



<b>Programme:</b>	Information Society Programme
<b>Key Action:</b>	1
<b>Action Line:</b>	1-5-1
<b>Project Number:</b>	IST-1999-11244
<b>Project Acronym:</b>	HEAVEN

<b>Deliverable Number:</b>	D 5.2
<b>Deliverable Title:</b>	Overall System Architecture and Implementation Action Plan
<b>Dissemination Level*:</b>	LI
<b>Nature**:</b>	RE
<b>Type***:</b>	PD
<b>Date of Preparation:</b>	9 March 2001
<b>Project's Internal Reference:</b>	
<b>Author(s):</b>	Inger Hoffmann, Fanny Mietlicki , Haibo Chen, Jiri Ctyroky, Tjoncho Wang, Emilio Cera, Andrea Teschioni, Fabrizio Biora, Carlo Di Taranto
<b>Editor(s):</b>	Carlo Di Taranto, Fabio Nussio, Hermann Heich

<b>Project Co-ordinator:</b>	<b>Maurizio Tomassini</b> <b>Società Trasporti Automobilistici</b> Via Ostiense 131/L 00154 Roma Italia Phone: +39-06-57118216 Fax: +39-06-57118547 E-mail: m.tomassini@sta.roma.it
------------------------------	---

\* PU-public usage, LI-limited to programme participants, RP-restricted to project participants

\*\* PR-prototype/demonstrator, RE-report, SP-specification, TO-tool, OT-other

\*\*\* PD-project deliverable, X-submitted on request deliverable

## Document Control Sheet

<b>Project:</b>	IST-1999-11244 HEAVEN
<b>Document name: Deliverable</b>	D 5.2 - Overall System Architecture and Implementation Action Plan
<b>Document reference:</b>	
<b>Other internal reference:</b>	
<b>Prepared by (organisation):</b>	Mizar Automazione SpA
<b>Author(s):</b>	Inger Hoffmann, Fanny Mietlicki , Haibo Chen, Jiri Ctyroky, Tjoncho Wang, Emilio Cera, Andrea Teschioni, Fabrizio Biora, Carlo Di Taranto
<b>Editor(s):</b>	Carlo Di Taranto, Fabio Nussio, Hermann Heich
<b>Reviewed by:</b>	

## Issue History

Issue	Description	Originator	Date of issue
Draft A	Interim draft for comments	Carlo Di Taranto	25/11/2000
Draft B	Draft for comments	Carlo Di Taranto	31/01/2001
Issue 01	Initial Issue	Carlo Di Taranto	09/03/2001

\*<sup>1</sup> Draft A, Draft B, ... , Issue 01, Issue 02, ...

\*<sup>2</sup> Draft for Comment, Final Draft for Review, Interim Issue, Initial Issue or Revised Issue

\*<sup>3</sup> In general the author. If more than one author, the (principal) editor is the originator.

## EXECUTIVE SUMMARY

The HEAVEN project has the twofold objective to develop and to demonstrate in six European sites a Decision Support System (DSS) able to evaluate the environmental effects – air and noise quality – of near real-time traffic flows and Traffic Demand Management Strategies (TDMS) in large urban areas.

The HEAVEN DSS will be demonstrated in the cities of Rome, Paris, Berlin, Rotterdam, Leicester and Prague. In each city the DSS general system concept is going to be implemented basing on already existing infrastructure, tools and models and taking into account local peculiarities concerning functional, physical and organisational requirements as well as context constraints.

In this framework, based on the detailed User Need Analysis performed in the context of WP4 “User Requirements and Implementation Framework”, WP5 produces the:

- Design of the Overall System Architecture corresponding to the HEAVEN DSS general concept.
- Design of the System Architecture of the Decision Support Systems to be implemented and demonstrated in each project site.
- Definition of the technical specifications for the local DSS components.
- Elaboration of infrastructure adaptation and local DSS implementation plans.

The outcomes of the work package directly feed the activities of WP6 “Build Integrated Systems”.

The Overall System Architecture designed for the HEAVEN DSS and the functional and information architectures designed for the local decision support systems were the subject of the project report D5.1 – Environment Monitoring and DSS Architecture.

This report (D5.2 – Overall System Architecture and Implementation Action Plan) presents the technical specifications of the SW and HW components of the local DSS’, and the activity plans that will lead to demonstrator implementation in the sites.

The report consists of six chapters. After the Chapter 1, devoted to the introduction, Chapter 2 presents the technical process that is leading the project from the definition of a common Overall System Architecture to the implementation of the different DSS demonstrators in the sites. Chapter 3 then defines the common aspects of the approach adopted by the sites to specify the components of the local demonstrators.

The technical specifications of the SW and HW components of the local demonstrators and the detailed list of activities planned to achieve the implementation of these systems are collected in Chapter 4. Chapter 5 describes the activity plans that will lead to implementation.

Finally, Chapter 6 includes some consideration emerging from the experience gained through the design of the HEAVEN DSS and of the local demonstrators.

The Annex A and Annex B provide complementary information on the elaboration processes and their consistency with the functional architecture building blocks described in D5.1.

## CONTENTS

1	Introduction	3
2	From a Common SA to site level implementation	4
2.1	The HEAVEN DSS concept	4
2.2	The technical approach to System Design	7
3	Summary of common aspects in providing system specifications	9
4	Implementation in the Sites	10
4.1	Physical Aspects of the Berlin DSS	11
4.1.1	SW Components	12
4.1.2	HW Components	15
4.1.3	Connections	15
4.1.4	Implementation constraints	16
4.1.5	Actions to implement	17
4.2	Physical Aspects of the Leicester DSS	19
4.2.1	SW Components	19
4.2.2	HW Components	22
4.2.3	Connections	23
4.2.4	Implementation constraints	23
4.2.5	Actions to implement	24
4.3	Physical Aspects of the Paris DSS	25
4.3.1	SW Components	25
4.3.2	HW Components	28
4.3.3	Connections	29
4.3.4	Implementation constraints	30
4.3.5	Actions to implement	30
4.4	Physical Aspects of the Prague DSS	33
4.4.1	SW Components	33
4.4.2	HW Components	36
4.4.3	Connections	36
4.4.4	Implementation constraints	37
4.4.5	Actions to implement	38
4.5	Physical Aspects of the Rome DSS	40
4.5.1	SW Components	40
4.5.2	HW Components	43
4.5.3	Connections	44
4.5.4	Implementation constraints	44
4.5.5	Actions to implement	45
4.6	Physical Aspects of the Rotterdam DSS	47
4.6.1	SW Components	48
4.6.2	HW Components	50
4.6.3	Connections	51
4.6.4	Implementation constraints	51
4.6.5	Actions to implement	52

5	Overall project planning	55
5.1	Site level GANTT charts	55
5.1.1	Berlin GANTT Chart	<b>Fehler!</b>
5.1.2	Leicester GANTT Chart	56
5.1.3	Paris GANTT Chart	57
5.1.4	Prague GANTT Chart	58
5.1.5	Rome GANTT Chart	59
5.1.6	Rotterdam GANTT Chart	60
6	Lessons learned	61
7	References	62

Annex A – Detailed process description tables for the Demonstration Site DSS’

Annex B – Detailed consistency tables mapping Functions and Data Flows of the DSS architectures on to Processes and Data Flows of the physical systems

# 1 INTRODUCTION

The HEAVEN project has the twofold objective to develop and to demonstrate in six European sites a Decision Support System (DSS) able to evaluate the environmental effects – air and noise quality – of near real-time traffic flows and Traffic Demand Management Strategies (TDMS) in large urban areas.

The HEAVEN DSS general concept has been identified to be consistent with the EU Converge Guidelines and to provide a framework to achieve the integration of legacy systems in the demonstration sites of the HEAVEN project.

The HEAVEN DSS will be demonstrated in the cities of Rome, Paris, Berlin, Rotterdam, Leicester and Prague. In each city the DSS general system concept is going to be implemented basing on already existing infrastructure, tools and models and taking into account local peculiarities concerning functional, physical and organisational requirements as well as context constraints.

In this framework, based on the detailed User Need Analysis performed in the context of WP4 “User Requirements and Implementation Framework”, WP5 produces the:

- Design of the Overall System Architecture corresponding to the HEAVEN DSS general concept.
- Design of the System Architecture of the Decision Support Systems to be implemented and demonstrated in each project site.
- Definition of the technical specifications for the local DSS components.
- Elaboration of infrastructure adaptation and local DSS implementation plans.

The outcomes of the work package directly feed the activities of WP6 “Build Integrated Systems”.

The Overall System Architecture designed for the HEAVEN DSS and the functional and information architectures designed for the local decision support systems were the subject of the project report D5.1 – Environment Monitoring and DSS Architecture.

This report (D5.2 – Overall System Architecture and Implementation Action Plan) presents the technical specifications of the SW and HW components of the local DSS’, and the activity plans that will lead to demonstrator implementation in the sites.

In particular:

- Chapter 2 describes the technical process that is leading the project from the definition of a common Overall System Architecture to the implementation of the different DSS demonstrators in the sites.
- Chapter 3 identifies the common aspects of the approach adopted by the sites to implement the local demonstrators.
- Chapter 4 provides the technical specifications of the SW and HW components of the local demonstrators and the detailed list of activities planned to achieve the implementation of these systems.
- Chapter 5 describes the activity plans elaborated by the sites.
- Chapter 6 includes some consideration emerging from the experience gained through the design of the HEAVEN DSS and of the local demonstrators.

The Annex A collects site by site the detailed description of the elaboration processes designed to implement the functionalities of the DSS demonstrators.

The Annex B collects the consistency tables that map the DSS functions and data flows described in D5.1 onto the elaboration processes and data flows identified in this report D5.2.

## 2 FROM A COMMON SA TO SITE LEVEL IMPLEMENTATION

This chapter describes the technical process that is leading the project from the definition of a common Overall System Architecture to the implementation of the different DSS demonstrators planned in the sites.

### 2.1 The HEAVEN DSS concept

The concept of the HEAVEN DSS was defined since the very beginning of the Project and then tuned and described in details through the first project deliverables [1] and [3]. The main aspects of such a concept are recalled in the following.

The HEAVEN DSS has been conceived as a tool suitable to:

- support the decision making processes concerned with the definition of temporary measures as well as medium-term and long-term TDMS, by facilitating the evaluation of their impacts on air and noise quality
- build a common information platform on traffic and environment for local authorities, including existing and potential users
- compile suitable information reports on traffic and environment to be delivered to end users through several media.

The HEAVEN DSS combines near real-time traffic flow information into emission models so as to determine the contribution of mobile sources to air quality and noise. In order to estimate emissions based on current traffic levels and on planned demand management scenarios, the system can operate on-line, based on current traffic and environmental information, and off-line, based on planned traffic and environmental conditions and pre-defined TDMS.

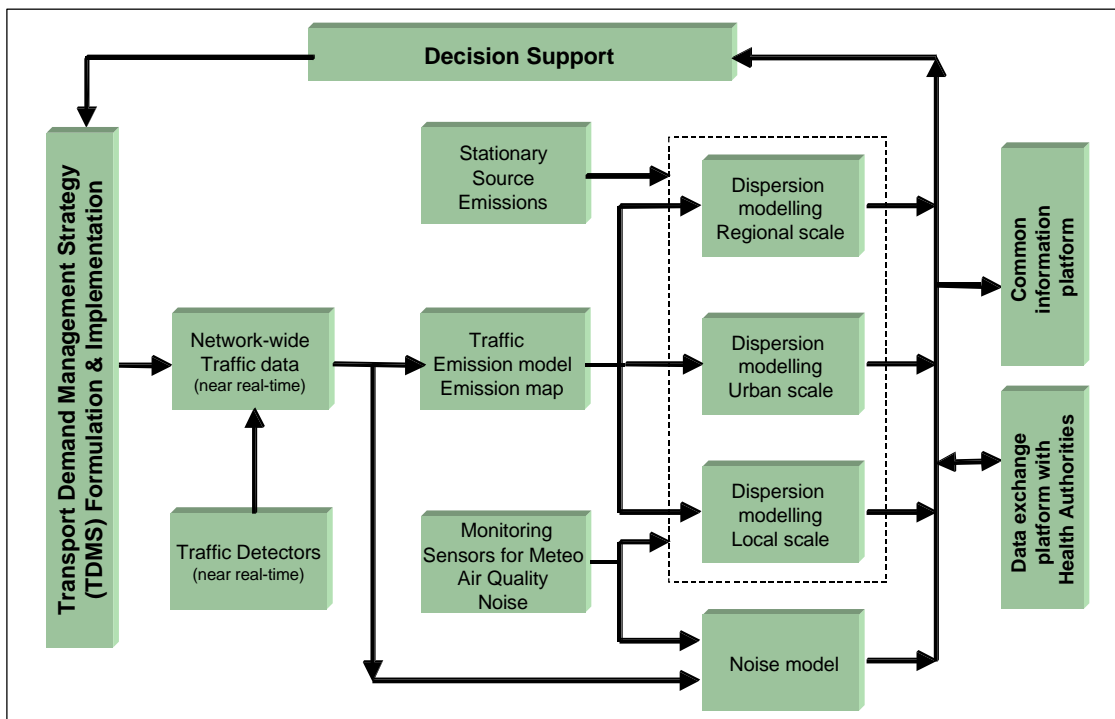


Figure 2.1 – The HEAVEN DSS general concept

The diagram above describes the dynamic chain that links the HEAVEN DSS building blocks concerned with Traffic and Environment modelling, and provides a representation of the general concept of the HEAVEN DSS. This dynamic chain can be commented through the following points:

- The TDMS implemented in the urban area is expected to influence the traffic status in the network. Traffic is partially measured in near real-time via traffic detectors. These measurements contribute to the near real-time traffic status estimation performed through a network-wide traffic model. The dynamic traffic representation forms the input for the subsequent environmental models.
- Emission models calculate the traffic related emissions for each link of the network for a number of primary pollutants based on measured and modelled traffic characteristics. The network-related traffic emissions are then fed into dispersion models to calculate air quality concentrations. This process considers emissions from stationary sources from emission inventories and data on background concentrations from monitoring networks when available. Depending on the type and the spatial scale of the dispersion model, other data (i.e., topographic, meteorological) will also be used in the modelling process.
- The noise model makes use of much of the same traffic data input, together with additional information, including road surface data, topographical data, and data about the physical built environment (i.e., surface attenuation by buildings, shielding, reflection, etc). The noise model also includes information concerning static sources.
- The infrastructures and the systems for traffic measurement and environmental monitoring are not included in the DSS concept. So, the DSS building blocks “Traffic Detectors” and “Monitoring Sensors for Meteo, Air Quality, Noise” that are shown in the diagram are data suppliers. These blocks have to be processed and interpreted as the DSS interfaces between the system and the infrastructures and monitoring systems.
- The results of the emissions, air quality and noise modelling and forecast are fed into the decision making process and together with the traffic and environment monitoring data contribute to the building of the Common Information Platform of the HEAVEN system. Contents of this platform are intended also for dissemination to professional users, as well as to the public. A specific channel is foreseen for data exchange with the Health Authorities.

The characteristics of the HEAVEN DSS were derived from system requirements – context, functional and non-functional requirements – that reflect the user needs identified by the User Needs Analysis [3] and include operational and organisational requirements.

The User Needs Analysis provided extensive and articulated requirements for each DSS demonstrator and highlighted different backgrounds (in terms of existing models and infrastructures) between the project sites.

So, aiming at the implementation of demonstrators consistent with the common HEAVEN DSS concept, although duly customised according to the needs of the sites, and addressing system modularity and transferability issues, the project as a whole took the decision to approach system design following a top-down process:

- To design at first a common Overall System Architecture consistent with the HEAVEN DSS (see the diagram in Figure 2.2)
- To derive subsequently the local DSS architectures from the common overall architecture taking care of the peculiarities and backgrounds of the sites (see Figure 2.3).

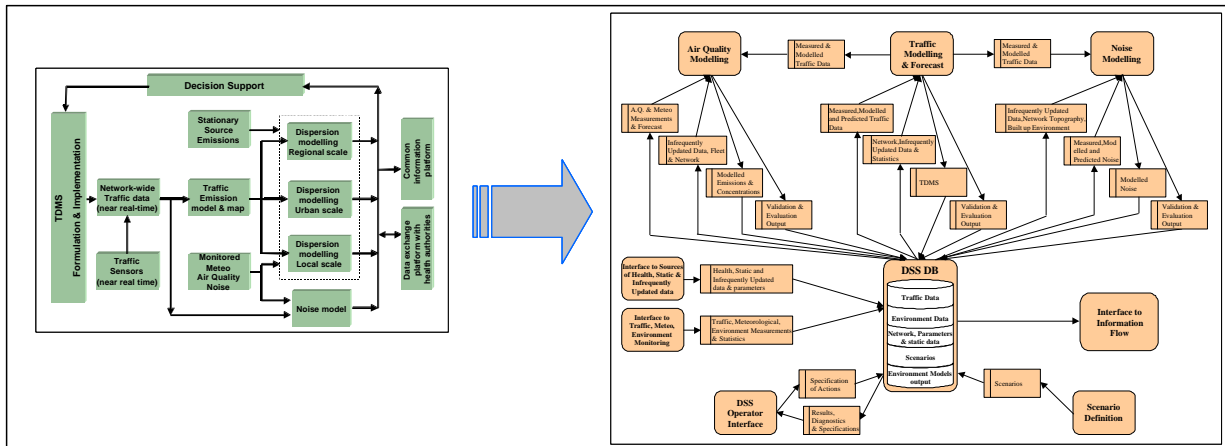


Figure 2.2 – From the HEAVEN DSS general concept to the Overall System Architecture

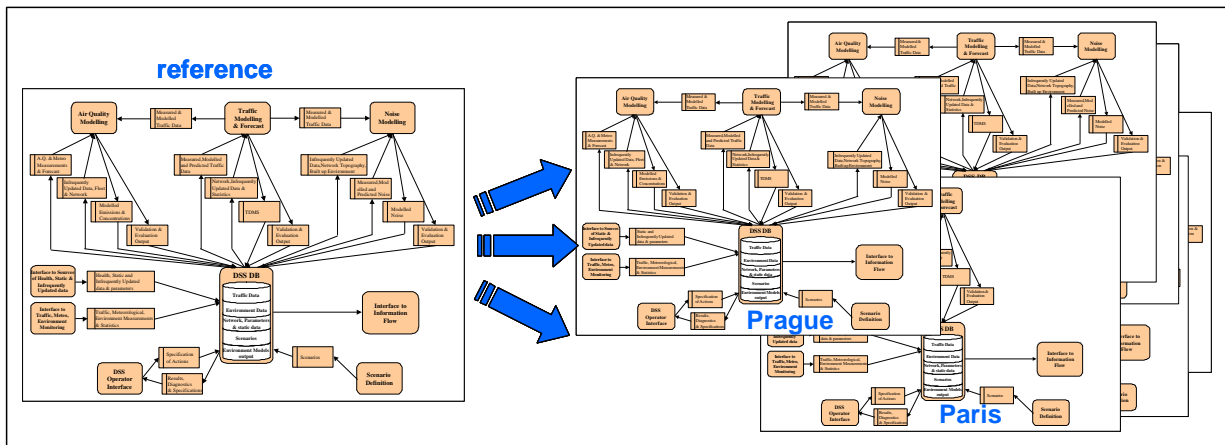
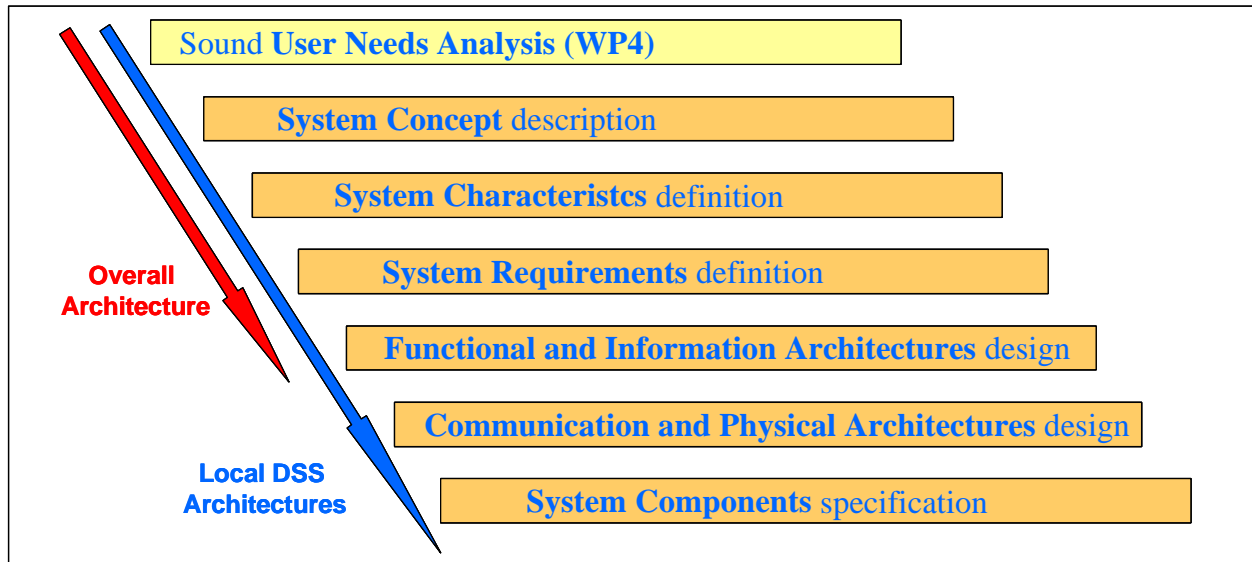


Figure 2.3 – The Overall System Architecture: a reference for the local DSS architectures design

## 2.2 The technical approach to System Design

The scheme in Figure 2.4 describes the steps performed to design the system architecture of both the HEAVEN DSS general concept and the local DSS demonstrators.



