

Du code existant à l'intégration de l'IA :

Modernisation agentique pour des logiciels sur mesure

Stratégies et pratiques pour les CIO et les décideurs informatiques

adesso |

SOMMAIRE

- 3 | Quand la technologie d'hier bloque la stratégie de demain
- 4 | Pourquoi maintenant ?
- 6 | Pourquoi la modernisation traditionnelle a atteint ses limites
- 8 | Comment l'IA transforme la modernisation : adSCA/LE l'approche agentique d'adesso
 - 10 | Phase 1
 - 13 | Phase 2
- 16 | L'humain au cœur du processus
- 17 | adesso comme partenaire pour la modernisation

Quand la technologie d'hier bloque la stratégie de demain

Deux entreprises sur trois exploitent des applications si obsolètes qu'elles freinent l'efficacité de toute l'organisation.¹

La modernisation n'est plus une simple question, c'est une crise permanente.

Les cycles de développement technologique liés à l'IA s'accroissent à un rythme qu'il est encore difficile d'imaginer il y a quelques mois. Ce qui n'était hier qu'un outil basé sur l'IA pour rédiger des e-mails automatise aujourd'hui des processus entiers. Ce qui était expérimental il y a deux ans est désormais un standard de production. Cette dynamique ne touche pas seulement les nouvelles technologies : elle redéfinit fondamentalement les critères d'évaluation des systèmes existants. Cela a des conséquences majeures pour les logiciels sur mesure développés il y a dix, quinze ou vingt ans. Les systèmes basés sur COBOL, les applications Java d'entreprise reposant sur J2EE, JEE, JSF, Struts ou EJB 2.x, et les solutions personnalisées sur d'anciennes architectures .NET constituent encore la base de processus importants dans de nombreuses entreprises. Près de la moitié d'entre elles exploitent des applications âgées de 11 à 20 ans, et plus d'une sur dix utilise même des logiciels vieux de 21 à 30 ans.² 40% de ces systèmes vieillissants sont des solutions développées sur mesure, pour lesquelles aucun éditeur de logiciels ne propose de plan de migration direct.³

La particularité des logiciels sur mesure réside dans le fait que les entreprises les ont développés pour obtenir un avantage concurrentiel spécifique. Et c'est exactement ce qu'ils ont fait. Aucun produit standard n'aurait pu modéliser aussi bien leurs processus. Mais contrairement aux logiciels standards, aucun éditeur ne travaille en arrière-plan pour fournir des mises à jour, aider à la migration ou imposer des échéances de fin de support. Avec un logiciel sur mesure, cette responsabilité incombe entièrement à l'entreprise seule.

En réalité, cela signifie souvent que le sujet est repoussé jusqu'à ce que le sujet devienne critique. Les coûts de maintenance augmentent, les mises en conformité réglementaires exigent des efforts disproportionnés et les initiatives d'innovation (intégration de l'IA, migration vers le cloud et les nouveaux services numériques) sont retardées parce que les systèmes centraux manquent de connectivité. Ce n'est pas un défaut du logiciel en soi, mais la conséquence de l'absence d'un mécanisme de modernisation continue.

Ce livre blanc se concentre sur la modernisation de ces systèmes : des logiciels sur mesure basés sur du code vieillissant, indépendamment de la technologie utilisée.

Si l'urgence de moderniser existe depuis des années, les outils ont fondamentalement changé. Aujourd'hui, les approches basées sur l'IA rendent les projets de modernisation plus rapides, plus rentables et plus faciles à gérer que jamais. Ce livre blanc présente les stratégies qui existent et montre comment une approche agentique redéfinit la modernisation.

¹ Lünendonk Étude « IT Modernization between Legacy, Cloud, and AI », 2025

² <https://www.der-bank-blog.de/veraltete-legacy-systeme-technologien/studien/37724967/>

³ <https://www.thinkwisesoftware.com/de/blog/neue-studie-legacy-modernisierung-von-computerwoche>

Pourquoi maintenant ?

Plusieurs évolutions convergent, et leur combinaison crée une pression telle que les entreprises ne peuvent plus l'ignorer.

L'IA exige des architectures modernes.

Pour de nombreuses entreprises, l'intelligence artificielle est passée d'une simple expérimentation à une priorité stratégique. L'IA se nourrit de données structurées, accessibles et disponibles en temps réel. Elle exige également des systèmes capables d'intégrer des services d'IA via des API. Les applications héritées en sont tout simplement incapables. Elles stockent les données dans des formats propriétaires, communiquent via des protocoles obsolètes et manquent d'interfaces pour une intégration moderne de l'IA. Pour exploiter l'IA efficacement, vous devez d'abord moderniser vos systèmes centraux. Sans ce socle, les stratégies d'IA ne restent que des présentations PowerPoint.

Les nouvelles exigences dépassent les capacités des systèmes vieillissants.

