
Data Catcher User's Guide

Ben-chin Cha, Tim Mooney, and Ned Arnold
Argonne National Laboratory
Advanced Photon Source
Experimental Facilities Division/Beamline Control
Draft (Updated March, 1998)

1. Introduction

Data catcher has evolved into version 2.2.2 which supports both 1D and 2D scan and is already been used for general purpose beamline data acquisition at APS.

It is written in IDL[1] programming language and external Channel Access(CA) interface functions defined in the ezcalDL[2] sharable library. It is a complete mouse driven program using the IDL's widgets and event handler.

The data catcher is an easy to use data acquisition tool for EPICS 3.12.2 and newer scan record[3]. Currently it provides 1D/2D real-time plot features while scan is progressing, and automatically saves the scan data at the end of each scan. It provides simple statistic calculation for peak, centroid, and full width half peak (FWHP). It provides plug-in to the simple data fit package. It also provides very flexible 1D/2D post scan data viewing and report features. Simple on-line help is also available.

A special **viewer** package written in IDL is also released with data catcher R2.2.2. For viewing data acquired by the data catcher the **viewer** is recommended because it is more efficient than using the data **catcher**.

Data catcher is available on sun4 and solaris at APS. Although in this report the solaris is used in the system setup example, which is equally applicable to the sun4 system.

With simple setup modification, **viewer** can be easily ported to any other operating system or platform which IDL supports. Porting **catcher** to other operating system requires that the CA has to be ported first.

Overview

Each scan record can contain up to 4 positioners and 15 detectors. Please refer chapter 29 of EPICS Record Reference Manual R3.12 about the detail definitions of scan record. For normal operation of data **catcher**, it requires that the scan record is properly loaded into the input output controller(IOC) and the detector and positioner process variable (PV) names of the scan record are properly set up for the database. Through monitoring the scan record name, the data catcher automatically captures and displays 1D/2D scan data.

The initiation of scan can be internally invoked by the start button of scan setup window of the data catcher or externally invoked by any other CA clients, e.g. **medm**[4].

At the end of each scan only the arrays for the defined (non zero) positioners and detectors are automatically saved. In addition to scan data, a user can use the environment file to save additional operating parameters. At the normal termination of data catcher the new configuration file will be automatically saved. At the restart of data catcher this configuration file will be used.

The default setup for data catcher is acquisition on, automatic save on, real-time displaying on. It operates in two modes: data acquisition and viewing. The program is initially in the data acquisition (i.e. data capture) mode. A user can temporarily select the viewing data mode to examine the scan data previously captured. At the end of view data mode it automatically returns to the data acquisition mode.

The acquisition mode supersedes the view data mode. During the 1D viewing mode if the data catcher detects that the scan is invoked by an outside CA client, it will close the 1D viewing mode right away and yields to the data acquisition mode.

There are three view data modes: 1D, 2D and 1D overlay. The view 1D data mode supports various 1D line plot, plot captions, flexible selection of any detectors, generating post script plot and ASCII reports. Currently the view 2D data mode provides TV, SURFACE, CONTOUR, SHOW3, and XSURFACE plot for 2D image data. The 1D overlay mode supports simple 1D plot for multiple scans.

The **catcher** can be started in different operation mode such as no save or view only mode. In view only mode, a user can start the data catcher as a data browser without turning on the data acquisition. In no save mode, a user can start the data catcher as a real-time scan data display without saving the scan data. If two copies of data catcher are running at the same time and they both are trying to write to the same output file, if any one detects that the other one already wrote the new data on the file then it will be automatically turned to no save operation.

2. Program Descriptions

The data **catcher** is written in IDL programming language. It uses the CA functions defined in the ezcalDL library to communicate with the IOC. For the goal of reaching user simplicity, easy maintenance, and re-usability the program is structured into few major stand-alone subprogram packages.

Currently the data **catcher** is composed of the following subprogram packages:

- **catcher** - integrated Unix script command

- **catcher_v1** - integrated 1D/2D data catcher and display package
- **view1d** - 1D scan data display package
- **view1d_overlay** - 1D multiple scans display package
- **view2d** - 2D image data display package
- **ez_fit** - easy curve fitting package
- **plot1d** - a general purpose Cartesian plot package

The **catcher** is a Unix script program, rest are IDL programs. They are briefly described below. The details of command syntax and keyword options for each subprogram, and examples can be found in the following WEB page link.

http://www.aps.anl.gov/~bccha/Doc.html#IDL_HTML

catcher

The **catcher** is a script program written in Unix shell command language. It automatically set up the files to be used, sets up the environment variables and library paths, starts up the IDL, and loads the subprograms into the IDL. One big advantage of using this script file is that the IDL system environment variables are seen internally by the data **catcher** only and they will not affect any other IDL applications.

The script file **catcher** automatically invokes the IDL data catcher, **catcher_v1**, for user. It can be run from any directory as long as the directory name

`‘/usr/local/epics/extensions/bin/HOST_ARCH’`

is included in the user search path. At APS the HOST_ARCH can be either *sun4* or *solaris* which depends on the user's operating system.

The **catcher** provides few command line options to let the user override the default settings. The default file names assumed are: 'catch1d.config' for configuration file, 'catch1d.trashcan.xdr' for data file, and 'catch1d.env' for environment file.

At the very first time of running the data catcher in a directory a user can override the default setting on the command line through using the keyword options(for example refer the appendix section). If the configuration file is found by the data catcher, the configuration file overrides the command line specification. If the internal file selection dialog is used, it overrides the configuration file specification.

The command line syntax, and the optional keyword are shown in the appendix.

catcher_v1

Catcher_v1 is a simple mouse driven 1D/2D data acquisition program for EPICS 3.12.2 scan record. It allows the user to set up the data catcher for listening to a desired scan record. For restart run it automatically sets up from the configuration file. It detects the initiation of scan event, shows the real-time live data, saves the data at the end of scan, displays the final data. Post acquisition it also provides the simple and flexible features of displaying data, option of simple statistic calculation on peak, centroid and FWHM, easy curve-fitting features, and hard copy plot and various ASCII reports generation.

Catcher_v1 is a stand alone complete IDL program. It can be run directly from the IDL prompt by issue the IDL command **CATCHER_V1** as long as the environment is properly set and the path `'/usr/local/epics/extensions/bin/solaris'` is in the IDL search path. The default file names used are `'catch1d.trashcan.xdr'`, `'catch1d.config'`, and `'catch1d.env'`.

Catcher_v1 automatically saves the 2D images at the end of 2D scan. It provides the view 2D feature by directly invoking the **view2d** program.

Now the script program **catcher** automatically setup the environment and load this program into IDL for user, the simplest way of starting the data catcher is through this script file.

The command line syntax, and the optional keyword are shown in the appendix.

view1d

View1d is a complete mouse driven 1D scan browser. It reads the 1D scan data which is automatically generated by the 1D scan of the data **catcher**.

It provides flexible 1D scan data accessing and displaying, simple statistical calculation, and easy curve fitting. It provides automatically hard copy plot generation and various simple ASCII report generation functions.

View1d is a stand alone complete IDL program. It supports the 1D data either in the native binary format or XDR platform independent binary format. It can be run directly from the IDL prompt by issue the IDL command **VIEW1D** as long as the environment is properly set and the path `'/usr/local/epics/extensions/bin/solaris'` is in the IDL search path. The default scan file assumed is `'catch1d.trashcan'`.

The command line syntax, and the optional keyword can be found from the web page link:

http://www.aps.anl.gov/~bccha/idl_html/view1dRef.html.

view2d

View2d is a complete mouse driven 2D image browser. It provides packaged IDL TV, SURFACE, CONTOUR, SHOW3, and XSURFACE plot for 2D image data. It provides flexible 2D image accessing and displaying. It provides automatically hard copy generation and simple xz, and yz profile plot, image data query functions and simple ASCII image array.

View2d is a stand alone complete IDL program. It supports the image data either in native binary format or XDR platform independent format. It can be run directly from the IDL prompt by issue the IDL command **VIEW2D** as long as the environment is properly set and the path `'/usr/local/epics/extensions/bin/solaris'` is in the IDL search path. The default image file assumed is `'catch1d.trashcan.image'`.

It is assumed that the image data is automatically generated by the 2D scan of the data catcher. 2D image generated by other package can be converted into **view2d** format and then displayed by this program.

The command line syntax, and the optional keyword can be found from the web page link:

http://www.aps.anl.gov/~bccha/idl_html/view2dRef.html

view1d_overlay

View1d_overlay is a complete mouse driven multiple 1D scans displayer. It reads data for user entered 1D scans for a selected detector and generates overlay plot on the same graph.

