OWEB :
Objectiver en mode SaaS

Robert Darimont, PhD
CEO
Robert.darimont@respect-it.be

Table des matières

- Introduction
  - Profil & Historique
  - Credo
  - ODE
- OWEB
  - Architecture
  - Survol des fonctionnalités avec démo
  - Le coin des techies
  - RAF

Une approche par partenariat

Recherche

UCL UCL

A. van Lamsweerde *E. Letier*

Objectiver

R. Darimont

C. Ponsard

Ocetic
Your Connection to ICT Research

Respect-IT

Recherche appliquée
Prototypes
Transfert de Technologies
Consultance technologique

Services
Editeur de logiciels

Business





Convient à tous les domaines

Aeronautics

EUROCONTROL

SAFEE

ONERA

NATS

ALCATEL

Industry

HUAWEI

EVS

ALCYON BELUX

delfood DIVISION OF LOUIS DELHAIZE GROUP

arcelor

Services

FÉDÉRATION WALLONIE-BRUXELLES

etnic

Ocetic Your Connection to ICT Research

Service Public Fédéral FINANCES

MJ

UCL

gan EUROCOURTAGE c'est avec l'esprit libre qu'on avance

Healthcare

Pfizer

Alpha Répartition

CENTRE HOSPITALIER REGIONAL DE LA CITADELLE BOULEVARD DU 12e DE LIGNE 1, 4000 LIEGE

Edition

LA LIBRE BELGIQUE

DUPUIS

GROUP plus

l'avenir

Un peu d'histoire…

2005

Objectiver

Edition Desktop (ODE)

2000

GRAIL **proto**

2006

Essaimage
Consultance en IE
Editeur d'Objectiver

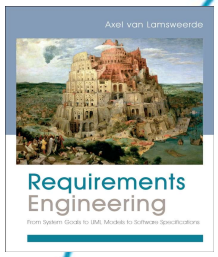
Respect-IT

1996

CEDITI

UCL Tech Transfer Center
Premières expérimentations

2009



Livre d'Axel
Wiley

1993

UCL

1ère publication scientifique
Goal-directed requirements acquisition
Dardenne, Fickas, van Lamsweerde
SCP, 1993

2019

Objectiver

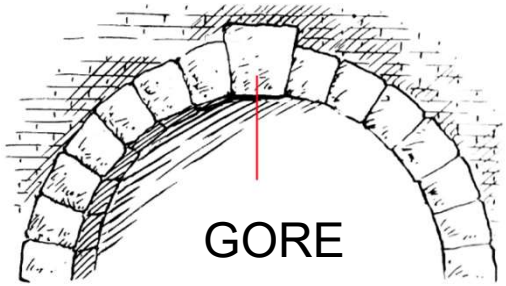
Edition Saas

Table des matières

- Introduction
  - Profil & Historique
  - Credo
  - ODE
- OWEB
  - Architecture
  - Survol des fonctionnalités avec démo
  - Le coin des techies
  - RAF

Ce que l'on propose…

- Une clef de voûte pour le développement de projets (quel que soit le cycle de vie)



- Une méthode pour dériver des SR à partir d'UR : focus sur l'analyse du PROBLEME
- Complémentarité avec des méthodes existantes (UML, SysML, …)
- Outillage

Orientation But (KAOS)

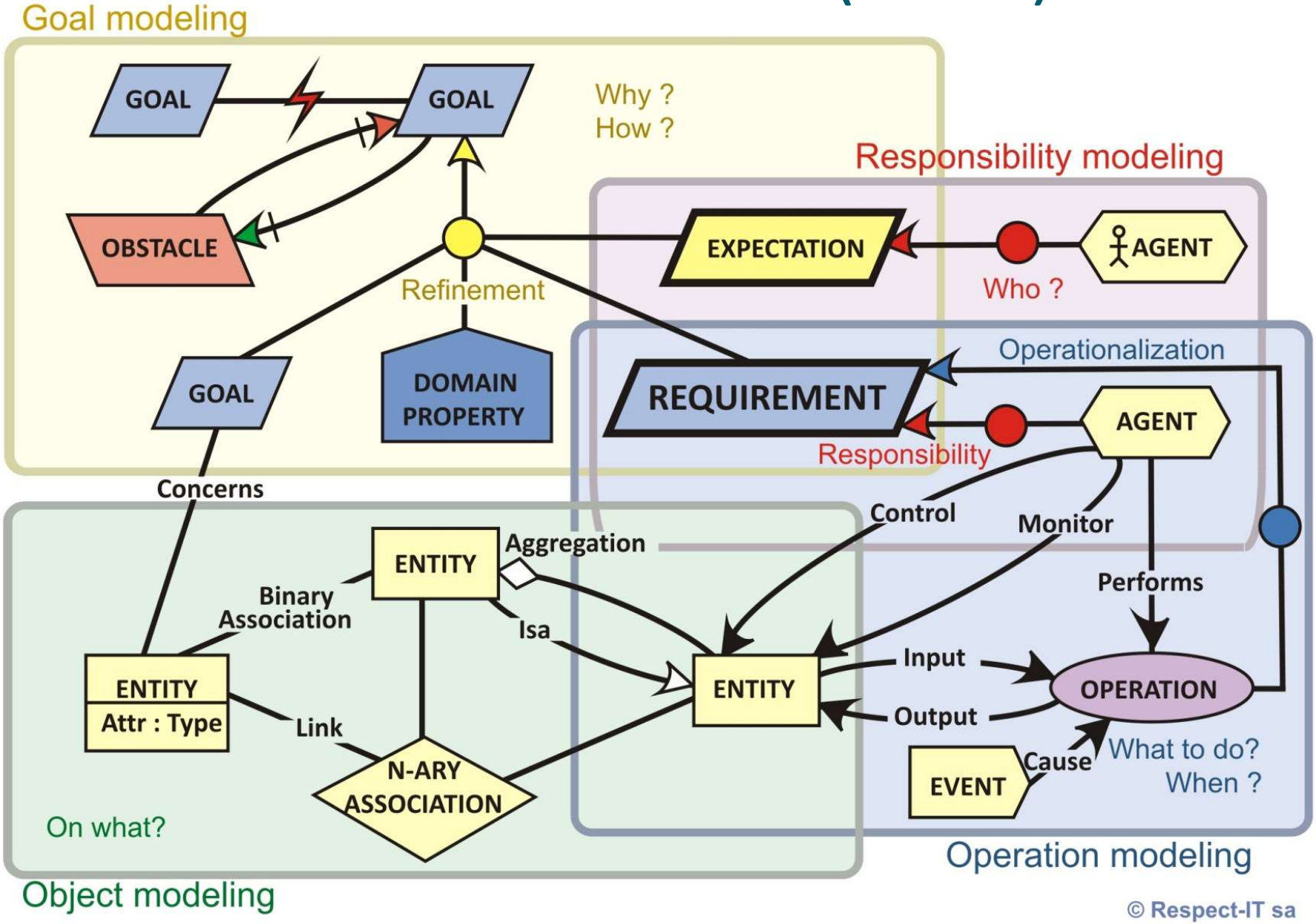
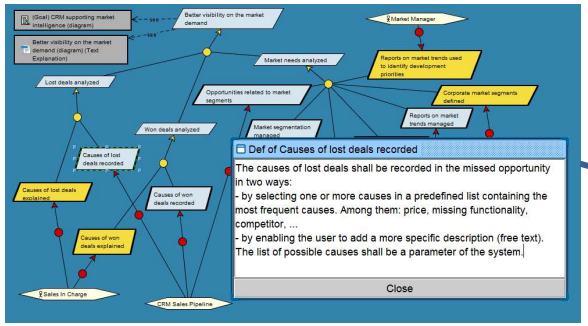


