THE GALILEO USER INTERFACE USER'S GUIDE

Paolo Marcucci, Mauro Pucillo
Osservatorio Astronomico di Trieste

Pubblicazione Osservatorio Astronomico di Trieste n. 1454

Abstract

In this document the User's Guide for the Galileo Telescope User Interface is presented. The Telescope User Interface (TUI) is the software module that keeps the communications between the hardware/software telescope system and the user, either an astronomer or a maintenance operator.

A brief description of the Workstation Software System is given, along with some guidelines and principles that governs the TUI.

Introduction

The Galileo Telescope Software System is basically divided in two main areas: **Telescope Local Processors (TLPs)** software (GATE [1]) and **Workstations** software (WSS [2]).

TLP software is loaded on dedicated controllers that "speak" directly with the telescope hardware. Hopefully, the average user is not even aware of the existence of this software: it works in the background, feeding the astronomer with data from the telescope and instruments while controlling the hardware.

Workstation (WS) software, on the other hand, manages the communications among TLP and workstations, among workstations themselves and among workstations and the final user. Several software modules are provided to accomplish this task, and the last one, the user interface, is described in this document.

Version

The current document describes the Galileo Workstation User Interface version 4.0 beta of 30th June 1994.

THE WORKSTATION SOFTWARE SYSTEM

The Workstation Software System (WSS) is composed by a set of seven processes, each dedicated to a unique task. This set of processes, replicated on every workstation connected to the TNG network, takes care of the correct flow of telescope and instrument incoming data (TMVER), of the inter-process and inter-workstation communications (WSCOM), and of user-software interaction (UIF). The last one is composed by a run-time module, called **`display`**, and by two off-line tools: the graphic editor **`(graphic)`** and the table editor **`(table)`**.

The table editor lets the programmer create and manage all the data tables needed for the definition of the entire Galileo Software System. Instrument providers add a new instrument to the TNG simply by adding a new set of tables to the Telescope Database (TDB), and eventually writing an *ancillay process* (APLIB [3]).

The graphic editor allows the programmer to design interactive panels, using the definitions in the TDB, which manages interactions with the end user. Instrument manufacturers could, and actually should, provide specialized panels to control their instruments. In all cases, the panel editor is the only way to build interactive panels that will be used by the **`display`** run-time module.

The User Interface

The TUI actually is the **`display`** run-time module. It runs simultaneously with the other WSS processes and the ancillary processes, providing the user with two basic services:

- data display
- command entry and validation

When the user chooses a set of data to be visualized, the TUI polls the TDB at a constant rate, retrieving values and displaying them on the screen.

The constant rate (actually fixed at one second) reflects the non real-time nature of the Unix operating system, in fact the telemetry parameters are sent to workstations asynchronously, whenever their values change, the same happens to alarms and warnings. The visualization process, instead, has to follow the event-driven nature of the UIF manager process.

This approach leads to a problem: if there are too much data to be visualized on the screen, the system may not be able to perform all data updates in one second (or whatever the update rate is fixed to). So the update process will restart from the first data and the last will be never updated. This is quite a thoretical problem; in fact during the tests with an HP9000/380 (a 68040-based machine with a 25Mhz clock) the problem never arose even when working with a hundred data widgets on screen (and that's a VERY crowded screen). The machine upgrade, made during 1994, to a HP9000/747i (PA/RISC based with a 50Mhz clock) has furthermore differed the

problem.

Principles

The User Interface follows the user's behavior. This is a crucial point in advanced interfaces for large projects. This means that the sequence of operations required to perform any task is under the direct control of the user, rather than accomplished following a predefined set of forms and commands with limited interactivity.

Models

The method of interaction between the interface and the user is called *model*. A model defines the channel that drives the application. For example, the *event-driven* model use the "user's channel" to step through the application; the user presses a button, moves a slider, enters a string, with little constraints. The application can stop the flow of operations only to alert the user about errors or warnings and waits until the user recognizes them. A user may open an interactive panel and look at data coming from instruments, and, without closing the previously panel used, open another panel to set telescope parameters. It is a real multitasking environment. The *application-driven* model, instead, forces the user to follow a predefined path in order to accomplish the task. A typical example of this model is a checkbook program. The user enters the customer data, then the check data and then he can choose to save or abort the operation. Typically this is the only point where the user has a (little) freedom of choice. Event-driven programming requires multitasking operating systems and windowing systems. Application-driven programming can be operated on single task operating systems on alphanumeric terminals.

The Galileo User Interface uses the *event-driven* interaction model.

Implementation

Different TUI on different workstations?

There are several workstations connected to the TNG LAN. Each of them has a specific purpose, typically telescope monitor, instrument control or data reduction and archiving. Obviously, these are radically different task typologies, requiring different approaches by the user. The Galileo User Interface is a dynamic adaptive module, which, by checking the type of the machine on which it is running, appears and behaves accordingly to the specific task required. So, for example, the TUI on the Telescope Control workstation will start displaying the main parameters of the telescope, while on the Instrument Control workstation it will display the linked instrument parameters.

This behavior can be easily modified by changing the **`TNGSYSTEM`** environment variable, thus changing the "type" of the default user interface. The TUI can also be changed dynamically during normal operations, by loading different sets of interactive panels, allowing the user, for example, to monitor telescope parameters while sitting at the instrument workstation.

