

Argumentation in AI

ProCLAIM: An Argument-Based Model for Deliberating Over Action Proposal

Pancho Tolchinsky
Knowledge Engineering & Machine Learning Group
Universitat Politècnica de Catalunya

Universitat Politècnica de Catalunya, Barcelona
12/12/2007



Presentation Outline

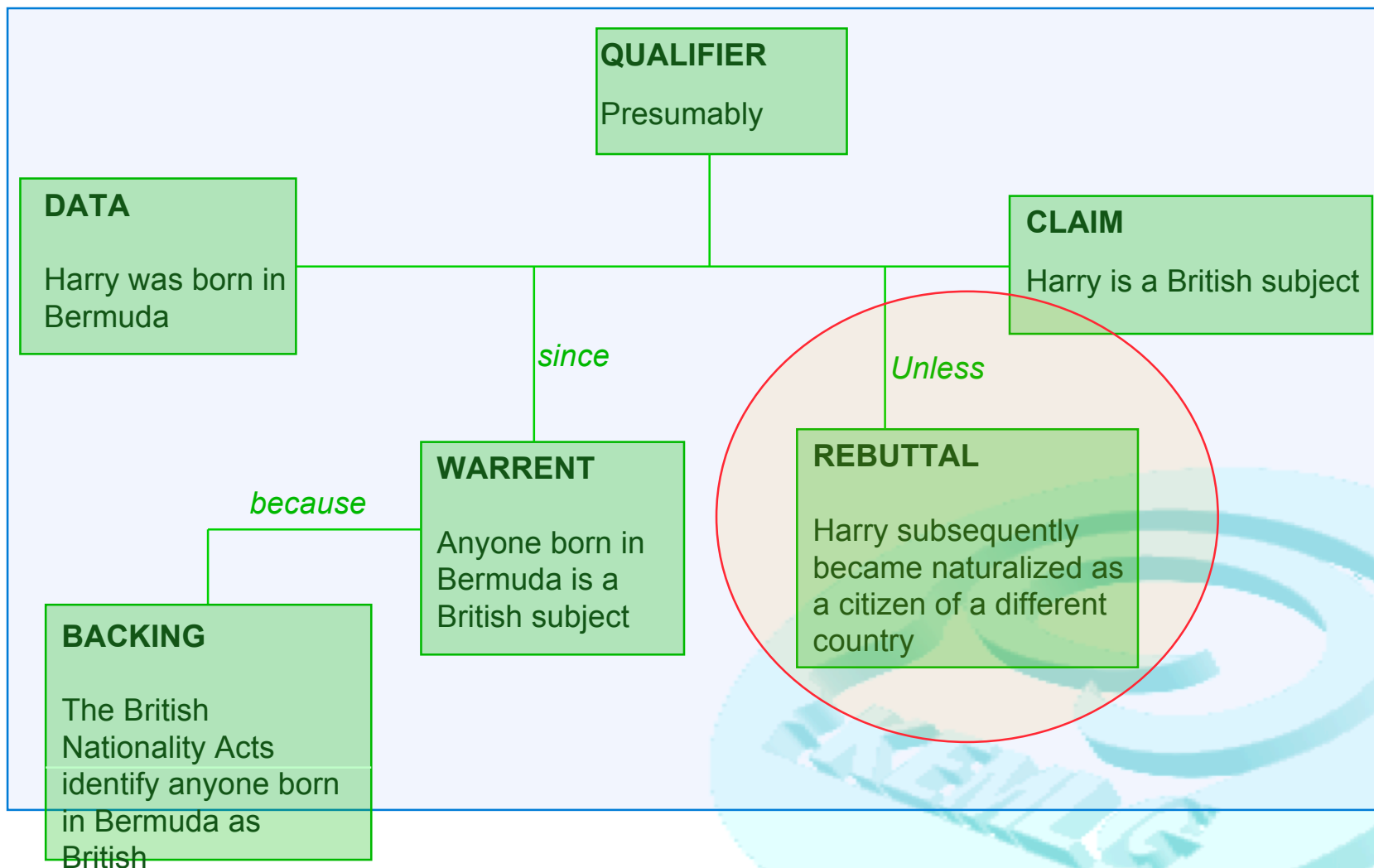
- Arguments and Argumentation
- Origins in AI
- Formal Arguments, Attack Relation, Argument Framework, Argument Acceptability
- Argument Schemes
- ProCALIM: An Argument-Based Model for Deliberating Over Action Proposal

Arguments and Argumentation

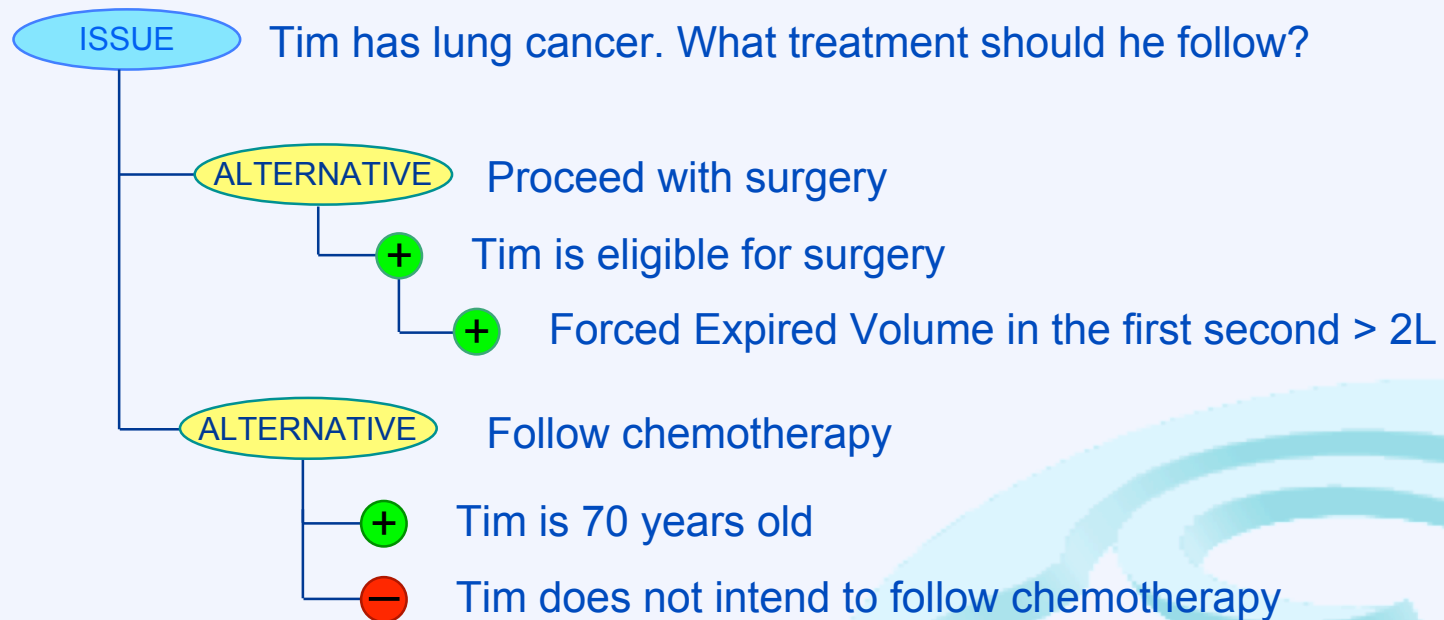
- **Argument:** Reasons to believe or to act
- **Argumentation:** The process of creating arguments for and against competing claims



Origins – Toulmin (1958)



Origins – IBIS: Kuntz and Rittle 1970



Origins: Formal and Informal Argumentation

- **Informal Argumentation:** Disagreement between two agents (audience) may not be due to a logical misconception (Rhetorics – *Argument Schemes*)
- **Formal Argumentation:** Limitation of classical logics for Commonsense Reasoning. Non-monotonic Logics are proposed. The idea of defeasible arguments together with the notion of argument acceptability is introduced.

Formal Representation of an Argument

- Define a Logical Language with a defeasible entailment \sim
- $\langle \textit{support}, \textit{claim} \rangle$ is an argument if:
 - $\textit{support} \sim \textit{claim}$
 - $\textit{support} : L_1, \dots, L_n$ is minimum w.r.t. the inclusion (not the case that $L_1, \dots, L_{n-1} \sim \textit{claim}$)

Conflict between Arguments

- A: $\langle \{a, a \rightarrow b\}, b \rangle$
- B: $\langle \{\neg b\}, \neg b \rangle$
- B attacks A
- A Attacks B
- A and B rebut attack each other

Conflict between Arguments

- A: $\langle \{a, a \rightarrow b\}, b \rangle$
- B: $\langle \{\neg a\}, \neg a \rangle$
- B attacks A
- A2 = $\langle \{\neg a\}, \neg a \rangle$

Conflict between Arguments

- A: $\langle \{a, a \rightarrow b\}, b \rangle$
- B: $\langle \{ \neg(a \rightarrow b) \}, \neg(a \rightarrow b) \rangle$
- B attacks A (Undercuts)

Dung Argument Framework (1995)

- Set of arguments Arg , and an attack relation R
 $\langle Arg, R \rangle$
- $A: \langle \{a, a \rightarrow b\}, b \rangle$
- $B: \langle \{\neg(a \rightarrow b)\}, \neg(a \rightarrow b) \rangle$
- $A, B \text{ in } Arg; (B, A) \text{ in } R$

$B \longrightarrow A$

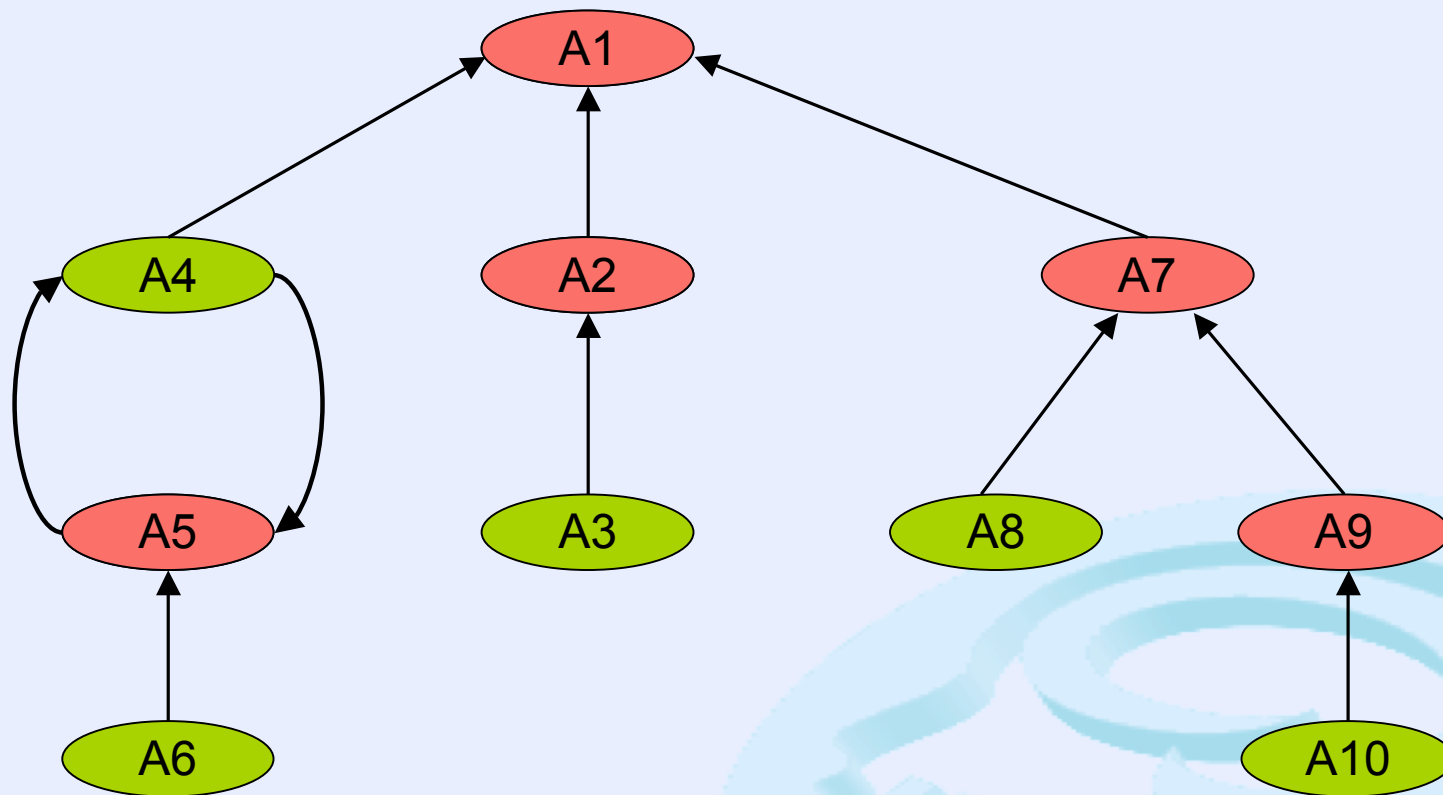
Dung Argument Framework (1995)

- Set of arguments Arg , and an attack relation R
 $\langle Arg, R \rangle$

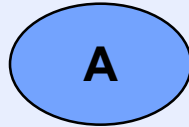
- $A, B \text{ in } Arg; (B, A) \text{ in } R$

