



---

Automatisation avec Ansible

---

Identifier les tâches à automatiser

---

Automne 2024

---

## Table des matières

1	Identifier les tâches à automatiser .....	3
1.1	Contexte .....	3
1.2	Automatisation .....	3
1.2.1	Étudier l'installation de MediaWiki .....	3
1.2.2	Élaborer l'architecture technique de MediaWiki .....	4
1.3	Préparer l'automatisation du déploiement.....	5
1.3.1	Choisir un outil d'automatisation .....	5
1.4	Taches à automatiser .....	5

# 1 Identifier les tâches à automatiser

## 1.1 Contexte

Dans n'importe quelle entreprise, il y a plusieurs employés qui ont besoin de partager de l'information. Notre mandat est de mettre en place un **wiki** pour répondre à ce besoin.

Un **wiki** est un **gestionnaire de contenu** orienté **rédaction collaborative** et de **partage d'informations structurées**.

## 1.2 Automatisation

Plusieurs questions nous viennent à l'esprit :

- Quel wiki installer ?
- Comment installer un wiki ?
- Comment automatiser le déploiement de ce wiki, et avec quel outil ?

La méthodologie à suivre pour répondre à toutes ces questions est de se concentrer sur des critères prioritaires qui sont à définir en fonction du contexte de l'entreprise.

Dans notre cas, on a fait le choix d'installer **MediaWiki**, car c'est une solution **open source éprouvée** et **modulaire**. Ce qui colle parfaitement avec le contexte de l'entreprise.

### 1.2.1 Étudier l'installation de MediaWiki

La meilleure solution pour comprendre comment installer MediaWiki est de se rendre sur le site de **MediaWiki** à la recherche du **guide d'installation**.

On peut résumer l'installation de MediaWiki avec les étapes suivantes:

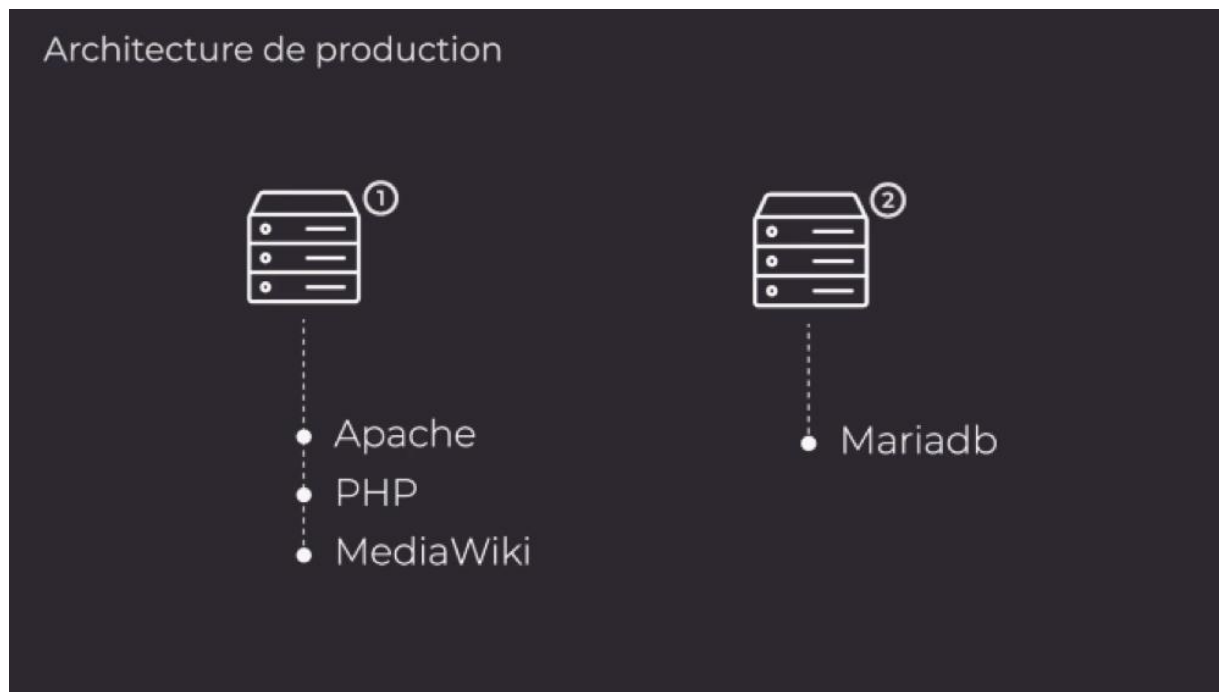
- Installer un **serveur web** pour servir les pages à un navigateur web.
- Installer **PHP** pour exécuter le logiciel.
- Installer une **base de données** pour stocker les pages.
- Télécharger les **fichiers sources** de MediaWiki et les mettre sur le serveur web.
- Configurer le serveur web pour pointer vers l'**URL MediaWiki**.
- Finaliser l'installation de MediaWiki via **le script d'installation**.

