A Unified Relational Approach to Grid Information Services

(GWD-GIS-012-1 (Informational))

Peter A. Dinda, Northwestern

Beth Plale, Georgia Tech

Claim

Applications need *common*[1] *compositional*[2] queries over information of *varying dynamicity*[3]

Approach

Build down from an RDBMS world-view

Relational = relational data model and queries

Unified = tables and streams

Research Questions

How “far down” must we go?

What extensions are needed?

Specific Components

- Extensible type hierarchy
- Extensible schemas and indices
- Data streams as relations
- High update rates and freshness
- Compositional queries -> joins
- Time-bounded non-deterministic queries
- Friendly interfaces for non-experts
- Decentralized administration and data

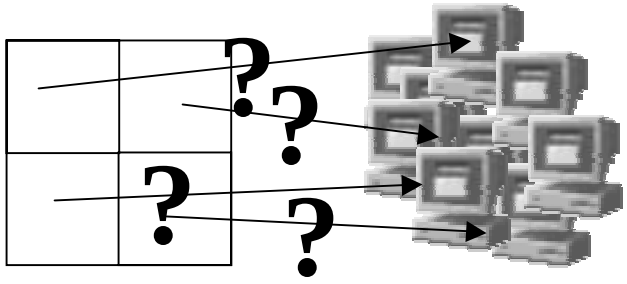
Outline

- Needs of Grid applications
- Limitations of current models
- Our approach (and research)
  - Prototype system
  - Schema and indices (including example)
  - Fast updates and streaming
  - Data stream support with dQUOB
  - Time-bounded non-deterministic queries

Needs of Grid Applications

- Compositional queries
  - Application-specific information aggregration
- Support for information of varying dynamicity
  - Varying update rates and freshness requirements
  - Seamless inclusion of streaming data
- A common data and query model
  - Powerful, high level, declarative, easy-to-optimize

Data Parallel SOR



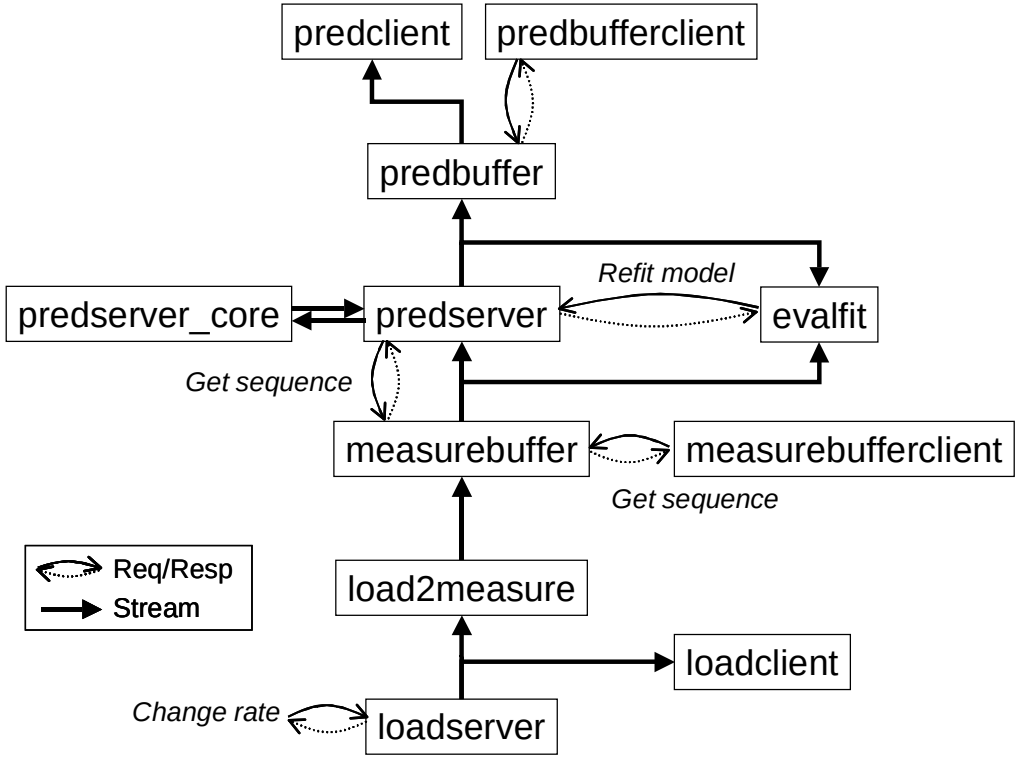
- Startup: "Find 4 hosts which all have the same architecture and have a combined memory of at least 2 GB and whose network path bandwidths to each other are comparable"