$B \longrightarrow A$

Argument Acceptability



Argument Acceptability / Semantics



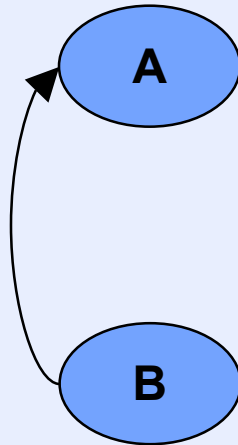
Grounded Semantics: $\{A\}$

Preferred Semantics: $\{A\}$

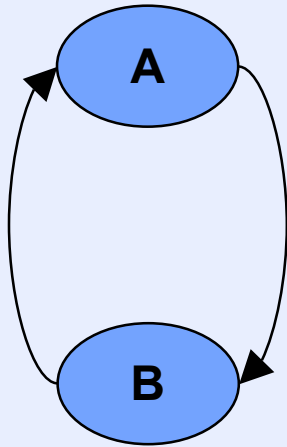
Argument Acceptability / Semantics

Grounded Semantics: {B}

Preferred Semantics: {B}



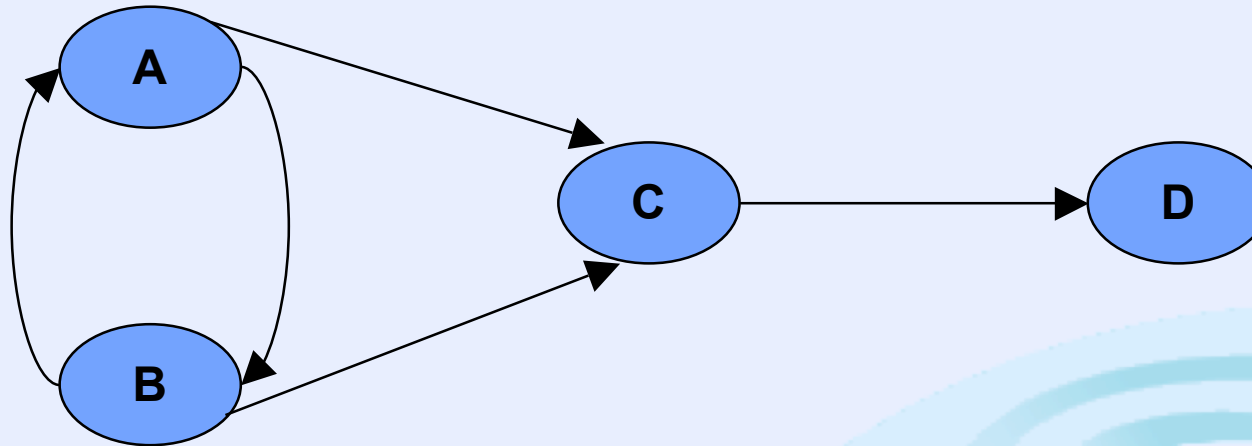
Argument Acceptability / Semantics



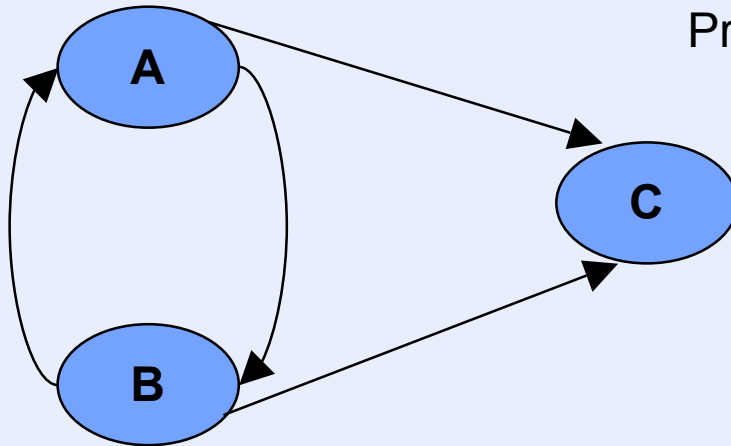
Grounded Semantics: $\{\}$

Preferred Semantics: $\{A\}, \{B\}$

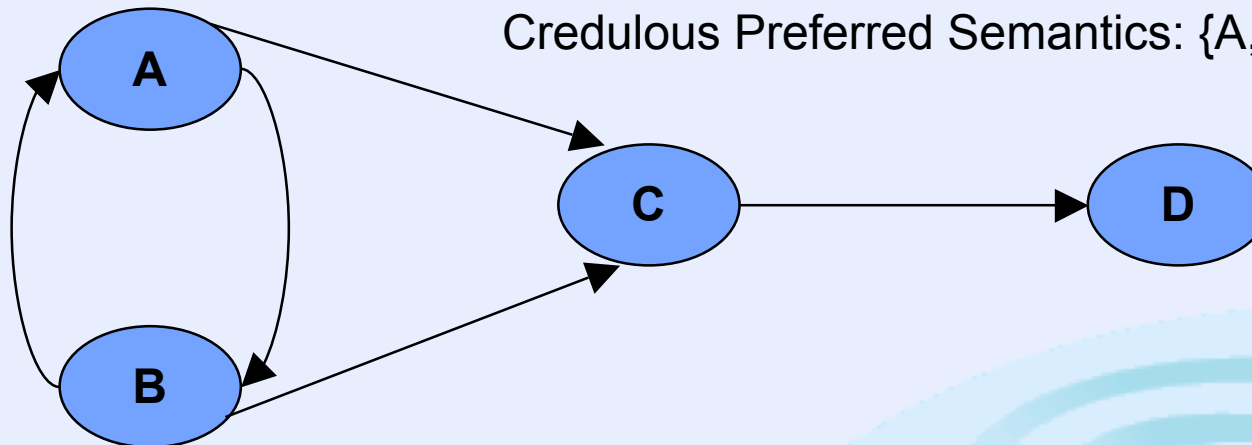
Argument Acceptability / Semantics



Argument Acceptability / Semantics

Grounded Semantics: $\{\}$ Preferred Semantics: $\{A\}, \{B\}$ 

Argument Acceptability / Semantics

Grounded Semantics: $\{\}$ Credulous Preferred Semantics: $\{A,D\},\{B,D\}$ Skeptical Preferred Semantics: $\{D\}$

Argument Schemes and Critical Questions

- Argument schemes are used to classify different types of argument that embody stereotypical patterns of reasoning:

Expert Opinion Argument Scheme:

Source **E** is an expert in the subject domain **S** containing proposition **A**

E asserts that proposition **A** in domain **S** is true

Therefore, **A** may plausibly be taken as true

Argument Schemes and Critical Questions

- Argument schemes have a dialectical aspect in that they come with a set of critical questions, which enumerate ways of challenging arguments instantiating the schemes.

Critical Questions:

- 📁👉 How credible is source **E** as an expert source?
- 📄👉 Is **E** an expert in the field that **A** is in?
- 📄👉 Does **E**'s testimony imply **A**?
- 📄👉 Is **E** reliable?
- 📄👉 Is **A** consistent with the testimony of other experts?
- 🕒👉 Is **A** supported by evidence?

Argument Schemes and Critical Questions

- Appeal to Witness Testimony
- Argument from Analogy
- Argument from Precedent
- Argument from Sign
- Practical Reasoning



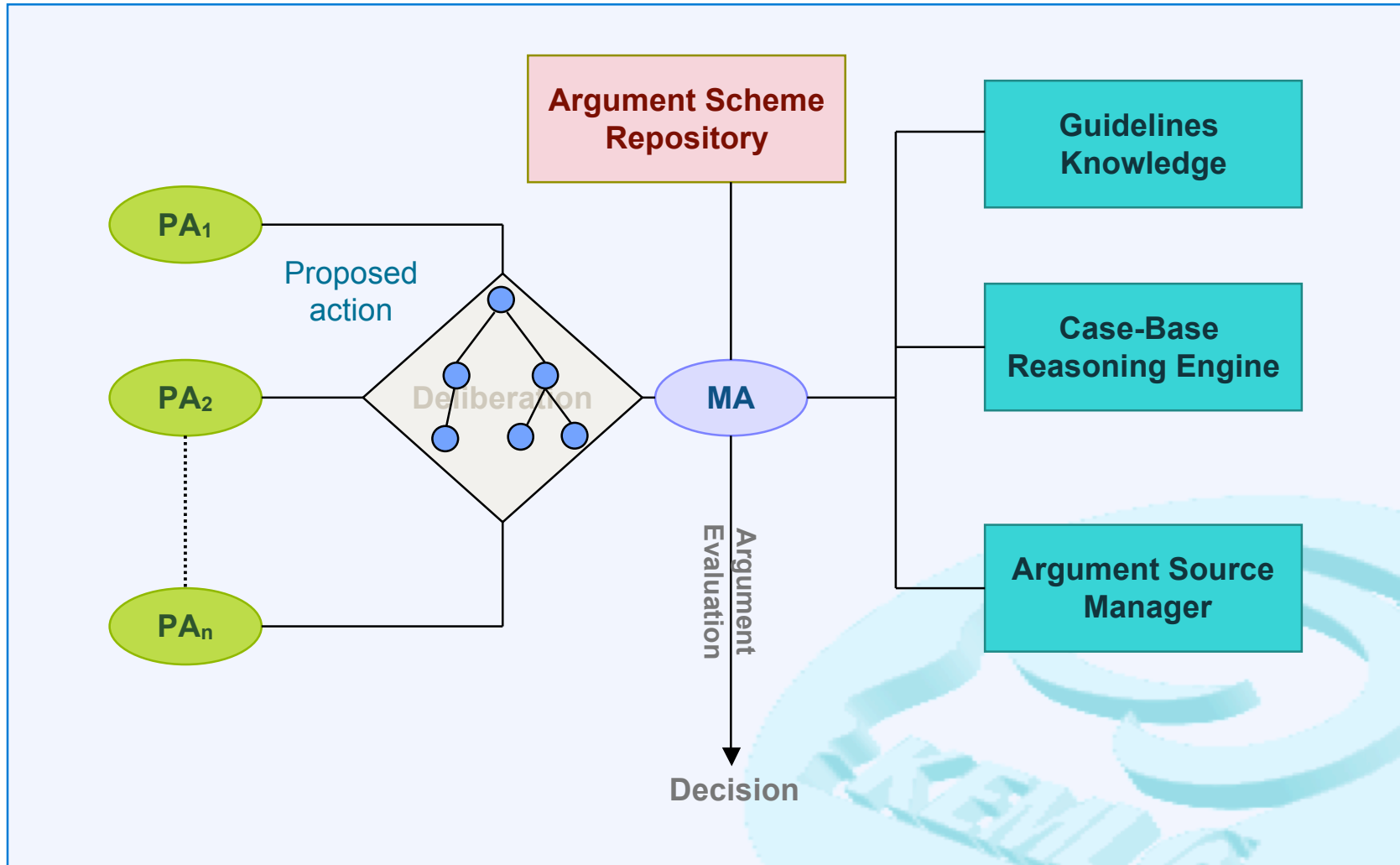
Presentation Outline

- *ProCLAIM*: An Argument-Based Framework for deliberating over the appropriateness of a proposed course of action
- Human Organ Transplantation, a Working Scenario
- Deliberating over Action Proposals using Argument Schemes and Critical Questions
 - A protocol-based exchange of arguments
 - *Towards constructing a Repository of Argument Schemes*
- Argumentation and Case-Based Reasoning

Introducing the *ProCLAIM* model

- Argument-based Framework
- Collaborative Decision Making
 - Provide an environment for agents to argue over the appropriateness of a proposed action.
whether it is justified to undertake a proposed action
- Safety-Critical domains
 - Guidelines Knowledge
 - *Wrong* Decisions/Actions may be catastrophic
 - Actions may be appropriate despite violating Guidelines

The *ProCLAIM* Architecture



PA Proponent Agent

UPC - 22/05/2007

MA Mediator Agent

Human Organ Transplantation

- **Human Organ Transplantation** is the only effective therapy for many life-threatening diseases.
- Commonplace medical event.
- Disparity between the demand for and the supply of organs for transplantation.
- Great percentage of human organs are discarded as being considered non-viable.

15 % livers 20% kidneys 60% hearts 85% lungs 95% pancreas

Human Organ Transplantation :: An example

Donor's data: ...smoking history.... no COPD....

Is the Lung Viable?



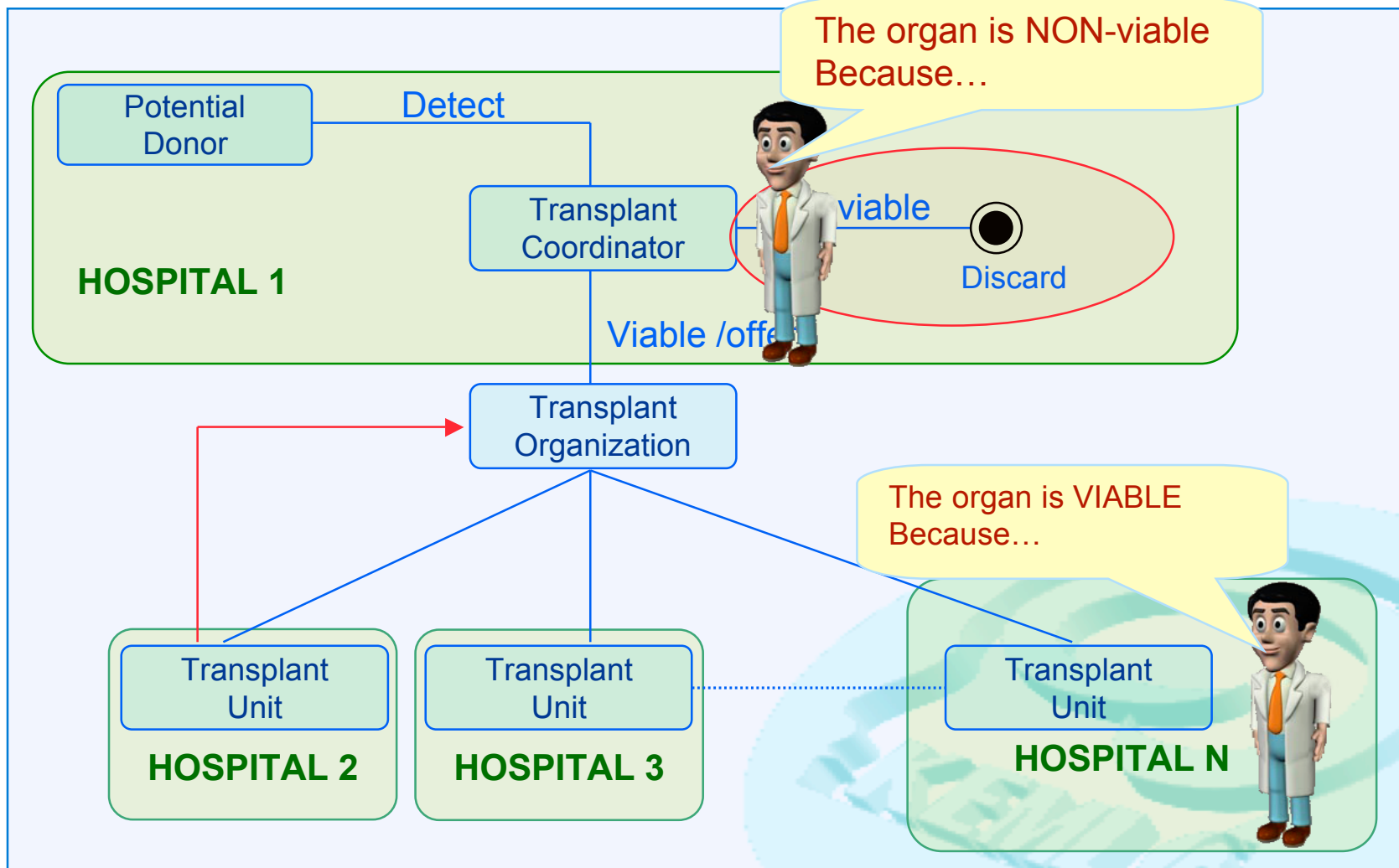
The Lung is non-viable because the donor has a smoking history.