**Figure 2.4 – The seven step process adopted to design the DSS system architectures**

This approach has been developed according to the CONVERGE guidelines presented in [2].

The sound analysis of the user needs [3] performed by WP4 “User Requirements and Implementation Framework”, covered the first step of the process and constituted the basis for the System Architectures design.

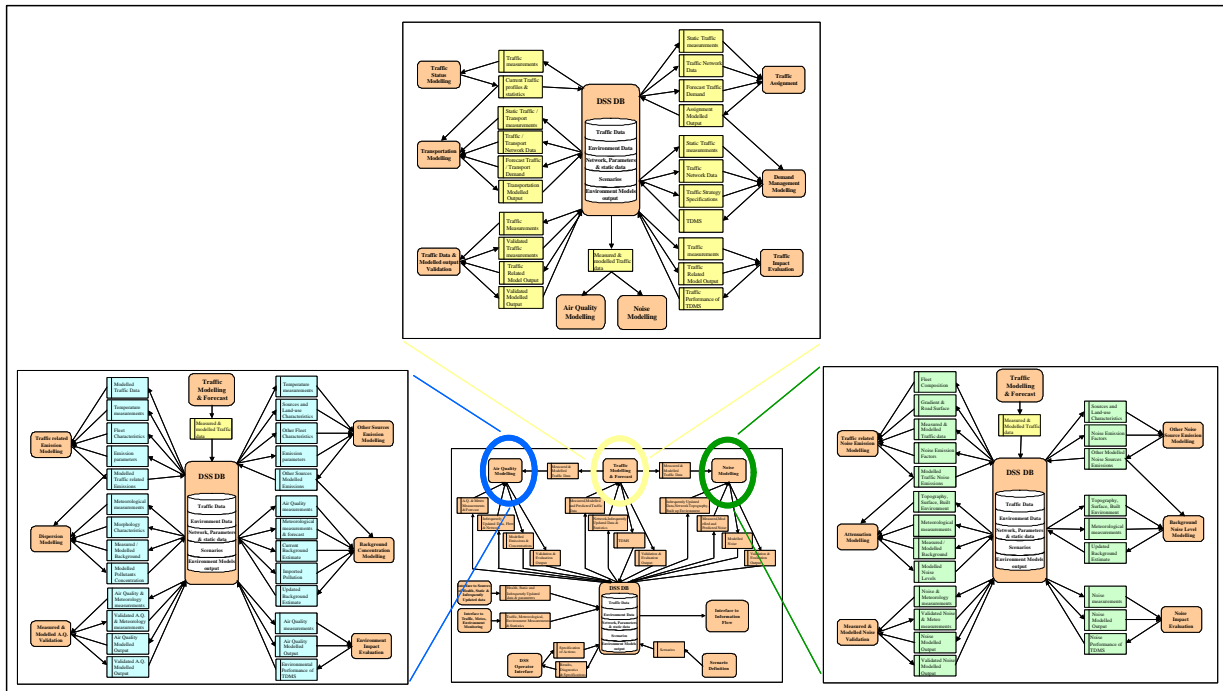
The execution of the remaining steps have been assigned to the experts in both the traffic and environmental modelling and the IST systems involved in WP5.

In particular the design of the Overall System Architecture was conducted through the first five steps only, the last two steps being strictly connected to the site implementation realities and so to the DSS demonstrator architectures.

The outcomes of the first five steps have been reported in [1] for both the HEAVEN DSS Overall Architecture and the local DSS demonstrator architectures. The outcomes of the last two steps are included in this report.

The design of the Functional and Information architectures conducted to the identification of three main modelling modules - namely the Air Quality Modelling, the Traffic Modelling and Forecast, the Noise Modelling - and of some other fundamental building blocks: the Operator Interface, the interface to the external systems and data sources, the Scenarios Management and the interface to the Common Information Platform.

The design of the Overall System Architecture achieved a deep level for the three modelling modules only (see the following Figure 2.5), whereas the design of the local DSS’ was conducted to deeper levels for all the building blocks.



**Figure 2.5 – The functional architecture of the Modelling Modules of the Overall SA**

The results of the Functional and Information Architectures design constitute the starting point of the detailed specification of the local DSS components.

These specifications are provided in terms useful for the further activities planned in WP6:

- i. Elaboration processes that should be performed by the SW modules to implement the functions identified inside the functional architectures
- ii. Characteristics and performance of the HW components
- iii. Communication and Physical system architectures.

The fundamental set of specifications is collected in the following Chapter 4. Chapter 3 presents a summary of the common information items to be covered by each site when proving the specifications.

Finally, in order to guarantee that elaboration processes identified and specified in this stage cover the functions and sub-functions designed in the context of the functional architectures, detailed tables are provided in the Annex B that map the functions and data flows described in [1] onto for the elaboration processes and data flows identified in this report.

### 3 SUMMARY OF COMMON ASPECTS IN PROVIDING SYSTEM SPECIFICATIONS

In order to obtain an homogeneous description of the local DSS demonstrator specifications, the project sites have been requested to follow a common methodology and to provide the following set of information items:

- i. Description of the elaboration processes that implement the functions of the DSS demonstrator
- ii. Requirements for the HW components that will host the elaboration processes
- iii. Connectivity aspects
- iv. System implementation constraints
- v. Activities planned to bring the site from system specification to system implementation
- vi. Activity time plan.

In particular:

- The description of the elaboration processes is expected based on Data Flow Diagrams, identifying the “processes”, “data stores”, “interfaces” and “data flows” involved in the system.
- The specification of the HW and connectivity characteristics are expected complemented by specific architectural diagrams.
- The system implementation constraints should cover technical items such as:
  - Operating systems
  - Network protocols
  - Database engine
  - Topological reference
- The activities planned to implement should address:
  - SW development
  - HW provision
  - Connection activation

and should be complemented by specific GANTT Charts.

## 4 IMPLEMENTATION IN THE SITES

In this chapter the DSS implementation in the project sites is addressed through the definition of:

- the DSS physical aspects, that include physical and communication architectures
- the implementation constraints
- the actions planned for the DSS implementation.

The implementation aspects are described site by site following a common methodology:

- at first the functions performed by the DSS are recalled and translated in terms of “processes”;
- secondly the software modules designed to perform such processes are identified and the requirements for the hardware components are established;
- the connectivity aspects are then defined;
- the subsequent step concerns the possible implementation constraints;
- finally the actions required to implement the system are fixed and for each action a responsible is identified at the site level.

The implementation action plans are presented in the subsequent chapter.

The DSS’ physical architecture description is supported by two types of graphical representations:

- i. the Data Flow Diagram (DFD)
- ii. the hardware components and interconnection schemes

In particular, the DFD diagram identifies the “processes”, “data stores”, “interfaces” and “data flows” involved in the system. These entities are defined as:

- **process** represents an activity that transforms data from one form to another
- **interface** represents external sources or consumer of data
- **data store** represents a repository for information that is output of one or more processes
- **data flow** represents a packet of data that is sent to or from a process

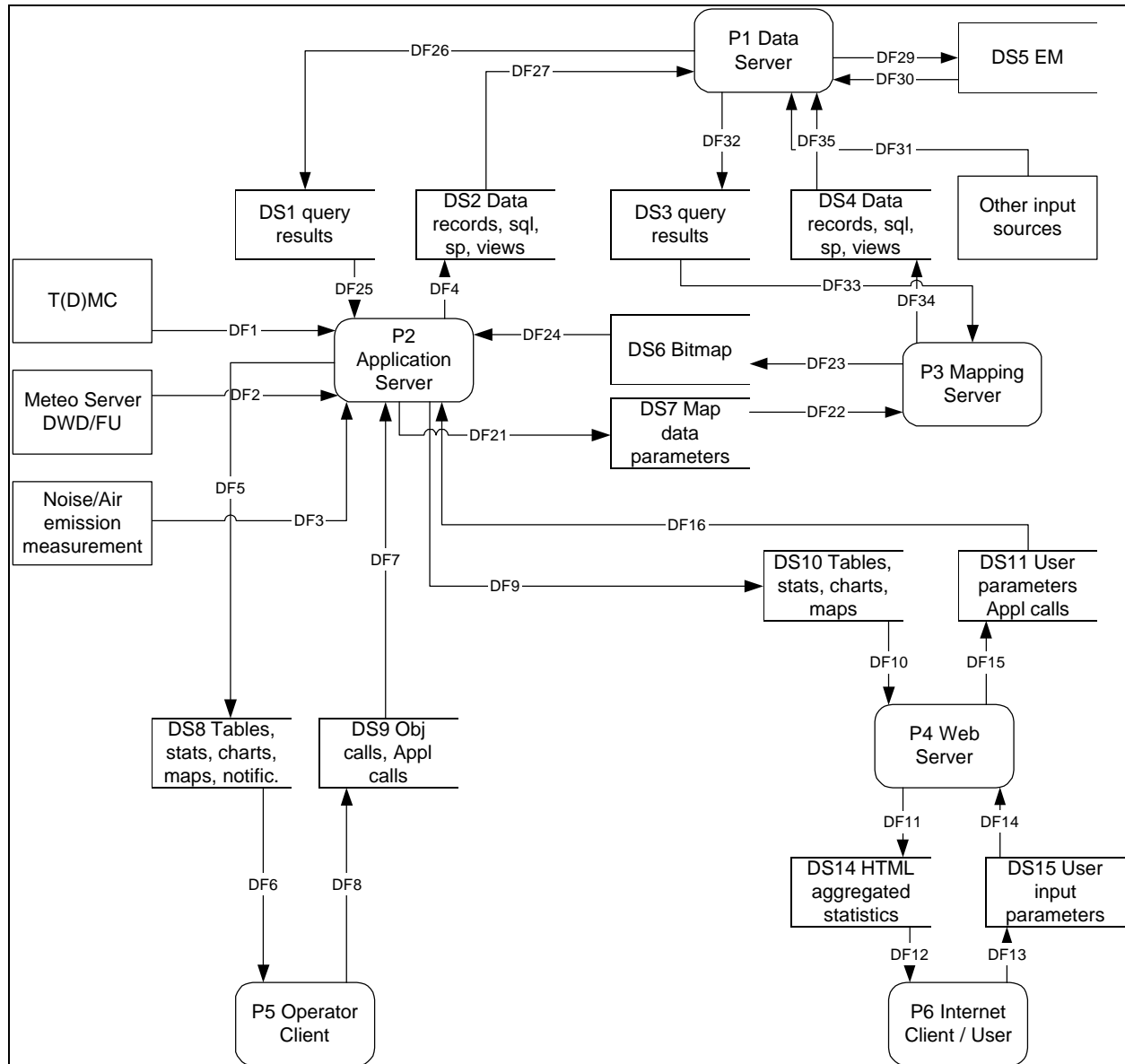
The process definition is hierarchical as processes can be split in sub-processes for several levels. In this chapter the top level diagrams are presented only. The lower level diagrams and the related information are presented in the Appendix A.

Finally, each process can be mapped on one or more software modules. Thus the software components are identified by the process or sub-process served.

Data flows specify the information exchanged and the timing of the information. Data stores are stored on a relational database in order to ease the access of the information by processes and tools for data analysis.

### 4.1 Physical Aspects of the Berlin DSS

The following figure 4.1.1 shows the top level DFD diagram of the Berlin DSS demonstrator and identifies the main processes, data stores and data flows.

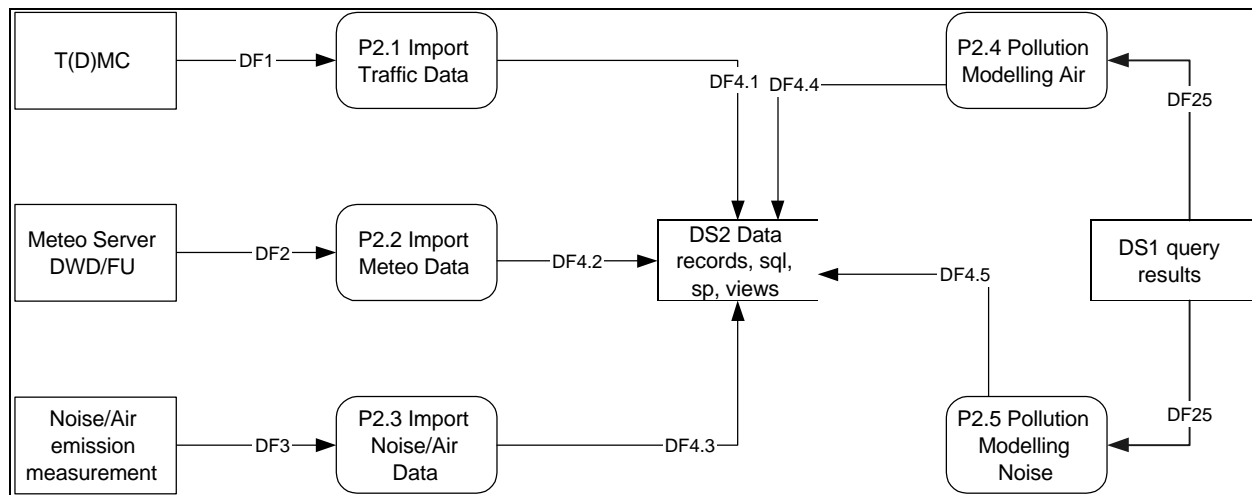


**Figure 4.1.1 – The top level Data Flow Diagram of the Berlin DSS**

The application server (P2) represents the core of the system. Under an application server different functions and processes with a uniform interface are comprised. An application server is designed to help make it easier for developers to isolate the business logic in their projects (usually through components) and develop three-tier applications. It is a program run on a mid-sized machine that handles all application operations between browser-based computers and a company's back-end business applications

or databases. Because many databases cannot interpret commands written in HTML, the application server works as a translator, allowing, for example, a customer with a browser to search an online retailer's database for pricing information.

Via the application server dynamic data is imported and transferred to the data server and the database. In addition the application server also handles the import of the input for the models and transfers the modelled results to the data server and also. The application server processes are rather complex and are therefore presented in the additional Figure 4.1.2.



**Figure 4.1.2 – Data input and models handled by the application server**

### 4.1.1 SW Components

The top level DFD identifies the main processes that implement the functions described in the functional architecture. The following table 4.1.1 summarises the main processes and sub-processes identified at the detailed specification stage.

Annex A reports detailed tables including specification of requirements, data flows and functions (as described in D5.1) to be implemented.

The detailed mapping of the functions and data flows as described in D5.1 to the processes and data flows specified in this report are collected in the Annex B.

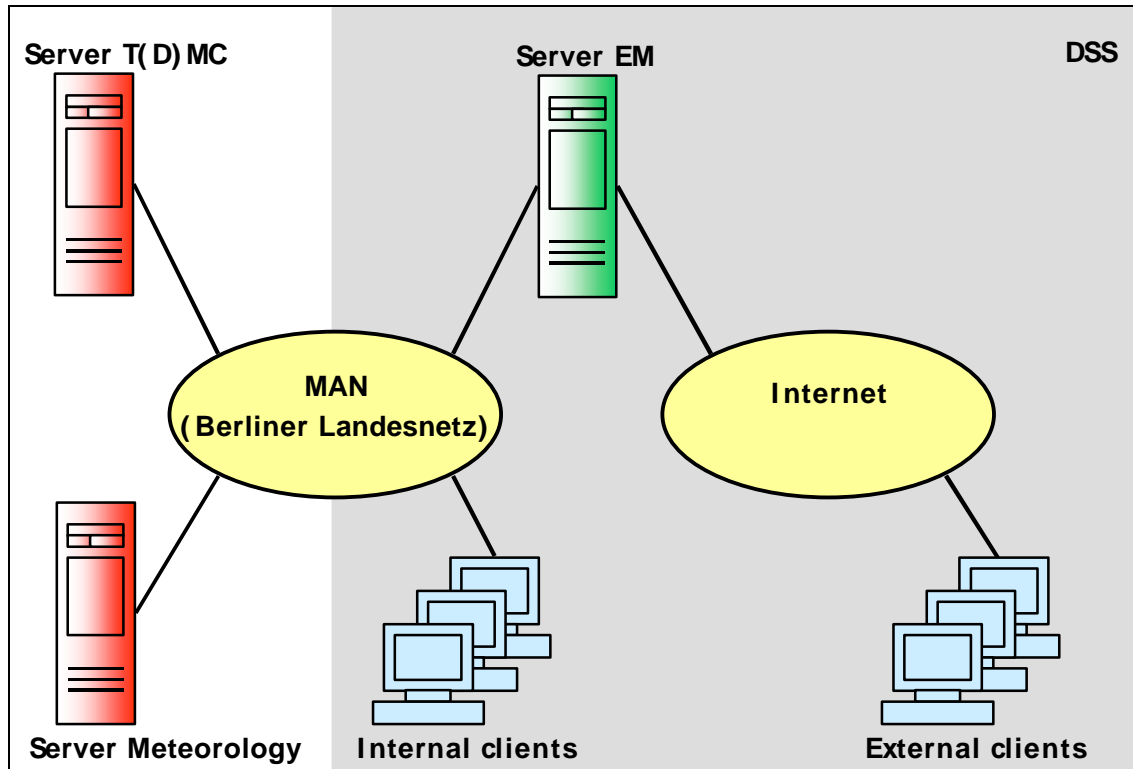
Process	Description
P1 Data Server	The data server is the program which manages the access to the data in the EM database. In this database all input data for the environmental modelling is stored. But also the modelling results are stored in the EM database for future use (generating historic statistics, generating maps, etc.).
P2 Application Server (manages dynamic data input, the environmental models and user / operator information and requests)	<p>The application server is the core of the system as it provides the business logic for an application program.</p> <p>Different processes are managed in the framework of the application server. All necessary input data for the system, such as traffic data, meteo data, noise/air measurement data, background pollution data is handled by the application server. In this sense the server can be regarded as an open interface to other data sources as described in processes p2.1 - P2.3. The data retrieved is then stored in the EM database for further use (modelling of emission/dispersion, scenario analysis, generation of maps and tables).</p> <p>But also the key environmental modelling tools are part of the application server. Data input for these models is provided from the EM database via the data server (processes P2.4 and P2.5)</p> <p>In addition the application server also provides information to the web server from where the information is provided to the clients. Requests from the users and the operator are also handled via the application server from where the requests are further processed.</p>
P2.1 Import Traffic Data	<p>Traffic data is provided by the Traffic (Demand) Management Centre Berlin (T(D)MC) which is not part of the HEAVEN system.</p> <p>This application program can be regarded as the interface of the HEAVEN system to the TDMC. Near real time traffic data as well as 24 hour forecast data are provided by the T(D)MC to the HEAVEN DSS. The traffic data is then imported from the application server by the EM database.</p>
P2.2 Import Meteo Data	As traffic data, near real-time and forecast meteorology data is provided by external sources and imported to the EM database via this application program.
P2.3 Import Noise/Air Measurement Data	For verification of the modelling results and also for the calculation of background pollution Noise / Air measurement data are imported to an application program. This data is also imported by the EM database.
P31 Import static data	Population and population development data, emission related parameters and statistics, air quality models parameters, topography, land-use, road network data as well as information from a background emission database (Emissionskataster Berlin) is updated in suited intervals.
P2.4 Pollution Modelling Air	Provides the modelling of emissions and pollution concentrations for the controlled network. The model considers the observed traffic from the T(D)MC, background pollution as monitored and modelled with a specific tool inside the model, the current meteorological conditions as retrieved from external sources and static data (build environment, etc.). The results of the environmental modelling is validated with the measured data. The final results are transferred to the EM database

