

Compare Query optimizer on different CEP products

Questions

NFA based/query plan tree based?

Support for multi query optimization?

Statistics/how to perform cost estimation?

Literature

SASE

Related papers

1. High-Performance Complex Event Processing over Streams, SIGMOD 2006
2. SASE: Complex Event Processing over Streams, CIDR 2007
3. Efficient Pattern Matching over Event Streams, SIGMOD 2008
4. Optimizing Expensive Queries in Complex Event Processing, SIGMOD 2014

Answers

1. NFA based. The optimizer is built to optimize one query at one time. Methods include:
 1. Sharing intermediate results between *Runs*(noted, this is a special concept in CEP, and it means a *potential matched pattern sequence*).
 2. partition active stack.
 3. postpone filtering for **Kleene closure pattern queries**.
2. No.
3. It does not provide a cost model. Heuristic only.

Remark: The contribution from SASE is that they are the first to proposed CEP product, (Before that, the closest concept is ESP) . And they are also the first proposed NFA based model on execution the CEP. They are also the best in optimizing single query in CEP.

Cayuga

Related papers

1. Rule-Based Multi-Query Optimization, EDBT 2009
2. Towards Expressive Publish/Subscribe Systems, EDBT 2006
3. Cayuga: A General Purpose Event Monitoring System, CIDR 2007

Answers

1. NFA based. The original engine is built from publish/subscribe system, which has an emphasize on multi-query optimization.
2. Yes, it support MQO by using three techniques:state merging, automaton indexing, channel, please find my detail study report for this.
3. It does not provide a cost model. Heuristic only. (Mentioned in the original paper[1], leave as future work)

Remark: The main contribution lays on they are the first who consider extending publish-subscribe system into CEP, the good thing for publish-subscribe system is it naturally support multi-query optimization, while CEP doesn't.

ZStream

Related papers

ZStream: A Cost-based Query Processor for Adaptively Detecting Composite Events, SIGMOD 2009

Answers

1. query plan tree based, it will optimize every single query based on its cost model.
2. No.
3. It provided a cost model, described in Section 5 of original paper.

In traditional databases, the estimated cost of a query plan consists of I/O and CPU costs. In ZStream, **I/O cost is not considered** because all primitive events are memory resident. ZStream computes CPU cost for each operator from three terms:

1. the cost to access the input data,
2. the cost of predicate evaluation
3. and the cost to generate the output data.

These costs are measured as the number of input events accessed and the number of output events combined.

The cost $C = C_i + (nk)C_i + pCo$ Please find detail in the original paper, quite easy to understand.

Remark: Obviously, they are the first to propose cost model into CEP.

Hp-Chaos(WPI)

I put HP-Chaos in literature section because I didn't find any HP-Chaos product in marketing, the top CEP products in market are: **IBM System S, SAP ESP, Software AG Amada**. According to Forrest marketing report 2014.

Related papers

1. E-Cube: Multi-Dimensional Event Sequence Analysis Using Hierarchical Pattern Query Sharing, SIGMOD 2011
2. Complex Event Analytics: Online Aggregation of Stream Sequence Patterns, SIGMOD 2014

Answers

1. Not available ? suspected to be NFA based. Since he mentioned "Stack based CEP". The optimizer will compute and generate the best overall execution plan for a set of queries based on its cost model.
2. Yes, actually E-cube is the state-of-the-art in the topic of "MQO on CEP".
3. Yes, It provides cost-model for each MQO optimization strategies.

Table 1: Terminology Used in Cost Estimation

Term	Definition
$C_{compute(qi qj)}$	The evaluation cost for query q_i basing on evaluation results for q_j
$C_{compute(qi)}$	The cost of computing results for a query q_i independently
$ S_i $	Number of tuples of type E_i that are in time window TW_P . This can be estimated as $Rate_E * TW_P * P_E$
TW_P	Time window specified in a pattern query P
$Rate_E$	Rate of primitive events for the event type E
P_E	Selectivity of all single-class predicates for event class E. This is the product of selectivity of each single-class predicate of E.
$Pt_{Ei,Ej}$	Selectivity of the implicit time predicate of sub-sequence (E_i, E_j) . The default value is set to 1/2.
$P_{Ei,Ej}$	Selectivity of multi-class predicates between event class E_i and E_j . If E1 and E2 do not have predicates, it is set to 1.
$ R_E $	Number of results for the composite event E
C_{type}	The unit cost to check type of one event instance
$q_i.length$	The number of event types in a query q_i
$NumE$	Number of total events received so far
$NumRE$	Number of relevant events received of the types in query set Q
C_{access}	The cost of accessing one event
C_{app}	The unit cost of appending one event to a stack and setting up pointers for the event
C_{ct}	The unit cost to compare timestamp of one event instance with another one

