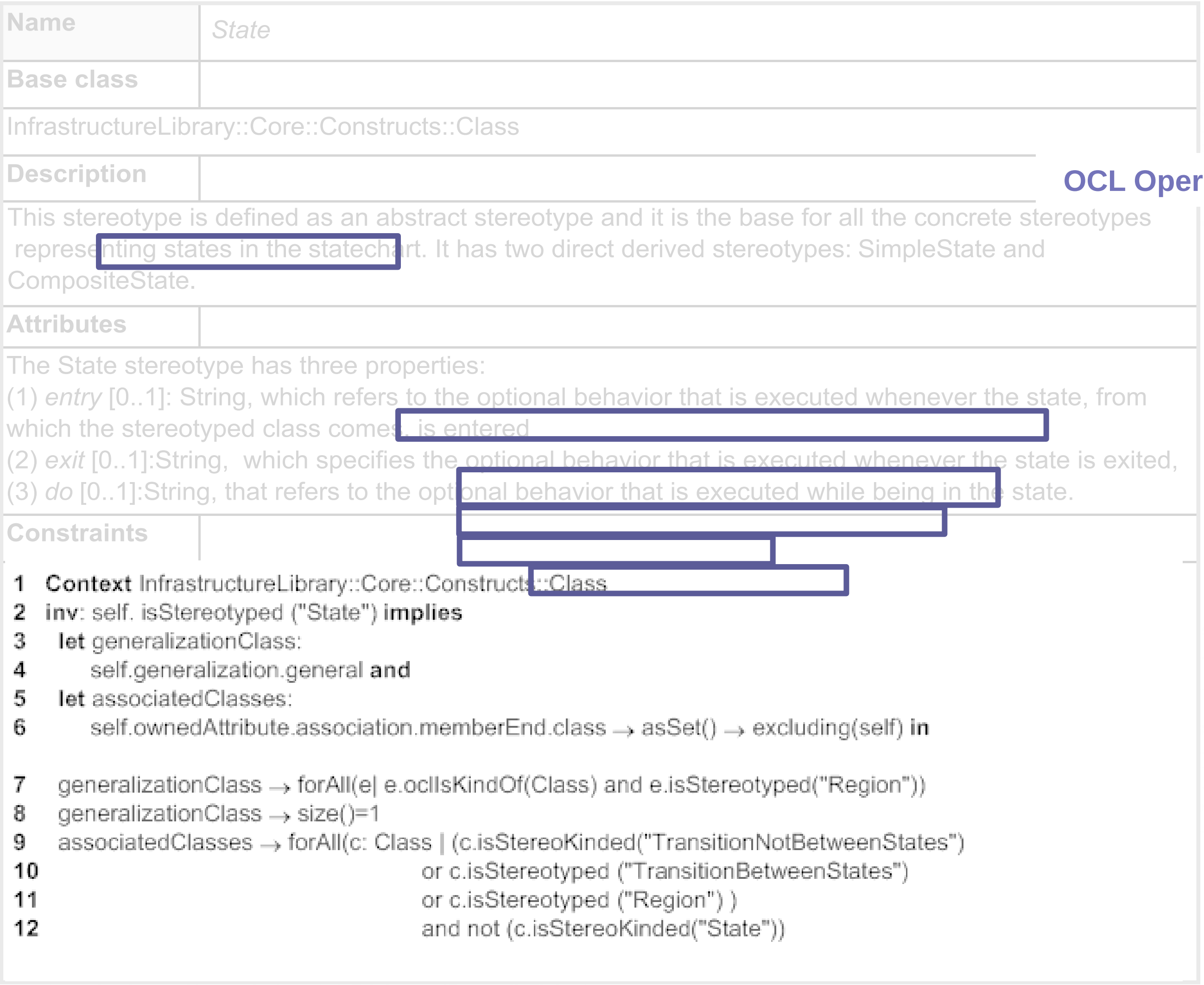
Base Class

Description

Tagged Values or Attributes

Constraints

OCL Operators



Name	*State*
Base class	
InfrastructureLibrary::Core::Constructs::Class	
Description	
This stereotype is defined as an abstract stereotype and it is the base for all the concrete stereotypes representing states in the statechart. It has two direct derived stereotypes: SimpleState and CompositeState.	
Attributes	
The State stereotype has three properties: (1) *entry* [0..1]: String, which refers to the optional behavior that is executed whenever the state, from which the stereotyped class comes, is entered (2) *exit* [0..1]:String, which specifies the optional behavior that is executed whenever the state is exited, (3) *do* [0..1]:String, that refers to the optional behavior that is executed while being in the state.	
Constraints	

```
Context InfrastructureLibrary::Core::Constructs::Class
inv: self. isStereotyped ("State") implies
  let generalizationClass:
      self.generalization.general and
  let associatedClasses:
      self.ownedAttribute.association.memberEnd.class → asSet() → excluding(self) in

  generalizationClass → forAll(e| e.oclIsKindOf(Class) and e.isStereotyped("Region"))
  generalizationClass → size()=1
  associatedClasses → forAll(c: Class | (c.isStereoKinded("TransitionNotBetweenStates")
                                        or c.isStereotyped ("TransitionBetweenStates")
                                        or c.isStereotyped ("Region") )
                                        and not (c.isStereoKinded("State"))
```

3. SEP Profile

Definition. Well-formedness rules

OCL-Operators

allStereotypes

This operation returns a set containing the Stereotypes of the UML Element to which the operator is applied and all the Stereotypes inherited by such stereotypes.

```
Context Element:: allStereotypes:Set(Stereotype);
post: allStereotypes = self.extension.ownedEnd.type -> asSet() ->
                union (self.extension.ownedEnd.type.generalization.general.allStereotypes)
```

isStereokinded

This operation determines whether the UML Element to which the operator is applied, has a stereotype whose name is equal to the input name or, if it has a stereotype, one of whose ancestors' name is equal to the input name.

```
Context Element::isStereokinded (StereotypeName:String): Boolean;
post: result=self.allStereotypes -> exists (s: Stereotype | s.name= StereotypeName)
```

isStereotyped

This operation determines whether the UML Element to which the operator is applied has a stereotype whose name is equal to the input name.

```
Context Element::isStereotyped(StereotypeName:String): Boolean;
post: result = self.extension -> exists (e:Extension | e.ownedEnd.type.name = StereotypeName)
```

obtStereotype

This operation returns the UML Stereotype Element applied to the context element and whose name is equal to the input name.

```
Context Element:: obtStereotype(StereotypeName:String): Stereotype;
post: result = self.extension.ownedEnd.type -> asSet() ->
                                      select (s:Stereotype | s. name = StereotypeName)
```

Table of contents

1. Introduction and background
2. Development of the DFMS Framework
3. The SEP Profile
4. Application and Contributions of our Approach
5. Conclusions and future work

