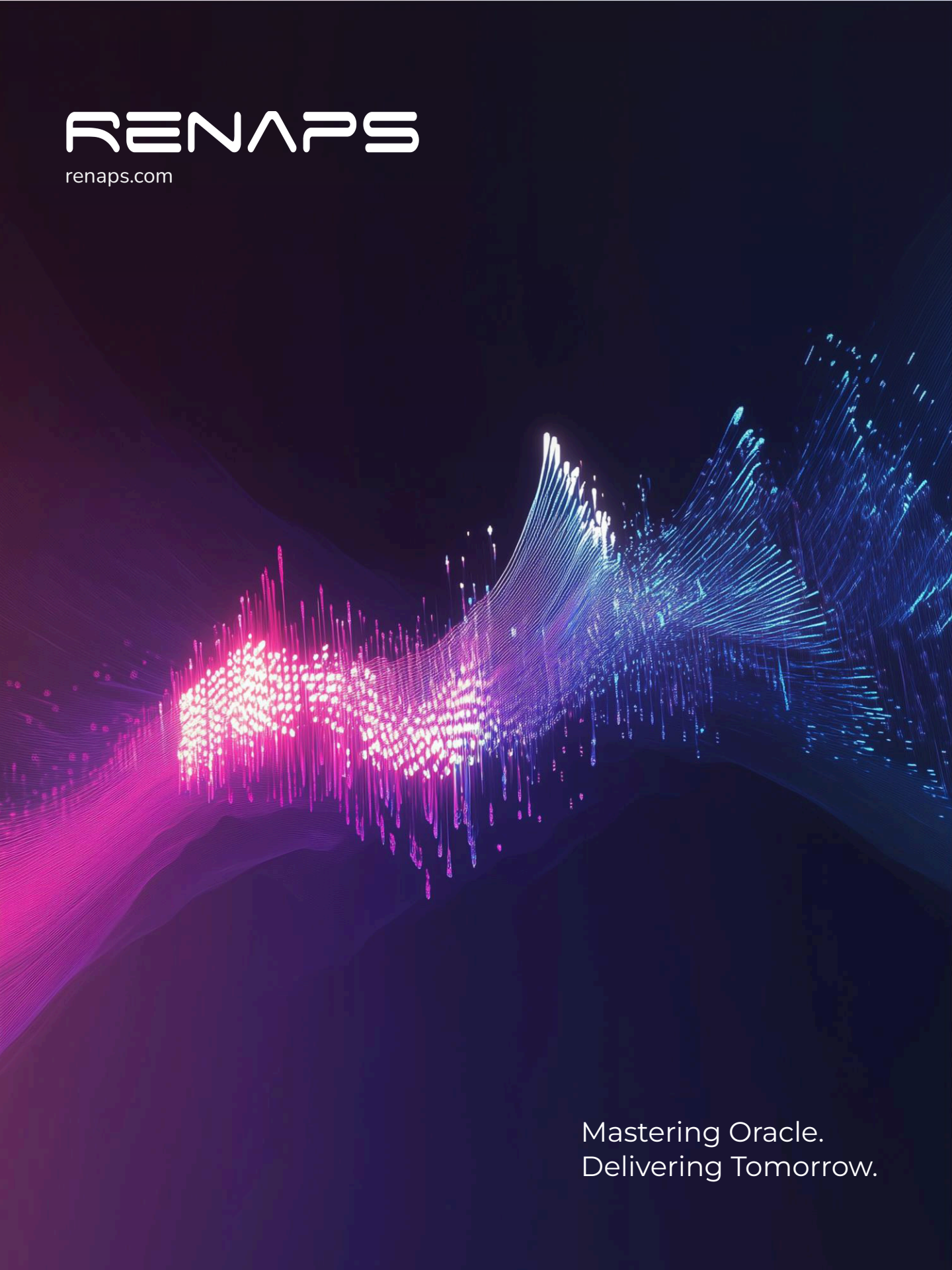


RENAPS

renaps.com



Mastering Oracle.
Delivering Tomorrow.



From Oracle Forms to AI Agents. Migrate the Rules, Not the Screens.

Oracle Forms in the agent era:

Why a migration must keep your business rules executable, so that the web application you deploy today is the one an AI agent can run safely tomorrow, without a second migration.

October 2026 · RENAPS

By

RENAPS

USA & International HQ: 2875 NE 191 Street, Suite 500, Aventura, Florida, 33180, USA

Canadian HQ: 101 Marcel-Laurin, Suite 300, Montréal, Québec H4N 2M3, Canada

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

Table of Contents

1.	EXECUTIVE SUMMARY	05
2.	2026: THE YEAR THE MARKET REPRICED SOFTWARE	06
	The SaaSocalypse, and the rebound	06
	What the analysts measure	06
	Gartner's trends for 2027: from adoption to governance	07
	Even the vendors say so	09
	Why screens are no longer the whole destination	09
3.	TOMORROW'S AI IS A BUREAUCRAT	10
	An agent, not a tool	10
	The bureaucracies that run on Oracle Forms	10
	No rules, no agent	11
4.	THE REWRITE TRAP	12
	Where the rules live	12
	What a rewrite loses	12
	Why a bolted-on assistant is not enough	12
	Oracle's own answer: APEX 26.1	13
5.	MIGRATE THE RULES, NOT THE SCREENS	14
	Rules that already run without a screen	14
	What you get today, and what it enables tomorrow	15
	What stays in the database stays in the database	16
	What this makes possible	16
	For the migration	16
	For the agent	16
	The context layer Gartner asks for, and where a migrated form already holds it	17

Table of Contents

6.	THE OFFER: AN APPLICATION ANY AGENT CAN OPERATE	18
	Two tiers, one source of rules	18
	Three ways to run a process, one source of rules	19
	A worked example	19
	What the rules cannot do	20
	What your people do instead	20
7.	SAFE BY CONSTRUCTION	21
	Identity and accountability	21
	Progressive trust	21
	Your data, your region, your choice of model	21
	A measured cost	21
	The refusal test	21
8.	THE FIRST STEP	23
	What RENAPS brings	24
	SOURCES	25

For twenty years, the question for Oracle Forms owners was *which web stack* to migrate to. In 2026 the question changed. The software market, the analysts and the largest vendors now agree on where enterprise applications are going: the work moves to AI agents, and the screen becomes the exception.

