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!
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!

Other relevant docs:
- CPL Documentation
- DICB specs
- Gasgano manual
- Reflex development
Deliverables and integration

- Observation Handling
  - Phase II Proposals
  - ST/MT Scheduler
  - OB Repository
  - Ambient Data Configuration Status
- Observation Blocks
- VLT Control System
  - OBSERVATION
  - Data Frames ObsBlocks
- Science Archive
  - Catalogue Server
  - On-line Archive
  - Storage & Data Distribution

- Programme Handling
  - Phase I Proposals
  - LT Scheduler
- Quality Control
  - Calibration Plan
  - Simulators
  - Quality Monitor
  - Quality Scheduler
  - Technical Programmes
- Pipeline
  - Data Organizer
  - Reduction Scheduler
  - Reduction Block
  - Reduced and Calibrated Frames
  - Reduced Frames
  - Recipes
  - Data Frames
  - Calibration Data
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).
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
Deliverables: instrument package

---

PLATOSpec workshop 2, Ondrejov 29.-30.10.2018
Deliverables: instrument package

EVM

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

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

# Collect Phase 1 info
collectPhaseInfo

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 list0fObsBlocks {}
  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 {[lindex $objType] > 1} {
    set containerInfo [lindex $object 0]

    set objType [lindex $containerInfo 6]; # Type of container
    set containerID [lindex $containerInfo 7]; # 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

**Templates exist!**

Deliverables: ETC

**PLATOSpec workshop 2, Ondrejov 29.-30.10.2018**
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.
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

- Observatory Pipeline
- Quality Control Pipeline
- Science-Grade Desktop Environment

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

Reflex Workflow

+ Reflex Workflow

DICB/Archive
Data Flow Operations
INS Commissioning Team
Paranal Science Operations
Public Release
User Community
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 functions 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 functions):
5 Quick start
This section describes how to make immediate usage of the UVE.

5.1 UVES pipeline recipes
The current UVE pipeline is based on a set of 8 stand-alone recipes:

- uves_cal_phos: Creates a master bias frame.
- uves_cal_mdark: Creates a master dark frame.
- uves_cal_prord: Implements the UVE physical model.
- uves_cal_redaxes: Defines the echelle order positions.
- uves_cal_hdf: Creates a master flat field frame.
- uves_cal_wavelength: Performs a wavelength calibration.
- uves_cal_response: Determines the response function and quality.
- uves_obs_science: Reduces a science frame.

Note: the documentation manual contains more detailed information about the UVE pipeline. The example recipe uves_cal_mdark illustrates the usage of the pipeline.

10.5 uves_cal_phos
This recipe uves_cal_phos creates a master flat frame.

10.5.1 Input
- filename: Input flat file
- master_file: Master dark frame
- master_darks: Bias frames
- master_charge: Charge transfer frames
- master_darks: Dark frames
- master_dark0: Dark frames

10.5.2 Output
- master_file: Master flat frame
- master_charge: Charge transfer frames
- master_dark0: Dark frames
- master_darks: Dark frames
- master_file: Bias frames

The optimal input source dark is provided, if provided. The UVE dark source is usually negligible.

10.5.3 Quality control
The pipeline monitors the number of invalid frames (BadFrac) in the output.

10.5.4 Parameters
- filename: Input flat file
- master_file: Master dark frame
- master_darks: Bias frames
- master_charge: Charge transfer frames
- master_darks: Dark frames
- master_dark0: Dark frames
- master_file: Bias frames
- master_charge: Charge transfer frames
- master_darks: Dark frames
- master_dark0: Dark frames
- master_darks: Dark frames

Note that the default parameters are:...
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:** describes 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

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

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

**Pipeline Recipe**
- Default parameters
- Fixed calibration data

**Reduced Data**
- Physical units
- Random error estimates
- Systematic errors

**Accuracy**
**Efficiency**
**Robustness**
Science-Grade Data Products

DICD - Data Interface Control Document

Data Interface Control Document

<table>
<thead>
<tr>
<th>Table 1: Primary FITS keywords used at ESO in primary HDU and extensions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong></td>
</tr>
<tr>
<td>----------</td>
</tr>
<tr>
<td>(L)</td>
</tr>
<tr>
<td>(I)</td>
</tr>
<tr>
<td>(I)</td>
</tr>
<tr>
<td>(I)</td>
</tr>
<tr>
<td>(I)</td>
</tr>
<tr>
<td>(L)</td>
</tr>
<tr>
<td>(R)</td>
</tr>
<tr>
<td>(R)</td>
</tr>
<tr>
<td>(S)</td>
</tr>
</tbody>
</table>

