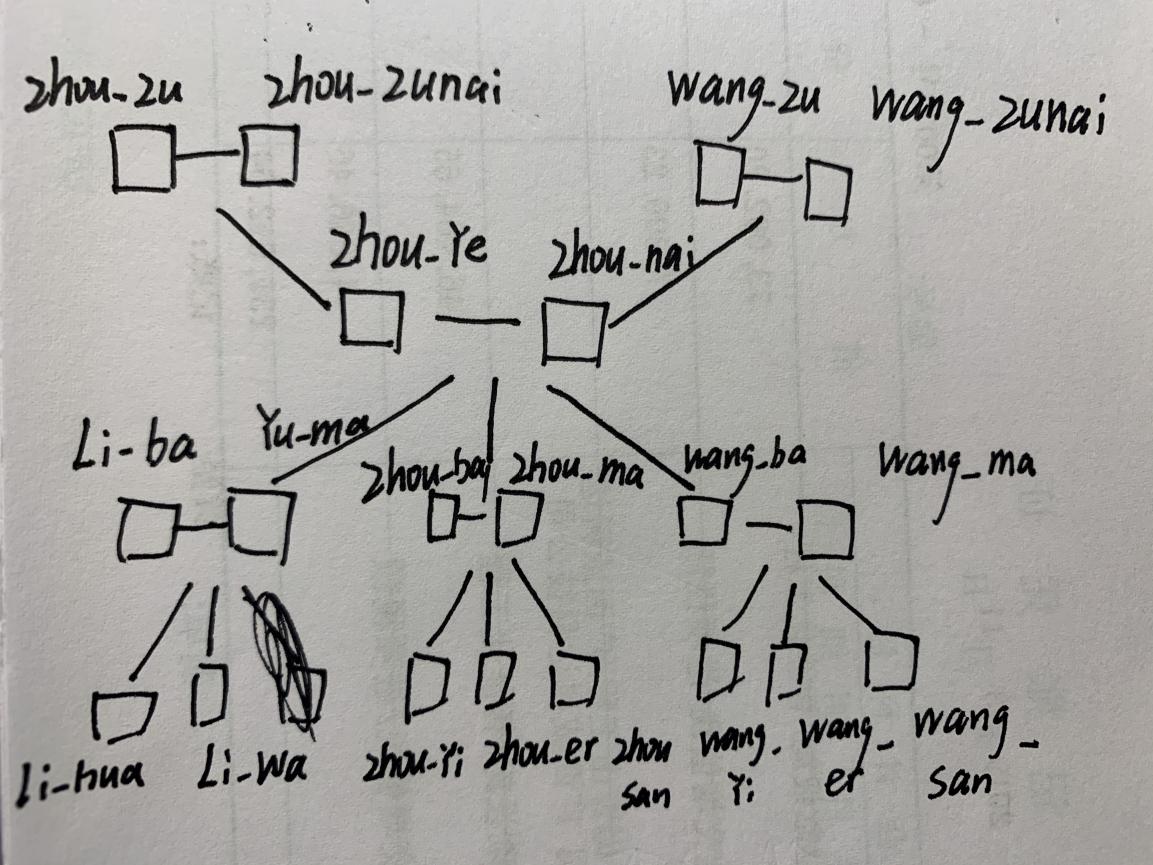
**Artificial Intelligence Lab2**

1. **Download and install the ‘Semantic’ program.**

I have installed the Visual Prolog with the version of 7.3. After installing Visual Prolog I open the Semantic project successfully.

1. **Transfer the knowledge base created in the Lab.1 to the ‘Semantic’ environment.**

This is the family tree of my homework:



Then I will use prolog to define the relationship of these family member:

I use children and marital relationships as connections, and define the marital relationships and parent-child relationships of different family members in order to build the genealogy of the entire family.

# define the first generation

f("zhou\_zu","has a spouse","zhou\_zunai").

f("wang\_zu","has a spouse","wang\_zunai").

