mdfreader Documentation

Release 4.0

Aymeric Rateau

CONTENTS

1	ndfreader module documentation 1 Platform and python version	3		
2	ndf module documentation 2.1 Dependencies			
3	ndf3reader module documentation .1 mdf3reader	19		
4	ndfinfo3 module documentation .1 Dependencies			
5	ndf4reader module documentation 1.1 Dependencies			
6	ndfinfo4 module documentation 5.1 mdfinfo4	35		
7	hannel module documentation	45		
8	ndices and tables	53		
Ру	on Module Index	55		
In	ndex			

Contents:

CONTENTS 1

2 CONTENTS

CHAPTER

ONE

MDFREADER MODULE DOCUMENTATION

Measured Data Format file reader main module

1.1 Platform and python version

With Unix and Windows for python 2.6+ and 3.2+

Author Aymeric Rateau

Created on Sun Oct 10 12:57:28 2010

1.2 Dependencies

- Python >3.4 http://www.python.org
- Numpy >1.14 http://numpy.scipy.org
- Sympy to convert channels with formula
- bitarray for not byte aligned data parsing
- Matplotlib >1.0 http://matplotlib.sourceforge.net
- · scipy for NetCDF
- h5py for the HDF5 export
- xlwt for the excel export (not existing for python3)
- openpyxl >2.0 for the excel 2007 export
- hdf5storage for the Matlab file conversion
- zlib to uncompress data block if needed

1.3 mdfreader

Bases: mdfreader.mdf4reader.Mdf4, mdfreader.mdf3reader.Mdf3

Mdf class

fileName

file name

Type str

MDFVersionNumber

mdf file version number

Type int

masterChannelList

Represents data structure: a key per master channel with corresponding value containing a list of channels One key or master channel represents then a data group having same sampling interval.

Type dict

multiProc

Flag to request channel conversion multi processed for performance improvement. One thread per data group.

Type bool

fileMetadata

file metadata with minimum keys: author, organisation, project, subject, comment, time, date

Type dict

read(file_name = None, multi_processed = False, channel_list=None, convert_after_read

filter_channel_names=False, no_data_loading=False, compression=False)

reads mdf file version 3.x and 4.x

write (file_name=None)

writes simple mdf file

get_channel_data(channel_name)

returns channel numpy array

convert_all_channel()

converts all channel data according to CCBlock information

get_channel_unit (channel_name)

returns channel unit

plot (channels)

Plot channels with Matplotlib

${\tt resample} \ (sampling_time = 0.1, master_channel = None)$

Resamples all data groups

export_to_csv (file_name = None, sampling = 0.1)

Exports mdf data into CSV file

export_to_NetCDF (file_name = None, sampling = None)

Exports mdf data into netcdf file

export_to_hdf5 (file_name = None, sampling = None)

Exports mdf class data structure into hdf5 file

export_to_matlab (file_name = None)

Exports mdf class data structure into Matlab file

export_to_excel (file_name = None)

Exports mdf data into excel 95 to 2003 file

```
export_to_xlsx (file_name=None)
    Exports mdf data into excel 2007 and 2010 file

convert_to_pandas (sampling=None)
    converts mdf data structure into pandas dataframe(s)

keep_channels (channel_list)
    keeps only list of channels and removes the other channels

merge_mdf ( mdf_class ):
    Merges data of 2 mdf classes
```

mdf class is a nested dict. Channel name is the primary dict key of mdf class. At a higher level, each channel includes the following keys:

- 'data' : containing vector of data (numpy)
- 'unit' : unit (string)
- 'master': master channel of channel (time, crank angle, etc.)
- · 'description': Description of channel
- 'conversion': mdfinfo nested dict for CCBlock. Exist if channel not converted, used to convert with getChannelData method

Examples

```
>>> import mdfreader
>>> vop=mdfreader.Mdf('NameOfFile')
>>> yop.keys() # list channels names
# list channels grouped by raster or master channel
>>> yop.masterChannelList
>>> yop.plot('channelName') or yop.plot({'channel1','channel2'})
>>> yop.resample(0.1) or yop.resample()
>>> yop.export_to_csv(sampling=0.01)
>>> yop.export_to_NetCDF()
>>> yop.export_to_hdf5()
>>> yop.export_to_matlab()
>>> yop.export_to_excel()
>>> yop.export_to_xlsx()
>>> yop.convert_to_pandas() # converts data groups into pandas dataframes
>>> yop.write() # writes mdf file
# drops all the channels except the one in argument
>>> yop.keep_channels(['channel1','channel2','channel3'])
>>> yop.get_channel_data('channelName') # returns channel numpy array
>>> yop=mdfreader.Mdf() # create an empty Mdf object
# add channel in Mdf object
>>> yop.add_channel(channel_name, data, master_channel, master_type, unit='lumen',

→ description='what you want')
>>> yop.write('filename') # change version with yop.MDFVersionNumber or,,
→specifically use write3/4()
```

concat_mdf (mdf_class)

Concatenate data of input mdf class after the current one.

Parameters mdf_class (Mdf) - mdf class instance to be concatenated after self

1.3. mdfreader 5

It creates union of both channel lists and fills with Nan for unknown sections in channels If one channel is not present in both classes, masked array is created If invalid bytes are present, masked array are created

convert_all_channels()

Converts all channels from raw data to converted data according to CCBlock information. Converted data will take more memory.

convert_to_pandas (sampling=None)

converts mdf data structure into pandas dataframe(s)

Parameters sampling (float, optional) - resampling interval

Notes

One pandas dataframe is converted per data group (one master per data group)

cut (master_channel, begin=None, end=None)

Cut data

Parameters

- master channel (str) channel to cut data (can be master channel)
- **begin** (*float*) beginning value in master channel from which to start cutting in all channels
- end (float) ending value in master channel from which to start cutting in all channels

Notes

Only the data groups with same master type as master_channel will be cut (only for mdf4)

```
export_to_NetCDF (file_name=None, sampling=None)
```

Exports mdf data into netcdf file

Parameters

- **file_name** (*str*, *optional*) file name. If no name defined, it will use original mdf name and path
- sampling (float, optional) sampling interval

Notes

Dependency: scipy

export to csv(file name=None, sampling=None)

Exports mdf data into CSV file

Parameters

- **file_name** (*str*, *optional*) file name. If no name defined, it will use original mdf name and path
- sampling (float, optional) sampling interval. None by default

Data saved in CSV file be automatically resampled as it is difficult to save in this format data not sharing same master channel -> not applicable for mdf4 in case there are master channels

with various types

Warning: this can be slow for big data, CSV is text format after all

```
export_to_excel (file_name=None)
```

Exports mdf data into excel 95 to 2003 file

Parameters file_name (str, optional) - file name. If no name defined, it will use original mdf name and path

Notes

xlwt is not fast even for small files, consider other binary formats like HDF5 or Matlab. If there are more than 256 channels, data will be saved over different worksheets. Also Excel 2003 is becoming rare these days, prefer using exportToXlsx. Dependencies: xlwt for python 2.6+, xlwt3 for python 3.2+

export_to_hdf5 (file_name=None, sampling=None, compression=None, compression_opts=None) Exports mdf class data structure into hdf5 file

Parameters

- **file_name** (*str*, *optional*) file name. If no name defined, it will use original mdf name and path
- sampling (float, optional) sampling interval.
- **compression** (*str*, *optional*) HDF5 compression algorithm. Valid options are 'gzip', 'lzf'. gzip compression recommended for portability. szip compression not supported due to legal reasons.
- **compression_opts** (*int*, *optional*) HDF5 gzip compression level, 0-9. Only valid if gzip compression is used. Level 4 (default) recommended for best balance between compression and time.

Notes

The maximum attributes will be stored. Data structure will be similar has it is in masterChannelList attribute. Dependency: h5py

```
export_to_matlab (file_name=None)
```

Export mdf data into Matlab file preferrably in format 7.3

Parameters file_name (str, optional) - file name. If no name defined, it will use original mdf name and path

Notes

This method will dump all data into Matlab file but you will loose below information: - unit and descriptions of channel - data structure, what is corresponding master channel to a channel. Channels might have then different lengths. Dependency: hdf5storage, scipy

export_to_parquet (file_name=None)

Exports mdf data into parquet file

1.3. mdfreader 7

Parameters file_name (str, optional) – file name. If no name defined, it will use original mdf name and path with .parquet extension

```
export_to_xlsx (file_name=None)
```

Exports mdf data into excel 2007 and 2010 file

Parameters file_name (str, optional) – file name. If no name defined, it will use original mdf name and path

Notes

It is recommended to export resampled data for performances Dependency: openpyxl

```
get_channel_data (channel_name, raw_data=False)
```

Return channel numpy array

Parameters

- channel_name (str) channel name
- raw_data (bool) flag to return non converted data

Returns converted, if not already done, data corresponding to channel name

Return type numpy array

Notes

This method is the safest to get channel data as numpy array from 'data' dict key might contain raw data

```
keep_channels (channel_set)
```

keeps only a list or set of channels and removes the other channels

Parameters channel set (list or set of str) – list or set of channel names

```
merge_mdf (mdf_class)
```

merge data of input mdf class with the current one.

Parameters mdf_class (Mdf) - mdf class instance to be merged with self

Notes

If there are common channel names between the 2 mdf, channels are renamed to make them unique

```
plot (channel_name_list_of_list)
```

Plot channels with Matplotlib

Parameters channel_name_list_of_list (str or list of str or list of list of str) – channel name or list of channel names or list of list of channel names list of list will create multiplots

Notes

Channel description and unit will be tentatively displayed with axis labels

```
plot_all()
```

read (file_name=None, multi_processed=False, channel_list=None, convert_after_read=True, filter_channel_names=False, no_data_loading=False, compression=False, metadata=2) reads mdf file version 3.x and 4.x

Parameters

- file_name (str, optional) file name
- multi_processed (bool) flag to activate multiprocessing of channel data conversion.
- **channel_list** (list of str, optional) list of channel names to be read. If you use channelList, reading might be much slower but it will save you memory. Can be used to read big files.
- **convert_after_read** (bool, optional) flag to convert channel after read, True by default. If you use convertAfterRead by setting it to false, all data from channels will be kept raw, no conversion applied. If many float are stored in file, you can gain from 3 to 4 times memory footprint. To calculate value from channel, you can then use method getChannelData()
- **filter_channel_names** (bool, optional) flag to filter long channel names from its module names separated by '.'
- no_data_loading (bool, optional) Flag to read only file info but no data to have minimum memory use.
- **compression** (bool or str, optional) To compress data in memory using blosc or bcolz, takes cpu time. if compression = int(1 to 9), uses bcolz for compression. if compression = 'blosc', uses blosc for compression. Choice given, efficiency depends of data.
- metadata (int, optional, default = 2) Reading metadata has impact on performance, especially for mdf 4.x using xml. 2: minimal metadata reading (mostly channel blocks). 1: used for noDataLoading. 0: all metadata reading, including Source Information, Attachment, etc..

Notes

If you keep convertAfterRead to true, you can set attribute mdf.multiProc to activate channel conversion in multiprocessing. Gain in reading time can be around 30% if file is big and using a lot of float channels

Warning: MultiProc use should be avoided when reading several files in a batch, it is not thread safe. You should better multi process instances of mdf rather than using multiproc in mdf class (see implementation of mdfconverter)

resample (sampling=None, channel=None, master_channel=None)

Resamples as much as possible all data groups into one data group having defined sampling interval or sharing same defined master channel

Parameters

• sampling(float, optional) -

resampling interval, None by default. If None, will rely on channel or master_channel parameters to define reference data group. If both are undefined, picking the first master

1.3. mdfreader 9

- or | and **(**)-
- channel (str, optional) channel name to be resampled
- or | and **-
- master_channel (str, optional) master channel name to be used as reference

- 1. resampling will be applied only to master channels that have same type as the one given by channel or master_channel parameters (applicable only to mdf4)
- 2. resampling will convert all your channels so be careful for big files and memory consumption

resample_group (sampling, channel, new_master_data=None)

Resamples one channel along with its dataGroup

Parameters

- **sampling** (*float*) resampling interval
- **channel** (str) channel name to be resampled (could the master channel)
- new_master_data (array, optional) master channel data to be applied to the group identified by channel

Notes

Resampling will convert all channels so be careful for big files and memory consumption

return_pandas_dataframe (master_channel_name)

returns a dataframe of a raster described by its master channel name

Parameters master_channel_name (str) – master channel name, key to a raster to be returned as pandas dataframe

Returns

Return type pandas dataframe of raster or data group

write (file_name=None, compression=False, column_oriented=False)

Writes simple mdf file, same format as originally read, default is 4.x

Parameters

- **file_name** (*str*, *optional*) Name of file If file name is not input, written file name will be the one read with appended '_new' string before extension
- **compression** (bool) Flag to store data compressed (from mdf version 4.1) If activated, will write in version 4.1 even if original file is in version 3.x
- column_oriented (bool) Flag to store, column oriented channels

Notes

All channels will be converted, so size might be bigger than original file

class mdfreader.mdfreader.MdfInfo() -> new empty dictionary dict(mapping) -> new dictionary initialized from a mapping object's (key, value) pairs
dict(iterable) -> new dictionary initialized as if via: d = {}
for k, v in iterable: d[k] = v dict(**kwargs) -> new dictionary initialized with the name=value pairs in the keyword
argument list. For example: dict(one=1, two=2)

Bases: dict

fid

fileName

filterChannelNames

list_channels(file_name=None)

Read MDF file blocks and returns a list of contained channels

Parameters file_name (string) - file name

Returns nameList – list of channel names

Return type list of string

mdfversion

read_info (file_name=None, fid=None, minimal=0)

Reads MDF file and extracts its complete structure

Parameters

- file_name (str, optional) file name. If not input, uses fileName attribute
- fid(file identifier, optional)-
- minimal (int) 0 will load every metadata 1 will load DG, CG, CN and CC 2 will load only DG

zipfile

1.3. mdfreader 11

MDF MODULE DOCUMENTATION

mdf_skeleton module describing basic mdf structure and methods

Created on Thu Sept 24 2015

Author Aymeric Rateau

2.1 Dependencies

Bases: object

- Python >3.4 http://www.python.org
- Numpy >1.6 http://numpy.scipy.org

class mdfreader.mdf.CompressedData

2.2 mdf

```
compression(a)
           data compression method
               Parameters a (numpy array) – data to be compresses
      data
      decompression()
           data decompression
               Returns
               Return type uncompressed numpy array
      dtype
class mdfreader.mdf.MdfSkeleton() -> new empty dictionary dict(mapping) -> new dictionary ini-
                                              tialized from a mapping object's (key, value) pairs dict(iterable)
                                              -> new dictionary initialized as if via: d = \{\} for k, v in iterable:
                                              d[k] = v \operatorname{dict}(**kwargs) \rightarrow new \operatorname{dictionary} initialized with the
                                              name=value pairs in the keyword argument list. For example:
                                              dict(one=1, two=2)
      Bases: dict
      MDFVersionNumber
```

add_channel (channel_name, data, master_channel, master_type=1, unit=", description=", conversion=None, info=None, compression=False, identifier=None)
adds channel to mdf dict.

