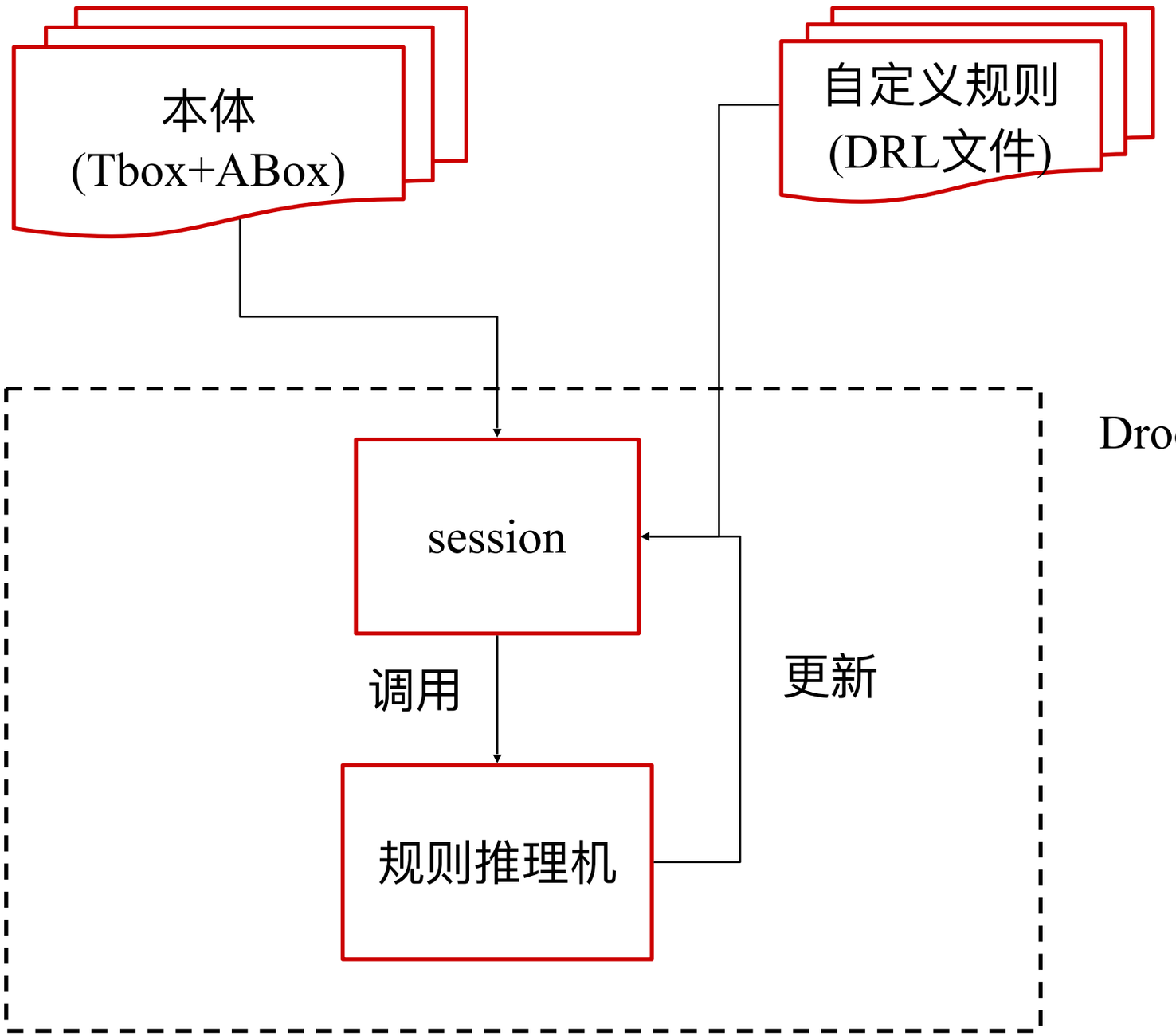
Tutorial

基于*Drools*的规则推理

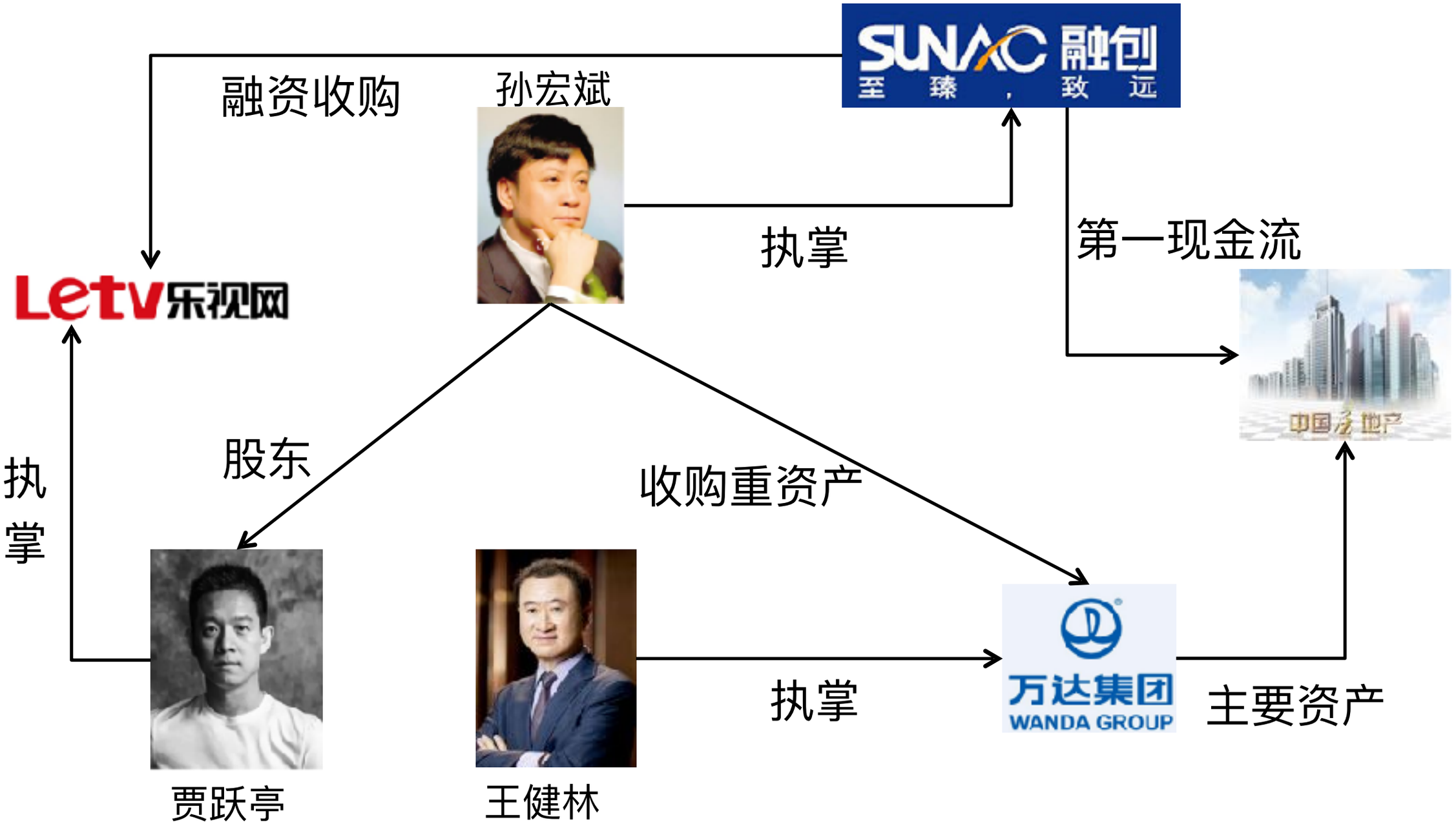
This tutorial is prepared by Prof. Guilin Qi from Southeast University

Drools实践



注意：
由于Drools不仅仅面向语义数据，因此所有三元组均需以对象形式输入

Drools实践



Drools实践

- 输入
  - TBox与ABox均以三元组的形式输入
  - 创建Triple类，每个三元组都以该类的对象输入

本体

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE rdf:RDF [
  <!ENTITY finance "http://www.example.org/kse/finance#">
  <!ENTITY owl "http://www.w3.org/2002/07/owl#">
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#">
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">
]>
<rdf:RDF xml:base="&finance;"
         xmlns:owl="&owl;"
         xmlns:rdf="&rdf;"
         xmlns:rdfs="&rdfs;">

<!-- Ontology Information -->
  <owl:Ontology rdf:about=""/>

<owl:Class rdf:ID="PublicCompany">
  <rdfs:subClassOf rdf:resource="Company"/>
</owl:Class>


<owl:ObjectProperty rdf:ID="control">
  <rdfs:domain rdf:resource="Person"/>
  <rdfs:range rdf:resource="Company"/>
</owl:ObjectProperty>