Table des matières

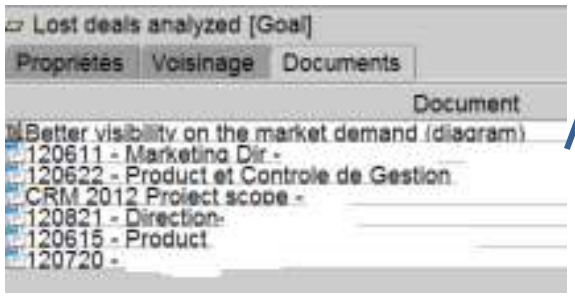
- Introduction
  - Profil & Historique
  - Credo
  - ODE
- OWEB
  - Architecture
  - Survol des fonctionnalités avec démo
  - Le coin des techies
  - RAF

Objectiver Desktop

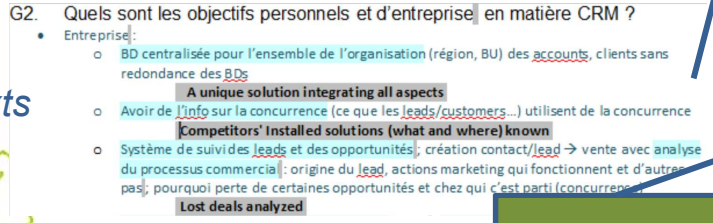
Diagrams



Properties



Texts



Internal representation

Model editors

Text editor (Open Office)

Query tool(OQL)

Import Export

Plugins

Generators

.jar file

Fichier .ob

.xml file

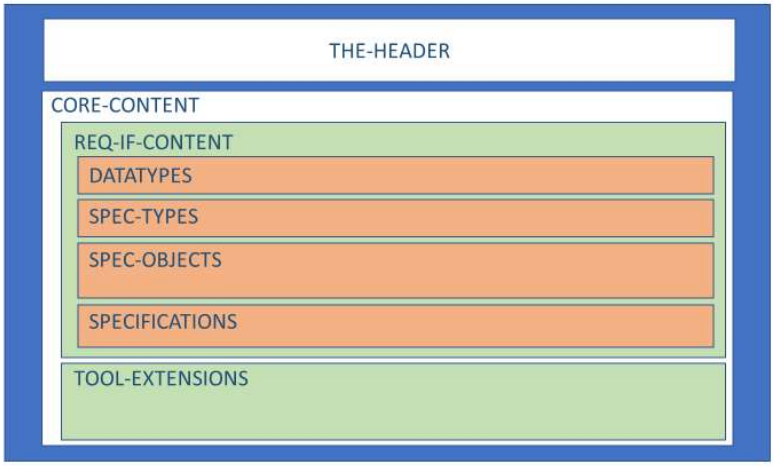
.odt file

.odt file

.rtf file

Plugins récents : ReqIF

- Export des exigences et attentes au format ReqIF
- Standard pour IBM (Doors, …)
- Export vers ReqIF Studio



Plugins récents : USE

- Formalisation des exigences en OCL
- Pré/post sur les opérations en OCL
- Renforcement des pré-post pour satisfaire les exigences
- Export vers USE

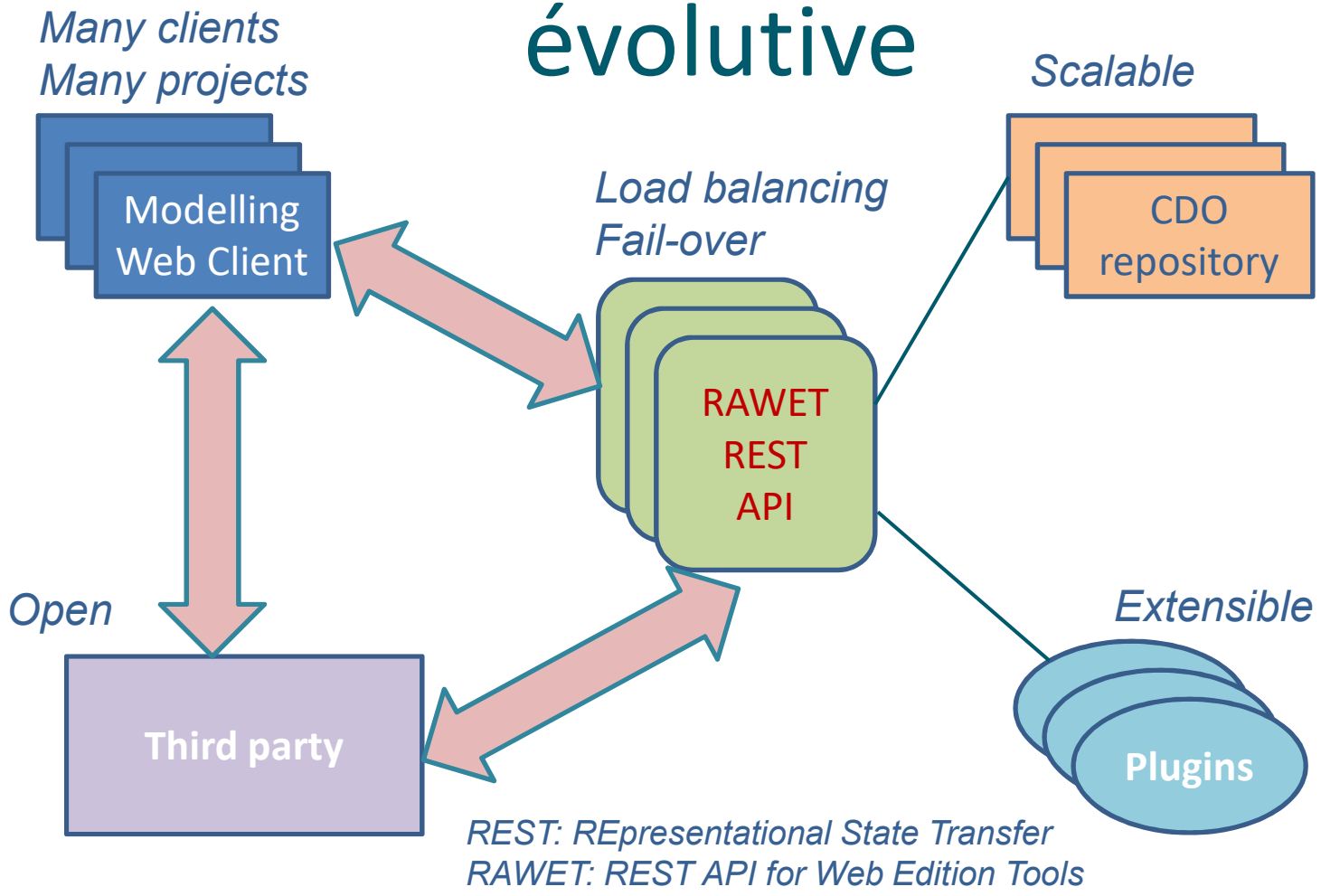
Pourquoi OWEB ?

