

# Créer des diagrammes en AsciiDoc

Equipe SNED Incubateur

# Table of Contents

1. Le contexte .....	1
2. Utilisation de Dita .....	1
3. Utilisation de PlantUML .....	1
4. Utilisation de Graphviz .....	2
5. Intégration dans Maven .....	2
6. Limitations .....	3
7. Les liens .....	3

# 1. Le contexte

Un bon schéma valant souvent mieux qu'un long discours, AsciiDoc ne permet pas seulement de mettre en forme du texte via une syntaxe légère. Il permet aussi l'utilisation d'outils/plugins qui vont permettre d'illustrer facilement des documents.

Cela se fait sans avoir recours à des outils externes et toujours en générant les images au moment du build. Votre IDE préféré suffit maintenant pour créer des diagrammes complexes.

## 2. Utilisation de Dita

Dita est une petite lib qui permet de générer des diagrammes à partir de dessins en "Ascii Art".

*Exemple de diagramme en Dita :*

```
[dita,exemple-dita,png]
---
/-----\      /-----\
|c9FC      |      |c9FF      |
|  pn034   |      |  pn035   |
| (Tapestry)|      | (PN NG)  |
|          |      |          |
\-----+-----/  \-----+-----/
      |               |
      +-----+-----+
      |
      v
/-----\      /-----\
|cFF9 Facade de ressources securisees +----->|cCCC      |
+-----+-----+      |  Open AM  |
|          ex012         |      |          |
|cFF9      (SLD NG)      |      \-----/
\-----+-----/
---

```

*Diagramme produit par l'exemple ci-dessus.*

[exemple dita]

Dita produit des schémas propres et clairs qui sont suffisants dans beaucoup de cas.

### CAUTION

A noter tout de même que la facilité de maintenance est inversement proportionnelle à la taille du schéma.

## 3. Utilisation de PlantUML

La création de certains diagrammes UML est possible avec Dita (voir [Limitations](#)) mais il n'a pas été conçu pour cela.

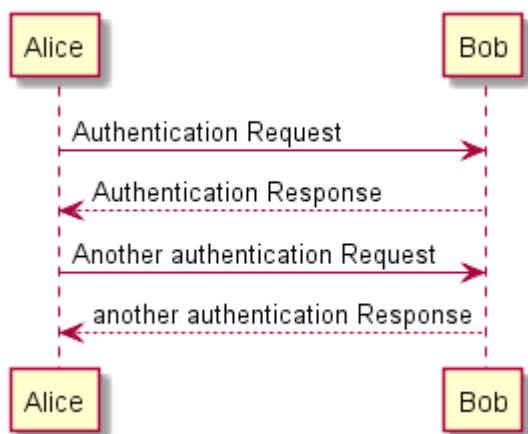
PlantUML propose une syntaxe simple et élégante pour produire facilement des diagrammes UML

très complets.

Le code suivant permet de produire le diagramme de séquence ci-dessous

```
[plantuml,exemple-sequence,png]
---
Alice -> Bob: Authentication Request
Bob --> Alice: Authentication Response

Alice -> Bob: Another authentication Request
Alice <-- Bob: another authentication Response
---
```



PlantUML permet de produire tous les types de diagrammes UML et propose une grande variété d'options pour chacun d'eux.

## 4. Utilisation de Graphviz

Il est aussi possible d'utiliser l'outil Graphviz pour générer des schémas à partir d'Asciidoc mais cela n'a pas été mis en oeuvre dans le contexte du SNED. (voir [Limitations](#))

## 5. Intégration dans Maven

Pour Dita, l'intégration dans Maven se fait simplement en ajoutant une dépendance au plugin *asciidoc-maven-plugin*.

Pour PlantUML, on a besoin de la même dépendance mais également de Graphviz. Voir [Limitations](#)

```
<plugin>
  <groupId>org.asciidoctor</groupId>
  <artifactId>asciidoctor-maven-plugin</artifactId>
  <version>1.5.3</version>
  ...
  <dependencies>
  ...
    <dependency>
      <groupId>org.asciidoctor</groupId>
      <artifactId>asciidoctorj-diagram</artifactId>
      <version>1.3.1</version>
    </dependency>
  </dependencies>
</plugin>
```

## 6. Limitations

Lors du développement de notre doc, nous avons rencontré un problème de génération des diagrammes PlantUML dû aux règles de sécurité Pôle Emploi sur les serveurs Linux.

En effet, PlantUML nécessite Graphviz pour fonctionner correctement. Si, sur un PC standard, il est possible d'installer et d'utiliser Graphviz, il en va tout autrement pour les machines PE qui n'ont pas accès au Net.

Actuellement, pour palier le problème, nous utilisons Ditaa pour générer nos diagrammes de séquence. Mais si ces derniers sont gros, la création et la maintenance sont bien plus compliqués qu'avec PlantUML.

Le GitHub de AsciiDoctor propose une image Docker qui semble contenir tous les éléments pour générer des documents AsciiDoc. L'image Docker embarque aussi tous les outils pour générer les diagrammes qui vont avec, mais nous n'avons pas encore pu essayer cette solution qui semble prometteuse.

## 7. Les liens

[La doc de base pour les diagrammes](#)

[Les diagrammes avec Ditaa](#)

[Les diagrammes avec PlantUML](#)

[Graphviz](#)