Context Spaces as the Cornerstone of a Near-Transparent & Self-Reorganizing Semantic Desktop

Christian Jilek, Markus Schröder, Sven Schwarz, Heiko Maus, Andreas Dengel

Motivation
- existing Semantic Desktops:
  - rather complicated to use
  - not scaling well
  - no real "killer app" available
- 2 categories of Semantic Desktop apps:
  - newly created semantic ones
  - plug-ins to enhance traditional non-semantic ones

Abstract
Existing Semantic Desktops are still reproached for being too complicated to use or not scaling well. Besides, a real "killer app" is still missing. In this paper, we present a new prototype inspired by NEPOMUK and its successors having a semantic graph and ontologies as its basis. In addition, we introduce the idea of context spaces that users can directly interact with and work on. To make them available in all applications without further ado, the system is transparently integrated using mostly standard protocols complemented by a sidebar for advanced features. By exploiting collected context information and applying Managed Forgetting features (like hiding, condensation or deletion), the system is able to dynamically reorganize itself, which also includes a kind of tidy-up-itself functionality. We therefore expect it to be more scalable while providing new levels of user support. An early prototype has been implemented and is presented in this demo.

Intuition: every action is performed in a certain context → every stored information item is related to ≥1 contexts

Context Spaces
- extend Semantic Desktop technology with context spaces
- enable users to directly interact with and work on these context spaces
- make existing applications respect these context spaces without further ado

Near-Transparent Integration
- transparent integration into the rest of the system using standard protocols
- a context space is not just a folder → more contextual information, e.g. a context space could represent a meeting → additional information: date & time, repetition (if applies), location, attendees, organizer, etc.

Conclusion
- established foundation for enabling users to interact with and work on context spaces
- making context spaces available in all their applications without further ado
- providing new levels of user support by self-(re)organization

Outlook
- extensive user studies planned
- several features need to be fully implemented

Example of Forgetting

Further Self-(Re)Organization

Context Model