Processing and Querying Temporal Data

Johann Gamper

Free University of Bozen-Bolzano, Italy

www.inf.unibz.it/∼gamper

Acknowledgments

Joint work with

- M. Böhlen, University of Zurich
- A. Dignös, Free University of Bozen-Bolzano
- C. S. Jensen, Aalborg University

Partially funded by

- Autonomous Province of Bozen-Bolzano
- Swiss National Science Foundation
- EU (Chorochronos)
- unibz (TPG)

Outline

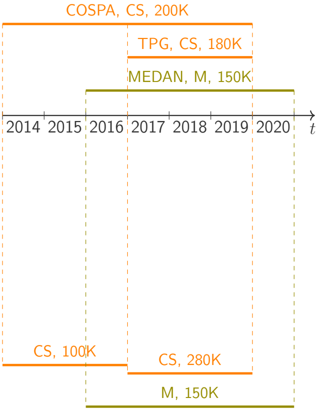
Outline

Temporal Aggregation Example

unibz

- Project relation with timestamps

Project	Dept	Budget	Time
COSPA	CS	200K	2014 – 2019
TPG	CS	180K	2017 – 2019
MEDAN	M	150K	2016 – 2020



- What is the project budget per department?
  - Independ. of time: ⇒ (CS,380K), (M,150K)
  - At the current time (now): ⇒ (M,30K)
  - At each time point (sequenced):

Dept	Budget	Time
CS	100K	2014 – 2016
CS	280K	2017 – 2019
M	150K	2016 – 2020

- Timestamps must be adjusted for the result
- Some values must be scaled to the adjusted timestamps

Temporal Database Research History

- Four overlapping phases
  - 1956–1985: Concept development
  - 1978–1994: Design of query languages
  - 1988–present: Implementation aspects
    - Storage and index structures
    - Operator algorithms (join, aggregation)
    - First framework with comprehensive query support
  - 1993–present: Consolidation phase
    - Consensus glossary of temporal database concepts
    - Temporal features in SQL
- Still an active research area today
  - New application domains need new operations
  - e.g., moving objects, data streams, temporally evolving graphs, etc.

Temporal Features in SQL:2011

- Period specification for tables (application-time and system-time)

```
CREATE TABLE Emp (
    Name VARCHAR,
    Dept VARCHAR,
    Start DATE,
    End DATE,
    PERIOD FOR Period (Start, End) );
```

- Temporal UPDATE/DELETE behavior
- Temporal primary and foreign keys

```
ALTER TABLE Emp
ADD PRIMARY KEY (Name, Period WITHOUT OVERLAPS)
```

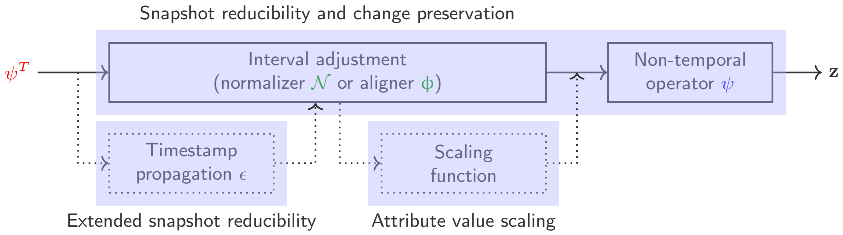
- Predicates/Functions for periods to support querying

- Support for storage and update
- Limited support for **query formulation**!

Outline

Temporal Alignment

- Reduce temporal operators ψ^T to nontemporal operators ψ
  - Adjust time periods of input relations
  - Apply non-temporal operator



- Minimal changes to DBMS: normalizer and aligner primitives
- Existing query optimization/indexing works

Example Temporal Aggregation – Normalizer

- What is the budget per department: ${}_{\mathrm{D}}\vartheta^{T}_{SUM(Budget)}(\mathbf{p})$

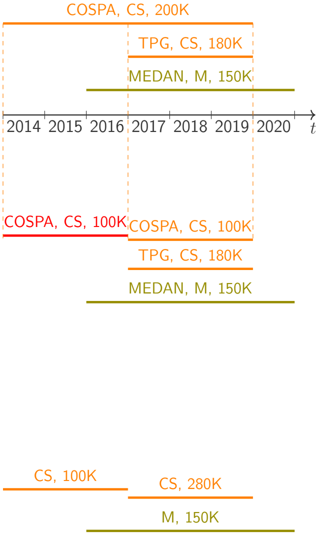
Project	Dept	Budget	Time
COSPA	CS	200K	2014 – 2019
TPG	CS	180K	2017 – 2019
MEDAN	M	150K	2016 – 2020

adjustment & scaling

Project	Dept	Budget	Time
COSPA	CS	100K	2014 – 2016
COSPA	CS	100K	2017 – 2019
TPG	CS	180K	2017 – 2019
MEDAN	M	150K	2016 – 2020

nontemp. aggregation by **Dept and Time**

Dept	Sum	Time
CS	100K	2014 – 2016
CS	280K	2017 – 2019
M	150K	2016 – 2020



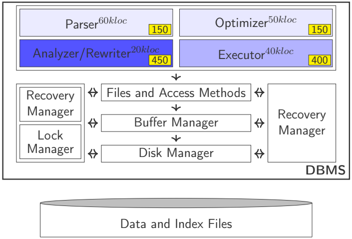
Reduction Rules: $\psi^T \longrightarrow \{\mathcal{N}, \phi\} + \psi$

