Drools Rule Engine Overview and Real Example

Danilo Ercoli

Spring Day 2008, 14 giugno Cagliari

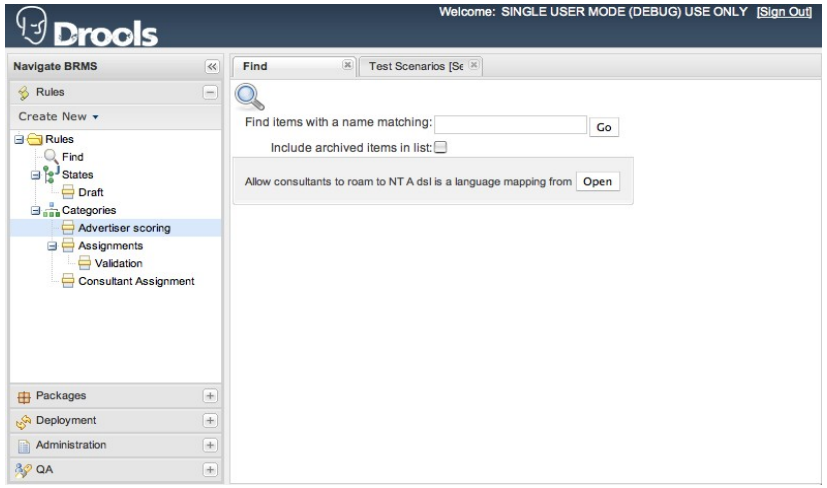
www.**exmachina**.ch
in <code> we trust

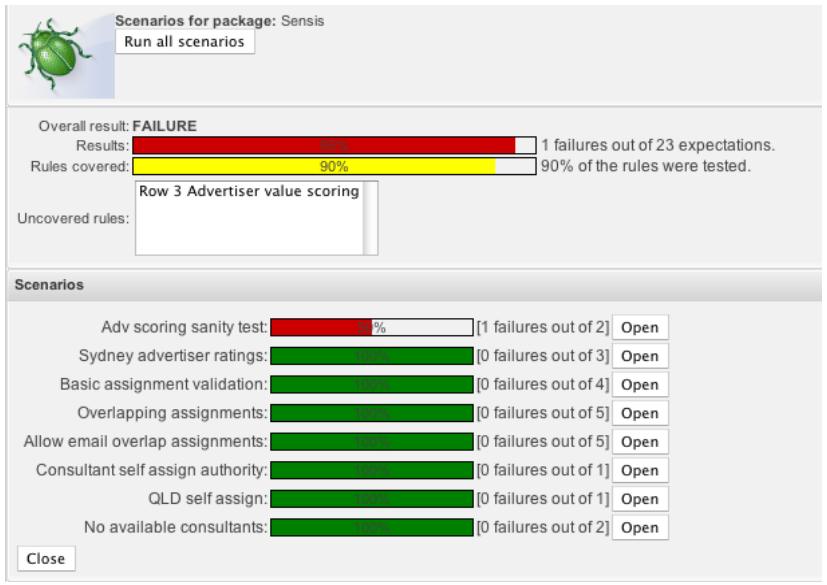
Agenda

- What is a Drools ?
  - Drools History
  - Drools & Rules Engine
  - Rete Algorithm
  - Agenda Conflict Resolution
  - When to Use a Rule Engine
  - When Not to Use a Rule Engine
  - Who Should Use a Rule Engine
  - Authoring and Runtime
- A Real Example
  - Our Business Problem
  - Example of functionality
  - How we have express a Rules
  - Domain Specific Language
  - Spring and Drools

What is Drools

- Drools is a business rule management system (BRMS) with a inference based rules engine, using an enhanced implementation of the Rete algorithm.
- Drools is an Open Source project, written by Bob McWhirter
- A complete implementation of the JSR94 Rule Engine API
- Eclipse IDE, plugin makes it easier than ever to use Drools



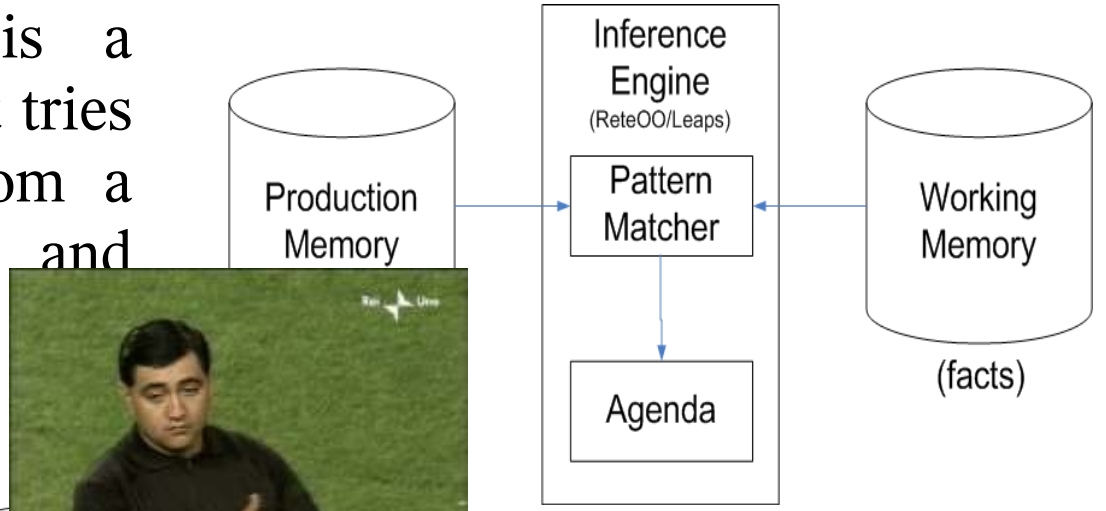


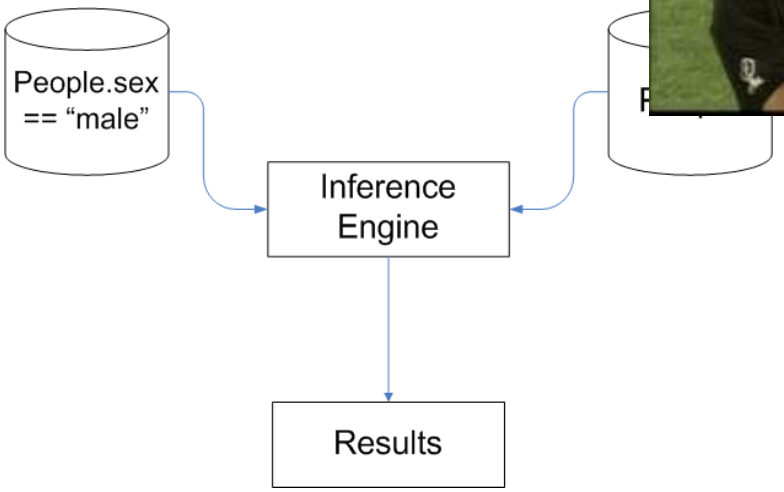
Drools History

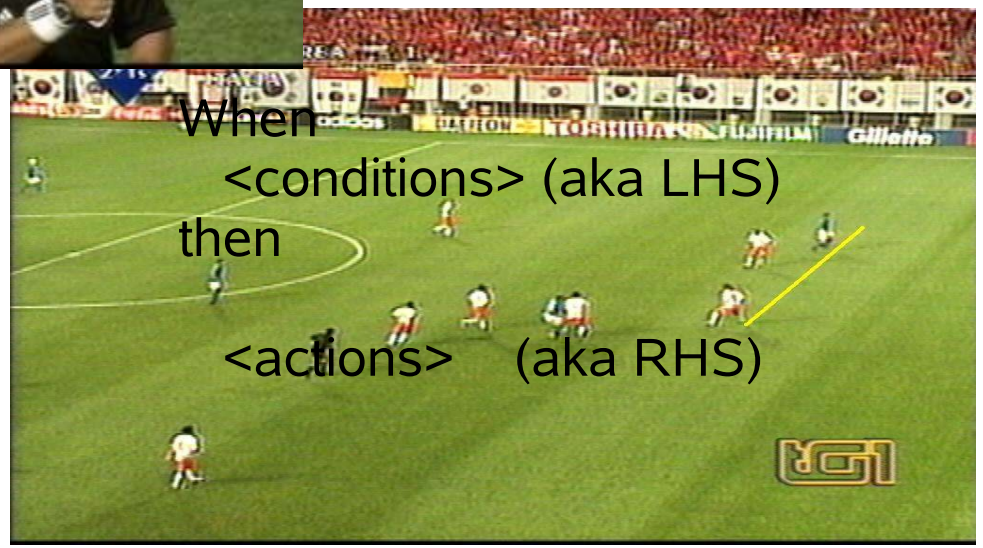
- 2001: Started by Bob McWhirter (Drools 1.0 never released)
- Mark Proctor became the project leader during the 2.0 development cycle (at Codehaus) and moved the project to a final 2.0 release
- 2005: Drools was incorporated into JBoss Enterprise Middleware System (JEMS)
- 2006: JBoss was acquired by Red Hat
- 2007: JBoss Rules 4.0 is a rewrite with a full and enhanced Rete implementation with GUI tooling.
- 2008: Drools-4.0.4, was released Jan. 15th, 2008

