Wireless Graffiti - Data, data everywhere[1]

Tomasz Imielinski and Badri Nath

Department of Computer Science
Rutgers University

Piscataway, NJ 08854, USA
{imielins,badri}@cs.rutgers.edu

Abstract

In this paper, we take a retrospective look at the problem of querying and updating location dependent data in massively distributed mobile environments. Looking forward, we paint our vision of the future *dataspace* – physical space enhanced with embedded digital information. Finally we describe a few of the applications enabled by dataspace due to the availability of large scale ad-hoc sensor networks, short-range wireless communications, and fine-grain location information.

1. Introduction

Back in early 1992, when we were hastily preparing and submitting the paper "Querying in highly mobile, wireless environments" to the 18th VLDB conference, the idea of a cell phone becoming personal digital assistants, or palm/pocket computers was still largely a fantasy. Today, we have over half a billion cell phone users and close to a million wireless data users.

But in early 1991, motivated by a talk given by Dr. David Goodman, the founder and the then Director of Wireless Information Network Laboratory (WINLAB) at Rutgers, and the many discussions with him, both of us engaged in an exciting intellectual speculation game of "what if", or perhaps "what when". What if cell phones turn into small computers with wireless connectivity and small displays, and perhaps with voice driven input/output interface? What will be the key software issues? What new challenge does this pose? Will communication protocols, data processing, and data management remain the same as for regular desktops, or will there be a fundamental change and if so how? Some of the differences were clear – the PDAs would no longer be at a fixed location. Instead, they will be mobile. Wireless bandwidth will be precious and limited as compared to the bandwidth of wired networks. More importantly, the quality of the wireless connection will vary depending more on the layout of the physical terrain and even weather conditions.

There were other issues that were brought to our attention by Dr. David Goodman and his colleagues. As computer scientists, we would have never guessed that we need techniques to extend battery life on portable devices or that power/energy was even an issue. We later learned that contrary to the exponential growth in processor speed and internal/external memory bandwidth, improvement in battery lifetime is far slower – measured in low double digits over many years. Thus, energy efficiency was predicted to be one of the key issues in developing operating system support as well as communication protocols for mobile devices. In other papers [ImBa93, ImBa94] we have summarized the three key challenges confronting software design to enable mobile computing: *Mobility*, Bandwidth and *Energy*. Our work in the early nineties addressed all of the three components.

In [vldb92] we focused on mobility and location management for millions of mobile terminals. This paper not only introduced a new area of research and raised some interesting research questions, it also proposed solutions, some of which were widely adopted in wireless networks of today. These proposed solutions were in the area of location management, for updating and querying location dependent data.

2. Fundamental Nature of Location

Before the emergence of mobile computers geographic location of the device had no importance. While each

**Proceedings of the 28th VLDB Conference,
Hong Kong, China, 2002**

---

[1] Invited paper- VLDB 10 year best paper award

desktop always had a clock and time was a shell variable that can be used in any application, there was no need for a similar variable that could refer to the current location of the terminal since its location changes only rarely. With mobility, the current location of a device became as important as the current time. Eventually location will become a shell variable (or variable accessible to any programming API) and thus become as significant to the applications as the value of the local clock. In [vldb92], we examined the consequences of location becoming a "first class citizen" to databases. The entire cellular infrastructure was viewed as a gigantic database of locations of millions of terminals, which constantly move between cells and location servers. These terminals are constantly searched for as well as they continuously update their current locations to the location servers. From the existing literature on location management in cellular networks we learned that it is impractical to assume precise knowledge of location of all terminals at all times. Such dense traffic of location updates would simply bring down the cellular network or at least seriously degrade its functioning. Instead, information about individual locations of mobile terminals/cell phone is almost always imprecise. Consequently, querying location dependent data is a problem of querying rapidly changing database that never has complete information. However this incomplete information can be completed if necessary by additional actions through "paging" and finding out where a particular terminal is located at a given time. Stated in database terms that amounted to querying incomplete data with an option of "completing it" in real time if the query requires it. In other words, a new class of query processing problems emerged where data acquisition became a part of query evaluation.

Location management of course existed as research field long before our paper was published. However, our work was the first to raise the issue of ad hoc querying for locations of other users for purposes different than simply establishing a voice connection. The motivating examples dealt with finding positions of other mobile objects such as taxis, ambulances and other emergency vehicles located in the vicinity of a mobile user. Such queries were fundamentally different from the one-to-one connection oriented task of finding a specific user (cell phone) in order to set up a phone call. Indeed, by their very nature, such location queries would return potentially many answers, or aggregate location information from many mobile terminals. To our knowledge [vldb92] was the first paper introducing querying of location dependent data where both the querying and queried terminals were mobile. In addition, the paper made contributions to the location update problem as well. Below we briefly summarize some of them.

2.1. Updating Locations "Do not tell me if I do not want to know".

In [vldb92], we proposed a family of adaptive location updating schemes. In these schemes, location information is distributed on a "need to know" basis and certain degree of ignorance about locations is postulated in order to avoid excessive volumes of location updates. However, ignorance about location is bounded, i.e., the whereabouts of a specific terminal are always known within a certain predefined zone. In one such scheme terminal always informs the location server while crossing the border between zones but does not inform when moves are contained with a single zone; in other schemes the informing policy depends on the so called call/mobility ratio. The call/mobility ratio is high when there are many requests for a location of a user while she/he stays at any location (more requests/calls than location changes). The call/mobility ratio is low if a terminal is moving more often than it is queried (or searched/paged). Low call/mobility ratio implies lesser need for location updates than the high call/mobility ratio. Additionally, we proposed adaptive zones for individual users. Such zones depend on individual mobility patterns – inter-zone moves are rare, while intra-zone moves may be much more frequent; for example, there may be a work *zone* and *home zone* with the border is typically crossed only twice a day on the way to and from work. This idea of updating only on an inter-zone move is also adopted in location management schemes in 3rd generation wireless networks [Be99].