View1d_overlay is a stand alone complete IDL program. It supports the 1D data either in the native binary format or XDR platform independent binary format. It can be run directly from the IDL prompt by issue the IDL command VIEW1D_OVERLAY as long as the environment is properly set and the path '/usr/local/epics/extensions/bin/solaris' is in the IDL search path. The default scan file assumed is 'catch1d.trashcan'.

The command line syntax, and the optional keyword can be found from the web page link:

http://www.aps.anl.gov/~bccha/idl_html/v1dOvIRef.html

ez_fit

Ez_fit is a complete mouse driven curve fitting package. It reads in 1D or 2D data array from an input file which can be either in XDR binary format or ASCII table format. The default binary file name assumed is 'fitting.bin1d' for 1D array, 'fitting.bin' for 2D array. In 'fitting.bin1d' file it contains x and y two XDR binary arrays. In 'fitting.bin' it contains the x, y, and im three XDR binary arrays. **Ez_fit** can also accept the data array directly from the command line.

Ez_fit is a stand alone complete IDL program. It can be run directly from the IDL prompt by issue the IDL command EZ_FIT as long as the environment is properly set and the path '/usr/local/epics/extensions/bin/solaris' is in the IDL search path. In data catcher the scan data is automatically plug into the **ez_fit** by 1D detector selection or 2D image array.

The command line syntax, and the optional keyword can be found from the web page link:

http://www.aps.anl.gov/~bccha/idl_html/ez_fitRef.html

plot1d

This program is a re-sizable IDL widget application. It is a general purpose flexible Cartesian line plot package. It provides simple to use automatic feature of line style, thickness, symbols, and color options on plot. User can specify the title, x label, y label, legend, footer, header, and comment on the command line. It also provides a print plot button to let the user easily to get a PS hard copy. The **view2d** program calls this program to generate the simple xz and yz profile cartesian plots. The **ez_fit** program calls this program to generate the simple curve fitting plot.

The command line syntax, and the optional keyword can be found from the web page link:

http://www.aps.anl.gov/~bccha/idl_html/plot1dRef.html

3. System and Access Setup

The set up for data catcher include two parts. The epics system administrator must install the complete package first. Then the user should set up his environment to access this package.

System Setup

The system administrator should install the following four directories from the EPICS extensions release 3.12.2 or newer release.

- epics/extensions/src/ezca
- epics/extensions/src/Ezca
- epics/extensions/src/ezcalDL
- epics/extensions/src/idl

The following files will be automatically installed in the epics/extension/bin/solaris directory:

- ezcalDL.so
- ezcalDL.pro
- ezcalDLWidgets.pro
- setup_ezcalDL
- ezcaidl_startup.pro
- catch1d.tbl
- PS_open.pro
- u_read.pro
- catcher
- catcher.exe2.idl5
- catcher_v1.init
- catcher_v1.pro
- dcviewer.pro
- viewer
- viewer_startup.pro
- view1d.init
- view1d.pro
- view1d_overlay.pro
- view2d.init
- view2d.pro
- ezfit
- ez_fit.pro
- plo1d.pro

The following files will be automatically installed in the epics/extension/doc directory:

- catcher.README
- catcher.html
- catcher_help.txt
- view2d_help.txt
- ezcalDLRef.html

- ezcalDLGuide.html

User Setup

The user only need to set the following:

1. setenv EPICS_EXTENSIONS /usr/local/epics/extensions
2. include the directory /usr/local/epics/extensions/bin/solaris in his/her search path.

Run Access

To access the data catcher just change to the directory where you want to store the data and type the command

catcher

The default file names 'catch1d.config', 'catch1d.env', and 'catch1d.trashcan.xdr' will be assumed. To obtain help on the command syntax of the **catcher** just enter

catcher -h

Examples

`catcher`

This will start the data catcher with complete default features. It includes real-time plot, data acquisition, saving acquired data, real-time text window, viewing old data option, listening to new data, etc.

`catcher -n`

This will start the data catcher with all the default options except that the acquired data will not be saved. It is assumed that there is another version of data catcher is running and the new scan data is automatically saved by that run.

`catcher -v`

This will start the data catcher as a data browser/viewer, no data acquisition will be done. It will only access the viewing features of the data catcher, no data acquisition is performed by this option.

User should know that it is better to use **viewer** than '**catcher -v**' because it is more efficient.

Program Limits

Summary of program system limits are given below:

<i>catch1d.env</i>	1000	Maximum number of IOC environment PV names allowed
<i>Scan Record</i>	15	Maximum detectors allowed in 1D scan record
	4	Maximum positioners allowed in scan record
<i>1D data file</i>	10000	Maximum number of 1D scans allowed in 1D data file
	4000	Maximum data points can be requested in 1D scan

4. Used Files

<i>2D image file</i>	1000	Maximum positioner data points can be requested in 2D scan
	1000x1000	Maximum image array allowed
	10000	Maximum number of 2D images allowed in 2D image file
	15	Maximum number of detector images associated with a 2D scan

4. Used Files

Two types of files are used by the data catcher: the user prepared files and the automatically generated files. Currently only the environment file is prepared by the user. The environment file is not required for the normal operation of the data catcher unless extra IOC variables to be saved with scan.

The configuration file normally is automatically generated/updated by the data catcher at the termination time if the scan record is properly connected. The modification of configuration file by the user is not recommended unless if he/she really knows what he/she is doing.

All the report files created by the data catcher are always stored in the directory where the data catcher was invoked. Normally the data file and environment file are also stored in the same directory but the user may change the default data directory by the file selection dialog.

The important files used by the data catcher include environment file, configuration file, data file, index file, image data file, and report files. They are briefly described in the following.

Configuration Files

Two configuration files can be prepared by users: environment file and configuration file.

env.file

The existence of the environment file is optional for the data catcher. If it is not given on the command line, 'catch1d.env' is assumed.

The 'catch1d.env' provides the flexible feature of saving additional IOC setting values with the scan data. It should contain a complete list of environment PV variables to be saved with the 1D scan. For each PV name defined in the 'catch1d.env' file the field value and field description is saved with the scan data.

A complete list of environment values plus a subset of important fields of the scan record will be saved for each reference case. For subsequent scans only those values changed from the reference case will be saved, or the PV name is prefixed by '*' will always be saved.

The reference case is defined as the data catcher is ready for acquiring new scan data, i.e. whenever the set up for a new scan PV names is called. PV name setup is automatically called at the initial start-up of the data catcher, at the close of the viewing mode, and at the user changing the scan record positioner and detector names.

4. Used Files

File Format Description

Each line in the 'catch1d.env' consists of a single valid PV name plus the field name followed with the field description. If the field name is not given then the default '.VAL' is assumed. The PV name and the description is separated at least by one blank space. The PV name plus the field name can not exceed 28 characters, and the field description can not exceed 40 characters. No waveform record is allowed in 'catch1d.env' because only single value can be returned. Currently, the number of lines allowed in 'catch1d.env' is 1000.

If the description is not specified for a given PV name in 'catch1d.env', then the description from the record's '.DESC' field will be used in the output data file. Any line begins with # in the first column will be treated as a comment line. Any PV name is prefixed by an '*' in the env.file, it will always be saved with each scan.

The existence of the environment file is optional for the data catcher. If it is not given on the command line, 'catch1d.env' is assumed.

Example

```
#
# catch1d.env
#
# Any line starting with # sign at the first column will be
# considered as a comment in this file.
#
# This file should contain the environment PV names which
# will be saved with scan data.
#
# At the end of the first scan the complete list of PV names
# and the corresponding reference values are saved with the
# scan data. For the subsequent scans only the value changed
# from the reference case are saved again.
#
# Any PV name prefixed with an "*" the value will be saved with every
# scan.
#
chademoai1          "Any description about the PV"
chademoai2          "Any applicable description"
*chaenergylevel    "Always saved "
*chacurrent         "Always saved"
.
.
```

The following fields of the scan record which are automatically included in the reference case, a user does not need to include them in his/her environment list if they do not change from scan to scan. They are given below, the first string is the field name, the second string is the description used in the data catcher:

<UserID:scan1>.NPTS	NPTS Number of Points	:
<UserID:scan1>.P1PV	P1PV Positioner # 1 PV Name	:
<UserID:scan1>.R1PV	R1PV Readback # 1 PV Name	:
<UserID:scan1>.P1SM	P1SM Positioner # 1 Step Mode	:
<UserID:scan1>.P2PV	P2PV Positioner # 2 PV Name	:
<UserID:scan1>.R2PV	R2PV Readback # 2 PV Name	:
<UserID:scan1>.P2SM	P2SM Positioner # 2 Step Mode	:
<UserID:scan1>.P3PV	P3PV Positioner # 3 PV Name	:
<UserID:scan1>.R3PV	R3PV Readback # 3 PV Name	:
<UserID:scan1>.P3SM	P3SM Positioner # 3 Step Mode	:
<UserID:scan1>.P4PV	P4PV Positioner # 4 PV Name	:
<UserID:scan1>.R4PV	R4PV Readback # 4 PV Name	:

