

ShareLatex, éditeur Latex collaboratif auto-hebergé : retour d'expérience au LE2I

Arnaud DA COSTA (IE CNRS), Benoit DARTIES (MCF 27)

LE2I, UMR CNRS/uB 6306

26 juin 2014

Retour sur l'installation et l'utilisation ShareLatex : un éditeur Latex collaboratif auto-hebergé LE2I

1 Introduction

2 Installation et configuration sous Debian 7.0

- Installation de TexLive
- Installation de NodeJS/NPM
- Mis en place de ShareLatex

3 Configuration locale

4 Retour d'expérience

LE2I

Laboratoire Electronique, Informatique et Image (LE2I). UMR 6306. départements INSIS (principal) et INS2I (secondaire)

- UMR 6306 CNRS / uB, 100 membres permanents
- Multi-sites : Dijon, Le Creusot, Auxerre et Chalon sur Saone

Organisation Scientifique

Directeur		
Fabrice Meriaudeau		
Informatique	Electronique	Vision
Olivier Togni	Stéphane Binčík	David Fofi
Image	SNL	SINC
Christian Gentil	Patrick Marquié	Franck Marzanni
AlgoComb	ARCHI	IMAC
Vincent Vajnovszki	Dominique Grinjac	François Brunotte
SISI		MOTI
Nadine Cullot		Olivier Laligant
CheckSem		
Christophe Nicolle		
RAHA		
Albert Dipanda		

Latex

Wikipedia : \LaTeX est un langage et un système de composition de documents créé par Leslie Lamport en 1983

\LaTeX permet de se concentrer sur la structure logique du document et son contenu

Très utilisé pour la documentation de documents en Recherche.

Notre découverte de ShareLatex

Annonce sur Linuxfr.org

- Plateforme en ligne
- Libération du code de ShareLatex.com 02/2014
- Essai d'installation (Debian) "juste pour voir" : pas de blocage
- Proposition d'essai en situation

Prérequis système

ShareLaTeX should run on OS X and Linux. You need :

- TeXLive 2013 or later with the latexmk program installed.
- The grunt command line tools (Run `npm install -g grunt-cli` to install them)
- A local instance of Redis (version 2.6.12 or later) and MongoDB running on their standard ports.
- Node.js 0.10 or greater. We recommend that you use nvm to install it.

TexLive : installation par le réseau

2600+ paquets TexLive pour l'installation par défaut

Installation par le réseau

```
apt-get install screen
screen -S
wget http://mirror.ctan.org/systems/texlive/tlnet/install-tl-unx.tar.gz
tar xzf install-tl-unx.tar.gz
cd install-tl-20140403/
time ./install-tl
[...]
Welcome to TeX Live!
Logfile: /usr/local/texlive/2013/install-tl.log

real      32m58.775s
user      3m49.686s
sys       1m1.964s
```

TexLive : installation par image ISO

ISO texlive2013-20130530.iso (wget => 29 secondes)

Récupération de l'installateur

```
md5sum -s texlive2013-20130530.iso.md5
sha256sum -c texlive2013-20130530.iso.sha256
mkdir loop
mount -o loop texlive2013-20130530.iso loop/
cd loop/

time ./install-tl
[...]
Installing [0001/2691, time/total: ?:?:?/??:?]: 12many [376k]
Installing [0002/2691, time/total: 00:00/00:00]: 2up [66k]
[...]
Welcome to TeX Live!
Logfile: /usr/local/texlive/2013/install-tl.log

real      10m33.088s
user      5m48.098s
sys       1m53.967s
```



TexLive : Configuration du système

Edition du fichier /etc/profile

```
# A ajouter dans /etc/profile
# juste avant la ligne "export PATH"
PATH="$PATH:/usr/local/texlive/2013/bin/x86_64-linux"
```

Ensuite executer la commande "source /etc/profile" ou mieux : rédémarrer.

Installation de NodeJS/NPM

```
apt-get install git mongodb-server gcc make flex bison tcl8.5 g++ curl

wget http://nodejs.org/dist/v0.10.29/node-v0.10.29.tar.gz
tar xvzf node-v0.10.29.tar.gz
cd node-v0.10.29/
./configure
time make
make install

# node -v
v0.10.29
# npm -v
1.4.14

npm install -g grunt grunt-cli
```

