



ÉCOLE 3IL

MÉMOIRE DE FIN D'ÉTUDES

QU'EST-CE QUE L'INTELLIGENCE ARTIFICIELLE PEUT APPORTER À UN ERP ?

Auteur :

Thomas RANVIER

Tuteur en entreprise :

Florent PALIER

Tuteur pédagogique :

Odile DUVALET

16 août 2019

Remerciements

Je tiens tout d'abord à remercier toutes les personnes m'ayant soutenu et accompagné au cours de mes trois années d'apprentissage à l'école 3iL et au sein de la compagnie DISA.

Je remercie notamment mon tuteur en entreprise M. Florent PALIER, il fut toujours présent et m'a toujours soutenu dans mes projets.

Je remercie également M. Patrick MERY, directeur de la production de DISA pour sa bienveillance à mon égard.

Je remercie M. Nicolas STECLEBOUTE, nouveau responsable informatique de DISA, pour son aide lors de la rédaction de ce mémoire de fin d'études.

Enfin je remercie toute autre personne m'ayant accompagné durant ces trois années, tous mes camarades, ainsi que tous les employés de DISA.

Table des matières

Introduction	4
1 Rapport d'activité	5
1.1 Le contexte de mon apprentissage	6
1.1.1 Présentation de DISA	6
1.1.1.1 Historique	6
1.1.1.2 Le métier d'imprimerie	7
1.1.2 Le service informatique	8
1.1.2.1 L'infrastructure matérielle	8
1.1.2.2 L'environnement logiciel de DISA	9
1.1.2.3 Mon rôle au sein du service	9
1.2 Le projet SIGMA	11
1.2.1 Pourquoi ce projet ?	11
1.2.2 Environnement de développement	11
1.2.2.1 Outils utilisés	11
1.2.2.2 Architecture client lourd-serveur	12
1.2.3 La conception théorique de SIGMA	12
1.2.3.1 Récolte et analyse des besoins des futurs utilisateurs	12
1.2.3.2 Choix de la méthode de travail	13
1.2.3.3 Proposition des fonctionnalités	13
1.3 Le développement de SIGMA	15
1.3.1 Création de la structure de la base de données	15
1.3.2 Application de la méthode itérative incrémentale	15
1.3.3 Mise en production progressive de SIGMA	16
1.4 Mes développements notables sur SIGMA	17
1.4.1 Configurateur de devis	17
1.4.1.1 Le besoin exprimé par les deviseurs	17
1.4.1.2 La solution apportée	17
1.4.1.3 Les raisons de l'échec et abandon de ce projet	19
1.4.1.4 Les solutions compensatoires développées	19
1.4.1.5 Leçons tirées de cette expérience	19
1.4.2 Planning de production	21
1.4.2.1 Le besoin d'informatiser le processus de planification	21
1.4.2.2 Le développement du planning	21
1.4.3 La SFAO de SIGMA	24
1.4.3.1 Le fonctionnement de l'ancienne SFAO	24
1.4.3.2 Le développement de la SFAO sous SIGMA	24
1.4.4 Module des ressources humaines	26
1.4.4.1 La conception du module	26

1.4.4.2	Les développements futurs	27
1.4.5	Mon travail lors de cette dernière mission	28
1.4.5.1	Amélioration de la gestion du travail au sein du service informatique	28
1.4.5.2	Amélioration des performances des requêtes	28
1.4.5.3	Nouvelles fonctionnalités	29
1.5	Retour d'expérience sur mon apprentissage	30
2	Sujet de recherche	31
2.1	Qu'est-ce que l'intelligence artificielle ?	32
2.1.1	Définition de l'intelligence	32
2.1.2	Définition de l'intelligence artificielle	32
2.1.2.1	Penser humainement, la science cognitive	32
2.1.2.2	Penser rationnellement, les lois de la pensée	33
2.1.2.3	Agir humainement, le test de TURING	33
2.1.2.4	Agir rationnellement, les agents rationnels	33
2.1.2.5	Conclusion	34
2.1.3	L'histoire de l'intelligence artificielle	35
2.1.3.1	Avant l'intelligence artificielle (1943–1955)	35
2.1.3.2	La naissance de l'intelligence artificielle (1956)	35
2.1.3.3	Les premières grandes avancées (1952–1969)	35
2.1.3.4	Les premières difficultés (1966–1973)	36
2.1.3.5	L'hiver de l'intelligence artificielle (1974–1980)	36
2.1.3.6	Le retour de l'intelligence artificielle (1980-aujourd'hui)	36
2.1.3.7	Le deep learning et le big data (2011-aujourd'hui)	37
2.2	Que permet de faire l'intelligence artificielle ?	38
2.2.1	Systèmes multi-agents	38
2.2.1.1	Automates cellulaires	38
2.2.1.2	Simulation de foules	39
2.2.2	Algorithmes génétiques	39
2.2.3	Recherche de chemin	40
2.2.4	Modèles de MARKOV cachés	40
2.2.5	L'apprentissage profond	40
2.2.5.1	L'émergence d'agents coopératifs complexes	41
2.2.5.2	AlphaStar	41
2.2.5.3	Autres avancées de DEEPMIND	42
2.2.5.4	Pilote automatique de TESLA v9.0	43
2.2.5.5	GauGAN de NVIDIA	44
2.3	Que doit offrir un ERP ?	45
2.4	Les applications possibles de l'intelligence artificielle au sein des ERP	47
2.4.1	Gestionnaire Électronique des Documents	47
2.4.1.1	Classification de documents	47
2.4.1.2	Partitionnement de documents	50
2.4.1.3	Extraction de données pour saisie automatique	53
2.4.2	Résumés de textes	56
2.4.3	Gestion des stocks intelligente	58
2.4.4	Ordonnancement de phases de fabrication	59
2.4.5	Assistant intelligent permettant d'aider les utilisateurs au quotidien	60
Conclusion		61

Introduction

J'ai effectué ma huitième et dernière mission en entreprise, dans le cadre de ma formation d'ingénieur en informatique à l'école 3iL, de février à septembre 2019. Lors de mes trois années de formation, mon apprentissage s'est déroulé au sein du service informatique de la compagnie DISA. Celle-ci est une imprimerie de labeur basée à Limoges, spécialisée dans l'impression de publicité sur lieu de vente.

Dans ce mémoire, nous allons nous intéresser à la problématique « Qu'est-ce que l'intelligence artificielle peut apporter à un ERP ? ». L'intérêt de cette problématique est de corrélérer un sujet qui me passionne et dans lequel je vais continuer mes études, l'intelligence artificielle, au sujet des ERP, sachant que j'ai travaillé sur le développement d'un ERP dans le cadre de mon apprentissage.

Ce mémoire est divisé en deux principales parties.

La première consiste en un rapport d'activité de mon apprentissage. Le but est d'y décrire le travail que j'ai effectué durant ces trois années. Je vais tout d'abord présenter l'entreprise DISA, puis le projet SIGMA, projet de développement d'un ERP sur lequel j'ai travaillé durant la totalité de mon apprentissage.

La seconde partie est consacrée à mon travail de recherche sur la problématique citée ci-dessus. Dans un premier temps, nous verrons un historique, ainsi que l'état de l'art de l'intelligence artificielle. Ensuite, nous allons aborder différentes applications possibles de l'intelligence artificielle au sein des ERP. Nous comparerons plusieurs algorithmes pour chaque application afin de déterminer les avantages et inconvénients de chacun d'entre eux sans pour autant aller trop loin d'un point de vue technique.

Chapitre 1

Rapport d'activité

MON TRAVAIL ET MON RÔLE DURANT MES TROIS ANNÉES D'APPRENTISSAGE À DISA

1.1 Le contexte de mon apprentissage

1.1.1 Présentation de DISA

1.1.1.1 Historique

En 1902, une firme allemande fabriquant des décalcomanies industrielles crée une succursale à Paris : « Auto décors ».

L'année 1914 mettant un terme aux relations avec l'Allemagne, M. LACARRIÈRE, alors directeur, achète un petit atelier rue Boileau, à Limoges. Huit ans plus tard, les bénéfices engendrés lui permettent de racheter le fonds de commerce de la société et il fonde la manufacture de DÉCALCOMANIES INDUSTRIELLE SA.

De 1922 à 1965, l'entreprise prospère jusqu'à devenir la première fabrique de décalcomanies en France. L'atelier rue Boileau, pourtant aménagé sur 3 étages, ne suffit plus. La société entreprend un déménagement dans la zone industrielle de Magré-Romanet, à Limoges. Au cours des trente années qui suivent, aucune évolution majeure n'est à noter. Divers travaux étendent la superficie du bâtiment à 8600 m² et l'entreprise continue de se développer. L'activité est tournée à 70 % vers l'industrie et 30 % vers le milieu publicitaire.

En 1995, à la suite d'un dépôt de bilan, Christophe RENARD, le PDG associé à Patrick TURPIN, ont repris cette entreprise de marquage industriel en difficulté, afin de disposer de leur propre solution de marquage dans le domaine de la signalétique et de l'automobile. L'ensemble du personnel, soit 130 salariés, a été repris, de même que les locaux et les machines, ainsi l'activité a pu se poursuivre dans le même secteur d'activité.

En 2006, Jean François ARCHER, directeur général de DISA, envisage de déplacer une partie de l'usine dans de nouveaux locaux. Ce processus est accéléré en 2007 par l'arrivée d'une nouvelle activité au sein de DISA exigeant une surface au sol importante, la PLV : Publicité sur Lieu de Ventes.

En 2008, le déménagement d'une partie des outils de production et du personnel se fait dans de nouveaux locaux, à 200 mètres de DISA, c'est la naissance de DISA TECHNOLOGY, qui est abrégé en DISATECH. Cette décision est prise dans le but de séparer les deux secteurs de clientèle principaux. Cela permettra d'accélérer la réactivité et d'accroître la productivité.

DISATECH est orientée vers le secteur industriel, tandis que le milieu publicitaire continue d'être géré par DISA.

En 2011, SERICA COMMUNICATION basée à Nancy rejoint le groupe DISA et complète son offre en sérigraphie grand format grâce à son parc de machines uniques. Malheureusement, la situation financière de SERICA n'est pas idéale et la société est liquidée en 2018.

En 2018, la société DISATECH est rachetée par le groupe international AKZONOBEL. Cela a pour effet la mutation de nombreux employés de DISA vers DISATECH, dont mon tuteur en entreprise, M. Florent PALIER. M. Jean François ARCHER n'est désormais plus le directeur général de DISA, il est passé directeur à DISATECH. Au sein de DISA, M^{me} Sylvie JARRAUD est devenue responsable administrative et comptable, M. Patrick MERY est maintenant directeur de la production et M. Bertrand COTTERET est directeur commercial.

1.1.1.2 Le métier d'imprimerie

La société DISA est une imprimerie de labeur, elle traite des travaux d'impression publicitaire. Certains clients demandent la création de modèles en 3 dimensions, pour de l'exposition de produits notamment, DISA est donc en mesure de proposer des produits en volume ou non.



FIGURE 1.1 – Camion en carton, PLV 3D

Ses produits sont basés sur trois techniques d'impression présentant toutes des avantages et inconvénients différents :

- **La sérigraphie** : Cette méthode est basée sur des pochoirs qui sont placés entre l'encre et le support d'impression. Cette technique permet l'utilisation de n'importe quel type de support. Les écrans utilisés en tant que pochoir offrent une taille d'impression conséquente. Cette méthode est assez lente, mais offre une qualité d'impression difficilement atteignable avec les autres méthodes.
- **L'offset** : Cette méthode nécessite la création d'une plaque en aluminium sur laquelle est gravé le motif voulu. Une fois la plaque créée, elle est installée sur un cylindre dans la machine. Celle-ci peut ensuite imprimer des feuilles à la chaîne à une vitesse extrêmement rapide. Cette méthode est donc idéale lorsqu'il faut imprimer un nombre important d'exemplaires d'un même motif. Les supports d'impression utilisés sont généralement de l'adhésif ou du PVC.
- **Le numérique** : Cette méthode utilise le même principe qu'une imprimante de bureau à jet d'encre. Ce procédé est très précis, mais assez lent (150 fois plus lent que l'offset). En revanche, contrairement aux deux autres méthodes celle-ci ne nécessite pas de travail en amont et quasiment aucune intervention de la part des ouvriers sur la machine.

Selon le cahier des charges des clients, la méthode d'impression la plus efficace est utilisée.

1.1.2 Le service informatique

Le service informatique de DISA est d'effectif très restreint et sa composition a beaucoup évolué durant mes trois années d'apprentissage.

Les deux premières années, M. Florent PALIER en était le responsable. Il fut responsable du service informatique de DISA de 2001 jusque fin 2018 avant de devenir un employé de DISATECH après son rachat par AkzoNobel. En préparation de son départ, M. Nicolas STECLEBOUTE fut recruté en septembre 2018 et formé par Florent PALIER pour devenir le nouveau responsable du service.

À mon arrivée au sein de DISA, en septembre 2016, le service comprenait M. PALIER, ainsi qu'un autre apprenti, Alexandre MEUNIER. Celui-ci était étudiant en dernière année de master à l'école 3iL Academy. Durant les mois de mai et juin 2017, le service comprit aussi un stagiaire étudiant en informatique à l'IUT de Limoges, Jeremy COUDERT. Florent PALIER avait souvent recours à des apprentis et stagiaires lorsque la charge de travail du service était plus haute.

En septembre 2017, Alexandre MEUNIER termina son apprentissage au sein de DISA, nous n'étions ensuite plus que deux avec M. Florent PALIER.

Depuis mon retour de ma mobilité en Suède, en janvier 2019, nous sommes deux au service informatique avec Nicolas STECLEBOUTE. Florent PALIER reste mon tuteur en entreprise, même s'il ne fait techniquement plus partie de DISA. Il travaille désormais de son côté pour DISATECH. Cependant il participe toujours au projet SIGMA, le projet principal du service sur ces trois dernières années, qui consiste en la création d'un nouvel ERP¹. Il est en train de préparer une version de SIGMA pleinement adapté à sa compagnie afin que cette version ne dépende plus de celle de DISA.

Les membres du service informatique s'assurent du bon fonctionnement du système d'information de la compagnie. Nous sommes responsables du matériel informatique, des télécommunications, du réseau, des serveurs, ainsi que de tous les logiciels et les applications utilisées par les employés de la compagnie. À ce travail quotidien s'ajoute le développement de nouveaux outils informatiques destinés à améliorer les conditions de travail de nos collaborateurs.

1.1.2.1 L'infrastructure matérielle

Le parc informatique de DISA est principalement constitué de PC fonctionnant sous Windows 10 ou des versions plus anciennes. Les membres du pôle d'infographie sont les seuls n'utilisant pas Windows. Ils utilisent des Mac et ont également leurs propres serveurs de stockage réservés à leur service. Chaque employé en ayant le besoin possède un téléphone relié au réseau interne de la compagnie.

En 2014, DISA a fait l'acquisition d'un mini data center, un PowerEdge VRTX, système permettant notamment la virtualisation de serveurs et le stockage de données. Ainsi, tous les serveurs de la compagnie sont centralisés sous cette machine, il regroupe :

- Quatre disques de 1To servant pour le stockage des données des serveurs partagés.
- Un serveur Windows Server 2010 qui héberge Sage, le logiciel servant à la comptabilité.
- Un serveur sous Linux permettant la réalisation de tâches planifiées.
- Un serveur web hébergeant l'intranet.
- Un serveur de test pour le développement du nouvel ERP SIGMA.
- Un serveur de production pour le déploiement et l'utilisation de SIGMA.

Toutes les données sur les disques du VRTX sont protégées par un système RAID 5, ce système répartit les données sur plusieurs disques de manière à pouvoir récupérer toutes les informations même si un disque venait à tomber en panne.

Pour le moment, la compagnie DISATECH est encore reliée au réseau de DISA par l'intermédiaire d'un canon Wi-Fi, car l'accès au serveur de stockage de DISA leur est encore nécessaire. Cependant dès lors que le réseau et

1. ERP : Enterprise Ressource Planning, détaillé dans la partie « Le projet SIGMA »

la gestion des serveurs du côté de DISATECH seront entièrement opérationnels et indépendants, cette connexion n'aura plus lieu d'être et sera supprimée.

1.1.2.2 L'environnement logiciel de DISA

L'environnement logiciel de la compagnie a évolué durant mon apprentissage et est encore en cours de mutation. Nous sommes en train de développer un nouvel ERP pour l'entreprise, nommé SIGMA, l'objectif de ce projet est de fournir un outil unique aux employés afin de faciliter les liens entre chaque étape de la fabrication.

Actuellement, les employés utilisent principalement SIGMA pour leur travail. En parallèle, ils utilisent encore ASAP (l'ancien ERP qui vise à être remplacé) pour effectuer des tâches qui n'ont pas encore été implémentées sous SIGMA. L'intranet de l'entreprise, DISANET, est encore utilisé par certains employés au même titre qu'ASAP, afin d'effectuer des tâches pour le moment non réalisables sous SIGMA. Pour la comptabilité, le logiciel SAGE est utilisé, celui-ci est un logiciel propriétaire, nous ne pouvons donc pas le modifier.

- **ASAP :** C'est l'ERP utilisé à DISA depuis l'année 2002, jusqu'au passage sous SIGMA. Permettant la gestion complète de tous les flux internes de DISA, il est devenu rapidement indispensable pour gérer la production dans de bonnes conditions depuis son installation.
Cependant, avec le temps, des besoins non gérés par ce dernier sont apparus et c'est pourquoi en 2006 un intranet fut créé.
- **DISANET :** C'est l'intranet de l'entreprise, il fut créé en 2006 par un stagiaire de l'école 3iL. Dans un premier temps, DISANET fut vu comme une opportunité permettant de combler les manques d'ASAP. C'est pour cela que, depuis sa création, une centaine d'applications ont été développées sur ce support : il est donc devenu un complément indispensable d'ASAP. Désormais, le nombre d'applications sur DISANET est tel que les employés sont constamment amenés à passer de DISANET à ASAP et inversement, ce qui ne facilite pas leur travail.
C'est pour cette raison que M. PALIER a décidé de commencer le projet SIGMA en 2016. Ce dernier vise à centraliser toutes les applications métiers, actuellement éparses entre l'ERP et l'intranet, au sein d'un seul et unique outil. M. PALIER souhaite donc que l'intranet retrouve un rôle de communication plutôt qu'un rôle applicatif et que nous cessions d'utiliser ASAP.
- **SIGMA :** Depuis la fin de l'année 2018, les employés ont commencé à utiliser ce nouvel ERP quotidiennement. Le passage à l'utilisation de celui-ci s'est fait progressivement afin d'avoir le temps de former les employés à l'utilisation de l'outil et de ne pas générer une affluence de problèmes à résoudre trop importante. Cette migration a continué durant quelques mois. Désormais, les employés ayant encore besoin d'utiliser ASAP et DISANET sont rares. Il reste encore des applications à développer et intégrer sur SIGMA et d'autres à corriger et/ou améliorer (ce travail est en cours).
- **SFAO :** Avant l'utilisation de SIGMA, les ouvriers utilisaient la SFAO : Suivi de Fabrication Assisté par Ordinateur. Cet outil permettait de superviser la production et de garder un historique de la fabrication. Concrètement, il permettait de savoir quel salarié avait travaillé sur chaque tâche de fabrication. Il permettait aussi de récupérer des informations permettant de générer des statistiques sur le processus de production et de gérer plus facilement les soucis rencontrés par les ouvriers.
Cet outil est désormais entièrement intégré au sein de SIGMA, l'ancien système n'est plus utilisé du tout.

1.1.2.3 Mon rôle au sein du service

Durant mes trois années d'apprentissage, la tâche principale qui m'a été confiée est la participation à la conception, au développement et à la mise en place du projet SIGMA. C'est un projet de grande ampleur, et travailler sur les différentes phases de celui-ci m'a permis de découvrir de nombreux aspects différents du travail de développeur et de chef de projet.

Mon quotidien fut globalement découpé entre l'analyse des besoins des futurs utilisateurs de SIGMA et le

développement d'applications adaptées pour leur permettre de travailler le plus simplement possible. Une fois la mise à disposition de SIGMA aux employés, s'en est suivi une période focalisée sur la résolution de bugs et l'amélioration des applications afin de les faire mieux correspondre aux besoins de chacun.

En plus de ce travail quotidien, je porte aussi assistance aux employés lorsqu'ils rencontrent des problèmes informatiques.

L'effectif très réduit du service oblige souvent à être confronté à des problèmes jamais rencontrés auparavant et sollicite la mise en application de compétences variées : cela favorise la polyvalence.

1.2 Le projet SIGMA

1.2.1 Pourquoi ce projet ?

Le mode de fonctionnement utilisé au sein de la compagnie depuis la création de DISANET présentait plusieurs défauts. En effet, certaines applications de l'intranet utilisent des bases de données indépendantes. En plus de cela, il n'était pas toujours possible de travailler sur la base de données d'ASAP depuis l'intranet et inversement. Cela rendait les liens complexes et obligeait les employés à saisir plusieurs fois certaines données.

De plus, ASAP est un système vieillissant qui se montrait de moins en moins adapté aux besoins modernes de DISA. Compenser ses défauts par la création d'applications sous DISANET n'était donc plus une solution viable. Le principal défaut de cet ERP résidait dans le fait qu'il était fourni par une société tierce. Car, bien qu'il soit possible d'effectuer des modifications, et donc d'ajouter du contenu à ASAP, c'est une solution non viable à long terme, notamment à cause de la gestion des mises à jour et de l'ancienneté du code déjà existant.

L'entreprise était donc dans le besoin d'un nouvel outil, plus performant, plus rapide et plus simple d'utilisation. Pour répondre à cette problématique, M. PALIER avait deux principales solutions :

- La première étant de recourir à un ERP développé par une entreprise tierce. Le souci étant que cela voulait dire se confronter à nouveau aux mêmes problèmes qu'avec ASAP sur le long terme.
- La seconde solution étant de développer en interne un nouvel ERP parfaitement adapté aux besoins de chacun. Le principal défaut de cette solution étant bien sûr, une charge de travail conséquente pour le service informatique pour les années à venir.

En 2016, Florent PALIER décida de développer le nouvel ERP en interne afin de pouvoir s'assurer que celui-ci soit adapté aux besoins spécifiques des employés et qu'il soit possible de le faire évoluer facilement.

Le projet fut nommé SIGMA. L'objectif était donc de créer un outil unique permettant de regrouper tous les besoins des employés et ainsi remplacer ASAP, toutes les applications de DISANET et la SFAO. Cet outil se reposera sur une unique base de données, ainsi les liens entre chaque application seront faits automatiquement et le suivi d'un même dossier au sein de la compagnie pourra être effectué sans obliger à la saisie redondante d'informations.

Le fait que tous les employés travaillent tous sur le même outil est aussi beaucoup plus simple à gérer d'un point de vue informatique, une seule technologie est utilisée pour le développement, ce qui rend la maintenance plus simple.

