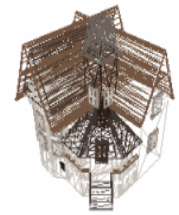


ASP – Answer Set Programming
An operational formalism *(Baral 2003)*

A program Π is a set of expression ρ

$$\rho : L_0 \; or \; L_1 \; or \; ... \; L_k \leftarrow L_{k+1}, L_{k+2}, ... L_m, not \; L_{m+1}, ..., not \; L_n$$

where

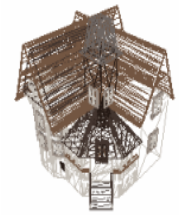
- the L_i are literals ***(atoms or atom negations)***
- The « not » is a negation by failure

Intuitive meaning: for all Herbrand interpretation such that

$\{L_{k+1}, L_{k+2}, ... , L_m\}$ is true

while $\{L_{m+1}, ..., L_n\}$ failed to be proved

one can derive $\{L_0, L_1, ... L_k\}$



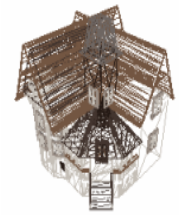
An Artificial Agent

*act(P,S,**G**,**A**)* ← *person(P),*
situation(S), goal(G), action(A),
*will(P, S, **G**),*
*solve_goal(P, S, G, **A**).*

← *act(P, S, G, A), act(P, S, G, B), A ≠ B.*

Autonomy

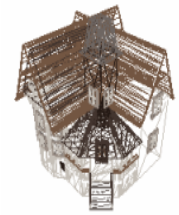
Intelligence



An Ethical Artificial Agent

*act(P,S,**G**,**A**)* ← *person(P),*
situation(S), goal(G), action(A),
*will(P, S, **G**),*
*solve_goal(P, S, G, **A**),*
moral(P, S, G, A).

← *act(P, S, G, A), act(P, S, G, B), A ≠ B.*



An "Aristotelian" Perspective

Predicates:

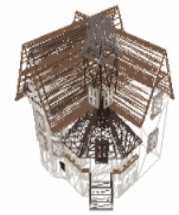
`csq(A,S,C)`: *consequence*

`worse(A,B)`: *comparison of action*

`worst_csq(A,S,C)`: *worst consequence*

*good(P,S,**G**,**A**)* ← *will(P, S, **G**),*
solve_goal(P,S,G,A),worst_csq(A,S,C),
will(P,S,U), solve_goal(P,S,U,B),
csq(B,S,D), worse(D,C).

*bad(P,S,**G**,**A**)* ← *will(P, S, **G**),*
solve_goal(P,S,G,A),worst_csq(A,S,C),
will(P,S,U), solve_goal(P,S,U,B), A ≠ B, csq(B,S,D),
***not** worse(D,C).*
The same action may be both good and bad!

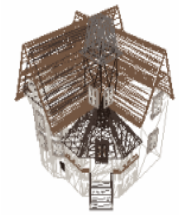


Moral and immoral

*moral(P,S,**G**,**A**)* ← ***not*** *bad(P,S,G,A).*

*moral(P,S,**G**,**A**)* ← *good(P,S,G,A),*
not *immoral(P,S,G,A).*

*immoral(P,S,**G**,**A**)* ← *bad(P,S,G,A),*
not *moral(P,S,G,A).*



The Lying Example

- Three persons: “I”, Peter and Paul
- Two possibilities: tell(P, truth) or tell(P, lie)
- Consequence: tell(“I”, truth) generates a murder

csq(A,S,A) ← .

csq(A,S,B) ← *csq(A,S,C), csq(C,S,B).*

csq(tell(“I”, truth),s0,murder) ←.

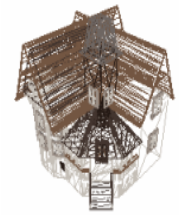
worse(A,B) ← *better(B,A),* ***not*** *better(A,B).*

worse(A,B) ← *worse(A,C), worse(C,B).*

better(A, tell(P, lie))←.

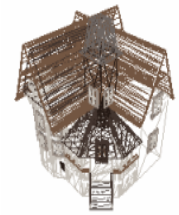
better(A, murder)←.

better(A,A) ←.



The Lying Example

- Half of the answser sets contain:
  act("I", answer("I"),s0, tell("I", truth)
- And half of the answser sets contain:
  act("I", answer("I"),s0, tell("I", lie)
- If we add *worse(murder, lie)* then all the answer sets that contain
  act("I", answer("I"),s0, tell("I", truth) are removed.



Torture example

- Three persons: “I”, Peter and Paul
- Two possibilities: interrogate(P, torture) or interrogate(P, soft)
- Consequence: interrogate(“I”, soft) generates an attack

csq(A,S,A) ← .

csq(A,S,B) ← *csq(A,S,C), csq(C,S,B).*

csq(interrogate(“I”, soft),s0,attack) ←.

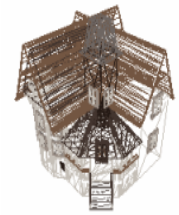
worse(A,B) ← *better(B,A),* ***not*** *better(A,B).*

worse(A,B) ← *worse(A,C), worse(C,B).*

better(A, interrogate(P, torture))←.

better(A, attack)←.

better(A,A) ←.



The Torture Example

- Half of the answser sets contain:
  act("I", question("I"),s0, interrogate("I", torture)
- And half of the answser sets contain:
  act("I", question("I"),s0, interrogate("I", soft)
- If we add *worse(attack, torture)* then all the answer sets that contain
  act("I", question("I"),s0, interrogate("I", soft) are removed.