Processing of ATSR-2 data for July 2003 to 2008

	Name	Organisation	Signature and date
Prepared	A Smith, M. Bates	RAL	A Smith Digitally signed by Andy Smith DN: cn=Andy Smith, o=STFC, ou=RAL Space, email=andy.smith@stfc.ac.uk, c=GB Date: 2016.12.01 17:20:48 Z
Checked	Hugh Kelliher	Space ConneXions Limited	
Approved		RAL	

Science and Technology Facilities Council
Rutherford Appleton Laboratory
Harwell Science and Innovation Campus
Didcot
Oxfordshire OX11 0QX
United Kingdom

Distribution

Name	Organisation
ATSR team	RAL Space
Hugh Kelliher	Space ConneXions Limited
Ruth Wilson	Space ConneXions Limited
Miguel Martinez-Boti	BEIS
ATSR QWG	Various

Change Record

Issue	Date	Description
0.1	27-Oct-2016	First draft
0.2	25-Nov-2016	Martin Bates checked and updated HEY data section.
0.3	29-Nov-2016	Comments received from Hugh Kelliher and Ruth Wilson. Minor updates made in response.
1.0	01-Dec-2016	Updates accepted, prepared issue 1 for release.

Table of Contents

1 Scope of Document

This Technical Note describes the work performed by RAL Space to prepare and process the ATSR-2 data set for the period July 2003 to February 2008. This period follows the failure of the ERS-2 on-board tape recorder and the data available are highly fragmented. The preparation involves the testing and development of the APP software and the HEY data required to process from UBT to Level 1B.

This work is performed as Work Package 4.5 of the proposal described in [AD 1].

2 Terms, Definitions and Abbreviations

2.1 Acronyms

AATSR	Advanced Along Track Scanning Radiometer
APP	Archive Product Processor (RAL ATSR-1&2 processor)
ATSR	Along-Track Scanning Radiometer
CEDA	Centre for Environmental Data Analysis
CfE	Consolidate for Envisat (1st stage APP processor software)
ERS	European Remote Sensing satellites
ERS-CFI	ERS Customer Furnished Item (software library provided by ESA for orbital state vector processing)
HEY	Helpful ESA Yaw file (containing yaw correction information)
IDL	Interactive Data Language
NEODC	NERC Earth Observation Data Centre
QUASAR	IDL software package for visualisation of ATSR data
QWG	ATSR Quality Working Group
SADIST(-2)	Synthesis of ATSR(-2) Data Into Sea surface Temperature (RAL software to process Level 0 ATSR data from tape into UBT products)
SPR	Software Problem Report
STEP	SADIST to Envisat Processor (2nd stage APP processor software)
SUPPLE	Sadist UBT Processor Linux Environment. Linux-ported version of the SADIST L0 to UBT processor software
UBT	Ungridded Brightness Temperature (Level 1 ATSR product, generated at single-scene spatial coverage, 512x512km)

3 Documents

3.1 Applicable Documents

Ref	Title	Document code	Version	Date
AD 1	ATSR Satellite Dataset Supporting Activities, 2014 - 2017	Proposal 2014-07-001 (response to DECC ITT : TRN 829/06/2014)	2	28-Jul-2014

3.2 Reference Documents

Ref	Title	Document code	Version	Date
RD 1				

4 Overview

The proposal [AD 1] describes the science case for this work package.

The ERS-2 on-board tape-recorders failed in mid-2003 (the last full-orbit data archived by the NEODC/CEDA are from 22-Jun-2003). Following the failure it was no longer possible to fulfil a global mission for ATSR-2. Data were acquired only through real-time down-link while the spacecraft was in communication with ground stations. Nonetheless, it was expected that it should be possible to acquire good coverage of the North Atlantic and, possibly, other regions such as the Mediterranean.

Prior to this work, ATSR-2 scientific data acquired after mid-2003 had never been processed.

The data from this period differ substantially from earlier data, described by [AD 1] as follows:

“Unfortunately, the data from post July 2003 are very fragmented and as such cannot be processed using the “standard” set of software components since these expect to be provided with a whole orbit product. An added complication is that the ‘HEY’ files that are used to provide geolocation corrections are possibly likewise segmented.”

“This raises several issues with respect to first deriving a HEY correction file for the processor to use; then in processing sub-orbital fragments.”

“This activity requires:

- The corresponding L0 data - these are now available at RAL and transferred to disk (~2Tbytes) to allow efficient bulk data processing.
- Acquisition of the HEY-correction files (now at RAL), cataloguing and processing to extract the data subset required. Investigating the degree of fragmentation (if present).
- Analysis of the “HEYstack” concept (used by the APP) and whether it can be extended to handle the partial and fragmented real-time-only orbits of the July 2003 and onwards era. It is clear that this is a significant task with substantial re-working of the current approach being required.
- UBT production using SUPPLE. For ATSR-2, L0 orbit fragments can be processed already to UBT. A script will need to be prepared to execute bulk-processing and I/O management efficiently.
- The processor expects to be handling orbital products; the processing of suborbital segments requires substantial work to allow the processor to function. In the early stage of the post gyro failure, the data was very segmented and this presents a processing challenge, to adapt a processor that was designed to handle only whole orbit products.”

