

Automatisez la production de vos documents dans Tiny ERP avec OpenOffice

Fabien Pinckaers

Bien qu'il existe de nombreux logiciels libres pour les infrastructures réseau et la bureautique, ces derniers se montrent quasiment absents du domaine des progiciels de gestion intégrés. Jusqu'il y a peu, les entreprises devaient encore se tourner vers des solutions propriétaires extrêmement coûteuses pour leur gestion avancée.

Ce premier novembre est sortie la version 3.1.1 de Tiny ERP, et avec elle un tout nouveau système de rapports intégrés à OpenOffice, que nous décrirons dans cet article.

Tiny ERP est un progiciel de gestion intégré libre pour petites et moyennes entreprises. Il intègre la plupart des processus d'affaire : ventes, gestion des stocks, logistique, comptabilité, gestion de projets, finance, services après ventes, chaîne de production, ...

Tiny ERP (Entreprise Ressource Planning) est caractérisé par une très forte flexibilité lui permettant de s'adapter aux besoins des différentes entreprises. Dans cet article, nous allons analyser le système permettant de créer des rapports sur mesure dans Tiny ERP.

Le moteur de génération de rapports de Tiny ERP s'appuie sur trois logiciels ;

- OpenOffice ; pour la création des templates des documents,
- La base de données objet de Tiny ERP pour l'extraction des données,
- Le langage Python pour les expressions spécifiques.

Dans cet article, nous allons adapter le format des factures au logo de notre entreprise test et nous allons créer un nouveau rapport qui affiche la liste des clients sélectionnés.

L'installation de Tiny ERP ainsi que les manuels utilisateurs et administrateurs sont disponibles sur le site: <http://tinyerp.org>.

Introduction à la génération des rapports

Il existe trois méthodes pour générer des rapports dans Tiny ERP;

- Les rapports OpenOffice
- Les rapports XML et XSL:RML

Sur l'auteur :

Fabien Pinckaers, 26 ans, Belgique. Directeur de la société Tiny.be, éditeur de logiciels libres. Principal développeur et fondateur de nombreux projets libres : Tiny ERP, TeCommerce, OpenReport, Tiny Report, Tiny Raytracing, Tiny Project, Tiny Accounting, Gérant du magasin Linux en ligne : <http://Openstuff.net>

Pour contacter l'auteur : fp@tiny.be

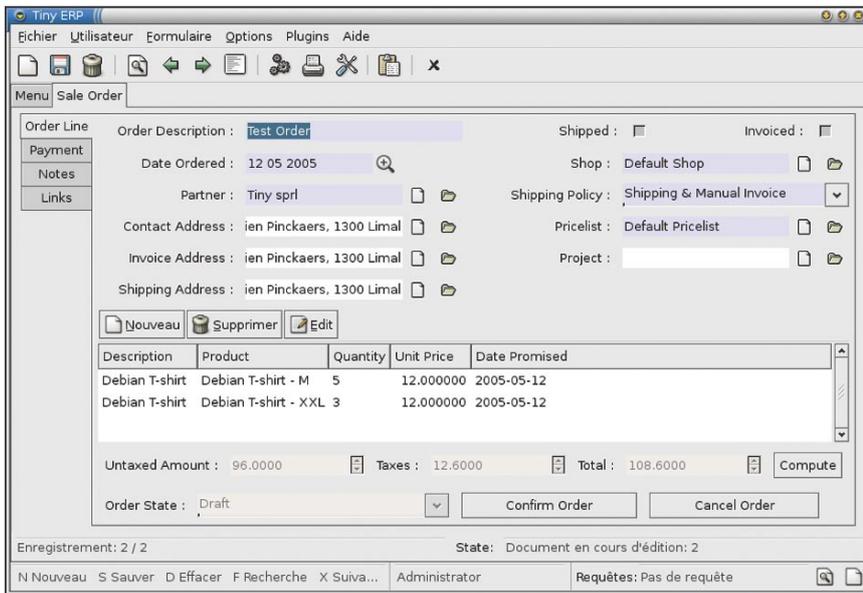


Figure 1. Le bon de commande dans Tiny ERP

- Les rapports colonnes créés directement dans le client Tiny ERP.

