

JORDANE QUINCY

Étudiant en Master 1 TNSI
Université de Valenciennes
Année scolaire 2015/2016

Rapport d'alternance

Entreprises : OPEN & DECATHLON

Tuteur Entreprise : M. Luc SCHRUFF

Tuteur Universitaire : M. Mikael DESERTOT

Sommaire

Remerciements.....	3
Introduction.....	4
Présentation des entreprises	5
Présentation du projet.....	6
Les personnes clefs.....	7
Environnement techniques	8
Général.....	8
GWT	8
Framework v4	8
Serveur	8
Oracle	8
Infrastructure	8
Environnements.....	9
Impressions	9
Le projet	10
Qualité.....	10
Déplacements.....	11
Urgence	11
Factures	12
Zoom technique - La gestion des kits	12
Évolutions à venir.....	14
Conclusion.....	15
Annexe 1	16
Annexe 2	17
Notes	18

Remerciements

Je tiens tout d'abord à remercier Mr Mikael Desertot, enseignant chercheur à l'université de Valenciennes et du Hainaut-Cambrésis, pour son excellent suivi en tant que tuteur universitaire durant cette première année de master.



Je souhaite ensuite remercier mon maître d'apprentissage, Mr Luc Schruff, qui continue à me fournir de précieux conseils à chaque visite et qui me permet de mieux vivre les valeurs d'Open.



Je remercie également Décathlon pour m'avoir permis d'effectuer cette mission et ses responsables pour la confiance qu'ils m'ont accordée.

Enfin un grand merci aux équipes des projets Twist et Shipperbox pour leur convivialité et plus particulièrement à Baptiste, François et Kévin.



Introduction

Le Master TNSI peut être effectué en alternance ce qui permet d'une part de mettre en application les connaissances théoriques et d'autre part d'avoir un lien fort avec le monde du travail.

Grâce au Groupe OPEN, un des Leaders français des services informatique, j'ai eu la chance de reprendre les études afin d'élargir mes compétences tout en continuant ma prestation chez notre client Décathlon.

Dans un premier temps nous présenterons le Groupe Open ainsi que l'entreprise Décathlon puis nous détaillerons ma mission chez ce client et enfin nous dresserons les enseignements de cette année sur le plan professionnel.

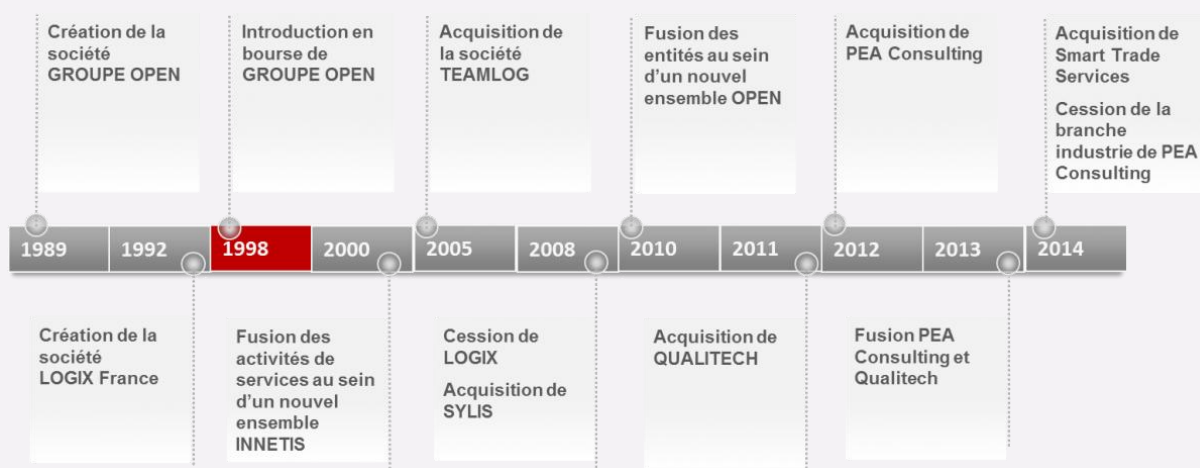
Présentation des entreprises

- **Groupe Open**

Le Groupe Open est une des premières ESNⁱ de France, elle prend racine en 1989 et se développe au point d'entrer en bourse en 1998.

