SeaDAS and BEAM User Tools

Data Processing, Analysis and Exploitation Tools

Sean Bailey – NASA GSFC
Norman Fomferra – Brockmann Consult

International Ocean Colour Science Meeting 2013
Darmstadt, 07.05.2013
The BEAM Project

- ESA project kicked off for exploitation of Envisat data
- Open source, agile software development
- Platform neutral, 100% Java

BEAM today
- 24 public releases, hundreds of module updates
- Thousands of users worldwide
- Dozens of projects use and support it (ESA, EU, EUMETSAT, NASA)
- Dozens of supported sensors and data formats, data processors
- Dozens of tools and data processors
- Hundreds of universities, institutes, companies use it

Active user forum (daily posts), frequently visited website, issue tracker, tutorials, manuals
Which once was the “Basic Envisat (A)ATSR and MERIS Toolbox” became a general **EO Toolbox and Development Platform**

- Supported sensors: MERIS, (A)ATSR, ASAR, Chris, AVNIR-2, PRISM, MODIS, AVHRR/3, TM Thematic Mapper, SPOT-VGT, MODIS, SeaWiFS, VIIRS, OCM, ...
- Generic formats: NetCDF/CF, HDF-EOS, GeoTIFF, ENVI

- Derived Toolboxes based on the BEAM Platform
  - NASA SeaDAS 7 Ocean Colour Processing Toolbox
  - ESA NEST & InSAR Processing Toolbox
  - ESA LeoWorks Remote Sensing Training Software
The Tools

- **Visualisation:**
  - Very fast image display and navigation, RGB, colour bars, lots of layer types (masks, GIS layers), fast band arithmetics ... >10 more

- **Processing**
  - Reprojections, GCP rectification, collocation, L3 binning, mosaicing, spectral unmixing, clustering, ...
  - QAA IOP, NN-based AC, FLH-MCI, SST algorithms, and many 3rd party contributions, ... >20 more

- **Analysis:**
  - Flexible mask & ROI management, ROI-based statistics,
  - Interactive scatter-, density-, profile-, histogram-plots,
  - Interactive spectra-, pixel-, flag-, time-series-views,
  - ... >30 more
BEAM User Interfaces

- VISAT: Graphical User Interface
- GPT: Command-line Interface
- API: Application Programming Interfaces
  - EO Data Model
  - EO Application Programming Interfaces
  - EO Rich Client Platform
  - EO Graph Processing Framework
  - Dynamic extensions via plug-in modules
- Open single bands and RGB images
- Colour Manipulation
- Linking displays/views
• Overlaying Masks
• Editing Masks
  • Colour & Transparency
• Creating Masks
  • Flags
  • Geometries
  • Vector Data
  • Value ranges
  • Mathematic expression
• Combining Masks
  • Union, Intersection, Inverse...
• **Pins**
  - Tabular view on band information of dedicated positions

• **Spectrum View**
  - Mouse over
  - Linked to pins (colours and selection)

• **Transect**
  - Pixel values along linear geometry and surroundings
• Masks
  • Editing colours and visibility
  • Definition of new masks
  • Combination of masks

• Statistics
  • Whole bands
  • From Region of interest (ROIs)
  • Selection from Masks list
**Importing vector data**
- Import of vector data in different formats (csv, shapefile)
- Point, line and polygon data
- Editing appearance in layer manager
- Overlay of different bands / images

**Correlative Plot**
- Correlation between EO data and point data
- Regression line
- Variability around in-situ positions
• **Layer Manager**
  - Visibility of layers
  - Transparency of layers
  - Overlay of different bands / images
Processors

- Generic Processors
  - Cluster Analyses
  - Reprojection / Orthorectification
  - Mosaic
  - Collocation
  - Level-3 Binning
  - Pixel Extraction

