SAPPORO
Final Report

Intelligent Traffic Control for Urban Networks
Development of AI Concepts for Traffic Management

by

Bernd Wild, FZI
Axel Saffran, FZI
Ekkehard Arleth, FZI

November 1992

FZI report 10

Sapporo Final Report

November 1992

Intelligent Traffic Control for Urban Networks

Development of AI Concepts for Traffic Management Systems

Joint Basic Research Project
NTT Data Communications Systems
FZI Forschungszentrum Informatik
Dept. Technical Expert Systems and Robotics
Prof. Dr. U. Rembold

Authors
Bernd Wild, Axel Saffran and Ekkehard Arleth

Contents

1 Introduction

1.1 Synopsis

This report reflects the work done in the 3rd phase of the project which started in April 1992 and ended November 15th, 1992.

The research work followed the accepted workpackage plan as described in the addendum to the contract of April, 1992 (see also section 1.2). Major topics have been the completion of the ***Sapporo*** prototype, the introduction of logical sensors and the integration of the signal control architecture. Various techniques and methods of the domain of AI, DAI, OOP and GUI have been used. Except the complete realization of the distributed control architecture based on actors and the contract net protocol all working topics are implemented in the ***Sapporo*** prototype which will be handed over to NTT Data. During the on-site installation all details about the concept, design and implementation of ***Sapporo*** can be explained using the sources, publications and technical documents to achieve a complete knowledge transfer.

The character of this document is that of a progress report. The basics of all topics described in this report have already been discussed in previous reports. Therefore, only the work done in the 3rd phase will be reflected in this paper. Most of that work was software engineering using the ***Sapporo*** prototype which hence greatly influenced this report.

1.2 Project Workpackages of 1992

The workpackages for the 3rd phase of the project are given below:

- **Completion of the prototype *Sapporo***
  The current user interface and simulation algorithm still needs testing and debugging in order to increase functionality and performance. Moreover the required port from the Symbolics platform to Sun/Allegro will be part of the work.

- **Implementation and integration of the signal control architecture**
  The signal control architecture will be further refined by theoretical work and implemented during the 3rd phase. The actor concept and the DAI multi-agent approach seem to be promising for the design of a flexible and easy adaptable hierarchical control structure.

- **Design of a logical sensor interface for integration of different types of sensors and sensor data into the traffic model**
  Up to now no sensor data are considered in the model. Due to the uncertainty about the final sensor interface we will start work on a logical sensor component which serves as an interface between arbitrary types of sensors and multiple schemes for sensor data update.

- **On-site handing over, installation and training**

1.3 Results

Main results of the 3rd phase have been

- the introduction of a new and more powerful data and object model based on associative object networks with persistency.
  This was achieved by a complete re-implementation of the class-instance model of the software prototype of ***Sapporo*** Version April 92 by using the CLOS-DB software [Arleth 92]. The CLOS-DB system is in the public domain and was developed at FZI.

- the design of a logical sensor interface for the inclusion of different sensor types in ***Sapporo*** for later embedding of the system.
  A generic sensor object has been defined and implemented in ***Sapporo*** which allows the monitoring of simulation data at arbitrary locations on the network. Manipulation and browsing of sensor objects can be done interactively. The sensor object can be coupled with the signal plan control module.

- the design of exception points for modeling flow obstacles.
  Flow obstacles like road works, accidents, detours and speed limits are frequent incidents in a realistic road traffic scenario. In order to be able to cope with these events the concept of exception points has been introduced into **Sapporo** . Exception points are logically identical with node objects whereby they have only restricted persistency. In contrast to “real” nodes exception points can be dynamically created and deleted. Even an integration in the persistent network representation is possible due to the new object model. **Sapporo** offers a variety of tools for manipulation of exception points.

- the design and integration of the floating cars concept.
  As already described in *Annual Report April 92* the floating cars concept is a valuable tool for measuring the quality of a traffic control regime. The main problem of the floating cars concept is that it represents a microscopic model while the density calculus of **Sapporo** is a qualitative and hence macroscopic model. By using an actor-based approach and a tight integration in the DEVS scheme the user has now the ability to define cars with certain characteristics and let them move along a pre-calculated route. **Sapporo** offers an information panel for each car instance which monitors traffic relevant parameters to the user and to the signal plan control module. In a real world embedded system the data would then come from real cars equipped with sensors and transponders.

- the design and more conceptual work on the distributed traffic control architecture.
  The traffic control module is a separate module of **Sapporo** . It uses the simulation and topology data for the calculation and selection of signal plans. One of the goals of the project is the realization of a distributed traffic control architecture. For this purpose the DAI approach *contract net* was chosen where actors represent autonomous agents for signal plan calculation, signal plan selection, density estimation, delay estimation, coordination and reconfiguration.
  A first implementation of the signal control module was undertaken in the 3rd phase but a final and stable version couldn’t be achieved. This should be part of a continuation and of further research at FZI.
  Nevertheless, all nodes in **Sapporo** have now signal plan objects attached to them. Even non-signalized intersections are controlled by a signal plan (one-stage). On an abstract level a signal plan is only a time-dependent flow connection matrix for an intersection. This gives the user a homogeneous view onto the control of the traffic flow at networks. Moreover, in the actual **Sapporo** prototype all signal plans can be browsed and edited interactively and multiple signal plans can be attached to node. The signal plan objects can be created from scratch or by the rule-based signal plan generator of the Intersection Designer tool.

1.4 Publications

During the 3rd phase of the ***Sapporo*** project the following publications have been made which are related to the project:

- **[Wild et al. 92]**
  FZI Project Report ***Sapporo***
  FZI June 1992
  (The official version of the *Annual Report April 92*).

- **[Wild 92]**
  Sapporo – Towards an Intelligent Integrated Traffic Management System
  Proc. of the 1st International Conference on Artificial Intelligence Applications in Transportation, San Buenaventura, June 1992
  The paper will also be published in a special issue on AI in transportation of the Journal of Transportation Research (due to 1st quarter 93)

The following publication is pending

- **[Wild 93]**
  Künstliche Intelligenz im Verkehrswesen (AI in Traffic Engineering)
  Proc. of HEUREKA conference, Karlsruhe, May 1993 (in German)

1.5 Proposals for Further Research Topics

The major work on traffic modeling and simulation can be regarded as finished. With the exception of the inclusion of more real world details (e.g. exact representation of intersection and lane geometry, extension to public transport and pedestrians, etc.) the basic framework for working with traffic models and making traffic forecasts is operating.

In contrast to the modeling and simulation part the control part is still open for research. Although a control architecture has been proposed and a laboratory system has been implemented there’s still a long way to go for a fully operational traffic control system using DAI and distributed computing technology. It is the intention of the FZI to work on these domains in the next future. Chapter 6.2 presents the perspective for further research from the viewpoint of the FZI: the continuation of the SAPPORO activity with emphasis on DAI-based integrative platform for neatless integration and evaluation of traffic engineering optimization modules. The ***Sapporo*** prototype was only the first step while a number of other steps will follow.

2 Redesign of Data and Object Modelling

2.1 Associative Object Network

The data modeling of an application which should be extensible, flexible and modular is a non-trivial problem. The relations between entities in a system like ***Sapporo*** are usually of different types (e.g. has-as-component, belongs-to, is-endpoint-of, etc.) and we encountered the phenomenon that we had to make modifications to our data model each time we added a new functionality or a new model level. Thus the data model itself was subject to change relatively often. We therefore looked for methods which allow for rapid and flexible changes without leaving the original data structure. The only way to handle such evolving interrelationships between data entities was the use of the object-oriented paradigm in combination with an associative object network.

Similar to techniques developed for semantic networks ([Quillian 68],[Brachman 83],[Shastri 88]) an object has links which are simply pointer to other objects. These links have names which express the semantics of the link. The overall structure of the system can thus be seen as an associative semantic network. By following the links the user can apply reasoning processes on network structures in order to retrieve information about the connections of an object within different contexts. Table 2.1 presents the most important semantic connections among classes in ***Sapporo*** .

The network can become arbitrarily complex but gives the user the flexibility to add new objects, new models and new functionality very quickly. On the following pages the actual associative data model of ***Sapporo*** is shown. According to the 6 model levels the total network is structured into 6 subnets, each reflecting a different view of the system. If a model has interconnections to another model this is illustrated by gray-colored ovals carrying the name of the object to which the connection exists. The gray-tone is a symbol for the model type.

Tabelle 2.1 The most important associations in the ***Sapporo*** system

Association	Multi-plicity	Restrictions	Semantics
`is-member-of`	n : 1	$n \in \{1\ldots N\}$	All instances of classes which can be assigned to a network context have a **`is-member-of`** link to the corresponding network object. Moreover, the **`is-member-of`** association forms the transitive closure of the association **`belongs-to`**.
`belongs-to`	1 : n	$n \in \{1\ldots N\}$	This link exists between two instances while the class of the first instance is an aggregate of the class of the second instance.
`has-as-neighbour`	1 : n	$n \in \{0\ldots N\}$	Association among two networks.
`has-as-forward-section`	1 : n	$n \in \{0\ldots N\}$	Association among roads and the associated sections.
`has-as-backward-section`	1 : n	$n \in \{0\ldots N\}$	Association among roads and the associated sections.
`has-as-end`	1 : n	$n = 1$	Association among sections and nodes.
`has-as-start`	1 : n	$n = 1$	Association among sections and nodes.
`has-as-lane`	1 : n	$n \in \{1\ldots N\}$	Association among sections and lanes.
`has-as-right-neigbour`	1 : n	$n \in \{0, 1\}$	Association among lanes.
`has-as-left-neighbour`	1 : n	$n \in \{0, 1\}$	Association among lanes.
`has-dynamic-rep`	1 : n	$n = 1$	Association between static objects an the dynamic representation in the simulation.
`has-as-source`	1 : n	$n \in \{0, 1\}$	Association among nodes and sources.
`has-as-sink`	1 : n	$n \in \{0, 1\}$	Association among nodes and sinks.
`has-as-density-calculus`	n : 1	$n \in \{1\ldots N\}$	Association between dynamic links and a valid density calculus.

Tabelle 2.1 (cont.)The most important associations in the ***Sapporo*** system

Association	Multi-plicity	Restrictions	Semantics
`has-as-density-value`	1 : n	$n \in \{1...N\}$	Association between traffic state and density values.
`has-as-coordinator`	1 : 1	$n = 1$	Association between network and simulation coordinator.
`has-as-signal-plan`	1 : n	$n \in \{1...N\}$	Association between crossings and valid signal plans.
`is-part-of`	n : 1	$n \in \{1...N\}$	Association between density values and traffic state.

Sapporo implements all these objects via classes and associations between classes. Following the terminology of Rumbaugh et al. [Rumbaugh et al. 91] we differentiate between *associations* as relations between classes and *links* for inter-instance relationships.

2.2 CLOS-DB

In order to keep the network structure persistent, i.e. not to re-build the network completely each time the system is started, we use a proprietary object storage system called CLOS-DB [Arleth 92]. CLOS-DB was developed at FZI and is available in the public domain. It gives the user the ability to store instances, class objects and ordinary lisp entities in textual form by making only slight modification to the class specifications. Therefore, a fully instantiated object network can be saved to text file and re-read transparently to the user. Additionally, CLOS-DB has integrated an instance manager which allows for immediate access to all instances of a given class. Furthermore, associations to other classes can be defined and even indexed in order to achieve a fast access within an object network. Classical database applications with sorted indexes can be realized as well as associative object networks.

The decision to include a database system was motivated by the need of storing and retrieving a complete object network. While this can be achieved by a conventional relational or an object-oriented database a proprietary approach was favored to avoid license problems and to keep the software portable. CLOS-DB is available for Genera and Allegro Common LISP. Ports to other platforms can be done easily.

We re-implemented all class definitions in ***Sapporo*** in CLOS-DB and made nearly all objects persistent. The text files which contain the data representation are transferrable to other

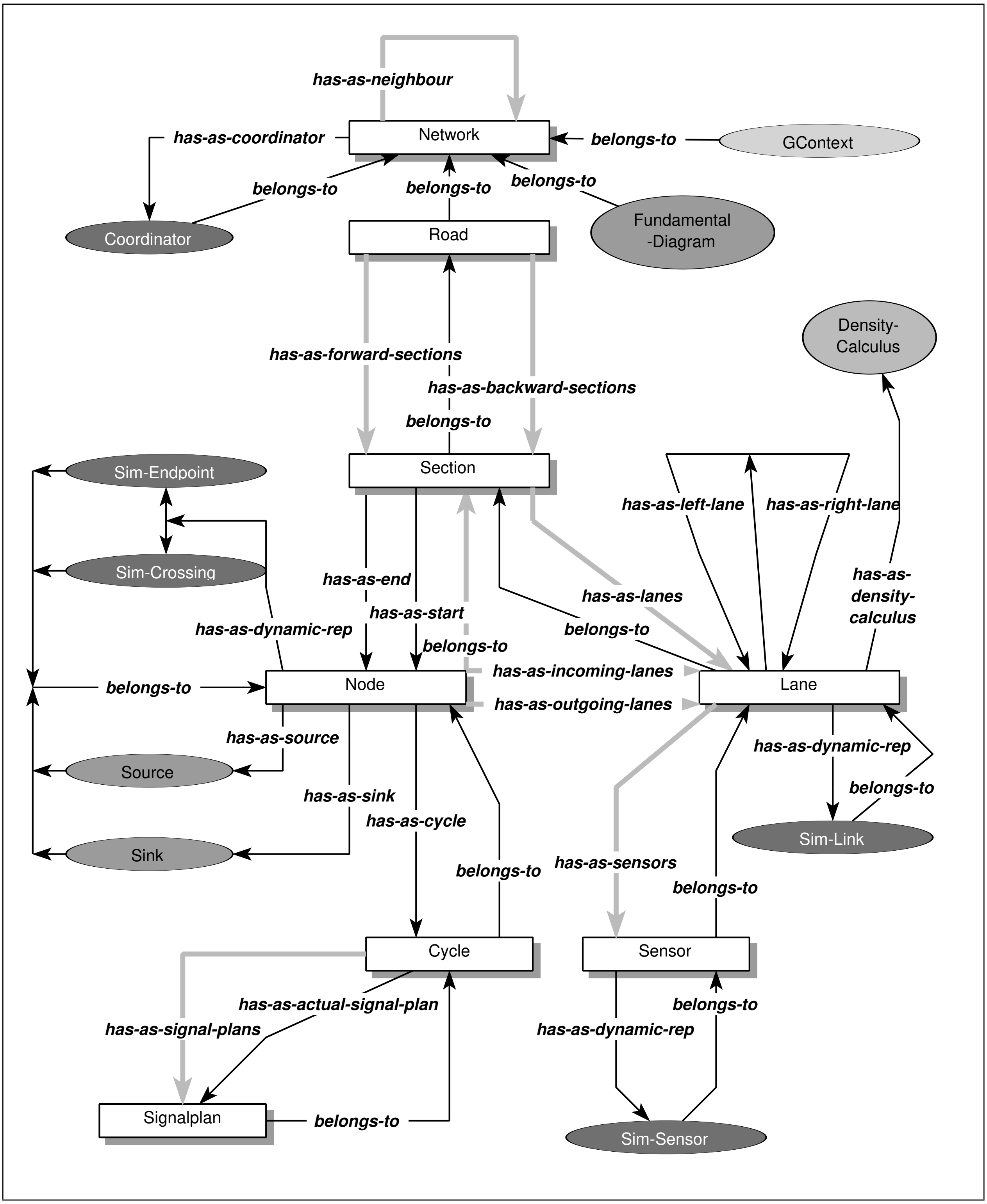


Figure 2.1 The topological object model of ***Sapporo*** . The link labels indicate the semantic connection between objects while the gray-scaled ovals symbolize connections to other models, e.g. traffic engineering model, dynamical model, density calculus model, geometrical model.

