



HAL
open science

RESINFO propulse WAPT sur l'orbite ESR, la science en paquets !

Philippe Hortolland, Jérôme Colombet, Fabien Muller, Christophe Millien,
Gabriel Pierre André Moreau, Sophie Nicoud

► To cite this version:

Philippe Hortolland, Jérôme Colombet, Fabien Muller, Christophe Millien, Gabriel Pierre André Moreau, et al.. RESINFO propulse WAPT sur l'orbite ESR, la science en paquets!. Congrès JRES : Les Journées Réseaux de l'Enseignement et de la Recherche, RENATER, May 2022, Marseille, France. hal-03608854

HAL Id: hal-03608854

<https://hal.science/hal-03608854v1>

Submitted on 18 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC0 - Public Domain Dedication 4.0 International License

RESINFO propulse WAPT sur l'orbite ESR, la science en paquets !

Philippe Hortolland

Institut de mécanique et d'ingénierie
Site ENSAM – Esplanade Arts et Métiers – 33 400 Talence

Jérôme Colombet

Laboratoire de Chimie de Coordination
205 route de Narbonne – 31 400 Toulouse

Fabien Muller

Institut de Physique et de Chimie des Matériaux de Strasbourg
23 Rue du Loess – 67 034 Strasbourg Cedex

Christophe Millien

Institut de biologie intégrative de la cellule (I2BC)
Bat 34 Avenue de la terrasse – 91 190 Gif sur Yvette

Gabriel Moreau

Laboratoire LEGI
Domaine Universitaire
1209-1211 rue de la piscine – 38 400 Saint-Martin-d'Hères

Sophie Nicoud

Institut d'Électronique et des Systèmes
Campus Saint Priest – 860 rue Saint Priest – 34 000 Montpellier

Résumé

WAPT [1] est une solution logicielle développée par la société française Tranquil IT [2] basée à Nantes. À travers une console de gestion centralisée, WAPT permet d'administrer un parc informatique en seulement quelques clics ! Il donne ainsi la possibilité de déployer une pléiade de logiciels à partir d'un magasin local dont les modèles de scripts de déploiement sont mis à disposition par l'éditeur ou bien à partir d'un magasin tiers, par exemple un magasin commun pour tout l'ESR.

Un groupe de travail RESINFO s'est mis en orbite autour de la solution WAPT à partir de 2018. Nous travaillons à fédérer les actions et les connaissances autour de l'outil WAPT, à co-construire et à partager les recettes de déploiement des logiciels. Le GT-WAPT [3] propose un magasin pour l'ESR sur un dépôt central sécurisé hébergé sur l'infrastructure cloud Mathrice (Plateforme en Ligne pour les Mathématiques). L'ensemble des entités peuvent ainsi interconnecter leur propre dépôt logiciel WAPT avec la solution communautaire. Chaque site aura ainsi le choix de piocher ou non les logiciels dans ce magasin partagé, et de participer à l'effort collectif de maintenance et d'ajout de nouveaux logiciels.

La création et le suivis dans le temps de ces paquets « scientifiques » et « bureautiques » pour les thématiques non exhaustives (chimie, biologie, physique, électronique, mécanique...) peuvent ainsi profiter à tout l'ESR. Ces paquets sont éditables, adaptables et redistribuables à votre structure via l'ajout de vos licences et de vos paramètres spécifiques. Le magasin central peut ainsi gérer la problématique des logiciels libres et des logiciels propriétaires.

Mot Clefs

WAPT, sécurité, déploiement, mutualisation, Windows, MacOS, Linux, middleware, apt-get

Introduction

Discovery, Enterprise... Ces mots nous baignent dans une ambiance. Pour certains, ils font référence à ces fameuses navettes spatiales américaines, pour d'autres, ils nous plongent dans un univers intersidéral, celui de *Star Trek*. L'objectif de cet article, à l'instar de ces vaisseaux auxquels nous faisons référence, est de nous faire voyager. Cependant d'une manière plus terre-à-terre puisque nous resterons dans l'espace de nos parcs informatiques avec la préoccupation de résoudre d'une manière efficace la problématique du déploiement des logiciels en faisant appel à la communauté. Durant ce court voyage d'une dizaine de pages qui accompagne un poster, nous vous proposons plusieurs escales.

Retours vers le futur de la solution WAPT

Pour la fin d'année 2021, l'éditeur Tranquil IT [4] nous avait réservé plusieurs nouveautés, avec d'une part la sortie de WAPT *Enterprise* 2.1, solution complète comportant un faible coût unitaire pour chaque poste et d'autre part le lancement de WAPT *Discovery* venant remplacer la populaire version *Community*. Tranquil IT conserve ainsi son choix de proposer une solution gratuite destinée aux parcs composés de moins de 300 postes. Cette solution répond, au moins au départ, aux besoins de la plupart des petites structures informatiques. WAPT est une solution logicielle développée par la société française Tranquil IT basée à Nantes, qui permet de déployer une pléiade de logiciels à partir d'un magasin local que vous pouvez alimenter seul, via des modèles mis à disposition par l'éditeur ou bien à partir d'un magasin tiers, par exemple un magasin commun pour tout l'ESR !

