Workshop: Recomputability 2014

11. November, 2014

Emulation-as-a-Service

Workflows and Infrastructure to Support Recomputable Science

UNI FREIBURG

Albert-Ludwigs-Universität Freiburg

Dennis Wehrle (dennis.wehrle@rz.uni-freiburg.de)

Observations

UNI FREIBURG

- Currently we observe a fast technical development
  - New tools, data-formats and methodology
  - Data-driven science builds on data + software
  - → Processing environments matter
    - To verify, replicate & re-use
- Research data management gets more complicated
  - Quality assurance
  - Peer-review
  - Verification
  - Replication
  - Re-use
  - Risk assessment
  - Long-term availability
- Tools for management and preservation of data processing environments are required

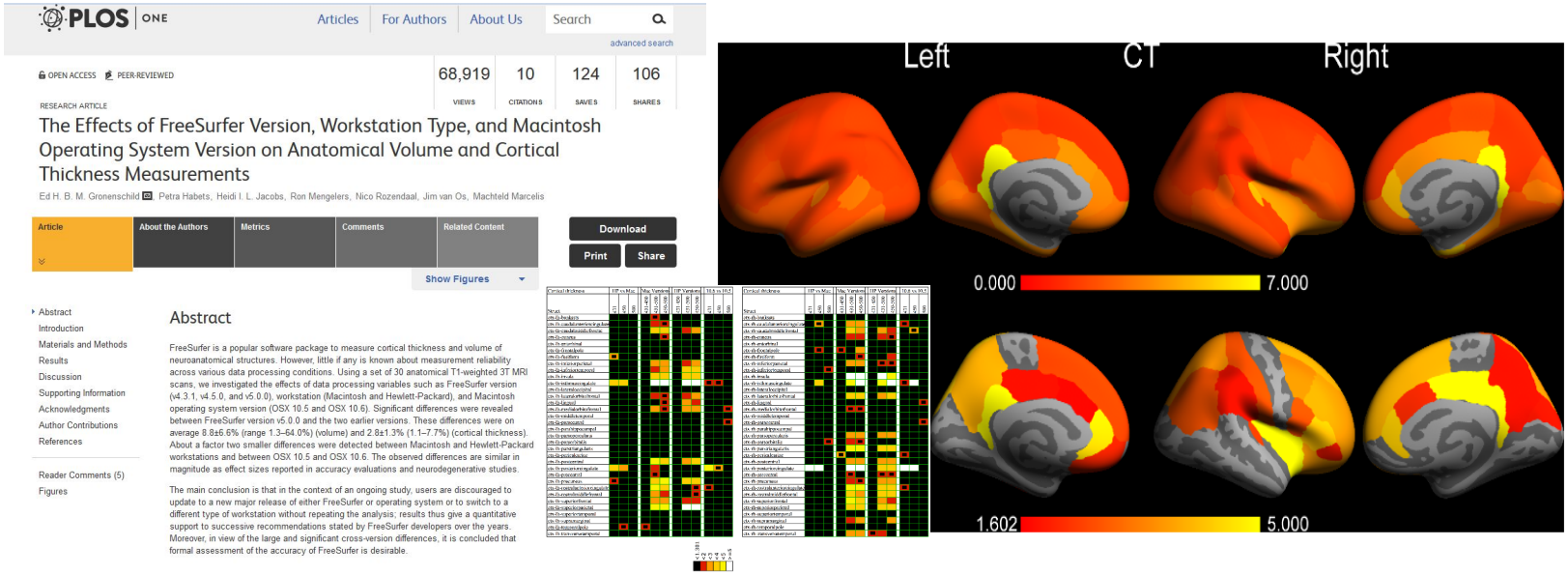
How to Use Emulation as a Tool?

UNI FREIBURG

1. Contextualization
   - Describe & preserve data processing environments
2. Generalization
   - Decouple environments to run „everywhere“
3. Preservation Planning
   - Prepare environments to run “forever”
4. Publication
   - Citation of data in context
5. Access
   - Prerequisite for peer-review, verification, replication, re-use

Emulation as a Tool: Contextualization

- Describe & preserve data processing environments
  - Software components evolve over time
    - Undesired side effects on other components are possible
  - Even apparently insignificant changes to the processing environment may alter the result (statistically) significantly[1]



[1] Gronenschild, Ed HBM, et al. "The effects of FreeSurfer version, workstation type, and Macintosh operating system version on anatomical volume and cortical thickness measurements." PLoS One 7.6 (2012)