At the outset it was not clear whether the UBT data should be processed to Level 1B as individual fragments or as orbits. Following discussion with the QWG it was decided that processing should be orbit-based as the large number of products resulting from fragment processing would be difficult for users of the data set to manage.

5 Processing from Level 0 to UBT

As stated in [AD 1], the Level 0 data for this period had been transferred to RAL by the time of the proposal. ESA supplied RAL with 57,728 ATSR-2 Level 0 files of orbit-segment, unconsolidated data, spanning 17th July 2003 through to the end of the mission (early February, 2008).

Processing of these Level 0 files to UBT was completed in 2014 using the SUPPLE software on a Linux processor. Approximately 20% of the Level 0 data files failed to process, with the SUPPLE processor reporting ERS-CFI errors related to the state-vector and/or time-correlation data available in the Level 0 headers.

All 502,921 UBT files generated from the Level 0 data have been archived by NEODC and are directly accessible from the development server for UBT to Level 1B processing.

6 Tests and updates to the APP processor

As noted in the overview (section 4), it was not clear initially whether the data for this period could be processed as whole orbits, or whether the APP processor would run more successfully on partial orbits. It was decided that the better approach would be to process whole orbits, in order to give continuity with the data set for the rest of the mission and to make the resulting data set easier for end users to deal with.

Since this data set is restricted to data obtained by real-time down-link, the data are characterised by orbit segments typically containing about 6 UBT products, as compared to typically 80 products for a full orbit. The development work was therefore focussed on processing of short segments and on the behaviour of the software in orbit sections where large data gaps are present.

Processing tests were performed on small subsets of orbit UBT data from before July 2003, artificially creating gaps by selective removal of UBT products to give a controlled test environment in which comparisons could be made between products with and without data gaps. Removal of a single UBT product from a sequence generates a gap of 512 scans (assuming all scans are present in a given UBT product).

The APP Level 1B processor consists of 3 components:

1. the “APP” control script, which sets up environment variables and auxiliary data for the orbit based on the selected instrument, and executes the other 2 components,
2. the first-stage processor, “CfE” (Consolidate for Envisat) which reads in the individual UBT products, deals with missing and duplicated scans and generates a “stitched”, intermediate UBT product
3. the main processor, “STEP” (SADIST to Envisat Processor)

The handling of data gaps is therefore performed by the CfE component.

At the outset of this work the APP software had not been re-compiled since the 3rd reprocessing of ATSR-2 data was performed in 2013. Re-compilation in an updated Linux environment revealed some minor software faults. Work was done throughout this development to “tighten up” areas of the software which generated compiler warnings, to eliminate possible causes of error.

For example, in an early test a section of an orbit was processed a number of times, firstly with a complete orbit segment and then with UBT products successively removed from the input on each iteration to produce a gap of length 1, 2, 3, … UBT products. On inspection of images from the resulting output products, a coastal feature was found to have “moved” to a position that suggested 96 scans were missing. This change was traced to the absence of a compiler flag “-ansi” when the CfE software was re-compiled. Such a change in behaviour in response to a change in compilation flag was unexpected. The reason for the difference was not understood but it may have indicated a vulnerability in the software.

Figure 1 illustrates the “-ansi” compiler flag error through a comparison of QUASAR images of an archived product from the 3rd reprocessing (left) and a product generated by the re-compiled CfE (right). Although the coastlines at the top of the two images are not exactly aligned (because QUASAR only allows stepping by one complete screen at a time), it is clear that the coastal features and image features in the lower part of the right-hand image have moved relative to the equivalent features in the left-hand image.

