

# Kofax RPA

Guide des meilleures pratiques pour la gestion du cycle de vie du robot

Version : 11.2.0

Date : 2021-07-09

The logo for KOFAX, consisting of the word "KOFAX" in a bold, blue, sans-serif font.

© 2019–2021 Kofax. All rights reserved.

Kofax is a trademark of Kofax, Inc., registered in the U.S. and/or other countries. All other trademarks are the property of their respective owners. No part of this publication may be reproduced, stored, or transmitted in any form without the prior written permission of Kofax.

# Table des matières

<b>Préface.....</b>	<b>4</b>
Emplacement de la documentation.....	4
Formation.....	5
Obtenir de l'aide sur les produits Kofax.....	5
<b>Chapitre 1 : Gestion du cycle de vie du robot.....</b>	<b>7</b>
Configuration de base sans la gestion du cycle de vie du robot.....	7
Configuration de base avec la gestion du cycle de vie du robot.....	8
<b>Chapitre 2 : Configuration de base avec la gestion du cycle de vie du robot.....</b>	<b>10</b>
Choisir une stratégie de ramification.....	10
Stratégie de ramification recommandée.....	10
Autres stratégies de ramification.....	12
Créer un répertoire nu.....	14
Configurer les Management Consoles.....	15
Mettre en place le développement Management Console.....	15
Mettre en place la production Management Console.....	16
Démarrer la synchronisation.....	17
Promouvoir et inverser les changements.....	20
Vérifier le résultat de la synchronisation.....	20
<b>Chapitre 3 : Droits d'accès et conditions préalables.....</b>	<b>22</b>

# Préface

Ce guide propose des méthodes et des techniques recommandées pour vous aider à optimiser vos performances et à garantir le succès de votre utilisation du Kofax RPA.

## Emplacement de la documentation

Le jeu de documents de Kofax RPA est disponible ici : <sup>1</sup>

[https://docshield.kofax.com/Portal/Products/RPA/11.2.0\\_ea1ydbmwk9/RPA.htm](https://docshield.kofax.com/Portal/Products/RPA/11.2.0_ea1ydbmwk9/RPA.htm)

Outre ce guide, le jeu de documents comprend les éléments suivants :

### ***Notes de mise à jour Kofax RPA***

Contient des informations de dernière minute et d'autres informations qui ne sont pas disponibles dans l'autre documentation Kofax RPA dont vous disposez.

### ***Spécifications techniques de Kofax RPA***

Contient des informations sur les systèmes d'exploitation pris en charge et les autres exigences du système

### ***Guide d'installation de Kofax RPA***

Contient des instructions sur l'installation de Kofax RPA et de ses composants dans un environnement de développement

### ***Guide de mise à niveau de Kofax RPA***

Contient les instructions de mise à niveau de Kofax RPA et de ses composants vers une nouvelle version.

### ***Guide de l'administrateur de Kofax RPA***

Décrit les tâches administration et de gestion dans Kofax RPA.

### ***Aide de Kofax RPA***

Explique comment utiliser Kofax RPA. L'aide est également disponible sous la forme d'un fichier PDF appelé *Guide de l'utilisateur de Kofax RPA*.

### ***Kofax RPA Guide de mise en route pour la création de robots***

Fournit un tutoriel qui explique l'utilisation de Kofax RPA pour créer un robot.

---

<sup>1</sup> vous devez être connecté à Internet pour pouvoir accéder en ligne au jeu complet des documents.

### **Guide Kofax RPA de démarrage avec Document Transformation**

Fournit un tutoriel qui explique comment utiliser la fonctionnalité Document Transformation dans un environnement Kofax RPA, y compris OCR, l'extraction, le formatage des champs et la validation

### **Guide de configuration du Desktop Automation Service de Kofax RPA**

Explique comment configurer le Desktop Automation Service nécessaire pour utiliser Desktop Automation sur un ordinateur distant

### **Guide du développeur Kofax RPA**

Contient des informations sur l'API qui est utilisée pour exécuter les robots sur RoboServer.

### **Documentation sur l'API d'intégration Kofax RPA**

Contient des informations sur l'API Java Kofax RPA et l'API .NET Kofax RPA qui fournissent un accès programmatique au produit Kofax RPA. La documentation de l'API Java est disponible en ligne et hors ligne à l'adresse Kofax RPA, tandis que la documentation de l'API .NET n'est disponible que hors ligne.

**Remarque** Les API de Kofax RPA comprennent de nombreuses références à RoboSuite, le nom original du produit. Le nom de RoboSuite est conservé dans les API pour assurer la rétrocompatibilité. Dans le contexte de la documentation des API, le terme RoboSuite est équivalent à Kofax RPA.

## Formation

Kofax propose des formations en classe et sur ordinateur pour vous aider à tirer le meilleur parti de votre solution Kofax RPA. Viz le portail de formation Kofax à l'adresse suivante : <https://learn.kofax.com/> pour obtenir des détails sur les formations et les planifications disponibles.

Vous pouvez également consulter le Kofax Intelligent Automation SmartHub à l'adresse suivante : <https://smarthub.kofax.com/> pour explorer d'autres solutions, robots, connecteurs, etc.

## Obtenir de l'aide sur les produits Kofax

Le répertoire de la [Base de connaissances Kofax](#) contient des articles qui sont régulièrement mis à jour pour vous tenir informé des produits Kofax. Nous vous encourageons à utiliser cette base de connaissances pour obtenir des réponses à vos questions sur les produits.