Drools & Rules Engine

- **Inference Engine:** is a computer program that tries to derive answers from a knowledge base(rules and facts)



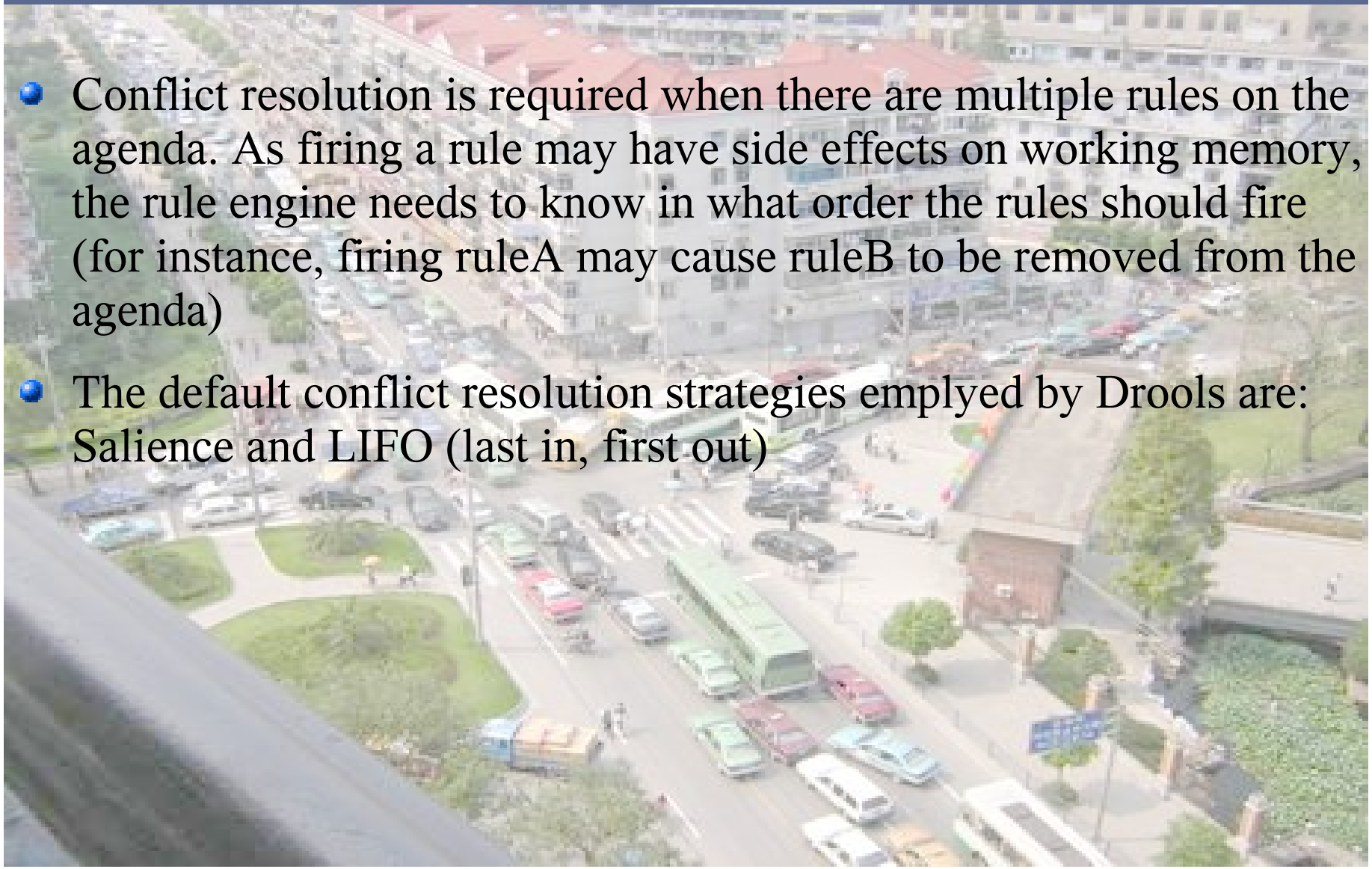




Rete Algorithm

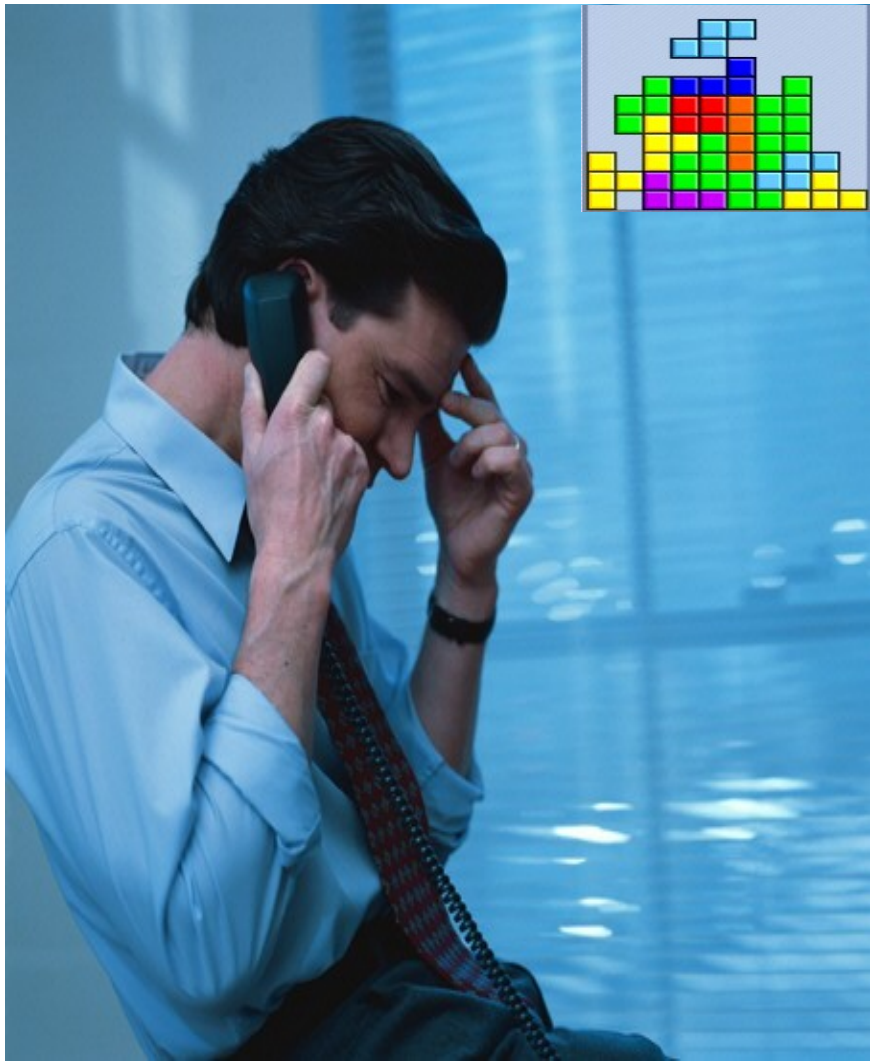
- The Rete algorithm is an efficient pattern matching algorithm for implementing production rule systems
- The Rete algorithm is designed to sacrifice memory for increased speed
- The Rete algorithm exhibits the following important characteristics:
  - It reduces or eliminates certain types of redundancy through the use of node sharing
  - It stores partial matches when performing joins between different fact types.
  - It allows for efficient removal of memory elements when facts are retracted from working memory.