L'audace est une des valeurs d'Open et avec la fusion des SSII Teamlog et Syllis, l'entreprise prend véritablement son envol.

Notre Groupe, résolument tourné vers l'international avec une présence en France, Belgique, Luxembourg et Pays-Bas, partage sa passion avec plus de 3 500 collaborateurs.



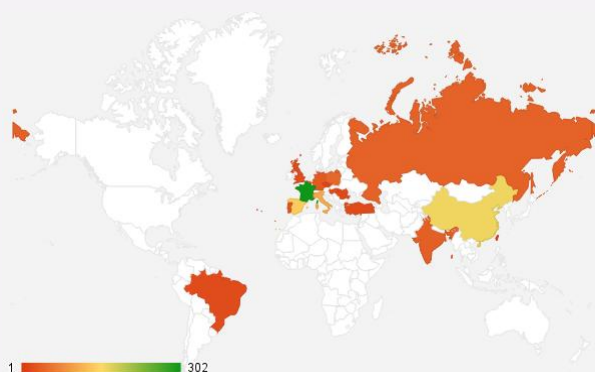
- **Décathlon**

Décathlon est une entreprise française de distribution d'articles de sport.

En juillet 1976, Michel Leclercq a ouvert le premier magasin en libre-service spécialisé dans les produits sportifs à Englos (Nord de la France), son concept est d'offrir à tout le monde (débutant, amateur ou professionnel) la possibilité d'acheter des produits de sport attrayants et à un prix raisonnable.

Par conséquent, son fondateur voulait gérer l'ensemble de la chaîne à partir de la production jusqu'au client, c'est-à-dire, de concevoir, fabriquer et vendre ses produits grâce aux magasins Décathlon.

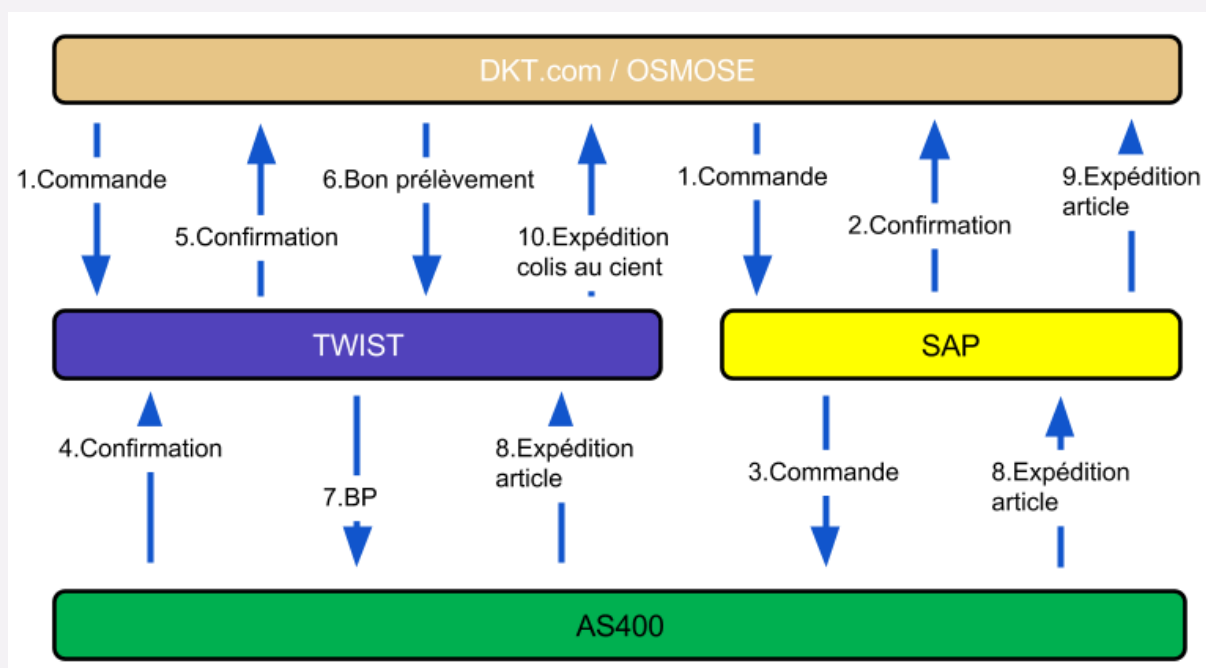
Décathlon compte maintenant plus de 70 000 employés et plus de 1 000 magasins répartis dans 29 pays.



