LA-UR-16-27377
Approved for public release; distribution is unlimited.

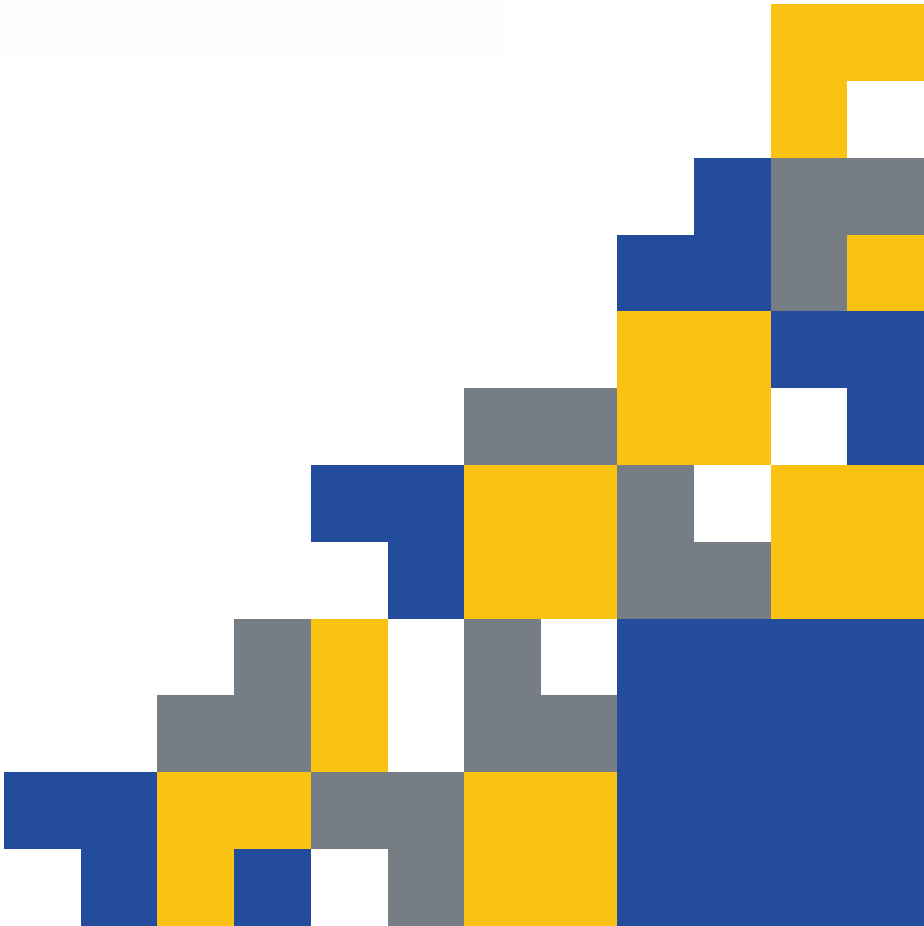
Title: Echo

Author(s): Harvey, Dustin Yewell

Intended for: white paper marketing material

Issued: 2018-05-18 (rev.2)

Los Alamos
NATIONAL LABORATORY
EST. 1943



LA-UR-16-27377

Echo

Echo™ is a scripted data analysis environment that replaces conventional data types such as arrays of numbers and strings with objects that contain data, metadata, and provenance. It is a replacement for traditional scripted data analysis in environments like Python, MATLAB, and R that enables analysts to complete analysis tasks with far less and far simpler code than in traditional scripting environments. By encapsulating metadata with data, Echo removes the opportunity for many human errors in indexing and reporting, and it enables analysts' scripts to indicate their intent much more clearly than in traditional scripting environments. Capturing the provenance of each analysis result produces fully self-describing data records, enabling analysts to share results rather than code. Echo dramatically increases analyst efficiency, preventing errors and saving time.