Parameters

- channel_name (str) channel name
- data (numpy array) numpy array of channel's data
- master_channel (str) master channel name
- master_type(int, optional) master channel type: 0=None, 1=Time, 2=Angle, 3=Distance, 4=index
- unit (str, optional) unit description
- description (str, optional) channel description
- conversion (info class, optional) conversion description from info class
- info(info class for CNBlock, optional) used for CABlock axis creation and channel conversion
- compression (bool) flag to ask for channel data compression
- **identifier** (tuple) tuple of int and str following below structure: (data group number, channel group number, channel number), (channel name, channel source, channel path), (group name, group source, group path)

add_metadata (author=", organisation=", project=", subject=", comment=", time=")
adds basic metadata to mdf class

Parameters

- author (str) author of file
- **organisation** (str) organisation of author
- project (str)-
- subject (str)-
- comment (str) -
- time (float (epoch)) -

Notes

All fields are optional, default being empty string

convertAfterRead

convertTables

copy()

copy a mdf class

Returns mdf skeleton – copy of a mdf skeleton class

Return type class instance

fid

fileMetadata

fileName

```
filterChannelNames
get_channel (channel_name)
    Extract channel dict from mdf structure
        Parameters channel_name (str) - channel name
        Returns
        Return type channel dictionnary containing data, description, unit, etc.
get_channel_conversion(channel_name)
    Extract channel conversion dict from mdf structure
        Parameters channel_name (str) - channel name
        Returns
        Return type channel conversion dict
get_channel_desc(channel_name)
    Extract channel description information from mdf structure
        Parameters channel_name (str) - channel name
        Returns
        Return type channel description string
get channel master(channel name)
    Extract channel master name from mdf structure
        Parameters channel_name (str) - channel name
        Returns
        Return type channel master name string
get_channel_master_type (channel_name)
    Extract channel master type information from mdf structure
        Parameters channel_name (str) - channel name
        Returns channel mater type integer
        Return type 0=None, 1=Time, 2=Angle, 3=Distance, 4=index
get_channel_unit (channel_name)
    Returns channel unit string Implemented for a future integration of pint
        Parameters channel_name (str) - channel name
        Returns unit string description
        Return type str
get_invalid_bit(channel_name)
get_invalid_channel(channel_name)
info
masterChannelList
multiProc
remove_channel (channel_name)
    removes channel from mdf dict.
```

2.2. mdf 15

```
Parameters channel_name (str) - channel name
```

Returns

Return type value of mdf dict key=channel_name

remove_channel_conversion(channel_name)

removes conversion key from mdf channel dict.

Parameters channel_name (str) - channel name

Returns

Return type removed value from dict

rename_channel (channel_name, new_name)

Modifies name of channel

Parameters

- channel_name (str) channel name
- new name (str) new channel name

set_channel_attachment (channel_name, attachment)

Modifies channel attachment

Parameters

- channel_name (str) channel name
- attachment channel attachment

set_channel_conversion(channel_name, conversion)

Modifies conversion dict of channel

Parameters

- channel_name (str) channel name
- conversion (dict) conversion dictionary

set_channel_data (channel_name, data, compression=False)

Modifies data of channel

Parameters

- $channel_name(str) channel name$
- data (numpy array) channel data
- compression (bool or str) trigger for data compression

set_channel_desc (channel_name, desc)

Modifies description of channel

Parameters

- $channel_name(str)$ channel name
- desc (str) channel description

set_channel_master (channel_name, master)

Modifies channel master name

Parameters

• channel name (str) - channel name

```
• master (str) - master channel name
```

set_channel_master_type (channel_name, master_type)

Modifies master channel type

Parameters

- **channel_name** (str) channel name
- master_type (int) master channel type

```
set_channel_unit (channel_name, unit)
```

Modifies unit of channel

Parameters

- channel_name (str) channel name
- unit (str) channel unit

set_invalid_bit (channel_name, bit_position)

returns the invalid bit position of channel

Parameters

- **channel_name** (str) channel name
- bit_position invalid bit position of channel within invalid channel bytes

Returns

Return type bit position

```
set_invalid_channel (channel_name, invalid_channel)
zipfile
```

2.2. mdf 17

MDF3READER MODULE DOCUMENTATION

Measured Data Format file reader module for version 3.x

Author Aymeric Rateau

Created on Sun Oct 10 12:57:28 2010

3.1 mdf3reader

class mdfreader.mdf3reader.DATA(fid, pointer)

Bases: dict

DATA class is organizing record classes itself made of channel. This class inherits from dict. Keys are corresponding to channel group recordID. A DATA class corresponds to a data block, a dict of record classes (one per channel group). Each record class contains a list of channel class representing the structure of channel record.

fid

file identifier

Type io.open

pointerToData

position of Data block in mdf file

Type int

BlockLength

total size of data block

Type int

add_record(record)

Adds a new record in DATA class dict

read (channelSet)

Reads data block

load_sorted(record, nameList=None)

Reads sorted data block from record definition

load unsorted(nameList=None)

Reads unsorted data block, not yet implemented

add_record(record)

Adds a new record in DATA class dict

Parameters record (class) - channel group definition listing record channel classes

```
load sorted(record, name list=None)
```

Reads sorted data block from record definition

Parameters

- record (class) channel group definition listing record channel classes
- name_list(set of str, optional) list of channel names

Returns

Return type numpy recarray of data

load_unsorted(name_set=None)

Reads unsorted data block from record definition

Parameters name_set (set of str, optional) - set of channel names

Returns

Return type numpy recarray of data

read (channel_set, file_name)

Reads data block

Parameters

- channel_set (set of str, optional) list of channel names
- **file_name** (str) name of file

Bases: mdfreader.mdf.MdfSkeleton

mdf file version 3.0 to 3.3 class

fileName

file name

Type str

MDFVersionNumber

mdf file version number

Type int

masterChannelList

Represents data structure: a key per master channel with corresponding value containing a list of channels One key or master channel represents then a data group having same sampling interval.

Type dict

multiProc

Flag to request channel conversion multi processed for performance improvement. One thread per data group.

Type bool

convertAfterRead

flag to convert raw data to physical just after read

Type bool

filterChannelNames

flag to filter long channel names from its module names separated by '.'

Type bool

fileMetadata

file metadata with minimum keys: author, organisation, project, subject, comment, time, date

Type dict

read3 (fileName=None, info=None, multiProc=False, channelList=None, convertAfterRead=True)
Reads mdf 3.x file data and stores it in dict

_get_channel_data3 (channelName)

Returns channel numpy array

_convert_channel3 (channelName)

converts specific channel from raw to physical data according to CCBlock information

_convert_all_channel3()

Converts all channels from raw data to converted data according to CCBlock information

write3 (fileName=None)

Writes simple mdf 3.3 file

read3 (file_name=None, info=None, multi_processed=False, channel_list=None, convert_after_read=True, filter_channel_names=False, compression=False, metadata=2)
Reads mdf 3.x file data and stores it in dict

Parameters

- **file_name** (str, optional) file name
- info (mdfinfo3.info3 class) info3 class containing all MDF Blocks
- multi_processed (bool) flag to activate multiprocessing of channel data conversion
- **channel_list** (*list of str*, *optional*) list of channel names to be read If you use channelList, reading might be much slower but it will save you memory. Can be used to read big files
- convert_after_read (bool, optional) flag to convert channel after read, True by default If you use convertAfterRead by setting it to false, all data from channels will be kept raw, no conversion applied. If many float are stored in file, you can gain from 3 to 4 times memory footprint To calculate value from channel, you can then use method .get_channel_data()
- **filter_channel_names** (bool, optional) flag to filter long channel names from its module names separated by '.'
- compression (bool, optional) flag to activate data compression with blosc
- metadata (int, optional, default = 2) Reading metadata has impact on performance, especially for mdf 4.x using xml. 2: minimal metadata reading (mostly channel blocks) 1: used for noDataLoading 0: all metadata reading

write3 (file_name=None)

Writes simple mdf 3.3 file

Parameters file_name (str, optional) – Name of file If file name is not input, written file name will be the one read with appended '_new' string before extension

3.1. mdf3reader 21

All channels will be converted to physical data, so size might be bigger than original file

class mdfreader.mdf3reader.Record(data_group, channel_group)

Bases: list

record class lists Channel classes, it is representing a channel group

CGrecordLength

length of record from channel group block information in Byte

Type int

recordLength

length of record from channels information in Byte

Type int

numberOfRecords

number of records in data block

Type int

recordID

recordID corresponding to channel group

Type int

recordIDnumber

size of recordID

Type int

dataGroup

data group number

Type int:

channelGroup

channel group number

Type int

numpyDataRecordFormat

list of numpy (dtype) for each channel

Type list

dataRecordName

list of channel names used for recarray attribute definition

Type list

master

define name and number of master channel

Type dict

recordToChannelMatching

helps to identify nested bits in byte

Type dict

channelNames

channel names to be stored, useful for low memory consumption but slow

```
Type set
```

hiddenBytes

flag in case of non declared channels in record

Type Bool, False by default

byte_aligned

flag for byte aligned record

Type Bool, True by default

addChannel (info, channelNumber)

loadInfo(info)

readSortedRecord (fid, pointer, channelSet=None)

readRecordBuf (buf, channelSet=None)

readRecordBits (bita, channelSet=None)

add channel (info, channel number)

add a channel in class

Parameters

- info (mdfinfo3.info3 class) -
- **channel_number** (*int*) **channel** number in mdfinfo3.info3 class

load_info(info)

gathers records related from info class

Parameters info (mdfinfo3.info3 class) -

read_record_bits (bit_stream, channel_set=None)

read stream of record bits by bits in case of not aligned or hidden bytes

Parameters

- bit_stream (stream) stream of bytes read in file
- channel_set (Set of str, optional) list of channel to read

Returns rec – returns dictionary of channel with its corresponding values

Return type dict

read_record_buf (buf, channel_set=None)

read stream of record bytes

Parameters

- **buf** (stream) stream of bytes read in file
- channel_set (Set of str, optional) list of channel to read

Returns rec – returns dictionary of channel with its corresponding values

Return type dict

read_sorted_record (fid, pointer, channel_set=None)

reads record, only one channel group per data group

Parameters

• fid - file identifier

3.1. mdf3reader 23

- pointer position in file of data block beginning
- channel_set (Set of str, optional) list of channel to read

Returns rec – contains a matrix of raw data in a recarray (attributes corresponding to channel name)

Return type numpy recarray

Notes

If channelSet is None, read data using numpy.core.records.fromfile that is rather quick. However, in case of large file, you can use channelSet to load only interesting channels or only one channel on demand, but be aware it might be much slower.

MDFINFO3 MODULE DOCUMENTATION

Measured Data Format blocks parser for version 3.x

Created on Thu Dec 9 12:57:28 2014

Author Aymeric Rateau

4.1 Dependencies

- Python >3.4 http://www.python.org
- Numpy >1.14 http://numpy.scipy.org

4.2 mdfinfo3

```
class mdfreader.mdfinfo3.Info3() -> new empty dictionary dict(mapping) -> new dictionary ini-
                                            tialized from a mapping object's (key, value) pairs dict(iterable)
                                            -> new dictionary initialized as if via: d = \{\} for k, v in iterable:
                                            d[k] = v \ dict(**kwargs) \rightarrow new \ dictionary \ initialized \ with \ the
                                            name=value pairs in the keyword argument list. For example:
                                            dict(one=1, two=2)
     Bases: dict
     {\tt clean\_dg\_info}(dg)
           delete CN,CC and CG blocks related to data group
               Parameters dg(int) – data group number
     fid
     fileName
     filterChannelNames
     list_channels3 (file_name=None, fid=None)
           reads data, channel group and channel blocks to list channel names
           file name
               file name
                   Type str
               Returns
               Return type list of channel names
```

read_cg_block (fid, dg, minimal=0) read all CG blocks and relying CN & CC

Parameters

- **fid** (float) file identifier
- dg (int) data group number
- minimal (int) 0 will load every metadata 1 will load DG, CG, CN and CC 2 will load only DG

read_info3 (fid, minimal=0)
 read all file blocks except data

Parameters

- **fid** (float) file identifier
- minimal (int) 0 will load every metadata 1 will load DG, CG, CN and CC 2 will load only DG

CHAPTER

FIVE

MDF4READER MODULE DOCUMENTATION

Measured Data Format file reader module for version 4.x.

Author Aymeric Rateau

Created on Thu Dec 10 12:57:28 2013

5.1 Dependencies

- Python >3.4 http://www.python.org
- Numpy >1.6 http://numpy.scipy.org
- · bitarray to parse bits in not aligned bytes
- · Sympy to convert channels with formula if needed
- · zlib to uncompress data block if needed

5.2 mdf4reader

class mdfreader.mdf4reader.Data()-> new empty dictionary dict(mapping) -> new dictionary initialized from a mapping object's (key, value) pairs dict(iterable) -> new dictionary initialized as if via: $d = \{\}$ for k, v in iterable: d[k] = v dict(**kwargs) -> new dictionary initialized with the name=value pairs in the keyword argument list. For example: dict(one=1, two=2)

Bases: dict

add_record(record)

Adds a new record in Data class dict.

