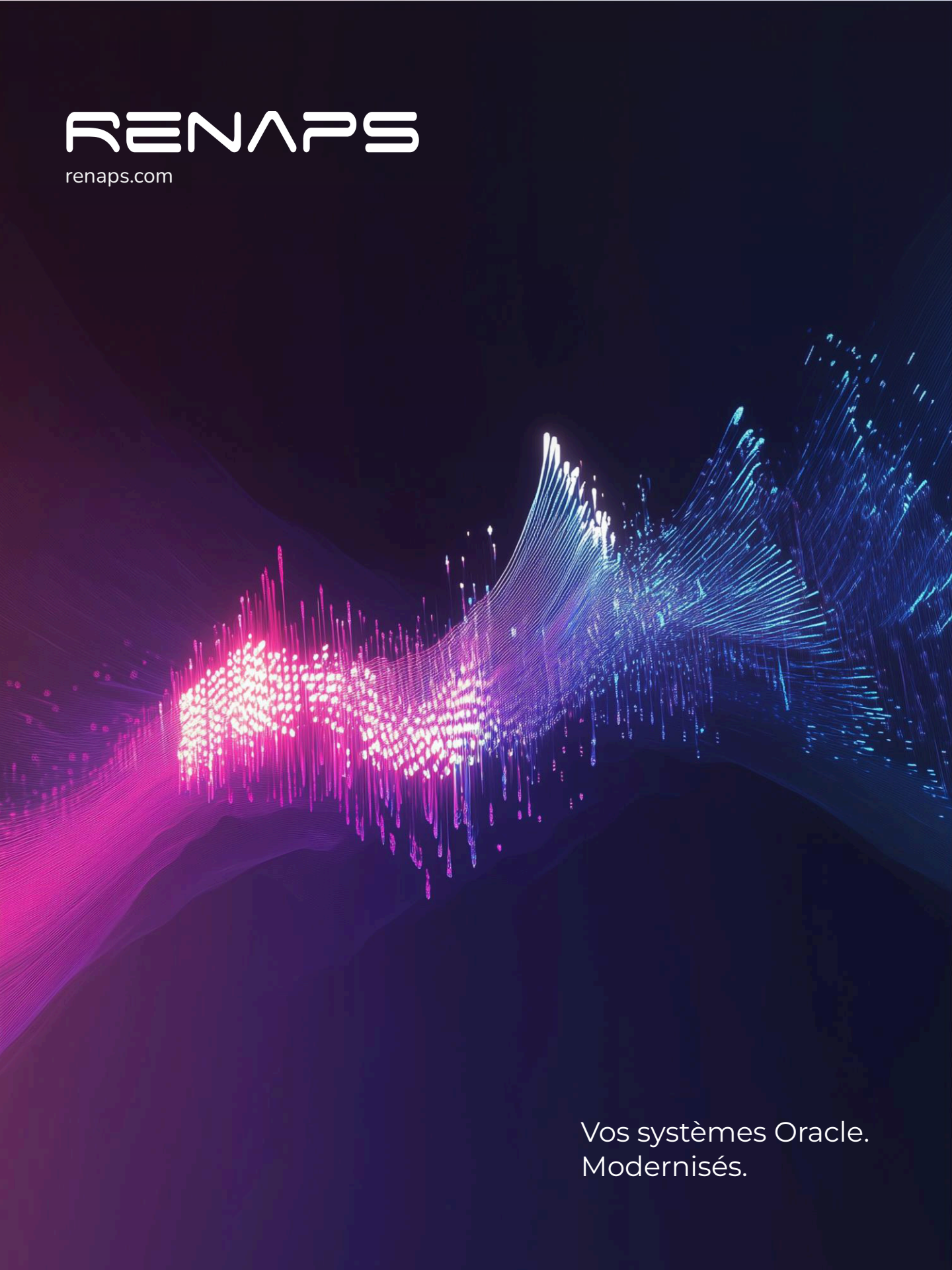
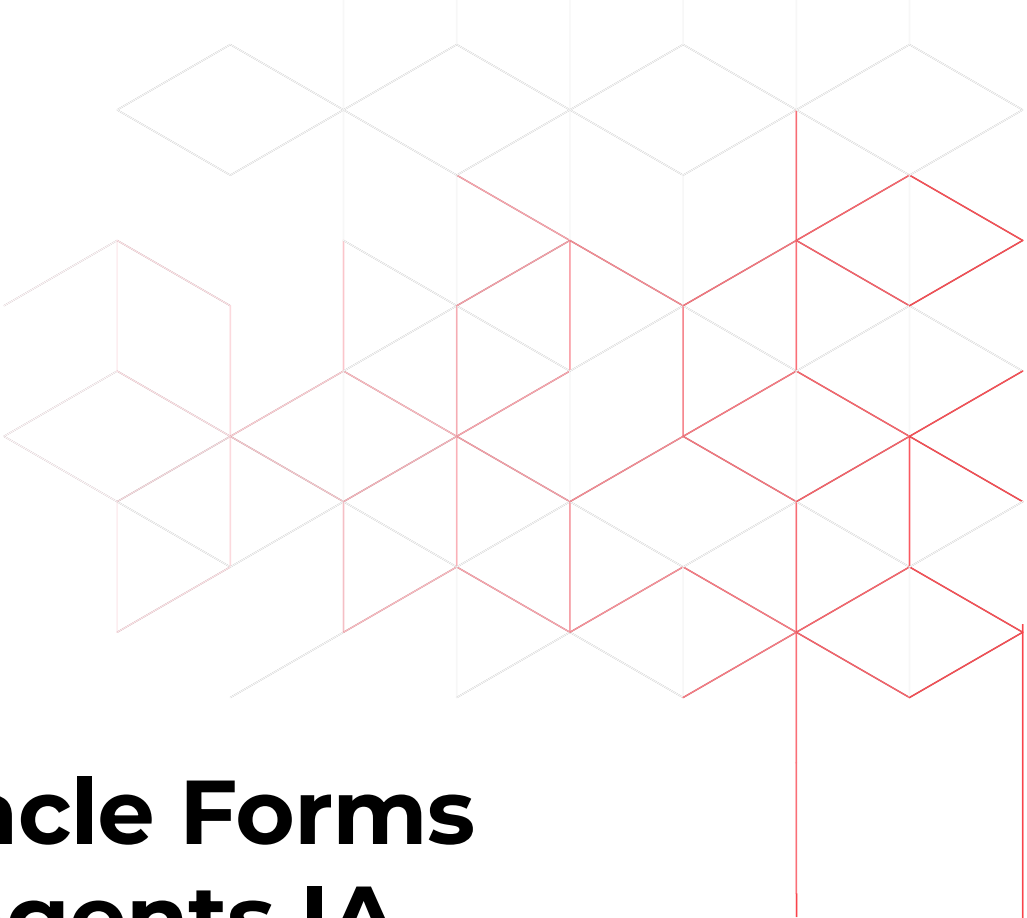


RENAPS

renaps.com



Vos systèmes Oracle.
Modernisés.



D'Oracle Forms aux agents IA. Migrer les règles, pas les écrans.

Oracle Forms à l'ère des agents :

Pourquoi une migration doit garder vos règles d'affaires exécutables, pour que l'application Web que vous déployez aujourd'hui soit celle qu'un agent IA pourra exploiter en toute sécurité demain, sans seconde migration.

