Proof explanations: using natural language and graph view

Frédérique GUILHOT
Hanane NACIRI
Loïc POTTIER

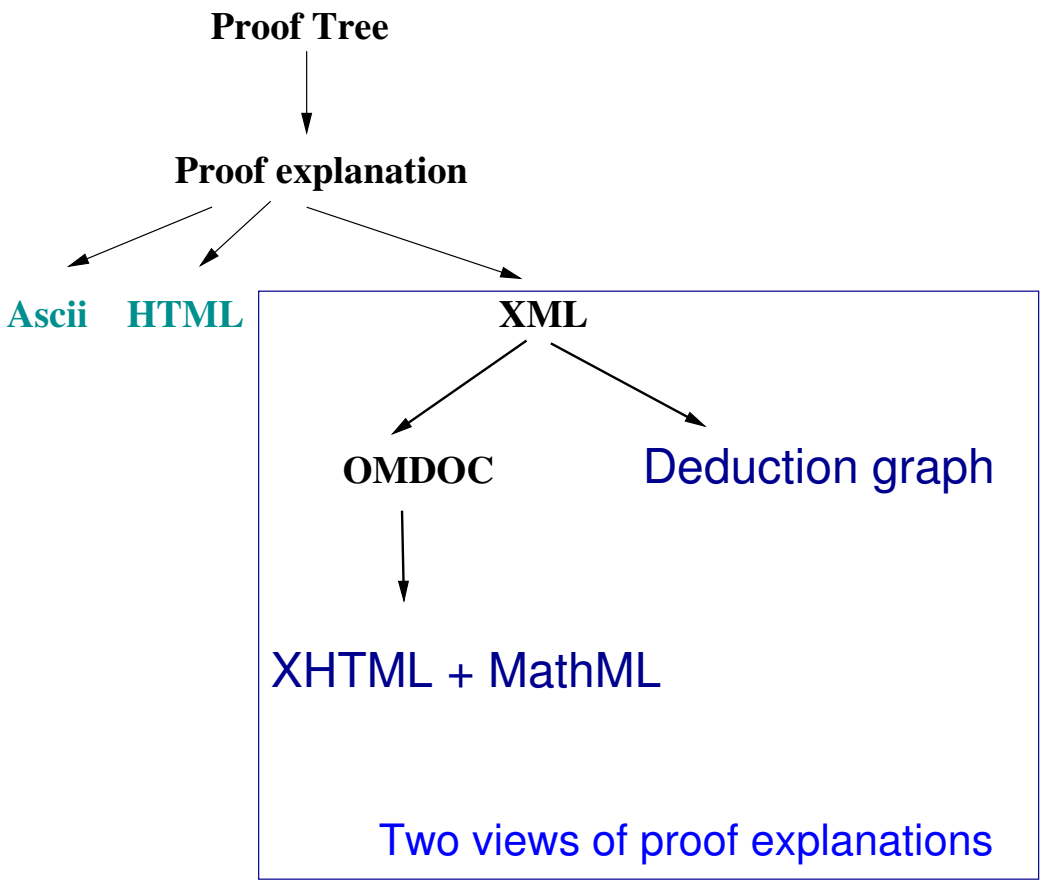
September 2003

MOWGLI

Proof explanations: document and graph view

- From a proof tree, we want to provide:
  - a web document presenting the proof explanation in natural language with appropriate mathematical notations
  - a deduction graph of the proof (to help understanding the proof steps and possibly improving the proof)

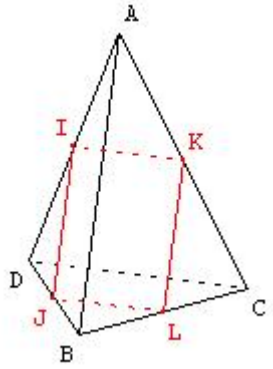
Proof explanations: document and graph view (2)



Example: geometry exercise - tetrahedron

- Proof script in Coq

```
Lemma deux_milieux_tetraedre:
(A, B, C, D, I, J, K, L : PO)
(tetraedre A B C D) ->
(I == (milieu D A)) ->
(J == (milieu D B)) ->
(K == (milieu C A)) ->
(L == (milieu C B)) ->
(vec I J) == (vec K L).
```




```
Intros.
Cut (mult_PP (Rplus R1 R1) (vec I J)) == (vec A B); Intros.
Cut (mult_PP (Rplus R1 R1) (vec K L)) == (vec A B); Intros.
Apply mult_PP_regulier with (Rplus R1 R1); Auto with real.
Rewrite H5; Trivial.
Apply droite_milieu with C; Auto.
Apply droite_milieu with D; Auto.
Qed.
```

Example: geometry exercise-tetrahedron (2)

- Proof explanation in natural language

THEOREM: $\forall A : PO, \forall B : PO, \forall C : PO, \forall D : PO, \forall I : PO, \forall J : PO, \forall K : PO, \forall L : PO\ ((tetraedre\ A\ B\ C\ D) \rightarrow\ I = (milieu\ D\ A) \rightarrow J = (milieu\ D\ B) \rightarrow K = (milieu\ C\ A) \rightarrow L = (milieu\ C\ B) \rightarrow \overrightarrow{IJ} = \overrightarrow{KL})$

PROOF: Let A, B, C, D, I, J, K and L be elements of PO such that $(tetraedre\ A\ B\ C\ D)$ (H), $I = (milieu\ D\ A)$ (H0), $J = (milieu\ D\ B)$ (H1), $K = (milieu\ C\ A)$ (H2) and $L = (milieu\ C\ B)$ (H3)

Let's prove $\overrightarrow{IJ} = \overrightarrow{KL}$

- From $I = (milieu\ D\ A)$ and $J = (milieu\ D\ B)$ we deduce $2 * \overrightarrow{IJ} = \overrightarrow{AB}$ by using $droite\_milieu$.

  We have $2 * \overrightarrow{IJ} = \overrightarrow{AB}$ (H4).

- From $K = (milieu\ C\ A)$ and $L = (milieu\ C\ B)$ we deduce $2 * \overrightarrow{KL} = \overrightarrow{AB}$ by using $droite\_milieu$.

  We have $2 * \overrightarrow{KL} = \overrightarrow{AB}$ (H5).