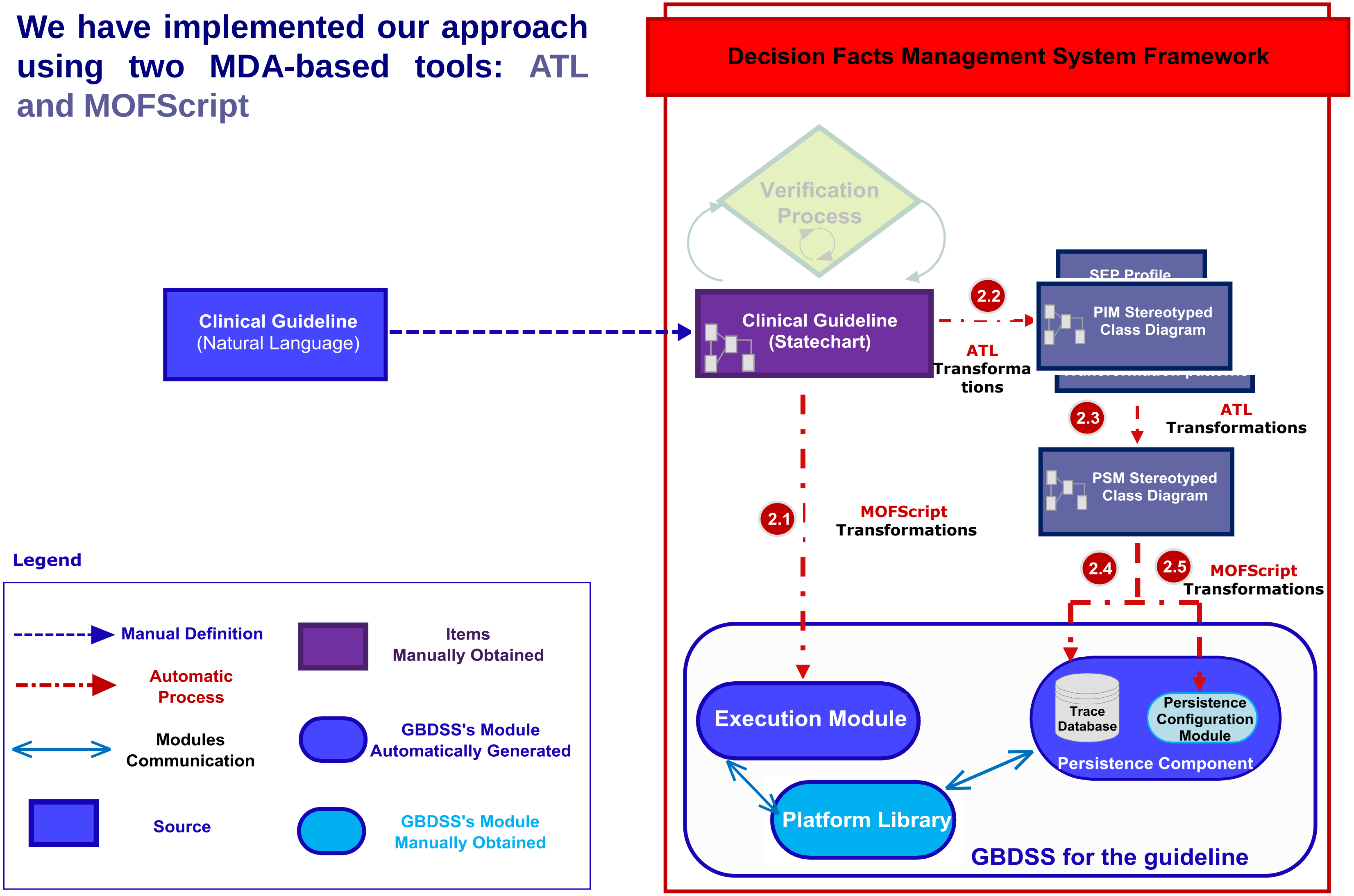
4. Application and Contributions of our Approach

- **The DFMS framework has been implemented as an Eclipse plug-in resulting in a first prototype.**
  - **We have implemented our model-to-model and model-to-text transformations using two MDD-based Eclipse plug-ins (the ATL M2M transformation tool and the MOFScript M2T transformation tool), which have been integrated into the developed plug-in.**

4. Application and Contributions of our Approach

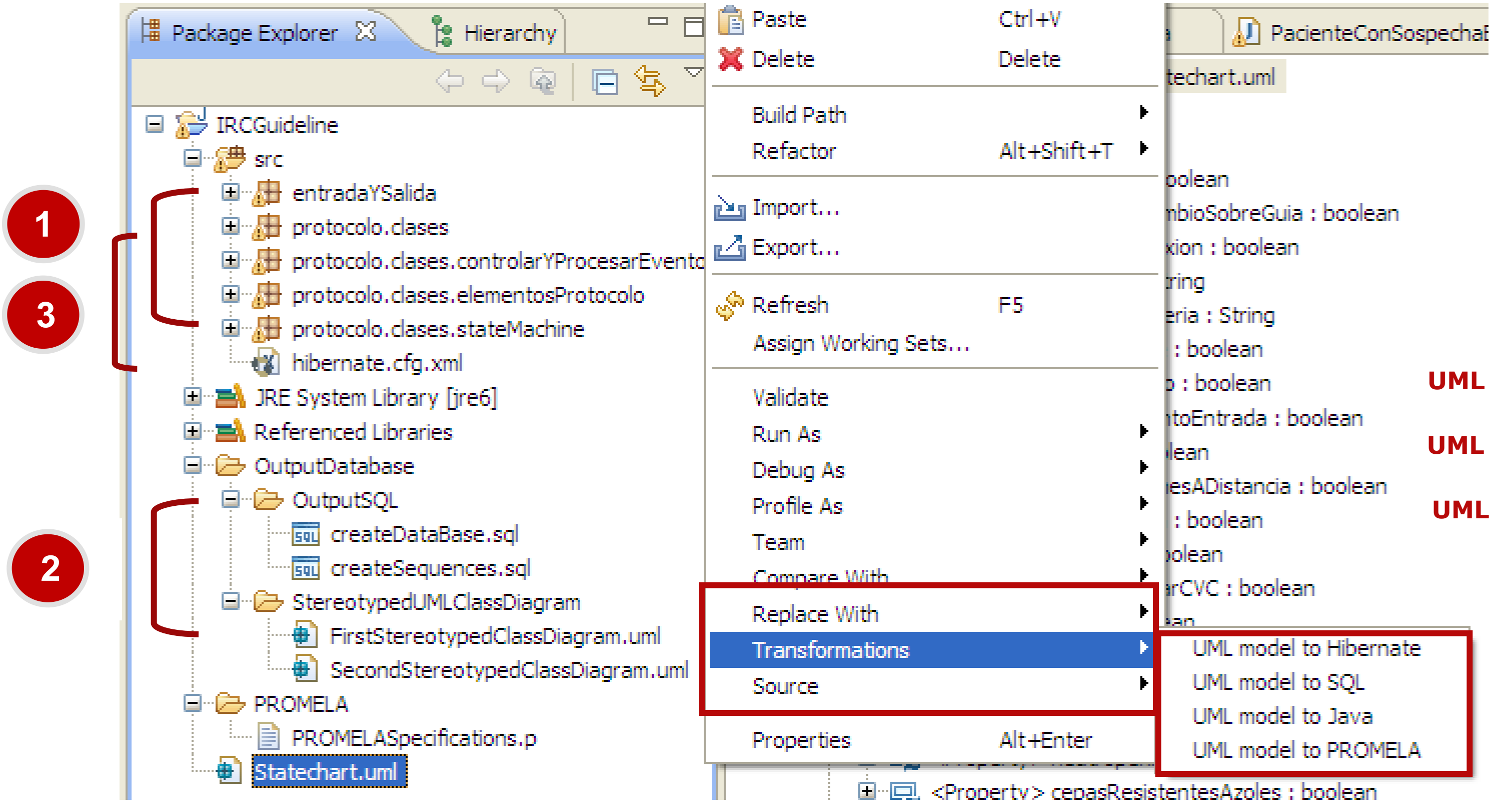
Development of the DFMS Framework

We have implemented our approach using two MDA-based tools: ATL and MOFScript



Decision Facts Management System Framework
Verification Process
Clinical Guideline (Natural Language)
Clinical Guideline (Statechart)
SEP Profile
PIM Stereotyped Class Diagram
2.2
ATL Transformations
2.3
ATL Transformations
PSM Stereotyped Class Diagram
2.1
MOFScript Transformations
2.4
2.5
MOFScript Transformations
Execution Module
Trace Database
Persistence Configuration Module
Persistence Component
Platform Library
GBDSS for the guideline
Legend
Manual Definition
Automatic Process
Modules Communication
Source
Items Manually Obtained
GBDSS's Module Automatically Generated
GBDSS's Module Manually Obtained

Development of the DFMS Framework



```
IRCGuideline
  src
    entradaYSalida
    protocolo.clases
    protocolo.clases.controlarYProcesarEvento
    protocolo.clases.elementosProtocolo
    protocolo.clases.stateMachine
    hibernate.cfg.xml
  JRE System Library [jre6]
  Referenced Libraries
  OutputDatabase
    OutputSQL
      createDataBase.sql
      createSequences.sql
    StereotypedUMLClassDiagram
      FirstStereotypedClassDiagram.uml
      SecondStereotypedClassDiagram.uml
  PROMELA
    PROMELASpecifications.p
  Statechart.uml
```

```
oolean
mbioSobreGuia : boolean
xion : boolean
tring
eria : String
: boolean
o : boolean
ntoEntrada : boolean
lean
esADistancia : boolean
: boolean
oolean
arCVC : boolean
ean
<Property> cepasResistentesAzoles : boolean
```