Modèles digitaux, portails en libre-service, traitement des données en temps réel, intégration aux plateformes : les stakeholders attendent de nouvelles fonctionnalités en quelques semaines, pas en plusieurs mois. Les anciens systèmes ne peuvent structurellement pas suivre le rythme. Ce n'est pas parce qu'ils ont été mal conçus, mais parce qu'ils ont été pensés pour une époque où les cycles de livraison étaient plus longs et les exigences d'intégration étaient minimales. La modernisation de vos systèmes centraux est donc un prérequis pour presque tous les objectifs de votre agenda stratégique.

La fenêtre pour une modernisation basée sur la connaissance se referme.

Le changement humain n'est pas une prévision abstraite : c'est une réalité vécue dans les départements informatiques. Lorsque les derniers experts d'un ancien système partent à la retraite, ils emportent avec eux bien plus qu'un simple savoir-faire technique. Ils emmènent aussi la compréhension des règles codées en dur dans le logiciel. Pourquoi le système calcule-t-il ce cas particulier de cette manière ? Quelle exigence réglementaire justifie cette logique de validation ? Ces connaissances sont rarement documentées. Elles existent de manière implicite, dans l'esprit de personnes qui quitteront bientôt l'entreprise. Si vous attendez que ces experts soient partis, vous moderniserez à l'aveugle à l'avenir. Chaque année de retard rend la transition non seulement plus coûteuse, mais aussi beaucoup plus risquée.

Les exigences réglementaires se durcissent rapidement.

La directive NIS2 introduit de nouvelles normes strictes en matière de cybersécurité, tandis que DORA impose une résilience opérationnelle numérique aux institutions financières. Parallèlement, l'application du RGPD devient de plus en plus rigoureuse. Les réglementations sectorielles ne cessent de se multiplier. Les systèmes ne doivent pas seulement fonctionner : ils doivent être manifestement sécurisés, auditable et conformes. Dans ces conditions, les anciens systèmes dotés d'une documentation partielle et d'une couverture de test incomplète deviennent un risque majeur, non seulement pour le département informatique, mais aussi pour la direction générale.

L'équation économique évolue.

La maintenance des systèmes vieillissants consomme des ressources financières qui pourraient être utilisées ailleurs. Les analyses montrent régulièrement que la maintenance des systèmes existants consomme 60 à 80 % des dépenses informatiques. Plus les systèmes sont anciens, plus cette part augmente, et moins il reste de place pour l'innovation. La modernisation n'est pas une simple dépense supplémentaire. Menée à bien, elle déplace l'équilibre entre les budgets de fonctionnement (« run ») et de transformation (« change ») au profit de l'avenir.

Chacun de ces facteurs constituerait à lui seul un moteur puissant. Combinés, ils mettent en évidence une évidence : le risque de ne rien faire dépasse désormais largement celui de la modernisation.

[Encadré décisionnel]

La modernisation applicative n'est pas un sujet anodin. Le blocage des stratégies d'IA, la responsabilité réglementaire, la perte des connaissances et la réduction de la marge d'innovation la placent directement à l'ordre du jour de la direction. Ne pas agir aujourd'hui, c'est risquer non seulement une hausse des coûts, mais aussi une paralysie stratégique.

Facteurs de la modernisation applicative

Facteurs technologiques

- Stack technologique EoL
- Absence de mises à jour de sécurité
- Pénurie de compétences
- Maintenance difficile et coûteuse
- Risque de pannes système
- Problèmes d'intégration
- Perte de données

Facteurs métiers

- Time-to-market
- Flexibilité face aux opportunités du marché
- Attentes des clients
- Pression concurrentielle
- Optimisation des coûts

Facteurs réglementaires et de sécurité

- Conformité
- Cybersécurité
- Auditabilité

Facteurs stratégiques

- Transformation digitale des modèles
- Intégration de technologies innovantes comme l'IA
- Continuité des activités
- Adoption par les utilisateurs

Pourquoi la modernisation traditionnelle a atteint ses limites

Par le passé, les projets de modernisation étaient fastidieux, consommateurs de ressources et très risqués. Les analystes passaient des mois à analyser le code, rédiger de la documentation, mener des entretiens et cartographier les dépendances. Bien que rigoureuse, cette approche ne passe tout simplement pas viable aujourd'hui. L'effort augmente de manière presque linéaire avec la taille de votre parc applicatif. Pour les grands environnements comptant des dizaines ou des centaines de systèmes, les phases d'analyse s'éternisent pendant des mois avant même de pouvoir prendre une décision. Les équipes projet restent mobilisées pendant de longues périodes sans moderniser une seule ligne de code, et les budgets s'épuisent dès la phase d'analyse, bien avant que la transformation proprement dite ne commence.

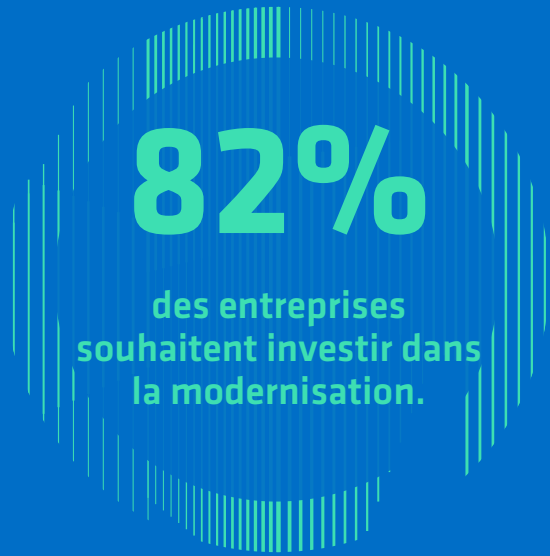
C'est précisément pour cela que de nombreuses entreprises repoussent sans cesse la modernisation, même lorsqu'elles en connaissent la nécessité. Les projets sont tout simplement trop massifs, coûteux et risqués pour être gérés en parallèle des activités quotidiennes.