Octobre 2026 · RENAPS

Par

RENAPS

Siège social, États-Unis et international : 2875 NE 191 Street, Suite 500, Aventura, Florida, 33180, USA

Siège social canadien : 101, Marcel-Laurin, bureau 300, Montréal (Québec) H4N 2M3, Canada

1-888-98-RENAPS (1-888-987-3627) · info@renaps.com

Table des matières

1.	SYNTHÈSE	05
2.	2026 : L'ANNÉE OÙ LE MARCHÉ A RÉÉVALUÉ LE LOGICIEL	06
	La SaaSocalypse, puis le rebond	06
	Ce que mesurent les analystes	06
	Les tendances de Gartner pour 2027 : de l'adoption à la gouvernance	07
	Même les éditeurs le disent	09
	Pourquoi les écrans ne sont plus toute la destination	09
3.	L'IA DE DEMAIN EST UN BUREAUCRATE	10
	Un agent, pas un outil	10
	Les bureaucraties qui fonctionnent sur Oracle Forms	10
	Pas de règles, pas d'agent	11
4.	LE PIÈGE DE LA RÉÉCRITURE	12
	Où vivent les règles	12
	Ce qu'une réécriture perd	12
	Pourquoi un assistant greffé ne suffit pas	12
	La réponse d'Oracle : APEX 26.1	13
5.	MIGRER LES RÈGLES, PAS LES ÉCRANS	14
	Des règles qui s'exécutent déjà sans écran	14
	Ce que vous obtenez aujourd'hui, et ce que cela permet demain	15
	Ce qui reste dans la base de données y reste	16
	Ce que cela rend possible	16
	Pour la migration	16
	Pour l'agent	16
	La couche de contexte que demande Gartner, et où un formulaire migré la contient déjà	17

Table des matières

6.	L'OFFRE : UNE APPLICATION QUE TOUT AGENT PEUT EXPLOITER	18
	Deux volets, une seule source de règles	18
	Trois façons d'exécuter un processus, une seule source de règles	19
	Un exemple concret	19
	Ce que les règles ne peuvent pas faire	20
	Ce que font vos équipes à la place	20
7.	SÛR PAR CONSTRUCTION	21
	Identité et responsabilité	21
	Une confiance progressive	21
	Vos données, votre région, votre choix de modèle	21
	Un coût mesuré	21
	Le test du refus	21
8.	LA PREMIÈRE ÉTAPE	23
	Ce qu'apporte RENAPS	24
	SOURCES	25

Pendant vingt ans, les organisations qui exploitent Oracle Forms se sont demandé *vers quelle pile Web* migrer. En 2026, la question a changé. Le marché du logiciel, les analystes et les plus grands éditeurs s'entendent désormais sur la direction que prennent les applications d'entreprise : le travail passe aux agents IA, et l'écran devient l'exception.

C'est une bonne nouvelle pour ces organisations, en raison de ce que contiennent leurs formulaires. Derrière chaque écran se trouvent des centaines de déclencheurs, de validations et de procédures : les règles qui déterminent ce qu'est une transaction valide, qui peut modifier quoi, et quand une écriture est refusée. Ces règles sont exactement ce dont un agent IA a besoin pour mériter la confiance dans un système transactionnel, et exactement ce qu'une réécriture, manuelle ou par IA, perd en chemin.

RENAPS migre les règles avec les écrans. OFCS (Oracle Forms Compatibility Service), au cœur d'ORMIT™ - OpenJava, conserve chaque déclencheur, validation et procédure sous forme de code exécutable derrière l'application React ou Angular livrée, et ce code s'exécute déjà sans écran. La migration que vous déployez aujourd'hui est donc prête pour les agents, et RENAPS bâtit sur elle en deux volets. D'abord, l'application migrée peut être exploitée par n'importe quel agent, le vôtre, celui d'un éditeur de plateforme ou celui de RENAPS : un connecteur ouvert (MCP, ou les mêmes opérations en ligne de commande ou par API), une identité propre à l'agent et un journal de chaque action, devant les mêmes règles qu'un employé, où un refus du formulaire arrête l'agent comme il arrêterait une personne. Ensuite, processus par processus, un agent prend le travail en charge, et vos employés passent de la saisie à la supervision, sur l'application existante.

Ce livre blanc montre

- Pourquoi 2026 a mis fin à l'idée qu'une application Web moderne est la destination finale, chiffres du marché, de Forrester et de Gartner à l'appui, dont ses tendances pour 2027, et pourquoi elle reste la première étape nécessaire.
- À quoi ressemblera l'IA d'entreprise de demain : un agent au travail dans les bureaucraties de la finance, de l'assurance, de la santé et du secteur public, celles qui tournent sur Oracle Forms.
- Pourquoi les règles de vos formulaires sont le contrôle qui manque à la plupart des projets d'IA agentique, et pourquoi les réécrire, c'est jeter ce contrôle.
- Comment une application migrée peut devenir exploitable par n'importe quel agent, en toute sécurité : un connecteur ouvert, une identité propre à l'agent et un journal de chaque action, devant les règles qui servent déjà vos utilisateurs. Puis comment confier un processus à un agent, vos équipes approuvant son travail.
- Une première étape assez petite pour être décidée ce trimestre : migrer, puis ajouter un agent à un processus, avec des résultats mesurés.

2026 : l'année où le marché a réévalué le logiciel

02

La SaaSpocalypse, puis le rebond

Le 12 janvier 2026, Anthropic a lancé Claude Cowork, un agent qui accomplit du travail professionnel avec les fichiers et les outils d'affaires d'un utilisateur. Le 30 janvier, il y a ajouté onze modules d'extension à code source ouvert, un par fonction de bureau : finance, revue juridique, ventes, service à la clientèle et d'autres. Le 5 février, OpenAI a lancé Frontier, une plateforme pour bâtir et exécuter des agents qui opèrent d'autres logiciels d'entreprise.

Les investisseurs ont lu le signal sur-le-champ. Entre le 12 janvier et le 23 février, l'indice S&P Software & Services a perdu le quart de sa valeur. La séance qui a suivi la sortie des modules a effacé à elle seule plus de 285 milliards de dollars américains de capitalisation boursière. Un stratège de Jefferies a donné son nom à l'épisode : la SaaSpocalypse.

-25 %

Indice S&P Software & Services, du 12 janvier au 23 février 2026

285 G\$

Valeur boursière effacée en une séance après la sortie des modules d'agents

+41 %

Le même indice, du 13 avril au 12 octobre 2026, revenu à ses sommets de l'année

La crainte était précise. Le logiciel d'entreprise se vend par utilisateur, et un agent qui exécute un flux de travail n'occupe aucune licence. Quand la logique passe à l'agent, l'application derrière lui devient une base de données dotée d'une interface que personne n'ouvre.

Puis le marché a changé d'avis, et c'est ce qui compte le plus. En octobre 2026, l'indice du logiciel avait regagné tout ce qu'il avait perdu : en hausse de 41 % depuis son creux d'avril, à de nouveaux sommets pour l'année, stable sur douze mois. La reprise n'a pas été menée par les entreprises qui n'iaient le changement, mais par celles qui avaient intégré des agents au cœur de leurs produits. Le verdict n'était pas que le logiciel est mort. C'était que le logiciel sans agents est réévalué à la baisse, et que le logiciel qui les intègre garde sa valeur.

Ce que mesurent les analystes

Forrester a répondu à la débâcle dans la semaine, sans en contester la direction. Quelle que soit la façon dont une entreprise déploie des agents, écrivait Forrester, « cela érodera le périmètre de vos applications SaaS d'entreprise actuelles », et la tarification par utilisateur cédera la place à une tarification à la consommation et au résultat.

Gartner y met des dates :



D'ici la fin de 2026, 40 % des applications d'entreprise intégreront des agents IA spécialisés par tâche, contre moins de 5 % en 2025.



D'ici 2028, le tiers des expériences utilisateur passeront des applications natives aux interfaces d'agents.



D'ici 2030, 234 milliards de dollars américains de dépenses en applications d'entreprise sont menacés par l'IA agentique, et l'interface utilisateur cesse d'être un facteur de différenciation.



D'ici 2028, plus de la moitié des entreprises cesseront de payer pour une IA qui assiste les employés, au profit de plateformes qui livrent le résultat d'un processus. Les éditeurs qui se contentent de greffer l'IA à des applications existantes s'exposent à une compression de leurs marges pouvant atteindre 80 % d'ici 2030.