UML model to Java

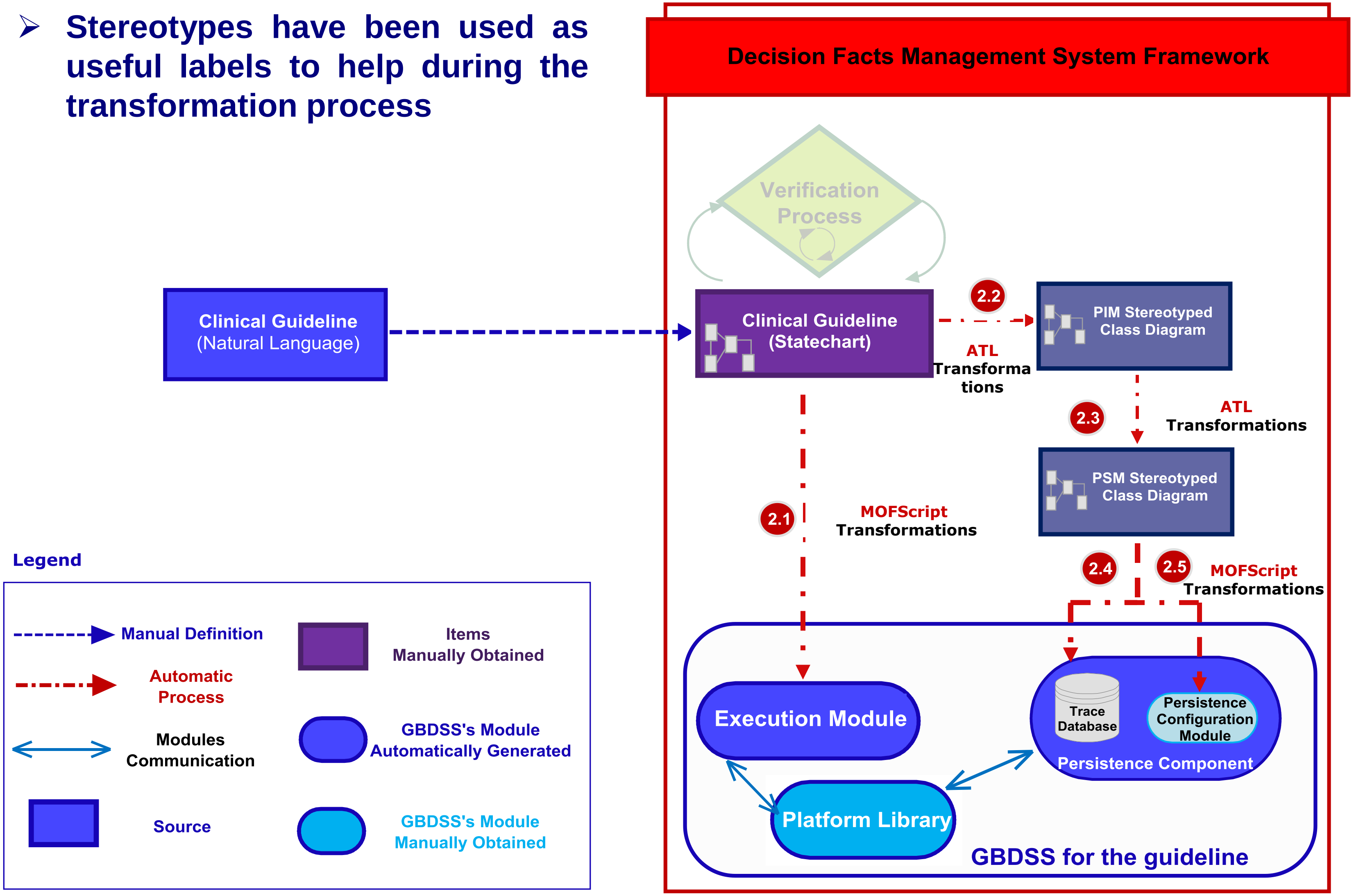
UML model to SQL

UML model to Hibernate

4. Application and Contributions of our Approach

Advantages of the SEP Profile

- Stereotypes have been used as useful labels to help during the transformation process

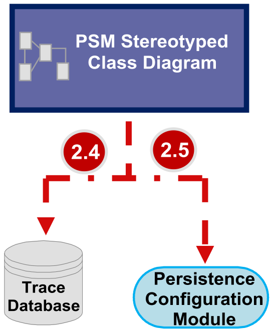


Decision Facts Management System Framework
Verification Process
Clinical Guideline (Natural Language)
Clinical Guideline (Statechart)
2.2
ATL Transformations
PIM Stereotyped Class Diagram
2.3
ATL Transformations
PSM Stereotyped Class Diagram
2.1
MOFScript Transformations
2.4
2.5
MOFScript Transformations
Execution Module
Trace Database
Persistence Configuration Module
Persistence Component
Platform Library
GBDSS for the guideline
Legend
Manual Definition
Automatic Process
Modules Communication
Source
Items Manually Obtained
GBDSS's Module Automatically Generated
GBDSS's Module Manually Obtained

4. Application and Contributions of our Approach

Advantages of the SEP Profile

- **Stereotypes have been used as useful labels to help during the transformation process**


PSM Stereotyped Class Diagram
2.4
2.5
Trace Database
Persistence Configuration Module

MOFScript rule

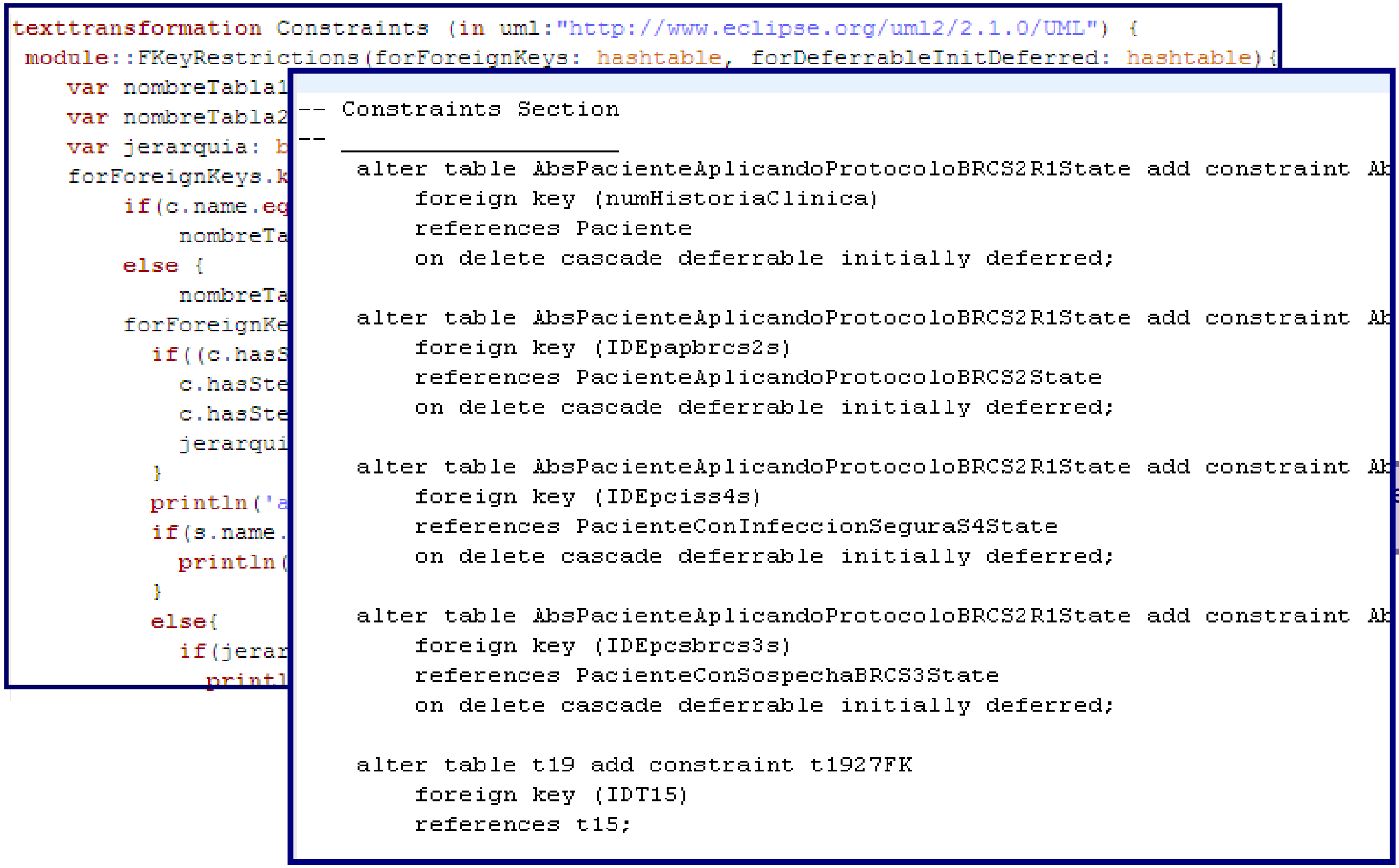
```
uml.Property::attributeToColumn(c:object){
  //Declaration of variables...
   //Create a column for primary keys
  if(self.name.startsWith('ID')){
    print('         '+self.name+'  ')
    self.primitiveTypeToSQLType()
    println( ' not null,')
    print('          constraint PK_'+c.reduceName()+' primary key ('+self.name+')')
    aux.add(''+self.name+'')
    indexConsUnique.put(c, aux)
  }
  //Create columns related to association properties in the class diagram wich will be foreign keys
  else if (self.association !=null){
    self.association.memberEnd -> forEach(pp:uml.Property| pp.name != self.name){
       theOtherRoleProperty=pp
    }
    //Particular case 1: Association properties which come from the
                     //composite properties in composite states
    if(c.hasStereotype("NonOrthogonalCompositeState") or c.hasStereotype("OrthogonalCompositeState")
                                               and self.name.startsWith("roleLastStateInRegion")){
      if(self.type.hasStereotype("Region") and self.name.startsWith("roleLastState")){
        foreignKey= self.type.obtainPrimaryKey()
        print('       '+foreignKey.name+'LastS  ')
        foreignKey.primitiveTypeToSQLType()
        println(' null, ')
        //Gather information in collections
        //Variable used in createContraintsInCreateTable rule for defining Uniqueness constraints
        //for foreign keys (FKs) in 'Create table'
        if(forContraintsInCreateTable.get(c)==null){
           aux2.add(self)
           forConstraintsInCreateTable.put(c, aux2)
        }
        else{
           forConstraintsInCreateTable.get(c).add(self)
        }
        //Variable used in FKeyConstraints rule for creating FKs in 'Alter table' instructions
        if(forForeignKeys.get(c)==null){
```