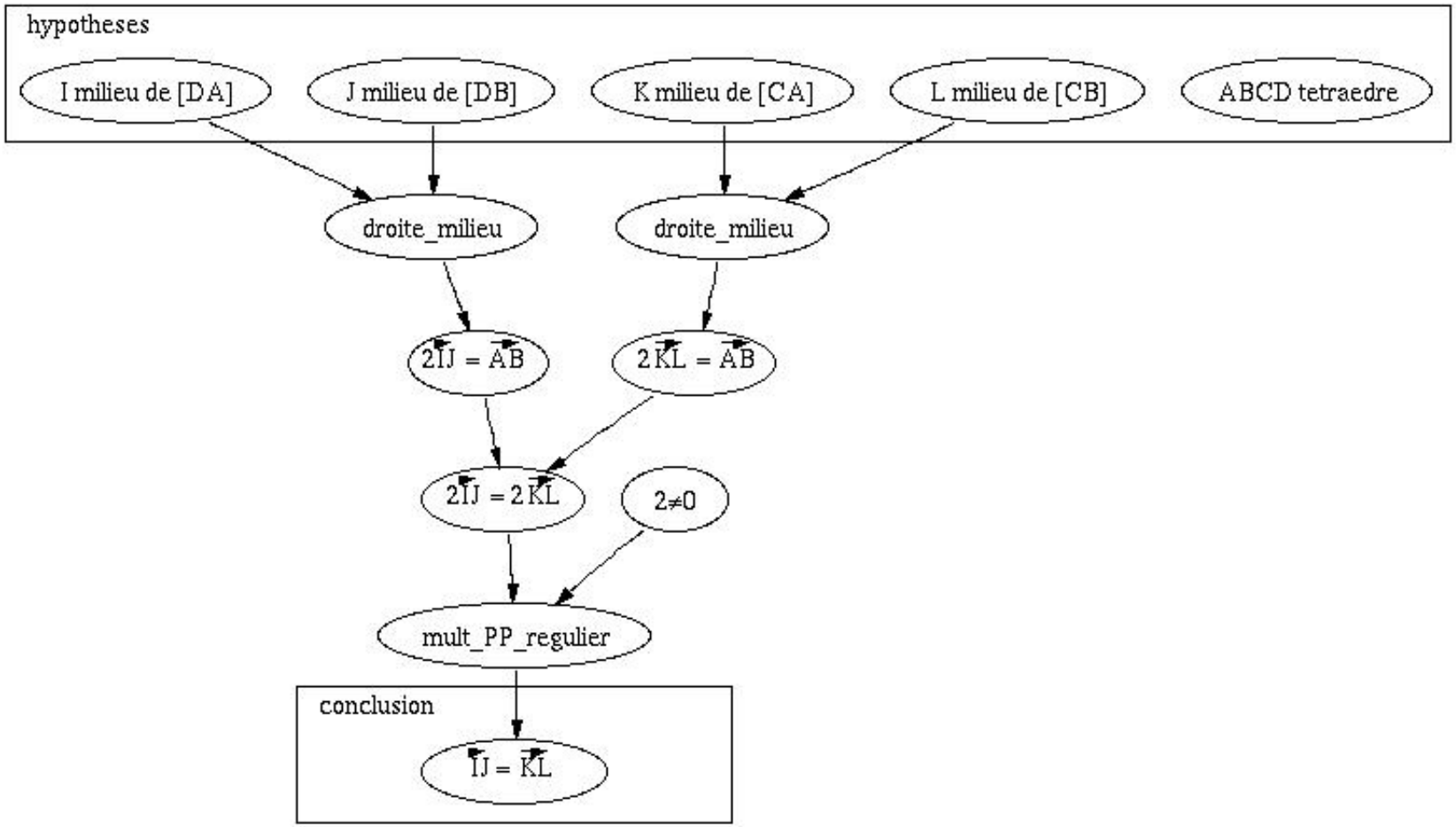
- $2 \neq 0$ is obvious.

  From $2 * \overrightarrow{IJ} = \overrightarrow{AB}$ and $2 * \overrightarrow{KL} = \overrightarrow{AB}$ we deduce $2 * \overrightarrow{IJ} = 2 * \overrightarrow{KL}$

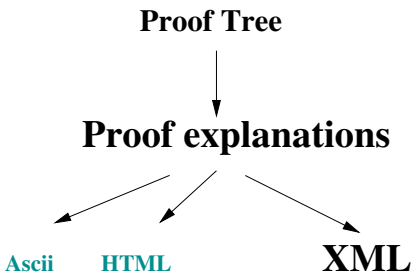
From $2 \neq 0$ and $2 * \overrightarrow{IJ} = 2 * \overrightarrow{KL}$ we deduce $\overrightarrow{IJ} = \overrightarrow{KL}$ by using $mult\_PP\_regulier$

Example: geometry exercise-tetrahedron (3)