- Dedicated Processors
  - Radiometric Correction
  - Case-2 Regional Processor
  - ICOL Processor
  - FUB/WeW Water Processor
  - MERIS Case-2 Waters Processors
  - QAA for IOPs
  - FLH/MCI Processor
- **Geo-processing**
  - Subsetting
  - Reprojection
  - Collocation of products
- **L3 binning**
  - Temporal and spatial aggregation
  - Statistics per bin cell
  - Definition of output boundary and bin cell size
- **Mosaicing**
• L2 Water processing
  • FUB WeW Water processor
  • Case 2 Regional processor
  • FLI/MCI processor
  • Lakes processor
• L2 Land processing
  • Vegetations Processors
  • FAPAR processor
  • NDVI processor
Sentinel-3 Reader available
Command-Line Interface - GPT

BEAM Command Line

-T target=<file> Defines a target product. Valid for graphs only. target must be the identifier of a node in the graph. The node's output will be written to file.
-S source=<file> Defines a source product. source is specified by the operator or the graph. In an XML graph, all occurrences of $\{<source>\}$ will be replaced with references to a source product located at file.
-P name=<value> Defines a processing parameter, name is specific for the used operator or graph. In an XML graph, all occurrences of $\{<name>\}$ will be replaced with value. Overwrites parameter values specified by the -p option.

Operators:
- Aatsr.SST Computes sea surface temperature (SST) from (A)ATSR products.
- BandMaths Creates a product with one or more bands using mathematical expressions.
- Binning Performs spatial and temporal aggregation of pixel values into 'bin' cells.
- Collocate Collocates two products based on their geo-codings.
- EMClusterAnalysis Performs an expectation-maximization (EM) cluster analysis.
- Flmhci Computes fluorescence line height (FLH) or maximum chlorophyll index (MCI).
- KMeansClusterAnalysis Performs a K-Means cluster analysis.
- Merge Allows copying raster data from any number of source products to a specified ct.
- Meris.Brr Compute the BRR of a MERIS L1b product.
- Meris.Case2Regional Performs IOP retrieval on L1b MERIS products, including atmospheric correction.
- Meris.CorrectRadiometry Performs radiometric corrections on MERIS L1b data products.
- Meris.GlintCorrection MERIS atmospheric correction using a neural net.
- Meris.Lakes Performs IOP retrieval for eutrophic and boreal Lakes on L1b MERIS products,
- spheric correction.
- Meris.N1Patcher Copies an existing N1 file and replaces the data for the radiance bands.
- Mosaic Creates a mosaic out of a set of source products.
- PixEx Extracts pixels from given locations and source products.
- Read Reads a product from disk.
- Reproject Reprojection of a source product to a target Coordinate Reference System.
- Subset Create a spatial and/or spectral subset of a data product.
- Unmix Performs a linear spectral unmixing.
- Write Writes a data product to a file.
- glint.Flint Flint Processor.
BEAM 4.11

- Released in April 2013
- New Features
  - Interactive Time Series Tools
  - OPeNDAP Access
  - Temporal percentile and gap-filling operator
  - New, faster and more flexible Level-3 binning
  - NetCDF 4 output format
  - All SeaDAS / OBPG input formats (MODIS, SeaWiFS, VIIRS, OCS, etc.)
- Fixes and optimisations
Time Series Tools

- Time Series Player
- Time Series Graph
- Time Series Matrix
OPeNDAP Access
BEAM 5 Plans

- Prototype reader modules for
  - Sentinel-3 OLCI and SLSTR
  - Sentinel-2 MSI, ATCOR Integration
  - (Sentinel-1 SAR through NEST)

- C and Python API
  - Embedding BEAM: Scripting, batch mode processing
  - Extending BEAM: Tools and processors

- “Backport” SeaDAS extensions into BEAM

- Release in Fall 2013
Sentinel-3 and -2 Support in BEAM

- Sentinel-3 Products
  - OLCI L1
  - OLCI Water L2
  - OLCI Land L2
  - SLSTR L1
  - SLSTR Water L2
  - SLSTR Land L2
  - SYN L2
  - VEG L2

