# # Prise en main d’Elasticsearch

## ## Résumé

Ces derniers jours, je me suis initié à [Elasticsearch](https://www.elastic.co/). Je me suis donc dis qu’un petit billet sur le sujet permettrait d’éclairer certains. A travers cet article, je vais donc vous expliquer :

* Comment mettre en place une suite Elasticsearch et Kibana sur votre poste ?
* Comment tester les fonctionnalités basiques d’Elasticsearch grâce à Java ?

Bonne lecture !

## ## Introduction

Comme l’indique le titre de cet article, je ne me présente pas en expert d’Elasticsearch. Je me dis plutôt que le cheminement d’un novice du domaine pourrait vous être utile.

Cet article a vocation ainsi à :

* Vous permettre de maîtriser l’installation de votre Elasticsearch de test
* D’interagir avec Elasticsearch via Kibana
* Puis d’interagir avec Elasticsearch avec un peu de code Java

Warning : Comme souvent, la [documentation officielle](https://www.elastic.co/guide/index.html) a été mon principal point d’information. Je ne vous saurais donc trop en recommander la lecture. Elle vous indiquera pas à pas les étapes.

## ## Prérequis

Avant de débuter, je dois vous préciser que ce billet n’a pas vocation à vous expliquer des intérêts ou principes de fonctionnement d’Elasticsearch. Je pars du principe que vous les connaissez ou que cela vous importe peu au vu de vos besoins. Ce billet se centre donc exclusivement sur sa mise en place et son utilisation.

Je préfère également vous préciser la pile applicative que j’utilise afin de mettre en place Elasticsearch :

* Système d’exploitation : Windows 10
* JDK Java : 1.8.0\_201
* Elasticsearch 6.6.0
* Kibana 6.6.0

## ## 1ère étape : Installer la suite Elastic

Afin de prendre en main Elasticsearch, je m’appuie sur plusieurs outils de la suite Elastic. La documentation officielle s’appuie sur la suite :

* Beats
* Elasticsearch
* Elasticsearch Hadoop
* Kibana
* Logstash

Warning : Dans le cadre de la suite Elastic, il est nécessaire d’utiliser la même version pour l’ensemble des produits de la suite.

Pour mon cas pratique, je m’appuie simplement sur Elasticsearch et Kibana en version 6.6.0. Vous trouverez ci-dessous le détail de la procédure d’installation de ces outils.

### ### Comment installer Elasticsearch ?

Elasticsearch est un moteur de recherche utilisant « Lucene » pour l’indexation et la recherche de données.

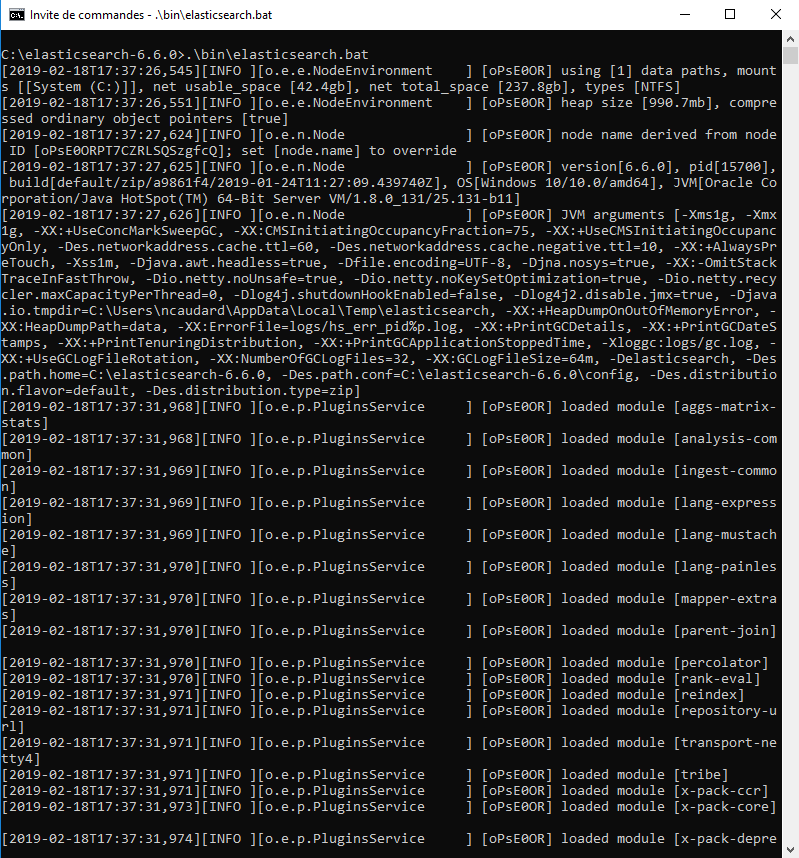
#### #### Installer Elasticsearch sur Windows avec une archive ZIP

Elasticsearch peut s’installer de plusieurs manières : de l’archive, au package Debian en passant par le conteneur Docker ou l’installeur Windows. Dans le cadre de la prise en main d’Elasticsearch, je me suis appuyé sur la [procédure d’installation Windows](https://www.elastic.co/guide/en/elasticsearch/reference/6.6/zip-windows.html).

Pour installer Elasticsearch, vous devez réaliser les actions suivantes :

1. Télécharger l’archive (au format ZIP) d’Elasticsearch V6.6.0 sur le [site officiel](<https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-6.6.0.zip>)
2. Dézipper cette archive au sein du chemin : « C:\elasticsearch-6.6.0 »
3. Ouvrir une invite de commande Windows :

|  |
| --- |
| cd C:\elasticsearch-6.6.0  .\bin\elasticsearch.bat |

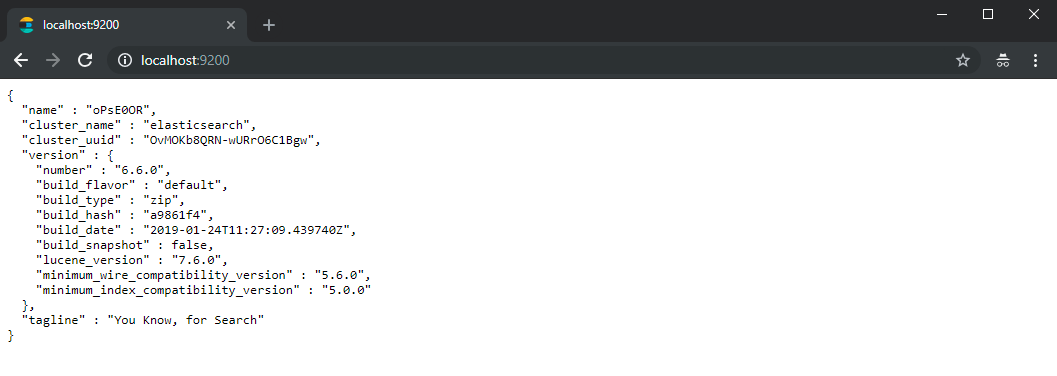


Warning : Par défaut, Elasticsearch trace ses logs au sein de l’invite de commande. Il est alors possible de le couper avec la commande « CTRL + C ».

