

# RuleML

Γλώσσα Σήμανσης  
Κανόνων



## Η ανάγκη για κάτι παραπάνω...

- Γνωρίζουμε ότι τα RDF έγγραφα αποτελούν το πρώτο βήμα για την αναπαράσταση γνώσης στον παγκόσμιο ιστό ώστε να εκμεταλλευόμαστε πλήρως τις δυνατότητές του.
- Προέκυψε όμως η ανάγκη για την εξαγωγή λογικών συμπερασμάτων χρησιμοποιώντας όλη αυτή την πληροφορία του ιστού.
- Έτσι δημιουργήθηκαν διάφορα **συστήματα κανόνων** που παρέχουν ένα φυσικό και ευρέως αποδεκτό μηχανισμό για την αυτοματοποιημένη εξαγωγή συμπερασμάτων με χρήση της τεχνολογίας.

## Συστήματα Κανόνων

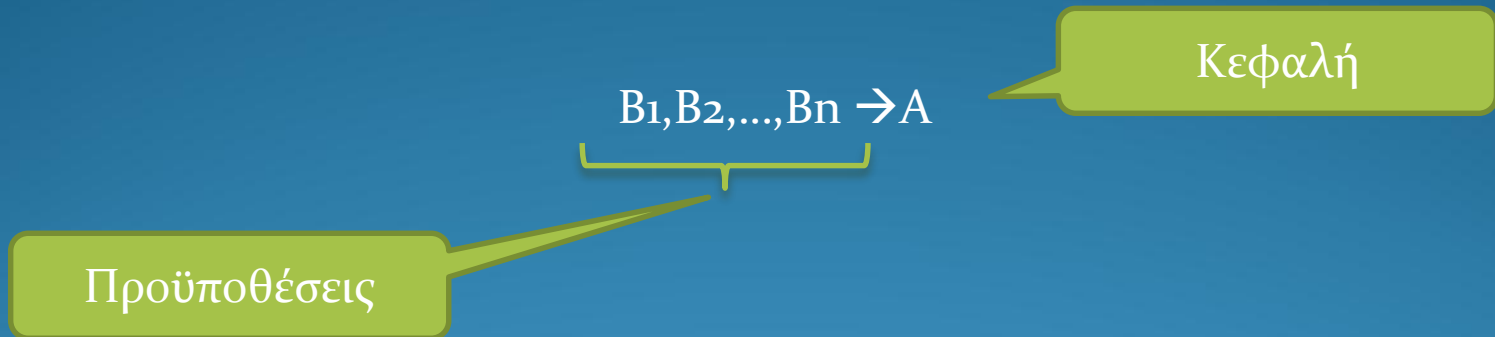


Υποσύνολο της κατηγορηματικής λογικής με αποδοτικά συστήματα αποδείξεων

Ένα σύστημα αποδείξεων περιλαμβάνει:

- Κανόνες
- Γεγονότα
- Λογικά προγράμματα
- Στόχους

## ΚΑΝΟΝΑΣ



όπου τα  $A, B_1, B_2, \dots, B_n$  είναι ατομικοί τύποι και τα κόμματα  $(,)$  αποτελούν συνδετικούς συνδέσμους.

Για παράδειγμα ένας κανόνας θα μπορούσε να είναι:

$\text{Σπουδάζει}(X, \Psi), \text{ΜένειΜόνιμα}(X, Z), \text{Τοποθεσία}(\Psi, A), \text{Τοποθεσία}(Z, A) \rightarrow \text{ΝτόπιοςΦοιτητής}(X)$

## ΓΕΓΟΝΟΣ

ατομικοί τύποι με εμμέσως καθολικά ποσοτικοποιημένες μεταβλητές

*Π.χ. Ντόπιος Φοιτητής (Γιώργος)*

## ΛΟΓΙΚΟ ΠΡΟΓΡΑΜΜΑ

πεπερασμένο σύνολο γεγονότων και κανόνων που συμβολίζεται με  $pl(P)$

## ΣΤΟΧΟΣ

$B_1, B_2, \dots, B_n \rightarrow$



Προϋποθέσεις

Ερώτημα που υποβάλλεται σε ένα λογικό πρόγραμμα  
Αν  $n$  είναι 0 τότε έχουμε κενό στόχο

## Τι είναι η RuleML?

- Η RuleML είναι μια σημαντική προσπάθεια προτυποποίησης για την σήμανση κανόνων στον Παγκόσμιο Ιστό.
- Δεν είναι μια γλώσσα αλλά πρόκειται στην πραγματικότητα για μια οικογένεια γλωσσών που αντιστοιχούν σε διαφορετικά είδη κανόνων.
- Οι κανόνες αυτοί είναι τα μέσα για την εξαγωγή συμπερασμάτων, την έκφραση περιορισμών, τον προσδιορισμό πολιτικών, την αντίδραση σε συμβάντα και αλλαγές, τον μετασχηματισμό δεδομένων κλπ.
- Πυρήνας της είναι η Datalog → μια λογική Horn χωρίς συναρτήσεις.

Η οικογένεια της RuleML παρέχει περιγραφές για τις γλώσσες σήμανσης κανόνων σε XML με τη μορφή σχημάτων XML. Η αναπαράσταση των συστατικών στοιχείων των κανόνων είναι σαφής.