Smoking history is not a contraindication, since the donor did not have any COPD

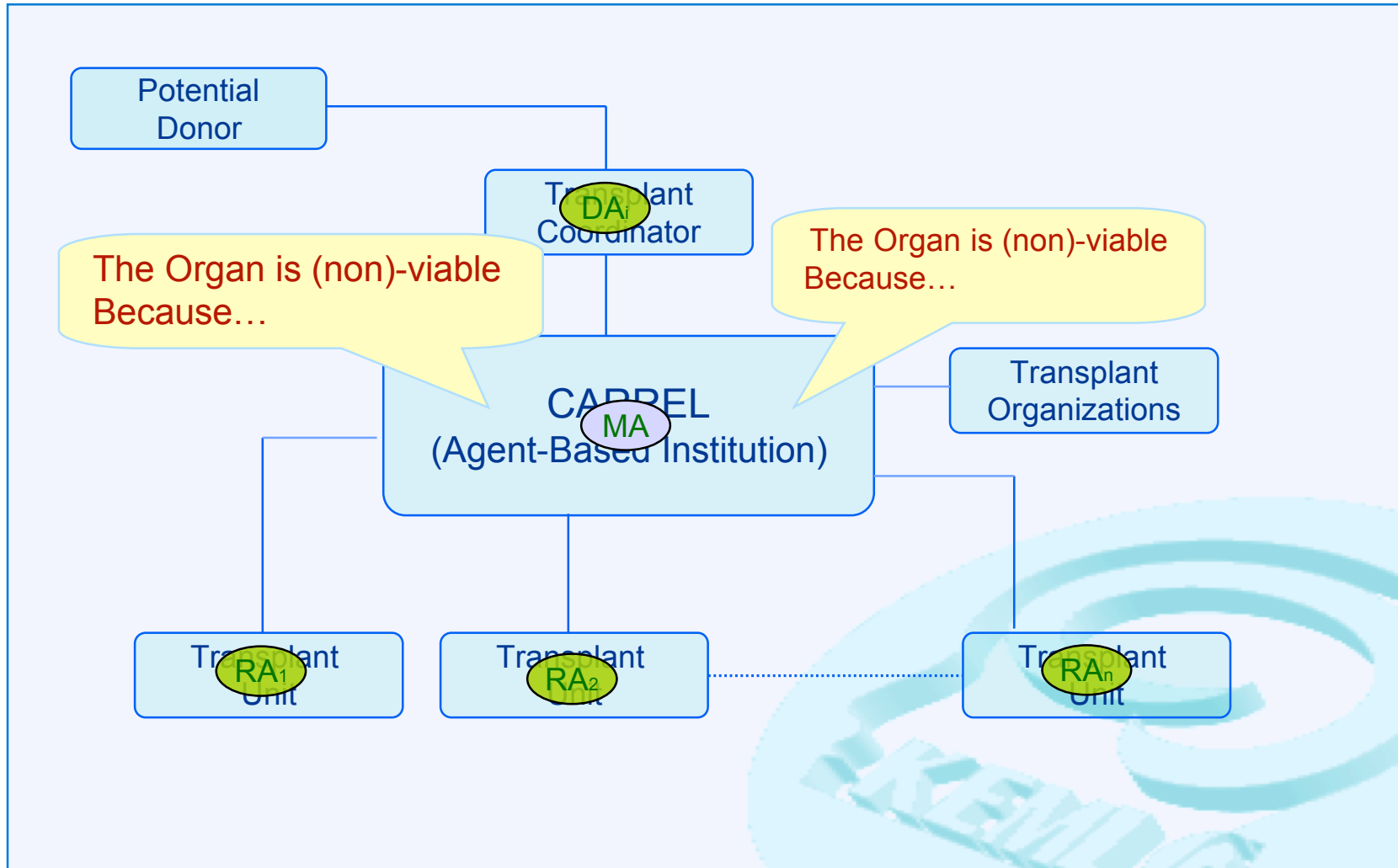


Any kind of smoking history could be acceptable for donors unless there is a COPD of more than 20-30 pack-year.

Human Organ Selection Process



Organ Selection Process managed by CARREL

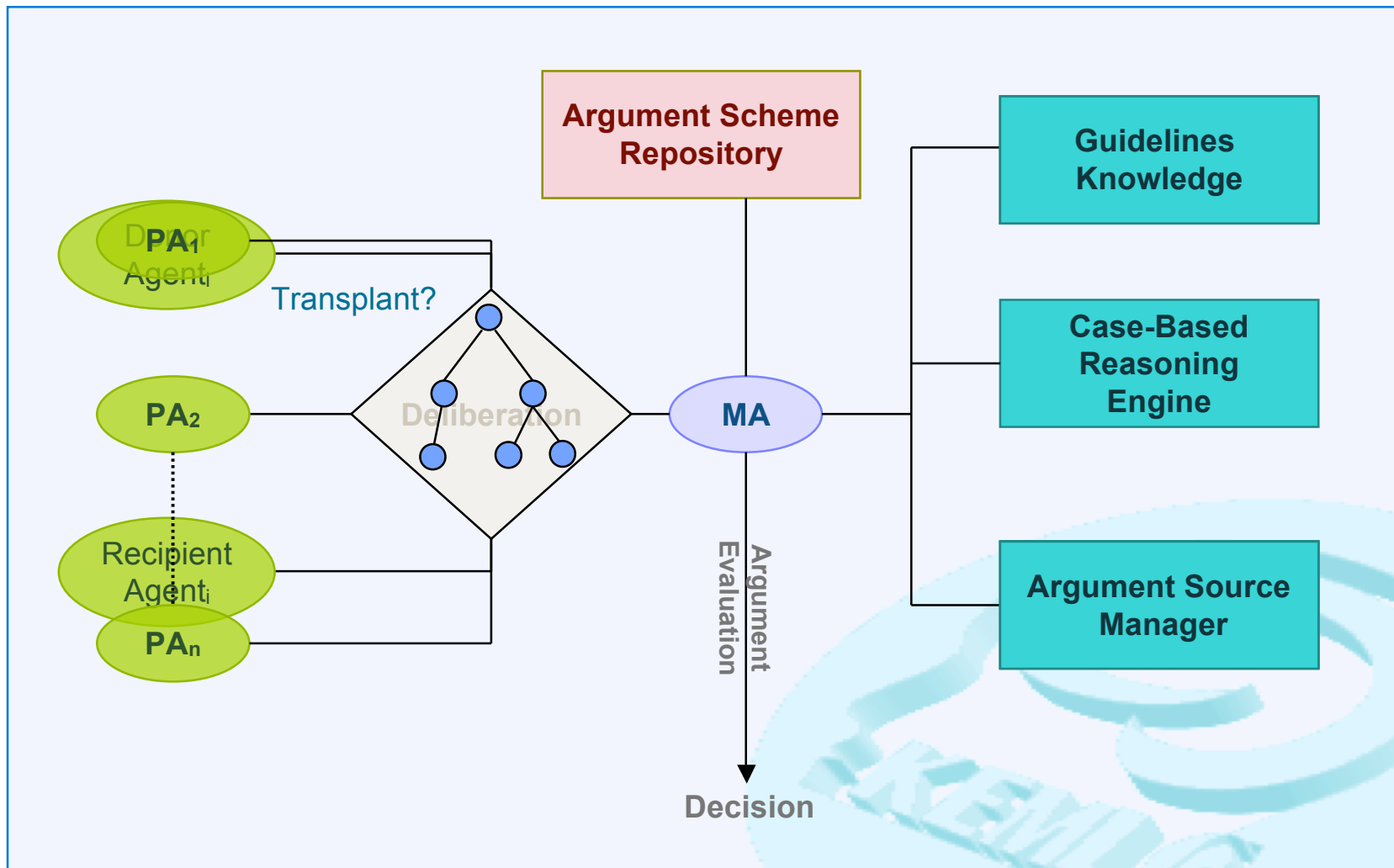


DA Donor Agent

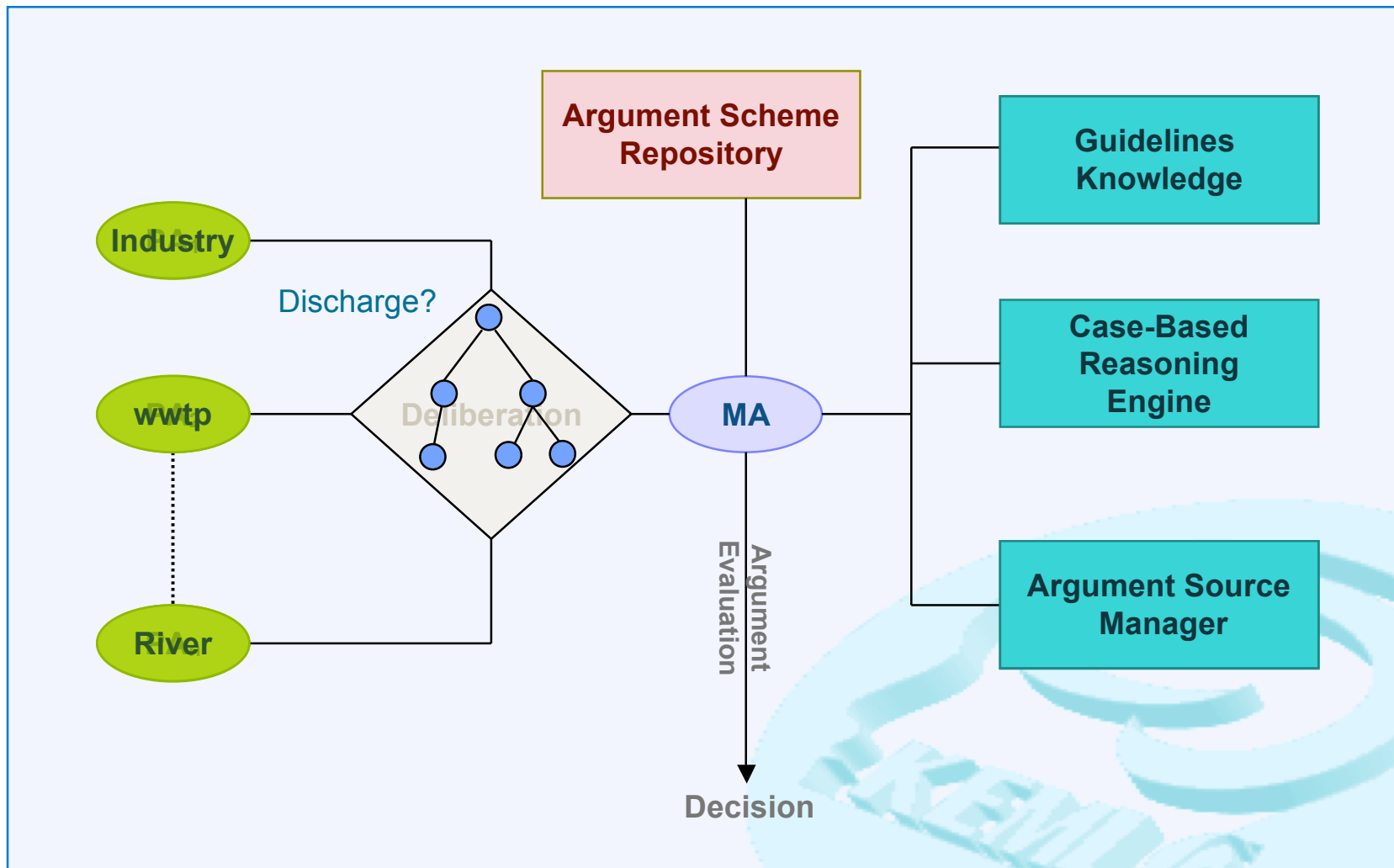
UPC - 22/05/2007

RA Recipient Agent

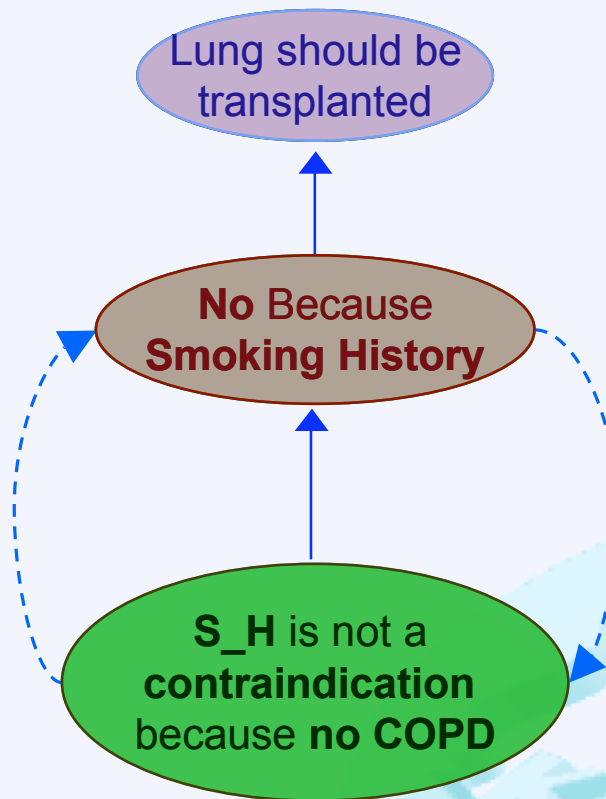
Architecture of the Transplant Scenario



Should a Industrial Waste be Discharged?



