



TKN

Telecommunication
Networks Group

Technical University Berlin
Telecommunication Networks Group

Concept for Hierarchical and Distributed Processing of Area Based Triggers

Sven D. Hermann and Günter
Schäfer

[hermann, schaefer]@tkn.tu-berlin.de

Berlin, October 2005

TKN Technical Report TKN-05-006

This work has been supported by Siemens AG, Mobile Communications in the context of the project "Routing to Geographical Addresses in Vertical Networks/ABT"

TKN Technical Reports Series
Editor: Prof. Dr.-Ing. Adam Wolisz

Abstract

Area based triggers denote notifications being generated if a mobile client changes its spatial relation to a defined area, e.g. if it enters or leaves it. By these triggers, new valuable services can be provided for mobile users.

One of the main challenges is to design a suited approach for the computation of the triggers based on the defined areas and the current whereabouts of the mobile clients. With a rising number of areas and participants of these services, the need for a scalable solution to process the triggers emerges.

This report examines a new approach for the hierarchical and distributed processing of area based triggers based on the aggregation and segregation of both triggers and the setup messages requesting the services in a carrier access network.

We investigate the computational and signaling effort of the approach based on the required effort for an emerging position update of a mobile client in the network. The results are compared analytically with two other approaches, a centralistic and a hierarchical one. Our analysis shows that our approach leads to a superior performance in terms of the effort for signaling and computation.

Contents

1	Introduction	2
2	Related Work	4
3	Assumptions and Proposed Concept	6
3.1	Request Segregation and Distribution to ABT Nodes	7
3.2	Trigger Aggregation	7
3.3	Request Aggregation	8
3.4	Trigger Segregation	8
4	Evaluation of the Approach	9
4.1	Analyzed Approaches	9
4.2	Network Model	9
4.3	Distribution of Requests to Network Stages	10
4.4	Request Model	11
4.5	Model for the Mobile Clients	13
4.6	Centralistic Approach	13
4.7	Hierarchical Approach	14
4.8	ABT Approach	14
5	Performance Discussion	17
6	Conclusion	20

Chapter 1

Introduction

By *location based services* (LBS) [11] [9], users can be provided with useful services and information being related to their current geographical positions. A service can be regionally bounded, making it available for a user depending on the fact that he is in the region or that his distance to a point of interest does not exceed a certain value.

In general, the current systems implementing LBS mainly realize proximity based *pull services*. Depending on the current position of single *mobile client* (MC), they are able to determine the nearest point of interest for a user or to provide services and information after he issued an appropriate request.

While the investigation and rollout of pull services has been widely completed, there is still no unified solution for the provisioning of LBS *push services*, especially in networks with heterogeneous access technologies, even though there are various new applications which could be launched like moving area restrictions, statistical and information delivery services.

Two categories of push services with different application areas can be distinguished. The first one is constituted by the *network initiated services*. They are provided in a spatial region, i.e. broadcasted to all appropriate devices which currently reside in it. Typical applications are warnings, being sent nonrecurring or recurring. Second, there are *user movement initiated services*. Without an explicit request, they are implicitly triggered if a mobile client changes its spatial relation to a certain predetermined area. For instance, a user MC_1 can receive a warning if he enters an area or another user MC_2 gets a notification if MC_1 leaves it.

This report presents a new approach for the provisioning of user movement initiated push services. They require a mechanism which detects the spatial relation of a MC to a requested area. If the relation changes after a MC entered or left an area, special events have to be generated by the mechanism in order trigger appropriate actions (e.g. send a warning or notify another client). The events are denoted *area based triggers* (ABT).

With currently specified and standardized solutions for LBS, it is hardly possible to efficiently realize the envisaged push services. LBS provide services based on the ID of a single MC. Therefore, it requires the movement tracking of all MCs in order to enable service provisioning based on the area in which the clients reside. An obvious drawback is that the tracking of all MCs leads to a high network load for its signaling.

Additional problems arise when the ABTs have to be computed. In general, two different kinds of information are required for this. First, the location of every MC in the geographic territory being covered by the network is required. Second, the information about all requested areas is needed. In

order to generate the triggers, a comparison of the whereabouts of the MCs with the borders of the areas must continuously be performed. If a change of a position matches (at least) one request, an ABT is emitted.

The processing of these comparisons requires an immense computing power. The location information and the requested areas are continually changing, making the process highly dynamical. Someone can easily imagine that this mechanism does not scale for a large amount of requests and MCs in the network if the comparison should be performed by one central entity in the network.

The other extreme, a processing of the area requests close to (or in) the base stations of an access network, i.e. an excessive copying and forwarding of the requests to all involved systems, could be used for the distribution of the computational effort. But this method would lead to a significantly increased signaling overhead for two reasons, first for the distribution of the requests and second for the intersystem communication of last MC whereabouts, especially if requested areas are covered by several systems.

Three main challenges are thus encountered and have to be solved

- An efficient and scalable way is required for the processing of updated position information and area requests
- The mechanism should be distributable to different entities in the network
- The signaling of the position information introduces a high network load. The mechanism should be designed in such a manner that it reduces the signaling amount, although this requirement seems to be in conflict with the last mentioned one

In this report we make the following contributions. In the next chapter, related approaches are presented which can be utilized for the processing of area based triggers. In chapter 3 we present our new approach for the computation of ABTs, which is based on a distributed algorithm. It hierarchically processes area requests and position updates in an access network being divided into several processing stages. The approach is intended for the usage in carrier access networks for e.g. UMTS, GSM or WLAN integrated in a wide area mobile communications network. Chapter 4 describes the evaluation of the approach and the deployed modeling. A probabilistic analysis is presented. It determines probabilities for the distribution of requests for rectangular service areas to the different stages. With these probabilities, the resulting effort for the signaling and computation is calculated. We show that the performance gains of the approach are mainly influenced by the distribution of the requested area sizes. Afterwards, the results are explained in chapter 5. In chapter 6, we draw the conclusions and describe the next steps of the ongoing work.

