-

Visible & Infrared Survey Telescope for Astronomy

IR CAMERA

Document Title: **IR Camera Maintenance and Verification Software Design Description**

Document Number: **VIS-DES-ATC-06086-0001**

Issue: **1.0**

Date: **17 August 2005**

Document Prepared By:	**S.M. Beard** **IR Camera Software**	**Signature and Date:**	
Document Approved By:	**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
0.1	10/06/04	N/A	First version following software integration meeting.
0.2	17/08/05		Brought up to date following VDFS FDR and implementation of ICS and OS.
1.0			Signed version

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
Guy Woodhouse
Gavin Dalton
ATC: Malcolm Stewart
Steven Beard
Stewart McLay

TABLE OF CONTENTS

1 INTRODUCTION

1.1 *Purpose*

This document gives a detailed description of the VISTA IR Camera Maintenance and Verification Software. It follows on from the architecture described in the *"VISTA IR Camera Functional Specification"*, [AD2], and accompanies the software design described in [RD4], [RD5] and [RD7].

1.2 *Scope*

The term "Maintenance and Verification Software" covers a wide range of software utilities in ESO-VLT parlance. This document describes only the facilities that are needed to test multiple instrument components together. It is assumed that individual instrument components are released with their own unit-testing utilities, and these utilities are described in their own software design descriptions, [RD4], [RD5], [RD6], [RD7] and [RD8].

Note that this document shows the design as it was in August 2005. The overall design is the same but some details have changed. The "VISTA IR Camera Software User and Maintenance Manual", [RD10], is more up to date than this document.

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 2.3, 8 April 2004.

[AD3] *VISTA Infrared Camera Technical Specification*, VIS-SPE-ATC-06000-0004, Issue 2.0, 20 November 2003.

1.4 *Reference Documents*

1.4.1 VISTA IR Documents

[RD1] *VISTA IR Camera Software Acronym and Abbreviation Glossary*, VIS-LST-ATC-06080-0030, Issue 1.4, 8 April 2004.

[RD2] *VISTA IR Camera Software Management Plan*, VIS-PLA-ATC-06016-0001, Issue 0.11, 29 April 2004.

[RD3] *VISTA IR Camera Software Integration Plan*, VIS-PLA-ATC-06016-0003, Draft 0.2, 3 June 2004.

[RD4] *VISTA IR Camera Observation Software Design Description*, VIS-DES-ATC-06084-0001, Issue 3.4, 17 June 2005.

[RD5] *VISTA IR Camera Instrument Control Software Design Description*, VIS-DES-ATC-06083-0001, Issue 1.10, 22 July 2005.

[RD6] *VISTA IR Camera Low Order Wavefront Sensor Software Design Description*, VIS-DES-UOD-06048-0001, Issue 1.0, 4 March 2004.

[RD7] *VISTA IR Camera High Order Wavefront Sensor Software Design Description*, VIS-DES-UOD-06048-0002, Issue 2.0, 4 March 2004.

[RD8] *VISTA IR Camera Autoguider Software Design Description*, VIS-DES-UOD-06048-0003, Issue 1.0, 4 March 2004.

[RD9] *VISTA IR Camera Software Acceptance Test Plan*, VIS-PLA-ATC-06087-0001, Issue 1.2, 15 August 2005.

[RD10] *VISTA IR Camera Software User and Maintenance Manual*, VIS-MAN-ATC-06080-0020, Issue 3.0, 18 August 2008.

1.4.2 VISTA Data Flow project documents

[RD11] *VISTA IR Camera Calibration Plan*, VIS-SPE-IOA-20000-0002, Issue 1.1, April 2005.

1.4.3 ESO-VLT Documents

[RD12] *VLT IRACE-DCS User Manual*, VLT-MAN-ESO-14100-1878, Issue 1.4, 5 December 2003.

[RD13] *VLT CCD Detectors Control Software — User Manual*, VLT-MAN-ESO-17240-0672, Issue 1.6, 25 September 1998.

[RD14] *VLT Tools for Automated Testing (TAT) User Manual*, VLT-MAN-ESO-17200-0908, Issue 1.4, 15 February 2001.

[RD15] *VLT Configuration Management Module (CMM) User Manual*, VLT-MAN-ESO-17200-0780, Issue 2.0, 22 October 2001.

