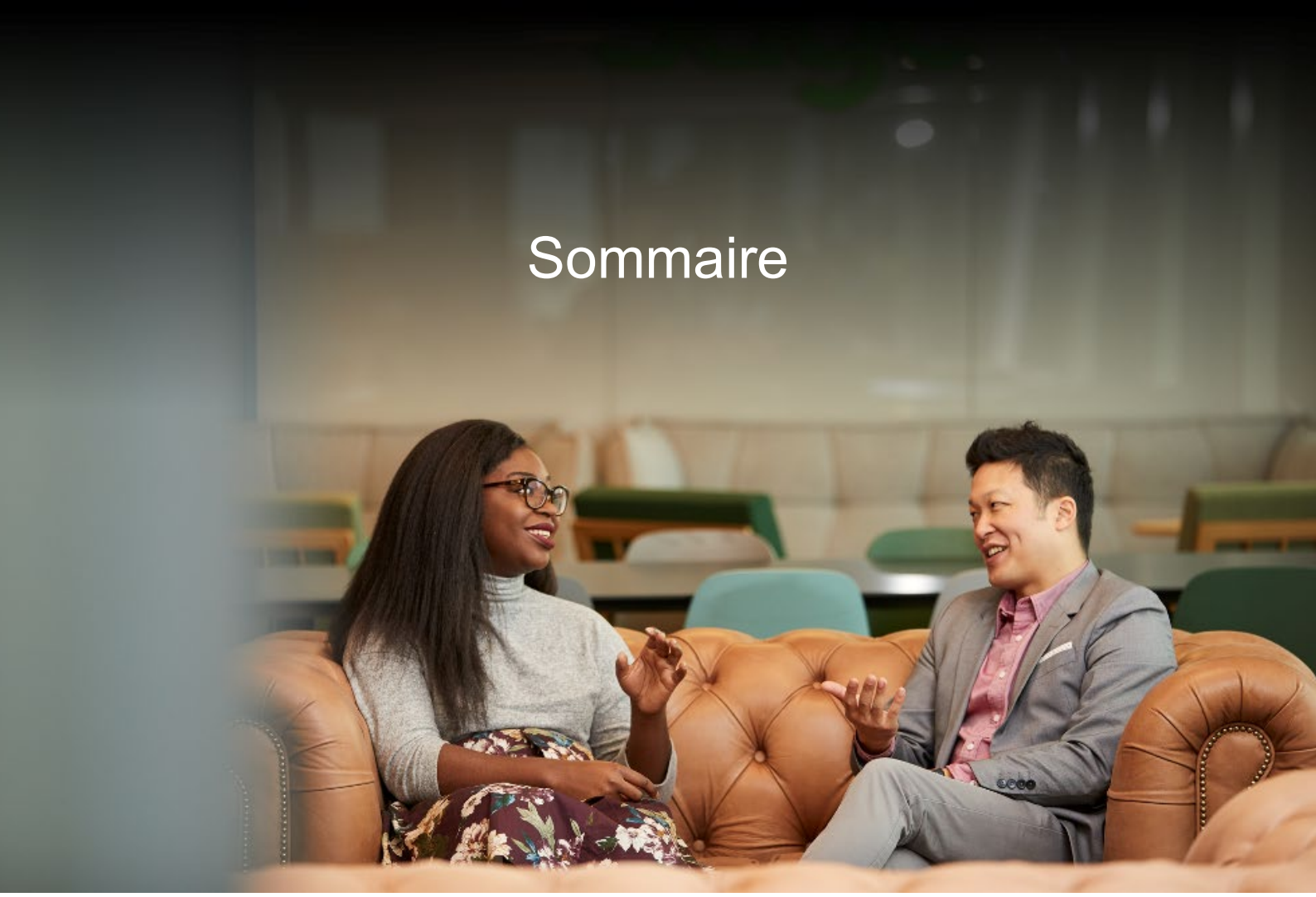
Sage XRT Common Services

Version 5.1

Guide Utilisateur

Sommaire



Installation

Présentation

La version 5.0 est la première version de **Sage XRT Common Services** à proposer les fonctionnalités les plus utilisées dans une interface web.

Seules persistent dans une interface *Win32* la gestion des tenants (ex. : **workgroups**) et la partie *DBInstaller*.

Sage XRT Common Services offre les fonctions :

- Gestion d'activation des données et politique des *4 yeux*
- Authentification *SAML V2*
- Support de *Crystal Report* 13.0.23
- API Rest de gestion de l'authentification
- API Rest de transformation des données (bibliothèque des formats)
- API Rest de gestion des paiements avec prise en charge du lien avec un service de Signature ou de Communication

Auxquelles, la version 5.1 ajoute :

- Authentification **CloudID**
- **Java Open Source**
- Améliorations du **DB Installer**
- API Rest de gestion de relevés bancaires (cf. documentation dédiée)

Recommandations

Sage XRT Common Services	Recommandations
Serveur	• Moteur de base de données : SQL Server 2016, SQL Server 2017 • Serveur Microsoft : IIS 10 ou supérieur • Version JRE 8 (64 bits) pour le bon fonctionnement de Sage XRT Bank Format Library • Activation de l'exécution du PowerShell : PS> Set-ExecutionPolicy -ExecutionPolicy RemoteSigned
Poste client	Version du provider SQL identique à celle du serveur

Important !

L'installation de **Sage XRT Common Services** sur chaque poste client n'est désormais plus nécessaire lors de l'installation des produits **Sage XRT Communication & Signature** et **Sage XRT Advanced**.

Sage XRT Common Services doit être installé uniquement sur le poste serveur.

Pour **Sage XRT Communication & Signature**, **Sage XRT Common Services** doit être installé sur la même machine que le serveur de communication.

Pour **Sage XRT Business Exchange**, tant que le *wrapper* **sage.fcs.client** n'est pas intégré, **Sage XRT Common Services** doit être installé sur la même machine que **Sage XRT Business Exchange**.

Lorsqu'un tenant est créé avant l'installation d'autre produit, la base de données de **Sage XRT Common Services** doit être mise à jour.

Déploiement

Configuration 1

Dans le cas où tout est installé sur la même machine, il est recommandé de suivre l'ordre d'installation suivant :

1. **Sage XRT Bank Format Library** 4.5.1
2. **Sage XRT Common Services** 5.0.6
3. **Sage XRT Treasury Communication** 4.2.0
4. **Sage XRT Treasury Signature** 4.2.0
5. **Sage XRT Advanced** 2.4.3 HF1

Vous devez utiliser :

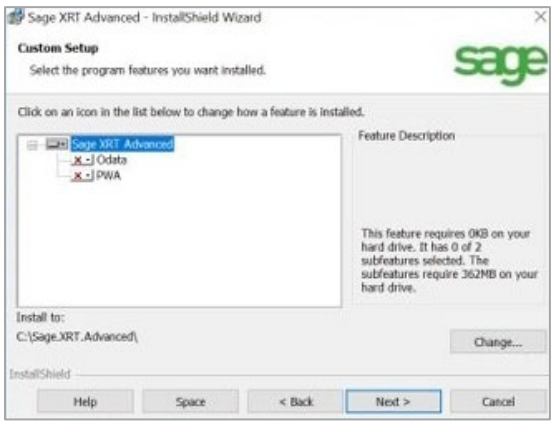
- La procédure d'installation **Complète** de **Sage XRT Common Services**
- La procédure d'installation **Complète** de **Sage XRT Communication**
- La procédure d'installation **Complète** de **Sage XRT Signature**

La base de données de **Sage XRT Common Services** doit être mise à jour avec le *DBInstaller*.

- La procédure d'installation **Custom** ou **Complète** de **Sage XRT Advanced**.

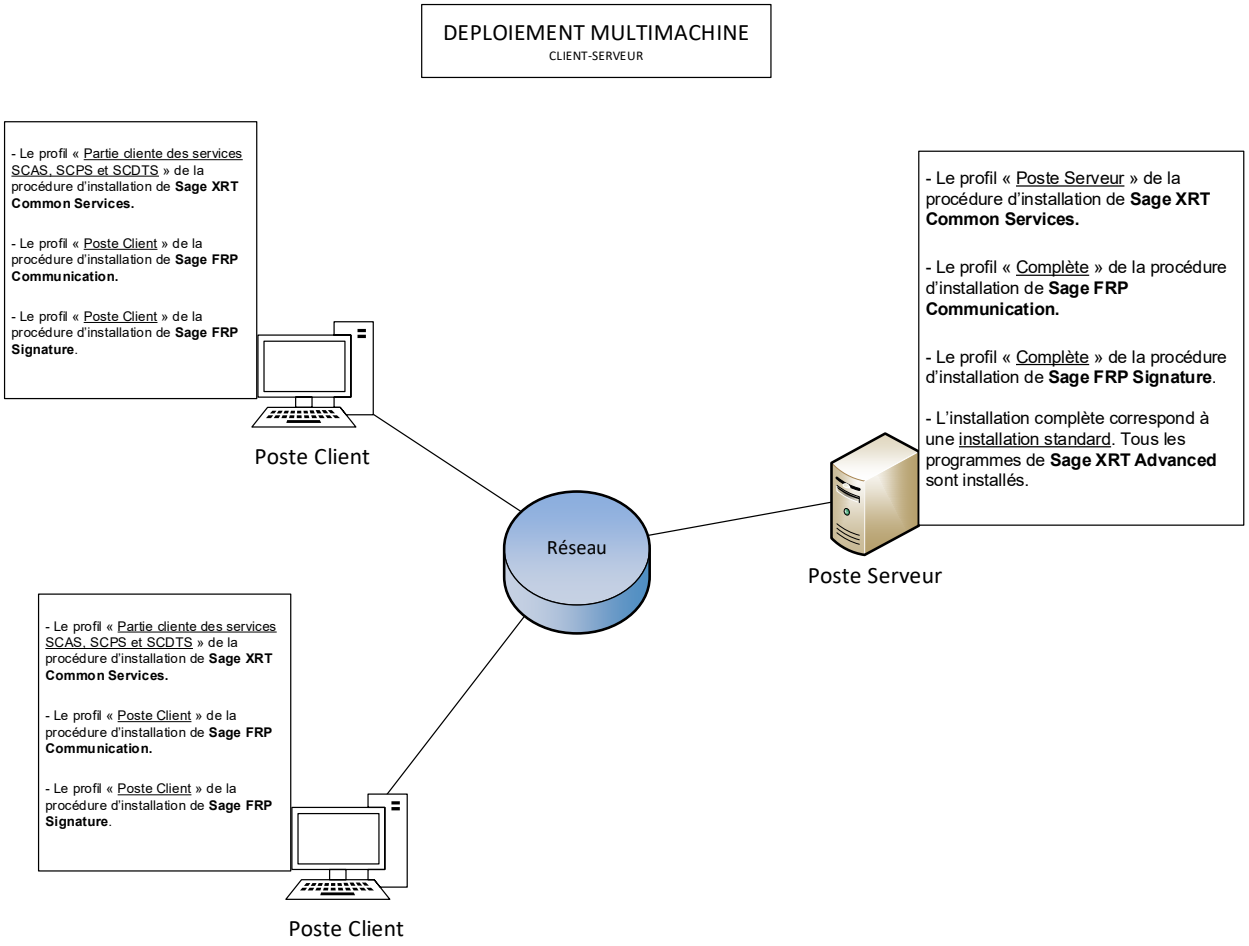
  L'installation **Complète** correspond à une installation standard où tous les programmes de **Sage XRT Advanced** sont installés.

  L'installation **Custom** ouvre la fenêtre suivante.



Configuration 2

Cette configuration concerne l'installation standard dite *on-premise* (sur site).



Dans le cas de 1 à n machine cliente et une machine serveur, sur la machine cliente, vous devez utiliser :

- La procédure d'installation **Partie cliente des services SCAS, SCPS et SCDTS** de **Sage XRT Common Services**
- La procédure d'installation **Poste Client** de **Sage XRT Communication**
- La procédure d'installation **Poste Client** de **Sage XRT Signature**

Et sur la machine serveur :

- La procédure d'installation **Poste Serveur** de **Sage XRT Common Services**
- La procédure d'installation **Complète** de **Sage XRT Treasury Communication**

- La procédure d'installation **Complète** de **Sage XRT Treasury Signature**
- La procédure d'installation **Complète** ou **Custom** de **Sage XRT Advanced**. L'installation complète correspond à une installation standard. Tous les programmes de **Sage XRT Advanced** sont installés.

La base de données de **Sage XRT Common Services** doit ensuite être mise à jour avec le *DBInstaller*.

Et sur la machine serveur :

- La procédure d'installation **Poste Serveur** de **Sage XRT Common Services**

Configuration 3

Dans le cas de 1 à n machines clientes et une machine serveur pour **Sage XRT Communication/Signature** et une machine serveur pour **Sage XRT Common Services**, sur la machine cliente, vous devez utiliser :

- La procédure d'installation **Partie cliente des services SCAS, SCPS et SCDTS** de **Sage XRT Common Services**
- La procédure d'installation **Poste Client** de **Sage XRT Treasury Communication**
- La procédure d'installation **Poste Client** de **Sage XRT Treasury Signature**

Sur la machine serveur de **Sage XRT Communication & Signature**, vous devez utiliser :

- La procédure d'installation **Poste Client** de **Sage XRT Common Services**
- La procédure d'installation **Complète** de **Sage XRT Treasury Communication**
- La procédure d'installation **Complète** de **Sage XRT Treasury Signature**

Sur la machine serveur de **Sage XRT Common Services**, vous devez utiliser :

- La procédure d'installation **Poste Serveur** de **Sage XRT Common Services**
- La procédure d'installation **Scripts de base de données** de **Sage XRT Common Services**
- La procédure d'installation **Scripts de base de données** de **Sage XRT Treasury Signature**

La base de données de **Sage XRT Common Services** doit ensuite être mise à jour avec le *DBInstaller*.

Configuration avec SXA et SXCS

Dans le cas d'une installation avec **Sage XRT Advanced** et **Sage XRT Communication & Signature**, suivez la procédure qui suit.

1. Activez *powershell* : *Set-ExecutionPolicy - ExecutionPolicy RemoteSigned -Scope LocalMachine*

2. Dans le fichier de config *Sage.SCDTSServer.Service.exe.config*, paramétrez l'URL du service **REST SXCS** pour la demande de statuts et activez le traitement : action=**YES**.

3. Paramétrez l'URL du service **REST SXCS** pour la demande d'ajout.

En cas de migration de **Sage XRT Common Services** 3.9 vers 5.0, l'utilisateur standard *XRT* pré-paramétré dans les fichiers de configuration des services n'existe pas encore. Vous devez soit le créer, soit modifier les fichiers de configuration en utilisant un nom d'utilisateur qui existe.

Exemple d'éléments mis en œuvre – Gestion des paiements depuis SXA

Workflow et traitements

Les paragraphes de ce chapitre présentent le workflow d'un paiement et ses traitements par produit, ainsi que les actions manuelles à effectuer.

Produit	Acronyme
Sage XRT Advanced	SXA
Sage XRT Common Services	SCS
Sage XRT Communication & Signature	SXCS

Sage XRT Advanced

L'utilisateur initie un paiement depuis la plateforme **SXA**.

SXA génère une demande au format *JSON* qu'elle transmet ensuite à **SCS** via le protocole **API Rest**.

SXA suit cette demande (toujours via **API Rest**) et tient informé en temps réel l'utilisateur de l'état de son paiement en fonction de son cheminement à travers les différentes étapes.

Sage XRT Common Services

Le service **SCDTS** reçoit la demande de paiement de **SXA** et fournit un ID de transaction à **SXA** pour les demandes de statuts postérieures. Par la suite, plusieurs étapes s'enchainent :

1. La conversion du flux *JSON* initial en fichier bancaire.
2. L'enrichissement des données initiales si l'option a été activée.

Ce fichier sera ensuite posté sur le module de **Signature**.

Important ! Une phase d'initialisation manuelle du flux des paiements est nécessaire par banque (session/connexion). Elle est réalisée en effectuant le premier envoi d'un paiement depuis **SXA**. Ensuite il faut modifier la correspondance dans la table de transcodage EXITPOSTGEN.

Dans le fichier *Sage.SCDTSServer.Service.exe.config,* vous devez effectuer les actions suivantes :

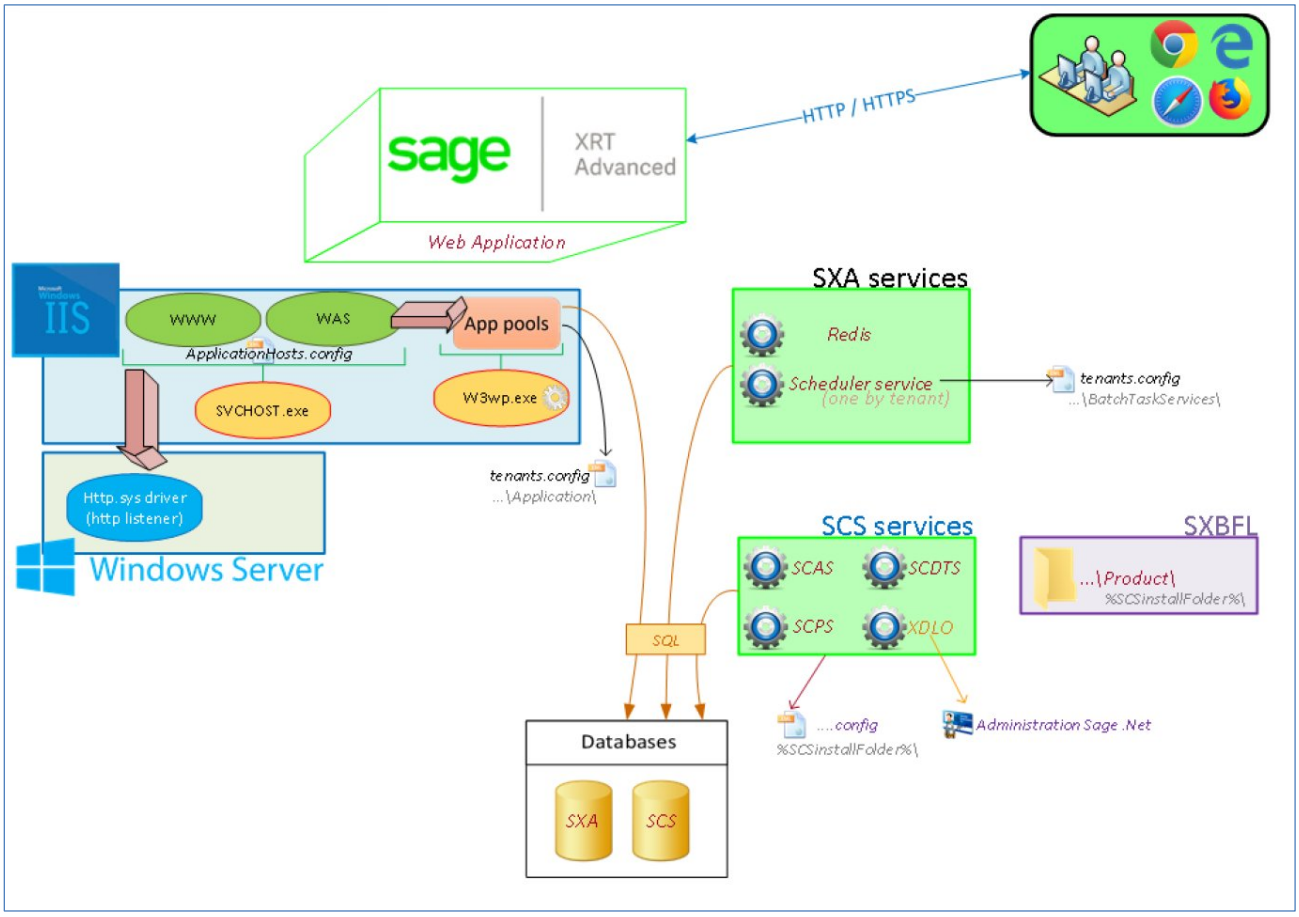
1. Paramétrer l'URL du service REST SXCS pour la demande de statut
2. Activer le traitement : action=YES.
3. Paramétrer l'URL du service REST SXCS pour la demande d'ajout.

Sage XRT Communication & Signature

Le service **ComSignAPI** prend en charge la demande en provenance de **SCDTS** pour son insertion dans le flux de travail correspondant au contrat sélectionné.

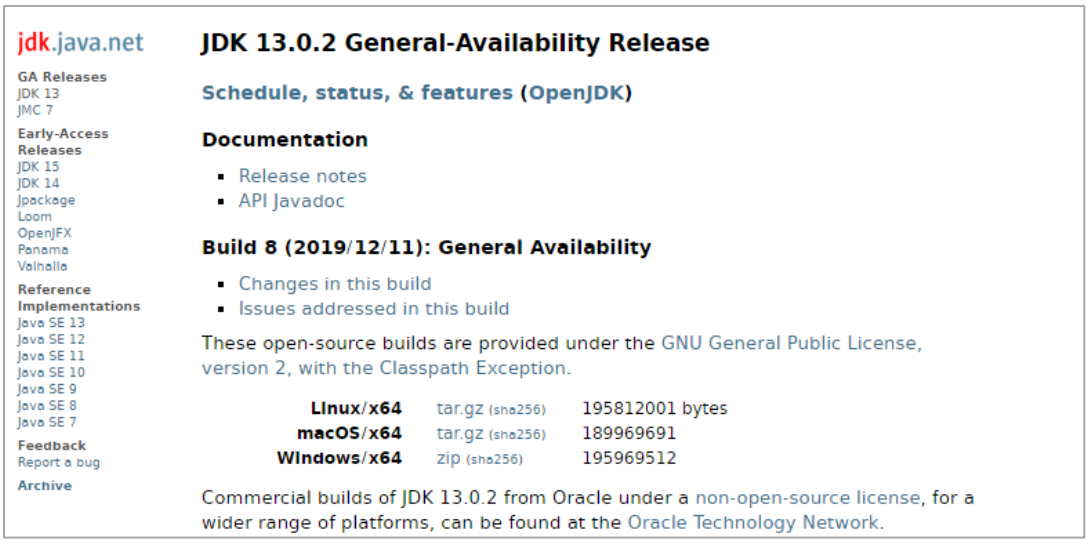
Important ! N'oubliez pas que sur le poste client dans une configuration multi-machine, vous devez paramétrer l'adresse du service LAD et les adresses des services **SCDTS**, **SCAS** et **SCPS** dans le fichier <*INSTALLPATH_DE_SCS_CLIENT*>*sage.fcs.client.dll.config*

Schéma d'échange des flux à partir de la solution d'initiation des Paiements de **Sage XRT Advanced**



Java Open Source

1. Désinstallez le programme *Java* existant.
2. Testez grâce à une ligne de commande la génération d'un fichier bancaire à partir d'un fichier intermédiaire. Ce test aboutit au message d'erreur : **jvm.dll required for Sun JDK**
3. Téléchargez le fichier *Windows/x64* zip de la dernière version disponible depuis le site https://jdk.java.net : *openjdk-13.0.2_windows-x64_bin.zip* (187 Mo)

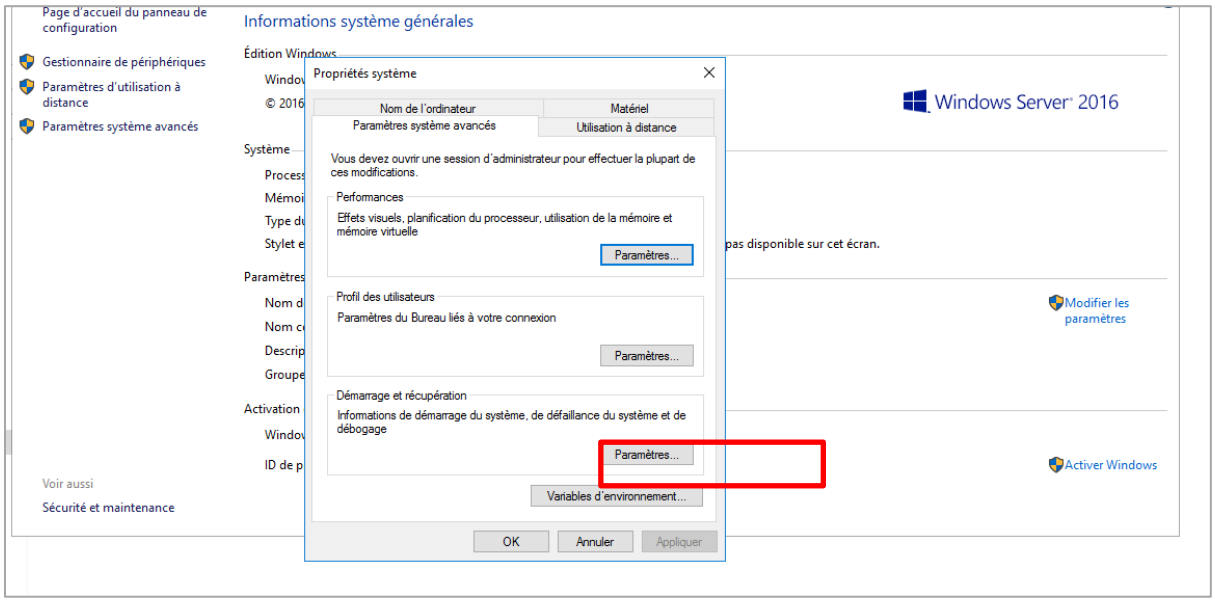


4. Créez un répertoire **JAVA** sous **C:\Programmes** et dézipper le zip téléchargé dans ce nouveau répertoire : un dossier *jdk* est généré.

5. Vous devez créer des variables d'environnement.

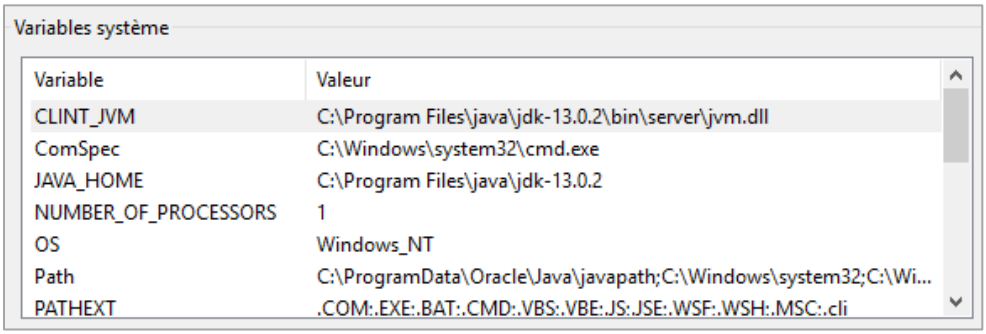
   Dans l'explorateur *Windows*, ouvrez par un clic droit sur **Ce PC** le menu contextuel puis cliquez sur **Propriétés**.

   Dans le panneau de gauche, sélectionnez **Paramètres système avancés**. Cliquez sur **Variables d'environnement …**



Dans la rubrique **Variables utilisateur pour Administrateur**, double-cliquez sur la variable **PATH**, puis sélectionnez **Nouveau** et ajoutez la variable **C:\Program Files\java\jdk-13.0.2\bin**

Dans la rubrique **Variables système**, cliquez sur **Nouvelle**. Créez la variable **JAVA_HOME** avec la valeur *C:\Program Files\java\jdk-13.0.2*.
Créez la variable **CLINT_JVM** avec la valeur **C:\Program Files\java\jdk-13.0.2\bin\server\jvm.dll**



6. A partir d'une nouvelle ligne de commande, testez la génération d'un fichier bancaire à partir du fichier intermédiaire : le message d'erreur a disparu.

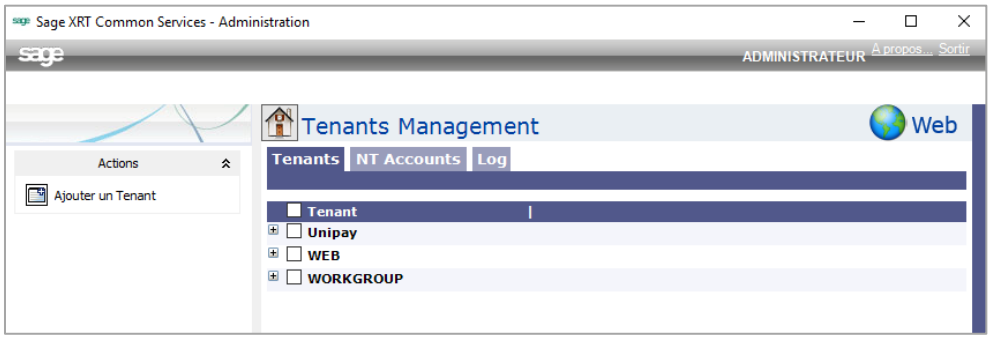
Interface Tenants Management

Tenants

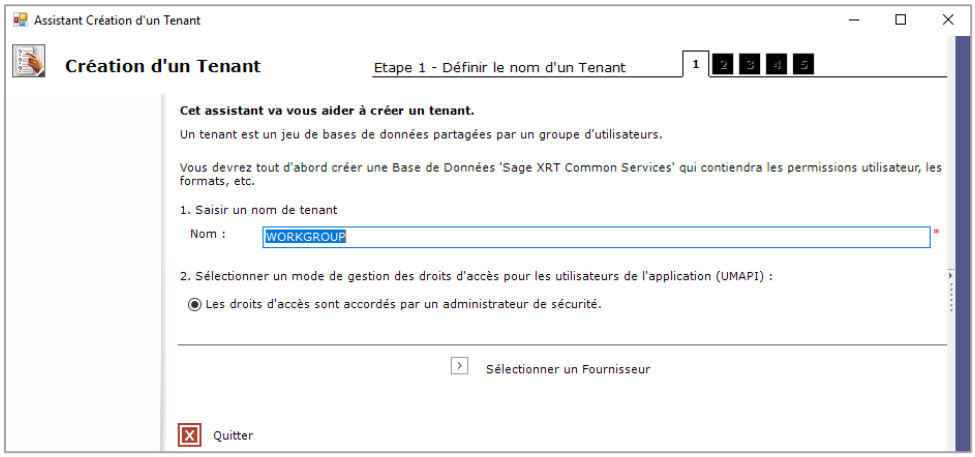
Dans le cas d'une première installation de **Sage XRT Common Services**, aucun tenant n'existe : la première étape consiste à créer un ou plusieurs tenants en fonction du type de base de données utilisée (*SQL Server* ou *Oracle*).

Création

Dans le menu **Démarrer**, rendez-vous dans **Programmes – Sage – Administration XRT .NET**. L'interface **Tenants Management** s'affiche.



Cliquez sur le lien **Ajouter un tenant** pour lancer l'**Assistant Création d'un Tenant**.



Note : Lorsqu'aucun tenant n'est défini, l'**Assistant Création d'un Tenant** s'affiche immédiatement à l'écran.

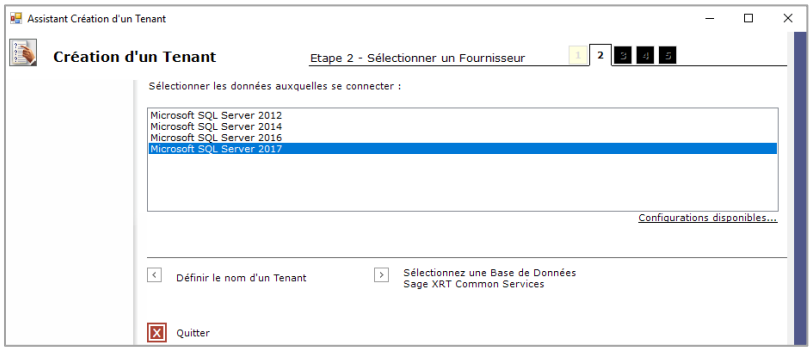
Nom

Saisissez un nom de tenant dans le champ **Nom**. Le nom par défaut est **WORKGROUP**.

Définissez le mode de fonctionnement de la gestion des droits d'accès des utilisateurs aux applications **Sage XRT**.

Note : A partir de la version 5.0, la validation des opérations par un administrateur de sécurité de niveau 2 a été remplacée par la fonctionnalité d'activation des données. Cette activation peut être requise ou non selon le paramétrage effectué.

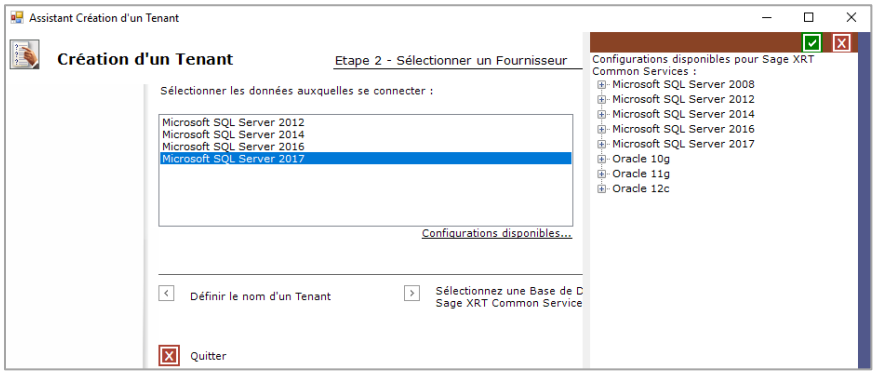
Cliquez sur le lien **Sélectionner un Fournisseur**.



Sélection du fournisseur

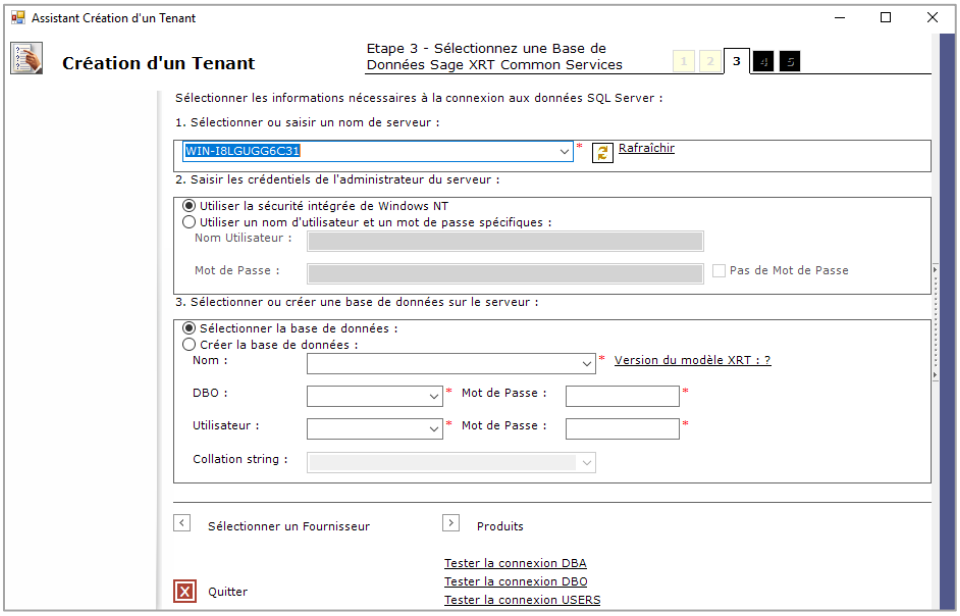
Sélectionnez dans la liste, le serveur ou client de base de données installé.

Cliquez sur **Configurations disponibles** pour visualiser le détail des serveurs ou clients de base de données installés sur la machine et les opérations admises (création et mise à jour).



Cliquez sur pour fermer la page et revenir à la page de sélection des fournisseurs d'accès aux bases de données.

Cliquez sur **Sélectionnez une Base de Données Sage XRT Common Services**.