#### #### Vérifier qu’Elasticsearch fonctionne

Pour vérifier qu’Elasticsearch est correctement installé, ouvrez un navigateur web à l’adresse suivante : <http://localhost:9200/>

Le navigateur devrait alors vous donner l’affichage suivant :



### ### Comment installer Kibana ?

Kibana est un outil de visualisation de données pour Elasticsearch. Il permet la visualisation de contenu indexé au sein d’Elasticsearch.

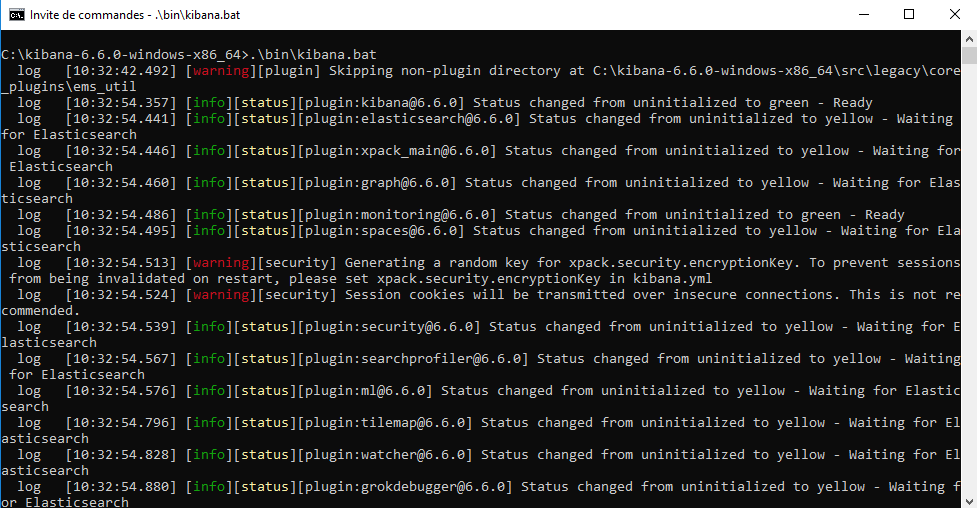
#### #### Installer Kibana sur Windows avec une archive ZIP

Kibana comme Elasticsearch peut s’installer de plusieurs manières : de l’archive au package Debian en passant par le package RPM. Dans le cadre de la prise en main d’Elasticsearch, je me suis appuyé sur la [procédure d’installation Windows](<https://www.elastic.co/guide/en/kibana/6.6/windows.html>).

Pour installer Kibana, vous devez réaliser les actions suivantes :

1. Télécharger l’archive (au format ZIP) de Kibana V6.6.0 sur le [site officiel]( <https://artifacts.elastic.co/downloads/kibana/kibana-6.6.0-windows-x86_64.zip>)
2. Dézipper cette archive au sein du chemin : « C\kibana-6.6.0-windows-x86\_64 »
3. Ouvrir une invite de commande Windows

|  |
| --- |
| cd C:\kibana-6.6.0-windows-x86\_64  .\bin\kibana.bat |

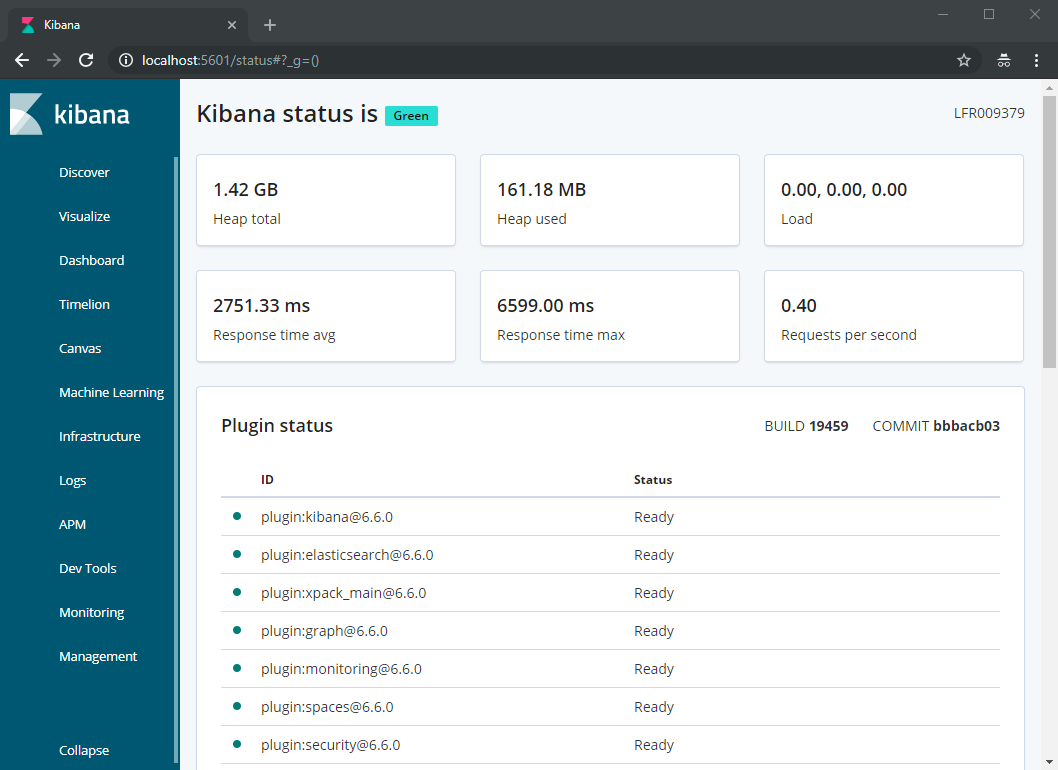


Warning : Par défaut, Kibana trace ses logs au sein de l’invite de commande. Il est alors possible de le couper avec la commande « CTRL + C ».