Chapter 2

Related Work

In this chapter, we present approaches for the processing of ABTs and related mechanisms which could be principally applied to the problem.

Location enabling servers (LES) [10] are designed for the provisioning of area based services. An LES is installed by a cellular operator (i.e. carrier) for its access networks. The location information of mobile clients is continuously transmitted to the server and mapped to a suited coordinate system. A second component, the so-called *geo information system* (GIS) stores the location based information related to its area of operation, e.g. the coordinates of points of interest and service areas. The LES utilizes the longitude/latitude information of both client position and entries in an assigned GIS to offer appropriate services and information. Currently, the LES has no explicit facility for the processing of area based triggers.

Location alert trigger servers [8] offer customers who want to make use of LBS the ability to specify trigger conditions in LBS applications. These conditions are submitted to a so-called LAT server and stored by it. The positions of MCs are transmitted by the carriers in form of traffic data to the server which translates the data into geographical coordinates. This is achieved with appropriate information about the network, called site data. After having performed this additional processing step, the server is able to compare the location information with the requested areas and sends notifications or performs other actions if a trigger condition is fulfilled.

Although not being especially designed for the processing of ABTs, the *instant messaging and presence services* (IMPS) [5] [4] approach of the Internet Engineering Task Force (IETF) can be utilized for it. The mechanism allows principals (i.e. users) to provide a so-called presence service with their presence information. The service accepts, stores and distributes this information. For the processing of area based triggers, it would have to contain the geographical coordinates, allowing another user to request defined presence information. A requestor of an area could specify its request for presence information based on geographical attributes, i.e. the coordinates of an area. *Publish/subscribe systems* [2] [3] could be utilized in a similar way. Providers of ABS content could be considered as publishers, sending the area and the corresponding content to an ABS server. In this case, an MC would act as a subscriber, i.e. a change of his position including the change of the spatial relation to an area would be handled as a new subscription for the related service. For those ABS where e.g. other parties are interested in the whereabouts of facilities, pets or persons, the role of the publisher and subscriber is inverted, i.e. the MC publishes its position and the observing parties would be the subscribers.

The first and second approach presented above are centralistic ones. One entity in a network

collects and processes the location information from all mobile clients, being sent through the network to it. Additionally, the information about every requested area with the specified trigger criteria has to be available to it. They do not scale for a large number of nodes, areas and a large coverage area, due to the required immense centralized processing effort for the comparison of the client's whereabouts with the borders of the areas and the emission of triggers.

The drawback of the IMPS approach is that obviously the presence information of all users has to be available to one presence service for a complete processing of the area requests. In fact, the presence information of different principals can be distributed on several servers all over the network, requiring inter-server communication and does not solve the scalability problems. Pub/sub systems being utilized for the processing are facing the same problems.

In order to overcome the shortcomings of the approaches presented above, we suggest to deploy a hierarchical mechanism for the processing of ABTs which scales for a large number of mobile clients and area requests. The tasks should be distributed to different entities in the network. Additionally, it is desirable to process the location information of MCs as close to the origin as possible, to avoid cumbersome collection and processing of location information from regions where no ABTs are requested.

Chapter 3

Assumptions and Proposed Concept

Usually, if a certain *requestor* (e.g. an ABS content provider or observing party) wants to use a service which requires ABTs, he defines his service and the related trigger conditions. Afterwards, a portal [9] [1] of a suited network carrier is contacted which can provide the trigger notifications, i.e. which possesses the technical facilities (servers) for the computation of the trigger events out of the defined areas and MC whereabouts. The execution of the service is delegated to the carrier.

Instead of processing the service requests of the requestors in one central entity in the network, the requests are distributed to several entities in the network which are able to compute the ABTs, i.e. they compare the position changes of the MCs with the borders of the requested areas and generate the triggers. We call these entities *ABT nodes*.

The ABT nodes are aligned on different stages of the access network. At the lowest stage (i.e. as close to the base stations as possible), a node is only responsible for a small part of the geographical territory being covered by the complete network, e.g. for a certain district of a city. At the top level, one node is responsible for the whole territory. On the stages in between, ABT nodes may be responsible for whole cities, i.e. the unions of their districts, provinces/states etc., which leads to a hierarchical alignment of the nodes. The subdivision may also not correlate with any underlying topological structure. An example of a carrier access network with the definition of uplink and downlink is displayed in figure 3.1.

After the requestor contacted the portal being located near the gateway of the carrier access network and after he issued his request, a special setup message containing the request is directly sent through the network towards the selected geographical territory.

The setup messages are forwarded in downlink direction until they reach the last respective ABT node in the network. During the message forwarding, the nodes are informed that they have to handle trigger events which will be emitted by downlink nodes to them.

Our approach performs a partial trigger processing at each network stage. An event from a node in downlink direction may not constitute the final ABT event but requires further treatment, i.e. information from other downlink node. The information about the requests remains stored until its lifetime is exceeded or until it is explicitly revoked.

The approach is based on four basic tasks, the aggregation and segregation of setup messages as well as area based triggers during the processing in the carrier networks. The next sections describe the mechanisms in detail.

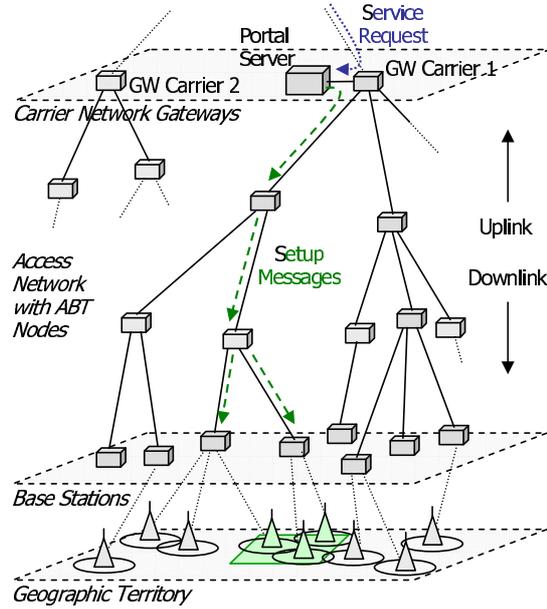


Figure 3.1: Network topology example. The ABTs are computed in special ABT nodes

3.1 Request Segregation and Distribution to ABT Nodes

We assume that the setup messages being send through the network contain the geographical coordinates of the requested area, a unique ID_{req} of the request, the duration, an action that should be performed due to a trigger (with content that should be provided to the triggering MC, if required) and at least one criterion for the trigger (e.g. leave, enter). They are forwarded via a mechanism for routing to geographical addresses, e.g. GeoCast [7].