Parameters record (class) – channel group definition listing record channel classes

fid

load (record, info, name_list=None, sorted_flag=True, vlsd=None)
Reads data block from record definition

Parameters

- record (class) channel group definition listing record channel classes
- info (class) contains blocks
- name_list(list of str, optional) list of channel names

- **sorted_flag** (bool, optional) flag to know if data block is sorted (only one Channel Group in block) or unsorted (several Channel Groups identified by a recordID). As unsorted block can contain CG records in random order, block is processed iteratively, not in raw like sorted -> much slower reading
- vlsd (array or None) indicate a sd block, compressed (DZ) or not (SD)

Returns

Return type numpy recarray of data

pointer_to_data

read (channel_set, info, filename)

Reads data block

Parameters

- channel_set (set of str) set of channel names
- info (info object) contains blocks structures
- filename name of file of read

read data list (field, nBytes, temps, record, info, name list, sorted flag, vlsd)

read_record (record_id, info, buf)

read record from a buffer

Parameters

- record id (int) record identifier
- info (class) contains blocks
- **buf** (str) buffer of data from file to be converted to channel raw data

type

Bases: mdfreader.mdf.MdfSkeleton

mdf file reader class from version 4.0 to 4.1.1

fileName

file name

Type str

MDFVersionNumber

mdf file version number

Type int

masterChannelList

Represents data structure: a key per master channel with corresponding value containing a list of channels One key or master channel represents then a data group having same sampling interval.

Type dict

multiProc

Flag to request channel conversion multi processed for performance improvement. One thread per data group.

Type bool

convertAfterRead

flag to convert raw data to physical just after read

Type bool

filterChannelNames

flag to filter long channel names from its module names separated by '.'

Type bool

fileMetadata

file metadata with minimum keys: author, organisation, project, subject, comment, time, date

Type dict

read4 (fileName=None, info=None, multiProc=False, channelList=None, convertAfterRead=True)
Reads mdf 4.x file data and stores it in dict

_get_channel_data_4 (channelName)

Returns channel numpy array

_convert_channel_data_4 (channel, channel_name, convert_tables, multiProc=False, Q=None) select right conversion and calculates it

_convert_channel_4 (channelName)

converts specific channel from raw to physical data according to CCBlock information

_convert_all_channel_4()

Converts all channels from raw data to converted data according to CCBlock information

write4 (file_name=None, compression=False)

writes mdf 4.1 file

apply_invalid_bit (channel_name)

mask data from invalid bit channel if existing

get_channel_name_4 (name, path)

returns a list of tuples

apply_all_invalid_bit()

Mask data of all channels based on its invalid bit definition if there is

apply_invalid_bit (channel_name)

Mask data of channel based on its invalid bit definition if there is

Parameters channel_name (str) - Name of channel

get channel name4 (name, path)

finds mdf channel name from name and path

Parameters

- name (str) channel name
- path (str) source path or name, or channel group name, source name or path

Returns

Return type list of tuples (channel_name, (ndg, ncg, ncn))

get_invalid_mask(channel_name)

read4 (file_name=None, info=None, multi_processed=False, channel_list=None, convert_after_read=True, filter_channel_names=False, compression=False, metadata=2)
Reads mdf 4.x file data and stores it in dict

5.2. mdf4reader 29

Parameters

- file_name (str, optional) file name
- info (mdfinfo4.info4 class) info4 class containing all MDF Blocks
- multi_processed (bool) flag to activate multiprocessing of channel data conversion
- **channel_list** (list of str, optional) list of channel names to be read If you use channelList, reading might be much slower but it will save you memory. Can be used to read big files
- **convert_after_read** (bool, optional) flag to convert channel after read, True by default If you use convertAfterRead by setting it to false, all data from channels will be kept raw, no conversion applied. If many float are stored in file, you can gain from 3 to 4 times memory footprint To calculate value from channel, you can then use method .get_channel_data()
- **filter_channel_names** (bool, optional) flag to filter long channel names from its module names separated by '.'
- compression (bool, optional) flag to activate data compression with blosc
- metadata (int, optional, default = 2) Reading metadata has impact on performance, especially for mdf 4.x using xml. 2: minimal metadata reading (mostly channel blocks) 1: used for noDataLoading 0: all metadata reading, including Source Information, Attachment, etc..

write4 (file_name=None, compression=False, column_oriented=False)
Writes simple mdf file

Parameters

- **file_name** (*str*, *optional*) Name of file If file name is not input, written file name will be the one read with appended '_new' string before extension
- compression (bool) flag to store data compressed
- **column_oriented** (bool) flag to store data in columns, faster reading channel by channel and not jumping in records

Notes

All channels will be converted to physical data, so size might be bigger than original file

class mdfreader.mdf4reader.Record() -> new empty dictionary dict(mapping) -> new dictionary initialized from a mapping object's (key, value) pairs
dict(iterable) -> new dictionary initialized as if via: d = {}
for k, v in iterable: d[k] = v dict(**kwargs) -> new dictionary initialized with the name=value pairs in the keyword
argument list. For example: dict(one=1, two=2)

Bases: dict

CANOpen

CGrecordLength

Flags

MLSD

VLSD

```
VLSD CG
add_channel (info, channel_number)
    add a channel in class
        Parameters
            • info(mdfinfo4.info4 class) -
            • channel_number (int) – channel number in mdfinfo4.info4 class
byte_aligned
channelGroup
channelNames
dataGroup
dataRecordName
generate_chunks()
    calculate data split
        Returns
        Return type (n_record_chunk, chunk_size)
hiddenBytes
initialise_recarray(info, channel_set, n_records, dtype=None, channels_indexes=None)
    Initialise recarray
        Parameters
            • info(info class)-
            • channel_set (set of str, optional) - set of channel to read
            • n_records (int) - number of records
            • dtype (numpy dtype, optional) -
            • channels_indexes(list of int, optional)-
        Returns rec – contains a matrix of raw data in a recarray (attributes corresponding to channel
            name)
        Return type numpy recarray
invalid_channel
load info(info)
    gathers records related from info class
        Parameters info (mdfinfo4.info4 class) -
master
numberOfRecords
numpyDataRecordFormat
read\_all\_channels\_sorted\_record(fid)
    reads all channels from file using numpy fromstring, chunk by chunk
        Parameters fid - file identifier
```

5.2. mdf4reader 31

Returns rec – contains a matrix of raw data in a recarray (attributes corresponding to channel name)

Return type numpy recarray

 $\begin{tabular}{ll} \textbf{read_channels_from_bytes} & (bit_stream, info, channel_set=None, n_records=None, dtype=None, channels_indexes=None) \end{tabular}$

reads stream of record bytes using dataRead module if available otherwise bitarray

Parameters

- bit_stream (stream) stream of bytes
- info(info class)-
- channel_set (set of str, optional) set of channel to read
- n_records (int) number of records
- dtype (numpy dtype) -
- channels_indexes(list of int)-

Returns rec – contains a matrix of raw data in a recarray (attributes corresponding to channel name)

Return type numpy recarray

read_channels_from_bytes_fallback (bit_stream, info, channel_set=None, n_records=None, dtype=None, channels_indexes=None)
reads stream of record bytes using bitarray in case no dataRead available

Parameters

- bit_stream (stream) stream of bytes
- info(info class)-
- channel_set (set of str, optional) set of channel to read
- n_records (int) number of records
- dtype (numpy dtype) -
- channels_indexes(list of int)-

Returns rec – contains a matrix of raw data in a recarray (attributes corresponding to channel name)

Return type numpy recarray

$\verb"read_not_all_channels_sorted_record" (\textit{fid}, \textit{info}, \textit{channel_set})$

reads channels from file listed in channelSet

Parameters

- fid file identifier
- info(info class) -
- channel_set (set of str, optional) set of channel to read

Returns rec – contains a matrix of raw data in a recarray (attributes corresponding to channel name)

Return type numpy recarray

```
read_record_buf (buf, info, channel_set=None)
read stream of record bytes
```

Parameters

- **buf** (stream) stream of bytes read in file
- info (class) contains blocks structure
- channel_set (set of str, optional) set of channel to read

Returns rec – returns dictionary of channel with its corresponding values

Return type dict

read_sorted_record (fid, info, channel_set=None)

reads record, only one channel group per datagroup

Parameters

- fid file identifier
- info info class
- channel_set (set of str, optional) set of channel to read

Returns rec – contains a matrix of raw data in a recarray (attributes corresponding to channel name)

Return type numpy recarray

Notes

If channelSet is None, read data using numpy.core.records.fromfile that is rather quick. However, in case of large file, you can use channelSet to load only interesting channels or only one channel on demand, but be aware it might be much slower.

read_unique_channel (fid, info)

reads all channels from file using numpy fromstring, chunk by chunk

Parameters

- fid file identifier
- info info class

Returns rec – contains a matrix of raw data in a recarray (attributes corresponding to channel name)

Return type numpy recarray

recordID

recordIDCFormat

recordIDsize

 ${\tt recordLength}$

recordToChannelMatching

unique_channel_in_DG

5.2. mdf4reader 33

MDFINFO4 MODULE DOCUMENTATION

Measured Data Format blocks parser for version 4.x

Created on Sun Dec 15 12:57:28 2013

Author Aymeric Rateau

6.1 mdfinfo4

```
class mdfreader.mdfinfo4.ATBlock (fid, pointer)
     Bases: dict
     reads Attachment block and saves in class dict
class mdfreader.mdfinfo4.CABlock
     Bases: dict
     reads Channel Array block and saves in class dict
     load (byte_offset_base)
     read(fid, pointer)
     write (fid)
class mdfreader.mdfinfo4.CCBlock
     Bases: dict
     reads Channel Conversion block and saves in class dict
     read_cc (fid, pointer)
class mdfreader.mdfinfo4.CGBlock (fid=None, pointer=None)
     Bases: dict
     reads Channel Group block and saves in class dict
     read_cg (fid, pointer)
     write (fid)
class mdfreader.mdfinfo4.CHBlock (fid, pointer)
     Bases: dict
     reads Channel Hierarchy block and saves in class dict
class mdfreader.mdfinfo4.CNBlock
     Bases: dict
     reads Channel block and saves in class dict
```

```
read_cn (**kargs)
write (fid)

class mdfreader.mdfinfo4.CommentBlock
    Bases: dict
    reads or writes Comment block and saves in class dict
    load (data, md_type)
    read_cm_at (fid, pointer)
    reads Comment block from attachment block
    Parameters
```

- **fid** file identifier
- pointer (int) position in file

read_cm_cc (fid, pointer)

reads Comment block from channel conversion block

Parameters

- **fid** file identifier
- pointer (int) position in file

read_cm_cc_unit (fid, pointer)

reads Comment block for channel conversion unit

Parameters

- **fid** file identifier
- pointer (int) position in file

 ${\tt read_cm_cg}\,(\mathit{fid},\mathit{pointer})$

reads Comment block from channel group block

Parameters

- **fid** file identifier
- pointer (int) position in file

 ${\tt read_cm_ch}\ (\mathit{fid}, \mathit{pointer})$

reads Comment block from file channel hierarchy block

Parameters

- **fid** file identifier
- pointer (int) position in file

read_cm_cn (fid, pointer, minimal=True)

reads Comment block from channel block

Parameters

- **fid** file identifier
- pointer (int) position in file
- minimal (boolean) flag to reduce metadata parsing

read_cm_cn_unit (fid, pointer)

reads Comment block for channel unit

Parameters

- **fid** file identifier
- pointer (int) position in file

read_cm_dg (fid, pointer)

reads Comment block from data group block

Parameters

- **fid** file identifier
- pointer (int) position in file

read_cm_ev (fid, pointer)

reads Comment block from event block

Parameters

- **fid** file identifier
- pointer (int) position in file

read_cm_fh (fid, pointer)

reads Comment block from file history block

Parameters

- **fid** file identifier
- pointer (int) position in file

read_cm_hd (fid, pointer)

reads Comment block from header block

Parameters

- **fid** file identifier
- pointer (int) position in file

read_cm_header (fid, pointer)

reads Comment block header

Parameters

- **fid** file identifier
- pointer (int) position in file

read cm si(fid, pointer)

reads Comment block from source information block

Parameters

- **fid** file identifier
- pointer (int) position in file

read_tx (fid, pointer)

reads TX block

Parameters

- **fid** file identifier
- pointer (int) position in file

6.1. mdfinfo4 37

```
read xml(fid)
           reads Comment block xml and objectifies it
                Parameters
                    • fid – file identifier
                    • block (Metadata) -
                    • normal 0 at end(removes)-
      write (fid)
class mdfreader.mdfinfo4.DGBlock (fid=None, pointer=None)
      Bases: dict
      reads Data Group block and saves in class dict
      read_dg (fid, pointer)
      write(fid)
class mdfreader.mdfinfo4.DIBlock() -> new empty dictionary dict(mapping) -> new dictio-
                                                nary initialized from a mapping object's (key, value) pairs
                                                dict(iterable) \rightarrow new dictionary initialized as if via: <math>d = \{\}
                                                for k, v in iterable: d[k] = v \ dict(**kwargs) -> new \ dictionary
                                                initialized with the name=value pairs in the keyword argument
                                                list. For example: dict(one=1, two=2)
      Bases: dict
      load (invalid_bytes, nRecords, pointer)
      write (fid, data)
class mdfreader.mdfinfo4.DLBlock
      Bases: dict
      reads List Data block
      read dl (fid, link count)
      write (fid, chunks)
class mdfreader.mdfinfo4.DTBlock() -> new empty dictionary dict(mapping) -> new dictio-
                                                nary initialized from a mapping object's (key, value) pairs
                                                dict(iterable) \rightarrow new dictionary initialized as if via: <math>d = \{\}
                                                for k, v in iterable: d[k] = v \operatorname{dict}(**kwargs) \rightarrow new \operatorname{dictionary}
                                                initialized with the name=value pairs in the keyword argument
                                                list. For example: dict(one=1, two=2)
      Bases: dict
      load (record_byte_offset, nRecords, pointer)
      write (fid, data)
class mdfreader.mdfinfo4.DVBlock() -> new empty dictionary dict(mapping) -> new dictio-
                                                nary initialized from a mapping object's (key, value) pairs
                                                dict(iterable) \rightarrow new dictionary initialized as if via: <math>d = \{\}
                                                for k, v in iterable: d[k] = v \operatorname{dict}(**kwargs) -> new \operatorname{dictionary}
                                                initialized with the name=value pairs in the keyword argument
                                                list. For example: dict(one=1, two=2)
      Bases: dict
      load (record_byte_offset, nRecords, pointer)
```

```
write(fid, data)
class mdfreader.mdfinfo4.DZBlock
     Bases: dict
     reads Data List block
     static decompress_data_block (block, zip_type, zip_parameter, org_data_length)
          decompress datablock.
              Parameters
                  • block (bytes) - raw data compressed
                  • zip\_type(int) - 0 for non transposed, 1 for transposed data
                  • zip_parameter (int) - first dimension of matrix to be transposed
                  • org_data_length (int) - uncompressed data length
              Returns
              Return type uncompressed raw data
     read_dz (fid)
     write (fid, data, record_length)
class mdfreader.mdfinfo4.EVBlock(fid, pointer)
     Bases: dict
     reads Event block and saves in class dict
class mdfreader.mdfinfo4.FHBlock (fid=None, pointer=None)
     Bases: dict
     reads File History block and save in class dict
     read (fid, pointer)
     write(fid)
class mdfreader.mdfinfo4.HDBlock(fid=None)
     Bases: dict
     reads Header block and save in class dict
     read (fid=None)
     write (fid)
class mdfreader.mdfinfo4.HLBlock
     Bases: dict
     reads Header List block
     load (record_byte_offset, n_records, position)
     read_hl(fid)
     write (fid, data)
class mdfreader.mdfinfo4.IDBlock(fid=None)
     Bases: dict
     reads or writes ID Block
     read(fid)
          reads IDBlock
```

