

GUIDES COMPARATIFS



**RELEASE  
AUTOMATION ET  
IMPLÉMENTATION  
DEVOPS**

# A PROPOS DE CE GUIDE

## Release automation, implémentation DevOps et déploiement automatique d'applications

### 1 UTILISER CE GUIDE

La structure et le contenu de ces guides constituent une excellente base pour préparer un cahier des charges ou un comparatif.

[En savoir plus](#)

### 2 DROITS D'USAGE

guidescomparatifs.com autorise toute personne physique ou morale à utiliser et reproduire ce document pour son propre usage à condition d'en citer la source.

[En savoir plus](#)

### 3 COMMUNAUTÉ

Partagez votre expertise, échangez autour de vos projets IT et faites-nous part de vos retours d'expérience sur l'utilisation des modèles de cahiers des charges.

[En savoir plus](#)

### 4 INFOGRAPHIES

Des statistiques, comptes rendus d'étude, éléments de réflexion sur une cinquantaine de sujets IT. Téléchargez librement ces infographies sur guidescomparatifs.com.

[En savoir plus](#)

### 5 INTERVIEWS

Les responsables informatiques s'expriment sur la mise en œuvre opérationnelle de leurs projets : conseils, anecdotes pratiques, pièges à éviter...

[En savoir plus](#)

### 6 FORMATIONS

Une gamme de sessions d'une journée destinées à approfondir un sujet et à matérialiser la démarche de préparation d'un projet.

[En savoir plus](#)

GUIDES COMPARATIFS

Le portail collaboratif du cahier des charges

# INTRODUCTION :

## Contexte technologique, méthodologie et éléments de cadrage

### DevOps : un concept, une méthodologie

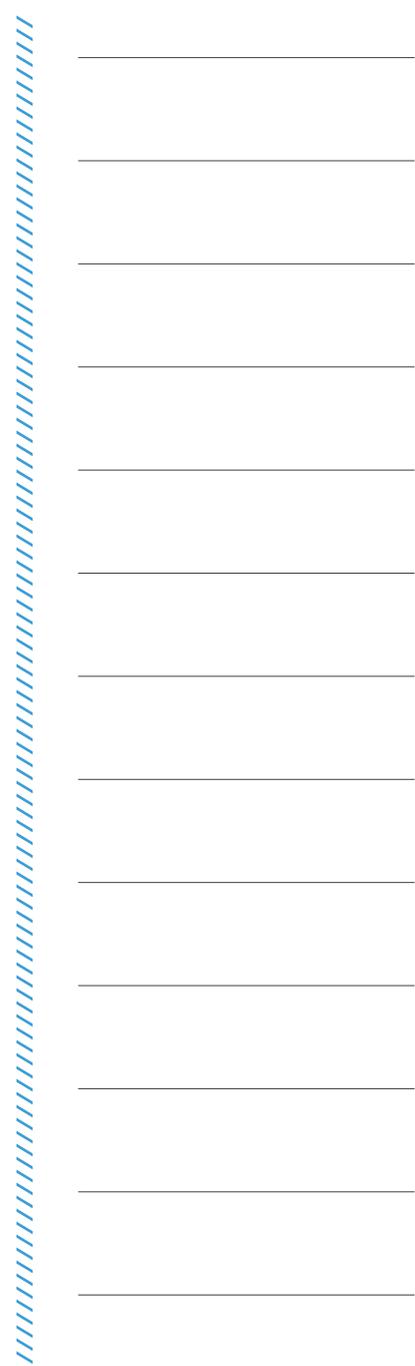
Selon le cabinet Gartner, d'ici 2018, 50% des entreprises s'appuieront sur une solution de release automation pour soutenir leurs initiatives DevOps (contre seulement 10% aujourd'hui).