During the forwarding of the setup message, an ABT node n_{a1} may discover that the requested area is covered by several nodes (e.g. n_{b1} and n_{b2}) in downlink direction. In this case, the routing decision in the intermediate node n_{a1} will force a duplication and forwarding of the message to n_{b1} and n_{b2} being responsible for the subareas, i.e. a *segregation of the setup message* is performed.

Before a segregated setup message is forwarded, the requested action and its assigned content are removed. The segregated messages ask the nodes n_{b1} and n_{b2} to generate basic triggers if the borders of their subareas are crossed due to the emerging movement of mobile clients. These basic triggers are send in uplink direction to node n_{a1} , which has to combine (i.e. aggregate) them.

3.2 Trigger Aggregation

If an ABT node receives basic triggers from nodes in downlink direction for different spatial parts of the same requested area, an *aggregation of triggers* has to be performed. The following items show the required actions.

1. If enter triggers are requested for a certain area, the processing of a basic enter trigger in a node depends on the information if a mobile client with ID_{mc} has left one of the other subareas shortly before. For instance, if an area is covered by two nodes n_{b1} & n_{b2} in downlink direction

and n_{b1} sends the upper level node n_{a1} the information that a certain mobile client entered the subarea being covered by it, n_{a1} has to check if n_{b2} has already announced that the MC left its subarea. In that case, n_{a1} knows that the client did not enter the area like assumed by n_{b1} , because it has formerly already been in it and does not generate an enter trigger.

2. The other way round, the decision if a leave trigger must be generated by a node after receiving a basic leave trigger depends on basic enter triggers for the ID_{mc} from the nodes which control other parts of the same area. If one enter trigger of any other involved router is received in addition to the leave trigger, the mobile client has not left the area and the uplink node does not generate the trigger.

The combination of basic triggers to more complex ones can be performed over several network stages, until they sum up to the originally requested ABT. The processing of the partial triggers ends at least when they reach the first node where the original setup message has been segregated.

3.3 Request Aggregation

During the forwarding of the segregated message, each node keeps a copy of the request. If a new setup message is received, it checks if an ongoing request for the identical area already exists. If this is true, the forwarding is aborted and the request is linked with the existing one. The *requests are aggregated* in this case. If the aggregating node receives a basic trigger for the first request, it interprets this one also as trigger for the aggregated one.

The mechanism is not only utilized for the aggregation of requested areas being completely identical, but also for requests which are identical regarding the part of the geographical territory being covered by a certain ABT node in downlink direction. For instance, a node n_{a1} has forwarded a setup message to two nodes n_{b1} & n_{b2} in downlink direction. Afterwards, a second message arrives, requesting an area being covered by node n_{b2} & n_{b3} , where the part of the area being covered by n_{b2} is identical to the first request. In this case, the new setup message is only forwarded to node n_{b3} .

3.4 Trigger Segregation

If no aggregation of the request could be performed in the intermediate nodes, it finally reaches the last ABT node in downlink direction. These nodes play a central role in the mapping of location information of mobile clients to trigger information. Each node stores the area requests and compares the current whereabouts of the clients with them. A trigger is only emitted by a node, if the change of the mobile node's positions fulfills a requested criterion.

When a trigger criterion is fulfilled, an appropriate trigger event with ID_{req} and the ID of the mobile client which triggered it (ID_{mc}) is sent in upward direction through the network. At this point it should be noted that some service may not require the IDs of the clients, e.g. statistical services. The first ABT node can perform a first step of service processing by only transmitting information of e.g. the amount of clients which leave or enter an area.

During the forwarding of the triggers in upward direction, each ABT node performs a check if the event must be assigned to different requests and independently be processed, which have formerly been aggregated. This is called *segregation of triggers*.

Chapter 4

Evaluation of the Approach

In order to assess the potential performance improvement of our approach, we want to solve the basic question whether and how much gain in terms of processing effort and signaling amount can be achieved if the computation of ABTs is distributed to several entities in an access network. For an initial assessment, we conducted a performance study based on an analytical model that concentrates on the two mechanisms of *request segregation* and *trigger aggregation* at different stages of the network.

4.1 Analyzed Approaches

In the remainder of this report, three different approaches for the processing of ABTs are compared with each other. The first approach to be assessed is a centralistic method. The triggers are computed in one central entity in the network by comparing the borders of the requested areas with the changes of the MC whereabouts. The information about all requested areas and positions converges in this entity. The second one is a purely hierarchical approach. Like in the approach that we presented in this report, ABT nodes are aligned on different network stages. A setup message is forwarded through the network in downlink direction to the node which is responsible for the part of the geographic territory that has been requested, i.e. the forwarding is performed as long as no segregation is required. The triggers are computed at those stages where the forwarding of the messages ends. Afterwards, these two variants will be compared with our approach for the hierarchical and distributed processing of ABTs.

The approaches are analyzed regarding the required amount of signaling information being exchanged on the links and the effort for the computation of the ABTs. We first calculate the average effort for each position update being sent by a MC. Afterwards, we determine the total effort being required in the network per second for a certain number of MCs and an update frequency.

4.2 Network Model

For the analysis, we consider a carrier access network covering a rectangular geographic territory. The three different approaches are assessed without the influence that may result from an alignment of the ABT nodes to the topological structure as described in chapter 3. The part of the territory a node is responsible for is split in north-south and east-west direction in downlink direction, leading to a quartering of the territory at each stage.

