Integrated Data Management for Mobile Services in the Real World

C. Hage[1] C. S. Jensen[2] T. B. Pedersen[2] L. Speicys[2] I. Timko[2]

[1]Euman A/S
Vodroffsvej 7, DK-1900 Frederiksberg C
Denmark
ch@euman.com

[2]Department of Computer Science, Aalborg University
Fredrik Bajers Vej 7E, DK-9220 Aalborg Ø
Denmark
{csj, tbp, laurynas, timko}@cs.auc.dk

Abstract

Market research companies predict a huge market for services to be delivered to mobile users. Services include route guidance, point-of-interest search, metering services such as road pricing and parking payment, traffic monitoring, etc. We believe that no single such service will be the killer service, but that suites of integrated services are called for. Such integrated services reuse integrated content obtained from multiple content providers.

This paper describes concepts and techniques underlying the data management system deployed by a Danish mobile content integrator. While georeferencing of content is important, it is even more important to relate content to the transportation infrastructure. The data management system thus relies on several sophisticated, integrated representations of the infrastructure, each of which supports its own kind of use. The paper covers data modeling, querying, and update, as well as the applications using the system.

1 Introduction

Strategy Analytics, a leader in providing strategic and tactical support for business planners, recently concluded that: “Demand for mobile information services is skyrocketing and interest in coupling them with positioning technologies [is] at an all time high.” A USD 9 billion and a USD 7 billion annual revenue opportunity from location-based services is foreseen by 2005 in Western Europe and the USA, respectively [15]. Strategy Analytics expect that location technologies will augment existing wireless data applications as well as spawn a host of entirely new services, including alerts and ads, and personal location and guidance services.

The mobile services value chain involves a range of different services, including at least content provision, content integration, service development, service hosting, wireless communication provision, and billing. Following the maxim that “content is king,” we believe that the integration of content and the reuse of content across multiple services will be central to the cost-effective delivery of competitive mobile services as well as to the rapid development of new mobile services. Consequently, we see content integration as essential to the mobile services of the future.

The user’s location is central to many mobile services. This is so because the location is quite indicative of the user’s context. Thus, knowing the location is helpful in delivering the desired service while requiring less extensive interaction with the user. This is important. First, the interaction with the user is often constrained in comparison to a desktop computing scenario. Second, interaction is often a secondary activity, the primary activity being to, e.g., travel safely.

The transportation infrastructure is often essential to mobile service users. Indeed, a general-purpose foundation for the delivery of location-based services requires that multiple, integrated representations of the infrastructure are available. For example, a graph-based representation is needed for route planning. In addition, the infrastructure must be geo-referenced, i.e., the geographical coordinates of roads and intersections must be captured.

Next, content (also termed business data) must be georeferenced and must also be positioned with respect to the infrastructure. For example, points of interest must be positioned in the infrastructure so that guidance services can

---

Proceedings of the 29th VLDB Conference,
Berlin, Germany, 2003

determine appropriate routes to these.

The paper describes in detail Danish company Euman A/S's [7] content integration and service delivery system. This system captures multiple, integrated representations of the transportation infrastructure and positions content with respect to this infrastructure. This system offers a general-purpose and flexible foundation for the rapid development and delivery of diverse, integrated mobile services. Initial services based on this system are available to consumers in Denmark, and advanced services are being prototyped and tested.

The paper's description of the content integration and service delivery system focuses on the data modeling employed by the system, but update and querying are also covered. The paper additionally offers the reader insight into the real-world requirements posed to a system that integrates mobile content.

The database schema shown in the paper represents the core of the real schema. The real schema includes more attributes than those shown, in addition to some 700 other tables. The database instance shown is extracted from the real database instance, although a single table has been populated with generated data, for conciseness and simplicity. In spite of the necessary conciseness, it is our hope that this paper will shed light on the cross-disciplinary application domain, will demonstrate some of the complexity of the data management problem addressed, and will inform future research.

Past related work in the scientific domain of computer science generally makes simple assumptions about the problem setting. First, much work assumes that mobile objects and content are points embedded in, typically, two-dimensional Euclidean space. The efficient support for different types of range queries, nearest neighbor queries, and reverse nearest neighbor queries have been explored in this context (e.g., [2, 10, 17, 18, 20]).

The transportation infrastructure is not taken into account. As a result, the notions of proximity and distance used are inappropriate to many services. Further, only limited services can be supported—guidance services cannot be supported.

Second, other scientific work has considered problems where the infrastructure is central. The shortest path problem [1, 3] is a good example. Here, the infrastructure is represented as a graph. The resulting solutions do not take into account Euclidean distances and do not support well the geographically-based integration and querying of content. Critics of graph representations [8, 9] argue that they fail to capture advanced real-world properties of transportation infrastructures, e.g., link characteristics that change between vertices.

These past works must be integrated in order to provide a general-purpose foundation for the delivery of mobile services. The present paper is the first paper known to the authors that takes steps towards such an integration. In doing so, the paper contributes to making the advances from the scientific domain relevant in practice.

In the industrial domain, linear referencing [16] has been used quite widely for the capture of content located along linear elements (e.g., routes) in transportation infrastructures. For example, Oracle Spatial [12] offers support for linear referencing. In addition, a generic data model, or ER diagram, has been recommended for the capture of different aspects of entire transportation infrastructures and related content [13], and a number of variants of this model have been reported in the literature (see, e.g., [4,6,11,14,19,22]).

The data model underlying the system presented in this paper employs linear referencing, but improves on other data models in several respects. Perhaps most notably, it is the only model that integrates different representations of a transportation infrastructure via a purely internal model of the infrastructure. In a sense, this approach is a "geographical generalization" of the use of surrogate keys for the management of business data. Use of the internal infrastructure model simplifies data management in much the same way that does the use of keys that does not carry any meaning outside the system.

The paper is structured as follows. The next section describes the functionality expected from a mobile services data management system, and it illustrates the nature of the related content and the transportation infrastructure itself. Section 3 describes the overall architecture of the system. Section 4 offers a fairly detailed description of the core part of the system's data model and concepts underlying its design. The next two sections cover update and querying, respectively. Finally, Section 7 summarizes the paper and outlines some of the directions in which the system is expected to be extended in the future.