### 1.2.2 Élaborer l'architecture technique de MediaWiki

Maintenant qu'on a la bonne recette pour installer MediaWiki, on va élaborer l'architecture technique de **MediaWiki**.

Pour être conforme aux critères de départ, on a besoin de mettre en place une architecture modulaire. On a fait le choix d'installer un serveur Linux (**Centos**) avec un service web (**Apache**, **PHP**) et un serveur Linux avec une base de données (**MariaDB**). On utilise **AlmaLinux**, **Apache** et **MariaDB**, car ces solutions sont compatibles avec l'installation de MediaWiki, et elles sont déjà utilisées dans l'entreprise.

Voici le schéma de l'architecture :



Pourquoi séparer les services sur deux serveurs, alors qu'un serveur aurait pu suffire ?

Eh bien, dans une optique de **scalabilité**, il est toujours intéressant de séparer les services et d'intégrer une architecture **modulaire** dès la conception. Ainsi, il sera très facile pour faire évoluer l'infrastructure. Par exemple, si un seul serveur web n'est pas suffisant pour absorber tout le trafic web, alors il sera possible d'ajouter un serveur web supplémentaire.

### 1.3 Préparer l'automatisation du déploiement

Pourquoi faire une installation manuelle alors qu'on peut scripter le déploiement et l'automatiser?

C'est le principe de l'**Infrastructure-As-Code**; c'est un concept qui permet de gérer les tâches d'administration automatiquement, via du code, à l'aide de fichiers de définition plutôt que faire des traitements manuels. C'est très populaire en ce moment.

Ainsi, on peut réutiliser ces scripts pour un autre déploiement ; on gagne du temps et on peut partager les scripts avec nos collègues.

#### 1.3.1 Choisir un outil d'automatisation

Il faut trouver un **outil de gestion** de configuration. Il en existe plusieurs, mais les plus connus et les plus utilisés sont **Chef, Puppet et Ansible**.

Ils ont tous des avantages et des inconvénients ; le choix doit porter sur :

- 1) le **contexte de l'environnement technique** (un outil est déjà utilisé en interne, le nombre de serveurs à gérer est important...) ;
- 2) les **connaissances internes** des équipes techniques : Chef et Puppet utilisent **Ruby**, alors que Ansible utilise **Python**. **Ansible est simple à utiliser** ; les autres outils, un peu moins, le temps d'appropriation est donc un peu plus long ;
- 3) le **besoin** : est-ce qu'on veut faire de la conformité, du déploiement, de la performance, de la simplicité, de l'intégration, de la sécurité... ?

Comme il n'y a pas encore d'outil de gestion de configuration dans l'entreprise, on choisit **Ansible** pour sa simplicité de prise en main.

En effet, on connaît **Python**, on a pour habitude de privilégier la simplicité, et on ne veut pas que l'intégration de cet outil soit une contrainte sur les serveurs.

### 1.4 Taches à automatiser

On va automatiser les étapes suivantes avec **Ansible** :

- 1) Installer un serveur web **Apache** sur le **premier serveur**.
- 2) Installer **PHP** également sur le **premier serveur**.
- 3) Installer une base de données **MariaDB** sur le **deuxième serveur**.
- 4) Télécharger les **fichiers sources de MediaWiki** et les mettre sur le **serveur web Apache**.
- 5) Configurer le **serveur web Apache** pour pointer vers l'URL **<http://http1/mediawiki>**.
- 6) Finaliser l'installation de MediaWiki avec **un script d'installation** qui est détaillé dans la documentation.

Il faudra donc automatiser les **6 étapes** nécessaires à l'installation de **MediaWiki**.