



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

AMBER Pipeline User Manual

VLT-MAN-ESO-19500-XXXX

Issue 0.8

Date 2006-09-18

Prepared: ESO AMBER Pipeline Team 2006-09-18
Name Date Signature

Approved: P.Ballester, M. Peron
Name Date Signature

Released: P. Quinn
Name Date Signature

This page was intentionally left blank

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	3 of 46

Change record

Issue/Rev.	Date	Section/Parag. affected	Reason/Initiation/Documents/Remarks
1.0	09/07/2004	All	First version

This page was intentionally left blank

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	5 of 46

Contents

1	Introduction	9
1.1	Purpose	9
1.2	Acknowledgements	9
1.3	Scope	9
1.4	Applicable documents	9
1.5	Reference documents	9
1.6	Abbreviations and acronyms	10
1.7	Glossary	10
2	Overview	11
3	AMBER Instrument Description	12
3.1	Overview	12
3.2	Observation Modes	13
3.3	Angular Resolution	13
3.4	Flux Sensitivity	14
3.5	Field Of View	15
4	Instrument Data Description	16
4.1	Calibration Frames	16
4.2	Scientific Frames	18
5	Ancillary Data Description	20
6	Data Reduction	21
6.1	Reduction Cascade	21
6.2	AMBER pipeline recipes	21
6.3	Set of frames	22
6.4	Recipe configuration	23
6.5	Running a recipe using EsoRex	23
6.6	Running a complete data reduction using GASGANO	25

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	6 of 46

6.7	Creating a primary FITS-Image of the Channels	25
6.8	Calculating the Pixel-To-Visibility-Matrix	27
6.9	Calculating the scientific results	28
6.10	Plotting the scientific results	29
7	Pipeline Recipes Interfaces	31
7.1	amber-raw-to-fitsimage	31
7.2	amber-detector	32
7.3	amber-beampos	32
7.4	amber-p2vm	33
7.5	amber-SciCal	34
7.6	amber-selector	34
7.7	amber-divider	36
8	Product Data Description	38
8.1	Intermediate Products	38
8.2	Scientific Products	38
9	Algorithms	39
9.1	General Algorithms	39
9.1.1	Dark subtraction	39
9.1.2	Flat field correction	39
9.1.3	fringe search	39
9.2	Recipes Algorithms	39
9.2.1	recreation of active CCD sensor area	39
A	Installation	40
A.1	Supported platforms	40
A.2	Building the AMBER pipeline	40
A.2.1	Requirements	40
A.2.2	Downloading the AMBER pipeline source distribution	41
A.2.3	Compiling and installing the AMBER pipeline	41

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	7 of 46

A.3	Configuring the pipeline recipe front-end applications	43
A.3.1	Setting up the EsoRex command-line tool	43
A.3.2	Setting up Gasgano	44
B	Troubleshooting Guide	45
C	Geometrical distortions models	46

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	8 of 46

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	9 of 46

1 Introduction

1.1 Purpose

The AMBER pipeline is a subsystem of the *VLTI Data Flow System* (DFS). Its target user is ESO *Data Flow Operations* (DFO) in the generation of master calibration data, in the reduction of scientific exposures and in the data quality control. It should also serve as a quick look tool for *Paranal Science Operations* (PSO). Additionally, the AMBER pipeline recipes are made public to the user community, to allow a more personalised processing of the data from the instrument.

This manual is a complete description of the data reduction recipes used by the the AMBER pipeline, reflecting the status of the AMBER pipeline as of June 30, 2006 (version 2.0.5).

1.2 Acknowledgements

The software package on which the AMBER pipeline is based was in large parts developed by the AMBER Consortium, and it is still the foundation of the current AMBER interferometry data reduction.

1.3 Scope

This document describes the AMBER pipeline used at ESO-Garching and ESO-Paranal for the purpose of data assessment and data quality control.

1.4 Applicable documents

- [2] VLT Data Flow System Specifications for Pipeline and Quality Control
VLT-SPE-ESO-19600-1233
- [3] Data Flow for VLT instruments Requirement Specification
VLT-SPE-ESO-19000-1618/2.0
- [4] DFS Pipeline & Quality Control – User Manual
VLT-MAN-ESO-19500-1619
- [5] ESO DICB – Data Interface Control Document
GEN-SPE-ESO-00000-0794
- [6] Common Pipeline Library User Manual
VLT-MAN-ESO-19500-2720
- [7] Gasgano User’s Manual
VLT-PRO-ESO-19000-1932
- [8] AMBER Calibration Plan
VLT-PLA-AMB-15830-0004

1.5 Reference documents

- [2] X.Yyyyyy and Z.Zzzzzz (1996) TODO, A&A Supp. v.117 p. 393-404

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	10 of 46

1.6 Abbreviations and acronyms

ADF	Aperture Definition File
ADM	Aperture Definition in mm
AMBER	Astronomical Multi BEam combineR
AT	Auxilliary Telescope
CPL	Common Pipeline Library
CRV	Spectral curvature model
DFS	Data Flow System
DMD	Data Management Division
DO	Data Organiser
DRS	Data Reduction System
ESO–MIDAS	ESO's Munich Image Data Analysis System
FITS	Flexible Image Transport System
HR	High Resolution
ICS	Instrument Control Software
IDS	Inverse Dispersion Solution
IRAF	Image Reduction and Analysis Facility
IWS	Instrument WorkStation
LR	Low Resolution
MMU	Mask Manufacturing Unit
MOS	Multi Object Spectroscopy
MR	Medium Resolution
P2VM	Pixel To Visibility Matrix
PAF	VLT PArAmeter File
PWS	Pipeline WorkStation
QC	Quality Control
RB	Reduction Block
RBS	Reduction Block Scheduler
SAO	Smithsonian Astrophysical Observatory
SNR	Signal to Noise Ratio
SOF	Set Of Frames
TCS	Telescope Control Software
UT	Unit Telescope
VLT	Very Large Telescope
VLI	Very Large Telescope Interferometer
WCS	World Coordinate System

1.7 Glossary

Optional section: Remove if necessary

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	11 of 46

2 Overview

The AMBER pipeline is a collection of data reduction recipes administrated by the higher level programs of the *VLT Data Flow System* (DFS) [1].

The *VLT Data Flow System* provides an environment where the definition and the execution of the data reduction steps can be made entirely automatic. The fundamental DFS software components making up the required functionality are the *Data Organiser* (DO) and the *Reduction Block Scheduler* (RBS). The DO applies some defined instrument-dependent rules for the classification of the incoming frames, for their association to suitable calibration data, and for launching the appropriate recipe for their standard reduction. The product of the DO is the *Reduction Block* (RB), an ASCII file containing all the necessary information for the specific data reduction. The RB is executed by the RBS, which calls the assigned pipeline recipe with the appropriate parameters and makes the data reduction products available to the *Data Handling Server* (DHS).

The AMBER pipeline recipes can also be used outside the pipeline context, as stand-alone programs. In this way they may be easily used by astronomers at their home institutes, or taken as building blocks for different data reduction strategies. However, the pipeline recipes, being designed to operate within the DFS environment, do not repeat the checks already performed by the DO. In other words, data classification and appropriate association are not proofed in any way within a recipe, being taken for granted. The *Gasgano* ([7]) data browser, that is capable of classifying the data frames in the same way as the DO does, greatly assist in the administration of a large quantity of different data frames.

The AMBER instrument and the different types of AMBER raw frames are briefly described in Sections 3 and 5, while the usage of the available reduction recipes is presented in Section 6.

More detailed descriptions of the data reduction algorithms used by the individual pipeline recipes can be found in Sections C and 9.

In Section ?? an overview of what's new on AMBER Pipeline release 2.0.7 is given.

In Appendix A the installation of the AMBER pipeline recipes is described.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	12 of 46

3 AMBER Instrument Description

AMBER has been developed under ESO contract by the AMBER Consortium, headed by the Laboratoire d'Astrophysique de Grenoble with contributions from Observatoire de la Cote d'Azur, Laboratoire Universitaire d'Astrophysique de Nice, MPI für Radioastronomie - Bonn and Osservatorio Astrofisico di Arcetri.

The instrument has been made available to the community and started operations in Paranal on October 1st, 2005.

In this chapter a brief description of the AMBER instrument is given. A more complete documentation can be found in the AMBER User Manual, downloadable from

<http://www.eso.org/instruments/amber/>

3.1 Overview

AMBER is the first-generation near-infrared three-way beam combiner. This instrument provides simultaneously spectrally dispersed visibility for three baselines and a closure phase at three different spectral resolutions. AMBER has been designed to investigate the milli-arcsec surrounding of astrophysical sources like young and evolved stars, active galactic nuclei and possibly detect exoplanet signal.

