-

Visible & Infrared Survey Telescope for Astronomy

IR CAMERA

Document Title:	**IR Camera Observation Software Design Description**
Document Number:	**VIS-DES-ATC-06084-0001**
Issue:	**1.3**
Date:	**1 March 2004**

Document Prepared By:	**S.M. Beard IR Camera Software Manager**	**Signature and Date:**	
Document Approved By:	**Mel Strachan ATC Project Manager Martin Caldwell IR Camera Systems Engineer**	**Signature and Date:**	
Document Released By:	**Kim Ward IR Camera Project Manager**	**Signature and Date:**	
Document Reviewed By:	**Gavin Dalton IR Camera Scientist**	**Signatures and Date:**	

CHANGE RECORD

Issue	Date	Section(s) Affected	Description of Change/Change Request Reference/Remarks
1.0	12/11/03		FDR version
1.1	06/01/04	1.4, 2.1.1	Version of Data Interface Control document corrected. VCAM renamed to VIRCAM. Updated following comments received at the FDR: • OS does not run self-test at the beginning of the night.
1.2	12/01/04	3.2.6, 5.6.1	Comments from Joerg Stegmeier incorporated. Reference to IRACE software ICD.
1.3	1/03/04	5.3, Figure 16 ,8.4	Rotator calibration template removed, following telescope calibration meeting. IMAGE mode renamed to IMAGING (in line with modes used by OmegaCAM).

NOTIFICATION LIST

The following people should be notified by email that a new version of this document has been issued and is available on the IR Camera document database:

RAL:	Martin Caldwell
	Kim Ward
	Gavin Dalton
ATC:	Malcolm Stewart
	Mel Strachan
	Steven Beard
	Will Sutherland
Durham:	Paul Clark
	Nirmal Bissonauth
CASU:	Mike Irwin
	Peter Bunclark
QMUL	Jim Emerson

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Purpose

This document gives the detailed design of the VISTA IR Camera Observation Software. It follows on from the architecture described in the *"VISTA IR Camera Functional Specification",* [AD2], and together with the detailed design documents of the VISTA IR Instrument Control System, [RD5], and wavefront sensing system, [RD6], [RD7], [RD8], forms the Software Design Description of the VISTA IR Camera instrument software.

1.2 Scope

This document describes the detailed design of the VISTA IR Camera Observation Software only. A complete software overview may be found in the "*VISTA IR Software User and Maintenance Manual*", [RD10], and the "*VISTA IR Software Functional Specification*", [AD2].

1.3 Applicable Documents

[AD1] *VISTA IR Camera Software Requirements*, VIS-SPE-ATC-06080-0010, Issue 2.2, 12 January 2004.

[AD2] *VISTA IR Camera Software Functional Specification*, VIS-DES-ATC-06083-0001, Issue 2.2, 5 February 2004.

[AD3] *VISTA Infrared Camera Technical Specification*, VIS-SPE-ATC-06000-0004, Issue 1.0, 28 October 2002.

1.4 Reference Documents

1.4.1 VISTA IR Documents

[RD1] *VISTA IR Camera System Description*, VIS-SPE-RAL-06013-0001, Issue 2.0, November 2003.

[RD2] *VISTA IR Camera System Block Diagram,* VIS-DES-RAL-06013-9001, Issue 1.0, November 2003.

[RD3] *VISTA IR Camera Software Acronym and Abbreviation Glossary,* VIS-LST-ATC-06080-0030, Issue 1.0, 12 November 2003.

[RD4] *VISTA IR Camera Software Management Plan*, VIS-PLA-ATC-06016-0001, Draft 0.6, 3 September 2003.

[RD5] *VISTA IR Camera Instrument Control Software Design Description,* VIS-DES-ATC-06083-0001, Issue 1.2, 12 January 2004.

[RD6] *VISTA IR Camera Low Order Wavefront Sensor Software Design Description,* VIS-DES-UOD-06048-0001, Issue 1.0, 12 November 2003.

[RD7] *VISTA IR Camera High Order Wavefront Sensor Software Design Description,* VIS-DES-UOD-06048-0002, Issue 1.0, 12 November 2003.

[RD8] *VISTA IR Camera Autoguider Software Design Description,* VIS-DES-UOD-06048-0003, Issue 1.0, 12 November 2003.

[RD9] *VISTA IR Camera Software Acceptance Test Plan*, VIS-PLA-ATC-06087-0001, Issue 1.0, 12 November 2003.

[RD10] *VISTA IR Camera Software User and Maintenance Manual,* VIS-MAN-ATC-06080-0020, Issue 1.0, 12 November 2003.

[RD11] *VISTA IR Detector Controller Technical Specification,* VIS-SPE-ATC-06020-0005, Issue 2.0, November 2003.

[RD12] *VISTA IR Camera Focal Plane Subsystem Design,* VIS-DES-RAL-06031-0002, Issue 2.0, November 2003.

[RD13] *VISTA IR Camera Wavefront Sensors Subsystem Design,* VIS-DES-UOD-06042-0001, Issue 2.0, 28 October 2003.

1.4.2 VISTA Data Flow Documents

[RD14] *VISTA IR Observing Strategy*, VIS-SPE-IOA-20000-0005, Issue 0.1, 10 October 2003.

[RD15] *Overview of VISTA IR Camera Data Interface Dictionaries*, VIS-SPE-IOA-20000-0004, Issue 0.1, 14 November 2003.

1.4.3 VISTA Documents

[RD16] *VISTA Instrument Software Requirements,* VIS-SPE-ATC-00150-0003, Issue 2.2, 25 July 2002.

[RD17] *VISTA Technical Specification,* VIS-SPE-ATC-00000-0003, Issue 2.37, 11 August 2003.

[RD18] *VISTA Software Management Plan*, VIS-PLA-ATC-00150-0006, Issue 2.0, 27 September 2001.

[RD19] *VISTA Software Architectural Design,* VIS-TRE-ATC-00150-0001, Issue 2.0, 2 October 2001.

[RD20] *VISTA Computer Hardware Architectural Design*, VIS-TRE-ATC-00150-0002, Issue 2.1, 5 March 2002.

[RD21] *VISTA Active Optics and Guiding Control Functional Specification,* VIS-SPE-ATC-13030-0002, Issue 1.0, 9 September 2003.

[RD22] *VISTA Active Optics and Guiding Workstation Software Detailed Design,* VIS-SPE-RAL-13030-0003, Draft 0.1, 9 October 2003.

1.4.4 ESO-VLT Documents

[RD23] *VLT Instrument Software Specification*, VLT-SPE-ESO-17212-0001, Issue 2.0, 12 April 1995.

[RD24] *VLT Software Programming Standards*, VLT-PRO-ESO-10000-0228, Issue 1.0, 10 March 1993.

[RD25] *VLT Software Management Plan,* VLT-PLA-ESO-00000-0006, Issue 2.0, 21 May 1992.

[RD26] *VLT Instrumentation Control Software — Relevant Documents (web page)*, see http://www.eso.org/projects/vlt/sw-dev/ins_doc/ins_doc.html.

[RD27] *VLT Software Requirements Specification,* VLT-SPE-ESO-10000-0011, Issue 2.0, 30 September 1992.

[RD28] *Data Flow for VLT Instruments Requirements Specification*, VLT-SPE-ESO-19000-1618, Issue 1.00, 21 April 1999.

[RD29] *VLT Common Software Overview*, VLT-MAN-ESO-17200-0888, Issue 1.0, 17 August 1995.

[RD30] *VLT INS Common Software Specification,* VLT-SPE-ESO-17240-0385, Issue 2.1, 15 July 1996.

[RD31] *VLT Guidelines for the Development of VLT Application Software*, VLT-MAN-ESO-17210-0667, Issue 1.2, 8 October 2001.

[RD32] *VLT Telescope Control System (TCS) User Manual,* VLT-MAN-ESO-17230-0942, Issue 2, 22 March 2002.

[RD33] *TCS AutoGuiding and Field Stabilisation Design Description,* VLT-SPE-ESO-17230-0933, Issue 3.0, 10 April 2000.

[RD34] *VLT Active Optics Design Description*, VLT-SPE-ESO-17210-1173, Draft, 20 October 1997.

[RD35] *VLT High Level Operating Software (HOS) / Sequencer User Manual,* VLT-MAN-ESO-17220-0737, Issue 3, 28 March 2002.

[RD36] *VLT High Level Operating Software (HOS) / Broker for Observation Blocks (BOB) User Manual,* VLT-MAN-ESO-17220-1332, Issue 3, 24 March 2003.

[RD37] *VLT Software Template Instrument Software User & Maintenance Manual*, VLT-MAN-ESO-17240-1973, Issue 4, 31 March 2003.

[RD38] *Base Observation Software Stub (BOSS) User Manual,* VLT-MAN-ESO-17240-2265, Issue 1.2, 20 March 2002.

[RD39] *VLT INS Common Software for Templates — User Manual*, VLT-MAN-ESO-17240-2240, Issue 3, 31 March 2003.

[RD40] *VLT INS Common Software Configuration Tool (ctoo) — User Manual*, VLT-MAN-ESO-17240-2235, Issue 3, 31 March 2003.

[RD41] *VLT INS Common Software Startup Tool (stoo) — User Manual*, VLT-MAN-ESO-17240-2153, Issue 3, 31 March 2003.

[RD42] *VLT INS Common Software / Base ICS (icb) User Manual*, VLT-MAN-ESO-17240-0934, Issue 4, 31 March 2003.

[RD43] *HOS/Broker for Observation Blocks User Manual*, VLT-MAN-ESO-17220-1332, Issue 2, 27 March 2002.

[RD44] *IRACE-DCS User Manual*, VLT-MAN-ESO-14100-1878, Issue 1.3, 12 February 2001.

[RD45] *VLT Real Time Display — User Manual,* VLT-MAN-ESO-17240-0866, Issue 2.8, 16 May 1999.

[RD46] *VLT Central Control Software (CCS) User Manual,* VLT-MAN-ESO-17210-0619, Issue 2.3, 31 March 2003.

[RD47] *VLT Extended CCS User Manual*, VLT-MAN-ESO-17210-0770, Issue 1.8, 30 September 2001.

[RD48] *VLT INS Common Software / Setup Files and FITS Log Handling (SLX) User Manual,* VLT-MAN-ESO-17240-0726, Issue 2, 25 March 2003.

[RD49] *VLT INS Common Software / Objective SLX (OSLX) User Manual,* VLT-MAN-ESO-17240-0853, Issue 2, 25 March 2003.

[RD50] *VLT CCS Event Toolkit* — EVH User Manual, VLT-MAN-ESO-17210-0771, Issue 1.8, 6 October 2001.

[RD51] *VLT Tools for Automated Testing (TAT) User Manual,* VLT-MAN-ESO-17200-0908, Issue 1.4, 15 February 2001.

[RD52] *VLT Configuration Management Module (CMM) User Manual,* VLT-MAN-ESO-17200-0780, Issue 2.0, 22 October 2001.

[RD53] *VLT Java Database Monitor — CcseiDb User Manual*, VLT-ESO-MAN-17210-2522, Issue 2, 1 April 2003.

[RD54] *VLT CCS On-Line Database Loader User Manual*, VLT-MAN-ESO-17210-0707, Issue 1.6, 30 September 2002.

[RD55] *VLT Software Graphical User Interface User Manual*, VLT-MAN-ESO-17210-0690, Issue 5, 31 March 2002.

[RD56] *VLT Common Software Installation Manual,* VLT-MAN-ESO-17200-0642, Issue 2, 30 March 2002.

[RD57] *VLT Common Software — Combined OS Installation Manual*, VLT-MAN-ESO-17200-2238, Issue 3, 13 March 2003.

[RD58] *VLT Software Installation Tool for VLT Software Packages (pkgin) — User and Maintenance Manual*, VLT-MAN-ESO-17240-1913, Issue 3, 31 March 2003.

[RD59] *Final Layout of VLT Control LANs*, VLT-SPE-ESO-17120-1355, Issue 2, 21 July 2003.

1.4.5 Interface Control Documents

[RD60] *ICD between the VLT Control Software and the Observation Handling System*, VLT-ICD-ESO-17240-19200, Issue 1.3, 7 June 2000.

[RD61] *ICD between Instrumentation Software and VLT Archive System*, VLT-ICD-ESO-17240-0415, Issue 1.0, 14 Sept. 1995.

[RD62] *Data Interface Control Document*, GEN-SPE-ESO-19400-794, Issue 2.0, 21 May 2002.

[RD63] *VISTA IR Camera to IRACE Software Interface Control Document*, VIS-ICD-ATC-06081-5001, Issue 0.1, 20 January 2003.

[RD64] *IR Camera to VISTA Telescope Control System Interface Control Document*, VIS-ICD-ATC-06000-13010, Issue 1.0, 28 October 2002.

1.4.6 OmegaCAM Documents

[RD65] Baruffolo, A, Bortolussi, A., De Pizzol, L. Magagna, C., *"Design of the OmegaCAM Instrument Software"*, Proc SPIE 4848, 2002.

[RD66] *"OmegaCAM Instrument Software Functional Specification"*, VST-SPE-OCM-23100-3062, Issue 1.2, 31 October 2001.

[RD67] *"OmegaCAM Observation Software Design Description"*, VST-SPE-OCM-23100-3064, Issue 1.1, 31 October 2001.

[RD68] *"OmegaCAM Guide and Image Analysis Software Design Description",* VST-SPE-OCM-23100-3065.

1.5 Abbreviations and Acronyms

To save duplication, all the VISTA IR software abbreviations and acronyms have been collected into one document, [RD3].

1.6 Glossary

To save duplication, all the VISTA IR software definitions have been collected into one document, [RD3].

2 OVERVIEW

2.1 Requirements and Constraints

The Observation Software has to coordinate the actions of all the other IR Camera software modules with the TCS and the data handling in response to the commands and parameters that it receives from the higher level software. It has to:

- execute commands/parameters received from BOB or from the operator;
- allow the telescope operator to interrupt an observation and/or submit a new one directly;
- coordinate the actions of the IRACE DCS, ICS and TCS when making science observations;
- configure the instrument hardware via the ICS;
- control the operation of the telescope via the TCS;

- control the acquisition of science and calibration data via the IRACE DCS;
- forward guide and LOWFS star. candidate information to the TCS;
- forward autoguider and LOWFS configuration information to the TCS;
- coordinate the IRACE DCS and HOWFS image analysis subsystem when making HOWFS observations;
- forward HOWFS coefficients to the TCS;
- ensure ancillary data is included in the FITS headers;
- provide a graphical user interface for the operator;
- generate observation logs.

2.1.1 Software Requirements Traceability Table

The following tables associate design features described in this document with software requirements listed in [AD1].

2.1.1.1 General Software Requirements

Software Requirement	**Design Feature**	**Reference**
SWR 2.1.1.01 SWR 2.1.2.01 SWR 2.1.3.01 SWR 2.1.3.02	The software conforms to the standard ESO-VLT software architecture and ESO-VLT programming standards and guidelines.	Applies to all the Software.
SWR 2.1.2.01 SWR 2.3.02 SWR 2.4.02 SWR 2.5.03 SWR 2.7.1.05 SWR 2.7.2.03	Existing ESO-VLT software will be reused wherever possible.	Applies to all the Software.
SWR 2.11.02 SWR 2.11.03	The software will be delivered with the required documentation.	Applies to all the Software.
SWR 2.1.2.02 SWR 2.2.1.01	The software design is compliant with existing ESO-VLT hardware and allows the instrument to be operated from the Paranal control room.	Section 6
SWR 2.1.4.01 SWR 2.1.4.02 SWR 2.1.4.03 SWR 2.5.09	The software will be designed to be robust and reliable and tolerant of faults.	Applies to all the Software.
SWR 2.7.1.01 SWR 2.7.1.02	The software will conform to ESO-VLT and VISTA safety requirements.	Applies to all the Software.
SWR 2.2.1.02 SWR 2.2.1.03 SWR 2.2.1.04	The VISTA IR Camera software is designed to be independent of any other instrument.	Applies to all the Software.
SWR 2.11.01	All external interfaces will be specified and conform	Section 3.2

	to ESO-VLT requirements.	