Nous nous intéresserons à la première méthode qui est à la fois la plus simple et la plus flexible.

Le processus de création d'un nouveau rapport dans Tiny ERP est le suivant :

1. Conception du rapport dans Open Office avec des expressions pour la base de données
2. Génération du fichier intermédiaire .RML
3. Inscription de ce fichier dans Tiny ERP

Principe

La production des documents dans Tiny ERP (factures, notes d'envoi, bons de commandes, ...) se fait grâce à des fichiers RML. RML est un format de fichier XML décrit par Reportlab [http://reportlab.com/docs/RML_UserGuide_1_0.pdf]

Le RML étant un dialecte du XML, ces fichiers sont plus simples à manipuler que directement des PDF. Ils sont ensuite convertis en PDF grâce au programme tiny_rml2pdf du projet <http://OpenReport.org>.

Ces fichiers RML décrivent la structure d'un document ainsi que les expressions et les champs de la base de données à y inclure. Exemple de fichier RML.

```
<!DOCTYPE document SYSTEM "rml_1_0.dtd">
<document filename="example_2.pdf">
<template>
  <pageTemplate id="main">
    <frame id="first" x1="2cm"
      y1="2cm" width="17cm"
      height="26cm"/>
  </pageTemplate>
</template>
<stylesheet/>
<story>
  <para>
    Les textes <b>RML</b> sont placés dans
    les tags "story" qui composent le corps du
    document.
  </para>
  <para>
    Les paragraphes sont placés entre les
    tags "para" et "/para".
  </para>
</story>
</document>
```

Pour l'incorporation des données provenant de la base de données de Tiny ERP dans le document, on y ajoute des expressions entre double crochets.

Ces expressions sont au format du langage de programmation Python et les variables seront remplacées par les valeurs de la base de données.

Seulement, écrire à la main ce genre de fichiers .RML n'est pas toujours simple. On utilise alors OpenOffice pour construire un template de document que l'on sauve en .SXW. Ce fichier .SXW est alors converti en .RML grâce au programme : tiny_sxw2rml.py, du projet <http://TinyReport.org>.

La chaîne complète de transformation du document est présentée sous la Figure 2. Le seul fichier à créer pour construire des nouveaux rapports dynamiques dans Tiny ERP est le fichier Open Office (.sxw) contenant les expressions entre double crochets.

Modification des factures

La première étape consiste à de construire le template du rapport dans Open Office.

Pour notre premier document, nous allons modifier un rapport existant dans Tiny ERP ; l'impression des factures.

Ouvrez le fichier 'invoice.sxw' qui se trouve sur le serveur Tiny ERP à l'adresse : server/bin/addons/account/report/invoice.sxw. Ce document est le template d'une facture Tiny ERP.

