**Simone’s Thesis Plan**

Title

Declaration

Acknowledgments

Abstract

Table of Contents

Abbreviations and Symbols

1. Introduction
   1. Problem definition
   2. Fundamental challenges
   3. Use cases
      1. Actors
      2. Terms
      3. Use Cases: fragmentation, evolution, metadata, provenance, space definition
   4. Hypothesis
   5. Thesis contributions
      1. Primary contributions
      2. Secondary contributions
   6. Thesis structure
   7. Conclusions
2. Literature Review
   1. Data management
      1. Definition
      2. Dimensions: Adding, storing, fetching, updating, deleting, sharing, metadata, search, aggregation, versioning
   2. Namespace
      1. Dimensions: Naming, Scalability, Computability of the name, security
      2. Hierarchical DNS
      3. P2P
   3. Storage solutions
      1. File systems (local, networked, distributed)
      2. Cloud solutions (e.g. dropbox, onedrive, google drive, creative cloud, etc)
      3. P2P (ipfs, camlistore, oceanstore, ivy, etc)
      4. DAM solutions
   4. Databases
      1. Relational
      2. NoSQL
      3. Triplestores
      4. Graph
   5. Version Control Systems (VCS)
   6. Metadata, Context, Provenance and Search
      1. Definitions and taxonomies
      2. Existing solutions
      3. Context
      4. Provenance (prov)
      5. Search + systems like spotlight
   7. General Discussion
   8. Conclusions
3. Design of the Sea of Stuff
   1. Requirements
   2. User model
   3. Namespace
   4. Manifest model
      1. Atoms and Atoms Manifests
      2. Compounds
      3. Versions and Assets
      4. Discussion
         1. Security advantages and issues
   5. Metadata model
      1. Types of metadata
      2. Querying related data
      3. Limitations
   6. Contextual space
      1. Definitions
      2. Use cases
      3. Taxonomy: identity, time, geolocation, etc..
      4. Design
      5. Example of contexts (e.g. how they are designed, how they work)
      6. Limitations, issues, security implications
   7. Protection
      1. Manifest and data protection
4. SOS Architecture and Implementation issues
   1. Challenges
   2. Architecture and Components
      1. Roles
      2. Patterns
      3. Drawbacks
      4. Resistance to failure
      5. Security issues
   3. Discussion
      1. Security
         1. Introduction
         2. Types of attacks
         3. Solutions
      2. Compatibility with existing technologies
   4. Conclusions
5. Comparative Evaluation
   1. Refer to hypothesis to justify this type of evaluation
   2. Systems to evaluate (based on lit. review)
      1. File Systems
      2. Distributed FS, P2P
      3. Version control systems
      4. Storage services
      5. DAM/CMS
      6. Apps – mobile apps, spotlight, etc.
   3. Model
   4. Operations
   5. Architecture
   6. Conclusions
6. A use case: webdav
   1. Webdav on unix, sos (local only), sos (distributed)
7. Evaluation
   1. Goals – refer to hypothesis
   2. Challenges
   3. Experimental design
      1. Experiment 1
      2. Experiment 2
      3. Experiment 3
   4. Test bed/configuration used in the experiments
   5. Evaluations
      1. Results
   6. Conclusions
8. Future Work
   1. Model and architecture variation
   2. Scalability
   3. Metadata/context
   4. Security
   5. Integration with existing silos/architectures
9. Conclusions
10. Appendix
    1. Node Configuration
    2. Experimental configuration