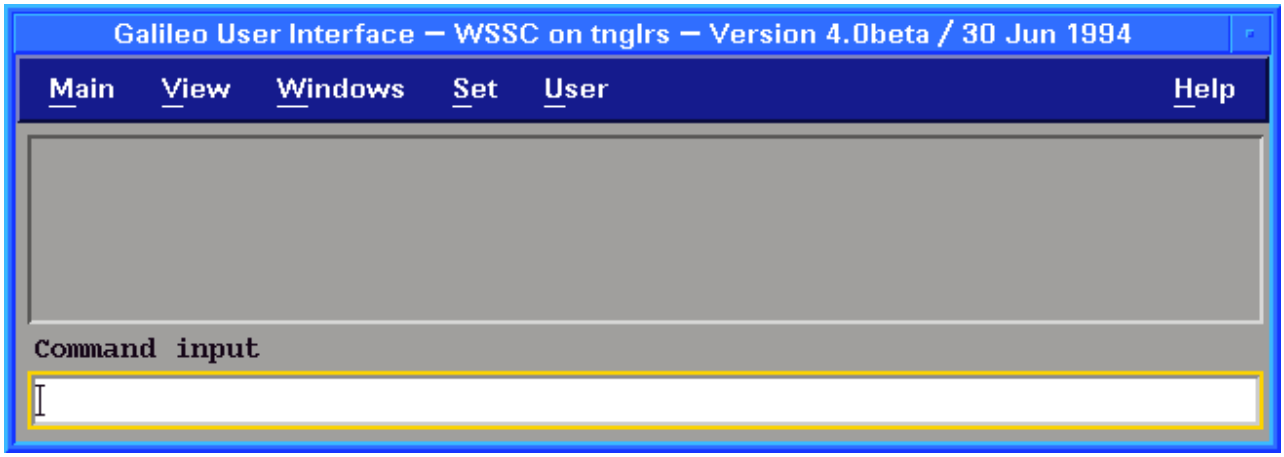
The TUI is, basically, a set of little modules that can be combined to form any type of interactive layout for the user. What the user sees, in fact, are just interactive panels and (if enabled) the command line interface; the Galileo User Interface is (hopefully) non-intrusive.

Basic layout

The basic TUI appear at the top of the workstation screen (the main screen, in case of multiscreen workstations) as a wide window, spawning from left to right margins of the monitor. In this window (called the main window) there are:

The menu bar

The command line interface with the history list



The TUI main window

The menu bar

The menu bar appear at the top of the main window -and of main screen, of course- and gives the user quick access to most-often used system commands. They are divided in six top level items: Main, View, Windows, Set, User and Help. The last one, Help, does not present a related pulldown menu, but is a simple push button that recalls the Help system (HELP [4]). The others, when selected, display a submenu.

Main submenu

EXIT

This entry, if selected, sends an **`exit`** command to the connected TLPs, closes all TLP connections, detaches from the TDB and exits the User Interface. When the User Interface exits, the **`tng_init`** main process recognizes the event and executes a regular shutdown of the WSS.

View submenu

NEW VIEW PANEL

See "Creating View Panels"