[RD16] *VLT Software Installation Tool for VLT Software Packages (pkgin) — User and Maintenance Manual*, VLT-MAN-ESO-17240-1913, Issue 4, 31 March 2004.

1.5 Abbreviations and Acronyms

See the abbreviations section in the *"VISTA IR Camera Software User and Maintenance Manual"*, *[RD10]*.

1.6 Glossary

See the glossary in the *"VISTA IR Camera Software User and Maintenance Manual"*, *[RD10]*.

2 OVERVIEW

The requirements on the Maintenance and Verification Software are described in [AD1] (SWR2.8…). Individual instrument subsystems are assumed to provide their own maintenance and testing software (within the "test" subdirectory of each module, see [RD12]). These facilities are listed in Section 2.1, below. The high-level maintenance software described in this document provides additional testing and calibration facilities that need to coordinate more than one instrument subsystem. These higher level facilities are listed in Section 2.2 on page 8.

The *VISTA IR Camera Software Acceptance Test Plan*, [RD9], also gives a list of instrument tests for software acceptance testing. These are achieved through a combination of automated tests described in this document and manual checks described in [RD9]. The *VISTA IR Camera Software Integration Plan*, [RD3], shows the schedule when the subsystem unit tests are expected to take place.

2.1 Requirements Covered by Instrument Subsystem Software

The individual instrument subsystems already provide the following test facilities of their own, as described below.

2.1.1 Observation Software

The Observation Software, [RD4], has engineering and self-testing facilities that:

- self-test the system by checking that all known commands work as expected ("vcoTest" command);
- write an observation log;
- display more detailed instrument information on an engineering GUI;
- have the ability to run processes in debugging/verification mode with extra output.

2.1.2 Instrument Control System

The Instrument Control Software, [RD5], has engineering and self-testing facilities that:

- self-test the subsystem by checking that all known commands work as expected ("vciTest" command);

- check that all devices present on the LCU bus have connected (automatic during ONLINE command);
- check that all motors and microswitches are working (though for safety reasons no mechanisms should move unless specifically commanded to do so);
- verify correct functionality with the primary and backup reference switches (manual procedure);
- allow the positions of named filters to be defined through a configuration file;
- can put a filter in the beam or move it to the load position;
- can move the filter wheel to any position defined in motor steps;
- can measure the accuracy and repeatability of motor movement by counting motor steps between activations of a microswitch;
- check that all sensors are functioning correctly (STATUS command);
- check that temperature control is functioning correctly;
- log instrument faults and alarms;
- display detailed instrument configuration information on an engineering GUI;
- have the ability to run processes in debugging/verification mode with extra output.

2.1.3 Detector Control System

The VISTA IRACE Detector Control Software, [RD12], has engineering and self-test facilities that:

- self-test the subsystem by checking that all known commands work as expected ("vcdTest" command);
- check that all detector hardware is functioning correctly (automatic during ONLINE command);
- provide stand-alone engineering utilities to measure the quality, gain, saturation and QE of each science detector (already used to characterise the science detectors);
- can make an exposure on demand and read out the result to a file;
- can read out a window on the science detectors;
- log detector faults and alarms;
- display detailed detector configuration information on an engineering GUI;
- have the ability to run processes in debugging/verification mode with extra output.

2.1.4 Low Order Wavefront Sensor and Autoguider

It is assumed Low Order Wavefront Sensor software, [RD6], Autoguider software, [RD8], and the underlying Technical CCD controller software, [RD13], have engineering and self-test facilities that:

- self-test the subsystems by checking that all known commands work as expected;
- can make an exposure on demand and read out the result to a file (TCCD utilities);
- provide a utility or procedure to check for electrical interference between the LOWFS and AG detector readouts (already used for AIT1);
- verify using test data that the image analysis and centroiding algorithms are functioning correctly;
- log TCCD faults and alarms;

- display detailed configuration information on an engineering GUI;
- have the ability to run processes in debugging/verification mode with extra output.

2.1.5 High Order Wavefront Sensor

It is assumed High Order Wavefront Sensor Software, [RD7], has engineering and self-test facilities that:

- self-test the subsystem by checking that all known commands work as expected ("vchTest" command?);
- verify using test data that the image analysis algorithm is functioning correctly;
- log faults and alarms (if any);
- display detailed information (if any) on an engineering GUI;
- have the ability to run processes in debugging/verification mode with extra output.

