Modeling & Management of Mobility Data

Yannis Theodoridis & Nikos Pelekis

InfoLab | University of Piraeus | Greece
infolab.cs.unipi.gr

Apr. 2013

"πάντα ρει - Everything changes and nothing remains still"
Heraclitus (500 BC)

Outline

- Acquiring trajectories from raw data
  - About mobility data
  - The trajectory reconstruction problem
- Location-aware querying
  - From straightforward to advanced mobility queries
- Efficient trajectory indexing and storage in MODs
  - Indexing techniques
  - MOD engines
- The next wave: “semantic” trajectories
- Summary

Acquiring trajectories from raw data

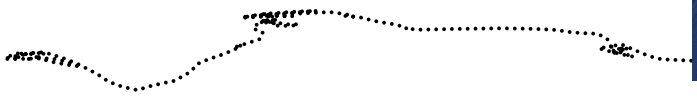
About mobility data

The trajectory reconstruction problem

GPS Data

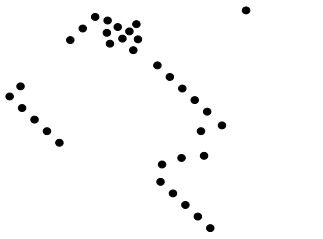
- Raw data: GPS recordings

```
objectID, trajectoryID, timestamp, longitude, latitude
201100024,1,2009-01-02 08:54:07,24.609728324369,38.013503319816
201100024,1,2009-01-02 08:54:25,24.6094016577037,38.0127699864845
201100024,1,2009-01-02 08:55:06,24.6086749910399,38.011116653155
201100024,1,2009-01-02 08:55:56,24.6076299910435,38.0092066531597
201100024,1,2009-01-02 08:56:16,24.6071983243782,38.0084733198281
201100034,1,2009-01-02 04:19:26,23.1092366579214,38.5853616531322
201100034,1,2009-01-02 04:19:36,22.927219990932
201100034,1,2009-01-02 04:19:45,23.035993324356
201100034,1,2009-01-02 04:19:55,22.935544990962
201100034,1,2009-01-02 04:20:05,23.063861657875
...
```

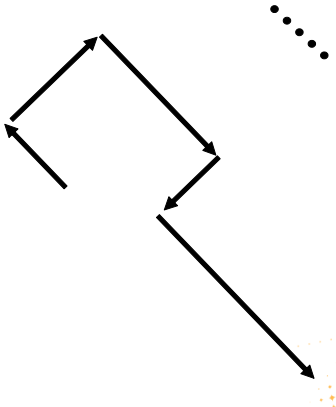


The trajectory reconstruction problem

- From raw data, i.e., time-stamped locations
  - Raw data (3D points) arrive either one-by-one or in bulks

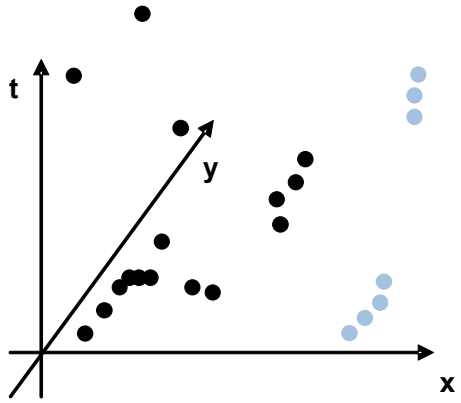


- … to trajectories, i.e., continuous evolutions
  - Redundancy is reduced, noise is removed, etc.



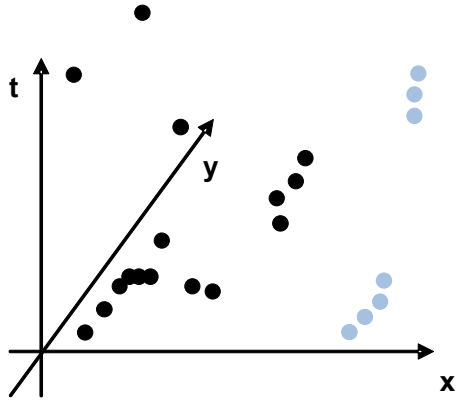
Reconstructing trajectories

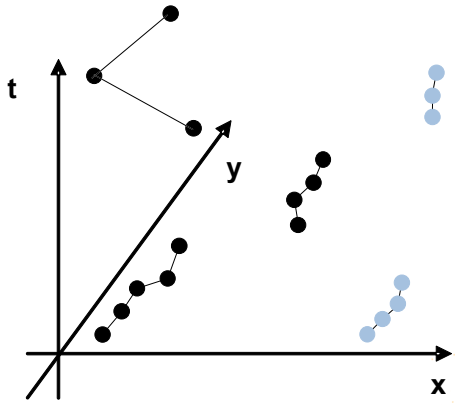
- Collected raw data represent time-stamped geo-locations
- Raw data (3D points) arrive either one-by-one or in bulks
- **Any idea?**



Reconstructing trajectories

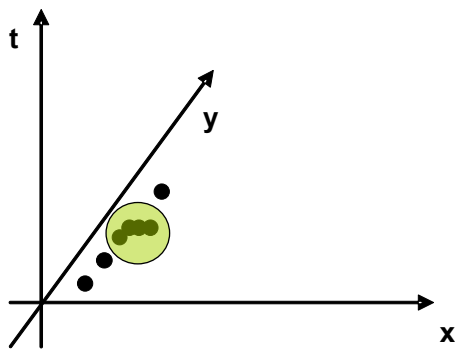
- [Marketos et al. 2008] filters / thresholds that decide whether the new series of data is to
  - be appended to an existing trajectory, or
  - initiate a new trajectory, or
  - be considered as noise

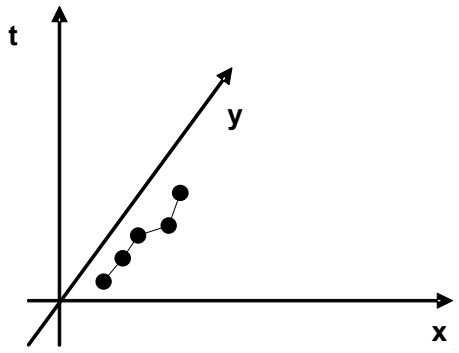




Reconstructing trajectories

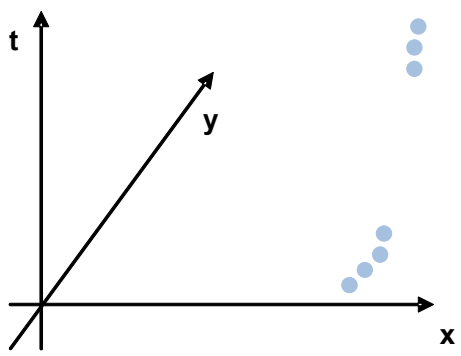
- 1st parameter: tolerance distance
  - The tolerance of the transmitted time-stamped positions
  - In other words: the **maximum distance between two consecutive time-stamped positions** of the same object in order for the object to be considered as **stationary**

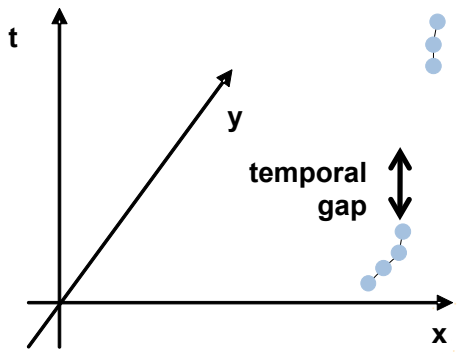




Reconstructing trajectories

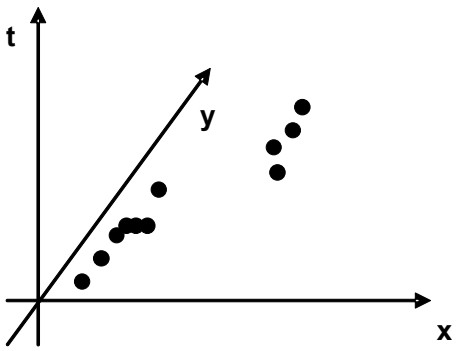
- tolerance distance
- 2nd parameter: temporal gap between trajectories
  - The **maximum allowed time interval** between two consecutive time-stamped positions of the same trajectory for a single moving object