  Compositional Query Over Static Information

- Adaptation: "Tell me about instances in which the predicted load on any one of those 4 hosts exceeds the average of their predicted loads by 50%"

  Compositional Query Over Dynamic Information

Resource Prediction System



- Software Configuration Management: "For each of those hosts, find an RPS prediction stream corresponding to a measurement stream from a load sensor on the host"

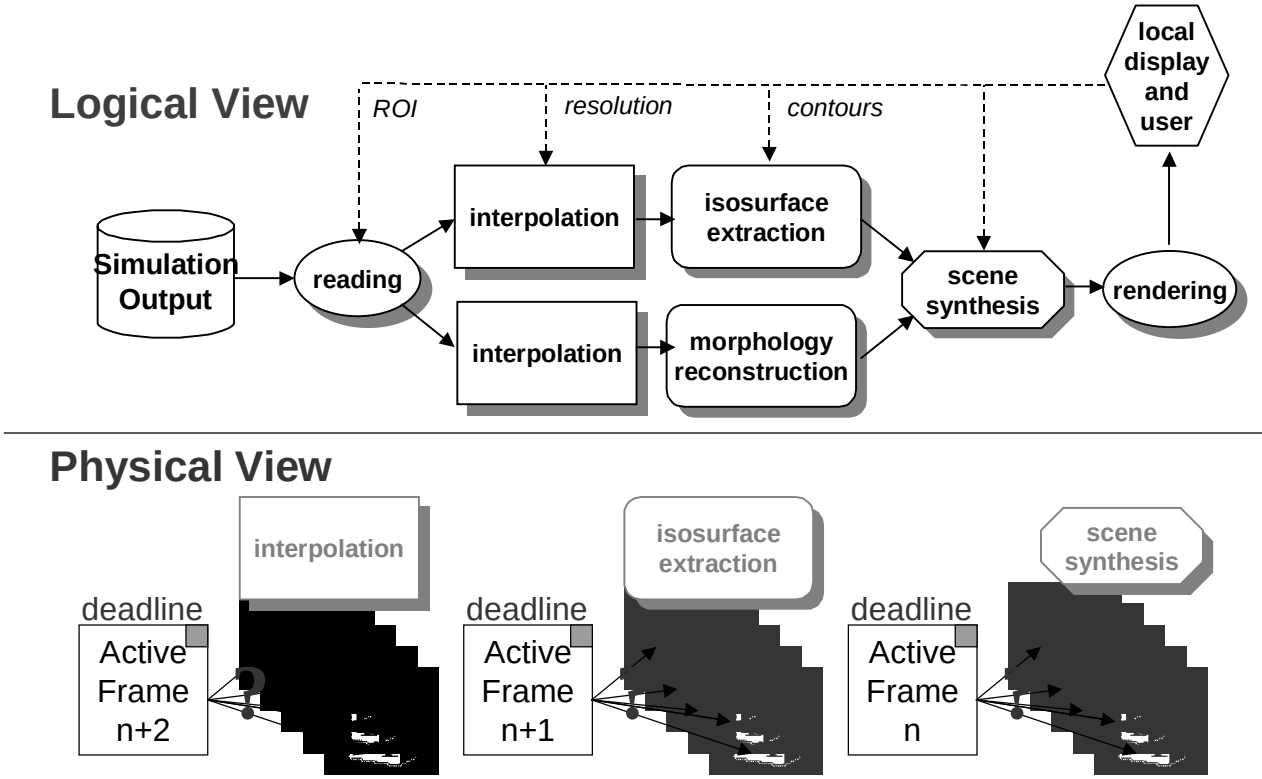
  Compositional Query Over Semistatic Information

- Performance Monitoring Streams: "Tell me about instances in which the predicted load on any one of those 4 hosts exceeds the average of their predicted loads by 50%"

  Compositional Query Over Dynamic Streams

Dv

(and traditional workflow)