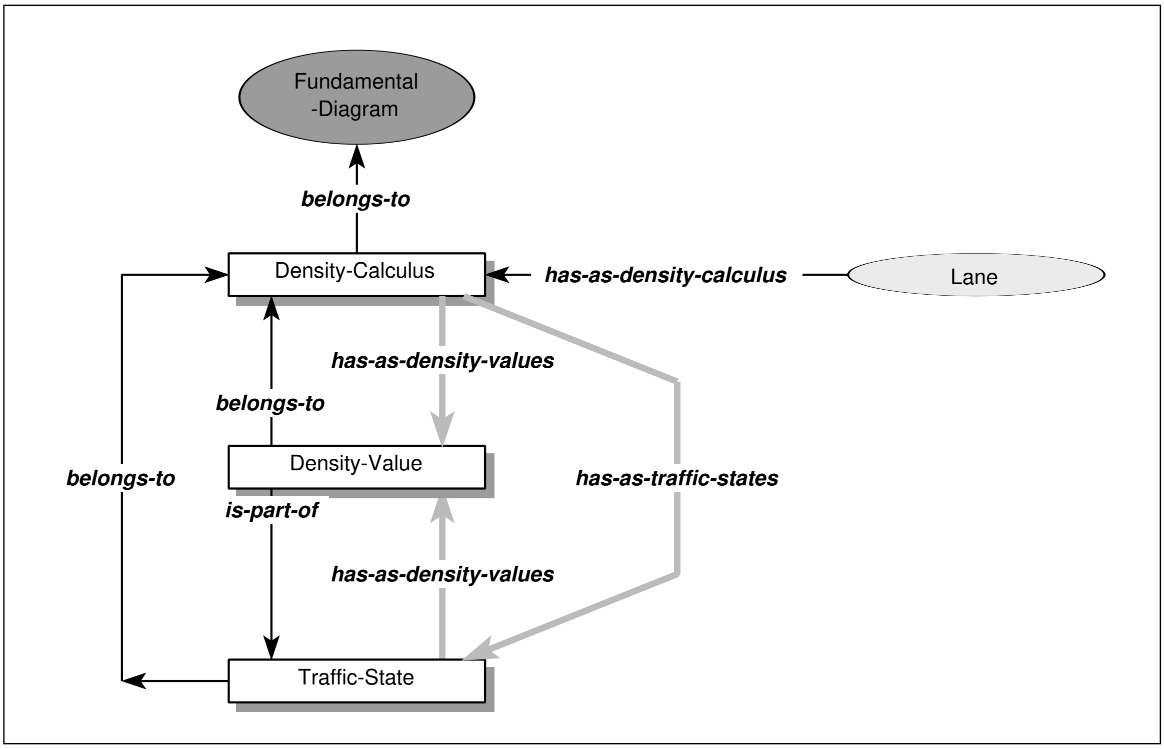


Figure 2.2 The density calculus object model. The link labels indicate the semantic connection between objects while the gray-scaled ovals symbolize connections to other models, e.g. traffic engineering model, dynamical model, topological model, geometrical model.

platforms without problems and the text format is open for other applications. An example for such an application is given in the appendix, where a NeXTSTEP program is used to create, draw or modify object topologies and spool them back to ***Sapporo*** . The user can digitize maps which have been scanned into as TIFF file and can create nodes, sections, lanes and roads. Moreover, the topology can be refined further and can be stored in the format of CLOS-DB. This is a more elegant way to input a road map or to inspect topological data.

2.3 Database Integration

CLOS-DB is made up by several files and represents a sub-system which can be loaded separately. It's a programmer's tool, not a database with an own user interface or query language. The main interface between user code and CLOS-DB is the definition clause for object classes. Beside some additional keywords the only difference between a CLOS declaration and

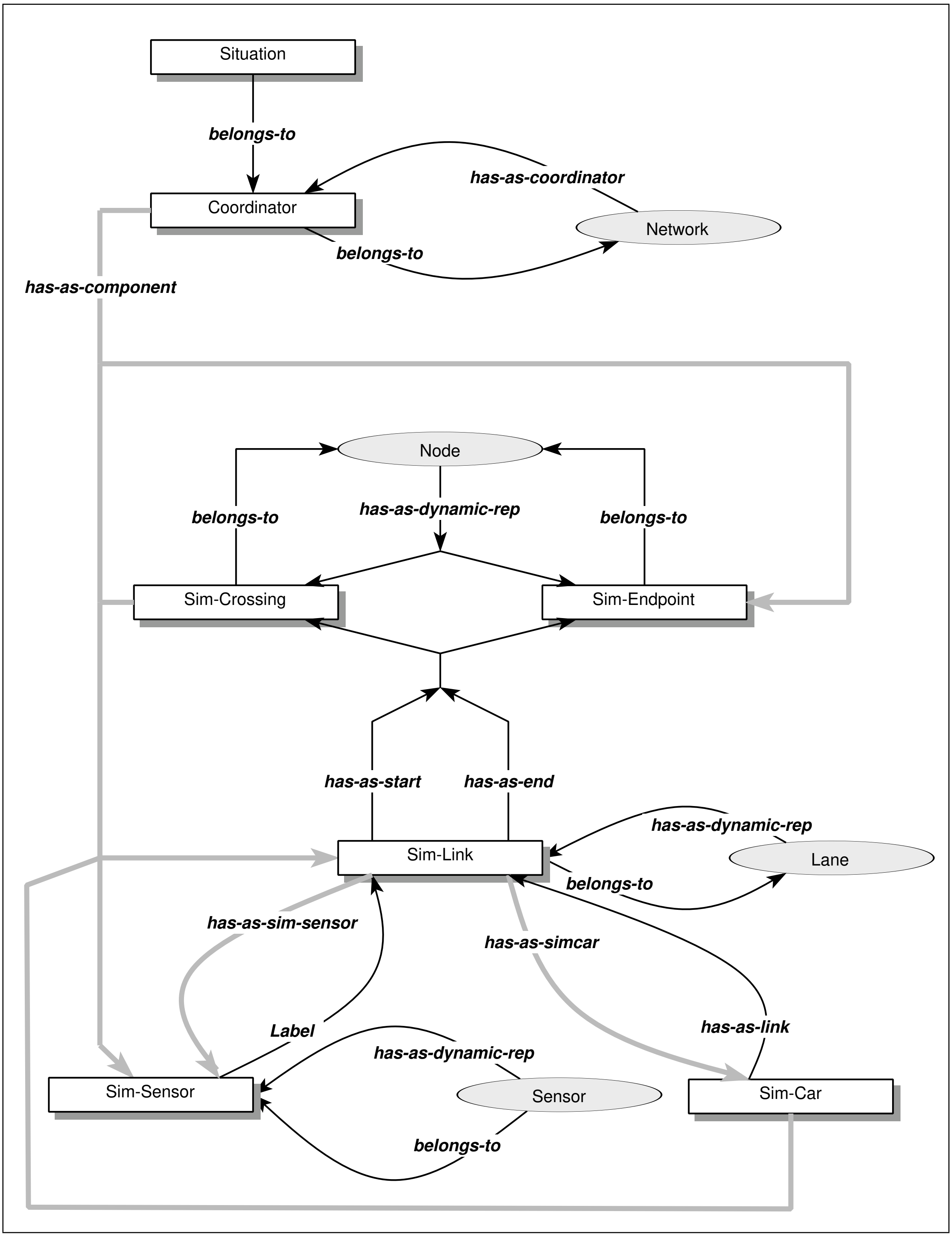


Figure 2.3 The dynamical object model. The link labels indicate the semantic connection between objects while the gray-scaled ovals symbolize connections to other models, e.g. traffic engineering model, density calculus model, topological model, geometrical model.

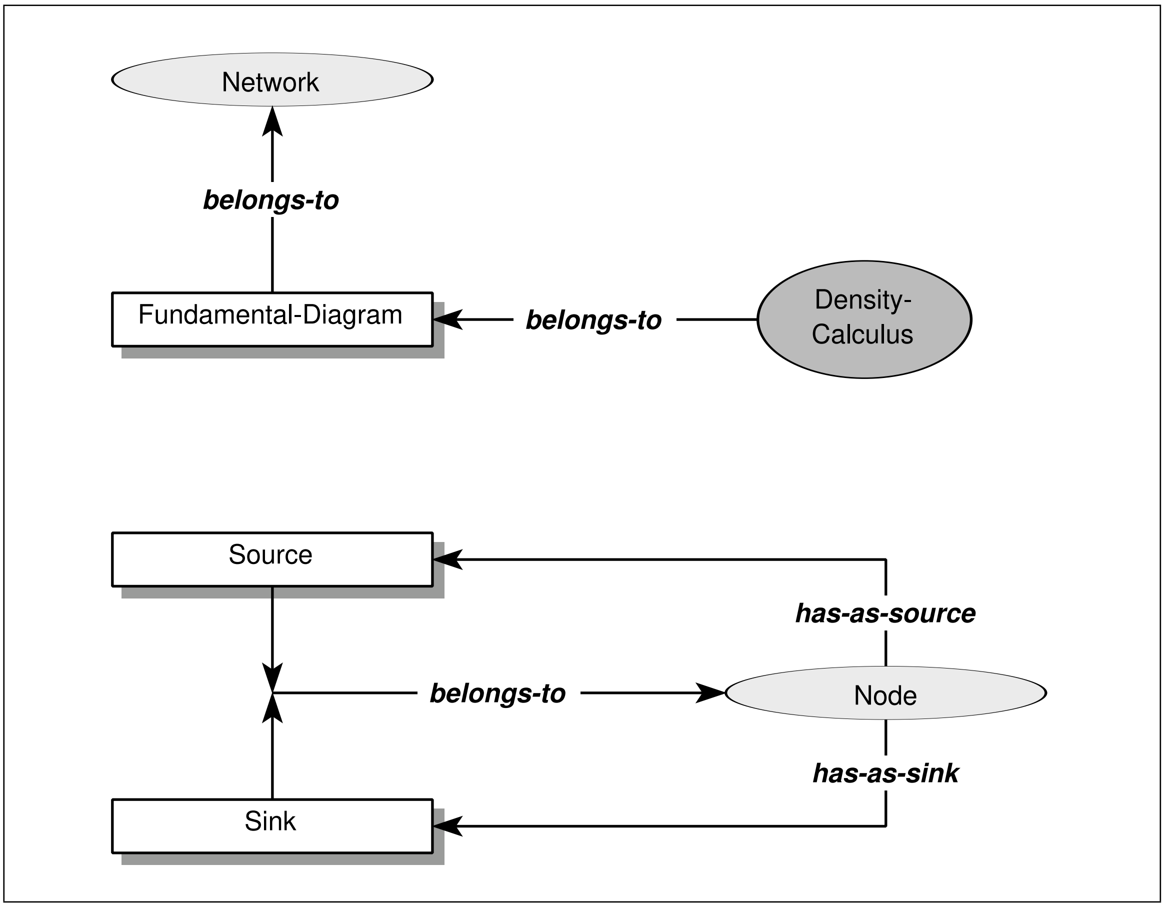


Figure 2.4 The traffic engineering object model. The link labels indicate the semantic connection between objects while the gray-scaled ovals symbolize connections to other models, e.g.density calculus model, dynamical model, topological model, geometrical model.

a CLOS-DB declaration is the **`-db-`** substring which appears in the definition clause. This is illustrated in Figure 2.6.

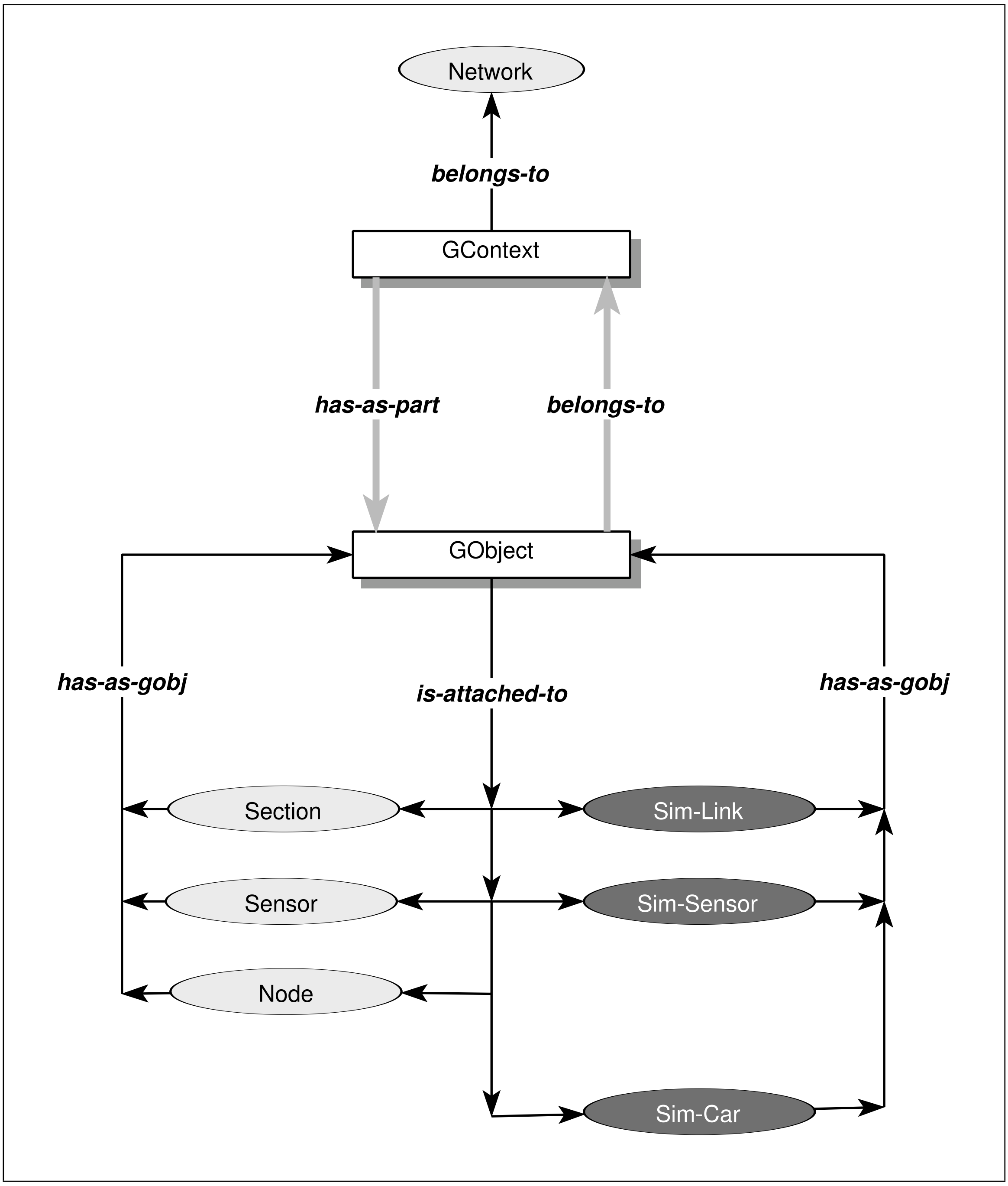


Figure 2.5 The geometrical object model. The link labels indicate the semantic connection between objects while the gray-scaled ovals symbolize connections to other models, e.g.traffic engineering model, dynamical model, topological model, density calculus model.