Présentation du projet

Twist est le nom de l'application utilisée par la partie e-commerce des entrepôts de Décathlon sur laquelle je travaille.

Cet outil fait partie d'une chaîne logicielle permettant la gestion des commandes internet de l'enseigne.



Son action s'étend de la prise en charge des magnumsⁱⁱ en provenance du prélèvementⁱⁱⁱ des articles en entrepôts jusqu'à l'expédition aux transporteurs.

Pour ce faire, nous nous appuyons sur plusieurs processus dont les principaux sont :

- La réception : Lorsqu'un article vient à manquer dans un entrepôt CAR^{iv}, celui-ci est commandé sur un entrepôt CAC^v. Une fois arrivé à destination, le terrain^{vi} scanne un code-barres pour confirmer sa bonne réception.
- Le tri/colisage : Les articles sont stockés par secteur^{vii} dans l'entrepôt donc si des commandes contiennent des articles de différents secteurs, il faut que ce prélèvement soit regroupé puis trié afin de réunir les articles de chacune des commandes. Parfois, l'étape de tri n'est pas nécessaire si l'entrepôt décide de prélever l'ensemble des articles d'une commande, c'est ce que l'on appelle le prélèvement à la commande. Une fois les articles regroupés par commande, il faut les mettre dans un emballage adéquat (carton ou sachet) avec la facture et générer une étiquette transporteur.
- L'expédition : Une fois l'étape de colisage terminée, les colis client sont acheminés dans la zone d'expédition où ils peuvent être associés sur une UAT^{viii} ou directement à une expédition^{ix}, via le scan du code-barres présent sur l'étiquette transporteur.

Twist est un projet né en 2010 et avec plus de 50 développeurs différents, le code source est victime du syndrome du plat de spaghettis^x c'est pourquoi j'ai émis l'idée d'un refactoring^{xi} du cœur du système : le colisage.

Depuis 2013, j'ai mis en évidence les gains qui pouvaient être retirés d'un remaniement en profondeur de ce processus dont les principaux sont :

- gain de performance
- diminution des bugs
- évolutions facilitées

Le sujet fut abordé plusieurs fois mais il ne pouvait être réalisé par manque de moyens financiers et humains.

L'arrivée d'une nouvelle recrue en provenance d'un des leaders du commerce électronique allait changer la donne.

Découvrant une autre méthodologie de colisage dans son précédent emploi et ayant approuvé l'efficacité des magasiniers, il souhaitait la mettre en œuvre pour Décathlon.

Il chiffré le retour sur investissement ce qui permit de débloquer les moyens afin de lancer les développements de refonte du colisage, le projet est alors nommé : projet **Packman**.



Les personnes clefs

À l'origine du projet Éric G. est le responsable fonctionnel, c'est lui qui rédige le cahier des charges fonctionnel et qui prend en charge une partie des tests.

La responsable technique, Caroline L., écrit le cahier des charges technique et suit l'avancement du projet.

L'équipe de développement est composée d'Antoine E. et de moi-même.

Environnement techniques

Général

Twist suit le modèle MVC^{xii} et il est construit à l'aide du Framework^{xiii} v4, qui est un Framework interne de Décathlon lui-même basé sur le Framework Spring^{xiv}.

GWT s'occupe de la partie Vue, tandis que le reste de l'application est en J2E.

GWT

GWT est une boîte à outils développée par Google afin de simplifier les développements de site utilisant la technologie Ajax^{xv} en apportant au JavaScript les outils de développements et de débogages de Java, en minimisant les écarts entre navigateurs et en facilitant la gestion des appels asynchrones.