Echo provides tools that allow users to focus on data analysis and decisions with confidence that results are reported accurately. Echo's extensive feature set allows users to:

- QUERY datasets through name-based indexing for fast retrieval of comparable measurements and results
- SCALE analyses from MB to TB with dynamic allocation of memory and computational resources
- REPORT results accurately and efficiently by populating figure, table, and document content directly from data and linked metadata

- SHARE archival-quality HDF5 files for effective distribution of data and results to varied applications and personnel
- REVIEW analyses workflows directly from results with helpful visualization tools

LA-UR-16-27377

Case Study

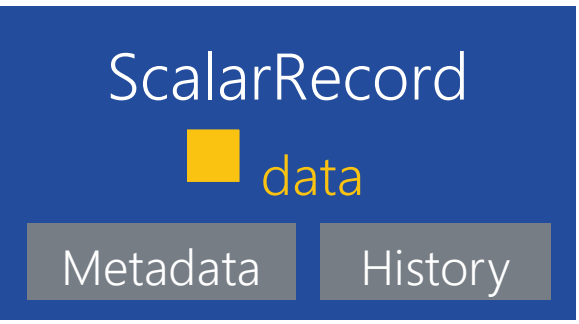
At Los Alamos National Laboratory, Echo is utilized in Weapons Systems Engineering by Life Extension Programs and other programs to organize, analyze, and archive data including mechanical, thermal, and electrical time series and spectral quantities derived from experiments, simulations, or analysis results. For example, environmental specifications for multiple components are derived through Echo involving synthesis of thousands of measurement records from multiple system-level tests. These specifications must be updated periodically to incorporate new datasets, a task that would be laborious and error-prone without the Echo software. Additionally, programmatic decision making involving directed and exploratory analysis tasks are quickly addressed in Echo since datasets can be accessed and compared quickly and accurately. Echo enables analyst collaboration in these programs through all stages of analysis on many types of data using a standard format and framework.

Echo's success in these programs is demonstrated by over:

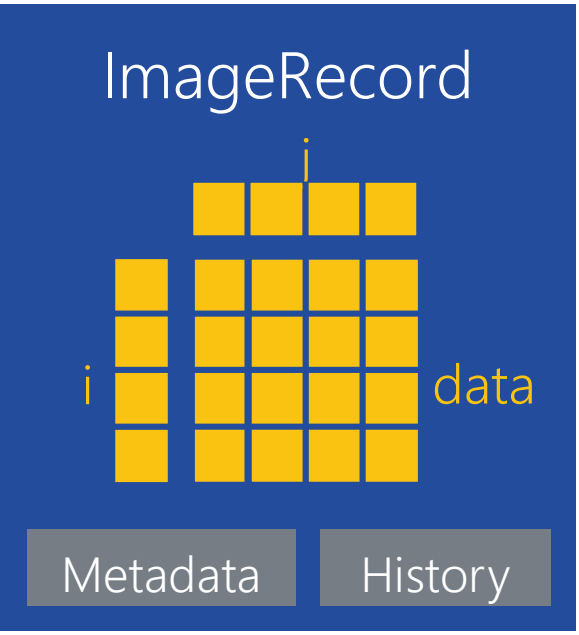
- 1.5 MILLION Echo records
- 15 MILLION searchable metadata strings
- 6 TB of data in EchoH5 format
- 50 official reports produced with Echo content
- 60 Echo users