Process	Description
P2.5 Pollution Modelling Noise	Provides the modelling of noise concentrations for the controlled network. The modelling considers the observed traffic, tram schedule and other static data (build environment, road conditions, etc.). The results of the environmental modelling is validated with the measured data. The final results are transferred to the EM database.
P3 Mapping Server	Data on pollution concentration is provided in pure data / figures from the modelling tools. The mapping server transfers the modelled environmental data from the EM database into a GIS system which allows for a graphical presentation of near real time pollution levels and forecasts to the users.
P4 Web Server	A computer that delivers Web pages. Every Web server has an IP address and possibly a domain name. If an URL is entered in a browser, this sends a request to the server. The server then fetches the respective page and sends it to the browser. Via the Web Server pollution data and emission maps are made available to Web users. The Web server also handles requests from the users (for statistics, forecasts, etc.) as it transfers the requests to the application server where the requests are handled.
P5 Operator Client	<p>The operator of the HEAVEN system gets access to the system via a Windows-based graphical user interface. Via this client, the operator can retrieve statistics from the system (historical databases) and can request forecasts when pollution concentration is likely to raise above previously defined limits.</p> <p>The operator can also start a scenario analysis, i.e. modelling the effects of TDMS and other transport policies on the environmental situation. The operator can also change other input parameters such as traffic conditions and meteorological conditions.</p>
P6 Internet Client	Web browser-based graphical user interface, usually at a PC or workstation. Access is regulated by access rights. While some information such as pollution levels will be available to the general public other information, and functionalities will only be available to selected users at the administration

**Table 4.1.1 – Processes and sub-processes of the Berlin DSS**

### 4.1.2 HW Components

Figure 4.1.3 shows the top-level physical architecture of the Berlin DSS. This diagram identifies the actuators implied by the system context diagram:



**Figure 4.1.3 – The hardware components and connection links of the Berlin DSS**

Three subsystems are shown, the T(D)MC, the meteorological information server and the EM DSS. The data transmission between the Traffic (Demand) Management Centre and Environmental Monitoring System will be done via the Berliner Landesnetz. There will be two different kinds of clients / users. The general public (external clients) will get a strictly defined set of information while the users at the administration (internal clients) will get additional rights and access to the DSS via the internet.

### 4.1.3 Connections

The Communication Architecture depicted by Figure 4.1.4 describes the channel characteristics identified within the Physical Architecture.

The data transmission between the Traffic Management Centre and Environmental Monitoring System will be done via a TCP/IP. The application protocol will be done via FTP.

The Mapping Server visualises and analyses data on interactive maps.

The Application Server provides a common interface. The Client Applications are completely independent, to enable a connection in the future to additional clients, such as WAP Handy users.

Users are connected via TCP/IP with the web server. They use the HTTP-protocol for communication.

RPC provides the basis of communication and inter-operability between the DSS and Operator Clients and the Application Server.

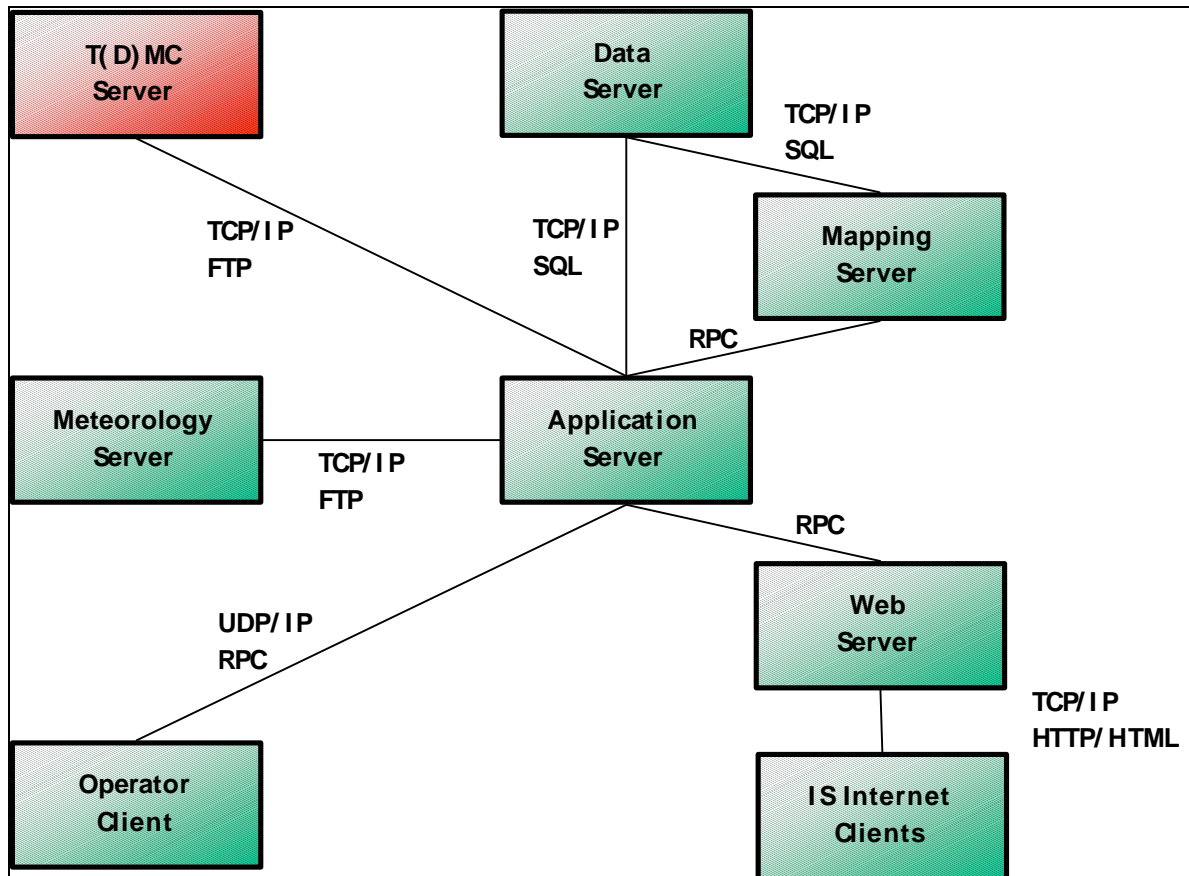


Figure 4.1.4 – The Communication architecture of the Berlin DSS

#### 4.1.4 Implementation constraints

The following constraints have been identified and will be considered in the specification phase of the implementation.

##### 4.1.4.1 Operating system

All the Server-Software will be developed for a Microsoft Windows 2000 Server environment.

##### 4.1.4.2 Network protocols

The network protocols adopted are:

- TCP/IP
- HTTP
- FTP
- RPC

#### **4.1.4.3 Inter-process communication**

Inter-process communication is based on the RPC protocol.

#### **4.1.4.4 Process synchronisation**

The synchronisation between dependant processes is based on notification of availability of data for a specific time interval.

#### **4.1.4.5 Database engine**

As Database Management System for the shared data stores ORACLE 8 database server will be used.

### **4.1.5 Actions to implement**

The actions identified to implement the DSS concern three different aspects:

- SW development
- HW provision
- Connections activation.

Such activities are described in the following paragraphs.

#### **4.1.5.1 SW development**

The implementation for the DSS demonstrator has been split into several phases as described in the following. Each phase has been given a global deadline and thus activities related to the phase have been planned to cope with the overall phase deadline.

##### **Specification phase**

The specification phase has the objective to draw a detailed specification of each process, data store and data flow and hardware identified by the overall architecture. The result of this phase is a full set of software specification for the development of software modules and the integration of the overall system.

Deadline: End of March 2001

##### **Implementation phase**

The implementation phase will take care of the integration of different processes up to build the overall system running in real time.

Deadline: October 2001

##### **Testing phase**

Each component will be tested for its functional correctness in relation to its specification.

Deadline: December 2001

##### **Verification phase**

Following up the implementation and testing phase the overall system needs to be verified. The verification phase could highlight needs for re-engineering or modifications of components (e.g. according to aspects of efficiency).

Deadline: March 2002

## ACTION SUMMARY

The following actions are identified:

Action	Responsible partner
<b>Specification activities</b>	
Processes specification	IVU
Data stores specification	IVU
Data flows specification	IVU
Operator Client specification	IVU/Senat
Internet Client specification	IVU/Senat
<b>Development activities</b>	
P1 Data Server	IVU
P2 Application Server	IVU
P3 Mapping Server	IVU
P4 Web Server	IVU
P5 Operator Client	IVU
P6 Internet Client	IVU

### 4.1.5.2 HW provision

The following actions are identified:

Action	Responsible partner
Hardware specification	IVU
Hardware provision	Senat
System and LAN configuration	IVU/Senat

### 4.1.5.3 Connection activation

The following actions are identified:

Action	Responsible partner
Provision of an Internet connection for the EM Server.	Senat
Provision of a connection between the EM Server and the T(D)MC Server.	Senat
Provision of a connection between the EM Server and the Meteorology Server.	Senat

## 4.2 Physical Aspects of the Leicester DSS

### 4.2.1 SW Components

Table 4.2.1 summarises the main processes and sub-processes of the Leicester DSS identified at the detailed specification stage. Annex A gives more detailed tables for requirements specification, data flows and mapping to the functions to be implemented.

Process	Description
P1 Traffic data collection	The Leicester UTC (Urban Traffic Control) system is responsible for acquiring traffic characteristics from the City via SCOOT, REMAC (remote monitoring ), CCTV and VMS. This process involves data pre-processing, distribution and storage via the Leicester DSS operator interface.
P2 Traffic modelling	This process undertakes a modelling of traffic across Leicestershire, Leicester and more comprehensively in the SCOOT controlled areas. The process consists of several sub-processes as described below.
P2.1 At the MAGNOSCOPIC level	This process (i.e. TRIPS) provides both current and historical data for the modelling network. It processes, stores, and reports information on roadways, traffic and accidents for the network. The process produces traffic outputs for on-line air pollution, noise and demand management strategy design and assessment.
P2.2 At the MACROSCOPIC level: Assignment Modelling	See sub-processes
P2.2.1 Estimating network-based O/D matrices	This process provides the modelling system a library of updated O/D matrix estimations by day types over a certain time period of the day. The process uses maximum entropy for matrix estimation from traffic counts available for links in the network. It takes an old or best estimate of a prior trip matrix and uses the current traffic counts to estimate the most likely trip matrix consistent with the information contained in the traffic counts. If there if no prior trip matrix is available, the model assumes that all trips are equally likely so that, in effect, it starts with a prior trip matrix in which all elements are equal.
P2.2.2 Network Simulation Generating signal timings	This process implements several programs for network analysis. It produces results for <ol style="list-style-type: none"> <li>(1) the analysis of current situation and traffic management schemes over relatively localised networks (typically of the order of up to 100 or 200 nodes though models have been developed with over 1000 junctions);</li> <li>(2) the analysis of much larger networks (e.g., up to 6000 links in the standard PC version); and</li> <li>(3) the simulation of individual junctions.</li> </ol>
P2.3 At the MACROSCOPIC level: Signal Optimisation	This process carries out an off-line macroscopic, deterministic simulation and optimisation modelling to determine how best to co-ordinate fixed-time traffic signal timings by using average traffic flows in a network of roads. It measures the network performance (or cost of traffic congestion) by a weighted combination of the delay and stops with due concern for the queue lengths on all the links in the network.

Process	Description
P2.4 At the MESOSCOPIC level: Congestion analysis	The process classifies traffic congestion in urban areas into different levels such as quiet, busy and congested at 5-minute intervals. The process employs COMIS (COngestion Management Information Systems) which formulates a graph theoretic approach to enable the spatial and temporal patterns of traffic building-up and decaying across a network to be mapped and categorised into families or "congestion events".
P2.5 At the MESOSCOPIC level: Incident detection	This process can automatically detect traffic incident in the SCOOT modelled areas of the City. The system uses information from the SCOOT traffic control system via the ASTRID database of traffic conditions and produces the information about time, duration, location, area affected and severity of an incident, and confidence that the incident is real.
P3 Environmental modelling	Traffic data is passed from the monitoring systems by way of the statistical processing and/or modelling elements of the traffic systems and are input to the air quality or noise model. Data on meteorological conditions and weather forecasts are fed into the wide area models OPANA (MACROSCOPIC scale) and NAME (MAGNOSCOPIC scale), which predict background levels to feed into the Airviro system (MESOSCOPIC scale). Meteorological conditions monitoring are used by Airviro to predict pollutant concentrations and air quality forecasts. The noise model AVTUNE requires traffic data, meteorological conditions monitoring and network characteristics.
P3.1 MAGNOSCOPIC air quality modelling	This process NAME runs an atmospheric pollution dispersion model to produce nation wide air quality forecasting. The model runs simulations on scales from a few kilometres to thousands of kilometres, for example on a 150 km square grid over the UK, enabling it to handle problems of imported pollution. The NAME model maintained by the UK meteorology office provides background data for the Leicestershire are modelling by Airviro.
P3.2 MACROSCOPIC air quality modelling	This process OPANA provides a facility to study long-term changes in air quality caused by transport and environment policy, and to assess the potential impact on health. It takes Outputs from the MAGNOSCOPIC air quality model, UK Meteorological Office weather forecast data, Land use, Emissions database and Topography and produces air quality forecasting within a time horizon of 24-48 hours to establish a pollution background level.
P3.3 MESOSCOPIC air quality modelling	This process has three tasks: (1) to produce 24 and 48 hour air quality forecasts, (2) to broadcast the forecast via radio during the afternoon and evening when people return home; and (3) to provide information to allow the network manager to select an appropriate transport demand management strategy, including regulatory restrictions. The process requires information about the major sources of air pollution in the area (point sources such as factories, offices, shops; area sources such as housing or industrial estates; and line/road sources). Dynamic and static information is collected for each source. Dynamic information includes that derived from formulae which describe the emission as a function of outdoor temperature or as a function of day, week and time and monitored traffic data. Static information covers

Process	Description
P3.4 Noise modelling	<p>chimney height, exhaust gas temperature, co-ordinates, etc.</p> <p>This process is being developed for Leicester within the HEAVEN project. It includes both source emissions from segmented links (road sections) and point sources coupled with sound propagation using proven techniques (e.g. search triangle or ray tracing methods). The model uses as input both static and live traffic data from the Leicester SCOOT and TRIPS systems, which is already collected for use by Airviro. Algorithms for source noise generation are taken both from existing EC national standards, as well as SCOOT data specific algorithms developed from use of the TUNE micro-simulation model. Output from the model will be in the format of Airviro grid data for a number of noise parameters (e.g. <math>L_{A10}</math>, <math>L_{Aeq}</math>, <math>L_{A50}</math>, <math>L_{A90}</math>). These grids may then be used for noise mapping purposes or incorporated into appropriate GIS (Geographical Information Systems) software databases to give a measure of the number of people affected by noise.</p>
P4 Scenario development	<p>This process covers a number of sub-processes:</p> <ul style="list-style-type: none"> <li>• The identification of a particular traffic-related problem;</li> <li>• The development of a number of potential scenarios to alleviate that problem;</li> <li>• The collection of data to describe the scenarios;</li> <li>• The off-line modelling of the scenarios using the relevant tools;</li> <li>• The evaluation of the output from the modelling;</li> </ul> <p>The collection of data and modelling for an individual scenario requires the following steps:</p> <ul style="list-style-type: none"> <li>• Collection of required scenario data (network data, site geometry etc);</li> <li>• Collection of the base traffic demand levels;</li> <li>• Collection of base meteorological and environmental data.</li> <li>• Modelling the traffic-related impacts due to the scenario;</li> <li>• Use of the revised the revised traffic estimates to model the noise and air-quality impacts of the scenario.</li> <li>•</li> </ul> <p>Finally an objective function, based on the weighted output of all three modelling steps, (traffic, air-quality and noise) is used to ‘score’ the scenario. This score, along with other required scenario data is stored in the system database. New scenarios may be added or old scenarios removed from the database at any time.</p>
P5 Interface to information platform	<p>This process is responsible for providing the data about traffic, air quality, meteorological conditions, imported pollution, health and noise, and the modelling outputs to the various end-users of the system over the internet platform.</p>
P6 DSS operator interface	<p>This process consists of a user-friendly environment that supports the operator activities concerned with the DSS maintenance and operation.</p>
P7 Evaluation	<p>This process is responsible for continuously comparing modelled with predicted traffic, air quality and noise levels to assess the</p>

Process	Description
	appropriateness of the action and indicating the need for a different action if required.

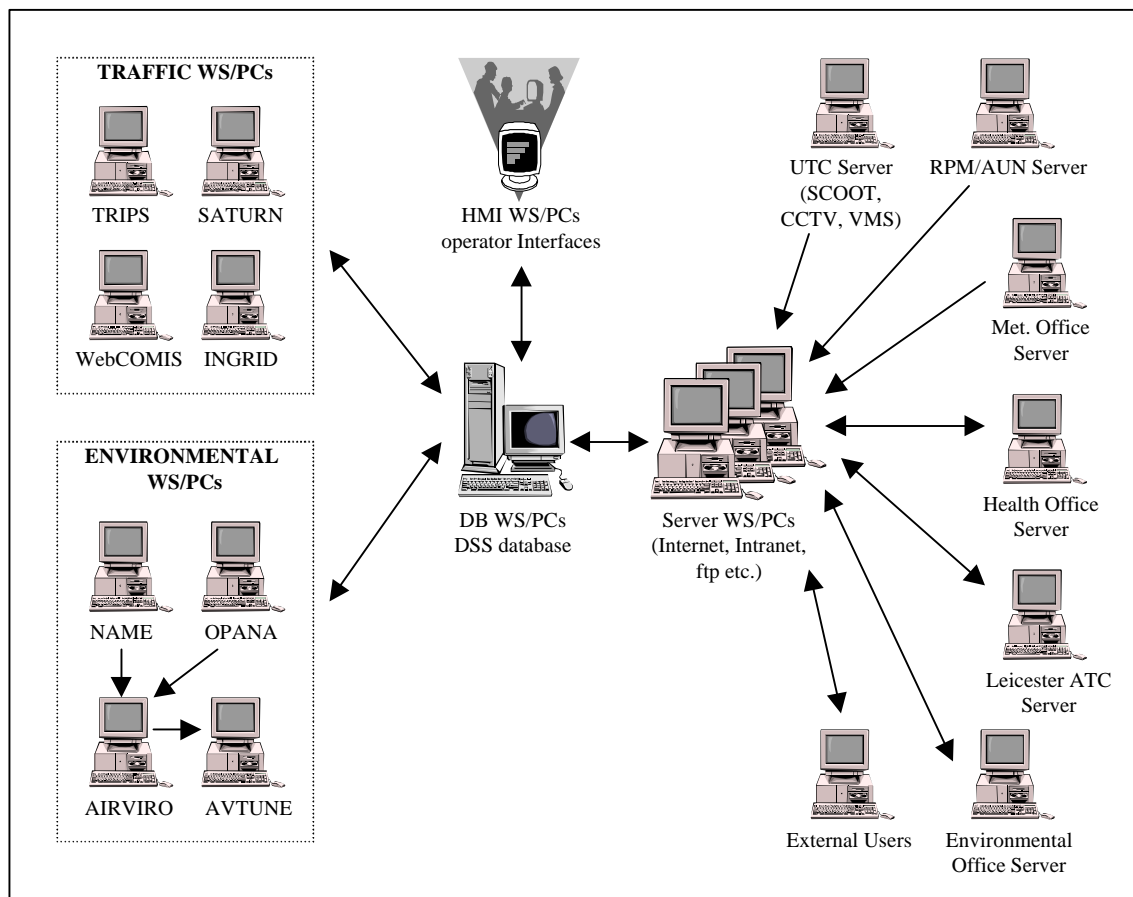
**Table 4.2.1 – Processes and sub-processes of the Leicester DSS**

### 4.2.2 HW Components

The Leicester DSS is based on a distributed architecture. Several workstations are dedicated to run the software processes grouped as:

- Data collection processes devoted to interface with external traffic data sources
- Traffic processes devoted to traffic modelling
- Air quality processes devoted to air quality modelling
- Noise processes devoted to traffic noise modelling
- Scenario processes devoted to scenario management
- HMI operator interfaces

Figure 4.2.2 below shows the hardware components of the top-level physical architecture of the Leicester DSS. On the left of the figure are the modelling processes and on the right are external network servers which provide input data for the models via internet, intranet or ftp. In the centre of the system is the database workstation/PC which is connected to HMI for operator interfaces.