1.2.2 Environnement de développement

1.2.2.1 Outils utilisés

Pour ce projet, M. Palier a décidé de travailler sous WinDev. WinDev est un atelier de génie logiciel qui utilise son propre langage de programmation : le Wlanguage. C'est un langage de quatrième génération, ce qui signifie qu'il est assez proche d'une langue naturelle. Cela permet de développer des applications bien plus rapidement qu'avec des langages plus classiques tels que le PHP. L'utilisateur de WinDev peut également se passer de la programmation pour les fonctionnalités les plus simples. Par exemple, pour la création d'une liste déroulante dont les données proviennent d'un fichier de la base de données, il suffit de renseigner la source de données et le lien se fait automatiquement.

WinDev présente une interface graphique sur laquelle il est possible de placer de nombreux éléments (Boutons, champs de saisie, sélecteurs, tables, etc.) par drag and drop². Il est donc très simple et rapide de créer des mock-ups³ afin de se représenter une fenêtre future et de pouvoir la présenter aux futurs utilisateurs.

Afin d'organiser le développement entre plusieurs développeurs et d'éviter les conflits de versions de fichiers lors de modifications en simultanés, nous utilisons le gestionnaire de sources intégré à WinDev. Celui-ci permet

2. Glisser-déposer.

3. Maquette d'une interface utilisateur servant de support pour une présentation.

d'extraire un élément du projet. Lorsqu'un élément est extrait en mode exclusif les autres développeurs ne peuvent pas effectuer de modifications sur le fichier. Lorsque nous avons terminé nos modifications au sein du fichier, nous réintégrons l'élément au projet, les autres développeurs peuvent alors utiliser l'élément modifié. Il en va de même pour la gestion de la base de données, il est nécessaire d'extraire la base avant de pouvoir appliquer des modifications. Un tel gestionnaire permet de développer sans conflit de versions de fichiers. Il garde également un historique des accès et de toutes les modifications effectuées sur n'importe quel fichier, permettant un retour en arrière instantané si nécessaire.

1.2.2.2 Architecture client lourd-serveur

Le modèle client-serveur est une architecture d'application utilisant un réseau pour l'échange de données. Il permet de répartir les traitements et le stockage des données sur les serveurs et/ou les clients. Dans cette architecture, un client est l'entité que représente un utilisateur utilisant l'application.

C'est ce modèle qui fut choisi pour l'ERP SIGMA, les données sont enregistrées sur un serveur dédié à cet effet, sur lequel est installé une base de données HFSQL⁴. L'avantage principal du modèle client-serveur est qu'il permet de n'enregistrer aucune donnée sur les postes client. Cela permet aux utilisateurs de l'application de pouvoir se connecter sur leur session depuis n'importe quel terminal et de retrouver exactement les mêmes informations. C'est notamment un très gros avantage pour les ouvriers utilisant la SFAO intégrée à SIGMA lorsqu'ils sont amenés à se déplacer sur un autre poste dans l'usine.

Cette architecture permet aussi une gestion des mises à jour simple. Lorsque l'application est démarrée sur un poste, la version de celle-ci est comparée avec celle du référentiel stocké sur le serveur. Si les deux versions ne coïncident pas, l'application locale du poste est mise à jour automatiquement.

De plus nous utilisons aussi une architecture client lourd, cela signifie que les calculs nécessaires à l'utilisation de l'outil sont effectués sur le poste de l'utilisateur et non sur le serveur. L'avantage est évidemment de soulager la charge de travail du serveur, sachant que les ordinateurs des employés utilisés à DISA sont récents et bien assez performants pour supporter cette charge.

1.2.3 La conception théorique de SIGMA

Avant d'amorcer le développement du nouvel ERP, le service informatique s'est concentré sur sa conception. Il fallait déterminer une approche viable pour que la réalisation d'un tel projet ne se termine pas par un échec, ce qui est courant pour un projet d'une telle ampleur. Ce travail débuta en juin 2016, le service informatique était alors constitué de M. Florent PALIER, Clément MARTAILLE, apprenti ingénieur et d'Alexandre MEUNIER, apprenti en master.

1.2.3.1 Récolte et analyse des besoins des futurs utilisateurs

La première étape fut de récolter, puis d'analyser les besoins des divers employés de DISA. Lors de cette étape, il est crucial de tenter de mettre en situation les employés afin d'obtenir une formulation du besoin la plus concrète possible. Il faut donc revenir vers eux plusieurs fois pour leur poser des questions diverses sur leur travail quotidien afin de s'assurer que leurs idées et besoins n'entrent pas en conflit. Si l'on détecte des incohérences à ce moment, il est indispensable de les faire remarquer aux employés afin de réfléchir à une solution convenable. Cette étape est indispensable, car il arrive souvent que les attentes d'une personne soient assez abstraites et que l'on s'aperçoive à la fin du développement que le produit conçu ne leur convient pas.

M. PALIER s'est entretenu avec les responsables de chaque service de l'entreprise afin qu'ils lui fassent part de leurs besoins et de leurs attentes quant à ce nouvel outil. L'objectif n'était pas de générer un énorme cahier des charges, mais plutôt de noter un maximum d'idées et de suggestions.

Après l'annonce du développement d'un nouvel ERP, de nombreux employés firent parvenir des requêtes au service informatique. Clément MARTAILLE et Alexandre MEUNIER développèrent alors un outil de type « boîte à idées » sur l'intranet pour que chacun puisse y inscrire ses demandes.

4. Le type HFSQL, ou HyperFileSQL, est le type de base de données utilisé par WinDev.

1.2.3.2 Choix de la méthode de travail

Pour le développement d'un projet aussi vaste, la création d'un cahier des charges complet et final demanderait trop de temps. Pour cette raison, et afin de pouvoir répondre aux besoins de chacun de la manière la plus souple possible, il fut décidé d'adopter une démarche itérative incrémentale.

Cette démarche est fortement inspirée des méthodes agiles : elle combine le développement itératif au développement incrémental dans l'objectif de ne garder que les côtés positifs des deux.

- Le développement incrémental s'appuie sur une idée parfaitement définie, on construit ensuite le produit final morceau par morceau. On ne peut présenter le produit qu'une fois terminé puisque celui-ci n'est pas fonctionnel avant le dernier moment. En théorie, cette méthode est simple et fonctionnelle, cependant dans la réalité, comme précisé plus haut, les besoins des utilisateurs ne sont jamais aussi clairs qu'on le pense. Sachant cela on peut vite comprendre que présenter le produit à l'utilisateur seulement lorsque celui-ci est terminé à 100% n'est pas une idée viable.
- Le développement itératif part d'une idée grossière à partir de laquelle on peut construire un mock-up à présenter à l'utilisateur. Si la maquette ne plaît pas à l'utilisateur, on peut alors la modifier afin de l'adapter à ses besoins. Ensuite, ce processus recommence jusqu'à l'obtention d'un produit satisfaisant. Cette méthode est bien plus réalisable dans la pratique, elle met l'accent sur l'approbation de l'utilisateur.

En alliant ces deux méthodes, on obtient une démarche permettant de se baser sur les retours des utilisateurs tout en prenant des libertés afin de limiter leur implication. Cette démarche est aussi très similaire à la méthode XP, Extreme Programming, celle-ci est particulièrement orientée sur l'aspect réalisation du projet sans pour autant négliger l'aspect gestion de projet. Cette méthode, tout comme la méthode itérative incrémentale, est adaptée aux équipes réduites avec des besoins changeants et s'appuie beaucoup sur le retour des futurs utilisateurs.

Pour pouvoir mettre en place cette méthode, le développement de SIGMA doit donc s'effectuer sous forme de modules, chaque module étant composé de plusieurs applications. Cela permet de récolter les besoins des utilisateurs de manière quasiment individuelle et de pouvoir se consacrer pleinement aux besoins de quelques employés en même temps tout au plus.

Un désavantage de cette méthode est que l'on développe les applications les unes après les autres, il faut donc faire attention à anticiper les liens et besoins que de futurs applications ou modules pourront avoir avec les applications en cours de développement. L'étape de récupération des besoins de manière globale, réalisée par M. PALIER, fut très utile pour ce genre d'anticipation, si elle n'avait pas été effectuée nous aurions eu de gros soucis et de nombreuses applications à modifier pour prendre en compte des besoins non anticipés.

1.2.3.3 Proposition des fonctionnalités

Après la phase de récupération des besoins et le choix de la méthode de travail, M. PALIER et les apprentis proposèrent une liste non exhaustive des modules à développer.

- Gestion commerciale
 - Gestion des clients et des prospects (CRM : gestion de la relation client)
 - Gestion des commandes client
 - Gestion des devis
- Gestion des achats
- Gestion des ressources humaines
- Données techniques
- Outils comptables
- Suivi de la production (SFAO et statistiques)
- Gestion des stocks et inventaires

- Logistique
- Gestion des processus qualité
- Maintenance

1.3 Le développement de SIGMA

1.3.1 Crédit de la structure de la base de données

Avant de commencer le développement, Clément MARTAILLE et Alexandre MEUNIER commencèrent par concevoir la base de données. L'idée était de créer la structure générale de la base de données afin de visualiser plus facilement l'ensemble du projet et de pouvoir déterminer les priorités dans le développement. En effet, une fois la structure globale de la base construite, il est possible de voir les tables de la base qui seront utilisées par le plus d'applications et modules différents. Ce sont ces applications qui devront être développées en premières et pour lesquelles il faudra porter une attention particulière à ce qu'elles soient simples et stables, étant donné qu'elles serviront de base au reste du projet.

Une fois la structure de la base de données terminée, il fut déterminé que le premier module à développer serait celui de la gestion des clients, le CRM. Comme dit auparavant, ce choix fut pris en raison du fait que la gestion des clients, de leurs adresses, des contacts, etc., représente le cœur du projet.

1.3.2 Application de la méthode itérative incrémentale

Pour chaque application de SIGMA, nous appliquons la même méthode de travail, sa mise en pratique respecte toujours les mêmes grandes étapes.

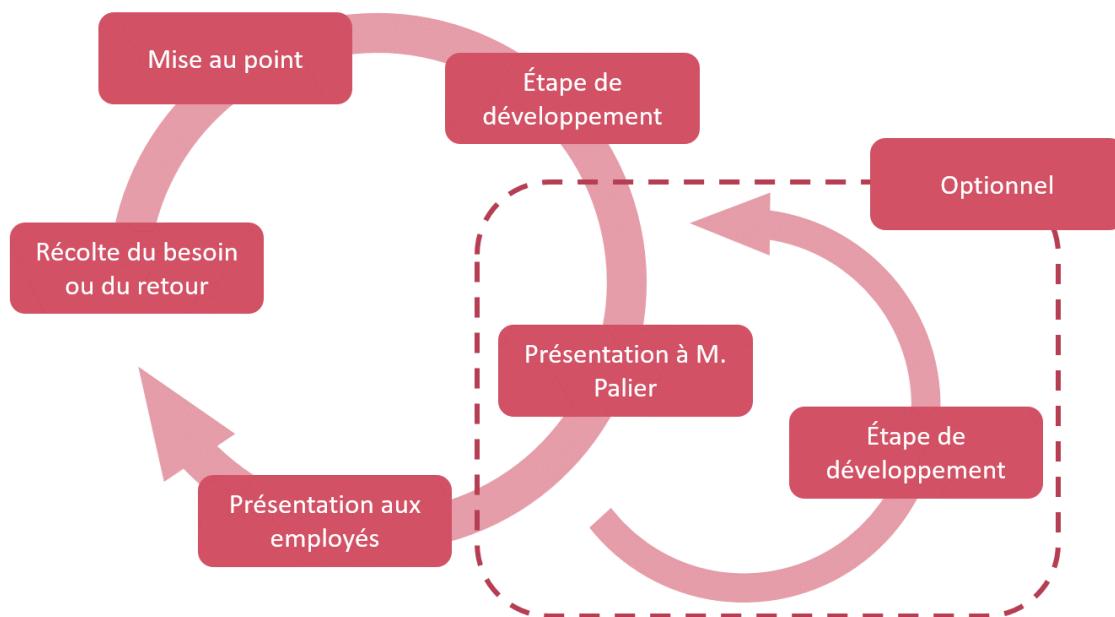


FIGURE 1.2 – Cycle de développement

La première étape est la récolte du besoin. Dans un premier temps M. PALIER, s'entretenait avec les futurs utilisateurs pour recueillir leurs besoins et me les retransmettre de manière claire et plus adaptée pour le développement. Au cours de l'avancement de mon apprentissage, il me laissa plus de responsabilités vis-à-vis de la récolte et de l'analyse des besoins. Désormais, M. PALIER suit toujours l'avancement de mon travail, mais il me laisse m'entretenir avec les employés et responsables de service lorsque j'en ai besoin.

Une fois le besoin éclairci, si nécessaire, je m'entretiens avec M. PALIER pour discuter des potentielles modifications à apporter à la base de données et également des modifications et ajouts nécessaires au niveau

des applications de SIGMA. Cette mise au point permet de s'assurer que le développement à prévoir n'entre pas en conflit avec un autre développement passé, en cours, ou futur.

Suite à cette mise au point, je commence les modifications de la base de données et la création des mock-ups des différentes fenêtres à développer. Ces mock-ups permettent de présenter les avancées aux futurs utilisateurs en leur expliquant de manière concrète les fonctionnalités imaginées. Les futurs utilisateurs peuvent alors se faire une idée claire du résultat futur et nous faire part de leurs retours. Au début de mon apprentissage je présentais constamment les mock-ups à M. PALIER pour avoir son retour avant de les présenter aux employés, mais désormais il est courant que je les présente directement aux utilisateurs finaux. Cela permet d'économiser du temps, ce qui est d'autant plus vrai maintenant que celui-ci n'est plus présent dans les locaux de DISA.

Une fois les maquettes présentées, nous utilisons le retour afin de les améliorer et de commencer à implémenter les fonctionnalités imaginées. C'est un cycle de développement qui se répète jusqu'à l'obtention d'un produit final satisfaisant pour les utilisateurs comme pour le développeur. Ce cycle est basé sur le retour des utilisateurs ce qui permet de s'assurer un développement dans la bonne direction et ainsi garantir un produit adapté à leurs besoins.

1.3.3 Mise en production progressive de SIGMA

La mise en production de SIGMA s'est effectuée très progressivement et commença dès le début de l'année 2018.

Dans un premier temps, nous avons installé l'application sur les postes des responsables projets et nous les avons formés à l'utilisation de l'outil d'édition de devis. Cette première étape de mise en production nous a permis de commencer la formation de certains employés et surtout d'obtenir des retours concrets, et ainsi, de corriger de nombreux défauts.

Nous avons ensuite suivi cette même méthodologie pour la suite de la mise en production. Cela nous a permis de former les employés à l'utilisation des parties de SIGMA qui leur sont dédiés, le tout sans être débordés. Un gros avantage avec cette méthode est que les retours des utilisateurs se font progressivement, ce qui nous laisse le temps d'apporter des modifications et de nous assurer que celles-ci conviennent pour tous les utilisateurs.

À l'heure actuelle, tous les employés de DISA utilisent SIGMA et ils sont formés à l'utilisation de leurs applications dédiées. Nous nous concentrons sur la résolution des bugs qui se font de moins en moins présents, et surtout sur l'amélioration de certaines fonctionnalités sur la demande des utilisateurs. Il y a encore plusieurs applications à créer sous SIGMA afin de pouvoir remplacer complètement ASAP et DISANET.

1.4 Mes développements notables sur SIGMA

Dans cette partie, je vais évoquer certains développements intéressants sur lesquels j'ai travaillé durant mon apprentissage au sein de DISA.

1.4.1 Configurateur de devis

1.4.1.1 Le besoin exprimé par les deviseurs

Nous avons développé, au sein de SIGMA, un module permettant aux deviseurs de créer des devis. Le développement de ce module fut long et complexe, il était déjà en cours lors de mon arrivée à DISA et se poursuivit encore sur plusieurs mois. À terme, l'outil de génération des devis créé était fonctionnel et permettait bien de créer des devis. Cependant, après une présentation de cet outil aux deviseurs, ceux-ci nous ont demandé s'il était possible de leur offrir un outil permettant de créer une version simplifiée d'un devis, contenant seulement les informations minimales. Le but de cet outil serait de permettre la création de devis de manière simplifiée et plus rapide.

Les devis créés par les deviseurs sont souvent très semblables et c'est donc une perte de temps pour eux d'avoir à saisir les mêmes informations pour plusieurs devis différents. Le but était donc de garder le module initial pour la création de devis complexes et d'utiliser un nouvel outil pour la génération de devis simples, composés d'opérations similaires.

Afin de concrétiser leurs idées, nous avons demandé aux deviseurs de créer un fichier EXCEL représentant leur vision de ce configurateur. Avec M. PALIER, nous avons relevé de nombreux points qui manquaient de clarté, un certain nombre d'incohérences et surtout des fonctionnalités indispensables qui avaient été omises dans ce document. Nous nous sommes donc entretenus de nombreuses fois avec les deviseurs afin de corriger et améliorer cette ébauche. Nous sommes finalement arrivés à une version qui nous semblait réalisable sous SIGMA.

1.4.1.2 La solution apportée

J'ai effectué la conception et le développement de ce configurateur lors de ma seconde mission en entreprise. L'objectif du configurateur de devis était de simplifier le travail des deviseurs. J'ai donc fait en sorte que celui-ci ne comporte qu'une seule fenêtre et se présente de la manière la plus claire possible.

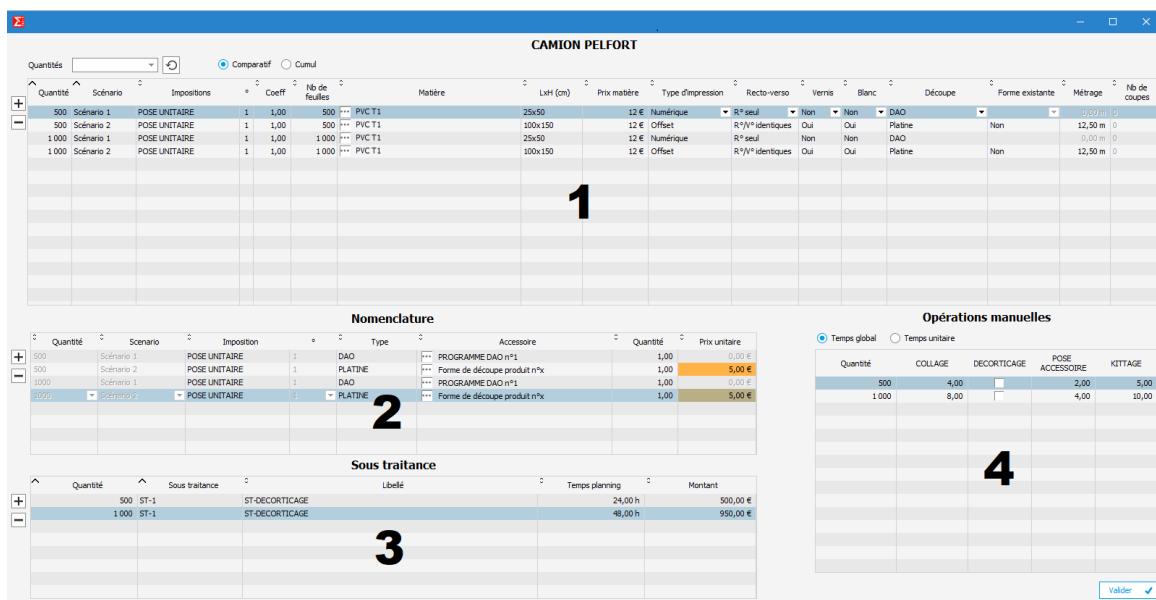
Le configurateur ne devait gérer que les cas de devis composés uniquement d'impositions constituées d'une pose unitaire. Une imposition constituée d'une pose unitaire est le cas le plus simple existant. La pose unitaire signifie que sur chaque feuille imprimée il n'y a qu'un seul motif, celui-ci est répété sur toutes les feuilles. Les autres cas sont les multiposes et les amalgames : les multiposes consistent en l'impression d'un même motif en plusieurs exemplaires sur chaque feuille et les amalgames consistent en l'impression de plusieurs motifs différents sur chaque feuille.

Le configurateur est constitué de quatre tableaux.

1. Le premier tableau permet de gérer les impositions. On peut en ajouter ou supprimer, on peut sélectionner le type d'impression, la matière d'impression, le format et plusieurs autres informations. La saisie des informations au sein de ce premier tableau entraîne l'ajout automatique de plusieurs opérations lors de la validation du configurateur. La saisie entraîne aussi l'ajout automatique de lignes prérenseignées dans le second tableau.
2. Le second tableau permet de gérer la nomenclature du devis. La nomenclature du devis permet de renseigner la liste des outils, matières, encres et accessoires qui seront utilisés lors des étapes de production. La majorité des lignes de ce tableau sont ajoutées automatiquement lors de la saisie des informations du premier tableau. L'utilisateur peut tout de même ajouter manuellement des lignes au sein de la nomenclature s'il le souhaite.

3. Le troisième tableau permet de gérer les phases de sous-traitance. Toutes les lignes de cette table doivent être générées manuellement, elles permettent d'indiquer si certaines opérations du devis doivent être sous-traitées. Il faut donc renseigner le temps et le prix de chaque sous-traitance.
4. Le quatrième tableau permet de gérer les opérations manuelles. Ce sont des opérations dites de « kit-tage », elles sont réalisées après les phases d'impression et permettent le conditionnement du produit en préparation de sa livraison au client. Ici, l'utilisateur ne peut pas ajouter ou supprimer de lignes, mais seulement renseigner une estimation des temps que chacune de ces opérations prendra.

Sur l'image ci-dessous on peut voir la fenêtre du configurateur telle qu'elle l'était à ce moment du développement.



The screenshot displays a software interface for a configurator tool, specifically for a 'CAMION PELFORT'. The interface is divided into four main sections, each with a large number indicating its corresponding step from the list above:

- Section 1 (Top Left):** A table titled 'CAMION PELFORT' showing material details. It includes columns for Quantité, Scénario, Imposition, Coeff, Nb de feuilles, Matière, LxH (cm), Prix matière, Type d'impression, Recto-verso, Vernis, Blanc, Découpe, Forme existante, Métrage, and Nb de coupes. Data rows include '500 Scénario 1 POSE UNITAIRE' and '1000 Scénario 2 POSE UNITAIRE'.
- Section 2 (Bottom Left):** A table titled 'Nomenclature' showing parts details. It includes columns for Quantité, Scenario, Imposition, Type, Accessoire, Quantité, and Prix unitaire. Data rows include '500 Scénario 1 POSE UNITAIRE DAO', '500 Scénario 2 POSE UNITAIRE PLATINE', '1000 Scénario 1 POSE UNITAIRE DAO', and '1000 Scénario 2 POSE UNITAIRE PLATINE'.
- Section 3 (Bottom Center):** A table titled 'Sous traitance' showing subcontracting details. It includes columns for Quantité, Sous traiteuse, Libellé, Temps planning, and Montant. Data rows include '300 ST-1 ST-DECORTICAGE' and '1000 ST-1 ST-DECORTICAGE'.
- Section 4 (Bottom Right):** A table titled 'Opérations manuelles' showing manual operation details. It includes columns for Quantité, Temps global, Temps unitaire, COLLAGE, DECORTICAGE, POSE ACCESSOIRE, and KITTING. Data rows include '500 4,00 2,00 5,00' and '1000 8,00 4,00 10,00'.