Pour accéder à la Kofax Knowledge Base [Base de connaissances Kofax], ouvrez le [Kofaxsite web](#) et sélectionnez **Assistance** sur la page d'accueil.

**Remarque** La Kofax Knowledge Base [Base de connaissances Kofax] est optimisée pour Google Chrome, Mozilla Firefox ou Microsoft Edge.

La Kofax Knowledge Base [Base de connaissances Kofax] propose :

- Puissantes fonctionnalités de recherche pour vous aider à localiser rapidement les informations dont vous avez besoin.

Saisissez vos termes ou votre phrase à rechercher dans le champ **Search** [Recherche], puis cliquez sur l'icône Loupe.

- Informations produit, détails de configuration et documentation, notamment les actualités des nouvelles versions.

Faites défiler la page d'accueil de la Kofax Knowledge Base [Base de connaissances Kofax] pour localiser une famille de produits. Cliquez ensuite sur le nom d'une famille de produits pour afficher une liste d'articles sélectionnés. Veuillez noter que certaines familles de produits nécessitent un identifiant valide du portail Kofax pour afficher ces articles sélectionnés.

- Accès au Kofax Customer Portal [Portail client Kofax] (pour les clients éligibles).

Cliquez sur le lien **Customer Support** [Assistance client] en haut de la page, puis cliquez sur **Log in to the Customer Portal** [Connexion au portail client].

- Accès au Kofax Partner Portal [Portail partenaire Kofax] (pour les partenaires éligibles).

Cliquez sur le lien **Partner Support** [Assistance partenaire] en haut de la page, puis cliquez sur **Log in to the Partner Portal** [Connexion au portail partenaire].

- Accès aux validations de l'assistance Kofax, aux politiques de cycle de vie, aux détails d'exécution électroniques et aux outils en libre-service.

Faites défiler la page jusqu'à la section **General Support** [Assistance générale], cliquez sur **Support Details** [Détails de l'assistance], puis sélectionnez l'onglet approprié.

## Chapitre 1

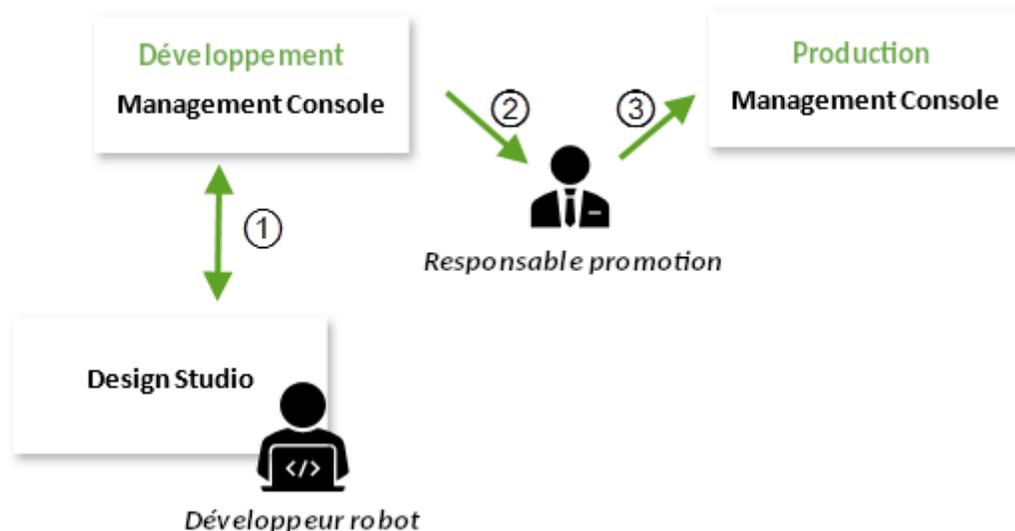
# Gestion du cycle de vie du robot

La fonction de gestion du cycle de vie du robot vous permet de contrôler des objets de travail de différents types dans un système de contrôle de version tel que Git. Grâce à cette fonctionnalité, vous pouvez comparer et synchroniser le statut des objets entre le Management Console et votre répertoire à l'aide du Kofax RPA Synchronizer inclus dans votre installation. Vous pouvez synchroniser les objets suivants : robots, types, snippets, ressources, planifications et OAuth.

Ces bonnes pratiques sont écrites en supposant que vous avez une connaissance pratique du système de contrôle de version Git.

## Configuration de base sans la gestion du cycle de vie du robot

Le schéma suivant illustre une configuration de base recommandée Kofax RPA sans gestion du cycle de vie du robot. Cette configuration se compose de deux Management Consoles : l'une pour le développement et l'autre pour la production.



Dans cet exemple de configuration, le développeur de robot établit la synchronisation (1) avec un projet stocké dans le développement Management Console, crée de nouveaux objets ou apporte des modifications aux objets existants, puis synchronise à nouveau le projet avec ce Management Console.

Lorsque le responsable de la promotion envisage un projet pour la production, il effectue une sauvegarde de ce projet **(2)** ou d'objets particuliers du développement Management Console et télécharge ensuite la sauvegarde **(3)** vers la production Management Console.

