ATLAS Technical Coordination Expert System

Ignacio Asensi Tortajada[1*], *André* Rummler[2], *George* Salukvadze[3], *Carlos* Solans Sánchez[2], and *Kendall* Reeves[4] on behalf of the ATLAS Collaboration.

[1]University of Valencia, Spain
[2]CERN, Geneva, Switzerland
[3]Tbilisi State University, Georgia
[4]University of Texas at Dallas, US

Abstract.
When planning an intervention on a complex experiment like ATLAS, the detailed knowledge of the system under intervention and of the interconnection with all the other systems is mandatory. In order to improve the understanding of the parties involved in an intervention, a rule-based expert system has been developed. On the one hand this helps to recognise dependencies that are not always evident and on the other hand it facilitates communication between experts with different backgrounds by translating between vocabularies of specific domains. To simulate an event this tool combines information from different areas such as detector control (DCS) and safety (DSS) systems, gas, cooling, ventilation, and electricity distribution. The inference engine provides a list of the systems impacted by an intervention even if they are connected at a very low level and belong to different domains. It also predicts the probability of failure for each of the components affected by an intervention. Risk assessment models considered are fault tree analysis and principal component analysis. The user interface is a web-based application that uses graphics and text to provide different views of the detector system adapted to the different user needs and to interpret the data

1 Introduction

The ATLAS [1] Expert System is a diagnostic tool created by Technical Coordination to increase the knowledge base of the ATLAS experiment, allow easier turn around of knowledge between experts and foresee complications before the interventions take place. It describes different systems like sub-detectors, gas, cooling, ventilation, electricity distribution and detector safety systems which result in an extremely complex tree of relations between them. There is a friendly user interface in the form of a graphical simulator which allows the user to simulate an intervention and to foresee its consequences on all the other systems of the experiment.

The requirements of the ATLAS Expert System are the following:

- Provide a description of ATLAS and its elements in a way that is understandable to a multidisciplinary team of experts.

---

*e-mail: ignacio.asensi@cern.ch

- Provide a user friendly representation of the elements and their dependencies in graphical and text manner.

- Emulate the behaviour of the sub-systems by means of a simulator with different scenarios.

- The simulator has to accept input from the user and quickly answer how ATLAS would behave with the given input.

- Use standard technologies if possible to simplify maintenance.

The goal of the ATLAS Expert System is to become an important tool in the Technical Coordination of ATLAS as part of the standard procedure prior to an intervention and to be a helpful tool in the ATLAS Control Room in many situations. It can help in the diagnostic of a complicated problem. When a system is out of power and the reason is unknown it can help to find what are the possible causes. It also can help to take a quick and well educated action when time is critical and experts are not available.

The ATLAS Expert System contains a virtual representation of the ATLAS experiment which is presented to the user in the form of visual diagrams. This representation is also a simulation that imitates the behavior of the infrastructure of ATLAS. In this simulation the user can take actions like switching off a system or triggering an alarm. Once an action is taken, the simulation is triggered and the user can immediately see the consequences. The unit of the structure are the systems, which normally can accept one input, either to be switched on or off. Different ways of obtaining information are available to the user such as list-oriented interfaces and different deduction explanation levels.

The construction of the ATLAS experiment was completed in 2008. Ten years after the construction of ATLAS, the reliability of certain systems is decreasing over time. The ATLAS Expert System makes usage of its knowledge base to study the principal component analysis and the Probability of Failure of the ATLAS systems.

2 System design

From a technical perspective the system is divided in three well separated components: database, python server and a web application with server and client endpoints.

The database stores all the information about the elements that describe ATLAS and the relations between them. The database used is the ATLAS TDAQ object oriented configuration database, so called OKS (Object Kernel Support) [3], which supports the description of objects, classes, relationships and inheritance. It is expected to be maintained during the life of the ATLAS experiment. To describe ATLAS we require three very general types of objects: systems, alarms and actions. Therefore there are three classes. To describe different types of systems we use the inheritance to describe many subclasses of *system*. Each type of system has different attributes. For example, water systems have an attribute of *waterTo* and gas systems have *gasTo*.