Depending on the applied stereotypes, different text code is generated

4. Application and Contributions of our Approach

Advantages of the SEP Profile

- Specific stereotypes are transformed into concrete SQL statements in the data base (foreign keys and triggers).



```
texttransformation Constraints (in uml:"http://www.eclipse.org/uml2/2.1.0/UML") {
 module::FKeyRestrictions(forForeignKeys: hashtable, forDeferrableInitDeferred: hashtable){
    var nombreTabla1
    var nombreTabla2
    var jerarquia: b
    forForeignKeys.k
        if(c.name.eq
            nombreTa
        else {
            nombreTa
        forForeignKe
          if((c.hasS
            c.hasSte
            c.hasSte
            jerarqui
          }
          println('a
          if(s.name.
            println(
          }
          else{
            if(jerar
              printl
```

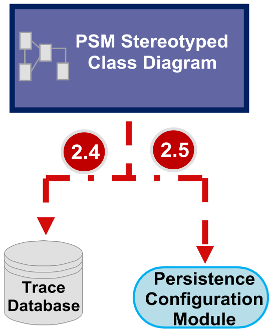
```
-- Constraints Section
-- ________________
    alter table AbsPacienteAplicandoProtocoloBRCS2R1State add constraint Ab
        foreign key (numHistoriaClinica)
        references Paciente
        on delete cascade deferrable initially deferred;

    alter table AbsPacienteAplicandoProtocoloBRCS2R1State add constraint Ab
        foreign key (IDEpapbrcs2s)
        references PacienteAplicandoProtocoloBRCS2State
        on delete cascade deferrable initially deferred;

    alter table AbsPacienteAplicandoProtocoloBRCS2R1State add constraint Ab
        foreign key (IDEpciss4s)
        references PacienteConInfeccionSeguraS4State
        on delete cascade deferrable initially deferred;

    alter table AbsPacienteAplicandoProtocoloBRCS2R1State add constraint Ab
        foreign key (IDEpcsbrcs3s)
        references PacienteConSospechaBRCS3State
        on delete cascade deferrable initially deferred;

    alter table t19 add constraint t1927FK
        foreign key (IDT15)
        references t15;
```


PSM Stereotyped Class Diagram
2.4
2.5
Trace Database
Persistence Configuration Module

eating foreign keys using `alter table` statements

Generated constraints

4. Application and Contributions of our Approach

Advantages of the SEP Profile

- **Ensures accuracy and consistency of data**

  - **Specific semantics of the stereotypes have been implemented as concrete constraints and triggers in the trace database.**

- **Guides the transformation process**

  - **We have defined the M2M and the M2T transformations (from the PIM to the persistence component) in such a way that they use the stereotype names as useful marks to help during the transformation process.**

- **Eases audit and process improvement**

  - **By means of the semantics provided by the stereotypes of the SEP profile, we obtain a more complete trace of the system, represented in the persistence structures of the trace database.**

  - **Such a trace constitutes a meaningful source of the information generated during the application of the guideline to the patient, easing enhanced auditing processes.**

Case Study. IRC Guideline

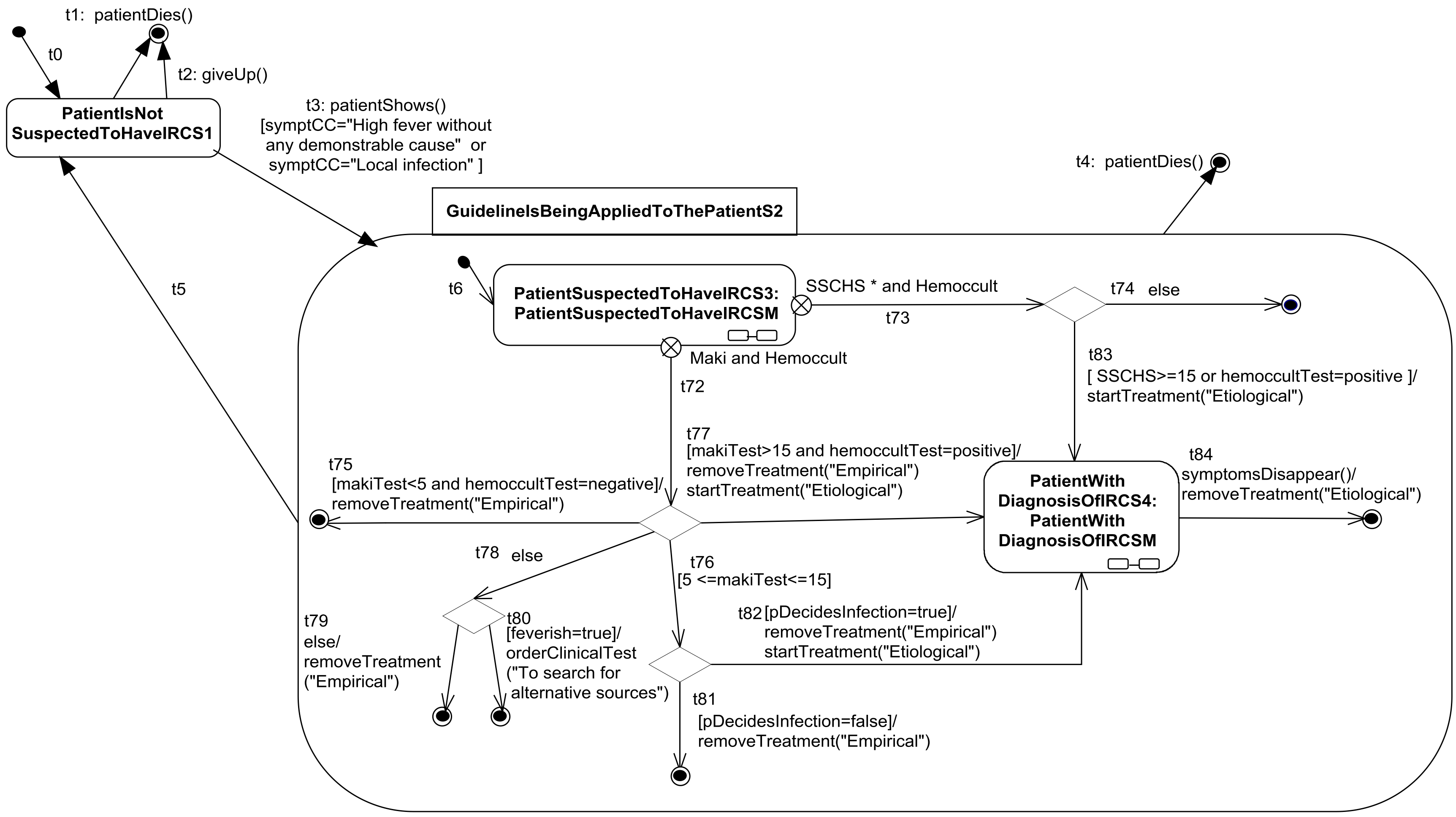
IRC Guideline

- It is used in a Spanish hospital and has been developed on the basis of a guideline published by the US Agency for Health Care Research and Quality (AHRQ) National Guideline Clearing House (NGC).

