



arcOpole PRO Intervention Mobile

Guide Installation et Paramétrage Version 3.1

30/07/2024

À propos du Guide

Les informations contenues dans ce document pourront faire l'objet de modifications sans préavis de la part de 1Spatial.

Sauf mention contraire, les sociétés, les noms et les données utilisés dans les exemples sont fictifs.

Aucune partie de ce document ne peut être reproduite ou transmise à quelque fin ou par quelque moyen que ce soit, électronique ou mécanique, sans la permission expresse et écrite de 1Spatial.

Limitation de garantie et de responsabilité

1Spatial a réalisé tous les tests nécessaires et suffisants du Progiciel et a vérifié la conformité de son guide.

Le client reconnaît que dans l'état actuel de la technologie informatique, le fonctionnement du Progiciel est susceptible d'être interrompu ou affecté par des bogues. 1Spatial ne garantit pas que le Progiciel, le média d'installation, la clef ou la documentation livrés soient exempts d'erreurs, de bogues ou d'imperfections.

Ainsi, le client doit effectuer toutes les sauvegardes, prévoir et effectuer toutes les procédures de remplacement en vue d'éventuelles défaillances du Progiciel, prendre toutes les mesures appropriées pour se prémunir contre toute conséquence dommageable due à l'utilisation ou la non utilisation du Progiciel.

Il est expressément convenu que 1Spatial ne sera en aucun cas responsable des dommages directs ou indirects dus à l'utilisation du Progiciel.

L'utilisation du Progiciel est soumise à la signature par le client du contrat de droit d'utilisation des logiciels de 1Spatial.

Marques déposées

Ce progiciel est une marque déposée de 1Spatial.

Ce progiciel, développé par 1Spatial, est une marque déposée et est la propriété exclusive de 1Spatial. Toutes les autres marques citées appartiennent à leurs propriétaires respectifs.

Copyright © 2024, 1Spatial. Tous droits réservés.

TABLE DES MATIÈRES

1 PRÉAMBULE	4
1.1 Pré requis	4
1.2 Configuration de la tablette pour un bon fonctionnement de l'application	4
1.2.1 Configuration minimum	4
1.2.2 Configuration confortable	4
1.2.3 Configuration optimale	5
2 PRINCIPES.....	6
2.1 Carte.....	6
2.2 Intervention	6
3 INSTALLATION	8
3.1 Répertoire de travail	8
3.2 Serveur.....	9
4 INFORMATIONS SYSTÈME.....	10
4.1 Autorisations.....	10
4.2 Désinstallation et mise à jour de l'application.....	10
4.3 Gestion de la mémoire.....	10
5 PARAMÉTRAGES	12
5.1.1 Paramétrage global	12
5.1.2 Paramétrage associé à une carte	17
5.2 Traitement des données	22
5.2.1 Fonds de plan tuilé	22
5.2.2 Données vectorielles	24
6 MISE À JOUR DU PATRIMOINE	32
6.1 Principes.....	32
6.2 Mise en œuvre	32
6.2.1 Organisation	32
6.2.2 Paramétrage accès au répertoire contenant le patrimoine	33
6.2.3 Configuration du script.....	33
6.2.4 Définition de la tâche planifiée	34
7 ANNEXES	36
7.1 Déclaration certificat SSL	36
7.1.1 Certificat serveur	36
7.1.2 Clé privée	36
7.1.3 Autorité de certification	36
7.2 Principe de connexion.....	36

1 PRÉAMBULE

Ce document a pour but de détailler la procédure d'installation et de paramétrage du volet mobile de l'application arcOpole PRO Intervention.

Ce document concerne arcOpole PRO Intervention Mobile 3.0.

arcOpole PRO Intervention Mobile 3.0 est compatible avec arcOpole PRO Intervention Back Office 3.0.

1.1 PRÉ REQUIS

Système d'exploitation :

- ▶ Android 8.1 ;
- ▶ Android 9.0 (Pie) ;
- ▶ Android 10.0 ;
- ▶ Android 11.0 ;
- ▶ Android 12.0 ;
- ▶ Android 13.0 (Tiramisu) : le mode « splité » n'est pas supporté.

Données :

- ▶ L'application nécessite obligatoirement un paquetage de cartes mobiles.

Connexion :

- ▶ Portal for ArcGIS ;
- ▶ Utilisateur nommé de type « Viewer ».

La distribution d'arcOpole PRO Intervention Mobile est livrée avec arcOpole PRO Serveur dans le répertoire : « ...\\aOpServeur\\Ressources_Installation\\Intervention\\installation\\aopmobile ».

Cette distribution contient un répertoire nommé « *files* » contenant les sous répertoires nommés « *data* » et « *config* » et un répertoire nommé « *apk* » contenant un fichier nommé « *aOPInterventionMobile_3.0.apk* ».

1.2 CONFIGURATION DE LA TABLETTE POUR UN BON FONCTIONNEMENT DE L'APPLICATION

1.2.1 Configuration minimum

- ▶ Processeur : >= 2 GHz ;
- ▶ RAM : 3 Go ;
- ▶ Stockage interne : 32 Go.

1.2.2 Configuration confortable

- ▶ Processeur : >= 2.4 GHz ;

- ▶ RAM : 4Go ;
- ▶ Stockage interne : 64 Go.

1.2.3 Configuration optimale

- ▶ Processeur : >= 2.6 GHz ;
- ▶ RAM : 6 Go ;
- ▶ Stockage interne : 128 Go.

2 PRINCIPES

arcOpole PRO Intervention Mobile est composé de deux modules :

- ▶ Carte ;
- ▶ Intervention.

2.1 CARTE

La carte est composée à minima d'un paquetage de cartes mobiles et peut être complétée par un fond de plan sous la forme d'un paquetage de tuiles de carte.

Les fonctions suivantes sont disponibles :

- ▶ Gestion de l'affichage des couches contenues dans le paquetage de cartes mobiles ;
- ▶ Gestion de l'affichage du fond de plan ;
- ▶ Consultation des informations et des documents attachés aux éléments des couches contenues dans le paquetage de cartes mobiles ;
- ▶ Localisation des éléments des couches contenues dans le paquetage de cartes mobiles.

La prise en compte des documents attachés nécessite la création de pièces jointes.

Le module prend également en compte une organisation hiérarchique des éléments. Pour ce faire, il est nécessaire de créer des classes de relation entre les différentes couches concernées.

Les propriétés de consultation des éléments de chaque couche sont définies dans le fichier de configuration « **map.json** ».

Les propriétés de localisation des éléments de chaque couche sont définies dans le fichier de paramétrage « **localisation.json** ».

Le paquetage de cartes mobiles peut être mis à jour par une synchronisation demandée par l'utilisateur de l'application.

La solution met à disposition un script Python permettant de générer un nouveau paquetage de cartes mobiles à partir d'un fichier carte ArcGIS Pro.

Les propriétés de la synchronisation sont définies dans le fichier de paramétrage « **infosmmpk.json** ».

Chaque carte permet la gestion d'un réseau.

L'application permet de gérer plusieurs cartes.

Si plusieurs cartes sont paramétrées, au démarrage l'application propose le choix de la carte à exploiter.

2.2 INTERVENTION

Ce module permet de traiter toutes les interventions affectées à l'utilisateur connecté à l'application.

Les fonctions suivantes sont disponibles :

- ▶ Récupération des interventions à traiter ;

- ▶ Envoi des interventions traitées ;
- ▶ Traitement des interventions ;
- ▶ Création d'une intervention corrective ;
- ▶ Saisie d'une intervention curative.

3 INSTALLATION

- ▶ Copier le fichier apk dans le répertoire Download de la tablette ;
- ▶ Pour installer l'application, ouvrir une application de gestion de fichiers sur la tablette et cliquer sur le fichier : « **\Download\arcOpole_Pro_3.0.apk** ».

L'installation est réalisée à partir du gestionnaire de fichiers de la tablette ce qui nécessite une autorisation.

La demande d'autorisation d'installation à partir de cette source de données va être demandée au cours de la procédure d'installation de l'application.

! L'application ne doit pas être ouverte après son installation.

3.1 RÉPERTOIRE DE TRAVAIL

Situé dans la mémoire interne, ce répertoire contient l'ensemble des fichiers exploités par l'application. Après avoir connecté la tablette à un ordinateur et ouvert un explorateur sur la mémoire interne de la tablette, créer l'arborescence des répertoires suivants s'ils n'existent pas déjà : **/Android/data/fr.onespatial.aopinterventionmobile/files**

Pour information, ce répertoire contient :

- ▶ Le répertoire « **config** », qui contient :
 - Le fichier de configuration principal « **application.json** » ;
 - Un sous-répertoire par réseau, contenant lui-même :
 - Le fichier « **map.json** » ,
 - Le fichier « **localisation.json** » .
- ▶ Le répertoire « **data** », qui contient :
 - Un sous-répertoire par réseau, contenant lui-même :
 - Le fichier MMPK,
 - Le fichier « **infosmmpk.json** » associé,
 - Le fichier « **aopmobile.bks** » .
 - L'application va également créer dans ce répertoire les fichiers <utilisateur>.zip, <utilisateur>.db et <utilisateur>.db-journal de la base de données des interventions propres à chaque utilisateur, téléchargées depuis le serveur ;
 - Eventuellement, un répertoire contenant les fichiers .TPK utilisés pour les fonds de plan.

