Managing contacts, events, and locations with KOffice's new RDF support

Hidden Meaning

The KOffice suite lets you use the RDF capability built into the ODF standard to save, manage, and transmit semantic information. *By Ben Martin*

AUTHOR

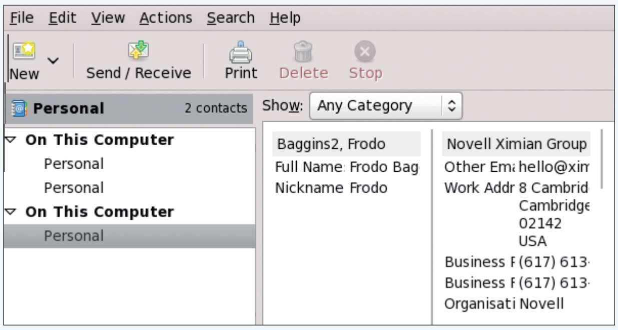
Ben Martin has been working on filesystems for more than 10 years. He completed his doctorate and now offers consulting services focused on the libferris suite, filesystems, and Qt/C++ development.

The Resource Description Framework (RDF) [1] is a central feature in the spectrum of technologies at the heart of the quest for what is known as the *semantic web*. The goal of the semantic web is to add meaning to electronic information such as websites and office documents, so that a term referenced through the text is more than just text. For instance, the program would know that the string "Fred" is not just a permutation of four ASCII characters but refers to a specific person named Fred who has a specific address and phone number.

Of course, many office suites provide an interface with some kind of personal information management system; however, as you will learn in this article, RDF supports a deeper and more automatic level of integration that keeps the meta-information in easy reach and allows it to travel effortlessly with the text. Also, the universal nature of RDF means that other programs supporting RDF will know how to interpret this semantic information. The semantic knowledge becomes part of the document, and later programs can apply that information in ways that might not even be envisioned at the time the data is saved.

The Open Document Format (ODF) [2] provides a means for integrating RDF with office documents, and developers are slowly implementing RDF features with the major open source office suites. In early 2010, the KOffice suite picked up RDF support, letting you define contacts, events, and locations. This means KOffice can now know for certain that "Fred" means a particular contact, that an event starts in a specific time zone, and that a meeting will occur at an exact location.

Although many office programs can operate on individual information formats like iCalendar and vCard, each one of these formats has a specific way to encode information, and generally programs must be written to process these files.



Show: Any Category

Baggins2, Frodo
Full Name: Frodo Bag
Nickname Frodo

Novell Ximian Group
Other Em: hello@xir
Work Addr 8 Cambrid
Cambridge
02142
USA
Business F (617) 613-
Business F (617) 613-
Organisati Novell

Figure 1: A contact for Frodo added to the Evolution application.

With RDF, an application like KOffice can load and save such information without needing to understand everything about it.

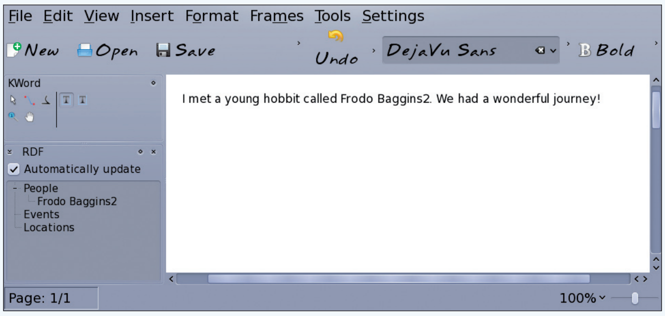
When you open that file in an RDF-aware office suite, you should be able to show the location on a map of your choice or send the contact and event information to other desktop applications or your mobile phone. Gone are the days of re-keyed information and sloppy cut-and-paste operations.

OpenOffice has also recently gained RDF support, and I am hoping to add support for RDF AbiWord this year. This gradual transition to RDF-enabled applications could eventually change the way users create and use office documents.

Getting into It …

The easiest way to get RDF into your KOffice document is open a mail program like Evolution (Figure 1) and drag a contact or event into KOffice. Currently, you'll have to add the RDF side panel to KOffice and drop the contact or event onto it. In the future, you'll be able to drop contacts or events directly into the main document region. When you drop a contact into KOffice, you should then see something in the document text at the current cursor location. In Figure 2, I have dropped Frodo into the document, and the RDF side panel knows that "Frodo Baggins2" refers to a person. The side panel shows the RDF corresponding to the text under the cursor.

If you right-click on *Frodo* in the side panel, you can import Frodo into your desktop address book, export him as a vCard, select the whole text that relates to him in the document (i.e., *Frodo Baggins2*) edit his details, and add other options.

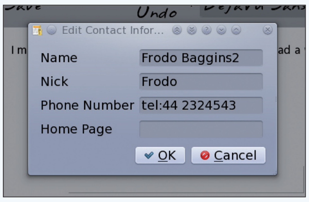


I met a young hobbit called Frodo Baggins2. We had a wonderful journey!