The nodes for the processing of the ABTs are aligned on $v = 4$ stages L0 to L3. Stage L0 consists of one ABT node possessing the knowledge about all whereabouts of the MCs which are roaming through a rectangular geographic territory $G = [0, a] \times [0, b]$ being covered by the network. Therefore, the geographic territory G being covered by L0, G_0 , is G . In the stage L1 below, the computation is performed by four different ABT nodes, each being only responsible for a fourth of the territory $G_1(i) \subset G_0$, with $\bigcup_{i=1}^4 G_1(i) = G_0$. Each fourth is divided again into four parts by nodes on stage L2 and each node on stage L3 is responsible for a fourth of a node on stage L2. An example for a resulting network covering a rectangular part of a territory is displayed in figure 4.1.

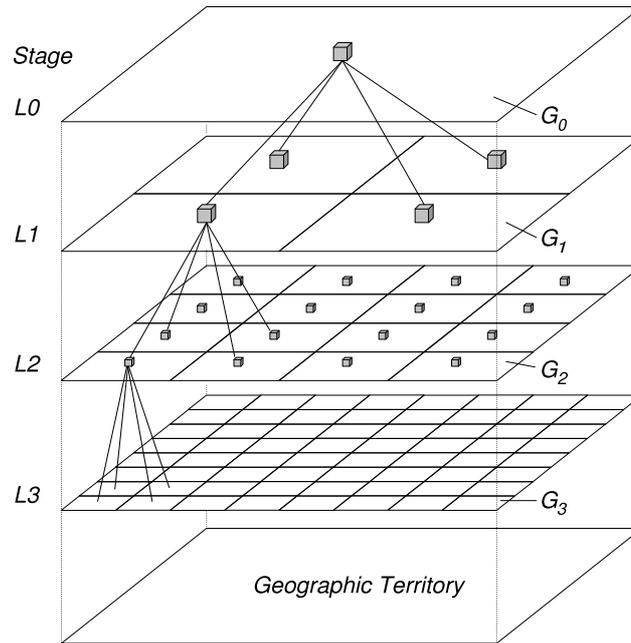


Figure 4.1: The network model. We consider a geographical territory. The ABT processing is performed on 4 stages. On each of them, the workload is quartered

4.3 Distribution of Requests to Network Stages

In the first step, the forwarding of the requests $A \in \mathbb{R}^2$ to the different network stages will be analyzed. A setup message is forwarded through the network in downlink direction as long as the edges of the requested area $a \in A$ do not intersect the borders of the geographical territory $G_m(i)$ a forwarding node i on stage m is responsible for, i.e. as long as $\partial a \cap \partial G_m(i) \neq \emptyset$. The ∂ denotes the borders of an area. This means that the setup message is forwarded as long as the requested area is entirely part of the territory being covered by a node on stage i . Any further forwarding would require a segregation of the setup message.

For this initial model we assume that the requested areas A are rectangular and that their borders are aligned in parallel to those of the covered territory.

If the length of the edges in x and y direction of a certain requested area j , $a_x(j)$ and $a_y(j)$ are given, its center has to be $\frac{|a_x(j)|}{2}$ away from the border dividing the covered area in x direction and

$\frac{|a_y(j)|}{2}$ away from the border dividing the covered area in y direction, respectively.

With the length of the borders of the geographic territory G_x and G_y the probabilities that the edges of the area intersect the territory borders are the following

$$P(a_x \cap G_x \neq \emptyset) = p_x = \frac{|a_x|}{|G_x|} \quad (4.1)$$

and

$$P(a_y \cap G_y \neq \emptyset) = p_y = \frac{|a_y|}{|G_y|} \quad (4.2)$$

if the centers of the areas are uniformly distributed over the territory.

The probability that any border on stage L0 is intersected is therefore given by

$$\begin{aligned} P(\partial a \cap \partial G_0 \neq \emptyset) &= p_{L0} = p_x \cup p_y \\ &= p_x + p_y - p_x p_y \end{aligned} \quad (4.3)$$

In stage L1 the territory is divided into four parts. Therefore, two new borders are introduced, resulting in the probability that an area that has been forwarded by the node on stage L0 intersects a border of L1

$$\begin{aligned} P(\partial a \cap \partial G_1 \neq \emptyset) &= p_{L1} \\ &= 2(p_x + p_y - 2p_x p_y) - p_{L0} \end{aligned} \quad (4.4)$$

In general, the equation for the probability that an area intersects the borders on stage $Lm, m \geq 1$ is

$$\begin{aligned} P(\partial a \cap \partial G_m \neq \emptyset) &= p_{Lm} \\ &= f_1(m, p_x, p_y) - f_1(m-1, p_x, p_y) \end{aligned} \quad (4.5)$$

With the function f_1 , the total probability is calculated that an area intersects the borders on Lm , i.e. without being previously passed through the stages above

$$f_1(m, p_x, p_y) = \begin{cases} 1 & \text{if } \frac{\partial f_1(m, p_x, p_y)}{\partial m} \leq 0 \vee m = v - 1 \\ 2^m(p_x + p_y - 2^m p_x p_y) & \text{else} \end{cases} \quad (4.6)$$

the solution for the partial derivative is

$$\frac{\partial f_1(m, p_x, p_y)}{\partial m} = 2^m \log_e(2)(p_x + p_y - 2^{m+1} p_x p_y) \quad (4.7)$$

4.4 Request Model

This section describes the assumptions for the size and the location of the areas being selected in the setup messages and the kind of requested service.

By the proposed hierarchical and distributed approach for the processing of the ABTs, the amount of segregated and forwarded setup messages highly depends on the size of the requested areas. If only large areas are requested, a huge amount of setup messages will be forwarded to the lower stages,

resulting in an accordingly huge number of requests. The ratio between small and large requested areas becomes thus the main factor in the assessment of the approach.