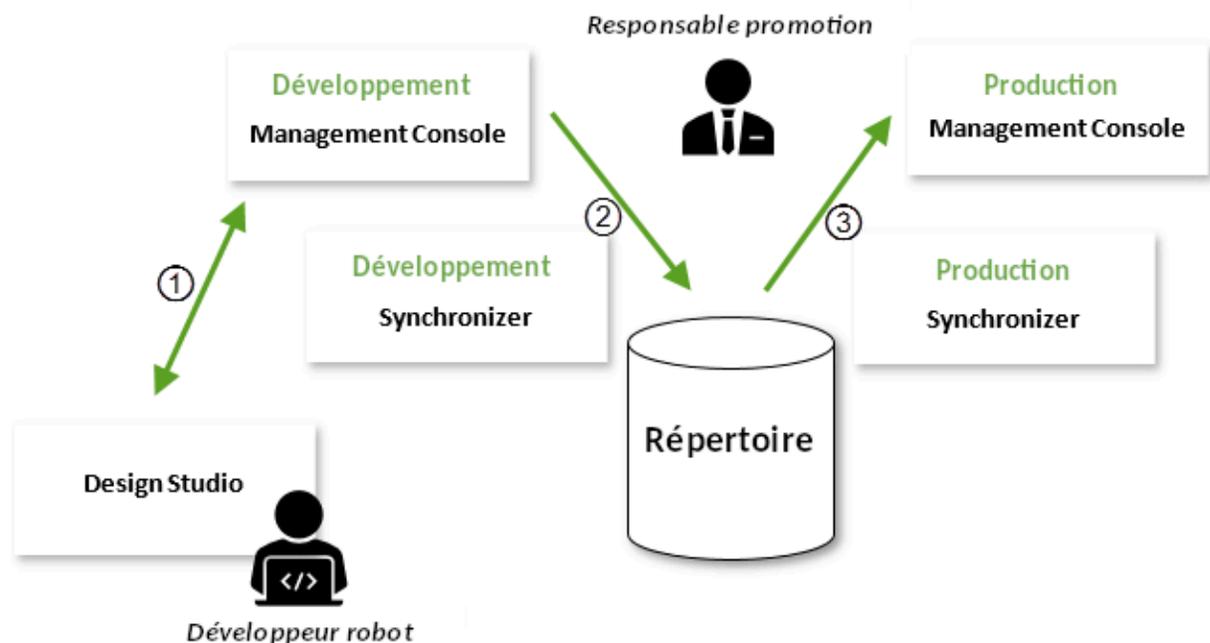
Sans configuration de la gestion du cycle de vie du robot, cette configuration ne prend pas en charge les fonctionnalités suivantes :

- Historique précis des versions pour chaque objet d'un projet.
- Possibilité de savoir quelle version est actuellement en production, de voir l'auteur d'une modification, la date de la dernière modification d'un objet et un message expliquant les modifications.
- Capacité à promouvoir la version spécifique d'un objet, du développement à la production.
- Possibilité de revenir rapidement à une version précédente de l'objet en production.
- Capacité à inverser les changements de rupture dans l'environnement de développement.

Pour inclure toutes ces fonctionnalités dans votre environnement Kofax RPA, vous pouvez définir une configuration de base avec la gestion du cycle de vie du robot, comme indiqué dans la rubrique suivante.

## Configuration de base avec la gestion du cycle de vie du robot

Le schéma suivant illustre une configuration de base Kofax RPA recommandée avec la gestion du cycle de vie du robot. Cette configuration montre deux instances de Management Console qui partagent un seul répertoire : une instance pour le développement et une instance pour la production.



Dans cet exemple, le développeur du robot établit une synchronisation **(1)** avec un projet stocké dans le développement Management Console.

Chaque fois qu'un développeur de robot met à jour le projet dans le développement Management Console, les changements sont automatiquement synchronisés **(2)** avec le répertoire partagé ; ils sont envoyés comme une validation de changement sur la branche spécifiée. La responsabilité du directeur de la production est d'indiquer la branche à utiliser dans la production.

À ce stade, deux méthodes sont possibles pour promouvoir le changement :

- Fusionner, refondre, choisir ou apporter des modifications à la branche spécifiée dans la production Management Console.
- Il suffit de spécifier la production Management Console à la nouvelle version.

Lorsque le responsable de la promotion approuve les changements, la production Synchronizer prend les changements **(3)** du répertoire et les pousse ensuite vers la production Management Console.

Dans les rubriques suivantes, nous allons recréer cette configuration étape par étape.

## Chapitre 2

# Configuration de base avec la gestion du cycle de vie du robot

Utilisez la procédure suivante pour recréer [la configuration de base avec la gestion du cycle de vie du robot](#). La procédure donne des détails étape par étape qui montrent comment synchroniser les entrées et sorties d'un simple répertoire Git nu, basé sur des fichiers, communément appelé répertoire distant.

Le Synchronizer permet à tous les types de répertoires Git d'être le répertoire distant, à condition qu'ils respectent les normes du protocole Git pour le transfert de données. La connexion à un répertoire distant par HTTP ne peut être établie que si le répertoire ne requiert aucune autorisation.