Agenda: conflict resolution



- Conflict resolution is required when there are multiple rules on the agenda. As firing a rule may have side effects on working memory, the rule engine needs to know in what order the rules should fire (for instance, firing ruleA may cause ruleB to be removed from the agenda)
- The default conflict resolution strategies emplyed by Drools are: Salience and LIFO (last in, first out)

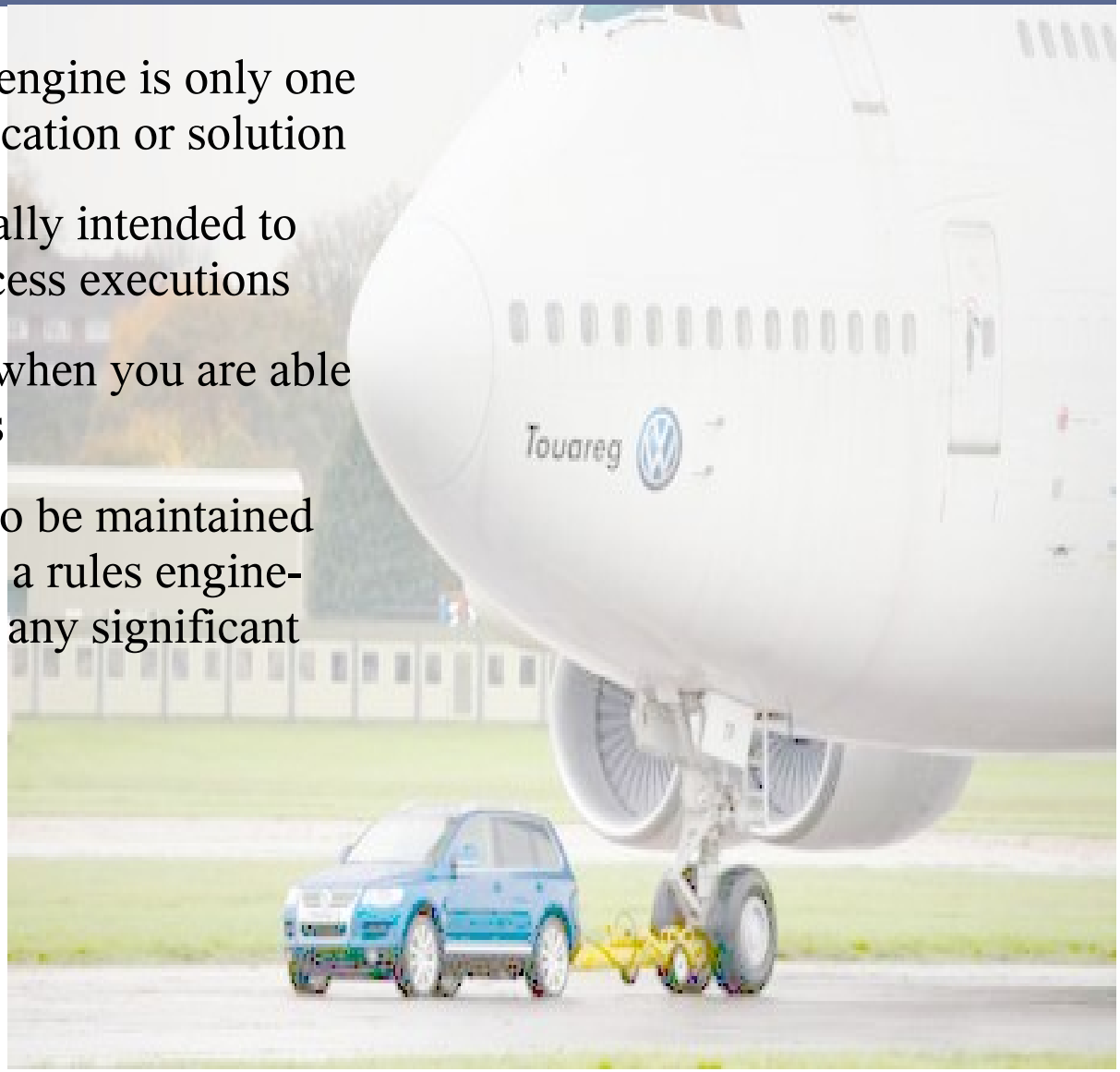
When to use a Rule Engine



- The logic changes often
- Domain experts (or business analysts) are readily available, but are nontechnical
- The problem is beyond any obvious algorithm based solution
- The problem may not be complex, but you can't see a non-fragile way of building it

When not to use a Rule Engine

- Don't forget that a rules engine is only one piece of a complex application or solution
- Rules engines are not really intended to handle workflow or process executions
- Rule engines work best when you are able to write declarative rules
- If the code is not going to be maintained over time, then don't use a rules engine- you probably won't gain any significant advantage from it.

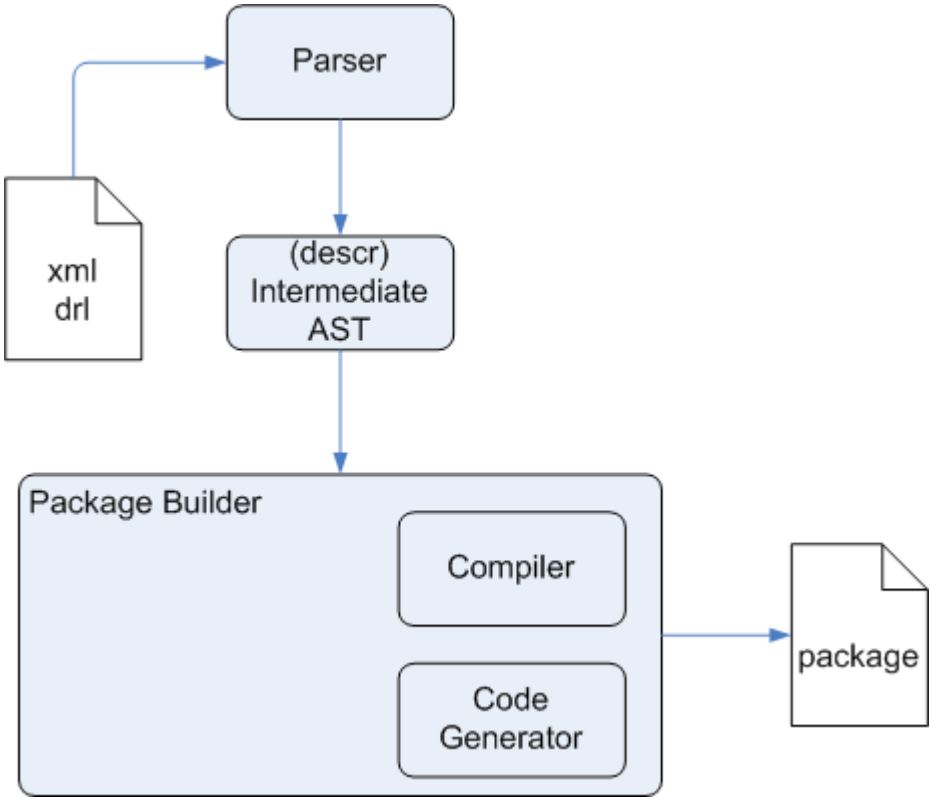


Who to use a Rule Engine

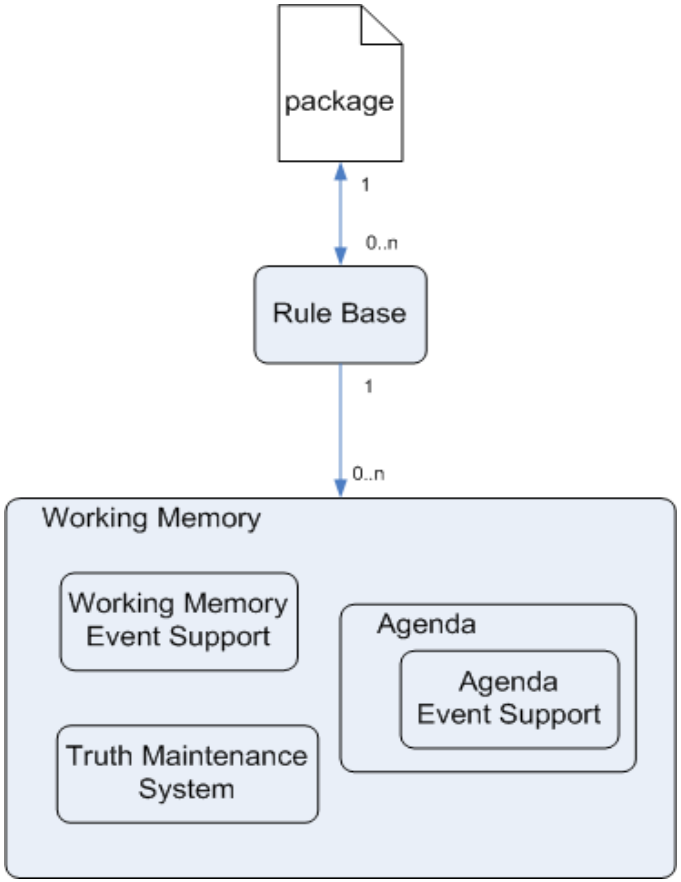
- Insurance
  - Rating
  - Automated underwriting
  - Claims routing and management
  - Suitability/Compliance
- Financial Services
  - Loan origination
  - Pro-forma trading models
  - Fraud detection
  - Asset management
- Ecommerce
  - Cross selling
  - Sales configuration
  - Content and access security
- Government
  - Tax calculations
  - Fee calculations
  - Application processing
- Manufacturing
  - Supply chain management
  - Product configuration