- Deduction graph

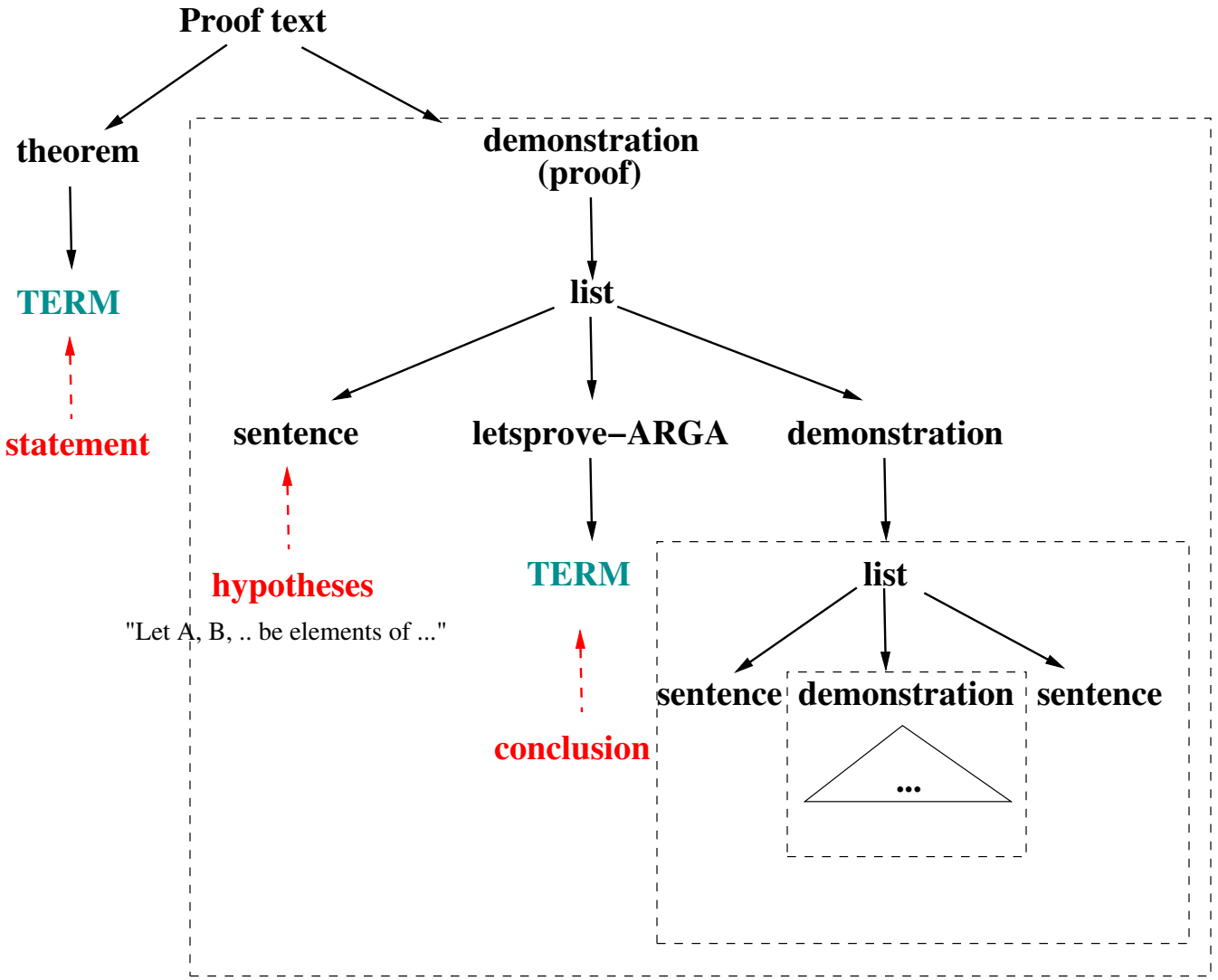


New XML Proof explanation structure



- the document is composed of a theorem statement and of its demonstration
  - The main proof is a list of explanation sentences and of sub proofs
  - The logical and mathematical formulas in proof explanations are CIC (The Calculus of Inductive Constructions) terms

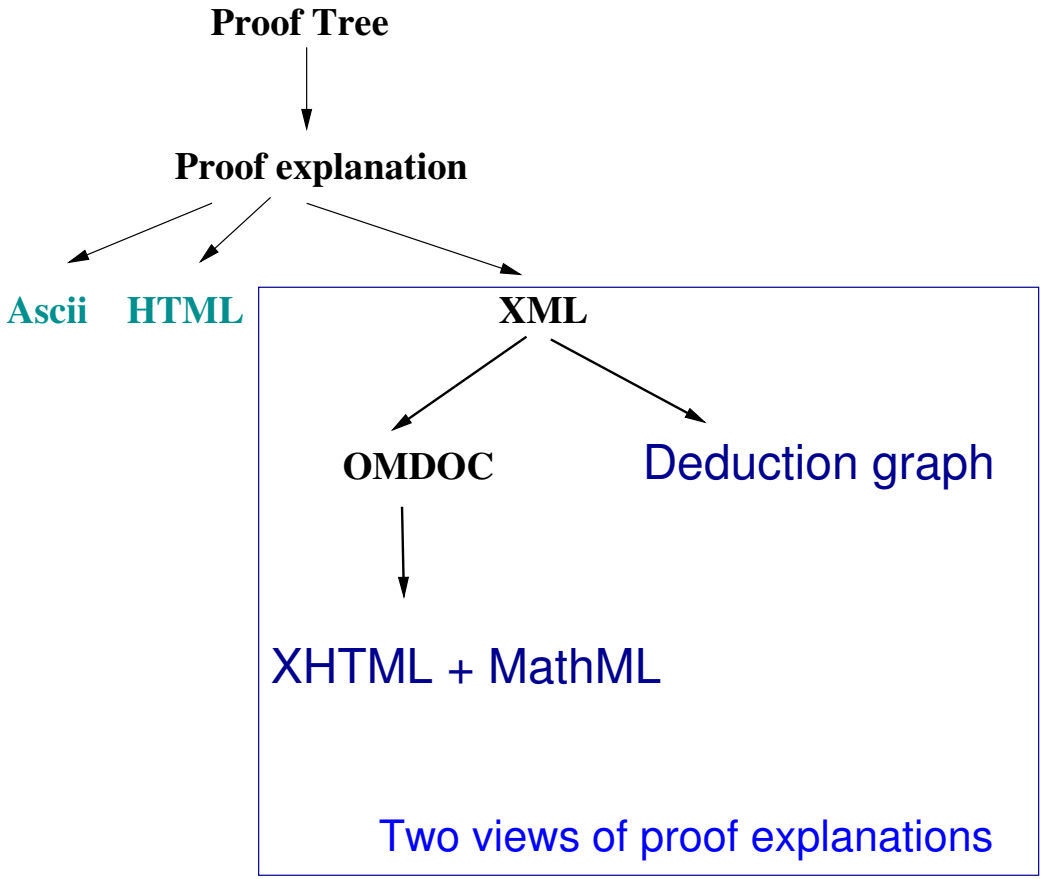
New XML proof explanation structure (2)



New XML proof explanation structure (3)

- *TERM*s are CIC terms
- Several sentence types exist:
  - Example: From $I = (milieu\ D\ A)$ and .. we deduce .. by using ..
    <By-using>
    <From-ARGA-we-deduce-ARGB>..</From-ARGA-we-deduce-ARGB>
    <by-using-ARGA>..</by-using-ARGA>
    </By-using>
  - Example : Let A, B.. be elements of PO such that ...
    <List>
    <List>..</List>
    <Text>..</Text>
    <List-comma-and>
    <Hypothesis>..</Hypothesis>
    <Hypothesis>..</Hypothesis>
    </List-comma-and>
    </List>

Proof explanation in XML → OMDOC → XHTML/MathML



Representing proof explanations in OMDOC

Proof explanation in XML $\longrightarrow$ OMDOC $\longrightarrow$ XHTML/MathML

- We apply XSLT transformation rules on the XML proof explanation in order to obtain the OMDOC proof document:

<proof-text>
 <theorem>..</theorem>
 <demonstration>..</demonstration>
</proof-text>

$\longrightarrow$

<omdoc>
 <assertion id="a1">..</assertion>
 <proof for="a1">..</proof>
</omdoc>

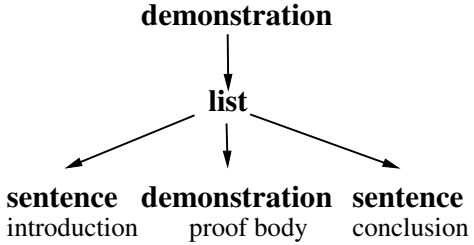
- **proof** :

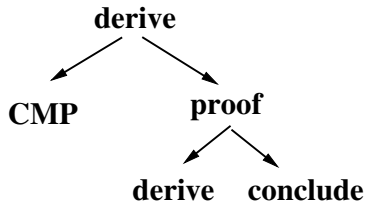
Representing proof explanations in OMDOC (2)

- **sub proof**: each proof step that induces a new claim is represented by a *derive* element
  - a sequence of demonstration elements gives a sequence of derive elements

<demonstration> c_1 .. c_n </demonstration> $\longrightarrow$ <omdoc:proof> $derive_1$ .. $derive_n$ </omdoc:proof>

  - In some cases, the demonstration structure is sentence (introduction), proof (body), sentence (conclusion).





