Progetto
(project)

MDPS SYSTEM

Multi-Satellite Data Processing Systems

Titolo

(Title)

ERS Browse Document

Version 1.0

	NOME - FUNZIONE (NAME - FUNCTION)	DATE	SIGNATURE
Preparato da (prepared by)	V. Spaventa	9/09/98	
Verificato da: (Verified by)	A.Vollono	15/09/98	
Approvato da: (Approved by)	R.Medri	15/09/98	
Autorizzato da: (Authorized by)			

REGISTRO DELLE VERSIONI
(DOCUMENT CHANGE LOG)

Edizione (Issue)	Date	N° di modificazioni (Modification number)	Pagine modificate (Modified pages)	Osservazioni Observations
1.0	9/09/98		all	First version

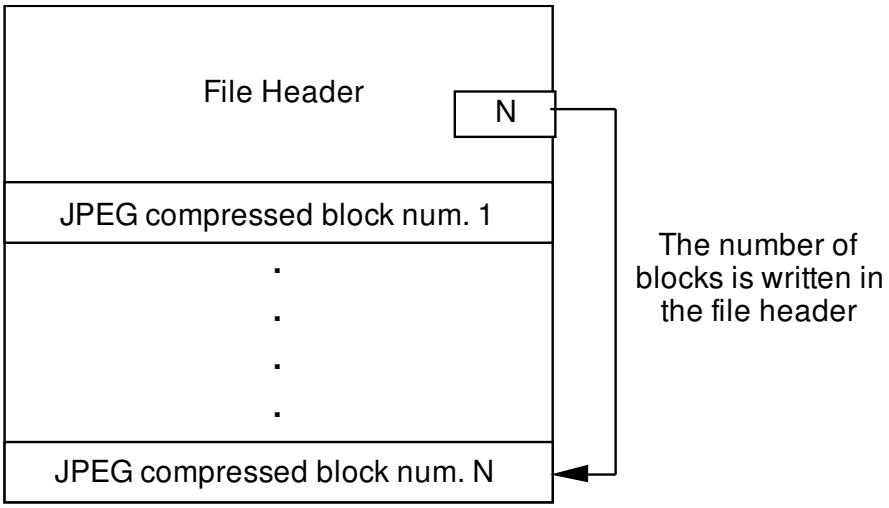
Table of Contents

1. ERS BROWSE PRODUCT

The ERS Browse Product (EBP) is composed by a Quick Look Image File (ERS Browse Image file) and a corresponding Inventory File. The Browse Image file has the extension ".jpeg", while the Inventory file has the extension ".inv". The structures of such files are described hereunder.
The EBP is generated as a post processing of the ERS Screening function; Quick looks of the SAR data and auxiliary files generated during the execution of the Screening function are processed to generate the EBP.

1.1. ERS BROWSE IMAGE FILE STRUCTURE

The file is organised with a first record (header record) containing general information, addresses and sizes of the successive compressed blocks. The successive records (block records) contain compressed data. Each block is generated from the compression of a fixed number of lines (256). JPEG compression methods are applied. The number of lines is selected to optimise the floating display of the data when retrieved. Pointers to standard frames are stored in the inventory file, for each frame. Format of the header record is given in paragraph 4.2.1



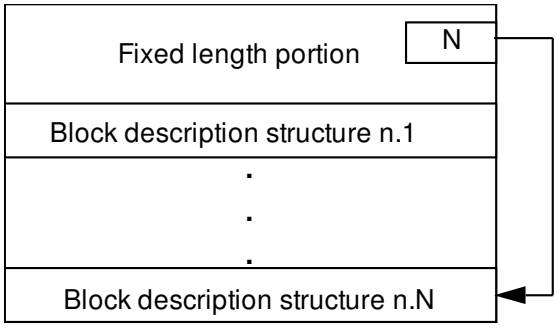
ERS Browse Product - Browse Image File (.jpeg)

The Jpeg compression software library used for the EBP generation allows the definition of the quality of the processed image, through a parameter ranging from 1 to 20. As an example, with a value of the parameter set to 20 the compression software will process the image minimising the losses (highest quality). The compression ratio in this case is very low (around 3). For the EBP preliminary tests we set the parameter to 15, thus obtaining a compression ratio around 8.

