Observation Control Systems – the Onsala Perspective

Mikael Lerner

Florence – 7 October 2015
Outline of presentation

- Personal background
- The Pegasus control system
- Thoughts on a new control system
- Towards the BIFROST control system
Personal background

- Received a Ph.D. in (numerical) astronomy at Onsala 1998
- Responsible for Pegasus control system at SEST 1999-2003
- Responsible for CIMA control system at Arecibo 2004-2009
- Responsible for Pegasus control system at Onsala 2009-present (two telescopes + ozone monitoring station)
Pegasus history

- Developed late 80s-early 90s at CFHT (but no longer used there)

- Imported to Onsala in early 90s and modified for radio telescopes

- Imported to SEST 1998 (used until closure 2003)

- Used at Onsala for 20m-telescope, 25m-telescope and ozone monitor station (the latter system replaced 2012)
Pegasus structure

- Mainly written in C using its own graphic library based on X
- A few framework programs are running continuously (main menu, feedback, communication daemons)
- Most programs are independent C-programs or shell scripts called from the main menu
- Information passing between programs is mainly done via variables in *par-files* (ASCII text files)
- Graphic layout is also defined in the same *par-files*
- Graphics is quite limited (B/W only, no lists or menus, no dynamic elements) forcing the use of other graphic solutions for more demanding windows (Tcl/Tk at SEST, Qt at Onsala)
Pegasus pros

- Fully menu-based GUI
- Simple, minimalistic graphic design
- Easy to learn and use (important since there are no telescope operators)
- The *Job*-facility provides means for simple scripting (for example pointing jobs)
- The map editor provides GUI allowing easy design of arbitrarily shaped maps
- Integrated quick-look display for the data
Pegasus cons

- Scripting possibilities are limited
- Too limited graphics capabilities within Pegasus framework
- Problems with coordination between different programs
- Bad error handling – bad variable values may block windows from appearing
- Difficult to install / no off-line demo-version available – requires modifications in operating system
- Complex structure with lots of “old luggage” that makes most modifications labour-intensive efforts
Wish list for a new control system

- Better scripting capabilities (sequential execution) – later also auto-observing mode (dynamic decision)
- Save configurations with automatic descriptions
- Reconfigurable for remote users (window size, update rate)
- Flexible options for feedback/logging with retroactive debug messages added upon errors
- Auto-sort and create meta-database of data taken
- More advanced quick-look facility with multiple plot windows and auto-configuration depending on data type
- Automatic syntax checking when loading user-generated objects (source lists, command scripts)
- All GUI features must have help texts (on-line manual)
Main design goal

BIFROST should provide everything that Pegasus provides and take user support to the next level.
Main design goals

BIFROST should provide everything that Pegasus provides and take user support to the next level.

I should be able to use the system at 04:00 AM.
Main design goals

BIFROST should provide everything that Pegasus provides and take user support to the next level.

I should be able to use the system at 04:00 AM. (without swearing)
Design goals for BIFROST

- Intuitive, easy-to-use GUI needed
- On-the-fly input error checking
- Logical support (for example don't allow an observation to start if the backend has not yet been configured)
- Provide good awareness (what is the current status of the system)

- Use high-level scripting language
- Provide advanced graphics
- Use standard packages
- Avoid need for customization of operating system
- Be easy to maintain (even for non-programmers)
CIMA as a base for BIFROST

- Don't start from scratch: use CIMA imported from Arecibo as base for new control system

- Modern, flexible, script-based control system (Tcl/Tk)

- Tried and tested in challenging environment (many users, lots of different observing modes, remote users)

- Comes with a lot of desired features that Pegasus is lacking already included

- Large in-house knowledge about the system – no learning curve (major part of CIMA written by M. Lerner)
Some goodies CIMA offers

- Selection of font size (=window size) which is useful for remote observing
- Better command script facility
- Library functions for on-the-fly error checking of inputs
- Multiple versions available in parallel (e.g. new + stable)
- Off-line mode for training or preparation
- Flexible selection of what is logged and shown in the feedback as well as in what format
- *Shamecast*-support (*shared memory multicast*) for getting instrument information
- Existing high-level procedures can be used as templates – thus adaptation instead of designing and writing from scratch
Towards BIFROST

- The BIFROST-platform has been created: a set of Tcl/Tk libraries with a standardized graphic design and special versions of *wish* and *tclsh* (standard Tcl/Tk linked with some extra libraries supporting *shamecasts*, astronomical calculations and Jeff Hagen's *socklib* package)

- Various peripheral systems are built on the BIFROST-platform: instrument *shamecasts*, monitor and web displays, engineering parameter logging, log viewers, alarms

- Gradual transition: several BIFROST-programs already used in Pegasus (spectral line selection, receiver tuning), other systems are BIFROST-compatible (VLBI-daemon)
Current BIFROST status

- The BIFROST-platform is used in a number of engineering systems: instrument monitoring, display windows, alarms
- A Pegasus system used for the ozone station was replaced by two BIFROST systems in 2012 (for parallel 24/7 operation of two independent instruments)
- A BIFROST system for the 25m telescope is ready for commissioning
- A BIFROST system for the 20m telescope could be available in 2016 (depending on priorities)
Observation Control Systems – the Onsala Perspective
Onsala Space Observatory

CHALMERS
Mikael Lerner 2015-10-07 19 / 20
Chalmers University of Technology
Conclusion

- Pegasus has been used at Onsala since early 90s
- Pegasus was innovative for its epoch, but it has limitations and it is now time to move on to the next level
- The CIMA system used at Arecibo has been selected as a base for the new BIFROST system
- BIFROST is expected to become the next-generation control system providing enhanced functionality in many aspects