En pratique, nous constatons également un schéma récurrent : des projets de modernisation qui échouent en phase de mise en œuvre. Et ce, malgré une analyse approfondie, parce que les équipes ont sous-estimé la complexité réelle de leurs anciens systèmes. Des dépendances cachées ne surgissent que lors de la migration. Des règles critiques, enfouies dans le

code hérité, n'ont jamais été documentées et font défaut lorsque le nouveau système ne parvient pas à les reproduire. Après des mois de travail acharné, les équipes réalisent que la stratégie choisie n'est pas viable et doivent tout recommencer depuis le début.

Même lorsque les projets aboutissent, de nombreuses approches ne résolvent qu'une partie du problème. Le rehosting déplace les applications vers le cloud sans traiter leurs faiblesses sous-jacentes. Vous migrez simplement votre dette technique vers une nouvelle infrastructure. Les convertisseurs de code syntaxiques traduisent le code dans un nouveau langage, sans en améliorer l'architecture. Le code compile dans un framework moderne, mais la structure monolithique, le manque de testability et les dépendances opaques subsistent. C'est ce que nous appelons du « legacy moderne dans une nouvelle technologie » : un système d'apparence moderne, mais tout aussi difficile à faire évoluer que son prédécesseur. De leur côté, les reconstructions complètes (greenfield) sous-estiment systématiquement la complexité fonctionnelle profonde et les règles implicites du système hérité, ce qui donne un nouveau système qui fait en réalité moins de choses que celui qu'il devait remplacer.

Ce qui manquait jusqu'à présent, c'était une approche combinant la profondeur de l'analyse manuelle avec la rapidité et l'évolutivité du traitement automatisé.



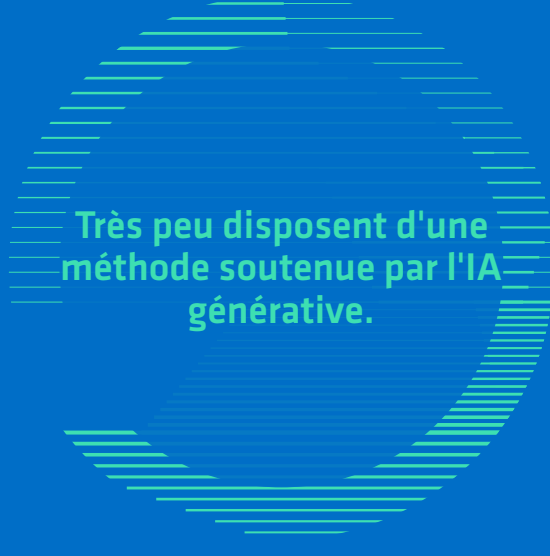
82%

des entreprises souhaitent investir dans la modernisation.



62%

des entreprises admettent que leurs applications centrales sont obsolètes.



Très peu disposent d'une méthode soutenue par l'IA générative.

Comment l'IA transforme la modernisation : adSCA/LE, l'approche agentique d'adesso

Jusqu'à présent, la modernisation applicative était avant tout quelque chose de complexe. Comprendre, évaluer et transformer des systèmes vieillissants comptant des centaines de milliers de lignes de code mobilisait des spécialistes hautement qualifiés, pendant des mois ou des années. C'est précisément ce qui rendait la modernisation coûteuse, et incitait les entreprises à la reporter sans cesse.

Avec adSCA/LE, adesso change la donne. Qu'est-ce que adSCA/LE ?

adSCA/LE signifie adesso Smart Cycle for AI-enhanced Lightweight Software Engineering : un cadre qui réunit des agents d'IA et l'expertise humaine dans un cycle structuré et reproductible. Ce framework comprend des meilleures

pratiques, des modèles et des normes afin de définir, mettre en œuvre et optimiser les flux agentiques dans l'ingénierie logicielle.

Le principe fondamental : l'humain orchestre, l'IA exécute. Des agents d'IA spécialisés prennent en charge les tâches intellectuelles répétitives (analyse, exploration du code, documentation et détection des dépendances). Les humains conservent le contrôle stratégique : ils valident les résultats, priorisent les applications, définissent les états et prennent les décisions critiques pour l'entreprise. Il en résulte des cycles de développement ultra-courts, des itérations plus rapides, une meilleure qualité et une intégration durable des approches d'IA agentique. adSCA/LE repose sur plus de 25 ans d'expérience en ingénierie logicielle, une connaissance

adSCA/LE

sectorielle approfondie et plus de six ans d'expérience dans les technologies d'IA. Cette alliance de maturité et de compétence fait toute la différence entre une simple expérimentation d'IA et une approche de modernisation.

adSCA/LE peut également accélérer massivement le développement de logiciels embarqués.

