A Database for Silicon Microstrip Detectors in the SCT Central Cluster Working Group

Philippe Demierre and Didier Ferrère
DPNC, University of Geneva
CH-1211 Genève 4, Switzerland.

September 4th, 1998

Abstract

Before the construction of the ATLAS Silicon Central Tracker (SCT), several types of silicon microstrip detectors are tested across the working groups. A Web Oracle Database concept has been setup to be able to have access to relevant information from anywhere on a web server in the Central Cluster working group. This note describes the concept of this database.

1 Introduction

The concept of the database (DB) for silicon microstrip detectors in the SCT[1] Central Cluster in ATLAS is quite simple and is based on an Oracle web interface. Prior to the ATLAS SCT DB[2] planned for the large scale production of silicon microstrip modules, it was decided to set-up a temporar DB in our cluster composed of institutes located at different places (CERN, Cracow, Geneva, Ljubjana, Melbourne and Moscow). The aim of the DB is not only to store data for the book keeping, but also to have an easy access from any cluster site to the existing data and a capability to sort and report on a great number of silicon detectors.

More precisely data which are internal to Oracle and pictures at any URL link can be accessed without any restrictions. Access to the DB is limited via password to the authorized people for selected secure informations.

One of the important features of this Oracle DB is its great flexibility, allowing the continual addition of functionality and development of the data structure.

This note is intended to help people to access to the DB describing the tool, the functionalities and the access.

2 Concept

2.1 Web interface

Communication is established from a Web Browser like Netscape Navigator to the URL of the detector database stand in an Oracle server using an Oracle web server.

The Oracle Web Server 3.0 is responsible for the management of http requests from a Web browser client to the detector database (see the principle of the scheme on Figure 1).

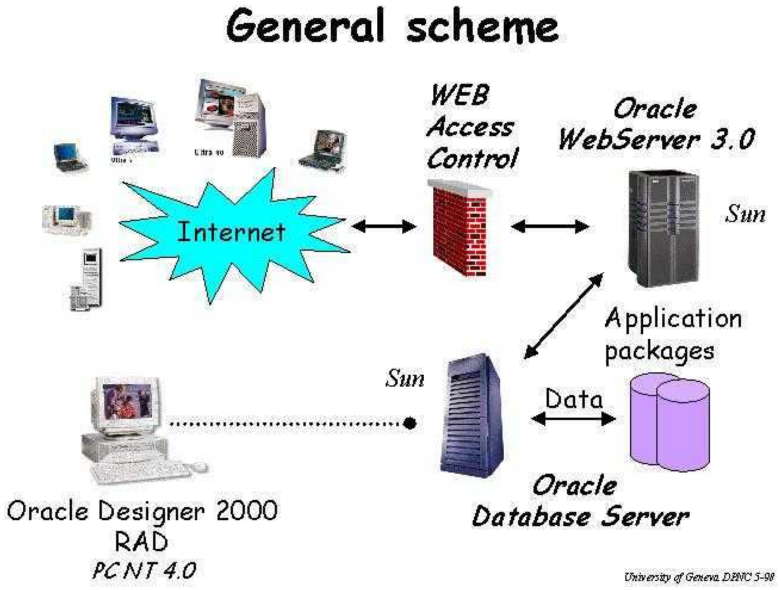

Figure 1: Principle scheme of the web concept of the detector database.

When receiving a http request, the Webserver transmits the request to the Oracle Database Derver 7.3, where the Webdatabase application and the detector data are stored. The database server then responds to the Webserver request and sends the corresponding recorded set result in a wrapped html format. The Webserver then sends the html result of the client http request to the Web client program.

Each access is registered in a logfile describing each command performed by the user. A report of the logfile histories is prepared on a daily basis. As for most of the computing system a back-up of the data is also performed daily.

2.2 Tool

The Oracle Designer/2000 Rapid Application Development Tool was used to develop this Web application. This tool provides a great flexibility of defining a prototype design and generates the application code for the user.

The major contribution of a user to develop a database is to define a clear Entity-Relationship diagram (see Figure 3) thanks to the tool facility. This diagram defines the links and the constraints between entities. Once this is well defined, Designer/2000 generates the tables and the PL/SQL packages and installs them on the Oracle database server. These PL/SQL packages are reponsible to get and/or put the data in the database and to wrap these data with a html layer format.

