Preserving Scientific Processes from Design to Publications

Rudolf Mayer, Andreas Rauber
rmayer@sba-research.at, rauber@ifs.tuwien.ac.at

Secure Business Austria
http://www.sba-research.org

Vienna University of Technology, Austr
http://www.ifs.tuwien.ac.at/dp

Martin Alexander Neumann
Karlsruhe Institute of Technology
Karlsruhe, Germany

John Thomson
Caixa Magica Software
Lisbon, Portugal

Goncalo Antunes
INESC-ID
Lisbon, Portugal

Overview

- Preservation of research
- Process preservation
- Process Context model to describe processes
- Use Case example: music classification experiment
- Conclusions

Preservation of Research

- Long-term access to research publications desired
  - Storage, understandability of data format sufficiently addressed today

- Today’s research often based on experiments
  - Huge amounts of data (Big Data, Fourth Paradigm)
  - Often complex steps of preprocessing, transformation and analysis
  - Verification, Comparability, Repeatability, …
- Publication is often only the last step in the research process – preservation needs to go beyond the document

Preservation of Research

- Comparability: often tackled through benchmark environments, using a standard data set and evaluation measure (campaigns such as TREC, CLEF, MIREX, …)
- Repeatability: Documentation of experiment environment, publication of source code developed, …
- Complexity of processes: can be tackled by using (scientific) workflow engines such (Taverna, Kepler, …)
  - Allow exact definition of steps executed, configuration employed, data input and output, …
  - Facilitates repeatability

Preservation of Research

- Some aspects beyond the control of workflow engines
  - Computing environment outside the engine
    - Hardware, operating system, software packages installed can all have an effect on the results
    - Changes might occur without initiative and notification to the researcher (e.g. software updates)
  - Use of external services becoming more popular
    - Web services providing specialised computation, e.g. frequently used in the Bio-informatics domain
    - May become unavailable, change their interface, behaviour, …

Process Preservation

- Preserving repeatability of processes emerging topic in Digital Preservation Research (e.g. projects TIMBUS, Workflows4ever, …)
- Goal: allow re-execution of the complete process at a later stage, when e.g. changes in the technology render the original process obsolete

Process Preservation Challenges

- What needs to be captured at ingest?
  - Need to go beyond single files (and their metadata)
  - Up to complete computer systems, including additional documents needed to understand & operate process

- How do these digital objects need to be described?
  - Need to characterise several aspects of the process
  - From a top-level: organisational parameters
  - Down to technical description of systems, including hardware, operating systems, software, third-party libraries and services.

Process Preservation Challenges

- Which preservation actions are applicable?
  - Combination of several different preservation actions, such as migration of specifications and documents, code migration/cross-compilation, or emulation of hardware or software utilised in the process.

- How can a preserved process be verified and evaluated?
  - Need to ensure that the execution of the (modified) process at a later stage is equivalent to the original process

