MANCHESTER 1824

The University of Manchester

Time-completeness trade-offs in record linkage using Adaptive query Processing

Paolo Missier, Alvaro A. A. Fernandes

School of Computer Science, University of Manchester

Roald Lengu, Giovanna Guerrini

DISI, Universita' di Genova, Italy

Marco Mesiti

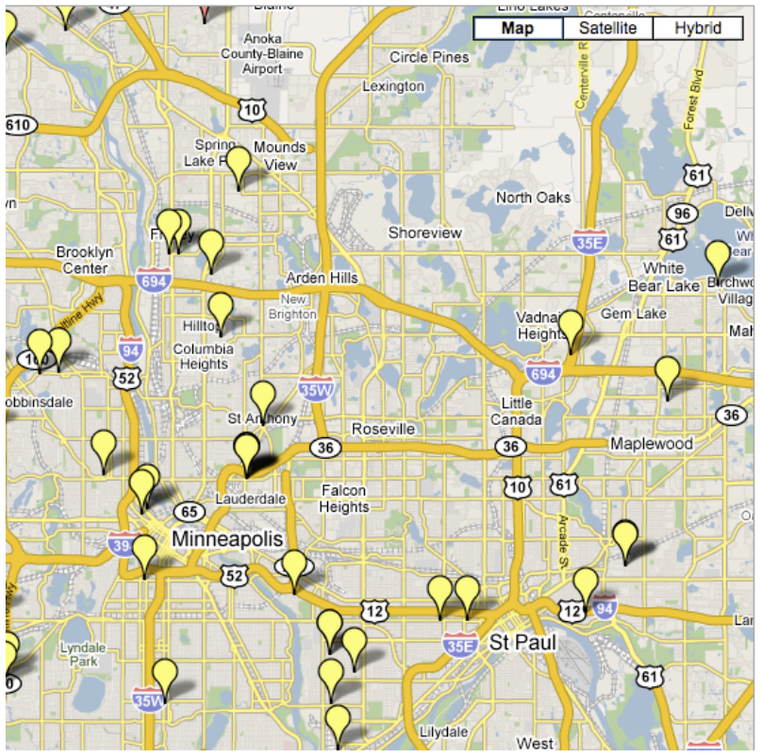
DiCo, Universita' di Milano, Italy

EDBT 2009, St. Petersburg, Russia
March 26th, 2009

Context

- "Situational Applications"
  - designed to address a transient need and short-lasting
  - constructed "on the fly"
- Data mashups and personal dataspaces

  - sources 1..n: collection of car insurance DBs
  - source n+1: reference street atlas
  - target app: mapping accidents hotspots

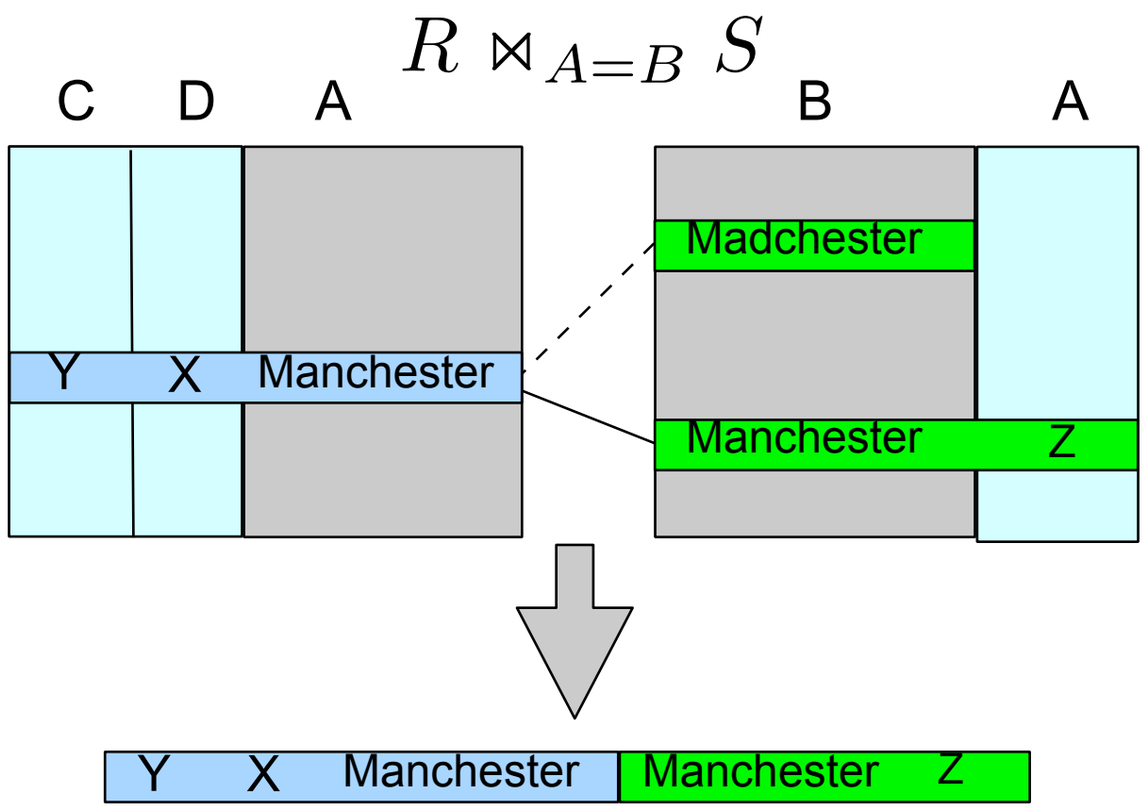


pics	price	description	city	d
	$160K	ChaskaTownhome with Unique floor plan, Great Price! -	Chaska	8
	$160K	11637 Kennelly Circle-Motivated to sell! -	Burnsville	8
	$236K	beautiful house available in Blaine -	Minneapolis	8
	$260K	Walk To Work! Take The Train! Save On Gas! Live In The City Now! -	Minneapolis	8
	$268K	YouLl Love This Home In Champlin! -	Champlin	8
	$190K	Immaculate End Unit Townhome -	Minneapolis	8
	$210K	Stunning Renovation Near Minnehaha Creek -	Minneapolis	8
	$290K	Open House this Weekend: Saturday 8/9/08 12noon – 3pm -	Anoka	8
	$171K	Studio Loft 5% Sale – Save over 8K -	Minneapolis	8
	$270K	4 Bed/3 Bath Home On Cul-De-Sac! Open House Sat 8/10 1-4pm! -	Shakopee	8
	$200K	Featured Unit – One of the last units with a 360 Sq. Ft. Patio -	Minneapolis	8
	$265K	360 sq ft deck, Last South Facing unit – Save %5 on any unit -	Minneapolis	8
	$256K	Huge Sale 5% off any unit at Cw Lofts -	Minneapolis	8
	$155K	Perfect home to raise a family in. On cul-de-sak w/fenced in back yard -	Isanti	8
	$280K	Loft For Sale.Don't miss this opportunity! -	Minneapolis	8
	$200K	Updated 4 bed/2 Bath Huge Yard, Great Neighbors -	Minneapolis	8
	$155K	Charming One Story in Great Area! -	Minneapolis	8
	$169K	Maple Grove Townhome – Demand Location -	Osseo	8
	$175K	Rum River Up North-like Home Opportunity -	Isanti	8

  – no prior knowledge of data sets (streams) to be joined
  – reasonable to join on some of the attributes
  – but, no guarantee of matching values

On-the-fly integration of relational data

- slight mismatches in records lead to incomplete integration
  - due to different encodings, conventions
  - due to errors in data



A case of record linkage

Offline vs online linkage

- Offline record linkage:
  - performed once before queries involving joins
  - 1. reconcile R and S on joining attributes R.A, S.B using your favourite record linkage technique

$$\langle R,S\rangle \rightarrow \langle R',S'\rangle$$

  - 2. perform regular equijoin on the transformed tables:

$$R' \bowtie S'$$

➡ ok for tables that can be analysed ahead of the join

➡ ok when multiple queries issued on integrated tables

Offline vs online linkage

- Offline record linkage:
  - performed once before queries involving joins
  - 1. reconcile R and S on joining attributes R.A, S.B using your favourite record linkage technique

$$\langle R,S\rangle \rightarrow \langle R',S'\rangle$$

  - 2. perform regular equijoin on the transformed tables:

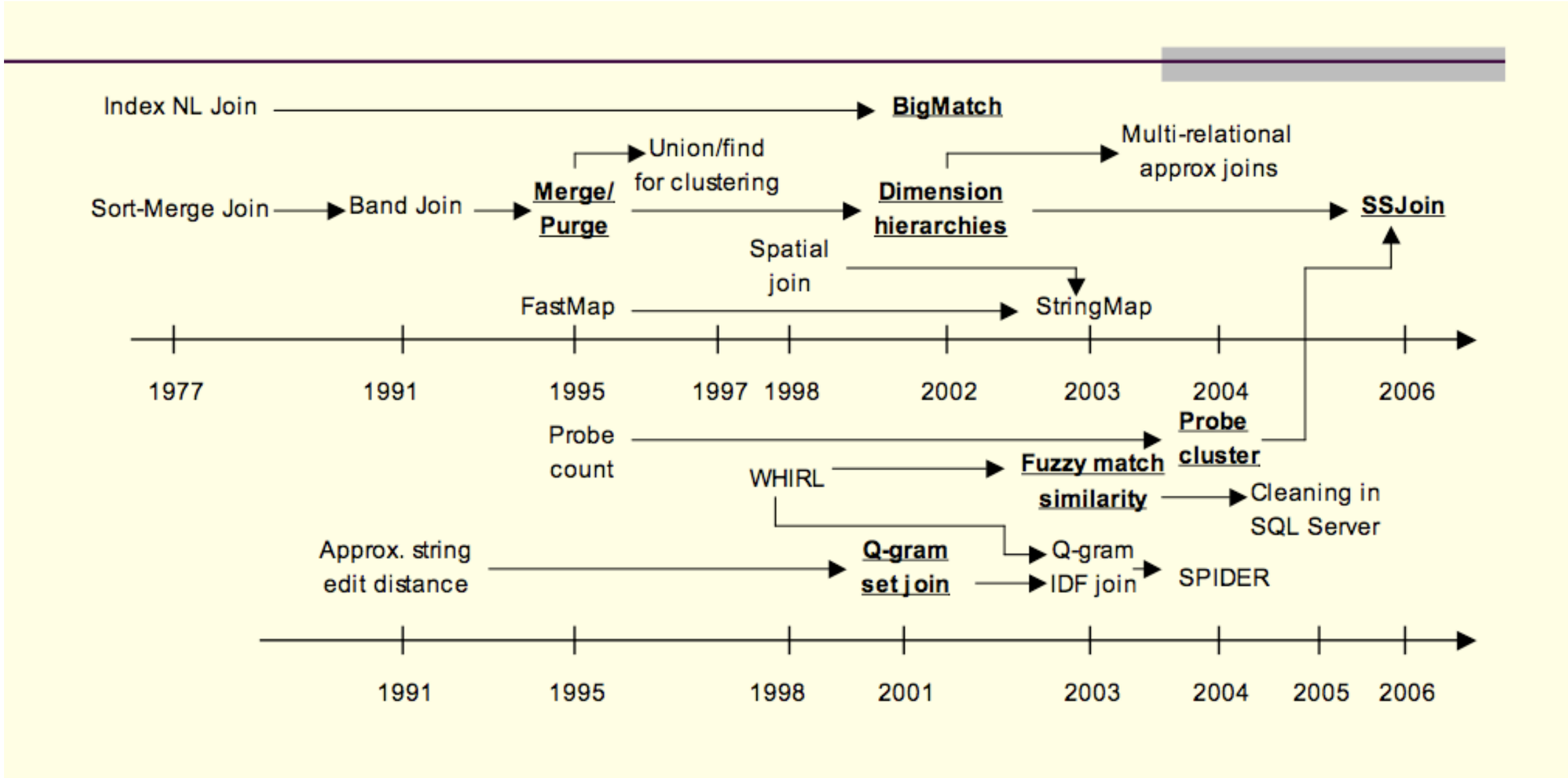
$$R' \bowtie S'$$

    - ➡ ok for tables that can be analysed ahead of the join
    - ➡ ok when multiple queries issued on integrated tables

- Online linkage:
  - performed *while answering* a query
  - exact join ⇒ similarity (or approximate) join