- Limitations ODE
  - essentiellement mono-utilisateur, mono-poste
  - ouverture limitée
  - fichier XML propriétaire
- Demande du marché
  - paradigme Cloud/Web
  - très ouvert pour intégration aisée dans des chaînes d'outils
  - travail coopératif
  - base de données

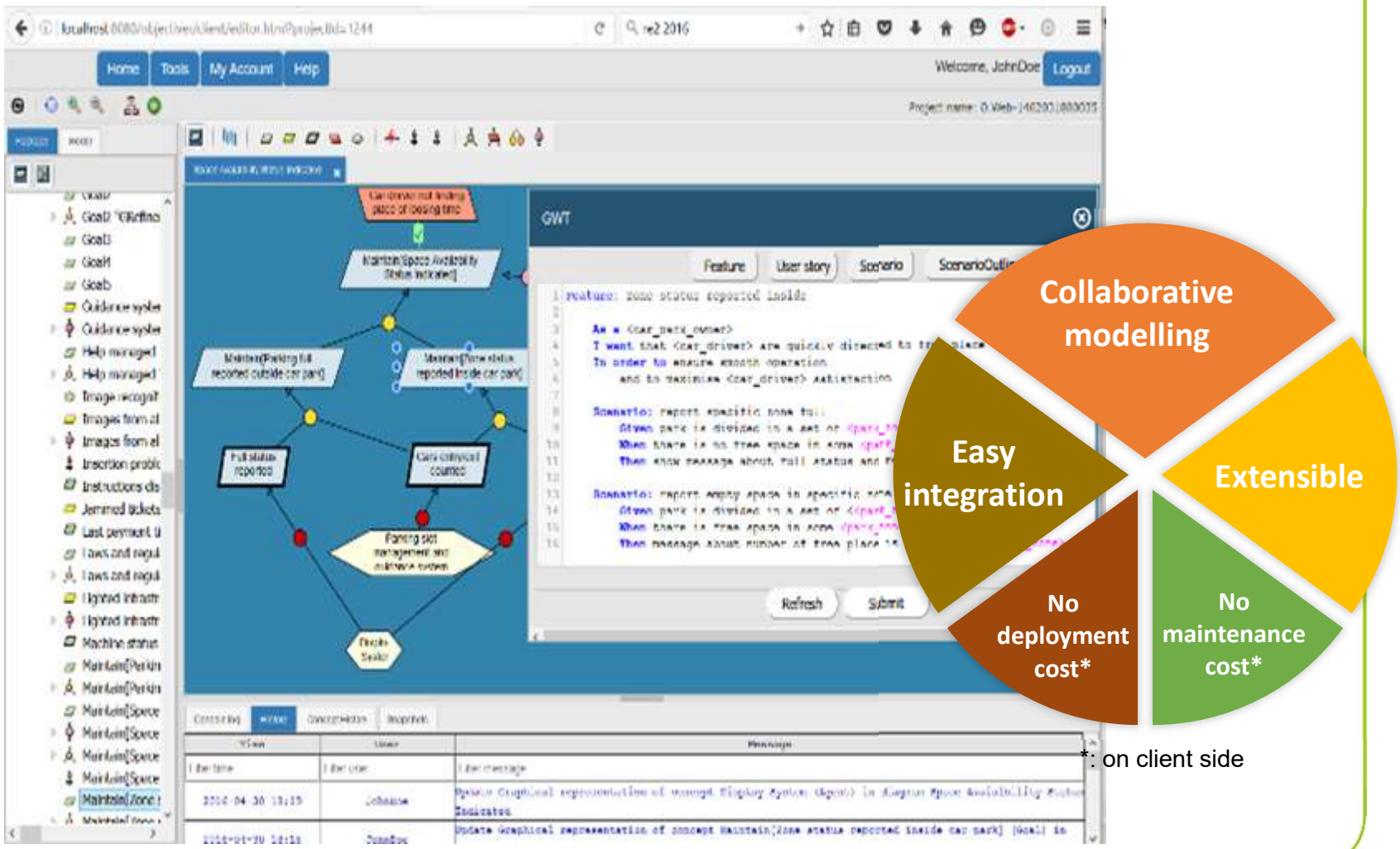
Table des matières

- Introduction
  - Profil & Historique
  - Credo
  - ODE
- OWEB
  - Architecture
  - Survol des fonctionnalités avec démo
  - Le coin des techies
  - RAF

Objectiver Web : architecture ouverte, robuste & évolutive



Objectiver Web :
Web editor utilisant le mode « **SaaS** »



*: on client side

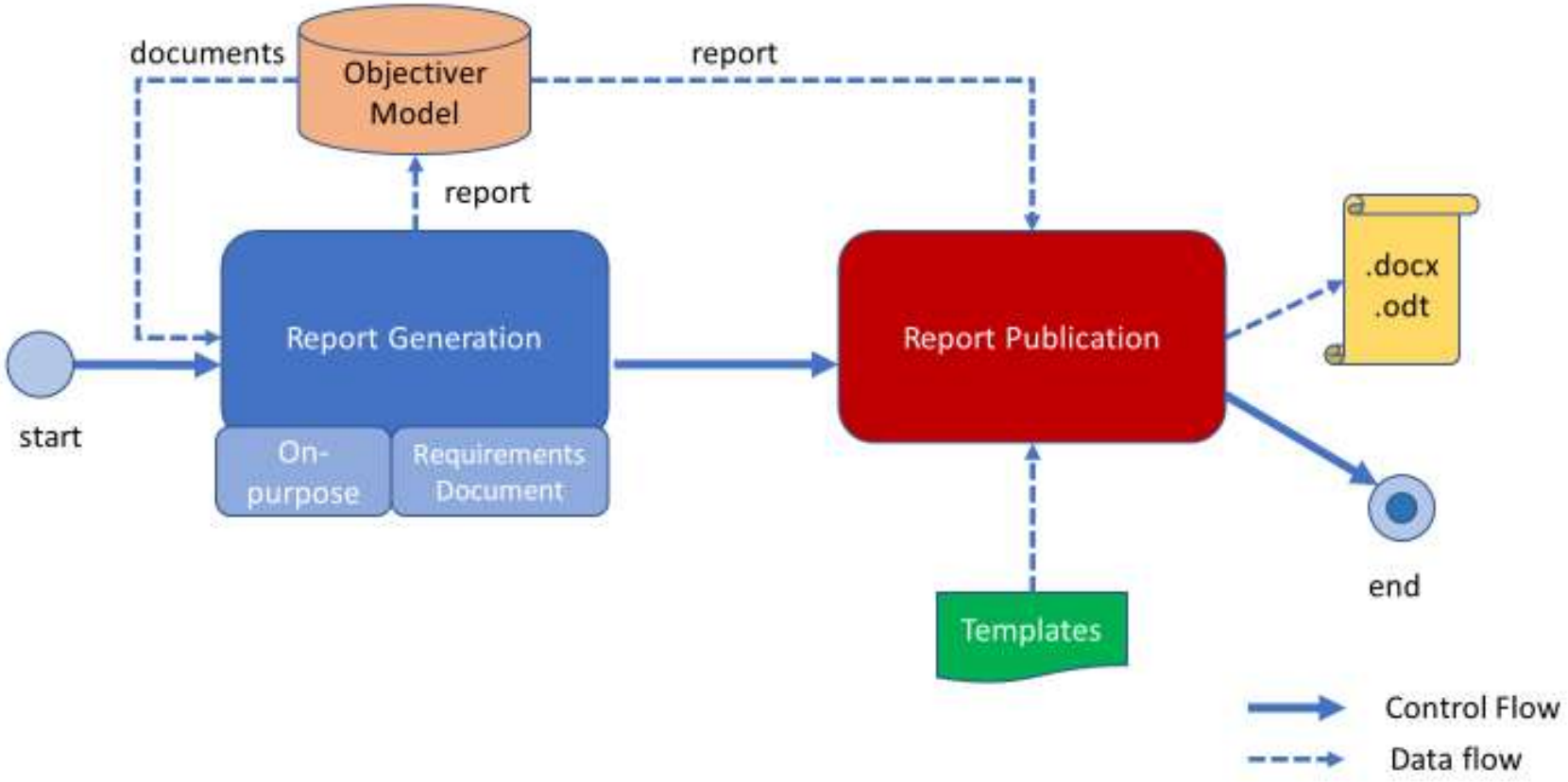
Table des matières

- Introduction
  - Profil & Historique
  - Credo
  - ODE
- OWEB
  - Architecture
  - Survol des fonctionnalités avec démo
  - Le coin des techies
  - RAF