- We distinguish 2 cases : proof body
- Hence, for every case we have proved ...

Representing Proof explanations in OMDOC (3)

- **FMP (Formal mathematical property):** CIC terms are represented by a FMP elements that include MathML content

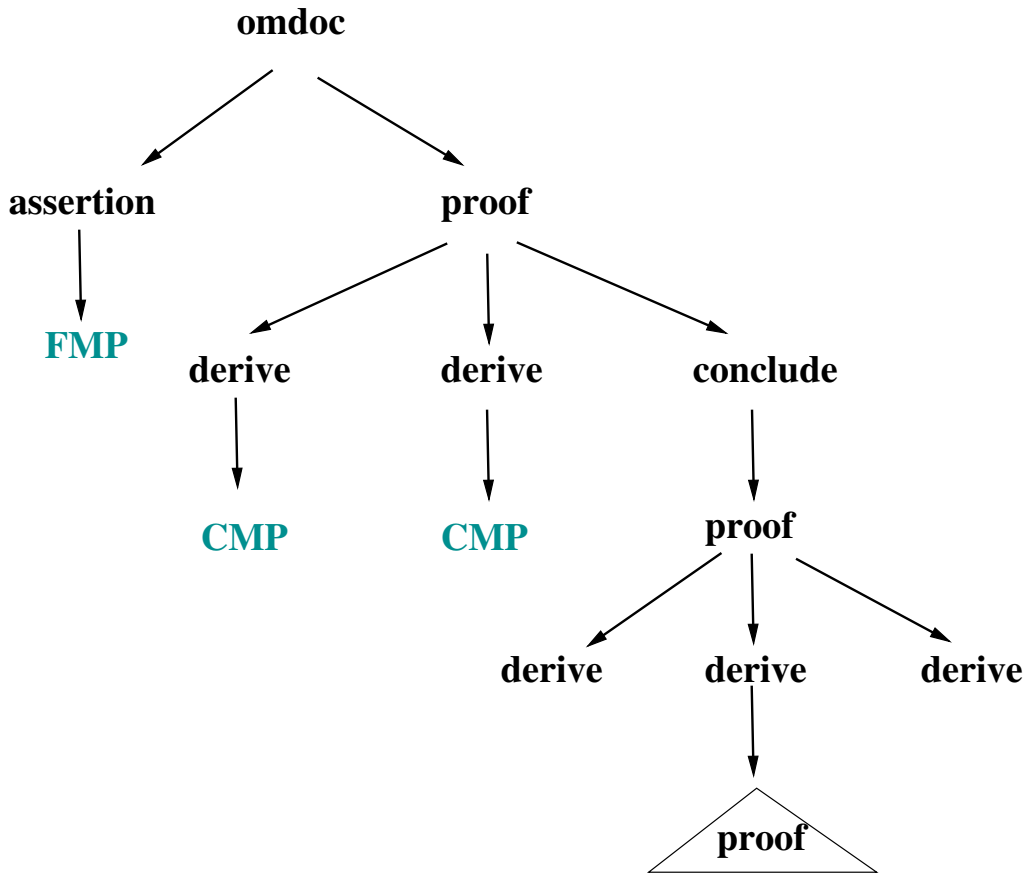
CIC TERM $\rightarrow$ MathML content

<omdoc:FMP > $...$ </omdoc:FMP>

- **CMP (commented mathematical property):** every explanation sentence is represented by CMP element

<omdoc:CMP xml:lang="en">
Let A, B, C, .., K and L be elements of .. such us ..
</omdoc:CMP>

General structure of proof explanations in OMDOC



From OMDOC to XHTML/MathML

- We apply XSLT rules on the OMDOC document to obtain an XHTML document organized with blocks.
  - <omdoc:assertion> ..</omdoc:assertion> → THEOREM: ...
  - <omdoc:proof> ..</omdoc:proof> → PROOF: ...
  - <omdoc:derive> ..</omdoc:derive> → <blockquote> ..</blockquote>
  - <omdoc:FMP> ..</omdoc:FMP> → MathML presentation (using MathML content to mathML presentation transformation)
- Customized (non standard) mathematical notations remain linear

(vec I J) = (vec A B)

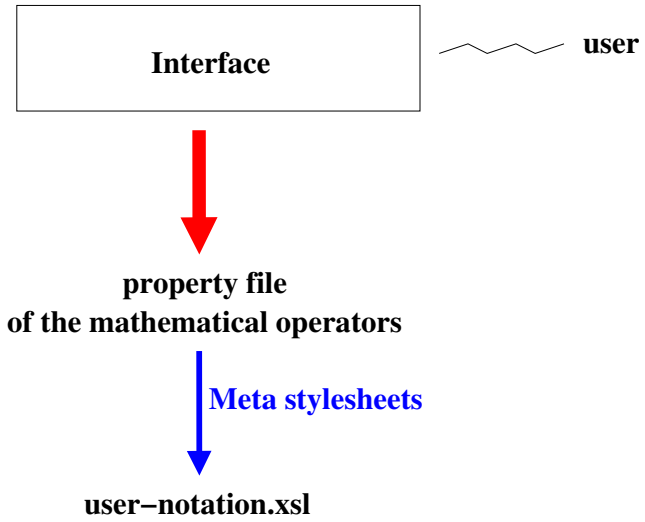
- The user needs to add the appropriate transformation rules for his customized notations (in file user-notation.xsl)