2.3 Access

In order to keep secure information in the database there is an access privilege and the user is supposed to enter the password in the frame view shown on figure2. Two types of restricted accesses were defined for an extended use through the ATLAS SCT collaboration. One access is restricted to read only and the user access is defined as: web_det. The other access dedicated to detector responsibles of each institution in the Central

Cluster working group allows reading, adding and changing data. In this case the user access is defined as: web_app.

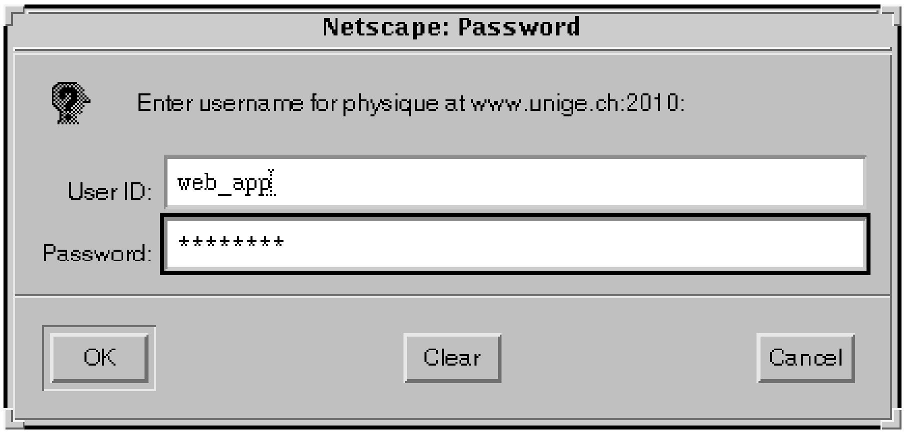

Figure 2: User frame view for web access.

3 Structure

3.1 Architecture

The general scheme and link of the DB is described on the entity-relationship diagram (Figure 3). All types of link are referred directly or indirectly to the main entity **DETECTOR** for which the data have to be set first. Mainly each of the entities have a primary key defined as a sequence number and are preceded with '#'. Then the other attributes in the table are either mandatory or optional preceded respectively with a '$*$' or 'o'.

In this schemactic there are 2 categories: types and physical entities. The three defined types are:

- DETECTOR TYPE:

Barrel p-on-n polysilicon biasing;

Barrel p-on-n implanted biasing;

Barrel n-on-n polysilicon biasing;

Forward W31 p-on-n polysilicon biasing, ...

- MEASUREMENT TYPE:

detector;

diode of $0.5cm^2$;

baby detector, ...

- DEFECT TYPE

Manufactured defect;

pin holes;

short;

open, ...

The types must be defined in the DB before entering any entity linked to them. For instance one detector item could be entered if he has his type defined in **DETECTOR TYPE**.

The events which happen to the main physical entity **DETECTOR** are expressed as other entities linked to it which are:

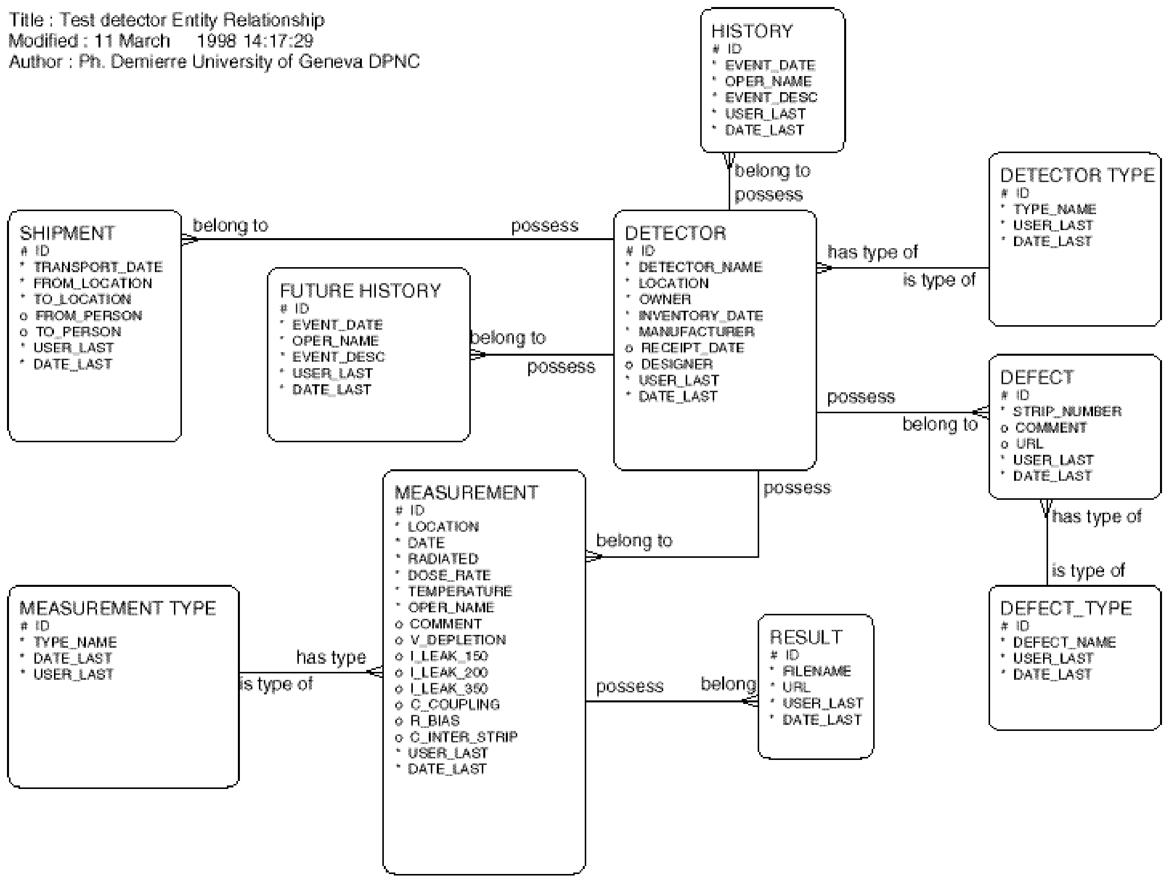

Figure 3: Entity-relationship diagram.

- **MEASUREMENT** - Shows the specification measured and also the results thanks to a wek link;
- **SHIPMENT** - Gives a history of the shipment of an item;
- **DEFECT** - Describes and lists the strip defects
- **FUTURE HISTORY** - Describes the finality of a detector which is normally mounted on a module;
- **HISTORY** - Describes all types of events which happen to a detector.

3.2 User view

When starting from the main menu*, two possibilities are offered to enter in the DB of detectors: either by linking on "manage detector's data" (allows to view, to modify or to add data) or by clicking on "Get measurement data" (allows to view only the results).

The first layout page propose to select criteria for detector or for measurement. After submitting this selection it will list the detector items which will correspond to the criteria or the range of value selected. For any type of selection query, the blank or '%' will means that all records are selected. '%' character helps to find a record that matchs with an existing substring[3].

As shown on figure 4 the layout window of the wrapped Oracle data is based on a left page where all the requested data is listed and on a right page where detailed informations are presented. On the right window the information of a specific entry is displayed and could be modified or deleted and at the bottom some links exist allowing to access the data in the lower stage of the entity-relationship scheme.

4 External options

Insertion of data sheet measurement into the detector DB directly from a local datafile is possible thanks to a Java application. The Java language has been choosen for its neutral architecture. This concept has been developped to allow skilled people to insert quickly special data sheet into the DB. The data can be directly loaded into the DB from a local datafile. A home made Java application is responsible of interpreting the data file and loading them into the DB tables.

The DB connection is controlled by the JDBC (Java Data Base Connectivity) Oracle classes. It is required that the client station installs the JDK 1.1 (Java Development Kit) or JRE 1.1 (Java Runtime Environment) package from Sun microsystem to become a Java virtual machine in order to be able to run Java applications.

5 Conclusion

This web Oracle DB concept was set to simplify the book keeping of caracterisation of several detectors in the SCT central cluster and allows an easy access from anywhere. This DB is a temporar solution for the central cluster before the ATLAS SCT DB will be operational. The concept of this DB is also considered for the official SCT DB and in spite of the complexity of the tables once the entity-relationship diagram is fixed the tool Designer/2000 of Oracle allow a quick, safe and bug free development.

Acknowledgements

First we wish to thank Allan Clark for useful suggestions.

We wish also to acknowledge the computing support of the University of Geneva allowing the use of Oracle development tools and for the skilled help received during our development. We thank particularly Franco Serena for teaching us the use of the RAD tool Designer/2000.