Authoring and runtime

Package building



Package consummation

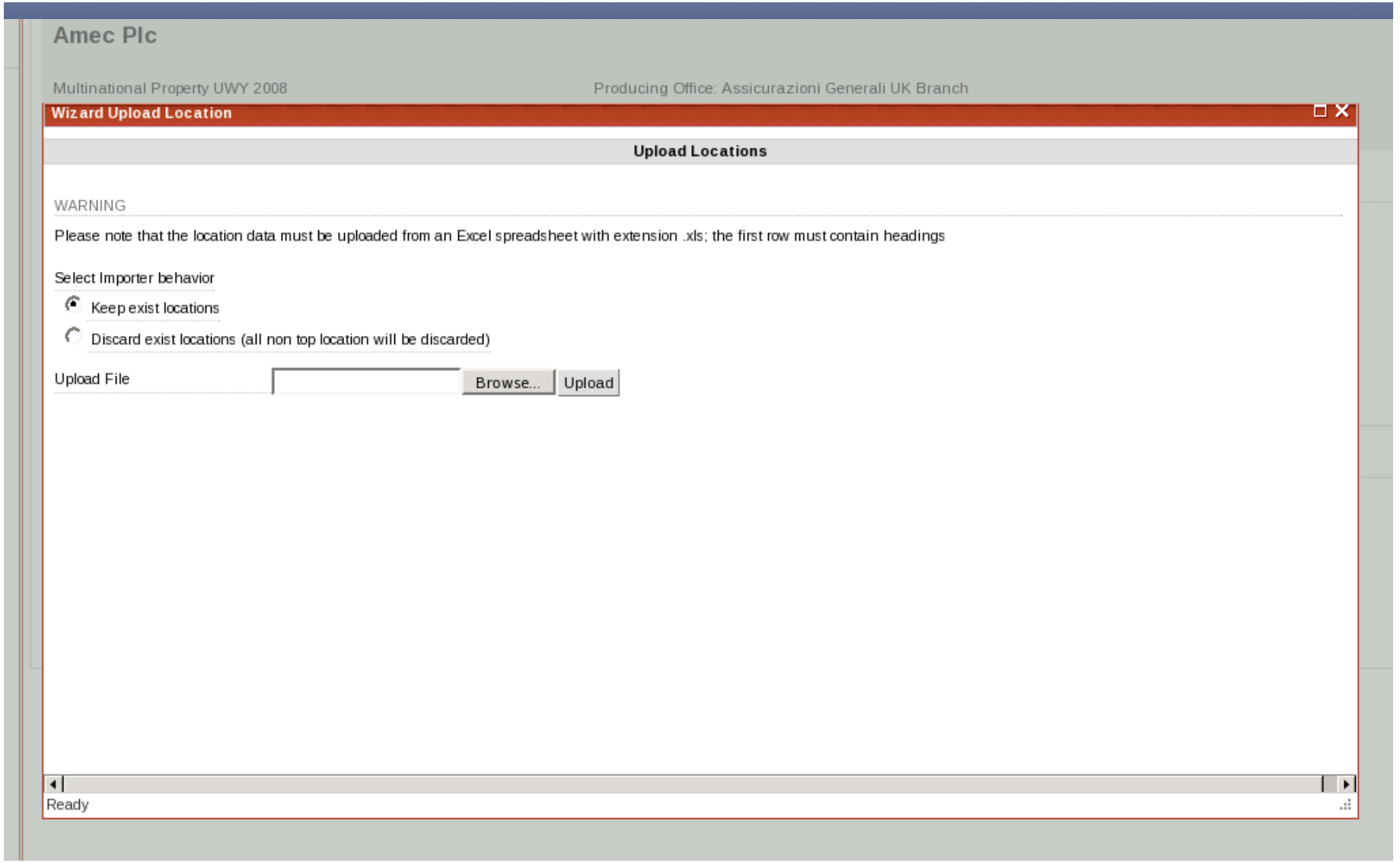


A Real Example: Business Requirement

- Acquire from a spreadsheet a massive location upload, every location has about 60 information, many value to be verified and validate before accept into system(risk management system)
- We have decided to use Drools for validation of the single row of spreadsheet

Inception Date	Expiration Date	Local Policy Number	UWY	Sequence	Country Name	Country ISO Code	State/Province Name	State/Province Code info	Town Name
25/03/2008	24/03/2009		2007		Brazil	BR			Sao Paulo
25/03/2008	24/03/2009		2007		Australia	AU			Moorebank
25/03/2008	24/03/2009		2007		Austria	AT			Höchst
25/03/2008	24/03/2009		2007		France	FR			Alby sur Ch
25/03/2008	24/03/2009		2007		China	CN			Guangzhou
25/03/2008	24/03/2009		2007		China	CN			Shanghai

