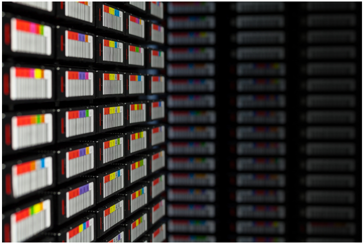
ExtractIng - Automated metadata extraction for computational engineering applications and high-performance computing

Björn Schembera (schembera@hlrs.de)

RDA Research Data Management in Engineering IG

Seminar Series *Exploring Annotation and Metadata Initiatives for Engineering Data*

September 15th, 2020



Outline

Introduction

ExtractIng - Automated metadata extraction
- Introduction to ExtractIng
- Extractability of metadata information
- Architecture and Implementation
- Configuration
- Running

Evaluation

Conclusion and Future Work

Introduction

Introduction

- (Explicit) Metadata is a main contributor to FAIR data management
- However metadata annotation is a burden
- Low incentives due to low scientific recognition in computational engineering
- Manual metadata tagging is bothersome

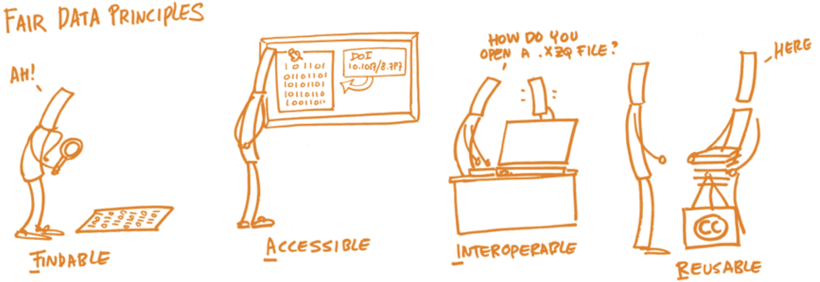


Figure: FAIR data principles in a nutshell (http://www.openaire.eu)

ExtractIng - Automated metadata extraction

Introduction to ExtractIng

Use Case

- High-Performance Computing
- Engineering Applications, in particular
  - Thermodynamics
  - Aerodynamics

Role of the metadata model EngMeta

- Serves as a convention
- ExtractIng can also be seen as a use case of EngMeta

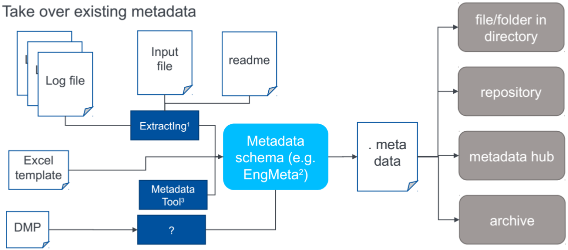


Figure: ExtractIng in the existing RDM ecosystem

Introduction to ExtractIng

Some metadata is already available

- Explicit and implicit file attributes
- Metadata in (output) files of the simulation codes, schedulers, ...
  - In standardizied file formats such as HDF5 or NetCDF
  - In non-standardizied file formats
  - In job or log files of simulation codes (z.B. nodes, version)
- Lots of semi-structured metadata available

```
hpcbsche@atlas:~/Projekte/DIPL-ING/metadaten/harvester/sample_data/protein/020415_1800_meo_1800_vac_5400_tol$ head 05_log.log
Log file opened on Fri Jan 12 06:08:19 2018
Host: node154  pid: 106311  nodeid: 0  nnodes:  16
Gromacs version:    VERSION 4.6.7
Precision:          double
Memory model:       64 bit
MPI library:        MPI
OpenMP support:     enabled
GPU support:        disabled
invsqrt routine:    gmx_software_invsqrt(x)
CPU acceleration:   AVX_256
```

Figure: Head of a GROMACS Log file.