That is good news for organizations that run on Oracle Forms, because of what their forms contain. Behind every screen sit hundreds of triggers, validations and procedures: the rules that decide what a valid transaction is, who may change what, and when a posting is refused. Those rules are exactly what an AI agent needs in order to be trusted in a transactional system, and exactly what a rewrite, by hand or with AI, loses along the way.

RENAPS migrates the rules with the screens. OFCS, the Oracle Forms Compatibility Service at the heart of ORMIT™-OpenJava, keeps every trigger, validation and procedure as executable code behind the React or Angular application it delivers, and that code already runs without a screen. The migration you deploy today is therefore ready for agents, and RENAPS builds on it in two tiers. First, the migrated application becomes operable by any agent, your own, a platform vendor's or RENAPS's: an open connector (MCP, or the same operations as a command line or an API), an identity of the agent's own and a record of every action, in front of the same rules an employee goes through, where a refusal from the form stops the agent the way it would stop a person. Then, process by process, an agent takes over the work, and your employees move from data entry to supervision, on the application you already have.

This white paper shows

- Why 2026 ended the idea that a modern web application is the final destination, with the figures from the market, Forrester and Gartner, including Gartner's technology trends for 2027, and why it remains the necessary first step.
- What tomorrow's enterprise AI looks like: an agent working inside the bureaucracies of finance, insurance, healthcare and government, the systems that run on Oracle Forms today.
- Why the rules in your forms are the control that most agentic AI projects lack, and why rewriting them throws that control away.
- How a migrated application can become operable by any agent, safely: an open connector, an identity of the agent's own and a record of every action, in front of the same rules that serve your users today. Then how a process is handed to an agent, with your people approving its work.
- A first step small enough to decide this quarter: migrate, then add one agent to one process, with measured results.

2026: The Year the Market Repriced Software

02

The SaaSpocalypse, and the rebound

On January 12, 2026, Anthropic released Claude Cowork, an agent that does professional work with a user's files and business tools. On January 30 it added eleven open-source plugins, one per white-collar function: finance, legal review, sales, customer support and others. On February 5, OpenAI launched Frontier, a platform to build and run agents that operate other enterprise software.

Investors read the signal at once. Between January 12 and February 23, the S&P Software & Services index lost a quarter of its value. The session after the plugin release alone erased more than 285 billion US dollars of market capitalization. A Jefferies strategist gave the episode its name: the SaaSpocalypse.

-25%

S&P Software & Services index, January 12 to February 23, 2026

\$285B

Market value erased in one session after the agent plugins shipped

+41%

The same index from its April 13 low to October 12, 2026, back at its highs for the year

The fear was specific. Enterprise software is sold per seat, and an agent that executes a workflow needs no seat. When the logic moves to the agent, the application behind it becomes a database with an interface nobody opens.

Then the market changed its mind, in a way that matters more than the scare. By October 2026 the software index had regained everything it lost: up 41% from its April low, at new highs for the year, flat over twelve months. The companies that led the recovery were not the ones that denied the shift. They were the ones that built agents into their core products. The verdict was not that software is dead. It was that software without agents gets repriced, and software that embeds them keeps its value.

What the analysts measure

Forrester answered the selloff within the week, and did not dispute its direction. Whichever way a company deploys agents, Forrester wrote, "it'll erode the remit of your current enterprise SaaS applications", and seat pricing will give way to consumption and outcome pricing.

Gartner puts dates on it:



By the end of 2026, 40% of enterprise applications will embed task-specific AI agents, up from less than 5% in 2025.



By 2028, a third of user experiences will move from native applications to agent interfaces.



By 2030, 234 billion US dollars of enterprise application spending is at risk from agentic AI, and the user interface stops being a differentiator.



By 2028, more than half of enterprises will stop paying for AI that assists employees, in favour of platforms that deliver the outcome of a process. Vendors that only bolt AI onto existing applications face margin compression of up to 80% by 2030.

Gartner's trends for 2027: from adoption to governance

On October 20, 2026, Gartner published its Top 10 Strategic Technology Trends for 2027. Agents are no longer on the list as a trend. They are the platform the list assumes every organization now runs, and the question has become what governs them. In Gartner's words, the conversation in 2027 "shifts from adoption to governance": traditional IT relied on software scarcity, 2027 brings software abundance, and CIOs "must shift from managing technology assets to governing machine behavior".

+80%

Agentic AI accuracy for organizations that prioritize semantics in AI-ready data, by 2027

25%

Enterprise security breaches that will trace back to AI agent abuse, by 2028

80%

New applications that leading organizations will make disposable and easy to replace, by 2030

Three of the ten trends describe what an Oracle Forms migration has to preserve:



AI context platforms.

The knowledge an agent needs “to interpret inputs, make decisions and act in alignment with business objectives”. Gartner gives the layer three components: semantics, which include the business rules and policies; operational state, the live data; and provenance, the record of decisions, actions and outcomes. Without it, organizations face “overprivileged agents, compliance failures, low trust and excessive token consumption”.



AI agent identity.

Agents “expose massive vulnerabilities in legacy systems that can’t deliver just-in-time access”, and organizations that do not resolve it will see over half of their AI initiatives halt. Gartner’s first instruction: an agent never shares a person’s credentials. It gets an identity of its own, with policy-controlled access.



Disposable applications.

The first action Gartner lists is to “classify which core capabilities require permanent software and which can be served by temporary, on-demand solutions”. The application portfolio becomes “a dynamic system that continuously renews itself as redundant assets are retired”.

Read together, they say where the value of an Oracle Forms application sits. The screens are the replaceable part. The rules are the permanent core, and they are the one component of Gartner’s context layer that no platform vendor can write for an organization, because they exist only in its forms. A migration that keeps them executable, under an identity the database controls, is the part of the 2027 agenda an Oracle Forms owner can deliver from what it already has. Chapter 5 shows the mapping in detail.