Therefore, we chose areas having different border lengths. The lengths of the borders are exponentially distributed. For a first study, the resulting p_x will be equal to p_y . We used 100 different area sizes with $p_x \in [0.01, 1]$.

By varying the reciprocal mean value λ of the exponential distribution, the ratio between small and large areas changes. The resulting amount of areas with a certain border length (i.e. with a certain $p_x = p_y$) is computed via

$$|\{a \in A : p_x\}| = |A| \frac{F(p_x, \lambda) - F(p_x - 0.01, \lambda)}{F(1, \lambda)} \quad (4.8)$$

$F(p_x, \lambda) = 1 - e^{-\lambda p_x}$ is the cumulative density function of the exponential distribution.

Afterwards, the resulting distribution of the single areas to the different network stages is computed with equation 4.5. The areas being forwarded to network stage Lm without a segregation are called A_m . The resulting $|A_m|$ for a total amount of $|A| = 10000$ requests are displayed in figure 4.2.

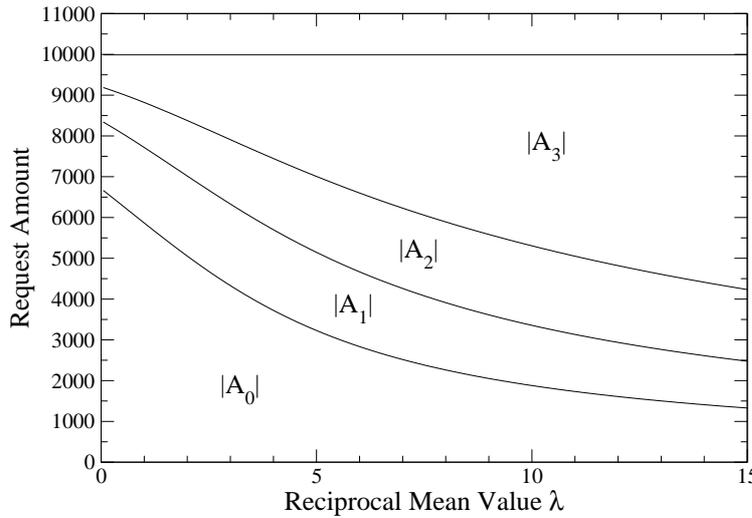


Figure 4.2: The stacked graphs show the distribution of $|A| = 10000$ requests depending on λ

Most of the ABT applications will be intended to notify the user of a MC after he entered an area, i.e. a message will be send to the MC. In this report, we perform the analysis under the assumption that after the generation of an ABT a notification with an average size σ_r will be send by the ABT node to the MC.

In our model for the setup messages, we assume that a request will be substituted N_s times per second in the network by another one. We assume that the substitution will not influence the distribution to the single network stages on average and the $|A_m|$ remain constant.

4.5 Model for the Mobile Clients

For the analysis of the performance of the approach the modeling of the mobile client behavior is widely simplified. We assume an amount of N_{MC} MCs. Each of them generates every t_{pu} seconds a position update consisting of the $MCID$, timestamp, geographical coordinates, status and message type and sends it through the network. The size of a position update is σ_p .

In order to directly compare the different ratios of the area sizes, we assume a constant rate η_{tr} between position updates and ABTs that are triggered in the network, independently from a certain λ .

4.6 Centralistic Approach

In the centralistic approach, the triggers are computed in one central network entity where all the position updates and setup messages converge. In our assumed network architecture this corresponds to the entity on stage L0.

For the calculation of the exchanged signaling information we consider the exchanged signaling bytes and the amount of network links being used. A position update with size σ_p being sent by a mobile client to an entity on stage L3 and between the different stages to the ABT node on stage L0 uses ν links. Therefore, the amount of signaled position information b_p per position update is given by

$$b_p = \sigma_p \nu \quad (4.9)$$

The signaled information for the replies per position update is calculated in an analogous way

$$b_r = \sigma_r \nu \eta_{tr} \quad (4.10)$$

The total exchanged amount of signaling information per second for the centralistic approach, $B_{\text{central}} = N_p(b_p + b_r)$ results in

$$B_{\text{central}} = N_p \nu (\sigma_p + \eta_{tr} \sigma_r) \quad (4.11)$$

where N_p is the total amount of all MCs in the geographic territory times the number of position updates per second.

The computational effort for the processing of a position update depends on the amount of requested areas that need to be checked. With suited algorithms and data structures [6], the effort can be reduced to $O(\log(n) + k)$, where n is the number of requested areas and k the number of candidates which need to be checked. In our case, the required processing effort per position update is thus

$$w_{tr} = \omega_{tr}(\log(|A|) + |X|) \quad (4.12)$$

ω_{tr} is a measure for the required processing effort and $|X|$ the number of candidates. The total processing effort per second is

$$W_{\text{central}} = N_p w_{tr} = N_p \omega_{tr}(\log(|A|) + |X|) \quad (4.13)$$

4.7 Hierarchical Approach

For the hierarchical processing, the additional signaling of setup messages b_s per setup message to the intermediate nodes performing the processing of the position updates on the different stages in the network has to be taken into considerations. Therefore, the resulting signaling amount per second is computed by

$$B_{\text{hierarchical}} = N_p(b_p + b_r) + N_s b_s \quad (4.14)$$

In order to compute the ABTs, the knowledge about the position updates is required in every node on the different network stages. Therefore, equation 4.9 for b_p holds also in this case.

The ratio $|A_m|$ to $|A|$ determines the probability that a request is forwarded to a certain stage m . We obtain the resulting signaling effort per setup messages via

$$b_s = \frac{\sigma_s}{|A|} \sum_{m=1}^{v-1} m |A_m| \quad (4.15)$$

with σ_s being the average size of a setup message in bytes.

