

# NEPOMUK



## Introduction

Human-kind is one of the few tool-using species on this planet. We have created replacements for many parts of our own body; cars instead of our legs, wrenches instead of our hands...

In 1943, the World's first digital replacement for the Brain appeared in Bletchley Park, England. Colossus, the first digital computer. This machine instigated the rise of the computer as we know it. But one of it's Creators, Alan Turing, did not also cause the rise of the computer, but also instigated the field of Artificial Intelligence through his Turing Test.

Since then, both computers and AI have moved on, becoming far more popular AND more powerful then anyone could of imagined. Like Linux, Artificial Intelligence infiltrated peoples lives. It could be in your router, your washing machine...even your mobile phone, naturally it's all over the internet. News sites provide me with the news I like because it knows what I like. New musician's are recommend to me, my friends and I are automatically identified and geo-tagged in photos...this isn't the crazy kind of Artificial Intelligence where machines go about killing everybody. This is about giving your computer the power of reasoning!

The purpose of this document is to introduce you, the end user (and you, the interested developer) to NEPOMUK, one of the greatest technological innovations to ever hit the desktop (hey, don't believe me? Read this and find out why I've come to this conclusion)

***First off, a warning; Below I've created a few mockups (though some, such as the scribo-shell you'll see below are real) and these are mockups ONLY. I will be very interested in helping anyone build it and improve it (my expertise lies in Artificial Intelligence and Man-Machine Interfaces)***

## Introducing NEPOMUK

NEPOMUK (**Networked Environment for Personalized, Ontology-based Management of Unified Knowledge**) is a Semantic Desktop tool that uses an 'Ontology'. Ontologies are computer readable Encyclopaedias and thesauruses. Their purpose is simple; they give our machine brethren the ability to think.

The ability to *understand*.

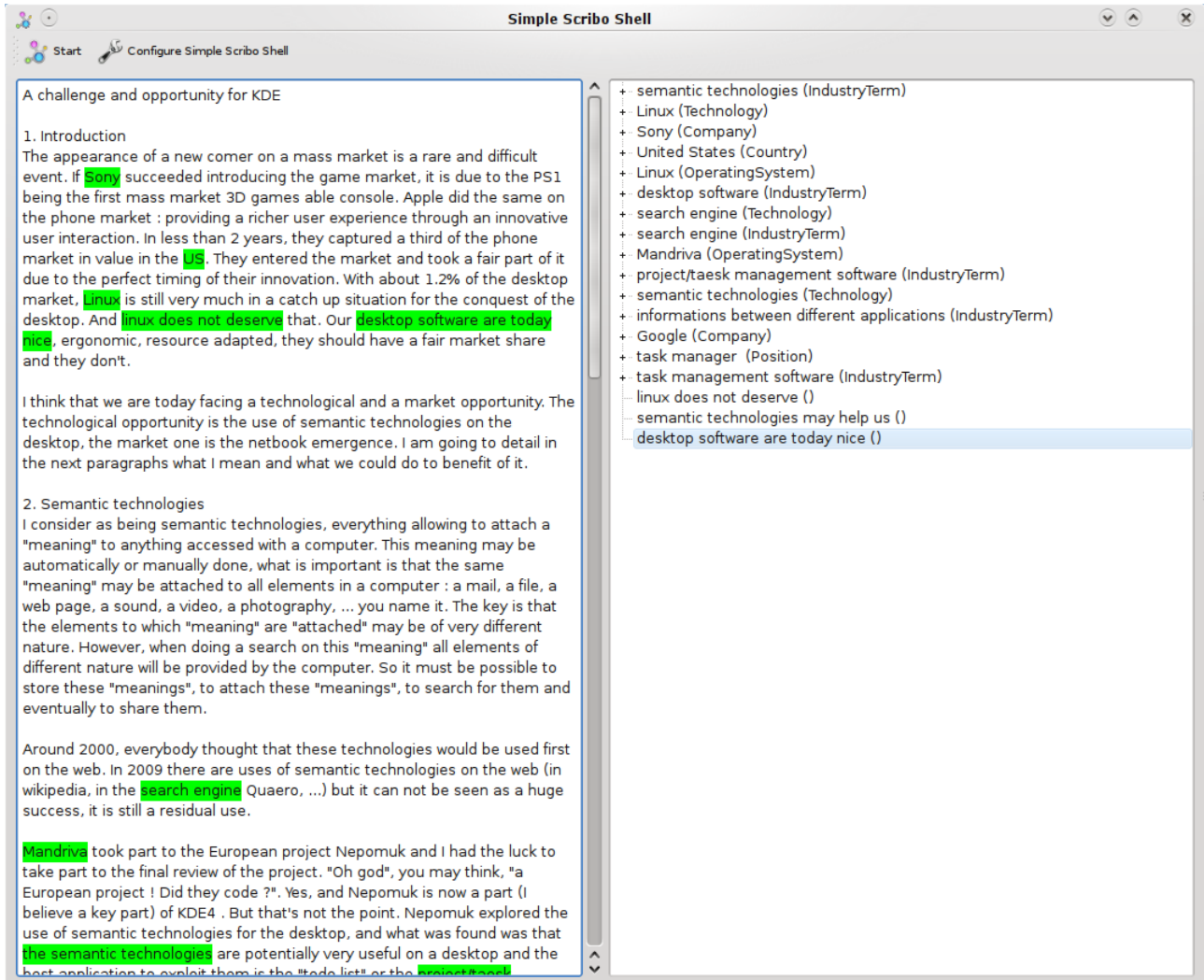
Think about that for a moment.