Fonctionnalités de base

- Le + proche possible d'ODE avec quelques améliorations au passage :
  - Raffinement ET et raffinement OU **séparés**
  - *Refine into Diagram* a son inverse : ***Merge***
  - Editeur de **texte riche** pour les exigences et attentes
  - *Fold/Unfold*

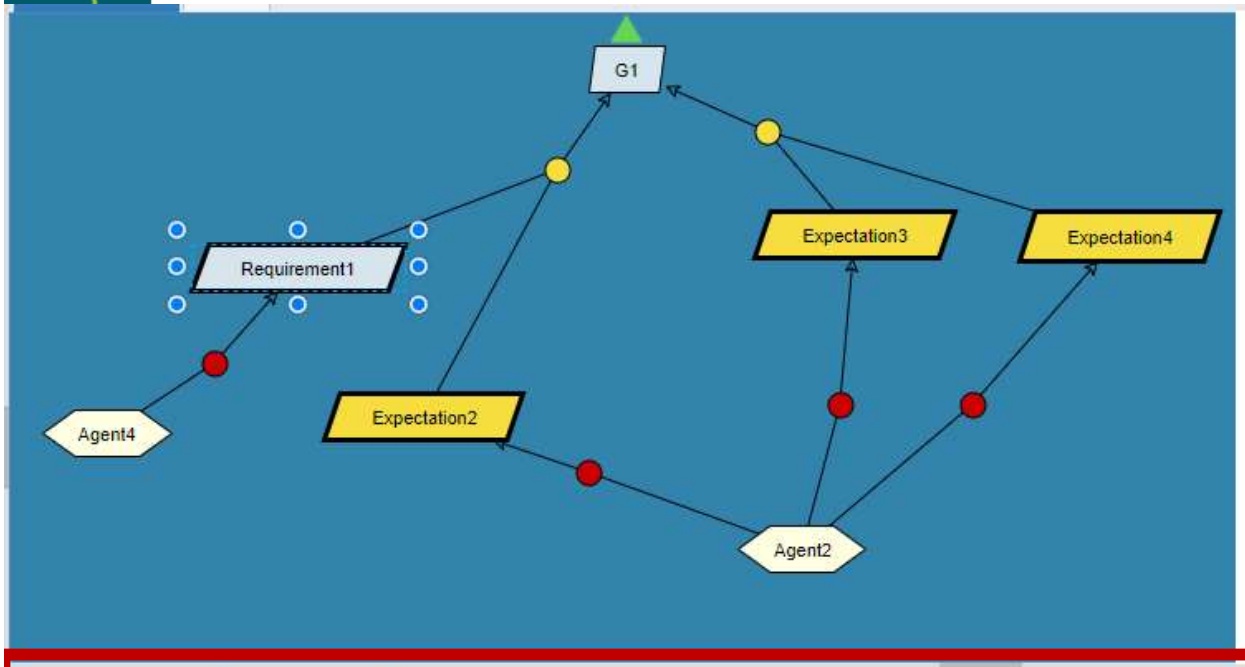
Reporting


documents
Objectiver Model
report
report
Report Generation
start
On-purpose
Requirements Document
Report Publication
.docx
.odt
end
Templates
Control Flow
Data flow

Nouveautés

- Historique des **modifications** d'un concept
- **Versioning** de diagrammes avec outil de comparaison
- Langage des requêtes : **OCL** et non plus OQL
- Model **dump** en Excel

Historique des modifications



classement chronologique inverse

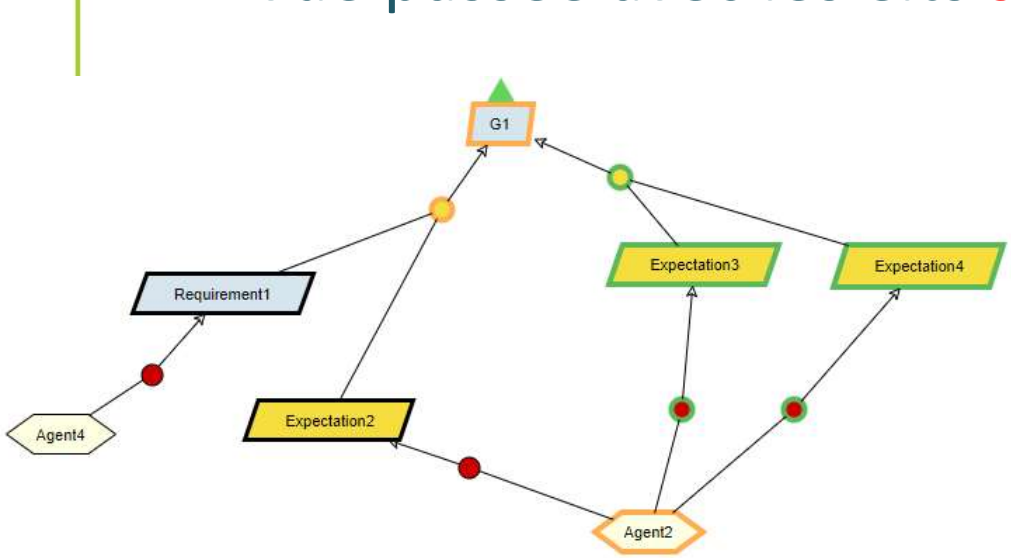
Console log | History | Concept History | Snapshots

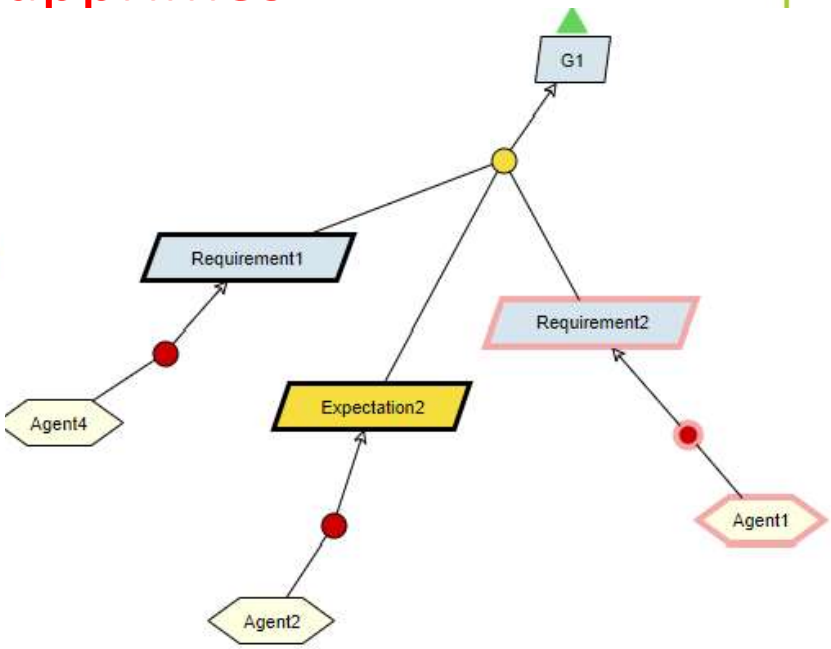
Time	User	Message
Filter time	Filter user	Filter message
2018-09-29 17:26	USER1	Create Graphical representation of concept Requirement1 "Responsibility" Agent4 (Responsibility) in diagram G1 (Diagram)
2018-09-29 17:26	USER1	Create relation Responsibility between Requirement1 and Agent4,
2018-09-29 17:24	USER1	Create relation GRefinement between G1 and Requirement1,
2018-09-29 17:24	USER1	Create Graphical representation of concept Requirement1 (Goal) in diagram D1
2018-09-29 17:24	USER1	Create Goal with name Requirement1

Permet de savoir qui a fait quoi

Outil de comparaison

- Deux vues complémentaires :
  - Vue actuelle avec les elts ajoutés et modifiés
  - Vue passée avec les elts supprimés