Framework v4

Ce Framework interne tente de poser de bonnes bases de développement via l'utilisation de Maven^{xvi} et de Spring.

Il prend en charge une partie de la gestion des droits des utilisateurs, offre un tableau de bord, permet une gestion facilitée des paramètres et de l'accès aux données.

Serveur

Notre application étant utilisée dans dix pays (pour un total de 24 entrepôts), elle est considérée comme critique d'où la nécessité d'avoir un load balancing^{xvii}.

Le premier avantage est d'augmenter la tolérance aux pannes et le second de pouvoir mettre à jour l'application en minimisant l'impact sur le terrain.

A l'heure actuelle, les instances sont réparties sur six serveurs différents.

WebLogic est la solution retenue pour déployer nos ear.

Oracle

Twist utilise le SGBDR^{xviii} Oracle afin de répartir, sur trois serveurs, l'ensemble de ses données.

L'utilisation d'hibernate^{xix} permet de s'affranchir des spécificités de cette base de données bien que pour certaines manipulations lourdes, les requêtes natives restent privilégiées.

Le framework v4 de Décathlon permet d'obtenir très rapidement des opérations CRUD^{xx} sur de nouvelles entités ce qui permet un gain de productivité important lors de la création de nouvelles tables.

Infrastructure

Deux datacenters^{xxi} indépendants et situés dans des localisations éloignées, mais dont l'emplacement reste secret pour des raisons évidentes de sécurité, hébergent notre infrastructure, qui se trouve ainsi doublée.

Un DRP^{xxii} est effectué plusieurs fois par an afin d'en valider le fonctionnement, la complétude et l'exactitude.

Environnements

Nous avons à notre disposition deux environnements de Pré production (dont un réservé au projet Packman) et un de Production.

La gestion des environnements et plus généralement de l'infrastructure est déléguée à deux autres équipes : une en France et une en Chine.

Ce mode de fonctionnement permet d'avoir un support technique joignable 24H/24 par la personne en charge du support Twist le jour ou celle d'astreinte la nuit.

Impressions

Twist est une application relativement gourmande en impressions, nous utilisons donc des serveurs d'impressions CUPS^{xxiii}.

Il y a deux CUPS par instance afin que le second prenne immédiatement la relève en cas de défaillance du premier.

Le projet



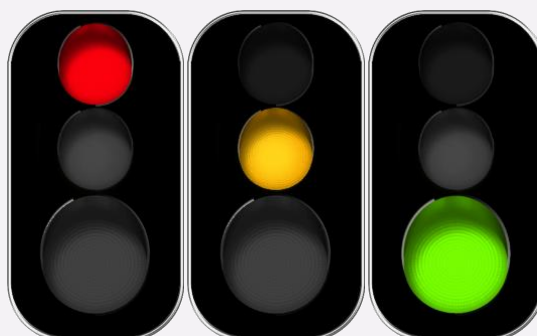
Qualité

Avant ce projet, la qualité des commandes n'était pas toujours au rendez-vous et cela peut s'expliquer par le fait que les contrôles étaient réalisés physiquement et de manière ponctuelle après le colisage. C'est-à-dire qu'en bout de chaîne, la personne ayant effectué l'emballage devait parfois rouvrir un carton afin de rescanner l'ensemble des articles pour en valider le contenu.

Cette manière de faire est contraignante pour les magasiniers et coûteuse pour l'entrepôt, que ce soit en temps ou en consommable car cette opération nécessite la réimpression d'une étiquette transporteur ainsi que du scotch et un nouveau carton.

Grâce à Packman, plus besoin d'éventrer un carton destiné à l'expédition. Le sticker transporteur sera imprimé si et seulement si la commande est complète.

En plus d'avoir un effet direct sur la productivité des magasiniers et leur motivation, nous avons ajouté un feu tricolore dans l'interface afin que l'état d'avancement d'un processus soit visualisable en direct :



Le feu rouge indique que le système attend une action de l'utilisateur, le feu orange indique un changement et le feu vert signifie la réussite du processus.