#### #### Vérifier que Kibana fonctionne

Pour vérifier que Kibana est correctement installé, ouvrez un navigateur web à l’adresse suivante : <http://localhost:5601/status>

Le navigateur devrait alors vous donner l’affichage suivant :



## ## 2ème étape : Tester Elasticsearch avec Kibana

Maintenant que les outils Elasticsearch et Kibana sont correctement installés, nous allons tester le moteur Elasticsearch. Kibana permet de tester les fonctionnalités basiques très facilement.

Warning : Au sein d’Elasticsearch, les concepts suivants sont manipulés : Cluster, Node, Index et Document. Dans la suite de cet article, nous utiliserons essentiellement les mentions Index et Document.

* Un index est une collection de documents qui disposent de caractéristiques similaires
* Un document est une unité basique d’information qui peut être indexé

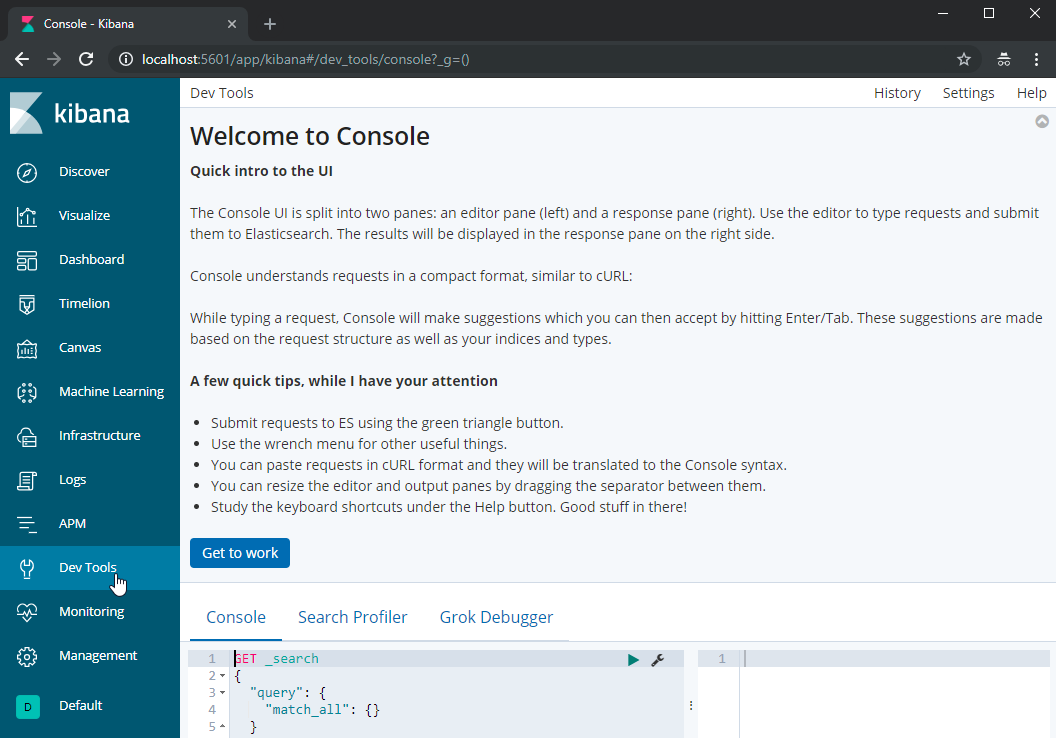
Dans notre cadre de test, nous allons partir du scénario suivant : « à travers Elasticsearch, nous allons y stocker une liste de documents. Nous nous appuierons sur Elasticsearch pour rechercher les documents les plus appropriés au vu de mes critères de recherche. Chaque document est constitué des informations suivantes :

* « Titre » : titre du document
* « Contenu » : contenu du fichier PDF d’un document
* « Auteur » : auteur du document
* « MotsCles » : liste de mots-clés associés au document

Nous allons donc voir comment réaliser les opérations simples suivante :

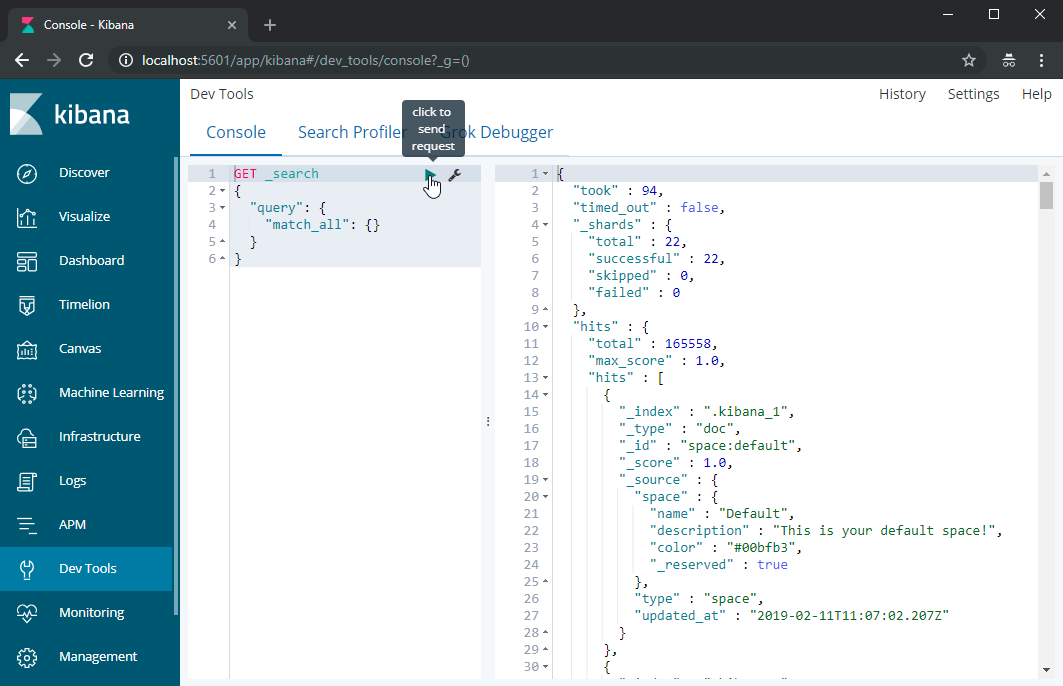
* Création d’un document
* Modification d’un document
* Suppression d’un document
* Recherche de documents

### ### Manipuler Elasticsearch avec la console DevTools

A travers cette 2ème étape, nous nous appuierons sur Kibana via son menu « DevTools » :

Une console est présentée permettant de simuler des interactions avec le moteur Elasticsearch :

* La partie gauche permet d’y inscrire sa requête à destination du moteur de recherche.
* La partie droite permet d’y vérifier la véracité du résultat.



