On a Location Model for Fine-Grained Geocast

Frank Dürr and Kurt Rothermel

Institute of Parallel and Distributed Systems (IPVS),
University of Stuttgart,
Universitätsstraße 38, 70569 Stuttgart, Germany
{Frank.Duerr,Kurt.Rothermel}@informatik.uni-stuttgart.de

Abstract. Geographic communication (geocast) is used to send messages to geographic areas, e.g. to distribute warning messages or other information within these areas. It is based on a location model which is used to define a message's target area and the receivers' positions and therefore has strong influence on the achievable granularity of geographic addressing.

A hybrid location model and a fine-grained addressing scheme for geocast based on this model are presented in this paper which support two- and three-dimensional geometric locations as well as symbolic locations like room numbers, embedded local coordinate systems, and mobile target areas like trains.

1 Introduction

The availability of small and mobile devices as well as various positioning systems which can be used to determine the position of these devices enable new forms of communication like geographic communication (geocast). Geocast is used to send messages to certain geographic areas. A typical scenario is the distribution of warning messages. For instance, a geocast message could be sent to all people close to the location where a fire started to ask them to leave this area immediately or just to keep their windows shut because of the toxic smoke. Announcements about traffic jams or car accidents are also interesting mainly for people in the vicinity. An indoor scenario would be the distribution of the slides or other additional information about a talk to all listeners located in the conference room.

Therefore, geocast can be seen as a special kind of multicast. In multicast, a message is sent to a group of receivers. Such a group can be established through explicit join requests by the participants, or it can be established implicitly. In geocast, a group is defined implicitly through the geographic positions of clients and the given geographic target area of the message. The idea is that some messages are only interesting for clients in a certain geographic area. Therefore, all clients inside this target area should receive the message.

As a location-based service, geocast builds on an underlying location model. This location model is used by receivers and the senders of geocast messages: The sender uses the model to define the target area of a geocast message, and

Published in UbiComp 2003: Ubiquitous Computing
(Proceedings of the 5th International Conference on
Ubiquitous Computing 2003), LNCS 2864, pp. 18-35, 2003.

the receivers determine whether or not they are in the target area. A client will only deliver a message, if its position is inside the specified target area. Clearly, senders and receivers need a common understanding of locations, which is provided by a location model.

[1] distinguishes between geometric models which define locations by geometric figures and symbolic models using abstract identifiers like street names, room numbers, etc. Most location models used by current geocast implementations like [2] are restricted to two-dimensional geometric models using a global reference system. But for urban areas, it should be possible to use also three-dimensional target areas, defined either by three-dimensional geometric figures or symbolic identifiers which are often more intuitive to use, because these are the location identifiers people are used to. Otherwise for example only a whole building could be addressed, even if only the clients inside a small room in this building are to receive a message (see conference example above). Another example for fine-grained geographic addressing using three-dimensional target areas would be a message to all cars on a bridge, e.g. to inform them about an accident or traffic jam, which is irrelevant for the cars passing below the bridge.

Different hybrid models supporting geometric and symbolic locations have already been proposed, e.g. [1, 3]. In this paper, the conceptual design of a hybrid location model for geocast is presented which additionally supports different forms of embedded local geometric and symbolic models enabling e.g. addresses of the form "geometric → symbolic", i.e. a symbolic location within a geometrically defined area, as well as mobile target areas like ships and trains. The possibly high modelling effort for a fine-grained hybrid location model is reduced by approximation, and model information is used to cope with inaccurate client positions. Although the focus is on a location model for geocast, the presented model may also be applicable to other areas, e.g. to a general location service like [4] that answers so-called range queries (Which objects are in a certain geographic area?).

The rest of this paper is structured as follows. Section 2 presents the system model and gives a short overview of the functionality of the different components used for geocast. Then, the requirements for a location model for fine-grained geocast are stated in Sec. 3. Based on these requirements, a location model is designed in Sec. 4. It will be shown in detail how target areas can be defined by geographic addresses and how they can be compared to client positions based on model information. Finally, an overview of related work is given in Sec. 5, before this paper is concluded with a short summary and an outlook on future work in Sec. 6.

2 System Model

The following components are involved in the delivery of geocast messages.

The *geocast client* is a software component running on a mobile or stationary device like a PDA or PC. It is responsible for sending geocast messages it has received from local *applications* together with the target area address of the message to a geocast router for forwarding. The client is also responsible for delivering messages it has received from a router to applications on its device which are listening for these messages. It knows the position of the device and filters out messages with target areas not containing this client position.

Routers are responsible for forwarding messages from the sender to the clients. According to [2], there are three different ways of geocast message forwarding. **Geographic routing** uses geographic information directly for message forwarding, i.e. special routers compare the target area of the message and the areas covered by sub-networks to decide where to forward the message. The **directory-based approach** stores mappings between geographic locations and IP addresses in a directory and uses this information to determine all receivers in the target area. Messages are sent to each IP address with an associated area which overlaps the target area of the message. The **multicast-based approach** divides the world into so-called partitions with associated multicast addresses. To send a geocast message, first the multicast address of the partition is determined that encloses the target area. Then, a multicast message is sent to this group, that all receivers in this area have joined. A simplified version would be to just broadcast the message and let the receivers filter out messages based on the target area and their current position. Looking at these approaches it is clear that no matter which forwarding approach is used, the target area and the client position always have to be compared before the message is delivered to geocast applications. Where this comparison takes place – at the sender's or the receiver's geocast client, or at some component (e.g. at routers) between sender and receiver – is irrelevant for the conceptual design of the location model, because the model has to provide the necessary information to carry out this comparison in either case. Therefore, the following discussion is independent of the type of forwarding.