2.1.1.2 *Observation Planning and Scheduling Requirements*

Software Requirement	Design Feature	Reference
SWR 2.2.2.01 SWR 2.2.2.05 SWR 2.2.2.02	The Observation Software will normally be operated automatically by commands and parameters sent from BOB and the scheduler. An Observation Block will completely specify an observation so that no additional information is required from the operator. Intervention by the operator will only be sought if something goes wrong (e.g. poor guide star signal detected or a critical device fails).	Section 8.
SWR 2.2.2.03 SWR 2.2.2.04	The Observation Software can also be controlled directly by the operator using the GUI. However, it is normal practice for BOB to be used to start or restart an observation. The operator can use the OS GUI to pause, continue or interrupt an observation once it has started. An automatic operation should be interruptable within the 10 minute goal, *depending on the interruption capabilities of the underlying subsystems*.	Section 3.2.1
SWR 2.2.2.06	The Observation Software will supply a template signature file stating the instrument's capabilities so the ESO-VLT scheduling software can check each Observation Block before submitting it.	Section 8.
SWR 2.2.2.07	The Maintenance Software will include a self-test procedure which the operator can run at the beginning of each night to check that all systems are ready for science observations.	Section 3.4.1
SWR 2.2.2.09	The Observation Software will check that the camera and telescope have successfully reconfigured before starting an observation.	Section 3.4

2.1.2 Global operations

Software Requirement	Design Feature	Reference
SWR 2.2.3.01	The instrument engineering GUI will allow any single mechanism to be operated during detector readout for EMI noise pickup testing. *This is more of an engineering requirement on the ICS and DCS*	

	GUIs than an OS requirement.	
SWR 2.2.3.02	The Observation Software will query the ICS for the focus adjustment necessary after placing a new science filter in the beam and will command the TCS to make the focus adjustment.	Section 3.4
SWR 2.2.3.03	The VISTA IR Camera software will be capable of synchronising the readout from the detectors so that (in the event of electromagnetic interference) the AG, LOWFS and science detectors do not read out at the same time. *Achieving the requirement depends on each detector controller:* • *providing a "reading out" signal* • *allow its readout to be suspended on demand.* *This requirement is TBC.*	Section 3.4.5

2.1.3 Monitoring and logging

Software Requirement	Design Feature	Reference
SWR 2.2.4.01 SWR 2.2.4.02 SWR 2.2.4.04 SWR 2.2.4.05 SWR 2.2.4.09 SWR 2.2.4.10	The Observation Software will maintain a nightly log recording • all observations, including calibrations • telescope motions • camera configurations • faults • weather monitoring information. The ESO-VLT CCS logging facility [RD46] and [RD49], will be used to write the logs. The log will record the effect of any faults (e.g. non-operating detectors) on the status of the data (e.g. record the area of sky not covered). Each nightly log is expected to be transmitted to Garching during the following day.	Section 5.8
SWR 2.2.4.03	The Observation Software will also maintain a nightly engineering log recording • significant camera parameters, such as detector temperature, cryostat temperature, cryostat vacuum and TTL signals (such as the power fail	Section 5.8

	UPS event) • any instrument faults The information will be sufficient for an engineer to be able to follow the instrument status through the night and diagnose any problems. It will also be sufficient to monitor the change of performance of the instrument from night to night.	
SWR 2.2.4.07	All faults will be reported to the telescope operator as soon as they are detected.	
SWR 2.2.4.11	The Observation Software GUI will allow the operator to access real time status displays giving - The instrument configuration, status and health. - Sky and environmental conditions. - A "quick look" display of the latest science data. *Achieving this requirement assumes that the ICS, TCS and DCS provide screens containing this information.*	Section 3.2.1

2.1.3.3 Observation Software Requirements

Software Requirement	Design Feature	Reference
SWR 2.4.01	The Observation Software will (a) be controllable from BOB, via Observation Blocks, in the manner described in [RD60]. (b) configure the instrument hardware via the ICS. (c) control the operation of the telescope via the TCS. (d) control the operation of the guider and LOWFS via the TCS, supplying guide star candidates. (e) control the acquisition of science and calibration data via the IRACE DCS. (f) control the acquisition of high resolution wavefront sensor data via the IRACE DCS. (g) control the processing of HOWFS data and their transfer to the TCS. (h) ensure ancillary data is included in the FITS headers. (i) provide a graphical user interface for the operator. (j) generate observation logs.	Section 3 Section 4
SWR 2.4.02	(a) The OS will be based on the Base Observation Software Stub [RD38] provided by ESO. (b) The OS will run on the Instrument Workstation.	Section 3 Section 6

SWR 2.2.2.10 SWR 2.4.04	The Observation Software will operate efficiently: (a) It will allow concurrent operations where these are sensible (e.g. telescope offset concurrent with storage of data to disk). (b) It will be designed so any software delays are much less than the instrument and telescope reconfiguration time and do not therefore drive the observation overhead time. (c) The Observation Software will be designed to maximise survey speed.	Section 3.4

2.1.3.4 Data Handing Requirements

Software Requirement	**Design Feature**	**Reference**
SWR 2.10.04	The Observation Software must be allow the IRACE DCS to achieve its performance target of sustaining the storage of one exposure to disk every 10 seconds for 14 hours. The OS must be capable of assembling and adding the FITS header information at the same rate.	Section 7
SWR 2.10.05	The Observation Software will ensure adequate free disk space is available to store data before an exposure is initiated.	Section 3.4
SWR 2.10.06	The Observation Software will allow the storage of data from one observation to be overlapped with the configuration of the telescope and instrument for the next observation.	Section 3.4
SWR 2.10.09	The Observation Software will comply with the requirements stated in ESO's Data Interface Control Document.	Section 5
SWR 2.9.02	The templates and sequencer scripts provided by the Observation Software will acquire and store all data necessary fully to reduce the data, including - bias frame - dark frame - dome-flat frame - sky-flat frame - sky background frame	Section 8.
SWR 2.10.10 SWR 2.2.4.06	The Observation Software will ensure that sufficient metadata is included for each exposure to be completely identified and its data reduced automatically. This information will be provided in the data header and within the log files stored alongside the data. The log file belonging to each set	Section 5

	of data will be identified. The existence of missing or poor quality data will be indicated in the data headers	

2.2 *Observing Strategy*

The Observation Software needs to control the instrument and obtain data in a manner that is compatible with the VISTA IR observing and data reduction strategy outlined by the Data Flow project and described in [AD2] and [RD14].

- The basic data unit is a single IRACE exposure[1] made at a single telescope "pointing".
- Multiple exposures may be combined together while microstepping the telescope (i.e. shifting the telescope by a fraction of a detector pixel) to make a microstepped exposure. The observer can choose from a selection of microstep patterns.
- Multiple exposures (with or without microstepping) may be combined together while dithering the telescope (i.e. shifting the telescope by small amounts less than the field of view of the guide star) to make a dithered map. The observer can choose from a selection of different dithering patterns.
- Six dithered maps or exposures (with or without microstepping) may be combined together while offsetting the telescope by amounts which fill in the gaps between the detectors to make a "tile".

In practise microstepping is very similar to dithering, the only difference being the way in which the individual exposures are combined (microstepped exposures being interleaved and dithered exposures being coadded).

3 ANALYSIS

3.1 *Context*

Data flow diagrams for the entire VISTA IR software may be found in [AD2]. Figure 1 shows the context of the Observation Software. The software receives commands either from BOB or from the telescope operator through the GUI. The software configures the TCS, HOWFS image analysis, ICS and DCS subsystems and coordinates the actions of these systems to make observations. The raw data files from each observation are stored to disk by the DCS. The Observation Software collects FITS header information and adds it to the raw data. Observation logs are also written to disk. When each observation is complete a "newdata" signal is sent to the VLT on-line archive system (VOLAC), which makes the data available to the VISTA Data Flow System for processing.

[1] The exposure can itself be made from a series of coadded readouts, but any pre-processing of the data by IRACE is beyond the scope of the Observation Software.

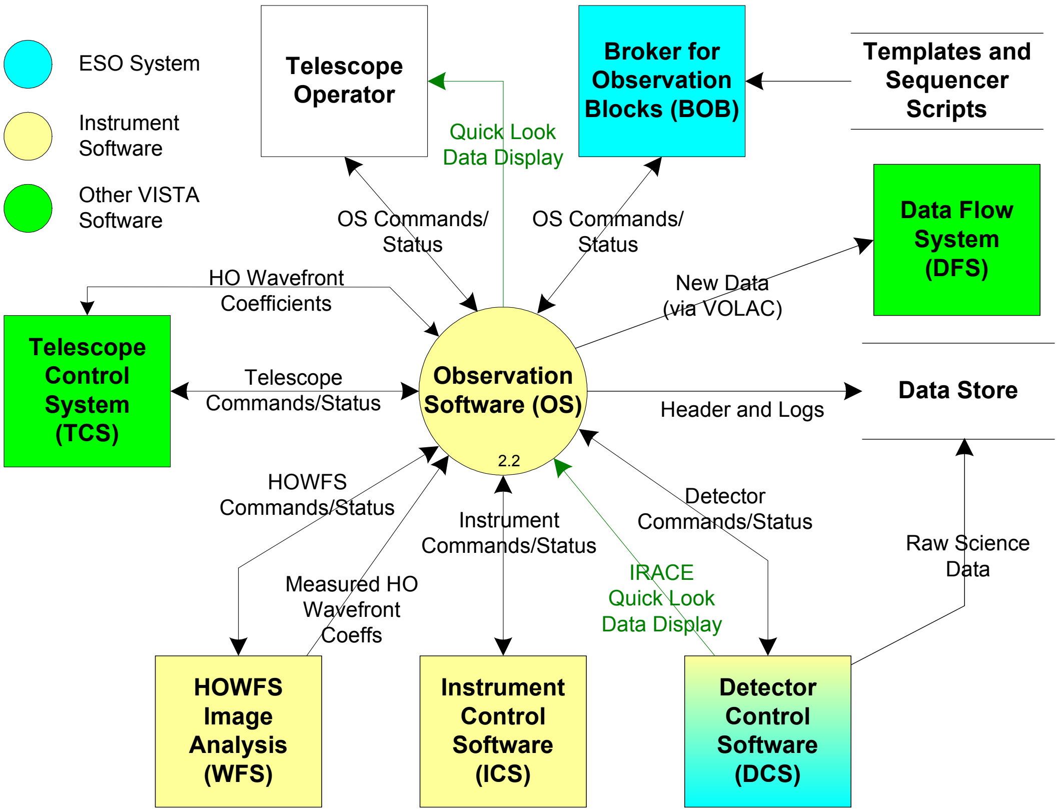

Figure 1 VISTA IR Camera Observation Software Context

The Observation Software decomposes into three parts, as shown in Figure 2.

- The OS GUI (module "vcopan") supplies the operator with a graphical user interface. It provides status information and allows the operator to issue engineering commands.

The other two processes are part of the "vco" module:

- The OS Server process (vcoServer) is based on the bossSERVER class supplied by the BOSS package. It is responsible for instrument configuration, state handling and command handling and for coordinating the underlying subsystems.

- The OS Archive process (bossArchiver_vco) is based on the BOSS archiver process[2]. This is responsible for collecting FITS header information and adding it to the raw data, signalling the on-line archiver (VOLAC) when the data are available for processing. It is also responsible for converting the IRACE data for each observation into a single multi-extension FITS file.

[2] BOSS archiver is named this way for historical reasons but its real function is a file handler process.

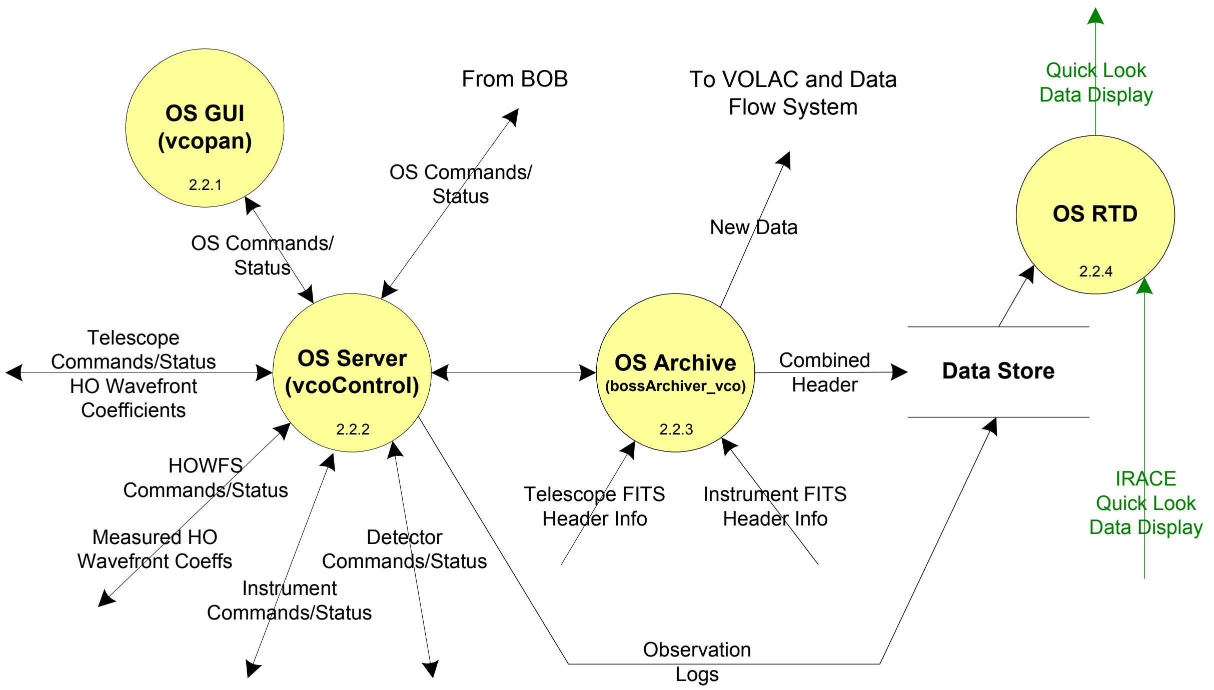

Figure 2 Decomposition of the VISTA IR Camera Observation Software

3.2 Interfaces

3.2.1 User Interfaces

The telescope operator interacts with the Observation Software through the OS GUI, through which OS commands can be executed and the status of the OS and its subsystems can be displayed.

3.2.1.5 OS Control Screen

The OS Control Screen is the top level control screen for the VISTA IR Camera. It should be a compact screen but have all the facilities a telescope operator would need frequent access to. The screen has to provide controls to allow the operator to:

- Change the overall state of the OS to “STANDBY” or “ONLINE” or cleanly shut down the software (OFF).
- PAUSE, CONTINUE, END or ABORT the current observation. (N.B. It is normal practice to start or restart an observation by interacting with BOB rather than the OS screen).
- Add a comment to the observation log.
- Start up the OS engineering screen (which in turn leads to the ICS, DCS and HOWFS engineering screens and the TCS simulator screen).
    - Note that in normal operation ICS, DCS and HOWFS parameters are defined either in a configuration file read during software startup or in the templates

processed by BOB. Adjusting instrument parameters by any other means is an engineering activity only.
  - The TCS simulator screen is used for testing the software in the absence of the real VISTA TCS. In normal operation the VISTA TCS screen will have been started separately on the telescope control workstation.
- Start up the ICS instrument status screen.
- Start up the DCS status screen.
- Start up the HOWFS status screen.
- Plus any other frequently used commands.

