



Corese Goodies

olivier.corby@inria.fr



Corese

- **RDF** : Resource Description Framework
- **RDFS** : RDF Schema
- **SPARQL** : Query & Update
- **Inference Rules**
- **Goodies**
- **API**

Construct + graph pattern

```
construct {  
    graph ?g {?doc c:author ?x}  
}  
  
where {  
    graph ?g {?x c:hasCreated ?doc}  
}
```

Approximate Search

```
select * where {  
  ?x rdf:type c:Engineer  
  ?doc rdf:type c:Report  
  ?x c:hasCreated ?doc  
  ?doc c:subject ?topic  
  ?topic rdf:type c:Java  
}
```

Approximate Search

```
select more * where {  
  ?x rdf:type c:Engineer  
  ?doc rdf:type c:Report  
  ?x c:hasCreated ?doc  
  ?doc c:subject ?topic  
  ?topic rdf:type c:Java  
}  
  
c:Engineer -> c:Team  
c:Report -> c:Slide
```

Approximate Search

```
select more * (kg:similarity() as ?sim)
  where {
    ?x rdf:type c:Engineer
    ?doc rdf:type c:Report
    ?x c:hasCreated ?doc
    ?doc c:subject ?topic
    ?topic rdf:type c:Java
  }
order by desc(?sim)
```

Approximate Search

- Best approximation according to ontology

- Exemple:

- Query

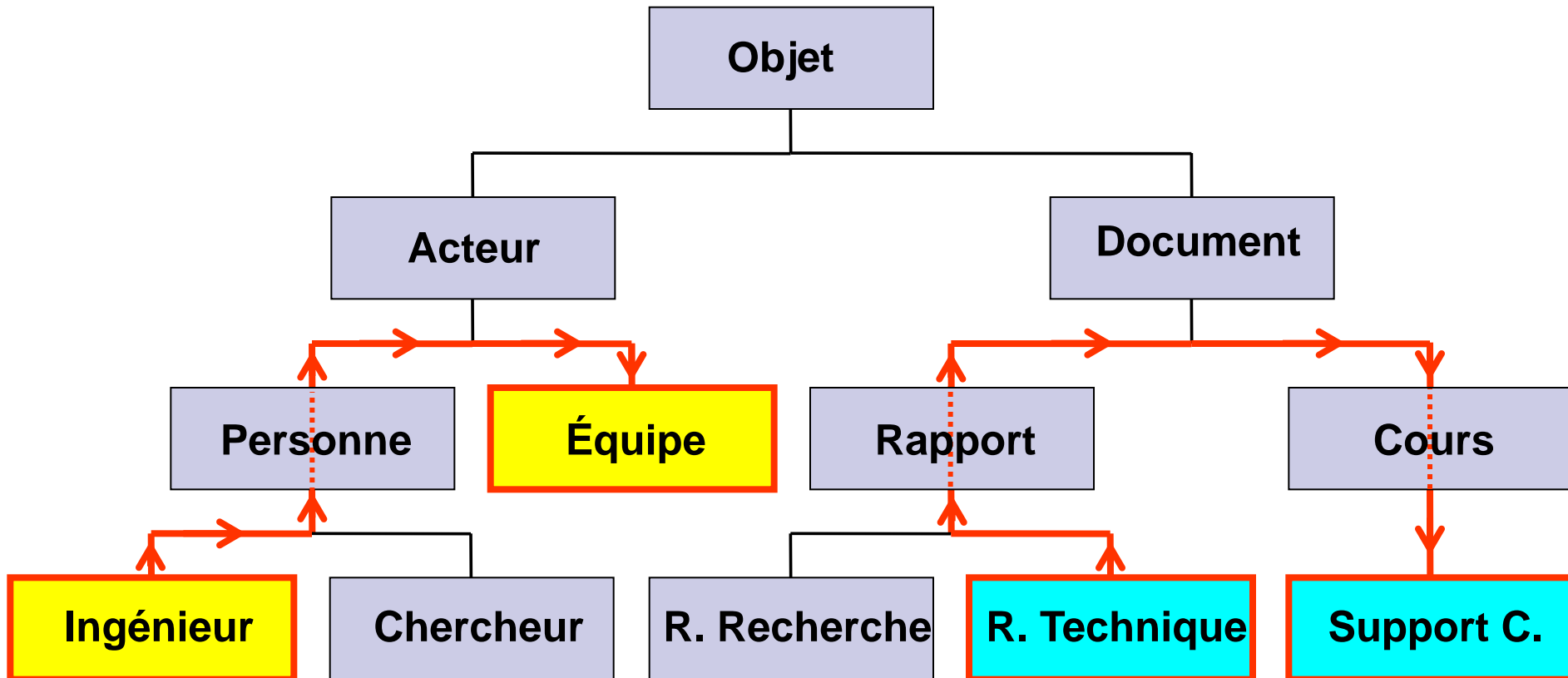
Rapport Technique sur Java écrit par un ingénieur ?