**Figure 4.4.2 – Hardware components and interconnections of the Leicester DSS**

### 4.2.3 Connections

The connections of the hardware components of the Leicester DSS are depicted in Figure 4.2.3.

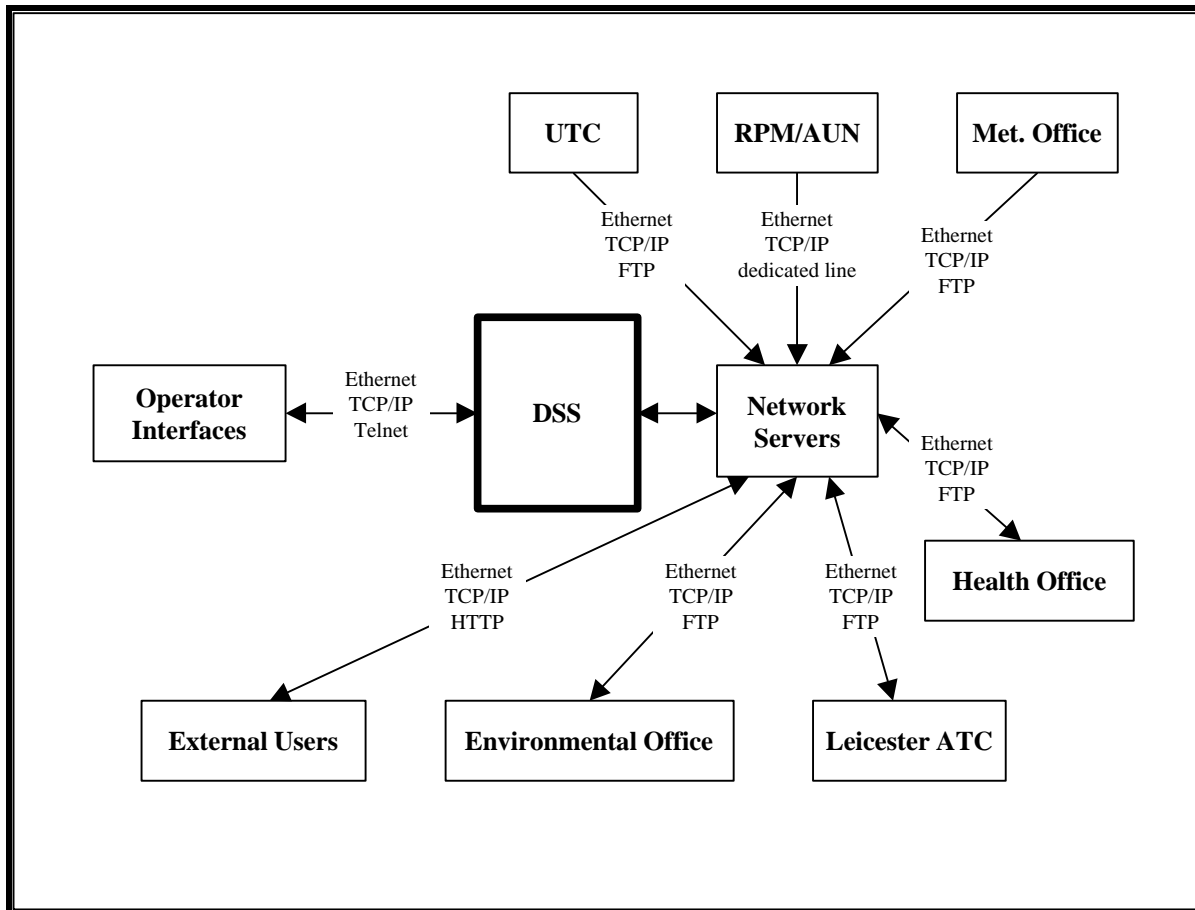


Fig. 4.2.3 – Communication architecture of the Leicester DSS

### 4.2.4 Implementation constraints

The following constraints have been identified and will be considered in the specification phase of the implementation.

#### 4.2.4.1 Operating system

The Airviro system operates in the UNIX environment (HP-UX). Other PC systems at LCC use Win NT4 Workstation.

#### 4.2.4.2 Network protocols

The various systems are connected using TCP/IP or LAN networking protocols. Provision for public access to information will be through HTTP access to a specified service provider.

#### 4.2.4.3 Database engine

The database engine used will be the one nested inside the Airviro System. No other database development is envisaged. Traffic and air pollution monitoring data are pre-processed separately before entry into the Airviro database.

#### 4.2.4.4 Topological reference

Not significant

#### 4.2.5 Actions to implement

The actions identified to implement the DSS concern two different aspects:

- SW development
- HW provision

Such activities are described in the following paragraphs. Their time frame is described through the GANTT chart in Chapter 5. Concerning system components connection, all requirements are currently in place at Leicester City Council as part of the EFFECT and EQUAL projects, previously funded by the EU Forth Framework Programme.

##### 4.2.5.1 SW development

The main actions in software development are the development of all of the components of the AVTUNE noise model (i.e. all processes 3.4) between ITS and SMHI. Additionally SMHI are to implement the RISO dispersion model within Airviro and configure the health statistics database.

#### ACTION SUMMARY

Action	Responsible partner
<b>Specification activities</b>	
Specification of Input Data: Source Emissions Model	ITS/SMHI
Background Noise Model	ITS/SMHI
Propagation Noise Model	ITS/SMHI
Verification Activities	ITS/SMHI
Noise Exposure Model	ITS/SMHI
Specification of Output Data: Noise Level Data	ITS/SMHI
Noise Grid/Contour Data	ITS/SMHI
Noise Exposure Data	ITS/SMHI
<b>Development activities</b>	
3.3.4 Installation of Dispersion Model (RISO)	SMHI
3.4.1 Development of Source Emissions Model	ITS/SMHI
3.4.2 Development of Background Noise Model	ITS/SMHI
3.4.3 Development of Noise Propagation Model	ITS/SMHI
3.4.4 Development of Validation Activities	ITS/SMHI

##### 4.2.5.2 HW provision

The following actions are identified:

Action	Responsible partner
Hardware specification	ITS/SMHI
Hardware provision	SMHI
System and LAN configurations	SMHI

### 4.3 Physical Aspects of the Paris DSS

The following figure 4.3.1 shows the top level DFD diagram of the Paris DSS demonstrator and identifies the main processes, data stores and data flows.

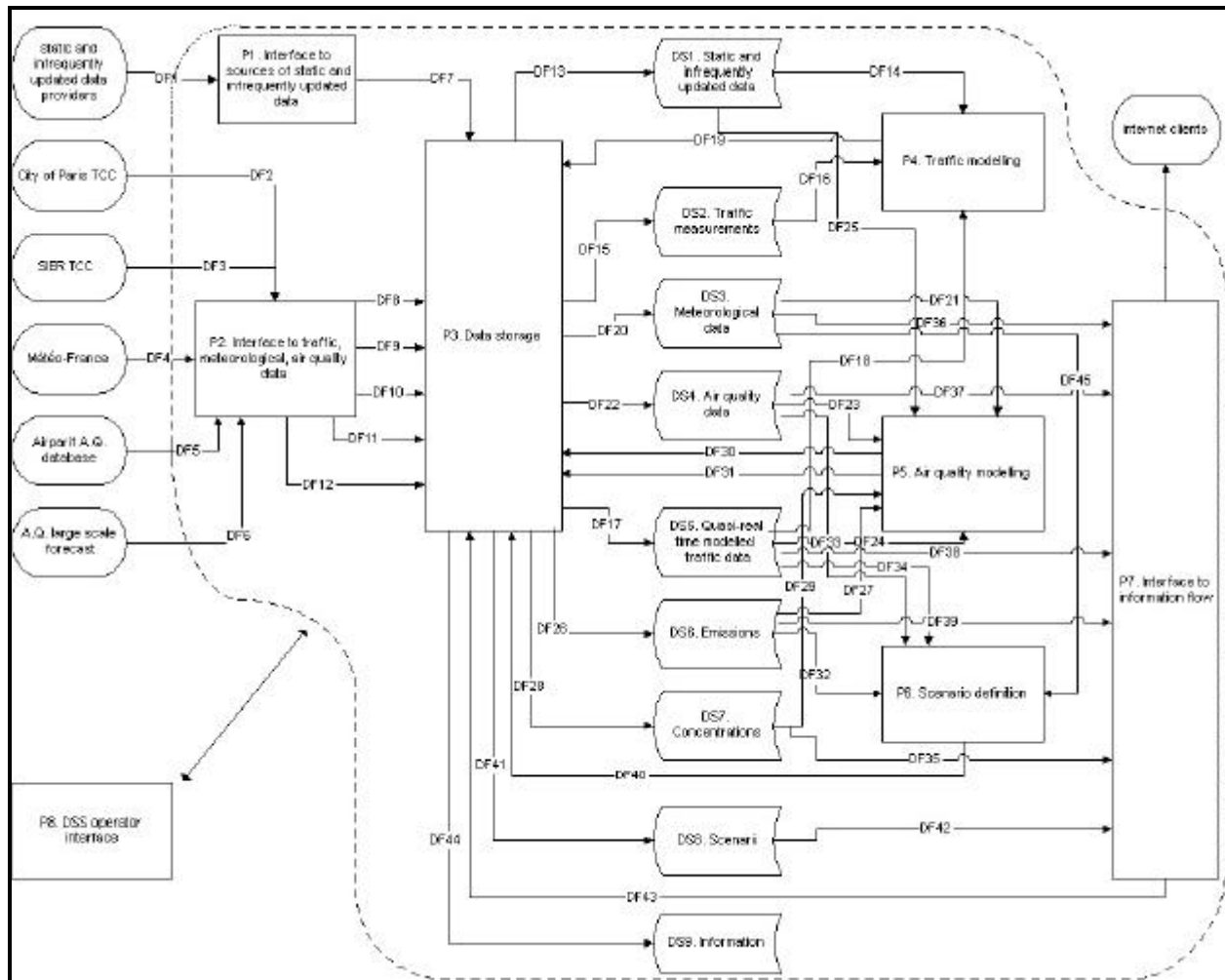


Figure 4.3.1. – The top level Data Flow Diagram of the Paris DSS

#### 4.3.1 SW Components

The following table 4.3.1 summarises the main processes and sub-processes identified at the detailed specification stage. More detailed tables are reported in the Annex A. These tables include specification of requirements, data flows and mapping to the functions to be implemented.

Process	Description
P1 Interface to sources of static & infrequently updated data	This process is a database application that import static and infrequently updated data into the DSS database with the period upgrade considering modifications of the different types of data.
P2 Interface to traffic, meteorological, air quality data	This process is in charge of interfacing with sources of data for traffic, meteorological and air quality measurements.
P2.1 Interface to traffic control centre of City of Paris	This process acquires real-time traffic flows and average traffic speed data from around 50 pinpoint measurement stations from City of

Process	Description
	Paris network.
P2.2 Interface to traffic control centre of SIER	This process acquires real-time traffic flows and average traffic speed data from around 50 pinpoint measurement stations from the SIER network.
P2.3 Interface to meteorological data sources (Météo-France)	This process acquires meteorological conditions and forecast from the French Meteorological Office (Météo-France).
P2.4 Interfaces to air quality data sources	This process acquires air quality data measurements from the Airparif air quality database and air quality large scale forecast available from some European web sites.
P3 Data storage	This process is in charge to load into the DSS databases or to store all the data received by the external interfaces as well as the results of the modelling processes.
P4 Traffic modelling	This process provides for a modelling of traffic for the whole controlled network. The modelling considers 100 pinpoint traffic measurements available and modelled traffic assignments.
P4.1 Traffic status modelling	This process computes an estimate traffic status on more than 34000 links of the regional traffic network. The computing considers the modelled traffic assignment for the 34000 links and 100 real-time pinpoint measurements. The estimate traffic status is computed each hour, with a time resolution of one hour (“quasi real-time”). The process applies data validation methodologies in order to assure that the provided status is based on the best data available. It maintains an historical archive of traffic data classifying data on typical day types on day of the week and meteorological conditions.
P4.2 Transportation modelling	This process computes a periodical update of the modelled traffic assignments. It takes as input the previous assignments, the network data, and the estimate traffic status matrices computed by the process P3.1 “Traffic status modelling”.
P4.3 Demand management modelling	Based on traffic network data and on specifications of traffic management strategies provided by traffic authorities, this process constructs traffic matrices corresponding to TDMS scenarios, for subsequent modelling of impact on air quality.
P4.4 Traffic modelled output validation	The process compares estimate traffic status matrices and traffic measurements in order to assess the quality of the estimate traffic status data.
P4.5 Traffic impact evaluation	This process consists in comparing modelled traffic assignments for typical days with traffic status estimates delivered by the traffic status modelling function during days where particular TDMS measures are in force. It yields data on the performance of these TDMS measures in terms of impact on traffic, as opposed to their final performance in terms of impact on air quality.
P5 Air quality modelling	Provides for a modelling of pollutants emissions and concentrations for the controlled network and for the whole Ile-de-France region. The modelling considers the observed traffic and the current meteorological conditions available.
P5.1 Traffic related emissions modelling	This process computes the linear traffic related emissions based on the estimate traffic status and emission related parameters. The modelled traffic related emissions are computed each hour with a

Process	Description
	time resolution of one hour (“quasi real-time”).
P5.2 Other sources emission modelling	This process computes the gridded emissions for the other sources of emissions (industries, airports, diffuse sources like domestic heating or biogenic emissions). The input data are the annual emissions of each source, the time profiles allowing the distribution of annual emissions into hourly emissions and sources and land-use characteristics. The other sources modelled emissions are computed once a year, with a time resolution of one year.
P5.3 Regional background dispersion modelling and forecast	This process supplies a quasi real-time evaluation and a forecast to J, J+1 and J+2 of background concentration of NO <sub>2</sub> and O <sub>3</sub> over all the regional domain. It considers the meteorological forecast, the emissions data and the air quality data. The regional background pollutant concentrations forecast for J, J+1 and J+2 is computed once a day (early in the morning), with a time resolution of one hour. The quasi-real-time evaluation of regional background pollutants concentration is expected to run every 3 hours and to provide for concentration fields for the three previous hours.
P5.4 Street level dispersion modelling	This process supplies an evaluation of pollutant dispersion providing values of NO <sub>2</sub> , CO, PM <sub>10</sub> , C <sub>6</sub> H <sub>6</sub> along all the traffic network links, for every hour. The input data for this process are the street characteristics and built environment of streets, the meteorological data for the previous hours, the background concentrations of pollutants for the previous hours coming as output from the “regional background dispersion modelling and forecast” process and the linear hourly traffic related emissions provided as output by the “traffic related emission modelling” process. The modelled street level pollutants concentration are computed every three hours.
P5.5 Modelled air quality validation	This process determines the performance of the simulations by comparison of the model outputs with the air quality observed data. This process supplies the information platform with uncertainties and levels of confidence associated to the results.
P5.6 Air quality impact evaluation	This process supplies an evaluation of the performance of a specific scenario based on a TDMS by making a comparison between the air quality modelled output based on this scenario and the air quality modelled output based on a “normal” situation. This process will be used only three to four times per year to evaluate specific TDMS that could be proposed by authorities.
P6 Scenario definition	This process is a tool for the operator to design scenarios for off-line evaluation. It collects and stores as a specific item the minimum set of information that qualifies a “scenario” for a specific time period. This set of coherent information consists of: <ul style="list-style-type: none"> <li>• the description of traffic parameters for the defined time period which are stored in the system traffic database</li> <li>• the emissions data stored in the emissions database</li> <li>• the meteorological data stored in the meteorological database</li> <li>• the measurements performed by the air quality monitoring network and the air quality forecasts which are stored in the air quality database</li> </ul>

Process	Description
	Such information classes is organised and stored in the system data base on a regular basis. Then scenarios can be qualified just defining and storing the references (names) to the data sets for future use together the time to which they refer. The operator can use such a process to build its own scenarios for simulation and evaluation purposes.
P7 Interface to information flow	This process supplies selective traffic, emissions and air quality observations and modelling results to the internet and intranet platforms in order to communicate information to the various end-users of the system
P8 DSS operator interface	This process consists of a user-friendly environment that supports the operator activities concerned with the DSS maintenance and operation.

Table 4.3.1 – Processes and sub-processes of the Paris DSS

### 4.3.2 HW Components

The following figure 4.3.2 shows the hardware components of the DSS, the connection scheme at the central level and the interconnections envisaged with the external subsystems.

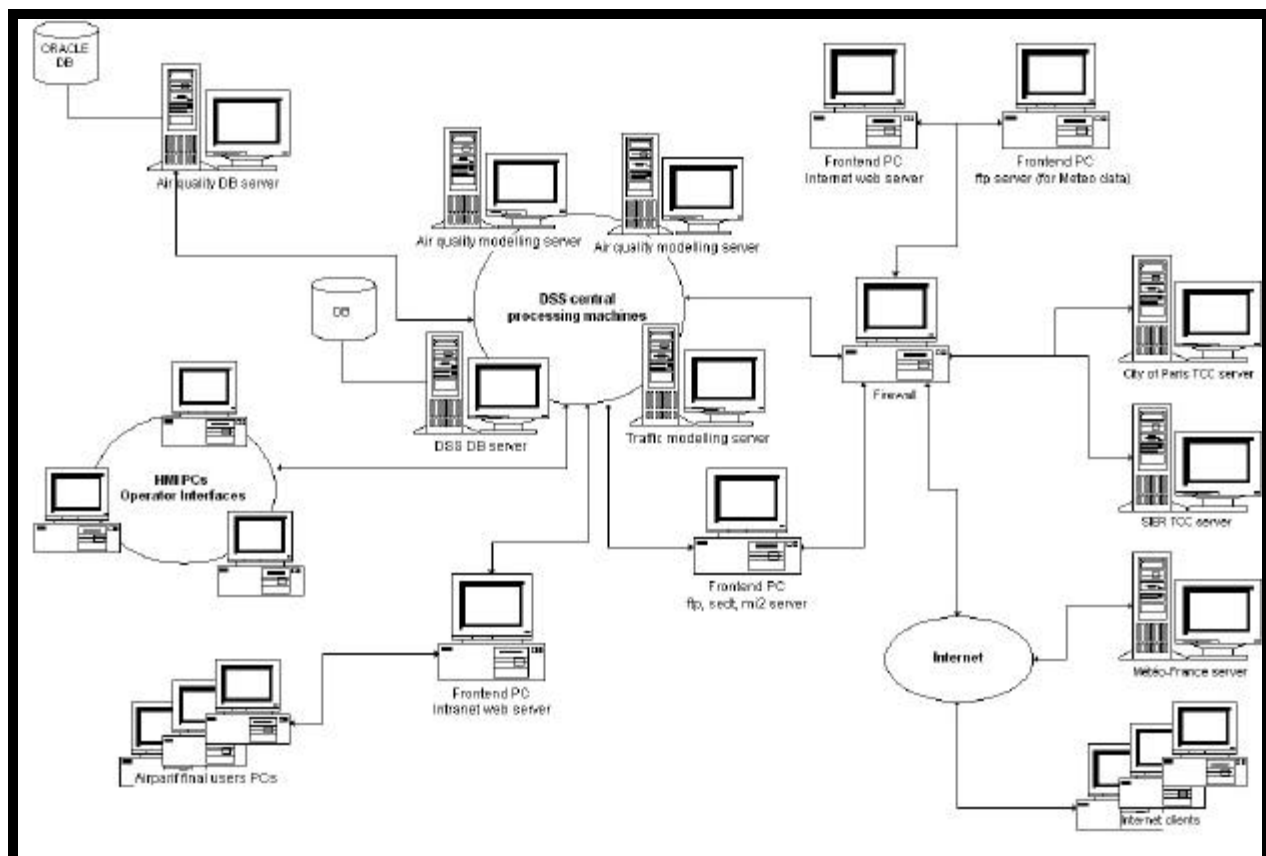


Figure 4.3.2 – Hardware components and interconnections of the Paris DSS

The physical architecture describes the allocation of physical units that will perform the different software processes grouped as :

- frontend processes devoted to connect to external interfaces and data sources
- DB processes devoted to store data
- traffic processes devoted to traffic modelling
- pollution processes devoted to air quality modelling
- HMI operator interfaces

A great part of data stores are stored on relational databases. For compliance with databases already available at Airparif, the adopted database engines are Oracle or Access.