$$\overrightarrow{IJ} = \overrightarrow{AB}$$

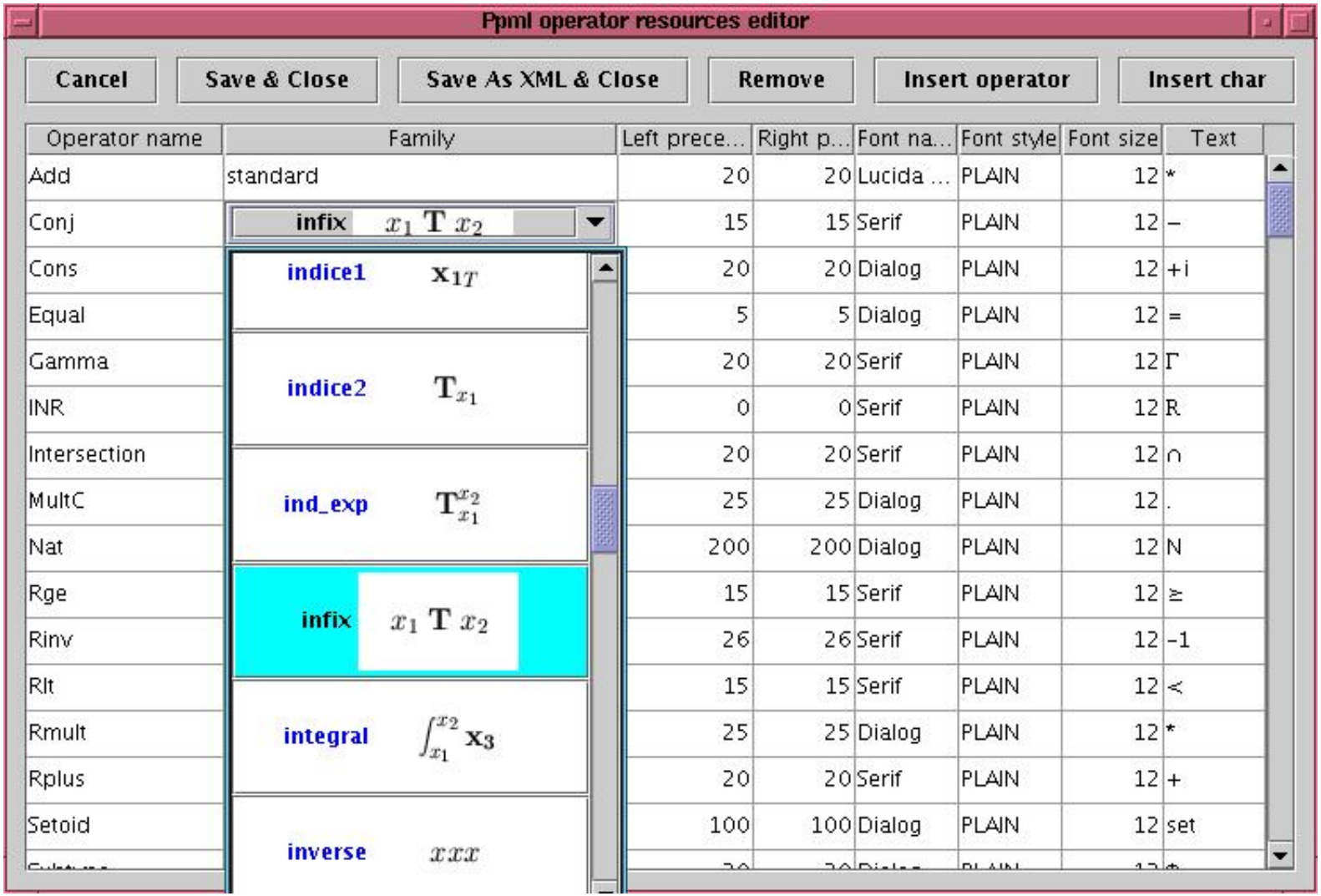
User-friendly customization of mathematical notations

So the user has not to write the XSL transformation rules

- We provide an interface to edit the display properties of the mathematical operators (this interface was originally integrated in Pcoq interface)



Mathematical formula property Editor



Generating transformation rules using the property file

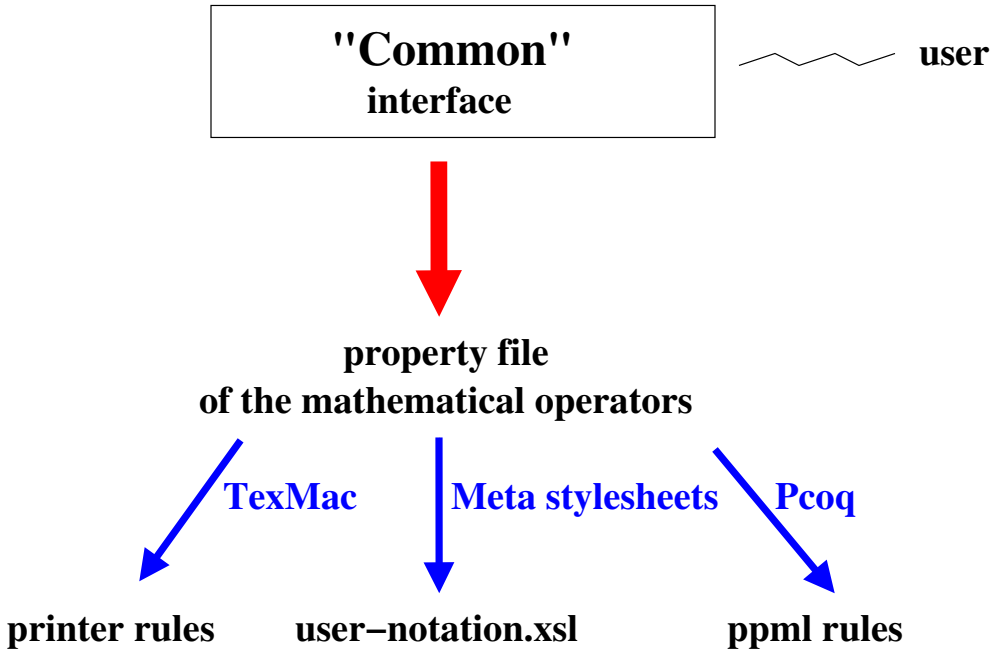
- Each mathematical operator has its own properties.
  - Family : prefix, infix, postfix, nroot, vector, constant ..
  - Text: operator symbol
- For each operator family, there is a Metastylesheet rule that generate the display rule

