

EPELFI

Marché de Maintenance applicative du système AMALFI

Demande de travaux ponctuels (EXTRAIT)

Résumé

Type de document	TMA-TP1-PTP
Référence	TMA-TP1-PTP#TMAC-4333
Version	1.9
Date	15/03/2016
Nom de fichier	TMA-8.2.8.1 Lanceur de batchs.docx
Titre	Demande de travaux ponctuels
Sujet	Prestation TP1 : Travaux ponctuels
Statut du document	Pour acceptation

Diffusion

Société	Nom	Fonction
EPELFI	Dominique Doyen	Responsable TMA
ASTEK	Ndongo Fall	Responsable TMA

Historique du document

HISTORIQUE DU DOCUMENT			
VERSION	DATE	AUTEUR	DESCRIPTION
1.0	26/01/2016	Pierre-Alain Keyser Michel Gomis	Version initiale
1.1	27/01/2016	EPELFI	Relecture
1.2	28/01/2016	Pierre-Alain Keyser Michel Gomis	Prise en compte des remarques
1.3	10/02/2016	EPELFI	RELECTURE
1.4	15/02/2016	Pierre-Alain Keyser	Prise en compte des remarques
1.6	25/02/2016	Pierre-Alain Keyser	Précisions apportées suite à la réunion du 25/02/16
1.7	07/03/2016	EPELFI	RELECTURE
1.8	11/03/2016	Pierre-Alain Keyser Michel Gomis	Prise en compte des remarques de l'Exploitation
1.9	15/03/2016	Ndongo FALL	Mise à jour du devis

TABLE DES MATIERES

TMA-8.2.8.1 Lanceur de batchs.docx	1
1 Description	4
1.1 Description de la demande à étudier	4
1.1.1 Titre.....	4
1.1.2 Description.....	4
2 Fonctionnalités cibles.....	4
3 Détail d'un agent	6
4 Modalité de vérification du fonctionnement des AGENTS.....	7
5 Détail du serveur.....	7
6 Protocole de communication.....	7
7 Batchs sélectionnés pour la version beta du lanceur de batchs	8
8 Contraintes pour les batchs	8

1 DESCRIPTION

1.1 Description de la demande à étudier

Les informations en italique qui suivent, décrivant la demande, sont extraites de Jira.

1.1.1 Titre

Lanceur permettant le passage des batchs à la demande en pré-production

1.1.2 Description

Passage des batchs à la demande en pré-production :

Constat

Actuellement, les batchs de pré-production sont lancés directement par le groupe PI via un poste d'exploitant suite à une demande écrite ou orale du groupe APP.

Le groupe APP ne peut donc pas lancer directement les batchs.

Demande

Il faudrait mettre en place un « lanceur » de batchs via une application (hors AMALFI).

Cette fonctionnalité pourrait permettre de:

- *Sélectionner le batch à lancer,*
- *Renseigner les paramètres attendus en entrée du batch (paramètres obligatoires ou facultatifs).*
- *Lancer le batch.*
- *Consulter les logs du passage de ce batch.*
- *Consulter les fichiers en sortie du batch (rapports du batch DGI, listes des personnes sans droit.....).*

Cette fonctionnalité ne doit pas être utilisable en production.

Cette fonctionnalité doit pouvoir lancer aussi bien les batchs AMALFI, RED FAC.....

2 FONCTIONNALITES CIBLES

Une architecture de type « maitre-esclave » est envisagée pour le lanceur de batch. La solution sera constituée :

- d'un serveur web, hébergé en zone ELFI sur le serveur hébergeant les batchs, connecté à différents agents installés sur les serveurs exécutant les batchs.

- d'un agent qui sera installé sur chaque serveur de batchs (un pour AMALFI, un pour Redevance-Facturation)

Afin d'éviter les problématiques de communication inter-plateforme. Un serveur sera déployé par plateforme. Les agents d'une plateforme ne se connecteront qu'au serveur associé à leur plateforme.