2 Case Study and Requirements

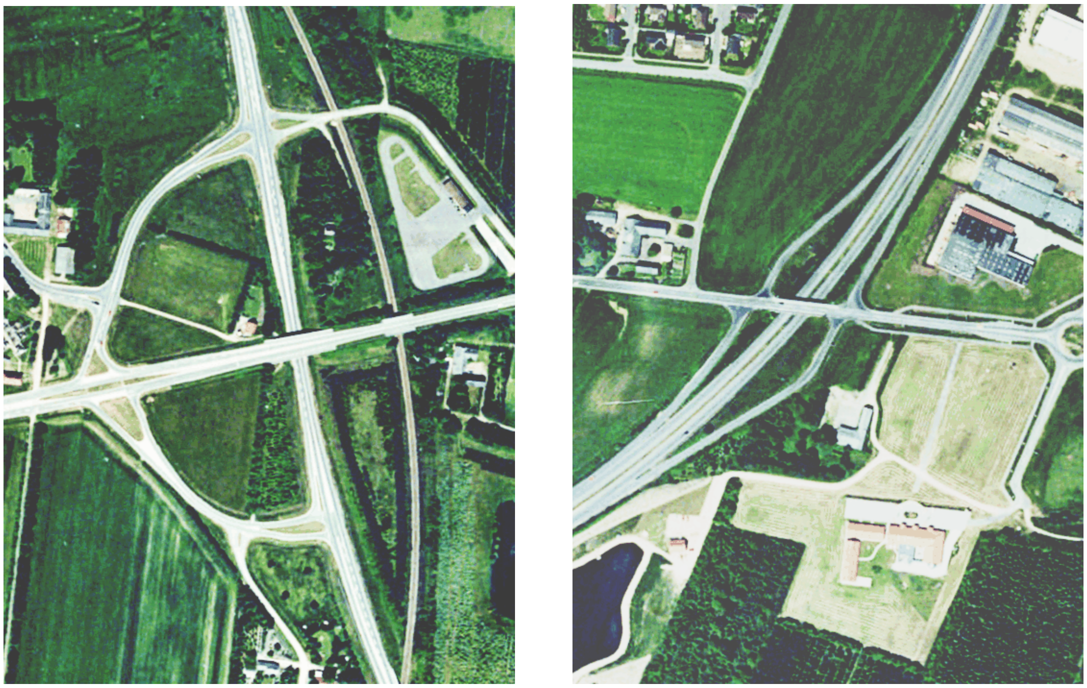
This section aims to illustrate the data management challenges posed by mobile services. More specifically, we consider the real-world setting of mobile services. We initially consider the types of queries that are needed to support typical services.

2.1 Location-Based Queries

A simple type of query in mobile services computes the distance or expected travel time from a mobile user's current position to a destination or point of interest, such as a particular art gallery or tourist attraction.

Another type of query concerns advanced route planning, where the service user wants to retrieve the route to a certain point of interest, in the shortest time (i.e., taking both distance and expected travel speeds into account) while passing one or more points of interest enroute.

Yet another type of query retrieves the "nearest" point of interest, such as a particular type of store or gas station. The term "nearest" may be given several meanings. For example, it may denote the shortest Euclidean distance, or it may denote the shortest travel time along the road network. The distance to the point of interest may be in relation to the current position of the service user, or it may be in relation to the remainder of the route on which the user is traveling.



(a) First Intersection (b) Second Intersection

Figure 1: Aerial Photos of Road Intersections

Queries such as these require different representations of the underlying data. One may initially distinguish among two types of data that must be available for querying, the infrastructure and the “remaining” data. The former encompasses the geographical space and the transportation infrastructure according to which the remaining data is positioned, either directly or indirectly. The remaining data is sometimes termed business data or content. Examples include the points of interest mentioned in the queries above, but this type of data is quite open-ended. For example, for an art gallery the content encompasses information about the current exhibition, the associated artists, and the artists’ other works and exhibitions.

To position a service user, for whom we have the geographical coordinates from a GPS receiver, wrt. the infrastructure, it is necessary to geo-reference the infrastructure. Put differently, we need the coordinates of the roads. To perform route planning, a graph-like representation of the transportation infrastructure is needed.

Some content, e.g., accident data, is traditionally positioned based on mile-posts or so-called known makers. To make such content available to queries, a representation of the infrastructure based on known markers is also required.

Queries inherently involve content: distances, speed limits, estimated speeds, sights, attractions, destinations, etc. In order to support the queries listed, the content must be geo-positioned as well as positioned wrt. the infrastructure.

Further, the content must be accessible via different representations of the infrastructure. The implication is that the infrastructure representations must be interrelated, meaning that it must be possible to translate from one representation to another.

2.2 Content

Content generally falls into one of two categories. First, *point data* concerns entities that are located at a specific geographic location and have no relevant spatial extent. This type of data is attached to specific points in the transportation infrastructure. Examples of point data include traffic accidents, museums, gas stations, and hotels. Second, *interval data* concern data that are considered to relate to a *part* of road and are thus attached to intervals of given roads.

Interval data can be categorized according to two orthogonal characteristics: (1) *overlapping versus non-overlapping* and (2) *covering versus non-covering*. Specifically, if it is possible for more than one piece of content of the same type to be attached to the same (sub-)interval of a road, that type of content is overlapping; otherwise, it is non-overlapping. Next, a type of interval content is covering if there is no part of the infrastructure that does not have at least one piece of content of that type attached to it; otherwise, it is non-covering. We say that a type of interval content is *partitioning* if it is non-overlapping and covering.

Partitioning content includes speed limits and road surface type. Non-overlapping, non-covering content includes u-turn restrictions and road constructions, as well as more temporary phenomena such as traffic congestion and jams. Examples of overlapping, non-covering content includes tourist sights. A scenic mountain top and a castle may be visible from overlapping stretches of the same road. Other part of roads have no sights. Another example is warning signs. Overlapping, covering content include service availabilities, e.g., a car repair service may be available from some service provider anywhere, and several repair service providers may be available in areas.

2.3 Transportation Infrastructure

Section 4 describes in detail how the transportation infrastructure and content are represented in the content integration system. Here, we simply describe the actual infrastructure that is to be represented as discussed earlier in this section. To be specific, we consider three consecutive "intersections" along a single road. Aerial photos of the first two of these are given in Figure 1(a)–(b). In these photos, the road we consider first stretches from West to East (a), then bends and goes from South-West to North-East (b). We describe each intersection in turn.

While our road is generally a bidirectional highway with one lane in each direction and no median strip dividing the traffic, the first intersection, in Figure 1(a), introduces a median and includes two bidirectional exit and entry roads. Major concerns underlying this design are those of safety and ease of traffic flow. Vehicles traveling East, i.e., using the right lane of the highway, must use the first road to the right for exiting. A short deceleration lane is available. The second road connected to the right side of the highway is used by vehicles, originating from the crossing road that wish to travel East on the highway. A short acceleration lane is available. A similar arrangement applies to the highway's left lane.

At the point of the second intersection, in Figure 1(b), the highway has two lanes in each direction, a median, and four unidirectional exit and entry lanes. This intersection is safer for vehicles on the highway than the previous one. Entry and exit lanes dedicated to acceleration and deceleration provide higher safety for vehicles on the highway. Specifically, a vehicle traveling North-East, i.e., using the right lane of the highway, can decelerate in the long exit lane, while North-East bound vehicles must enter and can accelerate via the long right entry lane. A corresponding arrangement applies to the left lane of the highway.

The third intersection (no photo shown) is a five-road rotary that connects our road with another major road and a small road that leads to a developed area. The other major road has a bicycle path along it. It is possible to enter the rotary from any of the five roads, and to exit it onto any of the roads. Only right turns are possible.