</rdf:RDF>
```

```java
public class Triple {
    private String subject;
    private String predicate;
    private String object;

    public Triple(String subject, String predicate, String object) {
        this.subject = subject;
        this.predicate = predicate;
        this.object = object;
    }
```

Triple类

三元组

```
<http://www.example.org/kse/finance> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.w3.org/2002/07/owl#Ontology> .
<http://www.example.org/kse/finance#control> <http://www.w3.org/2000/01/rdf-schema#range> <http://www.example.org/kse/Company> .
<http://www.example.org/kse/finance#control> <http://www.w3.org/2000/01/rdf-schema#domain> <http://www.example.org/kse/Person> .
<http://www.example.org/kse/finance#control> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.w3.org/2002/07/owl#ObjectProperty> .
<http://www.example.org/kse/finance#PublicCompany> <http://www.w3.org/2000/01/rdf-schema#subClassOf> <http://www.example.org/kse/Company> .
<http://www.example.org/kse/finance#PublicCompany> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.w3.org/2002/07/owl#Class> .
```

Drools实践

- 输入
  - TBox与ABox均以三元组的形式输入
  - 创建Triple类，每个三元组都以该类的对象输入

实例

```
<http://www.example.org/kse/finance#融创中国> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.example.org/kse/finance#地产事业> .
<http://www.example.org/kse/finance#孙宏斌> <http://www.example.org/kse/finance#control> <http://www.example.org/kse/finance#融创中国> .
<http://www.example.org/kse/finance#贾跃亭> <http://www.example.org/kse/finance#control> <http://www.example.org/kse/finance#乐视网> .
<http://www.example.org/kse/finance#王健林> <http://www.example.org/kse/finance#control> <http://www.example.org/kse/finance#万达集团> .
<http://www.example.org/kse/finance#孙宏斌> <http://www.example.org/kse/finance#hold_share> <http://www.example.org/kse/finance#乐视网> .
<http://www.example.org/kse/finance#万达集团> <http://www.example.org/kse/finance#main_income> <http://www.example.org/kse/finance#地产事业> .
<http://www.example.org/kse/finance#融创中国> <http://www.example.org/kse/finance#acquire> <http://www.example.org/kse/finance#乐视网> .
<http://www.example.org/kse/finance#融创中国> <http://www.example.org/kse/finance#acquire> <http://www.example.org/kse/finance#万达集团> .
```

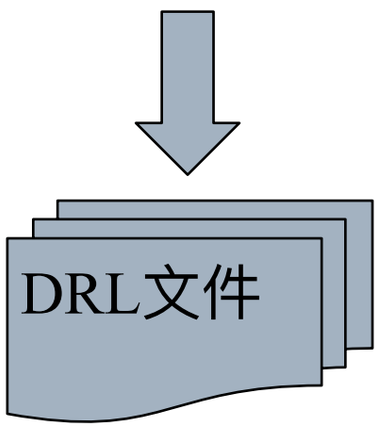
Drools实践

- 自定义规则

1) 执掌一家公司就一定是这家公司的股东；
2) 某人同时是两家公司的股东，那么这两家公司一定有关联交易；

形式化

```
finance:hold_share(X,Y) :- finance:control(X,Y).
finance:conn_trans(Y,Z) :- finance:hold_share(X,Y), finance:hold_share(X,Z).
```



Drools实践

- 自定义规则
  - DRL文件示例

```
1  package data.drools.rules;
2  dialect  "mvel"
3
4  import drools.Triple
5
6  rule "finance1"
7      when
8          c1: Triple($X: subject, predicate == "<http://www.example.org/kse/finance#control>", $Y: object)
9      then
10         insert(new Triple($X, "<http://www.example.org/kse/finance#hold_share>", $Y));
11 end
12
13 rule "finance2"
14     when
15         c1: Triple($X: subject, predicate == "<http://www.example.org/kse/finance#hold_share>", $Y: object)
16         c2: Triple(subject == $X, predicate == "<http://www.example.org/kse/finance#hold_share>", $Z: object)
17     then
18         insert(new Triple($Y, "<http://www.example.org/kse/finance#conn_trans>", $Z))
19 end
```

finance:hold_share(X,Y) :- finance:control(X,Y).

finance:conn_trans(Y,Z) :- finance:hold_share(X,Y), finance:hold_share(X,Z).

Drools实践

- Drools工程结构

```
src
  --main
    --java
      --drools
        --Drools_tutorial.java
    --resources
      --META-INF
        --kmodule.xml
      --data
        --finance.drl
        --finance-onto.owl
        --finance-onto.nt
        --finance-data.nt
```

配置文件，用于指定规则路径

数据

规则在classpath下的路径