Les tendances de Gartner pour 2027 : de l'adoption à la gouvernance

Le 20 octobre 2026, Gartner a publié ses 10 grandes tendances technologiques stratégiques pour 2027. Les agents n'y figurent plus comme une tendance. Ils sont la plateforme que la liste suppose déjà en place dans toute organisation, et la question est devenue de savoir ce qui les gouverne. Selon Gartner, la conversation en 2027 « passe de l'adoption à la gouvernance » : les TI traditionnelles reposaient sur la rareté du logiciel, 2027 en apporte l'abondance, et les responsables des TI « doivent passer de la gestion des actifs technologiques à la gouvernance du comportement des machines ».

+80 %

Précision de l'IA agentique pour les organisations qui priorisent la sémantique dans des données prêtes pour l'IA, d'ici 2027

25 %

Des atteintes à la sécurité en entreprise découleront d'un usage abusif d'agents IA, d'ici 2028

80 %

Des nouvelles applications seront rendues jetables et faciles à remplacer par les organisations de pointe, d'ici 2030

Trois des dix tendances décrivent ce qu'une migration d'Oracle Forms doit préserver :



Plateformes de contexte d'IA.

Les connaissances dont un agent a besoin « pour interpréter les intrants, prendre des décisions et agir en accord avec les objectifs d'affaires ». Gartner donne à cette couche trois composantes : la sémantique, qui inclut les règles d'affaires et les politiques; l'état opérationnel, les données en temps réel; et la provenance, la trace des décisions, des actions et des résultats. Sans elle, les organisations s'exposent à « des agents aux privilèges excessifs, des manquements à la conformité, une faible confiance et une consommation excessive de jetons ».



Identité des agents IA.

Les agents « exposent des vulnérabilités massives dans les systèmes hérités incapables d'accorder des accès juste-à-temps », et les organisations qui ne régleront pas la question verront plus de la moitié de leurs initiatives d'IA s'arrêter. Première consigne de Gartner : un agent ne partage jamais les identifiants d'une personne. Il reçoit une identité propre, avec des accès régis par des politiques.



Applications jetables.

La première action que cite Gartner consiste à « déterminer quelles capacités essentielles exigent un logiciel permanent et lesquelles peuvent être servies par des solutions temporaires, à la demande ». Le portefeuille d'applications devient « un système dynamique qui se renouvelle continuellement à mesure que les actifs redondants sont retirés ».

Lues ensemble, elles disent où se trouve la valeur d'une application Oracle Forms. Les écrans en sont la partie remplaçable. Les règles en sont le noyau permanent, et elles sont la seule composante de la couche de contexte de Gartner qu'aucun éditeur de plateforme ne peut écrire pour une organisation, parce qu'elles n'existent que dans ses formulaires. Une migration qui les garde exécutables, sous une identité que la base de données contrôle, est la part du programme 2027 qu'une organisation qui exploite Oracle Forms peut livrer à partir de ce qu'elle possède déjà. Le chapitre 5 détaille la correspondance.



Même les éditeurs le disent

En septembre 2026, Salesforce a ouvert sa conférence Dreamforce en abordant la débâcle de front. Son PDG, Marc Benioff, a écarté les discours sur la fin du logiciel, puis a concédé le point qui compte :

« La SaaS apocalypse ne portait pas sur la fin du logiciel, mais elle pourrait bien porter sur la fin du logiciel qui fait faire tout le travail aux humains. »

Marc Benioff, PDG de Salesforce, discours d'ouverture de Dreamforce, septembre 2026

Il l'a dit en lançant une interface d'agent pour les propres applications de Salesforce, bâtie avec Claude d'Anthropic. L'éditeur qui raillait l'apocalypse y a répondu en intégrant des agents. Un logiciel qui fait faire tout le travail aux humains, c'est la description exacte d'un écran de saisie. C'est aussi la description exacte de ce que livre une réécriture classique d'Oracle Forms : le même écran, dans un navigateur, sans les règles qui étaient derrière, perdues en chemin.

Pourquoi les écrans ne sont plus toute la destination

Rien de cela ne signifie que les écrans disparaissent. La saisie au clavier à haut volume reste plus rapide sur un écran, et la plupart des tâches en garderont un pendant des années. Cela ne signifie pas non plus qu'il faille attendre. Selon les dates de support publiées par Oracle, Forms 12c est couvert par le Premier Support jusqu'en décembre 2026, et Forms 14.1.2 jusqu'en 2030 : la migration a une échéance dans tous les cas, et elle doit avoir lieu. Cela signifie que la migration doit être choisie pour ce qu'elle rend possible ensuite. Un projet de migration dure d'un à trois ans; les chiffres ci-dessus décrivent le même horizon. Une migration qui livre des écrans et perd les règles finit là où elle a commencé. Une migration qui livre des écrans et garde les règles exécutables est la plateforme sur laquelle les agents fonctionneront, sans seconde migration.

La question pour une organisation qui exploite Oracle Forms en 2026

Non plus seulement « quelle technologie Web? », mais « quand un agent exécute ce processus sur l'application migrée, qu'est-ce qui l'empêche de mal faire? » Le prochain chapitre porte sur la nature de cet agent. Le suivant porte sur cette question, et sur la raison pour laquelle vos formulaires y répondent déjà, à condition que la migration garde la réponse.

L'IA de demain est un bureaucrate

03

Un agent, pas un outil

L'historien Yuval Noah Harari passe depuis deux ans à corriger l'image que la plupart des gens se font de l'intelligence artificielle. Dans sa Tanner Lecture de 2026 à Oxford, il a résumé la distinction en une phrase :

« Ce n'est pas un outil entre nos mains. C'est un agent qui a ses propres mains. »

Yuval Noah Harari, Tanner Lecture on Human Values, Oxford, 2026

Et il a dit où cet agent vivra. Pas dans la rue, ni dans un corps de robot :

« Comme les poissons vivent dans les océans et les singes dans les forêts, les IA vivent dans les bureaucraties. »

« Dans les années qui viennent, des millions de bureaucrates IA prendront de plus en plus le contrôle des bureaucraties du monde. Des banquiers IA décideront s'ils vous accordent un prêt. »

Yuval Noah Harari, même conférence

Les bureaucraties qui fonctionnent sur Oracle Forms

Harari le dit en guise d'avertissement. Pour une organisation qui fonctionne sur Oracle Forms, c'est aussi une prévision, et elle soulève une question : qui écrit les règles que suivra le bureaucrate IA ?

Une bureaucratie, au sens de Harari, est tout système qui permet à des inconnus de se faire confiance à grande échelle : une banque, un assureur, la facturation d'un hôpital, les permis d'un ministère, une paie. Ces systèmes sont faits de dossiers, de règles et d'approbations. Depuis trois décennies, une grande partie d'entre eux sont bâtis sur Oracle Forms, précisément parce que Forms rendait les règles et les approbations faciles à exprimer : un déclencheur sur chaque événement, une validation sur chaque champ, une procédure derrière chaque bouton.