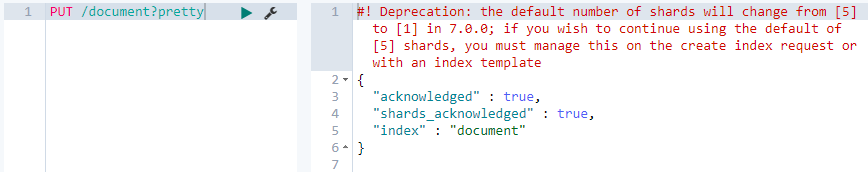
### ### Créer son premier document

#### #### Créer un index

Pour débuter, nous allons créer un index intitulé « document » avec la commande suivante :

|  |
| --- |
| PUT /document?pretty |

Cette commande crée l’index intitulé « document » comme le précise la sortie sous Kibana :



Warning : Il est possible de créer un index par l’intermédiaire de la commande suivante :

|  |
| --- |
| DELETE /document?pretty |

#### #### Ajouter un document

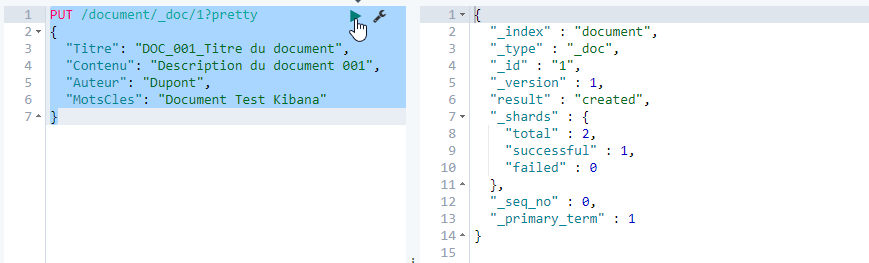
Maintenant que nous avons créé un index, nous allons maintenant insérer notre premier document. Nous y insérons les critères de recherche nécessaire :

* Le titre : « DOC\_001\_Titre du document »
* Le contenu : « Description du document 001 »
* L’auteur : « Dupont »
* Des mots clés : « Document Test Kibana »

Pour cela, utiliser la commande suivante :

|  |
| --- |
| PUT /document/\_doc/1?pretty  {  "Titre": "DOC\_001\_Titre du document",  "Contenu": "Description du document 001",  "Auteur": "Dupont",  "MotsCles": "Document Test Kibana"  } |

Cette commande insère le document comme le précise la sortie sous Kibana :



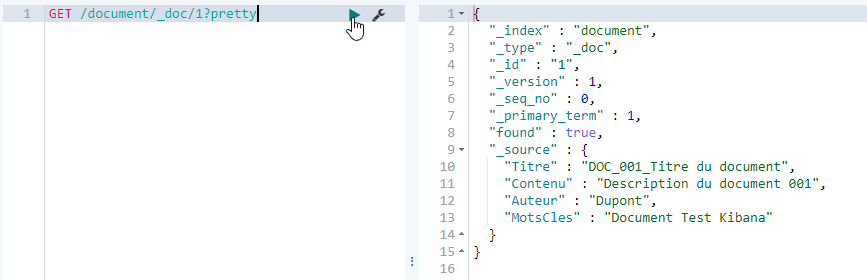
Warning : La valeur « successful » précise que la commande s’est réalisée avec succès et que celle-ci était une création via la mention « created » de la valeur « result ».

#### #### Récupérer un document

Maintenant que notre document est correctement inséré, nous pouvons le vérifier. Pour cela, utiliser la commande suivante :

|  |
| --- |
| GET /document/\_doc/1?pretty |

Cette commande insère le document comme le précise la sortie sous Kibana :



Warning : La valeur « \_source » précise le contenu du document.

### ### Modifier un document

Nous allons maintenant modifier l’une des valeurs d’un document grâce à la commande suivante :

|  |
| --- |
| POST /document/\_doc/1/\_update?pretty  {  "doc": { "Contenu": "Description du document 001 après une mise à jour" }  } |

Cette commande modifie le document comme le précise la sortie sous Kibana :



Warning : Le document est mis à jour et celui-ci incrémente sa version comme l’indique le champ « \_version ».

Vous pouvez vérifier sa bonne modification en vous référant à la commande de récupération d’un document.

### ### Rechercher un document

#### #### Ajouter des documents supplémentaires

La force d’Elasticsearch est bien dans ses capacités de requêtage. Nous allons effleurer ses capacités à partir de 2 types de recherche :

* Une recherche mono-critère classique
* Une recherche multi-critère basée sur des notions de pertinences

En amont de cette phase, je vous invite à y ajouter deux nouveaux documents. Pour ma part, je lance la commande suivante :

* Pour le 2nd document :

|  |
| --- |
| PUT /document/\_doc/2?pretty  {  "Titre": "DOC\_002\_Titre du document Word",  "Contenu": "Description du document 002 (au format autre que PDF)",  "Auteur": "Durant",  "MotsCles": "Document Test Word"  } |

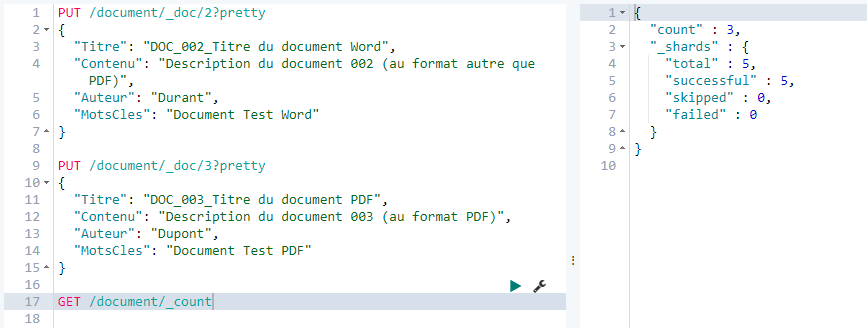
* Pour le 3ème document :

|  |
| --- |
| PUT /document/\_doc/3?pretty  {  "Titre": "DOC\_003\_Titre du document PDF",  "Contenu": "Description du document 003 (au format PDF)",  "Auteur": "Dupont",  "MotsCles": "Document Test PDF"  } |