- **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's time system via a dedicated time and date principle. UTC and LST should be compatible.
### Table 11: DPR.TECH: examples of principal values (first group), qualifiers (second group) and instrument-specific qualifiers (third group)

<table>
<thead>
<tr>
<th>Value</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMAGE</td>
<td>Any picture</td>
</tr>
<tr>
<td>SPECTRUM</td>
<td>Single-order spectrum</td>
</tr>
<tr>
<td>ECHELLE</td>
<td>Cross-dispersed spectrum</td>
</tr>
<tr>
<td>MOS</td>
<td>Frame with spectra of several objects</td>
</tr>
<tr>
<td>MXU</td>
<td>Frame with spectra of several objects using a pre-manufactured mask</td>
</tr>
<tr>
<td>IFU</td>
<td>Integral Field Unit observation</td>
</tr>
<tr>
<td>POLARIMETRY</td>
<td>Polarimetric exposure</td>
</tr>
<tr>
<td>CORONOGRAPHY</td>
<td>Coronography exposure</td>
</tr>
<tr>
<td>INTERFEROMETRY</td>
<td>Coherent exposure with more than one telescope beam</td>
</tr>
</tbody>
</table>

### Table 16: Physical units allowed in ESO DIDs

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Unit String</th>
<th>Meaning or use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>m</td>
<td>meter</td>
</tr>
<tr>
<td>Mass</td>
<td>kg</td>
<td>kilogram</td>
</tr>
<tr>
<td>Time</td>
<td>s</td>
<td>second of time</td>
</tr>
<tr>
<td>Plane angle</td>
<td>rad</td>
<td>radian</td>
</tr>
<tr>
<td>Solid angle</td>
<td>sr</td>
<td>steradian</td>
</tr>
<tr>
<td>Temperature</td>
<td>K</td>
<td>kelvin</td>
</tr>
</tbody>
</table>

**SI base and supplementary units**

- Length
- Mass
- Time
- Plane angle
- Solid angle
- Temperature
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.
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.

<table>
<thead>
<tr>
<th>Type</th>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(S)</td>
<td>ASSONi</td>
<td>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 ASSONi and ASSOCi should appear n times.</td>
</tr>
</tbody>
</table>
Science-Grade Data Products

ESO Science Data Products Standard

2.4 Processing Provenance

2.4.1 PROV keywords

<table>
<thead>
<tr>
<th>Type</th>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
</table>
| (S)  | PROVi   | 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. PROVi should appear as many times as needed to identify the complete set of science data files this product has been generated from. PROVi 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

- PROVi keywords must reside in the primary HDU of the FITS file;
- PROVi 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.
- 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 PROVi 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.

<table>
<thead>
<tr>
<th>PROVi</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROV1</td>
<td><code>VCAM.2010-03-27T08:58:58.083.fits</code> / Originating raw science file</td>
</tr>
<tr>
<td>PROV2</td>
<td><code>VCAM.2010-03-27T08:59:10.530.fits</code> / Originating raw science file</td>
</tr>
<tr>
<td>PROV3</td>
<td><code>VCAM.2010-03-27T08:59:24.541.fits</code> / Originating raw science file</td>
</tr>
<tr>
<td>PROV4</td>
<td><code>VCAM.2010-03-27T08:59:36.978.fits</code> / Originating raw science file</td>
</tr>
<tr>
<td>PROV5</td>
<td><code>VCAM.2010-03-27T08:59:51.043.fits</code> / Originating raw science file</td>
</tr>
</tbody>
</table>
## Science-Grade Data Products