Emulation as a Tool: Generalization

UNI FREIBURG

- A "**generalized**" environment depends only on a few (documented and well understood) interfaces between OS and machine
  - Goal: releasing an environment from a specific setup
    - E.g. a scientist's workstation, a specific lab setup

  → An emulated environment can/should run everywhere

- Generalizing an environment is then a first step towards preservation planning
  - Depends on the availability of emulators (of course)
  - If a migration to a new emulator is required the migration process should be uniform among all preserved environments

  → If we are lucky the environment is available "forever"

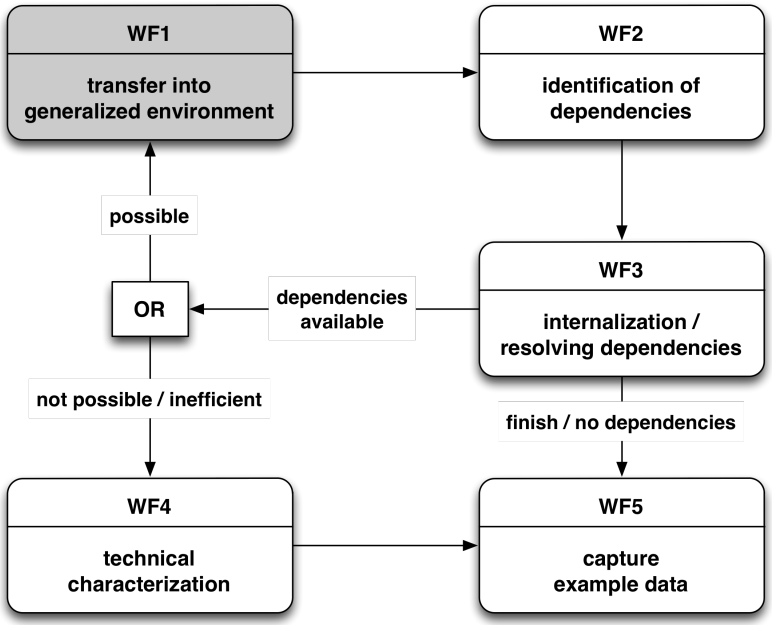
Emulation as a Tool: Generalization

→ Remaining tasks:

- Determine (implicit/external) dependencies → "**expectations**"
- Create risk register → PP / risk assessment & risk management
- Quality assurance and peer-review of the generalization process

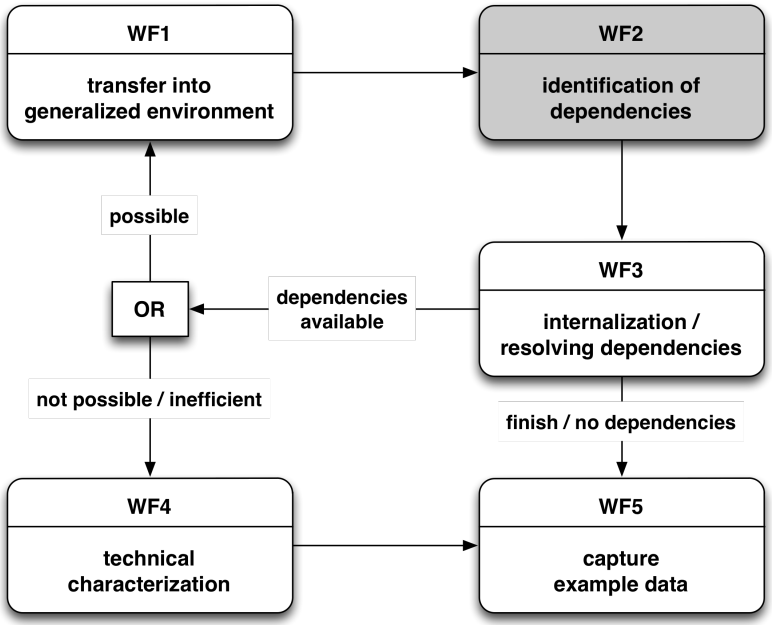
Workflow: Making Expectations Explicit

- WF1 – Migration to a generalized environment
  - First step towards a stable environment
  - Migrate the system into an emulated / virtualized environment
  - Exchange of specific hardware configuration with a well documented and understood configuration