2.2 High Level Maintenance Software Requirements

The following requirements involve the coordination of more than one instrument subsystem, and need to be fulfilled by the high-level maintenance software.

2.2.1 General utilities

a) provide automated test procedures for verifying the health and functionality of the entire instrument.
b) provide a utility to test the performance of the system by measuring the achievable data rate;
c) provide a utility an engineer can use to load or unload filters.

2.2.2 Science detector utilities

d) provide a utility to generate data for the measurement of the detector cross talk;
e) provide a utility to generate data for the measurement of the detector dark current and linearity;
f) provide a utility to check for light leaks, ghost images and detector remnants.

2.2.3 World Coordinates

g) provide a utility to generate data for the calibration of the spatial location of the science detectors;

2.2.4 Filter wheel/Science detector utilities

h) provide a utility to calibrate the focus offset of each science filter;
i) provide a utility to test the repeatability of the filter wheel in positioning a known object over the science detectors.

2.2.5 Wavefront Sensor/Science detector utilities

j) ~~provide a utility to generate data for the calibration of the HOWFS null wavefront.~~

(No longer needed, since this information has been derived from the VISTA IR camera optical model.)

3 DETAILED DESCRIPTION

This section contains a detailed description of the utilities fulfilling the previously-listed requirements. It is assumed that:

- Test facilities provided by the instrument subsystems are available and can be executed automatically.
- Standard data reduction facilities are available (e.g. through MIDAS) and the VISTA IR Camera software work package *does not* need to provide any custom data reduction software.

Maintenance procedures fall into three main categories: calibration templates or technical templates executed by BOB, or independent test scripts executed directly from the command line. The calibration templates are used to execute standard calibration procedures from the GUI, and can also be scheduled as part of an observing session. Technical templates are used to execute high level engineering procedures. They can be executed from the GUI but are never scheduled as part of an observing session. The test scripts tend to be used for verifying the functionality of the instrument, especially after new software releases.

3.1 Calibration Templates

VIRCAM has the following calibration templates which cover some of the maintenance and verification requirements. These templates are already implemented by the Observation Software, [RD4]. See their full description in the VISTA IR Camera Calibration Plan, [RD11] and in CMM modules vcotsf and vcoseq.

- **VIRCAM_howfs_cal_reset.tsf** — Obtain a detector reset frame to calibrate HOWFS observations.

- **VIRCAM_howfs_cal_dark.tsf** — Obtain a detector dark frame to calibrate HOWFS observations.

- **VIRCAM_howfs_cal_domeflat.tsf** — Obtain a dome flat-field frame to calibrate HOWFS observations.

- **VIRCAM_img_cal_reset.tsf** — Obtain a detector reset frame to calibrate science observations.

- **VIRCAM_img_cal_dark.tsf** — Obtain a detector dark frame to calibrate science observations. This frame can also be used to check for bad pixels.

- **VIRCAM_img_cal_domeflat.tsf** — Obtain a dome flat-field frame to calibrate science observations.

- **VIRCAM_img_cal_twiflat.tsf** — Obtain a twilight sky flat-field frame to calibrate science observations.

- **VIRCAM_img_cal_std.tsf** — Observe a field of photometric standard stars to calibrate science observations.

- **VIRCAM_img_cal_crosstalk.tsf** — Calibrate the detector cross talk by making a series of exposures of a bright star with a series of telescope offsets.

- **VIRCAM_img_cal_darkcurrent.tsf** — Calibrate the detector dark current by making a series of dark exposures using a specified sequence of exposure times.

- **VIRCAM_img_cal_illumination.tsf** — Calibrate the illumination function across the camera field of view by making a series of exposures of a bright star with a series of telescope offsets. This template will detect any internal scattering within the instrument, and can also be used to generate data for verifying the World Coordinates.

- **VIRCAM_img_cal_linearity.tsf** — Calibrate the detector linearity by making a series of dome-flat exposures using a specified sequence of exposure times.

- **VIRCAM_img_cal_noisegain.tsf** — Calibrate the readout noise and gain of the detectors by making a pair of dark and flat-field exposures at a series of exposure times.

- **VIRCAM_img_cal_persistence.tsf** — Calibrate any detector persistence by making an exposure of a bright star and following this with a series of dark.

