

13 FÉVRIER 2025 GUIDES

# Les six niveaux du comportement agentique

Un aperçu de l'évolution de l'IA depuis les systèmes de base basés sur des règles jusqu'aux flux de travail agentiques entièrement créatifs.

[↑ Partager la publication](#)

Tout le monde se précipite pour créer des agents IA, mais demandez à cinq ingénieurs ce que cela signifie réellement et vous obtiendrez cinq réponses différentes. Au lieu de débattre de définitions, parlons de ce qui compte vraiment : ce que ces systèmes peuvent réellement faire.

Quel est leur degré d'autonomie, de raisonnement et d'adaptabilité ? Où se trouvent leurs limites ? Et à quel point sommes-nous proches d'agents capables d'agir de manière véritablement autonome ?

*C'est là que les choses deviennent intéressantes.*

En fin de compte, chaque système d'IA dispose d'un certain niveau d'autonomie, de contrôle et de prise de décision. Mais toutes les autonomies ne se valent pas.

Pour donner un sens à tout cela, nous avons élaboré un cadre à six niveaux (N0-N5) qui le décompose. L'idée vient de la façon dont les véhicules autonomes définissent l'autonomie - non pas comme un saut soudain, mais comme une progression graduelle et structurée. Les voitures autonomes n'atteignent pas l'autonomie L3+ sans maîtriser au

Les agents d'IA suivent le même modèle, chaque niveau ajoutant plus de complexité, de raisonnement et d'indépendance.

Voici comment nous décomposons cela. Si vous avez des idées, nous serions ravis de les entendre : c'est l'un de ces sujets dont nous pourrions parler pendant des heures.

## L0 : Flux de travail basé sur des règles (Suiveur)

À ce niveau, il n'y a pas d'intelligence, juste une logique du type « si ceci, alors cela ». Considérez cela comme une macro Excel :

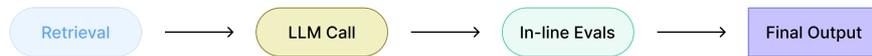
- Aucune prise de décision — il suffit de suivre des règles prédéfinies.
- Aucune adaptation : toute modification nécessite des mises à jour manuelles.
- Aucun raisonnement : il ne « pense » pas, il exécute simplement.

Exemples ? Des systèmes d'automatisation traditionnels comme les workflows Zapier, les planificateurs de pipelines et les robots scriptés. Utiles, mais rigides : ils s'arrêtent dès que les conditions changent.

## L1 : Répondeur de base (Exécuteur)

Maintenant, nous commençons à voir un *tout petit* peu d'autonomie.

À ce niveau, l'IA peut traiter des entrées, récupérer des données pertinentes et générer des réponses basées sur des modèles. Mais elle manque encore de véritable capacité d'action : elle est purement réactive, ne planifie pas et n'a pas de mémoire.



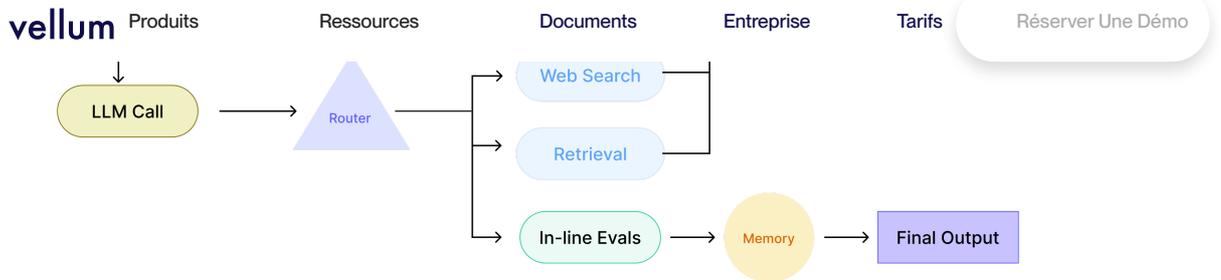
Mais voici la principale limitation : pas de boucle de contrôle. Pas de mémoire, pas de raisonnement itératif, pas de prise de décision autonome.

*C'est purement réactif.*

À mesure que vous progressez dans les niveaux, vous verrez comment de petits changements, comme l'ajout de mémoire, le raisonnement en plusieurs étapes ou l'interaction avec l'environnement, commencent à libérer une véritable autonomie.

## L2 : Utilisation des outils (Acteur)

À ce stade, l'IA ne se contente pas de répondre, elle *exécute*. Elle peut décider d'[appeler des outils externes](#), de récupérer des données et d'incorporer des résultats dans sa sortie. C'est là que l'IA cesse d'être une autocomplétion glorifiée et *fait* réellement quelque chose. Cet agent peut prendre des décisions d'exécution (par exemple, « Dois-je rechercher ceci ? »).