# define the second generation

f("wang\_zu","has a child","zhou\_nai").

f("zhou\_zu","has a child","zhou\_ye").

f("zhou\_ye","has a spouse","zhou\_nai").

# define the third generation

f("zhou\_ye","has a child","yu\_ma").

f("zhou\_ye","has a child","yu\_ma").

f("zhou\_ye","has a child","zhou\_ba").

f("zhou\_ye","has a child","wang\_ba").

f("yu\_ma","has a spouse","li\_ba").

f("zhou\_ba","has a spouse","zhou\_ma").

f("wang\_ba","has a spouse","wang\_ma").

# define the last generation

f("yu\_ma","has a child","li\_hua").

f("yu\_ma","has a child","li\_wa").

f("zhou\_ba","has a child","zhou\_yi").

f("zhou\_ba","has a child","zhou\_er").

f("zhou\_ba","has a child","zhou\_san").

f("wang\_ba","has a child","wang\_yi").

f("wang\_ba","has a child","wang\_er").

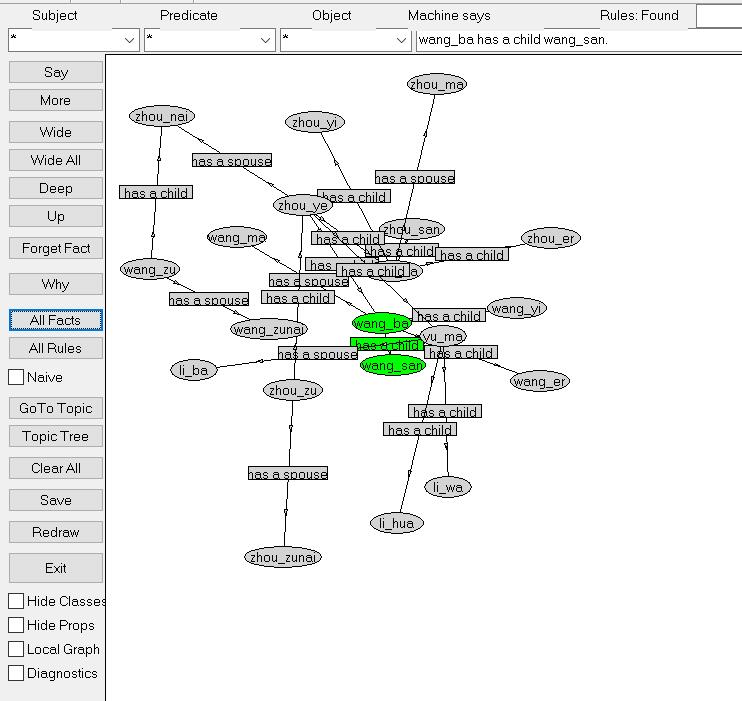
f("wang\_ba","has a child","wang\_san").

f("li\_hua","has a profession","pop star").

f("wang\_yi","has a hobby","guitar").

f("zhou\_yi","lives in","London").

After define all these facts, I run the program and get the result:



1. **Add to objects in the knowledge base attributes: profession, hobby, etc. by your choice.**

Define the 3 facts about profession, hobby, address.

Li hua has professsion of pop star.

