Ubiquitous Tracking for Augmented Reality

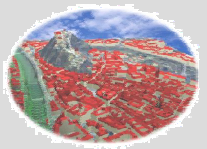
IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'04)

Martin Wagner, Martin Bauer, Asa MacWilliams, Dagmar Beyer, Daniel Pustka, Franz Strasser, Gudrun Klinker

Institut für Informatik
Technische Universität München
martin@augmentedreality.de

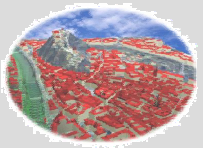
Joe Newman, Thomas Pintaric, Dieter Schmalstieg

Institut für Maschinelles Sehen und Darstellen
Technische Universität Graz
jfn@icg.tu-graz.ac.at



Overview

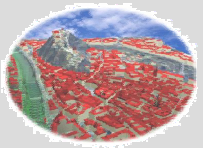
- Why we need Ubiquitous Tracking
- Formal model
- Implementation concepts
- DWARF-based implementation
- Simulation environment
- Conclusions & Future work



Why we need Ubiquitous Tracking

Bringing AR to intelligent environments:

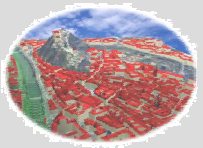
- AR applications extend their range of operation
  - Mobile AR
  - Powerful wearable devices
- Ubicomp applications extend their immersivity
  - “Natural” interaction benefits from accurate location information
- Combining tracking requirements from ubicomp *and* AR allows to use AR interaction in ubiquitous environments



Why we need Ubiquitous Tracking

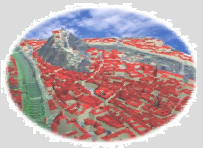
Enhancing AR tracking technology:

- No single sensor is perfect for all AR applications
  - Sensor fusion gains attention
  - Reusable solutions required
- Tracking technologies tend to build upon each other
  - Initialization problem for natural feature tracking
  - Stabilize results of absolute by relative tracker



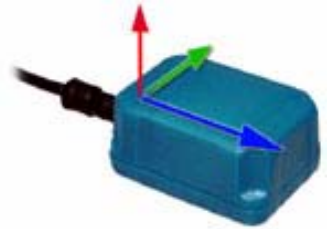
What is Ubiquitous Tracking?

- Abstraction Layer between location sensors and applications
- Gathers all available spatial relationships from sensors
- Provides inferences to deduce “best” possible spatial relationship between arbitrary objects in the system
  - Semantics of “best” is application dependent
  - Existing inferences (i.e. filter and fusion components) have to be integrated

