Expiration of Database Histories

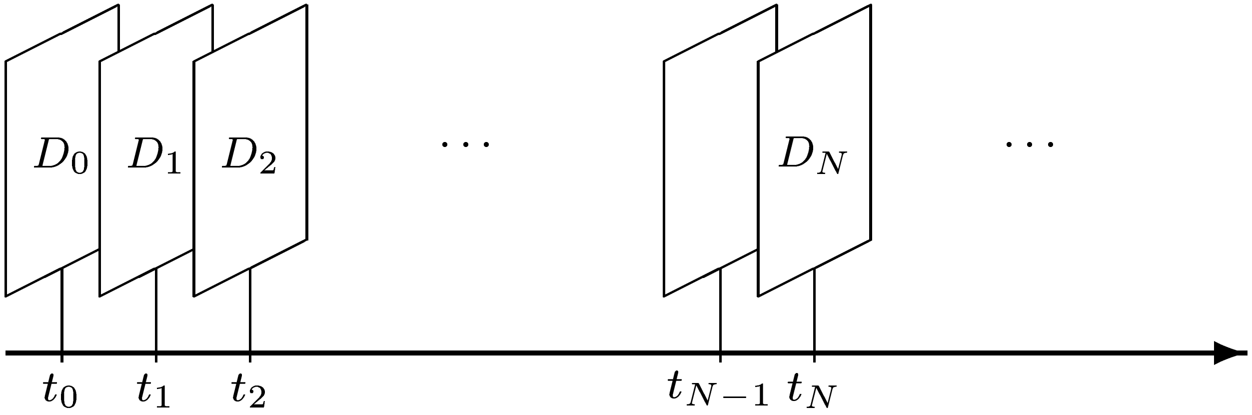
David Toman

Department of Computer Science

University of Waterloo

List of Slides

Database Histories



- fixed schema (for each D_i)
- discrete (integer-like) time $\{0, \ldots, N\}$ (finite sequence)
- **append**-only: $H; D_{N+1}$

: Expiration

- not necessary to “expire” consecutively from the beginning

Examples

- Record keeping/business rules:

  $\Rightarrow$ tax forms must be kept 5 years back
- Dynamic integrity constraints:

  $\Rightarrow$ don't hire anyone you've fired in the past
- Caching policies

  $\Rightarrow$ what data should be moved to backup storage?
- Moving window queries, etc. . .

The data to be removed (expired) is determined by
the (class of) queries we are allowed to ask later

What do we Want?

An **expiration operator** is a map $\mathcal{E} : Q \to Q_{\mathcal{E}}, \mathcal{E}_Q$ for a query Q in a temporal query language $\mathcal{L}$ that

- preserves answers: $Q(H) = Q_{\mathcal{E}}(\mathcal{E}_Q(H))$
- is incrementally maintainable: $\mathcal{E}_Q(H; D) = f(\mathcal{E}_Q(H), D)$ for some function f

for any history H and a "new" state D.

How Good is it?

- Is there a non-trivial (computable) expiration operator?

  $\Rightarrow$ if so, what is the complexity (for an arbitrary $Q \in \mathcal{L}$)?

- What is the space needed by $\mathcal{E}_Q\, H$ in terms of
  1. length of H (number of states, $|\mathbf{dom}_T|$),
  2. the size of the *active data domain* of H (number of constants that have appeared in H, $|\mathbf{dom}_D|$),
  3. size of Q.

Goal: make the size of $\mathcal{E}_Q\, H$ independent of length of H.

$\Rightarrow$ **bounded expiration operator**

Principal Approaches

1. Removal of "old" states (expiration)

$\Rightarrow$ removes a **subset** of existing states

$\Rightarrow$ no other changes (maps history to history)

2. Auxiliary (non-temporal) view maintenance

$\Rightarrow$ maintains **auxiliary** relations

$\Rightarrow$ maps history to *a single extended state*

3. Specialization of queries

$\Rightarrow$ specializes a given query w.r.t. the known prefix H.

Trimming the History

- FO queries with bounded temporal quantifiers to $[E, N]$

  $\Rightarrow E$ is the first non-expired state, N is the current state.

- Trivial implementation
- Periodically reappearing [Garcia-Molina et al., 1998]
- Very limited queries: can't handle $\{x : \exists t.R(t, x)\}$

  only queries over bounded-length past.

Past Temporal Logic

- First-order temporal logic (FO + $\{\CIRCLE, \blacklozenge, \blacksquare, \mathbf{since}\}$)

  $\Rightarrow$ queries over unbounded past: $\{x : \blacklozenge R(x)\}$

- Crux of the approach:

  $\Rightarrow$ make an auxiliary view for each temporal subformula

  $\Rightarrow$ use recurrent definitions of PastTL connectives (e.g., $\blacklozenge\varphi = \varphi \vee \CIRCLE\underline{\blacklozenge\varphi}$) to maintain the views.