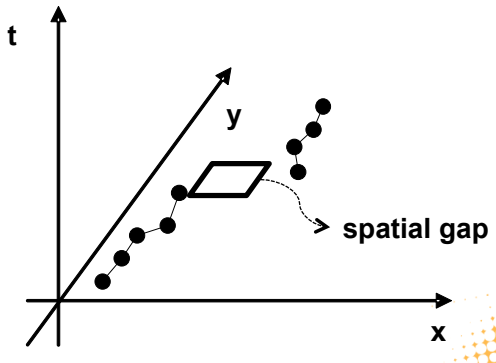




Reconstructing trajectories

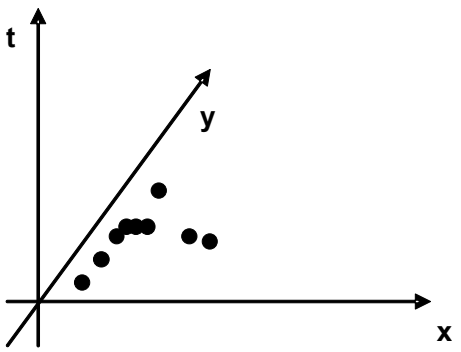
- tolerance distance, temporal gap
- 3rd parameter: spatial gap between trajectories
  - The **maximum allowed distance** in 2D plane between two consecutive time-stamped positions of the same trajectory

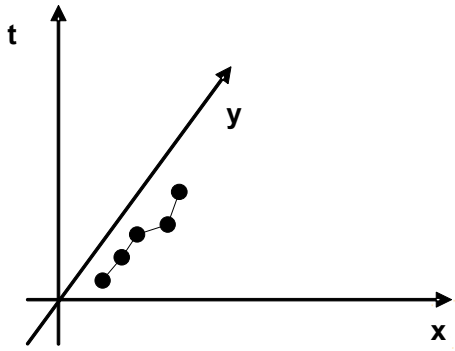




Reconstructing trajectories

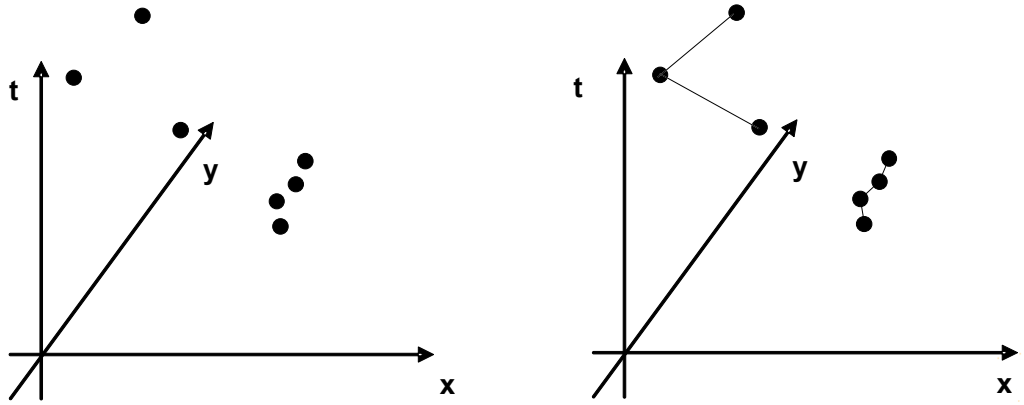
- tolerance distance, temporal gap, spatial gap
- 4th parameter: maximum speed
  - Decides whether a reported time-stamped location is **noise**, hence to be discarded from the output trajectory





Reconstructing trajectories

- tolerance distance, temporal gap, spatial gap, maximum speed
- 5th parameter: maximum noise duration
  - The **maximum duration of a noisy part** of a trajectory. If 'noise' continues longer than $noise_{max}$, most probably it is not noise but, instead, the start of a new trajectory!

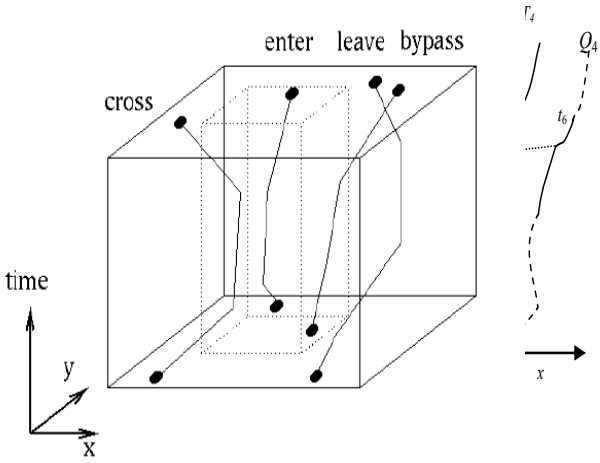


Location-aware querying

From straightforward to advanced mobility queries

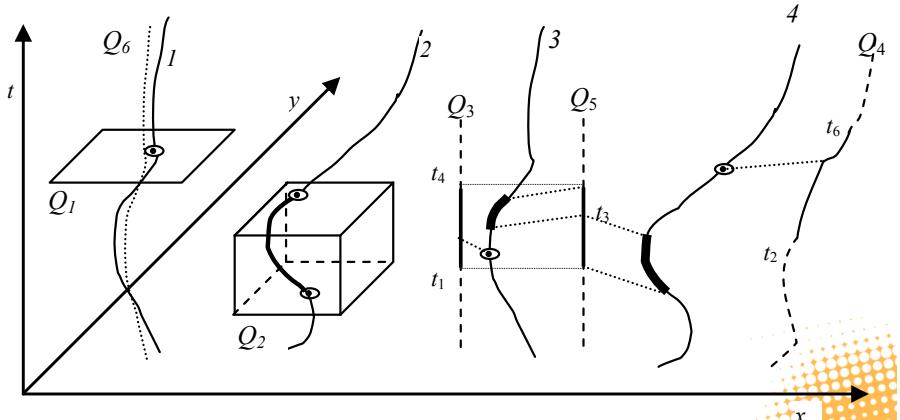
What kind of queries?

- The nature of trajectory data provides us with the ability to query them with a variety of operators.
- Straightforward queries:
  - Coordinate-based
    - Range, NN
  - Trajectory-based
    - Topological, Directional



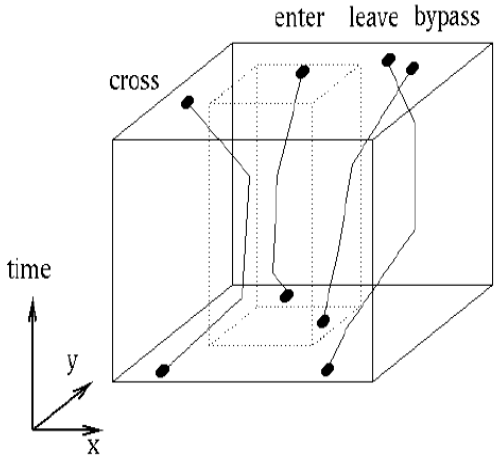
Coordinate-based queries

- Spatial (range or NN) search
  - "Find all trajectories that were **inside** area A at time instant t (or time interval I)" or
  - "Find the trajectory that was **closest** to point B at time instant t (or time interval I)"



Trajectory-based queries

- Topological queries: Search for trajectories that
  - **entered** (**crossed**, **left**, **bypassed**, etc.) a reference area, or
  - **intersected** (**moved in parallel**, etc.) a reference trajectory
- Directional queries: Search for trajectories that were located
  - **west of** (**south of**, etc.) a reference area, or
  - **left of** (**right of**, **in front of**, etc.) a reference trajectory

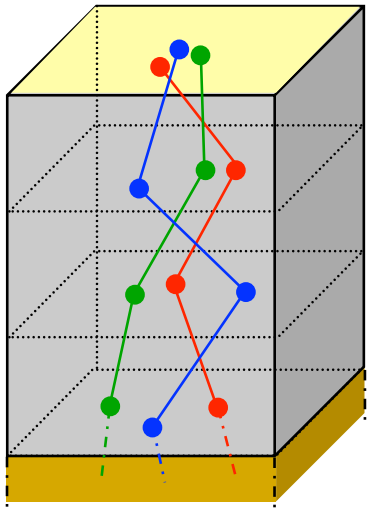


Taxonomy of location-aware queries

- Dimensions:
  - **Type**: range, NN, reverse NN, closest-point, …
  - **Time**: past, present, future
  - **Duration**: snapshot, continuous
  - **Query (reference) object**: stationary, moving
  - **Data objects**: stationary, moving
- Queries: several combinations of the above dimensions

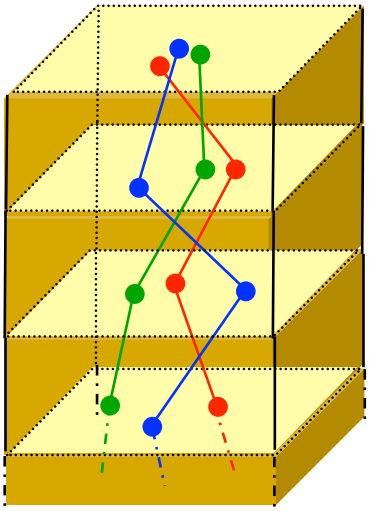
Querying the Past

- Examples:
  - Querying along the Temporal Dim.: What was the location of a certain object from 7:00 AM to 10:00 AM yesterday?
  - Querying Along the Spatial Dim.: Find all objects that were in a certain area at 7:00 AM yesterday
  - Querying Along the Spatio-temporal Dim.: Find all objects that were close to each other from 7:00 AM to 8:00 AM yesterday
- Features:
  - Large number of historical trajectories
  - Persistent read-only data