6.1. mdfinfo4 39

```
write(fid)
```

Writes IDBlock

class mdfreader.mdfinfo4.Info4() -> new empty dictionary dict(mapping) -> new dictionary initialized from a mapping object's (key, value) pairs dict(iterable) -> new dictionary initialized as if via: $d = \{\}$ for k, v in iterable: d[k] = v dict(**kwargs) -> new dictionary initialized with the name=value pairs in the keyword argument list. For example: dict(one=1, two=2)

Bases: dict

${\tt clean_dg_info}\,(dg)$

delete CN,CC and CG blocks related to data group

Parameters dg(int) – data group number

fid

fileName

filterChannelNames

list_channels4 (file_name=None, fid=None)
Read MDF file and extract its complete structure

Parameters

- **file_name** (str) file name
- fid-

Returns

Return type list of channel names contained in file

read_cg_block (fid, dg, cg, pointer, vlsd_cg_block, channel_name_list=False, minimal=0) reads one Channel Group block

Parameters

- **fid** (float) file identifier
- **dg** (*int*) data group number
- cg (int) channel group number
- channel_name_list (bool) Flag to reads only channel blocks for listChannels4 method
- minimal (falg) to activate minimum content reading for raw data fetching

Returns vlsd_cg_block

Return type boolean

read_cg_blocks (fid, dg, channel_name_list=False, minimal=0) reads Channel Group blocks linked to same Data Block dg

Parameters

- **fid** (float) file identifier
- **dg** (*int*) data group number
- channel_name_list (bool) Flag to reads only channel blocks for listChannels4 method
- minimal(falg) to activate minimum content reading for raw data fetching

read_ch_block (fid, pointer)

reads channel hierarchy Blocks

Parameters

- fid (identifier) file identifier
- pointer (int) position of EVBlock in file

Returns

Return type channel hierarchy Blocks in a dict

read_cn_block (fid, pointer, dg, cg, mlsd_channels, vlsd, minimal, channel_name_list)
reads single Channel block

Parameters

- **fid** (float) file identifier
- pointer (int) position in file
- dg (int) data group number
- cg (int) channel group number in data group
- mlsd_channels (list of int) list of maximum length data channel numbers
- minimal (flag) to activate minimum content reading for raw data fetching
- channel_name_list (bool) Flag to reads only channel blocks for listChannels4 method

Returns

- cn (integer) channel number
- MLSDChannels list of appended Maximum Length Sampling Data channels
- vlsd (boolean)

read_cn_blocks (fid, dg, cg, channel_name_list=False, minimal=0)
reads Channel blocks link to CG Block

Parameters

- **fid** (float) file identifier
- dg (int) data group number
- $\operatorname{\mathbf{cg}}(\operatorname{int})$ channel group number in data group
- channel_name_list (bool) Flag to reads only channel blocks for listChannels4 method
- minimal (flag) to activate minimum content reading for raw data fetching

Returns vlsd

Return type boolean

read_dg_block (fid, channel_name_list=False, minimal=0)
reads Data Group Blocks

Parameters

• **fid** (float) – file identifier

6.1. mdfinfo4 41

- channel_name_list (bool) Flag to reads only channel blocks for listChannels4 method
- minimal (falg) to activate minimum content reading for raw data fetching

static read_ev_block (fid, pointer)

reads Events Blocks

Parameters

- fid (identifier) file identifier
- pointer (int) position of EVBlock in file

Returns

Return type Event Blocks in a dict

read_info (fid, minimal)

read all file blocks except data

Parameters

- **fid** (identifier) file identifier
- minimal (flag) to activate minimum content reading for raw data fetching

static read_sr_block (fid, pointer)

reads Sample Reduction Blocks

Parameters

- **fid** (float) file identifier
- pointer (int) position of SRBlock in file

Returns

Return type Sample Reduction Blocks in a dict

```
unique_id (ndg, ncg, ncn)
generate unique id tuples
```

Parameters

- **ndg** (*int*) data group number
- ncg(int) channel group number
- ncn (int) channel number

Returns tuples – (channel name, channel source, channel path), (group name, group source, group path)

Return type (data group number, channel group number, channel number),

zipfile

```
class mdfreader.mdfinfo4.LDBlock
    Bases: dict
    reads List Data block
    load (record_byte_offset, n_records, position, invalid_bytes=0, column_oriented_flag=False)
    read_ld(fid, link_count)
    write(fid, data, invalid_data=None, compression_flag=False)
```

class mdfreader.mdfinfo4.SIBlock

Bases: dict

reads Source Information block and saves in class dict

read_si (fid, pointer)

class mdfreader.mdfinfo4.SRBlock (fid, pointer)

Bases: dict

reads Sample Reduction block and saves in class dict

6.1. mdfinfo4 43

CHANNEL MODULE DOCUMENTATION

```
Measured Data Format file reader module.
     Author Aymeric Rateau
Created on Wed Oct 04 21:13:28 2017
mdfreader.channel.PythonVersion
     Python version currently running, needed for compatibility of both python 3.4+
          Type float
mdfreader.channel.channel
class mdfreader.channel.Channel3(info,
                                                   data_group,
                                                                 channel_group,
                                                                                   channel_number,
                                            record_id_number)
     Bases: object
     Channel class gathers all about channel structure in a record
     name
          Name of channel
              Type str
     unit
          channel unit
              Type str, default empty string
     desc
          channel description
              Type str
     conversion
          conversion dictionnary
              Type info class
     channelNumber
          channel number corresponding to mdfinfo3.info3 class
              Type int
     signalDataType
          signal type according to specification
```

Type int

bitCount

number of bits used to store channel record

Type int

nBytes_aligned

number of bytes (1 byte = 8 bits) taken by channel record

Type int

dataFormat

numpy dtype as string

Type str

CFormat

struct instance to convert from C Format

Type struct class instance

byteOffset

position of channel record in complete record in bytes

Type int

bitOffset

bit position of channel value inside byte in case of channel having bit count below 8

Type int

recAttributeName

channel name compliant to a valid python identifier (recarray attribute)

Type str

RecordFormat

dtype format used for numpy.core.records functions ((name_title,name),str_stype)

Type list of str

channelType

channel type

Type int

posByteBeg

start position in number of bit of channel record in complete record

Type int

posByteEnd

end position in number of bit of channel record in complete record

Type int

bit_masking_needed

True if bit masking needed after data read

Type bool, default false

__init__ (info, dataGroup, channelGroup, channelNumber, recordIDnumber)
constructor

__str__()

to print class attributes

```
rename duplicated channel name within unsorted channel groups
     change_channel_name (channel_group)
          In case of duplicate channel names within several channel groups for unsorted data, rename channel name
              Parameters channel group (int) - channelGroup number
class mdfreader.channel.Channel4(data_group, channel_group, channel_number)
     Bases: object
     CANOpen_offset()
          CANopen channel bytes offset
              Returns
              Return type integer, channel bytes offset
     VLSD_CG_Flag
     attachment (fid, info)
          In case of sync channel attached to channel
              Parameters
                  • fid (class) – file identifier
                  • info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
              Returns
              Return type ATBlock class from mdfinfo4 module
     bit_count (info)
          calculates channel number of bits
              Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
              Returns
              Return type integer corresponding to channel number of bits
     bit_masking_need(info)
          Valid if bit masking need
              Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
              Return type boolean True if channel needs bit masking, otherwise False
     bit offset (info)
          channel data bit offset in record
              Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
              Return type integer, channel bit offset
     byteOffset
     c_format (info)
          channel data C format
              Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
              Returns
```

change_channel_name (channel_group)

Return type string data C format

c_format_structure (info)

channel data C format struct object

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

Return type string data C format struct object

ca_block (info)

Extracts channel CA Block from info4

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

Return type CABlock object from mdfinfo4 module

calc_byte_offset (info)

channel data bytes offset in record (without record id)

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

Return type integer, channel bytes offset

calc_bytes (info, aligned=True)

calculates channel aligned bytes number

Parameters

- info (mdfinfo4.info4 class) info4 class containing all MDF Blocks
- aligned (boolean) with or without aligned bytes

Returns

Return type number of bytes integer

change_channel_name(channel_group)

In case of duplicate channel names within several channel groups for unsorted data, rename channel name

Parameters channel_group (int) - channelGroup number

channelGroup

channelNumber

channel sync type (info)

Extracts channel sync type from info4

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

- integer corresponding to channel sync type
- 0 no sync, normal data
- 1 time
- 2 angle
- 3 distance
- 4 index

channel_type (info)

Extracts channel type from info4

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

- integer describing channel type
- 0 normal channel
- 1 variable length
- 2 master channel
- 3 virtual master channel
- 4 sync channel
- 5 max length data
- 6 virtual data channel

cn block (info)

channel block

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

Return type CNBlock class from mdfinfo4 module

${\tt conversion}\ (\mathit{info})$

channel conversion CCBlock

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

Return type CCBlock

data(info)

returns data block pointer for VLSD, MLD or sync channels

dataGroup

data_format (info)

channel numpy.core.records data format

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

Return type string data format

desc(info)

channel description

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

Return type channel description string

has_invalid_bit(info)

invalid bit (info)

extracts from info4 the channels valid bits positions

```
Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
         Returns
         Return type channel valid bit position
is_ca_block (info)
isnumeric(info)
     check this is numeric channel from data type
         Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
         Returns
         Return type boolean, true if numeric channel, otherwise false
little_endian(info)
     check if channel is little endian
         Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
         Returns
         Return type boolean
nBytes_aligned
name
native data format(info)
numpy format (info)
     channel numpy.core.records data format
         Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
         Returns endian, dataType
         Return type string data format
pos_bit_beg
pos_bit_begin(info)
     channel data bit starting position in record
         Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
         Returns
         Return type integer, channel bit starting position
pos bit end(info)
     channel data bit ending position in record
         Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
         Returns
         Return type integer, channel bit ending position
pos_byte_beg(info)
     channel data bytes starting position in record
         Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks
         Returns
         Return type integer, channel bytes starting position
```

pos_byte_end(info)

channel data bytes ending position in record

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

Return type integer, channel bytes ending position

record_attribute_name()

clean up channel name from unauthorised characters

Returns

Return type channel name compliant to python attributes names (for recarray)

record_id_size(info)

Extracts record id size from info4

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

- integer describing record id size
- 0 no record id used
- 1 uint8
- 2 uint16
- 4 uint32
- 8 uint64

set (info)

channel initialisation

Parameters info (mdfinfo4.info4 class) -

set_CANOpen (info, name)

CANOpen channel intialisation

Parameters

- info (mdfinfo4.info4 class) info4 class containing all MDF Blocks
- name (str) name of channel. Should be in ('ms', 'day', 'days', 'hour', 'month', 'minute', 'year')

set_invalid_bytes(info)

invalid bytes channel initialisation

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

signal_data_type (info, byte_aligned=True)

extract signal data type from info4 class

Parameters

- info (mdfinfo4.info4 class) info4 class containing all MDF Blocks
- byte_aligned (bool) flag activated if channel is part of a record byte aligned

Returns

- integer corresponding to channel data type
- 0 unsigned integer little endian

- 1 unsigned integer big endian
- 2 signed integer little endian
- 3 signed integer big endian
- 4 float little endian
- 5 float big endian
- 6 string latin
- 7 string utf-8
- 9 string utf-16
- 10 byte array
- 11 mime sample
- 12 mime stream
- 13 CANopen date
- 14 CANopen time
- 15 LE Complex
- 16 BE Complex

type

unit (info)

channel unit

Parameters info (mdfinfo4.info4 class) - info4 class containing all MDF Blocks

Returns

Return type channel unit string

mdfreader.channel.array_format4 (signal_data_type, number_of_bytes) function returning numpy style string from channel data type and number of bits

Parameters

- **signal_data_type** (*int*) channel data type according to specification
- number_of_bytes (int) number of bytes taken by channel data in a record

Returns endian, data_type – numpy dtype format used by numpy.core.records to read channel raw data

Return type str

mdfreader.channel.data_type_format4 (signal_data_type, number_of_bytes)
function returning C format string from channel data type and number of bits

Parameters

- **signal_data_type** (*int*) channel data type according to specification
- number_of_bytes (int) number of bytes taken by channel data in a record

Returns data_type - C format used by fread to read channel raw data

Return type str

CHAPTER

EIGHT

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

m

mdfreader.channel, 45 mdfreader.mdf, 13 mdfreader.mdf3reader, 19 mdfreader.mdf4reader, 27 mdfreader.mdfinfo3, 25 mdfreader.mdfinfo4, 35 mdfreader.mdfreader, 3

