

JCOP Project Team Meeting

Thursday, March 11th, 2004 @ 10.30

There were about 37 persons present at this meeting.

1. Minutes of the previous meeting

There were a no comments to the minutes of the last meeting.

2. News

There were two points of news:

- Manuel reported that a new version of the FW is available since Tuesday. He also informed the meeting that there had been a decision to produce new versions every two weeks. This will require extra testing effort and so he is interested in automated testing tools. However, there is limited experience in the group and he requested help and advice from anyone with experience in this area.
- Wayne reported that he had received some test results from the LOFAR project, which is building a huge radio telescopes array centred in the Netherlands. They are looking at scalability same issues and have been testing communications throughput using 110 computers running PVSS pre-release V-3 with 10 systems generating data and 100 consuming it. They have a report showing that high bandwidths with stable operation. They also stated that building the system was straight forward.

3. The Finite State Machine toolkit of the JCOP Framework – Clara Gaspar

Clara gave a detailed overview of SMI++ and its integration with PVSS and the JCOP FW. An abstract of her presentation is given below.

“SMI++ is a tool to build hierarchical, distributed, control systems. It allows for the sequencing and automation of operations. This tool has been integrated in the JCOP Framework, in order to implement the partitioning rules and to allow users to describe and implement the behaviour of their (sub)systems. This talk will explain in depth how SMI++ works and how it has been integrated with PVSS and the Framework.

Even though the tool can be used just by pointing and clicking for very small applications, when integrating several sub-systems together and in order to implement more complex behaviour a better knowledge of the underlying SMI++ mechanism is necessary. The talk is particularly aimed at such users.”

Her slides are available at:

<http://agenda.cern.ch/askArchive.php?base=agenda&categ=a041114&id=a041114s1t4/transparencies>

The following questions were raised:

- Can an SMI++ domain in one machine be started from another machine? This is currently not possible with the tools provided by Clara. She will consider this for the future.
- What granularity should one aim for a DU, e.g. individual HV channel or HV system? Clara’s recommendation would be not to exceed 500-1000 DUs for any one CU.
- Diagnosis possibilities were announced as a future development. What possibilities exist currently, say if a DU or CU were to block? From the UIM it is possible to see which DU/CU is blocking as this will remain greyed out. Clicking down the hierarchy to the lowest level object greyed out will identify the problem area. As the SML code is relatively simple to read and understand it should then be possible to see if this is a programming error or a rather a problem with the process.
- What happens if several commands or state changes arrive at a CU at the same time? If the Architecture Working Group (AWG) rules are followed it would not be possible for several commands to arrive at the same time. However, in either case the commands or state changes would be queued and processed in sequence.

- Can one send a parameter up the hierarchy at anytime? A parameter would only be sent with an associated state change. However, one could resend the current state with the desired parameter.
- Can one include a set of When clauses with the #Include statement? Yes.
- Clara had stated that a CU with 500 associated HV channel DUs had taken 5 second to process a single command to all of them without any H/W connected. Sorina stated that she had a similar performance for only 26 HV channels with the real H/W connected. This implies that the bottleneck would not be in SMI++/PVSS but rather in the communication with the H/W.
- Does the use of SMI++ not double the traffic in PVSS? Most of the traffic would be present even if the command had been sent directly from a UIM with no SMI++. However, it is true that PVSS also has to derive and store the state of the DU which is an additional load. However, this also then provides additional functionality. Hence, the additional load is simply the price to pay for additional functionality gained though the use of SMI++. Reproducing this functionality in PVSS would certainly produce a far heavier load on PVSS.
- If a DU or CU is included in a hierarchy which is being controlled globally can it be commanded directly from a local UIM? No.
- If one is using the FSM/PVSS integration is it necessary to have PVSS systems connected via the distribution manager as inter-system connection is effectively handled via SMI++? Although this might be the case for the control, it is not the case for monitoring. Hence, the distribution manager connection is still essential.

4. A.o.B.

There were no items under A.o.B.