Querying the Present

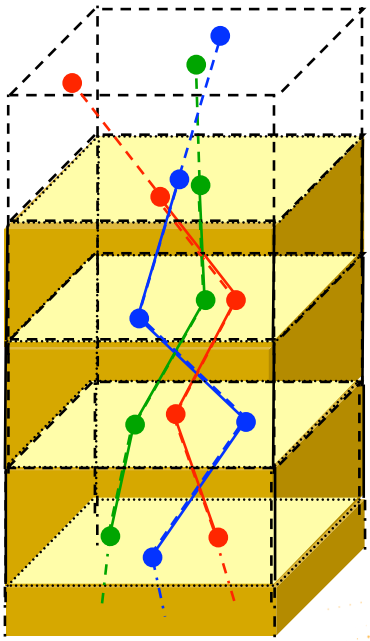
- Time is always NOW
- Example Queries:
  - Find the number of objects in a certain area
  - What is the current location of a certain object?
- Features:
  - Continuously changing data
  - Real-time query support is required
  - Index structures should be update-tolerant



- Present data is always accessed through continuous queries

Querying the Future

- Examples:
  - What will my nearest restaurant be after 30 minutes?
  - Does my path conflict with any other cars for the next hour?
- Features:
  - Predict the movement through a velocity vector
  - Prediction could be valid for only a limited time horizon in the future



Examples

slides from [Mokbel & Aref, 2007]

Where are my nearest McDonalds for the next hour?

- *Type:* Nearest-Neighbor query
- *Time:* Future
- *Duration:* Continuous
- *Query:* Moving
- *Object:* Stationary

Send E-coupons to all cars that I am their nearest gas station

- *Type:* Reverse NN query
- *Time:* Present
- *Duration:* Snapshot
- *Query:* Stationary
- *Object:* Moving

Examples
slides from [Mokbel & Aref, 2007]

Continuously report the number of cars in the freeway

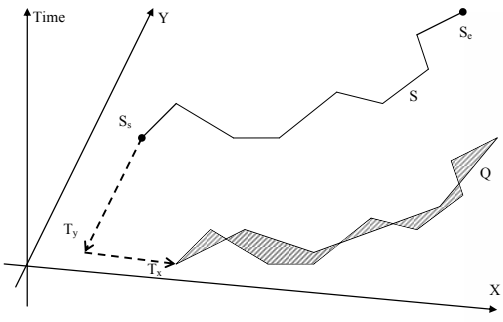
- *Type:* Range query
- *Time:* Present
- *Duration:* Continuous

What was the closest dist. between Taxi A & me yesterday?

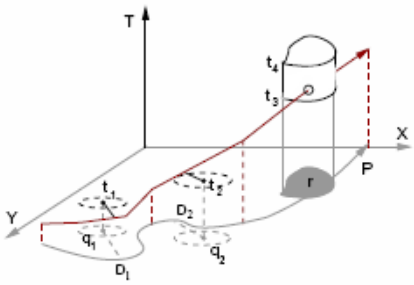
- *Type:* Closest-point query
- *Time:* Past
- *Duration:* Snapshot
- *Query:* Moving
- *Object:* Moving

… as well as more advanced queries

- **Trajectory similarity** queries
  - [Frentzos et al. 2007] "Given a query trajectory *Q*, find the k- most similar trajectories to *Q* (perhaps, constrained is space and/or time)"

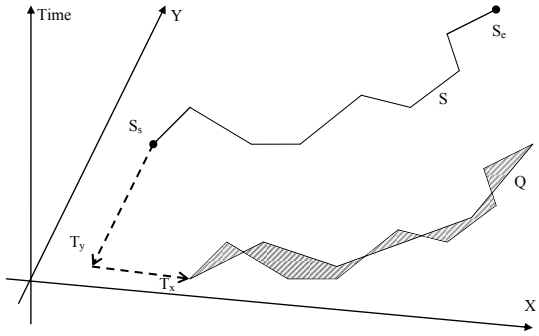


- **Motion pattern** queries
  - [Hadjieleftheriou et al. 2005] e.g. "Find objects that crossed through region *A* at time t_1, came as close as possible to point *B* at a later time t_2 and then stopped inside circle *C* during interval (t_3, t_4)"



Trajectory Similarity Queries

- Key question:
  - How do we measure **distance** or **(dis-)similarity** between two trajectories?



- Similarity variations:
  - [Pelekis et al. 2007, 2011] Similarity in space and/or time, or w.r.t. derived information (e.g. speed and direction)
- Similarity queries have been studied extensively in time-series literature
  - But, things are different here! Both where and when are important

Trajectory Similarity Queries (cont.)

- Different points of view: **Moving clusters**, **Flock queries**
  - What is a flock?
    - a large enough subset of objects moving along paths close to each other for a certain time
  - In the flock, identify **leaders** and **followers**

- Solutions:
  - [Benkert et al. 2008], [Gudmundsson & van Kreveld, 2006]

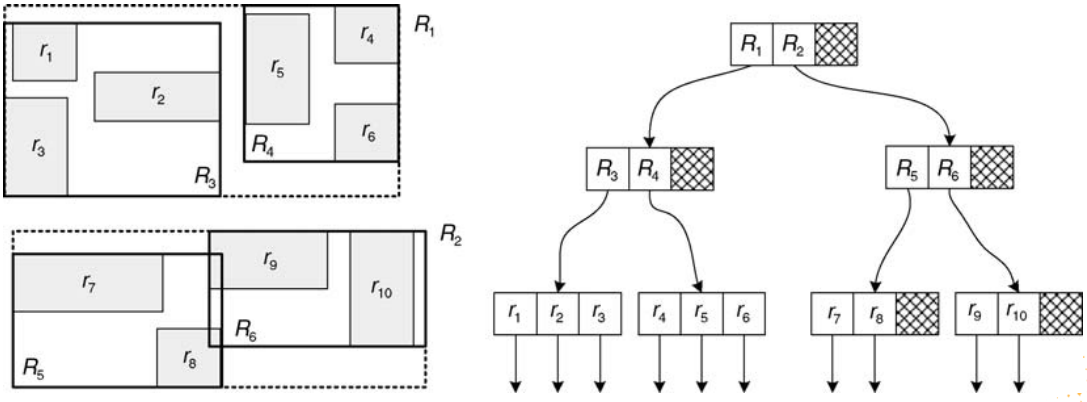
Efficient trajectory indexing and storage in MODs

Indexing techniques

MOD engines

R-trees for spatial data

- For d-dimensional point or region data
- Is it portable to mobility data?
  - In other words, is space + time simply a 3D space?



What ?? for mobility data

- Challenges:
  - Both space and time are equally important!
  - But! time is not simply a 3rd dimension
    - think of its monotonicity
    - Also, density in the 'spatio-temporal' space could be quite different from place to place

- Diversity in space, in time, in space-time …

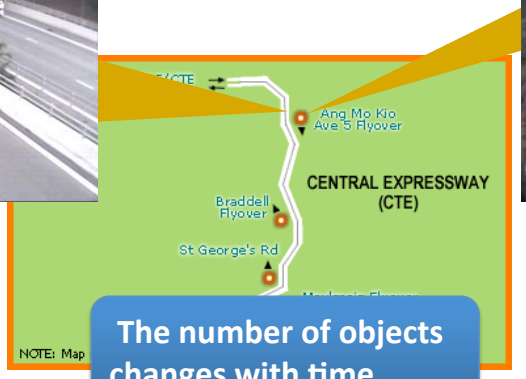
Diversity in space

Time: 05/06/2008 05:03 pm

The number of objects varies in different regions

Diversity in time

Time: 05/06/2008 05:03 pm

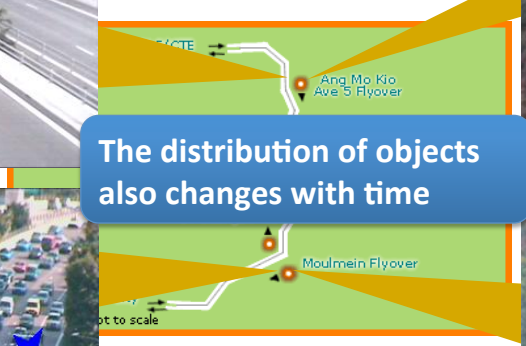


Time: 05/06/2008 08:05 am

The number of objects changes with time.