56 Python Module Index

INDEX

Symbols	В
init() (mdfreader.channel.Channel3 method),	<pre>bit_count() (mdfreader.channel.Channel4 method), 47</pre>
str() (mdfreader.channel.Channel3 method), 46 _convert_all_channel3() (mdf-	<pre>bit_masking_need() (mdfreader.channel.Channel4</pre>
reader.mdf3reader.Mdf3 method), 21 _convert_all_channel_4() (mdf-	bit_masking_needed (mdfreader.channel.Channel3 attribute), 46
reader.mdf4reader.Mdf4 method), 29	bit_offset() (mdfreader.channel.Channel4
_convert_channel3() (mdf-	method), 47
reader.mdf3reader.Mdf3 method), 21	bitCount (mdfreader.channel.Channel3 attribute), 45
_convert_channel_4() (mdf-	bitOffset (mdfreader.channel.Channel3 attribute), 46
reader.mdf4reader.Mdf4 method), 29	BlockLength (mdfreader.mdf3reader.DATA attribute),
_convert_channel_data_4() (mdf-	byte_aligned (mdfreader.mdf3reader.Record at-
<pre>reader.mdf4reader.Mdf4 method), 29 _get_channel_data3()</pre>	tribute), 23
reader.mdf3reader.Mdf3 method), 21	byte_aligned (mdfreader.mdf4reader.Record at-
_get_channel_data_4() (mdf-	tribute), 31
reader.mdf4reader.Mdf4 method), 29	byteOffset (mdfreader.channel.Channel3 attribute),
-	46
A	byteOffset (mdfreader.channel.Channel4 attribute),
add_channel() (mdfreader.mdf.MdfSkeleton	47
method), 13	_
memoa), 13	
add_channel() (mdfreader.mdf3reader.Record	C
add_channel() (mdfreader.mdf3reader.Record method), 23	c_format() (mdfreader.channel.Channel4 method),
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record	c_format() (mdfreader.channel.Channel4 method), 47
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record method), 31	<pre>c_format() (mdfreader.channel.Channel4 method),</pre>
<pre>add_channel() method), 23 add_channel() method), 31 add_metadata() (mdfreader.mdf3reader.Record</pre>	<pre>c_format() (mdfreader.channel.Channel4 method),</pre>
<pre>add_channel() method), 23 add_channel() method), 31 add_metadata() method), 14</pre> <pre>(mdfreader.mdf3reader.Record (mdfreader.mdf4reader.Record (mdfreader.mdf.MdfSkeleton</pre>	<pre>c_format() (mdfreader.channel.Channel4 method),</pre>
<pre>add_channel()</pre>	<pre>c_format() (mdfreader.channel.Channel4 method),</pre>
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record method), 31 add_metadata() (mdfreader.mdf.MdfSkeleton method), 14 add_record() (mdfreader.mdf3reader.DATA method), 19 add_record() (mdfreader.mdf4reader.Data method),	<pre>c_format() (mdfreader.channel.Channel4 method),</pre>
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record method), 31 add_metadata() (mdfreader.mdf.MdfSkeleton method), 14 add_record() (mdfreader.mdf3reader.DATA method), 19 add_record() (mdfreader.mdf4reader.Data method), 27	<pre>c_format() (mdfreader.channel.Channel4 method),</pre>
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record method), 31 add_metadata() (mdfreader.mdf.MdfSkeleton method), 14 add_record() (mdfreader.mdf3reader.DATA method), 19 add_record() (mdfreader.mdf4reader.Data method), 27 addChannel() (mdfreader.mdf3reader.Record	<pre>c_format() (mdfreader.channel.Channel4 method),</pre>
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record method), 31 add_metadata() (mdfreader.mdf.MdfSkeleton method), 14 add_record() (mdfreader.mdf3reader.DATA method), 19 add_record() (mdfreader.mdf4reader.Data method), 27 addChannel() (mdfreader.mdf3reader.Record method), 23	<pre>c_format() (mdfreader.channel.Channel4 method),</pre>
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record method), 31 add_metadata() (mdfreader.mdf.MdfSkeleton method), 14 add_record() (mdfreader.mdf3reader.DATA method), 19 add_record() (mdfreader.mdf4reader.Data method), 27 addChannel() (mdfreader.mdf3reader.Record method), 23 apply_all_invalid_bit() (mdf-	<pre>c_format() (mdfreader.channel.Channel4 method),</pre>
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record method), 31 add_metadata() (mdfreader.mdf.MdfSkeleton method), 14 add_record() (mdfreader.mdf3reader.DATA method), 19 add_record() (mdfreader.mdf4reader.Data method), 27 addChannel() (mdfreader.mdf3reader.Record method), 23 apply_all_invalid_bit() (mdfreader.mdf4reader.Mdf4 method), 29	<pre>c_format() (mdfreader.channel.Channel4 method),</pre>
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record method), 31 add_metadata() (mdfreader.mdf.MdfSkeleton method), 14 add_record() (mdfreader.mdf3reader.DATA method), 19 add_record() (mdfreader.mdf4reader.Data method), 27 addChannel() (mdfreader.mdf3reader.Record method), 23 apply_all_invalid_bit() (mdf-	c_format() (mdfreader.channel.Channel4 method), 47 c_format_structure() (mdf- reader.channel.Channel4 method), 48 ca_block() (mdfreader.channel.Channel4 method), 48 CABlock (class in mdfreader.mdfinfo4), 35 calc_byte_offset() (mdfreader.channel.Channel4 method), 48 calc_bytes() (mdfreader.channel.Channel4 method), 48 CANOpen (mdfreader.mdf4reader.Record attribute), 30 CANOpen_offset() (mdfreader.channel.Channel4 method), 47 CCBlock (class in mdfreader.mdfinfo4), 35
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record method), 31 add_metadata() (mdfreader.mdf.MdfSkeleton method), 14 add_record() (mdfreader.mdf3reader.DATA method), 19 add_record() (mdfreader.mdf4reader.Data method), 27 addChannel() (mdfreader.mdf3reader.Record method), 23 apply_all_invalid_bit() (mdfreader.mdf4reader.Mdf4 method), 29 apply_invalid_bit() (mdfreader.mdf4reader.Mdf4 method), 29 array_format4() (in module mdfreader.channel), 52	c_format() (mdfreader.channel.Channel4 method), 47 c_format_structure() (mdf- reader.channel.Channel4 method), 48 ca_block() (mdfreader.channel.Channel4 method), 48 CABlock (class in mdfreader.mdfinfo4), 35 calc_byte_offset() (mdfreader.channel.Channel4 method), 48 calc_bytes() (mdfreader.channel.Channel4 method), 48 CANOpen (mdfreader.mdf4reader.Record attribute), 30 CANOpen_offset() (mdfreader.channel.Channel4 method), 47 CCBlock (class in mdfreader.mdfinfo4), 35 CFormat (mdfreader.channel.Channel3 attribute), 46
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record method), 31 add_metadata() (mdfreader.mdf.MdfSkeleton method), 14 add_record() (mdfreader.mdf3reader.DATA method), 19 add_record() (mdfreader.mdf4reader.Data method), 27 addChannel() (mdfreader.mdf3reader.Record method), 23 apply_all_invalid_bit() (mdfreader.mdf4reader.Mdf4 method), 29 apply_invalid_bit() (mdfreader.mdf4reader.Mdf4 method), 29 array_format4() (in module mdfreader.channel), 52 ATBlock (class in mdfreader.mdfinfo4), 35	c_format() (mdfreader.channel.Channel4 method), 47 c_format_structure() (mdfreader.channel4 method), 48 ca_block() (mdfreader.channel.Channel4 method), 48 CABlock (class in mdfreader.mdfinfo4), 35 calc_byte_offset() (mdfreader.channel.Channel4 method), 48 calc_bytes() (mdfreader.channel.Channel4 method), 48 CANOpen (mdfreader.mdf4reader.Record attribute), 30 CANOpen_offset() (mdfreader.channel.Channel4 method), 47 CCBlock (class in mdfreader.mdfinfo4), 35 CFormat (mdfreader.channel.Channel3 attribute), 46 CGBlock (class in mdfreader.mdfinfo4), 35
add_channel() (mdfreader.mdf3reader.Record method), 23 add_channel() (mdfreader.mdf4reader.Record method), 31 add_metadata() (mdfreader.mdf.MdfSkeleton method), 14 add_record() (mdfreader.mdf3reader.DATA method), 19 add_record() (mdfreader.mdf4reader.Data method), 27 addChannel() (mdfreader.mdf3reader.Record method), 23 apply_all_invalid_bit() (mdfreader.mdf4reader.Mdf4 method), 29 apply_invalid_bit() (mdfreader.mdf4reader.Mdf4 method), 29 array_format4() (in module mdfreader.channel), 52	c_format() (mdfreader.channel.Channel4 method), 47 c_format_structure() (mdf- reader.channel.Channel4 method), 48 ca_block() (mdfreader.channel.Channel4 method), 48 CABlock (class in mdfreader.mdfinfo4), 35 calc_byte_offset() (mdfreader.channel.Channel4 method), 48 calc_bytes() (mdfreader.channel.Channel4 method), 48 CANOpen (mdfreader.mdf4reader.Record attribute), 30 CANOpen_offset() (mdfreader.channel.Channel4 method), 47 CCBlock (class in mdfreader.mdfinfo4), 35 CFormat (mdfreader.channel.Channel3 attribute), 46

CGrecordLength (mdfreader.mdf4reader.Record attribute), 30	<pre>convertAfterRead (mdfreader.mdf3reader.Mdf3 at- tribute), 20</pre>
change_channel_name() (mdf-	convertAfterRead (mdfreader.mdf4reader.Mdf4 at-
reader.channel.Channel3 method), 46, 47	tribute), 29
change_channel_name() (mdf-	convertTables (mdfreader.mdf.MdfSkeleton at-
reader.channel.Channel4 method), 48	tribute), 14
channel (in module mdfreader.channel), 45	copy () (mdfreader.mdf.MdfSkeleton method), 14
Channel3 (class in mdfreader.channel), 45	cut () (mdfreader.mdfreader.Mdf method), 6
Channel 4 (class in mdfreader.channel), 47	D
channel_sync_type() (mdf-	D
reader.channel.Channel4 method), 48	DATA (class in mdfreader.mdf3reader), 19
channel_type() (mdfreader.channel.Channel4	Data (class in mdfreader.mdf4reader), 27
method), 48	data (mdfreader.mdf.CompressedData attribute), 13
channelGroup (mdfreader.channel.Channel4 at-	data() (mdfreader.channel.Channel4 method), 49
tribute), 48	data_format() (mdfreader.channel.Channel4
channelGroup (mdfreader.mdf3reader.Record at-	method), 49
tribute), 22	data_type_format4() (in module mdf-
channelGroup (mdfreader.mdf4reader.Record at-	reader.channel), 52
tribute), 31	dataFormat (mdfreader.channel.Channel3 attribute),
channelNames (mdfreader.mdf3reader.Record at-	46
tribute), 22	dataGroup (mdfreader.channel.Channel4 attribute), 49
channelNames (mdfreader.mdf4reader.Record at-	dataGroup (mdfreader.mdf3reader.Record attribute),
tribute), 31	22
channelNumber (mdfreader.channel.Channel3 attribute), 45	dataGroup (mdfreader.mdf4reader.Record attribute),
	31
channelNumber (mdfreader.channel.Channel4 at- tribute), 48	dataRecordName (mdfreader.mdf3reader.Record attribute), 22
channelType (mdfreader.channel.Channel3 attribute),	dataRecordName (mdfreader.mdf4reader.Record at-
46	tribute), 31
CHBlock (class in mdfreader.mdfinfo4), 35	decompress_data_block() (mdf-
clean_dg_info() (mdfreader.mdfinfo3.Info3	reader.mdfinfo4.DZBlock static method),
method), 25	39
clean_dg_info() (mdfreader.mdfinfo4.Info4	decompression() (mdfreader.mdf.CompressedData
method), 40	method), 13
<pre>cn_block() (mdfreader.channel.Channel4 method),</pre>	desc (mdfreader.channel.Channel3 attribute), 45
49	desc() (mdfreader.channel.Channel4 method), 49
CNBlock (class in mdfreader.mdfinfo4), 35	DGBlock (class in mdfreader.mdfinfo4), 38
CommentBlock (class in mdfreader.mdfinfo4), 36	DIBlock (class in mdfreader.mdfinfo4), 38
CompressedData (class in mdfreader.mdf), 13	DLBlock (class in mdfreader.mdfinfo4), 38
compression() (mdfreader.mdf.CompressedData	DTBlock (class in mdfreader.mdfinfo4), 38
method), 13	dtype (mdfreader.mdf.CompressedData attribute), 13
concat_mdf() (mdfreader.mdfreader.Mdf method), 5	DVBlock (class in mdfreader.mdfinfo4), 38
conversion (mdfreader.channel.Channel3 attribute),	DZBlock (class in mdfreader.mdfinfo4), 39
45	Г
conversion() (mdfreader.channel.Channel4	E
method), 49	EVBlock (class in mdfreader.mdfinfo4), 39
convert_all_channel() (mdf-	<pre>export_to_csv() (mdfreader.mdfreader.Mdf</pre>
reader.mdfreader.Mdf method), 4	method), 4, 6
convert_all_channels() (mdf-	<pre>export_to_excel() (mdfreader.mdfreader.Mdf</pre>
reader.mdfreader.Mdf method), 6	method), 4, 7
convert_to_pandas() (mdfreader.mdfreader.Mdf method), 5, 6	export_to_hdf5() (mdfreader.mdfreader.Mdf
convertAfterRead (mdfreader.mdf.MdfSkeleton at-	method), 4, 7
tribute), 14	export_to_matlab() (mdfreader.mdfreader.Mdf
vi vo vivo j, 1 i	method) 4 7