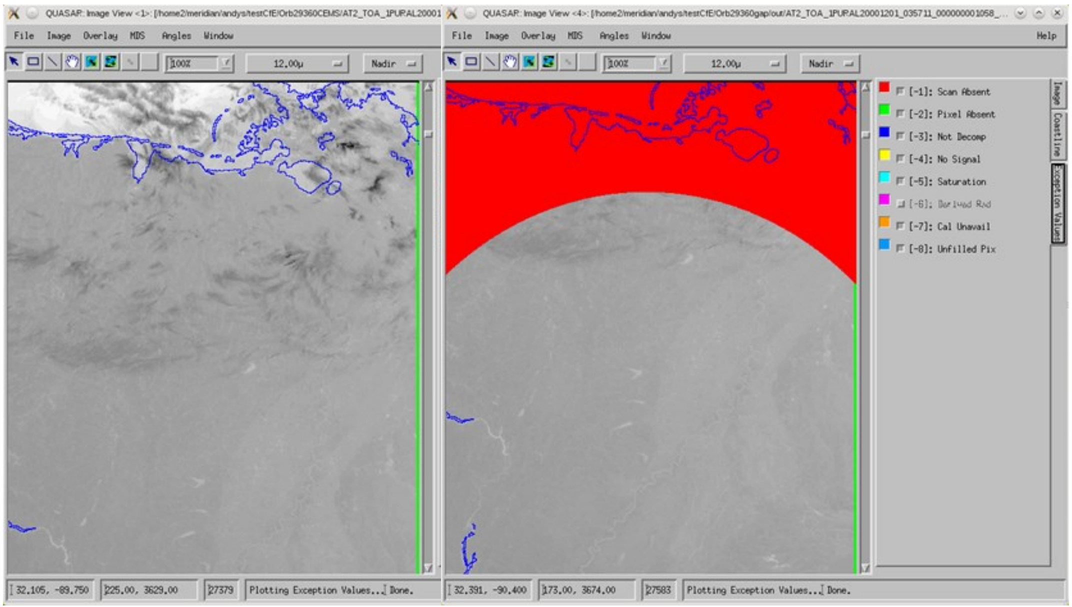

Figure 1: Comparison of two QUASAR screenshots from Level 1B products for the same scene. Left: archived product from the 3rd reprocessing. Right: test product with an artificial gap created by removing UBT products from the input.

The red area in the right-hand image in Figure 1 represents an Exception Value flag, in this case a value of -1: “Scan Absent”. On correcting the flag in the CfE compilation and reprocessing, the output product contained different flag values in the gap, as shown in Figure 2. A single flag value indicating “scan absent” was expected in the data gap. In this new case flags for both “scan absent” and “unfilled pixel” were present, as well as scans containing apparently valid data where no UBT input data had been supplied.

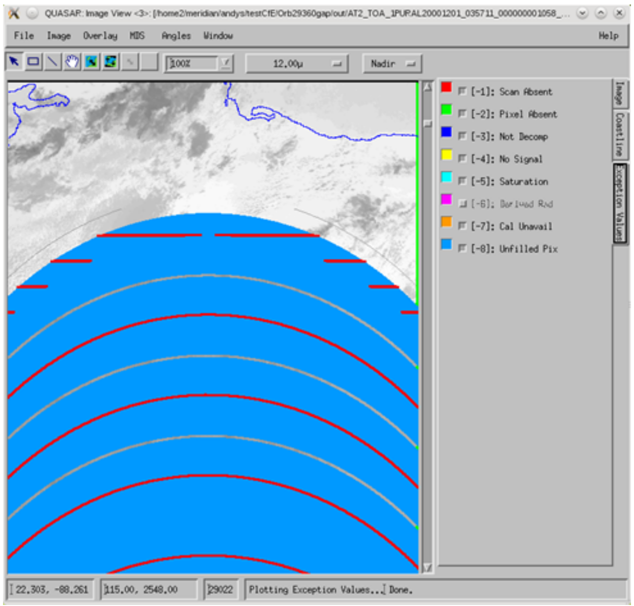

Figure 2: QUASAR screenshot of the Level 1B product resulting from application of the –ansi compiler flag.

Testing of the APP software prior to this point had relied on visual inspection of the output Level 1B "TOA" product, e.g. using the QUASAR IDL software. Visual inspection of images can only give a qualitative idea of the differences. Furthermore, the Level 1B product is the output of the second stage of processing, so it is not possible to determine whether errors detected at this point originate in CfE or STEP. There was no way of checking the outputs from CfE. A new software tool was written to list part of the contents of the intermediate orbit UBT product generated by CfE.

The new listing enabled inspection of data values within the orbit UBT product and isolation of errors to CfE or STEP. On inspection of the listings it was found that the satellite binary time incremented by 1 in each null source packet inserted by CfE to replace a missing scan. The typical increment between scans is expected to be of order 38-39. This problem was traced to the use of a rounding function from a standard maths library, which returned a floating-point data type when an integer type was assigned. The function call was replaced with one of a different type, resulting in more consistent flagging within the gap, although some unexpected features remained (see Figure 3, left-hand image). Subsequently, a small cumulative rounding error was corrected, leading to "clean" handling of the data gap (right-hand image of Figure 3).