Dans les systèmes industriels et critiques pour la sécurité, ce sont la traçabilité, la certifiabilité et la souveraineté des données qui garantissent le succès.

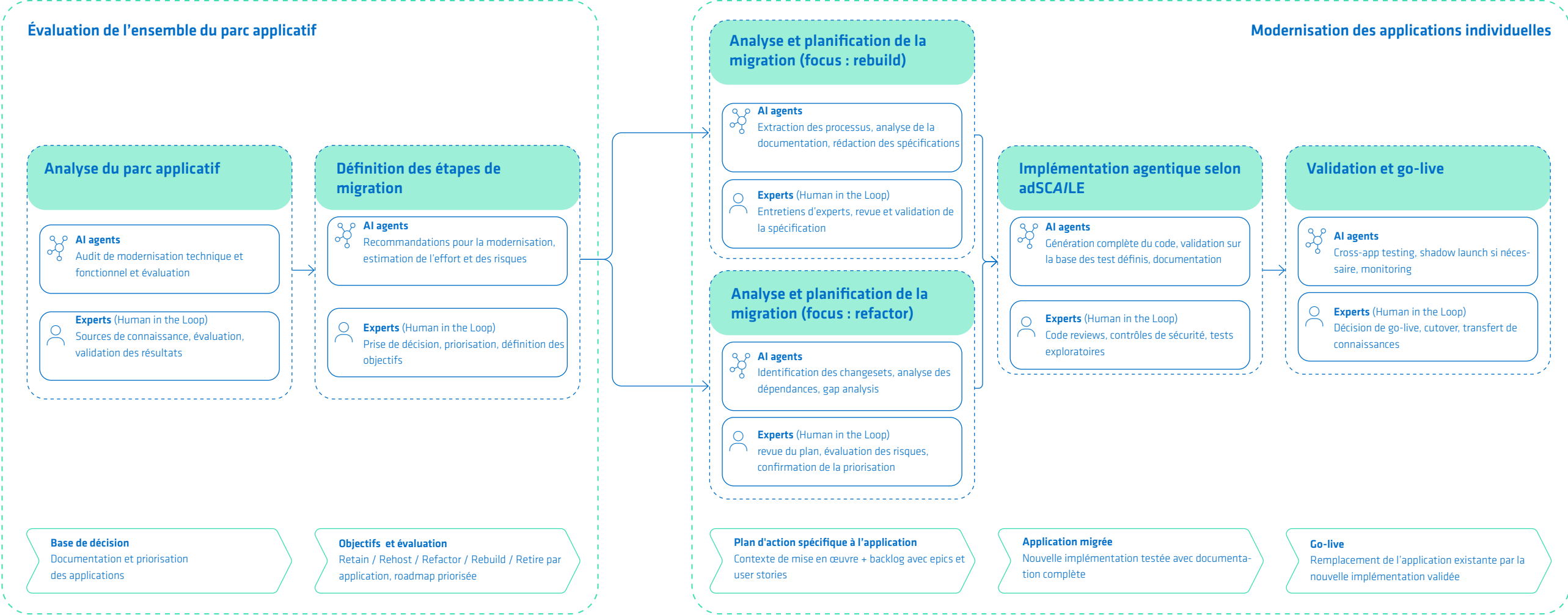
Alors que les projets logiciels classiques doivent avant tout répondre à des exigences fonctionnelles, des critères de réussite supplémentaires s'appliquent aux systèmes embarqués et en temps réel : comportement temporel déterministe, dépendances matérielles, exigences réglementaires et protection de la propriété intellectuelle. En particulier dans les environnements normés (par ex. IEC

61508 ou IEC 62304), l'assistance de l'IA doit être documentée et validée par des responsables qualifiés.

adSCA/LE transpose les avantages du développement logiciel basé sur des agents à ces conditions-cadres particulières. Le framework combine le développement assisté par IA avec une gouvernance « Human-in-the-Loop », une traçabilité complète et une assurance qualité conforme aux normes. Cela permet non seulement d'accélérer les cycles de développement, mais aussi d'améliorer durablement l'auditabilité, la conformité et la qualité technique. adSCA/LE génère des cas de test, des tests de régression et des données de test. La vérification finale dans les environnements embarqués s'effectue ensuite sur le matériel cible, sur des simulateurs ou dans un environnement HIL.

adSCA/LE aide les équipes de développement dans le domaine de l'embarqué à créer, tester et documenter de manière nettement plus efficace des logiciels certifiables et traçables.