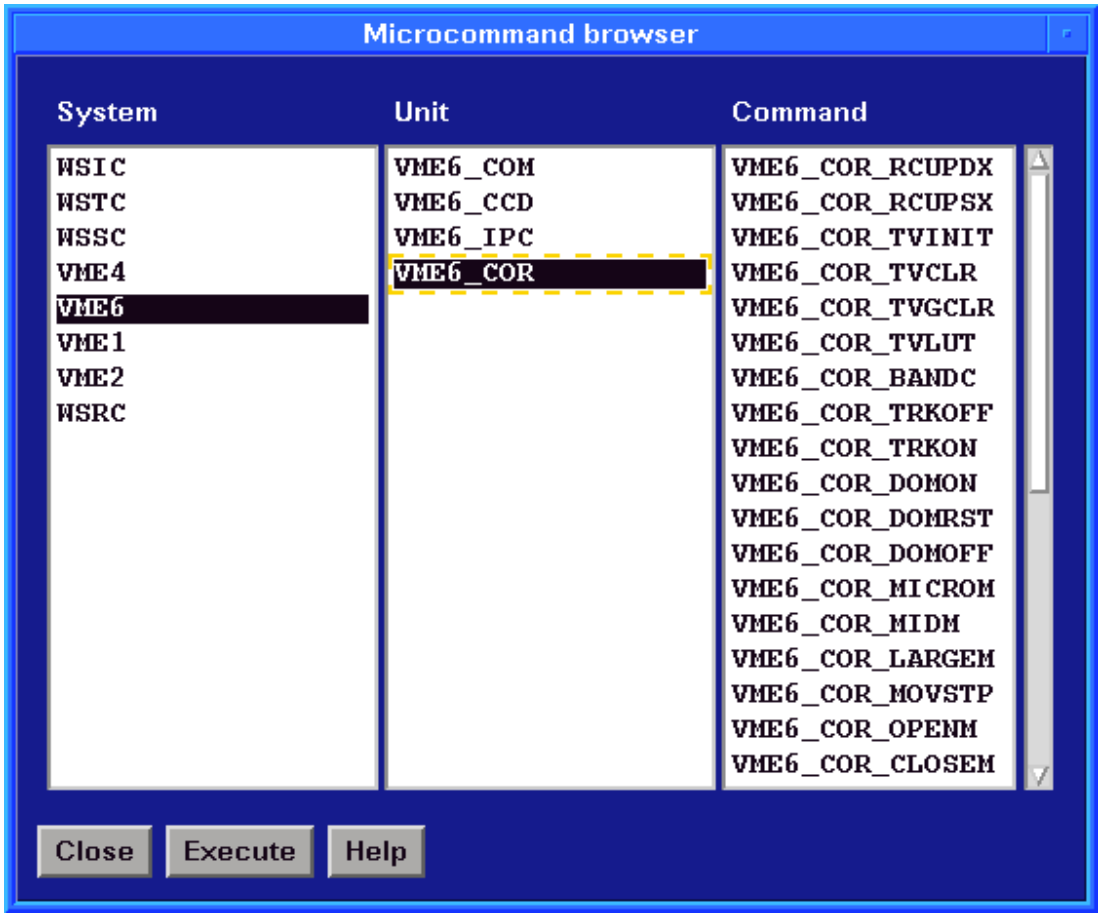
LOAD VIEW PANEL

See "Opening View Panels"

Windows submenu

COMMAND BROWSER

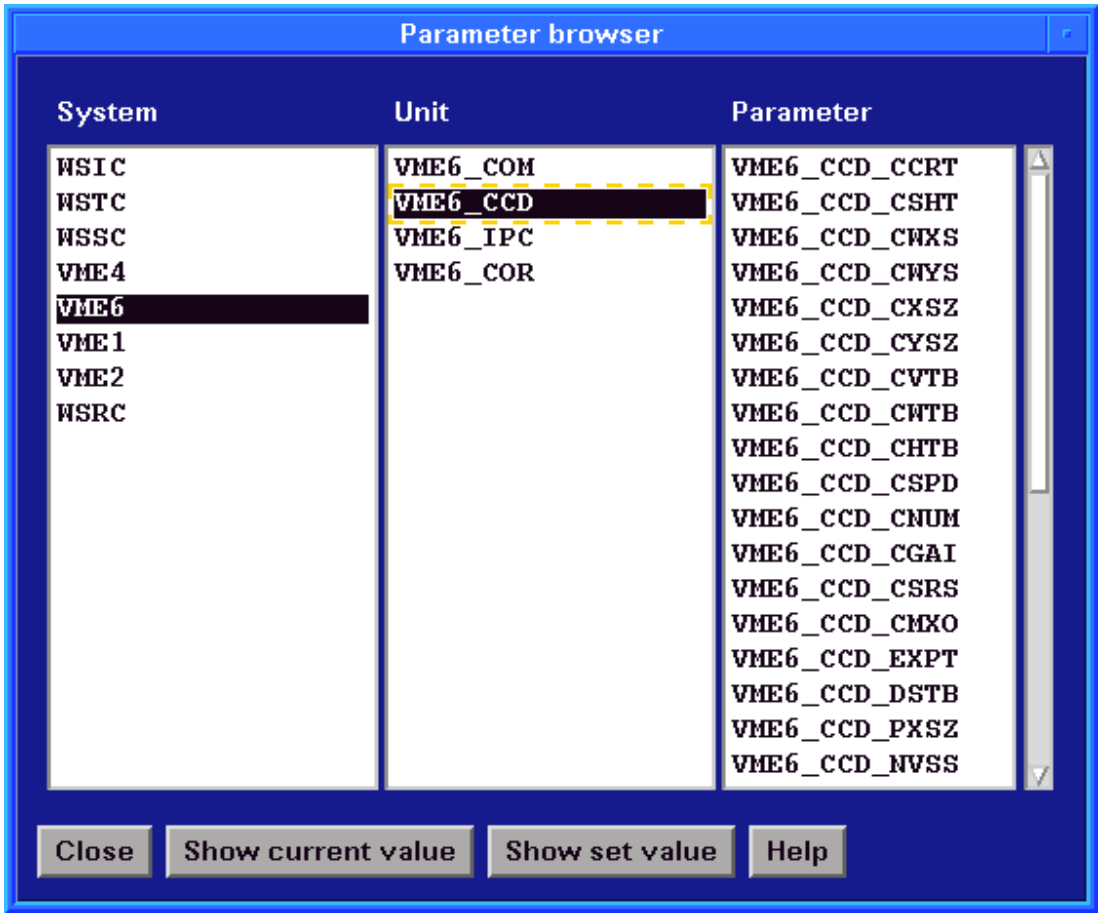
This entry displays the command browser. This is a panel where the user can browse in the list of ALL the systems and the units defined in the WSS environment and execute EVERY defined microcommand. By selecting a microcommand acronym and then the EXECUTE button, or by double clicking on it, the command will be executed. Selecting a microcommand acronym and the HELP button, the Help System will be invoked with the related help topic.



