

# paggr - Smart Data Portals

Benjamin Nowack

semsol, Düsseldorf, Germany  
bnowack@semsol.com

**Abstract.** paggr<sup>1</sup> is an interactive application that simplifies the organization and integration of distributed web information. In a few simple steps, users can create personalized, "smart data" portals from a variety of sources and formats such as RDF, remote SPARQL endpoints, microformats, RSS, Atom, RDFa, or selected APIs. Data items in paggr can be linked to each other using an intuitive drag&drop mechanism. Additionally, paggr offers a developer zone where everyone can collaborate on reusable data widgets via simple web forms and a SPARQL-based scripting language with templating mechanism.

## 1 Introduction

The current Web offers a multitude of attractive and user-friendly applications to create and distribute content. Trends like mashups, AJAX interfaces, or portable widgets have significantly improved the user experience and the level of involvement on the Web. The border between consuming and publishing, or even developing is increasingly blurred. The amount of machine-readable data is constantly growing, too. Still missing, though, is an easy way to access, combine, and remix these structured data for everyday purposes.

## 2 Feature Overview

paggr is a personal data organizer that utilizes a widget-based portal metaphor, similar to several services found on the Web already, but with the key difference that integration capabilities are available at the data level, not just visually.

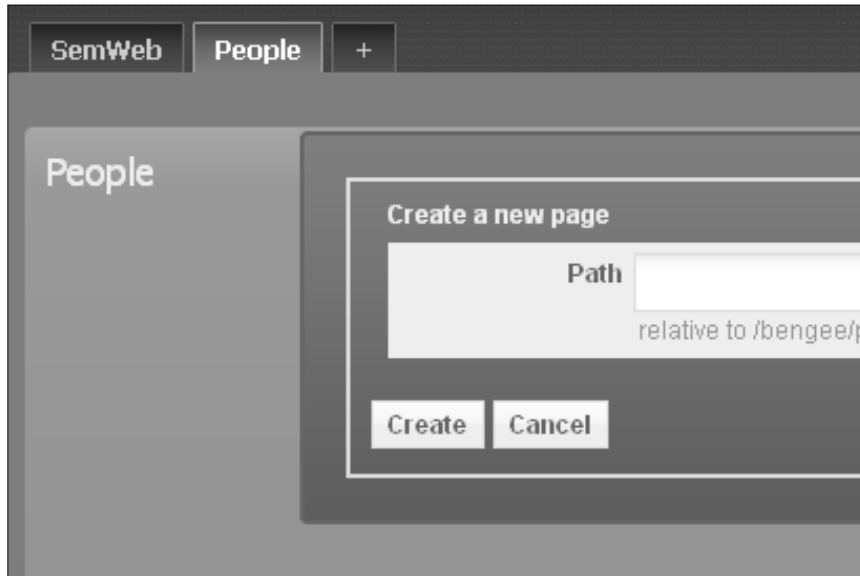
### 2.1 Wiki-like Portal Creation

Each registered user gets a personal RDF store and a Wiki area where pages and a simple sub-page hierarchy can be created. Pages can then be populated with data or markup widgets. It is possible to make selected pages publicly accessible.

---

<sup>1</sup> <http://paggr.com/>

Even though individual widgets can be refreshed using client-side JavaScript code, there is always a server-generated rendering of the whole page available, so that software (not just humans) can access the portals, too.



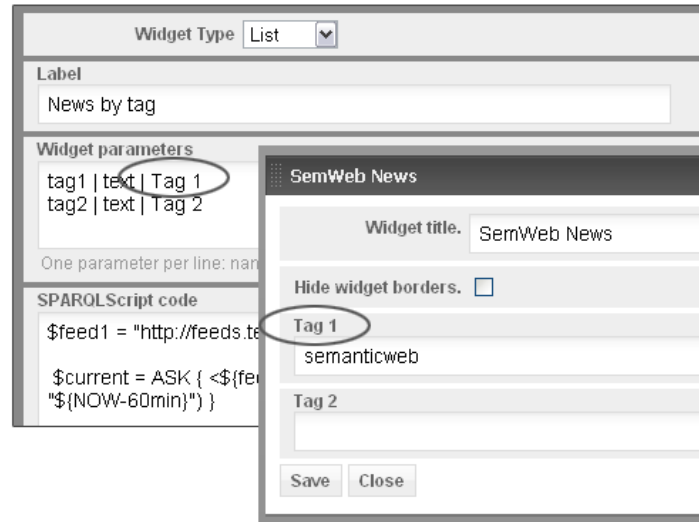
**Figure 1.** Page creation. Page and navigation data is stored in the user's personal RDF store.