  Example : generated rule for an operator with family=”vector” Metastylesheet

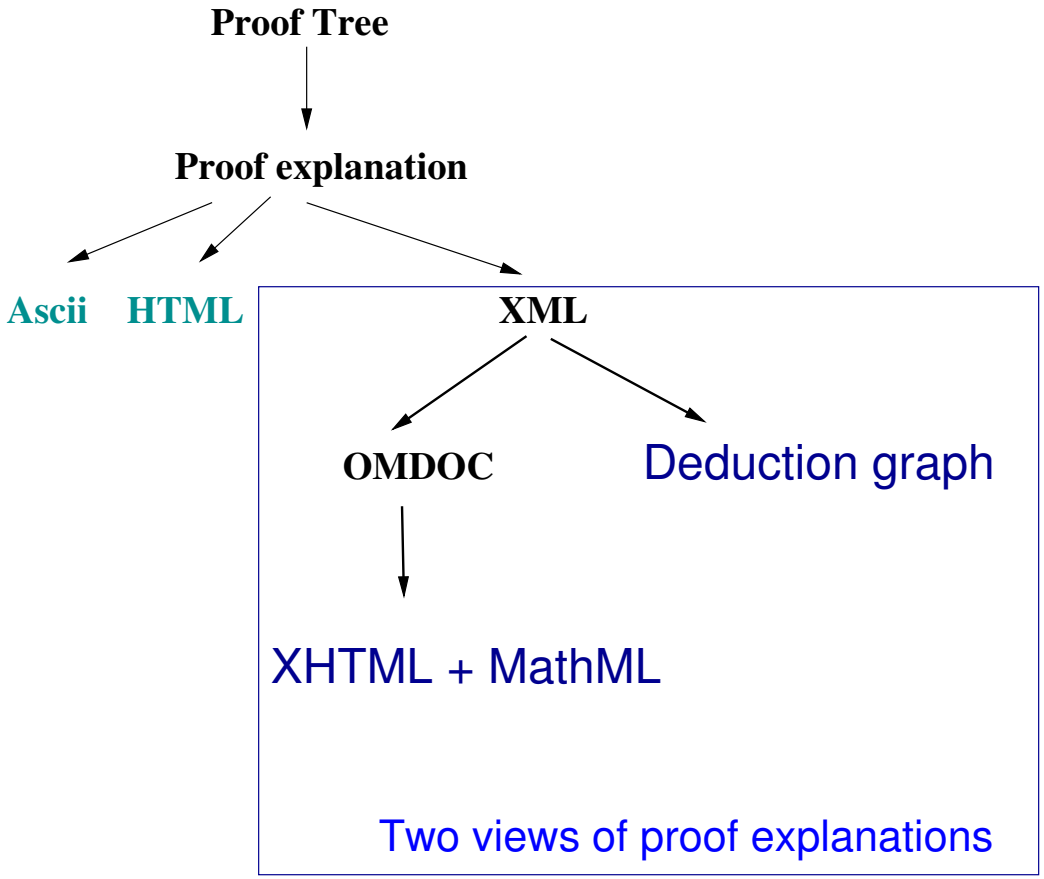
```
<xsl:template match="operator">
  <xsl:choose>
  <xsl:when test="*[1]=family[text()=’vector’]">
        XSLT Rule
  </xsl:when> ..
</xsl:choose>
</xsl:template>
```

  The XSLT rule specifies how to display an operator with “vector” family

Our aim: to provide a common interface with standard operator properties



Deduction Graph



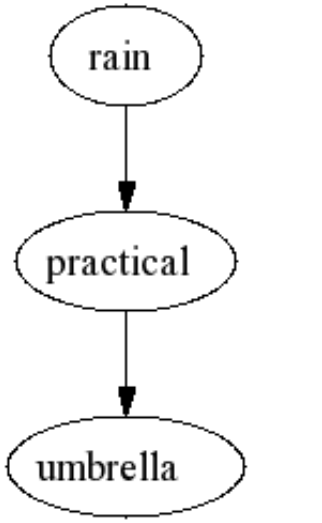
Graph extraction from proof explanation

- We apply XSLT rules on the XML proof explanation in order to obtain the graph description (in dot format)

  XML Proof explanations → graph in dot format (.dot)

- The dot format (node and edge definition)

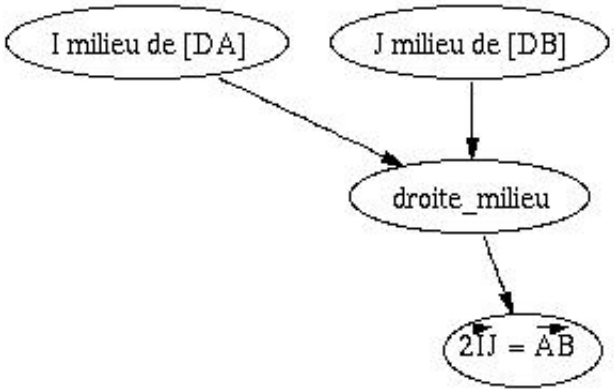
```
digraph ”Graph” {
”P1” [label=”rain”];
”P2” [label=”umbrella”];
”T1” [URL=”./T1.html”, label=”practical”];
”P1” → ”T1”;
”T1” → ”P2”;}
```



Graph by WebDot

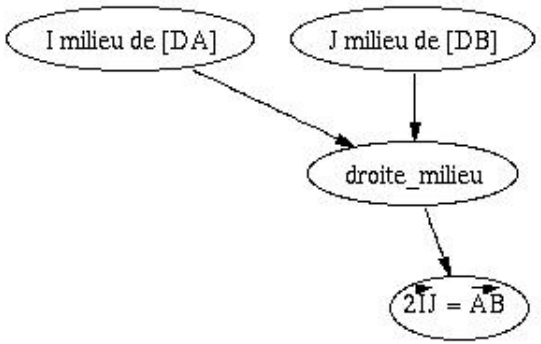
Graph extraction from proof explanation (2)

- How to view the dot graph ?
  - install WebDot, a WWW Graph Server which requires *graphviz* (a collection of tools for manipulating graph) and a httpd web server
  - output: active image, svg (mathematical formulas)



Transformation rules to obtain a graph from the proof explanation

- Each proof step (sentence) gives a subgraph (there is a rule for each sentence type)
  - create “hypothesis” nodes (h_i) and “conclusion” node (c)
  - possibly create “justification” node (j)
  - create edges $h_i \rightarrow c$ or ($h_i \rightarrow j$ and $j \rightarrow c$)