Το βασικό λεξιλόγιο φαίνεται στον παρακάτω πίνακα:

Συστατικό στοιχείο κανόνα	RuleML
γεγονός	Asserted Atom
κανόνας	Asserted Implies
κεφαλή	Head
σώμα	Body
άτομο	Atom
σύζευξη	And
κατηγορημα	Rel
σταθερά	Ind
μεταβλητή	var

## Αναπαράσταση κανόνων με RuleML

Πώς αναπαριστάται κάθε δομικό στοιχείο της γλώσσας αυτής:

### Όροι

Οι όροι σε ένα αρχείο κανόνων RuleML μπορούν να είναι :

- Σταθερές, οπότε περικλείονται σε μια ετικέτα <ind>
- Μεταβλητές, οπότε περικλείονται σε μια ετικέτα <var>
- Συναρτήσεις, οπότε περιγράφονται από μια ετικέτα <function\_call>
- Κατηγορήματα, οπότε περιγράφονται από μια ετικέτα <\_opr>



## Αναπαράσταση κανόνων με RuleML

Πώς αναπαριστάται κάθε δομικό στοιχείο της γλώσσας αυτής:

### Ατομικές εκφράσεις

Οι ατομικές εκφράσεις περικλείονται σε μια ετικέτα <atom>. Για να αναπαραστήσουμε μια ατομική έκφραση, χρειαζόμαστε το κατηγορημα και της παραμέτρους του. Οπότε μια ατομική έκφραση αναπαρίσταται:

- Από μια ετικέτα <\_org> για το κατηγορημα και
- Από μια λίστα στοιχείων <\_slot> αντίστοιχη με τον αριθμό των παραμέτρων. Κάθε <\_slot> μπορεί να είναι είτε σταθερά, είτε μεταβλητή, είτε συνάρτηση, είτε μια λίστα των προηγούμενων, οπότε χρησιμοποιείται το στοιχείο <\_and>.

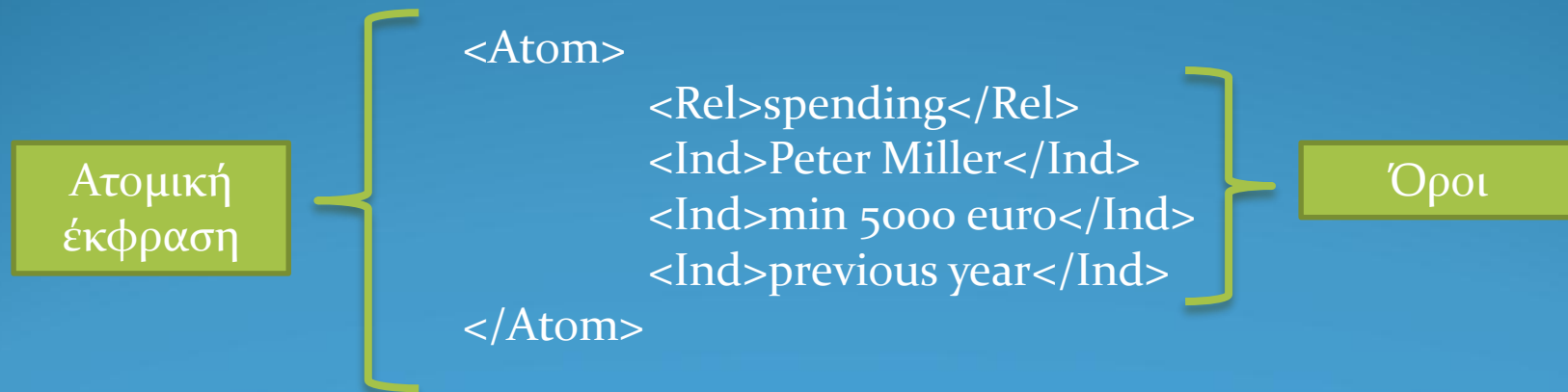
## Αναπαράσταση κανόνων με RuleML

### Παράδειγμα ατομικής έκφρασης :

Έστω ότι έχουμε την πρόταση :

*"Peter Miller's spending has been min 5000 euro in the previous year."*

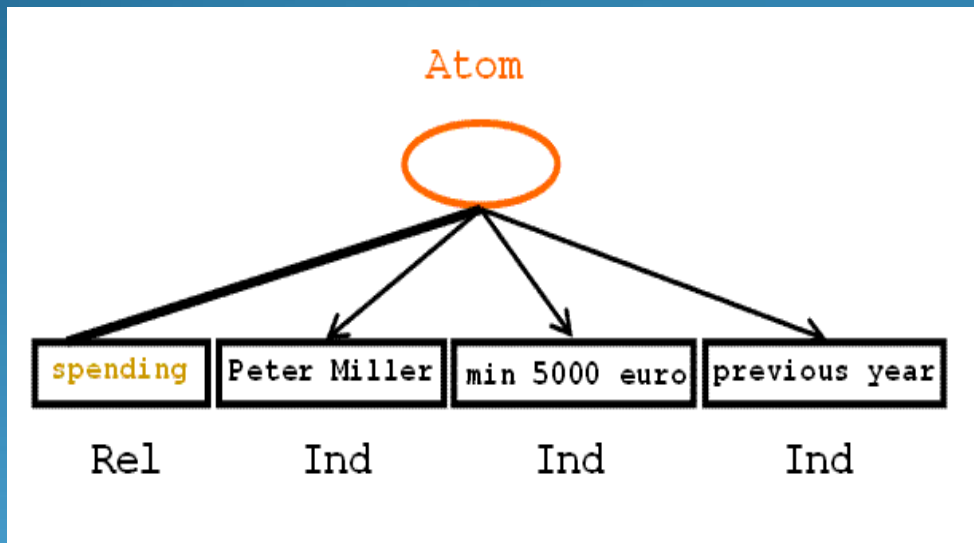
Το πρόγραμμα σε RuleML θα είναι:



# Αναπαράσταση κανόνων με RuleML

Παράδειγμα ατομικής έκφρασης :

*"Peter Miller's spending has been min 5000 euro in the previous year."*



<Atom>

<Rel>spending</Rel>

<Ind>Peter Miller</Ind>

<Ind>min 5000 euro</Ind>

<Ind>previous year</Ind>

</Atom>

## Αναπαράσταση κανόνων με RuleML

Πώς αναπαριστάται κάθε δομικό στοιχείο της γλώσσας αυτής:

### Κανόνες

Ένας κανόνας περικλείεται από την ετικέτα `<imp>`. Η κεφαλή ενός κανόνα περικλείεται από την ετικέτα `<_head>`, ενώ το σώμα ή οι προϋποθέσεις (premises) περικλείονται από την ετικέτα `<_body>`. Εάν στο σώμα του κανόνα υπάρχουν παραπάνω από μια ατομικές εκφράσεις, τότε αυτές περικλείονται από την ετικέτα λίστας `<and>`

