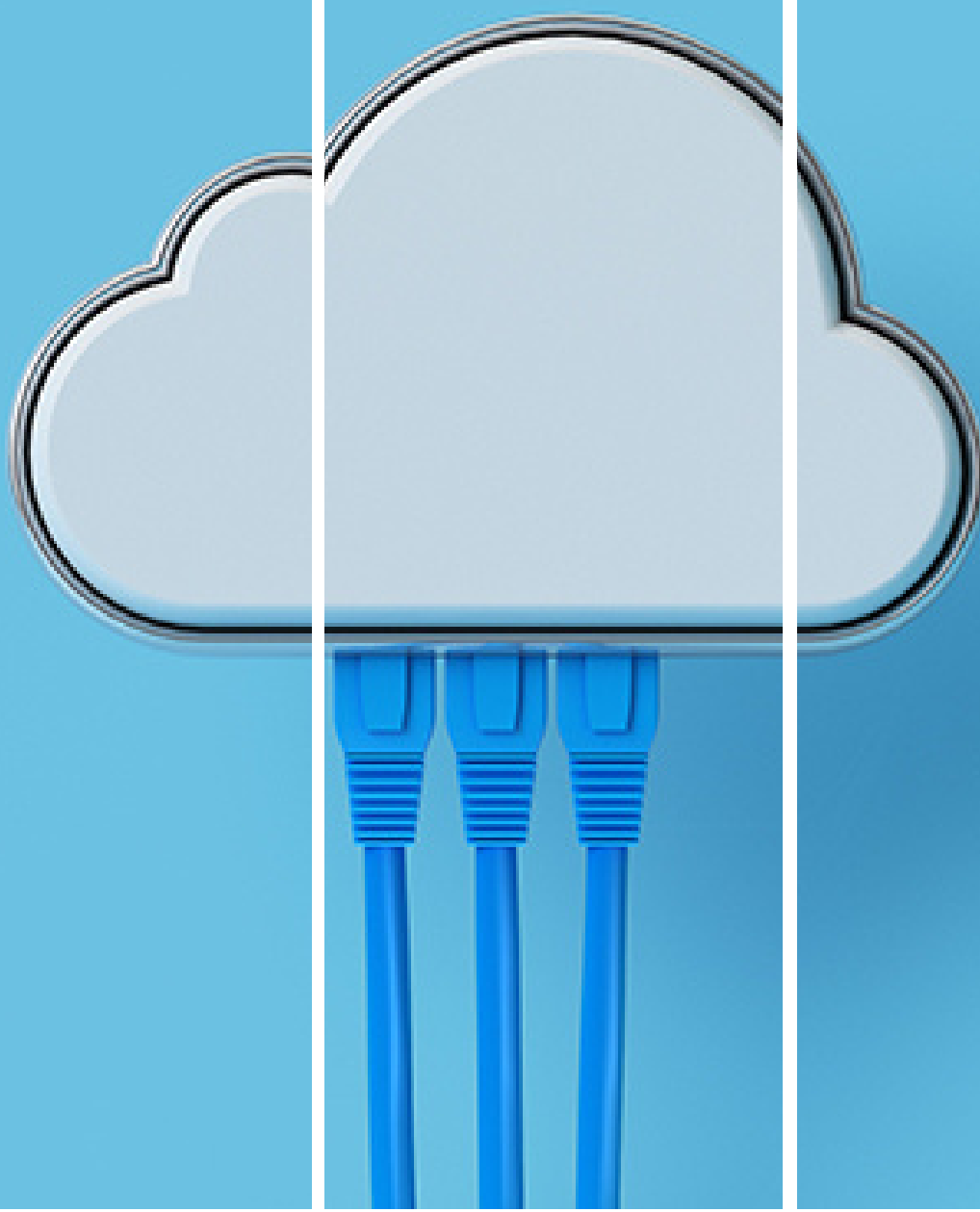


DÉSENGAGEMENT DU *MAINFRAME*

ENJEUX ET MISE EN ŒUVRE AU
CŒUR D'UN SYSTÈME
D'INFORMATION



SOMMAIRE

RÉSUMÉ	3
INTRODUCTION	4
LA STRATÉGIE DE MIGRATION	5
Pourquoi migrer ?	
Les différentes approches de migration	
NOTRE APPROCHE	7
Pourquoi Infotel préconise le <i>Retire</i> et le <i>Refactor</i> ?	
Le <i>Retire</i>	
Les avantages	
Le <i>Refactor</i> ou <i>Rebuild</i>	
Les avantages	
La migration de Cobol vers le langage Java	
NOTRE MÉTHODOLOGIE	10
L'agilité, notre force	
Les étapes du projet de migration	
Le cadrage et la planification	
Le <i>reverse engineering</i> & la conception de la cible	
Le développement logiciel et les tests unitaires	
L'analyse et le suivi des risques	
La phase d' <i>acceptance</i>	
La migration des <i>process</i> et le démarrage	
L'EXPÉRIENCE CLIENT	17
CONCLUSION	25

RÉSUMÉ

Ce livre blanc a pour objectif d'accompagner les managers de SI dans leur choix d'une stratégie de désengagement du *mainframe*.

Pour atteindre cet objectif, **Infotel** propose une approche singulière qui associe deux stratégies complémentaires.

Dans un premier temps, la technique du *Retire* permet d'analyser l'applicatif d'un SI afin d'identifier les composants « obsolètes », réduisant ainsi l'effort de migration ultérieur.

Puis, la stratégie de *Refactor* assure une réécriture de qualité de l'ancien code (principalement Cobol) vers un langage plus moderne et plus adapté aux nouveaux challenges de l'applicatif (en général Java).

Ce livre blanc explique la mise en œuvre de cette approche, en mettant l'accent sur l'importance de notre méthodologie de conduite pour ce type de projet.

Enfin, dans l'optique de "démystifier" un processus de migration qui peut paraître extrêmement complexe, ce livre blanc est illustré par l'une de nos expériences client, détaillant chaque étape de cette transformation du SI.

Par son expertise technique et fonctionnelle, **Infotel** vous accompagne tout au long de votre projet de désengagement du *mainframe* !

INTRODUCTION

Au sein des entreprises, la recherche de la performance du Système d'Information est une préoccupation permanente.

La fiabilité reconnue d'un système *mainframe* d'IBM ne suffit parfois plus à garantir son maintien au sein du SI au regard du bilan des coûts de licence, de la CPU utilisée et de la dette technique induite logiquement par des années de maintenance corrective et évolutive.

Force est de constater que la pression concurrentielle sur les coûts et l'innovation, le besoin d'interopérabilité toujours croissant, les évolutions technologiques mises à disposition de plus en plus fréquemment en mode *open source*, ainsi que l'appétence naturelle des utilisateurs pour un traitement et une restitution des données en temps réel poussent les responsables de SI, encore détenteurs d'une solution *mainframe*, à envisager une modernisation de leur parc.

Se pose alors la question du choix de la migration technique et des impacts du désengagement de la partie *mainframe* au sein du SI.