With such a selection of parameters, the process reduces the image file containing the quick look image to a size not exceeding 3 Mbytes, for the maximum length of the acquired pass (10 minutes).

1.1.1. EBP Image File Header

The EBP Image file header is a variable length data structure, written at the beginning of the EBP image file. The data structure consists of a fixed length portion (containing identification information) plus a block description portion, containing addresses for each block of compressed SAR data. The block description structure is repeated as many times as the number of compressed blocks.



The fixed portion of the header file and the block description structure formats are shown below.

Field	Type	Length	Position	Name	Comments
1	long	4	01-04	MagicNumber	UNIX file type code
2	long	4	05-08	Video_Format	Format of the Video Data: 1 = Black and White (BW) Format 3 = Red, Green, Blue (RGB) Format For ERS SAR data always set to 1
3	long	4	09-12	Line_Size	Length in pixels of the EBP Line
4	long	4	13-16	Lines_Number	Number of Lines of the decompressed EBP file
5	long	4	17-20	Lines_per_Jpeg_Block	Number of original lines in each Jpeg Compressed Block
6	long	4	21-24	JPEG_Block_Number	Number of JPEG Blocks
7	long	4	25-28	Lines_per_Last_Jpeg_Block	Number of Compressed Lines for the last Jpeg Block
8	long	4	29-32	Padding_at_segment_start	Number of Black Lines inserted at the beginning of the EBP decompressed file
9	long	4	33-36	Padding_at_segment_end	Number of Black Lines inserted at the end of the EBP file
10	float	4	37-40	PixelSizeX	Pixel Size (X direction in meters)
11	float	4	41-44	PixelSizeY	Pixel Size (Y direction in meters)

JPEG block addresses

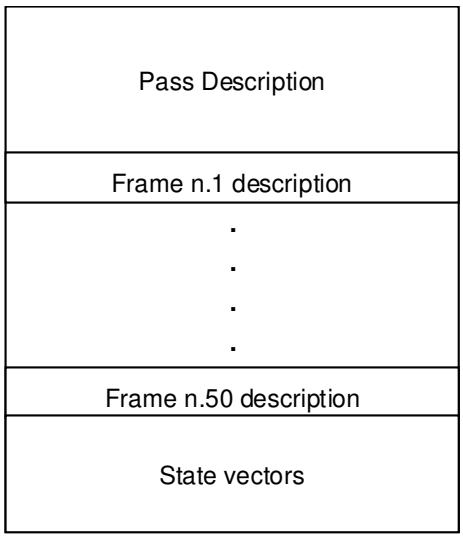
The following structure is repeated as many time as the number of block composing the Browse image file (.jpeg).

1	long	4	01-04	Jpeg_Block_Start	Address of the block from file beginning (0 relative - in bytes)
2	long	4	05-08	Jpeg_Block_Size	Size of the compressed block (in bytes)

1.2. INVENTORY FILE STRUCTURE

The inventory file contains the description data for the acquired segment. The file contains three types of data structures:

- a segment description data structure, containing information valid for the whole segment,
- a frame description data structure, containing standard frame related information,
- a state vector data structure, containing orbital data valid for the segment.



ERS Browse Product - Inventory File (.inv)

The frame description data structure is repeated 50 times in the record, being 50 the maximum number of frames the satellite can acquire in one orbit. This space is always occupied in the file, even though the number of frames really acquired is lower. Only the first N frames structures are filled with information, where N is the number of frames really acquired. The number of acquired frames (N) is given in field 83 of the segment description structure.
The frame description is also provided for frames partially covered by the segment (this occurs for the first and last frame of the segment).

Format for the three data structures is given hereafter.

1.2.1. Segment description

The Segment description area is divide in two main parts: the "LOCALISATION_SEGMENT" and the "SEGMENT_ERS, which,in turn, are divided in variables and sub-structures.
The segment description structure is described hereunder. It is worth noting that in some points of the following description, the number of bytes effectively used for a given type of data exceeds the theoretical one. This is due to the padding rules used in the standard C structures.