Pour notre premier essai, nous ne ferons que des modifications simples, sans

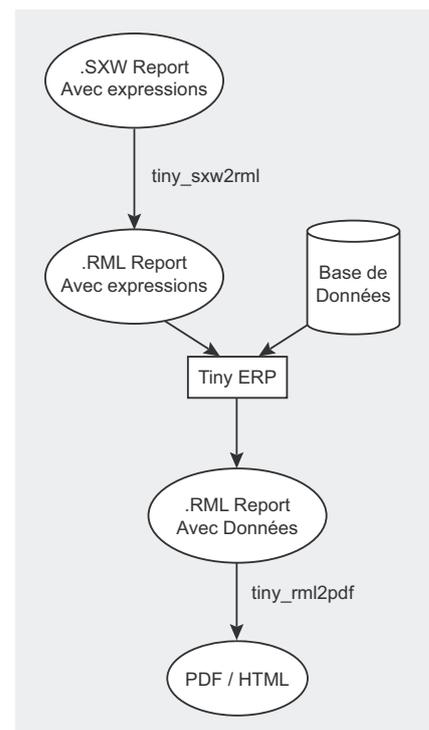


Figure 2. Le processus de transformation du document vers le PDF

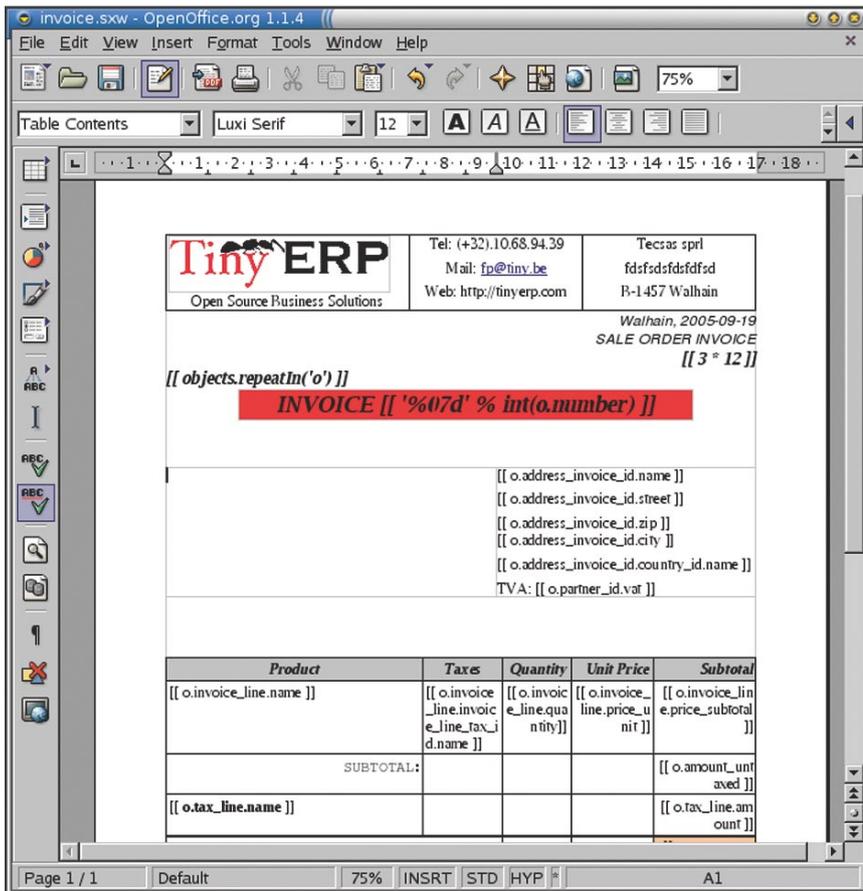


Figure 3. Le template de facture dans OpenOffice

toucher aux expressions entre `[[]]`. Par exemple, vous pouvez changer le logo, les coordonnées de votre entreprise, les couleurs de fonds, ...

Ensuite, il faut pouvoir importer le document généré dans Tiny ERP. Pour ce faire nous utiliserons le programme `tiny_sxw2rml`. Vous le trouverez à l'adresse : <http://tinyreport.org>, section download.

La facture OpenOffice contenant les expressions.

Convertissez alors votre fichier :

```
./tiny_sxw2rml.py path_to/invoice.sxw
>path_to/invoice.rml
```

Lorsque vous imprimerez des factures dans Tiny ERP, celles-ci seront adaptées aux modifications apportées.

Les expressions

Les expressions entre double crochets `[[...]]`, sont des expressions au format du langage Python qui font appel à la base de données. Les champs de la base de données sont accédés comme de simples variables. Les relations d'une table à l'autre sont accessibles avec un point ".".