FIGURE 1.3 – Configurateur de devis avec les tableaux numérotés

La partie la plus complexe et laborieuse du développement fut pour gérer l'étape de validation du configurateur. En effet, la fenêtre est simple, comporte peu d'informations et permet de simplifier grandement le travail des deviseurs, en revanche cela complexifie beaucoup le traitement d'enregistrement des informations en base de données. Afin de générer un devis à partir des informations présentes dans les quatre tableaux, il faut renseigner une quinzaine de tables en base de données, en prenant garde à bien relier tous les éléments entre eux.

Pour un tel travail de développement, il est primordial de s'organiser afin de ne pas perdre le fil. Sans organisation il n'est pas rare de se lancer dans un long développement de plusieurs heures, voire jours et, à terme, de ne plus comprendre comment son code fonctionne et de devoir tout recommencer plus méthodiquement.

Afin d'aborder ce développement, j'ai donc schématisé toutes les relations de la base de données sur un cahier, en préparant ainsi par la même occasion l'ordre de chaque import en base de données. Car en plus de devoir relier tous les enregistrements entre eux, il faut aussi faire attention à les ajouter dans le bon ordre afin de garantir une cohérence des données. Une fois ce travail effectué, j'ai pu développer le traitement de validation, j'ai également commenté proprement le code afin de permettre une reprise et modification simple du code à l'avenir.

Je suis parvenu à terminer le développement du configurateur avant la fin de ma seconde mission. L'outil était fonctionnel et répondait au besoin des deviseurs tel qu'il fut décrit initialement.

1.4.1.3 Les raisons de l'échec et abandon de ce projet

Lors de mon retour pour la troisième mission en entreprise, on m'annonça que le configurateur de devis était peu utilisé, car de nombreux cas très courants n'étaient pas gérés. Ces cas étaient jugés complexes et c'est pourquoi ils n'avaient pas été pris en compte pour le configurateur, puisque l'objectif n'était de l'utiliser que pour les cas courants et simples.

Les deviseurs avaient donc surestimé le taux de réalisation de devis simples et nous n'étions pas parvenus à détecter ce problème plus tôt, avec M. PALIER. Nous étions tout d'abord opposés à l'idée de modifier le configurateur pour prendre en compte ces cas complexes, mais nous sommes parvenu au constat que la grande majorité des devis ne sont finalement pas composés de cas simples, au contraire. Par conséquent, bien que fonctionnel, l'outil n'était pas assez adapté aux besoins des deviseurs et n'était donc pas utilisé.

Nous avons donc décidé de prendre en compte plus de cas de figure. Parmi les cas de figure supplémentaires qu'il fallait gérer se trouvait la gestion des multiposes ainsi que des amalgames. La gestion de tous les nouveaux cas de figure complexifia énormément le traitement de génération des lignes de nomenclatures automatique. Cela complexifia par la même occasion le traitement des enregistrements en base de données.

Le principal souci causé par l'ajout de tous ces cas de figure est que nous avions spécifiquement décidé de ne pas les gérer. La totalité de la conception et du développement de l'outil a donc volontairement été effectuée sans les prendre en compte. La modification du configurateur s'est donc effectuée dans de mauvaises conditions, il était impossible d'ajouter les fonctionnalités demandées sans entraîner de grosses modifications au traitement de base.

Malgré les tentatives, il fut impossible d'ajouter les fonctionnalités sans provoquer des bugs empêchant totalement l'utilisation du configurateur. Il a alors fallu refuser de faire ces modifications.

Le configurateur ne répondant toujours pas aux besoins des deviseurs, nous avons préféré mettre le projet de côté afin de ne pas perdre plus de temps.

1.4.1.4 Les solutions compensatoires développées

Même si le configurateur de devis ne fut pas un succès, nous avons tout de même développé plusieurs fonctionnalités permettant de simplifier le travail des deviseurs.

Parmi ces fonctionnalités, ils ont la possibilité de créer des devis à partir d'une copie de la gamme⁵ ou de la nomenclature d'un autre devis. Ainsi, cela leur permet de créer très rapidement tous les devis similaires. Nous avons aussi généré des gammes et nomenclatures type qui correspondent à la majorité des cas de devis. Celles-ci servent de modèle au devis et nécessitent peu de modifications de la part des deviseurs afin de les adapter aux besoins du client.

Bien que le configurateur fut un échec et que nous avons ajouté des fonctionnalités permettant de soulager le travail des deviseurs, l'idée de la création d'un nouveau configurateur est toujours présente.

1.4.1.5 Leçons tirées de cette expérience

L'échec du développement du configurateur de devis est principalement dû à la compréhension seulement partielle du besoin des utilisateurs et à une étape de récolte du besoin qui fut complexe. Le développement aurait pu être rendu plus simple si nous nous étions rendu compte plus tôt que la gestion des cas jugés simples n'allait pas suffire à satisfaire les besoins des deviseurs.

Étonnamment, cet échec est le seul que j'ai rencontré lors de mon apprentissage au sein de DISA, alors que c'est un module pour lequel nous portions justement une attention toute particulière à la compréhension du besoin. Nous savions, avec M. PALIER, que la réalisation d'un tel configurateur était ambitieuse et allait être laborieuse et nous voulions alors nous assurer de bien concevoir l'outil, afin de ne pas nous retrouver dans une mauvaise situation par la suite. C'est pourtant exactement ce qui est arrivé.

5. La gamme d'un devis est la liste de toutes les opérations de fabrication à effectuer.

Cela montre que même lorsque l'on pense avoir assez clarifié les besoins des utilisateurs, il est nécessaire de prendre ses précautions et de s'entretenir avec eux régulièrement, afin de leur faire valider l'avancement du développement et de pouvoir déceler les éventuelles divergences entre le produit développé et le besoin réel.

1.4.2 Planning de production

1.4.2.1 Le besoin d'informatiser le processus de planification

Afin de planifier la production de l'entreprise, les responsables de la production utilisaient depuis de nombreuses années de grands tableaux permettant d'accueillir des papiers sur lesquels étaient indiquées les informations des phases de fabrication. Ces tableaux permettaient une organisation de toutes les phases, par machine et par jour, ainsi, lorsqu'un ouvrier voulait savoir ce qu'il avait à faire, il devait venir voir les phases sur le tableau et ensuite récupérer le dossier papier correspondant à l'OF⁶ en question, puis retourner à sa machine.



FIGURE 1.4 – Exemple d'une petite partie d'un tableau de production

Avec le projet SIGMA, M. Palier entendait supprimer au maximum les besoins d'impression des documents de fabrication. Chaque dossier de fabrication pouvait rapidement atteindre entre 10 et 20 pages, le nombre de dossiers pouvant aller jusqu'à plusieurs dizaines par jours, la quantité de papier et d'encre utilisée était énorme.

Il fallait donc remédier à cette consommation excessive de papier et d'encre. Cela passe par le développement de deux outils : le premier est l'outil de planning de production et le second est le développement de la SFAO sous SIGMA, décrit dans la prochaine section.

1.4.2.2 Le développement du planning

J'ai effectué le développement du planning lors de ma quatrième mission en entreprise, le but de ce planning était de remplacer les tableaux physiques. Nous avons décidé de développer un planning sur lequel les responsables de la production pourraient organiser toutes les phases de fabrication. En parallèle, il fallait développer la SFAO en s'assurant que les deux outils puissent fonctionner ensemble et en même temps. L'organisation des phases de production doit, par exemple, entraîner en direct l'affichage des phases à réaliser pour chaque machine sur la SFAO. Remplacer le tableau physique a permis de simplifier le travail des ouvriers qui n'ont plus à traverser toute l'usine pour prendre connaissance des prochaines tâches à effectuer.

Nous avons décidé de présenter le planning de fabrication de manière assez similaire au tableau physique, c'est-à-dire les machines sur la gauche et les jours de la semaine en haut. Les utilisateurs peuvent glisser les phases depuis un tableau sur la droite directement à l'endroit souhaité dans le planning. L'ordonnancement des phases se fait manuellement sur le planning.

6. Ordre de fabrication, document présentant la liste des phases de fabrication

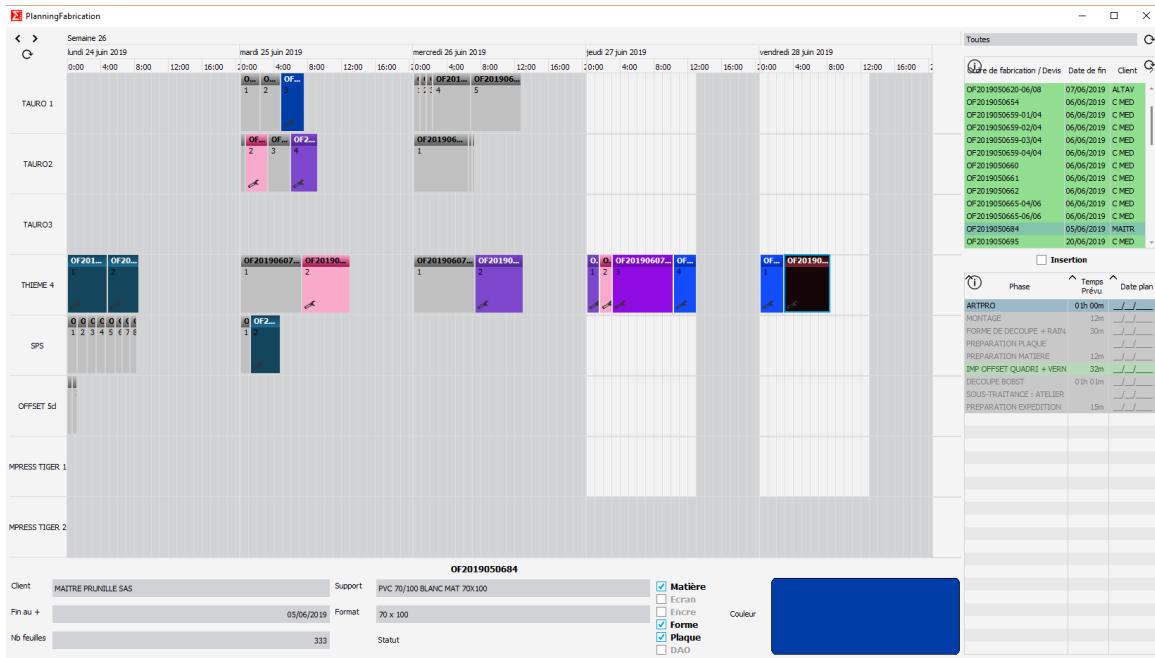


FIGURE 1.5 – Aperçu du planning de fabrication intégré à SIGMA

Bien que le planning soit très simple d'utilisation, ce qui fut long dans le développement de celui-ci fut la création de tous les traitements permettant une organisation fluide des phases au sein du planning. Car, lorsque l'on souhaite déplacer une phase il faut, en arrière-plan, gérer la réorganisation de toutes les autres phases sur cette machine. Il faut aussi prendre en compte les heures d'utilisation des machines qui peuvent varier d'un jour à un autre, etc. Il est important de toujours garder une cohérence au niveau de la planification des phases puisque celles-ci sont ensuite affichées à l'écran de la SFAO pour les ouvriers.

En plus de planifier les OF en cours, dans un souci d'anticipation, nous avons donné la possibilité de planifier les devis les plus susceptibles d'aboutir et de donner lieu à une fabrication. Ainsi, les responsables de la production ont une vision plus précise sur la charge de travail des semaines à venir.

Grâce à la SFAO, les ouvriers peuvent signaler informatiquement lorsqu'il y a un souci sur une machine, la machine en question apparaît alors en rouge sur le planning. Ainsi, il est possible de réagir rapidement pour réorganiser les phases sur d'autres machines par exemple.

Afin de permettre un tri des OF en cours, et ainsi, permettre une organisation plus simple, j'ai développé une seconde fenêtre qui fonctionne de pair avec le planning. Les postes des responsables de production, possédant deux écrans, cela leur permet d'avoir le planning sur l'un et la fenêtre de tri sur l'autre. La fenêtre de tri a pour objectif de servir de détail sur les OF à planifier. En effet, dans la fenêtre de planning il y a peu de place pour afficher plus de détails alors que la seconde permet d'afficher les détails concernant l'avancement des phases de préparation. Tous les OF possèdent des phases de préparation et celles-ci doivent être terminées avant de pouvoir lancer en production les phases d'impression.

Machine	Libilité de l'OF	designation	Client	Quantité	Date fin au plus tard	Format	Matière	Matère	Ecran	Plaque	Forme	DAO	Envoyer
MACINTOSH I	OF2019050611-0106	FC-MAC-10942/C004 TB-PLV ALTAIA LILLE		223.00000	06/06/2019	122 x 175	PS DIFFUSANT 1mm 12.						
MACINTOSH I	OF2019050611-0206	FC-MAC-10942/C004 TB-PLV ALTAIA LILLE		5.00000	06/06/2019	122 x 175	PS DIFFUSANT 1mm 12.						
MACINTOSH I	OF2019050617	MDN981 - FC-MAC-10979/C ALTAIA LILLE		32.00000	06/06/2019	122 x 244	PVC EXPANSÉ 3 MM						
MACINTOSH I	OF2019050618	MDN981 - FC-MAC-10979/C ALTAIA LILLE		1478.00000	07/06/2019	70 x 180	STATTIQUE 115GCL 20X						
MACINTOSH I	OF2019050654	ARCHES DOUCHE SONS LAI - C MEDIA		68.00000	06/06/2019	140 x 301	R2 205/80 CLEAR ENL 1						
MACINTOSH I	OF2019050659-0101	ADHESIF VERTICAL MOT- C MEDIA		100.00000	06/06/2019	140 x 385	82.29 MACAL ENLEVABLE						
MACINTOSH I	OF2019050659-0201	ADHESIF VERTICAL MOT- C MEDIA		250.00000	06/06/2019	140 x 385	82.29 MACAL ENLEVABLE						
MACINTOSH I	OF2019050659-0301	ADHESIF VERTICAL MOT- C MEDIA		175.00000	06/06/2019	140 x 385	82.29 MACAL ENLEVABLE						
MACINTOSH I	OF2019050659-0401	ADHESIF VERTICAL MOT- C MEDIA		550.00000	06/06/2019	140 x 385	82.29 MACAL ENLEVABLE						
MACINTOSH I	OF2019050660	BANDES DE CONFIDENTIALITÉ - C MEDIA		67.00000	06/06/2019	137 x 190	R1212/200 CODE01210						
CONDITIONNEMENT	OF2019050661	PAS DE SOL OZY MOT-PA- C MEDIA		13.00000	06/06/2019	140 x 275	FLOOR ECO 140 X250M						
EMBALLAGE	OF2019050662	STICKERS DE COMPTOR C M MEDIA		18.00000	06/06/2019	140 x 360	R1 145 BLANC ENL L12						
MACINTOSH I	OF2019050663-0401	COVERING STANDARD MOT- C M MEDIA		200.00000	06/06/2019	140 x 375	MACAL 8129 ENLEVABLE						
MACINTOSH I	OF2019050663-0501	COVERING STANDARD MOT- C M MEDIA		300.00000	06/06/2019	140 x 375	MACAL 8129 ENLEVABLE						
MACINTOSH I	OF2019050684	BALISAGE RAYON-BALISAGE MÂTIÈRE PRUNILLE SAS		125.00000	05/06/2019	70 x 100	PVC 70/100 BLANC MAT						
MACINTOSH I	OF2019050695	BANDEAU AIR FRANCE-BAN- C M MEDIA		7.00000	06/06/2019	140 x 100	R1 145 BLANC ENL L120						
MACINTOSH I	OF2019060701	REASSORT MOULLES B+C-A FRANCIAISE DES JEUX B-2 100.00000		23/09/2019	122 x 180	SCPS100/F03 122 x 91							
FORME	OF2019060702	MARQUES OMBRELLES LOTC FRANCIAISE DES JEUX B-		300.00000	23/09/2019	80 x 137	CTY4C-10 LAZIE 137x9						
MACINTOSH I	OF2019060703	STICKERS DECLINAISON AM FRANCIAISE DES JEUX B-		351.00000	23/09/2019	137 x 95	CTY4C-10 - 1,37X9X9L						
MACINTOSH I	OF2019060704	AMALGAME MOT(PTD) + MO FRANCIAISE DES JEUX B-		350.00000	23/09/2019	137 x 90	CTY4C-10 - 1,37X9X9L						
MACINTOSH I	OF2019060706-0101	KITS HABILLAGES RACK FUZ ADARE		18.00000	26/06/2019	122 x 244	PVC EXPANSÉ BLANC 31						
MACINTOSH I	OF2019060706-0201	REGLLETTE BANIER-REGLETT GRAPHPROD		91.00000	25/06/2019	50 x 70	PVC 50/100 BLANC MAT						
MACINTOSH I	OF2019060706-0301	RENT A CAR TEL PERSONO RENT A CAR		2.00000	26/06/2019	137 x 180	V7000 FELLOUCLAGE L12						
MACINTOSH I	OF2019060706-0401	OMBRELLES LOTO SANS GAF FRANCIAISE DES JEUX B-		786.00000	19/08/2019	80 x 137	CTY4C-10 LAZIE 137x9						
MACINTOSH I	OF2019060709-0101	KITS VITRINE LUNETTES GRASSET ASS		500.00000	20/08/2019	140 x 200	R1 145 BLANC ENL L12						
MACINTOSH I	OF2019060709-0201	MARQUES OMBRELLES LOTC FRANCIAISE DES JEUX B-		300.00000	12/09/2019	140 x 195	R1 145 BLANC ENL L12						
MACINTOSH I	OF2019060709-0301	KITS VITRINE LUNETTES GRASSET ASS		300.00000	12/09/2019	140 x 195	R1 145 BLANC ENL L12						
MACINTOSH I	OF2019060709-0401	KITS VITRINE LUNETTES GRASSET ASS		300.00000	12/09/2019	140 x 195	R1 145 BLANC ENL L12						
NPRESS TIGER	OF2019060770-0201	DISPLAY 3 PIÈCES-FRONTOP DEMETZ		150.00000	13/09/2019	72 x 102	PRISMAPACK 300G						
FONDICA CO	OF2019060771	PLV DE PARKING LA HALLE K GUTENBERG LEVALLOIS		5.00000	28/06/2019	200 x 140	PP ALIEO 450 200x14						
MACINTOSH I	OF2019060772	KIT ASIE KITS STICKERS ASI MEGAMARK DALMEINSL		350.00000	04/07/2019	70 x 67	MACAL 9829 P 140x200						
MACINTOSH I	OF2019060773	FECHES COMPARATURS DE THE LINCS		9 683.00000	30/07/2019	70 x 100	PVC 30/100 CRYSTAL 7C						
MACINTOSH I	OF2019060774	STICKERS 3 MODÈLES STICK WEBER STEPHEN FRANC		209.00000	15/07/2019	72 x 102	YUPO TAHO 320 102C						
EMBALLAGE	OF2019060775-0101	EMBOITE RET 2016 RETRO BOT RET BY MITCHON I C		1.000.00000	25/07/2019	75 x 105	NAUT 988 988 664						

FIGURE 1.6 – Aperçu de la fenêtre de tri du planning de fabrication avec l'avancement des phases de préparation

Le développement du planning de fabrication de SIGMA s'est déroulé en parallèle de celui de la SFAO et il n'y a eu aucun problème majeur.

1.4.3 La SFAO de SIGMA

Avant le développement de SIGMA, il existait une SFAO sur DISANET qui permettait de suivre le processus de fabrication de la compagnie. Au sein de SIGMA, nous avons décidé de développer un outil similaire, permettant de remplir le même rôle, mais en y apportant des améliorations.

1.4.3.1 Le fonctionnement de l'ancienne SFAO

Pour savoir le travail qu'ils avaient à effectuer, les ouvriers devaient se rendre au tableau de planification. C'est ici que leur travail leur était attribué, ils devaient récupérer un dossier papier contenant l'OF, ainsi que toutes les informations nécessaires à l'impression.

Ils allaient ensuite sur un poste informatique afin d'enregistrer le début de leur phase de préparation. Pour s'identifier, ils devaient scanner un code-barres présent sur leur badge personnel. Ensuite, ils devaient scanner le code-barre propre à la phase de fabrication, imprimé sur le dossier papier. Une fois la phase terminée, il fallait refaire ce même processus pour clôturer la phase.

Un système de synchronisation des données entre la SFAO et ASAP avait été mis en place, ce qui permettait de renseigner l'avancement des phases de fabrication automatiquement au sein d'ASAP.

L'objectif avec le développement de la SFAO au sein de SIGMA était de profiter de la base de données unique pour proposer des fonctionnalités jusqu'alors impossibles et également pour diminuer la consommation de papier.

1.4.3.2 Le développement de la SFAO sous SIGMA

J'ai effectué la quasi-totalité du développement de la SFAO. Afin de contrôler mon travail je travaillais principalement avec M. MERY, responsable de la production.

La SFAO doit permettre aux ouvriers de toujours savoir le travail qu'ils ont à effectuer, en fonction de la machine sur laquelle ils travaillent. Les ouvriers doivent pouvoir s'identifier et sélectionner leur machine facilement afin de voir les phases à faire. La sélection de l'ouvrier et de la machine se fait simplement via une liste déroulante et les informations de phases s'affichent juste à côté de la machine sélectionnée.

Les phases s'affichent dans l'ordre de priorité défini par le planning de fabrication. L'opérateur peut cliquer sur les phases afin d'afficher des informations complémentaires et lorsque tout est prêt il suffit de cliquer sur « Démarrer » et la phase se lance. D'un point de vue technique lorsqu'un employé démarre une phase, un triptyque se crée en base de données, reliant la personne à la machine ainsi qu'à la phase de fabrication. Ce triptyque permet de garder un historique complet de tout le processus de fabrication. Il contient des informations complémentaires, telles que la date de début et de fin de la phase, les quantités préparées, des commentaires, etc.

La SFAO permet aussi de signaler les aléas, et d'en indiquer la raison. Cela à deux intérêts, le premier est de pouvoir informer directement les responsables du souci. Ces derniers sont informés de par l'affichage de la machine en mode erreur dans le planning et également dans le plan d'usine, un outil développé par Alexandre MEUNIER qui permet d'exploiter les données de la SFAO pour afficher un statut de toutes les machines de l'usine en direct. Le second intérêt de signaler les aléas sur SIGMA est l'enregistrement d'un historique en base de données. Ce sont des données importantes qu'il faut conserver pour l'élaboration de statistiques. Des statistiques sur les aléas des machines peuvent, par exemple, mettre en évidence une machine présentant des problèmes plus fréquemment que les autres, etc.