Metadata model EngMeta – Four metadata categories

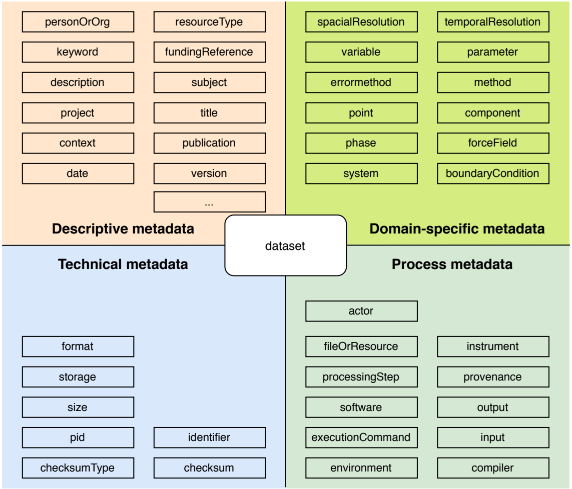


Figure: EngMeta, with categories. `https://www.izus.uni-stuttgart.de/fokus/engmeta/`

Extractability of the different metadata categories

Type of metadata	Extractability
Technical metadata	high, as available via file attributes
Process metadata	medium, as available in log-, job- or system files
Domain-specific metadata	medium, as available in log- or output files
Descriptive metadata	poor, as it's a description from a higher level

Table: Extractability of the different metadata categories. It is strongly dependent on the field of science.

Approach of ExtractIng

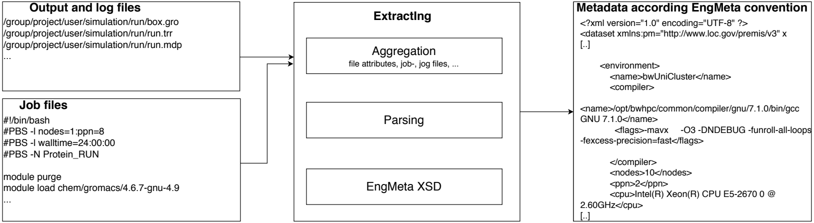


Figure: Architecture of the metadata extraction

ExtractIng: Implementation

- Based on Java in two variants
  - Native: Java Scanner API
  - Parallel: Spark Data Analytics Framework
- Run of ExtractIng refers to a directory
- A subdirectory *.metadata* then stores the metadata information in XML

```
[hpcbsche@nid00030 .metadata]$ pwd
/mnt/lustre/hpcbsche/itt_data/binary/educt_hexane/300_020_080/run/.metadata
[hpcbsche@nid00030 .metadata]$ ls -alrt
total 20
drwxr-xr-x 2 hpcbsche s29931 4096 Jan 29 15:39 .
-rw-r--r-- 1 hpcbsche s29931 1520 Feb  6 11:46 metadata.txt
-rw-r--r-- 1 hpcbsche s29931 2717 Feb  6 11:46 engMeta.xml
-rw-r--r-- 1 hpcbsche s29931  630 Feb  6 11:46 atom.xml
drwxr-xr-x 3 hpcbsche s29931 4096 Feb 13 11:49 ..
[hpcbsche@nid00030 .metadata]$ tail engMeta.xml
                    <flags>-mavx     -O3 -DNDEBUG -funroll-all-loops -fexcess-precision=fast</flags>
                </compiler>
                <nodes>1</nodes>
                <ppn>8</ppn>
                <cpu>Intel(R) Xeon(R) CPU E5-2670 0 @ 2.60GHz</cpu>
            </environment>
        </step>
    </provenance>
    <size>58</size>
</dataset>
[hpcbsche@nid00030 .metadata]$
```

Figure: Directory with parsed metadata and a part of the EngMeta XML file.

ExtractIng: Configuration

- Everything regarding the extraction is configured externally
- External configuration file based on the EngMeta convention
- Syntax:

```
<EngMetaKey>,<filename>,<searchKey>,<delimiter>,<semantics>
```

```
hpcbsche@nid00030:~/dev/harvester
controlledVariable.name,usermd,var1.name,=,1
controlledVariable.value,mdp,ref_t,=,1
controlledVariable.name,usermd,var2.name,=,2
controlledVariable.value,mdp,tcoupl,=,2
controlledVariable.name,usermd,var3.name,=,3
controlledVariable.value,mdp,ref_p,=,3
controlledVariable.name,usermd,var4.name,=,4
controlledVariable.value,mdp,pcoupl,=,4
provenance.processingStep.executionCommand,mdp,gmx_mpi grompp,na,1
provenance.processingStep.executionCommand,log,gmx_mpi mdrun,na,2
provenance.processingStep.environment.compiler.name,log,g++ ,na,1
provenance.processingStep.environment.compiler.flags,log,C++ compiler flags,:,1
provenance.processingStep.environment.nodes,job,nodes,=,
provenance.processingStep.environment.ppn,job,ppn,=,
provenance.processingStep.environment.cpu,log,Build CPU brand,:
system.temporalResolution.numberOfTimesteps,mdp,nsteps,=,
                                                        214,1          55%
```

Figure: Sample part of a configuration file for GROMACS.

