

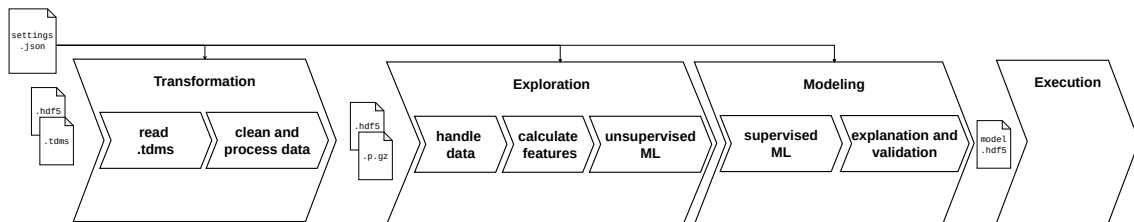


Transformation ML Framework

on basis of the XBox2 Data Set

Lorenz Fischl

Introduction



Introduction

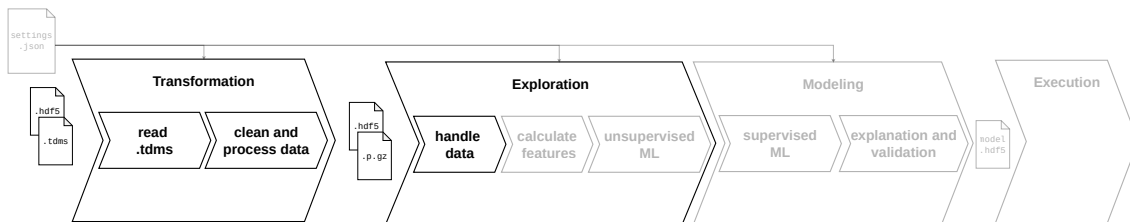


Table of Contents

XBox2 File Structure

Event-&TrendData

Choosing a Data Format

Requirements

reading tdms files

pandas + compressed pickle

.hdf5

Conclusion

Table of Contents

XBox2 File Structure

Event-&TrendData

Choosing a Data Format

Requirements

reading tdms files

pandas + compressed pickle

.hdf5

Conclusion

Table of Contents

XBox2 File Structure

Event-&TrendData

Choosing a Data Format

Requirements

reading tdms files

pandas + compressed pickle

.hdf5

Conclusion

XBox2 Data

XBox2 Data

Every 20 ms a pulse is sent into the RF cavity for particle acceleration.

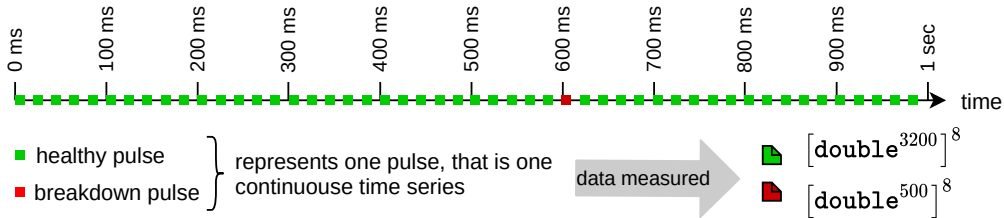
XBox2 Data

Every $20ms$ a pulse is sent into the RF cavity for particle acceleration. Sometimes an arc forms. Those events are called breakdown.

XBox2 Data

Every 20ms a pulse is sent into the RF cavity for particle acceleration. Sometimes an arc forms. Those events are called breakdown.

vspace1cm



XBox2 EventData

XBox2 EventData

A log group of one pulse is stored every minute.

XBox2 EventData

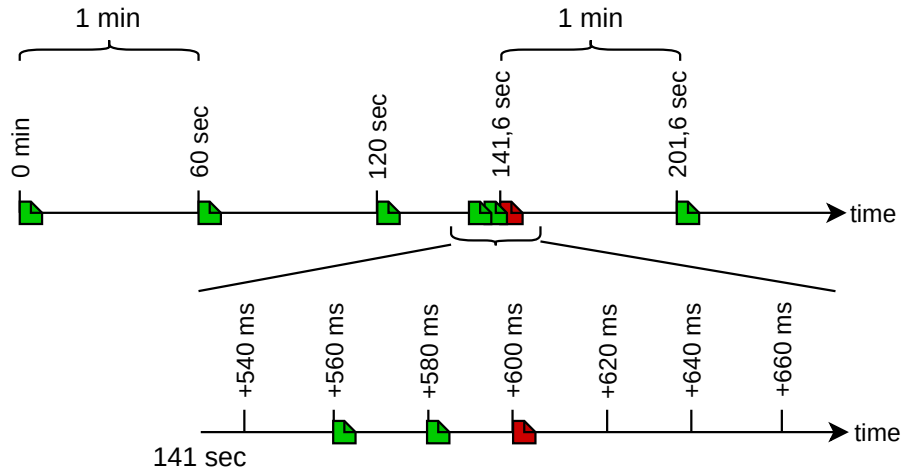
A log group of one pulse is stored every minute.