Diversity in space & time

Time: 05/06/2008 05:03 pm

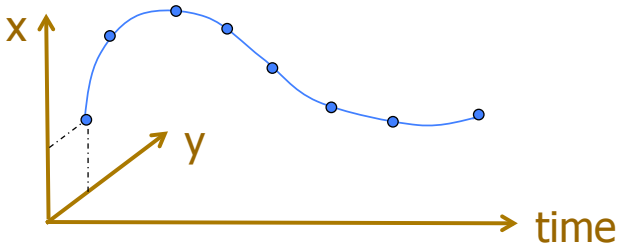


Time: 05/06/2008 08:05 am

The distribution of objects also changes with time

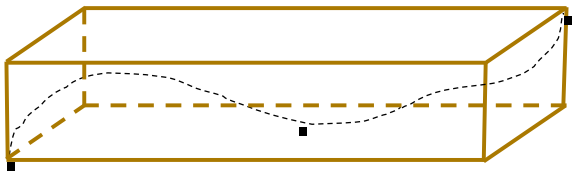
Back to technical stuff …

- Assume we have stored trajectories (i.e., the recorded locations of a moving object over time)
- Main Question: how can we approximate a trajectory?
  - Like, e.g. MBRs for (static) spatial objects?
  - Then, we could build indices upon the approximations

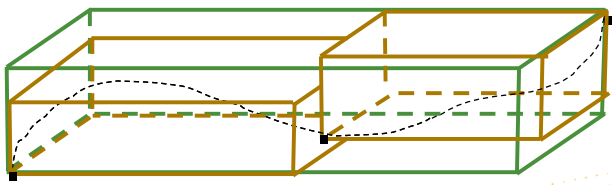


Two approaches: native vs. parametric space

- Typically approximate using MBRs; then index these MBRs
  - we can use R-trees etc. ☺
  - trajectories are lines, thus MBRs add extensive empty space ☹



- How many MBRs per trajectory?
  - One MBR per trajectory (too much empty space…) or one MBR per segment (too many MBRs…)
- Can we do something better?
  - Smart "partitioning" for MBRs [Hadjieleftheriou et al. 2002]



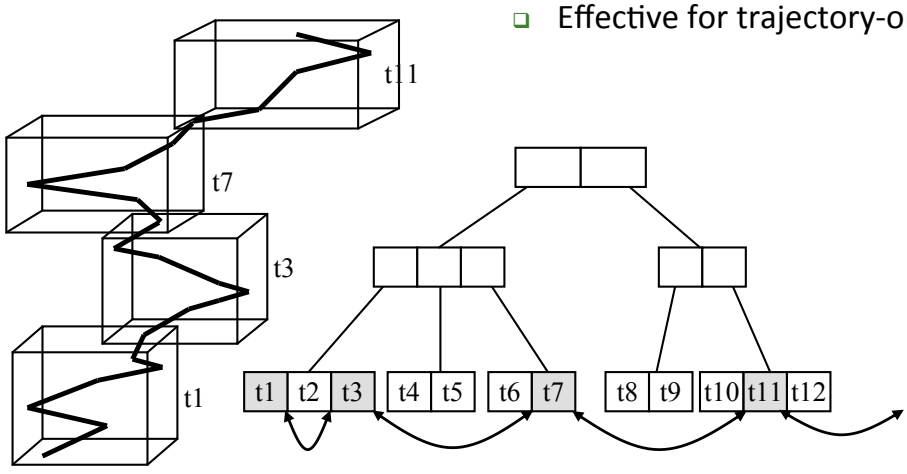
MOD indexing techniques

- Indexing the past (= trajectories)
  - unconstrained movement : **TB-tree** [Pfoser et al. 2000]
  - network-constrained movement: **FNR-tree** [Frentzos, 2003]
- Indexing the present (and anticipated future)
  - Data partitioning: **TPR-tree** [Saltenis et al. 2000], **TPR*-tree** [Tao et al. 2003]
  - Space partitioning: **B^x-tree** [Jensen et al. 2004], **ST^2B-tree** [Chen et al. 2008]
- (Hybrid solution for) Indexing the past & present
  - **R^{PPF}-tree** [Pelanis et al. 2006]

- We focus on historical tracks of moving objects (trajectories) ➔ concentrate on the 1st group

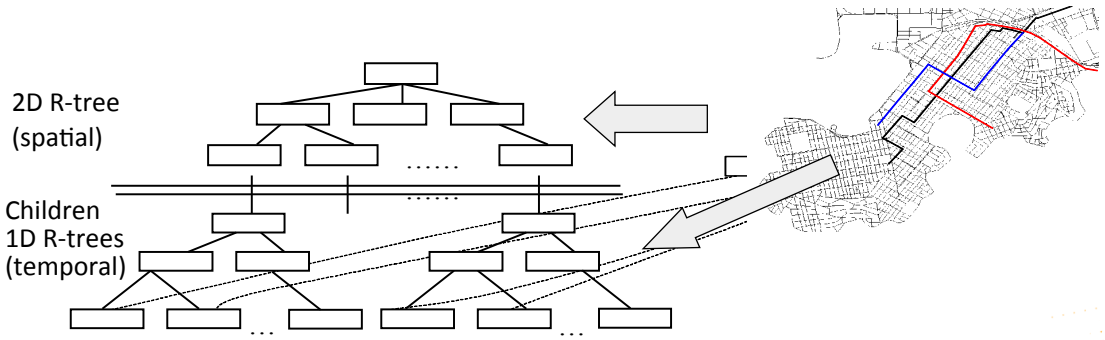
TB-tree

- [Pfoser et al. 2000] Maintains the 'trajectory' concept
  - Each node consists of segments of a single trajectory
  - nodes corresponding to the same trajectory are linked together in a chain
  - Effective for trajectory-oriented queries



FNR-tree

- (Frentzos, 2003) a forest of 1D (temporal) R-trees on top of a 2D (spatial) R-tree
  - There is an additional "Parent" 1D R-tree which indexes the temporal intervals of the 1D R-trees leaf nodes



Moving Objects Database Systems

- From traditional DBMS to …
  … **Moving Object Database (MOD) engines**
  - Objective: spatial and temporal dimensions to be considered as first-class citizens.
  - Technically: data types, indices, query processing & optimization strategies for trajectories of moving objects

- Current MOD solutions:
  - State-of-the-art MODs (Secondo, Hermes, etc.) vs.
  - SDBMS simulated to work as MODs (e.g. PostGIS)

Moving Objects Database Systems

- A few prototype MOD engines
  - **SECONDO** by Güting et. al. @ Hagen
  - **PLACE** by Aref et al. @ Purdue
  - **HERMES** by Pelekis et. al. @ Piraeus

- A quick note on PostGIS solution:
  - **PostGIS 2.0** new features include 3D/4D indexing
  - A trajectory can be simulated by a 3D path (= sequence of 3D points)

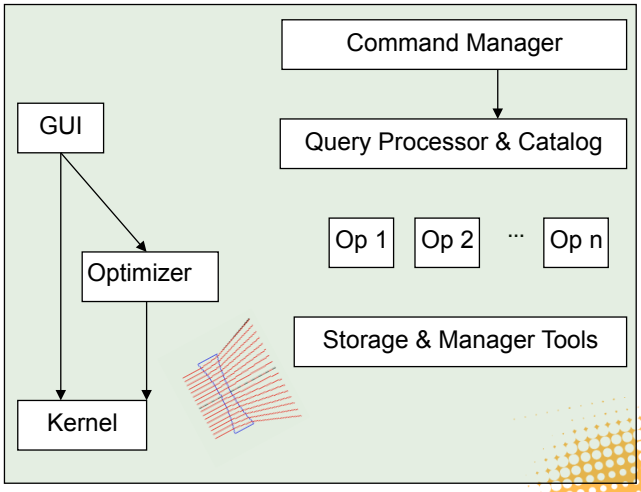
SECONDO

http://dna.fernuni-hagen.de/Secondo.html/index.html

- A generic DBMS framework that can be filled with implementation of various data models (R, OR, XML) and data types (spatial data, moving objects)
- Built on top of Berkeley DB.
- A MOD is a set of SECONDO objects of the form *(name, type, value)*, where *type* is one of the implemented algebras
- About 20 implemented algebras
  - standard algebra, relational algebra, R-Tree algebra, spatial algebra, etc.
- Query optimizer includes optimization of conjunctive queries, selectivity estimation, and implementation of an SQL-like query language

SECONDO Architecture

- Generic GUI independent of data models
  - The interface includes command prompt and is extensible by a set of different viewers
  - The core functionality is the optimization of conjunctive queries, i.e., producing an efficient query plan
  - On top of the query optimizer, there is a SQL-like language in a PROLOG notation