Souvent présenté comme une méthodologie de travail, le DevOps consiste à unifier les équipes en charge du développement et de l'exploitation afin de déployer des mises à jour logicielle en production le plus fréquemment possible et aboutir à un environnement de travail plus cohérent. Sur un plan organisationnel, le concept de DevOps passe par un rapprochement des équipes de développement, de tests et d'exploitation, le but étant de les inciter à coordonner leur travail et à collaborer plus étroitement que dans les organisations traditionnelles en silos.

### Release Automation : un outil de réconciliation des équipes Dev et Ops ?

La transposition de la démarche DevOps au sein d'un outil capable d'automatiser les déploiements applicatifs n'est pas toujours facile à appréhender car les populations Dev et Ops n'ont pas les mêmes besoins et doivent assurer des tâches techniques très différentes.

Dans un objectif d'intégration continue, les équipes en charge du développement souhaitent avant tout bénéficier d'une solution de release automation qui s'intègre facilement avec leur outillage de développement. Sous pression pour livrer toujours plus de fonctionnalités sur des cycles courts, ils s'investissent généralement peu dans les tâches de configuration et de déploiement. D'un autre côté, les équipes Ops doivent prendre en charge les contraintes de la production. Problématiques de DNS, de load balancer, d'adresse IP, de firewall, de répartiteurs de charge ou de sauvegarde... submergées par de nombreuses actions à assurer quotidiennement, les équipes de production ont rarement le temps de pallier les problèmes issus du développement. Elles auront besoin d'un outil facile à superviser,



avec la possibilité de fixer des cycles d'approbation et de visualiser simplement les actions passées, en cours ou à venir.

L'intégration des fonctions Run et Build au sein d'une solution d'automatisation exige également une plateforme qui offre une vue agnostique des technologies. Pour synthétiser, cette plateforme devra présenter les capacités suivantes :

- Une interface utilisateur proposant des objets et modèles réutilisables afin de faciliter la conception de flux de traitements et réduisant le temps nécessaire pour construire les processus. Des tableaux de bord seront également présents pour contrôler l'état de santé des processus.
- Un moteur d'automatisation s'appuyant sur un référentiel central, robuste et sécurisé.
- Une certaine « extensibilité » propre à supporter des environnements plus complexes (solutions ITSM, workloads, PRA...).
- Une intégration transparente dans les processus existants.
- Des technologies distribuées pour assurer la connectivité avec l'infrastructure et les applications, à la fois dans le Datacenter et dans le Cloud.

### Déploiement de packages d'application, modélisation des actions de déploiement, des environnements et de l'application elle-même

Automatiser le déploiement d'une application, c'est contrôler un ou plusieurs packages d'installation et déclencher leur installation sur différents environnements en respectant le processus de déploiement de l'entreprise.

Le premier objectif du release automation est donc de déployer un package d'applications en production, ce qui implique les actions suivantes :

- Fabriquer le package
- Le déployer sur différents environnements afin de réaliser des tests techniques, unitaires et d'intégration
- Transférer le package vers des environnements de tests fonctionnels (utilisateurs)
- Le valider puis le lancer en production