Field	Start	Stop	Bytes	Type	Description
NumOfVertex	1	12	12	long	Physical Address on Tape of the Block
Lon	17	20	4	float	Vertex #1 Longitude
Lat	21	24	4	float	Vertex #1 Latitude
. . .					
Lon	809	812	4	float	Vertex #100 Longitude
Lat	813	816	4	float	Vertex #100 Latitude
MediumType	817	828	12	char	HD-96, DIR 1000, etc.
MediumId	829	840	12	char	Station dependent
OrigMediumType	841	852	12	char	Only for transcription
OrigMediumId	853	864	12	char	Station dependent
NumOfPasses	865	868	4	long	Passes on original medium
TimeCodeType	869	876	8	char	IRIG A, IRIG B, NASA 36
StorageStation	877	880	4	long	Centre where medium is currently stored
MediumLoc	881	892	12	char	Station dependent
MediumSpare	893	912	20	4*long	
NPass	913	916	4	long	Passes within storage medium
AscendingFlag	917	920	4	long	0 =desc 1=asc
SatId	921	924	4	long	Set to 5 for ERS
SatMis	925	928	4	long	1 - 2 ERS
SensId	929	936	8	long	Set to 10 for ERS AMI SAR
BegRecordDate	937	944	8	double	Julian Date - Pass recording start time
EndRecordDate	945	952	8	double	Julian Date - Pass recording stop time
Orbit	953	956	4	long	Orbit number
StartBlock	957	960	4	long	Block number of pass start
EndBlock	961	964	4	long	Block number of pass end
StartFeet	965	968	4	long	Feet number of pass start
EndFeet	969	972	4	long	Feet number of pass end
FirstAddress	973	976	4	long	Address for pass start on transcription medium
SecondAddress	977	980	4	long	Address for pass header on transcription medium
ReceiveStdRec	981	984	4	long	Acquisition station, listed in Station.txt
SegNum	985	988	4	long	Num of valid segments for SPOT, ERS
Cycle	989	992	4	long	Satellite cycle in days
ProcStation	993	1000	8	long	Station that generated inventory. Coded as

					acquisition one
dBInsertDate	1001	1008	8	double	Date of the record insertion into local catalogue
Version	1009	1020	12	char	Version of inventory program
Passspare	1021	1056	36	long	
SegmentOrder	1057	1060	4	long	Number of present segments in the sequence of acquired segments
RollAngle	1061	1064	4	long	SAR incidence angle
BegTimeCod	1065	1072	8	double	Segment start time
EndTimeCod	1073	1080	8	double	Segment stop time
BegFormat	1081	1084	4	u_long	Segment start format counter
EndFormat	1085	1088	4	u_long	Segment stop format counter
ICUOnBoardBegT	1089	1092	4	u_long	Satellite binary counter at segment start
ICUOnBoardEndT	1093	1096	4	u_long	Satellite binary counter at segment end
ILatMin	1097	1100	4	float	Lower latitude coded as degrees decimal
ILonMin	1101	1104	4	float	Lower longitude coded as degrees decimal
ILatMax	1105	1108	4	float	Upper latitude coded as degrees decimal
ILonMax	1109	1112	4	float	Upper longitude coded as degrees decimal
CompressionMode	1113	1120	8	char	Coded as "OGRC$$$$" or "OBRC$$$$"
FirstFrameNum	1121	1124	4	long	Number of first standard frame
LastFrameNum	1125	1128	4	long	Number of last standard frame
Spare	1129	1136	8	2*long	
PulseRepInt	1137	1144	8	double	Pulse repetition interval (unit Hz)
SamplingRate	1145	1152	8	double	Range Sampling Frequency
CalibSubAtt	1153	1156	4	long	Calibration Sub-attenuation
ReceivGain	1157	1160	4	long	Receiving gain
Ellipsoid	1161	1168	8	char	Ellipsoid name
EllipsParam	1169	1184	16	float	Ellipsoid parameters (radius, flattening)
NoiseFlag	1185	1188	4	long	Noise Availability
SWSTFlag	1189	1192	4	long	SWST Availability
CalibFlag	1193	1196	4	long	Calibration Values
QualityFlag	1197	1200	4	long	Quality Estimation
DopplerFlag	1201	1204	4	long	Doppler Data Availability
QLFlag	1205	1208	4	long	Quick Look Availability
HistogFlag	1209	1212	4	long	Histogram Availability
BegFormatNoise1	1213	1216	4	long	Start format of noise pulse (Segment start)
EndFormatNoise1	1217	1220	4	long	End format of noise pulse (Segment end)
BegFormatNoise2	1221	1224	4	long	
EndFormatNoise2	1225	1228	4	long	
BegFormatCalib1	1229	1232	4	long	Start format of calib pulse