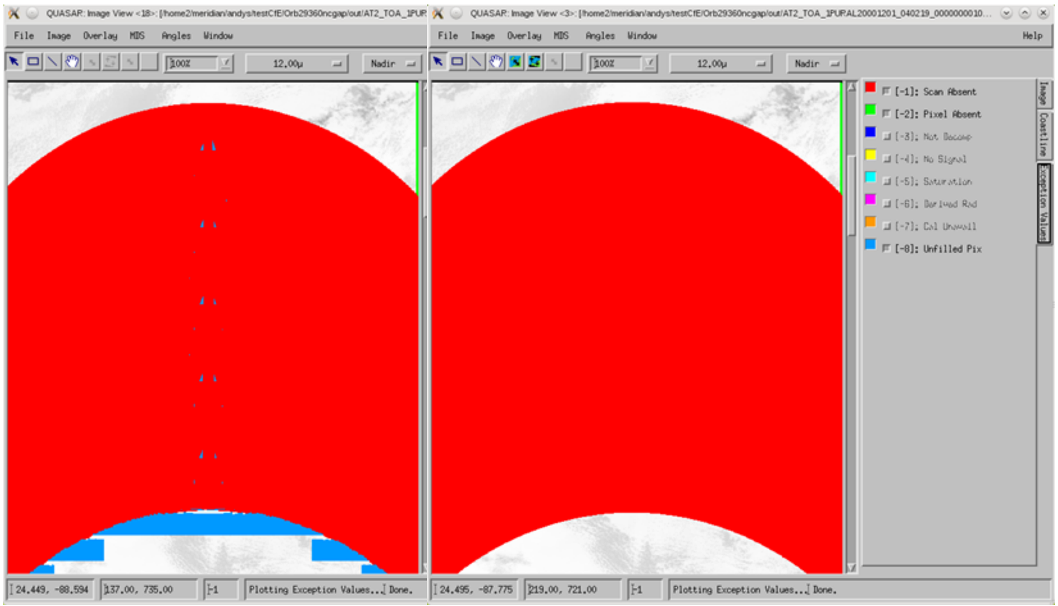

Figure 3: QUASAR screenshots illustrating two successive fixes to the satellite binary times assigned to null packets by CfE

During the work to trace and correct the software faults above, a standard set of tests for CfE was established, allowing any further changes to be regression-tested against the previous results.

Comparisons of test outputs were made for different processing runs consisting of full and partial orbits, performed on the Linux development hardware and also on the CEMS hardware which is used for bulk reprocessing of ATSR data. These test runs gave confidence that consistent results could be achieved on the two processing platforms.

Wherever possible, APP is required to generate “consolidated” products, in which the product for a given orbit extends a little way beyond the ascending node crossing point at each end of the orbit, so that there is full-orbit coverage in both the forward and nadir ATSR views. STEP also requires data to be present for a continuous sequence of scan lines, hence CfE deals with missing scans by “padding” with null data. Checks on the available UBT data showed that ascending node data appear to be absent from all orbits in the July 2003 to 2008 period, due to the location of the available receiving stations.

An attempt was therefore made to update the CfE code to fill the gap from the beginning of each orbit (the ascending node) to the first available scan, extrapolating backwards in time so that the output products would start consistently from the ascending node. Such products would be consistent with the data set up to July 2003, both in data coverage and in product naming. Some progress was made in the development of updated CfE software to fill the gap, but problems were encountered with the approach. The update was more complex than originally thought, as in addition to padding the gap with null source packets, further updates were required to generate the time information required from the orbit UBT product in the subsequent processing stage. A further problem was found when the filling was done; there were further apparently missing scans in the output product, in the filled range. Following consideration of the likely effort required to solve these problems and the usefulness of the resulting updated products, it was decided to discontinue development in this area. As a result, the final Level 1B products have start times which match the first scan present, rather than the time of the ascending node. The time intervals between successive product start times may be irregular, depending on the level of fragmentation of data from one orbit to the next. It is assumed that user software does not rely on easily-predicted product times in filenames.

Finally, following the development work on HEY data described in section 7 the APP control script needed a minor update to locate the orbit-based HEY files for this period.

7 Testing and preparation of HEY data

The HEY data set provides a correction to the ERS-2 satellite yaw steering information for the period following the gyro failure in 2001. Use of this data set improves the accuracy of the geolocation derived by the APP processor. The data set is derived by ESA and supplied separately from the Level 0 ATSR-2 data set. The data must be pre-processed before use by the APP software.

The ESA HEY files contain 74 values at each time interval, of which only one is required by STEP. The HEY pre-processor first stage, “StackHEY”, strips out the additional values to generate a file of a more manageable size and concatenates the data into one file per cycle. STEP is coded to expect HEY data at intervals of 30 seconds. The original data delivered by ESA are sampled at approximately 0.64s intervals, and are therefore re-sampled by the 2nd stage pre-processor, “ResampleHEY”, using spline interpolation. The final output is one large HEY file covering the whole section of the mission from February 2001 to June 2003 (known as the “HEYStack”). During processing, STEP re-interpolates the HEY data from the 30s intervals to give continuous coverage.

As the post-July 2003 data set is highly fragmented it was considered that the best approach would be to generate single-orbit HEY files in the HEYStack format. Generation of a single file covering the whole period between July 2003 and March 2008, given the sparse coverage in this period, would require interpolation over frequent, very large data gaps.

This part of the work package therefore involved the following questions and sub-tasks.

- What is the coverage of the HEY data relative to the UBT data?
- How well does the fitting done to re-sample the HEY data perform over gaps in the data?
- Is it possible to extrapolate the spline fit to cover the start and end of the available UBT data? (In cases where the HEY data starts later or ends earlier than the UBT data)
- Preparation of the bulk processing software used in generation of the “production” HEY data set for use in the current processing environment.
- Update the HEY processing code to generate HEY data files on a per-orbit timescale.

The pre-processor code was updated to run in the current Linux processing environment and a trial run was performed on the 5-year HEY data set. A full processing requires about a day.

