Argumentation & Compliance
MEDICA: an illustrative Example

Antonis Kakas

with

Nikos Spanoudakis and Pavlos Moraitits
University of Crete & University of Paris

Elena Constantinou and Adamos Koumi
University of Cyprus
Bridging Aim of Talk

Strengthen the link between argumentation and RuleML

Via

Example rather than Abstract Theory
Data Access/Sharing

**General Problem:**
- Decide **Level of Access** according to **user** and **current circumstances** in compliance with **legal and/or business, etc, requirements.**

**Challenge:**
- **Formalize and automate Legislation, Etc policies**
  - **Facilitate the management/compliance of policies**
- **Cognitive Interaction**
  - **Explain & “Persuade” the user (possibly via a dialogue)**
Example of MEDICA

Patient Data Access: Legislation

- There are 6 Access Levels (Read & Write):
  - Full Access
  - Partial Access
  - Read Only Access
  - Restricted Read Access
  - Suspended Access
  - No Access

- Decide Level of Access according to user and current circumstances, etc.
Options, e.g. different levels of medical access

Preferences
- Dynamic preferences over changing environment of the application
- Multi-Level preferences over different CONTEXTS in the application

General form of Preferences:
- “Generally, in SITUATION prefer Oi, but in particular CONTEXT prefer Oj.”

MEDICA Example:
- “Generally, deny access but for the {owner} give full access.”
- “Generally, allow full access to {owner} but when {critical tests} suspend access.”
Argumentation & Compliance in Gorgias

- **STEP 1**: Translate the law/policy to Scenario-Based Preferences

  - **Example**: «Generally, no one has access to medical files. But the owner CAN have full access.»
  - `<1, {}, access(Agn, DataID, no_access)>`
  - `<2, {owner(Agn)}, access(Agn, DataID, full_access)>`

- **STEP 2**: Automatic formalization and Generation of Execution Code in Gorgias
Medical Data Access: MEDICA

Law **138(I)/2001**: Personal Data Protection

Law **N. 1(I)/2005**: Patient Rights

**Example**: «**Generally, no one has access** to medical files. **But** the owner **CAN have** full access.»

- `<1, {}, access(Agn, DataID, no_access)>`
- `<2, {owner(Agn)}, access(Agn, DataID, full_access)>`

**Execution Code in Gorgias**:

```
rule(d1(Agn), access(Agn, DataID, full_access), []).
rule(d2(Agn), access(Agn, DataID, no_access), []).
rule(hpr21(Agn), prefer(d2(Agn), d1(Agn)), []).
rule(hpr12(Agn), prefer(d1(Agn), d2(Agn)), []) :- owner(Agn).
rule(hpr_12_21(Agn), prefer(hpr12(Agn), hpr21(Agn)), []).
```
Medical Data Access: MEDICA

- MEDICA: http://medica.cs.ucy.ac.cy

- Demo Online
  - user1
  - 12user12

- Note in demo:
  - Explanation
  - Hypothetical Reasoning
Medical Data Access: MEDICA

Law 138(I)/2001: Personal Data Protection
Law N. 1(I)/2005: Patient Rights

Example: «Generally, no one has access to medical files. But the owner CAN have full access.»

- <1, {}, access(Agn, DataID, no_access)>
- <2, {owner(Agn)}, access(Agn, DataID, full_access)>

FURTHER Example: «Generally, allow full access to {owner} but when {critical tests} suspend access.»

- <3, {owner(Agn),critical(DataID)}, access(Agn,DataID, suspend_access)>
Medical Data Access: MEDICA

Law 138(I)/2001: Personal Data Protection
Law N. 1(I)/2005: Patient Rights

«Generally, no one has access to medical files. But the doctor CAN have limited plus access for therapy/medical use.»

- <1, {}, access(Agn, DataID, no_access)>
- <2, {doctor(Agn), medical_use(DataID)}, access(Agn, DataID, limited_plus_access)>

«Generally, no one has access to medical files. But the doctor CAN have limited access for research purpose.»

- Scenarios-based preferences ...
Medical Data Access: Legislation

Law 138(I)/2001: Personal Data Protection
Law N. 1(I)/2005: Patient Rights

«Generally, no one has access to medical files. But owner’s family member CAN have limited plus.»

- <1, {}, access(Agn, DataID, no_access)>
- <2, {family_member(Agn)}, access(Agn, DataID, limited_access)>

«Generally, no one has access to medical files. But owner’s legal representative CAN have full access for personal use.»

Scenarios ...
Software Development via Argumentation (SoDA Methodology)

- **SoDA methodology** we model a (compliance) application by:
  - Considering **application scenarios** and stating the preferred/desired option(s) in each scenario: **scenario-based preferences**
  - Successively **refine** the scenarios and consider **combined** scenarios, restating at each case (level) the preferred option(s).

- **No programming or Cognitive Programming!**
  - Just **record** your expert application or user policy/knowledge.
Software Development via Argumentation (SoDA Methodology)

Programming = Authoring scenario-based preferences

(Call it Cognitive Programming)
This is the home page of the *Gorgias-B* tool for developing applications under preference-based argumentation with the use of a graphical user interface.

*Gorgias-B* supports the *SoDA* (Software Development for Argumentation) methodology, which guides the developer through his/her decision problem by an incremental refinement of application scenarios, where he/she considers the several (usually conflicting) alternatives and evaluates them by using generic or contextual knowledge.

The *Gorgias-B* tool is based on the *Gorgias* general argumentation framework.
Evaluation of Argumentation

- **Argumentation vs “Carefully crafted set of rules”**
  - Direct development from the domain expert!
  - No programming – Cognitive Programming!

