DNMI DET NORSKE METEOROLOGISKE INSTITUTT

klima

KLIBAS - THE DNMI CLIMATOLOGICAL DATABASE SYSTEM

MARGARETH MOE

REPORT NO. 22/95 KLIMA

DNMI - RAPPORT

DET NORSKE METEOROLOGISKE INSTITUTT
POSTBOKS 43 BLINDERN 0313 OSLO

TELEPHONE: 22 96 30 00

ISSN
0805-9918

REPORT NO.
22/95 KLIMA

DATE
31.05.1995

TITLE

KLIBAS - THE DNMI CLIMATOLOGICAL DATABASE SYSTEM

PREPARED BY

Margareth Moe

ORDERED BY

DNMI

SUMMARY

The report describes the climatological database established at the Norwegian Meteorological Institute (DNMI) during the years 1990 to 1995. The report also access the evaluation process choosing computer and commercial RDBMS as well as feasibility studies and database-design.

The solution implemented at DNMI is based on a Silicon Graphic «Power series» (multiprocessor) computer holding an ORACLE7 database (RDBMS), ORACLE tools (SQL*FORMS, SQL*MENU, PL*SQL, SQL*PLUS, PRO*C, PRO*FORTRAN) and 3. generation program-languages as C and FORTRAN.

Pr. 1.5.95, KLIBAS is holding data and run in production for Precipitation stations, while data for Weather stations, Automatic Weather stations, Maritime stations, Upper Air Sounding stations and some other types of stations are being loaded. Quality-control routines are developed for Precipitation stations and some for Automatic stations. Some applications (reports) for selection of data from Precipitation stations are developed. Metadata is also stored and updated in the database. Further work will access development of quality control routines, general reports and putting routines for Weather, Maritime, Upper Air Sounding and other data into production.

When historical data has been loaded, the size of the climatological database will be of approximately 12 Gigabyte, with a yearly increase of about 400 Megabyte pr. year.

SIGNATURE

Margareth Moe	Bjørn Aune
RESEARCH SCIENTIST	HEAD OF DIVISION

KLIBAS - THE DNMI CLIMATOLOGICAL DATABASE SYSTEM

1.0 INTRODUCTION

In this report we will show how the datasystem KLIBAS (including both the climatological database, and applications) at DNMI has come true. It has been a long and quite thorny way to go from the whole idea was announced in 1989 till today when nearly all data and station-types are loaded into the database (though yet, only Precipitation stations are running in «production»).

This report will besides giving a description of the implemented system, try to show how and why the system has become as it is, with special emphasis to actual problems and gained experience during the work. We will also give a brief description of the project background, time schedules and performed feasibility-studies.

The work with KLIBAS has involved several people in the Climatological Division during the last 5 years. Specially in the initialisation phase and feasibility studies, many people was involved, setting up the system requirement specification. Later a smaller group of 5 people has developed the system during the last 2 ½ year (2-3 man-labour year each year), including database design, application development and database administration, besides handling current duties.

Participants in the project group at the Climatological Division are:

Per Ove Kjensli (senior research scientist) with special responsibility for / knowledge of the implemented system for Precipitation stations and application programming (PL/SQL).
Margareth Moe (research scientist) with special responsibility for / knowledge of SQL*FORMS, SQL*MENU, application programming (PL/SQL), project co-ordination.
Åse Moen Vidal (executive officer, responsible for EDB routines) with special responsibility for / knowledge of the RDBMS configuration, DBA tasks, SQL*FORMS and application programming (PRO*FORTRAN).
Petter Øgland (research scientist) with special responsibility for / knowledge of control-algorithms and designing control routines and application programming (PRO*C and UNIX programming).
Tom Aasen (research scientist) with special responsibility for / knowledge of client / server configuration (SQL*NET), application programming (PRO*C and PL/SQL).

Chapter 1.1 gives a historical overview of the project from it was started in 1989 and up to day.

Chapter 2 gives a short description of the old data-system at DNMI and the motivation of converting to an other system. The feasibility study and requirements to the new system will also be accessed.

Chapter 3 shows how we worked with the information-model of the database and which aspects were taken to be determinate for the database design.

Chapter 4 shows the realisation of the system, that is the configuration of the system (both hardware and software), table-structure, how current data is loaded and controlled and how data is selected from the database and reports produced.

Chapter 5 gives a summary of gained experience and critical factors.

Appendix A gives an overview of all reports written in the project. The main part of these are written in Norwegian, but in the list, we have tried to give every report a title in English, to show which topics are accessed in the report. If interested in any of the reports, please contact us, and we will try to give a summary of the report in English and of cause we would be glad to answer or discuss questions with you!

1.1 An overview of the database project.

Late in 1989 DNMI initialised the work which should result in buying new computer(s) / database-system in replacement of existing systems for both real-time-weather-data and climatological data. A group with representatives from all divisions involved was established. Detailed requirements for the computer and database-system was outlined. At the same time it was essential for the group-members to update and achieve new knowledge both in computer technology and database-management-systems.

It was early clear that implementation of the system had to be done within available resources. At the Climatological Division a group of approximately 8 persons (now reduced to 5) was established, first to accomplish the requirements for climatological data, and later to design and implement the system. This group is later in this report named as «the database-group in the Climatological Division» or simply as the «database-group» or «we».

Requirements were sent potential vendors late in 1990. Evaluations and discussions of offered systems took place during the spring 1991, and the decision was made early summer 1991. Education in the operating system and database took place second half of 1991.

The work of the database-group was organised as a project, giving goals, estimations, work-packages, activities and schedules. These plans were followed, specially in the first phases of the project, and made us take time in the design and specification phases. Later in the implementation phase, as unpredictable problems and activities have arised, a lot of the work done is therefore based on experiments performed successively.

In 1992 we started the design of the climatological database, involving gaining experience from experimenting, literature studies, analyses of different use of the data and discussions with database experts. This work ended up in a specification of the system regarding database-design (KLIBAS 42/92) late in 1992, followed by specifications for «loading data», «controlling data» and «data-reports». In fact even now, 2 ½ years after this detailed design was made, quite few revisions to the specifications have shown necessary.

In 1993 we started loading Precipitation-data (approximately 700 stations or about 1.5 GB of data) into the database, and routines for loading (manually), controlling (logical and spatial-control) and selecting data were made and set into production. Tthis was a typical 1. version of the system, and a lot of improvements and new applications had to be developed.

In 1994 the system for Precipitation data was further developed, and a lot of the applications were reprogrammed, using other programming-tools to achieve better functionality and response time. A system for collecting synop-data and data from Automatic Weather stations as they are received at DNMI was established. The conversion from ORACLE6 to ORACLE7 was also done during the autumn 1994.

In 1995 about 12 GB disk was configured for the database, and loading historical data for the other station-types started. This work is estimated to be finished during the spring 1995, and routines (loading new data, controlling new data and report-applications) for these station-types data will then be developed.

2.0 PROJECT BACKGROUND AND FEASIBILITY STUDY.

The goal of the project started late in 1989 was to buy a new computer and a commercial database-system to replace the existing systems. It was given that this new system should undertake both activities necessary for real-time data, such as the field-data, models and coding / decoding routines, and climatological data, such as storing of long time-series, routine handling and reports on dataseries and statistics.

The system in use at this time, was composed of several Norsk Data (ND) - computers (which has been in use for 15 years) handling most of the climatological «database» and the field-data, PC-based climatological «databases» and an IBM mainframe which was handling weather models and research activities. New technology giving much faster computers with great storage capacity besides requirements for open solutions (not based on proprietary solutions) and standardisation were main arguments for buying new computer and database-system.

General requirements to a new data-system were obvious, i.e. the system had to be fast, able to store large amounts (several Gigabyte) of data on-line, able to handle quite different types of jobs effectively (as field-processing and database running), use standard operating system and have a good graphical interface.

Among the benefits from the system are huge on-line data-storage capacity, ability to use different sort of data together, easier and standardised data-access, simplified operator routines and powerful model and presentation facilities.

2.1 Specification and evaluation of new computer and database.

Having the above requirements in mind, the work with setting up a specification started. To understand the technology and to be able to set up an appropriate specification including all necessary parts, education and updating both in data-computer technology and database-systems was necessary. Besides we needed to get as complete descriptions of the old system and the jobs executed here, as possible.

This took about ¾ year, but in late 1990, the specification document was finished and sent potential suppliers. It was then clear that the solution should include two quite identical computers, most likely based on UNIX operating system, and a Relational Database Management System.

The database-group in the Climatological Division had been established as early as in 1990, though it was first in 1992 the real project planning the database and the system was started. The first thing to do was, as a part of the specification report, to examine our existing system to be able to write a description of all the data-types, how they were treated and the amount of data stored. We also tried to see into the see into the future according to future demands, increase in data and necessary equipment / tools.

Having received and evaluated / discussed the different answers with the suppliers, during the spring 1991, it was decided to buy two Silicon Graphics' Power Serie computers (multiprocessor computers) using UNIX and Motif and as for database we decided to go for ORACLE.

Important elements in the evaluation were fast processors, multiprocessor capability, storage capacity, short shut-down time, possibility to easily switch between the two computers in an emergency situation, standardisation, price and support. According to the database, choosing a Silicon Graphics computer, left out some database-suppliers whose databases were not running on SG. However, there were still suppliers fulfilling the requirements. In our choise of ORACLE, standardisation, national support, reputation of the company, symmetric multiprocessing capability and spectre of application and querying tools were the most important factors.

The price of ORACLE on the SG computer was approximately ¾ million NOK, including database tools as given in chapter 4.1. As it became necessary, client versions («ORACLE office», or now «CDE tools») for PC were bought (approximately 4.000 NOK each). In addition to this «one-time cost», a service-agreement (including free telephone support, some consultant support and upgrading / new versions of the database and tools) to the cost of approximately 140.000 NOK a year is paid.

After having installed the computers and a test-database in summer 1991, the rest of the year went taking part in different training courses (both in UNIX, the database-system and some ORACLE tools) and experimenting on the test-database.

3.0 INFORMATION MODELLING AND DATABASE DESIGN.

Aware of the importance of finding an optimal design, we used a lot of time discussing the database design. This is especially of great importance when storing large amounts of data. We also worked a lot making an information model (defining entities and relations) for the information archive (meta-data).

3.1 Analysis of data-flow and use of data.

When evaluating different structures for the meteorological database, we analysed the amount of data for each station, the station parameters, in which way and intervals data from the different stations are received by DNMI, loaded into the database and controlled and last but maybe the most important factor, how data is used (for scientific research, in statistics, standard reports and ad-hoc queries).

From these investigations we found there were two basic differences in use or rather in the way we work with data.

New data are frequently controlled as more data is coming into the database. There are different controls to be performed in sequence, i.e. «real time controls» or checks as data is loaded the first time, comparisons between parameters in one observation, comparisons across observations and controls in space across stations. With basis in the controls, data is eventually corrected and updated. These operations involves only new data and often data from many different stations is needed / handled at the same time.

Old data or historical data is normally only accessed (selected), that is, no controls are routinely performed. When accessing data, the investigation in our division showed that in most cases, only data from one, or a few stations were extracted at a time, involving long time-series. These operations involves all data (new and old), but only one or a few stations at a time. The response time is often critical (i.e. when answering questions by phone).

3.2 Database design - meteorological data.

The two fundamentally different ways in using data show that there may not be one data-structure giving optimal response-time in both cases. One alternative might be to store **all** data together in one table. This **table** would however contain billions of rows which is far beyond the scope of any database-table.

Because of different reporting routines for different station-types, data from some station-types is loaded nearly in real-time, while other station-type-data is loaded once a week, once a month or even half a year after observed.

After deep considerations and discussions with database-experts, we found it probably most efficient having different types of tables for the most recent data (new data still to be controlled and updated) and old data (historical data only selected from the database). The historical data is voluminous, but only accessed data, while the most recent data is frequently used in comparisons and updates. This distinct use of data approve different structures for the two kinds of data.

We found the most efficient structure for historical data to be separate tables for each station holding all observations of the station. Anyhow, we found that there would be great advantages creating identical tables (having the same attributes or columns) for each station-type (we call these tables or rather part of the database «the historical storage» or HL).

For new data, we found it most efficient having **one** table pr. station-type holding all observations for all stations (a «working table» or AL for each station-type). When the data controls are finished, the data will be moved to the HL-tables and deleted in the AL-table.

The most recent data for each station-type is first loaded into one table, holding all the stations of the actual station-type. This means there is only one table holding the most recent data for each station type, which is most efficient using stations together (as for instance in a spatial-control). In the start we tried to update the historical tables as often as possible (observations not yet controlled were flagged) so selection of data only was necessary from one table. As the database expand, these update-jobs most likely are to take too much time to be effectuated in «real-time», with the consequences of «night-jobs» and delayed access to the most recent data. To assure access to **all data** and to avoid frequent very resource-intensive jobs, we are now going to access both the tables where the most recent data is stored and the historical tables when selecting data. The most recent data which is not yet completely controlled will exist only in the table for the most recent data. First when completely controlled, they will be inserted into HL and deleted from AL.

Besides AL and HL we talk about the information tables as IL. The IL-tables holds information about the stations, their parameters, equipment, the observers and so on. IL is very important in the system and the system is actively using this information when inserting new observations or stations, updating information and extracting information from the database.