### 4.3.3 Connections

The following figure 4.3.3 describes the way the different elements of the system communicate and the communication protocols the element use.

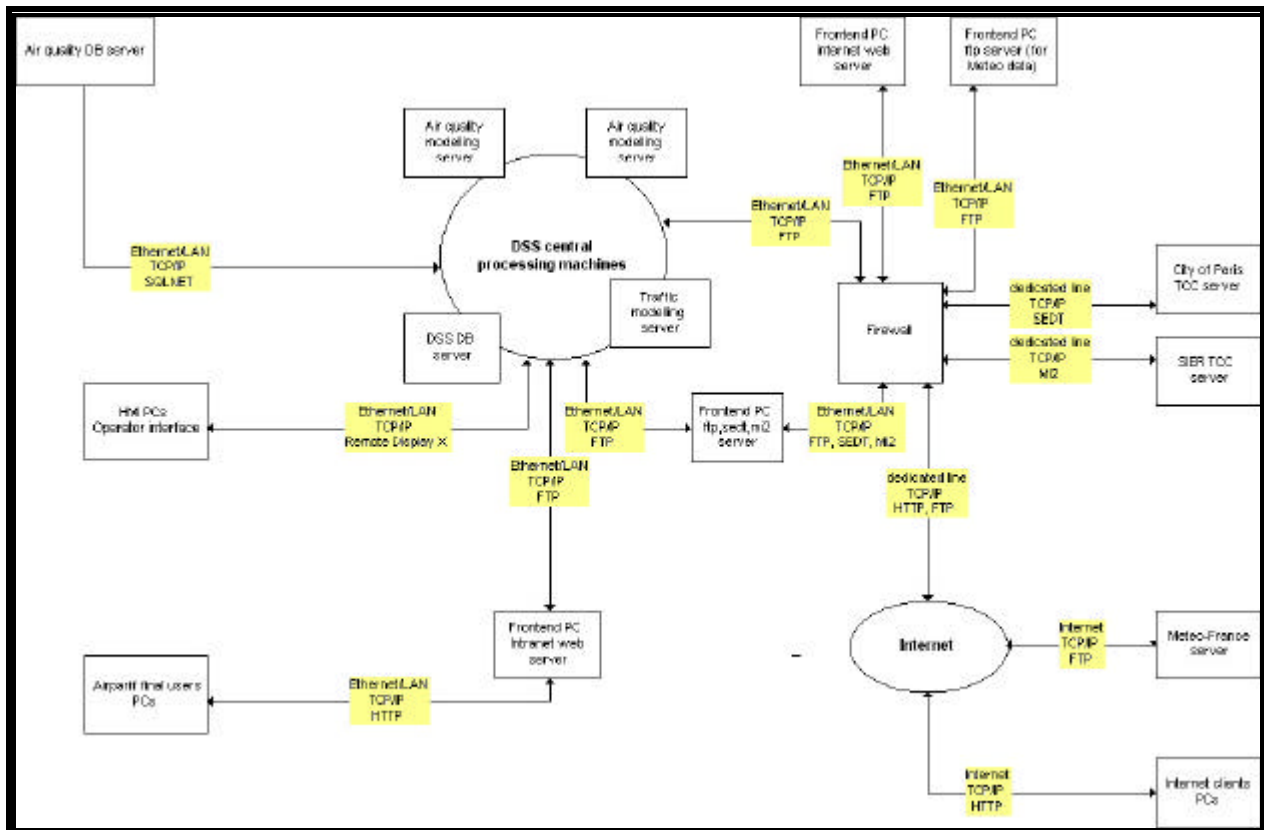


Figure 4.3.3 – Communication architecture of the Paris DSS

The DSS is connected to external interfaces identified as :

- City of Paris TCC, the source of information for the traffic measurements in Paris. The DSS connects to the City of Paris using a specific protocol called SEDT via a dedicated line.
- SIER TCC, the source of information for traffic measurements in the rest of the Ile-de-France region. The DSS connects to the SIER using a specific protocol called MI2 via a dedicated line.

- Météo-France servers, the source of information for meteorological conditions and forecast. The DSS connects to the Météo-France servers using ftp protocols.

The DSS is connected to the air quality database of Airparif - the source of information for pollutants measurements, by the local area network by the way of SQL requests.

The network physical layer is based on Ethernet. The transport layer is based on TCP-IP interfaces. The used network applications protocols are FTP, "SEDT", "MI2", HTTP, SMTP, SQLNET, Remote Display X or Telnet.

The DSS operators run and supervise the whole system via Remote Display X.

#### **4.3.4 Implementation constraints**

The following constraints have been identified and will be considered in the specification phase of the implementation.

##### **4.3.4.1 Operating system**

No specific constraints have been identified for the operating systems. Processes could be developed in a Unix environment or in a Microsoft Windows NT-2000 environment.

##### **4.3.4.2 Network protocols**

The basic network protocol adopted for communication within the local area network and with the external interfaces is TCP-IP.

##### **4.3.4.3 Database engine**

Shared data stores are stored on an ORACLE database server or on an ACCESS database server.

##### **4.3.4.4 Topological reference**

All processes uses a common topological reference to store and retrieve common data objects. Internally to the processes other reference can be used providing for a mapping from the internal to the common topological reference.

Topological elements are geo-referenced using standard referencing systems.

#### **4.3.5 Actions to implement**

The actions identified to implement the DSS concern three different aspects:

- SW development
- HW provision
- Connections activation.

Such activities are described in the following paragraphs.

#### 4.3.5.1 SW development

The implementation for the DSS demonstrator has been split into several phases as described in the following. Each phase has been given a global deadline and thus activities related to the phase have been planned to cope with the overall phase deadline.

Main processes will be able either to run off-line on a static set of data or on-line getting data in real time from the existing system.

The off-line capability will give the possibility to simulate the effects of different traffic scenarios changing the static set of data given as starting point. Thus it is the base for scenario simulation and assessment of impacts derived from different demand management actions.

##### Specification phase

The specification phase has the objective to draw a detailed specification of each process, data store and data flow identified by the overall architecture. The result of this phase is a full set of software specification for the development of software modules and the integration of the overall system.

Deadline: End of May 2001

##### Off-line development phase

The goal of this phase is to have all process in operation providing their results from static sets of data. This will enable to develop each process independently from the others and to carry out a preliminary technical assessment of each implemented function. The phase cover both the development of new software modules and the integration of existing packages into the DSS environment.

Deadline: End of October 2001

##### On-line development phase

The on-line phase will take care of the integration of different processes up to build the overall system running in real time. The result of this phase allows for an overall verification of the system. The verification could highlight needs for re-engineering or software modifications before the evaluation of the demonstrator.

Deadline: End of February 2002

The off-line and on-line phases are linked by a technical verification of the system and if needed by re-engineering actions on the system.

#### ACTION SUMMARY

The following actions are identified:

Action	Responsible partner
<b>Specification activities</b>	
Static and infrequently updated data flows specification	AIRPARIF
Traffic data flows specification	SIER/City of Paris/CBC/AIRPARIF
Meteorological data flows specification	Météo-France/AIRPARIF
Air quality data flows specification	AIRPARIF
Data stores specification	AIRPARIF/DREIF(SETEC)
Traffic modelling processes specification	DREIF/City of Paris /AIRPARIF
Air quality modelling processes specification	AIRPARIF
Scenario definition process specification	AIRPARIF/DREIF/City of Paris
Interface to information flow specification	AIRPARIF/DREIF/City of Paris
DSS operator interface specification	AIRPARIF
<b>Development activities</b>	
P1 Static data import	AIRPARIF
P2.1 Interface to TCC of City of Paris	AIRPARIF/CBC

P2.2 Interface to TCC of SIER	AIRPARIF/CBC
P2.3 Interface to meteorological data sources	AIRPARIF
P2.4 Interface to A.Q. data sources	AIRPARIF
P3 Data stores	AIRPARIF
P4 Traffic modelling	DREIF(SETEC)
P5 Air Quality modelling	AIRPARIF
P6 Scenario definition process	AIRPARIF/DREIF(SETEC)
P7 Interface to information flow	AIRPARIF/DREIF/City of Paris
P8 DSS operator interface	AIRPARIF

#### 4.3.5.2 HW provision

The following actions are identified:

Action	Responsible partner
Hardware specification	AIRPARIF
Hardware provision	AIRPARIF
System and LAN configurations	AIRPARIF

#### 4.3.5.3 Connection activation

The following actions are identified:

Action	Responsible partner
Provision of a dedicated line between City of Paris and Airparif	AIRPARIF/City of Paris
Provision of a dedicated line between SIER and Airparif	AIRPARIF/SIER
Configuration of the connections between Météo-France servers and Airparif	AIRPARIF/Météo-France

## 4.4 Physical Aspects of the Prague DSS

The following figure 4.4.1 shows the top level DFD diagram of the Rome DSS demonstrator and identifies the main processes, data stores and data flows.

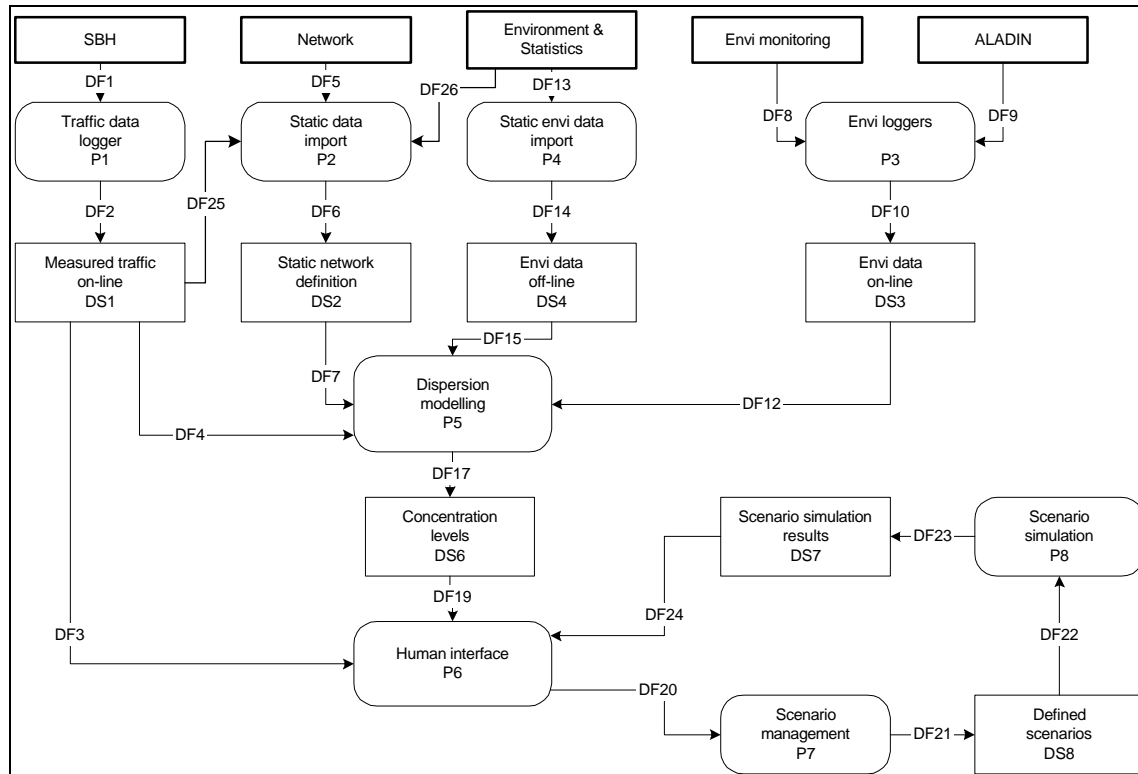


Figure 4.4.1 – The top level Data Flow Diagram of the Prague DSS

### 4.4.1 SW Components

The top level DFD identifies the main processes that implement the functions described in the functional architecture. The following table 4.4.1 summarises the main processes and sub-processes identified at the detailed specification stage. Annex A reports detailed tables including specification of requirements, data flows and mapping to the functions to be implemented.

Process	Description
P1 Traffic data logger	Provides reading on-line traffic data from the internal SBH system format and writing into the DSS databases. Enables setting the time resolution parameters and data type selection.
P1.2 SBH gateway	The process interfaces with the SBH system to access the selected data for modelling purposes. It enables the time resolution parameters setting. The SBH-DSS location reference translation is provided.
P1.2 DSS-SBH interface	The process is developed to read the data made accessible by the SBH gateway and load it into the DSS databases. Data conversion into the modelling-required format is included.
P2 Static data import	The processes under this category are off-line services developed to transfer road network description source data into the DSS databases. The processes are GIS and database human-controlled applications.
P2.1 Road network import	The process is in charge of conversion of road layout data into the DSS GIS format and modelling database format. Enables update of the road description database for nowcast and scenario evaluation. The process is GIS application requiring the human interface.
P2.2 Traffic survey	The process is an interface provided update of average traffic volumes on the non-monitored network. Provides inputs for the off-line traffic modelling.
P2.3. Off-line traffic modelling	The process serves for modelling the traffic volumes on the selected network. The surveyed data are validated and traffic volumes are assigned to each road segment.
P2.3.1 O/D matrix building	The process is a database application for estimation of locations of traffic sources and destinations. It is calculated on basis of land-use data, population data and preset parameters. The matrix is necessary for traffic assignment calculation.
P2.3.2 Traffic assignment	The O/D matrix data are distributed along the modelled road network and traffic volumes are calculated. Also, the surveyed traffic volumes are compared with the modelled results and the calculated volumes are calibrated. The process result is the balanced data set of traffic volumes used for the emission modelling and scenario management.
P3 Envi loggers	The process for transmission of the Air Quality and meteo measurements and meteo forecast from the ALADIN model into the DSS dispersion model database. The loggers maintain on-line connection and regular DSS databases update.
P3.1 AIM interface	The procedure serves in order to transmit and re-format the on-line data from Air quality measurement system into the DSS dispersion model database. It maintains the connection and provides also the location reference translation.
P3.2 Meteo interface	The procedure serves in order to transmit and re-format the on-line data from meteo measurement system in SWING format into the DSS dispersion model database. It maintains the connection and provides also the location reference translation.
P3.3 ALADIN interface	The procedure serves in order to transmit and re-format the on-line data from meteo forecast model ALADIN system in GRIB format into the DSS dispersion model database. It maintains the connection and provides also the location reference translation.
P3.4 Mast interface	The procedure serves in order to transmit and re-format the on-line data from mast data monitoring station into the DSS dispersion model

Process	Description
	database. It maintains the connection and provides also the location reference translation. The process is the modular application within the dispersion database time series module.
P4 Static envi data import	The set of procedures built in order to provide maintenance of surface data and dispersion model parameters. The process is semi-manual database and GIS application.
P4.1 Surface data import	The procedure is a GIS application in order to analyse the land-use planning data and topography data into the DSS dispersion modelling database.
P4.2 REZZO import	The procedure is a database application in order to import data from central register of air pollution sources into the DSS dispersion modelling database.
P4.3 Model parameters import	The procedure is semi-manual process of model parameters re-format and import to the Airviro modelling database
P5 Dispersion modelling	The process consists of set of modular sub-processes linked within common modelling tool in order to provide maintenance of DSS envi model database and dispersion modelling. The process is a complex database application.
P5.1 Time series module	Provides maintenance of on-line data inputs into the DSS modelling database. Keeps communication with P1 and P3 procedures.
P5.2 Time series data presentation	The procedure is an application for human verification of the time series data and analysis providing up on the data. The procedure is the database application.
P5.3. Dispersion modelling procedure	Database application, which enables to run in pre-set time resolution regimes. Basically, there will be two regimes in operation within HEAVEN project: <ul style="list-style-type: none"> <li>on-line, including the hourly nowcast estimation and hourly update of 24 hours prediction of air pollution concentration.</li> <li>off-line, including the 10 years prediction for land-use planning and air pollution assessment of scenario proposals.</li> </ul> The application enables to provide modelling by both Eulerian and Grid model as well as the basic Canyon model equations. The calculation of the results includes the emission modelling and, as a next step, the calculation of dispersion. These two sub-processes are integrated into one logical module.
P6 Human interface	This process is a set of user interfaces, which enables to provide DSS maintenance and operation. The three sub-processes are identified: <ul style="list-style-type: none"> <li>information representation and management</li> <li>operator intervention</li> <li>scenario results evaluation</li> </ul>
P7 Scenario management	The process covers three main sub-processes in order to handle the preparation and implementation of UTC scenarios: <ul style="list-style-type: none"> <li>definition of scenarios and measures</li> <li>evaluation of possible impacts</li> <li>building the scenario implementation plan</li> </ul> The scenario is set of information and action for defined time period: <ul style="list-style-type: none"> <li>the traffic demand</li> <li>the available road network</li> <li>the possible traffic restrictions</li> </ul>

Process	Description
	<ul style="list-style-type: none"> <li>• the real-time traffic flows</li> <li>• the current UTC status</li> <li>• the real-time environmental data</li> <li>• the real-time dispersion prediction</li> <li>• the possible measures</li> </ul> <p>The measure is an action to implement in order to improve the current traffic or environmental status, which contains:</p> <ul style="list-style-type: none"> <li>• information for drivers, passengers and general public</li> <li>• changes in UTC signal plans</li> <li>• traffic restrictions</li> <li>• temporary changes in public transport services</li> </ul>
P8 Scenario simulation	The process is built to provide the simulation of scenarios impact in order to predict the traffic loads and air pollution concentration levels.

**Table 4.4.1 – Processes and sub-processes of the Prague DSS**

#### 4.4.2 HW Components

The DSS system is based on a distributed architecture. Several workstations are dedicated to run the software processes grouped as:

- loggers                      modules devoted to connect to external on-line interfaces and data sources
- importers                    modules devoted to import the data from external off-line data sources
- traffic                        processes devoted to traffic off-line modelling
- dispersion                    processes devoted to dispersion modelling
- scenario                      processes devoted to scenarios management
- HMI                            operator interfaces

Workstations are connected on a Wide Area Network (WAN) inter-linked by internet protocol together and with the external interfaces. There is no central point where the majority of processes is to be provided.

The DSS databases is distributed set of GIS and relational databases. The core database is on-purpose built module within the Airviro system.