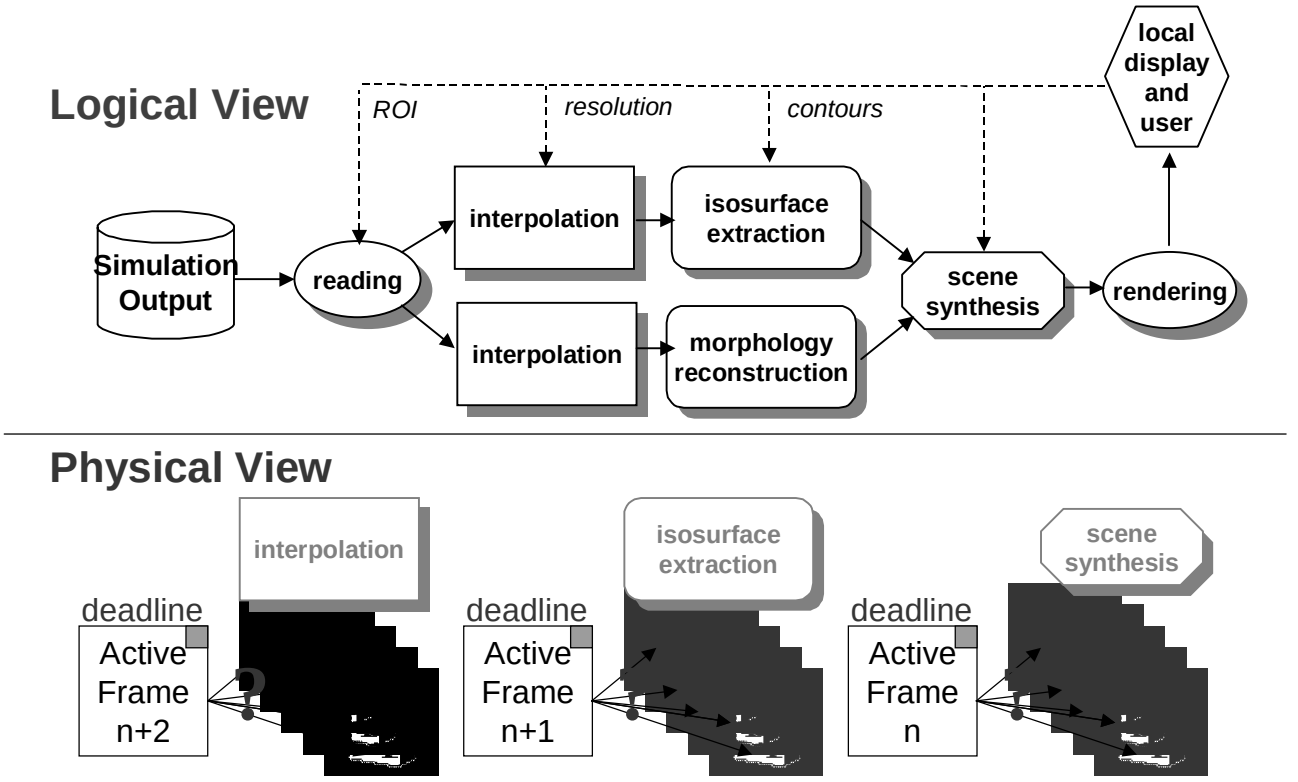
- Startup: "Find a pool of five hosts each of which have at least a GB of memory for interpolation, a second pool of five different hosts with at least 1 GFLOP/s performance for isosurface extraction, and a third pool of five different hosts with special scene synthesis hardware, where the inter-pool bandwidth is at least 10 MB/s."

  Compositional Query Over Static Information

- Adaptation: "What is the host within the isosurface extraction pool which is expected to have the minimum load over the next 10 seconds?"

  Compositional Query Over Dynamic Streams

Dv as a Query



- "Show me the results of rendering the scene synthesized by combining the results of isosurface extraction and morphology reconstruction over regularly grided data resulting from interpolation of this region of the simulation database"

Compositional Query Describing An Application

No Specific Query Plan is Implied

Grid Schedulers

- Similar needs, more flexibility
- But these abstractions are important
  - GridSearcher [Schopf]
    - Compositional Queries over MDS

Supporting Compositional Queries

Set operations -> Relational Algebra -> RDBMS

- ANSI SQL
- Time-bounded Non-deterministic queries

Type Hierarchies

Query Example (RPSDB)

Schemas and Indices

Non-deterministic Time-bounded Queries

Data Stream Support

Distributed Operation

Interaction with other GIS and Grid Performance Systems

Fast Updates and Freshness

Performance Evaluation

Tensions to explore

- RDBMS versus distributed data and decentralized administration and multiple security domains
- RDBMS versus expensive queries
- Power versus usability (SQL)

Prototype System(s)

- RPSDB
- dQUOB

Unification

ACID?

Conclusions

- Come join us