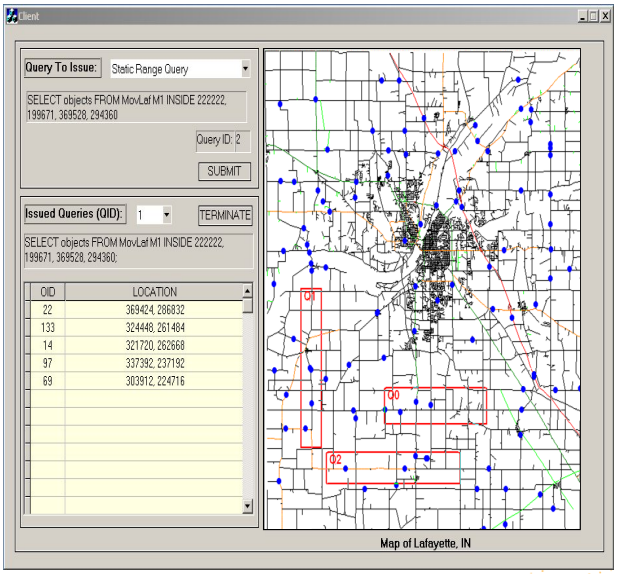


PLACE

Pervasive Location Aware Computing Environments

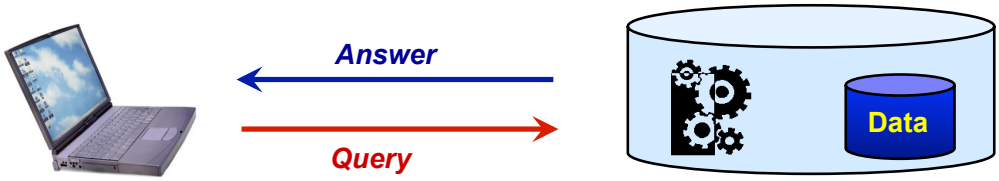
http://www.cs.purdue.edu/place/

- **Continuous** evaluation of queries over spatio-temporal **data streams**
- Shared execution among concurrent continuous queries
- Built on top of PREDATOR database system
- Incremental evaluation of continuous queries
- Spatio-temporal query operators

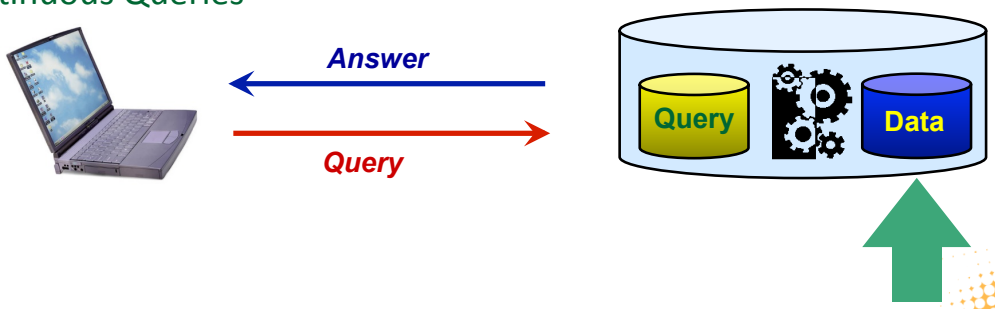


Snapshot vs. Continuous Query Processing

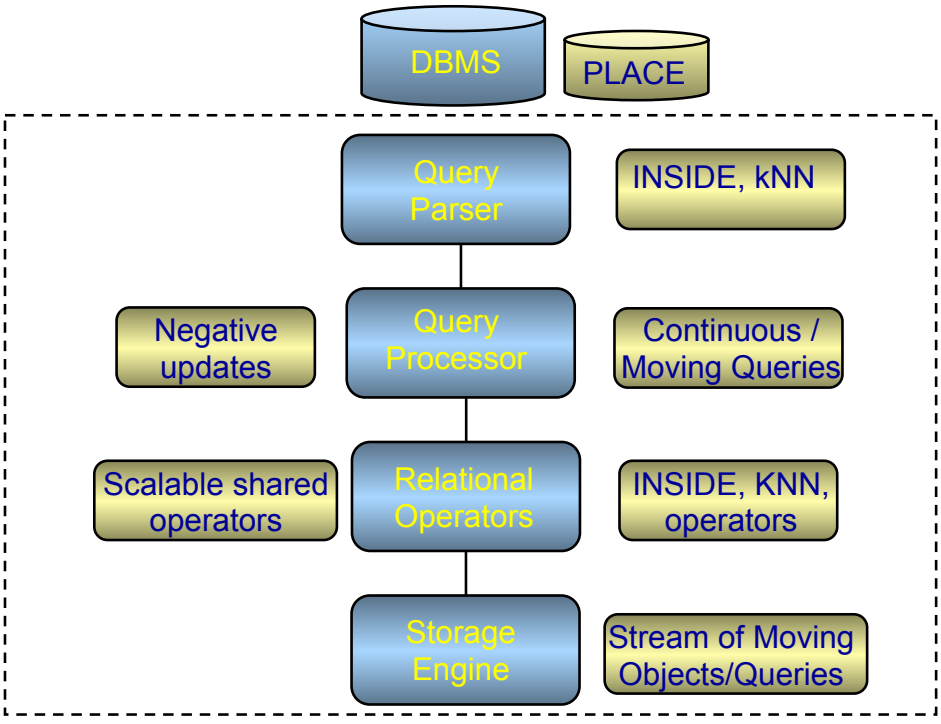
- Traditional (Snapshot) Queries



- Continuous Queries



PLACE Architecture



Extended SQL Syntax in PLACE

```
SELECT select_clause
FROM from_clause
WHERE where_clause
INSIDE inside_clause
kNN knn_clause
WINDOW window_clause
```

- Stationary query: (x_1, y_1, x_2, y_2)
- Moving query: ('M', OID, width, length)

- Stationary query: (k, x, y)
- Moving query: ('M', OID, k)

The Hermes MOD engine

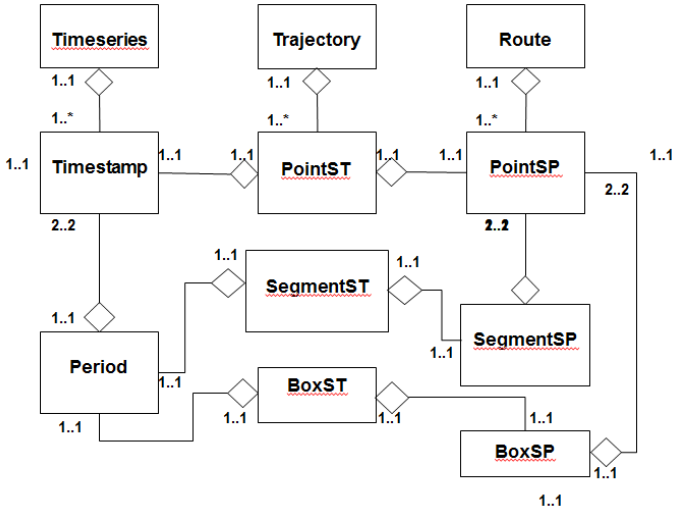
- A palette of data types
  - … methods and indexes
- … on top of an extensible DBMS
  - Oracle Spatial (1)
  - PostgreSQL (2)