What if your computer could understand what you are? Where you live? What you like to do? What if it could understand what yoTo understand how it knows the subject, lets look atthe next bit firstur saying in that email? In that conversation? What if it understood what a document was about? Or that email?

NEPOMUK is not an end to itself, rather it's a means to an end. The NEPOMUK Project builds the foundation to make

your computer *intelligent*.

NEPOMUK itself is the name for a collection of tools, and as it stands now, the system has limited ability. When it is complete, however, it will be VERY powerful and will change the entire way you work with your computer. Speaking of tools, let us meet one of them now; Scribo.



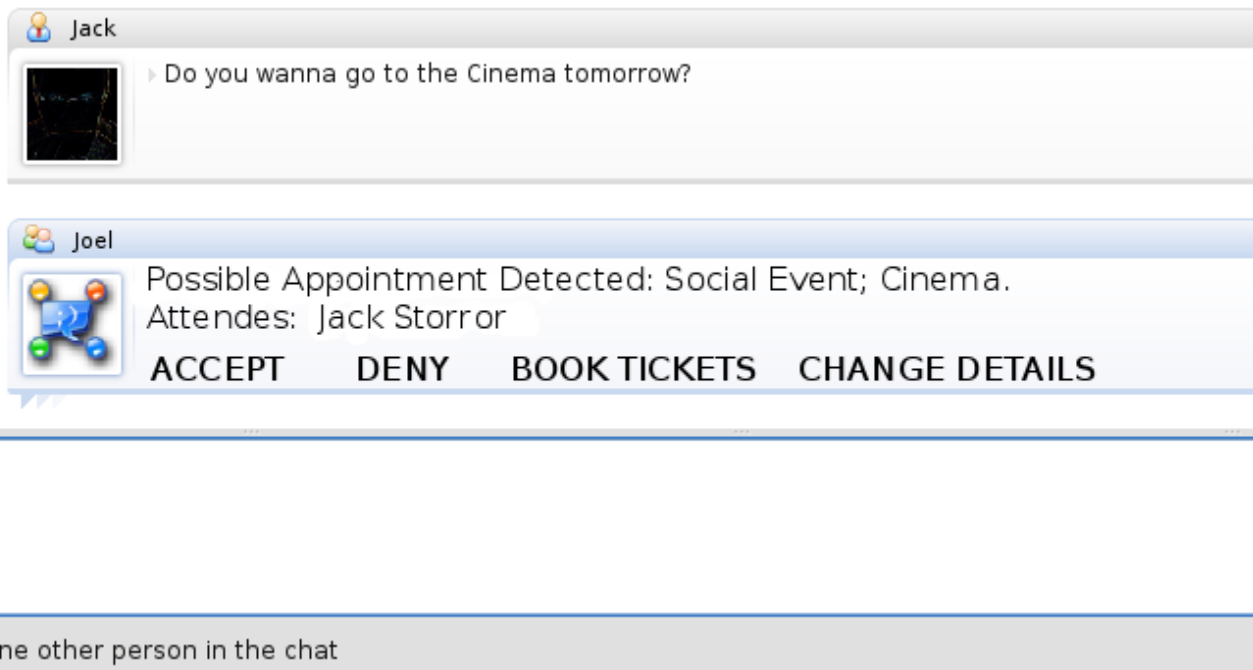
*A simple Scribo shell, showing topics within a document, on the left is the document, the parts highlighted in green are what Scribo has recognized and the right window lists topics found, taken from the extracted highlighted keywords.*

