

# A New Offline Data Processing System for TanDEM-X and TerraSAR-X Mission

Michael Wendler<sup>1</sup>, Heinz Wacker<sup>2</sup>  
*German Space Operations Center  
DLR Oberpfaffenhofen  
82230 Wessling, Germany*

Michael Staub<sup>3</sup>, Alexander Brandt<sup>4</sup>  
*CAM Systems GmbH  
Westendstr. 193  
80686 München, Germany*

**In contemporary and future spacecraft systems, it is of highest interest that the downloaded house-keeping data can be accessed very fast as all subsystem engineers rely on these data. For this purpose a new offline telemetry processing system has been developed by GSOC's sub-contractor CAM Systems for TerraSAR-X and TanDEM-X project and further future missions. The experience gained with the current TerraSAR-X offline processing system showed that the performance of this system is not sufficient to fulfill all tasks satisfactory. For the follow-on project TanDEM-X the demands for fast data access are very stringent due to the requirements of the formation flight of the two spacecraft. The new processing system is much more powerful than the existing system in terms of processing time due to parallel processing and intelligent distribution of processing tasks on virtual machines. This paper describes the Mission Offline Processing System (MOPS) in basics.**

## I. Introduction

The German SAR imaging spacecraft TerraSAR-X was launched in June 2007. In this project, a completely new way of data ordering for space-borne radar image products has been implemented. This posed a high degree of flexibility, but also reliability to the applied software tools. Especially the offline processing system had to run unattended most of the time and process the incoming data dump files automatically.

With the upcoming TanDEM-X mission, the requirements for reliable and fast offline data processing performance have even been increased. Especially the formation flight of both satellites will introduce much stricter performance requirements as the total amount of data to be processed will increase significantly and the timely requirements of both missions have to be met at the same time. Furthermore, the operational aspect in the sense of easy usability has been taken into account. These requirements consequently led to the implementation of a generic offline-processing and -assessment system for different mission types. Future missions have the possibility to make use of an existing and reliable system and its generated products.

Taking the pressure of time into account the first step taken was the elimination of performance issues experienced with the TerraSAR-X (TSX) offline-processing system. For that purpose a completely new offline processing system has been designed and implemented. The design and implementation benefit from already existing and proven components, such as the telemetry processor and the task monitoring process (TASKMO). Additionally, completely new components, e.g. the CRPlay process, have been developed to interface with the offline processing needs.

The second paramount task was the production of an offline processing system all of a piece. Once the system is configured it shall operate in a silent manner without necessary runtime manipulation or adaption. This goal has been achieved by implementing the core functionality (the actual TM frame and packet processing) against the

---

<sup>1</sup> Data Systems Engineer, Mission Operations, Michael.Wendler@dlr.de

<sup>2</sup> Data Systems Engineer, Mission Operations, Heinz.Wacker@dlr.de

<sup>3</sup> TMTC Engineer, Mission Operations, Michael.Staub@dlr.de

<sup>4</sup> TMTC Engineer, Mission Operations, Alexander.Brandt@dlr.de

infrastructure of the TASKMO and outsourcing of the configurable items such as composition of the different offline products to a scripting framework.

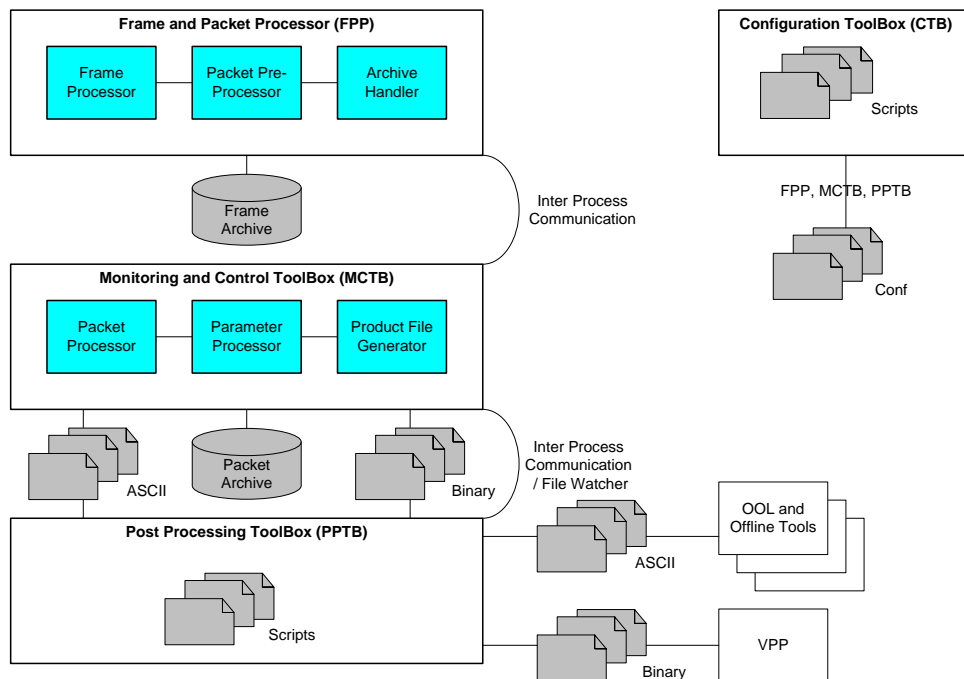
Throughout the development progress exhaustive and well documented tests have been performed to guarantee a fast, effective and traceable evolution of the Mission Offline Processing System (MOPS).