Si une carte SD amovible est utilisée pour le stockage des fichiers .TPK, il faut également créer l'arborescence « **/Android/data/fr.onespatial.aopinterventionmobile/files/data** » sur la carte SD.

! Par défaut les cartes SD sont livrées formatées en FAT32 et ne peuvent contenir de fichiers de plus de 4 Go, il faut avant utilisation les formater en exFAT.

3.2 SERVEUR

Le bon fonctionnement de l'application mobile nécessite la présence des fichiers suivants côté serveur :

- ▶ « *sqlite-jdbc-3.30.1.jar* » : répertoire « *\1Spatial\OpServeur\server\apache-tomcat-9\shared* » ;
- ▶ « *export.db* » : répertoire « *\1Spatial\OpServeur\modules\intervention* ».

Ces fichiers sont nécessaires à la synchronisation des interventions associées à l'utilisateur authentifié.

4 INFORMATIONS SYSTÈME

4.1 AUTORISATIONS

Lors du premier lancement, des autorisations d'accès sont demandées par l'application. Les autorisations nécessaires sont les suivantes :

- ▶ Accès à la localisation (demandé lors du premier accès à la carte) ;
- ▶ Accès aux photos et aux fichiers multimédia ;
- ▶ Utilisation de l'appareil photo.

Sous Android 11, lorsque les permissions sont rejetées plus d'une fois, l'application peut être amenée à ne plus demander les autorisations aux lancements suivants. En effet, le système considère implicitement que lorsque l'utilisateur a refusé les permissions plus d'une fois, il n'a plus à les redemander. La modification des permissions ne pourra alors se faire que par l'accès aux détails de l'application dans les paramètres du système.

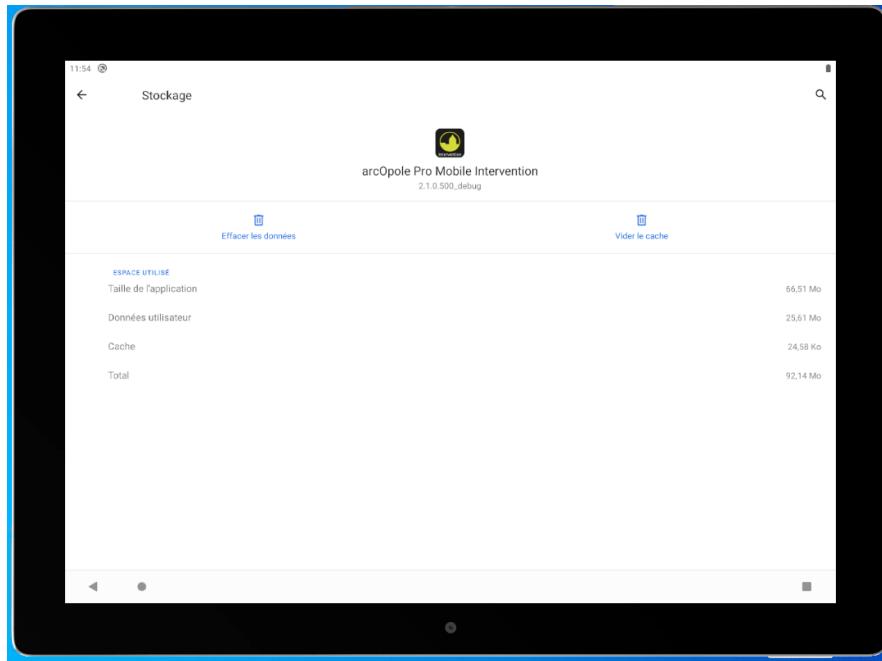
4.2 DÉSINSTALLATION ET MISE À JOUR DE L'APPLICATION

Lors de la désinstallation de l'application, le répertoire « **Android/data/fr.onespatial.aopinterventionmobile** » ainsi que son contenu, sont supprimés aussi bien de la mémoire interne que de la carte SD externe. Le répertoire « **Documents/arcOpole PRO Intervention Mobile** » n'est pas supprimé.

Lors d'une mise à jour de l'application vers une version supérieure à la 2.0, il est donc recommandé de ne pas désinstaller l'ancienne version au préalable, sinon les fichiers utilisés par l'application seront effacés et devront être rétablis après installation de la nouvelle version.

4.3 GESTION DE LA MÉMOIRE

En consultant les informations sur l'application, il est possible d'effacer les données utilisées par celle-ci dans l'espace de stockage :



Cette opération va supprimer le répertoire « ***Android/data/fr.onespatial.aopinterventionmobile*** » et tout son contenu. Cette opération ne doit donc jamais être réalisée sans avoir au préalable effectué une copie de sauvegarde du répertoire, afin de le restaurer par la suite.

Par contre la commande « Vider le cache » disponible sur cette même page n'a pas d'incidence sur le fonctionnement de l'application.

5 PARAMÉTRAGES

Les fichiers de configuration de l'application sont disponibles dans le répertoire « ... \aOpServeur\Ressources_Installation\Intervention\installation\aoopmobile\files\config » de la distribution de arcOpole PRO Serveur.

Ce répertoire contient le fichier principal de configuration « **application.json** », ainsi qu'un sous-répertoire « **RESEAU** » qui contient le fichier de paramétrage de la carte « **map.json** » et le fichier de paramétrage de la localisation « **localisation.json** ». Ces fichiers doivent être copiés dans des répertoires différents pour chaque carte et donc chaque réseau pris en charge par l'application.

Une fois les fichiers de configuration édités sur un ordinateur, il faut copier le répertoire « config » et tout son contenu sur la tablette dans le répertoire de travail « **Android/data/fr.onespatial.aopinterventionmobile/files** ».

5.1.1 Paramétrage global

Les paramètres généraux sont décrits dans le fichier « **application.json** ».

5.1.1.1 Sécurité et authentification

La balise « **portalUrl** » contient l'url d'accès à Portal pour une authentification par utilisateur nommé.

```
"security": {
  "portalUrl": "https://aop-d-arcgis11.1spatial.fr/portal/"
},
```

5.1.1.2 Echanges avec le serveur arcOpole PRO

```
"aopServerUrl": "https://geowbs.montpellier3m.fr:443/arcopolepro",
```

La balise « **aopServerUrl** » indique l'url d'accès au serveur arcOpole PRO :

```
"certificatCrtFileName": "",
```

La balise « **certificatCrtFileName** » contient le nom du certificat du serveur arcOpole PRO.

Dans le cas où celui-ci n'est pas certifié et validé par un organisme de confiance, il faut :

- ▶ Copier le certificat dans le répertoire « **data** » sur la tablette ;
- ▶ Installer le certificat :

```
"byPassCertificat": true,
```

La balise « **byPassCertificat** » (true/false) définit les échanges avec le serveur.

A la valeur « true » les échanges avec le serveur ne seront plus encryptés et protégés par un certificat SSL/TLS.

5.1.1.3 Import et export des interventions

```
"aopApprovalUrl":"/rest/auth/oauth2/approval",
"aopProfilsUrl":"/rest/module/intervention/user/profils",
"aopDownloadSqliteUrl":"/rest/module/intervention/sqllite/export?reseau=[RESEAU]",
"aopUploadSqliteJobIdUrl":"/rest/module/intervention/sqllite/import/jobId",
"aopUploadSqliteFileUrl":"/rest/file/upload",
"aopUploadSqliteImportUrl":"/rest/module/intervention/sqllite/import",
"aopUploadSqliteStatusUrl":"/rest/module/intervention/sqllite/import/status",
```

L'ensemble de ces paramètres définissent la synchronisation des interventions sur la tablette.

Ces balises ne sont pas à modifier.

```
"restartAppAfterDownloadInterv" : false,
```

La balise « **restartAppAfterDownloadInterv** » (true/false) indique si l'application doit être redémarrée suite à l'import des interventions associées à l'utilisateur connecté.

5.1.1.4 Mise à jour du patrimoine

```
"updatePatrimoine":"manual",
```

La balise « **updatePatrimoine** » (auto/manual) indique le mode de mise à jour du patrimoine réseau (MMPK) :

- ▶ auto : vérification toutes les heures d'une mise à jour du fichier, même si l'application est fermée et téléchargement si nouvelle mise à jour ;
- ▶ manual : mise à jour réalisée par l'utilisateur à partir de l'application et de l'option « Mise à jour patrimoine ».