The main new feature of the instrument compared to other interferometric instruments is the simultaneous use of modal filters (optical fibres) and a dispersed fringe combiner using a spatial coding. The AMBER team has therefore investigated carefully a data processing strategy for this instrument and is providing a new type of data reduction method.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	13 of 46

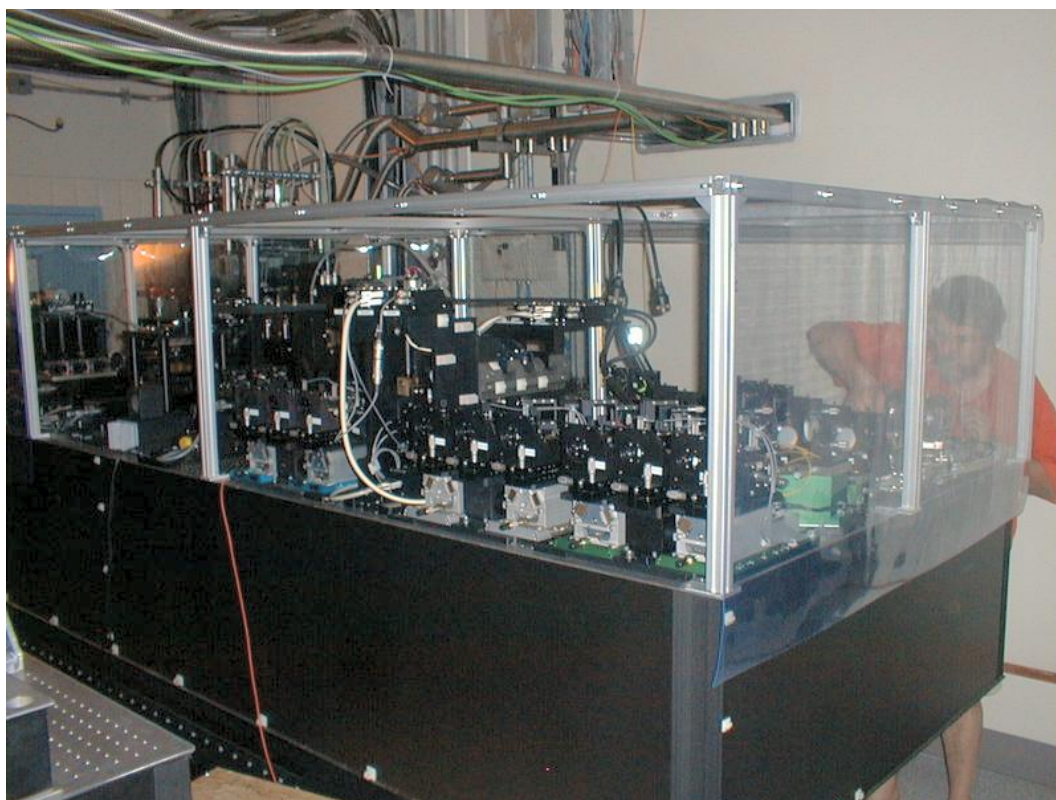


Figure 3.1.1: *The complex instrument AMBER consists of various optical and mechanical components to allow interferometric and spectroscopic observations at the same time.*

3.2 Observation Modes

AMBER offers three different modes of resolutions and three different bands in the following combinations:

One should note that the spectral coverage of the high spectral resolution does not cover the entire bands.

Depending on the type of science required, one can choose to observe with AMBER in classical way or in differential mode. The differential mode involves slight differences in the observing procedures but allows to obtain larger precision on the differential phase. The expected accuracy are typically 1% for the high sensitivity mode, 0.1% for the high precision mode, and 0.01% for the phase in the high precision mode in the differential configuration. This can be used either to obtain angular information on very unresolved objects or to aim at very high accuracy like in the extrasolar planets programs.

3.3 Angular Resolution

Angular resolution is set by the maximum available baseline, which is about 200 meters for the ATs and about 130 meters for the UTs. Accordingly, the limit will be about 2 milliarcseconds (mas) for the ATs, and about 3 mas for the UTs, in the K band. These values must be roughly halved for the J-band. Of course, the actual

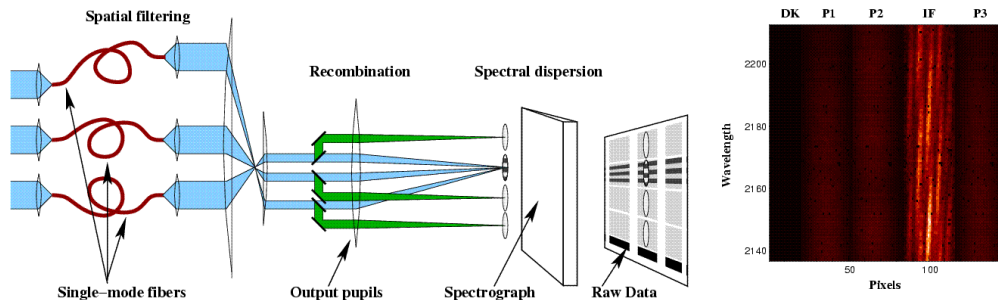


Figure 3.1.2: Left panel: Sketch of the AMBER instrument. The light enters the instrument from left. Right panel: Reconstituted image from AMBER raw data in three-telescopes mode. DK corresponds to a dark region of the detector. Pk are vertically dispersed spectra from each telescope and IF is the spectrally dispersed interferogram.

Resolution Mode	Abbreviation	Description
Low Resolution	JHK-LR	The J, H and K bands are covered simultaneously
Medium Resolution	JH-MR	The J and H bands are covered simultaneously
Medium Resolution	HK-MR	The H and K bands are covered simultaneously
High Resolution	J-HR	The J band is covered
High Resolution	H-HR	The H band is covered
High Resolution	K-HR	The K band is covered

Table 3.2.1: AMBER modes for observations.

resolution will depend also on the signal-to-noise ratio (SNR) available, which is in turn a function of the source brightness, as well as of many factors mentioned below. Under conditions of very high SNR, it is in principle feasible to obtain higher resolutions than the broad limit given above. This applies in particular to cases in which a source model is available and determinations of the visibility before the first zero are sufficient to constraint the size of the object (for instance, angular diameters and binary stars). Reversely, conditions of low SNR (for instance, for very faint sources), will limit the actual maximum resolution to values which can be substantially lower than the figures mentioned above. Another example is differential interferometry. If the source is not resolved, i.e. substantially smaller than the λ/B limit, then the differential phase still yields the displacement of the source photocenter with wavelength, which in many cases has been shown to be able to provide decisive spatial information. The accuracy on the photocenter displacement is typically the λ/B resolution divided by the fringe SNR. For example in the case of AGN BLRs (proposal 4) it has been shown that in 20 minutes of actual observations on K = 12 object (with tracked fringes) we can achieve a 10 mas accuracy on the photocenter displacement.

3.4 Flux Sensitivity

Concerning the limits in sensitivity, these depend on a large number of factors: observing modes that correspond to the elementary exposure time (high sensitivity : 50ms, high precision : 10ms and long exposures : any

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	15 of 46

but with fringe tracking), the type of telescopes, the spectral resolution, the seeing, etc. Some of these are relatively well identified by the design of AMBER (for instance, transmission efficiency, fiber coupling, detector characteristics etc.), while many others are more difficult to characterize at this time: for instance the quality of fringe tracking and adaptive optics correction.

3.5 Field Of View

AMBER is a single-mode instrument, therefore theoretically the field of view is limited to the Airy disk of each individual aperture, i.e. 250 mas for the ATs in K and 60 mas for the UTs in K.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	16 of 46

4 Instrument Data Description

AMBER data can be separated into *raw* frames and *product* frames. Raw frames are the unprocessed output of the AMBER instrument observations, while product frames are either the result of the AMBER pipeline processing (as reduced frames, master calibration frames, etc.), or are the result of the calculation of the Pixel-To-Visibility-Matrix (P2VM).

Any raw or product frame can be classified on the basis of a set of keywords read from its header. Data classification is typically carried out by the DO or by *Gasgano* [7], that applies the same set of classification rules. The association of a raw frame with calibration data (*e.g.*, of a science frame with a master flat field frame) can be obtained by matching the values of a different set of header keywords.

Each kind of raw frame is typically associated to a single AMBER pipeline recipe, *i.e.*, the recipe assigned to the reduction of that specific frame type. In the pipeline environment this recipe would be launched automatically.

A product frame may be input to more than one AMBER pipeline recipe, but it may be created by just one pipeline recipe (with the same exceptions mentioned above). In the automatic pipeline environment a product data frame alone wouldn't trigger the launch of any recipe.

In the following all raw and product AMBER data frames are listed, together with the keywords used for their classification and correct association. The indicated *DO category* is a label assigned to any data type after it has been classified, which is then used to identify the frames listed in the *Set of Frames* (see Section 6.3, page 22).

Raw frames can be distinguished in *calibration* frames and *science* frames. Their intended use is implicitly defined by the assigned recipe.

4.1 Calibration Frames

These are frames that are used for instrument calibration.

- **Detector Calibration and Flat-Field:**

These raw frames are for the creation of a master flat field and a bad pixel map.