## II. Architectural Concept of the Offline Processing System

The MOPS is split-up mainly into three parts:

1. The PRE-processing part is covering the satellite (mission) specific part of the offline data processing. This part acts as an interface to all satellite missions and therefore should be unchanged for all development phases once implemented except of course the adaption to new data structures such as G/S specific frame recording formats. The PRE-processing part handles all tasks associated to telemetry frame processing, telemetry packet processing and on-board housekeeping store processing that are not directly involved in the parameter processing, such as telemetry packet extraction from the telemetry frames and generation of the source packet file store and archive.
2. The MAIN part is dealing with all issues associated to parameter processing, such as extraction from the telemetry packet, calibration of TM parameters, processing of derived (synthetic) parameters and generation of offline processing product files. Variable packet processing including the processing of grouped variable packets is actually done by the Variable Packet Processor (VPP). The input to the VPP is delivered from the MOPS using the current data format of group packet data files (\*.grp).
3. The POST-processing part is handling all tasks related to formatting (e.g. for special products) and compiling (e.g. for 24h product files).

The conceptual schema of the MOPS shown in Fig.1 is reflecting our aim to take over as all of the actually implemented offline processing system as possible to allow a migration as seamless as possible.



**Figure 1. MOPS Offline Processing Concept.**

As to be seen in our concept the TanDEM-X (TDX) MOPS is consisting of four main components. The Configuration ToolBox (CTB) is a dedicated set of scripts processing all necessary input (such as the S2K MiB) to allow all other subcomponents to run in the most efficient manner. The Frame and Packet Processor (FPP) plays the role of the Offline Pre-Processing. The Monitoring and Control ToolBox (MCTB) is responsible for the Main

Offline processing tasks. Finally the Post Processing ToolBox (PPTB) is handling all actions concerning the Offline Post-Processing.

Fig.1 is also showing the products generated by the single components, i.e. binary (source packet) files and ASCII (calibrated parameter) files, and the interfaces between the components, such as direct inter process communication to allow fast and reliable fully automated processing throughout the complete offline processing chain.

More detailed descriptions of the in Fig.1 presented components will be given in the following.

The configuration for the whole processing chain is generated in an automated way as far as possible. There are some manual adaptations necessary which are caused by the variety of products to be generated, such as FD products, which are completely out of the common processing cycle.

The Configuration ToolBox (CTB) is implementing the tasks necessary for the setup of the MOPS components. This comprises dedicated configuration settings for single components as well as general configuration settings that are affecting two or more components. E.g. running the system in multi-processor mode requires the configuration of the Frame and Packet Processor (FPP) with respect to the distribution of the extracted telemetry source packets to the appropriate packet processor which is located in the FPP component. This component has mainly to be configured with the according SCOS2000 mission information database (S2K MiB), where all telemetry frames and packets are defined with respect to their identification, primarily the SCOS Packet Id (SPID).

The Monitoring and Control ToolBox (MCTB) has a genuine need of the parameter definitions and their allocation to the telemetry source packets. Furthermore all offline processing specific setups are provided by the CTB to the MCTB, e.g. the assembly of the offline ASCII files or the distribution of the single independent source packets (SPIDs) to their dedicated processors, when running in multi-processor mode.