Figure 2: Dragging Frodo into the document shows his name, but as you can see in the RDF side panel, KOffice also knows this is a person.

When you edit the contact, you'll see something similar to Figure 3. This dialog will eventually expand to allow more in-depth editing of the contact information stored in RDF inside your documents. Once you have entered the contact information, you are using RDF. If you send the ODF file to another user, the user can import or export Frodo through vCard onto a mobile phone. A single file contains both a human-readable document and semantics that a computer can understand.

In addition to dragging information into a document, you can insert a new contact (Figure 4), which brings up the same dialog used to edit a contact (Figure 3), allowing you to insert a fresh RDF contact directly from KOffice. As you can see in the menu, you can also insert a reference. This way, you can refer to Frodo 10 times throughout your docu-



Name: Frodo Baggins2
Nick: Frodo
Phone Number: tel:44 2324543
Home Page:

Figure 3: Editing the Frodo contact within KOffice.

UNDERSTANDING RDF

Covering RDF in any detail would require many articles, but a bird's eye view will help explain how the RDF format adds semantic information to documents. An RDF object has three parts – a *subject*, a *predicate*, and an *object* – that form what is known as a *triple*. The subject specifies the thing for which you are defining the semantic record. The predicate specifies a relationship between the subject and the object. The object can refer to another subject or it can be a literal value. If the object is a literal, it can also have a type, like *string* or *date*. If you say, "Frodo has-phone-number 12312343," *Frodo* is the subject, *12312343* the object, and *has-phone-number* the predicate. To get a close look at raw RDF, use the *File | Document Information* menu option and select *Rdf*.

Predicates are often specified with URIs. Instead of just naming *has-phone-number* as a predicate (as in the preceding example), your RDF could point to a universal definition of the predicate *has-phone-number* at *http://my.example.com/contacts/has-phone-number* so the program would know exactly what "has-phone-number" means. Like XML, RDF can have namespaces, and in this case, *foaf:name* will actually resolve to the URI *http://xmlns.com/foaf/0.1/name*. The *foaf* portion of the URI refers to the Friend of a Friend project [6], which provides a standard system for machine-readable definitions of people and relationships. FOAF URIs are a well-known way to encode contact information in RDFs so that many programs can access it.

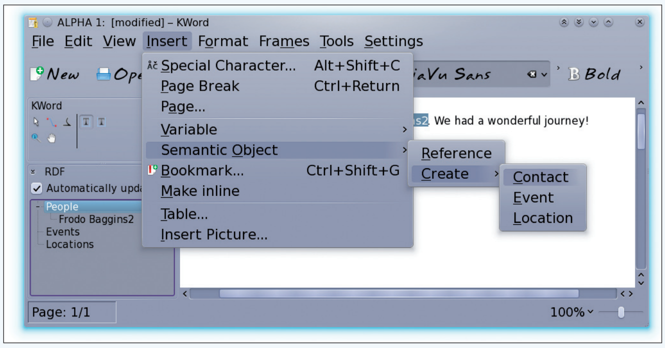
As you might have guessed, the *Namespaces* tab of the document information window will show you the mapping that relates a shortcut like *foaf* to a URI prefix like *http://xmlns.com/foaf/0.1/name*. Although many common namespaces come with KOffice, you can add your own, which are saved in RDF within your ODF file.

Whereas the RDF side panel shows you the objects that relate to where the document cursor is currently located, the *Semantic* tab shows you all the semantic objects for the entire document. This way, you don't have to search for Frodo in the text to export him as a vCard.

SPARQL is to RDF what SQL is to relational databases. The *SPARQL* tab lets you run SPARQL queries on the RDF contained in the document. The *Stylesheets* tab lets you apply a default semantic stylesheet for each type of RDF object in the document.

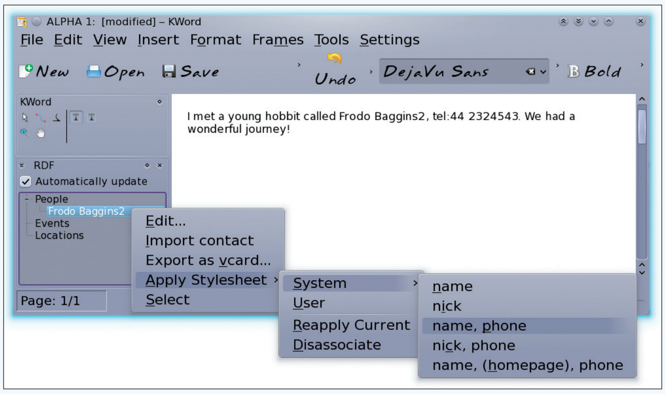
To add your own RDF stylesheet, use the *Format | Semantic Stylesheets* menu option (Figure 6). The bottom right shows which variables are available for the current RDF object type. To add a variable to the stylesheet definition shown (top left), click on it. Anything *%INSIDE%* percent signs will be replaced with the value from the RDF object where a stylesheet is applied.