DO category: AMBER_DETECTOR_DARK

Processed by: `amber_detector`

Classification keywords:	Association keywords:	Note:
DPR CATG = CALIB	-	
DPR TYPE = DETDARK	-	
DPR TECH = -		

DO category: AMBER_DETECTOR_FLAT

Processed by: `amber_detector`

Classification keywords:	Association keywords:	Note:
DPR CATG = CALIB	-	
DPR TYPE = DETFLAT	-	
DPR TECH = -		

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	17 of 46

- **Pixel To Visibility Matrix and Spectral Calibration:**

These frames are for calculation of a Pixel-To-Visibility-Matrix.

Optionally the 2WAVE and 3WAVE files can be sent to the recipe. In this case the the recipe will also perform a new spectral calibration.

DO category: AMBER_2P2V

Processed by: amber_p2vm

Classification keywords:

DPR CATG = CALIB

DPR TYPE = 2P2V

DPR TECH = -

Association keywords:

-

DET NTEL

Note:

Number of Telescopes used

DO category: AMBER_2WAVE

Processed by: amber_p2vm

Classification keywords:

DPR CATG = CALIB

DPR TYPE = WAVE, 2TEL

DPR TECH = -

Association keywords:

DET NTEL

Note:

Number of Telescopes used

DO category: AMBER_3P2V

Processed by: amber_p2vm

Classification keywords:

DPR CATG = CALIB

DPR TYPE = 3P2V

DPR TECH = -

Association keywords:

-

DET NTEL

Note:

Number of Telescopes used

DO category: AMBER_3WAVE

Processed by: amber_p2vm

Classification keywords:

DPR CATG = CALIB

DPR TYPE = WAVE, 3TEL

DPR TECH = -

Association keywords:

DET NTEL

Note:

Number of Telescopes used

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	18 of 46

4.2 Scientific Frames

- **Scientific Observation:**

These frames are for scientific observations. For the calculation of visibilities there is at least one science object frame needed. In addition there must be a valid P2VM as well as a Flat-Field and a Bad-Pixel-Map.

For improved SNR it is useful to input the optional frames for sky and dark detector illumination.

DO category: AMBER_SCIENCE

Processed by: amber_SciCal

Classification keywords:

DPR CATG = SCIENCE

DPR TYPE = OBJECT

DPR TECH = -

Association keywords:

DET NTEL

Note:

Number of Telescopes used

DO category: AMBER_SCIENCE_DARK

Processed by: amber_SciCal

Classification keywords:

DPR CATG = SCIENCE

DPR TYPE = DARK

DPR TECH = -

Association keywords:

DET NTEL

Note:

Number of Telescopes used

DO category: AMBER_SCIENCE_SKY

Processed by: amber_SciCal

Classification keywords:

DPR CATG = SCIENCE

DPR TYPE = SKY

DPR TECH = -

Association keywords:

DET NTEL

Note:

Number of Telescopes used

DO category: AMBER_CALIB

Processed by: amber_SciCal

Classification keywords:

DPR CATG = CALIB

DPR TYPE = OBJECT

DPR TECH = -

Association keywords:

DET NTEL

Note:

Number of Telescopes used

DO category: AMBER_SCIENCE_CALIB

Processed by: amber_SciCal

Classification keywords:

DPR CATG = CALIB

DPR TYPE = STD

DPR TECH = -

Association keywords:

DET NTEL

Note:

Number of Telescopes used

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	19 of 46

DO category: AMBER_P2VM
 Processed by: amber_SciCal

Classification keywords:

DPR CATG = CALIB
 DPR TYPE = P2VM
 DPR TECH = -

Association keywords:

DET NTEL

Note:

Number of Telescopes used

DO category: AMBER_BADPIX
 Processed by: amber_SciCal

Classification keywords:

DPR CATG = CALIB
 DPR TYPE = BADPIX
 DPR TECH = -

Association keywords:

DET NTEL

Note:

Number of Telescopes used

DO category: AMBER_FLATFIELD
 Processed by: amber_SciCal

Classification keywords:

DPR CATG = CALIB
 DPR TYPE = FLATFIELD
 DPR TECH = -

Association keywords:

DET NTEL

Note:

Number of Telescopes used

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	20 of 46

5 Ancillary Data Description

Currently there is no static calibration data.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	21 of 46

6 Data Reduction

In this section the usage of the AMBER pipeline recipes is described. In particular, typical data reduction sessions for each instrument operating mode are presented in Sections ?? and ??.

6.1 Reduction Cascade

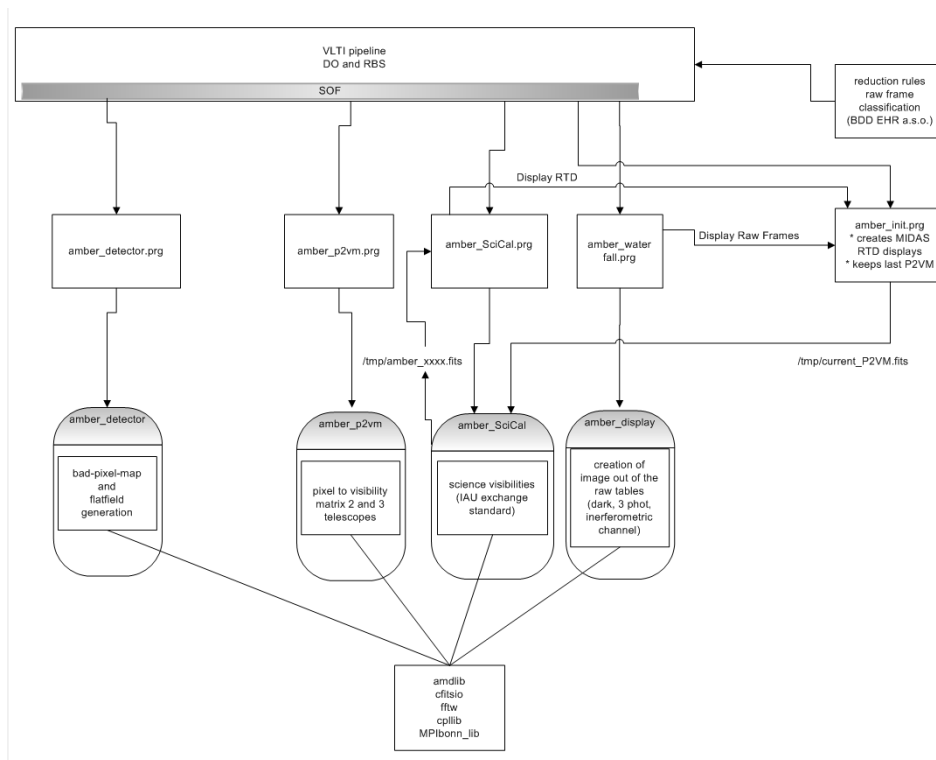


Figure 6.1.1: <AMBER Association Map

6.2 AMBER pipeline recipes

The current AMBER pipeline is based on a set of stand-alone recipes, assigned to different fundamental operations:

Creation of general calibration data or informational output:

amber_raw_to_fitsimage: The AMBER raw frames are containing tables of numbers only. This recipe is re-creating an image of what the detector has recorded.

amber_detector: This recipe creates the master flat field and the bad pixel map for the optical train and detector of the instrument.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	22 of 46

amber_p2vm: Calculation of the Pixel-To-Visibility-Matrix to be able to disentangle the interferometric channels for 2 and 3 telescopes. Optionally it will also perform a spectral calibration.

amber_SciCal: This recipe calculates the visibilities along with additional information like the SNR, error bars and the closure phase for the given band.

amber_selector: This recipe can select or filter the visibilities by various algorithm such as Fringe Contrast SNR

In the next sections a general description on the use of recipes is given, together with more detailed information on the individual recipes.

First there will be a description of how to use the recipes with EsoRex and later how to use them with GASGANO.

6.3 Set of frames

Each pipeline recipe is run on a set of input FITS data files. The filenames are listed together with their DO category in an ASCII file, the *Set of Frames* (SOF), that is required when launching a recipe. SOF files containing the frames selected by the user are automatically created by *Gasgano* [7].

Here is an example of SOF, valid for the *amber_SciCal* recipe:

```

AMBER.2003-02-04T16:31:53.012.fits AMBER_FLATFIELD
AMBER.2003-02-04T16:31:54.014.fits AMBER_BADPIX
AMBER.2003-02-04T16:31:55.038.fits AMBER_P2VM
AMBER.2003-02-04T16:31:56.040.fits AMBER_SCIENCE_DARK
AMBER.2003-02-04T16:31:57.045.fits AMBER_SCIENCE
AMBER.2003-02-04T16:31:57.050.fits AMBER_SCIENCE_SKY

```

The pipeline recipe will access the listed files when required by the reduction algorithm.

Note that the AMBER pipeline recipes do not proof in any way the correctness of the classification tags specified in the SOF. In the above example, the recipe *amber_SciCal* will treat the frame `AMBER.2003-02-04T16:31:53.012` as a `AMBER_FLATFIELD`, the frame `AMBER.2003-02-04T16:31:54.014.fits AMBER_BADPIX` as a `AMBER_BADPIX`, etc., even when they do not contain this type of data.