Positioning systems are required to determine client positions. Considering their output, two classes can be distinguished [1, 5]: geometric and symbolic systems.

Geometric positioning systems return client positions as geometric figures using coordinates relative to a geometric reference system. This can be a global reference system, e.g. the World Geodetic System 1984 (WGS84) [6] used by the Global Positioning System (GPS), or a local reference system. Especially geometric positioning systems in the indoor domain use local reference systems. Typically, they can only be used in a limited area, i.e. the local coordinate system has a limited scope in contrast to the global scope of for example the WGS84. The Active Bat System [7] – a highly accurate geometric indoor positioning system – for instance uses ultrasound which is blocked by obstacles like walls. If a room is equipped with such a system, only people inside this room can be tracked, i.e. the room is the scope of the system.

Symbolic positioning systems return a symbolic location identifier when a client gets close to a positioning sensor. The Active Badge System [8] for example uses badges transmitting infrared signals, which are picked up by fixed sensors in the environment, so the system can identify the sensors which are within sight of a badge. But identifying nearby sensors is usually only the first step of positioning. Next, the sensor identifiers are mapped by the positioning system onto a symbolic location identifier according to the location model, e.g. a room number, that identifies the client position.

We assume that geometric as well as symbolic positioning systems will be used by clients, e.g. a geometric system like GPS outdoors and a symbolic one like Active Badge indoors. A client possibly knows either a symbolic or a geometric position, but not necessarily both.

Additionally, we assume that the output of positioning systems is not perfectly accurate, i.e. the client position is not defined by a point but by an area called the client area c, e.g. a polygon or the identifier of a room. Therefore, two areas (target and client area) have to be compared, which may overlap partially. The following function is defined that calculates the probability that a client with client area c is in the target area t of a geocast message:

$$p(t, c) \mapsto [0, 1] \quad (1)$$

A value of 1.0 means that the client is completely inside t; 0.0 that it is outside t. If p is greater or equal a specified threshold, then the message will be delivered, otherwise discarded. No matter whether the sender or the receiver defines this threshold, model information can be used to calculate this probability as shown later in Sec. 4.1 and Sec. 4.2.

Finally, the *Spatial Model Service* manages a world model containing location information about countries, cities, buildings, etc. This paper will present in detail which information has to be stored by such a service in order to realize fine-grained geocast.

3 Requirements

Hybrid Model Because clients can use symbolic as well as geometric positioning systems, a hybrid location model is required to define both kinds of client positions. From a sender's point of view, symbolic locations are often more intuitive to use than geometric coordinates as already mentioned in Sec. 1. Additionally, a symbolic model can be set up with little effort compared to a geometric model, especially for big buildings with many rooms. On the other hand, arbitrary areas defined by geometric coordinates are useful, if there is no or only a coarse-grained predetermined symbolic structure given, which holds especially outdoors, e.g. if some small part of a big plaza is to be addressed. But also if very fine-grained locations are required, a geometric figure might be the better choice. Of course, geometric coordinates should not be limited to two dimensions but three-dimensional figures should be supported as already mentioned in Sec. 1.

Heterogeneous Hybrid Model We cannot assume that the world is modelled symbolically *and* geometrically at the same level of granularity. For instance, the indoor domain will often be modelled only symbolically, especially if only a symbolic positioning system is available within a building. For other areas, only geometric model information is available. Therefore, a global location model will be built of many partial models, i.e. means are required to integrate local systems into other systems, e.g. a local geometric system into a symbolic one or vice versa. Such a heterogeneous model also leads to situations where the sender specifies the target area of a message geometrically (e.g. by drawing a figure on a map), and only a symbolic client position is known (e.g. if the client is in a building equipped with an Active Badge System) or vice versa. Therefore, a mapping of geometric to symbolic areas or vice versa based on model information is required in order to compare these heterogeneous areas.

Mobile Target Areas Geocast messages cannot only be sent to static target areas like buildings, but the target area itself may be a mobile object or a location within a mobile object. Typical examples are messages to the dining car of a train or the first deck of a cruise ship. While these objects move, their global coordinates and the coordinates of locations within the object are changing. If these global coordinates were used to specify the target area of a geocast message, this address would also change. Surely, such a constantly changing address is not useful, and thus global coordinates are not appropriate to address these targets. Therefore, means are required to use local coordinates relative to the moving object, which do not change on movement, to address targets within this object.

4 Fine-Grained Location Model

In this section, the design of a hybrid location model is presented. Also an addressing scheme for geocast based on this model is shown and how target areas and client areas can be compared, i.e. how to implement the function specified in Equation 1 that returns the probability that a client is within the target area.

