ESO data format and compatibility

Valentin D. Ivanov (ESO)

EUROPEAN SOUTHERN OBSERVATORY

Organisation Européenne pour des Recherches Astronomiques dans l'Hémisphère Austral

Europäische Organisation für astronomische Forschung in der südlichen Hemisphäre

VERY LARGE TELESCOPE

Data Flow for VLT/VLTI Instruments

Deliverables Specification

VLT–SPE–ESO–19000–1618

Issue 3

Date 2011–02–01

... This document defines the interface between a VLT Instrument Consortium (VIC) and Data Flow System in

terms of deliverables and dependencies.

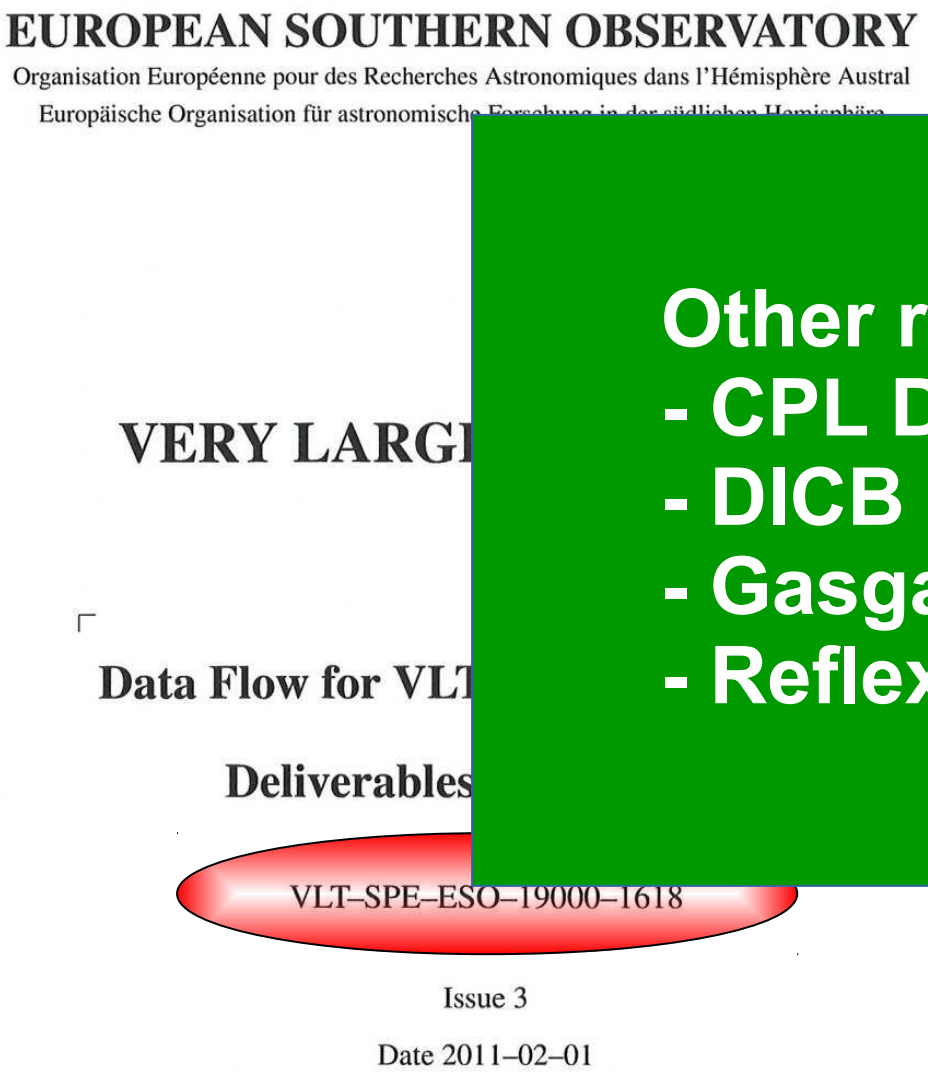
It lists a complete set of Data Flow System related tasks which should be completed by a VIC and specifies the products to be delivered...

Dependencies on products provided by ESO are also listed to make it easier for a VIC to perform its planning…

Version 4 coming soon!

ESO data format and compatibility

Valentin D. Ivanov (ESO)


EUROPEAN SOUTHERN OBSERVATORY
Organisation Européenne pour des Recherches Astronomiques dans l'Hémisphère Austral
Europäische Organisation für astronomische Forschung in der südlichen Hemisphäre

VERY LARGE

Data Flow for VLT

Deliverables

VLT–SPE–ESO–19000–1618

Issue 3

Date 2011–02–01

... This document defines the
interface between a VLT Instrument
ata Flow

d

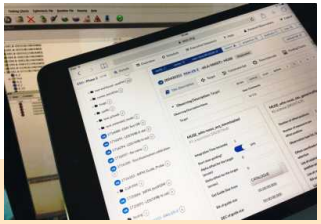
Data Flow
ich should be
specifies the
...

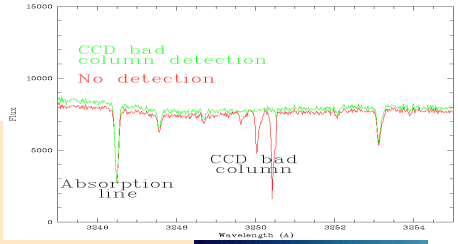
cts provided
o make it
easier for a VIC to perform its
planning…

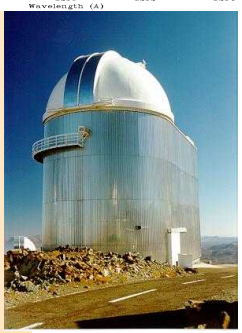
Version 4 coming soon!

Other relevant docs:
- **CPL Documentation**
- **DICB specs**
- **Gasgano manual**
- **Reflex development**

Deliverables and integration







OB

Observation Handling

Phase II Proposals
ST/MT Scheduler
OB Repository

Long-term Schedule
Observing Programmes

OB

Observation Blocks

Ambient Data
Configuration
Status

VLT Control System

OBSERVATION

OB

Programme Handling

Phase I Proposals
LT Scheduler

OB

Quality status

OB

Data Frames
ObsBlocks

Science Archive

Catalogue Server
On-line Archive
Storage & Data Distribution

Technical
Programmes

Quality Control

Calibration Plan
Simulators
Quality Monitor
Quality Scheduler

Reduced and Calibrated Frames

Reduced Frames

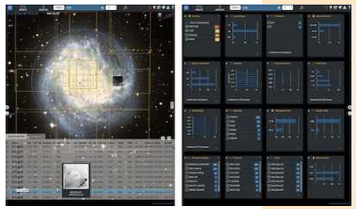
Recipes

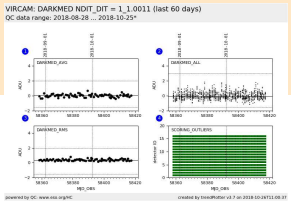
Pipeline

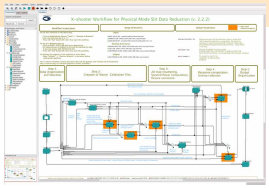
Data Organizer
Reduction Scheduler
Reduction Block

Data Frames
Calibration Data

OB







Moto and outline

The pipeline serves the community!

- Deliverables
- Raw data and Science-grade data products
- Project organization and integration process

Deliverables: calibration plan

The Calibration Plan describes:

(1) the **scientific data calibrations** and

(2) the **instrument health monitoring.**

For every calibration it should contain:

- Responsible group (e.g. Science Operations or User), Phase (day or nighttime), how often the calibration task has to be carried out, Purpose, Procedure (how), OBs and Templates to be used, Duration, Prerequisites

- Outputs, including data products, Quality Control (QC) parameters and/or keywords entered into various databases and archives

All this – in a Calibration Plan document (MS Word or Latex).

Deliverables: instrument package

Function: to definition of the Observation Blocks (OB = sequence of templates that operate the instrument). The templates are used by the instrument software (written in the VLT Sequencer script language). It is not a part of the DFS.

Components:

- ***Template Signature Files (TSF)*** - one for each template, defined the list of template parameters.
- ***Instrument Summary File (ISF)*** - contains the allowed values or ranges of each parameter in the TSFs.
- ***External Verification Modules (EVM)*** – OB validity checks.
- ***External Time Report Modules (ETRM)*** - algorithms for the computation of the exposure and execution times

Deliverables: instrument package

ISF

```
#-- Red Arm ------------------------------------------------------------
#
#SLIT3.WID.RANGE     "0.15..10.0"
SLIT3.WID.RANGE     "0.15 0.2 0.3 0.4 0.44 0.5 0.6 0.68 0.7 0.8 0.9
                     1.0 1.1 1.2 1.4 1.6 1.8 2.0 2.2 2.4 2.6 2.8
                     3.0 3.2 3.4 3.6 3.8 4.0 4.2 4.4 4.6 4.8 5.0 10.0"
SLIT3.WID.DEFAULT   "0.6"
SLIT3.WIDSTEP.RANGE "0.1..10.0"
#
SLIT3.LEN.RANGE     "0.2..30.0"
SLIT3.LEN.DEFAULT   "10.0"
#
FILT3.NAME.RANGE    "FREE
                     BG40 SHP700 OG590 BK7_5 BK7_
                     HALPHA HBETA OIII5007 OIII43
                     OI6300 SII6724 HeII4686"
FILT3.NAME.DEFAULT  "FREE"
FILT3.FREE          "FREE"
#
GRAT2.NAME.RANGE    "CD#3 CD#4"
GRAT2.NAME.DEFAULT  "CD#3"
#
GRAT2.WLEN.RANGE        "500.0..1100.0"
GRAT2.WLEN.DEFAULT      "520.0"
#
TILT2.POS.RANGE     "-220.0..220.0"
TILT2.POS.DEFAULT   "0.0"
#
DET6.UIT.RANGE      "1..10"
DET6.UIT.DEFAULT    "1"
DET6.UITTOT.RANGE   "10..300"
DET6.UITTOT.DEFAULT "60"
```