However the recipe will check all frames if the optical train is the same like it has been during recording the P2VM files. It will also check if all the contained regions are consistent with each other. Otherwise the usage of a particular P2VM file will result in useless data.

The reason of this lack of control is that the AMBER recipes are just the DRS component of the complete pipeline running on Paranal, where the task of data classification and association is carried out by the Data Organiser. On the other side, using *Gasgano* [7] as an interface to the pipeline recipes will permit to classify the data frames exactly as the DO does.

A recipe handling an incorrect SOF may stop with not immediately understandable error messages at best. In the worst cases, the recipe would apparently run without any problem, producing results that may look reasonable while they are instead flawed.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	23 of 46

6.4 Recipe configuration

Each pipeline recipe may be assigned a configuration file, containing the default values of the parameters related to that recipe. The possibility to create recipe configuration files, and the way information is conventionally stored in such files, depends on the front-end application used for launching the pipeline recipes¹. In this section the recipe configuration files created and handled by *EsoRex* are described. Please refer to the Gasgano User's Manual [7] for information about saving recipe configuration parameters.

An *EsoRex* recipe configuration file can be generated in the directory `$HOME/.esorex` by *EsoRex* (see Section 6.5, on page 23) at installation time, and has the same name as the recipe to which it is related, with the filename extension `.rc`. For instance, the recipe *amber_SciCal* has its *EsoRex* generated configuration file named `amber_SciCal.rc`.

Without a recipe configuration file the recipe will use the default values.

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

```
# --Binning
# Binning of frames, default 1
amber.Parameters.binning=10
```

In this example, the parameter `amber.Parameters.binning` is set to the value `10`. 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 (see again Section 6.5 on page 23).

The hierarchy introduced in the parameter names has currently no effect. Although it is not used at the moment, it is envisaged that this feature will be used in future releases to avoid potential name clashes. The shorter parameter aliases are made available for use on the command line.

The parameter names belonging to the recipe specific configuration files are described in the corresponding recipe sections.

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/.esorex`, or having a name different from the recipe name, should be explicitly specified when launching a recipe (see next section).

6.5 Running a recipe using EsoRex

The AMBER pipeline recipes are implemented as plugins, used by front-end applications like *EsoRex* or *Gasgano* [7] to execute the recipes.

The file organiser *Gasgano* [7] provides an intuitive GUI for launching the recipes, applying the same rules used by the DO for the available data frames classification, whereas *EsoRex* is the most basic way for executing the pipeline recipes from the shell prompt, that shall be described here.

¹The plugin concept allows the implementation of different front-end applications with specific functionalities. For a complete description on how to create a pipeline recipe launcher, please refer to the CPL User Manual [6].

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	24 of 46

The basic format for using *EsoRex* is as follows:

```
esorex [esorex_options] recipe_name [recipe_options] set_of_frames
```

A list of all the available recipes, each with a one-line description, can be obtained using the command:

```
esorex --recipes
```

To get help for an individual recipe (in the subsequent examples, *vmbias* is used), the following is used:

```
esorex --help amber_SciCal
```

In order to display the current parameters setting of a recipe, either the recipe configuration file located under `$HOME/.esorex` may be viewed (if present), or the following command may be used:

```
esorex --params amber_SciCal
```

If the default recipe configuration file is not found, or a particular value is not configured within this file, then the system defaults will be shown and used.

A recipe can be run by specifying its name to *EsoRex*, together with the name of a SOF (see Section 6.3, page 22). For instance, the following command line would be used to run the recipe *amber_SciCal* for processing the files specified in the SOF `amber_SciCal.sof`:

```
esorex amber\_SciCal amber_SciCal.sof
```

A recipe configuration file different from the default one (see Section 6.4, page 23) can also be specified on the command line:

```
esorex --recipe-config=my_alternative_config.rc amber_SciCal amber_SciCal.sof
```

The recipe parameters can be modified either by editing directly the used configuration file, 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. For instance, to set the *amber_SciCal* recipe `binning` parameter to 10, the following should be typed:

```
esorex amber_SciCal --binning=10 amber_SciCal.sof
```

There are also parameters used to configure the *EsoRex* launcher, that may be listed in an `esorex.rc` configuration file located under `$HOME/.esorex`. On the command line, the *EsoRex* options must be inserted before, and not after, the specified recipe name. The *EsoRex* options are those that are recipe independent, as for instance the verbosity level, the directory where the recipe products should be written, or the permission to overwrite old products with new ones.

For more information on *EsoRex*, see <http://www.eso.org/cpl/esorex.html>.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	25 of 46

6.6 Running a complete data reduction using GASGANO

For AMBER we need a two step data reduction. First we must create a P2VM and with it we can reduce the scientific data taken with the same optical train.

6.7 Creating a primary FITS-Image of the Channels

This recipe can display one AMBER raw frame at a time. Select any of the following frames:

TO DO: Table of classifications.

By right clicking and choosing *amber_raw_to_fitsimage* one can run the recipe.

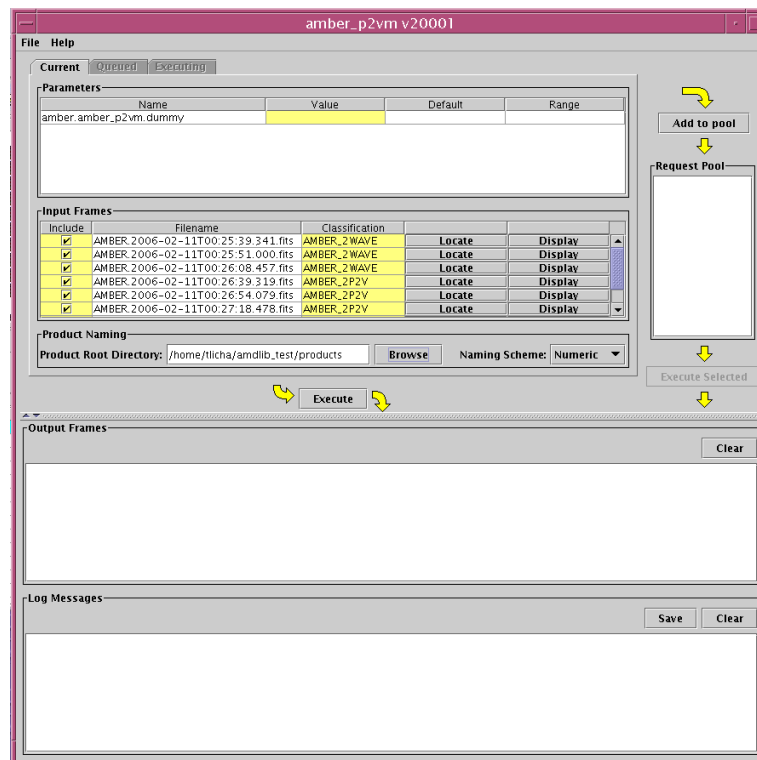


Figure 6.7.1: The recipe *amber_raw_to_fitsimage* when launched in GASGANO.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	26 of 46

The recipe will store the FITS-image at the location `/tmp/amber_raw_to_fitsimage.fits`. It can be displayed by any capable FITS viewer, i.e. the *FV*.

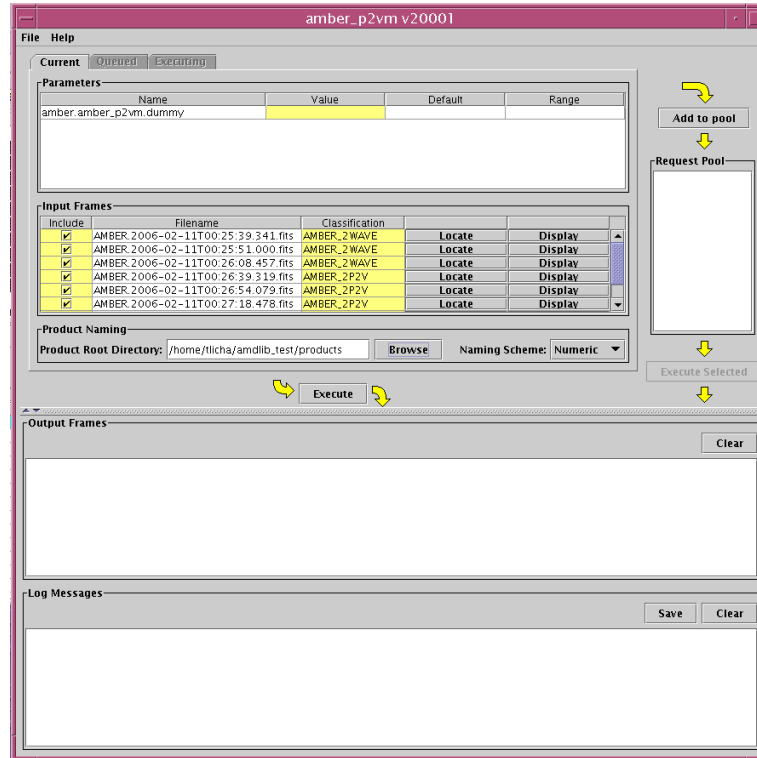


Figure 6.7.2: Example FITS image.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	27 of 46

6.8 Calculating the Pixel-To-Visibility-Matrix

TODO: Description, selecting proper frame-types, mandatory and optional frames.