Langage des requêtes

- ODE/OQL : requête de type DB relationnelle :

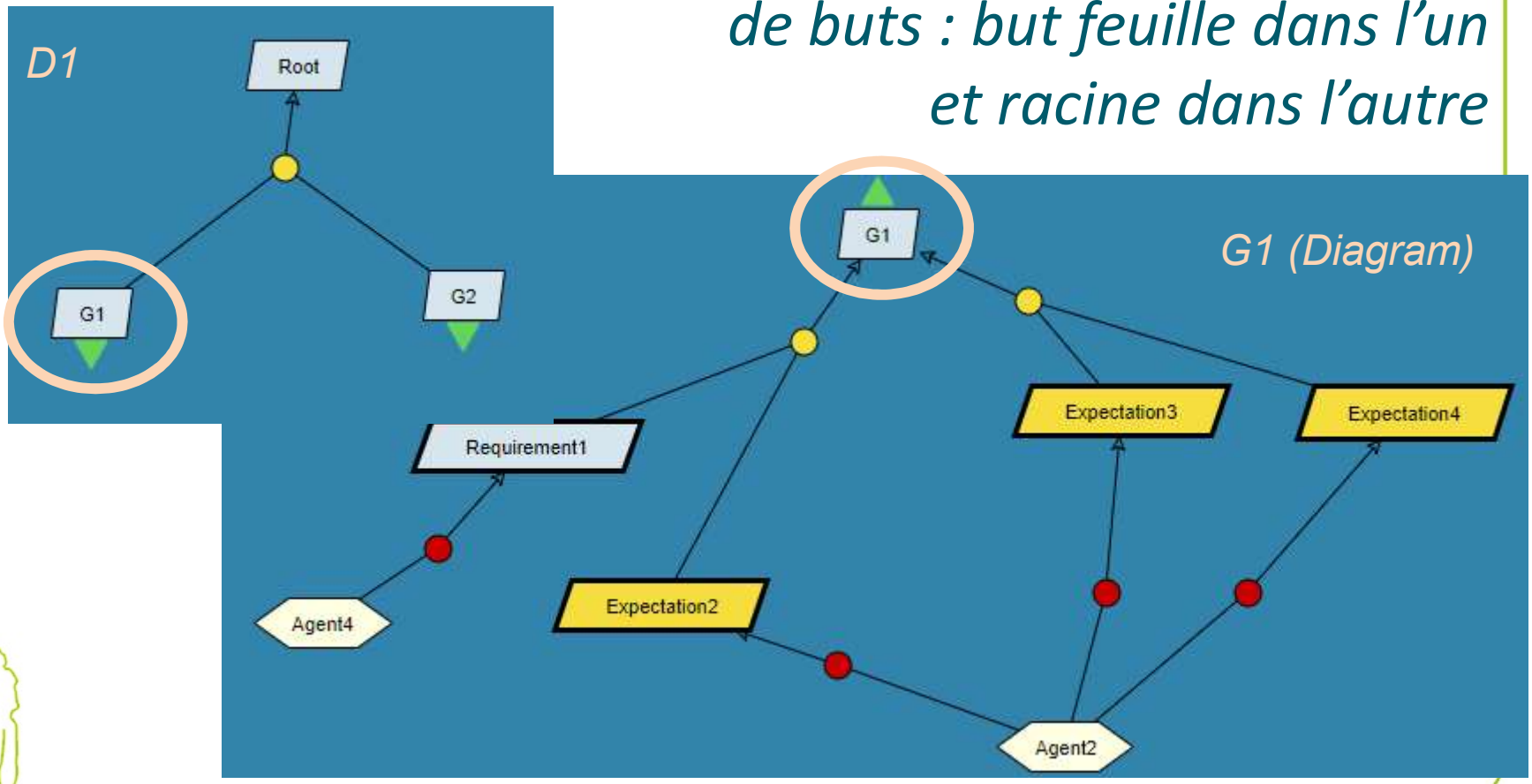
  SELECT…FROM…WHERE…

- OWEB/OCL :

  requête de type ensembliste + navigation

Exemple de requête

*Trouver les **points d'ancrage** entre 2 diagrammes de buts : but feuille dans l'un et racine dans l'autre*



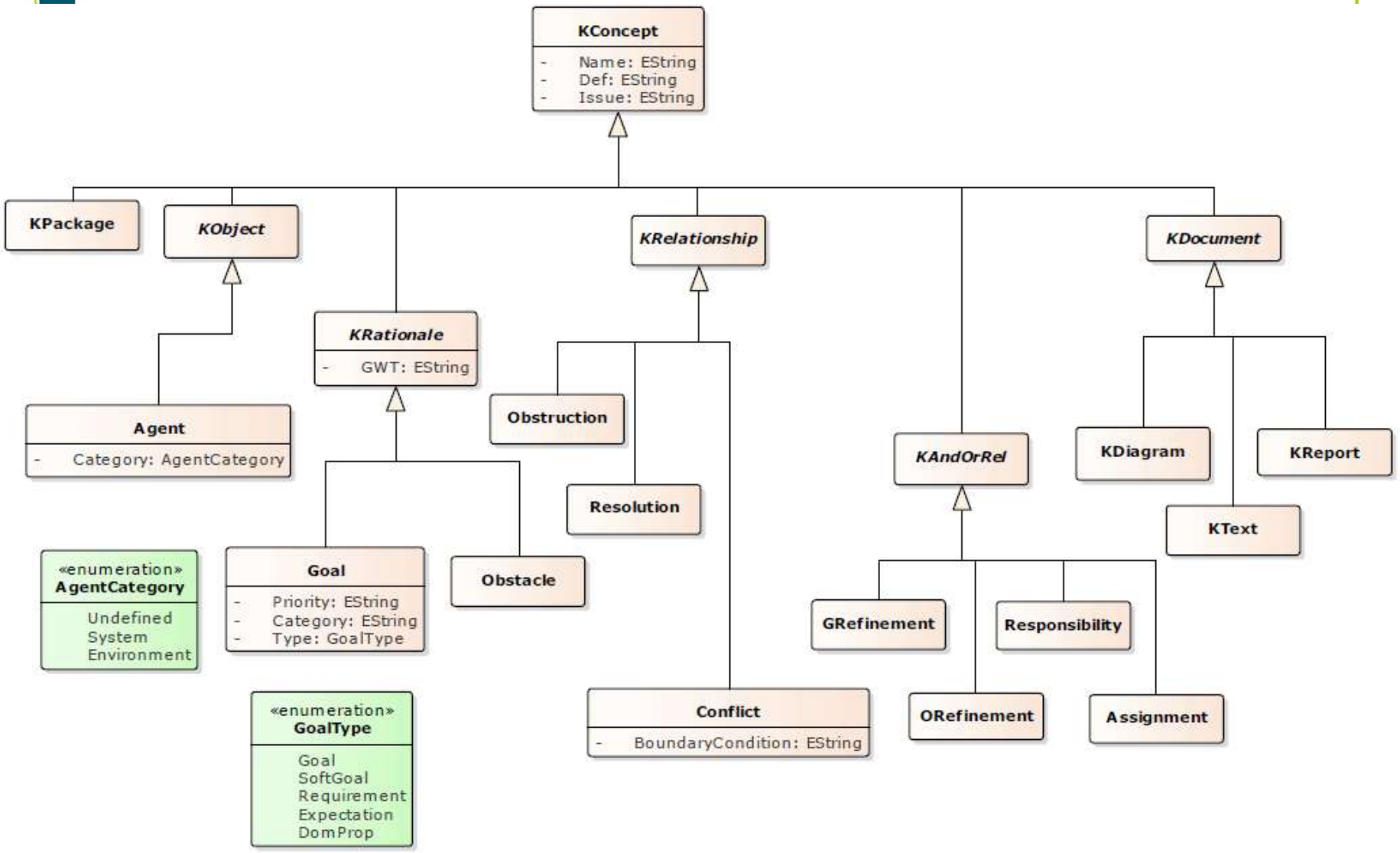
Exemple ODE/OQL

```
SELECT c
FROM Diagram AS d, d.elements AS e, e.concept AS c, c.isOf AS ct
WHERE ct.Name = "Goal" AND d.Name = " D1"
INTERSECT
SELECT c
FROM Diagram AS d, d.elements AS e, e.concept AS c, c.isOf AS ct
WHERE ct.Name = "Goal" AND d.Name = "G1 (diagram)"
EXCEPT
SELECT c
FROM Diagram AS d, d.elements AS e, e.concept AS g, g.isOf AS gt, Refinement AS r,
r.sons.son AS c, r.parent AS g2
WHERE d.Name = "G1 (diagram)" AND gt.Name = "Goal" AND g2.id = g.id
```