The Microcommand Browser

PARAMETER BROWSER

This entry displays the parameter browser. This is a panel where the user can browse in the list of ALL the systems and the units defined in the WSS environment and examine EVERY parameter defined. By selecting a parameter acronym and then the SHOW CURRENT VALUE button, or by double clicking on it, the current value of the parameter will be displayed in the system messages area. By selecting a parameter acronym and then the SHOW SET VALUE button, or by double clicking on it, the set value of the parameter will be displayed in the system messages area. By selecting a parameter acronym and the HELP button, the Help System will be invoked with the related help topic.



The Parameter Browser

PANEL BROWSER

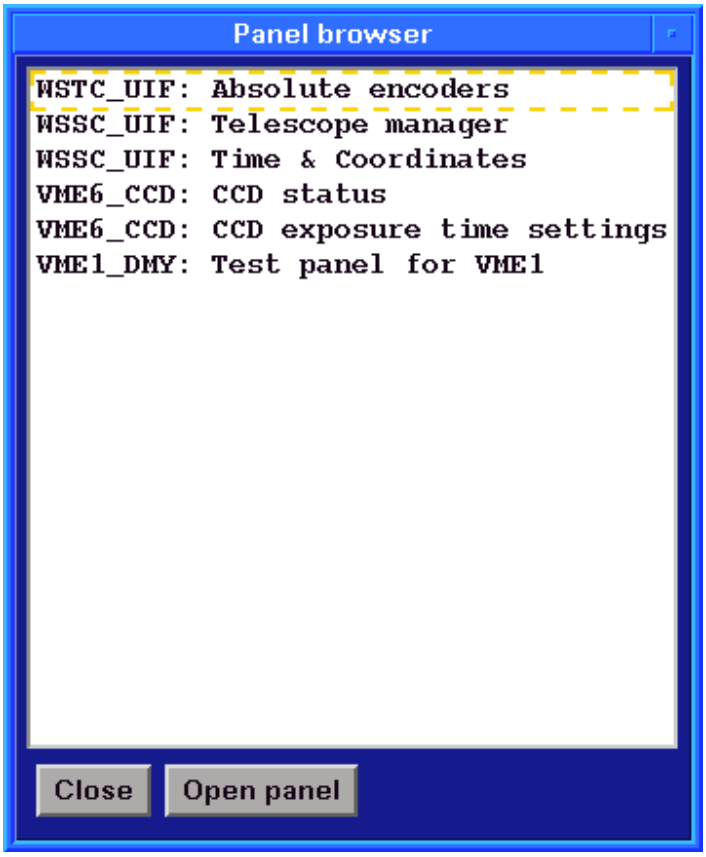
This entry displays the panel browser. This is a panel where the user can browse in the list of ALL the interactive panel defined in the WSS environment and open every one of them. By selecting a panel acronym and then the OPEN PANEL button, or by double clicking on it, the selected panel will be opened and positioned on the related screen.

There are some consideration to be made about panels:

Permissions: only authorized users can open some panels. Every panel has a *protection code* that, for the panel to be opened, has to be lower than the user's protection level. System programmers have level 0, system mantainers have level 1 and current users have level 2. The protection code of the panel is set in the table editor program.

Positioning: the panel will be displayed at the position fixed during its creation. The output screen (in a multiscreen workstation) is set in this phase also. If a panel designed to be used in a multiscreen workstation is opened on a workstation that has a smaller number of screens, its output screen will be defaulted to the last available screen. E.g.: if a panel is designed to be user on screen #3, and it is opened on a two screen workstation, it will be displayed on the screen #2; if it is opened on a single screen workstation, it will be displayed on screen #1.

Selecting a panel acronym and the HELP button, the Help System will be invoked with the related help topic.



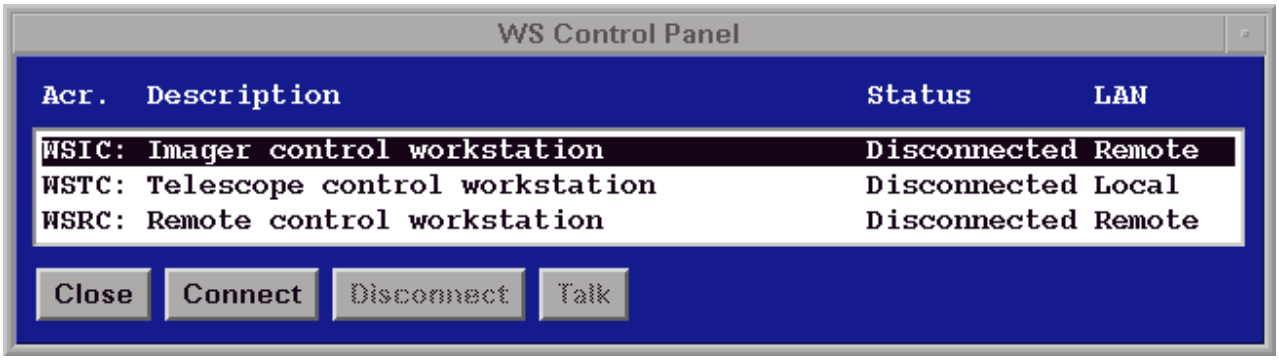
The Panel Browser

WS CONTROL PANEL

This entry displays the Workstation Control Panel. This is a panel where the user can read relevant information about the status of the TNG LAN and remote connections.

The list shows all the workstations defined in the TDB. Near the acronym, there is a short description of the "official" use of that workstation. This description does not always correspond to the actual use of that workstation, due to the fact that every workstation in the TNG LAN can be used for every purpose (Telescope control, instrument control, data reduction and management).