Suite à une mise à jour, une notification apparaît spécifiant qu'une mise à jour vient d'être effectuée. Il est également indiqué sur la carte la date de dernière mise à jour.

```
"restartAppAfterUpdateMapPatrimoine" : false,
```

La balise « **restartAppAfterUpdateMapPatrimoine** » (true/false) indique si l'application doit être redémarrée suite à la mise à jour du patrimoine.

5.1.1.5 Gestion de la fiche d'informations d'une intervention

```
"equipmentTabFirst" : true,
```

Cette balise indique si la fiche d'informations d'une intervention s'ouvre sur l'onglet « Equipements » ou pas. Par défaut, sa valeur est **false**.

- ▶ **false** ou balise absente : la fiche d'informations d'une intervention s'ouvre sur l'onglet « Demande » ;
- ▶ **true** : la fiche d'informations d'une intervention s'ouvre sur l'onglet « Equipements ».

5.1.1.6 Déclaration de la nomenclature associée aux communes

```
"communeNomenclature": "COMMUNE",
```

Cette balise indique la nomenclature associée aux communes.

En l'absence de cette balise, la nomenclature utilisée par défaut est « COMMUNES ».

5.1.1.7 Sectorisation

```
"sectorizedUser": true,  
"sectorNomenclature" : "ENTREPRISE",
```

- ▶ **"sectorizedUser"**: false : Balise indiquant si l'utilisateur est en mode Sectorisation. Par défaut, sa valeur est **false**.
 - **true** : l'utilisateur est en mode Sectorisation ;
 - **false** ou balise absente : l'utilisateur n'est pas en mode Sectorisation.
- ▶ **"sectorNomenclature" : "<nom nomenclature>"** : Balise donnant le nom de la nomenclature utilisée pour la liste des secteurs. Cette valeur doit être renseignée et valide pour permettre la création d'interventions géographiques quand la gestion de la sectorisation est activée.

5.1.1.8 Gestion de l'affichage des labels associés aux nomenclatures dans la fiche d'un équipement

Il s'agit de définir les labels utilisés pour l'affichage des données Commune et Secteur issues de la nomenclature dans la fiche d'informations d'un équipement.

```
"nomenclatureAttributes":  
{  
    "commune": "COMMUNE PRINCIPALE",  
    "secteur": "SECTEUR"  
},
```

Ces paramètres sont facultatifs. Par défaut, le label associé :

- ▶ à la commune est « COMMUNE » ;
- ▶ à secteur est « NOM DU SECTEUR ».

5.1.1.9 Intervention géographique

```
"geographicInterventions" : true,
```

Balise indiquant si l'utilisateur peut créer/gérer des interventions géographiques. Par défaut, sa valeur est **false**.

- **true** : l'utilisateur peut créer/gérer des interventions géographiques ;
- **false** ou balise absente : l'utilisateur ne peut pas créer/gérer des interventions géographiques.
- ▶ Le bloc « **geographicMarker** » permet de modifier l'aspect des interventions géographiques sur la carte. Par défaut, une intervention géographique est représenté par un triangle rouge.

```

    "geographicMarker":
    {
        "style": "TRIANGLE",
        "size": 20,
        "color": "#FFFF0000",
        "textSize": 10,
        "textColor": "#FF000000"
    },

```

- **"style": "TRIANGLE"**: Style du marqueur utilisé. Valeur par défaut : **TRIANGLE**. Les valeurs acceptées sont : [**CIRCLE** | **CROSS** | **DIAMOND** | **SQUARE** | **TRIANGLE** | **X**].
 - **"size"**: 20 : Taille du marqueur en density-independent pixels (dp). Valeur par défaut : 20.
 - **"color"**: "#**FFFF0000**" : Couleur du marqueur. Valeur par défaut : "**#FFFF0000** (rouge). Cette valeur est en hexadécimal au format #AARRGGBB.
<AA> = canal alpha (transparence), <RR> = composante rouge, <GG> = composante vert, <BB> = composante bleu.
 - Les balises **"textSize"** et **"textColor"** ne sont pas utilisées et sont réservées pour un usage ultérieur.
- Le bloc « **geographicLayer** » permet de modifier le nom des interventions géographiques dans la légende de la carte. Par défaut, une intervention géographique a pour nom « **Intervention géographique** ».

```

    "geographicLayer":
    {
        "layerName": "Intervention géographique",
        "canIdentify": true,
        "identifyOnlyWhenVisible": true
    },

```

- **"layerName": "Intervention géographique"**: Le nom du layer tel qu'il apparaîtra dans la légende.
- **"canIdentify"** : [**true** | **false**]. Balise indiquant si l'utilisateur peut identifier les objets de ce layer :
 - **true** : l'utilisateur pourra identifier un objet de ce layer ;
 - **false** : l'utilisateur ne pourra pas identifier un objet de ce layer.
- **"identifyOnlyWhenVisible"** : [**true** | **false**]. Balise indiquant si l'utilisateur peut identifier les objets de ce layer suivant sa visibilité :
 - **true** : l'utilisateur pourra identifier un objet de ce layer seulement s'il est visible ;
 - **false** : l'utilisateur pourra identifier un objet de ce layer même s'il n'est pas visible.

5.1.1.10 Equipement hors patrimoine

Un équipement hors patrimoine est un équipement qui n'est pas présent dans la base patrimoniale et qui peut être ajouté à une campagne.

- Le bloc « **equipmentMarker** » permet de modifier l'aspect des équipements hors patrimoine sur la carte. Par défaut, un équipement hors patrimoine est représenté par un triangle rouge.

```

    "equipmentMarker": {
        "style": "CROSS",
        "size": 20,
        "color": "#FFFF0000",
        "textSize": 10,
        "textColor": "#FF000000"
    },

```

- **"style": "TRIANGLE"** : Style du marqueur utilisé. Valeur par défaut : **TRIANGLE**. Les valeurs acceptées sont : [**CIRCLE** | **CROSS** | **DIAMOND** | **SQUARE** | **TRIANGLE** | **X**].
 - **"size": 20** : Taille du marqueur en density-independent pixels (dp). Valeur par défaut : 20.
 - **"color": "#FFFF0000"** : Couleur du marqueur. Valeur par défaut : **"#FFFF0000** (rouge). Cette valeur est en hexadécimal au format #AARRGGBB.
 <AA> = canal alpha (transparence), <RR> = composante rouge, <GG> = composante vert, <BB> = composante bleu.
 - Les balises **"textSize"** et **"textColor"** ne sont pas utilisées et sont réservées pour un usage ultérieur.
- Le bloc « **equipmentLayer** » permet de modifier le nom des équipements hors patrimoine dans la légende de la carte. Par défaut, un équipement hors patrimoine a pour nom « **Equipement hors patrimoine** ».

```

    "equipmentLayer": {
        "layerName": "Equipement hors patrimoine",
        "canIdentify": true,
        "identifyOnlyWhenVisible": true
    },

```

- **"layerName"** : "Equipement hors patrimoine" : Le nom du layer tel qu'il apparaîtra dans la légende.
- **"canIdentify"** : [true | false] : Balise indiquant si l'utilisateur peut identifier les objets de ce layer :
 - **true** : l'utilisateur pourra identifier un objet de ce layer ;
 - **false** : l'utilisateur ne pourra pas identifier un objet de ce layer.
- **"identifyOnlyWhenVisible"** : [true | false] : Balise indiquant si l'utilisateur peut identifier les objets de ce layer suivant sa visibilité :
 - **true** : l'utilisateur pourra identifier un objet de ce layer seulement s'il est visible ;
 - **false** : l'utilisateur pourra identifier un objet de ce layer même s'il n'est pas visible.

5.1.1.11 Suivi application

```

    "logLevel": "DEBUG"

```

Cette balise indique le niveau d'écriture des messages de logs. Par défaut, sa valeur est **ERROR** : seuls les messages d'erreurs seront écrits.

Les valeurs acceptées sont : [**VERBOSE** | **DEBUG** | **INFO** | **WARN** | **ERROR** | **FATAL**]. Pour des raisons de performance, il est recommandé de laisser la valeur par défaut.

5.1.1.12 Emplacement des fichiers de log

Les fichiers de logs générés sont nommés **AOPIMTrace-<suffixe_date_jour>.txt**

- ▶ **Android 9 et inférieur :** Les logs de l'application sont enregistrés dans le répertoire *Logs* du répertoire de travail. Ce répertoire est créé automatiquement ;
- ▶ **Android 10 et supérieur :** Les logs de l'application sont enregistrés dans le répertoire */Documents/arcOpole PRO Intervention Mobile/Logs* de la mémoire interne. Ce répertoire est créé automatiquement.

