Back to the Future: Bringing Original Hypermedia and Cross-Media Concepts to Modern Desktop Environments

Signer, Beat; Roels, Reinout; van Barlingen, Robert; Willems, Brent

Published in:
Proceedings of Hypertext 2021, 32nd ACM Conference on Hypertext and Social Media, Virtual Event, August 2021

DOI:
10.1145/3465336.3475122

Publication date:
2021

License:
Other

Document Version:
Accepted author manuscript

Back to the Future: Bringing Original Hypermedia and Cross-Media Concepts to Modern Desktop Environments

Beat Signer
WISE Lab, Vrije Universiteit Brussel
Brussels, Belgium
bsigner@vub.be

Reinout Roels
WISE Lab, Vrije Universiteit Brussel
Brussels, Belgium
rroels@vub.be

Robert van Barlingen
WISE Lab, Vrije Universiteit Brussel
Brussels, Belgium
rvbarling@vub.be

Brent Willems
WISE Lab, Vrije Universiteit Brussel
Brussels, Belgium
brwillem@vub.be

ABSTRACT

Over the last few decades, we have seen massive improvements in computing power, but nevertheless we still rely on digital documents and file systems that were originally created by mimicking the characteristics of physical storage media with all its limitations. This is quite surprising given that even before the existence of the computer, Information Science visionaries such as Vannevar Bush described more powerful information management solutions. We therefore aim to improve the way information is managed in modern desktop environments by *embedding a hypermedia engine offering rich hypermedia and cross-media concepts at the level of an operating system*. We discuss the resource-selector-link (RSL) hypermedia metamodel as a candidate for realising such a general hypermedia engine and highlight its flexibility based on a number of domain-specific applications that have been developed over the last two decades. The underlying content repository will no longer rely on monolithic files, but rather contain a user's data in the form of content fragments, such as snippets of text or images, which are structurally linked to form the corresponding documents, and can be reused in other documents or even shared across computers. By increasing the scope to a system-wide hypermedia engine, we have to deal with fundamental challenges related to granularity, interoperability or context resolving. We strongly believe that computing technology has evolved enough to revisit and address these challenges, laying the foundation for a wide range of innovative use cases for efficiently managing cross-media content in modern desktop environments.

CCS CONCEPTS

• **Applied computing → Hypertext / hypermedia creation**; **Document management**; • **Software and its engineering → File systems management**.

HT '21, August 30-September 2, 2021, Virtual Event, Ireland

ACM ISBN 978-1-4503-8551-0/21/08…$15.00
https://doi.org/10.1145/3465336.3475122

KEYWORDS

hypertext, hypermedia, cross-media spaces, file management

ACM Reference Format:
Beat Signer, Reinout Roels, Robert van Barlingen, and Brent Willems. 2021. Back to the Future: Bringing Original Hypermedia and Cross-Media Concepts to Modern Desktop Environments. In *Proceedings of the 32nd ACM Conference on Hypertext and Social Media (HT '21), August 30–September 2, 2021, Virtual Event, Ireland*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3465336.3475122

1 INTRODUCTION

The idea of managing information by defining associations (trails) between documents has been introduced in Vannevar Bush's seminal article 'As We May Think' [3] and later been applied in early hypermedia approaches such as the oN-Line System (NLS) [7] or Xanadu [12]. The concepts of hypertext and hypermedia have been popularised by the World Wide Web, where navigational links can be defined between HTML pages as well as to other types of media such as images or videos. The Web further enables some basic form of transclusion [12] by, for instance, using an image on multiple web pages. It is important to note that the hypermedia solutions envisioned by its pioneers offered more powerful concepts than most people know from the Web. For example, Engelbart's NLS included rich text editing and collaboration based on hypermedia concepts. In contrast, hypermedia on the Web is limited to creating unidirectional and embedded navigational links with a single anchor (source) and a single target. More advanced hypermedia models offer *different types of links* for other purposes such as giving meaning to a piece of content or to an association (e.g. "this is a definition" or "this image is related to that concept"). In some of these models, hypermedia links might also be *bidirectional* with *multiple sources or targets* and point to specific parts of resources. More importantly, a hypermedia document is not necessarily a monolithic file, but might be a composition of small content fragments that are combined into a coherent document with the help of structural links. This allows the same content fragment (e.g. text or image) to be transcluded (reused) in different documents without duplication, and the separation between content, structure and visualisation enables the same document to be visualised differently in different contexts. Given all these rich concepts by the hypermedia community, it is surprising that our daily management of digital files and documents still relies mainly on traditional hierarchical