With the knowledge about the requested service, a node can directly send the ABT result to the triggering MC in the network. As already stated, the ratio between the triggered ABTs is approximately equal and b_r is thus

$$b_r = \frac{\sigma_r \eta_{\text{tr}}}{|A|} \sum_{m=0}^{v-1} |A_m| (v - m) \quad (4.16)$$

The effort for the computation of the area based triggers depends on the number of stored requests and the amount of requests that need to be checked. On stage m of the network, the number of request is allocated to 4^m nodes. The effort is calculated similar to equation 4.13, but is constituted by the sum of the portions on every network stage

$$W_{\text{hierarchical}} = N_p \omega_{\text{tr}} \sum_{m=0}^{v-1} \left(\log \left(\frac{|A_m|}{4^m} \right) + \frac{|A_m|}{|A|} \cdot \frac{|X|}{4^m} \right) \quad (4.17)$$

4.8 ABT Approach

In the ABT approach, all requests are forwarded in downlink direction to stage $Lv - 1$. In the remainder of this report, we use the following notion for the forwarded areas. S_m are the requests being forwarded by stage Lm . Due to the fact that all setup messages are forwarded, S_m is equal to A_m in the ABT approach. With $S_{i,j}$ we denote the resulting segregated requests on stage j that have been forwarded by stage i . In general, $S_{i,j} = \emptyset \forall j \leq i$.

During the forwarding process, the setup messages are segregated, i.e. the number of resulting requests increases. We first have to determine the amount of resulting requests on stage $m + 1$, if a request is forwarded by an ABT node on stage m . If the requested area intersects one border of $G_m(i)$, two segregated setup messages have to be sent in downlink direction. If two borders are intersected, four have to be sent. The probability that an area with p_x intersects two borders is calculated via

$$f_4(m, p_x, p_y) = \min \left\{ 1, \frac{f_3(m-1, p_x, p_y)}{f_1(m, p_x, p_y) - f_1(m-1, p_x, p_y)} \right\} \quad (4.18)$$

Function f_3 is the probability that an area intersects two borders referring the total area

$$f_3(m, p_x, p_y) = \min \{1, 4^m p_x p_y\} \quad (4.19)$$

If a request has to be forwarded through several stages, e.g. from L0 to L3, the amount of resulting requests is approximated with

$$S_{m,m+3} = 2(\lceil 8p_x \rceil + \lceil 8p_y \rceil) \quad (4.20)$$

This means the ABT nodes being responsible for the borders of the requested area receive the setup message. If requests are forwarded from Lm to $Lm + 2$, the number of resulting requests is approximated with

$$S_{m,m+2} = 2(\lceil 4p_x \rceil + \lceil 4p_y \rceil) \quad (4.21)$$

In figure 4.3 the resulting amount of segregated and forwarded requests on stage $v - 1$ is displayed.

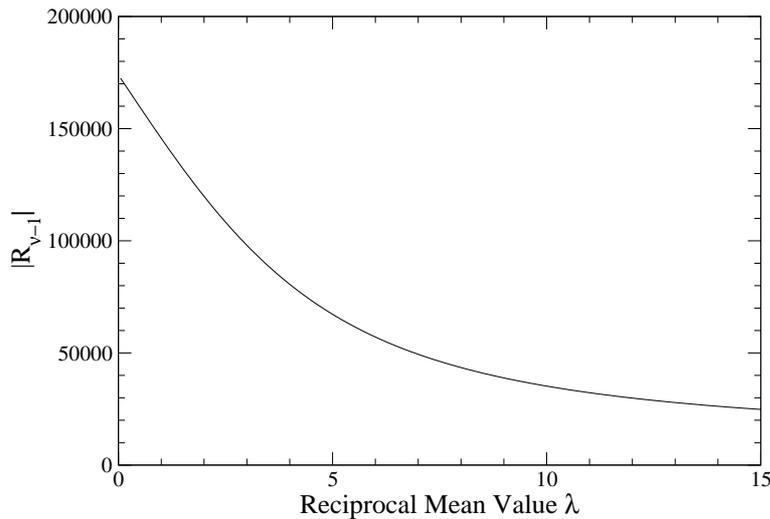


Figure 4.3: The graph shows the resulting amount of segregated and forwarded requests $|R_{v-1}|$ on stage $v - 1$ for $|A| = 10000$ requests

With the knowledge about the forwarded requests, the resulting signaling and processing effort can be computed.

Similar to the hierarchical approach, the total amount of exchanged bytes over the links consists of the position updates, replies and setup messages. Additionally, the signaling amount b_{pt} for the partial trigger information has to be considered. Thus, the total amount is

$$B_{ABT} = N_p(b_p + b_r + b_{pt}) + N_s b_s \quad (4.22)$$

Due to the fact that all setup messages have been forwarded to stage $Lv - 1$ in the network, the original position information from the MCs is only required on that stage, i.e. the resulting signaling

amount for one position update is

$$b_p = \sigma_p \quad (4.23)$$

In the ABT approach, the partial triggers being generated in the areas where the segregated setup messages have been sent are aggregated at the stage where the original setup message has been forwarded to without segregation. The amount b_r is therefore identical to the hierarchical approach, and equation 4.16 also holds.

The same is true for equation 4.15 determining the amount b_s , but the segregated setup messages have to be taken into consideration. The resulting amount of setup messages is the sum of all messages being sent from a stage to all of the stages below it. Therefore, the resulting signaling amount for the setup messages is

$$b_s = \frac{1}{|A|} \left(\sigma_s \sum_{m=1}^{v-1} m|A_m| + \sigma_{ss} \sum_{m=1}^{v-1} \sum_{l=m+1}^{v-1} |S_{m,l}| \right) \quad (4.24)$$

We introduce σ_{ss} as the average size for the segregated setup message at this point. An ABT node performing the computation of partial ABTs in an area being requested by a segregated setup message is not sending the reply to the MC. Therefore, the content is not part of the segregated message, i.e. $\sigma_{ss} = \sigma_s - \sigma_r$.