Even the vendors say so

In September 2026, Salesforce opened its Dreamforce conference by addressing the selloff head on. Its CEO, Marc Benioff, dismissed the talk of an end of software, then conceded the point that matters:

“SaaS apocalypse was not about the end of software but it may be about the end of software that makes humans do all the work.”

Marc Benioff, CEO of Salesforce, Dreamforce keynote, September 2026

He said it while launching an agent interface over Salesforce’s own applications, built with Anthropic’s Claude. The vendor that mocked the apocalypse answered it by embedding agents. Software that makes humans do all the work is a precise description of a data-entry screen. It is also a precise description of what a conventional Oracle Forms rewrite delivers: the same screen, in a browser, with the rules behind it lost on the way.

Why screens are no longer the whole destination

None of this means that screens disappear. High-volume keyboard entry stays faster on a screen, and most tasks will keep one for years. Nor does it mean waiting. Oracle’s published support dates give Forms 12c until December 2026 on Premier Support, and Forms 14.1.2 until 2030: the migration has a date either way, and it still has to happen. It means that the migration has to be chosen for what it makes possible afterwards. A migration project runs one to three years; the figures above describe the same horizon. A migration that delivers screens and loses the rules ends where it started. A migration that delivers screens and keeps the rules executable is the platform the agents will run on, without a second migration.

The question for an Oracle Forms owner in 2026

Not only “which web framework?”, but “when an agent runs this process on the migrated application, what stops it from doing the wrong thing?” The next chapter is about who that agent is. The chapter after it is about that question, and why your forms already answer it, provided the migration keeps the answer.

Tomorrow's AI Is a Bureaucrat

03

An agent, not a tool

The historian Yuval Noah Harari has spent two years correcting the picture most people hold of artificial intelligence. In his 2026 Tanner Lecture at Oxford, he put the distinction in one sentence:

"It's not a tool in our hands. It is an agent with its own hands."

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

And he named where that agent will live. Not in the streets, and not in a robot body:

"Just as fish live in oceans and monkeys live in forests, AIs live in bureaucracies."

"In the coming years, millions of AI bureaucrats will increasingly take over the world's bureaucracies. AI bankers will decide whether to give you a loan."

Yuval Noah Harari, same lecture

The bureaucracies that run on Oracle Forms

Harari says this as a warning. For an organization that runs on Oracle Forms, it is also a forecast, and it raises one question: who writes the rules the AI bureaucrat will follow?

A bureaucracy, in Harari's sense, is any system that lets strangers trust each other at scale: a bank, an insurer, a hospital's billing, a ministry's permits, a payroll. These systems are made of records, rules and approvals. For three decades, a large share of them have been built on Oracle Forms, precisely because Forms made rules and approvals easy to express: a trigger on every event, a validation on every field, a procedure behind every button.

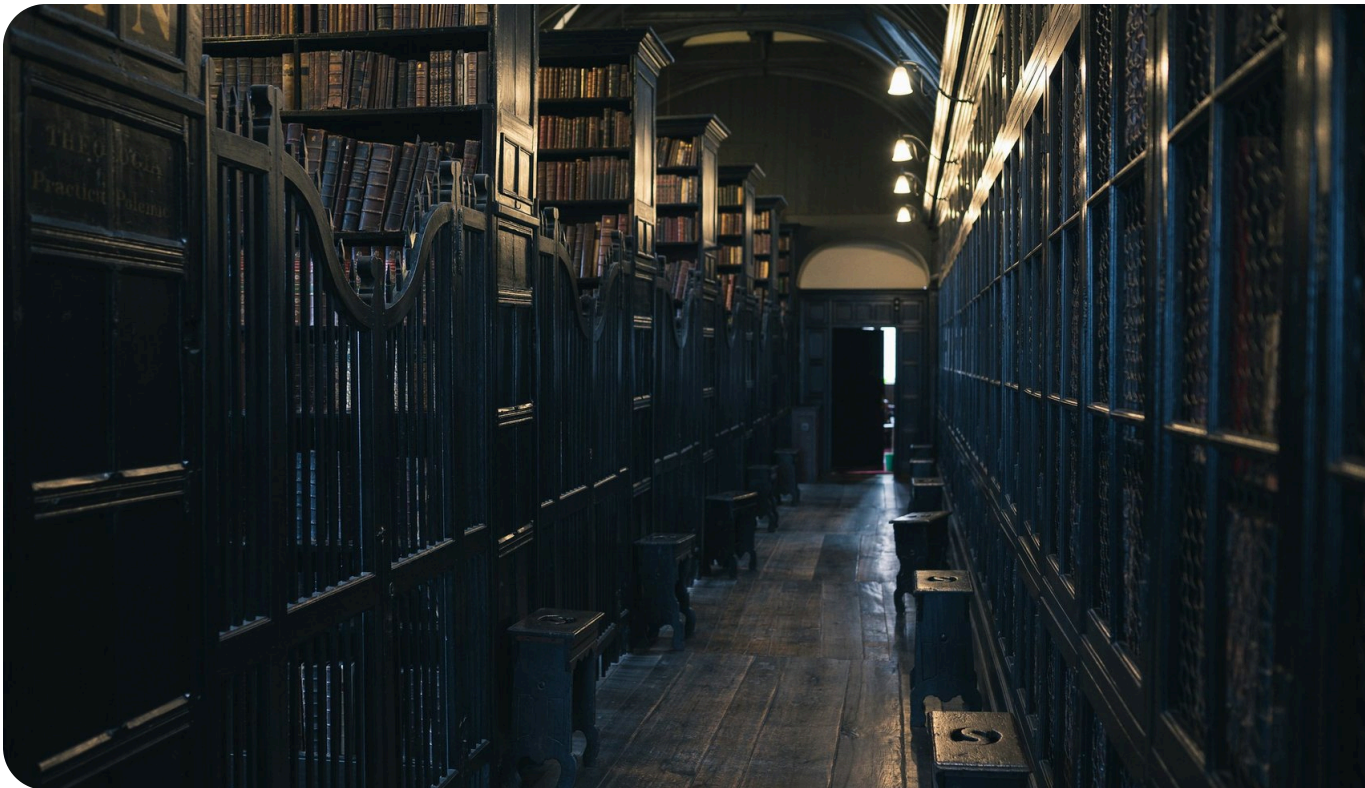
So the AI bureaucrat Harari describes will not arrive in a new system. It will arrive in yours. The loan it decides on is in a block of a form. The claim it settles, the invoice it posts, the leave it approves are records that today pass through a WHEN-VALIDATE-RECORD trigger before they reach the database.

