CycQL: A SPARQL Adapter for OpenCyc

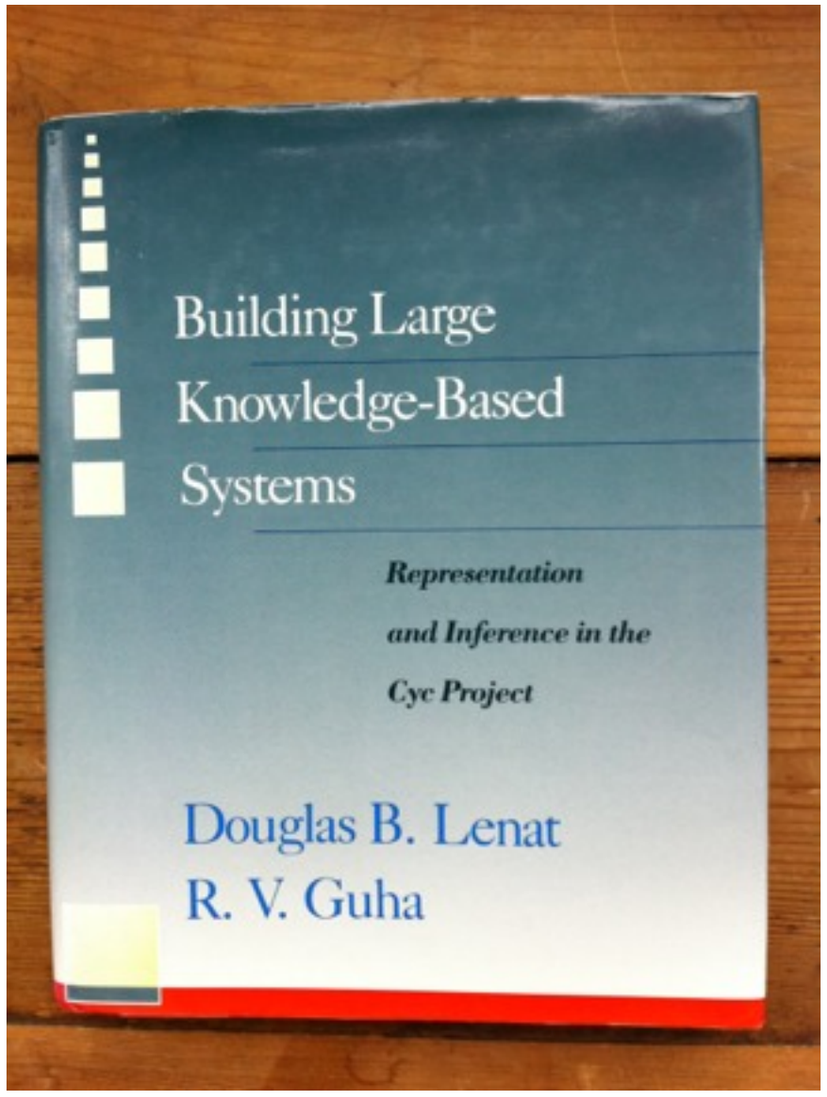
Steve Battle
@stevebattle
stevebattle@me.com
http://www.stevebattle.me
http://code.google.com/p/gloze/

OpenCyc 4.0

- Cyc was one of the first Artificial Intelligence systems to develop an ontological approach to organizing knowledge.

- OpenCyc 4.0, released in June 2012, includes the full Cyc ontology.

- Java API.



CycL

- A declarative language based on classical first-order logic.
- Includes *Atomic* and *Non-Atomic Terms* (NATs).

```
(#$orbits #$PlanetPluto #$TheSun)

(#$isa #$PlanetPluto #$DwarfPlanet)

(#$orbitalPeriod #$PlanetPluto (#$DaysDuration 90739))

(#$isa #$Charon-MoonOfPluto (#$MoonFn #$PlanetPluto))

(#$and
  (#$isa ?PLANET #$DwarfPlanet)
  (#$orbits ?PLANET #$TheSun)
)
```

RDF(CycL)