Examples of Functionality 1 of 9

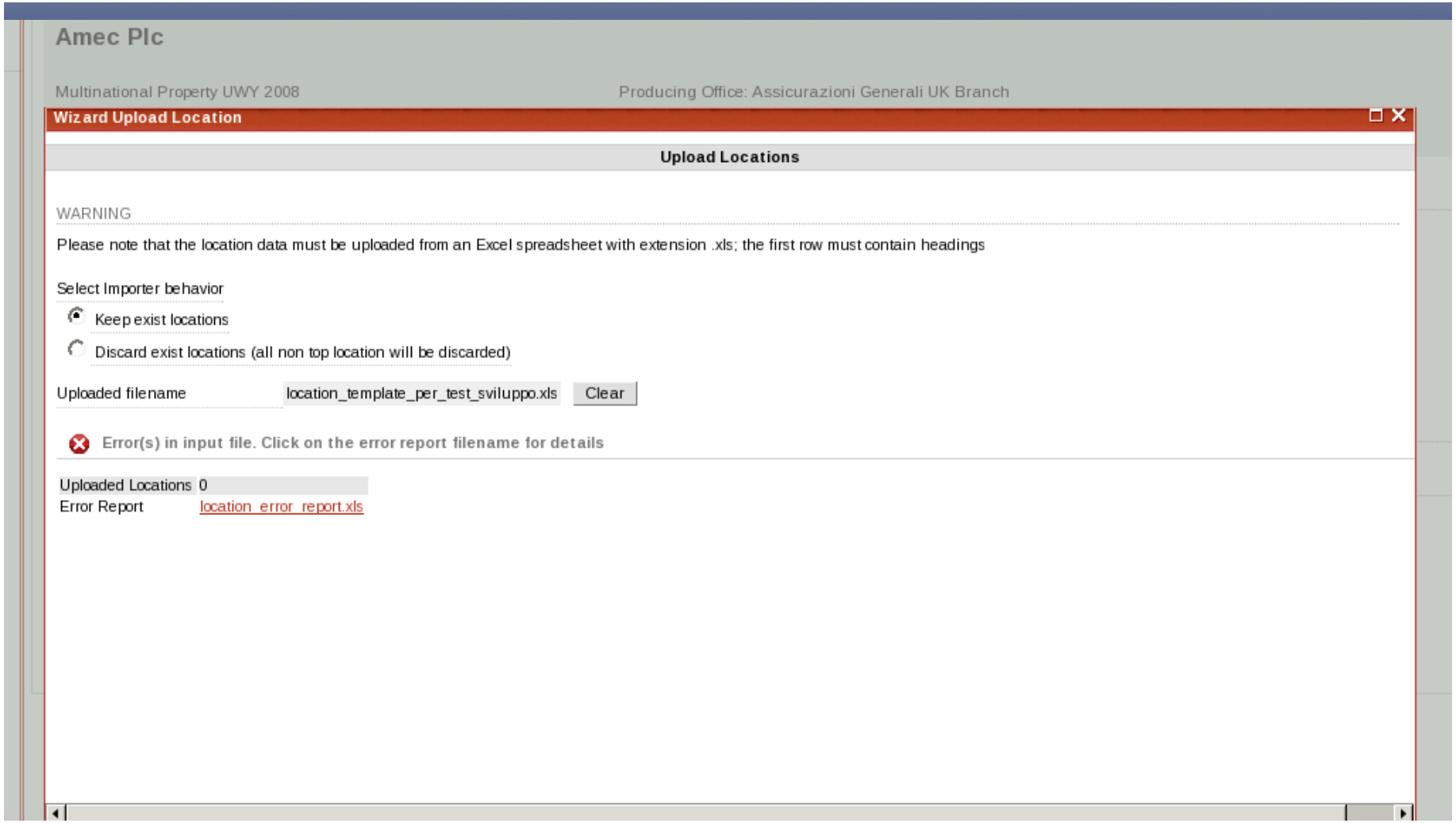


Examples of Functionality 2 of 9

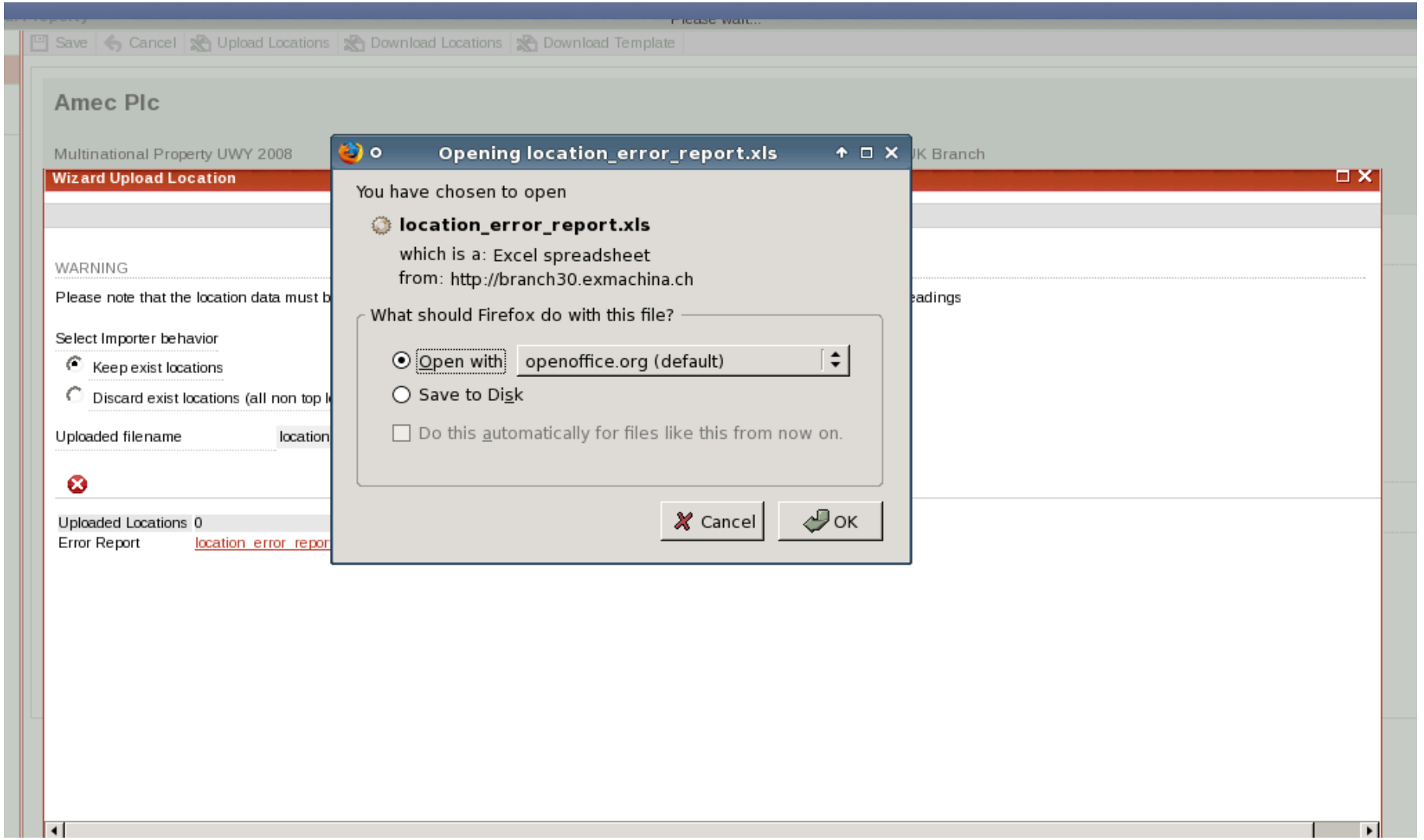
A2 | 06/10/2008

	A	B	C	D	E	F	G	H
1	Inception Date	Expiration Date	Local Policy Number	UWY	Sequence	Country Name	Country ISO Code	State/Province Na
2	10. Jun. 2008	11. Jun. 2007	DM03362	2006	0	ARGENTINA	AR	
3	1. Jun. 2008	30. May. 2009	DM03362	2006	0	ARGENTINA	AR	
4	1. Jun. 2008	30. May. 2009	DM03362	2006	0	ARGENTINA	AR	
5	1. Jun. 2008	30. May. 2009	DM03362	2006	0	ARGENTINA	AR	
6	1. Jun. 2008	30. May. 2009	DM03362	2006	0	ARGENTINA	AR	
7	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
8	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
9	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
10	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
11	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
12	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
13	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
14	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
15	1. Jun. 2008	30. May. 2009	DM03362	2006	test1	ARGENTINA	AR	
16	1. Jun. 2008	30. May. 2009	DM03362	2006	test2	ARGENTINA	AR	
17	1. Jun. 2008	30. May. 2009	DM03362	2006	test3	ARGENTINA	AR	
18	1. Jun. 2008	30. May. 2009	DM03362	2006	test0	ITALY	IT	
19	1. Jun. 2008	30. May. 2009	DM03362	2006	test4	AUSTRALIA	AU	
20								
21								
22								
23								
24								
25								
26								