5.1.2 Paramétrage associé à une carte

Une carte est composée :

- ▶ D'un fond de plan : paquetage de tuiles de carte (TPK) ;
- ▶ Des couches du patrimoine : paquetage de cartes mobiles (MMPK) ;

Les paramètres associés à une carte sont décrits dans les fichiers :

- ▶ « **map.json** » ;
- ▶ « **localisation.json** ».

Les propriétés de chaque carte sont stockées dans un répertoire particulier.

Le nom du répertoire associé à une carte correspond au nom du réseau déclaré dans arcOpole PRO Intervention (fichier « **configuration.json** »).

5.1.2.1 Fichier « map.json »

5.1.2.1.1 Principe

Ce fichier décrit le contenu de la carte exploitée par l'application.

Fonds de plan tuilés

```
"baseLayers": [
    {
        "layerName": "Fond de plan",
        "url": "file:///FdP/Cadastre.tpk",
        "visible": true
    }
],
```

- ▶ **baseLayers** : liste des fonds de plan tuilés disponibles pour la carte. Pour un fond de plan :
 - **layerName** : nom du fond de plan affiché dans la légende de la carte,
 - **url** : chemin d'accès du fichier TPK du fond de plan relatif par rapport au répertoire « */Android/data/fr.onespatial.aopinterventionmobile/files/data* » :
 - Si la valeur commence par « file:// » : fichier stocké en interne,
 - Si la valeur commence par « file:/// » : fichier stocké sur une carte SD externe.
 - **visible** : fond de plan affiché par défaut dans la carte.

Informations

```
"mapTools":  
{  
    "scaleBarVisibility":true,  
    "showVersionMmpk":true  
},
```

► **Gestion de l'affichage d'éléments d'information sur la carte :**

- **scaleBarVisibility** : (*true/false*) : affichage de l'échelle sur la carte ;
- **showVersionMmpk** (*true/false*) : affichage de la date de mise à jour du patrimoine.

Légende

```
"legendGroupOrders": ["Eclairage Public"],
```

► **legendGroupOrders** : définition de la liste des groupes de couches de la carte. Ce paramètre définit également l'ordre d'affichage dans la légende.

Couches

```
"layers": [  
    {  
        "layerName": "Armoire",  
        "groupName": "Eclairage Public",  
        "canIdentify": true,  
        "identifyOnlyWhenVisible": true,  
        "fields": [  
            "id_armoir",  
            "lib_ar moi",  
            "commune",  
            "rue",  
            "competence",  
            "date_trans",  
            "type_arm",  
            "etat_gen"  
        ]  
    },  
],
```

► **layers** : liste des couches présentes dans la carte. Pour chaque couche :

- **layerName** : nom de la couche dans le projet ArcGIS Pro utilisé pour générer le fichier MMPK ;
- **groupName** : groupe de couches auquel appartient la couche ;
- **canIdentify** (*true/false*) : consultation des informations des éléments de la couche ;
- **identifyOnlyWhenVisible** :
 - **true** : la fiche d'informations d'un élément de la couche est accessible uniquement si la couche est visible ;
 - **false** : la fiche d'informations d'un élément de la couche est accessible même lorsque la couche n'est pas visible.
- **fields** : liste des champs consultables (rubrique « FieldName » sous ArcGIS Pro). Dans le cas, où la balise est absente, l'application affiche par défaut tous les champs visibles dans la fenêtre contextuelle associée à la couche.

L'alias du champ est affiché dans la fiche d'informations.

Tables

```
"tables": [
    {
        "tableName": "ENVELOPPE",
        "fields": [
            "MARQUE",
            "MODELE",
            "MATIERE",
            "TYPE_SERRURE",
            "DATE_POSE",
            "CODEINSEE"
        ]
    },
]
```

- **tables** : liste des tables à consulter présentes dans le fichier MMPK. Pour chaque table :
- **tableName** : nom de la couche dans le projet ArcGIS Pro utilisé pour générer le fichier MMPK,
 - **fields** : liste des champs consultables (rubrique « **FieldName** » sous ArcGIS Pro). Dans le cas, où la balise est absente, l'application affiche par défaut tous les champs visibles dans la fenêtre contextuelle associée à la table.

L'alias du champ est affiché dans la fiche d'informations.

5.1.2.1.2 Génération automatique du fichier « map.json »

- Depuis le répertoire :
- « ..\aOpServeur\Ressources_Installation\Intervention\installation\aoopmobile\python », copier le répertoire « **python** » vers un répertoire de travail.
Ce répertoire contient des scripts Python facilitant le paramétrage de l'application Back Office et de l'application mobile.
Ces scripts utilisent Python 3 disponible à partir d'ArcGIS Pro.
 - A partir du répertoire « **python** », sélectionner le fichier « **layers2MobileJson.py** » puis ouvrir le menu contextuel associé.
 - Cliquer sur l'option « **Edit with IDLE (ArcGIS Pro)** », la fenêtre suivante apparaît :



```
layers2MobileJson.py - D:\EEL_logiciels\arcOpolePRO\python\layers2MobileJson.py (3.6.6)
File Edit Format Run Options Window Help
import arcpy, json
import os

class JsonMapLayersConfig:
    def __init__(self, layerName, canIdentify, groupName, identifyOnlyWhenVisible,
                 fields):
        self.layerName = layerName
        self.canIdentify = canIdentify
        self.groupName = groupName
        self.identifyOnlyWhenVisible = identifyOnlyWhenVisible
        self.fields = fields

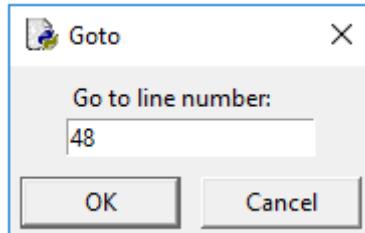
    def toJson(self):
        json = """
        json += "\\"layerName\\":\"" + self.layerName + """
        json += "\\"canIdentify\\":\"" + self.canIdentify + """
        json += "\\"groupName\\":\"" + self.groupName + """
        json += "\\"identifyOnlyWhenVisible\\":\"" + self.identifyOnlyWhenVisible +
        json += "\\"fields\\":[" + "\\" " + "\\", \\"".join(self.fields) + "\\""]
        json += "\"
        return json

class JsonMapTablesConfig:
    def __init__(self, tableName, fields):
        self.tableName = tableName
        self.fields = fields

    def toJson(self):
        json = """
        json += "\\"tableName\\":\"" + self.tableName + """
        json += "\\"fields\\":[" + "\\" " + "\\", \\"".join(self.fields) + "\\"]"
        json += "\"
        return json

class JsonResult:
    def __init__(self, layers, tables):
        self.layers = layers
```

- Se positionner sur la ligne 48 du script en utilisant l'option « Go to Line » du menu « Edit » :



Renseigner le paramètre suivant :

pathFileAprx : chemin d'accès au projet ArcGIS Pro contenant la carte exploitée avec l'application mobile sous la forme d'un fichier MMPK ;

```
pathFileAprx = "D:\\arcgisserver\\arcopolepro\\intervention\\EP\\EP.aprx"
```

! Le caractère « \ » doit être doublé

- Exécuter le script à partir de l'option « Run Module » du menu « Run » ou appuyer sur la touche « F5 ».

Le fichier « map.json » est créé dans le même répertoire que le fichier APRX et porte le nom de la carte associée.

Le script produit un fichier JSON pour chaque carte contenue dans le projet ArcGIS Pro.

5.1.2.2 Fichier « localisation.json »

Ce fichier décrit les localisations proposées par l'application.

5.1.2.2.1 Propriétés des localisations

```
"zoomMinSize": 50,  
"zoomMargin": 30,
```

- ▶ **zoomMinSize** : taille minimum de la zone visible, en mètre, pour le cadrage sur les objets ponctuels ou de petite taille ;
- ▶ **zoomMargin** : marge, en density-independent pixels (dp), ajoutée au cadrage sur un ensemble d'objets. Cela permet d'éviter que des objets soient trop proches du bord de la carte.