- Approximate answer

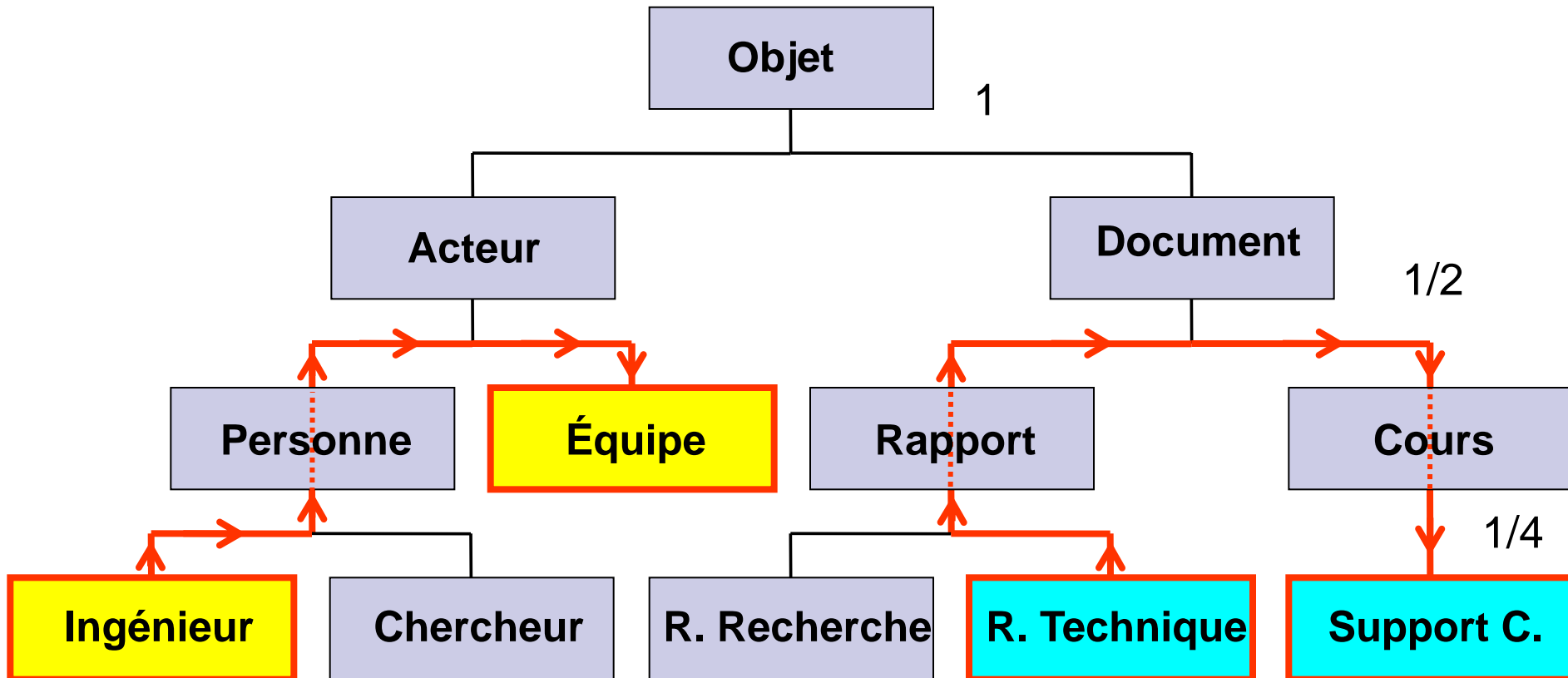
Rapport Technique → Support de cours

Ingénieur → Equipe

Distance in Ontology



Distance in Ontology



SPARQL Path

```
select * where {  
  ?list rdf:rest*/rdf:first ?elem  
}
```

Path Variable

```
select * where {  
    ?list rdf:rest*/rdf:first :: $path ?elem  
}
```

Path Length

```
select * (pathLength($path) as ?l) where {  
  ?list rdf:rest*/rdf:first :: $path ?elem  
}
```

Path Length

```
select * (pathLength($path) as ?l) where {  
    ?l list rdf:rest*/rdf:first :: $path ?elem  
}  
  
order by ?l
```

Path Length

```
select * (pathLength($path) as ?l) where {  
  ?l list rdf:rest*/rdf:first :: $path ?elem  
  filter(pathLength($path) <= 10)  
}
```

Path Edge Enumeration

```
select ?a ?b where {  
  ex:John foaf:knows* :: $path ?x  
  graph $path { ?a ?p ?b }  
}
```

Path Constraint

```
select ?a where {  
  ex:John foaf:knows* :: $path ?x  
  graph $path { ?a ?p ex:Jack }  
}
```


Path Constraint

```
select * where {
```

```
  ex:John foaf:knows [a foaf:Person]* ?x
```

```
}
```

Path Compiler

```
select * where {  
  ex:John (ex:p1/ex:p2)+ ?x  
}  
  
pragma {kg:path kg:expand 2}
```

Path Compiler

```
select * where {  
  {ex:John ex:p1 ?x1 . ?x1 ex:p2 ?x} union  
  {ex:John ex:p1 ?x1 . ?x1 ex:p2 ?x2  
    ?x2 ex:p1 ?x3 . ?x3 ex:p2 ?x}  
}
```

RDF & XML Literal

```
<rdf:Description>