4. Used Files

```
<UserID:scan1>.P4SM      P4SM Positioner # 4 Step Mode   :
<UserID:scan1>.D1PV      D1PV Data Detector # 1 PV Name  :
<UserID:scan1>.D2PV      D2PV Data Detector # 2 PV Name  :
<UserID:scan1>.D3PV      D3PV Data Detector # 3 PV Name  :
<UserID:scan1>.D4PV      D4PV Data Detector # 4 PV Name  :
<UserID:scan1>.D5PV      D5PV Data Detector # 5 PV Name  :
<UserID:scan1>.D6PV      D6PV Data Detector # 6 PV Name  :
<UserID:scan1>.D7PV      D7PV Data Detector # 7 PV Name  :
<UserID:scan1>.D8PV      D8PV Data Detector # 8 PV Name  :
<UserID:scan1>.D9PV      D9PV Data Detector # 9 PV Name  :
<UserID:scan1>.DAPV      DAPV Data Detector # A PV Name  :
<UserID:scan1>.DBPV      DBPV Data Detector # B PV Name  :
<UserID:scan1>.DCPV      DCPV Data Detector # C PV Name  :
<UserID:scan1>.DDPV      DDPV Data Detector # D PV Name  :
<UserID:scan1>.DEPV      DEPV Data Detector # E PV Name  :
<UserID:scan1>.DFPV      DFPV Data Detector # F PV Name  :
<UserID:scan1>.T1PV      T1PV Trigger # 1 PV Name       :
<UserID:scan1>.T1CD      T1CD Trigger # 1 Command      :
<UserID:scan1>.T2PV      T2PV Trigger # 2 PV Name       :
<UserID:scan1>.T2CD      T2CD Trigger # 2 Command      :
```

config.file

At the normal termination of the data catcher a configuration file will be automatically saved. If the configuration file is not specified on the command line, the default name 'catch1d.config' is assumed. Only if the scan record is properly connected at the termination time then the configuration file will be updated. The configuration file is always saved at the home directory where data catcher was initially invoked.

Be aware of that configuration file will be updated only at the normal exit of data catcher and meanwhile the scan record is properly connected at the exiting time.

File Format Description

The configuration file is automatically generated by the data catcher. The hand modification of this file is not recommended.

The contents of 'catch1d.config' include scan record name, the data directory and the data file name where the last scan data was saved, and important scan setup fields of the scan record. Be aware that the 'catch1d.config' only remembers the very last status of the data catcher when you normally exit the program.

The configuration variable can be parsed by the data catcher include the following:

```
scanData.pv      - gives the monitoring 1D scan record name
scanData.y_pv    - gives the monitoring 2D scan record name
scanData.y_handshake - gives the 2D scan handshake pv name
scanData.path    - specifies the last data directory accessed by the data
catcher
scanData.trashcan - specifies the data file name was last used by the data
catcher
scanData.envfile - specifies the environment file used by the data catcher
scanData.config - specifies the configuration file used by the data catcher
```

If the scanData.debug='1' is set in the configuration file, the diagnostic status output will be printed by the data catcher in the startup terminal window.

Example

A typical example of a configuration file generated by **catcher** is given below.

```
; Generated by CATCHER_V1(R2.2.2)
scanData.pv='cha:scanRec3SC'
scanData.y_pv='cha:scanRec4SC'
scanData.path='/tmp_mnt/home/oxygen6/CHA1/test/idl/catcher'
scanData.trashcan='catch1d.trashcan'
scanData.envfile='catch1d.env'
scanData.config='catch1d.config'
```

If configuration file is prepared by user, the user has to ensure that the scanData.config specified filename must be the same as the external file name.

Data Files

Three types of data files can be saved by the data catcher: 1D scan data file, 1D index file, and 2D scan image file. Data catcher R2.2.x support native binary format as well as XDR binary format. The XDR binary format file is platform independent.

If the 1D scan data is created by the File selection dialog then it will be saved in native binary format. If the 1D scan file is directly created by modifying the configuration file or by command line keyword specification then it will be saved in XDR binary format. Earlier versions generated data files are all in native binary format.

The 2D scan image file is always saved in native binary format.

Porting data to platform independent format please refer the following URL for data conversion.

<http://www.aps.anl.gov/~bccha/viewer.html>

How the data is recorded for 1D and 2D scan are given in this section. This will provide the user with necessary information in the case of post acquisition data processing in extracting data from the data file for non IDL applications.

data.file

This file is used to record the 1D scan data. Unformatted data is automatically saved at the end of each scan. Complete set of 1D scan data is sequentially appended at the end of the file. At the very first time, if the data file name is not given on the command line, 'catch1d.trashcan.xdr' is assumed.

File Format Description

The data recorded for the 1D data is in unformatted string or binary arrays. The size of environment array, data arrays varies from scan to scan. It uses the *u_read* and *u_write* procedures defined in the data catcher for I/O. Data are stored with the corresponding type and size information.

For each scan the following pieces of information are recorded. They include the version number, 1D scan PV name, positioner, detector arrays, plot legends, environment arrays, etc.

<i>version(1)</i>	string array, 'CATCHER_V1'
<i>pv (1)</i>	string array, 1D scan record PV name
<i>num_pts(1)</i>	short array, acquired points - 1
<i>realtime_id.def(19)</i>	short array, for 4 positioners and 15 detectors; 1 defined, 0 not defined.

4. Used Files

<i>dim(15)</i>	short array for 15 detectors, data points per position; default to 1.
<i>pa(num_pts+1)</i>	double precision array, number of positioner arrays found depends on the number of defined positioners for 1D scan record
<i>da(num_pts+1)</i>	float array, number of detector arrays found depends on the number of defined detectors for 1D scan record
<i>labels(30,57)</i>	byte array for storing PV names, descscs, engus (each byte string contains up to 30 characters) labels(0:29,0:18) - stored PV names for 4 positioners and 15 detectors labels(0:29,19:37) - stored PV descriptions for 4 positioners and 15 detectors labels(0:29,38:56) - stored PV engineering units for 4 positioners and 15 detectors
<i>strings(6)</i>	six plot string arrays for title, label, data filename, comment (each string contains up to 60 characters) strings(0) - Plot title strings(1) - Plot x label strings(2) - Plot y label strings(3) - Scan data file name strings(4) - Time stamp and user ID strings(5) - Comment string
<i>seqno(8)</i>	float array for scan ID numbers seqno(0) - 1D scan number, zero based sequence number seqno(1) - 1D scan starting reference case number seqno(2) - 2D scanning sequence number, Y loop index number seqno(3) - 2D scan number, zero based sequence number seqno(4) - 2D scanning indicator (1 is on, 0 is off) seqno(5) - 2D scan requested number of data points seqno(6) - 2D scan current acquired number seqno(7) - Y positioner value at given 2D scan number
<i>env_no(1)</i>	short array, total number of saved environmental variables for the scan. For the reference case all the pvnames from the environmental file are saved, for other case only pvnames prefixed with "*" or value changed are saved. Therefore, this number may vary from scan to scan.
<i>env(env_no)</i>	string array for saved environmental variables. Each string contains up to 110 characters, and is composed of 3 sub-strings: pvname, value string, and description env(0:29,i) - PV name, up to 30 characters env(30:69,i) - Value string, up to 40 characters

env(70:109, i) - Description, up to 40 characters

data.file.index

Adopting the feature of using a background index file for the saved scan data, this speeds up the access of the scan data greatly. The index file name is composed of the data file name suffixed with '.index'. If the default data file name 'catch1d.trashcan.xdr' is used, then the default 'catch1d.trashcan.xdr.index' is assumed.

The scan data index file is also automatically generated by the data catcher. This index file reduces the wait time in reading and greatly improves the scan data access time of the data catcher. The index file is written in unformatted binary array by the data catcher, it contains filename, filesize, no of scans, and byte offsets array for the scans.

<i>filename(1)</i>	string array, the associated 1D scan filename for this index file.
<i>fsize(1)</i>	long array, 1D scan file size in byte
<i>maxno(1)</i>	short array, total number of 1D scans in the 1D scan file
<i>array(maxno+1)</i>	long array, scan record offset file pointer locations

data.file.image

For 2D scan, in addition to 1D scan data the 2D image data is also saved. The name convention for the image file is the 1D data file name suffixed with the '.image'. If the default data file name 'catch1d.trashcan' is used, then the default 'catch1d.trashcan.image' is assumed.

For 2D scan, each detector defined for the inner scan record possesses its own 2D image. Complete set of 2D images data is sequentially appended at the end of 2D scan.

