



## DSMW: Distributed Semantic MediaWiki

Hala Skaf-Molli, G r me Canals, Pascal Molli

► **To cite this version:**

Hala Skaf-Molli, G r me Canals, Pascal Molli. DSMW: Distributed Semantic MediaWiki. 7th Extended Semantic Web Conference (ESCW 2010), May 2010, Heraklion, Greece. inria-00540209

**HAL Id: inria-00540209**

**<https://hal.inria.fr/inria-00540209>**

Submitted on 26 Nov 2010

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destin e au d p t et   la diffusion de documents scientifiques de niveau recherche, publi s ou non,  manant des  tablissements d'enseignement et de recherche fran ais ou  trangers, des laboratoires publics ou priv s.

# DSMW: Distributed Semantic MediaWiki

Hala Skaf-Molli, G er ome Canals and Pascal Molli

Universit e de Lorraine, Nancy, LORIA – INRIA Nancy-Grand Est, France  
{skaf, canals, molli}@loria.fr

**Abstract.** DSMW is an extension to Semantic Mediawiki (SMW), it allows to create a network of SMW servers that share common semantic wiki pages. DSMW users can create communication channels between servers and use a publish-subscribe approach to manage the change propagation. DSMW synchronizes concurrent updates of shared semantic pages to ensure their consistency. It offers new collaboration modes to semantic wiki users and supports dataflow-oriented processes.

## 1 Research Background: Collaborative Editing

Semantic wikis allow to create and edit collaboratively semantically annotated documents. However, compared with other collaborative systems, semantic wikis do not support offline work or multi-synchronous editing [1]. In existing semantic wikis, every change in a page is immediately visible for both end users and semantic engines. However, in many cases it is necessary to change multiple pages before to make them visible. Existing semantic wikis cannot prevent users to access, navigate or query inconsistent pages. Moreover, the lack of multi-synchronous support prevents users to work isolated [2] and also prevents semantic wikis to support dataflow oriented processes.

To overcome these limitations, we propose a distributed approach for semantic wikis. In this approach, semantic wiki pages are replicated over a network of interconnected semantic wiki servers. Changes issued on one server are local but can be published to other servers. Remote servers can subscribe to these changes, pull them and integrate them to their local pages. Changes propagation remains under the control of the users. Concurrent changes on a page issued by different servers are handled by a merge procedure.

Using this approach, users can alternate between isolated periods of work and synchronization sequences with remote servers. They can introduce changes to multiple pages before to atomically render these changes public. They can choose when to incorporate, or not, remote changes. In addition, the approach can be the basis for implementing processes in which flows of semantic wiki pages can traverse a network of semantic wiki servers. Each wiki server can implement one or several steps of a particular process for the creation and maintenance of semantic pages.

## 2 DSMW approach

DSMW is an extension to Semantic MediaWiki (SMW) [3]. It allows to create a network of SMW servers that share common semantic wiki pages. DSMW manages the propagation and the integration of changes issued on one SMW server to remote servers on the network. The system ensures the consistency of the whole set of replicated pages.

DSMW users can create and edit semantically annotated wiki pages as with a regular SMW server. Then she/he can manage pages changes as a software developer does with her/his source code using a distributed version control system: she/he can work in isolation while editing pages and semantic annotation on a single server, then she/he can publish part or all of her own changes by pushing them to DSMW public feeds, and she/he can subscribe to any remote public DSMW feeds, pull changes from remote servers and integrate them to the local pages.

The DSMW extension adds two main features to SMW: an optimistic replication algorithm, and an ontology to manage changes, publication and integration of changes sets.

*Page replication in DSMW* is handled by a dedicated replication procedure. Since semantic annotations are embedded in page content in SMW, DSMW replicates only page contents and there is no need to deal with annotations. DSMW uses the Logoot algorithm to synchronize concurrent changes[4]. Logoot guarantees the consistency of the shared pages based on the CCI model (Causality, Convergence, Intentions [5], the model used also by Google Wave). The propagation technic is publish-subscribe: changes issued on one server can be published by pushing them to one or several *pushFeeds*. Remote servers can subscribe to these feeds by connecting *pullFeeds* to existing remote *pushFeeds*. Then, they can pull changes and integrate them to the local pages. Concurrent changes are merged by the Logoot algorithm. Hereafter a brief description of the operations related to replication.