Status and type of connection are also listed.

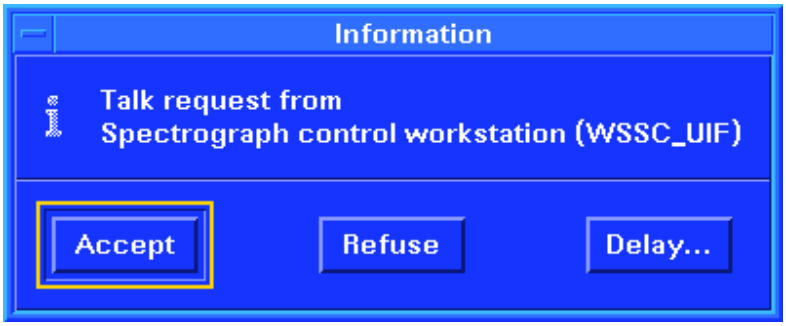


The Workstations control panel

This panel can be used by a system mantainer at the telescope site. It cannot be used from a remote workstation. Via the CONNECT button, a remote workstation can be linked to the TNG LAN to perform remote observing. This function has to be controlled by a local operator, because, due to the design of the WSS software, a remote workstation has almost the same capabilities of a local workstation. It is not like a remote observing station of the current generation, but it links to the control LAN in a way that is not distinguishable from a local control workstation.

A local workstation cannot be connected. It connects by itself at the system startup. So, for a local machine, the CONNECT button is "greyed".

Another feature found in this panel is the TALK button. Clicking on this button will send a *talk request* message to the selected workstation where a dialog box is shown.



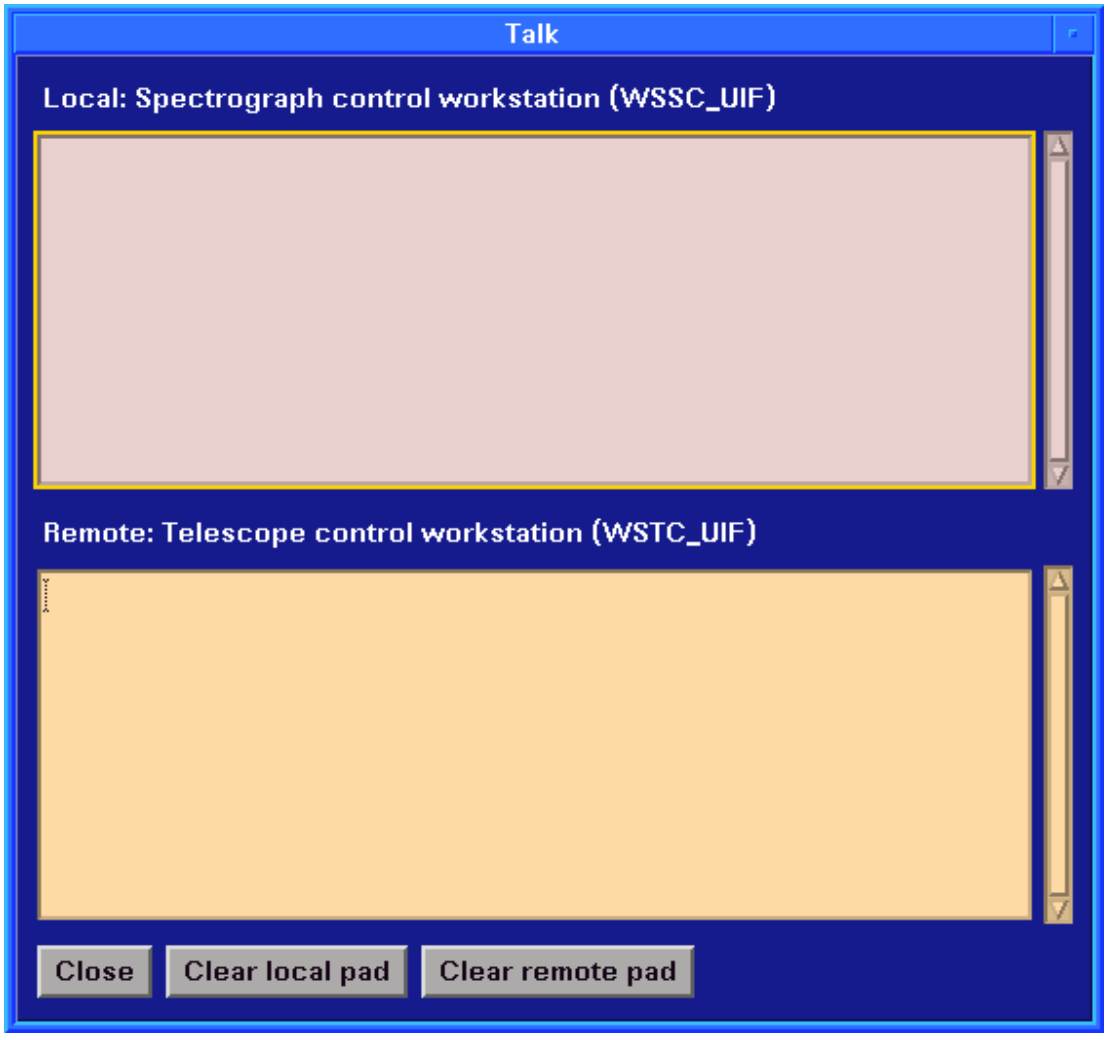
The talk request dialog box

If the remote user clicks on the REFUSE button, the following dialog box will appear on the

screen of the caller:

The talk request refused dialog box

On a successful approval via the ACCEPT button, the dialog box will be substituted by the *Talk Pad*, divided in two separate sections, with the local pad and the remote pad aligned vertically.