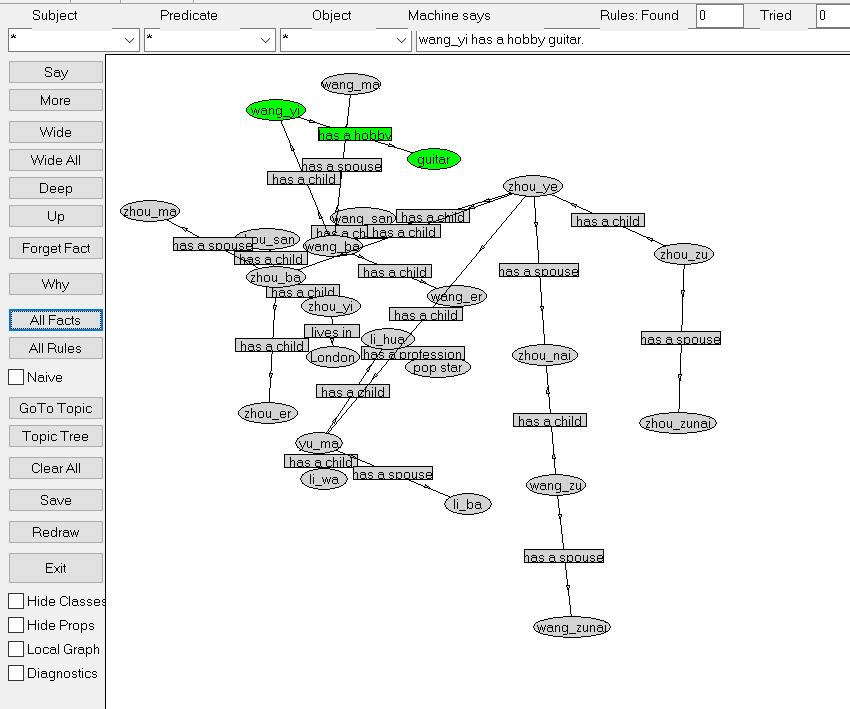
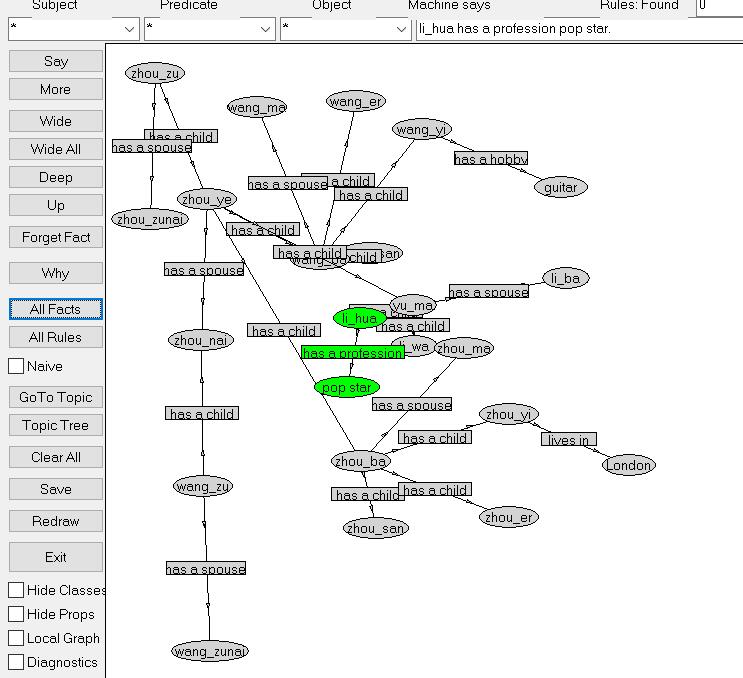
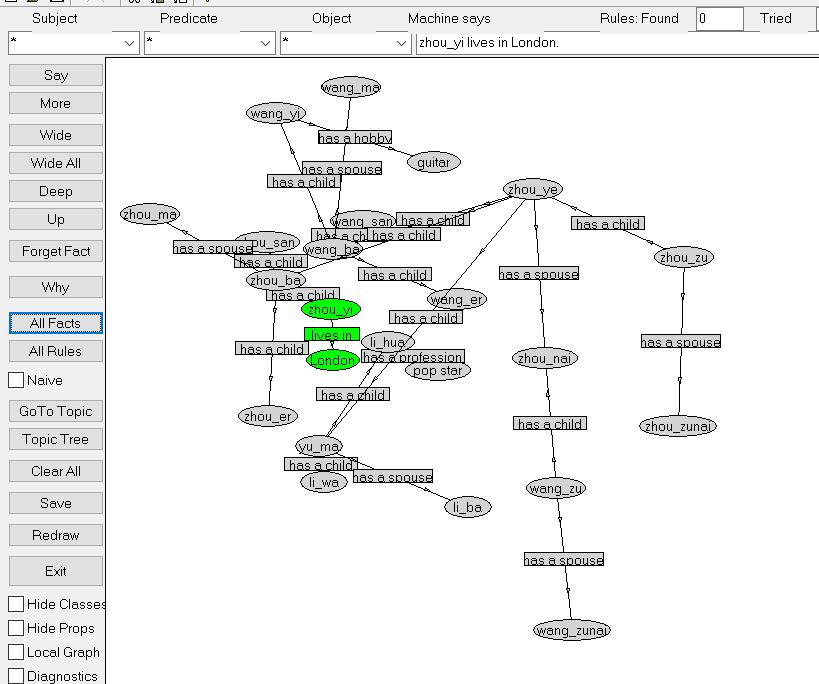
Wang yi has a hobby of guitar.