2.2. Querying Locations - "I don't know but I can find out"

In [vldb92], we represented the problem of querying location data as a mixture of query processing in the traditional database sense and the problem of adaptive data acquisition. Under the heading "I do not know but can find out" scheme we reduced the problem of deciding whether there is a need to "page" or find out locations of some of the terminals to a classification problem (in the sense of machine learning). Intuitively, some queries (general aggregate queries such as count of a number of terminals in specific areas) could be answered on the basis of incomplete information which is currently available in location servers; other queries may require extra effort to "find out" more precise locations of the terminals, thus incurring additional network cost. Therefore, querying location databases could be viewed as the process of finding the right data acquisition strategy since location data can be acquired with an extra effort and extra latency. This led to a new family of query processing problems where data acquisition is built into query processing. Queries are processed not necessarily just using distributed databases which store locations, but also invoking a significant number of network messages due to paging of exact locations of selected terminals. We demonstrated that the problem of optimal data acquisition strategy could be reduced to the problem of optimal classification tree in a machine learning sense.

2.3. 1992-2002- A decade of research

Although our research project DataMan (www.cs.rutgers.edu/dataman) started a year or so earlier, the paper [vldb92] was one of our first publications in the area of wireless and mobile computing, particularly on issues related to data management in mobile computing. Subsequently, in other papers [ImBa94] and [ImBa93] we introduced and discussed broad issues and challenges that mobile computing brings to data management.

The location management research area has grown significantly with numerous papers proposing new approaches to current and future location management problems. The concept of location has since evolved from a broad "cellular" concept (simply a cell id) to a more precise notion of location based on the GPS and signal triangulation schemes which are used where GPS no longer works satisfactorily [Bu00,Ba00,Pr00,NiNa01]. As we will discuss in the next section, querying and managing location dependent data is still an open research area due to the emergence of sensor networks and ad hoc wireless networks. Location management as seen in [vldb92] used a very coarse granularity of location, a few thousand feet as opposed to the problem of location management for the present and the future, where the resolution is expected to be in the order of a few feet[HiBo01,Hi02].

Before we move on to our vision of the future research in this area, let us summarize briefly the impact of other research work in the DataMan group.

In papers such as [ViImBa94] and [ImViBa97] we were first to introduce the concept of "data on air" – that is data which is periodically broadcast or multicast via a push mechanism over the wireless medium. In [ViImBa94] and [BaIm94] we discussed issues of energy efficient indexing and caching. In the area of wireless publishing, a hybrid push/pull approach to query and management of wide area wireless dissemination systems was introduced [ImVi96]. Research efforts by others include the work on Air-disks [Ac95, Ac96] and related work on dissemination or broadcast schemes for wireless and location management [Ba92,Ja96,Sh97,LiTs98, RoYa97].

In 1994, together with Hank Korth we organized a MobiData workshop at Rutgers – this brought together many key contributors in the areas of mobile databases and mobile data management. Later, jointly with Hank Korth, an edited a volume on Mobile Computing was published [ImKo96]. In the late nineties mobile computing became a mainstream research area with its own ACM conference (Mobicom), journals and numerous workshops on many topics which were initially addressed in [vldb92] and other papers from our group.

3. What's next? DATASPACE[1]

As a result of the explosive growth of wireless Internet technologies in the late nineties, what was fantasy in 1992 became reality in 2002. The bursting of the Internet bubble and excessive spending on the part of many wireless infrastructure companies on wireless spectrum has slowed down the rapid pace of progress in deploying future generations of wireless networks. The so-called 2.5G (GPRS) and 3G wireless networks are emerging much more slowly than predicted a few years ago. This slower progress is undoubtedly a function of slowing economy and a need for new types of wireless terminals to "catch up" with requirements of the broader wireless band. User expectations have grown tremendously since the early days of wireless Internet. New wireless terminals run standard browsers such as Internet Explorer or Netscape. Terminals that combine browser and computer functions with standard voice functions of a telephone are now emerging in the market. Soon, GPS capabilities will be built into these devices as well. These terminals, which Dr. David Goodman calls "cell phones on steroids" still face old problems plaguing cell phones – miniature displays and often difficult and obscure data entry. Wearable computing with miniature screen displays embedded into glasses and one handed "chordal" keyboards are still in early experimental stages.

We also see physical devices such as various types of sensors becoming network aware and connected to the Internet through various types of wireless connections. Such sensors include road sensors (bridges, close loop highway sensors), environmental sensors such as rainfall-level sensors, pollution sensors, vending machines, home sensors, light detectors etc. These sensors are not only producing data but often are able to store it. Sensors are not only static but also mobile such as GPS units deployed on cars, other car sensors such as accelerometers etc. GPS units are now optionally available as part of navigation systems of all major car models and precision of GPS allows location (outside of buildings) often within a few meters (even within a few centimetres using differential GPS).

---

[1] This is joint work with Samir Goel

Thus the major technological trends today are: 3G networks, proliferation of GPS, and miniaturizations and rapid growth of different types of devices with sensors that are connected to the network and are capable of not only producing and sending but also storing information.

What is the significance of these trends to the data management community?