4.1 Symbolic Model

The symbolic part of the presented model consists of a hierarchy of symbolic locations, modelled according to the spatial “contains” relationship. In the example depicted in Fig. 1a, floor F_2 contains the rooms R_3, R_4, and R_5, and building B contains the floors F_1 and F_2, etc. A client located in room R_3 is also located on floor F_2 and in building B. Although this example is rather simple, it illustrates that locations may overlap. For instance, building B contains two wings W_1 and W_2. Wings and floors overlap, e.g. room R_3 is on floor F_2 as well as in wing W_1. In order to be able to model this situation, a model that supports overlapping locations is required. [9] proposed such a model based on the spatial inclusion relationship. That means, for two locations $l_1, l_2 \in L$ where L is the set of symbolic locations, it holds $l_1 \leq l_2$, if l_2 spatially contains l_1. There are two special elements in L named *everywhere* and *nowhere*. The following holds: $\forall l \in L : l \leq everywhere$, and $\forall l \in L : nowhere \leq l$. Mathematically, L and $\leq$ form a lattice, i.e. for each pair of elements $x, y \in L$ there is a (unique) least

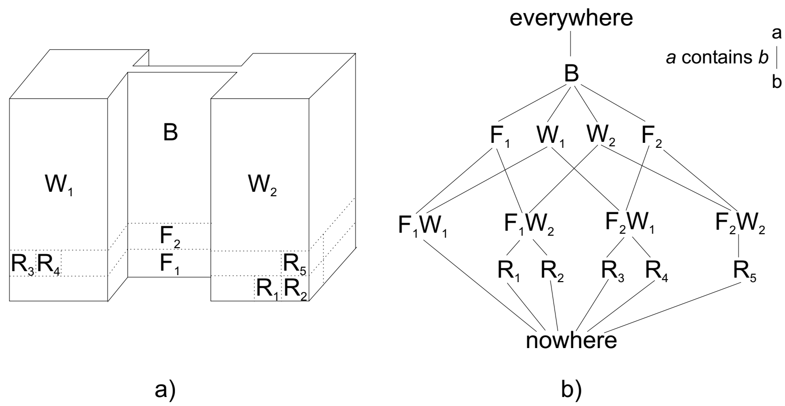

Fig. 1. Hierarchical Symbolic Locations

upper bound denoted by $x \sqcup y$ and a (unique) greatest lower bound denoted by $x \sqcap y$. Figure 1b shows the resulting lattice of the example.

Note that this definition of a symbolic location model includes the often used tree-based models, which only allow at most one parent for each location. But a tree is too limited as already shown in the rather simple example above. Neither floors are completely contained in wings, nor wings are completely contained in floors, but each room is part of one floor as well as one wing. A tree cannot model this situation, but the more general and powerful lattice can.

The location hierarchy is used to define symbolic geocast addresses. An address consists of the concatenation of multiple relative identifiers. Each identifier uniquely identifies a location in the context of its parent(s). For instance in the hierarchy shown in Fig. 2, the identifier `room72` identifies room 72 in the context of the location `/de/berlin/keplerstr/9/floor1/wing1`. We use Universal Resource Identifiers (URI) [10] in combination with an XML-based language as syntax for symbolic addresses. The following address stands for the above mentioned location:

```
<targetarea>
 <symbol>loc:/de/berlin/keplerstr/9/floor1/wing1/room72</symbol>
</targetarea>
```

Note that locations may have multiple addresses (e.g. room 72 can also be referred to by the address `/de/berlin/keplerstr/9/wing1/floor1/room72`), whereas each address only stands for exactly one location.

Based on the definition of a symbolic location model, it can now be defined how to calculate the intersection of two symbolic locations. Given a pair of symbolic locations $l_1, l_2 \in L$, l_1 and l_2 overlap, if $l_\sqcap = l_1 \sqcap l_2 \neq \mathit{nowhere}$. $l_\sqcap$ is the intersection of l_1 and l_2. This intersection is used in the implementation of the probability function. First the intersection $l_\sqcap$ of the target area t and the

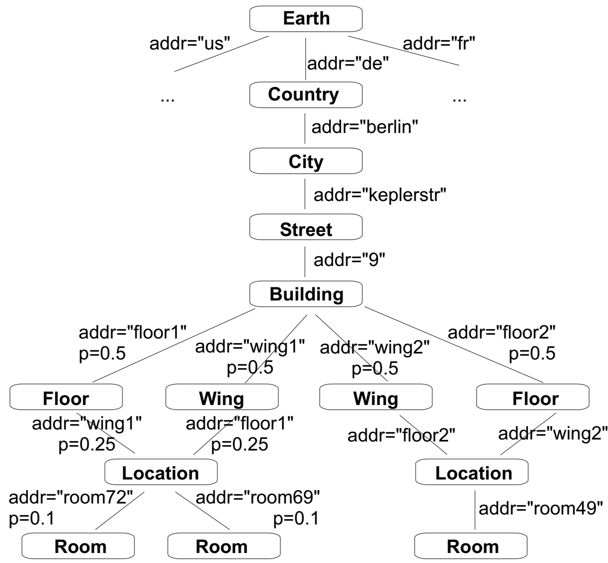

Fig. 2. Symbolic Location Model (some locations like the intersection of floor 1 and wing 2 were omitted for clarity)

client area c is determined by calculating $l_\sqcap = t \sqcap c$. Then, three cases can be distinguished:

- $l_\sqcap = nowhere$: Client area and target area do not overlap at all.
  Example 1: $c =$ `/de/berlin/keplerstr/9/floor1/wing1/room72` and $t =$ `/de/berlin/keplerstr/9/floor1/wing1/room69`, i.e. a message is sent to one room, and the client is located in another room.
- $l_\sqcap = c$: The client is completely within the target area.
  Example 2: $c =$ `/de/berlin/keplerstr/9/floor1/wing1/room72` and $t =$ `/de/berlin/keplerstr/9`, i.e. a message sent to the whole building, and a client in one room of this building.
- $l_\sqcap \neq c$ and $l_\sqcap \neq nowhere$: Target and client area overlap partially.
  Example 3: $c =$ `/de/berlin/keplerstr/9` and $t =$ `/de/berlin/keplerstr/9/floor1/wing1/room72`, i.e. a message addressed to a certain room in a building and the client only knows that it is inside this building, but it does not know the room it is currently in. This may be due to an inaccurate positioning system and a location model that permits target area addresses of a finer granularity than this system can provide (positioning systems used by other clients may be more accurate, and therefore it makes sense to have such a fine-grained model).
  Example 4: $c =$ `/de/berlin/keplerstr/9/floor1` and $t =$ `/de/berlin/keplerstr/9/wing1`, i.e. a message sent to a certain wing of the building and a client that only knows the floor it is currently on. The floor and the addressed wing overlap.