Enfin, un ouvrier peut suspendre une phase, pour faire une pause ou en commencer une autre pour quelque raison que ce soit. Il peut également terminer une phase en cours et saisir les quantités préparées lors de la réalisation de celle-ci.

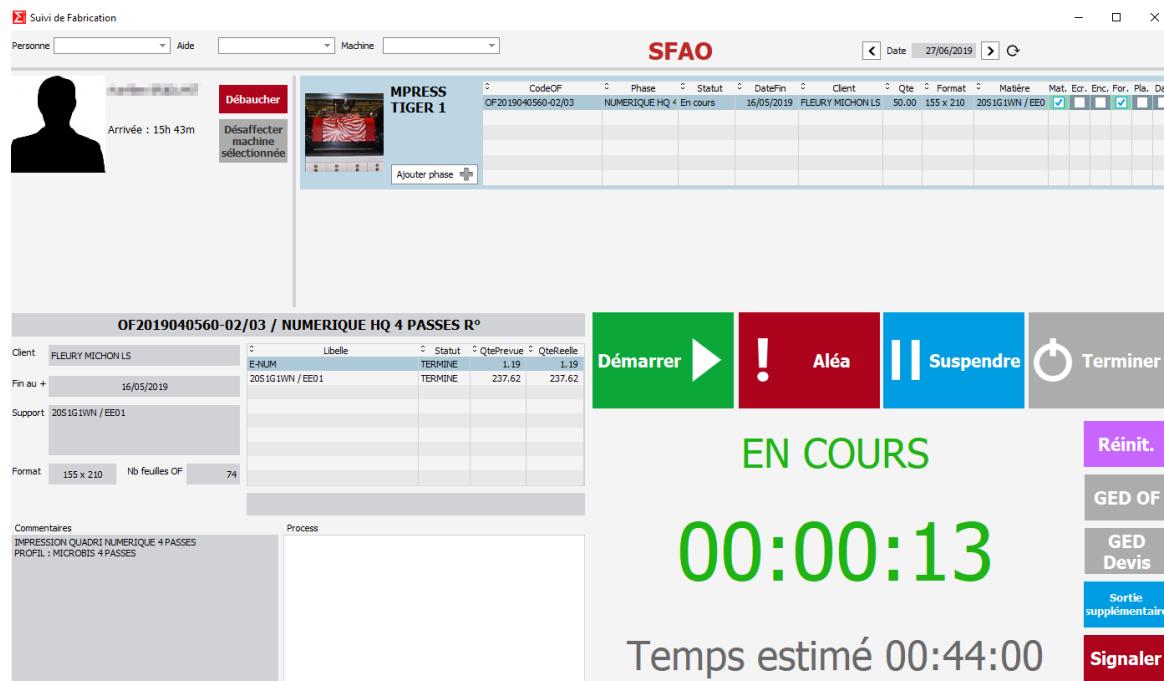


FIGURE 1.7 – Aperçu de la fenêtre de la SFAO, avec une phase d'impression numérique en cours

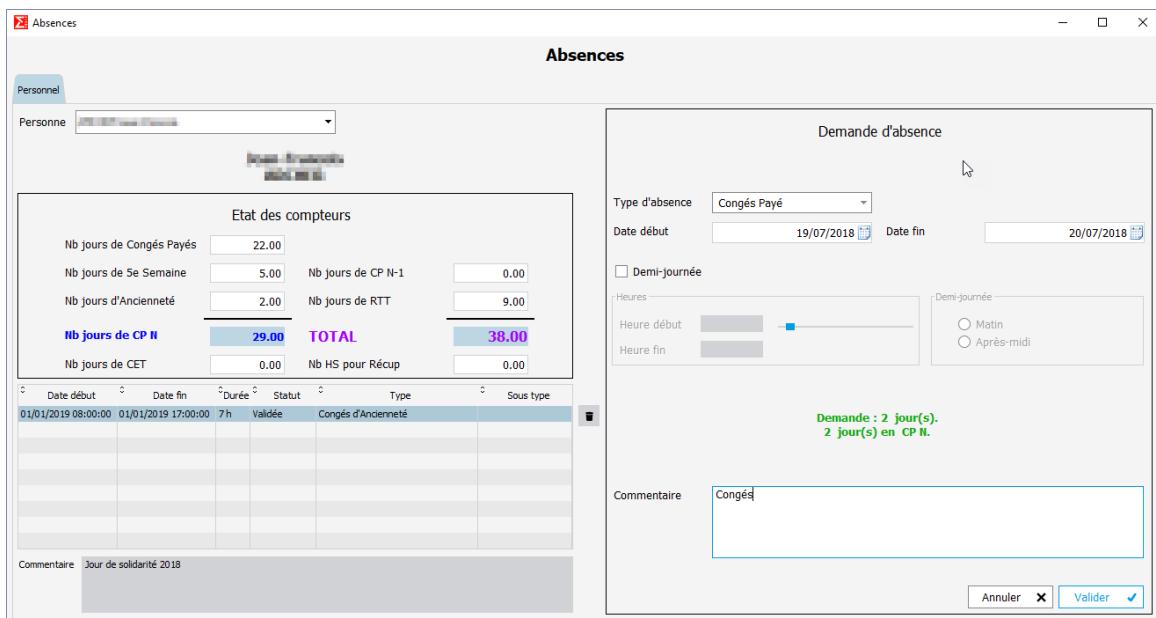
La SFAO est un outil qui fut intéressant à développer, notamment dû au fait que nous avons réalisé son développement en parallèle avec le développement du planning de fabrication, ainsi que le plan d'usine. Cela nous a permis à moi et Alexandre MEUNIER de développer notre capacité à travailler ensemble.

1.4.4 Module des ressources humaines

1.4.4.1 La conception du module

Lors de ma septième mission en entreprise, il m'a été demandé de développer l'ensemble des applications permettant de gérer les absences, primes, frais professionnels, permanences, heures supplémentaires, etc. C'est donc un module qui doit permettre des fonctionnalités variées. Il est, dans ce cas, particulièrement important de prévoir à l'avance les différents liens entre toutes les applications, afin d'éviter des problèmes d'incohérences.

La première étape fut la préparation de la base de données. Chaque employé possède des compteurs pour leurs congés payés de l'année en cours, ceux de l'année précédente, leurs jours d'ancienneté, etc. Pour le développement des outils, il est nécessaire de prendre en compte la manière dont tous ces compteurs sont mis à jour. L'ordre de développement déterminé fut tout d'abord l'outil permettant de gérer les absences, puis celui permettant de gérer les heures supplémentaires. Une fois ces outils développés, la totalité des compteurs en base sont gérés, on peut donc développer la suite des outils sans suivre un ordre en particulier.



The screenshot displays two windows side-by-side. The left window, titled 'Absences', shows the 'Etat des compteurs' (Account Status) for an employee named 'Personne'. It lists various leave types with their current balances: Congés Payés (22.00), 5e Semaine (5.00), Ancienneté (2.00), CP N-1 (0.00), RTT (9.00), CP H (29.00), TOTAL (38.00), and CET (0.00). The right window, titled 'Demande d'absence', is a form for creating a new absence record. It includes fields for 'Type d'absence' (Congés Payé), 'Date début' (19/07/2018), 'Date fin' (20/07/2018), and 'Heures' (2 hours). It also includes a 'Commentaire' field containing 'Jour de solidarité 2018'. A message at the bottom states 'Demande : 2 jour(s). 2 jour(s) en CP H.' (Request: 2 days. 2 days in CP H.).

FIGURE 1.8 – Aperçu de la fenêtre permettant de gérer ses absences

La gestion des fiches de payes est externalisée. Chaque mois, pour tous les employés de DISA, un fichier EXCEL contenant toutes les informations nécessaires à la création des fiches de paie est transmis au sous-traitant. Grâce aux nouveaux outils développés, nous avons pu automatiser la création de l'ensemble de ces fichiers en un clic.

NOM SALARIE : RANVIER Thomas
Compteur récup. au 20/06 18.5

MOIS DE : Juillet 2018

Equipe

Date	Nombre heures absence	Absences excep.	Heures absence prises sur récup	Maladie	AT	CP	RTT	Heures supplém.	Heures sup en récup	Heures nuit	Equipe
Jeudi, 21 Juin											
Vendredi, 22 Juin		7									
Samedi, 23 Juin											
Dimanche, 24 Juin											
Lundi, 25 Juin		7				7					
Mardi, 26 Juin		7				7					
Mercredi, 27 Juin		7		3.5				1			
Jeudi, 28 Juin								0.5			
Vendredi, 29 Juin										X	
Samedi, 30 Juin								6.5			
Dimanche, 1 Juil											
Lundi, 2 Juil											
Mardi, 3 Juil											
Mercredi, 4 Juil											
Jeudi, 5 Juil											
Vendredi, 6 Juil											
Samedi, 7 Juil											
Dimanche, 8 Juil											
Lundi, 9 Juil											
Mardi, 10 Juil											
Mercredi, 11 Juil											
Jeudi, 12 Juil											
Vendredi, 13 Juil											
Samedi, 14 Juil											
Dimanche, 15 Juil											
Lundi, 16 Juil	Formation 7										
Mardi, 17 Juil	Formation 7		5								
Mercredi, 18 Juil	Formation 7										
Jeudi, 19 Juil	Formation 7										
Vendredi, 20 Juil	Formation 7										
			5						6.5		
REGUL	0	0	0	0	0	28	0	0	2	0	0

Variable	Montant / nombre
Prime polyvalence	
Prime approvisionnement	
Prime programmation table	
Prime resp sécurité	
Prime prod logistique	
Prime déchets	
Prime véhicule	
Prime ponctuelle	
Permanences maintenance (en nombre)	
Rachat CET (en h)	2
Acompte	
Tickets resto	17
Frais Prof	
Compteur récup à reporter	20

FIGURE 1.9 – Exemple de fichier généré à partir de données fictives

1.4.4.2 Les développements futurs

Le module des ressources humaines tel qu'il est développé actuellement ne permet pas de gérer les congés dans le respect complet des lois. Le principal problème est que les droits à des jours de congés ne sont actualisés qu'une fois par an alors que les congés acquis chaque mois devraient être automatiquement ajoutés aux compteurs des employés. Cela a pour effet d'obliger les employés à poser des congés par anticipation au sein de SIGMA alors que ce sont en fait des congés qu'ils ont acquis au cours de l'année.

La correction de ce souci peut sembler plutôt simple d'un œil extérieur, mais en fait l'adaptation du module des ressources humaines à ce fonctionnement entraînerait beaucoup de modifications nécessaires pour retrouver un fonctionnement stable. Nous sommes donc actuellement en réflexion avec M. STECLEBOUTE, le nouveau responsable informatique, pour trouver la meilleure manière de s'y prendre pour mettre à jour ce système.

1.4.5 Mon travail lors de cette dernière mission

La fin de la mise en production de SIGMA s'est effectuée en grande partie lors de mon séjour à l'étranger, je n'étais donc pas présent. Beaucoup d'employés utilisaient déjà SIGMA au quotidien avant cette fin de mise en production, puisque celle-ci s'est effectuée graduellement dans le temps. Durant la période de fin d'année 2018, M. PALIER et M. STECLEBOUTE ont formés les employés qui n'utilisaient pas encore SIGMA.

Lors de cette dernière mission, le travail que j'ai eu à réaliser était donc principalement de la correction de bug, la modification et l'amélioration de certaines fonctionnalités afin de mieux les adapter aux besoins des utilisateurs et aussi le développement de quelques nouvelles fonctionnalités.

1.4.5.1 Amélioration de la gestion du travail au sein du service informatique

Afin de gérer les corrections de bugs nous avons mis en place une adresse mail sur laquelle les utilisateurs peuvent nous faire parvenir tous les soucis qu'ils rencontrent. Afin de simplifier la gestion de ces corrections en tant qu'équipe, j'ai proposé d'utiliser l'outil de gestion de projet TRELLO. Nous avons synchronisé la boîte mail des bugs avec cet outil, ainsi dès qu'un mail est reçu, une tâche est créée au sein de TRELLO. Les tâches créées au sein de TRELLO peuvent être attribuées à une personne, il est possible de les commenter, les déplacer dans différentes catégories, etc. L'utilisation de cet outil a permis de simplifier notre gestion du travail au sein du service informatique. Désormais lorsque nous voulons développer une nouvelle fonctionnalité ou en améliorer une ancienne, on y crée une tâche qui permet de suivre l'avancement du travail. TRELLO nous permet aussi de garder un historique simple à parcourir dans lequel on peut retrouver des informations concernant différentes tâches réalisées plus tôt.

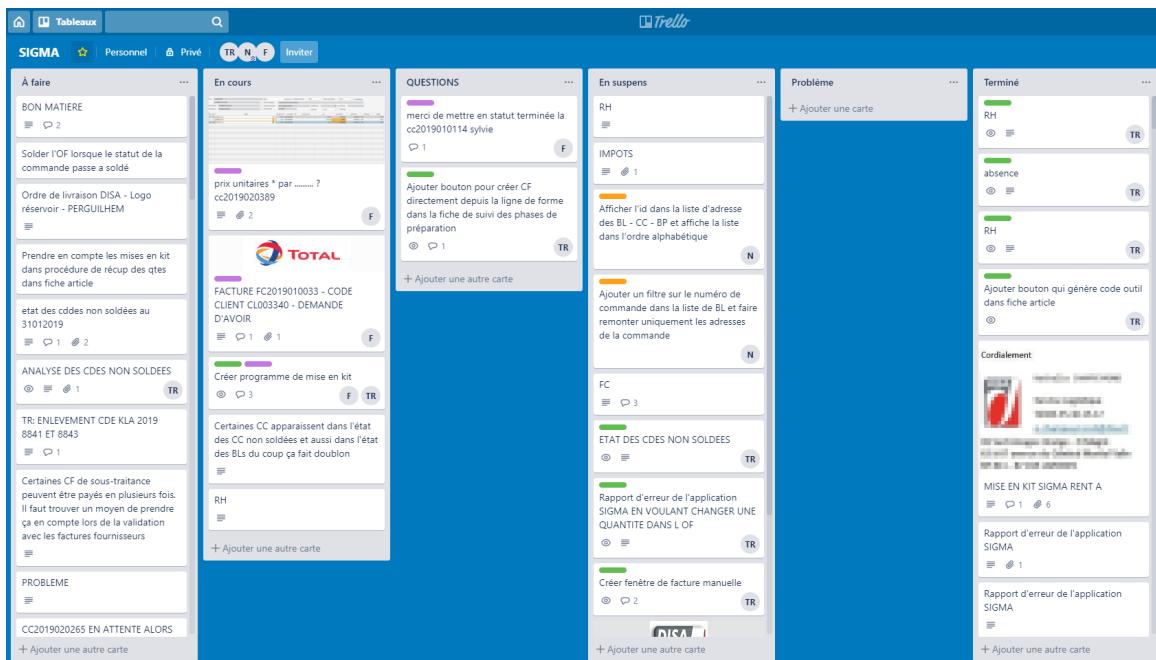


FIGURE 1.10 – Aperçu du tableau du projet SIGMA sous TRELLO

1.4.5.2 Amélioration des performances des requêtes

Lors du développement de certaines fenêtres permettant l'affichage de données pour des statistiques nous nous sommes aperçus que certaines requêtes en base de données étaient particulièrement longues à exécuter.

Après quelques recherches, nous avons découvert avec M. STECLEBOUTE que les requêtes utilisant le Wlangage étaient plus longues à exécuter que les requêtes SQL. En remplaçant certaines de ces anciennes requêtes par des requêtes SQL nous avons noté une amélioration de performances par un facteur de 2, voire 3.

Il s'avère que nous n'avions jamais remarqué cette différence notable de performance plus tôt puisque les requêtes habituelles étaient bien moins conséquentes et avaient donc un temps d'exécution négligeable par rapport aux requêtes nécessaires pour l'affichage de ces statistiques. Nous avons donc remplacé beaucoup de requêtes écrites avec le Wlangage par leur équivalent en SQL afin d'améliorer les performances de celles-ci. Désormais lorsque nous faisons des requêtes en base de données nous nous efforçons de toujours les écrire en SQL.

1.4.5.3 Nouvelles fonctionnalités

Au cours de cette dernière mission, la très grande majorité de mon travail était concentrée sur la résolution de bugs rencontrés par les employés et l'amélioration de fonctionnalités permettant de faciliter leur travail quotidien.

Une autre partie de mon travail fut aussi de développer des fonctionnalités jusqu'alors inexistantes. J'ai notamment développé une application permettant de gérer les mises en kits⁷. Il fallait donc développer une application permettant de facilement connaître l'état des stocks de tous les composants du kit à envoyer.

J'ai aussi développé, avec M. STECLEBOUTE, des fonctionnalités permettant la gestion des formes de découpes. Jusqu'à présent, la gestion des formes de découpes se faisait encore à l'aide de DISANET. Les formes de découpes possèdent un statut qui leur est propre, il est aussi important de sauvegarder le devis de fabrication de chaque forme, etc. Les formes de découpes sont un type particulier d'article, or tous les types d'articles sont gérés au sein de SIGMA. Nous avons donc adapté le module de gestion des articles afin de prendre en compte les particularités des formes de découpes. Désormais, elles peuvent être entièrement gérées avec SIGMA.

7. Les mises en kits consistent en le conditionnement et l'expédition de produits déjà fabriqués.

1.5 Retour d'expérience sur mon apprentissage

Mon apprentissage au sein de la compagnie DISA fut enrichissant de par la diversité des tâches que j'ai eu à effectuer.

J'ai principalement travaillé sur le projet SIGMA, qui est un projet de grande ampleur qui s'est déroulé sur la totalité de mon apprentissage. J'ai effectué des tâches de récolte du besoin, analyse du besoin, conception, développement, puis déploiement de l'ERP. C'est donc un travail très diversifié, demandant de la réflexion et qui m'a permis d'acquérir des connaissances et une expérience qui me sera précieuse.

Le projet SIGMA peut être considéré comme un succès. La totalité des employés de la compagnie l'utilisent au quotidien et il leur permet d'effectuer leur travail plus facilement et dans de meilleures conditions qu'auparavant. Le travail de correction des bugs a permis de se débarrasser de tous les problèmes empêchant les employés de travailler correctement, cependant, il reste de nombreux points à améliorer. Ce travail devra donc se poursuivre jusqu'à l'obtention d'un ERP stable ne nécessitant que des modifications exceptionnelles.

Bien sûr, il y aura toujours de nouvelles applications à développer ou à modifier sur SIGMA afin de l'adapter aux besoins changeants des employés.

Je suis, dans l'ensemble, satisfait par mon travail sur ce projet.

Chapitre 2

Sujet de recherche

QU'EST-CE QUE L'INTELLIGENCE ARTIFICIELLE PEUT APPORTER À UN ERP ?

2.1 Qu'est-ce que l'intelligence artificielle ?

Cette première partie a pour objectif de définir ce qu'est l'intelligence artificielle et de montrer les domaines d'application dans lesquels celle-ci peut être utilisée.

2.1.1 Définition de l'intelligence

L'intelligence est souvent vue comme étant propre à l'être humain, on emploie d'ailleurs le terme « bête » pour désigner à la fois les animaux et les personnes que l'on considère comme ayant peu d'intelligence.

Pourtant, on pourrait la définir comme étant la capacité à s'adapter à des situations nouvelles, ce qui permet de prendre en compte de nombreux comportements que l'on trouve chez les animaux, et même plus globalement chez les êtres vivants [1].

En exemple de cette capacité d'adaptation, on peut citer le cas de Hans le malin, un cheval qui « savait » compter et donnait le résultat d'un calcul par des coups de sabot sur le sol. En réalité, le cheval n'était en capacité ni de comprendre la question ni de calculer la réponse, mais il était capable de détecter les micro-expressions du public ou de son maître pour savoir quand s'arrêter de frapper.

Une autre forme d'intelligence intéressante est l'intelligence collective, on la retrouve chez les humains, dans des mouvements de foule notamment, mais aussi chez beaucoup d'animaux et insectes. Les fourmis sont un autre très bon exemple avec la présence de rôles : reine, ouvrières, nourrices, gardiennes, combattantes, etc.

Il existe encore beaucoup d'autres formes d'intelligences et toutes celles-ci sont très intéressantes à analyser et à imiter à l'aide de l'intelligence artificielle. Nous verrons plus en détail par la suite les mises en application possibles.

2.1.2 Définition de l'intelligence artificielle

Dans leur livre « Artificial Intelligence : A Modern Approach » [2], Stuart J. RUSSELL et Peter NORVIG, deux acteurs importants de la recherche en intelligence artificielle, la définissent sous quatre aspects différents.

Ces aspects opposent le fait de penser au fait d'agir et le fait de le faire humainement ou rationnellement. Ces quatre approches ont toutes été suivies par différentes personnes en employant des méthodes différentes dans leurs recherches.

Penser humainement	Penser rationnellement
Système développé dans l'idée d'imiter la manière de penser humaine	Système développé dans l'idée de penser de manière rationnelle
Agir humainement	Agir rationnellement
Système développé dans l'idée d'agir comme un humain le ferait face à une même situation	Système développé dans l'idée d'agir de la manière la plus rationnelle possible

FIGURE 2.1 – Quatre approches de l'intelligence artificielle

2.1.2.1 Penser humainement, la science cognitive

Avant de pouvoir faire en sorte qu'un programme pense comme un être humain, il faut déjà être en mesure de déterminer comment un humain pense.

Il y a trois manières pour cela :

1. L'introspection : essayer d'analyser nos pensées et notre manière de penser.

-
2. L'expérimentation psychologique : observer une personne en action.
 3. L'imagerie cérébrale : observer les stimuli d'un cerveau en action.

2.1.2.2 Penser rationnellement, les lois de la pensée

Le philosophe ARISTOTE fut le premier à réfléchir à ce qu'est « penser rationnellement ». Il est à l'origine du syllogisme qu'il définit comme « un discours dans lequel, certaines choses étant posées, quelque chose d'autre que ces données en résulte nécessairement par le seul fait de ces données » [3].

Le syllogisme repose sur deux prémisses, une majeure et une mineure, desquelles on peut tirer une conclusion, par exemple :

Majeure : SOCRATE est un homme.
Mineure : Les hommes sont mortels.
Conclusion : SOCRATE est mortel.

2.1.2.3 Agir humainement, le test de TURING

En 1950, dans son livre « Computer and thought » [4], Alan TURING décrit un test d'intelligence artificielle consistant à mettre un humain en confrontation, à l'aveugle, avec un ordinateur et un autre humain. Avec cet ouvrage il souhaitait remplacer la question « les machines peuvent-elles penser ? » par « un ordinateur digital peut-il tenir la place d'un être humain dans le jeu de l'imitation ? », la première n'ayant que trop peu de sens pour mériter une discussion [5].

Le test se présente comme tel :

Un être humain noté C est confronté à l'aveugle à un ordinateur et un autre être humain, ces derniers sont notés A et B sans que C ne sache ce qu'ils sont. C peut poser des questions à A et B , ceux-ci répondent via un message écrit à l'ordinateur afin que le ton de la voix n'aide pas C . Si C n'est pas en mesure de déterminer lequel est humain ou non, l'ordinateur passe le test. Les résultats de ce test ne dépendent pas de la capacité de l'ordinateur à donner de bonnes réponses, mais plutôt de la capacité à imiter les réponses d'un humain.