Après une première découverte, présentée par Jérôme COLOMBET lors d'un *lightning talk* aux JRES 2015, la solution WAPT de Tranquil IT a été intégrée au catalogue du Groupe Logiciel [5]. Une nouvelle présentation, intitulée « WAPT – solution de déploiement logiciel sous licence GPLv3 (ou comment dire au revoir à SCCM) » fut présentée par Nathalie VIERRA aux JRES 2017 [6]. Lors de l'édition suivante en 2019, ce fut à L'Université de Valenciennes de présenter sa « solution de déploiement et de maintenance logicielle basée sur la solution WAPT » [7]. Cette présentation faisait apparaître quelques points restant cependant à améliorer : qu'en est-il de la gestion des logiciels « lourds », certains logiciels spécifiques à la science soumis à licences par exemple ? En parallèle, une communication a été diffusée en 2018, dans la newsletter de RESINFO [8]. L'objectif était d'identifier les établissements où la solution était déjà utilisée. Suite aux retours de nombreux collègues, un groupe de travail du réseau national RESINFO [9] s'est mis en orbite autour de la solution WAPT. Son but est de fédérer les actions et les connaissances concernant l'outil WAPT, mais aussi de construire et de partager les recettes de déploiement des logiciels.

La fusée était lancée, mais pour atteindre son objectif, l'équipage composé des membres du groupe de travail devait se soumettre à certaines règles. À commencer par une animation soutenue et régulière ainsi qu'au respect scrupuleux du carnet de bord dont voici quelques extraits.

Fonctionnement de l'équipage, carnet de bord

Des visioconférences sont organisées régulièrement sur la plateforme Rendez-Vous (RENATER) [10] pour confronter les idées, partager les connaissances, faire des démonstrations, des tests, selon les environnements de chacun. Hors période de vacances scolaires, la fréquence de nos réunions est fixée à 15 jours, laissant à chacun le temps de travailler sur ses objectifs. Les comptes rendus et suivis de chaque réunion sont saisis dans l'outil collaboratif Etherpad de l'IN2P3 [11]. Ces *pads* sont ensuite sauves sous forme *Markdown* dans le dépôt *Git* de suivis du GT. Pour des raisons de liberté d'expression lors des réunions, mais aussi parce que certaines de nos idées sont parfois nébuleuses, ces notes ne sont pas en accès libre, mais réservées aux seuls spationautes du GT. En dehors de nos réunions, plusieurs instruments de bord sont utilisés. Une liste de diffusion restreinte est hébergée par RESINFO [12], elle est associée à un chat accessible sur la plateforme RocketChat de l'IN2P3 [13] via l'authentification quasi biométrique d'eduGAIN [14]. Ces instruments garantissent les échanges entre membres de l'équipage. Ces outils de communications sont indispensables en raison des particularités de notre groupe, ils nous permettent de garder le contact. Nous sommes tous issus de régions différentes, la plupart d'entre nous ne se sont encore jamais rencontrés physiquement. Aussi, chacun utilise WAPT à sa manière derrière un serveur local installé tantôt sous Linux, tantôt sous Windows, tantôt physique, tantôt virtuel, qu'importe, il n'y a pas de règle. L'important est que tous parlent le même langage même s'il faut l'avouer, après bientôt deux années de travail en commun, chacun prononce encore le mot WAPT à sa manière.

Notre objectif est avant tout d'apporter des solutions à la communauté. Pour transmettre des messages en dehors de la navette, nous avons mis au point plusieurs canaux de communication : une liste de diffusion nationale destinée à tous les agents de l'ESR intéressés par le projet [15] ; un site web arborant le logo spécifique à notre groupe rappelant son attachement à RESINFO ; un *Git* accessible à toute la communauté [16].

Notre projet n'est pas fait que de parole dans le vide, il repose aussi sur une base technique solide comparable à la salle des machines de notre vaisseau dans laquelle se retrouve toute notre infrastructure.

Infrastructure du service, description

Le cœur de notre projet est hébergé par notre partenaire Mathrice (Plateforme en Ligne pour les Mathématiques) [17], dans un environnement virtuel, organisé autour de plusieurs serveurs. L'aspect le plus critique de notre infrastructure est son besoin de place, son réservoir. Notre projet est en effet très consommateur en espace disque sachant que chaque paquet réalisé peut facilement peser plusieurs centaines de Mégaoctets. À cette échelle, on arrive vite à de très gros volumes si on considère la multitude des logiciels dont nous avons besoin dans nos établissements. Entrons plus en détail dans le cœur de notre infrastructure.