File Format Description The data recorded for the 2D image data is in unformatted binary arrays. For each detector the following pieces of information are recorded.

<i>name(60,6)</i>	Six byte array of size 60 characters. name(0:59,0) - Inner scan PV name name(0:59,1) - Outer scan PV name name(0:59,2) - 1D data file name name(0:59,3) - 1D scan positioner 1 PV name name(0:59,4) - 2D scan positioner 1 PV name name(0:59,5) - Detector 1 PV name
<i>qno(6)</i>	Short integer array of 6. qno(0) - 1D scanno at the end of 2D scan qno(1) - Image X width (1D scan requested NPTS) qno(2) - Image Y height (2D scan actually acquired points) qno(3) - Detector number of the 2D image qno(4) - 2D scanno qno(5) - Max Y height (2D scan requested NPTS)

5. User Interface

<i>X(width)</i>	Double array, 1D scan X positioner array
<i>Y(height)</i>	Double array, 2D scan Y positioner array
<i>Image(width,height)</i>	Float array, 2D image of the associated detector

Output Files

Report files generated by data catcher include ASCII report files and PostScript plot files. Normally report files are generated in the data directory. If the user has no write permission in the data directory, the report files will be saved in the startup directory.

Report files

Various levels of report files can be generated by the data catcher, they are all prefixed with the same 1D data file name. If the default file name 'catch1d.trashcan.xdr' is used, then the generated report file name will be 'catch1d.trashcan.xdr.xxxx_yyyy'. The xxxx_yyyy is automatically generated scan ranges, the xxxx is the 4 digit starting scan number and yyyy is the 4 digit ending scan number.

Three types of 1D scan report can be generated: full header, abbreviated header, or no header. At the no header case only the positioner and detector arrays are listed.

The 2D ASCII file for the given image can be generated by suffix the 'catch1d.trashcan.xdr.image' with zzzz where zzzz is the 4 digit image sequence number in the 2D image file. If the ASCII file already exists in the data directory the default file name view2d.data will be used.

PostScript files

Three postscript files catch1d.ps, view2d.ps, and idl.ps can be generated by the data catcher.

<i>catch1d.ps</i>	Postscript 1D plot file created by the catcher_v1 program.
<i>view2d.ps</i>	Postscript 2D plot file created by the view2d program.
<i>idl.ps</i>	Postscript 1D plot file created by the plot1d program.

5. User Interface

The data catcher is a complete mouse driven application. The user interface is written in IDL basic and compound widgets application. It lets the user set up the scan record to be monitored by the data catcher and automatically detecting and displaying the scan data while the realtime scan is going on. It also provides the user with the control of displaying data in various kind of plot option. At the end of scan the scan data is automatically saved for user.

The user interface will be discussed in the following groups: File, Setup, Plot Options, Help, ViewData, Drawing Area, Report Generation, Zoom, Statistic, Detectors, and Image menu. They are briefly described here. The purpose here is to provide the user with adequate knowledge in running this program.

In general the text field can be modified by directly typing in the field or simple text copy and paste. No text widget event generated until the carriage return is entered in the text field or some acceptance button is pressed. If the push button's label is suffixed with the '...' which implies that there is a pop up dialog window or widget application program associated with it.

The general messages generated by the data catcher are listed in the terminal window where the IDL was invoked. Wherever it is applicable, the popup message window is adopted to show self explanatory information, warning or error messages.

Startup Message Windows

Data catcher can start with / without configuration file.

If no configuration file is found the following message window pops up

Note: There is no configuration file found.
You first have to set up the scan PV name by using the
Setup->Scan ... menu

For normal operation, a user has to first set up the scan PV name to be listened to by the data catcher. At the normal exit of data catcher the new configuration file will be created by the data catcher.

If the configuration file is found in the startup directory then the following checks will be done. At the end of initialization, the data catcher will check whether there is any scan data held in the scan record. If either the scan record is empty or the scan record is busy with data taking then the data catcher is ready for data taking right away.

If the scan record holds scan data and it is idling, and the timestamp of the scan record is newer than the last scan data set from the saved file, then the following message window

Notice: At TimeStamp = MON DD, YYYY HH:MM:SS
Scan record "cha:scanRecSC" holds scanned data.
If you desired, you may enter Y to save this data
before you do any new scan.

Get Scan Data and Save (Y/N) ? Y

Accept Cancel

will pop up. If no save is to be done, just select the Cancel button right away. If the scan data is to be saved just select the Accept button, the new scan data will be saved and plot will be updated, then the data catcher will be ready for normal operation.

Main Window

File Menu

This menu let the user dynamically change the data file to be used by the data catcher.

The File menu provides the 'New', 'Open', 'Close', 'Save As', 'GetScanData + Save...', 'Copy', and 'Quit' options. If the user uses the script program 'catcher' with file specification on the command line, he may never need to use any other option except the 'Quit' to exit the data catcher program. The configuration file supersedes the command line data file specification. The File menu supersedes the configuration file data file specification.

This menu is not accessible while either the scan is going on or the view 1D dialog is opened.

New ... The 'New' option uses the IDL file selection dialog to let the user enter a new data file for storing the scan data from now on. The entered new file must not previously exist for the file system.

Open ... The 'Open' option uses the IDL file selection dialog to let the user switch to another data file for storing the scan data from now on. The file entered can be new or already existing. If nothing is selected, the default file name 'catch1d.trashcan' is assumed.

For a selected file, the message window pops up and shows the path name, data file name, number of scan found. If an existing scan data file is selected, the drawing area will be updated by the last scan from the selected file, and a question to ask the user to confirm whether it is okay to append the future scan data to the selected file. The default option is appending to the file. If 'N' is entered, a user will have to re-select the desired file.

Close The 'Close' option closes the currently selected data file and resumes the default file 'catch1d.trashcan' for storing any new scanned data.

Save As ... The 'Save as' option pops up the IDL file selection dialog to let the user enter a previously non-existing file name. It will save the currently acquired scan data to a brand new file.

GetScanData + Save ... If the auto save option is on, the 'GetScanData + Save' option will pop up the message dialog to let the user have a second change of deciding whether he/she really wants to get the scan data from the IOC and appended to the saved data file.

Copy ... The 'Copy' option lets the user copy the current selected data file to a new data file. In this newly created file, the internal data file name will be replaced by the external file name, the scans is re-sequenced.

Quit The 'Quit' option will destroy all the widgets, close the CA connection, and terminate the data catcher. If the configuration or index files do not exist yet they will be created otherwise they will be updated.

The 'Quit' is the only correct way of terminating the data catcher. This provides the safeguard for integrity of the data file.

Setup Menu

This menu sets up the program operation mode for the data catcher. It consists of the following items: 'Acquisition', 'AutoSave', 'Realtime', 'TextWin', 'Debug', 'Scan ...', and 'Color ...'. The availability of these items depends on how the data catcher is started.

At the normal acquisition mode, every item on this menu is settable by the user.

If the data catcher is started in a no save mode, then the item 'AutoSave' option is not available from this menu. The data catcher can display the old scan data from the data file or display the real-time live data but it will not save the new data to the data file. When the data catcher is operated in the no save mode, a big 'No Save' label will be shown on the upper right hand corner of the main window.

If the data catcher is started as a data browser, i.e. view only option, then only the item 'Color ...' option is available from this menu. When the data catcher is operated in the view only mode, a big 'View Only' label will be shown on the upper right hand corner of the main window.

Acquisition

The 'Acquisition' menu turns off/on the data acquisition mode. The default mode is 'On', it always acquires data while the scan is going on, displays the real-time plot, and saves the data. The 'Off' mode temporarily turns off data acquisition, the data catcher becoming a data browser and it will not listen to any new scan.

AutoSave

The 'AutoSave' menu turns off/on the data save mode. The default mode is 'On', it always saves the acquired scan data. The 'Off' mode causes the data catcher still acquiring the new scan data but it will not save the data. If the data catcher detects there is another copy of data catcher already using the exactly same data file for storing the scan data then the 'AutoSave' will be turned off automatically to avoid data corruption. When the second copy of the data catcher will be operated in no save mode and a big 'No Save' label will be shown on the upper right hand corner of the main window.

Realtime

The 'Realtime' menu turns off/on the real-time plot option. The default mode is 'On', it always displays the real-time plot. The 'Off' mode turns off the real-time option.

TextWin

The 'TextWin' menu turns off/on the real-time text window option. The default mode is 'Off'. If the real-time text window is desired a user has to explicitly turns on this option. The text window is able to list the real-time live data while the scan is going on. The 'Realtime' option must be on in order to have data in the 'TextWin'.

Debug

The 'Debug' menu turns off/on the diagnostic status message. The default mode is 'Off', it turns off the diagnostic message.