LA-UR-16-27377

Echo Records

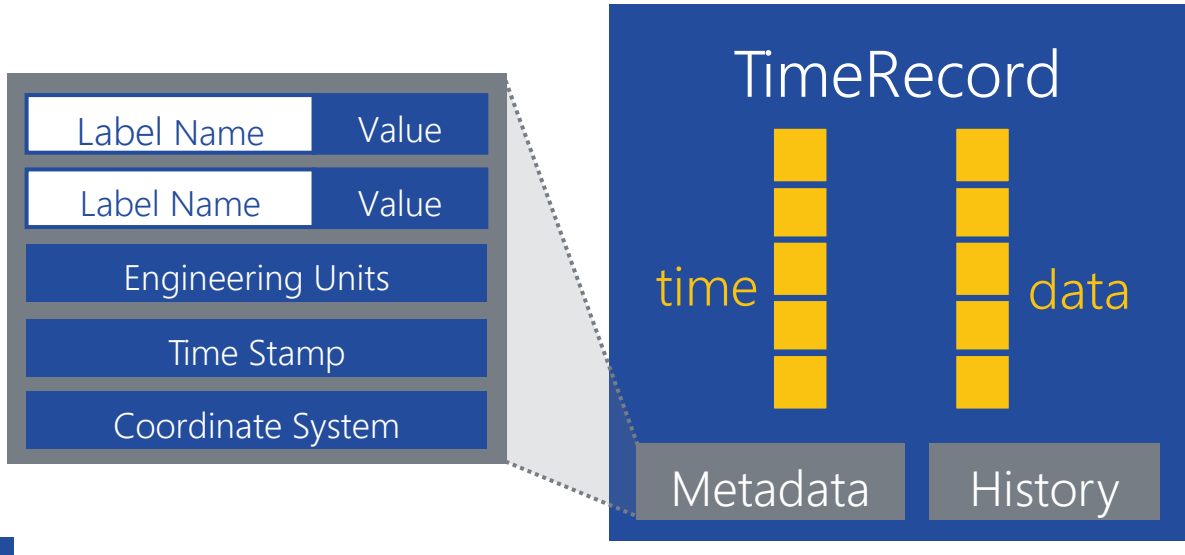


Echo takes an object-oriented approach to data analysis, encapsulating data with all associated metadata. Echo includes many different record classes providing data structures for various types of data. For instance, a ScalarRecord stores scalar-valued data, an ImageRecord stores imagery as a matrix, and a TimeRecord stores time series data. All record types provide the same structure and interfaces for metadata including a user-specified set of labels, engineering units, time stamps, and coordinate system information.



Records in a dataset are indexed by textual metadata reducing opportunities for error and greatly improving code readability. Each user-specified label acts as an index enabling efficient slicing, sorting, and grouping across datasets for instant access to comparable data.

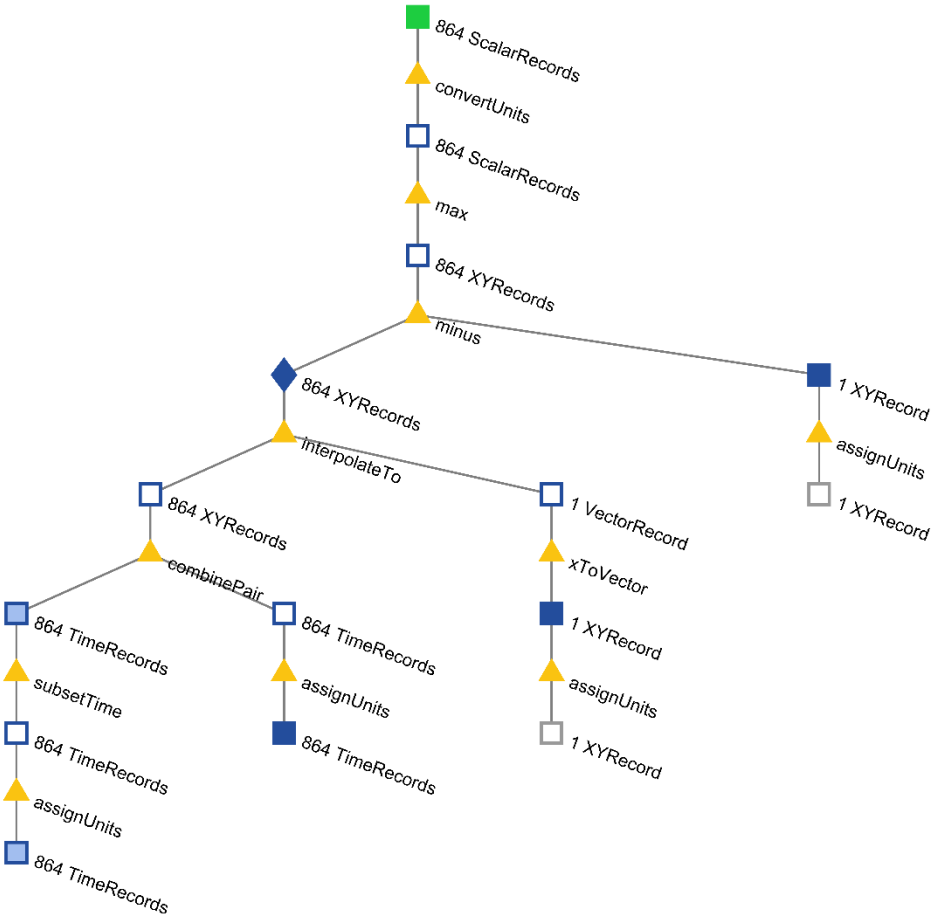
Echo analysis operations include basic math, curve fitting, signal processing, domain transformations, and much more. Each operation automatically propagates metadata from source records to results ensuring the accuracy of analysis products produced through complex workflows. Echo includes a complete engine for engineering units providing propagation of units through analysis operations and conversion and documentation of units for results.