In addition, the OS control screen needs to display the following status items:

- The state, substate, access and simulation mode of all the instrument subsystems and TCS.
- The current instrument mode.
- The current exposure ID, exposure status, exposure time, remaining exposure time and data file name.
  - More detailed detector parameters can be queried by bringing up the DCS status screen.
- A disk space monitor.
- The current telescope coordinates.
  - More detailed telescope information can be queried by looking at the TCS console running on the telescope control workstation.
- The currently selected filter.
  - Other mechanism details can be queried by bringing up the ICS status screen.
- The current cryostat window temperature (compared with current dewpoint) and focal plane temperature.
  - Other temperatures and instrument sensor readings cane be queried by bringing up the ICS status screen.
- Any alarms reported by the instrument.
  - There should be a means of finding out more about an alarm by bringing up a status screen.

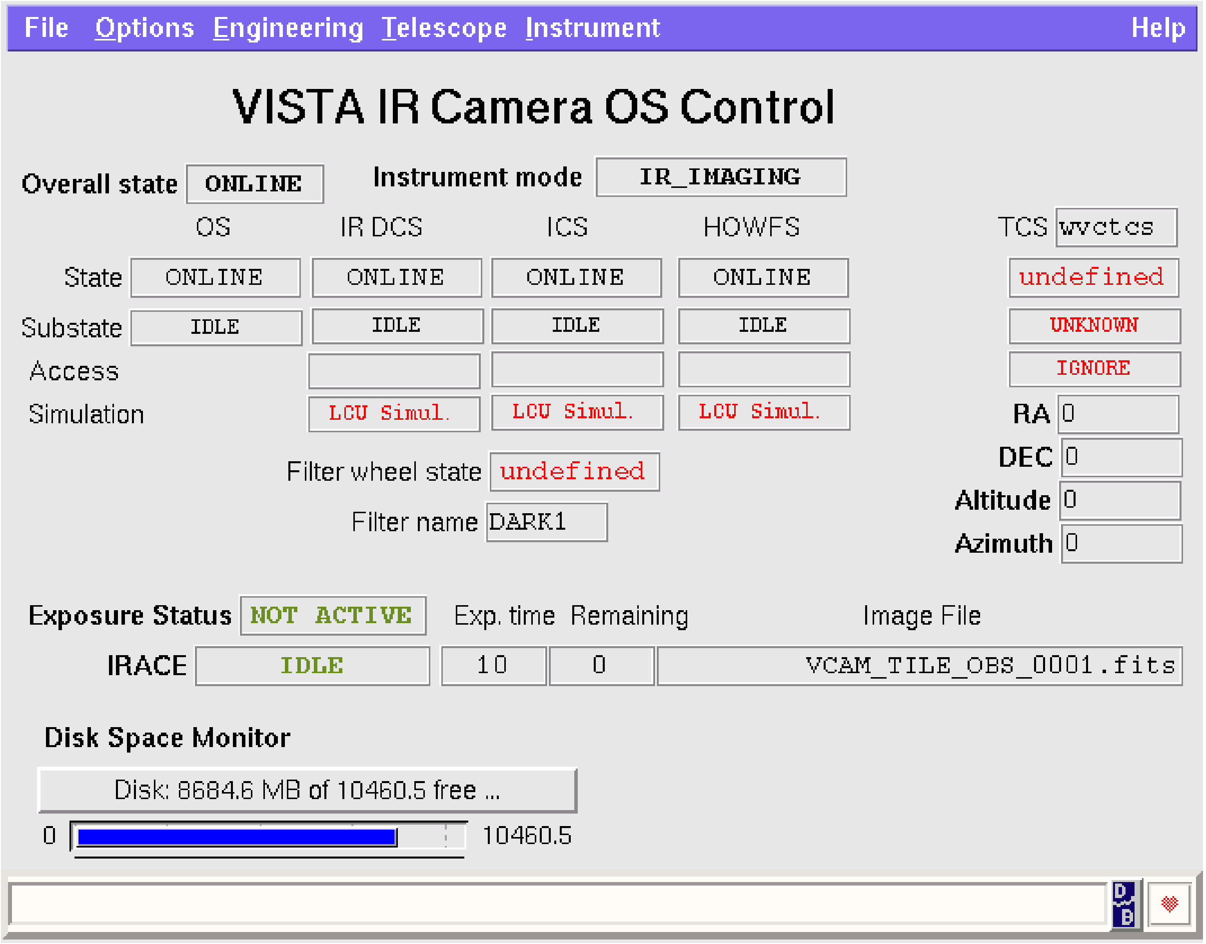

Figure 3 A Possible OS Control Screen

Figure 3 shows a possible OS control screen, based on the screen supplied by the ESO-VLT template instrument. The screen needs enhancing to include all the facilities mentioned above (and also needs updating to reflect the actual modes and templates described in this document), but gives the gist of what the OS screen will look like.

3.2.1.6 OS Engineering Screen

In addition to the standard OS Control Screen, there will be an OS engineering screen allowing access to the facilities needed for diagnosis, maintenance and testing of the instrument. The screen needs to provide controls or links to other screens to allow the operator or engineer to:

- Execute any OS command.
- Interrogate any database point.
- Check the instrument status in detail.
- Start up any subsystem status screen.
- Start up any subsystem engineering screen.
- Start up other VLT engineering screens.

- Interrogate the engineering log.
- Run a diagnostic test on the instrument.

3.2.2 Interface with BOB and Observation Handling System

The interface between the ESO observation handling software and instrument software is described in the "*ICD between the VLT Control Software and the Observation Handling System", *[RD60].

The Observation Software also supplies the ESO-VLT Observation Handling Software with the following files:

- **Instrument summary file** (VIRCAM.isf) containing a summary of the capabilities of the instrument (filters that may be selected, allowed ranges for detector temperature, readout modes of detector, etc.)

- For each instrument mode, a **reference setup file** (VIRCAM_<mode>_xxx.ref) giving all the parameter settings necessary to make an exposure in that instrument mode.

- For each instrument template, a **template signature file** (VIRCAM_<mode>_xxx.tsf) describing the parameters and procedure necessary to execute that template.

- One or more **template sequencer scripts** (VIRCAM_<mode>_xxx.seq) associated with the instrument templates.

The P2PP software allows an Observer to define observations to be made by the instrument by selecting templates and assigning values to the parameters associated with each template. The templates signature files, instrument summary file and reference setup files are used to build an "Observation Block" describing each observation, as shown in Figure 4 below. More details about the templates supplied with the VISTA IR Camera may be found in Section 8 on page 48.

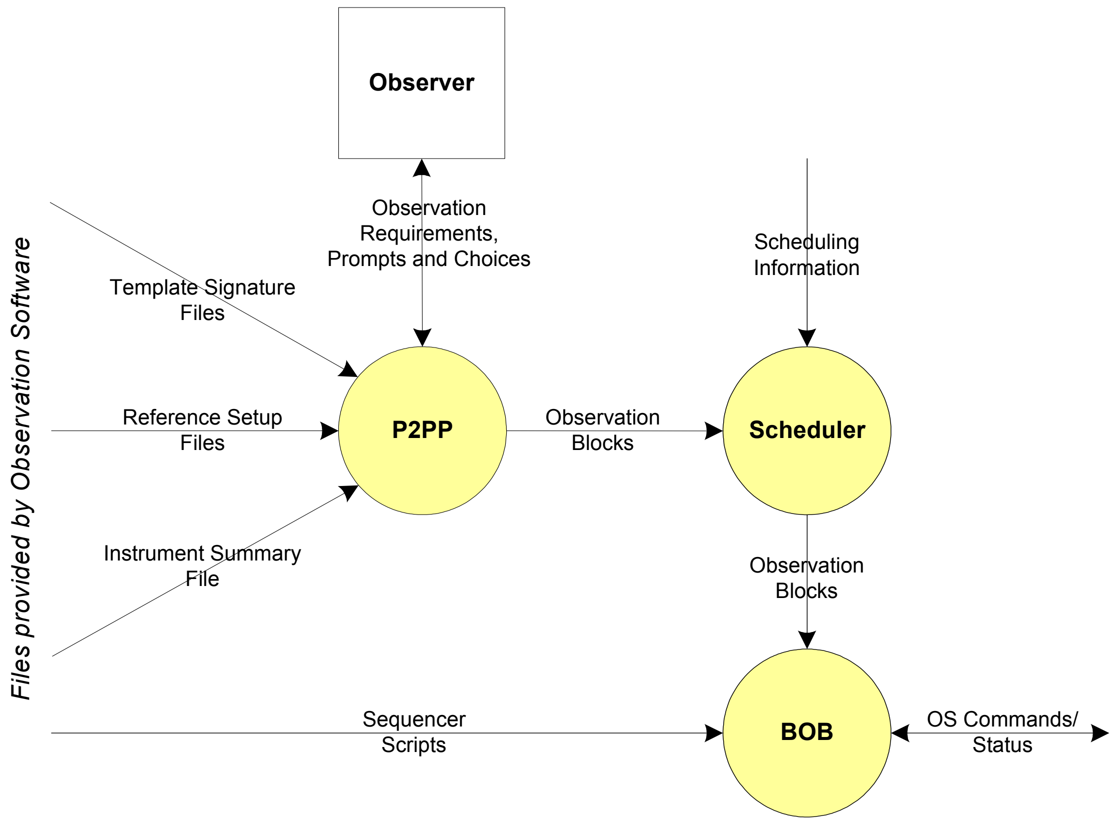

Figure 4 Use of Observation Software files by ESO-VLT Observation Handling System

BOB receives Observation Blocks and executes the template sequencer scripts. BOB interacts with the Observation Software by sending standard commands and parameters as described in Section 3.3.

3.2.3 Interface with TCS

The VISTA Telescope Control System (TCS) is based on the existing ESO-VLT TCS described in [RD32] and has very much the same interface with a science instrument. The interface is described in detail in the interface control document, [RD63]. The Observation Software needs the following operations to be possible over this interface:

- Preset the telescope by supplying to the TCS the required [RA, Dec, Epoch, Equinox, Radecsys] of a pointing centre, plus the position angle of the instrument Y axis[3]. The TCS responds by pointing so the required field centre falls on the focal plane at the rotator axis.

[3] The exact definition of the position angle is outside the scope of the camera software but should be robust enough to be stable near the celestial poles. It is used to set the rotator angle so the observer's image has the correct orientation on the sky. Default should be N at the top and E to the left. The parameter will be extracted by the OS from each acquisition template and passed on to the TCS.

- Preset the telescope by supplying to the TCS the required [RA, Dec, Epoch, Equinox, Radecsys, Proper motion, Effective wavelength and (X,Y)[4] in the focal plane in instrument coordinates] of an object to be acquired. The TCS responds by pointing so the specified object falls at the specified (X,Y) on the focal plane.
  - This pointing method can be used to ensure that a guide star falls on a particular detector pixel.
- Offset the telescope by providing a [ΔX, ΔY] to the TCS in instrument coordinates, ensuring that the (X,Y) axes maintain the same position angle on the sky so there is a seamless overlap (to allow accurate dithering and offsetting (see [AD2]). Getting the correct overlap will be especially important when observing near the celestial pole.
- Offset the telescope by providing an increment in the celestial coordinates of the pointing centre (in RA, Dec or in arcseconds on the sky).
- Offset the telescope to a new rotator angle (or position angle) while keeping the other pointing parameters fixed (for calibrating the location of the rotator centre).
- Instruct the TCS to drift at a specified non-sidereal rate specified as a (ΔRA/s, ΔDec/s). This is needed for open loop tracking of a non-sidereal object.
- Offset the telescope focus by supplying to the TCS a focus offset for the currently selected science filter.
- Wait for a LOWFS measurement cycle to complete (which will need to happen after a significant elevation change).
- Instruct the TCS to drift the rotator so that trailed exposures can be made for engineering purposes, to verify the location of the rotator centre (TBD – likely to be done through a telescope engineering console instead).
- Deliberately define a wavefront error vector for engineering purposes (TBD – likely to be done through a telescope engineering console instead).

The Observation Software will need to query the following status information from the TCS:

- Obtain from the TCS the current altitude and azimuth of the telescope (if required for flexure correction).
- Obtain from the TCS sufficient information (e.g. pointing direction, rotator angle, plate scale, etc.) to derive a mapping of each science detector pixel into (RA, Dec), and hence be able to store World Coordinate information along with the data.
- Obtain FITS header information (e.g. airmass at start, moon phase, angular distance from the moon) at the start of an observation (EXPSTART).
- Obtain FITS header information (e.g. airmass at end, average airmass, average seeing, guide and LOWFS star. quality measured during observation) at the end of an observation (EXPEND). This header information should flag whether there have been any periods of open loop active optics correction during the observation due to poor LOWFS star. quality.

[4] (X,Y) coordinates in the focal plane are specified in instrument coordinates in millimetres, as described in the "Software Functional Specification", [AD2]. The X axis points towards the central axis of filter wheel.

The TCS and OS between them need the ability to detect when a "significant elevation change" has taken place. The OS does not know anything about previous observations or telescope elevations, therefore the best solution would be for the TCS to make available its current elevation and its previous elevation. The OS takes the difference and decides whether to wait for a new aO measurement cycle (and convergence) before starting an observation.

3.2.4 Interface with VLT On-line Archive (VOLAC) and Data Flow Software

The Observation Software will inform the VLT on-line archive (VOLAC) each time a new observation file is available. The standard BOSS interface described in [RD38] will be used, which consists of writing the name of the new file to a "newdata" database point. A full description of the interface is described in the "*ICD between Instrumentation Software and VLT Archive System*", [RD61].

Besides telling VOLAC when new data are available, the VISTA IR Camera also needs to provide information about the structure of the data and the information supplied, in the following categories:

- Format of raw data.
- Format of header information supplied with the raw data.
- Format of observing logs and any other ancillary information saved alongside the data.

Instructions for how to supply this information and described in the "*Data Interface Control Document*", [RD62]. The VISTA IR Camera data description will be documented in an VISTA IR Camera Data interface dictionaries, [RD15].

3.2.5 Interface with ICS

The VISTA IR ICS is described in detail in the *"IR Camera Instrument Control Software Design Description Document"*, [RD5], which contains a detailed list of the commands recognised. The following features are needed on this interface:

- Put a science filter in the beam
- Put an intermediate filter in the beam at a specified angular offset.
- Put any filter in the loading position.
- Define the cryostat tube or detector target temperature, or put the ICS into "detector protection mode".
- Get FITS header items (e.g. name and ID of filter in beam, important instrument temperature and pressure readings) at exposure start (EXPSTART) and exposure end (EXPEND).
- Get the ICS status.

3.2.6 Interface with DCS

The IRACE detector controller software is described [RD44] and its requirements for the VISTA IR Camera are described in [RD11]. The IR Camera to IRACE software interface is described in detail in [RD63]. An overview is given here. The following features are needed on this interface:

- Specify exposure time.
- Specify readout mode and readout speed.
- Specify region of interest (window) to be read (from one or more science detectors – same window on each detector).
- Disable any science detector (if known to be broken).
- Specify path and file name for data, plus file naming mode.
- Set a FITS header item.
- Get FITS header items at exposure start (EXPSTART) and exposure end (EXPEND).
- Get the DCS status.

The Observation Software is responsible for completing the FITS header and for:

- specifying the location of each science detector on the focal plane;
- specifying the World Coordinates coefficients for each detector (using information obtained from the TCS, as described in Section 3.2.3 on page 22).