A python server reproduces the behaviour of the ATLAS experiment using the elements and relations in the database: it loads elements and relations from the database, provides the scenario to the user, receives user inputs in the scenario and provides the answers to the given inputs.

As shown in Figure 1 the user interface in the client side consist of a web application with JavaScript in the client side and PHP in the web server. Python server, which is kept alive by a service watchdog, creates instances of the database for each user session. Python server, web server and client side communicate with each other using standard protocols like JSON, AJAX and HTTP.
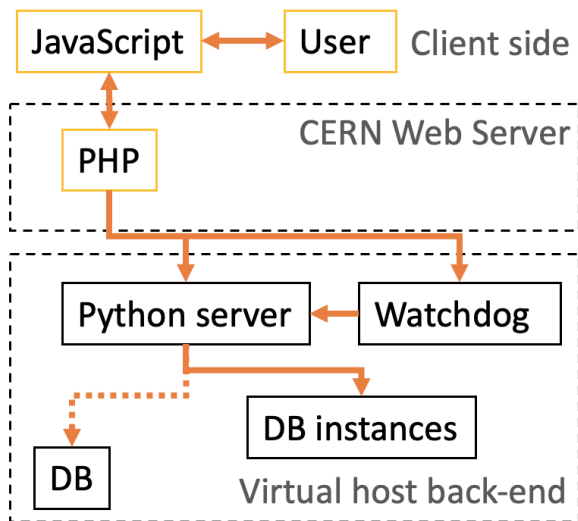
Figure 1. Representation of the Expert System design.

3 Expert System

The acquisition of knowledge has been carried out following different approaches: searching and documenting engineering repositories of technical implementations and electrical layouts, including direct evaluation of systems and visual inspections. Direct contact with experts of the different systems to implement the rules of deduction and analysing events where outcomes where not correctly foreseen. This process has been automatised by means of scripts to ease the transfer of knowledge to the Knowledge Base.

As a result of the knowledge acquisition, the knowledge base containing the rules about how systems interact with each other, is implemented in the database as shown in Figure 2. The ATLAS elements are represented as objects in the database, which include the rules as relationships. They are interpreted by the Inference Engine. Object relationships are a representation of systems inputs and outputs. As an example in Figure 3, a system like the rack Y.33-05.X8 can have more than one type of inputs or relationships. Each input is represented as one arrow pointing to the element and the color indicates the type of relationship. In this case blue is for cooling and black for power.

In order to resolve the state of a system, all the inputs that belong to a relationship have to be resolved as a node first and then logically summed into a single result. This process is done for every system starting from the deepest parent of the object, using a deep-first algorithm shown in Figure 4 to determine the parents tree. Each node is represented by a circle and each step by a number. A deep-first algorithm search will navigate the object, searching parents in increasing number, starting from number 1 to the deepest level in number 4 and then in the same level to number 5. The navigation continues to the upper levels.

The Inference Engine is a control structure implemented in Python. It loads the database and depending on the type of established relations it deducts the behaviour of the elements.

The user interface is based on a web application that provides an easy way to interact with the expert system. It provides information about individual ATLAS systems, relations between them and simulates how systems react to manipulation. Information about elements such as systems or alarms are shown in search pages. They also provide information about
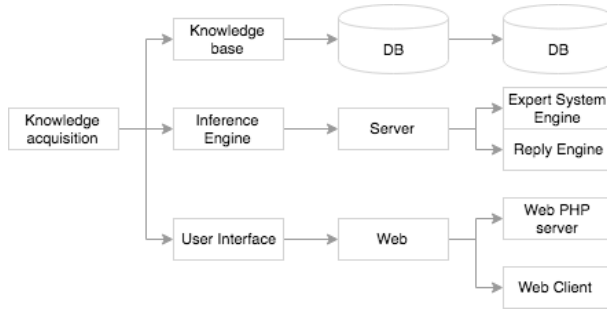
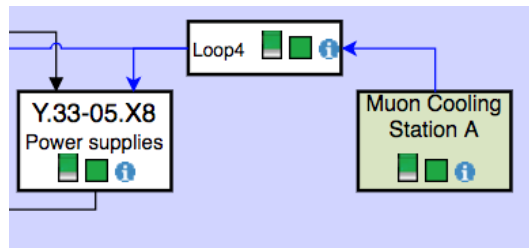**Figure 2.** Three level representation of the Expert System. From abstract to technology implementation