Exemple OWEB/OCL

```
KDiagram.allInstances()
    ->select (Name='D1')
    ->collect(containing.conceptOf)
    ->select(oclIsTypeOf(Goal))
    ->intersection(KDiagram.allInstances()
    ->select (Name='G1 (Diagram)')
    ->collect(containing.conceptOf)
    ->select(oclIsTypeOf(Goal))
    ->collect(oclAsType(Goal))
    ->reject(g:Goal |
       KDiagram.allInstances()
        ->select (Name='G1 (Diagram)')
        ->collect(containing.conceptOf)
        ->select(oclIsTypeOf(GRefinement) )
           ->collect(oclAsType(GRefinement).sons)
           ->includes (g)))
```

Méta-modèle OWEB



Méta-modèle OWEB

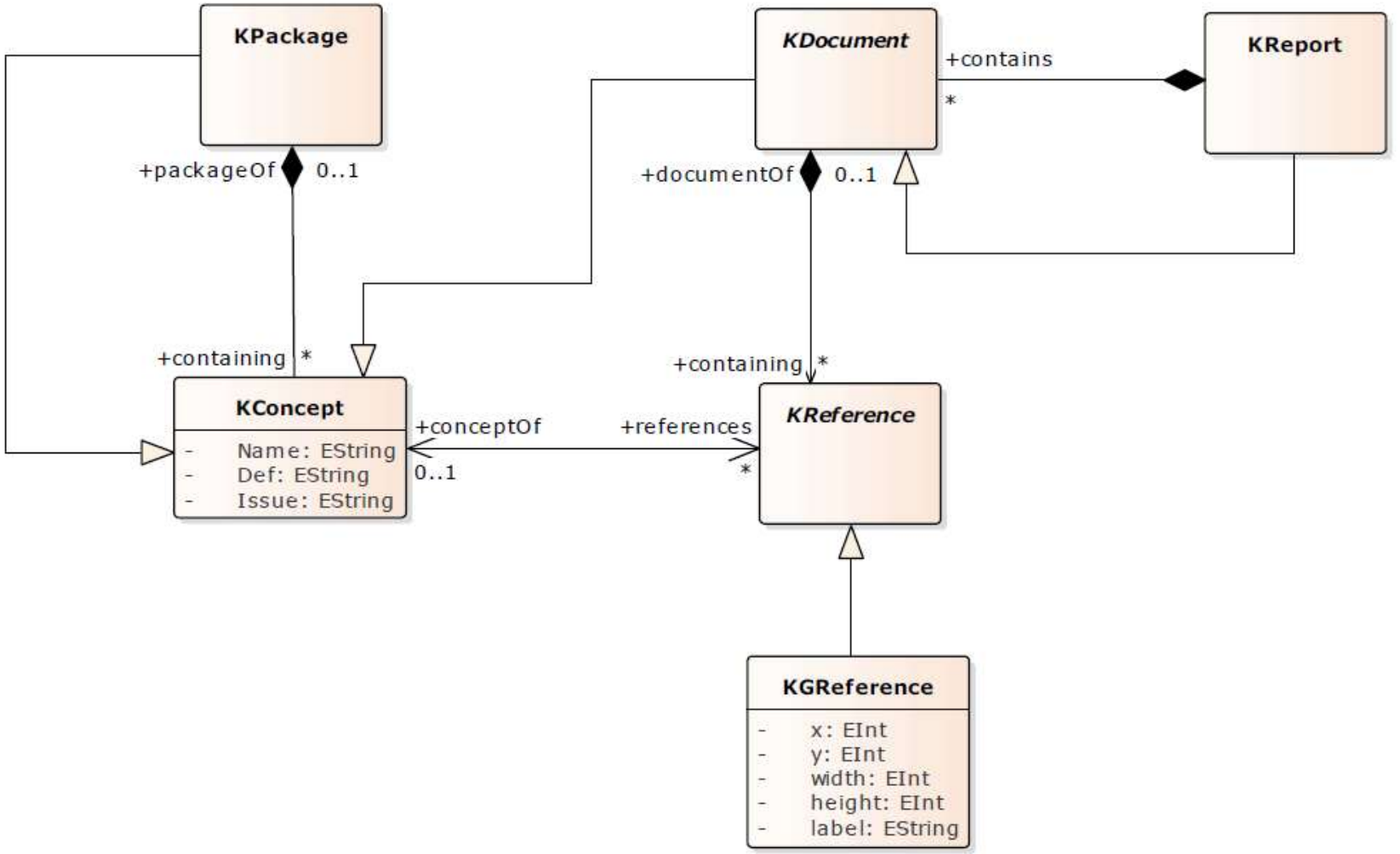


Table des matières

- Introduction
  - Profil & Historique
  - Credo
  - ODE
- OWEB
  - Architecture
  - Survol des fonctionnalités avec démo
  - Le coin des techies
  - RAF

Le coin des techies…

- **API REST** :
  - standard du Web (requête HTTP GET, POST, …)
  - pas d'état du client connu par le serveur
  - cache-able
- Format d'échange : **JSON** (Javascript Object Notation)
- Implémentation des services : **JAX-RS** (Java API for RESTful Web Services)

API REST (Project) + JAX-RS

2.11.7. Create project

```
POST /projects
```

Description

Create a new project on the repository

Parameters

Type	Name	Description	Required	Schema	Default
HeaderParameter	UserId		false	string	
BodyParameter	project	[ProjectSummary]	true	string	
HeaderParameter	Token	The access token that identify the user	true	string	

Responses

HTTP Code	Description	Schema
204	Project correctly created	No Content
400	Name already exists	No Content
403	Forbidden access	No Content

Consumes

- application/json

Produces

- application/json

3.21. Project

Name	Description	Required	Schema	Default
name		false	string	
id		false	integer (int64)	
allConcepts		false	KConcept array	
cdoresource		false	CDOResource	

```
@GET
@Path("/{projectId}")
@Produces(MediaType.APPLICATION_JSON)
/**
 * Get the details for the project identified by projectId
 * @param userId: the identifier of the user
 * @param projectId: the identifier of the project
 * @return HTTP 200 if success, HTTP 400 in error cases
 * Data response in success case @see objectiver.rest.api.json.Project
 */
 public Response getProject(
                        @HeaderParam("userId") String userId,
                        @PathParam("projectId") long projectId){
...
}
```