Broken detectors must be identified in the FITS header and the data saved in a format such that broken detectors do not cause a problem further down the data processing pipeline (see Section 5.6 on page 45).

3.2.7 Interface with HOWFS Image Analysis

This interface is new to the VISTA IR Camera, although similar to the image analysis subsystem interface of OmegaCAM, [RD67] and [RD68]. The HOWFS image analysis system is described in detail in "*IR Camera High Order Wavefront Sensing Software Design Description*", [RD7]. The following features are needed on this interface:

- Define a file containing the current HOWFS calibration measurement (DARK, FLAT).
- Trigger a HOWFS image analysis with the ANASTAR command, specifying the name of a file containing HOWFS data. This file should contain header information describing:
    - a unique ID for each exposure.
    - the location of the HOWFS star in the focal plane and the orientation of the HOWFS special filter.
    - the region of interest on the science detectors.
- Read back the HOWFS coefficients.
- Read back the HOWFS status (state, substate, health etc…).
- Get FITS header items (if any) at exposure start (EXPSTART) and exposure end (EXPEND).

The HOWFS states are: OFF, LOADED, STANDBY, ONLINE (see the "*VISTA IR Camera Software Functional Specification*", [AD2]).

3.3 *Commands*

3.3.1 Standard BOSS commands

The VISTA IR Camera Observation Software is based on the BOSS and therefore obeys all the standard BOSS commands described in the following tables. These commands are described in detail in [RD34]. Some parameters that are irrelevant for the VISTA IR Camera (such as the detector controller ID[5], -detId, are not described here).

3.3.1.7 State changing commands

The standard instrument states are described in [RD29] and [AD1]. The STANDBY state is normally used during the daytime while the instrument is running but not operational and the ONLINE state is the night time operational state. The instrument can be monitored in the STANDBY or ONLINE state, but can only be controlled from the ONLINE state. The software is in the LOADED state when it has yet to be initialised or is about to be shut down.

Command	**Parameters**	**Description**
OFF	Change state of whole instrument or subsystem to OFF. Returns –done <string>	
	-subsystem <string>	Subsystem to forward command to (optional)
ONLINE	Change state of whole instrument or subsystem to ONLINE. Returns –done <string>	
	-subsystem <string>	Subsystem to forward command to (optional)
STANDBY	Change state of whole instrument or subsystem to STANDBY. Returns –done <string>	
	-subsystem <string>	Subsystem to forward command to (optional)
STATE	Return the state/substate of the instrument or a subsystem. Returns –state <string>	
	-subsystem <string>	Subsystem whose state is to be examined

[5] Since there is only one science detector controller – the IRACE DCS.

		(optional)

3.3.1.8 Instrument configuration and observation commands

Command	Parameters	Description
ABORT	Abort all currently running actions as soon as possible. Exposures are terminated without saving data. A "WAIT" command is recommended after this command. (See also END). Returns –done <string>	
	-expoId <integer>	ID of exposure to be aborted (optional). Defaults to current exposure.
ADDFITS	Add information to the FITS header. Returns –done <string>	
	-expoId <integer>	ID of exposure whose header is to be modified (optional). Defaults to current exposure.
	-info <keyword> <value> [<keyword> <value>]	List of keywords and values to be added to the FITS header[6].
COMMENT	Add a comment to the FITS header. Returns –done <string>	
	-expoId <integer>	ID of exposure whose header is to be modified (optional). Defaults to current exposure.
	-clear	If argument present, clear any previous comment strings.
	-string <string>	String to be added to FITS header.
CONT	Continue paused exposure at a given optional time. (See PAUSE)	
	-expoId <integer>	ID of exposure to be continued (optional). Defaults to current exposure.
	-at <string>	Time when exposure to be continued (optional).

[6] These keywords must have been defined in the IR Camera Data Interface Dictionary, which is where their data types are specified.

END	End current exposure as soon as possible and save data. Cancels an exposure that has been set up but not yet started. (See also ABORT). Returns –done <string>	
	-expoId <integer>	ID of exposure to be ended (optional). Defaults to current exposure.
FORWARD	Forward command to subsystem. Returns –reply <string>	
	-subsystem <string>	Subsystem to forward command to
	-command <string>	Command to be forwarded
	-arguments <string>	Command arguments to be forwarded
PAUSE	Pause current exposure at given optional time. (See CONT). Returns –done <string>	
	-expoId <integer>	ID of exposure to be paused (optional). Defaults to current exposure.
	-at <string>	Time when exposure to be paused (optional)
SETUP	Configure the instrument. *(This is the main configuration command).* Returns –expoId <integer>	
	-expoId <integer>	ID of exposure (compulsory!). New exposure defined if specified as zero.
	-function <keyword> <value> [<keyword> <value>]	Define a list of instrument keywords and values to be assigned (optional). In particular INS.MODE declares the instrument mode.
	-file <filename>	Define a file containing a list of keywords and values to be interpreted (optional).
	-noMove	If argument present, do not move?
	-check	If argument present, check configuration only. Do not adopt it.
START	Start or repeat exposure at an optional time. Returns –done <string>	
	-expoId <integer>	ID of exposure to be started (optional). Defaults to most recently defined exposure.
	-at <string>	Time when exposure to be started (optional)

STATUS	Get the status of the functions specified in the list of arguments. Returns "–status <string>" containing "<no.of.keywords>,<keyword1>,<value1>,etc…".	
	-expoId <integer>	ID of exposure whose status is to be queried (optional). Defaults to current exposure.
	-function <string>	List of instrument keywords whose status is to be queried.
WAIT	Wait for specified exposure to finish. Returns expStatus <integer>	
	-expoId <integer>	ID of exposure to wait for (optional). Defaults to current exposure.
	-header	If argument present, wait only for the header to be collected, even if the detectors are still reading out.

3.3.1.9 Miscellaneous commands

Command	Parameters	Description
EXIT	Shut down the Observation Software control processes. Returns –done <string>	
PING	Test presence of Observation Software. Does nothing except return a reply string. Returns –done <string>	
SELFTST	Self-test the software without moving any mechanism (minor self-test). The subsystems are PINGed to ensure they are responding and then the SELFTST command is forwarded to them all in turn. Returns –done <string>	
VERBOSE	Switch verbose mode on or off. Returns –done <string>	
	ON/OFF	
VERSION	Return the OS software version. Returns –version <string>	

3.3.2 Additional VISTA IR specific commands

None known.

3.4 Dynamic Model

The sequence diagrams show how the Observation Software will accomplish various tasks. The simpler tasks are described first in chronological order, building up to a description of the more complex tasks.

3.4.1 Beginning of night systems check

Figure 5 on page 30 shows the beginning of night systems check. The Maintenance Software[7] sends a "SelfTest" command to the OS Server (???) which is forwarded in turn to all the underlying subsystems. The TCS executes a "status" command.

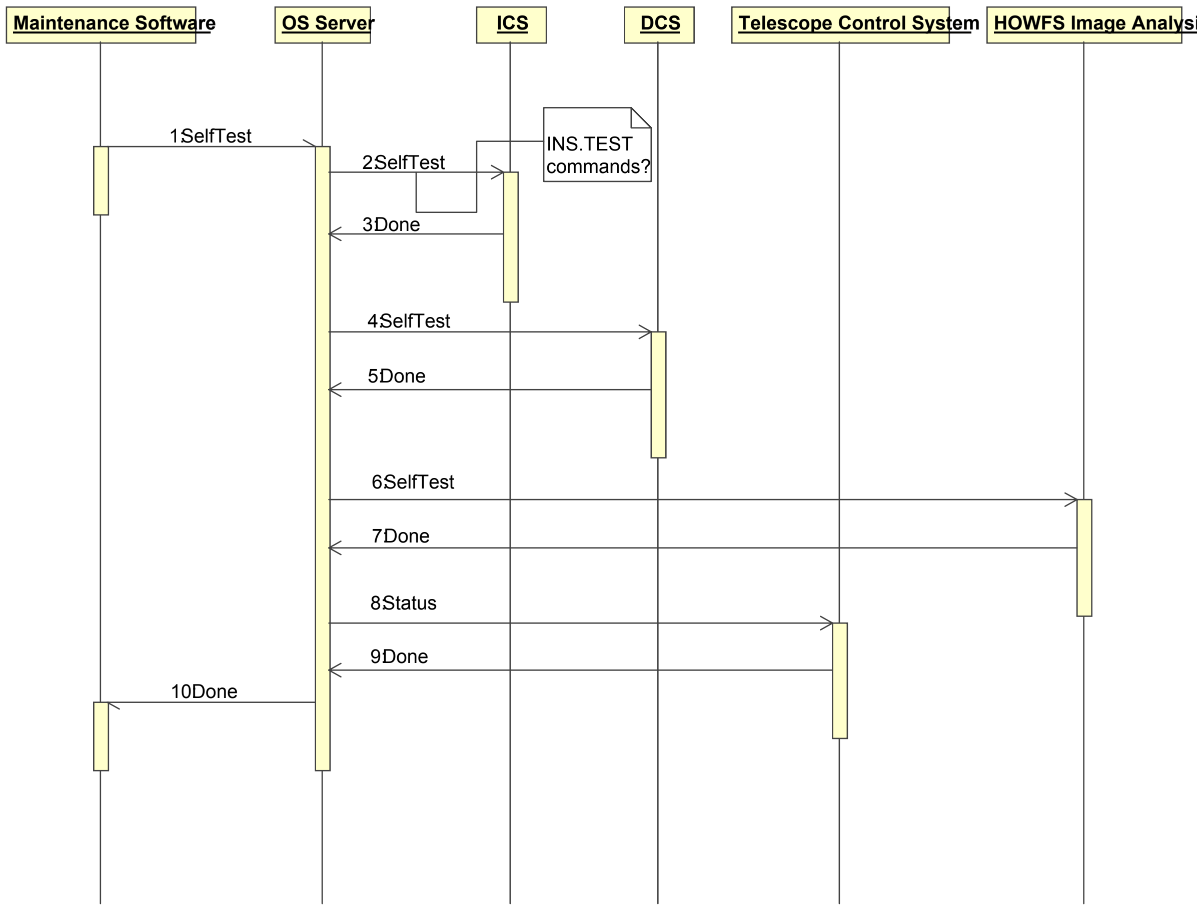

Figure 5 Sequence Diagram — Beginning of night systems check

[7] I am informed that the beginning of night self-test is run via MS software by Paranal staff.

3.4.2 Instrument setup

Figure 6 below shows how the instrument is configured. BOB sends the OS a SETUP command with a complete set of parameters describing the new instrument configuration. The OS forwards the SETUP command to the ICS, along with an ICS subset of parameters, and the ICS starts reconfiguring. The OS does not block waiting for the ICS to complete its configuration. Other operations, such as a telescope slew, could be carried out in parallel for efficiency. The detector controller and (if necessary) the HOWFS subsystems are configured in turn. When ready, the OS waits for the ICS to finish configuring. The ICS returns the focus offset and effective wavelength for the currently selected science filter, which the OS forwards to the TCS. The detector controller returns a parameter indicating the size of a single exposure made at the current configuration parameters. When everything has successfully adopted the new configuration the OS finally reports a "Done"