User-defined semantic stylesheets are saved in RDF along with the office document. Therefore, one very powerful aspect of RDF is that you can save information about RDF in the same RDF store. For example, I could save information about the RDF stylesheets into the RDF store.



We had a wonderful journey!

Figure 4: Creating a new RDF object.



I met a young hobbit called Frodo Baggins2, tel:44 2324543. We had a wonderful journey!

Figure 5: Applying a system stylesheet to determine how this reference to Frodo will be formatted in the document text.

ment without having to duplicate the details. If you change Frodo's phone number through the RDF side panel, any place that refers to him will also know his new details.

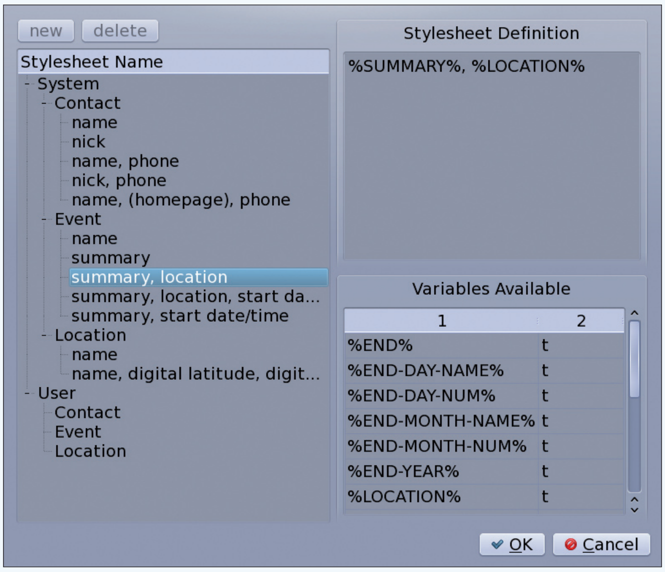
Stylesheets

The ability to refer to the same contact or event multiple times in the document is good, but you might also want someone who reads the document to see more than just Frodo's name in the text. For example, you might want to display his homepage or phone number, although you don't want to display it every time you refer to Frodo. Perhaps in the first reference, his phone number can appear, but in the letterhead you might also want to see his address. This is exactly what I created RDF stylesheets to do. When you right-click an object in the RDF side panel, you can apply either a system or user stylesheet to format how that object appears in the document text. In Figure 5, I have applied the *name, phone* system stylesheet so this reference to Frodo includes his phone number.

Portability

If you select all or part of a contact, event, or location that has an RDF associated with it and copy the text, the RDF goes with it. In fact, if you copy an event or contact from one office document directly into another document, the semantics are copied over too. This portability is possible because the text that is copied between documents is a full ODF file. Note, though, that the file is not saved to disk, so using ODF instead of just copying plain text is not a huge overhead.

One great upside to this is that you should also be able to copy and paste RDF between KOffice, AbiWord, and OpenOffice documents through the regular copy-and-paste interface.



%SUMMARY%, %LOCATION%

1	2
%END%	t
%END-DAY-NAME%	t
%END-DAY-NUM%	t
%END-MONTH-NAME%	t
%END-MONTH-NUM%	t
%END-YEAR%	t
%LOCATION%	t

Figure 6: View system stylesheets and add user-defined stylesheets.

Other toolkits that understand ODF can also get at the RDF inside the pasted text. For example, if you paste text from KOffice into KOrganizer, all the contacts from the RDF will automatically added to your address book.

Future Improvements

In the future, I hope to make the user interface expose the links between RDF entries. For example, you could right-click on Sam and see who he does gardening for. Then you would be able to navigate in the document to Frodo from a reference to Sam. Also, you should be able to link websites to events or jobs. This feature would let you know, for example, which five URLs are the most relevant to a particular invoice.

Wrap Up

Although the ODF specification provides strong support for RDF, RDF support in office suites has only just started hitting the free and open source desktop. KOffice and OpenOffice both have it now, and AbiWord should have it later this year.

Using desktop metaphors like drag and drop, you can easily add RDF objects to your documents and reference the same RDF objects multiple times, thus saving time and providing up-to-date information on contacts, events, and locations at a glance. ■■■

INFO

[1] Resource Description Framework (RDF): *http://en.wikipedia.org/wiki/Resource_Description_Framework*

[2] Open Document Format (ODF): *http://en.wikipedia.org/wiki/OpenDocument*

[3] KOffice and RDF: Say it with Style …: *http://monkeyiq.blogspot.com/2010/01/koffice-and-rdf-say-it-with-style.html*

[4] KOffice & RDF: Who, What, When, Where?: *http://monkeyiq.blogspot.com/2010/01/koffice-rdf-who-what-when-where.html*

[5] Office documents that mean something?: *http://monkeyiq.blogspot.com/2009/12/office-documents-that-mean-something.html*

[6] Friend of a Friend Project: *http://www.foaf-project.org/*