Pour gérer le service WAPT national, le groupe s'est appuyé sur une architecture OpenStack [18], celle de la PLM du GDS Mathrice [19]. Les *pilliers de la création* [20] de cette structure sont basés sur quatre serveurs principaux, toutes ces machines virtuelles fonctionnent sous GNU/Linux et offrent différents services : bastion SSH, *reverse proxy*, serveur WAPT, et *builder* (figure 1).

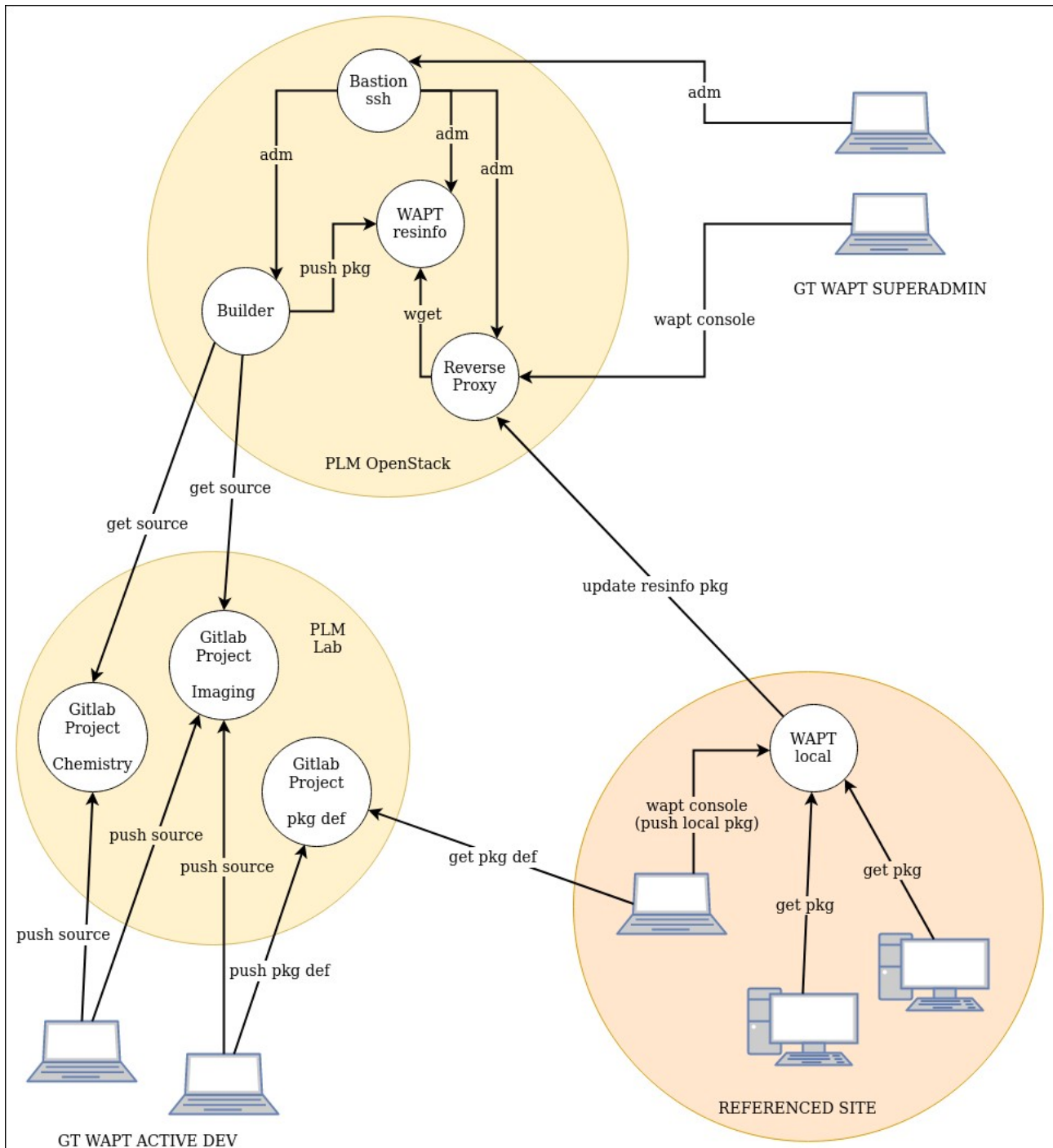


Figure 1 : Schéma général de l'infrastructure du service.