Approche « agentic-first » de la modernisation applicative →



Phase 1:

Comprendre et évaluer

Étape 1 :

Analyse du parc applicatif

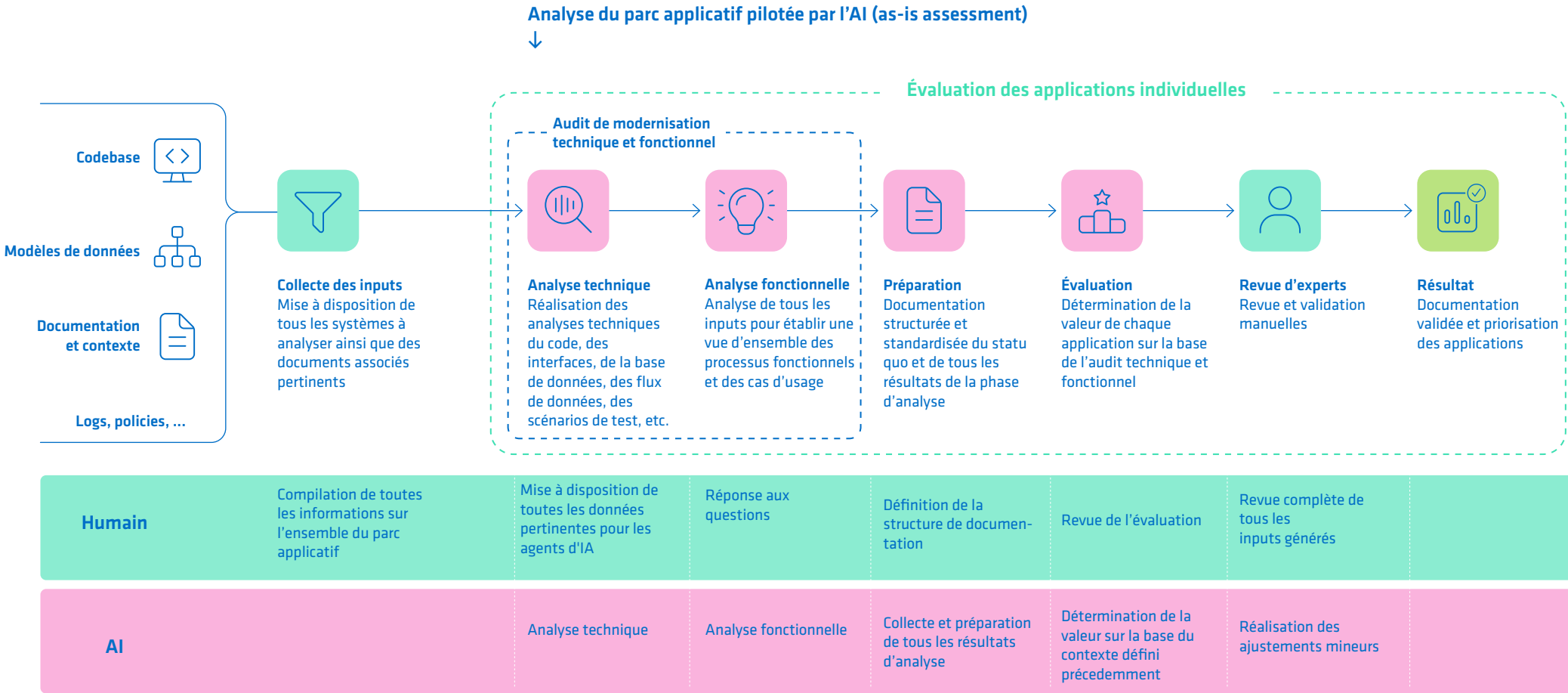
Traditionnellement, l'analyse complète d'un grand parc applicatif prend des mois : les analystes lisent le code, mènent des entretiens, documentent les dépendances et rédigent des évaluations. L'effort augmente de manière linéaire avec la taille du parc, et dépend de la disponibilité des bons experts.

Les agents d'IA transforment fondamentalement cette phase. Ils analysent automatiquement les bases de code, les architectures, les modèles de données, les interfaces API, les scénarios de test et les activités, même au-delà des frontières des systèmes. Ils en extraient les informations pour générer une documentation technique et fonctionnelle complète, structurée et standardisée. Ce qui s'apparentait auparavant à une boîte noire (Qui comprend ce code ? Quels processus y sont intégrés ? À quoi ne faut-il absolument pas toucher ?) devient rapidement transparent grâce à l'analyse assistée par l'IA.

Les experts humains apportent leur contribution et valident les résultats. Ils veillent à ce que le contexte métier soit correctement saisi et ajoutent des connaissances qui ne figurent pas dans le code lui-même : contexte commercial, contraintes réglementaires et dépendances organisationnelles.

Le résultat est un audit de modernisation complet : une transparence totale sur les dépendances, la dette technique, les risques et les coûts, constituant une base de décision fiable pour la direction et les départements. Le ROI de cette phase d'analyse est positif en soi.

Même si une entreprise décide, après l'analyse, de laisser certaines applications en l'état, elle dispose désormais d'une base de décision solide qui n'existait pas auparavant. Cela réduit considérablement la barrière à l'entrée d'un programme de modernisation, qui est un argument de poids à présenter à la direction.



[Encadré décisionnel]

Les analyses assistées par l'IA fournissent rapidement une base de décision fiable, avec un ROI positif. Vous savez ensuite exactement où se situent vos plus grands risques, quels systèmes doivent être modernisés en priorité et combien cela coûtera. Sans cette transparence, les décisions de modernisation ne reposent que sur l'intuition.

