Models of Networked Analysis at Regional Centers for LHC Experiments - MONARC

- Initiated in the fall '97 (K.S); brought into existence as a joint CMS/ATLAS Project by the ATLAS World-Wide Computing Group and CMS Computing Groups (H. Newman and K. Sliwa)
- Approved by LCB in December '98
  - Spokesman — H. Newman
  - Project Leader — L. Perini

  (plenty of ATLAS vs CMS politics involved)
- Work divided into 4 working groups:
  - Analysis Group — P. Capiluppi
  - Architecture — J. Butler
  - Simulation and Modelling — K. Sliwa
  - Test Beds — L. Luminari
- Existing simulation tools developed by Iosif Legrand (CERN), together with K. Sliwa and (from June '99) Alex Nazarenko
- MONARC is viewed by LCB as the body in which Regional Center related issues will be studied and decided upon – regular meetings with Regional Center directors; next one August 26, 1999
- MONARC will be (most likely) extended to Phase-3, which will focus on architectural issues and LHC experiments

DESIGN ELEMENTS OF THE MONARC SIMULATION TOOL

ACTIVE AND PASSIVE JAVA OBJECTS

Java allows multi-threading of objects. This feature makes it a very attractive environment to simulate concurrent, data driven, processes, like DAQ or data transfers between processing nodes and data server nodes. Provided that one develops an independent scheduler (*to keep track of flow of time*) Java seems an ideal tool with which to build discrete event simulations of distributed computing systems.

CPU, MEMORY, TAPE and DISK are passive objects. JOBS and LINKS are active objects. DISK, MEMORY and CPU are now simulated with realistic granularity. Data containers with different data are assigned to specific data servers (AMS servers) and an elaborate book-keeping system simulates the Objectivity/DB database catalogue. JOBS can be submitted to the queues, but they will only be executed if sufficient MEMORY and CPU is available. No memory swapping is allowed.

DESIGN ELEMENTS OF THE MONARC SIMULATION TOOL

ACTIVE AND PASSIVE OBJECTS

Any activity involving data is described by JOBS. There are data processing JOBS (RAW->ESD, ESD->AOD, AOD->TAG) and analysis JOBS (TAG, AOD, ESD, RAW), each with different CPU and MEMORY requirements. Each JOB may invoke input data and output data, NULL is allowed, which means that almost all activities can be described in the JOB framework (like e-mail and Video-conferencing, although this has not yet been implemented).

NETWORK is described in an "interrupt" driven way. For each new message an interrupt is created, which triggers a re-calculation of the transfer speed and the estimated time to complete a transfer for all active objects. Such a scheme provides an efficient and realistic way to simulate concurrent transfers using even very different object sizes and protocols. Logically, there is no difference in the way LANs and WANs are simulated.

A multi-tasking processing model for shared resources (CPU, MEMORY, I/O channels) has been implemented. It provides an efficient way to simulate multi-tasking and I/O sharing.

DESIGN ELEMENTS OF THE MONARC SIMULATION TOOL

DATA MODEL:

As envisaged in Computing Proposals of LHC experiments, all data is organized in objects and managed within the framework of an ODBMS database. In our case we consider specifically an Objectivity/DB federated database, which allows for distributing sets objects onto different, geographically and physically, media (tape, disk…) and data servers (AMS Objectivity) servers, while maintaining a coherent and logically uniform view of the entire distributed database. Various objects can contain pointers to each other, associations, which allow for navigation across the entire database. Each object has an object identifier (OID), which has the following structure, providing its absolute address in the database.

database id - container id - database page - logical slot

There is no object catalogue, per se, in Objectivity. The *database catalogue,* residing on a single AMS server per autonomous partition of the federated database, contains the translation between the *database index* and the physical database files. Each database file contains the translation between the *container index* and the physical container files. Each container provides a translation between the *database page index* and the physical database pages residing in a given container, and so on.

DESIGN ELEMENTS OF THE MONARC SIMULATION TOOL

DATA MODEL (continued)

We consider 4 functionally different groups of objects:

- RAW data; about 1 MB/event; most likely to be written on tape (unless it becomes financially feasible to put all RAW data on disk – recent Hitachi/UK universities technological advances may indeed make tape obsolete)

- ESD data; (Event Summary Data) – objects with reconstructed data; about 0.1 MB/event

- AOD (Analysis Object Data) – a subset of ESD, perhaps non-overlapping which is possible using associations; about 0.01 MB/event

- TAG – a small set of essential information describing an event (jet and lepton multiplicity, trigger masks, E_t of three highest transverse energy jets and leptons, et cetera; about 0.0001 MB/event.

- Collections – like TAG but just for a subset of all event, could also be a list of pointers to TAG objects corresponding to a given pre-selected physics dataset (for example a standard electron or muon+jets sample)

DESIGN ELEMENTS OF THE MONARC SIMULATION TOOL

DATA MODEL (continued - 2)

Assuming 1 PB/year/LHC experiment; the ESD sample will require 100 TB of storage, the AOD – 10 TB, the TAG – 100 GB of storage. CERN site will have all the data, various regional centers all TAG and AOD, and subsets – or all – ESD.