CLOS

```
(defclass abc
  (super-a super-b)
  ((slot-a … )
   (slot-b … )
  :default-initargs …
 )
```

CLOS-DB

```
(def-db-class abc
  (super-a super-b)
  ((slot-a … )
   (slot-b … )
  :inits (:default-initargs … )
  :persistent <slot-list>
  :non-pers <slot-list>
  :indexed <slot-list>
 )
```

Figure 2.6 The difference between the declaration of a conventional CLOS class (left) and the declaration of the persistent variant in CLOS-DB (right) which is identical to the left one.

3 Logical Sensors

3.1 Requirements

An important aspect for data acquisition in real road traffic are sensors. Sensors can be inductive loops in the road surface, ultra-sonic detectors at the road-side, video cameras positioned overhead or infra-red detectors. The problems with sensors are

- some types are expensive (e.g. video)
- some types are cheap but expensively to maintain (e.g. inductive loops)
- some types have only limited accuracy and failure tolerance (e.g. inductive loops, ultra-sonic, infra-red)
- some types are usable only during specific traffic situations (e.g. inductive loops, ultra-sonic)
- some types require considerable post-processing (e.g. video).

Beside these problems almost all detector types should deliver the same sort of data:

- mean occupancy
- mean density
- mean speed

Only these data can be used for traffic control and traffic optimizing purposes. Therefore, each traffic control system has a component called *data acquisition* which can have arbitrary complexity depending on the detector performance and the accuracy of data analysis.

For ***Sapporo*** no specific sensor types have been fixed[1]. We therefore introduced the concept of the "logical sensor" which represents different types of physical sensors by reducing them onto the data they deliver to the data analysis and control module.

With ***Sapporo*** logical sensors can be used for two purposes

- to re-calibrate a simulation, i.e. to feed in real processed sensor data in order to preset the current situation in accordance with real data[2],
- and to give the user a better representation of the situation at designated points on the network, e.g. profiles and statistics.

3.2 Creation of Sensors

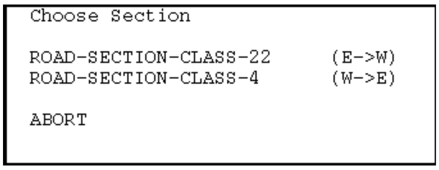
The user can attach a sensor object at an arbitrary location on a lane object simply by a mouse click. Due to the change in the section architecture from ***Sapporo*** Version April 92 to ***Sapporo*** Version November 92[3] the user will then be asked to select the section branch he want to attach to the sensor. In massive symmetric networks this facilitates the attachment process because an exact choice can be made to which lane of the available section objects of a road section the sensor should be linked to. This change is reflected in Figure 3.1. The information to the left of the section names indicates the topological direction of the section, e.g. "E->W" means a section leading from east to west in the map.

Following this the user obtains a list of available lanes of the selected section object (Figure 3.2). The numbering of the lanes correspond to the position of the lane with respect to the inner border. For example, for a section which consists of 4 lanes the lane number 1 designates the innermost lane while lane number 4 specifies the outermost one. Finally, the distance between the start node of the lane and the location where the sensor should be attached to has to be given (Figure 3.3). Additionally a sensor type can be specified, e.g. **`loop`**, **`inductive`** or **`video`**. After this input phase the sensor object will be created and attached to the lane.

[1] This task should have been carried out in a separate work package.

[2] This is a problem which has been proposed as a separate working topic.

[3] A **`section`** encompasses now only the lanes in one direction. Therefore, a bidirectional road section between two nodes consists of exactly two **`section`** objects, i.e. one for each direction.

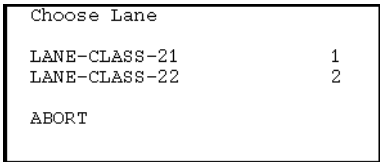


```
Choose Section

ROAD-SECTION-CLASS-22     (E->W)
ROAD-SECTION-CLASS-4      (W->E)

ABORT
```

Figure 3.1 Selection of a section object for the placement of a sensor object.

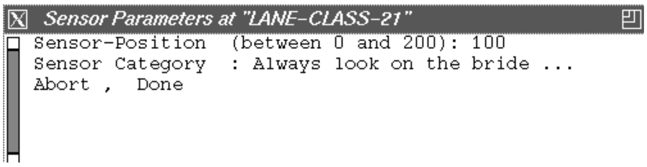


```
Choose Lane

LANE-CLASS-21             1
LANE-CLASS-22             2

ABORT
```

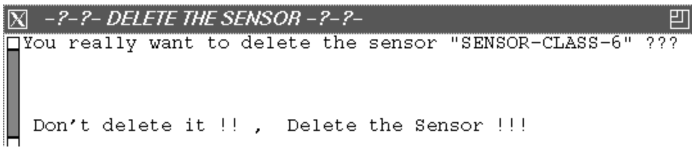
Figure 3.2 Selection of a lane object for the initialization of a sensor object.



```
Sensor Parameters at "LANE-CLASS-21"
Sensor-Position  (between 0 and 200): 100
Sensor Category  : Always look on the bride ...
Abort ,  Done
```

Figure 3.3 Specification of the distance of the location of a sensor object along a lane object.

A sensor object can be deleted (see Figure 3.4), browsed or simulated. Beside the usual “Get Information …” on a sensor object the simulated profile which would be delivered by a real sensor if attached to the lane can be calculated and visualized after a simulation run. For a compilation of all interactive mouse gestures on sensor objects we refer to the Appendix.



-?-?- DELETE THE SENSOR -?-?-

You really want to delete the sensor "SENSOR-CLASS-6" ???

Don’t delete it !! , Delete the Sensor !!!

Figure 3.4 Warning message for the deletion of a sensor.

3.3 Deletion of Sensors

If a sensor object is deleted by the appropriate mouse gesture all objects which are associated with the sensor object are removed, too. This includes the graphics and the simulation object. In order to maintain consistency of the ***Sapporo*** system it isn’t possible to create a situation after the deletion of a sensor.

3.4 Sensors within Simulation

A sensor represents a topological object. Therefore, a dynamic object will be created and associated to the sensor. The dynamic object is of type **`sim-sensor`** and manages all simulation relevant informations. Within a simulation the **`sim-sensor`** behaves as all other simulation objects like **`links`**, **`crossings`** and **`endpoints`**.

If there’s a sensor s attached to position x on link l and the density value at x changes due to the density calculus, then l sends a message to s. The message contains sender l, receiver s, event time t and density value information. The receipt of this message causes an external event at s[1].

The simulated density values at the sensor position can be visualized and browsed by appropriate mouse gestures (see Figure 3.5).

After the insertion of a new sensor the simulation can be started immediately. This is absolutely necessary in order to save the new configuration to file as a situation object.

[1] Please see the description of the DEVS method in the *Annual Report April 92*.

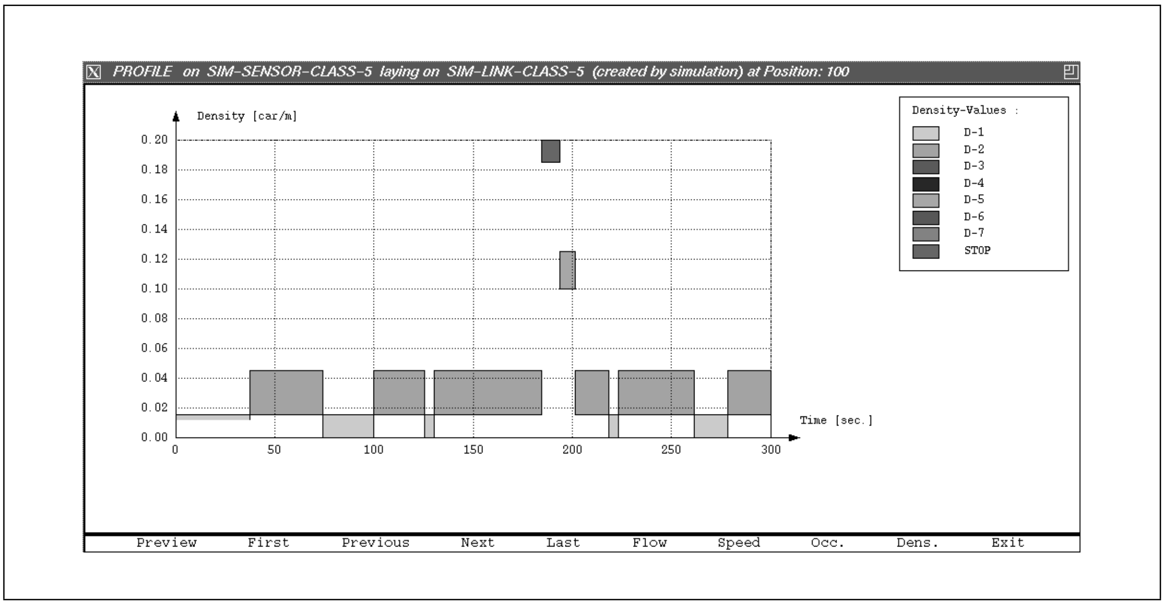


Figure 3.5 Visualization of the simulated density values at a sensor position.

4 Exceptions Points

4.1 Requirements

Road traffic without disturbances, accidents, road works and other incidents is not realistic. To take this fact into account the new version of ***Sapporo*** has a feature called *exception points*.

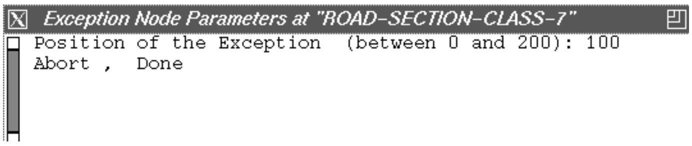
Exception points allow the simulation of exceptional events along a road. Similar to sensor objects exception points can be placed at every location along a road section. Basically, exception points are node objects like ordinary crossings or endpoints. The only difference between crossing nodes and exception points is the type name. With respect to the flow topology an exception point can be handled as a crossing. Thus an exception point has an own signal plan object attached to it and characteristics like turning ratios and lane connectivity. If the user wants to model road works which closes one lane of a 4-lane road (two lanes in each direction) then an exception point is placed at the location of the road works onto the corresponding road section. After the placement the user specifies turning ratios which results in a zero volume inflow to the lane which is blocked by the road work while the remaining open lane receives the complete flow of the two incoming lanes. If the exception has a significant spatial dimension (e.g. an extended road works project) two exception points have to be inserted in order to reflect the decrease and increase of the lane cardinality.

Another application field for exception points are the modeling of incidents which have been detected by sensors, video cameras or emergency units. A blocking of a lane caused by an accident can be modeled analogue to a road works project. The difference of both is usually the total duration of the blocking. While an accidents has a life time in the range of minutes a road works project can last more than a year. In contrast to an exception point representing an accident such a road works exception will be persistent, i.e. will be stored as a situation which will be valid for the next future.

The third target for exception points is the speed or flow limitation along road sections due to speed limits or partially blocked lanes by parking vehicles. If this happens on a regular base such exception points can also be part of a persistent situation object.

4.2 Insertion of Exception Points

The user can insert an exception point into a road section by a mouse gesture. By the dialog shown in Figure 4.1 the parameters for the creation are required from the user.



Exception Node Parameters at "ROAD-SECTION-CLASS-7"

```
Position of the Exception  (between 0 and 200): 100
Abort ,  Done
```

Figure 4.1 Input of the data for the creation of an exception point.

The system then checks if a section in the opposite direction is located in a topological “near” distance to the section the user has clicked on. If yes, the exception point will be inserted in both section objects, the one the user has selected and the opposite section. If not, only the selected section will get an exception point. This feature was included to support road topologies where the sections for the two flow directions are very symmetrical (e.g. normal case) and where there’s a significant distance between both sections (e.g. big alleys, more complex road topologies).

An example for both types of exception points is given in Figure 4.2.

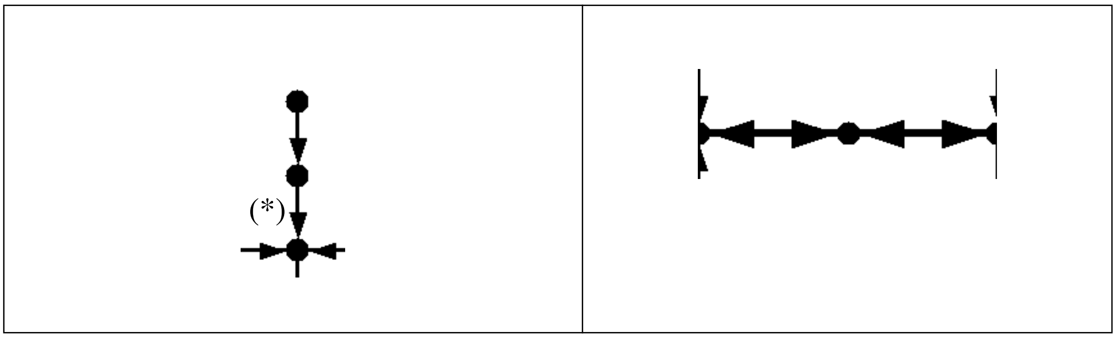


Figure 4.2 Examples for exception points. To the left an EP affecting only one section. To the right an EP affecting both section of a road section. The star indicates the original end node of the splitted section.

The insertion of an exception point into a section object l with start node n_s and end node n_e leads to the following activities (one-section insertion):

- A new node object n_{ep} representing the exception point is created.
- The present section object s gets n_{ep} as the new end node n'_e
- A new section object l' is created. l' has the same attributes as l, e.g. CATEGORY, MAX-SPEED-LORRY, MAX-SPEED-CAR, etc.
- New lane objects are created for l'. The number of lanes is identical to the number of lanes attached to section l.
- The new section l' has n_{ep} as start point n'_s and n_e as end point.
- If sensors were located on the original lanes they will be moved to the new lanes automatically.
- The creation of topological objects like nodes, sections and lanes is followed by the automatic generation of graphical, dynamical and density value objects.

Initially the exception point has 1:1 turning ratios as shown in Figure 4.3. Another feature which can be seen in Figure 4.3 is the ability to edit the flow characteristics of an exception point with the Intersection Designer tool analogue to the modification of a signalized intersection. This is possible due to the handling of an EP as an ordinary node object. Therefore, even the attachment of a signal plan can be done.

The simulation can be started immediately after the insertion of an exception point. Again this is a must in order to ensure consistency of the persistent object representation.

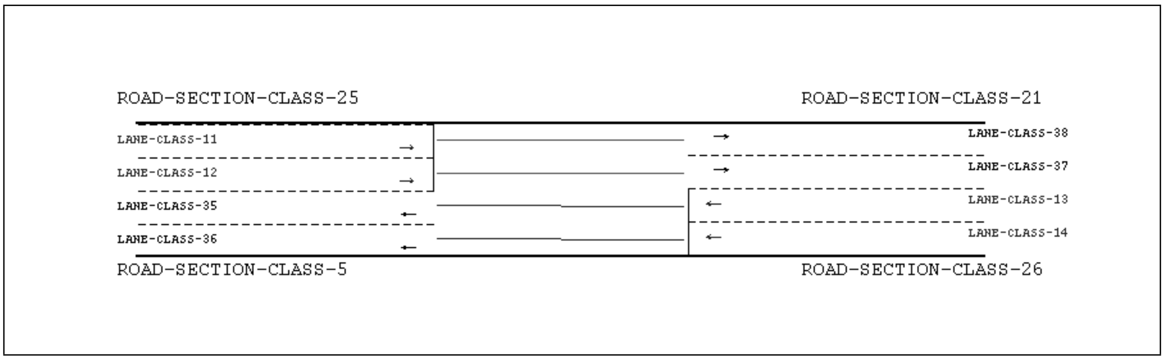
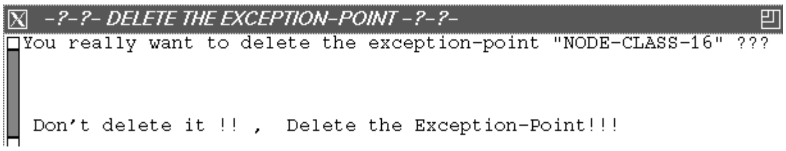


Figure 4.3 1:1-connections between old and new lane objects.