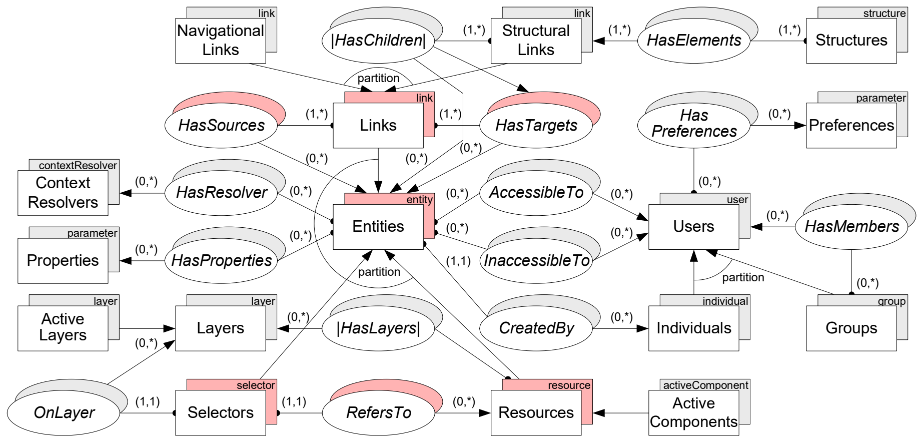

Figure 1: RSL hypermedia metamodel [20]

folder structures without any of the advanced associative linking functionality envisioned by early hypertext pioneers.

The hypermedia research community developed different models and conceptual frameworks with the necessary abstractions for different types of hypermedia systems. In order to support interoperability between different hypermedia models and systems, the Open Hypermedia Protocol (OHP) and some extensions have been introduced [11]. Structural computing further developed the idea of Open Hypermedia and introduced structure awareness by treating structural abstractions as first-class citizens and by supporting multiple domains [13, 24]. Compelling arguments for a general hypermedia metamodel supporting different domains as well as structural abstractions have been provided by Signer and Norrie in the form of the *resource-selector-link (RSL)* hypermedia metamodel [20], which is briefly described in the next section.

We start by introducing the RSL hypermedia metamodel forming the basis of the hypermedia and cross-media solutions that have been developed for different domains and are described in the use case section. After outlining ongoing work on an RSL-based next generation file manager, we present our vision of enhancing our daily work in cross-media information spaces by integrating an RSL-based hypermedia engine at the level of an operating system and thereby enabling innovative forms of information and document interaction as well as cross-application interoperability.

2 RSL HYPERMEDIA METAMODEL

The RSL hypermedia metamodel [20] illustrated in Figure 1 (in OM notation) provides the necessary abstract concepts to support different hypermedia domains. The most general concept is the notion of an `entity`. While an entity represents an abstract concept, there are three concrete subtypes in the form of the `resource`, `selector` and `link` types. A bidirectional link can have one or multiple source entities and one or multiple target entities, enabling associations between any of the three entity subtypes. Note that the `resource` and `selector` types are general types which have to be extended for a specific media type, with the selector allowing us to address specific parts of a resource. The RSL metamodel further distinguishes between navigational and structural links, a first-class concept for arbitrary structural abstractions. Access control is provided at the entity level and context resolvers can further be defined for individual entities, enabling context-aware navigational links as well as context-aware structures.