Scribo analyses text. This is more important than you think. If your computer can understand what is being said...well...as of the time of writing, Scribo is still in the playground, but it could, say, watch over a conversation. Best if we do a few examples to show you just what this could do;

## Example Programs and functions

Here, we'll look at an extended NEPOMUK and Scribo system and just see what could be in our very near and very REAL future...we just need developers to do what they do best, and users to suggest improvements ;)

Joel and Jack are friends, Jack messages Joel and says "Hey buddy, you wanna go to the cinema?"



*A (rather rough) mockup of this system functioning in Kopete.*

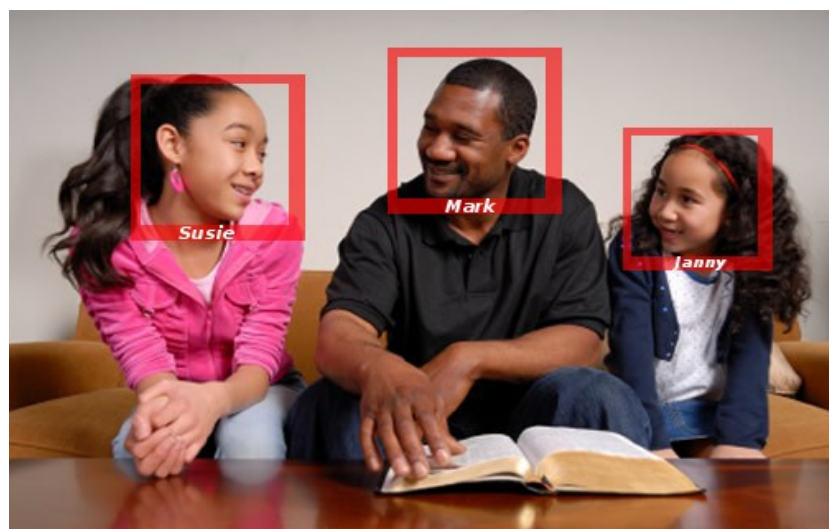
Clicking 'Accept' would insert it into the 'Appointments' and start up a plugin to book tickets. Clicking 'Deny' would, of course, do the opposite.

'Book Tickets' cause the plugin to activate before an appointment was set (which would be useful, in case there was going to be a possible conflict with already existing plans).

'Change Details' would, of course, change details about it in case you wanted to invite other people. It may even be possible to get this whole system to be autonomous and transparent to the user, should it be worked on enough!

*Aftermath; Here, Scribo was used to understand what was being said. NEPOMUK made the conceptual metadata available, such as this being a social event at a cinema and Cinema's normally have a set start time for the curtains to go up. Most Cinema companies post viewing and booking data online in a set format so development for reading and booking tickets online should not be a problem.*

Mark takes a lot of photos. Most cameras provide metadata about the camera used to take the picture, the time when the picture was taken, and other assorted metadata. A great deal of cameras can automatically detect faces and focus properly on them. What they can't do very well is provide metadata of who those people actually are.