7.1 HEY data coverage

The initial tests performed on the HEY data examined the orbital coverage for sample orbits at different points in the 2003-2007 time range. The coverage improves as time goes on, due to the use of an increasing number of ground stations for downlink. For example, in cycle 86, 2003 the typical coverage per orbit was found to be approximately 7%, while in 2007, cycle 132, coverage had improved to approximately 13%. This low level of coverage was taken as confirmation that the generation of a single HEY “stack” file, as used in the earlier part of the mission, would not be possible as the data gaps to be interpolated are much bigger than the available data periods.

Comparisons of sample UBT files and corresponding HEY data files revealed the following:

- orbit files can consist of one or more orbit fragments;
- the (raw) HEY data range for each orbit fragment was generally found to be slightly narrower than the corresponding UBT data: the UBT data typically extends approximately 1s further at each end of the data range;
- there are exceptions, where the UBT data extends out much further than the corresponding HEY data. In the worst case seen, based on processing a small sample of data, there was a period of 13 minutes of UBT data with no corresponding HEY data.

By contrast, in order to process the UBT data, STEP expects HEY files which (a) extend out further than the UBT data being processed and (b) are complete (i.e. no missing 30s sample points with the time range covered by the HEY “stack” file).

In order to get HEYStack format orbit files which STEP can use, it is therefore necessary to:

- interpolate between orbit fragments within an orbit, to ensure that all 30s sampling points are populated;
- extrapolate at least to the next 30s sampling point at the start of the first fragment and the end of the last fragment, since STEP requires HEY data sampling points at either side (in time) of the UBT data being processed.

The spline fit used in generating the 2001-2003 HEY data was adapted to work “per orbit” on orbit files for the 2003-2008 period: a single fit is performed on all the orbit fragments in an orbit. A number of orbits were studied to see how well the fit performed in the interpolation and extrapolation regions.

- Firstly, the fit works well in generating 30 second sampling values within the time range covered by the HEY data orbit fragments;
- the time gaps between orbit fragments, together with the variability of the value of the yaw parameter, mean that the interpolated values between orbit fragments are not in general meaningful – there is not enough data to specify how the yaw data behaves in the gaps between orbit fragments;
- the spline fit very quickly becomes unstable and unreliable when extrapolating beyond the first/last orbit fragment.

It was decided that:

- only the interpolation (between orbit fragments) to the first 30s sample point would be treated as meaningful;
- the other interpolated sample points need to be kept in order for the STEP processing to function mechanically, but the values at these points cannot be considered reliable (these points are marked invalid – see Section 7.2);
- the spline fit extrapolation even to the first 30s sample point is too unstable to be useable – instead, this sample point is populated with a linear extrapolation made from the last 10 available raw HEY values.

This will at least produce full HEY coverage in the most usual cases, up to where the UBT data extends beyond the range of the HEY data by a few seconds, but not where the UBT data extends further beyond the range of the HEY data.

Figure 4 illustrates the results of interpolation and extrapolation on one sample orbit.