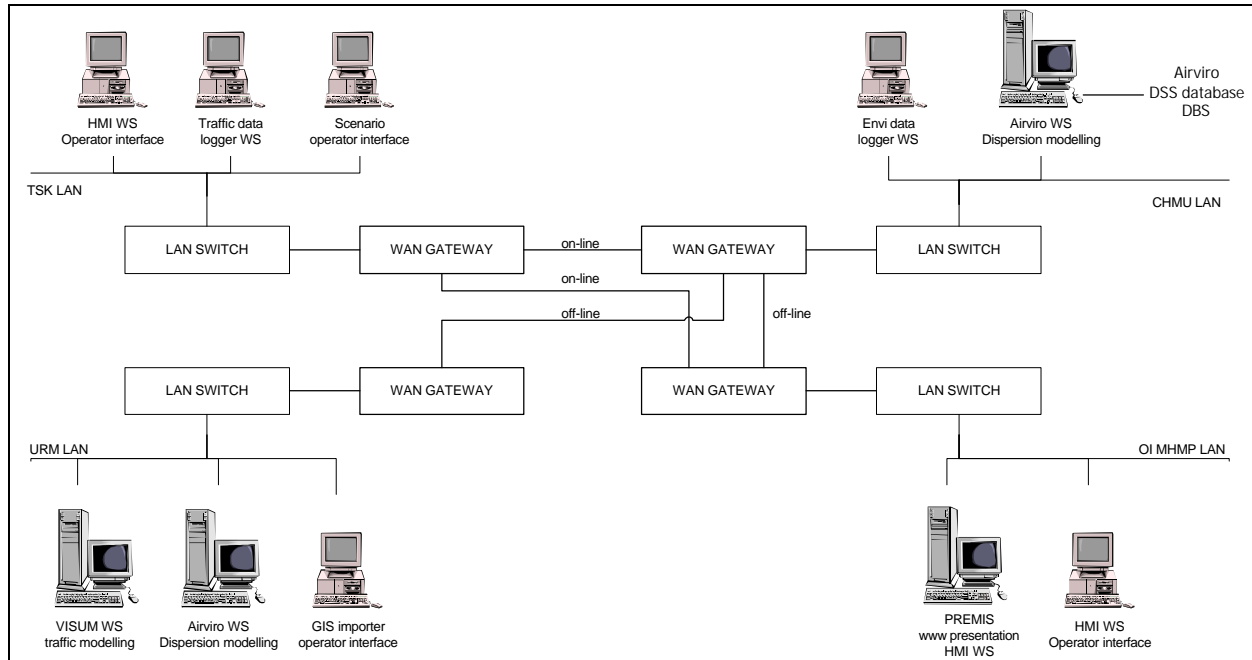
The following figure 4.4.2 shows the WAN diagram for the DSS system.

#### 4.4.3 Connections

The DSS system is connected to external interfaces identified as :

- SBH, the source of information of traffic load measurements and current UTC signal scenario status
- AIM, the source of information of Air quality measurements
- Meteo, the source of information of current meteorological conditions
- ALADIN, the source of information of weather forecast
- Mast, the source of information of mast complementary meteorological measurement

All the connections are based on TCP-IP interfaces making use of WAN technologies for the physical connections. The only exception is Mast connection, which is to be provided by modem connection.



**Figure 4.4.2 – The Wide Area Network scheme for the Prague DSS**

#### 4.4.4 Implementation constraints

The following constraints have been identified and will be considered in the specification phase of the implementation.

##### 4.4.4.1 Operating system

The basic processes – traffic monitoring, dispersion modelling and environmental monitoring are based on UNIX operation system.

The traffic modelling, import utilities and www information server are developed under MS Windows NT operating system.

##### 4.4.4.2 Network protocols

The basic network protocol adopted for communication within the LAN and with the external interfaces is TCP-IP.

##### 4.4.4.3 Inter-process communication

Inter process communication is based on socket connections using the TCP-IP protocol.

#### **4.4.4.4 Process synchronisation**

All the workstation on the TSK DSS-LAN will be time-synchronised on a common time server. Synchronisation with the external interfaced is devoted to the front/end processes. The synchronisation between dependant processes is based on notification of availability of data for a specific time interval.

#### **4.4.4.5 Database engine**

The data available for dispersion modelling are stored in internal on-purpose built robust and fast database system within Airviro module. No other central database is to be developed.

#### **4.4.4.6 Topological reference**

All processes uses a common topological reference to store and retrieve common data objects. Internally to the processes other reference can be used providing for a mapping from the internal to the common topological reference.

The basic co-ordination system for DSS is S-JTSK. Dispersion modelling is developed for S-42. The conversion routines are integrated into static envi data import processes.

### **4.4.5 Actions to implement**

The actions identified to implement the DSS concern three different aspects:

- SW development
- HW provision
- Connections activation.

Such activities are described in the following paragraphs.

#### **4.4.5.1 SW development**

The action plan integrates the software development and the connection building. The overall approach is to develop the off-line dispersion modelling as soon as possible and start the model parameters calibration in order to speed up the later calibration, which will take place after the on-line connection is established. The four milestones has been specified in the process implementation period.

##### **Specification phase**

The specification phase has the objective to draw a detailed specification of each process, data store and data flow identified by the overall architecture. The result of this phase is a full set of software specification for the development of software modules and the integration of the overall system.

Deadline: March 2001

##### **Off-line modelling phase**

The aim of the stage is to set up the air quality dispersion module and prepare all the static input data in order to start model calibration. The traffic modelling has already been put into operation and calibrated.

Deadline: April 2001

##### **On-line modellingdevelopment phase**

The on-line phase will take care of the integration of different processes up to build the overall system running in real time. The result of this phase allows for an overall verification of the system. The

verification could highlight needs for re-engineering or software modifications before the evaluation of the demonstrator.

Deadline: July 2001

The off-line and on-line phases are linked by a technical verification of the system and if needed by re-engineering actions on the system.

### ACTION SUMMARY

The following actions are identified:

Action	Responsible partner
data specification	URM/CHMU/TSK
data flows specification	URM/CHMU/TSK
procedures specification	URM/CHMU/TSK
static data preparation and import	URM/CHMU
data links procedures building	CHMU/TSK
on-line model upgrade	CHMU/TSK
off-line noise modelling launch	URM/CHMU
on-line dissemination procedures building	URM/OI MHMP

#### 4.4.5.2 HW provision

The following actions are identified:

Action	Responsible partner
hardware acquisition	URM
mast installation	CHMU
system and link configuration	CHMU/TSK

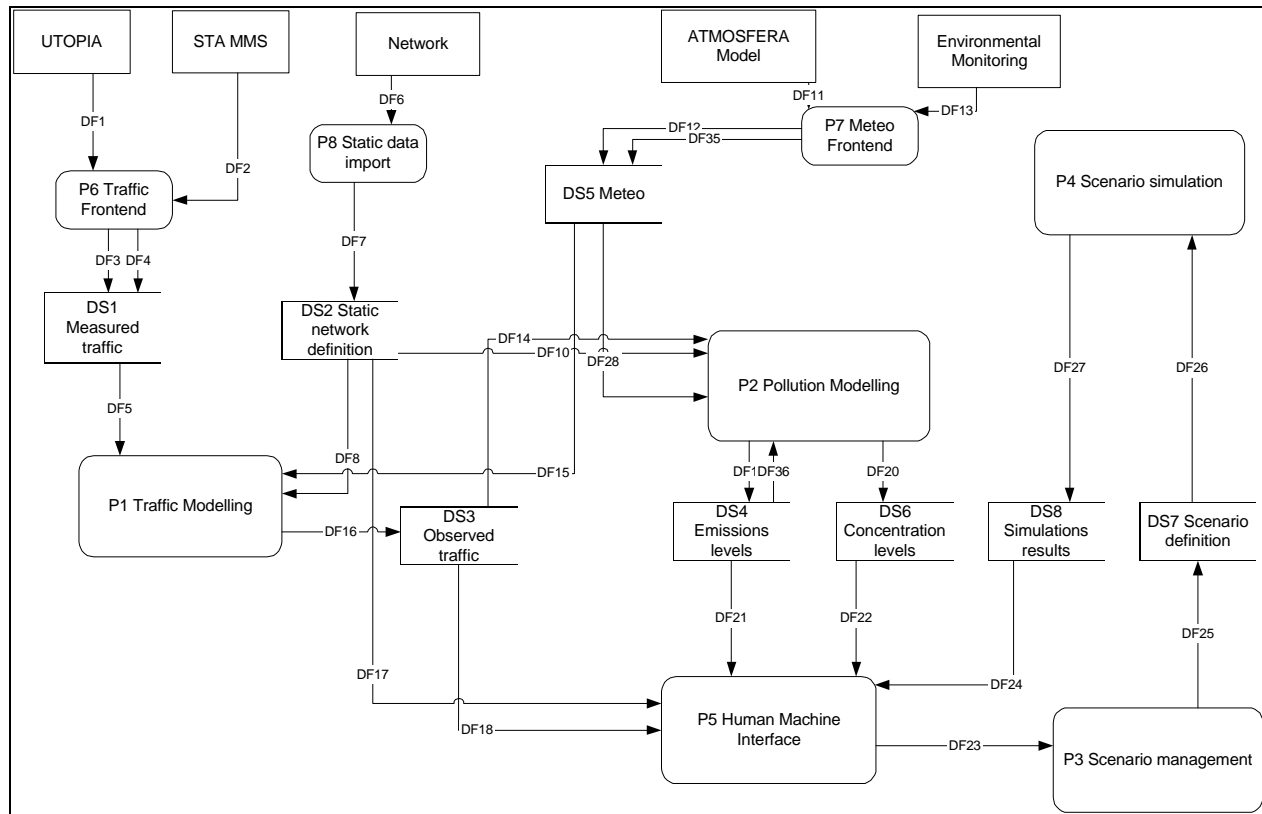
#### 4.4.5.3 Connection activation

The following actions are identified:

Action	Responsible partner
traffic data grabbing	TSK
meteo data grabbing	CHMU
AIM data grabbing	CHMU
meteo forecast grabbing	CHMU
data links building	URM/CHMU/TSK
on-line dissemination data links building	OI MHMP

### 4.5 Physical Aspects of the Rome DSS

The following figure 4.5.1 shows the top level DFD diagram of the Rome DSS demonstrator and identifies the main processes, data stores and data flows.



**Figure 4.5.1 – The top level Data Flow Diagram of the Rome DSS**

#### 4.5.1 SW Components

The top level DFD identifies the main processes that implement the functions described in the functional architecture. The following table 4.5.1 summarises the main processes and sub-processes identified at the detailed specification stage. Annex A reports detailed tables including specification of requirements, data flows and mapping to the functions to be implemented.

Process	Description
P1 Traffic modelling	Provides for a modelling of traffic for the controlled network. The modelling considers the traffic measurements available and the equilibrium assignment.
P1.1 Network observation	Provides the other process with an observed status of the network combining measured data and assigned traffic. It applies data validation methodologies in order to assure that the provided status is based on the best data available. It maintains an historical archive of traffic data classifying data on typical day types on day of the week and meteorological conditions.
P1.2 Network assignment	See sub-processes
P1.2.1 O/D Estimation	The O/D Matrix Estimation allows the system to build and to maintain a library of up-to-date O/D matrices each related to a type of

Process	Description
	<p>day and to a particular period of the day. Such a period is pre-defined one hour long.</p> <p>The process iterates his operation at over a defined time interval (between 30 and 90 minutes) and makes use of the traffic data collected during the previous time interval. Based on these data the function updates the O/D matrix available for such type of day and time of the day. The result is a new O/D matrix useful to perform the traffic assignment the next day (statistically compatible) during the time interval for which the O/D matrix is computed.</p>
P1.2.2 Traffic assignment	<p>The process employs user equilibrium criteria to determine how traffic propagates in the network according to the O/D matrix available for the current type of day and time of the day, and according the network made available to traffic.</p> <p>Traffic assignment covers all the links of the reference network and constitutes the tool to model traffic in the links where no measurement is available from traffic monitoring and control systems.</p>
P2 Pollution modelling	<p>Provides for a modelling of pollution emissions and concentrations for the controlled network. The modelling considers the observed traffic and the current meteorological conditions available.</p>
P2.1 Emissions modelling	<p>The process consists of three main sub-processes , each one dedicated to the emissions estimation for a specific set of links:</p> <ul style="list-style-type: none"> <li>◇ Modal Models 1, that handles the emission estimates for the links controlled by the UTOPIA UTC system, where traffic detailed information is complemented by the duty-cycle data</li> <li>◇ Modal Models 2, that handles the emission estimates for the links controlled by other monitoring and control systems (Roma '90, VMS, MMS, ZTL), where traffic information is less detailed and duty-cycle data are not available</li> <li>◇ Average Models, for the links where only traffic assignment information is available.</li> </ul> <p>A final sub-process performs the “fusion of the emission” data provided by the different models above, ensuring consistency of information related to different link categories.</p> <p>The suite of models covers at least the following pollutants: CO, O<sub>3</sub>, C<sub>6</sub>H<sub>6</sub>.</p>
P2.2 Concentration modelling	<p>The process runs every hour based on the emission data estimated for the previous hour. It estimates pollutant dispersions on the network links and provides the concentration in correspondence of number of hot-spot and other key points.</p> <p>Elaboration is based on meteo forecast (confirmed with a delay of some hours), measured and modelled traffic data.</p> <p>Concentration measurements, available only with a delay of one day, are used for off-line model calibration. In real time are available predictions of concentrations. These are used for a general check of consistency of the dispersions elaboration.</p>
P3 Scenario management	<p>This process implements the management of scenarios both in terms of supporting the definition of the scenario as to manager the evaluation of its possible impacts.</p>

Process	Description
	<p>A scenario is defined as a set of coherent information related to a well defined time period. It consists of:</p> <ul style="list-style-type: none"> <li>• the traffic demand</li> <li>• the available road network representation</li> <li>• the possible traffic restrictions</li> <li>• the real-time traffic related information provided by the -</li> <li>• traffic control systems</li> <li>• meteo data</li> <li>• the possible measurements performed by the environmental monitoring network</li> </ul> <p>Such information classes is organised and stored in the system data base on a regular basis. Then scenarios can be qualified just defining and storing the references (names) to the data sets for future use together the time to which they refer.</p> <p>The operator can use such a function to build its own scenarios for simulation and evaluation purposes.</p>
P4 Scenario simulation	This process deals with the management of the simulation of the defined scenarios in order to asses the impacts on traffic assignment, pollutant emissions and concentration levels.
P5 Human machine interface	<p>This process consists of a user-friendly environment that supports the operator activities concerned with the DSS maintenance and operation. For what is strictly concerned with the use of the DSS for TDMS evaluation and environment monitoring, three main sets of sub-process are identified:</p> <ul style="list-style-type: none"> <li>- Information representation and management</li> <li>- Operator intervention</li> <li>- Scenario results evaluation</li> </ul>
P6 Traffic front/end	The process is in charge of interfacing with sources of data for traffic measurements. It maintains communication and provides for location referencing translation if needed.
P6.1 UTOPIA gateway	The process interfaces with the UTOPIA UTS system in order to retrieve the traffic information needed for traffic modelling and emission estimation. The retrieved data is made of estimated turning percentages and duty cycles on the links controlled by UTOPIA. A location reference translation is performed in order to transform the UTOPIA reference into the DSS location reference.
P6.2 DSS-UTOPIA interface	The process is in charge to load into the DSS database the information received by UTOPIA. The process is mainly a database application that transfers the raw data received into tables of a relational database.
P6.3 DSS-MMS interface	The process is in charge to load into the DSS database the information available from the MMS database. The process is mainly a database application that transfers the data between the two databases.
P7 Meteo frontend	The process is in charge of interfacing with sources of data for environmental and meteo data measurements and forecasts. It interfaces with the network of monitoring outstations and with the ATMOSFERA model and provides for location referencing translation if needed.

Process	Description
P8 Static data import	The process import the network description into the DSS database. The process allows for updating the reference topological network with the period upgrade considering modifications of the network or topological data. Th process is a database application that import the reference network into the DSS database.

**Table 4.5.1 – Processes and sub-processes of the Rome DSS**

### 4.5.2 HW Components

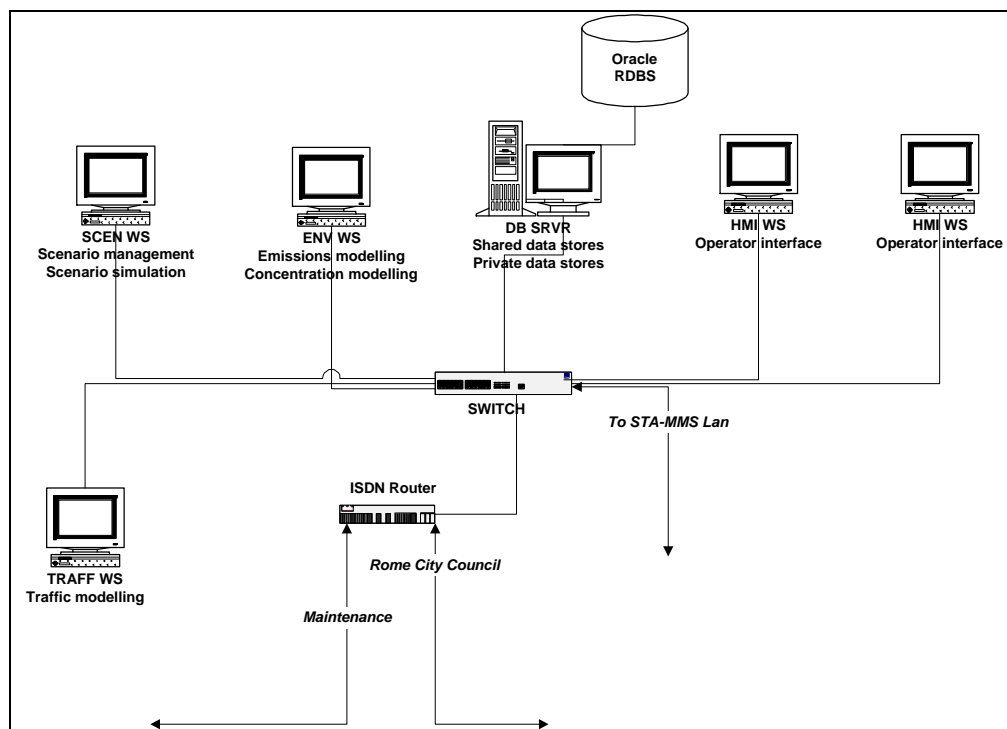
The Rome DSS is based on a distributed architecture. Different workstations are dedicated to run different software processes categories as follows:

- frontend SW modules devoted to connect to external interfaces and data sources
- traffic processes devoted to traffic modelling
- pollution processes devoted to pollution modelling
- scenario processes devoted to scenarios management
- HMI operator interfaces

The workstations are connected on a Local Area Network (LAN) linked on a Wide Area Network (WAN) with the external interfaces.

All data stores are stored on relational databases served by a system database server. For compliance with databases already available by STA the adopted database engine is Oracle.

The following figure 4.5.2 shows the LAN diagram for the DSS system.



**Figure 4.5.2 – The Local Area Network scheme for the Rome DSS**

The architecture adopted allows for system scalability in case the computing power of any of the workstation would not be sufficient for the on-line operation of some processes.

### 4.5.3 Connections

The DSS is connected to external interfaces identified as :

- MMS, the source of information for the main traffic measurement. The systems runs on a LAN directly connected to the DSS LAN.
- UTOPIA, the source of information for complementary traffic data. The system runs on a LAN directly connected to the DSS LAM
- Environmental monitoring, the source of information for pollutants measurements and forecast given by a set of outstation. The DSS connects to the monitoring centre of the outstations to gather information. Such connection is currently planned based on manual operator intervention.
- ATMOSFERA model, the source of information for meteorological conditions and forecast. The DSS connects to the ATMOSFERA system. Such connection is currently planned based on manual operator intervention.

All the connections are based on TCP-IP interfaces making use of WAN technologies for the physical connections. Referring to the connections with Environment monitoring and ATMOSFERA a switched line connection using an ISDN line will be used.

### 4.5.4 Implementation constraints

The following constraints have been identified and will be considered in the specification phase of the implementation.

#### 4.5.4.1 Operating system

All processes will be developed in a Microsoft Windows NT-2000 environment.

#### 4.5.4.2 Network protocols

The basic network protocol adopted for communication within the LAN and with the external interfaces is TCP-IP.

If needed other network protocols can be used within the DSS LAN but not with the external connections.

#### 4.5.4.3 Inter-process communication

Inter process communication is based on socket connections using the TCP-IP protocol.

#### 4.5.4.4 Process synchronisation

All the workstation on the DSS-LAN will be time-synchronised on a common time server. Synchronisation with the external interfaced is devoted to the front/end processes.

The synchronisation between dependant processes is based on notification of availability of data for a specific time interval.

#### 4.5.4.5 Database engine

All shared data stores are stored on an ORACLE database server. Processes may use local database servers for local data stores.