If the client is outside the target area ($l_\sqcap = nowhere$), then p is 0.0; if it is completely within t ($l_\sqcap = c$), then p is 1.0. Partially overlapping locations (case 3) require knowledge about the probability that a client located at c is at the same time at $l_\sqcap$ with $c \geq l_\sqcap$. Therefore, the probability has to be modelled that a client in area c (e.g. building 9) is also located at the sub-locations of c (e.g. floor 1). The following rules hold for a consistent model:

1. If a location is completely partitioned by its sub-locations, then the summed up probability of all disjoint[1] direct sub-locations in a partition is 1.0, because if it is for instance known that a client is in a certain building, then it also must be at some sub-location, e.g. on floor 1.
2. If a location is not completely partitioned (i.e. if for example not all locations within a building have been modelled), then this sum is less than 1.0, because the client may be at an "unknown" sub-location.
3. The summed up probabilities of all disjoint direct sub-locations of a location must never be greater than 1.0, because a client can never be at two disjoint locations of a partition at the same time.

Based on this information, the wanted probability for being at location $l_\sqcap$ can be calculated by multiplying the probabilities on the path from c to $l_\sqcap$. The following example shows how simple rules of thumb can be used to calculate this probability.

Let us assume that building 9 has two floors (and two wings). Then, the probability that a client in building 9 is also on the first floor (or in the first wing) is $\frac{1}{2}$. If there are two floors and two wings, then they overlap at 4 regions, thus the probability of a client on floor $i \in \{1, 2\}$ for being in wing $j \in \{1, 2\}$ at the same time is $\frac{1}{4}$. The rest of the probabilities are given as shown in Fig. 2. Now, the probability of a client in building 9 for being at the same time in room 72 (example 3) is $p = \frac{1}{2} \cdot \frac{1}{4} \cdot \frac{1}{10}$. This estimation can be improved. Room sizes can be used to give larger rooms higher probabilities, or further context information can be taken into account. For instance, the probability that a person is in his own office during working hours might be much greater than for being in another office. In this paper, we will refrain from integrating additional context information.

4.2 Geometric Model

The geometric model contains locations in form of geometric figures, e.g. two-dimensional polygons or circles. The WGS84 is used as global reference system, i.e. coordinates are given by longitude, latitude, and altitude values. These two-dimensional figures are well-suited for objects like countries and cities, but also three-dimensional objects like rooms have to be addressed as mentioned in Sec. 1. But arbitrary three-dimensional figures can be very complex. Thus, not fully-fledged three-dimensional but so-called 2.5-dimensional figures are used. Such a figure is defined by a two-dimensional figure (the base, e.g. a circle or a polygon),

[1] Note that because a lattice is used, sub-locations like floors and wings may overlap.

a fixed height, and the altitude of the base (see Fig. 3). They are only a little bit more complex than two-dimensional figures (the only additional values are the height and altitude of the base), and most common target areas (e.g. rooms, floors, etc.) can be approximated fairly well.

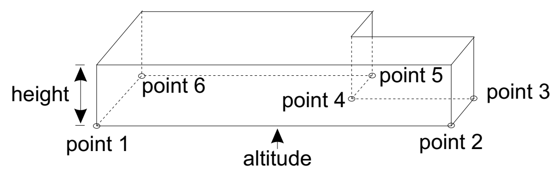

Fig. 3. 2.5-Dimensional Figure

The geometric figures described above can be used as target area addresses of geocast messages. Here is a simple example using a two-dimensional polygon:

```
<targetarea>
 <polygon><vertex>9.126052E 48.721938N</vertex>...</polygon>
</targetarea>
```

To implement the probability function, first the intersection $c \cap t$ of the client area c and the target area t is calculated by geometric operations (see Fig. 4). Then, the following equation is used to calculate the probability p (note the similarity to the symbolic case: p is calculated based on the degree of overlap of target area and client area):

$$p = \frac{A(c \cap t)}{A(c)} \qquad \text{with } A(x)\text{: area of figure } x \tag{2}$$

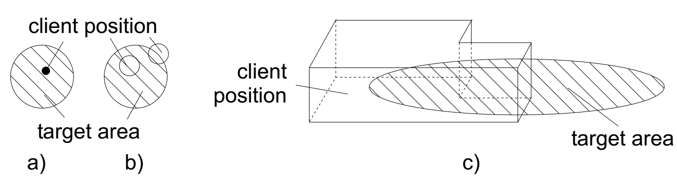

Fig. 4. Comparison of Geometric Client Positions and Target Areas. a) Client position (point) inside target area. b) Client position (two-dimensional figure) overlaps target area (totally and partially). c) Comparison of 2.5-dimensional client position and two-dimensional target area using the base of the 2.5-dimensional figure.