RAF

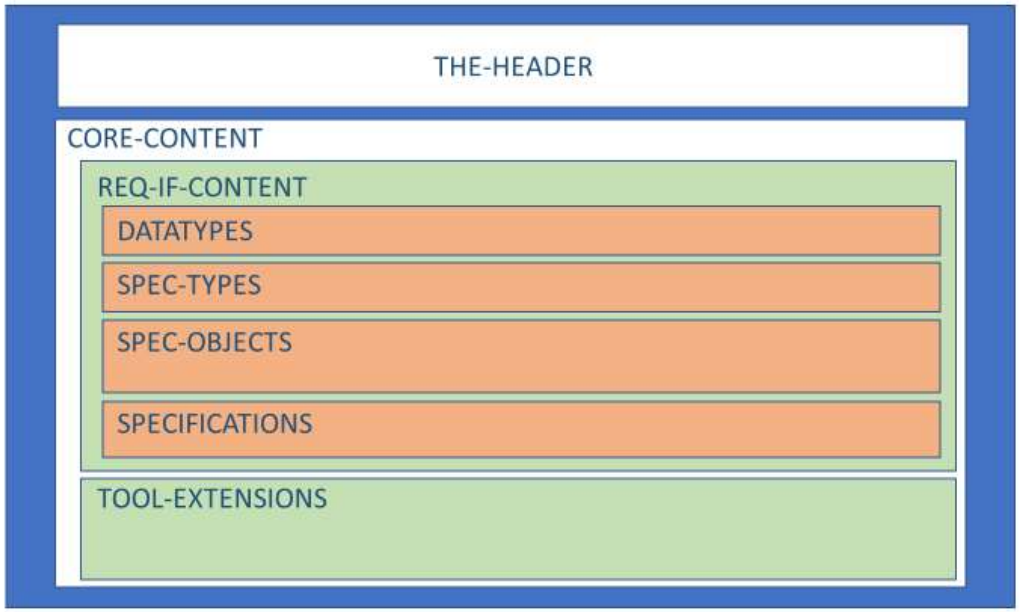
- Génération de CDC
  - amélioration des performances
  - algorithmes de génération
- Gestion des utilisateurs (groupes, …)
- Modèle Objet Objectiver

Conclusion

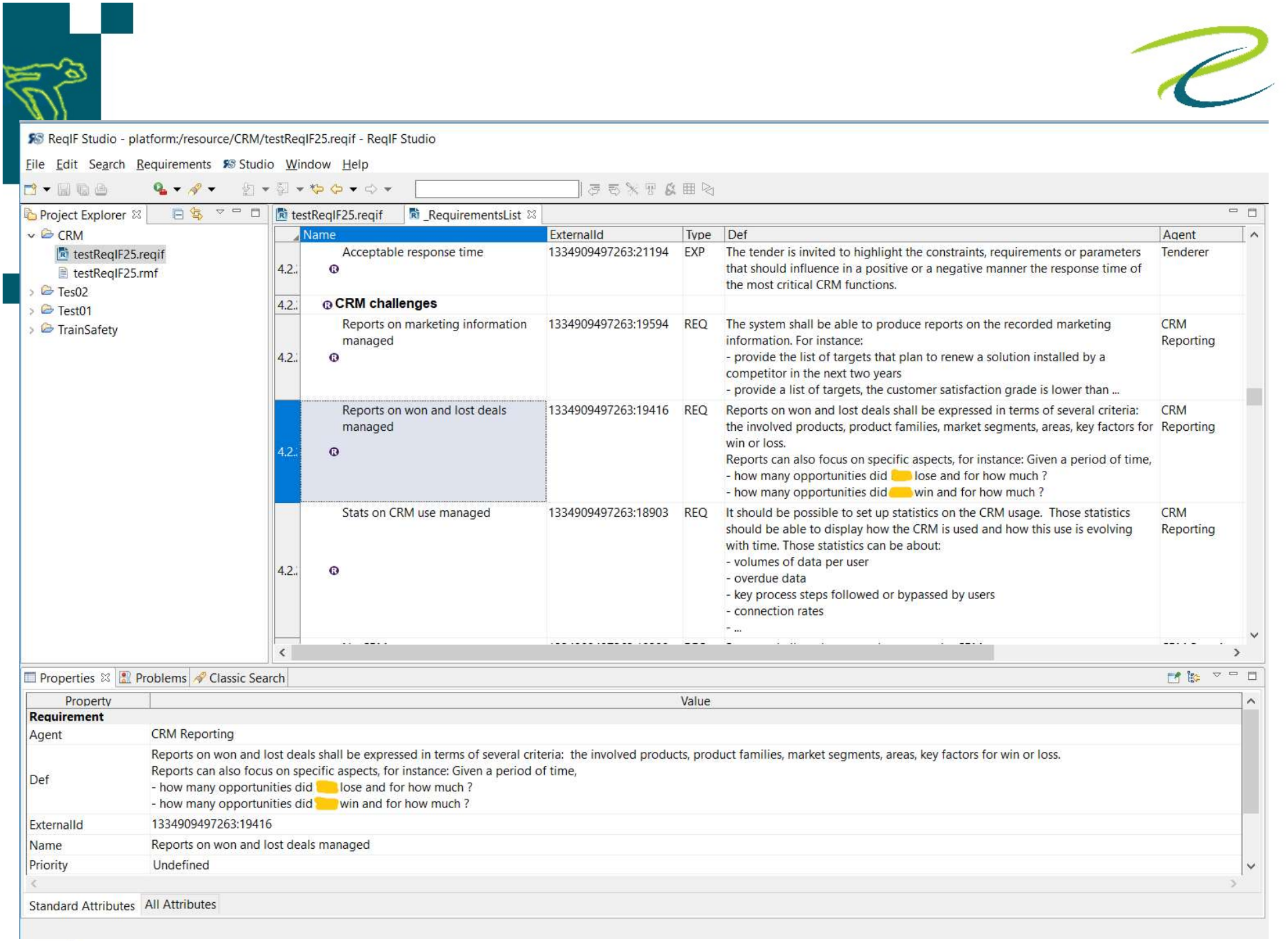
- Outil GORE répondant aux exigences actuelles
  - Architecture Cloud/Web
  - Ouverture
  - Simplicité
- **Valeur ajoutée** : support d'une méthode qui apporte un réel bénéfice dans la définition des exigences
- Déjà utilisé avec succès par une des plus grosses entreprises mondiales en télécom

Plugins récents : ReqIF

- Export des exigences et attentes au format ReqIF
- Standard pour IBM (Doors, …)