À propos d'Infotel

Chiffre d'affaires : 263,4 M€
en 2021

Taux de croissance : + 12 %

730 clients nous font
confiance
2 600 collaborateurs

Le groupe Infotel est coté sur
Euronext Paris –
Compartiment B

Nos valeurs :

Excellence
Engagement
Qualité

Le Groupe accompagne ses clients dans la gestion et l'évolution de leur patrimoine informatique. Nous apportons également notre expertise en ce qui concerne le conseil et l'innovation auprès des métiers et directions IT.

Spécialiste de la transformation digitale end-to-end », Infotel co-construit les plans de transformation, co-conçoit l'évolution des offres, services et parcours clients et co-opère les changements de processus internes ainsi que les plans d'action opérationnels en mode agile.

Nous développons ces savoir-faire dans divers secteurs d'activité.

LA STRATÉGIE DE MIGRATION

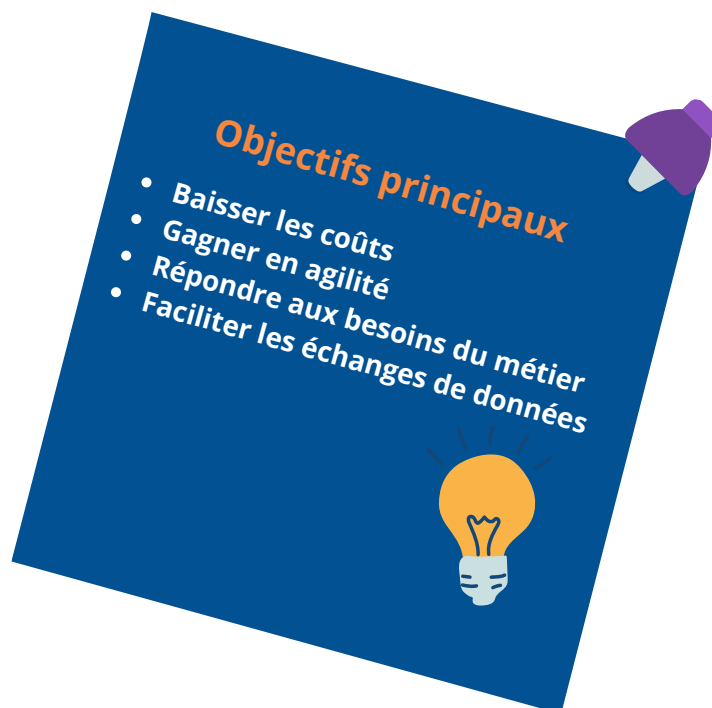
Pourquoi migrer ?

L'intérêt principal de la migration consiste en une optimisation de la rentabilité, via la baisse des coûts liés aux licences, d'une solution basée sur une plateforme propriétaire IBM *mainframe*.

Par ailleurs, la migration vers une solution non-propriétaire (*open source*) apparaît comme une occasion de pouvoir délivrer de manière plus agile de nouvelles fonctionnalités attendues par les métiers, en bénéficiant au passage des nouveautés proposées par un socle technologique plus récent.

Enfin, la migration va garantir et faciliter les échanges de données plus nombreux entre le parc applicatif jusqu'alors hébergé sur un système *mainframe* et le reste du SI.

Le fait notamment de pouvoir exposer ou appeler des services qui répondent à des standards techniques communs sur l'ensemble du SI du client constitue une attente forte.



LA STRATÉGIE DE MIGRATION

Les différentes approches de migration

Se désengager du *mainframe*, signifie en tout état de cause de migrer l'ensemble applicatif, comprenant souvent une ou plusieurs bases de données, sur une nouvelle infrastructure impliquant une plus ou moins grande modification du code applicatif.

En 2011, Gartner regroupait les approches pour analyser une migration vers le *cloud* en cinq catégories dans un premier temps, avant une évolution vers **six recommandations pour finir : les 6 R**. Cette approche peut être parfaitement appliquée dans le cadre d'un désengagement *mainframe*, d'autant plus, que dans la plupart des cas, **Infotel** recommande dans le même temps de migrer vers une infrastructure *cloud*.

Infotel applique ainsi cette méthodologie dans ses projets de désengagement *mainframe*.

Rehost

- Connue aussi sous le nom de "*lift and shift*", cette stratégie consiste à migrer tout le périmètre applicatif sans aucun changement de code ni de recompilation. Dans le cadre d'une migration *mainframe*, cela implique l'utilisation d'une solution tierce.

Replatform

- Contrairement au *rehost*, cette stratégie implique des changements normalement limités dans le code et une recompilation afin de permettre à l'applicatif de fonctionner sur la nouvelle plateforme retenue.

Repurchase

- Rachat d'une nouvelle solution. Ce n'est en fait pas une migration car il s'agit tout simplement de démarrer sur un nouveau système.

Refactor

- Refonte applicative complète pouvant impliquer des changements d'architecture logicielle parfois très importants. Réduire la dette technique, revoir les fonctionnalités et améliorer l'interopérabilité sont les maîtres-mots de cette approche.

Retire

- Retrait de certaines fonctionnalités obsolètes ou inutiles afin de réduire le périmètre de migration. À noter que l'on estime à 10 % du portfolio d'une entreprise le volume d'applications pouvant être décommissionnées.

Retain

- Conservation de certaines applications telles quelles. Elles ne sont donc pas migrées. Dans le cadre d'un projet de désengagement *mainframe*, cette stratégie ne présente que très peu d'intérêt.

NOTRE APPROCHE

Pourquoi Infotel préconise le *Retire* et le *Refactor* ?

Fort de son expérience en ingénierie technico-fonctionnelle auprès de ses clients, **Infotel** est en mesure d'accompagner ses clients sur chacune des stratégies de sortie du *mainframe* présentées précédemment.

Néanmoins, il est important de souligner que certains clients qui sont encore détenteurs de solutions *mainframe* restent généralement attachés, dans leur recherche de modernisation de SI, à la maîtrise et à la sauvegarde de leur savoir-faire métier. Cette exigence les conduit à privilégier davantage **une technique de migration centrée sur l'application plutôt que sur l'infrastructure technique proprement dite**.

Dans cette perspective, **Infotel** privilégie de mener une stratégie de *Retire* qui sera ensuite complétée par une stratégie de *Refactor* afin de répondre au plus près aux nouveaux besoins du client.

Chacune de ces approches offre des avantages notables.

En effet, dans un premier temps le *Retire* va permettre d'épurer le parc applicatif. Dans un second temps, le *Refactor* offre la possibilité de réorganiser le parc applicatif voire d'implémenter de nouveaux logiciels pour une meilleure performance.

Ces deux stratégies peuvent être combinées pour un résultat optimum ou dissociées selon le besoin du client. Enfin, il convient de préciser que notre démarche de désengagement du *mainframe* s'effectue vers le langage de programmation le plus utilisé et le plus qualitatif aujourd'hui, à savoir Java.

Le Retire

L'analyse préalable des applications est un processus complexe et essentiel pour décider quelles applications retirer. Cela doit être fait dans les premières étapes de la planification afin de migrer les applications ou les services essentiels et économiser des ressources.

On estime à 10 % du portfolio informatique d'une entreprise qui peut être désactivé du fait de son inutilité.

Cette approche nécessite une implication du client pour un travail collaboratif efficient.