Étape 2 :

Définition du parcours de migration

Sur la base de l'analyse, les agents d'IA proposent le parcours de migration approprié pour chaque application ou composant. Ils évaluent la dette technique, la valeur, les risques et la complexité, et en déduisent une recommandation d'action selon les « 7 R » :

Retain (conserver) : laisse délibérément les systèmes stables dans leur état actuel : une décision active associée à une réévaluation régulière.

Rehost (réhéberger) : déplace l'application vers un nouvel environnement sans modification du code. Une première étape pragmatique qui résout les problèmes d'infrastructure mais ne traite pas les faiblesses architecturales.

Replatform (re-plateformer) : va plus loin et adapte sélectivement l'application pour tirer parti des avantages de la plateforme, tels que les Managed Services ou la mise à l'échelle automatique.

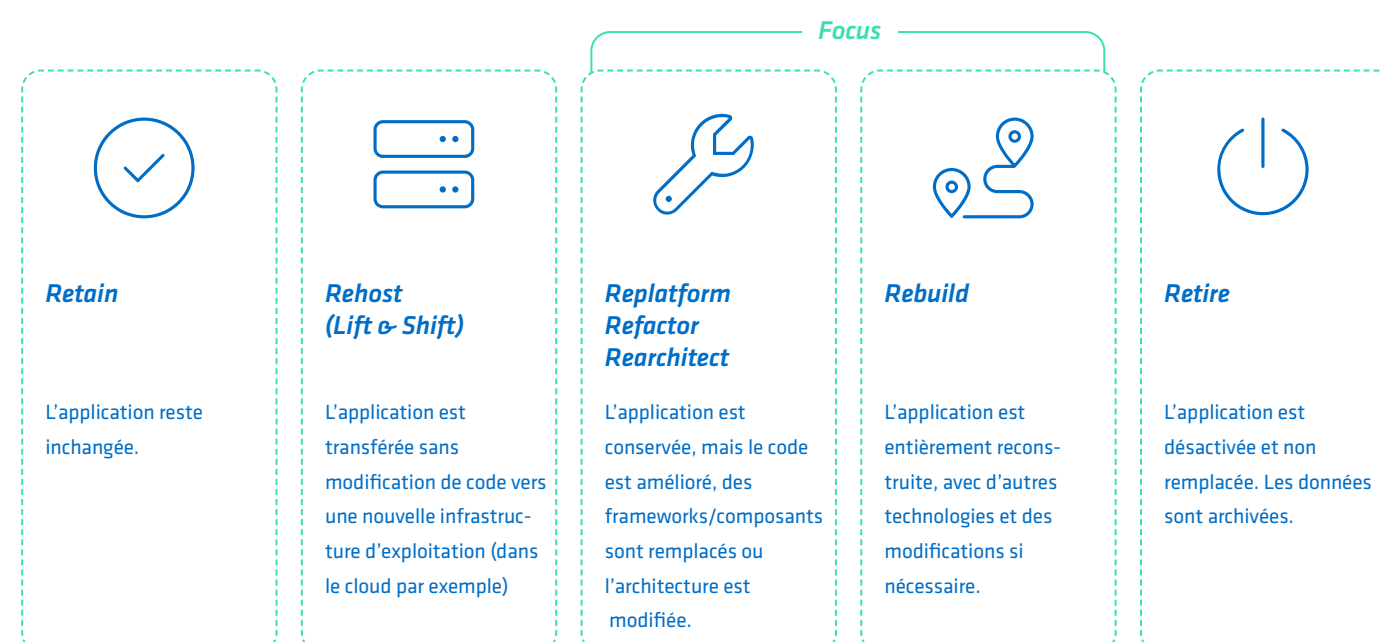
Refactor (refactoriser) : restructure le code interne : moins de complexité, plus de modularité, meilleure testabilité, sans modifier le comportement externe.

Rearchitect (réarchitecturer) : modifie fondamentalement l'architecture, par exemple en passant d'un monolithe à une structure orientée services.

Rebuild (reconstruire) : consiste à rebâtir à partir de zéro sur la base de la logique extraite : utile lorsque le socle technique est si obsolète que toute modernisation serait plus coûteuse qu'un nouveau départ.

Retire (retirer) : est enfin la mise hors service délibérée et ordonnée. Elle comprend l'archivage des données, le transfert de connaissances et la prise en compte des obligations légales de conservation : un processus de fin de vie contrôlé qui réduit les coûts et apporte de la clarté.

Les experts prennent la décision finale et fixent les priorités. En pratique, il n'y a pas un parcours unique, mais plutôt un portefeuille différencié de stratégies, justifié sur le plan économique et loin de toute approche uniforme.



Phase 2: Modernisation d'applications individuelles

Étape 3 :

Analyse spécifique à l'application et planification de la migration

Une fois le parcours de migration d'une application défini, la planification détaillée commence. Le processus diffère selon la stratégie choisie. Le choix entre Refactor et Rebuild est l'une des décisions les plus lourdes de conséquences de tout le processus de modernisation.

