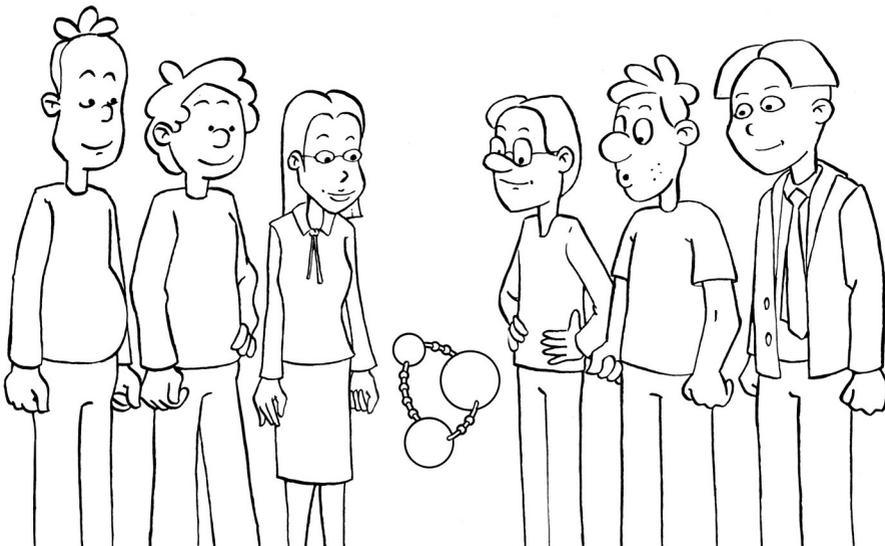


# Bienvenue chez GraphITs.Tech !

---



Bienvenue dans l'étrange univers de GraphITs.Tech !

Le lieu où le génie informatique côtoie des ingénieurs de génie (des gens souvent pas d'accord et pas toujours très commodes). On y a vu des stagiaires apprendre des choses, des développeurs hurler contre l'hérésie d'un code bâclé dont ils avaient oublié qu'ils en étaient eux-mêmes l'auteur, des chefs de projet anxieux, démunis, au mieux pacifiques, et parfois même des commerciaux frisant l'hystérie technique. Tout cela sous le regard binaire d'une équipe dirigeante qui a bien compris que FF ça fait 100000 euros et non pas 255 comme les ingénieurs sus-nommés le leur certifient depuis toujours (on se demande bien ce qu'ils apprennent à l'école ces ingénieurs sus-nommés). Une bonne base en somme.

GraphITs n'est pas une société de services comme les autres. En effet, elle travaille principalement sur des sujets dans lesquels les données sont fortement connectées. Elle peut offrir différents services tels que la détection de fraude, la recommandation ou bien encore l'architecture technique de réseaux sociaux. Quelles que soient les râleries qu'on y entend parfois, tout le monde est très fier de faire partie de cette entreprise novatrice.

Et c'est dans ce joyeux climat qu'un beau matin, en ce beau mois d'une belle année, un nouveau client franchit les portes de GraphITs. Il paraît qu'il a un besoin et ça tombe bien. Car ici, ils aiment bien répondre aux besoins.

Le client en question est un éditeur d'ouvrages techniques connu sous le nom de Editor. Editor a souhaité rencontrer l'équipe de GraphITs afin de lui exposer sa problématique. Et c'est ici que commence notre histoire...

## Personnages

Ilko Presto, *ingénieur d'affaires*

Patricia, *chef de projet*

Christophe, Sylvain, *experts techniques*

Philippe, *expert DevOp et réseaux*

Brian, *stagiaire*

Client, société Editor spécialisée dans la publication d'ouvrages techniques

**Avertissement** > Nous avons volontairement forcé le caractère de chacun des personnages à des fins pédagogiques. N'y voyez aucune moquerie de notre part.

## 1. Réunion de démarrage

*Ilko Presto, ingénieur d'affaires chez GraphITs.Tech, reçoit le client dans la grande salle de réunion où sont présents tous les membres de l'équipe qui sera concernée par ce projet.*

*Ilko* : Bonjour, et bienvenue dans les locaux de GraphITs Services. Vous souhaitez nous rencontrer afin de nous exposer votre problématique, nous vous écoutons.

*Client* : Bonjour, et merci de votre accueil. Comme vous le savez Editor est en passe de devenir l'une des références françaises incontournables dans le domaine de l'édition d'ouvrages techniques portant sur l'informatique, que ces ouvrages soient fournis au format papier, EPUB, PDF ou HTML. Cependant, nous commençons également à traduire nos ouvrages pour les exporter, dans un premier temps, dans les pays anglophones. Nous souhaiterions en profiter pour mettre à jour une partie de notre système d'information afin de le rendre plus efficace face à ces nouveaux enjeux.

*Ilko* : Bien, pourquoi désiriez-vous nous rencontrer ?

*Client* : Il paraît que vous faites dans le NoSQL ?