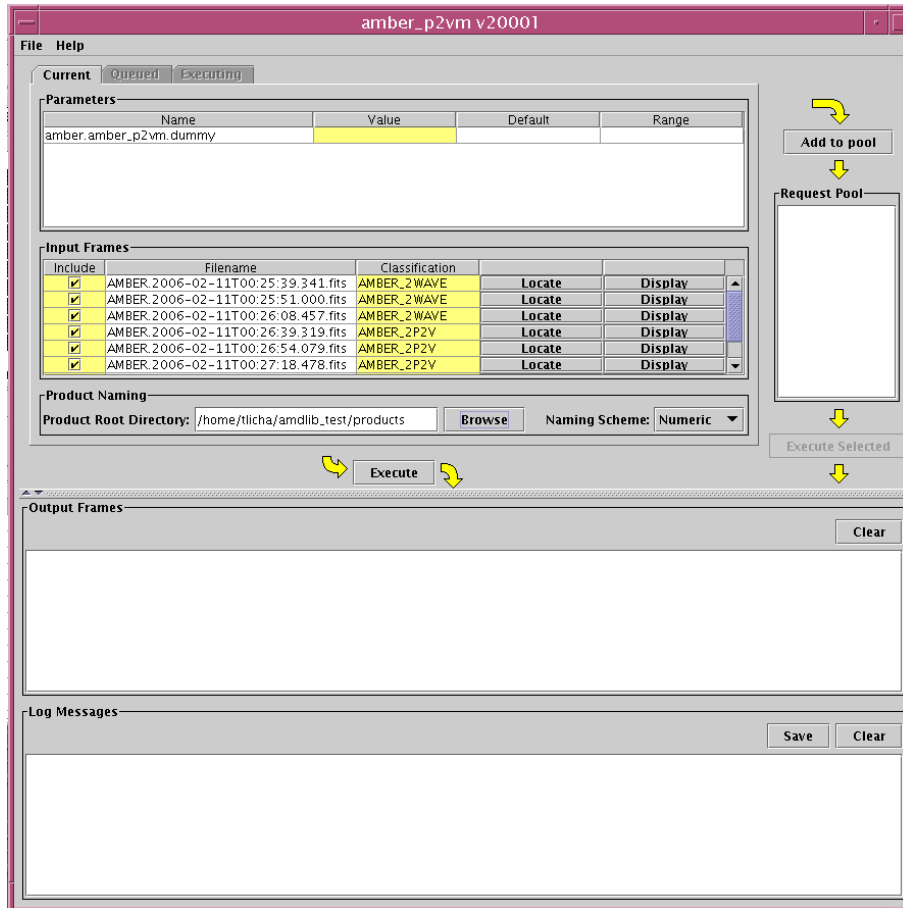


Figure 6.8.1: The recipe `amber_p2vm` when launched in GASGANO.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	28 of 46

6.9 Calculating the scientific results

TODO: Description, selecting proper frame-types, mandatory and optional frames.

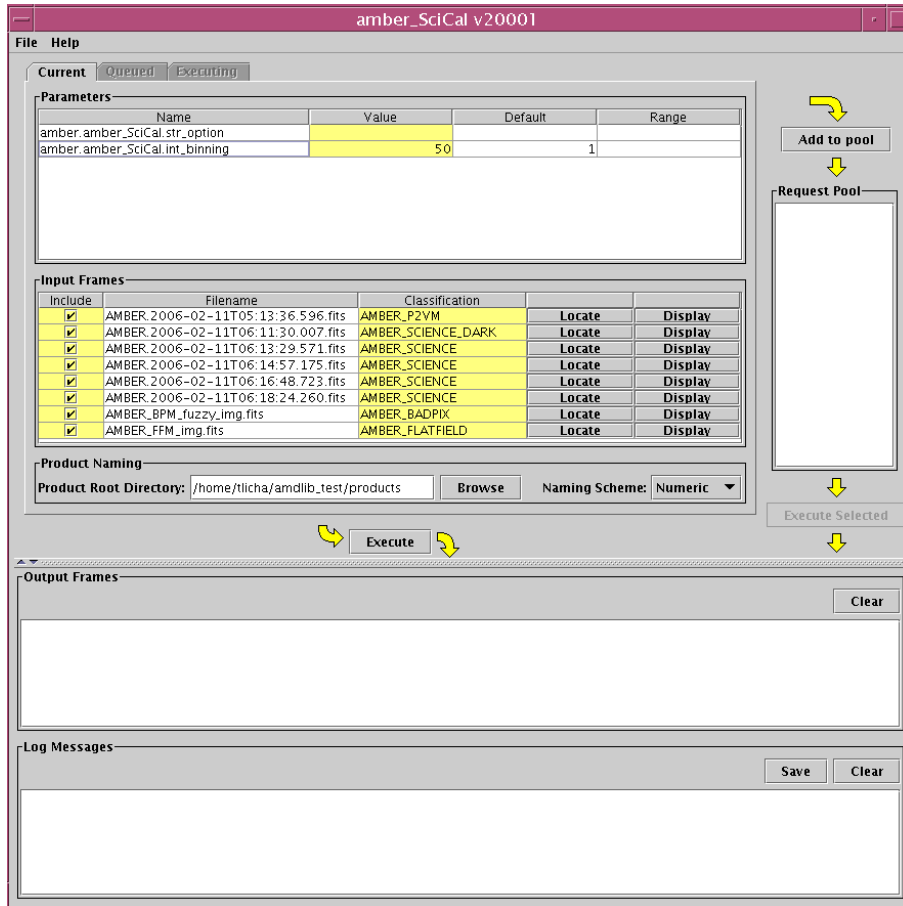


Figure 6.9.1: The recipe *amber_SciCal* when launched in GASGANO.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	29 of 46

6.10 Plotting the scientific results

It is possible to plot the scientific results from previously stored products. GASGANO will group the product at the end of the raw files of the observation so that they can easily be identified. The particular product must be selected and right clicked. In the menu "Run..." must be chosen. Selecting 'amber_gasgano_plot.sh' will start the plotting.

Please make sure that MIDAS is installed and that the `amber_gasgano_plot.prg` resides in the same path as the script does.

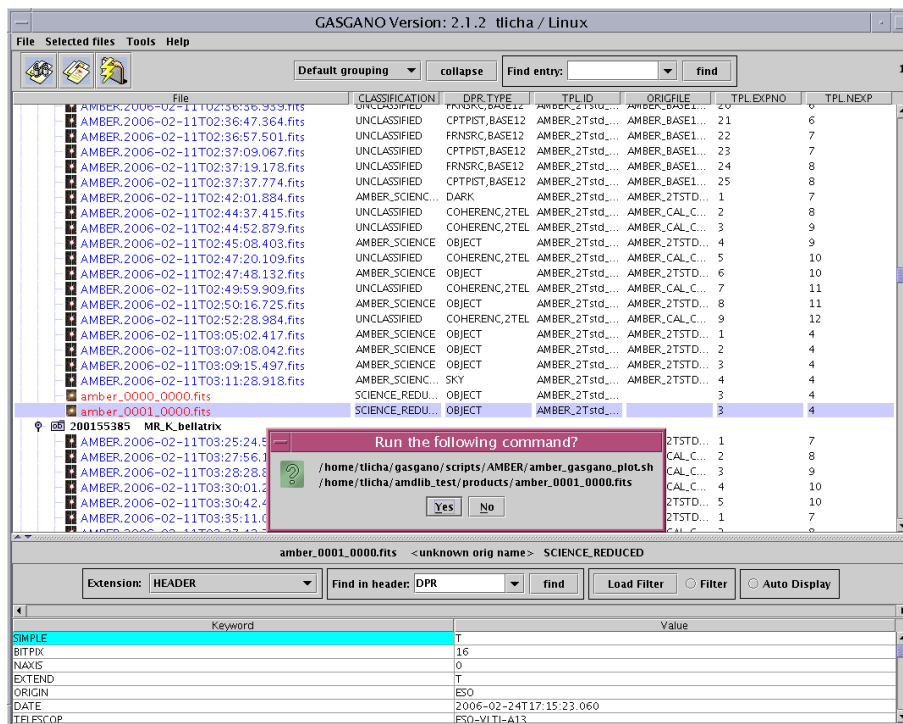


Figure 6.10.1: Starting the plotting script from GASGANO

To finish the plot and return to GASGANO one has to type `bye` in the window named `inmidas`. After the plot has completed on screen there is a corresponding post script file available named `/tmp/amber_0000.ps`. However this will be overwritten by the next plot, so one has to copy it to a safe place for further usage.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	30 of 46

The plots are different for the 2 and 3 telescopes case, of course. In the 3 telescopes case there are 3 baselines while in the 2 telescope case there is only one.

In the upper left of the plot there is the statistics of how many frames the observation contained and how many of them have been reduced giving a SNR greater than 2.

