



Extension des services de l'agrégateur PMP

Version 1.1 – 15 février 2024

Historique du document

Version	Date	Authors	Changes
1.0	14.02.2024	Stéphane Spahni	Rédaction initiale
1.1	15.02.2024	Stéphane Spahni	Révision Daniel Vazquez
1.2	19.02.2024	S. Spahni / D. Vazquez	Clarification en lien avec le support d'autres communautés

Document	eMedication Primary Aggregator - Extensions de l'agrégateur PMP		
Auteur(s)	Stéphane Spahni & Daniel Vazquez Rivera, service cybersanté & télémédecine HUG		
Créé le	13.02.2024	Dernière modification	
Fichier	2024-02-19 Extensions PMP.docx	Version	1.2
Révision par	D. Vazquez Rivera	Date de révision	19.02.2024
Approuvé par		Date d'approbation	

1. Introduction

Ce document décrit trois extensions des services du PMP visant à le rendre plus facilement intégrable dans différents environnements et processus :

- Le support pour des patients enrôlés dans d'autres communautés ;
- Un service permettant de déterminer si un patient dispose ou non d'un PMP ;
- Un service permettant de savoir quand les dernières modifications sur un ensemble de PMPs ont été effectuées.

Toutes ces extensions services se basent sur des transactions IHE standards.

2. Support pour les patients d'autres communautés

L'architecture nationale prévoit d'avoir un agrégateur par communauté de référence et un agrégateur « proxy » par communauté simple.

L'implémentation actuelle ne tient pas compte du cross-communauté, ces travaux étant prévus ultérieurement sauf si une version centralisée revenait comme option possible. Il serait cependant utile de pouvoir dès le début du pilote accepter des patients enrôlés dans d'autres communautés et de permettre ainsi à l'agrégateur de gérer des patients de plusieurs communautés de référence.

Afin de réaliser cette fonctionnalité, les modifications suivantes sont apportées :

- Un identifiant patient associé à une instance du PMP est créé : le PMP-XAD-PID ;
- Les recherches et les publications de documents (y compris le document APPC) doivent se faire en indiquant l'identifiant PMP-XAD-PID du patient dans les métadonnées au lieu du XAD-PID CARA. Cet identifiant peut être récupéré via la transaction « PIXV3 Query [ITI-45] » décrite dans la section suivante. Cette modification est nécessaire et impacte tous les systèmes primaires connectés ;
- S'il ne connaît pas (encore) le PMP-XAD-PID, le système primaire qui crée le document APPC lors de l'enrôlement initial du patient doit faire au préalable une demande de création via la transaction Patient Identity Feed HL7 V3 [ITI-44] afin de forcer la génération du PMP-XAD-PID. Cet appel doit être ajouté aux composants permettant d'enrôler un patient dans le PMP.

L'implémentation dans le PMP de la transaction Patient Identity Feed HL7 V3 [ITI-44] est la suivante :

1. Recherche du patient dans le MPI de la communauté resp. à la CdC (Patient Demographics Query HL7 V3 [ITI-47] avec le numéro EPR-SPID), ce qui permet de valider le EPR-SPID ;
2. Création d'un identifiant patient dans le domaine « agrégateur PMP » (OID spécifique au PMP) si celui-ci n'existe pas encore pour cet EPR-SPID ;
3. Si le patient n'existe pas encore dans le MPI de CARA (pas de numéro XAD-PID dans la réponse du point 1, donc enrôlé dans une autre communauté de référence et jamais matché dans CARA) :
 - a. Création du patient dans le MPI de CARA (Patient Identity Feed HL7 V3 [ITI-44]) en documentant le numéro de patient PMP en lien avec l'EPR-SPID ;
4. Si le patient existe déjà dans le MPI de CARA (qu'il soit enrôlé dans la communauté ou déjà « matché » par un autre système primaire) :
 - a. Mise à jour du patient dans le MPI de la communauté (Patient Identity Feed HL7 V3 [ITI-44]) en ajoutant le numéro de patient PMP ;
 - b. Recherche du patient dans le MPI de la communauté (Patient Demographics Query HL7 V3 [ITI-47] avec le numéro de patient PMP) pour obtenir le numéro XAD-PID.

Suite à ces actions, le mini-MPI de l'agrégateur PMP contient les 3 identifiants : PMP-XAD-PID, EPR-SPID et XAD-PID CARA.

Grâce à ce mécanisme, les systèmes primaires connectés à d'autres communautés peuvent utiliser les services de l'agrégateur pour la partie PMP. La reconnaissance des utilisateurs affiliés à d'autres communautés fonctionne pour autant que l'agrégateur ait la connaissance non seulement du certificat utilisé par la communauté CARA pour signer les assertions XUA mais également de ceux utilisés par les autres communautés acceptées. Ces certificats étant publics, cela ne pose pas de problème de sécurité.

3. Existence d'un PMP

La récupération du contenu d'un PMP suppose que le professionnel se soit identifié avec son identification forte. Dans les premières années, le pourcentage de patients disposant d'un PMP sera très faible. Forcer le professionnel à effectuer une authentification forte pour finalement avoir la réponse dans la grande majorité des cas que le patient n'a pas de PMP sera particulièrement décourageant. La même question s'est posée par rapport à l'existence d'un DEP et une transaction librement accessible¹ peut être utilisée pour déterminer si un patient a un DEP ou non : la transaction « Patient Demographics Query HL7 V3 [ITI-47] » retourne le numéro EPR-SPID du patient, ce qui indique si celui-ci dispose d'un DEP ou non.