(1) URL: http://infolab.cs.unipi.gr/hermes/

(2) URL: http://hermes-mod.java.net/

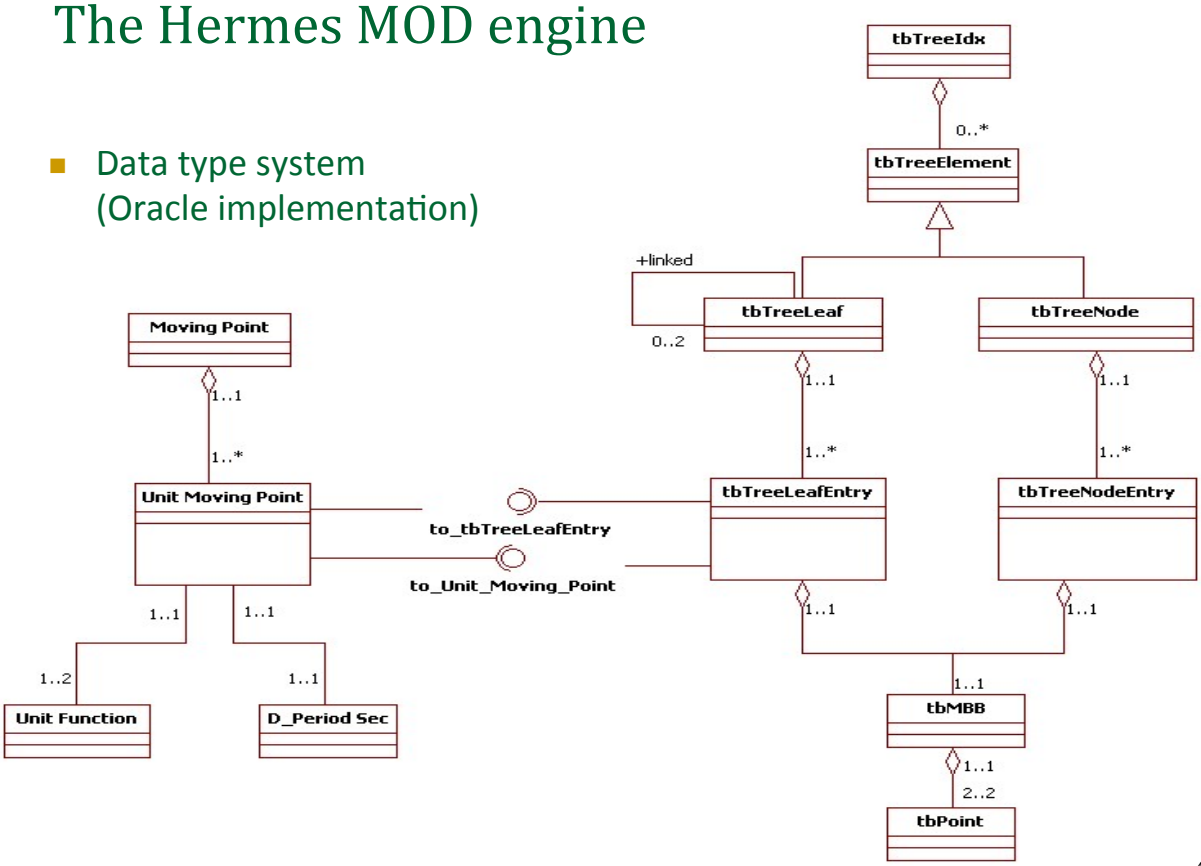
The Hermes MOD engine

- Data type system (Postgres implementation)
  - **Temporal** datatypes
    - Timeseries; Timestamp; Period
  - **Spatial** datatypes
    - Route; PointSP; SegmentSP; BoxSP
  - **Spatio-temporal** datatypes
    - Trajectory; PointST; SegmentST; BoxST



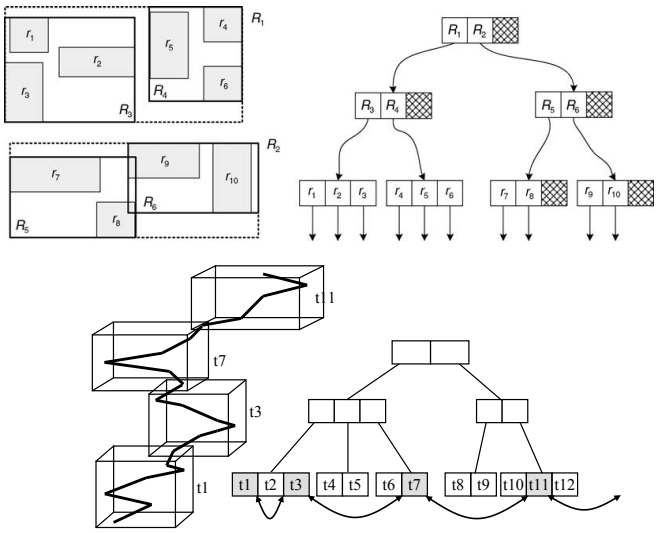
The Hermes MOD engine

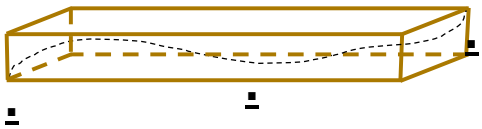
- Data type system (Oracle implementation)



The Hermes MOD engine

- Indexing support
  - **3D R-trees** (in both implementations)
  - **TB-trees** (in Oracle implementation only)
  - options for indexing a trajectory (both are supported by Hermes):
    1. a single MBB for the entire trajectory
    2. one MBB per trajectory segment





The Hermes MOD engine

- Methods supported
  - Find the (1D) **timestamp** (**temporal period**) of a trajectory …
    - … when located at a given spatial point (within a given area, resp.)
  - Find the (2D) **location** (**route**) of a trajectory …
    - … at a given timestamp (during a given temporal period, resp.)
  - Find the (3D) **sub-trajectory** of a trajectory …
    - … spanning during a given temporal period (or within a given spatial area)
  - Find the **speed** (or **acceleration**) of a trajectory …
    - … at a given timestamp

The Hermes MOD engine

- Methods supported (cont.)
  - Find the **enter-leave points** of a trajectory …
    - … w.r.t. a given spatial area
  - Apply **spatio-temporal compression** on a trajectory
    - … w.r.t. state-of-the-art trajectory compression algorithm (TD-TR)
  - Find the **Potential Area of Activity** (PAA) of a trajectory …
    - … between (sequences of) sampled locations
  - etc. etc. methods

The Hermes MOD engine

- Queries supported
  - **Timeslice** queries
  - **Range** queries, 3 variations:
    - spatial only vs. temporal only vs. spatio-temporal range window
  - **Nearest Neighbor** (NN) queries, 2 variations:
    - w.r.t. a reference location (spatial point or area)
    - w.r.t. a reference trajectory
  - Pair-wise **trajectory similarity** queries
    - … w.r.t. popular and state-of-the-art trajectory similarity functions:
      - Manhattan (L_1), Euclidean (L_2), Chebyshev (L_∞), …
      - LCSS, DTW, ERP, EDR, DISSIM, …

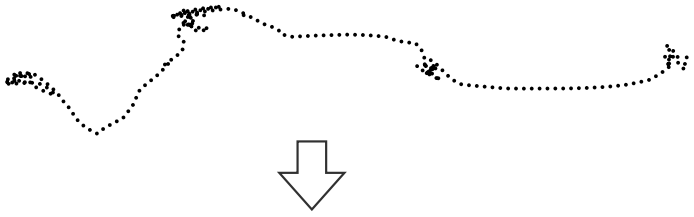
The Hermes MOD engine

- Queries supported (cont.)
  - **Trajectory cluster analysis** queries:
    - … w.r.t. state-of-the-art trajectory clustering algorithms (T-Optics, S2T-Clustering
  - **OD-matrix** queries, 2 variations:
    - origin/destination pairs be spatial or spatio-temporal ranges

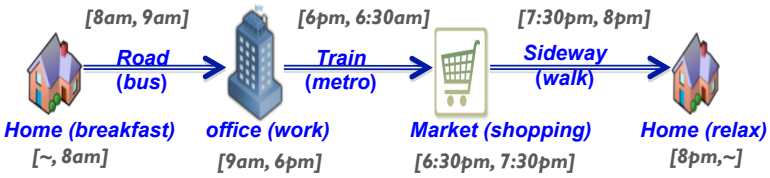
The next wave: semantic trajectories

From raw to "semantic" trajectories

raw mobility data
sequence (x,y,t) points
e.g., GPS feeds



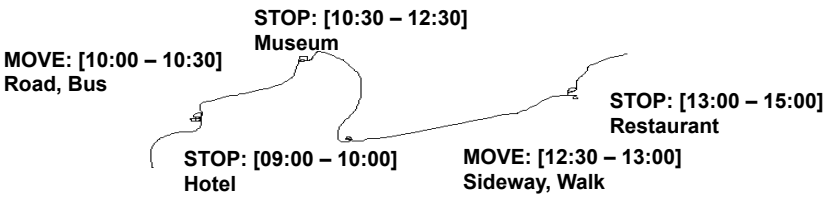
meaningful mobility tuples
<place, $time_{in}$, $time_{out}$, tags>