The rapid growth in the number of network aware and network connected physical devices creates a new kind of physical space, a space that is digitally "wired" to the network (although the term "wired" is used here loosely since most of the time the connection is actually wireless). We call this new space, the **dataspace** [ImGo99a,ImGo00]**.** In dataspace information is stored where it is produced, and not at some remote server. Because of the highly refined notion of location, dataspace can be viewed just like physical space, but enhanced with digital information. This information may be produced locally or "parked" at a given location by a passing car. For example if a car detects a road hazard (e.g., pothole, slippery or icy conditions on the road) it may "park" a warning for other cars; a user may want to leave the last mile directions at a given location. Thus one can imagine a dataspace as a "habitat" with physical devices and humans constantly producing and consuming location-dependent and location-independent data. These inhabitants can post/attach messages and information at a location. Thus each location has its own "history channel": the time stamped "news channel" of past and present postings by humans and sensors. Such a space is bound to emerge just as World Wide Web has emerged in late eighties and early nineties.

Dataspace is digitally enabled physical space; it is as if additional "sense" has been added to the physical reality.

When the physical space around us is illuminated with different types of radiation, specific sets of objects respond to specific types of radiation, even though the radiation may fall on all objects. For example, most of the objects in our environment reflect visible light, bones reflect X-ray radiation, tissues resonate in magnetic fields. On similar lines, we can imagine "illuminating" the dataspace around us by "beaming" various types of requests on the deeply networked collections of objects contained in the space. A request "beamed" on the network may ask for the retrieval of data or the completion of a task. Further, we can imagine specific sets of objects responding to the request placed by a user, depending upon the request type and the properties of the object. Through such a network-mediated illumination of the physical world, one can experience and control billions of objects in one's immediate or remote environment. Dataspace can be viewed as the 3-D physical space, for example starting from 10 kilometers below the surface of earth and extending up to 100 kilometers above the surface of earth (Figure 1). The dataspace is populated by very large number of objects that produce and locally store data pertaining to them. In the dataspace, physical objects are not only characterized by shape, size, and color, but also- by processor type, the amount of memory, and the network connectivity.

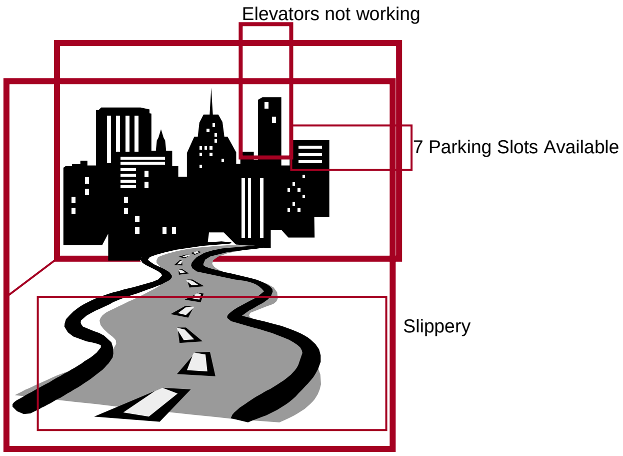

Figure 1 Wireless Graffiti in Dataspaces

Here are examples of physical devices that may constitute the future dataspace

1. Sensors, such as pollution, temperature, humidity, rainfall level, door locks, GPS
2. Remote cameras
3. Tags on physical objects such as the tags manufactured by Texas Instrument's TagIt
4. Messages, location anchored pages, location based bulletin boards

Information in dataspace will range from sensory data along the roads and bridges, air contamination/pollution information, video feeds from remote cameras, - to medical and biological data such as data about individual's vital statistics like blood pressure, heartbeat rate, to perhaps more futuristic biological sensors which can detect presence of cancer cells in the parts of the body. Messages posted and anchored to locations may include directions, announcements about road repairs, a bill board on a local street, etc.

Just like in *real space* in which we move through, spatial coordinates are the basic points of reference to navigate and query dataspace. However, unlike in real space, users can navigate through and query objects that are much beyond the range of their senses. Thus, by "moving" through dataspace, a user, connected to the network, acquires an enhanced awareness of his surroundings as well as of even very remote areas including information that would be previously invisible or out of his access. Dataspace in effect gives the user a "sixth sense" through which to perceive the world. Users can also point at physical objects with their location and direction aware terminals and receive digital information associated with the object for example its web page.

This leads to two different types of dataspace querying: *virtual* querying and *physical* querying. Virtual querying is querying in standard sense of the word – a query is sent to a server or directly to the queried dataspace elements (sensors) over the digital network. In contrast, physical querying is guided visually by the user himself, by what the user sees or hears. Here, querying is a physical act of pointing the terminal towards the object (e.g., building, mountain, road). This can be combined with remote physical querying by operating a remote camera.

The concept of dataspace is in contrast to the traditional concept of database. A database stores information locally about remote physical objects. Here physical objects become merely the artifacts of their corresponding entry in a database. In dataspace, on the other hand, data is inherently dispersed and connected --- it "lives on" the physical object; it is an inherent part of that physical object; it is stored *with* the object, and may be queried by reaching that object through the network. Data is another natural characteristic of the object similar to its weight, color, and general appearance.

