
Article

[Lorenzo Scalese](#) · Jan 9, 2023 109m de lecture

Nouvelles extensions SDA de HealthShare

Création et utilisation des nouvelles extensions SDA pour le stockage d'éléments de données personnalisés

Dans HSCore 15.01, il existe une nouvelle façon de stocker les éléments de données personnalisés. HealthShare peut désormais utiliser des extensions personnalisées pour de nombreux éléments SDA.

Cet article a pour but de :

1. Montrer comment configurer votre système pour utiliser les extensions SDA.
2. Créer une nouvelle propriété d'extension SDA
3. Utiliser la nouvelle propriété d'extension SDA dans les transactions HL7
4. Interagir avec les nouvelles données
5. Montrer la nouvelle extension SDA utilisée dans une personnalisation du Rapport de résumé du patient (Patient Summary Report).

Note: Pour cet article, je me sers de build :

HS-2016.1.1.108.0-hscore15.01_hsaa15_hspi15_hsvviewer15.01_linkage15-b2136-win_x64
J'ai également créé le système en utilisant "d ##class(HS.Util.Installer).InstallBusD
emo()"

Configurez votre système pour les extensions SDA

Cette section décrit comment configurer les extensions SDA pour un environnement HSCore 15.01.

Création d'un nouvel espace de noms

Dans le cadre des nouvelles extensions SDA, le nom de l'espace de noms personnalisé doit être HSCUSTOM.

Vous pouvez l'ajouter en allant dans : Management Portal->System Administration->Configuration->System Configuration->Namespaces (Portail de gestion->Administration du système->Configuration->Configuration du système->Espaces de noms).

Étapes pour créer un nouvel espace de noms :

1. Cliquez sur le bouton "Create New Namespace" (créer un nouvel espace de noms).
2. Entrez HSCUSTOM dans le champ " Name of the namespace " (Nom de l'espace de noms) (obligatoire).
3. Sélectionnez le bouton "Create New Database" (Créer une nouvelle base de données)

- Saisissez HSCUSTOM dans le champ "Enter the name of your database" (Entrez le nom de votre base de

données).

- Pour le champ " Database directory " (Répertoire de la base de données) :
 1. Cliquez sur le bouton "Browse...".
 2. Créez un nouveau dossier/répertoire, j'ai nommé mon répertoire "HSCUSTOM", il se trouve dans le répertoire " mgr " .
 3. Cliquez sur le bouton " OK "
- Cliquez sur le bouton " Next "
- Acceptez les valeurs par défaut, et cliquez sur le bouton "Next".
- Créez une nouvelle ressource appelée %DBHSCUSTOM et octroyez-lui des droits de lecture et d'écriture
- Cliquez sur le bouton "Finish" (terminer).

4. De retour à l'écran "New Namespace" (nouvel espace de noms), cliquez sur le bouton "Save" (enregistrer).

Cette procédure a pour effet de créer un nouvel espace de noms HSCUSTOM avec tous les mappings par défaut.

Exportation du paquet HS.Local

Une des choses que nous devons faire est de copier les classes et le code de la base de données HSLIB vers la base de données HSCUSTOM.

Vous pouvez le faire de plusieurs façons. Je vais vous parler de la façon de le faire à partir de Studio ou de Terminal.

Exportation à partir de Studio :

1. Connectez-vous à Studio
2. Changez l'espace de noms en espace de noms HSLIB (remarque : ceci peut être fait à partir de n'importe quel espace de noms qui a un paquetage HS mappé à HSLIB)
3. Allez dans le menu "Tools->Export" (outils - exportation).
4. Cliquez sur le bouton "Add" (ajouter)
5. Sélectionnez le dossier HS/Local, sélectionnez tout et cliquez sur le bouton "Open" (ouvrir).
6. Cela va tout sélectionner
7. Sélectionnez un fichier local ou distant pour exporter ces classes
 - Dans cet exemple, j'ai nommé le fichier "HSLocal.xml"

8. Cliquez sur le bouton "OK".

Exportation à partir d'une session Terminal :

1. Connectez-vous à Terminal
2. Changez l'espace de noms en espace de noms HSLIB (remarque : ceci peut être fait à partir de n'importe quel espace de noms qui a un paquetage HS mappé à HSLIB)
3. `HSLIB>d $system.OBJ.Export("HS.Local*.cls","C:/InterSystems/Export/HSLocal.xml")`

Ajoutez une nouvelle cartographie de paquet

Le paquet "HS.Local" doit être référencé à partir du nouvel espace de noms HSCUSTOM. Lorsque HSCUSTOM sera créé, "HS" sera mappé vers HSLIB. Vous devrez ajouter "HS.Local" à l'espace de noms HSCUSTOM, car il est actuellement pointé vers HSLIB.

Vous pouvez le faire manuellement via le Management Portal (Portail de gestion) ou de manière programmatique.