TSF

```
# ------------------------------------------------------------------
TPL.PARAM               "INS.OPTI3.NAME";          # Next template parameter
INS.OPTI3.NAME.TYPE     "keyword";                 # Keyword type
INS.OPTI3.NAME.RANGE    "ISF UVB.SLIT.NAMES_IP87"; # Valid range
INS.OPTI3.NAME.DEFAULT  "5.0x11";                  # Default value
INS.OPTI3.NAME.LABEL    "UVB Slit";                # Label used in P2PP
INS.OPTI3.NAME.MINIHELP "UVB Slit";                # Short Help for keyword

# ------------------------------------------------------------------
TPL.PARAM               "INS.OPTI4.NAME";          # Next template parameter
INS.OPTI4.NAME.TYPE     "keyword";                 # Keyword type
INS.OPTI4.NAME.RANGE    "ISF VIS.SLIT.NAMES_IP87"; # Valid range
INS.OPTI4.NAME.DEFAULT  "5.0x11";                  # Default value
INS.OPTI4.NAME.LABEL    "VIS Slit";                # Label used in P2PP
INS.OPTI4.NAME.MINIHELP "VIS Slit";                # Short Help for keyword

# ------------------------------------------------------------------
TPL.PARAM               "INS.OPTI5.NAME";          # Next template parameter
INS.OPTI5.NAME.TYPE     "keyword";                 # Keyword type
INS.OPTI5.NAME.RANGE    "ISF NIR.SLIT.NAMES_IP87"; # Valid range
INS.OPTI5.NAME.DEFAULT  "5.0x11";                  # Default value
INS.OPTI5.NAME.LABEL    "NIR Slit";                # Label used in P2PP
INS.OPTI5.NAME.MINIHELP "NIR Slit";                # Short Help for keyword

# ------------------------------------------------------------------
TPL.PARAM               "DET1.WIN1.UIT1";          # Next template parameter
DET1.WIN1.UIT1.TYPE     "number";                  # Keyword type
DET1.WIN1.UIT1.RANGE    "ISF DET.UIT.RANGE";       # Valid range
DET1.WIN1.UIT1.DEFAULT  "";                        # Default value
DET1.WIN1.UIT1.LABEL    "UVB Exposure time";       # Label used in P2PP
DET1.WIN1.UIT1.MINIHELP "UVB integration time in seconds"; # Short Help for keyword
```

Deliverables: instrument package

ETRM

```
#*********************************************************************
# E.S.O. - VLT project
#
# "@(#) $Id: exec_XSHOOTER.tcl 303302 2017-11-09 14:23:45Z smarteau $"
#
# Makefile of ohsvlt
#
# who       when        what
# --------  ----------  ---------------------------------------------
# rschmutz  2016-10-07  this header added, dbRead filter address updated
#

# File: execXSHOOTER.tcl
# Purpose: computation of execution/exposure time for XSHOOTE
# Author: Stephane Marteau

# $Id: exec_XSHOOTER.tcl 303302 2017-11-09 14:23:45Z smarteau
# This program contains an implementation of the execution ti
# calculation for the VIS/UVB arms of X-Shooter. The rational
# is that:
#
# 1) there is only one CCD controller that reads both detecto
#
# 2) the detectors can only be read sequentially;
#
# 3) if the exposure of CCD1 ends while the controller is rea
#    the read-out of CCD1 takes place only once the read-out
#    finished.
```

```
# Procedure: getInstrumentSetupOverhe
# arguments:
#  - ob_id: OB ID
#  - tpl_idx: template ID in the OB
#
# returns: overhead due to change of
#
proc getInstrumentSetupOverhead { ob

    global execCommon OBdata

    set overhead 0

    set tplMode1 [evmSystemParameterValue "lastTemplateMode"]
    set tplMode2 ""

    regexp {XSHOOTER(_slt_|_ifu_|_img_)} $OBdata($ob_id,$tpl_idx,tplName) dummy tplMode2

    if { $tpl_idx != 1 && $tplMode1 == $tplMode2 } {

        set overhead 0

        if { $tplMode2 == "_slt_" } {

            set lastUVBSlit [evmSystemParameterValue "lastUVBSlit"]
            set lastVISSlit [evmSystemParameterValue "lastVISSlit"]
            set lastNIRSlit [evmSystemParameterValue "lastNIRSlit"]

            if { $OBdata($ob_id,$tpl_idx,INS.OPTI3.NAME) != "not-found" && \
                    $OBdata($ob_id,$tpl_idx,INS.OPTI3.NAME) != $lastUVBSlit } {
                set overhead $execCommon(SLT_SETUP)
```

```
# Readout times for NIR arm
set readout(NIR) 1.46

# Readout time for AGCCD (imaging mode)
set readout(AGCCD) 1.1

# Telescope preset / AG overhead
set execCommon(TEL_PRESET) 360
set execCommon(AG_OVERHEAD) 1.1

# Offset overheads
set execCommon(OFFSET_OVERHEAD)         15
set execCommon(BLIND_OFFSET_OVERHEAD)   30
set execCommon(NOD_OVERHEAD)             5
set execCommon(NIR_EXTRA_OVERHEAD)       1.34

# Instrument setup
set execCommon(SLT_SETUP)   30
set execCommon(IFU_SETUP)   60
set execCommon(IMG_SETUP)   20
set execCommon(IMG_SLT_SETUP)   20
set execCommon(IMG_IFU_SETUP)   50
set execCommon(NIR_CAL_SETUP)   6

# Filter change
set execCommon(FILTER_CHANGE)   6

# Hardcoded first exposure delay
set execCommon(DelayTime_UVB)   0.0
set execCommon(DelayTime_VIS)   5.0
set execCommon(DelayTime_NIR)   10.0
```

Deliverables: instrument package

EVM

```
# Analyze selection of OBs + containers
# and parse all relevant OBs
retrieveAndParse

# Exception: if there are no OBs in the parsed mater
if { $OBdata(nbOBs) == 0 } { return }

# Collect Phase 1 info
collectPhase1Info

set OBdata(totalExecutionTime) 0


# Loop over all elements in the compound list
foreach object $compoundList {

    # Level 1 checks
    checkXSHOOTERLevel1 $object

    # Level 2 checks
    checkXSHOOTERLevel2 $object

    # Level 3 checks
    checkXSHOOTERLevel3 $object

    reportSHOOTER_ExposureTimes $object

}

# Check on the total allocated time
```

```
# XSHOOTER Level 2
#
# Check various OB Structures.
#
# REMARK: following tests are handled by P2PP:
#
#  - all OBs must start with acquisition template
#  - all OBs must have at least one template
#
proc checkXSHOOTERLevel2 { object } {

    global OBdata evmXSHOOTERTpl isArcsAttTemplateUsed

    # Get 'object type' (first element of object list)
    set objType [lindex $object 0]

    set listOfObsBlocks {}
    set isContainer false

    # If the object type is a list, then we are dealing
    # with a container: the actual type has to be fetched
    # deeper inside the object list
    if { [llength $objType] > 1 } {

        set containerInfo [lindex $object 0]

        set objType [lindex $containerInfo 0] ; # Type of container
        set containerID [lindex $containerInfo 1] ; # Container ID
        set objRef  [lindex $containerInfo 2] ; # Object reference (Java handle)
        set objSelected [lindex $containerInfo 3] ; # Is container selected or not
        set containerName [evmName $objRef]
```

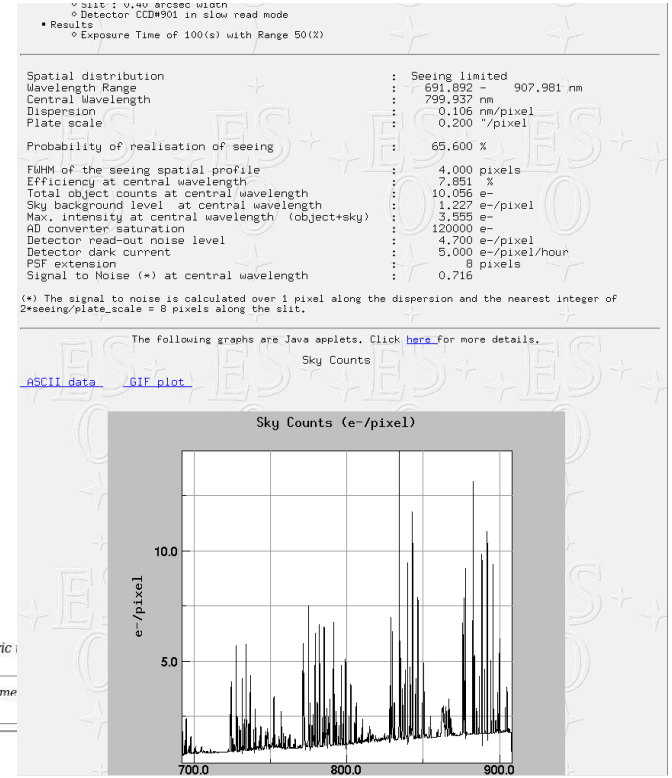
Deliverables: ETC

Detected Counts

Order	FSR Min Wavelength					Wavelength of central column						FSR Max Wavelength				
	Eff. (%)	Obj (e-)	Sky (e-)	Imax (e-)	S/N*	lambda (nm)	bin size (nm)	Eff. (%)	Obj (e-)	Sky (e-)	Imax (e-)	S/N*	Eff. (%)	Obj (e-)	Sky (e-)	Imax (e-)	S/N*
99	2.5	161	30.7	45	8.4	616.85	0.0025	5.1	273	53.2	76	12	2.6	122	23.4	34	6.8
100	2.5	160	30.3	44	8.4	610.69	0.0025	5.1	274	52.5	76	12	2.5	121	23.1	34	6.8
101	2.5	159	29.8	44	8.4	604.64	0.0024	5	276	51.7	76	12	2.5	121	22.8	34	6.8
102	2.5	157	29.3	44	8.3	598.72	0.0024	4.9	274	50.9	76	12	2.5	121	22.4	33	6.8
103	2.4	153	28.7	42	8.1	592.90	0.0024	4.9	272	49.7	75	12	2.4	119	21.9	33	6.7
104	2.4	158	28	44	8.3	587.21	0.0024	4.8	274	48.6	76	12	2.4	115	21.5	32	6.6
105	2.4	154	27.7	43	8.2	581.61	0.0024	4.7	268	47.7	74	12	2.4	120	21.1	33	6.8
106	2.3	154	26.9	43	8.2	576.13	0.0023	4.6	266	46	74	12	2.3	116	20.2	32	6.6
107	2.2	150	25.3	41	8	570.75	0.0023	4.4	258	44	71	12	2.2	115	19.4	32	6.6
108	2.2	149	24.7	41	8	565.46	0.0023	4.4	257	43	71	12	2.2	115	19.1	32	6.6
109	2.2	149	24.1	41	8	560.28	0.0023	4.3	257	42.2	71	12	2.2	114	18.8	31	6.5
110	2.1	148	23.7	41	8	555.18	0.0023	4.3	259	41.4	71	12	2.1	115	18.4	32	6.6
111	2.1	147	23.1	40	7.9	550.18	0.0022	4.2	256	40.5	70	12	2.1	114	18.1	31	6.5
112	2.1	145	22.8	40	7.9	545.27	0.0022	4.2	256	39.9	70	12	2.1	114	17.9	31	6.5
113	2.1	145	22.4	40	7.9	540.45	0.0022	4.1	251	39.2	69	12	2.1	113	17.6	31	6.5
114	2	143	22	39	7.8	535.71	0.0022	4.1	258	38.6	71	12	2	113	17.3	31	6.5
115	2	141	21.5	39	7.7	531.05	0.0022	4	250	37.7	68	12	2	111	16.9	31	6.4
116	2	143	21.1	39	7.8	526.48	0.0021	4	244	36.9	67	11	2	110	16.6	30	6.4

*er spectral bin. For point sources, **Eff** refers to the total efficiency including the slit loss and atmospheric*

*a one-hour execution time limit for Service Mode OBs, and that the times returned here **do not** include instrume*
en constructing compliant OBs.



