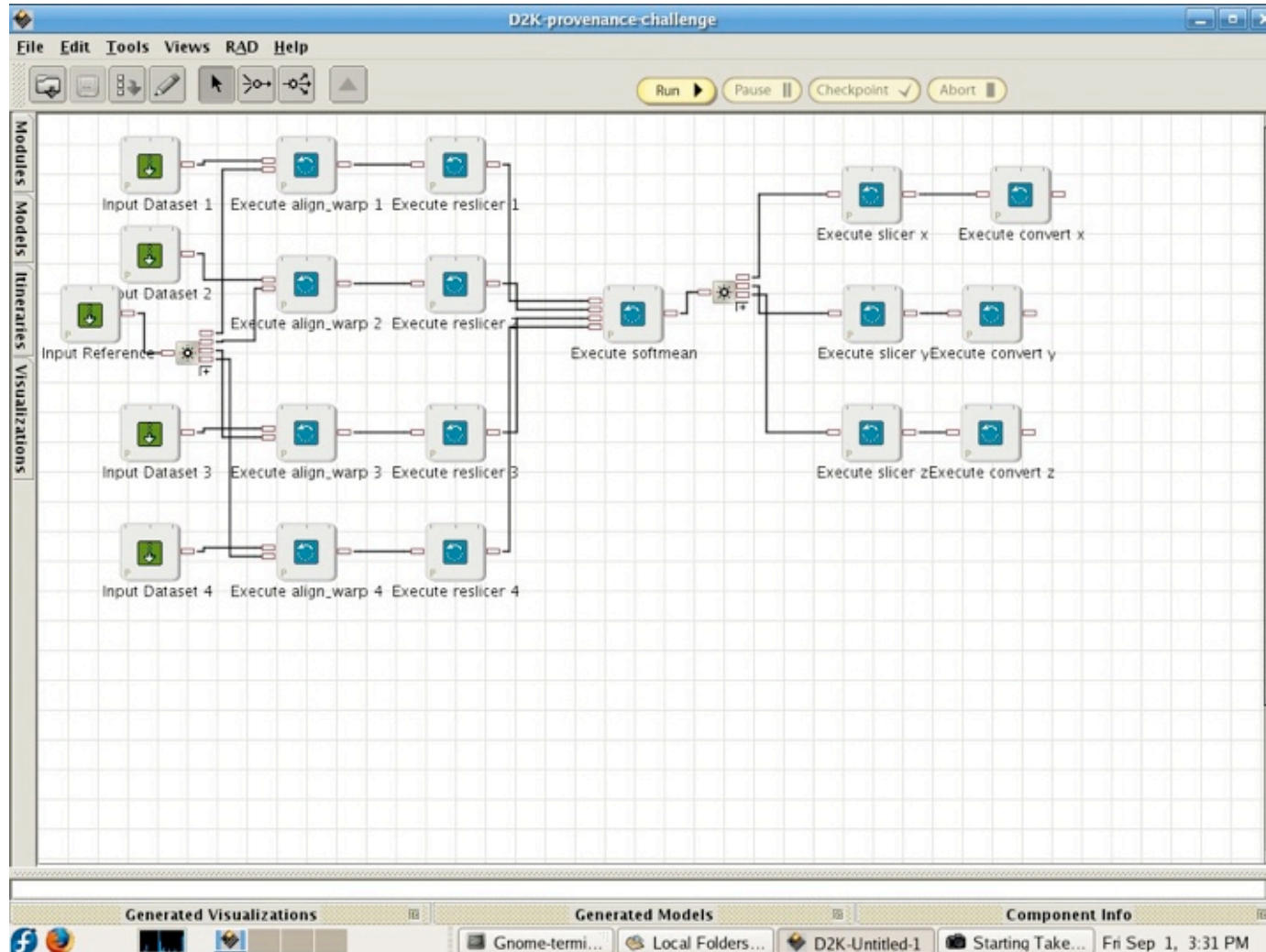


# NCSA provenance challenge

- **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

The screenshot displays the CyberIntegrator application window, which is divided into several functional panels:

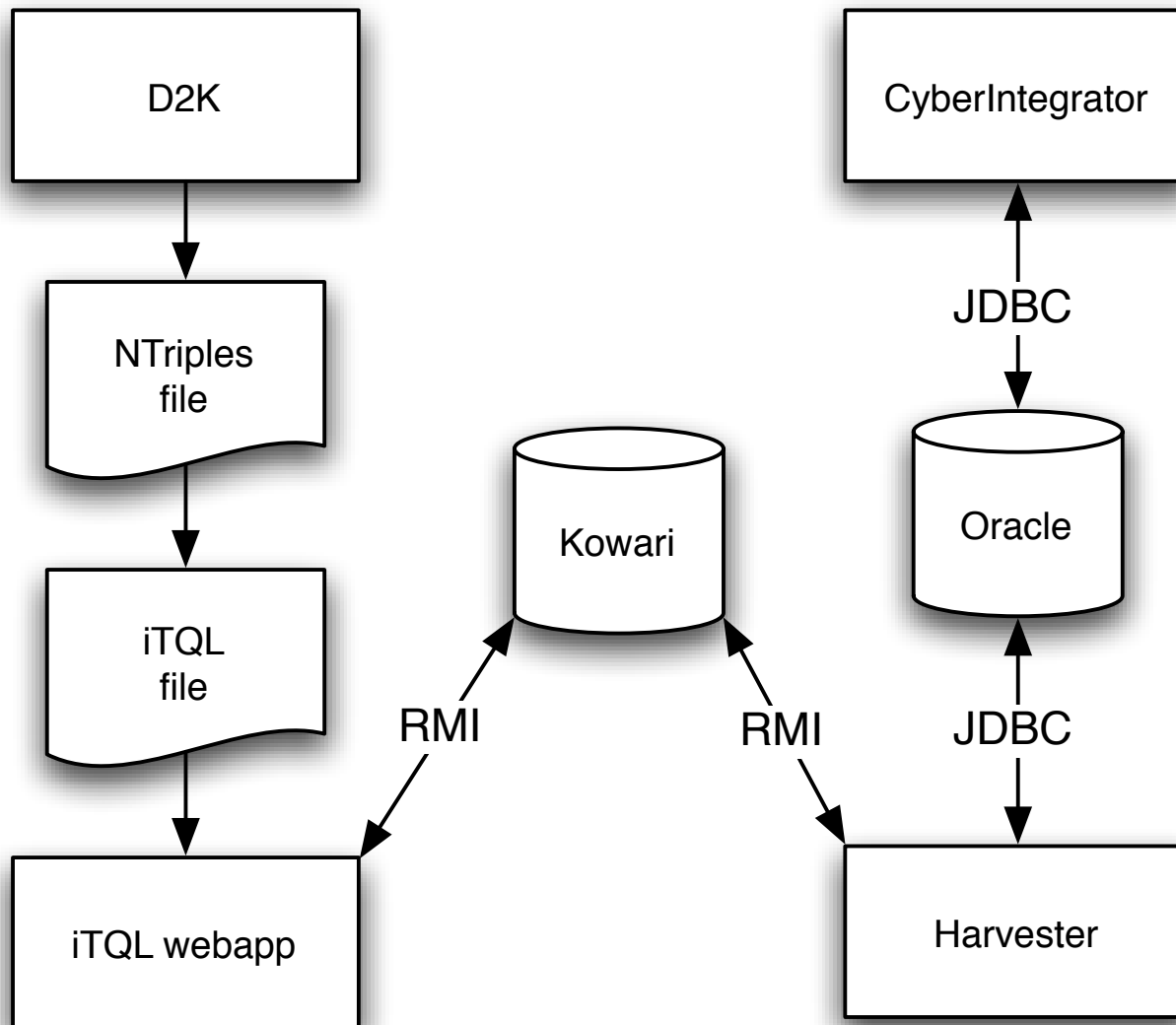
- Data Panel:** A table listing various data items and their current status. The status column includes WAITING, RUNNING, and DONE.
- Tools Panel:** A list of available tools, such as 'Call External (alignWarp)', 'Load Image', and 'Calculate Flow Accumulation'.
- Resources Panel:** A table listing resources with columns for 'Executor' and 'Host', showing built-in resources like 'copanobaydemo' and 'd3ktoolkit'.
- Status Panel:** A workflow graph showing the execution flow between tasks. Tasks include 'Get a Filename' and 'Prov-1 (warp)'. A 'RUNNING' indicator is visible on the right side of the graph.

Name	Status
Cross section image [X,0.5]	WAITING
Cross section image [Y,0.5]	WAITING
Cross section image [Z,0.5]	WAITING
External App. (reslice:reference...	RUNNING
External App. (reslice:reference...	RUNNING
External App. (reslice:reference...	DONE
External App. (softmean: atlas) ...	WAITING
External App. (align warp:referen...	DONE
External App. (align warp:referen...	DONE
External App. (align warp:referen...	DONE
External App. (align warp:referen...	DONE
External Application [C:\Program...	WAITING
External Application [C:\Program...	WAITING
External Application [C:\Program...	WAITING
Filename [anatomy1.hdr]	DONE
Filename [anatomy2.hdr]	DONE
Filename [anatomy3.hdr]	DONE
Filename [anatomy4.hdr]	DONE
Filename [reference.hdr]	DONE