4.3 Deletion of Exception Points

Every exception point can be deleted by a simple mouse gesture. If a deletion is triggered a similar process as described in Section 4.2 is run. Although now the additional section, lanes and dynamic objects are consistently removed. An appropriate warning message is presented to the user (Figure 4.4). The length of the original section is then adjusted and the related graphical object is updated.

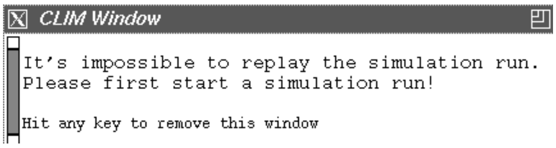

-?-?- DELETE THE EXCEPTION-POINT -?-?-

You really want to delete the exception-point "NODE-CLASS-16" ???

Figure 4.4 Dialog for the deletion of an exception point.

4.4 Exception Points within Simulation

After an insertion of an exception point the simulation can be restarted immediately. A replay of the animated density zone display is not possible until a new simulation run has been executed (see Figure 4.5).



```
It's impossible to replay the simulation run.
Please first start a simulation run!

Hit any key to remove this window
```

Figure 4.5 Warning message in a system state which doesn't allow saving a situation or replay of the density zone display.

During a simulation an exception point behaves exactly like a crossing object (see Figure 4.6).

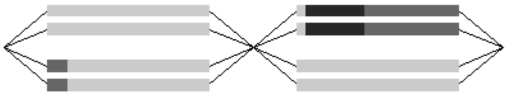


Figure 4.6 Exception point during simulation

5 Floating Cars

5.1 Background

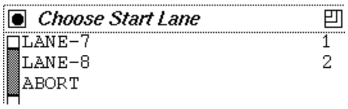
For the determination of quality criteria of a signalized control strategy for a road traffic network not only the simulation results of the qualitative density simulation are significant. Moreover traffic related data from individual cars reflect much more the quality of a control regime than global parameters. Additionally, a flexible control architecture should not only rely on qualitative data but on specific route-related data like mean travel time on a main route through a city or the number of stops encountered along this route. Another criterion is the mean speed which was achieved during such a mission. [Hunt et al. 81] have shown, that *Floating Cars* are the only measurement method for experimental validation of a traffic control regime with a reasonable cost/benefit ratio. Different methods like counting of cars and number plate matching requires a lot more efforts and costs.

An implementation issue of *Floating Cars* is the idea that special cars like taxis, police cars and city staff vehicles which are floating around in the network all the time can be equipped with a dedicated sensor and transponder hardware for continuous acquisition and transmission of moving vehicle data. As this isn't realized up to now in reality floating cars will be simulated in ***Sapporo*** in a microscopic fashion based on the actor paradigm. For more details please we refer to the *Annual Report April 92*.

5.2 Initialization of Floating Cars

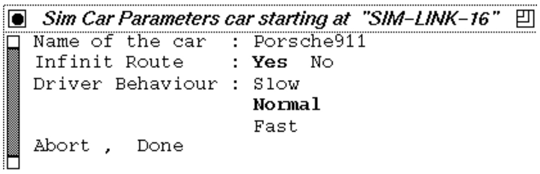
Floating cars are topological objects. Thievery, we need a dynamic representation of the topological object. Similar to sensors and their dynamic counterpart **`sim-sensor`** we introduce the **`sim-car`** as an object which represents the floating car in the simulation. The association between a floating car and a sim-car is **`has-as-dynamic-rep`**.

The user can initialize a floating car simply by clicking on a section object[1]. After the selection of a start lane (Figure 5.1) the user is asked for the floating car parameters (Figure



```
Choose Start Lane
LANE-7                   1
LANE-8                   2
ABORT
```

Figure 5.1 Selection of the start lane for a **`sim-car`**.



```
Sim Car Parameters car starting at "SIM-LINK-16"
Name of the car   : Porsche911
Infinit Route     : Yes  No
Driver Behaviour  : Slow
                    Normal
                    Fast
Abort ,  Done
```

Figure 5.2 Specification of the car parameters.

5.1).

In the actual version the parameters are:

- **Name of the car**
  The name serves as distinction from other **`sim-cars`** on the network.
- **Speed behavior of the driver** (i.e. slow, normal and fast)
  The actual speed of the **`sim-car`** is limited by the density zone within the car actually moves. Only a value in the speed range between lower border speed and higher border

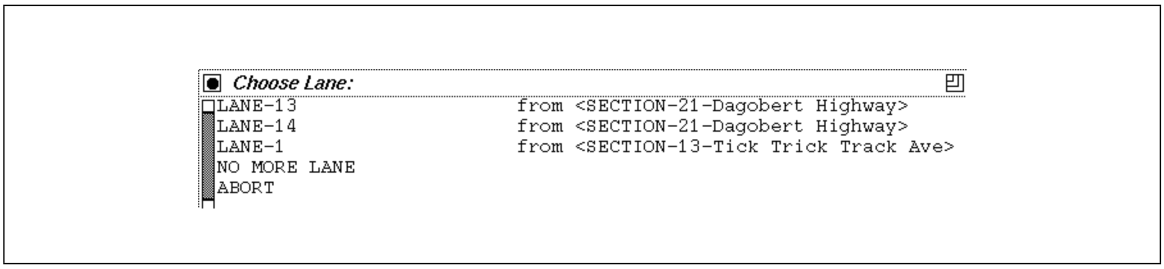
[1] For the appropriate gesture commands please refer to the Appendix.

speed of the density zone can be set. In order to model different driver behaviors the user can specify wether the actual speed of the **`sim-car`** should always be

- in the range between lower border speed and mean speed (slow driver),
- the mean speed of the density zone (normal driver), or
- in the range between mean speed and higher border speed (fast driver).

- **Endless route mode**
  If selected the car restarts at the beginning of the route after reaching the end.

Finally the route which the car should follow has to be specified. This happens in a dialog as shown in Figure 5.1. Starting out from the start lane the user has already determined in the dialog in Figure 5.1 all possible downstream lanes are presented to the user. If he selects a follow-on lane he again gets presented a window with the possible connections of the second lane. By repeating this process the user can interactively construct a route consisting of a list of lanes the car should follow during its mission. The end of a route can be defined when clicking on "NO MORE LANES" in the dialog in Figure 5.1[1].



Choose Lane:

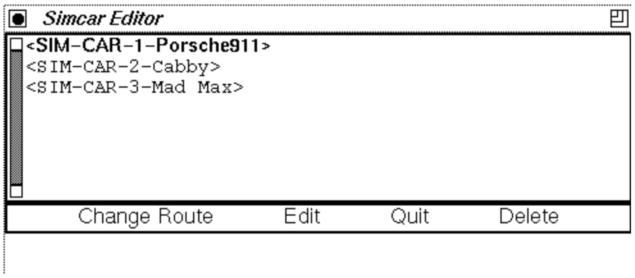
```
LANE-13          from <SECTION-21-Dagobert Highway>
LANE-14          from <SECTION-21-Dagobert Highway>
LANE-1           from <SECTION-13-Tick Trick Track Ave>
NO MORE LANE
ABORT
```

Figure 5.3 Route input.

5.3 The Sim-Car Editor

The parameters of a **`sim-car`** can be modified after its creation by using the **`sim-car`** editor (see Figure 5.4). The menu entry for this functionality is SC Editor in the main window

[1] In a more sophisticated version of ***Sapporo*** various route searching algorithms can be used for route determination. But the intention of Floating Cars is the continuous measurement of quality criteria along a main arterial, i.e. fixed route.



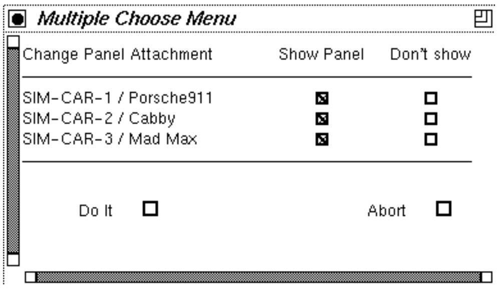
<SIM-CAR-1-Porsche911>
<SIM-CAR-2-Cabby>
<SIM-CAR-3-Mad Max>

Figure 5.4 The **`sim-car`** editor

of ***Sapporo*** . The commands are analogue to the commands in the signal plan editor. The **`sim-car`** instance presented in bold letters is the selected instance and can be edited by menu commands.

5.4 Sim-Car Panels

It is possible to display the actual traffic parameters of all **`sim-cars`** in the network during the animated simulation output or during the replay. For this purpose a **`sim-car`** panel has to be attached to each **`sim-car`** which should be monitored (Figure 5.5).

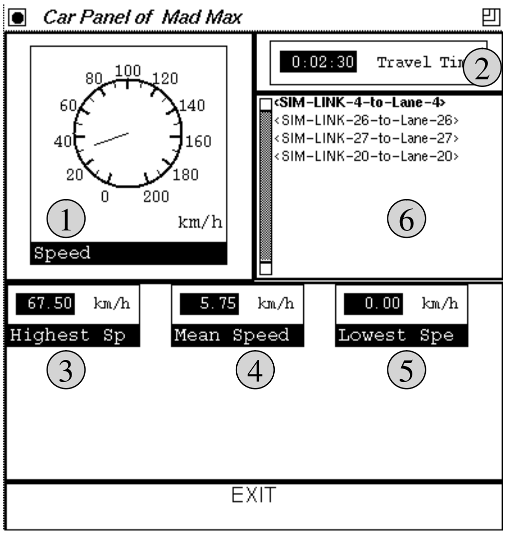


Change Panel Attachment	Show Panel	Don't show
SIM-CAR-1 / Porsche911	☒	□
SIM-CAR-2 / Cabby	☒	□
SIM-CAR-3 / Mad Max	☒	□

Figure 5.5 The **`sim-car`** attachment dialog.

All **sim-cars** with an “Show Panel” indication in the dialog in Figure 5.5 will get an own display panel after pressing the “Do-It” button.

A typical **sim-car** panel looks like the panel shown in Figure 5.5.



Car Panel of Mad Max

0:02:30 Travel Ti

<SIM-LINK-4-to-Lane-4>
<SIM-LINK-26-to-Lane-26>
<SIM-LINK-27-to-Lane-27>
<SIM-LINK-20-to-Lane-20>

67.50 km/h Highest Sp

5.75 km/h Mean Speed

0.00 km/h Lowest Spe

Figure 5.6 A **sim-car** panel

The values which can be monitored in the panel in Figure 5.5 are

the actual speed ①,

the travel time since the start of the simulation ②,

the maximum speed ③,

the mean speed ④,

the minimum speed ⑤ and

the actual link ⑥ (i.e. lane) where the **sim-car** is located on.

The selection of the “Exit” button has the same effect as the pressing of the “Don’t Show” button in Figure 5.5: the panel disappears and no monitoring output will be displayed during simulation.

5.5 Sim-Cars within Simulation

Analogue to **`sim-sensors`** a **`sim-car`** is completely integrated in the DEVS simulation architecture. To demonstrate the events which are generated and exchanged between the various objects within DEVS the messaging structure of **`sim-cars`** and **`links`** is shown in Figure 5.7.

When a **`sim-car`** enters a **`link`** he sends a "Hey-Link" message to the link to signalize the **`link`** that he's now attached and that he awaits from now on messages from all density zones which affect his actual position on the **`link`**. The **`link`** itself acknowledges this message with a "Hello" message. During the movement of the **`sim-car`** along the **`link`** the **`sim-car`** gets transmitted a message every time he enters a new density zone. Moreover, he is informed about the density value of the new density zone. When the **`sim-car`** reaches the end of the **`link`** he will send a "Bye-Bye" message to the **`link`**. Hereby the attachment between the **`sim-car`** and the **`link`** is terminated.

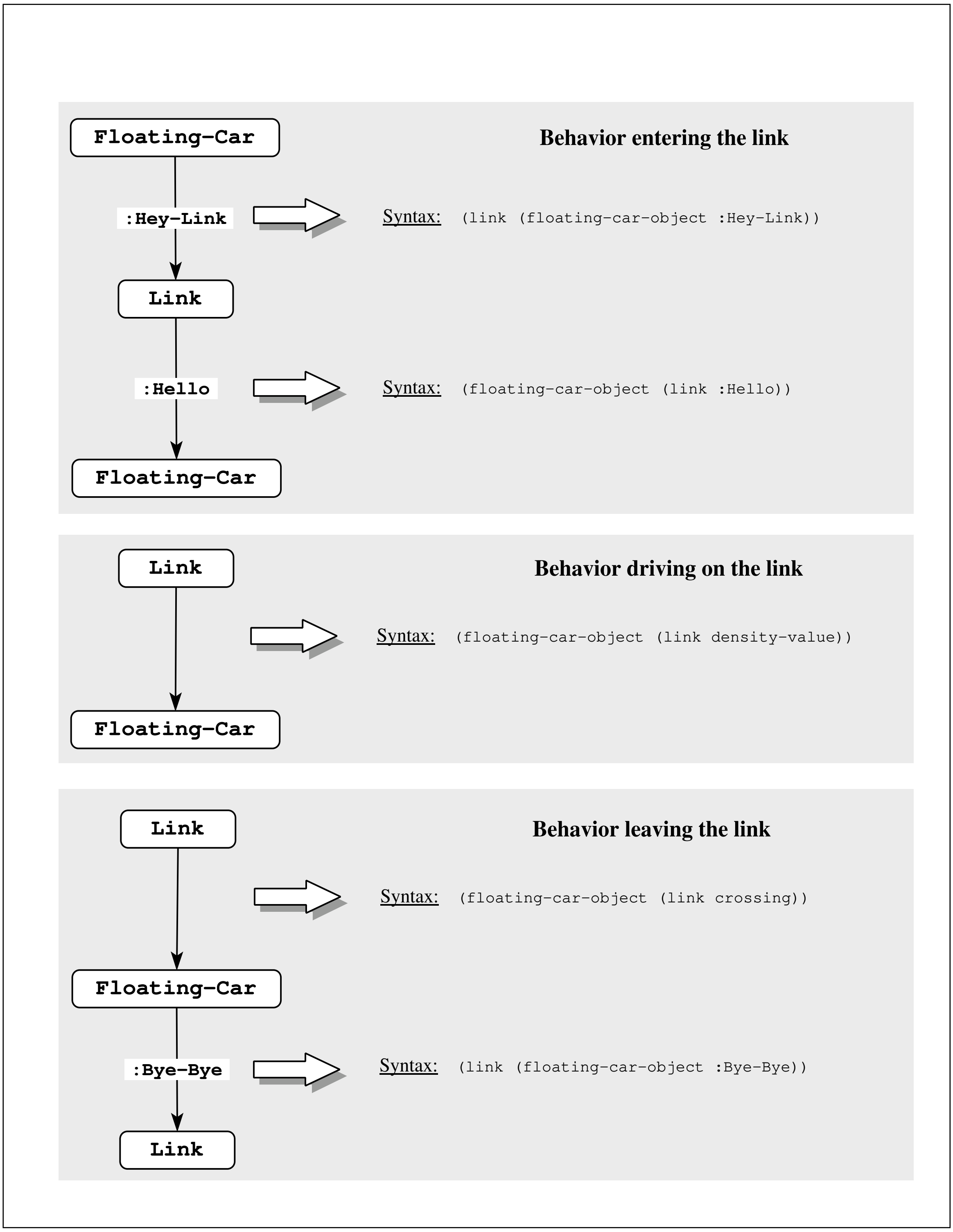

Floating-Car
Behavior entering the link
:Hey-Link
Syntax: (link (floating-car-object :Hey-Link))
Link
:Hello
Syntax: (floating-car-object (link :Hello))
Floating-Car
Link
Behavior driving on the link
Syntax: (floating-car-object (link density-value))
Floating-Car
Link
Behavior leaving the link
Syntax: (floating-car-object (link crossing))
Floating-Car
:Bye-Bye
Syntax: (link (floating-car-object :Bye-Bye))
Link

Figure 5.7 Communication protocol between a `sim-car` and a `link`.