It should be clear from this description and the description of the need for multiple representations of a transportation infrastructure that a transportation infrastructure is not just a mathematical graph. While some aspects may be described as a directed, non-planar graph, other aspects are left unaccounted for by such a simple representation, e.g., the geographical coordinates of the roads and their intersections.

3 System Architecture

An overview of the system architecture can be seen in Figure 2. We now describe the individual components, starting at the top.

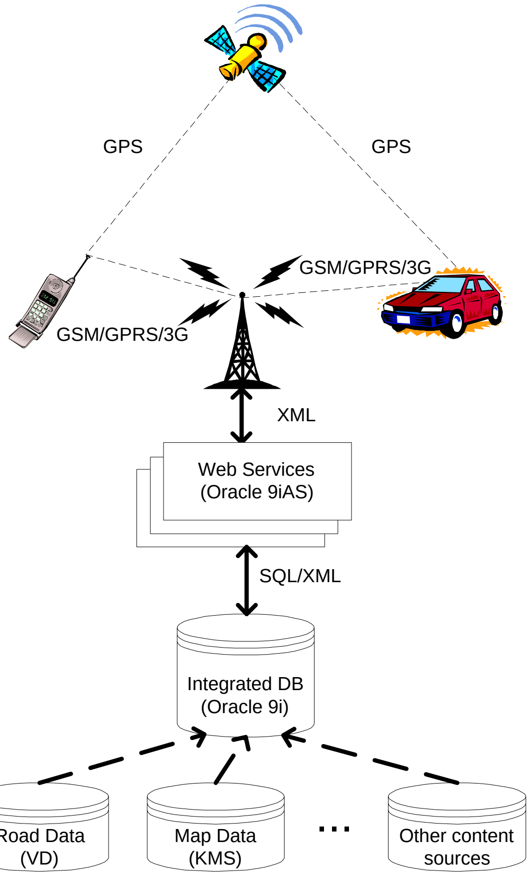


Figure 2: System Architecture

The two most common types of clients are advanced Java-enabled mobile phones/PDAs such as the Nokia 9210 Communicator and the Nokia 7650 (for person-related services) and cars with on-board, on-line computers (for road traffic services, etc.). The clients receive their positional information from GPS satellites using their associated GPS receivers. The clients use a wireless WAN network to communicate with the other system components. Currently, most communication uses GSM-based technologies such as High-Speed Data and SMS. However, packet-based 2.5G protocols such as GPRS are gaining popularity, and 3G UMTS communication is in the horizon. Other types of clients are also possible, e.g., PCs with browsers (communicating over a fixed line network) are used for planning purposes.

The clients use data and functionality provided by a number of *web services*, e.g., route-planning services or map services. The web services are based on the W3C Web Service standard proposal [21]. The clients send web service requests to the web services and get the desired information back in web service responses. All requests and

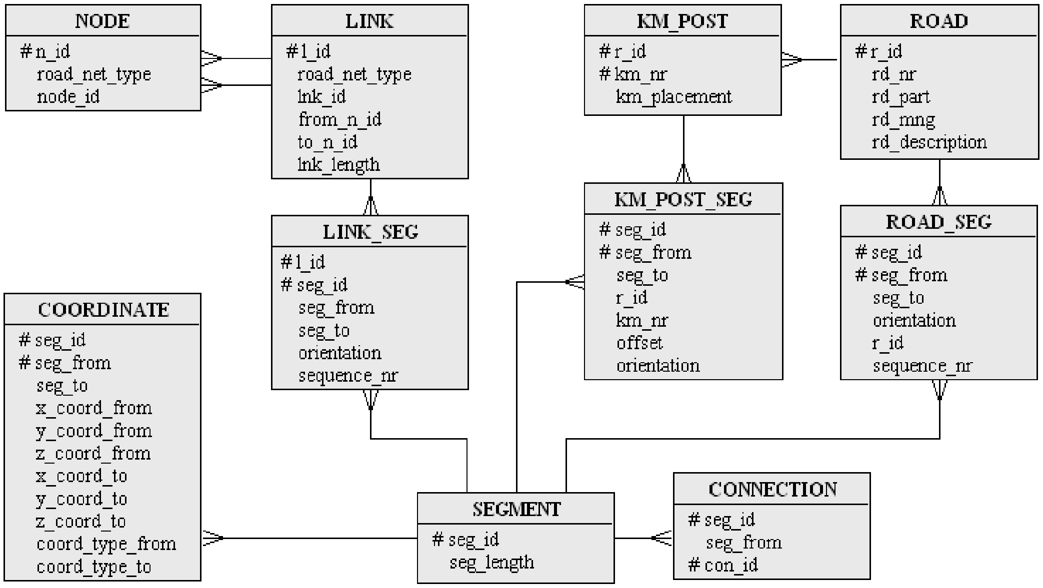


Figure 3: The Core Data Model Entities

responses are encoded using XML-based formats. The web services run on the Oracle 9i Application Server.

The web services get their data from the Integrated DB (IDB) that implements the data model described in Section 4. All data describing infrastructure and content attached to the infrastructure is stored in the IDB, i.e., a “data warehouse”-like approach is employed. The IDB runs on an Oracle 9i RDBMS. The web services issue SQL requests to the DB, which then returns relational and/or XML data.

Finally, the IDB is fed from a number of sources. The two most important ones are the transportation infrastructure data, provided by Vejdirektoratet (the Danish Road Directorate), and map data, provided by Kort og Matrikelstyrelsen (the Danish Map and Cadastre Agency). In addition to these, a number of data providers supply the content that is attached to the infrastructure.

4 Data Modeling

The data model underlying the system provides several external, user-accessible, representations of the transportation infrastructure, namely the *kilometer-post*, *link-node*, and *geographic* representations. The external representations are connected by an internal *segment* representation. The core of the data model described in this paper is given by the diagram in Figure 3. Condensed tables that give example instances for this diagram are shown in Figure 7. As a precursor to exploring the representations in some detail, we initially consider their uses.

4.1 Overview of Infrastructure Representations

The kilometer-post representation (the most commonly used type of known-marker representation) is used for road administration. It is convenient for relating a physical location to a location stored in a database and vice versa. Location is expressed in terms of the road, the distance marker on the road (e.g., kilometer post), and the offset from the distance marker. The representation is used by administrative authorities for collecting and utilizing data on field conditions, e.g., entering content into the system. Primitive technological means, such as a simple measuring device and a map and a ruler, suffice for identifying a position on a road, rendering the use of the representation cost effective and thus practical.

The link-node representation is based on the concepts of undirected and directed mathematical graphs. A node is a place with a significant change of traffic properties, e.g., a road intersection. A link is a route that connects two nodes. Such a representation abstracts away geographical detail, but at the same time preserves the topology of the transportation infrastructure. For this reason, link-node representations are suitable for tasks such as traffic and route planning. The former task refers to (re)designing road networks taking traffic flows into consideration. In this case, undirected graphs are sufficient. The latter task refers to finding traversable routes in road networks that satisfy certain criteria. Directed graphs that capture traffic directions are appropriate for this task.