## Αναπαράσταση κανόνων με RuleML

### Παράδειγμα κανόνα :

*"A customer is premium if their spending has been min 5000 euro in the previous year."*

Το πρόγραμμα σε RuleML θα είναι:

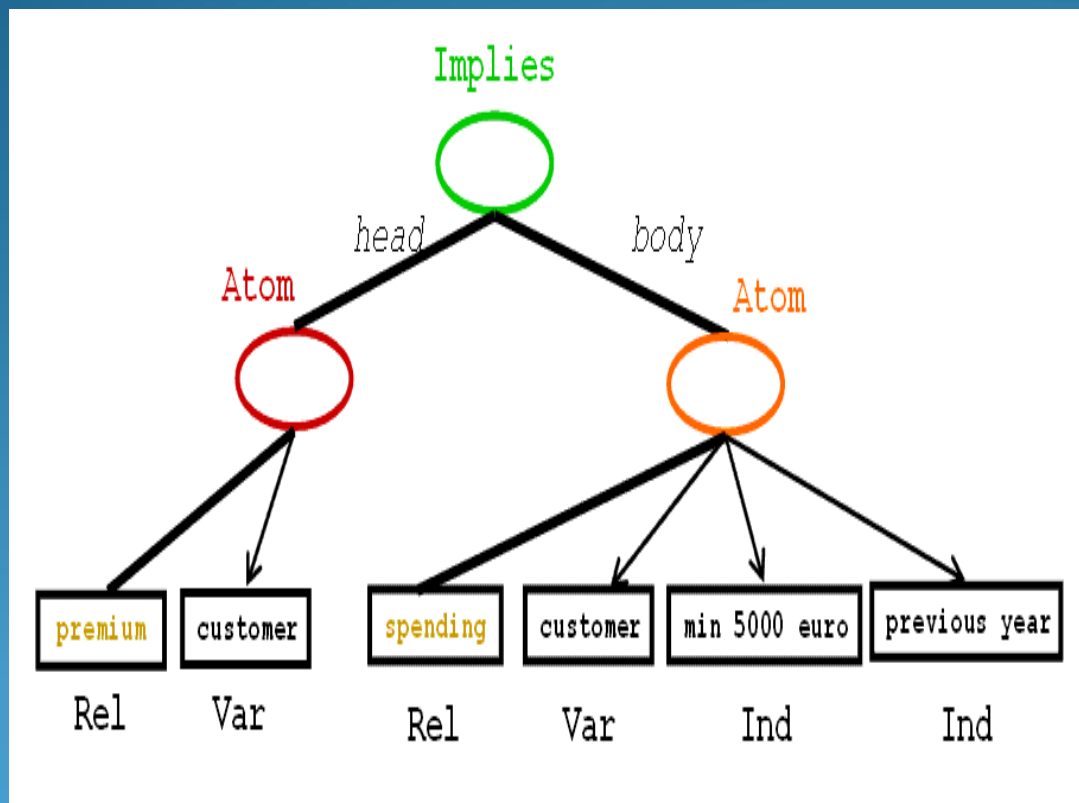
```
<Implies>
  <head>
    <Atom>
      <Rel>premium</Rel>
      <Var>customer</Var>
    </Atom>
  </head>
  <body>
    <Atom>
      <Rel>spending</Rel>
      <Var>customer</Var>
      <Ind>min 5000 euro</Ind>
      <Ind>previous year</Ind>
    </Atom>
  </body>
</Implies>
```

## Αναπαράσταση κανόνων με RuleML

Παράδειγμα κανόνα:

"A customer is premium if their spending has been min 5000 euro in the previous year."

$Spending(customer, \text{min } 5000\text{euro}, \text{previous year}) \rightarrow premium(customer)$



```
<Implies>
  <head>
    <Atom>
      <Rel>premium</Rel>
      <Var>customer</Var>
    </Atom>
  </head>
  <body>
    <Atom>
      <Rel>spending</Rel>
      <Var>customer</Var>
      <Ind>min 5000 euro</Ind>
      <Ind>previous year</Ind>
    </Atom>
  </body>
</Implies>
```

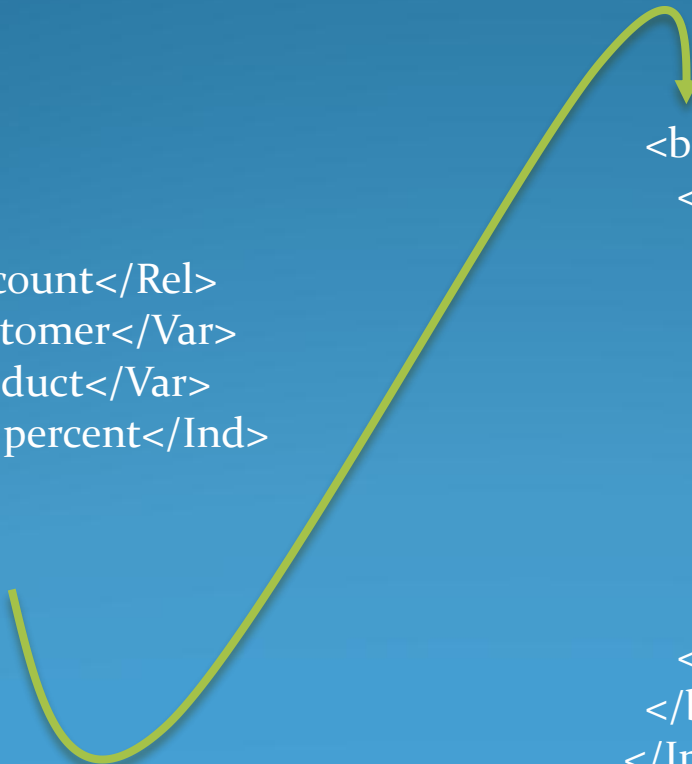
## Αναπαράσταση κανόνων με RuleML

Ακόμα ένα παράδειγμα κανόνα με χρήση ετικέτας λίστας <and> :

Έστω ότι έχουμε την πρόταση :