When a breakdown happens the corresponding log group + the two prior log groups are stored.

XBox2 EventData

A log group of one pulse is stored every minute.

When a breakdown happens the corresponding log group + the two prior log groups are stored.



XBox2 TrendData

XBox2 TrendData

35 values about the environmental conditions (that don't change rapidly) are stored roughly every 1,5 sec.

XBox2 TrendData

35 values about the environmental conditions (that don't change rapidly) are stored roughly every 1,5 sec.

All TrendData of one day is stored in 1-2 groups.

XBox2 TrendData

35 values about the environmental conditions (that don't change rapidly) are stored roughly every 1,5 sec.

All TrendData of one day is stored in 1-2 groups.

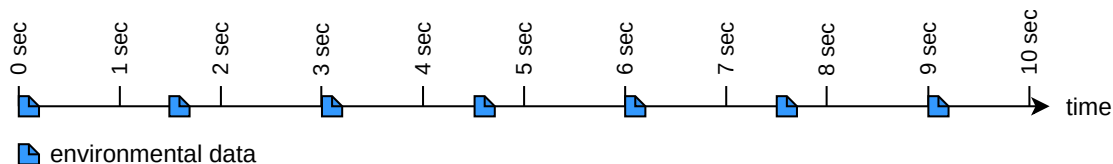


Table of Contents

XBox2 File Structure

Event-&TrendData

Choosing a Data Format

- Requirements

- reading tdms files

- pandas + compressed pickle

- .hdf5

- Conclusion

Table of Contents

XBox2 File Structure

Event-&TrendData

Choosing a Data Format

Requirements

reading tdms files

pandas + compressed pickle

.hdf5

Conclusion

Requirements

We need a data format that

Requirements

We need a data format that

- is compatible between systems (SWAN, htcondor, locally)

Requirements

We need a data format that

- is compatible between systems (SWAN, htcondor, locally)
- can read data column wise fast

Requirements

We need a data format that

- is compatible between systems (SWAN, htcondor, locally)
- can read data column wise fast
- can be Transformed and handled (sorted/filtered/processed) relatively easy

Requirements

We need a data format that

- is compatible between systems (SWAN, htcondor, locally)
- can read data column wise fast
- can be Transformed and handled (sorted/filtered/processed) relatively easy
- can be transformed into pandas DataFrame

Requirements

We need a data format that

- is compatible between systems (SWAN, htcondor, locally)
- can read data column wise fast
- can be Transformed and handled (sorted/filtered/processed) relatively easy
- can be transformed into pandas DataFrame
- outputs numpy arrays for machine learning

Table of Contents

XBox2 File Structure

Event-&TrendData

Choosing a Data Format

Requirements

reading tdms files

pandas + compressed pickle

.hdf5

Conclusion

nptdms

nptdms

- package nptdms can read and handle .tdms files

nptdms

- package nptdms can read and handle .tdms files
- very slow (ex.: read of a 100MB file can take >30sec)

nptdms

- package nptdms can read and handle .tdms files
- very slow (ex.: read of a 100MB file can take >30sec)
- very space inefficient (ex. TrendData: 20,5 GB in .tdms → 2.8 GB of data)