Exemple, supposons que l'on fait un document pour une facture, possédant les champs suivants (décrits au format SQL):

- `[[3*2+4]]` sera remplacé par 10.
- `[[name]]` sera remplacé par le champ 'name' de la facture

- `[[len(name)]]` sera remplacé par la taille du champ name de la facture (len est une fonction Python)
- `[[o.partner_id.name]]` sera remplacé par le champ 'name' du partenaire référencé par le champ partner_id de la facture.
- `[[o.line_ids.name]]` la ligne du tableau dans lequel se trouve cette expression sera dupliquée pour chaque ligne de facture et dans cette colonne, la valeur du champ « name » de chaque ligne sera ajoutée. La ligne est dupliquée car le champ line_ids de la facture est une relation de cardinalité one2many.

Ainsi, lorsque vous utilisez les champs one2many de Tiny ERP (par exemple, une facture qui possède plusieurs lignes), le moteur de Tiny ERP va dupliquer la partie du rapport dans lequel se trouve cette expression. L'effet dépend de l'emplacement de cette expression :

- dans un tableau: duplique la ligne,
- dans une section OpenOffice : duplique la section,
- ni dans un tableau, ni dans une section : duplique la page en cous.

La seule variable accessible au départ est : "objects". Elle contient la liste des objets sélectionnés dans Tiny ERP lorsque l'utilisateur a cliqué sur l'icône "Print".

Vous pouvez créer d'autres variables grâce à la méthode "repeatln(variable_

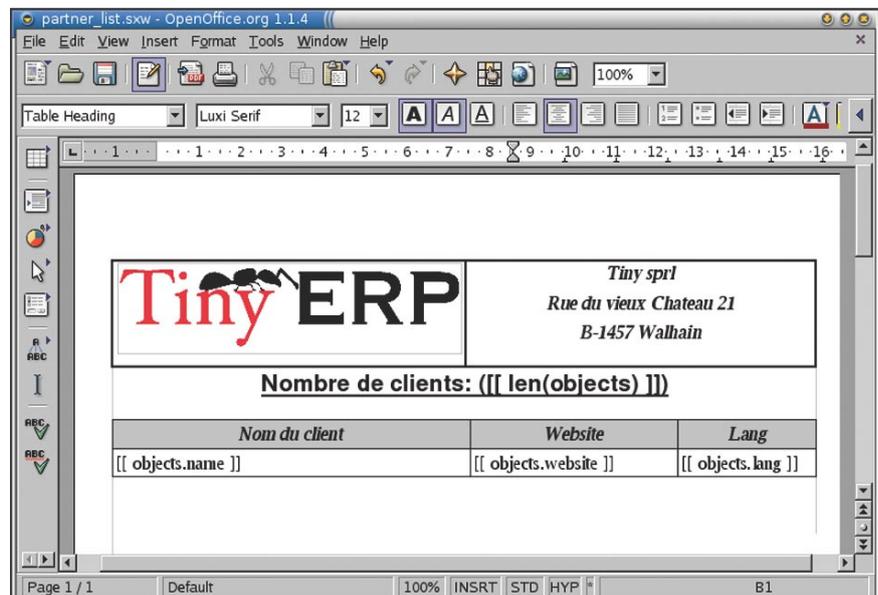


Figure 4. Le template de listing client dans OpenOffice



Figure 5. Le listing produit en PDF depuis Tiny ERP

name)"). Ainsi, l'expression : "objects.repeatln('o')\" dupliquera la ligne ou la section en cours pour chaque objet et vous pourrez utiliser la variable "o" pour accéder à chaque objet indépendamment.

Création d'un nouveau rapport

Dans les précédents rapports, nous avons modifié un rapport existant dans Tiny ERP. Nous allons maintenant créer notre propre rapport. Pour ce faire, les étapes sont similaires, si ce n'est qu'il faut également inscrire le rapport dans le serveur Tiny ERP.

Créez le fichier suivant dans OpenOffice : et l'on sauve ce fichier dans le répertoire du serveur Tiny ERP : "*server/addons/base/res/partner/report/partner_list.sxw*".