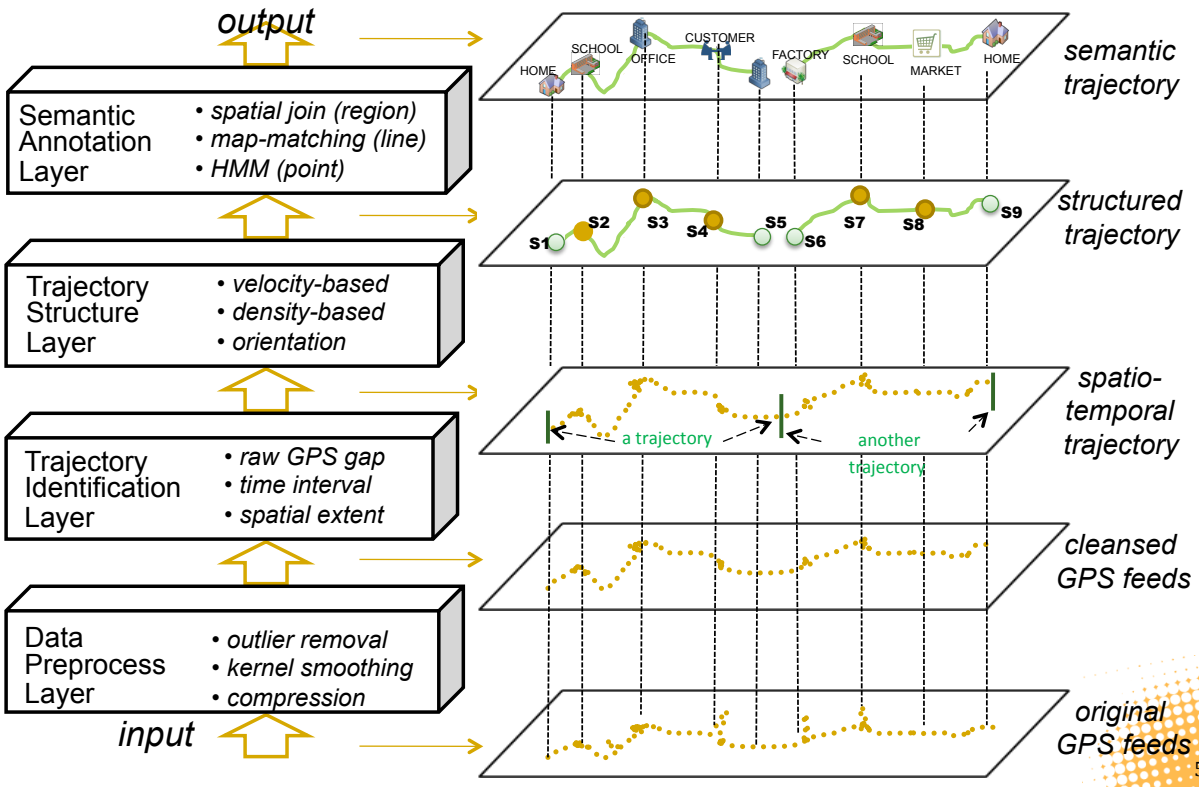
- **Semantic Trajectory: $T=\{e_{first},\ldots,e_{last}\}$**
- **Episode: e_i = (STOP | MOVE, t_{from}, t_{to}, *place, tag*)**

From raw to "semantic" trajectories

- Detection of homogenous fractions of movement,
  - Trajectory is reconstructed as a **sequence of episodes (stops/moves)**
- **Stops** are places (points, regions) where the object stays "static"
- **Moves** are the parts of the object's trajectory in between two Stops, i.e. where the object is "moving"
- **Tags** are meta-data associated with Stops and Moves
  - information about (at least…) **when? where?** (also…) **how? what? why?**



Offline construction of semantic trajectories
(Yan et al. ESWC '10, EDBT '11)



Summary

Summary on Mobility Data Management

- From (stationary) spatial to moving object databases
- Advances in …
  - Modeling and representation (at logical database level)
    - Moving point, trajectory etc. data types
  - Indexing and query processing (at physical database level)
    - 3D R-trees, TB-trees, etc.
    - Coordinate- vs. trajectory- based queries
- A few prototype MOD engines are already out there …
  - state-of-the-art: SECONDO and HERMES (offline mode), PLACE (streaming mode)

Questions

Reading list

Mobility data modeling & MOD engines

- de Almeida, V.T. et al. (2006) Querying Moving Objects in SECONDO. Proceedings of MDM.
- Behr, T. and Güting, R.H. (2005) Fuzzy Spatial Objects: An Algebra Implementation in SECONDO. Proceedings of ICDE.
- Cao, H. and Wolfson, O. (2005) Nonmaterialized Motion Information in Transport Networks. Proceedings of ICDT.
- Chen, C.X. and Zaniolo, C. (2000) SQLST: A Spatio-Temporal Data Model and Query Language. Proceedings of ER.
- Cheng, R. et al. (2004) Efficient Indexing Methods for Probabilistic Threshold Queries over Uncertain Data. Proceedings of VLDB.
- Dieker, S. and Güting, R.H. (2000) Plug and Play with Query Algebras: SECONDO – A Generic DBMS Development Environment. Proceedings of IDEAS.

Mobility data modeling & MOD engines (cont.)

- Güting, R.H. et al. (2000) A Foundation for Representing and Querying Moving Objects. ACM Transactions on Database Systems, 25(1):1-42.
- Güting, R.H. et al. (2006) Modeling and querying moving objects in networks. VLDB Journal, 15(2): 165-190.
- Karimi, H. and Liu, X. (2003) A Predictive Location Model for Location-Based Services. Proceedings of ACM-GIS.
- Marketos, G. et al. (2008) Building real-world trajectory warehouses. Proceedings of MobiDE.
- Mokbel, M.F. et al. (2004a) Continuous Query Processing of Spatio-temporal Data Streams in PLACE. Proceedings of SSDBM.
- Mokbel, M.F. et al. (2004b) PLACE: A Query Processor for Handling Real-time Spatio-temporal Data Streams. Proceedings of VLDB.

Mobility data modeling & MOD engines (cont.)

- Mokhtar, H., and Su, J. (2005) A Query Language for Moving Object Trajectories. Proceedings of SSDBM.
- Patroumpas, K. and Sellis, T.K. (2004) Managing Trajectories of Moving Objects as Data Streams. Proceedings of STDBM.
- Pelekis, N. and Theodoridis, Y. (2007) An Oracle Data Cartridge for Moving Objects. Technical Report, TR-2007-04, University of Piraeus.
- Pelekis, N. et al. (2004) Literature Review of Spatio-temporal Database Models. Knowledge Engineering Review, 19(3): 235-274.
- Pelekis, N. et al. (2006) Hermes - A Framework for Location-Based Data Management. Proceedings of EDBT.
- Pelekis, N. et al. (2008) HERMES: aggregative LBS via a trajectory DB engine. Proceedings of ACM SIGMOD.
- Pelekis, N. et al. (2011) HERMES: A Trajectory DB Engine for Mobility-Centric Applications, International Journal of Knowledge-based Organizations, in press..

Semantic Trajectories

- Baglioni, M. et al. (2009). Towards Semantic Interpretation of Movement Behavior. Proceedings of AGILE .
- Spaccapietra, S. et al. (2008). A Conceptual View On Trajectories. Data and Knowledge Engineering, 65(1).
- Spinsanti, L. et al. (2010). Where you stop is who you are: understanding peoples' activities. Proceedings of BMI.
- Xie, K. et al. (2009). From trajectories to activities: a spatio-temporal join approach. Proceedings of GIS-LBSN.
- Yan, Z. et al. (2010). A Hybrid Model and Computing Platform for Spatio-Semantic Trajectories. Proceedings of ESWC
- Yan, Z. et al. (2011). SeMiTri: A Framework for Semantic Annotation of Heterogeneous Trajectories. Proceedings of EDBT.
- Yan, Z. et al. (2011). SeTraStream: Semantic-aware Trajectory Construction over Streaming Movement Data, Proceedings of SSTD.
- Zheng, Y.et al. (2010). Understanding transportation modes based on GPS data for Web applications, ACM Transaction on the Web, 4(1)

Mobility data generators & benchmarks

- Brinkhoff, T. (2002) A Framework for Generating Network-based Moving Objects. GeoInformatica, 6(2): 153-180.
- Düntgen, C. et al. (2009) BerlinMOD: a Benchmark for Moving Object Databases. The VLDB Journal, 18: 1335-1368.
- Theodoridis, Y. (2003) Ten Benchmark Database Queries for Location-based Services. The Computer Journal, 46(6): 713-725
- Theodoridis, Y. and M. Nascimento (2000) Generating Spatiotemporal Datasets on the WWW. SIGMOD Record, 29(3): 39-43.

MOD query processing

- Benetis, R. et al. (2002) Nearest Neighbor and Reverse Nearest Neighbor Queries for Moving Objects. Proceedings of IDEAS.
- Benkert, M. et al. (2008) Reporting Flock Patterns. Computational Geometry, 41: 111-125.
- Frentzos, E. et al. (2005) Nearest Neighbor Search on Moving Object Trajectories. Proceedings of SSTD.
- Frentzos, E. et al. (2007) Index-based Most Similar Trajectory Search. Proceedings of ICDE.
- Gedik, B., and Liu, L. (2004) MobiEyes: Distributed Processing of Continuously Moving Queries on Moving Objects in a Mobile System. Proceedings of EDBT.
- Gudmundsson, J. and M. van Kreveld (2006) Computing longest duration flocks in trajectory data. Proceedings of ACM-GIS.
- Hu, H. et al. (2005) A Generic Framework for Monitoring Continuous Spatial Queries over Moving Objects. Proceedings of ACM SIGMOD.

MOD query processing (cont.)