Tool
Call External (alignWarp)
Call External (Convert)
Call External (reslice)
Call External (softMean)
Copano Bay Fecal Coliform Concentration
Create Array of Points
Fire Test Event
Get a Filename
Listen For Anomalies
Load ANALYZE
Load Excel Data (Algal Biomass)
Load Image
Load image from URL
Load Shapefile
Load Table
Process HDF Files
Receive Event Data
Register Images
USGS WS Copano Bay Streamflows
Wait Test Event
3D Slicer
3D SlicerFile
Agglomerative Clustering
Attribute Selection
Calc Bounding Box
Calculate Aspect
Calculate Curvature
Calculate Flow Accumulation
Calculate Flow Direction

Executor	Host
copanobaydemo	built-in
d3ktoolkit	built-in
events	built-in
excel	built-in
exceldemo	built-in
geolearn	built-in
im2learn	built-in
medvolume	built-in

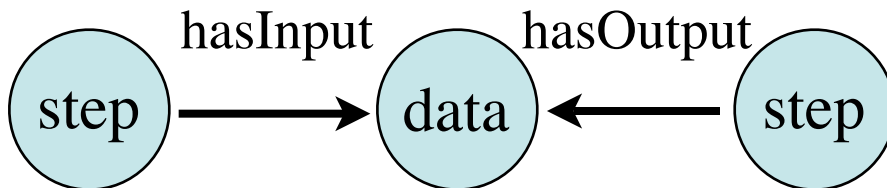
# Collecting the execution traces



# 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

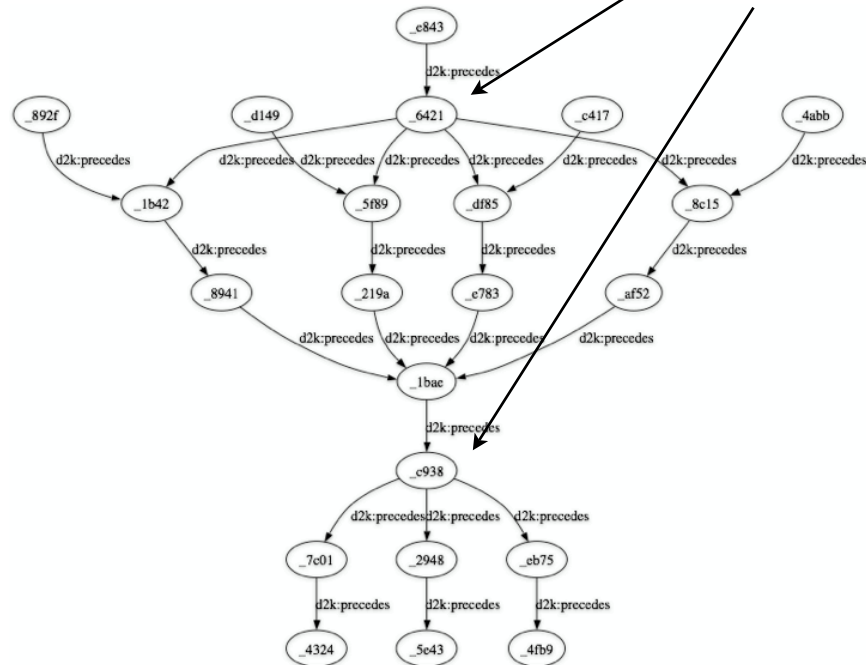
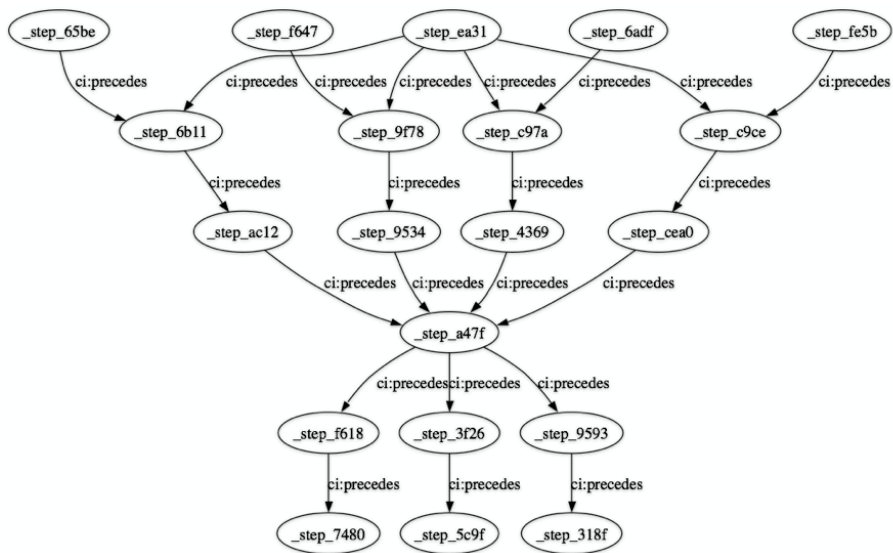
# Indexing precedence



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?