Scan ...

The 'Scan ...' option pops up the scan setup window. It consists 1D scan pvname control, 2D scan pvname control, and 2D handshake control. If no configuration file exists yet, this setup window must be used to properly set up the scan pvname for data catcher. 1D pvname is required for 1D scan, both 1D and 2D pvnames are required for 2D scan. 2D handshake name may be not required if it is already taking cared by the database.

Color ...

The 'Color ...' option runs the XLOADCT program from the IDL library. It allows the user dynamically change the IDL color table. Currently, the total number of colors used by the data catcher is 32. The default color table number used is 39 which is rainbow colors plus white.

Plot Options Menu

The Plot Options pull down menu allows the user override the default plot settings in the data catcher.

The plot will be automatically scaled to the maximum data range plus 5% margin in both ends. The plot title and labels will be generated directly from the EPICS database. The plot will be stamped with the file name, scan number, time as the scan initiated, and the user name.

This menu consists of eight selection items: 'Colors', 'Lines', 'Symbols', 'Grid', 'Err Bars', 'Y scale', 'Ranges ...', and 'Labels ...' They provide flexible controls on the line plot for the 1D scan data.

- Colors* Colors / Black&White
Selects the colors or BW plot option. The Colors option reloads the rainbow plus white color table into data catcher. The defaults option is colors.
- Lines* Solid/Dotted/etc or Solid Only
Plots data as different line style or as solid line only. It defaults to different line style.
- Symbols* Line Only / Symbol Only / Both.
Data plotted as line only, symbol only, or both.
- Grid* Off/On.
Turns grid lines off or on. It defaults to grid lines off.
- Err Bars* Off/On.
Turns error bar (10%) option off or on. It defaults to off.
- Y Scale* Linear / Log scale.
Y axis uses linear or log scale. It defaults to linear scale.
- Ranges ...* Pops up a plot ranges specification dialog which allows the user to modify the X-, Y- plot ranges.
- Labels ...* Pops up a plot labels specification dialog which allows the user to modify the plot X-, Y- labels, title, and comments for the 1D plot.

Help Menu

The Help pull down menu provides simple on-line helpful information to users.

The Help pull down menu consists of five selection items: 'Version ...', 'Release Note ...', 'Help ...', 'Catcher Html ...', and 'ezcalDL Html ...'. The last two items open **netscape** window and are only available at the view only mode.

Note if the **netscape** window is opened by this menu it temporarily grabs the control from the IDL and it interrupts the data catcher's normal execution. For normal operation of the data catcher, the user should close the **netscape** window right after viewing the document.

- Version ...* Pops up the message window to show the version information.
- Release Note ...* Pops up the release note window.
- Help ...* Pops up simple on-line help text window.

- Catcher Html ...* Pops up a **netscape** window to show the packaged data catcher reference manual.
- ezcalDL Html ...* Pops up a **netscape** window to show the ezcalDL reference manual.

ViewData Menu

As the data catcher is idle this menu allows the user to access 1D or 2D data freely. The ViewData pull down menu consists of two selection items: '1D ...', and '2D ...'.

- 1D ...* Pops up the 1D data displaying dialog. It allows the user to browse through previously captured 1D scan data.
- 2D ...* Runs the IDL VIEW2D program. It allows the user to browse through previously captured 2D image data.
- 1D Overlay ...* Runs the IDL VIEW1D_OVERLAY program. It allows the user to superpose few 1D scans on one single plot for a selected detector.

Status Text Field

This text field gives the current status of the SCAN Record. It is used to reflect the acquired data points for both 1D and 2D scan. It is used for the status for PV setup. It is also used for the status of scanning while the scan is going on, e. g.

- At 1D PV Setup:* >> PV Valid <<
- At 2D PV Setup:* >> PV2 Valid <<
- During 1D scanning:* SCANNING: 12 of 100 Pts
- During 2D scanning:* SCANNING: 12 of 100 Pts At 10'th Scan
- At End of Scan:* IDLE : Acquired 100 Pts

Print Menu

This menu let the user generate printer copies of the plot and ASCII reports.

The Print pull down menu consists of two selection items: 'Plot', and 'Report ...'.

- Plot* Creates the 'catch1d.ps' file and generates a hard copy of 1D plot at the user's default printer.
- Report ...* Pops up a report generation dialog. Three types of reports can be generated: with complete header, abbreviated header, and data only. It provides flexible creating, viewing, and printing features. Before viewing or printing the report, it must be generated first.

Zoom Menu

This menu provides various features of zooming the plot area.

The Zoom pull down menu consists of five selection items: 'Zoom To Box', 'Zoom In/Out', 'Calc Slopes', 'Auto Scale (Refresh)', and 'User Scale ...'. Be aware that the Zoom in the drawing area must always be terminated by the right mouse button.

5. User Interface

<i>Zoom To Box</i>	Selects bounding box zoom option. Steps: Move mouse into the drawing area Mouse buttons function: Left: drag the box Middle: resize the box Right: zoom to the box
<i>Zoom In / Out</i>	Selects zoom in / out mode (<u>must be terminated by the right mouse button</u>) Steps: Move mouse into the drawing area Mouse buttons function: Left: zoom in to 1/2 scale Middle: zoom out to 2 times Right: terminate zoom in/out mode
<i>Calc Slopes</i>	Slope calculation mode (<u>must be terminated by the right mouse button</u>). It allows the user repetitively to select the two end points and displays the calculated slope in a pop-up window. Steps: Move mouse into the drawing area Mouse buttons function: Left: pick the start point Middle: pick the end point Right: quit slope calc mode
<i>Auto Scale</i>	Replots with automatic scaling plus 5% margin.
<i>User Scale ...</i>	It pops up the set user scale dialog which allows replot either auto-scale or with user specified Xmin, Xmax, Ymin and Ymax ranges.

Statistic Menu

The Statistic pull down menu consists of three selection items: 'None', 'Peak/Centroid/FWHM on plot', 'Peak/Centroid/FWHM ...', and 'Average/Deviation ...'.

<i>None</i>	No statistic values shown on the plot
<i>Peak/Centroid/FWHM on plot</i>	This button turns on the statistic calculation and displays the Peak, Centroid, and FWHM on the plot for the selected detector.
<i>Peak/Centroid/FWHM...</i>	This button also pops up the statistic scroll window to display all the local Peaks and X Locations, Centroid, and FWHM for each selected detector. The statistic calculation can be turned off by close the statistic scroll window.
<i>Average/Deviation ...</i>	This button turns on the standard deviation calculation and displays the mean, standard deviation, and average deviation on the plot. It also pops up the statistic scroll window to display all statistic values (mean, standard deviation, mean absolute deviation, variance, skewness, kurtosis) for the selected detectors. The statistic calculation can be turned off by close the statistic scroll window.

Fitting Menu

The Fitting pull down menu consists of two selection items: 'Ez_fit...', and '1D Binary'.

Ez_fit ...

This button plugs the positioner array x and detector arrays y directly into the **ez_fit** program without saving the 'fitting.bin1d' file. Through selection the detector data from the data image area in ez_fit, a user can perform various kind of curve fitting.

1D Binary

This button saves the current scan's positioner array x and detector arrays y in the form can be directly loaded into **ez_fit** in the future. The file name used is 'fitting.bin1d'. The data is saved in XDR unformatted binary format.

Drawing Area

The drawing area is used for real-time and post data acquisition 1D plot. In real-time only live data, title, x label for the selected detectors are plotted. Post data scan the detailed plot title, labels, legends, timestamp, user ID, etc. are also drawn in the drawing area.

The top left hand corner contains the data file name. The top right hand corner contains the scan #. The bottom left hand corner contains the time stamp when this scan started and a comment line. The bottom right hand corner contains the user name.

The plot title, xlabel, ylabel and comment can be changed by the user through using the 'Plot Options->Labels ...' option. The default scan record database name, description and engineering units will be used for the plot. A user always can use the Plot Labels ... dialog to modify the plot specification at the view data mode.

The plot ranges can be automatically scaled or user specified. The default is auto-scaled, with a 5% margin. A user can use the Plot Ranges ... dialog to modify the user scale plot ranges Xmin, Xmax, Ymin, and Ymax in the user-scaled plot.

Note, if the user scale plot is selected and the data is outside the user plot ranges, then no data will be show on the plot. The data will always be properly plotted by the auto-scale plot option.

Y Selection

The Y selection buttons allows the user dynamically to change the detectors or the positioners to be plotted against the X axis. It consists of 19 radio buttons: 15 detectors and 4 positioners. It is also available in the real-time mode starting from R1.5.3.

The selection button indicates whether the detector will be plotted or not. If it is pressed in, the corresponding detector data will be plotted. The default set up with detector 1 and 2 are selected.

