(Java) CA Client Libraries Status

Matej Šekoranja

matej.sekoranja@cosylab.com

Agenda

- JCA update
  - Set of Java CA interfaces
- JCA/JNI update
  - Java CA implementation calling C++ CA library via JNI
- JCA/CAJ update
  - Pure Java CA implementation

QuickTime™ and a
TIFF (Uncompressed) decompressor
are needed to see this picture.

- WebCA (browser CA plug-in)

- Java Channel Access Server (JCAS) talk in the afternoon

JCA update

- Priority support added.
- Support for Alarm Handler ACKT/S puts.
- QueuedEventDispatcher improvements
  - deadlock detection
  - queue blocking added (to enable CA flow control)

JCA/JNI update

- Priority support added.
- Support for Alarm Handler ACKT/S puts.
- 64-bit platform support added.

- JVM memory leak fix/workaround
  - 1MB/s memory LEAK under decent load.
  - Bug in JVM
    - 1.5.0_06 and later including JDK 1.6.0 builds suffer from this bug
  - Clean solution for 3.14.9, workaround for earlier versions.
- Ported CAJ test to JCA/JNI
  - Discovered and fixed some (copy-paste) bugs in the code.
- Random context initialization failure bug fixed.
- Global message callbacks fixed.

JCA/CAJ update

- Priority support added.
- Support for Alarm Handler ACKT/S puts.
- Java CA repeater implementation.
- Channel reference counting implemented
  - CAJ shares Channel instances within one context.
- Allow monitor creation before Channel is really connected.

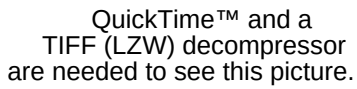
- CAJ connection/search algorithm reimplemented
  - To mimic new 3.14.7 implementation.
- CA flow control implemented.
- Behaviour of auto_addr_list fixed.

- Discovered an IOC deadlock (fixed in latest EPICS base).

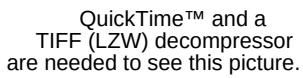
What is WebCA?

- WebCA is an internet plugin that empowers your web browser as an EPICS CA client
- Supports Netscape plugin compatible browsers:
  - Mozilla (e.g. Firefox)
    - Linux, Mac OS X, Windows
  - Safari
    - Mac OS X
- How it differs from Canone, etc…?

WebCA Architecture
(slide from Tom Pelaila's presentation)



Screenshot (a few lines of code, out-of-the-box)



WebCA Future Effort

- Create an installer for easy installation
- Develop controls, meters and graphs

- Develop an EPICS display language in XML along with XSLT transforms to generate the web page content
  - Provides a consistent look and feel
  - Quicker page development

Acknowledgments

- SNS/ORNL team
  - Tom Pelaia
  - David Purcell
  - Ernest Williams

Thank you!