**Conseil** Vous pouvez consulter le matériel de référence en ligne de Git pour obtenir des informations détaillées sur l'utilisation de chaque type de répertoire. Sur le site [git-scm.com](http://git-scm.com), recherchez des articles sur les protocoles, tels que « Git sur le Server - Les protocoles ». Le site Synchronizer prend en charge les normes suivantes du protocole Git pour le transfert de données : HTTP(S), SSH, et Git. La principale différence est que, alors qu'un simple répertoire Git nu réside localement sur un système de fichiers, les autres solutions peuvent être connectées à distance. Ce choix a un impact sur les exigences physiques et la composition des conteneurs.

Pour effectuer une configuration de base qui inclut la gestion du cycle de vie du robot, suivez les étapes suivantes :

1. [Choisir une stratégie de ramification applicable](#)
2. [Créer un répertoire Git nu et basé sur des fichiers](#)
3. [Configurer les consoles de gestion](#)
4. [Démarrer les Synchronizers](#)
5. [Promouvoir les objets à la production et inverser les changements](#)
6. [Vérifier le résultat de la synchronisation](#)

## Choisir une stratégie de ramification

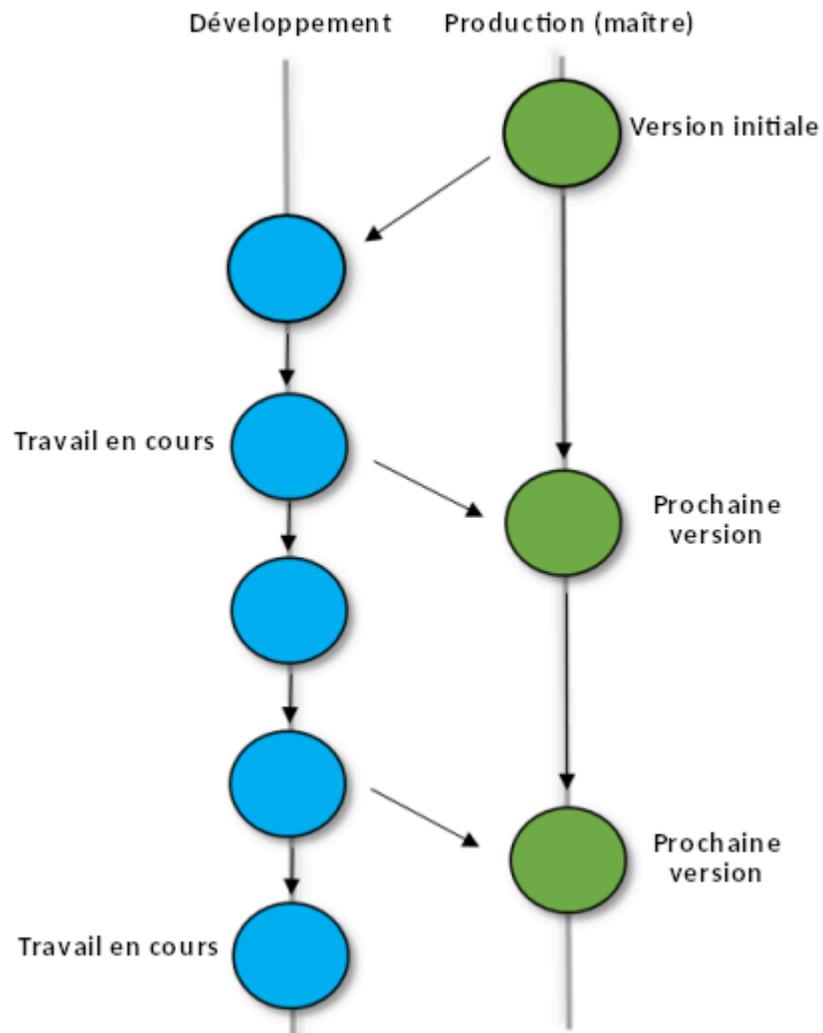
De nombreuses approches sont disponibles pour gérer vos objets de travail avec un répertoire Git. Nous vous recommandons de suivre la stratégie et les recommandations présentées dans la rubrique suivante.

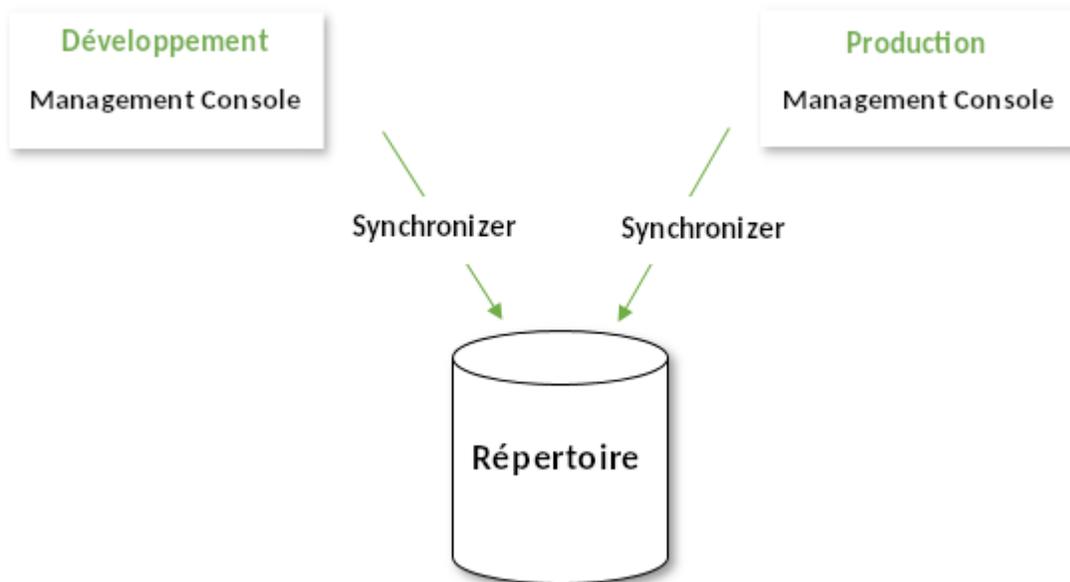
### Stratégie de ramification recommandée

Selon cette stratégie, le responsable de la branche principale contient toujours la version actuelle en cours de production. Vous pouvez ajouter une autre branche à la branche principale (production), par

exemple pour le développement, et faire synchroniser le développement Management Console avec cette branche.