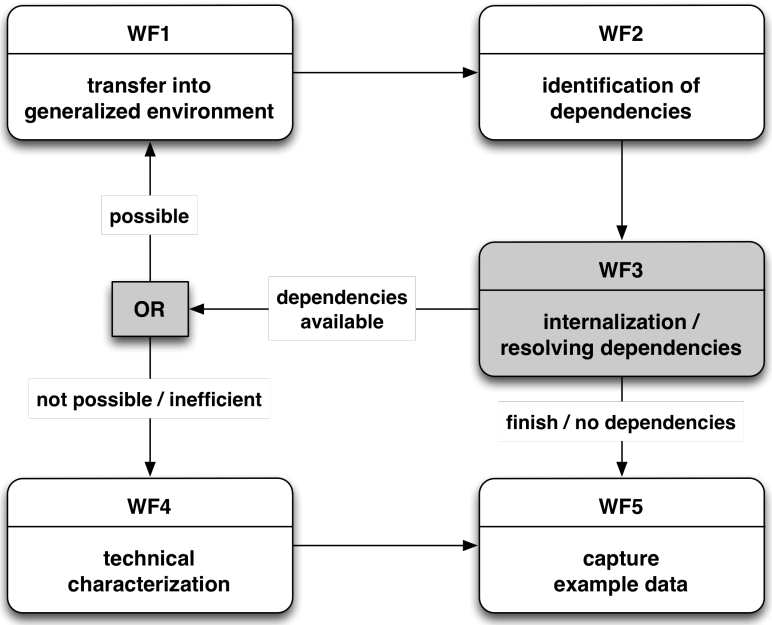
Workflow: Making Expectations Explicit

- WF2 – Identification of process (inter-) dependencies
  - All dependencies with external entities have to be identified
    - → Complete assessment of long-term availability becomes possible
    - → Environment should be independent of ext. dependencies
      e.g. future changes or discontinued services



Workflow: Making Expectations Explicit

- WF3 – Internalize / resolve dependencies
  - Internalize dependencies
    - (recursively) migrate external dependencies into a new generalized environment
  - Resolve dependencies
    - Copying necessary data or installing services within the same environment

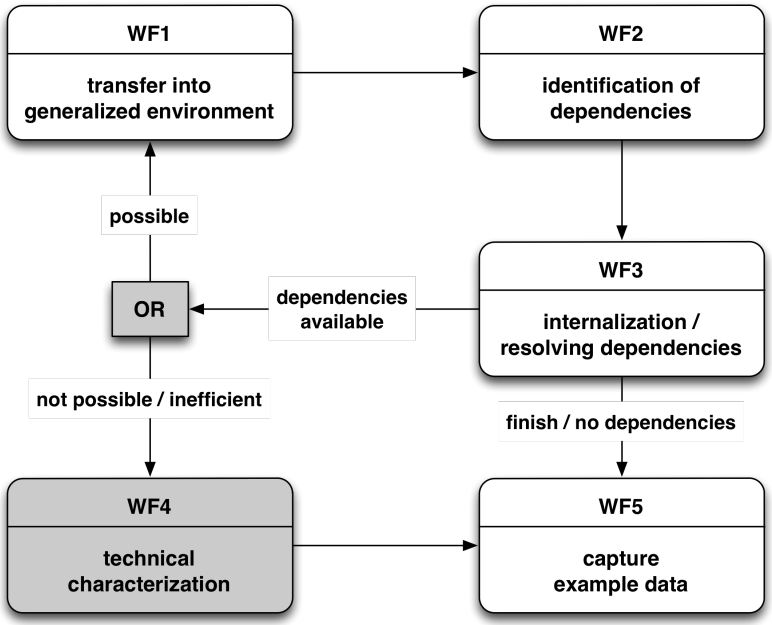


Workflow: Making Expectations Explicit

- WF4 – Technical characterization of external interfaces
  - Characterization
    - Expected data format
    - Technical description of external interfaces