- Set the base URI base and prepend to Atomic terms.
  e.g. to <<http://sw.opencyc.org/2012/05/10/concept/en/>
- Transpose the order of the terms to give the typical subject, predicate, object.

```
@base <http://sw.opencyc.org/2012/05/10/concept/en/> .

<PlanetPluto> <orbits> <TheSun> .
<PlanetPluto a <DwarfPlanet> .
<Charon-MoonOfPluto> <orbits> <PlanetPluto> .
<PlanetPluto> <orbitalPeriod> "90739"^^<DaysDuration> .

<PlanetPluto> <MoonFn> _:MoonOfPluto .
<Charon-MoonOfPluto> a _:MoonOfPluto .
```

SPARQL(RDF(CycL))

```
PREFIX : <>
SELECT ?planet ?moon
WHERE {
  ?planet a :DwarfPlanet ;
    :orbits :TheSun
  ?moon :orbits ?planet .
}
```

```
(#$and
  (#$isa ?PLANET #$DwarfPlanet)
  (#$orbits ?PLANET #$TheSun)
  (#$orbits ?MOON ?PLANET)
)
```

planet	moon
:PlanetPluto	:Charon-MoonOfPluto
:PlanetPluto	:Nix-MoonOfPluto
:PlanetPluto	:Hydra-MoonOfPluto
:PlanetPluto	:Vulcan-MoonOfPluto
:PlanetPluto	:Cerberus-MoonOfPluto

Full-query compilation issues:

- OPTIONAL left-join
- ORDER BY clause
- Named GRAPH clause

Microtheories

- Cyc's knowledge-base is divided into microtheories.
- Microtheories are hierarchically organized.

CurrentWorldDataCollectorMt-NonHomocentric

```
(#$isa #$PlanetPluto #$DwarfPlanet)
```

