Grid-based Data Stream Processing in e-Science

Richard Kuntschke\textsuperscript{1}, Tobias Scholl\textsuperscript{1}, Sebastian Huber\textsuperscript{1}, Alfons Kemper\textsuperscript{1}, Angelika Reiser\textsuperscript{1}, Hans-Martin Adorf\textsuperscript{2}, Gerard Lemson\textsuperscript{3}, and Wolfgang Voges\textsuperscript{3}

\textsuperscript{1}Lehrstuhl Informatik III: Datenbanksysteme
Fakultät für Informatik
Technische Universität München

\textsuperscript{2}Max-Planck-Institut für Astrophysik

\textsuperscript{3}Max-Planck-Institut für extraterrestrische Physik
Important Challenges in e-Science

■ In general:
  - Large and exponentially growing amounts of data
  - Distributed data archives
  - No unique identifiers
  - Uncertainty

■ In astrophysics:
  Spectral Energy Distributions (SEDs)
  - Used to classify celestial objects (active galactic nuclei, brown dwarfs, neutron stars, ...)
  - Generation requires spatial (astrometric) matching
Spatial (Astrometric) Matching

Current solutions …

- … load all data into main memory
  - Uses a lot of memory
  - Infeasible if memory size is insufficient
- … process all data at once and deliver the complete result at the end
  - Inefficient
  - No results until all processing has completed
Our Contributions

- **StarGlobe**
  - Grid-based P2P Data Stream Management System implemented on top of Globus
  - In-network processing
    - Early filtering
    - Parallelization
    - Pipelining
    - Load-balancing
  - Mobile user-defined operators

- **Astrophysical Example Workflow**
  - Astrometric matching
  - Performance evaluation
The StarGlobe Architecture

- Stream 0
- Stream 1
- Super-Peer
- Backbone
- Publish
- Subscribe
- Query 1
- Query 2
- Fct-Provider
- Transform
- Filter

Grid-based Data Stream Processing in e-Science
Traditional Approach: Bring Data to Code
New Approach: Bring Code to Data

Grid-based Data Stream Processing in e-Science
Mobile User-Defined Operators

- Load user-defined operators from function provider servers in the network
- Common interface for integrating external operators
- Push-based iterator
- Flexibility
StreamIterator Interface

- open(Config, StreamWriter)
  - Configuration parameters
  - Writer for result stream
- next(StreamIteratorEvent)
  - Next element in input stream
  - Writing output to result stream using StreamWriter.write()
- close()
Communication between StreamProcessor and StreamIterator

XML InputStream 1  -->  StreamHandler 1  -->  StreamIterator  -->  StreamWriter  -->  XML OutputStream

XML InputStream 2  -->  StreamHandler 2

...  -->  StreamHandler n

XML InputStream n  -->  StreamProcessor

Item 1  Item 2  Item n  Result Item
Astrophysical Example Workflow

Grid-based Data Stream Processing in e-Science
Distributed Query Evaluation Plan
Distributed Query Evaluation Plan

plan-6
at peer-6

\( \chi^2 \text{filter-0} \)

join-0

plan-0
at peer-0

enrich-0

transform-0

stream-0

plan-1
at peer-1

enrich-1

transform-1

stream-1
Distributed Query Evaluation Plan

plan-10
at peer-10

display

$\chi^2$filter-4

join-4
# Evaluation of Early Filtering

<table>
<thead>
<tr>
<th></th>
<th>WITH early filtering</th>
<th></th>
<th>WITHOUT early filtering</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Stream size</td>
<td># Match candidates</td>
<td>Stream size</td>
<td># Match candidates</td>
</tr>
<tr>
<td>After join-0</td>
<td>808 KB</td>
<td>611</td>
<td>808 KB</td>
<td>611</td>
</tr>
<tr>
<td>After join-1</td>
<td>1,874 KB</td>
<td>1,138</td>
<td>1,874 KB</td>
<td>1,138</td>
</tr>
<tr>
<td>After join-2</td>
<td>1,387 KB</td>
<td>826</td>
<td>1,387 KB</td>
<td>826</td>
</tr>
<tr>
<td>After join-3</td>
<td>6,355 KB</td>
<td>2,522</td>
<td>46,525 KB</td>
<td>15,489</td>
</tr>
<tr>
<td>After join-4</td>
<td>14,356 KB</td>
<td>3,815</td>
<td>1,838,648 KB</td>
<td>364,299</td>
</tr>
<tr>
<td>After filter-4</td>
<td>1,364 KB</td>
<td>318</td>
<td>1,364 KB</td>
<td>318</td>
</tr>
<tr>
<td>Duration h:m:s</td>
<td>00:02:58</td>
<td></td>
<td>02:46:00</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th># MATCH CANDIDATES</th>
<th>FILTER RATIO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Before filtering</td>
<td>After filtering</td>
</tr>
<tr>
<td>At join-0</td>
<td>611</td>
<td>289</td>
</tr>
<tr>
<td>At join-1</td>
<td>1,138</td>
<td>452</td>
</tr>
<tr>
<td>At join-2</td>
<td>826</td>
<td>458</td>
</tr>
<tr>
<td>At join-3</td>
<td>2,522</td>
<td>400</td>
</tr>
<tr>
<td>At join-4</td>
<td>3,815</td>
<td>318</td>
</tr>
</tbody>
</table>
Conclusion

- Synergies between research in computer science and other scientific disciplines, e.g., astrophysics

- StarGlobe
  - Handling large data volumes efficiently
    - Early filtering, parallelization, pipelining
  - Returning first results early on
    - Pipelining
  - Flexible support of domain-specific application logic
    - Mobile user-defined operators

- Results also applicable to other domains