- Sentinel-2 Products
  - MSI L1C
  - MSI L2A

- Applicable tools in BEAM
  - Image analysis
  - Layer management
  - Flag overlay
  - Mask management
  - Spectrum view
  - Spectral unmixing
  - Band arithmetic
  - Geo-corrections / -projections
  - Transect profiles
  - Region of interest statistics
  - Time series analysis
  - Mosaicking
  - Level-3 binning
SeaDAS 7.0

Staff:
Aynur Abdurazik
Sean Bailey
Matt Elliott
Danny Knowles
Don Shea
SeaDAS 7.0

- **Objective**
  - Renewal of the "outdated" SeaDAS 6 user interface
  - Away from commercial IDL to an open-source approach
  - Simplify configuration and launching of SeaDAS L[01] - L3 data processors

- **Collaboration**
  - Joint effort of NASA Ocean Biology Processing Group (OBPG) and the BEAM development team
  - Informal meeting at NASA GSFC in May, 2010
    - Decided that *SeaDAS 7 would use the BEAM Development Platform*
  - Bilateral collaboration started in June, 2011
  - Beta Release in June 2012, Final Release April 2013
SeaDAS 7.0 Features

- Exchange of the IDL-based SeaDAS GUI by a frontend based on BEAM VISAT “Rich Client Platform”
- Add BEAM support for OPBP maintained data products
  - Aquarius, CZCS, HICO, MERIS, MODIS, MOS, OCM, OCTS, SeaWiFS, VIIRS
- Integrate SeaDAS’ robust and fast data processing suite
  - NASA operational OC processors used for production
  - greatly simplified usage of SeaDAS data processors (e.g. l1bgen, l2gen, l2bin, etc.
  - sensor-independent approach
  - data processor user interfaces dynamically created from XML files
    - created by the processing programs.
    - Modifying the programs automatically modifies the
  - Linux and MacOSX only, use virtual machine on Windows platforms
### L2gen SeaDAS 7.0

The image shows a screenshot of the L2gen SeaDAS interface with various parameters and options. The interface includes a table with multiple options, such as `giop_adg_file`, `giop_adg_opt`, `giop_aph_opt`, and several others related to parameter settings.

#### Section: IOP Options
- `giop_adg_file`: SOCDATAROOT/common/adg_default.txt
- `giop_adg_opt`: 1 - exponential with exponent supplied via `giop_adg_s`
- `giop_adg_s`: 0.0145
- `giop_aph_file`: SOCDATAROOT/common/aph_default.txt
- `giop_aph_opt`: 2 - Bricaud et al. 1995 chlorophyll supplied via default empirical algor...
- `giop_aph_s`: -1000.0
- `giop_bbp_file`: SOCDATAROOT/common/bbp_default.txt
- `giop_bbp_opt`: 3 - power-law with exponent derived via Lee et al. (2002)
- `giop_bbp_s`: -1000.0
- `giop_fit_opt`: 1 - Levenberg-Marquardt optimization
- `giop_grd`: [0.0949, 0.0794]
- `giop_maxiter`: 50
- `giop_rss_opt`: 1 - Morel f/Q
- `giop_wave`: [412, 443, 488, 547, 667]
- `gsm_adg_s`: 0.02061
- `gsm_aphs`: [0.00665, 0.05582, 0.02055, 0.01910, 0.01015, 0.01424]
- `gsm_aphw`: [412.0, 443.0, 490.0, 510.0, 555.0, 670.0]
- `gsm_bbp_s`: 1.03373
- `gsm_fit`: 0 - Amoeba
- `gsm_opt`: 0 - default coefficients
- `iop_opt`: 0 - None (products requiring a or bb will fail)
- `qaq_adg_s`: 0.015
- `qaq_wave`: [443.488, 547.667]

