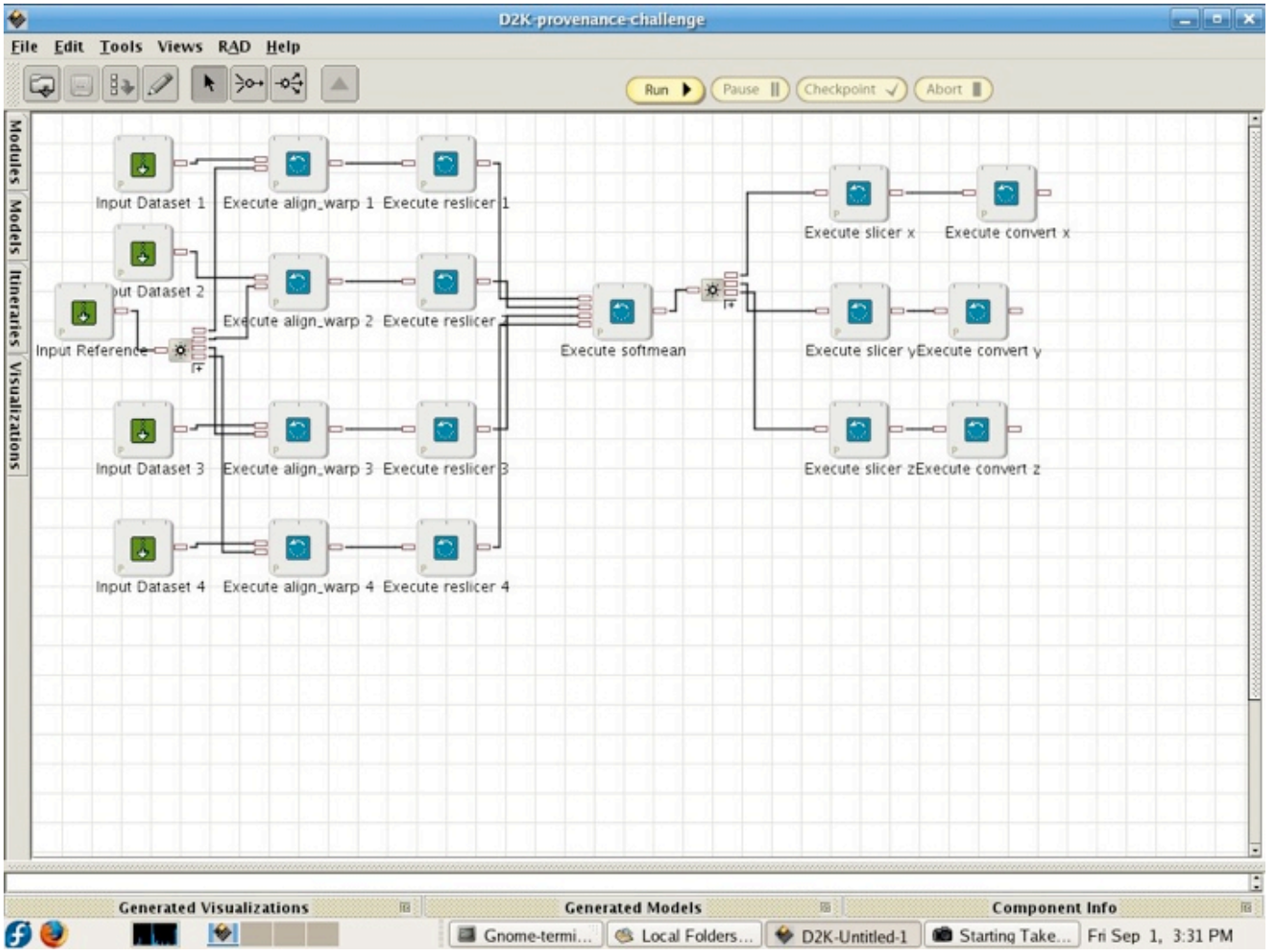