No rules, no agent

Gartner also measured how these projects fail. More than 40% of agentic AI projects will be cancelled by the end of 2027, for three reasons: escalating costs, unclear business value, and inadequate risk controls. In October 2026, its trends for 2027 named the missing control: a context layer whose semantics are the organization's rules and policies, without which agents are overprivileged and untrusted. Only 22% of enterprises capture significant value from generative AI today, while 42% plan to deploy agents by the end of 2026. That control is the one a transactional system cannot work around. No organization will let an agent approve payments, settle claims or post journal entries without guardrails it can audit.

The conclusion

An AI bureaucrat is accepted only if it obeys the rules. Those rules have to be code, enforced before the database, and the same for the agent as for the employee. Oracle Forms owners already have them. The only question is whether the migration keeps them.



Where the rules live

Ask an Oracle Forms team where the business rules are and they will point at the database packages. Half of the answer is there. The other half is in the forms: WHEN-VALIDATE-ITEM triggers that refuse a date outside the open period, PRE-INSERT triggers that derive a value the user never types, KEY-COMMIT triggers that call a package in a precise order, properties that make an item read-only once a record is posted. A large application holds thousands of these, written over twenty years by people who have since retired. Nobody has a complete list.

What a rewrite loses

Every rewrite, whether by hand or with a code-generating AI, starts from a specification. The specification is written from the screens and the obvious workflows. The rules are discovered later, one incident at a time: a posting accepted into a closed period, a discount applied without the approval the old form demanded, a record saved with a field the old trigger would have derived.

One lost rule, and what it costs with an agent

A form carries a validation trigger that refuses a posting date outside the accounting period currently open. The rewrite reproduces the screen, the lookup and the save, but not the trigger, because nobody asked for it. With a human operator, the error surfaces at month end and gets corrected. With an agent executing hundreds of postings a day, the same gap produces hundreds of entries in a closed period before anyone looks. The rule the agent needed most is the one the rewrite dropped.

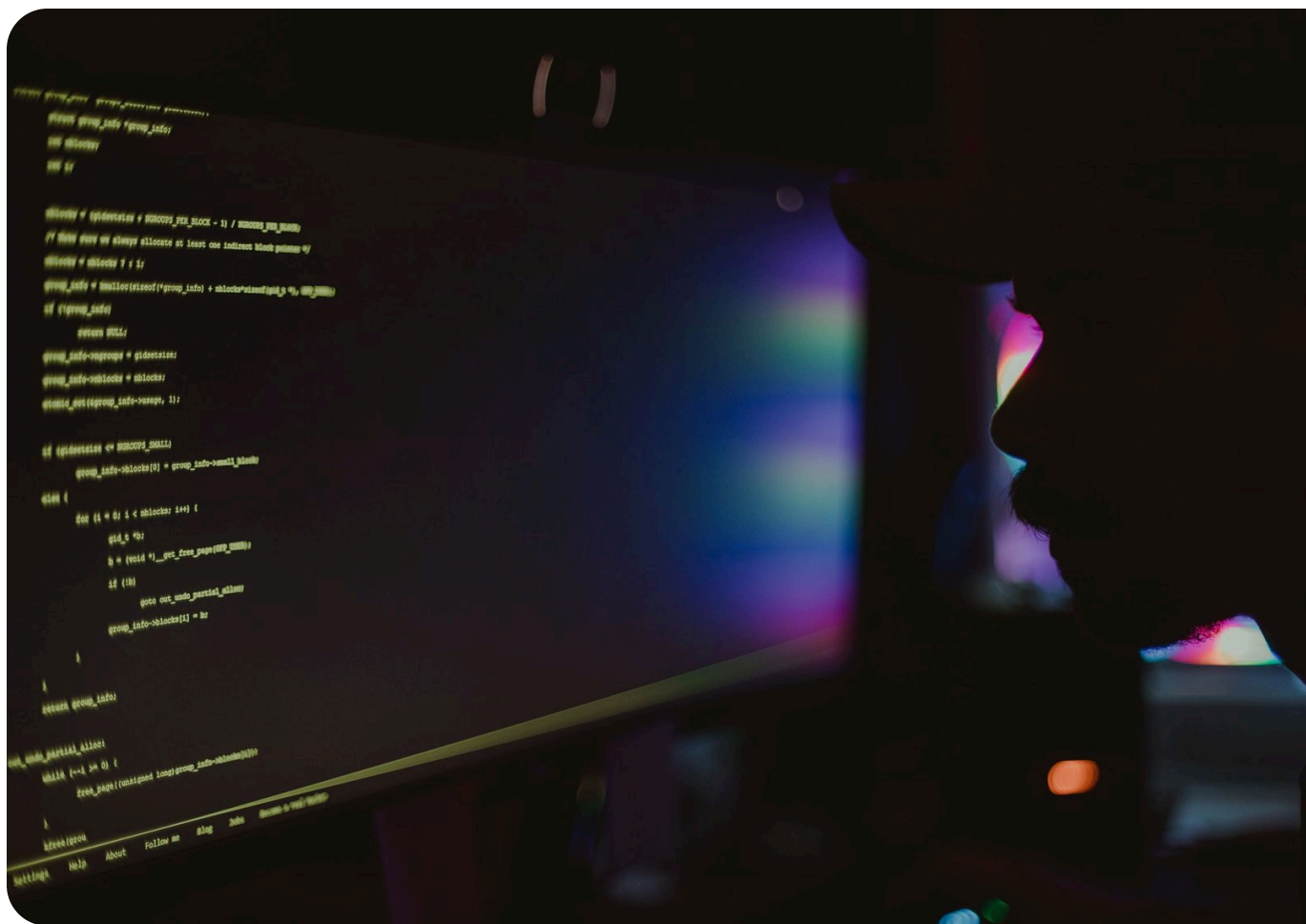
Why a bolted-on assistant is not enough

The other shortcut is to keep the screens and add an AI assistant beside them. It helps the operator type faster. It does not change who does the work, and it is the strategy Gartner expects to compress vendor margins by up to 80% by 2030, because the buyer stops paying for assistance and starts paying for outcomes.