Sélection de la base de données

Saisissez le nom du serveur sur lequel la base de données doit être créée. Les caractères possibles pour indiquer ce serveur sont :

- (local)
- (LOCAL)
- .
- *nom serveur*

Le bouton **Rafraîchir** permet d'obtenir la liste des serveurs *Microsoft SQL Server* connectés au réseau de l'entreprise.

Suivant le type d'authentification utilisé par le DBA (Administrateur de base de données) pour se connecter au serveur de bases de données, sélectionnez une des options qui suivent.

- **Utiliser la sécurité intégrée de Windows NT** : le DBA est authentifié grâce à son compte NT.
- **Utiliser un nom d'utilisateur et un mot de passe** : le DBA est authentifié grâce à un nom d'utilisateur et un mot de passe.

Note : Cliquez sur le lien **Tester la connexion DBA** au bas de la page pour vérifier les identifiants du DBA.

Vous avez ensuite deux possibilités :

- **Sélectionner la base de données** si vous souhaitez travailler sur une base de données existante.
    - Sélectionnez la base de données dans la liste déroulante. Les bases de données existantes sont actualisées au premier affichage de la liste.
    - Saisissez le mot de passe correspondant au nom du DBO (propriétaire de la base de données) affiché dans le champ **DBO**. Le mot de passe proposé par défaut par l'assistant est : **password#2005**. Lorsque vous sélectionnez une base de données dans la liste, l'assistant recherche automatiquement le nom du propriétaire de celle-ci en utilisant le compte DBA.
    - Saisissez le mot de passe du compte **XRTUSERS**. Le mot de passe par défaut est : **password#2005**

Note : Cliquez sur le lien **Tester la connexion DBO** en bas de la page pour vérifier les identifiants du DBO.

- **Créer la base de données** si vous souhaitez créer une base de données.
    - Saisissez un nom de base de données. L'assistant vérifie qu'aucune base ne porte ce nom lorsque vous cliquez sur **Créer/Modifier les modèles**.

Important ! Le nom de la base de données ne doit comprendre aucun espace, ni caractère spécial (*, ?, \, /, etc.).

    - Saisissez les identifiants du propriétaire de la base de données. L'assistant propose par défaut un compte avec l'identifiant **XRT** et le mot de passe **XRT**. Il crée, si nécessaire, le compte et lui affecte le rôle de **db_owner** sur la base.
    - Sélectionnez la **Collation string** (chaîne d'interclassement) ou laissez par défaut **French_CI_AS** (pas de distinction entre majuscule et minuscule).

Cliquez sur **Produits**.

Configuration des unités logiques

L'assistant propose par défaut un scénario dans lequel les tables *filegroup* **DATA** et les index *filegroup* **INDEX** du modèle sont créés dans le *filegroup* **PRIMARY** (*filegroup* par défaut lors de la création d'une base de données *SQL Server*).

Le panneau de propriétés vous permet de :

- Modifier le scénario proposé et installer les tables et les index dans deux *filegroups* différents (exemple : **XCS_DATA** et **XCS_INDEX**).

- Modifier les paramètres de création des *filegroups* (répertoire de stockage, taille initiale, taille limite, taux de croissance). Le répertoire de stockage doit exister pour que l'opération de création fonctionne correctement.

Important ! Les scripts modèle de **Sage XRT Common Services** font référence à un *filegroup* logique **DATA** pour les tables et un *filegroup* logique **INDEX** pour les index.

Lors de l'exécution des opérations de création du modèle, l'assistant remplace ces noms logiques par les valeurs saisies dans le panneau de propriétés (**PRIMARY** dans le cas du scénario par défaut).

Si les groupes de fichiers cibles n'existent pas (par exemple : **XCS_DATA** et **XCS_INDEX**), ceux-ci sont automatiquement créés par l'assistant.

Cliquez sur **Créer/Modifier les modèles**.

Création et modification des modèles

La liste intitulée **Scripts à exécuter** contient l'ensemble des scripts utilisés pour la création du modèle **Sage XRT Common Services** :

- **createlogicalunits.sql** : Script de création des unités logiques. Une unité logique représente un *filegroup* dans le cas de la création d'une base de données *Microsoft SQL Server*).

- **xl_configuration createxl_configuration.sql** : Script de création de la table dans laquelle est enregistrée la version du modèle

- **registerlogicalunits.sql** : Script d'enregistrement des unités logiques

Les scripts produit sont traités par la suite.

Suivant leur type, les scripts sont exécutés avec le compte du DBA ou du DBO.

Activez la case **Sélectionner les données à importer** et choisissez une langue dans la liste. Cette importation concerne les données (format *XML*) pour **APIFMT**, **TRANSCO** et **UMAPI**.

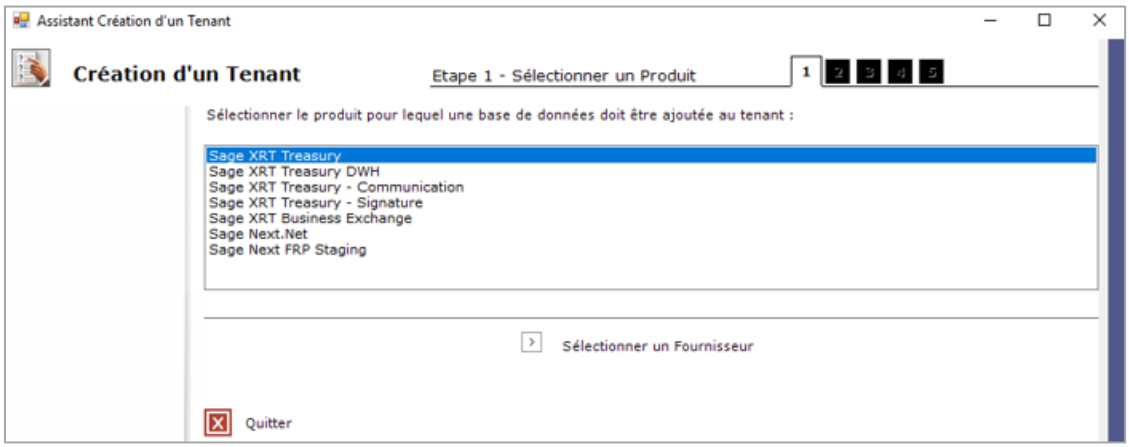
Cliquez sur **Valider toutes les étapes** pour procéder à l'exécution des opérations configurées dans les étapes 1, 2, 3, 4 et 5 de l'assistant.

Exécution des opérations

L'exécution peut durer quelques minutes. A ce stade, le modèle **XCS** est créé.

Vous pouvez quitter l'assistant **Création d'un tenant** en cliquant sur l'icône **Quitter** et lancer le service d'**Administration** ou ajouter un modèle de Produit.

Cliquez sur **Ajouter une base de données Produit**, l'assistant de création de tenant s'affiche.



Sélection du produit

Sélectionnez un produit dans la liste.

Cliquez sur le lien **Sélectionner un Fournisseur**.

Note : Pour plus d'informations sur la marche à suivre pour sélectionner un fournisseur, reportez-vous à la section intitulée **Sélectionner un fournisseur**.

Bases de données

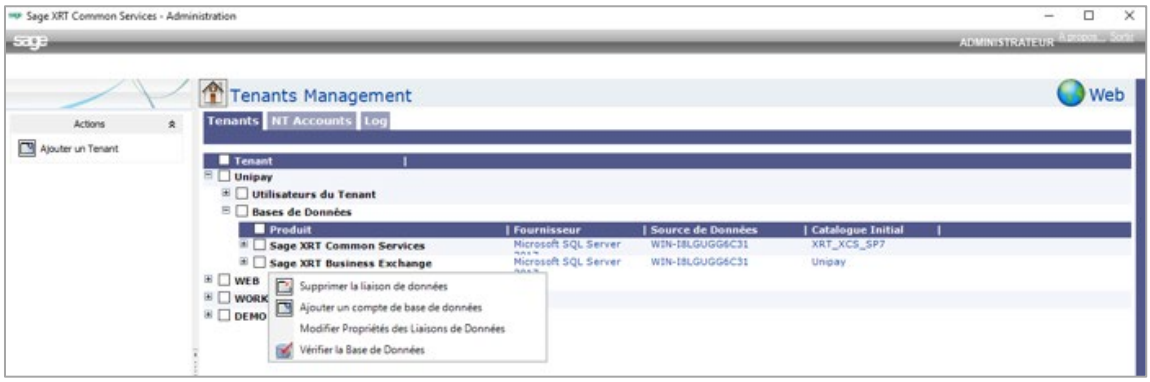
L'interface **Tenants Management** rassemble tenants, utilisateurs et bases de données.

Important ! La mise à jour d'une base de données n'est pas sans risque pour les données de l'utilisateur. Il est donc impératif de sauvegarder ces données au préalable.

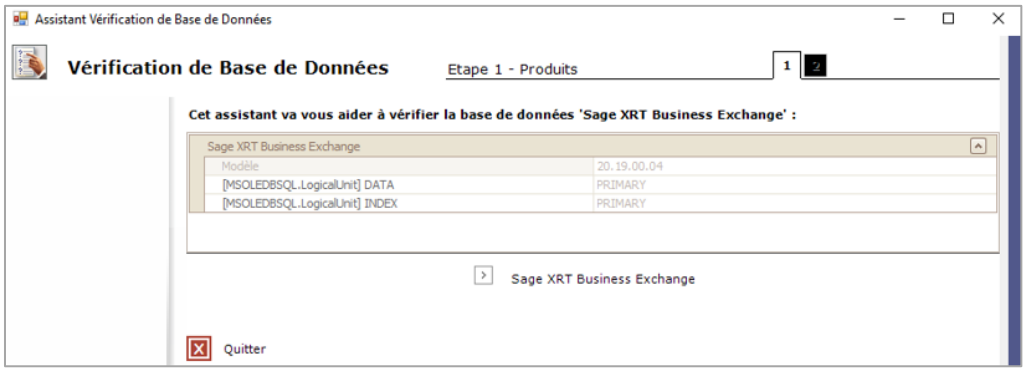
Mise à jour

Développez l'arborescence **Tenant**, puis le tenant concerné, et enfin **Bases de données**.

Après un clic droit sur la ligne correspondant à la base de données à mettre à jour, sélectionnez **Vérifier la base de données** dans le menu contextuel.



L'**Assistant Vérification de Base de Données** s'affiche.

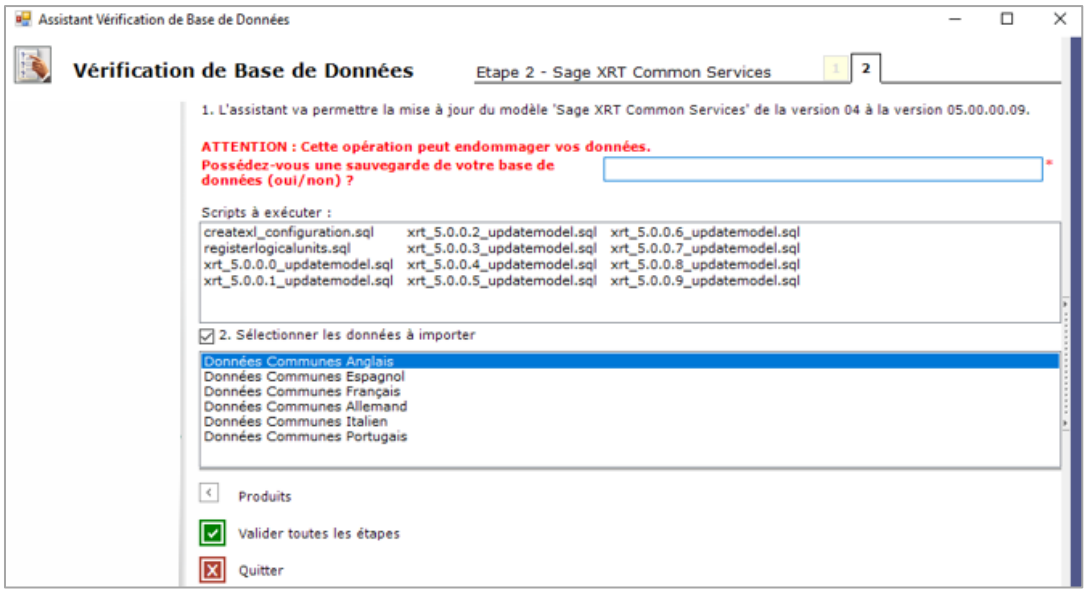


Note : Si l'utilisateur *Windows* n'est pas enregistré en tant que *DBO* dans le tenant approprié, il n'est pas autorisé à mettre à jour la base de données. L'assistant ne propose aucune mise à jour de base de données et le lien **Produit** n'est pas affiché.

Cliquez sur le produit.

Si l'assistant détecte une incohérence dans les versions, vous êtes renvoyé au processus de mise à jour de base de données.

Dans le cas contraire, l'**Etape 2** de l'assistant s'affiche.



Répondez à la question par oui ou non : **Possédez-vous une sauvegarde de votre base de données ?**

Activez l'option **Sélectionner les données à importer** si les données de la base ne sont pas à jour.

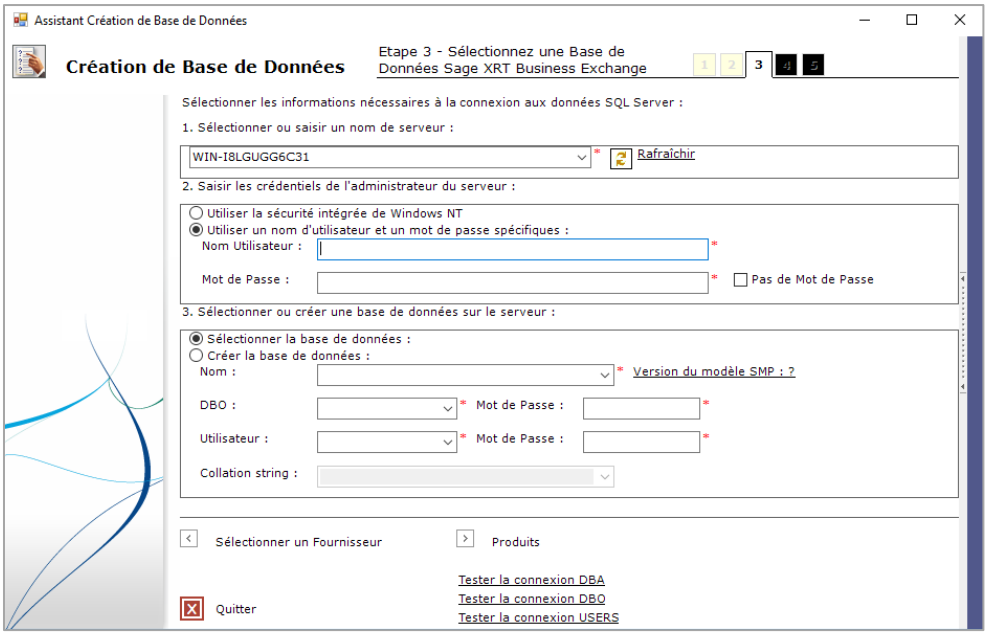
Cliquez sur le lien **Valider toutes les étapes**.

SQL Server et Utilisateurs non-SYSADMIN

Suivez la procédure décrite dans ce paragraphe

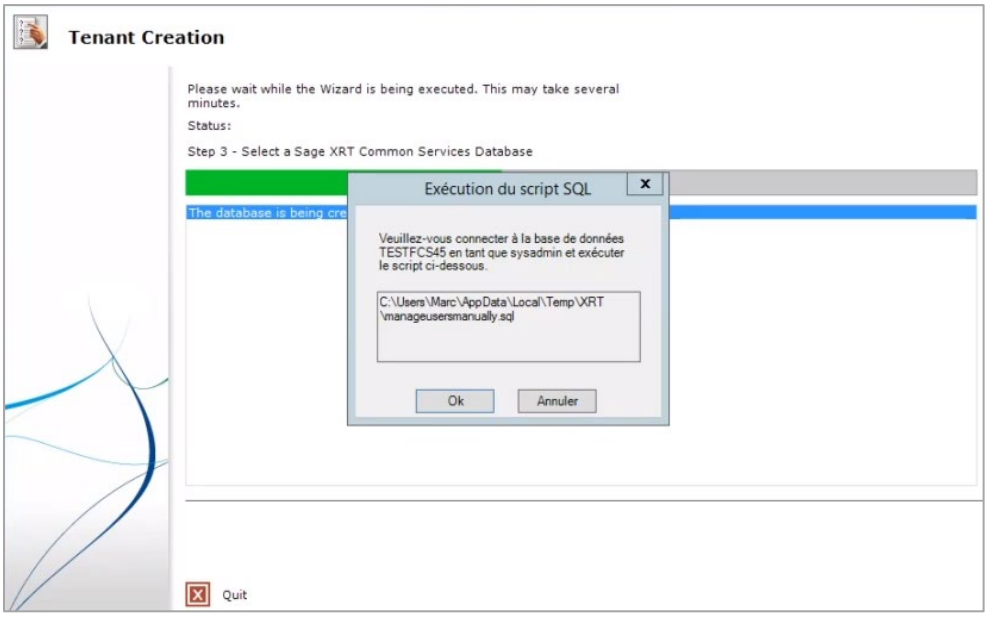
1. Vous devez d'abord créer et mettre à jour une base de données avec un utilisateur de type *dbcreator*.

2. Lors de la création d'une base de données, renseignez le nom et mot de passe de l'utilisateur de type *dbcreator*.



Note : Lors du test de connexion *DBA*, le message informant de la nécessité d'une connexion en tant que SYSADMIN, a disparu.

3. Reprenez le processus habituel de gestion de base de données : lors de l'étape de passage des scripts, un message requiert l'intervention d'un utilisateur de type SYSADMIN.

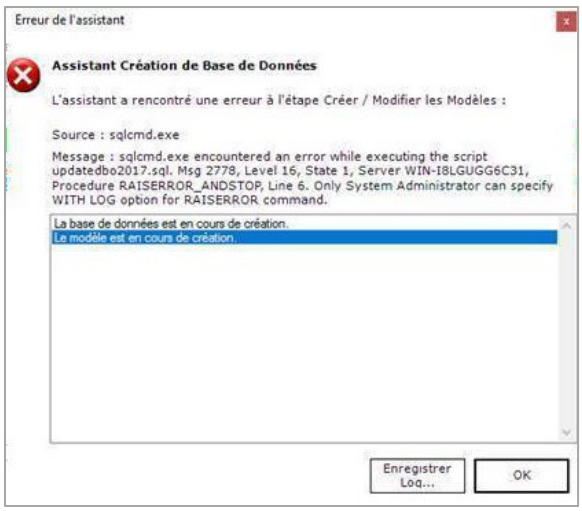


4. Connectez-vous à **SQL Server** avec un compte SYSADMIN et ouvrez le fichier indiqué.

   Lancez le script.

5. Revenez sur l'**Assistant Création de Base de Données** et cliquez sur **OK** dans la boite de dialogue pour terminer le processus.

Si vous rencontrez l'erreur de la capture ci-dessous, c'est que vous utilisez la version **Microsoft SQL Server 2017 (RTM-GDR) (KB4293803) - 14.0.2002.14 (X64) Jul 21 2018**.

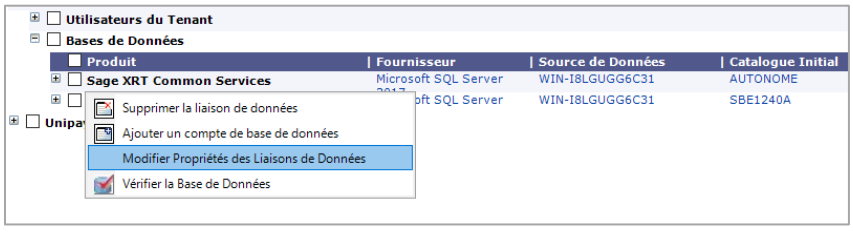


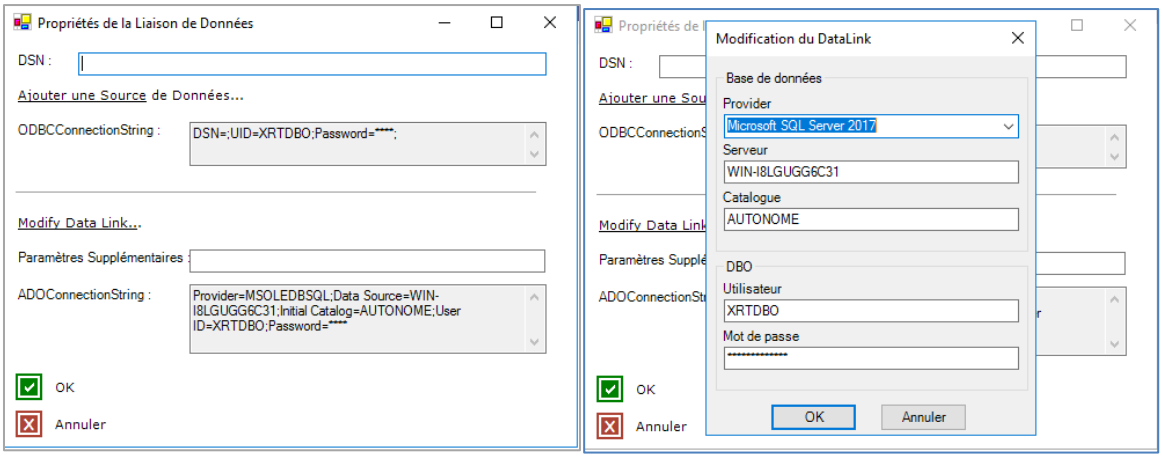
Pour remédier à cette situation, vous devez mettre à jour **SQL Server** pour utiliser la version **Microsoft SQL Server 2017 (RTM-GDR) (KB4505224) – 14.0.2027.2 (X64) Jun 15 2019**.

Modifications

Vous pouvez modifier les **Provider**, **Source de données** (serveur) et **Catalogue** sans passer par le DB Installer (**Oracle** et **SQL Server**), si vous souhaitez, par exemple, passer d'une base de test à une base de pré-production ou de production.

Pour une base **SQL Server**, sélectionnez la base de données, après un clic droit pour ouvrir le menu contextuel, cliquez sur **Modifier Propriétés des Liaisons de Données**.

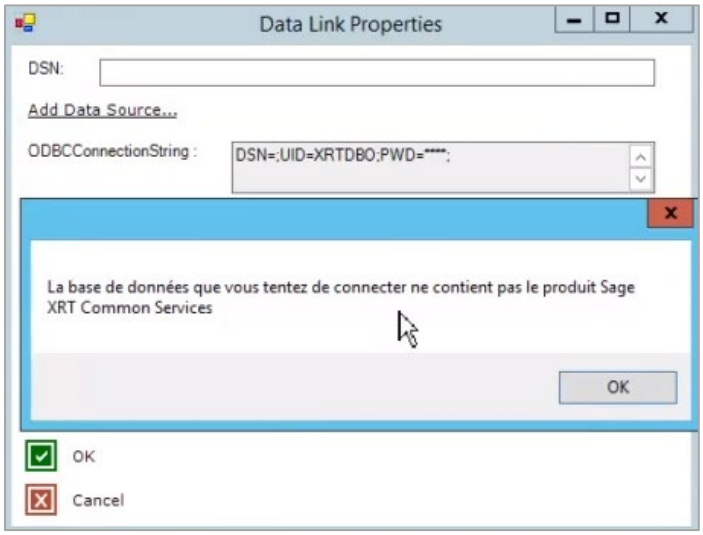




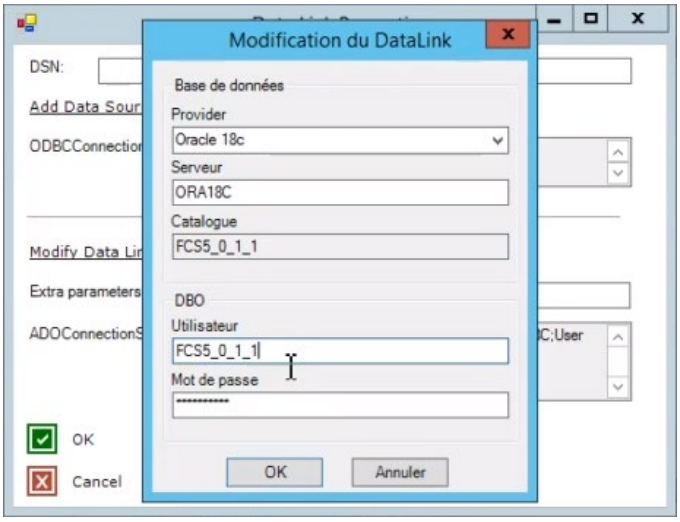
Utilisez le lien **Modify Data Link**.

La modification du **Provider** peut se faire uniquement dans la même famille (**SQL Server 2017** à **SQL Server 2019**, **Oracle** 12 à 18).

Celle du catalogue ne peut se faire que pour une base de même nature ou contenant les données de la base modifiée (modification d'une base **FCS** par une autre base **FCS** ou base **FCS**+**SXBE**), sinon un message bloquant la modification apparaît.



Pour une base **Oracle** : pas d'accès à la modification du catalogue car le catalogue reprend l'utilisateur renseigné.



Première connexion

Lors de la première connexion après création de la base de données, vous pouvez utiliser :

- le compte NT utilisé pour installer le produit
- le login **XRT** et mot de passe **S3cret#2018**

Important ! Les identifiants **XRT** et **S3cret#2018** ne sont valables qu'une seule journée !

Utilisateurs

Lors de la création d'un tenant, les groupes locaux **Windows NT Administrators** et **XRTDBAdministrators** sont automatiquement déclarés comme administrateurs du nouveau groupe.

Sage XRT Common Services propose un assistant pour la gestion des utilisateurs au sein des groupes de travail.

Cet assistant permet les actions suivantes :

- **Ajout d'un utilisateur réseau** : pour ajouter un utilisateur, saisissez le Compte NT d'un utilisateur réseau ou cliquez sur **Rechercher…** pour accéder à l'outil *Microsoft* de recherche d'un utilisateur *Windows NT*.

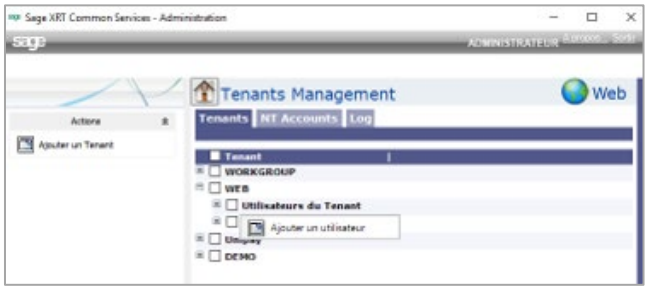
- **Ajout d'un groupe réseau** : cette option vous permet d'associer un groupe d'utilisateurs *Windows NT* à un tenant. Pour cela, cliquez sur **Rechercher…** et accédez ainsi à l'outil *Microsoft* de recherche d'un utilisateur *Windows NT*.

- **Ajout d'un compte système local** : ce type de compte est utilisé par un service système exécuté pour le compte du système local et doit accéder à une base de données.

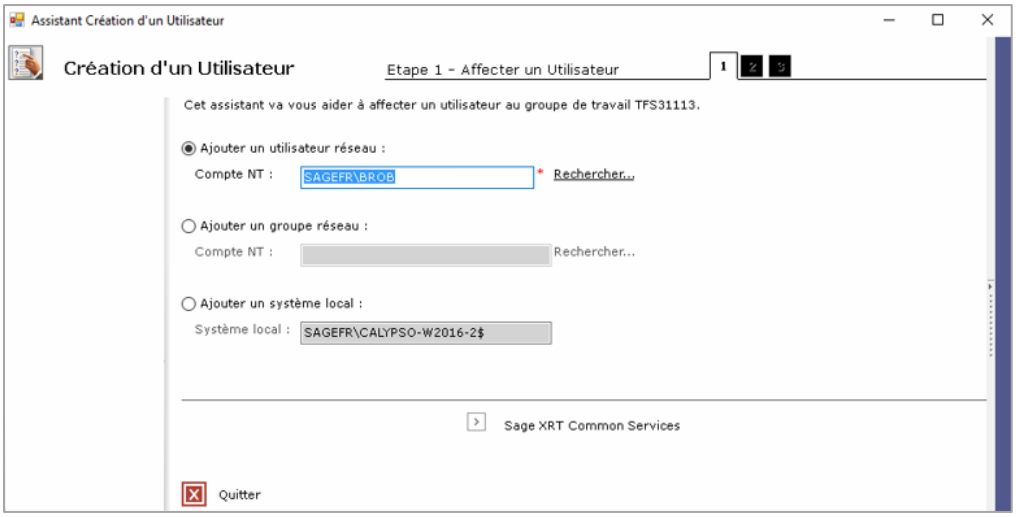
Ajout à un tenant

1. Pour assigner un utilisateur au tenant, développez l'élément **Tenants** dans l'arborescence.

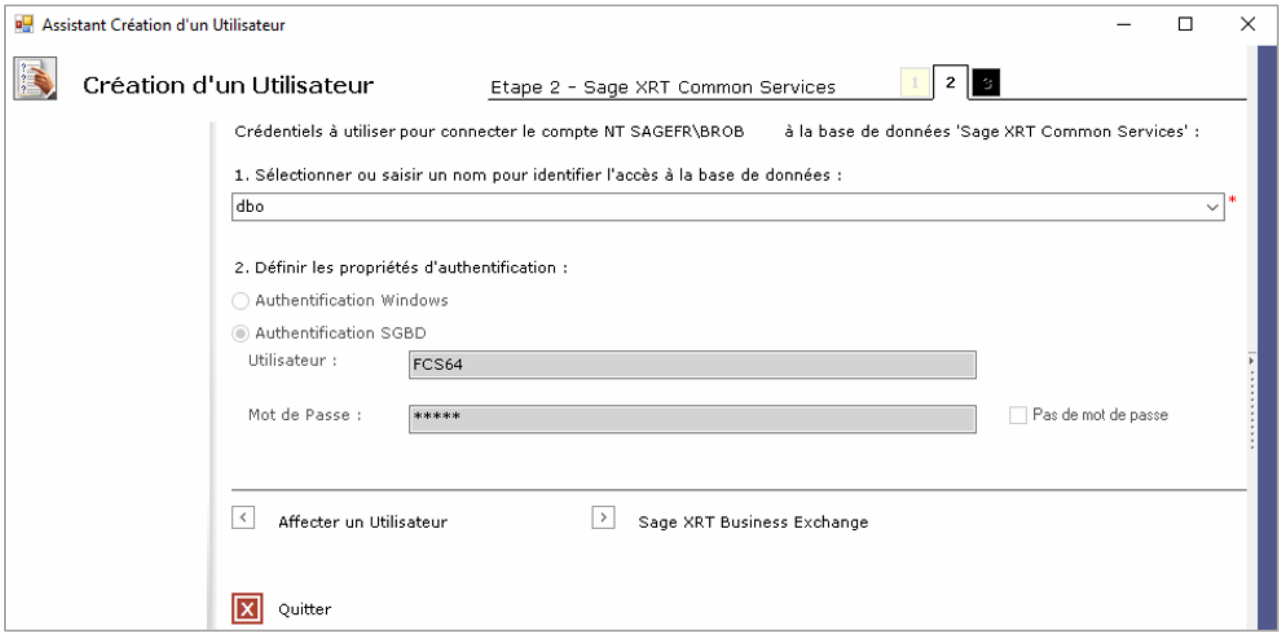
2. Sélectionnez un tenant dans la liste, et effectuez un clic droit sur le niveau **Utilisateurs du tenant**.



3. Cliquez sur **Ajouter un utilisateur** pour ouvrir l'**Assistant Création d'un utilisateur**.



4. Cliquez sur le lien **Sage XRT Common Services** pour atteindre la seconde étape de l'assistant.



5. Sélectionnez un type d'accès aux données dans la liste déroulante, ou saisissez un nouveau nom. Par défaut, l'assistant propose deux types d'accès prédéfinis :
    - **DBO** : Ce type d'accès doit être réservé au propriétaire de la base de données.
    - **Users** : Ce type d'accès doit être utilisé par les utilisateurs sans pouvoir.
6. Sélectionnez le mode d'authentification de l'utilisateur sur le serveur de bases de données :

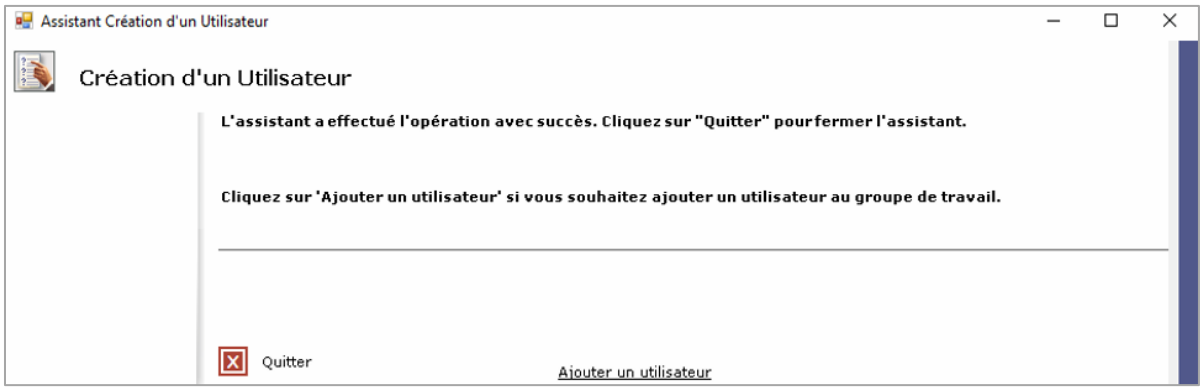
- **Authentification Windows** : l'utilisateur est authentifié par son compte NT.
- **Authentification SGBD** : l'utilisateur est authentifié par un compte qui lui a été affecté par l'administrateur du serveur de bases de données.

Important ! Il est recommandé de définir l'accès avec un compte *SQL Server*, car l'utilisation de l'authentification NT ne permet pas le pooling de connexion.

Il est possible de créer un autre nom d'accès pour un groupe d'utilisateurs donné. Ce nouvel accès est de type **User**.

Ex. : définition d'un accès *TRESORIER* pour la base **Sage XRT Treasury**, avec un compte *SQL Server* spécifique intitulé *TRESO*.

7. Cliquez sur **Valider toutes les étapes**.

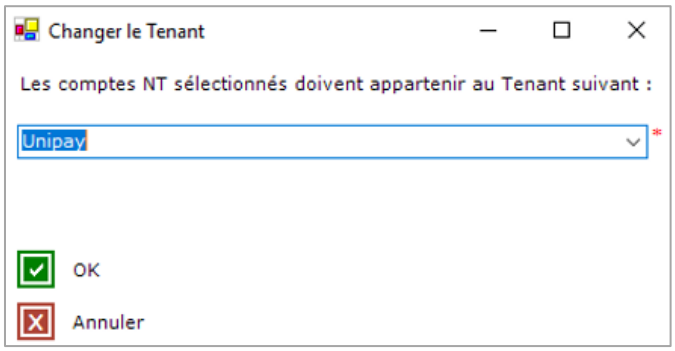


Note : Vous pouvez également gérer les utilisateurs de **Sage XRT Treasury** en répétant l'opération d'accès à la base **Sage XRT Common Services**.

Modification

Vous pouvez modifier l'affectation d'un utilisateur à un tenant.