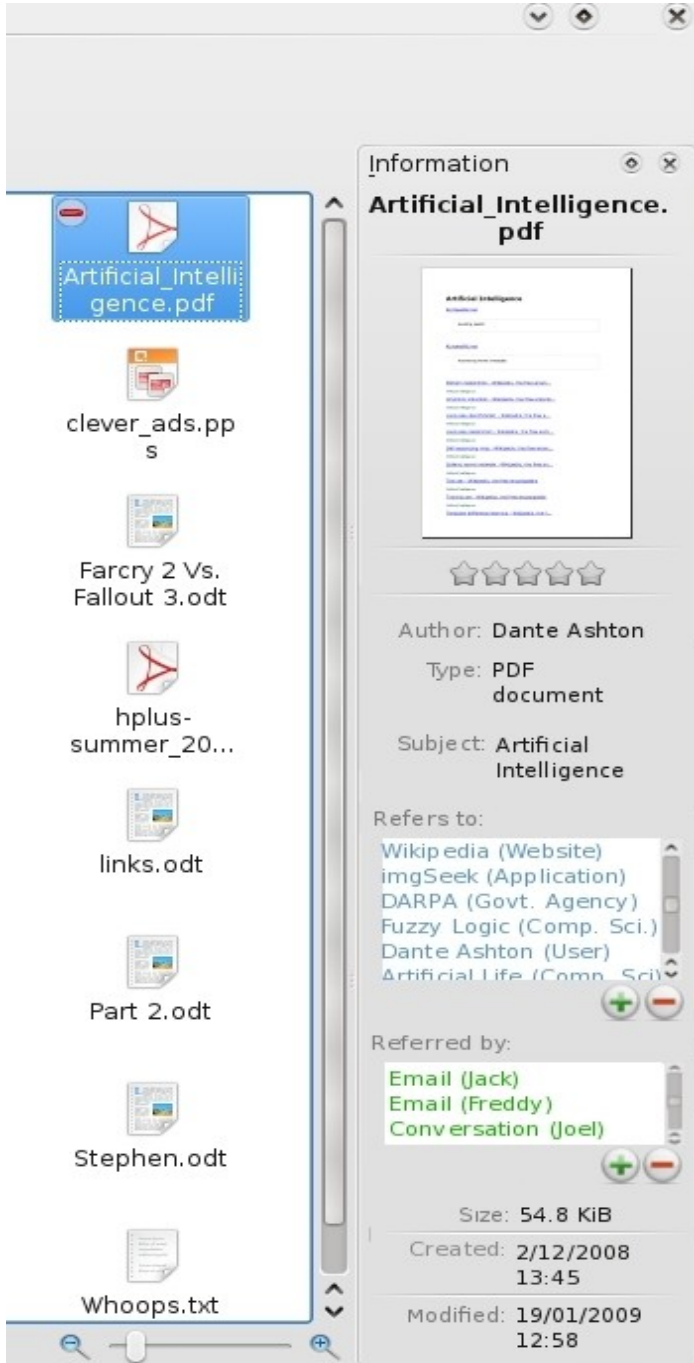


In this picture, a photo taken and placed on a computer (which then employed a face recognition system) and the face recognition system then added this meta-data to the photograph. NEPOMUK then read this data, and knows that Mark is one of the computer's users. It knows Janny is one of it's users as well. It knows Susie does not use it as there is no user account for her.

However, it knows her email address and mobile telephone number and can send this picture, or any other picture, at Mark's command, to her email account or mobile phone.

Let's take a more advanced use case here, lets imagine Mark's a professional photographer, but for one reason or

another, doesn't have access to his digital portfolio right when he needs it. He could thus send a command to his computer through email (raising the interesting question of providing the computer itself with an email address to allow it to communicate with it's users at a distance) which gives his computer the command to email all photos containing himself to his account (where he can access it via another computer) or send them to his mobile and show his potential clients his work.



Now, here's an interesting picture. This is an *enhanced* information pane in Dolphin. Let's work through it, starting from the top.

First, item name. Then thumbnail, then visual rating (stars).

Now it gets interesting;

**Author;**

Many word processors include the name of the author, this is either something set at install/first boot up, or is taken from the username of the logged in user. In many other cases (such as scanned books, for instance) the Author's name is easily located and I can pretty much guarantee writing a script to extract it would be an easy task. That particular bit of metadata is very useful to those who have a lot of documents to sort through and might want, for instance a list of all documents written by one particular person.

**Type:**

Again, something already here, but I might want to combine a search and say "all PDF's by Dante Ashton"

**Subject:**

..now, this is also VERY interesting, and surprisingly simple to do. To understand how it knows the subject, lets look at the next bit first;

**Refers to:** Scribo has gone through the document and found it relates to many things, it links to Wikipedia many times (hence why it is at the top of that list in the picture) but the overall amount of topics (Artificial Life, Fuzzy Logic etc.) is known collectively as 'Artificial Intelligence'. Thus that is it's subject. Another case would be if the document referred to specific musicians that only do Heavy Metal music, in that case the topic

would be: Musicians (Heavy Metal)

Clicking on one of those items would bring up a list of items who also relate to that topic.

This is how 'Subject' is found, by looking at the overall topics within the document. This information not only helps the user, but can also help the system solve ambiguity issues (Subway: fast food store, or Subway, underground train? If the document talks about methods of transport, it's probably the latter case)

**Referred By:**

Would be a list of conversations, emails, etc. that involved this document, I could of mentioned it or sent it in an email. Made it part of a to-do. Transferred it to a USB drive...  
Clicking on one of those items could bring up the reference in full: Example, Kopete's history.