AL, HL and IL is only logical and practical names when talking about the different types of data in the database. They are not **objects** in the database.

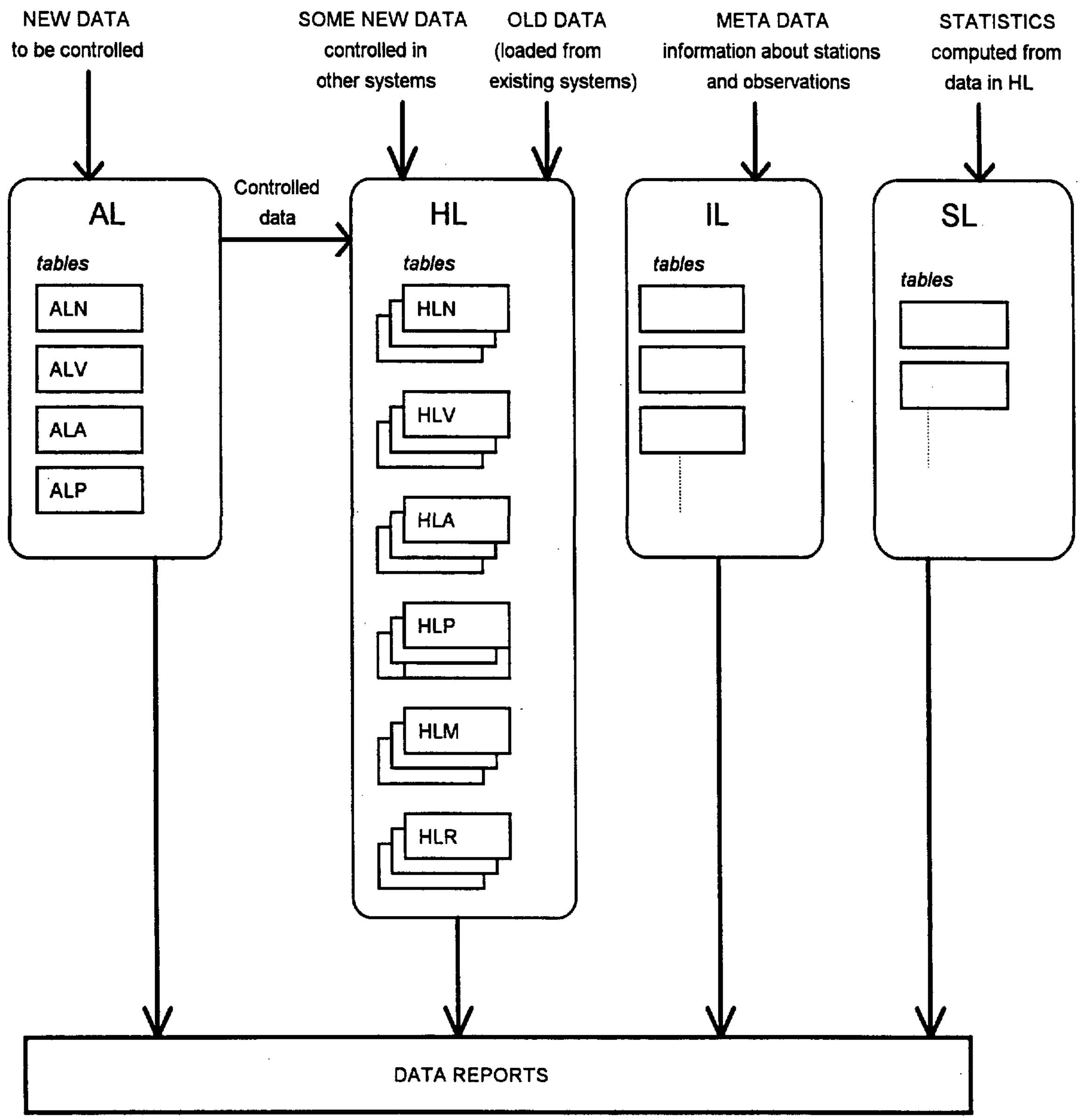

Fig. 3.2.1 Conceptual view of AL, HL and IL and the dataflow.

3.3 Information model of the information data (meta-data).

In the new system, we wished to store not only the observations themselves, but also information about the stations and the observations (IL-tables). Here we can take full advantage of the nature of a relational database with entities and relations. A information model was developed and in some degree normalised, though later it has been denormalised to achieve a more efficient structure.

At present time some parts of IL is implemented, while there still is a great job getting the information complete according to historical events since a lot of this information is not yet digitized. Anyhow we have tried implementing sufficient information to assure that only real stations and parameters are loaded, and enough information to ensure correct statistical presentations. The IL-archive is actively used both when generating tables, loading data, controlling data, extracting data and in administration of the stations. Most of the tables contain station number, station type, and starting date / ending date to ensure historical information.

Other tables to be implemented are ST_HISTORIE (station history) and PARA_KODE (code used for actual parameters).

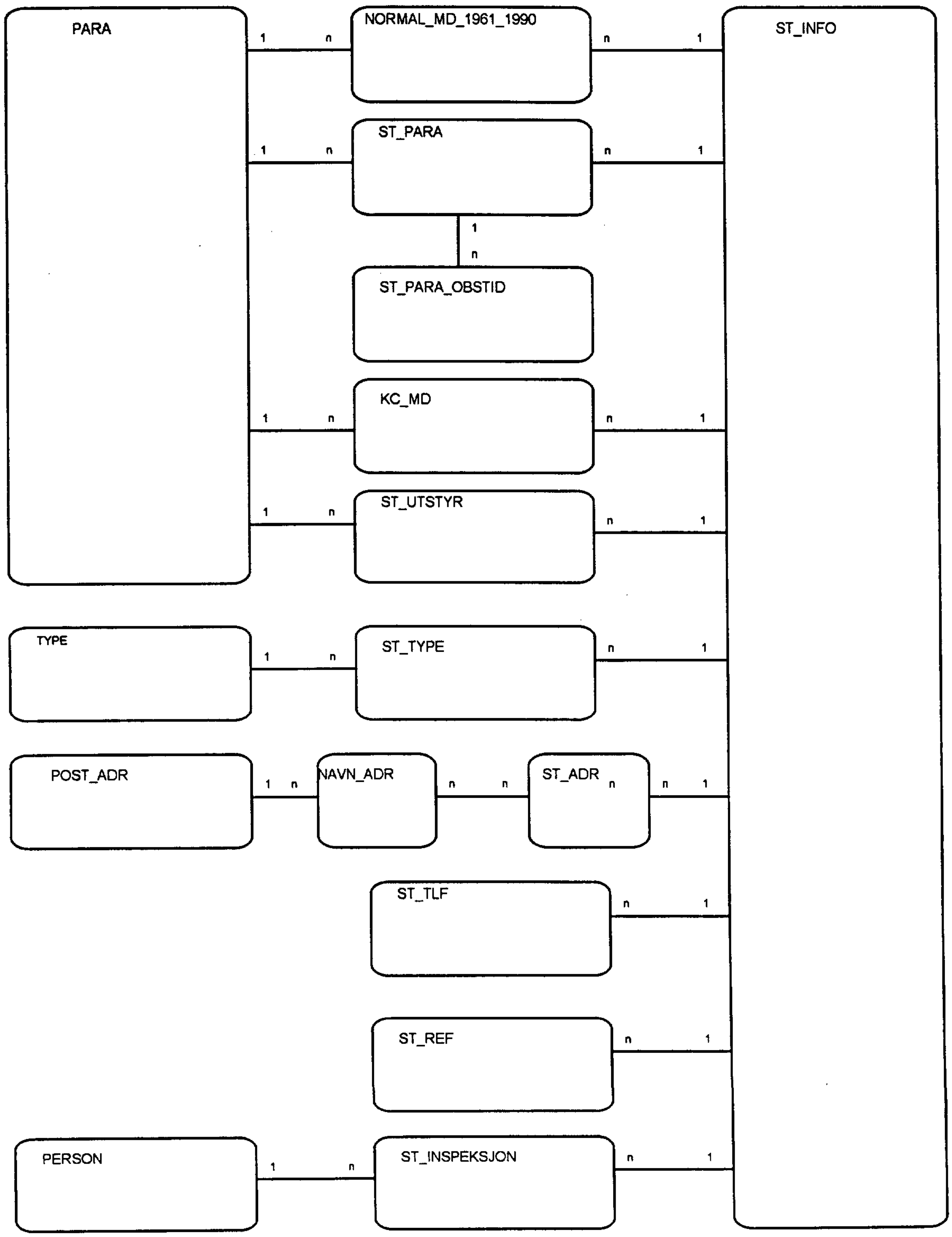

Fig. 3.3.1 Information model of IL pr. 10.5.95

4.0 KLIBAS - A CLIMATOLOGICAL DATABASE SYSTEM.

After 1 ½ year to specify and buy the database, ½ a year of training and experimenting and another year of planning the system (database-design, information modelling, establishing quality control routines, writing specifications for loading, controlling and reporting applications and routines) it was with both excitement and anticipation we started the implementation work in 1993.

Up to this we had worked with all the station-types at the same time, trying to get a picture as complete as possible of the whole database. In 1993 we started the work implementing the Precipitation station-data into the database. A work trying to establish general applications for data-reports was initialised. This showed to be quite complex and according to time-schedules for the Precipitation stations, we decided to stop this work and develop special applications for the Precipitation stations instead.

During the first ¾ year of 1993 historical Precipitation station-data was loaded into ORACLE. Forms for punching / updating data, applications controlling data and some applications calculating and presenting data were developed. The system was put into production during the autumn 1993. Anyhow, the system was not complete and about half of 1994 went to improve and further develop the system for Precipitation data. Late in 1994 ORACLE7 was installed and the work with all the other station-types (Synoptic-data, Automatic Weather Station-data, Maritime-data (sea-synop), Upper Air Sounding-data, Radiation-data and data from other types of stations) was started.

During the first half of 1995 all old data is copied out from our old computers and right now this data is being loaded into the database. The amount of data in the database when loaded will be about 12 Gigabyte (including system-space and indexes).

Data from some station-types are being loaded automatically from files on other UNIX computers or from PC's, while other are manually inserted (punched into forms). forms, control-programs and applications are developed.

The work with developing a general system for data-reports has started up again, though with some less ambitions than earlier. A user-group is established to give advises, rules and specifications in this work. This has shown very efficient and useful, and hopefully contribute to a system having the «right» applications giving «correct» data.

Probably all data will be loaded during the summer 1995 and routines and applications established within 1996. Even if there still is a long way to go and a lot of problems to be solved, we believe this will be overcome and make KLIBAS come true as the climatological database-system at DNMI within near future.

year:	1990		1991		1992		1993		1994		1995		1996	
activity — *1/2 year:*	1	2	1	2	1	2	1	2	1	2	1	2	1	2
Requirements for computer and database	■	■	■											
Training / experimenting				■										
Databasedesign, Informationmodelling, preparing work					■	■								
Specifying routines / applications						■	■							
Implementing Precipitation stations								■	■	■				
Experimenting loading/controlling Synops and Automates									■	■	■			
Loading historical data from other stations										■	■			
General / special data-report applications										■	■	■	■	■
Implementing routines and applications, all station-types											■	■	■	■

Fig. 4.0.1 Time schedule for the work.

4.1 Configuration of the system and software tools.

The relational database management system (RDBMS) is installed on a Silicon Graphics UNIX server (Typhoon) having two processors and about 20 Gigabytes of disk. Typhoon is connected to several other UNIX computers (both servers and work-stations) and PC's through TCP/IP and FTP. The PC's are connected to each other in a NOVEL netware.

Besides the ORACLE7 32-user database residing at Typhoon, different Oracle tools are installed: ORACLE Transaction Processing Option, ORACLE FORMS4 (and prior versions SQL*FORMS 3.0 and SQL*MENU 5.0), ORACLE's 4.GL; PL/SQL, ORACLE's extensions to SQL; SQL*PLUS, ORACLE's tool for loading data into the database; SQL*LOADER, ORACLE's precompilers (PRO*C and PRO*FORTRAN) for the 3. generation languages C and FORTRAN, ORACLE's CASE-tools (early versions).

ORACLE tools (FORMS4, REPORTWRITER and earlier versions SQL*FORMS 3.0 and SQL*MENU 5.0), PL/SQL, SQL*PLUS and SQL*LOADER are also installed at the PC's (CDE packages). The PC's are connected to ORACLE via SQL*NET.

As the system is implemented today, most applications having an user-interface running from PC's and there is developed an interface using Oracle's tools in the MS WINDOWS interface. Only forms are run as real client-server applications, while the other applications are started from the PC (by the RSH command in FTP), but executed on the UNIX-server as precompiled C programs. This is done due to requirements to response time and data-transfer through the network.