Xaxis Drop List

The Xaxis selection drop list allows the user dynamically to change the plot X axis. It consists of 5 types of x axes: 4 positioner arrays and data index array.

The positioner 1 array is used for default X axis selection. If an undefined positioner array is selected by the user then the index array will be used for the X array.

Image Drop List

Only one 2D image real-time window is available. This image drop list allows the user to switch to different detector image option during 2D scanning. At the end of each 1D scan the 2D image window is updated once. Switch the detector in this drop list will cause the corresponding image been displayed.

Image drop list consists of 'Small', 'Large', and 15 detector entries. This 2D real-time image indicator will show the progressing of 2D scan. The small 2D image strip will plot each detector image in 60x60 pixels area. The large 2D image strip will plot each detector image in 120x120 pixels area. The single detector picked then the image window with 200x200 pixels shows only the image associated with the selected detector.

At 2D scanning, the small or large image pan choice contains 15 detector image area. It clearly reflects all detectors currently has been defined in scan record. Image drop list defaults to small strip.

Sub-Windows

XY-COORD

This dialog shows the corresponding drawing area X and Y values at the cursor cross-hair position.

Click the Left Mouse Button (LMB) on the drawing area will draw a cross-hair and pops up a XY-COORD dialog. It contains a label of the reference positioner number (based on the X axis), two text fields, a 'GoTo ...' button, and a 'Close' button. The X and Y fields show the intersection coordinates of the cross hair. Press the 'GoTo ...' button pops up the set positioner locations dialog. Press the 'Close' button terminates the query of cursor location.

The X field can be modified by the user through typing in the field. The 'GoTo ...' dialog will reflect the final value entered in the X field.

GoTo ... Dialog

This dialog provides the feature of setting the location(s) for all the defined positioner(s) in the scan record.

The set positioner dialog displays a list of all the positioner location(s) corresponding to the drawing area cross-hair cursor (or user modified X value), and a question whether to set the new motor position(s) accordingly, an 'Accept' , and a 'Cancel' button. If the user press the carriage return for 'Y' or the 'Accept' button, then a channel access array put for all the specified positioner(s) will be issued and the hardware physically drives the positioner(s) to the new location(s). The 'Cancel' button closes the dialog.

Scan ... Dialog

The scan setup dialog is used to setup which scan record will be monitored by the data catcher.

It contains a row of widgets for 1D scan control, a row of widgets for 2D scan control, a row of widgets for alternate 2D scan handshake, an 'Ok' button, and a 'Close' button.

The 'Ok' button accepts all the text fields and properly sets up the data catcher for monitoring the future scan. The 'Close' button closes this dialog and let outside CA clients have sole control of scanning.

- 1D Scan Control* The text field specifies the PV name of a SCAN record. The input should end with a carriage return. The drag and drop of any scan record field name from the **medm** window is supported.
- The control of scanning consists of two buttons: 'Start' and 'Stop'. The 'Start' button initiates the 1D scan and displays the real-time plot of captured data at the drawing area. The 'Stop' button aborts the scan before the normal completion of 1D scan.
- 2D Scan Control* The text field specifies the PV name of the second SCAN record which controls the position of the outer loop. The input should end with a carriage return. The drag and drop of any scan record field name from the **medm** window is supported.
- The control of scanning consists of two buttons: 'Start' and 'Stop'. The 'Start' button initiates the 2D scan and displays the real-time 1D plot of captured data at the drawing area, and pops up a detector image window to show the 2D image data. The 'Stop' button aborts the 2D scan before the normal completion of 2D scan.
- 2D Handshake Control* The first text field specifies the PV name for 2D scan handshake, the second text field specifies the value to be used for handshake. The input should end with a carriage return. If the handshake PV name is specified a channel access put will be performed at the end of each (inner loop) 1D line scan.

Plot Labels ... Dialog

The plot labels dialog let the user dynamically modify the plot labels.

The plot labels dialog contains two information labels: the scan number, and the data file name. They reflect where the plot data come from. Plot setup dialog consists of four text fields which allow the user to modify the plot title, xlabel, ylabel, and comment.

The data catcher prepares the plot title, x and y labels directly from the EPICS database, they can be modified by the text fields defined in this dialog. In addition a plot comment text field up to 60 characters can be specified.

The text field entered will be in effect by carriage return or until either the 'Apply' or 'Done' is pressed. The 'Done' button will also close the dialog. If the 'Cancel' button is pressed, it terminates the modification and closes the dialog.

Plot Ranges ... Dialog / User Scale ... Dialog

The user scale dialog allows the user to dynamically specify the X and Y plot ranges.

It consists of Xmin, Xmax, Ymin, Ymax text fields and three push buttons. Slider bars are provided for the easy task of entering the plot ranges. A user can type directly into these fields.

The 'User Scale' button takes the X and Y ranges and replots the data. Data falls outside the plot ranges will not be shown in the user scale plot. However, in realtime mode only the Y range is used by the realtime plot.

The 'Auto Scale' button re-plots the whole data range plus 5% margins.

The 'Done' button closes the dialog.

View 1D ... Dialog

View 1D dialog allows the user freely to browse through the 1D scan data file.

View 1D yields to the real-time scanning. Scan can be initiated by any CA clients. The View 1D dialog will be automatically closed by the data catcher if the real-time scanning is detected.

As the view 1D dialog is opened, the File pull down menu is inaccessible. A user has to explicitly close the view 1D dialog, in order to access the File menu. For example, a user wants to quit the data catcher, he has to close the view 1D dialog first, then uses File->Quit menu.

View 1D dialog consists of: a label describes the current file opened for viewing, a label gives the ranges of scan data found in the file, a text field shows the selected record # from the opened file, a slider bar for selecting the desired scan record, and two sets of push buttons. They can be divided into scan selection control and output control.

Selection Control

The text field entry allows you to enter a scan # to be selected. It is also used as an indicator to show the current scan # selected.

The slider bar allows you freely to view any valid scan #.

Press 'First' to view the scan # 1.

Press 'Next' to view the next scan #.

Press 'Prev' to view the previous scan #.

Press 'Last' to view the last scan #.

Press 'Done' to close the view mode.

Output Control

A Column Format text field is added in this dialog, it controls the ASCII data array output. It defaults to G18.8, this implies that a 18 character columns is used with 8 significant digits. If higher accuracy is desired, a user can modify this text field to get more significant digits for array output. It accepts any scientific fortran format.

The 'Print Plot', 'Modify Plot', 'ASCII View', and 'ASCII Print' buttons will be active only if the selection control has been activated and the selected scan data has been loaded into the data catcher.

Press 'Print Plot' generates the postscript plot file 'catch1d.ps' and sends it to the user default printer.

Press 'Modify Plot' allows you to modify the plot labels according to your changes.

Press 'ASCII View' saves the ASCII scan data in '<filename>.tmp' file and calls the XDISPLAY program to display the file.

Press 'ASCII Print' sends the file '<filename>.tmp' to the user's default printer. It is accessible only if the 'ASCII View' has been pressed for the selected scan #.

Report ... Dialog

This dialog allows the user to generate three types of ASCII report for a selected range of scans.

This dialog consists of 6 rows of widgets: Header Options, Data File Name, Start Scan #, End Scan #, Output File Name, Action Buttons.

Header Options

Three types of report can be generated: full header / abbreviated / none.

If none option is selected, only the scan #, and the name, description, engineering units, data array for all defined positioners and detectors are reported.

If the abbreviated header option is selected, the version, scan record name, full path data file name, timestamp, user ID, comment are included in the ASCII data report.

If the full header option is selected, in addition of the abbreviated headers the environment variables, descriptions, values, the plot specifications are also included in the report.

<i>Data File Name</i>	This field specifies the data file name to be used.
<i>Start Scan #</i>	This field specifies the beginning scan number to be included in the report. It defaults to the last.
<i>End Scan #</i>	This field specifies the ending scan number to be included in the report. It defaults to the last.
<i>Output File Name</i>	This field specifies the output report name. It is automatically generated when either the start or the end scan # field is entered with carriage return. The user can modify the name if he desired.
<i>Action Buttons</i>	<p>It includes 4 buttons: 'Generate Report', 'View Report', 'Print Report', and 'Done'</p> <p>The 'Generate Report' button accepts the user specifications and generates the report and saves in the output file name. During the report generation, the 'View Report', and 'Print report' are inaccessible.</p> <p>The 'View Report' button runs the XDISPLAY program to view the ASCII report.</p> <p>The 'Print Report' button sends the ASCII report to the user's default printer.</p> <p>The 'Done' button closes the dialog.</p>

View 2D ... Dialog

This is a stand-alone 2D image browser program. It allows the user flexibly to browse the 2D scan image data saved by the data catcher. For a given image it provides TV, SURFACE, CONTOUR, SHOW3, and XSURFACE plot. For more information please refer to VIEW2D on line help.