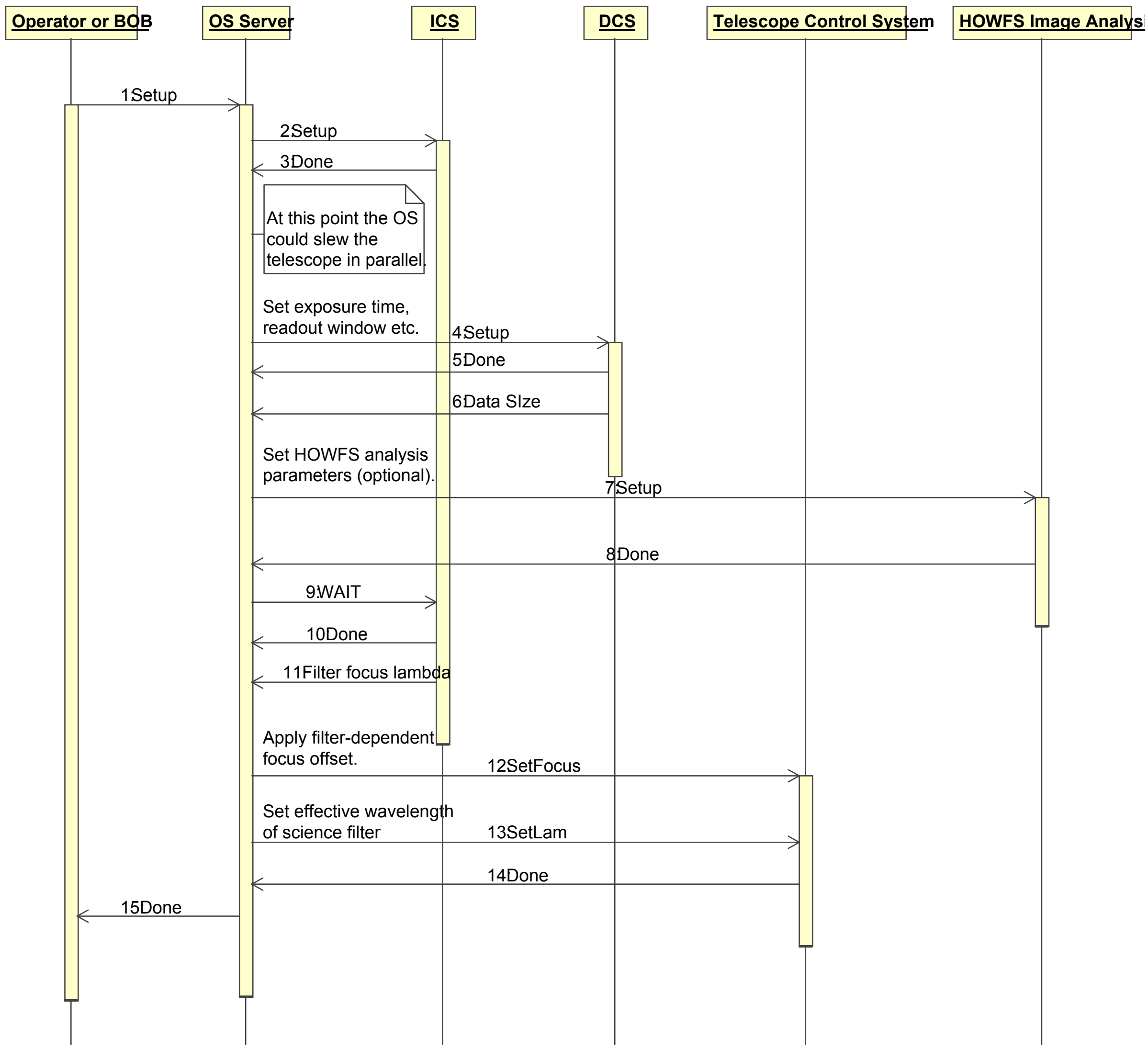

Figure 6 Sequence Diagram — Configure Instrument

3.4.3 Acquire target — Preset

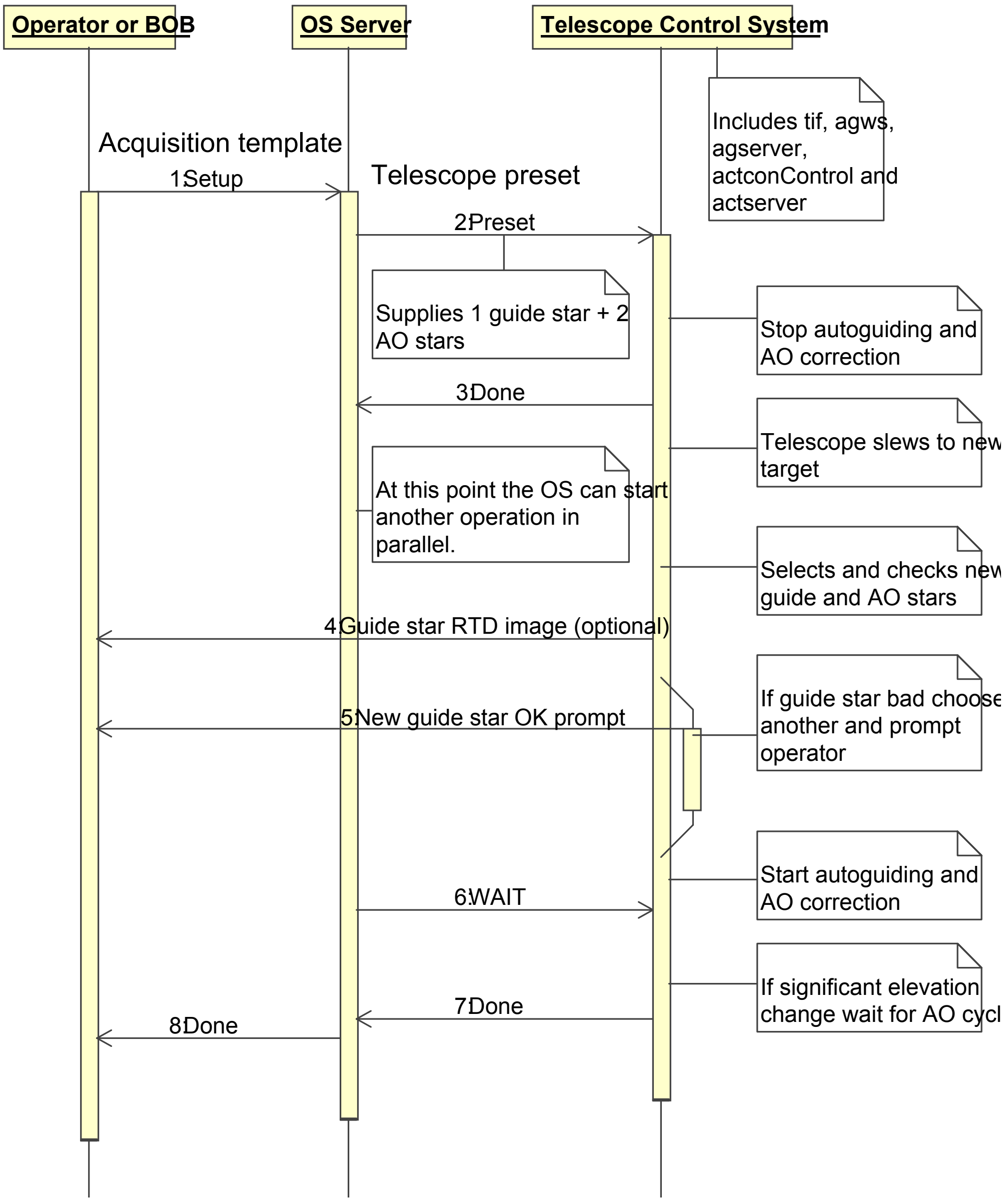

Figure 7 Sequence Diagram — Acquire Target with Preset

Figure 7 above shows the sequence of events the OS goes through to acquire a new target using a preset (i.e. acquiring a target that requires a considerable slew of the telescope). (See also Section 3.2.3 describing the OS to TCS interface). While executing a "preset" acquisition template, BOB sends the OS a SETUP command with a set of parameters describing a new acquisition target. The parameters are forwarded to the TCS as a "preset" and the TCS in turn configures its subsystems. The OS does not block waiting for the TCS to acquire the target, so it could execute other tasks in parallel. The TCS slews to the new target and automatically locks on to the chosen guide and LOWFS stars. If the guide star is found to be unsuitable a new one is chosen and the operator asked to confirm the choice. Autoguiding and active optics correction are automatically started. If a significant elevation changes has taken place the TCS could automatically wait for an aO cycle to complete (if this is necessary). When it is ready, the OS waits for the TCS to finish acquiring the target. and reports a "Done". See the *"VISTA Active Optics and Guiding Workstation Software Detailed Design",* [RD22], for a detailed description of how the autoguiding and active optics is started by the TCS.

3.4.4 Acquire target — Offset and Dither

An offset is a minor telescope pointing adjustment around an already-acquired target. This doesn't require a major slew and can therefore be carried out much more efficiently. There are two kinds of small telescope adjustments as far as the VISTA IR Camera is concerned:

- adjustments in which the telescope moves by an amount greater than the field of view (FOV) of the autoguider and LOWFS so new guide and LOWFS stars need to be chosen.
- adjustments in which the telescope moves by an amount smaller that the field of view of the autoguider and LOWFS so the same guide and LOWFS stars can be used. Such a move is known as a "dither". If the dither is made by an amount that is not a whole number of detector pixels the move is also known as a "microstep".

Figure 8 and Figure 9 below shows the sequence of events for these two kinds of telescope moves while executing an acquisition template. The main difference is that when the telescope makes a large offset new guide and LOWFS stars are provided and acquired in the same manner as a preset. It is assumed there is never a "significant elevation change" when executing an offset, so there is no need to wait for an aO cycle to complete.

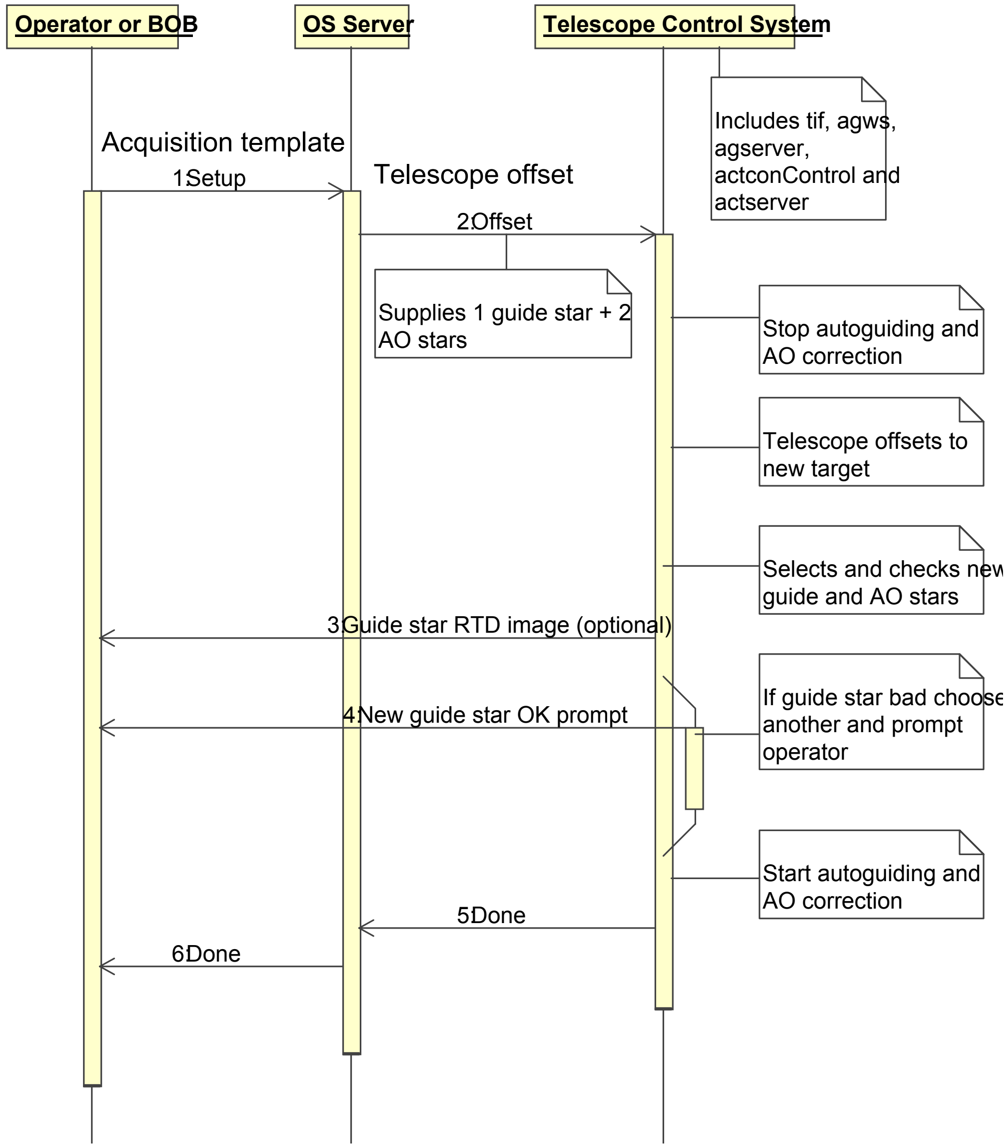

Figure 8 Sequence Diagram — Acquire Target with "Offset"

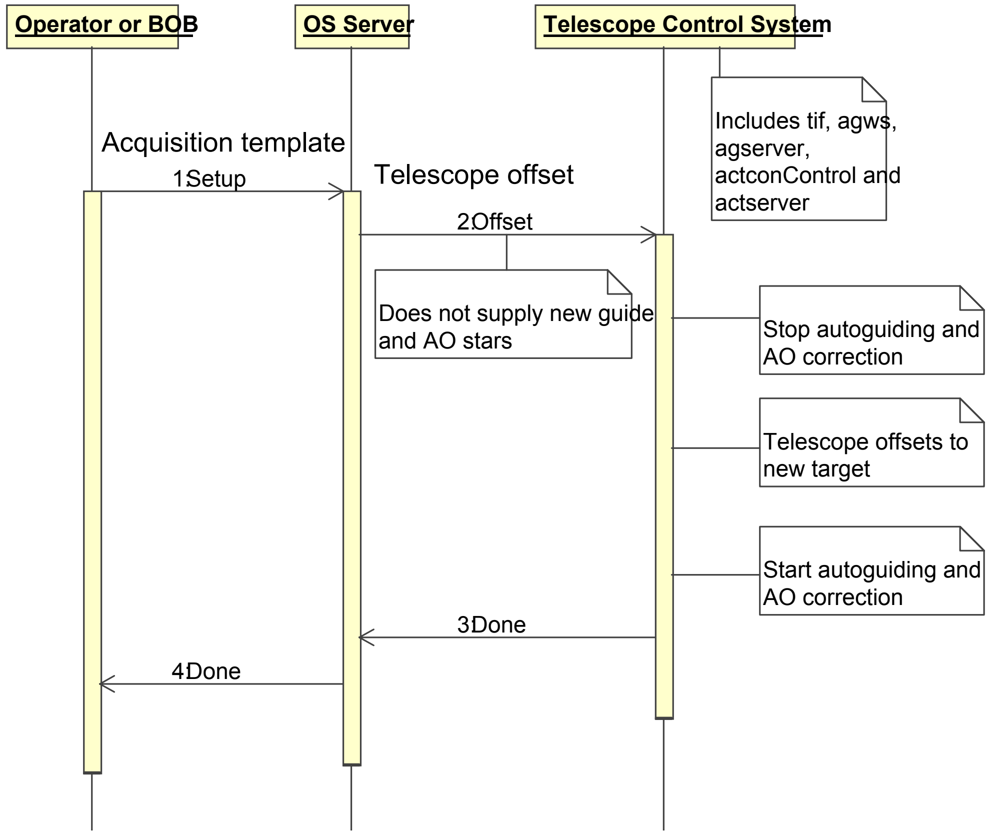

Figure 9 Sequence Diagram — Acquire Target with "Dither"

3.4.5 Autoguiding and Active Optics

Figure 10 below shows the sequence of operations during autoguiding and active optics correction, a process which happens in parallel with each science exposure. (The sequence for starting a science exposure is shown in Section 3.4.6.) Guide and LOWFS stars will already have been acquired during a preset or offset (described in Sections 3.4.3 and 3.4.4). The sequence diagram shows how the IR Camera software systems interact with themselves and with the TCS[8] while autoguiding and active optics are operational. Details of the interaction between the TCS and agServer and actServer are not shown here, and the reader is referred to the *"VISTA Active Optics and Guiding Workstation Software Detailed Design"*, [RD22], for a complete description of how autoguiding and active optics is handled by the TCS.

Significant things to note on this diagram are that the agServer and actServer processes are constantly reporting the quality of the autoguider and LOWFS data to the TCS. If the quality falls below an acceptable threshold it is assumed the TCS will report this to the operator and will write a flag into the header it saves with the EXPEND command (Sections 3.2.3 and

[8] Note that in the context of this document "TCS" refers to everything in the TCS that is not part of the camera software — i.e. the tiff, agws and actconControl, etc.

5.6). Tracking will continue open-loop if the LOWFS data quality is unacceptable. The other significant thing is that the agServer and actServer are synchronising their readout with the IRACE DCS controller (with the DCS as master) to ensure they don't interfere with each other. This synchronisation will only be necessary if there is EMI between these detectors. See [RD13] for a description of the readout synchronisation strategy.