*"The discount for a customer buying a product is 7.5 percent if the customer is premium and the product is luxury."*

```
<Implies>
  <head>
    <Atom>
      <Rel>discount</Rel>
      <Var>customer</Var>
      <Var>product</Var>
      <Ind>7.5 percent</Ind>
    </Atom>
  </head>
```



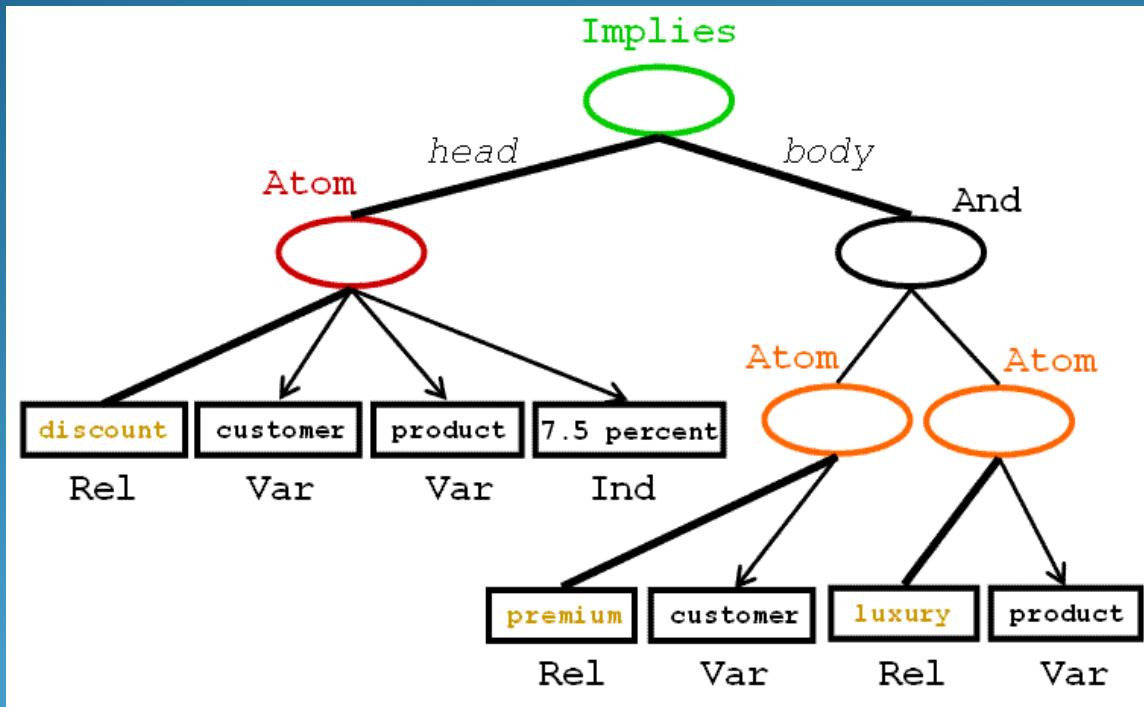
```
<body>
  <And>
    <Atom>
      <Rel>premium</Rel>
      <Var>customer</Var>
    </Atom>
    <Atom>
      <Rel>luxury</Rel>
      <Var>product</Var>
    </Atom>
  </And>
</body>
</Implies>
```

## Αναπαράσταση κανόνων με RuleML

Ακόμα ένα παράδειγμα κανόνα με χρήση ετικέτας λίστας <and> :

"The discount for a customer buying a product is 7.5 percent if the customer is premium and the product is luxury."

$Premium(customer), luxury(product) \rightarrow discount(customer, product, 7.5\text{percent})$



```
<Implies>
  <head>
    <Atom>
      <Rel>discount</Rel>
      <Var>customer</Var>
      <Var>product</Var>
      <Ind>7.5 percent</Ind>
    </Atom>
  </head>
  <body>
    <And>
      <Atom>
        <Rel>premium</Rel>
        <Var>customer</Var>
      </Atom>
      <Atom>
        <Rel>luxury</Rel>
        <Var>product</Var>
      </Atom>
    </And>
  </body>
</Implies>
```