Some applications are implemented in PL/SQL only, and some applications are both started, i.e. by a CRON-job and ran directly on the UNIX-server (especially those not manually initialised)

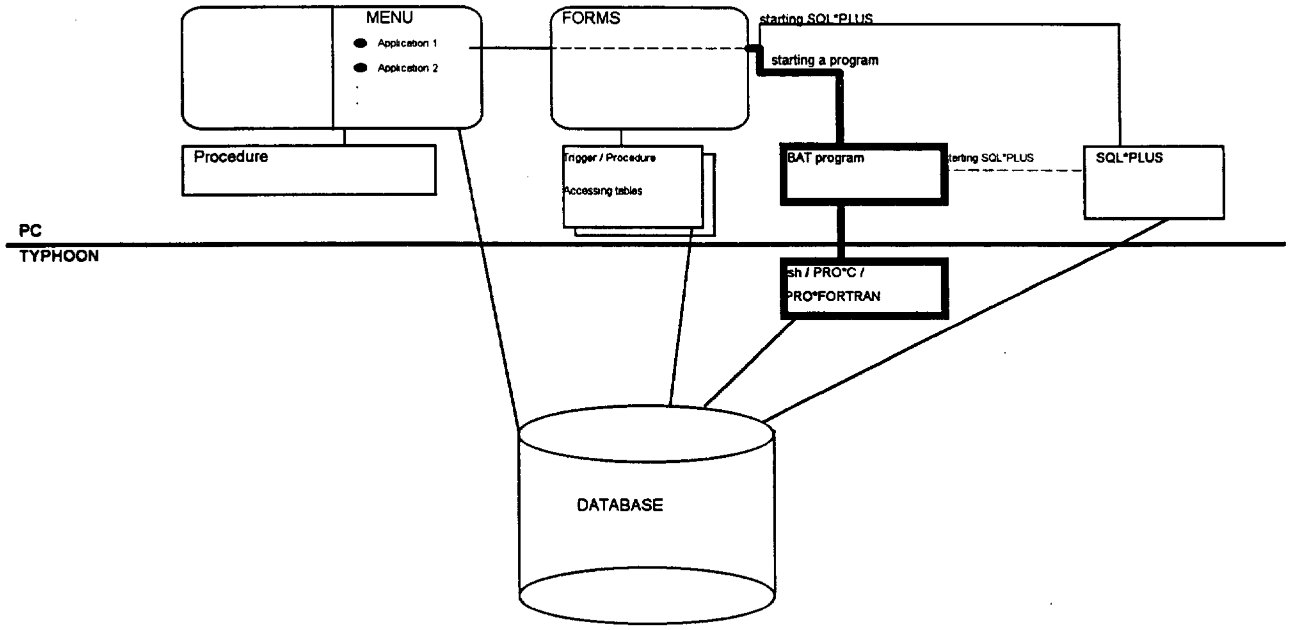

Fig. 4.1.1 Configuration of the system. The bold line shows the most used way starting an application.

4.2 KLIMAPRO - the production database.

KLIMAPRO is the observational-database holding all data and information (HL, AL, IL) for all stations. The database is of approximately 12 Gigabyte, increasing by about 300 - 400 Megabyte each year. A conceptual design of the database is given in fig. 3.2.1.

Tables stored in the database are given names according to a naming convention involving other station-type abbreviation (abbreviations for the Norwegian names). That is the AL-table for i.e. Precipitation stations is named NBV, the HL tables for these stations are named N_____, i.e. N18700 where the number is the station local number (climatological number).

Station type (abbreviated)	Station type (in Norwegian)	Station type (in English)	obs. pr. day	no. of stations	amount of data (Mb)
N	Nedbørstasjoner	Precipitation stations	1	600	2000
V	Værstasjoner	Weather stations (synoptic, climatological, evaporation and radiation-stations)	3 - 4	250	4000
A	Automatstasjoner	Automatic Weather stations	24	50	1500
M	Maritime stasjoner	Maritime stations (ship-synop)	24	60	400
R	Radiosondestasjoner	Upper Air Sounding stations	2, 4	8	2000
P	Plumaticstasjoner	Plumatic stations (pluviograph)	24	50	300

The number of parameters varies from station-type to station type (from only 6 parameters up to about 60 parameters pr. record).

All tables in AL and HL have station number, year, month, day, time as primary key (unique identifier) and the index is also made up of these columns. After experimenting with the *date* datatype in ORACLE, we decided to implement the columns year, month, day and time instead. This gives radical reduced response time in queries, although it leads to quite complicated conversions when converting to the date format is necessary.

The station number will always be foreign key to the information tables, sometimes together with the station-type literal and sometimes also the year, month and day.

See appendix 2 for «CREATE» statements run for the different station types, both for the HL-tables and the AL-tables (where such exists).

4.3 Dataflow into the database.

Different station-type data is loaded into the database in quite different ways, depending on how they are transmitted from the stations and how they are received by DNMI.

Precipitation data is written by the observer on cards and posted once a week. When received they are manually punched (in a form) into the database (ALN) and controlled.

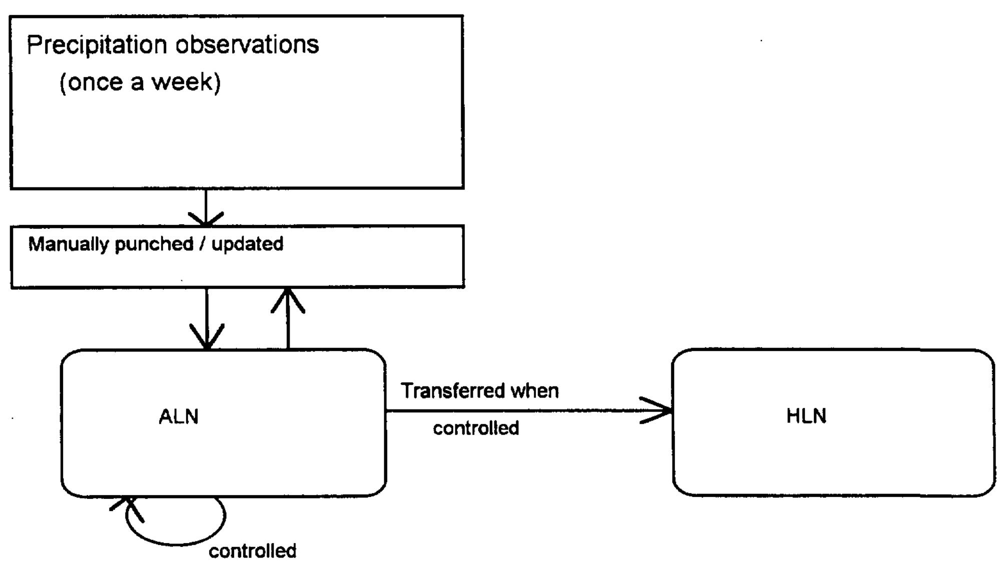

Fig. 4.3.1 Dataflow from Precipitation stations into the database

Weather station-data is loaded into the database (ALV) by two channels. The synoptic-data is loaded (by a PRO*C program) after each observation time, while data from the Climatological stations and parameters from the Synoptic-stations not telegraphed, is manually punched (in a form) into the database once a month.

Automatic Weather station-data is automatically loaded (by SQL*LOADER) into the database (ALA) once a day.

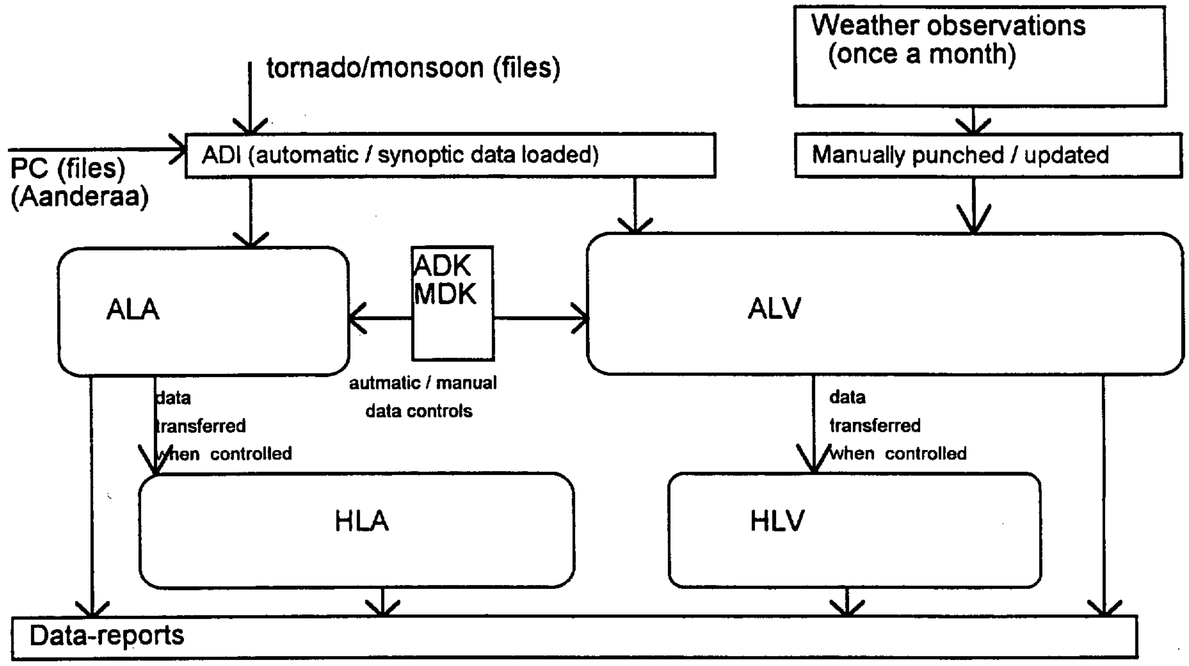

Fig. 4.3.2 Dataflow from Weather and Automatic Weather stations into the database

Data from Maritime stations is first received and controlled on another computer and written to a PC-diskette. These data is loaded into the database (HL) by SQL*LOADER.

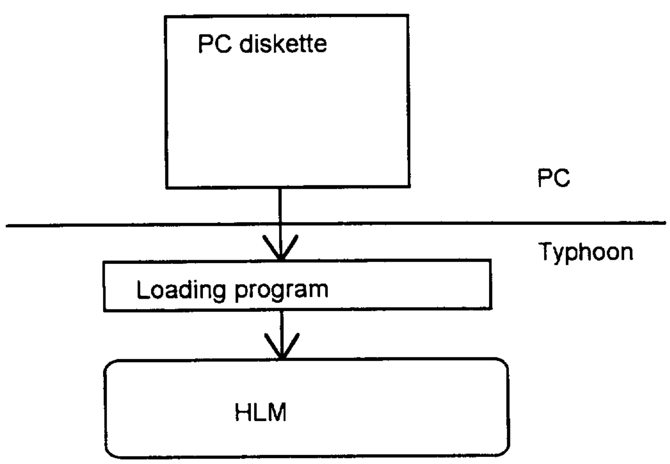

Fig. 4.3.3 Dataflow from Maritime stations into the database.

Data from Upper Air Sounding stations is loaded into the database (HL) from a PC, after controls have been performed on the PC, about once a month, by SQL*LOADER.

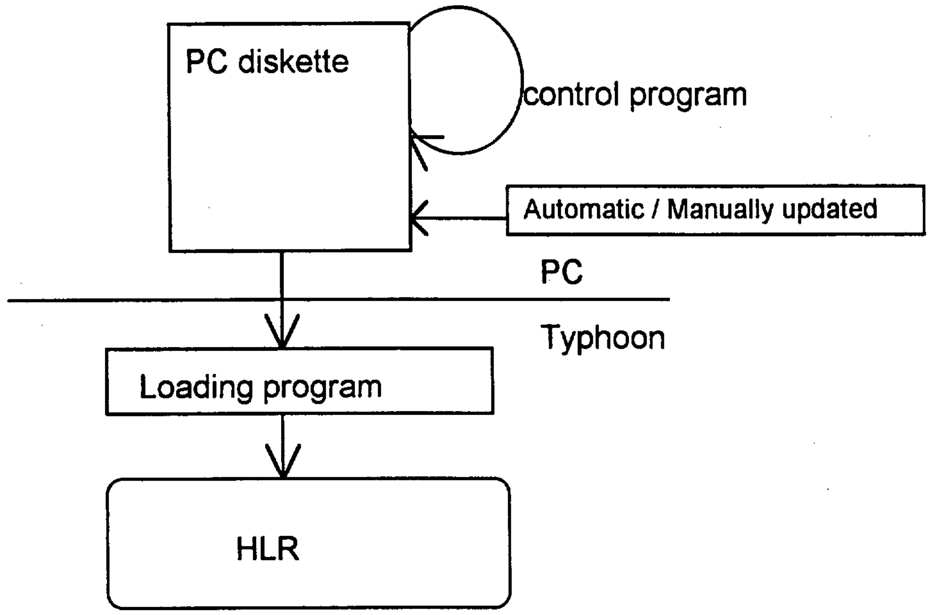

Fig. 4.3.4 Dataflow from Upper Air Sounding stations into the database.

Plumatic station-data is controlled and transformed to hourly data on a PC system before loaded into the database (HL) by SQL*LOADER.