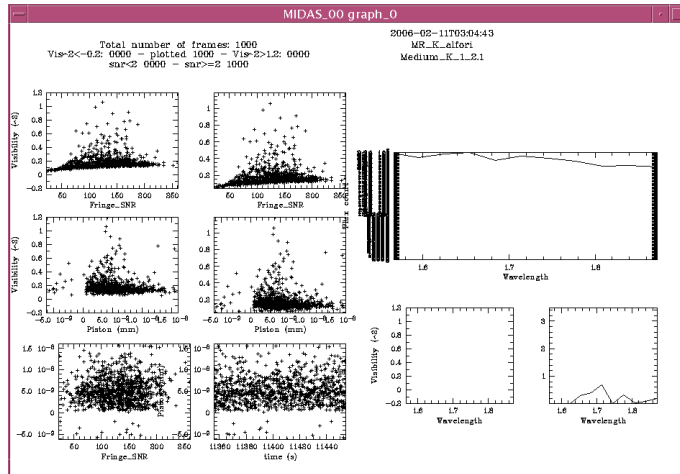


Figure 6.10.2: Example plot for the 2-telescopes mode. Note that only one baseline is present.

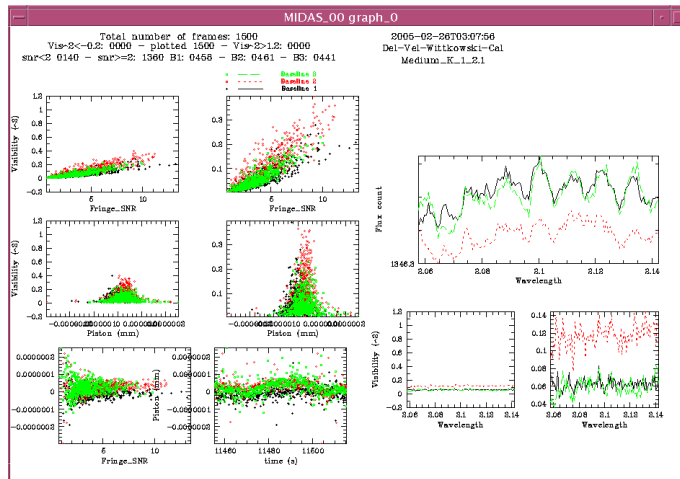


Figure 6.10.3: Example plot for the 3-telescopes mode. Note that three baselines are present and displayed in colours.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	31 of 46

7 Pipeline Recipes Interfaces

7.1 amber-raw-to-fitsimage

The AMBER pipeline recipe *amber_raw_to_fitsimage* is used to create a primary image from AMBER rawfiles which do contain tables of numbers only. This image can be regarded as a recreation of the active area of the CCD sensor. It is useful to check if the interferometric channel is actually showing fringes and if flux is present in all photometric channels.

It can handle raw files taken in 2 and 3 telescope mode. The image shows the vertically dispersed channels of the AMBER instrument as follows:

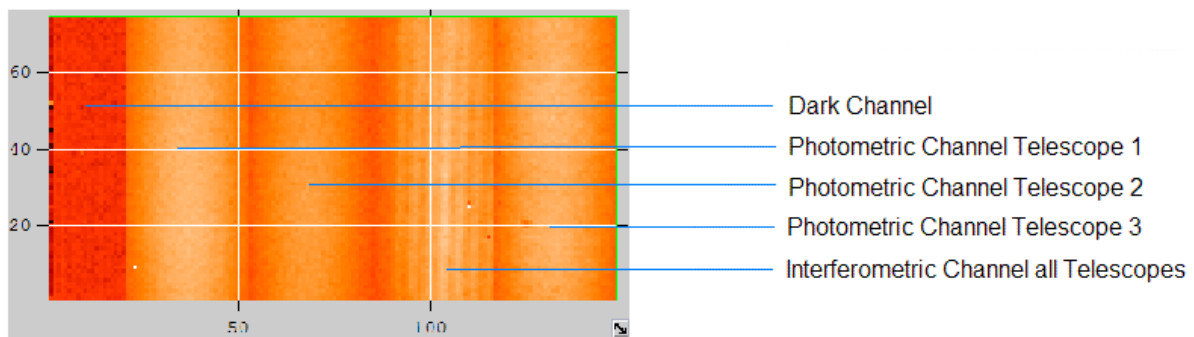


Figure 7.1.1: Example image for the 3-telescopes mode.

At least one of the files listed below must be included in the SOF. If there are more files only the first one will be displayed.

DO category	Type	Explanation	Required
AMBER_2P2V	Raw frame	2 telescopes P2VM	
AMBER_2WAVE	Raw frame	2 telescopes P2VM, spectral calibration	
AMBER_3P2V	Raw frame	3 telescopes P2VM	
AMBER_3WAVE	Raw frame	3 telescopes P2VM, spectral calibration	
AMBER_SCIENCE_DARK	Raw frame	dark frame from scientific observation	
AMBER_SCIENCE_SKY	Raw frame	sky frame from scientific observation	
AMBER_SCIENCE	Raw frame	science object frame from scientific observation	
AMBER_CALIB	Raw frame	calibrator object frame from scientific observation	

Table 7.1.1: Input files for the *amber_raw_to_fitsimage* recipe.

Any of the AMBER raw frames can be displayed with that recipe.

This recipe is only creating an intermediate product, namely a FITS file containing the primary image of the reconstructed image of the active regions of the detector. In addition the header is completed with a copy of the raw frame's header to describe the frame. A copy of the product will always be stored at the location `/tmp/amber_raw_to_fitsimage.fits` with a fixed name to make it easier to write scripts for automation purposes.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	32 of 46

There are no *amber_raw_to_fitsimage* parameters.

A description of the algorithms used in this recipe is given in Section 9.2.1, page 39.

7.2 amber-detector

The AMBER pipeline recipe *amber_detector* is used to create the bad pixel map and the master flat field for the AMBER instrument.

(Recipe currently under development)

DO category	Type	Explanation	Required
AMBER_DETECTOR_DARK	Raw frame	dark exposure	✓
AMBER_DETECTOR_FLAT	Raw frame	flat exposure	✓

Table 7.2.1: *Input files for the amber_detector recipe.*

7.3 amber-beampos

The AMBER pipeline recipe *amber_beampos* is used to monitor the beam alignment for the AMBER instrument. A technical template will create the 22 necessary files.

DO category	Type	Explanation	Required
AMBER_BEAMPOS	Raw frame	exposures with various optics settings	✓

Table 7.3.1: *Input files for the amber_detector recipe.*

From these files the recipe will derive an image of the detector as well as QC parameters and write them to a product as well as to the QC log. For each of the frames the following values are written:

QC Parameter	Explanation
X	the x position of the beam centroid
Y	the y position of the beam centroid
SIZX	the size in x direction of the beam
SIZY	the size in y direction of the beam
FLUX	the flux count of the beam

Table 7.3.2: *QC values written by amber_beampos recipe.*

Depending on the type of the frame there will be a prefix written to identify the specific optics used.

Example for a complete QC parameter including the prefix:

HIERARCH.ESO.QC.CLDSTP.SIZX=23.45

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	33 of 46

QC Prefix	Explanation
CLDSTP	cold stop
A.K	A in K band
B.K	B in K band
C.K	C in K band
A.H	A in H band
B.H	B in H band
C.H	C in H band
A.J	A in J band
B.J	B in J band
C.J	C in J band
BCD.A.K	BCD and A in K band
BCD.B.K	BCD and B in K band
BCD.C.K	BCD and C in K band
BCD.A.H	BCD and A in H band
BCD.B.H	BCD and B in H band
BCD.C.H	BCD and C in H band
BCD.A.J	BCD and A in J band
BCD.B.J	BCD and B in J band
BCD.C.J	BCD and C in J band
OUTF.A	OUTF and A
OUTF.B	OUTF and B
OUTF.C	OUTF and C

Table 7.3.3: *QC value prefixes written by amber_beampos recipe.*

File name	DO category	Type	Explanation
amber_beampos_xxxx.fits	BEAMPOS_REDUCED	FITS	FITS file with primary image of the beam and the QC p

Table 7.3.4: *Product of the amber_beampos recipe.*

7.4 amber-p2vm

The AMBER pipeline recipe *amber_p2vm* is used to create a pixel to visibility matrix (P2VM). This P2VM is containing the translation matrix for the interferometric channel to be able to calculate visibilities from scientific observations .

This recipe is able to handle 2- and 3-telescope P2VMs. The DET NTEL keyword from the first rawfile's header will determine the mode.