La configuration des services est poussée sur ces machines via des scripts Ansible [21]. Le bastion SSH est invisible pour l'utilisateur du service. Cette machine nous permet simplement d'accéder à distance à toutes les autres pour l'administration des systèmes. Le *reverse proxy* est une machine en frontal sur internet. Elle est quotidiennement mise à jour, filtre et transmet les requêtes au serveur WAPT. Le serveur de diffusion WAPT gère les paquets, il n'est pas accessible directement. Celui-ci est associé au *builder* de paquets qui fonctionne comme un automate, allant chercher les sources dans les différents dépôts *Git* pour les pousser, après les avoir signés, sur le serveur de diffusion.

Toutes ces machines virtuelles ne demandent pas de grande capacité pour fonctionner. Elles sont peu gourmandes en termes de capacité CPU et en mémoire vive, l'ensemble demande néanmoins beaucoup d'espace de stockage pour fonctionner. Il s'agit ici de la description du cœur de notre système, c'est un peu le HAL 9000 de notre odysée autour duquel gravitent plusieurs modules.

Parmi ces modules, deux consoles d'administration sont principalement utilisées pour tester le bon fonctionnement du système et opérer quelques réglages. Elles sont installées sur des machines de type Windows 10, nous pourrions en avoir davantage, trois, quatre ou plus, mais cela n'aurait pas de réel intérêt dans notre mode de fonctionnement. Enfin, nous nous appuyons sur plusieurs dépôts *Git* institutionnels pour héberger notre projet.

Tout est en place pour que les systèmes distants puissent se connecter et récupérer les paquets WAPT dont ils ont besoin. Pour cela il faut être autorisé à accéder au serveur de diffusion. Plusieurs solutions ont été envisagées, au final nous avons fait le choix d'une autorisation sur la base d'un filtrage par adresse IP. Les services informatiques qui veulent se connecter au dépôt RESINFO doivent donc nous en faire la demande en nous communiquant simplement leurs plages d'adresses IP à autoriser. Ainsi, sur chaque site, les ASR des laboratoires ou de l'université peuvent ajouter le magasin WAPT RESINFO pour alimenter leur propre magasin. Nous nous sommes aussi interrogés sur le degré d'importance de la continuité de service de notre serveur. Notons que l'indisponibilité de notre dépôt n'affecte en rien le bon fonctionnement des serveurs mis en place localement sur les différents sites. En cas de panne, le site local ne peut pas intégrer de nouveau paquet RESINFO, par contre, tous ceux déjà intégrés dans la console fonctionnent comme d'habitude. Le service WAPT RESINFO n'est donc pas soumis à des règles de fonctionnement critiques avec des besoins de continuité de service stricte. Une interruption temporaire du service central n'aura qu'un faible impact en local.

À ce stade, il est important de présenter la méthode permettant d'alimenter les dépôts locaux à partir du magasin RESINFO. Cette étape, comme nous allons le voir, est très simple. En revanche la méthode que nous employons pour réaliser des paquets peut s'avérer être très compliquée. Nous allons développer ces parties maintenant.

Méthodologie de distribution et de développement

Méthodologie de distribution

La première étape de notre projet consiste à mettre à disposition toute sorte de paquets logiciels destinés à nos environnements scientifiques à partir d'un magasin unique hébergé par RESINFO. Dans la pratique, nous offrons une solution permettant d'alimenter les magasins locaux rattachés aux serveurs WAPT installés dans les établissements.

Nous précisons qu'il n'est pas prévu que les postes clients des utilisateurs finaux se servent directement à partir du serveur WAPT RESINFO pour installer des logiciels.

La figure 2 indique comment récupérer, à l'aide de la console WAPT, des modèles de paquets mis à disposition par le GT WAPT.

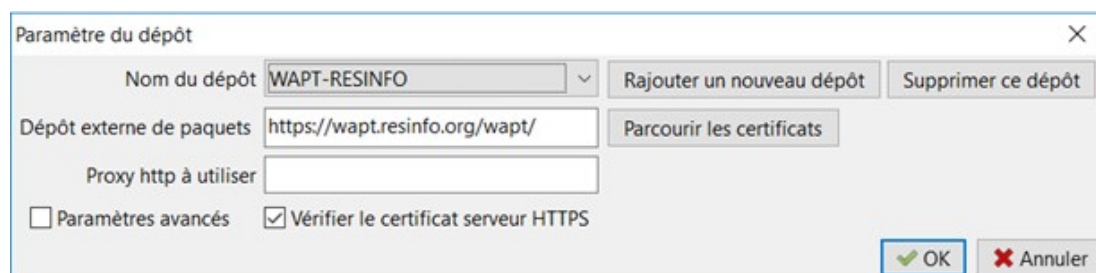


Figure 2 : Ajout du dépôt privé WAPT RESINFO en tant que source à un magasin local.