Emergence of dataspace will create new challenges to data management research community such as:

1. Querying and Updating – query and update processing will need to have significant networking component and use network communication primitives such as multicast and broadcast to process queries and updates. The fundamental tradeoffs between updates versus search to locate data apply here as well but the extent of the scale that we envision has never been explored.
2. Browsing – Dataspace will require new type of spatio-temporal browsers which are map-centric and equipped with the powerful zooming "in and out" capabilities.
3. Authorization – Access control issues over who is allowed to attach what information to which location will have to be addressed. Dataspace can be viewed as a "spatial file system" where different groups of users may be allowed to "write" at different locations (what we term as "wireless graffiti" in [Sigir01]). Similarly, reading and querying location dependent information may be restricted to certain groups.
4. Communication – Research into protocols for supporting large scale deployment of ad-hoc sensor networks is just beginning [Es00, NRC01]. What are the wireless technologies that will enable short-range communication in dense networks [Le01,Le01a,Bh01,Mu01]? How will wireless spectrum be managed in such a chaotic environment [Ha68]? What are the rules that govern usage of wireless spectrum? Can usage and availability of spectrum at a given location be modelled as another dataspace?
5. Networking – When every object that can sense and store data is networked as a large-scale flat ad-hoc network, how will many of the networking functions be implemented? What is a natural namespace for describing the elements of the network? Many of the traditional naming conventions do not apply here. The fleeting nature of the objects, mobility and locality of communication makes static mapping unworkable. Since, many of the networked objects traverse well known paths and queries of interest will be constrained to the physical structure of the surrounding, a possible routing strategy could be based on trajectories or other geometric shapes that describe the routes of packet traversal [NiNa02].
6. Data management – Every object, reporting a value at every location is a huge data gathering process. Obviously, compressing data locally and globally are needed. Here again, there is tradeoff between the ability of precise replay and storage requirements.

Below we briefly discuss how the above challenges can be met by using the solution space of networks and database management systems.

3.1. Querying and updating dataspace: network as a database engine

Dataspace can be organized as a collection of three-dimensional *administrative* "cubes" and *geometric* cubes of data. Administrative data*cubes* can encapsulate a city, a street, a building, a basement, a room and even a shelf or a drawer; it can also include the interior of an engine (one of the cylinders for example) or the left hemisphere of someone's brain. Geometric datacubes can range from a *cubic mile* around the Empire State Building to a *cubic millimeter* inside human body (with adequate biological sensors reporting the information).

In dataspace, collections of objects inhabit various administrative and geometric datacubes. In order that these objects can be usefully accessed and manipulated, they need to be organized into logical clusters. This is achieved by grouping objects into classes called *dataflocks*. Dataflocks are classes of often-mobile objects that move through the physical world while still maintaining (some level of) connectivity to the network.

Dataflocks are accessed through the datacubes they are located in and through their own class properties.

New objects become part of a datacube by a simple "plug and play" mechanism. Rather than requiring a complex prior protocol, this event involves little processing other than registering of a physical object in the scope of its "vision". The Plug and Play is emphasized in JINI [Ji99], which is the most serious attempt yet to build "smart spaces" of devices such as home appliances, printers, photocopy machines. Using JINI, these devices can be connected to the local area network and can be electronically controlled. However, JINI is still oriented towards the logical structure of the Internet where locality and distance are defined not by geographic space but rather by the Internet's logical structure of subnets. In contrast, dataspace is spatial and embedded in the physical reality that surrounds us. Here, the "local area network" is replaced by a room, a street, or the top of a mountain, depending on where the user is located.

We support querying and monitoring dataflocks at the network level using multicast mechanisms. "Beaming" a request to a specific datacube (datacube "illumination") such that specific sets of objects respond to it is naturally amenable to implementation via multicasting. For this purpose one may use shared multicast trees [BaFrCr93,De96], where multicast group membership for a given datacube is calculated on the basis of its physical location. This way the network itself can take care of making sure that a given message "illuminates" only a specific datacube. Moreover, the network can also support indexing on selected attributes of dataflocks through multicasting. More precisely, for an indexed attribute, each pair *h = (Attribute, value)* is mapped to a corresponding multicast address and all objects which satisfy the predicate *(Attribute = value)* belong to that multicast group[2]. Mapping an *(attribute, value)* pair to a multicast address is performed by a simple, standard, pre-defined mechanism. The mechanism may be as simple as a hash function.

The process of querying in dataspace parallels that in a traditional database. In a traditional database, a query is processed by first filtering tuples based on the condition on the index attribute. The tuples that satisfy the condition on the index attribute are then sequentially checked for the satisfaction of the remaining conditions of the query. In a similar way, a query is processed in dataspace by first filtering the objects based on the condition on the index attribute. This is done by mapping the condition on the index attribute to a multicast address. It is important to note that *only* the condition on the index attribute is mapped to a multicast address; the whole query is *not* mapped to a multicast address. A message containing the query is sent at this multicast address. This message reaches all those objects that satisfy the condition on the index attribute. This way, in dataspace, multicast serves as an indexing mechanism for this distributed collection of data. The objects that receive the multicast message extract the query from the message and check if they satisfy the remaining conditions of the query. If they do, they respond with their *id*, otherwise they silently discard the received message. Thus the filtering on the index attribute happens at the network layer, while the filtering on the remaining conditions of the query is done at the application layer.

An alternative architecture for dataspace uses Geocast (geographic routing) [NaIm97] as the network primitive along with network indices and builds dataspace on top of it. Geographic routing aims at routing a message based on the geographical location of the destination rather than IP address. The destination may even be a region bounded by a set of GPS co-ordinates, in which case, the message is broadcast in the specified region.