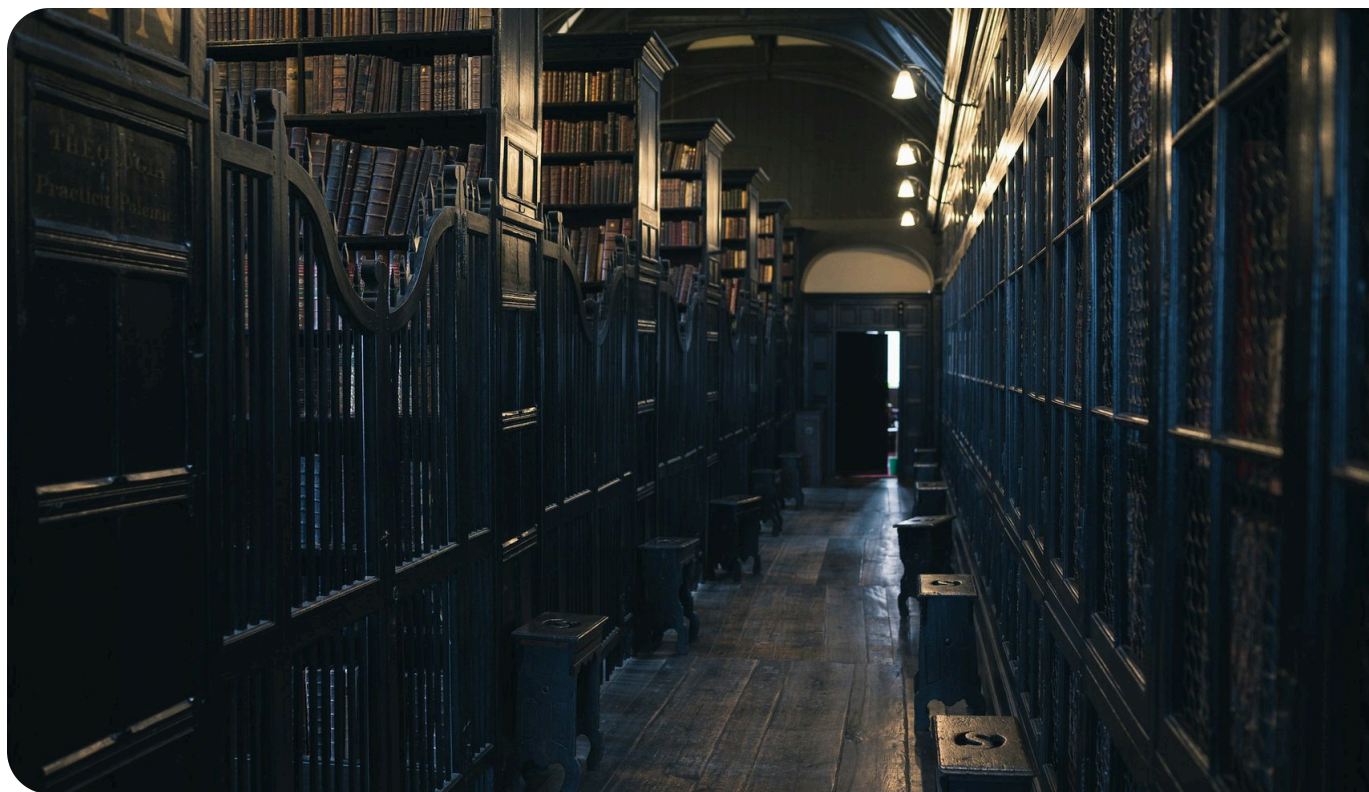
Le bureaucrate IA que décrit Harari n'arrivera donc pas dans un nouveau système. Il arrivera dans le vôtre. Le prêt sur lequel il statue se trouve dans un bloc d'un formulaire. La réclamation qu'il règle, la facture qu'il comptabilise, le congé qu'il approuve sont des enregistrements qui passent aujourd'hui par un déclencheur WHEN-VALIDATE-RECORD avant d'atteindre la base de données.

Pas de règles, pas d'agent

Gartner a aussi mesuré comment ces projets échouent. Plus de 40 % des projets d'IA agentique seront annulés d'ici la fin de 2027, pour trois raisons : des coûts qui s'envolent, une valeur d'affaires floue et des contrôles des risques insuffisants. En octobre 2026, ses tendances pour 2027 ont nommé le contrôle manquant : une couche de contexte dont la sémantique est faite des règles et des politiques de l'organisation, sans laquelle les agents ont trop de privilèges et n'inspirent pas confiance. Seulement 22 % des entreprises tirent aujourd'hui une valeur importante de l'IA générative, alors que 42 % prévoient de déployer des agents d'ici la fin de 2026. Ce contrôle est celui qu'un système transactionnel ne peut pas contourner. Aucune organisation ne laissera un agent approuver des paiements, régler des réclamations ou passer des écritures de journal sans garde-fous qu'elle peut auditer.

La conclusion

Un bureaucrate IA n'est accepté que s'il obéit aux règles. Ces règles doivent être du code, appliqué avant la base de données, et le même pour l'agent que pour l'employé. Les organisations qui exploitent Oracle Forms les ont déjà. La seule question est de savoir si la migration les garde.



Où vivent les règles

Demandez à une équipe Oracle Forms où se trouvent les règles d'affaires : elle vous montrera les paquetages de la base de données. La moitié de la réponse est là. L'autre moitié est dans les formulaires : des déclencheurs WHEN-VALIDATE-ITEM qui refusent une date hors de la période ouverte, des déclencheurs PRE-INSERT qui dérivent une valeur que l'utilisateur ne saisit jamais, des déclencheurs KEY-COMMIT qui appellent un paquetage dans un ordre précis, des propriétés qui rendent un élément non modifiable une fois l'enregistrement comptabilisé. Une grande application en contient des milliers, écrits sur vingt ans par des gens aujourd'hui à la retraite. Personne n'en a la liste complète.

Ce qu'une réécriture perd

Toute réécriture, manuelle ou par une IA qui génère du code, part d'une spécification. La spécification est rédigée à partir des écrans et des flux de travail évidents. Les règles sont découvertes plus tard, un incident à la fois : une écriture acceptée dans une période fermée, un rabais appliqué sans l'approbation qu'exigeait l'ancien formulaire, un enregistrement sauvegardé avec un champ que l'ancien déclencheur aurait dérivé.

Une règle perdue, et ce qu'elle coûte avec un agent

Un formulaire porte un déclencheur de validation qui refuse une date de comptabilisation hors de la période comptable ouverte. La réécriture reproduit l'écran, la liste de valeurs et la sauvegarde, mais pas le déclencheur, parce que personne ne l'a demandé. Avec un opérateur humain, l'erreur apparaît à la fin du mois et elle est corrigée. Avec un agent qui passe des centaines d'écritures par jour, la même lacune produit des centaines d'écritures dans une période fermée avant que quiconque y jette un œil. La règle dont l'agent avait le plus besoin est celle que la réécriture a laissé tomber.