Vous pouvez vérifier le bon remplissage de votre index via la commande suivante :

|  |
| --- |
| GET /document/\_count |

Cette commande vous précisera le nombre de documents via le champ « count » :



Nous disposons donc des données suivantes au sein du moteur Elasticsearch :

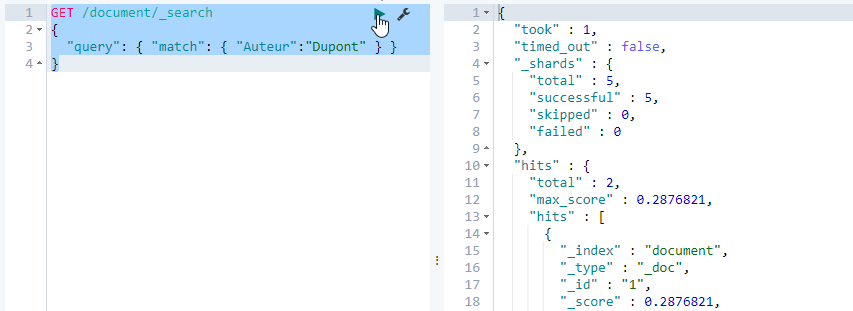
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Titre | Contenu | Auteur | MotsCles |
| 1er document | DOC\_001\_Titre du document | Description du document 001 après une mise à jour | Dupont | Document Test Kibana |
| 2ème document | DOC\_002\_Titre du document Word | Description du document 002 (au format autre que PDF) | Durant | Document Test Word |
| 3ème document | DOC\_003\_Titre du document PDF | Description du document 003 (au format PDF) | Dupont | Document Test PDF |

#### #### Effectuer une recherche mono-critère

Nous allons rechercher la liste des documents dont l’auteur est « Dupont » via la commande suivante :

|  |
| --- |
| GET /document/\_search  {  "query": { "match": { "Auteur":"Dupont" } }  } |

Cette commande liste l’ensemble des documents en accord avec le critère précisé :



Suite à la recherche, Elasticsearch donne les informations suivantes :

* « hits » : la liste des documents
* « total » : le nombre de documents validant les critères demandés
* « max\_score » : le score de pertinence de la donnée jugée la plus pertinente avec la recherche demandée
* « \_score » : pour chaque document retourné, il précise la pertinence de la donnée au regard de la recherche demandée

Au vu de la recherche demandée, les 2 documents disposent du même degré de pertinence.

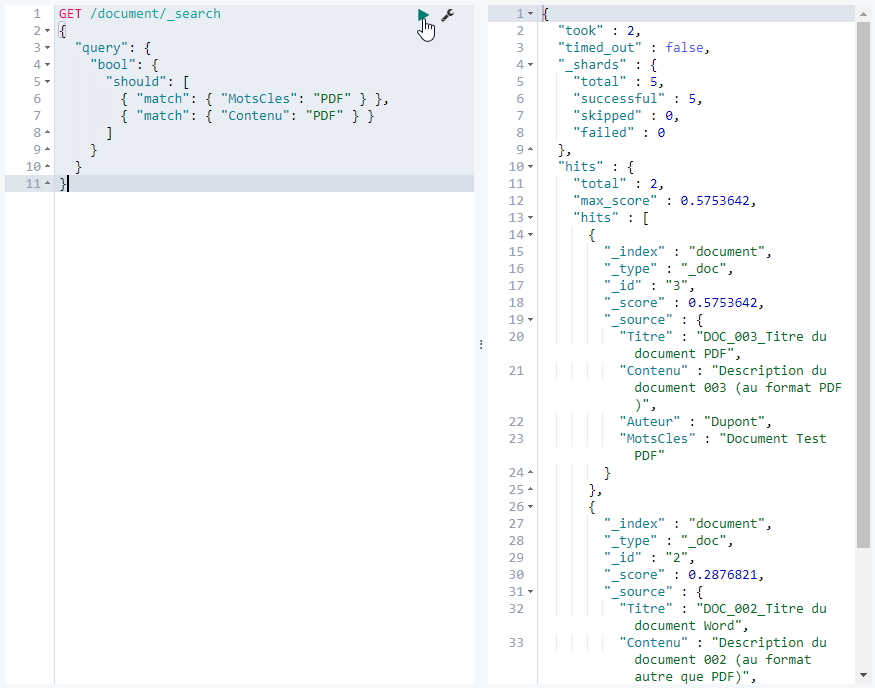
#### #### Effectuer une recherche mono-critère

Nous allons désormais rechercher la liste des documents mentionnant la notion « PDF ». Nous voulons identifier le document PDF le plus approprié à ma recherche.

|  |
| --- |
| GET /document/\_search  {  "query": {  "bool": {  "should": [  { "match": { "MotsCles": "PDF" } },  { "match": { "Contenu": "PDF" } }  ]  }  }  } |

Cette commande recherche les documents dont les mots clés et le contenu contiennent le terme « PDF ». Via l’instruction « should », cette recherche dispose d’une tolérance dans les résultats qu’elle propose.

Cette commande liste l’ensemble des documents en accord avec le critère précisé :



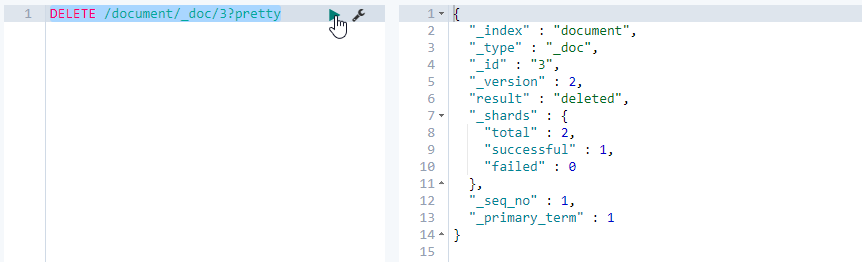
Suite à la recherche, on peut voir que cette recherche donne 2 résultats. Ces résultats disposent d’un score de pertinence qui diverge. L’un des documents dispose d’un score plus élevé que le second.

### ### Supprimer un document

Nous allons maintenant supprimer un document grâce à la commande suivante :

|  |
| --- |
| DELETE /document/\_doc/3?pretty |

Cette commande supprime le document comme le précise la sortie sous Kibana :



Warning : Le document est supprimé et l’index « document » est désormais constitué de 2 tuples.

## ## 2ème étape : Demander un devis

Maintenant que nous connaissons les quelques leviers généraux, nous pouvons effectuer une demande de devis à Oracle.