Record linkage and similarity joins

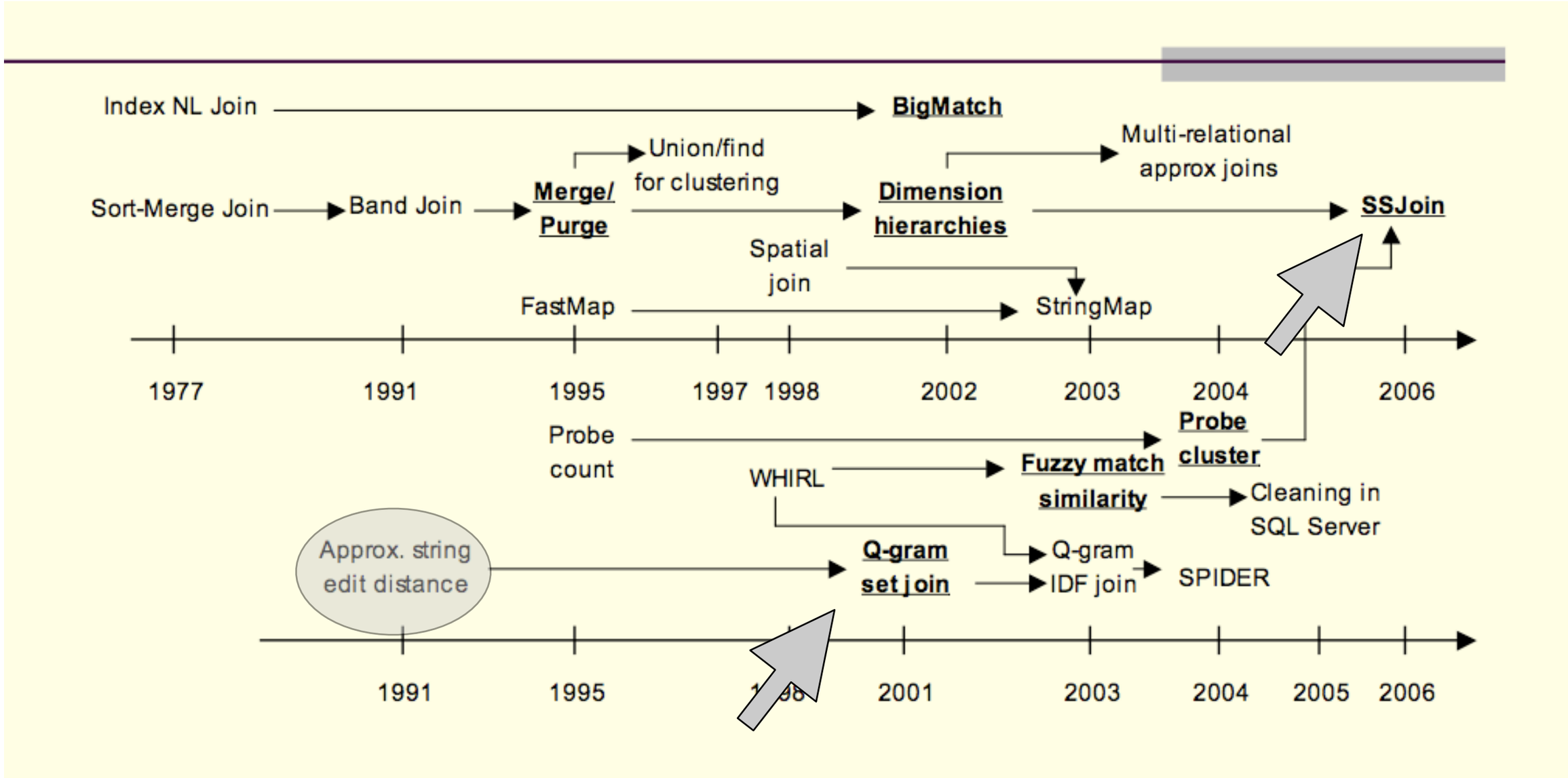
Historical timeline:



from:
N. Koudas, S. Sarawagi, and D. Srivastava. Record linkage: similarity measures and algorithms. Tutorial in SIGMOD '06.

Record linkage and similarity joins

Historical timeline:



from:
N. Koudas, S. Sarawagi, and D. Srivastava. Record linkage: similarity measures and algorithms. Tutorial in SIGMOD '06.

Measuring string similarity using q-grams

- *q-grams* map string *s* to a set *q(s)* of substrings of length *q*:

  Ex.: 3-grams:

  q("Manchester") =
  {'Man', 'anc', 'nch', 'che', 'hes', 'est ', 'ste', 'ter'}.

  q("Ma**d**chester") =
  {'Mad', 'adc', 'dch', 'che', 'hes', 'est ', 'ste', 'ter'}.

$$sim(s_1, s_2) = \frac{|q(s_1) \cap q(s_2)|}{|q(s_1) \cup q(s_2)|}$$ (Jaccard coefficient)

sim("Manchester", "Madchester") = 5/8

$$sim(r_1, r_2) < \theta_1 \rightarrow \text{ not match}$$

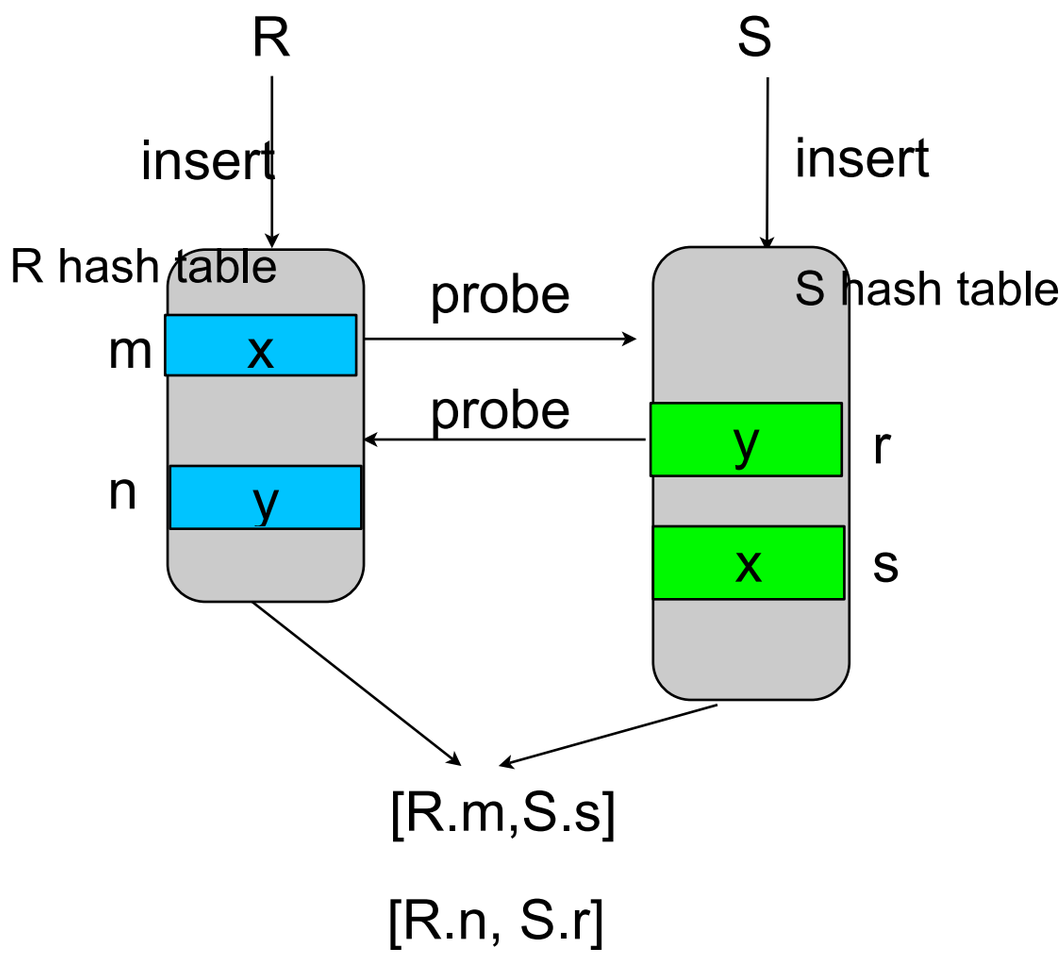
$$\theta_2 < sim(r_1, r_2) \rightarrow \text{ match}$$

Similarity symmetric hash join

- primitive join operator [CGK06]

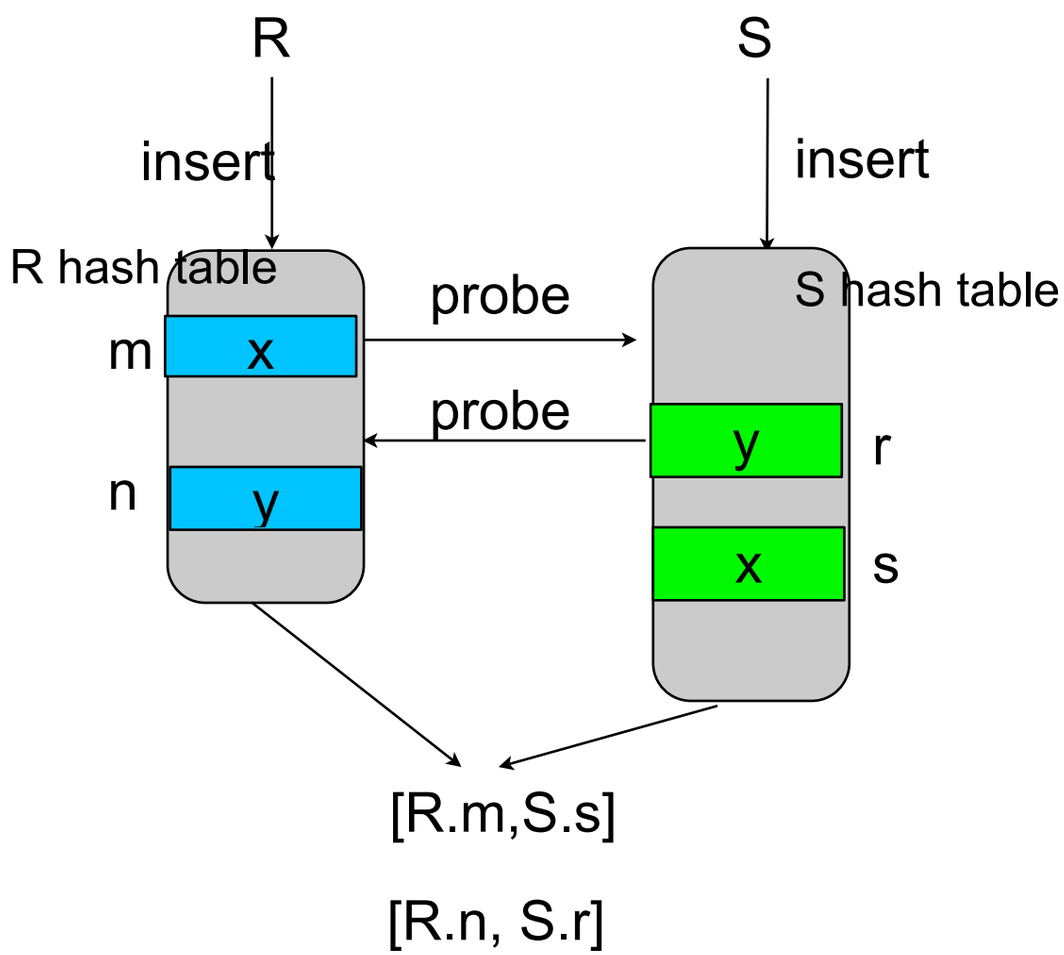
Efficient relational representation:
[CGK06] S. Chaudhuri, V. Ganti and R. Kaushik,
"A primitive operator for similarity joins in data cleaning" (ICDE'06)

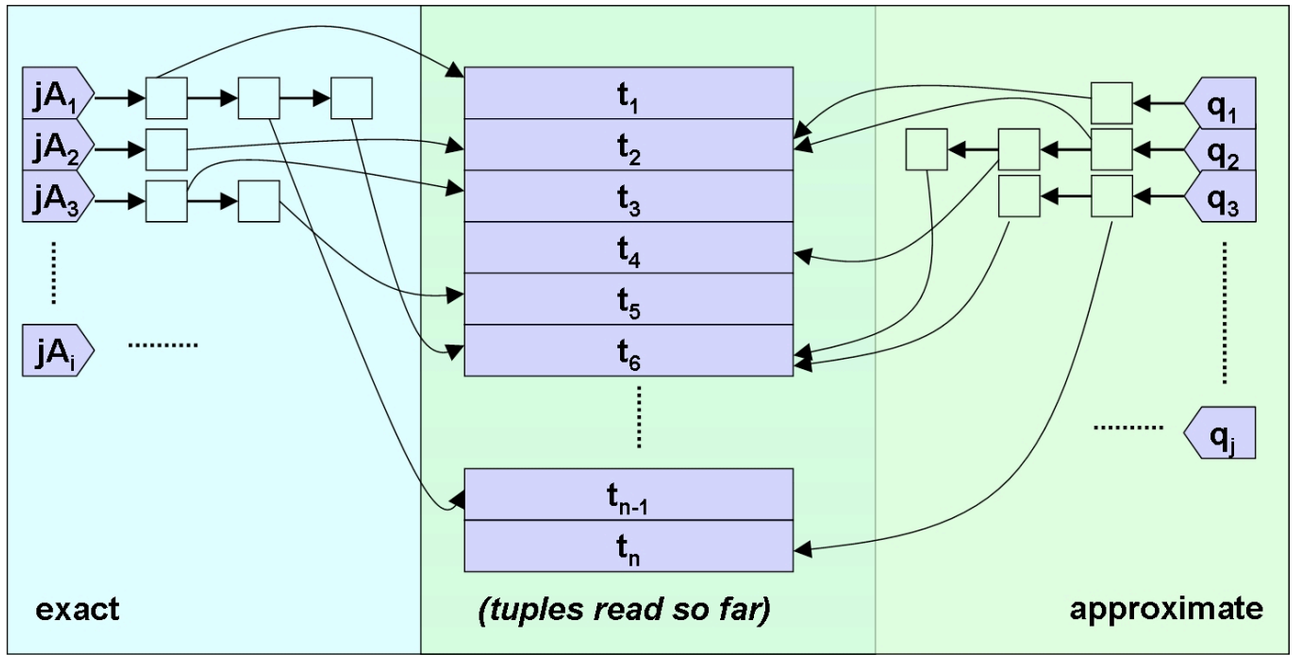
Similarity symmetric hash join



Efficient relational representation:
[CGK06] S. Chaudhuri, V. Ganti and R. Kaushik,
"A primitive operator for similarity joins in data cleaning" (ICDE'06)