- Space utilization: $|\,\mathcal{E}_Q\, H| \in \mathcal{O}(|\mathbf{dom}_D|^{|Q|})$

  $\Rightarrow |\,\mathcal{E}_{\CIRCLE(R(x_1)\wedge\ldots\wedge R(x_k))}\, H| \sim |R| + |R|^k$ (worst case)

Full details: [Chomicki, 1995], subsumes approaches based on TRA [Yang and Widom, 1998, Yang and Widom, 2000].

Two-sorted First-order Language

- Temporal Relational Calculus (2-FOL)

$$L ::= R(t, \mathbf{x}) | x = x' | t < t' | t = t' | L \wedge L | L \wedge \neg L | L \vee L | \exists x.L | \exists t.L$$

where $R(t, \mathbf{x})$ is true in H iff $R(\mathbf{x})$ is true in D_t

- FOTL cannot express all 2-FOL queries

  $\Rightarrow$ [Abiteboul et al., 1996, Toman and Niwinski, 1996]

Does a **bounded expiration operator** exist for 2-FOL?

$\Rightarrow$ conjectured that it does NOT exist

Expiration Revisited

Idea: remove those states that

1. do not contribute to query answer (due to $\wedge$)
2. contribute duplicate information (due to $\exists$)

Easy for a fixed history:

$\Rightarrow$ compute answer to Q bottom-up

$\Rightarrow$ propagate "back" to remove redundant data

Optimal expiration: **undecidable**.

Handling History Extensions

Atomic formulas:

- $\left[{x \atop a}\right] \equiv \begin{cases} x = a & a \in \mathbf{dom}_D \\ \forall a \in \mathbf{dom}_D . x \neq a & a = \bullet \end{cases}$

- $\left[{t \atop s}\right] \equiv \begin{cases} t = s & s \in \mathbf{dom}_T \\ t > \text{maxtime}(\mathbf{dom}_T) & s = \bullet \end{cases}$

Specialization of base relations *and their extensions*:

$$R(t, \mathbf{x}) \equiv \left(\bigvee_{\mathbf{a} \in R_{D_s}} \text{true}\left[{t\mathbf{x} \atop s\mathbf{a}}\right] \right) \vee \left(\bigvee_{\mathbf{a} \in \mathbf{dom}_D \cup \{\bullet\}} R(t, \mathbf{x})\left[{t\mathbf{x} \atop \bullet\mathbf{a}}\right] \right)$$