**Run** | **Cancel** | **Apply**
SeaDAS 7.0 Objectives (cont.)

- Improve SeaDAS/BEAM w.r.t. validation activities
- Improvements to the SeaDAS/BEAM point and vector data support (e.g. support for SeaBASS-formatted files)
- Added a global, high resolution land-water mask
- Will add a global, accurate bathymetry map
- Added auxiliary data management (e.g., download, ...)
- Added a simplified interface to the Color Manipulation Tool
- Odds and ends
  - Add a layer for legends in image views
  - Extend processing capabilities to Windows operating system
  - Add additional user-defined preferences
  - Color manipulation tool preferences
  - Processing option preferences
Thanks for your attention!

→ Don’t forget: You get instant support in the BEAM and SeaDAS user forums.
Architecture Overview

**BEAM VISAT**
- MERIS Smile Correction
- MERIS Case 2 Waterconst.
- Common Orthorectification
- Common Collocation
- (A)ATSR SST Processor
- Common L3 Binning
- Common L3 Mosaicing
- Sensor data formats Envisat, Chris, Alos,...
- Common Spectral Unmixing
- Common Kmeans/EM Clustering
- Common data formats GeoTIFF, NetCDF, ...

**BEAM VISAT RCP**
- Rich client platform

**BEAM UI**
- User interface

**BEAM GPF**
- Graph processing

**BEAM Core**
- Data Model, I/O

**Ceres Core**
- Module Management

**Ceres GLayer**
- Multi-Layer Views

**Ceres Binio**
- Binary data I/O

**JIDE**
- GUI Components

**JFreeChart**
- Plots & Charts

**JAI**
- Tiled Imaging

**Xstream**
- XML binding

**Java SE 1.6 Platform**
VISAT Module Manager

The Level-3 Binning Processor distributes the contributions of a Level 2 pixels in satellite coordinates to a fixed Level 3 grid using a geographic reference system.

Changing:
Generic EO Data Model

Data I/O API
- Reader
- Writer

Data Model
- Internal EO data abstraction

VISAT Rich Client Platform

Processing APIs

Visualisation and Analysis APIs

EO data format abstraction

EO tooling frameworks
Raster Data Management

- Tiled images
- Multi-resolution image pyramids
- Multi-threaded tile processing
- Loading visible tiles, caching invisible
- Allows browsing giga-pixel images
Graph Processing Framework

- Inversion of execution flow, “pull” processing
- Requests are propagated from sink to source: only requested data is processed
- Independent tile computation is parallelised, multi-threading
- Intermediate results are kept in-memory, no I/O overhead
BEAM Success Factors

- Abstraction of EO data products:
  - Generic Product Model (internal representation)
- Abstraction of data input, output, processors
  - Readers, Writers, Operators
- Module-based architecture
  - Every module is a versioned, exchangeable plugin
- Tile-based raster data management
  - Image display
  - Data processing
reflProduct = \nProductIO.readProduct(sys.argv[1])

b1 = reflProduct.getBand('reflec_5')
b2 = reflProduct.getBand('reflec_7')
b3 = reflProduct.getBand('reflec_9')

w1 = b1.getSpectralWavelength()
w2 = b2.getSpectralWavelength()
w3 = b3.getSpectralWavelength()

a = (w2 - w1) / (w3 - w1)
k = 1.03

flhProduct = Product.newProduct('FLH.nc', 'test', width, height)
flhBand = flhProduct.addNewBand('FLH', ProductData.TYPE_FLOAT32)

... for y in range(height):
    b1.readPixelsFloat(0, y, width, 1, r1)
    b2.readPixelsFloat(0, y, width, 1, r2)
    b2.readPixelsFloat(0, y, width, 1, r3)
    print("processing line ", y, " of ", height)
    FLH = r2 – k * (r1 + a * (r3 – r1))
    flhBand.writePixelsFloat(0, y, width, 1, FLH)