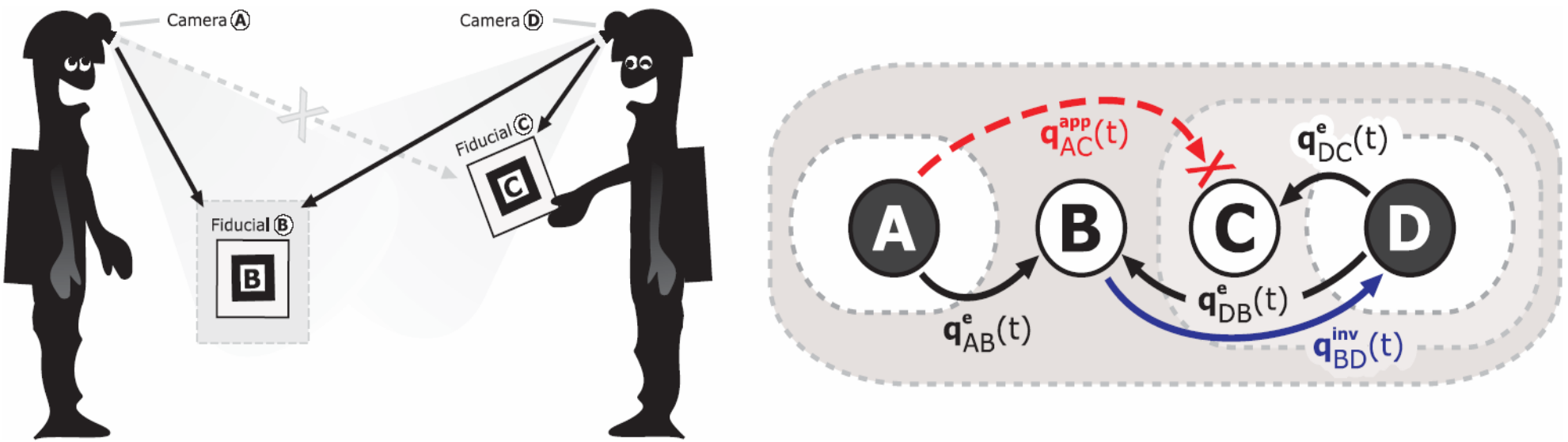


Definition of Terms

- A ***spatial relationship*** between two objects can be expressed in terms of multiple parameters (e.g. any dimension of position, orientation and their derivatives)
- A ***sensor*** performs a ***measurement*** of some physical property and computes an ***estimate*** of some spatial relationship parameter
- A ***locatable*** is an object whose spatial relationship to some reference coordinate system is estimated by a sensor
- An ***inference*** is an estimate of a spatial relationship computed from single or multiple estimates of spatial relationships



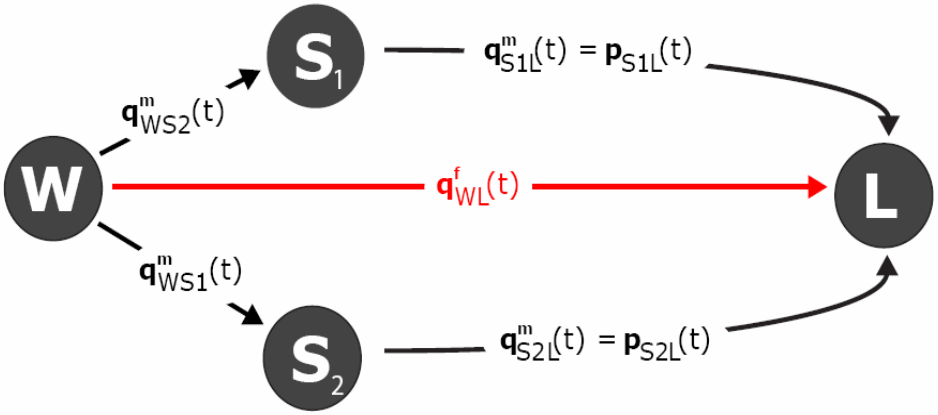
Formal Model



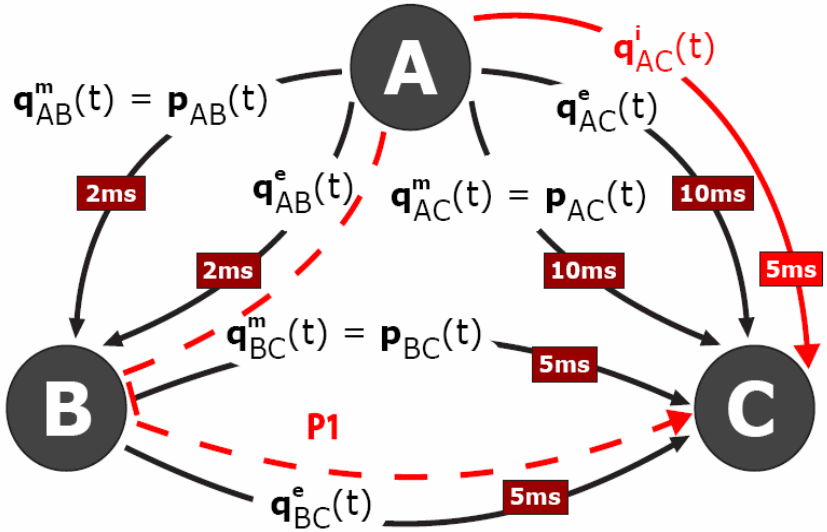
- Goal: uniform modelling of all spatial relationships
  - Handle estimates of diverse sensor classes
  - Handle inferences (i.e. filtering data, sensor fusion)
- Approach: directed ***spatial relationship graph***
  - Describe spatial relationships as functions of time
  - Functions yield estimates of spatial relationship characterised by attributes