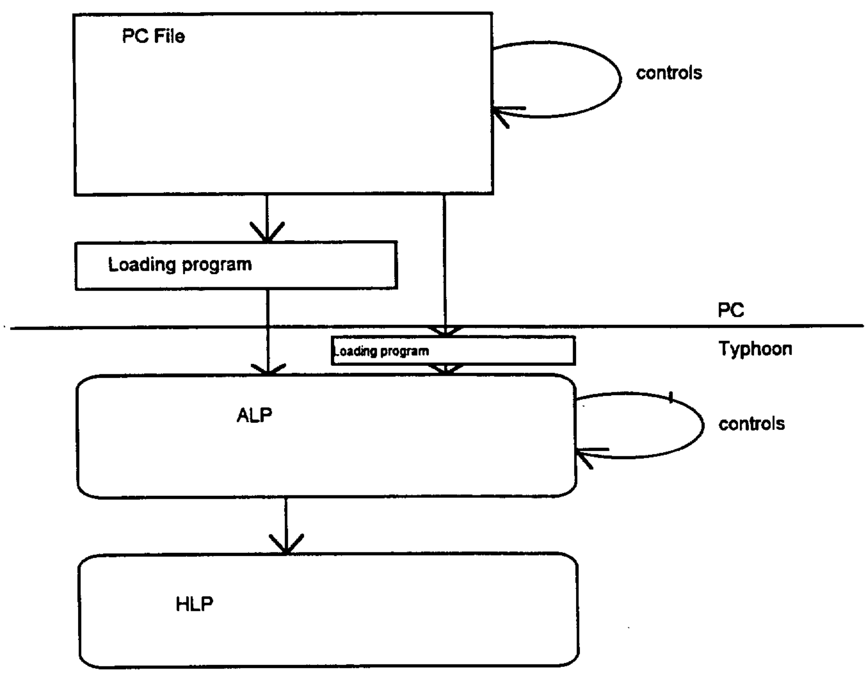

Fig. 4.3.5 Dataflow from Plumatic stations into the database.

4.4 Data control performed in KLIBAS.

As given in chapter 4.3 only Precipitation station-data, Weather station-data and Automatic Weather station-data is controlled in KLIBAS. Controls regarding the other station-types are performed on other systems and is not commented here.

The total control of precipitation data is made up of several controls performed at different times. First a general value check for each parameter is performed when punched. Some control sums are also calculated and checked in the form.

Normally a logical geophysical consistence check (within each observation) is performed once a week. Each month a spatial control is performed. No updates are done automatically, but reports are produced and the suspicious values interpreted and perhaps corrected. When data is marked «all control finished» in ALN, HLN is updated.

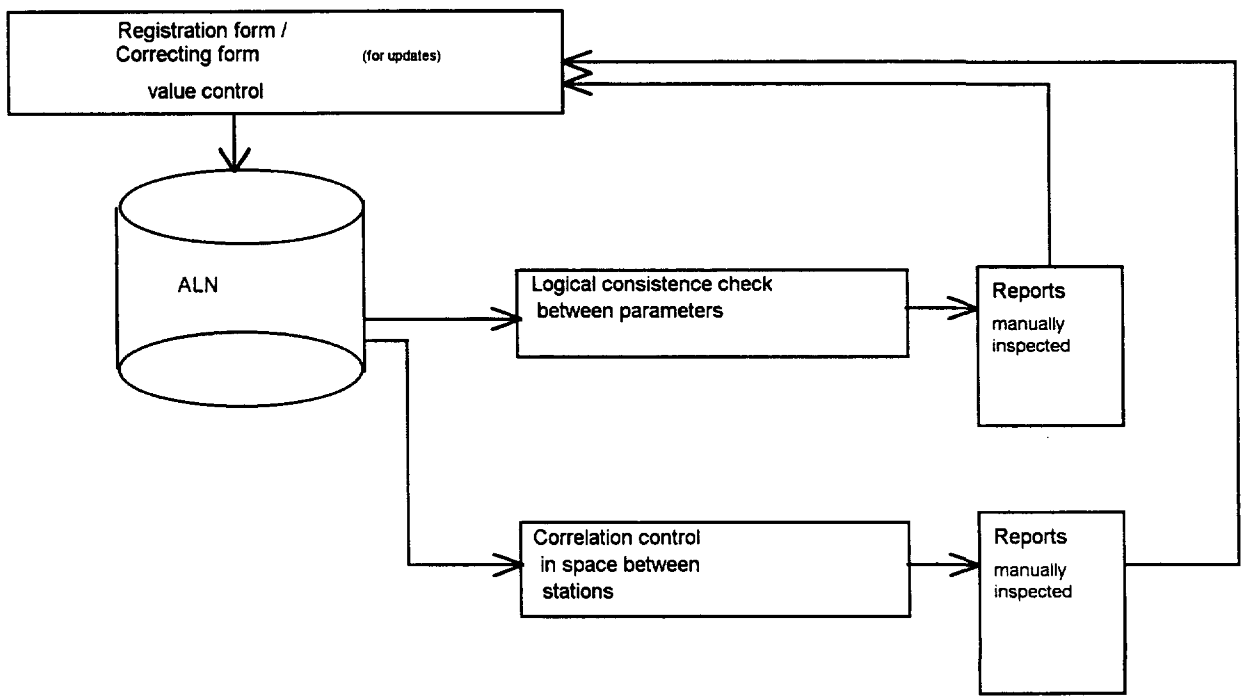

Fig. 4.4.1 Control-filters for Precipitation station-data

The controls performed for Synoptic Weather stations and Automatic Weather stations are being developed in this moment. There is a value check as the data is loaded into the database for the first time. Then different types of controls, as logical tests between parameters in each observation and time controls are performed. The data may be updated automatically or manually.

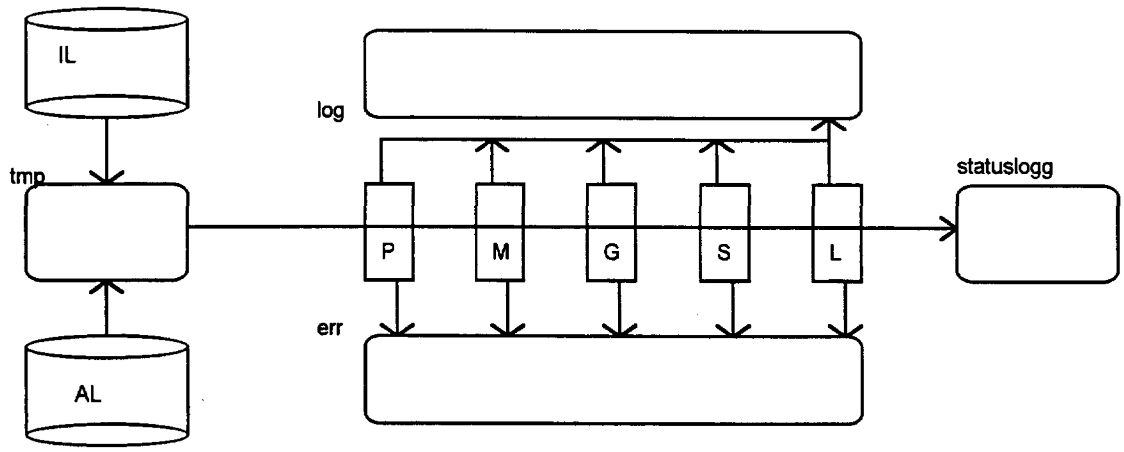

Fig. 4.4.2 Control-filters, P - data coverage, M - outer-layer controls, G time control, S - formal logical controls and L - suspicious data registration, for Automatic Weather station-data.

Weather stations will in addition to the controls performed for Automatic Weather stations also have validation of the data being punched in the form. The controls described for Automatic Weather stations might be run several times, first for the synoptic station-data when they are loaded into the database, and then repeated after all data have being manually punched (may be a month later).

4.5 Extracting data from the database - data reports.

The old system comprised hundreds of programs, doing all sorts of statistics and presentations. It will take quite a long time to develop all these programs or transform them to the new computer and database. Nearly all these programs only accessed one station-type. To reduce the amount of applications to be developed, great work has been made in specifying the most needed general applications, including data-presentations and statistics. In addition special applications have to be developed. Even if we hope to use commercial software as statistic packages, the work is formidable.

As people are skilled in using the database, there might also be possible for some users to extract data directly through SQL. The general applications will be the standard way of getting data.

A great effort is put in specifying the general applications, algorithms and general handling rules. This work is done by a usergroup familiar with the old applications.

In figure 4.5.1 we give an example of a general application specified by the usergroup

DATAUTLISTING FOR DAGLIGE VERDIER AV LUFTTEMPERATUR, T, SKYDEKKE, N, OG NEDBØR, R.

1870 Oslo-Blindern (59.57N, 10.43Ø)
Januar 1994

Stasjonshøyde: 94 m

	Lufttemperatur						Skydekke				Nebørmengde		
DT	T07	T13	T19	Tm	Tx	Tn	N07	N13	N19	N	R07	R19	R
1	-10,2	-8,5	-8,3	-7,3	0,5	-11,1	4	6	8	18			0,0
2	-9,0	-6,0	-9,0	-8,7	-5,9	-10,9	6	6	2	14		0,0	
3	-6,8	-6,8	-8,8	-7,5	-5,3	-9,0	7	7	6	20		0,1	0,0
4	-8,2	-6,5	-5,4	-7,0	-5,4	-8,9	8	7	8	23		0,5	0,1
5	-3,7	-2,1	-0,7	-2,6	-0,5	-5,4	8	8	8	24	4,9	4,4	5,4
6	-0,2	0,1	0,4	0,0	0,4	-0,8	8	8	8	24	0,9	0,4	5,3
7	0,7	1,1	0,6	0,6	1,2	0,0	8	8	7	23	0,2	6,0	0,6
8	-1,2	-2,2	-6,2	-3,2	0,7	-6,3	8	4	3	15	2,4		8,4
9	-11,6	-8,7	-7,4	-9,2	-6,2	-11,6	4	6	8	18		0,1	
10	-6,0	-4,6	-4,2	-5,5	-4,2	-7,6	8	8	8	24	0,5	0,1	0,6
11	-4,4	-4,0	-4,0	-4,2	-3,8	-4,6	8	8	7	23	0,0	1,0	0,1
12	-4,5	-3,3	-1,2	-3,2	-1,2	-5,8	8	8	8	24	1,6	2,8	2,6
13	0,4	0,7	1,4	0,5	1,4	-1,3	8	8	5	21	4,5	0,0	17,3
14	2,4	2,5	0,2	1,5	3,2	0,2	8	8	4	20	0,8	0,0	0,8
15	-3,0	-1,7	-2,7	-2,3	0,3	-3,9	8	2	1	11			0,0
16	-7,4	-5,0	-9,6	-7,3	-2,5	-9,9	0	1	1	2			
17	-13,6	-9,0	-10,7	-11,5	-7,9	-14,0	3	2	5	10			
18	-11,3	-8,1	-7,5	-9,9	-7,5	-13,4	7	7	8	22		0,5	
19	-5,2	-4,1	-4,9	-5,3	-3,6	-7,5	8	8	8	24	9,1	0,2	9,6
20	-3,9	-0,9	0,8	-1,8	1,0	-5,3	9	8	8	25			0,2
21	3,2	3,0	3,2	2,9	5,3	0,0	5	9	4	18			
22	0,3	3,0	2,2	1,1	4,1	-2,4	0	3	2	5			
23	-0,9	2,8	0,8	0,0	3,6	-3,6	6	6	3	15		0,0	
24	-3,8	2,1	1,3	-1,4	2,4	-5,7	1	0	1	2			0,0
25	-5,6	-4,4	-5,2	-4,2	1,3	-7,4	1	7	9	17		1,1	
26	-4,1	-3,4	-7,3	-5,2	-2,0	-7,3	8	3	4	15	2,6	0,2	3,7
27	-3,4	-1,8	-3,6	-3,9	0,0	-8,5	7	3	8	18	4,1		4,3
28	-4,1	-4,6	-6,3	-4,8	-2,3	-6,5	8	7	4	19	0,4	0,3	0,4
29	-13,9	-7,8	-9,2	-11,0	-6,3	-14,5	2	5	6	13			0,3
30	-6,0	-5,4	-3,8	-5,7	-3,6	-9,5	8	8	8	24	6,0	0,0	6,0
31	-6,8	-3,1	-2,8	-4,8	-1,8	-7,8	7	3	6	16	0,0		0,0
SUM											38,0	17,7	65,7
MID	-4,9	-3,1	-3,8	-4,1	-1,4	-6,8	6	6	6	18			
ST.A	4,4	3,7	4,0	3,8	3,6	4,1	3	3	3	6			
N 61-90				-4,3									49
% / AVV.				0,2									135
MIN	-13,9	-9,0	-10,7	-11,5	-7,9	-14,5							
DAG													
MAKS	3,2	3,0	3,2	2,9	5,3	0,2					9,1	6,0	17,3
DAG													
MED	-4,4	-3,4	-4,0	-4,2	-1,2	-7,3	8	7	6	18			

ANT.D.
MED:

Tm<0	24		
Tm>=10	0		
Tn<-10	6		
Tn<0	28		
Tx>=0	14		
Tx>=10	0		
N<=4		2	
N>=20		14	
R>=0.1			17
R>=1.0			9
R>=10.0			1

Tm er døgnets middeltemperatur, basert på T07,T19,Tn(døgn) og Tx(døgn). I tabellene er Tn minimumstemperatur kl.07 og Tx maksimumstemperatur kl.19. Skydekket er total skymengde målt i 8-deler. 0=skyfritt, 8=overskyet, 9=himmel ikke synlig. R07: Nedbør kl.19-07. R19: Nedbør kl.07-19. R: Nedbør kl.07-07. Noen stasjoner har ikke R19. MID(Tm) er basert på Køppens formel, ikke på middelet av døgnmidler.