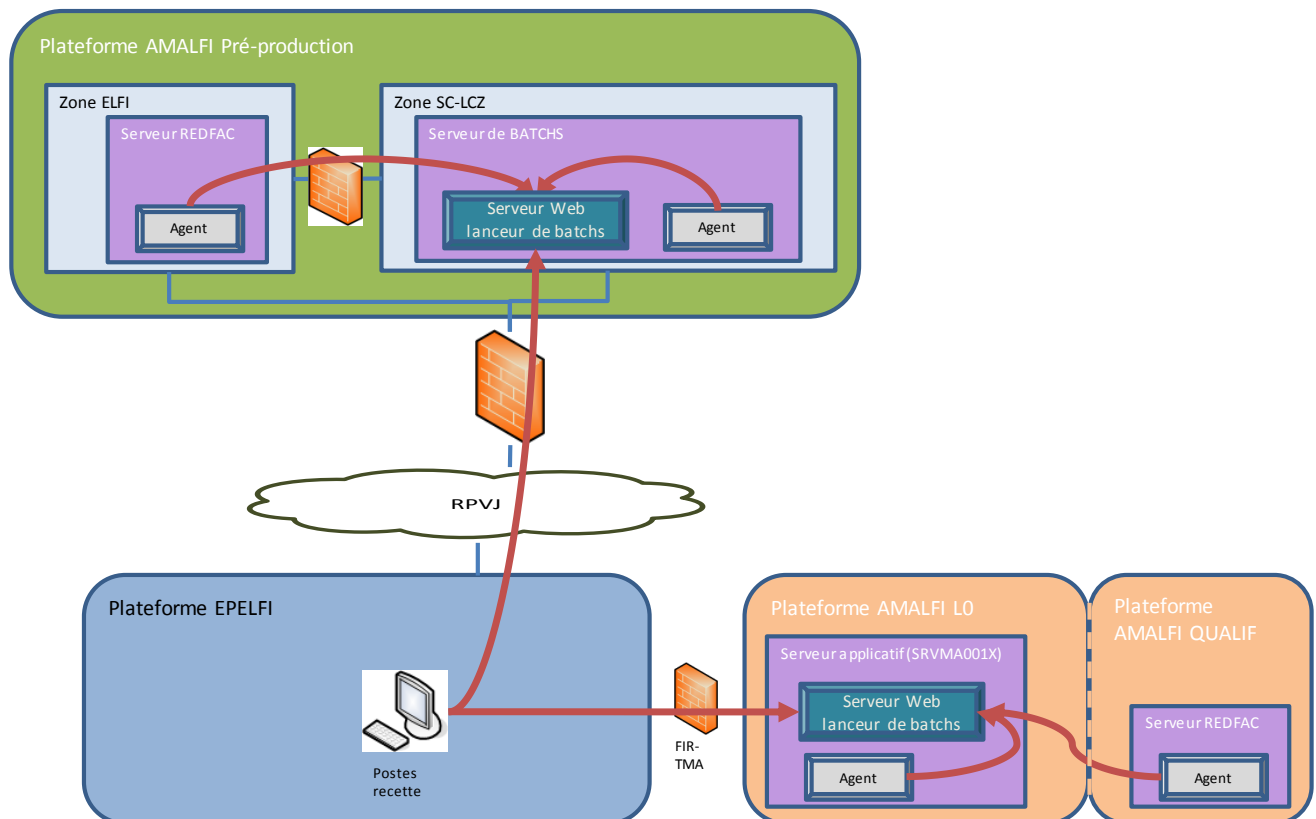


Figure 1 : schéma de connexion des agents au lanceur de batch

Le serveur web contrôlera les agents qui géreront les batchs, notamment en ce qui concerne la version et l'environnement cible. Les agents exécuteront les batchs et remonteront les fichiers de logs au serveur web. Les agents pourront être arrêtés par l'exploitant à tout moment.

Pour lancer les batchs, les agents n'utiliseront pas les fichiers d'exécution qui se trouvent dans le répertoire bin/. Ceci sera fait afin de contrôler plus finement les paramètres fonctionnels transmis au batch, minimisant ainsi les impacts sur les batchs existants. En effet actuellement, les batchs sont démarrés par un fichier « lanceur », celui-ci accepte un jeu d'argument plus petit que le jeu d'argument accepté par les batchs Amalfi. I.E. dans les batchs Amalfi, si on ne passe pas par le script « lanceur » tous les arguments du fichier de

configuration sont accessibles avec des variables système. Et les variables systèmes sont passables sur la ligne de commande d'un exécutable java en tant qu'argument de la JVM, c'est-à-dire avec une notation – Dcle=valeur. Il faut noter que les arguments de la JVM doivent être placés avant les arguments du programmes, ce que le script « lanceur » ne sait pas faire.

Les fichiers de logs, et fichier de rapports seront scrutés et remontés au serveur web qui les présentera en temps réel aux utilisateurs. Ainsi les utilisateurs pourront consulter la progression de l'avancement du batch.

Les données des X dernières exécutions seront conservés sur le serveur, puis automatiquement supprimées.

3 DETAIL D'UN AGENT

L'agent devra être capable de lancer des processus en générant les paramètres de lignes de commandes appropriés.

L'agent sera capable de filtrer les fichiers produits par les batchs (logs et rapport) en utilisant des expressions régulières. Il aura la responsabilité de scruter ces fichiers durant l'exécution du processus afin de les transmettre dès que possible au serveur web.

L'agent déterminera le statut d'un batch (réussite ou échec) en interprétant le code retour du processus correspondant au batch.