Déplacements

Historiquement, le tri s'effectue dans un ensemble de racks^{xxiv} attendant à la plateforme, en partant du rack le plus en haut à gauche jusqu'à celui en bas à droite, sans autre forme de hiérarchisation ce qui implique un grand nombre de déplacements, qui sont une perte de temps et d'énergie.

Avec Packman, le concept d'armoire fait son apparition. Une armoire contient un ensemble de racks et se lie à une plateforme. Cela n'a l'air de rien mais en introduisant ce concept, nous pouvons complètement séparer les processus de tri et de colisage qui étaient auparavant fortement couplés.

Fonctionnellement, un magasinier se connecte sur une plateforme, scanne un code-barres en provenance du prélèvement. L'application détecte si un tri est nécessaire et demande à l'utilisateur de placer une armoire en position A, puis si c'est insuffisant il devra placer une armoire en position B puis C.

Selon le nombre de commandes à trier et le nombre d'articles par commande, notre algorithme va permettre de placer les commandes les plus volumineuses au plus près du magasinier (Cf. Annexe 2).

Urgence

Antérieurement à Packman, le terrain n'avait pas d'informations par rapport à l'urgence des commandes donc parfois une commande livrable deux jours plus tard pouvait être prête pour l'expédition chez le transporteur avant une commande qui devait être remise au client dans la journée car il n'y avait pas d'ordre à suivre dans le colisage.

D'autre part, certains transporteurs ne font qu'une ramasse par jour alors que d'autres peuvent en faire plusieurs mais rien ne permettait de coliser au plus vite les commandes devant être chargées dans un camion arrivant prochainement.

Les clients se plaignaient du décalage entre la promesse faite sur le site et la date de livraison effective.

Afin d'améliorer le service au consommateur, c'est le système qui indique maintenant combien de commandes sont urgentes et les prioritaires seront colisées en premier.

Factures

Dans certains pays, il est obligatoire d'imprimer une ou des factures.

A l'heure actuelle, les factures sont éditées en même temps que le sticker transporteur mais cela ralentit considérablement le magasinier qui doit doubler son geste : prendre d'abord l'étiquette puis attendre plusieurs secondes afin que l'imprimante laser atteigne la température nécessaire pour imprimer les factures d'une commande.

Pour des raisons légales, nous ne pouvons supprimer purement et simplement les factures, de plus le temps de chauffe d'une imprimante laser n'étant pas améliorable de manière logicielle, nous avons cherché comment optimiser ce point dans Packman.

Avec notre projet, les temps d'impressions sont d'une part réduits car les factures de toutes les commandes sont éditées en une fois donc nous n'avons qu'une seule phase de montée de température et d'autre part ce temps est également complètement masqué pour l'utilisateur car les tâches d'impressions sont lancées dès le début du tri en parallèle des actions physiques réalisées.

Zoom technique - La gestion des kits

L'exemple typique d'un kit est le banc de musculation qui est généralement divisé en trois colis physiques qui doivent nécessairement être envoyés ensemble au client.



Voici le déroulement du processus avant Packman :

Le magasinier scanne le code-barres d'un article ce qui déclenche un appel au serveur qui retourne une multitude d'informations sur l'article. La page côté client se charge alors de vérifier s'il s'agit d'un kit et si c'est le cas, fait un second appel au serveur pour lancer l'impression d'un sticker pour chaque partie du kit et afficher une pop-up afin de scanner la première partie du kit.

Une fois la première partie du kit scannée, un nouvel appel est fait au serveur. Celui-ci valide qu'il n'y a pas eu de scan de partie inattendue (dans notre cas, cela vérifie que l'utilisateur n'a pas scanné un code-barres de la forme `articleCode_XX_03` où XX est supérieur à 03) et déplace Y quantité de l'article, Y étant un divisé par le nombre de parties du kit (ici, Y vaut donc environ 0,3333333...).

Juste avant de retourner côté client, une vérification est faite pour voir si l'ensemble du kit a été scanné en arrondissant à sept chiffres après la virgule la quantité, le but étant à la fin du processus de passer de trois fois 0,33 soit 0,99 à 1.

Puis le magasinier scanne la seconde et la troisième partie du kit de la même manière.