In order to support simple metadata extensions, any properties (key/value pairs) can be assigned at the entity level. The RSL metamodel further supports the concept of so-called active components representing arbitrary source code that gets executed by a runtime environment when a link to an active component is followed. Recently, support for versioning at the entity level has been added to the RSL metamodel [15]. While we briefly introduced the main concepts of the RSL metamodel, more details can be found in [20].

3 USE CASES

In the following, we describe a number of solutions that have been realised for different domains based on the RSL hypermedia metamodel over the last two decades, in order to show the expressiveness and flexibility of the RSL metamodel.

Interactive Paper. In the context of the interdisciplinary European Paper++ and PaperWorks projects, the iPaper solution has been realised based on the RSL metamodel and its iServer implementation in order to support the seamless transition between paper and digital information and services [17]. An iPaper plug-in provides the media-specific implementation of the RSL model's resource and selector concepts to enable bidirectional navigational links between paper documents and digital media as well as digital services.

Cross-Media Link Browser. The cross-media link browser [21] illustrates the flexibility of the RSL-based link service in terms of the integration of new document formats as well as third-party document viewers. It further highlights the possibility of creating advanced hyperlinks across heterogeneous document formats and viewers that cannot easily be realised with existing linking solutions or the link models of existing document formats.

PimVis. The domain of personal information management (PIM) has been addressed in PimVis [22], a solution for exploring and re-finding digital and paper documents in cross-media information