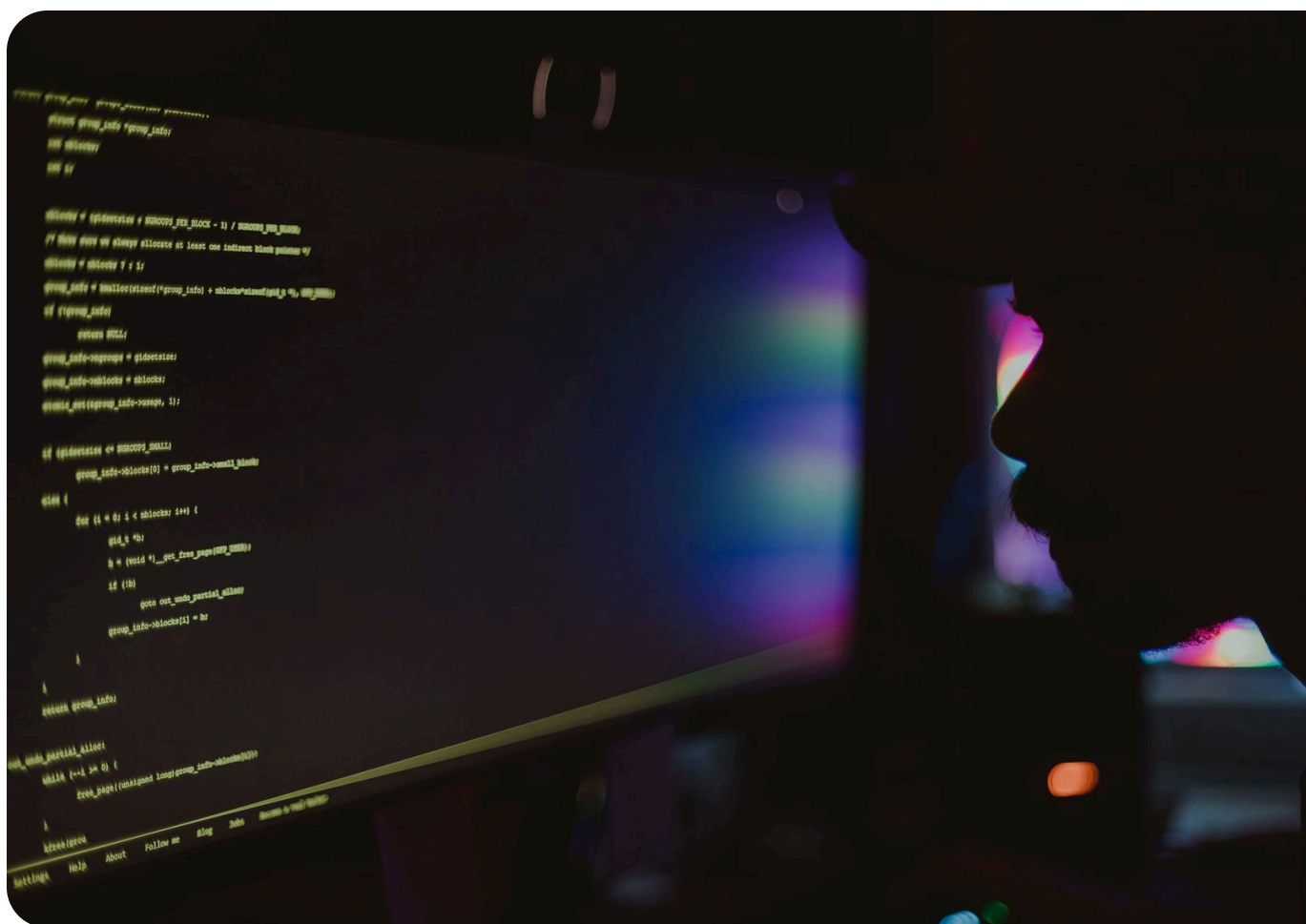
Pourquoi un assistant greffé ne suffit pas

L'autre raccourci consiste à garder les écrans et à ajouter un assistant IA à côté. Il aide l'opérateur à saisir plus vite. Il ne change pas qui fait le travail, et c'est la stratégie dont Gartner prévoit qu'elle comprimera les marges des éditeurs jusqu'à 80 % d'ici 2030, parce que l'acheteur cesse de payer pour de l'assistance et commence à payer pour des résultats.

La réponse d'Oracle : APEX 26.1

Oracle n'est pas resté immobile. APEX 26.1, sorti en mai 2026, livre des agents IA dont les outils sont écrits en SQL ou en PL/SQL, un choix de modèles et une étape d'approbation pour les actions sensibles. Oracle publie aussi un assistant IA qui transforme les modules Forms en artefacts de modernisation APEX : une spécification fonctionnelle, une matrice de traçabilité, un rapport. C'est une option sérieuse, et elle a la même faiblesse que toute réécriture. Les modules Forms sont analysés pour en tirer une spécification, et l'application APEX, agents compris, est bâtie à neuf à partir de celle-ci. Les déclencheurs qui encodaient vingt ans de règles deviennent de la documentation, et les outils de l'agent sont du code nouveau. APEX maintient aussi l'application sur Oracle Database pour de bon. La question à poser à toute voie de migration n'est pas de savoir si elle offre des agents, mais si l'agent fonctionne sous les règles que vous avez déjà.

Une voie sûre vers les agents doit faire deux choses à la fois : garder chaque règle que contiennent les formulaires, sous forme de code exécutable, et laisser un agent passer par ces règles comme le fait un employé. C'est ce que décrit le chapitre suivant.



Migrer les règles, pas les écrans

05

ORMIT™-OpenJava ne réinterprète pas un formulaire. Il le migre : chaque déclencheur, validation et procédure devient du TypeScript qu'OFCS, le service de compatibilité Oracle Forms, exécute exactement là où Oracle Forms l'exécutait.

Ce qu'est OFCS

OFCS (Oracle Forms Compatibility Service), le service de compatibilité Oracle Forms, est l'environnement d'exécution au cœur d'ORMIT™-OpenJava. Il exécute les règles d'un formulaire migré, chaque déclencheur, validation et procédure, en TypeScript, là où Oracle Forms les exécutait : dans le navigateur pendant que l'utilisateur travaille, et dans un serveur Node.js sans écran. Il est livré au client avec son code source. Tout ce que décrit ce livre blanc repose sur lui.

93

Types de déclencheurs Oracle Forms reproduits par OFCS, déclenchés au moment où Forms les déclenche

232

Fonctions intégrées d'Oracle Forms et fonctions SQL offertes au code migré

2×

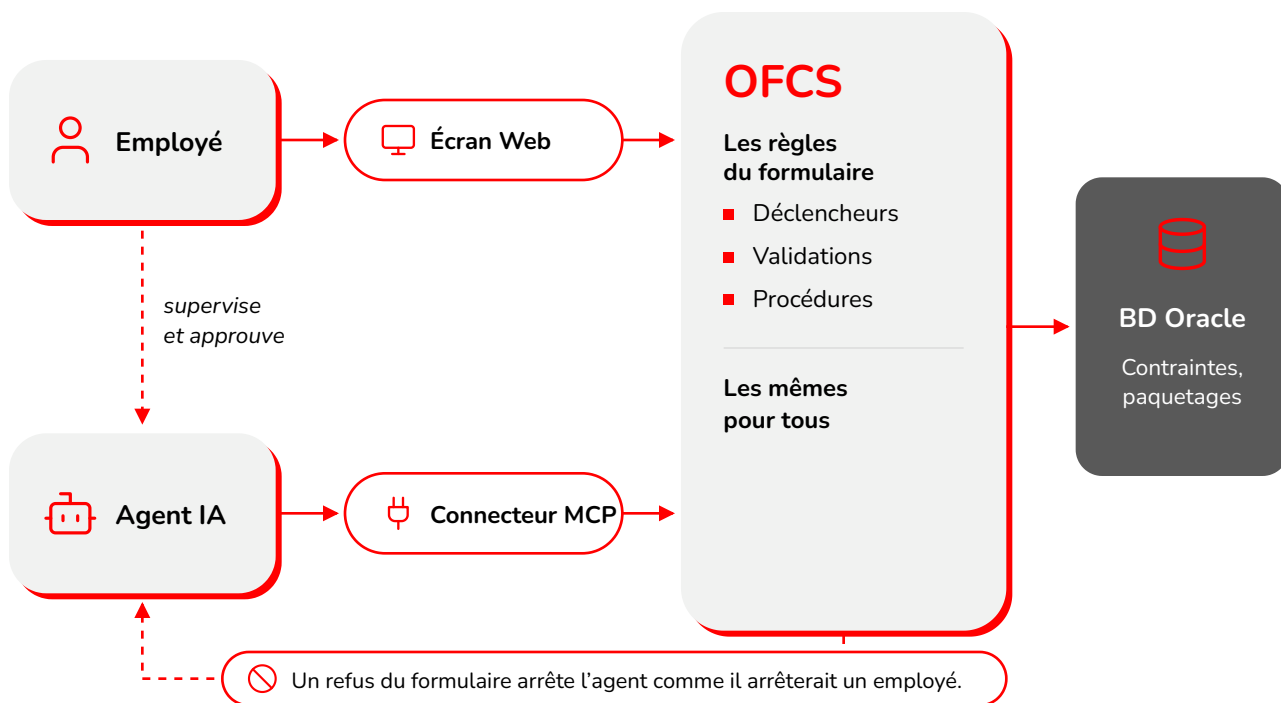
Chaque sauvegarde est validée deux fois : dans le navigateur pendant que l'utilisateur travaille, puis sur le serveur avant la base de données

Des règles qui s'exécutent déjà sans écran

La seconde validation est le détail qui compte pour les agents. Depuis 2026, chaque application migrée est livrée avec un serveur de validation : un processus Node.js qui exécute le même OFCS, sans écran. Avant qu'une insertion ou une mise à jour n'atteigne Oracle, ce serveur charge le formulaire, rejoue ses déclencheurs de validation et de commit sur l'enregistrement, et refuse la sauvegarde si le formulaire l'aurait refusée. Contourner l'écran ne permet pas de contourner les règles.

