

Introduction

This document provides an overview of the tcs3 software. Its purpose is to provide a starting point for understanding the tcs3 software and provide points on where to look next for further details. The intended audience is the programmer assigned to support the TCS software. This document focuses on the following applications:

Main – The “main” tcs3 applications. This is the control program for the telescope HA and Dec axis, as well as facility control. The MCC GUI, t3remote, t3io are also part of the ‘main’ source tree.

1. Account, Source Code, and Installation

1.1. Accounts

On the IRTF workstation the **tcs3** account is the development account. This account contains all the source code and is used as the login for development. The account is available via the IRTF NIS tables, so it is available from any IRTF workstation, BUT you should use it on the t1, t2, t1hilo computers.

The T1, T2, T3 computers have a **to** user account. This account is local to each computer (defined in /etc/passwd and home directory located on internal disk). The **to** account is used to execute the TCS3 application on the summit. The TCS3 software is installed under t1:/home/to. Since this is a local directory, we also have a cron job on T1 to rsync the t1:/home/to files to irtfas1:/aux1/home/projects/to. The /home/to/data directory on the T1 computer is a symbolic link to /home/tcs3/data. The /home/to is a local account on T1 so the TCS can be run in any event the other IRTF servers are down. Having the /home/to/data link allows data files to be created and stored on the network disk server (which are duplicated and also backed up to tape).

A **to** user account on the IRTF NIS table also exists. As stated in the above paragraph, this account contains a daily rsync of t1:/home/to. Its purpose is to preserve the contents of t1:/home/to. In event of a disk crash on T1, the /home/to account can be quickly restored. If T1 were to crash, you could put T2 into service by changing the cables, and restoring the IRTFNAS1:/home/to account to T2.

1.2. Source Code

The tcs3 account hosts many applications. All source code is located under /home/tcs3/src. You should note that tcs1 support code is located here (ie, tcsd). Also starcatalog, aka starcat, is hosted by the tcs3 account. The ‘main’ tcs3 control application is located in **/home/tcs3/src/tcs3/main**.

Under main/ you will see a few directories:

Dev/ - is the current development branch of the tcs3.

Eng/ - is a temporary engineering copy (this may or may not exist).

vYYMMDD – Once I freeze or install a version, I would rename dev/ to ‘vYYMMDD’ and install this into the /home/to account..

To keep the number of directories down, some older versions are moved to ~tcs3/src/tcs3/archive/main/

When compiling main, you also need the following to be installed on your system:

/home/tcs3/src/libir1 - copy of denault’s IR1 library (used by a few applications, but I made an effort not to use this in main/).

/home/tcs3/src/slalib – copy of sla library. Patrick Wallace excellence astronomy position library.

/home/tcs3/src/libelsa – Functions that build upon sla, esla stands for enhance sla.

1.3 Installation Notes.

For development, you can run the main/dev/ copy from the tcs3 account:

- Login into your system (t1 or t1hilo)
- Execute the IC, MCC to run the tcs3.