Les avantages

- Réduire les coûts de la migration
- Simplification du parc applicatif
- Le métier se recentre sur les applications essentielles
- Technique facilitant la migration ultérieure par *rehosting*, *replatforming* ou *rebuilding*

NOTRE APPROCHE

Le Refactor ou Rebuild

Au fil du développement d'une application, la dette technique s'accumule et altère la maintenabilité de certains programmes, ce qui menace les possibilités d'évolution et la compatibilité de l'application au sein du SI.

Pour résoudre ce problème, on peut soit écrire un code source entièrement nouveau, soit restructurer le code par petites étapes. Plus précisément, nous procédons à la réécriture du code des applications (Cobol vers Java). Dans le même temps, en cas d'utilisation de bases de données sur le *mainframe*, nous préconisons également la migration vers des bases de données du type PostgreSQL ou MongoDB en parallèle de la refonte applicative.

Nous proposons donc le « réusinage » de code pour optimiser à long terme un logiciel fonctionnel et réduire les coûts de maintenance futurs.

Bien souvent, les solutions de *Rehost* ou de *Replatform* sont mises en avant face à une stratégie de *Refactor*. Bien qu'attrayantes de prime abord de par leur aspect "clés en main" et un coût global limité en apparence, ces solutions impliquent obligatoirement, *in fine*, des changements substantiels au niveau du code qui entraînent un impact financier important et qui les rend finalement beaucoup moins concurrentielles face à une refonte applicative.

C'est pourquoi Infotel préconise en premier lieu la réécriture personnalisée du code applicatif en offrant des prestations d'ingénierie à la fois techniques et fonctionnelles.

Avantages

- Meilleure qualité du code: suppression du code dupliqué, du code mort, optimisation algorithmique.
- Plus de souplesse et de modularité (architecture de micro-services par exemple).
- Amélioration de la testabilité des programmes, en bénéficiant des outils standards du marché.
- Amélioration des performances.
- Avoir un code customisé au plus près des attentes du client *versus* solution automatisée.

Objectifs principaux

Réduire la dette technique et réactualiser les règles fonctionnelles ainsi que l'interopérabilité.



NOTRE APPROCHE

La migration du langage Cobol vers le langage Java

Pourquoi passer du Cobol au langage Java ?

Plus de 220 milliards de lignes de code dit Cobol (*Common Business Oriented Language*) sont utilisées dans le monde; on estime que 70 % des transactions commerciales mondiales sont traitées par une application *mainframe* sous langage Cobol.

Néanmoins, les architectures des applications *mainframe* n'ont pas été pensées pour les tendances actuelles : les innovations mobiles, l'IoT, le *cloud*, etc., ni même pour la transformation numérique. De plus, le *mainframe* implique des frais de maintenance mais également une compétence qui se raréfie. Cette technologie présente un manque d'extensibilité des applications et d'interopérabilité avec d'autres applications non *mainframe*.

L'expertise Infotel :

Infotel préconise de migrer les parcs applicatifs *mainframe* généralement vers Java, au vu des avantages que cela présente.

Cette tâche reste néanmoins extrêmement complexe et nécessite des compétences en ingénierie technique et fonctionnelle pointues.

Dans ce domaine, Infotel jouit d'une expertise approuvée par ses clients. Ainsi, au regard de notre expérience nous observons que les coûts et les risques qui peuvent être engendrés sont en général compensés par la rentabilité que peut apporter la migration d'un parc applicatif Cobol vers Java.

Les avantages du passage au langage Java sont les suivants :

- les frais de support sont négligeables par rapport à Cobol (même avec les politiques de licence actualisées d'Oracle) ;
- Java est l'un des langages de programmation les plus utilisés aujourd'hui (langage de prédilection des programmes informatiques des grandes entreprises) ;
- la documentation du langage, les bibliothèques logicielles et les outils de développement sont disponibles gratuitement ;
- l'interopérabilité et la portabilité de Java sont excellentes par rapport aux autres technologies ;
- Java fonctionne avec pratiquement toutes les technologies de bases de données et middleware utilisées aujourd'hui ;
- les applications Java fonctionnent sur n'importe quel type d'infrastructure, parmi celles disponibles sur le marché.
- permet aussi de migrer facilement vers d'autres langages ultérieurement, si nécessaire.

NOTRE MÉTHODOLOGIE

L'agile, notre force

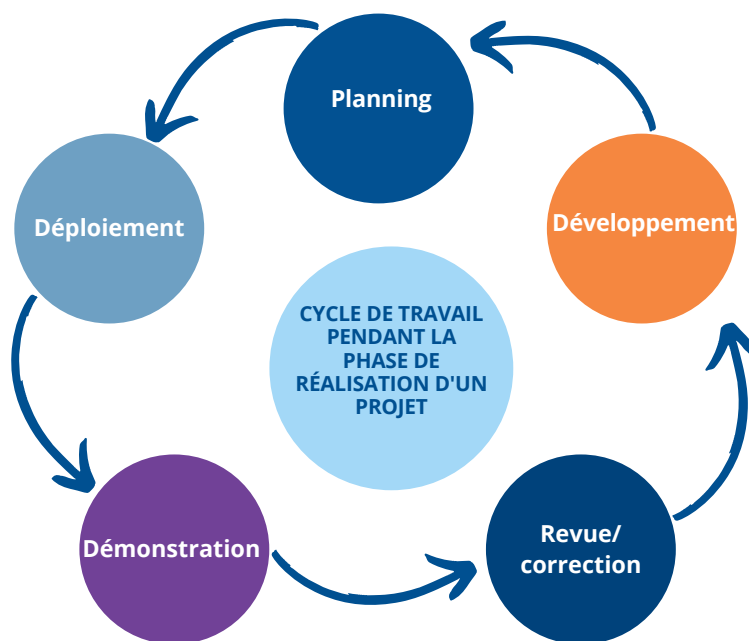
Au niveau de ce type de projet de migration, pour lequel il est difficile d'obtenir dès le départ une validation formelle de spécifications fonctionnelles et techniques détaillées pour chaque *process* à migrer, les équipes projet d'**Infotel** adoptent un mode de gestion agile afin de générer un premier produit logiciel délivrable aux clients dans un délai raisonnable en fonction d'un planning défini avec le pilote.

Les itérations (*sprints*) successives permettent d'améliorer chaque *process batch*/interface utilisateur jusqu'à la validation finale.

En fonction de l'envergure du projet de migration, **Infotel** propose de mettre en place une ou plusieurs équipes en charge du développement. Au sein de chaque équipe, des réunions quotidiennes brèves de suivi des tâches en cours sont organisées, afin d'identifier les points de blocage au plus vite et de faire monter en compétences les ressources encadrées par des « sachants ».