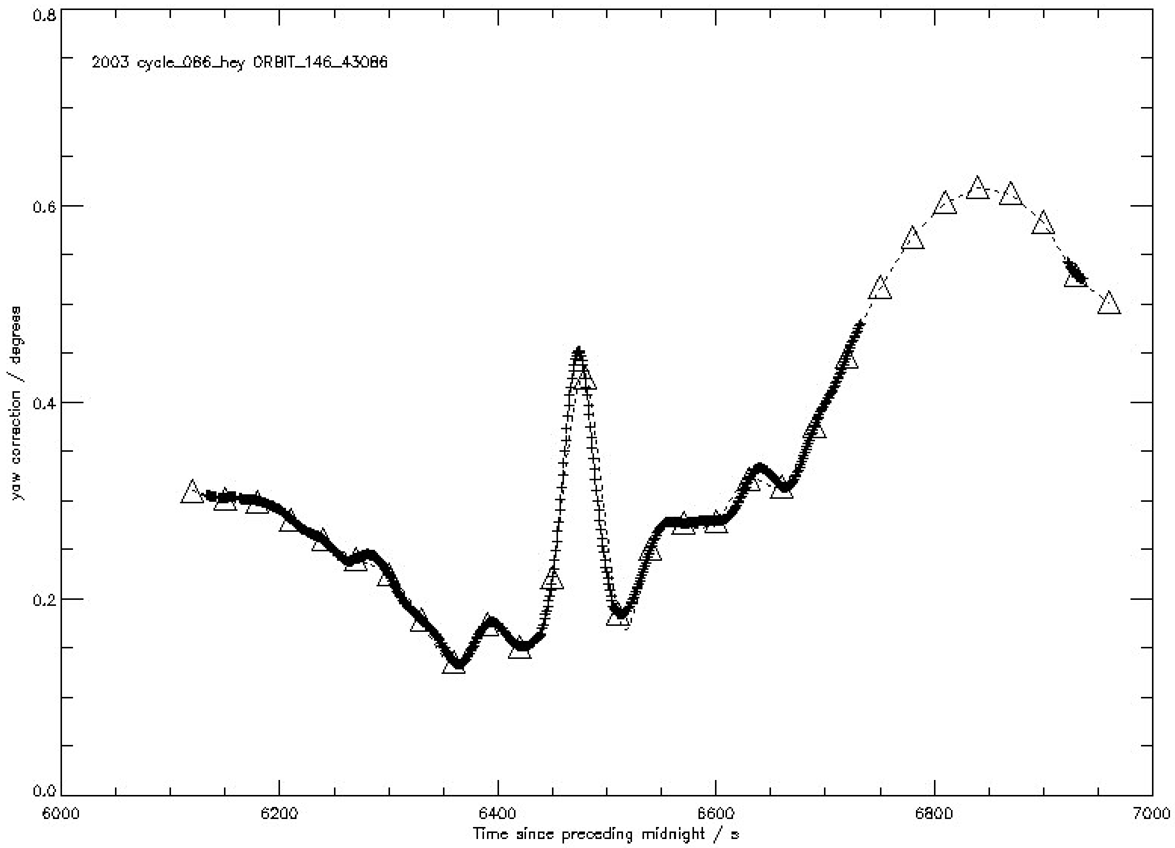

Figure 4: Interpolation and extrapolation of HEY data. Symbols: ++ HEY data, ΔΔ HEY "stack" (spline fit, 30s samples), -- HEY data (reconstructed from stack). Each end of the curve is extended (to the next sample point) by "external" extrapolation using a linear fit to the last 10 data points. The large gap towards the right-hand side is filled by interpolation using the spline fit.

7.2 Flagging of invalid HEY data

Following the investigation into HEY data coverage compared to UBT data, it is apparent that there are periods in the 2003-8 data set for which no HEY data exists and for which the geolocation quality will be lower than elsewhere. It was therefore considered useful to introduce a flag into the HEY data to indicate which values were potentially invalid as they were reconstructed based on a spline interpolation between periods of available data. Since the spline fit generates a value every 30 seconds, including filling in the gaps between data fragments, inter-fragment gaps are not normally visible to the STEP or other processing code.

Flagging could serve two purposes, (1) to enable the Level 1B processor to adapt, e.g. switching on or off options to use the HEY data, and (2) to provide quality information to the end user. It was intended that a flag value could be propagated into the Level 1B products by the APP software, so that the user had an indication of the likely geolocation quality of each product.

The HEY pre-processor was updated to add a flag column into the re-sampled HEY data file, to indicate invalid values (flag value 1) where data have been interpolated, or valid values (flag 0) where HEY data are available. The APP STEP processor was updated to cope with the presence of the flag value in the HEY file when reading the data. However, the intended update to switch HEY correction on or off within STEP depending on the invalid flag value was not done. The proposed change represents a more complex approach than that which was coded in the existing STEP processor, requiring a considerable amount of effort.

Difficulties were encountered in finding an appropriate way to add a flag value to the Level 1B product, since there are no spare fields within the current product definition. This work was eventually discontinued due to time and effort constraints. The default is therefore that the Level 1B product does not distinguish between data with valid HEY data and data with interpolated HEY data.

Users of the current orbit HEY data files should be aware of the presence of the new flag field, which may affect existing code to read the data.

7.3 APP test processing

A number of tests and comparisons of APP processing runs with and without HEY data were performed. The main purpose of these tests was as an integration test, to verify that orbit HEY files in HEYStack format could be used by the APP processor to process UBT data. Quality assessment of the resulting products was based on image inspection using QUASAR, and is therefore subjective. The approach taken was to identify suitable scenes, i.e. cloud-free scenes containing coastlines, and to compare the results of processing with and without HEY data.

The APP processor successfully generated level 1B products. It was hoped to make a first check of the resulting geolocation quality, but initial results from the small number of suitable scenes identified were inconclusive. Examples are shown in Figure 5 and Figure 6 below. While it was possible to identify differences between scenes processed with and without HEY data, and certain coastal features might appear to be better matched in some cases, there is no clear “good” or “bad” geolocation result for a whole scene, hence the inconclusive nature of the test.

It should be borne in mind that the aim of the processing tests was not to validate the HEY data itself, or to improve the APP geolocation software, but to check that the available HEY data could be used by APP and that no adverse side-effects had been introduced by the conversion to orbit-based HEY products.

