StreamGlobe: P2P Stream Sharing

Data Stream Sharing in a Grid-based P2P Data Stream Management System

Richard Kuntschke
Lehrstuhl Informatik III: Datenbanksysteme
Technische Universität München – Fakultät für Informatik

Network Basics:

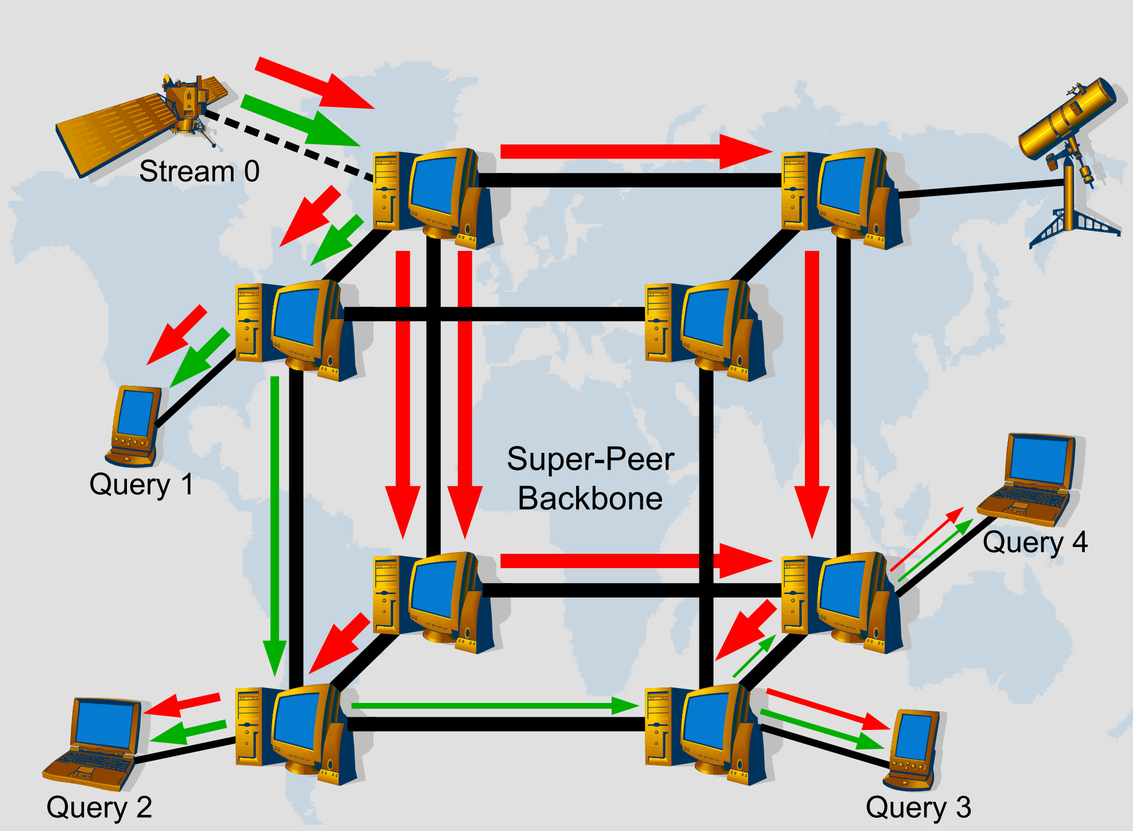
- Grid-based P2P network
- Super-Peer Backbone
- Super-Peers: Powerful stationary servers
- Thin-Peers: Less powerful, possibly mobile peers, sensor devices, etc.

Deficiencies of traditional approach:

- Redundant transmission of data streams
- Redundant execution of stream transforming operators
- Transmission of unnecessary data

⇨ Increased network traffic
⇨ Increased peer load

Publish & Subscribe in a P2P Network



StreamGlobe Basics:

- StreamGlobe: Distributed Data Stream Management System (DSMS)
- Super-Peers process and route data streams
- Thin-Peers publish and subscribe to data streams

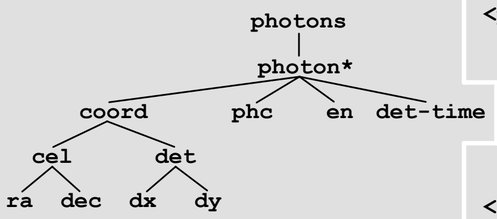
Benefits of StreamGlobe approach:

- Stream sharing avoids redundant stream transmission
- Sharing computational results avoids redundant computation
- Early filtering and aggregation avoid unnecessary data transmission

⇨ Reduced network traffic
⇨ Reduced peer load

WXQuery Subscription Language

Example Stream DTD:



Example Query 1:

```
<photons>
  for $p in stream("photons")/photons/photon
  where $p/phc < 150
  return
    <photon>
      { $p/phc } { $p/en }
    </photon>
</photons>
```

Example Query 2:

```
<photons>
  for $p in stream("photons")/photons/photon
  return
    if ($p/phc > 100 and $p/phc <= 200) then
      <photon>
        { $p/coord } { $p/phc } { $p/en }
      </photon>
    else
      <photon>
        { $p/coord }
      </photon>
</photons>
```

WXQuery Subscription Language:

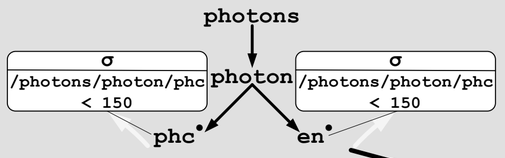
- Operates on XML data streams and persistent XML data
- Based on XQuery
- Supports time-based and item-based data windows
- Query processing with *FluX*: An efficient, buffer-aware query engine for streaming XML data

Example Aggregation Query:

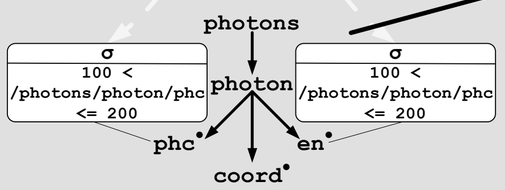
```
<photons>
  for $w in stream("photons")/photons/photon|det_time diff 20 step 10|
  let $a := avg($w/en)
  return
    <avg_en> { $a } </avg_en>
</photons>
```

Representing and Merging Query and Stream Properties

Properties of Query 1:



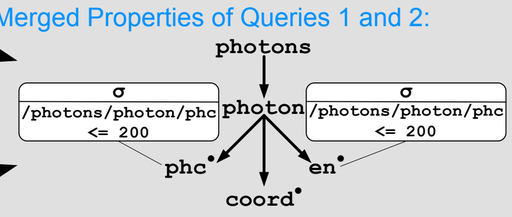
Properties of Query 2:



Properties Data Structure:

- Annotated tree
- Tree reflects *structural* properties
- Annotations reflect *content-based* properties

Merged Properties of Queries 1 and 2:

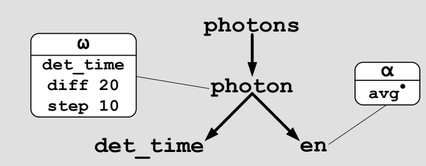


Compare and merge properties

Abstract Properties Approach:

- Treat queries and streams symmetrically
- Represent both internally using the same properties data structure
- Properties reflect relevant aspects for query optimization using *in-network query processing* and *multi-subscription optimization*

Properties of Aggregation Query:



Merging Properties:

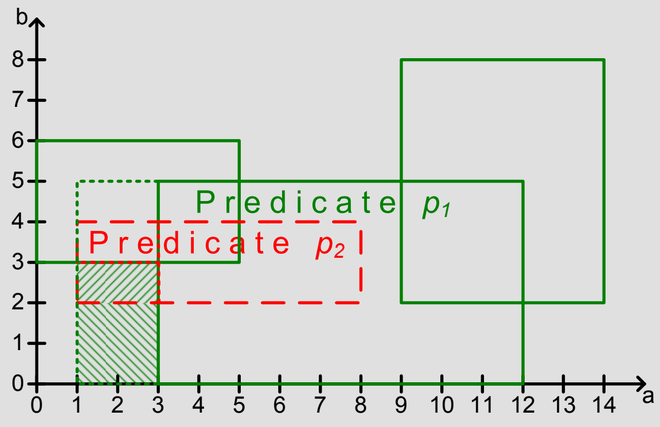
- Necessary for generating data streams that can be used to satisfy multiple queries
- Requires predicate matching strategy