5.1.2.2.2 Thématique

Cela concerne les interventions de type campagne et permet l'affichage des équipements en fonction de leur état (traité, non traité).

```
"thematicDisplay" : true,
```

- ▶ **thematicDisplay** : affichage des équipements en fonction de leur état :
 - **true** : les équipements sont affichés en fonction de leur état (traité, non traité) avec une représentation définie pour chaque état ;
 - **false** : les équipements sont affichés selon leurs symbologie standard.
- ▶ **thematicLegend** : définition de la représentation des équipements en fonction de leur état.
- ▶ **undone** : définition de la représentation des équipements à l'état « non traité ».

```
"undone" : {  
    "color": "#FF0000",  
    "markerSize": 30,  
    "markerOpacity" : 100,  
    "lineWidth" : 3,  
    "lineOpacity" : 50,  
    "polygonOpacity" : 25  
},
```

- **color** : couleur au format #RRGGBB (valeurs en hexadécimal) :
 - <RR> = composante rouge,
 - <GG> = composante vert,
 - <BB> = composante bleu.
- **markerSize** : taille du marqueur en density-independent pixels (dp), pour un équipement de type point ;
- **markerOpacity** : transparence en %, pour un équipement de type point ;
- **lineWidth** : épaisseur du trait en density-independent pixels (dp), pour un équipement de type ligne ;
- **lineOpacity** : transparence en %, pour un équipement de type ligne ;
- **polygonOpacity** : transparence en %, pour un équipement de type polygone ;

- ▶ **done** : définition de la représentation des équipements à l'état « traité ».

```
"done" : {  
    "color": "#00FF00",  
    "markerSize": 30,  
    "markerOpacity" : 100,  
    "lineWidth" : 3,  
    "lineOpacity" : 50,  
    "polygonOpacity" : 25  
}
```

- **color** : couleur au format #RRGGBB (valeurs en hexadécimal) :

- <RR> = composante rouge,
- <GG> = composante vert,
- <BB> = composante bleu.
- **markerSize** : taille du marqueur en density-independent pixels (dp), pour un équipement de type point ;
- **markerOpacity** : transparence en %, pour un équipement de type point ;
- **lineWidth** : épaisseur du trait en density-independent pixels (dp), pour un équipement de type ligne ;
- **lineOpacity** : transparence en %, pour un équipement de type ligne ;
- **polygonOpacity** : transparence en %, pour un équipement de type polygone.

5.1.2.2.3 Définition des localisations

```
"layers": [  
    {  
        "displayName": "Armoire",  
        "layerName": "Armoire",  
        "displayFieldName": "LIB_ARMOI",  
        "sqlSearchExpression": "LIB_ARMOI LIKE upper('?%')",  
        "format": "string"  
    }  
]
```

- **displayName** : nom de la couche présente dans le menu de localisation de l'application ;
- **layerName** : nom de la couche correspondante dans le projet ArcGIS Pro utilisé pour générer le fichier MMPK ;
- **displayFieldName** : champ dont les valeurs sont présentées dans la liste déroulante de localisation ;
- **sqlSearchExpression** : requête SQL utilisée pour construire la liste déroulante de localisation ;
- **Format** : type du champ utilisé pour la recherche.

5.2 TRAITEMENT DES DONNÉES

5.2.1 Fonds de plan tuilé

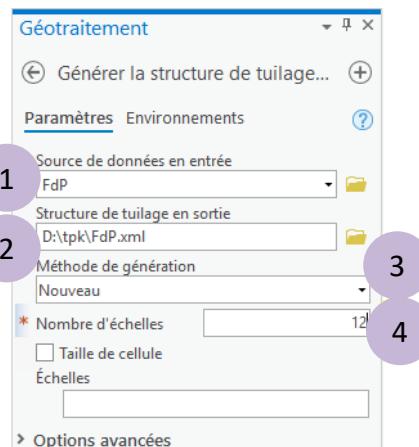
Chaque fond de plan est produit à partir d'une carte ArcGIS Pro en utilisant des outils de géotraitement proposés par l'application.

La fabrication comporte les opérations suivantes :

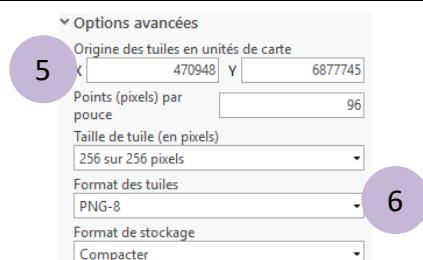
- Création d'un fichier de structure de tuiles ;
 - Création d'un paquetage de tuiles de carte.
- ! **Dans le cas où le paquetage de tuiles est créé à partir d'une source de données vectorielles stockées dans une géodatabase d'entreprise, la connexion aux couches à utiliser doit être réalisée avec le propriétaire de la données.**

5.2.1.1 Création d'un fichier de structure de tuiles

A partir de la fenêtre « Géotraitements » d'ArcGIS Pro, ouvrir l'outil « Générer la structure de tuilage du cache de tuiles ».



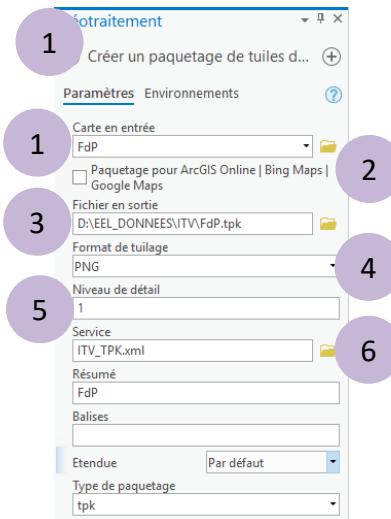
1	Indiquer la source de données à utiliser pour générer la structure de tuilage. En général, il s'agit d'une carte du projet ArcGIS Pro.
2	Indiquer le nom et l'emplacement du fichier qui va contenir la description de la structure de tuilage.
3	Indiquer la méthode de génération de la structure de tuilage.
4	Indiquer le nombre de niveaux d'échelle à créer dans la structure de tuilage. La liste des échelles est affichée suite à la saisie d'une valeur.



5	Indiquer l'origine de la structure de tuilage, point haut gauche (facultatif).
6	Indiquer le format de fichier des tuiles dans le cache (facultatif).

5.2.1.2 Création du paquetage de tuiles de carte

A partir de la fenêtre « Géotraitements » d'ArcGIS Pro, ouvrir l'outil « Crée un paquetage de tuiles de carte ».



1	Indiquer la source de données à utiliser pour générer la structure de tuilage. En général, il s'agit d'une carte du projet ArcGIS Pro.
2	Décocher l'option.
3	Indiquer le nom et l'emplacement du fichier qui va contenir le paquetage de tuiles de carte.
4	Indiquer le format image des tuiles qui vont être générées.
5	Indiquer le nombre de niveaux d'échelle attendu (déclaré dans la structure de tuilages de cartes).
6	Indiquer le fichier de structure de tuiles utilisé pour générer le paquetage.

Les autres options sont facultatives.

Le fichier .TPK généré par cette opération doit être copié sur la tablette dans le répertoire « **/Android/data/fr.onespatial.aopinterventionmobile/files/data** », en respectant le chemin relatif indiqué dans le fichier « **map.json** ».

5.2.2 Données vectorielles

5.2.2.1 Principe

Le paquetage de cartes mobiles utilisé par l'application est produit à partir d'une carte ArcGIS Pro en utilisant des outils de géotraitement proposés par l'application.

Pour un bon fonctionnement de l'application, il est nécessaire au préalable de réaliser les paramétrages suivants :

- Création des classes de relations dans le cas d'un réseau hiérarchique ;
- Définition des propriétés des couches dans la carte utilisée pour la génération du fichier MMPK.

5.2.2.2 Création des classes de relations

Une classe de relations est nécessaire, en consultation, pour accéder aux équipements « fils » de l'équipement consultée.

Deux méthodes sont disponibles pour la créations des classes de relations :

- A partir d'un outil de géotraitements du Catalogue ArcGIS Pro ;
- A partir d'un script Python.

L'utilisation du script Python permet de créer l'ensemble des classes de relations en une fois pour tous les équipements concernés du patrimoine.

5.2.2.2.1 Catalogue

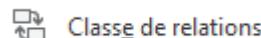
Pour la bonne compréhension du mécanisme de création d'une classe de relation, nous prenons pour exemple les équipements d'un réseau d'éclairage public.

Dans ce cas de figure, une armoire alimente plusieurs points lumineux.

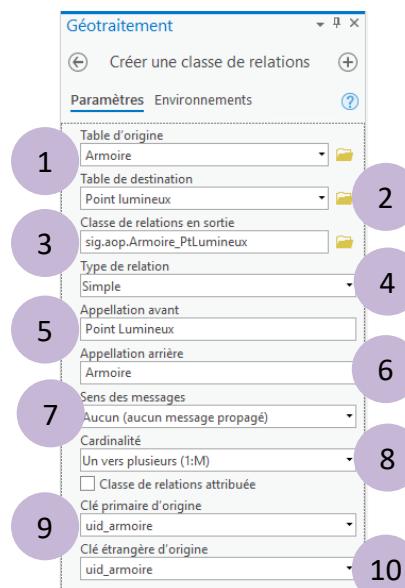
Armoire
AA
Point lumineux
AA-01
AA-02

A partir de la géodatabase :

- Clic droit pour accéder au menu contextuel ;
- Option « Nouveau » ;
- Sous option « Classe de relations ».