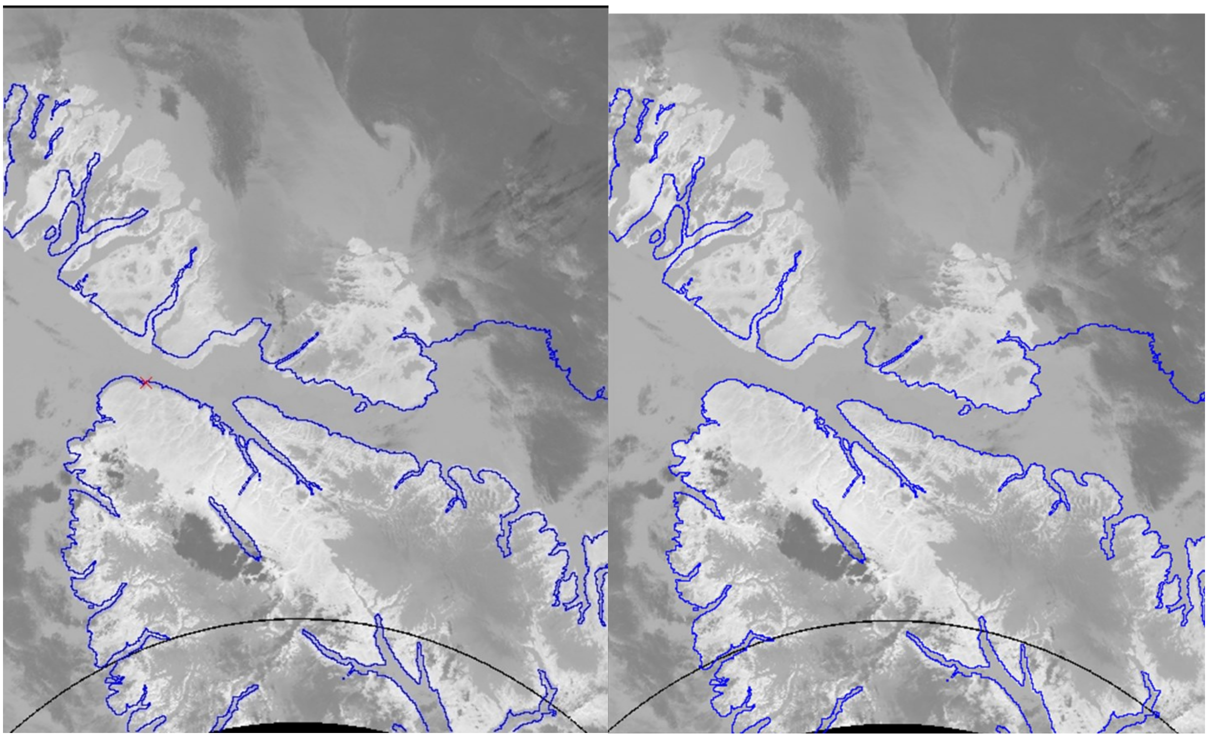

Figure 5: QUASAR screen shot showing a 12µm nadir scene from a Level 1B product processed without HEY data (left) and with HEY data (right). Data from orbit 43098, 18-Jul-2003 21:58:00.