Au total, pour un kit en trois parties, nous avons donc cinq appels Ajax (2 + 1 par partie), ce qui pénalise grandement la réactivité du système car entre le déclenchement de l'appel et le retour, il s'écoule en moyenne 2 000ms.

Avec Packman, il était hors de question de faire autant d'appels serveurs et d'arrondis approximatifs. Voici comment j'ai implémenté la gestion des kits :

Le magasinier scanne le code-barres d'un article ce qui déclenche un appel au serveur, reconnaissant un kit. Nous lançons alors l'impression des stickers et renvoyons un DTO avec uniquement les données nécessaires, c'est-à-dire le code article et le nombre de parties du kit.

Côté client, nous affichons une pop-up pour scanner les différentes parties du kit. Connaissant précisément ce que le magasinier doit scanner et dans quel ordre, aucun appel serveur n'est effectué. Grâce à un compteur initialisé à 1, la partie client peut valider seule le scan du magasinier avec la chaîne de caractère qu'elle a générée (le code article suivi d'un tiret bas, de la valeur du compteur complétée à gauche par des zéros si besoin est pour qu'elle soit sur 2 caractères, d'un autre tiret bas puis du nombre de partie du kit également complété à gauche par des zéros si besoin est pour qu'il soit sur 2 caractères).

Une fois que toutes les parties sont correctement scannées, nous lançons l'appel serveur du scan article avec le nombre de parties du kit qui ont été scannées (au lieu de 1 dans le cas d'un article non kit), nous pouvons donc côté serveur déplacer une quantité de cet article.

Au total, quel que soit le nombre de parties, il ne persiste plus que deux appels Ajax ce qui permet à Packman de ne pas ralentir le magasinier car la pop-up est immédiatement rafraîchie et côté serveur, la manière de récupérer un article a également été mise à jour afin d'utiliser la clé primaire pour de bien meilleures performances.

Évolutions à venir

Lors du colisage, choisir le carton le plus adéquat pour une commande est un exercice difficile et chronophage en cas d'erreur. Pour améliorer cet état de fait, nous allons fournir le type de carton à utiliser pour chaque commande.

Lorsque le système détecte qu'un article attendu n'a pas été scanné, nous affichons actuellement le code de l'article ainsi que sa description mais cette dernière étant parfois incomplète, le magasinier peut avoir du mal à retrouver dans son magnum cet article. Une photo de l'article sera ajoutée prochainement.

La chaîne de production pose désormais des tags RFID^{xxv} ce qui n'obligera plus les magasiniers à trouver l'étiquette de chaque article pour pouvoir scanner le code-barres, un simple glissement sur une tablette RFID permet de récupérer le code article. Cette fonctionnalité est déjà disponible mais elle sera mise à jour afin de supporter la multi-détection, c'est-à-dire la reconnaissance de plusieurs articles en un seul passage.

Conclusion

Cette année m'a permis de m'améliorer sur plusieurs points :

Premièrement, la gestion de projet m'est apparue comme étant un élément central, que ce soit sur les jalons, les livrables ou les cahiers des charges fonctionnels et techniques.

Ensuite, d'un point de vue technique, ce projet m'a permis de mettre en application de bonnes pratiques telles que le monitoring de performances et le refactoring.

Enfin, d'un point de vue humain, je ressors grandi de cette année car j'ai pris conscience de la nécessité d'exiger et de savoir dire non lorsque les conditions le nécessitent. Cette prise de conscience n'est que le premier pas, je dois désormais m'améliorer dans cette démarche et je suis persuadé qu'avec les nouveaux challenges de l'année prochaine les conditions seront réunies afin de mettre cela en pratique.

Annexe 1

5 Most Important things about PACKMAN

1 1st target is QUALITY



We refuse to get more customers complaining about

- WRONG items
- MISSING items
- Full order switches between Mr A and Mr B

So we will control all items with RFID during the last packing step : 100% QUALITY

2 2nd target is PROMISE

On the field, our teams need to see what is urgent and what is not .

