**Introduction**

The Trigger and Data AcQuisition (TDAQ) system for the ATLAS will have a significant number of users each with a set of responsibilities and rights within the system.

The Access Manager (AM) is responsible for authorizing users based on their rights in the system and to stop actions that are not authorized. Currently the scope of the component is limited to the CORBA part of the TDAQ.

**Design**

The CORBA protocol is being used for interprocess communication in the TDAQ system and provides a basis for building the AM component. Its responsibility is to authorize all requests from client to a server. The AM component is divided into 3 main parts:

1. **Client side**
   - Consists mainly of the AM interceptor. To be able to perform the authorization transparently to both client and server CORBA interceptors are being used. The interceptors are basically functions that will be called by the CORBA framework at set points in the CORBA call chain.

2. **AM server**
   - The AM server is responsible for authorizing a user for a given action.
   - Has access to the database describing all the rights within the system.
   - Receives authorization requests from AM interceptors and returns result.
   - To improve performance the AM server loads all data into memory at startup.
   - Hash tables are constructed to increase the speed of lookups in the data.

3. **Database and Administration**
   - Consists of database of permissions and tools to administrate these. The database is currently implemented in a MySQL dbms. A graphical tool to modify the database has been developed using the Qt® framework.

**Access Control Model**

The RBAC (Role Based Access Control) model is used to best reflect the working structure in the TDAQ. In the RBAC model access to resources are based on the role of the user.

A role typically represents a job function within the system (e.g. a shifter) with some associated permissions. Users can be assigned one or more roles and are authorized for an action if it is member of a role with the appropriate permissions.

Advantages or RBAC:
- Flexible and adaptable system
- Easy to administrate
- Naturally reflects the organization structure

**Program flow**

A client sends a CORBA request → The request is intercepted by the AM interceptor → Does the action require any authorization? → No → Request is sent to server → Is there a valid result in the cache? → Yes → Authorization request sent to AM server → AM server checks permissions and returns result → Was action authorized? → Yes → Authorization granted → Request is stopped

Access Control in the ATLAS TDAQ

John Erik Sloper, Marius Leahu, Marc Dobson and Giovanna Lehmann

ATLAS TDAQ Controls Group, CERN

**Conclusions and outlook:**

We have managed to build a flexible access control system while keeping the performance and administration overhead at a minimum. The use of CORBA interceptors has made it possible to implement the AM transparently for both the clients and servers in the system, and thus minimal changes to the existing system have been necessary. The RBAC model has proved to be a good choice for the TDAQ system. It lets us properly represent the working structure while keeping the system flexible.

The scope of the AM is currently limited to the CORBA part of the system. Plans to extend the functionality and widen the scope has been made, and work on this will start up mid 2005.

**Contact information**

John Erik Sloper – Bergen University College/CERN John.Erik.Sloper@cern.ch
Marius Leahu – CERN Marius.Leahu@cern.ch
Marc Dobson – CERN Marc.Dobson@cern.ch
Giovanna Lehmann – CERN Giovanna.Lehmann@cern.ch

**References**