**Created:**

Most word processors not only provide the author's name, virtually all of them provide the date of creation of the file.

---

Now for a fun (and less graphically intensive) few use cases, each use case is divided up into two; one where NEPOMUK is not present, the other is where it IS present.

**Location-awareness**

*Here we see geo-location being put to use.*

**Without:**

*Steven uses his laptop at work and at home. He is frequently on the move. His data, however, is not. He has had to modify his system so that he can accommodate both work related tools and files, and tools and files for home use. His applications are arranged on what he most uses, but it is lacking context; Because he has made so many modifications to the whole system to accommodate both recreational duties and business ones, he has a very messy desktop where the search tool is his best friend for finding data.*

**With:**

*Steven uses his laptop at work and at home. He is frequently on the move. His data, however, also moves with him. The computer has identified, by a geolocation technology, where it is. NEPOMUK knows that at a certain places, his applications change, as do his use of contacts. So, when at work, applications he uses the most there are put within easy reach, Amarok is deactivated (due to it's non-usage record at work) and OpenOffice is preloaded instead. NEPOMUK knows he will be contacting Dave, a workmate, more often at work then he will be contacting his Son. The system has also stored configuration data for other applications, like the wireless connection manager and in-company IRC and email network, and applications are adjusted accordingly to provide a seamless, trouble free user experience. This is a 'smart' system, after all.*

---

**Content-awareness**

*NEPOMUK provides content relevant to the task at hand.*

**Without:**

*George has had a email from Fred. Both of them are lawyers in the same firm and Fred has contacted George about a client, one Mrs. Farren. Fred wants to know Mrs. Farren's full address, date of birth, etc..  
George has a lot of people on file. George uses a search tool...and it comes back with 3000 results for partial matches...*

**With:**

*George has had a email from Fred. Both of them are lawyers in the same firm and Fred has contacted George about a client, one Mrs. Farren. George has a lot of people on file. NEPOMUK displays related files in a window next to the email. The file Fred is after contains a full account of Mrs. Farren, her name, age, location, NEPOMUK finds the file and offers a contextual action, "Send to Fred"....*

---

## A Vision Of the Future.

Many technologies were developed and thought of, but then dumped, simply because this system did not exist. Now, it exists and is being worked on. The above cases were simple examples of NEPOMUK in about...2 years, or much less if developers and users alike work on it.

Below are other use cases of the near future. I cannot stress enough that this will NOT happen if developers and users are not interested. Get involved and help make something brilliant. :-)

You like a particular musician. NEPOMUK is aware that this artist is on tour and is coming to your town. It alerts you and pro-offers you a link to the website to allow you to buy tickets. You do so. NEPOMUK knows the date and adds it to your calendar, as well as figuring out a route to the concert. On the day, it'll warn you of problems, like bad weather.

You wish to book a holiday to Cyprus. Through your computer, you have booked the flight and the hotel. As well as adding it to your calendar, and figuring out a route starting from your house and ending at the hotel; it also informs you of delays of your flight and NEPOMUK has already made a reservation for you at the hotel through email.

You are looking for a particular product, perhaps a console game, music CD or car insurance. Or perhaps all of them. NEPOMUK searches with you; scanning sites and giving you prices. It will, of course, select the lowest price first. (This would simply require NEPOMUK scanning websites, rather than websites advertising. Notice the difference. :-)

The above examples are just a taste. NEPOMUK opens the door to many such possibilities.

## Conclusion and How to Help

NEPOMUK is a system entirely dedicated to not only changing the way you operate with your computer (and, more importantly, opening the doors to allow it to operate with *you*) it also opens up technologies that, before NEPOMUK came along, could not be built. It is the *very foundation* of all those computer systems in science fiction, your computer can know everything; everything about its applications, everything about your data and everything about you...As I hope you've learnt, it has MASSIVE potential, but like everything, it needs careful attention and work put into it to make it grow.

If you are a User and wish to help, you can provide mock-ups, thoughts, ideas and suggestions. Better still, you can help build interfaces with QT Designer. Or provide log entries if there is a problem.

If you are a Developer/programmer, and you are interested in NEPOMUK, please, feel free to step forward and assist with the code.

