

The EPS-SG Microwave Imager (MWI) and Ice Cloud Imager (ICI) Test Data release includes the following products and documents:

- MWI L1B Spectral Radiance product
- ICI L1B Spectral Radiance product
- MWI-ICI L2 product providing cloud and atmospheric parameters
- Product Format Specifications (PFS) V3A documents each of the above products and
- Generic Product Format Specifications (GPFS) V3D

All the Products have been generated by the L1 In-House Prototype (v1) and L2 In-house Prototype (v1). The products are available in netCDF format and the Product Format Specification documents gives an overview of the products, their detailed format description including meta data and an estimate of the size per orbit of the data.

MWI and ICI Test data are based on a quarter of the Metop-A orbit number 4655, dated 12/09/2007.

- The ICI L1B - data start time is 08:46:59 UTC, while end time is 09:12:19 UTC.
- The MWI L1B - data start time is 08:47:00 UTC, while end time is 09:12:19 UTC.
- MWI-ICI L2 data - start time is 08:46:59 UTC, accounting for the start time of ICI, while end time is 09:12:19 UTC, according to the end time of MWI and ICI.

The following compressed files are provided for each product.

Product	Filename
MWI L1B	MWI_L1B_CrossRef_TDP_V1.tar.gz
ICI L1B	ICI_L1B_CrossRef_TDP_V1.tar.gz
MWI-ICI L2	USC_MWI-ICI_L2_TDP_v1.tar.gz

Note to Fortran users:

The test data has variable length string attributes, which is incompatible with the FORTRAN API of the netCDF library. Note that variable-length strings are part of the netCDF standard. The FORTRAN netCDF API in its current form does not fully implement the netCDF standard. Although FORTRAN does support variable length strings since 2003, the netCDF API is still aligned with the outdated F90.

Regarding the operational products, a variable-length strings-free implementation cannot be enforced, as there is no requirement (at any level, from user to PDAP) to adhere to any particular API. However, functioning netCDF APIs do exist in several programming languages (e.g. Java, C/C++, Python) and a workaround is readily available for Fortran users (HDF library).