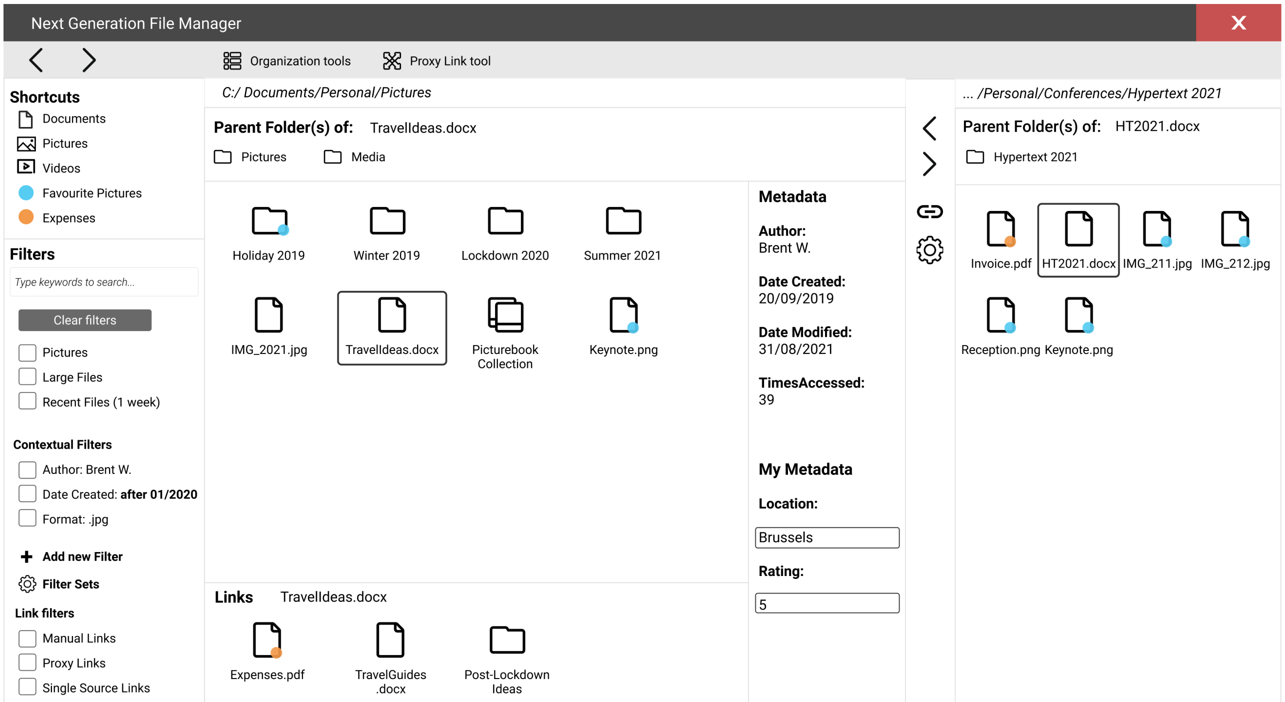

Figure 2: Next generation file manager mock-up

spaces. PimVis provides a unified organisation of digital and paper documents via bidirectional links between the digital and physical space. Thereby, the object-concept-context (OC2) conceptual framework for context-aware personal cross-media information management has been realised as a domain-specific application of the RSL hypermedia metamodel.

MindXpres. The MindXpres presentation platform [15] introduces state-of-the-art presentation functionality based on a new content model for presentation solutions that has been realised on top of the RSL metamodel. MindXpres supports the non-linear navigation within presentations and makes extensive use of RSL's structural links to model any presentation content. A presentation is no longer represented by a monolithic container with its content, but rather all the content is stored in a single RSL instance and individual presentations are defined as context-aware structures over the shared content. Therefore, a presenter can deviate from a presentation's slide order at any time and show content forming part of another presentation (e.g. to answer a question). MindXpres further supports transclusions with optional on-the-fly operational transformations based on RSL's structural links.

eSPACE. In the context of the Internet of Things (IoT), the eSPACE reference framework and conceptual model [16] has been developed based on the RSL metamodel for the unification of the domains of adaptive, distributed and hybrid user interfaces. eSPACE supports an authoring rather than programming approach for the end-user development of IoT applications based on RSL's navigational and structural links, and its concept of active components.

4 NEXT GENERATION FILE MANAGER

As mentioned in the introduction, existing file management (FM) solutions are mainly based on hierarchical folder structures without any associative linking between files. A recent survey from 2019 [10] indicates that the productivity of knowledge workers is heavily burdened by inefficient information management systems, resulting in a 21% loss of an organisation's total productivity and costing nearly $20'000 per worker per year. A detailed classification of file management studies seeking to understand user behaviour along the themes of storing, organising, retrieving as well as sharing has been provided by Dinneen and Julien [4]. They further analysed a larger amount of personal file collections in order to find out how people organise their digital file collections [5]. In existing literature as well in a survey on file management behaviour that we recently conducted, it has been shown that users often encode substantial metadata in file as well as folder (path) names. However, PIM research tells us that we can gain some flexibility by no longer directly encoding this metadata in the file/path names but rather attaching it to so-called "placeless documents" via flexible key/value pairs [6]. Such a flexible annotation and linking of files is, for example, supported in semantic desktop applications such as Haystack [9] via the Resource Description Framework (RDF).

In existing file management solutions, most of the information is stored in application-specific formats, only allowing the application that "owns" the file to use it [18]. The storage of information in monolithic files further prohibits the re-use of parts of a file via transclusion [19]. This leads to the problem of project fragmentation, where different types of information are stored in separated storage containers with no associations (links) between them [2].

A simple example to illustrate this problem are emails, which are often stored in a single inbox file and without the possibility to, for instance, link an individual email to an attachment that has been stored somewhere in the file system. The automatic generation of a bi-directional navigational link between an email and its stored attachment could be used by a file manager to point a user to the original email when later accessing the file. A positive example where the originally proprietary document formats have been made more accessible and reusable by third-party applications is the Office Open XML format which is, for instance, used by all Microsoft Office applications.