4.3 Hybrid Model

The hybrid location model is formed by associating symbolic locations with their geometric representations, i.e. two- or 2.5-dimensional figures. A perfect model would contain the exact geometry of every symbolic location, but this would mean great modelling effort. A symbolic *and* a geometric client area have to be known as already mentioned in Sec. 3 in order to receive symbolically *and* geometrically addressed geocast messages, because only homogeneous locations can be compared. The following approach does not associate the exact geometry but uses approximation to reduce modelling effort and guarantees that every symbolic location is associated with a (possibly approximated) geometric representation.

Symbolic locations can be *explicitly* associated with their "exact" geometry. We assume that locations high up in the hierarchy tend to have such an explicitly defined geometry. For instance, the outline of most buildings is known or can be determined with little effort from existing plans. At least the borderlines of a city or country are well-known. If a symbolic location is not explicitly associated with a geometric figure, then it inherits the geometry of its parent(s), which in turn may recursively inherit the geometry of their parents, if they have no explicitly associated geometry. If a location has only one parent, then it simply inherits its parent's geometry. If a location has more than one parent[2], then the intersection of the parents' associated geometric figures is used, because the lattice-based model defines a sub-location as the intersection of its parents, if it has multiple parents.

Inherited geometric figures are only approximations. They contain the approximated symbolic location completely but they are too big (e.g. a room may inherit the geometry of the whole building, and in the worst case, the geometry of the whole earth is inherited).

When a geometric and a symbolic location – i.e. a geometric target area and a symbolic client area or vice versa – have to be compared in order to determine the probability that a client is inside the target area, the geometric information associated with symbolic locations can be used to translate either the symbolic location to a geometric figure or the geometric figure to a symbolic location. Then, homogeneous locations can be compared as shown in Sec. 4.2 and Sec. 4.1, respectively.

To translate a symbolic location to a geometric figure, the associated geometry of the symbolic location is used.

For the translation of a geometric figure to a symbolic location, there are several approaches. One approach is to associate the smallest symbolic location whose geometric representation encloses the figure entirely. But this may lead to very coarse approximations as shown in Fig. 5. The small geometric area L is translated to the very large symbolic location representing the whole earth, because only the whole earth contains L *entirely*. A second approach is to translate L to multiple smaller symbolic locations, all of which overlap the geometric

[2] Note that this is possible because a lattice is used.

area at least partially. This results in an approximation that is more accurate. In the depicted example, location L could be translated to two symbolic locations representing country A and B. This may result in several necessary comparisons of target area and client position. But in general, only very few symbolic locations will be the result. Therefore, the second approach seems to be the better compromise, if the possibly very coarse approximation of the first approach is regarded.

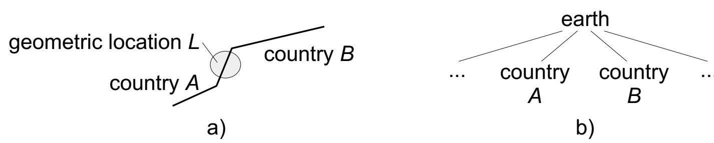

Fig. 5. Translation from Geometric to Symbolic Coordinates a) Small geometric area L overlapping two large symbolic areas. b) Symbolic hierarchy. L can be translated to $\{earth\}$ or $\{A, B\}$

As already mentioned, the geometry or symbolic location resulting from the translation may only be a more or less accurate approximation. How this approximation affects the delivery of geocast messages is dependent on whether the target area or the client position is translated and therefore possibly approximated.

If the client area is translated, then the translated client area may be bigger than the original area. Thus, according to Equation 2, the calculated probability can be very low, and therefore the message may be discarded depending on the threshold value. Let us for instance assume that the symbolic location $l =$ `/de/berlin/keplerstr/9/floor2/room172` is associated with the (inherited) geometry of the whole building because neither the exact geometry of the room nor the floors are modelled explicitly. Suppose a client located at l, i.e. $c = l$. If a geometrically addressed message is sent to a room of this building and the geometric approximation of l is used in Equation 2, then p will be very low, because the intersection of the target area and the geometric representation of l (the whole building) is very small compared to the geometry of c. But if a geometrically addressed message is sent to the city of Berlin containing this building and the room, then the probability will be 1.0 even though the client area has been approximated, because the target area contains the approximated geometry of l completely, and thus the message will be delivered. Therefore, this approximation is only effective for target areas which are bigger than the geometry used as approximation for the symbolic position. This approach should only be used, if it is acceptable that some clients might not deliver a message even though they are in the target area (unimportant messages with high threshold).

If the target area is translated, then an approximated target area is bigger than the original one. Suppose the example shown above and a target area

$t = l$. If now this symbolic target area address is translated to its geometric representation, which is the whole building, not only the clients inside the wanted room will deliver the message, but also every client in this building who knows its position geometrically (if it knows its position symbolically, then this position can be compared without translation). Therefore, this approach should be used, if it is acceptable that clients outside the target area deliver the message (important messages with low threshold).

Note that the shown problems related to approximation also hold for the translation of geometric to symbolic locations, because in this case the approximated geometry is used to select one or multiple appropriate symbolic locations as shown above which also may be too big. Therefore, approximation is always only a compromise.