#### 4.5.4.6 Topological reference

All processes uses a common topological reference to store and retrieve common data objects. Internally to the processes other reference can be used providing for a mapping from the internal to the common topological reference.

The common topological reference adopted for the DSS is the reference graph named as GdR (Grafo di Riferimento) currently used by the existing applications. Topological elements are geo-referenced using standard referencing systems.

### 4.5.5 Actions to implement

The actions identified to implement the DSS concern three different aspects:

- SW development
- HW provision
- Connections activation.

Such activities are described in the following paragraphs.

#### 4.5.5.1 SW development

The implementation for the DSS demonstrator has been split into several phases as described in the following. Each phase has been given a global deadline and thus activities related to the phase have been planned to cope with the overall phase deadline.

Main processes will be able either to run off line on a static set of data or on-line getting data in real time from the existing system.

The off-line capability will give the possibility to simulate the effects of different traffic scenarios changing the static set of data given as starting point. Thus it is the base for scenario simulation and assessment of impacts derived from different demand management actions.

##### Specification phase

The specification phase has the objective to draw a detailed specification of each process, data store and data flow identified by the overall architecture. The result of this phase is a full set of software specification for the development of software modules and the integration of the overall system.

Deadline: End of March 2001

##### Off-line development phase

The goal of this phase is to have all process in operation providing their results from static sets of data. This will enable to develop each process independently from the others and to carry out a preliminary technical assessment of each implemented function. The phase cover both the development of new software modules and the integration of existing packages into the DSS environment.

Deadline: September 2001

##### On-line development phase

The on-line phase will take care of the integration of different processes up to build the overall system running in real time. The result of this phase allows for an overall verification of the system. The verification could highlight needs for re-engineering or software modifications before the evaluation of the demonstrator.

Deadline: January 2002

The off-line and on-line phases are linked by a technical verification of the system and if needed by re-engineering actions on the system.

### ACTION SUMMARY

The following actions are identified:

Action	Responsible partner
<b>Specification activities</b>	
Processes specification	ELSAG/MIZAR/STA
Data stores specification	ELSAG/MIZAR/STA
Data flows specification	ELSAG/MIZAR/STA
HMI specification	STA/ELSAG
Scenario analysis	ELSAG/MIZAR/STA
<b>Development activities</b>	
P8 Static data import	STA
P6 Traffic frontend	MIZAR
P7 Meteo frontend	STA
P1 Traffic modelling	MIZAR/STA
P2 Pollution modelling	ELSAG
P5 HMI	ELSAG
P4 Scenario management	ELSAG/MIZAR/STA
P3 Scenario simulation	ELSAG/MIZAR/STA

#### 4.5.5.2 HW provision

The following actions are identified:

Action	Responsible partner
Hardware specification	STA
Hardware provision	STA
System and LAN configuration	STA

#### 4.5.5.3 Connection activation

The following actions are identified:

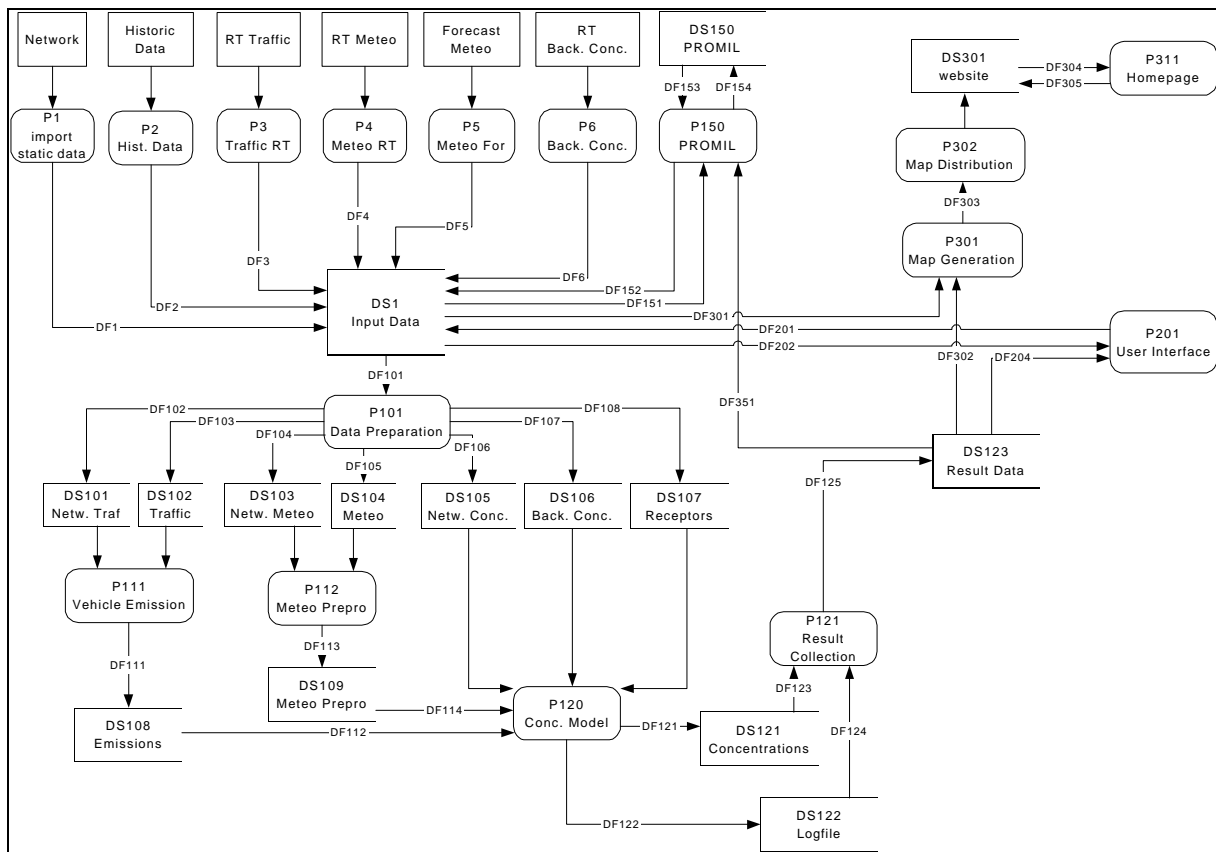
Action	Responsible partner
Provision of an ISDN line and router at the DSS system	STA
Organisation of interaction (currently planned as manual) with the Environment monitoring centre	STA
Organisation of interaction (currently planned as manual) with the ATMOSFERA system	STA

### 4.6 Physical Aspects of the Rotterdam DSS

The Heaven system in Rotterdam does not have a complex Traffic Modelling process. Input to the system are either real-time measurements or data from an external static traffic model (OMNITRANS). The system does neither take automated actions to influence the traffic based on measurements or forecasts of traffic, nor based on calculated pollution concentrations. Furthermore the system has no scenario development or evaluation functionality. Users will have to specify scenarios with combinations of input data to evaluate specific situations in order to develop policy.

Therefore the Heaven system in Rotterdam is quite simple compared to the other project sites, and all the processes can be identified in one diagram.

The following figure 4.6.1 shows the top level DFD diagram for such system and identifies the main processes, data stores and data flows.



**Figure 4.6.1 – The top level Data Flow Diagram of the Rotterdam DSS**

The system is built around two major data stores: DS1 - Input Data, and DS123 - Result Data. At one side interfaces and processes feed the Input Data with continuous data. When a complete set of data is collected environment modelling can be executed. The results from this modelling is eventually stored in Result Data. From this data store data is retrieved to present to users.

### 4.6.1 SW Components

The top level DFD identifies the main processes that implement the functions described in the functional architecture. The following table 4.6.1 summarises the main processes and sub-processes identified at the detailed specification stage. Annex A reports detailed tables including specification of requirements, data flows and mapping to the functions to be implemented.

Process	Description
P1 import static data	Module to import static network data. This is done initially to start the system and for major updates. Small changes can be made by the operator through his users interface (P201). In order to provide comparison of network variants multiple networks can be imported into the data store Input Data.
P2 historic data	Import historic data for all three sources of dynamic data: traffic, meteo and background concentrations. This data can be used for forecasts or if the appropriate RT data is unavailable. The import enables import of multiple sets of each data-kind, to distinguish types of day, season influences etc.
P3 RT traffic data	Acquire RT traffic data from Rotterdam dS+V. Traffic data is based on averages per hour. Data is downloaded using a dialup connection.
P4 RT meteo data	Acquire RT meteo data from KNMI (Royal Dutch Meteorological Institute). Meteo data is average per hour, measured on Rotterdam Airport. Data is downloaded using a dialup connection.
P5 forecast meteo data	Acquire meteo forecast (next day average) from KNMI. Data is downloaded using a dialup connection.
P6 RT background concentration	Acquire background concentration from DCMR (regional environmental protection authority). DCMR provides hourly averages from several measuring points in and around Rotterdam. Though both concentrations and the Heaven system are running in the DCMR office, data is downloaded using a dialup connection.
P101 Data Preparation	Generates a set of data stores that is required to perform environmental calculations. Based on directions this can be: <ul style="list-style-type: none"> <li>• “real-time”, using collected RT data</li> <li>• forecasting, using forecasted meteo and historic traffic + background concentrations</li> <li>• scenarios, using selected data for traffic, meteo and background concentrations</li> </ul> This process can combine data from different periods to built a specific scenario.
P111 Vehicle Emission	this is a pre-process for concentrations modelling. With the CAR2 model vehicle emissions are calculated for road sections, based on network and traffic data. Vehicle emissions are calculated based on: <ul style="list-style-type: none"> <li>• 3 vehicle type (3 types distinguished)</li> <li>• emission characteristics per vehicle type</li> <li>• average speed per vehicle type (hour based)</li> <li>• intensity per vehicle type (hour based)</li> </ul> The emission results are only used to feed the concentration model (P120) and therefore only stored temporary.
P112 Meteo Prepro	Influences from the weather are combined with area characteristics

Process	Description
	<p>("canyoning").</p> <p>Area characteristics are expressed using an indicator for terrain roughness. For each occurrence of this indicator the process is executed.</p>
P120 Concentration Model	<p>the main environmental modelling process. Using pre-processed data (vehicle emissions and meteo) and other data, pollution concentrations are calculated for specified measuring points: receptors. The process produces a log report and a file containing concentrations (<math>\mu\text{g}/\text{m}^3</math>) per receptor. Both files are ASCII format.</p> <p>The concentration model has to be executed for each pollutant that has to be modelled. The required calculation time for this part of the Heaven system is considerable, even on modern PC's. If required, the calculation per pollutant can be spread over multiple systems.</p>
P121 Result Collection	<p>this process stores the results from the concentration modelling into the second main data store DS123, as the results from the concentration model (DS121) are in an inefficient format</p>
P150 PROMIL	<p>This is an existing application, currently running standalone. The program is used for long term policy: to calculate yearly reports, to investigate scenarios and to develop strategy. It has its own user interface and data store. In Rotterdam this program is used to generate annual reports.</p> <p>PROMIL is connected to the Heaven system to be able to use the new concentration model (P120), to benefit from better emission / concentration models and from new pollutants (black smoke, <math>\text{PM}_{2.5}</math> and <math>\text{PM}_{10}</math>).</p>
P201 Key-User interface	<p>GIS-interface for local key-users in the Heaven project.</p> <p>Primary object for the GIS interface is to present data from the Heaven system graphically overlaid on maps from the Rotterdam area.</p> <p>Secondly GIS is used to produce reports.</p>
P301 Map Generation	<p>this process generates a map with concentrations that can be displayed on the internet site.</p> <p>The results from the concentration model (P120) are points with concentrations. These points however are isolated. GIS offers functionality to generate concentration bands based on values at points which result in clear representation of the concentrations.</p> <p>The map generation process generates a bitmap picture.</p>
P302 Map Distribution	<p>When new maps are available from the map generation process (P301) these maps need to be transported to the server of the web site to update the homepage.</p> <p>This process takes uploads the maps using a dialup connection.</p>
P311 Homepage	<p>Homepage for public for information about environmental situation. Currently foreseen functions:</p> <ul style="list-style-type: none"> <li>• information</li> <li>• link to other Heaven cities</li> <li>• maps current situation of: <math>\text{NO}_2</math>, <math>\text{PM}_{10}</math> and Benzene</li> <li>• maps forecast of : <math>\text{NO}_2</math>, <math>\text{PM}_{10}</math> and Benzene</li> <li>• instructions/advises</li> </ul>

**Table 4.6.1 – Processes and sub-processes of the Rotterdam DSS**

### 4.6.2 HW Components

Though most functionality is concentrated on one system (Heaven EIP system), several systems are required to complete the Heaven system in Rotterdam. The Heaven EIP system has its own console in order to enable direct maintenance.

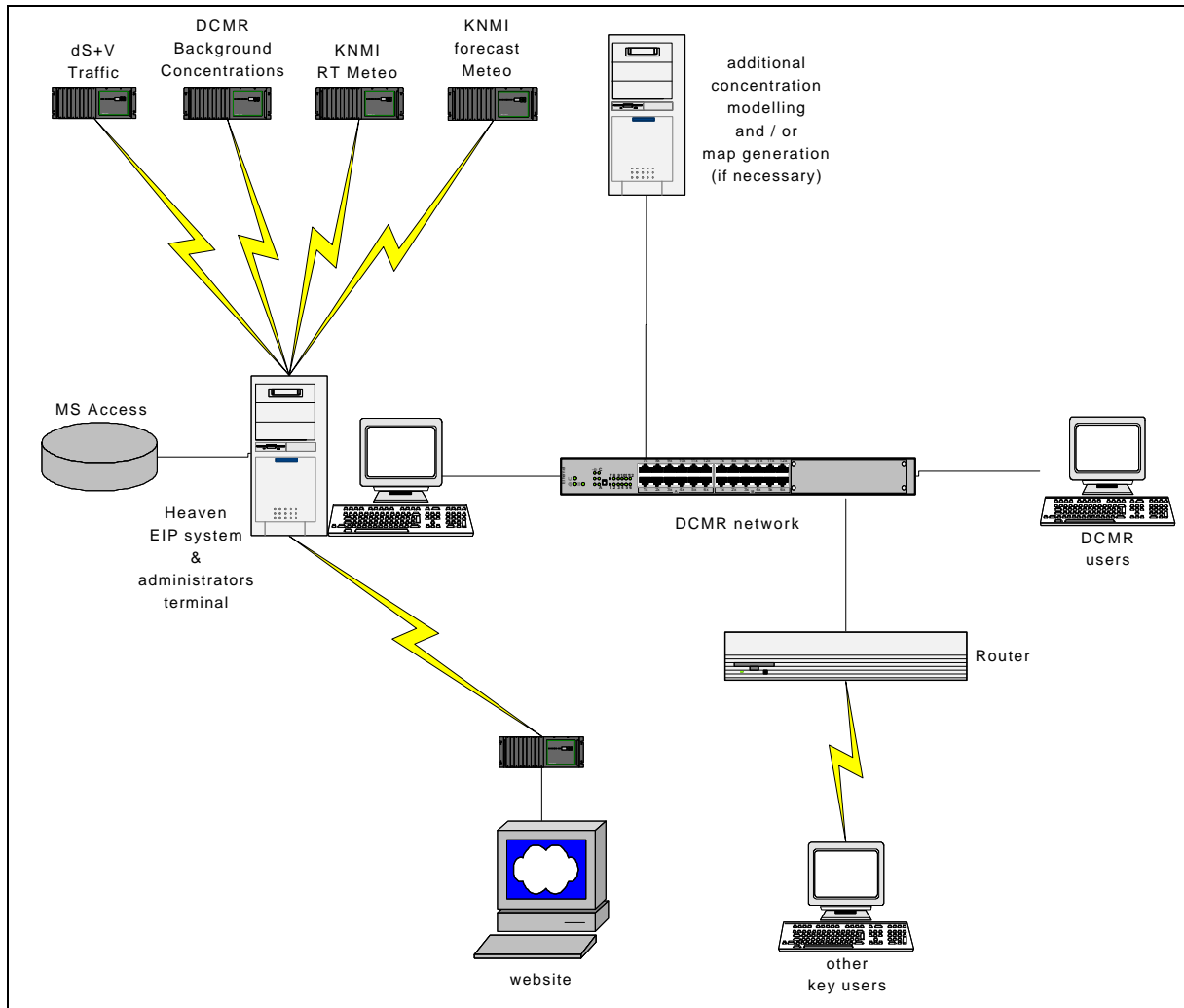


Figure 4.6.2 – Hardware components of the Rotterdam DSS

As the system is located at DCMR (regional environmental protection authority), it is connected to their local network. Through this network the main databases of the Heaven system (in data flow diagram: DS 1 Input Data, and DS123 Result Data) are accessible. The databases will be built in MS Access.

The input of real-time measurements and meteo forecast is done through dialup connections. The real-time background concentrations from DCMR are the latest to be available after completion of a hour: approximately 25 minutes. In the time between completion of an hour and those 25 minutes other sources can be approached and data downloaded. At this moment dialup connections are planned as the amount of data to be transported is relatively little and thus communication sessions are short.

Transport of background concentrations from DCMR itself is also done by dialup connection as this is an easier solution for the existing system than transport through the internal network.

If necessary extra calculating stations may be added to the system when environmental calculations take too much time. The architecture and the way the concentration model (process P120 in data flow diagram) work, enable it quite easily to spread calculation load over several systems. The same goes for the map generation.

Key users from DCMR and other local authorities have direct access to the Heaven system. DCMR users through the internal network. Outside users through a dialup connection to the DCMR network.

The DCMR is running its own web site. However it is not yet decided if the Heaven web site will be stored on this. If it is updated information (generated maps, data flow DF303 in the data flow diagram) can be transported over the DCMR network. Otherwise an extra dialup connection is required, as is shown in the network diagram:

### 4.6.3 Connections

The connections from the Heaven system to other systems and users are as following:

- connections to systems for dynamic data input are all dial-up lines using specific protocols
- connections to users (DCMR and other key-users) are routed through the DCMR network, based on Novell
- connection to the website is established through dial-up line using FTP (TCP/IP) protocol

### 4.6.4 Implementation constraints

The following constraints have been identified and will be considered in the specification phase of the implementation.

#### 4.6.4.1 Operating system

The EIP-server for the Heaven system will be running Windows 2000 Server operating system

#### 4.6.4.2 Network protocols

The adopted network protocols are:

- IPX – Novell Network
- FTP – file transfer to web site
- TCP/IP – connection to web site
- specific – dynamic data input

#### 4.6.4.3 Inter-process communication

Inter-process communication between the Heaven system and other systems is based on the mentioned network protocols.

Communication between processes running on the EIP-server is directly.

#### 4.6.4.4 Process synchronisation

Synchronisation between processes running on the EIP-server depends on availability of data. Processes notify each other directly.

Synchronisation between the EIP-server and dynamic data inputs (traffic, meteo and background concentrations) is responsibility to the front-end process.

#### 4.6.4.5 Database engine

All shared data will be stored using a Microsoft Access database server.

DCMR does have an Oracle database server, used for other critical processes. Within the Heaven project it is not feasible to extend this server.

### 4.6.5 Actions to implement

The actions identified to implement the DSS concern three different aspects:

- SW development
- HW provision
- Connections activation.

Such activities are described in the following paragraphs. The actions lead to an operational system by the third quarter of 2001.

#### 4.6.5.1 SW development

The implementation for the DSS demonstrator has been split into several phases as described in the following. Each phase has been given a global deadline and thus activities related to the phase have been planned to cope with the overall phase deadline.

The phases are made up by several activities. This implies that even though the nature of the phases is sequential, activities of a next phase can already start - even when the previous phase has not been fully completed. Thus creating overlap.

##### **Specification phase**

The specification phase has the objective to draw a detailed specification of each process, data store and data flow identified by the overall architecture. The result of this phase is a full set of software specification for the development of software modules and the integration of the overall system.

Deadline: May 15<sup>th</sup> 2001

##### **Equipment provision phase**

The equipment (hardware, operating system and database) on which the EIP-server will run has to be ordered and delivered in a quite early stage.

Deadline: May 15<sup>th</sup> 2001

##### **Off-line/On-line development phase**

During the off-line phase individual processes are developed and tested running their part of the system. Functionality is tested using static data, is the process that should prepare data is not yet available special tools will be programmed to prepare the data to run a process and/or to process its output.