RSL offers rich concepts to be used in enhanced file management solutions with associative navigational links between files or parts of files. Structural links can further be used to model any organisational structures on top of files as an alternative to hierarchical folder structures only. While hierarchical organisation strategies can also still be realised via RSL's structural links, files can now belong to multiple organisational structures (multiple classification). A first prototype of an RSL-based associative file manager has been described in [18]. We are currently developing a new version of a next generation file manager connecting to the underlying RSL-based hypermedia engine, and an early mock-up of the file manager is outlined in Figure 2. Note that in the design we have to find a balance between optimally visualising the stored information and at the same time trying to retain familiar file manager features, such as the advanced search functionality on macOS, in order to ease the transition from traditional file managers.

In our recent file management survey, we identified that users often lack some advanced search functionality. Our next generation file manager prototype therefore offers the concept of context-sensitive filters that can be optimised for certain tasks. They can further be combined with tags and associations between files in order to support the retrieval process. Search filters can not only be applied to the content and tags of a file, but also on links between files in order to find certain types of relationships between files. The default view of our file manager consists of two content browsers that are shown next to each other and various other optional views (e.g. search bar on the left) for faceted browsing. Users can easily create new associative links between files or folders in the two browser windows. While we still offer the concept of folders, they are no longer directly mapped to file system folders but rather defined as an RSL-based overlay structure on top of individual files. This enables files to be in multiple folders and we no longer deal with a hierarchical tree structure but a more general graph. It further implies that for a given folder our file manager might have to visualise multiple parent folders.

Note that we offer users different possibilities to organise their files, since they might have their preferences in how to best organise their information spaces. In addition to the grouping of files in folders and subfolders, we also offer the first-class concept of a proxy which can be used to structurally group files located anywhere in the file system by making use of RSL's structural links. These proxies (e.g. `Picturebook Collection` in Figure 2) represent a powerful concept to group files that, for instance, belong to a specific project, without having to move them in the folder hierarchy, and proxies themselves can further be used as sources or targets of navigational as well as structural links. Since we apply some orthogonal structural overlays, a file can belong to any number of proxies. These proxies can then be used in the search bar in order to automatically show related files and in the future, third-party applications that are aware of the advanced file management API might use proxies to optimise certain task. A proxy might for example also be used to represent certain dependencies between files (e.g. a file that depends on some other files for further processing) and if such a file is attached to an email in a hypermedia-enabled email client, it might automatically create a ZIP file containing all the necessary files. While we offer the possibility to organise files and groups of files via navigational as well as structural links, this does not mean that users will always have to create those links manually. Many links will be generated automatically by third-party applications. For instance, if a user stores an email attachment, a hypermedia-enabled email client will automatically create a bidirectional navigational link between the stored attachment and the original email for later retrieval. Further, the system can monitor a user's interactions over time and automatically create navigational or structural links between files that are often accessed at the same time as suggested in FileWeaver [8]. It might even monitor whether users make use of these dynamically generated links and let them fade out over time. Finally, RSL's context awareness at the level of entities will have to be further investigated since it enables navigational links and structural overlays to change based on a given context such as a selected project.

5 HYPERMEDIA ENGINE ROADMAP

We have illustrated the flexibility of the RSL hypermedia metamodel to support different domains and discussed our work on a next generation file manager for enhanced file management based on hypermedia principles such as associative links or overlay structures. While the presented use cases rely on the same underlying RSL model, they are based on separate instances of RSL-enabled data stores on top of existing file and operating systems. In order to enable the integration of information across applications, we propose the embedding of an RSL-based hypermedia engine at the level of an operating system as outlined in Figure 3.