Table of Contents

XBox2 File Structure

Event-&TrendData

Choosing a Data Format

Requirements

reading tdms files

pandas + compressed pickle

.hdf5

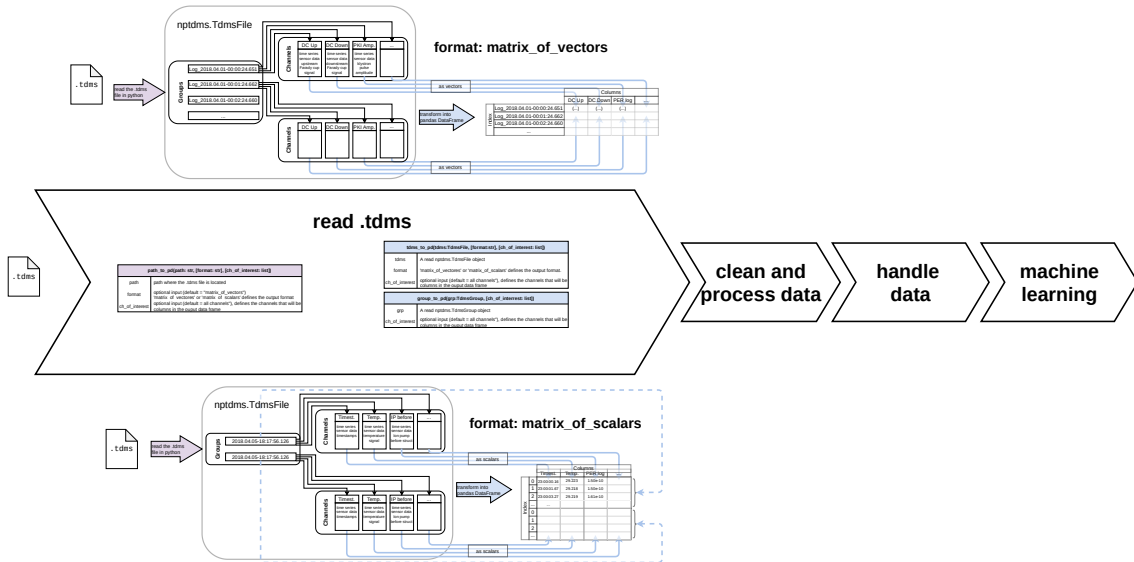
Conclusion

Transformation

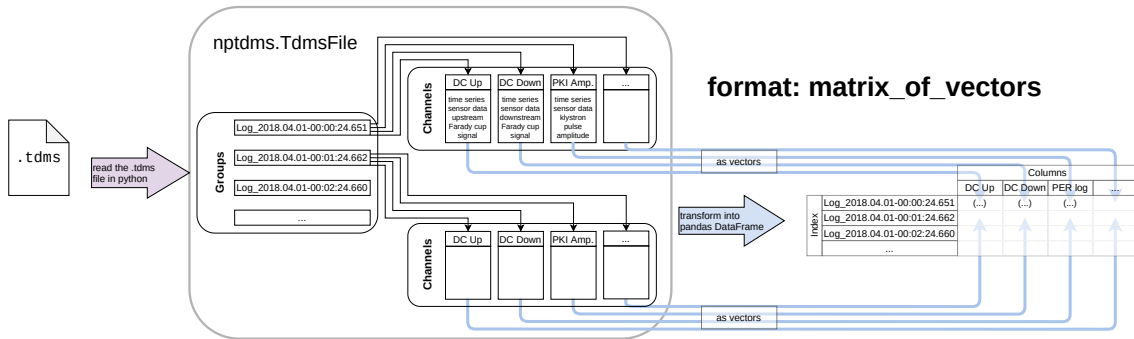
Transformation

`.tdms` \longrightarrow `pd df/ dictionary` \longrightarrow `pickle` \longrightarrow `.gzip`