ExtractIng: Running

ExtraxtIng uses a wrapper script to shield some preperatory steps.

Listing 1: Syntax of ExtractIng

```
./fdm.sh -c <configFile> -p <directory>|"<dir1> <dir2> ..." \\
         -m [scanner|spark] [-e <executorCores>]
```

Listing 2: Sample call of the metadata extraction

```
./fdm.sh -c fdm.conf -p /mnt/lustre/data/educt_hexane/300_020_080/run/ \\
         -m scanner
```

Evaluation

Evaluation: ExtractIng – Adaptability

- Adaptability to other simulation codes: configuration file
  - Tested:
    GROMACS
    NS3D (EAS3)
    CCSM 3.0 (NetCDF in CF-Convention)
  - The more standardized, the easier to configure
  - Strongly depended on the output of the simulation code
- Adaptability to metadata models
  - Implementation of the model as Java class
  - Can partly be automated with JAXB

Extractable metadata from GROMACS

Metadata key (according toEngMeta)	Appearance	search key/line
processingStep.date	*.mdp	At date
controlledVariable.name	*.usermd	var1.name
controlledVariable.value	*.mdp	ref_t
controlledVariable.name	*.usermd	var2.name
controlledVariable.value	*.mdp	tcoupl
controlledVariable.name	*.usermd	var3.name
controlledVariable.value	*.mdp	ref_p
controlledVariable.name	*.usermd	var4.name
controlledVariable.value	*.mdp	pcoupl
processingStep.tool.name	*.log	GROMACS
processingStep.tool.softwareVersion	*.log	GROMACS version
processingStep.tool.operatingSystem	*.log	Build OS/arch
processingStep.executionCommand	*.log	gmx_mpi mdrun
processingStep.executionCommand	*.log	gmx_mpi grompp
processingStep.environment.compiler.name	*.log	C++ compiler
processingStep.environment.compiler.flags	*.log	C++ compiler flags
processingStep.environment.compiler.name	*.log	C compiler
processingStep.environment.compiler.flags	*.log	C compiler flags
processingStep.environment.nodes	*.job	nodes
processingStep.environment.ppn	*.job	ppn
processingStep.environment.cpu	*.log	Build CPU brand
system.grid.countX	*.gro	last line
system.grid.countY	*.gro	last line
system.grid.countZ	*.gro	last line
system.temporalResolution.numberOfTimesteps	*.mdp	nsteps
system.temporalResolution.interval	*.mdp	dt

Evaluation: ExtractIng – Adaptability

	native Scanner	parallel Spark
Worskstation		
Ubuntu 18.04	✓	✓
Windows 10	✓	—
bwUniCluster		
RHEL 7.5	✓	✓
Cray XC40		
CLE 6.0.UP05	✓	—
Cray URIKA		
Urika-GX-2.2UP00	✓	✓

Table: ExtractIng adaptability to compute environments

Evaluation: ExtractIng – Performance

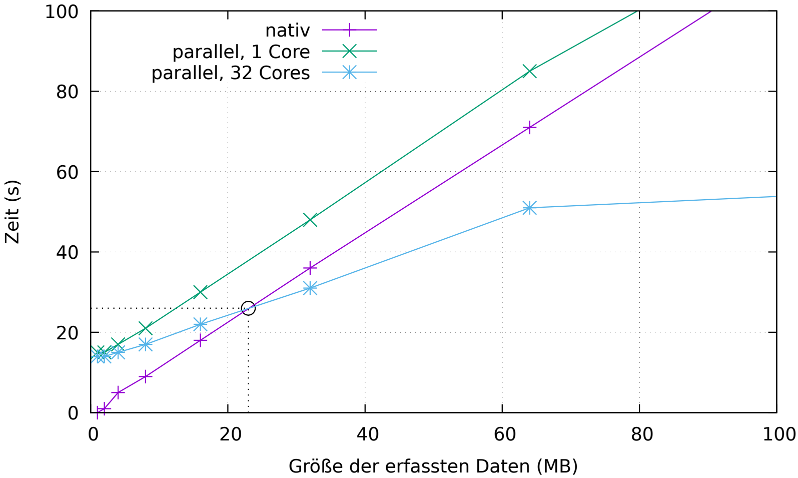


Figure: Performance comparison of native (Scanner) and parallel (Spark) implementation. Measured on Cray URIKA.