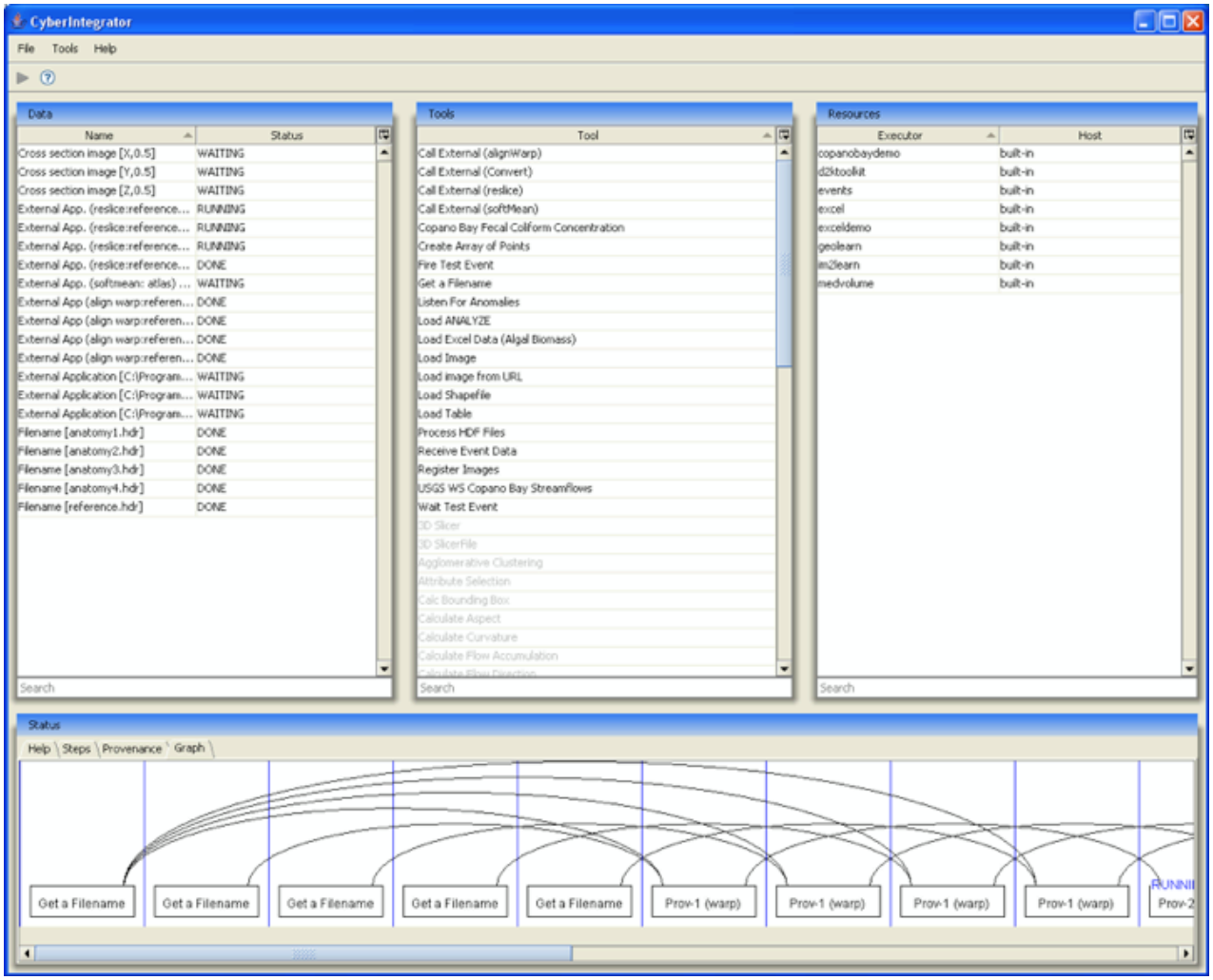
NCSA provenance challenge