Should the Lung be Transplanted?



Dung, 1995

Argument Schemes and Critical Questions as a Protocol

VS: Viability Scheme:

Donor **D** of organ **O** is available
 And no contraindications are known for
 donating **O** to Recipient **R**
 Therefore Organ **O** is viable

VS_CQ1: Does donor **D** has a
 contraindication **C** for
 transplanting organ **O**
 into recipient **R**?

NVS: Non-Viability Scheme:

Donor **D** of organ **O** had condition **C**
 And **C** is a contraindication for donating **O**
 Therefore **O** is non-viable

NVS_CQ2: Is **C** a contraindication?

NDAS : No Disease Associated Scheme:

If donor **D** did not have the disease **E**
 that is a manifestation of **C**
 Then **C** is not a contraindication for donating

O

GFS : Graft Failure Scheme:

When transplanting organ **O** from donor **D** with condition
C to a recipient **R**, **R** may end up having a *Graft Failure*
 Therefore, **C** is a contraindication for transplanting **O** into **R**

Argument Schemes and Critical Questions as a Protocol

VS: Viability Scheme:

Donor D of organ O is available
 And no contraindications are known for
 donating O to Recipient R
 Therefore Organ O is viable

NVS: Non-Viability Scheme:

Donor D of organ O had condition C
 And C is a contraindication for donating O
 Therefore O is non-viable

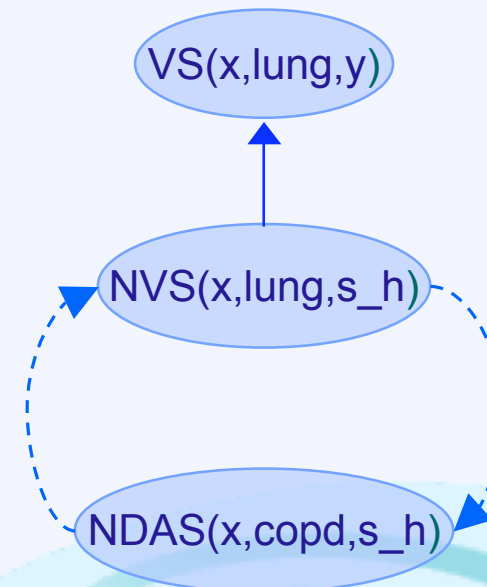
NDAS : No Disease Associated Scheme:

If donor D did not have the disease E
 that is a manifestation of C
 Then C is not a contraindication for donating

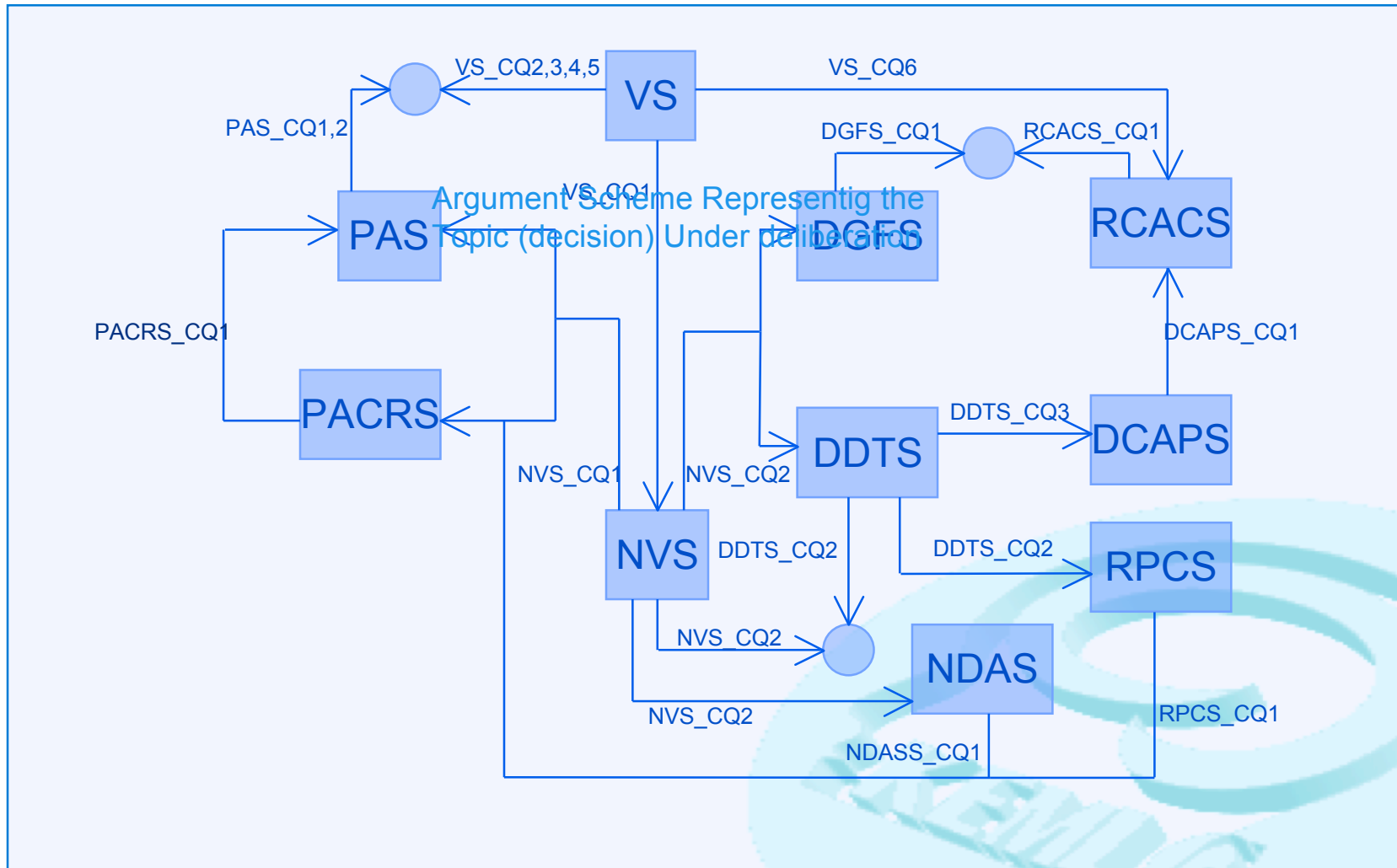
O

GFS : Graft Failure Scheme:

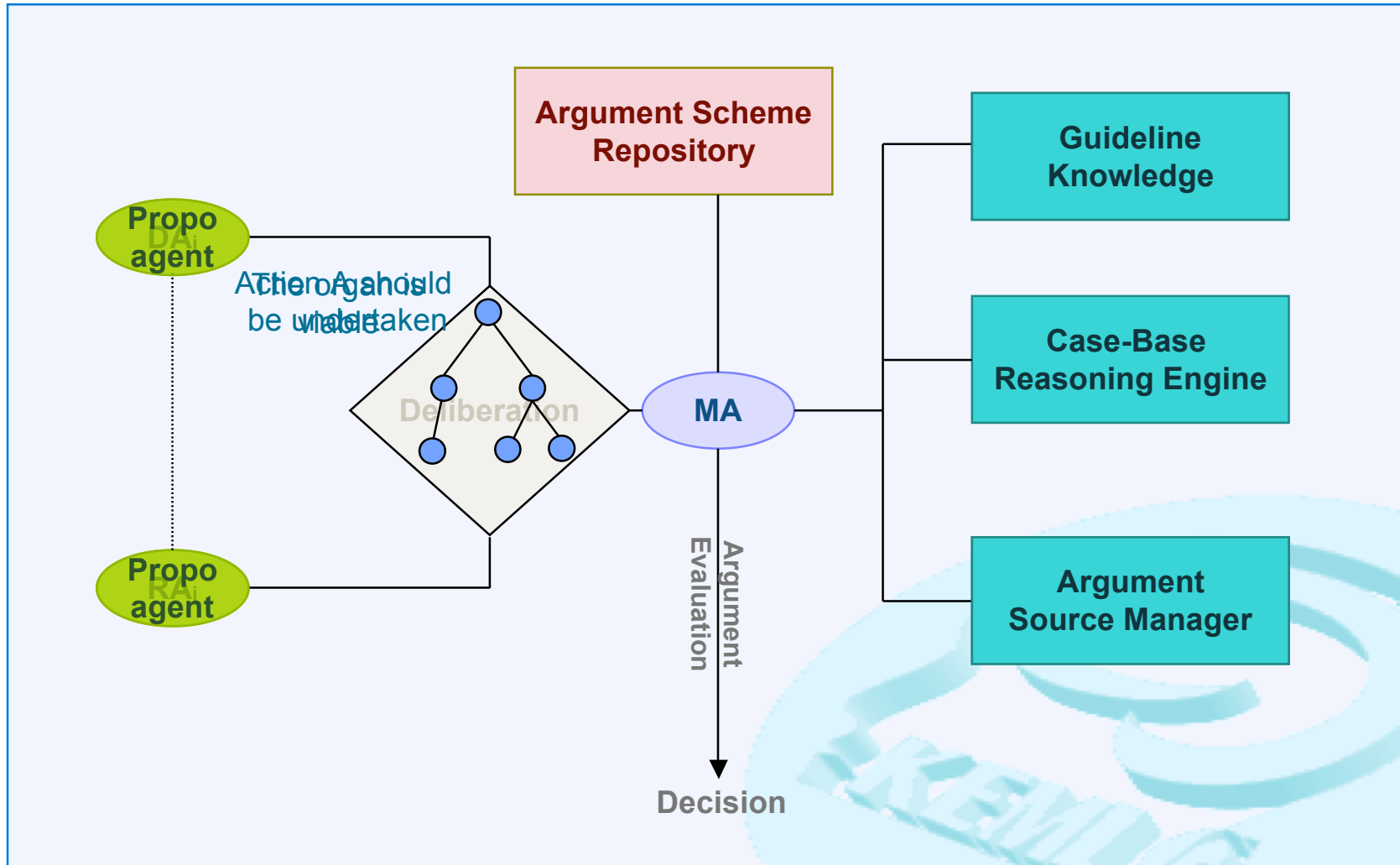
When transplanting organ O from donor D with condition
 C to a recipient R , R may end up having a Graft Failure
 Therefore, C is a contraindication for transplanting O into R



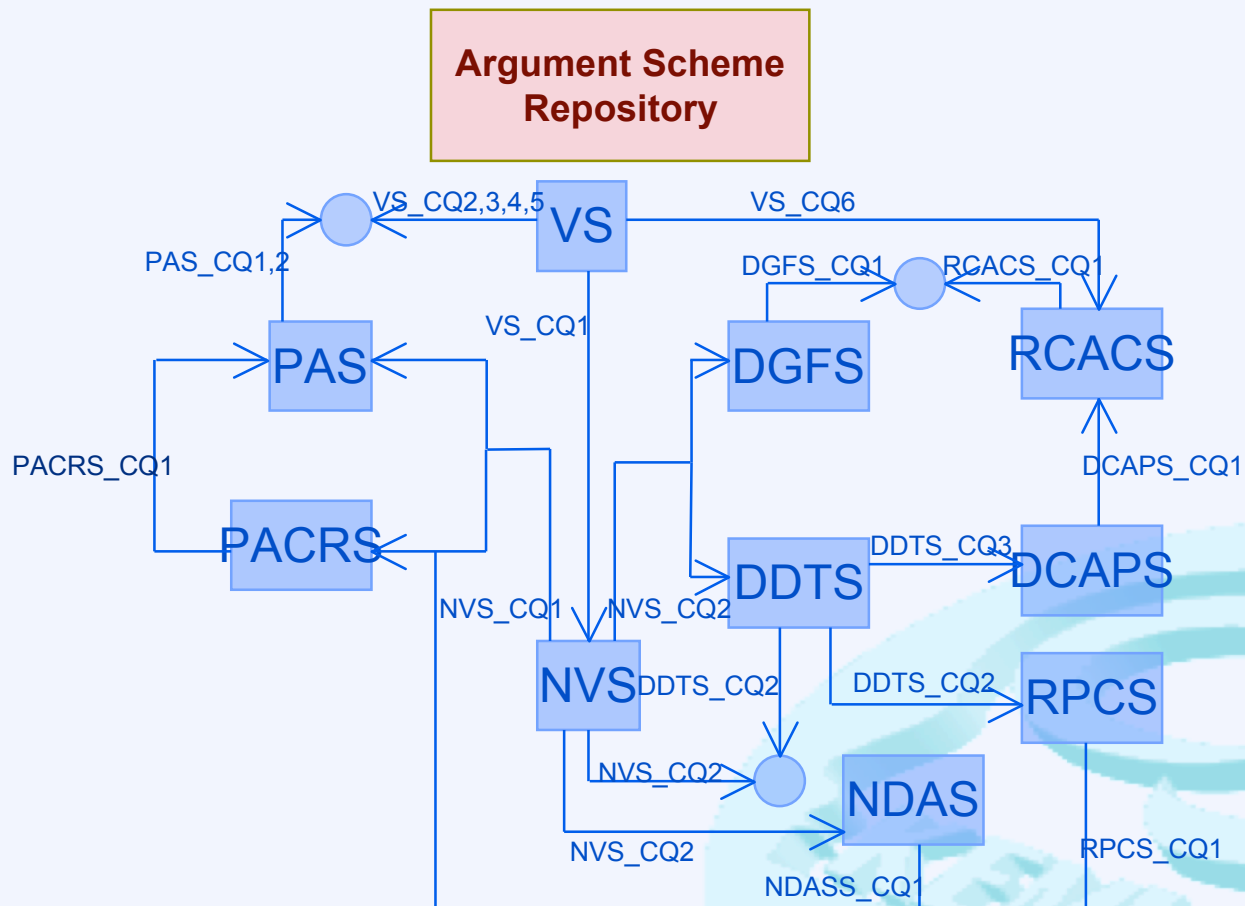
Argument Schemes and Critical Questions as a Protocol



Architecture of the *ProCLAIM* model



Argument Scheme Repository “Problem”



Argument Scheme Repository “Problem”

DCS Donor Contraindication Scheme:

back

The donor has which is a contraindication for donating a

add comment

Example

Critical Questions

Is a contraindication for donating a ?