### ### Les règles à savoir pour contacter Oracle

Lorsque vous contactez Oracle, sachez qu’ils répartissent leur prestation selon le client à adresser. En fonction du client final (ce n’est parfois pas vous-même mais l’entreprise pour laquelle vous réalisez le projet), Oracle vous mettra en contact avec un commercial ou un autre. Il faut donc bien savoir quelle est l’entreprise cliente destinée à disposer de la base de données Oracle.

Ensuite, n’hésitez pas à exposer votre cas d’utilisation à Oracle. Vous pourrez ainsi vous appuyer sur leurs conseils pour adapter votre solution technique. Ne partez pas du principe que vous savez précisément ce que vous voulez. Prenez plutôt l’explication des mécanismes de licence Oracle comme un bonus d’explication afin de pouvoir demander des justifications de coût à Oracle dans son devis.

### ### Les métriques nécessaires à Oracle

Pour effectuer un devis, Oracle vous demandera les données suivantes :

* Si vous effectuez une approche NUP ou Processor
* Dans le cas NUP
  + Edition de la base de données
  + Options supplémentaires souhaitées
  + Nombre d’utilisateurs global (cf. explication dans la 1ère étape)
  + Durée de licence
* Dans le cas Processor
  + Edition de la base de données
  + Options supplémentaires souhaitées
  + Typologie du serveur : modèle de processeur et nombre de cœur physique
  + Durée de licence

Je détaillerais par la suite le cas « Processor » uniquement.

#### #### Métrique N°1 : Edition de la base de données

Plusieurs éditions de base de données Oracle sont disponibles et diffèrent fortement vis-à-vis des fonctionnalités, tarifs et limitations engendrées.

Je vous citerais ici les 4 versions les plus courantes (il en existe d’autres) :

* Oracle Database Enterprise Edition (EE) : version la plus complète et la plus coûteuse. Grâce à cette version, il n’y a pas de limitation matérielle ou utilisateurs. Il est possible d’accéder au paramétrage de la base de données pour des besoins de tuning (entre autres). Il est également possible d’y ajouter des options à acheter en complément du coût de la licence.

Vous pourrez retrouver l’ensemble des options disponibles décrites au sein de [cette documentation](<https://docs.oracle.com/en/database/oracle/oracle-database/18/dblic/Licensing-Information.html#GUID-0F9EB85D-4610-4EDF-89C2-4916A0E7AC87>).

* Cette version est intéressante dans un cadre professionnel où il y a des besoins importants en ressource et/ou des besoins de tuning.
* Oracle Database Standard Edition 2 (SE2) : version plus limitée que la version EE. Il n’existe pas d’option disponible. Il n’est pas possible d’accéder au paramétrage de la base de données (tout du moins, seulement de façon limitée). De plus, cette version est limitée à 2 sockets et une utilisation à 16 threads maximum par CPU.
* Cette version est intéressante dans un cadre professionnel standard où les besoins en ressources sont maîtrisés et conforme aux limitations évoquées ci-dessus.
* Oracle Database Personal Edition (PE) : version à un seul utilisateur qui permet une compatibilité avec les versions EE et SE2. Celle-ci inclue plusieurs composants de la version EE.
* Cette version est intéressante dans un cadre de développement et de tests mono-utilisateur.
* Oracle Database Express Edition (XE) : version gratuite permettant d’utiliser Oracle Database pour ses besoins personnels. Celle-ci dispose de fortes limitations et est davantage à réserver à un cadre non professionnel.
* Cette version est à limiter à un cadre non professionnel (elle n’est pas adaptée à une utilisation en production par exemple). Plus de détail disponible [ici](<https://www.oracle.com/database/technologies/appdev/xe.html>).

Généralement, dans un cadre d’application en production, vous hésiterez entre la version EE et la version SE2. Et au vu des écarts de tarif (conséquents), vous vous poserez la question : les limitations de la version SE2 sont-elles compatibles avec l’utilisation que je prévoie de ma base de données ?

#### #### Métrique N°2 : Options supplémentaires souhaitées

Je ne vais pas être très verbeux sur le sujet. La documentation Oracle vous en dira davantage.

* Sachez que chaque édition d’Oracle permet de disposer d’une option de manière plus ou moins complète. Ce référencement est disponible [ici](<https://docs.oracle.com/en/database/oracle/oracle-database/18/dblic/Licensing-Information.html#GUID-0F9EB85D-4610-4EDF-89C2-4916A0E7AC87>).
* Sachez que pour certaines éditions d’Oracle (la EE entre autres), il est possible d’y ajouter des fonctionnalités supplémentaires. La liste des fonctionnalités supplémentaires est disponible [ici](<https://docs.oracle.com/en/database/oracle/oracle-database/18/dblic/Licensing-Information.html#GUID-AB56CEE3-955E-4E56-8B44-6075E889C283>).
* Sachez que pour certaines d’éditions d’Oracle (la EE entre autres), il est possible de souscrire à des packs comprenant une liste de fonctionnalités supplémentaires. Ce descriptif est disponible [ici]( <https://docs.oracle.com/en/database/oracle/oracle-database/18/dblic/Licensing-Information.html#GUID-68A4128C-4F52-4441-8BC0-A66F5B3EEC35>).

Warning : Oracle fournit également une procédure pour vérifier les options activées sur votre base de données. Ceci nécessite l’usage d’un script SQL. L’ensemble de la procédure est disponible [ici]( <https://docs.oracle.com/en/database/oracle/oracle-database/18/dblic/Licensing-Information.html#GUID-C3042D9A-5596-41A3-A08A-4581FED7634F>).

Il peut résider une vraie difficulté pour savoir ce qui est activé ou non de base au sein de vos options de base de données. Cela est accentué par le fait que plusieurs fonctionnalités sont disponibles mais non activées dans une version SE2. Il suffit alors simplement de les activer. Il est intéressant d’appliquer cette vérification car toute activation d’une fonctionnalité non prévue dans l’édition que vous avez causera une requalification automatique en EE (et donc Oracle sera en mesure de vous demander un réalignement financier).

#### #### Métrique N°3 : Typologie du serveur

Dans une approche « Processor », Oracle part du principe que l’utilisation de son produit Oracle diffèrera d’un serveur à un autre en fonction de sa puissance. Un coefficient est donc appliqué selon le serveur choisit pour héberger votre Oracle : le Core Factor. Ceci signifie qu’il vous faut préciser le modèle de processeur que vous souhaitez utiliser.