*Ilko* : C'est exact. Si je peux me permettre, pourquoi avez-vous le sentiment que le NoSQL pourrait vous aider à résoudre votre besoin ?

*Client* : En fait, je n'en suis pas sûr mais je suis encore moins sûr de ce que l'avenir nous réserve. Pour faire court, j'ai entendu dire que le NoSQL facilitait la mise en place de changements dans les structures de données. Voyez-vous, à chaque fois que nous souhaitons mettre à jour nos bases de données SQL, c'est compliqué.

*Ilko* : C'est partiellement juste, tout dépend de quoi on parle. Le NoSQL ne dispense pas d'une conception bien pensée en amont. Ce type de système peut simplifier certaines tâches, notamment sur les modifications structurelles. Mais là encore, tout dépend du type de système que l'on utilise.

### **RAPPEL : Les quatre grands types de systèmes NoSQL**

- *Key-Value store*, données stockées sous forme de paires clé-valeur ;
- *Column store*, données stockées dans un tableau (on parle aussi de *Big Table*) sous forme de colonnes et de super colonnes ;
- *Document store*, données stockées sous forme d'objets littéraux (généralement au format JSON ou XML) ;
- *Graph store*, données stockées sous forme de graphes.

*Client* : Il paraît que vous faites également dans le graphe ?

*Ilko* : C'est encore exact. Je vous en prie, poursuivez.

*Client* : Eh bien, vos concurrents ont eu l'air bien embêtés lorsque je leur ai parlé de relations entre des données que je n'ai pas encore totalement définies. En somme, ils me proposent de stocker l'aspect relationnel dans une base SQL et le reste (les données moins structurées) dans un système NoSQL.

*Ilko* : C'est assez fréquent. De plus en plus, les systèmes d'information conjuguent les avantages des deux types de systèmes.

*Client* : Certes, mais je préférerais un système cohérent. Et l'aspect rigide des relations dans les systèmes SQL est un frein à la mise en œuvre rapide de nouveaux services.

*Ilko* : C'est surprenant ! Vous avez l'air bien au courant des aspects techniques !

*Client* : Comme je vous l'ai dit, nous avons eu beaucoup de soucis lorsque nous avons voulu mettre en place des évolutions dans nos applications. Mes collaborateurs, qui sont loin d'être des débutants, m'ont fait un topo sur le sujet et sur les problèmes qu'ils ont rencontrés.

*Ilko* : Très bien, je vous propose de nous expliquer ce que vous désirez mettre en place, ensuite nous ferons un tour de table pour vous présenter notre équipe.

*Client* : Voici notre besoin : afin de mieux gérer nos commandes qui augmentent de manière croissante, je voudrais mettre en place un processus qui nous permettrait, comme au client, de suivre l'état des transactions, de la commande jusqu'à l'expédition. Ajoutons à cela que je veux voir mon catalogue d'ouvrages relié à ce processus.

*Ilko* : Bien entendu. Je crois que nous en avons saisi l'essentiel pour le moment. Je vais à présent vous présenter les membres de notre équipe. Voici Patricia, qui interviendra en qualité de chef de projet. Patricia, peux-tu nous donner ton avis sur ce besoin ?

*Patricia* : Bonjour, en fait il ne s'agit pas d'un besoin mais de plusieurs petits. Si je comprends bien il y a le catalogue de livres et la gestion d'un processus de commande.

*Client* : C'est cela.

*Ilko* : Voici à présent Christophe, Sylvain et Philippe, nos experts techniques.

*Christophe, Sylvain et Philippe* : B'jour.

*Ilko* : Et voici Brian, notre aspirant développeur... Mais... où est-il passé ?

*Philippe* : In the kitchen, of course !

*Ilko* : Ah. Bien. Hmm. Nous allons les libérer afin qu'ils puissent commencer à s'organiser et ils reviendront vers vous avec un début de solution...

*Client* : Mon besoin n'est-il pas trop classique pour vos équipes ?

*Ilko* : Nous vous montrerons que non seulement un besoin classique peut être résolu par une base de données orientée graphe, mais qu'en plus l'évolution du système s'en trouvera facilitée !

*Patricia* : Oui, il ne fait aucun doute que Neo4j est NoSQL donc simplifie l'utilisation de données aux structures distinctes et imparfaites. Toutefois, ce système peut aussi satisfaire des schémas aux contraintes fortes. Il s'agit aussi (ou d'abord selon les besoins) d'une

base de données transactionnelle ACID, donc fortement cohérente, qui en plus offre une gestion des relations, entre les entités, particulièrement poussée. Or qui peut le plus peut le moins, enfin souvent...

*Client* : En gros, tous les avantages du SQL et du NoSQL sans les inconvénients...

*Patricia* : Eh bien, imaginez qu'auparavant le but était de stocker ses données et que maintenant il s'agit d'aller plus loin. D'où votre démarche.

