Panel Sessions

Panel Session 1:
Cyber Infrastructure for Informatics
Panel Chair: Chaitan Baru

Panel Session 2:
Databases for Ambient Intelligence
Panel Chair: Martin Kersten
Panel Write up

Organic Database Systems to support an Ambient World
Martin Kersten, Peter Boncz, CWI, Netherlands
Martin.Kersten@cwi.nl, Peter.Boncz@cwi.nl

1 Panel Overview
Ambient Intelligence has been identified by the research councils of the European Community as an area of great social and economic potential. The 6-th Framework program (2003-2007) has been crafted along this vision, calling for targeted research and development to make it come true. For the database community it is one of the rare occasions where our proven technology could make a difference, provided we are able to identify its requirements and we are willing to relax our drive for incremental research activities.

The play-ground for Ambient Intelligence is a physical area with many autonomous digital devices, which are sensitive, adaptive, and responsive to the presence of people. Within a home setting it is geared at improving the quality of life by creating a desired atmosphere and functionality via intelligent, personalised interconnected systems which blend with the physical background. The people are not aware of their existence. In a health-care setting it is concerned with the many devices surrounding a patient, or even devices implanted to keep you moving for many years with a minor body defect. Likewise, the ambient setting can be created for the car-industry to improve both security and to handle your job duties during the in productive hours in traffic jams.

Progress in the area of Ambient Intelligence calls for large investments in many areas of computer science, e.g. embedded sensor systems, (radio) networks with low energy consumption, small footprint component-based software development, effective learning algorithms, ... but also data management. An ambient environment contains many (small/large) data sources, which are accessed by highly distributed applications.

It is up to the database community to avoid re-invention of the wheel by timely provision of the necessary technology, algorithms, and software components.

The panelists are challenged to comment on the opportunities, pitfalls, and laboratory progress on database technology for the Ambient setting. A futuristic scenario is used as a frame of reference to derive the requirements for a DBMS for the Ambient arena.

2 A Climate Control System (ACCS)
At 19:00 the garden sprinkler installation starts collecting the temperature and humidity levels of the thermometer pods distributed in the garden by wireless access. It subsequently contacts the local weather forecast site and obtains the next 24 hours weather prediction. It generates plans for mass flooding and dripping, sent to the respective sprinkler installations, which realize the plan autonomously using up to date information of the thermometer pods on soil humidity.

At 20:00 the lawn sprinkler starts its 40 minute rain program, but becomes interrupted by the dog, which disconnects the water hose. The sprinkler detects the defect and issues a wireless telephone call to the main building to warn the gardener. It so happens that he is not at home, and the message is forwarded to his car. The sprinkler system is instructed to suspend the program until he returns home by 22:00.

The reply message is also picked up by the home air conditioning system, which activates the cooling system, to lower the temperature to 18°C after the exceptional sunny day.

At 22:15 the water hose is reconnected, and the sprinkler continues the plan taking into account the actual soil humidity.
reached before it was interrupted. The gardener installs a few new thermometer pods, because the latest version had better learning capabilities and included an acid test sensor. Upon activation they gracefully integrate with the other pods, sharing historical data about the premises.

### 3 How to make this happening

The ingredients to realize our ACCS are readily available. The hardware can be produced at low cost, powering the system can range from batteries to solar cells, and the processing capabilities of the gizmos involved will reach mid-90 PC capabilities. Furthermore, the software layer can be constructed out of Java beans with JINI technology as a basis to distribute and publicize services. The complexity of this environment does not stress the need for complex learning algorithms, simple statistics and small decision trees may suffice.

And still, the ACCS would only be constructed by a partnership of a few world-leading companies through a tremendous investment unless the data management issues can be abstracted away, much like the networking issues have been encapsulated in standards. This can be illustrated as follows:

- **Autonomy.** Every gizmo in the ACS is an independent and autonomous device. Its communication with and reliance on the environment is not a priori fixed. Yet, the basic functionality includes (persistent) storage of data and the ability to answer queries form its environment. As such, it should also provide the traditional database functionality, at least as long as its energy lasts.

- **Self-descriptiveness.** Most gizmos will be operational for many years, e.g. new brands will be added on a regular basis. This involves both replacement of functional identical copies to merging with gizmos with new data items and functionality. The consequence for the data management infrastructure is to be able to correctly interpret the database schemas, at least to the point where it guarantees backward compatibility. Methods and technology to plan for such evolutionary paths (even in the confines of a simple ACCS) are in open area of research.

- **Self-organization.** Our gizmos are hardware constrained, we simply cannot add more memory, a processor, or a hard disk. They are mass produced with fixed hardware characteristics. Yet, changes initiated from an outside request or as a result of an overload of local sensory information (e.g. a shadow of a plant waving in the wind may lead to quick changes in the temperature being measured) should be handled locally by involving information gathered from nearby sensors or to delegate a (sub) task transparently for the user. Software technology to specify, implement and monitor self-adaptation are scarcely researched beyond niches in AI. In the database area it is limited to dynamic replication and query scheduling in a heterogeneous, distributed system.

- **Self-repair.** Once a gizmo fails to operate (being crushed by the toys of my son) we should be able to simply place a new ‘empty’ gizmo in place. It should take over any accessible information from the broken one or its space is being conquered by the remaining gizmos through self-replication of their software state. Although we should not expect the gizmos be capable of hardware self-repair, we should stress at least the level of software stability. An a posteriori recognized software/hardware bug should be isolated and automatic software detours have to be activated to maintain a functional system. What database system has been designed with this feature in mind?

- **Self-awareness.** The gizmos should be location aware to properly (inter) operate and to provide a coherent view on the state of the ambient environment. Likewise, they should be time aware. Both from the perspective of system functionality (don’t water the plants at noon), as for being able to manage and deal with historical data. Finally, they should be aware of trust relationships maintained with their peers, which calls for an board security technology.

Assuming you haven’t left this futuristic scheme by now, we charter a grand challenge for database researchers:

*Develop an “Organic” DBMS which can be embedded in a wide collection of hardware appliances and provide an autonomous, self-descriptive, self-organizing, self-repairing, self-aware, secure and stable data management functionality to its environment.*

Isolation of only the data management issues involved in an ambient setting, i.e. keeping track of historical data, answering unforeseen queries about this history, and compaction of replication of information to deal with storage limitations, already triggers the need to develop a new generation of database management software. It is phrased an Organic DBMS, because it should be designed from the grounds up as an evolutionary system, which occupies multiple physical spaces and still maintains a unified view to those interested in its content.

On top of this database, software agents, learning techniques, like communication methods, have been encapsulated in standards and robotics can be combined into an application that realizes our ACCS.

### 4 Panelist Challenges

The ADORA principles (*autonomy, self-descriptiveness, -organizing, -repair, -awareness*) of the organic database are not yet found in current product offerings or system prototypes. The challenge to the panelists is to indicate either where progress is to be expected in the near future, or what other research challenges lay ahead to realize the vision expressed.

The shortlist of panelists contacted to contribute: please describe in one sentence your answer/question/position.