If a setup message is segregated, it will result in several requests on the network stages below. Simultaneously, the number of partial triggers being generated in the requested area parts increases. In the worst case, a MC roams through all area parts of the same request, generating partial triggers, before the original trigger is fulfilled. We approximate the amount of partial triggers with

$$b_{pt} = \frac{\sigma_{pt} \eta_{tr}}{|A|} \sum_{m=1}^{v-1} \sum_{l=m+1}^{v-1} |S_{m,l}| \quad (4.25)$$

The constant σ_{tr} is the size of a partial trigger. A partial trigger does not have to contain all information a position update has to, but at least the MC_{ID} , timestamp, Req_{ID} and flags.

In order to calculate the processing effort for the ABTs in our approach, the effort for the generation of the partial triggers w_{tr} on stage $Lv - 1$ and aggregation of the triggers w_{ag} on all stages on the network has to be considered

$$W_{ABT} = N_p(w_{tr} + w_{ag}) \quad (4.26)$$

Since the setup messages have been forwarded to stage $Lv - 1$, the resulting effort for the processing of the triggers is according to equation 4.12

$$w_{tr} = \omega_{tr} \left(\log \left(\frac{|R_{v-1}|}{4^{v-1}} \right) + \frac{|X|}{4^{v-1}} \right) \quad (4.27)$$

With R_{v-1} we denote the resulting set of requests that are handled on stage $Lv - 1$ or that have been forwarded to it

$$R_{v-1} = A_{v-1} \bigcup_{m=0}^{v-2} S_{m,v-1} \quad (4.28)$$

The effort for the aggregation of the triggers is from a computational point of view quite low in contrast to the computation of a trigger based on a position update. It required the lookup in a database for all segregated requests and a second lookup if a partial trigger from a MC has been received

$$w_{ag} = \frac{\omega_{ag} \eta_{tr}}{|A|} \sum_{m=0}^{v-2} \left(\log \left(\frac{1}{4^m} \left(|S_m| + \sum_{l=0}^m |S_{l,m}| \right) \right) + \log \frac{N_{MC}}{4^m} \right) \quad (4.29)$$

Chapter 5

Performance Discussion

In this section we evaluate the metrics which have been derived in the previous sections. Table 5.1 shows the actual parameters from existing technologies that have been chosen for the analysis together with those that we assume for our analysis. With the assumed position update interval t_{pu} and the ratio between triggers and position updates η_{tr} , each MC generates two triggers per hour on average. Furthermore, we assume that 2 setup messages are send every minute. The size of the reply is based on the mean value of an SMS and MMS message. The size of the setup message has been chosen with respect to the additional data for the service management (e.g. accounting).

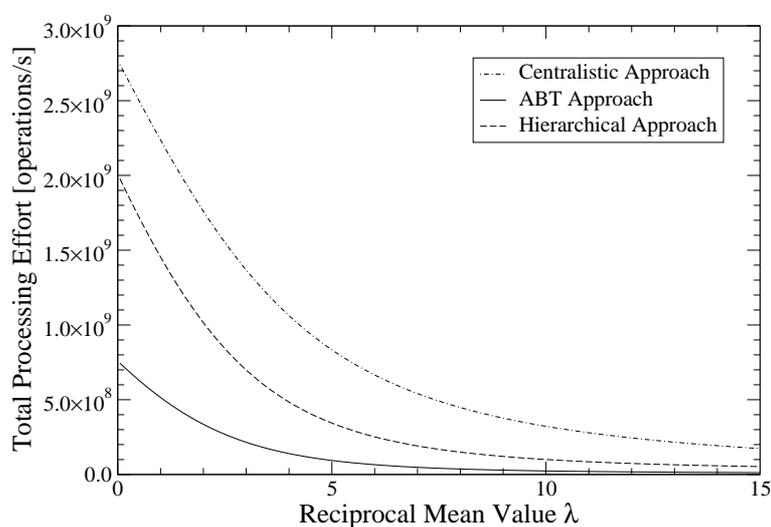


Figure 5.1: The processing effort for the three approaches

In figure 5.1 the computational effort is displayed which would be required for the realization of the ABTs in relation to the reciprocal mean values λ and the resulting distribution of the requests to the different stages. In the worst case, i.e. for small λ being equivalent to a great number of large areas, the required effort can be reduced to about a three quarters with a purely hierarchical process-

Table 5.1: Parameters for the evaluation of the approaches

Parameter	Value
Number of network stages ν	4
Requested areas A	10 000
Mobile clients N_{MC}	1 000 000
Position update interval t_{pu}	10 s
Amount of setup messages N_s	1/30s
Amount of position updates N_p	100 000/1s
Ratio triggers/position updates η_{tr}	1/180
Setup message size σ_s	10 000 bytes
Size of a reply σ_r	5 070 bytes
Size of a position update σ_p	38 bytes
Size of a partial trigger σ_{pt}	21 bytes
Processing of a position update ω_{tr}	24 operations
Aggregation of a trigger ω_{ag}	3 operations

ing of ABTs. With the hierarchical and distributed approach including segregation and aggregation presented in this report, the amount can finally be reduced to nearly a fourth of the centralistic one. With increasing λ , the amount of exchanged bytes decreases. The ratio between the centralistic and both hierarchical approaches also decreases with increasing λ . In figure 5.2, the right section of the graph with $\lambda \in [13, 15]$ is displayed. As can be seen, the computational effort is approximately 7.7 % of the centralistic one for $\lambda = 15$.