<http://nodejs.org/download/>

Récupération de ShareLatex

Récupération du dépôt initial

```
git clone https://github.com/sharelatex/sharelatex.git
Cloning into 'sharelatex'...
remote: Reusing existing pack: 449, done.
remote: Total 449 (delta 0), reused 0 (delta 0)
Receiving objects: 100% (449/449), 112.67 KiB, done.
Resolving deltas: 100% (223/223), done.
```

```
cd sharelatex
grunt install
```

```
14M   .
root@erato9:~/tmp/sharelatex# grunt install
Running "check:make" task
Checking make is installed... OK.
Running "install:web" task
Cloning into 'web'...
remote: Reusing existing pack: 3782, done.
remote: Counting objects: 85, done.
remote: Compressing objects: 100% (84/84), done.
Receiving objects: 37% (1431/3867), 5.57 MiB | 1.25 MiB/s
```

Lancement de ShareLatex

De préférence dans des "screen"

Lancement du serveur REDIS

redis-server

Lancement de ShareLatex

```
cd sharelatex  
grunt run
```

Démarrage automatique

```
diff Z-redis skeleton
< DESC="server\_REDIS\_pour\_ShareLatex"
< NAME=REDIS
> DESC="Description\_of\_the\_service"
> NAME=daemonexecutablename
< DAEMON=/usr/local/bin/redis-server
< DAEMON_ARGS=""
<start-stop-daemon --start --background --quiet \
--pidfile $PIDFILE --exec $DAEMON --test > /dev/null \
>start-stop-daemon --start --quiet --pidfile $PIDFILE \
--exec $DAEMON --test > /dev/null \
<start-stop-daemon --start --background --quiet \
--pidfile $PIDFILE --exec $DAEMON -- \
>start-stop-daemon --start --quiet --pidfile $PIDFILE \
--exec $DAEMON -- \
```

Démarrage automatique

```
diff Z-ShareLatex skeleton | grep -v "#"
< PATH=/sbin:/usr/sbin:/bin:/usr/bin:/usr/local/bin:/\
/usr/local/texlive/2013/bin/x86_64-linux
< DESC="server_Sharelatex"
< NAME=Z-ShareLatex
< DAEMON="/usr/local/bin/grunt"
> PATH=/sbin:/usr/sbin:/bin:/usr/bin
> DESC="Description_of_the_service"
> NAME=daemonexecutename
> DAEMON=/usr/sbin/$NAME
< DAEMON_ARGS="run"
<     start-stop-daemon —start —background —pidfile $PIDFILE \
—chdir /root/tmp/sharelatex/ —exec $DAEMON —test > /dev/null \
>     start-stop-daemon —start —quiet —pidfile $PIDFILE \
—exec $DAEMON —test > /dev/null \
<     start-stop-daemon —start —background —pidfile $PIDFILE \
—chdir /root/tmp/sharelatex/ —exec $DAEMON \
<             $DAEMON_ARGS \
>     start-stop-daemon —start —quiet —pidfile $PIDFILE \
—exec $DAEMON \
>             $DAEMON_ARGS \
```

Configuration de la langue

Installation des fichiers de langue correspondant

```
apt-get install aspell-fr
```

Modification du fichier config/settings.development.coffee

```
# Spelling languages
# -----
languages: [
    {name: "English", code: "en"},
    {name: "Francais", code: "fr"}
]
```

Configuration du mail sortant

Modification du fichier config/settings.development.coffee

```
# Email support
# -----
email:
  fromAddress: "noname@u-bourgogne.fr"
  replyTo: "noname@u-bourgogne.fr"
  lifecycle: false
  transport: "SMTP"
  parameters:
    host: "smtp.u-bourgogne.fr"
    port: "25"
```

Utiliser des TABULATIONS

Correction de l'URL dans les mails

Dans les mails (reset password par exemple), l'URL est celle de ShareLatex.com, pas celle de notre serveur.

Recherche des fichiers contenant l'URL ShareLatex.com

```
find . -type f -exec grep -H -i "sharelatex.com" {} \;  
  
. /web/app/coffee/Features/Email/EmailBuilder.coffee  
. /web/app/js/Features/Email/EmailBuilder.js
```

Substitutions à effectuer pour les fichiers concernés (ex sous Vim)

```
:%s/www.sharelatex.com/MonServeur.u-bourgogne.fr:3000/g  
:%s/https/http/g
```

Retour ASR sur la solution 1/2

- Limitation technique (perso) : pas de maîtrise de l'architecture
 - mises à jour : fail
 - pas de paquet Debian
 - Pas de fonction d'admin globales
 - difficulté de retrouver les mails pour partage/création de projets
 - LDAP serait bienvenu
 - réassigner le document principal
 - supprimer un projet
- Plus gourmand en RAM (4Go) qu'en socket