- Jensen, C.S. et al. (2003) Nearest Neighbor Queries in Road Networks. Proceedings of ACM-GIS.
- Lema, J.A.C. et al. (2003) Algorithms for Moving Objects Databases. The Computer Journal, 46(6): 680-712.
- Li, F. et al. (2005) On Trip Planning Queries in Spatial Databases. Proceedings of SSTD.
- Mokbel, M.F. et al. (2004) SINA: Scalable Incremental Processing of Continuous Queries in Spatio-temporal Databases. Proceedings of ACM SIGMOD.
- Mokbel, M.F. and W.G. Aref (2007) Location-aware Query Processing and Optimization: A Tutorial. Proceedings of MDM.
- Papadias, D. et al. (2003) Query Processing in Spatial Network Databases. Proceedings of VLDB.
- Pelekis, N. et al. (2007) Similarity Search in Trajectory Databases. Proceedings of TIME.
- Pelekis, N. et al. (2011), Visually Exploring Movement Data via Similarity-based Analysis, Journal of Intelligent Information Systems.

MOD query processing (cont.)

- Pfoser, D. and C.S. Jensen (2001) Querying the Trajectories of On-line Mobile Objects. Proceedings of MobiDE.
- Porkaew, K. et al. (2001) Querying Mobile Objects in Spatio-Temporal Databases. Proceedings of SSTD.
- Shekhar, S. and Yoo, J. S. (2003) Processing In-Route Nearest Neighbor Queries: A Comparison of Alternative Approaches. Proceedings of GIS.
- Sankaranarayanan, J. et al. (2005) Efficient Query Processing on Spatial Networks. Proceedings of ACM-GIS.
- Seydim, A.V. et al. (2001) Location Dependent Query Processing. Proceedings of MobiDE.
- Tao, Y. et al. (2002) Continuous Nearest Neighbor Search. Proceedings of VLDB.
- Xia, T. and Zhang, D. (2006) Continuous Reverse Nearest Neighbor Monitoring. Proceedings of ICDE.

MOD Indexing

- Cai, Y. and Ng, R.T. (2004) Indexing Spatio-Temporal Trajectories with Chebyshev Polynomials. Proceedings of ACM SIGMOD.
- Chen, S. et al. (2008) ST^{2}B-tree: A Self-Tunable Spatio-Temporal B+-tree Index for Moving Objects. Proceedings of ACM SIGMOD.
- Frentzos, E. (2003) Indexing Objects Moving on Fixed Networks. Proceedings of SSTD.
- Hadjieleftheriou, M. et al. (2006) Indexing Spatio-temporal Archives. VLDB Journal, 15(2): 143-164.
- Jensen, C. S. et al. (2004) Query and update efficient B$^+$-tree based indexing of moving objects. Proceedings of VLDB.
- Kollios, G. et al. (2001) Indexing Animated Objects Using Spatiotemporal Access Methods. IEEE Trans. Knowledge and Data Engineering, 13(5): 758-777.

MOD Indexing (cont.)

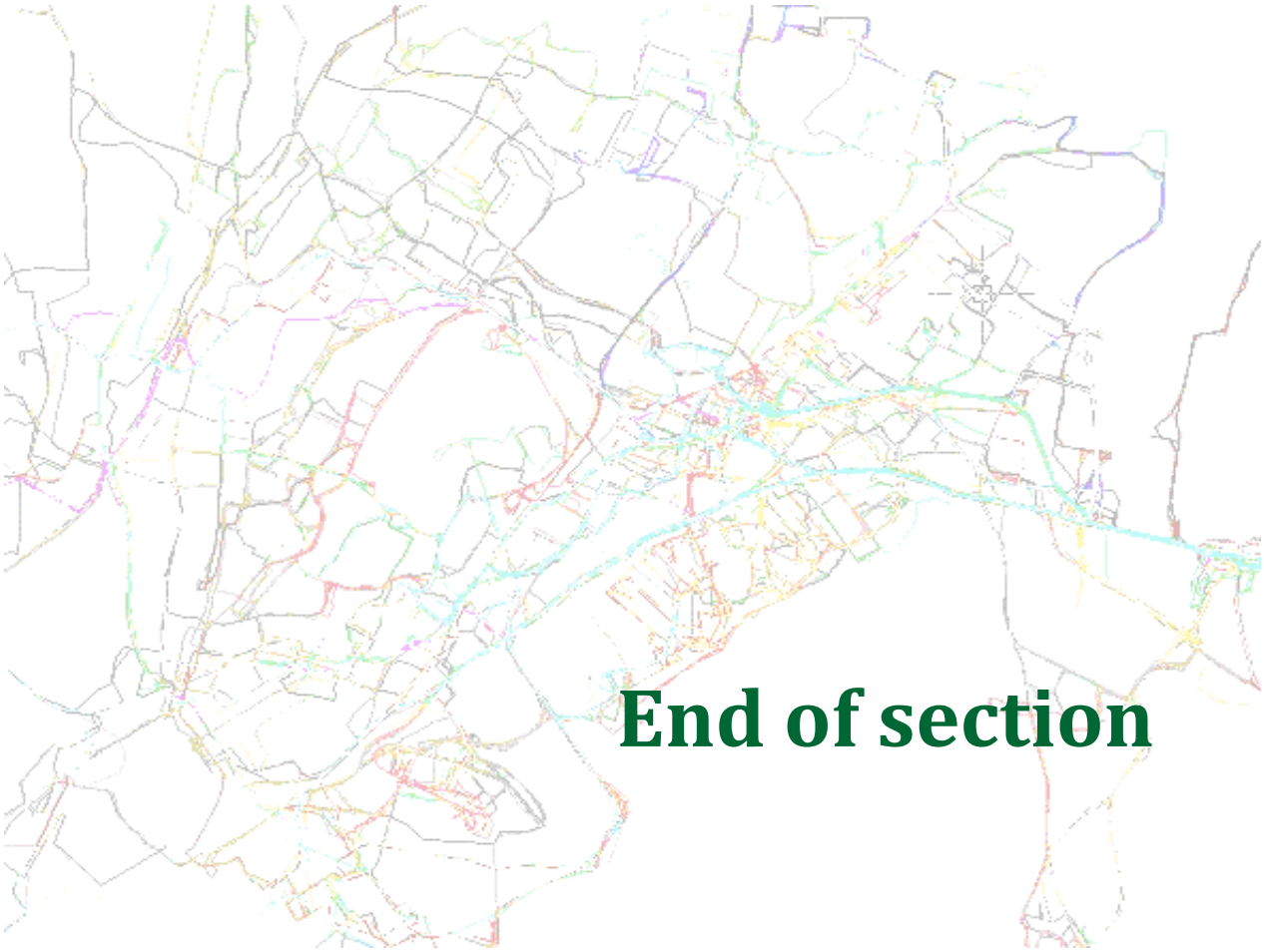
- Myllymaki, J. and Kaufman, J. (2003) High-Performance Spatial Indexing for Location-Based Services. Proceedings of WWW.
- Ni, J. and Ravishankar, C.V. (2007) Indexing Spatio-Temporal Trajectories with Efficient Polynomial Approximations. IEEE Trans. Knowledge and Data Engineering, 19(5): 663-678.
- Pelanis, M. et al. (2006). Indexing the past, present, and anticipated future positions of moving objects. ACM Trans. Database Systems, 31(1): 255-298.
- Pfoser, D. et al. (2000) Novel Approaches to the Indexing of Moving Object Trajectories. Proceedings of VLDB.
- Rasetic, S. et al. (2005) A Trajectory Splitting Model for Efficient Spatio-Temporal Indexing. Proceedings of VLDB.

MOD Indexing (cont.)

- Saltenis, S. et al. (2000) Indexing the Positions of Continuously Moving Objects. Proceedings of ACM SIGMOD.
- Saltenis, S. and C.S. Jensen (2002) Indexing of Moving Objects for Location-Based Services. Proceedings of ICDE.
- Tao, Y. and Papadias, D. (2001) MV3R-Tree: A Spatio-Temporal Access Method for Timestamp and Interval Queries. Proceedings of VLDB.
- Tao, Y. et al. (2003) The TPR*-tree: an optimized spatio-temporal access method for predictive queries. Proceedings of VLDB.

Trajectory Map Matching

- Brakatsoulas, S. et al. (2005) On Map-matching Vehicle Tracking Data. Proceedings of VLDB.
- Kellaris, G. et al. (2009) Trajectory Compression under Network Constraints. Proceedings of SSTD.
- Quddus, M.A. et al. (2000) A General Map Matching Algorithm for Transport Telematics Applications. GPS Solutions, 7(3): 157-167.
- White, C.E. et al. (2000) Some Map Matching Algorithms for Personal Navigation Assistants. Transportation Research Part C: Emerging Technologies, 8: 91-108.



End of section