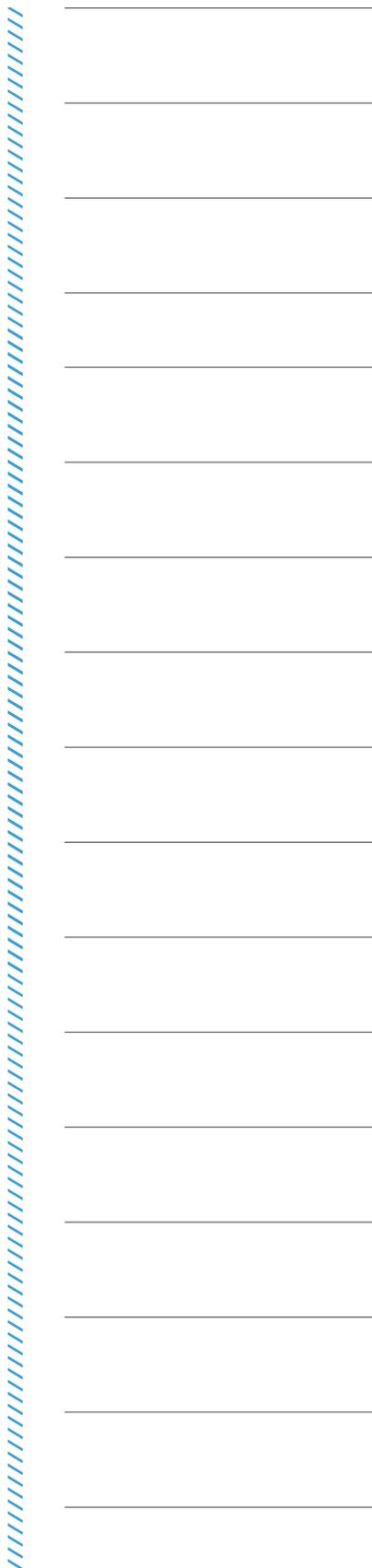
La solution de release Automation permettra d'automatiser l'ensemble de ce processus. Pour y parvenir, il faudra modéliser les différentes actions de déploiement. Cette étape se matérialise par des workflows de déploiement, c'est-à-dire des actions techniques exécutées par des machines, comme par exemple :

- Récupérer un package d'applications stocké sur un serveur de fichier
- le déplacer vers une machine cible
- si ce package est compressé, le dézipper
- déclencher un script d'installation
- injecter des variables dans les éléments de configuration, afin de donner des informations sur l'environnement de travail spécifique dans lequel on se situe (numéro de port, adresse IP...),
- démarrer l'application
- interroger une page http pour vérifier que l'application fonctionne.

Toutes ces tâches habituellement supportées par les équipes OPS pourront être exécutées par la solution de release Automation.

La plateforme pourra également modéliser des environnements – connexions de machines qui représentent les différentes étapes du cycle de déploiement : développement, test, pré-production, production. Elle ne se limitera donc pas à définir des machines cibles sur lesquelles les actions seront exécutées. Au final, l'objectif est de définir un pipeline, en précisant l'ordre dans lequel les environnements se succèdent ainsi que les droits accordés à chaque étape.

Enfin, la solution devra permettre de modéliser l'application elle-même. Sachant qu'une application peut être une agrégation de plusieurs composants, cela entraînera la création de plusieurs packages d'installation (exemple : une application web installée avec une base de données implique deux composants et donc 2 packages).



## Du déploiement automatisé des applications à la livraison continue d'infrastructure...

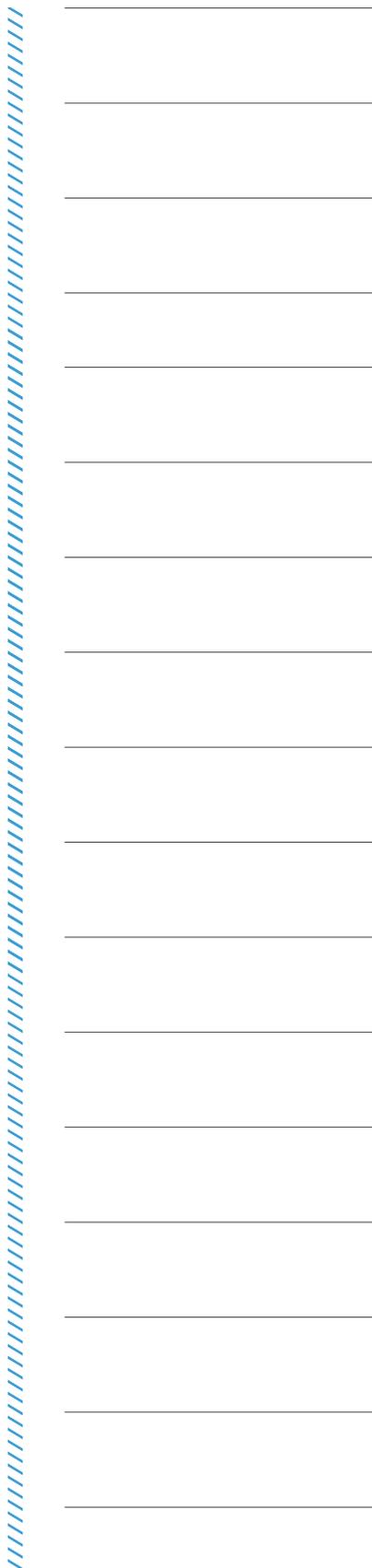
Pour synthétiser, une plateforme de Release Automation doit permettre :