Les chiffres suivants montrent la stratégie et la configuration minimales recommandées pour les ramifications. Les cercles bleus et verts sont des têtes de Git contenant les versions actuelles dans les branches du développement et de la production, respectivement.





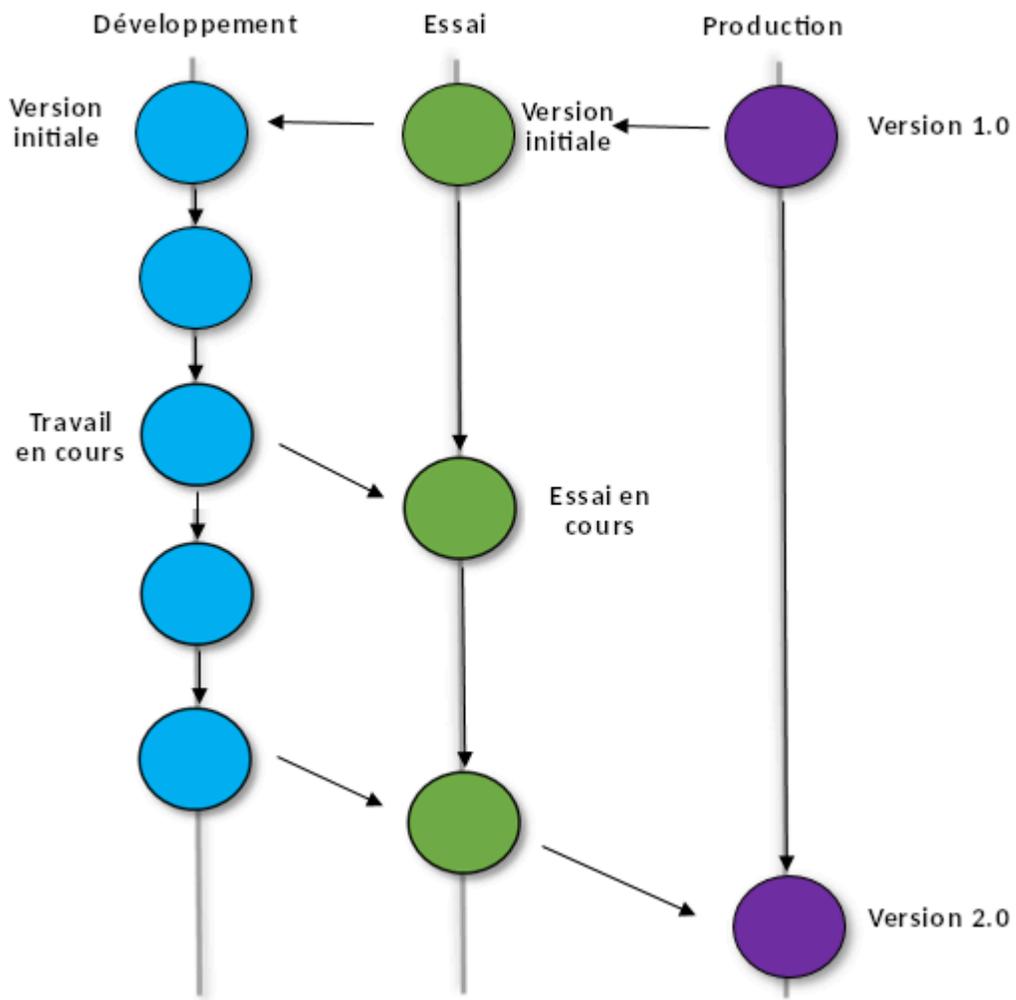
Recommandation 1 : Utilisez toujours la branche principale pour la production.

Recommandation 2 : Lorsque vous fusionnez avec la branche principale, utilisez le drapeau `--no-ff` qui empêche Git de s'exécuter en "avance rapide" s'il détecte que votre tête actuelle est un ancêtre de la validation que vous fusionnez. Il est utile d'avoir les validations de fusion sur votre branche de production pour suivre la date et l'heure exactes de la fusion.

