Monitoring Streams

A New Class of Data management Applications

Don Carney Uğur Çetintemel Mitch Cherniack
Christian Convey Sangdon Lee Greg Seidman
Michael Stonebraker Nesime Tatbul Stan Zdonik

Original slides by: Kenji Okuma
Presentation: Jiemin Zhang
Discussion: Mohammed Alam

RAP1

Outline

- Introduction
- Aurora System Model
- Aurora Optimization
- Run-Time Operation
- Conclusion

Traditional DBMSs

5 Assumptions

- Human-Active, DBMS-Passive (HADP) Model
- Current state is important: Previous data needs to be extracted from the log
- Triggers and alerters are second-class citizens
- Perfect synchronization of data elements and exact query answers
- Require no real-time services

Monitoring Applications

- Applications that monitor continuous streams of data
- Where all five assumptions are wrong
- Ex: *military apps*
  financial analysis apps
  tracking apps

Monitoring Applications

- DBMS-Active, Human-Passive (DAHP) model
- History of data is important: Not only the current state but also the previous history
- Triggers and alerters are first-class citizens RAP3
- Missing or imprecise data, and approximate query answers
- Require real-time services

Monitoring Application Example: Car Navigation System RAP2

- Data (e.g., the location of the car) comes from external sources
- History of the data is required (e.g., display a trajectory of your car in the past 20 minutes)
- Trigger and alerters oriented: an alert for the driver when the car is approaching to an intersection
- The location of the car is not always perfectly transmitted due to interferences etc..
- Real-time services (e.g., the current location of the car)

Slide 2

RAP1 I cut the second slide (which was the same slide as this one, only without the introduction highlighted), to save time
Rachel Pottinger, 07/11/2006

Slide 5

RAP3 I removed two "the"s here. The first one was just for space. The second one was a quibble (and don't worry, I don't think it was a quibble with what you had - I think it was in the original slides and I didn't catch it the first time)
Rachel Pottinger, 07/11/2006

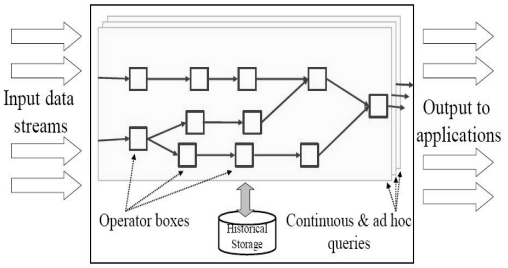
Slide 6

RAP2 I changed the title on this slide to make it clearer that it was an example
Rachel Pottinger, 07/11/2006

Outline

- Introduction
- Aurora System Model
- Aurora Optimization
- Run-Time Operation
- Conclusion

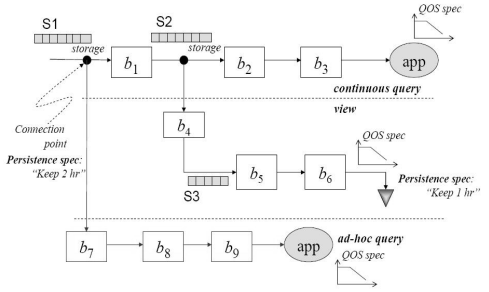
Aurora System Model



Operators

- Windowed operators: Operate on sets of consecutive tuples from a stream
  - *Slide*
  - *Tumble*
  - *Latch*
  - *Resample*
- Operators for single tuple
  - *Filter*
  - *Map*
  - *GroupBy*
  - *Join*

Aurora Query Model



Discussion #1

- Why did we have to wait so long for monitoring applications? Can it be related to other concepts (from or outside db)?
- Given that extensions can only happen at checkpoints, do you think it is enough?

Outline

- Introduction
- Aurora System Model
- Aurora Optimization
- Run-Time Operation
- Conclusion

Reasons

- Traditional optimization

  To minimize the number of iterations over large data set
- Aurora network
  - *A large number of boxes*
  - *High data rates*
  - *Lots of changes over time*

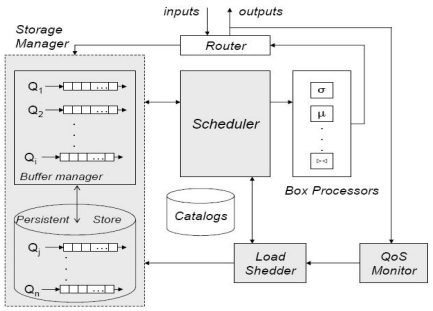
Aurora Optimization

- Dynamic Continuous Query Optimization
  - *Insert projections*
  - *Combining Boxes*
  - *Reordering Boxes*
- Ad-Hoc Query Optimization

Outline

- Introduction
- Aurora System Model
- Aurora Optimization
- Run-Time Operation
- Conclusion

Aurora Run-Time Architecture



QoS Data Structure

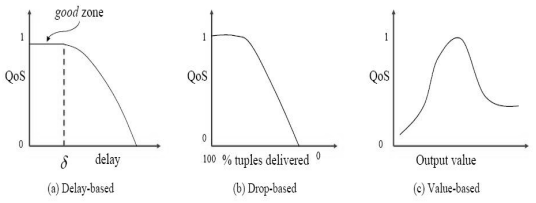
- Quality of Service, a multidimensional function of several attributes

  Response times

  Tuple drops

  Values produced

QoS Data Structure



Real-Time Scheduling

- Train Scheduling

  Two basic non-linearities: Inter-box & Intra-box
  - *Have boxes queue as many tuples as possible without processing*
  - *Process complete trains at once*
  - *Pass them to subsequent boxes without having to go to disk*
- Priority Assignment

  State-based approach

  Feedback-based approach

Outline

- Introduction
- Aurora System Model
- Aurora Optimization
- Run-Time Operation
- Conclusion

Conclusion

- Monitoring applications are those where streams of information, triggers, real-time requirement, and imprecise data are prevalent
- Aurora is a DAHP system oriented towards monitoring applications
- Future directions:

  Aurora for distributed processing*

  More efficient data handling algorithm

Discussion #2

- QoS graph
  - Do they help?
  - what can go wrong with user defined "good zones"
- Train scheduling
  - Will it be effective?
- Dynamic optimization of sub-networks
  - Potential problems?

Thank You!