Le même principe s'applique au PMP :

Un appel à la transaction « PIXV3 Query [ITI-45] » avec comme paramètre un identifiant parmi :

- le numéro XAD-PID CARA du patient,
- le numéro PMP-XAD-PID du patient,
- le numéro EPR-SPID du patient

ainsi que l'OID du PMP comme « dataSource » et optionnellement l'OID du MPI CARA comme « dataSource » additionnelle. La transaction retourne alors l'identifiant patient du PMP (« PMP-XAD-PID ») si le patient est enrôlé dans le PMP (ainsi que le XAD-PID de CARA si demandé) ou « *Application Error* » sinon.

L'implémentation dans le PMP est la suivante :

1. Réception de la requête ITI-45 avec un identifiant reconnaissable par le PMP
2. Renvoi de l'information « Numéro patient PMP » (PMP-XAD-PID) vers le client (ou erreur si inexistant)

4. Gestion des notifications

Un certain nombre d'applications vont maintenir un « état » du PMP : c'est le cas par exemple de SwissMeds, qui ne va pas aller chercher le contenu actuel du PMP à chaque ouverture de l'app car cela demanderait à chaque fois de faire l'authentification forte. Afin de supporter au mieux ce type d'applications, un service d'abonnement à des notifications sera mis en place dans l'agrégateur du PMP. Ce service fonctionnera selon les principes suivants :

Back-end

L'application doit disposer d'un backend – il n'est pas possible d'avoir des interactions directement avec le PMP depuis une app mobile. Le backend doit disposer d'un certificat SecureNode (qui lui permet d'appeler

¹ Seule contrainte : disposer d'une connexion « secure node ».

aussi bien les services du DEP que du PMP). Pour accéder aux notifications, le backend doit également disposer d'un compte « TCU » (utilisateur technique sous la responsabilité d'un professionnel de santé).

Abonnement

Le patient doit s'abonner (i.e. abonner le backend de son application) au service de notification. Cela se fait en donnant un droit d'accès à l'utilisateur technique (TCU) du backend de son application. Option : création d'une policy spécifique à la gestion des notifications (ne donnant accès qu'à la transaction utilisée) au lieu d'utiliser un droit d'accès classique.

Récupération

L'agrégateur PMP ne notifie pas proactivement les différents backends abonnés aux notifications : chaque backend doit appeler à intervalle régulier une transaction ad-hoc multi-patients avec les identifiants des patients pour lesquels il veut obtenir la date de dernière modification. La transaction retourne une liste d'identifiants de documents PMLC, la date de chaque document PMLC indiquant la date de dernière modification du plan du patient concerné. Seuls les identifiants des PMLC des patients ayant autorisé le TCU dans leurs droits sont retournés. La récupération des différents documents PMLC suppose une authentification forte soit d'un professionnel soit du patient lui-même (aucun accès au document avec l'identité du TCU).

Implémentation technique

Aucun service de notification correspondant aux besoins n'existe actuellement dans les profils IHE : une solution ad-hoc respectant la philosophie globale a donc dû être trouvée. Après évaluation des différentes possibilités, le choix s'est porté sur la transaction « Multi-Patient Stored Query [ITI-51] » qui offre de nombreux avantages :

- La spécification de la transaction correspond aux besoins (multi-patients, retour d'information minimal) ;
- Son utilisation est similaire à celle d'autres transactions déjà utilisées (ITI-18, PHARM-1) ;
- Elle n'est pas encore utilisée dans le cadre du DEP, et donc n'a pas de règles d'accès pré- établies.

Une version RESTful n'est par contre pas encore disponible. Un contact avec le domaine ITI de IHE devra être pris pour voir l'évolution future de cette transaction dans le monde FHIR / RESTful.

L'implémentation de la gestion des notifications se fait de la manière suivante :

- L'ajout du TCU dans le document APPC est du ressort de l'application elle-même. Il n'existe pas de transaction au niveau du PMP pour récupérer la liste des TCUs car une application ne devrait pas permettre au patient d'abonner d'autres applications aux notifications (donc pas d'utilité d'une transaction retournant la liste des TCUs connus). Une action « s'abonner aux notifications », qui va ajouter la règle ad-hoc dans le document APPC, doit donc être implémentée dans le client. Cette fonctionnalité doit être mise à disposition du patient lorsqu'il a fait son authentification forte (dans le module de gestion des droits d'accès).
- Chaque TCU doit être configuré dans un truststore ad-hoc de l'agrégateur du PMP. L'ajout manuel d'un TCU suppose l'existence d'un contrat entre l'agrégateur PMP et le fournisseur du back-end.
- La récupération des notifications par le back-end se fait via la transaction « Multi-Patient Stored Query [ITI-51] » en utilisant la requête « FindDocumentsForMultiplePatients ». Les autres requêtes de cette transaction ne sont pas supportées.
 - Paramètres : liste d'identifiants patient (identifiants PMP) pour lesquels le back-end souhaite avoir la date de dernière modification du PMP. Option : différencier, via le paramètre EventCodeList, le type de modification dans le PMP (p. ex. ajout d'ordonnance, dispensation, etc.).
 - Résultats : liste de XSDDocumentEntry, chacun décrivant le document PMLC du patient concerné. La date de modification du document (« creationTime ») indique la date de dernière modification du PMP du patient concerné.

Note :

Une implémentation plus proche de la requête « Find Document Lists [ITI-66] » aurait également été possible mais aurait retourné beaucoup plus d'informations (typiquement la liste de tous les documents ajoutés à chaque PMP dans un intervalle de dates, avec leurs métadonnées). Cette option n'a pas été choisie afin de limiter au strict minimum les informations retournées à un utilisateur technique.