- It has been defined mainly for diagnosis and prevention.

- Representation: text document of 10 pages, written in natural language with tables and flowcharts.

Statechart Representation



t1: patientDies()

t0

t2: giveUp()

PatientIsNot SuspectedToHaveIRCS1

t3: patientShows() [symptCC="High fever without any demonstrable cause" or symptCC="Local infection"]

t4: patientDies()

GuidelineIsBeingAppliedToThePatientS2

t5

t6

PatientSuspectedToHaveIRCS3: PatientSuspectedToHaveIRCSM

SSCHS * and Hemoccult

t73

t74 else

Maki and Hemoccult

t72

t83 [SSCHS>=15 or hemoccultTest=positive]/ startTreatment("Etiological")

t77 [makiTest>15 and hemoccultTest=positive]/ removeTreatment("Empirical") startTreatment("Etiological")

t75 [makiTest<5 and hemoccultTest=negative]/ removeTreatment("Empirical")

PatientWith DiagnosisOfIRCS4: PatientWith DiagnosisOfIRCSM

t84 symptomsDisappear()/ removeTreatment("Etiological")

t78 else

t76 [5 <=makiTest<=15]

t79 else/ removeTreatment ("Empirical")

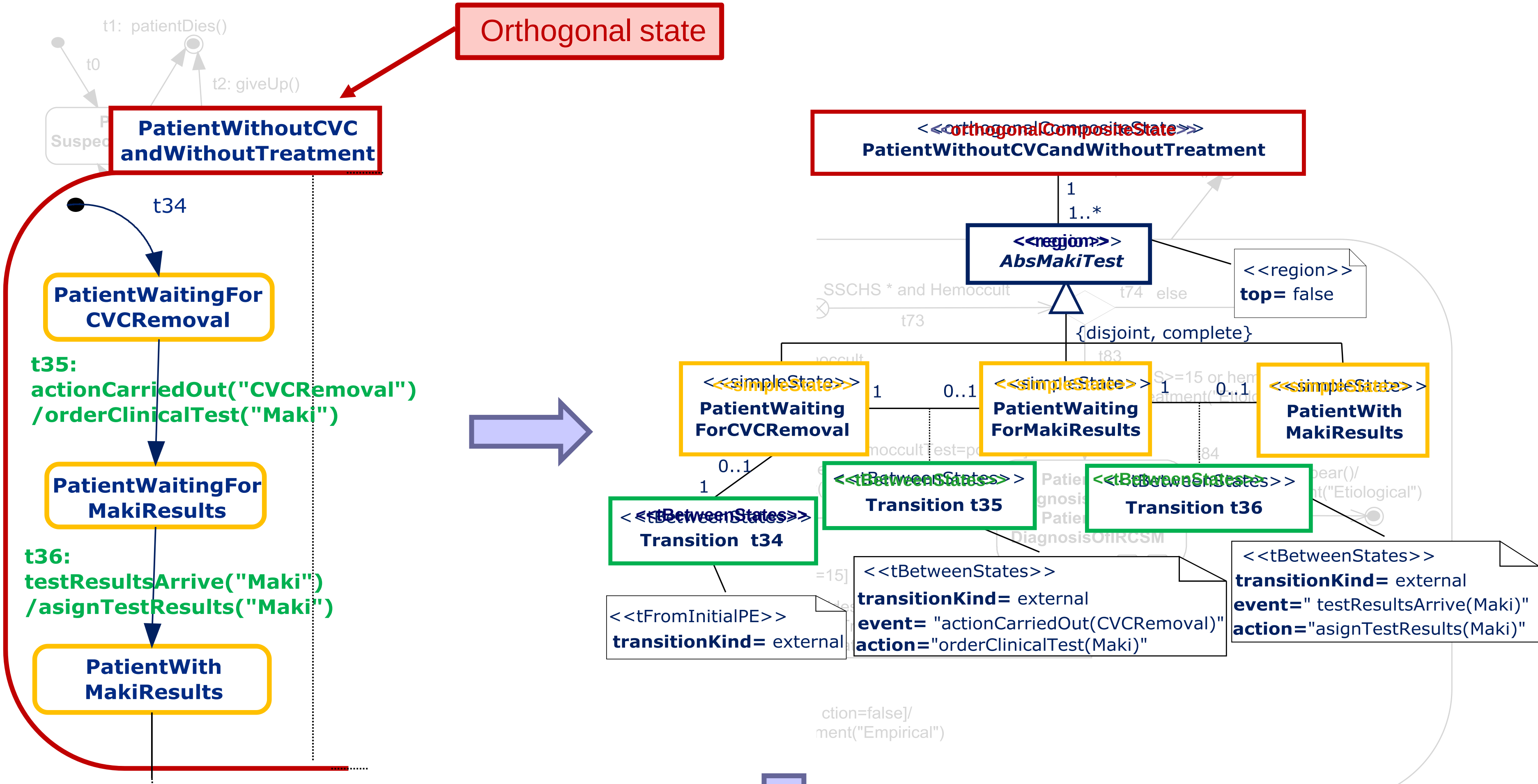
t80 [feverish=true]/ orderClinicalTest ("To search for alternative sources")

t82 [pDecidesInfection=true]/ removeTreatment("Empirical") startTreatment("Etiological")

t81 [pDecidesInfection=false]/ removeTreatment("Empirical")

* SSCHS: Study of semicuantitative cultures of catheter hub and skin

2.2 From the Statechart to the PIM Stereotyped Class Diagram

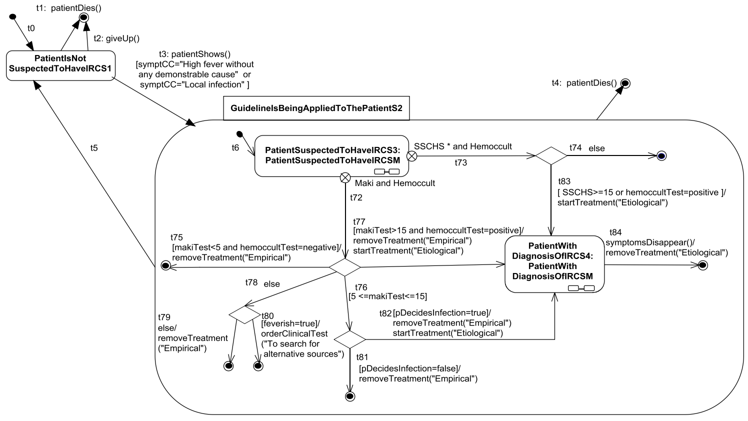

Orthogonal state
t1: patientDies()
t0
t2: giveUp()
Suspec
PatientWithoutCVC andWithoutTreatment
t34
PatientWaitingFor CVCRemoval
t35: actionCarriedOut("CVCRemoval") /orderClinicalTest("Maki")
PatientWaitingFor MakiResults
t36: testResultsArrive("Maki") /asignTestResults("Maki")
PatientWith MakiResults
<<orthogonalCompositeState>>
PatientWithoutCVCandWithoutTreatment
1
1..*
<<region>>
AbsMakiTest
<<region>>
top= false
{disjoint, complete}
<<simpleState>>
PatientWaiting ForCVCRemoval
1
0..1
<<simpleState>>
PatientWaiting ForMakiResults
1
0..1
<<simpleState>>
PatientWith MakiResults
0..1
1
<<tBetweenStates>>
Transition t34
<<tBetweenStates>>
Transition t35
<<tBetweenStates>>
Transition t36
<<tFromInitialPE>>
transitionKind= external
<<tBetweenStates>>
transitionKind= external
event= "actionCarriedOut(CVCRemoval)"
action="orderClinicalTest(Maki)"
<<tBetweenStates>>
transitionKind= external
event=" testResultsArrive(Maki)"
action="asignTestResults(Maki)"