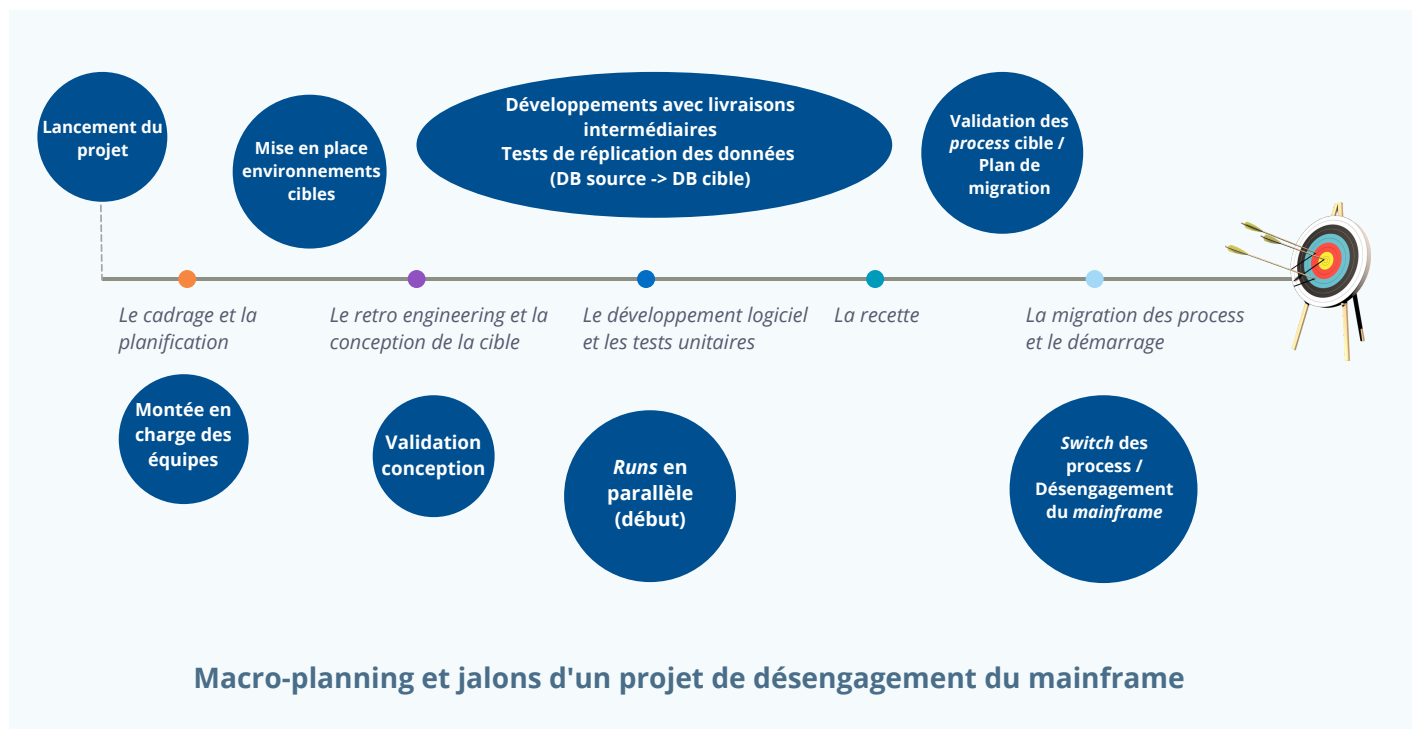
Enfin, des architectes et coordinateurs projet, rassemblés au sein d'une équipe dédiée, assurent un suivi de développement, l'objectif étant de regrouper les tâches redondantes des services communs utilisables par l'ensemble des équipes.

Infotel s'implique à la fois au niveau de la connaissance des dépendances fonctionnelles entre les différents périmètres applicatifs mais aussi au niveau des possibilités techniques offertes par le nouveau socle logiciel.



NOTRE MÉTHODOLOGIE

Les étapes majeures d'un projet de migration



Bonne pratique :

Il est primordial de travailler en collaboration étroite avec le client à chacune des étapes clés du projet de façon agile afin d'obtenir une migration optimum conforme aux besoins du client.

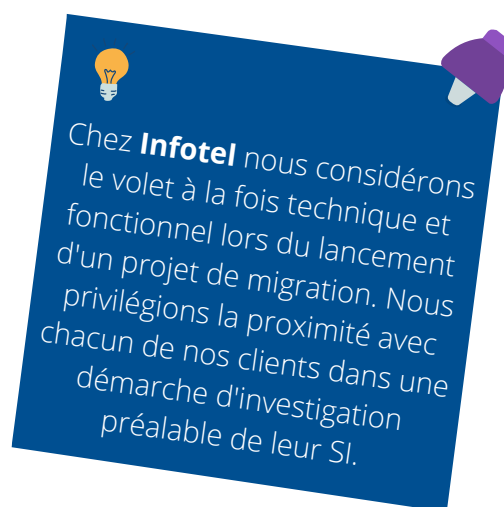
NOTRE MÉTHODOLOGIE

Les étapes du projet de migration

Le cadrage et la planification :

Dans la communication entre **Infotel** et notre client, l'étape de cadrage doit permettre de répondre dès le début aux questions suivantes :

- Quelle architecture technique va répondre de manière pertinente aux nouveaux besoins des utilisateurs du SI ?
- Quel est le périmètre applicatif et fonctionnel à migrer ? Un allotissement est-il possible ou souhaitable ?
- Quels sont les *process* devenus obsolètes (possibilité de *Retire*) ?
- Quels sont les besoins en termes de ressources, de compétences et d'organisation pour mener à bien le projet de migration ?
- Quel est le planning de migration souhaité ou imposé par le client ? Y-a-t-il des contraintes réglementaires à prendre en compte ?
- Quels sont les critères d'*acceptance* métier attendues pour ce projet de migration (non-régression, qualification et criticité des problèmes, etc.) ?
- Est-il pertinent d'effectuer une migration ISO fonctionnelle ou non ?



NOTRE MÉTHODOLOGIE

Le *reverse engineering* et la conception de la cible

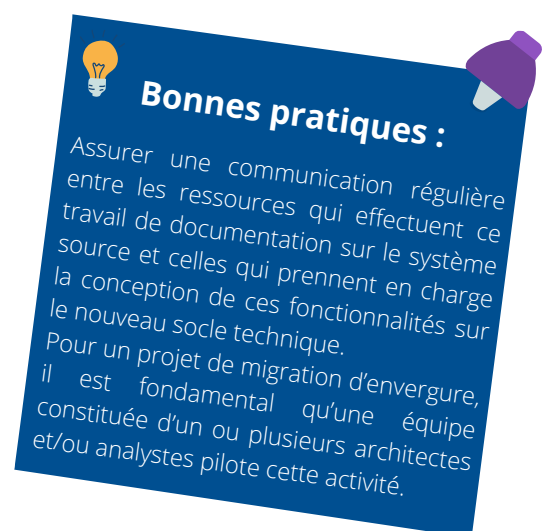
Lorsqu'un premier cadrage de la migration a été effectué et que le lancement du projet est prononcé, il est nécessaire de préciser et de **mettre en place le nouveau socle technique**, en lien avec des ressources dédiées côté client.

L'un des objectifs de cette étape est de pouvoir **valider rapidement la proposition faite lors du cadrage initial, à l'aide d'un prototype constitué**, par exemple d'un traitement *batch* simple à mettre en œuvre sur une plateforme cible de test.

L'un des autres objectifs de cette phase est de **pouvoir produire un ou plusieurs schémas d'architecture (généraux et détaillés)**.

Si cela est nécessaire, le *reverse engineering* des traitements (actifs !) côté *mainframe* a lieu également lors de cette étape.