It is anticipated that analyses will start from COLLECTIONS and TAGS, allowing a user to navigate across the AOD and, if requested, to the ESD and even the RAW data objects.

Different analyses may require different data access patterns, for example different fraction of data will be accessed following the TAG->AOD, AOD->ESD associations, et cetera.

The model must be flexible and powerful enough to allow for almost arbitrary data access pattern to be reproduced.

DESIGN ELEMENTS OF THE MONARC SIMULATION TOOL

IMPLEMENTATION OF THE DATA MODEL

A single integer number, corresponding to "event number", is the basis of all book-keeping within the simulated Objectivity database. The smallest unit of database operations is the container. Locking takes place on the container level as in real Objectivity database. Given the number of events for which the simulation is to be run, and the data type (RAW, ESD etc.), *each container* of a given data type *is assigned a range of event numbers* which belong to that container.
At present, the container sizes vary with the data type, i.e. RAW data containers are larger than TAG containers, and for each RAW data container there exist a corresponding ESD, AOD and TAG container mapped to the same event number range. Hash tables are used to provide this mapping, which simulates the Objectivity catalogue. Another set of hash tables provides the mapping between the containers and the AMS servers.

Every submitted JOB declares the range of event numbers it will process. In the current scheme, it is trivial to quickly find out which containers will be needed to process the JOB. The corresponding lists of needed containers are created, and handled by the scheduler. The transfers of data are requested between the AMS servers (local- where the CPU on which the job is ran, and the one which has a needed container assigned to it). If writing is taking place, the locking mechanism is invoked.

DESIGN ELEMENTS OF THE MONARC SIMULATION TOOL

IMPLEMENTATION OF THE DATA MODEL (cont.)

Since data transfers take place via a client-server mechanism on the AMS level, one can easily simulate a regional center by simply declaring that a number of AMS servers belong to it.

A clustering parameter defines what fraction of event number range, f, is to be processed. If f=1.0, all events will be read, simulating the sequential access. If f<1.0, only some events will be processed (at random). Three parameters are used to describe data access patterns, namely, the frequencies with which TAG->AOD (a), AOD->ESD (b) , ESD->RAW (c) associations are to be followed. With these 4 parameters, f,a,b and c, we believe that even the most intricate analysis models and data access patterns, as well as very different clustering schemes can be easily simulated.

WHAT CAN BE MODELLED NOW ?

i. EXISTING CLASSES OF JOBS

Reconstruction	RAW->ESD
creating AOD	ESD ->AOD
creating TAG	AOD->TAG
analysis:	starting from TAG, and following the TAG>AOD, AOD->ESD and ESD>RAW associations, as defined in the data model

fraction -parameters describing clustering of the data in database containers

a%	-	fraction of events for which the TAG->AOD associations are followed
b%	-	fraction of events for which the AOD->ESD associations are followed
c%	-	fraction of events for which the ESD->RAW associations are followed

WHAT CAN BE MODELLED NOW ?

Jobs may run concurrently in a single or several regional centers, each with different hardware configuration and connected with CERN over different networks

DISK which used to be a generic entity is now divided into the disk servers (AMS servers);

CPU is also divided into processing units, each with its own memory (allows to limit the number of jobs depending on the amount of memory they use)

TWO EXAMPLES EXIST AT PRESENT:

reconstruction (RAW->ESD) at CERN

analysis of data in OODMS (with following TAG->AOD, AOD->ESD and ESD->RAW associations) which can be performed in N sites

WHAT CAN BE MODELLED NOW ?

i. Validation of the basic elements of the model (AMS and network has to be performed

ii. Simple analysis involving distributed databases (with replicas of some data with and data model which use associations) should can be both modelled and measured at the test-beds

iii. Results of Iosif's simulation program could be compared with results of other simulations (for some existing system)

iv. Several scenarios of LHC-era analyses can be simulated right now:

Fully centralized scenarios:

a) all data at CERN; all users at CERN using the central site CPU and disk

b) a variation in which analysis is performed on users' workstation CPU and with some data stored on user workstation disks

WHAT CAN BE MODELLED NOW ?

Partially de-centralized scenarios:

a) RAW and ESD data at CERN; AOD and TAG at all centers; users perform analyses using the CERN and Regional Center resources (CPU and disk)

b) a variation in which analysis is performed on users' workstation CPU and with some data stored on user workstation disks; with the rest of the data accessed via associations from the Regional Centers (AOD) or CERN (ESD, RAW)

c) a variation of either of the above scenarios in which some ESD is also replicated to Regional Centers

d) obviously one could study arbitrary hardware and network configurations

MONARC MILESTONES FOR PHASE-2

- Completion of first cycle of simulation of possible computing models for LHC experiments. First classification of the models and first evaluation criteria - July '99
- Completion of coding and validation for second-round models - August '99
- First results of validation available - September '99
- First results of Model Comparison available - September '99
- Completion of a second simulation cycle of refined models for LHC experiments - November '99
- Guidelines for experiments - November '99
- Delivery of Phase-2 deliverables (tools, documentation…) and presentation to LCB of a proposal for the continuation of MONARC - December '99