- Priorizing 1 cart, 1 shelf, subgroup or order over another !
- Having the computer pointing out what is urgent

The departure time : DATE + HOUR coming from AS400 will be essential from now on



3 And we want PRODUCTIVITY better too !

BEFORE

We don't scan again the items in MULTI orders when packing.

We scan the destination bin when sorting.

Papers printed when packing.

NO Rfid

VS

AFTER

We use more and more RFID !

We add item scanning in Packing.

We stop scanning the bin when sorting

Papers can be printed when I am sorting (no more waiting!)

4 More than accepting a new version of TWIST, Packman need YOU to change : Equipments & Organisation



This layout and stations will change !



You need to work with your IT to get RFID, computers, printers.



You really should change your PACKstations tables, your SORTING shelves

5 We need someone in your WH to lead this !

Contact Eric GOSSET or Marjorie HORNEZ

Let's Get Ready !

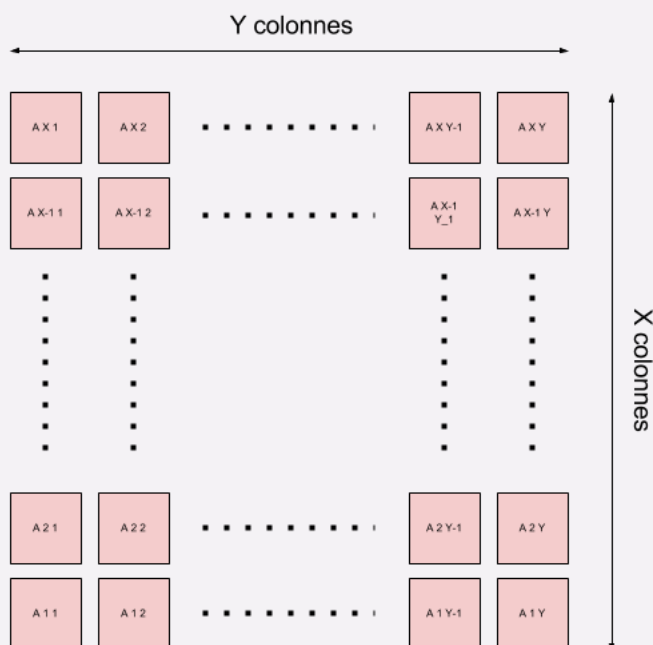
Launch the talks in your WH !

THANK YOU

PackMan Project Team

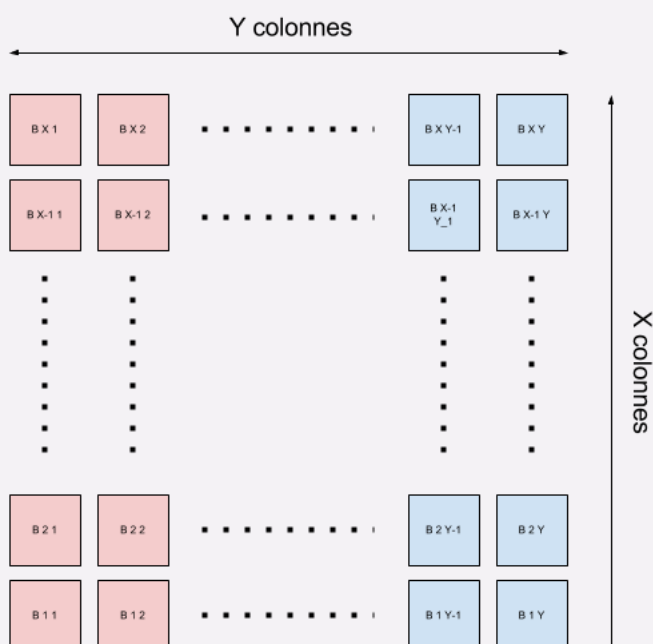
Annexe 2

Assignment des racks de l'armoire A et de l'armoire B dans le cas où le nombre de commandes à traiter est supérieur au nombre de racks de A mais inférieur ou égal au nombre de racks de A + 2 colonnes de l'armoire B :



La commande la plus volumineuse est assignée au rack du haut de la colonne de droite de l'armoire A, la seconde est assignée au second rack (en partant du haut) de cette colonne et ainsi de suite jusqu'au rack du bas de cette colonne. Ensuite c'est au tour de la colonne de gauche de l'armoire B d'être assignée de la même manière. Puis on remplit la seconde colonne de droite de l'armoire A puis la seconde de gauche de l'armoire B et enfin on remplit le reste des racks de A de haut en bas et de droite à gauche.