Detector: EEV

Spectral Format Red, CD#3

Order	wav of central column (nm)	y of central column (pix)	y of central column (arcsec)	FSR range (nm)	FSR l Min (nm)	FSR l Max (nm)	start wav (nm)	end wav (nm)	TS range (nm)
118	517.56	2047	373	4.38	514.84	519.22	512.75	521.37	8.61
119	513.21	1958	356	4.31	510.53	514.84	508.44	516.99	8.55
120	508.93	1871	341	4.24	506.30	510.53	504.20	512.68	8.48
121	504.73	1786	325	4.17	502.13	506.30	500.03	508.45	8.41
122	500.59	1701	310	4.10	498.03	502.13	495.93	504.28	8.35
123	496.52	1619	295	4.03	494.00	498.03	491.90	500.19	8.29
124	492.52	1537	280	3.97	490.03	494.00	487.93	496.16	8.22
125	488.58	1457	265	3.90	486.13	490.03	484.03	492.19	8.16
126	484.70	1379	251	3.84	482.28	486.13	480.18	488.29	8.10
127	480.89	1301	237	3.78	478.50	482.28	476.40	484.44	8.04
128	477.13	1225	223	3.72	474.78	478.50	472.68	480.66	7.99
129	473.43	1150	209	3.67	471.11	474.78	469.01	476.94	7.93
130	469.79	1077	196	3.61	467.50	471.11	465.40	473.27	7.87
131	466.21	1004	183	3.56	463.95	467.50	461.84	469.66	7.82
132	462.68	933	170	3.50	460.45	463.95	458.34	466.11	7.76
133	459.20	863	157	3.45	457.00	460.45	454.89	462.61	7.71
134	455.77	794	144	3.40	453.60	457.00	451.50	459.16	7.66
135	452.40	726	132	3.35	450.25	453.60	448.15	455.76	7.61
136	449.07	659	120	3.30	446.95	450.25	444.85	452.41	7.56
137	445.80	593	108	3.25	443.70	446.95	441.60	449.11	7.51
138	442.57	528	96	3.20	440.50	443.70	438.40	445.86	7.46
139	439.38	465	85	3.16	437.34	440.50	435.24	442.65	7.41
140	436.25	402	73	3.11	434.23	437.34	432.13	439.49	7.36
141	433.15	340	62	3.07	431.16	434.23	429.07	436.38	7.31
142	430.10	279	51	3.03	428.13	431.16	426.04	433.31	7.27
143	427.10	219	40	2.98	425.15	428.13	423.06	430.28	7.22
144	424.13	160	29	2.94	422.21	425.15	420.12	427.30	7.18
145	421.21	102	19	2.90	419.31	422.21	417.22	424.35	7.13
146	418.32	45	8	2.86	416.44	419.31	414.36	421.45	7.09

Templates exist!

Deliverables: pipeline

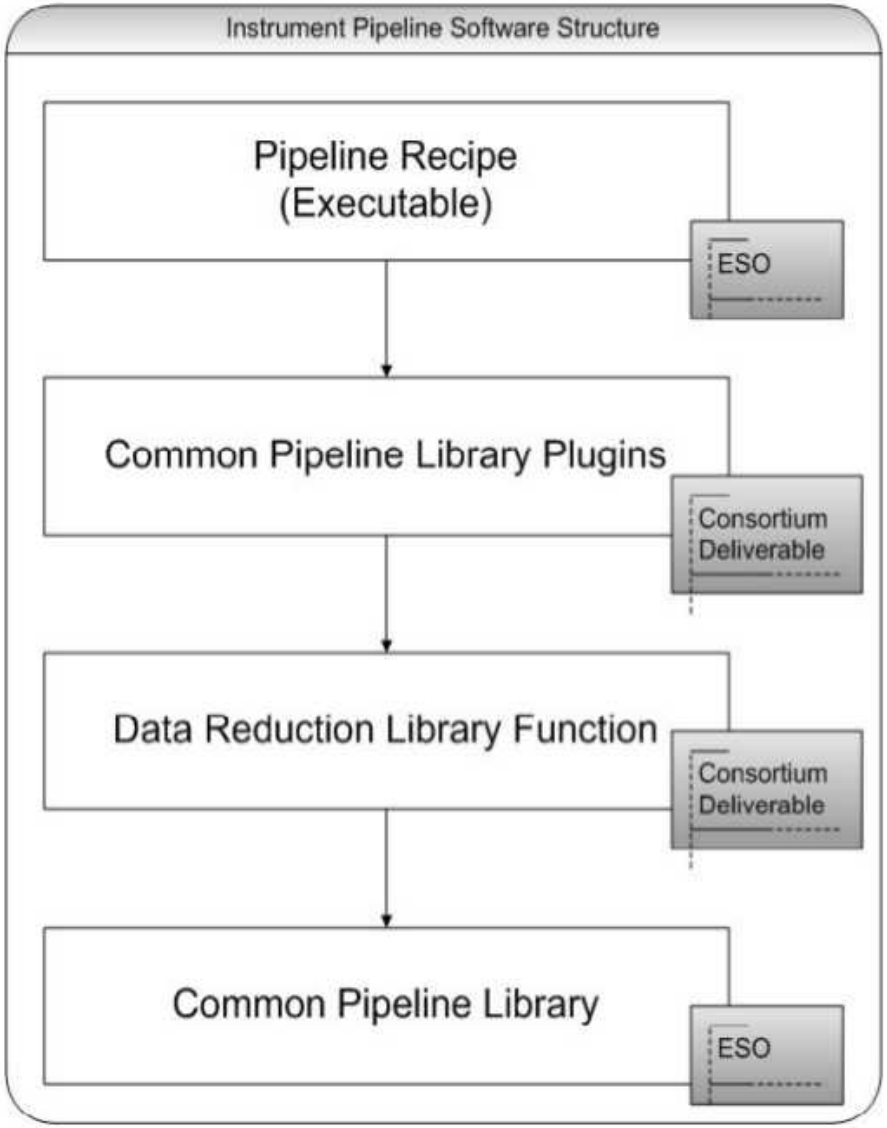
Within the end-to-end operations concept, the main pipeline missions are:

- to process raw calibration frames into **master calibration products**
- use these master calibrations **to produce QC parameters for monitoring telescope, instrument and detector performance**
- to process raw science frames, using master calibration products, into **science grade data products**. Those data products consist of data with the instrumental signature removed and whenever possible calibrated into **physical units** (e.g. flux, wavelength) with associated error bars.

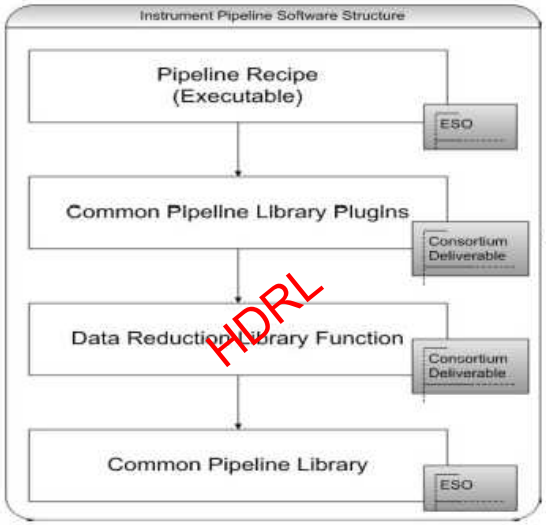
ESO lingua

CPL: The Common Pipeline Library is a set of C libraries, developed to standardize the instrument pipelines, to shorten their development cycle and to ease their maintenance. All ESO pipelines must be written in CPL.

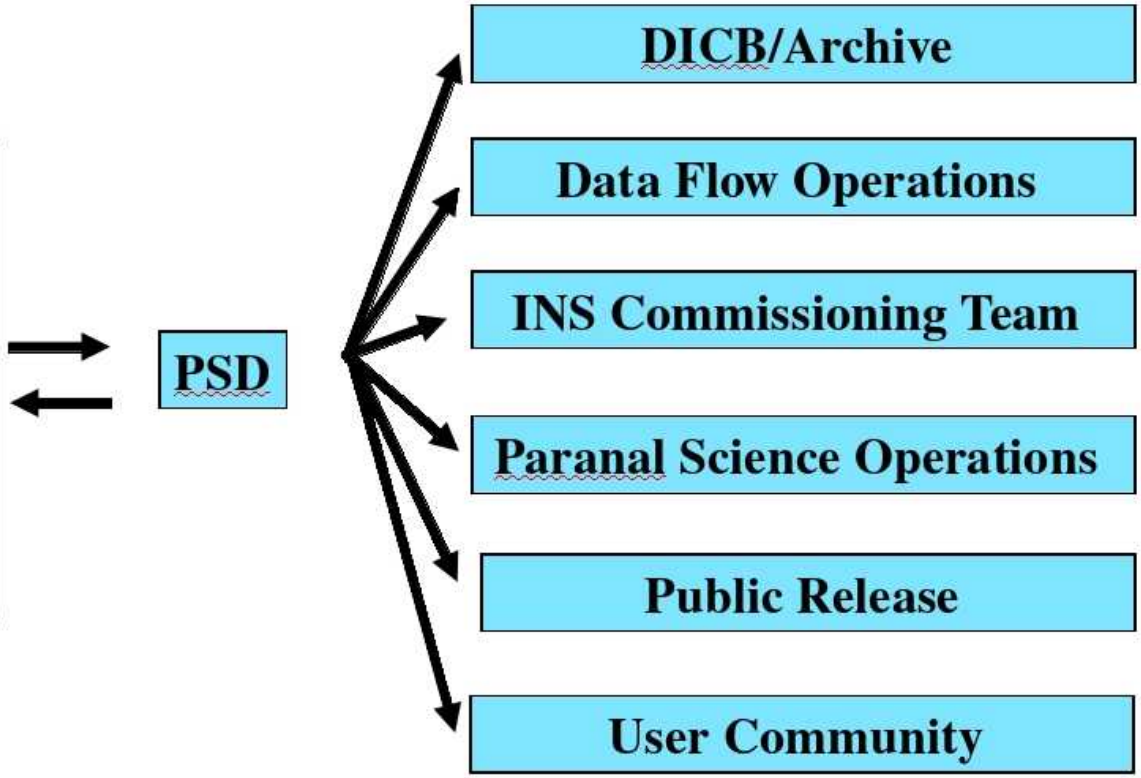
CPL Plugins: interface between the DFS pipeline (in CPL) and the Pipeline Recipes (executable programs performing a sequence of individual data processing steps).