Pour le moment, aucun programme n'est parvenu à passer ce test, il reste donc encore pertinent. Cependant, les chercheurs du domaine n'ont pas beaucoup cherché à passer ce test, préférant se concentrer sur l'étude des principes régissant l'intelligence plutôt que d'essayer d'imiter cette dernière. Effectivement, d'autres domaines scientifiques ont montré de gros progrès en appliquant cette méthode. Par exemple, c'est lorsque les frères WRIGHT ont cessé d'essayer d'imiter les oiseaux et se sont concentrés sur le principe sous-jacent d'aérodynamique qu'ils sont parvenus à créer une machine volante.

2.1.2.4 Agir rationnellement, les agents rationnels

Un agent est, par définition, simplement quelque chose qui agit. Un agent rationnel se doit cependant de prendre les décisions qui sont supposées lui permettre de maximiser sa mesure de performance. Il doit également être en mesure d'agir de manière autonome, de percevoir son environnement et de persister sur une durée de temps prolongée sans nécessiter aucune aide extérieure. Contrairement à la catégorie « penser rationnellement », ici l'agent n'est pas obligé de penser avant d'agir, il peut faire preuve de « réflexes » si cette action est la plus rationnelle.

2.1.2.5 Conclusion

Il existe donc des chercheurs qui pensent qu'une intelligence artificielle se doit de penser ou d'agir comme un être humain et d'autres que celle-ci doit penser ou agir rationnellement. Il n'est pas possible de dire lesquels ont raison, cela relève du débat philosophique et éthique.

L'encyclopédie Larousse la définit comme « l'ensemble des théories et des techniques mises en œuvre en vue de réaliser des machines capables de simuler l'intelligence » [6]. Cette définition est clairement basée sur le concept d'agir de manière humaine comme Alan TURING la définissait.

L'intelligence en elle-même est un concept difficile à définir précisément, car celle-ci peut prendre de nombreuses formes. Il existe des tests qui tentent de mesurer l'intelligence, ce sont les tests de Q.I. Mais ceux-ci sont sujets à de fortes polémiques, car avec un entraînement spécifique une personne peut fortement améliorer ses scores à ce genre de tests. Il en va de même avec le test de TURING pour l'intelligence artificielle.

2.1.3 L'histoire de l'intelligence artificielle

Maintenant que nous avons défini ce que l'on entend par le terme « Intelligence Artificielle », nous allons voir l'historique des recherches et applications ayant pris place dans ce domaine et l'ayant fait progresser.

2.1.3.1 Avant l'intelligence artificielle (1943–1955)

Les premiers travaux qui sont aujourd'hui généralement reconnus comme portant sur l'intelligence artificielle sont ceux de Warren MCCULLOCH et Walter PITTSANDREAS, en 1943 [7]. Ils proposèrent un modèle de neurones artificiels, dans lequel chaque neurone peut être soit « on », soit « off », avec un passage au statut « on » s'effectuant en réponse à une stimulation de suffisamment de neurones voisins. Ils montrèrent également que toute fonction calculable¹ peut être calculée en utilisant un réseau de neurones connectés et que toutes les portes logiques² peuvent être implémentées par des réseaux simples. Ils suggérèrent aussi dans le même article qu'un réseau structuré convenablement serait capable d'apprendre.

En 1949, dans son livre « The Organization of Behavior » [8], Donald HEBB introduit la règle de HEBB, celle-ci est souvent résumée comme suit : « cells that fire together, wire together », autrement dit « des neurones qui s'excitent ensemble se lient entre eux. ». Elle est encore utilisée à ce jour, à la fois comme une hypothèse dans le domaine des neurosciences et comme un concept dans les réseaux neuronaux.

En 1950, deux étudiants de l'université de HARVARD, Marvin MINSKY et Dean EDMONDS, ont créé le premier ordinateur à réseau de neurones. La machine se nommait SNARC et permettait de simuler un réseau d'une quarantaine de neurones. L'objectif était que la machine apprenne le trajet d'un labyrinthe afin de le résoudre.

Dès 1947, Alan TURING présentait déjà des séminaires sur le sujet de l'intelligence artificielle à la « London Mathematical Society ». En 1950 dans son livre « Computers and Thought » [4], Alan TURING introduit le test de TURING, le concept de « machine learning », littéralement « l'apprentissage machine », les algorithmes génétiques, ainsi que l'apprentissage par renforcement. Il proposa également l'idée de « child program », visant à se focaliser sur la simulation d'un esprit enfant avant de vouloir simuler l'intelligence d'un adulte. Ces différents concepts sont les bases de beaucoup de recherches encore actuelles.

2.1.3.2 La naissance de l'intelligence artificielle (1956)

En 1956, le professeur John MCCARTHY convint Marvin MINSKY, Claude SHANNON et Nathaniel ROCHESTER de l'aider à réunir des chercheurs intéressés par l'automatisation, les réseaux de neurones et l'étude de l'intelligence. Ils organisèrent alors une étude regroupant dix personnes sur deux mois, sur le sujet de l'intelligence artificielle au DARTMOUTH College, une université américaine [2]. Le but était d'essayer de trouver comment permettre à une machine d'utiliser un langage, de faire preuve d'abstraction et de résoudre des problèmes jusqu'alors réservés aux humains. Cette étude n'a pas mené à de nouvelles avancées, en revanche elle a permis de créer le terme « Artificial Intelligence » pour définir ce domaine. Elle permit également à toutes les grandes figures de l'intelligence artificielle de cette époque de se rencontrer et pour les vingt années à venir, ce domaine fut largement dominé par ceux-ci, ainsi que par leurs étudiants.

2.1.3.3 Les premières grandes avancées (1952–1969)

Les avancées dans le domaine au cours des premières années de recherches furent nombreuses et très rapides, le nombre de problèmes qu'une machine était désormais capable de résoudre ne faisait qu'augmenter.

Arthur SAMUEL débute en 1952 le développement d'une série de programmes jouant aux échecs. Ceux-ci parvinrent à atteindre un niveau supérieur à beaucoup d'amateurs. Son programme parvint rapidement à apprendre à mieux jouer aux échecs que son propre créateur. Il infirma par cette occasion l'idée qu'un ordinateur ne peut faire que ce qu'on lui demande. Dans son livre, écrit en 1959 [9], il crée le terme « machine learning ».

1. Une fonction calculable est une fonction semi-calculable qui est aussi totale, c'est-à-dire définie sur tout son domaine.

2. En électronique, les portes logiques s'appuient sur les principes de la logique binaire (0, 1), ce sont des circuits de base capables de représenter des fonctions logiques : et, ou, non, non-et, etc.

Il est aujourd’hui considéré comme l’un des principaux pionniers dans le domaine de la théorie des jeux et de l’intelligence artificielle.

En 1959, Allen NEWELL, John C. SHAW et Herbert A. SIMON créent le « General Problem Solver » [10]. En principe, n’importe quel problème formalisé peut être résolu par le GPS, par exemple des preuves de théorèmes ou des parties d’échecs. Cet algorithme fut le premier à être conçu dans l’idée de penser comme un humain.

En 1958, John MCCARTHY créa le langage de programmation de haut niveau, LISP. Il fut le langage dominant dans le domaine pour les trente années suivantes.

2.1.3.4 Les premières difficultés (1966–1973)

Les avancées dans le domaine étaient si rapides que Herbert A. SIMON fit la prédiction en 1957 que dans moins de dix ans un programme serait capable de battre le champion du monde d’échecs. Cette prédiction s’est révélée vraie, bien que trop ambitieuse : il fallut attendre environ quarante ans pour cela et non dix. Cet excès de confiance était dû au fait qu’à ce moment les systèmes développés montraient des performances très prometteuses sur des exemples simples. Cependant, il s’avère que ces systèmes donnèrent de très mauvais résultats lorsqu’ils furent testés sur des problèmes plus complexes.

Un exemple célèbre de cette observation concerne une recherche prenant lieu aux États-Unis en 1957 : le but était d’accélérer la traduction de documents russes en utilisant un algorithme de traduction. Le système développé était basé sur une présomption trop simpliste, alors qu’une traduction précise nécessite des connaissances permettant de résoudre l’ambiguïté et d’établir le sens d’une phrase. Cela donna une traduction célèbre dans le domaine de la traduction automatique. La phrase « the spirit is willing but the flesh is weak » (« l’esprit est fort mais le corps est faible ») fut traduite comme « the vodka is good but the meat is rotten » (« la vodka est bonne mais la charcuterie est pourrie »). En 1966, tous les financements des États-Unis pour la recherche en traduction furent stoppés, car aucune avancée importante n’avait eu lieu au cours des dernières années.

En 1969, Arthur BRYSON et Yu-Chi HO [11] découvrirent plusieurs algorithmes de backpropagation³, énormément utilisés de nos jours, pour entraîner les réseaux de neurones. Cependant, en 1969 il était impossible de mettre en pratique ces algorithmes par manque de ressources de calculs, il fallut attendre des recherches dans la fin des années 1980 pour redécouvrir ces algorithmes et les mettre en pratique.

2.1.3.5 L’hiver de l’intelligence artificielle (1974–1980)

Le domaine connut une période durant laquelle le nombre de recherches diminua radicalement, principalement à cause du manque de financement.

À ce moment le domaine faisait face à des critiques, du fait que les estimations des scientifiques travaillant sur le sujet s’étaient révélées trop ambitieuses.

En 1969, Marvin MINSKY publia le livre « Perceptrons » [12], dans lequel il expose ses doutes quant à l’avenir des systèmes de réseaux de neurones. Ce livre eut un tel effet dans le monde de la recherche que pour les dix années à venir il n’y eut plus de recherches effectuées sur le sujet.

2.1.3.6 Le retour de l’intelligence artificielle (1980-aujourd’hui)

Dans le milieu des années 1980, au moins quatre groupes de recherches indépendants réinventèrent la backpropagation, qui avait déjà été découverte en 1969 par Arthur BRYSON et Yu-Chi HO [11]. La technologie ayant beaucoup évolué, il était désormais possible de mettre en pratique ces algorithmes. Depuis ce moment les scientifiques continuent à faire de fortes avancées sur ce sujet.

Mais il était toujours compliqué de trouver des financements et c’est à partir de l’année 1993 que des avancées sur divers sujets de l’intelligence artificielle permirent au monde de l’industrie de s’y intéresser à nouveau.

Dans les années 1990, le paradigme d’agents intelligents fait son apparition. Un agent intelligent est un système capable de percevoir son environnement et de choisir une action permettant de maximiser ses chances de succès.

3. Rétropropagation du gradient, c'est une méthode de statistiques utilisée pour entraîner des réseaux de neurones

Le 11 mai 1997, Deep Blue devient le premier ordinateur à battre un champion du monde des échecs, Garry KASPAROV.

À cette époque beaucoup d'avancées en intelligence artificielle sont faites dans le monde de l'industrie, mais les chercheurs préfèrent employer d'autres termes pour qualifier leur travail, tels que « informatique », « systèmes cognitifs » ou encore « Intelligence computationnelle ». Cela s'explique par le fait que ces nouveaux noms rendaient les financements plus simples. Parmi ces grandes avancées, on peut compter l'engin de recherche de Google, l'apparition du data mining⁴, les robots industriels, les systèmes bancaires, etc.

2.1.3.7 Le deep learning et le big data (2011-aujourd'hui)

Les capacités des ordinateurs continuant à fortement augmenter et l'accès à de plus en plus de données firent apparaître le deep learning (apprentissage profond) et le big data.

L'apprentissage profond permet un niveau d'abstraction élevé sur des données, grâce à l'utilisation de réseaux neuronaux profonds. Le principe est qu'en ajoutant des niveaux de neurones supplémentaires dans un réseau, celui-ci va permettre d'atteindre un niveau d'abstraction bien plus élevé et ainsi résoudre des problèmes beaucoup plus complexes.

Le big data est un terme utilisé pour désigner un ensemble de données si imposant qu'il n'est plus possible de les analyser en tant qu'être humain. Il est alors nécessaire d'utiliser des méthodes de data mining afin de pouvoir visualiser et exploiter ces données.

4. La fouille de données, c'est l'extraction d'informations depuis une grande quantité de données brutes

2.2 Que permet de faire l'intelligence artificielle ?

Afin de pouvoir répondre à la problématique « Qu'est-ce que l'intelligence artificielle peut apporter à un ERP ? », il faut prendre connaissance d'où en est l'état de l'art de l'intelligence artificielle. Nous allons voir par la diversité des concepts avancés dans cette partie que l'intelligence artificielle est un domaine extrêmement vaste et qui permet des applications dans tous les domaines de l'industrie et de la recherche.

2.2.1 Systèmes multi-agents

Dans les systèmes multi-agents, des agents aux comportements individuels simples interagissent entre eux pour résoudre des problèmes complexes, par l'apparition d'une forme d'intelligence collective.

2.2.1.1 Automates cellulaires

Les automates cellulaires sont des outils permettant de mimer des processus auto-reproductifs (c'est un cas particulier de système multi-agents). Ils sont basés sur des règles simplistes et permettent cependant de faire émerger des comportements extrêmement complexes. Un automate cellulaire consiste en une grille de « cellules », chaque cellule possède un état au temps t et l'état de cette cellule au temps $t+1$ est défini par l'état des cellules avoisinantes au temps t en fonction des règles définies.

L'exemple d'automate cellulaire le plus célèbre est le « jeu de la vie » imaginé par John Horton CONWAY en 1970 [13]. Ce « jeu » se déroule sur une grille bi-dimensionnelle, les cases de celles-ci sont nommées des cellules et peuvent prendre deux états : « vivante (1) » ou « morte (0) ». Les règles sont définies telles que :

- Une cellule morte possédant exactement trois voisines vivantes devient vivante, sa valeur passe à 1.
- Une cellule vivante possédant deux ou trois voisines vivantes le reste, sinon elle meurt.

La grille est ensuite initialisée, soit avec des valeurs pré-définies, soit aléatoirement. L'univers généré paraît chaotique pendant quelques étapes, puis rapidement des structures apparaissent. Certaines sont stables, l'état des cellules les constituant ne change pas tant qu'aucun événement extérieur ne vient les perturber. D'autres sont périodiques, l'état des cellules les constituants évolue au cours du temps et finit toujours par revenir au même point. D'autres prennent la forme de vaisseaux, celles-ci sont capables de reproduire une copie d'elle-même après un certain nombre d'étapes, mais décalée dans l'univers du jeu.

La figure ci-dessous représente un GOSPER glider gun, sa partie principale se répète périodiquement et émet des vaisseaux à intervalles réguliers.

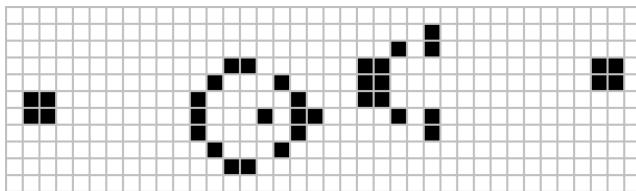


FIGURE 2.2 – GOSPER glider gun

Les automates cellulaires sont un exemple de systèmes constitués de composantes connues, mais faisant émerger des comportements bien plus complexes à appréhender. Ils sont donc particulièrement utiles pour mener des recherches concernant des systèmes de propagation encore peu compris. Dans une recherche de 2015, des chercheurs ont utilisé des automates cellulaires afin de modéliser et simuler la propagation et l'évolution de maladies [14]. Dans une autre étude de 2019 ils sont utilisés pour simuler la propagation d'incendies de forêt [15].

2.2.1.2 Simulation de foules

On utilise les systèmes multi-agents pour simuler des interactions existant entre agents autonomes. Le but est de déterminer l'évolution de ce système afin de prévoir l'organisation qui en résulte. Ils permettent notamment d'expérimenter des scénarios qui ne seraient pas réalisables sur des populations réelles, que ce soit pour des raisons techniques ou éthiques. Ce qui importe c'est le comportement d'ensemble et non pas le comportement individuel.

Ils sont par exemple utilisés dans le monde de l'industrie, dans les jeux vidéo et dans l'animation, avec notamment le logiciel MASSIVE (Multiple Agent Simulation System in Virtual Environment, littéralement « Système de simulation multi-agents dans un environnement virtuel »). Celui-ci permet de simuler des foules. Il a été développé à l'occasion de la trilogie cinématographique du Seigneur des Anneaux et est utilisé dans de nombreux films.

Ils sont aussi utilisés dans le monde de la recherche, par exemple, pour l'analyse de mouvements de foules, de mouvements de paniques, du trafic routier, de la propagation de maladies, etc.

Ci-dessous est un exemple de système multi-agents développé de manière à simuler des poissons formant des bancs. Sur cette image les traits fins sont des agents qui sont effrayés par les agents représentés par des traits épais. On peut voir apparaître des comportements de formation de bancs organisés par type d'agent (gros sur la droite et petits sur la gauche) et un comportement de fuite qui s'amorce de la part des petits poissons devant l'arrivée du banc de gros poissons. Cette image est extraite d'un de mes projets personnels [16].

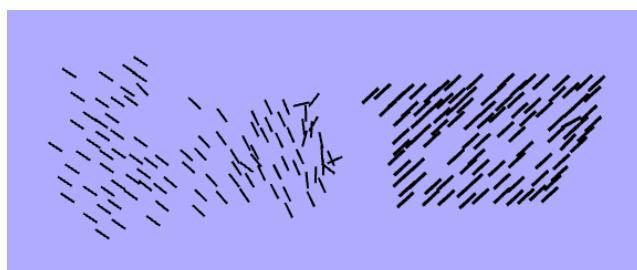


FIGURE 2.3 – Formation de deux bancs de poissons organisés par « espèces »

Il est intéressant de constater que lorsqu'une minorité des petits poissons est capable de voir le danger arriver et commence à fuir, c'est la totalité du banc qui débute un demi-tour. C'est ce genre de comportement qui permet à des poissons réels, tels que les sardines, d'augmenter les chances de survies d'un maximum d'individus en cas d'attaque d'un prédateur.

2.2.2 Algorithmes génétiques

Les algorithmes génétiques sont inspirés du processus de sélection naturelle et permettent d'obtenir une ou plusieurs solutions répondant à un problème d'optimisation.

Initialement, une population est générée aléatoirement, où chaque individu représente une solution du problème. Les individus possèdent un génoype qui leur est propre, c'est celui-ci qui définit les caractéristiques de chacun. Une sélection est ensuite effectuée parmi la population afin de ne sélectionner que les plus performants. Lorsque deux individus sont sélectionnés, un entrecroisement (ou cross-over) est effectué sur leurs génotypes. Ce croisement est une simulation d'un phénomène génétique qui contribue au brassage génétique lors de la reproduction. Ce croisement entraîne la création d'un nouveau génoype, sur lequel on peut appliquer des mutations aléatoires. Ainsi, une nouvelle population est générée à partir de la précédente, c'est donc une nouvelle génération. Ce processus est ensuite répété de nombreuses fois afin de simuler le principe d'évolution. À terme, les individus deviennent plus performants et tendent vers la solution optimale pour la résolution du problème posé.

Les problèmes pouvant être traités à l'aide de ce type d'algorithmes sont très variés, leur domaine d'application le plus impressionnant est sans doute l'optimisation de designs. La solution logicielle « Generative Design »,

de la compagnie AUTODESK, permet à un utilisateur de définir ses besoins en termes d'aérodynamisme, de solidité, de type de matière, etc. Le logiciel utilise ensuite un système d'algorithme génétique pour générer un ensemble de possibilités répondant aux besoins, qu'il propose à l'utilisateur. Le logiciel permet de générer des formes complexes permettant de mieux répondre aux contraintes que n'importe quel design qu'un être humain aurait pu imaginer.



FIGURE 2.4 – Châssis de drone léger et résistant, généré et proposé par Generative Design

2.2.3 Recherche de chemin

Le problème de la recherche de chemin consiste à trouver une solution pour se déplacer dans un environnement entre un point de départ et un point d'arrivée en prenant en compte différentes contraintes.

Il existe des algorithmes de recherche de chemin très célèbres, avec notamment l'algorithme de Dijkstra et l'algorithme A*. Le premier est utilisé pour déterminer un chemin optimal, il est par exemple utilisé dans la pratique dans des protocoles de routage au sein de réseaux informatiques. Le second présente l'avantage d'être plus rapide, mais n'offre cependant pas la certitude de l'optimalité.

Les algorithmes de recherche de chemin sont énormément utilisés dans le domaine de la robotique mobile et dans le domaine du jeu vidéo dans lesquels des entités doivent se déplacer.

À un niveau plus avancé, les outils de GPS, tels que Google Maps, utilisent aussi des algorithmes de recherche de chemin qui sont appliqués à une transformation de la carte sous forme de graphe.

2.2.4 Modèles de MARKOV cachés

Un modèle de MARKOV caché est un modèle statistique dans lequel le système modélisé est supposé être un processus de MARKOV de paramètres inconnus. C'est un processus stochastique possédant la propriété de MARKOV. Cette propriété est vérifiée si et seulement si la distribution conditionnelle de probabilité des états futurs ne dépend que de l'état présent et non pas des états passés. Concrètement, c'est un système sans mémoire.

Les modèles de MARKOV cachés sont beaucoup utilisés en classification d'images [17], segmentation d'images [18], traitement automatique du langage naturel [19], ou encore en modélisation de séquences en biologie [20].

2.2.5 L'apprentissage profond

L'apprentissage profond, appelé deep learning en anglais, est un ensemble de méthodes d'apprentissage automatique (machine learning) dont le but est de modéliser des données à un fort niveau d'abstraction afin de répondre à des problèmes complexes. Ces méthodes ne furent mises en pratique que récemment, grâce à la forte amélioration des capacités de calcul des ordinateurs. Elles ont permis des avancées spectaculaires dans les domaines de l'analyse de signaux sonores ou visuels et notamment de la reconnaissance faciale, vocale, de la vision par ordinateur, du traitement automatique du langage, etc. Les méthodes d'apprentissage profond susciteront l'intérêt de l'industrie dans les années 2000, avec notamment les GAFA (GOOGLE, APPLE, FACEBOOK et AMAZON), ce qui a permis des avancées très rapides ces dernières années.

Cette section n'est absolument pas une liste exhaustive de tous les concepts existants dans le domaine de l'apprentissage profond, cependant, les concepts les plus utilisés et donnant lieu aux plus grandes avancées sont abordés via la présentation d'une ou plusieurs de leurs applications.

2.2.5.1 L'émergence d'agents coopératifs complexes

Le 30 mai 2019, la société DEEPMIND, détenue par la société ALPHABET, a présenté un système multi-agent capable de jouer au jeu « Quake III Arena Capture the Flag » avec des performances supérieures à celles d'excellents joueurs [21].