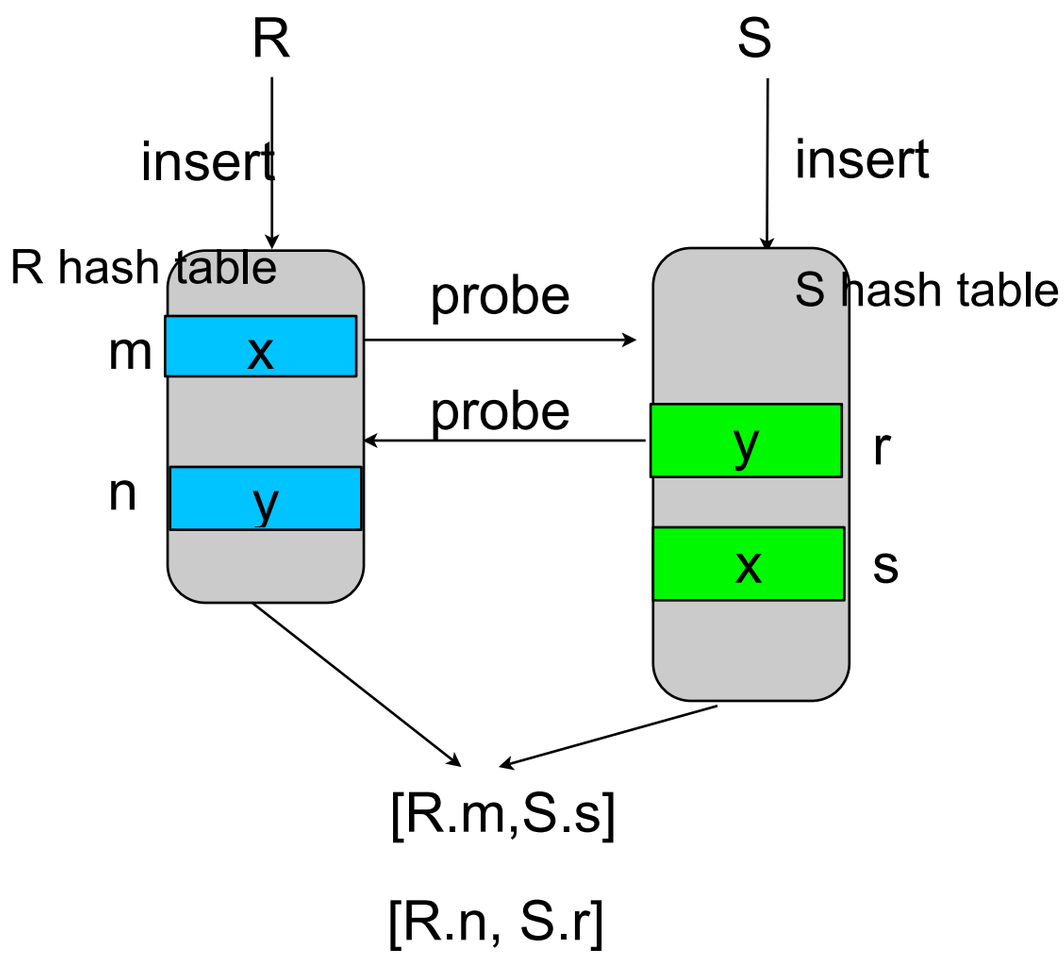
Similarity symmetric hash join

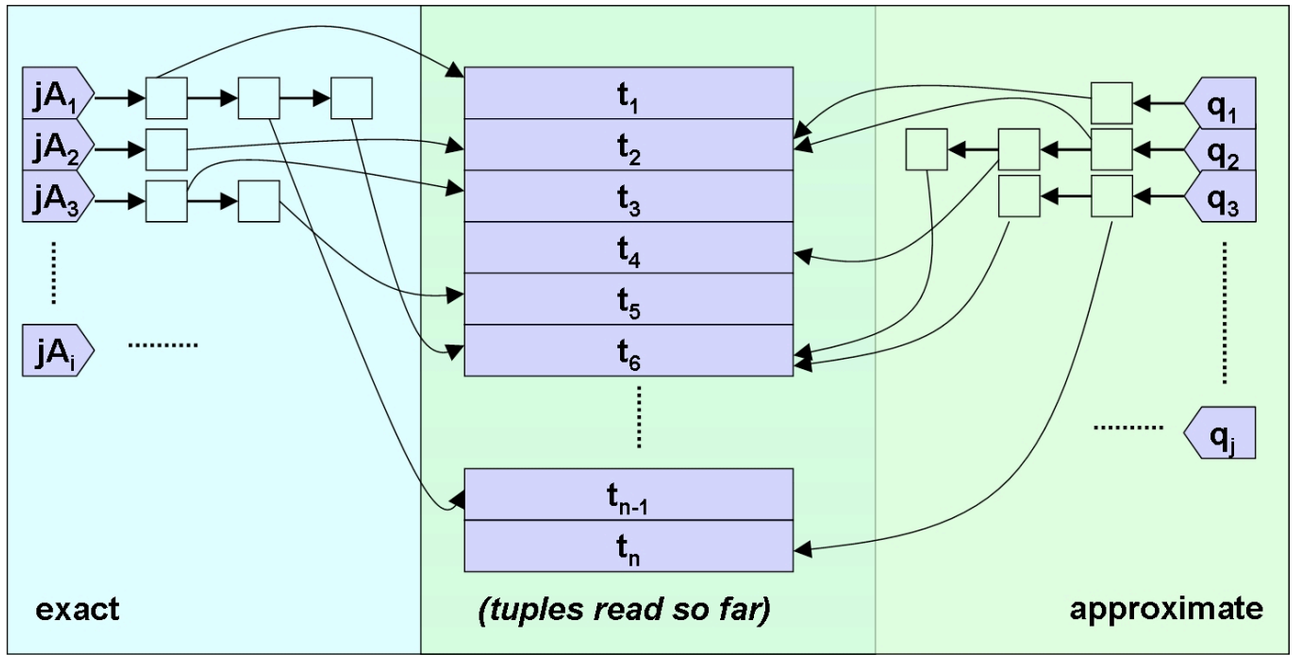




Efficient relational representation:
[CGK06] S. Chaudhuri, V. Ganti and R. Kaushik,
"A primitive operator for similarity joins in data cleaning" (ICDE'06)

Similarity symmetric hash join





- pipelined: suitable for stream processing
- Main sources of complexity:
  - overhead for storing and indexing q-grams
  - cost of computing set intersection

Is full similarity join always necessary?

- Pessimistic: always pays full complexity cost
- Typical mismatch rate in real datasets around 5%

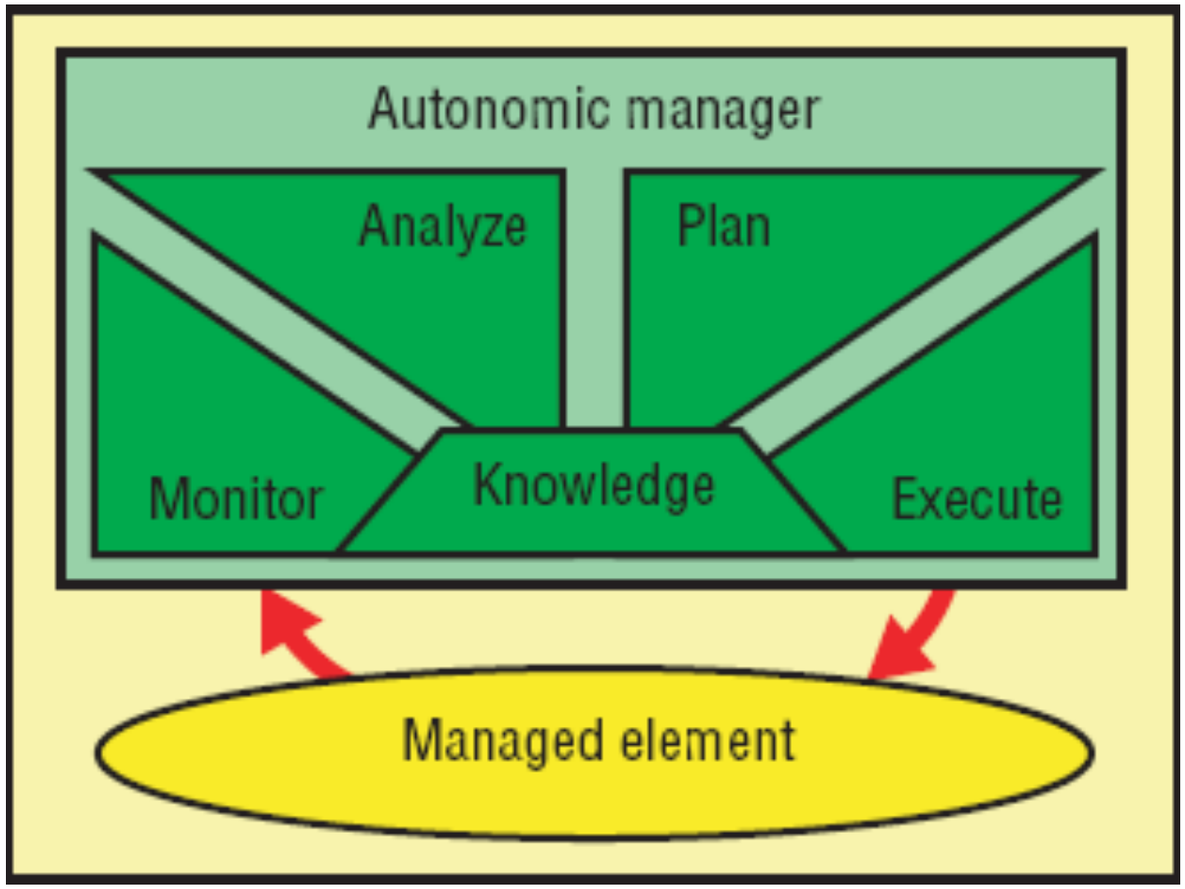
Research Goal: explore optimistic approach

- detect mismatches and react as you go
- requires estimates of incremental join result size
- statistical + reactive ⇒
  - expect to sacrifice join result completeness for faster execution

Approach:

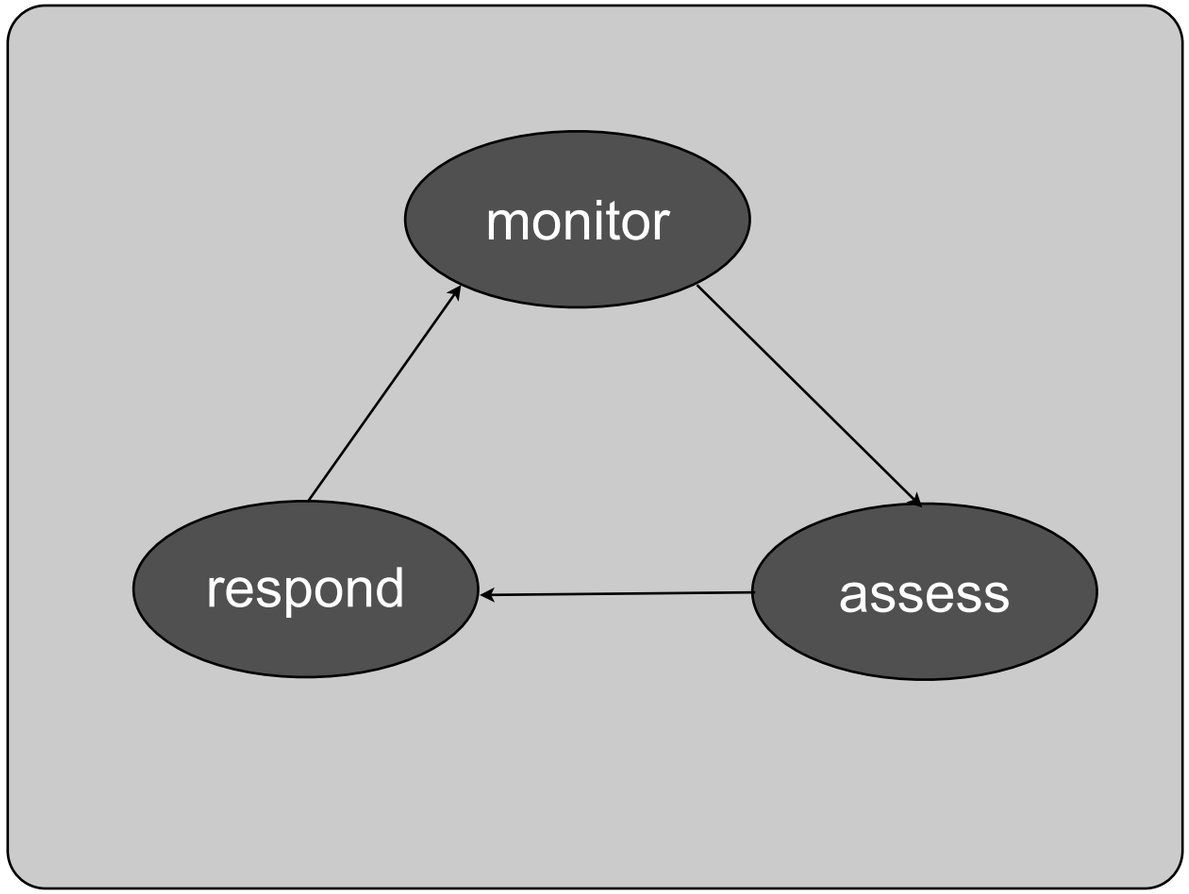
- combine exact and similarity join operators using Adaptive Query Processing techniques