Implementing indexes at the network layer has several advantages over implementing them at the application layer. Firstly, the network layer provides us with an efficient mechanism for sending a message to a group of objects via multicast. An application layer implementation will have to either use multiple unicast connections or use broadcast to achieve the same result. Clearly, performing a single multicast is much more resource efficient than performing multiple unicasts or performing broadcast. This makes the network layer solution much more scalable than an application layer solution. Secondly, a network layer solution delegates the task of providing the basic functionalities to the lower level of the abstraction hierarchy. This allows the network to deal with network-specific issues such as disconnection, router failure, etc., rather than having to re-implement what the network does on the application layer. This is closer to the "plug and play" philosophy of seamless network presence for devices and objects; implementing join/drop/response operations on the lower levels of the network hierarchy makes them easier to standardize and frees an application from unnecessary burden.

3.2. Query answering

Due to the wide geographic distribution of dataspace objects, collecting the full answer to a query may be

---

[2] In a practical implementation, one would probably choose a many-to-one mapping, mapping a set of possible values of the *index attribute* to a distinct multicast address. This is especially useful in case when an index attribute takes continuous values.

impossible for two reasons. Firstly, network availability cannot be guaranteed for the duration of the query. Secondly, even if connectivity is maintained, latency and delay necessary to collect the entire answer may simply be unacceptable. We identify two quality-of-service parameters for characterizing any architecture for dataspace: *observability* and *awareness*.

The query *q* is *R-observable* at an object *o*, if *o* can be provided with the answer to *q* such that the answer satisfies restriction R. The restriction R specifies certain assurances on the quality of the answer. For example, consider a dataspace of traffic sensors sensing vehicles passing by different locations on streets of Manhattan. Consider the query: *"Find the total number of vehicles in incoming traffic to Times Square"*. If a querier receives a response from all sensors satisfying this query within 60 seconds, then this query is said to be *60second-observable* at this querier.

An object *o* is *R-aware* of a query *q* if at any time it knows the current answer to *q*, where "current" is defined by restriction R. The restriction R specifies certain assurances on the quality of the answer known at *o*. For example, in a dataspace of sensors sensing temperature in all parts of a building, the building administrator is said to be *5seconds-aware* of the query: *"Is there fire in any part of the building?"*, if he receives a triggered/periodic update within a maximum of 5 seconds of time anytime some portion of the building catches fire.

R-observability and R-awareness reflect the need to define approximate (for e.g., 50%-complete-observable) query answering and query monitoring in volatile-networked environments with varying reachability, disconnections, etc.

In summary, *dataspace* is physical space enhanced with connectivity to the network. Dataspace is a collection of datacubes that are populated by classes of often-mobile objects called dataflocks. Objects in dataspace produce and store their own data. Such objects can be queried and monitored on the basis of their properties. Dataspace is the next generation of the world wide web, with two major differences: it is embedded in the physical reality --- organized in a geographic/spatial manner rather than logically as WWW is today; and it supports huge number of objects that produce and store their own data and move through dataspace. While browsing is the main navigation mechanism for the web, querying and monitoring are the main "navigation" mechanisms for dataspace

Dataspace will require a new type of a geographic browser which will be "map centric" and equipped with zoom-in and zoom-out capabilities. We have built some early prototypes for such browser in [NaIm97]. Zoom in capabilities should allow the user to browse the physical space in very similar way as we walk through the space. Such "walk throughs" should display a "dashboard" of controls are available at a given location. For example, specific data cube may be equipped with controllable video cameras, microphones or even robotic arms.

Writing and updating dataspace raises interesting authorization issues. Who is allow to update (post, park) information at a given location, within a given data cube? Data cubes in dataspace can be treated very much like real estate with its property rules. Just like URLs belong to corporations/organizations/individuals, spaces in dataspace may have individual and collective owners. Situation is analogous to file systems where instead of files we deal with regions of space in the dataspace.

Summarizing, the predicted emergence of dataspace will raise numerous research challenges. First and the most immediate of them is incorporation of location as first class citizen in the World Wide Web. Location will have granularity varying from a few centimeters to meters. How to query, monitor and update dataspace network layer translates to guarantees on the answer to a particular query, how to deal with the possible response implosion that will follow when in the number of objects becomes very large?

4. Applications

Each technology needs a "killer app". What is a killer app for dataspace? Sensor networks became recently one of the major research areas addressing networking issues for ad hoc routing, network self-configurations etc. Many of these applications are driven by military (like using sensors to monitor movement of heavy vehicles or intrusion detection). However, examples of utility of massive sensor networks in civilian sectors are rather unconvincing or simply contrived. Usual examples deal with massive number of temperature or air pollution sensors. Below, we present a number of more convincing applications, although some of them are admittedly quite futuristic.

.

4.1. Transportation Dataspace

The estimated cost of traffic congestion to the U.S. economy is around $100 billion a year. Without innovative and effective initiatives, traffic congestion in the U.S. will triple by 2005, with peak hour average speeds dropping from 35 mph to approximately 10 mph. Lower speed and stop and go conditions increase fuel consumption, which in turn increases the emissions. Congestion has also dramatic effects on the safety. It is well documented that accident rates triple under congestion conditions and 60% of urban congestion is caused by accidents.

The major roadblock faced by transportation is the investment needed for efficient and reliable collection and dissemination of traffic data such as:

1. Congestion information to drivers of commercial and non-commercial vehicles that can avoid congested areas by re-routing or by postponing their trips

2. Provide mobility solutions for fleets, emergency and time-sensitive service providers, and sales forces.

3. Diversion route information to travelers, virtual detour advice that varies in time depending on the traffic.

4.Road conditions such as ice, potholes, etc.