Formal Model: Inferring Knowledge

- Integrate existing inferences (e.g. Kalman Filter fusing two sensors) by adding new edges to SR graph

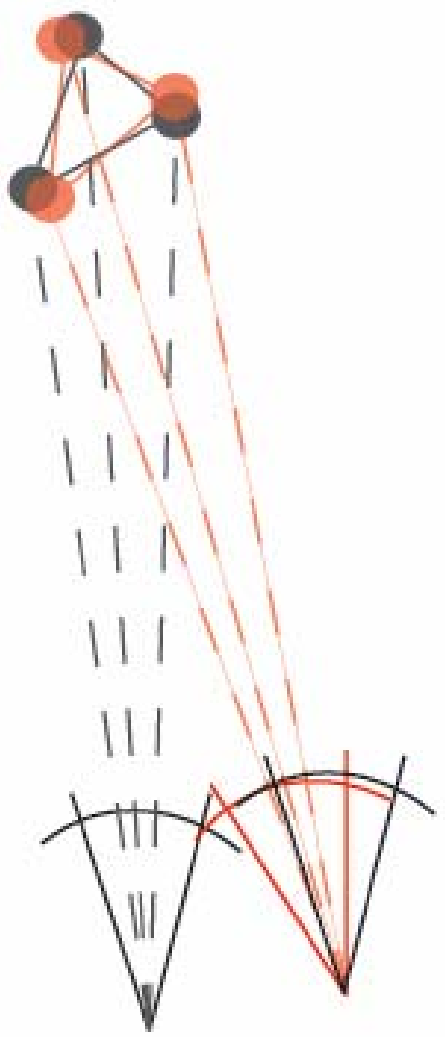


- Provide generic inferences by using transitivity property of spatial relationships
  - Search path in SR graph between relevant nodes



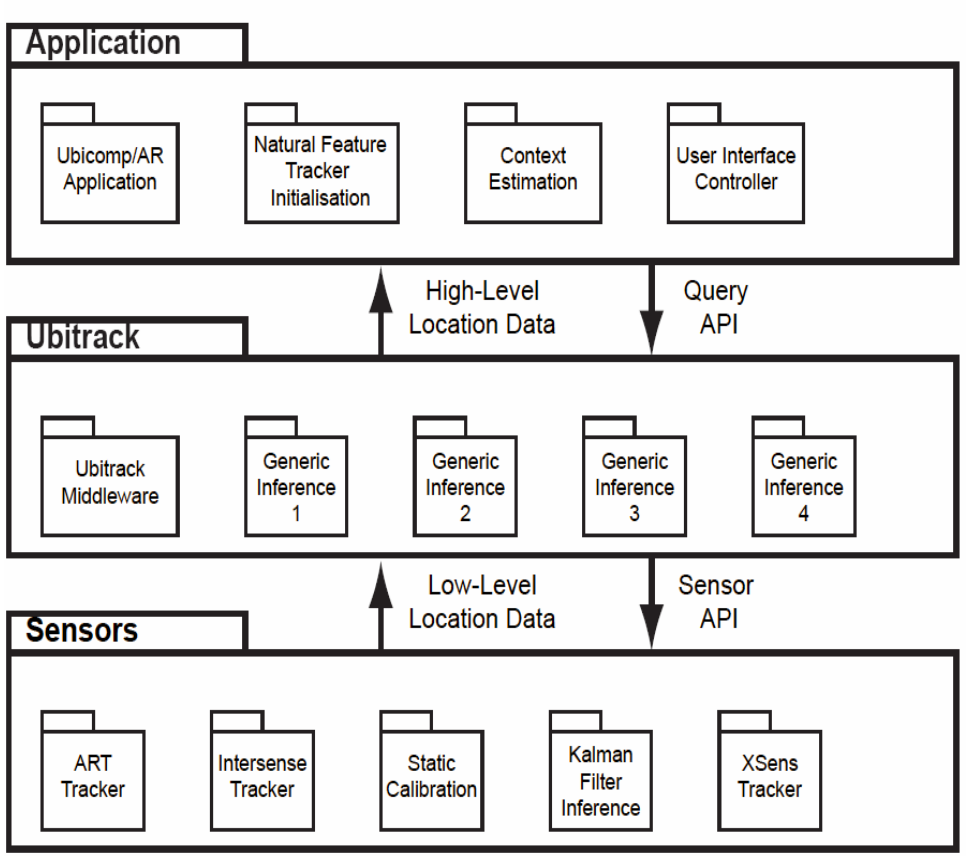
Formal Model: Challenges

- Directed graph: non-trivial inversion of edges
- Timing issues: measurements made at discrete points in time, demand for estimates in continuous time
- Should map onto real implementation without too many restrictive assumptions
- For this purpose: handle dynamic changes in availability of spatial relationships