Autonomic computing framework

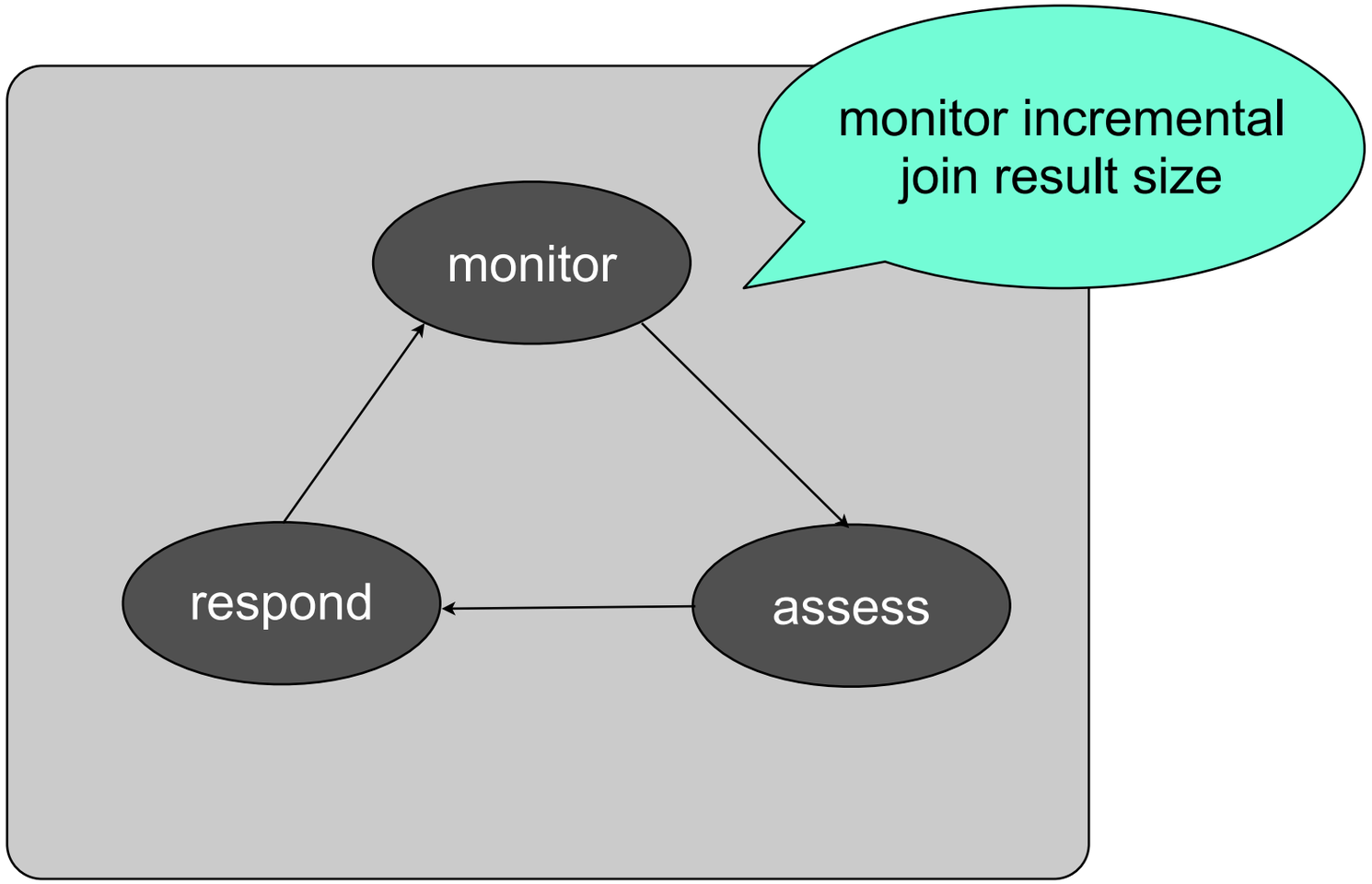


[KC03] J. O. Kephart and D. M. Chess. The vision of autonomic computing. IEEE Computer, 36(1):41–50, 2003.

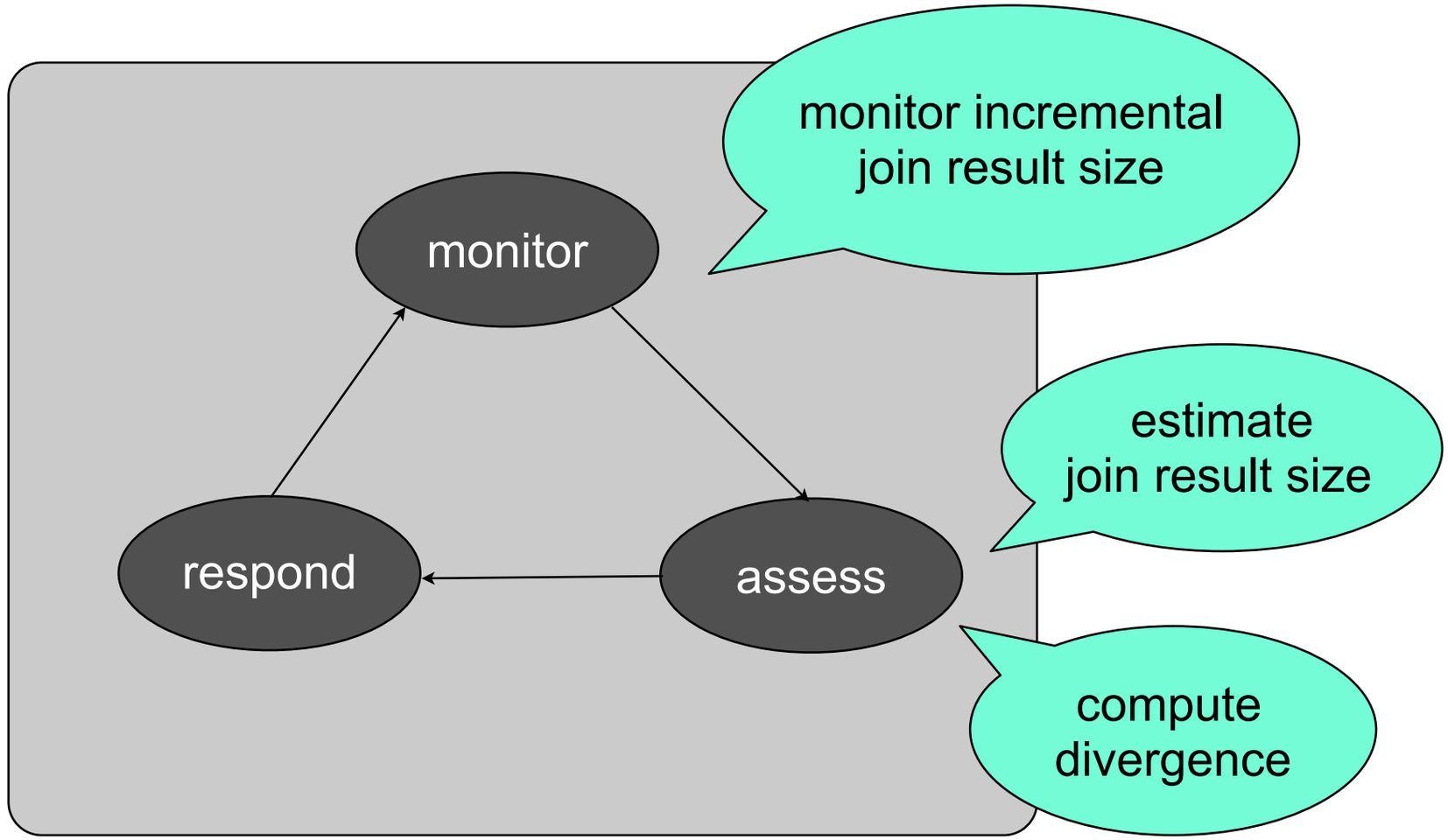
Autonomic computing framework



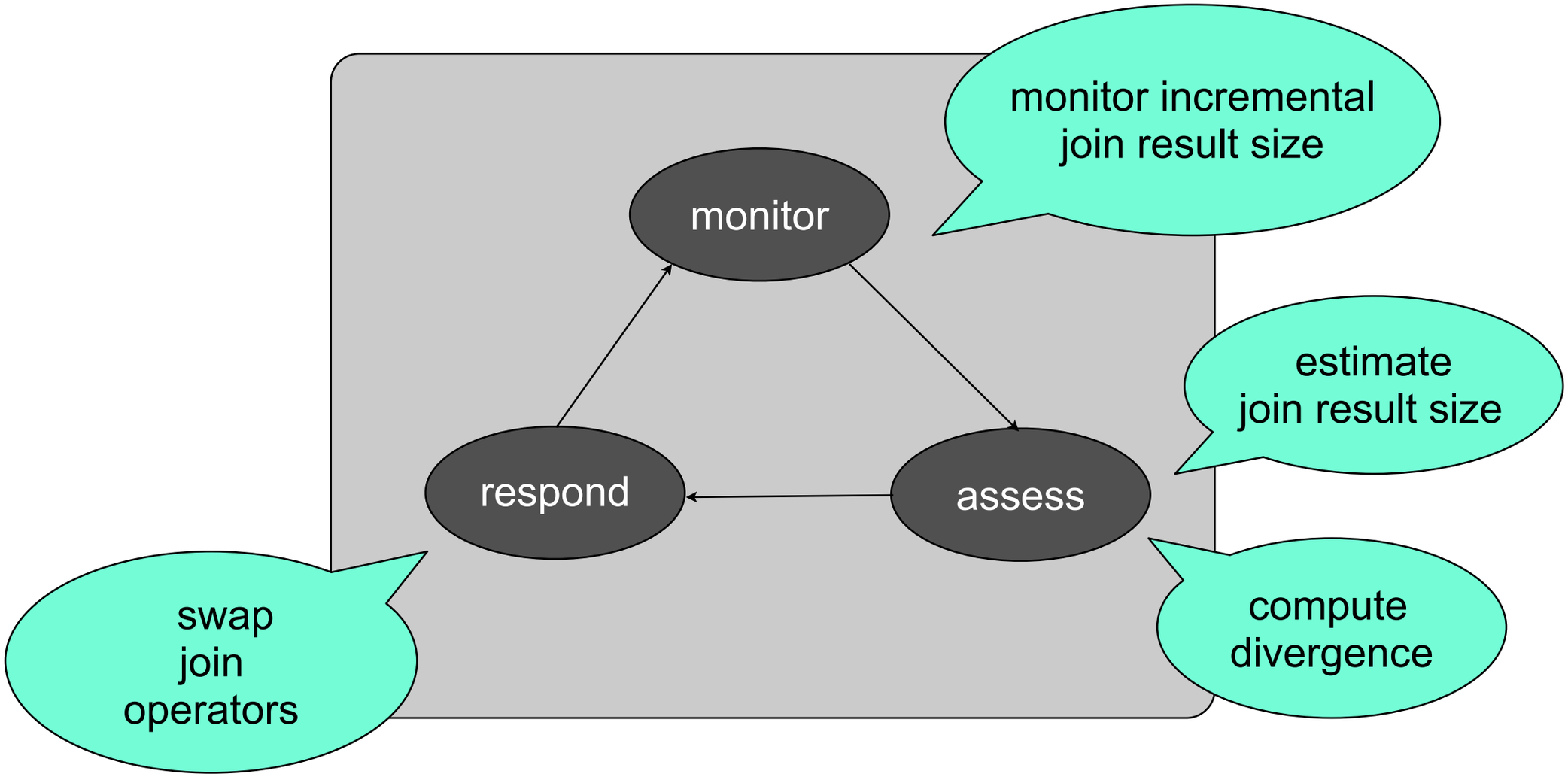
Autonomic computing framework



Autonomic computing framework



Autonomic computing framework



- when using exact join:
  - if observed/estimated sizes diverge “too much”, then switch to approximate join
- when using approximate join:
  - if observed/estimated converge, then switch to exact join

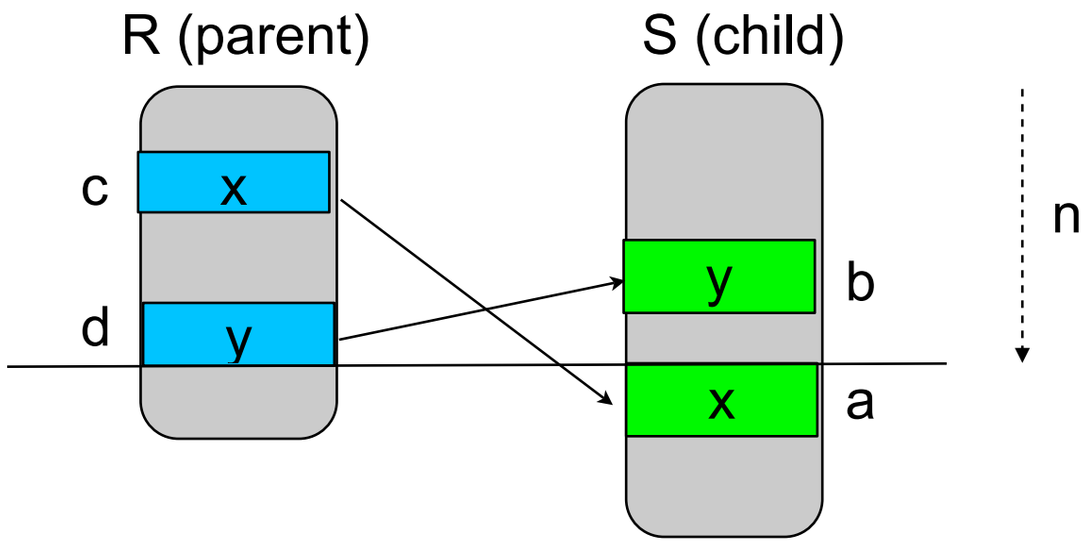
Technical approach and challenges

- Assess:
  - need additional assumption for result size estimation
  - estimating result size at specific points during join execution
- Respond:
  - when and how can physical join operators switched?
  - Can we avoid loss of work?
    - operator replacement in pipelined query plans [EFP06]

[EFP06] K. Eurviriyanukul, A. A. A. Fernandes, and N. W. Paton. A foundation for the replacement of pipelined physical join operators in adaptive query processing. In EDBT Workshops 2006, LNCS 4254

Assessment

- Expectation of matching records
  - implicit referential integrity between tables
- Used to derive simple result size estimation model



- When there are no mismatches:

  after scanning *n* < |S| tuples on S:

  P(a=x in |S| has been matched) = P(tuple c=x is in top n of R) = n/|R|

  Thus, join result size O_n after *n* tuples is a binomial random variable:

$$O_n \sim bin(n, \frac{n}{|R|})$$

Detecting divergent observed result size

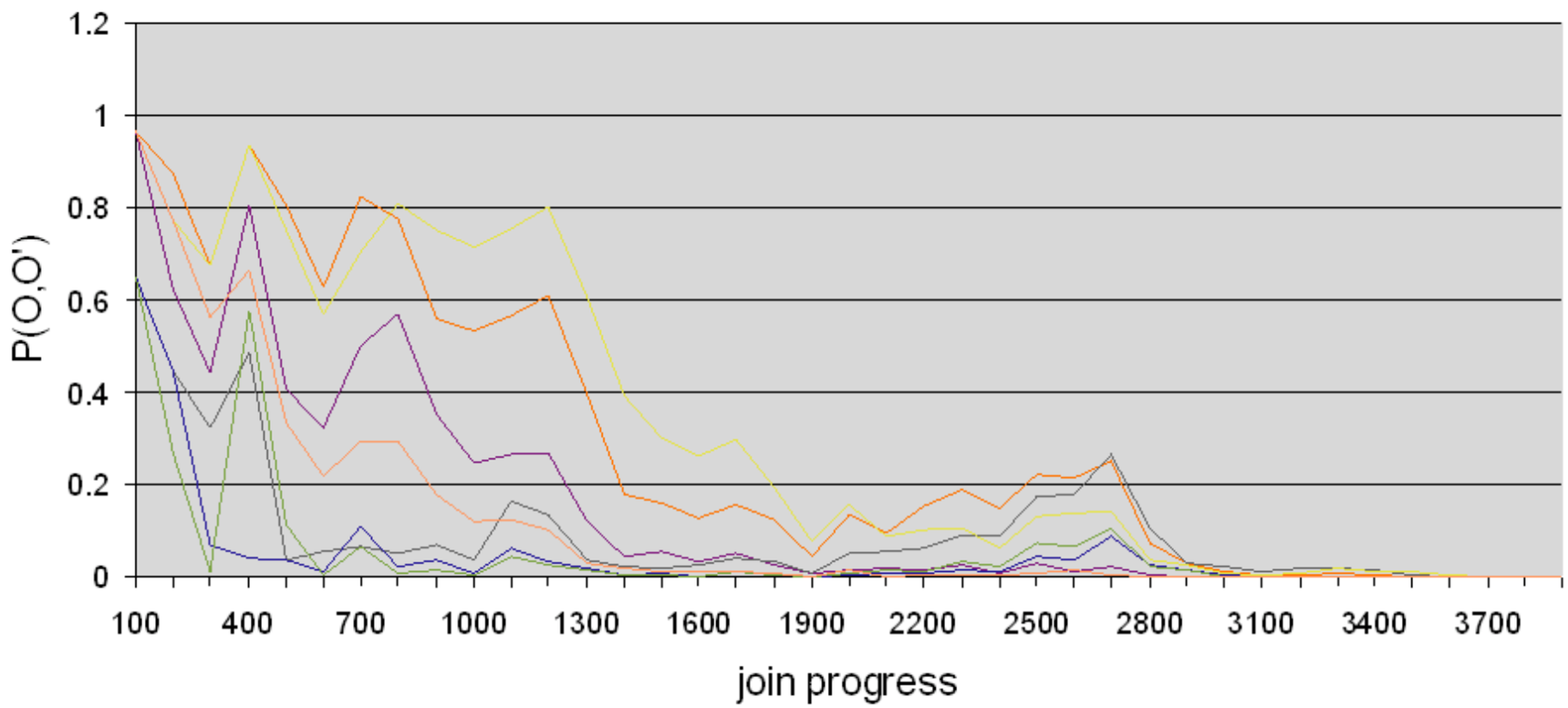
Observation $\bar{O}_n$ is outlier wrt expected result size O_n
=> divergence $P_{n,p(n)}(\bar{O}_n \leq O) \leq \theta_{out}$

where $P_{n,p(n)}$ (.) is the binomial cdf with parameters *n, p(n)*

Detecting divergent observed result size

Observation $\bar{O}_n$ is outlier wrt expected result size O_n
=> divergence $P_{n,p(n)}(\bar{O}_n \leq O) \leq \theta_{out}$

where $P_{n,p(n)}$ (.) is the binomial cdf with parameters *n, p(n)*

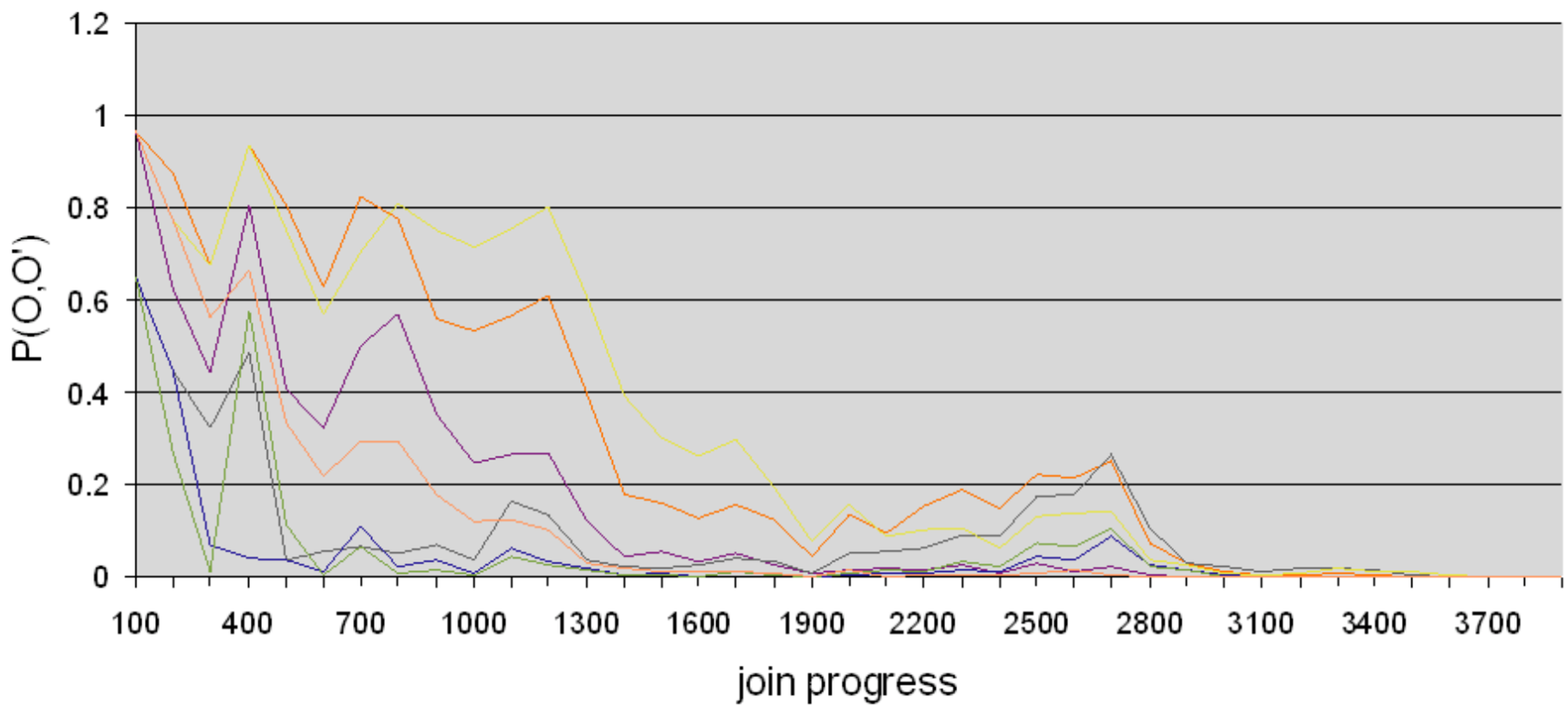


outlier detection on experimental datasets with various mismatch patterns

Detecting divergent observed result size

Observation $\bar{O}_n$ is outlier wrt expected result size O_n
=> divergence $P_{n,p(n)}(\bar{O}_n \leq O) \leq \theta_{out}$

where $P_{n,p(n)}$ (.) is the binomial cdf with parameters *n, p(n)*

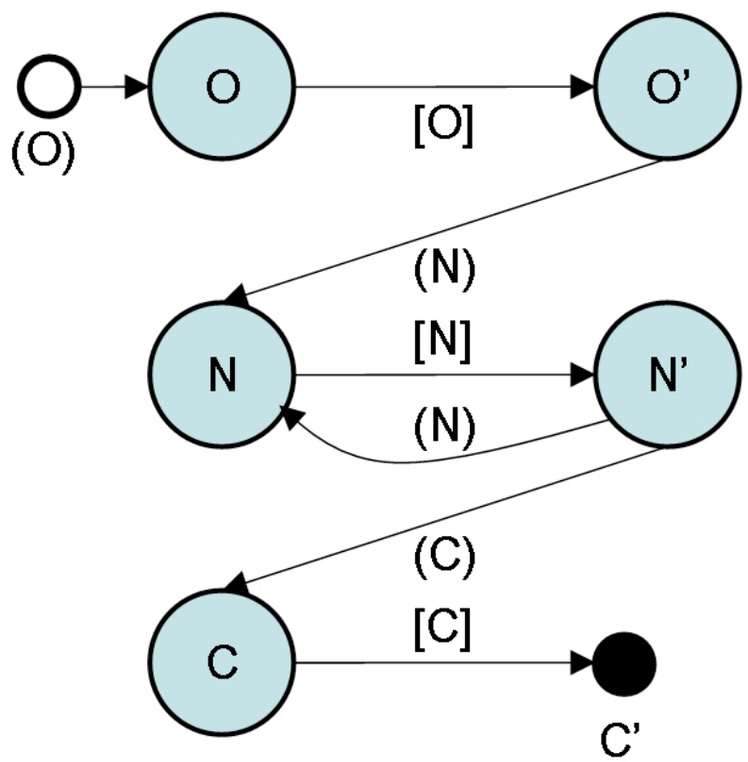


outlier detection on experimental datasets with various mismatch patterns

Note: when the data does not follow our referential constraint hypothesis, the model leads to a pure similarity join

Condition for operator replacement

- Goal of AQP:
  - switch operators without loss of intermediate work
- sufficient condition: switch when the operator reaches a *quiescent* state [EPF06]

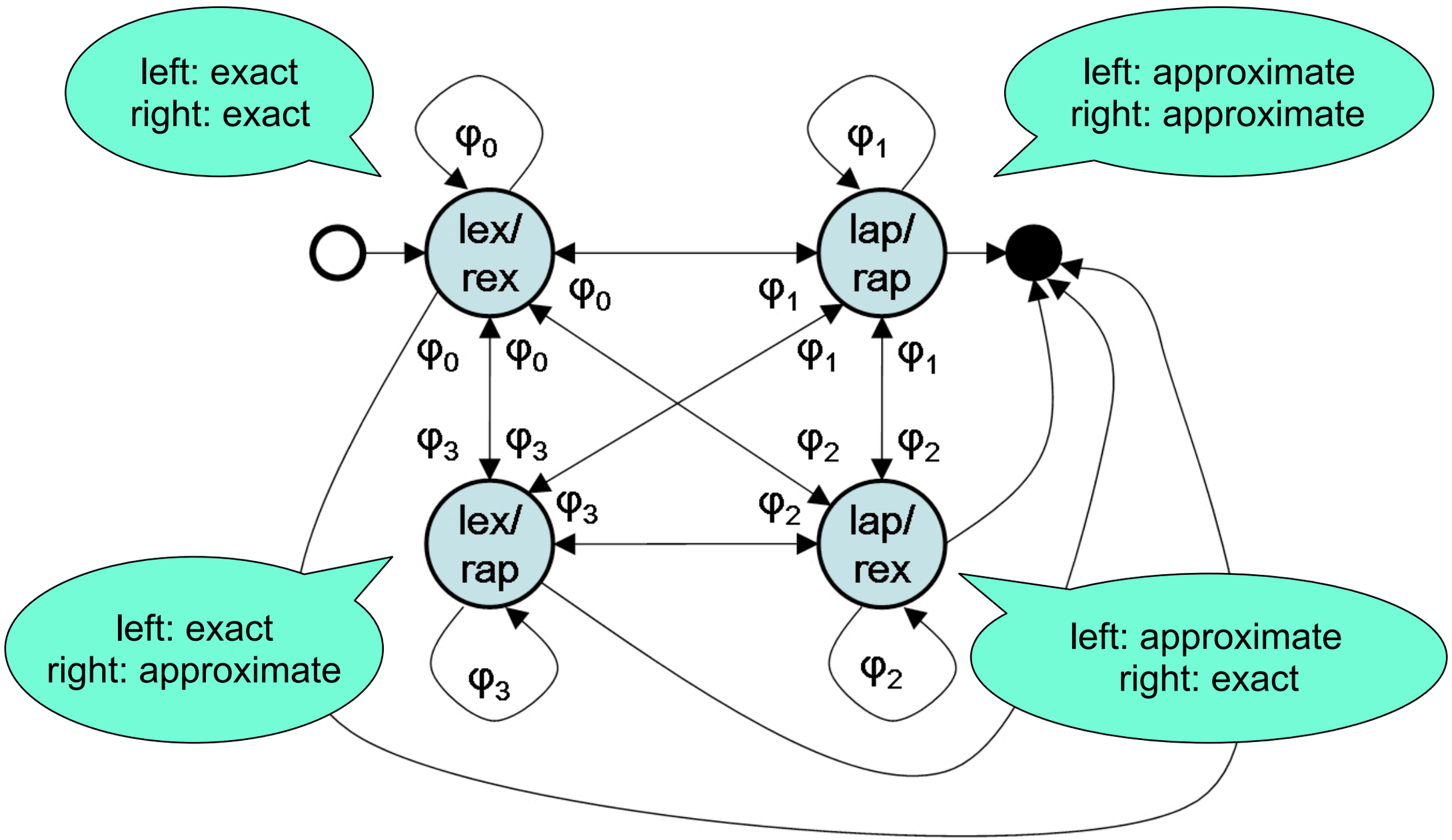


- (O) = Open() called
- O = Open() in progress
- [O] = Open() returned
- O' = Open() concluded
- (N) = Next() called
- N = Next() in progress
- [N] = Next() returned
- N' = Next() concluded
- (C) = Close() called
- C = Close() in progress
- [C] = Close() returned
- C' = Close() concluded