- de déployer de façon continue de nouvelles fonctionnalités applicatives sans impacter la production
- de migrer des applications métier d'un environnement à un autre à mesure que le logiciel gagne en maturité sur l'ensemble du processus de livraison continue.

Sur le terrain, l'automatisation du déploiement des applications permet de réduire les ressources à mettre en œuvre, d'éviter les problèmes de synchronisation entre les différents intervenants, d'éliminer les erreurs manuelles et les temps morts, de déclencher des actions parallèles (comme l'installation d'un grand nombre de machines simultanément) ou encore de rendre les mises à jour transparentes... Il s'agit des apports les plus immédiats d'une solution de release automation.

Cela dit, les véritables bénéfices d'une plateforme de release automation sont à rechercher au-delà de l'application elle-même. La possibilité de faire du provisioning permet non seulement de déployer les applications, mais également l'ensemble de l'environnement. Par exemple, on peut envisager de déployer l'application, la machine hôte, ainsi que toutes les actions d'administration déjà paramétrées (sauvegardes quotidiennes, relances de service, ...). L'automatisation peut également être intéressante dans le cadre d'un PRA ou intégrée à des solutions d'ITSM.

Ce guide vous donnera les clés pour cibler les fonctionnalités essentielles d'une solution de release automation et identifier comment l'automatisation du déploiement des applications peut accélérer l'implémentation de votre démarche DevOps.



# SOMMAIRE :

## Release automation, implémentation DevOps et déploiement automatique d'applications

### 1 MODÉLISATION

- 1.1 Packages d'installation et environnements cibles
- 1.2 Gestion de la bibliothèque d'actions
- 1.3 Administration des Plugins
- 1.4 Modélisation d'applications, gestion des versions de package

### 2 ADMINISTRATION

- 2.1 Gestion des utilisateurs et droits d'accès
- 2.2 Maintenance et mises à jour

### 3 EXÉCUTION / MONITORING

- 3.1 Déploiements
- 3.2 Gestion des environnements
- 3.3 Paramétrage des actions et workflows
- 3.4 Outils de monitoring