Retour ASR sur la solution 2/2

Pour les ASRs

- SSL – > reverse Proxy Apache2/NGinx
- pour les ASR & les utilisateurs
 - Pas de fonction d'admin globales
 - difficulté de retrouver les mails pour partage de projets
 - réassigner le document principal
 - supprimer un projet
- correcteur orthographique
- rafraîchissement du PDF
- commenter tout un bloc d'un coup

Retour d'expérience

(pub)Quelques mots sur l'ESIREM (/pub)

Ecole d'ingénieur ESIREM

- Ecole d'ingénieur de l'université de Bourgogne, campus Dijon
- Deux départements Matériaux / InfoTronique
- Une spécialité Service et Qualité des Réseaux
- Formation d'administrateurs réseaux
- Centre de formation certification CCNA 4
- => Stages assistant-ingénieur / PFE / contrat-pro

Besoins rédactionnels

Rédaction de documents

- Articles de recherche
- Rapports d'activité
- Dossiers de financement (ANR, ...)
- Autres éléments divers

Caractéristiques

- Co-auteurs locaux / distants
- Travail rédactionnel en parallèle requis
- Délais de plus en plus serrés
- Volumes conséquents (dizaines de pages)

L'existant

Edition collaborative avec Word

- Activation du mode versionning sur Word
- Découpage du travail rédactionnel en groupes
- Edition parallèle entre participants
- Echanges par mail
- Travail sur des versions parfois obsolètes
- Un chef de projet pour fusionner les modifications
- Solution "moindre mal"

L'existant

Edition collaborative avec Latex

- Plusieurs fichiers (images, segmentation de texte, ...)
- Echanges par mail : fichiers simples vs archives
- Espace de stockage partagé : rigueur dans l'édition
- Dropbox, Owncloud, SVN, GIT, ...

Besoins rédactionnels

Caractéristiques d'une bonne solution d'édition

- Travail rédactionnel en totale indépendance
- Aucune perte d'édition concurrente
- Suivi de version

Retour d'expérience

Mon expérience sur ShareLatex

- 12 projets d'édition : articles, dossiers de financement, présentations
- 2-4 collaborateurs par projet
- Conditions d'édition très différentes (durée, intensité)

Observations générales

- Adéquation totale avec le besoin
- Prise de main immédiate
- **Très faible résistance au changement**

Retour d'expérience

The screenshot shows the ShareLaTeX interface. On the left, there's a sidebar with project navigation (Projects, Blog, LaTeX Templates, Info), file list (including 'CoDIT'14.tex'), and account information. The main area has tabs for 'Recompile', 'Logs 10 3', and 'Download'. A preview window on the right shows the document's output.

Code Snippets:

```

376 connected.
377 The scheduling problem is transformed into a maximum flow
problem in the following ways:
378 - \begin{itemize}
379 \item We add a source $S$ and a sink $S$,
380 \item We add arcs $(x,y)$, $\forall$onl $x \in X, S$ (resp. $S, y \in Y$)
with capacity $\sum_{i=1}^n k_i$ (resp. $\sum_{i=1}^m l_i$)
381 \item $S$ is with $\sum_{i=1}^n k_i = \lfloor \frac{\alpha}{\beta} \rfloor$ where $\alpha = \sum_{i=1}^n k_i$ and $\beta = \sum_{i=1}^m l_i$.
382 Finally, we add $S(x,y)=1$, $\forall$onl $(x,y) \in E$.
383 \end{itemize}
384 The resulted graph is denoted by $G'=(S, t, X, Y, E')$.
385
386 Now, we show that the flow $G'$ admits a maximum flow with
$\sum_{i=1}^n k_i \leq \sum_{i=1}^m l_i$. Suppose that the flow $G'$ is
maximum with value $\sum_{i=1}^n k_i > \sum_{i=1}^m l_i$. Therefore it
exists one arc $(\tilde{x},y), \tilde{x} \in X$ unsaturated by the flow $G'$.
Consequently, we have one arc $(x^*,y), x^* \in X, y \in Y$.
387 Then it exists one vertex $y_* \in Y$ s.t. $y_* \neq y$ and
$(\tilde{x},y) \rightarrow y_*$, $(x^*,y) \rightarrow y_*$ otherwise the task $x^* \rightarrow y$ must be packed into $(\tilde{x},y)$. Moreover, we assume
that it exists
388 one arc $(\tilde{x},y), y \neq y_*$ with $\sum_{i=1}^n k_i > \sum_{i=1}^m l_i$.
389
390 Since $G'$ is connected, the chain $\sum_{i=1}^n k_i \leq \sum_{i=1}^m l_i$ is
of lenght even exists. We change the
flow value $\sum_{i=1}^n k_i \geq \sum_{i=1}^m l_i$ for all
pair of vertices in this chain, with $S \in \{0,1\}^S$ and
finally we put $G'=(X, Y, E')$. Therefore, it exists a
new feasible flow $G''$ with a value strictly greater than $G'$.
Since it exists a $S \rightarrow t$ cut with capacity $\sum_{i=1}^n k_i$,
then the maximum value of $G''$ is necessarily $\sum_{i=1}^n k_i$.
391
392 The $S$-tasks such that $S(x,y)=1$ in $G''$ are packed into
393