Dans ce jeu il y a deux équipes, chaque équipe possède une base dans laquelle se trouve un drapeau à leur couleur respective. Le but de chaque équipe est de coopérer pour aller voler le drapeau adverse afin de le ramener à la base alliée, tout en défendant le drapeau allié. Les joueurs peuvent aussi tirer sur ceux de l'équipe adverse afin de les faire réapparaître dans leur base. Les règles sont simples, mais la dynamique de jeu entre les équipes est complexe.

Dans le monde réel, chaque humain possède ses propres objectifs et possibles actions, mais nous sommes tout de même capables de nous allier pour former des équipes, des sociétés, en présentant une intelligence collective impressionnante. C'est quelque chose qui est appelé l'apprentissage multi-agent, c'est un problème extrêmement complexe, car avec des agents apprenants tous en parallèle, le monde évolue constamment. Afin d'investir ce problème, beaucoup de scientifiques s'intéressent aux jeux vidéo 3D multijoueurs à la première personne. Le fait que ces jeux soient à la première personne complexifie beaucoup le problème, car la vision du monde n'est pas omnisciente. L'environnement est vu du point de vue d'un individu unique. Il faut donc être capable de créer une représentation imaginaire de l'environnement non observable afin de pouvoir anticiper un maximum de situations.

Dans cette expérience, les agents de DEEPMIND ont dû apprendre, à partir de rien, comment voir, agir et coopérer dans un environnement vu à la première personne, le tout avec pour seule indication de leur performance le score de l'équipe. L'apprentissage est basé sur le principe d'apprentissage par renforcement, c'est-à-dire que l'on indique à l'agent si sa performance est bonne ou mauvaise et celui-ci en tire une expérience lui permettant de s'améliorer.

Le fonctionnement de chaque agent est basé sur un réseau de neurones récurrents. Contrairement à un réseau de neurones classique (qui nécessite des données d'entrées ayant une taille définie bien précise), ces derniers sont adaptés pour des données d'entrées de taille variable, en particulier pour des séries temporelles. Ils sont utilisés dans la reconnaissance de la parole ou de l'écriture manuscrite par exemple. Ce genre de réseau est adapté dans ce cas, car les agents doivent effectuer des actions en réponse à des séries d'actions effectuées par leurs alliés et les ennemis, il leur faut donc un semblant de mémoire des actions effectuées auparavant.

Pour entraîner les agents, ceux-ci jouent en équipes d'agents contre agents et apprennent tout d'eux-mêmes, ils sont capables d'apprendre seuls des tactiques connues des joueurs de haut niveau. Les agents ainsi entraînés sont aussi capables de s'adapter pour jouer sur des cartes inconnues et même à d'autres modes de jeux.

2.2.5.2 AlphaStar

Le 19 janvier 2019, AlphaStar bat le joueur Grzegorz KOMINCZ, appelé « MaNa » et l'un des meilleurs joueurs du monde, 5 parties à 0, sur le jeu StarCraft II [22]. AlphaStar est développé par la société DEEPMIND. StarCraft II est aujourd'hui considéré comme étant le jeu de stratégie en temps réel le plus complexe au monde et est devenu récemment l'un des « grands challenges » de l'intelligence artificielle. Le mode de jeu le plus célèbre, et celui sur lequel AlphaStar est entraîné est le mode 1v1, c'est-à-dire deux joueurs qui s'affrontent l'un contre l'autre, chacun devant construire sa base dans l'objectif de produire le plus de ressources possible tout en optimisant sa défense et son attaque.



FIGURE 2.5 – Vue de la base construite par AlphaStar pendant une des parties

AlphaStar est le premier programme à être capable de jouer à StarCraft II sans aucune modification des règles et directement depuis une vue brute des pixels du jeu. Le programme est basé sur un réseau de neurones profond, entraîné utilisant un apprentissage par renforcement.

Un réseau de neurones est composé de plusieurs niveaux de neurones artificiels, ces niveaux sont reliés entre eux par l'équivalent, grandement simplifié, des synapses présentent dans notre cerveau. Jusqu'à récemment, les réseaux de neurones dépassaient très rarement plus de 3 ou 4 niveaux, et ce, pour des raisons techniques. Mais des découvertes ont permis d'augmenter ce nombre de niveaux à plusieurs dizaines ; le principal facteur limitant étant actuellement la puissance de calcul nécessaire pour l'entraînement de ces réseaux. Un réseau de neurones profond est donc un réseau de neurones utilisant des méthodes permettant d'augmenter sa profondeur, c'est-à-dire son nombre de niveaux. Cela permet d'obtenir un système capable d'une grande abstraction et donc de montrer un raisonnement permettant de répondre à des problèmes très complexes.

StarCraft II est un jeu nécessitant de nombreuses optimisations à court terme qui se révèlent plus tard avoir un fort impact sur le long terme. Le programme doit donc être capable de trouver un équilibre entre les objectifs à court et long terme et de s'adapter à des situations inattendues. Contrairement à des jeux comme les échecs ou le jeu de go, StarCraft II nécessite de prendre des actions en temps réel et le joueur ne peut pas voir la totalité de son environnement à tout moment. Les contraintes de ce jeu sont donc immenses et similaires aux contraintes auxquelles les êtres humains sont confrontés continuellement. Développer un programme capable de s'adapter pour résoudre des situations comparables est donc une avancée scientifique de grande ampleur.

2.2.5.3 Autres avancées de DEEPMIND

La société DEEPMIND conduit de très nombreuses recherches dans le domaine de l'apprentissage profond, comme vu avec les deux précédentes sections.

On peut encore citer AlphaGo, le premier programme qui a battu un joueur humain professionnel et, par la même occasion, un champion du monde au jeu de Go, M. Fan HUI, en octobre 2015. Ce programme fonctionne grâce à un réseau de neurones profond et une méthode d'apprentissage par renforcement.

En 2017, la compagnie introduit AlphaZero, un programme capable de jouer au jeu de Go, aux échecs et au shōgi (jeu de société japonais) à un niveau sans précédent sur chacun des jeux.

En 2018, DEEPMIND présente AlphaFold, programme fonctionnant aussi grâce à un réseau de neurones profond et qui permet, à partir d'une séquence de protéines, de déterminer un modèle 3D de la structure de la protéine résultante. Cela à un fort intérêt pour les recherches scientifiques dans le domaine médical et notamment pour les recherches concernant les maladies d'Alzheimer ou de Parkinson par exemple.

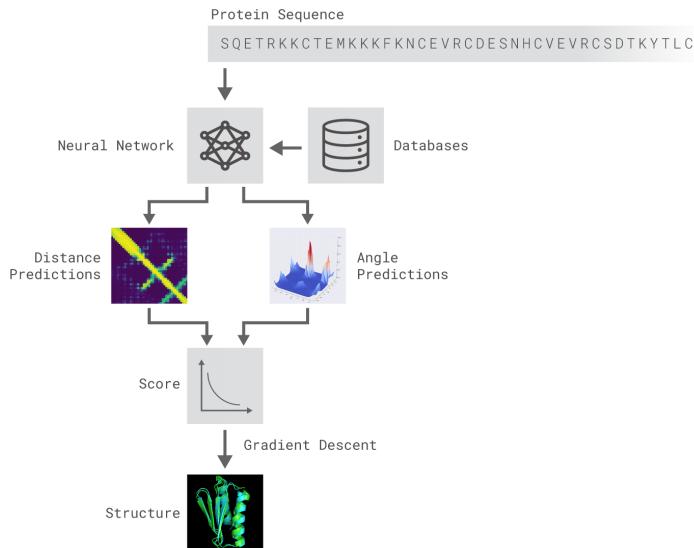


FIGURE 2.6 – Fonctionnement d’AlphaFold pour la génération d’un modèle 3D d’une protéine à partir d’une séquence protéique

2.2.5.4 Pilote automatique de TESLA v9.0

Depuis 2018, la dernière version du pilote automatique des véhicules de la marque TESLA est la version 9.0. Elle suit la version 8.1. Il est estimé qu’elle présente une augmentation des capacités de 400% par rapport à la version précédente. Il est complexe de trouver des informations sur le fonctionnement technique du programme de pilotage automatique de TESLA. En effet, il ne s’agit pas ici d’expériences, comme pour la société DEEPMIND, mais d’un système utilisé à un niveau industriel et sur lequel la vie des utilisateurs dépend.

On sait tout de même que le pilote automatique est basé sur un réseau de neurones capable de prendre en compte les données recueillies par les huit caméras entourant le véhicule simultanément. Il utilise un système de détection des objets très précis, ainsi il peut comprendre l’environnement du véhicule.

En termes de détection d’objets, segmentation d’images et reconnaissance d’images, les réseaux de neurones les mieux adaptés sont les réseaux neuronaux à convolution, il est donc très probable que le réseau du pilote automatique soit au moins partiellement de ce type.

Dans l’image ci-dessous on peut voir la détection des voitures et piétons ainsi que la segmentation de l’image effectuée pour détecter où se trouve la route à suivre.

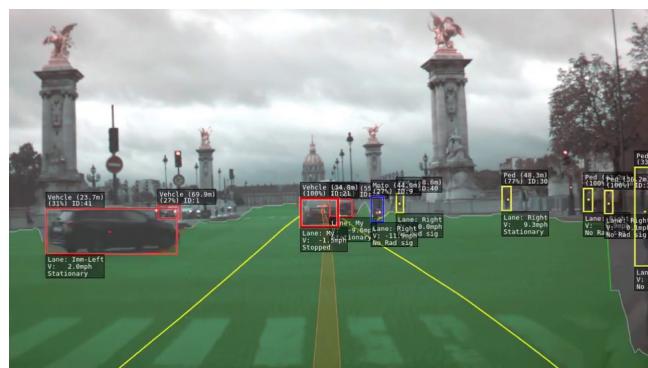


FIGURE 2.7 – Visualisation de la détection d’objet du pilote automatique de TESLA

Les réseaux de neurones à convolution diffèrent des autres par l’utilisation d’un motif de connexion entre

les neurones inspirés du cortex visuel des animaux. C'est pour cette raison qu'ils sont si performants dans le traitement des images.

2.2.5.5 GauGAN de NVIDIA

La société NVIDIA, spécialisée dans la création de matériel informatique, réalise de nombreuses avancées dans le domaine de l'apprentissage profond. En 2019, la compagnie a présenté et mis à disposition un outil permettant de transformer de simples croquis en paysage réaliste. Sur l'image ci-dessous on peut voir le croquis sur la gauche et le paysage généré automatiquement sur la droite.

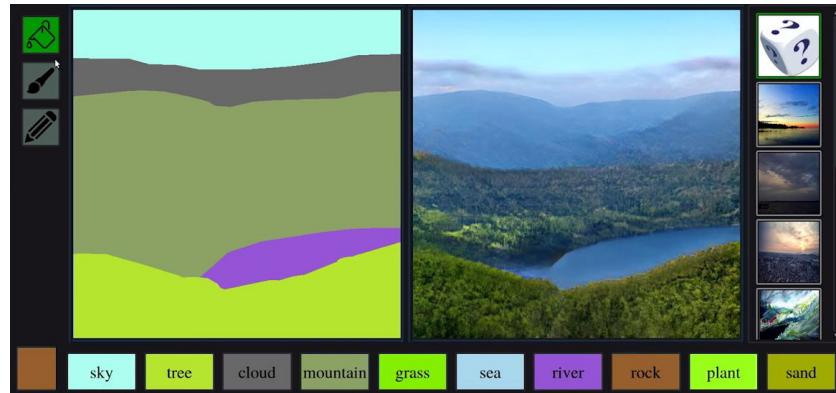


FIGURE 2.8 – Transformation d'un dessin simpliste en paysage en utilisant GauGAN

Cet outil se base sur un concept récent de l'apprentissage profond : les « generative adversarial networks » (GAN), que l'on peut traduire par réseaux antagonistes génératifs.

Ce nouveau type d'algorithme fut introduit en 2014 [23]. C'est un système dans lequel deux réseaux de neurones, généralement à convolution, sont placés en compétition, l'un est un générateur et l'autre est un discriminateur. Le premier génère un échantillon, par exemple une image, le second essaie ensuite de déterminer si l'échantillon résulte du générateur ou est un échantillon réel. L'entraînement d'un tel modèle a pour effet de permettre la génération d'échantillons très réalistes. Les mises en application sont encore rares et relèvent généralement plus du domaine artistique que scientifique.

L'entreprise a aussi développé un outil permettant de faire disparaître un élément d'une photo automatiquement sur le même principe et l'a présenté par la même occasion.

D'un point de vue technique, il faut savoir que la meilleure manière d'entraîner un modèle d'apprentissage profond est de faire effectuer les calculs à une ou des carte(s) graphique(s), plutôt qu'à un ou des processeur(s). Les cartes graphiques conviennent très bien à la réalisation de nombreux calculs en parallèle, contrairement aux processeurs, et donc conviennent parfaitement aux types de calculs requis lors de l'entraînement d'un réseau de neurones. Les deux principaux producteurs de cartes graphiques au monde sont NVIDIA et AMD. C'est NVIDIA qui s'est pour le moment le plus déporté par son investissement dans la recherche en apprentissage profond. NVIDIA a développé de nombreux outils permettant de grandement faciliter le développement de systèmes d'apprentissage profond sur ses cartes graphiques. Ainsi, les chercheurs du domaine travaillent donc quasiment exclusivement avec cette compagnie.

2.3 Que doit offrir un ERP ?

Le terme ERP vient de l'anglais « Enterprise Ressource Planning » et il a été traduit en français par l'acronyme PGI : Progiciel de Gestion Intégré. Il se définit comme un groupe de modules relié à une base de données unique. C'est un outil utilisé par la totalité des employés d'une entreprise. Il permet de faciliter les flux d'informations et de coordonner toutes les ressources et activités de la compagnie. Les fonctions qui sont typiquement supportées par un tel outil incluent la fabrication, l'inventaire, les expéditions, la logistique, la distribution, la facturation, la comptabilité, la gestion des clients et des ressources humaines. Il offre un environnement de travail uniforme et consistant pour tous les employés.

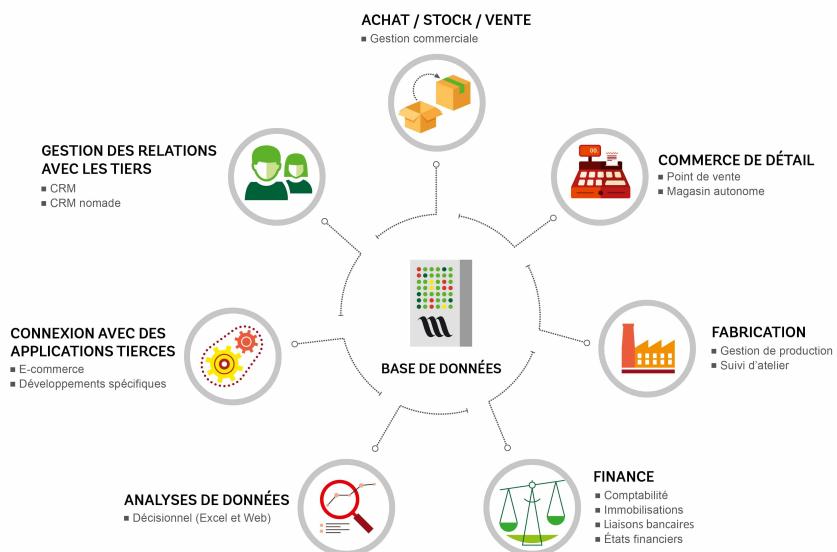


FIGURE 2.9 – Les services couverts par la plupart des ERP

De toutes les technologies pouvant être déployées au sein d'une compagnie, à l'heure actuelle c'est un ERP qui a le plus de potentiel et l'impact le plus direct sur la réduction des coûts dans la compagnie. Une étude de 2010, menée par le groupe ABERDEEN [24], sur de petites et moyennes entreprises a permis de mettre en avant les facteurs les plus importants pour l'utilisation d'un ERP au sein de celles-ci. La distribution de ces facteurs est affichée ci-dessous.

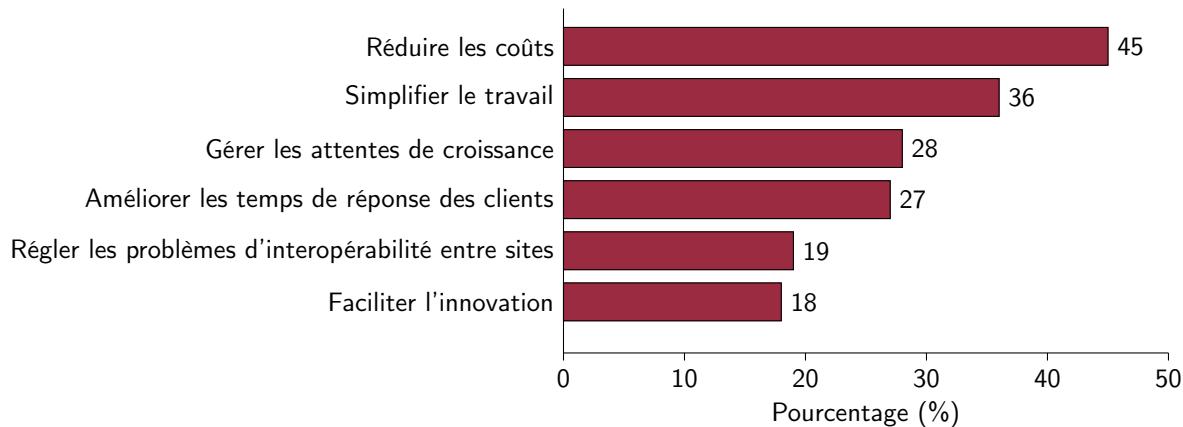


FIGURE 2.10 – Distribution des principaux facteurs d'adoption d'un ERP dans une PME

Au sein de DISA avant l'année 2000, l'entreprise avait besoin d'un système permettant de mieux suivre ses données (clients, devis, commandes, factures) et également un suivi de la fabrication, jusqu'alors inexistant. La compagnie décida donc de faire l'acquisition de l'ERP ASAP, produit par la société INNETIS. Sa mise en place dura deux ans et l'outil devint rapidement indispensable au bon fonctionnement de DISA. Suite au vieillissement d'ASAP, le développement de plus en plus d'applications sous DISANET entraîna la perte des avantages de l'utilisation d'un ERP et c'est pour cette raison que le service informatique commença le développement de SIGMA.

2.4 Les applications possibles de l'intelligence artificielle au sein des ERP

Maintenant que nous avons vu les applications possibles des principaux concepts de l'intelligence artificielle et les services devant être proposés par les ERP, nous pouvons réfléchir aux applications possibles de l'intelligence artificielle concernant les ERP.

2.4.1 Gestionnaire Électronique des Documents

Au sein d'un ERP, il est très courant de rencontrer un système de gestion électronique des documents, dont le rôle est de gérer les documents électroniques de la compagnie. Nous avons d'ailleurs développé une GED au sein de SIGMA au cours de mon apprentissage.

Il existe quatre étapes majeures dans la gestion électronique des documents : acquisition, traitement, stockage et diffusion.

- L'acquisition des documents au sein de la GED peut s'effectuer par l'intégration de documents préalablement scannés, ou par la production automatique de documents via un logiciel.
- Le traitement des documents consiste en leur indexation. C'est la description du document qui permettra de le retrouver facilement par la suite.
- Le stockage des documents doit être effectué sur un support prenant en compte le volume potentiel de documents que la GED devra gérer dans le futur, il doit être sécurisé de manière à garantir la sécurité des données.
- La diffusion des documents doit permettre aux utilisateurs de retrouver les documents dont ils ont besoin facilement et rapidement.

Différentes utilisations de l'intelligence artificielle peuvent intervenir lors de ces quatre étapes afin d'améliorer les capacités d'une GED.

2.4.1.1 Classification de documents

L'un des domaines dans lequel l'intelligence artificielle excelle et s'améliore rapidement de nos jours est la classification, que ce soit des documents, images, textes, mails, etc.

Dans une GED, avec une architecture et une organisation précise prédéfinie, il peut être intéressant d'utiliser un système permettant de classer les documents importés au bon endroit au sein de la structure sans que l'utilisateur n'ait besoin d'intervenir. Afin de fonctionner correctement, un tel système nécessite de nombreuses données d'entraînement. C'est grâce à ces dernières qu'il pourra apprendre à classifier correctement les documents au sein de l'architecture. Un tel système ne peut donc être mis en place que sur une GED ayant été classée manuellement depuis un certain temps, afin d'avoir plusieurs exemplaires de chaque type de documents déjà classés au sein de l'architecture.

Il existe de nombreux concepts d'intelligence artificielle permettant de classifier des fichiers. Les plus poussés sont actuellement les réseaux de neurones à convolution qui permettent de classifier des images avec des taux d'erreur extrêmement bas. Cependant, nous voulons ici classifier des documents et nous allons voir que nous pouvons utiliser des concepts beaucoup plus basiques et requérant bien moins de puissance de calcul pour atteindre un résultat satisfaisant.

Préparation des documents

Pour classer un document, les êtres humains sont capables de repérer quasiment instantanément les détails permettant d'en identifier la nature exacte, que ce soit par la lecture de certains mots clés ou par le repérage

d'un logo par exemple. L'objectif ici est de trouver un concept d'intelligence artificielle qui soit capable de faire la même chose.

Il y a deux possibilités majeures sur la manière dont le système peut traiter des documents.

- La première serait de considérer les documents comme des images. Le principal avantage serait que si des documents sont principalement discernables de par leurs caractéristiques graphiques plus que de par leur contenu, le système serait capable de les classifier sans difficulté. Cependant, le cas inverse se présente également et comme nous traitons ici des documents généralement administratifs, il est plus courant que ces documents aient une charte graphique semblable et se discernent principalement de par leur contenu. En plus de cela, les documents n'ont pas toujours le même nombre de pages. Or il est très préférable que toutes les images à classifier aient les mêmes dimensions pour que ce genre de systèmes fonctionnent correctement.
- La seconde possibilité est de préalablement extraire le texte des documents afin de pouvoir se baser sur le contenu de ceux-ci pour les classer. Pour extraire le texte brut d'un document il existe de nombreuses méthodes d'OCR, en français reconnaissance optique de caractères, qui sont robustes et utilisables sur des pdf complets en une simple ligne de code ; il n'est donc pas intéressant de les décrire plus que cela. Avant de donner tout le texte d'un document à un système de classification, il est cependant nécessaire d'effectuer des traitements permettant d'en extraire des « features », « caractéristiques » en français. C'est sur ces caractéristiques que le système pourra se baser afin de classer les documents. Une démarche intéressante serait par exemple d'utiliser la méthode de pondération TF-IDF (term frequency-inverse document frequency). Celle-ci est très utilisée en fouille de textes, car elle permet de mesurer l'importance de chaque terme employé dans un texte. Avec cette méthode on pourrait, par exemple, extraire les cinq termes les plus importants dans le document et utiliser uniquement ces cinq termes afin de classer ce dernier.