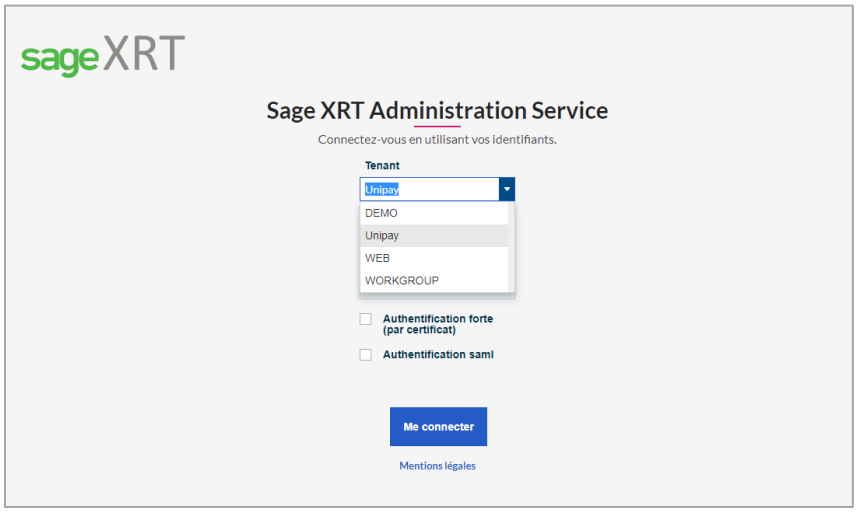
1. Après un clic droit sur l'utilisateur dont le tenant doit être modifié, sélectionnez l'action **Changer le tenant**.



2. Sélectionnez le nouveau tenant et cliquez sur le bouton **OK**.

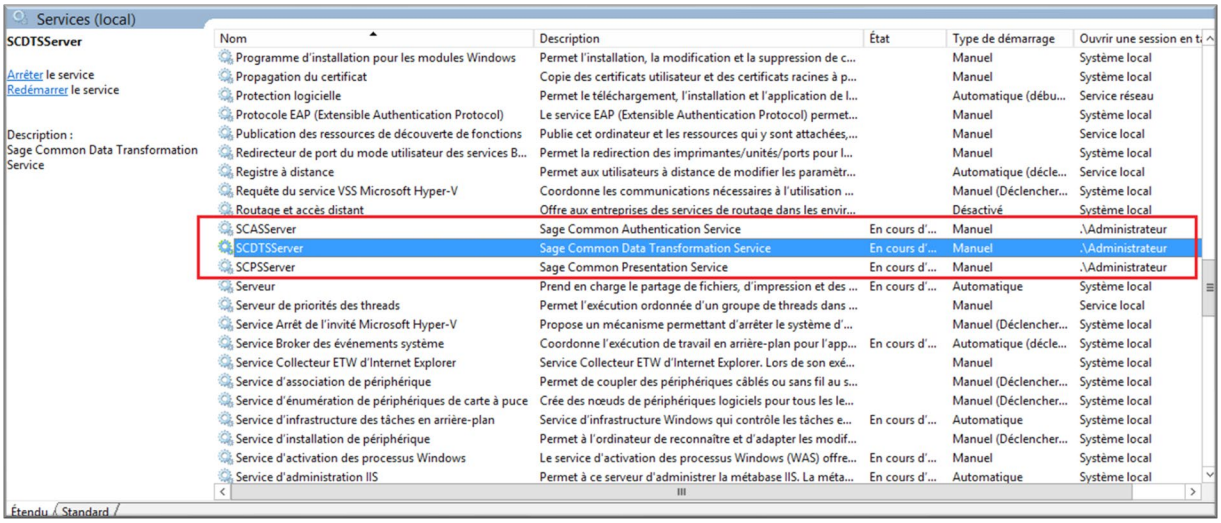
Sage XRT Administration Service

L'icône  située en haut à droite de l'interface **Tenants Management** permet un accès direct à **Sage XRT Administration Service**, le service d'administration des utilisateurs, profiles, droits, etc.



Sage XRT Common Services 5.0 est composé de trois services :

- Le service d'authentification (SCAS) qui permet de valider l'authentification des utilisateurs.
- Le service d'administration (SCPS) qui permet de gérer les parties Droits (Utilisateurs, Profils, Sites, etc.), Audits (Audits, Logs) et Transcodages (Conception et correspondances).
- Le service des fonctionnalités (SCDTS) qui couvre les thématiques suivantes :
    - Gestion des formats
    - Gestion de la sécurisation des tiers (non disponible en 5.0)
    - Gestion des paiements (virements de masse disponible en 5.0)
    - Gestion des relevés bancaires (non disponible en 5.0)



La documentation de ces API est générée par *Swagger*. Le fichier *swagger.json* correspond à une exportation de la documentation au format JSON.

Note : Accédez à la documentation et au fichier JSON (lien inscrit dans le fichier **.config* de chaque service) sous **C:\Program Files\Common Files\xrt**.

Chaque service dispose d'un fichier de configuration.

Service d'authentification

Configuration

Le fichier de configuration *Sage.SCASServer.Service.exe.config* se situe par défaut sous : **C:\Program Files\Common Files\xrt**

Activation des logs

Cf. nœud <system.diagnostics> et <diagnostics>

Définition de l'emplacement du site Web, des ports d'écoute et des hosts des services

Cf. nœud <ApplicationSettings>

```
<add key="websitehost" value="*" />

<add key="websitehostdefault" value="http://localhost" />

<add key="httpservicehost" value="http://*:80/Auth" />

<add key="httpsservicehost" value="https://*:443/Auth" />
```

Définition d'éléments de sécurité

Cf. nœud <ApplicationSettings>

```
<add key="showfriendlymessage" value="NO"/>
```

(messages d'erreur génériques type **Access Denied**)

```
<add key="redirectwhitelisthost" value="*"/>
```

(liste blanche des URL de redirection après authentification, * signifie une redirection sur le même domaine demandeur, possibilité d'ajouter d'autres noms de domaines)

(ex. : *, localhost, *yourdomainname*)

Documentation SWAGGER

URL de documentation et URL d'exportation

Paramétrage de la page de connexion

Initialisation du processus de connexion

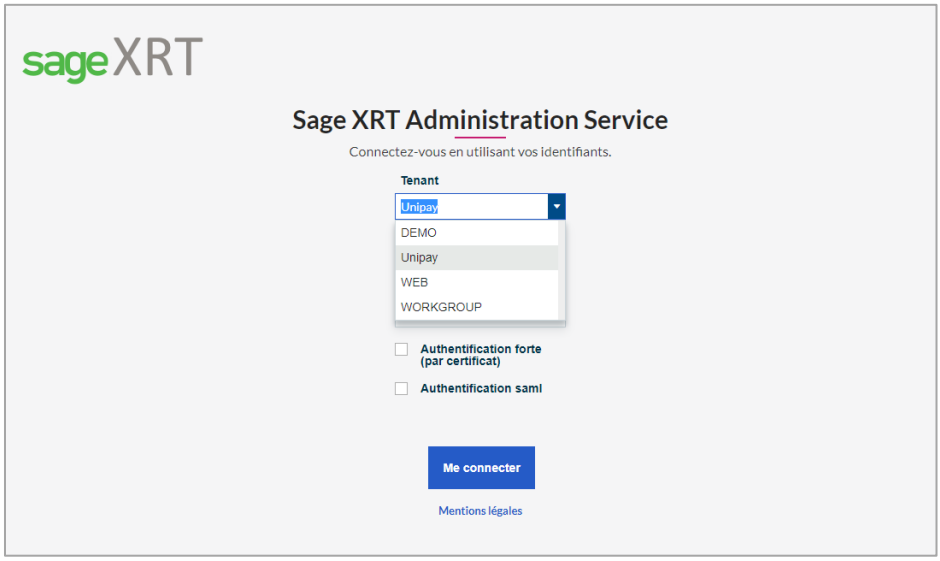
Pour initier le processus de login, postez le formulaire ci-dessous (méthode POST) au service d'authentification de **Sage XRT Common Services** :

http://*nomdemachine*:80/Auth/loginpage ou

https://*nomdemachine*:443/Auth/loginpage

```
<form name="loginpage" method="post"
action=”http://nomdemachine:80/Auth/loginpage”>

      <input type="hidden" name="workgroup" value="">

      <input type="hidden" name="strongauth" value="YES">

      <input type="hidden" name="samlv2" value="YES">

      <input type="hidden" name="product" value="SXSC">

      <input type="hidden" name="url"
value="http://nomdemachine:80/home/homepage">

<input type="hidden" name="xrtloginweborigin" value="">

&lt;input type="hidden" name="goto" value=""&gt;

</form>
```

La page de connexion s'affiche.



Variable du formulaire workgroup

Cette variable permet de pré-paramétrer un tenant. Elle pilote la présence du menu déroulant permettant de sélectionner un tenant.

Cependant, si elle est initialisée avec le nom d'un tenant, elle n'apparaît pas et le couple utilisateur/mot de passe de l'utilisateur est vérifié sur ce tenant.

Variable strongauth

Cette variable permet de donner accès à l'option d'authentification forte.

Si la variable a pour valeur **YES** alors la case à cocher permettant d'utiliser l'authentification forte apparait.

Si la variable est vide alors l'option est absente.

Variable samlv2

Cette variable permet de donner accès à l'option d'authentification SAML.

Si la variable a pour valeur **YES** alors la case à cocher permettant d'utiliser l'authentification SAML apparait.

Si la variable est vide alors l'option est absente.

Variable du formulaire product

Cette variable permet de gérer le nom du produit affiché sur la page de connexion. Elle admet soit un code prédéfini, soit un texte libre. Les codes prédéfinis sont les suivants :

- product=SXSC pour **Sage XRT Administration Service**
- product=SCSDTS pour **Sage XRT Functional Service**
- product=SXBEONLINEBANKING pour **Connexion à OnlineBanking**
- product=SXBEADMINISTRATION pour **Connexion à Business Exchange Administration**
- product=VIEWANDSIGN pour **Sage View & Sign**

Lorsqu'on utilise les noms de code, la localisation en FR, ES et US est gérée par le processus de connexion **XRTLoginWeb** de **Sage XRT Common Services**.

Un texte libre peut être utilisé comme valeur de la variable.

Variable url

Cette variable permet de définir l'URL de *callback* en cas de succès du processus d'authentification (cf. liste blanche des URL possibles).

Variable xrtloginweborigin

Cette variable permet de définir l'URL du site web du processus d'authentification lorsque celui-ci n'est pas hébergé sur la même machine que la page qui initie le formulaire d'initialisation du processus d'authentification **XRTLoginWeb**. Il y a également possibilité de ne pas utiliser cette variable et de réaliser le paramétrage au niveau du fichier de configuration du service d'authentification (SCAS).

Le processus d'authentification **XRTLoginWeb** permet également de définir si les mots de passe sont cryptés puis encodés en Base64 ou seulement encodés en Base64. Pour cela il suffit de définir un certificat au niveau du fichier de configuration du service d'authentification (SCAS).

Pour le certificat qui crypte le mot de passe :

```
<add key="serialnumberforpwdcrypt" value="" /> tag
```

Variable du formulaire goto

Cette variable admet deux valeurs : **SAML** ou **SAGEID**.

Pour être prise en compte, la variable **WORKGROUP** doit être renseignée.

Si ces valeurs sont utilisées, l'utilisateur est directement amené sur le processus d'authentification **SAML** ou **SAGEID** sans passer par la mire de connexion.

Service d'Administration

Configuration

Le fichier de configuration *Sage.SCPSServer.Service.exe.config* se situe par défaut sous : **C:\Program Files\Common Files\xrt**.

Activation des logs

Cf. nœud <system.diagnostics> et <diagnostics>

Définition de l'emplacement du site Web, des ports d'écoute et des hosts des services

Cf. nœud <ApplicationSettings>

```
<add key="websitehost" value="*" />

<add key="websitehostdefault" value="http://localhost" />

<add key="httpservicehost" value="http://*:80"/>

<add key="httpsservicehost" value="https://*:443"/>
```

Définition du compte et de la fréquence en seconde de synchronisation des groups NT/LDAP

Cf. nœud <ApplicationSettings>

```
<add key="syncprofilesiteuser" value="XRT"/>

<add key="syncprofilesitefrequency" value="3600"/>
```

Définition d'éléments de sécurité

Cf. nœud <ApplicationSettings>

```
<add key="showfriendlymessage" value="NO"/>
```

(numéro de version non présenté et messages d'erreur générique)

Documentation SWAGGER

URL de documentation et URL d'exportation

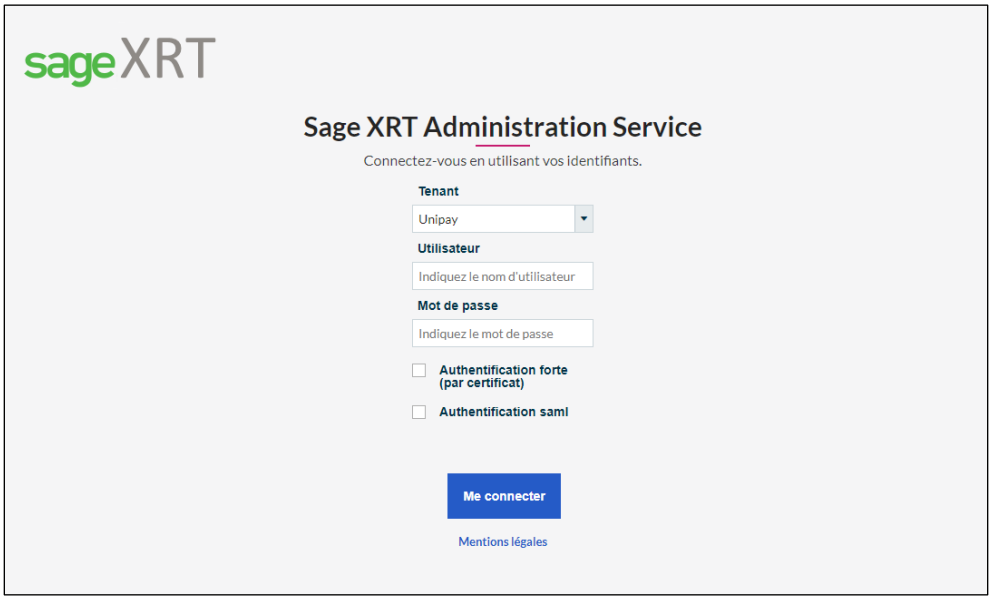
Connexion

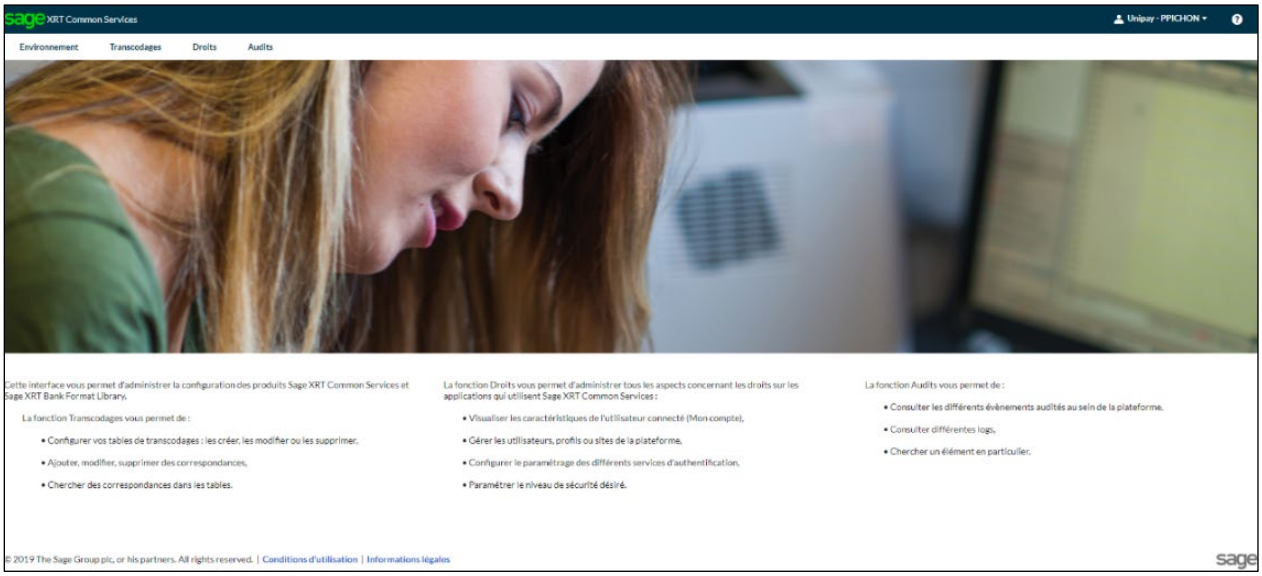
L'utilisation de l'interface du service d'Administration se fait via l'URL :

http://localhost/SCPS/index.html

Sage XRT Administration Service

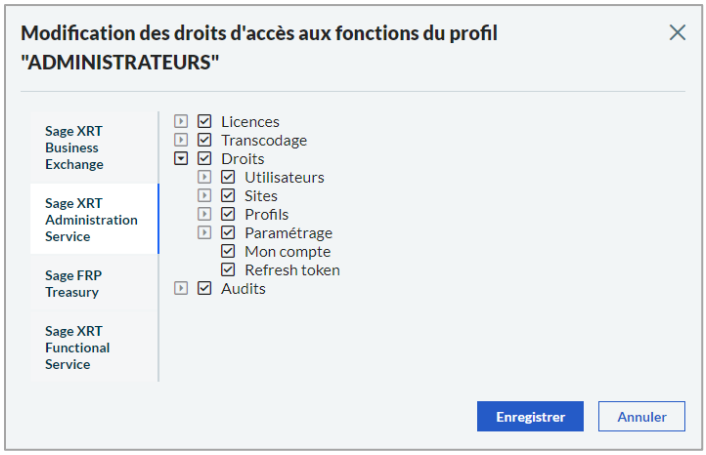
Elle nécessite le démarrage des services d'authentification et d'administration.





Refresh Token

Le **Refresh token** est utilisé par les services *Windows* qui consomment les API Rest de **Sage XRT Common Services**. Sa mise en place demande l'installation d'un certificat sur le poste client et sur le poste serveur.



Licences

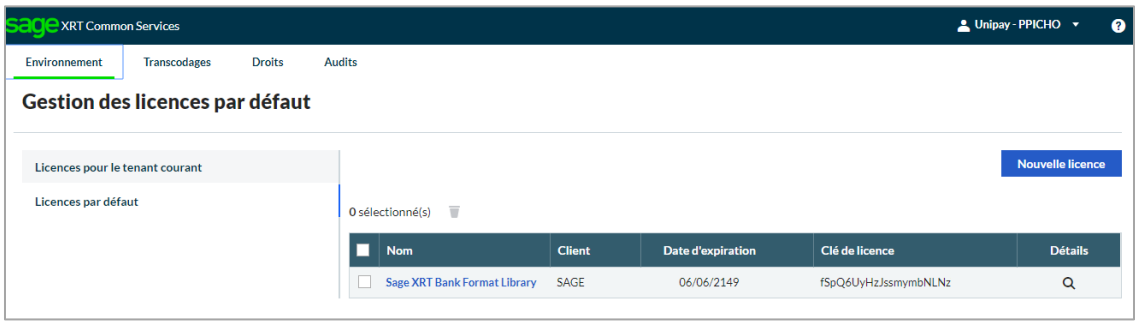
Les licences par défaut sont appliquées à tous les tenants pour lesquels aucune licence n'a été spécifiée.

La fonction **Licences par défaut** est la même que la fonction **Licences**. La licence reste stockée dans la base de registre.

Les interfaces sont identiques pour les deux types de licences.

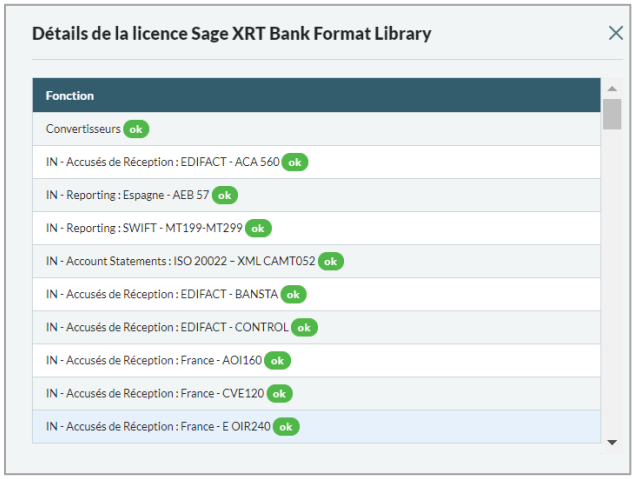
Les licences existant pour le tenant courant sont listées à l'appel de la fonction **Licences pour le tenant courant**.

Les licences existant par défaut sont listées à l'appel de la fonction **Licences par défaut**.



Pour visualiser le détail correspondant à un clé de licence, cliquez sur l'icône *loupe*.

Exemple pour **Sage XRT Bank Format Library** :

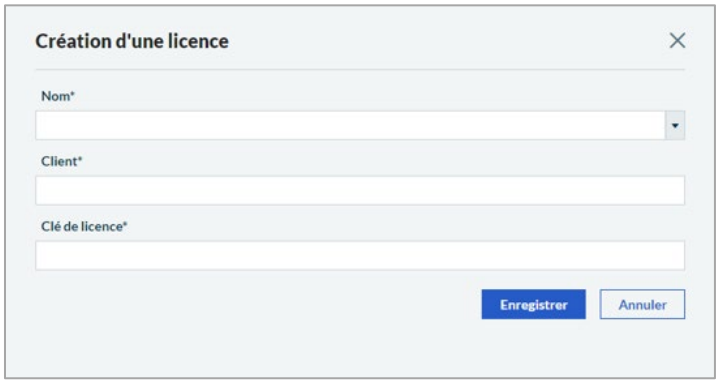


Création

Dans le menu **Environnement**, le bouton **Nouvelle licence** vous permet d'accéder à l'assistant de création.

Vous devez compléter tous champs marqués d'un astérisque.

Cliquez sur le bouton **Enregistrer** pour finaliser la création. Un message vous confirme la création de la licence.



Modification

Cliquez sur le lien disponible sur l'information **Nom** de la licence à modifier. Seules les informations **Client** et **Clé de licence** sont modifiables.

Cliquez sur le bouton **Enregistrer** pour finaliser la modification de la licence. Un message vous confirme la modification de la licence.

Suppression

A partir de la liste des licences, sélectionnez la licence à supprimer.

Cliquez sur l'icône *corbeille* pour finaliser la suppression de la licence. Un message vous confirme la suppression de la licence.

Application

Lorsqu'une licence pour un tenant est spécifiée, la licence par défaut ne s'applique pas.

Droits

Ce chapitre présente une description des différentes méthodes d'authentification des utilisateurs proposées par **Sage XRT Common Services** ainsi qu'une description des actions à mener en termes de gestion des utilisateurs et de leurs droits d'accès.

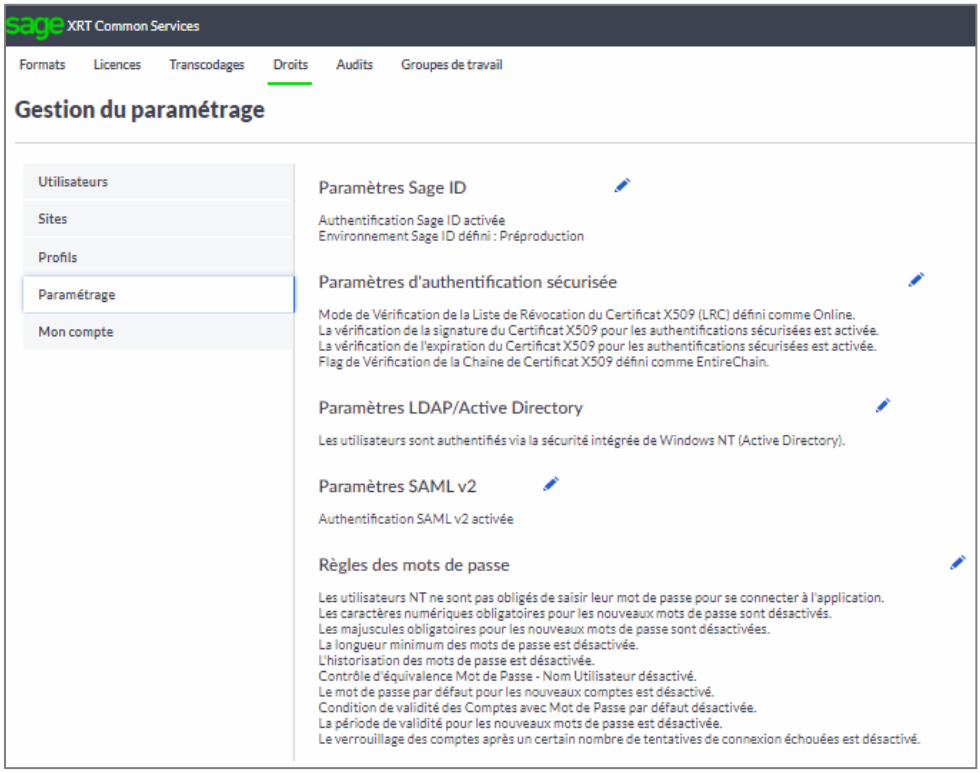
Paramétrage authentification

Le modèle de login UMAPI supporte plusieurs modes d'authentification :

- authentification **Windows**
- authentification **UMAPI**
- authentification **LDAP**
- authentification **SAML**
- authentification **CloudID** : **Sage ID V2**

Ces modes d'authentification sont à activer préalablement à leur rattachement à un utilisateur.

Dans le menu **Droits**, cliquez sur l'onglet **Paramétrage**.



Utilisez l'icône *crayon* pour modifier les paramètres de chaque mode d'authentification.

Authentification Windows NT

L'authentification *Windows NT* tire avantage de la sécurité *Windows NT* et de sa gestion des comptes utilisateur. Ce mode de sécurité permet aux applications *Sage XRT* d'utiliser les crédentiels des utilisateurs *Windows NT*.

Les applications *Sage XRT* proposent deux modes de fonctionnement avec ce type d'authentification :

- Le mode *trusted connection* (connexion sécurisée) n'est plus supporté à partir de la version 5.0 car **Sage XRT Common Services** est désormais une application web.
- Le mode standard : l'utilisateur doit saisir son mot de passe car il est contrôlé par le système via les *API Windows*.

Avantages du mode d'authentification *Windows NT* :

- Pas de crédentiels supplémentaires à mémoriser
- Pas de répercussion dans *UMAPI* lors d'un changement de mot de passe

- Gestion des mots de passe conforme aux exigences du *Sarbanes-Oxley Act*
- Accès à d'autres fonctionnalités du système comme le changement périodique de mot de passe et l'audit des accès

Note : La mise en place de l'authentification *Windows NT* nécessite de travailler en étroite collaboration avec l'administrateur *Windows* lors de la création des utilisateurs et des groupes. L'implémentation dans *UMAPI* de l'authentification *Windows* est basée sur la librairie de classes de bases du namespace *System.DirectoryServices* du framework .NET.

Authentification UMAPI

Lorsqu'il utilise l'authentification *UMAPI*, un utilisateur se connectant à une application *Sage XRT* fournit un nom d'utilisateur et un mot de passe contrôlés à partir d'informations contenues dans la base de données.

Avantages du mode d'authentification *UMAPI* :

- Gestion des mots de passe conforme aux exigences de la loi *Sarbanes-Oxley*
- Enregistrement des quatre derniers mots de passe qui ne peuvent être réutilisés lorsque le système demande un changement de mot de passe (fonctionnalité paramétrable)
- Compte utilisateur verrouillé après trois échecs successifs d'authentification (fonctionnalité paramétrable)
- Compte utilisateur verrouillé débloqué après une période paramétrable
- Seuls les codes de hachage *SHA1* des mots de passe sont enregistrés dans la base de données, et non les mots de passe
- Mot de passe d'au moins six caractères comprenant au moins une majuscule et un chiffre. Fonctionnalité paramétrable
- Mot de passe à changer périodiquement (fonctionnalité paramétrable)
- Possibilité pour l'administrateur de verrouiller un compte utilisateur pour une durée déterminée ou de façon permanente

Authentification LDAP

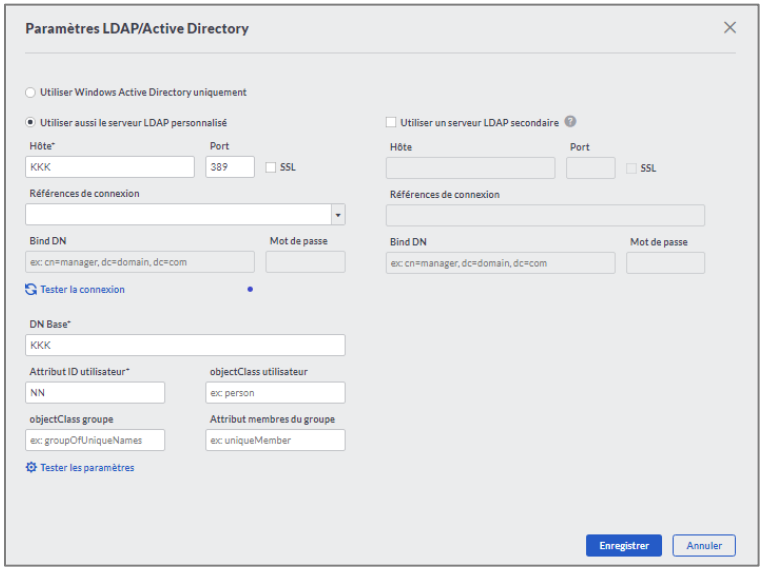
Lorsqu'il utilise l'authentification *LDAP*, un utilisateur se connectant à une application *Sage XRT* doit fournir un nom d'utilisateur et un mot de passe contrôlés à partir d'informations contenues dans l'annuaire *LDAP*.

Avantages du mode d'authentification *LDAP* :

- Alternative intéressante lorsqu'une société ne souhaite pas utiliser exclusivement le système d'authentification *Windows NT*
- Authentification applicative dans le cadre des produits *Sage XRT*

La configuration de l'accès à l'annuaire s'effectue à partir de l'écran de paramétrage de la gestion des utilisateurs. L'administrateur doit renseigner les paramètres suivants :

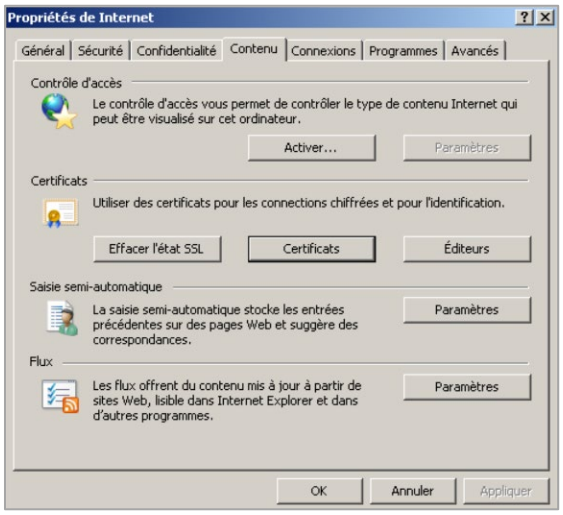
- L'adresse IP de la machine qui héberge le serveur *LDAP*
- Le numéro de port sur lequel le serveur *LDAP* doit être appelé
- Le paramètre **Base DN** de l'annuaire
- L'attribut **User ID attribute name** sur lequel doit se baser l'authentification de l'utilisateur
- Le nom de la classe **Utilisateur** à utiliser lors de la recherche d'un individu dans l'annuaire
- Le nom de la classe **Group** à utiliser lors de la recherche d'un groupe d'individus dans l'annuaire
- Les crédentiels permettant d'effectuer une recherche sur l'annuaire (le bouton **Test Connection** permet de vérifier ces crédentiels)



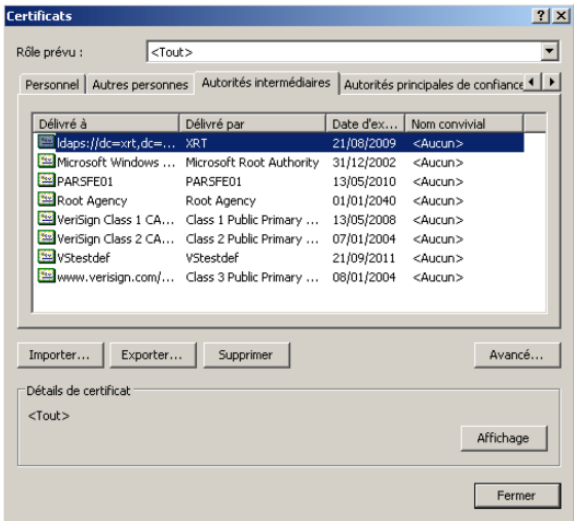
Note : L'implémentation dans UMAPI de l'authentification LDAP est basée sur la librairie de classes de bases du *namespace System.DirectoryServices* du framework .NET.

En règle générale, les échanges LDAP entre les clients et le serveur transitent par le port *TCP/IP* standard (port 389) sous forme cryptée ou via un tunnel SSL (port 636). La technologie SSL peut être activée en installant un certificat publié par une autorité de certification approuvée par le contrôleur de domaine et les clients LDAPS. L'approbation est établie en configurant les clients et le serveur de façon à approuver l'autorité de certification racine à laquelle est enchaînée l'autorité de certification émettrice.

Le certificat installé se trouve dans le magasin de certificats personnel de l'ordinateur local au niveau propriétés Internet du navigateur : onglet **Contenu**, bouton **Certificats et autorités intermédiaires**.



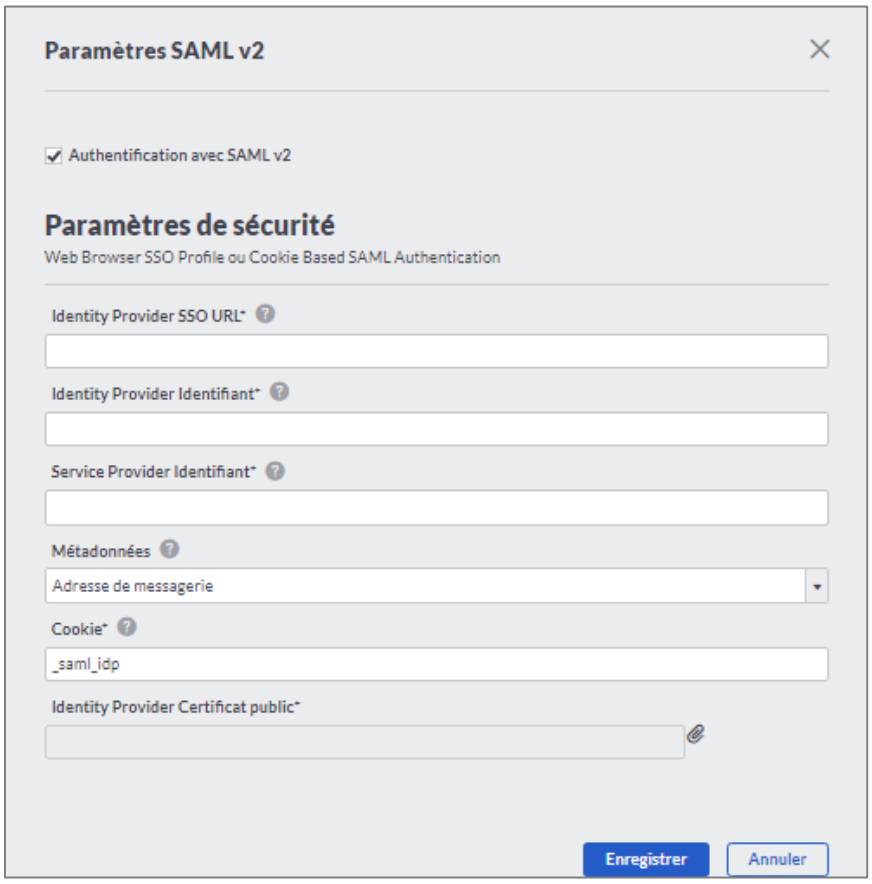
Cliquez sur le bouton **Certificats**. La boîte de dialogue suivante s'affiche.