```

```
<ex:feature rdf:parseType='&rdf;XMLLiteral'>
```

```
  <object>
```

```
    <id>314-2718</id>
```

```
    <label>name</label>
```

```
  </object>
```

```
</ex:feature>
```

```
</rdf:Description>
```

RDF & XML Literal

```
select * where {  
  ?x ex:feature ?obj  
  bind(xpath(?obj, '/object/label' ) as ?name)  
}
```

```
<rdf:Description>  
<ex:feature rdf:parseType='&rdf;XMLLiteral'>  
  <object>  
    <id>314-2718</id>  
    <label>name</label>  
  </object>  
</ex:feature>  
</rdf:Description>
```

RDF & XML Literal

```
select * where {  
  ?x ex:feature ?obj  
  bind(xpath(?obj, '/object/label' ) as ?name)  
}
```

```
<rdf:Description>
```

```
<ex:feature rdf:parseType='&rdf;XMLLiteral'>
```

```
  <object>
```

```
    <id>314-2718</id>
```

```
    <label>name</label>
```

```
  </object>
```

```
</ex:feature>
```

```
</rdf:Description>
```

SQL

```
construct {  
  ?emp ex:name ?name  
}  
  
where {  
  select  
  
    (kg:sql ( 'dbname' ,  
      'select emp, name from EMPLOYEE where age >= 18' )  
    as (?emp, ?name))  
  
  where {}  
}
```

SPARQL Template Transformation

```
template {  
    "allValuesFrom(" ?p " " ?c ")"  
}  
  
where {  
    ?in a owl:Restriction ;  
        owl:onProperty ?p  
        owl:allValuesFrom ?c  
}
```


SPARQL Template Transformation

- OWL
- SPIN (SPARQL in RDF)
- Turtle
- HTML
- SPARQL Query Results