LA-UR-16-27377

Echo History

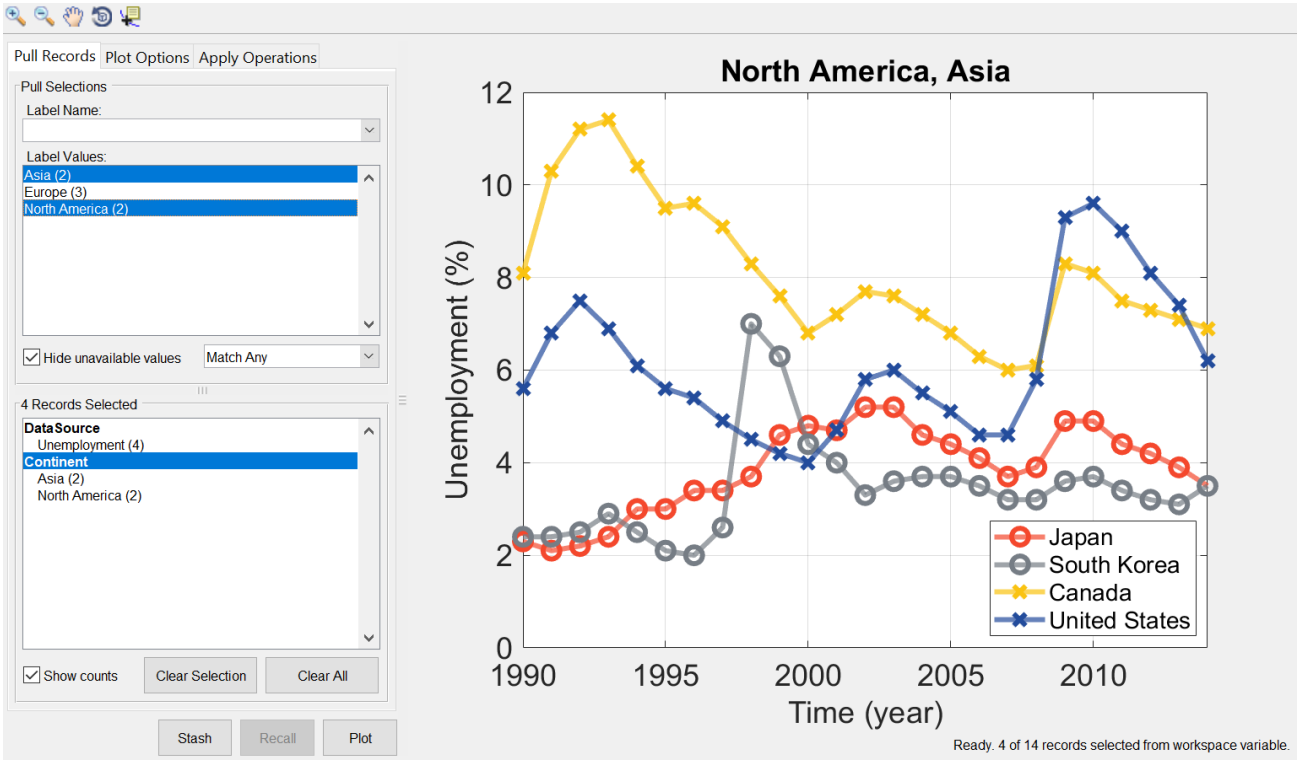
Echo's analysis management systems provide the ability to track, understand, and reproduce workflows used for arriving at results and decisions. The Echo history system provides automatic capture and documentation of operations such that analysis workflows can easily be reviewed. Echo operations capture data provenance including analysis parameters, computing environment information, and authorship. Automated capture of analysis workflow information ensures adequate documentation with minimal analyst burden. Echo history reporting and visualization methods provide auditability of analysis workflows in a standardized format.