DNMI / Klimaavdelingen, tlf. 22963000.
Postboks 43
0313 Oslo

Fig. 4.5.1 Example of a general report.

The general applications are implemented with a user-interface on PC, starting a script on the UNIX-computer which starts several programs (PRO*C programs and libraries) extracting, calculating and formatting the data (see fig. 4.5.2.). The resulting report is transferred back to the PC to be viewed or printed.

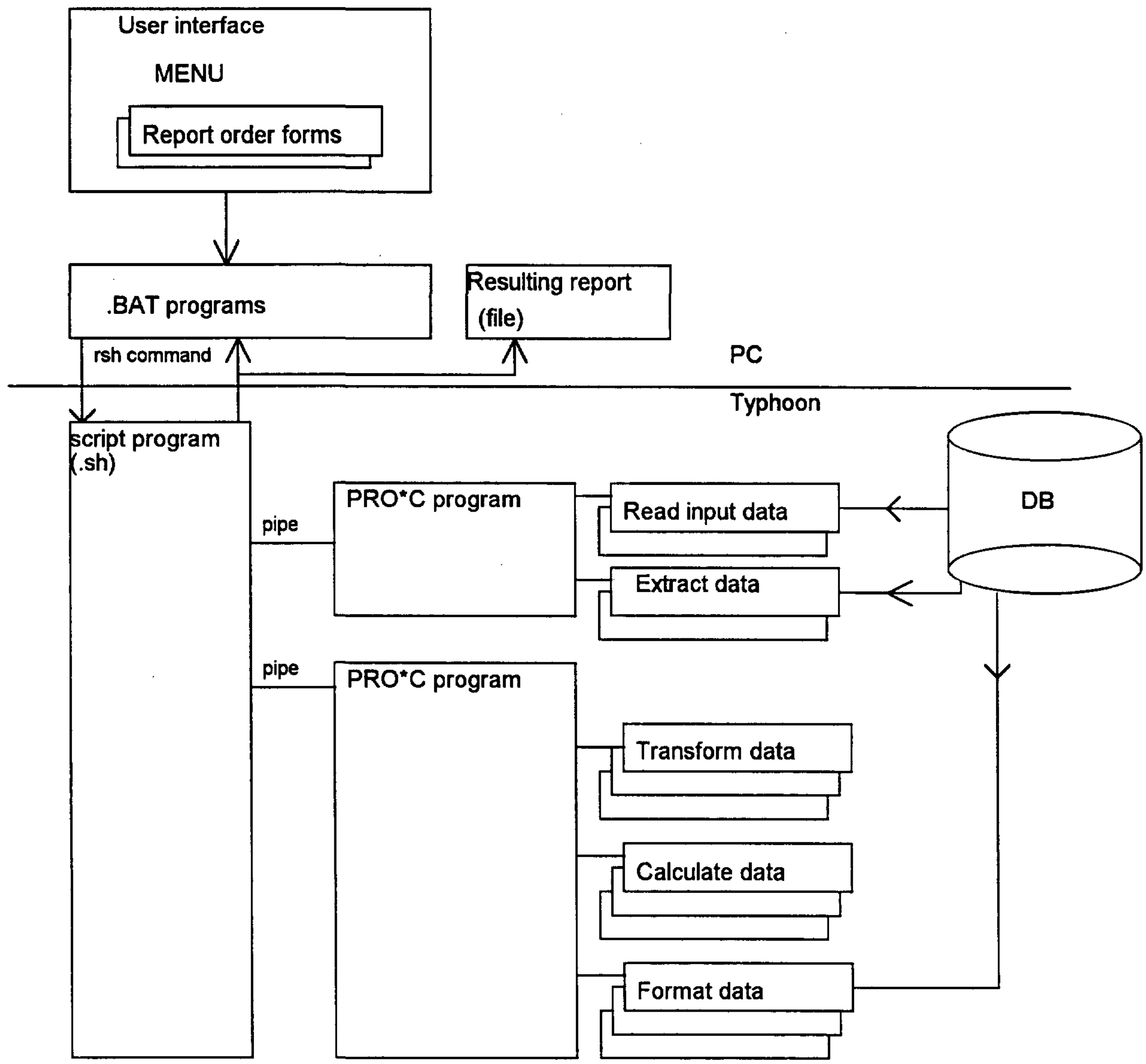

Fig. 4.5.2 System-design for general applications.

5.0 GAINED EXPERIENCE AND CRITICAL FACTORS.

First of all, starting to use new technology takes time (both according to getting knowledge of the system and getting accept in the organisation). Second, designing a great database takes time, and should be handled with care. Third implementing the database takes time (developing routines and applications).

Working with KLIBAS, we used a lot of time learning about RDBMS', ORACLE and UNIX and we made a deep time consuming investigation leading to the design of the database. Afterwards this has proven necessary indeed. When we started implementing the system, some modifications were necessary, but for the main part we just had to follow the specifications.

For one part of the work this might not be fully true. Much time was spent quite early specifying a general application for extracting data. This was before we had the whole picture of the complexity of the demanded applications, the information available in the information archive and all specialities in the data itself. Afterwards, the foundation of the user-group and gained experience from working with the data, application tools and different configurations, have shown quite necessary to develope such applications.

Using PL/SQL as the main application tool in the beginning showed to be a dead end, leading to unacceptable long response time. Many of these applications are now reprogrammed in PRO*C.

Though variable response time it seems as if we have reached acceptable values here, but it is yet to see what happens when the whole division, and may be others simultaneously access the database. Anyhow we believe such problems may be solved by updating the computer, supply more disks, memory and so on (the development here has been enormous only from we bought the computer and up today). One of the main advantages in buying an UNIX computer and ORACLE database should be the upgradability.

APPENDIX 1

Reports

N-22/90	KLIMA	Kravspesifikasjon til ny database/maskin M. Moe et al Requirement report to new database / computer.
32/91	KLIMA	Database/maskin-prosjektet i Klimaavdelingen. Informasjonsmodell, flagging og kontroller. Status pr. 30.06.91 M. Moe, K.A. Iden, P.O. Kjensli, S. Kristiansen, S.L. Lystad, B. Nordin, Å.M. Vidal, T. Aasen The database/computer-project in the Climatological division. Information modelling, flagging of parameters and data controls. Status pr. 30.06.91
40/92	KLIMA	Databaseprosjektet i Klimaavdelingen. Etablering av valgt datastruktur på Typhoon. Delprosjekt 3. Å.M. Vidal, S.L. Lystad, P. Øgland, M. Moe The database-project in the Climatological Division. Implementing chosen datastructure at Typhoon. Sub-project 3.
42/92	KLIMA	Databaseprosjektet i Klimaavdelingen. Utarbeiding og testing av ulike datastrukturer på Typhoon. Delprosjekt 2. P. Øgland, K.A. Iden, P.O. Kjensli, S. Kristiansen, S.L. Lystad, M. Moe, B. Nordin, Å.M. Vidal, T. Aasen The database-project in the Climatological Division Testing of different data-structures at Typhoon. Sub-project 2.
44/92	KLIMA	Databaseprosjektet i Klimaavdelingen. Standarder for systemutvikling. Delprosjekt 4. B. Nordin, M. Moe, P.O. Kjensli, K.A. Iden The database-project in the Climatological Division Standards for system-development. Sub-project 4.
45/92	KLIMA	Databaseprosjektet i Klimaavdelingen. Kvalitetsstyring for prosjektarbeid. Delprosjekt 5. P.O. Kjensli, M. Moe The database-project in the Climatological Division Quality assurance for project work. Sub-project 5.
53/92	KLIMA	Databaseprosjektet i Klimaavdelingen. Status pr. 23.12.92 P. Øgland, M. Moe, P.O. Kjensli, T. Aasen, B. Nordin, Å.M. Vidal, S.L. Lystad, K.A. Iden The database-project in the Climatological Division Status pr. 23.12.92.
08/93	KLIMA	Databaseprosjektet i Klimaavdelingen. Sikkerhetsrutiner. Delprosjekt 7.8 Tom Aasen The database-project in the Climatological Division Safety routines. Sub-project 7.8.
24/93	KLIMA	Theoretical Analysis of the Dip-Test in Quality Control of Geophysical Observations Petter Øgland
42/93	KLIMA	A Method of Weighted Linear Estimation Applied to Quality Control of Precipitation Data Petter Øgland

01/93	KLIBAS	Kontroll av data - spesifikasjonsrapport Per Ove Kjensli, Petter Øgland, Sofus Lystad Control of data - specification report.
02/93	KLIBAS	Data ut - spesifikasjonsrapport M. Moe, P. Øgland, B. Nordin, K.A. Iden, T. Aasen Data reports - specification report.
03/93	KLIBAS	Utskrift av nedbørdata fra arbeidslager Petter Øgland Reports from ALN.
DB-notat 07.09.93		Introduksjonskurs til KLIBAS V01 Margareth Moe et al Introductory course to KLIBAS V01.
01/94	KLIBAS	Data inn - spesifikasjonsrapport Petter Øgland, Tom Aasen, Åse Moen Vidal Incoming data - specification report.
02/94	KLIBAS	Overføring av nedbørdata fra arbeidslager til hovedlager Petter Øgland, Per Ove Kjensli Transferring Precipitation data from ALN to HLN.
03/94	KLIBAS	Databaseprosjektet i Klimaavdelingen. Status pr. 31.12.93 M. Moe, P. Øgland, Å.M. Vidal, T. Aasen, P.O. Kjensli The database-project in the Climatological Division Status pr. 31.12.93
04/94	KLIBAS	Overføring av nedbørdata fra Typhoon til ND-788 Petter Øgland, Tom Aasen, Einar Borvik Transferring Precipitation data from Typhoon to ND-788.
05/94	KLIBAS	Innlasting av synoptiske data til arbeidslager Petter Øgland Loading Synoptic data to ALV
06/94	KLIBAS	VSUKE, NSUKE og VSDUMP: Datautskrifter fra arbeidslager for vær- og nedbørstasjoner Petter Øgland VSUKE, NSUKE and VSDUMP: Reports from AL for Weather and Precipitation stations
07/94	KLIBAS	Romkontroll for nedbørstasjoner over et statisk referansenettverk Petter Øgland Control in space for Precipitation stations over a static reference-network.
08/94	KLIBAS	Først versjon av utapplikasjoner: MNOVS, DAGUT, FLAT, EKSMN, DAGFF, RRMNDSUM Tom Aasen 1. version of reports: MNOVS, DAGUT, FLAT, EKSMN, DAGFF, RRMNDSUM
09/94	KLIBAS	Omlegging av EDB-rutiner for Aanderaa-stasjoner: Spesifikasjonsrapport Petter Øgland, Per Øyvind Nordli Converting of EDB-routines for Aanderaa-stations. Specification report.

10/94	KLIBAS	Databaseprosjektet i Klimaavdelingen. KLIBAS systemoversikt-applikasjoner. Teknisk løsning, Systemoversikt, Meny, Aksessrettigheter, Brukerdialog, Applikasjonsarkitektur Margareth Moe The database-project in the Climatological Division KLIBAS an overview of the system - applications, technical solutions, system-overview, Menu, Accessrights, User-dialog, Application architecture.
11/94	KLIBAS	Utveksling av værstasjonsdata i SUPEROBS-format Petter Øgland Interchanging Weather station data in SUPEROBS-format.
12/94	KLIBAS	Omlegging av EDB-rutiner for MDS - Maritime stasjoner: Spesifikasjonsrapport. Margareth Moe, Helle Tønnesen, Knut A. Iden Converting EDB-routines for MDS - Maritime stations. Specification report
13/94	KLIBAS	Flaggstruktur for nedbørstasjonsdata. Per Ove Kjensli og Petter Øgland. Flag-structure for Precipitation data.
14/94	KLIBAS	Omlegging av EDB-rutine for Plumatic-stasjoner. Spesifikasjonsrapport. Per Ove Kjensli, Nils Langgård, Brita Ullestad og Knut Iden. Converting EDB-routines for Plumatic stations
15/94	KLIBAS	Automatisk overføring av data fra arbeidslager til hovedlager for vær- og nedbørstasjoner Petter Øgland Automatic transmission of data from AL to HL for Weather and Precipitation stations.
16/94	KLIBAS	Biblioteksfunksjoner, mal for utplukksapplikasjoner og miksing av C- og Fortran-kode. Tom Aasen Library-functions, skeleton for report-applications and mixing of C and FORTRAN code.
17/94	KLIBAS	Rapport fra brukergruppen. Forslag til spesifikasjon for data-ut-programmer. Lars Andresen Report from the user-group. Proposal to specification for data-extracting programs.
18/94	KLIBAS	Eksperimentering med enkel kvalitetskontroll og interpolasjon av værstasjonsdata med varierende tidsoppløsning. Petter Øgland Experimenting with simple quality-control and interpolation of weather station data on different timescales.
19/94	KLIBAS	En kort innføring i Z-systemet Sofus L. Lystad og Petter Øgland A short introduction to the Z-system.
20/94	KLIBAS	Utlisting av nedbørstasjonsdata fra arbeidslager: Revidert og modularisert programvare Petter Øgland Report of precipitation data from AL. Revised and modulated programs.