The geometry associated with symbolic locations can also be used to define target areas by combinations of symbolic and geometric attributes. For instance, an address like "200 m around the main station in Berlin" could be defined. Note that this does not require new forms of geographic addresses, because the sender can query the model for the geometric representation of the main station and then uses geometric operations to expand this figure by 200 m. The result is a purely geometric address as described in Sec. 4.2. It has to be mentioned that geometric approximation can make this kind of target address useless in some situations. If for instance a room is approximated by the geometry of a whole building, then it makes no sense to define an area of 5 m around this room, because the approximated geometry is too inaccurate for this short distance.

4.4 Embedded Local Coordinate Reference Systems

Only global coordinates have been considered so far. Now, local symbolic and geometric coordinate systems are introduced which are embedded in global or other local geometric or symbolic systems.

The main difference between global and local systems is the limited scope of the latter, i.e. local coordinates are only valid within this scope. To define the scope of a system, it is associated with a geometric or symbolic location. Each local system has a name that is unique within this area, therefore more than one local system for the same scope can be defined. The scope is used to embed a local system in another higher-level system, and therefore combined addresses of the form "symbolic → geometric", "geometric → symbolic", "geometric → symbolic → geometric", etc. can be specified.

Local coordinates are used in target area specifications by giving the scope and name of the wanted system. The following example uses the system named "sys_room2_72" with a symbolically defined scope (symbolic → geometric):

```
<targetarea>
  <refsys>
   <scope><symbol>loc:/de/.../room72</symbol></scope>
   <name>sys_room2_72</name>
  </refsys>
```

```
  <polygon><vertex>2.4 3.0</vertex>... </polygon>
</targetarea>
```

This would be a typical example for an Active Bat System installed in room 2.72, i.e. a local geometric system embedded in a symbolic one. Note that Cartesian coordinates are used for local geometric systems in contrast to geographic global coordinates defined by longitude, latitude, and altitude. If no reference system is given for a geometric figure, then global WGS84 coordinates are assumed.

The next example shows a target area using a local symbolic system embedded in a geometric one (geometric → symbolic).

```
<targetarea>
 <refsys>
   <scope><polygon>...</polygon></scope>
   <name>sys_building9</name>
 </refsys>
 <symbol>loc:floor2/room72</symbol>
</targetarea>
```

The symbolic location `floor2/room72` is interpreted relative to the local symbolic system of building 9. This could be a scenario for a building modelled symbolically, whereas geometric WGS84 coordinates are used outdoors.

Local reference systems can also be nested, i.e. it is for example possible to use a local geometric system embedded in a symbolic system which in turn is embedded in a geometric system (geometric → symbolic → geometric; Fig. 6). This could be necessary, if e.g. a geometric Active Bat System is installed in room 2.72 in the example above, and the symbolic partial model of the building containing room 2.72 is embedded in the geometric WGS84.

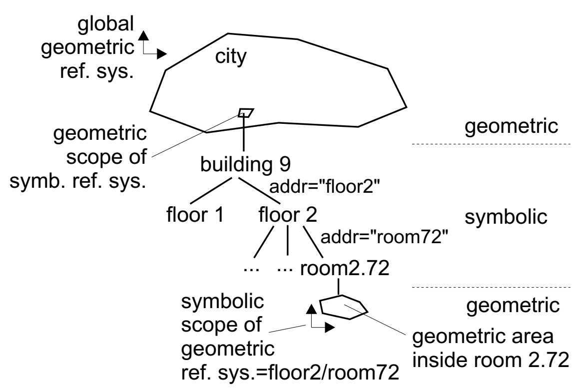

Fig. 6. Multiple Embedded Reference Systems

Multiple coordinate systems raise the question, what happens if the target area of a message is using system S_1, and a positioning system uses another system $S_2 \neq S_1$, i.e. the client position is known in coordinates relative to S_2? How can these two locations be compared? This question is only of relevance if the scopes of both systems overlap, because otherwise the probability p is 0.0. The answer is that the coordinates of system S_1 have to be translated to system S_2 or vice versa. Three cases have to be distinguished: S_1 and S_2 geometric, S_1 and S_2 symbolic, or S_1 geometric and S_2 symbolic or vice versa.

For two geometric systems, the position and orientation of system S_1 has to be known relative to system S_2 or vice versa. [3] defines the position and orientation of local systems relative to other higher-level geometric systems. Geometric systems are associated with symbolic locations forming a tree. Coordinates are translated by successive coordinate transformations along the path in this tree leading from S_1 to S_2 via other systems. This approach can also be used here.

If S_1 and S_2 are both symbolic systems, then a function mapping symbols from one system to the other is required, if not both systems are using the same symbols. We assume that local symbolic systems do not cover the same area, and therefore, this mapping will not be discussed further in this paper.

If S_1 is a geometric and S_2 a symbolic system, then the procedure described in Sec. 4.3 is used, i.e. the symbolic location in S_2 is translated to its geometric representation in S_1 or vice versa. The scope of the local system can be used as approximation. For instance, the geometric locations within the local geometric system "sys_room2_72" in the first example above can be approximated by the symbolic location – the scope of this system – `/de/berlin/keplerstr/9/floor2/room72`. The symbolic location `floor2/room72` and all other symbolic locations within system "sys_building9" in the second example above can be geometrically approximated by the polygon defining the scope of this system.

4.5 Mobile Target Areas

A last requirement for a location model for geocast is the support of mobile target areas, i.e. moving objects like trains and locations inside such objects.