LA-UR-16-27377

EchoPlot

Echo includes graphical interfaces for quick exploration and visualization of records. In EchoPlot, records are indexed by metadata to produce meaningful comparisons with minimal effort. The interface includes many visualization options for customization of plot figures. Additionally, basic analysis functionality is accessible directly in EchoPlot to produce on-the-fly results. Echo automatically populates figure content from selected metadata avoiding errors in manual entry and enabling quick generation of accurate analysis products. Similar functionality for faceted search and visualization is also available through a web-browser with the EchoWeb data exploration tool.



LA-UR-16-27377

EchoH5

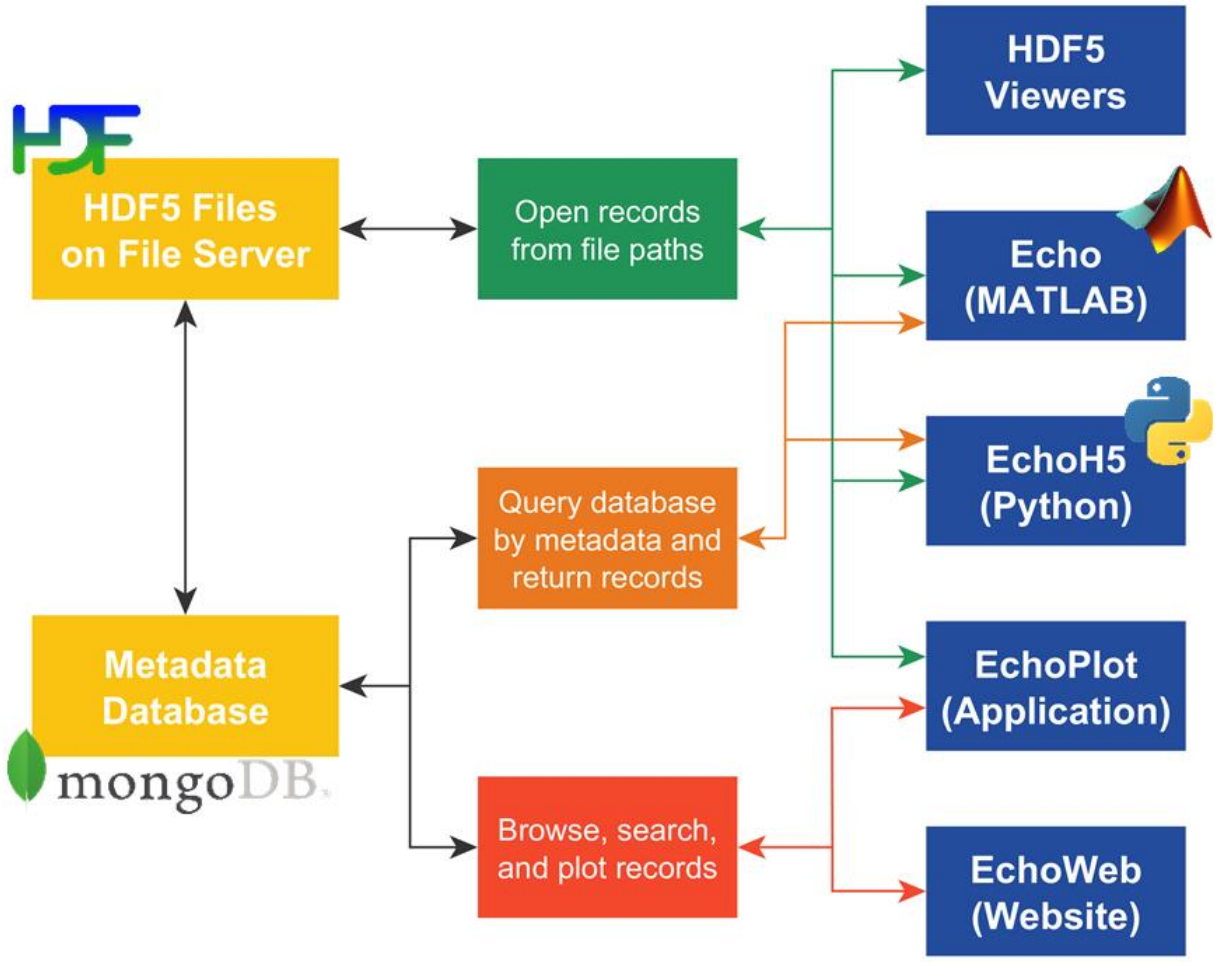
Echo saves datasets in the open-source, self-describing EchoH5 format enabling data accessibility and understanding in perpetuity. A consistent format for many different types of data allows complex datasets to be effortlessly consumed by analysts. The EchoH5 record format uses HDF5 files to store data inseparably with corresponding metadata and history information in a single file preventing mislabeling of results and facilitating data sharing between personnel and across applications.