Cela se fait en deux étapes :

- 1^{re} étape – À partir de la console WAPT du serveur local, aller dans « Outils » puis cliquer sur « Paramètre des dépôts externes ». Renseigner le champ « Nom du dépôt » : WAPT-RESINFO (par exemple) et indiquer l'URL du « Dépôt externe de paquets » : <https://wapt.resinfo.org/wapt/>.
- 2^e étape – Pour récupérer un paquet à partir du nouveau dépôt : cliquer sur « Dépôt privé » puis sur « Importer depuis internet » et sélectionner le nouveau dépôt récemment créé : WAPT-RESINFO. Il n'y a plus qu'à double-cliquer sur une ligne pour importer un paquet logiciel. Après avoir saisi le mot de passe de votre clé privée, le paquet sera automatiquement incorporé à la liste des paquets déjà disponible dans votre dépôt local.

Pour simplifier la recherche, une classification par thématique non exhaustive a été créée ; chimie, biologie, physique, électronique, mécanique, etc. Les paquets « scientifiques » et « bureautiques » sont éditables, adaptables et redistribuables pour chaque établissement via l'ajout de licences et de paramètres spécifiques locaux. Le magasin central peut ainsi gérer la problématique des logiciels libres et des logiciels propriétaires.

À ce niveau, certains voyageurs auront déjà décroché la lune en atteignant leurs destinations. Nous pensons aux ASR de services qui souhaitent simplement être consommateurs de la solution mise en place par le GT WAPT RESINFO, cette escale correspondra à la grande majorité de nos collègues qui liront cet article. En cela, nous sommes très heureux que notre travail puisse vous rendre service.

Jusqu'ici, nous avons décrit tout ce qu'il faut savoir pour vous permettre de piocher les logiciels de votre choix dans notre magasin partagé. Cependant, notre itinéraire ne s'arrête pas là, il est plus ambitieux puisqu'il fait appel à votre motivation pour contribuer à un objectif plus lointain visant à mutualiser la fabrication des paquets pour approvisionner notre magasin.

Dans ce but, nous aimerions à terme harmoniser nos pratiques en proposant un modèle. Dans la suite de l'article, nous présenterons donc la méthodologie de travail bien particulière que nous avons mise en place pour le développement et la distribution des paquets.

Méthodologie de développement

Ici, les choses se compliquent. C'est un peu comme lors du lancement d'une fusée dont la mission consiste à mettre en service un satellite dans l'espace. La réussite de la mission est effective lorsque toutes les étapes, tous les points de contrôle, sont validées. Notre fusée ne propulse pas des satellites, mais des paquets, ceux-ci sont beaucoup plus complexes qu'il n'y paraît. Notre méthode s'appuie sur un outil de gestion de versions décentralisé, un *Git*, afin que le travail du GT ne repose pas uniquement sur un serveur, mais sur un système de connexion pair-à-pair. Ce gestionnaire de versions est hébergé sur la forge *Gitlab* de Mathrice dédié au projet WAPT [22]. Il nous permet de conserver un historique des modifications de tous nos fichiers et donne un suivi de l'évolution de notre travail étape par étape. C'est le fil d'Ariane de notre projet.

Les paquets WAPT sont stockés non seulement sur le serveur *Gitlab* de la PLM, mais également sur les ordinateurs de chaque contributeur du projet. Selon la méthode, ces sources sont dupliquées à l'aide d'outils dédiés au clonage de *Git* tels que *Gitkraken* [23] ou bien *gitCmd* [24]. Cela demande une authentification qui s'effectue à l'aide d'un certificat ou bien en mode HTTPS avec identifiant et mot de passe. Le *Git* est un outil de bas niveau qui se veut simple et performant. Sa principale tâche est de gérer l'évolution des paquets, il permet aussi l'interfaçage à des environnements de développement intégré « IDE » comme les logiciels *Atom* [25], *VSCoDe* [26], *pyScripter* [27]...

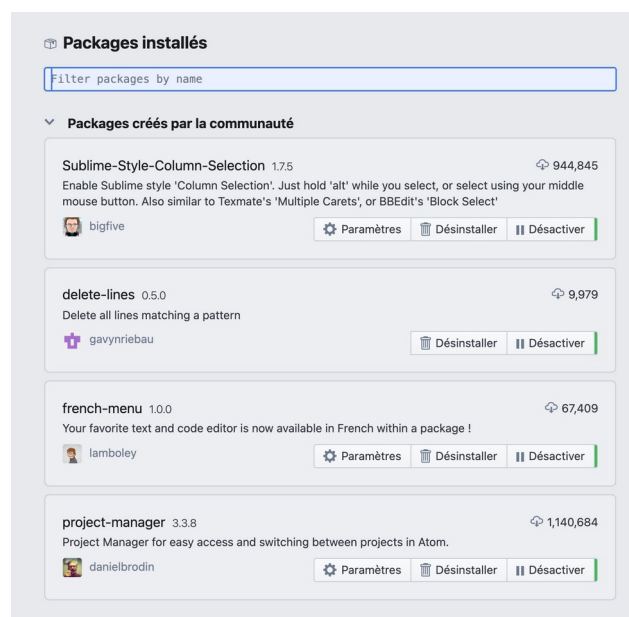


Figure 3 : Greffons Atom conseillé.