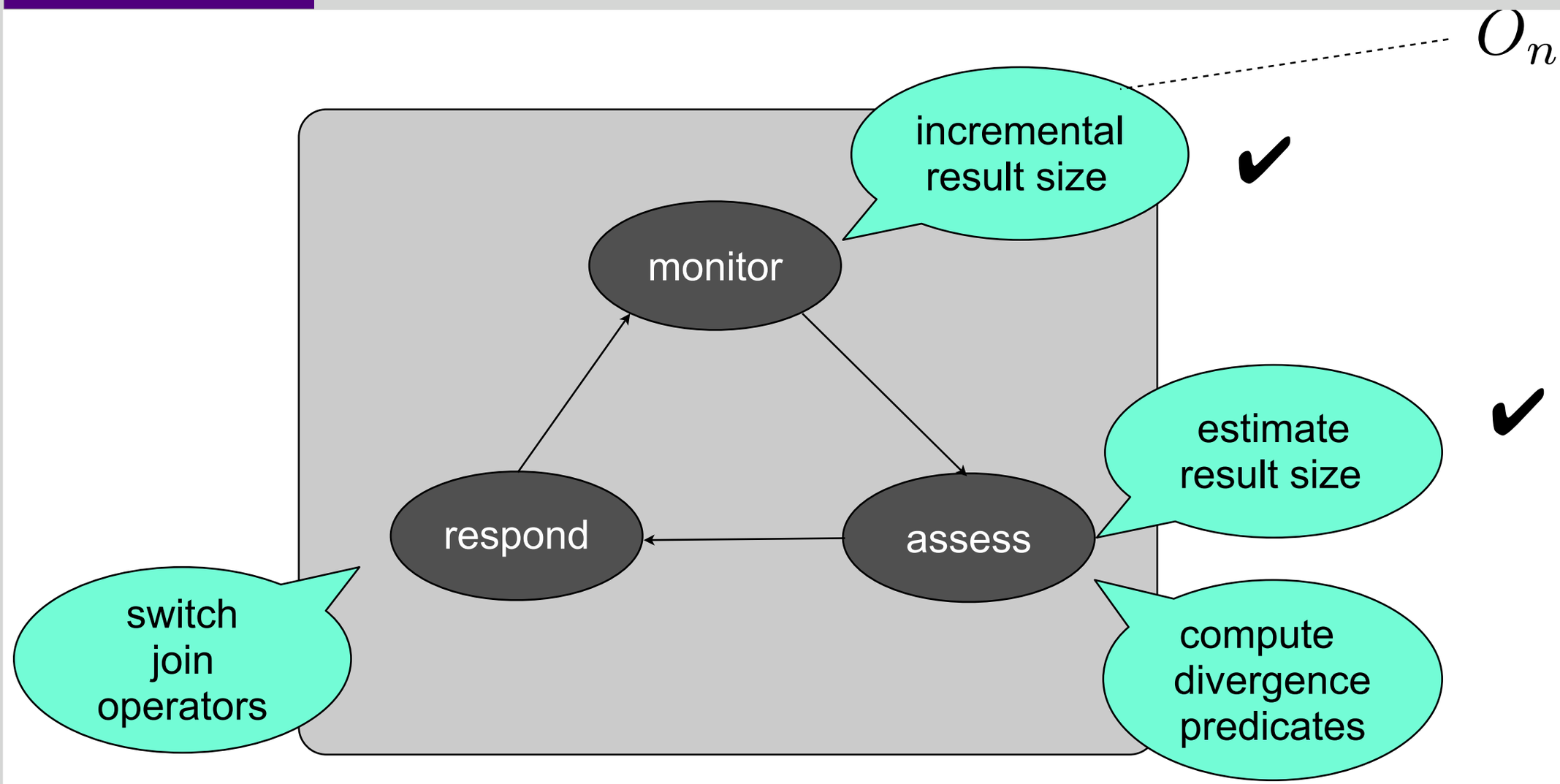
[EPF06] K. Eurviriyanukul, A. A. A. Fernandes, and N. W. Paton. A foundation for the replacement of pipelined physical join operators in adaptive query processing. In EDBT Workshops 2006, LNCS 4254

Responder's state machine

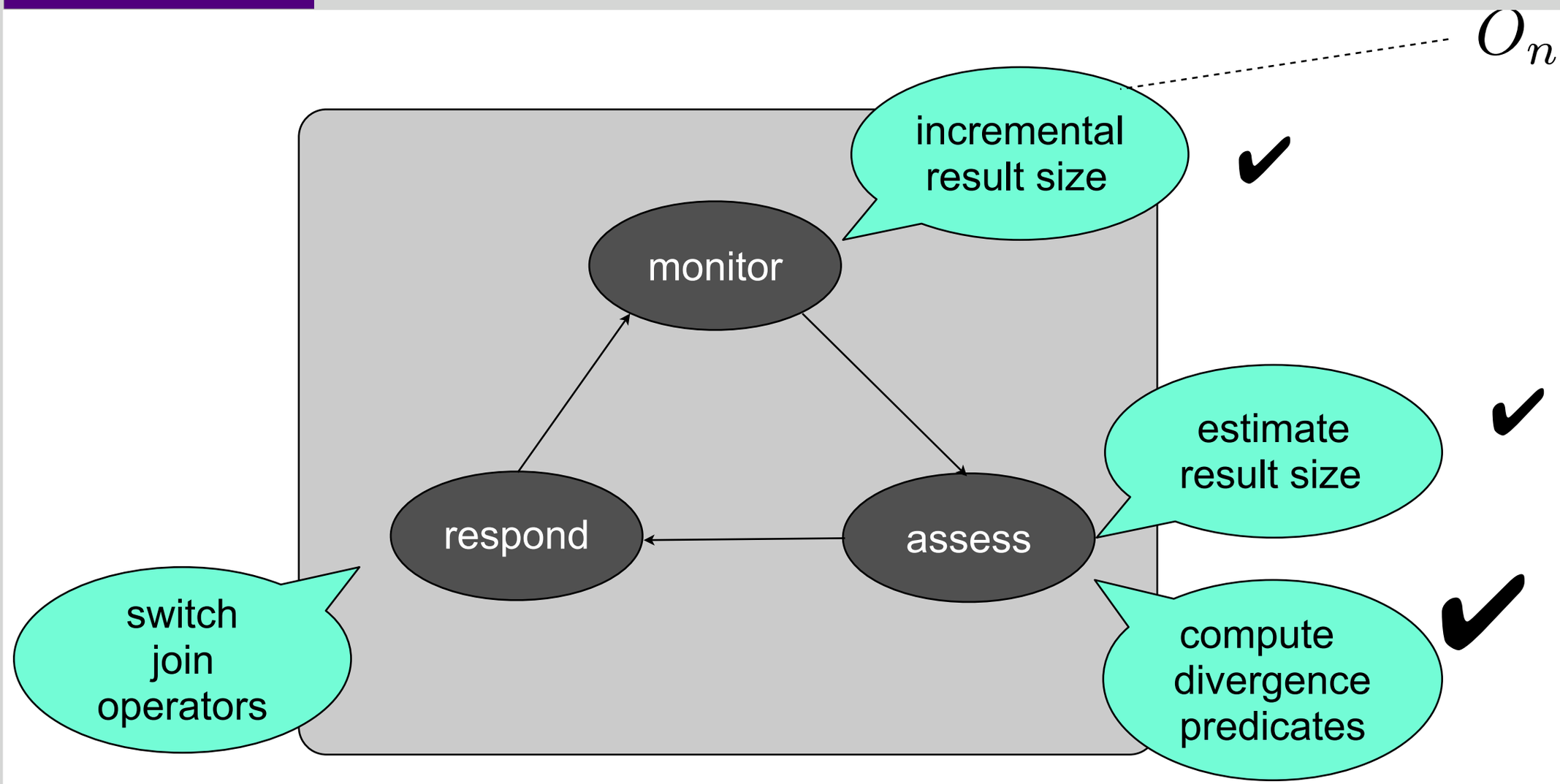
- Operator switch defined in terms of state transitions
- Owing to symmetry, we can use a different operator on each of the two tables



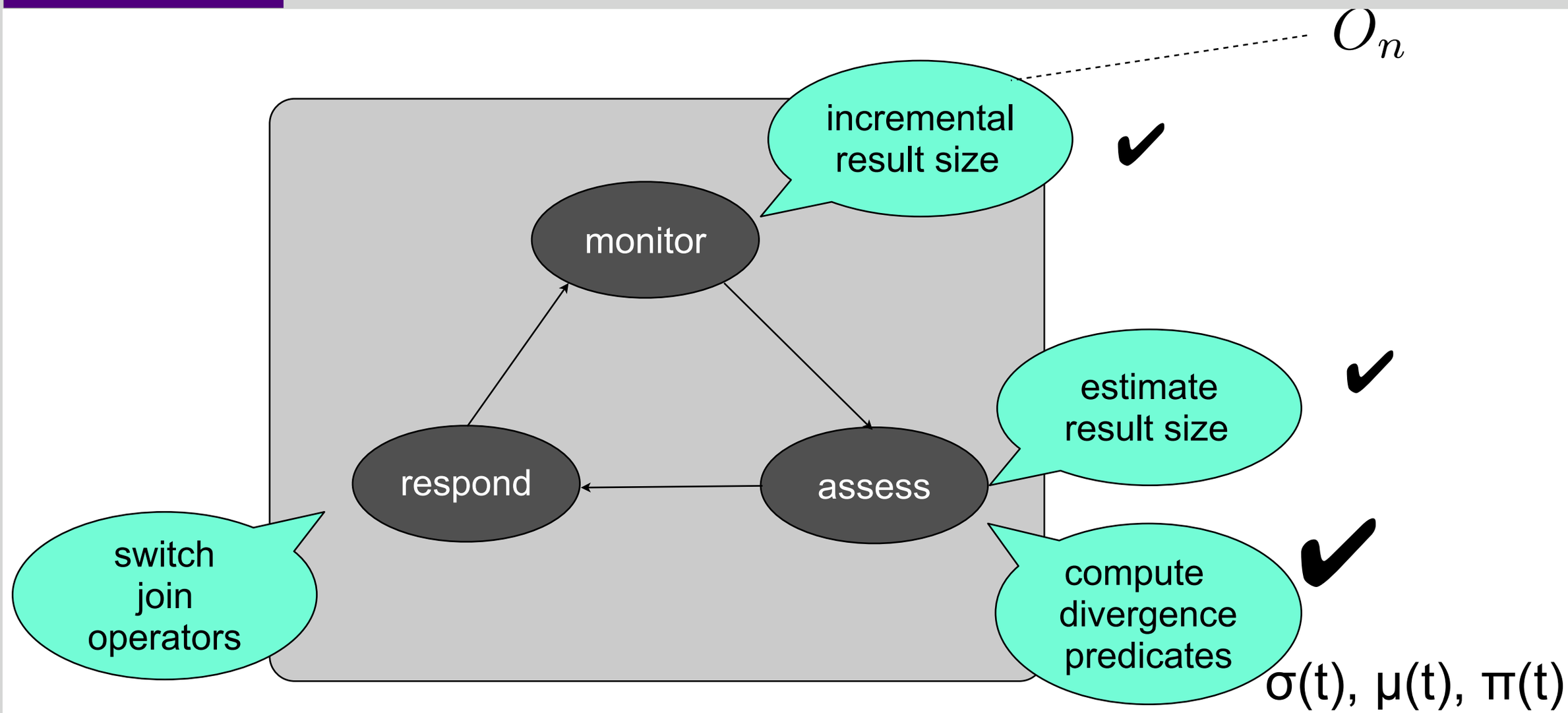
Instantiating the MAR framework



Instantiating the MAR framework



Instantiating the MAR framework



$$\sigma(n) \equiv P_{n,p(n)}(\bar{O}_n \leq O) \leq \theta_{out}$$

Discrepancy detected

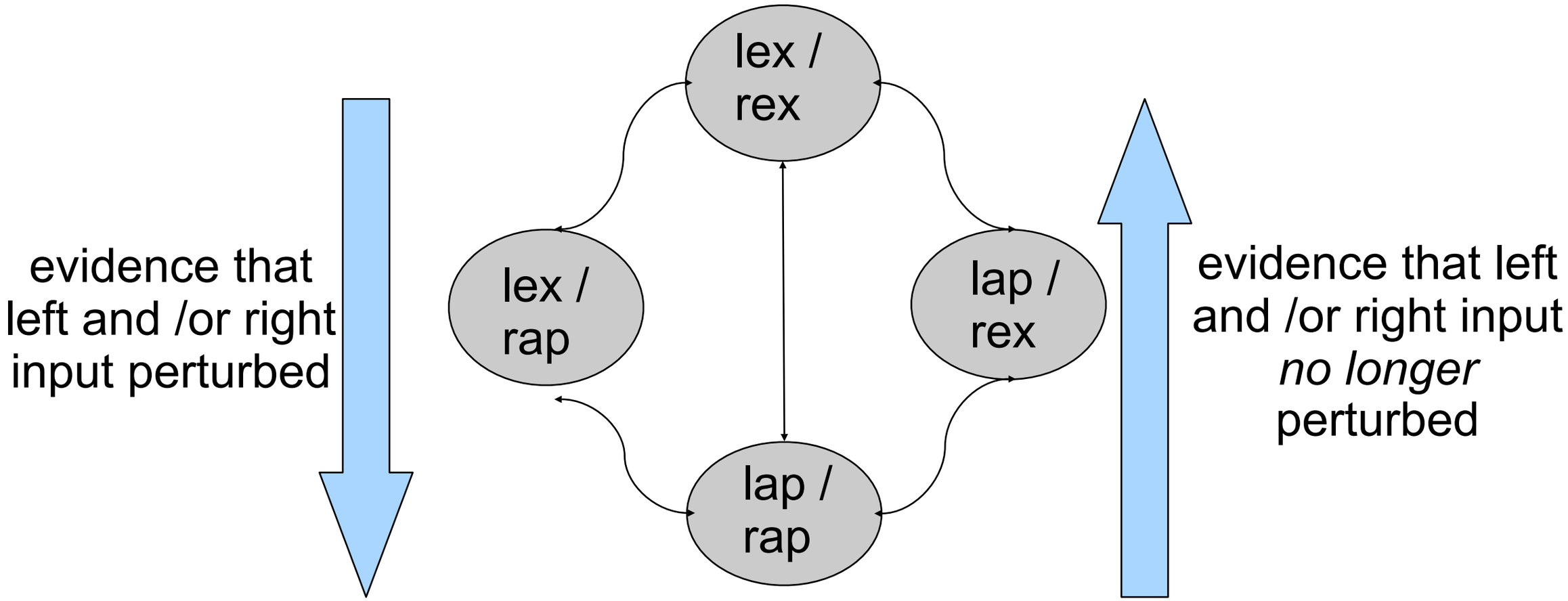
$$\mu_i(t) \equiv \frac{A_{t,W}}{W} \leq \theta_{curpert}$$

Current perturbations on left/right?

$$\pi_i(t) \equiv \sum_{t' < t} I(\mu_i(t')) \leq \theta_{pastpert}$$

Past perturbations on left/right?

