Seminar
Event Stream Processing (ESP)
Summer Term 2016
Matthias Weidlich
Contact Info

Research group: **Process-driven Architectures**

- Process-oriented systems
- Event-processing systems

http://hu.berlin/pda

Contact

- matthias.weidlich@hu
- Office 4.101
Goals

1) Enhance knowledge
   • Event stream processing terminology
   • Models, languages, and systems
   • Approaches to achieve robustness
   • Aggregations of events
   • Join processing
   • Optimisations

2) Practice to give talks

3) Improve your scientific reviewing skills
Requirements

Understand

• Read scientific research paper
• Identify the (main) problem discussed in the paper
• Show why the solved problem is important
• Develop your own examples!
• Read related work if required
• Link your own paper to other papers of the seminar

Explain & Discuss

• Present the paper to the seminar attendees
• Discuss your paper with the seminar attendees
• Write a review of the paper, also incorporating the discussion
Scenario: Logistics

Real-time planning in logistics aims at

- Reduced slack time
- Reduced risk of missed connections
- Efficient vehicle utilisation

Based on

- Positions of vehicles
- Recent processing times
- Current workloads
Scenario: Cluster Monitoring

Real-time cluster monitoring aims at

- Efficient job execution
- Reduced number of evicted jobs
- Identification of stragglers

Based on

- Resource availability
- Machine utilisation
- Job scheduling
Events in Everyday Life
Detection of Complex Events

Observation:
- Most events are not interesting
- New events supersede old events
- Ability to react to changing situations provides value

Derive complex events from simple events
Traditional Databases

Database Management System (DBMS):
Data relatively static but queries dynamic

Persistent relations
- Random access
- Low update rate
- Unbounded disk storage

One-time queries
- Finite query result
- Queries exploit (static) indices
Event Recognition System:
Queries static but data dynamic - input is time-dependant stream

Transient streams
- Sequential access
- Potentially high rate
- Bounded main memory

Continuous queries
- Produce time-dependant result stream
- Indexing?
Event Recognition: Performance Matters!

Value of analytics decreases over time

Decision making benefits from timeliness of analytics
- Limited windows of opportunities (now or never)
- Competitive advantage (quicker than the rest)

Compliance and performance assessment
- Early detection of deviations
- Early start of remedy actions
Events

What is an event?

An event is a happening of interest. An event type is a specification of a set of events of the same structure and semantics. [Etzion and Niblett (2011)]

Cluster monitoring use case:

- Events denote transitions in job/task lifecycle
- Events indicate availability of machines
Event Types

How to model events?

Event schema defined as set of attributes

- Payload of event is a set of key-value pairs
- Events often have associated time stamp
- E.g. arrival time, time of reading, ...

Cluster monitoring:

```
Task events table
The task events table contains the following fields:
1. timestamp
2. missing info
3. job ID
4. task index - within the job
5. machine ID
6. event type
7. user name
8. scheduling class
9. priority
10. resource request for CPU cores
11. resource request for RAM
12. resource request for local disk space
13. different-machine constraint
```

Schedule₁ (1444026993, -1, 239, 3, B-2, Schedule, rmalik, ...)
Streams

What is a stream?

A stream is a real-time, continuous, ordered (implicitly by arrival time or explicitly by timestamp) sequence of items. It is impossible to control the order in which items arrive, nor is it feasible to locally store a stream in its entirety. [Golab & Oszu (SIGMOD 2003)]

Data stream processing view: items are data tuples

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$t_2$</td>
<td>$t_3$</td>
<td>$t_4$</td>
<td>...</td>
</tr>
<tr>
<td>time</td>
<td>miss</td>
<td>job ID</td>
<td>time</td>
<td>miss</td>
</tr>
</tbody>
</table>

Complex event processing view: items are typed events

Time
SELECT timestamp, job, avg(cpu) AS avgCpu
FROM clusterEvents [range 60 slide 1]
WHERE eventType == 1
GROUP BY job
Event Pattern Languages

Zoo of pattern specification languages
- Common core concepts
- Different syntax
- Subtle differences in semantics

Pattern definition

```plaintext
Pattern SEQ(Schedule a, Schedule+ b[], Evict c)
Where skip-till-any-match
And b[].machine = a.machine
And a.job = c.job And a.task = c.task
Within 2 days
Return a.(job, task), b[].job
```
Seminar Scope

Event stream processing terminology

Models, languages, and systems
  - Pattern semantics
  - Matching algorithms

Approaches to achieve robustness
  - Fault-tolerance
  - Out-of-order event

Aggregations of events
  - Resource sharing
  - Incremental computation

Join processing
  - Join semantics
  - Heterogeneous architectures

Optimisations
  - Load shedding
  - Pattern rewriting
Organisation

When: Thursdays, 15:00 – 17:00
Where: Room 1.303
How: Course language is English

Requirements for earning LP:
- Give talk (30 min + 15 min discussion)
- Write scientific review
- Attend and participate in the seminar
Guidelines

Slides must be ready one week before(!) your talk

- Send slides via email to me (pdf only)

Write review after your talk (even better: after last talk)

- Send review via email to me (pdf only)
- Use Latex (document class article)
- Around 2-3 pages
- Deadline for the review: August 5, 2016
Hints on Writing a Review

1) Summary of paper
   - This needs to be only 1-3 sentences, but it demonstrates that you understand the paper and, moreover, can summarize it more concisely than the authors in their abstract.

2) Good things about the paper
   - This introduction is good psychology if you want the authors to revise their paper.

3) Major comments
   - Discuss the author’s assumptions, technical approach, analysis, results, conclusions, reference, etc. Be constructive, if possible, by suggesting improvements.
   - Focus on 2-3 key issues raised in the article. Make clear the authors’ own argument before you criticise and evaluate it. Support your criticisms with evidence from the text or from other writings.

4) Minor comments
   - This section contains comments on minor aspects, such as style, figures, grammar, graphs, etc., but also parts that need minor clarification or missing links to related literature. If any of these are especially poor, then they might escalate to the 'major comments' section.
   - It is acceptable to write these comments in list (or bullet) form.

5) Final Evaluation
   - A final evaluation of the overall contribution that the article has made to your understanding of the topic (and maybe its importance to the development of knowledge in this particular area or discipline, setting it in the context of other writings in the field).
Research Papers I

Terminology

Models, languages, and systems


Research Papers III

Robustness


Research Papers IV


Aggregation


Research Papers V


Joins


Optimisations


22) Nicholas Poul Schultz-Møller, Matteo Migliavacca, Peter R. Pietzuch: Distributed complex event processing with query rewriting. DEBS 2009

Summary

By Monday April 25, 2016: Submit by mail (matthias.weidlich@hu) your ranked selection of 3 topics

Assignment of topics/dates will happen soon after that

Announcements will be sent out via Goya

No meeting next week, April 28, 2016

Thanks!
Questions?