The Talk Pad

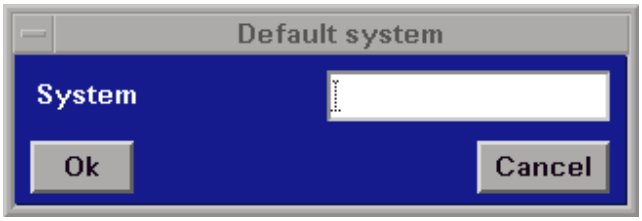
The user can write on the upper (local) pad, and the entered string will be sent to the remote system at the pressing of the <Enter> key. In the same way, strings written by the remote user will appear on the remote pad. The CLOSE button will close the communication, prompting the remote user with the following dialog box:

The talk session closed dialog box

Set submenu

DEFAULT SYSTEM

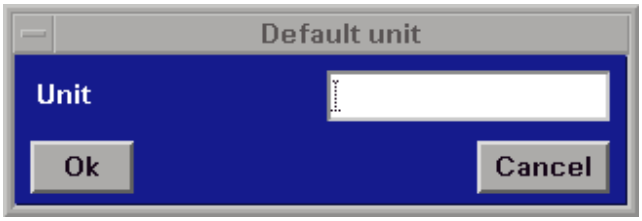
This entry displays a small dialog box where the user can enter any valid system name (e.g. WSTC or VMAZ or something like this) that will be prepended to all the commands entered via the command line control. To disable this feature, simply reselect this item and enter an empty system.



The Default System dialog box

DEFAULT UNIT

This entry displays a small dialog box where the user can enter any valid unit name (e.g. UIF or COM or something like this) that will be prepended to all the commands entered via the command line control. To disable this feature, simply reselect this item and enter an empty unit. This entry can be selected only after the default system has been set.



The Default Unit dialog box

User submenu

USER LEVEL

Not currently implemented

LOAD LAYOUT

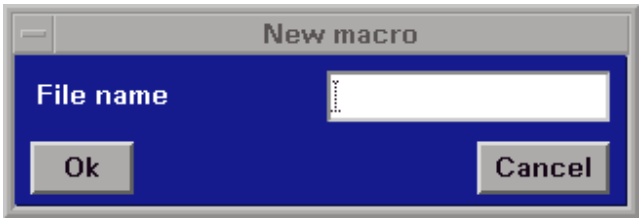
See "Opening user layouts"

SAVE LAYOUT

See "Saving user layouts"

NEW MACRO

This entry displays a small dialog box where the user can enter the name (a UNIX file name, without special characters in it) of a new macro sequence of commands. After doing that, an empty Macro Editor panel will appear (see LOAD MACRO).



The New Macro dialog box

LOAD MACRO

This entry displays a standard file selection dialog box that lets the user to load and edit a macro file (`.macro` extension).

Once the user has selected the file, this is loaded into the macro editor window. From here, the user can select, basically, two options: edit the macro file or execute it.

Edit a macro

Editing a macro is a simple process: the user enter commands and microcommands (or other macro's name) in a chronological sequence. This panel has basic editing capabilities (more, like Copy and Paste, will be added in later releases). Comments can be added by putting a # character at the beginning of the line.

The upper text box (labeled Macro name) will contain the readable name of the macro.

This is a typical example of a macro, shown in the Macro Editor panel below:

Load the telescope manager panel

----

loadpanel wssc_uif_telman

set system wssc

set unit uif

The Macro Editor

Execute a macro

To execute the macro currently in the edit window, press the EXECUTE button.

Recording a macro

This function is currently not supported.

The command line interface with the history list

This is one of the methods to enter commands to the WSS. By making full use of the TNG naming convention, the user can type EVERY microcommand here, regardless of its case, and it will be executed. All entries are saved in the history list, where they can be recalled, with a click of the mouse, and edited again in the command line control. The history box provides a scroll bar, to the right, to move across all saved entries.

The alarm log list

This is an optional panel that keeps log of all the alarm and warning messages notified to the user. This panel cannot be closed, only minimized.

Panels

What is a panel?

A panel is a window (as defined in the X11R5 and Motif 1.2 manuals) that provides a coherent area of interaction with the user. Panels are used extensively in the WSS, so extensively, in fact, that almost everything that appears on workstation screen is a panel. Technically speaking, the main window itself is a panel, but we don't consider it here, because it was not created with the Interactive Panel Editor, it is just the main window.

Interactive panels and view panels

There are two types of panels in the TUI: Interactive panels (or, simply, panels) and View panels. The first ones have, as the name suggest, interactive capabilities, i.e. the user can modify objects on them; the latter are output only, they exist just to display dynamic parameters as they come from the field.