Oracle's own answer: APEX 26.1

Oracle has not stood still. APEX 26.1, released in May 2026, ships AI agents with tools written in SQL or PL/SQL, a choice of models and an approval step for sensitive actions. Oracle also publishes an AI assistant that turns Forms modules into APEX modernization artifacts: a functional specification, a traceability matrix, a report. It is a serious option, and it has the same weakness as every rewrite. The Forms modules are analysed into a specification, and the APEX application, agents included, is built anew from it. The triggers that encoded twenty years of rules become documentation, and the agent's tools are new code. APEX also keeps the application on Oracle Database for good. The question to ask any migration path is not whether it offers agents, but whether the agent runs under the rules you already have.

A safe path to agents has to do two things at once: keep every rule the forms contain, as executable code, and let an agent go through those rules the way an employee does. That is what the next chapter describes.



Migrate the Rules, Not the Screens

05

ORMIT™-OpenJava does not reinterpret a form. It migrates it: every trigger, validation and procedure becomes TypeScript that OFCS, the Oracle Forms Compatibility Service, runs exactly where Oracle Forms ran it.

What OFCS is

OFCS, the Oracle Forms Compatibility Service, is the runtime at the heart of ORMIT™-OpenJava. It executes a migrated form's rules, every trigger, validation and procedure, as TypeScript, where Oracle Forms executed them: in the browser as the user works, and in a Node.js server with no screen attached. It is delivered to the client with its source code. Everything this white paper describes runs on it.

93

Oracle Forms trigger types reproduced by OFCS, fired at the point where Forms fires them

232

Oracle Forms built-ins and SQL functions available to migrated code

2×

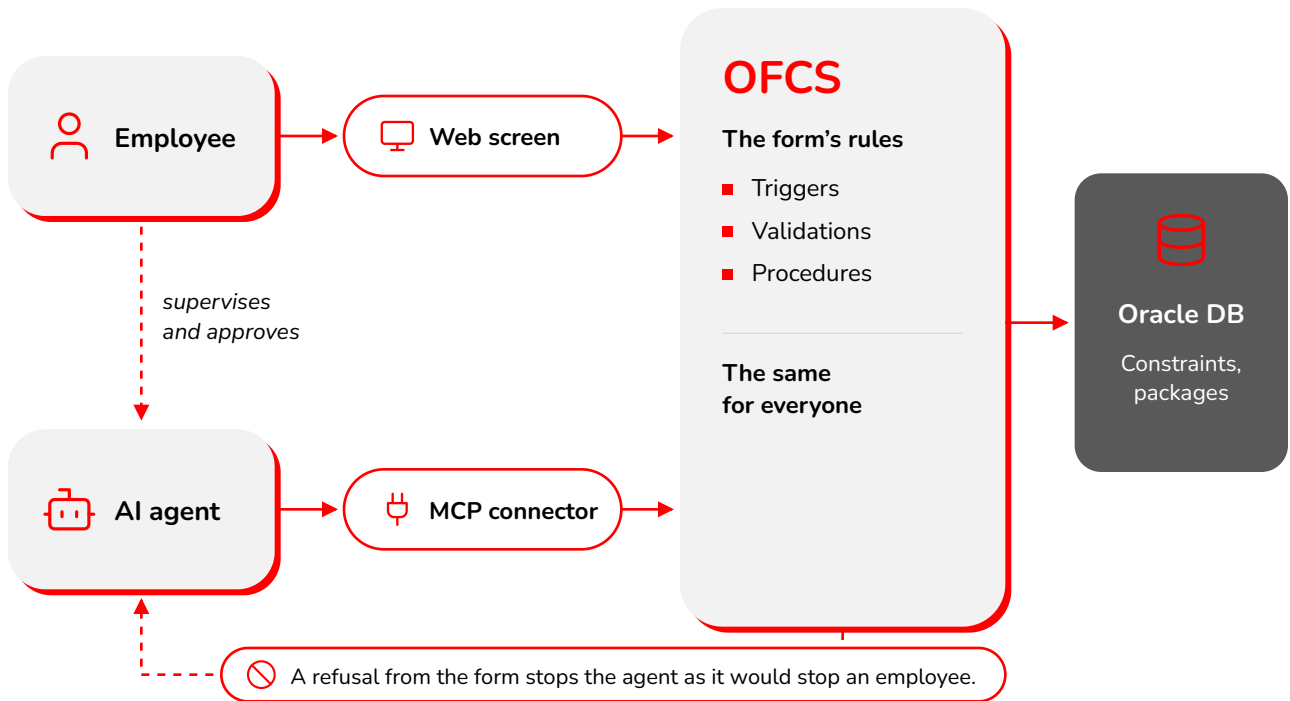
Every save is validated twice: in the browser as the user works, and again on the server before the database

Rules that already run without a screen

The second validation is the detail that matters for agents. Since 2026, every migrated application ships with a validation server: a Node.js process running the same OFCS, with no screen attached. Before any insert or update reaches Oracle, this server loads the form, replays its validation and commit triggers on the record, and refuses the save if the form would have refused it. Going around the screen does not go around the rules.