Transformation rules + **SEP Profile**

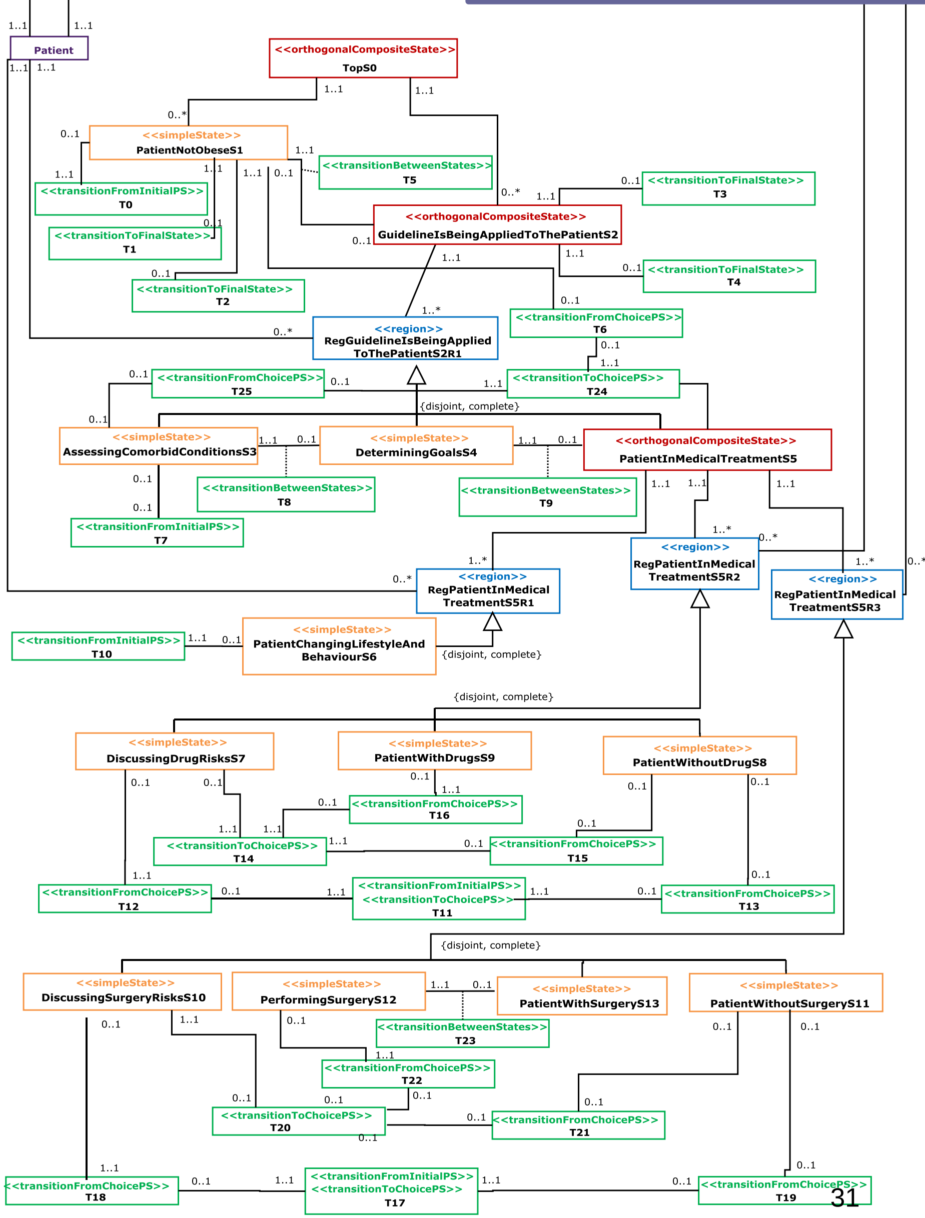
4. Application and Contributions of our Approach IRC Guideline

2.2 From the Statechart to the PIM Stereotyped Class Diagram

Clinical Guideline (Statechart) → 2.2 → PIM Stereotyped Class Diagram


t1: patientDies()
t0
t2: giveUp()
PatientIsNot SuspectedToHaveIRCS1
t3: patientShows() [symptCC="High fever without any demonstrable cause" or symptCC="Local infection"]
t4: patientDies()
GuidelineIsBeingAppliedToThePatientS2
t5
t6
PatientSuspectedToHaveIRCS3: PatientSuspectedToHaveIRCSM
SSCHS * and Hemoccult
t73
t74 else
Maki and Hemoccult
t72
t83 [SSCHS>=15 or hemoccultTest=positive]/ startTreatment("Etiological")
t77 [makiTest>15 and hemoccultTest=positive]/ removeTreatment("Empirical") startTreatment("Etiological")
t75 [makiTest<5 and hemoccultTest=negative]/ removeTreatment("Empirical")
PatientWith DiagnosisOfIRCS4: PatientWith DiagnosisOfIRCSM
t84 symptomsDisappear()/ removeTreatment("Etiological")
t78 else
t76 [5 <=makiTest<=15]
t79 else/ removeTreatment ("Empirical")
t80 [feverish=true]/ orderClinicalTest ("To search for alternative sources")
t82 [pDecidesInfection=true]/ removeTreatment("Empirical") startTreatment("Etiological")
t81 [pDecidesInfection=false]/ removeTreatment("Empirical")

* SSCHS: Study of semicuantitative cultures of catheter hub and skin


Patient
<<orthogonalCompositeState>> TopS0
<<simpleState>> PatientNotObeseS1
<<transitionBetweenStates>> T5
<<transitionFromInitialPS>> T0
<<transitionToFinalState>> T1
<<transitionToFinalState>> T2
<<transitionToFinalState>> T3
<<orthogonalCompositeState>> GuidelineIsBeingAppliedToThePatientS2
<<transitionToFinalState>> T4
<<transitionFromChoicePS>> T6
<<region>> RegGuidelineIsBeingApplied ToThePatientS2R1
<<transitionFromChoicePS>> T25
<<transitionToChoicePS>> T24
{disjoint, complete}
<<simpleState>> AssessingComorbidConditionsS3
<<simpleState>> DeterminingGoalsS4
<<orthogonalCompositeState>> PatientInMedicalTreatmentS5
<<transitionBetweenStates>> T8
<<transitionBetweenStates>> T9
<<transitionFromInitialPS>> T7
<<region>> RegPatientInMedical TreatmentS5R2
<<region>> RegPatientInMedical TreatmentS5R1
<<region>> RegPatientInMedical TreatmentS5R3
<<transitionFromInitialPS>> T10
<<simpleState>> PatientChangingLifestyleAnd BehaviourS6
{disjoint, complete}
{disjoint, complete}
<<simpleState>> DiscussingDrugRisksS7
<<simpleState>> PatientWithDrugsS9
<<simpleState>> PatientWithoutDrugS8
<<transitionFromChoicePS>> T16
<<transitionToChoicePS>> T14
<<transitionFromChoicePS>> T15
<<transitionFromChoicePS>> T12
<<transitionFromInitialPS>> <<transitionToChoicePS>> T11
<<transitionFromChoicePS>> T13
{disjoint, complete}
<<simpleState>> DiscussingSurgeryRisksS10
<<simpleState>> PerformingSurgeryS12
<<simpleState>> PatientWithSurgeryS13
<<simpleState>> PatientWithoutSurgeryS11
<<transitionBetweenStates>> T23
<<transitionFromChoicePS>> T22
<<transitionToChoicePS>> T20
<<transitionFromChoicePS>> T21
<<transitionFromChoicePS>> T18
<<transitionFromInitialPS>> <<transitionToChoicePS>> T17
<<transitionFromChoicePS>> T19
1..1
0..1
0..*
1..*

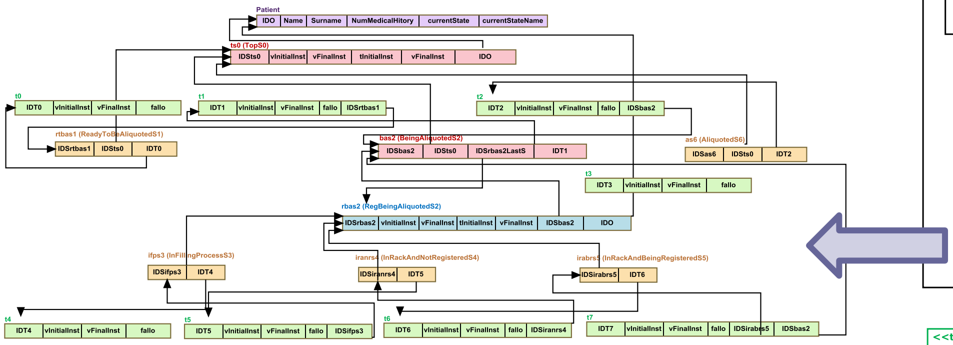
4. Application and Contributions of our Approach