Yes: Infection | Intoxication | Graft Failure | Risk Factor No: No Disease Associated | Urgency-0

Does the donor have ?

Yes: Tests | Clinical Records No:

Argument Schemes and Critical Questions

- Walton & Kreh
- Atkinson *et al.*



Argument Schemes and Critical Questions

- 16 Critical Questions:
 - Blah blah
 - Blah blah blah
- Atkinson *et al.*



Argument Scheme Over Action Proposal

In the current circumstances **R**
we should perform action **A**
to achieve new circumstances **S**
which will realise some goal **G**
~~which will promote some value **V**~~

V = safety



Atkinson *et al*, 2005

Argument Scheme Over Action Proposal

Undesirable Goals:

- severe_infection
- cancer
- acute_rejection
- ...



Argument Scheme Over Action Proposal

<Context, Fact, Prop_Action, Effect, Neg_Goal>

Context is a set of facts that are not under dispute.

Fact is a set of facts that, given the context **Context** and the proposed action (set of actions) **Prop_Action** result in a set of states **Effect** that realizes some undesirable goal **Neg_Goal**.

Argument Scheme Over Action Proposal

Abstract Argument Schema:

<Context, Fact, Prop_Action, Effect, Neg_Goal>

Argument Pro:

<Context, Fact, Prop_Action, Effect, nil >

Argument Con:

<Context, Fact, Prop_Action, Effect, seve_infect>

Protocol-based exchange of arguments

AS1: < min_context, {}, prop_action, {}, nil >

context = {donor(D,O),
potential_recip(R,O)}.

prop_action = {transplant(O,R)}.

Protocol-based exchange of arguments

AS1: < min_context, {}, prop_action, {}, nil >



AS2: < min_context, fact, prop_action, effect, neg_goal >

context = {donor(D,O), potential_recip(R,O)}.

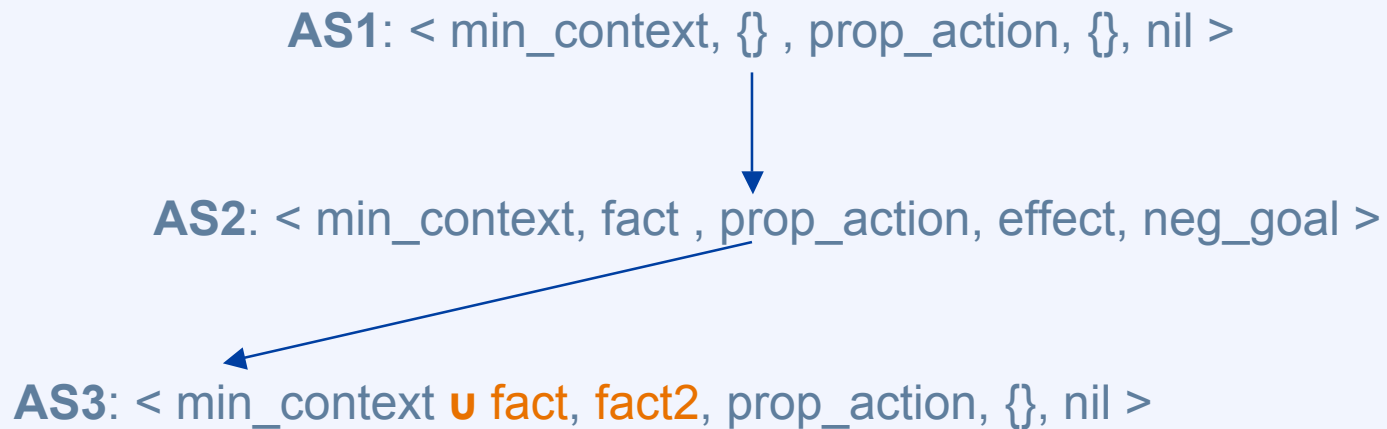
prop_action = {transplant(O,R)}.

fact = {donor_prop(D,P1)} (P1= Hepatitis C)

effect = {recipient_prop(R,P2)} (P2= Hepatitis C)

neg_goal = severe_infection.

Protocol-based exchange of arguments



context = {donor(D,O), potential_recip(R,O), donor_prop(D,P1)}.

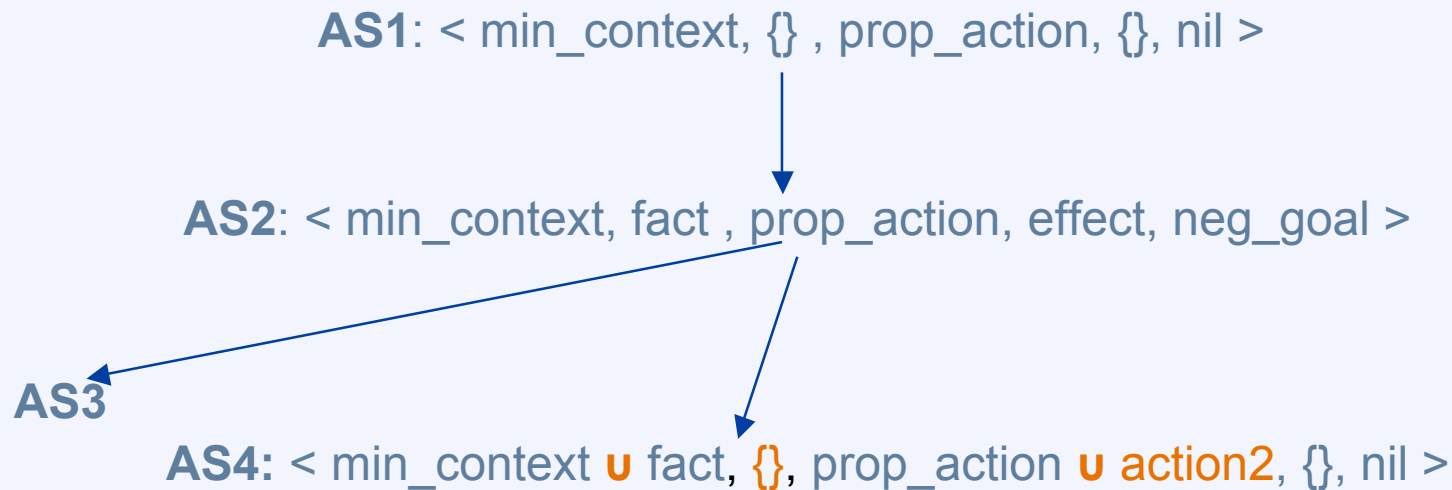
prop_action = {transplant(O,R)}.

P1 = Smoking H

fact2 = {donor_prop(D,P3)}

P3 = No COPD

Protocol-based exchange of arguments



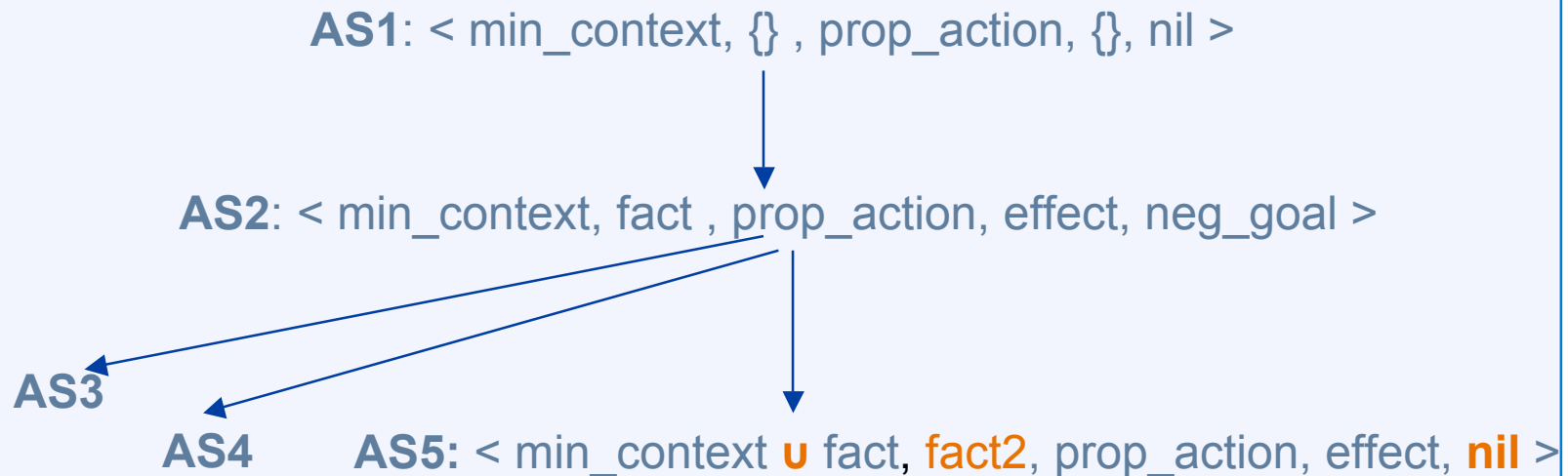
context = {donor(D,O), potential_recip(R,O), donor_prop(D,P1)}.

prop_action = {transplant(O,R), **treatment(R,T)**}.

P1 = Streptococcus Viridans Endocarditis (Effect = sv
infection)

T = Penicillin

Protocol-based exchange of arguments



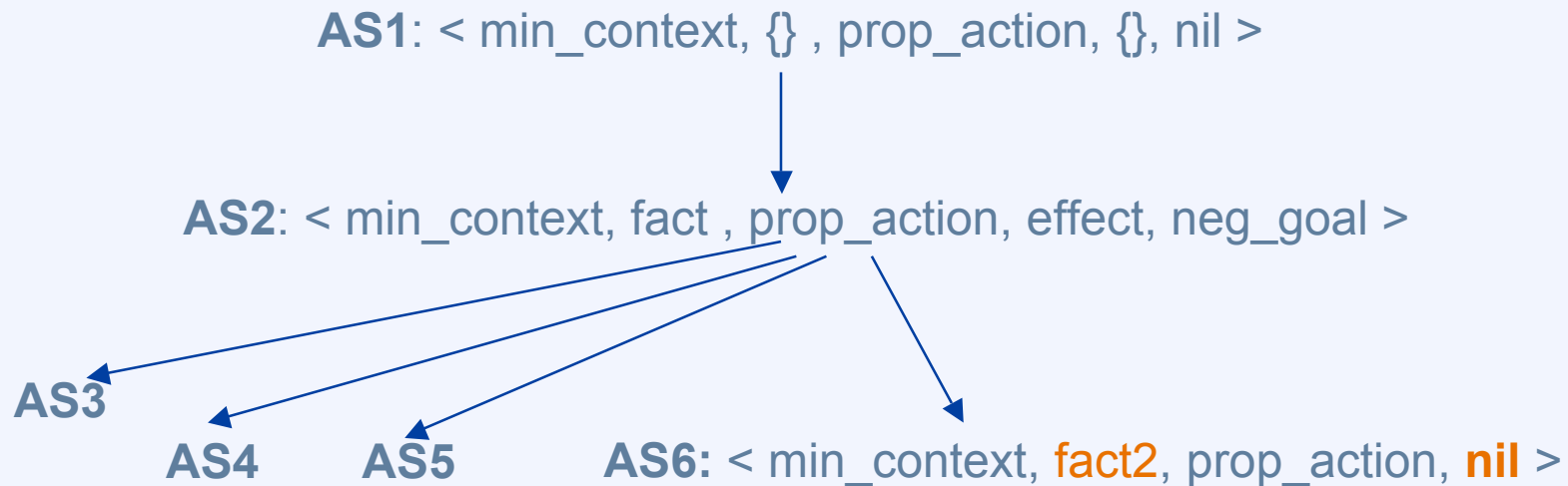
context = {donor(D,O), potential_recip(R,O), donor_prop(D,P1)}.

fact2 = { potential_recipient_prop(R,P2) } (P1= Hepatitis C)

prop_action = {transplant(O,R)}.

Effect = { recipient_prop(R,P2) } (P2= Hepatitis C)

Protocol-based exchange of arguments



Where **fact2** more specific that **fact**

context = {donor(D,O), potential_recip(R,O)}.