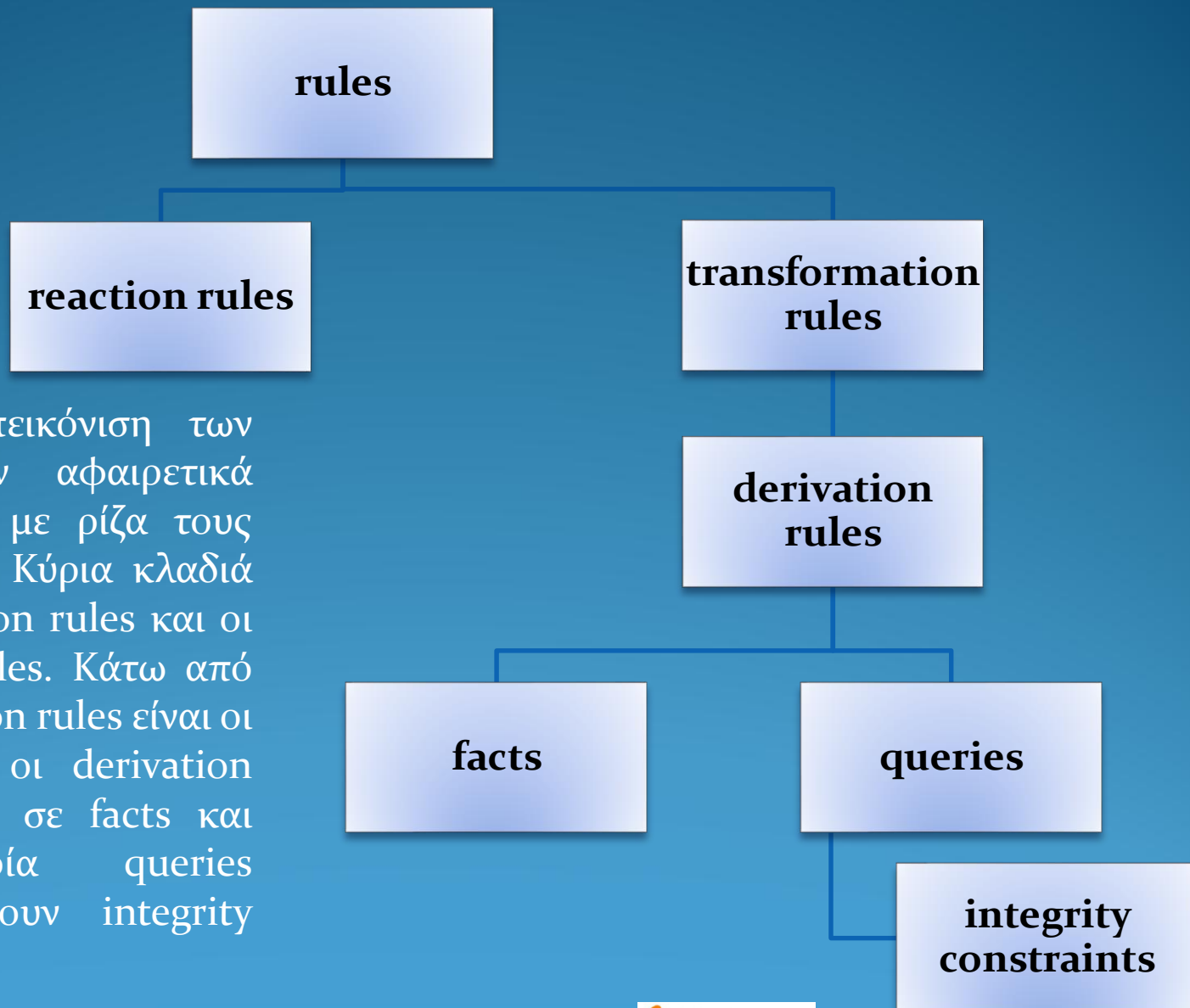
## Αναπαράσταση κανόνων με RuleML

Πώς αναπαριστάται κάθε δομικό στοιχείο της γλώσσας αυτής:

### Λογικό πρόγραμμα

Ένα λογικό πρόγραμμα περικλείεται από την ετικέτα <rulebase> και περιλαμβάνει όλα τα πιο πάνω δομικά στοιχεία. Κλείνει με χρήση του <rulebase>

# The RuleML Design



Μια γραφική απεικόνιση των RuleML κανόνων αφαιρετικά είναι ένα δέντρο με ρίζα τους γενικούς κανόνες. Κύρια κλαδιά του είναι οι reaction rules και οι transformation rules. Κάτω από τους transformation rules είναι οι derivation rules, οι derivation rules ειδικεύονται σε facts και queries, τα οποία queries μπορούν να γίνουν integrity constraints

## Τύποι-παραλλαγές της RuleML

Υπάρχουν πολλές παραλλαγές της RuleML. Μερικές από αυτές είναι :

### ➤ **Object-Oriented RuleML**

(είναι ένας συνδυασμός User-Level ρόλων, URI-Grounded προτάσεων και Order-Sorted όρων)

### ➤ **Fuzzy RuleML**

( περιλαμβάνει κανόνες που περιέχουν το στοιχείο της αβεβαιότητας με αποτέλεσμα να χρειάζεται ειδικός χειρισμός για να καταλήξουμε σε κάποιο αποτέλεσμα.)

### ➤ **Reaction RuleML**

(περιλαμβάνει κανόνες που ελέγχουν την «κατάσταση» και προβαίνουν σε κατάλληλους χειρισμούς.)

### ➤ **Functional RuleML**

(περιλαμβάνει την ύπαρξη συναρτήσεων που μας διευκολύνουν να αποτυπώσουμε σχέσεις και εξαρτήσεις που υπάρχουν.)

➤ ...

# Object-Oriented RuleML

Χαρακτηριστικά της OO RuleML είναι ότι :

1. τα slots μοιάζουν με τις κλάσεις και τα instances των κλάσεων. Έτσι έχουμε μια αντικειμενοστραφή προσέγγιση που δίνει την δυνατότητα της κληρονομικότητας μεταξύ των slots.
2. στηρίζει URI ως μοναδικά προσδιοριστικά αντικειμένων (OIDs) για τα γεγονότα και για τους κανόνες.
3. Υποστηρίζει ταξονομίες αντίστοιχες με τις ιεραρχίες κλάσεων.

# Fuzzy RuleML

$$reach(R, S) \leftarrow 0.9 - road(R, S) \& south(R, S)$$

which can be represented in our uncertainty extension as

```
<Implies kind="goguen">
  <degree>
    <Data>0.9</Data>
  </degree>
  <head>
    <Atom><op><Rel>reach</Rel></op>
      <Var>R</Var>
      <Var>S</Var>
    </Atom>
  </head>
  <body>
    <And kind="minimum">
      <Atom><op><Rel>road</Rel></op>
        <Var>R</Var>
        <Var>S</Var>
      </Atom>
      <Atom><op><Rel>south</Rel></op>
        <Var>R</Var>
        <Var>S</Var>
      </Atom>
    </And>
  </body>
</Implies>
```

Η κεφαλή του κανόνα

ο βαθμός αβεβαιότητας

Η κεφαλή του κανόνα

# Reaction RuleML

Καθορίζει μια σύνταξη για τους reaction rules

- Εισάγει γεγονότα
- Συνδυάζει γεγονότα και action algebra
- Διαθέτει μηνύματα ενημέρωσης
- Παρέχει περιορισμούς ακεραιότητας και μηχανισμούς ελέγχων

## Syntax

Core Reaction Rule 0.2 Syntax:

```
<Rule style="active" evaluation="strong">  
<label> <!-- meta data --> </label> <scope> <!-- general scope of rule --> </scope>  
<qualification> <!-- qualifications --> </qualification> <oid> <!-- object identifier -->  
</oid>  
<on> <!-- event --> </on>  
<if> <!-- condition --> </if>  
<then> <!-- conclusion --> </then>  
<do> <!-- action --> </do>  
<after> <!-- postcondition --> </after>  
<else> <!-- else conclusion --> </else>  
<elseDo> <!-- else/alternative action --> </elseDo>  
<elseAfter> <!-- else postcondition --> </elseAfter>  
</Rule>
```

Παράδειγμα  
σύνταξης:

## Υλοποιήσεις RuleML

ΟΟ jDREW, είναι μια παραγωγική reasoning engine για RuleML και ταυτόχρονα μια αντικειμενοστραφής επέκταση του [jDREW](#). ΟΟ jDREW εισάγει αντικειμενοστραφείς επεκτάσεις στην RuleML που περιλαμβάνουν :

- Ταξινομημένους τύπους
- Slots
- Προσδιοριστικά αντικειμένων

Το ΟΟ jDREW είναι γραμμένο σε Java.