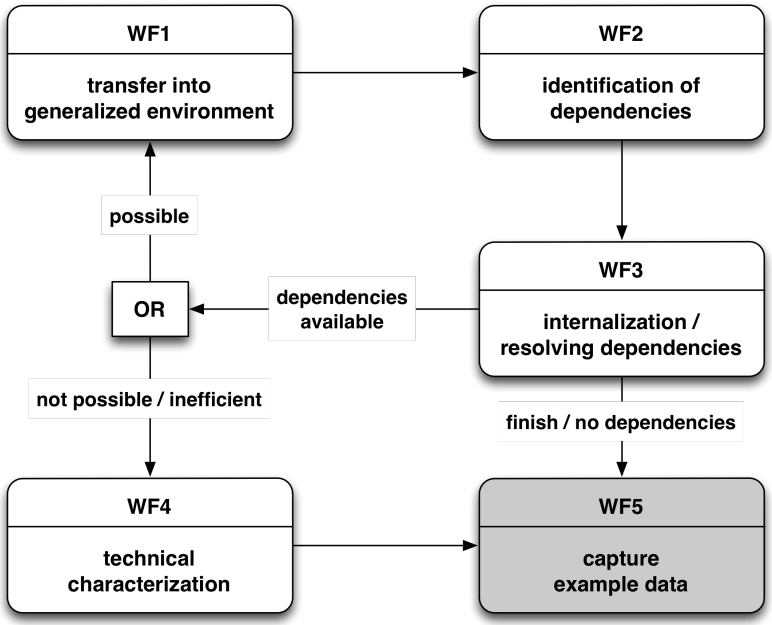
  → Future emulation or simulation of APIs and data structures

  → Keep scientific workflow operational



Workflow: Making Expectations Explicit

- WF5 – Capture samples of external data
  - Capture samples of input, intermediate, result and external data
  - Possible: (automatically) test and verify
    - Emulated interfaces / data == original data-set



Emulation as a Tool: Verification

UNI FREIBURG

- Technical verification
  - Draw conclusions about (generic) technical aspects
  - Strong indication that an environment has been captured, described and preserved completely
  - Abstract description of
    - Input data
    - Technical procedure of scientific workflow
    - Expectations regarding the result
  - Base for full contextual verification

Emulation as a Tool: Verification

UNI FREIBURG

- Full contextual verification
  - Only respective scientific community is able to assess the workflow's completeness and correctness

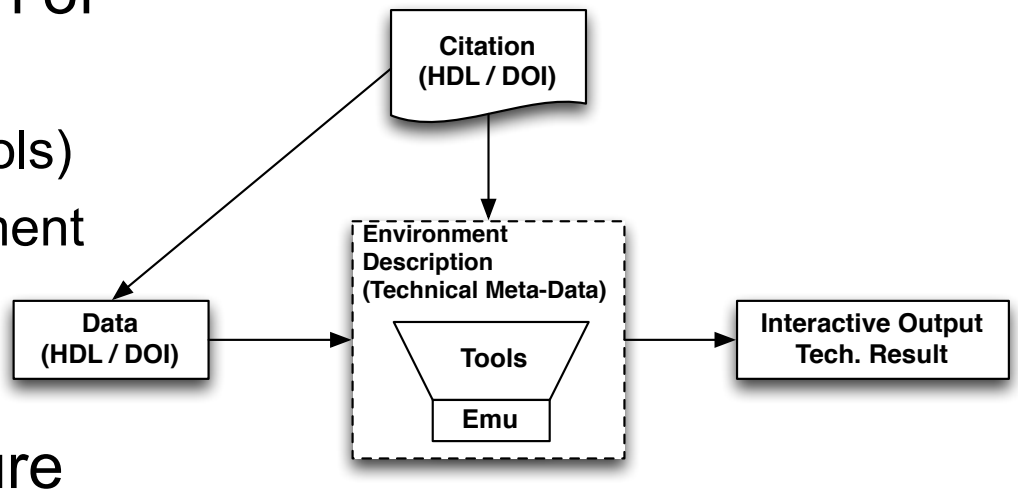
  →Requires support for
    - Citation of data in context
    - Peer-review

Peer-Review, Publication & Citation

- Preservation of contextualized and generalized environment enables publication of data and methods

- Publication & Citation of
  - Data
  - Environment (incl. tools)

  → Operational environment



→ Requires infrastructure

Emulation as a Tool: Handle It!

- Example: Create a **Handle** (hdl) to cite environment + data
  - HTML5 rendering in the browser, interactively usable
  - http://hdl.handle.net/11270/b69dd2d2-6db8-4262-9cd9-c1f2e3783be7