If the presented addressing concept for global coordinates was used, this would result in constantly changing addresses for locations inside the moving object. Figure 7 shows an example of a train moving from Germany to France.[3] This would also result in a change of the address of the dining car from `/europe/de/ec64/dining-car` to `/europe/fr/ec64/dining-car`. The same problem obviously occurs, if global geometric addresses are used, because the geometric coordinates of the train and therefore its dining car also change during movement.

We also want messages addressed to target areas including the moving object to reach all receivers inside this object. For instance, a message addressed to Germany (`/europe/de`) should also reach the passengers of a train (and its dining car) which is currently located in Germany.

The problem of addressing locations within the moving object can be solved by local coordinate systems as presented in Sec. 4.4. Such a local coordinate system is defined for every moving object, and locations inside these objects are

[3] Locations representing continents have been left out so far for the sake of brevity.

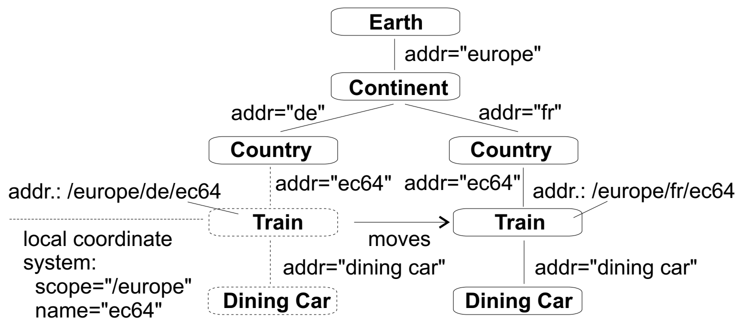

Fig. 7. Mobile Target Area

addressed by local coordinates relative to this system which do *not* change on movement. The following address specifies the dining car of the train "ec64":

```
<targetarea>
 <refsys>
  <scope><symbol>loc:/europe</symbol></scope>
  <name>ec64</name>
 </refsys>
 <symbol>loc:dining-car</symbol>
</targetarea>
```

Note that the scope of the local coordinate system is static, i.e. it does *not* change when the train moves. Because this is a train only travelling through Europe, the address `/europe` has been used as scope. `dining-car` is the static symbolic coordinate relative to the given local system "ec64". Of course, also local geometric coordinates could be used as shown in Sec. 4.4 assumed a local geometric positioning system is available. To filter received messages, the client's coordinates relative to the system "ec64" have to be known, i.e. a suitable positioning system is required that outputs such local coordinates.

Messages to areas containing the moving object and therefore all clients inside this object do not need special prerequisites as far as addressing is concerned, because global coordinates can be used. To filter globally addressed messages, a global client position has to be known. A global positioning system, e.g. GPS, can be used to determine a client's global position. If such a system is not available – e.g. GPS might not be available inside a railroad car –, the global position of the moving object can be used to approximate a client's global position. Therefore, a location service like [4] is necessary that can be queried for the current location of mobile objects, e.g. the train's position. In our example, the train's position would be `/europe/fr`, if the train is currently located in France. If the train's location is `/europe/fr` and a local positioning system reports that the client is inside this train, then the client's global position is also `/europe/fr`.

5 Related Work

This paper is focused on a location model that fulfils the requirements of fine-grained geographic communication. Most models in this context are rather simple based on geometric locations in a two-dimensional coordinate system. For instance, the geocast implementations of [2] supports simple two-dimensional target areas like polygons and circles in a global coordinate system. Our model goes one step further and also supports three-dimensional (more exactly 2.5-dimensional) and hierarchical symbolic locations. Furthermore, our model supports not only a global but also local reference systems and locations within moving objects. This location model enables users to define destination addresses of finer granularity, and therefore makes geocast applicable in urban areas.

In the context of Geographic Information Systems (GIS), [9] proposes to model spatial relations and operations with lattices. We use this approach as basis for the symbolic part of our hybrid model, extend it with an addressing scheme, and model probabilities for being at certain locations based on the symbolic location hierarchy.

Like the presented model, [11] also uses a lattice-based model. But geometric information is not modelled, and therefore the integration of widely used geometric positioning systems like GPS and also highly accurate geometric indoor positioning systems like the Active Bat System is difficult.

Some approaches from the field of qualitative spatial reasoning and GIS like the RCC-8 [12] and the 9-intersection-model [13], respectively, offer a richer set of supported topological relations. For instance, locations "touching" each other can be distinguished from overlapping regions. We think that our hierarchical approach based solely on the spatial part-of relationship is powerful enough to compare target and client areas and still yields models that can be set up easily. Additionally, we introduced probabilities for complex situations where a qualitative statement about the relationship of target and client areas is not sufficient to make decisions about message delivery.

The approach that we consider to be closest related to ours is [3]. This hybrid model also supports symbolic locations as well as geometric representations in local reference systems. In contrast to [3], we do not use a strict tree-based model for symbolic locations but a lattice, i.e. a more general and powerful model that has advantages in more complex situations with overlapping locations. Furthermore, our model supports different forms of embedded local systems, which enables the integration of purely geometrical or purely symbolical partial models. We consider this to be very important for a fine-grained model of global extent that is too complex to be set up by a single authority.

[14] describes another class of location models based on graphs which model locations as nodes and interconnections between these locations, e.g. doors connecting rooms, as edges. Such a model could also be used to define target and client areas *implicitly* by specifying a location and a distance to this location, e.g. everybody within 100 m distance to a certain room. We support similar addresses, but rely on the geometric representation of symbolic locations.