5. Complete network surveillance information to traffic engineers in charge of traffic management

Dataspace can provide a unique and robust solution to these important challenges.

Let us start with a futuristic scenario of "Driving in 2015" which illustrates how the concept of the dataspace will help commuters:

Driving in 2015 – a motivating fantasy

Jane is a devoted jazz fan. After work on Friday, she plans to meet a friend for an evening jazz concert in New Brunswick, New Jersey. Jane usually prefers to take River Road when she drives to New Brunswick, because it runs along the beautifully wooded Raritan River.

Unfortunately, it is a cold winter day in New Jersey, and River Road has a tendency to become icy or muddy during inclement weather. In fact, it has been snowing on and off all day, so parts of River Road - especially the bridges - have begun to ice up.

Within 200 yards of approaching the first of four small bridges on River Road, a display panel in the dashboard of Jane's car warns her of icy conditions ahead. This message was transmitted to her car by a local dataspace as she drove past it. The dataspace obtained this warning after being contacted by sensors in the tires of the last dozen drivers to cross the bridge. Heeding the warning, Jane slows significantly and successfully negotiates the icy bridge crossing. At the same time, sensors in her own tires confirm the icy conditions to a nearby information dataspace so that future drivers can be warned.

Increasingly discouraged by driving conditions as the weather worsens, Jane decides to request the current estimated driving time to New Brunswick. Using a voice interface in her car, she issues a query to her local dataspace, which in turn forwards the request to a regional information dispenser for all of Middlesex County. Prior to her query, the regional dispenser has been aggregating and averaging travel times recently reported by drivers along the same route. The response returns via the local dispenser, and Jane learns that, due to an accident which is currently clearing at the intersection of River Road and Route 18, drivers have been requiring an average of 25 minutes to reach New Brunswick, which is 10 minutes more than the usual time. Not wanting to get into an accident of her own, she proceeds a bit slower and more carefully than usual. As her journey continues, Jane crosses four more icy patches and one muddy patch; in each case, a message parked at a nearby dataspace by a previous driver's tire sensors had warned her of the impending conditions well ahead of time.

Five minutes prior to arriving in New Brunswick, Jane inquires about the parking situation at her destination. On Friday evenings, parking spaces tend to be scarce, and indeed that turns out to be the case this evening. A dataspace in New Brunswick informs her that there are 5 parking spots available on the upper deck of the George Street garage. It knows this, because magnetic sensors at the entrances and exits of the garage have been counting and storing the number of arriving and departing cars.

Before heading to George Street to park her car, Jane proceeds to the bus stop, where she plans to meet Harold, a friend coming from Metuchen by bus. From an information kiosk at the bus terminal, Jane determines that Harold's bus left nearby Edison just one minute ago.

The kiosk had just recently been updated with information by the dataspace in Edison, which was updated by a sensor in Harold's bus as it passed nearby. After a few more minutes, Harold's bus arrives and the two of them proceed together to George Street in order to park.

Not knowing the exact location of the building where the concert will be held, Jane isn't certain which direction to walk. Fortunately, one of the concert's organizers has parked a message with directions to the concert's location at a local dataspace. Using a handheld device, Jane issues a query to the dataspace using the search term ''jazz" and discovers the organizer's message. Jane and Harold proceed to the concert location and enjoy an evening of excellent music.

This scenario is not such a fantasy if one realizes that present-day cars, trucks collect tremendous amount of data that is currently of little or no use. Many cars are equipped with the so called data acquisition boards which are connected to a number of sensors such as GPS, accelerometer, brakes, etc. These sensors can collect time and location stamped information about location of a vehicle and events such as bumps and icy conditions detected by the sensors. Each vehicle can periodically “drop off” or “park” information about its travel log for the recently completed part of the trip. Such information can include the time to travel between point A and point B (road segment), bump/pothole detection, slippery/wet/icy rod conditions etc. This data can than be aggregated later to form a up to date local road report and disseminated to the local motorists.

This data acquisition and dissemination can be made anonymous without violation of privacy or security of individual vehicles. This works very well in rural environments and small towns and not just major metropolitan areas where special infrastructure exists such as the use of helicopters for generating traffic reports. Current approach of coarse-grain information restricted to major roads would simply not scale. The approach we are suggesting scales as it allows for consumers of information to be producers as well and follows the World Wide Web tradition of decentralization of information sources.

Individual cars could use standard interface of navigation systems maps that can be enhanced with aggregated real time traffic and road conditions reports. Aggregation could be performed at the network. One of the key challenges involves interpretation of sensor data – from a stream of bits collected by accelerometer to the bump/pothole detection. This is an important problem that involves a combination of machine learning, classification techniques for sensor calibration and sensory data interpretation.

4.2. Space capsule

The capability of every object (animate and inanimate) to continuously measure, record, and communicate values of its surroundings gives rise to a vision of a space capsule as opposed to a time capsule. A space capsule captures all “interesting” events at a given location. This can be constructed by a given individual, a mobile device (car, train, mobile terminal) which can be at a later time queried, mined, and correlated with other events or space capsules of other individuals or devices. Any information can be obtained by interrogating personal space capsules and can be indexed by location ranging from a very precise notion of location to a very coarse notion of location.

Location as opposed to time can also be easily revisited and one can then retrieve events that were recorded at that location in the past by simply visiting that location again. Should this location be visited infrequently, then location becomes almost a "key." The space capsule at a given location can also retrieve all previous timestamps of visit and retrieve data, such as files, voice recordings, pictures etc., which were created at that location. Revisiting physically the location causes the instant recall of the associated events/data that were produced there. Location can be used for producing an "instant replay" or for indexing of data - since human brain most often works this way, we may very well have a natural method of indexing information.