Ez_fit ... Dialog

This is a stand-alone easy fitting program. It allows the user flexibly to pick the detector data saved by the data catcher and perform the simple curve fitting. For more information please refer to EZ_FIT on line help.

6. A Tutorial Example

For the normal operation of the data catcher, a user has to load the scan record database into the IOC, and make sure that the scan record itself is functional properly before invoking the data catcher.

The procedure for running a 1D scan is given below:

Load Database

- Load the scan record database into the IOC
- Setup and initialize the scan record

Following example database gives a simple example of scan record database. The **scan** record consists of two positioners and two detectors. Each positioner is a soft **ao** record. Each detector is a soft **calc** record. And a soft **pulseDelay** record is used to control the positioners. The 'UserID' should be the user login id. The purpose here is only to provide a simple example and no hardware is involved.

Generally a user can use other tools, e.g. **medm**, to initialize or reset the scan record.

```
database(scanRec)
{
  record(ao, "UserID:xposController1AO")
  {
    field(FLNK, "UserID:detector1CC.PROC")
    field(DOL, "1")
    field(PREC, "2")
    field(DRVH, "100")
    field(DRVL, "-100")
  }
  record(ao, "UserID:xposController2AO")
  {
    field(FLNK, "UserID:detector2CC.PROC")
    field(DOL, "1")
    field(PREC, "2")
    field(DRVH, "100")
    field(DRVL, "-100")
  }
  record(calc, "UserID:detector1CC")
  {
    field(FLNK, "UserID:dwelldelayPD.PROC")
    field(CALC, "90*(SIN(a))")
    field(INPA, "UserID:xposController1AO.VAL NPP MS")
    field(INPB, "1")
    field(PREC, "1")
  }
  record(calc, "UserID:detector2CC")
  {
    field(FLNK, "UserID:dwelldelayPD.PROC")
    field(CALC, "(a>10)&&(a<20)?b*TAN(a):0")
    field(INPA, "UserID:xposController2AO.VAL NPP MS")
    field(INPB, "1")
    field(PREC, "1")
  }
  record(pulseDelay, "UserID:dwelldelayPD")
  {
    field(UNIT, "Milliseconds")
    field(DLY, "10")
    field(WIDE, "0.2")
    field(TTYP, "Software")
    field(STV, "Enable")
    field(PREC, "3")
  }
  record(scan, "UserID:scanRecSC")
  {
    field(EVNT, "5")
    field(SCAN, ".1 second")
    field(PRIO, "HIGH")
    field(MPTS, "512")
    field(P1PV, "UserID:xposController1AO")
    field(P2PV, "UserID:xposController2AO")
    field(D1PV, "UserID:detector1CC")
  }
}
```

```
        field(D2PV, "UserID:detector2CC")
        field(NPTS, "101")
        field(P1SP, "0.")
        field(P1EP, "10.")
        field(P2SP, "10.")
        field(P2EP, "20.")
    }
}
```

Invoke catcher

- Invoke the data catcher with all the default setup:
catcher

Setup Scan PVname and Plot Options

- Setup the data catcher to communicate with the IOC
Select the Setup->Scan ... menu from the main window
Enter 1D scan record name 'UserID:scanRecSC' with carriage return or 'Ok' button to set up the data catcher to listen to the specified scan record
Press 'Close' button to close the setup dialog
- Select Plot Options menu to turn on the options as you like
- Select Statistics->Peak, Centroid, FWHP on plot if it is desired

Data Scanning

- Adjust the scan record fields if it is desired
- Start the 1D scan either internally or by outside CA client
Internally - Push the 'Start' button from the scan setup dialog if it is not closed.
or by
Outside client - e.g. push the scan button on the **medm** screen of the scan record

The user can query the xy value in the drawing area, adjust the database, repeat the data scanning as long as it is desired. All the scan data will be automatically saved at the end of each scan and replotted on the drawing area.

View 1D Data

- Select the pull down menu ViewData->1D... to pop up the view 1D dialog
- Select Plot Options menu to modify the plot as you like
- Select Zoom menu to refine the plot or generate user scaled plot
- Select Statistics menu to include peak, centroid, FWHP calculations
- Select Print menu to generate PS plot or reports
- Close view 1D dialog

Quit Catcher

- Quit the data catcher while no scanning is going on
Select the File->Quit menu to exit

The user can only quit the data catcher through the File menu. The File menu is inaccessible while the scan is going on or the view 1D dialog is opened. If the scan is going on, a user has to wait until the scan is finished or aborted then the File menu is accessible again. If the view 1D dialog is opened, a user has to explicitly close the dialog then the File menu is accessible again.

At the normal termination of data catcher, the configuration file and data index file will be automatically updated.

7. How To Restore Damaged File?

All kinds of data protection consideration are incooperated into the data catcher. Normally, the data catcher will never failed to save the scan data properly unless the operator abnormally terminated the data catcher by killing the process or workstation powered off during the middle of writing scan data. A user rarely needs to do the following unless the data catcher failed to read in the whole data set.

To restore the damaged file the following procedure should be taken:

1) Make sure to access the developer license version of data catcher (in run time version no IDL prompt is allowed.)

```
catcher -D
```

2) To get the IDL> prompt at the IDL session window by typing in the Cntl-C and then point the mouse to any of the data catcher window

```
Cntl-C
```

3) Determine the number of good 1D scans saved in the file (Starting from data catcher R2.2 the newly created data file will be saved in XDR format). The keyword /XDR is required if the data was saved in XDR format.

```
IDL> u_openr,unit,'catch1d.dat' [,/XDR]
IDL> salvage_read,unit,no
```

Note: A list of scans successfully read in by the salvage_read is printed in the IDL session

4) Salvage the file up to the very last scan successfully read in

```
IDL> salvage_trashcan_problem,'catch1d.dat',no
IDL> u_close,unit
```

Note: A new file 'catch1d.dat' will be created and the original damaged file will be renamed as 'catch1d.dat.bk'

5) Remove the corresponding old index file and restart the data catcher

```
IDL> $rm 'catch1d.dat.index'
IDL> retail
IDL> xmanager
```

8. Appendix

The keyword included within the [] is optional.

catcher

The **catcher** is a Unix script file which automatically sets up and starts the IDL data catcher for the EPICS 3.12.2 user. The default IDL run-time license will be used by the data catcher. The command line keyword option -D is used for running data catcher with IDL developer license.

Command Syntax

catcher [-c file.config] [-d file.data] [-e file.env] [-h] [-n] [-v] [-D]

Keyword Option

-c file.config

Assigns the configuration file to be used. The file name should begin with an alphabetic character and must separate with the option keyword -c(onfig) with a blank space.

If this option is not specified, the default file 'catch1d.config' will be used. At the normal completion a new version of configuration will be saved or updated in the directory where the data catcher was invoked.

-d file.data

Assigns the acquisition data file to be used. The file name should begin with alphabetic character and must separate with the option keyword -d(ata) with a blank space.

If this option is not specified, the default file 'catch1d.trashcan' will be used. If the configuration file exists, then the data file specified in the configuration file overrides this default.

-e file.env

Assigns the environment file to be used. The file name should begin with alphabetic character and must separate with the option keyword -e(nv) with a blank space. This file allows additional environment variables to be saved with the acquired scan data.

If this option not specified, the default file 'catch1d.env' will be used. The default directory of the environment file is the same as that the data catcher is invoked. If the same name environment file is found in the data directory then that file overrides the default one.

-h

Gives the command line help page.

-n

Starts the data catcher in no save mode, new scan data will be displayed in this mode but they will not be saved in the data file. Normally when there is another copy of data catcher already running with default setting and a user has to examine the saved or life data on another session he can use this option. The configuration file will not be updated if this -n(osave) option is specified on command.

-v

Starts the data catcher as view only mode, in this mode the data catcher can only access the view old data functions. It will not listen to the new scan data at all. Normally it is assumed that there is no data acquisition to be

done by this option. The configuration file will not be updated for this -v(iewonly) operation mode.

-D

This option overrides the using of default IDL run time license by the developer IDL license at APS.

Examples

Start with IDL developer license

catcher -D

Start with default settings

catcher

Start as view only

catcher -v

Override the default setting for config, env, and data file, the 'test' is used for this example

First time creation:

catcher -c test.config -d test.data -e test.env

Restart with the configuration file 'test.config':

catcher -c test.config

catcher_v1

This program provides the EPICS user a convenient IDL widget application data acquisition tool. It calls the IDL external CA functions which are defined in the ezcalDL package.