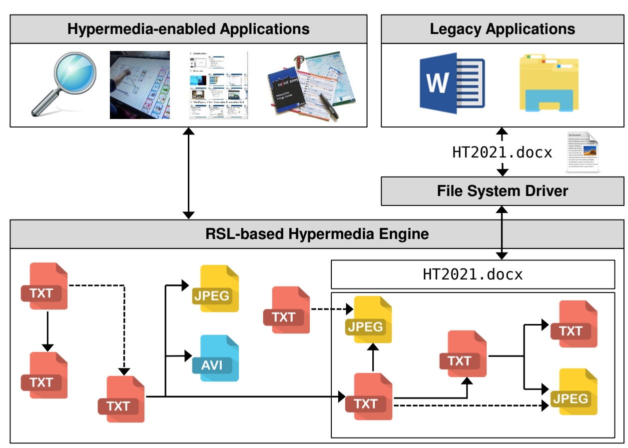

Figure 3: Interaction between different applications and the RSL-based hypermedia engine

Our hypermedia engine will allow applications to store fragments of content, associations, structures and manage content and association metadata. We further plan to address issues such as reuse or security at the core, which is important for a system-wide hypermedia content engine. Traditionally, applications had to be modified or replaced to make them "hypermedia compatible", which is far from desirable. In our approach, the hypermedia engine will be deep enough in the software stack that file system drivers can be developed on top of the hypermedia layer. This way, we do not force applications to adjust and interact directly with our hypermedia engine, but we can already offer novel functionality to existing applications out of the box. Figure 3 shows a text editor such as Microsoft Word simply saving a document to the file system, but our file system driver is aware of specific file formats such as `.docx` files, decomposes the file and stores it as linked RSL data fragments for reuse by other applications. Operations on unsupported file types are simply mapped to equivalent operations on RSL resources and links, supporting any legacy applications.

The native file explorer as well as other applications might still be used as a legacy application to navigate the complex web of linked information via the file system driver. However, the full expressiveness and power of the embedded RSL-based hypermedia engine can only be exploited by hypermedia-enabled applications making use of the API offered by the hypermedia engine, allowing applications to perform operations on resources, links and selectors. Instead of running the RSL-based applications presented in the use case section as well as the next generation file manager as separate applications, they can now be implemented natively on top of the embedded hypermedia engine and share all their data and RSL metadata. We would for instance no longer distinguish between content that is used in a report and content forming part of a presentation. A MindXpres presentation is just another structure defined on top of a single cross-media information space that might also be used by a cross-media PIM solution dealing with digital as well as physical files or the eSPACE application bridging the cross-media information space with IoT environments.

We see an urgent need to improve our daily work by offering system-wide hypermedia concepts as also proposed in structural computing [24] and by using hypermedia as a general method [1]. We are further encouraged to continue the challenging undertaking of integrating an RSL-enabled hypermedia engine by the many opportunities we see for enhanced interactions in cross-media information spaces, as well as by the fact that also some major operating system vendors were already playing with the idea. Almost two decades ago, Microsoft developed WinFS (Windows Future Storage), a storage and data management solution with a relational database and advanced data management features in its core. Unfortunately, WinFS was never released and its development was cancelled after a few years. Nevertheless, in an interview *Bill Gates stated that WinFS is the project that he regrets most never having been released to the public*[1]. Further, some developments in the domain of semantic desktops have outlined the potential of having PIM features at the level of a desktop environment and a number of research prototypes have shown the potential of offering new file management functionality at the level of an operating system [14].

[1]https://www.zdnet.com/article/bill-gates-biggest-microsoft-product-regret-winfs/

We are just at the beginning of implementing and integrating an RSL-based hypermedia engine at the level of an operating system, but already have various RSL-based and hypermedia-enabled applications such as the next generation file manager, the cross-media link browser or the MindXpres presentation platform which can immediately serve as proof-of concept applications. We are currently investigating different possibilities for the implementation and integration of the hypermedia engine. While we could always create an entirely new file system, we might consider to create a layer in between the file system and the application level. In this context we are currently also evaluating FUSE [23], an interesting framework allowing developers to define a file system in user space and bridging it to kernel interfaces.