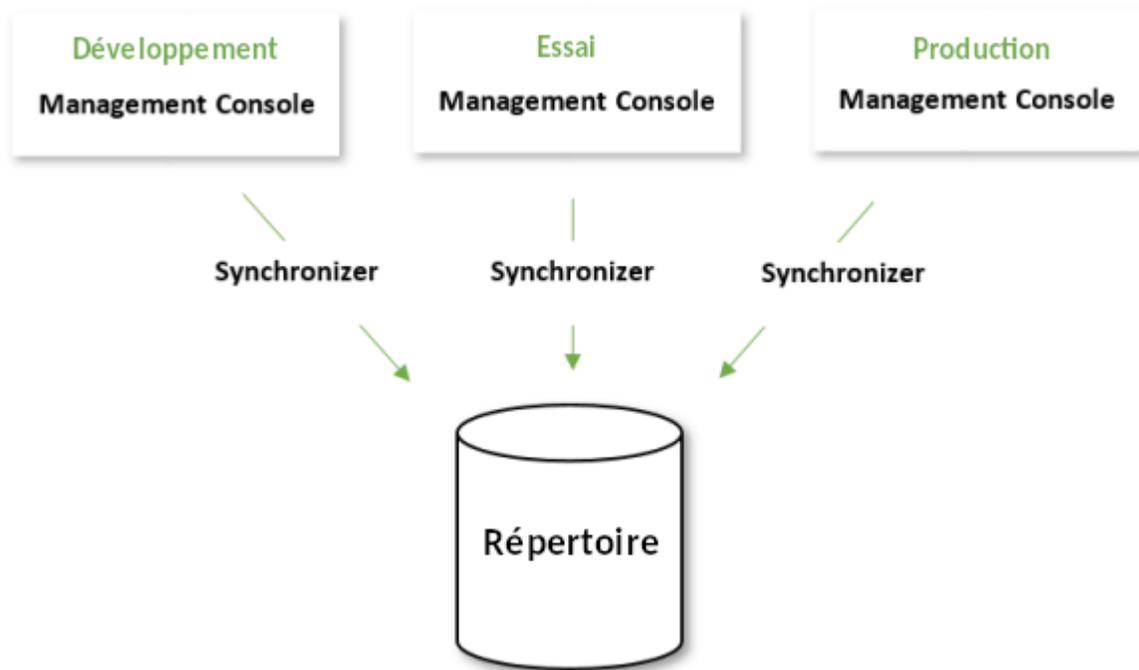
**Conseil** Pour en savoir plus sur la stratégie décrite ci-dessus, voir [Un modèle de ramification Git réussi](#).

## Autres stratégies de ramification

Si vous préférez une configuration plus complexe, la stratégie de ramification minimale présentée dans le thème précédent peut facilement être étendue pour couvrir une configuration plus large avec trois branches : développement, test et production.



Dans cette configuration, vous disposez de trois Management Consoles pour le développement, les tests et la production, synchronisées avec trois branches dans un répertoire Git via des Synchronizers.



## Créer un référitoire nu

Avant de démarrer les Synchronizers et de configurer les Management Consoles pour la synchronisation, vous devez initialiser votre référitoire Git. La méthode de création d'un nouveau référitoire nu dépend de l'outil tiers que vous utilisez.

Pour créer un référitoire nu et une branche de développement, vous pouvez exécuter les commandes suivantes.

```
# mkdir example.git
# cd example.git/
# git init --bare
Initialized empty Git repository in /gitrepos/example.git/
# cd ..
# git clone example.git
Cloning into 'example'...
warning: You appear to have cloned an empty repository.
done.
# git commit --allow-empty -m 'initial commit'
[master (root-commit) b96fdb8] initial commit
# git push origin
Everything up-to-date
# git checkout -b development
Switched to a new branch 'development'
# git push -u origin development:development
```

**Conseil** Vous pouvez également utiliser le Synchronizer pour créer automatiquement un répertoire nu. Lorsque vous démarrez le Synchronizer, il crée automatiquement un répertoire nu à l'endroit indiqué, s'il n'a pas déjà été créé, avec une validation initiale vide. Toutefois, le Synchronizer ne crée pas de branche si elle n'est pas déjà présente.

## Configurer les Management Consoles

Après avoir créé un répertoire, vous devez installer deux Management Consoles : l'une pour le développement et l'autre pour la production. Ils sont utilisés pour se synchroniser avec les branches de développement et de production (master) de votre répertoire de fichiers.

### Mettre en place le développement Management Console

1. Démarrer le développement Management Console.
2. Dans le menu, sélectionnez **Administration > Projets**. Pour synchroniser le projet avec le répertoire, cliquez sur le menu **⋮**, puis cliquez sur **Modifier**.
3. Dans la nouvelle boîte de dialogue, sélectionnez l'onglet **Répertoire**.
  - a. Dans la propriété de l'**URL**, saisissez le chemin d'accès au répertoire que vous avez créé [dans la rubrique précédente](#) : **/gitrepos/exemple.git/**
  - b. Dans la propriété **Branche**, saisissez la branche à utiliser : **développement**.
  - c. Pour activer la configuration spécifiée ci-dessus, sélectionnez Activer la **Configuration**.

- d. Sous **Objets à synchroniser**, sélectionnez les objets à inclure dans la synchronisation.  
**Robots, Types et Snippets**

Activer la configuration

URL \*  
/gitrepos/example.git

Branche \*  
development

Lecture seule

#### Objets à synchroniser

Planifications

Robots, types et snippets

Ressources

OAuth

- e. Enregistrez les modifications.