2.3 2.2

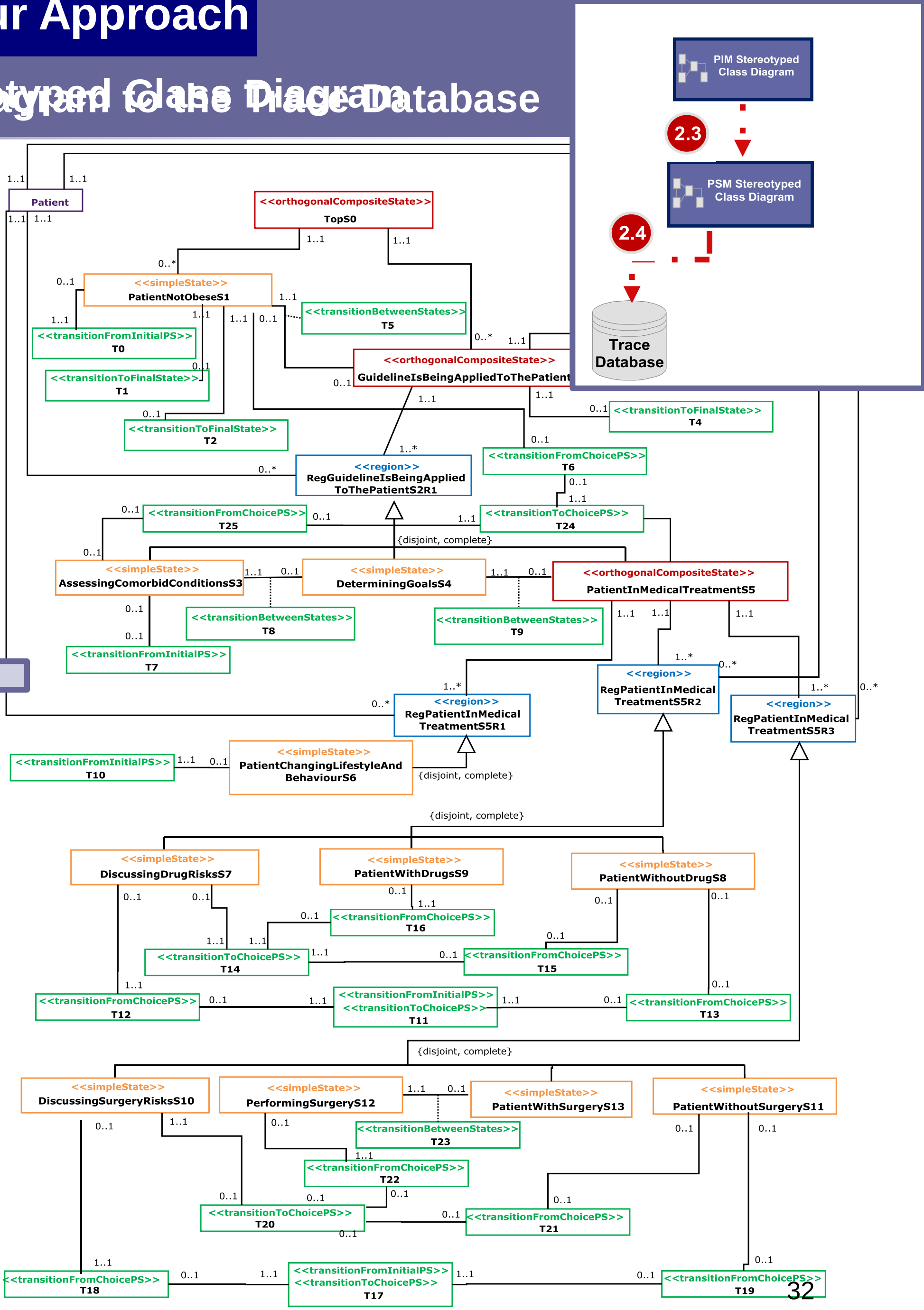
From the Statechart to the PIM Stereotyped Class Diagram

From the PIM Stereotyped Class Diagram to the Trace Database

The trace database comprises 152 tables.



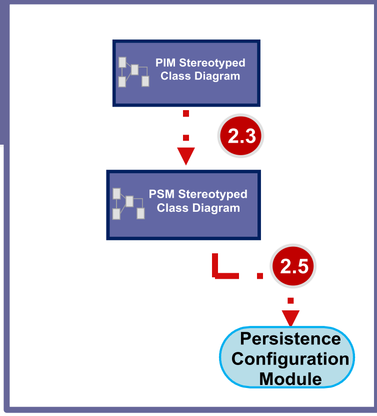
Patient
IDO | Name | Surname | NumMedicalHitory | currentState | currentStateName
ts0 (TopS0)
IDSts0 | vInitialInst | vFinalInst | tInitialInst | vFinalInst | IDO
t0
IDT0 | vInitialInst | vFinalInst | fallo
t1
IDT1 | vInitialInst | vFinalInst | fallo | IDSrtbas1
t2
IDT2 | vInitialInst | vFinalInst | fallo | IDSbas2
rtbas1 (ReadyToBeAliquotedS1)
IDSrtbas1 | IDSts0 | IDT0
bas2 (BeingAliquotedS2)
IDSbas2 | IDSts0 | IDSrbas2LastS | IDT1
as6 (AliquotedS6)
IDSas6 | IDSts0 | IDT2
t3
IDT3 | vInitialInst | vFinalInst | fallo
rbas2 (RegBeingAliquotedS2)
IDSrbas2 | vInitialInst | vFinalInst | tInitialInst | vFinalInst | IDSbas2 | IDO
ifps3 (InFillingProcessS3)
IDSifps3 | IDT4
iranrs4 (InRackAndNotRegisteredS4)
IDSiranrs4 | IDT5
irabrs5 (InRackAndBeingRegisteredS5)
IDSirabrs5 | IDT6
t4
IDT4 | vInitialInst | vFinalInst | fallo
t5
IDT5 | vInitialInst | vFinalInst | fallo | IDSifps3
t6
IDT6 | vInitialInst | vFinalInst | fallo | IDSiranrs4
t7
IDT7 | vInitialInst | vFinalInst | fallo | IDSirabrs5 | IDSbas2



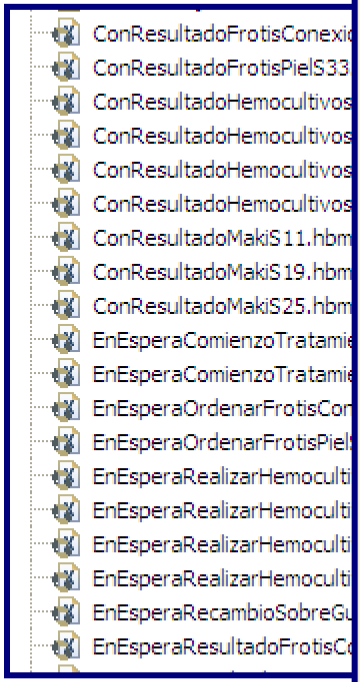
PIM Stereotyped Class Diagram
2.3
PSM Stereotyped Class Diagram
2.4
Trace Database
Patient
<<orthogonalCompositeState>> TopS0
<<simpleState>> PatientNotObeseS1
<<transitionFromInitialPS>> T0
<<transitionToFinalState>> T1
<<transitionToFinalState>> T2
<<transitionBetweenStates>> T5
<<orthogonalCompositeState>> GuidelineIsBeingAppliedToThePatient
<<transitionToFinalState>> T4
<<region>> RegGuidelineIsBeingApplied ToThePatientS2R1
<<transitionFromChoicePS>> T6
<<transitionFromChoicePS>> T25
<<transitionToChoicePS>> T24
{disjoint, complete}
<<simpleState>> AssessingComorbidConditionsS3
<<simpleState>> DeterminingGoalsS4
<<orthogonalCompositeState>> PatientInMedicalTreatmentS5
<<transitionBetweenStates>> T8
<<transitionBetweenStates>> T9
<<transitionFromInitialPS>> T7
<<region>> RegPatientInMedical TreatmentS5R1
<<region>> RegPatientInMedical TreatmentS5R2
<<region>> RegPatientInMedical TreatmentS5R3
<<transitionFromInitialPS>> T10
<<simpleState>> PatientChangingLifestyleAnd BehaviourS6
{disjoint, complete}
{disjoint, complete}
<<simpleState>> DiscussingDrugRisksS7
<<simpleState>> PatientWithDrugsS9
<<simpleState>> PatientWithoutDrugS8
<<transitionFromChoicePS>> T16
<<transitionToChoicePS>> T14
<<transitionFromChoicePS>> T15
<<transitionFromChoicePS>> T12
<<transitionFromInitialPS>> <<transitionToChoicePS>> T11
<<transitionFromChoicePS>> T13
{disjoint, complete}
<<simpleState>> DiscussingSurgeryRisksS10
<<simpleState>> PerformingSurgeryS12
<<simpleState>> PatientWithSurgeryS13
<<simpleState>> PatientWithoutSurgeryS11
<<transitionBetweenStates>> T23
<<transitionFromChoicePS>> T22
<<transitionToChoicePS>> T20
<<transitionFromChoicePS>> T21
<<transitionFromChoicePS>> T18
<<transitionFromInitialPS>> <<transitionToChoicePS>> T17
<<transitionFromChoicePS>> T19
1..1
0..1
0..*
1..*