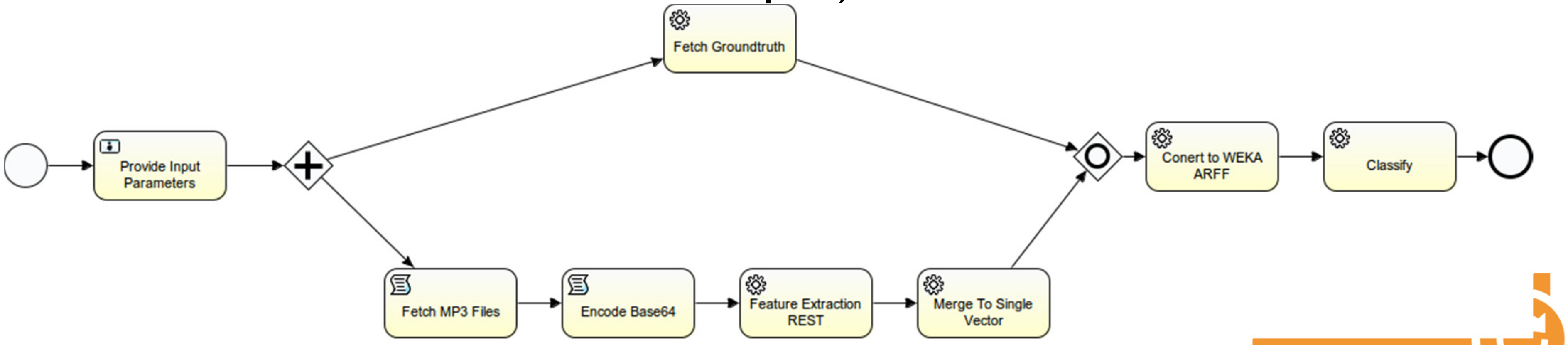
Example: Music Classification

- Scientific experiment from the machine learning/information retrieval domain

- Classification of music into predefined set of genres
- Learns a machine-learning model from given training data (i.e. data with manually assigned class/genre)
- Predicts genre for previously unseen data
- Useful e.g. for online music store, recommendation services, etc.

Example: Music Classification

- Input:
  - Music (e.g. MP3 format)
  - Ground truth/gold standard
- Output: Classification of music, e.g. into genres
- Intermediate steps
  - Extract numeric description (features) from music
  - Format conversions (feature extractor output to classification software input)



Music Classification: Aspects