EndFormatCalib 1	1233	1236	4	long	End format of calib pulse
BegFormatCalib 2	1237	1240	4	long	
EndFormatCalib 2	1241	1244	4	long	
CalibFileName	1245	1308	64	char	File name of associated calibration data
NoiseFileName	1309	1376	64	char	File name of associated noise data
SampleTChang e	1377	1384	8	long	Number of SWST changes (Max 20)
ChangTimeValu e	1385	1392	8	double	Values of sampling window start time #1
. . .					
ChangTimeValu e	1537	1544	8	double	Values of sampling window start time #20
ChangTimeFor mat	1545	1548	4	long	Format counter where change occurred #1
. . .					
ChangTimeFor mat	1621	1624	4	long	Format counter where change occurred #20
DCentrMeasure s	1625	1632	8	long	Num of computed Doppler centroids (Max 50)
DCentrValue	1633	1640	8	double	Value of the Doppler centroid #1
. . .					
DCentrValue	2025	2032	8	double	Value of the Doppler centroid #50
DCentrFormat	2033	2036	4	long	Format counter of Doppler Centr. computation #1
. . .					
DCentrFormat	2229	2232	4	long	Format counter of Doppler Centr. computation #50
NOfMissingLine s	2233	2236	4	long	Number of missing lines in the whole segment
OverallQuality	2237	2240	4	long	Overall segment quality estimation
QualityDensity	2241	2244	4	long	Num of lines of validity for each value
QualityVotes	2245	2500	256	u_char	256 quality estimations - see explanation below
QLBavFileNam e	2501	2564	64	char	File name of the associated Browse image file
HistFileName	2565	2628	64	char	File name of the histogram
NumOfFrames	2629	2632	4	long	Number of frames (standard frames - includes also partial ones)
PaddLinesBegF	2633	2636	4	long	Num of black lines added at QL file start to

F					complete the 1st standard scene
PaddLinesEndLF	2637	2640	4	long	Num of black lines added at QL file end to complete the last standard scene
BPID	2641	2660	20	char	Browse Product ID for MMBS Database link
SegmentSpare	2661	2696	36	long	

Julian Date.

The Julian date is expressed in days (and decimal fractions of days) since January 1st, 1950. It is contained in a double floating point variable (8 bytes). January 1st, 1950 is day 0.

As an example the value 16362.046313

means 19-OCT-1994 01:06:41.451.

Resolution is 1 millisecond. For the computation of the day fraction, the day is considered to be composed of 86,400,000 milliseconds.

Explanation for the Quality Density field.

Regardless the length of the segment, 256 votes equally distributed along the segment are given. The interval of validity (Quality Density field) of each vote can be obtained by dividing the total number of lines of the segment by 256. As an example, the Quality Density field set to 3000 means that the first quality vote is valid for lines from 1 to 3000 of the segment, the second from 3001 to 6000, ... Line number are given for the input raw data stream.

Explanation for the Quality Vote field.

Each quality vote is derived from the number of missing lines detected in the interval of validity. To fit in one byte the number of missing lines is scaled to a max of 255. The scale factor is fixed for all the values, and is computed as the ratio of the Quality Density (Interval of Validity) and 256. The ratio is rounded to the closest integer value. As an example, let us assume a Quality Density set to 1200 (segment length about 307200 lines, corresponding to about 3 minutes).

A quality vote 3 means a number of missing lines around 15.