6 Summary and Future Work

The conceptual design of a location model for fine-grained geocast has been presented in this paper. For geocast to be applicable in urban areas, a detailed location model is required to express target areas of geocast messages and client positions. This model has to support geometric locations as well as symbolic locations, embedded local reference systems, and mobile target areas like trains. A formally founded hybrid location model has been designed that supports all these types of locations and the embedding of local geometric and symbolic reference systems, so symbolic and geometric positioning systems can be integrated easily. Additionally, modelling effort is reduced by geometric approximation of symbolic locations. Based on this model, an addressing scheme for geocast has been presented, and it has been shown how homogeneous and heterogeneous locations representing target areas and client positions can be compared by calculating the probability that a client is inside the target area in order to decide, whether a geocast message is to be delivered or not.

The efficient implementation of a service that manages the presented hybrid location model is part of the future work of our project. In [4, 15], we have already proposed a scaleable architecture for the management of geometric location models and a location service that manages large numbers of mobile objects based on geometric location information. To extend this architecture to support also symbolic locations needed for the presented hybrid location model is a big challenge. It has to be scaleable to global extent, which requires a distributed architecture consisting of many servers storing parts of the world model. Geometric *and* symbolic locations have to be managed efficiently to quickly answer queries about spatial containment.

Additionally, a graph-based topological model as shortly described in Sec. 5 will be integrated which enables the implicit definition of areas. Furthermore, such an integrated model can also be used for other tasks like navigation.

The presented location model can be further refined, e.g. by modelling object attributes like class types. This additional information can be used to define groups of receivers within a geographic area, e.g. all taxis near the main station. Messages can then be sent to this group only and not to all receivers in an area.

The design and implementation of suitable geocast protocols is also part of our future work. In [16], we proposed an extension of the geographic routing approach from [2]. Further approaches will be investigated in order to realize fine-grained geocast.

Acknowledgements

The first author gratefully acknowledges the financial support by the Deutsche Forschungsgemeinschaft within the Centre of Excellence 627 “World Models for Mobile Context-Based Systems”. We would like to thank Martin Bauer, Tobias Drosdol, Dominique Dudkowski, Alexander Leonhardi, and the anonymous reviewers for their help to improve this paper, and especially Christian Becker for all the fruitful discussions about location models.

References

1. Leonhardt, U.: Supporting Location-Awareness in Open Distributed Systems. PhD thesis, Imperial College London, Department of Computing (1998)
2. Navas, J.C.: Geographic Routing in a Datagram Internetwork. PhD thesis, Rutgers University, Department of Computer Science (2001)
3. Jiang, C., Steenkiste, P.: A hybrid location model with a computable location identifier for ubiquitous computing. In: Proceedings of the Fourth International Conference on Ubiquitous Computing (UbiComp 2002). LNCS 2498, Göteborg, Sweden, Springer (2002) 246–263
4. Leonhardi, A., Rothermel, K.: Architecture of a large-scale location service. Technical Report TR-2001-01, University of Stuttgart, Institute of Parallel and Distributed Systems (2001)
5. Hightower, J., Borriello, G.: Location systems for ubiquitous computing. Computer **34** (2001) 57–66 IEEE Computer Society Press.
6. Department of Defense: WGS84 Military Standard. `http://164.214.2.59/publications/specs/printed/WGS84/wgs84.html` (1994)
7. Ward, A., Jones, A., Hopper, A.: A new location technique for the active office. IEEE Personal Communications **4** (1997) 42–47
8. Want, R., Hopper, A., Gibbons, J.: The active badge location system. ACM Transactions on Information Systems **10** (1992) 91–102
9. Kainz, W., Egenhofer, M.J., Greasley, I.: Modeling spatial relations and operations with partially ordered sets. International Journal of Geographic Information Systems **7** (1993) 215–229
10. Berners-Lee, T., Fielding, R., Irvine, U., Masinter, L.: Uniform resource identifiers (URI): Generic syntax. IETF RFC 2396 (1998)
11. Brumitt, B., Shafer, S.: Topological world modeling using semantic spaces. In: Proceedings of the Workshop on Location Modeling for Ubiquitous Computing, UbiComp 2001. (2001) 55–62
12. Randell, D.A., Cohn, A.G.: Modelling topological and metrical properties in physical processes. In Brachman, R., Levesque, H., Reiter, R., eds.: Proceedings of the First International Conference on the Principles of Knowledge Representation and Reasoning, Los Altos (1989) 55–66
13. Egenhofer, M.J., Herring, J.R.: Categorizing binary topological relations between regions, lines, and points in geographic databases. Technical report, Department of Surveying Engineering, University of Maine (1990)
14. Bauer, M., Becker, C., Rothermel, K.: Location models from the perspective of context-aware applications and mobile ad hoc networks. In: Proceedings of the Workshop on Location Modeling for Ubiquitous Computing, UbiComp 2001. (2001) 35–40
15. Nicklas, D., Großmann, M., Schwarz, T., Volz, S.: A model-based, open architecture for mobile, spatially aware applications. In Jensen, C.S., Schneider, M., Seeger, B., Tsotras, V.J., eds.: Proceedings of the 7th International Symposium on Spatial and Temporal Databases (SSTD 2001), Redondo Beach, CA, USA (2001)
16. Coschurba, P., Rothermel, K., Dürr, F.: A fine-grained addressing concept for geocast. In: Proceedings of the International Conference on Architecture of Computing Systems 2002 (ARCS '02), Karlsruhe, Germany (2002)