- Software environment including configuration (machine learning software, operating system)
- External services: feature extraction, ground truth, …
- Hardware (e.g. computation on GPUs)
- Licenses & access keys
- Experiment parameters
- Input data, intermediate data created in the process
- …

Process Context Model

- Allows to systematically capture aspects of processes which are essential for preservation and re-execution

- Model in the form of an Ontology
  - Elements organised in a hierarchy
  - Models relations between elements (e.g. dependencies)

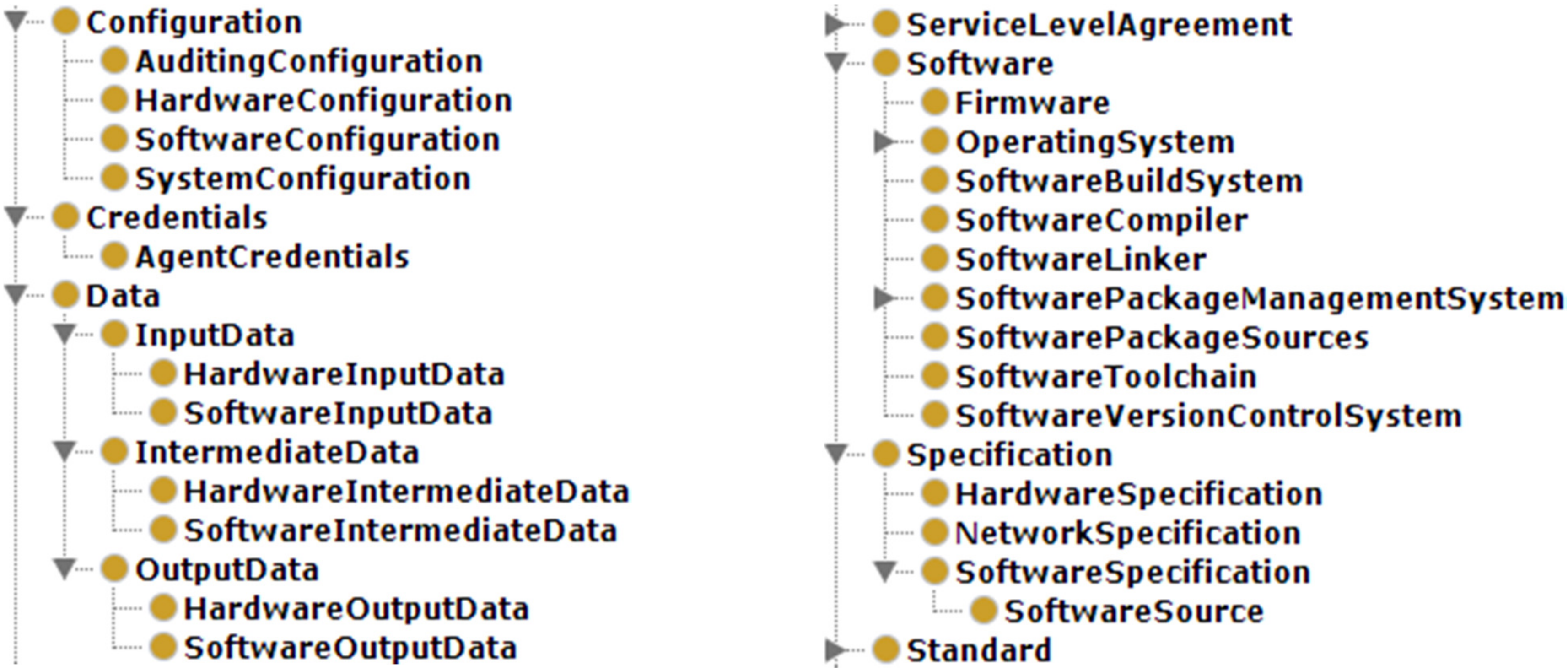
- Captures aspects diverse aspects such
  - Organisations, people, roles, legal requirements, …
  - Software, hardware, external systems, …