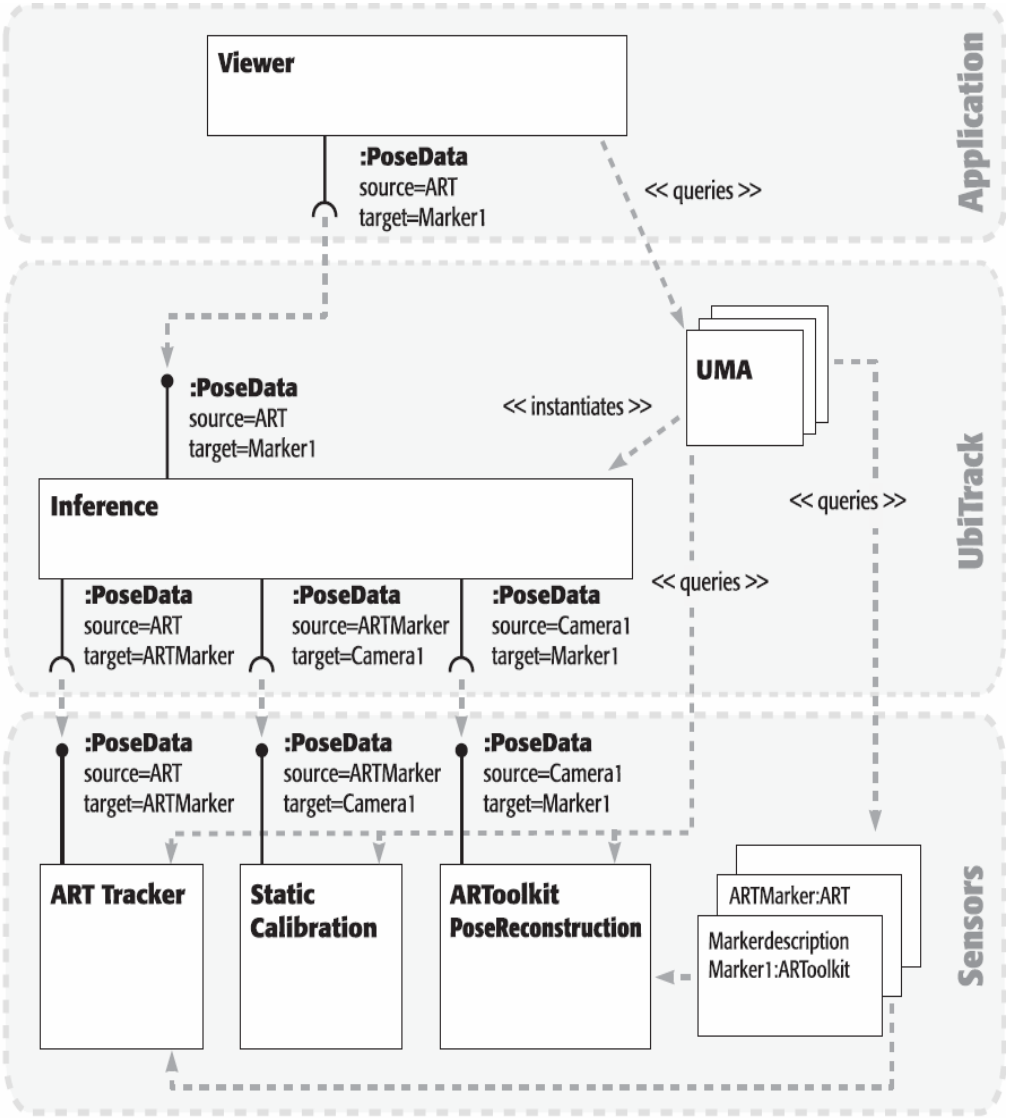
Implementation Concepts

- Layered Architecture:
  - Spatial relationship data moves from sensors to applications through filters inferring new spatial relationships
  - Set of filters built and connected on demand according to application's needs
- Data flow graphs
  - Flow of data through filters can be modeled as a graph
  - Assumption: form of data flow changes seldom compared to spatial relationships