Authentification SAML

Security Assertion Markup Language (SAML) est un standard informatique définissant un protocole pour échanger des informations liées à la sécurité, basé sur le langage *XML*.

SAML propose l'authentification unique (en anglais *Single Sign-On* ou *SSO*) sur le web. De cette manière, un utilisateur peut naviguer sur plusieurs sites différents en ne s'authentifiant qu'une seule fois.



L'authentification *SAML* fait intervenir :

- L'*Identity provider* (l'entité qui détient les identifications) : champs **Identity Provider SSO URL** et **Identity Provider Identifiant**)
- Les *Services providers* (les services qui nécessitent une authentification) : champ **Service Provider Identifiant**. Plusieurs *Services providers* peuvent être renseignés (ils forment le cercle de confiance des services par rapport à un *IdP*).
- L'utilisateur qui est identifié via une donnée déclarée dans les métadonnées (ex. : ID ou e-mail).

Double Authentification

La technologie choisie est celle du protocole *TOTP* (RFC 6238).

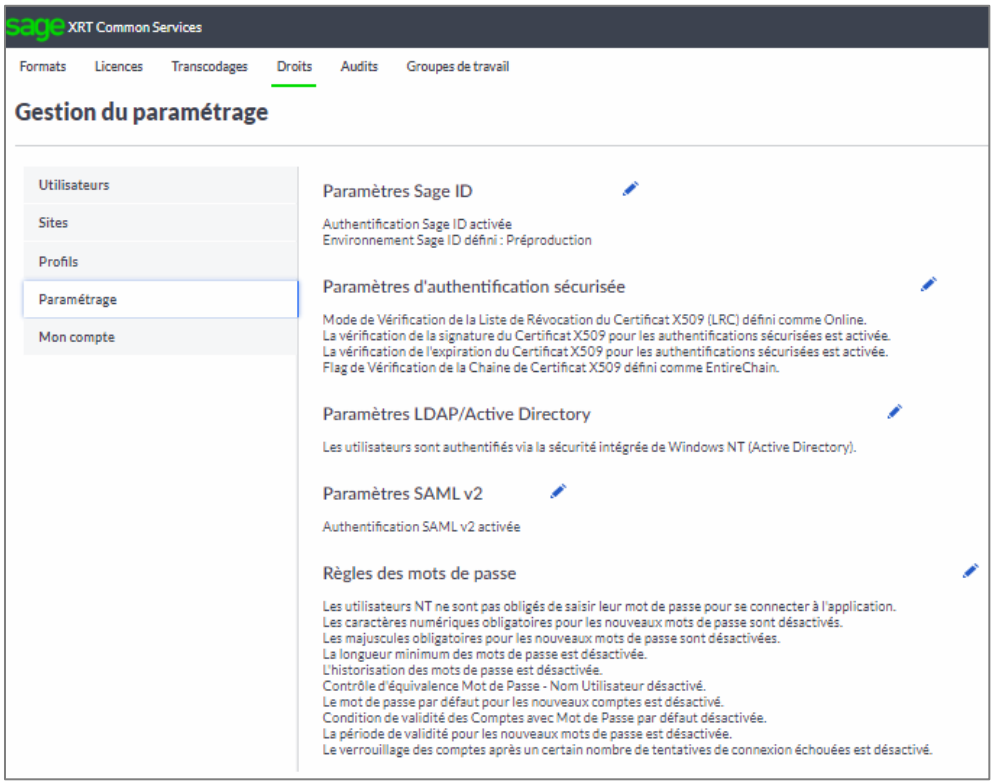
Ce protocole associe une clé secrète à l'heure, en utilisant une fonction de hachage cryptographique pour générer un mot de passe à usage unique. Prenant en compte la latence du réseau et les éventuels retards d'horloges, l'horodatage augmente par intervalles de 30 secondes, ce qui réduit l'espace de recherche potentiel.

L'adoption de ce protocole permet d'utiliser des applications mobiles déjà disponibles, comme par exemple *FreeOTP*, *Microsoft Authenticator* ou *Google Authenticator*.

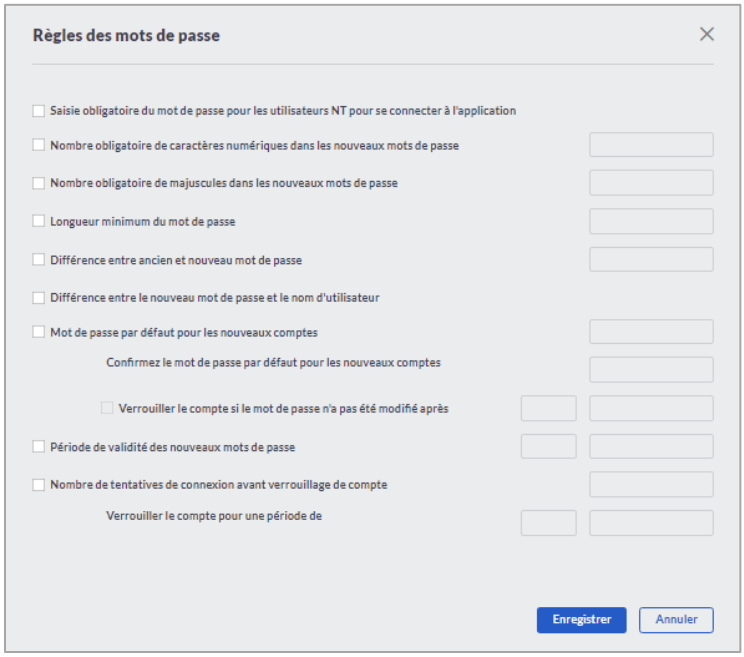
Note : Voir le document *SCS.5.0.DoubleAuthentication.UserGuide_FR*.

Règles des mots de passe

Ces règles sont définies à partir du menu **Droits**. Cliquez sur l'onglet **Paramétrage**.



Utilisez l'icône *crayon* pour modifier les règles des mots de passe.



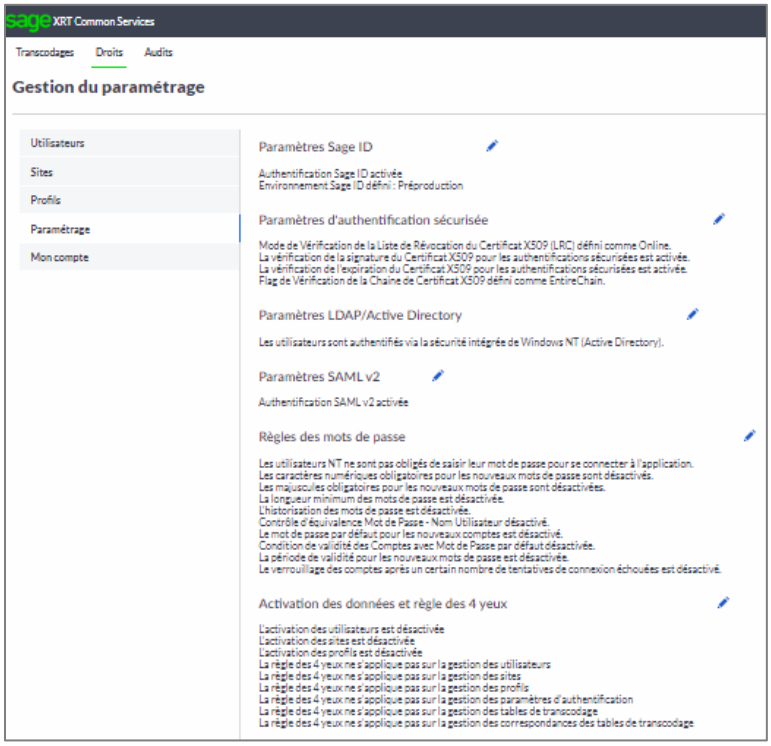
Activation des données et règle des 4 yeux

L'activation et l'application de la règles des *4 yeux* doivent être explicitement demandées. Par défaut, les données ne doivent pas être activées préalablement à leur utilisation et la règle des *4 yeux* ne s'applique pas.

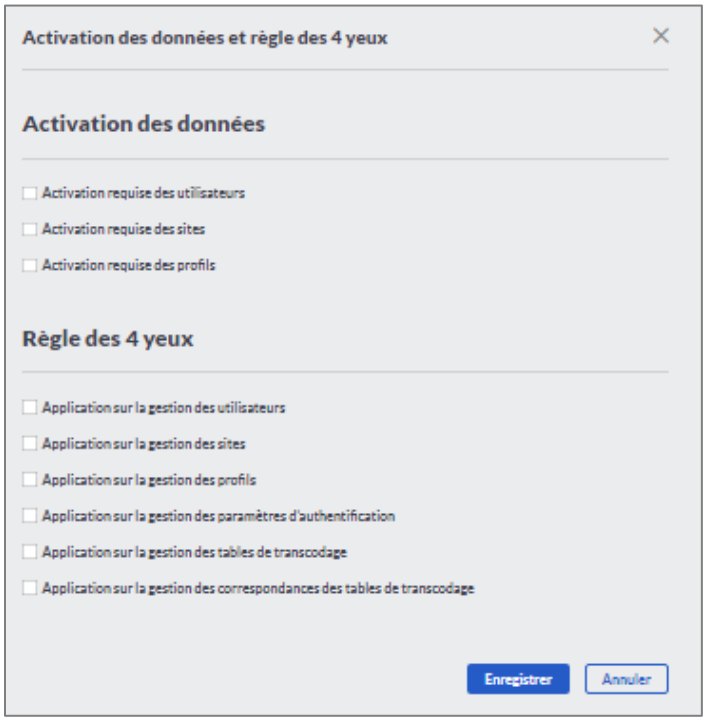
Important ! Rappel du principe de l'activation : une donnée inactive ne peut être utilisée (statut inactif). Cette donnée devra être activée pour pouvoir être utilisée (statut actif).

Rappel du principe de la règle des 4 yeux : un même utilisateur ne peut enchaîner 2 actions sur un même élément (création + modification, création + suppression, création + activation…).

Cliquez sur l'onglet **Paramétrage**.



Utilisez l'icône *crayon* pour modifier les règles d'activation des données et des *4 yeux*.



L’activation d’un élément ne pourra être demandée que s’il n’existe aucun élément en statut inactif.

L’activation peut être demandée sur :

- Les utilisateurs
- Les profils
- Les sites

Profils

En cas d’activation des profils requise :

- Les profils NT/LDAP sont toujours créés actifs et non désactivables. Les autres profils sont toujours créés en inactif.
- Si un utilisateur est rattaché à un profil inactif, il n'aura pas les droits associés à ce profil.
- Tous les utilisateurs rattachés à un profil NT/LDAP sont créés actifs et peuvent être désactivés.

En cas d’activation des profils non requise : tous les profils sont créés actifs.

Sites

En cas d’activation des sites requise :

- Les sites NT/LDAP sont toujours créés actifs et non désactivables. Les autres sites sont toujours créés en inactif.
- Tous les utilisateurs rattachés à un site NT/LDAP sont créés actifs et peuvent être désactivés.

En cas d’activation des sites non requise : tous les sites sont créés actifs.

Utilisateurs

En cas d’activation des utilisateurs requise :

- Tous les utilisateurs rattachés à un profil NT/LDAP sont créés actifs et peuvent être désactivés.
- Tous les utilisateurs rattachés à un site NT/LDAP sont créés actifs et peuvent être désactivés.
- Un utilisateur "générique" est créé inactif.

- Un utilisateur ne peut pas s'auto-activer.
- Un utilisateur inactif ne peut se connecter nulle part.

En cas d'activation des utilisateurs non requise : tous les utilisateurs sont créés actifs.

L'application de la règle des *4 yeux* peut être demandée sur :

- Les utilisateurs
- Les profils
- Les sites
- La gestion des paramètres d'authentification
- Les tables de transcodage
- Les correspondances des tables de transcodage

Le paramétrage de l'activation des données et de l'application de la règle des *4 yeux* est toujours soumis à la règle des *4 yeux*.

Compte utilisateur

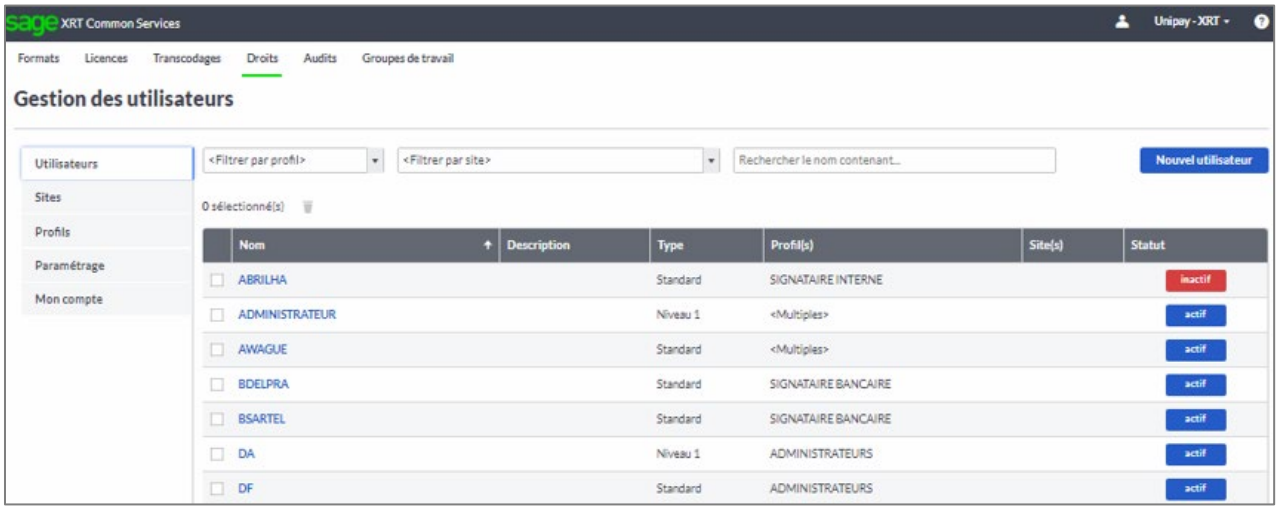
Un compte utilisateur permet à un utilisateur de s'authentifier auprès d'une application *Sage XRT*. Il permet également de gérer les autorisations d'accès de cet utilisateur aux fonctionnalités de l'application.

Un compte utilisateur comporte les éléments suivants :

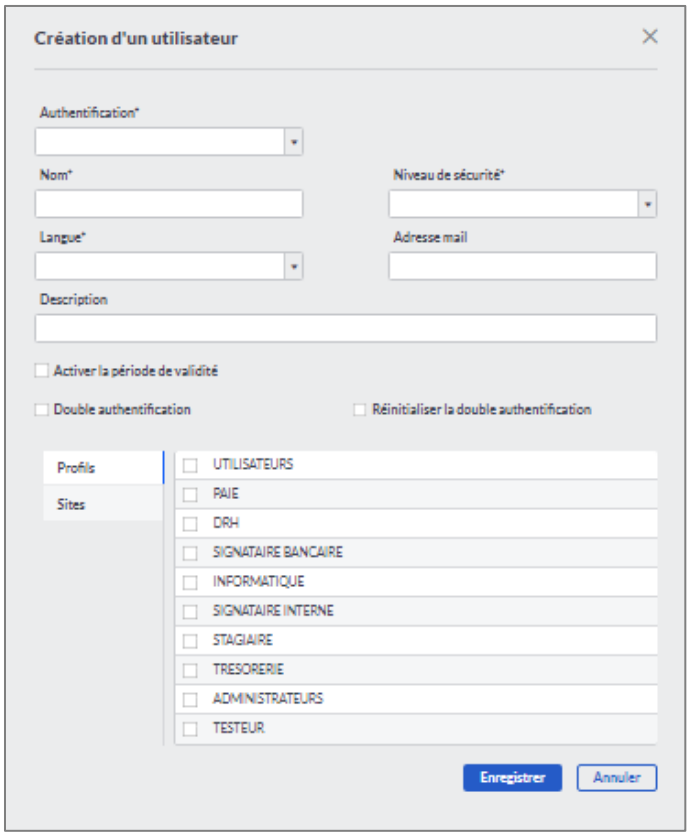
- Langue de l'utilisateur (Français, Anglais, Espagnol, Portugais, Italien, Allemand)
- Adresse électronique de l'utilisateur pour envoi des notifications
- Description
- Type d'utilisateur (administrateur ou simple utilisateur)

Ajouter un utilisateur

Dans le menu **Droits**, cliquez sur l'onglet **Utilisateurs**.



Dans l'onglet **Gestion des utilisateurs**, le bouton **Nouvel utilisateur** vous permet d'accéder à l'assistant de création.



Sélectionnez un mode d'authentification et renseignez le nom de l'utilisateur :

Authentification Windows : deux modes d'ajout d'un utilisateur NT :

- Ajout via la sélection dans la liste présentée dans la boîte de dialogue. Les accès à la base de données doivent être définis préalablement pour chaque utilisateur.
- Ajout via la recherche dans l'annuaire de l'entreprise. L'utilisateur hérite de l'accès à la base de données de type *XRTUsers*.

Authentification LDAP : recherche et sélection des utilisateurs appartenant à l'annuaire paramétré dans la configuration de l'authentification *LDAP* donné grâce au bouton **Recherche**.

Authentification standard : saisie d'un identifiant unique pour l'utilisateur.

Important ! Il est fortement conseillé de mettre en œuvre une gestion des accès reposant sur les comptes *NT*.

Authentification SAML : saisie d'un identifiant *SSO* obligatoire fourni par l'*Identity Provider*.

Authentification Sage Id

Option **Double authentification** : Ce champ pourra être coché ou décoché à la création de l'utilisateur mais aussi quand il aura déjà été créé auparavant (modification utilisateur) quel que soit le type d'authentification (*Standard*, *Windows*, *SageID*, etc.). Si cette option est cochée, lors de sa première connexion, l'utilisateur doit initialiser cette authentification via la saisie d'un code secret obtenu après scan du code QR (ou saisie du code équivalent) grâce à une application compatible avec le protocole *TOTP* 6 digits (*FreeOTP* par exemple). Tant que cette initialisation n'a pas lieu, l'option apparaît en orange. Par la suite, elle apparait en vert.

Si un utilisateur perd (ou remplace) son smartphone ou désinstalle l'application d'authentification, il faudra réinitialiser son état pour qu'il puisse recréer le lien.

Pour cela, une option **Réinitialiser Double Authentification** est disponible dans la liste des utilisateurs.

Choisissez le type d'utilisateur à créer :

- **Administrateur de sécurité de niveau 1** : gère les droits d'accès des utilisateurs du groupe de travail.
- **Utilisateur standard** : utilisateur n'ayant aucun droit d'écriture ou de modification.

Complétez les informations qui suivent.

- **Langue** : sélectionnez dans la liste déroulante la langue d'usage de l'utilisateur.
- **Description** : saisissez une description pour l'utilisateur.
- **Adresse mail** : saisissez l'adresse email de l'utilisateur.
- **Période de validité** : cochez la case pour activer les trois champs permettant de définir la période de validité de l'utilisateur.

Rattachez éventuellement l'utilisateur à un profil et site.

Cliquez sur **Enregistrer** ou **Annuler** pour sortir de la boite de dialogue et revenir à la liste des utilisateurs.

Activer un utilisateur

Un utilisateur ne peut pas s'auto-activer.

Utilisateur expiré

Un utilisateur obtient le statut expiré lorsque sa période de validité a expiré.

Utilisateur bloqué

Un utilisateur obtient le statut bloqué lorsque suite à l'application des règles de mot de passe, l'utilisateur n'est pas parvenu à se connecter.

Profils

La **Console d'Administration** est désormais activée. Vous avez la possibilité de créer un ou plusieurs profils pour les utilisateurs.

Par défaut, un utilisateur ne peut accéder à aucune fonctionnalité du produit. L'administrateur doit intervenir pour définir les droits d'accès des utilisateurs.

Un profil est constitué d'utilisateurs partageant les mêmes droits. Un droit autorise ou refuse l'accès à une fonction d'un produit par un utilisateur.

Important ! Un utilisateur peut appartenir à plusieurs profils.

Un utilisateur est autorisé à accéder à une fonction d'un produit si la permission connexe est ouverte dans au moins un des profils auquel il appartient.

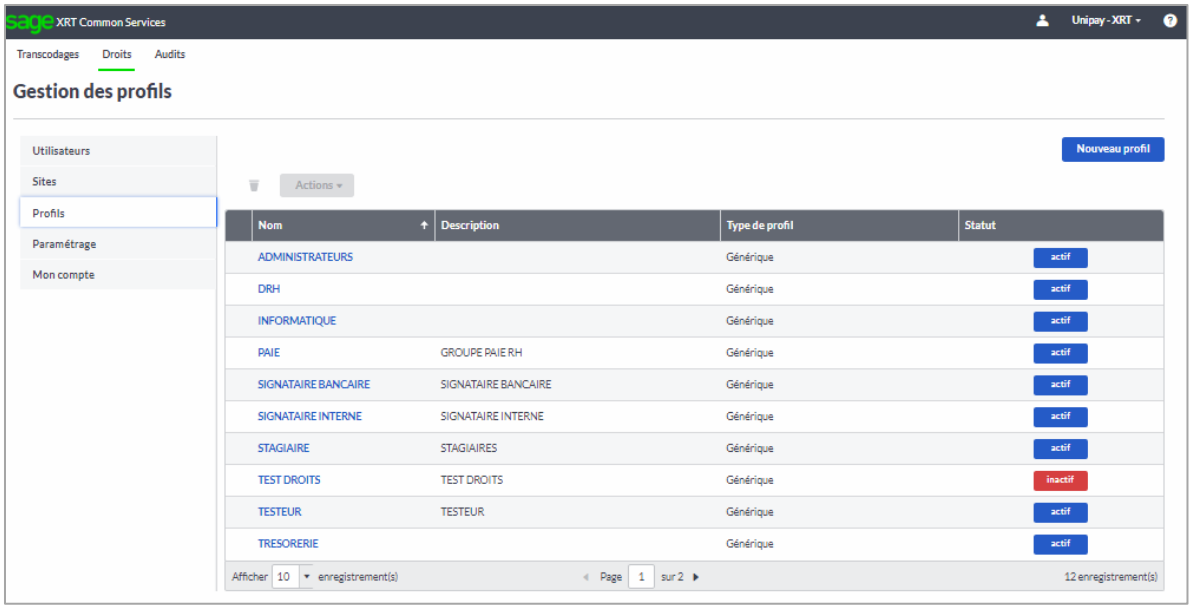
UMAPI exécute une opération de type *OU* sur les permissions. Ce mode de fonctionnement permet d'associer un profil à un groupe de personnes ayant les mêmes activités.

Un profil standard possède les propriétés suivantes :

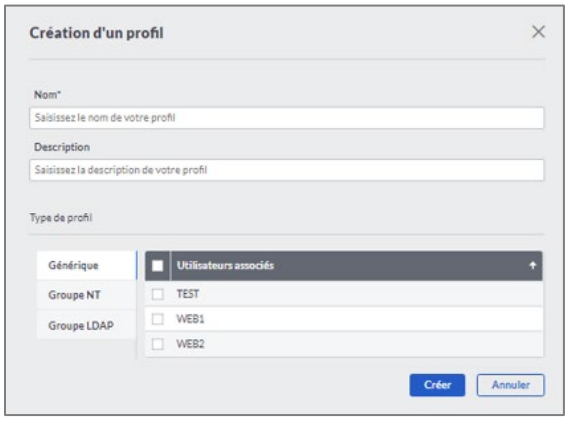
- un **Code** qui identifie le profil (sans espaces)
- une Description

Création

Dans le menu **Droits**, cliquez sur l'onglet **Profils**.



Dans l'onglet **Profils**, le bouton **Nouveau profil** vous permet d'accéder à l'assistant de création.



Renseignez les informations suivantes :

- **Nom** : saisissez un nom pour le profil. Ce champ doit être renseigné obligatoirement.
- **Description** : saisissez une description pour le profil.

Sélectionnez le type de profil :

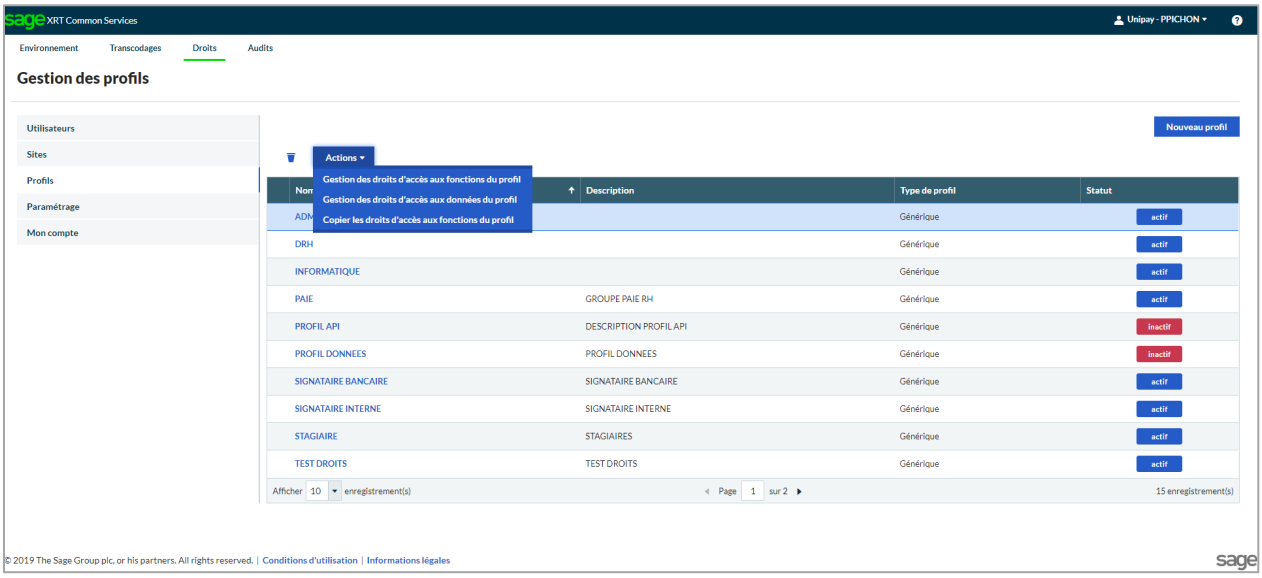
- **Générique** : sélectionnez les utilisateurs existants à associer au profil
- **Groupe NT** : tout utilisateur membre du groupe est automatiquement enregistré dans la base de données comme utilisateur des applications *Sage XRT*. Le profil de type *Groupe AD* s'appuie sur les données relatives aux comptes utilisateurs *Windows NT*. Sélectionnez la langue et le niveau de sécurité par défaut.
- **Groupe LDAP** : le profil de type *Groupe LDAP* s'appuie sur les données relatives à un annuaire d'entreprise. La création d'un groupe LDAP est effective uniquement si l'accès à l'annuaire d'entreprise a été paramétré. Sélectionnez la langue et le niveau de sécurité par défaut.

Lors de la création d'un profil NT ou d'un profil LDAP, tout utilisateur membre du groupe est automatiquement enregistré dans la base de données comme utilisateur des applications *Sage XRT*.

Cliquez sur le bouton **Créer** pour valider la création du profil.

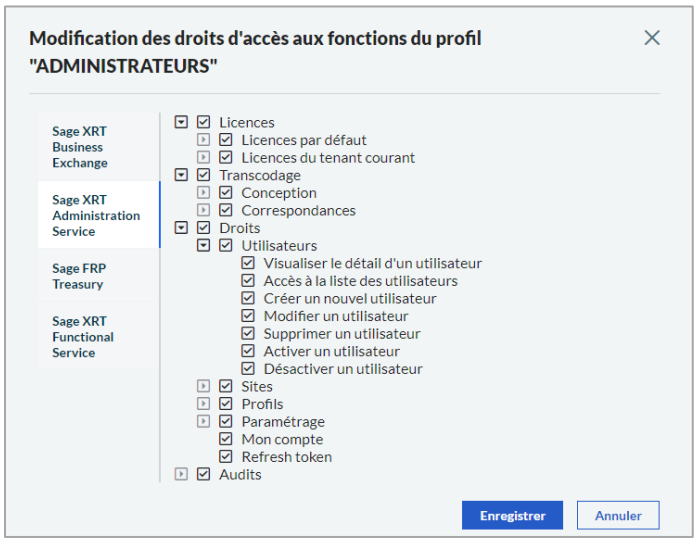
Droits d'accès

A partir de la liste des profils, sélectionnez un profil et sélectionnez l'action **Gérer les droits d'accès aux fonctions du profil**.



La page **Attribution des droits** permet de gérer les droits d'accès aux fonctions d'un profil aux différents produits *Sage XRT* installés sur le serveur.

Cliquez sur l'onglet **Sage XRT Administration Service**.



Activez les droits à accorder.

Répétez l'opération pour les produits **Sage XRT Treasury**, **Sage XRT Business Exchange** et **Sage XRT Functional Service** (service de fonctionnalités à venir de **Sage XRT Common Services**) si vous souhaitez attribuer des droits d'accès à ces deux produits pour le profil.

Cliquez sur **Enregistrer** pour enregistrer les modifications effectuées ou sur **Annuler** pour annuler les modifications.

Gestion des droits d'accès aux fonctions du profil

Cette fonctionnalité est exploitée par **Sage XRT Functional Service** en mode autonome, i.e. non intégré à un autre produit tel que **Sage XRT Business Exchange** ou **Sage XRT Advanced**.

A partir de la liste des **Profils**, sélectionnez un profil et sélectionnez l'action **Gestion des droits d'accès aux données du profil**.

Par type de données (onglets **Origine**, **Entité**, **Banque**, **Compte**, **Devise**), sélectionnez les éléments ne devant pas être accessibles aux utilisateurs rattachés à ce profil. Par défaut, tous les éléments sont accessibles.

Cliquez sur le bouton **Enregistrer** pour sauvegarder le paramétrage.

La case à cocher **Afficher uniquement les données inaccessibles**, décochée par défaut, permet de filtrer les éléments affichés pour ne faire apparaître que les éléments inaccessibles.

Duplication

A partir de la liste des **Profils**, sélectionnez un profil, puis l'action **Copier les droits d'accès aux fonctions du profil**. L'assistant de création d'un **Profil** s'affiche.

Complétez les informations du profil et cliquez sur **Créer.** Ce nouveau profil rassemble automatiquement les droits d'accès aux fonctions du profil copié.

Modification

Dans le menu **Droits**, cliquez sur l'onglet **Profils**. La liste des profils existants s'affiche.

Pour modifier un profil, utilisez le lien disponible sur le nom du profil.

Procédez aux modifications souhaitées et cliquez sur le bouton **Enregistrer**.

Suppression

Dans le menu **Droits**, cliquez sur l'onglet **Profils**. La liste des profils existants s'affiche.

Pour supprimer un profil, sélectionnez la case à cocher correspondante et cliquez sur l'icône *corbeille*.

Activation

Tout profil créé obtient le statut **Inactif** et devra faire l'objet d'une activation par un autre utilisateur de niveau Administrateur.

A partir de la liste des **Profils**, sélectionnez le profil à activer.

Cliquez sur le statut **Inactif** et confirmez l'activation du profil.

Un message vous confirme l'activation du profil.

Désactivation

A partir de la liste des **Profils**, sélectionnez le profil à désactiver.

Cliquez sur le statut **Actif** et confirmez la désactivation du profil.

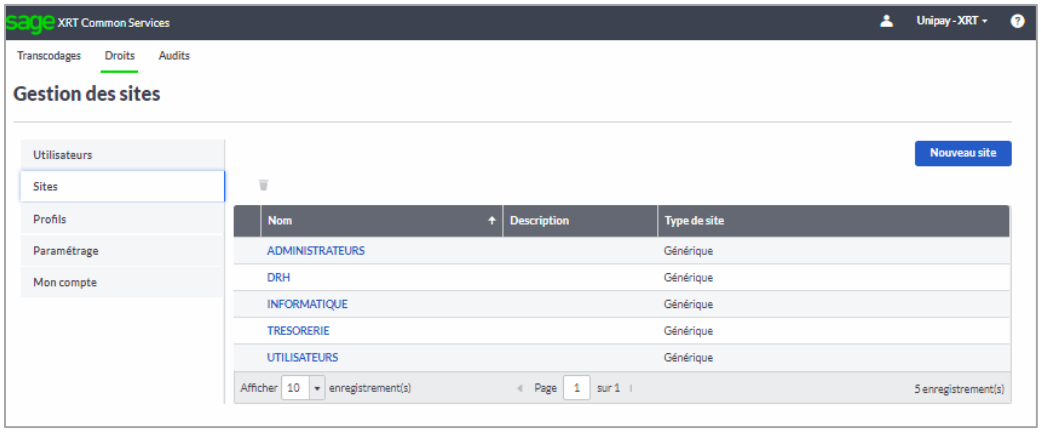
Un message vous confirme la désactivation du profil.

Sites

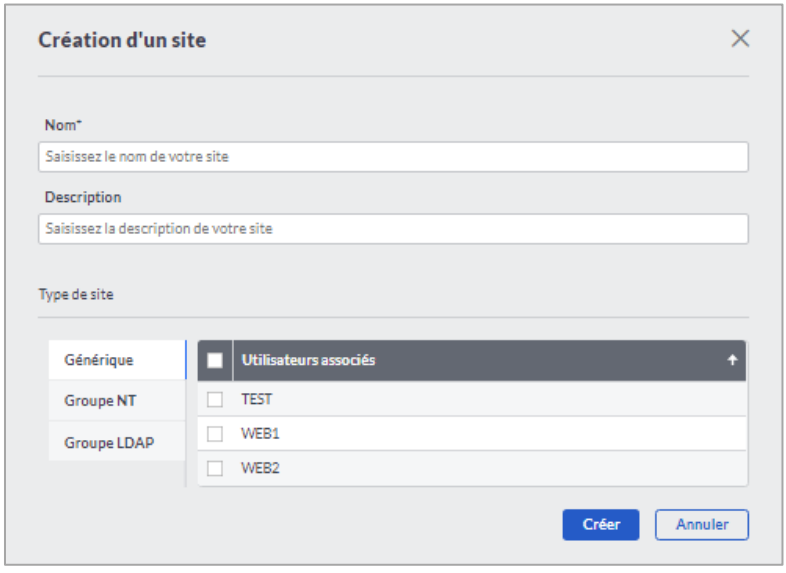
Création

La création des sites suit les mêmes règles que celle des profils.

Dans le menu **Droits**, cliquez sur l'onglet **Site**.



Le bouton **Nouveau site** vous permet d'accéder à l'assistant de création.



Renseignez les informations suivantes :

- **Nom** : saisissez un nom pour le site. Ce champ doit être renseigné obligatoirement.
- **Description** : saisissez une description pour le site.

Sélectionnez le type de site :

- **Générique** : sélectionnez les utilisateurs existants à associer au site
- **Groupe AD** : tout utilisateur membre du groupe est automatiquement enregistré dans la base de données comme utilisateur des applications *Sage XRT*. Le site de type **Groupe AD** s'appuie sur les données relatives aux comptes utilisateurs *Windows NT*.
- **Groupe LDAP** : le site de type **Groupe LDAP** s'appuie sur les données relatives à un annuaire d'entreprise. La création d'un groupe LDAP est effective uniquement si l'accès à l'annuaire d'entreprise a été paramétré.