6 Signal Plan Contract Net

The following sections underline the ideas and concepts as already described in the *Annual Report April 92*, chapter 10 and chapter 11. The actual software prototype of ***Sapporo*** doesn't include a fully operational implementation of the contract net although a laboratory version is running at FZI. With respect to the stability of the prototype and the intention of NTT Data to use the prototype as a software demo for people from public administration we decided not to integrate the lab version.

The research in distributed problem solving architectures like for example contract nets is still ongoing. Only some very dedicated prototypes – mainly in the military domain [Smith 80]– have been reported so far. One exception is the contract net for flexible manufacturing purposes by [Parunak et al. 85].

While all known realizations and contract net architectures use specialized languages for specifying the agents and the inter-agent protocol a different approach was favored by FZI. In order to construct a widely usable and non-dedicated tool for the specification of active objects, their relationships and possible organization build up by these objects, a system is under development called SIMAC. It serves as a run time environment for the testing and verification of actor based applications written in a language called ACTRICE. ACTRICE uses the actor paradigm to model active objects by their behaviors and inter-actor messaging. For this purpose a complete design methodology has been developed to support the analysis and design of actor-based applications. Two application examples have been used to demonstrate the concept of SIMAC and ACTRICE: floating cars and a contract net for traffic control.

As the floating cars module has already been described (from the functional perspective) in the following we give an overview about the contract net example in order to emphasize the need for general tools and a new methodology of handling self-organizing and self-configuring organizations of autonomous objects.

6.1 A trace of a simple contract net for control

On the following pages a trace of a prototype system is shown in order to demonstrate the way we used contract nets for traffic control. It will be obvious to the reader that even the complexity of the control of a relatively simple area will lead to an enormous administrative and debugging amount of data. The contract net protocol is a high level protocol and – in its original idea – independent of the application area. This abstraction and the independence cause naturally more overhead than if conventional and very dedicated techniques would have been used. Nevertheless, the goals *flexibility*, *explicity* and *extensibility* justify the selection of a DAI approach like contract nets.

The network to which the laboratory system CNTC[1] has been applied to is given in Figure 6.1. To keep the example comprehensible we restrict the actual control to two main intersections of the network which are shown in Figure 6.2.

[1] For economical reasons we abbreviate the contract net for traffic control by CNTC.

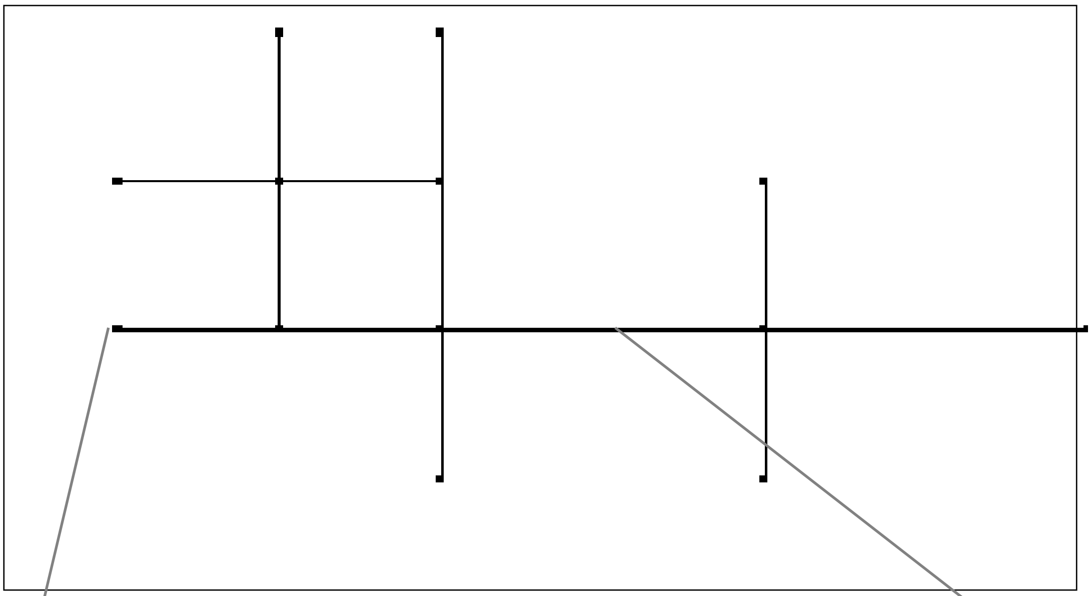


Figure 6.1 The overall traffic network (i.e. Ducktown Downtown).

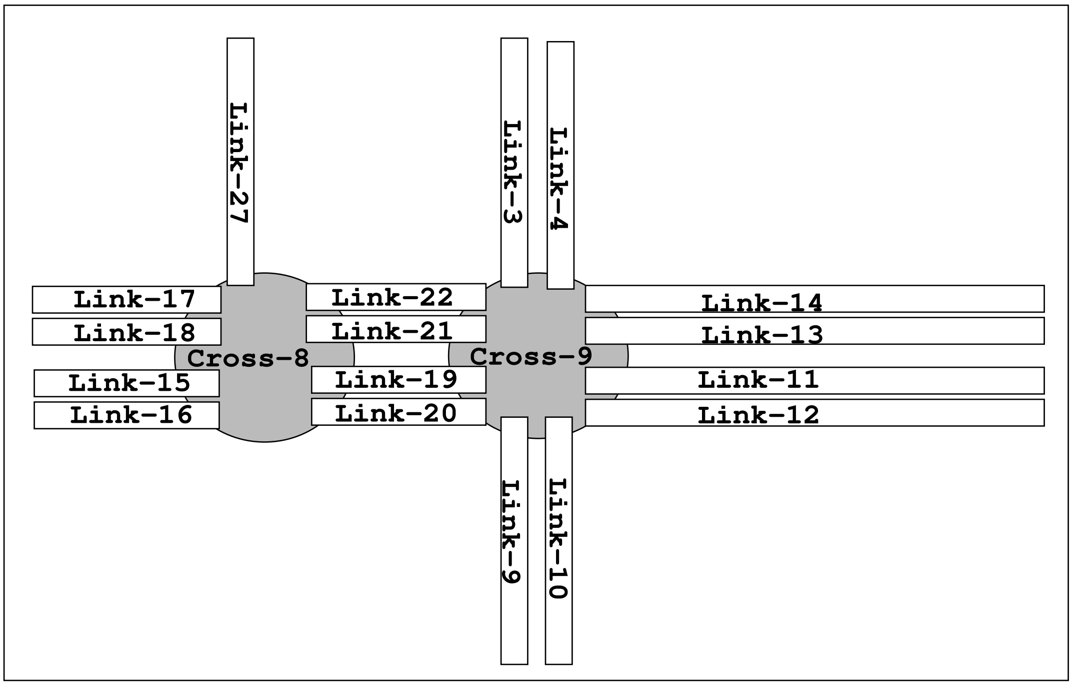


Figure 6.2 The subnet which is controlled by CNTC.

For the explanation of the actor types involved in CNTC, please, refer to *Annual Report April 92*, Chapter 11. The contract net which is specified follows the architectural guidelines of the same chapter. The code for the generation of a signal plan object is given in Figure 6.3 while the corresponding contract net can be seen in Figure 6.4. Note that the syntax for specification of a signal plan object has changed in Phase 3 of the ***Sapporo*** project.

The contract net hierarchy which will then be constructed is presented in Figure 6.5. In this version the manager actors of subordinate components are located on the same control lev-

```
(setf basic-cross-9-plan
 (make-instance
   `selection-signal-plan
   :events
   (create-events
    `( (0
    (“LINK-19” “LINK-11” )
    (“LINK-19” “LINK-12” )
    (“LINK-20” “LINK-11” )
    (“LINK-20” “LINK-12” )
    (“LINK-13” “LINK-21” )
    (“LINK-13” “LINK-22” )
    (“LINK-14” “LINK-21” )
    (“LINK-14” “LINK-22” ))
    (30
    (“LINK-10” “LINK-4”)
    (“LINK-3” “LINK-9” ))
    (50
    (“LINK-19” “LINK-9” )
    (“LINK-20” “LINK-9” )
    (“LINK-13” “LINK-4” )
    (“LINK-14” “LINK-4” )
    (“LINK-10” “LINK-21”)
    (“LINK-10” “LINK-22”)
    (“LINK-3” “LINK-11”)
    (“LINK-3” “LINK-12”))
    (75
    (“LINK-19” “LINK-4”)
    (“LINK-20” “LINK-4”)
    (“LINK-13” “LINK-9”)
    (“LINK-14” “LINK-9”)
    (“LINK-10” “LINK-11”)
    (“LINK-10” “LINK-12” )
    (“LINK-3” “LINK-21”)
    (“LINK-3” “LINK-22”))
    ))
   :cycle 100
   :start-point 75
   :end-point 0
   :conditions-generator
     #'simple-cross-conditions-generator
   :previous-plan-condition-generator
     #'simple-cross-previous-plan-condition
   ))
```

Figure 6.3 The specification of a signal plan object.

el than the worker actors of a network organizational unit. For example, an area manager is on level two together with the worker actors NETWORK-DENSITY-ESTIMATOR and NETWORK-DELAY-ESTMATOR. In the actual design using the ACTRICE methodology this has changed in order to reflect a better encapsulation of organizational units. A glance of the specification of a contract net in ACTRICE is given in Figure 6.6 while the graphical representation is illustrated in Figure 6.7.

```
(create-control-actors
 `(((user-interface)
    (blackboard)
    (manager))

  ((network "NET-1" "AREA-1")
    (blackboard)
    (manager)
    (delay-estimator)
    (density-estimator))

  ((area "AREA-1" "CROSS-9" "CROSS-8")
    (blackboard)
    (manager)
    (delay-estimator)
    (density-estimator)
    (coordinator simple)
    (coordinator fixed-cycle 60)
    (coordinator fixed-cycle 90)
    (coordinator fixed-cycle 120))

  ((cross "CROSS-9")
    (blackboard)
    (manager)
    (delay-estimator)
    (density-estimator 50)
    (signal-planer signal-selection
     ,(modify-signal-plan-cycle basic-cross-9-plan 60))
    (signal-planer signal-selection
     ,(modify-signal-plan-cycle basic-cross-9-plan 90))
    (signal-planer signal-selection
     ,(modify-signal-plan-cycle basic-cross-9-plan 120)))
  ((cross "CROSS-8")
   (blackboard)
   (manager)
   (delay-estimator)
   (density-estimator 50)
   (signal-planer signal-selection
     ,(modify-signal-plan-cycle basic-cross-8-plan 60))
   (signal-planer signal-selection
     ,(modify-signal-plan-cycle basic-cross-8-plan 90))
   (signal-planer signal-selection
     ,(modify-signal-plan-cycle basic-cross-8-plan 120)))))
```

Figure 6.4 The specification of the contract net for the 2 sections in Figure 6.2.

For the example in this section the signal planning has been reduced to a selection among three different fixed-time plans. These plans differ only by their cycle times (60, 90 and 120 sec). The task of the CNTC is now to find the signal plan for each crossing by negotiation

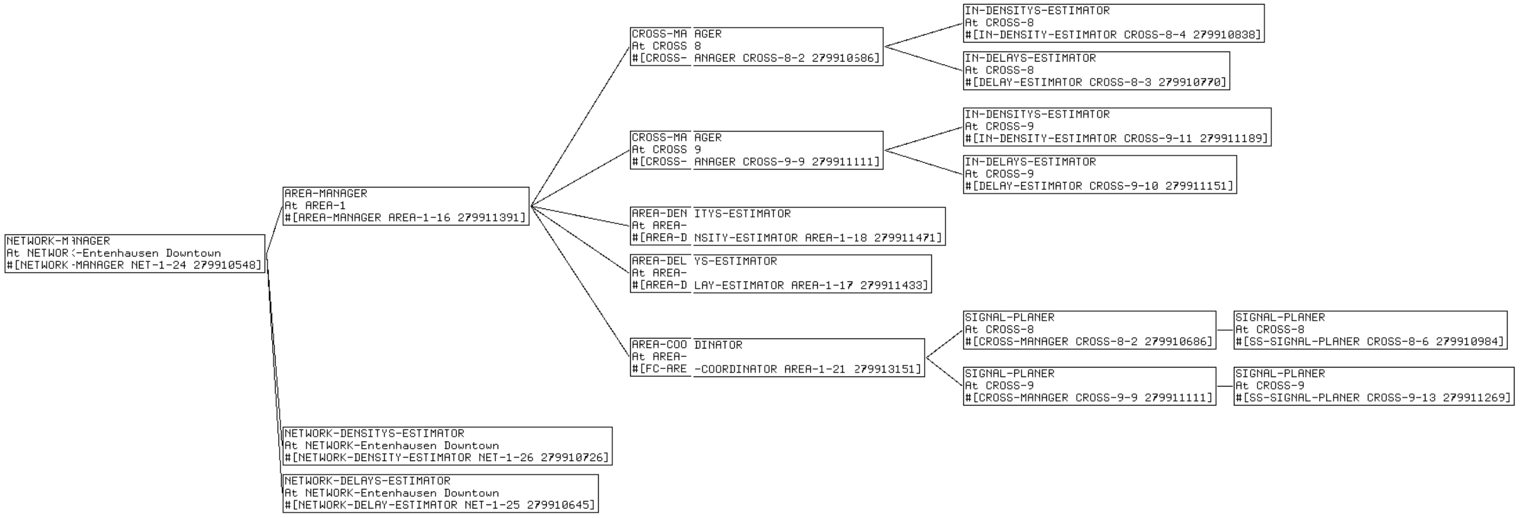

NETWORK-MANAGER
At NETWORK-Entenhausen Downtown
#[NETWORK-MANAGER NET-1-24 279910548]
AREA-MANAGER
At AREA-1
#[AREA-MANAGER AREA-1-16 279911391]
NETWORK-DENSITYS-ESTIMATOR
At NETWORK-Entenhausen Downtown
#[NETWORK-DENSITY-ESTIMATOR NET-1-26 279910726]
NETWORK-DELAYS-ESTIMATOR
At NETWORK-Entenhausen Downtown
#[NETWORK-DELAY-ESTIMATOR NET-1-25 279910645]
CROSS-MA AGER
At CROSS 8
#[CROSS- ANAGER CROSS-8-2 279910586]
CROSS-MA AGER
At CROSS 9
#[CROSS- ANAGER CROSS-9-9 279911111]
AREA-DEN ITYS-ESTIMATOR
At AREA-
#[AREA-D NSITY-ESTIMATOR AREA-1-18 279911471]
AREA-DEL YS-ESTIMATOR
At AREA-
#[AREA-D LAY-ESTIMATOR AREA-1-17 279911433]
AREA-COO DINATOR
At AREA-
#[FC-ARE -COORDINATOR AREA-1-21 279913151]
IN-DENSITYS-ESTIMATOR
At CROSS-8
#[IN-DENSITY-ESTIMATOR CROSS-8-4 279910838]
IN-DELAYS-ESTIMATOR
At CROSS-8
#[DELAY-ESTIMATOR CROSS-8-3 279910770]
IN-DENSITYS-ESTIMATOR
At CROSS-9
#[IN-DENSITY-ESTIMATOR CROSS-9-11 279911189]
IN-DELAYS-ESTIMATOR
At CROSS-9
#[DELAY-ESTIMATOR CROSS-9-10 279911151]
SIGNAL-PLANER
At CROSS-8
#[CROSS-MANAGER CROSS-8-2 279910686]
SIGNAL-PLANER
At CROSS-8
#[SS-SIGNAL-PLANER CROSS-8-6 279910984]
SIGNAL-PLANER
At CROSS-9
#[CROSS-MANAGER CROSS-9-9 279911111]
SIGNAL-PLANER
At CROSS-9
#[SS-SIGNAL-PLANER CROSS-9-13 279911269]