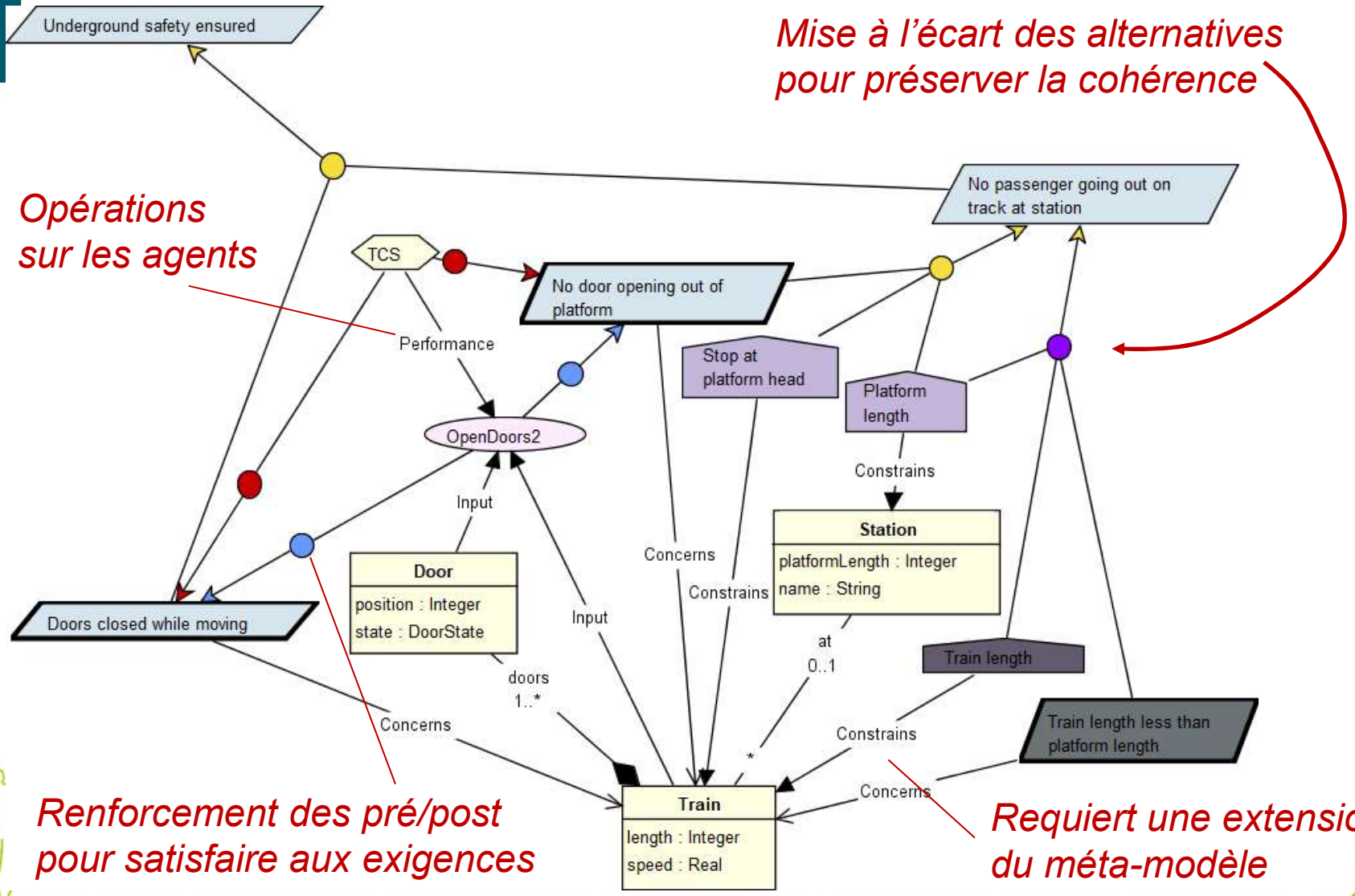
```
5619        <SPEC-OBJECT IDENTIFIER="PossibiliteDeRetoucherDesDossiersEnModeAdministrateur_131357453967235726" I
5620          <VALUES>
5621            <ATTRIBUTE-VALUE-STRING THE-VALUE="1313574539672:35726">
5622              <DEFINITION>
5623                <ATTRIBUTE-DEFINITION-STRING-REF>c0</ATTRIBUTE-DEFINITION-STRING-REF>
5624              </DEFINITION>
5625            </ATTRIBUTE-VALUE-STRING>
5626            <ATTRIBUTE-VALUE-STRING THE-VALUE="REQ">
5627              <DEFINITION>
5630            </ATTRIBUTE-VALUE-STRING>
5631            <ATTRIBUTE-VALUE-STRING THE-VALUE="Possibilité de retoucher des dossiers  en mode administrateu
5632              <DEFINITION>
5635            </ATTRIBUTE-VALUE-STRING>
5636            <ATTRIBUTE-VALUE-STRING THE-VALUE="1.   Une catégorie d&#8217;utilisateurs (administrateur systè
5637              <DEFINITION>
5640            </ATTRIBUTE-VALUE-STRING>
5641            <ATTRIBUTE-VALUE-STRING THE-VALUE="SIPAR2-Sys">
5642              <DEFINITION>
5645            </ATTRIBUTE-VALUE-STRING>
5646            <ATTRIBUTE-VALUE-ENUMERATION>
5647              <VALUES>
5648                <ENUM-VALUE-REF>_enumVal_Priority_Undefined</ENUM-VALUE-REF>
5649              </VALUES>
5650              <DEFINITION>
5651                <ATTRIBUTE-DEFINITION-ENUMERATION-REF>Priority</ATTRIBUTE-DEFINITION-ENUMERATION-REF>
5652              </DEFINITION>
5653            </ATTRIBUTE-VALUE-ENUMERATION>
5654          </VALUES>
5655          <TYPE>
5656            <SPEC-OBJECT-TYPE-REF>_T_Requirement</SPEC-OBJECT-TYPE-REF>
5657          </TYPE>
5658        </SPEC-OBJECT>
5659        <SPEC-OBJECT IDENTIFIER="PossibiliteDeReactiverUnDossierAVEnCasDinterventionNouvelle_13135745396723
5660          <VALUES>
```


ReqIF Studio - platform:/resource/CRM/testReqIF25.reqif - ReqIF Studio
File Edit Search Requirements Studio Window Help
Project Explorer
CRM
testReqIF25.reqif
testReqIF25.rmf
Tes02
Test01
TrainSafety
testReqIF25.reqif
_RequirementsList
Name
ExternalId
Type
Def
Agent
4.2.
Acceptable response time
1334909497263:21194
EXP
The tender is invited to highlight the constraints, requirements or parameters that should influence in a positive or a negative manner the response time of the most critical CRM functions.
Tenderer
4.2.
CRM challenges
4.2.
Reports on marketing information managed
1334909497263:19594
REQ
The system shall be able to produce reports on the recorded marketing information. For instance:
- provide the list of targets that plan to renew a solution installed by a competitor in the next two years
- provide a list of targets, the customer satisfaction grade is lower than ...
CRM Reporting
4.2.
Reports on won and lost deals managed
1334909497263:19416
REQ
Reports on won and lost deals shall be expressed in terms of several criteria: the involved products, product families, market segments, areas, key factors for win or loss.
Reports can also focus on specific aspects, for instance: Given a period of time,
- how many opportunities did lose and for how much ?
- how many opportunities did win and for how much ?
CRM Reporting
4.2.
Stats on CRM use managed
1334909497263:18903
REQ
It should be possible to set up statistics on the CRM usage. Those statistics should be able to display how the CRM is used and how this use is evolving with time. Those statistics can be about:
- volumes of data per user
- overdue data
- key process steps followed or bypassed by users
- connection rates
- ...
CRM Reporting
Properties
Problems
Classic Search
Property
Value
Requirement
Agent
CRM Reporting
Def
Reports on won and lost deals shall be expressed in terms of several criteria: the involved products, product families, market segments, areas, key factors for win or loss.
Reports can also focus on specific aspects, for instance: Given a period of time,
- how many opportunities did lose and for how much ?
- how many opportunities did win and for how much ?
ExternalId
1334909497263:19416
Name
Reports on won and lost deals managed
Priority
Undefined
Standard Attributes
All Attributes

Plugin USE

Mise à l'écart des alternatives pour préserver la cohérence

Opérations sur les agents



Renforcement des pré/post pour satisfaire aux exigences

Requiert une extension du méta-modèle

```
context Train
    -- Train doors shall be kept closed when the train is moving.
  inv DoorsClosedWhileMoving: (self.speed <> 0) implies self.doors->forAll (closed)

context Train
    -- Doors located in a part of the train exceeding the platform length shall be
maintained closed when the train is stopped in a station.
  inv: self.at.isDefined() and self.doors->exists (not closed) implies self.doors->forAll
(d:Door | d.closed implies d.position > self.at.platformLength)

class TrainController
operations
OpenDoors2 (tr: Train, selectedDoors: Set(Door))
      pre AllClosed: tr.doors->forAll (closed)
      pre Consistency: tr.doors->includesAll(selectedDoors)
      pre ZeroSpeed: selectedDoors->notEmpty implies tr.speed = 0
      pre OutOfPlatform:
      tr.doors->forAll(d:Door | not selectedDoors->includes (d) implies d.position >
tr.at.platformLength)
     post: tr.doors->forAll(d:Door | d.closed implies not selectedDoors->includes (d))
end
```