Cliquez sur le bouton **Créer** pour valider la création du site.

Modification

Dans le menu **Droits**, cliquez sur l'onglet **Site**. La liste des sites existants s'affiche.

Pour modifier un site, utilisez le lien disponible sur le nom du site.

Procédez aux modifications souhaitées et cliquez sur le bouton **Enregistrer**.

Suppression

Dans le menu **Droits**, cliquez sur l'onglet **Site**. La liste des sites existants s'affiche.

Pour supprimer un site, sélectionnez la case à cocher correspondante et utilisez l'icône *corbeille*.

Activation

Tout site créé obtient le statut **Inactif** et devra faire l'objet d'une activation par un autre utilisateur de niveau Administrateur.

A partir de la liste des sites, sélectionnez le site à activer.

Cliquez sur le statut **Inactif** et confirmez l'activation du site. Un message vous confirme l'activation du site.

Désactivation

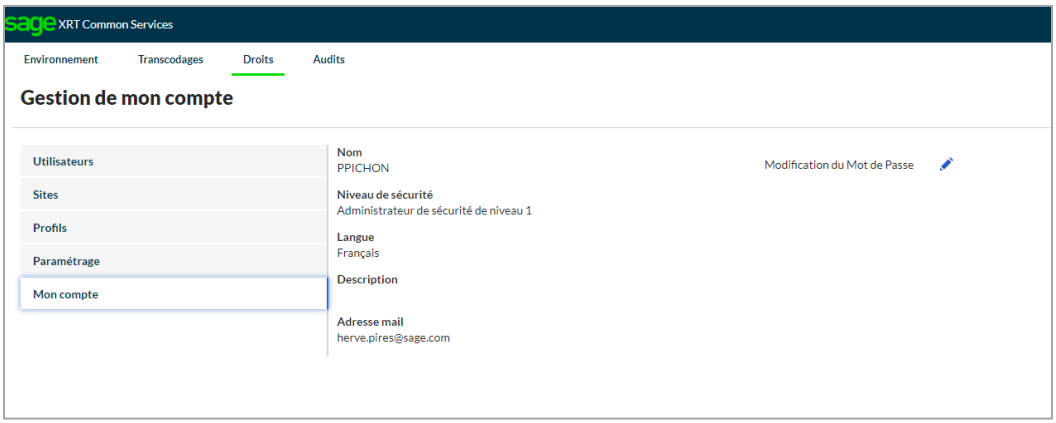
A partir de la liste des sites, sélectionnez le site à désactiver.

Cliquez sur le statut **Actif** et confirmez votre action. Un message vous confirme la désactivation du site.

Mon compte

Cet onglet est accessible aux utilisateurs de type **Standard**. Elle permet de modifier le mot de passe de l'utilisateur connecté.

Dans le menu **Droits**, cliquez sur l'onglet **Mon compte**.



Les informations de l'utilisateur sont rappelées : **Nom**, **Niveau de sécurité**, **Langue**, **Description**, **Adresse mail**.

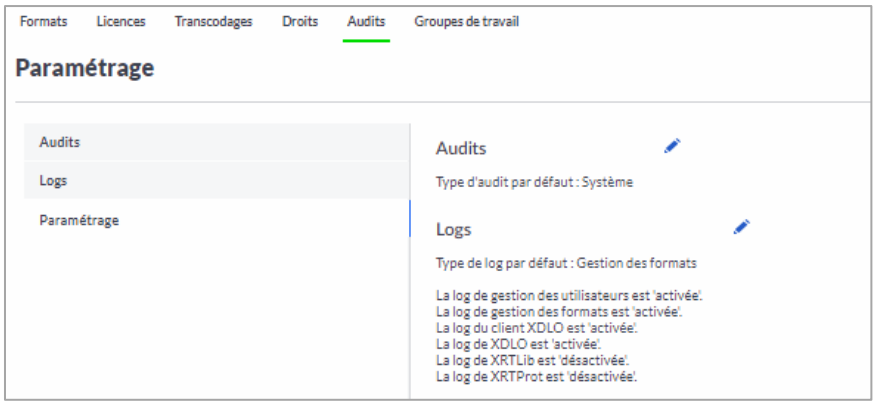
Seul le mot de passe peut être modifié en utilisant l'icône *crayon*.

Audits et logs

Paramétrage

Vous devez définir le type d'audit présenté ainsi que l'activation des logs avant de pouvoir consulter des informations.

Dans le menu **Audits**, cliquez sur l'onglet **Paramétrage**.



Le paramétrage est ici résumé. Il peut être modifié en cliquant sur l'icône *crayon*.

Sélectionnez le type d'audit à proposer par défaut :

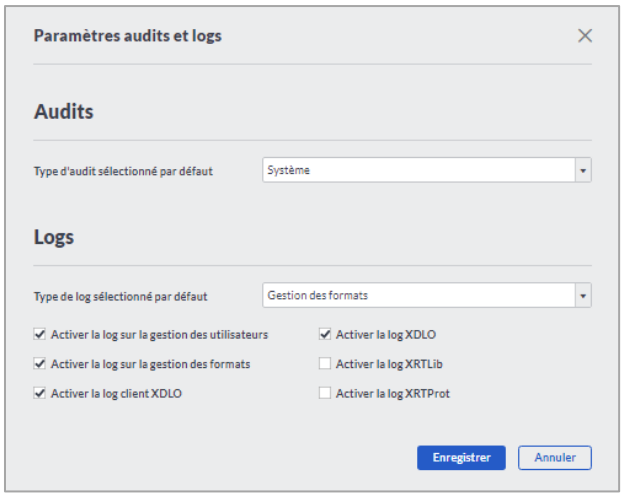
- Système
- Base de données
- Utilisateurs

Sélectionnez le type de log à proposer par défaut :

- Gestion des formats
- Gestion des utilisateurs
- Gestion base de données
- Gestion de la console
- Client XDLO
- XDLO
- Service XDLO
- XRTProt
- XRTLogin
- XRTLib

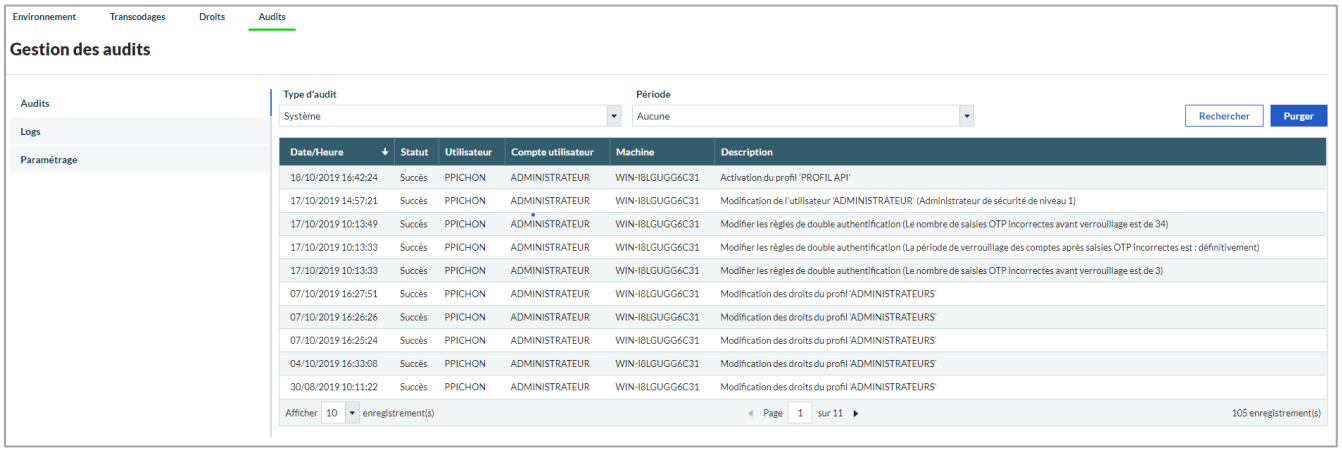
Sélectionnez les logs à activer.

Cliquez sur le bouton **Enregistrer** pour terminer votre paramétrage.



Audit

Dans le menu **Audits**, cliquez sur l'onglet **Audit**. Le type d'audit défini par défaut s'affiche.



Pour modifier le type par défaut, sélectionnez une option dans la liste déroulante **Type d'audit**.

Pour filtrer les informations du tableau, sélectionnez une **Période** dans liste déroulante :

- Aujourd'hui
- 7 derniers jours
- 30 derniers jours
- 12 derniers mois
- Cette semaine

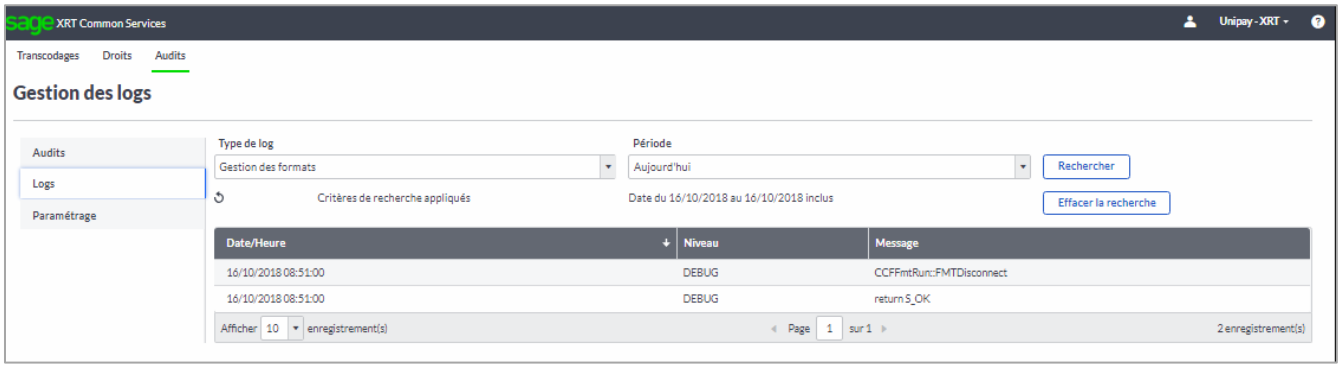
- Ce mois
- Cette année

Vous pouvez accéder à d'autres critères de sélection en utilisant le bouton **Rechercher**. Les critères de filtre appliqués sont rappelés au-dessus de la liste.

Cliquez sur le bouton **Purger** pour supprimer ou exporter des évènements de la liste.

Log

Dans le menu **Audits**, cliquez sur l'onglet **Log**. Le type de log par défaut s'affiche.



Pour le modifier, sélectionnez un **Type de log** dans la liste déroulante.

Pour filtrer les informations du tableau, sélectionnez une **Période** dans liste déroulante :

- Aujourd'hui
- 7 derniers jours
- 30 derniers jours
- 12 derniers mois

- Cette semaine
- Ce mois
- Cette année

Vous pouvez accéder à d'autres critères de sélection en utilisant le bouton **Rechercher**. Les critères de filtre appliqués sont rappelés au-dessus de la liste.

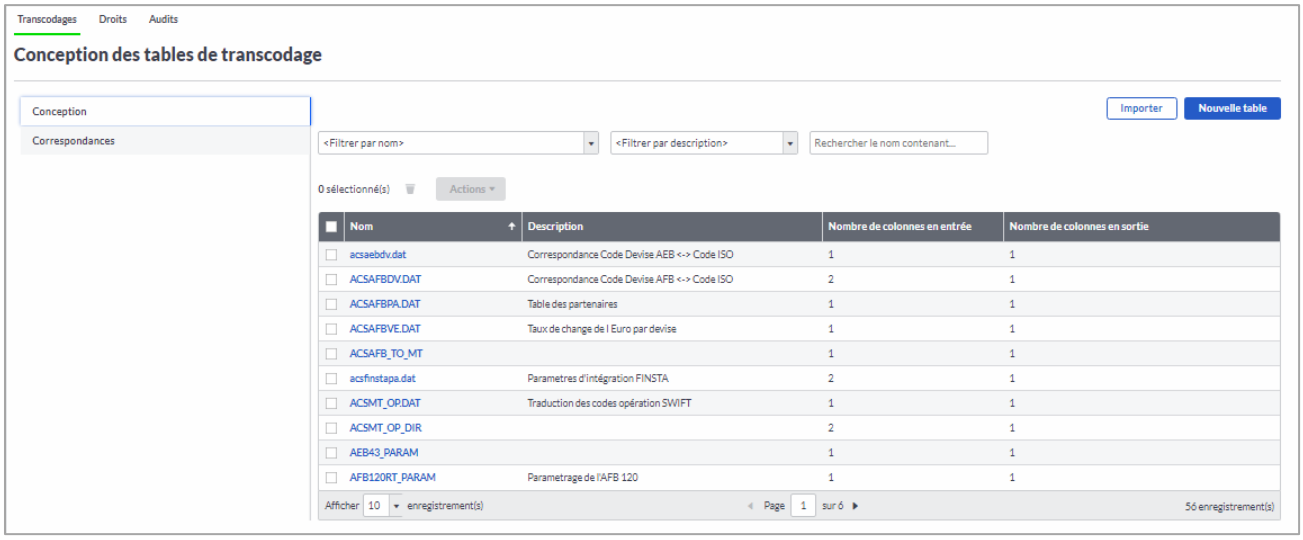
Transcodages

Conception

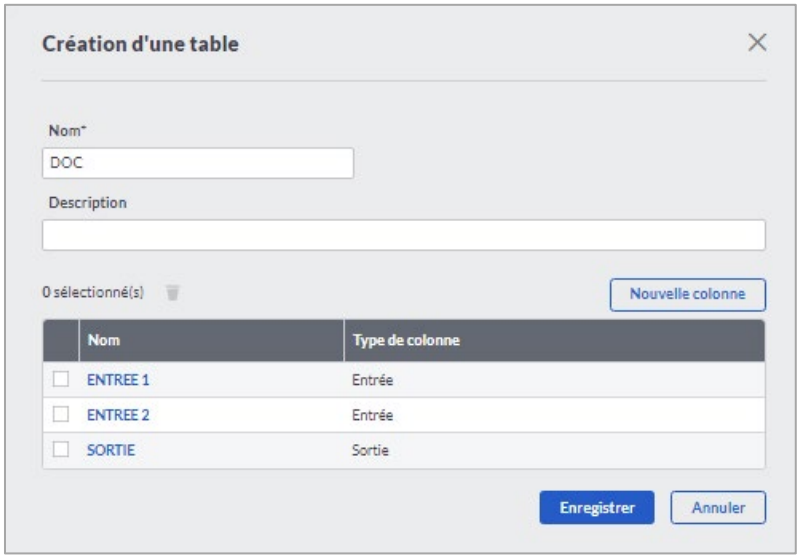
Création

Dans le menu **Transcodages**, cliquez sur l'onglet **Conception**.

La liste des tables livrées par défaut pour assurer le fonctionnement des produits *Sage XRT* s'affiche.



Le bouton **Nouvelle table** vous permet d'accéder à l'assistant de création.



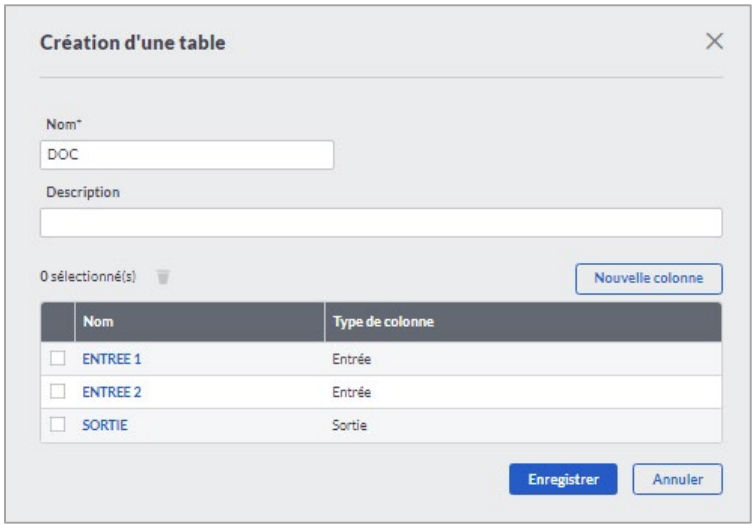
Renseignez obligatoirement un **Nom** pour la table et éventuellement une **Description**.

Le bouton **Nouvelle colonne** vous permet d'accéder à l'assistant de création des colonnes d'entrée et de sortie de la table.

Pour chaque colonne, renseignez obligatoirement un **Nom** et sélectionnez le **Type de colonne** (**Entrée** ou **Sortie**).

Cliquez sur **Valider** pour enregistrer la création de la colonne.

La colonne créée s'affiche dans la liste de colonnes constituant la table. Une colonne peut être modifiée en utilisant le lien sur son nom ou supprimée en la sélectionnant (case à cocher) et en utilisant l'icône *corbeille*.



Cliquez sur le bouton **Enregistrer** pour enregistrer la création de la table. La table s'affiche alors dans la liste.

Modification

Dans le menu **Transcodages**, cliquez sur l'onglet **Conception**. La liste des tables existantes s'affiche.

Pour modifier une table, utilisez le lien disponible sur le nom de la table.

Procédez aux modifications souhaitées et cliquez sur le bouton **Enregistrer**.

Suppression

Dans le menu **Transcodages**, cliquez sur l'onglet **Conception**. La liste des tables existantes s'affiche.

Pour supprimer une table, sélectionnez la case à cocher correspondante et utilisez l'icône *corbeille*.

Import

Dans le menu **Transcodages**, cliquez sur l'onglet **Conception**. La liste des tables existantes s'affiche.

Pour importer des tables de transcodage, cliquez sur le bouton **Importer**. Une boite de dialogue pour sélectionner le fichier à importer s'ouvre.

Sélectionnez un fichier et cliquez sur **Ouvrir**.

Export

Dans le menu **Transcodages**, cliquez sur l'onglet **Conception**. La liste des tables existantes s'affiche.

Pour exporter une table, activez la case à cocher correspondante, puis sélectionnez **Exporter** dans la liste déroulante **Actions**.

Correspondances

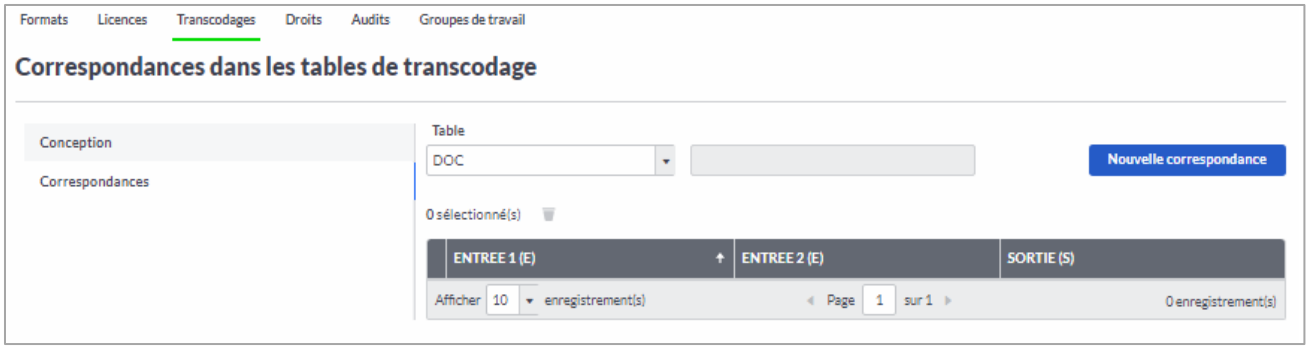
Une fois la table créée, les correspondances à appliquer doivent être renseignées.

Création

Dans le menu **Transcodages**, cliquez sur l'onglet **Correspondances**.

La table **RIBS-Table IBAN** est sélectionnée par défaut dans la liste déroulante **Table** pour l'affichage des correspondances.

Sélectionnez dans la liste déroulante la table pour laquelle les correspondances doivent être créées. La structure de la table s'affiche.

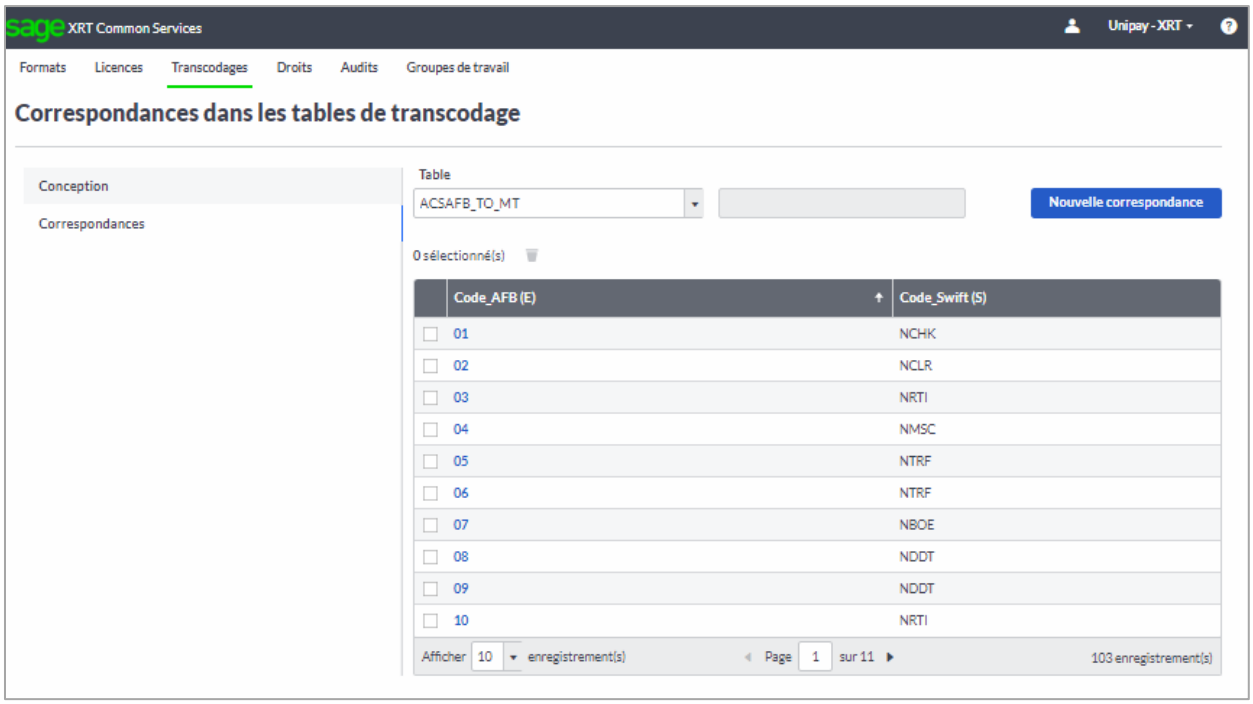


Le bouton **Nouvelle correspondance** vous permet d'accéder à l'assistant de création.

Renseignez pour chaque correspondance, des valeurs d'entrée et de sortie.

Cliquez sur **Enregistrer** pour valider la création de la correspondance, qui s'affiche dans la liste des correspondances de la table.

La table est créée en statut **Inactif**.



Modification

A partir du menu **Transcodages**, cliquez sur l'onglet **Correspondances**.

Sélectionnez la table de travail. La liste des correspondances existantes s'affiche.

Pour modifier une correspondance, utilisez le lien disponible sur la première colonne de la table.

Procédez aux modifications souhaitées et cliquez sur le bouton **Enregistrer**.

Suppression

Dans le menu **Transcodages**, cliquez sur l'onglet **Conception**.

Sélectionnez la table de travail. La liste des correspondances existantes s'affiche.

Pour supprimer une correspondance, sélectionnez la case à cocher correspondante et utilisez l'icône *corbeille*.

XDLO (deprecated)

Le service XDLO n'existe plus.

Sage XRT Functional Service

Configuration

Le fichier de configuration *Sage.SCDTSServer.Service.exe.config* pour le service des fonctionnalités se situe par défaut sous : **C:\Program Files\Common Files\xrt**.

Activation des logs

Cf. nœud <system.diagnostics> et <diagnostics>

Définition de l'emplacement du site Web, des ports d'écoute et des hosts des services

Cf. nœud <ApplicationSettings>

```
<add key="websitehost" value="*" />

<add key="websitehostdefault" value="http://localhost" />

<add key="httpservicehost" value="http://*:80"/>

<add key="httpsservicehost" value="https://*:443"/>
```

Définition du compte et de la fréquence en seconde de dépilement des jobs APIFMT

Cf. nœud <ApplicationSettings>

```
<add key="apifmtasyncuser" value="XRT" />

<add key="apifmtasyncfrequency" value="20" />
```

Définition du compte et de la fréquence en seconde de dépilement des jobs de création des paiements

Cf. nœud <ApplicationSettings>

```
<add key="payasyncuser" value="XRT" />

<add key="payasyncfrequency" value="30" />
```

Définition du nombre de threads d'actions sur flux JSON des paiements

Cf. nœud <ApplicationSettings>

```
<add key="payasyncthreadtaskI" value="10" />
```

```
<add key="payasyncthreadtaskII" value="10" />
```

Notes :

taskI => dématérialisation vers la base de données

taskII => création depuis la base de données

Définition des tenants à exclure lors du dépilement des jobs (APIFMT, PAIEMENTS)

Cf. nœud <ApplicationSettings>

```
<add key="apifmtasyncexception"
value="[YOUR_WORKGROUP_EXCEPT1],[YOUR_WORKGROUP_EXCEPT2]" />

<add key="payasyncexception"
value="[YOUR_WORKGROUP_EXCEPT1],[YOUR_WORKGROUP_EXCEPT2]"/>
```

Définition d'éléments de sécurité

Cf. nœud <ApplicationSettings>

```
<add key="showfriendlymessage" value="NO"/> (messages d'erreur
générique)
```

Paramétrage du lien avec un autre produit (RAPI) (postage et récupération statut)

Cf. nœud <ApplicationSettings>

```
<add key="paysendfilepostgen" value=" "/>

<add key="payfollowfilepostgen" value=" "/>
```

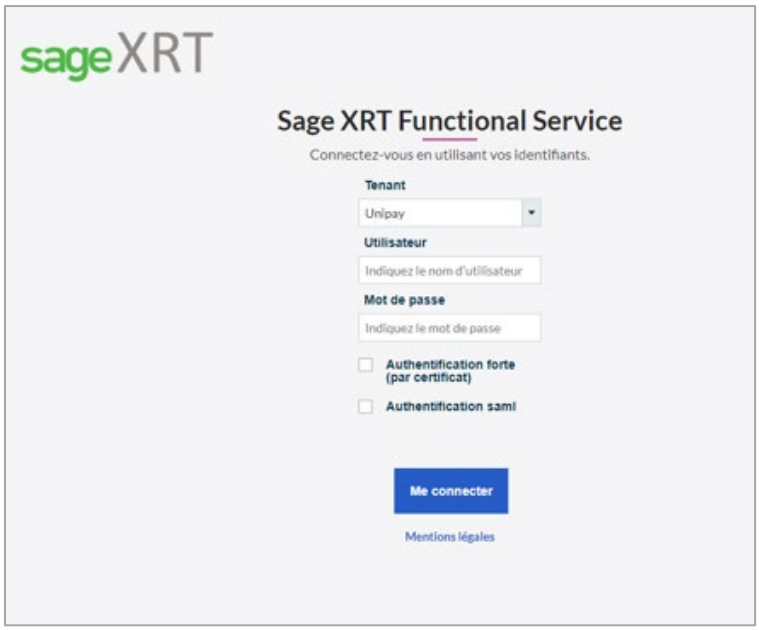
Documentation SWAGGER

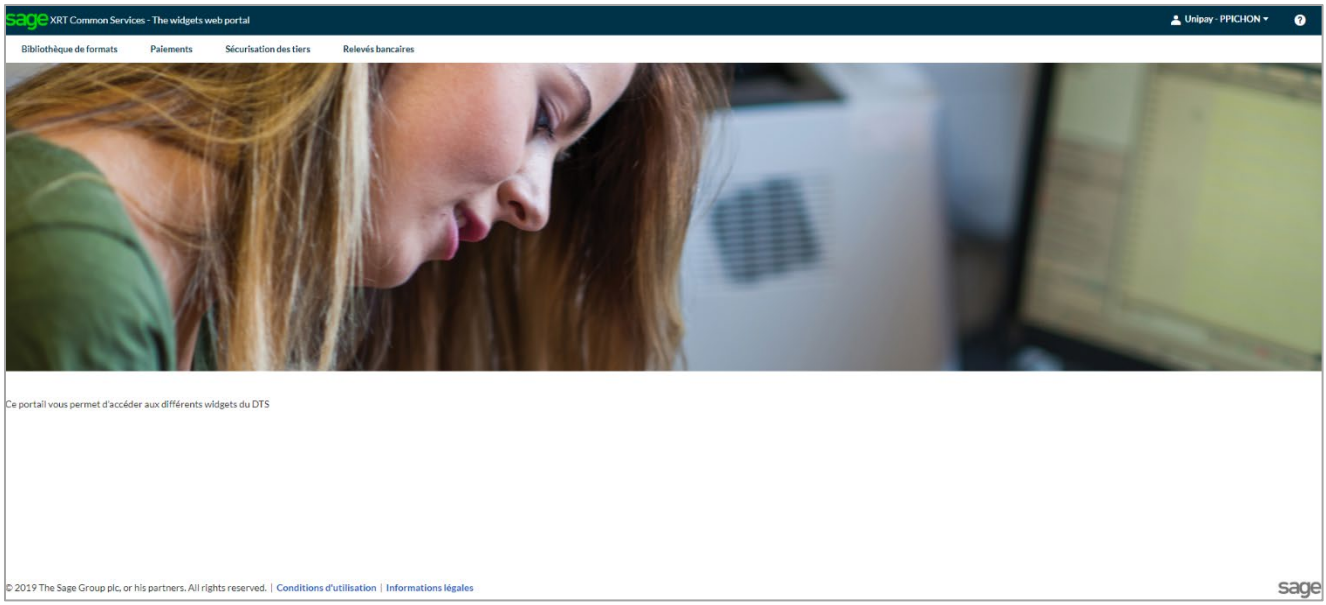
URL de documentation et URL d'exportation

Connexion

L'utilisation de l'interface de **Sage XRT Functional Service** se fait via l'URL **http://localhost/SCDTS/index.html** et nécessite le démarrage des services d'authentification et des fonctionnalités.

Sage XRT Administration Service





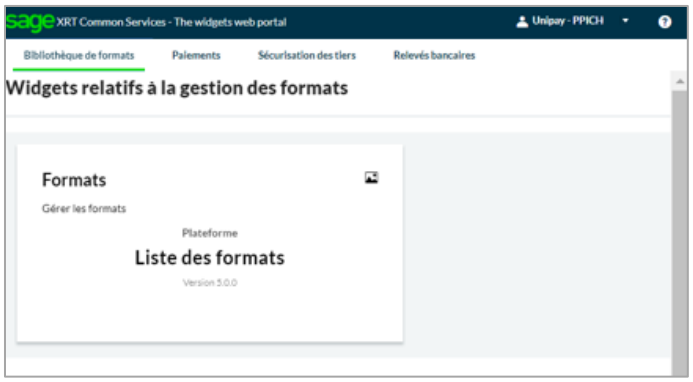
Ce service offre des fonctionnalités autour des thèmes :

- Bibliothèque des formats
- Gestion des paiements
- Sécurisation des tiers (non disponible en version 5.00)
- Relevés bancaires (non disponible en version 5.00)

Les fonctionnalités décrites ci-après correspondent à une utilisation du service en mode autonome, i.e. non intégré à un autre produit tel que **Sage XRT Business Exchange** ou **Sage XRT Advanced**.

Bibliothèque des formats - API Formats

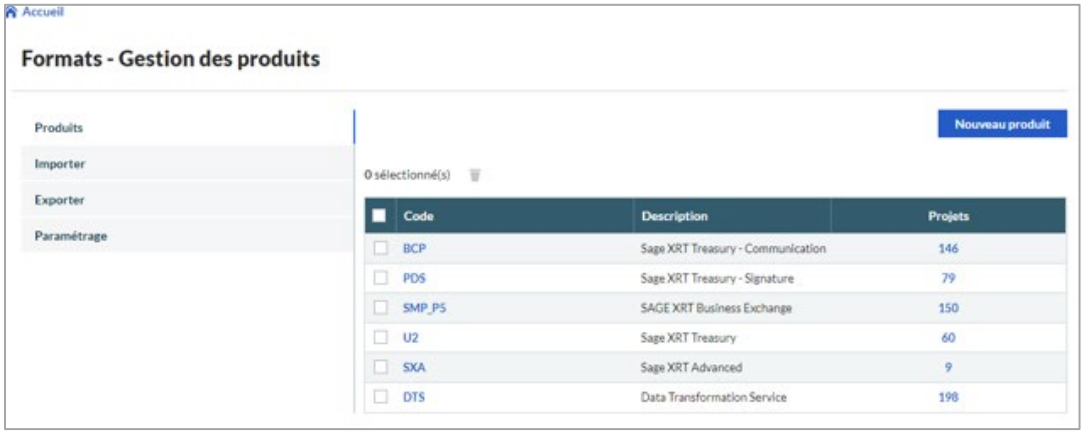
Pour accéder à la bibliothèque des formats, utilisez le widget **Liste des formats**.



Création de produit

Les éléments liés à la gestion des formats (projets, traitements) sont accessibles par **Produit**.

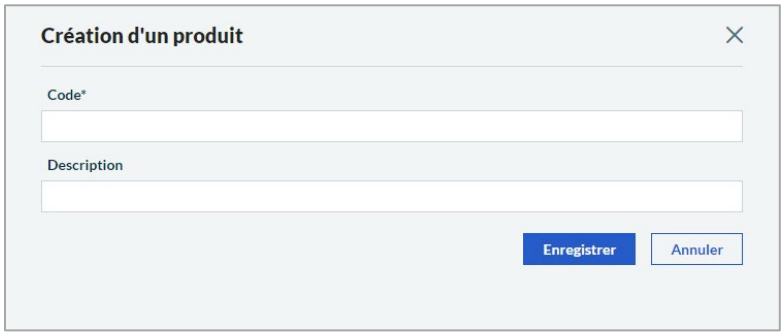
Les produits utilisables sont livrés par défaut avec l'ensemble de leurs projets et traitements associés en standard.



Dans l'onglet **Produit**, le bouton **Nouveau produit** vous permet d'accéder à l'assistant de création.

Renseignez obligatoirement un **Code** et optionnellement une **Description**.

Cliquez sur le bouton **Enregistrer** pour sauvegarder la création du produit ou sur le bouton **Annuler** pour revenir à la liste des produits sans créer le produit. Un message confirme la création du produit.



Modification de produit

A partir de la liste des produits, utilisez le lien sur le Code du produit. La fenêtre de modification s'affiche.

Seule la **Description** est modifiable.