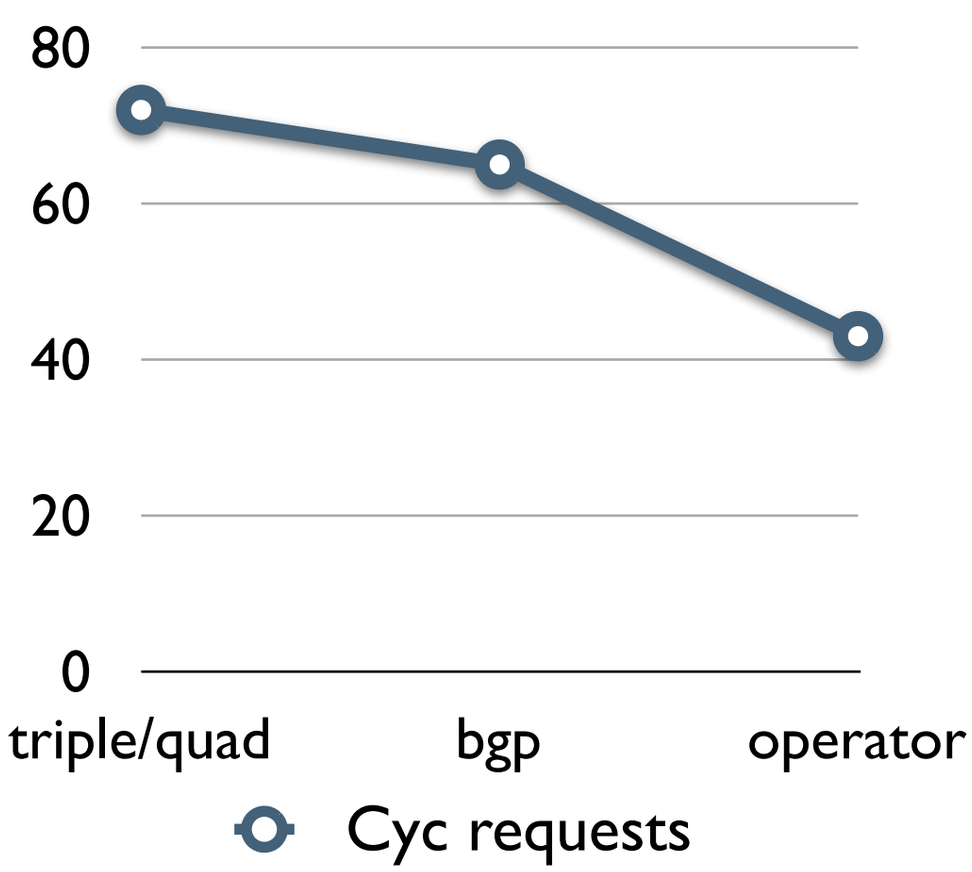
UniverseDataMt

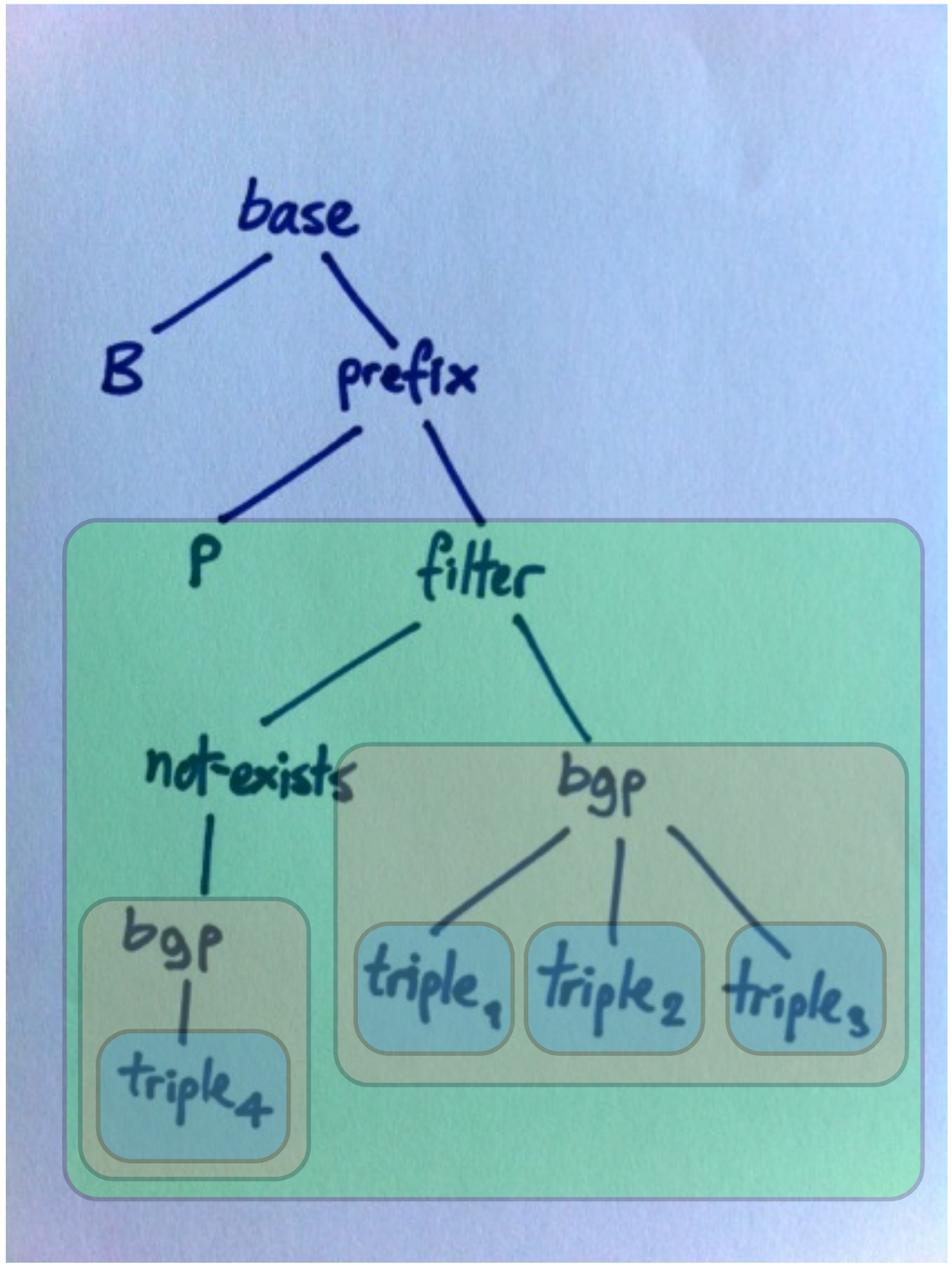
```
(#$orbits #$PlanetPluto #$TheSun)
(#$orbits #$PlanetEarth #$TheSun)
```

```
SELECT * FROM :CurrentWorldDataCollectorMt-NonHomocentric
FROM NAMED :UniverseDataMt
WHERE {
    GRAPH :UniverseDataMt {
        ?planet a :Planet ; :orbits :TheSun
    }
    FILTER (NOT EXISTS { ?planet a :DwarfPlanet })
}
```