- La fenêtre suivante apparaît :



1	Indiquer le nom de la table d'origine.
2	Indiquer la table de destination.
3	Indiquer le nom de la classe de relations à créer.
4	Indiquer le type de relation entre les tables.
5	Indiquer l'alias de la couche destination.
6	Indiquer l'alias de la couche origine.
7	Indiquer le sens du message entre les couches (sans objet).
8	Indiquer le nombre de relation entre l'origine et la destination.
9	Indiquer le nombre de relation entre l'origine et la destination.
10	Indiquer le champ de la table de relation en lien avec la clé primaire de la table d'origine.

5.2.2.2 Script Python

Depuis le répertoire

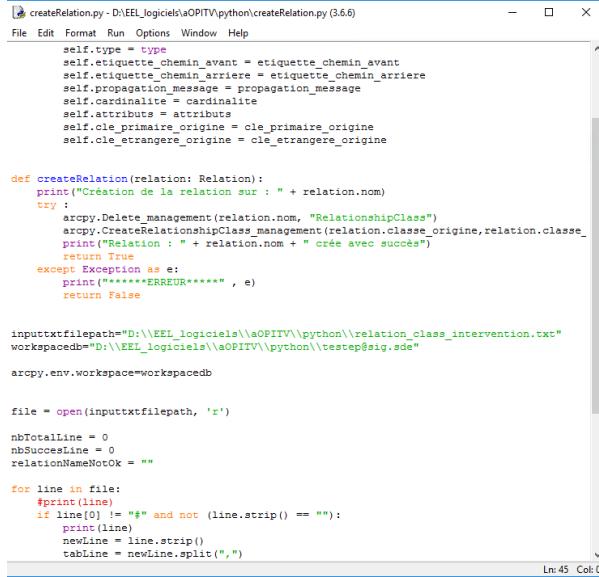
« ..\aOpServeur\Ressources_Installation\Intervention\installation\aoopmobile\python » :

- copier le répertoires « python » vers un répertoire de travail ;

Ce répertoire contient des scripts Python facilitant le paramétrage de l'application Back Office et de l'application mobile.

Ces script utilisent Python 3 disponible à partir d'ArcGIS Pro.

- A partir du répertoire « python », sélectionner le fichier « **createRelation.py** » puis ouvrir le menu contextuel associé.
- Cliquer sur l'option « *Edit with IDLE (ArcGIS Pro)* », la fenêtre suivante apparaît :



```
#!/usr/bin/python -u
# createRelation.py - D:\EEL_logiciels\aoPITV\python\createRelation.py (3.6.6)
File Edit Format Run Options Window Help
self.type = type
self.etiquette_chemin_avant = etiquette_chemin_avant
self.etiquette_chemin_arriere = etiquette_chemin_arriere
self.propagation_message = propagation_message
self.cardinalite = cardinalite
self.attributs = attributs
self.cle_primaire_origine = cle_primaire_origine
self.cle_etrangere_origine = cle_etrangere_origine

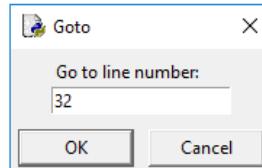
def createRelation(relation: Relation):
    print("Création de la relation sur : " + relation.nom)
    try :
        arcpy.Delete_management(relation.nom, "RelationshipClass")
        arcpy.CreateRelationshipClass_management(relation.classe_origine, relation.classe_destination, "Relation : " + relation.nom + " crée avec succès")
        return True
    except Exception as e:
        print("*****ERREUR*****", e)
        return False

inputtxtfilepath="D:\\EEL_logiciels\\aoPITV\\python\\relation_class_intervention.txt"
workspacedb="D:\\EEL_logiciels\\aoPITV\\python\\testep@sig.sde"
arcpy.env.workspace=workspacedb

file = open(inputtxtfilepath, 'r')
nbTotalLine = 0
nbSuccessLine = 0
relationNameNotOk = ""

for line in file:
    #print(line)
    if line[0] != "#" and not (line.strip() == ""):
        print(line)
        newLine = line.strip()
        tabLine = newLine.split(",")
Ln:45 Col:0
```

- Se positionner sur la ligne 32 du script en utilisant l'option « Go to Line » du menu « Edit ».



- Renseigner le paramètre suivant :

- **inputtxtfilepath** : chemin d'accès au fichier de description des classes de relations à créer.

```
inputtxtfilepath="D:\\EEL_logiciels\\aoPITV\\python\\relation_class_intervention.txt"
```

! Le caractère « \ » doit être doublé.

Le fichier de descriptions des classes de relations comporte une ligne par classe de relations à créer.

Chaque ligne comporte les informations suivantes :

- **nom** : nom de la classe de relations - **Armoire_PointLumineux**,
- **classe_origine** : nom de la couche origine en base de données - **lum_armoire**,
- **classe_destination** : nom de la couche destination en base de données - **lum_ptlum**,
- **type** : type de relation – **SIMPLE**,
- **etiquette_chemin_avant** : alias de la couche destination - **Point Lumineux**,
- **etiquette_chemin_arriere** : alias de la couche origine – **Armoire**,
- **propagation_message** : sens du message entre les couches (sans objet) – **NONE**,
- **cardinalite** : nombre de relation entre l'origine et la destination - **ONE_TO_MANY**,
- **attributs** : sans objet – **NONE**,
- **cle_primaire_origine** : champ de la table d'origine en lien avec la clé étrangère de la table de classe – **uid_armoire**,
- **cle_etrangere_origine** : champ de la table de relation en lien avec la clé primaire de la table d'origine – **uid_armoire**.

```
# Relation armoire
Armoire_PointLumineux,lum_armoire,lum_ptlum,SIMPLE,Point Lumineux,Armoire,NONE,ONE_TO_MANY,NONE,uid_armoire,uid_armoire
```

- **workspacedb** : fichier de connexion à la base de données contenant les couches à mettre en relation.

```
workspacedb="D:\\EEL_logiciels\\aOPITV\\python\\testep@sig.sde"
```

- ! Le caractère « \ » doit être doublé.**
- ! Pour une gestion des droits simplifiée, utiliser la connexion propriétaire des couches et tables concernées.**

- Exécuter le script à partir de l'option « Run Module » du menu « Run » ou appuyer sur la touche « F5 ».

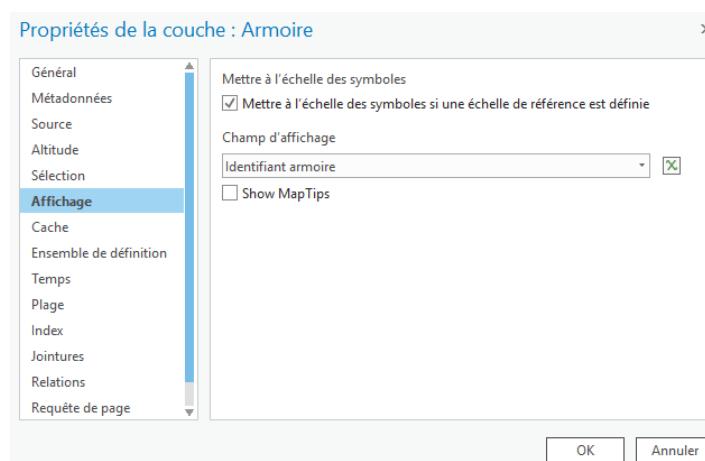
5.2.2.3 Définition des propriétés des couches

5.2.2.3.1 Champ d'affichage

Le champ d'affichage est le champ de l'équipement qui s'affiche dans la liste des équipements sélectionnés sur la carte.



Ce champ est défini à partir des propriétés de la couche dans la carte définie à partir d'ArcGIS Pro.



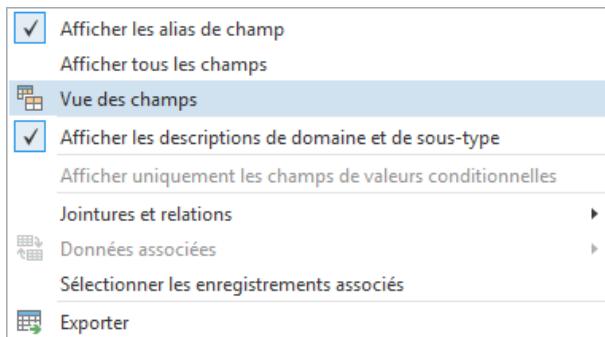
5.2.2.3.2 Champs consultables

La liste des champs consultables pour une couche est définie dans le fichier « **map.json** » à partir de la liste « **fields** ».

Il existe une autre manière de définir la liste des champs consultables. Le paramétrage est réalisé à partir de la table attributaire de la couche.

