Stanford Linear Accelerator Center

New SNL Ideas

Possible Next Steps

Ron Chestnut – PSI, May, 2001

With ideas from B. Dalesio and M. Clausen

Already new in 3.14

M. Kraimer, Eric Norum, Janet Anderson

- Works on Solaris, RTEMS, Linux, Windows too?
- SEQ shell now optional
  +m option for main() generation
  -i option for suppressing ioccrf registration (also C++ implications)
- Minor changes to distribution

SNL User Variable Server

- Must support dynamic load, kill, and reload
- Use “Serve” verb in place of “Assign”
- Add “Metadata” specification for use by Serve
- Need SEQCA task to serve variables for all SNL programs on an IOC

Syntax

Metadata q {Precision=3,Display Low=10,Engineering Units="Erg",MALM=20};

Metatadata r {Display High=200};

Float x; Serve x as "snl:xyz:good" using q;

Long y; Serve y as "snl:abc:ok" using r;

Double z[20]; Serve z as "snl:array" using q;

Normal macro replacements supported as well.

Metadata

- English, not database abbreviations
- Display High/Low
- Precision
- Engineering units
- Waveform lengths
- Any ENUM data
- More???

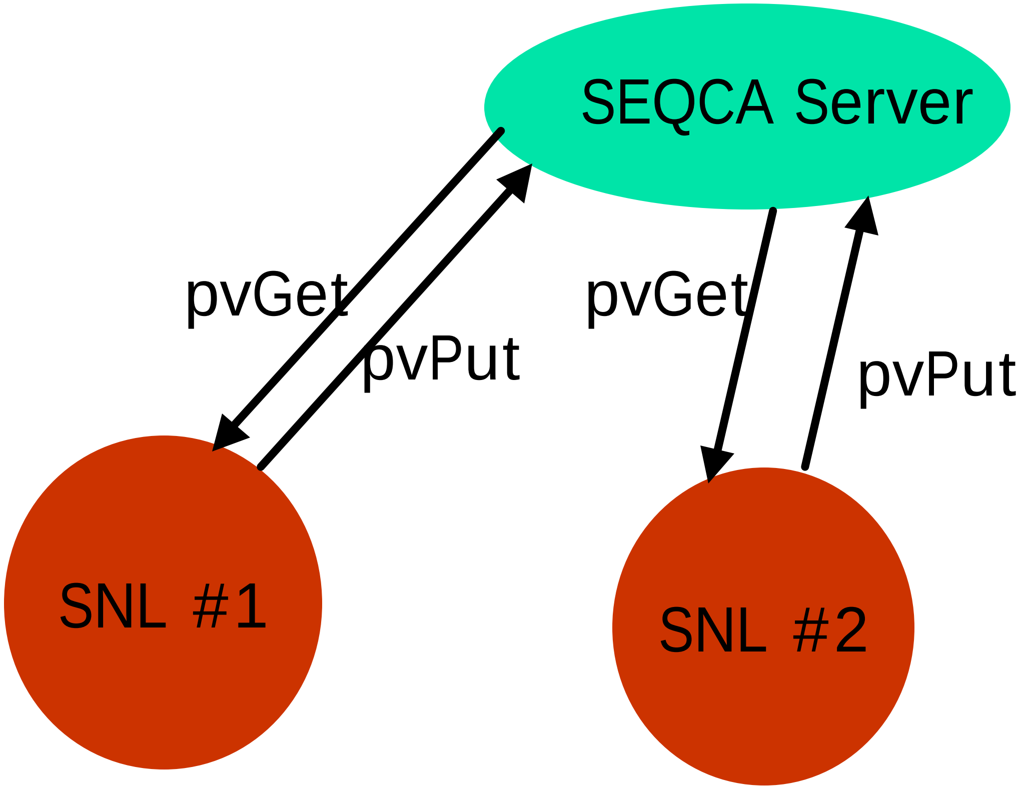
More internal functions

- pvServeCount (like pvAssignCount)
- pvServed (like pvAssigned)
- pvServe (like pvAssign)
  This runtime capability would probably be deferred until later.

SEQCA – the real server

- Register data from initializing SNL programs
- Communicate with individual SNL programs via pipes?
- Clean up on SNL crash or disconnect
- All updates via pvGet and pvPut for served variables (no monitor!)
- No explicit reference to individual fields

SEQCA – a picture



Work Involved

- Do a real design
- Syntax support, code generation
- SEQCA task development

Serving SNL Internals

- Some data now available only through seqShow *taskid* and seqChanShow *taskid*
- Name, #state sets, #channels, #assigned, #served, #connected
- For each state set: Name, current state, previous state, elapsed time since transition
- For each channel: Name, Unexpanded name, Assigned?, Connected?, Served?

Example names for internals (better ideas gladly taken)

- *IOC_name*:SNC:nprograms (integer)
- *IOC_name*:SNC:program (array of strings)
- *IOC_name*:SNC:program:nSS (array of ints)
- *IOC_name*:SNC:program:nchan (array of ints)
- *IOC_name*:SNC:prog_*N*:*SS*:name (array of strings)
- *IOC_name*:SNC:prog_*N*:*SS*:cur_state (array of strings)
- *IOC_name*:SNC:prog_*N*:CH:name (array of strings)
- *IOC_name*:SNC:prog_*N*:CH:assigned (array of ints)

Where _*N*,*SS* and *IOC_name* are automatic

Mechanics of serving internals

- SEQCA again does all the real work
- Add table similar to user variable table for accessing internals OR
- Append internals table to user variable table.

Both require restructuring of structures containing the internals' data.
This is all, of course, read-only.

Discussion

- This all seems doable – just a matter of time (Cubs win series? Sun burns out?)
- Need feedback
- We can possibly do the syntax and internals changes at SLAC. Others need to design and implement SEQCA.