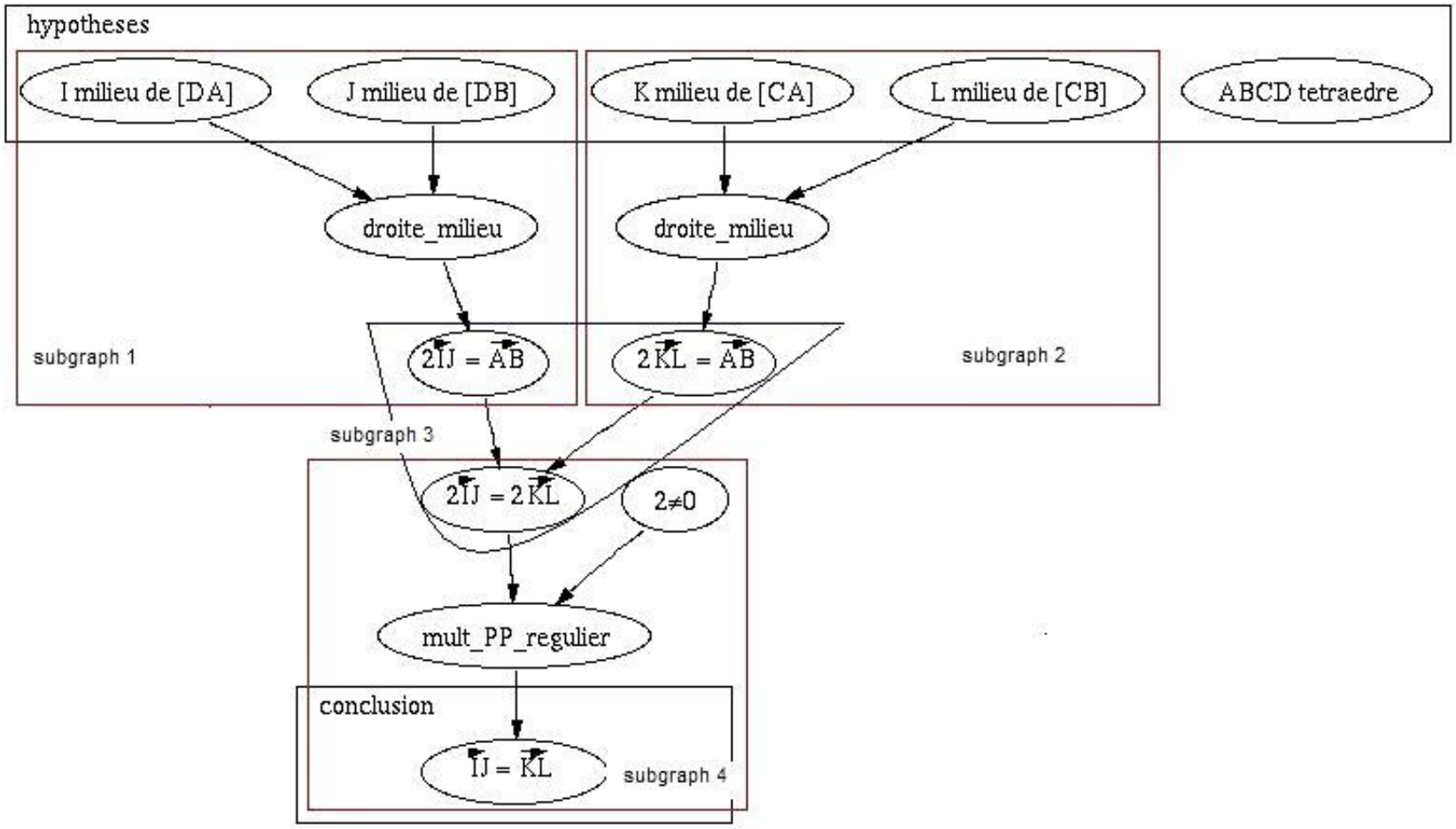


<By-using>
 <From-ARGA-we-deduce-ARGB>
 ..
 </From-ARGA-we-deduce-ARGB>
 <by-using-ARGA>..</by-using-ARGA>
</By-using>

Transformation rules to obtain a graph from the proof explanation (2)

- Each node has an identifier
  - For **hypothesis** and **conclusion**: nodes with the same content have the same identifier, which allows to bind subgraphs together
  - For **justification**: nodes with the same content may have different identifier, so the same justification can be repeated several times.

Example: geometry exercise - tetrahedron



Example: Proof by cases (Elim)

- The tetrahedron example only uses the first order logic (deduction $\Rightarrow$)
- In CIC, inductive definitions allow expressing proof by cases, proof by contradiction and proof by induction

Proof script in Coq

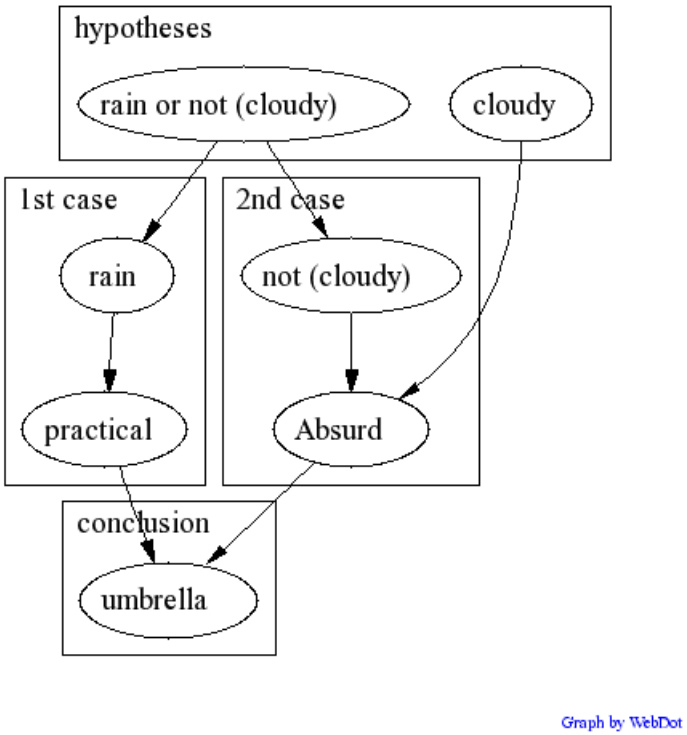
```
Parameters rain, cloudy, umbrella:Prop.
Axiom practical: rain -> umbrella.

Theorem meteo: (rain \/ ~ cloudy) -> cloudy -> umbrella.
Intros H H0.
Elim H; Intros.
Apply practical; Trivial.
Absurd cloudy; Trivial.
Qed.
```

Example: proof by cases (Elim) (2)