Deliverables: pipeline



+ Reflex Workflow



Observatory Pipeline
Quality Control Pipeline
Science-Grade Desktop Environment

PSD – Pipeline Systems Department
DICB – Data Interface Control Board
HDRL – High-Level Data Reduction Library

OCA Rules

OCA = organization, classification and association:

1. Classification:

```
if DPR.CATG=="CALIB" and DPR.TYPE=="LAMP,WAVE"  then
{
  DO.CLASS = "ARC_SPECTRUM";
  RAW.TYPE = "WAVE";
}
```

2. Organization:

```
select execute(GI_WAVE_CALIBRATION) from inputFiles
 where RAW.TYPE=="WAVE“ group by TPL.START
```

3. Association:

```
action GI_WAVE_CALIBRATION
{
select file as MASTER_BIAS from calibFiles where
PRO.CATG=="MASTER_BIAS" and inputFile.DET.WIN1.BINX==DET.WIN1.BINX;

select file as GRATING_DATA from calibFiles where
PRO.CATG=="GRATING_DATA" and inputFile.INS.GRAT.NAME==INS.GRAT.NAME;

recipe giwavecalibration { }

product mflat { PRO.CATG="MASTER_FLAT"; }
}
```

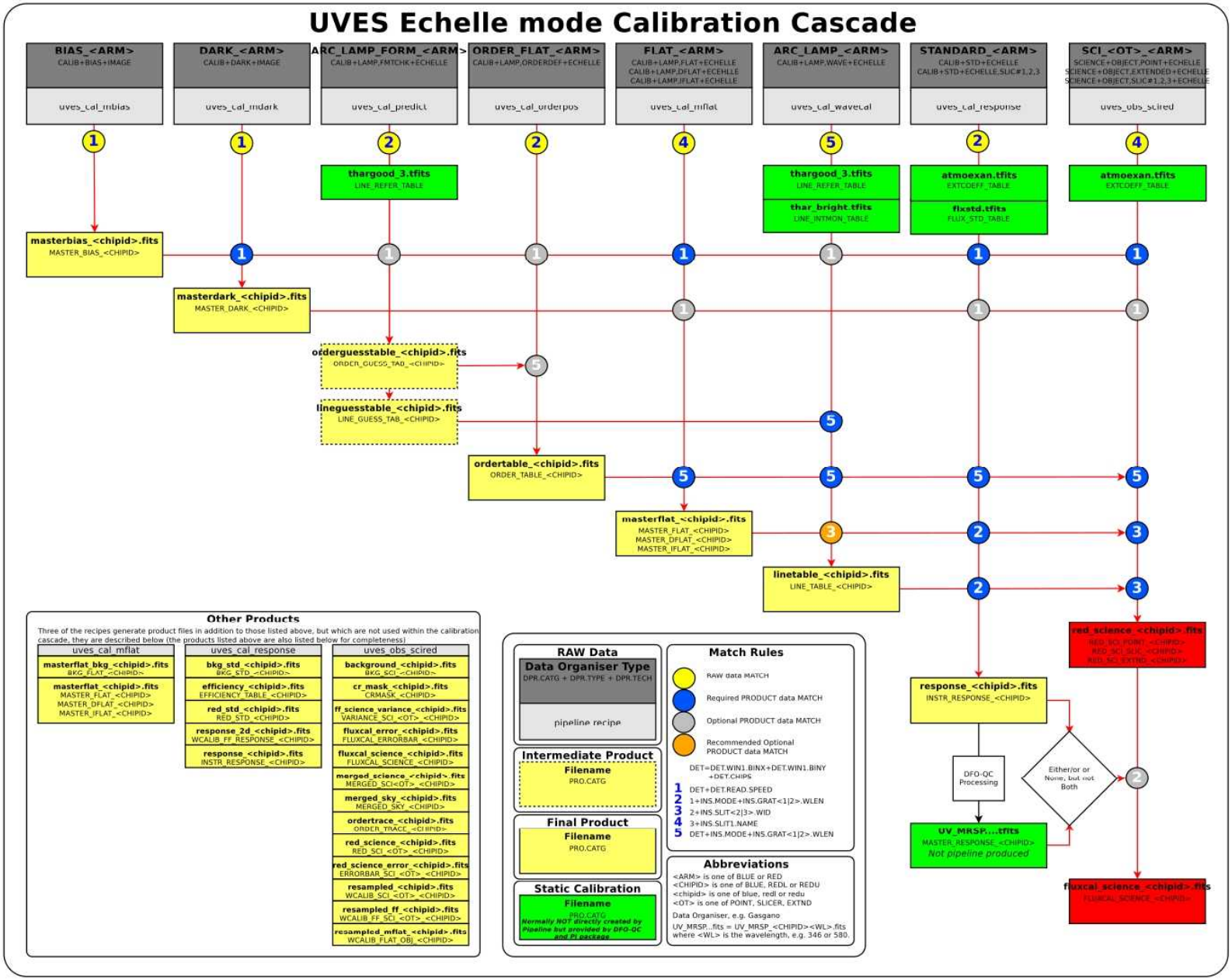
Deliverables: software and datasets

====> Data Reduction Library: ANSI/ISO-C99 implementation of the fnctions described in the documents above.

ETC Instrument Description Database: ascii files for each detector or optical component defining its properties and ETC calculation modules as defined by the ETC Specifications document.

Deliverables: pipeline

Calibration cascade, modular design (uses HDRL funcitons):



Deliverables: Documentation

EUROPEAN SOUTHERN OBSERVATORY

Organisation Européenne pour des Recherches Astronomiques dans l'Hémisphère Austral

Europäische Organisation für astronomische Forschung in der südlichen Hemisphäre

VERY LARGE TELESCOPE

UVES Pipeline User Manual

VLT-MAN-ESO-19500-2965

Issue 22.17

Date 2018-04-01

5 Quick start

This section describes how to make immediate usage of the UVE

5.1 UVES pipeline recipes

The current UVES pipeline is based on a set of 8 stand-alone reci

uves_cal_mbias: Creates a master bias frame.

uves_cal_mdark: Creates a master dark frame.

uves_cal_predict: Implements the UVES physical model.

uves_cal_orderpos: Defines the echelle order positions.

uves_cal_mflat: Creates a master flat field frame.

uves_cal_wavecal: Performs the wavelength calibration.

uves_cal_response: Determines the response function and quar

uves_obs_scired: Reduces a science frame.

ESO	UVES Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-2965
		Issue:	Issue 22.17
		Date:	Date 2018-04-01
		Page:	24 of 115

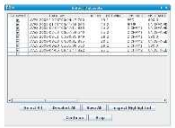
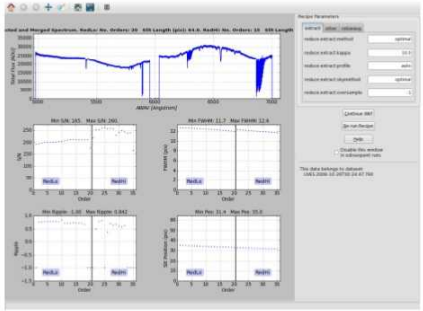


Figure 5.5: *The "Select Datasets" pop-up window. On purpose the demo includes an incomplete (grey) DataSet.*



ESO	UVES Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-2965
		Issue:	Issue 22.17
		Date:	Date 2018-04-01
		Page:	51 of 115

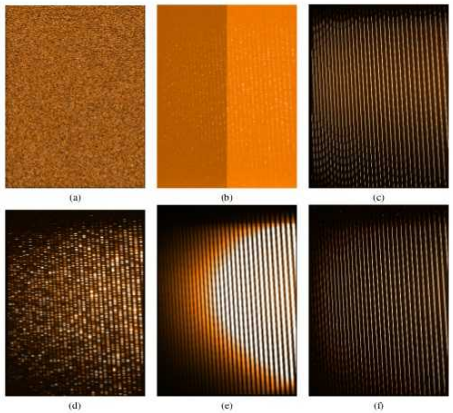


Figure 7.1: (a) a raw bias frame; (b) a raw format check frame; (c) a raw order frame; (d) a raw arc lamp frame; (e) a raw flatfield frame; (f) a raw std star frame.

esorex - -create-config uves_cal_wavecal

The definition of one parameter of a recipe may look like this:

```
# --tolerance
# tolerance or tit. if positive, 'tolerance' is in pixel units. if negative,
# abs('tolerance') is in wavelength units. Lines with residuals worse than
# the tolerance are excluded from the final fit. Unlike in previous versions,
# this parameter is not corrected for CCD binning.
uves_cal_wavecal.calibrate.tolerance=0.6
```

In this example, the parameter `uves_cal_wavecal.calibrate.tolerance` is set to the value `0.6`. In the configuration file generated by *EsoRex*, one or more comment lines are added containing information about the possible values of the parameter, and an alias that could be used as a command line option.

A recipe configuration file different from the default one may be specified on the command line:

esorex - -recipe-config=my_alternative_recipe_config

Recipe parameters are provided in Section 10 and their role is described in Section 11.

More than one configuration file may be maintained for the same recipe but, in order to be used, a configuration file not located under `$HOME/.cooros`, or having a name different from the recipe name, should be explicitly specified when launching a recipe.

Recipe execution: A recipe can be run by specifying its name to *EsoRex*, together with the name of a set-of-frames. For instance, the following command would be used to run the recipe *uves_cal_wavecal* to process the files specified in the set-of-frames `uves_cal_wavecal.sof`:

esorex uves_cal_wavecal uves_cal_wavecal.sof

The recipe parameters may be modified either by directly editing the configuration file that is used, or by specifying new parameter values on the command line using the command line options defined for this purpose. Such command line options should be inserted after the recipe name and before the SOF name, and they will supersede the system defaults and/or the configuration file settings. For instance, to set the *uves_cal_wavecal* recipe *tolerance* parameter to `0.07`, the following command should be entered:

esorex uves_cal_wavecal - -tolerance=0.07 uves_cal_wavecal.sof

Every recipe provides a parameter *debug* which – when enabled – causes intermediate results to be saved to the local directory. This allows a more detailed inspection of the recipe execution.

esorex uves_cal_wavecal - -debug uves_cal_wavecal.sof

Basic plotting functionality can be enabled on systems which have the gnuplot tool:

esorex uves_cal_wavecal - -plotter='gnuplot -persist' uves_cal_wavecal.sof

For more advanced visualisation a dedicated FITS viewer should be used.

For more information on *EsoRex*, see [11]

10.5 uves_cal_mflat

The recipe uves_cal_mflat creates the master flat frame.

10.5.1 Input

frame tag/category	nr	filename example
FLAT_ARM	+	/path_raw/uves_flat_arm.fits
MASTER_BIAS_CHIP	?!	/path_pro/masterbias_chip.fits
MASTER_DARK_CHIP	?	/path_pro/masterdark_chip.fits
ORDER_TABLE_CHIP	1	/path_pro/ordertable_chip.fits

The optional input master dark is subtracted if provided. The UVES dark count level is usually negligible.

10.5.2 Output

default recipe file name	format	PRO.CATG	short description
masterflat_chip.fits	image	MASTER_FLAT_CHIP	master flat frame
masterflat_bkg_chip.fits	image	BKG_FLAT_CHIP	background of master flat frame

10.5.3 Quality control

The pipeline monitors the number of coadded frames (PRO.DATANCOM). Additional quality control information is monitored by DFO and can be found at `www.eso.org/qc`.