Gage de qualité pour les développements à venir, ce travail permet également de **mutualiser plus facilement une partie de la conception**, lorsque cela s'avère possible. **Un gain important en matière de coût d'analyse et de développement ainsi que de délai de mise en œuvre sont à la clé.**



NOTRE MÉTHODOLOGIE

Le développement logiciel et les tests unitaires

Après la validation du prototype et de la conception générale par le client, chaque équipe en charge d'un périmètre applicatif bien défini lors du cadrage initial peut démarrer les tâches de développement logiciel, sous la houlette d'un chef de projet garant des engagements contractuels définis avec le client lors du lancement.

Cette étape permet à nos développeurs d'installer leurs outils de test unitaire et d'exécuter les premiers scénarios.

Quelques points de vigilance

Lorsque la migration implique un transfert des données de la base *mainframe*, cela constitue un enjeu majeur dans la réussite finale !

Dès le début du projet, ce chantier nécessite ainsi une coordination étroite entre les équipes de support Infrastructure, DBA, les architectes et consultants experts en cas d'utilisation d'outils de marché.

L'analyse et le suivi des risques

Le suivi des risques internes et externes n'est bien entendu pas spécifique aux projets de migration. Cependant, pour simplifier, plus le projet de migration embarque d'applications dans son périmètre, plus le risque est élevé de multiplier les points de blocage liés à des causes externes par rapport au périmètre initial des applications à migrer.

Parmi les risques rencontrés lors du projet de migration, on peut citer :

- montée de version ou changement d'architecture technique côté client (serveurs ou base de données) en cours de migration ;
- projets concurrentiels hors périmètre de la migration (évolutions techniques ou fonctionnelles, réglementaires) mais avec risque d'effet de bord ;
- recensement complexe et pas toujours exhaustif des besoins en interfaçage avec des applications connexes sur la nouvelle plateforme logicielle (création ou utilisation d'APIs existantes pour échanger des données de manière différente par rapport au *mainframe*) ;



Bonnes pratiques:

Limiter les changements techniques (montées de version, etc.) ou fonctionnels au strict minimum dans un périmètre suffisamment large autour du SI à migrer !
Penser à effectuer, en début et aussi en cours de projet une revue régulière de l'évolution des dépendances externes avec les équipes en charge du développement logiciel.

NOTRE MÉTHODOLOGIE

La phase d'*acceptance*

Sur un projet de migration d'envergure, avec changement de socle technique et reconstruction du code source, la validation par le client est bien entendu incontournable dans le *process*. Une partie de la validation finale du client peut être obtenue tacitement ou formellement lors de la phase itérative de développement, une fois que les critères d'acceptation sont atteints sur telle ou telle fonctionnalité.

Or, la mise au point d'un environnement de recette complet n'est pas toujours aisée, au gré des livraisons échelonnées dans le temps pour chaque périmètre applicatif du projet.

Aussi, il s'avère souvent nécessaire de formaliser la validation par la maîtrise d'ouvrage et le métier lors d'une étape dédiée qui garantit que :

- la recette des fonctionnalités est bien réalisée sur la plateforme cible de production ou, à défaut, sur un environnement stable similaire à l'environnement de production cible (volumétrie de données, paramétrage du socle technique, etc.) ;
- les versions déployées sur l'environnement de recette sont réputées testées de manière unitaire et au niveau de la non-régression notamment ;
- l'environnement dédié à la recette pourra être rafraîchi et ce à la demande des utilisateurs ;
- les traitements qui s'exécutent sur l'environnement de recette peuvent être monitorés simplement, tant au niveau de la consommation de la charge machine que des temps de traitement ;
- les résultats fonctionnels obtenus sur l'environnement de recette cible (pour les traitements *batch* par exemple) peuvent être comparés aux résultats obtenus sur l'environnement de PROD source *mainframe*.

NOTRE MÉTHODOLOGIE

La migration des *process* et le démarrage

Une fois la recette prononcée, il convient de procéder à la migration ou à l'activation des *process* sur la nouvelle plateforme technique.

Afin de synchroniser les étapes de la migration en production, **il s'agit d'établir et de valider, avec toutes les parties prenantes du projet, un plan général de migration pour le *switch* vers la nouvelle plateforme.**

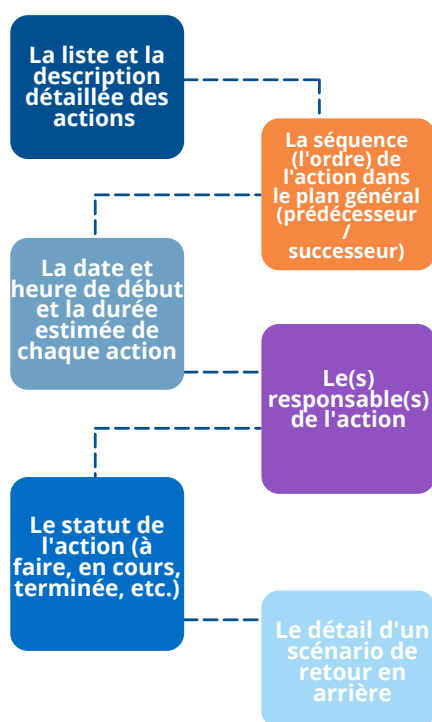
À noter que, parmi toutes les actions listées, certaines peuvent être anticipées par rapport à la migration prévue le jour J (tâches de communication, création de tickets, préparation et test de scripts, etc.).

En fonction de la taille du projet de migration, de la disponibilité des ressources et des environnements ainsi que des impératifs de délais, **une répétition du démarrage réel en production peut s'avérer nécessaire.**

Enfin, l'une des questions, lors de la migration entre l'ancienne architecture et la nouvelle, est de savoir s'il est possible de **limiter l'interruption de service pour les utilisateurs et de déterminer sa durée, le cas échéant.**

La connaissance des plages de disponibilité contractuelles des applications présentes sur *mainframe* permet alors de choisir un planning de démarrage optimal en fonction des contraintes techniques (arrêt et relance des serveurs applicatifs, arrêt de la synchronisation éventuelle des données entre la base de données *mainframe* et la base cible).

Informations contenues dans le planning de démarrage :



Bonne pratique:

La fin du projet de migration s'accompagne généralement d'un transfert de connaissances vers la ou les équipes en charge du maintien en condition opérationnelle avec un objectif d'automatisation et d'amélioration des outils de supervision.

L'EXPÉRIENCE CLIENT

Contexte et enjeux

Le DSI d'une organisation internationale de gestion documentaire souhaite se désengager du *mainframe* pour moderniser son environnement informatique et réduire les coûts directement liés à sa plateforme propriétaire (licences IBM, utilisation de la CPU, etc.).

Pour ce client, le projet de **refactoring** sur une plateforme **cloud** plus moderne et non propriétaire est l'occasion de :

- mieux maîtriser les coûts informatiques ;
- évoluer vers une plus grande pratique du DevOps (maîtrise des déploiements) ;
- de réduire la dette technique accumulée sur le mainframe ;
- de renforcer la compétence des équipes (analyse des règles métier) ;
- d'augmenter l'interopérabilité du SI avec des systèmes connexes et la prise en compte des besoins métier.

