



CONSERVATOIRE NATIONAL DES ARTS ET MÉTIERS
CENTRE RÉGIONAL ASSOCIÉ DE
NOUVELLE-AQUITAINE

Mémoire probatoire blanc présenté en vue d'obtenir
UE « Information et communication pour ingénieur »
Spécialité :
Informatique, Réseaux, Systèmes et Multimédia

par

Marc BENINCA

Concepts et outils pour
mieux produire des documents

Soutenu le 19 Mai 2020

JURY

PRÉSIDENT : Mme Catherine BRULATOUT **Maître de conférences associé**
Université de Bordeaux

MEMBRES : Mme Elisabeth ABRIVAT **Maître de conférences associé**
Université de Bordeaux

Abréviations

BASH Bourne Again SHell. 9

GPG GNU Privacy Guard. 11, 12

PDF Portable Document Format. 3

PDFTK PDF ToolKit. 10, 11

PGP Pretty Good Privacy. 11

SE Système d'Exploitation. 5

SI Système d'Information. 5, 6

SSH Secure SHell. 12

SSI Sécurité des Systèmes d'Information. 12

WYSIWYG What You See Is What You Get. 7

WYSIWYM What You See Is What You Mean. 7, 8, 11, 12



Concepts et outils pour mieux produire des documents

Plan

1 Introduction

À notre époque dans laquelle l'informatique est devenue omniprésente, la consultation d'informations numérisées n'a jamais été aussi importante.

Ainsi devient-il de plus en plus nécessaire de mener diverses réflexions, pour tendre vers une production toujours plus fiable de documents, afin de pouvoir véhiculer au mieux les informations numérisées.

Plusieurs supports numériques sont apparus au fil du temps :

- le texte
- l'image
- l'audio
- la vidéo

La forme la plus courante de transmission d'informations restant aujourd'hui le document, combinant à la fois des contenus textuels et imagés, le plus souvent encapsulés dans un fichier Portable Document Format (PDF).

Deux grandes parties seront ici abordées à son sujet :

1. quels objectifs se fixer pour améliorer la production de documents ?
2. vers quels moyens se tourner pour tendre vers de tels objectifs ?

2 Objectifs

Ces deux grands buts permettent d'aborder plus sereinement la production :

1. bénéficier d'un maximum de liberté par rapport aux outils
 2. minimiser les risques d'altération et de perte de travail
-

2.1 Indépendance numérique

Quatre niveaux fondamentaux sont à considérer pour s'en approcher :

1. les systèmes distants
2. les systèmes locaux
3. les applications utilisées
4. les fichiers manipulés

2.1.1 Plateformes en ligne

De nombreuses organisations et entreprises proposent des serveurs sur Internet permettant de stocker et synchroniser des données, voire de travailler sur des documents directement depuis un navigateur web.

Dans le cadre de l'utilisation d'une telle infrastructure, il convient de mener une réflexion en se posant les questions suivantes :

- est-il possible de continuer de travailler localement sur sa machine en cas de coupure de connexion à Internet pour une durée indéterminée ?
- la synchronisation des données passe-t-elle par un protocole standard, permettant l'utilisation de n'importe quel logiciel compatible pour ce faire, ou bien contraint-elle à l'utilisation d'un unique outil incontournable ?
- les données d'utilisateurs sont-elles vendues à des tierces parties ?
 - plateformes de diffusion publicitaire en ligne
 - moteurs d'apprentissage pour intelligence artificielle

2.1.2 Systèmes d'exploitation

Également dirigés et fournis par des organisations ou des entreprises, ils varient grandement par leurs objectifs et leur nature, et peuvent influencer de façon non négligeable sur la production de documents en leur sein.

Quelques questions à se poser au sujet d'un Système d'Exploitation (SE) :

- dispose-t-il de systèmes de fichiers modernes à base de transactions, permettant de minimiser grandement le risque de corruption de données ?
- permet-il aux utilisateurs de planifier les mises à jour du système, en respectant leurs desideratas pour éviter toute indisponibilité de travail ?
- quel est son coût réel ?
 - impose-t-il de la publicité et/ou d'autres distractions gênantes ?
 - respecte-t-il la vie privée des utilisateurs et de leurs données ?
 - pratique-t-il l'obsolescence programmée de ses versions successives, poussant ainsi les utilisateurs au rachat inutile de nouveau matériel ?

2.1.3 Logiciels de production

Outre le fait de varier aussi en fonction de l'organisation ou l'entreprise qui le développe, un logiciel de production peut être considéré comme un Système d'Information (SI) à part entière.