Predicate Matching and Evaluation

Example Predicates:

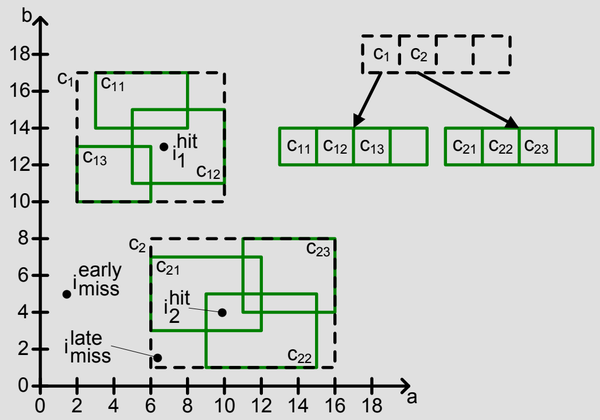
Stream Predicate *p1*:
$(a \geq 3) \wedge (a \leq 12) \wedge (b \geq 0) \wedge (b \leq 5) \vee$
$(a \geq 9) \wedge (a \leq 14) \wedge (b \geq 2) \wedge (b \leq 8) \vee$
$(a \geq 0) \wedge (a \leq 5) \wedge (b \geq 3) \wedge (b \leq 6)$

Query Predicate *p2*:
$(a \geq 1) \wedge (a \leq 8) \wedge (b \geq 2) \wedge (b \leq 4)$



Predicate Matching Algorithms:

- Quick Check (QC)
- Heuristics with Simple Relaxation (HSR)
- Heuristics with Complex Relaxation (HCR)
- Exact Matching (EM)



Predicate Evaluation Algorithms:

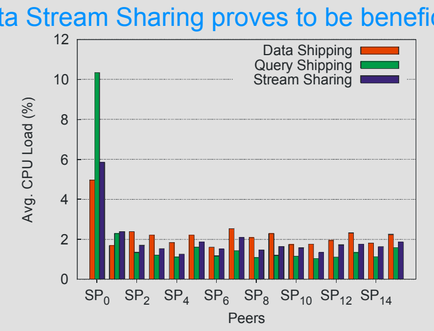
- Standard Evaluation (SE)
- Index-based Evaluation (IE)

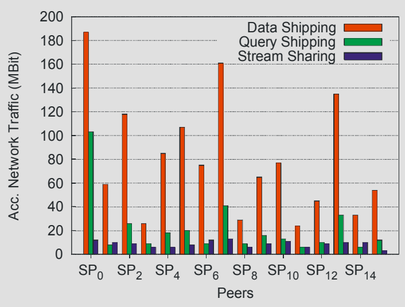
Optimization:

- Multi-dimensional index support (I) for predicate matching and evaluation
- Improve performance of evaluation index through short-circuiting (SC)

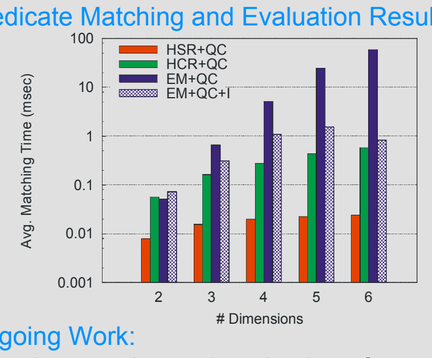
Results / Ongoing and Future Work

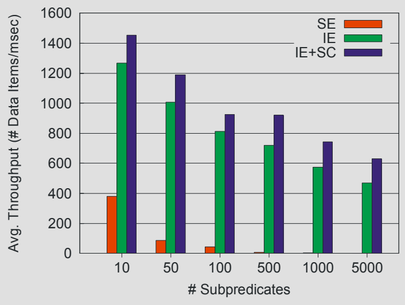
Data Stream Sharing proves to be beneficial:





Predicate Matching and Evaluation Results:





Ongoing Work:

- Implementation and evaluation of an advanced Data Stream Sharing technique including Data Stream Widening/Narrowing through merging properties structures

Future Work:

- Hierarchical or fully distributed network architecture to address scalability
- Dynamic optimization
- Specialized multi-dimensional index structure for predicate matching and evaluation

http://www-db.in.tum.de/research/projects/StreamGlobe