The geographical representation captures the geographical coordinates of the transportation infrastructure. The coordinate representation enables users to directly reference location rather than measure distances along roads from certain locations, such as kilometer posts. Additionally, the representation is used by geography-related applications, such as cartographic systems or certain GIS systems, that operate on coordinate data.

The segment representation models an infrastructure as a collection of segments that intersect at connections (locations where there is an exchange of traffic). This representation preserves the network topology and captures the

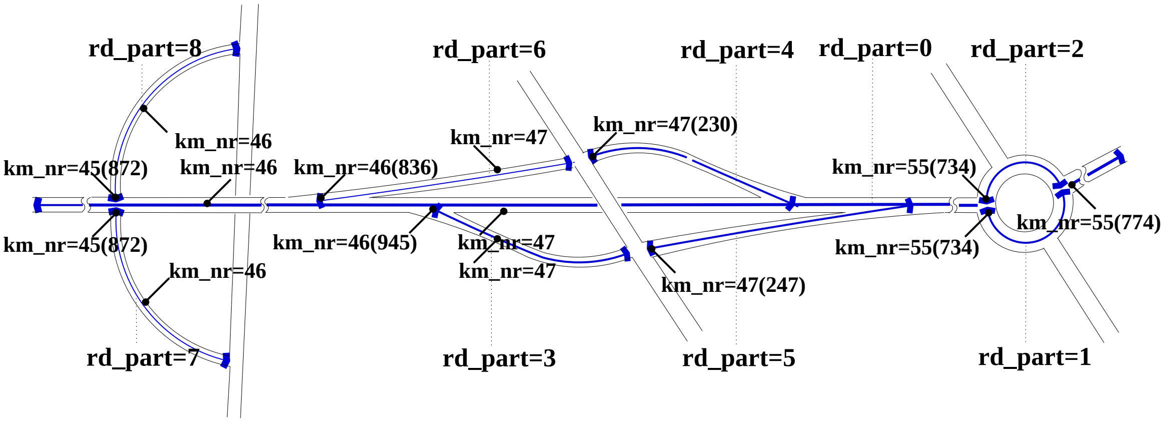


Figure 4: Kilometer-Post Infrastructure Representation Instance

complete set of roadways. The position of any content (e.g., speed limits, accident reports) references directly segments. In addition, the external representations of the infrastructure are mapped to the segment representation in a way that establishes one-to-one mappings between all the representations, i.e., the segment representation is the *integrator* of the different representations. For this reason, the segment representation is used by content integrators for efficient content position maintenance and translation between the external representations. The segment representation is purely *internal* to the system, which provides a number of benefits, as described in Section 4.4.

4.2 Kilometer-Post Representation

The kilometer-post representation is captured in the data model by tables ROAD and KM_POST; see Figures 3 and 7.

Table ROAD captures the administrative identification of roads. There is no single, commonly accepted road identification system. Systems tend to vary between countries, between counties of the same country, and between municipalities of the same county. Roads of different significance are managed at the state, county, and municipality levels. Our data model primarily concerns state and county roads.

The administrative identification consists of the road number, the road part, and the authorities responsible for the management of the road part, captured by attributes *rd_nr*, *rd_part*, and *rd_mng*, respectively. For example, our case captures parts of four roads that are managed by the authority 55. The main road we consider consists of parts numbered 0–8 (see Figure 7).

While road numbers and maintenance authorities partition the road infrastructure, road parts partition a single road. Road parts represent separate system lines of engineering structures that, administratively, constitute a single road. Roughly speaking, a single road part consists of lanes of a road based on the same roadbed. For example, in the second intersection of our case (Figure 1(b)), the main highway lanes reside on the same roadbed. Thus, although being divided by a median strip, they belong to the same system line and are modeled by a single road part (part 0; see Figure 4). On the other hand, exits and entrances to the highway belong to different system lines and are modeled by parts 3–6, 7, and 8, as defined by administratively determined part identification rules [5]. Additionally, lanes belong to different system lines if they are separated by significant dividers, such as a protective fence separating lanes or a circle-strip of a rotary.

Attribute *rd_description* captures user-friendly names for roads. For example, the name for part 0 of our main road consists of the names of the two towns it connects: “Korskro-Give.” Additional descriptions, denoting the type of a road part, may be captured for parts 1–8.

Table KM_POST captures information on road distance markers. A marker captures a full kilometer distance, or number (attribute *km_nr*), from the start of a certain road (attribute *r_id*). Markers may be located physically (observable) on roads, or may be imaginary. For example, markers 46 and 47 in Figure 4 represent physical road markers, while markers with residual parts, enclosed in parentheses, mark beginnings of road parts and are imaginary (except for the marker 55(774), which does not represent a separate marker, as discussed in Section 4.6). The residuals indicate the meter offset from the full kilometer, and that they are not captured in table KM_POST (offsets are captured in table ROAD_SEG). Attribute *km_placement* may be used for indicating the offset of the position of the physical marker from its logical position, e.g., when the logical marker position coincides with an intersection so that the physical marker cannot be placed there.

4.3 Link-Node Representation

The link-node representation of the transportation infrastructure defined by our schema is a collection of nodes (table NODE) connected by directed links (table LINK), i.e., a directed graph (see Figure 5). We do not consider undirected graphs here.

A record in table NODE describes a node. Nodes belong to one of several different road network types, e.g., to models of the same infrastructure, but with different resolutions. For example, for a certain area, there might be two link-node networks, a very fine-grained one used for detailed route planning, and a less detailed one used for higher-level

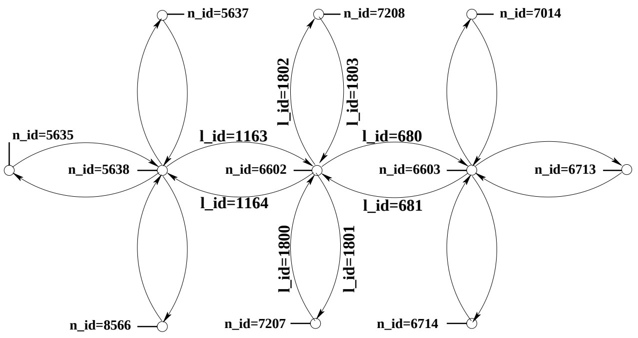


Figure 5: Link-Node Representation Instance

route or traffic planning. Conceptually, a node is identified by a pair of attributes (*road_net_type*, *node_id*), i.e., of a unique network ID and a unique node ID with the scope of the network. In order to simplify referencing from other tables and make query processing more efficient, each node is assigned a globally unique ID *n_id*, which is the primary key of the table.

A record in table LINK describes a link. Similarly to nodes, links belong to exactly one of a number of different road network types. Thus, links have attributes analogous to those of nodes, i.e., the pair (*road_net_type*, *lnk_id*) and *l_id*. Again, *l_id* is the primary key of table LINK. Moreover, each link has a start node *from_n_id* and an end node *to_n_id*. Attributes *from_n_id* and *to_n_id* are foreign keys, referencing *n_id* in table NODE. The road network type of a link is given by the road network type of its nodes. Further, each link has length *lnk_length*.