6 CONCLUSION

We have presented our vision of bringing original hypermedia and cross-media concepts to modern desktop environments by embedding an RSL-based hypermedia engine at the operating system level. By offering transclusion, user management, context awareness and versioning in our hypermedia engine, rich use cases are supported. The fine granularity of linked cross-media content simplifies content reuse and supports application interoperability. Further, the proposed hypermedia-based approach enables innovative forms of cross-media document management. In the near future, a flexible management of cross-media information across application boundaries might become even more important, and the proposed hypermedia engine could serve as a source in emerging augmented reality solutions for augmenting the human intellect.

REFERENCES

[1] Claus Atzenbeck and Peter J. Nürnberg. 2019. Hypertext as Method. In *Proceedings of Hypertext 2019, 30th ACM Conference on Hypertext and Social Media*. Association for Computing Machinery, Hof, Germany, 29–38. https://doi.org/10.1145/3342220.3343669

[2] Ofer Bergman, Ruth Beyth-Marom, and Rafi Nachmias. 2006. The Project Fragmentation Problem in Personal Information Management. In *Proceedings of CHI 2006, SIGCHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, Montréal, Canada, 271–274. https://doi.org/10.1145/1124772.1124813

[3] Vannevar Bush. 1945. As We May Think. *Atlantic Monthly* 176, 1 (July 1945), 35–46. https://doi.org/10.1145/227181.227186

[4] Jesse David Dinneen and Charles-Antoine Julien. 2019. The Ubiquitous Digital File: A Review of File Management Research. *Journal of the Association for Information Science and Technology (JASIST)* 1, 71 (December 2019), 1–32. https://doi.org/10.1002/asi.24222

[5] Jesse David Dinneen and Charles-Antoine Julien. 2019. What's in People's Digital File Collections? *Proceedings of the Association for Information Science and Technology* 56, 1 (2019), 68–77. https://doi.org/10.1002/pra2.64

[6] Paul Dourish. 2003. The Appropriation of Interactive Technologies: Some Lessons from Placeless Documents. *Computer Supported Cooperative Work (CSCW)* 12, 4 (2003), 465–490. https://doi.org/10.1023/A:1026149119426

[7] Douglas C. Engelbart and William K. English. 1968. A Research Center for Augmenting Human Intellect. In *Proceedings of the AFIPS Fall Joint Computer Conference*. Association for Computing Machinery, San Francisco, USA, 395–410. https://doi.org/10.1145/1476589.1476645

[8] Julien Gori, Han L. Han, and Michel Beaudouin-Lafon. 2020. FileWeaver: Flexible File Management with Automatic Dependency Tracking. In *Proceedings of UIST 2020, 33rd Annual ACM Symposium on User Interface Software and Technolog*. Association for Computing Machinery, Virtual Event, 22–34. https://doi.org/10.1145/3379337.3415830

[9] David R. Karger, Karun Bakshi, David Huynh, Dennis Quan, and Vineet Sinha. 2005. Haystack: A Customizable General-Purpose Information Management Tool for End Users of Semistructured Data. In *Proceedings of CIDR 2005, 2nd Biennial Conference on Innovative Data Systems Research,*. cidrdb.org, Asilomar, USA, 13–26. http://cidrdb.org/cidr2005/papers/P02.pdf

[10] M-Files. 2019. 2019 Intelligent Information Management Benchmark. White Paper. https://resources.m-files.com/ebooks/2019-intelligent-information-management-benchmark

[11] Dave E. Millard, Luc Moreau, Hugh C. Davis, and Siegfried Reich. 2000. FOHM: A Fundamental Open Hypertext Model for Investigating Interoperability between Hypertext Domains. In *Proceedings of Hypertext 2000, 11th ACM on Hypertext and Hypermedia*. Association for Computing Machinery, San Antonio, USA, 93–102. https://doi.org/10.1145/336296.336334

[12] Theodor Holm Nelson. 1999. Xanalogical Structure, Needed Now More than Ever: Parallel Documents, Deep Links to Content, Deep Versioning, and Deep Re-Use. *Comput. Surveys* 31, 4 (December 1999), 1–32. https://doi.org/10.1145/345966.346033