```

Approximation algorithm for constrained coupled-tasks scheduling problem

Gilles Nolinie¹ and Bertrand Darties² and Yann Claude Klein¹ and Régis de la Graudais¹

¹LIRMM UMR 5506 CNRS - Université de Montpellier II
34394 Montpellier Cedex 09, France
²INRIA Rocquencourt, Domaine de Voluceau, BP 103, 78154 Le Chesnay Cedex, France
³INRIA Paris-Rocquencourt, Domaine de Voluceau, BP 103, 78154 Le Chesnay Cedex, France
⁴Institute for Data Analysis - University College Cork - Ireland
gilles.nolinie@inria.fr; regis.dela.graudais@inria.fr

Abstract. We tackle the makespan minimization coupled-tasks scheduling problem. In this problem, a set of tasks, each having a processing mode once started, a subtask cannot be interrupted. The tasks are coupled by constraints. We prove that for any task started at \$t\$, the first sub-task is fully executed before the second one starts. We propose some complexity results according to several constraints and design an efficient approximation algorithm.

1. INTRODUCTION

In this article, we study the makespan minimization coupled-tasks scheduling problem. A coupled-task is a task which whose execution must be supported by a fixed interval of time. This interval is called the support of the task. The support of a coupled-task includes, among others, ride duration process times, or the time required to switch between two different sub-tasks, and finally time for an action such as a second sub-task. Between these actions, in both ends of the support of the task, there is a minimum time of rest. The aim is to minimize the makespan (schedule length) of a set of given tasks. This problem is NP-hard [1]. It is also known to contain one or several different subproblems during the search for an optimal solution. For example, it is a scheduling problem with events delays on single machine. This problem is NP-hard [2]. It is also known to contain various duration of the sub-tasks and their idle times. In this paper, we consider the makespan minimization coupled-tasks configuration and presented in [3].

2. PROBLEM STATEMENT

All coupled tasks in our problem are coupled tasks per task. The start times of the sub-tasks and the idle times are equal to the same value called the task stretch factor.

3. COUPLED-TASKS SCHEDULING

3.1. THE SCHEDULING COUPLED-TASKS

Each acquisition task is a coupled-task \$A_i = (x_i, y_i, k_i)\$ composed by two sub-tasks of processing time \$x_i\$ and \$y_i\$, respectively dedicated for work transportation and idle transport.

Between these two sub-tasks there is a third idle time \$z_i\$ which is the time between the end of the first sub-task and the start of the second sub-task. It is a non-processing mode once started, a subtask cannot be interrupted. The tasks are coupled by constraints. We prove that for any task started at \$t\$, the first sub-task is fully executed before the second one starts. We denote by \$S\$ the set of tasks and \$T = \{t_1, t_2, \dots, t_n\}\$ the set of times. We note \$A = \{A_1, \dots, A_n\}\$ the collection of coupled tasks to be scheduled.

This paper is organized as follows. In section 2, we present what are the duration of the sub-tasks, the second sub-task and the idle times. In section 3, we present how to model an original task \$(x_i, y_i, k_i) = (1, 1, 1)\$. Formally, a standard coupled-task is a triple \$(x_i, y_i, k_i) = (\sum_{j=1}^m x_{ij}, \sum_{j=1}^m y_{ij}, \sum_{j=1}^m k_{ij})\$, where \$x_{ij}, y_{ij}\$ and \$k_{ij}\$ are the stretch factor of the task. In the rest of the paper, coupled tasks are always named coupled tasks, and sub-tasks are always named sub-tasks. \$x_i\$ and \$y_i\$ are called a single identifier, i.e., identical. In each configuration, the tasks are labeled by their index \$i\$ and the stretch factor of the task is \$k_i\$. The stretch factor of the task is the sum of the following conditions must holds:

- 1) The stretch factor of the task is the same time of our task is fully exploited to schedule a subtask from the other tasks to be scheduled during \$[t_1, t_2]\$. It is called stretch factor of the task. \$k_i\$ is the stretch factor of the task. As \$k_i\$ is fully exploited during the idle time \$z_i\$ of \$A_i\$, for sake of simplicity, we note \$k_i = k_i z_i\$.
- 2) The stretch factor of the task is fully exploited during the idle time \$z_i\$ of \$A_i\$.