## 2.2 Parameterized SPARQL Widgets

paggr provides a developer zone where people familiar with SPARQL can specify small mashups which are then made available as widgets to portal maintainers, without the need for RDF expertise. The widget builder supports a SPARQL extension called SPARQLScript<sup>2</sup> which enables a limited, but working federation of SPARQL sub-queries to multiple endpoints. SPARQLScript also provides a templating mechanism, so that complete widgets can be defined via simple web forms. In order to improve the applicability of widgets, paggr allows the definition of parameters which can be set during widget instantiation (i.e. when a widget is added from the gallery to a portal). Figure 2 depicts a simple parameter definition in the widget builder, together with the generated end-user input form.

---

<sup>2</sup> <http://arc.semsol.org/docs/v2/sparqlscript>



**Figure 2.** The widget builder allows the definition of parameterized SPARQL operations.

### 2.3 URI-aware Widget Items.

Apart from the general possibility to join results from several online sources on the server, paggr also tracks when the user tries to relate individual resources on the client. Widget entries are marked up with their Web identifier, so that drag and drop operations can be detected and used to invoke SPARQL queries or simple annotations (Figure 3).



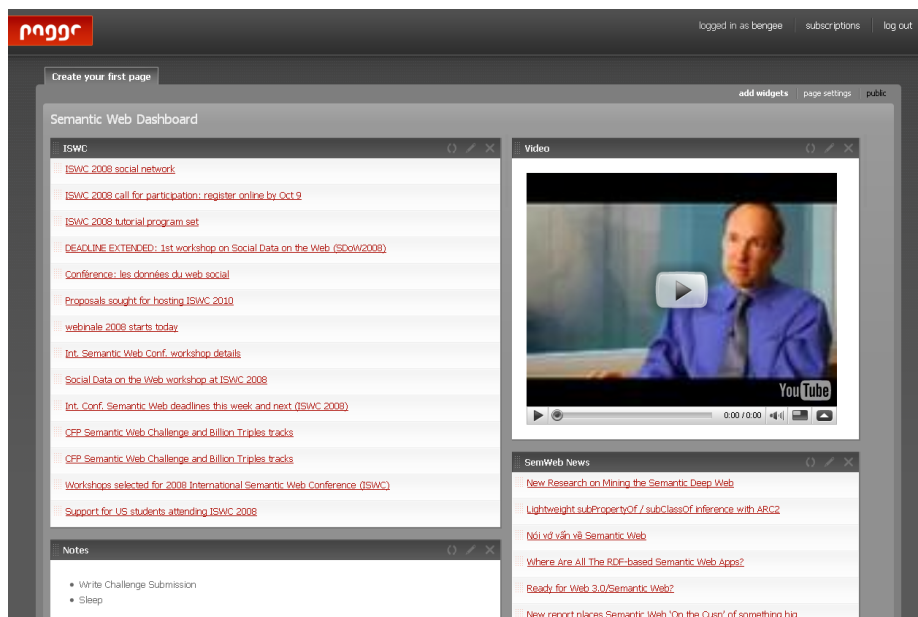
**Figure 3.** paggr can trigger pre-defined operations when two resources are connected by the user (still under development).

## 2.4 Data exchange

paggr supports several RDF formats (N-Triples, Turtle, RDF/XML) as well as in-HTML serializations such as RDFa, eRDF, or microformats. The main news feed formats (RSS2, Atom) can be converted to RDF, too. Through the use of a scripting-enabled SPARQL engine, paggr can integrate data from public SPARQL endpoints, e.g. company information from DBpedia and CrunchBase, or geo information from geonames. Likewise, each local portal is maintained in RDF and can be accessed through a SPARQL endpoint by the portal owner. Semantic markup can be embedded in the widget results, so that more direct re-use can be enabled.

## 3 Outlook

paggr is an experiment to hide RDF from end users, but still allow developers to easily extend the application with more powerful features, and also to let everyone benefit from the work done by widget developers and data providers. The current focus is on removing bugs and bringing true interaction to the widgets.



**Figure 4.** Example portal with news feeds and custom markup widgets.