10.5.4 Parameters

parameter	alias	default	min	max
uves_cal_mflat.norm_method	norm_method	explevel		
uves_cal_mflat.backsub.mmethod	backsub.mmethod	median		
uves_cal_mflat.backsub.npoints	backsub.npoints	82	0	
uves_cal_mflat.backsub.radiusy	backsub.radiusy	2	0	
uves_cal_mflat.backsub.sdegree	backsub.sdegree	1	0	
uves_cal_mflat.backsub.smoothx	backsub.smoothx	-1.0		
uves_cal_mflat.backsub.smoothy	backsub.smoothy	-1.0		

Note that the default parameters are robust.

Deliverables: Documentation

Data Reduction Library Specifications: describes instrument modes and configurations, data format, gives a data processing overview and detailed description of the pipeline recipes and QC parameters. It also includes a development plan.

Data Reduction Library Validation and Test: describes a validation plan based on simulated and actual instrument test data (at different stages these include data lab data, data from similar instruments, from commissioning and science verification).

Data Reduction Library Design: gives mathematical and functional description of the DRL, instrument data description, describes the CPL plugins, the validation, including tests and accuracy control; includes a development plan.

Exposure Time Calculator Specifications: descrives the instrument configurations, some characteristic data, the User Interface, a mathematical model and calculation functions, a validation sets.

Science-Grade Data Products

- Calibrated in physical units with **error estimates**
- No residual systematic error (put a grain of salt here)
- S/N close to the achievable optimum (e.g. ETC prediction)
- OB combination e.g. fully calibrated and mosaiced images / megred spectra
- **Standard** data formats
- SGDPs are independent of the science goals

Accuracy
Efficiency
Robustness

Instrument Model
Optical/Detector model
Stability and reproducibility
No assumptions on science
Optimization strategy
Computational complexity
Robustness, Fault tolerance

Input Data
Signal-to-noise
Missing data
Contamination
Stability (Atm., Ins.)
Multiplex/Volume

→

Pipeline Recipe
Default parameters
Fixed calibration data

→

Reduced Data
Physical units
Random error estimates
Systematic errors

Science-Grade Data Products

DICD - Data Interface Control Document

Data Interface Control Document

Document Number: ESO-04415

Document Version: 6

Document Type: Specification (S

Released On: 2016-06-21

Document Classification: Publ

Table 1: Primary FITS keywords used at ESO in primary HDU and extensions

Type	Keyword	Example	Explanation
		Primary HDU	
(L)	SIMPLE	T	Standard FITS (NOST-100-2.0)
(I)	BITPIX	16	# bits storing pix values
(I)	NAXIS	2	# of axes in frame
(I)	NAXIS1	2080	# of pixels/row
(I)	NAXIS2	2048	# of rows (also # of scan lines)
(L)	EXTEND	T	Extensions may be present
(R)	BZERO	32768.0	value = fits-value*BSCALE+BZERO
(R)	BSCALE	1.0	value = fits-value*BSCALE+BZERO
(S)	BUNIT	'adu'	Physical unit of array values

- MJD-END is the modified Julian Date (JD-2400000.5) of the end of the observation. Analogous to MJD-OBS.
- DATE-END gives the date in which the exposure was ended, in the restricted ISO 8601 format, YYYY-MM-DDThh:mm:ss.sss. Analogous to DATE-OBS.
- TIMESYS lists the standard abbreviation of the principal time system used for the time-related keywords and the data. This keyword must be present only if the system used is other than UTC. Allowed values are listed in Table 2 of [RD9].
- UTC and LST give the time in seconds elapsed since midnight of the start of the exposure as known to TCS. The time on TCS is synchronised with the observatory

Science-Grade Data Products

DICD - Data Interface Control Document

Data Interface Control Document

Document Number: ESO-04415

Document Version: 6

Document Type: Specification (S

Released On: 2016-06-21

Document Classification: Publ

Table 11: DPR.TECH: examples of principal values (first group), qualifiers (second group) and instrument-specific qualifiers (third group)

Value	Explanation
IMAGE	Any picture
SPECTRUM	Sngle-order spectrum
ECHELLE	Cross-dispersed spectrum
MOS	Frame with spectra of several objects
MXU	Frame with spectra of several objects using a pre-manufactured mask
IFU	Integral Field Unit observation
POLARIMETRY	Polarimetric exposure
CORONOGRAPHY	Coronography exposure
INTERFEROMETRY	Coherent exposure with more than one telescope beam

Table 16: Physical units allowed in ESO DIDs

Quantity	Unit String	Meaning or use
	SI base and supplementary units	
Length	m	meter
Mass	kg	kilogram
Time	s	second of time
Plane angle	rad	radian
Solid angle	sr	steradian
Temperature	K	kelvin

Science-Grade Data Products

DICD - Data Interface Control Document

Data Interface Control Document

Documen

Doc

Document

Relea

Documer

4.7.1 Products generated at ESO

This subsection is relevant for all products generated at ESO, i.e. Master Calibrations and products resulting from the runs of ESO-supported pipelines.

Keyword specifications described in the preceding sections shall also apply to ESO-generated products, with the following additional requirements:

- The files shall contain the Boolean keyword PRO.SCIENCE which shall be set to T for science files and F for all other files. Archived files generated prior to the introduction of this requirement shall assume the value of this keyword of F.
- The files shall not contain keywords from the DPR category, i.e. keywords described in Section 4.4.2.1.
- The files shall contain the PRO.TECH keyword, which shall contain the value from the DPR.TECH keyword in the original file.
- The following keywords from the PRO dictionary:

  PIPEFILE
  PRO.DID
  PRO.REC1.ID
  PRO.REC1.DRS.ID
  PRO.REC1.PIPE.ID
  PRO.DATANCOM
  PRO.CATG
  DATAMD5

  are mandatory in processed science and calibration frames.

Science-Grade Data Products

ESO Science Data Products Standard

Data Management and Operations Division

Phase 3 User Documentation

ESO Science Data Products Standard

Table 2: Pre-defined values for the keyword PRODCATG

PRODCATG	Data format	Characteristics
SCIENCE.IMAGE	Single image	Single image stored in the primary HDU.
SCIENCE.MEFIMAGE	MEF image	Multiple images stored in multi-extension

2.3 Science products and associated ancillary data

The ESO Archive is directly searchable for science data products using the respective query forms available on the ESO Archive web pages. Ancillary data products are associated to scientific data products to support their exploitation without being directly searchable through the ESO Archive interfaces.

Certain science data products require the submission of specific ancillary products as given in the data format definitions, for instance mosaicked images, like the VISTA tile image, require a weight map image to be associated.

Moreover, depending on the kind of scientific data, the data provider can deliver additional ancillary products. Typical examples are preview images, graphics or reports generated in the course of the data reduction process. The file type of ancillary products may be, for instance, JPEG or PNG image, PS or PDF graphics, ASCII text, or in FITS file format.

Ancillary data products are defined in the FITS header of the corresponding science data product in terms of the following indexed keywords.

Type[9]	Keyword	Description
(S)	ASSON*i*	The list of files associated to this data product. *i* is a sequential number starting from 1. If *n* files are associated to the product, the indexed keywords ASSON*i* and ASSOC*i* should appear *n* times

Science-Grade Data Products

ESO Science Data Products Standard

Data Manag

P

ES

2.4 *Processing Provenance*

2.4.1 PROV keywords

Type	Keyword	Description
(S)	PROV*i*	Processing provenance in terms of the list of science files originating this data product. *i* is a sequential number starting from 1 with no leading zeros. PROV*i* should appear as many times as needed to identify the complete set of science data files this product has been generated from. PROV*i* are pointers to files in the ESO Archive, which can be original raw data or (intermediate) data products depending on the level of reduction. Raw files are specified in terms of their ARCFILE name, e.g. 'VCAM.2010-03-27T08:58:58.083.fits'. Data products may be referenced by their ARCFILE name, or, alternatively, by their ORIGFILE name supposed that they belong to the same Phase 3 data collection as the referencing product file. ARCFILE and ORIGFILE references should not be mixed within the same file.

Guidelines

- PROV*i* keywords must reside in the primary HDU of the FITS file;
- PROV*i* records represent pointers to files rather than pointers to FITS extensions, i.e. there is no trailing extension number in square brackets (see also the examples below);
- The requirement that PROV must refer to files in the ESO archive has the consequence that intermediate products to be referenced must be submitted to ESO not later than the products resulting thereof. It means for example that one cannot submit extracted source lists in the first data release and then the originating images afterwards in the second release.[10]
- If the processing provenance exceeds 999 records, then it is required to record the complete list of files in one dedicated FITS binary table (BINTABLE) extension instead of using header keywords (cf. §2.4.2). The header should not contain any PROV*i* keyword in this case to avoid ambiguity.

Example 1: The shallow H-band survey image (16 seconds effective exposure time, part of VVV survey Data Release 1, archived under ADP.2011-06-24T14:56:11.033) originates from 12 raw science data files (6 pawprints with 2 jitter positions each) identified by their ARCFILE names.

```
PROV1   = 'VCAM.2010-03-27T08:58:58.083.fits' / Originating raw science file
PROV2   = 'VCAM.2010-03-27T08:59:10.530.fits' / Originating raw science file
PROV3   = 'VCAM.2010-03-27T08:59:24.541.fits' / Originating raw science file
PROV4   = 'VCAM.2010-03-27T08:59:36.978.fits' / Originating raw science file
PROV5   = 'VCAM.2010-03-27T08:59:51.043.fits' / Originating raw science file
```

Science-Grade Data Products

ESO Science Data Products Standard

Table 10: Keywords specific to the single object 1D spectrum in the binary table format

Type	Keyword	Description
(S)	VOCLASS	The data model name and version: 'SPECTRUM V1.0'.
(S)	VOPUB	The name of the publisher, namely 'ESO/SAF'.
(S)	TITLE	The title of the dataset is a short, human-readable description of a dataset, and should be less than one line of text. The value of the OBJECT keyword should be propagated here.
(R)	APERTURE	Aperture angular size, in degrees [deg]. It shall be set to the width of the slit or the diameter of the fiber.
(R)	TELAPSE	Total elapsed time in seconds [s], defined as MJD-END-MJ OBS.
(R)	TMID	Exposure midpoint (MJD). It shall be set to (MJD-OBS+MJD END)/2.0.
(R)	SPEC_VAL	Characteristic spectral coordinate value in nanometers [nm] Should WCS information be available, it can be calculated a CRVAL1+(0.5-CRPIX1+NAXIS1*0.5)*CDELT1. Otherwise, i to (WAVELMAX+WAVELMIN)/2.0.
(R)	SPEC_BW	Width of the spectrum in nanometers [nm]. Should WCS information be available, it can be calculated as NAXIS1*CD Otherwise, it is set to WAVELMAX-WAVELMIN.

Each field of the BINTABLE shall be further described in the extension header as spec 11. Mandatory fields shall be WAVE, FLUX, and ERR, in that particular order. Addition added to the BINTABLE, provided that at least the values for their type, format, and un provided and described by the PI in the release description. Examples of such addition given in Table 11.

Table 11: Keywords describing the BINTABLE columns in the 1D spectrum binary ta

Type	Keyword	Description	Allowed values
(I)	TFIELDS	Number of fields (columns) in the binary table.	