Process Context Model

- Similar to Representation Information Networks, but extended to capture process context

- Derived via top-down and bottom-up approach
  - Used enterprise frameworks such as ZACHMAN
  - Used existing taxonomies, such as PREMIS
  - Derived from scenarios developed by project partners
    - Intellectual property rights, data analysis, software escrow, multimedia services, …

Process Context Model



- Configuration
  - AuditingConfiguration
  - HardwareConfiguration
  - SoftwareConfiguration
  - SystemConfiguration
- Credentials
  - AgentCredentials
- Data
  - InputData
    - HardwareInputData
    - SoftwareInputData
  - IntermediateData
    - HardwareIntermediateData
    - SoftwareIntermediateData
  - OutputData
    - HardwareOutputData
    - SoftwareOutputData
- ServiceLevelAgreement
- Software
  - Firmware
  - OperatingSystem
  - SoftwareBuildSystem
  - SoftwareCompiler
  - SoftwareLinker
  - SoftwarePackageManagementSystem
  - SoftwarePackageSources
  - SoftwareToolchain
  - SoftwareVersionControlSystem
- Specification
  - HardwareSpecification
  - NetworkSpecification
  - SoftwareSpecification
    - SoftwareSource
- Standard

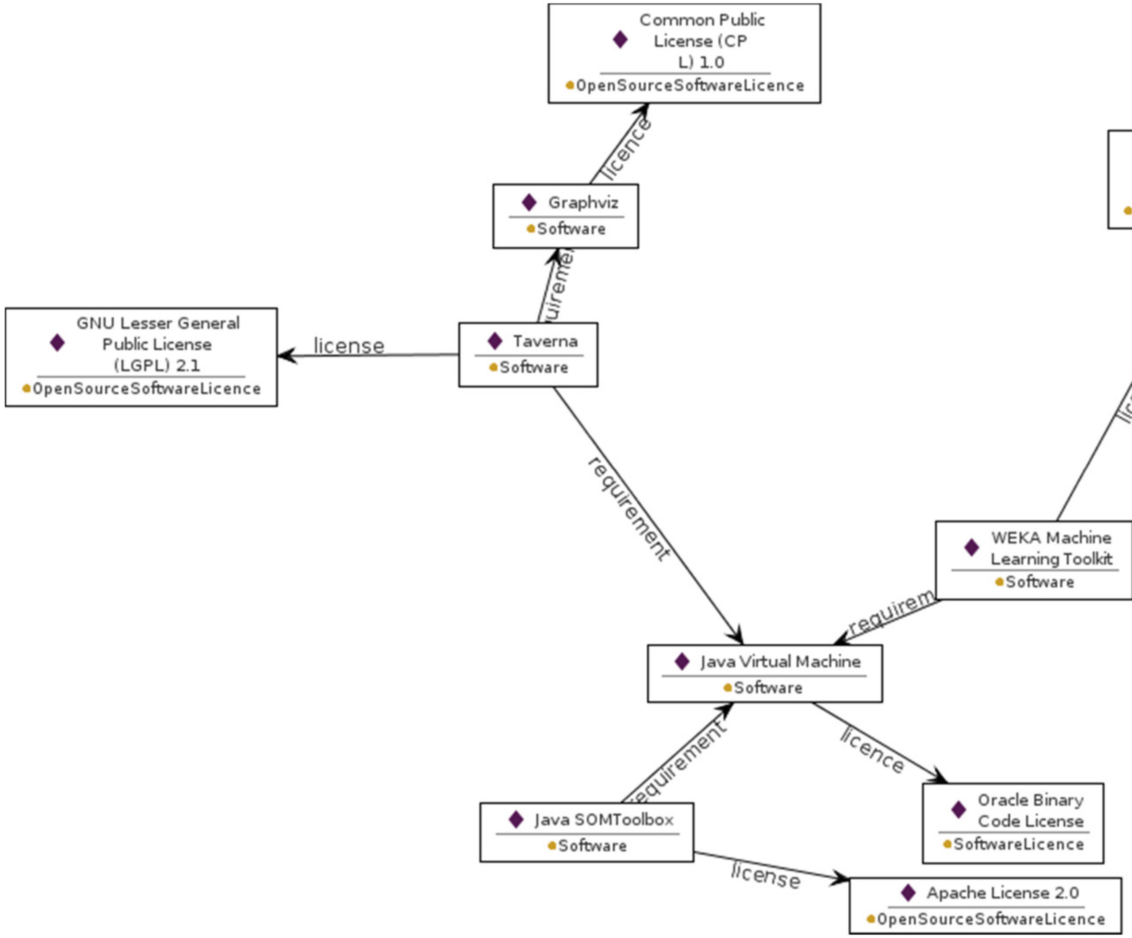
Process Context Model

- For a specific process to be preserved, an instance of this model is created
  - Creating individuals relevant to the process
  - Instance depends on nature of the process – e.g. technical focus in music classification example