export_to_NetCDF() method), 4, 6	(mdfreader.mdfreader.Mdf	<pre>get_channel_desc() method), 15</pre>	(mdfreader.mdf.MdfSkeleton
export_to_parquet() method),7	(mdfreader.mdfreader.Mdf	<pre>get_channel_master reader.mdf.MdfS.</pre>	(mdf- keleton method), 15
export_to_xlsx() method), 4, 8	(mdfreader.mdfreader.Mdf	<pre>get_channel_master reader.mdf.MdfS</pre>	type() (mdf- keleton method), 15
F		get_channel_name4() (mdf- er.Mdf4 method), 29
FHBlock (class in mdfreade	rmdfinfod) 30	get_channel_name_4	
fid (mdfreader.mdf.MdfSkel			er.Mdf4 method), 29
fid (mdfreader.mdf3reader.I		<pre>get_channel_unit()</pre>	
fid (mdfreader.mdf4reader.l		<i>method</i>), 15	(**************************************
fid (mdfreader.mdfinfo3.Info		<pre>get_channel_unit()</pre>	(mdfreader.mdfreader.Mdf
fid (mdfreader.mdfinfo4.Info		<i>method</i>), 4	(yyy
fid (mdfreader.mdfreader.M		<pre>get_invalid_bit()</pre>	(mdfreader.mdf.MdfSkeleton
file_name (mdfreader.mdf		<i>method</i>), 15	(**************************************
fileMetadata (mdfreader		get_invalid_channe	el () (mdf-
14	.maj.MajSketeton attribute),	=	keleton method), 15
= :	uder.mdf3reader.Mdf3 at-		(mdfreader.mdf4reader.Mdf4
	nder.mdf4reader.Mdf4 at-	Н	
fileMetadata (mdfreader	:mdfreader.Mdf attribute), 4	has_invalid_bit()	(mdfreader.channel.Channel4
fileName (mdfreader.mdf.M		method), 49	•
fileName (mdfreader.mdf3		HDBlock (class in mdfred	ader.mdfinfo4), 39
fileName (mdfreader.mdf4			eader.mdf3reader.Record at-
fileName (mdfreader.mdfin		tribute), 23	J
fileName (mdfreader.mdfin			eader.mdf4reader.Record at-
fileName (mdfreader.mdfre		tribute), 31	•
fileName (mdfreader.mdfre		HLBlock (class in mdfred	ader.mdfinfo4), 39
filterChannelNames		` '	3 3 //
attribute), 14		1	
	mdfreader.mdf3reader.Mdf3	IDBlock (class in mdfred	udermdfinfo4) 39
attribute), 20		info (mdfreader.mdf.Mdf.	
	mdfreader.mdf4reader.Mdf4	Info3 (class in mdfreade	
attribute), 29	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	Info4 (class in mdfreade)	
filterChannelNames (n	ndfreader.mdfinfo3.Info3 at-	initialise_recarra	
tribute), 25			er.Record method), 31
filterChannelNames (n tribute), 40	ndfreader.mdfinfo4.Info4 at-	invalid_bit()	(mdfreader.channel.Channel4
filterChannelNames	(mdf-	method), 49	16
	IdfInfo attribute), 11		lfreader.mdf4reader.Record at-
Flags (mdfreader.mdf4read		tribute), 31	/ 16 1 1 1 1 1 1
G	er.Recora annome), 50	is_ca_block() method), 50	(mdfreader.channel.Channel4
			er.channel.Channel4 method),
generate_chunks() (m method),31	dfreader.mdf4reader.Record	50	
get_channel() method), 15	(mdfreader.mdf.MdfSkeleton	K keep_channels()	(mdfreader.mdfreader.Mdf
get_channel_conversi reader.mdf.MdfSkel		method), 5, 8	, , , , , , , , , , , , , , , , , , ,
get_channel_data()	(mdfreader.mdfreader.Mdf	L	
method), 4, 8		LDBlock (class in mdfred	ider mdfinfo4) 12
,, ., .		ביים ביים (ciass in majred	

list_channels() (mdfreader.mdfreader.MdfInfo method), 11	MDFVersionNumber (<i>mdfreader.mdf.MdfSkeleton attribute</i>), 13
list_channels3() (mdfreader.mdfinfo3.Info3 method), 25	MDFVersionNumber (mdfreader.mdf3reader.Mdf3 attribute), 20
list_channels4() (mdfreader.mdfinfo4.Info4 method), 40	MDFVersionNumber (mdfreader.mdf4reader.Mdf4 attribute), 28
little_endian() (mdfreader.channel.Channel4 method), 50	MDFVersionNumber (mdfreader.mdfreader.Mdf attribute), 4
load() (mdfreader.mdf4reader.Data method), 27 load() (mdfreader.mdfinfo4.CABlock method), 35 load() (mdfreader.mdfinfo4.CommentBlock method), 36 load() (mdfreader.mdfinfo4.DIBlock method), 38 load() (mdfreader.mdfinfo4.DTBlock method), 38 load() (mdfreader.mdfinfo4.DVBlock method), 38	merge_mdf() (mdfreader.mdfreader.Mdf method), 8 MLSD (mdfreader.mdf4reader.Record attribute), 30 multiProc (mdfreader.mdf.MdfSkeleton attribute), 15 multiProc (mdfreader.mdf3reader.Mdf3 attribute), 20 multiProc (mdfreader.mdf4reader.Mdf4 attribute), 28 multiProc (mdfreader.mdfreader.Mdf attribute), 4
load() (mdfreader.mdfinfo4.HLBlock method), 39	N
<pre>load() (mdfreader.mdfinfo4.LDBlock method), 42 load_info() (mdfreader.mdf3reader.Record method),</pre>	name (mdfreader.channel.Channel3 attribute), 45 name (mdfreader.channel.Channel4 attribute), 50 native_data_format() (mdf- reader.channel.Channel4 method), 50
31 load_sorted() (mdfreader.mdf3reader.DATA	nBytes_aligned (<i>mdfreader.channel.Channel3 attribute</i>), 46
method), 19 load_unsorted() (mdfreader.mdf3reader.DATA method), 19, 20	nBytes_aligned (mdfreader.channel.Channel4 at- tribute), 50
loadInfo() (mdfreader.mdf3reader.Record method), 23	numberOfRecords (mdfreader.mdf3reader.Record at- tribute), 22 numberOfRecords (mdfreader.mdf4reader.Record at-
M	tribute), 31
master (mdfreader.mdf3reader.Record attribute), 22	numpy_format() (mdfreader.channel.Channel4
master (mdfreader.mdf4reader.Record attribute), 31	<pre>method), 50 numpyDataRecordFormat (mdf-</pre>
masterChannelList (mdfreader.mdf.MdfSkeleton attribute), 15	reader.mdf3reader.Record attribute), 22 numpyDataRecordFormat (mdf-
masterChannelList (mdfreader.mdf3reader.Mdf3 attribute), 20	reader.mdf4reader.Record attribute), 31
masterChannelList (mdfreader.mdf4reader.Mdf4 attribute), 28	P plot() (mdfreader.mdfreader.Mdf method), 4, 8
masterChannelList (mdfreader.mdfreader.Mdf at- tribute), 4 Mdf (class in mdfreader.mdfreader), 3	plot_all() (mdfreader.mdfreader.Mdf method), 8 pointer_to_data (mdfreader.mdf4reader.Data at- tribute), 28
Mdf3 (class in mdfreader.mdf3reader), 20 Mdf4 (class in mdfreader.mdf4reader), 28	pointerToData (mdfreader.mdf3reader.DATA attribute), 19
MdfInfo (class in mdfreader.mdfreader), 10 mdfreader.channel (module), 45	<pre>pos_bit_beg (mdfreader.channel.Channel4 attribute), 50</pre>
mdfreader.mdf (module), 13 mdfreader.mdf3reader (module), 19	pos_bit_begin() (mdfreader.channel.Channel4 method), 50
mdfreader.mdf4reader(module), 27 mdfreader.mdfinfo3(module), 25 mdfreader.mdfinfo4(module), 35	pos_bit_end() (mdfreader.channel.Channel4 method), 50
mdfreader.mdfinfo4 (<i>module</i>), 35 mdfreader.mdfreader (<i>module</i>), 3 MdfSkeleton (<i>class in mdfreader.mdf</i>), 13	pos_byte_beg() (mdfreader.channel.Channel4 method), 50
matsketeton (class in majreader.maj), 13 mafversion (mdfreader.mdfreader.MdfInfo attribute), 11	pos_byte_end() (mdfreader.channel.Channel4 method), 50
11	posByteBeg (mdfreader.channel.Channel3 attribute), 46