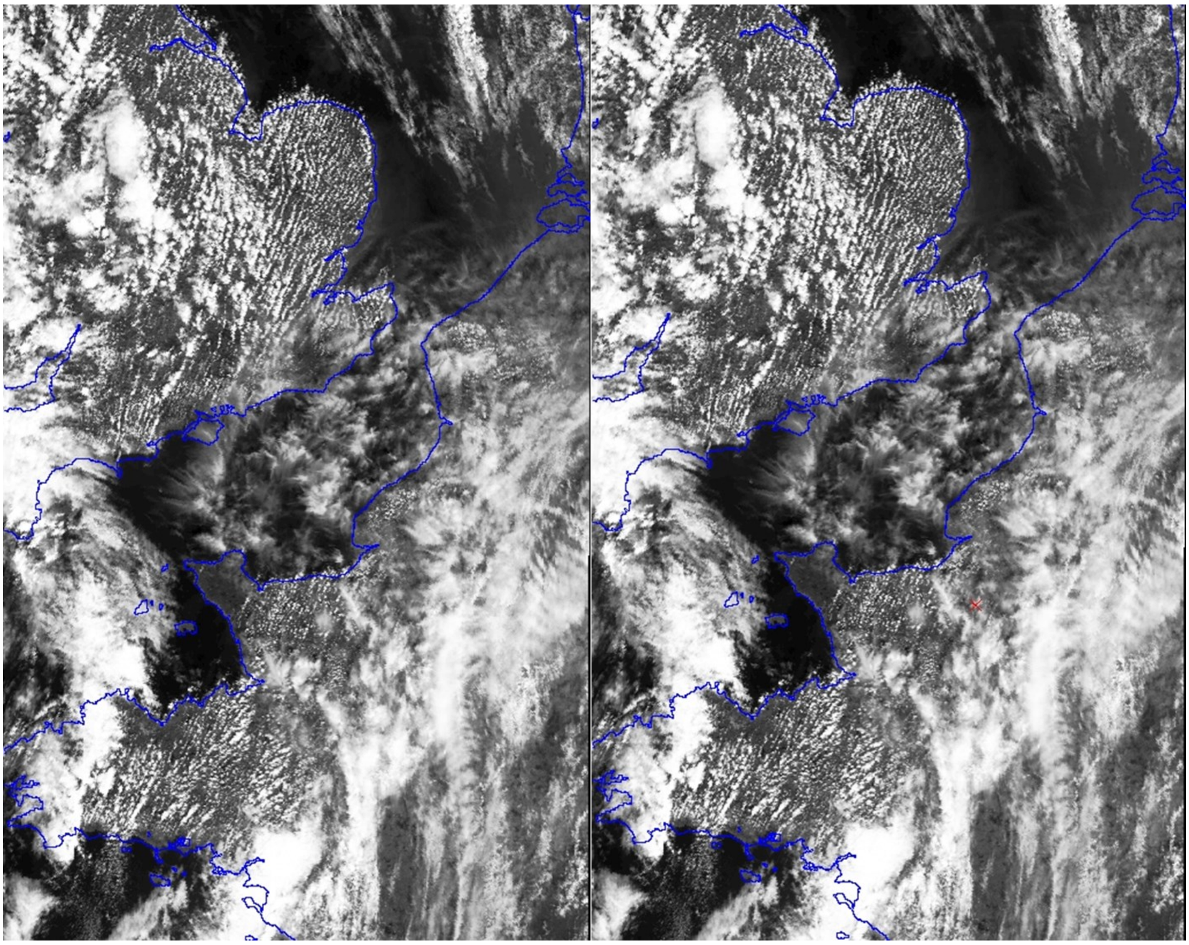

Figure 6: Comparison of 12μm nadir scenes from a Level 1B product processed without HEY data (left) and with HEY data (right). Data from orbit 43120, 20-Jul-2003 10:59:30.

8 Bulk processing and job management software

8.1 Software updates

Bulk, production processing of the post-July 2003 data set was done using essentially the same system as the 3rd reprocessing of ATSR-1 & 2. Processing was performed on the RAL CEMS/Jasmin system, making use of the available batch queues to process multiple jobs in parallel.

The APP software described above is capable of processing a single orbit of data. A script is run on the CEMS system to create a set of APP processing jobs for a requested range of orbit numbers. The script searches a catalogue database of available UBT products to identify the input products for each orbit. The database simplifies the task of identifying all UBT inputs for a consolidated orbit by allowing a search based on the time range of products, including the small overlap beyond the ascending node into the preceding and next orbits.

The existing UBT catalogue contained only the pre-2003 products and needed updating to include the new products. The database, originally maintained by NEODC (now CEDA), pre-dates the 3rd reprocessing and resides on old hardware. There was some difficulty in locating all the required software, as well as conflicting priorities with other tasks for the CEDA staff required to run the update. As a result it was not possible to update this database in time for the processing.

A work-around script was prepared, based on the assumption that all orbits in the post-July-2003 data set are incomplete, i.e. the data close to the ascending node are always missing. This assumption was tested on a sample of orbits from different dates. If the ascending node data are always missing there is no need to include data from the preceding or next orbits when processing a given orbit, so a simplified search was coded, based on listing the UBT archive directories and identifying products with matching ascending node times in the file names.

The APP processor requires an auxiliary file of metadata to link the mission orbit number and ascending node crossing time, which was found to be incomplete for the period 2003-4. The auxiliary data are generated from ESA "precise orbit" files, which existed for typically 4-5 of the 14 orbits per day. Replacement data was located through the ESA EO Help Desk and contacts in the ERS team.

8.2 Bulk processing and results

All UBT files for the period July 2003 to January 2008 were processed to Level 1B "TOA" products in October and November 2015.

The APP L2 processor was run on the resulting Level 1B products without any change to the processor software. Processing to Level 2 AR and NR products was completed for all years in the range. METEO products were generated from all L2 AR products.

All processed products are now available from the NEODC / CEDA archive. The products are stored as "segregated" due to the data quality issues affecting this period.

Table 1 below shows the number of orbit products successfully created at Level 1B and Level 2. The numbers for the year 2000 are shown for comparison. N.B. although products are generated for large numbers of orbits, the data coverage within each orbit is much sparser than in earlier years.

Year	No. of L1 TOA	No. of L2 AR	No. of L2 NR
2003 (July to Dec)	2115	2016	2115
2004	4625	4399	4624
2005	4952	4684	4952
2006	5079	4850	5079
2007	4840	4615	4840
2008 (Jan only)	426	417	426
2000 (for comparison)	4949	4926	4949

Table 1 lists the number of orbit products at Level 1 and Level 2 resulting from v3.0 processing of July 2003 to January 2008 data. Figures for the year 2000 are included as an example of a year of full-orbit data, for comparison.

9 Conclusions

The ATSR-2 data set for the period July 2003 to January 2008 has now been processed to UBT, Level 1B, Level 2 AR and NR and METEO products. All resulting products have been added to the NEODC archive, making ATSR data for this period available to the science community for the first time.

A number of software problems have been discovered and corrected in the APP processing code, particularly where the code handles gaps in the UBT data. These software problems were present when the 3rd reprocessing of the ATSR-1 and ATSR-2 pre-2003 data sets was performed. The ATSR QWG have been informed and a re-run of the 3rd reprocessing will be performed using corrected APP software.