Figure 6.5 The contract net graph of the example in Figure 6.4.

```
(deforganization network-control
  (cn)
  (<organization variables>)
  :behaviors
  (<organization behaviors>)
  :components
  (n-delay-estimator n-density-estimator)
  :acquaintances
  (<acquaintance declarations>)
  :restrictions
  (<restriction declarations>)
  :responsibility
  (network-control network-coordination)
  :evolutions
  nil)

(deforganization area-control
  (cn)
  (<organization variables>)
  :behaviors
  (<organization behaviors>)
  :components
  (a-delay-estimator a-density-estimator a-coordinator)
  …

(deforganization intersection-control
  (cn)
  (<organization variables>)
  :behaviors
  (<organization behaviors>)
  :components
  (i-delay-estimator i-density-estimator
   signal-planer-fixed-time-1 signal-planer-fixed-time-2
   signal-planer-fixed-time-3)
  …

(deforganization CNTC
  (hierarchical-contract-net)
  (<organization variables>)
  :behaviors
  (<organization behaviors>)
  :components
  (network-control area-control intersection-control)
  :acquaintances
  (<acquaintance declarations>)
  :restrictions
  ((network-control 1)
   (area-control (1 2))
   (intersection-control :attached-to-instances-of 'crossing))
   …
```

Figure 6.6 The specification of a contract net for the subnet in Figure 6.2 using the ACTRICE method.

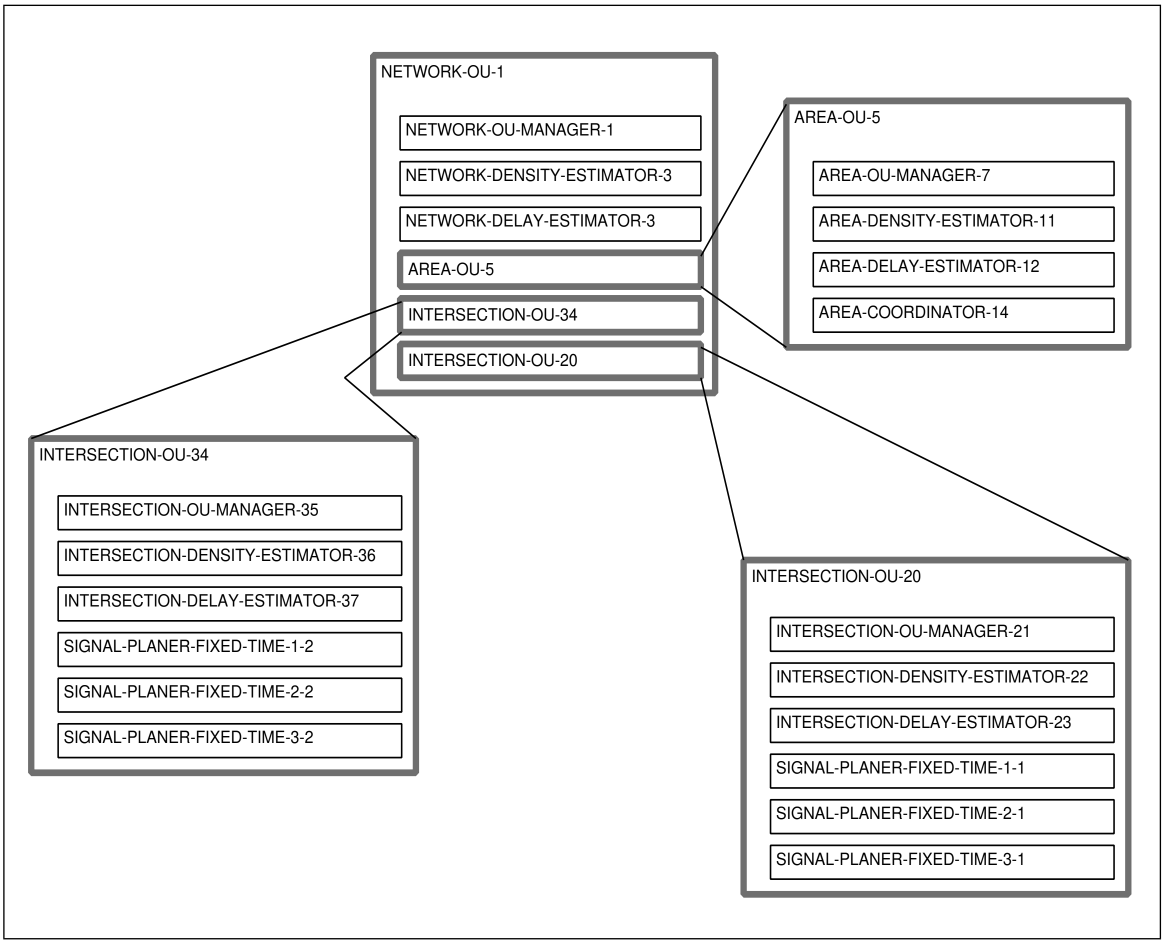
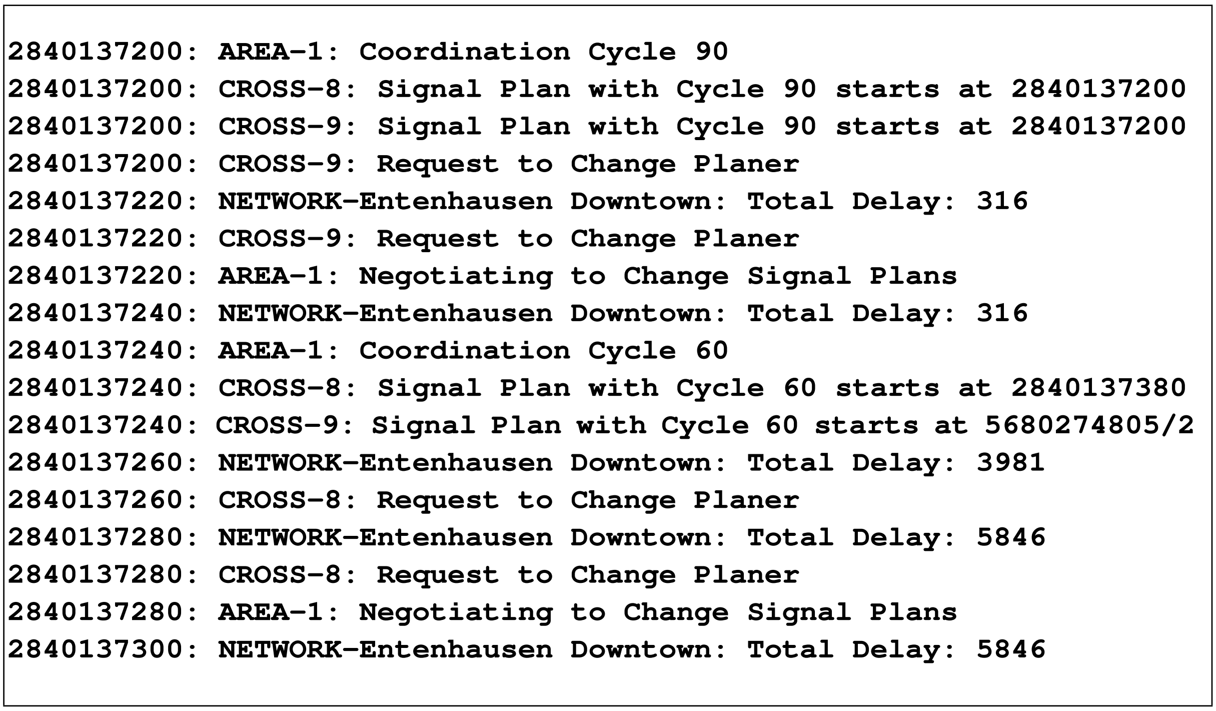

NETWORK-OU-1
NETWORK-OU-MANAGER-1
NETWORK-DENSITY-ESTIMATOR-3
NETWORK-DELAY-ESTIMATOR-3
AREA-OU-5
INTERSECTION-OU-34
INTERSECTION-OU-20
AREA-OU-5
AREA-OU-MANAGER-7
AREA-DENSITY-ESTIMATOR-11
AREA-DELAY-ESTIMATOR-12
AREA-COORDINATOR-14
INTERSECTION-OU-34
INTERSECTION-OU-MANAGER-35
INTERSECTION-DENSITY-ESTIMATOR-36
INTERSECTION-DELAY-ESTIMATOR-37
SIGNAL-PLANER-FIXED-TIME-1-2
SIGNAL-PLANER-FIXED-TIME-2-2
SIGNAL-PLANER-FIXED-TIME-3-2
INTERSECTION-OU-20
INTERSECTION-OU-MANAGER-21
INTERSECTION-DENSITY-ESTIMATOR-22
INTERSECTION-DELAY-ESTIMATOR-23
SIGNAL-PLANER-FIXED-TIME-1-1
SIGNAL-PLANER-FIXED-TIME-2-1
SIGNAL-PLANER-FIXED-TIME-3-1

Figure 6.7 The component graph of the contract net as described in Figure 6.6. An OU designates an organizational unit. The thick bordered rectangles represent non-atomar entities which consist itself of actors and/or organizational units.

between all participating actors which fits best into the expectations of traffic flow and traffic density as estimated by the simulation module of ***Sapporo*** .

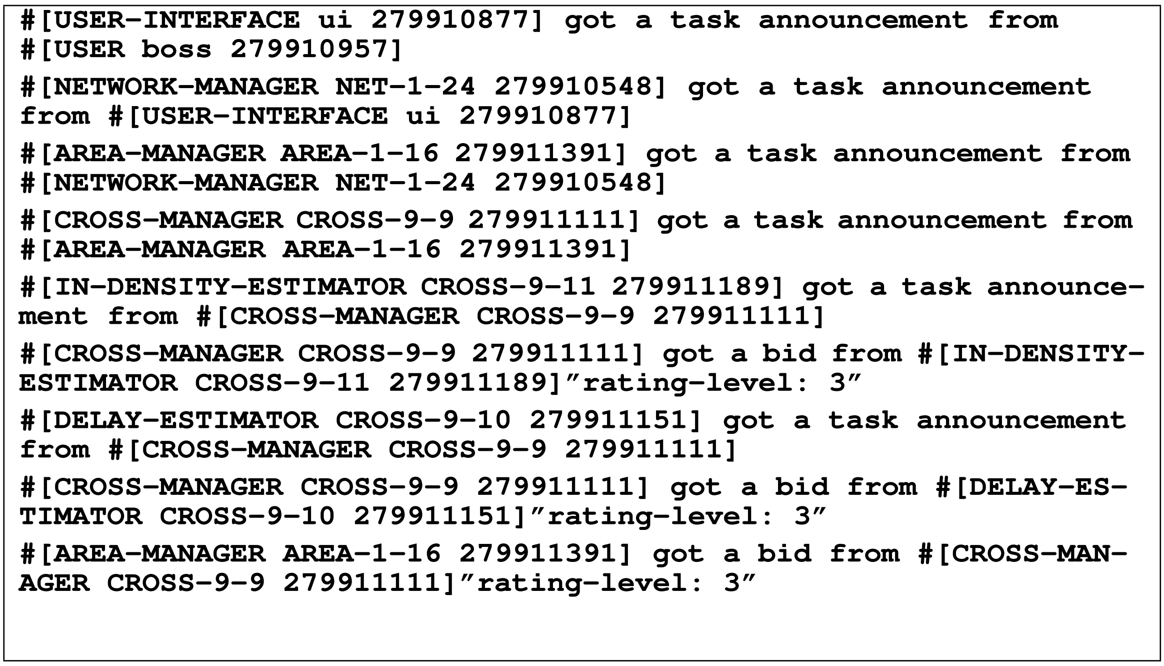
The monitoring output of the operation of CNTC shown in Figure 6.8 contains only messages relevant for traffic control. After sending a start message to the network control actor the proposal and bidding phase of CNTC starts (see). According to the rating scheme described in Chapter 11 of *Annual Report April 92* The managers try to get bids for the highest rating level first. Only after failing to receive bids the rating levels are lowered.

Figure 6.10 shows the contract phase while in Figure 6.11 the negotiations for a change of the actual signal planner and its replacement by a new one is shown. The initial signal plan had a cycle time of 90 sec. After the (lengthy) negotiations the cycle time in the area is set to 60 sec and the corresponding signal planners are contracted. The situation at CROSSING-8 isn’t satisfying which is expressed by the low rating level of 0. The reason for this non-opti-



```
2840137200: AREA-1: Coordination Cycle 90
2840137200: CROSS-8: Signal Plan with Cycle 90 starts at 2840137200
2840137200: CROSS-9: Signal Plan with Cycle 90 starts at 2840137200
2840137200: CROSS-9: Request to Change Planer
2840137220: NETWORK-Entenhausen Downtown: Total Delay: 316
2840137220: CROSS-9: Request to Change Planer
2840137220: AREA-1: Negotiating to Change Signal Plans
2840137240: NETWORK-Entenhausen Downtown: Total Delay: 316
2840137240: AREA-1: Coordination Cycle 60
2840137240: CROSS-8: Signal Plan with Cycle 60 starts at 2840137380
2840137240: CROSS-9: Signal Plan with Cycle 60 starts at 5680274805/2
2840137260: NETWORK-Entenhausen Downtown: Total Delay: 3981
2840137260: CROSS-8: Request to Change Planer
2840137280: NETWORK-Entenhausen Downtown: Total Delay: 5846
2840137280: CROSS-8: Request to Change Planer
2840137280: AREA-1: Negotiating to Change Signal Plans
2840137300: NETWORK-Entenhausen Downtown: Total Delay: 5846
```

Figure 6.8 The traffic control monitoring output of CNTC



```
#[USER-INTERFACE ui 279910877] got a task announcement from
#[USER boss 279910957]
#[NETWORK-MANAGER NET-1-24 279910548] got a task announcement
from #[USER-INTERFACE ui 279910877]
#[AREA-MANAGER AREA-1-16 279911391] got a task announcement from
#[NETWORK-MANAGER NET-1-24 279910548]
#[CROSS-MANAGER CROSS-9-9 279911111] got a task announcement from
#[AREA-MANAGER AREA-1-16 279911391]
#[IN-DENSITY-ESTIMATOR CROSS-9-11 279911189] got a task announce-
ment from #[CROSS-MANAGER CROSS-9-9 279911111]
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[IN-DENSITY-
ESTIMATOR CROSS-9-11 279911189]″rating-level: 3″
#[DELAY-ESTIMATOR CROSS-9-10 279911151] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[DELAY-ES-
TIMATOR CROSS-9-10 279911151]″rating-level: 3″
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[CROSS-MAN-
AGER CROSS-9-9 279911111]″rating-level: 3″
```