## Mettre en place la production Management Console

1. Démarrer la production Management Console.
2. Dans le menu, sélectionnez **Administration > Projets**. Pour synchroniser le projet avec le répertoire, cliquez sur le menu , puis cliquez sur **Modifier**.
3. Dans la nouvelle boîte de dialogue, sélectionnez l'onglet **Répertoire**.
  - a. Dans la propriété de l'**URL**, saisissez le chemin d'accès au répertoire que vous avez créé dans la [rubrique précédente](#) : **/gitrepos/exemple.git/**
  - b. Dans la propriété **Branche**, saisissez la branche à utiliser : **master**. Nous vous recommandons de toujours utiliser la branche principale pour la production.
  - c. Pour faire du répertoire la seule source de modification des objets, sélectionnez **Lecture seule**. Nous vous recommandons de sélectionner cette option pour éviter toute modification des objets appartenant au projet synchronisé dans la production Management Console.
  - d. Pour activer la configuration spécifiée ci-dessus, sélectionnez **Activer la configuration**.

- e. Sous **Objets à synchroniser**, sélectionnez les objets à inclure dans la synchronisation.  
**Robots, Types et Snippets**

Activer la configuration

URL \*  
/qitrepos/example.git

Branche \*  
master

Lecture seule

#### Objets à synchroniser

Planifications

Robots, types et snippets

Ressources

OAuth

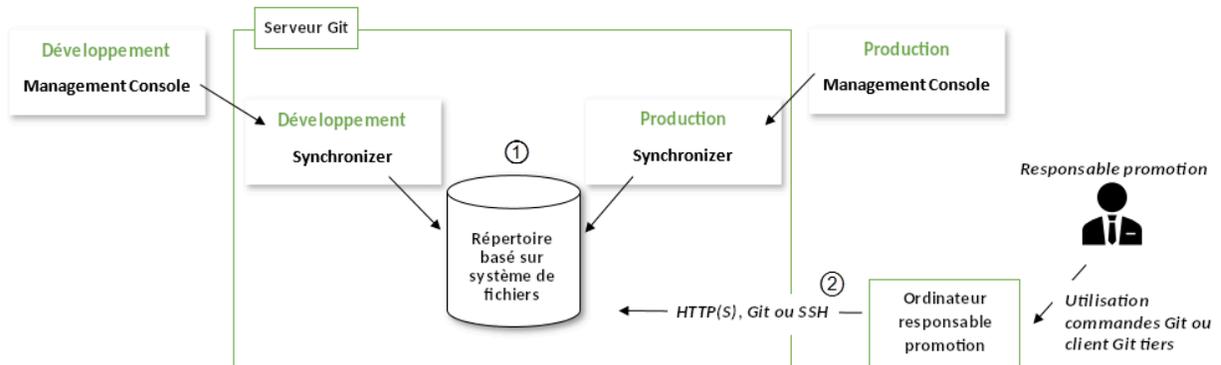
- f. Enregistrez les modifications.

Vous avez maintenant configuré vos Management Consoles et êtes prêt à lancer la synchronisation.

## Démarrer la synchronisation

Lorsque vous utilisez un répertoire Git basé sur des fichiers, les deux instances du Synchronizer, l'une pour le développement Management Console et l'autre pour la production Management Console, requièrent un accès direct au système de fichiers où se trouve le répertoire.

Nous vous recommandons d'exécuter les deux instances du Synchronizer sur le même ordinateur où le répertoire est stocké **(1)**. Grâce à cette configuration, le responsable de la promotion peut utiliser l'une des normes du protocole Git **(2)** pour accéder au répertoire et pousser les changements vers la production.



Recommandation 1 : Lors de la synchronisation avec un répertoire de fichiers nu, plusieurs Synchronizers peuvent fonctionner sur le même ordinateur.

Recommandation 2 : Pour plus de commodité, le gestionnaire de promotion peut utiliser un client Git tiers tel que SourceTree au lieu des commandes Git.

Recommandation 3 : Si Docker est utilisé pour le déploiement, les conteneurs Synchronizer peuvent partager un volume.

### Démarrer les Synchronizers

Pour relier les deux Management Consoles et le répertoire Git, procédez comme suit.

1. Dirigez le développement Synchronizer vers le développement Management Console en configurant les propriétés suivantes dans le fichier **synchronizer.settings**.

Dans la fenêtre d'invite de commande, indiquez d'abord la propriété `-c` du fichier **synchronizer.exe** pour qu'il ignore le fichier `synchronizer.settings` par défaut. Précisez ensuite les autres propriétés dont vous avez besoin.

#### Exemple

```
Synchronizer.exe -c --mc-url http://127.0.0.1:8080/ManagementConsole --username
admin --password pass --interval 10 --no-host-key false --private-key $USER_HOME
\.ssh\id_rsa -s
```