With existing ubiquitous support for HDF5, the EchoH5 file format allows datasets to be read from a wide array of other languages and applications including R, MATLAB, and Python. The EchoH5 Python package provides access to EchoH5 files as xarray datasets or native Python variables. The package also allows Python users to produce Echo history visualizations from EchoH5 files.

LA-UR-16-27377

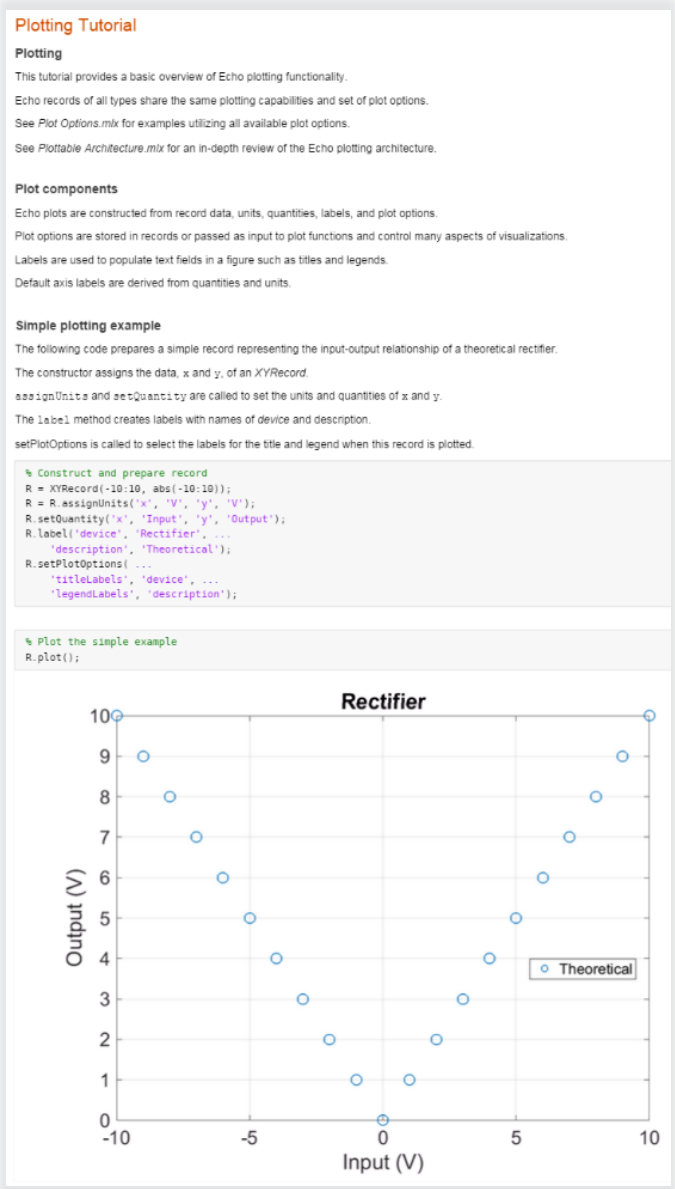
EchoServer

The Echo platform provides a powerful data management and curation solution allowing analysts to quickly find, access, and consume datasets. EchoServer provides a central data management solution for data sharing and analyst collaboration. Records are stored in the EchoH5 format using any file server implementation such as the Hadoop Distributed File System (HDFS) providing file virtualization and redundancy. The EchoServer database provides a searchable metadata cache to support efficient queries such that data can be quickly found across multiple datasets. EchoServer also hosts the EchoWeb tool for web-based data exploration. With direct access to streaming data sources, EchoServer can provide real-time access to curated data.



LA-UR-16-27377

Documentation



Plotting Tutorial

Plotting

This tutorial provides a basic overview of Echo plotting functionality.

Echo records of all types share the same plotting capabilities and set of plot options.

See *Plot Options.mlx* for examples utilizing all available plot options.

See *Plottable Architecture.mlx* for an in-depth review of the Echo plotting architecture.

Plot components

Echo plots are constructed from record data, units, quantities, labels, and plot options.

Plot options are stored in records or passed as input to plot functions and control many aspects of visualizations.

Labels are used to populate text fields in a figure such as titles and legends.

Default axis labels are derived from quantities and units.

Simple plotting example

The following code prepares a simple record representing the input-output relationship of a theoretical rectifier.

The constructor assigns the data, `x` and `y`, of an *XYRecord*.