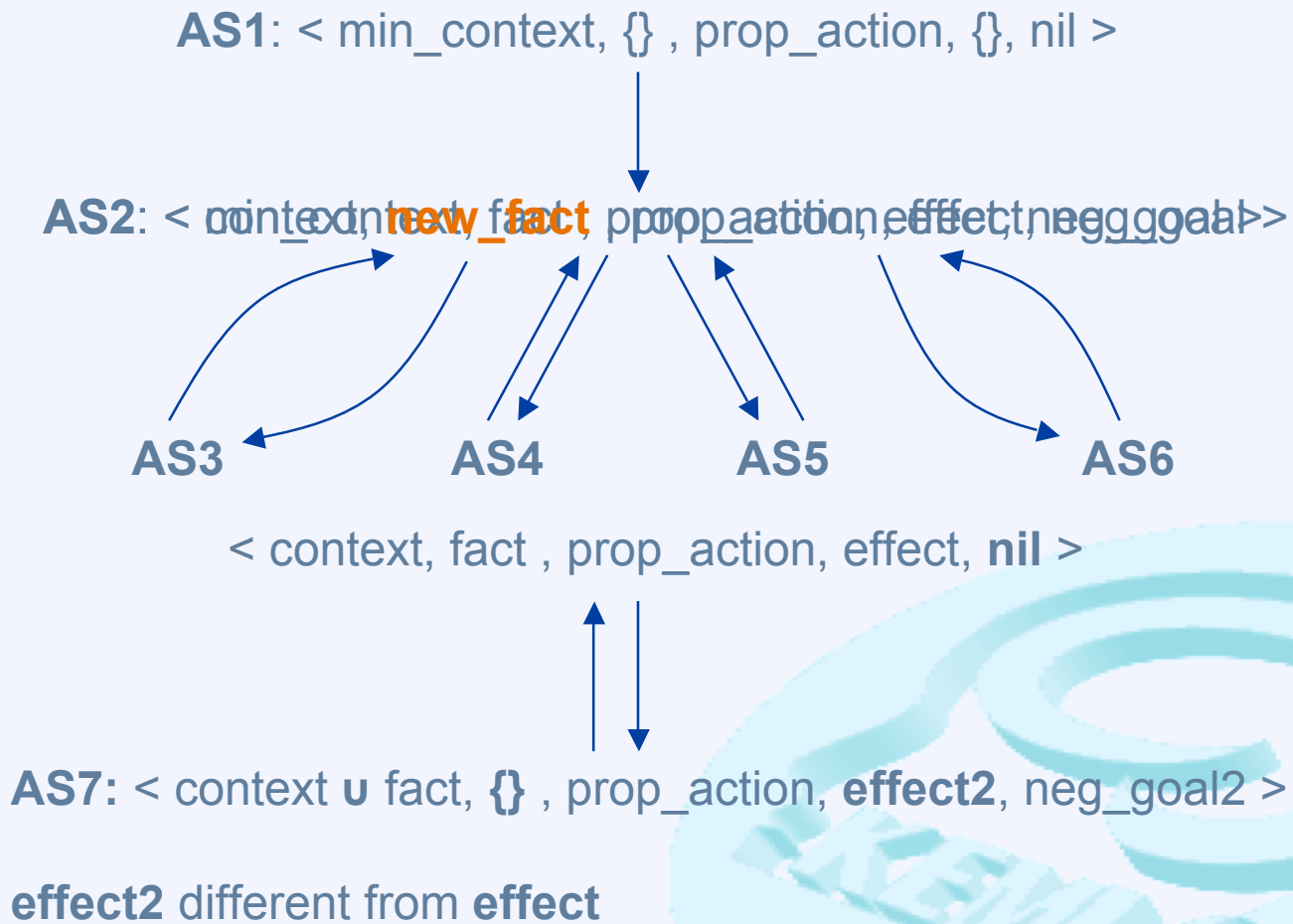
fact2 = {donor_prop(D,P2)} (P2 more specific than P1)

prop_action = {transplant(O,R)}.