One can join two personal space capsules at common location and time to refer to joint meeting notes etc. Location can vary from location of the car to the location on a whiteboard where the "discussion took place" through drawings etc or where the voice file was recorded. For example you can find out about all movies which you saw at specific location (from location one can find the time when one were present there and then get the movie which played at that time); all lectures one attended (join the location information with the schedule of colloquia), all meetings which you missed in the seminar room; any public events you attended or missed in general etc.

Location information may be used to monitor correlation between location of an individual and some of his vital signs such as blood pressure and heartbeat rate if this is monitored. This would help in correlating these important vital signs with situations which given individual experiences during the day and which are much easier indexable by location than by exact time. Stretching this even further – space capsule for an individual can help him/her realise how much time they spend at work, in the gym, at home, in a restaurant etc. Same applies to the mobile devices and objects such as cars. The space capsule of a car can help establish how often the car was driven over local, bumpy roads as opposed to highways and how that affected the maintenance problems of the car

Acknowledgements: We owe it all to our students without whom much of the work would not be possible. We would like to thank our funding sources that include NSF, DARPA, and CISCO for supporting much of the Dataman research effort. We also thank Samir Goel for comments on this paper.

References

1. [vldb92] **Querying in highly mobile distributed environments,** Tomasz Imielinski, B. R. Badrinath, *In proceedings of the eighteenth VLDB*, , pages 41-52, August 1992.
2. [Sigir01] **Wireless graffiti and annotating physical space with digital information**, Tomasz Imielinski, Key note talk, In 24th International Conference of ACM/SIGIR, September 2001.
3. [Le01] **A Long-Term View of Short-Range Wireless,** David G. Leeper, *IEEE Computer Magazine*, Vol. 34, No. 6, pages 39-44, June 2001.
4. [Le01a] **Wireless data blaster,** David G.Leeper, *Scientific American*, May 2001.
5. [Ha68] **The Tragedy of the Commons**, Garrett Hardin, *Science*, 162(1968):1243-1248.
6. [Be99] **GSM Phase 2+ General Packet Radio Service GPRS: Architecture, Protocols, and Air Interface.** Christian Bettstetter, Hans-Jörg Vögel, and Jörg Eberspächer, *IEEE Comsoc*, Vol. 2. No. 3, 3rd quarter, 1999
7. [ImViBa97] **Data on Air: Organization and access**, T. Imielinski, S. Viswanathan, and B. R. Badrinath, IEEE Transactions on Knowledge and Data Engineering, Vol. 9, No. 3, May/June 1997, pp. 353-372.
8. [ViImBa94] **Energy Efficient Indexing on Air**, S. Viswanathan, T. Imielinski, B. R. Badrinath, In Proceedings of the ACM SIGMOD, Minneapolis, Minnesota, pages 25-36, May, 1994**.**
9. [ImViBa97] **Data on Air: Organization and Access**, Tomasz Imielinski and S. Viswanathan and B. R. Badrinath, In IEEE TKDE, vol. 9, No. 3, pages 353-372, 1997
10. [BaIm94] **Sleepers and workaholics: caching strategies in mobile environments**, Daniel Barbara and Tomasz Imielinski, In Proceedings of the ACM SIGMOD, Minneapolis, Minnesota, pages 1-12, May, 1994**.**
11. [ImVi96] **Wireless Publishing: Issues and Solutions;** T. Imielinski, S. Viswanathan, In Mobile Computing, Volume 353, Kluwer Publishers, January 1996.
12. [ImKo96] Mobile Computing, Edited by Tomasz Imielinski and Hank Korth, Vol. 353, Kluwer Publishers, January 1996.
13. [ImBa94] **Wireless mobile computing: Challenges in data management**, T. Imielinski and B. R. Badrinath, Communications of the ACM, , pages 19-27, October 1994.
14. [ImBa93] **Datamanagement for mobile computing**, T. Imielinski and B. R. Badrinath, SIGMOD RECORD, Vol. 22, No. 1, pages 34-39, March 1993.
15. [NiNa02] **Trajectory-based forwarding and its applications**, Dragos Niculescu and Badri Nath, Rutgers University, Technical Report, DCS-TR-488, May 2002
16. [NRC01] **Embedded Everywhere, A research agenda for networked systems of embedded computers,** NRC report, national academy press, 2001
17. [St02] **Position paper on monitoring application,** Michael Stonebraker**,** *NSF Workshop on context-aware mobile database management (CAMM)***,** January 2002.
18. [Bh01] **Bluetooth: Technology for Short-Range Wireless Apps**. Pravin Bhagwat. *IEEE Internet Computing, Vol. 5, No. 3, May-June 2001*
19. [Mu01] **Bluetooth demystified, Nathan J.** Muller, McGraw-Hill, 2001
20. [HiBo01] **Location Systems for Ubiquitous Computing**, Jeffrey Hightower and Gaetano Borriello, *Computer*, vol. 34, no. 8, pages 57-66, IEEE Computer Society Press, Aug. 2001.
21. [Hi02] **The Location Stack: A Layered Model for Location in Ubiquitous Computing**, Jeffrey Hightower, Barry Brumitt, and Gaetano Borriello, in *Proceedings of the 4th IEEE Workshop on Mobile Computing Systems & Applications (WMCSA 2002)*, June 2002
22. [ImGo00] **DataSpace -- querying and monitoring deeply networked collections in physical space**, T. Imielinski and S. Goel, IEEE Personal Communications, Special issue on "Networking the physical world", vol. 7, no. 5, pp. 4-9, October, 2000.
23. [GoIm01] **Prediction-based Monitoring in Sensor Networks: Taking Lessons from MPEG**, S. Goel and T. Imielinski, ACM Computer Communication Review, vol. 31, no. 5, October, 2001.
24. [ImGo99a] **Dataspace - querying and monitoring deeply networked collections of physical objects, Part I - Concepts and Architecture,** Tomasz Imielinski and Samir Goel, Technical Report DCS-TR-381, Rutgers University, July 1999.
25. [ImGo99b] **Dataspace - querying and monitoring deeply networked collections of physical objects, Part II - Protocol Details**. Tomasz Imielinski and Samir Goel, Technical Report DCS-TR-400, Rutgers University, July 1999.
26. [Bo00] **Querying the physical world**, Phillipe Bonnet, J. Gehrke, and P. Seshadri, IEEE Personal Communications, Special issue on