`assignUnits` and `setQuantity` are called to set the units and quantities of `x` and `y`.

The `label` method creates labels with names of *device* and description.

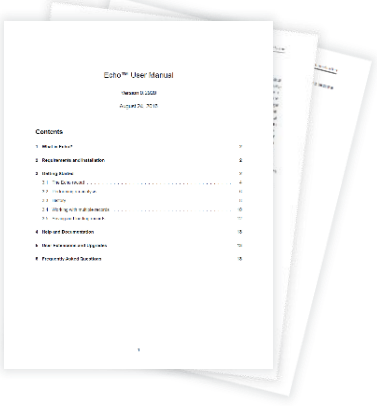
setPlotOptions is called to select the labels for the title and legend when this record is plotted.

```
% Construct and prepare record
R = XYRecord(-10:10, abs(-10:10));
R = R.assignUnits('x', 'V', 'y', 'V');
R.setQuantity('x', 'Input', 'y', 'Output');
R.label('device', 'Rectifier', ...
    'description', 'Theoretical');
R.setPlotOptions( ...
    'titleLabels', 'device', ...
    'legendLabels', 'description');
```

```
% Plot the simple example
R.plot();
```


Echo documentation includes a user manual, code help, and suite of tutorials. The manual includes a basic overview of the software, system requirements, and installation instructions. Code help in the programmatic interface documents individual Echo classes and routines. Tutorials are provided as MATLAB live scripts including extensive examples through interactive, executable code. Each tutorial introduces an Echo component or sub-system with topical discussion and examples.

The Echo development team offers in-person training courses for Echo users. Training covers the basic concepts of Echo records, labels, and visualization tools as well as advanced analysis approaches and use of the EchoPlot interface. Training courses are offered on-site or at one of LANL's computer training facilities.



LA-UR-16-27377

Extending Echo

Echo provides a versatile and extensible framework, allowing advanced users to add support according to their specific needs. Users can create data structures as Echo record classes to bring new data types into Echo. Additional analysis operations are easily added to take advantage of Echo's metadata propagation, history capture, and resource allocation systems. Furthermore, additional visualizations can be incorporated into the Echo graphics system.

Getting Started

Echo is implemented in MATLAB and requires MATLAB R2016b or later.

To begin using Echo's robust, efficient platform for data analysis, contact EchoSupport@lanl.gov.

LA-UR-16-27377