Figure 3. Example of database object relations representing groups of system-actual inputs and outputs
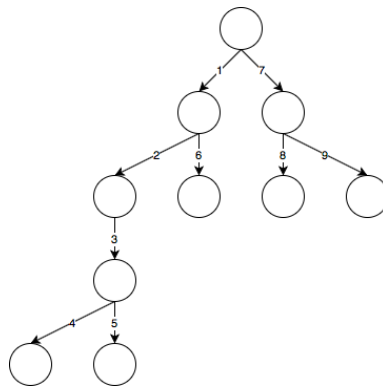

Figure 4. Deep-first search algorithm.

how they are linked together. Answers about consequences and behaviour of elements are best understood by means of simulation panels. These panels show individual systems in form of boxes connected by arrows. The user can interact with the systems and every time there is an interaction, the engine runs a simulation and the system updates its status.

Most of the elements have three icons: switch, state and info.

An explanation mechanism has been implemented for the ATLAS Expert System in order to explain its decisions and its behaviour to help users, experts and developers. Expert System deductions rely on inference and therefore, an explanation mechanism helps users to understand how the Expert System evaluates the scenario and all the steps taken to deduce

Figure 5. Two connected elements. EXD8/15A powering a group of gas racks.

the answer. The explanation mechanism is presented to the user as a list of decisions and explanations in the form of: "System *X* was switched off because it was interlocked by interlock *Y*" Experts of each system need to see a trace of how their knowledge has been applied. Furthermore the behavior of the deductive algorithm has to be evaluated beyond debug level.

4 Accuracy of the system

The Expert System is under continuous evaluation by experts. The accuracy of its predictions are compared with the outcomes of actual interventions and events in ATLAS. To illustrate this we consider the following two scenarios.

In scenario one, as shown on Figure 6, there is a simulation prior to an intervention on which a DSU (Detector Safety Unit), a very critical system, was required to be switched off. The simulation (on the left) foresees that the cooling plant of IBL (Insertable B-Layer, part of the pixel detector) would remain on. In reality (on the right), the switchboard FCTIR-00060 was switched off unexpectedly and subsequently all its dependent systems including the IBL cooling plants went to off. The result of the later investigation was that FCTIR-00060 was equipped with an interlock from the DSU2 that was not registered in the Knowledge Base. Because of this type of interlocks work in a positive logic, FCTIR-00060 was switched off unintentionally.
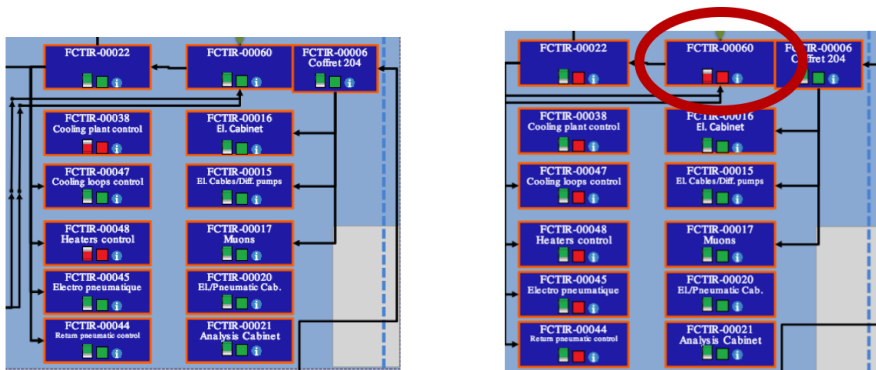

Figure 6. Scenario 1. Expectation (left) and reality (right).