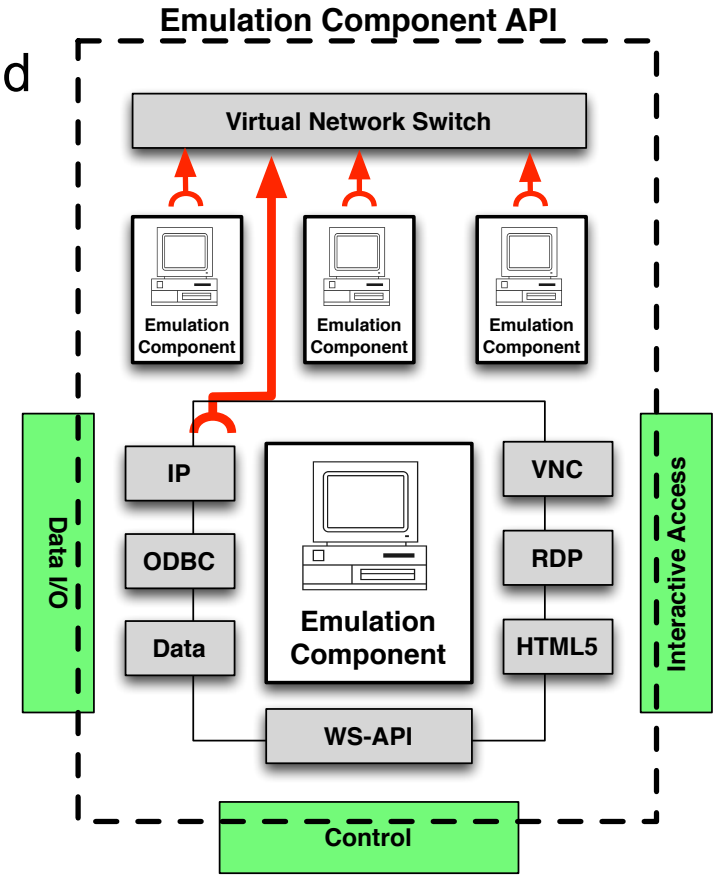
Emulation as a Tool: Does it Scale?

UNI FREIBURG

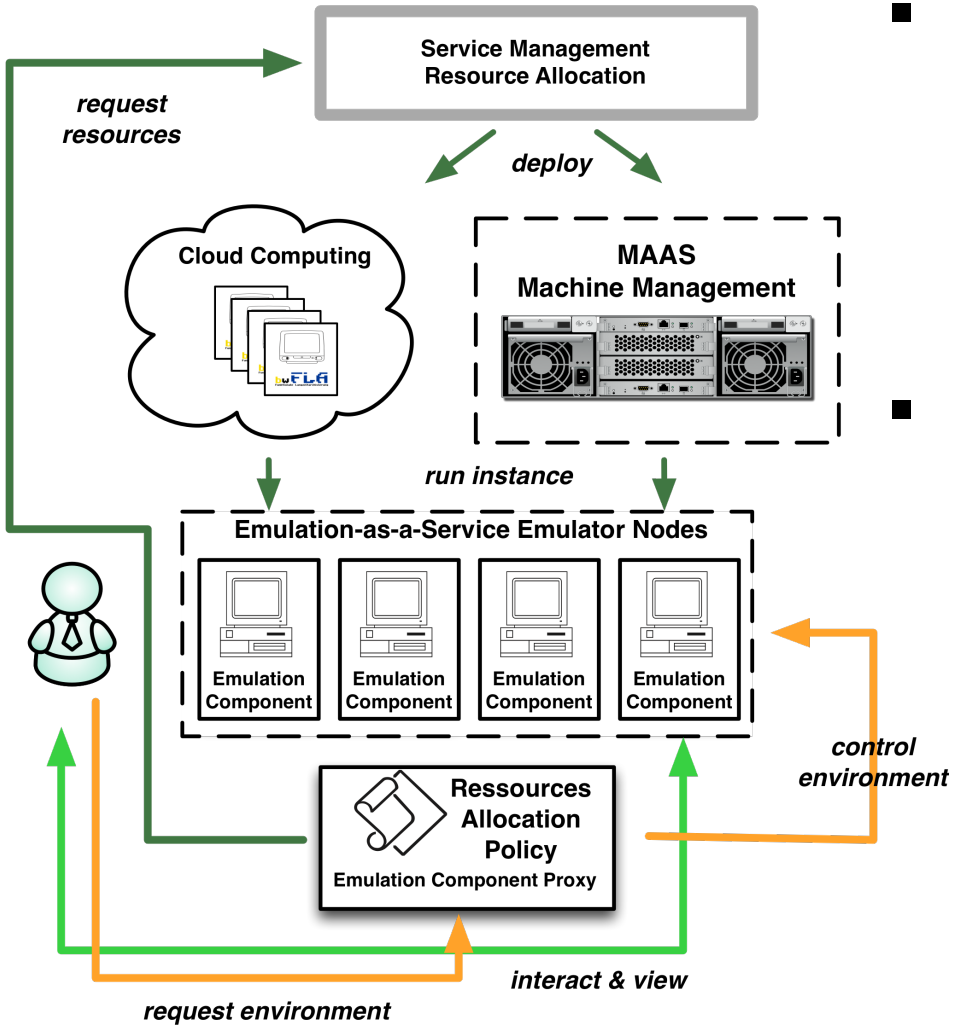
- bwFLA: Emulation-as-a-Service
  - There is only a limited number of computer platforms
    - → Centralize management and maintenance
    - → Share curating costs e.g. keeping emulators alive and usable