Quelques critères importants pour le choix d'un SI :

- fonctionne-t-il sous le SE retenu ?
- est-il toujours activement développé et maintenu ?
- quel est son coût d'utilisation ?
 - **achat périodique** : à chaque sortie d'une version majeure
 - **abonnement** : généralement mensuel ou annuel
 - **prix libre** : via des dons facultatifs sans montant fixé
- sous quelle licence est-il publié ?
 - **propriétaire** : interdisant toute analyse ou modification de son fonctionnement
 - **open source** : autorisant l'analyse, mais comportant des clauses restrictives
 - **libre** : autorisant toute analyse, modification ou redistribution du programme

2.1.4 Formats de fichiers

Tout SI permet de lire et écrire à partir de différents formats de fichiers.

Réflexions afin de retenir un format de fichier pour un document :

- dispose-t-il d'une documentation publique détaillant sa structure, permettant d'être lu et écrit par tout logiciel respectant ce standard ?
 - est-il documenté pour une plus grande pérennité ?
 - **fermé** : non documenté
seul le logiciel de l'éditeur en permet une manipulation correcte
 - **obfusqué** : partiellement documenté
afin de faire dysfonctionner les logiciels tiers au fil des versions, pour inciter à utiliser le logiciel éditeur, tout en feignant l'ouverture
 - **ouvert** : complètement documenté
permettant à de nombreux logiciels de s'interfacer avec
 - de quel type de format s'agit-il ?
 - **binaire** :
plus performant à l'utilisation par une machine,
mais nécessite des logiciels compatibles pour pouvoir les manipuler
 - **textuel** :
humainement lisible et modifiable avec n'importe quel éditeur de texte,
plus facile à utiliser pour détecter et appliquer des modifications,
mais prend un peu plus de temps à être interprété par une machine
(écart minime, avec la puissance de calcul disponible de nos jours)
-

2.2 Fiabilité des contenus

Quatre grands axes rendent les contenus produits plus fiables :

2.2.1 Disponibilité

Il s'agit de l'assurance de pouvoir accéder à son document à tout moment.

Cela peut passer par deux voies :

- choisir de l'hébergement garantissant une forte disponibilité
- multiplier les hébergements pour être résilient à toute panne

2.2.2 Intégrité

La garantie qu'un document n'a pas été altéré entre sa dernière modification et sa prochaine consultation.

Deux options peuvent être utilisées :

- stocker sur des systèmes de fichiers à transaction et somme de contrôle
- utiliser des outils de gestion de configuration, calculant eux-mêmes des sommes de contrôle au fil des sauvegardes

2.2.3 Authenticité

La preuve qu'un document a bien été produit par son auteur déclaré.

Deux mécanismes sont à mettre en œuvre, pour permettre :

- à l'auteur de signer numériquement les documents qu'il produit
- aux utilisateurs de vérifier eux-mêmes la validité de cette signature

2.2.4 Reproductibilité

Deux philosophies de fabrication de documents s'opposent :

- **What You See Is What You Get (WYSIWYG)** : la plus courante permettant de modifier un document visuellement depuis son rendu final
 - bureautique : LibreOffice [**lo**], MicroSoft Office [**mso**], WPS Office [**wps**]
- **What You See Is What You Mean (WYSIWYM)** : la plus pérenne consistant pour un document à décrire successivement dans un fichier les différents éléments à intégrer ou actions à effectuer, puis à compiler un rendu final à partir de ce programme
 - moteur T_EX : X_ƎΛT_EX, ΛT_EX [**latex**]
 - GraphViz [**graphviz**], GnuPlot [**gnuplot**], Sphinx [**sphinx**]

3 Moyens

L'accent sera ici mis sur l'utilisation de logiciels libres et de l'approche WYSIWYM pour améliorer la fiabilité de la production.

Choix d'outils pour l'implémentation de trois grandes phases :

- mettre en place des opérations clés incontournables
 - automatiser l'exécution de ces tâches maîtresses
 - étendre le spectre de toutes les tâches automatisables
-

3.1 Rationalisation

Au moins deux piliers sont indispensables :

3.1.1 Gestion de configuration distribuée

Utiliser un outil permettant à la fois de :

- garder une trace de toutes les tâches prévues et réalisées
- sauvegarder des modifications partielles de document en tant que telles
- bénéficier d'un contrôle d'intégrité automatique de ces sauvegardes
- travailler séparément sur différents contextes de modifications
- pouvoir revenir à un état antérieur cohérent à tout moment
- intégrer des modifications provenant de divers collaborateurs
- répliquer ces sauvegardes sur plusieurs serveurs pour plus de disponibilité

Les plus connus étant : Git [**git**], Mercurial [**hg**], Fossil [**fossil**], Bazaar [**bazaar**].

