OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

INF FAKULTÄT FÜR INFORMATIK

DKE Data & Knowledge Engineering Group

Towards Persistent Identification of Resources in Personal Information Management

Stefan Haun, Andreas Nürnberger

3rd int. Workshop on Semantic Digital Archives

Valetta, Malta Semptember 26, 2013

Content

- Motivation
- Related Work
- Identifiers
- Problems
- Case Study: Personal File System
- Conclusion

Motivation

- Today: many items in Personal Information Management (PIM) are digital
  - e.g. Contacts, Appointments, E-Mails, Documents
- Relationships between entities can be expressed as hyperlinks

  → URI provides a viable scheme for those links
  - But
    - when objects move/change, those links become invalid
    - May not be possible to repair links (read-only media)
- In contrast to many Linked Data Set applications, objects in PIM will change!

- Overall questions:
  - How to avoid broken links by URI scheme design?
  - How to repair a link that is broken nevertheless?

Related Work

- Two main areas
  - Geometry
    - Identify a point or part of an object even if parts of the object change or labels are not avaialble.

  - World Wide Web:
    - Create links that are stable regarding the server infrastructure or storage location

      PILIN (Persistent Identifier Linking Infrastructure)

      Digital Object Identifier (DOI)

      Persistent Uniform Resource Locators (PURL)
    - Digital Forensics

      recognize documents and e-mails between peer

- So far not in the context of PIM
- Solutions rely on centralized databases (like handle systems)

Identifiers

- “Identifier”
  - “any association of a name with a thing”
  - But only if it identifies something!

- *Uniform Resource Identifier* (URI)
  - “a compact sequence of characters that identifies an abstract or physical resource” (RFC 3986)
  - Widespread in WWW and Semantic Web

    (the identifier format?)
  - Used here as well due to its broad support

- Resolution
  1. resolve to a locator, i.e. the location of the resource
  2. retrieve the resource from the location

Problems

- Links can break
  - Example:

    IMAP e-mails are identified by their position in a specific sub-folder.

    If the positions changes or the mail is moved, the link breaks.
  - Links may not be correctible
    - e.g. archive media cannot be adapted (WORM) and outgoing links are no longer valid
    - References may not be known and incoming links cannot be updated

→ How to design links that will not break?

- Handle systems use centralized databases
  - Which may not be available (missing connection, server failure)
  - Registration can be quite expensive! (like DOI)

→ Can stable links be designed without a central registry?

Case Study: Personal File System (1)

- Personal files on the Desktop PC

  Problem: How can personal files be referenced?

- Identifiers
  - File Path (RFC1738)
    - **`file://C:/Documents%20and%20Settings/user1/...`**
    - Only valid in scope of the local machine
    - Breaks if the file is moved to another location
    - Identifier == Locator, can be resolved without external database
    - Suitable for stable paths.
  - Magnet Links
    - **`magnet:?xt=urn:sha1:YNCKHTQCWBTRNJIV4WNAE52SJUQCZO5C`**
    - Identifies file by its content
    - Breaks if the file changes
    - Needs resolution, but database can be built locally
    - Suitable for stable file content.

Case Study: Personal File System (2)

- Heuristics
  - Use heuristic to determine if files are equal
  - Example: If one file is missing and a new file appears, the file may just have moved (done in GIT version control system)
  - Using methods from *duplicate detection*
  - May lead to false positives!
  - Increase quality by adding meta-information to the URI

- Two corner cases of file usage:
  1. The generated identifier references a stable content.
  2. The generated identifier references a certain path of a file, i.e. *move* or *rename* operations will not be applied.

→ (How) can these cases be distinguished?

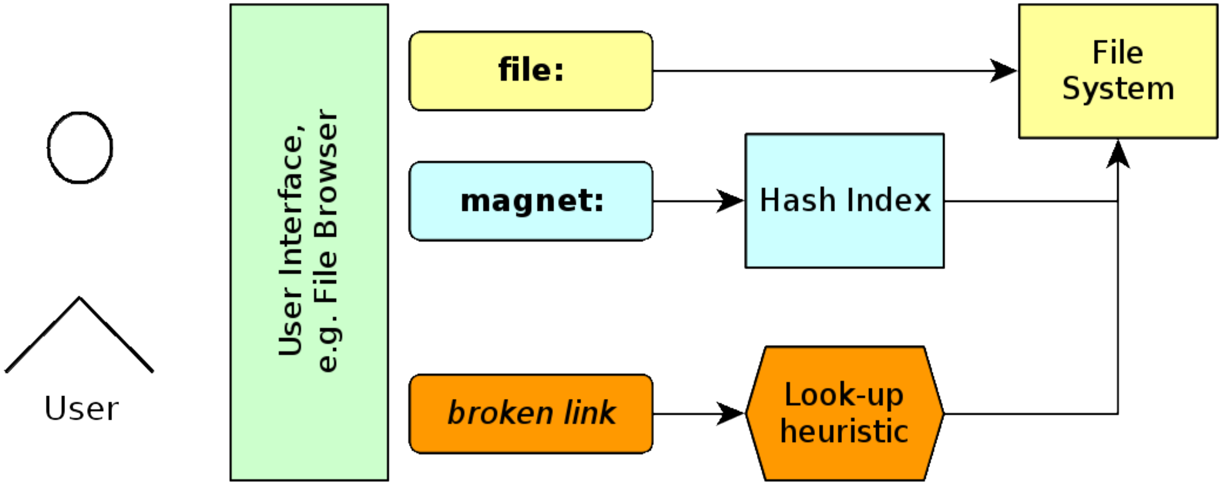
Case Study: Personal File System (3)

- Example Architecture

  User Interface uses URIs only

  Depending on URI namespace:

  - Access via file path
  - Path lookup in local hash index (magnet)
  - Fallback: Use a look-up heuristic if the link is broken



Conclusion

- Stable identifiers are necessary in today's PIM
- URI scheme is viable, but designing URIs needs
  - Stability
  - Independence from centralized databases

- Some questions have to be answered:
  - How to design links that will not break?
  - Can stable links be designed without a central registry?
  - In the context of personal file systems:
  - How can personal files be referenced?
  - (How) can the use-cases be distinguished?

- Next:
  - Find suitable URI schemes for further PIM elements (e-mail, contact, appointment)
  - Map those schemes to existing systems.

Thanks for your attention! :)