"Networking the physical world", vol. 7, no. 5, pp. 10-15, October, 2000.

27. [In00] **Directed Diffusion: A scalable and robust communication paradigm for sensor networks**, C. Intanagonwiwat, R. Govindan, and D. Estrin, Proc. of 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2000), Boston, August, 2000.
28. [Es00] **Embedding the Internet**, Deborah Estrin, Ramesh Govindan, and John Heidemann, Communications of the ACM, vol. 43, no.5, pp. 39-41, May, 2000.
29. [Bu00] **GPS-less low cost outdoor localization for very small devices,** Nirupama Bulusu, John Hieideman, and Deborah Estrin, *IEEE Personal Communicatons magazine,* Special issue on smart spaces and environments, October 2000.
30. [Ba00] **RADAR: An in-buliding RF-based user locatiob and tracking system,** Paramvir Bahl and Venkata N. padmanabhan, Proceedings of the IEEE Infocom 2000, Tel-Aviv, Israel, March 2000.
31. [Pr00] **The cricket location-support system,** N. B. Priyantha, A Chakraborty, H Balakrishnan, In Proceedings of the Sixth ACM Mobicom, August 2000.
32. [NiNa01] **Adhoc positioning system,** Dragos Niculescu and Badri Nath, In proceedings of GLOBECOM, November 2001.
33. [Hi00] **System Architecture Directions for Networked Sensors**, J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, Proc. of the 9th International Conference on Architectural Support for Programming Languages and Operating Systems, November, 2000.
34. [Ac96] **Disseminating updates on broadcast**, Swarup Acharya and Michael J. Franklin and Stanley Zdonik, The VLDB Journal, Pages 354-365, 1996.
35. [Ac95] **Broadcast Disks: Data Management for Asymmetric Communication Environments,** Swarup Acharya, Rafael Alonso, Michael Franklin, and Stanley Zdonik. . In Proceedings of ACM SIGMOD Conference, pages 199--210, 1995.
36. [Ja96] **Efficient and Flexible Location Management Techniques for Wireless Communication Systems**, J. Jannink, N. Shivakumar, D. C. Cox, and J. Widom, *In Proceedings of the Second Annual International Conference on Mobile Computing and Networking (MOBICOM'96),* pages 38-49, Rye NY, Nov 1996.
37. [Sh97] **Per-User Profile Replication in Mobile Environments: Algorithms, Analysis, and Simulation Results**, Narayanan Shivakumar and Jan Jannink and Jennifer Widom, , In Mobile Networks and Applications, Vol. 2, No. 2, Pages 129-140, 1997.
38. [LiTs98] **Location Tracking with Distributed HLRs and Pointer Forwarding**, Y. Lin and W. Tsai, IEEE Trans. Veh. Techn., 47(1):58--64, Jan. 1998.
39. [RoYa97] **Location Uncertainty in Mobile Networks: a theoretical framework** , C. Rose and R. Yates, IEEE Communications Magazine, 35(2), pp.94-101, Februrary 1997.
40. [Ba92] **Locating Strategies for Personal Communication Networks**, B. R. Badrinath, T. Imielinski, A. Virmani, In Workshop on Networking of Personal Communications Applications, IEEE Globecom, December, 1992
41. [PoKa00] **Wireless Integrated Network Sensors**, G.J. Pottie and W.J. Kaiser, Communications of the ACM, vol. 43, no.5, pp. 39-41, May, 2000.
42. [NaIm97] **Geographic addressing and routing**, Julio C. Navas and Tomasz Imielinski, In Proc. of the Third ACM/IEEE International Conference on MobiCom, Budapest, Hungary, pages 66-76, September 1997.
43. [Ji99] **Jini Architecture Specification**, Technical Specifications. Sun Microsystems Inc. http://www.sun.com/jini/specs/jini-spec.ps, January 1999.
44. [Pr01] **The Cricket Compass for Context-Aware Mobile Applications**, N. B. Priyantha, A. K. L. Miu, H. Balakrishnan, S. Teller, Proc. of 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2001), Rome, Italy, July 2001.
45. [Ha99] **The anatomy of a context aware application**, A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, Proc. of 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 1999), Seattle, pp. 59-68, August, 1999.
46. [BaFrCr93] **Core based trees (CBT): An architecture for scalable inter-domain multicast routing**, T. Ballardie, P. Francis, and J. Crowcroft.. In Proceedings of the SIGCOMM, 1993.
47. [De96] **The PIM architecture for wide-area multicast routing**, S. Deering et al., IEEE/ACM Transactions on Networking, 4(2):153-162, April 1996.