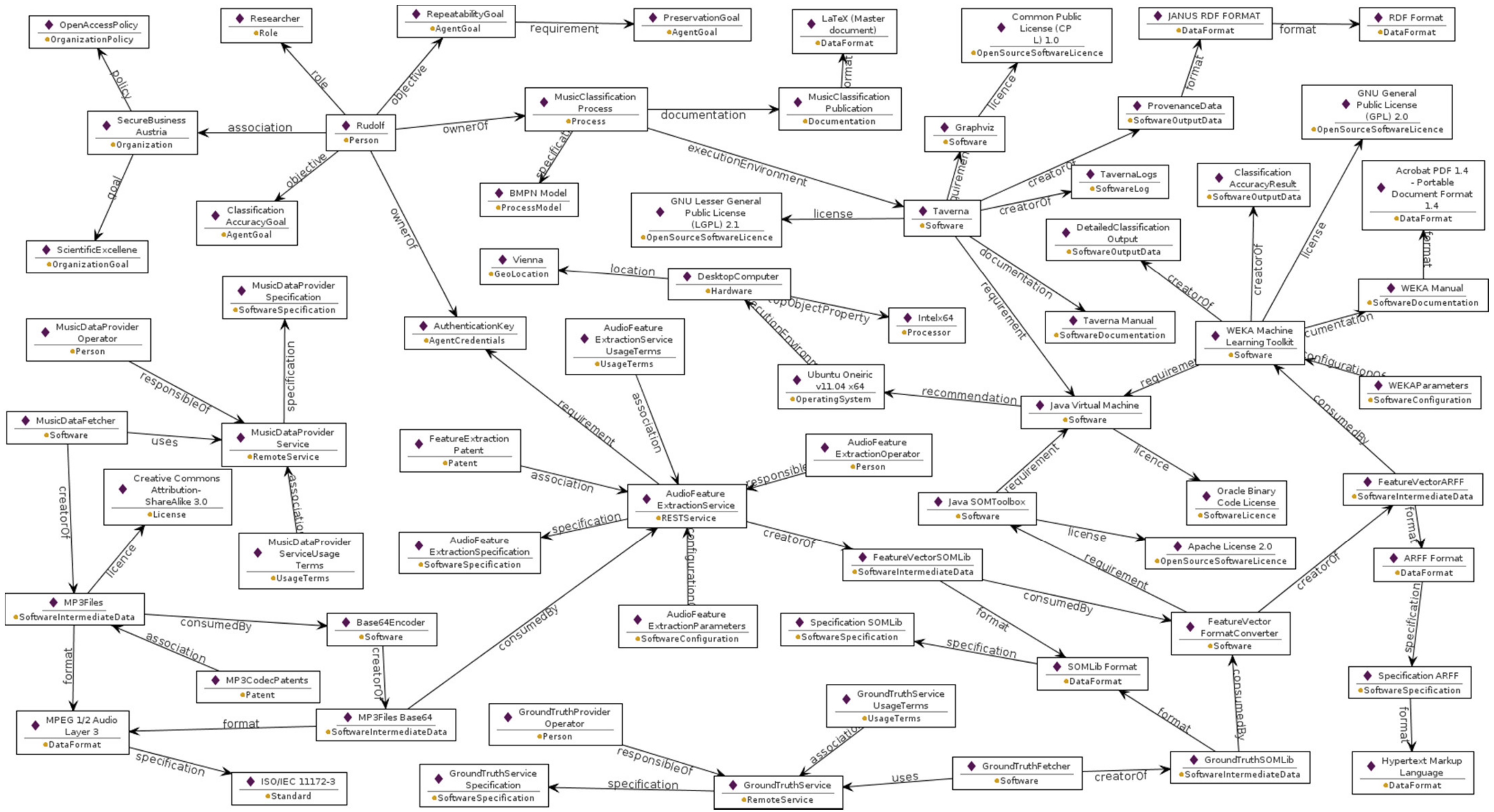
- Semi-automatic approach
  - Capturing tools provide some aspects automatically
  - Knowledge from experts (e.g. process owners) needs to be added manually

Music Classification: Context Model

- Software setup can be automatically detected in OS with software packages (e.g. Linux); allows detection of licenses


Common Public License (CPL) 1.0
OpenSourceSoftwareLicence
licence
Graphviz
Software
requirement
GNU General Public License (GPL) 2.0
OpenSourceSoftwareLicence
GNU Lesser General Public License (LGPL) 2.1
OpenSourceSoftwareLicence
license
Taverna
Software
requirement
license
WEKA Machine Learning Toolkit
Software
requirement
Java Virtual Machine
Software
requirement
licence
Java SOMToolbox
Software
Oracle Binary Code License
SoftwareLicence
license
Apache License 2.0
OpenSourceSoftwareLicence

Music Classification: Context Model



Preservation Actions

- Early stage of research
- Likely a combination of existing approaches
  - Documents needed in the process can be migrated or viewed via emulation software
  - Software and hardware can be emulated
  - External services are difficult
    - Simulation via mock-up services, for deterministic cases

  - Virtualisation to allow archival of modified systems

Validation and Verification

- Need to verify that the process execution is the same as before
- Need to define points of measurement as *significant properties* that can be compared
  - Causal relationships and information flow as significant property
- Music classification example: significant properties are the input and output data, as well as the data exchanged between intermediate steps
  - Captured in workflow systems as *provenance data*
    - Capturing more difficult in less formalised executions

Validation and Verification

- External services and third party libraries: how to detect changes?
  - Monitoring necessary, via watch service

Soccer/flickr

- Comparison of intermediate results in previously recorded executions and current executions

Conclusion

- Challenges of preserving scientific processes
  - More than just the publication
  - Complex service orchestration
  - Complex software environment; libraries, external services, …
- Context model to capture and describe processes
- Preservation actions: combination of existing approaches
- Later verification and re-execution is a crucial task
  - Capturing is not enough, validation is required

Thank you for your attention!

Contact:
rmayer@sba-research.at
www.ifs.tuwien.ac.at/~mayer/