1.2.2. Frame structure

The frame structure contains information on the frame location, timing, quality, plus addresses of the frame within the ERS Browse image file associated to the inventory record. Here follows the description table. It must be repeated that, like in the previous section, in some points of the following description, the number of bytes effectively used for a given type of data exceeds the theoretical one. This is due to the padding rules used in the standard C structures. The byte numbering is "Segment_Description relative", i.e. is the numbering of the first frame. The addresses of the successive frames must be correspondingly shifted.

	Field	Start	Stop	Bytes	Type	Description
1	FrameNum	2697	2704	8	long	According to standard ERS framing
2	BegTimeCod	2705	2712	8	double	Scene Start Time -Julian Date format
3	EndTimeCod	2713	2720	8	double	Scene Stop Time -Julian Date format
4	Spare	2721	2728	8	2*u_long	
5	ULLat	2729	2732	4	float	Upper left corner latitude (range -90.0 +90.0)
6	ULLon	2733	2736	4	float	Upper left corner longitude(range-180 +180)
7	URLat	2737	2740	4	float	Upper right corner latitude (range-90.0 +90.0)
8	URLon	2741	2744	4	float	Upper right corner longitude (range -180 +180)
9	LLLat	2745	2748	4	float	Lower left corner latitude (range -90.0 +90.0)
10	LLLon	2749	2752	4	float	Lower left corner longitude (range -180 +180)
11	LRLat	2753	2756	4	float	Lower right corner latitude (range -90.0 +90.0)
12	LRLon	2757	2760	4	float	Lower right corner longit(range -180 +180)
13	MeanI	2761	2764	4	float	Computed on input RAW data
14	MeanQ	2765	2768	4	float	Computed on input RAW data
15	SdevI	2769	2772	4	float	Computed on input RAW data
16	SdevQ	2773	2776	4	float	Computed on input RAW data
17	MissLinPerc	2777	2780	4	long	Percentage of missing input lines (0-99)
18	DopplerCentroid	2781	2784	4	float	Value of the Doppler Centroid valid for the frame
19	BlockNumber	2785	2788	4	long	Numb of JPEG block containing 1st frame line
20	LineNumber	2789	2792	4	long	Position in the block (i.e. number) of the first line of the frame
21	MaxI	2793	2796	4	u_long	Max value of I channel in input RAW data
22	MaxQ	2797	2800	4	u_long	Max value of Q channel in input RAW data

Fields 19 and 20 of the Frame structure can be better explained as follows:

To generate the ERS Browse Image file the intermediate ERS Browse intermediate file (“.bav”) is divided in blocks of 256 lines. Each block is compressed separately and added to the ERS Browse Image file (“.jpeg”). To extract data for a specific standard frame the user must where in the file the frame starts,

being the frame size fixed). Field 19 of the previous structure contains the number of the block where the first line of the frame is located. Field 20 contains the position of such line within the decompressed block.

1.2.3. State Vectors structure

The State Vector structure is added at the end of the ERS inventory file. It must be repeated that, like in the previous sections, in some points of the following description, the number of bytes effectively used for a given type of data exceeds the theoretical one. This is due to the padding rules used in the standard C structures. The byte numbering is "Segment_Description relative".

Field	Start	Stop	Bytes	Type	Description
SVtype	7897	7904	8	int	Type of state vector (0=predicted 1=restituted)
pos_x	7905	7912	8	double	Position (x axis) - unit is km
pos_y	7913	7920	8	double	Position (y axis) - unit is km
pos_z	7921	7928	8	double	Position (z axis) - unit is km
vel_x	7929	7936	8	double	Sat velocity (x axis) - unit is km/sec
vel_y	7937	7944	8	double	Sat velocity (y axis) - unit is km/sec
vel_z	7945	7952	8	double	Sat velocity (z axis) - unit is km/sec
AscNodeJdt	7953	7960	8	double	Time of the state vector (julian date.decimals)
ReferenceJdt	7961	7968	8	double	Reference time (julian date.decimals)
SatBinTime	7969	7972	4	u_long	Satellite binary counter
ClockStepLength	7973	7976	4	u_long	Sat. binary counter period