Appendix C - Results from PROLOG rules

Facts produced by the PROLOG rules. Altogether, the relations, the concepts and the knowledge sources constitute the domain model.

We start by making concepts, and thereafter ask which relations were found.

```
?- reconsult('C:\\svend prolog\\hr1_2.pro')
yes
?- make_concept.
yes


?- concept(C).

C = definit ;
C = agent ;
C = agent ;
C = reactiv ;
C = autonom ;
C = interfac ;
C = dictionari ;
C = softwar ;
C = intellig ;
C = agent ;
C = debat ;
C = user ;
C = agent ;
C = foley ;
no
```

This is the output of the PROLOG environment used, but internally, the facts hold the form listed below:

```
concept(agent).
concept(autonom).
concept(debat).
concept(definit).
concept(dictionari).
concept(foley ;
concept(intellig).
concept(interfac).
concept(reactiv).
concept(softwar).
concept(user).

?- reconsult('C:\\svend prolog\\hr1_2.pro')
yes
?- hr1.
yes
?- hr2.
yes
?- relation(R,E,T).

relation(has_deeper_explanation, definit, agent).
relation(has_deeper_explanation, definit, softwar).
relation(has_deeper_explanation, agent, softwar).
relation(has_deeper_explanation, softwar, debat).

relation(external, debat, foley).
```

The same procedure with finding the relations and asking which were found, yields the following results:

```
relation(parent, agent, reactiv).
relation(parent, agent, autom).
relation(parent, agent, dictionari).
relation(parent, agent, interfac).
relation(parent, agent, user).
relation(parent, autom, intellig).
relation(parent, debat, interfac).
relation(parent, debat, user).
relation(parent, debat, agent).
relation(parent, definit, agent).
relation(parent, definit, autom).
relation(parent, intellig, agent).
relation(parent, intellig, interfac).
relation(parent, intellig, user).
relation(parent, reactiv, autom).
relation(parent, softwar, intellig).
relation(parent, softwar, agent).
relation(parent, softwar, interfac).
relation(parent, softwar, intellig).
relation(parent, softwar, user).
relation(parent, user, interfac).

relation(synonym, agent, debat).
relation(synonym, agent, reactiv).
relation(synonym, agent, user).
relation(synonym, autonom, softwar).
relation(synonym, autonom, user).
relation(synonym, debat, user).
relation(synonym, definit, agent).
relation(synonym, definit, reactiv).
relation(synonym, intellig, agent).
relation(synonym, intellig, debat).
relation(synonym, intellig, user).
relation(synonym, interfac, agent).
relation(synonym, interfac, debat).
relation(synonym, interfac, intellig).
relation(synonym, interfac, softwar).
relation(synonym, interfac, user).
relation(synonym, reactiv, debat).
relation(synonym, reactiv, interfac).
relation(synonym, reactiv, user).
relation(synonym, softwar, agent).
relation(synonym, softwar, debat).
relation(synonym, softwar, intellig).
relation(synonym, softwar, user).
```

Finally, when asking for HR6 and HR7, nothing is found.

```
?- reconsult('C:\\svend prolog\\hr6.pro')
yes
?- hr6.

no
?- reconsult('C:\\svend prolog\\hr7.pro')
yes
?- hr7.

no
```