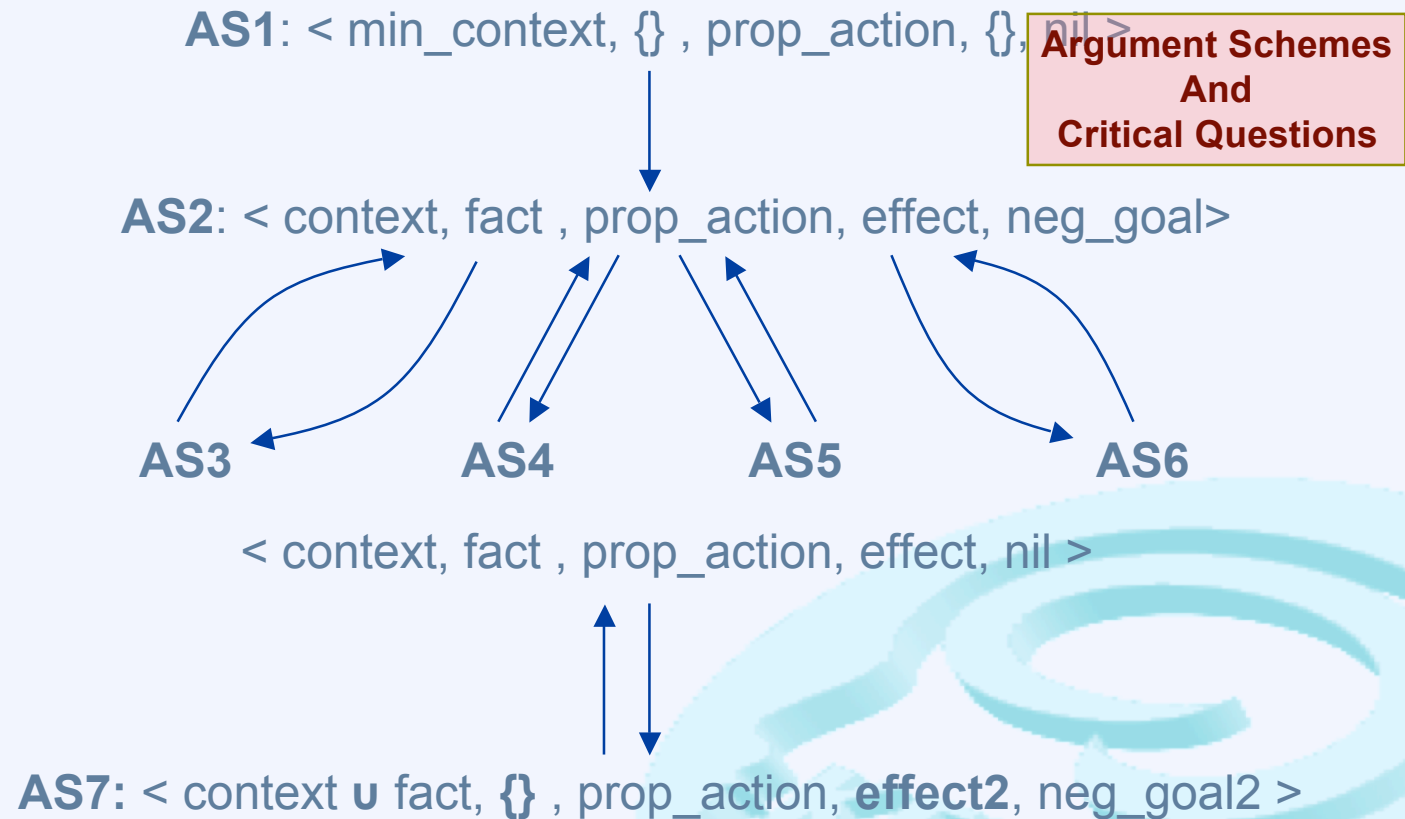
P1 = Cancer

P2 = non systemic cancer

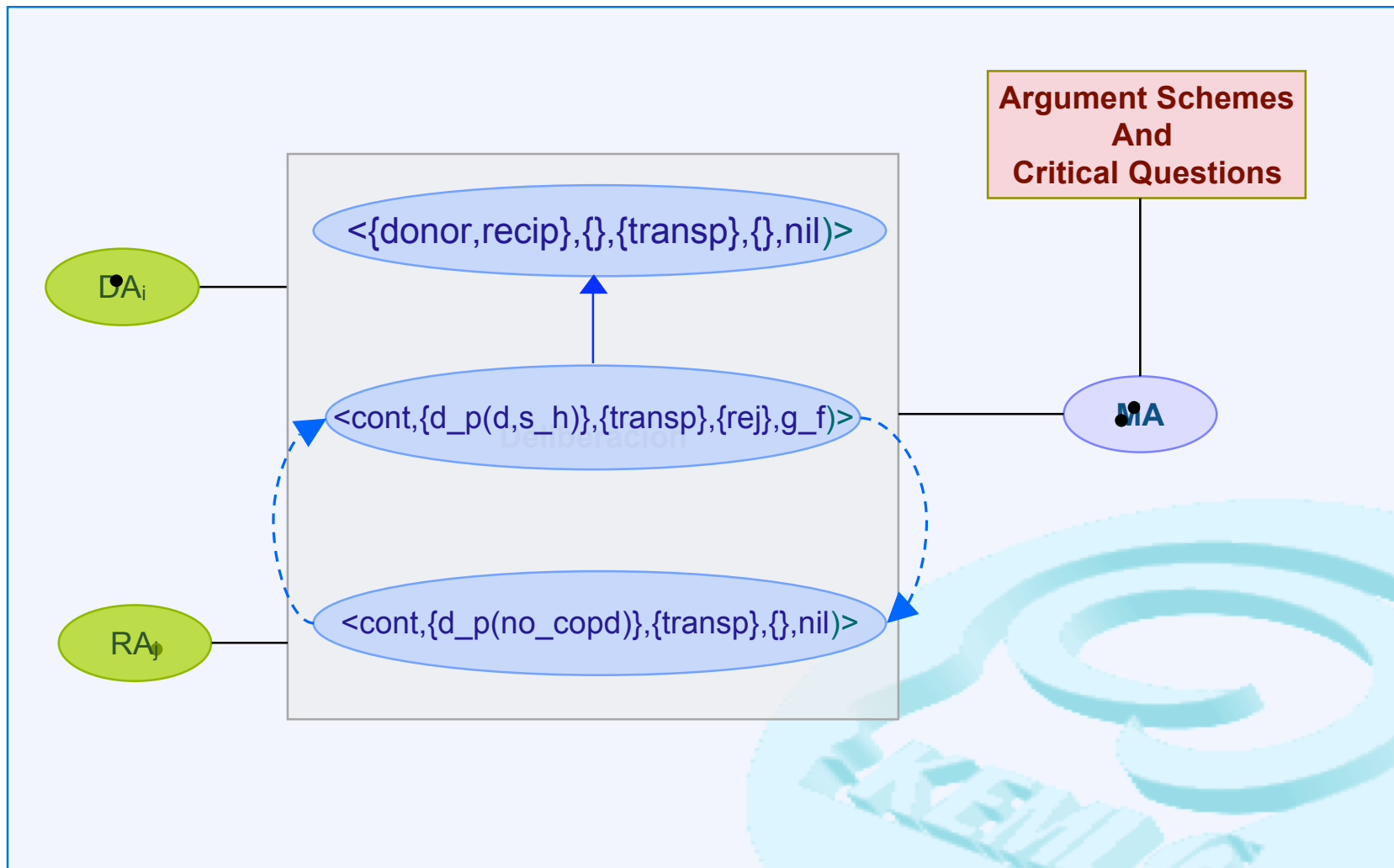
Protocol-based exchange of arguments



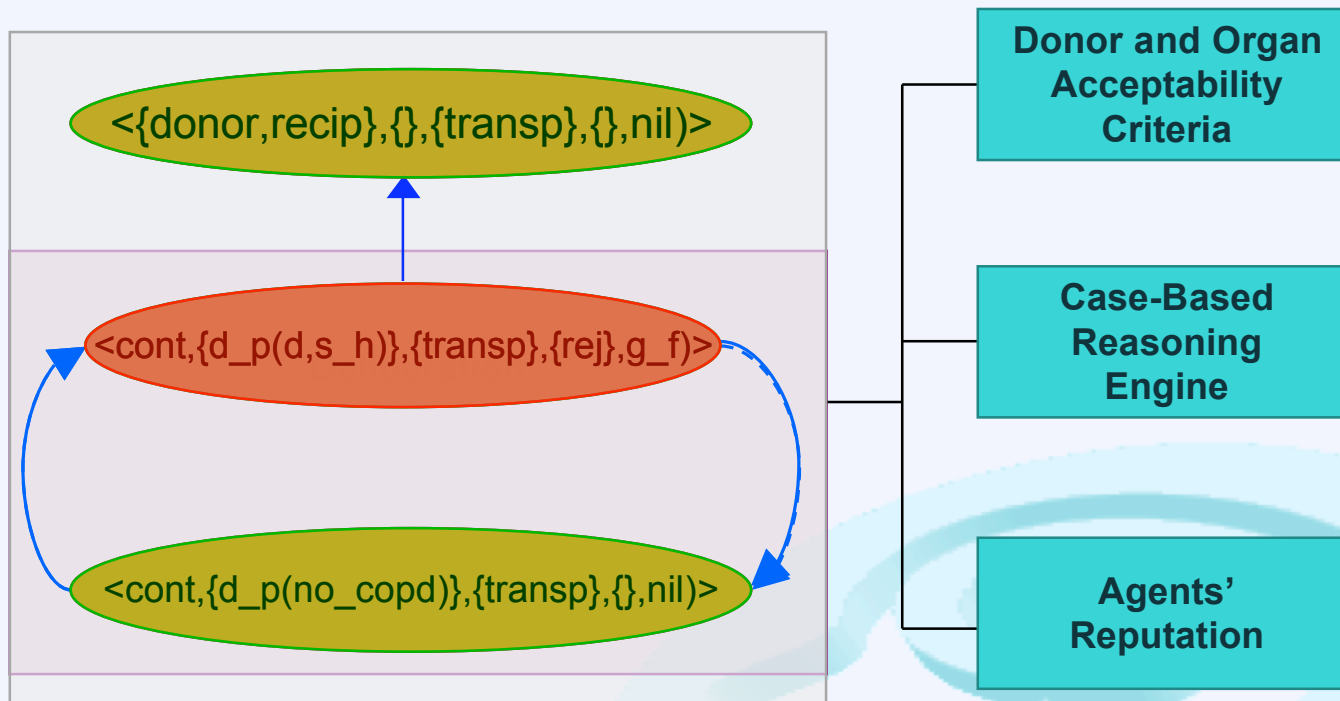
Argument Scheme Repository



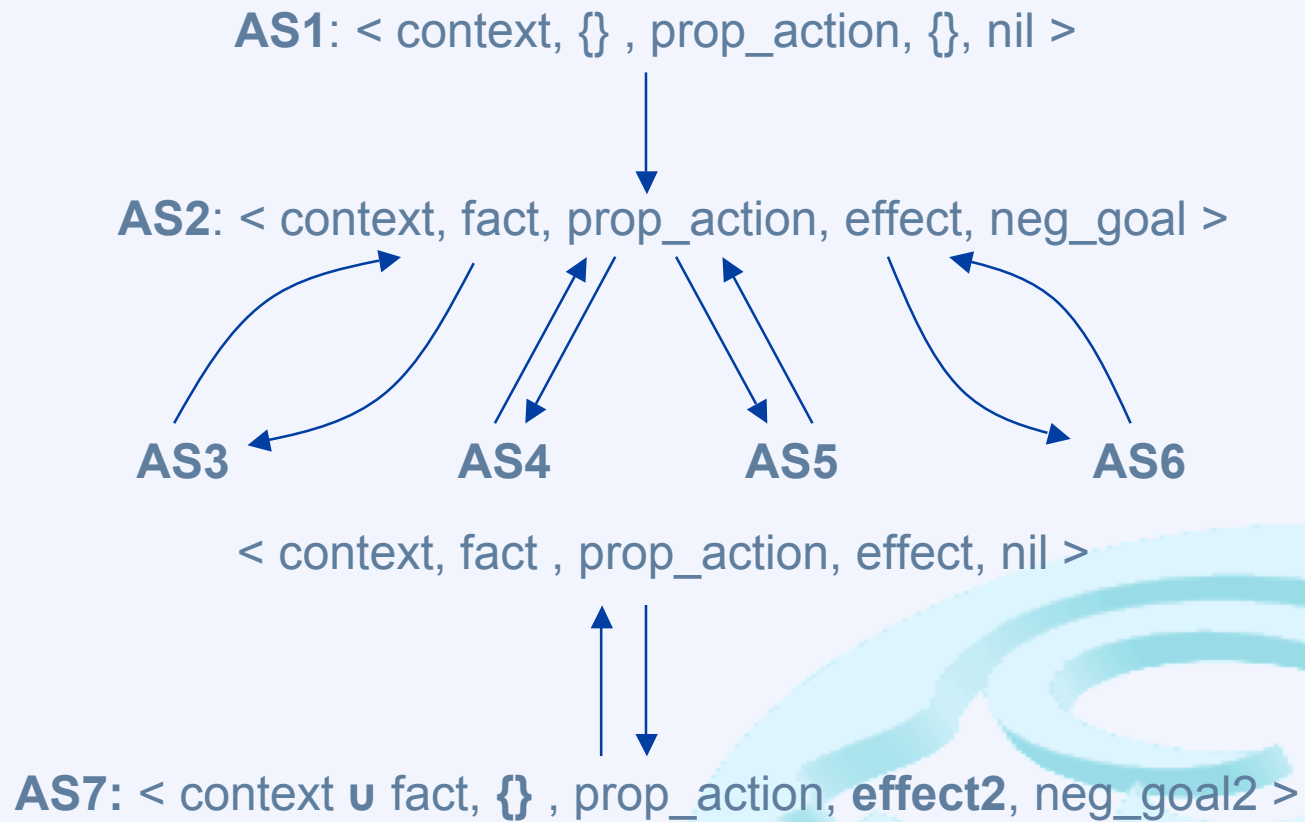
Argumentation Example



Argument Evaluation



Building the Argument Scheme Repository



Building the Argument Scheme Repository

AS1: $\langle \{ \text{donor}(D,O), \text{pot_recip}(R,O) \}, \{ \}, \{ \text{transp}(O,R) \}, \{ \}, \text{nil} \rangle$

AS2: $\langle \text{context}, \text{fact}, \text{prop_action}, \text{effect}, \text{neg_goal} \rangle$

AS3

AS4

AS5

AS6

$\langle \text{context}, \text{fact}, \text{prop_action}, \text{effect}, \text{nil} \rangle$

AS7: $\langle \text{context} \cup \text{fact}, \{ \}, \text{prop_action}, \text{effect2}, \text{neg_goal2} \rangle$

Building the Argument Scheme Repository

AS1: $\langle \{ \text{donor}(D,O), \text{pot_recip}(R,O) \}, \{ \}, \{ \text{transp}(O,R) \}, \{ \}, \text{nil} \rangle$

AS2: $\langle \text{context}, \{ \text{d_p}(D,P1) \}, \text{prop_action}, \{ \text{recip_prop}(P2) \}, \text{sever_infect} \rangle$

AS3

AS4

AS5

AS6

$\langle \text{context}, \text{fact}, \text{prop_action}, \text{effect}, \text{nil} \rangle$

AS7: $\langle \text{context} \cup \text{fact}, \{ \}, \text{prop_action}, \text{effect2}, \text{neg_goal2} \rangle$

Building the Argument Scheme Repository

AS1: $\langle \{ \text{donor}(D,O), \text{pot_recip}(R,O) \}, \{ \}, \{ \text{transp}(O,R) \}, \{ \}, \text{nil} \rangle$

AS2: $\langle \text{context}, \{ \text{d_p}(D,P1) \}, \text{prop_action}, \{ \text{reject}(R,O) \}, \text{graft_failure} \rangle$

AS3

AS4

AS5

AS6

$\langle \text{context}, \text{fact}, \text{prop_action}, \text{effect}, \text{nil} \rangle$

AS7: $\langle \text{context} \cup \text{fact}, \{ \}, \text{prop_action}, \text{effect2}, \text{neg_goal2} \rangle$

Building the Argument Scheme Repository

Facts	Actions	Effects	Neg_Goals
donor (D, O) pot_recip (R, O) d_prop (D, P) r_prop (R, P)	transplant (O, R) treat (R, T) move (H1, R, H2)	recip_prop (R, P) reject (R, O)	severe_infect graft_failure cancer intoxication



Building the Argument Scheme Repository

AS1: < context, {} , prop_action, {}, nil >

Actions

transplant(O, R)

treat(R, T)

move(H1, R, H2)

.....

Building the Argument Scheme Repository

AS1: $\langle \text{context}, \{\}, \{\text{transplant}(O,R)\}, \{\}, \text{nil} \rangle$

Facts

donor(D,O)

pot_recip(R,O)

d_prop(D,P)

r_prop(R,P)

.....

Building the Argument Scheme Repository

AS1: $\langle \{d(D,O), pot_re(R,O)\}, \{\}, \{transplant(O,R)\}, \{\}, nil \rangle$

AS2: $\langle \{d(D,O), pot_re(R,O)\}, fact, \{transplant(O,R)\}, effect, neg_goal \rangle$

Neg_Goals

severe_infect

graft_failure

cancer

intoxication

.....

Building the Argument Scheme Repository

AS1: $\langle \{d(D,O), \text{pot_re}(R,O)\}, \{\}, \{\text{transplant}(O,R)\}, \{\}, \text{nil} \rangle$

AS2: $\langle \{d(D,O), \text{pot_re}(R,O)\}, \text{fact}, \{\text{transplant}(O,R)\}, \text{effect}, \text{sev_inf} \rangle$

Effects

`recip_prop(R,P)`

`reject(R,O)`

.....

Building the Argument Scheme Repository

AS1: $\langle \{d(D,O), pot_re(R,O)\}, \{\}, \{transplant(O,R)\}, \{\}, nil \rangle$



AS2: $\langle \{d(D,O), pot_re(R,O)\}, fact, \{transplant(O,R)\}, \{r_p(R,P2)\}, sev_inf \rangle$

Facts

donor(D, O)

pot_recip(R, O)

d_prop(D, P)

r_prop(R, P)

.....

Building the Argument Scheme Repository

AS1: $\langle \{d(D,O), pot_re(R,O)\}, \{\}, \{transplant(O,R)\}, \{\}, nil \rangle$

AS2: $\langle \{d(D,O), pot_re(R,O)\}, \{d_p(D,P1)\}, \{transplant(O,R)\}, \{r_p(R,P2)\}, sev_inf \rangle$

AS2: $\langle \{d(D,O), pot_re(R,O)\}, fact, \{transplant(O,R)\}, effect, neg_goal \rangle$

Neg_Goals

severe_infect

graft_failure

cancer

intoxication

.....

Building the Argument Scheme Repository

AS1: $\langle \{d(D,O), pot_re(R,O)\}, \{\}, \{transplant(O,R)\}, \{\}, nil \rangle$

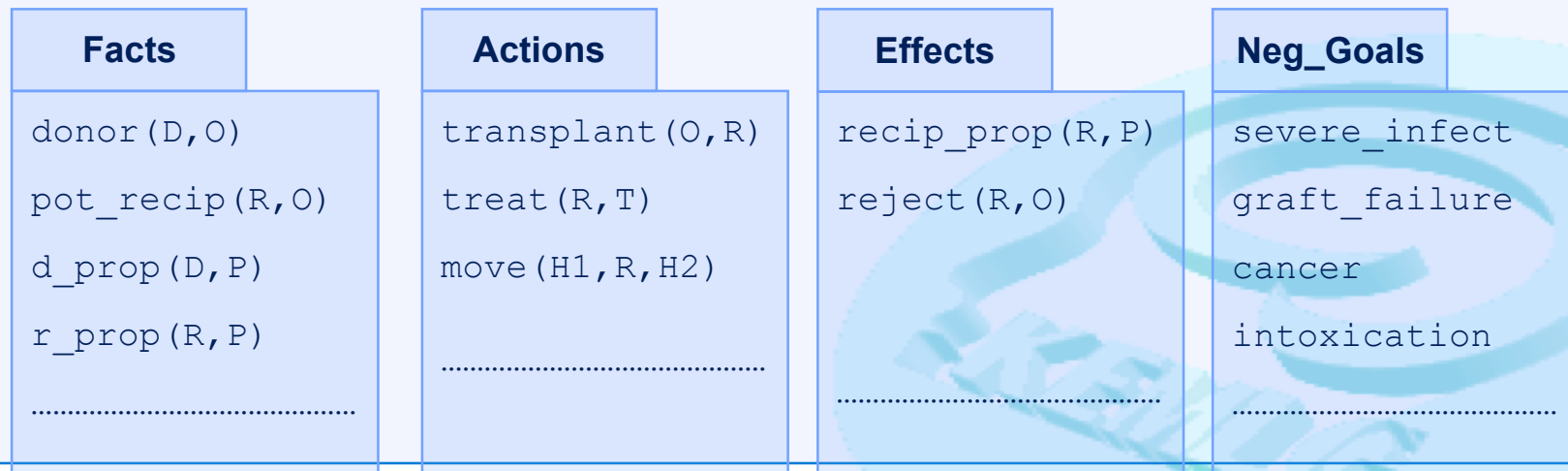
AS2: $\langle \{d(D,O), pot_re(R,O)\}, \{d_p(D,P1)\}, \{transplant(O,R)\}, \{r_p(R,P2)\}, sev_inf \rangle$