Operator			Reduction
Selection	$\sigma^T_\theta(\mathbf{r})$	$=$	$\sigma_\theta(\mathbf{r})$
Projection	$\pi^T_{\mathbf{B}}(\mathbf{r})$	$=$	$\pi_{\mathbf{B},\mathrm{T}}(\mathcal{N}_{\mathbf{B}}(\mathbf{r},\mathbf{r}))$
Aggregation	$_{\mathbf{B}}\vartheta^T_F(\mathbf{r})$	$=$	$_{\mathbf{B},\mathrm{T}}\vartheta_F(\mathcal{N}_{\mathbf{B}}(\mathbf{r},\mathbf{r}))$
Difference	$\mathbf{r} -^T \mathbf{s}$	$=$	$\mathcal{N}_{\mathbf{A}}(\mathbf{r},\mathbf{s}) - \mathcal{N}_{\mathbf{A}}(\mathbf{s},\mathbf{r})$
Union	$\mathbf{r} \cup^T \mathbf{s}$	$=$	$\mathcal{N}_{\mathbf{A}}(\mathbf{r},\mathbf{s}) \cup \mathcal{N}_{\mathbf{A}}(\mathbf{s},\mathbf{r})$
Intersection	$\mathbf{r} \cap^T \mathbf{s}$	$=$	$\mathcal{N}_{\mathbf{A}}(\mathbf{r},\mathbf{s}) \cap \mathcal{N}_{\mathbf{A}}(\mathbf{s},\mathbf{r})$
Cart. Prod.	$\mathbf{r} \times^T \mathbf{s}$	$=$	$\alpha(\phi_\top(\mathbf{r},\mathbf{s}) \bowtie_{\mathbf{r}.T=\mathbf{s}.T} \phi_\top(\mathbf{s},\mathbf{r}))$
Inner Join	$\mathbf{r} \bowtie^T_\theta \mathbf{s}$	$=$	$\alpha(\phi_\theta(\mathbf{r},\mathbf{s}) \bowtie_{\theta \wedge \mathbf{r}.\mathrm{T}=\mathbf{s}.\mathrm{T}} \phi_\theta(\mathbf{s},\mathbf{r}))$
Left O. Join	$\mathbf{r}$ ⟕$^T_\theta$ $\mathbf{s}$	$=$	$\alpha(\phi_\theta(\mathbf{r},\mathbf{s})$ ⟕$_{\theta \wedge \mathbf{r}.\mathrm{T}=\mathbf{s}.\mathrm{T}}$ $\phi_\theta(\mathbf{s},\mathbf{r}))$
Right O. Join	$\mathbf{r}$ ⟖$^T_\theta$ $\mathbf{s}$	$=$	$\alpha(\phi_\theta(\mathbf{r},\mathbf{s})$ ⟖$_{\theta \wedge \mathbf{r}.\mathrm{T}=\mathbf{s}.\mathrm{T}}$ $\phi_\theta(\mathbf{s},\mathbf{r}))$
Full O. Join	$\mathbf{r}$ ⟗$^T_\theta$ $\mathbf{s}$	$=$	$\alpha(\phi_\theta(\mathbf{r},\mathbf{s})$ ⟗$_{\theta \wedge \mathbf{r}.\mathrm{T}=\mathbf{s}.\mathrm{T}}$ $\phi_\theta(\mathbf{s},\mathbf{r}))$
Anti Join	$\mathbf{r} \rhd^T_\theta \mathbf{s}$	$=$	$\phi_\theta(\mathbf{r},\mathbf{s}) \rhd_{\theta \wedge \mathbf{r}.\mathrm{T}=\mathbf{s}.\mathrm{T}} \phi_\theta(\mathbf{s},\mathbf{r})$

Temporal Op. = Primitive + Traditional Op.

Temporal PostgreSQL @ UNIBZ

- Integration in DBMS kernel (approx. 1000 LOC)
- Dignös et al.: *Extending the kernel of a relational DBMS with comprehensive support for sequenced temporal queries*. TODS, 2016.
- Submitted as patch to PostgreSQL
- For more information go to: http://tpg.inf.unibz.it



- Ongoing project: we seek PhD students and PostDocs to join!