**Save:** the SMW save operation, called when a user saves edit modifications on a page, has been extended to build and log patches. Patches represent changes to a page as a sequence of elementary Insert and Delete operations. Patches are computed by the Logoot algorithm as a diff between the current and the new version of the page being saved. Logoot uses a special index to determine absolute insertion and deletion positions of the elements in a page. Once computed, a patch is applied to the local page and logged.

**CreatePushFeed:** creates a named communication channel to publish local changes. The content of the feed is specified by a semantic query. All pages in the query result belongs to that channel, meaning that changes on these pages will be published through that feed. Note that a page can belong to several channels.

**Push:** the push operation computes the set of patches for a given *pushFeed* to form a *ChangeSet*. A *changeSet* is the ordered set of all patches logged for all the pages belonging to that *pushFeed*. Once computed, the *changeSet* is added to the feed and can then be pulled by remote servers.

**CreatePullFeed:** creates a named communication channel to pull remote changes. A pullFeed is connected to one single remote pushFeed, so a pull feed is defined by the URL of the remote server and the pushFeed name.

**Pull:** the push operation download all the pending changeSets published in the pushFeed connected to a given pullFeed. Patches extracted from the changeSets are then locally applied in the order they appear. To do so, Logoot uses the absolute positions computed during the patch creation to insert or delete page elements.

The DSMW ontology shown in the figure 1 represents all the concepts of DSMW: changes, change sets, push and pull feeds. This ontology makes possible the DSMW users to query the current state of the wiki and its complete history using SMW semantic queries. For instance, queries can extract *the list of unpublished changes* or *the list of published changes on a given channel*. This ontology is populated through the user interaction with the system: all the operations described in the previous paragraph create or delete instances of the DSMW ontology (see [6] for more details).

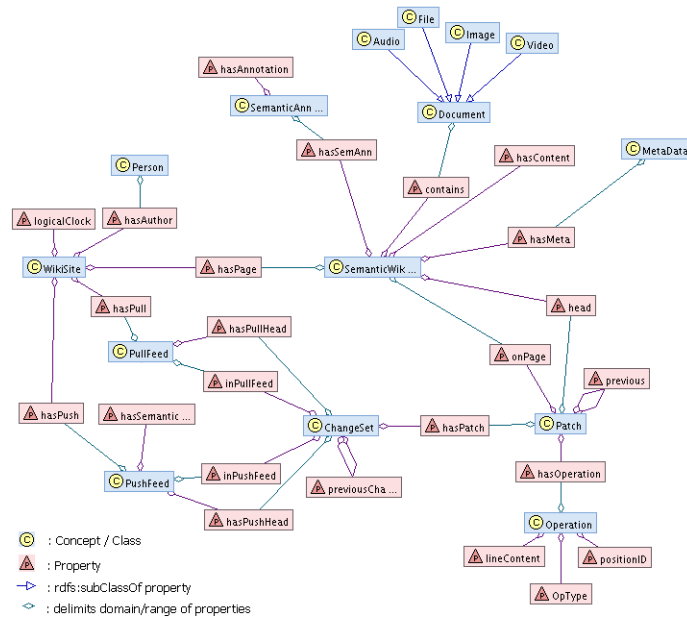


Fig. 1. Multi-synchronous ontology

### 3 Use cases and Applications

DSMW is used in ongoing French national projects: *WikiTaaable* and *CyWiki*. *WikiTaaable* is a distributed collaborative knowledge building systems for cooking recipes [7]. It integrates a case-based reasoning engine. WikiTaaable uses SMW as a central module to manage all data and knowledge used in the system. DSMW supports the humans and machines collaboration by deploying several DSMW servers to implement continuous integration processes as those used during software development. WikiTaaable is accessible at <http://taaable.fr>.

The *CyWiki* project uses DSMW as an infrastructure for the collaborative and assisted transformation of textual content into formal and structured knowledge. The transformation process is a decentralized process in which both human agents and automatic agents (text-mining agents, classification agents) collaborate to build knowledge units (in the form of ontology elements). This knowledge can then be used to query and make reasoning about the content. The experimental and application domain of the project is education.