In the on-line phase different processes are integrated into a overall real-time system. By the end of this phase the system is ready for verification.

As shown in the project level Gantt chart these two phases are combined, but a division has been made between processes up to concentration modelling and processes that output results.

**Concentration modelling development phase**

This phase combines the processes that are required for input, preparing and processing data to perform concentration modelling. The order of development of these processes is as described above (first off-line, then on-line). The result of this phase is the ability to run the system to collect data, prepare data stores and to execute concentration modelling.

Correct functionality of the modules can only be determined by investigating the data in the data stores (with special tools).

Deadline: End of June 2001

**Output results development phase**

In this phase the processes that present results from the Heaven system are developed. These processes use data that has been collected from dynamic input and results from concentration modelling. Correct functionality of the modules can be determined using the regular interfaces: administrator, key users and web site. At the completion of this phase the validation can start.

Deadline: End of September 2001

**ACTION SUMMARY**

The following actions are identified:

Action	Responsible partner
<i>Specification activities</i>	
Processes specification	GC/TNO
Data stores specification	DCMR/GC/TNO
Data flows specification	GC
HMI specification	Stuurgroep
<i>Development activities</i>	
P1-P6 Data import	GC
P101 Data preparation	GC
P111-P120 Emission and concentration modelling	TNO
P121 Result collection	GC
P150 PROMIL	GC
P301-P302 Maps for website	GC
P311 Website	GC

**4.6.5.2 Equipment provision**

The following actions are identified:

Action	Responsible partner
Hardware specification	DCMR/GC
Hardware provision	DCMR
Operating system specification	DCMR/GC
Operating system provision	DCMR
Database server specification	DCMR/GC

Database server provision	DCMR
System and LAN configuration	DCMR

#### 4.6.5.3 Connection activation

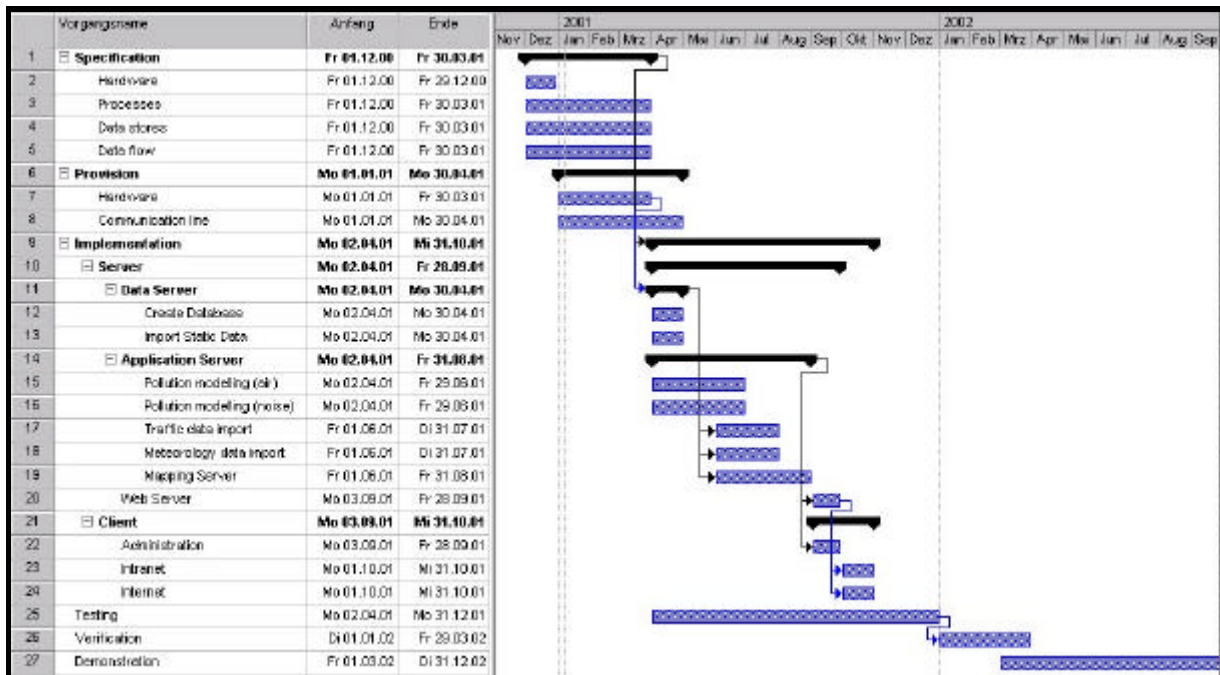
The following actions are identified:

Action	Responsible partner
Provision dynamic data inputs	DCMR
Provision internet-connection to web site	DCMR
Provision of connection between Heaven system and key users	DCMR

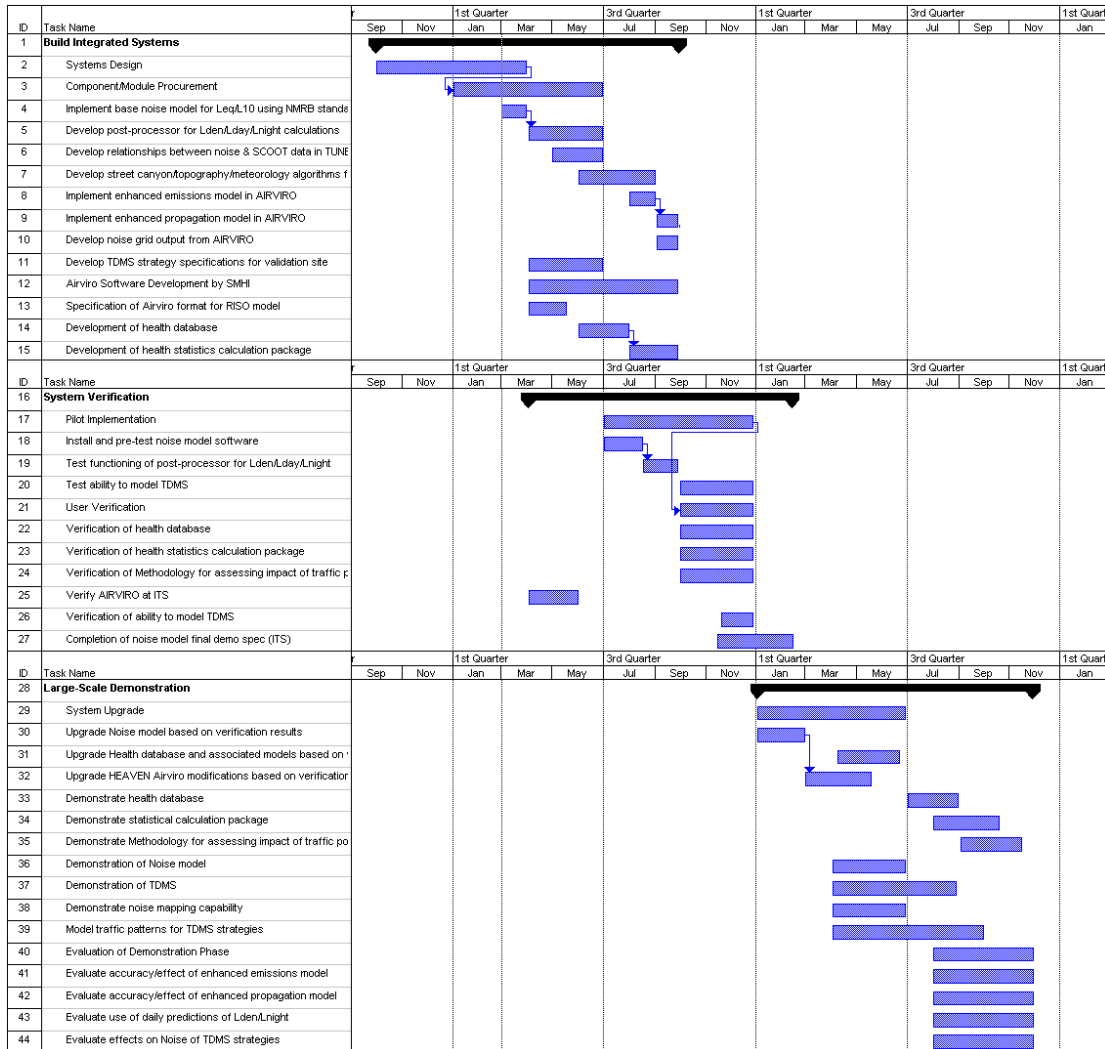
## 5 OVERALL PROJECT PLANNING

The following paragraphs report the GANTT charts elaborated by the project sites to achieve the local DSS implementation and demonstration phases.

### 5.1 Berlin GANTT chart



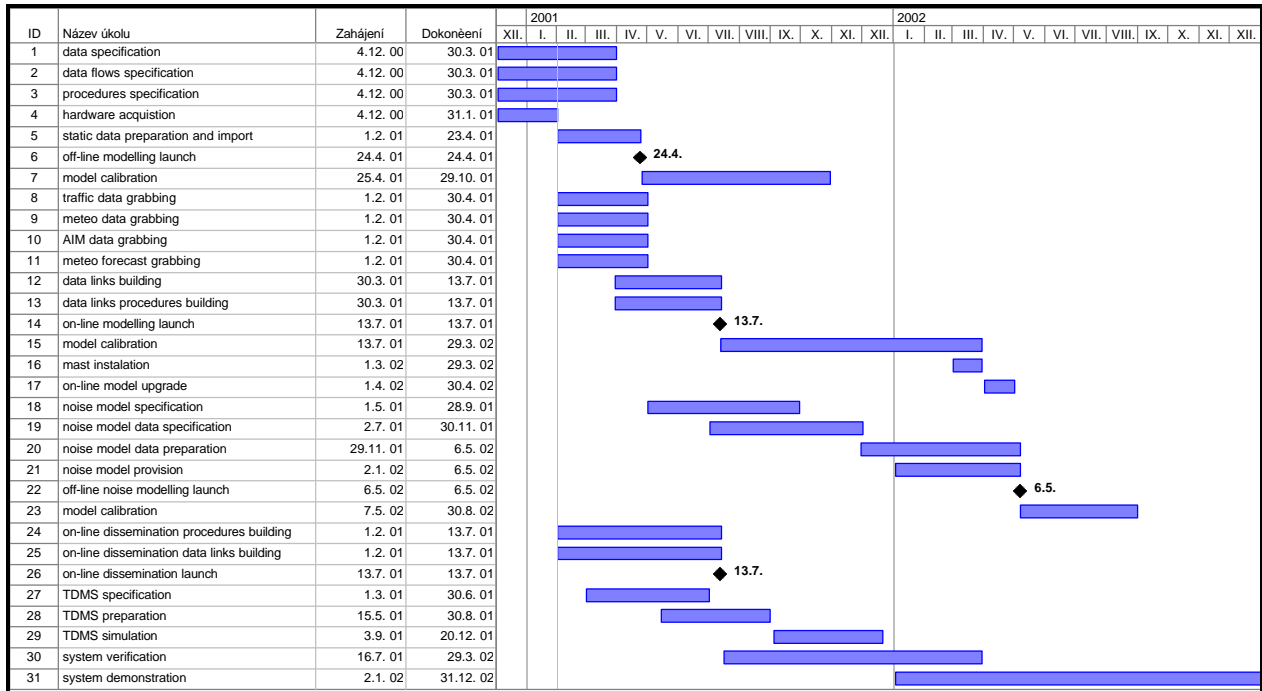
## 5.2 Leicester GANTT Chart



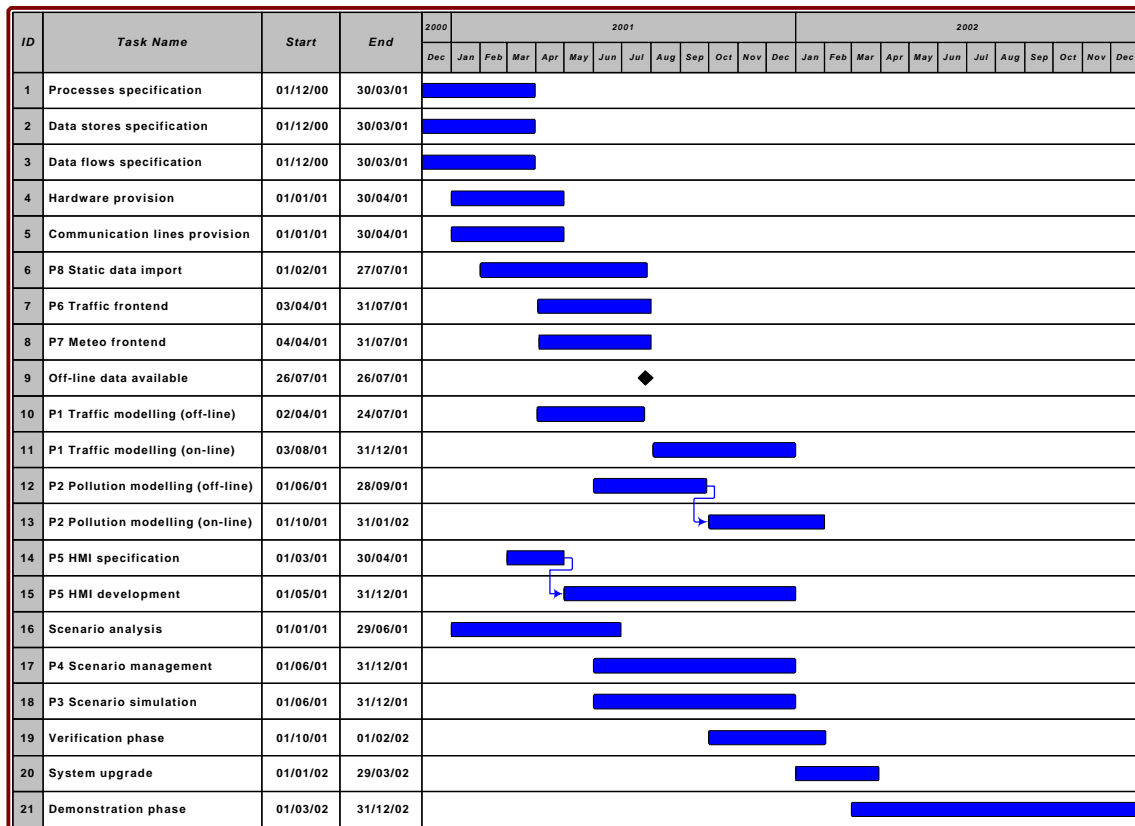
### 5.3 Paris GANTT Chart



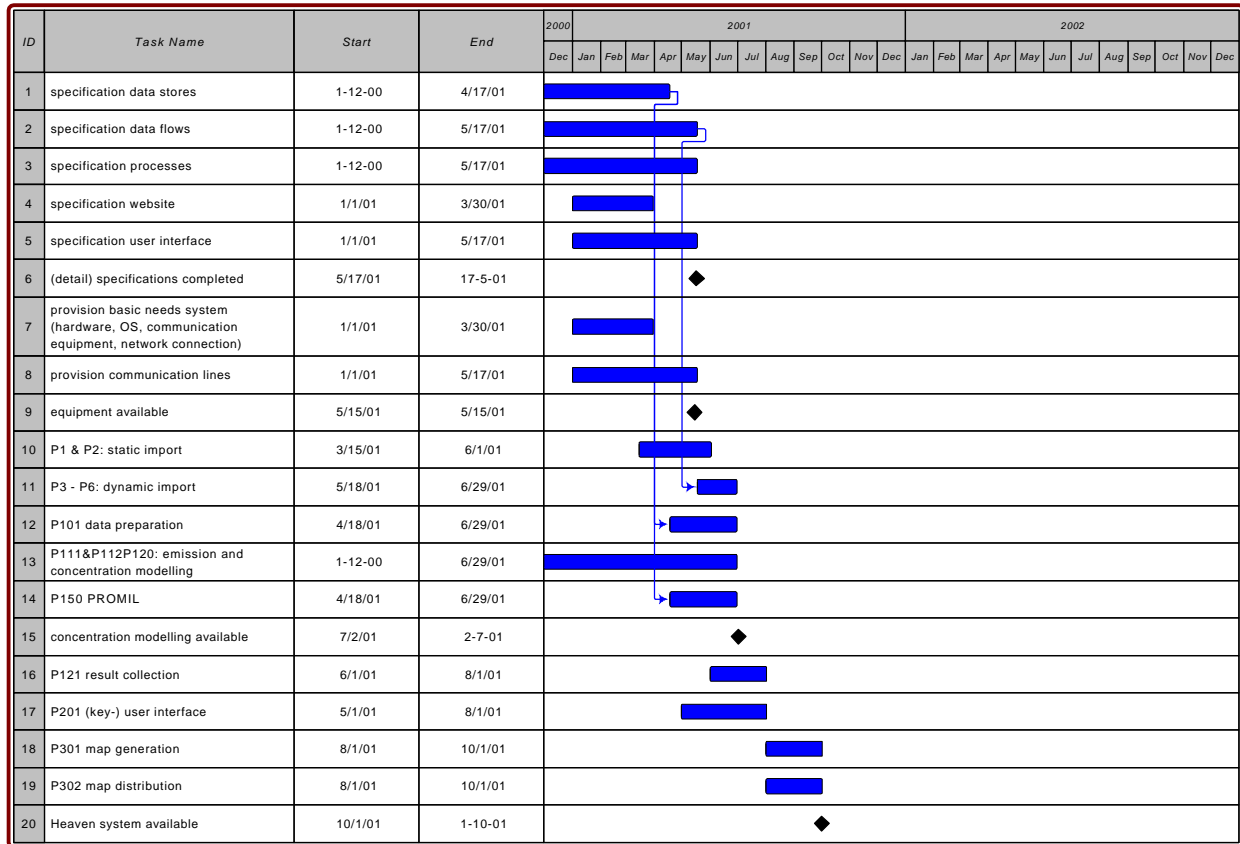
### 5.4 Prague GANTT Chart



## 5.5 Rome GANTT Chart



## 5.6 Rotterdam GANTT Chart



## 6 LESSONS LEARNED

The difficulties encountered in WP5 lay on different levels:

- i. The compilation of homogeneous formal descriptions of the DSS demonstrators planned in the sites.
- ii. The design of the common HEAVEN DSS Overall Architecture.
- iii. The design of local DSS demonstrators consistent with the HEAVEN DSS concept.

Key decisions were taken by the Project as a whole and by WP5 in particular that resulted successful for facing these difficulties:

- The conduction of a deep and sound User Needs Analysis involving Authorities and technical bodies in the project sites and in other European cities. The analyses performed in the context of WP4 [3] provided the detailed information needed to define system requirements for both the HEAVEN DSS and the local DSS demonstrators and formed the basis for their architecture design. Moreover this information will be useful for the subsequent phases concerned with system evaluation.
- The adoption of a tested and rigorous methodology to design and to describe both the common Overall System Architecture and the local DSS architectures. The CONVERGE guidelines [2] selected by WP5 led the design activities through fundamental steps, namely
  - the description of the System Concept
  - the definition of the System Characteristics
  - the definition of the System Requirements
  - the design of the Functional and Information Architectures
  - then design of the Communication and Physical Architectures
  - then detailed specification of the system components

that on one side required an effort to all the sites to formalise and to provide complete information expected, but on the other side guaranteed completeness and consistency of system characteristics and architecture descriptions. These descriptions allowed the identification of the peculiarities of each local DSS and highlighted key aspects to be maintained as fundamental characteristics of the HEAVEN DSS general concept.

- The decision to follow a top-down process to design at first the HEAVEN DSS Overall System Architecture and then the local DSS architectures. This approach brought to the design of an Overall Architecture consistent with the HEAVEN DSS concept suitable to be used as reference to design the local DSS architectures guaranteeing compatibility of the demonstrators at the higher level while allowing customisation according to the site peculiarities.

## 7 REFERENCES

- [1] IST-1999-11244 – HEAVEN Deliverable D5.1 – Environment Monitoring and DSS Architecture – C. Di Taranto et al. 15/01/2001
- [2] TR 1101 - CONVERGE Deliverable DSA 2.2 – Enhanced Architecture Guidelines – JF Gaillet et al. 10/10/96
- [3] IST 1999 11244 – HEAVEN Deliverable D4.1 - User Requirement report – C. Pykonen et al. 24/11/2000