- Après avoir ouvert la table attributaire de la couche, cliquer sur le bouton ;

- A partir du menu qui s'affiche, sélectionner l'option « Vue des champs » :



- Un onglet s'ouvre, contenant la liste des champs de la couche avec leurs propriétés, dont la visibilité :

Couche actuelle Armoire				
	Visible	Lecture seule	Nom de champ Alias	
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	objectid	gid
	<input type="checkbox"/>	<input type="checkbox"/>	x	x
	<input type="checkbox"/>	<input type="checkbox"/>	y	y
	<input checked="" type="checkbox"/>	<input type="checkbox"/>	id_armoir	Identifiant armoire
	<input checked="" type="checkbox"/>	<input type="checkbox"/>	lib_armoi	Libellé armoire
	<input checked="" type="checkbox"/>	<input type="checkbox"/>	commune	Commune
	<input type="checkbox"/>	<input type="checkbox"/>	rue	rue

La colonne « Visible » détermine, pour chaque champ, si celui-ci est visible ou pas à partir de la table attributaire de la couche ou bien à partir de la fenêtre contextuelle de la couche.

- Cocher tous les champs à consulter dans l'application ;



- Cliquer sur le bouton **Enregistrer** pour enregistrer la configuration.

- ! Le champ déclaré comme « uidfield » dans l'application BackOffice doit être visible.
! Le champ utilisé pour la localisation sur les équipements de la couche doit être visible.

5.2.2.4 Génération du fichier MMPK

Cette procédure permet de générer les fichiers MMPK exploités par l'application mobile à partir d'une ou plusieurs cartes (une par réseau) définies dans un projet ArcGIS Pro mais également les fichiers « **infosmmpk.json** » qui contiennent les informations nécessaires à la mise à jour du patrimoine à partir de l'application mobile.

- ! La carte utilisée pour fabriquer le paquetage ne doit pas référencer des contenus en ligne.

Ces fichiers contiennent les informations suivantes :

- « **urlServer** » : url d'accès au répertoire contenant les fichiers MMPK ;
- « **mmpkFileName** » : nom du fichier MMPK – correspond au nom de la carte associée au réseau dans le projet ArcGIS Pro ;
- « **createdDate** » : date de génération du fichier MMPK – calculée ;
- « **size** » : taille du fichier MMPK généré – calculée.

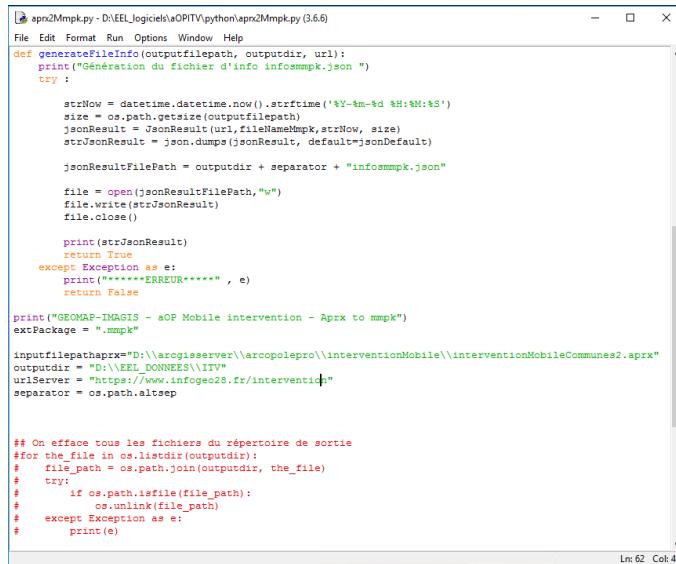
Depuis le répertoire :

« ..\aOpServer\Ressources_Installation\Intervention\installation\aoopmobile\python », copier le répertoires « python » vers un répertoire de travail.

Ce répertoire contient des scripts Python facilitant le paramétrage de l'application Back Office et de l'application mobile.

Ces scripts utilisent Python 3 disponible à partir d'ArcGIS Pro.

- A partir du répertoire « python », sélectionner le fichier « **apr2Mmpk.py** » puis ouvrir le menu contextuel associé ;
- Cliquer sur l'option « **Edit with IDLE (ArcGIS Pro)** », la fenêtre suivante apparaît :



The screenshot shows the ArcGIS Pro Python Editor window. The title bar reads "apr2Mmpk.py - D:\EEL_logiciels\aoPTIV\python\apr2Mmpk.py (3.6.6)". The menu bar includes File, Edit, Format, Run, Options, Window, Help. The code editor contains the following Python script:

```
apr2Mmpk.py - D:\EEL_logiciels\aoPTIV\python\apr2Mmpk.py (3.6.6)
File Edit Format Run Options Window Help
def generateFileInfo(outputfilepath, outputdir, url):
    print("Génération du fichier d'info infoMmpk.json")
    try:
        strNow = datetime.datetime.now().strftime('%Y-%m-%d %H:%M:%S')
        size = os.path.getsize(outputfilepath)
        jsonResult = JsonResponse(url, fileNameMmpk, strNow, size)
        strJsonResult = json.dumps(jsonResult, default=jsonDefault)

        jsonResultFilePath = outputdir + separator + "infoMmpk.json"

        file = open(jsonResultFilePath, "w")
        file.write(strJsonResult)
        file.close()

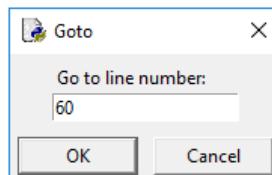
        print(strJsonResult)
        return True
    except Exception as e:
        print("*****ERREUR*****", e)
        return False

print("GEOMAP-INAOIS - aOP Mobile intervention - Apxr to mmpk")
extPackage = ".mmpk"
inputfilepathaprx = "D:\\arcgisserver\\arcopolepro\\interventionMobile\\interventionMobileCommunes2.aprx"
outputdir = "D:\\EEL_DONNEES\\ITV"
urlServer = "https://www.infogeo28.fr/intervention"
separator = os.path.altsep

## On efface tous les fichiers du répertoire de sortie
#for the_file in os.listdir(outputdir):
#    file_path = os.path.join(outputdir, the_file)
#    try:
#        if os.path.isfile(file_path):
#            os.unlink(file_path)
#    except Exception as e:
#        print(e)
```

Ln: 62 Col: 49

- Se positionner sur la ligne 60 du script en utilisant l'option « Go to Line » du menu « Edit » :



- Renseigner les paramètres suivants :

- **inputfilepathaprx** : chemin d'accès au projet ArcGIS Pro contenant les carte exploitée avec l'application mobile sous la forme d'un fichier MMPK.
! Le caractère « \ » doit être doublé.
- **outputdir** : répertoire d'accueil des fichiers MMPK et « **infoMmpk.json** ».

`outputdir = "D:\\EEL_DONNEES\\ITV"`

- !** Le caractère « \ » doit être doublé.

- **urlServer** : url d'accès aux repertoires contenant les fichiers MMPK.

`urlServer = "https://www.infogeo28.fr/intervention"`

! Le caractère « \ » doit être doublé.

► Exécuter le script à partir de l'option « Run Module » du menu « Run » ou appuyer sur la touche « F5 » ;

Le script produit un répertoire pour chaque carte contenue dans le projet ArcGIS Pro.

Ce répertoire contient :

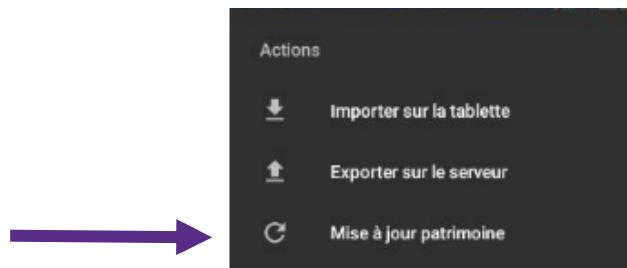
- Le fichier MMPK correspondant à la carte ;
- Le fichier « **infosmmpk.json** » associé au fichier MMPK.

Le répertoire porte le nom de la carte. Le fichier MMPK porte le nom de la carte.

Le répertoire ainsi généré et son contenu doivent être copiés sur la tablette dans le répertoire « **/Android/data/fr.onespatial/aopinterventionmobile/files/data** ».

6 MISE À JOUR DU PATRIMOINE

6.1 PRINCIPES



Cette fonction vérifie tout d'abord si une mise jour plus récente est disponible sur le serveur en consultant le fichier « `infosmmpk.json` » associé au fichier MMPK du réseau.

Dans le cas où une version plus récente est disponible, le fichier MMPK correspondant est téléchargée sur la tablette.

Un traitement batch, sur le serveur, permet de produire les fichiers nécessaires à la mise à jour du patrimoine pour chaque réseau exploité.

Ce traitement est lancé à partir d'une tâche planifiée.

La fréquence de mise à jour est donc adaptable en fonction de la politique de chaque structure.

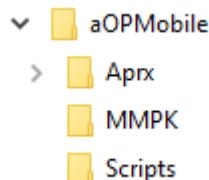
6.2 MISE EN ŒUVRE

6.2.1 Organisation

Créer le répertoire « `aOPMobile` » sous le répertoire « `1Spatial` ».