```xml
<?xml version="1.0" encoding="UTF-8"?>
<kmodule xmlns="http://www.drools.org/xsd/kmodule">
    <kbase name="rules" packages="data">
        <ksession name="ksession-rules"/>
    </kbase>
</kmodule>
```

配置文件样例

在程序中引用此ksession的名称

- 定义了kmodule、kbase、ksession从上到下的包含关系
- 项目运行时会自动解析classpath中META-INF/kmodule.xml文件，构造相应的对象供Drools引擎使用

Drools实践

- 代码示例 (Java)

```java
// 获取drools实现的 KieServices 实例
KieServices ks = KieServices.Factory.get();
// kieServices默认加载 classpath:META-INF/kmodule.xml 得到 KieContainer
KieContainer kContainer = ks.getKieClasspathContainer();
// 通过 kContainer获取 kmodule.xml 中定义的 ksession
KieSession kSession = kContainer.newKieSession("ksession-rules");
```

初始化服务及会话

```java
// 读取本体数据
BufferedReader ontoReader = new BufferedReader(new FileReader(new File(Drools_tutorial.class.getResource("/data/finance-onto.nt").
    toURI())));
String ontoLine = null;
while((ontoLine = ontoReader.readLine()) != null){
    if(ontoLine.isEmpty())
        continue;
    else {
        String[] lineArray = ontoLine.split(" ");
        // 向WorkingMemory插入三元组
        kSession.insert(new Triple(lineArray[0], lineArray[1], lineArray[2]));
    }
}
```

读取本体中的三元组

```java
// 读取实例数据
BufferedReader dataReader = new BufferedReader(new FileReader(new File(Drools_tutorial.class.getResource("/data/finance-data.nt").
    toURI())));
String dataLine = null;
while((dataLine = dataReader.readLine()) != null){
    if(dataLine.isEmpty())
        continue;
    else{
        String[] lineArray = dataLine.split(" ");
        // 向WorkingMemory插入三元组
        kSession.insert(new Triple(lineArray[0], lineArray[1], lineArray[2]));
    }
}
```

读取数据中的三元组

Drools实践

- 代码示例 (Java)

```
System.out.println("Facts num before reasoning: " + kSession.getObjects().toArray().length);
System.out.println("Facts Before Reasoning:");
Object[] array = kSession.getObjects().toArray();
for(int i = 0; i < array.length; i++){
    System.out.println(i + ": " + array[i]);
}

long startTime = System.currentTimeMillis();
System.out.println("Execute...");
kSession.fireAllRules();
long endTime = System.currentTimeMillis();
long runningTime = endTime - startTime;

System.out.println("Facts num after reasoning: " + kSession.getObjects().toArray().length);
System.out.println("Facts After Reasoning:");
Object[] array2 = kSession.getObjects().toArray();
for(int i = 0; i < array2.length; i++){
    System.out.println(i + ": " + array2[i]);
}

System.out.println("Total time cost: " + runningTime + "ms");
```

输出推理前的三元组

进行推理

输出推理后的三元组及推理时间

Drools实践

注: 此处推理结果与RDFox略有不同是因为Drools只进行规则推理

- 结果输出