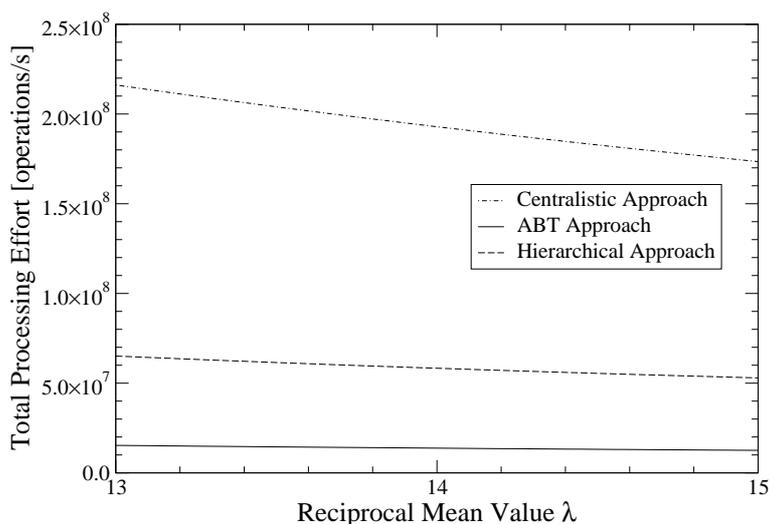


Figure 5.2: Enlarged section of graph 5.1

In figure 5.3 the totalized amount of bytes being exchanged over the links in the network is displayed. The amount for the centralistic approach depends on the number of mobile clients and requested areas in the network and remains therefore constant for different distributions of area sizes.

The graph shows that for small λ , the savings of the hierarchical approach compared to the centralistic one are relatively low with about 6% for $\lambda = 0$ and increases only to approximately a fourth. The analysis shows that a purely hierarchical approach significantly reduces the processing effort for the ABTs but provides only small gains regarding the exchanged signaling information compared with a centralistic approach.

The results for the new ABT approach show that the signaling amount can be further reduced to about the half of the centralist one in the worst case. Like for the hierarchical one, the amount decreases with increasing λ .

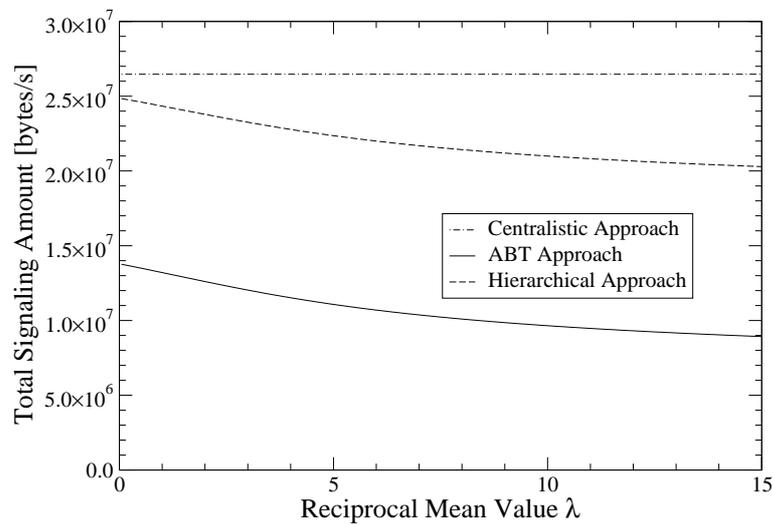


Figure 5.3: The total signaling amount of the three approaches

Chapter 6

Conclusion

In this report we presented a new approach for distributed computation of ABTs in different stages of a carrier network. Trigger events are derived by the segregation and aggregation of setup messages and partial events.

Our proposed mechanism offers a significant reduction of both exchanged signaling information and required effort for the processing of ABTs.

The analysis of the approach and the comparison with the centralistic and hierarchical one presented in this report has solved the basic question if a gain can be achieved regarding processing effort and signaling amount, although the split-up of the setup messages leads to huge multiplication of the amount of requested areas (see figure 4.3).

The models for the network, requested areas and mobile clients that have been chosen in this study are intended to give an impression about the order of magnitude by which our approach enhances others.

In following studies, more complex models will be used to get detailed insights about the capabilities of our approach, e.g. for certain mobility behaviors of the MCs and the probabilities for the triggered ABTs. Additionally, we will investigate further mechanisms for the reduction of signaling and processing effort under the aspects of applicability in networks with heterogeneous access technologies.

By the applied analytical approach, the hierarchical and distributed processing of ABTs has been investigated regarding the capabilities of the setup message segregation and trigger aggregation. With detailed models for the requested areas, possible gains by the aggregation of setup messages and the segregation of trigger messages will be analyzed by means of analytical modeling and simulation.

Bibliography

- [1] Open service access (osa); application programming interface (api); (parlay 5). ES 203 915, European Telecommunications Standard Institute (ETSI) and Parlay Group, April 2005.
- [2] M. K. Aguilera, R. E. Strom, D. C. Sturman, M. Astley, and T. D. Chandra. Matching events in a content-based subscription system. In *Proc. Symposium on Principles of Distributed Computing*, pages 53–61, 1999.
- [3] G. Banavar, T. Chandra, B. Mukherjee, J. Nagarajarao, R. E. Strom, and D. C. Sturman. An efficient multicast protocol for content-based publish-subscribe systems. In *Proc. 19th IEEE Intl. Conf. on Distributed Computing Systems*, Austin, Texas, USA, May 1999.
- [4] M. Day, S. Aggarwal, G. Mohr, and J. Vincent. RFC 2779: Instant messaging/presence protocol requirements, 2000.
- [5] M. Day, J. Rosenberg, and H. Sugano. RFC 2778: A model for presence and instant messaging, 2000.
- [6] Mark de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry. Algorithms and Applications*. Springer, second edition, 2000.
- [7] T. Imielinski and J. C. Navas. Gps-based addressing and routing, November 1996.
- [8] Mobilaris. Mobilaris location alert trigger server m-lats. http://www.mobilaris.com/download/Mobilaris_LBS_Trigger_solution_Feb2004.pdf, February 2004.
- [9] Jochen Schiller and Agnes Voisard. *Location-Based Services*. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, 2004.
- [10] Siemens. Location enabling server. http://communications.siemens.com/cds/frontdoor/0,2241,hq_en_0_67965_rArNrNrNrN_prodId%253A67965_prodParentId%253A68643,00.html.
- [11] Location-based services system (lbss). Technical Report Version 1.0.0, Technical Specification Group - Service and System Aspects (TSG-S), sep 2000.