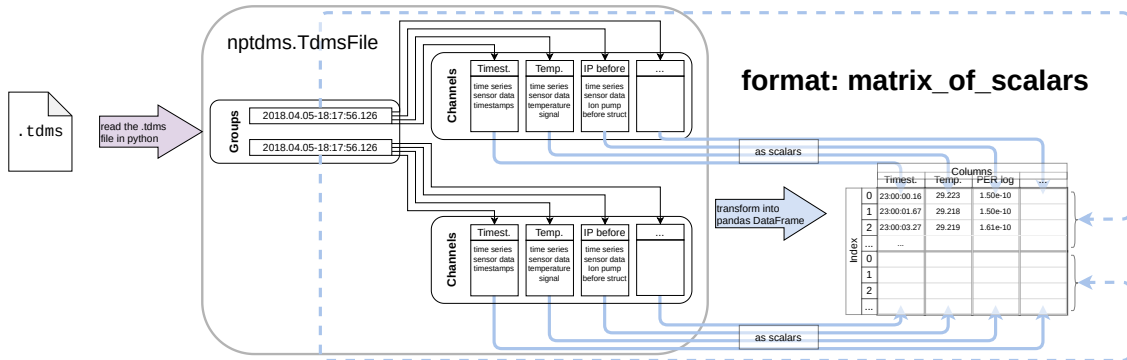
Transformation: read.tdms



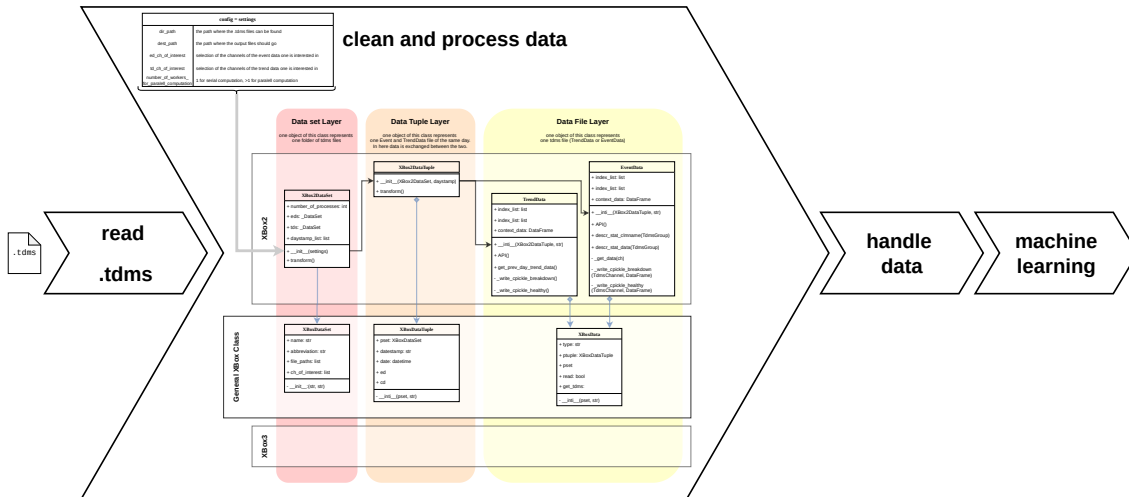
Transformation: read.tdms

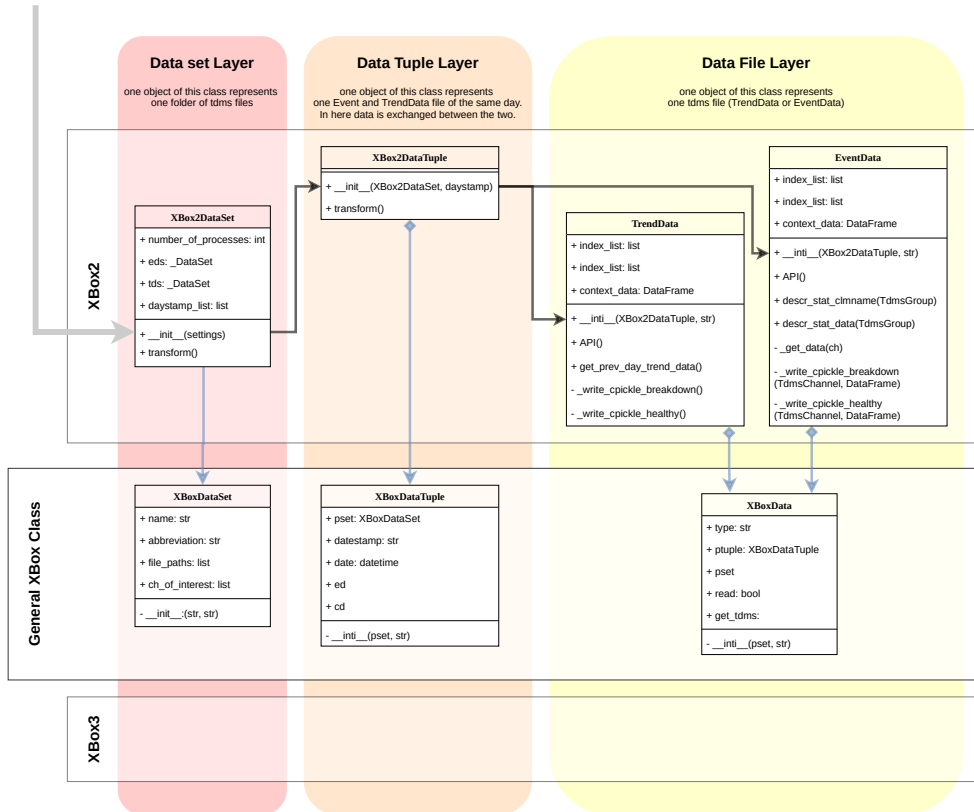


Transformation: read.tdms



Transformation: Clean and Process data with classes





Summary

Summary

- pandas DataFrame are easy to use in notebooks

Summary

- pandas DataFrame are easy to use in notebooks
- pickle speeds up reading time

Summary

- pandas DataFrame are easy to use in notebooks
- pickle speeds up reading time
- with compression takes up less space

Summary

- pandas DataFrame are easy to use in notebooks
 - pickle speeds up reading time
 - with compression takes up less space
-
- version issues with pickle protocol

Summary

- pandas DataFrame are easy to use in notebooks
 - pickle speeds up reading time
 - with compression takes up less space
-
- version issues with pickle protocol
 - part of the process should not be in the Datahandler instead of the Transformation

Summary

- pandas DataFrame are easy to use in notebooks
- pickle speeds up reading time
- with compression takes up less space

- version issues with pickle protocol
- part of the process should not be in the Datahandler instead of the Transformation
- EventData and TrenadData are stored differently

Summary

- pandas DataFrame are easy to use in notebooks
 - pickle speeds up reading time
 - with compression takes up less space
-
- version issues with pickle protocol
 - part of the process should not be in the Datahandler instead of the Transformation
 - EventData and TrenadData are stored differently
 - channel properties are lost

Summary

- pandas DataFrame are easy to use in notebooks
- pickle speeds up reading time
- with compression takes up less space

- version issues with pickle protocol
- part of the process should not be in the Datahandler instead of the Transformation
- EventData and TrenadData are stored differently
- channel properties are lost
- data was changed in place in notebooks in retrospect

Table of Contents

XBox2 File Structure

Event-&TrendData

Choosing a Data Format

Requirements

reading tdms files