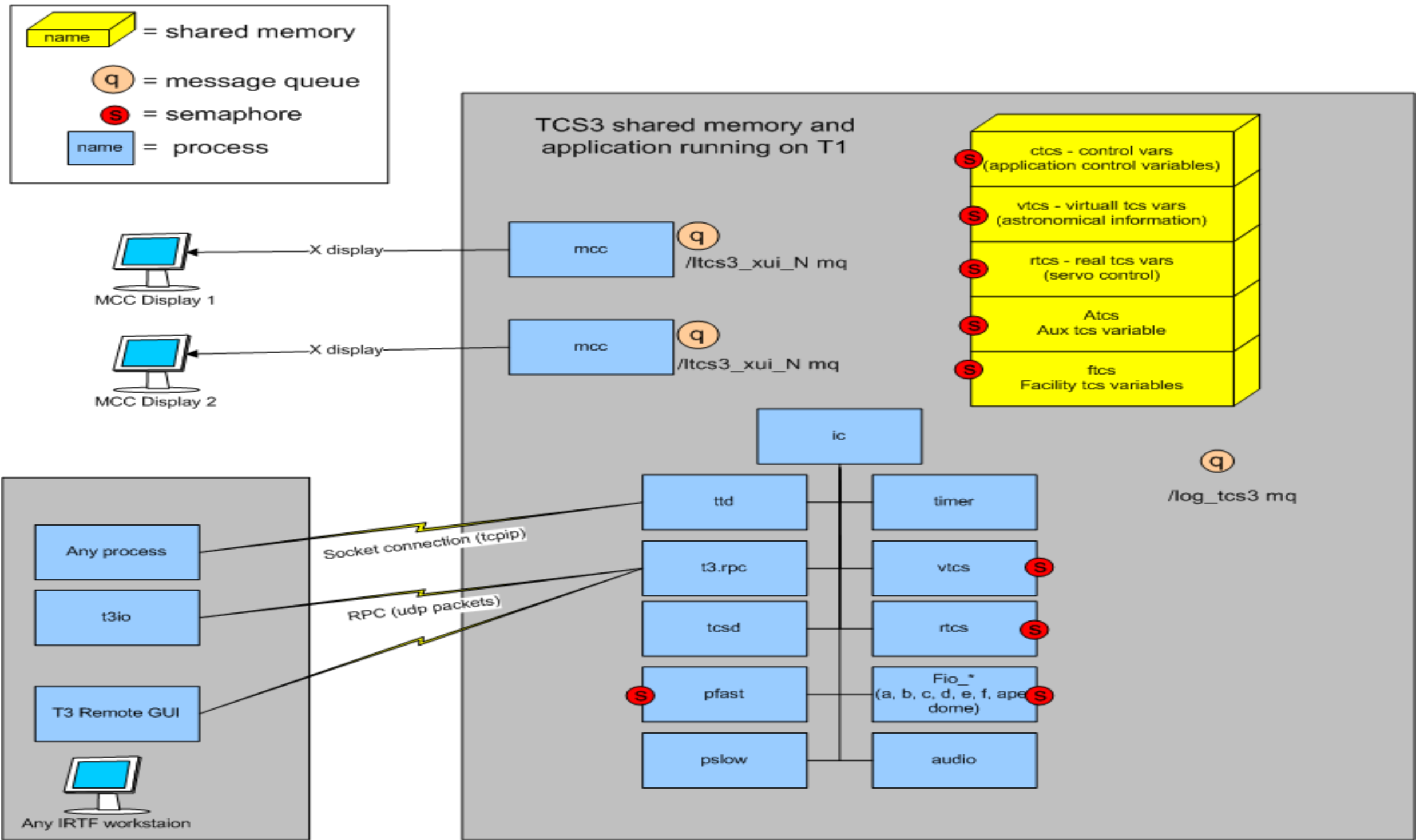
Instruction for installing the binaries to the /home/to account is located in the **main/README.txt**. Review and follow the directions in this file.

2. Main Block Diagram

The TCS3 main/ source tree contains 4 applications:

1. ic – the Instrument Control application. The instrument is the IRTF facility. This application set up the IPC (Inter Process Communication) and launches the various control processes.
2. mcc – The MCC is the master control console Graphical User Interface, or MCC GUI. Multiple GUI can run on the t1 computers, Usually 2 are running and can be viewed on T1's 2 monitors.
3. T3remote – is a graphical client applications that communications with the tcs3 main via RPC (Remote Procedure Calls). This client application is installed on T1 and all the IRTF workstations.
4. T3io – Is client command line application used to communicate command to the tcs3 using RPCs. This program is intended by be execute by other systems wish to communication with the TCS3 (not intended as a user application).

This is a block diagram of the main applications.