<http://www.jdrew.org/oojdrew/index.html>

## Υλοποιήσεις RuleML

**XpertRule**<sup>®</sup>

Knowledge Enabled Solutions

Expenses (claim underwriting demo)

Savings and Investments (needs analysis demo)

Pension (advisory demo)

Direct PC Sales (product recommendation and configuration demo)

Support (customer eService demo)

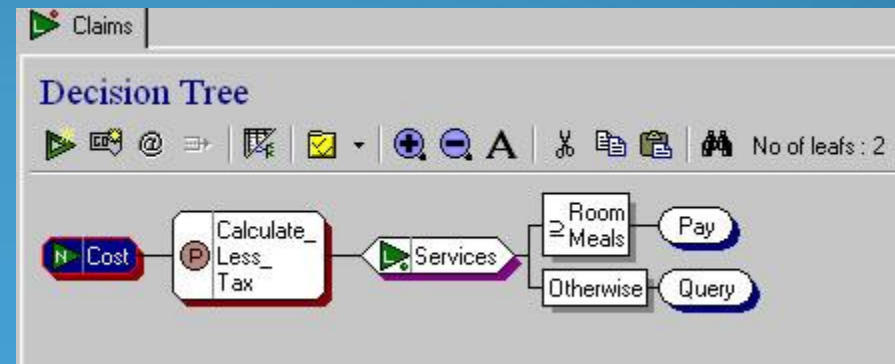
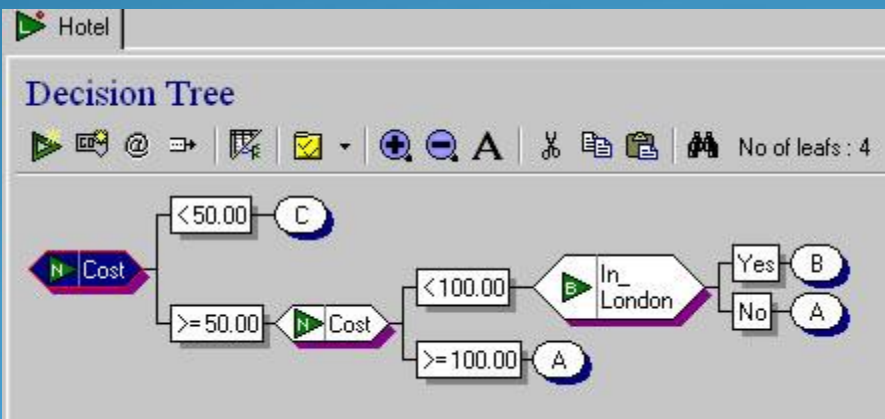
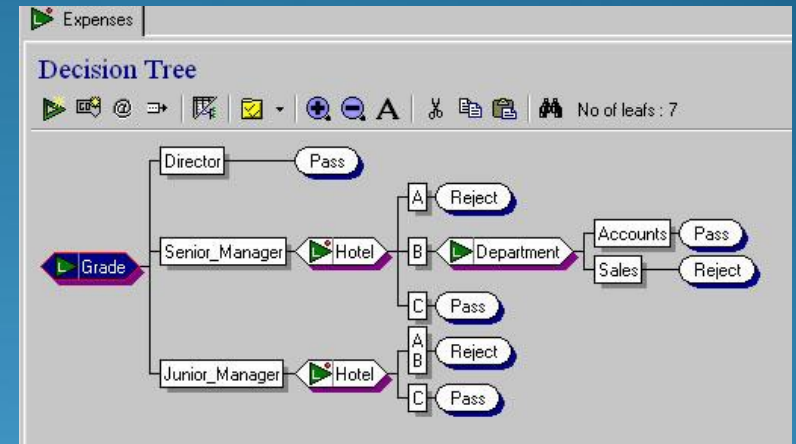
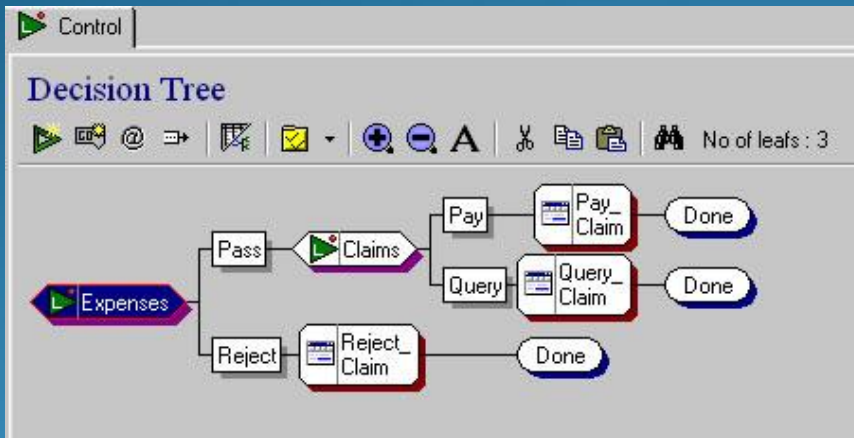