### ESO Science Data Products Standard

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

<table>
<thead>
<tr>
<th>Type</th>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(S)</td>
<td>VOCLASS</td>
<td>The data model name and version: ‘SPECTRUM V1.0’.</td>
</tr>
<tr>
<td>(S)</td>
<td>VOPUB</td>
<td>The name of the publisher, namely ‘ESO/SAF’.</td>
</tr>
<tr>
<td>(S)</td>
<td>TITLE</td>
<td>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.</td>
</tr>
<tr>
<td>(R)</td>
<td>APERTURE</td>
<td>Aperture angular size, in degrees [deg]. It shall be set to the width of the slit or the diameter of the fiber.</td>
</tr>
<tr>
<td>(R)</td>
<td>TELAPSE</td>
<td>Total elapsed time in seconds [s], defined as MJD-END-MJD-OBS.</td>
</tr>
<tr>
<td>(R)</td>
<td>TMID</td>
<td>Exposure midpoint (MJD). It shall be set to (MJD-OBS+MJD-END)/2.0.</td>
</tr>
<tr>
<td>(R)</td>
<td>SPEC_Val</td>
<td>Characteristic spectral coordinate value in nanometers [nm]. Should WCS information be available, it can be calculated as CRVAL1+(0.5-CRPIX1+NAXIS1*0.5)<em>CDELT1. Otherwise, it can be calculated as NAXIS1</em>CDELT1. Otherwise, it is set to WAVE_MAX-WAVELMIN.</td>
</tr>
<tr>
<td>(R)</td>
<td>SPEC_BW</td>
<td>Width of the spectrum in nanometers [nm]. Should WCS information be available, it can be calculated as NAXIS1*CDELT1. Otherwise, it is set to WAVE_MAX-WAVELMIN.</td>
</tr>
</tbody>
</table>

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

#### Table 11: Keywords describing the BINTABLE columns in the 1D spectrum binary table format

<table>
<thead>
<tr>
<th>Type</th>
<th>Keyword</th>
<th>Description</th>
<th>Allowed values</th>
</tr>
</thead>
<tbody>
<tr>
<td>(I)</td>
<td>TFIELD</td>
<td>Number of fields (columns) in the binary table.</td>
<td></td>
</tr>
</tbody>
</table>

---

### Appendix A: Sybase IQ Reserved Words


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

---

Each field of the BINTABLE shall be further described in the extension header as specified in Table 11. Mandatory fields shall be WAVE, FLUX, and ERR, in that particular order. Additional fields may be added to the BINTABLE, provided that at least the values for their type, format, and unit are provided and described by the PI in the release description. Examples of such additional fields are given in Table 11.
## Project organization

### Scheduling, Validation and Acceptance

<table>
<thead>
<tr>
<th>Phase A (via PDR) to FDR</th>
<th>Documentation/Prototyping</th>
<th>2-3 years</th>
</tr>
</thead>
<tbody>
<tr>
<td>FDR to PAE</td>
<td>Development/Simulation</td>
<td>2-3 years</td>
</tr>
<tr>
<td>PAE to SV and SOP</td>
<td>Deployment</td>
<td>6-12 months</td>
</tr>
<tr>
<td>Operational Phase</td>
<td>Maintenance</td>
<td>10 years</td>
</tr>
</tbody>
</table>
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/representativety 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...... 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

On-Line Archive System

Instrument
Raw Data

Raw data

PIPELINE

Processed data

OFFLINE

Further Analysis

Archive

Shipping
Observatory Pipeline

Data arriving from the instrument

Pipeline workstation

Configuration files
Calibration database

Data Organizer

Instrument

Package

Quality control

PRODUCTS

Data Reduction System

Common Pipeline Library

Reduction Block
- List of the raw frames
- List of the calibration data
- Name of the products
- DRS recipe to apply

Reduction Block Scheduler
QC pipeline: QC logs

12:21:51> START GROUP / Start [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 RAW1 NAME = 'AMBER.2007-03-21T12:13:38.479.fits' / File name of raw frame [AMBER]
12:21:51> PRO REC1 CAL1 NAME = 'FlatFieldMap.fits' / File name 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

FOR1 trend analysis: ZEROPORTS
SR COL: Last date: 2004-01-27
Latest PSO data (2004-01-30) - SR COL

Sun Jan 31 14:28:07 CET 2004

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
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...