The raw files are actually containing all possible variations of the shutter positions of the photometric channels and the response of the interferometric channel for the standard lamps used during that process. Because in 3-telescope mode there is one more photometric channel compared

The only product of the *amber_p2vm* recipe is the P2VM file, as shown in Table 7.4.2.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	34 of 46

DO category	Type	Explanation	Required
AMBER_BADPIX	Calibration frame	Bad Pixel Map of the detector	✓
AMBER_FLATFIELD	Calibration frame	Flat Field of the detector and the instrument's optical train	✓
AMBER_2P2V	Raw frame	2 telescopes P2VM, 5 frames are needed	✓
AMBER_2WAVE	Raw frame	2 telescopes P2VM, spectral calibration	
AMBER_3P2V	Raw frame	3 telescopes P2VM, 10 frames are needed	✓
AMBER_3WAVE	Raw frame	3 telescopes P2VM, spectral calibration	

Table 7.4.1: *Input files for the amber_p2vm recipe.*

File name	DO category	Type	Explanation
amber_p2vm.fits	P2VM_REDUCED	FITS	Pixel to Visibility Matrix

Table 7.4.2: *Product of the amber_p2vm recipe.*

There are no *amber_p2vm* parameters.

A description of the algorithms used in this recipe is given in Section ??, page ??.

7.5 amber-SciCal

The AMBER pipeline recipe *amber_SciCal* is used to calculate visibilities from interferometric observations.

This recipe is able to handle 2- and 3-telescope observations. The DET NTEL keyword from the first rawfile's header will determine the mode. It can also handle all of the different bands in any resolution. If there is more than one band present in the raw frame the recipe will split the bands and create a separate product for the J, H and K band.

All the files that must be included in the input SOF are listed in Table 7.4.1. In addition to the frames marked as required there must be at least one frame of the DO categories AMBER_SCIENCE, AMBER_CALIB or AMBER_SCIENCE_CALIB. For each of them corresponding product will be created.

The only product of the *amber_SciCal* recipe is a OI file, as shown in Table 7.5.2.

The *amber_SciCal* parameters are listed in Table 7.5.3.

A description of the algorithms used in this recipe is given in Section ??, page ??.

7.6 amber-selector

The AMBER pipeline recipe *amber_selector* is used to select matching frames from AMBER products created by *amber_SciCal*. The output is again an AMBER product (same OI FITS format like the input) but will only contain the matching frames which passed the filter. This means the resulting product can be plotted by the same tools like all other AMBER OI FITS files.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	35 of 46

DO category	Type	Explanation	Required
AMBER_BADPIX	Calibration frame	Bad Pixel Map of the detector	✓
AMBER_FLATFIELD	Calibration frame	Flat Field of the detector	✓
AMBER_P2VM	Calibration frame	Pixel-To-Visibility-Matrix of the instrument's optical train	✓
AMBER_SCIENCE_DARK	Raw frame	dark frame from scientific observation	
AMBER_SCIENCE_SKY	Raw frame	sky frame from scientific observation	
AMBER_SCIENCE	Raw frame	science object frame from scientific observation	✓
AMBER_CALIB	Raw frame	calibrator object frame from scientific observation	
AMBER_SCIENCE_CALIB	Raw frame	standard calibrator object frame	

Table 7.5.1: Input files for the *amber_SciCal* recipe.

File name	DO category	Type	Explanation
amber_XXXX.fits	SCIENCE_REDUCED	FITS	OI FITS file, with XXXX corresponding to the file's number

Table 7.5.2: Product of the *amber_SciCal* recipe.

There is various different algorithms to select and filter data. The algorithms can be selected and controlled by the parameters *selection_method* and *selection_x*.

selection_method First x Frames

This method will filter the first number of x frames of the observation, in other words the earliest frames. The purpose of this method is a more technical one. It can be used to check the if the instrument is properly suppressing the very first few frames after switching the detector to on. These frames usually contain garbage due to the design of the redout electronics of the detector.

selection_method Fringe SNR > x

This method will filter all the frames where the Signal-To-Noise-Ratio of the fringes is greater than the value of x. Useful values of x could be 2 and above.

selection_method Fringe SNR percentage x

This method will filter the frames where the Signal-To-Noise-Ratio of the fringes is highest and leaving x percent of frames. I.e. example if x=20 this method will filter the 20 percent of frames with the highest Fringe SNR and drop the other 80 percent.

selection_method Flux > x

This method will filter all the frames where the flux of (???) is greater than the value of x. Useful values of x could be (???) and above.

selection_method Flux percentage x

This method will filter the frames where the flux of (???) is highest and leaving x percent of frames. I.e. example if x=20 this method will filter the 20 percent of frames with the highest flux and drop the other 80 percent.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	36 of 46

Parameter	Possible values	Explanation
binning	0 to 9999 (maximum number of frames per observation)	the number of frames to be averaged for each of the visibility calculations

Table 7.5.3: *amber_SciCal* parameters.

DO category	Type	Explanation	Required
SCIENCE_REDUCED	product frame	OI FITS file	✓
CALIB_REDUCED	product frame	OI FITS file	✓

Table 7.6.1: *Input files for the amber_selector recipe.*

selection_method Exclude Frames by ASCII File

This method is an interface to any external tools which are able to identify frames to be excluded. Simply store a file called */tmp/amber_exclude.txt* containing the frame index. There shall be only one frame index per line.

If you want to exclude frame number 13, 20, 25 and 255 the file should look like

13
20
25
255

Table 7.6.2: *example of /tmp/amber_exclude.txt.*

As a result the output OI FITS file will contain all frames but not the four numbered above.

selection_method Include Frames by ASCII File

This method is the same like the exclude described above. It just works the other way round. It will leave the frames with the given index numbers and drop all the rest. The file must be called */tmp/amber_include.txt*

7.7 **amber-divider**

The AMBER pipeline recipe *amber_divider* is used to create calibrated visibilities from AMBER two products created by *amber_SciCal*. One product must be from calibrator and the other from a science object. The output is currently undefined.

(Recipe currently under development)

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	37 of 46

DO category	Type	Explanation	Required
SCIENCE_REDUCED	product frame	OI FITS file	✓
CALIB_REDUCED	product frame	OI FITS file	✓

Table 7.7.1: *Input files for the amber_selector recipe.*

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	38 of 46

8 Product Data Description

Products can be distinguished in *intermediate product* and *scientific product*. These products are classified by the DO of by *Gasgano* [7] according to their own DO category (keyword `PRO CATG`), assigned to them at creation time. For this reason no classification keyword is listed in this section. The name of the recipe (or recipes) used to create a given product is given here.

8.1 Intermediate Products

- **Flat Field:**

DO category: `AMBER_FLATFIELD`

Created by: `amber_detector`

Association keywords:

`DET NTEL`

Note:

Number of Telescopes used

- **Bad Pixel Map:**

DO category: `AMBER_BADPIX`

Created by: `amber_detector`

Association keywords:

`DET NTEL`

Note:

Number of Telescopes used

- **Pixel-To-Visibility-Matrix P2VM:**

DO category: `P2VM_REDUCED`

Created by: `amber_p2vm`

Association keywords:

`DET NTEL`

Note:

Number of Telescopes used

8.2 Scientific Products

- **OI Product:**

DO category: `SCIENCE_REDUCED`

Created by: `amber_SciCal`

Association keywords:

`DET NTEL`

Note:

Number of Telescopes used

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	39 of 46

9 Algorithms

In this section the data reduction procedures applied by the 23 pipeline recipes currently in use (see Section 6.2) are described in some detail. Common algorithms, as cosmic rays removal or bad pixel cleaning, are described separately.

9.1 General Algorithms

9.1.1 Dark subtraction

Dark frames are taken during the actual observation run. If no dark was taken though, the dark channel (left most on CCD chip) will be taken as a substitute - with less accuracy.

9.1.2 Flat field correction

The flat field correction merely consists of dividing the frame to be corrected by a given master lamp flat field frame produced by the recipe *amber_detector*

It is currently under discussion if the flat field is implicit with the p2vm correction.

9.1.3 fringe search

TODO.

9.2 Recipes Algorithms

9.2.1 recreation of active CCD sensor area

TODO.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	40 of 46

A Installation

This chapter gives generic instructions on how to obtain, build and install the AMBER pipeline. Even if this chapter is kept as up-to-date as much as possible, it may not be fully applicable to a particular release. This might especially happen for patch releases. One is therefore advised to read the installation instructions delivered with the AMBER pipeline distribution. These release-specific instructions can be found in the file `README` located in the top-level directory of the unpacked AMBER pipeline source tree. The supported platforms are listed in Section A.1. It is recommended reading through Section A.2.3 before starting the installation.

A bundled version of the AMBER pipeline with all the required tools and an installer script is available from <ftp://ftp.eso.org/pub/cpl/amber>, for users who are not familiar with the installation of software packages. The AMBER pipeline can also be downloaded from

<http://www.eso.org/projects/dfs/dfs-shared/web/vlt/vlt-instrument-pipelines.html>

or

<http://www.eso.org/observing/gasgano/amber-pipe-recipes.html>.

A.1 Supported platforms

The utilisation of the GNU build tools should allow to build and install the AMBER pipeline on a variety of UNIX platforms, but it has only been verified on the VLT target platforms:

- HP-UX 11.00 or later,
- Linux (glibc 2.1 or later),
- Sun Solaris 2.8 or later,

using the GNU C compiler (version 3.2 or newer).

A.2 Building the AMBER pipeline

This section shows how to obtain, build and install the AMBER pipeline from the official source distribution.