# Υλοποιήσεις RuleML



Knowledge Enabled Solutions

## Expenses (claim underwriting demo)

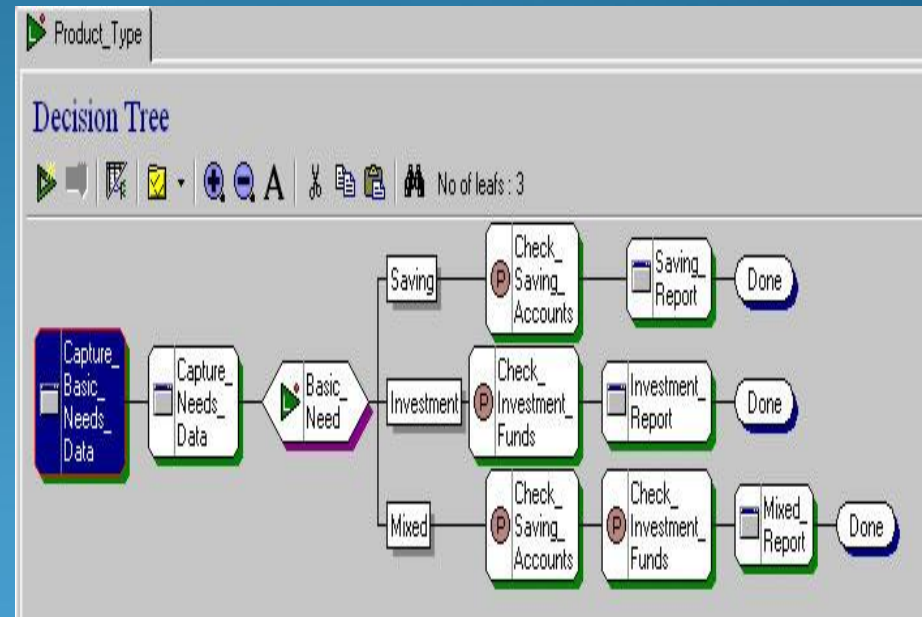
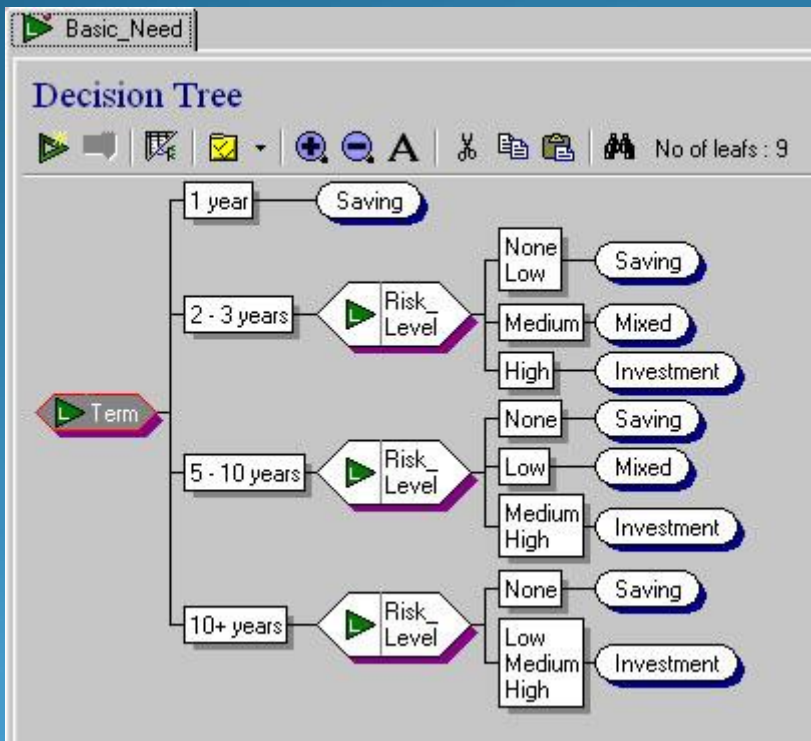


# Υλοποιήσεις RuleML

XpertRule®

Knowledge Enabled Solutions

## Savings and Investments (needs analysis demo)

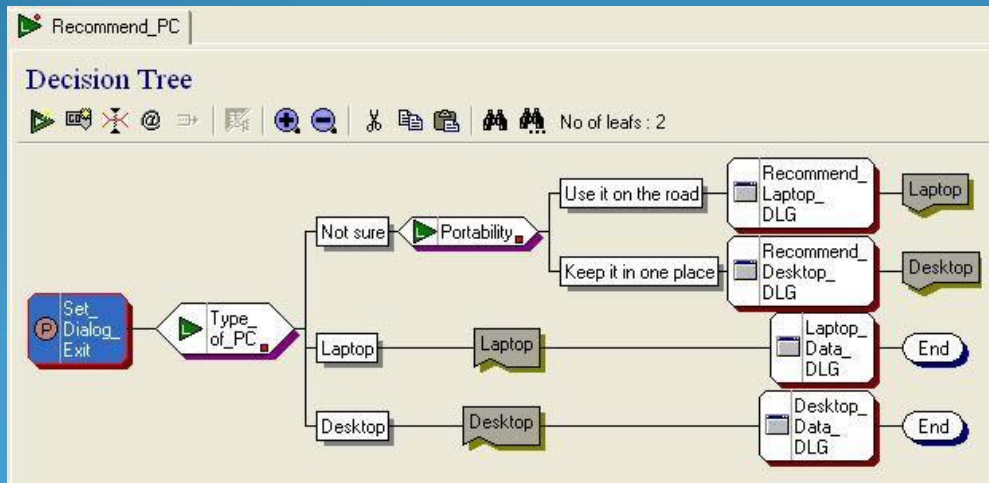
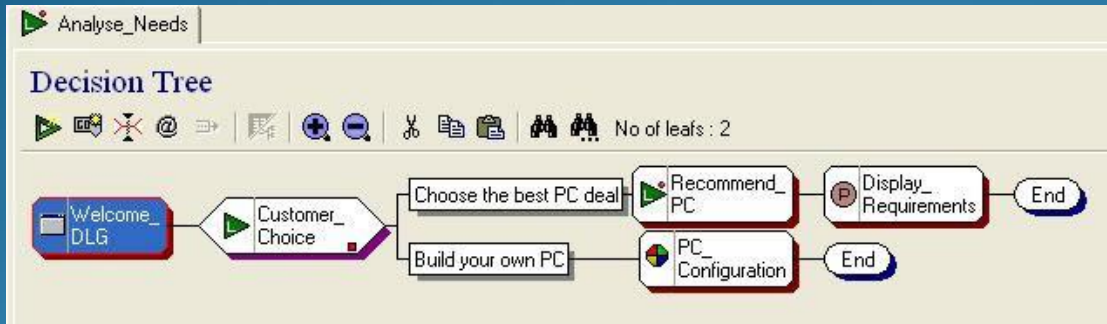