Un agent scannerera des répertoires configurés pour y détecter les versions et environnement des batchs. Le nom des répertoires contenant les batchs devra donc respecter une norme. Il existera plusieurs normes :

- Une spécifique aux applications redevance (cette norme est connue et est identique entre L0, qualification et pre-production)
- Une spécifique à l'environnement L0
- Une spécifique à la pré-production. Note : l'EPELFI devra transmettre un exemple de batch installé selon cette norme. Ainsi le prestataire pourra réaliser l'implémentation fonctionnant avec cette norme.

Les agents se connecteront au serveur pour recevoir les instructions de démarrage des batchs et pour pousser le statut de l'exécution en cours.

L'agent veillera à ce que les batchs ne puisse pas être lancé de façon concurrente, i.e. un batch dans une version et un environnement donné ne peut pas être exécuté s'il est déjà en cours d'exécution.

L'agent disposera d'une description des métadonnées des batchs permettant de préciser les paramètres du batch. Pour chaque paramètre sera précisé le nom, le type de données, le caractère obligatoire ou non, ainsi qu'une description fonctionnelle. Pour chaque batch sera positionné un "flag" de disponibilité, dans les paramètres de configuration de l'agent, indiquant si le batch peut être exécuté via le lanceur de batch ; la valeur de ce flag sera gérée par l'exploitation.

La procédure d'installation et de configuration d'un agent sera détaillée dans le rapport.

Les agents seront des applications java 1.6 compatible avec les JVM installées sur les machines Redevance-Facturation et sur les serveurs AIX hébergeant les batchs.

4 MODALITE DE VERIFICATION DU FONCTIONNEMENT DES AGENTS

Le fonctionnement lanceur de batch sera vérifié dans l'environnement L0, dont les normes de déploiement des batchs sont maîtrisées et bien connues du prestataire.

L'implémentation de la connexion au batch selon la norme de l'EPELFI sera réalisée à partir d'un exemple de batch transmis par l'EPELFI. Par conséquent le prestataire ne pourra pas tester l'implémentation de la connexion aux batchs avec la norme EPELFI sur tous les batchs.

La validation du lanceur de batch se fera en 2 étapes :

1. La première étape de validation du lanceur de batch se fera donc sur l'environnement L0.
2. Une fois celle-ci validée. Les configurations des agents relatives aux batchs pour la pré-production seront réalisées par le prestataire puis testé en pré-production sur les batchs. Dans le cas ou des ajustements seraient nécessaire le prestataire ira régler la configuration sur place.

5 DETAIL DU SERVEUR

Le serveur sera accéder par les utilisateurs en http à l'aide d'un navigateur web. Il permettra de :

- Consulter la liste des batchs installés, cette liste est construite à partir des données des agents.
- Exécuter un batch installé, en passant des paramètres
- Consulter les logs, rapports, et paramètres d'exécution des X dernières exécution. Il sera possible de télécharger les fichiers.

Le serveur sera une application REST non authentifié servie par spring-boot en java 8. Afin de pouvoir suivre la progression du batch, les fichiers de logs et autres rapports seront transmis sur le navigateur à l'aide de websockets.

6 PROTOCOLE DE COMMUNICATION

Le protocole de communication entre les agents et le serveur sera détaillé durant l'étude, mais devra à minima couvrir les fonctionnalités suivantes :

- Récupérer la liste des batchs d'un agent, notamment
 - Version, environnement du batch.
- Exécution d'un batch, en passant éventuellement des paramètres
- Remontée en « temps réel » des fichiers de logs, ainsi que les différents rapports durant l'exécution

Les échanges entre l'agent et le serveur seront bidirectionnels. La pile technologique pressentie pour l'implémentation de la communication est ZeroMQ (<http://zeromq.org/>), car elle permet d'échanger très facilement des messages, sans se soucier du sens de l'ouverture de la connexion TCP. Il est simple d'implémenter des patrons de messagerie comme le « publish-subscribe » sans avoir le besoin de passer par un broker de messages comme WebsphereMQ.

7 BATCHS SELECTIONNES POUR LA VERSION BETA DU LANCEUR DE BATCHS

Les batchs sélectionnés pour la version beta du lanceur de batchs sont :

- Batch envoi copie électronique, pour un batch AMALFI sans argument
- Batch Pacs/Pave, pour un batch AMALFI avec plusieurs arguments pour vérifier l'intégrité d'une requête, d'un bureau foncier, et de contrôler les vérifications verifScellementGreffes, verifContenu et verifSignature.
- Batch DGI pour vérifier la remontée des rapports d'un batch AMALFI, et un batch multi argument.
- Batch génération facture pour lancer un batch issu de Redevance-Facturation, avec un argument optionnel.

8 CONTRAINTES POUR LES BATCHS

Le rapport d'étude présentera :

- Comment les batchs reçoivent les paramètres.

FIN DU DOCUMENT