From now on, Panels will refer to interactive panels and output panels will be referred to as View Panels.

Interactive Panels

A set of interactive objects is defined in order to be used in the panels. The current version of the WSS provides the system with six drawing objects and eight interactive objects.

Drawing objects are:

LINE

RECTANGLE

BOX (a filled rectangle)

CIRCLE

FILLCIRCLE (a filled circle)

LABEL (a descriptive text)

Interactive objects are:

OUTPUT (a data output area)

SLIDER (a slider with an Apply button)

LEDBAR (a multi-color dynamic bar graph)

BUTTON (a single-state push button)

STATUS (a two-state indicator with changing colors and/or texts)

ANALOG (a dynamic gauge)

SETPAR (similar to OUTPUT but with editing capabilities)

CLOCK (a simple bar graph vith a numeric value)

CHECKBUTTON (a two-state switch)

Drawing objects

These objects are non-interactive lines, rectangles or circles. They serves only as borders, backgrounds, informative or decorative parts of an interactive panel. In addition, they can also have the dynamic property, that gives them the ability to move, rotate or change colors according to variations of related telemetry values. They are different from interactive objects, but can be used for animations or multi-state indicators.

Interactive objects

 OUTPUT

This is a text widget that can display data from the telescope internal database. No interactions are allowed on this object.

 SLIDER

This is a compound object that lets the user to enter a command with a single parameter. The value of the parameter is defined by the position of a slider widget, spawning between the minimum and maximum values of the first operand of the microcommand as defined in the TDB. To actually execute the microcommand with the specified value as the first operand, the user has to click on the Apply button (just below the slider). Every modification made to the slider position is unrelevant until the user presses the Apply button.


LEDBAR

A bar graph that displays the current value of a parameter. The color of the bar varies accordingly to the value and the warning and alarm threshold settings of the parameter. This object works only with numerical parameters.

BUTTON

This is a single-state push button that is linked to the execution of a command. The dispatching of the command issued is reported in the system messages log list, so, before pressing the button again, the user must watch there to see if the command has been dispatched correctly.

STATUS

This is an object with various appearances. Basically it is a two-state indicator based on the value (numerical) of a parameter and a definable threshold. There are four types of STATUS: Generic lamp, Status text, Fault text and X Pixmap.

Generic lamp:

a circular lamp-like widget that can assume two user definable colors

Status text:

an output text widget that can display two short texts. The background color changes to green when the threshold is exceeded.

Fault text:

an output text widget that can display two short texts. The background color changes to red when the threshold is exceeded.

X pixmap:

a little area where two pixmaps can be displayed.


ANALOG

This is an output only widget similar in appearance to the gauge found on analog multimeters. The meter scale is colored in a way similar to the LEDBAR object (red, yellow, green or gray, yellow, red). Its purpose is to show the dynamic value of a numeric parameter.


SETPAR

This object looks like the OUTPUT object, but has editing capabilities, too. When the user

presses the Return key, the value actually contained in the SETPAR object is transferred to the internal database as a set value. This can be used for time settings, and other parameters that have read/write capabilities.

 CLOCK

This object displays the value of a numeric parameter in a small bar graph. It is different from the LEDBAR object in the sense that a) the bar does not change color with the variation of the value, and b) it is more easy to modify the maximum value to allow different extensions. For example: this object can be used to show the elapsing time of an exposure. If a user takes different exposures with different times, the setting of the CLOCK object has to be modified every time. This is accomplished by entering a new set value for the exposure time telemetry parameter, so notifying the object to adjust its maximum value to this setting. As time goes on, the object will show a bar growing until it reaches the right end of the object.

 CHECKBUTTON

Not yet implemented.

Opening panels

Panels can be opened with the command LOADPANEL <panel_name or with the View|Load interactive panel menu entry. There is no need to specify its directory or suffix, the TUI automatically add them to the panel name. If the LOADPANEL command is used without parameters, or the menu entry is selected, the file open dialog box will display. This is a standard dialog box for file opening, and is a predefined composite widget from the Motif 1.2 widget set. The user is already located at the correct data path and only relevant files (in this case, interactive panels) are allowed for selection. Selecting a file and then the OK button or double-clicking the file name will result in the panel to be opened and displayed.

Closing panels

Panels can be closed in two ways: by clicking the CLOSE button or by issuing the **CLOSEPANEL** command. With the last method the user has to specify the panel number, visible in the panel's caption bar.

Reset to default values

By pressing the DEFAULTS button, all SETPAR and SLIDER objects are reset to their default values, as defined in the TDB.

Getting help

The HELP button pops up the Help System with the panel related topic.

Default starting panel

The default starting panel for every workstation is defined by the system programmer. It is created via the Interactive Panel Editor with the name **<TNGSYSTEM>.pan**. For example, the default starting panel for the Telescope Control workstation will have the name **wstc.pan**. A panel with this name will be loaded automatically at the startup of the WSS in the telescope control workstation.

View Panels

View panels are output only windows, that are used to visualize parameter values. The functionality of a view panel is limited, but it can provide a quick and simple method to continuosly monitor parameter values, without using an external, off-line, editor to build an interactive panel.