4. Application and Contributions of our Approach

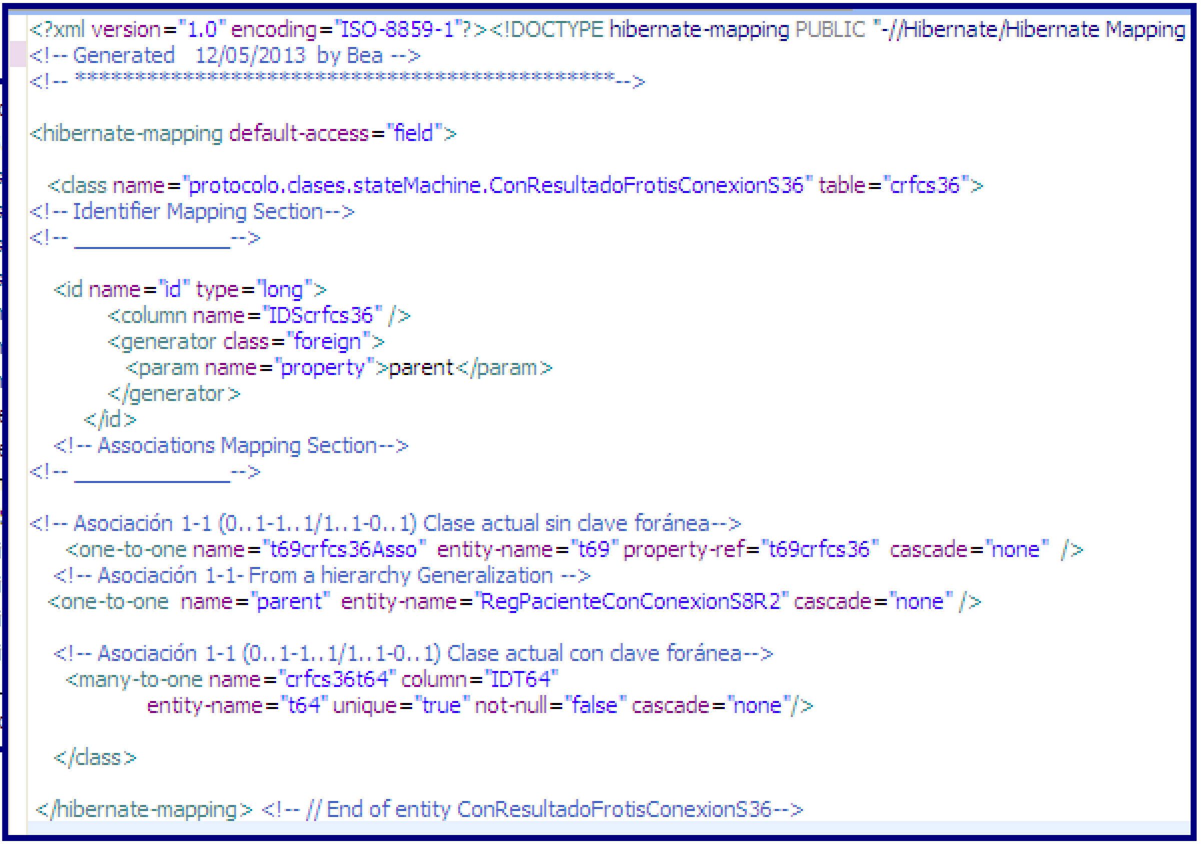
2.3 2.5 From the PIM Stereotyped Class Diagram to the Persistence C. M.


PIM Stereotyped Class Diagram
2.3
PSM Stereotyped Class Diagram
2.5
Persistence Configuration Module

The persistence configuration module comprises 151 mapping files, together with the `hibernate.cfg.xml` **configuration file.**



```
ConResultadoFrotisConexi
ConResultadoFrotisPielS33
ConResultadoHemocultivos
ConResultadoHemocultivos
ConResultadoHemocultivos
ConResultadoHemocultivos
ConResultadoMakiS11.hbm
ConResultadoMakiS19.hbm
ConResultadoMakiS25.hbm
EnEsperaComienzoTratami
EnEsperaComienzoTratami
EnEsperaOrdenarFrotisCon
EnEsperaOrdenarFrotisPiel
EnEsperaRealizarHemoculti
EnEsperaRealizarHemoculti
EnEsperaRealizarHemoculti
EnEsperaRealizarHemoculti
EnEsperaRecambioSobreGu
EnEsperaResultadoFrotisC
```



```
<?xml version="1.0" encoding="ISO-8859-1"?><!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping
<!-- Generated   12/05/2013  by Bea -->
<!-- ****************************************************-->

<hibernate-mapping default-access="field">

  <class name="protocolo.clases.stateMachine.ConResultadoFrotisConexionS36" table="crfcs36">
<!-- Identifier Mapping Section-->
<!-- _______________-->

   <id name="id" type="long">
         <column name="IDScrfcs36" />
         <generator class="foreign">
           <param name="property">parent</param>
         </generator>
      </id>
   <!-- Associations Mapping Section-->
<!-- _______________-->

<!-- Asociación 1-1 (0..1-1..1/1..1-0..1) Clase actual sin clave foránea-->
    <one-to-one name="t69crfcs36Asso"  entity-name="t69" property-ref="t69crfcs36"  cascade="none"  />
   <!-- Asociación  1-1- From a hierarchy Generalization -->
  <one-to-one  name="parent"  entity-name="RegPacienteConConexionS8R2" cascade="none" />

   <!-- Asociación 1-1 (0..1-1..1/1..1-0..1) Clase actual con clave foránea-->
    <many-to-one name="crfcs36t64" column="IDT64"
            entity-name="t64" unique="true" not-null="false" cascade="none"/>

   </class>

 </hibernate-mapping> <!-- // End of entity ConResultadoFrotisConexionS36-->
```

4. Application and Contributions of our Approach

- The DFMS framework has been implemented as an Eclipse plug-in resulting in a first prototype.
  - We have implemented our model-to-model and model-to-text transformations using two MDD-based Eclipse plug-ins (the ATL M2M transformation tool and the MOFScript M2T transformation tool), which have been integrated into the developed plug-in.

- The DFMS framework has been satisfactorily applied to several real-life guidelines used in different contexts within the medical care system, obtaining different GBDSSs working prototypes.
  - The guidelines used as case studies are:
    - IRC Guideline,
    - OPC Guideline: a clinical guideline for the management of obesity in primary care,
    - AP Guideline: a laboratory guideline to carry out the aliquoting process.

Table of contents

1. Introduction and background
2. Development of the DFMS Framework
3. The SEP Profile
4. Application and Contributions of our Approach
5. Conclusions and future work

Conclusions

- **We have given an overview of the definition and development of out DFMS Framework.**

- **We have presented the definition of the SEP profile:**

  - **It contributes to ease audit and process improvement.**

  - **The implementation of the semantics of its stereotypes helps to ensure accuracy and consistency of the data stored in the database.**

  - **Stereotype names have been used as useful marks to help during the transformation process.**

Future work

- To apply our approach to other frameworks in which statecharts are also used to represent the system behaviour.

- To apply our approach to other dynamic modeling languages that may be used to represent guidelines.

- To use our system trace proposal with a specific standard or best practice guide for information technology.

A UML Profile for Dynamic Execution Persistence with Monitoring Purposes

Questions?

MiSE’13

May 19th, San Francisco, CA