AS3

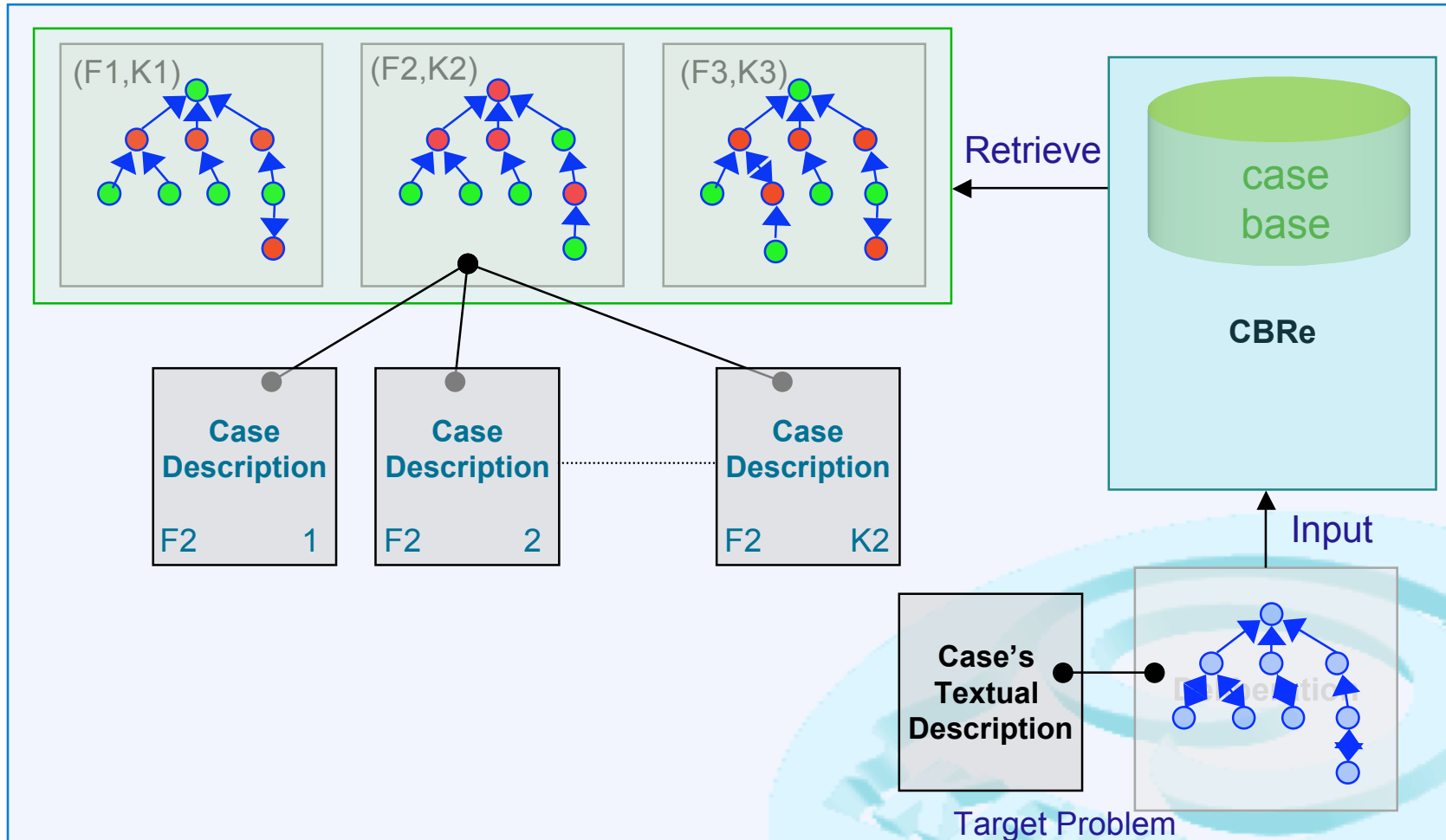
AS4

AS5

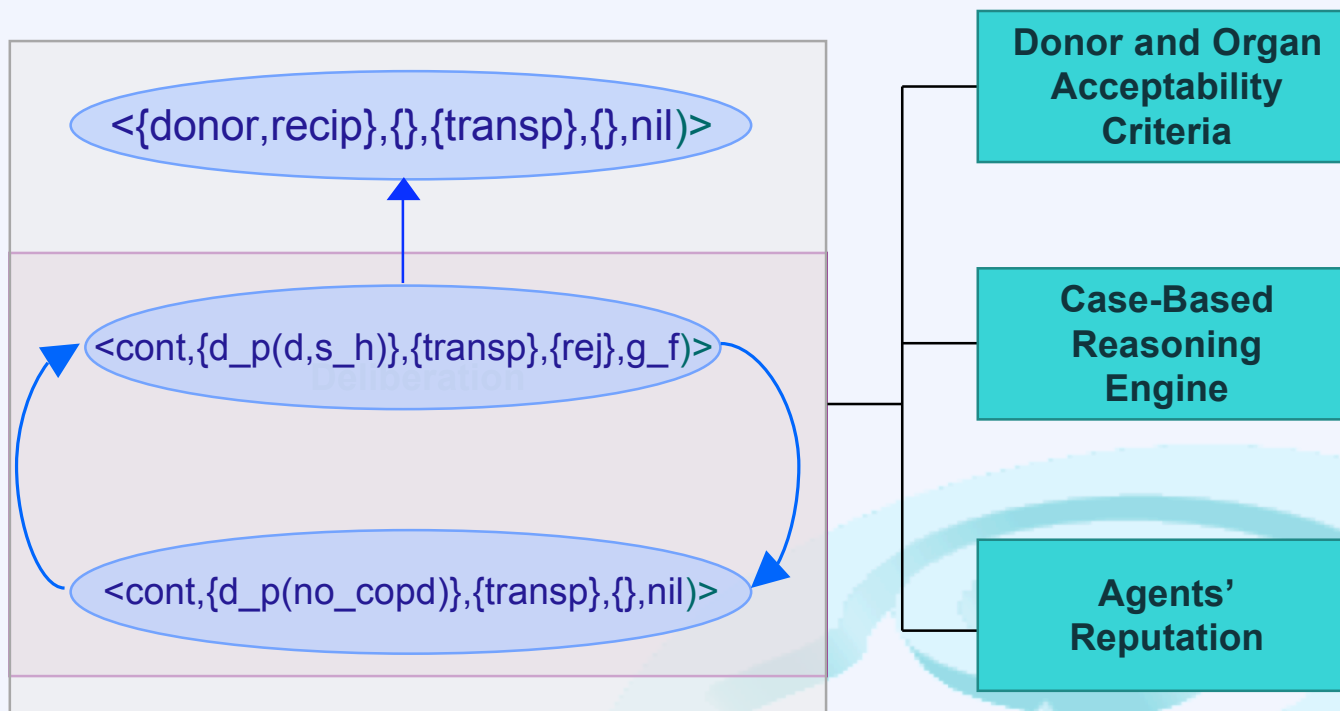
AS6



The CBRe: 4R Reasoning Cycle (Retrieval)



Argument Evaluation



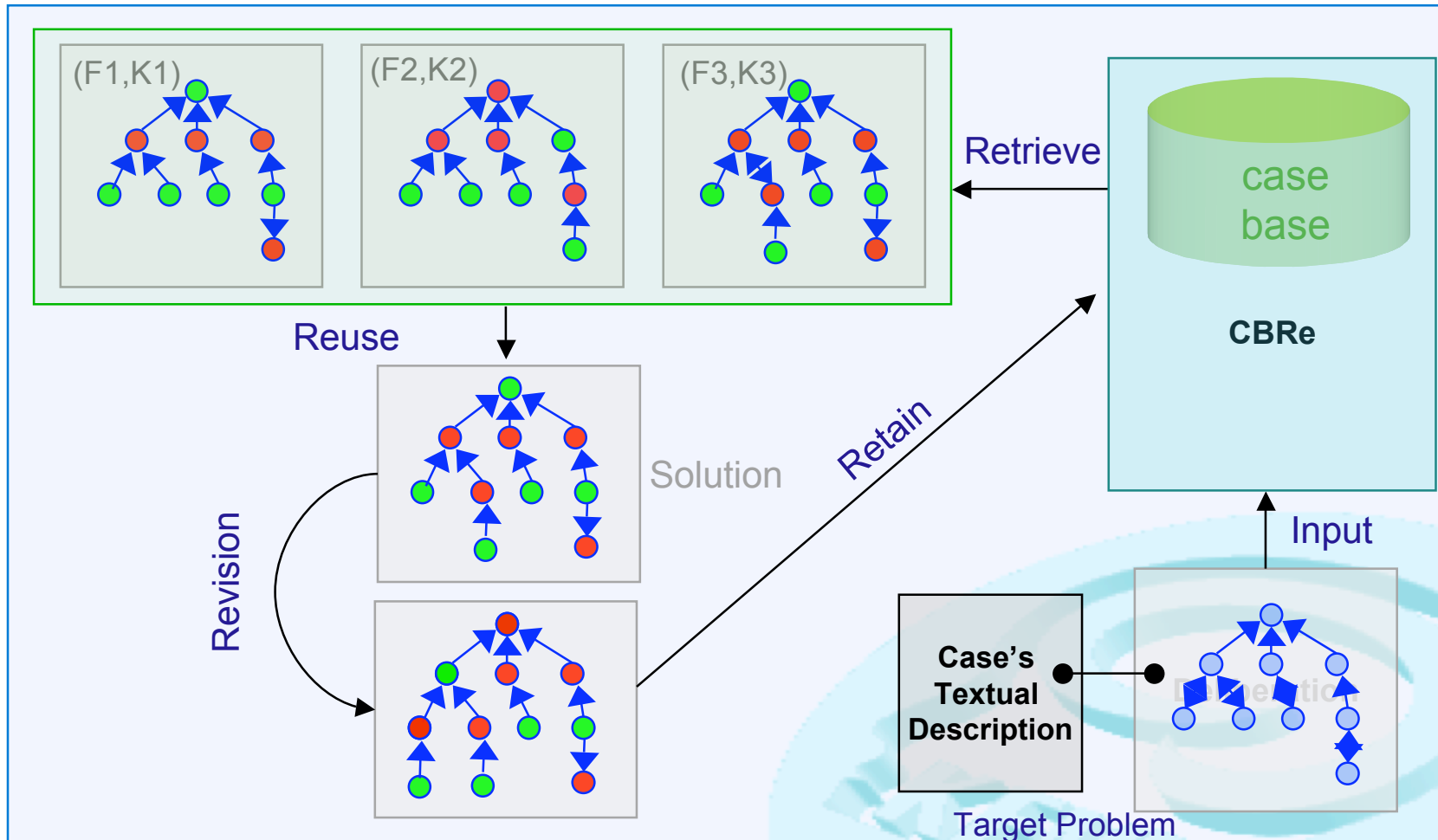
The CBR_e: 4R Reasoning Cycle (Reuse)

The screenshot displays the Case-Based Reasoning Engine (CBRe) interface during the 'Reuse' phase of the 4R Reasoning Cycle. The interface is titled 'Case-Based Reasoning Engine' and includes a menu bar (File, Edit, Help) and buttons for 'Revise', 'Show Target Case', and 'Show Solution Graph'. The main workspace is divided into several panels:

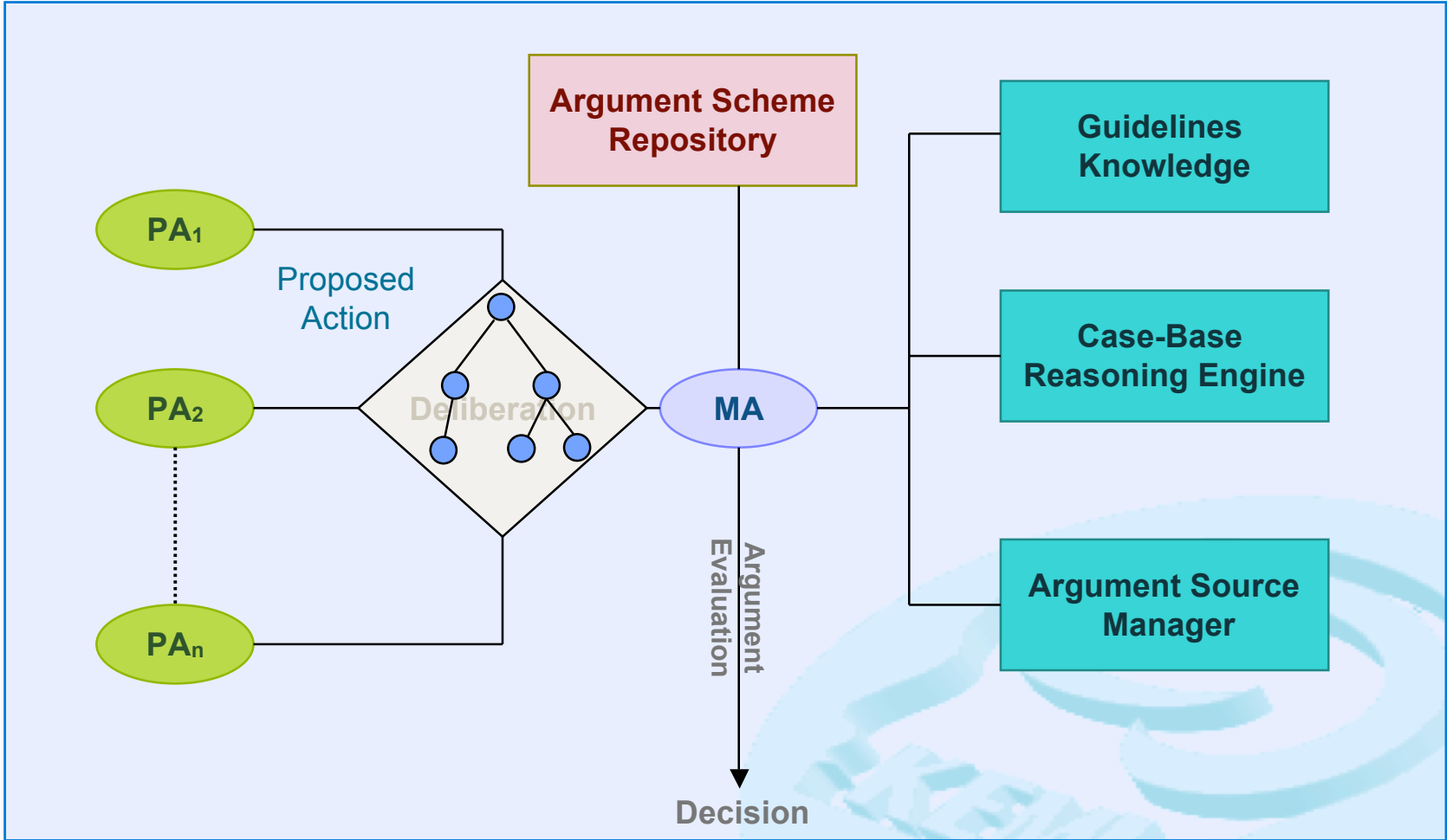
- List of Cases:** A vertical list of cases numbered 10 through 39.
- List of Acrued Argument Graphs:** A list of argument graphs numbered 1 through 3.
- Target Case's Argument Graph:** A hierarchical graph showing the structure of the target case. The root node is 'Topic vs(kidney)', which branches into 'nvs1(kidney, sve)', 'nvs1_cq1(kidney, sve)', 'ddts(sve, kidney, svi)', 'nvs1(kidney, hcv)', 'nvs1_cq1(kidney, hcv)', 'ddts(hcv, kidney, hcv)', and 'rpcs(hcv, kidney, hcv, hcv)'. Arrows indicate the flow of the argument.
- Proposed Solution:** A similar hierarchical graph showing the proposed solution. It includes the same nodes as the target case, plus 'caps(sve, kidney, svi, penicillin)'.

On the right side, a green cylinder represents the 'case base'. Below it, the text 'CBRe' is displayed. An 'Input' diagram, showing a tree structure with nodes and arrows, is positioned at the bottom right, with an arrow pointing upwards towards the 'case base'. The text 'Target Problem' is written below the input diagram.

The CBR_e: 4R Reasoning Cycle (Revision/Retain)



ProCLAIM's Architecture



Proponent Agent

UPC - 22/05/2007



Mediator Agent



<http://www.lsi.upc.edu/~webia/KEMLG/>