the following is the cost model for **General to Specific with Pattern Changes**:

$$\begin{aligned}
C_{compute(qj|qi).gp} = & |S_m| * |S_n| * Pt_{E_m,E_n} * P_{E_m,E_n} \\
& + |R_{SEQ(E_m,E_n)}| \log |R_{SEQ(E_m,E_n)}| \\
& + |R_{qi}| * |R_{SEQ(E_m,E_n)}| * Pt_{E_k,E_m} \\
& * P_{E_k,E_m} + |R_{SEQ(E_m,E_n)}| + |R_{qi}|
\end{aligned} \quad (4)$$

Similarly, GtoS with concept changes and StoG etc are also provided.

Remark: *Refinement* used in the paper is refer to simultaneously concept and pattern changes, then select minimum cost among two. The contribution is obviously they are the first to really study MQO on CEP, Cayuga also considered MQO on CEP, but they simply ignored the fact that CEP need to consider multiple different sequence queries (not only different filtering conditions).

Industry

Esper

I put Esper here, because it's the top open-source CEP product, many commercial CEP product are built from it's core engine.

Related papers

1. A Middleware Platform to Federate Complex Event Processing, IEEE International EDOC Conference, 2012

Answers

1. It's a mix style, look at these two claims from it's official document:
 1. The EPL pattern engine does not employ NFA, it is based on dynamic state trees where branches (active pattern sub-expressions) create and destroy.
 2. The match-recognize pattern matching functionality is built using nondeterministic finite automata (NFA).
2. Esper uses indexes, a data structure that improves the speed of data retrieval operations. For sorted access it may prefer a binary tree index while a hash-based index is great for key lookups. For efficient matching of incoming events to statements the engine uses inverted indexes.

Index is broadly used in publish-subscribe system, Esper support both ESP and CEP, thus no wonder to find index support. Interestingly, Esper only publish benchmark for ESP style queries, CEP style query benchmark is not provided yet.

3. No cost model mentioned.

SAP ESP

Related papers

No.

Answers

1. It's a mix style, since it support both ESP and CEP.
2. Very limited support, the optimizer will generate the best execution strategy(CCX) for a given CCL project(may include multiple queries), automatic partition and parallelism is the focus of this optimizer. But it does have mentioned "Sharing intermediate results among queries", for instance, if two queries defined same window, then they can share this window.
3. No cost model mentioned. suspected to be heuristic based.

IBM System S

Related papers

1. Partition and Compose:Parallel Complex Event Proecessing, DEBS 2012
2. IBM Streams Processing Language: Analyzing Big Data in motion,

Answers

1. It's automata style.
2. Possible not at all, at least not ever mentioned. The optimizer focus on program efficiency, besides, they also put main efforts on partition and parallelism.
3. No cost model mentioned.

Summarize

From the above study, we can see that, only limited projects in literature mentioned/tried Multi-query optimization on CEP, in industry prospective, no existing product support MQO on CEP, instead, their optimizer focuses on partition and parallelism.

The query optimizer for SAP ESP and IBM System S is the same level, and they represent the top CEP product in industry, but they both far behind the state-of-the-art in literature in the topic of Multi-query optimization.

Name	Industry/Literature	Execution style	Support for Multi-query optimization	Cost model?
SASE	Literature	Automata	Not at all	No
Cayuga	Literature	Automata	Limited support(automaton indexing, channel etc)	No
		Query plan		Yes, proposed

ZStream	Literature	query plan tree	Not at all	the first cost model in CEP	
HP-Chaos	Literature	?	Limited support(pattern sharing), state-of-the-art	Yes, very solid	
Esper	Industry	Mixed sytle	Not at all, require programmer to manualy handle optimization	No	
SAP ESP	Industry	Mixed style	Limited support(CCX optimizer, focuses on partition and parallism, but does have optimization for sharing intermediate results within one project)	?	
IBM S	Industry	Mixed style	Limited support ? (SPL optimizer, optimize user program via types and operator models)	?	