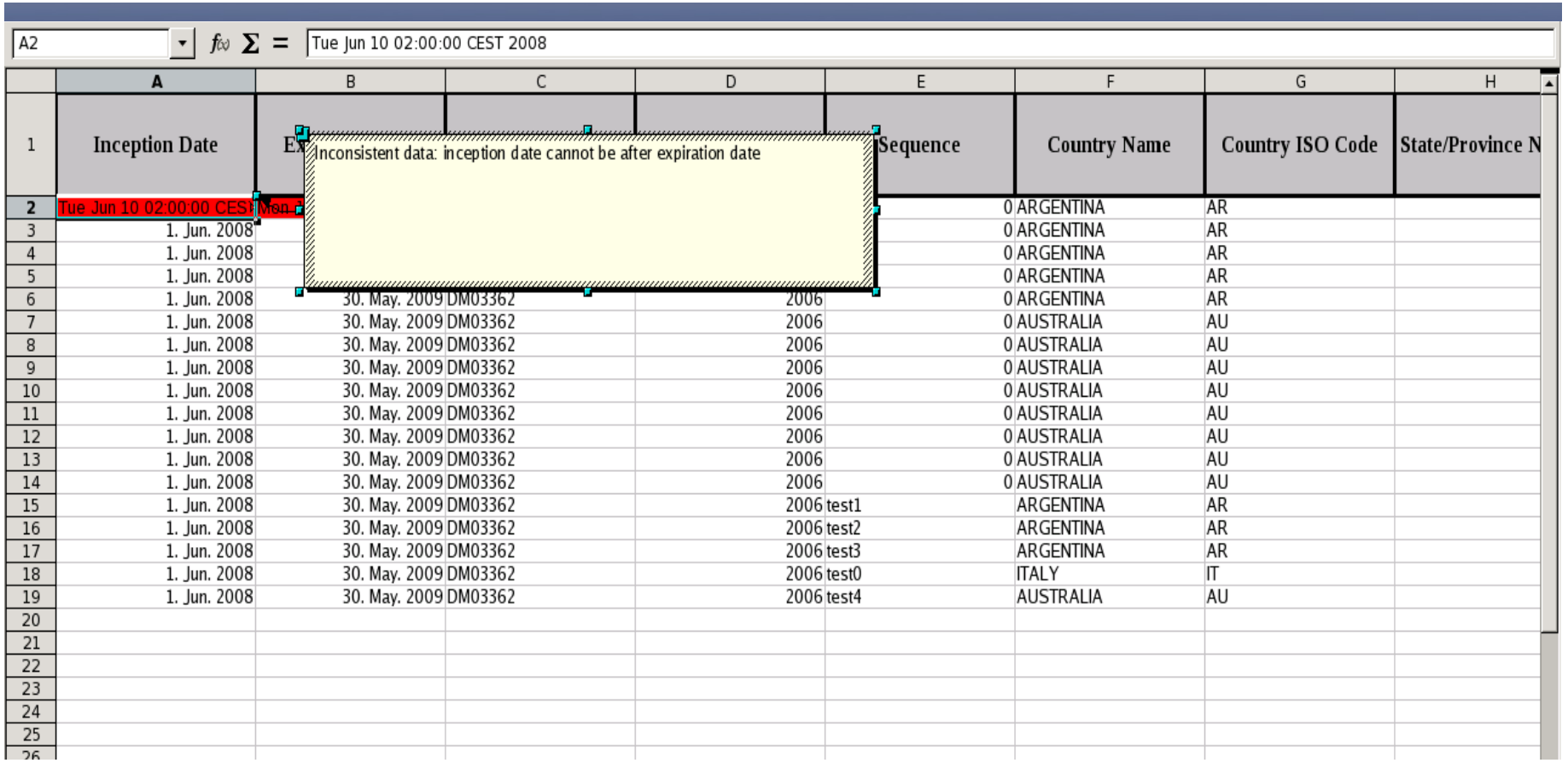
Examples of Functionality 3 of 9



Examples of Functionality 4 of 9



Examples of Functionality 5 of 9



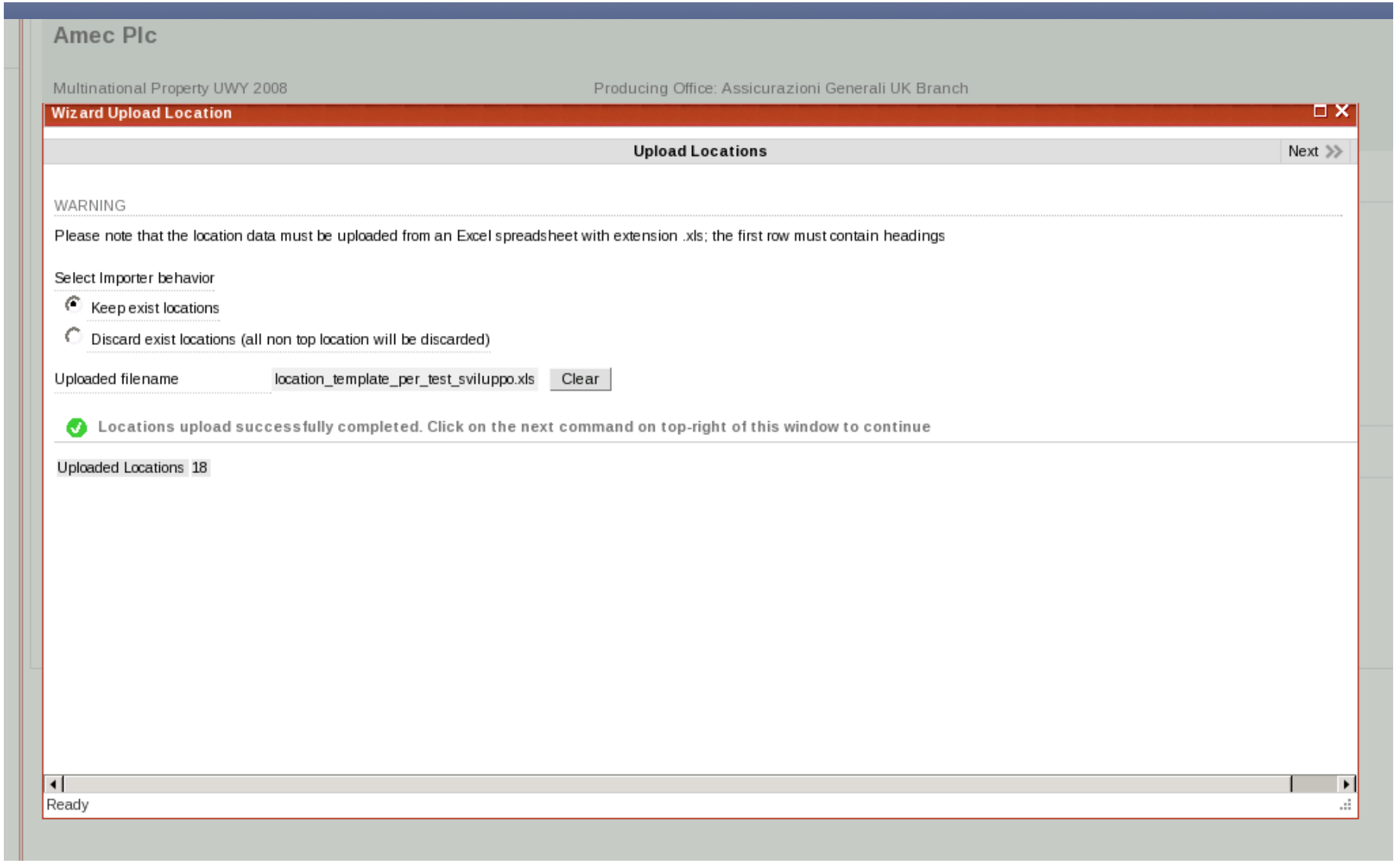
	A	B	C	D	E	F	G	H
1	Inception Date	EX			Sequence	Country Name	Country ISO Code	State/Province N
2	Tue Jun 10 02:00:00 CEST Mon				0	ARGENTINA	AR	
3	1. Jun. 2008				0	ARGENTINA	AR	
4	1. Jun. 2008				0	ARGENTINA	AR	
5	1. Jun. 2008				0	ARGENTINA	AR	
6	1. Jun. 2008	30. May. 2009	DM03362	2006	0	ARGENTINA	AR	
7	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
8	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
9	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
10	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
11	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
12	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
13	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
14	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
15	1. Jun. 2008	30. May. 2009	DM03362	2006	test1	ARGENTINA	AR	
16	1. Jun. 2008	30. May. 2009	DM03362	2006	test2	ARGENTINA	AR	
17	1. Jun. 2008	30. May. 2009	DM03362	2006	test3	ARGENTINA	AR	
18	1. Jun. 2008	30. May. 2009	DM03362	2006	test0	ITALY	IT	
19	1. Jun. 2008	30. May. 2009	DM03362	2006	test4	AUSTRALIA	AU	
20								
21								
22								
23								
24								
25								