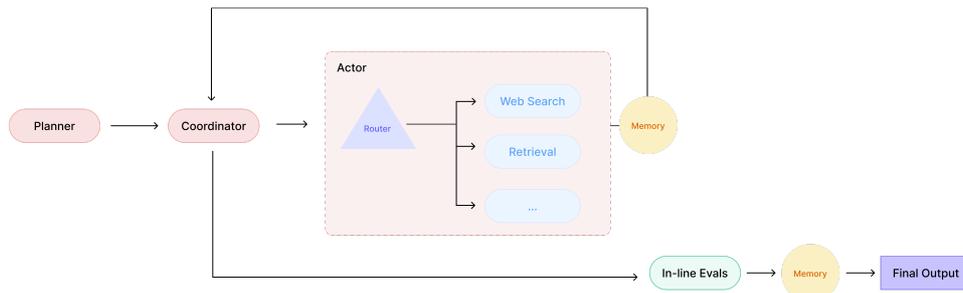


Le système décide quand récupérer des données à partir d'API, interroger des moteurs de recherche, extraire des bases de données ou référencer [la mémoire](#). Mais dès que l'IA commence à utiliser des outils, les choses se compliquent. Il lui faut une sorte de détecteur de conneries intégré, sinon elle risque de se retrouver avec confiance dans des hallucinations.

La plupart des applications d'IA actuelles se situent à ce niveau. C'est un pas vers l'agence, mais elles restent fondamentalement réactives : elles n'agissent que lorsqu'elles sont déclenchées, avec un peu de douceur d'orchestration en plus. Elles ne comportent pas non plus de raffinement itératif : si elles font une erreur, elles ne se corrigent pas d'elles-mêmes.

## L3 : Observer, Planifier, Agir (Opérateur)

Au niveau L2, l'IA ne se contente pas de réagir : elle *gère l'exécution*. Elle planifie les étapes, évalue ses propres résultats et s'ajuste avant d'avancer.



Voici ce qui change :

- **Détecte les changements d'état** – Surveille les déclencheurs tels que les mises à jour de la base de données, les nouveaux e-mails ou les messages Slack.
- **Planifie des flux de travail en plusieurs étapes** – Ne renvoie pas seulement une sortie ; séquence les actions en fonction des dépendances.
- **Exécute des évaluations internes** – Avant de passer à l'étape suivante, il vérifie si la dernière a réellement fonctionné.

C'est un grand pas en avant par rapport à l'utilisation d'un simple outil, mais il y a toujours une limite : une fois la tâche terminée, le système s'arrête. Il ne définit pas ses propres objectifs et ne fonctionne pas indéfiniment. Même lorsque Sam et son équipe livreront GPT-5, il sera toujours bloqué au niveau 2 : un orchestrateur sophistiqué, pas un agent véritablement autonome.

À l'heure actuelle, ces flux de travail sont plus proches de l'automatisation sophistiquée que de l'agence.

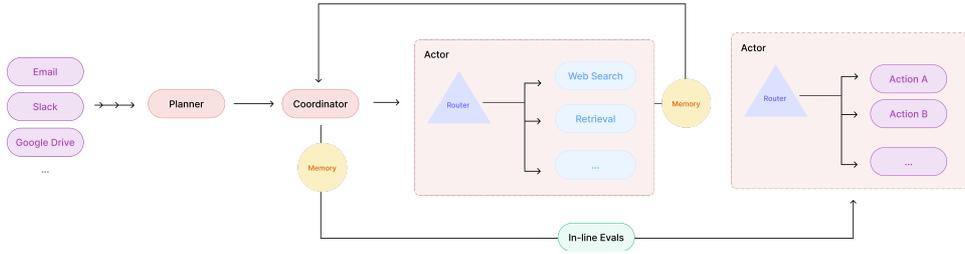
Puissant ? Absolument. Autodirigé ? Pas tout à fait.

## L4 : Entièrement autonome (Explorer)

Au niveau L3, les agents commencent à se comporter comme des systèmes à *état*. Au lieu d'exécuter des boucles de tâches isolées, ils :

- **Maintenir l'état** – Ils restent en vie, surveillent les environnements et persistent d'une session à l'autre.

- Améliorer l'exécution en temps réel - ils ajustent les stratégies en fonction des commentaires, et pas seulement de règles statiques.



Cela commence à ressembler à un système indépendant. Il peut « surveiller » plusieurs flux (e-mail, Slack, bases de données, API), planifier des actions et les exécuter sans intervention humaine constante.

Mais nous n'en sommes qu'aux débuts.