Dans le répertoire « `aOPMobile` » créer les répertoires suivants :

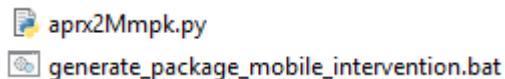
- « ***Aprx*** » : contient le projet ArcGIS Pro nécessaire à la génération des fichiers du patrimoine ;
- « ***MMPK*** » : répertoire de stockages des fichiers du patrimoine générés ;
- « ***Scripts*** » : contient les scripts nécessaires à la génération des fichiers du patrimoine :



Copier les fichiers suivants :

- « ***aprx2Mmpk.py*** » ;
- « ***generate_package_mobile_intervention.bat*** ».

Du répertoire « ..\aOpServeur\Ressources_Installation\Intervention\installation\aoopmobile\python » vers le répertoire « Scripts » :



6.2.2 Paramétrage accès au répertoire contenant le patrimoine

6.2.2.1 Contexte Tomcat

Copier le fichier « **arcopolepromobile.xml** » contenu dans le répertoire :

« ..\aOpServeur\Ressources_Installation\Intervention\installation\aoopmobile\aoOpServeur\server\apache-tomcat-9\conf\Catalina\localhost »,

Dans le répertoire : « ..\1Spatial\aoOpServeur\server\apache-tomcat-9\conf\Catalina\localhost ».

```
<?xml version="1.0" encoding="UTF-8"?>
<Context docBase="D:\geomap_imagis\aoPMobile\MMPK" path="/arcopolepromobile">
</Context>
```

Modifier la balise « docBase » si nécessaire.

Arrêter puis redémarrer le service « arcOpole PRO Serveur ».

6.2.2.2 Redirection Apache

Ouvrir le fichier «..\Apache24\conf\extra\ httpd-ssl.conf ».

Ajouter les lignes :

```
ProxyPass /arcopolepromobile ajp://localhost:8029/arcopolepromobile ttl=60 timeout=120
ProxyPassReverse /arcopolepromobile ajp://localhost:8029/arcopolepromobile
```

Arrêter puis redémarrer le service « Apache 2.4 ».

6.2.3 Configuration du script

6.2.3.1 Configuration du script

Ouvrir le fichier : « ..\1Spatial\aoPMobile\Scripts\ generate_package_mobile_intervention.bat ».

Renseigner les variables suivantes :

- « **pathpythonexe** » : exécutable Python pour ArcGIS Pro ;
- « **python** » : script Python de génération des fichiers du patrimoine ;
- « **inputpathfileaprx** » : fichier projet ArcGIS Pro utilisé pour la génération des fichiers ;
- « **outdirpath** » : répertoire d'accueil des fichiers du « patrimoine générés » ;
- « **urlServer** » : url d'accès aux fichiers du patrimoine.

```
set pathpythonexe="C:\Program Files\ArcGIS\Pro\bin\Python\envs\arcgispro-py3\python.exe"
set Python="D:\Geomap-Imagis\aoPMobile\Scripts\aprx2Mmpk.py"
set inputpathfileaprx="D:\Geomap-Imagis\aoPMobile\Aprx\interventionMobile.aprx"
set outdirpath="D:\Geomap-Imagis\aoPMobile\MMPK"
set urlServer="https://www.infogeo28.fr/arcopolepromobile"

%pathpythonexe% %python% %inputpathfileaprx% %outdirpath% %urlServer%
```

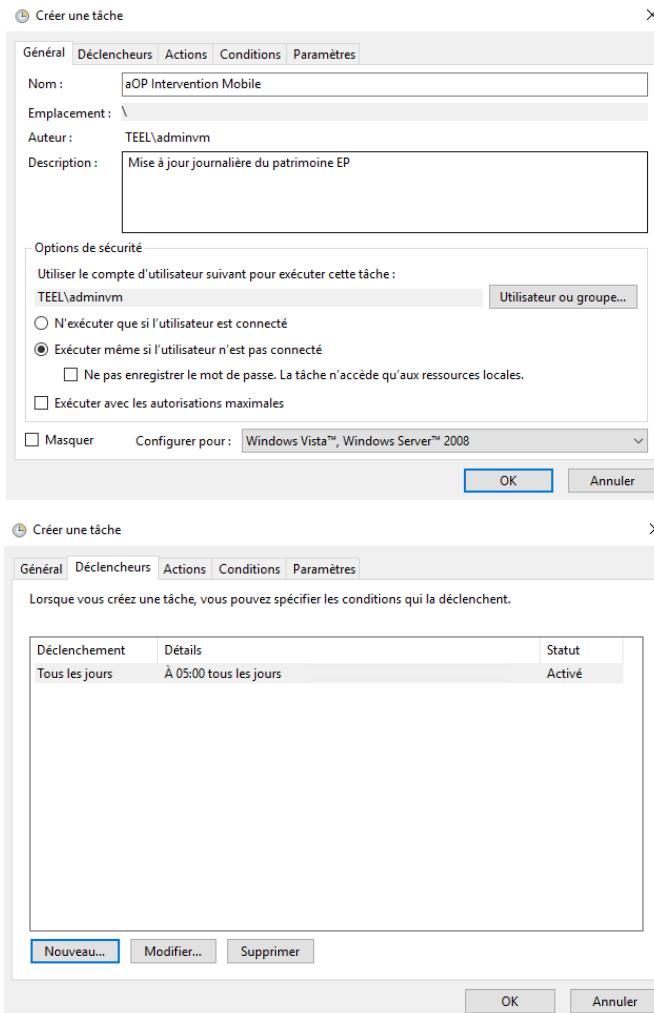
Pour tester le script, ouvrir l'invite de commande DOS :

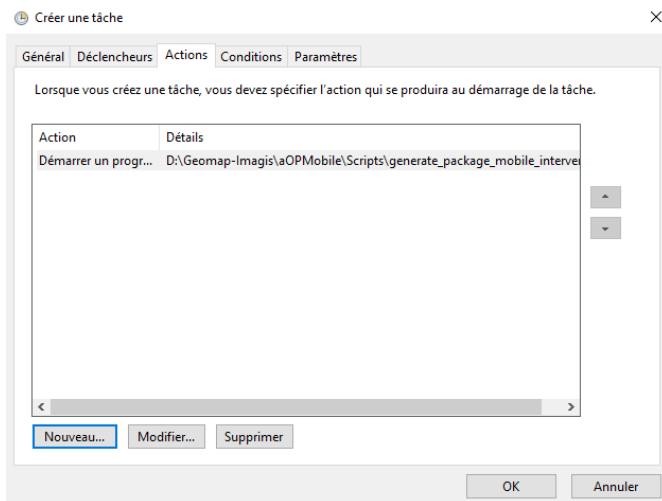
```
C:\1Spatial\arcOpolePRO\aoPServeur\v137\aoPMobile\Scripts>
C:\1Spatial\arcOpolePRO\aoPServeur\v137\aoPMobile\Scripts>generate_package_mobile_intervention.bat
```

6.2.4 Définition de la tâche planifiée

Ouvrir le planificateur de tâches.

A partir du menu contextuel de la Bibliothèque du Planificateur de tâches, choisir l'option « Crée une tâche... » :





Tester l'exécution de la tâche planifiée.

7 ANNEXES

7.1 DÉCLARATION CERTIFICAT SSL

arcOpole PRO Serveur est accessible à travers une connexion https qui nécessitant la déclaration d'un certificat SSL auprès du serveur Apache en frontal.

La déclaration du certificat SSL doit être complète pour assurer un bon fonctionnement d'arcOpole PRO Intervention Mobile :

- ▶ Les éléments suivants doivent être déclarés dans le fichier « ..\Apache24\conf\extra\httpd-ssl.conf » : Certificat serveur ;
- ▶ Clé privé ;
- ▶ Autorité de certification.

7.1.1 Certificat serveur

```
SSLCertificateFile "E:/geomap-imagis/Apache24/conf/ssl/lspatial.crt"
```

7.1.2 Clé privé

```
SSLCertificateKeyFile "E:/geomap-imagis/Apache24/conf/ssl/lspatial.key"
```

7.1.3 Autorité de certification

```
SSLCACertificateFile "E:/geomap-imagis/Apache24/conf/ssl/Intermediate-lspatial.cer"
```

7.2 PRINCIPE DE CONNEXION

La licence ArcGIS a une durée de validité d'un mois renouvelable par connexion.

Au lancement de l'application, et dans le cas où la licence ESRI a expiré , une connexion est proposée systématiquement à l'utilisateur pour renouveler la licence, suivie d'un import effectué après la connexion. Ce comportement est identique à celui d'une application nouvellement installée.

Si la connexion est refusée, l'utilisateur peut toujours se connecter pour renouveler sa licence en faisant manuellement un import.

Dans tous les cas, les interventions existantes sur la tablette sont perdues.