Management Portal (Portail de gestion) :

1. Allez dans le Portail de gestion->Administration du système->Configuration->Configuration du système->Espaces de noms.
2. Recherchez HSCUSTOM dans la colonne des espaces de noms et sélectionnez le lien "Package Mapping".
3. Cliquez sur le bouton "New"
4. Sélectionnez HSCUSTOM dans le menu déroulant "Package Database Location" (location de la base de données de paquet)
5. Sélectionnez le bouton radio "Specify a new package " (Spécifier un nouveau paquet).
6. Entrez HS.Local dans le champ "Package Name" (Nom du paquet)

7. Cliquez sur le bouton "OK"

Par programmation :

Vous pouvez créer du code pour ajouter ce mappage à un espace de noms.

1. Déplacez-vous vers l'espace de noms %SYS
 - HSCUSTOM> ZN " %SYS "
2. Définissez la propriété de la base de données
 - %SYS> set tProperties("Database")="HSCUSTOM"
3. Créez le mappage
 - %SYS>w ##class(Config.MapPackages).Create("HSCUSTOM","HS.Local",.tProperties)

Remarque: Vous devrez le faire pour chaque espace de noms et instance qui est un espace de noms HealthShare, à l'exception des espaces de noms Library. Il est important d'avoir ces mappages pour que les autres espaces de noms puissent accéder au code HSCUSTOM à utiliser dans leur traitement, comme les applications telles que Patient Index et Health Insight.

Importation du paquet HS.Local

Maintenant que les paquets HS.Local pointent vers HSCUSTOM, vous pouvez déplacer les classes que nous avons exportées précédemment dans l'espace de noms HSCUSTOM.

Vous pouvez le faire de plusieurs façons. Je vais vous parler de la façon de le faire à partir de Studio ou de Terminal.

Importation à partir de Studio:

1. Connectez-vous à Studio
2. Changez l'espace de noms en espace de noms HSCUSTOM.
3. Allez dans le menu Tools->Import Local (Outils->Importer Local).
4. Sélectionnez le fichier que vous avez exporté
5. Appuyez sur le bouton "Open" (ouvrir).
6. Vous devriez voir toutes les classes cochées et l'option "Compile Imported Items" (Compiler les éléments importés) cochée.

7. Cliquez sur le bouton "OK".

Importation à partir d'une session Terminal :

1. Connectez-vous à Terminal
2. Changer l'espace de noms en HSCUSTOM
3. HSLIB>d \$system.OBJ.Load("C:/InterSystems/Export/HSLocal.xml", " ck ")

Résumé

Nous avons maintenant l'infrastructure pour les nouvelles extensions SDA personnalisées de HSCore 15.01. Nous avons les classes HS.Local définies dans une nouvelle base de données HSCUSTOM et nous avons tous les espaces de noms qui pointent vers la localisation appropriée.

Si vous avez plus d'une instance de cache, l'espace de noms HSCUSTOM et les mappages HS.Local doivent se trouver sur chaque instance qui exécute HealthShare.

Création d'une nouvelle extension SDA personnalisée

Maintenant que nous avons les éléments nécessaires, nous allons créer une nouvelle propriété personnalisée.

Nous allons commencer par créer une propriété personnalisée pour le Patient SDA (SDA du patient).

En regardant les annotations HL7, il semble que "Veterans Military Status" (statut militaire des vétérans), qui est le PID, pièce 27, n'est pas utilisé dans SDA, alors essayons de créer ceci comme notre extension SDA personnalisée.

Comme la pièce PID 27 est un champ d'entrée codé, nous allons montrer que les nouvelles extensions SDA personnalisées sont plus que la paire nom/valeur précédente, il s'agit maintenant d'un type de données plus complexe. Dans cet exemple, nous créons un type de propriété personnalisé.

Edit HS.Local.SDA3.PatientExtension.cls

Nous devons ajouter la nouvelle propriété à HS.Local.SDA3.PatientExtension.cls

1. Connectez-vous à Studio
2. Changez l'espace de noms en HSCUSTOM
3. Modifiez HS.Local.SDA3.PatientExtension.cls
4. Ajouter une classe personnalisée "Custom Class"
 - Cette classe représente un type de données complexe qui aura :
 - Champ de code
 - Description du Champ

5. Ajoutez la propriété VeteransMilitaryStatus

- Propriété VeteransMilitaryStatus En tant que CUSTOM.SDA3.CodeTableDetail.VeteransMilitaryStatus;

6. Compilez HS.Local.SDA3.PatientExtension.cls

7. Compilez la classe HS.SDA3.Patient

8. Compilez la classe HS.Registry.Patient

Cette propriété est maintenant disponible pour être ajoutée/modifiée/supprimée à partir du streamlet SDA.

Utiliser la nouvelle propriété d'extension SDA dans les transactions HL7

Avant de pouvoir utiliser la propriété d'extension SDA, nous devons créer une nouvelle classe personnalisée qui étendra la classe HS.Gateway.HL7.HL7ToSDA3. Ce code sera exécuté sur la passerelle EDGE.