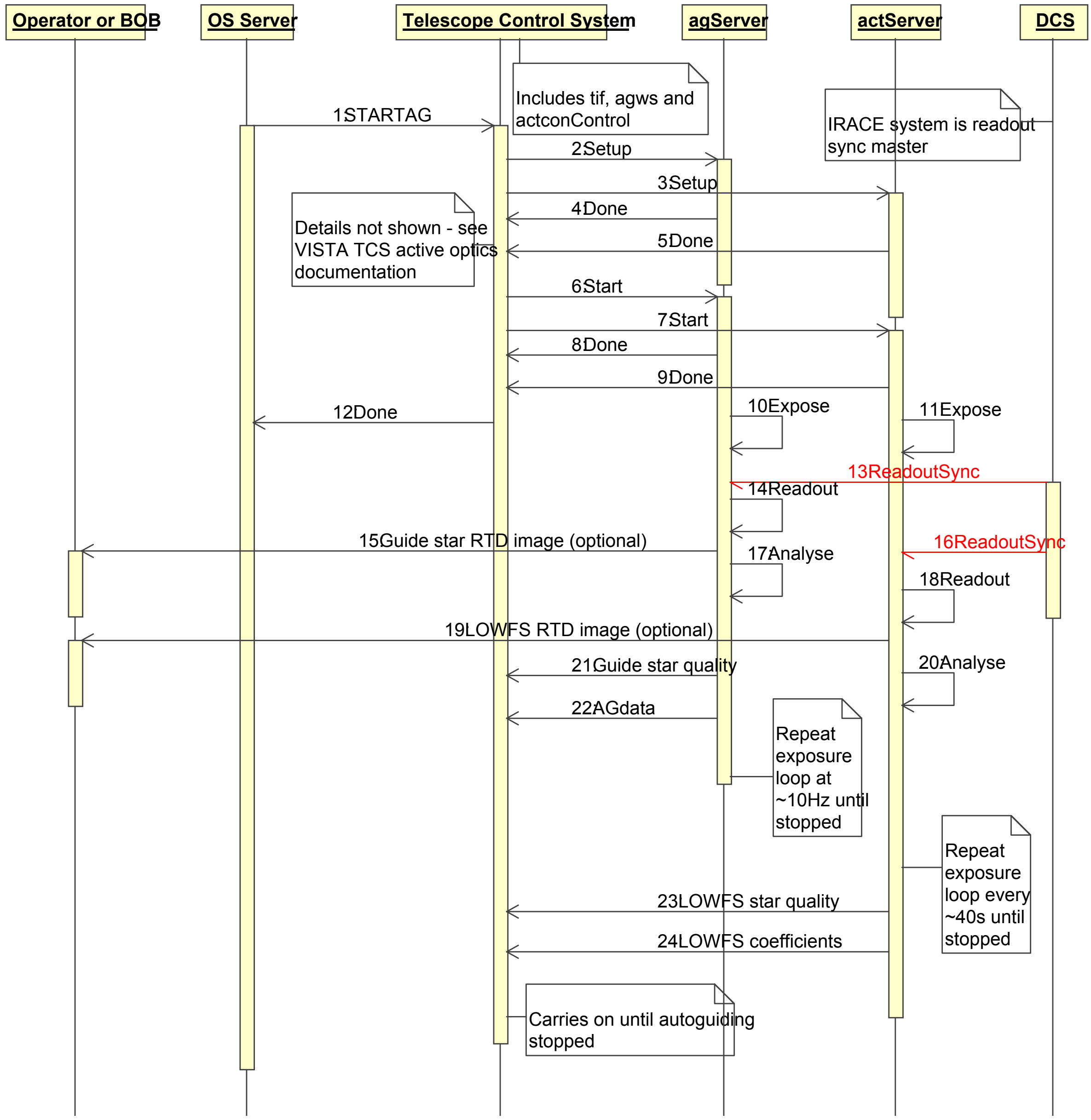

Figure 10 Sequence Diagram — Autoguiding

3.4.6 Science exposure

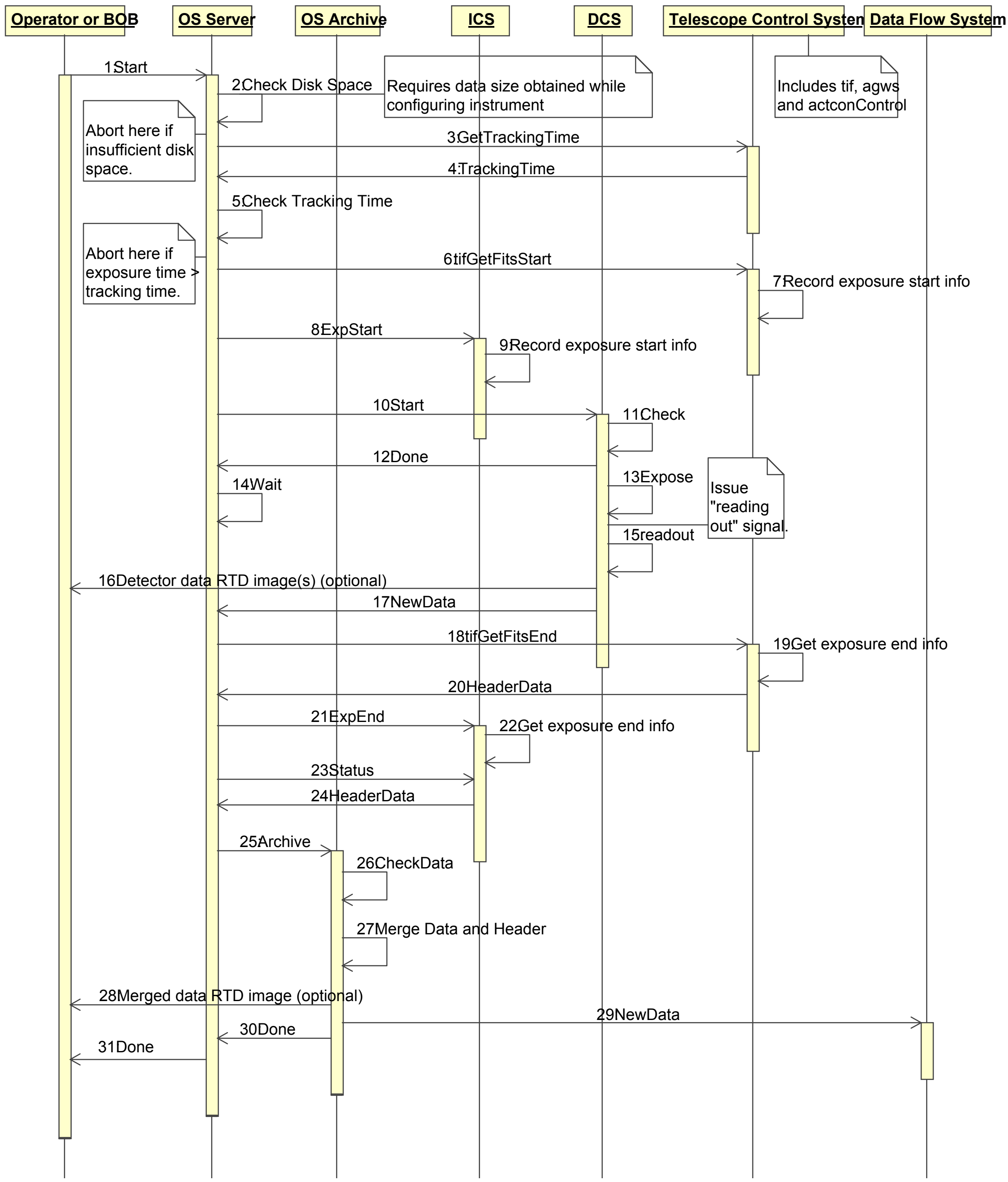

Figure 11 Sequence Diagram — Single Science Exposure

Figure 11 above shows the sequence for making a single science exposure. It is assumed that the target has already been acquired and autoguiding started as described in the previous sections. The Observation Software receives the START command and obtains the available tracking time for the current target from the TCS. If the exposure time exceeds the available tracking time the exposure is aborted. The exposure is also aborted if there is insufficient disk space to store the data. Before starting the actual exposure, the TCS and ICS are instructed to save FITS header data. The DCS is then commanded to start the exposure. The Operator can monitor the exposure using the DCS real-time display. When the exposure completes the DCS saves the raw data and informs the OS the exposure has completed. The OS then instructs the TCS and ICS to save any FITS header information relevant to the end of the exposure. The OS Archive process is then instructed to process the data. The FITS header information is merged with the raw data and the combined data is passed on to the Data Flow System (via VOLAC).

3.4.7 HOWFS exposure

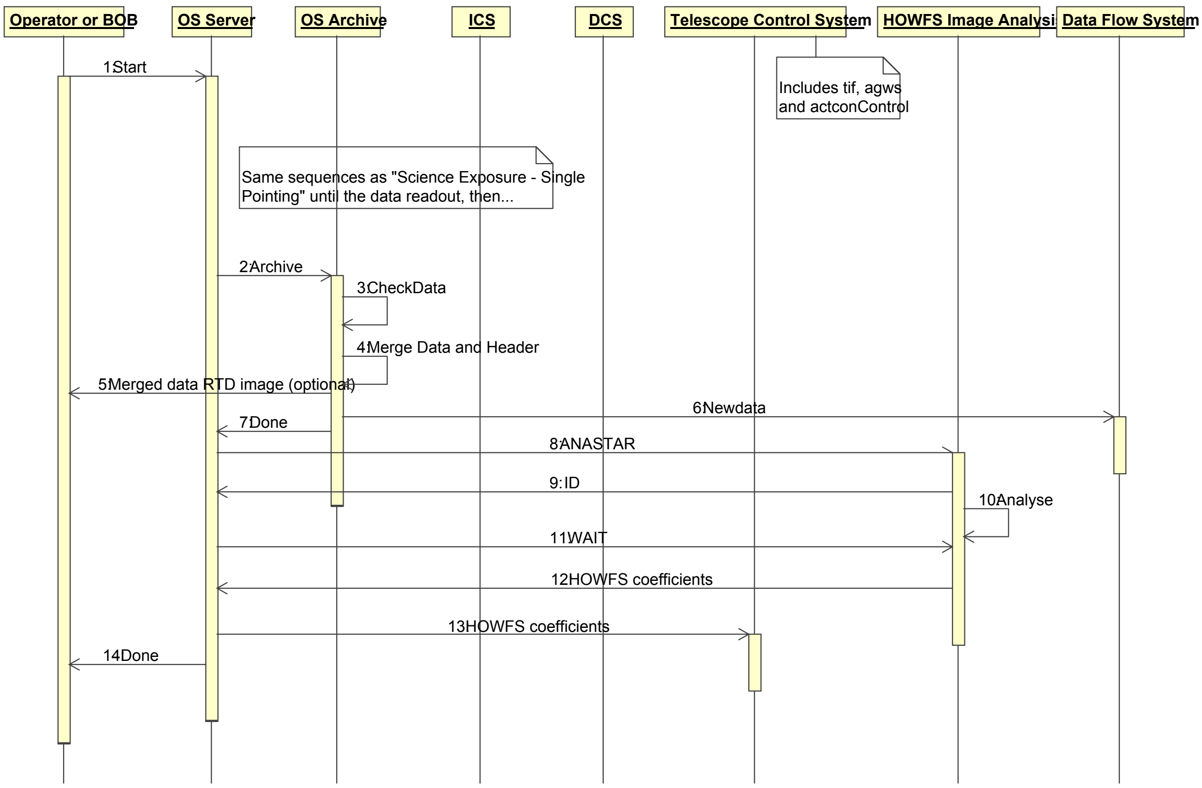

Figure 12 Sequence Diagram — HOWFS Exposure

Figure 12 shows the sequence of events that make up a HOWFS exposure. The sequence is exactly the same as for a science exposure, except that the Observation Software forwards the data to the HOWFS image analysis subsystem. This subsystem makes available the HOWFS coefficients, which are then forwarded to the TCS. The raw HOWFS data may be archived by the data flow system.

3.4.8 Multiple exposures with dithers

The Observation Software goes through the following procedure when making a compound observation out of multiple exposures separated by very small telescope offsets, known as "dithering":

1. The sequence begins with an acquisition template. The telescope is preset and slews to the new target, corresponding to the first pointing making up the observation. 3 guide/LOWFS stars are supplied and the TCS locks onto the guide/LOWFS stars (Figure 7).

2. At the same time the instrument is configured (Figure 6).

3. Autoguiding starts (Figure 10).

4. An exposure is made at the first pointing (Figure 11).

5. The telescope is dithered to the coordinates for the next pointing. The same set of three guide/LOWFS star candidates are supplied and autoguiding continues (Figure 9).

6. Steps 4 and 5 are repeated until the last exposure of the sky-flat is made. Autoguiding is then stopped.

7. The Observation Block completes.

This same sequence can be used for any kind of dithered observation. For example:

- building up a sky-flat made of a stack of exposures whose star images are eliminated by median filtering;
- a "deep stack" of observations made at slightly different target centres designed to be coadded together with a "shift-and-add" algorithm to eliminate bad pixels.

A separate data file is saved for each exposure, but the exposures are linked by keywords in their FITS header, as described in Section 5.6.3.

3.4.9 Multiple exposures with microsteps

These exposures are built in exactly the same way multiple exposures with dithers, except that the telescope is offset by a fraction of a pixel with the intention of combining the exposures into a larger more finely sampled data array.

The main difference between dithered exposures and microstepped exposures is that the microstep pattern is recorded in different FITS header parameters (see Section 5.6.3), so the data reduction pipeline knows to combine the data in a different way.

3.4.10 Multiple exposures with dithers and microsteps

It is possible to take data while dithering and microstepping at the same time. In this case dither pattern will contain separate microstepping and dithering instructions, and the microstepping and dithering parameters recorded in separate FITS header parameters (see Section 5.6.3).

3.4.11 Multiple exposures with offsets

The Observation Software goes through the following procedure when making a compound observation out of multiple exposures separated by telescope offsets:

1. The sequence begins with an acquisition template. The telescope is preset and slews to the new target, corresponding to the first pointing making up the observation. 3 guide/LOWFS stars are supplied and the TCS locks onto the guide/LOWFS stars (Figure 7).

2. At the same time the instrument is configured (Figure 6).

3. Autoguiding starts (Figure 10).

4. An exposure is made at the first pointing (Figure 11). *NOTE: At this point an entire microstepping/dithering sequence could be carried out, as described in Sections 3.4.8 and 3.4.9 above.*

5. The telescope is offset to the coordinates for the next pointing. Autoguiding is stopped and a new set of three guide/LOWFS star. candidates are supplied, which the TCS then locks onto (Figure 8).

6. Autoguiding restarts (Figure 10).

7. Steps 4, 5, 6 are repeated until the last exposure of the tile is made. Autoguiding is then stopped.

8. The Observation Block completes.

The most common example of offsetting is building up a complete tile (i.e. a set of exposures at different telescope offsets designed to fill in the gaps between detectors), as shown in the "*VISTA IR Camera Software Functional Specification*", [AD2].

Again, a separate data file is saved for each exposure, but the exposures are linked by keywords in their FITS header, as described in Section 5.6.3.

4 Detailed Design

The Observation Software decomposes into three parts, as shown in Figure 2 on page 18. The vcoControl process is based on the bossSERVER class supplied by the "Base Observation Software Stub" (BOSS) package [RD38]. The bossSERVER uses a variety of subsystem interface classes to communicate with different kinds of subsystem, as shown in Figure 13. The VISTA IR Observation Software needs to communicate with a HOWFS image analysis subsystem for which BOSS does not have a predefined class. A new interface class will therefore be derived from the bossINTERFACE base class, as shown in Figure 13.

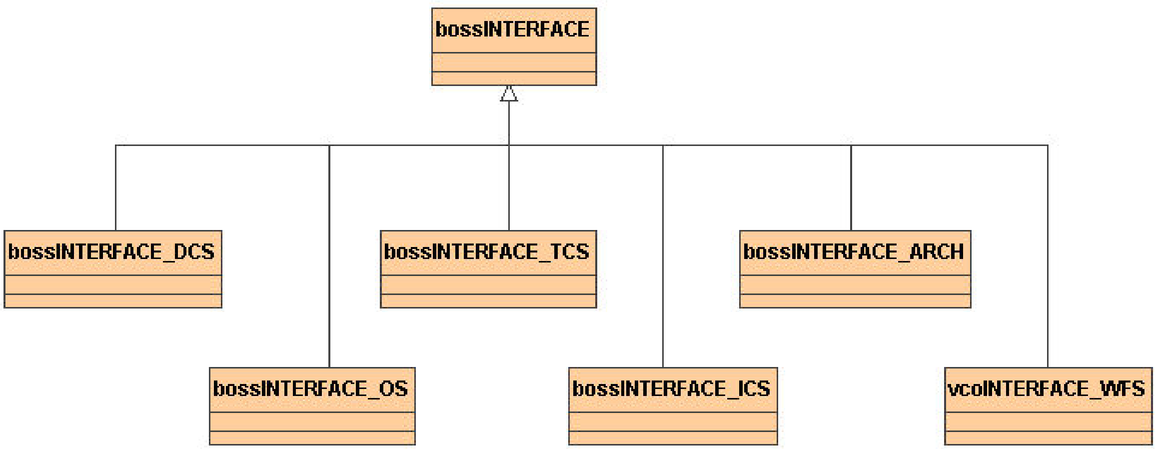

Figure 13 Class diagram showing relationship of vcoINTERFACE_WFS

4.1 Modules

The Observation Software breaks down into the following modules:

- **vco** — OS control module. This module contains the main control process (vcoControl) and the archiver process (bossARCHIVER_vco)
- **vcopan** — OS graphical user interface module
- **vcotsf** — OS template signature module
- **vcoseq** — OS sequencer script module
- **vcmcfg** — Instrument configuration module (for some reason classified as maintenance software)

4.2 OS Control (vcoControl)

This process is responsible for managing the subsystem interfaces. It receives commands from BOB, breaks the commands into those components targeted at the various subsystems and dispatches the commands to the subsystems. This process also listens to the underlying subsystems and generates a combined instrument status.