- `-c, --command-line` Utilise les paramètres spécifiés dans la ligne de commande et ignore le fichier de paramètres.
  - `-e, --environnement` Utilise les paramètres de l'environnement et ignore le fichier de paramètres.
  - `-g, --generate-ssh-keys<argument>` Génère une paire de clés pour l'authentification SSH et l'enregistre dans le dossier spécifié. Par exemple : `-g C:\Work\MyKeys`
  - `--mc_url<argument>` Requis. URL pour se connecter au développement ou à la "production" Management Console contenant le protocole et un numéro de port. Utilisé uniquement lorsque `-c` est spécifié.
  - `--username <argument>` Requis. Management Console utilisateur. Utilisé uniquement lorsque `-c` est spécifié.
  - `--password <argument>` Requis. Mot de passe pour l'utilisateur Management Console. Utilisé uniquement lorsque `-c` est spécifié.
  - `--interval <argument>` Requis. Intervalle en secondes entre les cycles de synchronisation. S'il est réglé sur une valeur égale ou inférieure à 0 ou sur une valeur non numérique, le Synchronizer fonctionne une fois et sort. Utilisé uniquement lorsque `-c` est spécifié.
  - `--no-host-key<argument>` Facultatif. Désactive le contrôle strict des clés de l'hôte en SSH. Par défaut, il est défini sur `faux`. Utilisé uniquement lorsque `-c` est spécifié.
  - `--clé privée<argument>` obligatoire. Chemin d'accès à un fichier contenant la clé privée SSH pour se connecter à un répertoire distant. Lors de la connexion à un répertoire local, cet attribut est ignoré, mais toute valeur doit être spécifiée. Utilisé uniquement lorsque `-c` est spécifié.
  - `-r, --reset-hard` Réinitialise les informations de version et purge l'ensemble du cache local.
  - `-s, --save` Enregistre les paramètres de configuration dans le fichier `synchronizer.settings` et quitte.
  - `-v, --version` Imprime les informations sur la version et les sorties.
  - `-h, --help` Imprime la description des propriétés et des sorties.
2. Répétez l'étape précédente pour la production Synchronizer et la production Management Console et modifiez les propriétés selon le cas.

Vous avez maintenant synchronisé les Management Consoles et le répertoire Git et vous êtes prêt à promouvoir les objets en production.

## Promouvoir et inverser les changements

Les robots, types et snippets peuvent désormais être synchronisés avec le développement Management Console à partir de Design Studio en utilisant la fonction de téléchargement ou à partir de l'interface Management Console en utilisant l'onglet Robots. À ce stade, vous pouvez apporter des modifications à la production Management Console uniquement en promouvant une version objet du développement à la production.

Utilisez la commande suivante pour fusionner de la branche de développement à la branche de production (principale).

```
# cd example
# git checkout development
# git pull
# git checkout master
# git merge --no-ff development
# git push
```

Si vous devez annuler des modifications, faites-le sur la branche de développement en rétablissant les validations à un état antérieur satisfaisant, puis en fusionnant les validations annulés à la branche de la production (principale).

## Vérifier le résultat de la synchronisation

Pour vérifier le résultat de la synchronisation, consultez le fichier journal correspondant, qui réside par défaut dans :home\AppData\Local\Kofax RPA\version\Logs. L'emplacement du fichier journal peut être configuré avec log4j2\_synchronizer.properties, qui réside dans :home\AppData\Local\Kofax RPA\version\Configuration.

Si la synchronisation est réussie, le journal contient le nom du projet synchronisé et une ID de validation de la synchronisation. La synchronisation s'exécute à l'intervalle spécifié, et les changements d'objets sont automatiquement synchronisés entre le Management Console et le répertoire dès que les changements apparaissent.

En cas de modifications contradictoires, la version de l'objet du répertoire aura la priorité sur les modifications du Management Console. Dans ce cas, les modifications conflictuelles de Management Console seront ignorées.

**Remarque** À ce point, il ne sera pas possible de renommer et de changer l'emplacement des fichiers synchronisés du répertoire parce que les références à ces fichiers dans le Management Console deviendront incorrectes.

Une synchronisation réussie comprend :

- une ID de validation. Désigne une validation des modifications de l'objet du répertoire synchronisé avec le Management Console.
- l'horodatage. Tous les objets modifiés dans le Management Console avant cette date sont synchronisés avec le répertoire.

Sur l'interface Management Console, pour voir l'auteur d'un objet nouvellement ajouté, le message de validation, la révision de l'objet et la date de la dernière modification, assurez-vous que les colonnes suivantes sont ajoutées à la vue respective : **Modifiée par**, respectivement, le **Message de validation**, le **Numéro de révision** et la **Dernière modification**.

## Chapitre 3

# Droits d'accès et conditions préalables

Par défaut, seuls les utilisateurs ayant le rôle **admin**, **Administrateur**, **Administrateurs de projet** ou **VCS Service User** peuvent utiliser la gestion du cycle de vie du robot.

Bien que tous ces rôles incluent le droit de démarrer le Synchronizer, **VCS Service User** est conçu spécifiquement à cette fin. L'utilisateur ayant ce rôle ne peut pas apporter de modifications à un Management Console, comme par exemple définir les paramètres de synchronisation d'un projet. Par conséquent, vous pouvez attribuer ce rôle à un utilisateur dont les responsabilités n'incluent pas la gestion de projet et concernent principalement la synchronisation avec un système de contrôle de version.

Pour permettre à un nouvel utilisateur d'accéder à la gestion du cycle de vie du robot, sur le Management Console, l'utilisateur doit être ajouté à un groupe qui a l'un de ces rôles. Pour plus d'informations sur les rôles des utilisateurs, voir « Gérer les utilisateurs et les groupes » dans l'*aide de Kofax RPA*.

Si le Synchronizer s'exécute sur un système de fichiers qui a des limitations spécifiques non standard sur les caractères et, par conséquent, peut appliquer certaines restrictions sur les noms d'objets, il peut causer des problèmes lors de la synchronisation de ces objets, même si le système de fichiers lui-même est pris en charge. Pour assurer une synchronisation réussie, vérifiez que votre système de fichiers autorise les caractères utilisés dans les noms d'objets synchronisés.