Examples of Functionality 6 of 9

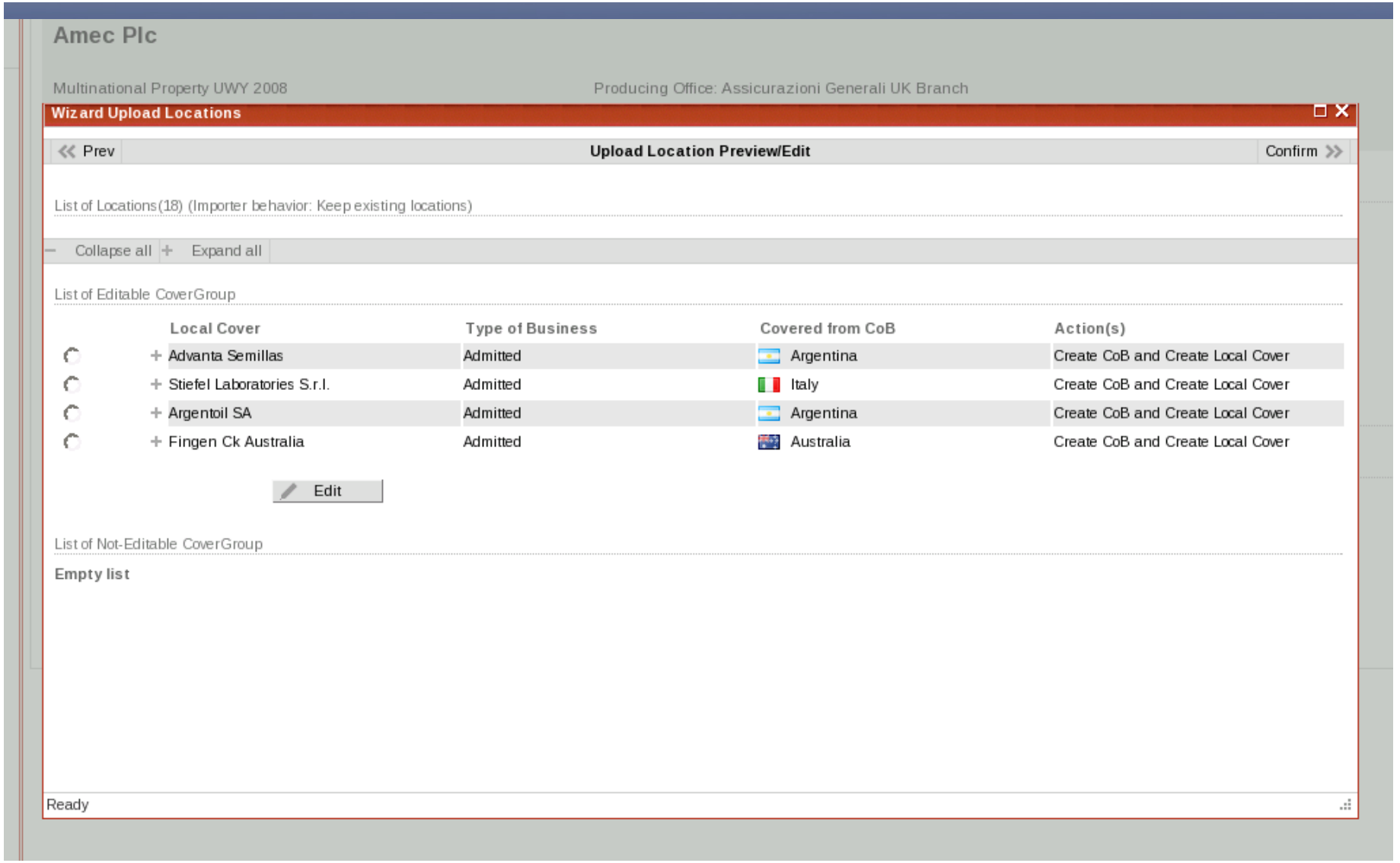
A2 | 06/01/2008

	A	B	C	D	E	F	G	H
1	Inception Date	Expiration Date	Local Policy Number	UWY	Sequence	Country Name	Country ISO Code	State/Province Na
2	1. Jun. 2008	30. May. 2009	DM03362	2006	0	ARGENTINA	AR	
3	1. Jun. 2008	30. May. 2009	DM03362	2006	0	ARGENTINA	AR	
4	1. Jun. 2008	30. May. 2009	DM03362	2006	0	ARGENTINA	AR	
5	1. Jun. 2008	30. May. 2009	DM03362	2006	0	ARGENTINA	AR	
6	1. Jun. 2008	30. May. 2009	DM03362	2006	0	ARGENTINA	AR	
7	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
8	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
9	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
10	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
11	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
12	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
13	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
14	1. Jun. 2008	30. May. 2009	DM03362	2006	0	AUSTRALIA	AU	
15	1. Jun. 2008	30. May. 2009	DM03362	2006	test1	ARGENTINA	AR	
16	1. Jun. 2008	30. May. 2009	DM03362	2006	test2	ARGENTINA	AR	
17	1. Jun. 2008	30. May. 2009	DM03362	2006	test3	ARGENTINA	AR	
18	1. Jun. 2008	30. May. 2009	DM03362	2006	test0	ITALY	IT	
19	1. Jun. 2008	30. May. 2009	DM03362	2006	test4	AUSTRALIA	AU	
20								
21								
22								
23								
24								
25								
26								

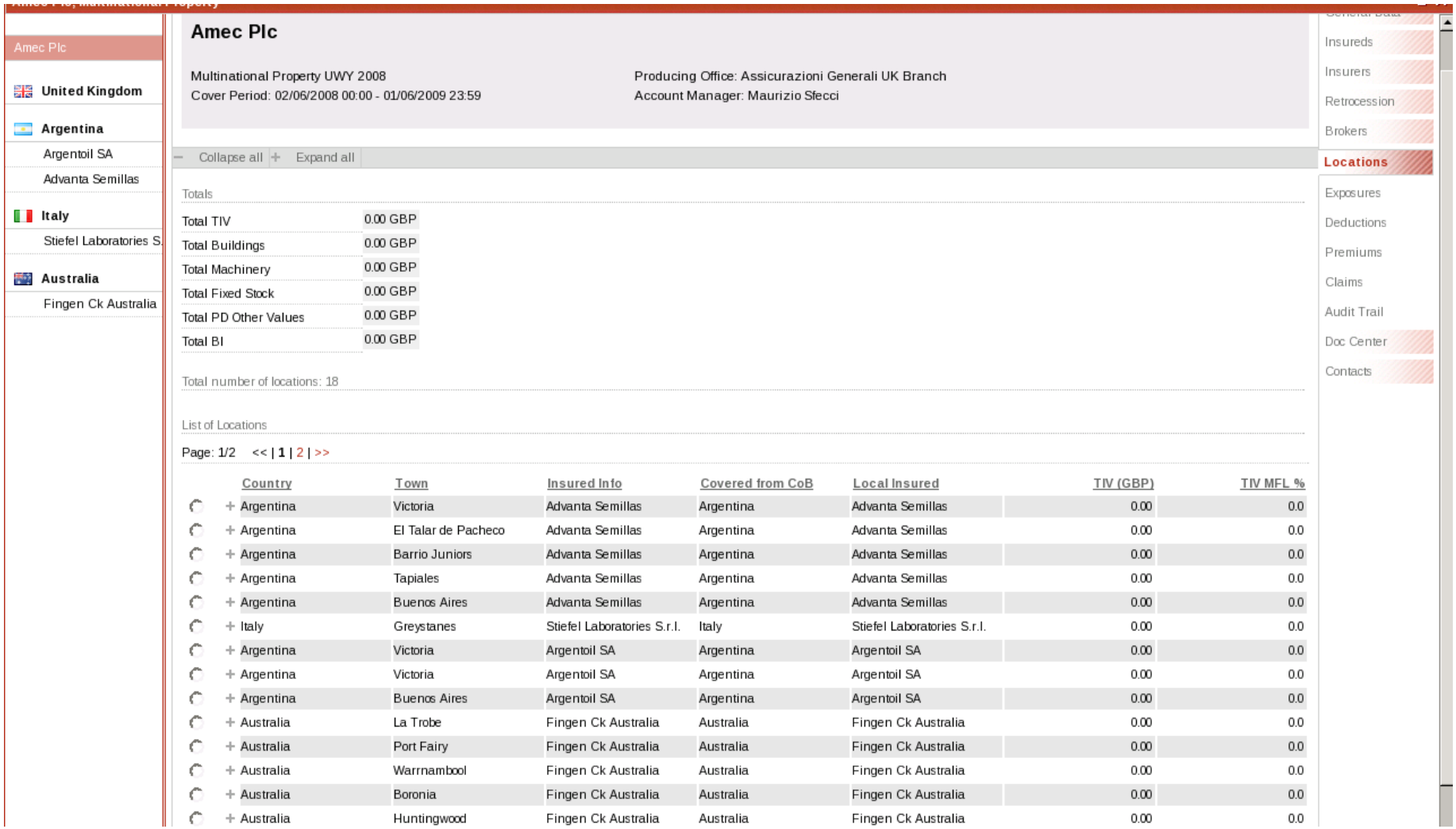
Examples of Functionality 7 of 9



Examples of Functionality 8 of 9



Examples of Functionality 9 of 9



Amec Plc

- United Kingdom
- Argentina
  - Argentoil SA
  - Advanta Semillas
- Italy
  - Stiefel Laboratories S.
- Australia
  - Fingen Ck Australia

Amec Plc

Multinational Property UWY 2008
Cover Period: 02/06/2008 00:00 - 01/06/2009 23:59

Producing Office: Assicurazioni Generali UK Branch
Account Manager: Maurizio Sfecci

Collapse all | Expand all

Totals

Total TIV	0.00 GBP
Total Buildings	0.00 GBP
Total Machinery	0.00 GBP
Total Fixed Stock	0.00 GBP
Total PD Other Values	0.00 GBP
Total BI	0.00 GBP