Sample tables NODE and LINK are presented in Figure 7. The data covers only one model of the infrastructure. For this reason, for all links and nodes, the *road_net_type* value is equal to 1, the *lnk_id* value is equal to the *l_id* value, and the *node_id* value is equal to the *n_id* value. These values are not shown because they are not interesting. Figure 5 presents the sample link-node data in graphical form.

The specific link-node data models the case study from Section 2 in a way that is appropriate for high-level route planning. The links represent the routes, not individual roads, e.g., links 1163 and 1164 represent the forward and backward routes between the first and second intersection, respectively. For this reason, the complex intersections, i.e., the two over-passes in Figures 1(a) and (b) and the rotary are each reduced to a single node, i.e., to the nodes 5638, 6602, and 6603, respectively. In our case, the *lnk_length* value for each link is approximately equal to the length of the corresponding route in meters. Thus, *lnk_length* values are equal for a pair of oppositely directed links. In general, *lnk_length* may be given more complex semantics, e.g., the minimum travel time that is needed to traverse the route.

Directed links in our representation, i.e., in representations suitable for route planning, indicate that node *to_n_id* can be reached from node *from_n_id*, i.e., a two-directional route is represented by a pair of oppositely directed links. Although our schema explicitly defines directed links, the same schema can be used for undirected links.

4.4 Segment Representation

The segment representation of the infrastructure defined by our schema is a collection of segments (table SEGMENT) that intersect at connections (table CONNECTION) This representation is illustrated graphically in Figure 6.

A record in table SEGMENT describes a segment. Each segment has a unique ID *seg_id* and the length *seg_length*. Attribute *seg_id* is the primary key of the table.

A record in table CONNECTION describes the fact that a segment *seg_id* intersects with a connection point identified by *con_id*. The intersection of the segment with the connection point occurs at *seg_from* units from the start of the segment. The pair (*seg_id*, *con_id*) constitutes the primary key. Finally, *seg_id* is a foreign key that references the attribute of table SEGMENT with the same name.

Sample tables SEGMENT and CONNECTION are presented in Figure 7. In the tables, the *seg_length* value for each segment is equal to the length of the corresponding road section in meters. Figure 6 presents the sample segment and connection data in graphical form.

The sample data models the case study from Section 2 in a way that preserves the network topology at a low level of detail. This level is necessary to accurately translate between the external representations. Each segment generally represents a road section in such a way that the segment is as long as possible, while preserving the network topology. For example, segment 893 corresponds to the "long" main road and is 78.326 meters long.

In order to preserve the topology, the additional segment 3522, which is only 62 meters long, is assigned to the "short" bottom semi-circumference of the rotary in our case study, which connects two disjoint sections of the main road. Connections are placed at road intersections. Again, this is necessary to preserve the topology.

Another constraint imposed on the process of creating segments is that the segments should partition the road network. The reasons why long segments are preferred is that these lead to a more compact segment representation and, more importantly, a more compact and thus update-efficient representation of the associated content. We revisit this in Section 5.

Note that some sections of a segment may not correspond to any roads. This is the case for the section of segment 893 that stretches between the connections 5387 and 5389 (more on this in Section 5).

4.5 Geographical Representation

The geographical representation is used for geo-referencing the road infrastructure and thereby the road-related content. Specifically, the geographical coordinates of all segments are captured by a table COORDINATE (an example instance of the table is omitted due to space constraints) that references table SEGMENT.

Rows of table COORDINATE contain pairs of three-dimensional points. The first point in a row is the building

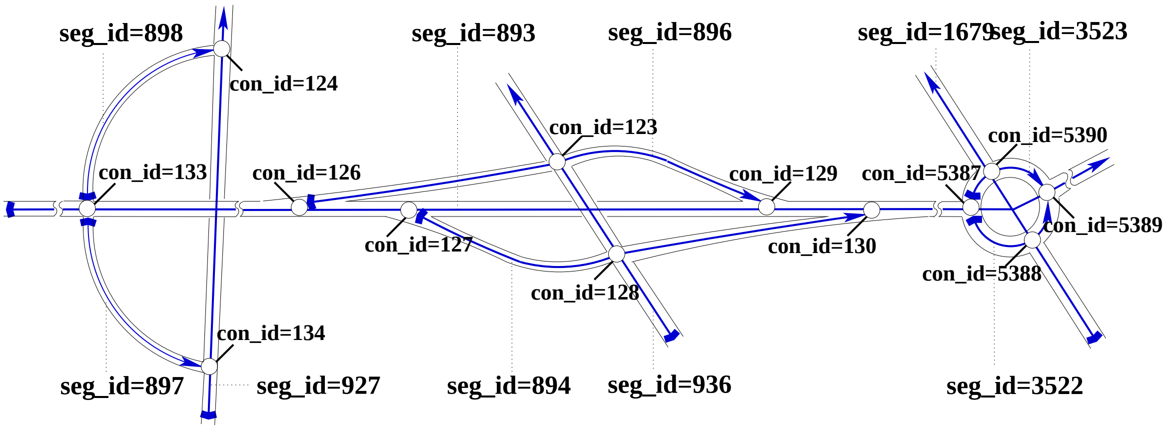


Figure 6: Segment-Based Infrastructure Representation Instance

block of the representation. Its coordinates *x_coord_from*, *y_coord_from*, *z_coord_from* capture a point on the centerline of a road, e.g., a point on the center of the separating strip on our road in Figure 1(b).

Several levels of detail are used for the geographical representation of segments. In the schema, this property of the first point of a row is captured by attribute *coord_type_from*. Levels of detail refer to different scales of the geographical representation. The lowest level of detail provides the coarsest representation of a segment. Higher levels of detail capture the location of a segment in more detail, by introducing additional points. This arrangement implies that in order to obtain a geographical representation of a segment at a certain level of detail, one has to select points with this or a lower level. As a result, the sequence of all points representing a segment at the most detailed level consists of points from all levels of detail.

The second point of a row, captured by *x_coord_to*, *y_coord_to*, *z_coord_to*, and *coord_type_to*, is used to enhance the efficiency of query processing. This point is the point following the first point of the tuple, in the sequence of all points representing the segment. Note that the second point can be of a different detail level, lower or higher, than that of the first point. The benefit of this is that a road is partitioned into small sections, each covered by a tuple. Whenever there is a need to calculate a geographical location of a point located some distance after the start of a segment, only the one row that covers the section is needed, not two rows.

The remaining attributes of a row, *seg_id, seg_from* and *seg_to*, map the two points of the tuple to the segment representation. They indicate on which segment and after which distances from the start of the segment these are located. Geo-referencing of content is provided only through the mapping of content to segments.

4.6 Interrelating the Representations

As pointed out earlier, the provision of integrated mobile services requires the ability to translate among the different external infrastructure representations. This is achieved in the data model by connecting the kilometer-post representation and the link-node representation to the segment representation, which is already integrated with the geographical representation (see Section 4.5). In our schema, these connections are achieved via tables KM_POST_SEG and ROAD_SEG for the kilometer-post representation and via table LINK_SEG, for the link-node representation.