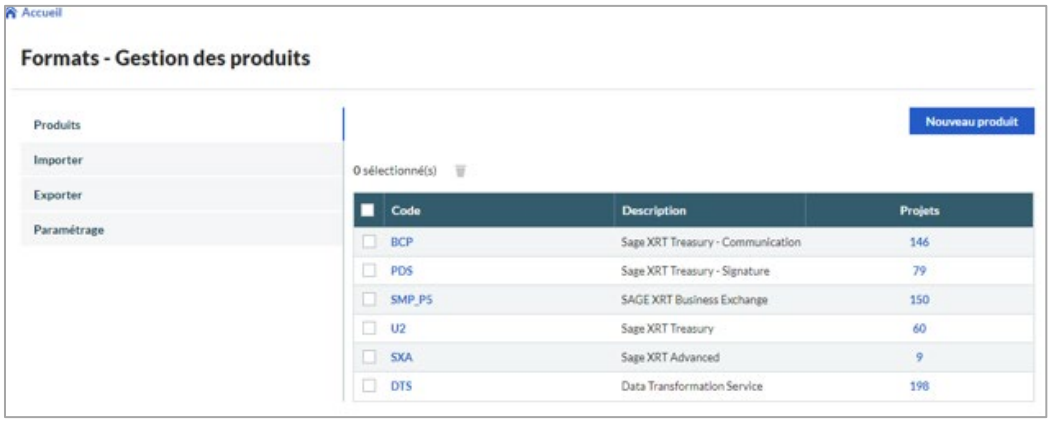
Cliquez sur le bouton **Enregistrer** pour sauvegarder la modification du produit ou sur le bouton Annuler pour revenir à la liste des produits sans modifier le produit. Un message confirme la modification du produit.

Suppression de produit

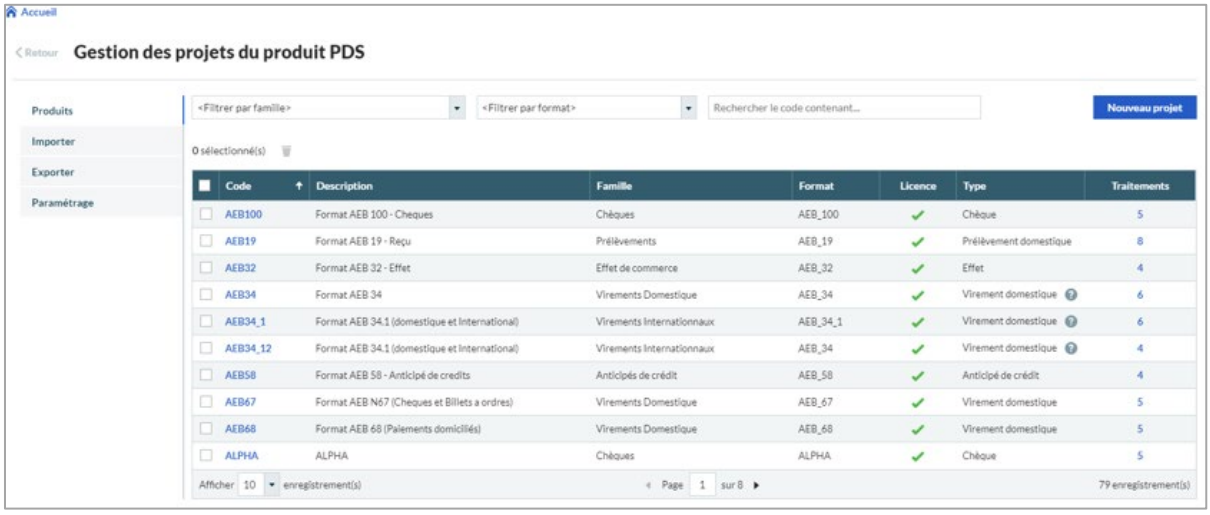
Pour supprimer un produit, sélectionnez-le dans la liste des produits, puis cliquez sur l'icône corbeille. Un message vous confirme la suppression du produit.

Création de projet

Pour accéder à la liste des projets associés en standard à un produit, cliquez sur le nombre indiqué dans la colonne **Projets**.



La liste des projets du produit s'affiche.



Cette liste peut être filtrée par format et par famille : virements internationaux, relevés de compte, etc.

Vous pouvez également rechercher directement le projet en indiquant tout ou partie du code projet.

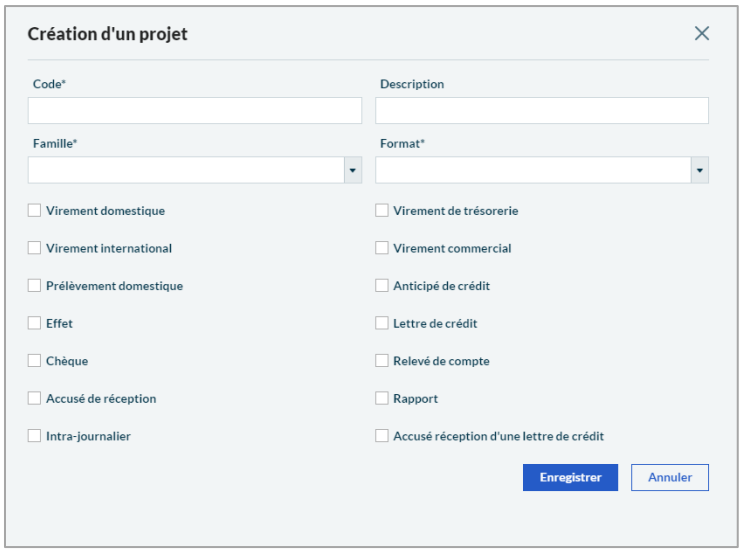
Par chaque projet identifié par un code, la liste présente :

- Une description
- La famille de rattachement
- Le format rattaché
- L'accessibilité de ce format en fonction de la clé de licence
- L'accès aux traitements associés à ce projet

Dans l'onglet **Produits**, le bouton **Nouveau projet** vous permet d'accéder à l'assistant de création.

Renseignez obligatoirement le **Code**, la **Famille** et le **Format**.

Cliquez sur le bouton **Enregistrer** pour sauvegarder la création du projet.



Un message confirme la création du projet.

Modification de projet

A partir de la liste des projets, utilisez le lien sur le **Code** du projet. La fenêtre de modification s'affiche.

Le **Code** n'est pas modifiable.

Cliquez sur le bouton **Enregistrer** pour sauvegarder la modification du projet ou sur le bouton **Annuler** pour revenir à la liste des projets sans modifier le projet.

Un message confirme la modification du projet.

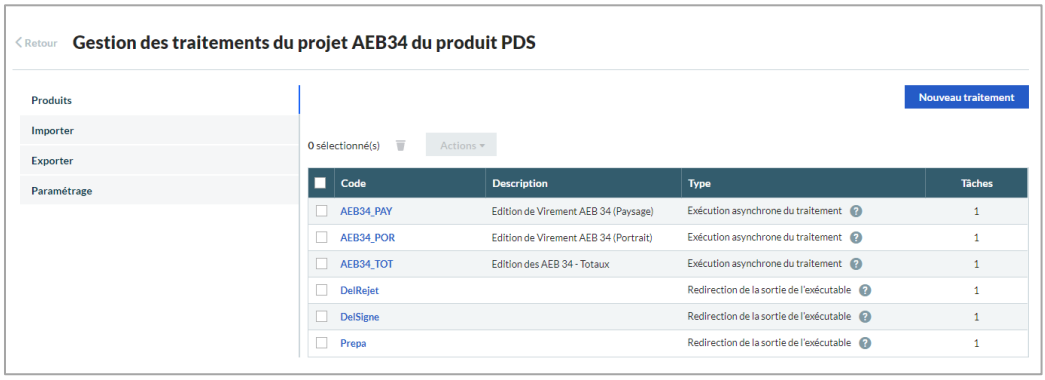
Suppression de projet

A partir de la liste des projets, sélectionnez le projet à supprimer et cliquez sur l'icône *corbeille*.

Un message confirme la suppression du projet.

Création de traitement

Pour accéder à la liste des traitements associés en standard à un projet, cliquez sur le nombre indiqué dans la colonne **Traitements**.



Par chaque traitement identifié par un code, la liste présente :

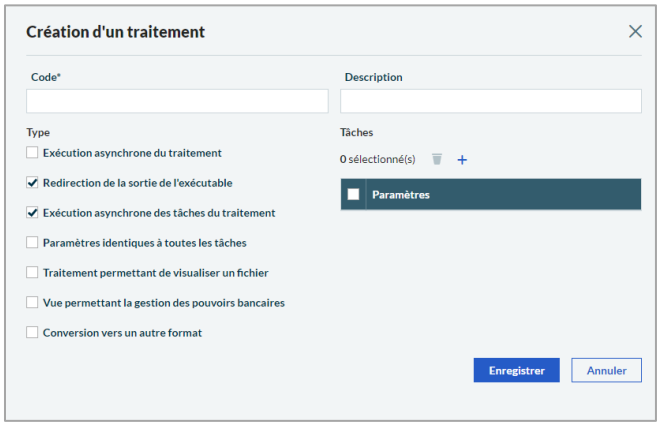
- Une description
- Le type de traitement
- Le nombre de tâches

Dans l'onglet **Produits**, le bouton **Nouveau traitement** vous permet d'accéder à l'assistant de création.

Renseignez obligatoirement le **Code**.

Pour gérer les tâches au traitement, utilisez l'icône **+** pour ajouter une tâche ou *corbeille* pour supprimer des tâches.

Cliquez sur le bouton **Enregistrer** pour sauvegarder la création du traitement. Un message confirme la création du traitement.



Modification de traitement

A partir de la liste des traitements, utilisez le lien sur le **Code** du traitement.

La fenêtre de modification s'affiche.

Seul le **Code** n'est pas modifiable.

Cliquez sur le bouton **Enregistrer** pour sauvegarder la modification du traitement. Un message confirme la modification du traitement.

Suppression de traitement

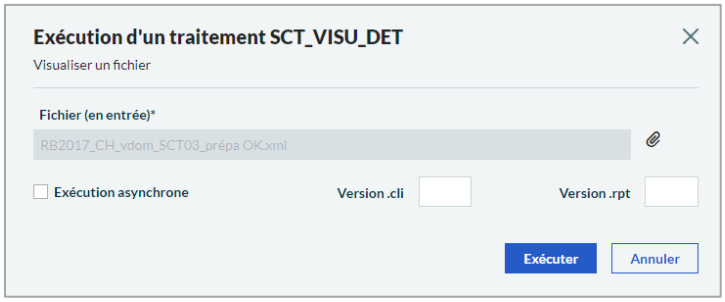
A partir de la liste des traitements, sélectionnez le traitement à supprimer, puis cliquez sur l'icône *corbeille*. Un message confirme la suppression du traitement.

Exécution de traitement

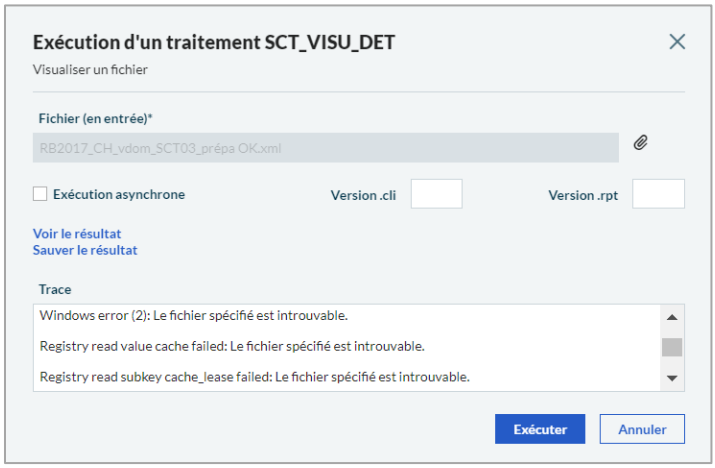
A partir de la liste des traitements, sélectionnez le traitement à exécuter.

Cliquez sur le bouton **Actions** et sélectionner l'action à exécuter.

Une fenêtre d'exécution s'affiche. Sélectionnez le fichier objet du traitement.



Cliquez sur le bouton **Exécuter** pour lancer le traitement.



Une fois le traitement terminé, utilisez le lien **Voir le résultat** pour accéder au résultat du traitement (ex. : PDF de visualisation d'un fichier) ou le lien **Sauver le résultat** pour enregistrer le résultat du traitement sans le visualiser.

Import

L'importation d'un fichier (apifmt.xml modifié) est possible via la fonction **Importer**.

Pour cela, cliquez dans le cadre pour ouvrir un explorateur de fichiers et sélectionner le fichier à importer.

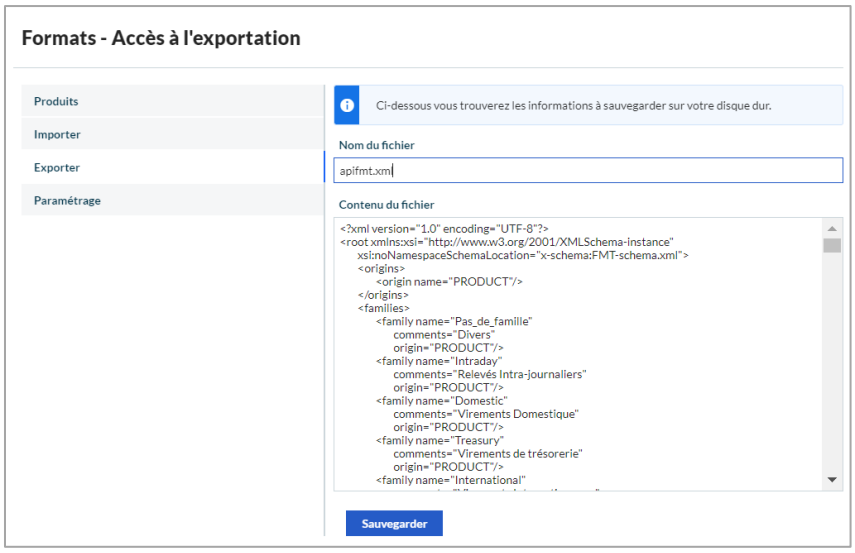
Cliquez sur le bouton **Importer**.

Export

L'exportation des éléments utilisés dans un fichier est possible via la fonction **Exporter**.

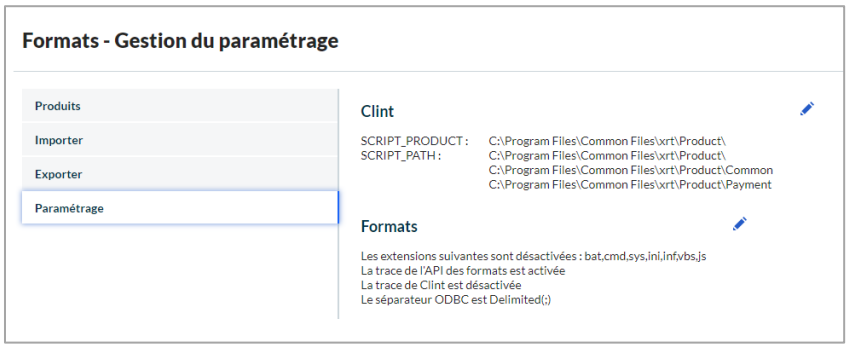
A l'appel de la fonction, le traitement d'exportation se lance automatiquement. Une fois l'export terminé, vous pouvez éventuellement modifier le nom du fichier *apifmt.xml* proposé par défaut et consulter son contenu.

Cliquez sur le bouton **Sauvegarder** pour enregistrer le fichier.

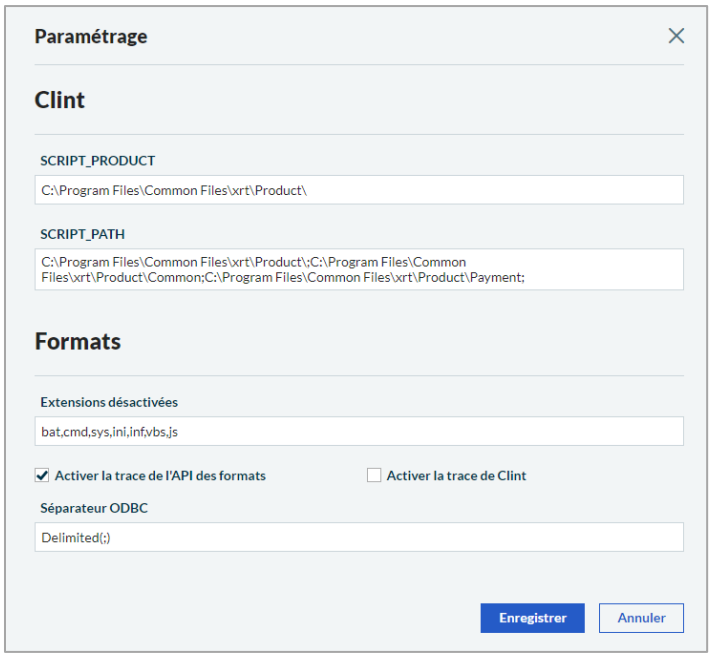


Paramétrage

Le résumé des paramétrages relatifs au moteur Clint et à la gestion des formats est affiché à l'appel de la fonction **Paramétrage**.



Ces paramétrages sont livrés par défaut et modifiables en utilisant l'icône *crayon*.



Cliquez sur le bouton **Enregistrer** pour sauvegarder les modifications effectuées ou sur le bouton **Annuler** pour les ignorer.

Paiements – Virements de masse - API Paiements

Cette fonctionnalité permet de gérer des transactions de virement (trésorerie ou fournisseurs) à partir d'un flux JSON contenant l'ensemble des transactions à traiter.

Flux JSON de transactions à traiter

Les informations surlignées en jaune sont obligatoires.

```
Flux JSON UTF-8 BOM

{

      "Transaction": [

            {

                  "TransactionType":

                  "IssuingAgentBIC" :

                  "IssuingAgentNationalId" :
                  "IssuingAgentContact" :

                  "IssuingAgentPhone" :
```

```
"IssuingAgentFax" :

"IssuingAgentName" :

"IssuingAgentTown" :

"IssuingAgentState" :

"IssuingAgentCountry" :

"IssuingAgentCurrency" :
"IssuingAgentInCurrencyIndicator" :
"DebtorAgentName":

"DebtorAgentBIC":

"DebtorAgentNationalId":
"DebtorAgentAdressLine1":
"DebtorAgentAdressLine2":
"DebtorAgentTown":

"DebtorAgentState":

"DebtorAgentCountryCode":
"DebtorAgentCountryLabel":
"DebtorAgentBankId":

"DebtorAgentBranchId":

"DebtorAgentContact":

"DebtorAgentPhone":

"DebtorAgentFax":

"DebtorAgentIdType":

"DebtorAgentId":

"DebtorAgentDomiciliation":
"InitiatingPartyName":

"InitiatingPartyNationalId":
"InitiatingPartyIdType":
"InitiatingPartyId":

"InitiatingPartyAdressLine1":
"InitiatingPartyAdressLine2":
"InitiatingPartyTown":
```

```
"InitiatingPartyState":

"InitiatingPartyProvince":
"InitiatingPartyCountryCode":
"InitiatingPartyContact":
"InitiatingPartyService":
"InitiatingPartyPhone":

"InitiatingPartyFax":

"InitiatingPartyEmail":

"InitiatingPartySenderId":
"InitiatingPartyContratNb":
"InitiatingPartyIssuerIndicator":
"DebtorName1":

"DebtorName2":

"DebtorAdressLine1":

"DebtorAdressLine2":

"DebtorTown":

"DebtorState":

"DebtorProvince":

"DebtorCountryCode":

"DebtorCountryLabel":

"DebtorNationalId":

"DebtorIdType":

"DebtorId":

"DebtorContact":

"DebtorService":

"DebtorPhone":

"DebtorFax":

"DebtorEmail":
```

```
"DebtorNonResidentIndicator":
"DebtorSenderId":

"DebtorContratNb":

"DebtorAcctCountryCode":
"DebtorAcctId":

"DebtorAcctLocalKey":

"DebtorAcctCurrency":

"DebtorAcctIssuerId":

"DebtorAcctType":

"DebtorAcctReferenceNumber":
"DebtorAcctIdCustomer":

"ChargesAcctCountryCode":
"ChargesAcctId":

"ChargesAcctLocalKey":

"ChargesAcctCurrency":

"ChargesAcctType":

"ChargesAcctBranchIdType":
"ChargesAcctBranchId":

"BeneficiaryName":

"BeneficiaryAdressLine1":
"BeneficiaryAdress Line2":
"BeneficiaryTown":

"BeneficiaryProvince":

"BeneficiaryCountryCode":
"BeneficiaryContact":

"BeneficiaryService":

"BeneficiaryPhone":

"BeneficiaryNationalId":
"BeneficiaryTypeId":

"BeneficiaryId":
```

```
"BeneficiaryAutorisationNb":
"BeneficiaryFax":

"BeneficiaryEmail":

"CreditorName":

"CreditorAdressLine1":

"CreditorAdressLine2":

"CreditorTown":

"CreditorProvince":

"CreditorState":

"CreditorCountryCode":

"CreditorCountryLabel":

"CreditorContact":

"CreditorService":

"CreditorPhone":

"CreditorNationalId":

"CreditorIdType":

"CreditorId":

"CreditorNonResidentIndicator":
"CreditorAutorisationNb":
"CreditorFax":

"CreditorEmail":

"CreditorAgtName":

"CreditorAgtBIC":

"CreditorAgtAdressLine1":
"CreditorAgtAdressLine2":
"CreditorAgtTown":

"CreditorAgtState":
```

```
"CreditorAgtCountryCode":
"CreditorAcctId":

"CreditorAcctCurrency":

"CreditorAcctDomiciliation":
"CreditorAcctIdType":

"CreditorAcctId":

"CreditorAcctBankId":

"CreditorAcctBIC1":

"CreditorAcctType":

"IntermdediaryAgtBIC":

"IntermdediaryAgtOccasionnalIndicator":
"IntermdediaryAgtIdType":
"IntermdediaryAgtId":

"IntermdediaryAgtName":

"IntermdediaryAgtAdressLine1":
"IntermdediaryAgtAdressLine2":
"IntermdediaryAgtAdressLine3":
"IntermdediaryAgtCountryCode":
"TransactionAmount":

"TransactionAmountDecimalNb":
"TransactionTransfertCurrency":
"TransactionReference":

"TransactionExecutionDate":
"TransactionFeesImputation":
"TransactionPaymentMode":
"TransactionTransactionCode":
"TransactionInternalNumber":
"TransactionStatus":

"TransactionValueDate":

"TransactionReason":

"TransactionCurrencyPurchaseIndicator":
"TransactionCurrencyPurchaseContractReference":
"TransactionCurrencyPurchaseDate":
"TransactionCurrencyPurchaseRate":
"TransactionBeneficiaryAlertIndicator":
```

```
"TransactionBeneficiaryAlertMode":
"TransactionBeneficiaryAlertNb":
"TransactionBankBeneficiaryAlertIndicator":
"TransactionBankBeneficiaryAlertMode":
"TransactionBankBeneficiaryAlertNb":
"TransactionContratSettlement":
"TransactionIntraCompanyIndicator":
"TransactionNettingIndicator":
"TransactionUrgentIndicator":
"TransactionRTGSIndicator":
"TransactionBilateralInstruction":
"TransactionBeneficiaryIdentificationIndicator":
"TransactionExclusivePaymentIndicator":
"TransactionFreeText":

"TransactionEconomicReason":
"TransactionEconomicCountry":
"TransactionBranchId":

"TransactionInvoiceReference":
"TransactionInvoiceDate":
"TransactionInvoicePaidAmount":
"TransactionInvoiceOriginalAmount":
"TransactionDiscountAmount":
"TransactionDeclaration": [
            {

            "DeclarationAmount" :

            "DeclarationDate":

            "DeclarationEcoReason":

"TransactionDeclarationdLZB":
"TransactionDeclarationBranchIdLZB":
"TransactionDeclarationPayorIdLZB":
"TransactionDeclarationBuyText":
"TransactionDeclarationBuyCode":
"TransactionDeclarationCountryCode":
"TransactionDeclarationBuyAmount":
"TransactionDeclarationSellText":
"TransactionDeclarationSellCode":
"TransactionDeclarationSellCountryCode":
"TransactionDeclarationSellAmount":
"TransactionDeclarationStockingPlace":
"TransactionDeclarationSellPlace":
"TransactionDeclarationExpirationDateUnsoldGoods":
"TransactionDeclarationActionType":
"TransactionDeclarationActionCode":
```

```
"TransactionDeclarationCapitalOriginCountry":
"TransactionDeclarationCapitalDestinationCountry":
"TransactionDeclarationProductCode":
"TransactionDeclarationDNBTradingTransitCode":
"TransactionDeclarationDNBRecordNumberCode":
"TransactionDeclarationDNBGoodsTransitCode":
"TransactionDecDirectIndicator":
"TransactionSource":

"TransactionPrivatePaymentIndicator":
"TransactionFinality":

"TransactionAmountExpression":
"TransactionBeneficaryReference":
"TransactionGoodsPurchaseIndicator":
"TransactionNIFIssuer":

"TransactionNOF":

"TransactionISINCode":

"TransactionSuffix":

"TransactionInvoiceDetail":
"TransactionAdviceReference":
"TransactionAdviceMode":
"TransactionAdviceNumber":
"TransactionAdviceName":
"TransactionAdviceAdress":
"TransactionAdviceCountry":
"TransactionDirectPaymentOrderbyCCCindicator":
"TransactionDocReference":
"TransactionBonificationCurrency":
"TransactionPostalAcctBankPayee":
"TransactionPaymentNumber":
"TransactionSubmissionCode":
"TransactionCommercialFunction":
"TransactionLetter":

"TransactionMobilisation":
"TransactionRate":

"TransactionMargin":

"TransactionInvoices: [

      {

            "InvoiceEdifactType":
```

```
"InvoiceLabel":

"InvoiceInternalReference":
"InvoiceIssuingDate":

"InvoicePaidAmount":

"InvoiceBeneficiaryReference":
"InvoiceSign":

"InvoiceXMLType":

"InvoicePayeeType":

"InvoiceAdditionnalInformation":
"InvoiceDueAmount":

"InvoiceGapReason":

"InvoiceDecision":

"InvoiceCommunication":

"InvoiceCurrency":
```

Table de transcodage DTSPAYCAT_TRANS_PAY

Cette table est très importante car elle permet de déduire des données du flux la catégorie de transaction à appliquer à chaque transaction (et indirectement le format du fichier banque qui contiendra les transactions).

Les données analysées pour attribuer la catégorie de transaction sont :

- Le type de transaction :
  - **FT** pour les virements de trésorerie (**TREA** dans le flux **JSON**)
  - **DO/IN** pour les virements fournisseurs (**SUPP** dans le flux **JSON**)
- Le caractère domestique ou international : **DO/IN**
- La banque émettrice (BIC et nom)
- La banque débitée (Pays, nom, BIC)
- Le pays de la banque créditée
- La devise de transfert
- Le caractère urgent de la transaction

Ces notions constituent les colonnes d'entrée de la table de transcodage. L'astérisque * signifie *Tous*. Le mot **SEPA** fait référence aux pays de la zone SEPA. Le mot **NSEPA** fait référence aux pays de la zone Non-SEPA.

TYPE	LOCALISATION	ISSUING BANK BIC	ISSUING BANK NAME	PAYOR BANK CONTRY CODE	PAYOR BANK NAME	PAYOR BANK BIC	PAYEE BANK COUNTRY CODE	TRANSFER CURRENCY	URGENT FLAG	TRANSACTION CATEGORY
FT	DO	*	*	SEPA	*	*	SEPA	EUR	0	[SCT VTSDOM 001.001.03][902100]
FT	DO	*	*	NSEPA	*	*	NSEPA	*	0	[VTSDO Pain 001.001.03][758796]
FT	IN	*	*	SEPA	*	*	SEPA	EUR	0	[SCT VTSIN 001.001.03][903100]
FT	IN	*	*	SEPA	*	*	NSEPA	*	0	[VTSIN Pain 001.001.03][739797]
FT	IN	*	*	NSEPA	*	*	NSEPA	*	0	[VTSIN Pain 001.001.03][739797]
FT	IN	*	*	NSEPA	*	*	SEPA	*	0	[VTSIN Pain 001.001.03][739797]

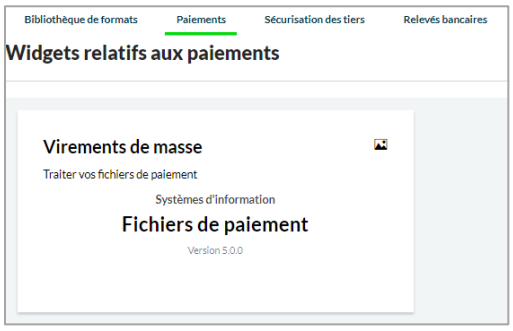
FT	DO	*	*	SEPA	*	*	SEPA	EUR	1	[SCT VTSDOM 001.001.03][902100]
FT	DO	*	*	NSEPA	*	*	NSEPA	*	1	[VTSDO Pain 001.001.03][758796]
FT	IN	*	*	SEPA	*	*	SEPA	EUR	1	[SCT VTSIN 001.001.03][903100]
FT	IN	*	*	SEPA	*	*	NSEPA	*	1	[VTSIN Pain 001.001.03][739797]
FT	IN	*	*	NSEPA	*	*	NSEPA	*	1	[VTSIN Pain 001.001.03][739797]
FT	IN	*	*	NSEPA	*	*	SEPA	*	1	[VTSIN Pain 001.001.03][739797]

La colonne de sortie **TRANSACTION CATEGORY** doit respecter la syntaxe suivante :

- nom du profil entre crochets […]
- couple format-régime entre crochets […]

Postage du flux JSON de transactions

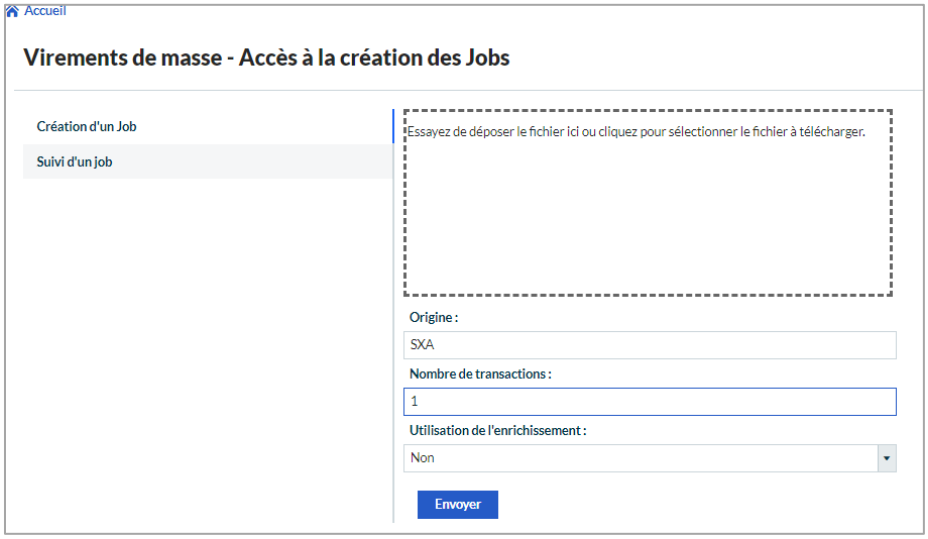
A partir de l'interface de **Sage XRT Functional Service**, sélectionnez le menu **Paiements** et ensuite le widget **Virements de masse**.



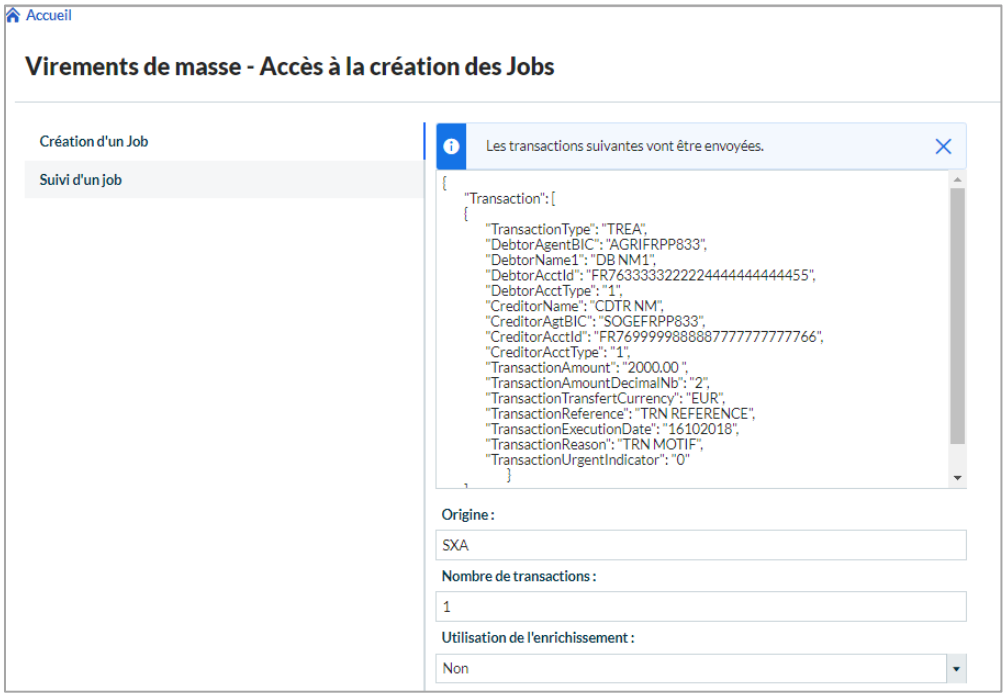
La fonction **Création d'un job** permet de sélectionner le fichier contenant les transactions à traiter (flux JSON) et d'indiquer les informations nécessaires au traitement de ce fichier :

- **Origine** des informations : **SXA**, **SXBE**, **X3**
- **Nombre de transactions** contenues dans le fichier à des fins de contrôle de sécurité
- Utilisation de l'**Enrichissement** des données initiales des transactions contenues dans le flux JSON de manière à générer des fichiers bancaires corrects non rejetés (exemple d'enrichissement : imputation des frais, finalité, etc.)

Une fois ces informations renseignées, cliquez sur le bouton **Envoyer**.



Un **ID Job** est ensuite attribué automatiquement.



Ce traitement comporte plusieurs étapes :

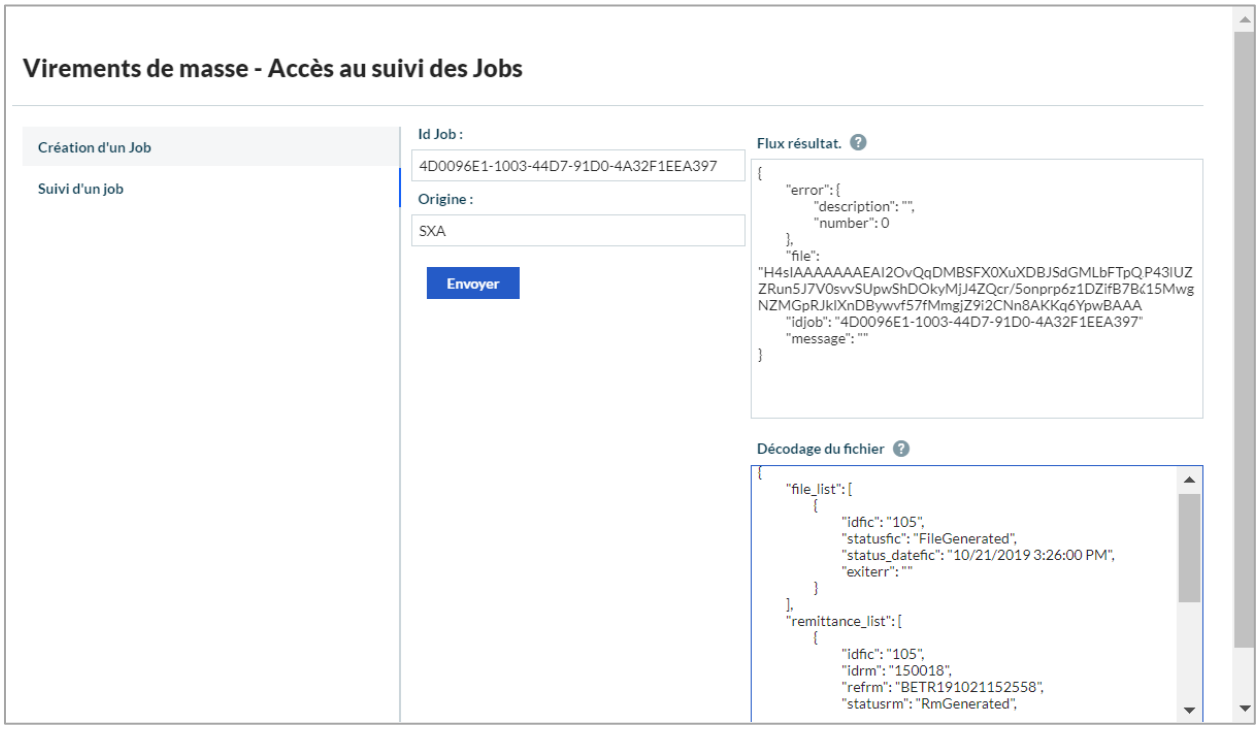
- La conversion du flux *JSON* initial en fichier *XML* pour traitement

- Le *splittage* du fichier *XML* converti en plusieurs fichiers *XML* en fonction de la catégorie de transaction (autres critères de *splittage* non activés)
- L'enrichissement des données initiales si l'option a été activée
- La dématérialisation des informations dans la base de données de **Sage XRT Common Services**

Suivi d'un JOB de traitement de flux JSON

La fonction **Suivi d'un job** permet de connaître l'avancement du traitement d'un flux *JSON* posté préalablement.