21/94 KLIBAS Eksempelkatalog over data-ut-programmer for værstasjoner på ND-100/788
Anne Eriksen og Petter Øgland
Example catalogue for applications extracting data for weather-stations on ND (the old system).

22/94 KLIBAS Data-kontroll for Aanderaastasjonar: PD-rutinen
Per Øyvind Nordli og Petter Øgland
Data controls for Aanderaa-stations. The PC-routine.

23/94 KLIBAS Kvalitetskontroll av værstasjonsdata i Klimaavdelingen
Lori Håland og Petter Øgland
Quality control of water-stations in the Climatological division.

24/94 KLIBAS Databaseprosjeket i Klimaavdelingen. Status pr. første halvår 1994.
Petter Øgland, Per Ove Kjensli, Margareth Moe, Åse Moen Vidal, Tom Aasen.
The database-project in the Climatological Division Status pr. 1. half year of 1994.

25/94 KLIBAS Skisse til et generelt data-ut-system for geofysiske data
Petter Øgland
Outlines to a general system extracting geophysical data.

26/94 KLIBAS Systemet for kvalitetskontroll av timevise værstasjonsdata på ND-100/788: SUPER-TEST
Petter Øgland og Margareth Moe
System for quality-control of timely weather-station data on ND-computer: SUPER-TEST.

27/94 KLIBAS Skisse til et generelt data-kontroll-system for geofysiske data
Petter Øgland
Outlines to a general data-control system for geophysical data.

28/94 KLIBAS Omlegging av databaserutiner fra Oracle6 til Oracle7
Petter Øgland, Per Ove Kjensli, Margareth Moe, Åse Moen Vidal, Tom Aasen.
Converting databaseroutines from ORACLE6 to ORACLE7.

29/94 KLIBAS System for ukentlig utskrift av automatstasjonsdata
Petter Øgland
System for weekly reports from Automatic Weather-stations.

30/94 KLIBAS Testrutine for ukeabonnement på dataoversikt for utvalgte værstasjoner
Petter Øgland
Test-routine for weekly-subscription on data-reports from selected weather-stations.

31/94 KLIBAS Brukarrettleiing AUTO. Datainnsamling frå automatiske vêrstasjonar (begrenset tilgjengelighet)
Espen Waage
Userguide AUTO. Data-collection from Automatic Weather-stations (restricted).

32/94 KLIBAS Månedlig rutine for Innlasting av automatstasjonsdata i arbeidslager
Petter Øgland
Monthly routine for loading Automatic Weather-station data into ALA.

33/94 KLIBAS Praktisk rutine for kvalitetssikring av programvare
Petter Øgland, Per Ove Kjensli og Nils O. Langgård
Practical routine for assuring quality of applications.

34/94	KLIBAS	Kvalitetshåndbok for databaseprosjektet i Klimaavdelingen Kjensli, Langgård, Moe, Vidal, Øgland, Aasen Quality-handbook for the database-project in the Climatological Division
35/94	KLIBAS	Rutine for kvalitetskontroll av automatstasjonsdata Petter Øgland og Sofus L. Lystad Routine for quality-control of Automatic Weather-station data.
36/94	KLIBAS	ADK/ADL: Datakontroll for EDAS automatstasjoner Sofus L. Lystad og Petter Øgland ADK/ADL: Data control for EDAS Automatic Weather-stations.
37/94	KLIBAS	Spesifikasjonsrapport for omlegging av sonderutinen Tom Aasen og Bjørn Nordin Specification report for converting of Upper Air Sounding data.
38/94	KLIBAS	System for ukentlig utskrift av automatstasjonsdata. Versjon 2 Petter Øgland og Lori Håland System for weekly report of Automatic Weather-station data. Version 2.
39/94	KLIBAS	Romkontroll for nedbørstasjoner over et dynamisk generert referansenettverk Petter Øgland Controls in space for precipitation-stations using a dynamic generated reference-network.
40/94	KLIBAS	Innlasting av synoptiske data i arbeidslager. Revidert utgave. Petter Øgland Loading synoptic data into AL. Revised report.
DB-notat 16.04.94		Innføring i Pro*Fortran Åse Moen Vidal og Per Ove Kjensli Introduction to Pro*Fortran.
DB-notat 29.04.94		Stasjonsarkiv Åse Moen Vidal The station archive.
DB-notat 21.07.94		Post-adresser med tilhørende kommune og fylke Åse Moen Vidal Post-addresses and matching local county / county.
DB-notat 22.08.94		Soltimer Åse Moen Vidal Sun-hours.
DB-notat 13.09.94		Oracle database. Versjon 7 = funksjonalitet Åse Moen Vidal Oracle database. Version 7 = functionality.
DB-notat 30.09.94		Dataoversikt for værstasjoner Åse Moen Vidal Data-overview for weather-stations.
DB-notat 23.11.94		Dataperiode for meteorologiske stasjoner i ORACLE Åse Moen Vidal Data period for meteorological stations in ORACLE.

DB-notat 05.12.94 — Kunderegister sortert etter kundenummer, alfabet, kode
Åse Moen Vidal
Customer-register, ordered after customer-number, alphabetical, code.

R-01/95 KLIBAS — Geofysisk konsistenskontroll av nedbørdata i arbeidslager
Per Ove Kjensli, Margareth Moe, Petter Øgland
Geophysical consistence control of precipitation-data in AL.

R-02/95 KLIBAS — Overføring av værstasjonsdata fra klimalageret (ND-788) til nedbørlageret (ALN på Typhoon)
Per Ove Kjensli
Transferring weather-station data from the ND-computer to ALN on Typhoon

R-03/95 KLIBAS — Kvalitetshåndbok for databaseprosjektet i Klimaavdelingen. Del II
Langgård, Kjensli, Moe, Vidal, Øgland, Aasen
Quality-handbook for the database-project in the Climatological Division. Part II

R-04/95 KLIBAS — Revisjon av data-ut-programmene RRUTM og VSUKE
Petter Øgland
Revision of the data extracting programs RRUTM and VSUKE.

R-05/95 KLIBAS — Programvare for nedbørrutinen ved Klimaavdelingen
P. Øgland, P.O. Kjensli, M. Moe, Å.M. Vidal, T. Aasen
Applications for the precipitation-routine at the Climatological Division.

R-06/95 KLIBAS — Databaseprosjektet i Klimaavdelingen. Status pr. årsskifte 1994/95
P. Øgland, P.O. Kjensli, M. Moe, Å.M. Vidal, T. Aasen.
The database-project in the Climatological Division Status pr. 31.12.94.

R-07/95 KLIBAS — Innlasting og uthenting av automatstasjonsdata. Ny utgave
Petter Øgland og Sofus L. Lystad
Loading and extracting data from Automatic Weather stations. New version.

R-08/95 KLIBAS — Programvare for automatrutinen ved Klimaavdelingen
Petter Øgland og Sofus L. Lystad
Applications for the Automatic Weather-station routine at the Climatological Division.

R-09/95 KLIBAS — Innlasting av Aanderaa-data til arbeidslager på Typhoon
Petter Øgland og Per Øyvind Nordli
Loading of Aanderaa-data into AL at Typhoon.

R-10/95 KLIBAS — Innlasting av Aanderaa-data til hovedlager på Typhoon
Petter Øgland og Bjørn Henning Halvorsen
Loading of Aanderaa-data into HL at Typhoon.

R-11/95 KLIBAS — Programvare for Aanderaa-rutinen ved Klimaavdelingen
Petter Øgland og Per Øyvind Nordli
Applications for the Aanderaa-routine at the Climatological Division.

R-12/95 KLIBAS — Automatisk datakontroll for SYNOP og AUTO: ADK v.2.0
Petter Øgland
Automatic data-control for SYNOP and AUTO: ADK v.2.0.

R-13/95 KLIBAS	ADI: Automatisk datainnlasting for AUTO til arbeidslager Petter Øgland ADI: Automatic loading of data for AUTO to AL.
R-14/95 KLIBAS	Konstruksjon av generelt system for uthenting av data Margareth Moe, Tom Aasen og Per Ove Kjensli Construction of a general system extracting data.
R-15/95 KLIBAS	MKK: Månedlig kvalitetskontroll av automatdata Petter Øgland MKK: Monthly quality-control of Automatic Weather station data.
N-01/95 KLIBAS	Kvalitetskontroll av automatstasjonsdata mars 1995 Petter Øgland Quality-control of Automatic Weather station data, March 1995.
N-02/95 KLIBAS	Fremdriftsrapport for automatstasjonene mars 1995 Petter Øgland Progression report for the Automatic Weather stations, March 1995.
N-03/95 KLIBAS	Kvalitetskontroll av automatstasjonsdata januar 1995 Petter Øgland Quality-control of Automatic Weather-station data, January 1995.
N-04/95 KLIBAS	Kvalitetskontroll av automatstasjonsdata april 1995 Petter Øgland Quality-control of Automatic Weather-station data, April 1995.
N-05/95 KLIBAS	Fremdriftsrapport for AUTO- og SYNOP-stasjoner mai 1995 Petter Øgland Progression report for AUTO- and SYNOP stations May 1995.
DB-notat 01.03.1995	KLIMAMENYEN pr. 1.3.1995 (SQL*Menu 5.0) Margareth Moe The climate-menu pr. 1.3.1995 (SQL*Menu 5.0).

APPENDIX 2

Create statements for AL and HL tables, Automatic Weather stations, Weather stations, Maritime stations, Precipitation stations, Plumatic stations and Upper Air Sounding stations.

ALN, AL for Precipitation station data:

```
CREATE TABLE aln
  (STNR NUMBER(5)  constraint aln_fk_stnr references st_info(stnr),
   AAR  NUMBER(4) CONSTRAINT alnch_aar check(aar between 1500 and 2100),
   MND  NUMBER(2) CONSTRAINT alnch_mnd check(mnd between 1 and 12),
   DAG  NUMBER(2) CONSTRAINT alnch_dag check(dag between 1 and 31),
   TIM  NUMBER(2) CONSTRAINT alnch_tim check(tim between 0 and 23),
   RR   NUMBER(5) CONSTRAINT alnch_rr check(rr between -1 and 5000),
   V4   number(2)   CONSTRAINT alnch_v4 check(v4 between -1 and 30),
   V5   number(2)   CONSTRAINT alnch_v5 check(v5 between -1 and 30),
   V6   number(2)   CONSTRAINT alnch_v6 check(v6 between -1 and 30),
   V7   number(2)   CONSTRAINT alnch_v7 check(v7 between -1 and 30),
   v8   number(2)   CONSTRAINT alnch_v8 check(v8 between -1 and 30),
   v9   number(2)   CONSTRAINT alnch_v9 check(v9 between -1 and 30),
   v10  number(2)   CONSTRAINT alnch_v10 check(v10 between -1 and 30),
   v11  number(2)   CONSTRAINT alnch_v11 check(v11 between -1 and 30),
   v12  number(2)   CONSTRAINT alnch_v12 check(v12 between -1 and 30),
   SS   NUMBER(3)   CONSTRAINT alnch_ss check(ss between -1 and 500),
   SD   NUMBER(1)   CONSTRAINT alnch_sd check(sd between -1 and 4),
   KVPAR VARCHAR2(18),
   KVOBS VARCHAR2(1),
   KONPAR VARCHAR2(18),
   STADIE VARCHAR2(18),
   INTENS VARCHAR2(9),
      CONSTRAINT aln_pk PRIMARY KEY (stnr,aar,mnd,dag)
      USING INDEX PCTFREE 20
      TABLESPACE nindex2 STORAGE (initial 100k next 100k))
      TABLESPACE narbeid
      STORAGE(INITIAL 400k NEXT 400k MINEXTENTS 1 MAXEXTENTS 249
      PCTINCREASE 0)
      PCTFREE 30
      PCTused 60;
```

HLN, HL for Precipitation station data:

```
CREATE TABLE N00060
 (STNR NUMBER(5)  constraint &1.fk_stnr references phyl.st_info(stnr),
  AAR  NUMBER(4) CONSTRAINT &1.ch_aar check(aar between 1500 and 2100),
  MND  NUMBER(2) CONSTRAINT &1.ch_mnd check(mnd between 1 and 12),
  DAG  NUMBER(2) CONSTRAINT &1.ch_dag check(dag between 1 and 31),
  TIM  NUMBER(2) CONSTRAINT &1.ch_tim check(tim between 0 and 23),
  RR   NUMBER(5,1) CONSTRAINT &1.ch_rr check(rr between -1 and 500.0),
  V4   number(2)   CONSTRAINT &1.ch_v4 check(v4 between -1 and 30),
  V5   number(2)   CONSTRAINT &1.ch_v5 check(v5 between -1 and 30),
  V6   number(2)   CONSTRAINT &1.ch_v6 check(v6 between -1 and 30),
  V7   number(2)   CONSTRAINT &1.ch_v7 check(v7 between -1 and 30),
  v8   number(2)   CONSTRAINT &1.ch_v8 check(v8 between -1 and 30),
  v9   number(2)   CONSTRAINT &1.ch_v9 check(v9 between -1 and 30),
  v10  number(2)   CONSTRAINT &1.ch_v10 check(v10 between -1 and 30),
  v11  number(2)   CONSTRAINT &1.ch_v11 check(v11 between -1 and 30),
  v12  number(2)   CONSTRAINT &1.ch_v12 check(v12 between -1 and 30),
  SS   NUMBER(3)   CONSTRAINT &1.ch_ss check(ss between -1 and 500),
  SD   NUMBER(1)   CONSTRAINT &1.ch_sd check(sd between -1 and 4),
  STATUS VARCHAR2(12),
  INTENS VARCHAR2(9),
        constraint &1.ch_stnr check(stnr=substr('&1',2,5)),
        CONSTRAINT &1.pk PRIMARY KEY (aar,mnd,dag)
       USING INDEX PCTFREE 5
        TABLESPACE nindex2 STORAGE (initial 100k next 100k))
        TABLESPACE nhoved2
        STORAGE(INITIAL 100k NEXT 100k MINEXTENTS 1 MAXEXTENTS 249
        PCTINCREASE 0)
        PCTFREE 5
        PCTused 90;
```