Signalons que ce client avait initié, quelques années auparavant, un projet de **replatforming** intégral sur un vaste périmètre applicatif, projet qui s'est avéré plus long et plus complexe que prévu !

Capitalisant sur ce premier retour d'expérience, le client a décidé de diversifier sa stratégie de migration vers le **cloud** en confiant à **Infotel**, fin 2020, la réalisation d'un projet de désengagement du mainframe sur une autre partie de son parc, en adoptant une stratégie de **refactoring** du code applicatif cette fois, associée à une migration de la base de données.

L'EXPÉRIENCE CLIENT

Architecture cible

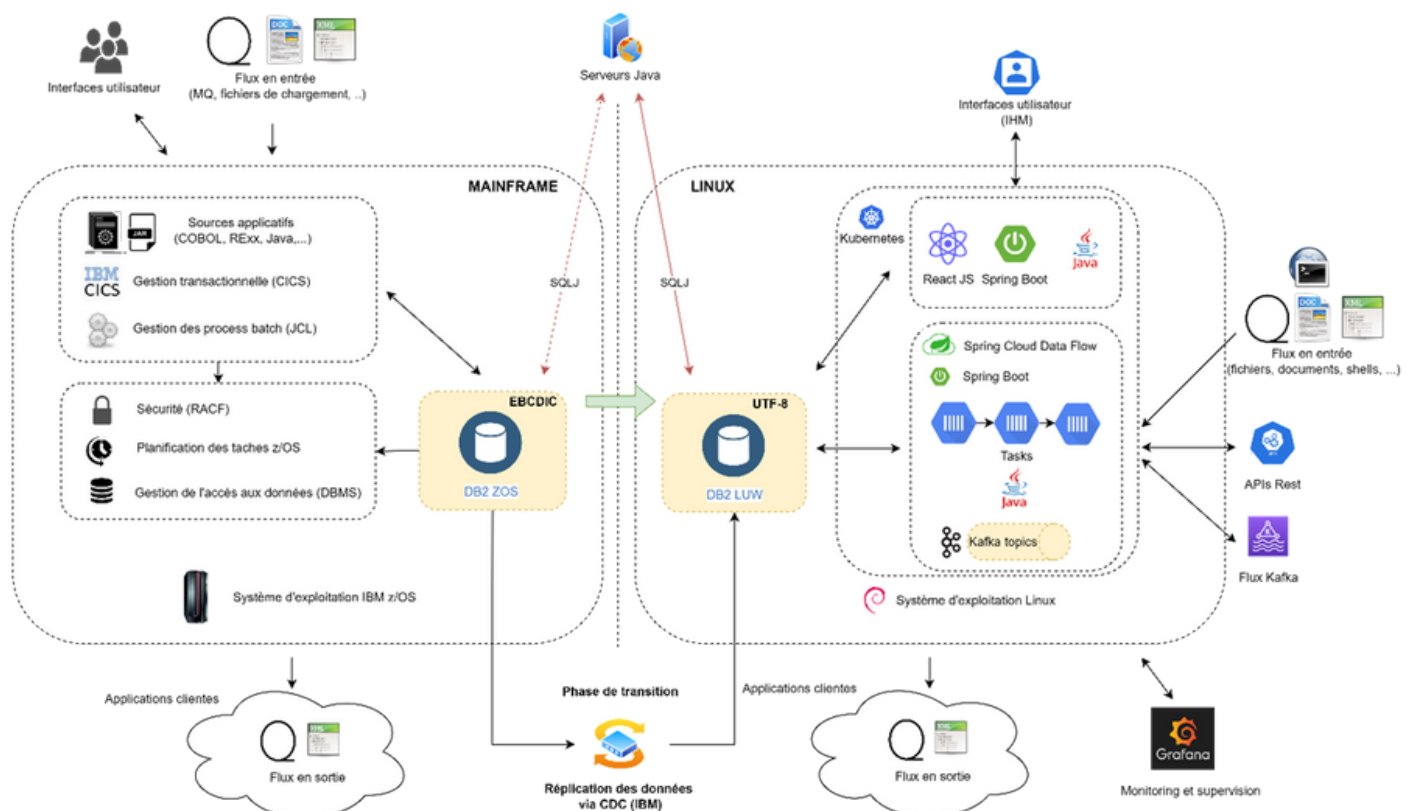
Sur le plan de l'architecture cible, le client a privilégié une architecture *cloud* native Linux, pour bénéficier de la conteneurisation Docker sur plateforme Kubernetes.

Les nouvelles interfaces utilisateurs (GUI) ont été repensées à l'aide de Java, Spring Boot et de la bibliothèque JavaScript React.

Les autres traitements *batch* utilisent principalement Java, Spring Cloud Data Flow, Spring Boot.

Les échanges de données inter-applicatifs sont assurés par des APIs Rest ou encore Apache Kafka pour la gestion de messages temps réel.

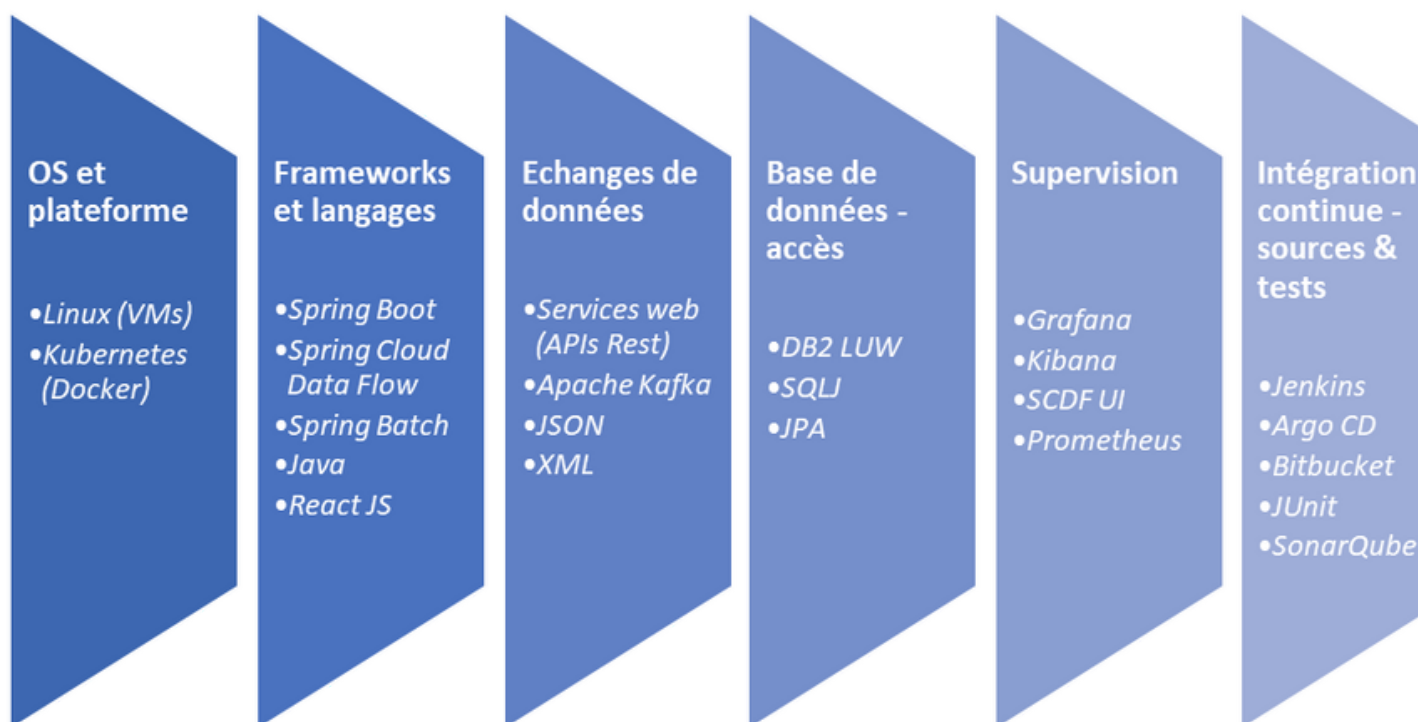
Les données sont sauvegardées dans une base DB2 LUW côté Linux et synchronisées avec les données DB2 z/OS côté *mainframe* pendant toute la phase de transition, grâce au logiciel CDC (*Change Data Capture*), réplique d'IBM.