```
Triples num before reasoning: 14
Triples Before Reasoning:
1: <http://www.example.org/kse/finance#融创中国> <http://www.example.org/kse/finance#acquire> <http://www.example.org/kse/finance#万达集团> .
2: <http://www.example.org/kse/finance#control> <http://www.w3.org/2000/01/rdf-schema#range> <http://www.example.org/kse/Company> .
3: <http://www.example.org/kse/finance#孙宏斌> <http://www.example.org/kse/finance#control> <http://www.example.org/kse/finance#融创中国> .
4: <http://www.example.org/kse/finance#control> <http://www.w3.org/2000/01/rdf schema#domain> <http://www.example.org/kse/Person> .
5: <http://www.example.org/kse/finance#万达集团> <http://www.example.org/kse/finance#main_income> <http://www.example.org/kse/finance#地产事业> .
6: <http://www.example.org/kse/finance#PublicCompany> <http://www.w3.org/2000/01/rdf-schema#subClassOf> <http://www.example.org/kse/Company> .
7: <http://www.example.org/kse/finance#融创中国> <http://www.example.org/kse/finance#acquire> <http://www.example.org/kse/finance#乐视网> .
8: <http://www.example.org/kse/finance#control> <http://www.w3.org/1999/02/22 rdf syntax ns#type> <http://www.w3.org/2002/07/owl#ObjectProperty> .
9: <http://www.example.org/kse/finance#贾跃亭> <http://www.example.org/kse/finance#control> <http://www.example.org/kse/finance#乐视网> .
10: <http://www.example.org/kse/finance#孙宏斌> <http://www.example.org/kse/finance#hold_share> <http://www.example.org/kse/finance#乐视网> .
11: <http://www.example.org/kse/finance#PublicCompany> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.w3.org/2002/07/owl#Class> .
12: <http://www.example.org/kse/finance> <http://www.w3.org/1999/02/22 rdf syntax ns#type> <http://www.w3.org/2002/07/owl#Ontology> .
13: <http://www.example.org/kse/finance#融创中国> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.example.org/kse/finance#地产事业> .
14: <http://www.example.org/kse/finance#王健林> <http://www.example.org/kse/finance#control> <http://www.example.org/kse/finance#万达集团> .
Execute...
Facts num after reasoning: 23
Facts After Reasoning:
1: <http://www.example.org/kse/finance#融创中国> <http://www.example.org/kse/finance#acquire> <http://www.example.org/kse/finance#万达集团> .
2: <http://www.example.org/kse/finance#control> <http://www.w3.org/2000/01/rdf-schema#range> <http://www.example.org/kse/Company> .
3: <http://www.example.org/kse/finance#孙宏斌> <http://www.example.org/kse/finance#control> <http://www.example.org/kse/finance#融创中国> .
4: <http://www.example.org/kse/finance#融创中国> <http://www.example.org/kse/finance#conn_trans> <http://www.example.org/kse/finance#融创中国> .
5: <http://www.example.org/kse/finance#control> <http://www.w3.org/2000/01/rdf-schema#domain> <http://www.example.org/kse/Person> .
6: <http://www.example.org/kse/finance#乐视网> <http://www.example.org/kse/finance#conn_trans> <http://www.example.org/kse/finance#乐视网> .
7: <http://www.example.org/kse/finance#万达集团> <http://www.example.org/kse/finance#main_income> <http://www.example.org/kse/finance#地产事业> .
8: <http://www.example.org/kse/finance#孙宏斌> <http://www.example.org/kse/finance#hold_share> <http://www.example.org/kse/finance#融创中国> .
9: <http://www.example.org/kse/finance#融创中国> <http://www.example.org/kse/finance#conn_trans> <http://www.example.org/kse/finance#乐视网> .
10: <http://www.example.org/kse/finance#PublicCompany> <http://www.w3.org/2000/01/rdf-schema#subClassOf> <http://www.example.org/kse/Company> .
11: <http://www.example.org/kse/finance#王健林> <http://www.example.org/kse/finance#hold_share> <http://www.example.org/kse/finance#万达集团> .
12: <http://www.example.org/kse/finance#融创中国> <http://www.example.org/kse/finance#acquire> <http://www.example.org/kse/finance#乐视网> .
13: <http://www.example.org/kse/finance#control> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.w3.org/2002/07/owl#ObjectProperty> .
14: <http://www.example.org/kse/finance#贾跃亭> <http://www.example.org/kse/finance#control> <http://www.example.org/kse/finance#乐视网> .
15: <http://www.example.org/kse/finance#孙宏斌> <http://www.example.org/kse/finance#hold_share> <http://www.example.org/kse/finance#乐视网> .
16: <http://www.example.org/kse/finance#乐视网> <http://www.example.org/kse/finance#conn_trans> <http://www.example.org/kse/finance#融创中国> .
17: <http://www.example.org/kse/finance#PublicCompany> <http://www.w3.org/1999/02/22 rdf syntax ns#type> <http://www.w3.org/2002/07/owl#Class> .
18: <http://www.example.org/kse/finance> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.w3.org/2002/07/owl#Ontology> .
19: <http://www.example.org/kse/finance#万达集团> <http://www.example.org/kse/finance#conn_trans> <http://www.example.org/kse/finance#万达集团> .
20: <http://www.example.org/kse/finance#贾跃亭> <http://www.example.org/kse/finance#hold_share> <http://www.example.org/kse/finance#乐视网> .
21: <http://www.example.org/kse/finance#乐视网> <http://www.example.org/kse/finance#conn_trans> <http://www.example.org/kse/finance#乐视网> .
22: <http://www.example.org/kse/finance#融创中国> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.example.org/kse/finance#地产事业> .
23: <http://www.example.org/kse/finance#王健林> <http://www.example.org/kse/finance#control> <http://www.example.org/kse/finance#万达集团> .
Total time cost: 58ms.
```