Warning : Vous pouvez retrouver une liste (qui date de 2018) des processeurs et du coefficient appliqué [ici](<http://www.oracle.com/us/corporate/contracts/processor-core-factor-table-070634.pdf>).

A partir du processeur utilisé et selon l’édition choisit, il faut :

* Soit prendre en compte le nombre de processeur
* Soit prendre en compte le nombre de cœur

Ce choix est explicité au sein de la documentation ci-dessous :

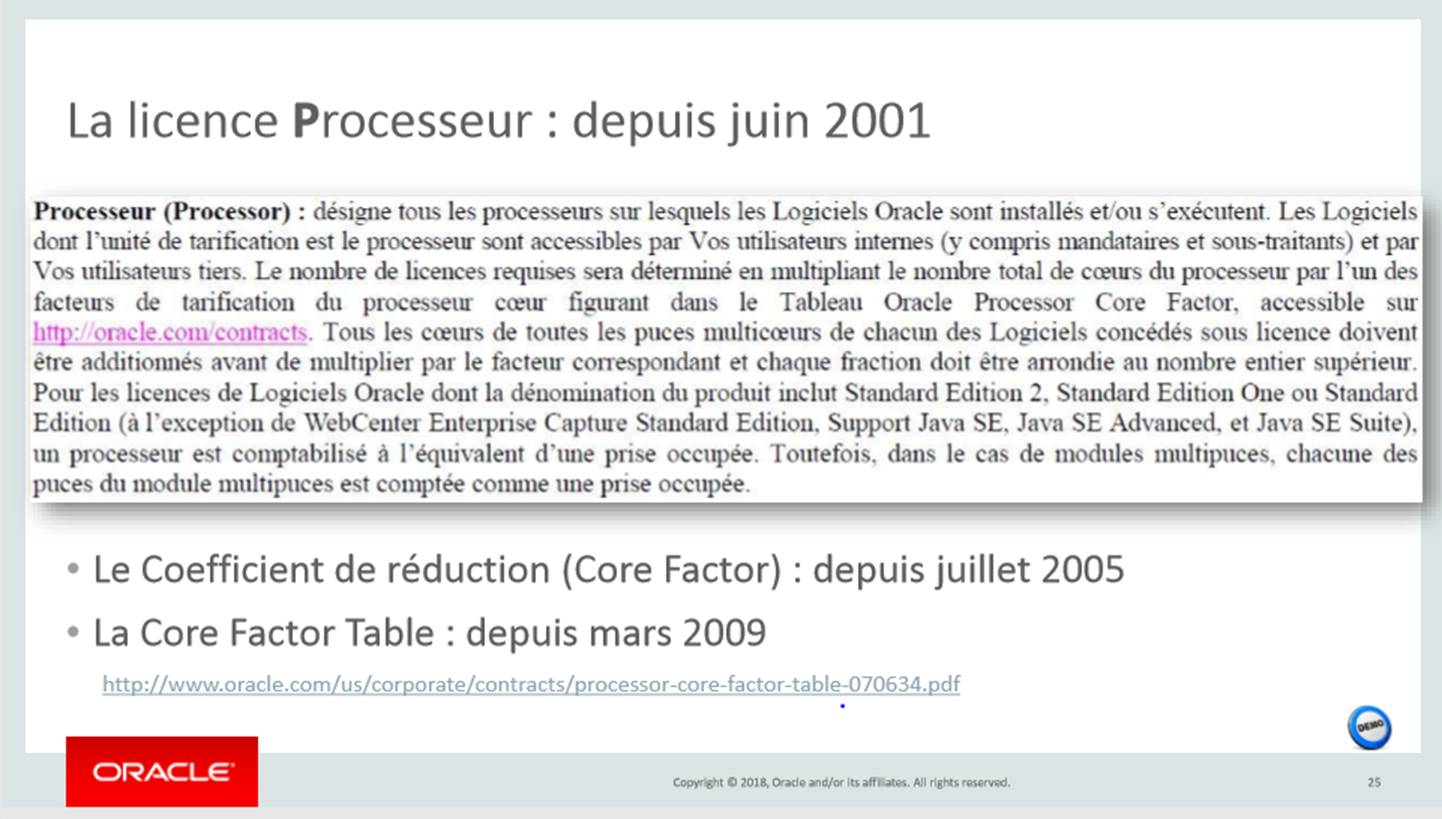


Figure 1 : Documentation officielle Oracle sur la licence Processeur

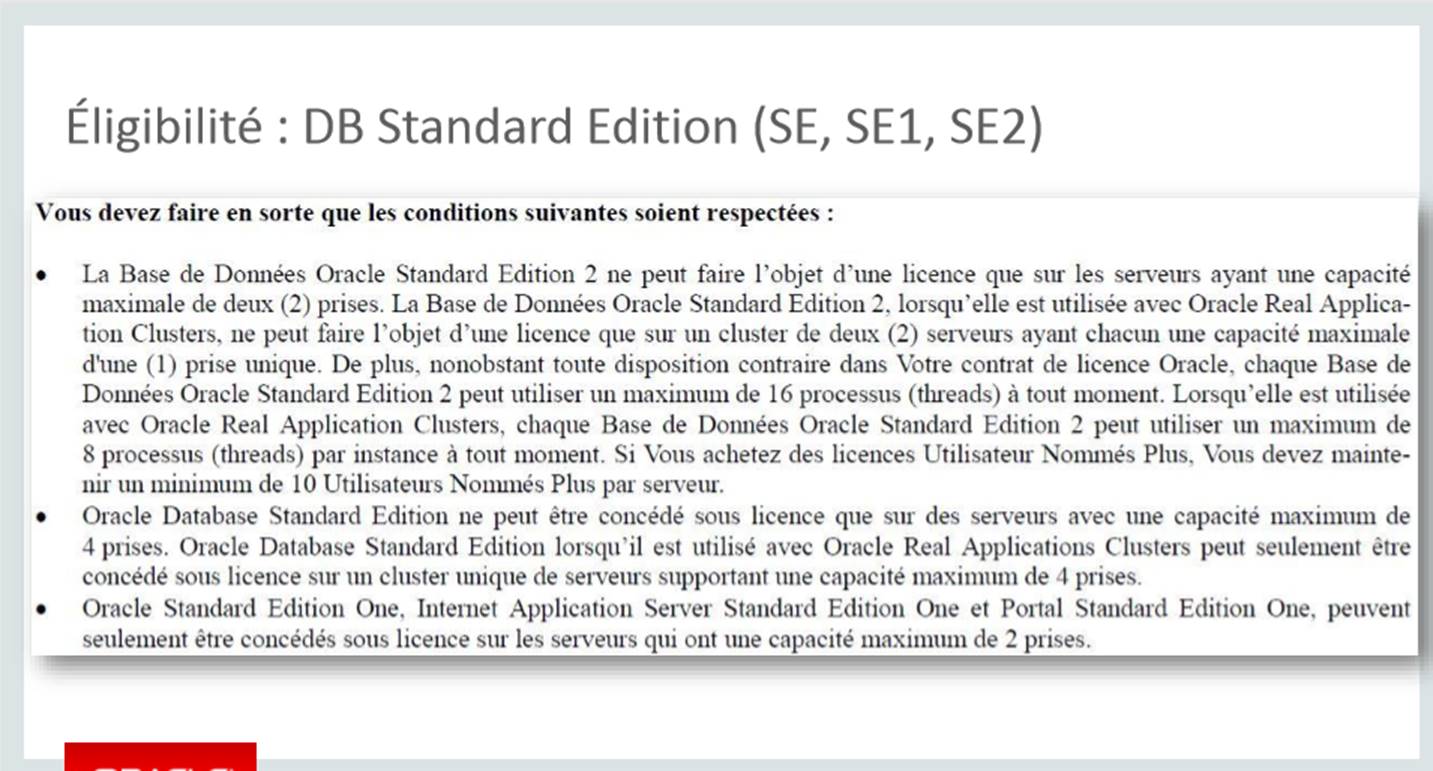


Figure 2 : Documentation officielle Oracle sur les conditions liées aux éditions standards

Dans le cas d’une édition SE, le calcul de licence se base sur le nombre de sockets disponibles. On peut partir du postula qu’un processeur équivaut à un socket. C’est donc le nombre de processeur qui sera nécessaire à Oracle pour estimer le coût.

Dans le cas d’une édition EE, le calcul de licence se base sur le nombre de cœurs disponibles. Il sera appliqué le « Core Factor » à ce nombre de cœurs. C’est donc le nombre de cœurs qui sera nécessaire à Oracle pour estimer le coût.

#### #### Métrique N°4 : Durée de licence

Oracle ne vous précisera peut-être pas ceci en premier lieu. Néanmoins, sachez qu’une licence Oracle n’est pas nécessairement perpétuelle. Si vous ne le précisez pas à la demande, sachez qu’Oracle prendra surement l’hypothèse d’une licence perpétuelle.

Oracle propose donc 2 types de licences :

* La licence perpétuelle : comme son nom l’indique, celle-ci ne dispose pas de fin de vie.
* La licence à terme : celle-ci présente un coût inférieur à la licence perpétuelle mais n’est valable que pour un nombre d’année données. Ce type de licence est possible pour des durées de 1 à 5 ans.

Warning : Retour d’Oracle : « Le prix affiché pour une licence à terme est basé sur un pourcentage spécifique du prix de la licence perpétuelle. Les licences de durées annuelles sont disponibles de 1 à 5 ans :

* 1 an - 20% de la liste ;
* 2 ans - 35% de la liste ;
* 3 ans - 50% de la liste ;
* 4 ans 60% de la liste ;
* 5 ans 70% de la liste. »

Warning : Attention au passage de licence à terme à licence perpétuelle. Oracle ne vous fera pas payer la différence mais vous comptera un coût de licence perpétuelle à 100%.

## ## 3ème étape : Réception du devis

Vous avez effectué votre demande de devis, Oracle vous transmet alors une proposition commerciale. Celle-ci précise généralement les informations suivantes :

* 1 ligne dédiée au coût de la licence proprement dites : celle-ci y précise l’édition de la base de données (cf. 3.3) et la durée de licence choisie (cf. 3.6).