bwFLA: Emulation Component

- Unified access to emulation:
  - Encapsulation of different emulators and technology to common component
  - Attachment of user-media
    - dynamically (e.g. Floppy, CD-Rom)
    - permanent (e.g. HDD)
  - Interactive access to emulated environments (e.g. HTML5 viewer)
  - Technical interaction with the environment (IP, specialized protocols)
  - Main building block for complex environments
    - Client/Server etc.
  - API exposed as Web Service (WS)
    - Interoperability to other systems



bwFLA: Emulation-as-a-Service



- On-demand Resources
  - EaaS components require almost no statically allocated resources
  - Allocation of computing resources „on-demand“
- Example:
  - 96 CPUs (Blade-Cluster / Demo)
  - 16 CPUs (Blade-Cluster / Testing)
  - On-demand resources via Cloud Computing
  - Currently supported
    - Amazon EC2
    - OpenStack

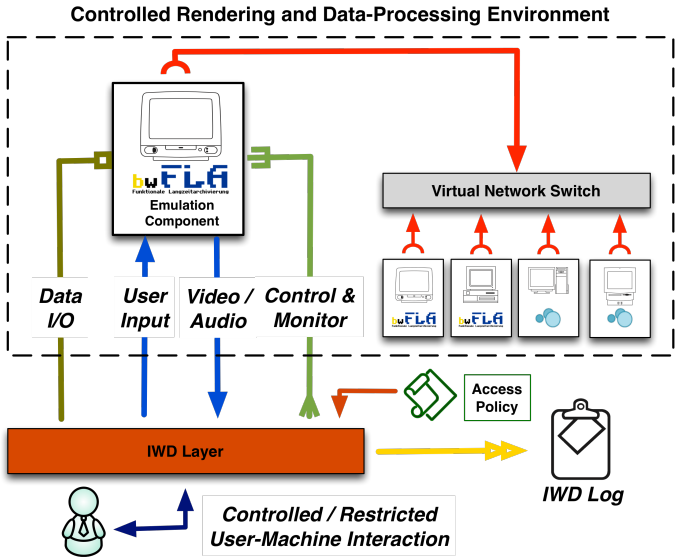
Public Verification vs. Privacy Issues

UNI FREIBURG

- However, publication of research environments data brings its own challenges
- Research data may contain
  - Personal data
  - Quasi identifier
  - Trade secrets
  - Intellectual property
- While we require
  - Authentic replication of a scientific process
  - Keep a digital artifact's (functional) utility
  - Avoid domain specific solutions

Controlled Processing Environment

- Two main concepts
  1. Fully encapsulate the environment including network
     - Object owner remains in control of the data and environment, i.e. the data / environment cannot be copied
  2. Controll any interaction between user and environment
     - Interactions can be captured and restricted (Interactive Workflow Description IWD)



Conclusion

UNI FREIBURG

- Emulation can be a useful tool for research data management
  - provides functional access to data in context
  - enables preservation of scientific models and methodology
  - enables citation of data in context
- Emulation-as-a-Service is able to provide easy to use and cost-effective access to emulation technologies
  - e.g. serves as basic infrastructure for scientific peer review
- More information & demo system:
  - http://bw-fla.uni-freiburg.de
- Examples
  - http://rhizome.org/editorial/2014/jun/24/emulating-bomb-iraq-arcangel/
  - http://hdl.handle.net/11270/b69dd2d2-6db8-4262-9cd9-c1f2e3783be7

About licenses

UNI FREIBURG

1. We only provide technology => user have to have the proper license
2. Big software companies know the problem

„The agreement between the National Archives and Microsoft centres on the use of virtualisation.

The archive will be able to read older file formats in the format they were originally saved by running emulated versions of the older Windows operating systems on modern PCs.“ [2]

[2] Warning of data ticking time bomb, http://news.bbc.co.uk/1/hi/6265976.stm