Du point de vue du formulaire, un agent IA est un appelant de plus, sans écran. Par le connecteur, il lit les blocs et les éléments du formulaire, saisit des valeurs dans l'ordre de navigation pour que la validation et les effets de chaque élément s'exécutent, appuie sur ses boutons, puis valide la transaction. Chacune de ces actions passe par OFCS. Quand un déclencheur lève une erreur, l'agent reçoit le message du formulaire lui-même et s'arrête, comme un employé s'arrêterait devant le même message.

L'agent et l'employé passent par les mêmes règles.



Deux façons d'opérer un formulaire, une seule source de règles. Dans cette conception, l'agent n'a aucun chemin à lui vers la base de données : il passe par le connecteur, sous une identité propre, puis par les mêmes règles que l'écran, avant que quoi que ce soit n'atteigne la base de données.

Ce que vous obtenez aujourd'hui, et ce que cela permet demain

Aujourd'hui, ORMIT™-OpenJava livre une application React ou Angular que vos utilisateurs exploitent comme ils exploitaient les formulaires, avec les mêmes touches, listes de valeurs, validations et messages. Cette application est le produit. Ce que la migration permet ensuite, et ce que RENAPS offre, c'est une application que n'importe quel agent peut exploiter : un connecteur ouvert devant le même OFCS et les mêmes règles, livré avec la migration ou ajouté à celle-ci. L'agent qui prend un processus en charge vient après, processus par processus, quand l'organisation est prête, et il peut être celui du client, celui d'un éditeur de plateforme ou celui de RENAPS. Les clients qui migrent maintenant n'achètent pas une solution provisoire. Ils achètent le seul type de migration auquel on peut ajouter un agent sans tout recommencer.

Ce qui reste dans la base de données y reste

Migrer les règles ne veut pas dire les déplacer. La logique des paquetages, les contraintes et les privilèges restent dans Oracle, intacts, et chaque écriture s'exécute dans la base de données sous l'identité Oracle de son appelant : celle de l'employé depuis un écran aujourd'hui, et celle de l'agent une fois le connecteur en place. Les rôles et les privilèges qu'applique déjà la base de données continuent de fonctionner. Les règles du formulaire, qu'Oracle Forms exécutait dans son propre environnement d'exécution, s'exécutent maintenant dans OFCS. Rien n'est relégué à une consigne donnée à l'IA.

Ce que cela rend possible

Pour la migration

- Le comportement du formulaire d'origine est reproduit, bogues compris, et prouvé par des tests de bout en bout rédigés à partir du PL/SQL d'origine.
- Les mêmes règles servent une interface React ou Angular, et la même suite de tests s'exécute sur les deux.
- L'application fonctionne sur la base de données Oracle existante du client.

Pour l'agent

- Les métadonnées du formulaire décrivent chaque bloc, élément, liste de valeurs et bouton : ce dont un agent a besoin pour agir.
- Le serveur de validation répond déjà aux appelants sans écran.
- Les tests de bout en bout qui prouvent la migration deviennent le moyen d'évaluer l'agent.

La différence, en une ligne

Tout le monde peut placer une IA devant un système transactionnel. L'y placer en toute sécurité exige des règles qui sont du code, appliquées avant la base de données, et les mêmes pour l'agent que pour l'employé. OFCS, au cœur d'ORMIT™-OpenJava, garde les règles que vous avez déjà exactement sous cette forme, et les exécute pour l'agent comme il les exécute pour l'employé.

La couche de contexte que demande Gartner, et où un formulaire migré la contient déjà

La plateforme de contexte d'IA de Gartner, quatrième tendance pour 2027, « unifie les connaissances dispersées de l'entreprise en une couche de vérité lisible par machine pour ancrer les travailleurs numériques autonomes ». Ses trois composantes correspondent à ce que préserve ORMIT™-OpenJava, et à ce qu'ajoute l'offre d'agents.

Composante selon Gartner	Ce qu'elle contient, selon Gartner	Dans une application migrée
Sémantique	Ontologie, indicateurs, règles, responsabilités, politiques	Les règles : chaque déclencheur, validation et procédure du formulaire, en TypeScript qu'exécute OFCS. Pas une description des règles. Les règles.
État opérationnel	Entités, conditions de l'environnement, données au bon moment	Les blocs, éléments et listes de valeurs du formulaire, et les enregistrements vivants de la base de données Oracle du client, lus par les requêtes qu'exécute le formulaire lui-même.
Provenance	Traçabilité des données, des processus, des décisions, des actions et des résultats	Le journal des actions qu'ajoute le volet « prêt pour les agents » : pour chaque action, la règle par laquelle elle est passée, ce que l'agent a proposé et la personne qui l'a approuvée.

Pour bâtir cette couche, Gartner recommande de « séparer les règles d'affaires passives des traces de décision actives, générées par les machines ». Cette séparation est la figure centrale de ce livre blanc. Les règles sont du code qui ne change pas quand l'agent s'exécute; le raisonnement de l'agent est une trace conservée à côté, et il ne devient jamais une règle.

Ce qu'un formulaire migré ne fournit pas, c'est le reste de la couche : l'ontologie qu'attend Gartner, dans des normes comme RDF et OWL, et les connaissances qu'une organisation conserve hors de ses formulaires. OFCS alimente une plateforme de contexte; il ne la remplace pas. Il fournit la composante qui n'existe aujourd'hui que dans vos formulaires, les règles, sous la seule forme à laquelle on peut astreindre un agent : exécutable.

La première tendance de Gartner pour 2027, les grands modèles quantitatifs, règle une autre question que posent les clients. Elle part du constat que les modèles de langage « échouent au raisonnement mathématique parce qu'ils devinent de manière probabiliste ». Dans une application migrée, l'agent ne calcule jamais un montant. Les procédures du formulaire le calculent, comme le faisait l'environnement d'exécution d'Oracle Forms, et l'agent lit le résultat.

L'offre : une application que tout agent peut exploiter

06

Une migration vers les agents ne remplace pas un écran par une fenêtre de conversation. Elle change ce que l'organisation achète, en deux volets : d'abord une application que n'importe quel agent peut exploiter sous ses règles, puis des processus exécutés par un agent, ses équipes demeurant responsables du résultat.

Deux volets, une seule source de règles

Volet	Ce que l'organisation obtient	L'agent de qui	Comment c'est prouvé
1. Prêt pour les agents, livré avec la migration ou ajouté à celle-ci	Un connecteur ouvert qui expose le formulaire migré à un agent, ses blocs, éléments, listes de valeurs et boutons, dans l'ordre propre au formulaire : le Model Context Protocol aujourd'hui, et les mêmes opérations en ligne de commande ou par une simple API quand la plateforme du client le préfère. Chaque appel passe par OFCS et par les règles du formulaire. L'agent agit sous une identité propre, avec les privilèges de base de données dont son processus a besoin, et chaque action est consignée avec la règle par laquelle elle est passée.	N'importe lequel : celui de l'organisation, d'un éditeur de plateforme, d'Oracle, de RENAPS.	Testé, comme le reste de la migration : la scène du refus, l'identité, le journal. Aucun modèle n'a besoin d'avoir raison.
2. Processus confiés à un agent, un processus à la fois	Un agent prend en charge le travail d'un processus sous supervision : il propose, un employé approuve, les règles s'appliquent. L'organisation choisit l'agent et le modèle, et fixe le rythme.	L'agent de RENAPS, ou celui choisi au volet 1.	Mesuré : précision sur les cas ambigus, coût et temps par exécution, part des cas qui n'ont plus besoin d'une personne.