Kilometer-post integration As for the kilometer-post representation, a row in table ROAD_SEG relates (a section of) a road part *r_id* to a part of a segment *seg_id*. The (section of the) road part positioned by a row corresponds to the section of the related segment with end points at *seg_from* and *seg_to* units from the start of the segment. The attribute *orientation* indicates whether or not the directions of linear referencing along the segment and along the road part coincide. Attributes *rd_id* and *seg_id* are foreign keys that reference the primary keys of tables ROAD and SEGMENT, respectively. Further, since road parts do not overlap, the pair (*seg_id*, *seg_from*) is the primary key of the table. Finally, *sequence_nr* is an ordinal number of the road part section that is used to conveniently distinguish among different sections of the same road part and to “reconstruct” road parts from a collection of segment sections.

A kilometer post is used as a linear reference point for a certain section of a road part, termed *usage scope*. A record in table KM_POST_SEG positions (a part of) the usage scope of a kilometer post (*rd_id*, *km_nr*) within a segment *seg_id*. The attribute *offset* indicates the position of (the part of) the usage scope as distance in meters from the kilometer *km_nr* measured in the direction of linear referencing. The other attributes have the same semantics as do their counterparts in table ROAD_SEG. The pair (*rd_id*, *km_nr*) and *seg_id* are the foreign keys that reference the primary keys in tables KM_POST and SEGMENT, respectively. Finally, since usage scopes do not overlap, the pair (*seg_id*, *seg_from*) is the primary key of the table.

The sample tables ROAD_SEG and KM_POST_SEG in Figure 7 position the road parts and the kilometer posts from tables ROAD and KM_POST (see Figure 4 as well) with respect to the segment representation (see tables SEGMENT and CONNECTION as well as Figure 6). In our case, for all the road parts, kilometer posts, and segments, the directions of linear referencing coincide, i.e., for all the records, the *orientation* value is equal to 1. These values are thus not shown. We also omit *sequence_nr* values in the table ROAD_SEG. The sample data illustrates the interesting aspects of the schema of tables ROAD_SEG and KM_POST. For example, in order to properly translate between representations of rotaries, the schema must allow to map road part sections, but not whole road parts, to segment sections (see lines 1 and 2). The same observation can be made for the mapping of usage scopes.

Note that table KM_POST_SEG alone fully defines the relation between the kilometer-post and other representations. Table ROAD_SEG has been included for query efficiency—it contains redundant information.

Link-node integration As for the link-node representation, a record in table LINK_SEG positions (a section of) a link *l_id* within a segment *seg_id*. The attribute *orientation* indicates whether the directions of linear referencing along the segment and of the link coincide. The other attributes have the same semantics as do their counterparts in table ROAD_SEG. Attributes *l_id* and *seg_id* are the foreign keys that reference the primary keys in tables LINK and SEGMENT, respectively. Further, since links "overlap," e.g., different links may belong to different models of the same infrastructure, the pair (*l_id*, *seg_id*) is the primary key of the table. On the other hand, since links model continuous routes, i.e., no "breaks" in links are allowed, *seg_from* is not in the primary key.

Table LINK_SEG in Figure 7 positions the links from the sample table LINK (see Figure 5 as well) with respect to the segment representation of the infrastructure from our case study. In our case, the directions of linear referencing along segments and of links do not necessarily coincide, i.e., for some records, the *orientation* value is 1 and for some, it is -1. These values are not shown because they are easy to determine. Again, the values of the *sequence_nr* attribute are also not shown because they are not interesting.

4.7 Integration of Content

In our model, each type of content is allocated a separate descriptive table. For example, tables ACCIDENT and SERVICE (not shown due to space constraints) describe instances of road accidents and car repair service providers, respectively. A row in tables ACCIDENT or SERVICE assigns an ID *id* for each accident or service provider respectively. Moreover, each row includes descriptive attributes. For example, for an accident the number of cars involved may be included. Further, each type of content is associated with a table that positions the content with respect to the infrastructure in the segment representation, e.g, table ACCIDENT_SEG (not shown) captures accident position and SERVICE_SEG (not shown) captures service availability ranges.

The principles of positioning interval data with respect to segments are described in Section 4.6. In particular, the tables for interval content have schemas similar to those of tables ROAD_SEG and LINK_SEG. For example, table SERVICE_SEG has the attributes *id*, *seg_id*, *seg_from*, and *seg_to*, possibly in addition to other attributes that characterize the positioning. The first and second attributes are the foreign keys that reference tables SERVICE and SEGMENT, respectively. The positioning of point data is analogous to the positioning of connections (see Section 4.4), e.g., compared to table SERVICE_SEG, table ACCIDENT_SEG has an attribute *seg_pos* instead of the pair (*seg_from*, *seg_to*).

Primary keys are defined depending on characteristics of the content. For example, since service availability content is overlapping, the triple (*seg_id*, *seg_from*, *id*) is the primary key of table SERVICE_SEG. Analogously, several accidents may happen at the same point, so the triple (*seg_id*, *seg_pos*, *id*) is the primary key of table ACCIDENT_SEG.

The same content must be accessible via different infrastructure representations. Given a type of content, our system includes a view for each necessary external representation. For example, accident data typically enters the database through the kilometer-post representation. This means that there exists a view ACCIDENT_KM_POST that is defined as a join of tables ACCIDENT_SEG and KM_POST_SEG on segments. As another example, it may be useful to maintain the number of accidents for each link (e.g., to find safe routes). This may be captured in a view ACCIDENT_LINK that is defined as a join of tables ACCIDENT_SEG and LINK_SEG followed by aggregation by each link.

5 Update

The model based on segments can be used to effectively accommodate updates of the transportation infrastructure and the content attached to it.

The key distinguishing feature of segments is that they are entirely *internal* to the system. Segments can be seen as a special kind of the linear elements (LEs) used in linear referencing models [16]. However, in other systems and models, *external* entities, most often *routes*, are used as the LEs, which causes problems when these external entities change. For instance, if a route changes, all the related content has to be updated. In contrast, segments are independent of external entities and are thus a far more stable concept onto which content can be attached. This arrangement reduces the amount of updates needed when external entities change.

Three types of infrastructure updates are of interest: transfer of road authority, road alterations, and roads under planning/construction. In Denmark, the road authority (the government body administering the road) is either the state (mostly interregional roads), the county (mostly intraregional roads), or the municipality (local roads and streets).

In some cases, authority of a road is transferred, e.g., from county to state if the road is upgraded to become an interregional road. The systems used for road identification, i.e., road numbers and road codes, differ among the different road authorities, including among counties and among municipalities. For example, state roads use a different system than county roads, and different counties use different systems. This means that road keys must be updated where ever they occur. In Euman's system, this is a relatively easy task, since the segments (and thus also the content attached to them) are unaffected by this change. Only the kilometer-post representation of particular roads must be updated. In systems that use the kilometer-post representation directly as an internal infrastructure representation, this update becomes a difficult task because the road keys are propagated throughout the system, in both infrastructure and content data.

