EPICS MODIFICATIONS ENABLING LUA – BASED DATA PROCESSING SUBSCRIPTION UPDATE FILTERS

Jeff Hill

DATA PROCESSING LUA SUBSCRIPTION FILTERS – LUA, A BRIEF INTRODUCTION (REVIEW)

- Lua, a Brief Introduction (review)

- EPICS Integration of Lua milestones (review)

- Data Processing Lua Subscription Filters, Motivation

- Data Processing Lua Subscription Filters, Implementation

- Conclusions

DATA PROCESSING LUA SUBSCRIPTION FILTERS – LUA, A BRIEF INTRODUCTION (REVIEW)

- Lua *embeddable* language was created in 1993

  - By members of the Computer Graphics Technology Group (Tecgraf) at the Pontifical Catholic University of Rio de Janeiro, in Brazil.

- "Lua" (pronounced **LOO-ah**) means "Moon" in Portuguese

- Interpreted, compiled at load-time to byte-code

- A mixture of C-like and Pascal-like syntax

- Dynamic typed, automated conversion between string and numeric types

- Efficient virtual machine execution, small footprint, incremental garbage collection, easily interfaced with C code

- Liberal MIT license

- Some negatives also, see my talk at Michigan EPICS meeting

  - In particular, variables are globally scoped by default

DATA PROCESSING LUA SUBSCRIPTION FILTERS – EPICS INTEGRATION OF LUA MILESTONES

- Lua 5.2.3, the current release, embedded inside of EPICS base

  - Built by the EPICS build system

- Lua 5.2.3 has the upgraded support for integer primitive types

- The current released version of Lua is now at 5.3

DATA PROCESSING LUA SUBSCRIPTION FILTERS – EPICS INTEGRATION OF LUA MILESTONES

- Lua based subscription filtering in the CA server

  - Event queue is order correct

  - Based on C++ 11 shared pointer

    - Subset of boost included in EPICS base supporting prior compilers

  - Based on Data Access abstract base class

    - Interface is independent of data source implementation

DATA PROCESSING LUA SUBSCRIPTION FILTERS – EPICS INTEGRATION OF LUA MILESTONES

- Lua based subscription filtering in the CA server

  - Filters specified in channel name postfix

    - Invoking Lua methods supplied when the IOC boots

  - Each client attaching to the server

    - Instantiates an independent Lua context

DATA PROCESSING LUA SUBSCRIPTION FILTERS – EPICS INTEGRATION OF LUA MILESTONES

- Alternative EPICS SHELL

    - In contrast, a fully functionality scripting language

        - Powerful libraries, built-in and community

- An environment well proven for use in

    - Configuration

    - Scripting

    - Rapid-prototyping

DATA PROCESSING LUA SUBSCRIPTION FILTERS – EPICS INTEGRATION OF LUA MILESTONES

- Currently we have two computational record-level building block components
  - EPICS calc record
    - Excellent rapid prototyping, but limited functionality
  - EPICS subroutine record
    - Excellent efficiency, but possibly less popular for rapid prototyping
- A new Lua based record provides
  - Comprehensive functionality set
  - A reasonable compromise runtime execution efficiency
  - The rapid prototyping we depend on with the calc record
    - Upgrade in-place
      - Runtime code updates via CA puts to lua record fields
  - And, hopefully the heavy lifting comes for free with Lua

DATA PROCESSING LUA SUBSCRIPTION FILTERS – EPICS INTEGRATION OF LUA MILESTONES

- IOC's registrar enhanced to allow registration of

  - C object code embedded Lua code

  - Lua interfaced C code

- Facilitate these components to be instantiated into Lua contexts when they initialize

  - EPICS Lua IOC Shell per-shell private Lua contexts

  - EPICS Lua record per-record private Lua contexts

  - EPICS CA server per-client private Lua contexts

- Use C++ shared_ptr for life time management of read-only Lua byte code chunks

  - Less overhead, no Mutex required

DATA PROCESSING LUA SUBSCRIPTION FILTERS – MOTIVATION

- At LANSCE, in addition to gate flavored subscriptions, we need
  - Application specific data attributes conveyed from server
    - To application specific CA clients
      - Defining server-to-client private application specific protocol
  - Conveying
    - An array time-slice, specified by channel name postfix
      - *Offset,* from gate rising / falling edge, time delay units
      - *Width*, time delay units
    - Bit mask identifying Array active beam gates when the data were captured
      - Implemented by inserting an additional array element
    - The status of the filter request
      - Implemented by inserting an additional array element

DATA PROCESSING LUA SUBSCRIPTION FILTERS – IMPLEMENTATION

- Lua wrapper objects for Data Access generic interfaces
    - Number, Integer, Boolean, String, TimeStamp, container (Catalog), Array, Nill
    - Enclosing
        - The data, or a reference to an Array interface or container (Catalog) interface
            - Array interface publishes element sequence with bounds
        - Reference to Catalog of subordinate properties
    - Property hierarchy traversal via Lua "dot" indexing
- Filters previously returned only {false,true}
    - False suppresses update, true sends update
- Filters now optionally return also {Nill, Data Object}
    - Nill return suppresses subscription update
    - A returned data object is proxy delivered in the CA subscription update payload
    - Windowing and array element insertion implemented by C based resequencing Lua snap-in
        - This does *not* result in *reallocation* of space for array or array *copying*

New

DATA PROCESSING LUA SUBSCRIPTION FILTERS – STATUS

- Lua features described here are in Bazaar branch

  - lp:~johill-lanl/epics-base/server0

- Lua features described here as new

  - Development branch, in-progress

  - lp:~johill-lanl/epics-base/server1

DATA PROCESSING LUA SUBSCRIPTION FILTERS – CONCLUSION

- LANSCE has implemented a comprehensive integration of Lua into EPICS base

  - Lua based EPICS shell

  - Lua script record

  - Registry loaded Lua chunks, Lua interfaced C code

  - CA server Lua subscription update filtering / data processing

    - Data Access Array is a Lua object

    - Filter optionally returns proxy data object delivered in data payload to client

New