PythonVersion (in module mdfreader.channel), 45 Read() (mdfreader.mdf3reader.DATA method), 19, 20 read() (mdfreader.mdfinfo-LCABlock method), 38 read() (mdfreader.mdfinfo-LABlock method), 39 read() (mdfreader.mdfinfo-LABlock method), 39 read() (mdfreader.mdfinfo-LABlock method), 39 read() (mdfreader.mdfinfo-LABlock method), 39 read() (mdfreader.mdfinfo-LBBlock method), 39 read() (mdfreader.mdfareader.Mdf method), 29 read_all_channels_sorted_vecord() (mdfreader.mdfinfo-LCBlock method), 35 read_cc_block() (in module mdfreader.mdfinfo-3), 36 read_cg_block() (in module mdfreader.mdfinfo-3), 37 read_cg_block() (in module mdfreader.mdfinfo-3), 35 read_cg_block() (in module mdfreader.mdfinfo-3), 35 read_cg_block() (in module mdfreader.mdfinfo-3), 36 read_cg_block() (in module mdfreader.mdfinfo-3), 36 read_cg_block() (in module mdfreader.mdfinfo-3), 37 read_cm_block() (in module mdfreader.mdfinfo-3), 36 read_cg_block() (indfreader.mdfinfo-4), 37 read_cm_block() (in module mdfreader.mdfinfo-3), 36 read_cg_block() (in module mdfreader.mdfinfo-3), 36 read_cg_block() (indfreader.mdfinfo-4), 36 read_cg_block() (indfreader.mdfinfo-4), 37 read_cd_block() (in module mdfreader.mdfinfo-3), 36 read_cg_block() (indfreader.mdfinfo-4), 36 read_cg_block() (indfreader.mdfinfo-4), 37 read_cg_block() (indfreader.mdfinfo-4), 38 read_dg_block() (in module mdfreader.mdfinfo-3), 36 read_cg_block() (indfreader.mdfinfo-4), 36 read_cg_block() (indfreader.mdfinfo-4), 37 read_cg_block() (indfreader.mdfinfo-4), 39 read_cg_block() (indfreader.mdfinfo-4), 39 read_cg_block() (indfreader.mdfinfo-4), 39 read_cg_block() (indfreader.mdfinfo-4),	posByteEnd (mdfreader.channel.Channel3 attribute), 46	read_cm_ev() (mdfreader.mdfinfo4.CommentBlock method), 37
method), 37 read() (mdfreader.mdffreader.DATA method), 19, 20 read() (mdfreader.mdffreader.DATA method), 28 read() (mdfreader.mdffnfo4.FHBlock method), 39 read() (mdfreader.mdffnfo4.FHBlock method), 39 read() (mdfreader.mdffnfo4.DBlock method), 39 read() (mdfreader.mdffnfo4.CBlock method), 39 read() (mdfreader.mdffnfo4.CBlock method), 31 read_c() (mdfreader.mdffnfo4.CBlock method), 32 read_c() (mdfreader.mdfinfo4.CBlock method), 35 read_c() (mdfreader.mdfinfo4.DBlock method), 30 read_c() () (mdf	PythonVersion (in module mdfreader.channel), 45	
read () (mdfreader.mdfirod-CABlock method), 39 read () (mdfreader.mdfinfod-LBlock method), 31 read_c () (mdfreader.mdfinfod-LCBlock method), 35 read_c () (mdfreader.mdfinfod-CBlock method), 36 read_c () (mdfreader.mdfinfod-LDlock method), 32 read_c () (mdfreader.mdfinfod-CmmentBlock method), 36 read_c () (mdfreader.mdfinfod-CommentBlock method), 36 read_c () (mdfreader.mdfinfo		
read() (mdfreader.mdfreader.DATA method), 28 read() (mdfreader.mdfreader.DATA method), 28 read() (mdfreader.mdfreader.DATA method), 39 read() (mdfreader.mdfinfo-l.FHBlock method), 39 read() (mdfreader.mdfinfo-l.DBlock method), 31 read_c() (mdfreader.mdfinfo-l.CBlock method), 35 read_c() (mdfreader.mdfinfo-l.CBlock method), 36 read_c() (mdfreader.mdfinfo-l.CBlock method), 32 read_c() (mdf	R	<pre>read_cm_hd() (mdfreader.mdfinfo4.CommentBlock</pre>
read() (mdfreadexmdfinfo4.CABlock method), 35 read() (mdfreadexmdfinfo4.FHBlock method), 39 read() (mdfreadexmdfinfo4.HDBlock method), 39 read() (mdfreadexmdfinfo4.HDBlock method), 39 read() (mdfreadexmdfireadexMdf method), 8 read() (mdfreadexmdfreadexMdf method), 8 read() (mdfreadexmdfreadexMdf method), 21 read() (mdfreadexmdfreadexMdf method), 22 read() (mdfreadexmdfinfo4.CCBlock method), 35 read() (mdfreadexmdfinfo4.CCBlock method), 36 read() (mdfreadexmdfinfo4.CCBlock method), 32 read() (mdfreadexmdfinfo4.CCBlock metho	read() (mdfreader.mdf3reader.DATA method), 19, 20	
read () (mdfreader.mdfinfo4.CABlock method), 39 read () (mdfreader.mdfinfo4.DBlock method), 39 read () (mdfreader.mdfiseader.Mdf method), 21 read () (mdfreader.mdfiseader.Mdf method), 21 read () (mdfreader.mdfiseader.Mdf method), 21 read () (mdfreader.mdfinfo4.Comtentblock method), 35 read () (mdfreader.mdfinfo4.CGBlock method), 35 read () (mdfreader.mdfinfo4.CGBlock method), 35 read () (mdfreader.mdfinfo4.CGBlock method), 35 read () (mdfreader.mdfinfo4.DGBlock method), 38 read () (mdfreader.mdfinfo4.DGBlock method), 39 read () (mdfreader.mdfinfo4.DGBlock		read_cm_header() (mdf-
read () (mdfreader.mdfinfo4.HBlock method), 39 read () (mdfreader.mdfinfo4.HBlock method), 39 read () (mdfreader.mdfinfo4.HBlock method), 39 read () (mdfreader.mdfareader.Mdf3 method), 21 read3 () (mdfreader.mdf3reader.Mdf3 method), 29 read_all_channels_sorted_record () (mdfreader.mdfinfo4.CBlock method), 31 read_cc_block () (in module mdfreader.mdfinfo3), 26 read_ce_block () (in module mdfreader.mdfinfo3), 26 read_cg_block () (in module mdfreader.mdfinfo4.Info4 method), 40 read_ch_block () (mdfreader.mdfinfo4.Info4 method), 30 read_cm_c () (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_d () (mdfreader.mdfinfo4.Comment		reader.mdfinfo4.CommentBlock method),
read() (mdfreader.mdfinfo4.HDBlock method), 39 read() (mdfreader.mdfinfo4.DBlock method), 39 read() (mdfreader.mdfinfo4.dBf3 method), 21 read() (mdfreader.mdf3reader.Mdf3 method), 21 read() (mdfreader.mdf3reader.Mdf3 method), 23 read_cc_block() (in module mdfreader.mdfinfo3), 26 read_cc_block() (in module mdfreader.mdfinfo3), 26 read_cg_block() (in module mdfreader.mdfinfo3), 36 read_cm_cl_ont() (mdfreader.mdfinfo4.Info4 method), 40 read_channels_from_bytes_fallback() (indfreader.mdfinfo4.DBlock method), 32 read_channels_from_bytes_fallback() (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (indfreader.mdfinfo		
read () (mdfreader.mdfinfo4.IDBlock method), 39 read () (mdfreader.mdfreader.Mdff method), 81 read () (mdfreader.mdfreader.Mdff method), 29 read all_channels_sorted_record () (mdfreader.mdfinfo4.CCBlock method), 31 read_cc_block () (in module mdfreader.mdfinfo3, 26 read_cc_block () (in module mdfreader.mdfinfo3.Info3 method), 40 method), 40 read_cc_block () (mdfreader.mdfinfo4.Info4 method), 40 read_cd_block () (mdfreader.mdfinfo4.Info4 method), 40 read_channels_from_bytes () (mdfreader.mdfinfo4.Info4 method), 41 read_channels_from_bytes () (mdfreader.mdfinfo4.Info4 method), 42 read_cm_at () (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_c () (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_co () (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit () (mdfreader.md		<pre>read_cm_si() (mdfreader.mdfinfo4.CommentBlock</pre>
read() (mdfreader.mdfreader.Mdff method), 81 read() (mdfreader.mdffreader.Mdfff method), 22 read_all_channels_sorted_record() (mdfreader.mdfinfo4.CRBlock method), 35 read_cc_block() (in module mdfreader.mdfinfo3), 26 read_ce_block() (in module mdfreader.mdfinfo3), 26 read_ce_block() (in module mdfreader.mdfinfo3), 26 read_cg_block() (in furfeader.mdfinfo3.Info3 method), 45 read_cg_block() (in furfeader.mdfinfo4.Info4 method), 41 read_channels_from_bytes() (mdfreader.mdfinfo4.Info4 method), 41 read_channels_from_bytes() (mdfreader.mdfinfo4.Info4 method), 42 read_channels_from_bytes_fallback() (inforeader.mdfinfo4.CommentBlock method), 36 read_cm_c() (indfreader.mdfinfo4.CommentBlock metho	· · · · · · · · · · · · · · · · · · ·	method), 37
read3() (mdfreader.mdf3reader.Mdf3 method), 29 read() (mdfreader.mdf4reader.Mdf4 method), 29 read() (mdfreader.mdfareader.Mdf4 method), 31 read() (mdfreader.mdfinfo4.CCBlock method), 35 read() (in module mdfreader.mdfinfo3), 26 read() (in module mdfreader.mdfinfo4.Info4 method), 28 read() (in module mdfreader.mdfinfo4.Info4 method), 28 read() (in module mdfreader.mdfinfo4.Info4 method), 28 read() (in module mdfreader.mdfinfo4.Info4 method), 20 read() (in module mdfreader.mdfinfo4.Info4 method), 28 read() (in module mdfreader.mdfinfo4.Info4 method), 20 read() (in module mdfreader.mdfinfo4.Info4 method), 28 read() (in module mdfreader.mdfinfo4.Info4 method), 20 read() (in module mdfre		<pre>read_cn() (mdfreader.mdfinfo4.CNBlock method), 35</pre>
read4() (mdfreader.mdffreader.Mdf4 method), 29 read_all_channels_sorted_record() (mdfreader.mdfinfo4.CCBlock method), 35 read_cc_block() (in module mdfreader.mdfinfo3), 26 read_cc_block() (in module mdfreader.mdfinfo3), 26 read_cc_block() (in module mdfreader.mdfinfo3), 26 read_cg_block() (in freader.mdfinfo4.Info4 method), 25 read_cg_block() (mdfreader.mdfinfo4.Info4 method), 40 read_cd_block() (mdfreader.mdfinfo4.Info4 method), 41 read_channels_from_bytes() (mdfreader.mdfinfo4.Info4 method), 42 read_channels_from_bytes_fallback() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.Com		<pre>read_cn_block() (in module mdfreader.mdfinfo3),</pre>
read_all_channels_sorted_record () (mdf-reader.mdfirfo4.lnfo4 reader.mdfirfeader.Record method), 31 read_cc_block () (in module mdfreader.mdfinfo3), 26 read_cc_block () (in module mdfreader.mdfinfo3), 26 read_cg_block () (in module mdfreader.mdfinfo3), 26 read_cg_block () (in module mdfreader.mdfinfo3), 26 read_cg_block () (in module mdfreader.mdfinfo3.lnfo3 method), 25 read_cg_block () (in module mdfreader.mdfinfo3.lnfo4 method), 40 read_cg_block () (indfreader.mdfinfo4.lnfo4 method), 40 read_cg_block () (indfreader.mdfinfo4.lnfo4 method), 40 read_channels_from_bytes () (indfreader.mdfinfo4.lnfo4 method), 40 read_channels_from_bytes_fallback () (indfreader.mdfinfo4.lnfo4 static method), 36 read_cm_cc () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc () (indfreader.		26
read_cc_() (mdfreader.mdfinfo4.CCBlock method), 35 read_cc_block () (in module mdfreader.mdfinfo3), 26 read_cc_block () (in module mdfreader.mdfinfo3), 26 read_cc_block () (in module mdfreader.mdfinfo3), 26 read_cg_block () (in module mdfreader.mdfinfo4.Info4 method), 40 read_cg_block () (indfreader.mdfinfo4.DBlock method), 32 read_clock () (indfreader.mdfinfo4.DBlock method		read_cn_block() (mdfreader.mdfinfo4.Info4
read_cc_block () (in module mdfreader.mdfinfo3), 26 read_cc_block () (in module mdfreader.mdfinfo4.Info4 method), 40 read_cb_block () (in module mdfreader.mdfinfo4.Info4 method), 40 read_ch_block () (in module mdfreader.mdfinfo4.Info4 method), 40 read_ch_block () (in module mdfreader.mdfinfo4.Info4 method), 40 read_ch_block () (in module mdfreader.mdfinfo4.Info4 method), 41 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 42 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 42 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 42 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 41 read_cd_block () (in module mdfreader.mdfinfo3), 26 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 41 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 42 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 41 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 42 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 42 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 42 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 41 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 42 read_cd_block () (in module mdfreader.mdfinfo3), 26 read_cd_block () (in module mdfreader.mdfinfo4.Info4 method), 41 read_cd_block () (in module mdfreader.mdfinfo3), 26 read_cd_block () (in module mdfreader.mdfinfo3), 26 read_cd_block () (in freader.mdfinfo4.Info4 method), 32 read_cm_at () (indfreader.mdfinfo4.Info4 method), 32 read_cm_at () (indfreader.mdfinfo4.Info4 method), 32 read_cm_at () (indfreader.mdfinfo4.Inf		method), 41
read_cc_block () (in module mdfreader.mdfinfo3), 26 read_cg_block () (in module mdfreader.mdfinfo4.Info4 method), 40 read_cg_block () (in module mdfreader.mdfinfo4.Info4 method), 40 read_ch_block () (in module mdfreader.mdfinfo4.Info4 method), 32 read_ch_block () (in mfreader.mdfinfo4.Info4 method), 32 read_ch_annels_from_bytes fallback () (in mfreader.mdfinfo4.Info4 method), 32 read_cm_at () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_dg () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_dg () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_dg () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_dg () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_co () (indfreader.mdfinfo4.CommentBlock method), 36 read_cm_co () (read_cn_blocks() (mdfreader.mdfinfo4.Info4
read_ce_block() (in module mdfreader.mdfinfo3), 26 read_cg_() (mdfreader.mdfinfo4.CGBlock method), 35 read_cg_block() (in module mdfreader.mdfinfo3), 26 read_cg_block() (in module mdfreader.mdfinfo3), 26 read_cg_block() (mdfreader.mdfinfo4.Info4 method), 25 read_cg_block() (mdfreader.mdfinfo4.Info4 method), 40 read_cg_block() (mdfreader.mdfinfo4.Info4 method), 40 read_cg_block() (mdfreader.mdfinfo4.Info4 method), 40 read_cd_block() (mdfreader.mdfinfo4.Info4 method), 40 read_ch_block() (mdfreader.mdfinfo4.Info4 method), 41 read_ch_block() (mdfreader.mdfinfo4.Info4 method), 42 read_hl_block() (in module mdfreader.mdfinfo4.Info4 method), 42 read_info() (inffreader.mdfinfo4.Info4 method), 42 read_info() (inffreader.mdfinfo4.Info4 method), 42 read_info() (inffreader.mdfinfo4.Info4 method), 32 read_cm_at() (inffreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (inffreader.mdfinfo4.CommentBlock metho		method), 41
read_cg) (mdfreader.mdfinfo4.CGBlock method), 35 read_cg block() (in module mdfreader.mdfinfo3), 26 read_cg block() (mdfreader.mdfinfo3.Info3 method), 25 read_cg block() (mdfreader.mdfinfo4.Info4 method), 40 read_cg block() (mdfreader.mdfinfo4.Info4 method), 40 read_channels_from_bytes() (mdfreader.mdfinfo4.Info4 method), 41 read_channels_from_bytes_fallback() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 37 read_cm_drader.mdfinfo4.CommentBlock method), 38 read_dl() (mdfreader.mdfinfo4.DBlock method), 39 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 30 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfi		<pre>read_data_list() (mdfreader.mdf4reader.Data</pre>
read_cg() (mdfreader.mdfinfo4.CGBlock method), 35	<pre>read_ce_block() (in module mdfreader.mdfinfo3),</pre>	
read_cg_block() (im module mdfreader.mdfinfo3), 26 read_cg_block() (mdfreader.mdfinfo3.Info3 method), 25 read_cg_block() (mdfreader.mdfinfo4.Info4 method), 40 read_cg_block() (mdfreader.mdfinfo4.Info4 method), 40 read_ch_block() (mdfreader.mdfinfo4.Info4 method), 41 read_channels_from_bytes() (mdfreader.mdfinfo4.Info4 reader.mdf4reader.Record method), 32 read_cm_at() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm		<pre>read_dg() (mdfreader.mdfinfo4.DGBlock method), 38</pre>
read_cg_block() (in module mdfreader.mdfinfo3), 26 read_cg_block() (mdfreader.mdfinfo3.Info3 method), 25 read_cg_block() (mdfreader.mdfinfo4.Info4 method), 40 read_cg_blocks() (mdfreader.mdfinfo4.Info4 method), 40 read_ch_block() (mdfreader.mdfinfo4.Info4 method), 41 read_ch_block() (mdfreader.mdfinfo4.Info4 method), 41 read_ch_annels_from_bytes() (mdfreader.mdfinfo4.Info4 method), 32 read_channels_from_bytes() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_d() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_d() (mdfreader.m	<pre>read_cg() (mdfreader.mdfinfo4.CGBlock method), 35</pre>	<pre>read_dg_block() (in module mdfreader.mdfinfo3),</pre>
read_cg_block() (mdfreader.mdfinfo3.Info3 method), 25 read_cg_block() (mdfreader.mdfinfo4.Info4 method), 40 read_cg_blocks() (mdfreader.mdfinfo4.Info4 method), 40 read_cg_blocks() (mdfreader.mdfinfo4.Info4 method), 40 read_ch_block() (mdfreader.mdfinfo4.Info4 method), 41 read_channels_from_bytes() (mdfreader.mdfinfo4.Info4 reader.mdf4reader.Record method), 32 read_channels_from_bytes_fallback() (mdfreader.mdfinfo4.Info4 method), 42 read_cm_at() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cd() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_dn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_dn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_dn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_dn() (mdfreader.mdfinfo4.SIBlock method), 33 read_cm_dn() (mdfreader.mdfinfo4.SIBlock method),		26
method), 25 read_cg_block()		read_dg_block() (mdfreader.mdfinfo4.Info4
read_cg_block() (mdfreader.mdfinfo4.Info4 method), 40 read_cg_blocks() (mdfreader.mdfinfo4.Info4 method), 40 read_ch_block() (mdfreader.mdfinfo4.Info4 method), 40 read_ch_block() (mdfreader.mdfinfo4.Info4 method), 41 read_ch_annels_from_bytes() (mdfreader.mdfinfo4.Info4 reader.mdf4reader.Record method), 32 read_ch_annels_from_bytes_fallback() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_at() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_dnfreader.mdfinfo4.CommentBlock method), 36 read_cm_dnfreader.mdfinfo4.CommentBlock method), 36 read_cm_dnfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_dnfreader.mdfinfo4.CommentBlock method), 36 read_cm_dnfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_dnfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_dnfreader.mdfinfo4.CommentBlock method), 36 read_cm_dnfreader.mdfinfo4.CommentBlock method), 36 read_cm_dnfreader.mdfinfo4.CommentBlock method), 32 read_cm_dnfreader.mdfinfo4.SIBlock method), 32 read_cm_dnfreader.mdfinfo4.CommentBlock method), 32 read_cm_dnfreader.mdfinfo4.SIBlock method), 32 read_cm_dnfreader.mdfinfo4.SIBlock method), 32 read_cm_dnfreader.mdfinfo4.CommentBlock method), 32 read_cm_dnfreader.mdfinfo4.SIBlock method), 32 read_cm_dnfreader.mdfinfo4.CommentBlock method), 32 read_cm_cn_() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_() (mdfreader.mdfinfo4.CommentBlock metho	<pre>read_cg_block()</pre>	
read_cg_blocks() (mdfreader.mdfinfo4.Info4 method), 40 read_ch_block() (mdfreader.mdfinfo4.Info4 method), 41 read_channels_from_bytes() (mdfreader.mdfinfo4.Info4 method), 42 read_channels_from_bytes_fallback() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock	method), 25	
read_cg_blocks() (mdfreader.mdfinfo4.Info4 method), 40 read_ch_block() (mdfreader.mdfinfo4.Info4 method), 41 read_channels_from_bytes() (mdfreader.mdfinfo4.Info4 reader.mdf4reader.Record method), 32 read_channels_from_bytes_fallback() (mdfreader.mdf4reader.Record method), 32 read_channels_from_bytes_fallback() (mdfreader.mdf4reader.Record method), 32 read_cm_at() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit() (mdfreader.mdf4reader.Record method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.md	<pre>read_cg_block()</pre>	
method), 40 read_ch_block() (mdfreader.mdfinfo4.Info4 method), 41 read_channels_from_bytes() (mdf-eader.mdfinfo4.Info4 reader.mdf4reader.Record method), 32 read_channels_from_bytes_fallback() (mdfreader.mdf4reader.Record method), 32 read_cm_at() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (m	method), 40	
read_ch_block()	read_cg_blocks() (mdfreader.mdfinfo4.Info4	
method), 41 read_channels_from_bytes() (mdf- reader.mdf4reader.Record method), 32 read_channels_from_bytes_fallback() (mdfreader.mdf1reader.Record method), 32 read_cm_at() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit() (mdfreader.mdf3reader.Record method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.md	method), 40	
read_channels_from_bytes() (mdf- reader.mdf4reader.Record method), 32 read_channels_from_bytes_fallback() (mdfreader.mdf4reader.Record method), 32 read_cm_at() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_n() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.SIBlock method), 43 read_sorted_record() (mdfreader.mdfinfo4.SIBlock method), 23 read_sorted_record() (mdfreader.mdfinfo4.SIBlock m	<pre>read_ch_block() (mdfreader.mdfinfo4.Info4</pre>	
reader.mdf4reader.Record method), 32 read_channels_from_bytes_fallback() (mdfreader.mdf4reader.Record method), 32 read_cm_at() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_df() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_df() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_df() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_df() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_df() (mdfreader.mdfinfo4.SIBlock method), 36 read_cm_df() (mdfreader.mdfinfo4.SIBlock method), 36 read_sorted_record() (mdfreader.mdfinfo4.SIBlock method),		
read_channels_from_bytes_fallback()		
read_cm_at() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit() (mdfreader.mdfafreader.Record method), 32 read_cm_cd_cm_cd_() (mdfreader.mdfafreader.Record method), 32 read_cm_cd_() (mdfreader.mdfafreader.Record method), 36 read_cm_cd_() (mdfreader.mdfafreader.Record method), 23 read_cm_cd_() (mdfreader.mdfafreader.Record method), 23 read_cm_cd_() (mdfreader.mdfafreader.Record method), 23 read_cm_cd_() (mdfreader.mdfafreader.Record method), 23 read_record_buf() (mdfreader.mdfafreader.Record method), 23 read_record_buf() (mdfreader.mdfafreader.Record method), 32 read_record_buf() (mdfreader.mdfafreader.Record method), 32 read_record_buf() (mdfreader.mdfafreader.Record method), 32 read_record_buf() (mdfreader.mdfafreader.Record method), 32 read_sorted_record() (mdfreader.mdfafreader.Record method), 33 read_sorted_record() (mdfreader.mdfafreader.Record method), 34 read_sorted_record() (mdfreader.mdfafreader.Record method), 36 read_sorted_record() (mdfreader.mdfafreader.Record method), 36 read_sorted_record() (mdfreader.mdfafreader.Record method), 36 read_sorted_record() (mdfreader.mdfafreader.Record method), 36		read_info() (mdfreader.mdfreader.MdfInfo method),
read_cm_at() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit() (mdfreader.mdfareader.Record method), 32 read_record() (mdfreader.mdfareader.Record method), 32 read_record() (mdfreader.mdfareader.Data method), 28 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfareader.Record method), 23 read_record_bits() (mdfreader.mdfareader.Record method), 23 read_sorted_record() (mdfreader.mdfareader.Recor		
method), 36 read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 37 read_record_buf() (mdfreader.mdfinfo4.SIBlock method), 38 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 37 read_record_buf() (mdfreader.mdfinfo4.SIBlock method), 37 read_sorted_record() (mdfreader.mdfinfo4.SIBlock method), 38 read_cm_dffreader.Record method), 38 read_record_buf() (mdfreader.mdfinfo4.SIBlock method), 39 read_sorted_record() (mdfreader.mdfinfo4.SIBlock method), 39 read_sorted_record() (mdfreader.mdfinfo4.SIBlock method), 39 read_sorted_record_sorted_record() (mdfreader.mdfinfo4.SIBlock method), 39 read_sorted_record_sorted_record() (mdfreader.mdfinfo4.SIBlock method), 39 read_sorted_record_s	· · · · · · · · · · · · · · · · · · ·	
read_cm_cc() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cc_unit() (mdf-eader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.SIBlock method), 32 read_cm_cn_unit() (mdfreader.mdfinfo4.SIBlock method), 33 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.SIBlock method), 33 read_cm_cn_dg() (mdfreader.mdfinfo4.CommentBlock method), 33 read_cm_cn_dg() (mdfreader.mdfinfo4.CommentBlock method), 33 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock method), 33		
method), 36 read_cm_cc_unit()		
read_cm_cc_unit() (mdf-reader.mdfinfo4.CommentBlock method), 28 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdf-reader.mdfinfo4.SIBlock method), 43 read_cm_cn_unit() (mdf-reader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdf-reader.mdfinfo4.CommentBlock method), 36 read_cm_dfinfo4.CommentBlock method), 36 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock read_sr_block() (mdfreader.mdfinfo4.Info4 static		
reader.mdfinfo4.CommentBlock method), 36 read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.SIBlock method), 43 reader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock reader.mdf3reader.Record method), 23 reader.mdf3reader.mdfinfo4.SIBlock method), 43 reader.mdf3reader.Record method), 23 read_cm_cn_unit() (mdfreader.mdf3reader.Record method), 23 read_cm_df4reader.Record method), 33 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock read_sr_block() (mdfreader.mdfinfo4.Info4 static		
read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 32 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 32 read_cm_ch() (mdfreader.mdfinfo4.SIBlock method), 43 read_sorted_record() (mdfreader.mdfinfo4.CommentBlock method), 23 read_cm_dfinfo4.CommentBlock method), 23 read_cm_dfinfo4.CommentBlock method), 36 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock method), 33 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock method), 33		
read_cm_cg() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 32 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.SIBlock method), 43 read_cm_cn_unit() (mdfreader.mdfinfo4.SIBlock method), 43 read_sorted_record() (mdfreader.mdfinfo4.CommentBlock method), 23 read_sorted_record() (mdfreader.mdfinfo4.CommentBlock method), 33 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock read_sr_block() (mdfreader.mdfinfo4.Info4 static		
method), 36 read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 32 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.SIBlock method), 43 read_cm_cn_unit() (mdfreader.mdf3reader.Record method), 23 reader.mdfinfo4.CommentBlock method), 36 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock read_sr_block() (mdfreader.mdfinfo4.Info4 static		
read_cm_ch() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.SIBlock method), 43 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock method), 23 reader.mdfinfo4.CommentBlock method), 36 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock read_sr_block() (mdfreader.Record method), 33 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock read_sr_block() (mdfreader.mdfinfo4.Info4 static		
method), 36 read_cm_cn() (mdfreader.mdfinfo4.CommentBlock method), 36 read_cm_cn_unit() (mdfreader.mdfinfo4.CommentBlock reader.mdfinfo4.CommentBlock method), 23 reader.mdfinfo4.CommentBlock method), 36 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock read_sr_block() (mdfreader.Record method), 33 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock read_sr_block() (mdfreader.mdfinfo4.Info4 static		<pre>read_record_buf() (mdfreader.mdf4reader.Record</pre>
method), 36 read_sorted_record() (mdf-reader.mdfinfo4.CommentBlock method), 23 reader.mdfinfo4.CommentBlock method), 36 reader.mdfinfo4.CommentBlock reader.mdf4reader.Record method), 33 read_cm_dg() (mdfreader.mdfinfo4.CommentBlock read_sr_block() (mdfreader.mdfinfo4.Info4 static		
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	<pre>read_cm_cn() (mdfreader.mdfinfo4.CommentBlock</pre>	
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$		·
$\begin{array}{cccccccccccccccccccccccccccccccccccc$		reader.mdf3reader.Record method), 23
read_cm_dg() (mdfreader.mdfinfo4.CommentBlock read_sr_block() (mdfreader.mdfinfo4.Info4 static		