* 1 ligne dédiée au coût de support et mise à jour logiciel : celle-ci, dans un cas nominal, est fournie pour une durée d’1 an.

Dans le cadre de vos appels d’offre, vous souhaitez potentiellement vous projeter sur plusieurs années et donc estimer le coût que cela engendrerait.

* Le coût de licence n’est qu’à estimer une seule fois car son coût est valable selon la durée choisie (donc potentiellement de façon perpétuel).
* Le coût de support et mise à jour logiciel suit une règle d’estimation au fil des années. Selon des retours d’Oracle, la règle actuelle est la suivante :

Warning : Retour d’Oracle : « Il faut compter 22 % du prix d’achat pour une licence pour la première année mais Oracle applique ensuite un IAR (inflation rate) qui est équivalent à environ 3 ou 4 % d’augmentation par an pour chaque année de support complémentaire. »

## ## Conclusion

J’espère vous avoir éclairé sur la logique de licences concernant la partie base de données Oracle. Ce billet n’a pas la prétention à donner une vision exhaustive du sujet mais plutôt à vous donner un vernis. Grâce à ces quelques éléments, vous devriez mieux savoir :

* Quelle édition répondra à vos besoins ?
* Quel matériel associé à vos besoins pour adapter vos coûts de licence de façon réaliste ?

Pour écrire ce billet, je me suis appuyé sur mon expérience personnelle mais également sur la lecture des documents suivants :

* [Cet article](<https://www.softwareone.com/en/blog/licensing-oracle-by-processor-or-named-user-plus-which-metric-is-right-for-you>) vous donnera un avant-goût du sujet en survolant quelques problématiques
* Et je recommande spécialement la lecture de [ce document](<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=7&ved=2ahUKEwippfWh0_7fAhXJzKQKHZXsBeQQFjAGegQIBRAC&url=https%3A%2F%2Fwww.aspera.com%2Ffile%2F33946%2Flivre_blanc_10_choses_a_savoir_audit_oracle%25253Fpage%25253D2214%252526L%25253D2&usg=AOvVaw1p5_oyiEwbDKbHfPSb8kzX>) qui clarifie de façon plus exhaustive les règles s’appliquant aux licences Oracle (même si celui-ci date de 2016).

Plusieurs éléments sont volontairement écartés de ce billet. Je vous précise ici les critères que j’ai pris en compte et dont j’ai une « relative » connaissance. Néanmoins, il faut savoir qu’il existe d’autres variables d’ajustement :

* Les minimums de licence NUP à utiliser selon l’édition choisie ;
* Les solutions de partitioning au niveau physique ou logique ;
* D’autres éditions d’Oracle.

Qui sait, peut être y’aura-t-il un complément à ce billet !