$\Rightarrow$ disjoint union

$\Rightarrow$ depends **only** on the future extensions

Query Specialization

$$
\mathsf{PE}_H(Q) = \begin{cases}
\{\mathsf{true}[{}^{t\mathbf{x}}_{s\mathbf{a}}] : R(s,\mathbf{a}) \in D\} & \\
\qquad \cup \{R(t,\mathbf{x})[{}^{t\mathbf{x}}_{\bullet\mathbf{a}}] : \mathbf{a} \in (\mathbf{dom}_D \cup \{\bullet\})^{|\mathbf{x}|}\} & Q \equiv R(t,\mathbf{x}) \\
\{Q'_1[{}^{\mathbf{x}}_{\mathbf{a}}] : Q'_1[{}^{\mathbf{x}}_{\mathbf{a}}] \in \mathsf{PE}_H(Q_1), \models [{}^{\mathbf{x}}_{\mathbf{a}}] \wedge F\} & Q \equiv Q_1 \wedge F \\
\{Q'_1 \wedge Q'_2[{}^{\mathbf{xy}}_{\mathbf{ab}}] : Q'_1[{}^{\mathbf{x}}_{\mathbf{a}}] \in \mathsf{PE}_H(Q_1), Q'_2[{}^{\mathbf{y}}_{\mathbf{b}}] \in \mathsf{PE}_H(Q_2), \models [{}^{\mathbf{xy}}_{\mathbf{ab}}]\} & Q \equiv Q_1 \wedge Q_2 \\
\{(\exists y. \bigvee_{Q'_1[{}^{\mathbf{x}y}_{\mathbf{a}b}] \in \mathsf{PE}_H(Q_1)} Q'_1)[{}^{\mathbf{x}}_{\mathbf{a}}] : \exists b. Q''_1[{}^{\mathbf{x}y}_{\mathbf{a}b}] \in \mathsf{PE}_H(Q_1)\} & Q \equiv \exists y. Q_1 \\
\{(\exists t. \bigvee_{Q'_1[{}^{\mathbf{x}t}_{\mathbf{a}s}] \in \mathsf{PE}_H(Q_1), s \in \mathsf{TB}_{\mathbf{a}}(t)} Q'_1)[{}^{\mathbf{x}}_{\mathbf{a}}] : \exists s. Q''_1[{}^{\mathbf{x}t}_{\mathbf{a}s}] \in \mathsf{PE}_H(Q_1)\} & Q \equiv \exists t. Q_1 \\
\{Q'_1 \wedge \neg Q'_2[{}^{\mathbf{x}}_{\mathbf{a}}] : Q'_1[{}^{\mathbf{x}}_{\mathbf{a}}] \in \mathsf{PE}_H(Q_1), Q'_2[{}^{\mathbf{x}}_{\mathbf{a}}] \in \mathsf{PE}_H(Q_2)\} & \\
\qquad \cup \{Q'_1[{}^{\mathbf{x}}_{\mathbf{a}}] : Q'_1[{}^{\mathbf{x}}_{\mathbf{a}}] \in \mathsf{PE}_H(Q_1), Q'_2[{}^{\mathbf{x}}_{\mathbf{a}}] \not\in \mathsf{PE}_H(Q_2)\} & Q \equiv Q_1 \wedge \neg Q_2 \\
\{Q'_1 \vee Q'_2[{}^{\mathbf{x}}_{\mathbf{a}}] : Q'_1 \in \mathsf{PE}_H(Q_1)[{}^{\mathbf{x}}_{\mathbf{a}}], Q'_2[{}^{\mathbf{x}}_{\mathbf{a}}] \in \mathsf{PE}_H(Q_2)\} & \\
\qquad \cup \{Q'_1[{}^{\mathbf{x}}_{\mathbf{a}}] : Q'_1[{}^{\mathbf{x}}_{\mathbf{a}}] \in \mathsf{PE}_H(Q_1), Q'_2[{}^{\mathbf{x}}_{\mathbf{a}}] \not\in \mathsf{PE}_H(Q_2)\} & \\
\qquad \cup \{Q'_2[{}^{\mathbf{x}}_{\mathbf{a}}] : Q'_1[{}^{\mathbf{x}}_{\mathbf{a}}] \not\in \mathsf{PE}_H(Q_1), Q'_2[{}^{\mathbf{x}}_{\mathbf{a}}] \in \mathsf{PE}_H(Q_2)\} & Q \equiv Q_1 \vee Q_2
\end{cases}
$$

Duplicate Information Removal

$$(\exists t. \bigvee_{\substack{Q_1'[{}^{\mathbf{x}t}_{\mathbf{a}s}] \in \mathsf{PE}_H(Q_1) \\ s \in \mathsf{TB}_{\mathbf{a}}(t)}} Q_1')[{}^{\mathbf{x}}_{\mathbf{a}}] \text{ where } Q_1''[{}^{\mathbf{x}t}_{\mathbf{a}s}] \in \mathsf{PE}_H(Q_1) \text{ for some } s$$

$\longleftarrow$ what is this??

Definition 1: Let $Q_1[{}^{\mathbf{x}t}_{\mathbf{a}s_1}], Q_2[{}^{\mathbf{x}t}_{\mathbf{a}s_2}] \in \mathsf{PE}_H(Q)$ for $s_1 \neq s_2$.

We define $[{}^{\mathbf{x}t}_{\mathbf{a}s_1}] \sim^D_Q [{}^{\mathbf{x}t}_{\mathbf{a}s_2}]$ iff for any extension D' of D

$$(\mathbf{a}, s_1) \in Q(D; D') \iff (\mathbf{a}, s_2) \in Q(D; D')$$

Definition 2: $\mathsf{TB}_{\mathbf{a}}(t)$ is the set of representatives of the $[{}^{\mathbf{x}t}_{\mathbf{a}s_1}] \sim^D_Q [{}^{\mathbf{x}t}_{\mathbf{a}s_2}]$ equivalence classes [min in time order].