read_tx() (mdfreader.mdfinfo4.CommentBlock method), 37	<pre>set_CANOpen() (mdfreader.channel.Channel4 method), 51</pre>
read_tx_block() (in module mdfreader.mdfinfo3), 26	<pre>set_channel_attachment() (mdf- reader.mdf.MdfSkeleton method), 16</pre>
read_unique_channel() (mdf-	set_channel_conversion() (mdf-
reader.mdf4reader.Record method), 33	reader.mdf.MdfSkeleton method), 16
read_xml() (mdfreader.mdfinfo4.CommentBlock	set_channel_data() (mdfreader.mdf.MdfSkeleton
method), 37	method), 16
readRecordBits() (mdfreader.mdf3reader.Record	set_channel_desc() (mdfreader.mdf.MdfSkeleton
method), 23	method), 16
readRecordBuf() (mdfreader.mdf3reader.Record	<pre>set_channel_master() (mdf-</pre>
method), 23	reader.mdf.MdfSkeleton method), 16
readSortedRecord() (mdf-	<pre>set_channel_master_type() (mdf-</pre>
reader.mdf3reader.Record method), 23	reader.mdf.MdfSkeleton method), 17
recAttributeName (mdfreader.channel.Channel3 at-	<pre>set_channel_unit() (mdfreader.mdf.MdfSkeleton</pre>
tribute), 46	method), 17
Record (class in mdfreader.mdf3reader), 22	<pre>set_invalid_bit() (mdfreader.mdf.MdfSkeleton</pre>
Record (class in mdfreader.mdf4reader), 30	method), 17
record_attribute_name() (mdf-	set_invalid_bytes() (mdf-
reader.channel.Channel4 method), 51	reader.channel.Channel4 method), 51
record_id_size() (mdfreader.channel.Channel4	set_invalid_channel() (mdf-
method), 51	reader.mdf.MdfSkeleton method), 17
RecordFormat (mdfreader.channel.Channel3 at-	SIBlock (class in mdfreader.mdfinfo4), 43
tribute), 46	signal_data_type() (mdfreader.channel.Channel4
recordID (mdfreader.mdf3reader.Record attribute), 22	method), 51
recordID (mdfreader.mdf4reader.Record attribute), 33	signalDataType (mdfreader.channel.Channel3 at-
recordIDCFormat (mdfreader.mdf4reader.Record at-	tribute), 45
tribute), 33	SRBlock (class in mdfreader.mdfinfo4), 43
recordIDnumber (mdfreader.mdf3reader.Record attribute), 22	Т
recordIDsize (mdfreader.mdf4reader.Record attribute), 33	type (mdfreader.channel.Channel4 attribute), 52 type (mdfreader.mdf4reader.Data attribute), 28
recordLength (mdfreader.mdf3reader.Record at-	
tribute), 22	U
recordLength (mdfreader.mdf4reader.Record attribute), 33	unique_channel_in_DG (mdf-reader.mdf4reader.Record attribute), 33
recordToChannelMatching (mdf-	unique_id() (mdfreader.mdfinfo4.Info4 method), 42
reader.mdf3reader.Record attribute), 22	unit (mdfreader.channel.Channel3 attribute), 45
recordToChannelMatching (mdf-	unit () (mdfreader.channel.Channel4 method), 52
reader.mdf4reader.Record attribute), 33	
remove_channel() (mdfreader.mdf.MdfSkeleton	V
method), 15	VLSD (mdfreader.mdf4reader.Record attribute), 30
remove_channel_conversion() (mdf-	VLSD_CG (mdfreader.mdf4reader.Record attribute), 30
reader.mdf.MdfSkeleton method), 16	VLSD_CG_Flag (mdfreader.channel.Channel4 at-
rename_channel() (mdfreader.mdf.MdfSkeleton method), 16	tribute), 47
resample() (mdfreader.mdfreader.Mdf method), 4, 9	W
resample_group() (mdfreader.Mdf	
method), 10	write() (mdfreader.mdfinfo4.CABlock method), 35 write() (mdfreader.mdfinfo4.CGBlock method), 35
return_pandas_dataframe() (mdf-	write() (majreader.majnij04.CGBlock method), 36 write() (mdfreader.mdfinfo4.CNBlock method), 36
reader.mdfreader.Mdf method), 10	write() (mafreader.mafinfo4.CommentBlock method), 30 write() (mdfreader.mdfinfo4.CommentBlock method),
S	38
	write() (mdfreader.mdfinfo4.DGBlock method), 38
set () (mdfreader.channel.Channel4 method), 51	Constitution of the contraction of the contr

```
write() (mdfreader.mdfinfo4.DIBlock method), 38
write() (mdfreader.mdfinfo4.DLBlock method), 38
write() (mdfreader.mdfinfo4.DTBlock method), 38
write() (mdfreader.mdfinfo4.DVBlock method), 38
write() (mdfreader.mdfinfo4.DZBlock method), 39
write() (mdfreader.mdfinfo4.FHBlock method), 39
write() (mdfreader.mdfinfo4.HDBlock method), 39
write() (mdfreader.mdfinfo4.HLBlock method), 39
write() (mdfreader.mdfinfo4.IDBlock method), 39
write() (mdfreader.mdfinfo4.LDBlock method), 42
write() (mdfreader.mdfreader.Mdf method), 4, 10
write3() (mdfreader.mdf3reader.Mdf3 method), 21
write4() (mdfreader.mdf4reader.Mdf4 method), 29, 30
write_DIV() (mdfreader.mdfinfo4.LDBlock method),
        42
write_DZ() (mdfreader.mdfinfo4.LDBlock method),
        43
Z
zipfile (mdfreader.mdf.MdfSkeleton attribute), 17
```

```
zipfile (mdfreader.mdfinfo4.Info4 attribute), 42
zipfile (mdfreader.mdfreader.MdfInfo attribute), 11
```