3.1.2 Processus de fabrication

Définir un cheminement menant à la reproductibilité d'un document.

La plupart du temps :

- extractions d'éléments depuis des documents sources
 - préparations ou conversions d'éléments à intégrer
 - compilation du document avec intégration de ressources
 - assemblage final avec d'autres documents si nécessaire
 - signature du document, si besoin d'authentification
 - déploiement du document final vers des hébergements
-

3.2 Automatisation

Il s'agit ici d'implémenter et orchestrer le processus de fabrication défini.

Deux approches sont possibles :

3.2.1 Fichiers de fabrication

Fichiers textuels très simples, se contentant de lister les différentes étapes, ainsi que toutes les commandes associées à chacune des étapes recensées.

Le standard : Make [**make**].

3.2.2 Scripts d'assemblage

Fichiers textuels plus ou moins complexes, laissant la liberté à l'auteur de programmer toutes les opérations qu'il souhaite autour des étapes.

Les plus courants : Bourne Again SHell (BASH) [**bash**], Python [**py**].

3.3 Opérations automatisables

De multiples étapes peuvent intervenir dans la production d'un document, et ainsi être intégrées au sein d'un fichier d'automatisation :

3.3.1 Conversion de documents

Transformation de documents existants dans un format plus propice à l'outil d'assemblage retenu pour le processus de fabrication.

Exemples : PanDoc [**pandoc**], ImageMagick [**imagemagick**].

3.3.2 Découpe de documents

Extraction de certaines pages de documents finaux déjà compilés.

La référence : PDF ToolKit (PDFTK) [**pdftk**].

3.3.3 Extraction d'éléments

Récupération d'une ou plusieurs images spécifiques d'un autre document.

Exemples : Poppler [**poppler**], GhostScript [**gs**].

3.3.4 Rotation d'éléments

Ajustement de l'orientation de pages ou images disponibles.

Exemples : PDFTK [**pdftk**], ImageMagick [**imagemagick**].

3.3.5 Compression d'images

Réduction du volume de données occupé par des images.

La référence : ImageMagick [**imagemagick**].

3.3.6 Résolution d'impression

Modification de la taille d'images par rapport au gabarit du document.

La référence : ImageMagick [**imagemagick**].

3.3.7 Compilation

Production de rendu du document principal, à partir d'un fichier WYSIWYM, à l'aide d'un moteur de rendu approprié au format WYSIWYM retenu.

Exemples : \LaTeX [**latex**], Sphinx [**sphinx**].

3.3.8 Assemblage de documents

Quand il est nécessaire pour obtenir un document final de mettre bout à bout des extraits préparés, le document principal compilé et certaines annexes.

La référence : PDFTK [**pdftk**].

3.3.9 Signature numérique

Afin de pouvoir vérifier l'authenticité du document, quand il est consulté par des personnes souhaitant s'en assurer.

Le standard : GNU Privacy Guard (GPG) [**gpg**],
une implémentation libre du standard ouvert Pretty Good Privacy (PGP) [**pgp**].

Signer un document :

```
1 gpg
2 --armor           # sous forme textuelle
3 --detach-sign    # signer dans un fichier séparé
4 'nom_du_document.pdf' # ce document
```

Vérifier une signature :

```
1 gpg
2 --verify          # vérifier la validité
3 'nom_du_document.pdf.asc' # de cette signature
4 'nom_du_document.pdf'   # pour ce document
```

3.3.10 Réplication de contenus

Répliquer vers plusieurs serveurs, pour une disponibilité à toute épreuve.

Exemple : Rsync [**rsync**] via une connexion sécurisée Secure SHell (SSH) [**ssh**].

4 Conclusion

Il est aujourd'hui possible de mettre en place une véritable industrialisation de la production de documents, en combinant :

- l'utilisation de logiciels libres
- le choix de formats ouverts et textuels
- une gestion de configuration distribuée
- l'automatisation des tâches de fabrication
- l'adoption d'une démarche WYSIWYM
- l'utilisation de la signature numérique
- un déploiement automatique multi-sites

Cette rationalisation permet de réduire drastiquement les risques, en renforçant des points essentiels de la Sécurité des Systèmes d'Information (SSI) :

- authentification
- intégrité
- disponibilité

La confidentialité via GPG [**gpg**] n'est ici pas abordée, car ce sujet aurait occupé une place disproportionnée par rapport aux autres parties.

Ce document applique lui-même toutes les recommandations qu'il préconise.

Suivent en annexes quelques fichiers exemples du processus mis en place...