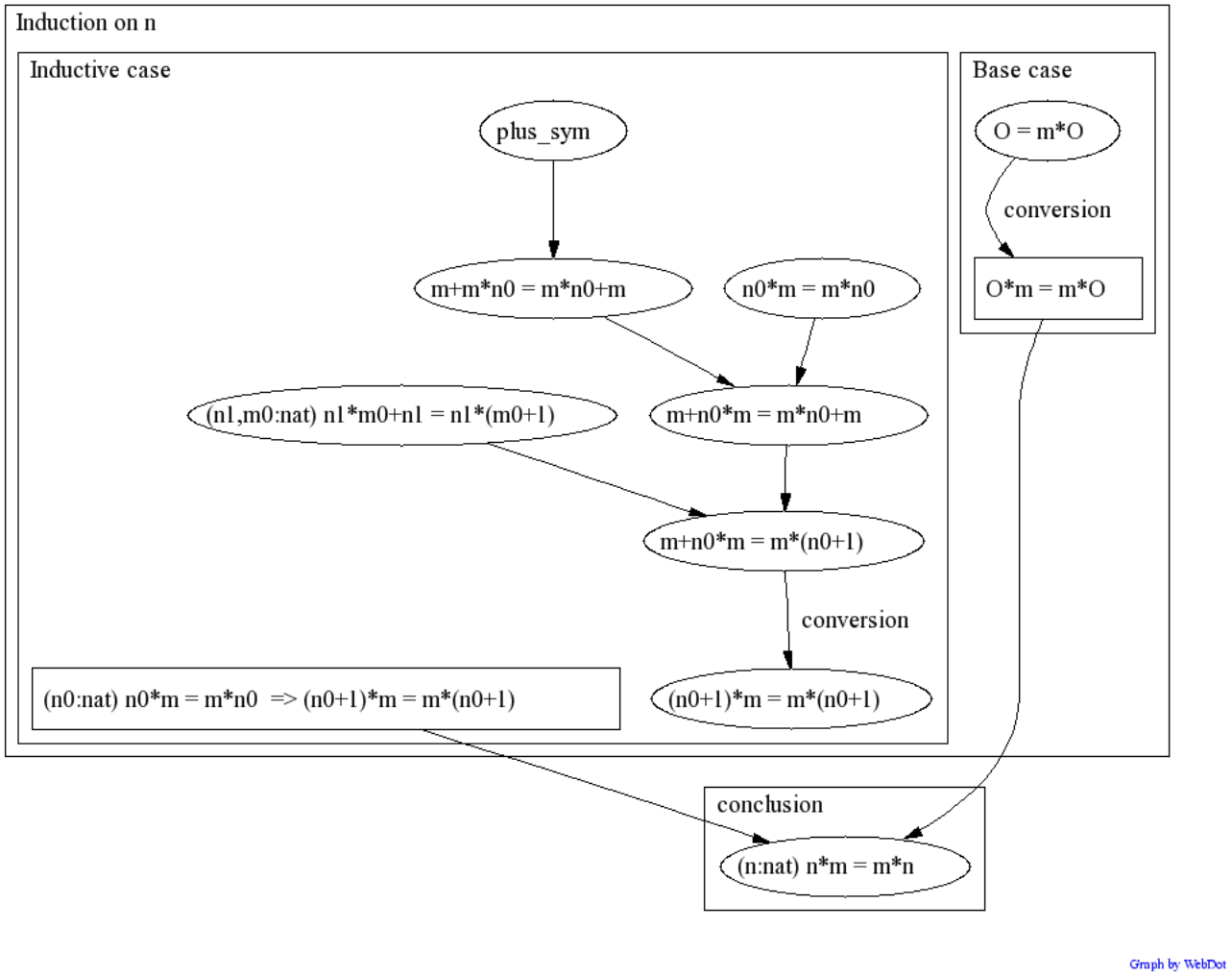
Let us suppose $rain \vee \backsim cloudy$ (H) and $cloudy$ (H0).
Let us prove $umbrella$:
We distinguish 2 cases for $rain \vee \backsim cloudy$:

1) Let us suppose $rain$ (H1).
From $rain$ we deduce $umbrella$ by $practical : rain \rightarrow umbrella$.
2) Let us suppose $\backsim cloudy$ (H1).
We get a contradiction with $\backsim cloudy$ and $cloudy$.

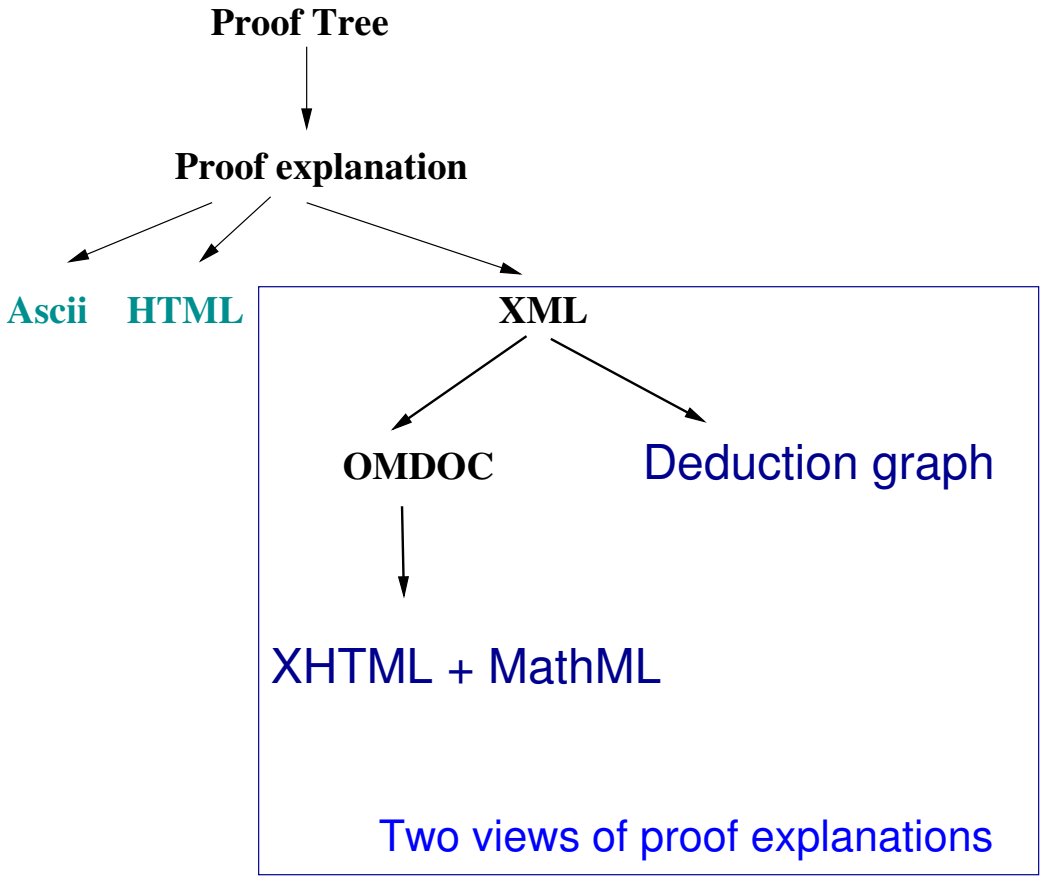
Hence, for every case we have proved $umbrella$.



Example: proof by induction



Conclusion


Proof Tree
Proof explanation
Ascii
HTML
XML
OMDOC
Deduction graph
XHTML + MathML
Two views of proof explanations