4.3 Archiver

This process is responsible for dealing with the data generated by the IR Camera. It receives the raw data from the IRACE DCS, merges the header information saved by the subsystems with the data and notifies the VOLAC system of the availability of the new data.

This process also needs to manage a real-time display which presents a low resolution view of all 16 detectors to the observer. The ESO-VLT real-time display utility, [RD45], should be able to produce this display as long as the data header contains sensible World Coordinates information.

5 DATA DESCRIPTION

5.1 Configuration files

The VISTA IR Camera software is configured using the following configuration files, which are stored in $INS_ROOT/SYSTEM/COMMON/CONFIGFILES. See the configuration tool documentation, [RD40]:

- **vcmcfgCONFIG.cfg** — The top level configuration file, which defines all other configuration sets (i.e. configuration file and dictionary references) to be used with the instrument.

- **vcmcfgINS.cfg** — This file contains the main configuration keywords for the VISTA IR camera, as listed in Section 5.2 below.

- **vcmcfgSTART.cfg** — This file contains the startup configuration keywords for the VISTA IR camera, such as the INS.CON.OPMODE simulation mode keyword. All the startup options are described in the BOSS documentation, [RD38], and startup tool documentation, [RD41].

5.2 Observation Software Configuration Keywords

The Observation Software uses the configuration keywords listed and described in the BOSS documentation, [RD38], and Template Instrument documentation, [RD37]. A full list is not repeated here but attention is drawn to keywords that have a special meaning for the VISTA IR Camera.

5.2.1 TCS subsystem configuration keywords

Keyword	Type	Description
OCS.TEL.NAME	string	Telescope name = 'VISTA' (not 'UT1', 'UT2', 'UT3' or 'UT4' as specified in the BOSS manual). 'UT0' is used for a simulated TCS.
OCS.TEL.FOCUS	string	Telescope focus position = 'CA' (VISTA only has a Cassegrain focus).
OCS.TEL.ID	string	TELESCOP FITS keyword defined by DICB (e.g. 'VISTA' – to be agreed).
Etc…		

5.2.2 ICS subsystem configuration keywords

Keyword	Type	Description
OCS.ICS.NAME	string	Instrument control system name = 'ICS'.
Etc…		

5.2.3 DCS subsystem configuration keywords

Keyword	Type	Description
OCS.DET1.NAME	string	Science detector controller name = 'IRDCS'.
OCS.DET1.TYPE	string	Science detector controller type = 'IRACE'
Etc…		

5.2.4 HOWFS image analysis subsystem configuration keywords

Keyword	Type	Description
OCS.WFS.NAME	string	HOWFS image analysis subsystem name = 'HOWFS'
OCS.WFS.DICT1	string	HOWFS subsystem dictionary = 'VIRCAM_WFS'
OCS.WFS.KEYWFILT	string	Keyword filter = 'WFS.*.*.*.*.*.*'
Etc…		

5.3 Instrument Operating Modes

The VISTA IR Camera will have the following instrument modes:

- **IMAGING** — An observation is made with the science detectors and the data file is passed to VOLAC. This mode is used for most observations.

- **HOWFS** — An observation is made with the science detectors using a special intermediate filter and a small region of interest. The data file is passed to the HOWFS analysis subsystem.

The HOWFS coefficients are then passed to the TCS.

NOTE: There isn't a separate mode for autoguider or LOWFS imaging because in normal operation these detectors are controlled by the telescope software.

5.4 *Instrument Path*

ESO-VLT instruments define a path for each light path through the instrument leading to a unique detector (e.g. for instruments with separate visible and infrared channels). The VISTA IR Camera has only one path:

- **INFRARED** — Light recorded on the infrared science detectors.

NOTE: There aren't separate instrument paths for the autoguider or LOWFS pre- and post-focus detectors because in normal operation these are controlled by the telescope software.

5.5 *Database structure*

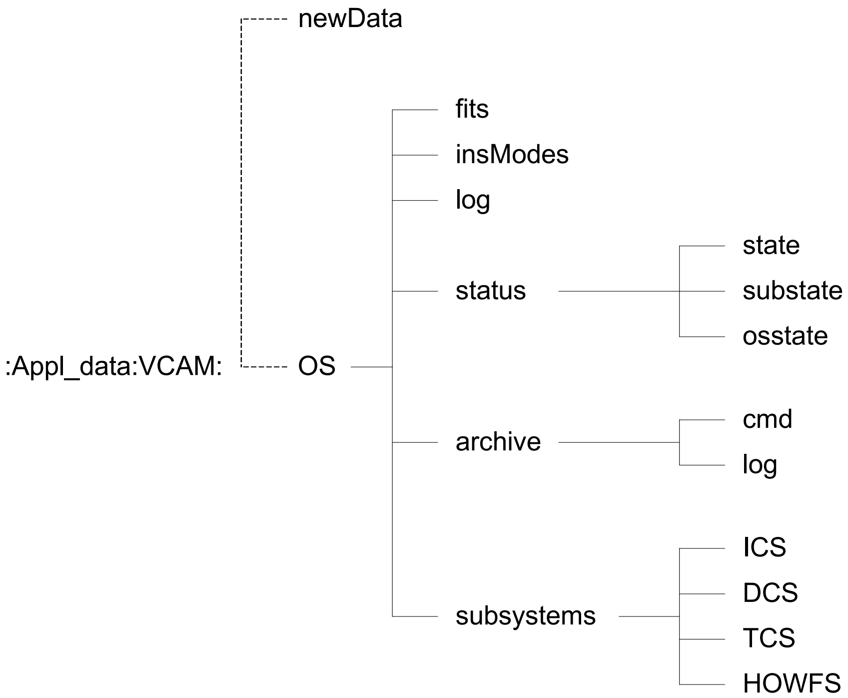

Figure 14 Top level database structure for the OS Control task.

Figure 14 shows the overall layout of the database structure for the VISTA IR Observation Software. The database follows the standard layout used by the Template Instrument. The database contains the state, substate and status of all the subsystems. The "newData" database point is outside the scope of the Observation Software database but is shown here because it is used to inform VOLAC of the availability of new data files.

5.6 *Data files*

Data files will be stored in $INS_ROOT/SYSTEM/DETDATA and will follow the naming convention described in *"Common Software for Templates",* [RD39]. In general, file names have the form

<INST>_<mode>_<type>_<doy>_nnnn[_<extension>].fits

Where:

- <INST> = Instrument name (uppercase) ("VIRCAM" for the VISTA IR Camera)
- <mode> = Instrument mode (Section 5.3)
- <type> = type of observation as defined in keyword DPR.TYPE (e.g. DARK, FLAT, etc…)
- <doy> = sequential day of the year
- nnnn = unique sequential number
- <extension> = Optional file name extension.

Compare with template names in Section 8.1.

5.6.1 Input files

The OS receives raw data as stored by the IRACE system (see the "*Detector Controller Technical Specification*", [RD11], and "*IR Camera to IRACE Software ICD*", [RD63]).

- Raw IRACE data: Multiple FITS files, one for each detector, containing minimal header information stored by the IRACE software. Each file either contains the full (2028 x 2048 pixel) data array from each detector or a smaller array if a region of interest has been specified. All the detectors are read out, even if the OS is only interested in one, and all the detectors have the same region of interest and therefore data arrays of identical size and shape.

All the data files must be readable even if one of the science detectors is broken. See Section 3.2.6 on page 25.

5.6.2 Output files

The OS archiver processes the raw data received from IRACE. It combines the individual files into one FITS file containing one image extension per detector. It also adds the header information supplied by all the instrument subsystems in response to the EXPSTART and EXPEND commands (see Section 3.2) and adds World Coordinate information to the data. The result is:

- A single raw science exposure: A multi-extension FITS file with complete header information added by the OS. The file has one extension for each detector, but for some (e.g. HOWFS) observations only one of the FITS extensions will contain useable data.

The data header is specified in the VISTA IR Camera Data Interface Dictionaries, [RD15], and will conform to the requirements specified in the ESO-VLT *"Data Interface Control Document",* [RD62].

5.6.3 Groups of files

Many of the files processed by the OS will be part of a group created by the execution of an observation block containing multiple templates, such as tile observations (Section 8.5.3). These files will be identified as being part of the group by means of these standard header keywords (see [RD15] for details).

OBSNUM	Observation number
GRPNUM	Group number
GRPMEM	Group membership (T/F)
NOFFSETS	Number of offsets
OFFSET_I	Serial number of offset
NDITHER	Number of positions in a dither pattern
DITHRNUM	Value of first OBSNUM in current dither sequence
DITHER_I	Serial number of dither pattern
DITHER_X	X offset in dither pattern
DITHER_Y	Y offset in dither pattern
NUSTEP	Number of positions in a microstep pattern
USTEPNUM	Value of first OBSNUM in current microstep sequence
USTEP_I	Serial number of microstep pattern
USTEP_X	X offset in microstep pattern
USTEP_Y	Y offset in microstep pattern
OBS.GRP	Observation Block Group ID (if any)
OBS.ID	Observation Block ID
OBS.NAME	Observation Block name
OBS.TPLNO	Template number in Observation Block
TPL.ID	Template ID
TPL.NAME	Template name
TPL.NEXP	Number of exposures in template
TPL.EXPNO	Exposure number in template
DPR.CATG	Observation category (TEST, CALIB or SCIENCE)
DPR.TECH	Observation technique (HOWFS, SNAPSHOT, TILE, etc…)
DPR.TYPE	Observation type (BIAS, DARK, DOMEFLAT, etc…)

The data flow pipeline can use these keywords to work out how to process the files.

5.7 *Dictionary*

The instrument dictionary is described in the VISTA IR Camera Data Interface Dictionaries, [RD15].

5.8 *Log files*

The Observation Software generates the following log files:

- Nightly observation log, describing all the major events that happened during a night's observing, all the observations made, weather information and operator comments.

- Observation Software engineering log, reporting a more detailed list of events and fault reports with diagnostic information.

The logging is carried out using the CCS logging system, as described in the *"CCS User Manual"*, [RD45].

6 PHYSICAL DEPLOYMENT

The Observation Software runs on an ESO-VLT Instrument Workstation, as shown in the "Software Functional Specification", [AD2]. It requires a powerful version of the workstation to achieve the necessary data rate (see next section).

7 PERFORMANCE ANALYSIS

The VISTA IR Camera has to store each 268 Mbyte exposure within 5 seconds, leading to a peak data rate requirement of 53.7 Mbytes / second. It also has to be able to sustain an exposure every 10 seconds for 14 hours, leading to a sustained data rate requirement of 26.8 Mbytes / second and a total data storage per night in excess of 1.35 Tbytes (see [AD2] for details).

Most of the onus on achieving this data rate falls on the IRACE system, which is responsible for making the exposures and saving the raw data. Assuming that the delivered IRACE system achieves its performance requirement, the Observation Software must be capable of performing the following operations:

- Adding the FITS header information (including World Coordinates information).
- Displaying the combined data (with World Coordinates information) to the operator via an ESO-VLT Real Time Display (RTD) [RD45], (if required).
- Passing the data to VOLAC.

At the maximum sustained data rate. This may entail the OS using the most powerful ESO-VLT Instrument Workstation available at the time. An analysis by the OmegaCAM project, [RD67], showed that the speed of the network and the speed of the disks were the most important factors in achieving their target data rate.

8 Templates

8.1 Overview of Templates

As mentioned in Section 3.2.2 on page 21, the Observation Software supplies the ESO-VLT Observation Handling software with:

- Template signature files
- Template sequencer scripts
- Reference setup files
- Instrument summary file

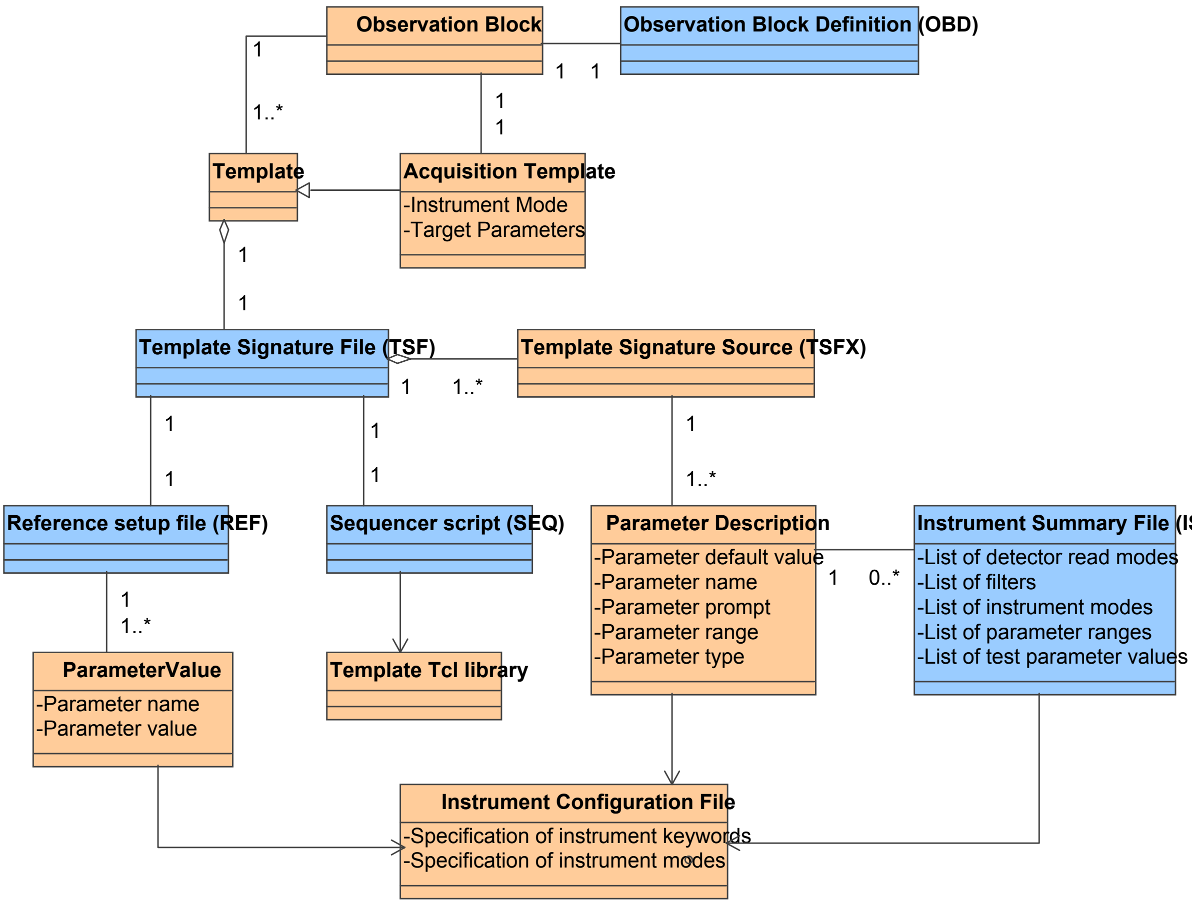

Figure 15 Relationship Between ESO-VLT Observation Block Entities

Figure 15 above shows the relationship between the various template files and the Observation Blocks processed by BOB. Specifically, there should be a template for each different kind of exposure that the instrument is capable of making. Each template is associated with an unique way of making an observation with the instrument, and is associated with a *reference setup file* that contains a list of the parameter keywords and values needed to set up the instrument for making that kind of observation. (For example, the reference setup file for a HOWFS observation will select the HOWFS intermediate filter). A template can incorporate one or more *template signature source files*, and these contain descriptions of the parameters associated with different instrument components (e.g. the filter mechanism). Template signature source files can refer to the instrument summary file, which contains a comprehensive description of the capabilities of the instrument — the allowed ranges for various instrument configuration parameters and the choices an observer can make.