Last not least, the Post-Processing ToolBox is also supplied with all necessary configuration settings by the CTB. In this case, mainly the generation of the final products is concerned, which are in wide parts (as far as possible) derived from the generic output format of the MCTB. One example would be the generation of the On-Board Event Product (OBEH), which can directly be deduced from the generic ASCII packet product files of the MCTB.

### **III. Pre Processing**

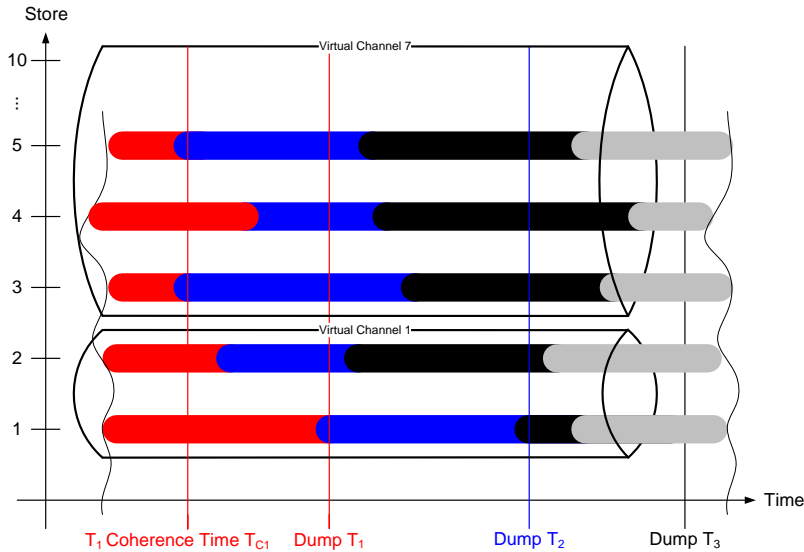
#### **A. Pre Processing Implementation**

The tasks specified with the frame processing requirements will be carried out by one (or more) processes called Frame and Packet Processor (FPP). The already existing frame and packet processor has been extended in the way that it is supporting all necessary functionality at least for the TDX mission that is using the CCSDS (Consultative Committee for Space Data Systems) recommendations for telemetry transfer frames and telemetry source packets.

During the TDX mission, up to 10 housekeeping data stores can be used. They are highly configurable concerning contained data, i.e. the telemetry packets stored in the single stores and the way they can be sent (dumped) to the ground. E.g. it can be specified which combination of stores shall be dumped, how many data words shall be dumped and which virtual channel (VC) is to be used for the corresponding dump. For TerraSAR-X currently only VC1 and VC7 are used for dump data, where VC1 is carrying “high priority” data, which shall be processed first to accommodate the operational requirements to assess the satellite’s status as early as possible after each HK dump. For real-time data received during contacts, VC0 will be used for the TerraSAR-X as well the TanDEM-X missions. The following graphic (Fig.2) shall give an overview of the operational scenario.

As to be seen in Fig.2 the highly configurable Housekeeping (HK) dump mechanism implies several complexities to process the received data. Hereby, the VC de-multiplexing and processing of the standard CCSDS telemetry frames and packets are some of the easier tasks. One of the challenges is to provide a coherent stream of data concerning source packets and the therein contained parameters to allow a consistent processing of parameter, especially of derived (synthetic) parameters, that are dependent on input parameters that possibly (and actually) are contained within different source packets or even are transported via a different virtual channel. Atop this complexity there is the time-independence of the commanded dumps. Dumps are performed solely using so called store pointers which are only reflecting an absolute position of binary words in the corresponding on-board-store. This is also shown in Fig.2. Dump  $T_1$ ,  $T_2$  and  $T_3$  shall indicate different dump initiations that are dumping data from the on-board stores 1 to 5. Each time, the dumps of the single stores are completely independent of each other. This means, to a specific time the overall data can only be completely coherent until the earliest time one of the on-board store dumps has finished (see  $T_1$  Coherence Time  $T_{C1}$  in Fig.2).

To overcome the complexities described in the previous paragraph, a (short-term) archive/store for the extracted telemetry (TM) source packets has been implemented according to the requirements for 24h dump data processing. This archive is filled directly by the frame and packet extraction sections of the FPP or the MCTB.



**Figure 2. Housekeeping Memory Scenario**

Concerning grouped variable packet, the MCTB will provide the according input files for the VPP for all variable source packets that are foreseen for further processing. The configuration of this is done by using the according S2K MiB table (namely PVD).

