Pericles Extraction Tool and Significant Environment Information

Fabio Corubolo, University of Liverpool
11 February, IDCC 2015, London
Objective

Ensure long term usability of DOs

Observation:
Use of DO $\Rightarrow$ access to DO’s environment
No object is an island, entire of itself

- Digital objects are used in a rich environment

Present

- Digital object
- Ext. Metadata
- Environment

Future

- Digital object
- Storage
Define a broad set of information

Consider its significance and purposes

Explore pragmatic methods to collect such information
Environment for a DO

- Technical system information
- DO metadata
- User, policy, process information
- Information necessary to use the DO:
  - Auxiliary data (e.g. calibration data)
  - External documentation (e.g. related documents)
  - Implicit knowledge (e.g. user knowledge about relevance in relation to purpose)
  - ...
Environment information definition

- All the entities that have some relationship to a DO through its lifecycle
- Entities: DOs, metadata, policies, rights, services, users, etc.

Refinement:
- Information about the set of relationships from the DO to any related objects
Significant Environment Information

- DOs are preserved for different uses, purposes
- Purposes give scope to the dependent environment information
- Weights can express based on purpose

(definition)
SEI is the set of relationships between a DO and its environment information qualified with purpose and weights
How and when to collect SEI?

- Observe the *use* of DOs throughout of lifecycle
  - Curation doesn’t start at the archive but *throughout DO’s life*

- Collect dependencies for use (SEI)

- Measure significance

- *Sheer curation*:
  - curation activities integrated in the use workflow;
  - lightweight and transparent
Pericles Extraction Tool (PET)

- Open source* framework – builds on the SEI
- Sheer curation – at the right time and place
- Generic, modular, domain agnostic
- Flexible configuration and profiles
- Monitoring changes in time
- Snapshot of the system environment
- User is in full control of the app and data
- To observe unstructured workflows

* Apache 2.0 licensed, on GitHub
PET Architecture and modules

PERICLES Extraction Tool

Extraction modules
- Storage information
- System specification
- Regex / XPath search
- Graphic properties
- Resource usage
- extract SEI

Profiles
- Define which modules to use for which DOs.
- keep set of DOs
- Inform when to extract information

Environment monitor Daemons

Digital Objects

Environment
How to setup PET for a use scenario

- Install PET, configure, leave it monitoring

- Profile is use case specific

- User interacts with DOs, PET collects in BG
  - Environment information,
  - DO events
  - Changes
General scenario for PET

1. **Collect EI**: User is using a machine, PET installed and running in BG
   --- We are now here ---

2. **SEI graph**: PET data analyzed, relationships between DOs discovered.

3. **Weighted SEI graph**: assign weights to relationships (with purpose and significance)

4. **Graphs can help:**
   1. understand inter-document relationships
   2. appraisal of documents; defining collections
3 Quick scenarios, demos

- Show aspects of PET tool
1 – System information

Pericles Extraction Tool

Select profile: SystemSpecificationSnapshot

SystemSpecificationSnapshot

Extraction Modules

- CPU specification snapshot
- File system information snapshot
- Graphic System properties snapshot
- Graphic card information module
- Installed software snapshot
- Java installation information snapshot
- List of network interfaces
- Network information
- Operating System properties snapshot

Add new modules
Remove selected module

Stop monitor
Snapshot
Show events

Pericles
FP7 Digital Preservation
2 – File information
3 – Monitoring scenario

- **Operator’s task**: resolve anomalies
- **Process**: extensive search in the archived data
- **Issue**: preserve implicit information, help with overload
- **PET task**: record SEI for a specific anomaly
  - monitor environment, record significant events, infer documentation useful to solve the anomaly
- **SEI**: to identify and debug a specific anomaly, that is the implicit operator knowledge
An anomaly is reported in an handover sheet.

The operator proceeds with documentation search and consultation, all tracked by PET.
Future work ideas

- Improve: filtering, dependency inference
- Semantics for SEI and significance weights
- Explore weighted dependency graphs to support appraisal
Can you think of other situations where PET could be useful in your practice?
Thank you!

Get involved! This is open source (¬:  

- https://github.com/pericles-project/pet