L'EXPÉRIENCE CLIENT

Environnement technique détaillé

Ci-après, un résumé détaillant les différentes couches logicielles, *frameworks* et langages mis en oeuvre sur la plateforme cible lors de ce projet de migration :



L'EXPÉRIENCE CLIENT

Planning de la migration

Le projet du client concerne la migration des applications de gestion documentaire et de la base de données DB2 hébergées sur *mainframe* vers une plateforme *cloud* native équipée de machines virtuelles Linux.

Lors du cadrage initial, Infotel a accompagné le client dans l'analyse de la criticité des *process* à migrer afin d'identifier et de statuer sur la possibilité ou non de migrer des *process* peu (ou pas) utilisés.

Côté planning, ci-après le résumé des jalons de ce projet de migration qui a nécessité un **refactoring complet du code applicatif, en moins de douze mois** :

Macro-planning du projet de désengagement du *mainframe* (applications et base de données)



L'EXPÉRIENCE CLIENT

Zoom sur la phase d'*acceptance* et de mise en production

En phase d'*acceptance*, Infotel a développé des outils de comparaison de données spécifiques entre le *mainframe* et le système cible, afin de faciliter la validation par le client lors des exécutions menées en parallèle (*parallel runs*) sur le *mainframe* et SCDF (Spring Cloud Data Flow).

Signalons que certains *process*, activés avant la date fixée pour le switch final, ont été spécifiquement adaptés afin de permettre une récupération des données à la fois dans la base DB2 z/OS et LUW (en fonction des tables du modèle de données) et cela pour **garantir une adéquation optimale des données avec les autres applications mainframe pendant la phase de transition.**

La synchronisation des données (via CDC) entre le mainframe et Linux a, quant à elle, été arrêtée de manière définitive lors du *switch* final !

Les équipes d'**Infotel** se sont mobilisées durant tout un week-end pour superviser le *switch* final et l'activation de l'ensemble des *process* sur la plateforme Linux de production, en garantissant ainsi **une continuité de service optimale** d'un point de vue utilisateurs durant toute l'opération.

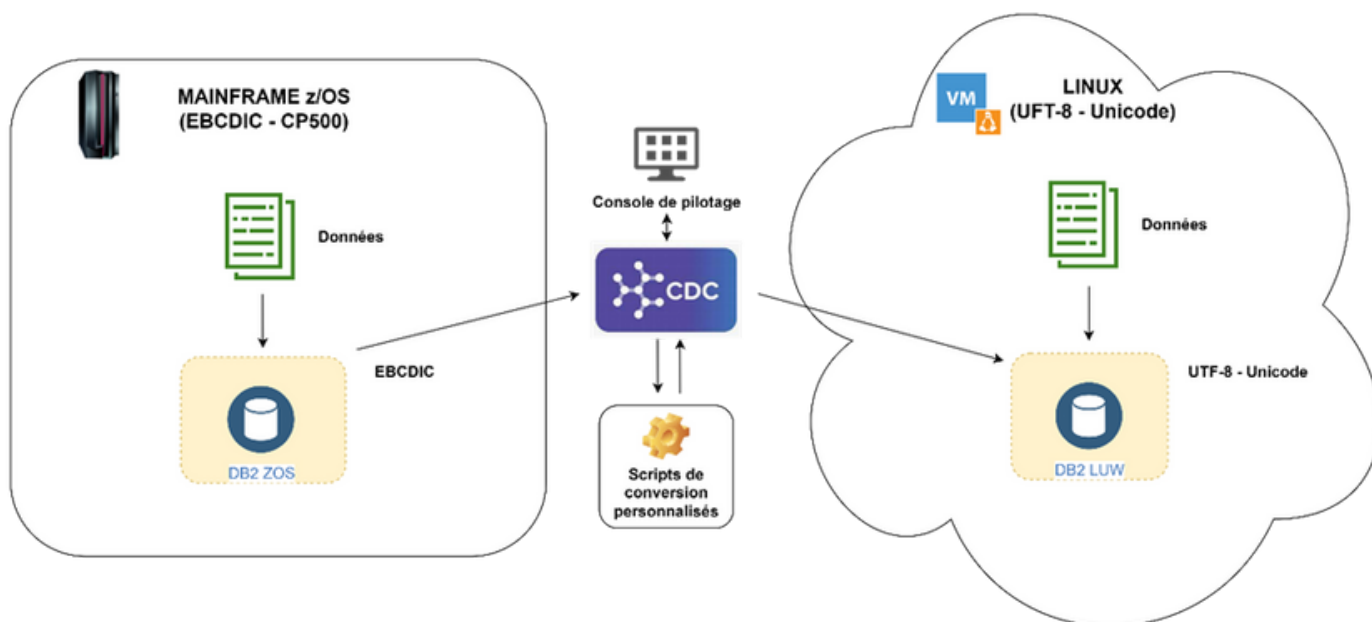
L'EXPÉRIENCE CLIENT

Focus sur la migration de données

Signalons que la reprise de données hétérogènes, avec des codes page distincts au sein de certaines tables DB2 côté *legacy*, peut se traduire par l'apparition de corruptions de données dans le système DB2 cible sur Linux !

C'est pourquoi **Infotel** a accompagné l'équipe support DBA du client dans la mise en place de scripts de conversion de données personnalisés afin de garantir une intégrité des données optimale entre la base DB2 *legacy* et la cible.