Finally, to break down the main tasks of the FPP three constituting components are identified (see also Fig.1: MOPS Concept). The Telemetry Frame Processor detects data units (DUs) containing CCSDS TM frames according to the specification of the input file formats and performs a virtual channel de-multiplexing and handling. CCSDS idle and data frames are detected and processed accordingly depending on their content in the context of the TSX/TDX mission. A correctness check of the received TM frames via Cyclic Redundancy Check (CRC) is being performed and frame counter discontinuities for each virtual channel are detected. The FPP is also responsible for the correct handling of spill-over packets, that means TM source packets starting at the end of one TM frame and ending at the beginning of the subsequent TM frame are re-assembled.

The Telemetry Packet Pre-Processor carries out a straightforward extraction of fixed length and variable length CCSDS telemetry (TM) source packets in the context of the TSX/TDX mission, e.g. with respect to assignment of the corresponding SPID. It performs a correctness check of TM packets via Cyclic Redundancy Check (CRC) and a completeness check of the received TM packets by comparing the contained data length field values with the actually received data lengths. The specific TM source packets are identified (SPID) by analysing the contained CCSDS header and sub-header data together with possible additional identifiers. Grouped TM packets are handled correctly, i.e. a subsequent set of TM source packets containing a data portion that is not fitting into one TM source packet. This is done by extracting the packet sequence control section of the CCSDS source packet that is delivered to the corresponding packet processor.

Finally, the Archive Handler generates and handles a short-term archive that contains source packets covering a time period of at least (configurable) 36 hours.

All three FPP components deliver statistical information, generate exhaustive logging information and are configured directly from the SCOS2000 mission information database (S2K MiB).

## **B. Frame Processing**

The tasks of the pre-processing, its input and output, are summarised in the following. It consists of a rather generic input file processing, frame processing, packet extraction, packet archiving, and packet merging. In addition the TDX specific on-board store separation must be performed during these processing steps.

Resulting products are files on raw data level, statistics information, and control information for the subsequent processes.

### **B.1 Generic Frame Processing**

The Generic Frame Processing extracts Virtual Channel Ids (VCIDs) from the CCSDS Transfer Frame (TF) Headers and uses them for further processing. This way it is possible to separate the processing of frames belonging to different VCs when necessary. Used VCs for the TerraSAR-X and TanDEM-X mission are VC-0 for Real Time

data, VC-1 for High Priority Dump data and then VC-7 for Normal Priority Dump data. Having extracted the VCIDs, the de-multiplexed data can be delivered (routed) to their according processing steps, i.e. VC-0 data will be processed different to VC-1 or VC-7 data. An example for the different processing would be the exclusive 24h product file generation for VC-1 and VC-7 data. The FPP generic TF processing is performing TF CRC checks, discontinuity checks on VC counters and TF First Header Pointer (TF-FHP) checks to detect any possible problem on TF level. Actually, frames which fail the CRC are discarded as well as extracted spill-over packets whose continuation is expected in the next TF which fails the CRC, is missing completely or has a FHP that does not match the expected FHP value. Idle frames are counted and discarded since they do not contain any useful data. Source packets are extracted for further processing. For each Telemetry Source Packet reaching over the end of the TF, the expected FHP in the next frame is calculated. Sufficient pre-processing detail is logged for failure analysis. Finally the following statistical information is generated: frames total, frames per VC, failed total CRC, failed CRC per VC, FHP errors per VC, total discontinuities and discontinuities per VC.

## **B.2 TDX specific Frame Processing**

The TDX specific Frame Processing performs the tasks described in this section. The pre-processing processes only transfer frames, which S/C Id corresponds to the S/C name in the input filename. Frames with wrong S/C Id are discarded. The pre-processing is able to recognise TDX PUS (Packet Utilisation Standard) packets. The packet definitions comply with the respective S2K MIB. These packets are identified by their SPID. They carry an additional sub-header with type, subtype and packet generation time. In addition two additional identifiers may be used (I1 and I2). TM Source Packets (SP) undergo a CRC. Packets with failed check are discarded. The pre-processing is able to distinguish both the on-board data stores and contiguous data parts within the on-board stores in the down-linked data dump, assign them individual Ids, and process them separately. The splitting can be enabled or disabled for virtual channel (VC), on-board store and time jumps. The resulting split products are distinguished by their corresponding file names. Pre-processing statistics is generated for the complete data file on TF and SP level, and all data parts represented by split and the applied split criterion. When PUS packets are encountered, which are not configured in the MIB ("pid.dat"), an according log message is written, and the statistics updated accordingly. A binary file is output with all so configured raw SPs, contained in the data part of the corresponding TFs. The filename additionally shows the on-board generation time of the first and of the last SP. Binary files for configurable subsets of source packets are generated. It is possible to configure two output modes, all-in-one and single. In the all-in-one mode all configured packets are written to one file. In the single mode, each configured packet is written to a separate file. The filenames show additionally the configuration name, respectively the SPID, and the first and last packet times. Packets are configured via their SPID. Raw grouped packet files (\*.grp) are written to the output, for each configured SPID. For each completely received group of packets, a new file is generated. Each packet is preceded by a specific header. Each incomplete raw grouped packet file will be discarded after processing. The output is subject to the offline VPP processing.