Appendix A Sybase IQ Reserved Words

Source of information:
http://infocenter.sybase.com/help/index.jsp?topic=38151.1510/html/iqrefbb/Alhakeywords.htm

SQL reserved words

active	add	all	algorithm
alter	and	any	append
as	asc	auto	backup
begin	between	bigint	binary
bit	bottom	break	by
calibrate	calibration	call	cancel
capability	cascade	case	cast
certificate	char	char_convert	character
check	checkpoint	checksum	clientport
close	columns	comment	commit
committed	comparisons	computes	conflict
connect	constraint	contains	continue

Project organization

Scheduling, Validation and Acceptance

Phase A (via PDR) to FDR	Documentation/ Prototyping	2-3 years
FDR to PAE	Development/ Simulation	2-3 years
PAE to SV and SOP	Deployment	6-12 months
Operational Phase	Maintenance	10 years

Project organization

Phase A and PDR Preparation:

- reuse existing documents as templates
- identify extra needs: observation preparation, visualization, link to the data analysis?

Learning CPL before the FDR:

- first recipes in CPL, establishing coding team, standards and procedures, memory management, …
- review existing pipelines, develop prototypes
- coding starts become official only after the FDR

FDR Preparation:

- prototype algorithms (test and simulated data)
- new document: Validation and Test Plan
- observatory Pipeline is template based
- further processing requirements
- data combination, interactive processing (Reflex)

Project organization

FDR to PAE:

- plan validation effort for the first recipes
- plan enough time for testing and finalizing
- regular intermediate software releases and test reports every 3-6 months
- improving simulated or adapted data

PAE:

- schedule the complete set of recipes to be ready
- validated data reduction algorithms (using lab and simulated data)
- if needed: alternative data and verification methods

Commissioning to Science Verification:

- test and validate on instrument and sky data
- identify and solve the unexpected problems

DICB Issues

DPR Keyword Values:

- deliver organization, classification and association (OCA) rules, Ideally, based only on DPR keywords
- only valid keyword values listed in the DICB 4.0 doc
- submit proposal to DICB by FDR if new values needed!

DICB Validation:

- no redefinition of existing keywords (database at www.eso.org/dicb)
- keyword names should not be too long
- avoid underscores and special characters

Multi-HDU files:

- always the same structure of data for a given data type
- data extensions come first, auxiliary and optional tables afterward

Acceptance Tests

Usage of CPL recipe template:
- following CPL coding standard
- usage of external libraries
- namespace protection

Execution Tests:
- completeness of the set of recipes and DRL functions
- availability/representativity of test data
- proper execution of recipes
- generation of products
- memory leaks
- unit tests
- documentation

Detailed Validation:
- correctness of results
- validation of input
- FITS compliance
- user-friendly documentation
- data reduction cascade
- unit tests

Performance and Portability:
- execution speed
- standard platform - Scientific Linux

Deliverables: …

… many pages of saying to plan ahead and to be prepared for the unexpected, because the unexpected is the only thing that can be expected to happen for sure..… many pages of saying to plan ahead and to be prepared for the unexpected, because the unexpected is the only thing that can be expected to happen for sure..… many pages of saying to plan ahead and to be prepared for the unexpected, because the unexpected is the only thing that can be expected to happen for sure..… many pages of saying to plan ahead and to be prepared for the unexpected, because the unexpected is the only thing that can be expected to happen for sure..… many pages of saying to plan ahead and to be prepared for the unexpected, because the unexpected is the only thing that can be expected to happen for sure...

Data reduction environments

Observatory Pipeline:

- on-the-fly data processing (event driven)
- template-based processing
- static calibration database (but certified)

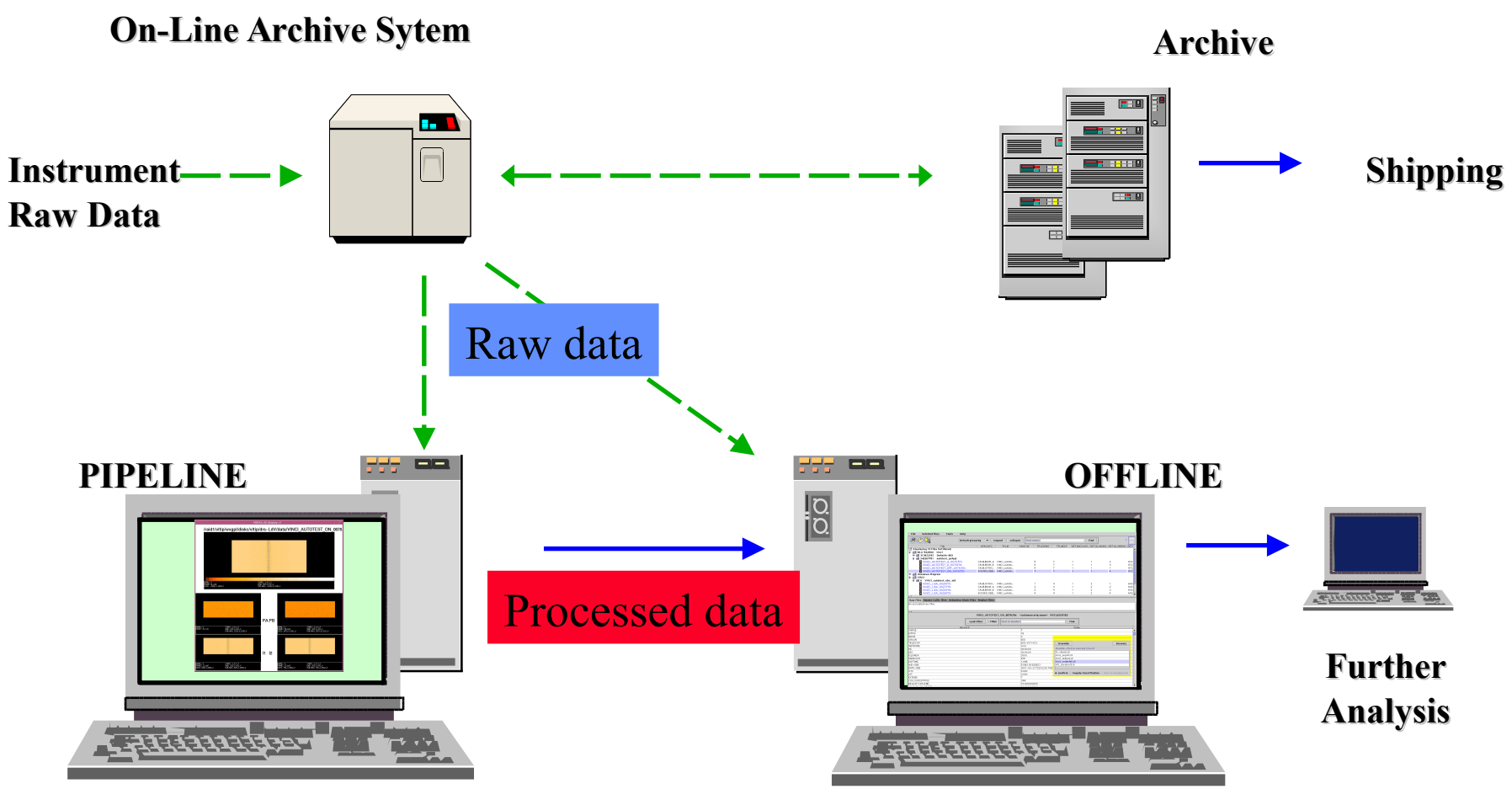
Quality Control Pipeline:

- batch processing of complete data sets (all science and calibration data produced by one ESO instrument in one night) (instead, Advance Data products, much later!)
- best available calibrations are used => data must be organized according to the Calibration Cascade

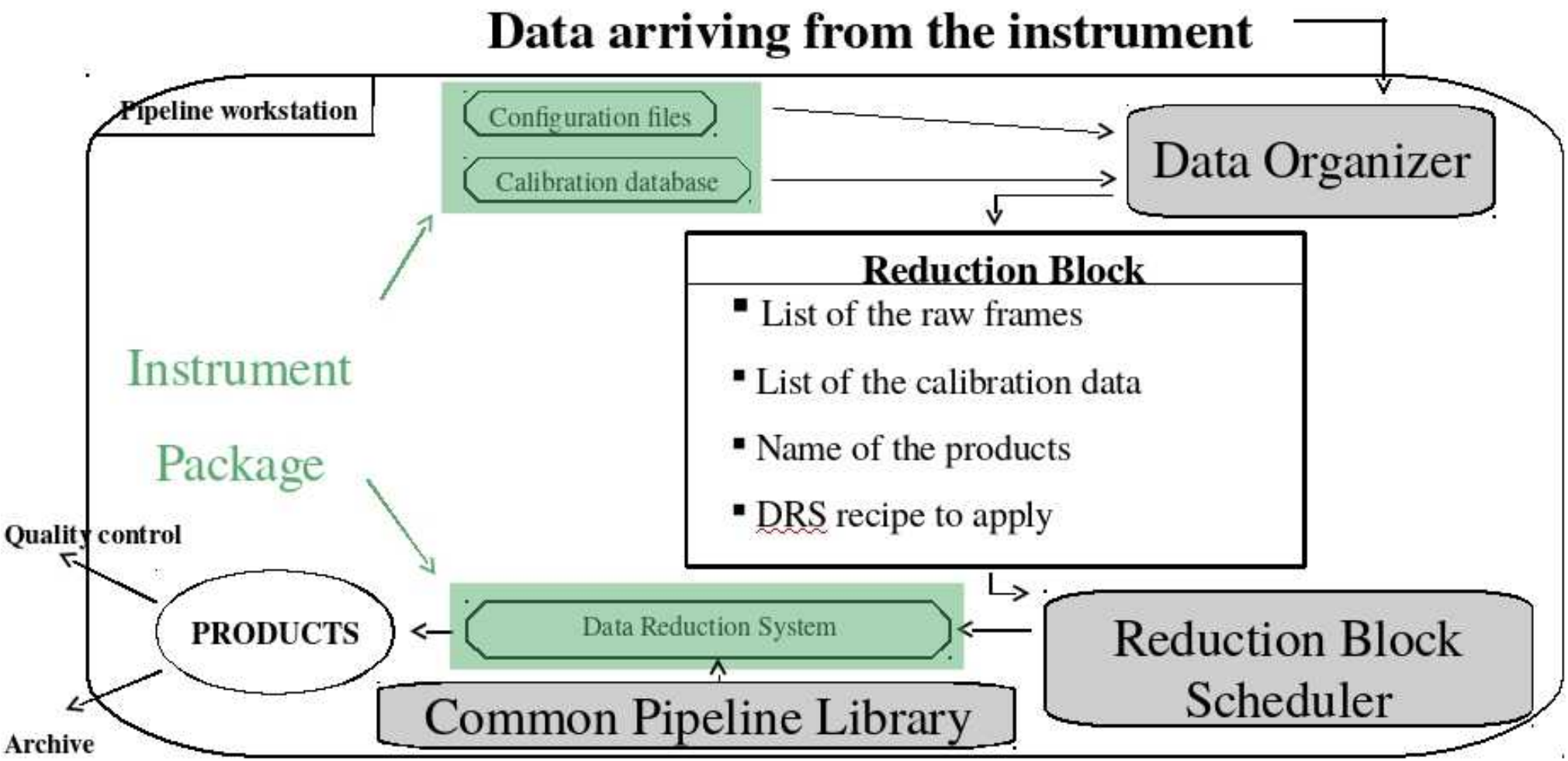
Desktop science-grade data reduction:

- modular and additional recipes are available
- several front-ends for scripting (esorex), browsing (Gasgano), interactive (Reflex)

Observatory Pipeline



Observatory Pipeline



QC pipeline: QC logs

```
12:21:51>-START GROUP / Start [AMBER]
12:21:51> ARCFILE = 'AMBER.2007-03-21T12:16:08.099.fits'  [AMBER]
12:21:51> TELESCOP = 'NOT_SPECIFIED' / Telescope [AMBER]
12:21:51> INSTRUME = 'AMBER' / Instrument name [AMBER]
12:21:51> OBSERVER = 'UNKNOWN' / Observer name [AMBER]
12:21:51> PIPEFILE = 'p2vm.fits' / Filename of data product [AMBER]
12:21:51> INS GRAT1 NAME = 'GHR' / Grating common name. [AMBER]
12:21:51> INS GRAT1 RESOL = 527.778; / Encoder resolution [Enc/deg]. [AMBER]
12:21:51> INS GRAT1 WLEN = 2364.972; / Grating central wavelength [nm]. [AMBER]
12:21:51> INS GRAT1 ZORDER = 40319; / Grating zero order position [Enc]. [AMBER]
12:21:51> INS GRIS1 NAME = 'NAR_SLT' / OPTIi name. [AMBER]
12:21:51> INS GRIS2 NAME = '3T_K' / OPTIi name. [AMBER]
12:21:51> INS MODE = '3Tstd_High_K_1_2.365' / Instrument mode used. [AMBER]
12:21:51> PRO DID = 'ESO-VLT-DIC.PRO-1.15' / Data dictionary for PRO [AMBER]
12:21:51> PRO CATG = 'P2VM_REDUCED' / pipeline product category [AMBER]
12:21:51> PRO TYPE = 'REDUCED' / Product type [AMBER]
12:21:51> PRO REC1 ID = 'amber_p2vm' / Pipeline recipe (unique) identifier [AMBER]
12:21:51> PRO REC1 DRS ID = 'cpl-3.0' / Data Reduction System identifier [AMBER]
12:21:51> PRO REC1 PIPE ID = 'AMBER/2.3.2' / Pipeline (unique) identifier [AMBER]
12:21:51> PRO REC1 RAW1 NAME = 'AMBER.2007-03-21T12:13:38.479.fits' / File name of raw frame [AMBER]
12:21:51> PRO REC1 RAW1 CATG = 'AMBER_3WAVE' / Frame category of raw frame [AMBER]
12:21:51> PRO DATANCOM = 14; / Number of frames combined [AMBER]
12:21:51> PRO REC1 CAL1 NAME = 'FlatFieldMap.fits' / File name of calibration frame [AMBER]
12:21:51> PRO REC1 CAL1 CATG = 'AMBER_FLATFIELD' / Frame category of calibration frame [AMBER]
12:21:51> PRO REC1 PARAM1 NAME = 'dummy' / Name of recipe parameter [AMBER]
12:21:51> DET NTEL = 3; / Number of telescopes [AMBER]
12:21:51> QC P1 OFFSETY = 0.02; / Offset wavelength calibration [AMBER]
12:21:51> QC P2 OFFSETY = 0.05; / Offset wavelength calibration [AMBER]
12:21:51> QC P3 OFFSETY = 0.03; / Offset wavelength calibration [AMBER]
12:21:51>-STOP GROUP / Stop [AMBER]
```

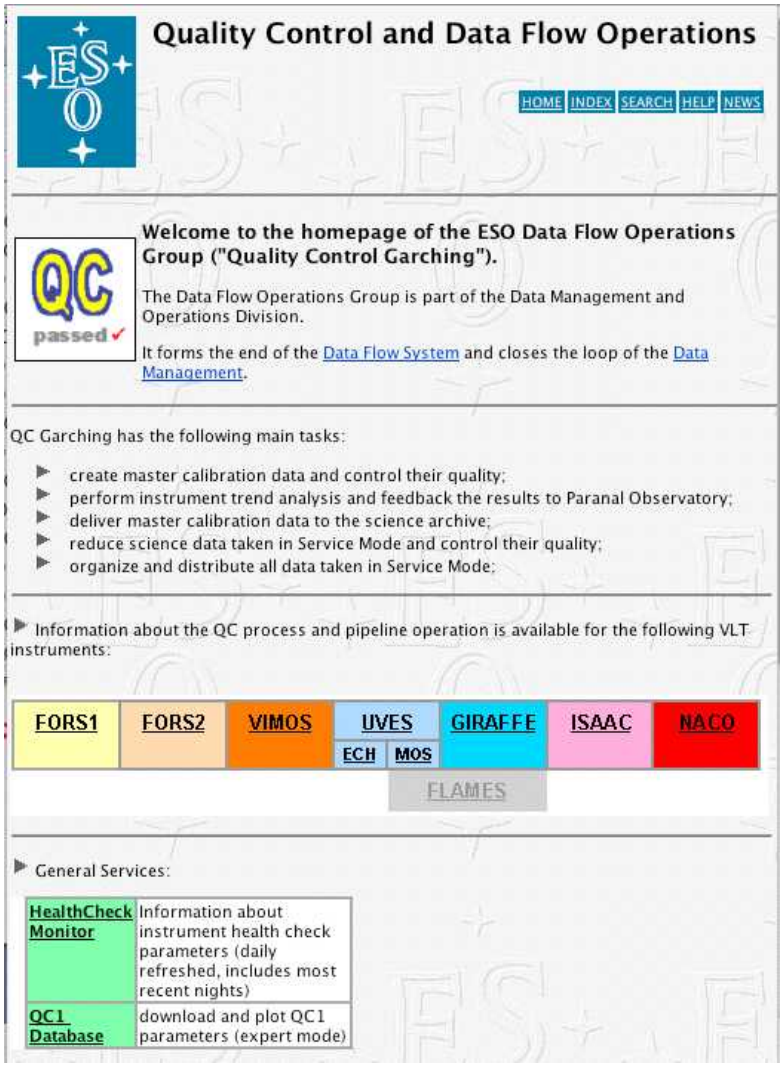
QC pipeline

Main Tasks:

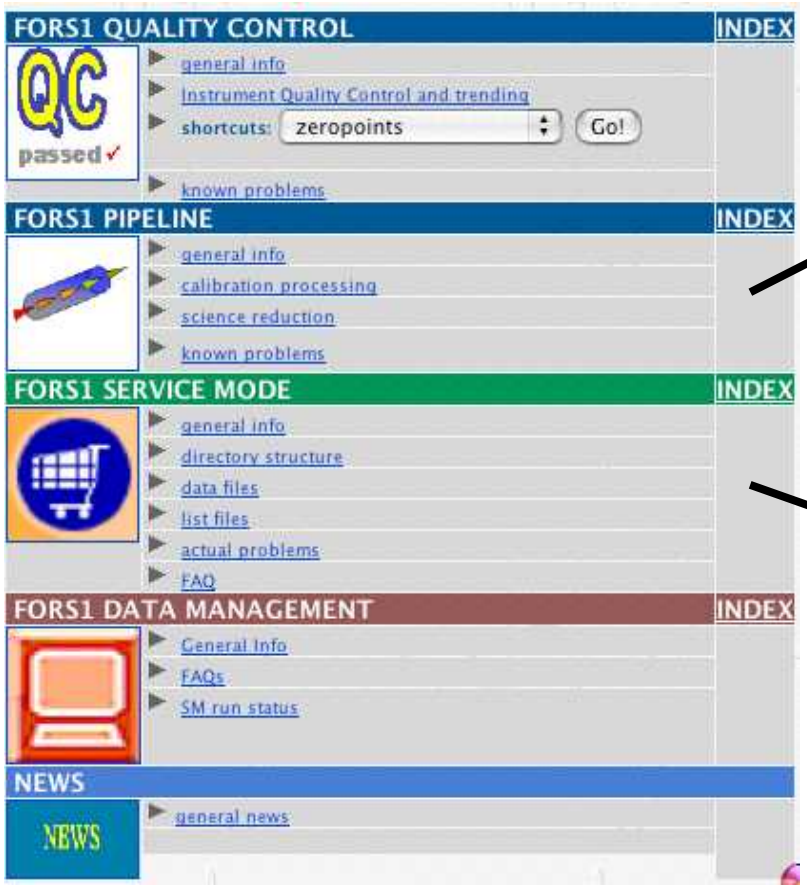
- create master calibrations
- pipelines & QC parameters: requirements & testing
- monitor QC trends
- create advance science products (new!)

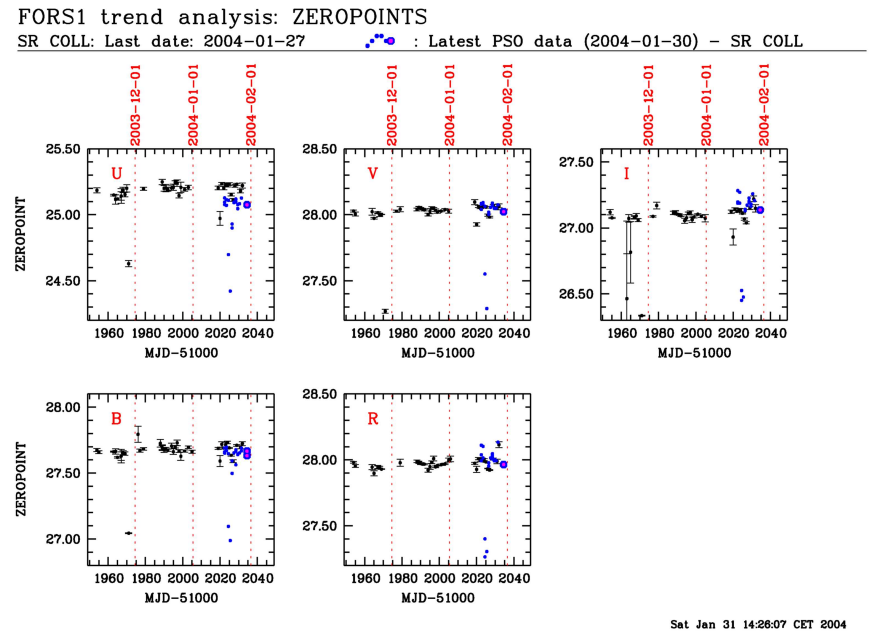
Customers:

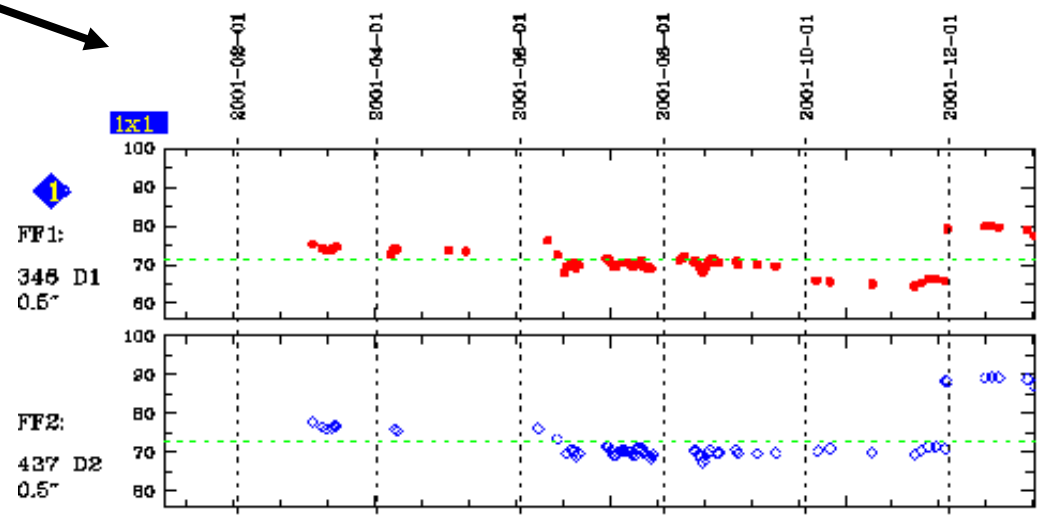
- Paranal Science Operations
- ESO community (PIs, archive users)



QC pipeline


FORS1 QUALITY CONTROL
INDEX
QC
passed
general info
Instrument Quality Control and trending
shortcuts:
zeropoints
Go!
known problems
FORS1 PIPELINE
INDEX
general info
calibration processing
science reduction
known problems
FORS1 SERVICE MODE
INDEX
general info
directory structure
data files
list files
actual problems
FAQ
FORS1 DATA MANAGEMENT
INDEX
General Info
FAQs
SM run status
NEWS
NEWS
general news


FORS1 trend analysis: ZEROPOINTS
SR COLL: Last date: 2004-01-27
: Latest PSO data (2004-01-30) - SR COLL
2003-12-01
2004-01-01
2004-02-01
U
V
I
B
R
ZEROPOINT
MJD-51000
Sat Jan 31 14:26:07 CET 2004


1x1
2001-02-01
2001-04-01
2001-06-01
2001-08-01
2001-10-01
2001-12-01
FF1:
346 D1
0.5"
FF2:
437 D2
0.5"

Desktop Data Reduction: Gasgano

VLT interactive data organisation tool

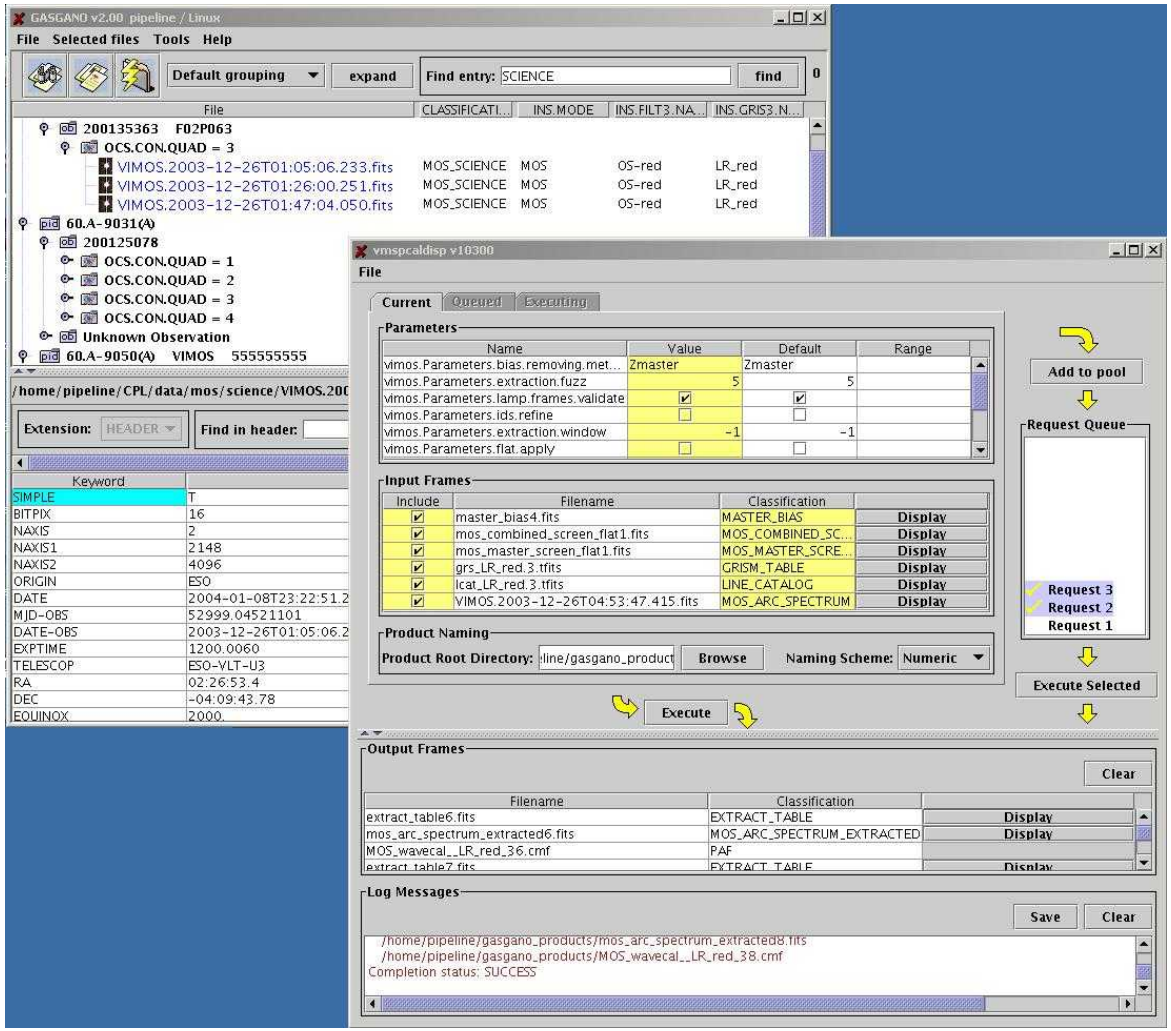
- FITS file browsing
- grouping
- classification

Interactive front-end

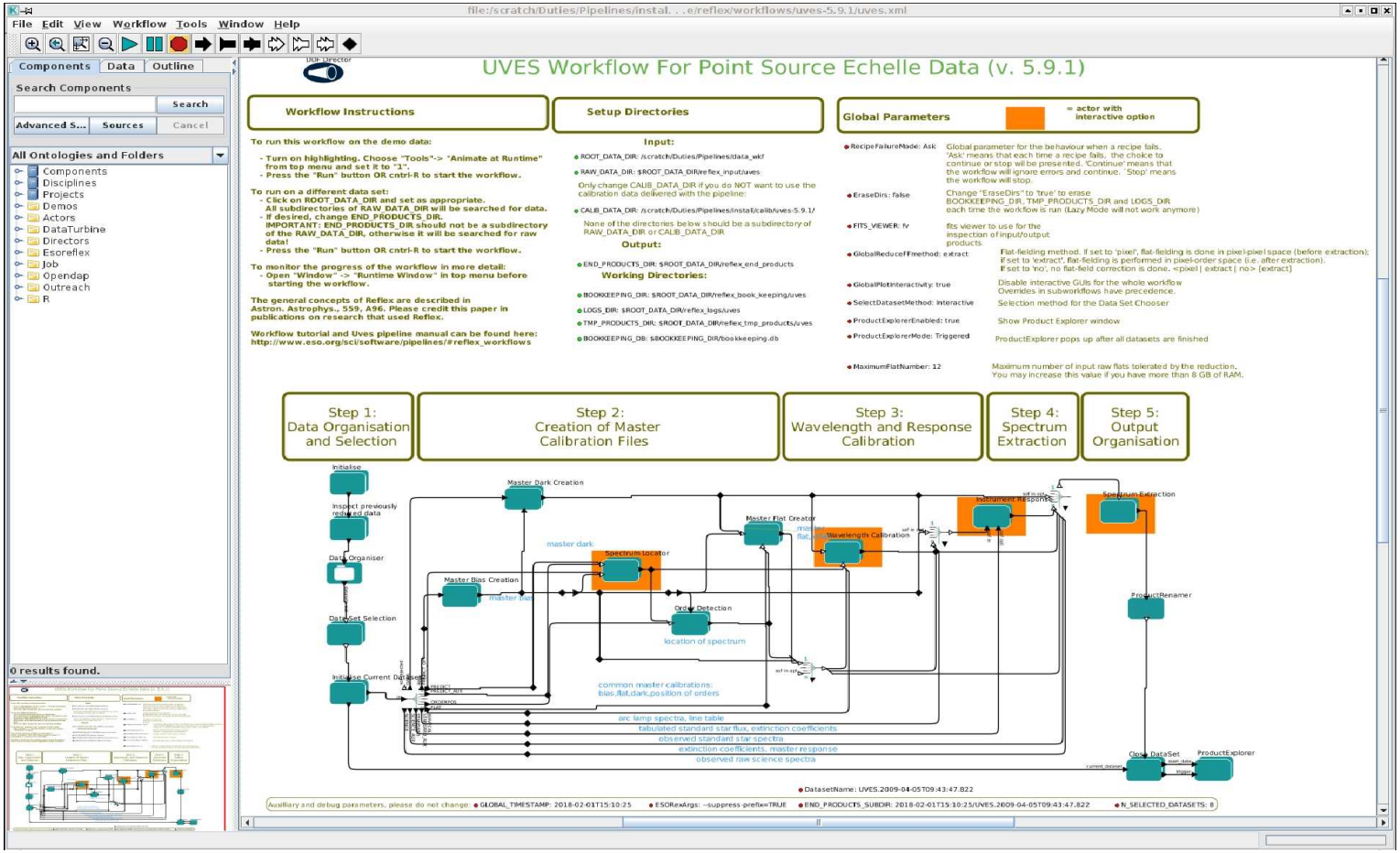
- interface to CPL plugins
- interface to visualization tools

Features

- java language
- FITS format



Desktop Data Reduction: Reflex



Interactive front-end to the (same!) CPL recipes plus external tools such as Python scripts and better visualization!

Pipeline Maintenance

DFS Tickets:

- issued by PSO, UC, USD and the user (via USD)

Pipeline priority meetings:

- bi-yearly meetings with representation of INS, DFO, PSO, SDD for projects/upgrades/development
- general issues and for each instrument closed, in process, open tickets
- priority setting for further pipeline development

Instrument Evolution:

- commissioning and pipeline support of new modes
- upgrades

Sumamry

It is complex...