Le GT s'est focalisé sur l'IDE multi-plateforme *Atom* associé aux greffons visibles sur la figure 3. L'utilisation d'une interface unique pour l'ensemble des membres du GT permet de diminuer les risques liés à l'utilisation du *Git* en ligne de commande et ainsi évite les mauvais *push* sur le dépôt RESINFO. Les administrateurs système du groupe ayant installé WAPT sur leurs sites se sont familiarisés avec ces outils de développement, notamment ceux plébiscités par Tranquil IT comme *pyScripter* qui permet le déploiement, la signature et le débogage des paquets. Les paquets sont tous testés *in situ* pour être validés en vérifiant leurs bons fonctionnements. Selon la méthode, avant chaque intervention sur le *Git* pour pousser un nouveau paquet, chaque contributeur rapatrie le contenu du dépôt amont vers son poste. Celui-ci est téléchargé et met à jour immédiatement le dépôt local du contributeur. Dans le jargon, on dit qu'il effectue un *fetch* avant d'effectuer un *push*.

La communauté grandissante, il a été décidé de scinder la création des paquets par catégorie thématique scientifique. Un découpage du dépôt *Git* (*chemistry, biology, imaging...*) a permis de désigner des responsables thématiques pour établir un point d'entrée unique pour les membres souhaitant soumettre leurs paquets. Dans chaque branche du dépôt, un fichier README détaille les coordonnées et les règles auxquelles sont soumis les paquets. Les responsables thématiques sont les garants du respect des règles (blocs Python d'installation, de désinstallation et de mise à jour pour tous les paquets) et de la bonne intégration des paquets proposés par la communauté dans le dépôt RESINFO. Créer un paquet WAPT demande du temps et de la rigueur, mais à la fin c'est magique : un clic, ça s'installe et ça fonctionne !

Ces étapes sont indispensables dans le processus de création des paquets métiers avant d'être proposés pour une intégration. Après validation et vérification par les responsables thématiques de RESINFO, ceux-ci sont mis à disposition pour la communauté au travers du magasin associé au serveur WAPT centralisé.

Jusqu'à maintenant, les contributeurs de ce projet sont les membres du groupe. Nous savons aussi qu'au niveau national, de nombreux sites réalisent, parfois, d'une manière très organisée, des paquets pour leurs propres besoins. Nous mettons ici le doigt sur notre deuxième objectif, le plus ultime, qui consiste à combattre pour un principe, celui de ne pas constamment réinventer la roue. Nous invitons donc tous ceux qui le souhaitent à nous contacter pour nous rejoindre et devenir non seulement des consommateurs, mais aussi des contributeurs pour ce projet.

Description de la cargaison, analyse d'un paquet

Examinons maintenant la structure de nos paquets et leurs singularités. Un paquet WAPT est en pratique une archive au format *Zip* qui contient plusieurs fichiers. Les fichiers les plus importants, ceux qui doivent être modifiés, sont les suivants :

- le fichier *Control* permet d'indiquer l'identité du paquet, sa version, sa description et le mainteneur du paquet, mais également ses dépendances (s'il y en a) ou ses incompatibilités ;
- le fichier *Setup.py* est le script en Python d'exécution de WAPT, la bibliothèque *setup-helper* en simplifie grandement l'écriture ;
- le fichier *Wapt.psproj* est l'équivalent d'un *Makefile* pour *pyScripter*.

Lorsque l'éditeur du logiciel le propose, il est très avantageux d'utiliser une source d'installation au format *MSI*, ceux-ci sont plus simples à compiler. Pour rappel, les options des fichiers *MSI* sont éditables sous Windows grâce à l'utilitaire *MSIEXEC* [28]. L'instruction *msiexec /?* dans l'invite de commande affiche tous les arguments et options disponibles permettant de personnaliser le déploiement ou la désinstallation du fichier.

Pour les fichiers d'installation de type *setup.exe*, le procédé est plus compliqué car les arguments d'installation silencieuse et de désinstallation ne sont pas normalisés (c'est cependant souvent */S* ou */silent*). Il faut parfois aller à la pêche sur internet pour les retrouver, par exemple :

```
setup.exe /silent /acceptcli /folderinstallation
```