Le premier volet s'appuie directement sur la migration. Le second est ce qu'il rend possible, et la suite de ce chapitre le décrit.

Trois façons d'exécuter un processus, une seule source de règles

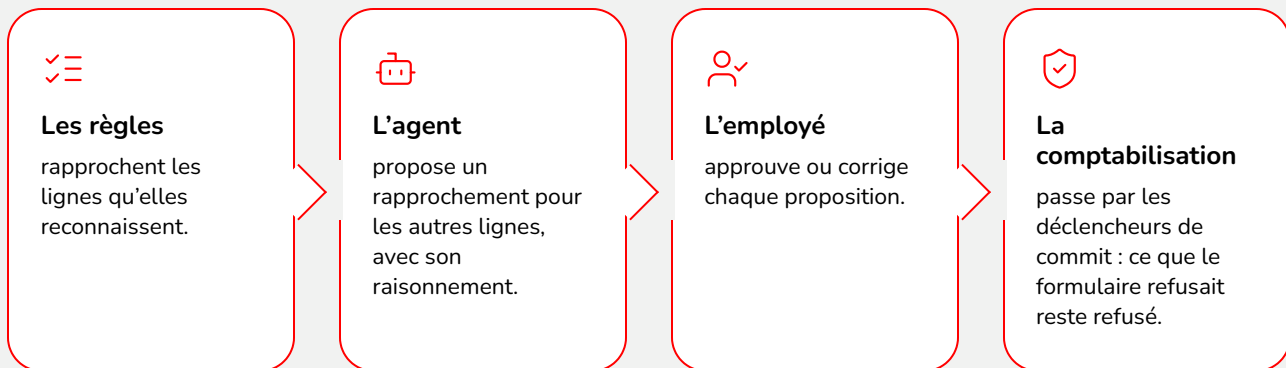
Qui fait le travail	Pour quels cas	Qui approuve
Un programme	Les chemins qui n'exigent aucun jugement : les lignes qu'une règle peut rapprocher seule, les écritures dont chaque champ est connu.	Personne, comme aujourd'hui. Les règles s'appliquent.
Un agent IA	Les cas ambigus : un paiement qui correspond presque, une réclamation à laquelle il manque un document, une demande qui exige une décision dans le cadre des règles.	Un employé, qui voit ce que l'agent propose et pourquoi.
Une personne, à l'écran	Là où un écran sert encore : la saisie au clavier à haut volume, et les exceptions que l'agent ne peut pas régler.	Comme aujourd'hui.

Les trois passent par les mêmes règles OFCS. Faire passer un cas d'une colonne à l'autre change qui agit, jamais ce qui est permis.

Un exemple concret

Prenons le traitement des relevés bancaires, présent dans la plupart des services financiers. Aujourd'hui, un employé ouvre le relevé, rapproche chaque ligne d'un poste ouvert, puis comptabilise. L'application migrée garde les règles du formulaire : quelles lignes peuvent être rapprochées automatiquement, quels comptes une écriture peut toucher, ce qui rend un rapprochement complet.

Avec l'agent en place, les règles rapprochent les lignes qu'elles reconnaissent. L'agent propose un rapprochement pour les autres lignes, avec son raisonnement, et l'employé approuve ou corrige chaque proposition. La comptabilisation passe par les déclencheurs de commit du formulaire : ce que le formulaire refusait reste refusé. La journée de l'employé passe de la saisie de chaque ligne au règlement des exceptions.

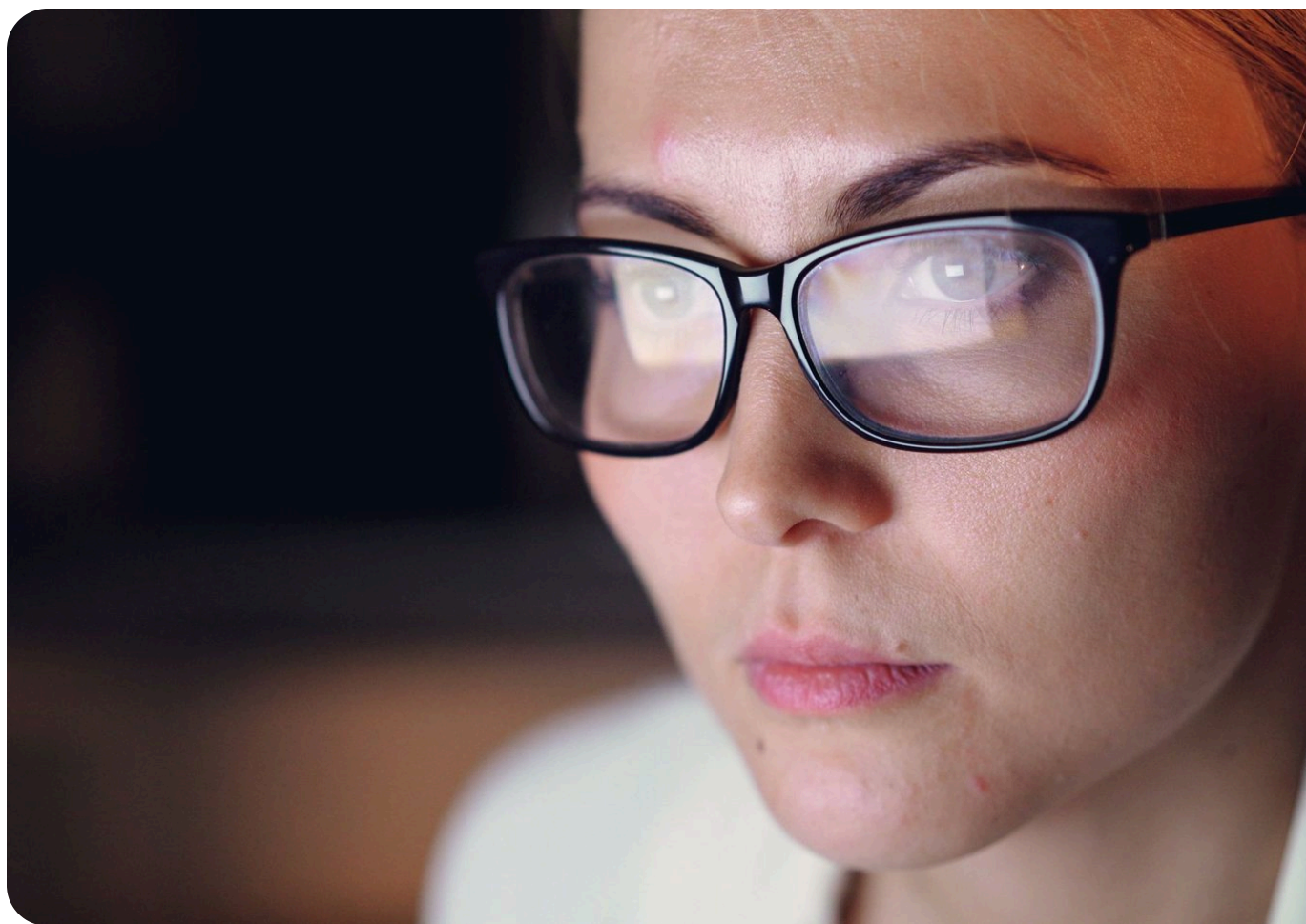


Ce que les règles ne peuvent pas faire

Les règles disent ce que le formulaire refuse. Elles ne disent pas quel rapprochement est le bon quand plusieurs sont valides, et c'est exactement le travail de la colonne de l'agent. Voilà pourquoi l'approbation n'est pas facultative, et pourquoi on mesure un agent à sa précision sur les cas ambigus, pas à son taux de réussite sur les cas faciles. Les règles rendent l'agent sûr à essayer. Les mesures décident s'il vaut la peine de le garder.