Armoire A



Armoire B

Notes

- ⁱ ESN : Entreprise de Services du Numérique (anciennement connu sous le nom de SSII (Société de Services en Ingénierie Informatique))
- ⁱⁱ Magnum : Réceptacle de grande taille contenant de nombreux articles.
- ⁱⁱⁱ Prélèvement : Action physique visant à aller chercher dans l'entrepôt un ou plusieurs articles en vue d'une étape de tri et/ou de colisage.
- ^{iv} CAR : Entrepôt logistique de taille plus modeste qu'un CAC ayant vocation à réapprovisionner les magasins et envoyer les commandes e-commerce.
- ^v CAC : Entrepôt logistique de grande taille réapprovisionnant les CAR
- ^{vi} Terrain : Le terrain est un terme utilisé ici pour désigner l'entrepôt et les activités qui lui sont associées.
- ^{vii} Secteur : Sous division d'un entrepôt, un secteur stocke un type d'articles (les volumineux, les standards, les vélos, etc.)
- ^{viii} UAT : Une UAT est une palette qui regroupe un certain nombre de colis client, elle a un sticker qui permet de l'identifier de manière unique et donc d'ajouter tout son contenu à une expédition.
- ^{ix} Expédition : Sous Twist, une expédition est le pendant informatique du contenu du camion d'un transporteur.
- ^{*} Syndrome du plat de spaghettis : « Un système informatique désordonné c'est comme une assiette de spaghettis: il suffit de tirer sur un fil d'un côté de l'assiette pour que l'enchevêtrement des fils provoque des mouvements jusqu'au côté opposé. »
Penny Grubb et Armstrong A. Takang, Software maintenance: concepts and practice, World Scientific
- ^{xi} Refactoring : Action visant à modifier une partie d'un code source afin de la rendre meilleure. Cette action n'a pas pour but d'ajouter une fonctionnalité ou de corriger un bug.
- ^{xii} MVC : Le patron de conception Modèle-Vue-Contrôleur vise à séparer la présentation, des données et de la logique métier.
- ^{xiii} Framework : Ensemble cohérent de composants informatiques utilisés pour suivre un squelette type d'application.
- ^{xiv} Spring : Framework java dont une des caractéristiques phares est l'inversion de contrôle.
- ^{xv} Ajax : L'Ajax permet des échanges entre le client et le serveur dynamiquement à l'intérieur d'une même page (par exemple suite à un click sur un bouton).
- ^{xvi} Maven : Apache Maven est un outil de gestion de build et de gestion de dépendances.
- ^{xvii} Load balancing : Technique permettant de répartir la charge entre plusieurs serveurs. Elle est ici utilisée afin qu'un serveur prenne le relais si l'autre fait défaut.
- ^{xviii} SGBDR : Système de Gestion de Base de Données Relationnel
- ^{xix} Hibernate : Logiciel permettant facilement de passer des objets dans une base de données relationnelle et inversement.
- ^{xx} CRUD : Pour Create, Read, Update, Delete désigne les opérations basiques pour la persistance des données.
- ^{xxi} Datacenter : un centre de données est un lieu dans lequel sont regroupés de nombreux composants d'un système d'informations.
- ^{xxii} DRP : Disaster Recovery Plan ou plan de reprise d'activité permet de s'assurer qu'en cas de crash majeur, l'activité puisse continuer.
- ^{xxiii} CUPS : Common Unix Printing System est un système d'impression modulaire, il est utilisé avec Twist en tant que gestionnaire pour les tâches d'impressions.
- ^{xxiv} Rack : Un rack est une case d'une plateforme de tri/colisage ou d'une armoire. C'est à l'intérieur d'un rack que sont placés les articles d'une commande.
- ^{xxv} RFID : Le RFID est une technique permettant la récupération de données sans contact dans un périmètre restreint.