- **Two workflow implementations**
  - D2K modules and itinerary
  - CyberIntegrator / im2learn tools and meta-workflow
- **Common execution trace format**
  - RDF
- **No common vocabulary or ontology**
  - D2K / CI teams developed execution trace formats independently w/o coordination

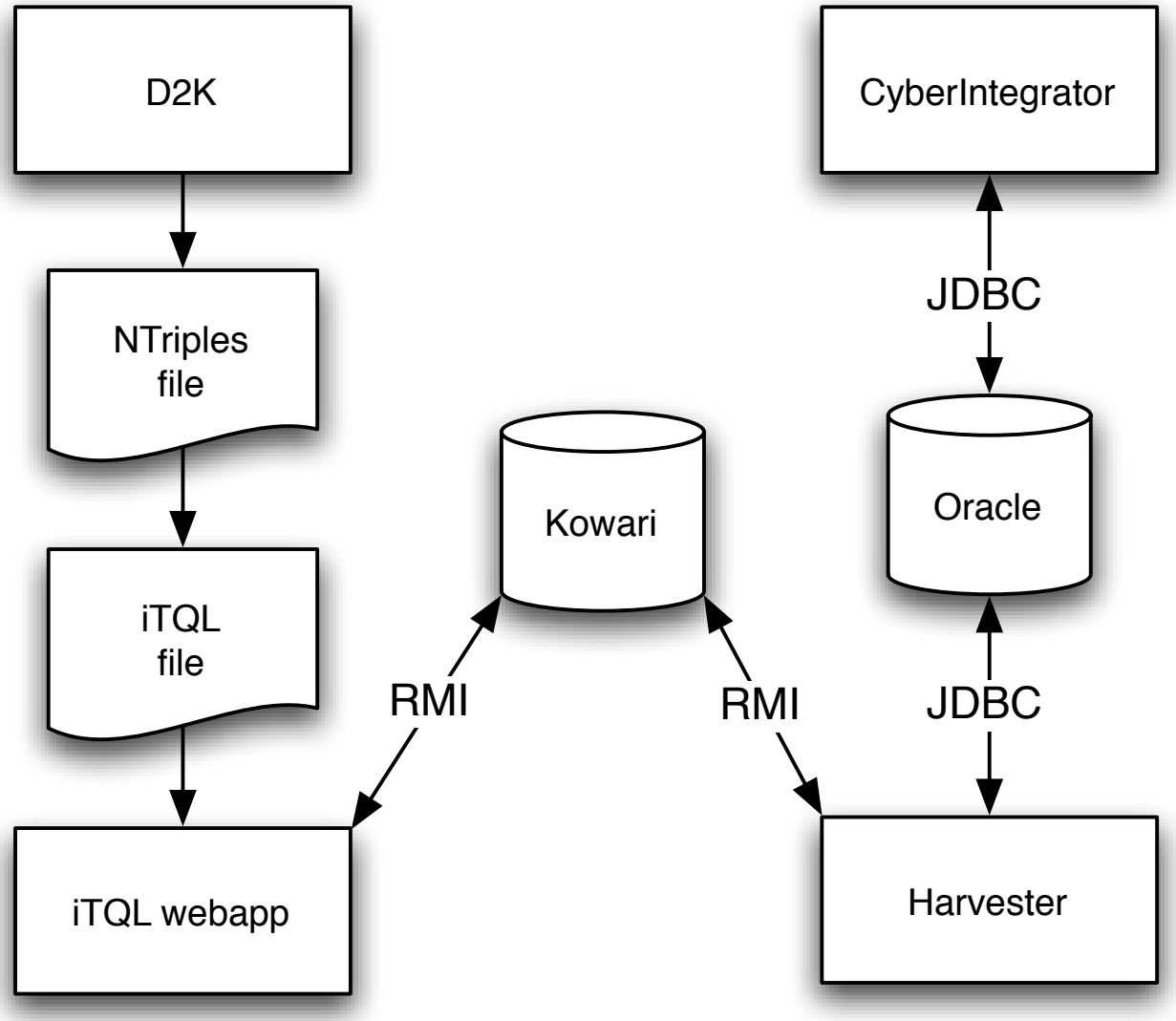
D2K implementation



CyberIntegrator implementation



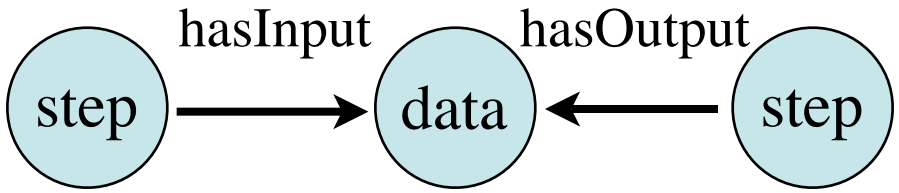
Collecting the execution traces



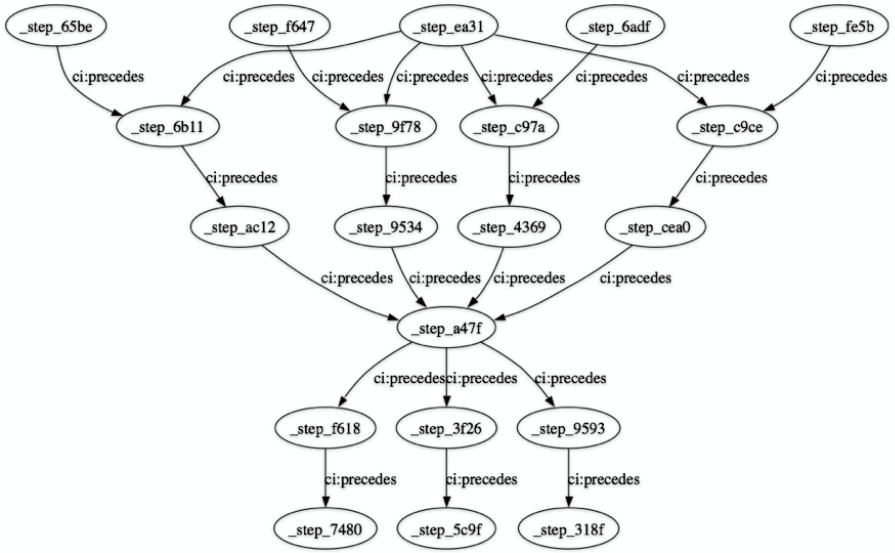
Answering the queries

- **RDF loaded into Kowari 1.2**
- **Guessed semantics**
  - properties named things like “hasInput”
  - inferred object classes (e.g., inputs, parameters) from associated properties
  - guessed what literals meant (e.g., “OK”)
- **Wrote iTQL to answer queries**
  - identify nodes representing answer (e.g., “find all invocations of ...”)
  - added external-to-workflow facts as required

Indexing precedence



CyberIntegrator



D2K

“fan out”

What's cool

- **D2K / CyberIntegrator teams worked independently on trace format**
  - no formal ontologies, identifier schemes
  - major problems with implied ontologies, but queries could still be answered
- **RDF / iTQL allows integrating multiple ontologies**
  - workflow trace + annotation
  - indexing (e.g., precedence)
  - can store either trace in any triple store
  - (SPARQL doesn't do transitive closure)

Discussion

- **How similar are the implied ontologies used by these tools?**
  - if the ontologies were explicit, how much could we do without having to hand-tune queries? (owl:sameAs? rules?)
  - how similar could they be? is there a useful taxonomy of workflow execution traces?
- **What about provenance outside of workflows?**
  - can we generalize the execution trace ontology to other cause/effect chains?