Figure 6.9 The monitoring output of the proposal and bidding phase on rating level 3.

```
#[CROSS-MANAGER CROSS-8-2 279910686] got a task announcement from
#[AREA-MANAGER AREA-1-16 279911391]
#[IN-DENSITY-ESTIMATOR CROSS-8-4 279910838] got a task announce-
ment from #[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[IN-DENSITY-
ESTIMATOR CROSS-8-4 279910838]"rating-level: 3"
#[DELAY-ESTIMATOR CROSS-8-3 279910770] got a task announcement
from #[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[DELAY-ES-
TIMATOR CROSS-8-3 279910770]"rating-level: 3"
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[CROSS-MAN-
AGER CROSS-8-2 279910686]"rating-level: 3"
#[AREA-DENSITY-ESTIMATOR AREA-1-18 279911471] got a task an-
nouncement from #[AREA-MANAGER AREA-1-16 279911391]
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[AREA-DENSI-
TY-ESTIMATOR AREA-1-18 279911471]"rating-level: 3"
#[AREA-DELAY-ESTIMATOR AREA-1-17 279911433] got a task announce-
ment from #[AREA-MANAGER AREA-1-16 279911391]
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[AREA-DELAY-
ESTIMATOR AREA-1-17 279911433]"rating-level: 3"
#[AREA-COORDINATOR AREA-1-19 279913110] got a task announcement
from #[AREA-MANAGER AREA-1-16 279911391]
#[FC-AREA-COORDINATOR AREA-1-20 279913005] got a task announce-
ment from #[AREA-MANAGER AREA-1-16 279911391]
#[FC-AREA-COORDINATOR AREA-1-21 279913151] got a task announce-
ment from #[AREA-MANAGER AREA-1-16 279911391]
#[FC-AREA-COORDINATOR AREA-1-22 279913193] got a task announce-
ment from #[AREA-MANAGER AREA-1-16 279911391]
#[CROSS-MANAGER CROSS-9-9 279911111] got a task announcement from
#[AREA-COORDINATOR AREA-1-19 279913110]
#[SS-SIGNAL-PLANER CROSS-9-12 279911228] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[SS-SIGNAL-PLANER CROSS-9-14 279911310] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]"rating-level: 3"
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[SS-SIGNAL-
PLANER CROSS-9-12 279911228]"rating-level: 3"
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[SS-SIGNAL-
PLANER CROSS-9-13 279911269]"rating-level: 3"
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[SS-SIGNAL-
PLANER CROSS-9-14 279911310]"rating-level: 3"
#[AREA-COORDINATOR AREA-1-19 279913110] got a bid from #[CROSS-
MANAGER CROSS-9-9 279911111]"rating-level: 3"
#[CROSS-MANAGER CROSS-8-2 279910686] got a task announcement from
#[AREA-COORDINATOR AREA-1-19 279913110]
```

Figure 6.9 (cont.) The monitoring output of the proposal and bidding phase on rating level 3.

```
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement
from #[CROSS-MANAGER CROSS-8-2 279910686]
#[SS-SIGNAL-PLANER CROSS-8-6 279910984] got a task announcement
from #[CROSS-MANAGER CROSS-8-2 279910686]
#[SS-SIGNAL-PLANER CROSS-8-7 279911025] got a task announcement
from #[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-9-9 279911111] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-22 279913193]
#[SS-SIGNAL-PLANER CROSS-9-14 279911310] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]"rating-level: 3"
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[SS-SIGNAL-
PLANER CROSS-9-14 279911310]"rating-level: 3"
#[FC-AREA-COORDINATOR AREA-1-22 279913193] got a bid from
#[CROSS-MANAGER CROSS-9-9 279911111]"rating-level: 3"
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[SS-SIGNAL-
PLANER CROSS-8-6 279910984]"rating-level: 3"
#[CROSS-MANAGER CROSS-9-9 279911111] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-21 279913151]
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]"rating-level: 3"
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[SS-SIGNAL-
PLANER CROSS-9-13 279911269]"rating-level: 3"
#[AREA-COORDINATOR AREA-1-19 279913110] got a bid from #[CROSS-
MANAGER CROSS-8-2 279910686]"rating-level: 3"
#[FC-AREA-COORDINATOR AREA-1-21 279913151] got a bid from
#[CROSS-MANAGER CROSS-9-9 279911111]"rating-level: 3"
#[CROSS-MANAGER CROSS-8-2 279910686] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-22 279913193]
#[SS-SIGNAL-PLANER CROSS-8-7 279911025] got a task announcement
from #[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-9-9 279911111] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-20 279913005]
#[SS-SIGNAL-PLANER CROSS-9-12 279911228] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[AREA-COOR-
DINATOR AREA-1-19 279913110]"rating-level: 3"
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[SS-SIGNAL-
PLANER CROSS-9-12 279911228]"rating-level: 3"
```

Figure 6.9 (cont.) The monitoring output of the proposal and bidding phase on rating level 3.

```
#[FC-AREA-COORDINATOR AREA-1-20 279913005] got a bid from
#[CROSS-MANAGER CROSS-9-9 279911111]"rating-level: 3"
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]"rating-level: 3"
#[FC-AREA-COORDINATOR AREA-1-22 279913193] got a bid from
#[CROSS-MANAGER CROSS-8-2 279910686]"rating-level: 3"
#[CROSS-MANAGER CROSS-8-2 279910686] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-20 279913005]
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement
from #[CROSS-MANAGER CROSS-8-2 279910686]
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[FC-AREA-CO-
ORDINATOR AREA-1-22 279913193]"rating-level: 3"
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]"rating-level: 3"
#[FC-AREA-COORDINATOR AREA-1-20 279913005] got a bid from
#[CROSS-MANAGER CROSS-8-2 279910686]"rating-level: 3"
#[CROSS-MANAGER CROSS-8-2 279910686] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-21 279913151]
#[SS-SIGNAL-PLANER CROSS-8-6 279910984] got a task announcement
from #[CROSS-MANAGER CROSS-8-2 279910686]
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[FC-AREA-CO-
ORDINATOR AREA-1-20 279913005]"rating-level: 3"
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[SS-SIGNAL-
PLANER CROSS-8-6 279910984]"rating-level: 3"
#[FC-AREA-COORDINATOR AREA-1-21 279913151] got a bid from
#[CROSS-MANAGER CROSS-8-2 279910686]"rating-level: 3"
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[FC-AREA-CO-
ORDINATOR AREA-1-21 279913151]"rating-level: 3"
#[NETWORK-MANAGER NET-1-24 279910548] got a bid from #[AREA-MAN-
AGER AREA-1-16 279911391]"rating-level: 3"
#[NETWORK-DENSITY-ESTIMATOR NET-1-26 279910726] got a task an-
nouncement from #[NETWORK-MANAGER NET-1-24 279910548]
#[NETWORK-MANAGER NET-1-24 279910548] got a bid from #[NETWORK-
DENSITY-ESTIMATOR NET-1-26 279910726]"rating-level: 3"
#[NETWORK-DELAY-ESTIMATOR NET-1-25 279910645] got a task an-
nouncement from #[NETWORK-MANAGER NET-1-24 279910548]
#[NETWORK-MANAGER NET-1-24 279910548] got a bid from #[NETWORK-
DELAY-ESTIMATOR NET-1-25 279910645]"rating-level: 3"
#[USER-INTERFACE ui 279910877] got a bid from #[NETWORK-MANAGER
NET-1-24 279910548]"rating-level: 3"
```

Figure 6.9 (cont.) The monitoring output of the proposal and bidding phase on rating level 3.

mum solution is the absence of a signal planner which could optimally handle the expected traffic situation. Therefore, the next best signal planner has been contracted. This situation is shown in Figure 6.12. The total density and delay in the network are estimated and intersection CROSS-8 requests for the change of its signal planner (see Figure 6.12).

```
#[NETWORK-MANAGER NET-1-24 279910548] got a contract from #[USER-
INTERFACE ui 279910877] starting at 2840137200
contract rating level 3
#[NETWORK-DELAY-ESTIMATOR NET-1-25 279910645] got a contract from
#[NETWORK-MANAGER NET-1-24 279910548] starting at 2840137200
contract rating level 3
#[NETWORK-DENSITY-ESTIMATOR NET-1-26 279910726] got a contract from
#[NETWORK-MANAGER NET-1-24 279910548] starting at 2840137200
contract rating level 3
#[AREA-MANAGER AREA-1-16 279911391] got a contract from #[NETWORK-
MANAGER NET-1-24 279910548] starting at 2840137200
contract rating level 3
#[AREA-DELAY-ESTIMATOR AREA-1-17 279911433] got a contract from
#[AREA-MANAGER AREA-1-16 279911391] starting at 2840137200
contract rating level 3
#[AREA-DENSITY-ESTIMATOR AREA-1-18 279911471] got a contract from
#[AREA-MANAGER AREA-1-16 279911391] starting at 2840137200
contract rating level 3
2840137200: AREA-1: Coordination Cycle 90
#[FC-AREA-COORDINATOR AREA-1-21 279913151] got a contract from
#[AREA-MANAGER AREA-1-16 279911391] starting at 2840137200
contract rating level 3
#[CROSS-MANAGER CROSS-8-2 279910686] got a contract from #[AREA-MAN-
AGER AREA-1-16 279911391] starting at 2840137200
contract rating level 3
#[DELAY-ESTIMATOR CROSS-8-3 279910770] got a contract from #[CROSS-
MANAGER CROSS-8-2 279910686] starting at 2840137200
contract rating level 3
#[IN-DENSITY-ESTIMATOR CROSS-8-4 279910838] got a contract from
#[CROSS-MANAGER CROSS-8-2 279910686] starting at 2840137200
contract rating level 3
#[CROSS-MANAGER CROSS-9-9 279911111] got a contract from #[AREA-MAN-
AGER AREA-1-16 279911391] starting at 2840137200
contract rating level 3
#[DELAY-ESTIMATOR CROSS-9-10 279911151] got a contract from #[CROSS-
MANAGER CROSS-9-9 279911111] starting at 2840137200
contract rating level 3
#[IN-DENSITY-ESTIMATOR CROSS-9-11 279911189] got a contract from
#[CROSS-MANAGER CROSS-9-9 279911111] starting at 2840137200
contract rating level 3
2840137200: CROSS-8: Signal Plan with Cycle 90 starts at 2840137200
#[SS-SIGNAL-PLANER CROSS-8-6 279910984] got a contract from #[CROSS-
MANAGER CROSS-8-2 279910686] starting at 2840137200
contract rating level 3
2840137200: CROSS-9: Signal Plan with Cycle 90 starts at 2840137200
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a contract from
#[CROSS-MANAGER CROSS-9-9 279911111] starting at 2840137200
contract rating level 3
```

Figure 6.10 The contract phase.

```
2840137200: CROSS-9: Request to Change Planer
2840137220: NETWORK-Entenhausen Downtown: Total Delay: 316
2840137220: CROSS-9: Request to Change Planer
2840137220: AREA-1: Negotiating to Change Signal Plans
#[AREA-COORDINATOR AREA-1-19 279913110] got a task announcement from
#[AREA-MANAGER AREA-1-16 279911391]
#[FC-AREA-COORDINATOR AREA-1-20 279913005] got a task announcement
from #[AREA-MANAGER AREA-1-16 279911391]
#[FC-AREA-COORDINATOR AREA-1-21 279913151] got a task announcement
from #[AREA-MANAGER AREA-1-16 279911391] for AREA-COORDINATOR
#[FC-AREA-COORDINATOR AREA-1-22 279913193] got a task announcement
from #[AREA-MANAGER AREA-1-16 279911391]
#[CROSS-MANAGER CROSS-9-9 279911111] got a task announcement from
#[AREA-COORDINATOR AREA-1-19 279913110] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-9-12 279911228] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-9-14 279911310] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]″rating-level: 3″
‶rating-level: 3″
‶rating-level: 3″
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-12 279911228]
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-13 279911269]″#[CROSS-MANAGER CROSS-9-9 279911111]
cannot find good workers for SIGNAL-PLANER″
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-14 279911310]
#[SS-SIGNAL-PLANER CROSS-9-12 279911228] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-9-14 279911310] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[SS-SIGNAL-
PLANER CROSS-9-12 279911228]″rating-level: 2″
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-13 279911269]
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-14 279911310]
#[AREA-COORDINATOR AREA-1-19 279913110] got a bid from #[CROSS-MAN-
AGER CROSS-9-9 279911111]″rating-level: 3″
#[CROSS-MANAGER CROSS-8-2 279910686] got a task announcement from
#[AREA-COORDINATOR AREA-1-19 279913110] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[SS-SIGNAL-PLANER CROSS-8-6 279910984] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686] for SIGNAL-PLANER
```

Figure 6.11 The negotiations for the change of the actual signal planner.

```
#[SS-SIGNAL-PLANER CROSS-8-7 279911025] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-9-9 279911111] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-22 279913193] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-9-14 279911310] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]"#[CROSS-MANAGER CROSS-9-9 279911111]
cannot find good workers for SIGNAL-PLANER"
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-14 279911310]
#[SS-SIGNAL-PLANER CROSS-9-14 279911310] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-6 279910984]"#[CROSS-MANAGER CROSS-9-9 279911111]
cannot find good workers for SIGNAL-PLANER"
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-14 279911310]
#[SS-SIGNAL-PLANER CROSS-9-14 279911310] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]"#[CROSS-MANAGER CROSS-8-
2 279910686] cannot find good workers for SIGNAL-PLANER"
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[SS-SIGNAL-PLANER CROSS-8-6 279910984] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-8-7 279911025] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-14 279911310]"#[FC-AREA-COORDINATOR AREA-1-22
279913193] cannot find good workers for SIGNAL-PLANER"
#[FC-AREA-COORDINATOR AREA-1-22 279913193] got a refusal from
#[CROSS-MANAGER CROSS-9-9 279911111]
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]
#[CROSS-MANAGER CROSS-9-9 279911111] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-21 279913151] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111] for SIGNAL-PLANER
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[SS-SIGNAL-
PLANER CROSS-8-6 279910984]"rating-level: 2"
"rating-level: 2"
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]"#[CROSS-MANAGER CROSS-9-9 279911111]
cannot find good workers for SIGNAL-PLANER"
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-13 279911269]
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111] for SIGNAL-PLANER
```

Figure 6.11 (cont.) The negotiations for the change of the actual signal planner.

```
#[AREA-COORDINATOR AREA-1-19 279913110] got a bid from #[CROSS-MAN-
AGER CROSS-8-2 279910686]”rating-level: 3”
“#[CROSS-MANAGER CROSS-9-9 279911111] cannot find good workers for
SIGNAL-PLANER”
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-13 279911269]
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111] for SIGNAL-PLANER
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[AREA-COORDINA-
TOR AREA-1-19 279913110]”rating-level: 3”
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-13 279911269]”#[FC-AREA-COORDINATOR AREA-1-21
279913151] cannot find good workers for SIGNAL-PLANER”
#[FC-AREA-COORDINATOR AREA-1-21 279913151] got a refusal from
#[CROSS-MANAGER CROSS-9-9 279911111]
#[CROSS-MANAGER CROSS-9-9 279911111] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-22 279913193] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-9-14 279911310] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]”rating-level: 2”
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-14 279911310]
#[SS-SIGNAL-PLANER CROSS-9-14 279911310] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]”#[CROSS-MANAGER CROSS-9-
9 279911111] cannot find good workers for SIGNAL-PLANER”
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-14 279911310]
#[SS-SIGNAL-PLANER CROSS-9-14 279911310] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]”#[CROSS-MANAGER CROSS-9-
9 279911111] cannot find good workers for SIGNAL-PLANER”
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-14 279911310]
#[SS-SIGNAL-PLANER CROSS-9-14 279911310] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[SS-SIGNAL-
PLANER CROSS-9-14 279911310]”rating-level: 0”
#[FC-AREA-COORDINATOR AREA-1-22 279913193] got a bid from #[CROSS-
MANAGER CROSS-9-9 279911111]”rating-level: 2”
#[CROSS-MANAGER CROSS-8-2 279910686] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-22 279913193] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-8-7 279911025] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-9-9 279911111] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-21 279913151] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111] for SIGNAL-PLANER”#[-
CROSS-MANAGER CROSS-8-2 279910686] cannot find good workers for SIG-
NAL-PLANER”
```