Currently, this program can capture both 1D and 2D scan data through setting monitor on the EPICS scan record. During data acquisition it also provides real-time plotting. Post acquisition it also provides primitive 1D and 2D post data plot and analysis. Each scan record can support up-to four positioners and fifteen detectors.”

IDL Command Syntax

```
catcher_v1 [,config='config.file'] [, data='data.file'] [, env='env.file'] [,  
nosave=n] [, viewonly=v] [, group=group_leader]
```

*Keyword Option***CONFIG='config.file'**

Specifies the configuration file to be saved when data catcher is normally terminated. If not specified, the default configuration file name 'catch1d.config' is assumed.

DATA='data.file'

Specifies the data file name to be used for appending new captured scan data. If not specified, the default data file name 'catch1d.trashcan' is assumed.

ENVFILE='env.file'

Specifies the environment (process variables) file to be saved with the captured scan data. If not specified, the default environment file name 'catch1d.env' is assumed.

NOSAVE=n

Specifies whether the data catcher should be started in a nosave mode. If n=1 no save is turned on.

VIEWONLY=v

Specifies whether the data catcher should be started as a data browser only. If v=1 then viewonly is assumed.

GROUP=group_leader

The widget ID of the group leader of the widget. If this keyword is specified, the death of the group leader results in the death of **catcher_v1**.

Output

It provides real-time plot and various post acquisition 1D and 2D surface plot. In addition of DATA file is saved, if 2D scan is detected the DATA.image file will also be saved at the end of 2D scan. Various levels of report files can also be generated by this program. All the file generated will be prefixed by the DATA file name.

Common Blocks

```
CATCH1D_COM  
W_PLOTSPEC_BLOCK  
W_VIEWSCAN_BLOCK
```

REALTIME_BLOCK
ENV_FIELD_BLOCK
FIELD_BLOCK

Side Effects

It is capable to detect scan invoked by outside channel access client.

Restrictions

Supports EPICS 3.12.2 and later scan record.

Procedure

For live data acquisition: boot the IOC first, setup and initialize the scan records, setup the run ezcalDL environment, run IDL, then start up the data catcher at the IDL prompt.

A simplified UNIX script file catcher can be used for starting up this program. For help on catcher enter "catcher -h" at the UNIX prompt.

Examples

Start with default setting for config, envfile, and data file

```
IDL> catcher_v1
```

Start as view only

```
IDL> catcher_v1, /viewonly
```

Override the default setting for config, env, and data file, the 'test' is used for this example.

First time creation:

```
IDL> catcher_v1, config='test.config', envfile='test.env',  
data='test.data'
```

Restart with the configuration file 'test.config'

```
IDL> catcher_v1, config='test.config'
```

view2d

This program provides the EPICS user a convenient IDL 2D scan data display tool. Its input image file is automatically saved by the data catcher program **catcher_v1**.

Currently, this program provides TV, SURFACE, CONTOUR, SHOW3, and XSURFACE plot. It also provides simple xz, yz line plot and data value query information.

IDL Command Syntax

view2d [, **group=group_leader**] [, **file='image.file'**]

Keyword Option

GROUP=group_leader

The widget ID of the group leader of the widget. If this keyword is specified, the death of the group leader results in the death of **view2d**.

FILE='image.file'

The input image file name. If this keyword is specified, the file should contain the image data which must be in the data catcher created format. If it is not specified, view2d assumed that the default file 'catch1d.trashcan.image' is used. A user can use the file open menu to specify the desired input image file.

Output

It provides option of postscript plot of drawing area.

Restrictions

Drawing area is 400 x 400 pixels.

Procedure

This program is available as an epics/extensions tool. It can be directly accessed from the view data menu of the **catcher_v1**.

Example

IDL> **view2d**

plot1d

This routine provides a general purpose flexible Cartesian plot package. It provides simple to use automatic feature of line style, thickness, symbols, and color options in plot.

The window generated by this routine will be re-sizable by the window manager.

Depress the 'Print' button will generate a postscript copy of the graph.

Normally it accepts two parameters X and Y. If the first parameter is not used then the data array is plotted on the ordinate versus the point number on the abscissa. Multiple curves (or variables) can be stacked into the second parameter as a two dimensional array, the first dimension gives the number of data points in each curve, the second dimension gives the number of curves to be plotted.

IDL Command Syntax

plot1d, [X ,] Y [, ID_TLB] [, ID_DRAW]

Input

X

The vector array for X abscissa.

Y

The Y data array for curve plot. The Y array can contain more than one curve, the first dimension gives the number of data to be plotted for the curve, the second dimension gives the number of curves to be plotted.

Output

ID_TLB

Optional output. The widget ID of the top level base returned by the PLOT1D.

ID_DRAW

Optional output. The widget ID of the drawing area used by the PLOT1D.

Keyword Option

TITLE='title string'

Set this keyword to specify the plot title string.

XTITLE = 'x label'

Set this keyword to specify the xtitle string.

YTITLE = 'y label'

Set this keyword to specify the ytitle string.

COLOR=n

Set this keyword to specify the color number used in the plot routine.

SYMBOL=1

Set this keyword to specify data plotted as symbol, set to -1 data plot as symbol and connected with line.

XLOG=1

Set this keyword to specify a logarithmic X axis.

YLOG=1

Set this keyword to specify a logarithmic Y axis.

XRANGE=[xmin, xmax]

Set this keyword to specify the desired data range for the X axis.

YRANGE=[ymin,ymax]

Set this keyword to specify the desired data range for the Y axis.

XMARGIN= [left, right]

Set this keyword to specify the left and right margin, default xmargin=[10,3]

YMARGIN=[top,bottom]

Set this keyword to specify the bottom and top margin default ymargin=[5,3]

THICK=n

Set this keyword to specify the line thickness for the axes and the line plot.

LINestyle=1

Set this keyword to turn on different line style used.

XSTYLE

Set this keyword to control x axis in IDL **plot** routine.

YSTYLE

Set this keyword to control Y axis in IDL **plot** routine.

LEGEND=legends(n_curve)

Set the legend strings corresponding to curves drawn.

XYLEGEND=[xloc, yloc]

Set the x, y location of the legend strings,% from the lower left corner from the graph window, default xylegend=[0.75, 0.35].

COMMENT=footnotes(n)

Set this keyword to write n lines of footnotes on the graph.

STAMP=1

Set this keyword to put the time stamp and user ID on the plot.

WTITLE='wtitle'

Set this keyword to specify the window title string, default to 'Plot1d'.

WIDTH=width

The initial window width at the creation time, which default to 250 pixel.

HEIGHT=height

The initial window height at the creation time, which default to 250 pixel.

GROUP=group_leader

The widget ID of the group leader of the widget. If this keyword is specified, the death of the group leader results in the death of **plot1d**.

BUTTON=1

Set this keyword if no print and close buttons to be created by the **plot1d**.

CLEANUP=1

Set this keyword the **plot1d** window can only be destroyed by the close button not by the window manager.

Side Effects

If more than one curves are stored in the Y array, it automatically uses different color for each curve. If the color keyword is set by the user, then the specified color will be used for the whole plot.

Restrictions

It is assumed that the X position array is the same for multiple curve plot.

Examples

Create a re-sizable line plot without any title or label specification.

```
IDL> x = !pi * indgen(100)/25
IDL> PLOT1D, x, sin(x)
```

Create a re-sizable line plot with title specifications.

```
IDL> PLOT1D, x, sin(x), title='title', xtitle='xtitle', ytitle='ytitle'
```

Plot two curves with different line style and legend at default location.

```
IDL> x=indgen(100)
IDL> y=make_array(100,2)
IDL> y(0,0)=sin(x * !pi / 50)
IDL> y(0,1)=cos(x * !pi / 50)
IDL> PLOT1D,x,y,legend=['line1','line2'],/linestyle
```

Same as the above example plus symbol and a specified legend location.

```
IDL> x=indgen(100)
IDL> y=make_array(100,2)
IDL> y(0,0)=sin(x * !pi / 50)
IDL> y(0,1)=cos(x * !pi / 50)
IDL> PLOT1D,x,y,/linestyle,symbol=-1, $
    legend=['line1','line2'], xylegend=[0.5,0.9]
```

Plot x, y array plus two lines of comment and a time stamp on the graph.

```
IDL> PLOT1D,x,y,comment=['Comment line1','Comment line2'],
/stamp
```

References

- [1] IDL Reference and User's Guide, Research Systems, Inc.
- [2] Modified version of EzcaIDL Reference manual
</usr/local/epics/extensions/doc/ezcaIDLRef.html>
- [3] EPICS Input Output Controller (IOC) Record Reference Manual R3.12
- [4] Motif-based Display Editor / Manager (MEDM)
<http://www.aps.anl.gov/asd/controls/epics/EpicsDocumentation/WWWPages/EpicsFrames.html>