pandas + compressed pickle

.hdf5

Conclusion

.hdf5

.hdf5

- Hierarchical Data Format with a Unix file system like tree structure

.hdf5

- Hierarchical Data Format with a Unix file system like tree structure
- build for data management and storage

.hdf5

- Hierarchical Data Format with a Unix file system like tree structure
- build for data management and storage
- keys work like pointers to the memory, channel wise reading possible

.hdf5

- Hierarchical Data Format with a Unix file system like tree structure
- build for data management and storage
- keys work like pointers to the memory, channel wise reading possible
- in nptdms the function `to_hdf()` exists

.hdf5

- Hierarchical Data Format with a Unix file system like tree structure
- build for data management and storage
- keys work like pointers to the memory, channel wise reading possible
- in `nptdms` the function `to_hdf()` exists
- no data is lost in the Transformation, also properties are stored

Table of Contents

XBox2 File Structure

Event-&TrendData

Choosing a Data Format

Requirements

reading tdms files

pandas + compressed pickle

.hdf5

Conclusion

Conclusion

Conclusion

- I implemented a generic class for converting .tdms files into `pd.df+cpickle`

Conclusion

- I implemented a generic class for converting `.tdms` files into `pd.df+cpickle`
- is there a better data format, maybe `.hdf5`?

Conclusion

- I implemented a generic class for converting `.tdms` files into `pd.df+cpickle`
- is there a better data format, maybe `.hdf5`?

Conclusion

- I implemented a generic class for converting .tdms files into pd.df+cpickle
- is there a better data format, maybe .hdf5?

| | nptdms | pd.df+cpickle | | .hdf5 | |
|-----------------------|---------|---------------|-------|---------|--------|
| | | w/o zip | w zip | w/o zip | w/ zip |
| space (GByte) | 20.5GB | 2.8GB | 1GB | 2.8GB | 1GB |
| read (TD 1 channel) | ~ 60min | 4sec | 12sec | 0.5 sec | |
| read (TD 3 channels) | ~ 60min | 4sec | 12sec | 1 sec | |
| read (TD 15 channels) | ~ 60min | 4sec | 12sec | 4 sec | |
| feature calc. (ED) | > 15min | | 7sec | 8 sec | |



home.cern