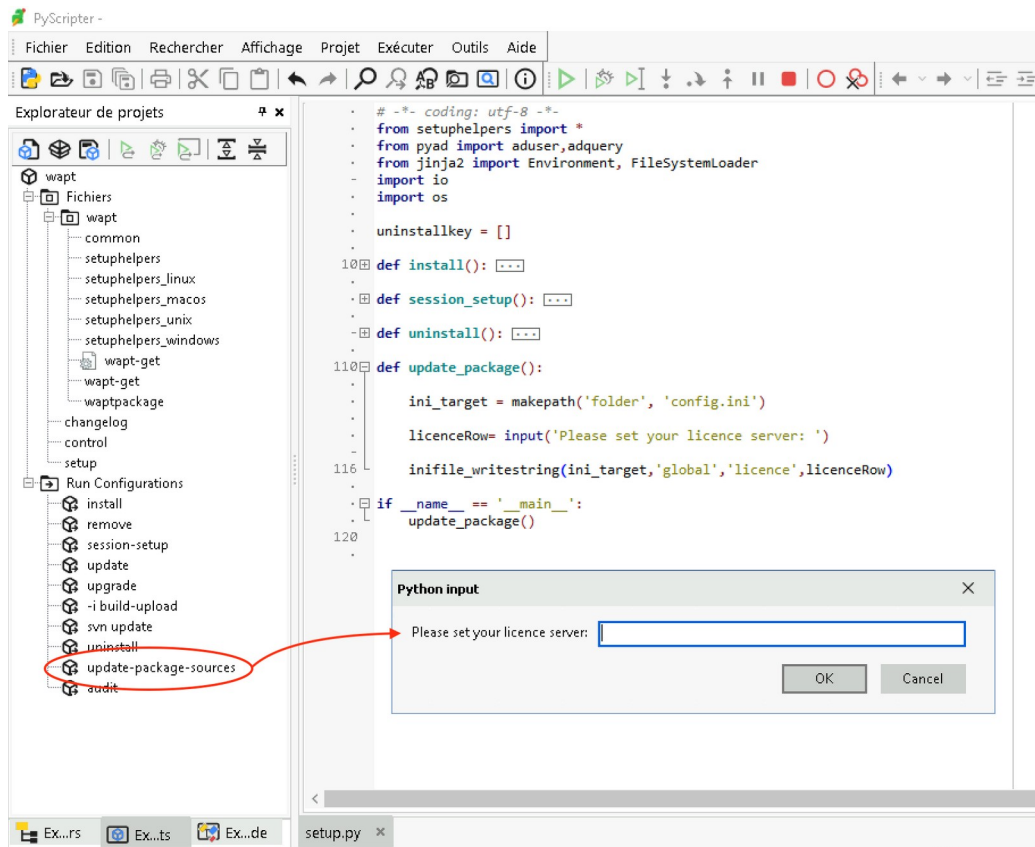


Figure 4 : Personnalisation du paquet via l'éditeur pyScripter - Ajout d'un numéro de licence.

Un autre point saillant de notre méthodologie concerne la gestion des licences. Celles-ci doivent être ajoutées dans le fichier de configuration associé au logiciel à déployer. Une fois le paquet logiciel importé dans votre dépôt local, l'éditeur *pyScripter* permettra d'exécuter dans le bloc *Run-Configuration* la commande « *update-package-source* » (figure 4). Une boîte de dialogue s'ouvre et demande de fournir les informations de personnalisations souhaitées (numéro de licence, ressources internes...). Le dépôt central RESINFO n'héberge aucune licence métier, mais uniquement les recettes de fabrication.

Voici un exemple de code pour les blocs installation, désinstallation et mise à jour du fichier *setup.py* qui permet une gestion par site d'un serveur de licence :

```
def install():
    version = control.version.split('-',1)[0]
    install_exe_if_needed(« BUZZ-Setup-%s.exe » % version, '/S')

def uninstall():
    run(r"%s » /S' % makepath(setup_path, r'Uninstall.exe'))

def update_package():
    ini_target = makepath("folder", 'config.ini')
    licenceRow= input('Please set your licence server: ')
    inifile_writestring(ini_target, "global", "licence", licenceRow)
```

Conclusion

Nous pourrions continuer notre voyage vers l'infini et au-delà du raisonnable en abordant d'innombrables détails passionnants de notre mission. Mais nous ne voulons pas dépasser la frontière qui nous a été imposée.