A.2.1 Requirements

To compile and install the AMBER pipeline one needs:

- the GNU C compiler (version 3.2 or later),
- the GNU `gzip` data compression program,
- a version of the `tar` file-archiving program, and,

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	41 of 46

- the GNU `make` utility.

An installation of the Common Pipeline library (CPL) must also be available on the system. Currently the CPL version 2.0 or newer is required. The CPL distribution can be obtained from <http://www.eso.org/cpl>.

Please note that CPL itself depends on an existing `qfits` installation. The `qfits` sources are available from the CPL download page or directly from the `qfits` homepage at <http://www.eso.org/projects/aot/qfits>. In conjunction with CPL 2.0 `qfits` 5.2.0 must be used.

In order to run the AMBER pipeline recipes a front-end application is also required. Currently there are two such applications available, a command-line tool called *EsoRex* and the Java based data file organizer, *Gasgano*, which provides an intuitive graphical user interface. At least one of them must be installed. The *EsoRex* and *Gasgano* packages are available at <http://www.eso.org/cpl/esorex.html> and <http://www.eso.org/gasgano> respectively.

For installation instructions of any of the additional packages mentioned before please refer to the documentation of these packages.

A.2.2 Downloading the AMBER pipeline source distribution

From the ESO ftp server, <ftp://ftp.eso.org/pub/cpl/amber>, the latest release of the AMBER pipeline distribution is available.

The AMBER pipeline sources are distributed as a gzipped tar archive named like `amber-X.Y.Z.tar.gz`, where `X` and `Y` are the major and minor release numbers, and `Z` indicates the patch level (which might be missing if no patch has been released).

A.2.3 Compiling and installing the AMBER pipeline

It is recommended to read through this section before starting with the installation.

1. First, if an appropriate version of CPL (c.f. section A.2.1) does not already exist on the system, compile and install the CPL libraries and their dependencies. For detailed instructions on how to install the CPL libraries please refer to the CPL documentation.
2. Unpack the AMBER pipeline sources in a chosen directory using

```
$ zcat -d amber-X.Y.Z.tar.gz | tar -xf -
```

at the system prompt. This will create a directory called `amber-X.Y.Z` containing the source tree.

3. Run the configuration script `configure` which is located in the top-level directory of the AMBER pipeline source tree.

If the CPL or `qfits` have not been installed into one of the system's standard directories, the configuration script must be told where the dependent libraries (CPL, `qfits`, etc.) can be found. This is either done using the command-line options `--with-cpl`, `--with-cext` and `--with-qfits` of the script or by defining the two environment variables `CPLDIR` and `QFITSDIR`.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	42 of 46

For example, if the CPL has been installed into `/somewhere`, (and therefore the CPL header files are located in `/somewhere/include` and the libraries are in `/somewhere/lib`), the argument of the option `--with-cpl` or the path assigned to the variable `CPLDIR` must be `/somewhere`, *i.e.*, the root directory of the CPL installation. The same is true for the option or environment variable corresponding to `qfits`.

If command-line options are used, the argument of `--with-cext` is usually the same as the one passed to `--with-cpl`, since both packages install together. This is also reflected by the fact that there is no separate environment variable for the `cext` package.

By default the configuration script will setup the installation path of the package as `/usr/local`. This means that root privileges would be required to complete the installation. If one does not have sufficient privileges, or the default installation path is just not appropriate, it is possible to change this default. There are two possibilities: using the script's `--prefix` option, or defining the environment variable `PIPE_HOME`. Using the option will override a possibly defined variable.

In either case the provided path is used for the installation of the AMBER pipeline. All its components will be located in the directory tree rooted at this path.

The following example assumes that the variables `CPLDIR` and `QFITSDIR` are properly set or the CPL and `qfits` have been installed into one of the system's standard directories. This should keep the following example commands as simple as possible.

Please note also that the usage of the default installation prefixes in the example below is just for demonstration purposes. Any directory for which one has write access can be used, although it is **not recommended** to use the distribution's source directory as the installation's target directory.

The simplest way to set up the package is to run the following command at the system prompt from the source-tree's top-level directory:

```
./configure --prefix=/usr/local/pipeline
```

Alternatively, if using the environment variable is preferred, `PIPE_HOME` may be defined in the shell's startup file. If the *Bourne* or a compatible shell (*i.e.* `sh`, `bash`, `ksh`, `zsh`, etc.) is used, to the file `.profile` (or `.bashrc` if `bash` is used) it should be added:

```
PIPE_HOME=/usr/local/pipeline
export PIPE_HOME
```

If the the C-shell (*i.e.* `csh` or `tcsh`) is used the commands above translate into:

```
setenv PIPE_HOME /usr/local/pipeline
```

and should be added to the C-shell startup file `.cshrc`.

To activate these settings one may either logout and login again, or source the startup script manually.

4. To compile and install the AMBER pipeline on the system, run the following sequence of commands from the source-tree's top-level directory:

```
$ make
$ make install
```

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	43 of 46

The `configure` script provides a variety of command-line options to customise the AMBER pipeline installation. The list of available options can be obtained by running `./configure --help`. Using a command line option always takes precedence over any previously set environment variable.

However it is important to include the options `-enable-type-prefix` `-enable-float` to force the `fftw` subsystem to compile correctly. In case your box has already the package `scisoft` pre-installed you also have to add the `F77="` for proper compilation of AMBER pipeline.

A typical configuration could look like that:

```
`../configure -prefix=$CPLDIR -enable-type-prefix -enable-float F77="`
```

After the installation has been completed the source tree is no longer needed and can be removed.

A.3 Configuring the pipeline recipe front-end applications

In this section an outline is given how to set up the recipe front-ends *EsoRex* and *Gasgano* so that the just installed AMBER pipeline recipes can be executed by these applications.

For detailed instructions on how to configure the two currently available front-end applications, *EsoRex* and *Gasgano*, please refer to their documentation, available at <http://www.eso.org/cpl/esorex.html> and <http://www.eso.org/gasgano> respectively.

In the following it is assumed that the AMBER pipeline was installed as described in Section A.2.3, *i.e.*, that the AMBER recipes have been copied into `/usr/local/pipeline/lib/amber/plugins/amber-X.Y.Z` (X, Y and Z indicate the version number of the recipes).

A.3.1 Setting up the EsoRex command-line tool

The general syntax for the *EsoRex* front-end is the following:

```
esorex [esorex_options] recipe_name [recipe_options] set_of_frames
```

In order to execute a recipe, *EsoRex* must be told where the recipes can be found. This location can be passed to the tool using its command line option `--recipe-dir` followed by the complete path to the recipes. In this scenario the command to be executed at the shell's prompt will look like:

```
$ esorex --recipe-dir \  
> /usr/local/pipeline/lib/amber/plugins/amber-X.Y.Z ...
```

However, the path to the recipe location(s) can also be set in the *EsoRex* configuration file. If an *EsoRex* configuration file does not already exist, it can be created by executing the command

```
$ esorex --create-config --recipe-dir \  
> /usr/local/pipeline/lib/amber/plugins/amber-X.Y.Z
```

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	44 of 46

In addition to just creating the configuration file in its standard location `$HOME/.esorex/esorex.rc`

the path to the recipes is also added by this command. This way for updating the configuration, can be repeated whenever a new version of the recipes is installed. It will replace the path to the recipes in the *EsoRex* configuration file.

If *EsoRex* has already been used for running recipes from other instruments, the path to the AMBER recipes must be appended to the existing configuration file. To do this edit the configuration file `$HOME/.esorex/esorex.rc`. Go to the entry starting with `esorex.caller.recipe-dir`.

This is a colon separated list of directories searched by *EsoRex* for recipes. To add the AMBER recipes just append the AMBER recipe installation directory, separated by a colon (:), to the end of this list.

To verify the updated configuration execute the following command at the shell's prompt:

```
esorex --recipes
```

This should display a list of the available recipes on the terminal screen.

At the *EsoRex* homepage, <http://www.eso.org/cpl/esorex.html>, a detailed description of the application can be found.

A.3.2 Setting up Gasgano

The AMBER recipe set can be incorporated into *Gasgano*'s configuration using the *Preferences* dialog from the *File* menu. Select the tab labeled *Recipe Configuration* and press the *Add Recipe* button. A file selection dialog pops up where the recipes to add can be selected. The selection has to be confirmed and *Gasgano* must be restarted to activate the new recipe configuration. Now the recipes are seamlessly integrated into the application and the files to process can be passed to the recipes using drag'n drop (for details please have a look into the *Gasgano* User Manual [11]).

Do not forget to activate the *AMBER.rul* in your *Gasgano* preferences to make it classify the AMBER files correctly.

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	45 of 46

B Troubleshooting Guide

ESO	AMBER Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.8
		Date:	Date 2006-09-18
		Page:	46 of 46

C Geometrical distortions models

The reduction of AMBER scientific data made by the AMBER pipeline is not based on a set of predefined models of the optical distortions affecting the instrument. However it relies on the P2VM to correct for distortions.

(?!?)