Equivalence in Extensions

$Q = R$: $\left[{\mathbf{x} \atop \mathbf{a}_1}\right] \sim^D_Q \left[{\mathbf{x} \atop \mathbf{a}_2}\right] \iff ((Q_1 = Q_2 = \text{ true}) \vee (\mathbf{a}_1 = \mathbf{a}_2))$

$Q = Q' \wedge F$: $\left[{\mathbf{x} \atop \mathbf{a}_1}\right] \sim^D_Q \left[{\mathbf{x} \atop \mathbf{a}_2}\right] \iff$

$\left(\left[{\mathbf{x} \atop \mathbf{a}_1}\right] \sim^D_{Q'} \left[{\mathbf{x} \atop \mathbf{a}_2}\right] \text{ where } \left[{\mathbf{x} \atop \mathbf{a}_1}\right] \wedge F \text{ and } \left[{\mathbf{x} \atop \mathbf{a}_2}\right] \wedge F \text{ are satisfiable}\right)$,

$Q = \exists y.Q'$: Let $S_1 = \{b : Q'_1\left[{y\mathbf{x} \atop b\mathbf{a}_1}\right] \in \mathsf{PE}_H(Q')\}$,

$S_2 = \{b : Q'_2\left[{y\mathbf{x} \atop b\mathbf{a}_2}\right] \in \mathsf{PE}_H(Q')\}$. $\left[{\mathbf{x} \atop \mathbf{a}_1}\right] \sim^D_Q \left[{\mathbf{x} \atop \mathbf{a}_2}\right] \iff$

$\left((\forall b \in S_1 \exists c \in S_2 . \left[{y\mathbf{x} \atop b\mathbf{a}_1}\right] \sim^D_{Q'} \left[{y\mathbf{x} \atop c\mathbf{a}_2}\right]) \wedge (\forall c \in S_2 \exists b \in S_1 . \left[{y\mathbf{x} \atop b\mathbf{a}_1}\right] \sim^D_{Q'} \left[{y\mathbf{x} \atop c\mathbf{a}_2}\right])\right)$,

$Q = \exists t.Q'$: Let $S_1 = \{s : Q'_1\left[{t\mathbf{x} \atop s\mathbf{a}_1}\right] \in \mathsf{PE}_H(Q')\}$,

$S_2 = \{s : Q'_2\left[{t\mathbf{x} \atop s\mathbf{a}_2}\right] \in \mathsf{PE}_H(Q')\}$. $\left[{\mathbf{x} \atop \mathbf{a}_1}\right] \sim^D_Q \left[{\mathbf{x} \atop \mathbf{a}_2}\right] \iff$

$\left((\forall r \in S_1 \exists s \in S_2 . \left[{t\mathbf{x} \atop r\mathbf{a}_1}\right] \sim^D_{Q'} \left[{t\mathbf{x} \atop s\mathbf{a}_2}\right]) \wedge (\forall r \in S_2 \exists s \in S_1 . \left[{t\mathbf{x} \atop r\mathbf{a}_1}\right] \sim^D_{Q'} \left[{t\mathbf{x} \atop s\mathbf{a}_2}\right])\right)$,

$Q = Q' \wedge Q''$: $\left[{\mathbf{x} \atop \mathbf{a}_1}\right] \sim^D_Q \left[{\mathbf{x} \atop \mathbf{a}_2}\right] \iff$

$\left(\left[{\mathbf{x}' \atop \mathbf{a}'_1}\right] \sim^D_{Q'} \left[{\mathbf{x}' \atop \mathbf{a}'_2}\right] \wedge \left[{\mathbf{x}'' \atop \mathbf{a}''_1}\right] \sim^D_{Q''} \left[{\mathbf{x}'' \atop \mathbf{a}''_2}\right], \text{ for } \left[{\mathbf{x} \atop \mathbf{a}_i}\right] = \left[{\mathbf{x}'\mathbf{x}'' \atop \mathbf{a}'_i\mathbf{a}''_i}\right] \text{ satisfiable}\right)$,

$Q = Q' \wedge \neg Q''$ or $Q = Q' \vee Q''$:

$\left[{\mathbf{x} \atop \mathbf{a}_1}\right] \sim^D_Q \left[{\mathbf{x} \atop \mathbf{a}_2}\right] \iff \left(\left[{\mathbf{x} \atop \mathbf{a}_1}\right] \sim^D_{Q'} \left[{\mathbf{x} \atop \mathbf{a}_2}\right] \wedge \left[{\mathbf{x} \atop \mathbf{a}_1}\right] \sim^D_{Q''} \left[{\mathbf{x} \atop \mathbf{a}_2}\right]\right)$.

Residual History Reconstruction

- *Specialization-based* expiration:

$$
\begin{aligned}
Q(H) &= \mathsf{PE}_H(Q)(\emptyset) \\
\mathsf{PE}_{H;H'}(Q) &\equiv \mathsf{PE}_{H'}(\mathsf{PE}_H(Q))
\end{aligned}
$$

- We use $\mathsf{PE}_H(Q)$ to *construct* $\mathcal{E}(H)$

$\Rightarrow$ for each temporal variable t_i we define a unary relation

$$T_i(t) = \bigcup_{\mathbf{a}} \mathsf{TB}_{\mathbf{a}}(t).$$

$\Rightarrow$ each quantifier $\exists t_i.Q'$ in Q is restricted to $T_i(t)$

$\Rightarrow$ a state $D_j \in H$ is expired if $j \notin \bigcup_i T_i$.