Le groupe de travail WAPT RESINFO a posé les bases du lancement au niveau national d'un projet permettant à chaque unité de l'enseignement supérieur et de la recherche de bénéficier de paquets prêts à l'emploi pour le déploiement d'applications scientifiques à travers l'application WAPT. Pour construire notre navette, nous nous sommes appuyés sur plusieurs partenaires. Le collectif des membres du GT tient grandement à remercier RENATER, le GDS Mathrice et le CC IN2P3 pour leur soutien et l'hébergement de l'ensemble de nos activités. Sans nos partenaires, nous ne serions pas en mesure de vous présenter ce projet dans cet état d'avancement. Comme nous l'avons décrit, nous avons pour ambition de mutualiser nos pratiques afin de mettre en commun nos compétences au service de la communauté. Nous pensons que notre investissement, qui disons-le, contribue dans un premier temps à nos propres besoins, prodiguera à beaucoup d'autres un gain de temps considérable lié à la gestion de leurs parcs informatiques. Nous pensons par exemple à tous nos collègues informaticiens isolés ou en sous-effectif dans leurs unités qui apprécieront cette aide à la gestion de leurs parcs. Plus que de la description d'un simple dépôt, c'est de l'émergence d'une constellation d'informaticiens qui met en commun ses compétences que nous avons voulu mettre en évidence. Comme nous vous l'avons proposé, nous vous invitons au voyage. Vous pouvez nous rejoindre et participer à cette belle aventure en intégrant le groupe pour nous faire profiter de vos réalisations. Aujourd'hui, la structure est en place, les moyens matériels sont là. Il ne reste plus qu'à alimenter le dépôt pour proposer davantage de paquets.

Licence

Cet article est soumis à la licence libre Art Libre [29]. Avec la Licence Art Libre, l'autorisation est donnée de copier, de diffuser et de transformer librement les œuvres dans le respect des droits de l'auteur. Cette licence ressemble dans l'esprit à la licence GPL sur les logiciels.

Bibliographie

- [1] WAPT : [https://fr.wikipedia.org/wiki/Wapt_\(logiciel\)](https://fr.wikipedia.org/wiki/Wapt_(logiciel))
- [2] Site de l'éditeur de WAPT : <https://www.tranquil.it/gerer-parc-informatique/decouvrir-wapt/>
- [3] GT-WAPT RESINFO : <https://resinfo.org/les-groupes-de-travail-11/groupe-de-travail-wapt/>
- [4] Les nouveautés 2021 dans WAPT : <https://www.it-connect.fr/wapt-enterprise-2-1-et-wapt-discovery-quelles-sont-les-nouveautes/>
- [5] Groupe Logiciel – Cellule Nationale Logicielle : <https://www.glesr.fr/>
- [6] Nathalie VIEIRA – WAPT – Solution de déploiement logiciel sous licence GPLv3 (ou comment dire au revoir à SCCM) : <https://2017.jres.org/fr/presentation?id=171>, https://conf-ng.jres.org/2017/document_revision_2932.html?download (PDF)
- [7] Florent FARENEAU – WAPT ou le déploiement de logiciel facilité : https://conf-ng.jres.org/2019/document_revision_5005.html?download#:~:text=WAPT

%20La%20solution%20WAPT%20permet, mises%20 %C3 %A0 %20jour%20des
%20logiciels.

- [8] Les NewsLetters de RESINFO : <https://resinfo.org/les-newsletters-de-resinfo/>
- [9] RESINFO : <https://resinfo.org>
- [10] Plateforme Rendez-Vous de RENATER : <https://www.renater.fr/fr/RENDEZ-VOUS>
- [11] Etherpad de l'IN2P3 : <https://www.in2p3.cnrs.fr/fr/portail-des-outils-collaboratifs>
- [12] Listes de diffusion RESINFO : <https://resinfo.org/notre-metier-asr/article/listes-de-diffusions-pour-les-asr>
- [13] Chat IN2P3 : <https://chat.in2p3.fr/>
- [14] Authentification eduGAIN : <https://edugain.org/>
- [15] Liste de diffusion RESINFO WAPT : <https://listes.resinfo.org/www/info/wapt>
- [16] Gitlab RESINFO WAPT : <https://gitlab.in2p3.fr/resinfo-gt/wapt>
- [17] PLMLab – La forge by Mathrice : <https://plmlab.math.cnrs.fr/>
- [18] OpenStack : <https://www.openstack.org/>
- [19] GDS Mathrice (Groupement de Service) : <https://www.mathrice.fr/>
- [20] Les piliers de la création : [https://fr.wikipedia.org/wiki/Piliers_de_la_cr%C3%A9ation_\(astronomie\)](https://fr.wikipedia.org/wiki/Piliers_de_la_cr%C3%A9ation_(astronomie))
- [21] Ansible : <https://github.com/ansible/ansible>
- [22] Forge Gitlab de Mathrice pour le GT WAPT : <https://plmlab.math.cnrs.fr/resinfo/gt/wapt>
- [23] Gitkraken : <https://www.gitkraken.com/>
- [24] gitCMD : <https://git-scm.com/book/fr/v2/D%C3%A9marrage-rapide-La-ligne-de-commande>
- [25] Atom : <https://atom.io/>
- [26] VSCode : <https://code.visualstudio.com/>
- [27] pyScripter : <https://github.com/pyscripter/pyscripter>
- [28] MSIEXEC : <https://docs.microsoft.com/fr-fr/windows-server/administration/windows-commands/msiexec>
- [29] Licence Art Libre – <https://artlibre.org/>