ALA, AL for Automatic Weather Station data:

```
create table ala(
stnr number(5) not null,
aar number(4) not null,
mnd number(2) not null,
dag number(2) not null,
tim number(2) not null,
tt number(5,1),
tm number(5,1),
tn number(5,1),
tx number(5,1),
tw number(5,1),
ts number(5,1),
tt#2 number(5,1),
tm#2 number(5,1),
tn#2 number(5,1),
tx#2 number(5,1),
tjm10 number(5,1),
tjm20 number(5,1),
tjm50 number(5,1),
tjm10#2 number(5,1),
tjm20#2 number(5,1),
tjm50#2 number(5,1),
tg number(5,1),
tgm number(5,1),
tgn number(5,1),
tgx number(5,1),
st number(5,1),
uu number(3),
um number(3),
ux number(3),
ff number(5,1),
fm number(5,1),
fg number(5,1),
fx number(5,1),
ff2 number(5,1),
fm2 number(5,1),
fg2 number(5,1),
fx2 number(5,1),
fm35 number(5,1),
dd number(3),
dd2 number(3),
dd35 number(3),
dm number(3),
dg number(3),
dx number(3),
q0 number(7,2),
qt number(7,2),
bt number(5,1),
ra number(7,2),
rw number(7,2),
rwa number(7,2),
rr number(7,2),
ss number(7,2),
rr#2 number(7,2),
rr_12 number(7,2),
rr_24 number(7,2),
rt number(2),
p0 number(7,1),
p0m number(7,1),
p0n number(7,1),
p0x number(7,1),
p number(7,1),
pf number(7,1),
pp number(5,1),
a number(1),
ic number(3),
constraint pk_ala primary key (stnr,aar,mnd,dag,tim)
using index pctfree 5
tablespace aarbeid storage (initial 500K next 500K))
tablespace aarbeid storage (initial 500K next 500K
minextents 2 maxextents 249 pctincrease 0)
pctfree 5
pctused 90;
```

HLA, HL for Automatic Weather Station data:

```
create table A18700(
stnr number(5) not null,
aar number(4) not null,
mnd number(2) not null,
dag number(2) not null,
tim number(2) not null,
tt number(5,1),
tm number(5,1),
tn number(5,1),
tx number(5,1),
tw number(5,1),
ts number(5,1),
tt#2 number(5,1),
tm#2 number(5,1),
tn#2 number(5,1),
tx#2 number(5,1),
tjm10 number(5,1),
tjm20 number(5,1),
tjm50 number(5,1),
tjm10#2 number(5,1),
tjm20#2 number(5,1),
tjm50#2 number(5,1),
tg number(5,1),
tgm number(5,1),
tgn number(5,1),
tgx number(5,1),
st number(5,1),
uu number(3),
um number(3),
ux number(3),
ff number(5,1),
fm number(5,1),
fg number(5,1),
fx number(5,1),
ff2 number(5,1),
fm2 number(5,1),
fg2 number(5,1),
fx2 number(5,1),
fm35 number(5,1),
dd number(3),
dd2 number(3),
dd35 number(3),
dm number(3),
dg number(3),
dx number(3),
q0 number(7,2),
qt number(7,2),
bt number(5,1),
ra number(7,2),
rw number(7,2),
rwa number(7,2),
rr number(7,2),
ss number(7,2),
rr#2 number(7,2),
rr_12 number(7,2),
rr_24 number(7,2),
rt number(2),
p0 number(7,1),
p0m number(7,1),
p0n number(7,1),
p0x number(7,1),
p number(7,1),
pf number(7,1),
pp number(5,1),
a number(1),
ic number(3),
constraint pk_ala primary key (stnr,aar,mnd,dag,tim)
using index pctfree 5
tablespace ahovedl storage (initial 500K next 500K))
tablespace ahovedl storage (initial 500K next 500K
minextents 2 maxextents 249 pctincrease 0)
pctfree 5
pctused 90;
```

ALV (preliminary version), AL for Weather station data:

CREATE TABLE ALV

```
(STNR NUMBER(5)   CONSTRAINT FK_STNR REFERENCES ST_INFO(STNR),
 AAR  NUMBER(4)   CONSTRAINT ALV_CH_AAR CHECK(AAR BETWEEN 1995  AND 1999),
 MND  NUMBER(2)   CONSTRAINT ALV_CH_MND CHECK(MND BETWEEN 1     AND 12),
 DAG  NUMBER(2)   CONSTRAINT ALV_CH_DAG CHECK(DAG BETWEEN 1     AND 31),
 TIM  NUMBER(2)   CONSTRAINT ALV_CH_TIM CHECK(TIM BETWEEN 0     AND 23),
 P0   NUMBER(5,1) CONSTRAINT ALV_CH_P0  CHECK(P0  BETWEEN 650.0 AND 1100.0),
 P    NUMBER(5,1) CONSTRAINT ALV_CH_P   CHECK(P   BETWEEN 800.0 AND 1100.0),
 A    NUMBER(1)   CONSTRAINT ALV_CH_A   CHECK(A   BETWEEN 0     AND 9),
 PP   NUMBER(3,1) CONSTRAINT ALV_CH_PP  CHECK(PP  BETWEEN -25.0 AND 25.0),
 TT   NUMBER(4,1) CONSTRAINT ALV_CH_TT  CHECK(TT  BETWEEN -60.0 AND 50.0),
 TN   NUMBER(4,1) CONSTRAINT ALV_CH_TN  CHECK(TN  BETWEEN -60.0 AND 50.0),
 TX   NUMBER(4,1) CONSTRAINT ALV_CH_TX  CHECK(TX  BETWEEN -60.0 AND 50.0),
 TG   NUMBER(4,1) CONSTRAINT ALV_CH_TG  CHECK(TG  BETWEEN -60.0 AND 50.0),
 TW   NUMBER(4,1) CONSTRAINT ALV_CH_TW  CHECK(TW  'BETWEEN -5.0  AND 50.0),
 S    NUMBER(1)   CONSTRAINT ALV_CH_S   CHECK(S   BETWEEN 0 AND 9),
 UU   NUMBER(3)   CONSTRAINT ALV_CH_UU  CHECK(UU BETWEEN  0 AND 100),
 DD   NUMBER(3)   CONSTRAINT ALV_CH_DD  CHECK(DD BETWEEN -1 AND 360),
 FF   NUMBER(4,1) CONSTRAINT ALV_CH_FF  CHECK(FF BETWEEN  0 AND 200.0),
 FB   NUMBER(3)   CONSTRAINT ALV_CH_FB  CHECK(FB BETWEEN  0 AND 12),
 RR   NUMBER(4,1) CONSTRAINT ALV_CH_RR  CHECK(RR BETWEEN -1 AND 200.0),
 EM   NUMBER(1)   CONSTRAINT ALV_CH_EM  CHECK(EM BETWEEN -1 AND 9),
 SS   NUMBER(4)   CONSTRAINT ALV_CH_SS  CHECK(SS BETWEEN -1 AND 800),
 N    NUMBER(1)   CONSTRAINT ALV_CH_N   CHECK(N  BETWEEN  0 AND 9),
 H    NUMBER(1)   CONSTRAINT ALV_CH_H   CHECK(H  BETWEEN -1 AND 9),
 VV   NUMBER(2)   CONSTRAINT ALV_CH_VV  CHECK(VV BETWEEN  0 AND 99),
 V1   NUMBER(2)   CONSTRAINT ALV_CH_V1  CHECK(V1 BETWEEN -1 AND 30),
 V2   NUMBER(2)   CONSTRAINT ALV_CH_V2  CHECK(V2 BETWEEN -1 AND 30),
 V3   NUMBER(2)   CONSTRAINT ALV_CH_V3  CHECK(V3 BETWEEN -1 AND 30),
 WW   NUMBER(2)   CONSTRAINT ALV_CH_WW  CHECK(WW BETWEEN -1 AND 99),
 V4   NUMBER(2)   CONSTRAINT ALV_CH_V4  CHECK(V4 BETWEEN -1 AND 30),
 V5   NUMBER(2)   CONSTRAINT ALV_CH_V5  CHECK(V5 BETWEEN -1 AND 30),
 V6   NUMBER(2)   CONSTRAINT ALV_CH_V6  CHECK(V6 BETWEEN -1 AND 30),
 V7   NUMBER(2)   CONSTRAINT ALV_CH_V7  CHECK(V7 BETWEEN -1 AND 30),
 W1   NUMBER(2)   CONSTRAINT ALV_CH_W1  CHECK(W1 BETWEEN  0  AND 9),
 W2   NUMBER(2)   CONSTRAINT ALV_CH_W2  CHECK(W2 BETWEEN  0  AND 9),
 FX   NUMBER(4,1) CONSTRAINT ALV_CH_FX  CHECK(FX BETWEEN  0  AND 200.0),
 FXB  NUMBER(3)   CONSTRAINT ALV_CH_FXB CHECK(FXB BETWEEN  0  AND 12),
 FG   NUMBER(4,1) CONSTRAINT ALV_CH_FG  CHECK(FG BETWEEN  0  AND 200.0),
 NH   NUMBER(1)   CONSTRAINT ALV_CH_NH  CHECK(NH BETWEEN -1  AND 9),
 CL   NUMBER(1)   CONSTRAINT ALV_CH_CL  CHECK(CL BETWEEN -1  AND 9),
 CM   NUMBER(1)   CONSTRAINT ALV_CH_CM  CHECK(CM BETWEEN -1  AND 9),
 CH   NUMBER(1)   CONSTRAINT ALV_CH_CH  CHECK(CH BETWEEN -1  AND 9),
 NS   NUMBER(1)   CONSTRAINT ALV_CH_NS  CHECK(NS BETWEEN -1  AND 9),
 C    NUMBER(1)   CONSTRAINT ALV_CH_C   CHECK(C  BETWEEN -1  AND 9),
 HS   NUMBER(1)   CONSTRAINT ALV_CH_HS  CHECK(HS BETWEEN -1  AND 89),
 WD1  NUMBER(3)   CONSTRAINT ALV_CH_WD1 CHECK(WD1 BETWEEN -1  AND 9),
 WD2  NUMBER(3)   CONSTRAINT ALV_CH_WD2 CHECK(WD2 BETWEEN -1  AND 9),
 WD3  NUMBER(3)   CONSTRAINT ALV_CH_WD3 CHECK(WD3 BETWEEN -1  AND 9),
 TD   NUMBER(4,1) CONSTRAINT ALV_CH_TD  CHECK(TD BETWEEN -75.0 AND 50.0),
 IR   NUMBER(1)   CONSTRAINT ALV_CH_IR  CHECK(IR BETWEEN  1  AND 4 ),
 IX   NUMBER(1)   CONSTRAINT ALV_CH_IX  CHECK(IX BETWEEN  1  AND 3 ),
 TR   NUMBER(1)   CONSTRAINT ALV_CH_TR  CHECK(TR BETWEEN  1  AND 4 ),
 RT   NUMBER(1)   CONSTRAINT ALV_CH_RT  CHECK(RT BETWEEN  0  AND 9 ),
CONSTRAINT pk_alv PRIMARY KEY (stnr,aar,mnd,dag,tim),
 USING INDEX PCTFREE 5
TABLESPACE varbeid STORAGE (initial 200k  next 200k))
TABLESPACE vindex3
STORAGE(INITIAL 200k NEXT 200k MINEXTENTS 1 MAXEXTENTS 249
PCTINCREASE 0)
PCTFREE 20
PCTused 60;
```