*Web link: http://schp5.unige.ch/atlas/atlaspage/module/detect/database.htm

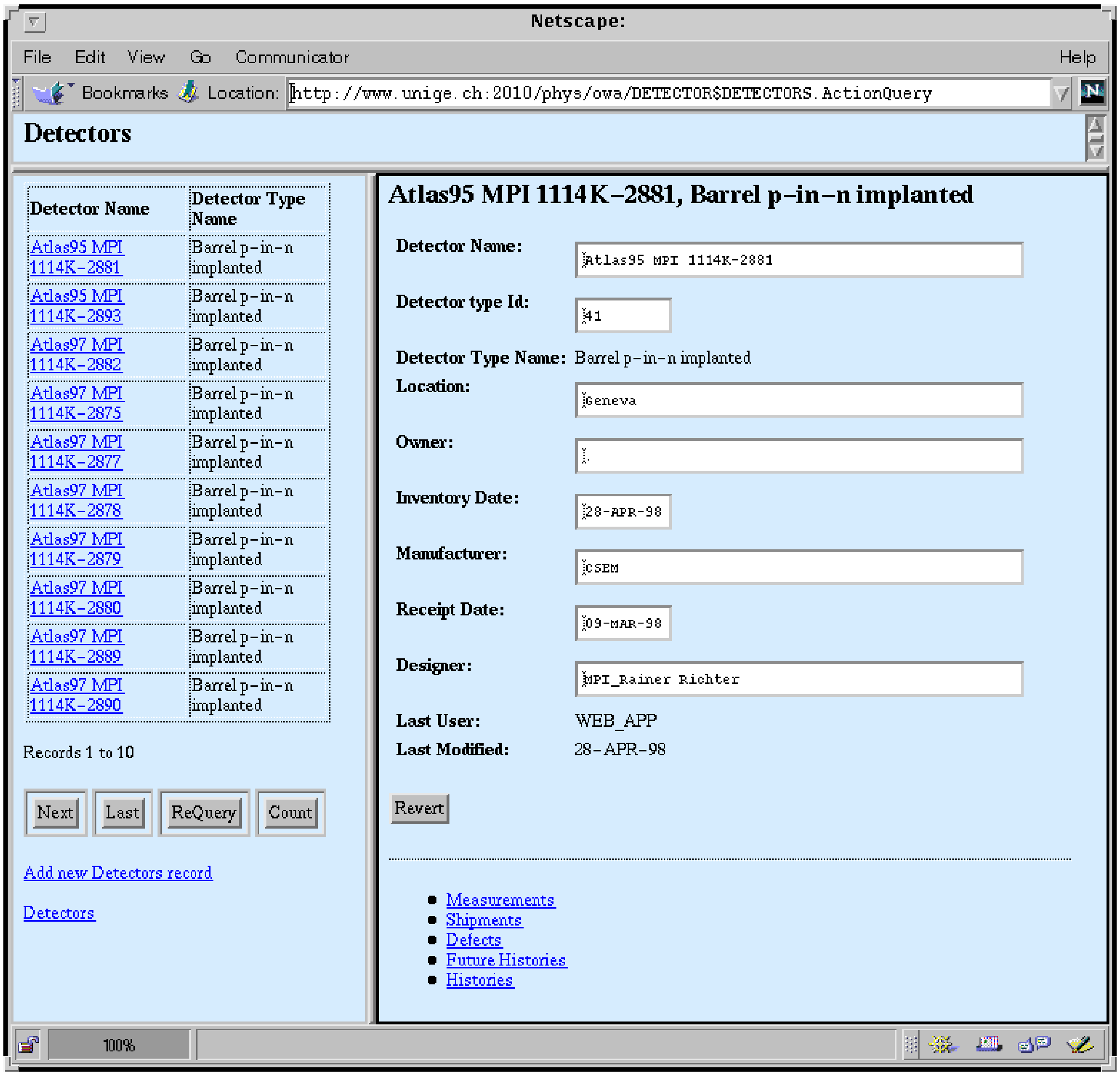

Figure 4: Web welcome page for Detector.

References

[1] ATLAS Inner Detector Thechnical Design Report, CERN Internal Report, CERN/LHCC/97–16 and CERN/LHCC/97–17, April 1997.

[2] Joe Foster and Steve Snow, Proposal for a Database for the ATLAS SCT Construction, ATLAS Internal Note, ATL-INDET-96-144.

[3] Oracle database documentation, Advanced search criteria.