The naming convention for template signature files is defined in *"Common Software for Templates",* [RD39]. Each file name has the form

<INST>_<mode>_<type>[_<description>].tsf

Where:

- <INST> = Instrument name (uppercase) ("VIRCAM" for the VISTA IR Camera)
- <mode> = Instrument mode ("gen" can be used for maintenance templates)
- <type> = type of template (acq – acquisition, obs – observation, cal – calibration, tec – technical)
- <description> = optional string of up to 16 characters identifying the purpose of the template, such as (bias - bias exposure, dark - dark exposure, domeflat - dome flat-field, skyflat - sky flat-field, wave - wavelength calibration, exp - science exposure, std - standard star exposure, etc…)

Figure 16 below shows the hierarchy of the templates defined for the VISTA IR Camera. There are a series of templates for each of the operating modes described in Section 5.3. In each mode there are three kinds of templates:

- Acquisition templates, which define the operating mode and telescope target parameters.
- Calibration templates, which obtain exposures necessary for calibrating observations in a particular instrument mode.
- Observation templates, which obtain the exposures necessary to make science observations.

The hierarchy shows templates which are derived from a parent (for example VIRCAM_img_cal_linearity is made from repeated operations of VIRCAM_img_cal_domeflat). The various templates will now be described in more detail

- HOWFS
  - VIRCAM_howfs_acq
  - VIRCAM_howfs_cal_dark
    - VIRCAM_howfs_cal_bias
  - VIRCAM_howfs_cal_flat
  - VIRCAM_howfs_obs_exp
    - VIRCAM_howfs_obs_wfront
- IMAGING
  - VIRCAM_img_acq
    - VIRCAM_img_acq_offset
      - VIRCAM_img_acq_dither
    - VIRCAM_img_acq_sky
    - VIRCAM_img_acq_domescreen
  - VIRCAM_img_cal_badpixels
  - VIRCAM_img_cal_dark
    - VIRCAM_img_cal_bias
    - VIRCAM_img_cal_darkcurrent
  - VIRCAM_img_cal_domeflat
    - VIRCAM_img_cal_linearity
  - VIRCAM_img_cal_sky
    - VIRCAM_img_cal_skyflat
  - VIRCAM_img_obs_exp
    - VIRCAM_img_obs_snapshot
    - VIRCAM_img_obs_telcalib
  - VIRCAM_img_obs_dither
    - VIRCAM_img_obs_microstep

Figure 16 Hierarchy of VISTA IR Camera Templates

8.2 HOWFS Templates

These templates will be used to collect HOWFS data.

8.2.1 Acquisition templates

- **VIRCAM_howfs_acq** — Acquire a HOWFS source. This template will set the instrument into HOWFS mode and point the telescope to a HOWFS standard star, specifying the sky coordinates of the star, the required (X,Y) in the instrument focal plane, the required rotator angle and other target acquisition parameters described in Section 8.3 on page 51. If autoguiding and active optics correction are required one guide star and two LOWFS stars are specified. The guide star is checked automatically and if found unsuitable the operator is given the chance to select a new one.

8.2.2 Calibration templates

- **VIRCAM_howfs_cal_dark** — This template makes a DARK exposure suitable for calibrating a HOWFS observation. The detector controller is configured with the same readout window as the target observation. The DARK filter is selected and an exposure made at the same exposure time as the intended observation.

- **VIRCAM_howfs_cal_bias** — This template makes a BIAS exposure suitable for calibrating a HOWFS observation. The detector controller is configured with the same readout window as the target observation. The DARK filter is selected and a minimal exposure made.

- **VIRCAM_howfs_cal_flat** — This template makes a flat-field exposure suitable for calibrating a HOWFS observation. The detector controller is configured with the same readout window as the target observation. The telescope is moved to acquire the dome flat-field and the calibration source is turned on. The HOWFS intermediate filter is selected (and positioned to the same orientation as the observation to be made) and an exposure made.

8.2.3 Observation templates

- **VIRCAM_howfs_obs_exp** — This template makes a HOWFS wavefront measurement. The detector controller is configured with a suitable readout window and the HOWFS intermediate filter is selected and positioned over the required detector. A HOWFS observation is made and when completed the HOWFS image analysis system is triggered. The coefficients are saved for future use.

- **VIRCAM_howfs_obs_wfront** — The same procedure as VIRCAM_howfs_obs_exp except when the analysis is finished the coefficients are read back and forwarded to the TCS.

8.3 Imaging Templates

These templates will be used to make imaging observations with the VISTA IR Camera.

8.3.1 Acquisition templates

- **VIRCAM_img_acq** — This is the basic VISTA IR target acquisition template. It will set the instrument into IMAGING mode and point the telescope to a new target (using a "preset"), specifying the sky coordinates of the field centre, the required rotator angle and other target acquisition parameters described in Section 8.3 on page 51. If autoguiding and active optics correction are required one guide star and two LOWFS stars are specified. The guide star is checked automatically and if found unsuitable the operator is given the chance to select a new one. If a significant elevation change has taken place there may be a pause until the LOWFS has completed a wavefront measurement cycle and the measurement has settled down.

- **VIRCAM_img_acq_offset** — This template is used for acquisition of a target which is near enough to the previous target for the telescope to be moved by an offset. The template is the same as VIRCAM_img_acq except the telescope is moved using an "offset" command. A new set of guide and LOWFS stars are selected in the normal way.

- **VIRCAM_img_acq_dither** — A variation of VIRCAM_img_acq_offset where the telescope is moved by an amount smaller than the field of view of the wavefront sensors. The same guide and LOWFS stars can be used for the new observation and new ones do not need to be selected.

- **VIRCAM_img_acq_sky** — This template is used to select a sky field. It is the same as VIRCAM_img_acq except the operator is given the opportunity to confirm there are no bright stars in the field. (The coordinates could be selected from a list of pre-defined blank sky fields).

- **VIRCAM_img_acq_domescreen** — This template is used to move the telescope to point at the flat-field screen in the dome. Telescope tracking is turned off and the required illumination level defined.

8.3.2 Calibration templates

- **VIRCAM_img_cal_dark** — This template makes a DARK exposure suitable for calibrating an imaging observation. The DARK filter is selected and an exposure made at the same exposure time as the science observation it is intended to calibrate.

- **VIRCAM_img_cal_bias** — This template makes a BIAS exposure suitable for calibrating an imaging observation. The DARK filter is selected and a minimal exposure made.

- **VIRCAM_img_cal_darkcurrent** — This template makes a detector dark current measurement by repeating the VIRCAM_img_cal_dark template at a variety of different exposure times.

- **VIRCAM_img_cal_badpixels** — This template examines a series of dark and/or flat-field observations suitable for detecting bad detector pixels.

- **VIRCAM_img_cal_domeflat** — This template makes a dome flat-field exposure suitable for calibrating an imaging observation. The telescope is moved to point towards the flat-field screen (using VIRCAM_img_acq_domescreen) and the calibration source is turned on. A suitable science filter is selected and an exposure made. The calibration source is then switched off

- **VIRCAM_img_cal_linearity** — This template makes a detector linearity measurement by repeating the VIRCAM_img_cal_domeflat template at a variety of different exposure times.

- **VIRCAM_img_cal_sky** — This template makes a sky background exposure suitable for calibrating an imaging observation (such as crowded field observations made at low Galactic latitude). The telescope is moved to a suitable sky field (using VIRCAM_img_acq_sky) and a suitable science filter is selected. A pre-defined dithering pattern is selected and the telescope is dithered to take a pattern of NDITHER exposures on the sky. The dithered exposures are median-filtered to remove stars.

- **VIRCAM_img_cal_skyflat** — This template is the same as VIRCAM_img_cal_sky except that the sky field is observed in twilight and the dithered exposures combined to make a sky flat-field suitable for calibrating an imaging observation.

8.3.3 Observation templates

- **VIRCAM_img_obs_stare** — This is the basic imaging template. A single observation is made at one "pointing" on the sky. A science or intermediate filter is selected and the detector controller configured with readout and exposure time parameters. The exposure is made and the data passed on to VOLAC.

- **VIRCAM_img_obs_snapshot** — The same as VIRCAM_img_obs_stare with a science filter tray selected and the detector controller configured to save the data from the entire 4 x 4 grid of detectors (or as many of the detectors as are functional).

- **VIRCAM_img_obs_telcalib**— The same as VIRCAM_img_obs_stare with an intermediate filter selected and the detector controller configured to save only the part of the data illuminated by the intermediate filter. Used for telescope pointing calibration observations.

- **VIRCAM_img_obs_dither** — This template makes a series of observations on the sky with a specified dither pattern of NDITHER exposures in a similar manner to VIRCAM_img_cal_skyflat.

- **VIRCAM_img_obs_microstep** — A variation of VIRCAM_img_obs_dither in which a set of observations are made in a microstep pattern of NUSTEP exposures (with telescope offsets of a fraction of a pixel) and (optionally) overlaid with a dither pattern of NDITHER dithered exposures. Microstepped data and dithered data are collected in the same way. The only difference is in the way the data are processed.

The golden rule with these observation templates is that one template will only require one set of guide and LOWFS stars. Observations that require more than one set of guide and LOWFS stars, such as the building up of a tile from 6 offsets (as described in Section 8.5.3), will be executed from several templates.

8.4 Technical Templates

In addition to the templates described above, a set of technical templates may be used to carry out testing and engineering procedures:

- **VIRCAM_gen_tec_SelfTest** — A template which tests the operation of the instrument by exercising all the instrument modes.

- **VIRCAM_gen_tec_CheckFilters** — A template which checks the operation of the filter wheel by selecting in turn each of the filters currently installed in the instrument.

- **VIRCAM_gen_tec_LoadFilters** — A template which allows an engineer to load and/or remove one or more filters.

- **VIRCAM_howfs_tec_CalibNull** — A template which makes a series of exposures for calibrating the HOWFS null wavefront.

Other engineering templates may be defined as the need arises.

8.5 Observation Blocks

The templates defined above may be combined together to make Observation Block which carry out standard VISTA IR observations as follows:

8.5.1 Telescope HOWFS LUT calibration

1) VIRCAM_howfs_acq — Move telescope to a HOWFS star and note elevation
2) VIRCAM_howfs_obs_exp — Observe and calculate wavefront
3) Repeat the previous steps at a variety of different elevations.
4) Convert the observations into a lookup table for the TCS.

8.5.2 One-off HOWFS wavefront calibration

1) VIRCAM_howfs_acq — Set mode to HOWFS. Move telescope to a HOWFS star
2) VIRCAM_howfs_obs_wfront — Observe and calculate wavefront. Send the coefficients to the TCS.

8.5.3 Science survey tile observation

1) VIRCAM_img_acq — Set mode to IMAGING. Move telescope to the first field and acquire guide and LOWFS stars.
2) VIRCAM_img_obs_snaphot — Select the science filter and exposure time and make a "pointing" observation.
3) VIRCAM_img_acq_offset — Offset the telescope to the next "pointing" within the tile and acquire new guide and LOWFS stars.
4) VIRCAM_img_obs_snaphot — Make another "pointing" observation.
5) Repeat the last two steps another four times, making a total of 6 observations.

The VIRCAM_img_obs_snaphot template may be substituted with VIRCAM_img_obs_dither or VIRCAM_img_obs_microstep to obtain a "tile" observation whose individual observations are themselves dithered and/or microstepped.

8.6 Target Acquisition Template Parameters

The exact definition of the parameters contained in acquisition templates is to be specified by the data flow and telescope software groups. However, it is expected that an acquisition template will contain the following parameters describing the target to be observed:

8.6.1 Administrative Parameters

- Programme identification
- Name of principal investigator
- Unique science programme identifier
- Programme title string (defined by scientist)
- Observation description (optional)
- Unique observation identifier

8.6.2 Observation Scheduling Parameters

- Name of instrument to be used ("VIRCAM")
- Programme priority (assigned by committee)
- Observation priority (defined by scientist)
- Nightly priority (used to define the priority of this observation within a nightly schedule)
- Special priority (used for special purposes)
- Required limiting magnitude and signal to noise at specified colour/magnitude (for use by the ESO-VLT exposure time calculator).
- Range of acceptable phases of the moon

- Range of acceptable atmospheric seeing values in arcseconds
- Range of acceptable airmass values
- Range of acceptable atmospheric transparency values
- Required range of absolute date/times for each observation (if any)
- Required ordering and timing of observation blocks belonging to the same programme (if possible).
- Calibration requirements, such as
  - Whether this observation requires a bias frame.
  - Whether this observation requires a dark frame.
  - Whether this observation requires a flat-field frame.
  - Whether this observation requires a sky background frame.

8.6.3 Acquisition Target Parameters (for preset)

- Standard name for the target being observed
- Description of target (defined by scientist)
- Celestial coordinates of the pointing target OR name of known telescope preset corresponding to the target
- Proper motion of pointing target (if any)
- Epoch of target coordinates
- RADECSYS of target coordinates
- Equinox of target coordinates (when RADECSYS is not 'ICRS').
- Position angle of camera Y axis defined on the sky.
- Effective wavelength of the target.
- Focal plane (X,Y) coordinate (in instrument units, millimetres on the focal plane, as defined in [AD2]) on which target image should fall, defaulting to the optical axis (0,0).

8.6.4 Acquisition Target Parameters (for offset)

- Offset to be applied to the celestial coordinates OR offset to be applied to focal plane (X,Y) coordinate in millimetres.
- New position angle of camera Y axis defined on the sky (if different)
- Whether a new set of guide and LOWFS stars is required.

8.6.5 Telescope Tracking Parameters

- Whether the telescope is to be tracked or not.
- Required non-sidereal drift rate in arcseconds/minute (if any).
- Absolute Universal Time to which the non-sidereal tracking rate applies (if any)

8.6.6 Autoguiding Parameters

- Whether autoguiding is required or not.
- Name of guide star
- Celestial coordinates of star
- Magnitude of guide star

- Colour of guide star
- Proper motion of guide star (if any)
- Epoch of guide star coordinates
- Equinox of guide star coordinates

8.6.7 Active Optics Parameters

- Whether active optics correction is required or not
- For each of the two LOWFS stars:
  - Name of star
  - Celestial coordinates of star
  - Magnitude of star
  - Colour of star
  - Proper motion of star (if any)
  - Epoch of star coordinates
  - Equinox of star coordinates

8.7 Observation Template Parameters

The following parameters need to be specified within the observation templates

8.7.1 Data Handling Requirements

- Description of the type of observation to be made (e.g. bias, dark, flat-field, sky, science)
- Observation ID (used to tie together multiple files belonging to the same observation)
- Offset number (if this observation is part of tile)
- Dither number (if this observation is part of a set of dithered exposures)

8.7.2 Instrument Parameters

- Name of science filter OR name of intermediate filter/position combination

8.7.3 Detector Parameters

- Total exposure time required in seconds.
- Detector of interest (if required)
- Region of interest on that detector (if any)
- IR detector readout mode (raw data, read-reset-read, coadded or differenced).
- Number of exposures to coadd.

9 PROCEDURES

9.1 *Installing and building the software*

The ESO Configuration Management Module (CMM), [RD52], will be used for software configuration management and software installation.

The software can be built using the pkgin utility, [RD58].

The procedures for installing and building the software are described in the User and Maintenance Manual, [RD10]

9.2 *Testing the software*

The ESO-VLT Tools for Automated Testing (TAT) package, [RD51], will be used for checking the correct functioning of the software. In addition there will be a set of maintenance templates for exercising all the possible instrument observing modes (as listed in Section on page 54).

See the Acceptance Test Plan, [RD9], for a list of instrument tests.

__oOo__