C'est un rapport sur une liste de clients (appelés partenaires) qui affiche un tableau avec le site Internet et la langue de chaque client.

Il faut ensuite transformer ce rapport en RML à l'aide de *tiny_sxw2rml.py* : *tiny_sxw2rml path_to/partner_list.sxw, path_to/partner_list.rml* (remplacez *path_to* par le chemin de votre installation Tiny ERP).



Figure 6. L'édition des données de l'utilisateur

La dernière étape est d'inscrire ce rapport dans un des modules de Tiny ERP, dans ce cas-ci : "base". Chaque module possède une liste de fichiers .XML qui seront chargés à la mise à jour du programme, cette liste peut être obtenue dans le fichier : *tinyerp/server/addons/base/__terp__.py*.

Dans notre exemple, nous allons inscrire le rapport dans le fichier : *server/addons/base/res/partner/partner_report.xml*.

Et nous ajoutons l'inscription suivante :

```
<report
  string="Partner List Test"
  model="res.partner"
  name="partner.list"
```

```
rml="base/res/partner/report/
partner_list.rml" />
```

Où :

- string: désigne l'intitulé du rapport (en Anglais, sera traduit en français après par le système de traduction de Tiny ERP).
- model : désigne l'objet sur lequel se fait le rapport
- name : nom du rapport, au choix.
- rml : chemin du fichier RML que l'on vient de générer.

Pour que Tiny ERP prenne en compte ce nouveau rapport, il faut l'exécuter avec l'argument *--update=base* : *tinyerp_server.py --update=base*.

Ensuite, dans le client Tiny ERP, on peut faire une sélection de partenaires et l'imprimer. Cela donne le PDF représenté sous la Figure 5.

Un moteur de rapport indépendant

L'équipe de développement de Tiny ERP travaille sur un projet permettant d'utiliser ce système de rapport indépendamment de Tiny ERP.

Ce projet est toujours en cours de développement mais vous pouvez suivre son évolution sur <http://tinyreport.org>.

Cela permettra de générer des fichiers PDF dynamiquement pour vos applications et vos sites webs sans devoir écrire une seule ligne de programmation. ■

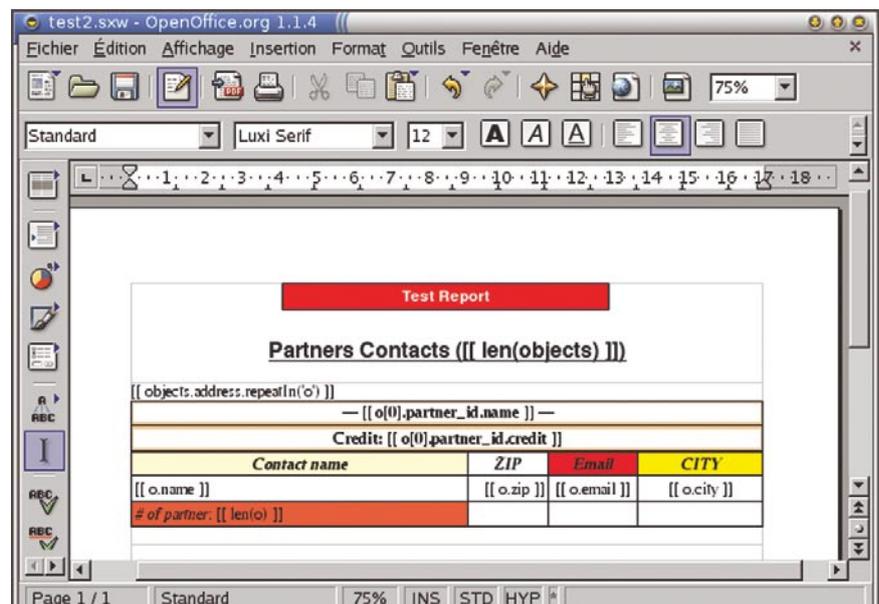


Figure 7. Le rapport du test dans OpenOffice