## **B.3 24h Raw Packet Archives**

The 24h Raw Packet Archive performs the following tasks. For the 24h offline processing mode, the MOPS processing produces a set of raw data files. Each set again contains a particular coherent set of TM source packets – a split. At the moment the criteria for a dump split are VC, on-board HK store and time coherence, i.e. the packets must have been sent via the same VC, must originate from the same HK store and finally must have been generated in a coherent time period. The raw data files consist of chronologically concatenated binary TM source packets of the same SPID. For the generation of products, which cover a larger time range and/or require an extended input packet composition, the archive files are indexed and can be processed in the appropriate way. The 24h packet archive behaves like a rolling archive, i.e. has a configurable life time (typically more than 24 h). This means after the configured time range the older files are replaced with the actual ones. The required 24h products are generated using the Raw Packet Archive and selecting the desired time range. Hereby, the start time (the raw packet time of the first TM source packet) and the time range can be configured easily.

## **IV. Main Processing**

The tasks of the main offline processing, its input and output, are summarised in the following. The main offline processing has been implemented straightforward to reduce both processing and configuration complexity and to increase reliability, robustness, maintainability and intuitive understanding for operators.

Output is calibrated parameter files in a standard format, binary packet files and binary grouped packet files. Special products – such as Telecommand Acknowledge History (TCAH) – are derived directly of these products.

## **A. Offline Packet and Parameter Processing**

The main processing runs completely automatically. It is easy to feed in new dump files into the processing chain. The input files are directly accessed and if necessary copied or moved to the appropriate processing location. It is possible to invoke special processing for specific dump files, e.g. depending on the virtual channel and it is possible to invoke additional special tasks such as acknowledgment processing, depending on the packet content of the input file. According to the delivered dump files the offline processing accesses source files and processes them sequentially. For each dump file a distinguished set of outputs is generated. For processing tasks, which span over the period of more than one dump, the merged packet archive, or the archived day files are accessed. The time range of the dump spanning products is freely configurable via configuration file.

In the following the tasks performed by the TM source packet processing are described. For each recognised source packet the Cyclic Redundancy Check (CRC) is evaluated. If the packet CRC check fails the packet is discarded and will not be processed further. The offline parameter processing is generally able to perform the same functions as the real time system with respect to parameter extraction and processing. The source packet processing is defined by the information contained in the MIB as used for the real time system. The source packet processing performs parameter extraction and processing for the parameters contained in the packet, including the derived parameters. Only parameter required in the output files or which are necessary for derived parameter processing, need to be processed. The composition of the output files is arranged as described in an additional MIB table (TMPT). The TM source packet processing performs splitting of output files belonging to one packet into several files when specified in the TMPT. The generated output shall be ASCII files with the calibrated parameter values. During packet processing, statistics is generated, with total packet count, number of packets per SPID, number of unknown packets, number of failed CRC, etc.

The features supported by the parameter processing are described in this section. In general, MOPS provides the same functionality for the parameter processing in the real time S2K system. All processing types, parameter attributes and out-of-limit processing, defined in the S2K MIB are supported. The functions defined in the S2K MiB are simply translated to the MOPS functions. Hard coded parameters can be simply transcoded to the MOPS database format and embedded in processing by including a special sub-database. As a base line, the offline processing is driven by the project database which is the S2K MIB (e.g. as used for TDX). The pre-processing (translation) of the S2K MIB for MOPS is performed fully automated via scripts. All information contained in the S2K-MIB is extracted and made available to the offline system. In the case certain parameters are not needed for the offline processing and analysis, these parameters are automatically excluded from the MOPS database by the MiB translation script. Additional configurations settings are handled via extra tables added to the MIB and are taken into account by the database pre-processed for the offline system. An example would be the TMPT table defining the composition of the standard output products. The MOPS database is generated fully automatically and therefore easily adapted to handle frequent updates of the database. The MOPS is able to process derived parameters in very much the same manner as the real time system, e.g. S2K static input parameters are handled as such by the MOPS. The products generated by the main offline processing are offline standard and special products. The former are the CX and CP ASCII product files, the latter concern more specialised products such as the FD AOCS or extracted HK parameter files.