### 4 System Demonstration

The demonstration scenario will focus on three main use-cases:

*The knowledge aggregation* corresponds to the use of a DSMW server to aggregate and combine wiki pages and knowledge from multiple sources. This server subscribe to these sources by creating pull feeds connected to the public push feeds at each source. By combining these sources, the system can answer new queries that could not be evaluated on a single source. The demonstration example will be the following:

- a first DSMW server holds semantic wiki pages about hotels in a city. Hotels are described with various properties (rooms, prices . . . ) and their location in the city relatively to well-known places (e.g. the train station, the main square),
- a second DSMW server holds semantic wiki pages about touristic information in the city. It describes sites of interest and cultural events with various properties and their location in the city, relatively to well known places.
- a third DSMW server subscribes to the public push feeds of the two previous, and regularly pull them. It then holds semantic wiki pages on both hotels and touristic information and their location in the city, and maintain these pages consistent with the original sources. This server can answer queries that cannot be evaluated on the original sources, typically to find an hotel close to a particular site of interest.

*The knowledge validation steps* corresponds to the use of one or several DSMW server to implement a validation process: prior to rendering public a set of semantic wiki pages, it can be desirable in some cases to validate their content by users or by running non-regression tests. The scenario will be based on the same hotel-tourist example. It consists in adding a fourth DSMW server

that will serve as a public front-end for querying the hotel-tourist knowledge base. The aggregation server will then serve to combine the original sources and validate the new knowledge base. This validation step is done by users verifying the semantic annotations and eventually modify them and running tests by evaluating a fixed set of queries whose results are known and should not change. Once validated, changes are propagated to the fourth server and are thus accessible to end-users. This validation step ensures the consistency and the stability of the final knowledge base. Any change to the original sources is tested, verified and eventually fixed before to be queried by end-users.

*The network construction use case* corresponds to the construction of a network of interconnected DSMW server. The demonstration will show how the push and pull feeds are created on the different servers of the hotel-tourist example, and connected to create the network.

## 5 Conclusion

In this demonstration we have presented a new collaborative tool, called DSMW, to support multi-synchronous collaboration and dataflow processes over semantic wiki pages. DSMW is developed as an extension of SMW. The first public release of DSMW was published in October 2009. A new release DSMW 0.5 is available at <http://dsmw.org>. We continue the development and the research on DSMW. Research concerns divergence awareness in DSMW and the analysis of the social networks built by the collaborative editing.

## References

1. Dourish, P.: The parting of the ways: Divergence, data management and collaborative work. In: 4th European Conference on Computer Supported Cooperative Work. (1995)
2. Molli, P., Skaf-Molli, H., Bouthier, C.: State Treemap: an Awareness Widget for Multi-Synchronous Groupware. In: Seventh International Workshop on Groupware - CRIWG, IEEE Computer Society (2001)
3. Krötzsch, M., Vrandečić, D., Völkel, M., Haller, H., Studer, R.: Semantic wikipedia. *Journal of Web Semantic* **5**(4) (2007) 251–261
4. Weiss, S., Urso, P., Molli, P.: Logoot : a scalable optimistic replication algorithm for collaborative editing on p2p networks. In: International Conference on Distributed Computing Systems (ICDCS), IEEE (2009)
5. Sun, C., Jia, X., Zhang, Y., Yang, Y., Chen, D.: Achieving Convergence, Causality Preservation, and Intention Preservation in Real-Time Cooperative Editing Systems. *ACM Transactions on Computer-Human Interaction* **5**(1) (1998)
6. Rahhal, C., Skaf-Molli, H., Molli, P., Weiss, S.: Multi-synchronous collaborative semantic wikis. In: 10th International Conference on Web Information Systems Engineering - WISE '09. Volume 5802 of LNCS., Springer (2009) 115–129
7. Cordier, A., Lieber, J., Molli, P., Nauer, E., Skaf-Molli, H., Toussaint, Y.: Wiki-*taable*: A semantic wiki as a blackboard for a textual case-based reasoning system. In: SemWiki 2009 - 4rd Semantic Wiki Workshop at the 6th European Semantic Web Conference - ESWC 2009, Grce Heraklion (2009-05-16)