

Allow KPlato project management features to include managing documents.

1. Introduction

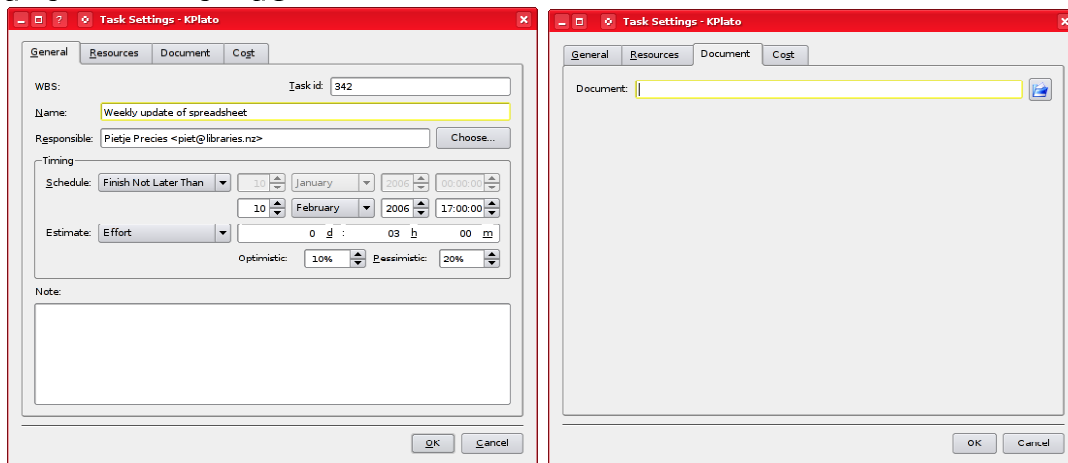
The project management software KPlato currently has little integration in KOffice. Integration of the project management features to extend the usefulness of all KOffice applications can be done with a small amount of KOffice wide features. This paper will outline the needed changes for a simple yet very powerful work-flow which allows a project manager using KPlato to manage document creation and updating tasks. To extend the work-flow to be group based the proposal also uses email notifications and the industry standard vcal fileformat to distribute the tasks as todo actions to the various individuals.

2. Creating a task creates a todo

KPlato uses tasks for all things that have to be done in a project. Example tasks can be "Buy paint" or "Paint house" for a house renovation project. For almost all projects one or more tasks will be dedicated to creation or updating as well as proof reading of office documents.

KPlato would get a new task type "Document Task", common subtypes (registered as names, so fully user adjustable) would be "create", "update", "review".

This type of task is special because it can have a document attached. This document would be used as a template for the "create" subtype or as startpoint for the other subtypes. Resources¹ attached to the task get to receive an email with a vcal entry and the original document copied from the KPlato task. Choosing the industry standard vcal fileformat allows automatic integration with Kontact.



New Task type

The person receiving the email will see an attached file. Opening the file from within KMail follows the convention to load the file from a temp directory in KDE3.5 (as registered in KDE). KWord and other KOffice applications should be adjusted to detect the source file coming from a temp directory and provide a "Save As..." dialog instead of saving changes back to the temp directory.

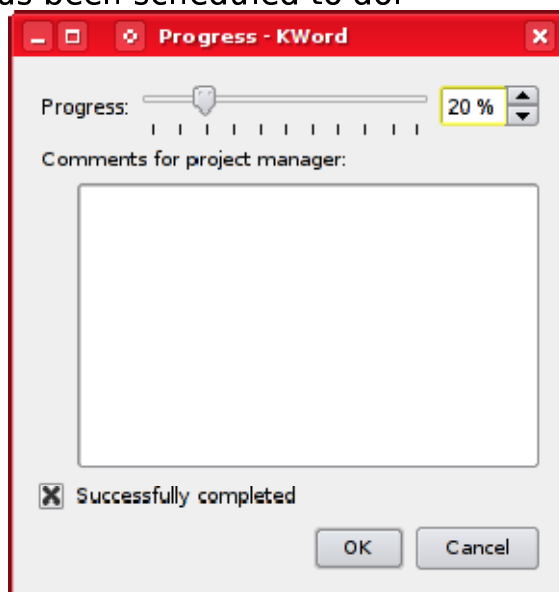
Alternatively ODF is extended to store a project management ID which

¹ A resource in KPlato is anything needed in the real world to do the task. This can be a computer, but also an employee.

indicates the TaskID and name as well as the project managers email address. This will have the effect that all files opened will be checked to have this project management ID embedded, and if it does it will be copied to a "projects directory" which is a directory under the \$KDEHOME hierarchy.

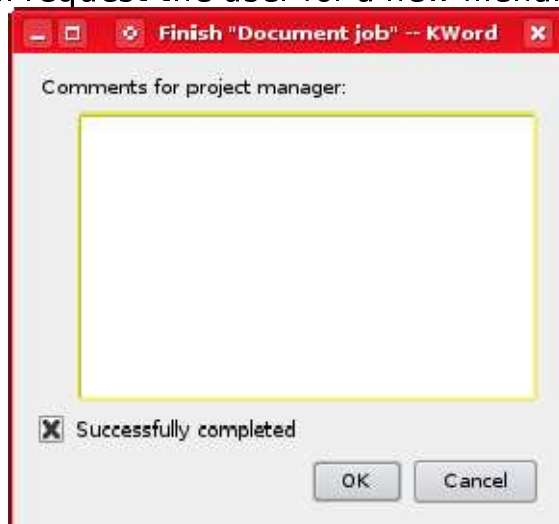
3. Finishing the edit

On closing the document that has such a project management ID, using File->Close or File->Quit, the KOffice application will ask the user if the document is finished, or what percentage of work has been done. If the document is completed a vcal-type "todo done" email is sent to the project manager, the address of which is embedded in the document. The document itself is attached to the email. For the user that typed the doc, this also finishes the task he has been scheduled to do.



Dialog on Quit or Close

The startup dialog will be extended to show a "Running Projects" main entry if there is at least one project in the "projects directory". The files in that directory are stored there by the KOffice application themselves somewhere in the \$KDEHOME hierarchy. Opening a project based file will make visible a new menu entry in the File menu called "Finish Document...". This menu entry will send the file and remove it from the projects todo queue. It will also mark the open document to have no name, and make it unmodified. This has the effect that pressing save will request the user for a new filename.



Additional ideas:

1. Make finishing a project leave behind some meta data in the project directory so the user can still open the file (read only) and sent another email if for some reason the first did not come through.
2. Instead of attaching the document to an email, allow a URL to be specified in the todo. It may be better to have a common directory in some corporate situations.

4. Reporting back to the project manager

The user that edited the document has sent his work back using a vcal-todo type email. If the project manager uses KMail the integration with KOrganizer is already there and it will result in the todo being marked done. To notify KPlato of the change there are 2 ways to go.

1. Use a hack. Simplest is to make a KPlato startup option so you can type 'kplato --finish myId foo.kpl' and to make KOrganizer execute that.
2. Integrate all KPlato tasks nicely by creating a new KOrganizer back-end (called a resource) based on a KPlato file.

I suggest using the last option since it is the more correct way that extends what technologies are already there. In this situation the resource would do the following things;

- Make each change in the KPlato project-file be propagated in the KOrganizer view.
- Make changes in the KOrganizer todo view be saved as changes in the tasks of the KPlato file.

5. Conclusion

With the work-flow introduced in this document it is possible for small users to full corporations to have document management fully integrated into the project management software, which would clearly make KOffice a big and powerful player in that sector.

This kind of integration has been requested by many corporate users and will make KOffice different enough to set it apart from the competition.