When a road has significant alterations, other challenges occur. For example, a crossroads may be replaced by a (large) rotary, which was how the rotary in our case study arose. In this case, two pieces of road (the innermost "cross" in the crossroads) *disappear* in the real world, and are replaced by four new pieces of road, meaning that the road becomes a little longer at this point. If content is attached directly to the roads themselves, e.g., at certain distances from kilometer posts, the posts after the crossroads must now be relocated to reflect the new reality, which triggers updates of the content positions. In contrast, Euman's system allows the major underlying segments of the crossing roads, i.e., segments 893 and 1679 from Figure 6, to remain the same, avoiding any update of attached content. Two (small) new segments, 3523 and 3522, are inserted to model the rotary.

Finally, a road being planned does not at first have a road number/code. When construction starts, the road is assigned a temporary number/code, which is replaced by the final number/code when the road is opened to traffic. Handling this process in Euman's system is unproblematic because the segment(s) representing the new road are created when planning starts, i.e., before the road exists physically. Already at this point, content can be attached to the segment(s). When construction starts and when the road is opened, only a few updates are necessary to reflect the change taking place in the real world.

An equally important issue is how to update, i.e., insert and delete, content that is attached to the transportation infrastructure. The same content may be attached to several segments, e.g., a certain sight can be viewed from several segments. Content may be attached to whole segments, e.g., some speed limits, or to parts of segments (intervals), e.g., views of sights. As stated in Section 4.4, in our model, segments are as long as possible. Having long segments generally results in fewer updates and faster queries, since a piece of content needs only be attached to few segments, meaning that relatively few rows are needed to attach content to the infrastructure. However, the segment length can be tuned to achieve the desired query/update performance independently of the real-world transportation infrastructure. This can be done because the segments are purely internal to the system.

6 Querying

The actual system includes several hundred thousands of lines of generated PL/SQL code that maintains and extracts information from the database. The system also implements its own high-level query language. However, in this section, we simply exemplify a couple of common types of queries for content performed by the system.

Simple queries consider one type of content and are concerned with a single table or view. Such queries generally fall into one of two categories. *Point queries* select point content, e.g., geographical coordinates of accident from a view ACCIDENT_COORDINATE. *Interval queries* select interval content, e.g., car repair service availability ranges with respect to the transportation infrastructure in the kilometer-post representation from a view SERVICE_KM_POST.

More advanced content-related queries combine two or more content types. Such a query may retrieve the accidents that fall into road sections with surface type gravel. These queries join several views or tables. Three general categories of binary joins are defined according to characteristics of the content they combine: point-point, point-interval, and interval-interval join queries.

The third category is the most interesting. When at least one content type is non-covering, e.g., the one from the "right" table, but it is necessary to retain all the data on the content type from the "left" table in the result, the query becomes a left outer interval join. Two pieces of content must belong to the same segment and must have overlapping intervals in order to contribute to the result. The interval associated with the result is the intersection of the two argument intervals. The outer-join condition ensures that content in the left table is in the result if there is no matching content in the right table.

As an example query, assume that we need to determine all housing properties located along roads with high traffic volumes. To display this information on a map, we perform a left outer join on tables PROPERTY_TYPE_SEG and TRAFFIC_SEG (both are content related to segments), respectively, to obtain relevant segment sections. A further join with tables ROAD_SEG and ROAD provides names of the roads involved, and a join with table COORDINATE provides the locations of the result on a map.

7 Summary and Research Directions

Mobile, location-based information services, including traffic, weather, tourist, and safety-related services, are seen as a very important new type of application in the near-future technology marketplace. A number of enabling technologies such as precise geo-positioning, increasingly available wireless communications, and highly functional, portable devices have converged to the point where such services are feasible.

The driving maxim behind location-based services is "content is king," i.e., truly useful services will only emerge when a range of diverse content is available and related to the geographical infrastructure (roads, etc.) in which the users are travelling. This renders advanced infrastructure representations and integration of content with the infrastructure essential. No single infrastructure representation is capable of serving all purposes, making multiple, interrelated representations of the infrastructure necessary.

This paper describes a real content integration and service delivery system developed and deployed by the Danish company Euman A/S [7]. The paper presents a case study, a set of requirements, and the technical architecture of the system. The primary part of the paper covers data modeling for the system, concentrating on the infrastructure representations used in the system, namely the kilometer-post, link-node, geographical, and segment representations, and the integration of these with each other and the attached content. Additionally, substantial challenges related to querying and updating are discussed.

The data model used in the Euman system generalizes previous models, mainly by using the purely internal concepts of segments as the integrating representation. As a result, the system data are much easier to maintain when the real world infrastructure changes.

The current status of the system is that SMS-based traffic and weather information services have been deployed. The deployment of these relatively simple services has not led to any significant problems. A prototype of an advanced intelligent Co-Pilot running on a Nokia 9210 Communicator and a Nokia 7650 is ready for deployment, but awaits, among other things, the broad availability of billing for GPRS. While the current functionality of the Co-Pilot includes route management and GPS-based speed limit checking, additional functionality is being added.

Future work includes a number of interesting challenges. Euman is planning to enhance the support for time-varying content, to offer better support for data streams, and to use a variety of business intelligence technologies in the system, e.g., for prediction of traffic jams. Integration of more types of content, including on-line integration of content is also a prime focus area. Finally, research will be done on the advanced query processing techniques needed to support, e.g., dynamic segmentation queries.

Acknowledgments

This work was supported in part by the Danish Centre for IT Research, grants 216 and 333, and by grants from the Euman, Nykredit, and Sonofon corporations. The photos in Figure 1 were made available to us for use in this paper by the COWI corporation, which holds the rights to the photo.

References

[1] J. Añez, T. de la Barra, and B. Pérez. Dual Graph Representation of Transport Networks. *Transportation Research* 30(3):209–216, 1996.

[2] R. Benetis, C. S. Jensen, G. Karčiauskas, and S. Šaltenis. Nearest Neighbor and Reverse Nearest Neighbor Queries for Moving Objects. In *Proc. IDEAS*, pp. 44–53, 2002.

[3] T. Caldwell. On Finding Minimum Routes in a Network With Turn Penalties. *CACM* 4(2):107–108, 1961.

[4] R. A. Deighton and D. G. Blake. Improvements to Utah's Location Referencing System to Allow Data Integration. 16 pp., 1993. *http://www.deighton.com/library/paper2.pdf*. Current as of February 16th, 2003.

[5] P. Djernæs, O. Knudsen, E. Sørensen, and S. Schrøder. VIS-brugerhåndbog: Vejledning i opmåling og inddatering. 58 pages, 2002. *http://www.vejsektoren.dk/imageblob/image.asp?objno=70508*. Current as of February 16th, 2003. (In Danish.)

[6] K. Dueker and J. A. Butler. GIS-T Enterprise Data Model with Suggested Implementation Choices. *Journal of the Urban and Regional Information Systems Association* 10(1):12–36, 1998.

[7] Euman. *http://www.euman.com*. Current as of February 16th, 2003.