- **Modularity of Approach**
  - **Claim**: Effort to accommodation new legislation in argumentation is comparable to effort to update the legal document.

- **Naturalness of argumentation via explanation and abductive/hypothetical reasoning**
  - Argumentation is native to human reasoning.
Further Reading/References

- MEDICA paper (from MEDICA website), 2017


- Gorgias-B website (http://gorgiasb.tuc.gr/)

- Other references on request ...
Gorgias

Greek Sophist c.485 — c.380 BCE
Extra (technical) Slides

Slides on Technical Translation of Scenario-based preferences to Gorgias framework & code
Medical Data Access: Legislation

Law 138(I)/2001: Personal Data Protection
Law N. 1(I)/2005: Patient Rights

Example: «Generally, no one has access to medical files. But the owner CAN have full access.»

- `<1, {}, access(Agn, DataID, no_access)>`
- `<2, {owner(Agn)}, access(Agn, DataID, full_access)>`

Example: «Generally, allow full access to {owner} but when {critical tests} suspend access.»

- `<3, {owner(Agn),critical(DataID)}, access(Agn,DataID, suspend_access)>`
MEDICA Decision Policy
(Expressed in GORGIAS pseudocode)

- **Object-level argument rules:**
  
  \[
  \begin{align*}
  r1(Agn,Data) & : \text{no\_access}(Agn,Data) \leftarrow \text{true} \\
  r2(Agn,Data) & : \text{full\_access}(Agn,Data) \leftarrow \text{true} \\
  r3(Agn,Data) & : \text{supsend\_access}(Agn,Data) \leftarrow \text{true}
  \end{align*}
  \]

- **Priority argument rules**
  
  - Default Policy - Scenario 1
  - Generally, no access:
    
    \[
    \begin{align*}
    R12(Agn,Data) & : r1(Agn,Data) > r2(Agn,Data) \leftarrow \text{true} \\
    R13(Agn,Data) & : r1(Agn,Data) > r3(Agn,Data) \leftarrow \text{true}
    \end{align*}
    \]
  
  - Special - Contextual- Priority: Scenario 2
  - Generally, full access to owner
    
    \[
    \begin{align*}
    R21(Agn,Data) & : r2(Agn,Data) > r1(Agn,Data) \leftarrow \text{owner}(Agn,Data) \\
    C21(Agn,Data) & : R21(Agn,Data) > R12(Agn,Data) \leftarrow \text{true}
    \end{align*}
    \]

    \[
    \begin{align*}
    R23(Agn,Data) & : r2(Agn,Data) > r3(Agn,Data) \leftarrow \text{owner}(Agn,Data)
    \end{align*}
    \]
MEDICA Decision Policy
(Expressed in GORGIAS pseudocode)

- **Object-level argument rules:**
  
  \[ r1(Agn, Data): \text{no\_access}(Agn, Data) \rightarrow \text{true} \]
  \[ r2(Agn, Data): \text{full\_access}(Agn, Data) \rightarrow \text{true} \]
  \[ r3(Agn, Data): \text{suspend\_access}(Agn, Data) \rightarrow \text{true} \]

- **Priority argument rules**
  
  - Default Policy - Scenario 1
    - Generally, no access:
    - \[ R13(Agn, Data): r1(Agn, Data) > r3(Agn, Data) \rightarrow \text{true} \]
  
  - Special - Contextual-Priority: Scenario 3
    - Generally, suspend access to owner when critical
    - \[ R23(Agn, Data): r2(Agn, Data) > r3(Agn, Data) \rightarrow \text{owner}(Agn, Data) \]
      - \[ R32(Agn, Data): r3(Agn, Data) > r2(Agn, Data) \rightarrow \text{critical}(Data) \]
      - \[ C32(Agn, Data): R32(Agn, Data) > R23(Agn, Data) \rightarrow \text{true} \]
    - \[ R31(Agn, Data): r3(Agn, Data) > r1(Agn, Data) \rightarrow \text{owner}(Agn, Data), \text{critical}(Data). \]
MEDICA: Argumentation in Scenarios

- \(<1, \{\}, \text{no\_access}(\text{Agn,Data})>\)
  - A1=\{r1(\text{Agn,Data})\} argument supports option no\_access.
  - A2=\{r2(\text{Agn,Data})\} argument supports option full\_access.
  - A3=\{r3(\text{Agn,Data})\} argument supports option suspend\_access.

- A1'=\{r1(\text{Agn,Data}), \text{R12}(\text{Agn,Data}), \text{R13}(\text{Agn,Data}), \}\) strengthens A1
  - A1' attacks A2 and A3 but they do not attack A1'
  - Also A2 or A3 cannot be strengthened (by any applicable priority rule)

- Hence A2 or A3 cannot be made admissible
- Hence sceptical decision: no\_access (from A1')
MEDICA: Argumentation in Scenarios

- `<2, {owner(Agn)}, access(Agn, DataID, full_access)>`

- $A1' = \{ r1(Agn,Data), R12(Agn,Data), R13(Agn,Data), \} \text{ strengthens } A1$
  - $A1'$ attacks $A2$ and $A3$ but they do not attack $A1'$

- $A2' = \{ r2(Agn,Data), R21(Agn,Data), R23(Agn,Data), \} \text{ strengthens } A2$
  - $A2'$ attacks $A1$ and $A3$ but they do not attack $A2'$
  - $A2'$ attacks $A1'$ and vice-versa

- $A2'' = A2' \cup \{ C21(Agn,Data) \}$
  - $A2''$ attacks $A1'$ but not vice-versa.
  - (also attacks $A1$ and $A3$ – not shown)

- Hence, only $A2''$ admissible: full_access.