Rationale for state transitions



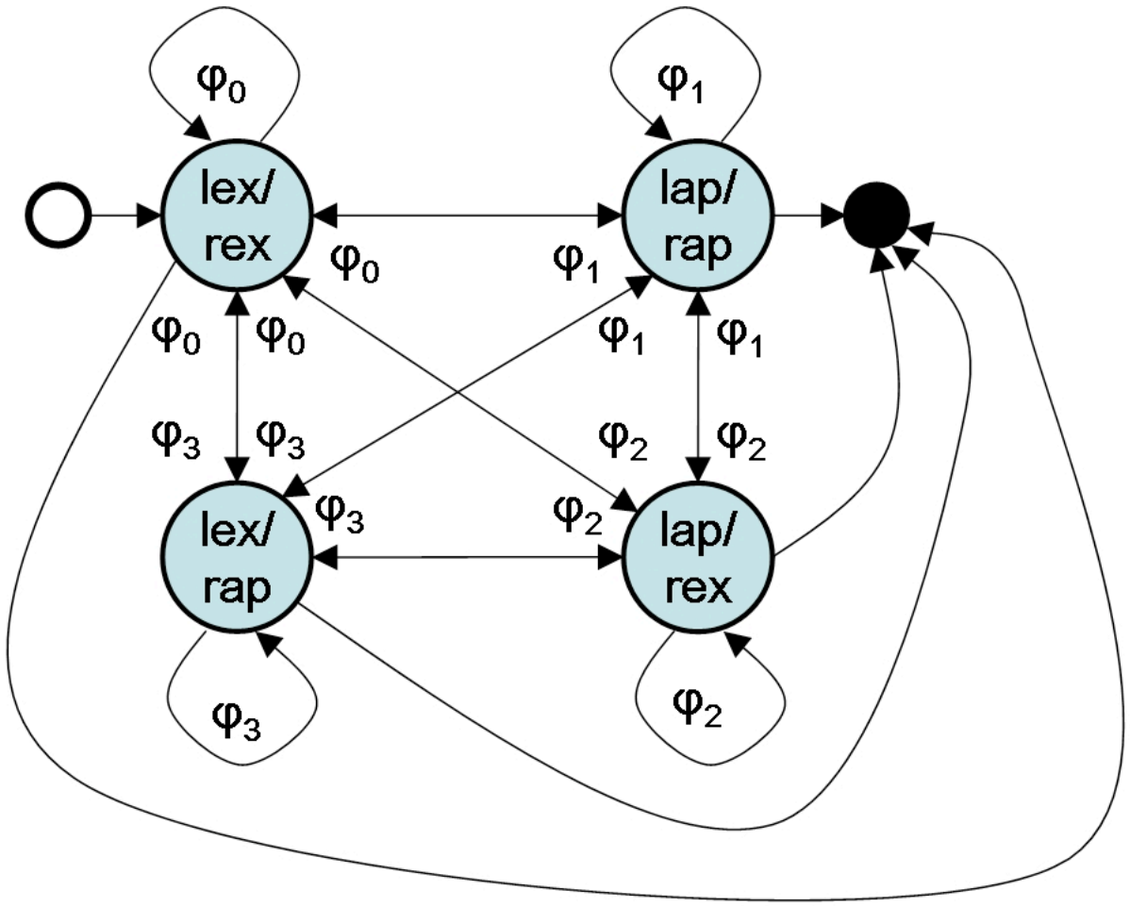
predicates σ(t), μ(t), π(t) provide the evidence needed to drive the transitions

Assessment → transitions predicates

$$\sigma(n) \equiv P_{n,p(n)}(\bar{O}_n \leq O) \leq \theta_{out}$$

$$\mu_i(t) \equiv \frac{A_{t,W}}{W} \leq \theta_{curpert}$$

$$\pi_i(t) \equiv \sum_{t'<t} I(\mu_i(t')) \leq \theta_{pastpert}$$

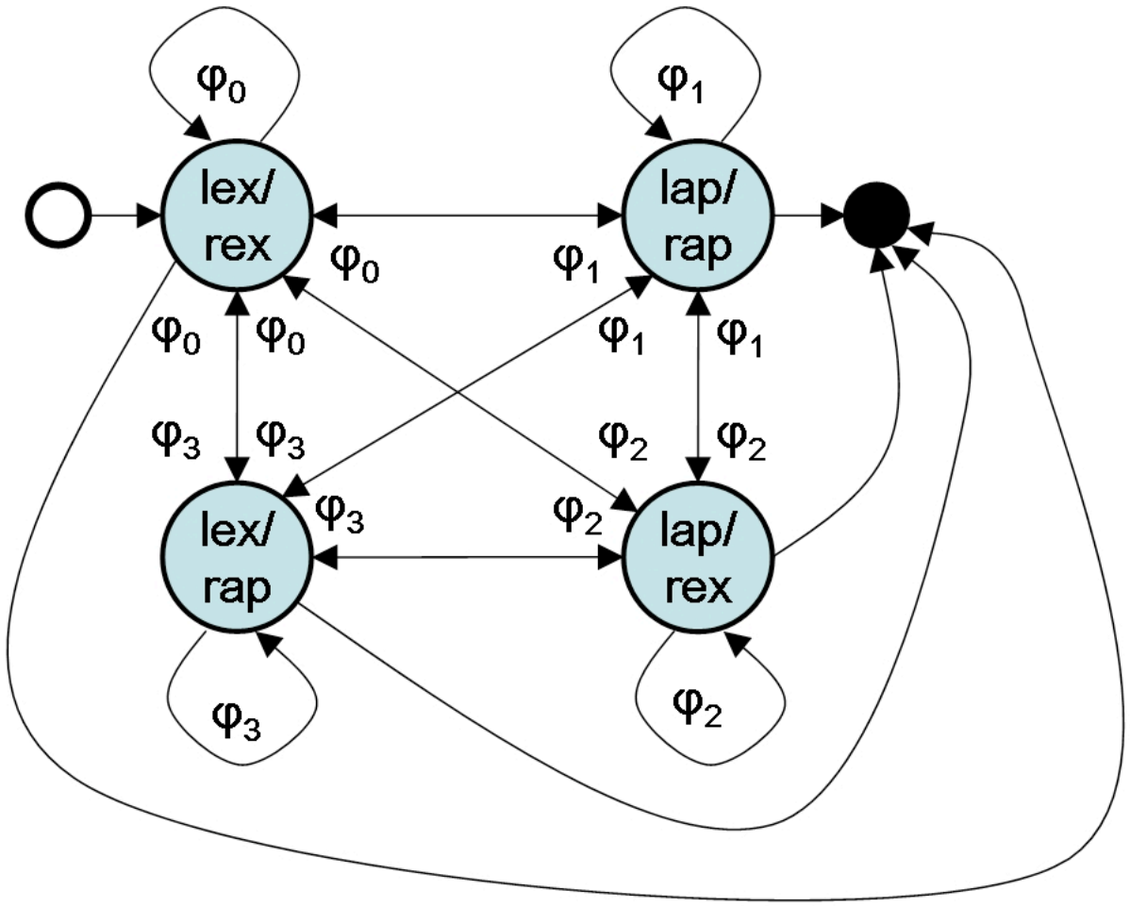


Assessment → transitions predicates

$$\sigma(n) \equiv P_{n,p(n)}(\bar{O}_n \leq O) \leq \theta_{out}$$

$$\mu_i(t) \equiv \frac{A_{t,W}}{W} \leq \theta_{curpert}$$

$$\pi_i(t) \equiv \sum_{t'<t} I(\mu_i(t')) \leq \theta_{pastpert}$$

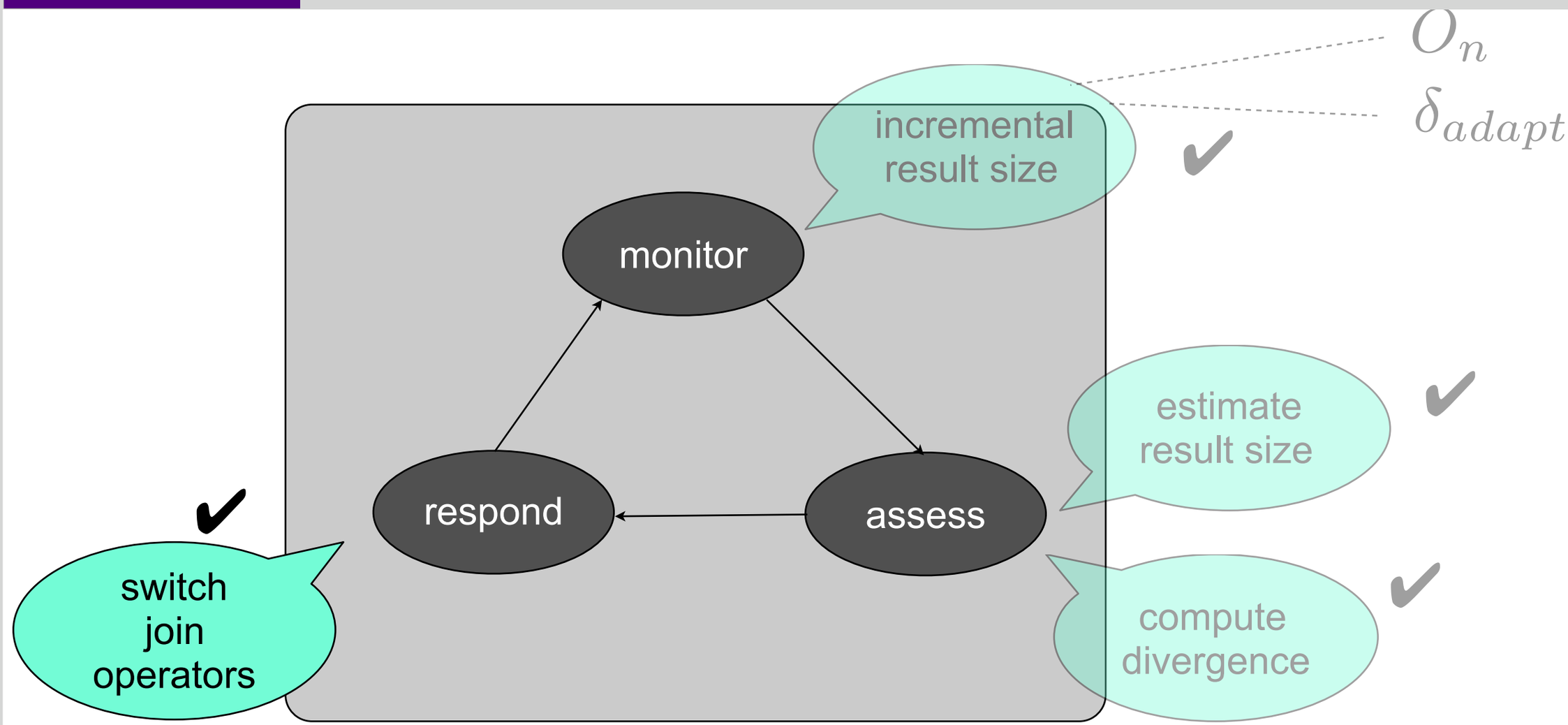


$$\varphi_0(t) = \neg\sigma(t) \wedge \mu_{\text{left}}(t) \wedge \mu_{\text{right}}(t)$$

$$\varphi_1(t) = \sigma(t) \wedge \neg\mu_{\text{left}}(t) \wedge \neg\mu_{\text{right}}(t)$$

$$\varphi_2(t) = \sigma(t) \wedge \neg\mu_{\text{left}}(t) \wedge \mu_{\text{right}}(t) \wedge \pi_{\text{left}}(t)$$

Completing the loop



$$\varphi_0(t) = \neg\sigma(t) \wedge \mu_{\text{left}}(t) \wedge \mu_{\text{right}}(t)$$

$$\varphi_1(t) = \sigma(t) \wedge \neg\mu_{\text{left}}(t) \wedge \neg\mu_{\text{right}}(t)$$

$$\varphi_2(t) = \sigma(t) \wedge \neg\mu_{\text{left}}(t) \wedge \mu_{\text{right}}(t) \wedge \pi_{\text{left}}(t)$$

$$\sigma(n) \equiv P_{n,p(n)}(\bar{O}_n \leq O) \leq \theta_{out}$$

$$\mu_i(t) \equiv \frac{A_{t,W}}{W} \leq \theta_{curpert}$$

$$\pi_i(t) \equiv \sum_{t' < t} I(\mu_i(t')) \leq \theta_{pastpert}$$

Experimental evaluation

- Marginal gain of hybrid algorithm:
  - level of completeness
    - R: result size for approx join only
    - r: result size for exact only
    - r_{abs}: result size actually observed

$$g_{rel} = (r_{abs} - r) / (R - r)$$

Experimental evaluation

- Marginal gain of hybrid algorithm:
  - level of completeness
    - R: result size for approx join only
    - r: result size for exact only
    - r_{abs}: result size actually observed