Renseignez l'**ID Job** recherché et précisez l'**Origine** du flux. Cliquez ensuite sur le bouton **Envoyer**.



La réponse peut être :

- Une erreur
- Un message **In Progress** si la requête est faite avant que la phase de stockage ne soit terminée
- Un fichier *JSON* qui se présente comme suit :


```
{
   "file_list":[
      {
         "idfic":"108",
         "statusfic":"50007",
         "status_datefic":"11/12/2018 10:06:16 AM"
      }
   ],
   "remittance_list":[
      {
         "idfic":"108",
         "idrm":"401",
         "refrm":"BETR181112100613",
         "statusrm":"50006",
         "status_daterm":"11/12/2018 10:06:16 AM"
      }
   ],
   "trn_list":[
      {
         "idfic":"108",
         "idrm":"401",
         "idtrn":"381",
         "reftrn":"EQUILIBRAGE01/06/18",
         "statustrn":"50005",
         "status_datetrn":"11/12/2018 10:06:16 AM"
      }
   ]
}
```

Statuts

!:	Transaction	Remise	Fichier	Job
Erreur totale				x
In progress				x
Erreur contrôle (à régénérer)	X (50 001 - TrnControlError)			
En attente	X (50 002 - TrnPending)	X (50 003 - RmPending		
Remisée	X (50 004 - TrnRemitted)			
Générée	X (50 005 - TrnGenerated)	X (50 006 - RmGenerated)	X (50 007 - FileGenerated)	
Statuts post-génération			X (codes statut de la signature et de la com = WMFILE)	
			50 008 – ExitPostGenerationSuccess ExitPostGenerationSuccess – No Action 50 009 - ExitPostGenerationError	

Dématérialisation dans la base de données SCS

Table **DTSPAYJOB** : contient les **ID JOBs** avec le fichier gzippé base 64 encodé avec la date de création.

Les statuts successifs (colonne **JOBSTATUS**) sont :

- 0 : en attente
- 1 : en cours
- 3 : terminé avec erreur
- 11 : terminé avec succès
- 13 : terminé : erreur technique

DTSPAYORIFILE : contient pour chaque ID JOB, les ID files pour les fichiers XML générés.

DTSPAYBATCH : contient pour chaque ID file, l'ID Batch associé.

DTSPAYPARTIES : contient les informations des entités de type banque, donneur d'ordre, payeur, bénéficiaire, payé des transactions.

DTSPAYAGENCE : contient les informations des agences débitées et créditées.

DTSPAYCPTE : contient les informations des comptes débités et crédités.

DTSPAYVIR_TMP : contient pour chaque ID Batch, la liste des transactions associées et toutes leurs informations (aucun contrôle sur les données).

DTSPAYVIR : contient pour chaque ID Batch, la liste des transactions associées et toutes leurs informations contrôlées par rapport au profil rattaché.

Les contrôles liés au profil reposent sur le fichier *dtspay_profilspay.xml* sous **C:\Program Files\Common Files\xrt** qui reprend le mécanisme des variables de **SXBE** pour déterminer les contrôles à effectuer sur la transaction.

Les statuts successifs (colonne **STATUS**) sont :

- 1 : en attente
- 6 : groupé
- 8 : généré

DTSPAYIMPXML : contient les erreurs liées à l'application des contrôles et au profil associé (colonne **VIRTMPERROR**). La colonne **VIRTMPERROR** reprend un code erreur expliqué dans le fichier *imptext.xml* (racine d'installation **SXBE**).

DTSPAYREMISE : contient la liste des remises créées à partir des transactions d'un ID Batch.

Les critères de regroupement liés au profil reposent sur le fichier *dtspay_profilspay.xml* sous **C:\Program Files\Common Files\xrt** qui reprend le mécanisme des variables *xxsame…* de **SXBE** pour déterminer les critères à utiliser.

Les statuts successifs (colonne **STATUS**) sont :

- 2 : en attente
- 5 : groupé
- 7 : généré

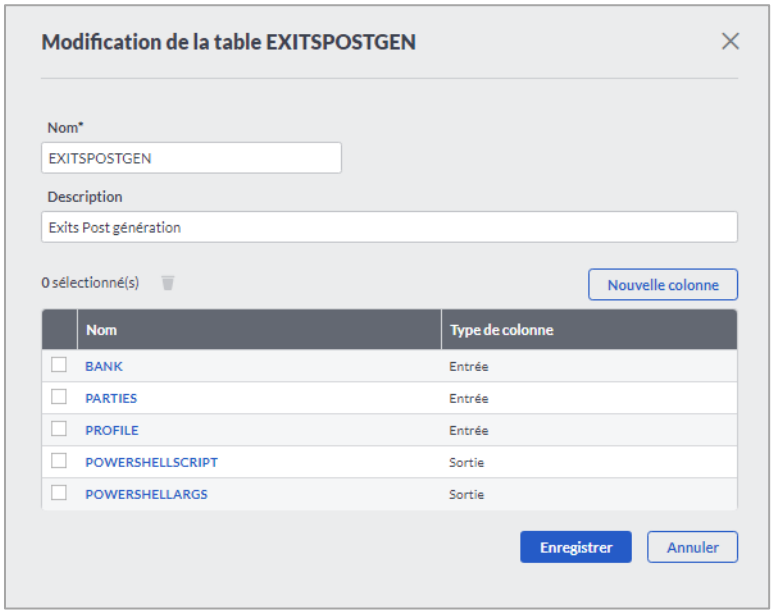
DTSPAYGENFILE : contient la liste des fichiers bancaires créés à partir des transactions d'un ID Batch.

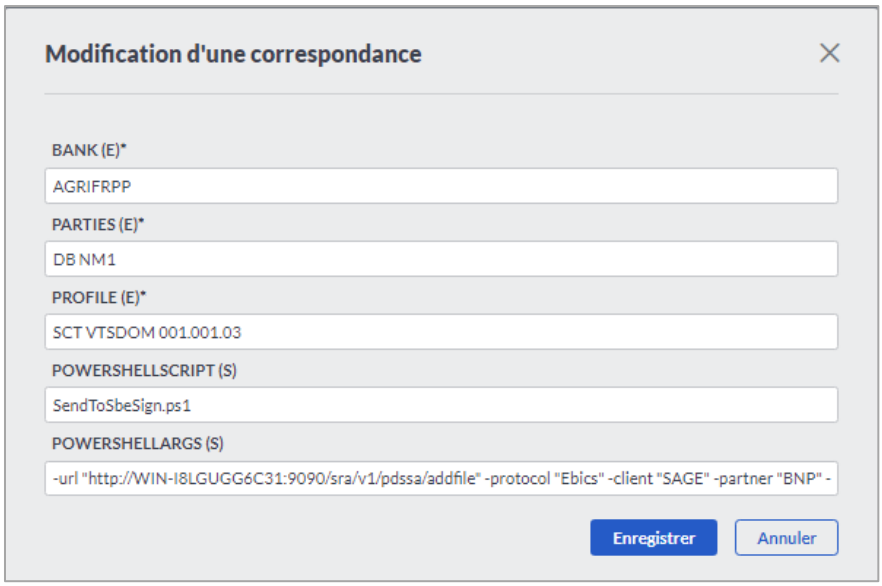
Exits post-génération

Une fois le fichier bancaire généré, il est possible de lancer un exit sur ce fichier (ex. : pour l'ajouter en signature).

La table **EXITSPOSTGENERATION** est livrée par défaut :

- Les colonnes d'entrée sont alimentées automatiquement avec les données des transactions à traiter.
- Les colonnes de sortie contiennent des scripts **POWERSHELL** (stocké **C:\Program Files\Common Files\xrt\Tasks**) et leurs arguments.





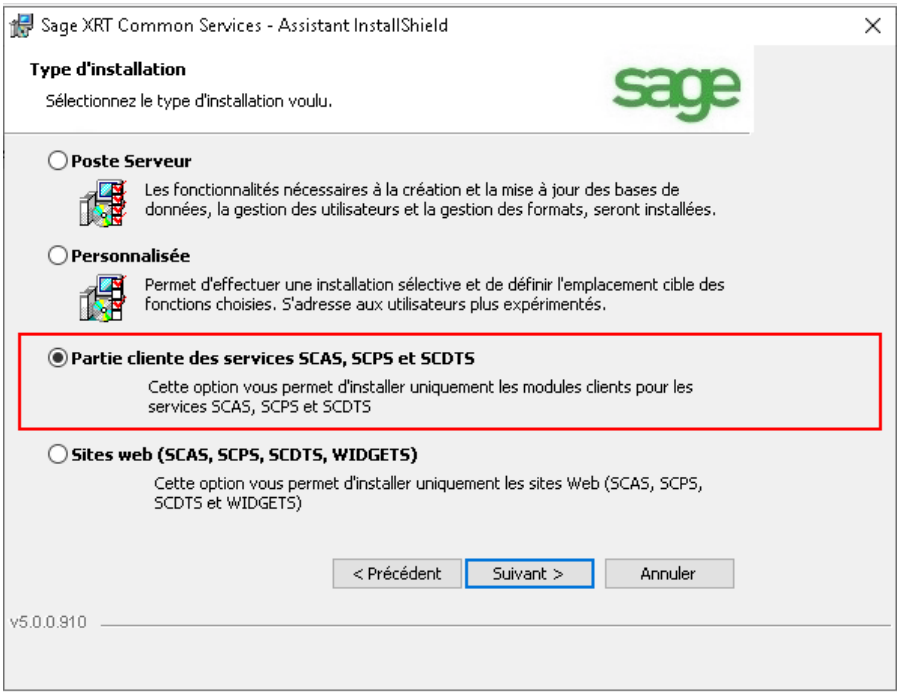
Exemple d'application : Ajout et préparation du fichier dans la signature de **Sage XRT Business Exchange**

Dans la table de transcodage, indiquez dans la colonne **Sortie** le *powershell SendToSbeSign.ps1*.

Indiquez le quadruplet à utiliser [*protocole – client – partenaire – service*]

Librairie Sage.FCS.Client

Cette librairie est une *DLL* compatible en 32 bits et 64 bits. Elle peut être utilisée soit comme une librairie *.NET* soit comme un composant *COM*. Cette librairie s'installe sur le poste client à travers le profil **Partie cliente des services SCAS, SCPS et SCDTS** de la procédure d'installation de **Sage XRT Common Services**.



Pour une intégration en tant que composant *COM* dans une application 32 bits il faut enregistrer la *DLL* avec le *RegAsm.exe* du framework *.NET* 32bits.

Utilisation

Sage.Fcs.Client permet de consommer de façon transparente un ensemble d'API Rest hébergé par les services d'Authentification (**SCASServer**), d'Administration (**SCPSServer**) et de Fonctionnalités (**SCDTSServer**) de **Sage XRT Common Services**. Ainsi les applications intégrant cette librairie feront juste des appels à des méthodes. La localisation du serveur **Sage XRT Common Services** se fait grâce au paramétrage du fichier de configuration *Sage.Fcs.Client.dll.config*.

Exemple de fichier de configuration :

```
<?xml version="1.0" encoding="utf-8" ?>

<configuration>

  <!-- The auto generate certificate CN is Sage.fcs.client -->

  <appSettings>

    <add key="securitylevel" value="http"/>

    <add key="http_servicehost_SCASServer"
value="http://localhost:80/Auth" />

    <add key="https_servicehost_SCASServer"
value="https://localhost:443/Auth" />

    <add key="http_servicehost_SCDTSServer" value="http://localhost:80" />

    <add key="https_servicehost_SCDTSServer" value="https://localhost:443"
/>

    <add key="http_servicehost_SCPSServer" value="http://localhost:80"/>

    <add key="https_servicehost_SCPSServer"
value="https://localhost:443"/>

    <add key="AutoGenerateCertificate" value="true"/>

    <add key="SerialNumber" value=""/>

    <add key="ApifmtResponseTimeout" value="180"/>

    <add key="ApifmtBaseTimeTry" value="2"/>

    <add key="ApifmtPdfViewer" value="none"/>

    <add key="ApifmtPrinter" value="none"/>

  </appSettings>

</configuration>
```

Description des méthodes intégrées

Connexion

void DoLogin(object p_oReq)

Cette méthode permet l'authentification et la génération d'un token d'accès au service. Ce token est utilisé de manière transparente lors des appels aux différentes méthodes.

Requête :

```
public class LoginReq : ILoginReq

{

    public LoginMode mode { get; set; }

    public string workgroup { get; set; }

    public string user { get; set; }

    public string password { get; set; }

    public string certificatebase64 { get; set; }

    public string challengerawbase64 { get; set; }

    public string challengesignedhex { get; set; }

}

public enum LoginMode

{

    SAMLV2_AUTH,

    STRONG_AUTH,

    STANDARD_AUTH

}
```

Réponse :

```
 public class LoginRep : ILoginRep
{
    public string authorization { get; set; }
    public string description { get; set; }
    public string twofactorauth { get; set; }
    public string fcsuser { get; set; }
    public string idpurl { get; set; }
}
```

void CheckSession()

Cette méthode permet de vérifier le *token* utilisé.

Réponse :

```
public class VerifCodeRep : IVerifCodeRep
{
    public string valid { get; set; }
}
```

void CheckUser(object p_oReq)

Cette méthode permet de vérifier l'accès d'un utilisateur.

Requête :

```
public class CheckUserReq : ICheckUserReq
{
    public string user { get; set; }
    public string password { get; set; }
    public bool ctrlpwd { get; set; }
}
```

void Disconnect()

Cette méthode permet de déconnecter un utilisateur.

object ConnectionString(object p_oReq)

Cette fonction permet de récupérer une chaîne de connexion au niveau DBA, DBO ou USER. Pour fonctionner cette méthode nécessite que la clé **SerialNumber** du fichier de configuration soit renseignée ou que la valeur **AutoGenerateCertificate** soit **true**. En effet pour des raisons de sécurité la chaîne de connexion est cryptée (RSA).

Requête :

```
public class ConnectionStringReqPub : IConnectionStringReqPub
{
    public DatabaseProductID product { get; set; }
    public DatabaseOwner type { get; set; }
}
public enum DatabaseOwner
{
    DBA,
    DBO,
    USER
}
```

Réponse :

```
public class ConnectionStringRep : IConnectionStringRep
{
    public string connectionstring { get; set; }
}
```

void RefreshToken()

Page de connexion

void DoLoginForm(LoginProductID product, string version, string component, LanguageID language)

Cette méthode permet l'authentification et la génération d'un token d'accès au service dans une interface *Win32*. Ce token est utilisé de manière transparente lors des appels aux différentes méthodes. La méthode prend en charge la vérification des licences.

Requête :

Product : identifiant du produit.

```
public enum LoginProductID
    {
        XCS,
        XBE,
        XBESign,
        U2,
        U2Sign,
        U2Com
    }
```

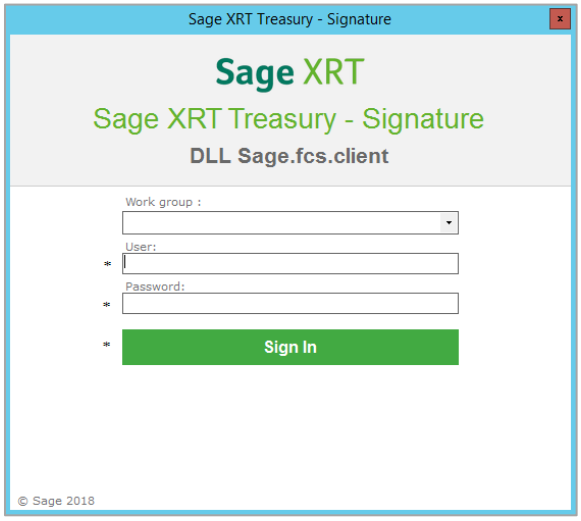
Version : version du produit. Ex. : 2019

Component : nom du composant sur lequel on se connecte. Ex. : DLL *Sage.fcs.client*

Language : langue de l'interface

```
public enum LanguageID
{
    French,
    English,
    Spanish,
    German,
```

```
        Italian,
        Portuguese,
        Polish,
        Catalan
    }
```



void DoLoginService(object p_oReq)

Cette méthode permet l'authentification et la génération d'un token d'accès au service.

Requête :

```
public class DoLoginServiceReq : IDoLoginServiceReq
{
    public ServiceName servicename { get; set; }
    public string workgroup { get; set; }
}
public enum ServiceName
{
    XRTCOM,         // Service XCMonitor
    XRTSIGN,        // Service XSMonitor
    SRA,
```

```
        RAPI,
        P5CWEB,
        P5COM,
        P5CAUT,
        XSMONITOR,
        BCPCOMM,          // BcpComm
        COMSIGNAPI,       // ComSign. Rest Api
        SXA
    }
```

Tenants

object WorkgroupList()

Cette méthode peut être appelée sans authentification et sans token d'accès. Elle renvoie un tableau représentant la liste des tenants disponibles.

Requête :

```
    public class WorkgroupListRep : IWorkgroupListRep
    {
        public string selectedworkgroup { get; set; }
        public object workgroups { get; set; }
    }
```

Licences

object LicenseDetails(object p_oReq)

Cette fonction permet de récupérer des informations détaillées sur la licence d'un produit donné.

Requête :

```
    public class LicenseDetailsReq : ILicenseDetailsReq
    {
        public string product { get; set; }
```

```
}
```

Réponse :

```
public class LicenseDetailsRep : ILicenseDetailsRep
{
    public object features { get; set; }
}
public class Feature : IFeature
{
    public string id { get; set; }
    public string description { get; set; }
    public string value { get; set; }
}
```

object LicenseList()

Cette fonction permet de récupérer la liste des licences disponibles.

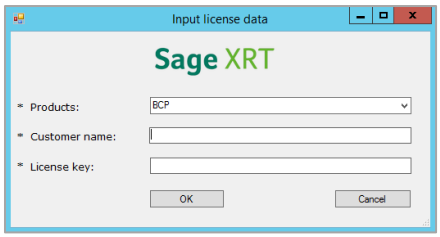
Réponse :

```
public class LicenseListRep : ILicenseListRep
{
    public object licenses { get; set; }
    public object products { get; set; }
}
public class License : ILicense
{
    public string product { get; set; }
    public string customer { get; set; }
    public string enddate { get; set; }
    public string key { get; set; }
```

```
}
public class Product : IProduct
{
    public string product { get; set; }
    public string name { get; set; }
}
```

void LicenseCreateForm()

Cette fonction permet de créer une licence à travers une interface graphique *Win32*.



void LicenseCreate()

Cette fonction permet de créer une licence.

Requête :

```
public class LicenseCreateReq : ILicenseCreateReq
{
    public string product { get; set; }
    public string customer { get; set; }
    public string key { get; set; }
}
```

Réponse :

```
public class LicenseCreateRep : ILicenseCreateRep
{
    public string enddate { get; set; }
}
```

void LicenseMigrate(object p_oReq)

Cette fonction permet de mettre à jour une licence. Elle est utilisée par exemple par la version de **Sage XRT Communication & Signature** qui fonctionne avec **Sage XRT Common Services** 5.0.0 pour migrer les licences vers le nouveau système de gestion des licences.

Requête :

```
public class LisenceMigrateReq : ILisenceMigrateReq
{
    public LicenseProductID product { get; set; }
    public string customername { get; set; }
    public string shortserialnumber { get; set; }
}
public enum LicenseProductID
{
    FORMATS,
    U2,
    PDS,
    BCP,
    CONVERTERS,
    SXA
};
```

Profils -Sites -Utilisateurs- Paramétrage

object FunctionRights(object p_oReq)

Cette fonction permet de récupérer les droits associés à une liste de fonctions, ceci au niveau de l'utilisateur connecté, d'un utilisateur passé en paramètre ou d'un profil passé en paramètre.

Requête :

```
public class FunctionRightsReq : IFunctionRightsReq
{
```

```
        public FunctionRightsLevel level { get; set; } =
FunctionRightsLevel.CURRENTUSER;

        public string code { get; set; }

        public string[] functions { get; set; }

    public enum FunctionRightsLevel

    {

        CURRENTUSER,

        USER,

        PROFILE

    }
```

Réponse :

```
    public class FunctionRightsRep : IFunctionRightsRep

    {

        public object functions { get; set; }

    }

    public class FuncRight: IFuncRight

    {

        public string function { get; set; }

        public bool right { get; set; }

    }
```

object ProfileList()

Cette fonction permet de récupérer la liste des profils utilisateurs associés au tenant du token.

Requête :

```
     public class ProfileListRep : IProfileListRep

    {

        public object profiles { get; set; }

    }
```

```
public class Profile : IProfile
{
    public string name { get; set; }
    public string description { get; set; }
    public int type { get; set; }
    public bool active { get; set; }
    public string lastactionuser { get; set; }
```

object UserList(object p_oReq)

Cette fonction permet de récupérer la liste des utilisateurs associée au tenant du token.

Requête :

```
 public class UserListReq : IUserListReq
{
    public bool bGetProfileList { get; set; } = false;
    public bool bGetSiteList { get; set; } = false;

}
```

Réponse :

```
public class UserListRep : IUserListRep
{
    public object users { get; set; }
    public object profiles { get; set; }
    public object sites { get; set; }
}
public class User : IUser
{
    public string name { get; set; }
```

```
        public string description { get; set; }
        public int securitylevel { get; set; }
        public string lang { get; set; }
        public string email { get; set; }
        public int authenticationtype { get; set; }
        public bool authenticatex509 { get; set; }
        public string validtilldate { get; set; }
        public bool active { get; set; }
        public string lastactionuser { get; set; }
        public bool isvirtual { get; set; }
        public bool issecurityleveldefined { get; set; }
        public bool islanguagedefined { get; set; }
        public bool twofactorauthlevel { get; set; }
        public string tfasecretcodevalid { get; set; }
        public string lockedtilldate { get; set; }
        public string lockedtilltime { get; set; }
        public object profiles { get; set; }
        public object sites { get; set; }
    }
    public class Profile : IProfile
    {
        public string name { get; set; }
        public string description { get; set; }
        public int type { get; set; }
        public bool active { get; set; }
        public string lastactionuser { get; set; }
```

```
public class Site : ISite
{
    public string name { get; set; }
    public string description { get; set; }
    public int type { get; set; }
    public bool active { get; set; }
    public string lastactionuser { get; set; }
}
```

object UserInfos(object p_oReq)

Cette fonction permet de récupérer les informations d'un utilisateur.

Requête :

```
public class DataUserInfosReq : IDataUserInfosReq
{
    public string username { get; set; }
    public bool bGetRules { get; set; } = false;
    public bool bGetProfilsAndSites { get; set; } = false;
}
```

Réponse :

```
public class UserInfosRep: IUserInfosRep
{
    public object user { get; set; }
    public object rules { get; set; }
}
public class User : IUser
{
    public string name { get; set; }
```

```
        public string description { get; set; }
        public int securitylevel { get; set; }
        public string lang { get; set; }
        public string email { get; set; }
        public int authenticationtype { get; set; }
        public bool authenticatex509 { get; set; }
        public string validtilldate { get; set; }
        public bool active { get; set; }
        public string lastactionuser { get; set; }
        public bool isvirtual { get; set; }
        public bool issecurityleveldefined { get; set; }
        public bool islanguagedefined { get; set; }
        public bool twofactorauthlevel { get; set; }
        public string tfasecretcodevalid { get; set; }
        public string lockedtilldate { get; set; }
        public string lockedtilltime { get; set; }
        public object profiles { get; set; }
        public object sites { get; set; }
    }
    public class UserInfosRules: IUserInfosRules
    {
        public int iAccountsMinDigitsInPassword { get; set; }
        public int iAccountsMinCapLettersInPassword { get; set; }
        public int iAccountsMinPasswordLength { get; set; }
        public int iAccountsCheckPasswordsHistoryLength { get; set; }
        public bool bAccountsCheckPasswordsDifferFromUserName { get; set; }
```

```
        public string sAccountsDefaultPassword { get; set; }
        public bool bUseCustomLDAPServer { get; set; }
        public bool bSageIDEnabled { get; set; }
        public bool bSAMLEnabled { get; set; }
        public bool bSAMLUseMail { get; set; }
        public bool bActivateUsers { get; set; }
        public bool bActivateSites { get; set; }
        public bool bActivateProfiles { get; set; }
        public bool bActivateBlacklist { get; set; }
        public bool bActivateWhitelist { get; set; }
        public bool bFourEyesUsers { get; set; }
        public bool bFourEyesSites { get; set; }
        public bool bFourEyesProfiles { get; set; }
        public bool bFourEyesParams { get; set; }
        public bool bFourEyesConception { get; set; }
        public bool bFourEyesCorrespondances { get; set; }
        public bool bFourEyesBlacklist { get; set; }
        public bool bFourEyesWhitelist { get; set; }
    }
```

Mot de passe

void Password ()

Cette méthode permet de mettre à jour le mot de passe d'un utilisateur.

Requête :

```
    public class PasswordReq : IPasswordReq
    {
```

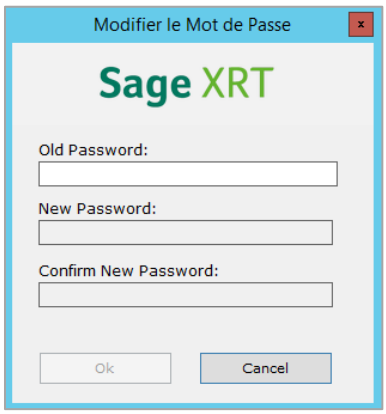
```
        public string workgroup { get; set; }
        public string user { get; set; }
        public string old_password { get; set; }
        public string new_password { get; set; }
    }
```

Réponse :

```
    public class PasswordRep : IPasswordRep
    {
        public string authorization { get; set; }
        public string description { get; set; }
        public string twofactorauth { get; set; }
    }
```

void PasswordUpdatingByActionForm()

Cette méthode permet de mettre à jour le mot de passe de l'utilisateur connecté via une interface *Win32*.



void PasswordUpdatingByAction(object p_oReq)

Cette méthode permet de mettre à jour le mot de passe de l'utilisateur connecté.

Requête :

```
    public class PasswordReqByAction : IPasswordReqByAction
    {
```

```
        public string old_password { get; set; }
        public string new_password { get; set; }
    }
```

object PwdInfos();

Cette fonction permet de récupérer les règles sur les mots de passe des utilisateurs standards de **Sage XRT Common Services**.

Réponse :

```
    public class PwdInfosRep : IPwdInfosRep
    {
        public object pwd { get; set; }
    }
    public class DataParamsUpdatePassword : IDataParamsUpdatePassword
    {
        public int bTrustedConnection { get; set; } = -1;
        public int iAccountsMinDigitsInPassword { get; set; } = -1;
        public int iAccountsMinCapLettersInPassword { get; set; } = -1;
        public int iAccountsMinPasswordLength { get; set; } = -1;
        public int iAccountsCheckPasswordsHistoryLength { get; set; } = -1;
        public int bAccountsCheckPasswordsDifferFromUserName { get; set; }
= -1;
        public string sAccountsDefaultPassword { get; set; }
        public int iAccountsDefaultPasswordValidityPeriodType { get; set; }
= -1;
        public int iAccountsDefaultPasswordValidityPeriodCount { get; set;
} = -1;
        public int iAccountsPasswordValidityPeriodType { get; set; } = -1;
        public int iAccountsPasswordValidityPeriodCount { get; set; } = -1;
        public int iAccountsLockoutAfterAttemtsCount { get; set; } = -1;
```

```
        public int iAccountsLockoutPeriodType { get; set; } = -1;

        public int iAccountsLockoutPeriodCount { get; set; } = -1;

    }
```

Bibliothèque des formats

object ProjectList(object p_oReq)

Cette fonction permet de récupérer la liste des projets de la bibliothèque des formats associée à un produit.

Requête :

```
    public class ProjectListReq : IProjectListReq

    {

        public string product { get; set; }

    }
```

Réponse :

```
     public class ProjectListRep : IProjectListRep

    {

        public object projects { get; set; }

    }

    public class ProjectList : IProjectList

    {

        public string code { get; set; }

        public string name { get; set; }

        public object types { get; set; }

        public int licence { get; set; }

        public ApifmtFamily family { get; set; }

        public ApifmtFormat format { get; set; }

        public int nbrwork { get; set; }
```

```
        public int productid { get; set; }
    }
```

object WorkList(object p_oReq)

Cette fonction permet de récupérer la liste des traitements de la bibliothèque des formats associée à un produit et à un projet.

Requête :

```
     public class WorkListReq : IWorkListReq
    {
        public string product { get; set; }
        public string project { get; set; }
    }
```

Réponse :

```
    public class WorkListRep : IWorkListRep
    {
        public object works { get; set; }
    }
    public class WorkList : IWorkList
    {
        public string code { get; set; }
        public string name { get; set; }
        public string version { get; set; }
        public object types { get; set; }
        public object tasks { get; set; }
    }
    public class ApifmtWorkType : IApifmtWorkType
    {
```

```
        public int code { get; set; }
        public string desc { get; set; }
    }
     public class Task : ITask
    {
        public string param { get; set; }
    }
```

object TranscoTableList()

Cette fonction permet de récupérer la liste des tables de transcodage associée au tenant du token.

Réponse :

```
     public class TranscoTableListRep : ITranscoTableListRep
    {
        public object tables { get; set; }
    }
    public class Table : ITable
    {
        public int id { get; set; }
        public string name { get; set; }
        public string description { get; set; }
        public int input { get; set; }
        public int output { get; set; }
        public string lastactionuser { get; set; }
    }
```

object FormatList()

Cette fonction permet de récupérer la liste des formats de la bibliothèque des formats.

Réponse :

```
 public class ApifmtFormatListRep : IApifmtFormatListRep
{
    public object formats { get; set; }
}
public class ApifmtFormat : IApifmtFormat
{
    public int id { get; set; }
    public string name { get; set; }
    public string desc { get; set; }
    public int licence { get; set; }
}
```

object FamilyList()

Cette fonction permet de récupérer la liste des familles de la bibliothèque des formats.

Réponse :

```
 public class ApifmtFamilyListRep : IApifmtFamilyListRep
{
    public object families { get; set; }
}
public class ApifmtFamily : IApifmtFamily
{
    public int id { get; set; }
    public string name { get; set; }
    public string desc { get; set; }
    public FamilyType type { get; set; }
}
```

object ProductList();

Cette fonction permet de récupérer la liste des produits de la bibliothèque des formats.

Réponse :

```
 public class ApifmtProductListRep : IApifmtProductListRep
{
    public object products { get; set; }
}
public class ProductList : IProductList
{
    public int id { get; set; }
    public string code { get; set; }
    public string name { get; set; }
    public int nbrproject { get; set; }
```

object WorkExec(object p_oReq);

Cette fonction permet d'exécuter un traitement de l'**apifmt** en passant dans la variable **parameters** la chaîne de caractère telle qu'elle était attendue dans l'ancienne librairie.

Exemples :

1)

```
Wscript.echo "-- WorkExec BCP_SOLDE"
    l_oWorkExecReq.product = 0 '"BCP"
    l_oWorkExecReq.project = "E_REL120"
    l_oWorkExecReq.work = "BCP_SOLDE"
    l_oWorkExecReq.parameters =
"""C:\DATA\SBE\Payment_Fichier_import_valides_1050\AFB120\Fichier 1
AFB120.TXT""
""C:\DATA\SBE\Payment_Fichier_import_valides_1050\AFB120\BCP_SOLDE.out"""
```

2)

```
Wscript.echo "-- WorkExec Prepa"

      l_oWorkExecReq.product = 2 '"PDS"

      l_oWorkExecReq.project = "SCT"

      l_oWorkExecReq.work = "Prepa"

      l_oWorkExecReq.parameters =
"""C:\Users\Marc\AppData\Local\Temp\XRT\PDS\CXF{378DEFE3-AE40-417E-8D57-
3D514E4FE6B3}.tmp""
""C:\Users\Marc\AppData\Local\Temp\XRT\PDS\PXF_c0b93023-fbea-4cbf-a0f5-
7c91884d6d53.TMP"" N  ; PASDEVISU LANG:0C
""Provider=OraOLEDB.Oracle.1;Password=XBEUNICODE;Persist Security
Info=True;User ID=XBEUNICODE;Data Source=ORAC12C;Extended Properties="""""
```

3)

```
Wscript.echo "-- WorkExec SQL_FORMULA_VISU (PRINTER)"

      l_oWorkExecReq.product = 2 '"PDS"

      l_oWorkExecReq.project = "EDITIONS"

      l_oWorkExecReq.work = "SQL_FORMULA_VISU"

      l_oWorkExecReq.parameters = "status.rpt ""Provider=SQLNCLI11;Data
Source=PO150073;Initial Catalog=XRTCOM;User ID=XRTDBO;Password=XRTDBO"" IL
""title='Statuts des fichiers de
signature';partner='*';protocol='*';service='*';client='*';int=' -
';ext='';cre=' - ';sta='Fichier ajouté, Fichier archivé, Fichier préparé,
Fichier signé';l0='Signature
interne';l1='Partenaire';l2='Protocole';l3='Service';l4='Client';c1='Référ
ence';c2='Référence externe';c3='Date d''ajout';c4='Montant (EUR)';c5='Nb
signatures réalisées';c6='Première signature';c7='Premier
signataire';c8='Deuxième signature';c9='Statut';edit='Edité
le';yes='Oui';no='Non';ajoute='Ajouté';prepare='Préparé';archive='Archivé'
;signe='Signé';refustotal='Refus total';rejetstotal='Rejets
total';avecrejets='avec Refus';avecrefus='avec
Rejets';verrouille='Verrouillé'"" ""{S_TFI.CNTR_TFI} in[1,2,32,16]"""
```

Requête :

```
    public class WorkExecReq : IWorkExecReq

    {

        public ApifmtProductID product { get; set; }
```

```
        public string project { get; set; }
        public string work { get; set; }
        public string parameters { get; set; }
    }
    public enum ApifmtProductID
    {
        BCP,
        DTS,
        PDS,
        SMP_P5,
        SXA,
        U2
    };
```

Réponse :

```
    public class WorkExecRep : IWorkExecRep
    {
        public string trace { get; set; }
    }
```

object GenericWorkExec(object p_oReq);

Cette fonction permet d'exécuter certains traitements de l'**apifmt** ou des traitements spécifiques qui auraient été paramétrés.