An AI agent is, from the form's point of view, another caller without a screen. Through the connector, it reads the form's blocks and items, sets values in the form's navigation order so that each item's validation and side effects run, presses the form's buttons, and commits. Every one of those actions goes through OFCS. When a trigger raises an error, the agent receives the form's own message and stops, as an employee would stop at the same message.

The agent and the employee go through the same rules.



Two ways to operate a form, one source of rules. In this design, the agent has no path of its own to the database: it goes through the connector, under an identity of its own, then through the same rules as the screen, before anything reaches the database.

What you get today, and what it enables tomorrow

Today, ORMIT™-OpenJava delivers a React or Angular application that your users operate as they operated the forms, with the same keys, lists of values, validations and messages. That application is the product. What the migration enables next, and what RENAPS offers, is an application any agent can operate: an open connector in front of the same OFCS and the same rules, delivered with the migration or added to it. The agent that takes over a process comes after that, process by process, when the organization is ready, and it can be the client's own, a platform vendor's or RENAPS's. Clients who migrate now are not buying a stopgap. They are buying the only kind of migration an agent can be added to without starting over.

What stays in the database stays in the database

Migrating the rules does not mean moving them. Package logic, constraints and grants remain in Oracle, untouched, and every write runs in the database under its caller's own Oracle identity: the employee's from a screen today, and the agent's own once the connector is in place. The roles and grants the database already enforces keep working. The form's rules, which Oracle Forms ran in its own runtime, now run in OFCS. Nothing is demoted to a prompt.

What this makes possible

For the migration

- The behaviour of the original form is reproduced, bugs included, and proven by end-to-end tests written from the original PL/SQL.
- The same rules serve a React or an Angular front end, and the same test suite runs on both.
- The application runs on the client's existing Oracle database.

For the agent

- The form's metadata describes every block, item, list of values and button: what an agent needs in order to act.
- The validation server already answers callers without a screen.
- The end-to-end tests that prove the migration become the way to grade the agent.

The difference, in one line

Everyone can put an AI in front of a transactional system. Putting one there safely takes rules that are code, enforced before the database, and the same for the agent as for the employee. OFCS, at the heart of ORMIT™-OpenJava, keeps the rules you already have in exactly that form, and runs them for the agent as it runs them for the employee.

The context layer Gartner asks for, and where a migrated form already holds it

Gartner’s AI context platform, trend 4 for 2027, “unifies scattered corporate knowledge into a machine-readable truth layer to ground autonomous digital workers”. Its three components map onto what ORMIT™ - OpenJava preserves, and onto what the agent offer adds.

Gartner’s component	What it holds, in Gartner’s terms	In a migrated application
Semantics	Ontology, metrics, rules, ownership, policies	The rules: every trigger, validation and procedure of the form, as TypeScript that OFCS executes. Not a description of the rules. The rules.
Operational state	Entities, environment conditions, right-time data	The form’s blocks, items and lists of values, and the live records in the client’s Oracle database, read through the queries the form itself runs.
Provenance	Tracking of data, processes, decisions, actions, outcomes	The action log the agent-ready tier adds: for each action, the rule it went through, what the agent proposed, and the person who approved it.

Gartner’s guidance for building that layer is to “separate passive business rules from active, machine-generated decision traces”. That separation is the central figure of this white paper. The rules are code that does not change when the agent runs; the agent’s reasoning is a trace kept beside them, and it never becomes a rule.

What a migrated form does not provide is the rest of the layer: the ontology Gartner expects, in standards such as RDF and OWL, and the knowledge an organization keeps outside its forms. OFCS feeds a context platform; it does not replace one. It supplies the component that exists only in your forms today, the rules, in the one form an agent can be held to: executable.

Gartner’s first trend for 2027, large quantitative models, settles one more question clients ask. It starts from the observation that language models “fail at mathematical reasoning because they guess probabilistically”. In a migrated application the agent never computes an amount. The form’s procedures compute it, as the Oracle Forms runtime did, and the agent reads the result.

The Offer: An Application Any Agent Can Operate

06

A migration to agents does not replace a screen with a chat window. It changes what the organization buys, in two tiers: first an application that any agent can operate under its rules, then processes executed by an agent, with its people in charge of the outcome.

Two tiers, one source of rules

Tier	What the organization gets	Whose agent	How it is proven
1. Agent-ready, delivered with the migration, or added to it	An open connector that exposes the migrated form to an agent, its blocks, items, lists of values and buttons, in the form's own order: the Model Context Protocol today, and the same operations as a command line or a plain API when the client's platform prefers it. Every call goes through OFCS and the form's rules. The agent acts under an identity of its own, with the database grants its process needs, and every action is recorded with the rule it went through.	Any: the organization's own, a platform vendor's, Oracle's, RENAPS's.	Tested, like the rest of the migration: the refusal scene, the identity, the record. No model has to be right.
2. Operated processes, one process at a time	An agent takes over the work of a process under supervision: it proposes, an employee approves, the rules apply. The organization chooses the agent and the model, and sets the pace.	RENAPS's agent, or the one chosen in tier 1.	Measured: precision on ambiguous cases, cost and time per execution, the share of cases that no longer need a person.

Tier one builds directly on the migration. Tier two is what it makes possible, and the rest of this chapter describes it.