Zhou yi lives in London

f("li\_hua","has a profession","pop star").

f("wang\_yi","has a hobby","guitar").

f("zhou\_yi","lives in","London").

****

1. **Create rules defining relations: colleague, common interests, etc. At least 3 rules.Make screenshots demonstrating results of inference from your rules.**

First we should build the connection of Facts and Rules.

t("x").

onto("question.ql").

Then I define 3 rules of relationship: sibling, grandparent, grand grandparent

r([t("?x","has a child","?y"),t("?x","has a child","?z"),t("?y","differs","?z")],

[t("?y","is\_sibling","?z")]).

r([t("?x","has a child","?y"),t("?y","has a child","?z"),t("?y","differs","?z")],

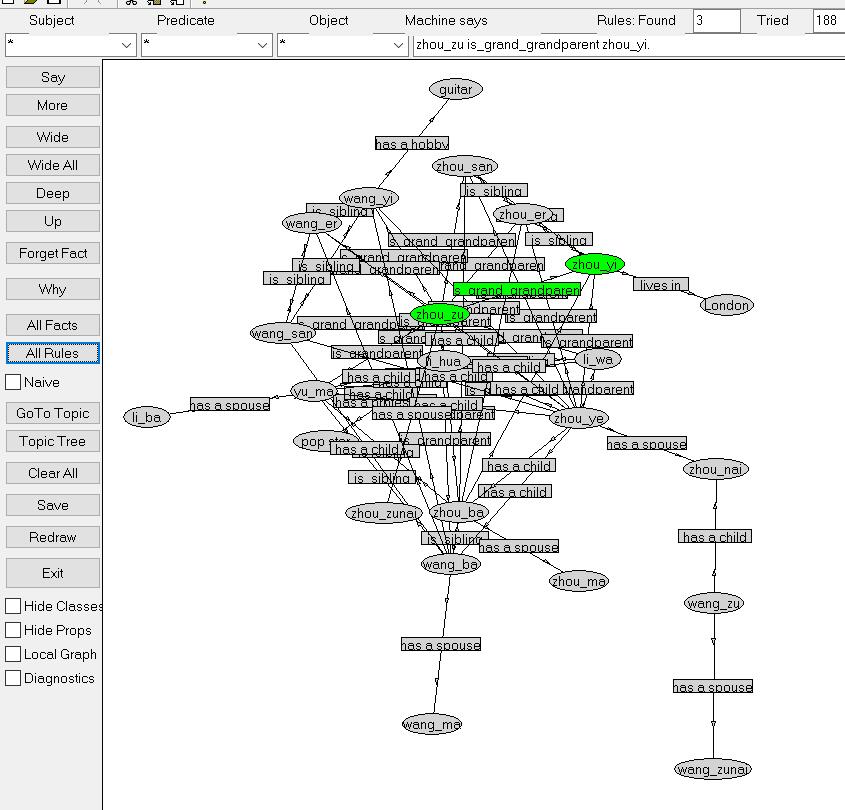
[t("?x","is\_grandparent","?z")]).

r([t("?x","has a child","?y"),t("?y","has a child","?z"),t("?z","has a child","?w"),t("?y","differs","?z")],