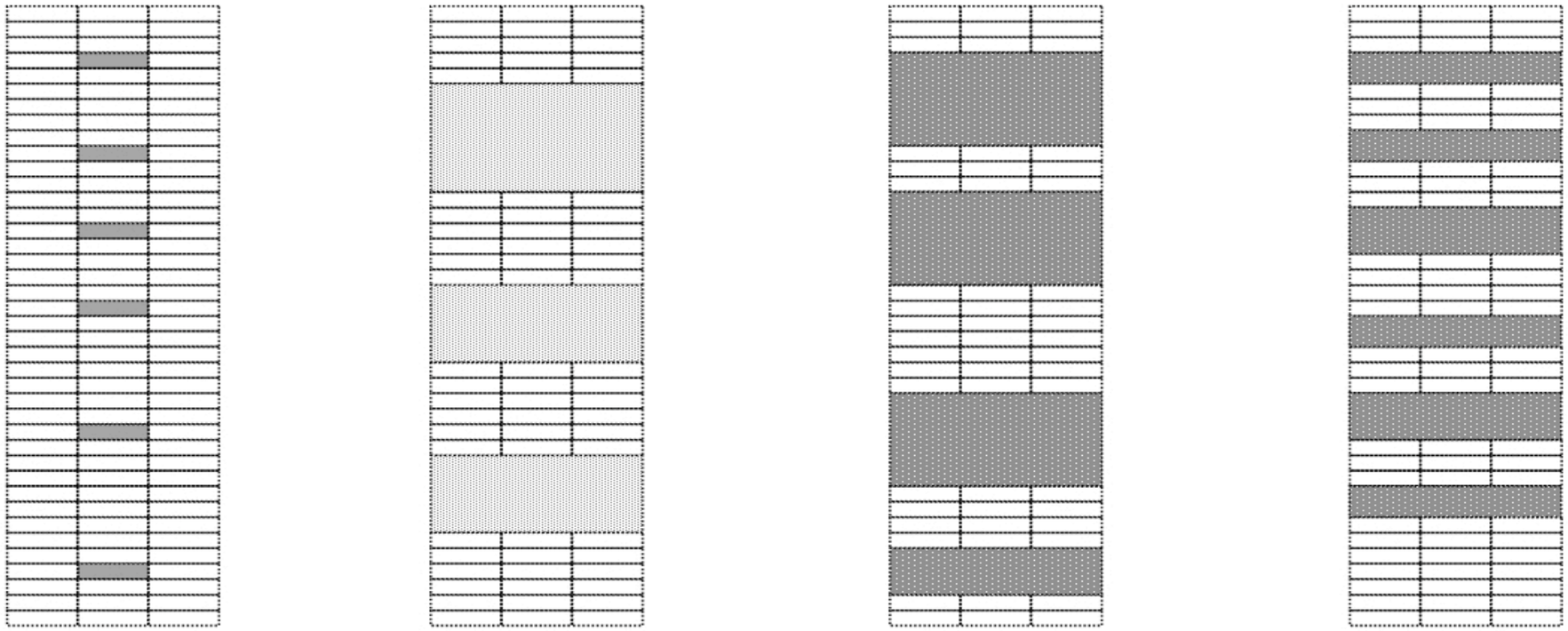
$$g_{rel} = (r_{abs} - r) / (R - r)$$

- Marginal Cost
  - baseline: exact join throughout
    - model marginal cost of hybrid algorithm

unit cost of executing one step in one state
  - (experimental)
- number of steps in each state
- unit state transition cost (experimental)
- number of state transitions over entire join

Test datasets

Datasets chosen as representative of 4 distinct patterns



a) No distinctly-marked perturbation regions

b) Few, long perturbation regions of low density

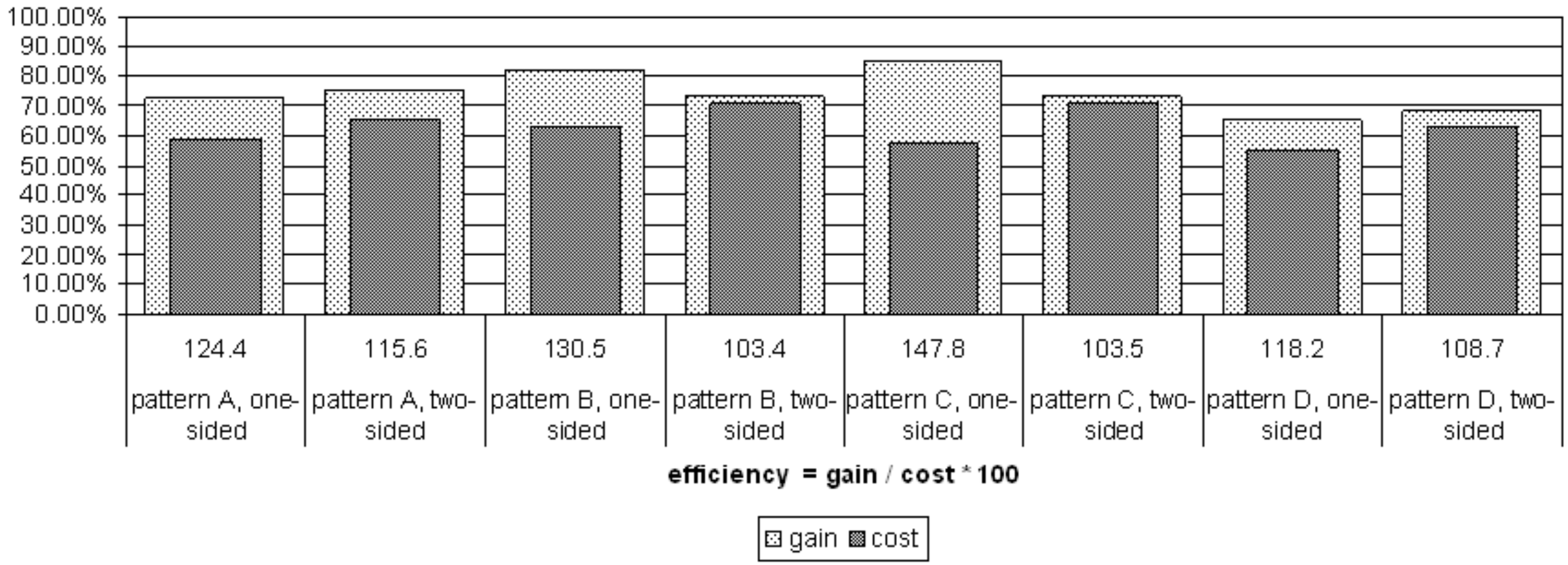
c) Few, long perturbation regions of high density

d) Many, narrow perturbation regions

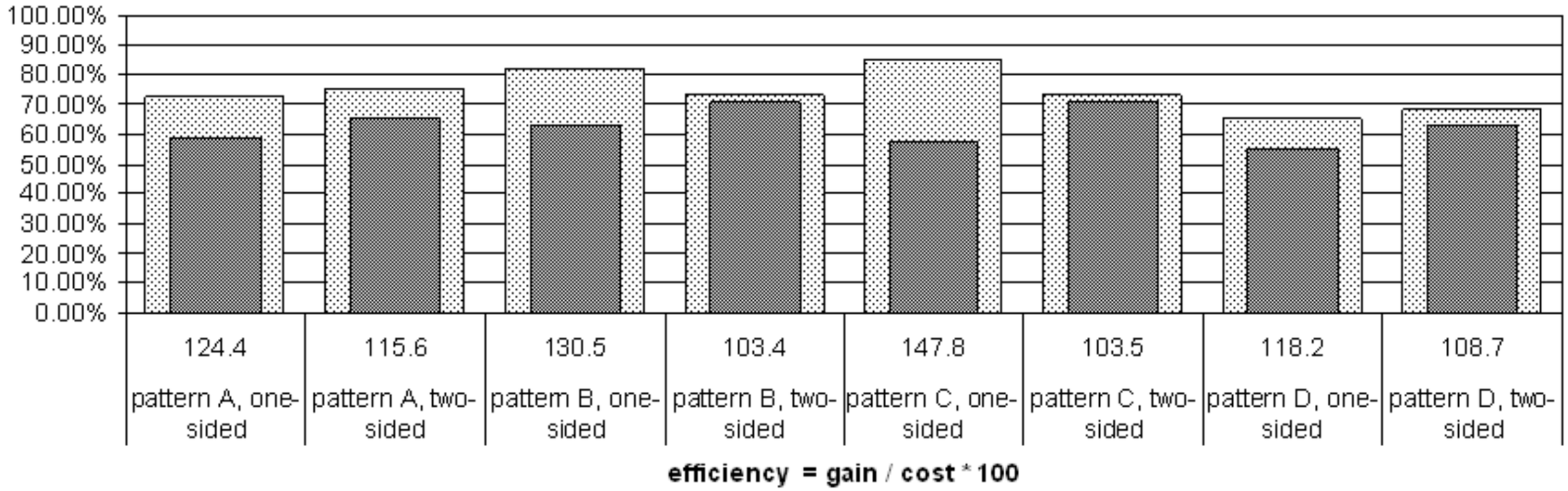
we expect our results to vary:

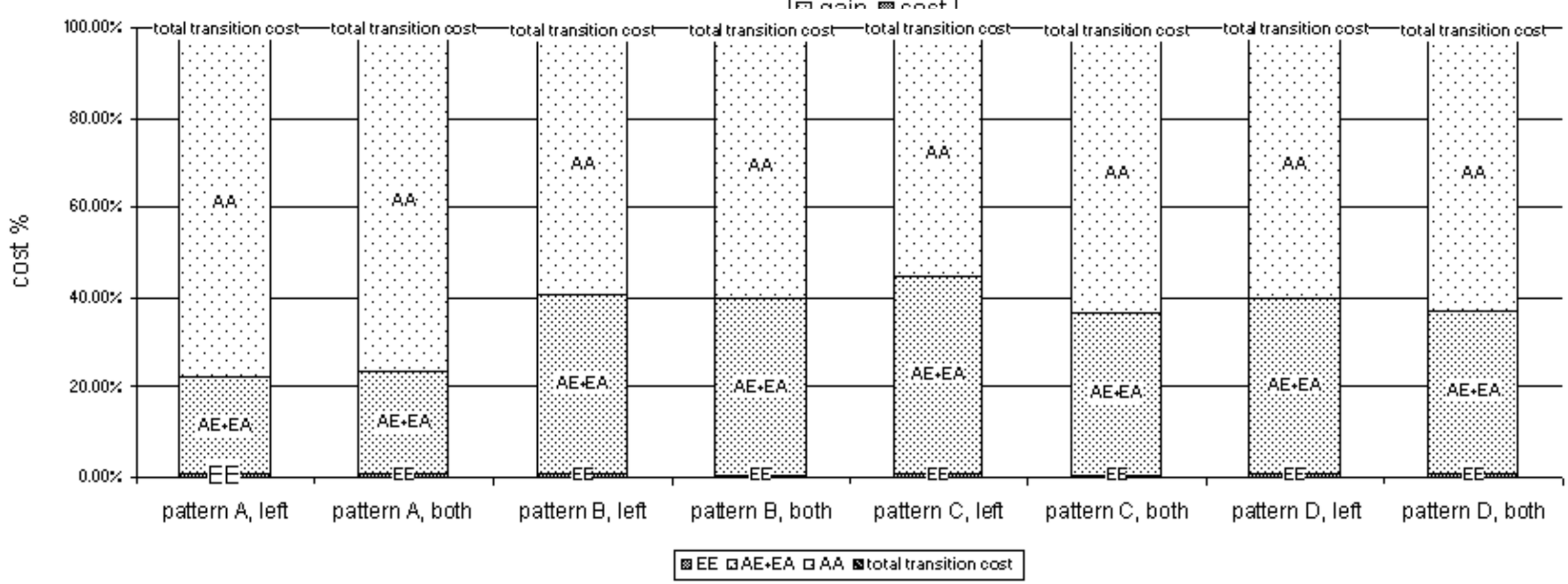
- uniform perturbation: evidence grows slowly => slow reaction
- bursty perturbation: strong evidence => timely reaction

Results


100.00%
90.00%
80.00%
70.00%
60.00%
50.00%
40.00%
30.00%
20.00%
10.00%
0.00%
124.4
115.6
130.5
103.4
147.8
103.5
118.2
108.7
pattern A, one-sided
pattern A, two-sided
pattern B, one-sided
pattern B, two-sided
pattern C, one-sided
pattern C, two-sided
pattern D, one-sided
pattern D, two-sided
efficiency = gain / cost * 100
gain
cost

Results


100.00%
90.00%
80.00%
70.00%
60.00%
50.00%
40.00%
30.00%
20.00%
10.00%
0.00%
124.4
115.6
130.5
103.4
147.8
103.5
118.2
108.7
pattern A, one-sided
pattern A, two-sided
pattern B, one-sided
pattern B, two-sided
pattern C, one-sided
pattern C, two-sided
pattern D, one-sided
pattern D, two-sided
efficiency = gain / cost * 100
gain
cost


100.00%
80.00%
60.00%
40.00%
20.00%
0.00%
cost %
total transition cost
AA
AE+EA
EE
pattern A, left
pattern A, both
pattern B, left
pattern B, both
pattern C, left
pattern C, both
pattern D, left
pattern D, both
EE
AE+EA
AA
total transition cost

Conclusions

- Optimistic approach to online record linkage
  - Based on implicit referential integrity assumption
  - When assumption not true, goes back to pessimistic
- Technical approach based on autonomic computing
  - Adaptive query processing
  - Mix of exact / approximate physical join operators
- Applications: on-the-fly integration scenarios (mashups, personal dataspaces, sensor data streams)
- Results: positive cost / completeness trade-off