Evaluation: Integration – Scientific Workflow

- Extraction can be integrated to the job script, see script:

Listing 3: Trigger ExtractIng inside the job script.

```
#!/bin/bash
#PBS -N Aero_Simulation
#PBS -l nodes=1:ppn=24
#PBS -l walltime=00:20:00
#PBS -M schembera@hlrs.de
module load java

# Change to the direcotry that the job was submitted from
cd $PBS_O_WORKDIR

# Launch the parallel job and the metadata collection right after
aprun -n 24 -N 24 ~/promotion/aeroCode > my_output_file 2>&1
~/harvester/fdm.sh ~/harvester/fdm_iag_eval.conf . scanner
```

- Then, data + metadata can be pushed to a repository, such as DaRUS.

Conclusion and Future Work

Conclusion and Future Work

Conclusion and Findings

- Metadata annotation as a burden, however as a key to FAIR data
- ExtractIng tries improve the situation by automated extraction
- It is designed not to alter the specific scientific workflow
- ExtractIng is available on `https://github.com/bjschembera/ExtractIng`
- This is a proof-of-concept implementation, lots of improvements to be done...
- The project provided lots of findings regarding usage and extractability of metadata

Limitations and Future Work

- Limited to extraction of $< \mathit{key} >< \mathit{derlimiter} >< \mathit{value} >$ patterns
- Extraction of unstructured data is not possible
- Hierarchical information is hard to extract
- Extraction function is currently limited to lines

References

Bruce, Thomas R., and Diane I. Hillmann. "The continuum of metadata quality: defining, expressing, exploiting." ALA editions, 2004.

DFG. Denkschrift. Sicherung guter wissenschaftlicher Praxis, 2013.

Edwards, Paul N., et al. "Science friction: Data, metadata, and collaboration." Social Studies of Science 41.5 (2011): 667-690.

https://www.ub.uni-stuttgart.de/forschen-publizieren/forschungsdatenmanagement/projekte/dipl_ing/materials/metadata/index.html

Hasse, Hans, and Johannes Lenhard. "Boon and bane: On the role of adjustable parameters in simulation models." Mathematics as a Tool. Springer, Cham, 2017. 93-115.

Hey, Tony, and Anne Trefethen. "The data deluge: An escience perspective." Grid computing: Making the global infrastructure a reality (2003): 809-824.

Heidorn, Bryan P. Shedding light on the dark data in the long tail of science. Library Trends, 57(2):280–299, 2008.

- Heene, Markus, et al. "Automatic Metadata Generation for Dark Data to Support Information Systems." AGU Fall Meeting Abstracts. 2016.
- Jones, Stephanie N., et al. (2011): Easing the burdens of HPC file management. Proceedings of the sixth workshop on Parallel Data Storage. ACM, 2011.
- Mattmann, Chris A.: Computing: A vision for data science Nature, Vol. 493, No. 7433. (23 January 2013), pp. 473-475, doi:10.1038/493473a
- Michener, William K., et al. "Nongeospatial metadata for the ecological sciences." Ecological Applications 7.1 (1997): 330-342.
- NISO: "Framework of Guidance for Building Good Digital Collections." (2007): 61-62.
- Iglezakis, Dorothea, and Björn Schembera. "Anforderungen der Ingenieurwissenschaften an das Forschungsdatenmanagement der Universitt Stuttgart-Ergebnisse der Bedarfsanalyse des Projektes DIPL-ING." o-bib. Das offene Bibliotheksjournal/Herausgeber VDB 5.3 (2018): 46-60.
- Reilly, Susan u. a. (2011). Report on integration of data and publications. Zugrgriffen: 3.5.2019. URL : `http://www.alliancepermanentaccess.org/wp-content/uploads/downloads/2011/11/ODE-ReportOnIntegrationOfDataAndPublications-1%5f1.pdf`.
- Wilkinson, Mark D., et al. The FAIR Guiding Principles for scientific data management and stewardship. Scientific data 3 (2016).