Future Work

- Boosting performance
  - Customized alignment primitives to reduce intermediate relation size
  - Precise cost estimates based on temporal distribution
  - Equivalence rules for the interaction of primitives with RA
- Support for multiple time dimensions
  - e.g., valid time and transaction time (supported by SQL:2011)
- SQL extension to facilitate the formulation of temporal queries

Outline

unibz

1. Temporal Databases
2. Temporal Alignment – Comprehensive Query Support
3. Time Series
4. Conclusion

Time Series

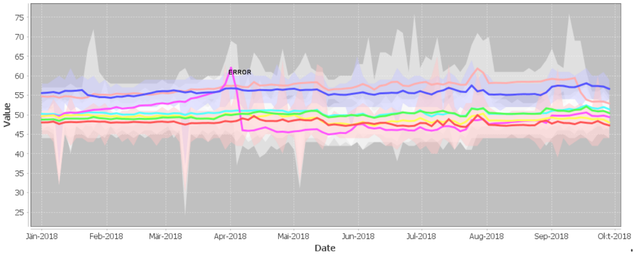
- A special type of temporal data: sequence of point values
- Analysis has to consider entire time series (not just single point)
- Mostly based on similarity: find most similar TS is a fundamental query

Temporal Databases vs. Time Series

	Operations	Technologies
Databases	Selection Projection Aggregation Join Difference Intersection Union	RA SQL RDBMS B-tree Hash
Time series	Alignment Missing value imputation Outlier detection Resampling Dimensionality reduction Aggregation Anomaly detection Classification Compression Correlation analysis Discord detection Forecasting Function approximation Motif discovery Prediction Predictive maintenance Seasonality analysis Segmentation Similarity search Subsequence search etc.	ED DTW PAA PLA DFT iSAX ADS SQL-TS TSMS ⋮

Project on Predictive Maintenance

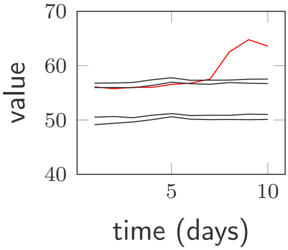
- **Goal:** Predict device errors or maintenance steps
- **Idea:** Spot patterns in sensor data with high prognostic accuracy



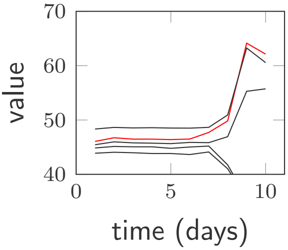
Preprocessing

- is essential for accurate/reliable analytics
- is work-intensive, ad-hoc, lack of methods to steer the process

Leads to an error



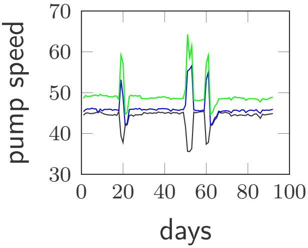
Does not lead to an error



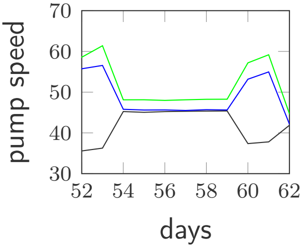
- Error pattern becomes only visible by looking at multivariate signals
- How to determine a good subset of signals?

Window Length

Signals with three errors

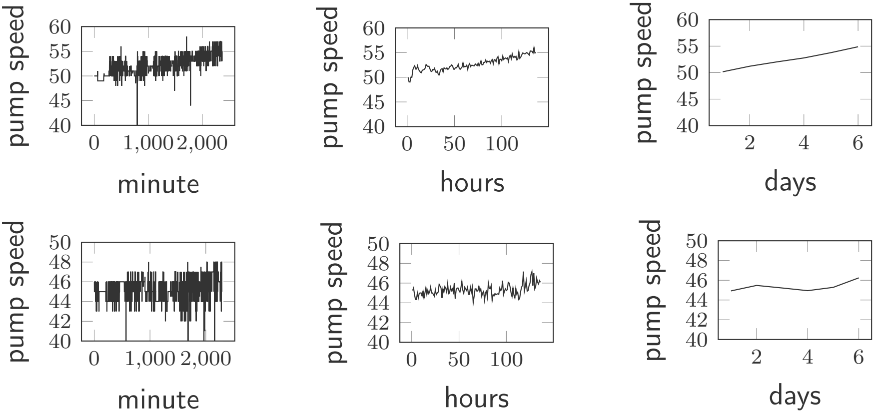


Window covers two errors



- Different window lengths might or might not detect an error
- How to determine a good window length?

Granularity/Representation



- Different granularities/representations emphasize different information
- How to determine a good granularity/representation?

Future Work

unibz

- Investigation of systematic preprocessing techniques and methods
  - Impact on downstream analysis
- Time series management system (TSMS)
  - Comprehensive support for time series management/analytics/mining
  - Integrate into RDBMs?

Outline

Conclusion

unibz

Temporal databases

- Temporal features in SQL:2011
- Query support is largely missing
- Temporal alignment offers comprehensive query support

Future work

- Performance/query optimization
- Multiple time dimensions
- SQL extension

Time series

- Special kind of temporal data, heavily based on similarity
- Operations studied in isolation
- Preprocessing not well studied

Future work

- Systematic investigation of preprocessing
- (Relational) Time series management system

Thank You!