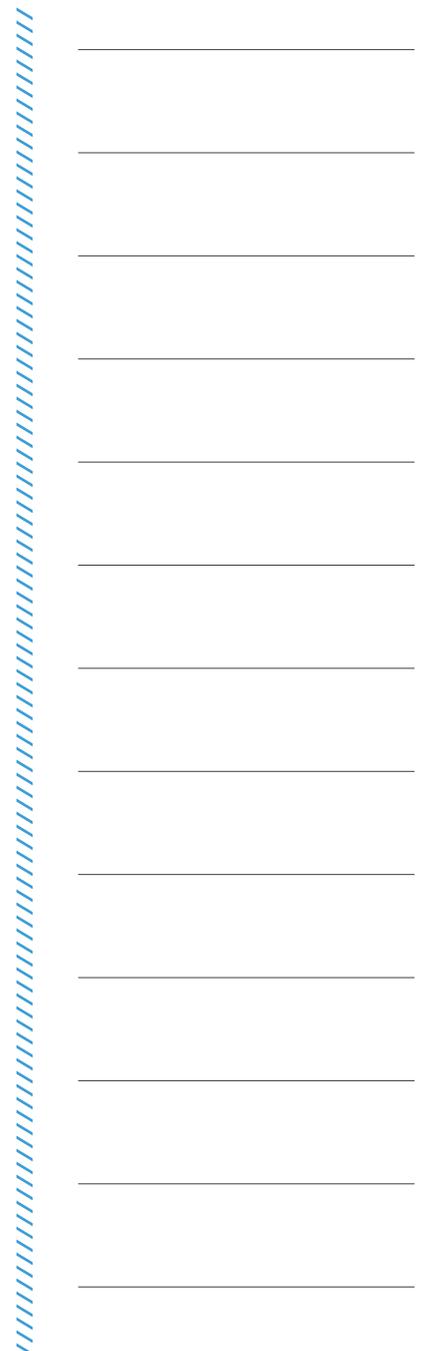
### 4 INTÉGRATION

- 4.1 Connecteurs
- 4.2 Performances, capacités d'intégration, gestion du changement

### 5 ARCHITECTURE

### 6 EXTENSIBILITÉ

- 6.1 Provisionnement des machines
- 6.2 Automatisation du PRA
- 6.3 Orchestration
- 6.4 Intégration avec l'ITSM
- 6.5 Workloads









**Peut-on attribuer des droits pour la définition ou la modification d'un mot de passe ou d'une chaîne de connexion ?**

- Oui
- Non

**Est-il possible de conditionner l'attribution de ces droits à certains environnements (de manière à limiter les utilisateurs habilités à des environnements bien définis ?)**

- Oui
- Non

## 2.2 Maintenance et mises à jour

**Peut-on réaliser la maintenance et/ou upgrades de la plateforme sans impacter les workflows en cours ?**

- Oui
- Non

**Existe-t-il un mécanisme de verrouillage des workflows afin d'éviter que plusieurs utilisateurs interviennent simultanément sur les mêmes actions et écrasent leur travail respectif ?**

- Oui
- Non

**Comment les mises à jour sont-elles gérées ?**

## 3. Exécution / Monitoring

### 3.1 Déploiements

**Peut-on exécuter des déploiements manuellement ?**

- Oui
- Non

**Peut-on exécuter des déploiements automatiquement (intégration continue) ?**

- Oui
- Non

**La solution permet-elle de planifier facilement les déploiements ?**

- Oui
- Non



- Arrêter un déploiement
- Redémarrer un déploiement en cas d'erreur

**Dispose-t-on d'outils de traçabilité des modifications du workflow ?**

- Oui
- Non

**Si Oui, est-il envisageable de :**

- Identifier l'utilisateur qui a modifié le workflow
- Identifier les actions qui ont fait l'objet d'une modification
- Savoir à quel moment la modification a eu lieu

## 4. Intégration

### 4.1 Connecteurs

**La solution dispose-t-elle de connecteurs ou de plugins pour se connecter aux systèmes de repository de code source ?**

- Oui
- Non

**Si oui, quels sont les protocoles supportés :**

- CVS
- SVN
- Git
- TFS
- Autres :

**Des connecteurs sont-ils prévus pour les systèmes d'Artifacts ?**

- Oui
- Non

**Si oui, quels sont les protocoles supportés ?**

- Nexus
- Artifactory
- Autres :

**Les artifacts sont-ils stockés directement par l'automate ?**

- Oui
- Non

**Peut-on optimiser les transferts d'artifacts de machine en machine ?**

- Oui
- Non



**L'outil est-il cross plateforme ?**

- Oui
- Non

**Si oui, supporte-t-il notamment :**

- Java
- Linux
- Dot net
- Mainframe

**La solution repose-t-elle sur une architecture multi-tenant ?**

- Oui
- Non

**Quels niveaux de scalabilité, de fiabilité et de robustesse propose la solution ?**

## 6. Extensibilité

### 6.1 Provisionnement des machines

**L'outil permet-il le provisioning :**

- Bare metal
- De machines virtuelles (VmWare, Virtual box...)
- Dans le cloud (services Amazon, Openstack, Azure...)
- D'outils de conteneurisation