Figure 6.11 (cont.) The negotiations for the change of the actual signal planner.

```
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]
#[SS-SIGNAL-PLANER CROSS-8-7 279911025] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-13 279911269]
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111] for SIGNAL-PLANER"#[-
CROSS-MANAGER CROSS-8-2 279910686] cannot find good workers for SIG-
NAL-PLANER"
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111] for SIGNAL-PLANER"#[-
CROSS-MANAGER CROSS-8-2 279910686] cannot find good workers for SIG-
NAL-PLANER"
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]
#[SS-SIGNAL-PLANER CROSS-8-7 279911025] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]"#[CROSS-MANAGER CROSS-9-9
279911111] cannot find good workers for SIGNAL-PLANER"
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-13 279911269]
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111] for SIGNAL-PLANER
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]"#[CROSS-MANAGER CROSS-9-9 279911111]
cannot find good workers for SIGNAL-PLANER"
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-13 279911269]
#[SS-SIGNAL-PLANER CROSS-9-13 279911269] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111] for SIGNAL-PLANER"#[FC-
AREA-COORDINATOR AREA-1-22 279913193] cannot find good workers for
SIGNAL-PLANER"
#[FC-AREA-COORDINATOR AREA-1-22 279913193] got a refusal from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-8-2 279910686] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-22 279913193] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-8-7 279911025] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[SS-SIGNAL-
PLANER CROSS-9-13 279911269]"rating-level: 0"
#[FC-AREA-COORDINATOR AREA-1-21 279913151] got a bid from #[CROSS-
MANAGER CROSS-9-9 279911111]"rating-level: 2"
```

Figure 6.11 (cont.) The negotiations for the change of the actual signal planner.

```
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]
#[SS-SIGNAL-PLANER CROSS-8-7 279911025] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-9-9 279911111] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-20 279913005] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-9-12 279911228] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]″rating-level: 3″
‶#[CROSS-MANAGER CROSS-9-9 279911111] cannot find good workers for
SIGNAL-PLANER″
#[CROSS-MANAGER CROSS-9-9 279911111] got a refusal from #[SS-SIGNAL-
PLANER CROSS-9-12 279911228]
#[SS-SIGNAL-PLANER CROSS-9-12 279911228] got a task announcement
from #[CROSS-MANAGER CROSS-9-9 279911111]
2840137240: NETWORK-Entenhausen Downtown: Total Delay: 316″#[CROSS-
MANAGER CROSS-8-2 279910686] cannot find good workers for SIGNAL-
PLANER″
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]
#[SS-SIGNAL-PLANER CROSS-8-7 279911025] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-9-9 279911111] got a bid from #[SS-SIGNAL-
PLANER CROSS-9-12 279911228]″rating-level: 2″
#[FC-AREA-COORDINATOR AREA-1-20 279913005] got a bid from #[CROSS-
MANAGER CROSS-9-9 279911111]″rating-level: 3″
‶#[CROSS-MANAGER CROSS-8-2 279910686] cannot find good workers for
SIGNAL-PLANER″
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]
#[SS-SIGNAL-PLANER CROSS-8-7 279911025] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]″rating-level: 3″
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[SS-SIGNAL-
PLANER CROSS-8-7 279911025]″rating-level: 0″
#[FC-AREA-COORDINATOR AREA-1-22 279913193] got a bid from #[CROSS-
MANAGER CROSS-8-2 279910686]″rating-level: 2″
#[CROSS-MANAGER CROSS-8-2 279910686] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-20 279913005] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[FC-AREA-COOR-
DINATOR AREA-1-22 279913193]″rating-level: 3″
‶#[CROSS-MANAGER CROSS-8-2 279910686] cannot find good workers for
SIGNAL-PLANER″
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]″#[CROSS-MANAGER CROSS-8-2
279910686] cannot find good workers for SIGNAL-PLANER″
```

Figure 6.11 (cont.) The negotiations for the change of the actual signal planner.

```
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]″#[CROSS-MANAGER CROSS-8-2
279910686] cannot find good workers for SIGNAL-PLANER″
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]″#[FC-AREA-COORDINATOR AREA-1-20
279913005] cannot find good workers for SIGNAL-PLANER″
#[FC-AREA-COORDINATOR AREA-1-20 279913005] got a refusal from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-8-2 279910686] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-21 279913151] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-8-6 279910984] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686] for SIGNAL-PLANER ″rating-lev-
el: 2″
‶#[CROSS-MANAGER CROSS-8-2 279910686] cannot find good workers for
SIGNAL-PLANER″
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-6 279910984]
#[SS-SIGNAL-PLANER CROSS-8-6 279910984] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686] for SIGNAL-PLANER
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[SS-SIGNAL-
PLANER CROSS-8-6 279910984]″rating-level: 2″
#[FC-AREA-COORDINATOR AREA-1-21 279913151] got a bid from #[CROSS-
MANAGER CROSS-8-2 279910686]″rating-level: 3″
#[CROSS-MANAGER CROSS-8-2 279910686] got a task announcement from
#[FC-AREA-COORDINATOR AREA-1-20 279913005] for SIGNAL-PLANER
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[FC-AREA-COOR-
DINATOR AREA-1-21 279913151]″rating-level: 3″
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]″#[CROSS-MANAGER CROSS-8-2
279910686] cannot find good workers for SIGNAL-PLANER″
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]″#[CROSS-MANAGER CROSS-8-2
279910686] cannot find good workers for SIGNAL-PLANER″
```

Figure 6.11 (cont.) The negotiations for the change of the actual signal planner.

```
#[CROSS-MANAGER CROSS-8-2 279910686] got a refusal from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a task announcement from
#[CROSS-MANAGER CROSS-8-2 279910686]
#[CROSS-MANAGER CROSS-8-2 279910686] got a bid from #[SS-SIGNAL-
PLANER CROSS-8-5 279910916]″rating-level: 0″
#[FC-AREA-COORDINATOR AREA-1-20 279913005] got a bid from #[CROSS-
MANAGER CROSS-8-2 279910686]″rating-level: 2″
#[AREA-MANAGER AREA-1-16 279911391] got a bid from #[FC-AREA-COOR-
DINATOR AREA-1-20 279913005]″rating-level: 3″
```

Figure 6.11 (cont.) The negotiations for the change of the actual signal planner.

```
2840137240: AREA-1: Coordination Cycle 60
#[FC-AREA-COORDINATOR AREA-1-20 279913005] got a contract from
#[AREA-MANAGER AREA-1-16 279911391] starting at 0
contract rating level 3
2840137240: CROSS-8: Signal Plan with Cycle 60 starts at
2840137380
#[SS-SIGNAL-PLANER CROSS-8-5 279910916] got a contract from
#[CROSS-MANAGER CROSS-8-2 279910686] starting at 2840137380
contract rating level 0
2840137240: CROSS-9: Signal Plan with Cycle 60 starts at
5680274805/2
#[SS-SIGNAL-PLANER CROSS-9-12 279911228] got a contract from
#[CROSS-MANAGER CROSS-9-9 279911111] starting at 5680274805/2
contract rating level 2
```

Figure 6.12 The cycle time in the area is set to 60 sec.

```
2840137260: NETWORK-Entenhausen Downtown: Total Delay: 3981
2840137260: CROSS-8: Request to Change Planer″
```

Figure 6.13 The intersection controller CROSS-8 requests for a change of its actual signal planner.

After a completed run (negotiation, contracting and simulation) the actors are assessed and their ratings are corrected. This is demonstrated in Figure 6.14.

```
#[DELAY-ESTIMATOR CROSS-8-3 279910770] note changed from 5.0 to
3.8 by #[CROSS-MANAGER CROSS-8-2 279910686]”
“#[IN-DENSITY-ESTIMATOR CROSS-8-4 279910838] note changed from
5.0 to 3.8 by #[CROSS-MANAGER CROSS-8-2 279910686]”
“#[FC-AREA-COORDINATOR AREA-1-20 279913005] note changed from 5.0
to 3.8 by #[AREA-MANAGER AREA-1-16 279911391]”
“#[AREA-DELAY-ESTIMATOR AREA-1-17 279911433] note changed from
5.0 to 3.8 by #[AREA-MANAGER AREA-1-16 279911391]”
“#[AREA-DENSITY-ESTIMATOR AREA-1-18 279911471] note changed from
5.0 to 3.8 by #[AREA-MANAGER AREA-1-16 279911391]”
“#[CROSS-MANAGER CROSS-9-9 279911111] note changed from 5.0 to
3.8 by #[AREA-MANAGER AREA-1-16 279911391]”
“#[CROSS-MANAGER CROSS-8-2 279910686] note changed from 5.0 to
3.8 by #[AREA-MANAGER AREA-1-16 279911391]”
```

Figure 6.14 The assessment of the actors.

6.2 Further Research Work

The problem of traffic control by signal plans and traffic lights in cities is a major domain in traffic engineering. The control methods and techniques are numerous, mostly analytical and normally complex. They represent the knowledge traffic engineers have got in the past 40 years in optimizing traffic flow. Any attempt to construct a new traffic control regime has to take the traffic engineering methods into account if any improvement over the current situation should be achieved. A pure control architecture which doesn’t integrate the optimizing techniques the know-how of traffic engineers and traffic flow theory will never have a chance in making a real step forward. The problem we encountered when designing and experimenting with DAI in the field of traffic control was the evaluation and assessment of the results we got. As long as the traffic engineering related parts of such a system like CNTC are only fixed time signal plans and some simple rule-based criteria for applicability of such a signal plan the optimizing gain of the CNTC architecture is only of academic value. The problem *signal-based traffic control* shows up to be so complex that it would need more than one research project in order to achieve results which can be presented in the traffic engineering world and which would not be only “game systems”.

Nevertheless, the computer science can give very useful and fruitful inputs to the traffic control domain. The research target should therefore more focused on the construction of an integrative environment in which traffic engineering modules can be plugged in, tested and evaluated against each other. In this sense the contract net was the first approach to realize a platform on which different, non-compatible optimizing techniques can be used in parallel in order to compute control measures in parallel and concurrently to each other by following the motto “the better one will win”.[1]

This leads to the research strategy for the next phase which should be followed:

1) The contract net approach should be further developed in the direction of an integrative platform for traffic engineering modules. An evaluation of various (usually “non-intelligent”) optimizing techniques should be possible and be supported by the architecture. The DAI part of the control system should not be responsible for traffic optimization (i.e. signal planning) but for the management of the cooperation of the optimizing modules (i.e. agents).
2) The traffic modeling and simulation module of ***Sapporo*** should be used for the generation of traffic data and traffic situation. Special attention should be given to the fact that traffic engineering modules use different and mostly non-compatible data representations and parametrization.
3) Clear guidelines for the integration of traffic engineering methods into the platform should be worked out and demonstrated by using an example.
4) A methodology for the specification of multi-actor/multi-agent systems should be delivered in order to facilitate the integration of foreign modules. The ACTRICE and SIMAC approach can give useful input and ideas to this task while the actual research work on multi-agent system in general should be viewed under the perspective of a unique description technique.
5) A prototype system for evaluating and testing the integrative platform, the methodology and the contract net protocol.
6) The evaluation of the architecture using real world data sets and real world control decisions. This part can only be carried out in close cooperation with a traffic engineering institute or bureau and a public administration.
7) The development of a formal method for describing traffic control as distributed problem solving system with cooperating traffic engineering agents. The current approach lacks a formal representation for a application independent description of a DAI control architecture. This formalism is needed for exact specification and verification of control measures and extensions to the original contract net approach.

The FZI will continue the research on these topics as far as it can be done with the local capacities. Especially the further development of the contract net (formal approach, item 7) and the methodology for the specification of multi-actor/multi-agent systems (item 4) will play a dominant role in the next time. Expected results can directly support the realization of the

[1] This ideas have been discussed during the AIATE 92 conference where various traffic engineers asked the question what AI really can do for the traffic problems. There was an agreement on the perspective, that AI can be used either as a further module in a traffic system (e.g. intelligent user interfaces) or as an integrative framework due to its explicit and declarative character.

remaining items and should lead to a couple of publications in traffic engineering and AI journals and conferences.

7 References

[Arleth 92]
E. Arleth
Persistente Objekte in CLOS - Entwicklung und Implementierung einer Schnittstelle in Common LISP zur Erzeugung, Bearbeitung und Speicherung von Objekten innerhalb des Common LISP Object Systems CLOS
Studienarbeit, FZI, 1992

[Brachman 83]
R. J. Brachman
What IS-A Is and Isn‘t: An Analysis of Taxonomic Links in Semantic Networks
IEEE Computer, October, 1983

[Hunt et al. 81]
P. B. Hunt, D. I. Robertson, R. D. Bretherton, R. I. Winston
SCOOT A Traffic Responsive Method of Coordinating Signals
TRRL LR 1014, Transport and Road Research Laboratory, Crowthorne, UK, 1981

[Parunak et al. 85]
H. V. D. Parunak, B. W. Irish, J. Kidrick, P. W. Lozo
Fractal Actors for Distributed Manufacturing Control
Proc. CAIA, 1985

[Quillian 68]
M. R. Quillian
Semantic Memory
in "Semantic Information Processing", M. Minsky (ed.), MIT Press, 1968"

[Rumbaugh et al. 91]
J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, W. Lorensen
Object-Oriented Modeling and Design
Prentice-Hall, Englewood Cliffs, NJ, 1991

[Shastri 88]
L. Shastri
Semantic Networks: An Evidential Formalization and its Connectionist Realization
Pitman, London, 1988

[Smith 80]
R. G. Smith
The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver
IEEE Transactions on Computers, C-29:12, 1980

A Appendix

A.1 Mouse Gesture Commands in *Sapporo*

Table A.2 shows all mouse gestures available in the actual prototype of ***Sapporo*** . The abbreviations of the gestures are:

Table A.1 Gesture abbreviations.

Gesture	Meaning
c	control
m	meta
s	shift
R	Right mouse button
M	Middle mouse button
L	Left mouse button
for example: c-m-s-R	control-meta-shift-Right click

Table A.2 Mouse gesture commands in ***Sapporo*** .

	Command							
	Delete	*Edit*	*Get Info*	*Show Simulated Profile at*			*Source Profile*	*Event List*
Object type				*Start*	*Position*	*End*		
Node	c-M (only exception points)	c-m-M (signal plans)	L	-	-	-	m-L	-
		M (Intersection Designer)						
Section		c-m-L (Insert Sensor)	L	m-L	m-M	m-R	-	s-L
		c-m-s-L (Insert Floating Car)						
		c-m-M (Insert Exception Point)						
		c-m-R (DC Attachm.)						
Links			L	m-L	m-M	m-R	-	s-L
Sensors	c-M		L	-	m-M	-	-	-
Sim-Sensors	c-M		L	-	m-M	-	-	-