Three ways to run a process, one source of rules

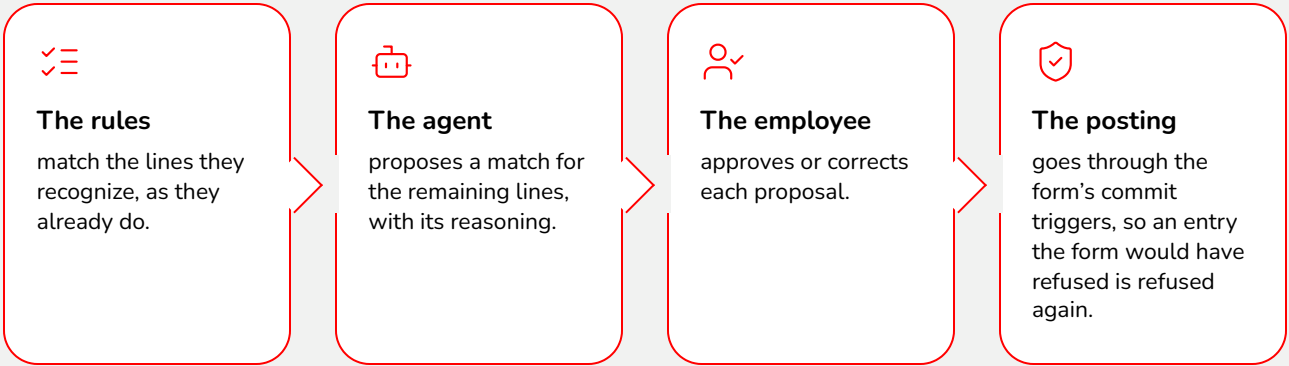
Who does the work	For which cases	Who approves
A program	Paths that require no judgement: the lines a rule can match on its own, the postings whose every field is known.	Nobody, as today. The rules apply.
An AI agent	Ambiguous cases: a payment that almost matches, a claim with a missing document, a request that needs a decision within the rules.	An employee, who sees what the agent proposes and why.
A person, on a screen	Where a screen still serves: high-volume keyboard entry, and the exceptions the agent cannot settle.	As today.

All three go through the same OFCS rules. Moving a case from one column to the next changes who acts, never what is allowed.

A worked example

Take the processing of bank statements, a form found in most finance departments. Today an employee opens the statement, matches each line to an open item, and posts. The migrated application keeps the form's rules: which lines may be matched automatically, which accounts a posting may touch, what makes a reconciliation complete.

With the agent in place, the rules match the lines they recognize, as they already do. The agent proposes a match for the remaining lines, with its reasoning, and the employee approves or corrects each proposal. The posting itself goes through the form's commit triggers, so an entry the form would have refused is refused again. The employee's day moves from typing every line to settling the exceptions.



What the rules cannot do

The rules say what the form refuses. They do not say which match is the right one when several are valid, and that is exactly the work in the agent's column. This is why approval is not optional, and why the measure of an agent is its precision on ambiguous cases, not its pass rate on the easy ones. The rules make the agent safe to try. The measurements decide whether it is worth keeping.

What your people do instead

Nobody is asked to trust the agent. They are asked to supervise it, with the same rules they know, and to approve its work before it counts. The institutional knowledge in the team becomes the set of exceptions the agent learns to raise, and the measure of the process becomes the share of cases that no longer need a person.



The controls that a transactional system requires are not added to the agent afterwards. They come from how the migration is built.

Identity and accountability

By design, the agent has an identity of its own: its own database account, with only the grants its process needs, and never an employee's credentials. Gartner's 2027 trends make this the first rule of AI agent identity; Oracle's roles and grants already enforce it. The agent prepares, an employee approves: the separation of duties the auditors expect, kept as it is. Every action is logged with the rule it went through, the agent that proposed it and the person who approved it.

Progressive trust

An agent starts in observation: it proposes, and its proposals are compared with the employees' own work. Then each action needs approval. Then only the exceptions do. The organization decides the pace, process by process.

Your data, your region, your choice of model

The model runs where the organization chooses. Through Amazon Bedrock or Google Cloud Vertex AI, prompts and results stay in the client's cloud account and region. On Microsoft Foundry, Claude is available in the United States, with Europe announced; Canada is not yet covered there. Nothing is sent to a consumer AI service. And nothing depends on one model: the migrated rules are the client's code, the connector uses an open protocol, MCP, with the same operations available as a command line or a plain API, and the model can be replaced, including by an open model run on the client's own infrastructure. OFCS itself is delivered with its source code, under a licence that does not expire. Those exit options are what Gartner's SovereignTech trend asks every organization outside the United States to secure by 2030.

A measured cost

Unlike a screen, each action an agent takes has a cost. RENAPS quotes it only once measured on a real form, per execution, and the client decides whether it pays the AI on its own cloud account or through the subscription.

The refusal test

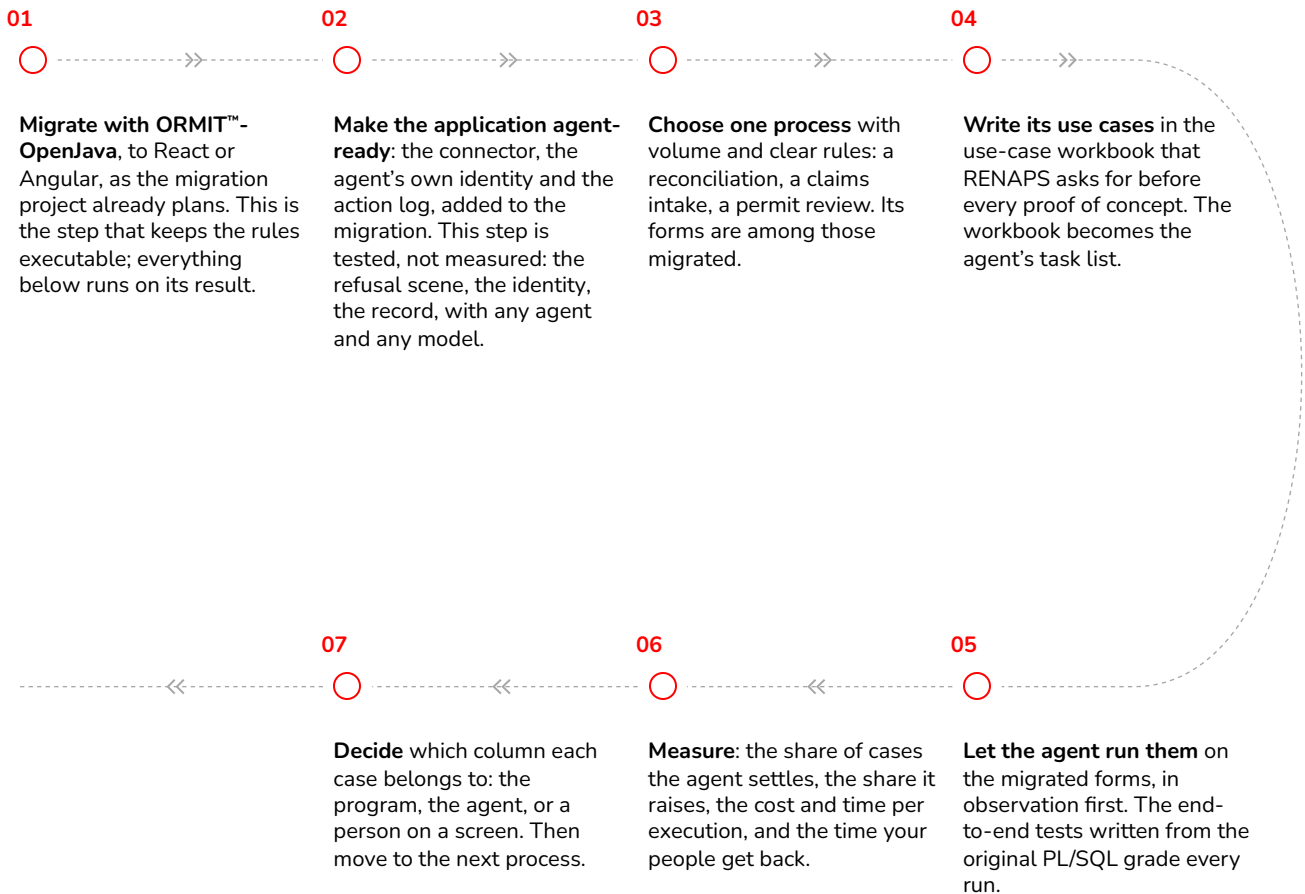
The decisive test of a safe migration is what happens when the agent tries something the form forbids. The answer has to be the same as for a person: the form refuses, the agent receives the form's message, reports it, and does not work around it.