Ci-après, un **schéma de conversion des données** utilisé lors du projet de migration :

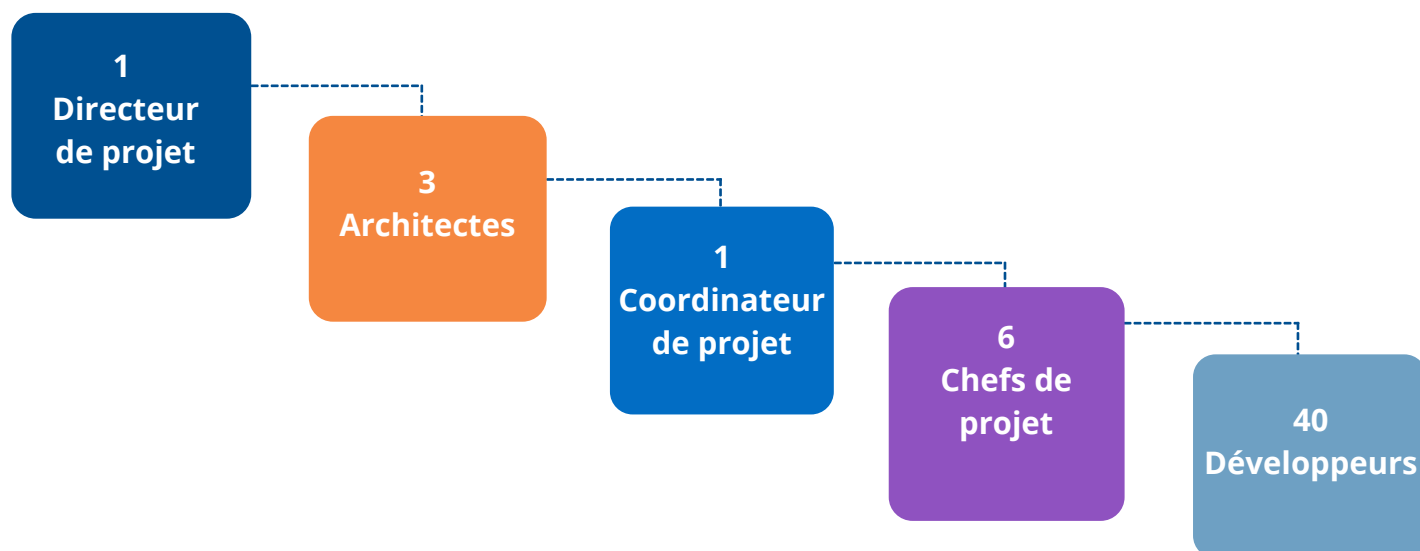


L'EXPÉRIENCE CLIENT

Organisation projet

En termes d'organisation projet, une équipe d'architectes, sous la direction d'un directeur de projet, coordonne l'implémentation technique et supervise l'avancement de six équipes projet déployées par **Infotel** pour couvrir l'intégralité du périmètre applicatif à migrer.

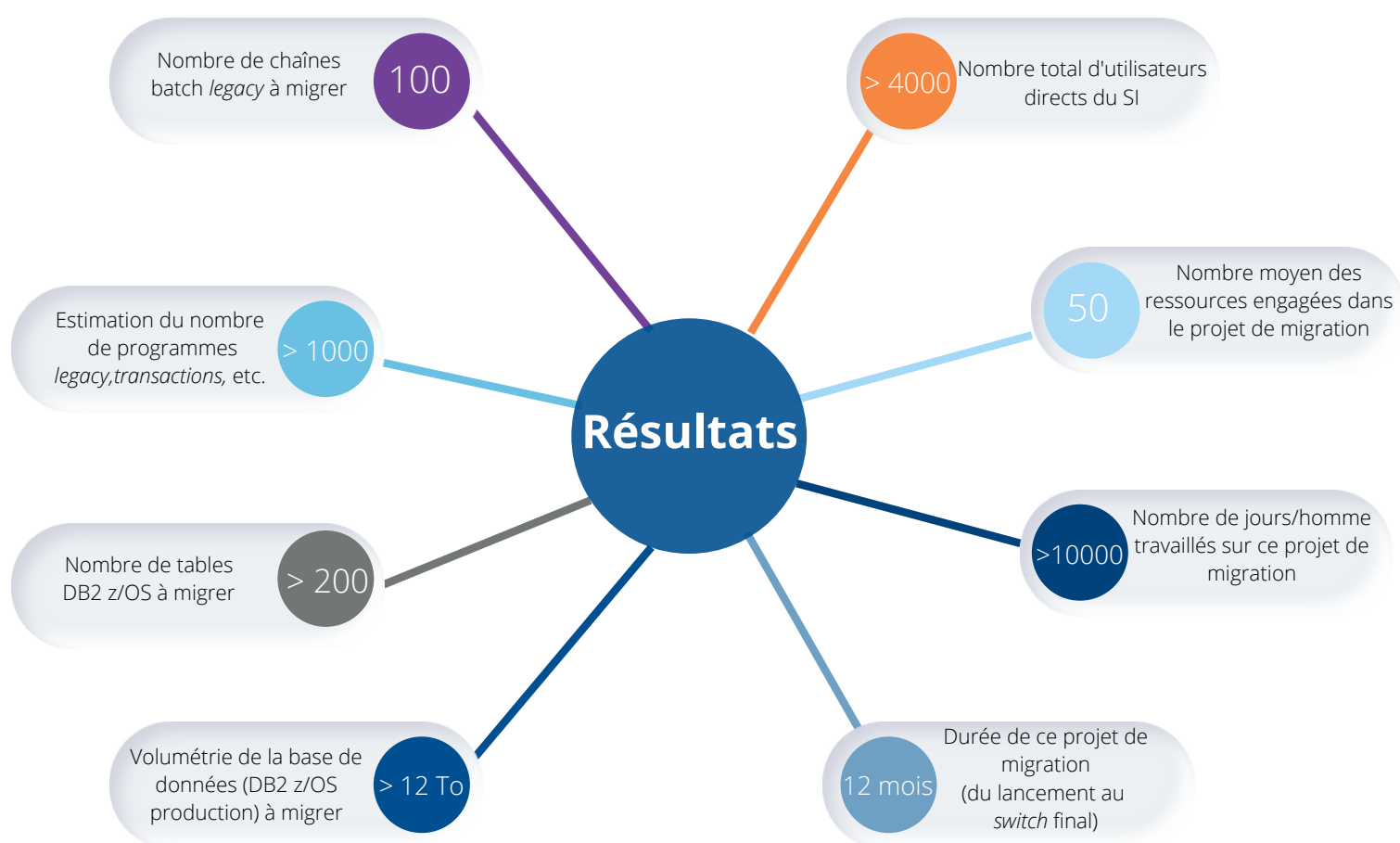
Les équipes de développement sont autonomes et agiles (Scrum), afin de délivrer au client les différentes fonctionnalités au fil de l'eau.



L'EXPÉRIENCE CLIENT

Résultats

Ci-après, quelques indicateurs résumant ce projet de *refactoring* du *mainframe* :



CONCLUSION

L'aboutissement d'un tel projet de désengagement du *mainframe* au profit de technologies plus modernes marque un virage majeur dans la perception de la qualité du SI par l'ensemble des utilisateurs.

En accompagnant le client dans sa démarche et en pilotant son projet de migration, depuis l'analyse initiale des besoins jusqu'à la livraison finale, les équipes mises en place par **Infotel** ont démontré un savoir-faire technique, une capacité à relever des challenges inédits en termes de planning pour des migrations d'envergure et une agilité dans la résolution des problèmes rencontrés.

Au terme de nos migrations réussies, la satisfaction du client est non seulement liée à la maîtrise des risques et au maintien du niveau de qualité de service (SLA) durant le projet, mais également à la réduction des coûts de maintenabilité et aux nouvelles projections d'évolution et d'interopérabilité désormais rendues possibles grâce à ce *gap* technologique.