In following, we focus on specific compatibility graph \$G_C\$. This graph is a bipartite graph with two sets of nodes \$X\$ and \$Y\$. The nodes of \$X\$ are the sub-tasks and the idle times, and the nodes of \$Y\$ are the tasks. The edges of \$G_C\$ connect the sub-tasks and the idle times to the tasks. We say that a node \$x \in X\$ is adjacent to a node \$y \in Y\$ if \$x\$ is compatible with \$y\$. We say that a node \$x \in X\$ is adjacent to a node \$y \in Y\$ if \$x\$ is adjacent to \$y\$ in the set \$X\$. Moreover, we say that a node \$x \in X\$ is adjacent to a node \$y \in Y\$ if \$x\$ is adjacent to \$y\$ in the set \$Y\$. If \$x \in X\$ is adjacent to \$y \in Y\$, we say the edge \$(x, y) \in E(G_C)\$.

3.2. HEURISTICS AND SOLVABLE CASE

We want to design a polynomial-time algorithm for the scheduling problem in which the maximum degree of incoming arcs on \$Y\$ tasks is at most two.

Retour d'expérience

Exemple de suivi de version

Recent changes		
Today		
	feedback.tex, main.tex, limitations.tex	10:15 pm
	You	
	Arnaud.dacosta	
	main.tex	9:21 pm
	Arnaud.dacosta	
	limitations.tex, main.tex, templates.tex	5:58 pm
	Arnaud.dacosta	
	limitations.tex, draft.tex, main.tex	5:32 pm
	Arnaud.dacosta	
	main.tex, draft.tex, redis.tex, limitations.tex	5:16 pm
	Arnaud.dacosta	
	main.tex	4:02 pm
	Arnaud.dacosta	



Retour d'expérience

Exemple de suivi de version

```
\begin{frame}
\f[Added by you on 25th Jun 2014, 9:37 pm] oins en tant que chercheur]
\begin{block}{Rédaction de documents}
-
\begin{itemize}
\item Articles de rechercheBugs de
\item Rapports d'activité
\item Dossiers de financement (ANR, \dots)
\item Autres éléments divers
\end{itemize}
\end{block}
```

Retour d'expérience

Caractéristiques d'utilisation

- Création d'un projet + partage très rapide
- Coloration syntaxique, auto-complétion
- Interface simple
- Importation de projet
- Espace de sauvegarde
- Visualisation de l'édition concurrente en direct
- Faible latence en distant / local
- La plupart des options des solutions locales sont supportées
- Gestion des déconnexions / reconnexions

Retour d'expérience

Limitations

- Dépendance à une connexion internet
- Peu de possibilités de bidouillage
- Un seul fichier pdf par projet
- Versioning à chaque compilation seulement
- Correspondance document PDF / ligne latex
- Problèmes "courants" de latex : DVI/PS vs PDF ; format des images ; encodage des caractères

Retour d'expérience

Quelques bugs et difficultés

- Manipulations mineures, suppression de projets
- Avantage :open-source, communauté active

The screenshot shows a list of issues from a GitHub-like interface. The left sidebar lists labels: big feature (1), bug (2), enhancement (10), good for beginners (2), high priority (1), and moved to another repository (2). The main area displays the following issues:

Issue #	Title	Opened By	Age	Comments
#123	no start in Production mode	fcarlier	8 days ago	7 comments
#121	Error Starting Server	Andrea79	9 days ago	1 comment
#120	I can not access my instalation	filipesaraiva	12 days ago	1 comment
#117	Adding fairy configuration	davidediger	19 days ago	1 comment
#116	grunt run:web fails when redis not on localhost	davidediger	19 days ago	
#115	no site at port 3000	joergre	21 days ago	7 comments

Pointeurs

- <http://linuxfr.org/news/sharelatex-devient-libre>
- FlyLatex à tester : <https://github.com/alabid/flylatex>
- <https://bitbucket.org/rivanvx/beamer/wiki/Home>
- Owncloud LatexTex Editor
[http://apps.owncloud.com/content/show.php/
LatexTex+Editor+and+Compiler?content=151441](http://apps.owncloud.com/content/show.php?LatexTex+Editor+and+Compiler&content=151441)