What this white paper does not claim

The migration, OFCS and its validation server ship with every migration, and the connector and the agent offer described in chapters 6 to 8 are built on that foundation. No capability described here depends on the model being right. The first tier, the connector, depends on no model at all, and is tested like the rest of the migration. The model proposes; the rules decide; a person approves. The rules are necessary, not sufficient: they stop what the form forbids, not a plausible choice the form allows. That is what the approval step and the measured precision are for. Where a rule does not exist in the form, the agent has no more authority than a new employee would, and the process stays with a person until the organization writes the rule.



The first step is the migration itself, then the connector, then one process, not a programme. It reuses what every RENAPS migration already produces, so it starts from the material the client has, and it ends with numbers.



What RENAPS brings

- ORMIT™ -OpenJava and OFCS, which keep the rules executable and already serve callers without a screen, and the connector that makes the migrated application operable by any agent.
- A migration pipeline built as what Gartner's 2027 trends call an AI coding agent harness: the AI works on one trigger or program unit at a time, inside automated checks, an audit of every change against the Oracle source, end-to-end tests derived from the original PL/SQL and proven able to fail, and performance measured against Oracle Forms. Nothing reaches a reviewer before the gates pass.
- Full-stack support for the migrated application, before and after the agent.

Ready to see it on one of your processes?

Book a working session with our experts: bring one process and its forms, and we will show you how its rules become the guardrails of an agent, and what the first measured step looks like for your organization.

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



Sources

All figures and quotations were checked at the source listed. Translations, where any, are RENAPS's.

1. The Batch, DeepLearning.AI, February 27, 2026. Claude Cowork plugins trigger a SaaS stock selloff. deeplearning.ai
2. Fortune, February 5, 2026. OpenAI launches Frontier, an AI agent platform for enterprises. fortune.com
3. Forrester, February 11, 2026. SaaS as we know it is dead: how to survive the SaaS-pocalypse. forrester.com
4. Gartner, August 26, 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, July 1, 2026. Gartner says US\$234 billion in enterprise application software spend is at risk from agentic AI. gartner.com
6. Gartner, April 2, 2026. Gartner expects most enterprises to abandon assistive AI for outcome-focused workflow by 2028. gartner.com
7. Gartner, June 25, 2025. Gartner predicts over 40% of agentic AI projects will be canceled by end of 2027. gartner.com
8. Gartner, October 20, 2026. Gartner identifies the Top Strategic Technology Trends for 2027, press release, and the eBook Top 10 Strategic Technology Trends for 2027, Gene Alvarez. gartner.com; gartner.com/en/articles/top-technology-trends-2027
9. Marc Benioff, Dreamforce 2026 keynote, September 2026. Transcript. singjupost.com
10. Yuval Noah Harari, Tanner Lecture on Human Values, University of Oxford, 2026: "AI has hacked the code of human civilization". Transcript. singjupost.com
11. CNBC, September 3, 2026. Software stocks are up nearly 40% since the "SaaSocalypse" bottom. cnbc.com; 24/7 Wall St., October 13, 2026, IGV up 41% from April 13 to October 12. 247wallst.com
12. Oracle, May 2026. Announcing Oracle APEX 26.1 General Availability. blogs.oracle.com; Oracle, October 1, 2026, the Modernization Forms to APEX Skill. blogs.oracle.com
13. Anthropic, regional compliance: where Claude stores and processes data by cloud provider. claude.com
14. Oracle, Lifetime Support Policy for Oracle Fusion Middleware, support dates of Oracle Forms 12c and 14.1.2. oracle.com

Oracle, Java, Oracle Forms and APEX are registered trademarks of Oracle and/or its affiliates. Salesforce and Dreamforce are trademarks of Salesforce, Inc. Claude is a trademark of Anthropic, PBC. Gartner is a registered trademark of Gartner, Inc. Amazon Bedrock, Google Cloud Vertex AI and Microsoft Foundry are trademarks of their respective owners. ORMIT™ is a trademark of RENAPS. All other names are trademarks of their respective owners.

Contact Us Today.

 www.renaps.com

 info@renaps.com

 1-888-987-3627