In the second scenario, as shown on Figure **??**, there is a simulation before an intervention that required switching off rack Y.38-23.X0. The procedure was to switch off the electric distributor EXD21/15X to work on the rack. The Expert System simulation successfully predicted that switching off EXD21/15X would also affect quadrant 3 of the SCT (Semiconductor tracker) and the Pixel detector (PIXEL_Q3).
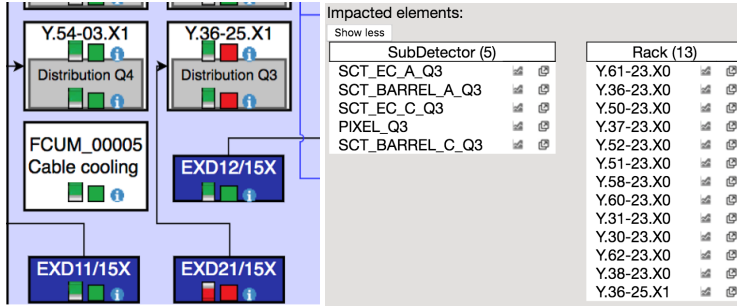
Figure 7. Scenario 2. EXD21/15A is switched off (left). Impacted elements list with Y.38-23.X0 and Q3 of SCT and Pixel (right).

5 Risk Analysis

Let P_s be the probability of success defined as the odds of a system of accomplishing its assigned task [2] and the probability of failure P_f is calculated as $P_f = 1 - P_s$. In the following paragraphs we will assume that P_s of individual components can be inferred from the Knowledge Base. We build a functional block diagram for a system as a fault tree in which all elements affecting the reliability of the system under study are represented as nodes with a given input and output. We will define the P_s of a given system as the composite P_s of all the nodes. We can distinguish the nodes that are required (in series) from those that only require one of its siblings to operate (in parallel).

The P_s^s of a system in series of X_i components is the product of the $P_s(X_i)$ of the components as described in equation 1

$$P_s^s = \prod_{i=1}^{n} P_s(X_i) \tag{1}$$

For systems in parallel the P_s^p is defined as the complementary of the dot product of the complementary of the components, as described in equation 2.

$$P_s^p = 1 - \prod_{i=1}^{n}(1 - P_s(X_i)) \tag{2}$$

We deduce the principal components of a given system by calculating the probability of failure of all its nodes. The calculation is done repeated times, on each the reliability of a different node is reduced. This way we can deduce which is the principal component of the system. While we do not have an individual P_s for each system we assigned one per type of system. Then, performing a probability of failure analysis on every system in the knowledge base we see interesting results. In a sample of 1762 entries with a mean of 96.2%, the object representing the switchboard FCTIR-00060 has a probability of success P_s of 46.63% with a p-value of 3%. Although each system has been assigned with an arbitrary P_s, one interesting observation is our calculations agree with the ATLAS records. Systems with history of being more problematic are indeed scored with lower P_s in the analysis.

6 Conclusions

An Expert System of the ATLAS experiment has been developed by the Technical Coordination group. The descriptions of the ATLAS components like sub-detectors, electrical supplies and distribution, gas systems, cryogenics and cooling systems have been implemented in the Expert System in the form of an interactive simulation that emulates their behaviour. The system is able to explain the results of the simulations which are deterministic. These results and explanations can be reviewed by experts to analyse how components and their relationships affect the system as a whole. It is capable of explaining and foreseeing events with complex dependencies and it is becoming an important tool for ATLAS operations. The granularity of the descriptions continues to be improved.

References

[1] ATLAS Collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider* (JINST, 2008) 437 p

[2] Albert Myers, *Complex system reliability* (Springer, London, 2010) 238 p

[3] Jones, R and Mapelli, Livio P and Ryabov, Yu and Soloviev, I, IEEE TNS **45** 1958-1964 (1998)