Dans notre cas, la meilleure solution serait sans doute d'utiliser la seconde méthode, consistant en l'extraction des principales caractéristiques directement depuis le texte de chaque document. Avec cette méthode on peut donc générer une base de données reliant les termes les plus importants avec chaque document dont ils sont extraits. Ainsi, nous posséderions une base de données avec tous les documents déjà présents dans la GED, ces données serviraient de données d'entraînement au système de classification. Une fois le système entraîné, lorsqu'un nouveau document sera importé, il suffira d'en extraire les caractéristiques et de les envoyer dans le système pour que le document soit classé automatiquement.

Algorithmes de classification

Une fois les données prêtées, nous pouvons analyser les documents en comparant le nombre de termes en communs parmi leurs caractéristiques extraites. C'est pour effectuer cette comparaison que l'algorithme de classification va entrer en jeu. Sachant que les caractéristiques extraites ne seront pas très nombreuses on peut utiliser des classificateurs linéaires relativement basiques, tels que les SVM (« support vector machine » traduit par « machine à vecteurs de support »), la classification naïve bayésienne, ou même la méthode des k plus proches voisins.

Les trois méthodes précédemment citées sont des concepts de base de l'intelligence artificielle qui sont enseignés en tant qu'introduction dans beaucoup de cours, portants sur l'intelligence artificielle. Cependant, cela ne signifie pas qu'ils sont inutiles ou inefficients ; ils sont au contraire, simple à développer et à appliquer, et peuvent être utilisés lorsque la situation s'y prête. Comme déjà précisé, nous nous basons sur un nombre de caractéristiques très limité pour cette classification. Ces classificateurs sont donc adaptés à notre cas de figure avec des avantages et inconvénients pour chacun d'entre eux.

- Le principe des machines à vecteurs de support (séparateurs à vaste marge) est de placer des frontières entre les catégories. On peut en visualiser le fonctionnement facilement si on prend un exemple simple de

classification ne contenant que deux classes. Considérons les données suivantes, dans lesquelles les deux classes sont « ronds » et « triangles » et la donnée que l'on veut classer dans une de ces dernières est le carré rouge.

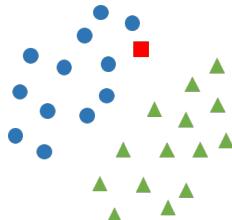


FIGURE 2.11 – Problème de classification en deux dimensions

En tant qu'être humain, il est trivial de déterminer que le carré devrait être classé avec les ronds, au vu de l'organisation présente. C'est cependant nettement plus complexe pour une machine. Le but du SVM va être de déterminer une droite « frontière » qui permette de séparer les données en deux classes. Le SVM choisit une frontière qui maximise la marge, c'est-à-dire qu'elle est le plus éloignée possible des deux classes. Cela permettra de classer avec plus de précision les futures données inconnues. Par exemple, dans la figure ci-dessous on voit deux frontières noires en pointillés qui permettent de classifier correctement les données d'entraînement. En revanche elles ne maximisent pas la marge entre les deux classes, ce qui peut donner lieu à de mauvaises classifications dans le futur. La frontière maximisant la marge, qui sera donc déterminée par le SVM, est symbolisée en rouge.

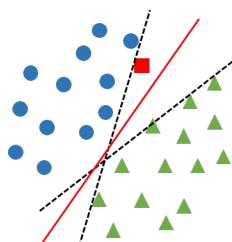


FIGURE 2.12 – Détermination de la frontière maximisant la marge par le SVM

Les SVM peuvent classifier des données avec plus de deux dimensions, ils s'appliquent donc correctement à notre cas.

Avant de pouvoir utiliser un SVM pour classer des données il faut déjà que celui-ci soit entraîné. Cet entraînement s'effectue grâce aux données déjà présentent au sein de la GED (plus le nombre de classes et de caractéristiques sera élevé et plus le temps et les besoins en puissance de calcul augmenteront). Cela signifie que pour entraîner un SVM il faut utiliser une machine assez performante. Dans notre cas, le nombre de classes de documents et de caractéristiques utilisées devrait rester assez limité pour qu'une machine récente soit capable de réaliser cette tâche. L'avantage est que, une fois l'entraînement du SVM effectué, il peut être utilisé sur un ordinateur extrêmement peu puissant afin de classer de nouvelles données. Les utilisateurs de la GED pourront donc utiliser n'importe quel type de matériel.

- La classification naïve bayésienne est basée sur le théorème de BAYES et utilise des formules de statistiques.

Le théorème de BAYES est le suivant :

$$P(h|d) = \frac{(P(d|h) \cdot P(h))}{P(d)}$$

Où :

- $P(h|d)$, est la probabilité de l'hypothèse h sachant la donnée d .
- $P(d|h)$, est la probabilité de la donnée d sachant que l'hypothèse h est vraie.
- $P(h)$, est la probabilité que l'hypothèse h soit vraie.

- $P(d)$, est la probabilité de la donnée d .

Dans notre cas, nous voulons calculer toutes les probabilités du type $P(\text{caractéristique}|\text{classe de document})$. Par exemple, la probabilité $P(\text{livraison}|\text{bon de livraison})$ sera élevée puisque le terme « livraison » sera une caractéristique forte des documents de classe « bon de livraison », tandis que $P(\text{facture}|\text{bon de livraison})$ sera basse.

On calcule ainsi toutes les probabilités d'apparition des caractéristiques pour chaque classe. Ensuite lorsque nous voulons classer un nouveau document il suffit d'extraire ses caractéristiques et d'observer les probabilités du type $P(\text{classe}|\text{caractéristique})$ et la plus haute donnera la classe du document.

- La méthode des k plus proches voisins est une méthode qui à l'avantage d'être plutôt intuitive et donc simple à comprendre et implémenter. Le but est ici de comparer les caractéristiques du nouveau document avec les caractéristiques de tous les documents déjà présents dans la GED. Une fois cette comparaison effectuée, le document le plus « proche » du nouveau est celui qui aura le plus de caractéristiques en commun avec lui. Le nouveau document est classé dans la même classe que le voisin le plus proche. L'explication ci-dessus ne prend en compte qu'un seul voisin le plus proche. Cependant pour de meilleures performances il peut être intéressant de sélectionner, par exemple, les cinq voisins les plus proches et attribuer au nouveau document la classe apparaissant le plus souvent parmi ces derniers. Le nombre de voisins considérés correspond à la variable k . Il est préférable de toujours choisir une valeur de k impaire afin de toujours pouvoir déterminer une classe sans souci.

Le principal désavantage de cette méthode est qu'elle ne possède pas de phase d'apprentissage à proprement parler. À la place la totalité des échantillons présents dans la base sont parcourus à chaque ajout de nouveau document. Cela est fortement contraignant, car plus le nombre de documents présents dans la GED sera grand, plus le processus de classification des nouveaux sera long. Cela signifie aussi que les utilisateurs de la GED doivent posséder un ordinateur capable d'une puissance de calcul suffisante pour classer chaque nouveau document.

Pour choisir une méthode de classification parmi les trois proposées ci-dessus, il faut analyser leurs défauts et avantages. En termes de puissance de calcul nécessaire, le choix se portera sans nul doute sur la classification naïve bayésienne. Celle-ci nécessite juste le calcul d'un nombre fortement limité de probabilités que n'importe quelle machine peut assumer, la classification des nouvelles données se fait ensuite instantanément. En revanche, le SVM n'est un choix valable qu'en possession d'une machine permettant son entraînement. La méthode des k plus proches voisins est, quant à elle, peu envisageable, sachant qu'elle nécessite que tous les utilisateurs possèdent une machine puissante, de plus, son processus de classification est lent.

Afin d'aller plus loin, et d'éventuellement améliorer la précision de la classification, on peut imaginer un système se basant à la fois sur des caractéristiques provenant du texte et d'autres provenant de la charte graphique du document.

2.4.1.2 Partitionnement de documents

Il y a un type d'apprentissage que nous n'avons pas encore abordé, l'apprentissage non supervisé. Le principe pour le système utilisant ce type d'apprentissage est de tenter de trouver des structures sous-jacentes à partir de données non étiquetées. Dans notre cas, les données non étiquetées correspondent aux documents bruts, non classés.

Un apprentissage non supervisé doit permettre de mettre en évidence des caractéristiques communes à des données, afin de les organiser par groupes ne contenant que des données similaires.

Dans notre cas, un système de partitionnement des documents pourrait être utile pour organiser automatiquement les documents. Un avantage est que ces systèmes nécessitent des données qui n'ont pas préalablement été classées.

Les systèmes de partitionnement de documents sont beaucoup utilisés pour les moteurs de recherches. En effet, il est courant d'obtenir des milliers de pages de résultats pour une requête et le partitionnement des

résultats en catégories permet de déterminer ceux qui sont le plus susceptibles d'intéresser l'utilisateur. Dans notre cas on pourrait par exemple imaginer un système de recherche de documents au sein de la GED qui permette de trier les documents sur les classes générées par le système. On pourrait même utiliser un système de partitionnement de documents en plus d'un système de classification et ainsi effectuer un partitionnement sur les factures qui permettrait de les regrouper automatiquement par clients, etc. Le partitionnement de documents pourrait donc s'employer à l'importation de nouveaux documents ainsi que durant les recherches avancées des utilisateurs. Les utilisations possibles sont multiples.

Afin d'effectuer un partitionnement sur des documents, il est nécessaire de faire certaines préparations préalables. Au même titre que pour la classification il faut nettoyer le texte de la ponctuation et des « mots vides », avant d'appliquer la méthode TF-IDF abordée plus tôt, ou une méthode comparable. On peut ensuite commencer le processus de partitionnement sur les caractéristiques ainsi extraites.

Des recherches sont en cours dans le domaine du partitionnement afin d'y appliquer des méthodes d'apprentissage profond, mais nous allons ici voir des méthodes plus courantes. Il existe actuellement quatre catégories d'algorithme de partitionnement de données.

Le partitionnement en k -moyennes

Cette méthode permet de partitionner des données en k groupes, de façon à minimiser la fonction de distance entre les données d'un même groupe.

Pour cette méthode il est nécessaire de choisir à l'avance la valeur de k qui détermine le nombre de groupes. Lorsque le nombre de groupes voulus est déjà connu, il suffit de renseigner k en conséquence, mais lorsque ce n'est pas le cas il faut appliquer une méthode pour le déterminer.

La méthode généralement appliquée est la « elbow method », la méthode du coude. Elle consiste à tracer une courbe exprimant les valeurs de k testées en fonction de la variance obtenue. La variance se calcule en divisant la variance entre les groupes par la variance entre toutes les données. La courbe ainsi obtenue forme un coude et c'est la valeur k en ce point qu'il faut choisir.

Le principal désavantage est que pour déterminer la meilleure valeur de k il faut exécuter l'algorithme pour toutes les valeurs testées. Un second désavantage est que la méthode définit toujours les groupes avec des tailles similaires, même si cela n'est pas adapté à la situation, comme on peut le voir ci-dessous, le groupe vert devrait être plus important que les bleus et rouges.

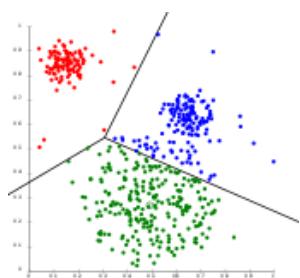


FIGURE 2.13 – Partitionnement de données en appliquant la méthode des k -moyennes

Cette méthode fait partie des méthodes de base qui sont enseignées pour la facilité de leur mise en place, mais les cas d'utilisations en situations réelles sont plutôt rares.

Le regroupement hiérarchique

Les méthodes de regroupement hiérarchique présentent l'avantage de ne pas avoir besoin de déterminer à l'avance le nombre de groupes à créer. Un autre avantage est qu'elles permettent une visualisation de la hiérarchie créée via un dendrogramme, qui est un type de diagramme.

Le but étant de placer chaque document existant au sein de l'architecture, il y a deux approches possibles : par agglomération ou par division.

- L'approche par agglomération consiste à initialement considérer chaque document comme un groupe à part entière. Ces derniers sont ensuite regroupés par leurs caractéristiques communes les plus proches jusqu'à ne former qu'un seul groupe.
- L'approche par division est l'exact opposé, on débute avec un seul groupe englobant la totalité des données. Ensuite, on sépare les données en groupes de plus en plus petits en fonction des caractéristiques, jusqu'à ce que tous les documents soient séparés dans des groupes individuels.

Ces méthodes permettent de générer une architecture hiérarchique que l'on peut facilement visualiser.

Ce type de regroupement est fortement efficace sur des données possédant des relations hiérarchiques fortes entre elles. Les documents stockés au sein d'une GED ne possèdent pas forcément de lien hiérarchique, par conséquent, ce type de regroupement n'est peut-être pas l'idéal pour notre cas de figure.

Le partitionnement basé sur les lois de probabilités

Ce modèle est basé sur des lois de probabilités. Les groupes sont définis comme des objets appartenant à une même distribution de probabilité.

Ce type de partitionnement diffère des autres de par le fait qu'ici tous les éléments font partie de chaque groupe, mais avec un degré d'appartenance différent. Par exemple, sur l'image ci-dessous on peut voir que les points bleus font partie du groupe rouge, cependant leur degré d'appartenance au groupe bleu est plus fort.

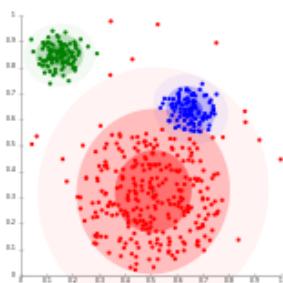


FIGURE 2.14 – Visualisation d'un partitionnement effectué en utilisant la loi de probabilité normale

Cette méthode est généralement combinée avec la méthode des k -moyennes, car il est nécessaire de déterminer préalablement le nombre k de groupes à former. Cependant, les deux méthodes combinées permettent de donner de très bons résultats.

Le partitionnement sur la densité

L'algorithme le plus populaire de cette catégorie est le DBSCAN (density-based spatial clustering of applications with noise), il se base sur les zones présentant une densité de points supérieure au reste de l'espace des données, pour les regrouper par groupes.

Cet algorithme permet de déterminer par lui-même le nombre de groupes à trouver. Il est capable de gérer les données aberrantes et ainsi est très peu sensible au bruit. C'est cette dernière raison qui en fait un algorithme autant utilisé dans le domaine. En revanche, il ne permet pas une définition précise des bords de chaque groupe, car il est courant que la densité des points soit moins haute à ces endroits.

Conclusion

Aucune méthode de partitionnement de données ne donne de résultats parfaits, mais il est important de noter que les applications de ces méthodes ne nécessitent généralement pas un partitionnement précis.

Selon le cas de figure, il peut être intéressant d'utiliser soit la méthode de partitionnement utilisant les distributions de probabilité, soit celle utilisant la densité.

La première garantie que toutes les données appartiennent à un groupe. Il peut donc être intéressant de l'utiliser à l'importation d'un nouveau document au sein de la GED afin de garantir qu'il soit classé dans un groupe.

Si l'on souhaite plutôt appliquer une méthode de partitionnement au moment d'une recherche par un utilisateur il peut être intéressant d'utiliser le partitionnement basé sur la densité. Puisque dans ce cas on souhaite diriger l'utilisateur vers les documents les plus susceptibles de l'intéresser.

2.4.1.3 Extraction de données pour saisie automatique

Il est très courant que des employés aient besoin de saisir dans un système informatique des informations qu'ils ont devant eux, sous format papier ou numérique. Cependant, il existe des méthodes qui peuvent permettre d'automatiser la saisie de ces données simplement en important un fichier au sein du système.

La compagnie ROSSUM, fondée en 2017, a créé l'outil ELIS. L'utilisation de cet outil est simple, on y importe une facture et l'outil est capable d'analyser automatiquement la structure de celle-ci afin d'extraire les informations utiles et de les ajouter directement au sein d'une base de données. Afin de rester compétitive, l'entreprise ne dévoile pas le fonctionnement de son logiciel, mais il est intéressant d'imaginer les méthodes d'intelligence artificielle employées.

Cette section est dédiée à la conception d'un système, inspiré d'ELIS, qui serait capable de gérer plusieurs types de documents.

Les différentes étapes au sein de ce système seraient les suivantes.

1. Détection automatique du type de document. Si la nature du document est mal déterminée, l'outil permet à l'utilisateur de le corriger. Après une correction de la part d'un utilisateur le système apprend de son erreur afin de continuellement s'adapter aux nouveaux documents importés.
2. Analyse de la structure du document. Ici l'outil annote sur le document les zones correspondantes aux informations à récupérer. L'utilisateur peut le corriger, ce qui permet encore une fois un apprentissage à partir des erreurs.
3. Application d'une méthode de reconnaissance de caractères afin de lire les informations présentent dans les zones délimitées à l'étape précédente.
4. Enregistrement des informations directement en base de données.

On peut voir ci-dessous un aperçu du fonctionnement de l'outil ELIS qui relie chaque champ à la zone du document détectée comme correspondante.

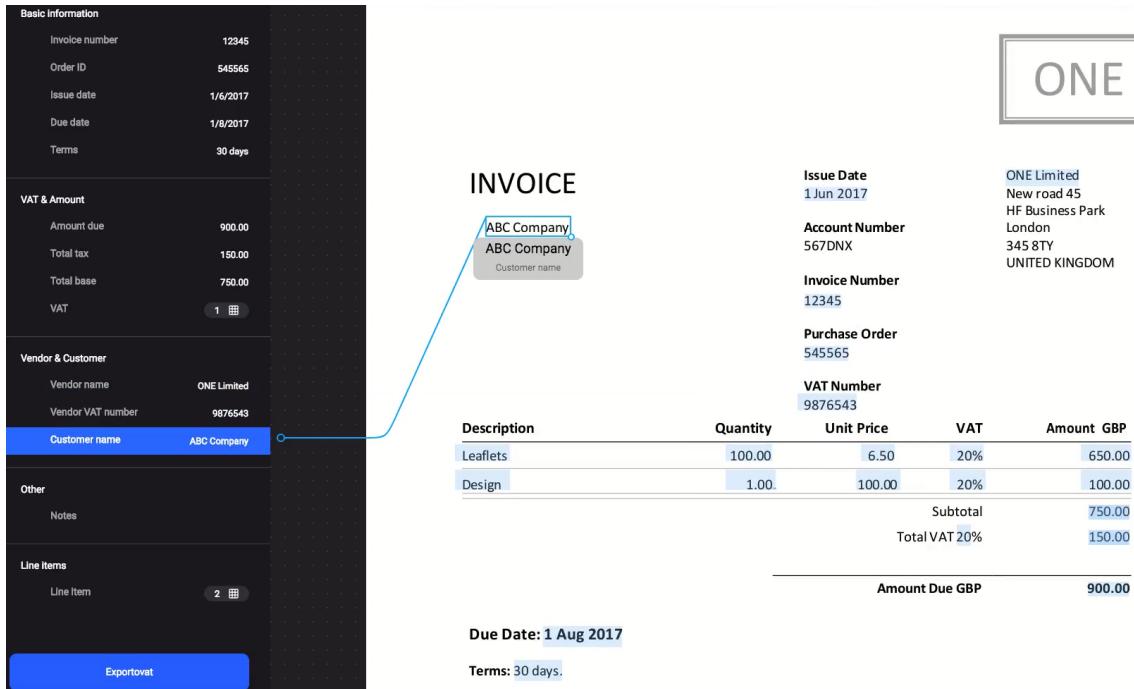


FIGURE 2.15 – Analyse de la structure d'une facture avec correction manuelle possible, avec l'outil ELIS

Détection automatique du document

Lorsque l'on parle de détection automatique de la nature du document, on parle en fait de classification, l'utilisation d'une des méthodes décrites précédemment est donc tout à fait envisageable dans ce cas.

Analyse de la structure du document

Nous avons vu précédemment que traiter un document en tant qu'image était plus complexe que d'en extraire le texte brut. Cela est vrai, mais le traitement en tant qu'image offre beaucoup plus de possibilités et n'entraîne aucune perte d'informations dans le processus. Comme nous voulons ici analyser la structure du document, il faut le traiter sous sa forme initiale, sous forme d'image. Pour réaliser cette analyse, il va donc être nécessaire d'utiliser des méthodes avancées d'apprentissage profond.

Comme précisé auparavant, ce sont les réseaux de neurones à convolution qui donne les meilleurs résultats concernant la reconnaissance d'images, la détection d'objet au sein d'une image et la segmentation d'images. Ce que nous voulons réaliser s'apparente à une détection d'objet au sein d'une image, où les objets à détecter correspondent aux zones où se trouvent les informations à extraire.

Il existe plusieurs manières de détecter et identifier des objets. Le système de détection YOLO (You Only Look Once) est actuellement le plus performant dans le domaine [25]. Il permet de gérer la détection d'objets au sein d'une image ainsi que leur classification, grâce à un réseau de neurones à convolution, en une seule étape. Auparavant, il fallait utiliser deux réseaux de neurones à convolution, l'un pour détecter les limites des objets, l'autre pour classifier chaque objet détecté, un par un. Le système YOLO est si efficient qu'il permet d'effectuer de la reconnaissance d'objets sur une vidéo en temps réel, à un rythme allant jusqu'à 45 images par secondes.

L'avantage de YOLO, qui pourrait permettre son fonctionnement dans notre cas, est qu'il identifie chaque objet en se basant sur l'image complète, ce qui lui permet de prendre en compte tout le contexte. Pour comprendre qu'une zone spécifique d'un document est importante et savoir quelles informations on peut y trouver,

il est nécessaire que le système considère le document dans sa globalité.

Les systèmes permettant la reconnaissance d'objets au sein d'une image ne sont pas pleinement adaptés à la reconnaissance de certaines zones d'un document. L'adaptation du système YOLO à notre cas de figure nécessiterait donc une recherche approfondie du sujet. Si l'adaptation est possible, le résultat serait donc un système permettant de détecter automatiquement les zones du document à récupérer et de corréler chaque zone détectée avec le champ correspondant au sein de la base de données.

Une fois la détection et l'attribution des zones effectuée par le système, l'utilisateur doit avoir la main afin de pouvoir vérifier qu'elles sont bien placées et si ce n'est pas le cas, les déplacer manuellement. Une correction effectuée par l'utilisateur aurait pour effet de venir en complément des données d'entraînement du système et ainsi lui permettre de s'adapter aux nouveaux documents.

Lecture des informations et importation en base

Une fois les zones détectées et attribuées aux champs de la base de données correspondants, l'étape restante est la lecture des informations, puis leur importation au sein de la base. La lecture peut s'effectuer grâce à une méthode de reconnaissance optique de caractères. Comme précisé auparavant, ces méthodes existent depuis un long moment et peuvent être mises en place en une simple ligne de code. L'importation en base des informations ainsi extraites peut ensuite être effectuée.

Conclusion

Si l'on pouvait mettre en place un tel système au sein de SIGMA, le travail de la comptabilité serait facilité. Le temps actuellement passé à la saisie des données, ainsi qu'à la vérification des concordances entre les commandes et les factures serait énormément réduit.

Bien sûr, il faut garder à l'esprit qu'il a fallu une équipe de plus de 15 développeurs et experts en intelligence artificielle à la compagnie ROSSUM pour effectuer les recherches, puis développer un outil comparable.