Total number of locations: 18

List of Locations

Page: 1/2 << | 1 | 2 | >>

Country	Town	Insured Info	Covered from CoB	Local Insured	TIV (GBP)	TIV MFL %
Argentina	Victoria	Advanta Semillas	Argentina	Advanta Semillas	0.00	0.0
Argentina	El Talar de Pacheco	Advanta Semillas	Argentina	Advanta Semillas	0.00	0.0
Argentina	Barrio Juniors	Advanta Semillas	Argentina	Advanta Semillas	0.00	0.0
Argentina	Tapiales	Advanta Semillas	Argentina	Advanta Semillas	0.00	0.0
Argentina	Buenos Aires	Advanta Semillas	Argentina	Advanta Semillas	0.00	0.0
Italy	Greystanes	Stiefel Laboratories S.r.l.	Italy	Stiefel Laboratories S.r.l.	0.00	0.0
Argentina	Victoria	Argentoil SA	Argentina	Argentoil SA	0.00	0.0
Argentina	Victoria	Argentoil SA	Argentina	Argentoil SA	0.00	0.0
Argentina	Buenos Aires	Argentoil SA	Argentina	Argentoil SA	0.00	0.0
Australia	La Trobe	Fingen Ck Australia	Australia	Fingen Ck Australia	0.00	0.0
Australia	Port Fairy	Fingen Ck Australia	Australia	Fingen Ck Australia	0.00	0.0
Australia	Warrnambool	Fingen Ck Australia	Australia	Fingen Ck Australia	0.00	0.0
Australia	Boronia	Fingen Ck Australia	Australia	Fingen Ck Australia	0.00	0.0
Australia	Huntingwood	Fingen Ck Australia	Australia	Fingen Ck Australia	0.00	0.0

Insureds
Insurers
Retrocession
Brokers
Locations
Exposures
Deductions
Premiums
Claims
Audit Trail
Doc Center
Contacts

Upload Locations Rules with DRL Language

```
rule "Inception date is after expiration data"
    dialect "java"
    when
    $row: SpreadsheetRowBean( $id: inceptionDate!=null, $ed: expirationDate!=null )

    eval($id.after($ed))

    then
    $row.addErrorForProperty("inceptionDate", "Inconsistent data: inception date cannot be after expiration date");
    $row.addErrorForProperty("expirationDate", "Inconsistent data: expiration date cannot be before inception date");
end


rule "Inception date is before inception date OR expiration date of Insured Cover"
    dialect "java"
    when
    $row: SpreadsheetRowBean( $id: inceptionDate!=null, $lic1: insuredCover !=null )

    eval($id.before($lic1.getInceptionDate()))
    or eval($id.after($lic1.getExpirationDate()))
    then
    $row.addErrorForProperty("Inconsistent data", "location inception date cannot be outside LIC cover period");
end


rule "Expiration date is before inception date OR is after expiration date of Insured Cover"
    dialect "java"
    when
    $row: SpreadsheetRowBean( $ed: expirationDate!=null, $lic1: insuredCover !=null )


    eval($ed.before($lic1.getInceptionDate()))
    or eval($ed.after($lic1.getExpirationDate()))
    then
    $row.addErrorForProperty("expirationDate", "Inconsistent data: location expiration date cannot be outside LIC cover period");
end
```

Inception date is after expiration date(DRL example)

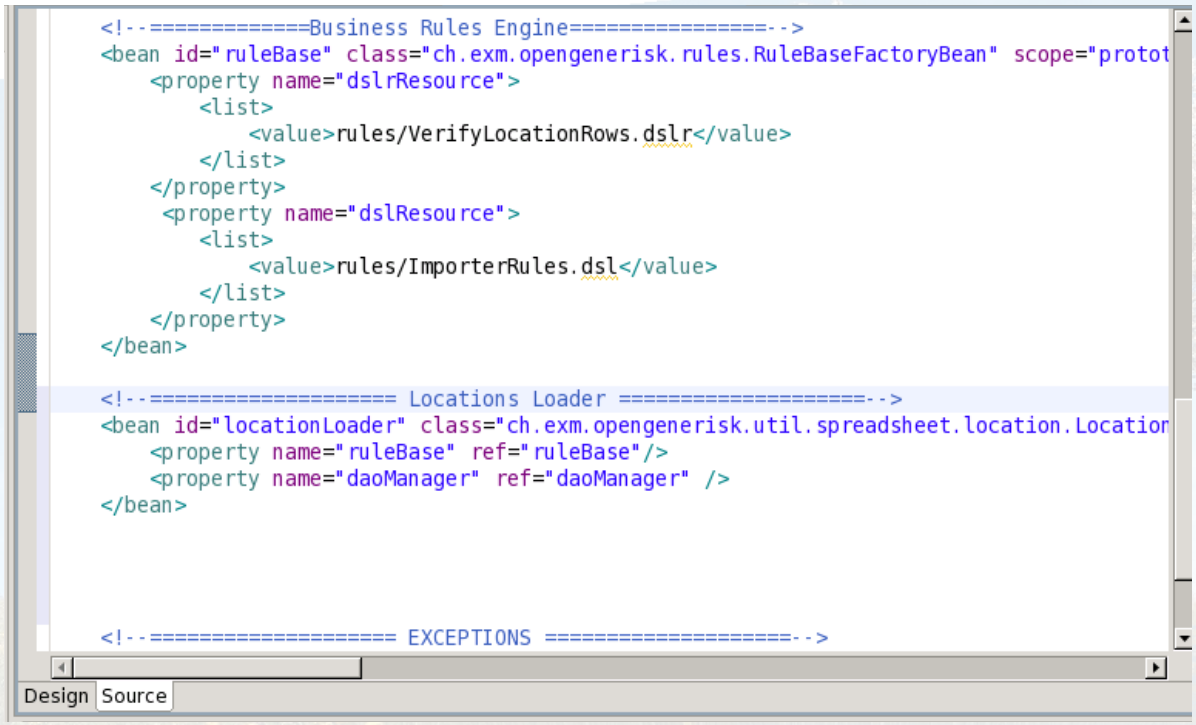
```
rule "Inception date is after expiration data"
    dialect "java"
    when
    $row: SpreadsheetRowBean( $id: inceptionDate!=null, $ed: expirationDate!=null )

    eval($id.after($ed))

    then
    $row.addErrorForProperty("inceptionDate", "Inconsistent data: inception date cannot be after expiration date");
    $row.addErrorForProperty("expirationDate", "Inconsistent data: expiration date cannot be before inception date");
end
```

Inception date is after expiration date(DSL example)

```
rule "Inception date is after expiration data"
    dialect "java"
    when
        There is a row in the spreadsheet where
            - inception date is not null
            - expiration date is not null

        Is inception date after expiration date

    then
        invalidate row for column with explanation : 'inceptionDate' 'Inconsistent data: inception date cannot be after expiration date'
        invalidate row for column with explanation : 'expirationDate' 'Inconsistent data: expiration date cannot be before inception date'
end

rule "Inception date is before inception dat
    dialect "java"
    when
```

- country is null	country == null
- get country iso code	$isocode: countryISOCode
Add loaded country to current row	$row.setCountry(ruleEngineHelper.getCountryByCountr
update row	update($row);
Read Business Entity	$be: BusinessEntity()
- country is not null	$country: country!=null
- local insured name is not null	$localInsuredName: localInsuredName!=null
- insured cover is null	insuredCover == null
There is a cover group where	$cg : CoverGroup()
- country Id equal to the current country Id	country.id == $country.id
Read cover groups from business entity	from $be.getCoverGroups()
There is insured cover with	$lic: InsuredCover()
- principal local insured name is equal to current lc	eval(principalLocalInsured.getName().equalsIgnoreCas
Read local insureds from cover group	from $cg.getLics()
Add insured cover to current row	$row.setInsuredCover($lic);
Add cover group of local insured to current row	$row.setCoverGroup($lic.getCoverGroup());

Spring & Drools



```
<!--=============Business Rules Engine================-->
<bean id="ruleBase" class="ch.exm.opengenerisk.rules.RuleBaseFactoryBean" scope="protot
    <property name="dslrResource">
        <list>
            <value>rules/VerifyLocationRows.dslr</value>
        </list>
    </property>
     <property name="dslResource">
        <list>
            <value>rules/ImporterRules.dsl</value>
        </list>
    </property>
</bean>

<!--==================== Locations Loader ====================-->
<bean id="locationLoader" class="ch.exm.opengenerisk.util.spreadsheet.location.Locatior
    <property name="ruleBase" ref="ruleBase"/>
    <property name="daoManager" ref="daoManager" />
</bean>



<!--==================== EXCEPTIONS ====================-->
```

Design Source

Drools Resource

- Drools
  - http://labs.jboss.com/drools/downloads.html
- Eclipse Plugin Installation
  - http://downloads.jboss.com/drools/updatesite3.3/
- Others resorces
  - http://en.wikipedia.org/wiki/Drools
  - http://en.wikipedia.org/wiki/Rete_algorithm
  - http://www.onjava.com/pub/a/onjava/2005/08/03/drools.html

Thank you !!!

danilo.ercoli@exmachina.ch

www.**exmachina**.ch
in <code> we trust