Ce que font vos équipes à la place

Personne n'a à faire confiance à l'agent. On demande aux gens de le superviser, avec les mêmes règles qu'ils connaissent, et d'approuver son travail avant qu'il compte. Le savoir institutionnel de l'équipe devient l'ensemble des exceptions que l'agent apprend à signaler, et la mesure du processus devient la part des cas qui n'ont plus besoin d'une personne.



Les contrôles qu'exige un système transactionnel ne sont pas ajoutés à l'agent après coup. Ils découlent de la façon dont la migration est bâtie.

Identité et responsabilité

Par conception, l'agent a une identité propre : son propre compte de base de données, avec seulement les privilèges dont son processus a besoin, et jamais les identifiants d'un employé. Les tendances 2027 de Gartner en font la première règle de l'identité des agents; les rôles et privilèges d'Oracle l'appliquent déjà. L'agent prépare, un employé approuve : la séparation des tâches qu'attendent les auditeurs, conservée telle quelle. Chaque action est journalisée avec la règle par laquelle elle est passée, l'agent qui l'a proposée et la personne qui l'a approuvée.

Une confiance progressive

Un agent commence en observation : il propose, et ses propositions sont comparées au travail des employés. Puis chaque action exige une approbation. Puis seules les exceptions en exigent une. L'organisation fixe le rythme, processus par processus.

Vos données, votre région, votre choix de modèle

Le modèle s'exécute là où l'organisation le choisit. Avec Amazon Bedrock ou Google Cloud Vertex AI, requêtes et résultats restent dans le compte infonuagique et la région du client. Sur Microsoft Foundry, Claude est offert aux États-Unis, l'Europe étant annoncée; le Canada n'y est pas encore couvert. Rien n'est envoyé à un service d'IA grand public. Et rien ne dépend d'un modèle : les règles migrées sont le code du client, le connecteur utilise un protocole ouvert, MCP, aussi offert en ligne de commande ou par API, et le modèle peut être remplacé, y compris par un modèle ouvert exécuté chez le client. OFCS est livré avec son code source, sous une licence qui n'expire pas. Ces options de sortie sont ce que la tendance SovereignTech de Gartner demande aux organisations hors des États-Unis de s'assurer d'ici 2030.

Un coût mesuré

Contrairement à un écran, chaque action d'un agent a un coût. RENAPS ne le chiffre qu'une fois mesuré sur un vrai formulaire, par exécution, et le client décide s'il paie l'IA sur son compte infonuagique ou par l'abonnement.

Le test du refus

Le test décisif d'une migration sûre, c'est ce qui arrive quand l'agent tente ce que le formulaire interdit. La réponse doit être la même que pour une personne : le formulaire refuse, l'agent reçoit le message du formulaire, le signale, et ne le contourne pas.

Ce que ce livre blanc n'affirme pas

La migration, OFCS et son serveur de validation sont livrés avec chaque migration, et le connecteur et l'offre d'agents décrits aux chapitres 6 à 8 sont bâtis sur cette base. Aucune capacité décrite ici ne dépend de la justesse du modèle. Le premier volet, le connecteur, ne dépend d'aucun modèle, et il est testé comme le reste de la migration. Le modèle propose; les règles décident; une personne approuve. Les règles sont nécessaires, mais pas suffisantes : elles arrêtent ce que le formulaire interdit, pas un choix plausible que le formulaire permet. C'est à cela que servent l'étape d'approbation et la précision mesurée. Là où une règle n'existe pas dans le formulaire, l'agent n'a pas plus d'autorité qu'un nouvel employé, et le processus reste confié à une personne jusqu'à ce que l'organisation écrive la règle.



La première étape, c'est la migration elle-même, puis le connecteur, puis un processus, pas un programme. Elle réutilise ce que produit déjà chaque migration RENAPS, part donc du matériel dont dispose le client, et se termine par des chiffres.



Ce qu'apporte RENAPS

- ORMIT™ -OpenJava et OFCS, qui gardent les règles exécutables et servent déjà des appelants sans écran, et le connecteur qui rend l'application migrée exploitable par n'importe quel agent.
- Une chaîne de migration bâtie comme ce que les tendances 2027 de Gartner appellent un harnais d'agent de codage IA : l'IA travaille sur un déclencheur ou une unité de programme à la fois, encadrée par des vérifications automatisées, un audit de chaque changement par rapport à la source Oracle, des tests de bout en bout dérivés du PL/SQL d'origine et dont on a prouvé qu'ils peuvent échouer, et une performance mesurée par rapport à Oracle Forms. Rien n'atteint un réviseur avant que les contrôles soient franchis.
- Un soutien complet de l'application migrée, avant et après l'agent.

Prêt à le voir sur l'un de vos processus?

Réservez une séance de travail avec nos experts : apportez un processus et ses formulaires, et nous vous montrerons comment ses règles deviennent les garde-fous d'un agent, et à quoi ressemble la première étape mesurée pour votre organisation.

www.renaps.com · info@renaps.com · 1-888-987-3627



Sources

Chiffres et citations vérifiés à la source indiquée; citations traduites par RENAPS.

1. The Batch, DeepLearning.AI, 27 février 2026. Claude Cowork plugins trigger a SaaS stock selloff. deeplearning.ai
2. Fortune, 5 février 2026. OpenAI launches Frontier, an AI agent platform for enterprises. fortune.com
3. Forrester, 11 février 2026. SaaS as we know it is dead: how to survive the SaaS-pocalypse. forrester.com
4. Gartner, 26 août 2025. Gartner predicts 40% of enterprise apps will feature task-specific AI agents by 2026, up from less than 5% in 2025. gartner.com
5. Gartner, 1er juillet 2026. Gartner says US\$234 billion in enterprise application software spend is at risk from agentic AI. gartner.com
6. Gartner, 2 avril 2026. Gartner expects most enterprises to abandon assistive AI for outcome-focused workflow by 2028. gartner.com
7. Gartner, 25 juin 2025. Gartner predicts over 40% of agentic AI projects will be canceled by end of 2027. gartner.com
8. Gartner, 20 octobre 2026. Gartner identifies the Top Strategic Technology Trends for 2027, communiqué de presse, et le livre numérique Top 10 Strategic Technology Trends for 2027, Gene Alvarez. gartner.com; gartner.com/en/articles/top-technology-trends-2027
9. Marc Benioff, discours d'ouverture de Dreamforce 2026, septembre 2026. Transcription. singjupost.com
10. Yuval Noah Harari, Tanner Lecture on Human Values, Université d'Oxford, 2026 : « AI has hacked the code of human civilization ». Transcription. singjupost.com
11. CNBC, 3 septembre 2026. Software stocks are up nearly 40% since the “SaaS-pocalypse” bottom. cnbc.com; 24/7 Wall St., 13 octobre 2026, l'IGV en hausse de 41 % du 13 avril au 12 octobre. 247wallst.com
12. Oracle, mai 2026. Announcing Oracle APEX 26.1 General Availability. blogs.oracle.com; Oracle, 1er octobre 2026, the Modernization Forms to APEX Skill. blogs.oracle.com
13. Anthropic, Regional compliance, lieux de traitement des données de Claude par fournisseur. claude.com
14. Oracle, Lifetime Support Policy for Oracle Fusion Middleware, dates de support d'Oracle Forms 12c et 14.1.2. oracle.com

Oracle, Java, Oracle Forms et APEX sont des marques déposées d'Oracle ou de ses sociétés affiliées. Salesforce et Dreamforce sont des marques de Salesforce, Inc. Claude est une marque d'Anthropic, PBC. Gartner est une marque déposée de Gartner, Inc. ORMIT™ est une marque de commerce de RENAPS. Les autres noms cités, dont Amazon Bedrock, Google Cloud Vertex AI et Microsoft Foundry, sont des marques de leurs propriétaires respectifs.

Contactez-nous dès
aujourd'hui.

 www.renaps.com

 info@renaps.com

 1-888-987-3627