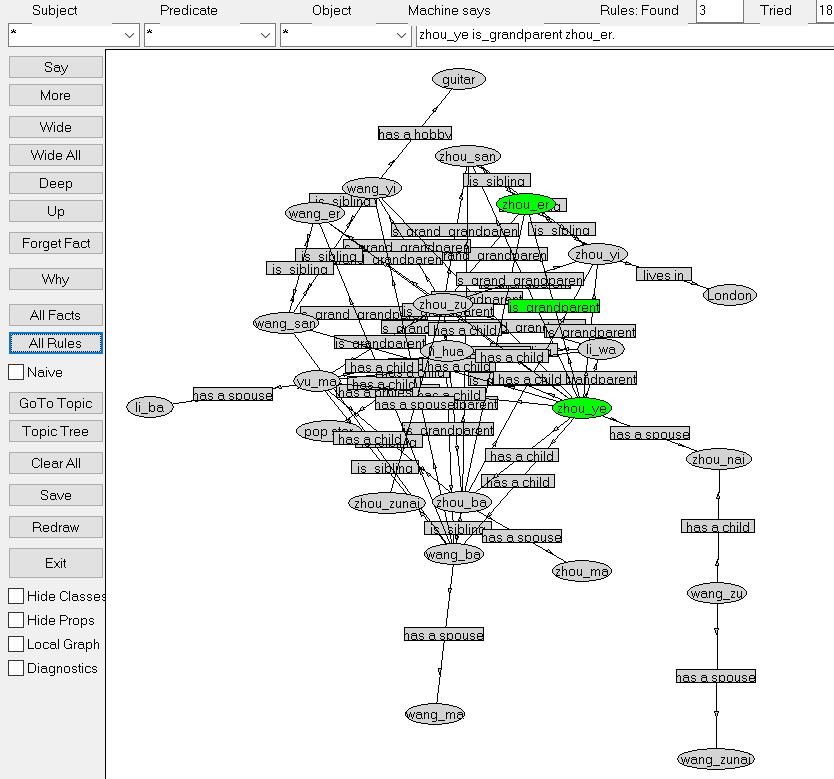
[t("?x","is\_grand\_grandparent","?w")]).

**Result :**

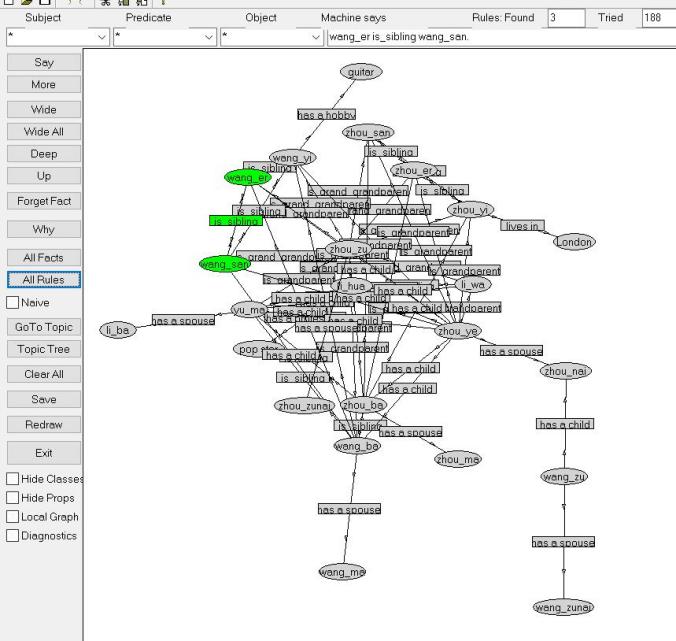
**Rules of Grand grandparent:**



**Rules of Grandparent:**



**Rules of Grand grandparent:**



1. **Summary**

This experiment taught me to write a semantic program using prolog. The main steps are：

1. Prepare the relationship of each element in advance, otherwise it will affect programming ideas.

2. Use prolog to define the relationship of each element and establish a semantic relationship diagram.

3. Define some rules based on facts. It should be noted here that facts and rules need to be defined in different files, and use on ("path") in the fact files to link to the rules files.

4. Get the final result.