[13] Peter J. Nürnberg, John J. Leggett, and Erich R. Schneider. 1997. As We Should Have Thought. In *Proceedings of Hypertext 1997, 8th ACM Conference on Hypertext*. Association for Computing Machinery, Southampton, UK, 96–101. https://doi.org/10.1145/267437.267448

[14] Gabrio Rivera. 2001. *From File Pathnames to File Objects: An Approach to Extending File System Functionality Integrating Object-oriented Database System Concepts*. Ph.D. Dissertation. ETH Zurich. https://doi.org/10.3929/ethz-a-004279487

[15] Reinout Roels and Beat Signer. 2019. A Conceptual Framework and Content Model for Next Generation Presentation Solutions. *Proceedings of the ACM on Human-Computer Interaction (PACM HCI)* 3, 7 (June 2019), 7:1–7:22. https://doi.org/10.1145/3331149

[16] Audrey Sanctorum and Beat Signer. 2019. A Unifying Reference Framework and Model for Adaptive Distributed Hybrid User Interfaces. In *Proceedings of RCIS 2019, 13th International Conference on Research Challenges in Information Science*. IEEE, Brussels, Belgium, 1–6. https://doi.org/10.1109/RCIS.2019.8877048

[17] Beat Signer. 2005. *Fundamental Concepts for Interactive Paper and Cross-Media Information Spaces*. Ph.D. Dissertation. ETH Zurich. https://doi.org/10.3929/ethz-a-005174378

[18] Beat Signer. 2010. What is Wrong with Digital Documents? A Conceptual Model for Structural Cross-Media Content Composition and Reuse. In *Proceedings of ER 2010, 29th International Conference on Conceptual Modeling*. Springer, Vancouver, Canada, 391–404. https://doi.org/10.1007/978-3-642-16373-9_28

[19] Beat Signer. 2019. Towards Cross-Media Information Spaces and Architectures. In *Proceedings of RCIS 2019, 13th International Conference on Research Challenges in Information Science*. IEEE, Brussels, Belgium, 1–7. https://doi.org/10.1109/RCIS.2019.8877105

[20] Beat Signer and Moira C. Norrie. 2007. As We May Link: A General Metamodel for Hypermedia Systems. In *Proceedings of ER 2007, 26th International Conference on Conceptual Modeling*. Springer, Auckland, New Zealand, 359–374. https://doi.org/10.1007/978-3-540-75563-0_25

[21] Ahmed A.O. Tayeh, Payam Ebrahimi, and Beat Signer. 2018. Cross-Media Document Linking and Navigation. In *Proceedings of DocEng 2018, ACM Symposium on Document Engineering*. Association for Computing Machinery, Halifax, Canada, 1–10. https://doi.org/10.1145/3209280.3209529

[22] Sandra Trullemans, Audrey Sanctorum, and Beat Signer. 2016. PimVis: Exploring and Re-finding Documents in Cross-Media Information Spaces. In *Proceedings of AVI 2016, International Working Conference on Advanced Visual Interfaces*. Association for Computing Machinery, Bari, Italy, 176–183. https://doi.org/10.1145/2909132.2909261

[23] Bharath Kumar Reddy Vangoor, Vasily Tarasov, and Erez Zadok. 2017. To FUSE or Not to FUSE: Performance of User-Space File Systems. In *Proceedings of FAST 2017, 15th USENIX Conference on File and Storage Technologies*. USENIX Association, Santa Clara, USA, 59–72. https://www.usenix.org/conference/fast17/technical-sessions/presentation/vangoor

[24] Uffe K. Wiil, Peter J. Nürnberg, and David L. Hicks. 2003. Editorial: Structural Computing: Research Directions, Systems and Issues. *Journal of Network and Computer Applications (JNCA)* 26, 1 (January 2003), 3–9. https://doi.org/10.1016/S1084-8045(02)00056-5