HLV, HL for Weather station data:

```
CREATE TABLE V18700 (STNR NUMBER(5)  constraint &1.fk_stnr references phyl.st_info(stnr),
                AAR   NUMBER(4) CONSTRAINT &1.ch_aar check(aar between 1500 and 2100),
                MND   NUMBER(2) CONSTRAINT &1.ch_mnd check(mnd between 1 and 12),
                DAG   NUMBER(2) CONSTRAINT &1.ch_dag check(dag between 1 and 31),
                TIM   NUMBER(2) CONSTRAINT &1.ch_tim check(tim between 0 and 23),
                TT    number(4,1) CONSTRAINT &1.ch_tt check(tt between -60.0 and 50.0),
                TN    number(4,1) CONSTRAINT &1.ch_tn check(tn between -60.0 and 50.0),
                TX    number(4,1) CONSTRAINT &1.ch_tx check(tx between -60.0 and 50.0),
                UU    number(3)   CONSTRAINT &1.ch_uu check(uu between 0 and 100),
                DD    NUMBER(3)   CONSTRAINT &1.ch_dd check(dd between 0 and 999),
                FF    NUMBER(5,1) CONSTRAINT &1.ch_ff check(ff between 0 and 200.0),
                FB    NUMBER(2)   CONSTRAINT &1.ch_fb check(fb between 0 and 12),
                FXB   NUMBER(2)   CONSTRAINT &1.ch_fxb check(fxb between 0 and 12),
                FX    NUMBER(5,1) CONSTRAINT &1.ch_fx check(fx between  0 and 200.0),
                RR    NUMBER(5,1) CONSTRAINT &1.ch_rr check(rr between -1 and 200.0),
                SS    NUMBER(3)   CONSTRAINT &1.ch_ss check(ss between -1 and 800),
                SD    NUMBER(1)   CONSTRAINT &1.ch_sd check(sd between -1 and 4),
                VV    NUMBER(2)   CONSTRAINT &1.ch_vv check(vv between 0 and 99),
                P0    NUMBER(7,1) CONSTRAINT &1.ch_p0 check(p0 between 650.0 and 1100.0),
                P     NUMBER(7,1) CONSTRAINT &1.ch_p  check(p  between 800.0 and 1100.0),
                A     NUMBER(1)   CONSTRAINT &1.ch_a  check(a  between 0 and 9),
                PP    NUMBER(3,1) CONSTRAINT &1.ch_pp check(pp between 0 and 25.0),
                E     NUMBER(1)   CONSTRAINT &1.ch_e  check(e  between -1  and 9),
                EM    NUMBER(1)   CONSTRAINT &1.ch_em check(em between -1 and 9),
                N     NUMBER(1)   CONSTRAINT &1.ch_n  check(n  between 0 and 9),
                H     NUMBER(1)   CONSTRAINT &1.ch_h  check(h  between -1 and 9),
               V1    number(2)   CONSTRAINT &1.ch_v1 check(v1 between -1 and 30),
                V2    number(2)   CONSTRAINT &1.ch_v2 check(v2 between -1 and 30),
                V3    number(2)   CONSTRAINT &1.ch_v3 check(v3 between -1 and 30),
               WW    NUMBER(2)   CONSTRAINT &1.ch_ww check(ww between -1 and 99),
               V4    number(2)   CONSTRAINT &1.ch_v4 check(v4 between -1 and 30),
                V5    number(2)   CONSTRAINT &1.ch_v5 check(v5 between -1 and 30),
                V6    number(2)   CONSTRAINT &1.ch_v6 check(v6 between -1 and 30),
                V7    number(2)   CONSTRAINT &1.ch_v7 check(v7 between -1 and 30),
                W1    number(2)   CONSTRAINT &1.ch_w1 check(w1 between 0  and 9),
                W2    number(2)   CONSTRAINT &1.ch_w2 check(w2 between 0  and 9),
                NH    number(1)   CONSTRAINT &1.ch_nh check(nh between 0  and 9),
                CL    number(1)   CONSTRAINT &1.ch_cl check(cl between 0  and 9),
                CM    number(1)   CONSTRAINT &1.ch_cm check(cm between 0  and 9),
                CH    number(1)   CONSTRAINT &1.ch_ch check(ch between 0  and 9),
                FG    number(5,1) CONSTRAINT &1.ch_fg check(fg between 0  and 200.0),
                 TG   number(4,1) CONSTRAINT &1.ch_tg check(tg between -60.0 and 50.0),
                TW    number(4,1) CONSTRAINT &1.ch_tw check(tw between -5.0  and 50.0),
                S     number(1)   CONSTRAINT &1.ch_s  check(s  between 0 and 9),
                NS    number(1)   CONSTRAINT &1.ch_ns check(ns between -1 and 9),
                C     number(1)   CONSTRAINT &1.ch_c  check(c  between -1 and 9),
                HS    number(2)   CONSTRAINT &1.ch_hs check(hs between -1 and 89),
      constraint &1.ch_stnr check(stnr=&2),
      CONSTRAINT &1.pk PRIMARY KEY (aar,mnd,dag,tim)
     USING INDEX PCTFREE 5
      TABLESPACE vindex1 STORAGE (initial 200k next 200k))
      TABLESPACE vhoved1
      STORAGE(INITIAL 200k NEXT 200k MINEXTENTS 1 MAXEXTENTS 249
      PCTINCREASE 0)
      PCTFREE 5
      PCTused 90;
```

HLM, HL for Matitime station data (ship-synop):

```
CREATE TABLE M87603
        (STNR NUMBER(5)    NOT NULL
                           CONSTRAINT MDS_REF_STNR_&1             REFERENCES ST_INFO(STNR),
         AAR  NUMBER(4)    NOT NULL
                           CONSTRAINT &1.ch_AAR          CHECK(AAR BETWEEN 1500 AND 2100),
         MND  NUMBER(2)    NOT NULL
                           CONSTRAINT &1.ch_MND          CHECK(MND BETWEEN 1 AND 12),
         DAG  NUMBER(2)    NOT NULL
                           CONSTRAINT &1.ch_DAG          CHECK(DAG BETWEEN 1 AND 31),
         TIM  NUMBER(2)    NOT NULL
                           CONSTRAINT &1.ch_TIM          CHECK(TIM BETWEEN 0 AND 23),
         HFLAGG RAW(11),
         DDDDDD CHAR(8),
         LALA NUMBER(4),
         LOLO NUMBER(4),
         OC   NUMBER(4),
         P    NUMBER(5,1) CONSTRAINT &1.ch_P   CHECK(P   BETWEEN 900.0 AND 1060.0),
         A    NUMBER(1)   CONSTRAINT &1.ch_A   CHECK(A   BETWEEN 0 AND 9),
         PP   NUMBER(3,1) CONSTRAINT &1.ch_PP  CHECK(PP  BETWEEN 0.0 AND 30.0),
         TT   NUMBER(4,1) CONSTRAINT &1.ch_TT  CHECK(TT  BETWEEN -60.0 AND 60.0),
         TW   NUMBER(4,1) CONSTRAINT &1.ch_TW  CHECK(TW  BETWEEN -5.0 AND 25.0),
         TD   NUMBER(4,1) CONSTRAINT &1.ch_TD  CHECK(TD  BETWEEN -60.0 AND 60.0),
         DD   NUMBER(3)   CONSTRAINT &1.ch_DD  CHECK(DD  BETWEEN -1 AND 360),
         FF   NUMBER(4,1) CONSTRAINT &1.ch_FF  CHECK(FF  BETWEEN 0.0 AND  75.0),
         F    NUMBER(2)   CONSTRAINT &1.ch_F   CHECK(F   BETWEEN 0 AND 12),
         N    NUMBER(1)   CONSTRAINT &1.ch_N   CHECK(N   BETWEEN 0 AND 9),
         H    NUMBER(1)   CONSTRAINT &1.ch_H   CHECK(H   BETWEEN 0 AND 9),
         VV   NUMBER(2)   CONSTRAINT &1.ch_VV  CHECK(VV  BETWEEN 90 AND 99),
         PW   NUMBER(2)   CONSTRAINT &1.ch_PW  CHECK(PW  BETWEEN 0 AND 30),
         HW   NUMBER(3,1) CONSTRAINT &1.ch_HW  CHECK(HW  BETWEEN 0.0 AND 20.0),
         PWA  NUMBER(2)   CONSTRAINT &1.ch_PWA CHECK(PWA BETWEEN 0 AND 30),
         HWA  NUMBER(3,1) CONSTRAINT &1.ch_HWA CHECK(HWA BETWEEN 0.0 AND 20.0),
         WW   NUMBER(2)   CONSTRAINT &1.ch_WW  CHECK(WW  BETWEEN -1 AND 99),
         W1   NUMBER(2)   CONSTRAINT &1.ch_W1  CHECK(W1  BETWEEN -1 AND 9),
         W2   NUMBER(2)   CONSTRAINT &1.ch_W2  CHECK(W2  BETWEEN -1 AND 9),
         WD   NUMBER(2)   CONSTRAINT &1.ch_WD  CHECK(WD  BETWEEN -1 AND 9),
         NH   NUMBER(1)   CONSTRAINT &1.ch_NH  CHECK(NH  BETWEEN 0 AND 9),
         CL   NUMBER(2)   CONSTRAINT &1.ch_CL  CHECK(CL  BETWEEN -1 AND 9),
         CM   NUMBER(2)   CONSTRAINT &1.ch_CM  CHECK(CM  BETWEEN -1 AND 9),
         CH   NUMBER(2)   CONSTRAINT &1.ch_CH  CHECK(CH  BETWEEN -1 AND 9),
         FFX  NUMBER(4,1) CONSTRAINT &1.ch_FFX CHECK(FFX BETWEEN 0.0 AND 75.0),
         TZ   NUMBER(1)   CONSTRAINT &1.ch_TZ  CHECK(TZ  BETWEEN 0 AND 9),
         IZ   NUMBER(1)   CONSTRAINT &1.ch_IZ  CHECK(IZ  BETWEEN 1 AND 5),
         ES   NUMBER(3)   CONSTRAINT &1.ch_ES  CHECK(ES  BETWEEN 0 AND 100),
         RS   NUMBER(1)   CONSTRAINT &1.ch_RS  CHECK(RS  BETWEEN 0 AND 4),
         DW1  NUMBER(2)   CONSTRAINT &1.ch_DW1 CHECK(DW1 BETWEEN 0 AND 36),
         DW2  NUMBER(2)   CONSTRAINT &1.ch_DW2 CHECK(DW2 BETWEEN 0 AND 36),
         PW1  NUMBER(2)   CONSTRAINT &1.ch_PW1 CHECK(PW1 BETWEEN 0 AND 60),
         PW2  NUMBER(2)   CONSTRAINT &1.ch_PW2 CHECK(PW2 BETWEEN 0 AND 60),
         HW1  NUMBER(2)   CONSTRAINT &1.ch_HW1 CHECK(HW1 BETWEEN 0.0 AND 10.0)    ,
         HW2  NUMBER(2)   CONSTRAINT &1.ch_HW2 CHECK(HW2 BETWEEN 0.0 AND 10.0)    ,
         CI   NUMBER(2)   CONSTRAINT &1.ch_CI  CHECK(CI  BETWEEN -1 AND 9),
         SI   NUMBER(2)   CONSTRAINT &1.ch_SI  CHECK(SI  BETWEEN -1 AND 9),
         BI   NUMBER(2)   CONSTRAINT &1.ch_BI  CHECK(BI  BETWEEN -1 AND 9),
         DI   NUMBER(2)   CONSTRAINT &1.ch_DI  CHECK(DI  BETWEEN -1 AND 9),
         ZI   NUMBER(2)   CONSTRAINT &1.ch_ZI  CHECK(ZI  BETWEEN -1 AND 9),
         NS1  NUMBER(1)   CONSTRAINT &1.ch_NS1 CHECK(NS1 BETWEEN 0 AND 9),
         C1   NUMBER(2)   CONSTRAINT &1.ch_C1  CHECK(C1  BETWEEN -1 AND 9),
         HS1  NUMBER(2)   CONSTRAINT &1.ch_HS1 CHECK(HS1 BETWEEN 0 AND 89),
constraint &1.ch_stnr check(stnr=substr('&1',2,5)),
CONSTRAINT &1.pk PRIMARY KEY (aar,mnd,dag,tim)
USING INDEX PCTFREE 5
TABLESPACE Mindex1 STORAGE (initial 200k next 200k))
TABLESPACE Mhoved1
STORAGE(INITIAL 500k NEXT 500k MINEXTENTS 2 MAXEXTENTS 249
PCTINCREASE 0)
PCTFREE 5
PCTUSED 90;
```

HLR, HL for Upper Air Sounding-station data:

```
create table R01152
(primary key (aar,mnd,dag,tim,hh),
stnr number(5) not null check (stnr between 1 and 99999),
aar number(4) not null check (aar between 1500 and 3000),
mnd number(2) not null check (mnd between 1 and 12),
dag number(2) not null check (dag between 1 and 31),
tim number(2) not null check (tim between 0 and 24),
asek number(4) check (asek between -999 and 9999),
hh number(5) check (hh between 0 and 99999),
p0 number(5,1) check (p0 between 0 and 1100),
tt number(4,1) check (tt between -199 and 99),
td number(4,1) check (td between -199 and 99),
uu number(3) check (uu between 0 and 105),
dd number(3) check (dd between 0 and 360),
ff number(4,1) check (ff between 0 and 500),
dt char(5))
tablespace data storage (initial 100K next 100K minextents 1
maxextents 240 pctincrease 0) pctfree 10 pctused 75;
```

ALP, HLP, AL and HL for Plumatic station data.

ALP and **HLP** are not implemented yet and create statements not made. Anyhow the table-structure (with columns and check constraints) are specified, though these may be changed later and must be considered as preliminary. The structure is the same for ALP and HLP.

Parameter	Datatype	Unit	Interval
STNR	number(5)		[00000 - 99999]
AAR	number(4)		[1900 - 2100]
MND	number(2)		[1 - 12]
DAG	number(2)		[1 - 31]
TIM	number(2)		[0 - 23]
MIN	number(2)		[0 - 59]
RR	number(4,2)	mm	[0.00 - 10.00] [-1. - -10.]
VIPP	number(3)		[0 - 50] [-1 - -10]
STATUS	char(1)		[0 - 9]