[8] D. Fletcher. Modelling GIS Transportation Networks. In *Proc. Annual Meeting of the Urban and Regional Information Systems Association*, pp. 84–92, 1987.

[9] M.F. Goodchild. Geographic Information Systems and Disaggregate Transportation Modeling. *Geographical Systems* 5(1):9–17, 1998.

[10] G. Kollios, D. Gunopulos, and V. J. Tsotras. Nearest Neighbor Queries in a Mobile Environment. In *Proc. STDBM*, pp. 119–134, 1999.

[11] N. Koncz and T. M. Adams. A Data Model for Multi-Dimensional Transportation Location Referencing Systems. *Journal of the Urban and Regional Information Systems Association* 14(2):27–41, 2002.

[12] C. Murray. Oracle Spatial User Guide and Reference, Release 9.2. *Oracle Corporation*, 486 pp., 2002.

[13] NCHRP. A Generic Data Model for Linear Referencing Systems. *Transportation Research Board, Washington, DC*, 28 pp., 1997.

[14] P. Okunieff, D. Siegel, and Q. Miao. Location Referencing Methods for Intelligent Transportation Systems (ITS) User Services: Recommended Approach. In *Proc. GIS-T Conference*, pp. 57–75, 1995.

[15] C. Raskind. Location-Based Services: Revenues & Applications. Strategy Analysts Market Report, 53 pp., 2000. *http://www.strategyanalytics.com/cgi-bin/greports.cgi?rid=152000020531*. Current as of February 16th, 2003.

[16] P. Scarponcini. Generalized Model for Linear Referencing in Transportation. *GeoInformatica* 6(1):35–55, 2002.

[17] S. Šaltenis, C. S. Jensen, S. T. Leutenegger, and M. A. Lopez. Indexing the Positions of Continuously Moving Objects. In *Proc. ACM SIGMOD*, pp. 331–342, 2000.

[18] Y. Tao, D. Papadias, and Q. Shen. Continuous Nearest Neighbor Search. In *Proc. VLDB*, pp. 287–298, 2002.

[19] Viggen Corporation. Location Referencing Systems: Analysis of Current Methods Applied to IVHS User Services. 1994.

[20] O. Wolfson, L. Jiang, A. P. Sistla, S. Chamberlain, N. Rishe, and M. Deng. Databases for Tracking Mobile Units in Real Time. In *Proc. ICDT*, pp. 169–186, 1998.

[21] World Wide Web Consortium. Web Services Activity. *http://www.w3.org/2002/ws/* . Current as of February 16th, 2003.

[22] M. Zeiler. Modeling Our World. *ESRI Press*, 200 pp., 1999.

ROAD

	rd_nr	rd_part	rd_mng	r_id
1	337	0	55	6068
2	337	1	55	6069
3	337	2	55	6070
4	337	3	55	6071
5	337	4	55	6073
6	337	5	55	6074
7	337	6	55	6075
8	337	7	55	6076
9	337	8	55	6077
10	362	0	55	371
11	363	0	55	6101
12	550	0	55	1061

ROAD_SEG

	seg_id	seg_from	seg_to	r_id
1	893	0	43802	6068
2	893	43842	44786	6068
3	894	0	338	6071
4	894	338	724	6074
5	896	0	373	6075
6	896	373	665	6073
7	897	0	243	6076
8	898	0	280	6077
9	927	1112	28678	371
10	936	60	314	6101
11	1679	7440	8414	1061
12	1679	8454	18693	1061
13	3522	0	62	6069
14	3523	0	64	6070

CONNECTION

	seg_id	seg_from	con_id
1	893	34900	126
2	893	35009	127
3	893	35586	129
4	893	35722	130
5	893	43802	5387
6	893	43842	5389
7	894	0	127
8	894	338	128
9	894	724	130
10	896	0	126
11	896	373	123
12	896	665	129
13	897	0	133
14	897	243	134
15	898	0	133
16	898	280	124
17	927	28339	134
18	927	28678	124
19	934	10448	132
20	936	234	128
21	936	314	123
22	1679	8414	5388
23	1679	8454	5390
24	3522	0	5387
25	3522	40	5388
26	3522	62	5389
27	3523	0	5387
28	3523	26	5390
29	3523	64	5389

KM_POST

	r_id	km_nr
1	6068	46
2	6068	47
3	6068	48
4	6068	55
5	6068	56
6	6069	55
7	6070	55
8	6071	46
9	6071	47
10	6073	47
11	6074	47
12	6075	46
13	6075	47

KM_POST_SEG

	seg_id	seg_from	seg_to	r_id	km_nr	offset
1	893	34064	35064	6068	46	0
2	893	35064	36069	6068	47	0
3	893	36069	37069	6068	48	0
4	893	43068	43802	6068	55	0
5	893	43842	44069	6068	55	774
6	893	44069	44786	6068	56	0
7	894	0	55	6071	46	945
8	894	55	338	6071	47	0
8	894	338	724	6074	47	272
10	896	0	164	6075	46	836
11	896	164	373	6075	47	0
12	896	373	665	6073	47	230
13	897	0	128	6076	45	872
14	897	128	243	6076	46	0
15	898	0	128	6077	45	872
16	898	128	280	6077	46	0
17	3522	0	62	6069	55	734
18	3523	0	64	6070	55	734

LINK

	l_id	from_n_id	to_n_id	lnk_length
1	45	5637	5638	6250
2	46	5638	5637	6250
3	678	5638	5635	1776
4	679	5635	5638	1776
5	680	6602	6603	8512
6	681	6603	6602	8512
7	1163	5638	6602	1226
8	1164	6602	5638	1226
9	1800	7207	6602	211
10	1801	6602	7207	211
11	1802	6602	7208	37
12	1803	7208	6602	37
13	1958	6603	7014	8800
14	1959	7014	6603	8800
15	2267	6603	6713	8121
16	2304	6603	6714	994
17	2463	6714	6603	994
18	2507	6713	6603	8121
19	4287	5638	8566	165
20	4288	8566	5638	165

LINK_SEG

	l_id	seg_id	seg_from	seg_to
1	679	893	32310	34086
2	678	893	32310	34086
3	1163	893	34086	35312
4	1164	893	34086	35312
5	680	893	35312	43824
6	681	893	35312	43824
7	2267	893	43824	51945
8	2507	893	43824	51945
9	45	927	22263	28513
10	46	927	22263	28513
11	4287	927	28513	28678
12	4288	927	28513	28678
13	1800	936	60	277
14	1801	936	60	277
15	1802	936	277	314
16	1803	936	277	314
17	2304	1679	7440	8434
18	2463	1679	7440	8434
19	1958	1679	8434	17226
20	1959	1679	8434	17226

SEGMENT

	seg_id	seg_length
1	893	78326
2	894	724
3	896	665
4	897	243
5	898	280
6	927	28678
7	936	314
8	1679	18693
9	3522	62
10	3523	64

NODE

	n_id
1	5635
2	5637
3	5638
4	6602
5	6603
6	6713
7	6714
8	7014
9	7207
10	7208
11	8566

Figure 7: Sample Data