## **B. Main Processing Implementation**

For the Main Processing part of the MOPS, an existing and operational package is reused, the Monitoring and Control ToolBox (MCTB). The MCTP is an improved version of the M&C processor still in use within e.g. DLR's GRACE mission. Another evolution of the M&C processor is in use for DLR's Columbus project infrastructure.

Highlight features of the MCTB relevant for the MOPS are described in the following. It is highly portable, i.e. it is currently running on WinXP and SLES10. The MCTB is database driven: its configuration is directly extracted from the S2K MiB. It is supporting all S2K parameter types and attributes. The MCTB has a powerful operations language that exceeds the S2K OL's capabilities by far.

Similar to the FPP the TDX MCTB can be structured into three modules, dividing the required tasks into appropriate subsets (see also Fig.1: TDX MOPS Concept).

The Packet Processor detects packet parameters within the TM source packets as specified by the Mission Information Database (MiB) and extracts the packet parameters according to their data types given with the MiB, such as 16 bit integer.

The Parameter Processor calculates the parameter raw and calibrated (engineering) values and additionally the derived parameter values.

The Product File Generator generates all products according to the underlying interface control documents (ICDs).

CRC and completeness check of the Transfer Frames and TM source packets are already performed by the FPP. This reduces the processing load of the MCTB significantly and guarantees optimal performance during source packet processing and (generic) product file generation.

In order to reduce the processing load MOPS is taking into account the TMprint table (TMPT, additional to the S2K MiB) to optimise the overall processing time. This is done by only taking these parameter values into account that are actually needed for offline analysis.

### **C. Further Processing and Special Tasks**

As MCTB natively is supporting out of limit (OOL) calculations, OOL product files can optionally be generated directly. At the moment this is done by an external process based on offline CX and CP files.

The tasks performed of the OBEH and TCAH Processing are described in this section. It generates an On Board Event History (OBEH) and a Tele-command Acceptance History (TCAH) output from input raw packet files, which contain the corresponding packets. The processing is functionally identical to the processing in the S2K real time system. A text output file is generated for the processing, which is identical to that generated by the processing of the real time system. For the offline processing the packet generation time in the sub-header is used as time stamp in contrast to the real time system using ERT. The processing combines inputs from all relevant sub-systems, i.e. is generating a compilation of all telemetered on-board events resp. command acknowledgements.

Finally, the tasks performed of the Variable Packet Processing are described in the following. The VPP detects variable CCSDS TM source packets driven by the MiB as well as variable packet grouping, i.e. analysis of TM source packet header grouping flags. It generates variable packet binary files (\*.grp) containing complete variable packet groups and discards incomplete variable TM source packet group files.

## **V. Post Processing, Distribution, Automation and Monitoring**

As the Post Processing main task consists of editing the product files generated by the MCTP, this task will be performed by appropriate scripts that are purpose-built for such functionality. Besides that scripts are easy to maintain and adapt to new interface requirements.

The MOPS post-processing part performs additional format conversions or further processing on above mentioned products, e.g. the preparation of products for archiving or selective packing of products for delivery to certain recipients. The MOPS post-processing supports the delivery and archiving of products by placing them at specific locations for pick up by automatic distribution systems. The post-processing is capable to reduce the standard outputs by outputting only every n-th line or by outputting lines only if one or more parameters of a configurable list have been changed.

The tasks of automation and process monitoring are targeting mainly on reducing the necessity to attend the offline processing system and the time for troubleshooting. All parts of the offline system run in an unattended fully automated mode. This spans from pre-processing, over main processing to post-processing, i.e. once the system is configured and started no further manipulation is required. MOPS is providing an appropriate monitoring of the system status, processing steps and generated products. The MOPS processing flow is configurable to a high degree.

## **VI. Conclusion**

The requirements – that are directly derived from the currently operated offline system – have been implemented in all areas where feasible. The result is a highly flexible and generic offline processing system which does not only support contemporary missions like TerraSAR-X and TanDEM-X, but also serves as a basis for future offline processing systems.