Refactor est la bonne voie lorsque la base de code existante est fondamentalement viable mais souffre de dette technique, de structures obsolètes ou d'un manque de testabilité. La logique métier est préservée et le code est modernisé de l'intérieur. Les agents d'IA créent des modifications et effectuent des analyses : où le code existant s'écarte-t-il des normes modernes ? Quelles dépendances doivent être résolues ? Quels composants peuvent être isolés ? Les experts évaluent les modifications proposées et garantissent le respect du contexte.

L'option Rebuild devient préférable lorsque le socle technique est si obsolète ou si complexe que toute amélioration coûterait plus cher qu'un nouveau départ. Les agents d'IA extraient les processus et les exigences existants du code hérité. Ils identifient les règles, les flux de données et les contrats d'interface, puis les transposent dans une structure. Les experts examinent cette extraction, y ajoutent le contexte et valident la spécification. C'est particulièrement crucial lorsque des connaissances implicites sont ancrées dans le code et ne peuvent être correctement transposées sans interprétation humaine.

Étape 4 :

Mise en œuvre agentique

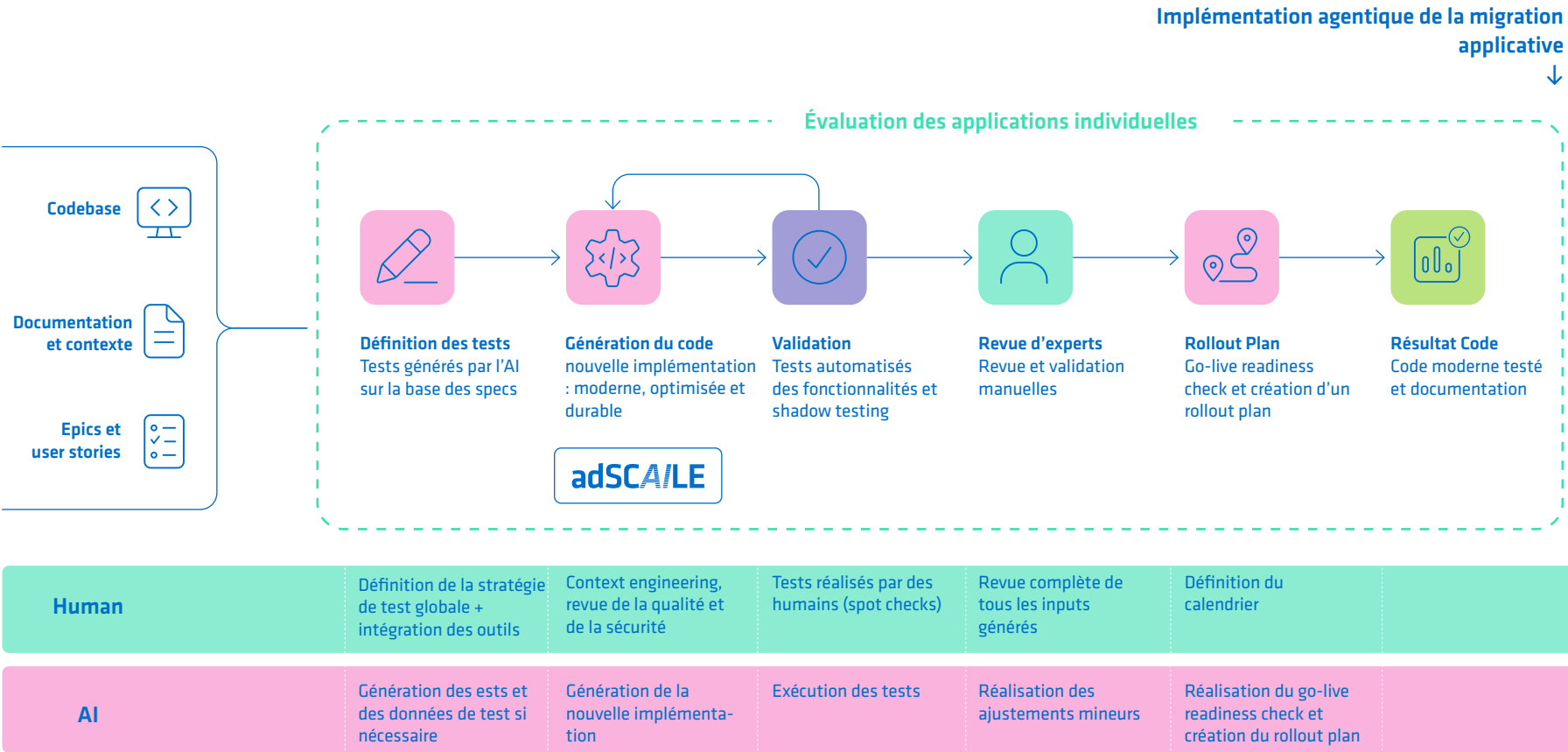
Pendant la phase de mise en œuvre, adSCALE déploie tout son potentiel. Les agents d'IA prennent en charge la génération de code et la documentation, créant automatiquement des propositions de refactorisation au lieu de nécessiter un effort manuel. Lors de la reconstruction, la logique métier extraite sert de base à un nouveau code moderne, tandis que le système génère automatiquement des tests d'architecture.