Ex. :

```
 1)l_oGenericWorkExecReq.product = 5 '"U2"
      l_oGenericWorkExecReq.project = "SCTDOM"
      l_oGenericWorkExecReq.work = "SCTVTSDOM_PAIN_03"
      l_oGenericWorkExecReq.infile =
"C:\DATA\SBE\Payment_Fichier_import_valides_1050\U2\Intermedio_traspaso_te
so_v40.TRT"
```

```
     l_oGenericWorkExecReq.outfile =
"C:\DATA\SBE\Payment_Fichier_import_valides_1050\U2\Intermedio_traspaso_te
so_v40.out"
```

Requête :

```
public class GenericWorkExecReq : IGenericWorkExecReq
{
    public ApifmtProductID product { get; set; }
    public string project { get; set; }
    public string work { get; set; }
    public string infile { get; set; }
    public string outfile { get; set; }
    public string cmdargs { get; set; }
}
public enum ApifmtProductID
{
    BCP,
    DTS,
    PDS,
    SMP_P5,
    SXA,
    U2
};
```

Réponse :

```
public class GenericWorkExecRep : IGenericWorkExecRep
{
    public string trace { get; set; }
}
```

Fichier de configuration

void SetConfigFile(object p_oReq);

Cette fonction permet de mettre à jour le fichier de configuration de la librairie.

Requête :

```
public class ConfigFileReq : IConfigFileReq
{
    public string securitylevel { get; set; }
    public string http_servicehost_SCASServer { get; set; }
    public string https_servicehost_SCASServer { get; set; }
    public string http_servicehost_SCDTSServer { get; set; }
    public string https_servicehost_SCDTSServer { get; set; }
    public string http_servicehost_SCPSServer { get; set; }
    public string https_servicehost_SCPSServer { get; set; }
    public string autogeneratecertificate { get; set; }
    public string serialnumber { get; set; }
    public string apifmtresponsetimeout { get; set; }
    public string apifmtbasetimetry { get; set; }
    public string apifmtpdfviewer { get; set; }
    public string apifmtprinter { get; set; }
}
```

Application sage.fcs.apifmt.exe

Cette application console permet d'obtenir le résultat de l'exécution distante d'un traitement de l'API des formats.

Installation

L'installation de cette application est effectuée par l'installation du produit **Sage XRT Common Services**, par la sélection du composant **FCS DLLs** ou du composant **Partie cliente des services SCAS, SCPS et SCDTS**. Hors désélection de ces deux composants lors d'une installation personnalisée, cette application est donc installée par défaut dans toutes les configurations sauf **Sites Web (SCAS, SCPS, SCDTS et WIDGETS)**.

Deux fichiers sont installés :

- *sage.fcs.apifmt.exe*
- *sage.fcs.apifmt.exe.config*

Le fichier de configuration *sage.fcs.apifmt.exe.config* associé à l'exécutable *sage.fcs.apifmt.exe* est utilisé pour la déclaration de l'emplacement du service d'authentification ainsi que celui du service de transformation des données.

Configuration

Le contenu du fichier *sage.fcs.apifmt.exe.config* est par défaut le suivant :

```
<?xml version="1.0" encoding="utf-8" ?>

<configuration>

  <appSettings>

    <add key="securitylevel" value="http"/>

    <add key="http_servicehost_SCASServer"
value="http://localhost:80/Auth" />

    <add key="https_servicehost_SCASServer"
value="https://localhost:443/Auth" />

    <add key="http_servicehost_SCDTSServer" value="http://localhost:80" />

    <add key="https_servicehost_SCDTSServer" value="https://localhost:443"
/>

  </appSettings>
```

```
</configuration>
```

Service d'authentification

Ce fichier permet d'effectuer la déclaration du service d'authentification qui est contacté pour la vérification du mot de passe de l'utilisateur.

Dans le cas où celui-ci fonctionne en mode sécurisé *https*, il faut apporter la modification à la ligne suivante :

```
<add key="securitylevel" value="https"/>
```

Pour un mode non sécurisé, la ligne suivante est exploitée :

```
<add key="http_servicehost_SCASServer" value="http://localhost:80/Auth" />
```

Pour un mode sécurisé, la ligne suivante est exploitée :

```
<add key="https_servicehost_SCASServer" value="https://localhost:443/Auth"
/>
```

Vous devez effectuer la modification de *value* pour tenir compte des spécificités de chaque installation.

Pour rappel, l'utilisation d'un certificat au niveau du service d'authentification (pour effectuer le déchiffrement du mot de passe) se fait en modifiant le fichier de configuration de ce dernier (*Sage.SCASServer.Service.exe.config*) et en renseignant le numéro de série du certificat sur la ligne suivante :

```
<add key="serialnumberforpwdcrypt"
value="3526df91b8be9ab046a226d0390a764f" />
```

Si la *value* de cette zone est laissée vide, alors le paramètre **/PWT=RSA** (cf. Utilisation) ne pourra pas être exploité avec ce service d'authentification.

Service de transformation des données

Ce fichier permet également d'effectuer la déclaration du service de transformation des données qui est contacté pour effectuer l'exécution du traitement demandé.

Dans le cas où celui-ci fonctionne en mode sécurisé *https*, il faut apporter la modification à la ligne suivante :

```
<add key="securitylevel" value="https"/>
```

Pour un mode non sécurisé, la ligne suivante est exploitée :

```
<add key="http_servicehost_SCDTSServer" value="http://localhost:80/Auth"
/>
```

Pour un mode sécurisé, la ligne suivante est exploitée :

```
<add key="https_servicehost_SCDTSServer"
value="https://localhost:443/Auth" />
```

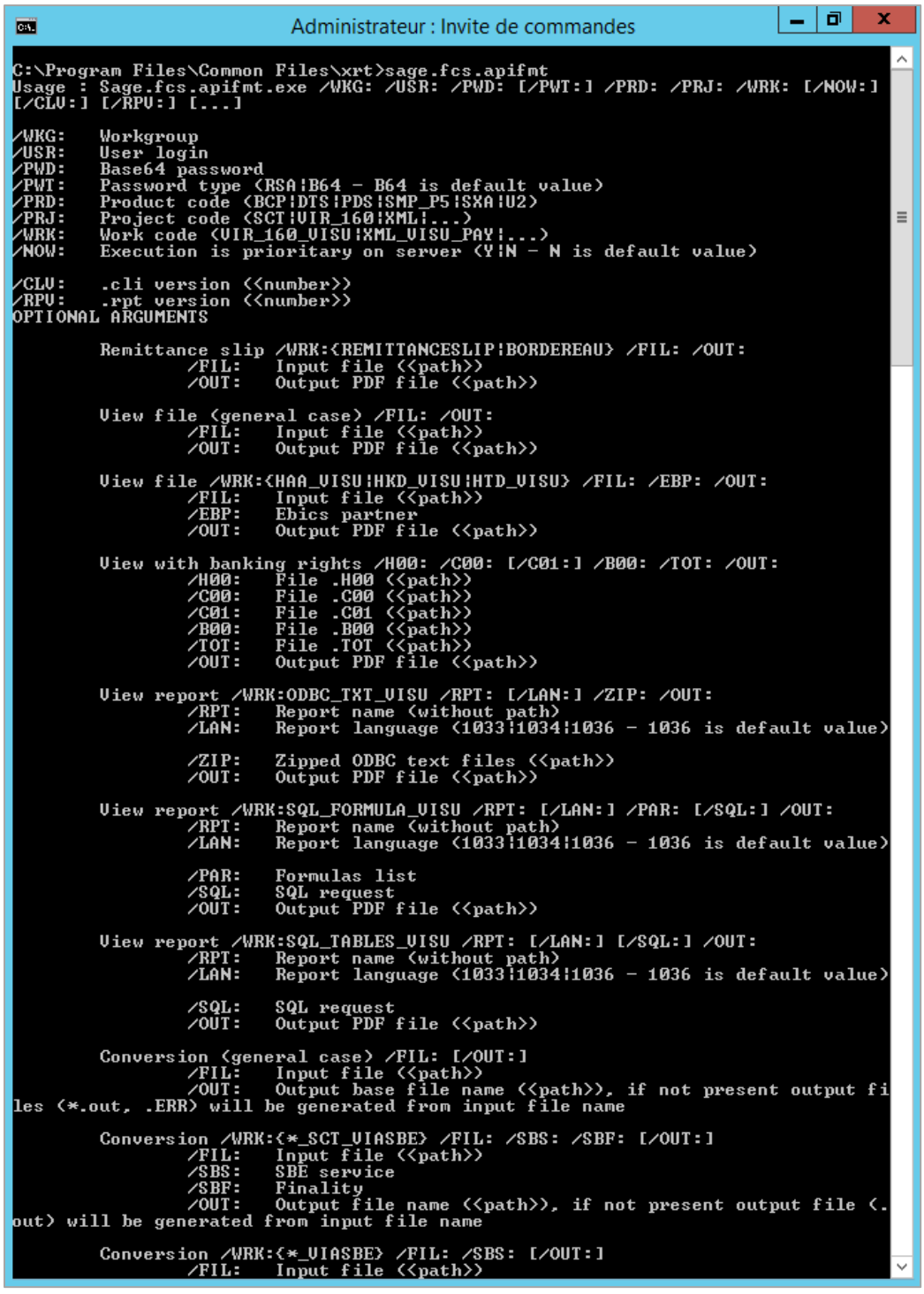
Vous devez effectuer la modification de *value* pour tenir compte des spécificités de chaque installation.

Permissions

Afin de pouvoir exécuter un traitement sans erreur, l'utilisateur passé en paramètre (**/USR**) doit appartenir à un profil ayant au minimum la permission dans **Sage XRT Functional Service**, partie **Bibliothèque de formats - Formats - Produits** :

- Visualiser les détails d'un traitement (dans tous les cas)
- Exécuter un traitement synchrone (si **/NOW :YES** est utilisé)
- Exécuter un traitement asynchrone et Récupérer le résultat d'un traitement asynchrone (si **/NOW :FALSE** est utilisé)

Application sage.fcs.apifmt.exe



Utilisation

Paramètres de base

Tenant /WKG

Ce paramètre est obligatoire puisqu'il permet d'indiquer quelle base de données est utilisée avec les services d'authentification et de transformation de données.

Utilisateur /USR

Ce paramètre est obligatoire puisqu'il permet d'indiquer quel utilisateur est utilisé pour l'authentification et les permissions associées au profil auquel celui-ci est rattaché.

Mot de passe /PWD

Ce paramètre est obligatoire puisqu'il permet d'indiquer quel est le mot de passe associé à l'utilisateur. Celui-ci ne doit pas apparaitre en clair : il doit être obtenu à l'aide de l'application console **sage.fcs.pwdencode.exe** (cf. ci-après).

Type de mot de passe /PWT

Ce paramètre est optionnel.

La valeur à positionner dépend du paramétrage de chiffrement souhaité pour le mot de passe du service d'authentification. Par défaut il a pour valeur **B64**, ce qui correspond à un encodage Base64 uniquement. Dans le cas où l'encodage par défaut ne convient pas, vous devez positionner la valeur **RSA** pour ce paramètre.

Produit /PRD

Ce paramètre est obligatoire puisqu'il correspond au code produit contenant le traitement à exécuter (ex. : **BCP** pour **Sage XRT Treasury – Communication**).

Projet /PRJ

Ce paramètre est obligatoire puisqu'il correspond au code projet contenant le traitement à exécuter (ex. : **SCT** pour Format SCT SEPA).

Traitement /WRK

Ce paramètre est obligatoire puisqu'il correspond au code du traitement à exécuter (ex. : **SCT_VISU_PAY** pour Edition du SCT SEPA).

Type d'exécution (/NOW)

Ce paramètre est optionnel.

Par défaut sa valeur est positionnée à **N**, ce qui correspond à une demande d'exécution asynchrone du traitement. L'application demande si l'exécution a été traitée par le serveur de transformation des données, l'attente étant déterminée par la taille du fichier en entrée. Par défaut, l'attente est de 30 secondes.

Un traitement déterminé comme relativement léger peut être exécuté en positionnant le paramètre à **Y**. L'exécution est alors synchrone et le résultat est obtenu aussitôt son traitement sur le serveur terminé.

Version du fichier Clint /CLV

Ce paramètre n'est pas encore exploitable. Il permet d'indiquer la version du fichier *.cli* à exécuter dans le cas où un traitement Clint spécifique a été dérivé du traitement *Clint* original.

Version du fichier Crystal Report /RPV

Ce paramètre n'est pas encore exploitable. Il permet d'indiquer la version du fichier *.rpt* à exécuter dans le cas où un rapport *Crystal Report* spécifique a été dérivé du rapport original.

Paramètres spécifiques

Plusieurs familles de traitements ont été définies par défaut et une aide sur l'utilisation de paramètres spécifiques est disponible sur la ligne de commande sous la rubrique **OPTIONAL ARGUMENTS**.

```
Remittance slip /WRK:{REMITTANCESLIP|BORDEREAU} /FIL: /OUT:

View file (general case) /FIL: /OUT:

View file (Ebics) /WRK:{HAA_VISU|HKD_VISU|HTD_VISU} /FIL: /EBP: /OUT:

View with banking rights /H00: /C00: [/C01:] /B00: /TOT: /OUT:

View report /WRK:ODBC_TXT_VISU /RPT: [/LAN:] /ZIP: /OUT:

View report /WRK:SQL_FORMULA_VISU /RPT: [/LAN:] /PAR: [/SQL:] /OUT:

View report /WRK:SQL_TABLES_VISU /RPT: [/LAN:] [/SQL:] /OUT:

Conversion (general case) /FIL: [/OUT:]

Conversion /WRK:{*_SCT_VIASBE} /FIL: /SBS: /SBF: [/OUT:]

Conversion /WRK:{*_VIASBE} /FIL: /SBS: [/OUT:]

Conversion /WRK:P160_2_SDD_VIA_FRPPAIEMENT /FIL: /FPT: /FPR: [/OUT:]
```

```
Conversion /WRK:{*_RB7|*_RB9} /FIL: /RBB: [/OUT:]

Prepare file /WRK:Prepa /FIL: /OUT:

Invalid records /WRK:DelRejet /FIL: /REJ: /OUT:

Valid records /WRK:DelSigne /FIL: /REJ: /OUT:

Bank Import /WRK:{BCP_GEN_CERG|CONV_FMT_CERG} /FIL: /OUT:

Filter bank statements /WRK:UPDATEFILE /FIL: /UFP: /UFT: /UFS: /UFC: /OUT:

Bank Import /WRK:BFI_MQ /FIL: /QUE: /BNK: /OUT:

File generation for U2 /FIL: /OUT:

Execute /ARG: [/FIL: /OUT:]
```

Exemples

Vous devez positionner les paramètres de base (cf. plus haut) avant les paramètres spécifiques des exemples qui suivent.

Exemple : Dans le cas où un utilisateur **XRT**, ayant pour mot de passe **XRT**, veut exécuter un traitement synchrone sur un workgroup **SXBE32TO64**, il faut ajouter cette partie de code au préalable :

```
Sage.fcs.apifmt.exe /NOW:Y /WKG:SXBE32TO64 /USR:XRT /PWD:WFJU
```

Remittance slip /WRK:{REMITTANCESLIP|BORDEREAU} /FIL: /OUT:

```
/PRD:BCP /PRJ:VIR_160 /WRK:BORDEREAU /FIL:"C:\file.160" /OUT:"C:\file.pdf"
```

View file (general case) /FIL: /OUT:

```
PRD:SMP_P5 /PRJ:SCT /WRK:SCT_VISU_PAY /FIL:"C:\file.xml"
/OUT:"C:\file.pdf"
```

View file (Ebics) /WRK:{HAA_VISU|HKD_VISU|HTD_VISU} /FIL: /EBP: /OUT:

```
/PRJ:EBICS_Requests /WRK:HKD_VISU /FIL:"C:\file.hkd" /EBP:Partner
/OUT:"C:\file.pdf"
```

View with banking rights /H00: /C00: [/C01:] /B00: /TOT: /OUT:

```
/PRD:PDS /PRJ:SCT /WRK:SCT_VISU_PAY /H00:"C:\file.H00" /C00:"C:\file.C00"
/B00:"C:\file.B00" /TOT:"C:\file.TOT" /OUT:"C:\file.pdf"
```

View report /WRK:ODBC_TXT_VISU /RPT: [/LAN:] /ZIP: /OUT:

```
/PRD:SMP_P5 /PRJ:EDITIONS /WRK:ODBC_TXT_VISU /RPT:ticketr.rpt
/ZIP:"C:\file.zip" /OUT:"C:\file.pdf"
```

View report /WRK:SQL_FORMULA_VISU /RPT: [/LAN:] /PAR: [/SQL:] /OUT:

```
/PRD:SMP_P5 /PRJ:EDITIONS /WRK:SQL_FORMULA_VISU /RPT:status.rpt
/PAR:"title='Statuts des fichiers de
signature';partner='*';protocol='*';service='*';client='*';int=' -
';ext='';cre=' - ';sta='Fichier ajouté, Fichier archivé, Fichier bloqué,
Fichier préparé, Fichier signé';l0='Signature
interne';l1='Entité';l2='Protocole';l3='Service';l4='Client';c1='Référence
';c2='Référence externe';c3='Date d''ajout';c4='Montant';c5='Nb signatures
réalisées';c6='Première signature';c7='Premier signataire';c8='Deuxième
signature';c9='Statut';edit='Edité
le';yes='Oui';no='Non';ajoute='Ajouté';prepare='Préparé';bloque='Bloqué';a
rchive='Archivé';signe='Signé';refustotal='Refus
total';rejetstotal='Rejets total';avecrejets='avec Rejets';avecrefus='avec
Refus';verrouille='Verrouillé';afreject='modifié par le module anti-
fraude'" /SQL:"{S_TFI.CNTR_TFI} in[1,2,64,32,16]" /OUT:"C:\file.pdf"
```

View report /WRK:SQL_TABLES_VISU /RPT: [/LAN:] [/SQL:] /OUT:

```
/PRD:SMP_P5 /PRJ:EDITIONS /WRK:SQL_TABLES_VISU /LAN:1033 /RPT:client.rpt
/SQL:"{S_SES."VPS_SES"}='BNP' AND {S_SES."PRO_SES"}=134217728 AND
{S_SES."VFS_SES"}='AFB160' AND {S_SES."CLIENT_SES"}='SAGE' AND
{S_SES."SIGNINT_SES"}=0" /OUT:"C:\file.pdf"
```

Conversion (general case) /FIL: [/OUT:]

```
/PRD:SMP_P5 /PRJ:CONVERTISSEURS /WRK:VIR160_TO_SCT03 /FIL:"C:\file.160"
```

Conversion /WRK:{*_SCT_VIASBE} /FIL: /SBS: /SBF: [/OUT:]

```
/PRD:SMP_P5 /PRJ:VIR_160 /WRK:VIR160_TO_SCT_VIASBE /FIL:"C:\file.160"
/SBS:SCT /SBF:SALARY
```

Conversion /WRK:{*_VIASBE} /FIL: /SBS: [/OUT:]

```
/PRD:SMP_P5 /PRJ:PRE_160 /WRK:PREL160_TO_SDD_VIASBE /FIL:"C:\file.160"
/SBS:SDD
```

Conversion /WRK:P160_2_SDD_VIA_FRPPAIEMENT /FIL: /FPT: /FPR: [/OUT:]

```
/PRD:U2 /PRJ:PRE_160 /WRK:P160_2_SDD_VIA_FRPPAIEMENT /FIL:"C:\file.160"
/FPT:160 /FPR:1
```

Conversion /WRK:{*_RB7|*_RB9} /FIL: /RBB: [/OUT:]

```
/PRD:SMP_P5 /PRJ:CONVERTISSEURS /WRK:VIR160_TO_SCT_RB7 /FIL:"C:\file.160"
/RBB:1
```

Prepare file /WRK:Prepa /FIL: /OUT:

```
/PRD:PDS /PRJ:SCT /WRK:Prepa /FIL:"C:\file.xml" /OUT:"C:\file.zip"
```

Invalid records /WRK:DelRejet /FIL: /REJ: /OUT:

```
/PRD:PDS /PRJ:SCT /WRK:DelRejet /REJ:"C:\fil.rej" /FIL:"C:\file.xml"
/OUT:"C:\filereject.xml"
```

Valid records /WRK:DelSigne /FIL: /REJ: /OUT:

```
/PRD:PDS /PRJ:SCT /WRK:DelSigne /REJ:"C:\fil.rej" /FIL:"C:\file.xml"
/OUT:"C:\fileaccept.xml"
```

Bank Import /WRK:{BCP_GEN_CERG|CONV_FMT_CERG} /FIL: /OUT:

```
/PRD:BCP /PRJ:AEB43 /WRK:BCP_GEN_CERG /FIL:"C:\file.aeb43"
/OUT:"C:\file.conv"
```

Filter bank statements /WRK:UPDATEFILE /FIL: /UFP: /UFT: /UFS: /UFC: /OUT:

```
/PRD:BCP /PRJ:AEB43 /WRK:UPDATEFILE /FIL:"C:\file.aeb43" /UFP:INTER
/UFT:FILE /UFS:AEB43 /UFC:SAGE /OUT:"C:\file.out"
```

Bank Import /WRK:BFI_MQ /FIL: /QUE: /BNK: /OUT:

```
/PRD:BCP /PRJ:AEB43 /WRK:BFI_MQ /FIL:"C:\file.aeb43" /QUE:QU1 /BNK:BNP
/OUT:"C:\file.out"
```

File generation for U2 /FIL: /OUT:

```
/PRD:U2 /PRJ:AFB320 /WRK:Standard /FIL:"C:\file.320" /OUT:"C:\file.conv"
```

Execute /ARG: [/FIL: /OUT:]

```
/PRD:PDS /PRJ:SCT /WRK:SCT_TEST /ARG:"$INPUTFILE$ Y P $OUTPUTFILE$"
/FIL:"C:\file.xml" /OUT:"C:\file.pdf"
```

Dans ce cas, le traitement **SCT_TEST** a été créé par copie du traitement SCT_VISU_DET, en supprimant les types **Traitement permettant de visualiser un fichier** et **Vue permettant la gestion des pouvoirs bancaires**.

Application sage.fcs.pwdencode.exe

Cette application console permet d'encoder le mot de passe d'un utilisateur afin d'éviter qu'il ne circule en clair sur le réseau.

L'encodage peut être effectué en base64 uniquement (par défaut), ou chiffré (avec la clé publique du service d'authentification) puis encodé en base64.

Installation

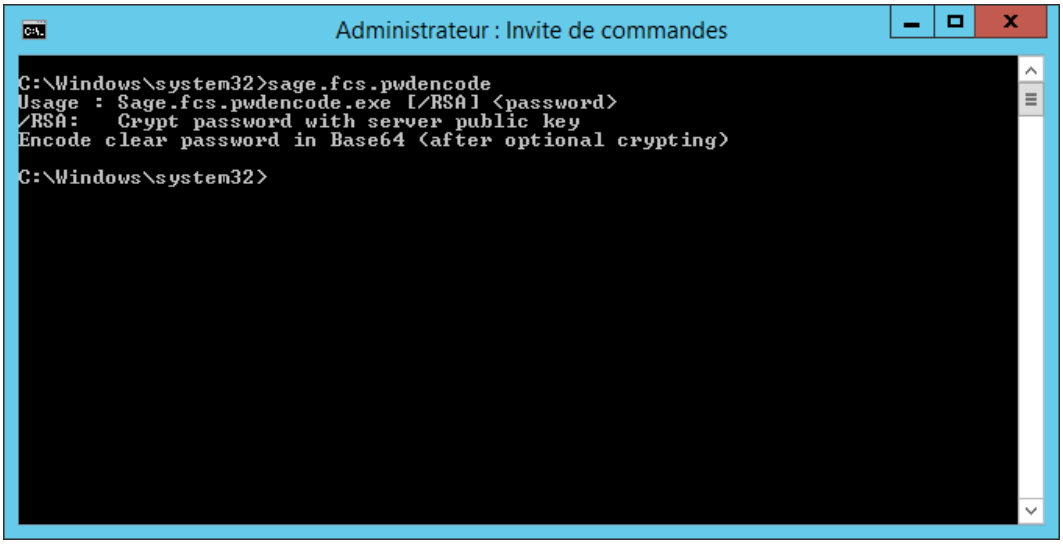
L'installation de cette application est effectuée par l'installation du produit **Sage XRT Common Services**, par la sélection du composant **FCS DLLs** ou du composant **Partie cliente des services SCAS, SCPS et SCDTS**. Hors désélection de ces deux composants lors d'une installation personnalisée, cette application est donc installée par défaut dans toutes les configurations, sauf **Sites Web (SCAS, SCPS, SCDTS et WIDGETS)**.

Deux fichiers sont installés :

- *sage.fcs.pwdencode.exe*
- *sage.fcs.pwdencode.exe.config*

Le fichier de configuration **sage.fcs.pwdencode.exe.config** associé à l'exécutable **sage.fcs.pwdencode.exe** est utilisé pour la déclaration de l'emplacement du service d'authentification dans le cas d'un chiffrement (obtention de la clé publique du certificat).

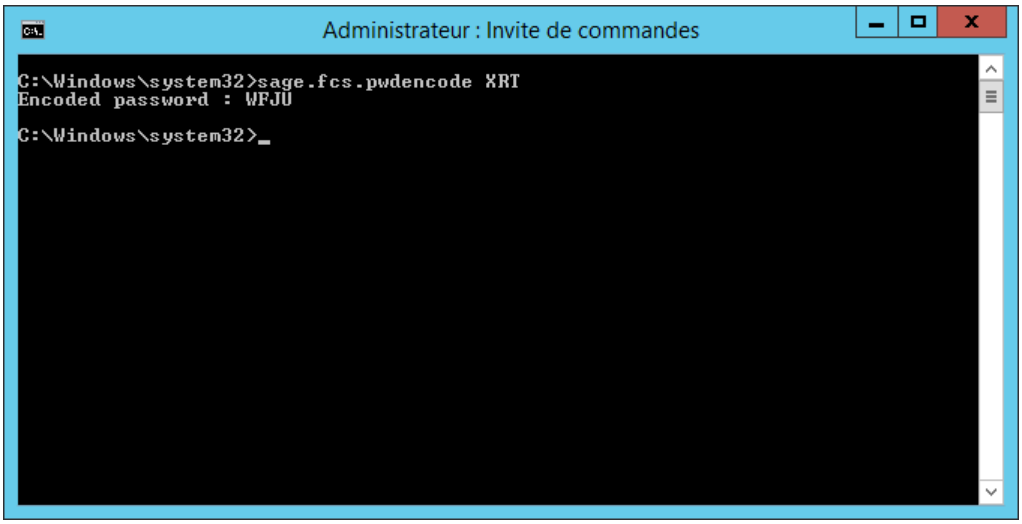
Utilisation



Encodage en base64 uniquement

L'utilisation du fichier de configuration *sage.fcs.pwdencode.exe.config* n'est pas nécessaire dans ce cas puisque l'encodage se fait en local (sans avoir besoin de contacter le service d'authentification).

Le mot de passe : **XRT** est traduit par : **WFJU**.



Chiffrement et encodage en base64

Description du fichier de configuration

Le contenu du fichier est par défaut le suivant :

```
<?xml version="1.0" encoding="utf-8" ?>

<configuration>

  <appSettings>

    <add key="securitylevel" value="http"/>

    <add key="http_servicehost_SCASServer"
value="http://localhost:80/Auth" />

    <add key="https_servicehost_SCASServer"
value="https://localhost:443/Auth" />

  </appSettings>

</configuration>
```

Ce fichier permet d'effectuer la déclaration du service d'authentification qui doit être contacté pour la vérification du mot de passe de l'utilisateur.

Dans le cas où celui-ci fonctionne en mode sécurisé *https*, il faut apporter la modification à la ligne suivante :

```
<add key="securitylevel" value="https"/>
```

Pour un mode non sécurisé, la ligne suivante est exploitée :

```
<add key="http_servicehost_SCASServer" value="http://localhost:80/Auth" />
```

Pour un mode sécurisé, la ligne suivante est exploitée :

```
<add key="https_servicehost_SCASServer" value="https://localhost:443/Auth"
/>
```

Vous devez effectuer la modification de *value* pour tenir compte des spécificités de chaque installation.

Rappel :

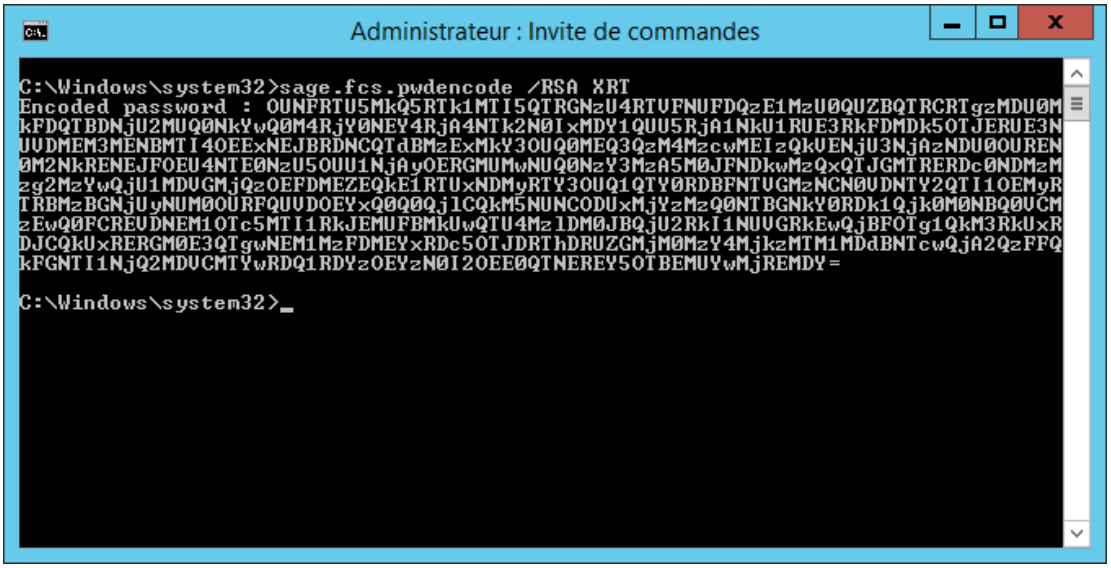
L'utilisation d'un certificat au niveau du service d'authentification (pour effectuer le déchiffrement du mot de passe) se fait en modifiant le fichier de configuration de ce dernier (*Sage.SCASServer.Service.exe.config*) et en renseignant le numéro de série du certificat sur la ligne suivante :

<add key="serialnumberforpwdcrypt" value="3526df91b8be9ab046a226d0390a764f" />

Si *value* dans cette zone est laissée vide, alors le paramètre **/RSA** ne peut pas être exploité avec ce service d'authentification.

Utilisation de l'exécutable

L'utilisation du chiffrement se fait en ajoutant un paramètre **/RSA** sur la ligne de commande.



Nous vous conseillons de rediriger la sortie de la console dans un fichier afin de récupérer le mot de passe chiffré/encodé pour une exploitation future (ex. : **> pwd.txt**).

Annexe

Colonne CATEGORY(S) de la table de transcodage TC DTSPAYCAT_TRANS_PAY :

[SCT VTSDOM 001.001.03][902100]

[SCT VTSIN 001.001.03][903100]

[VTSDO Pain 001.001.03][740796]

[VTSIN Pain 001.001.03][739797]

[VTSDO Pain 001.001.02][740802]

[VTSIN Pain 001.001.02][739801]

[VTSDO Pain 001.001.03 CBI][740825]

[VTSDO Pain 001.001.03 CBI NS][740826]

[VTSIN Pain 001.001.03 CBI][739798]

[VTSIN Pain 001.001.03 CBI NS][739799]

[SCT VTSDOM 001.001.03 BANK OF SPAIN][770002]

[SCT VTSDOM 001.001.03 CBI][770003]

[SCT VTSDOM 001.001.03 CBI NS][770004]

[SCT VTSIN 001.001.03 CBI][771005]

[SCT VTSIN 001.001.03 CBI NS][771006]

[AFB320URG-VTS][003800]

[MT101VTS][001101]

[AEB34VTS][016116]

[AFB160VTS][010191]

[DTAUSDTE-VTS][011111]

[DTAZVEUE-VTS][004105]

[PAY912VTSDO][008108]

[PAY912VTSIN][002102]

[PAY96AABNVTSDO][710733]

[PAY96AABNVTSIN][711734]

[PAY96ABOAVTSDO][724757]

[PAY96ABOAVTSIN][725759]

[PAY96ACHVTSDO][510561]

[PAY96ACHVTSIN][610611]

[PAY96ADBVTSDO][712737]

[PAY96ADBVTSIN][713738]

[PAY96ASCBVTSDO][722753]

[PAY96ASEBVTSDO][717745]

[PAY96ASEBVTSIN][718746]

[PAY96AVTSDO][500502]

[PAY96AVTSIN][600603]

[SCT VDOM 001.001.02][738774]

[SCT VDOM 001.001.03][756793]

[SCT VDOM 001.001.03 BANK OF SPAIN][738823]

[SCT VDOM 001.001.03 CBI][738819]

[SCT VDOM 001.001.03 CBI NS PMRQ][738820]

[SCT VDOM 001.003.03][780001]

[VDOM Pain 001.001.02][740776]

[VDOM Pain 001.001.03][758795]

[SCT VINT 001.001.02][737773]

[SCT VINT 001.001.03][755792]

[SCT VINT 001.001.03 CBI][737821]

[SCT VINT 001.001.03 CBI NS PMRQ][737822]

[SCT VINT 001.003.03][781002]

[VINT Pain 001.001.02][739775]

[VINT Pain 001.001.03][757794]

[VINT Pain 001.001.03 CBI][757828]

[VINT Pain 001.001.03 CBI NS][757829]

[AEB3412DOM][060060]

[AEB3412VINT][070070]

[AEB341VDOM][734734]

[AEB341VINT][729729]

[AEB34VDOM][016116]

[AEB68VDOM][732768]

[AFB160][010110]

[AFB320][003103]

[CBIIPCVDOM][014162]

[CIRI128VDOM][022187]

[CLIEOP3VDOM][012112]

[CMI101][001129]

[DTA826VDOM][759798]

[DTA827VDOM][520562]

[DTA830VINT][620621]

[DTA836VINT][630631]

[DTAUSDTE-VDOM][011111]

[DTAUSIZV-VDOM][011111]

[DTAUSVDOM][011111]

[DTAZV-AZV][004104]

[DTAZV-ESU][004106]

[DTAZV-EUE][004105]

[ETB101][001127]

[FIN101][001130]

[MT100][005105]

[MT101][001101]

[MT101C2B][741777]

[MT103][726760]

[MX101C2B][753789]

[OPAE22VDOM][706729]

[OPAE27VDOM][707730]

[PAY912VDOM][008108]

[PAY912VINT][002102]

[PAYEXT96ABOAVDOM][724756]

[PAYEXT96ABOAVINT][725758]

[PAYMUL96AABNVDOM][710733]

[PAYMUL96AABNVINT][711734]

[PAYMUL96ACHVDOM][510561]

[PAYMUL96ACHVINT][610611]

[PAYMUL96ACRGVDOM][500501]

[PAYMUL96ACRGVINT][600601]

[PAYMUL96ADBVDOM][712735]

[PAYMUL96ADBVINT][713736]

[PAYMUL96ASCBVDOM][723754]

[PAYMUL96ASCBVINT][722752]

[PAYMUL96ASEBVDOM][717744]

[PAYMUL96ASEBVINT][718746]

[PS2VDOM][015115]