### 6.2 Automatisation du PRA

**La solution est-elle pertinente pour des actions d'automatisation de la reprise d'activité après incident ou sinistre ?**

- Oui
- Non

**La solution permet-elle d'envisager l'automatisation :**

- Des procédures de reprise technique
- Des procédures de reprise applicative

### 6.3 Orchestration

**Un portail utilisateur permet-il de publier des services d'orchestration ou des workflows ?**

- Oui
- Non

**Récapitulez les différents types d'outils que l'on peut-on orchestrer à travers la chaîne de livraison continue ?**

- Outils de gestion du code source
- Outils d'automatisation des tests
- Serveur de Build
- Référentiel d'objet
- Automatisation des configurations
- Automatisation / déploiement des serveurs distants
- Autres outils de déploiement
- Coordination des releases
- RDBMS
- NoSQL
- Outils de déploiement de Middleware
- Automatisation des tests
- Outils de gestion des packages
- Containerology
- Cloud/laaS/PaaS
- Gestion de projet
- Collaboration / Communication

### 6.4 Intégration avec l'ITSM

**La solution s'intègre-t-elle aux outils d'ITSM de type :**

- systèmes de ticketing (systèmes de traçage des opérations de déploiement)
- outils d'ITSM de type « configuration management database » (CMDB)
- outils de suivi de qualité

### 6.5 Workloads

**La solution permet-elle d'automatiser les activités d'administration qu'implique un déploiement applicatif, et notamment :**

- Les sauvegardes quotidiennes
- Les relances de service
- La chaîne de traitement

**Lors du provisionnement d'une machine, peut-on fournir par avance un ensemble d'actions schedulées afin d'assurer son administration à long terme ?**

Oui  
Non

## Utiliser les guides

Les guides proposés en téléchargement sont des introductions aux principales fonctionnalités des solutions technologiques. La structure et le contenu de ces guides constituent une excellente base pour la prise en main de ce sujet et pour disposer d'une base solide pour préparer un cahier des charges ou un comparatif.

Ce guide a pour principale vocation de faciliter l'appropriation d'une telle démarche par les acteurs du projet. Il représente le meilleur compromis entre une démarche standardisée et une démarche personnalisée de choix.

Un projet de choix et de mise en oeuvre d'une solution s'appuie sur une démarche d'analyse, de compréhension et de modélisation des besoins. Chaque critère présenté se doit d'être qualifié, personnalisé et soumis à une évaluation comparative, au plus près des spécificités de l'entreprise.

En fonction de ces analyses, il sera possible de sélectionner et pondérer les critères du guide pour bâtir une grille d'évaluation personnalisée dont le remplissage et la lecture conduiront aux choix technologiques.

En résumé, un projet de choix et de mise en oeuvre d'une application de gestion intégrée s'appuie sur une démarche d'analyse, de compréhension et de modélisation des métiers de l'entreprise et de leurs interactions : ce guide a pour principale vocation de faciliter l'appropriation d'une telle démarche.

## Notations et classements d'offres

Les guides n'intègrent pas de notation, classement ou jugement de valeur sur les offres.

En matière de projet d'entreprise, tout classement universel est inadapté et faux : une offre est parfois plus adaptée que d'autres au contexte d'un projet ou d'une entreprise. Cette même offre sera peut-être moins adaptée que les autres pour un projet différent. C'est en ce sens que les guides ont été conçus. Sélectionner et pondérer les critères du guide en fonction de chaque projet permet de bâtir une grille d'évaluation personnalisée dont le remplissage et la lecture orienteront les choix technologiques.

Il n'y a donc volontairement aucune note ni classement dans les documents, comme on peut en trouver dans les comparateurs d'appareils numériques, caméscopes, matériels électroménagers...