2.4.2 Résumés de textes

La synthèse de textes est un domaine qui se développe et qui est de plus en plus utilisé. On peut noter, par exemple, l'apparition d'outils gratuits en ligne permettant de synthétiser des textes, par exemple « resoomer », ou « autosummarizer ». Il existe aussi des applications mobiles permettant de résumer des articles d'actualité, avec « inshorts ». Synthétiser des textes automatiquement est utile et peut faire gagner énormément de temps à une personne et même à des compagnies dans leur globalité si les outils de synthèse utilisés sont performants et produisent des résumés entraînant le moins de perte d'information possible.

Pour tirer parti des méthodes de synthèses de textes d'un point de vue global sur une compagnie, on peut imaginer la mise en place d'un outil de synthèse accessible à tout moment au sein de l'ERP de l'entreprise.

Il y a trois types de synthèse possible :

- La synthèse par extraction extrait les phrases jugées les plus importantes du texte et les concatène pour produire un résumé. Cette méthode est très utilisée dans les systèmes réels.
- La synthèse par abstraction vise à résumer un texte en générant de nouvelles phrases à partir des informations importantes comprises dans le texte initial. Cette méthode permet de générer des résumés intelligents qui reprennent les informations utiles du texte dans sa globalité. Elle est peu mise en place dans les systèmes réels, de par sa complexité.
- La synthèse par compression de phrases consiste en la suppression des mots jugés superflus au sein des phrases. Elle peut aussi éliminer des phrases complètes si celles-ci ne sont pas jugées utiles.

De nombreux outils utilisant la synthèse par extraction, parfois liée à la compression de phrases, existent sur internet. L'intérêt d'ajouter un outil réalisant la même tâche au sein d'un ERP a donc très peu d'intérêt. Cependant, la synthèse par abstraction permet des résumés de bien meilleure qualité et il serait donc utile de mettre un tel outil à disposition des employés.

Lorsqu'il s'agit de traiter des problèmes concernant le traitement automatique du langage naturel, les modèles d'apprentissage profond généralement utilisés sont les RNN (« Recurrent Neural Network », « réseaux de neurones récurrents »). Les RNN sont un type de réseaux de neurones qui possède un cycle dans sa structure neuronale et qui sont capables de gérer des données d'entrées de taille variables et notamment des séquences. Dans le cadre du traitement du langage naturel, les phrases correspondent à des suites de caractères, qui peuvent être considérées par un RNN comme des séquences.

Les RNN seuls ont beaucoup été utilisés dans la recherche sur le traitement du langage, mais des employés de GOOGLE ont développé un nouveau modèle en 2014, le « sequence to sequence model » [26], modèle de séquence à séquence. Ce modèle est composé de deux parties qui se complètent : la première est un encodeur, la seconde est un décodeur. Il s'agit de deux RNN différents qui sont combinés pour n'en former qu'un seul.

Le principe est le suivant : le rôle de l'encodeur est de comprendre la séquence donnée en entrée et de créer une représentation abstraite de celle-ci. C'est comparable à un être humain qui comprend une phrase, celle-ci est représentée au sein du cerveau de manière abstraite, en tant que pensée, ce qui permet à ce dernier de la traiter plus facilement.

Cette représentation abstraite est ensuite donnée en entrée au décodeur, celui-ci va créer une nouvelle séquence correspondante aux données d'entrées. Le fonctionnement est schématisé ci-dessous.

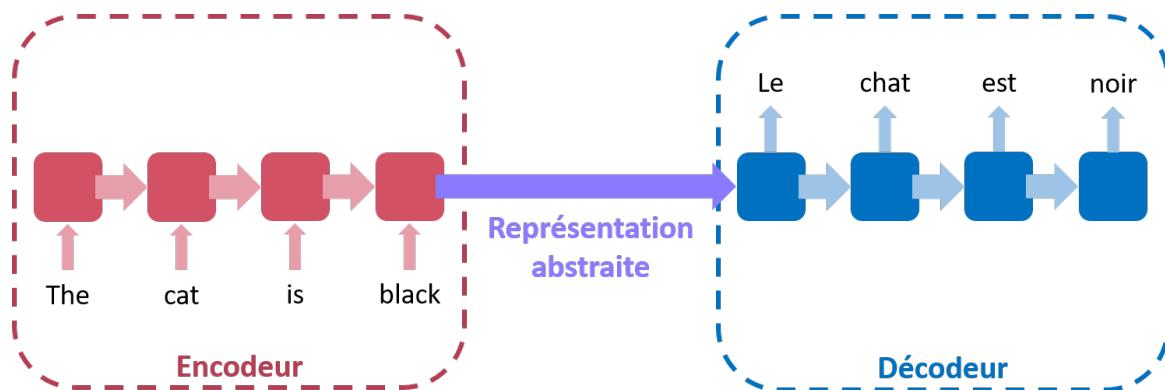


FIGURE 2.16 – Fonctionnement schématique d'un modèle de séquence à séquence

Ce type de modèle a permis d'accélérer grandement la recherche dans le traitement du langage naturel et peut notamment être utilisé pour la synthèse de texte. Correctement entraîné, un modèle de ce type est capable de comprendre les informations d'un texte qui ont une grande importance afin de les extraire grâce à l'encodeur. Le décodeur est ensuite capable d'utiliser ces données importantes pour recréer des phrases dans un langage correct et compréhensible. Le même modèle peut aussi être entraîné pour de la traduction automatique, par exemple.

Le principal problème que l'on pourrait rencontrer en développant un tel modèle est que son entraînement demande une grosse puissance de calcul. Il faut donc une excellente machine pour effectuer l'entraînement de ce modèle. En revanche, son utilisation demande peu de ressources, les machines des utilisateurs n'ont donc pas besoin d'être très performantes.

2.4.3 Gestion des stocks intelligente

La gestion des stocks au sein d'une entreprise doit permettre de ne jamais manquer de quoi que ce soit, tout en faisant attention à ne pas trop acheter dans le souci d'économiser de l'argent. Il faut constamment anticiper les besoins afin d'acheter au bon moment, ce qui n'est pas chose facile. Dans cette section nous allons voir comment l'intelligence artificielle peut aider à simplifier la gestion des stocks des entreprises.

Dans de grandes compagnies qui doivent gérer des stocks immenses, il y a de plus en plus d'automatisation. Par exemple, AMAZON a automatisé, grâce à l'intelligence artificielle et des robots, la plupart des fonctions d'enlèvement/ramassage, d'emballage et de stockage.



FIGURE 2.17 – Les robots Kiva d'AMAZON

Des compagnies ont même poussé ce concept encore plus loin, créant des « entrepôts sans lumière ». Comme SIEMENS qui possède des entrepôts capables de fonctionner dans le noir, de manière autonome, sans nécessiter le moindre être humain, et ce, pendant des semaines. Ce genre d'entrepôts peut même permettre d'éliminer les besoins de climatisation ou de chauffage.

Dans des compagnies de taille moyenne, il est possible de mettre en place des systèmes capables d'anticiper les demandes de stocks à venir, et ainsi, pouvoir anticiper les achats nécessaires.

Pour réaliser de telles prédictions, les modèles les plus utilisés sont généralement des LSTM. Un LSTM est un « long short-term memory », c'est un réseau de neurones récurrents à mémoire court-terme et long terme. C'est une version plus avancée d'un RNN qui convient mieux que ces derniers pour réaliser des prédictions. Pour réaliser des prédictions solides, un tel modèle nécessite une quantité de données considérable, dans l'objectif de trouver des liens de cause à effet au sein de celles-ci. Utiliser un tel système pour gérer les stocks au sein de DISA serait peu utile, car les données récoltées ne sont sûrement pas suffisantes.

2.4.4 Ordonnancement de phases de fabrication

Lors de mon apprentissage au sein de DISA, il m'a été confié la tâche de développer un module sur SIGMA permettant de planifier toutes les tâches de production. La solution que j'ai développée n'est pas automatique, elle nécessite un travail manuel de placement de chaque phase par les responsables de production, dans cette section nous allons tenter de trouver une solution d'intelligence artificielle qui aurait pu me permettre d'automatiser ce processus.

Le problème se présente ainsi : on a une liste d'ordres de fabrication (OF) à réaliser. Chacun possède une date de mise à disposition. Le but est évidemment de finir la production complète de l'OF avant celle-ci. Chaque OF est composé d'une liste de phases de fabrication qui doivent être réalisées sur des machines qui leur sont dédiées. Il faut donc organiser sur un planning toutes les phases de fabrication sur leur machine, le tout en prenant en compte un ordre de priorité sur les OF dont la date de mise à disposition est proche.

Pour simplifier, on peut poser une liste de contraintes que les solutions devront respecter pour être valables.

- Les phases doivent être ordonnancées sur les machines qui permettent leur fabrication.
- Les phases ne peuvent pas se chevaucher sur une même machine.
- Les phases d'un même ordre de fabrication doivent être réalisées dans l'ordre.
- Pour commencer une phase, la précédente doit être terminée.
- Toutes les phases doivent être incluses au sein de l'ordonnancement.

Après de longues recherches, la solution relevant de l'intelligence artificielle semblant de loin la mieux adaptée à un problème d'ordonnancement est l'utilisation d'un algorithme génétique.

Les algorithmes génétiques permettent de proposer des solutions à un problème en utilisant une simulation du concept de la sélection naturelle (voir la section 2.2.2 pour plus de détails). Comme il n'y a pas qu'une seule solution possible à un problème d'ordonnancement, l'utilisation de cette méthode pourrait être très efficace.

L'idée serait donc de générer une population de manière pseudo-aléatoire, en s'assurant que chaque solution de cette population prenne en compte les contraintes définies précédemment.

Les solutions de cette population ayant le plus de chances de se reproduire seraient celles avec un ordonnancement des phases laissant le moins de temps libre possible aux machines. Ainsi, en laissant tourner l'algorithme génétique et en s'assurant que toutes les solutions générées respectent les contraintes définies, on obtiendrait, à terme, un ensemble de solutions valables. Ces solutions peuvent être proposées au responsable de la production afin qu'il en sélectionne une qui lui convient.

Le planning de fabrication pourrait alors être automatiquement mis à jour avec l'ordonnancement de phases choisi. Ce fonctionnement pourrait bien entendu être généralisé pour un usage en dehors du contexte de DISA.

L'ordonnancement de tâches par l'intelligence artificielle est un domaine pour le moment très peu exploré. L'ordonnancement par des réseaux de neurones pourrait être un domaine de recherche intéressant, puisqu'aucune recherche ne semble avoir été menée sur ce sujet pour le moment [27].

2.4.5 Assistant intelligent permettant d'aider les utilisateurs au quotidien

Afin d'aider les utilisateurs dans leur utilisation d'un ERP on pourrait imaginer un système d'assistant qui serait présent pour faciliter les saisies d'informations et jouer un rôle d'anticipation tel que le ferait un assistant comme GOOGLE HOME ou ALEXA d'AMAZON. L'idée serait de pouvoir « communiquer » à l'oral ou à l'écrit avec l'ERP. On pourrait ainsi demander des informations à l'assistant telles que l'état des stocks d'un article, le statut actuel de différentes phases de fabrication, etc.

Un tel outil permettrait de faciliter les interactions entre l'utilisateur et l'ERP. Actuellement, lorsque l'on souhaite accéder à des informations sur n'importe quelle application il est nécessaire de naviguer soi-même jusqu'aux informations voulues. L'intérêt de l'assistant serait justement de naviguer automatiquement jusqu'aux informations, à partir d'une demande verbale. L'utilisation d'un tel assistant mènerait donc à un gain de temps certain au quotidien.

La société AVAAMO, spécialisée dans le développement d'assistants intelligents, a présenté un assistant permettant de faciliter l'accès à l'ERP SAP [28].

On peut imaginer que dans un futur proche, ce genre d'assistants soient intégrés par défaut au sein de beaucoup d'ERP, au même titre que les assistants intelligents sur les smartphones.

Conclusion

J'ai eu la chance d'intégrer la compagnie DISA au départ du projet SIGMA, et ainsi, de pouvoir travailler sur toutes les étapes de son développement. Allant de la conception, jusqu'à assister à sa mise en production et à son utilisation par les employés, désormais quotidienne. Mon travail sur ce projet fut très varié, ce qui a rendu mon apprentissage fortement intéressant. Au début de mon apprentissage, M. PALIER s'occupait de toute la partie récolte et analyse des besoins, tandis que je me concentrerais sur le développement et la prise en main des outils de développement. Puis, avec le temps, la confiance de M. PALIER à mon égard s'est instaurée et il me laissa de plus en plus d'autonomie dans mon travail, en me laissant m'occuper de récolter et analyser les besoins des employés. Cette autonomie m'a aidée à améliorer mes compétences relationnelles, ma confiance personnelle, ainsi que mes capacités d'adaptation à des situations nouvelles.

Mon travail m'a permis de découvrir le fonctionnement d'une PME industrielle et de nombreux métiers au sein de celle-ci. J'ai énormément appris en analysant le travail des employés afin de pouvoir concevoir des outils qui leur soient adaptés.

Nous avons vu dans ce mémoire un certain nombre de concepts connus de l'intelligence artificielle pouvant s'appliquer aux ERP. Nous avons notamment abordé des méthodes de classification et de partitionnement automatique qui peuvent être utilisées pour classer automatiquement des documents que l'on importe au sein d'une GED. L'avantage d'utiliser une méthode de classification intelligente au sein d'une GED serait de grandement simplifier les imports de fichiers. Différentes méthodes ont été présentées et analysées afin de déterminer les avantages et inconvénients de chacune d'entre elles. Nous avons évoqué des outils permettant de détecter des informations importantes au sein d'un document afin de pouvoir les extraire et les enregistrer directement en base de données, sans nécessiter de saisie manuelle par un utilisateur. Nous avons vu plusieurs méthodes de synthèse de texte et décrit le fonctionnement de la synthèse par abstraction. Pour finir, nous avons brièvement parlé d'applications possibles de l'intelligence artificielle concernant la gestion des stocks ainsi que les assistants intelligents.

Les domaines d'application de l'intelligence artificielle sont vastes et nous n'en avons abordé qu'une infime partie avec cette problématique concernant les ERP.

Je vais continuer mes études dans le domaine de l'intelligence artificielle, plus précisément en Master 2 en intelligence artificielle à l'université CLAUDE BERNARD de Lyon, la problématique du sujet de recherche m'a donc particulièrement intéressée. La rédaction de ce mémoire m'a aussi permis d'élargir mes connaissances de l'historique et de l'état de l'art de ce domaine, ce qui me sera fortement bénéfique.

Table des figures

1.1	Camion en carton, PLV 3D	7
1.2	Cycle de développement	15
1.3	Configurateur de devis avec les tableaux numérotés	18
1.4	Exemple d'une petite partie d'un tableau de production	21
1.5	Aperçu du planning de fabrication intégré à SIGMA	22
1.6	Aperçu de la fenêtre de tri du planning de fabrication avec l'avancement des phases de préparation	23
1.7	Aperçu de la fenêtre de la SFAO, avec une phase d'impression numérique en cours	25
1.8	Aperçu de la fenêtre permettant de gérer ses absences	26
1.9	Exemple de fichier généré à partir de données fictives	27
1.10	Aperçu du tableau du projet SIGMA sous TRELLO	28
2.1	Quatre approches de l'intelligence artificielle	32
2.2	GOSPER glider gun	38
2.3	Formation de deux bancs de poissons organisés par « espèces »	39
2.4	Châssis de drone léger et résistant, généré et proposé par Generative Design	40
2.5	Vue de la base construite par AlphaStar pendant une des parties	42
2.6	Fonctionnement d'AlphaFold pour la génération d'un modèle 3D d'une protéine à partir d'une séquence protéique	43
2.7	Visualisation de la détection d'objet du pilote automatique de TESLA	43
2.8	Transformation d'un dessin simpliste en paysage en utilisant GauGAN	44
2.9	Les services couverts par la plupart des ERP	45
2.10	Distribution des principaux facteurs d'adoption d'un ERP dans une PME	46
2.11	Problème de classification en deux dimensions	49
2.12	Détermination de la frontière maximisant la marge par le SVM	49
2.13	Partitionnement de données en appliquant la méthode des k -moyennes	51
2.14	Visualisation d'un partitionnement effectué en utilisant la loi de probabilité normale	52
2.15	Analyse de la structure d'une facture avec correction manuelle possible, avec l'outil ELIS	54
2.16	Fonctionnement schématique d'un modèle de séquence à séquence	57
2.17	Les robots Kiva d'AMAZON	58

Bibliographie

- [1] Virginie MATHIVET. *L'Intelligence Artificielle pour les développeurs*. ENI, 2ème édition, 2017.
- [2] Stuart J. RUSSELL and Peter NORVIG. *Artificial Intelligence : A Modern Approach*. Prentice Hall Press, Upper Saddle River, NJ, USA, 3rd edition, 2009.
- [3] ARISTOTE. Test de turing : jeu d'imitation ou test d'intelligence ?
- [4] A. M. Turing. Computers and thought. chapter Computing Machinery and Intelligence, pages 11–35. MIT Press, Cambridge, MA, USA, 1995.
- [5] Jean-Michel BERTRAND. Test de turing : jeu d'imitation ou test d'intelligence ? https://www.persee.fr/doc/quad_0987-1381_1987_num_1_1_2095, 1987. page 35.
- [6] définition de l'intelligence artificielle de l'encyclopédie larousse. https://www.larousse.fr/encyclopedie/divers/intelligence_artificielle/187257.
- [7] Warren MCCULLOCH and Walter PITTSANDREAS. A logical calculus of the ideas immanent in nervous activity. <http://www.cse.chalmers.se/~coquand/AUTOMATA/mcp.pdf>, 1943.
- [8] Donald O. HEBB. *The organization of behavior : A neuropsychological theory*. Wiley, New York, 1949.
- [9] Arthur SAMUEL. Some studies in machine learning using the game of checkers. *IBM J. Res. Dev.*, 3(3) :210–229, July 1959.
- [10] Allen NEWELL, John C. SHAW, and Herbert A. SIMON. Report on a general problem-solving program. In *Proceedings of the International Conference on Information Processing*, pages 256–264, 1959.
- [11] Arthur BRYSON and Yu-Chi Ho. Applied optimal control : optimization, estimation, and control. <https://catalogue.nla.gov.au/Record/1773065>, 1969.
- [12] Marvin L. MINSKY and Seymour A. PAPERT. *Perceptrons : Expanded Edition*. MIT Press, Cambridge, MA, USA, 1988.
- [13] Martin GARDNER. Mathematical games – the fantastic combinations of john conway's new solitaire game "life". https://web.archive.org/web/20090603015231/http://ddi.cs.uni-potsdam.de/HyFISCH/Produzieren/lis_projekt/proj_gameline/ConwayScientificAmerican.htm, 1970.
- [14] Rafael Rodríguez PUENTE Yadian Guillermo Pérez BETANCOURT and Tulimevava Kaunapawa MUFETI. Cellular automata and its applications in modeling and simulating the evolution of diseases. https://www.researchgate.net/publication/282249343_Cellular_Automata_And_Its_Applications_In_Modeling_And_Simulating_The_Evolution_Of_Diseases, 2015.
- [15] Joana Gouveia FREIRE and Carlos Castro DACAMARA. Using cellular automata to simulate wildfire propagation and to assist in fire management. <https://www.nat-hazards-earth-syst-sci.net/19/169/2019/>, 2019.
- [16] Thomas RANVIER. Fish shoal. https://github.com/ThomasRanvier/fish_shoal, 2018.
- [17] Christian WOLF Marc MOURETA, Christine SOLNONA. Classification of images based on hidden markov models. <https://perso.liris.cnrs.fr/cwolf/papers/cbmi2009-marc.pdf>, 2007.
- [18] Joakim Jitén SÖDERBERG. Multidimensional hidden markov model applied to image and video analysis. <https://pastel.archives-ouvertes.fr/pastel-00002454/document>, 2007.

- [19] Mark GALES and Steve YOUNG. The application of hidden markov models in speech recognition. https://mi.eng.cam.ac.uk/~mjfg/mjfg_NOW.pdf, 2007.
- [20] Carolin KOSIOL. Markov models for protein sequence evolution. <https://www.ebi.ac.uk/sites/ebi.ac.uk/files/shared/documents/phdtheses/carolinkosiolthesis.pdf>, 2006.
- [21] Max JADERBERG, Wojciech M. CZARNECKI, Iain DUNNING, Luke MARRIS, Guy LEVER, Antonio GARCÍA CASTAÑEDA, Charles BEATTIE, Neil C. RABINOWITZ, Ari S. MORCOS, Avraham RUDERMAN, Nicolas SONNERAT, Tim GREEN, Louise DEASON, Joel Z. LEIBO, David SILVER, Demis HASSABIS, Koray KAVUKCUOGLU, and Thore GRAEPEL. Human-level performance in first-person multiplayer games with population-based deep reinforcement learning. *CoRR*, abs/1807.01281, 2018.
- [22] Oriol Vinyals, Igor Babuschkin, Junyoung Chung, Michael Mathieu, Max Jaderberg, Wojciech M. Czarnecki, Andrew Dudzik, Aja Huang, Petko Georgiev, Richard Powell, Timo Ewalds, Dan Horgan, Manuel Kroiss, Ivo Danihelka, John Agapiou, Junhyuk Oh, Valentin Dalibard, David Choi, Laurent Sifre, Yury Sulsky, Sasha Vezhnevets, James Molloy, Trevor Cai, David Budden, Tom Paine, Caglar Gulcehre, Ziyu Wang, Tobias Pfaff, Toby Pohlen, Yuhuai Wu, Dani Yogatama, Julia Cohen, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy Lillicrap, Chris Apps, Koray Kavukcuoglu, Demis Hassabis, and David Silver. AlphaStar : Mastering the Real-Time Strategy Game StarCraft II. <https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/>, 2019.
- [23] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.
- [24] Cindy JUTRAS. Erp in the sme : Fueling growth and profits. <https://www.iiia.nl/SiteFiles/enterprise-resource-planning.pdf>, 2010.
- [25] Joseph REDMON, Santosh Kumar DIVVALA, Ross B. GIRSHICK, and Ali FARHADI. You only look once : Unified, real-time object detection. *CoRR*, abs/1506.02640, 2015.
- [26] Ilya SUTSKEVER, Oriol VINYALS, and Quoc V. LE. Sequence to Sequence Learning with Neural Networks. <https://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf>, 2014.
- [27] Peter BEX. Implementing a Process Scheduler Using Neural Network Technology. <https://theses.ubn.ru.nl/bitstream/handle/123456789/168/Bex,%20P.MScThesis.pdf?sequence=1>.
- [28] Avaamo. <https://avaamo.ai>.
- [29] Andreas KAPLAN and Michael HAENLEIN. Siri, siri, in my hand : Who's the fairest in the land ? on the interpretations, illustrations, and implications of artificial intelligence. <https://www.sciencedirect.com/science/article/pii/S0007681318301393#bib10005>, 2018. Business Horizons.