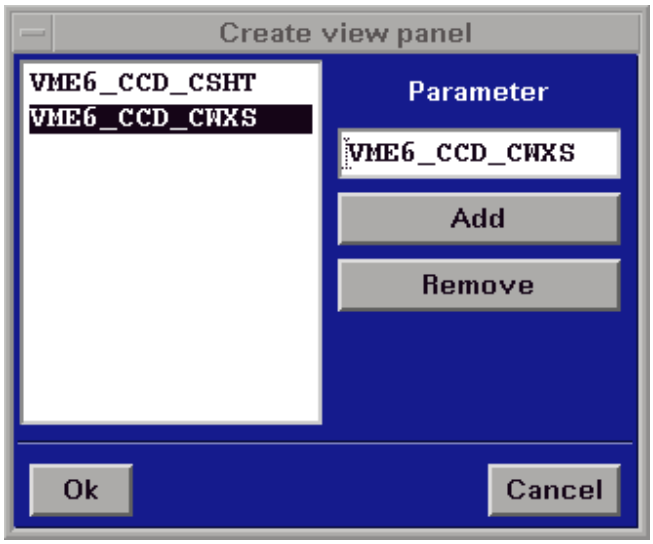
Creating View Panels

View panels are built by the user himself, using the View|New view panel entry of the main menu. A dialog box will appear, and the user will insert the View Panel name in the edit field. This will be the name reported in the caption bar of the view panel.

The New View Panel dialog box

By pressing Return or selecting the OK button, this dialog box will be closed, replaced by another dialog box, which will contain some objects that will allow the user to build a new view panel: on the left there is the parameters list, where all the selected parameters are displayed, on the right there are an edit field and two buttons. In the edit field there is displayed the acronym of the currently selected parameter (acronyms can be entered in either upper or lower case), the ADD button takes the acronym in the edit field and adds it to the parameter list, the REMOVE button deletes the currently selected parameter from the parameter list. If no parameter is selected, this button is not enabled.

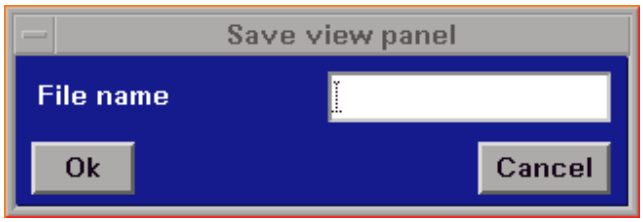
Another method to add parameter acronyms to the list is by double-clicking on them in the Parameter Browser. By pressing the OK button, the view panel will be created and displayed on the screen.



The second New View Panel dialog box

Saving View Panels

Newly created View Panels can be saved by pressing the SAVE button. A dialog box requesting the view panel name will appear, and the user shall enter a unique name for its view panel to save it on disk. There is no need to specify directory or suffixes, the TUI automatically will add them to the view panel name.



The Save View Panel dialog box

Opening View Panels

View Panels can be opened with the command **LOADVIEW** `<panel_name>` or with the View|Load view panel menu entry. There is no need to specify directory or suffixes, the TUI automatically add them to the view panel name. See "Opening panels" for a description of the generic file open dialog box.

Closing View Panels

View Panels can be closed by pressing the CLOSE button on the view panel.

Layouts

A layout is a collection of interactive panels and view panels. These panels can be related to each

other or not, there is no need for that. For example, if the telescope movement involves two or three panels, these can be grouped together and so recalled later; this set of panels is called a layout. A layout keeps track of the currently opened interactive panels and view panels and their positions.

User-defined layouts

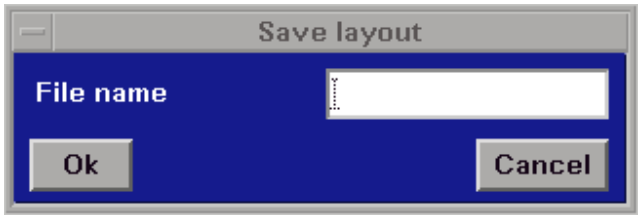
After the system start, the user can open and close any panel. He can close the default starting panel, too. At any time, however, the current layout of view and interactive panels can be saved or a new layout can be recalled from disk. Loading a new layout will not close the current one.

Opening user layouts

User layouts can be loaded selecting the User|Load layout menu entry. See "Opening panels" for a description of the generic file open dialog box

Saving user layouts

User layouts can be saved selecting the User|Save layout menu entry. A simple dialog box will ask the user the name of the layout. Any valid unix name can be given, without need to specify path or extension.



The Save Layout dialog box

Default starting layout

If the user saves a layout with the name **`<TNGSYSTEM>_uif.dsk`** (i.e: if **`TNGSYSTEM`** is **`wstc`** then the name will be **`wstc_uif.dsk`**), this layout will be loaded automatically at the WSS start-up on this workstation.

REFERENCES

GATE environment and module description
Baruffolo, C. Bonoli
TNG Technical Report n.

Galileo Project: Workstation Software System
Balestra, P. Marcucci, F. Pasian, M. Pucillo, R. Smareglia, C. Vuerli
TNG Technical Report n. 9

The Galileo User Interface Programmer's Guide
P. Marcucci, M. Pucillo
Pubblicazione Osservatorio Astronomico di Trieste n. 1455

The Galileo Help System
P. Marcucci
Pubblicazione Osservatorio Astronomico di Trieste n. 1456