3.2 Technical Templates

All these technical templates (except the first, which is provided by the OS) need to be implemented as part of the maintenance software (in CMM modules vcmtsf and vcmseq):

- **VIRCAM_gen_tec_SelfTest.tsf** — A template that tests the operation of the instrument by exercising all the instrument modes. Every science and calibration template is executed in turn.
  Parameters: None. The test parameters for each individual template are preset in advance (using .obd files).

- **VIRCAM_gen_tec_exp.tsf** — A template that tests the system data rate performance by making and timing a rapid series of exposures.
  Parameters: Exposure time, number of repeats.

- **VIRCAM_gen_tec_CheckFilters.tsf** — A template that checks the operation of the filter wheel by selecting each filter from a list of filters in turn. The time taken to move the filter wheel is recorded.
  Parameters: List of filters, number of repeats, randomise movements flag, wait time between filter wheel movements.

- **VIRCAM_gen_tec_CalibFilter.tsf** — A template that checks the accuracy of the filter wheel by making a series of repeated movements. A specified filter is placed in the beam, an exposure made, the filter moved away and back again, and another exposure made. The difference between the exposures can be used to verify the repeatability of the filter wheel. The test can be repeated with a variety of different filters. The most useful filters to try would be a pinhole or the HOWFS beam-splitter.
  Parameters: Filter, direction and distance to move the filter wheel, detector window, exposure time, number of repeat measurements
  Data reduction: This template needs a data reduction utility that can divide or compare the pairs of measurements and show any mismatch between them.

- **VIRCAM_gen_tec_LoadFilters.tsf** — A template that allows an engineer to load and/or remove one or more filters in sequence. The template prompts the engineer each time a filter has been moved to the load position for replacement. The utility generates a new filter wheel configuration file.
  Parameters: List of filters to be changed.

- **VIRCAM_img_tec_FocusFilters.tsf** — A template that calibrates the focus offset for a set of science filters. For each filter, the template makes a series of exposures with the telescope focus offset by different amounts.
  Parameters: List of filters, starting focus, focus increment, number of focus increments, detector window, exposure time.
  Data reduction: This template needs a data reduction utility that analyses a series of exposures at different focus settings and returns the setting that minimises the image size.

3.3 Test scripts

All test scripts will be included in the “test” subdirectory of the software package to which they refer. For example:

- vcd/test — VIRCAM detector control software test scripts.
- vci/test — VIRCAM instrument control software test scripts. These are described in more detail in **Error! Reference source not found.**.
- vco/test — VIRCAM observation software test scripts.

Scripts that test the entire instrument are included in the instrument integration module test directory, “vcins/test”.

3.3.1 Instrument self-test

A complete test of the vcins software package can be executed (once built) by running the facility

```
pkginTest vcins
```

which executes each of the self-test facilities provided with the instrument subsystems. This facility can be used, for example, to check the instrument software has survived an upgrade to the ESO-VLT software. The ESO-VLT Tools for Automated Testing (TAT) package, [RD12], are normally used to run these tests.

4 PHYSICAL DEPLOYMENT

VISTA IR Camera maintenance and verification utilities run on the Instrument Workstation.

5 DEVELOPMENT PLAN

The maintenance and verification software will need to be constructed from utilities developed at the UKATC and at the University of Durham. Some utilities may also be contributed by engineers at RAL.

- The UKATC is expected to provide the utilities testing the entire instrument or testing the interaction between the filter wheel and science detectors.
- The University of Durham is expected to provide any procedures or utilities for detecting any electromagnetic interference between the wavefront sensor detectors and science detectors.
- ESO is expected to provide utilities for testing the IRACE detector controller software.

Test utilities should be regarded as part of the deliverable software, rather than to be thrown away. Test scripts and source code specifically designed for testing should be separated from the operational software by storing them within the “test” subdirectory.

6 PROCEDURES

6.1 Installing and building the software

The ESO Configuration Management Module (CMM), [RD15], will be used for software configuration management and software installation.

The software can be built using the pkgin utility, [RD16]. When the VISTA software is operational, the test software will not be built by default. An additional configuration file will be provided that will build the test software on demand. So

```
pkginBuild vcins –env wvcam lvcics1
```

would build the operational software and

```
pkginBuild vcins –cfg vcinsINSTALL_TEST.cfg –env wvcam lvcics1
```

would in addition build the test utilities. The procedures for installing and building the software will be described in full in the *VISTA IR Camera Software User and Maintenance Manual*, [RD10].

Instrument control test utilities are described in detail in [RD10].

__oOo__