# Υλοποιήσεις RuleML



Knowledge Enabled Solutions

## Direct PC Sales (product recommendation and configuration demo)



Desktop\_Deals

Cases

	Type_of_Customer	Main_Usage	Desktop_Deals	
1	Personal	*	DV102	
2	Personal	*	DX464	
3	*	Client	DX528	
4	Business	Client	DX656	
5	*	*	DX756	
6	*	*	DP558	
+				

# Εργαλεία: Acumen Business Rule Manager

## Acumen Business Rule Manager



Γραφικό περιβάλλον  
για Επιχειρηματικούς Κανόνες  
στην RuleML



# Εργαλεία: Acumen Business Rule Manager

## Rule Editor

young female

if gender is equal to female  
and age is less than 20  
then age category = young

---

Name: young male Revert Save Delete

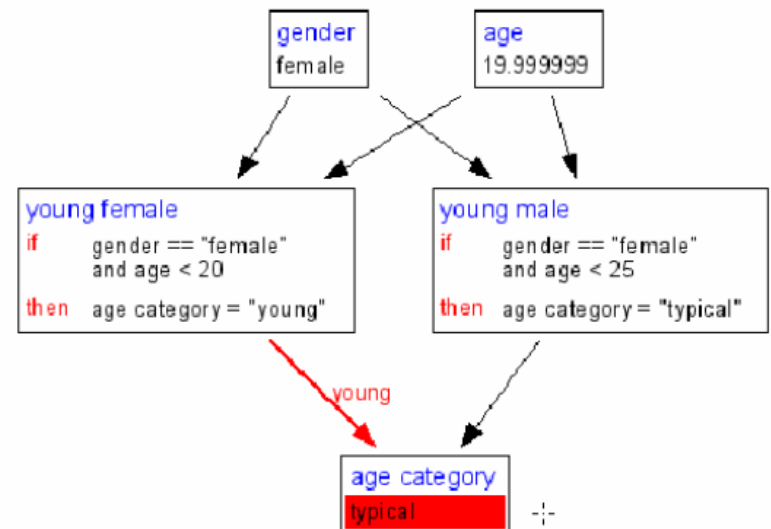
Description:

Priority: 0 Active Active Author: Marco Ensing

if gender is equal to female  
and age is less than 25  
then age category = typical

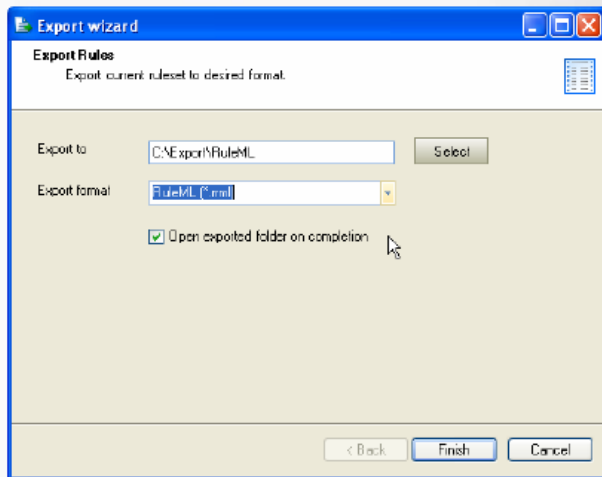
*(Note: In the original image, a dropdown menu is open for the 'then' clause, showing 'typical' and 'young' options.)*

## Rule Conflict Network



# Εργαλεία: Acumen Business Rule Manager

## Export wizard



## RuleML output

The screenshot shows a Notepad window titled 'Driver Eligibility.rml - Notepad'. The window contains the following XML code:

```
</Expr>
</plex>
</label>
<scope>
  <Ind
    uri="#Driver Eligibility" />
</scope>
<id>rules</id>
<if>
  <And>
    <Equal>
      <lhs>
        <Atom>
          <Rel>gender</Rel>
          <Var>Driver</Var>
        </Atom>
      </lhs>
      <rhs>
        <Ind
          type="string">"male"</Ind>
        </rhs>
      </Equal>
    </And>
    <Atom>
      <Rel>lessthan</Rel>
      <Atom>
        <Rel>age</Rel>
        <Var>Driver</Var>
      </Atom>
      <Ind
        type="int">"25"</Ind>
      </Atom>
    </And>
  </if>
  <do>
    <Atom>
      <Rel>age category</Rel>
      <Var>Driver</Var>
      <Ind
        type="string">"young"</Ind>
      </Atom>
    </do>
  </Rule>
</Rulebase>
</RuleML>
```

## Χρήσεις κανόνων RuleML και εφαρμογές

Οι κανόνες μπορούν να χρησιμοποιηθούν σε αρκετές περιπτώσεις όπου πραγματικά μπορούν να δείξουν πόσο σημαντικό ρόλο έχουν στην αυτοματοποίηση διαδικασιών, όπως :

- διαπραγμάτευση συμβολαίων e-επιχειρείν
- διαπραγμάτευση συναλλαγών ηλεκτρονικού εμπορίου
- δια-εταιρικές επιχειρηματικές πολιτικές
- υποστήριξη αποφάσεων σε ιατρικά θέματα
- κατανεμημένη αναζήτηση
- σε έργα data integration μπορούν να χρησιμοποιηθούν σαν κοινή γλώσσα αντιστοίχισης όρων (vocabulary mapping)
- απομακρυσμένη μάθηση/διδασκαλία
- αυτοματοποιημένη αξιολόγηση περιεχομένων του διαδικτύου.



# Βιβλιογραφία

Στις ιστοσελίδες :

- [www.ruleml.org](http://www.ruleml.org)
- [www.xpertrule.com](http://www.xpertrule.com)
- <http://www.jdrew.org/ooidrew/index.html>

...και με γενικότερη χρήση του διαδικτύου.