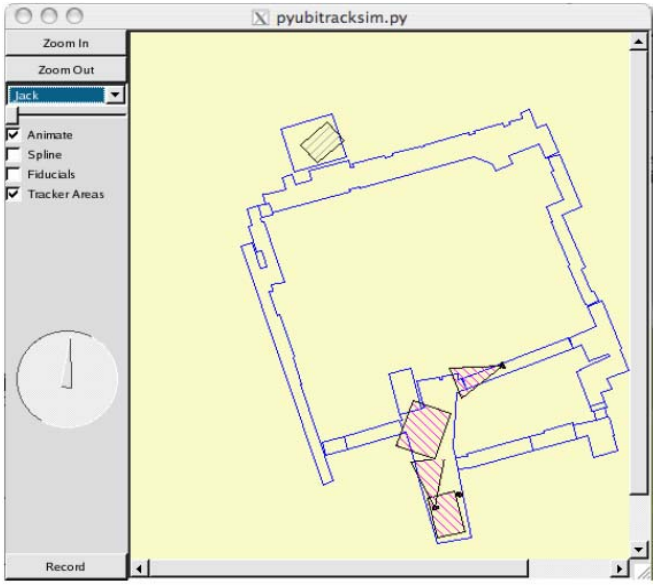
DWARF-Based Implementation

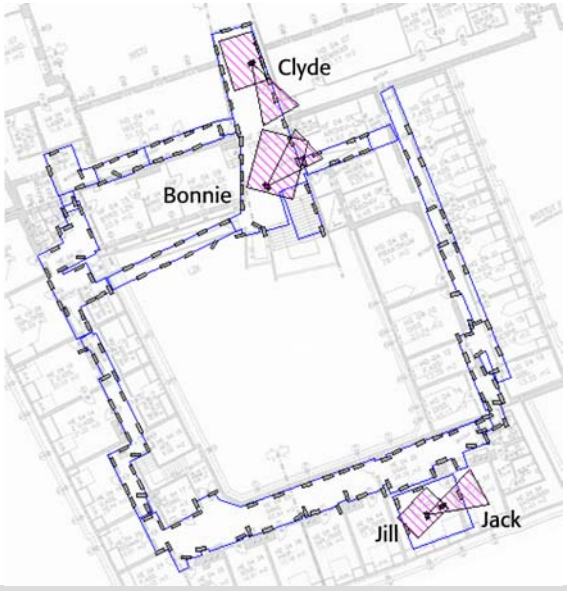
- DWARF is a distributed peer-to-peer middleware, modelling AR applications as set of distributed *services*
- Extension of DWARF middleware to allow generic Ubitrack inferences
- Resulting data flow consists of a set of services:
  - Sensor services encapsulate hardware devices
  - Inference services aggregate data (on multiple levels)
  - Application services consume data



Simulation Environment

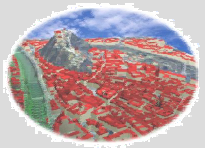
- Large-scale Ubitrack environments are not yet ready
  - Limited amount of sensors
- Ubitrack simulation environment allows to generate artificial multi-sensor tracking data
  - Test Ubitrack implementation by comparing results to simulation ground truth
- Generation of simulated images of scenes for feeding vision-based trackers





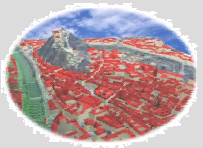
Conclusions

- Automated reusable sensor fusion is a prerequisite for bringing AR applications into large intelligent environments
- Formal model allows automated handling of large multi-sensor setups
- DWARF-based implementation shows feasibility of approach



Future Work

- Build large-scale setups for real world applications
- Incorporate sensors using different representations of spatial relationships (e.g. cell-based trackers)
- Exploit Ubitrack for natural feature trackers
- Autocalibrate parts of Ubitrack setups



Thank you.

- Any questions?