Voici un exemple de code de la nouvelle classe personnalisée :

```
Class CUSTOM.Gateway.HL7.HL7ToSDA3 Extends HS.Gateway.HL7.HL7ToSDA3 [ Not ProcedureBlock ]
{
```

```

/// Méthode de rappel pour le traitement personnalisé du streamlet Patient
ClassMethod OnPatient()
{
    do ..write(cr_"<Extension>")
    set tvMScCode = $$$xml($g(^|d(s,27,1)))
    set tvMScDescription = $$$xml($g(^|d(s,27,2)))
    if tvMScCode="" {
        do ..write(cr_"<VeteransMilitaryStatus>")
        do ..write(cr_"_tvMScCode_")
        do ..write(cr_"<Description>_tvMScDescription_<
;/Description>")
        do ..write(cr_"</VeteransMilitaryStatus>")
    }
    do ..write(cr_"</Extension>")
    Quit
}

```

Mise à jour de Production de Edge Gateway Ensemble.

Modifiez l'opération : HS.Gateway.HL7.InboundProcess et changez le paramètre "HL7ToSDA3Class" pour utiliser la nouvelle classe que nous venons de créer.

Cliquez sur le bouton "Apply" (appliquer) pour enregistrer les modifications.

Nous utilisons le message HL7 suivant : (Notez que la pièce PID 27 a une valeur de "V^Veteran")

```

MSH|^~\&||HC6||||ADT^A04|||2.5
EVN|A04|20160711094500
PID|||STM123^^^HC6^MR||Bolton^George||19271014|M|||1 Memorial Drive^^Cambridge^MA^021
42|||||||028345081|||||||V^Veteran

```

Nous traitons maintenant ce HL7 sur une passerelle Edge Gateway.

Maintenant si nous regardons la trace, nous pouvons voir nos données dans le <Patient> SDA.

Interaction avec les nouvelles données

Maintenant que nous avons les données stockées dans SDA qui se trouvent sur le Edge (et le Registry (Registre)), nous pouvons les examiner de plusieurs façons.

Sur l'Edge, je peux exécuter le code suivant :

```

ClassMethod ListVeteransStatus()
{
    #dim tPatient as HS.SDA3.Patient
    set sql="Select ID from HS_SDA3_Streamlet.Patient"
    set statement = ##class(%SQL.Statement).%New()
    set tSC = statement.%Prepare(sql)
    quit:$$$ISERR(tSC) tSC
    set result = statement.%Execute()
    while result.%Next() {
        set tStreamletID = result.ID
    }
}

```

```

        set tSC = ##class(HS.SDA3.Container).LoadSDAObject(tStreamletID
, .tPatient)
        w !, tPatient.Name.GivenName_ " " _tPatient.Name.FamilyName
        w ": Veteran Status = " _tPatient.Extension.VeteransMilitaryStat
us.Description
    }
    Quit
}

```

Et j'obtiens ce qui suit :

```

HSEdge1>d ##class([sample class]).ListVeteransStatus()
George Bolton: Veteran Status = Veteran

```

On peut voir que nous utilisons la propriété "Extension.VeteransMilitaryStatus.Description" pour afficher les nouvelles données personnalisées.

Sur HSREGISTRY, nous pouvons consulter les éléments suivants :

```

HSREGISTRY>s oPat=##class(HS.Registry.Patient).%OpenId(1)
HSREGISTRY>w oPat.Extension.VeteransMilitaryStatus.Code
V
HSREGISTRY>w oPat.Extension.VeteransMilitaryStatus.Description
Veteran

```

Pour les informations sur les patients, nous pouvons voir que la propriété personnalisée est disponible non seulement dans le SDA qui est stocké sur la passerelle Edge Gateway, mais aussi dans le registre des patients "Patient Registry".

Utilisation d'un SDA personnalisé dans une personnalisation du rapport sommaire du patient "Patient Summary Report".

Ceci n'est qu'un aperçu rapide de la manière dont vous pouvez utiliser la nouvelle extension SDA dans un code personnalisé.

Dans cet exemple, nous allons créer un rapport sommaire du patient "Patient Summary Report".

J'ai créé une classe CUSTOM.Reports.Patient.Summary, je peux simplement ajouter un nouvel élément à afficher en utilisant le code suivant :

```

<item field="Extension/VeteransMilitaryStatus/Description" >
    <caption value="Military Status" style="padding:5px;"/>
</item>

```

Now, when we bring up the patient and view Patient Summary Reports we can see the following:

Summary

This article showed how to set up the environment to use the new SDA extensions. It showed how to populate the new property using HL7. We showed how to access the data. In this case, we were using patient data, so we were able to access it from both the Edge gateway and Registry. We also used this new information in an example using a custom patient summary report. Now that this data is in the SDA, it can be used by Health Insight and Patient Index as well as any other place that uses SDA.

[#Mise en correspondance](#) [#Studio](#) [#Terminal](#) [#HealthShare](#)

URL de la source: <https://fr.community.intersystems.com/post/nouvelles-extensions-sda-de-healthshare>