Properties of the Expiration Operator

- $Q(H; H') = Q(\mathcal{E}_Q(H); H')$

  for all H, H' histories and Q FO query

- $|\,\mathcal{E}_Q(H)| \leq f(|\mathbf{dom}_D|, |Q|)$,

  f is an exponential tower in number of nested $\exists$.

- $|\,\mathcal{E}_Q(H)| \leq |H| + |\mathbf{dom}_T||Q|$

. . . and can be implemented by FO queries/updates.

Space: Lower Bound

Example: $\exists t_1, t_2. t_1 < t_2 \wedge \forall x. R(t_1, x) \iff R(t_2, x)$

- Potentially we need to keep all states for which R contains distinct subsets of $\mathbf{dom}_D$

  $\Rightarrow$ potentially all subsets of $\mathbf{dom}_D$

  $\Rightarrow$ therefore any expired history is exponential in $|\mathbf{dom}_D|$.

- Cannot be expressed in TL

  [Abiteboul et al., 1996, Toman and Niwinski, 1996]

Improvements

- More precise abstraction (e.g., for "$\vee$" and "$\wedge\neg$")
- Coordination of the choices of expired values for leaves

  $\Rightarrow$ may lead to space savings

  – how hard is optimizing for this?
  – implementable in FO?
- Expiration after batches of updates

  $\Rightarrow$ amortize cost
- Aggregation

  $\Rightarrow$ fixed-size sliding windows – no problem

What about “Real” Aggregation?

Example: “is the number of states containing a
greater that the number of states containing b?”

$\Rightarrow$ we need $\Omega(\log(|\mathbf{dom}_T|))$ space to represent counter(s)

$\Rightarrow$ $\mathcal{O}(\log(|\mathbf{dom}_T|))$ is sufficient.

Conjecture: we can use the above technique (but remember counts of the expired values) to answer queries with counting

$\Rightarrow$ $\mathcal{E}_Q(H)| \leq POLY(\log(|\mathbf{dom}_T|))$

What about Retroactive Updates?

Example:

while $\exists t.R(t,a) \wedge \exists t.R(t,b)$ **do** **{** while both a and b exist in R **}**
  delete $R(t,a)$
    where $\forall t'.R(t',a) \supset t' > t$; **{** delete (chronologically) first a **}**
  delete $R(t,b)$
    where $\forall t'.R(t',b) \supset t' > t$; **{** delete (chronologically) first b **}**
return $\exists t.R(t,a)$ **{** return true if R contains an a **}**

$\Rightarrow$ we need $\Omega(\log(|\mathbf{dom}_T|))$ space to represent counter(s)

$\Rightarrow$ just like for counting . . .

Summary

Result: there is an *expiration operator* that
for linear time histories

1. preserves answers of a fixed finite set of arbitrary first-order queries, and
2. guarantees that the size of the expired history is independent of its original length

It can be automatically generated from a given query (set).

References

[Abiteboul et al., 1996] Abiteboul, S., Herr, L., and den Bussche, J. V. (1996). Temporal versus first-order logic to query temporal databases. In *Proceedings of the Fifteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 49–57.

[Chomicki, 1995] Chomicki, J. (1995). Efficient checking of temporal integrity constraints using bounded history encoding. *TODS*, 20(2):149–186.

[Garcia-Molina et al., 1998] Garcia-Molina, H., Labio, W., and Yang, J. (1998). Expiring data in a warehouse. In Gupta, A., Shmueli, O., and Widom, J., editors, *24rd International Conference on Very Large Data Bases, VLDB'98*, pages 500–511. Morgan Kaufmann.

[Toman and Niwinski, 1996] Toman, D. and Niwinski, D. (1996). First-order queries over temporal databases inexpressible in temporal logic. In Apers, P. M. G., Bouzeghoub, M., and Gardarin, G., editors, *Advances in Database Technology - EDBT'96*, volume 1057, pages 307–324. Springer.

[Yang and Widom, 1998] Yang, J. and Widom, J. (1998). Maintaining temporal views over non-temporal information sources for data warehousing. In *Proceedings of EDBT 1998*, pages 389–403.

[Yang and Widom, 2000] Yang, J. and Widom, J. (2000). Temporal view self-maintenance. In *Proceedings of EDBT 2000*, pages 395–412.