2.1 IC Process and IPCs

The IC process creates the following IPC

A struct `tcs_sm_t` is the **IC shared memory area** is created. This area is subdivided in to:

```
struct ctcs_sm_t – control variable for the IC program
struct vtcs_sm_t – virtual tcs shared memory structure
struct rtcs_sm_t – real tcs shared memory structure
struct atcs_sm_t – auxiliary tcs shared memroy
struct ftcs_sm_t – fio tcs shared memory
```

Each of the above structure has a **sem_t guard** variable. This is a semaphore to control access to the shared memory. All process must lock the memory before accessing it. Also included in the `sm_t` are the following semaphores:

```
sem_t vtcs_task_guard;
sem_t rtcs_task_guard;
sem_t pfast_task_guard;
sem_t fio_a_task_guard;
sem_t fio_b_task_gurard;
sem_t fio_c_task_guard;
sem_t fio_d_task_guard;
sem_t fio_e_task_guard;
```

These semaphore are used to trigger execution of the `vtcs`, `rtcs`, `pfast`, `fio_*` tasks. These processes are block on the semaphore until they are scheduled to run.

The `/log_tcs3` message queue is create for the logging function. Any process can attach a text message to be logged by the `tcs`.

A short description of each IC process is given:

- IC – The parent process to the other TCS3 IC processes. The IC creates and initialize the IC IPCS, and starts all the children task. At termination, the IC destroys all IPCs.
- TIMER – This tasked schedules the execution of the child tasks that are triggered using a semaphore. Some processes, like the VTCS are require to run at 20Hz. All the FIO tasks run at 10hz. The timer task controls the executions of these type of periodic task. Other tasks (without semaphores) just run based on of internal process timers.
- VTCS – The Virtual TCS primary performs the astronomical coordinate computation at 20 Hz. It also triggers the RTCS, which usually needs to run after the VTCS data is ready.
- RTCS – The Real TCS is the tasking have communicates with the servo hardware (in our case the PMAC controller). It is also a 20Hz task, primary responsible for handling servo related control functions.
- PFAST – The Process FAST is a 10 hz process that handle period task requiring faster loop. This includes the logging function, and check notices and warning events.
- PSLOW – The process slow task is a 4 Hz looping task that take care of slower periodical function: recomputes slow changing slalib parameters, laser traffic output, position logs, etc.
- FIO_A, B, C, D, E, F – The facility IO hardware for TCS3 are Ethernet based IO unit called Opto22s. We have a number of opto22 referred to as `fioa`, `fiob`, `fioc`, etc. Each opto22 is control by an `FIO_*` task. These task run at 10 Hz and are responsible to communicating with the opto22, reading their input and setting their output. The task also performs some of the control logic for the devices they control. For example, `FIO_C` controls the Counter-Weights, so the counterweight algorithms are located in the `FIO_C` source code.

- FIO_APE – a 5 Hz task that communication with the Absolute Position Sensor Embedded computers.
- FIO_DOME – The dome's position sensor is a serial (RS-232) stream. A task is dedicated to handle this IO (even though the serial input hardware is in opto22 A). This task reads the incoming serial data and interrupts the dome position.
- AUDIO – A 2Hz task responsible for output voice message to the T1 speakers.
- TTD – The TCS Telnet Daemon is a process that accepts connections to the TCS3 telnet port. This allow other to connect to the TCS and access it command interpreter in a TCP/IP session.
- T3RPC – This process provides a RPC interface to the TCS3 commands.
- TCSD – A copy of the old TCS1 daemon is executed under TCS3. It accepts a very small subset of FORTH command and executes their function under TCS3.

2.2 MCC Process and ICP

The MCC provide the Graphical User Interface for the TCS3. They are standalone applications, that require the IC to be running (since they need access to the shared memory). Multiple copies can be started (up to 5). Two are usually running on the T1 graphics monitors. Each MCC creates a message queue to allow the logging function (pfast) to display message on the MCC.

2.3 T3Remote

T3remote is the remote GUI for TCS3. It is installed and executed from the IRTF workstations. It communicates with the TCS3 using its RPC interface. You can review ~/main/dev/README.txt for installation instructions.

2.4 T3io

T3io is a command line utility used to send command and receive reply from TCS3. It is installed and executed from the IRTF workstations. It communicates with the TCS3 using its RPC interface. You can review ~/main/dev/README.txt for installation instructions. This also can be given to visitor so that they can communicate with the TCS, but not have to write RPC support in their applications.