Les experts examinent et valident ensuite les résultats. Le processus reste le même que dans un projet d'ingénierie logicielle classique. Seul l'effort requis change. Les projets de modernisation fournissent des résultats valides dans des délais nettement plus courts après l'analyse des applications individuelles.

Étape 5 :

Validation et mise en service

Les agents d'IA effectuent des tests ainsi qu'une surveillance automatisée. Le shadow testing (l'exécution parallèle de l'ancien et du nouveau système) garantit que les applications modernisées sont fonctionnellement identiques, voire améliorées. Les experts prennent la décision finale de mise en service.



L'humain au cœur du processus :

À aucun moment du processus l'IA ne prend de décisions autonomes sur des sujets critiques. Les agents d'IA analysent, génèrent, testent et documentent. Les humains valident, priorisent, décident, assument la responsabilité et apportent des connaissances implicites propres à l'entreprise, que l'IA ne peut pas connaître. Cette répartition du travail n'est pas un compromis : c'est la condition sine qua non pour que cette approche fonctionne de manière fiable dans des environnements réglementés.

La combinaison de l'automatisation par l'IA et de l'expertise humaine transforme fondamentalement la structure des coûts des projets de modernisation. En pratique, adesso permet de réduire les coûts de modernisation de 30 à 60 % par rapport aux approches traditionnelles. Les délais des projets sont nettement plus courts et les risques diminuent, car les analyses basées sur l'IA apportent rapidement de la transparence. Cela rend également les logiciels sur mesure à nouveau compétitifs face aux solutions standards coûteuses, permettant aux entreprises de continuer à modéliser leurs processus différenciateurs dans des logiciels personnalisés, à un coût qui rend la modernisation viable.

[Encadré décisionnel]

La phase d'analyse assistée par l'IA fournit rapidement une base de décision fiable, avec un ROI indépendant et positif. Vous saurez exactement quels sont vos plus grands risques, quels systèmes moderniser en priorité et combien cela coûtera. Sans cette transparence, les décisions de modernisation ne sont que de pures suppositions.

adesso comme partenaire pour la modernisation

Les programmes de modernisation dans les environnements de développement sur mesure sont complexes à long terme, et critiques pour l'entreprise. Ils nécessitent un partenaire qui ne se contente pas de fournir l'analyse initiale, mais qui vous accompagne tout au long du processus : de l'évaluation initiale à l'exploitation des systèmes modernisés.

adesso propose une combinaison rare d'atouts afin d'y parvenir.

Plus de 25 ans d'expérience en ingénierie logicielle.

adesso développe et modernise des solutions logicielles sur mesure depuis plus de deux décennies, sur une grande variété de stacks technologiques et de secteurs d'activité. Nous connaissons les schémas des parcs applicatifs vieillissants pour en avoir construit une grande partie nous-mêmes ou pour les avoir accompagnés pendant des années.

Plus de six ans d'expérience opérationnelle en IA.

adesso travaille avec l'IA et les grands modèles de langage (LLM) depuis plus de six ans. Cette maturité opérationnelle fait toute la différence entre une simple démonstration technologique et une approche de modernisation industriellement viable.

Une expertise sectorielle, pas seulement technologique.

Grâce à son expérience dans des secteurs clés (des services financiers et assurances à l'administration publique, en passant par l'automobile, l'énergie, l'industrie manufacturière et le commerce de détail) adesso comprend en détail les aspects commerciaux et réglementaires de votre secteur. La modernisation est toujours aussi un enjeu : la logique des anciens systèmes doit être comprise, évaluée et correctement migrée vers le nouvel environnement. Cela n'est possible qu'avec une véritable expertise sectorielle.

De A à Z, sans accroc.

adesso accompagne les programmes de modernisation de A à Z : de la stratégie et de la planification architecturale à la mise en œuvre et à l'exploitation. Il n'y a pas de rupture entre le conseil et l'implémentation, ni de perte de connaissances entre les phases du projet.

Indépendance technologique.

adesso n'est biaisé par aucun fournisseur technologique. Qu'il s'agisse d'Azure, d'AWS, d'une solution sur site ou hybride, notre recommandation pour une architecture repose uniquement sur vos besoins, et non sur un accord de partenariat.

Par où commencer : une clarté rapide

Le chemin le plus rapide vers la modernisation commence par une analyse de votre parc applicatif assistée par l'IA. En peu de temps, vous obtenez une évaluation complète de votre situation technique, de votre valeur, de vos risques et de vos priorités de modernisation. Cette phase d'analyse offre un ROI indépendant et fournit la base pour les décisions à venir.

Les projets pilotes se transforment en références fiables. Les références se transforment en un programme de modernisation qui pérennise systématiquement votre parc applicatif, non pas sous la forme d'un grand projet unique, mais comme une capacité permanente de renouvellement.

Identifions ensemble les leviers les plus puissants au sein de votre parc applicatif.



Michaela Gasser
CEO

marketing@adesso.ch

adesso Suisse SA
Rue Pichard 13,
CH-1003 Lausanne
T +41 58 520 97 10
E marketing@adesso.ch
www.adesso.ch