*Client* : Vous me vendez du rêve, c'est certain. Mais les équipes techniques, suivent-elles ? Dans quelle mesure puis-je m'assurer que le projet sera une réussite ?

*Ilko* : L'expertise, c'est le point fort de nos équipes, c'est pour cela que vous faites appel à nous. Pour l'instant. Une fois qu'on vous aura démontré la puissance de notre solution technologique, il y a de grandes chances pour que vous choisissiez d'en acquérir la compétence. Notre travail sera de vous montrer, étape par étape, que cette réussite est possible. Vous verrez qu'en posant les questions différemment (avec moins de contraintes techniques) on obtient des résultats surprenants.

*Client* : Eh bien, je vous laisse travailler. Quant à vous Ilko, je vous retiens pour parler des modalités pratiques...

## 2. Debriefing

*Ilko, Patricia, Christophe, Sylvain, Philippe et Brian se réunissent afin de tracer les grands axes du projet.*

*Ilko* : Alors, quel est votre sentiment sur ce projet ?

*Christophe* : Pour le catalogue, c'est pas bien compliqué, on pourrait presque le faire avec une base SQL...

*Sylvain* : Et pourquoi pas avec des fichiers CSV tant qu'on y est ?! Même si c'est vrai qu'on pourrait, on ne va pas le faire comme ça. Je ne veux pas me retrouver à gérer deux systèmes là où un seul conviendrait.

*Patricia* : Je suis d'accord avec Sylvain, ce n'est pas parce que ça marcherait aussi avec une base SQL qu'on est obligé d'en passer par là, même si l'intérêt de stocker le catalogue dans le graphe est moindre que pour les autres parties. Si on parlait des technos ?

*Christophe* : Pour ce qui est du catalogue, c'est assez fixe. On pourrait utiliser de l'[OGM \(Object Graph Mapper\)](#) voire même du [SDN \(Spring Data Neo4j\)](#).

Patricia : Ça me paraît pertinent ; et pour le reste ?

Sylvain : Pour les autres parties je propose du CYPHER, sans forcément passer par du *mapping objet* ; éventuellement utiliser les [procédures APOC](#) (Awesome Procedures On Cypher), histoire d'optimiser les traitements entre Neo4j et nos services. On pourra utiliser Bolt comme client TCP, ce sera plus efficace que travailler avec l'API REST.

Patricia : Et s'il nous demande une connexion avec ses outils de reporting ?

Sylvain : Des requêtes CYPHER avec JDBC devraient aller, on vérifiera ça.

Patricia : Il y aura certainement des données existantes à récupérer...

Christophe : Il y a ce qu'il faut dans Neo4j pour faire de l'[import/export de données](#), même massivement.

Philippe : Excusez-moi les gars, mais le client souhaitait accrocher le marché anglophone !

Christophe : Et alors ?

Philippe : Et alors parmi les pays anglophones, il y a par exemple ce petit pays qu'on appelle les États-Unis, avec un décalage horaire d'au moins six heures.

Patricia : Où veux-tu en venir ?

Philippe : Je veux en venir au fait qu'on ne peut plus vraiment compter sur un système à heures creuses, il doit fonctionner vingt-quatre heures sur vingt-quatre. Ce qui signifie [haute disponibilité \(HA\)](#) donc cluster de serveurs Neo4j et sauvegarde des données à chaud.

Patricia : Bien vu !

Brian : Ça fait beaucoup de trucs dis donc... par quoi on commence ?

Christophe et Sylvain : Par [la méthode](#) !

Sylvain : Tiens c'est marrant, pour une fois on est d'accord.

Ilko : OK, je vois que c'est bien parti, je vous retrouve à la prochaine étape.

## Au programme : les grandes étapes

- **Un peu de méthode et d'analyse** : du comment on procède et de l'analyse du besoin jusqu'à l'obtention des graphes ;
- **Interactions avec Neo4j** : sur la mise en place des moyens techniques permettant d'interagir avec Neo4j ;
- **Importation des données** : pour chacun des graphes, importer les données existantes ;
- **Exploitation** : sur la mise en place de la haute disponibilité via un cluster de serveurs Neo4j ;
- **Sécurité** : sur les moyens disponibles pour sécuriser a minima le cluster Neo4j.

### Sources des exemples

Les sources des exemples sont téléchargeables depuis la page de présentation du livre, onglet COMPLÉMENTS, sur le [site des éditions D-BookeR](#).

### URL raccourcies

Dans un souci de lisibilité, et pour pouvoir les maintenir à jour, nous avons pris le parti de remplacer toutes les adresses internet par ce qu'on appelle des URL raccourcies. Une fois que vous avez accédé à la page cible, nous vous invitons à l'enregistrer avec un marque-page si vous souhaitez y revenir fréquemment. Vous disposerez alors du lien direct. Si celui-ci se périmé n'hésitez pas à repasser par l'URL raccourcie. Si cette dernière aussi échoue, vous pouvez nous le signaler !