Compilation Levels

```
PREFIX : <>
SELECT ?planet ?orbital_period
FROM :CurrentWorldDataCollectorMt-NonHomocentric
WHERE {
  ?planet a :Planet ;
    :orbits :TheSun ;
    :orbitalPeriod ?orbital_period
  FILTER (NOT EXISTS {
    ?planet a :DwarfPlanet
  })
}
```





CycL Functions

- CycL defines functions expressed as NATs.
  e.g. `#$ExponentFn, #$QuotientFn`

- This expression defines Kepler's 3rd law,
  the relationship between the orbital radius in AUs,
  and its orbital period, P:

```
(#$evaluate ?AU (#$ExponentFn (#$QuotientFn ?P 365) (#$QuotientFn 2 3)))
```

- Expressed in SPARQL using ARQ's LET clause:

```
PREFIX cyc: <http://www.opencyc.org#>
PREFIX nat: <java:org.opencyc.sparql.function.nat.>
PREFIX : <>
SELECT ?planet ?AU FROM :UniverseDataMt
WHERE {
  ?planet a :Planet ;
    :orbits :TheSun ; :orbitalPeriod ?p
    LET (?AU :=
    cyc:ExponentFn(cyc:QuotientFn(nat:Double(?p),365), cyc:QuotientFn(2,3)))
}
ORDER BY ?AU
```

CycL Rules

- The real purpose of CycQL is to provide access to Cyc inference from SPARQL.
- Define a CycL rule that computes the orbital radius:

```
(#$implies
  (#$and
    (#$orbitalPeriod ?BODY (#$DaysDuration ?P))
    (#$evaluate ?AU
     (#$ExponentFn (#$QuotientFn ?P 365)
       (#$QuotientFn 2 3))))
  (#$orbitalRadius ?BODY (#$AstronomicalUnits ?AU))
)
```

Magic Predicates

- The *Goldilock's Zone* lies between 0.725 and 3 AUs.

```
PREFIX nat: <java:org.opencyc.sparql.function.nat.>
PREFIX : <>
SELECT * FROM :UniverseDataMt
WHERE {
  ?planet a :Planet ;
    :orbits :TheSun ;
    :orbitalRadius ?AU
  FILTER (0.725 < nat:Double(?AU)
        && nat:Double(?AU) < 3.0)
}
ORDER BY nat:Double(?AU)
```

When Cyc searches for `#$orbitalRadius` backward-chaining rule evaluation is triggered.

```
-----------------------------------------------
| planet        | AU                          |
===============================================
| :PlanetEarth  | "1.0"^^xsd:double           |
| :PlanetMars   | "1.524436183195034"^^xsd:double |
-----------------------------------------------
```

Conclusions

- CyCQL-2.0 is open source <http://code.google.com/p/gloze/>.
- A useful addition to the OpenCyc tool-box.
- Jena/ARQ is a flexible framework for developing SPARQL adapters.
- Enables backward-chaining reasoning in RDF.
- Need to develop the RDF mapping to support NARTs (Non-Atomic Reified Terms) e.g. `(#$MoonFn #$PlanetPluto)`
- Support multiple FROM clauses?
- Add SPARQL 1.1 Update