La plupart des workflows d'agents L3 actuels ne persistent pas *de manière fiable* d'une session à l'autre, ne s'adaptent pas de manière dynamique ou n'itèrent pas au-delà des boucles prédéfinies. Le mot clé ici est « fiable ». Il existe des solutions, mais fonctionnent-elles vraiment bien ? C'est discutable.

## L5 : Pleinement créatif (inventeur)

À ce stade, l'IA ne se contente pas d'exécuter des tâches prédéfinies : elle crée sa propre logique, construit des outils à la volée et compose dynamiquement des fonctions pour résoudre des problèmes auxquels nous n'avons pas encore de réponse. Elle ne se contente pas de suivre n'importe quoi ; elle conçoit ses utilitaires à partir de zéro en fonction de la tâche qui lui est confiée.

Nous n'en sommes pas encore là.

Les modèles actuels sont encore sur-ajustés : ils sont bons pour régurgiter, mais mauvais pour raisonner réellement.

Même les modèles les plus puissants (par exemple o1, o3, Deepseek R1) sont toujours sur-ajustés et suivent des heuristiques codées en dur.

Mais c'est là l'objectif : une IA qui ne se contente pas de suivre des instructions, mais qui trouve de nouvelles façons d'améliorer, de créer et de résoudre des problèmes de *manière innovante*.

## Où sommes-nous actuellement?

	Production-Grade		Most innovation		
	L0: Responding	L1: Using Tools	L2: Observing/Reacting	L3: Fully Autonomous	L4: Fully Creative
	Generates <b>outputs</b> for specific tasks using AI.	Selects <b>tools</b> , loops, and runs actions in parallel.	Reacts to <b>triggers</b> in the environment, and plans actions.	Runs <b>proactively</b> , and never terminates.	Can create its own workflow, and <b>invent</b> new things
CALLER	✓	✗	✗	✗	✗
ACTOR	✓	✓	✗	✗	✗
OPERATOR	✓	✓	✓	✗	✗
EXPLORER	✓	✓	✓	✓	✗
INVENTOR	✓	✓	✓	✓	✓

La plupart des systèmes d'IA actuels se situent au niveau L1.

L'accent est mis sur l'orchestration : optimiser la façon dont les modèles interagissent avec le reste du système, peaufiner les invites, optimiser la récupération et les évaluations, et expérimenter différentes modalités. Ces éléments sont également plus faciles à gérer et à contrôler en production : le débogage est un peu plus facile de nos jours et les modes d'échec sont *plutôt* prévisibles.

C'est au niveau L2 que se déroule la majeure partie de l'action en ce moment.

Des modèles comme O1, O3-mini et DeepSeek ouvrent la voie à des workflows multi-étapes plus intelligents. Nous voyons également apparaître de nouveaux produits et expériences d'interface utilisateur vraiment intéressants.

La plupart des entreprises ne touchent pas au L2. Pour l'instant, ce sont presque exclusivement les startups qui se lancent dans ce domaine. Il y a une raison pour laquelle la plupart des flux de production d'IA reposent toujours sur l'intervention humaine : les LLM ne gèrent pas bien les cas extrêmes, et déboguer un agent qui a déraillé il y a trois étapes est un cauchemar.

L3 et L4 sont toujours limités.

La technologie n'est tout simplement pas encore au point, tant au niveau du modèle (*les LLM s'accrochent à leurs données de formation comme une couverture de sécurité*) qu'au niveau de l'infrastructure, où nous manquons de primitives clés pour une véritable autonomie.

## Limites actuelles

Même les modèles les plus puissants sont encore suréquipés comme des fous .

La semaine dernière, nous avons effectué [une évaluation](#) à l'aide d'énigmes bien connues, que ces modèles ont *certainement* rencontrées lors de leur formation. Nous les avons ensuite légèrement modifiées. Le résultat ? Les modèles n'ont pas pu s'adapter et ont simplement régurgité les solutions qu'ils avaient apprises, même lorsqu'elles ne correspondaient pas à la nouvelle version du problème.

Prenons DeepSeek-R1, formé principalement avec du RL pur au lieu d'un corpus massif. On pourrait penser qu'il se généraliserait mieux, n'est-ce pas ? Non. Il y a toujours *des surajustements* . On a l'impression que nous sommes confrontés à un maximum local avec ces modèles.

Et voici le problème : les flux de travail véritablement autonomes dépendent de modèles capables de raisonner, et pas seulement de remixer les données d'entraînement. Pour l'instant, nous n'en sommes pas encore là.

Alors oui, nous verrons des améliorations progressives. Mais un véritable saut vers le niveau 3 ou le niveau 4 ?

Rien n'est sûr. Il faudra peut-être une percée fondamentale (*je te regarde, Ilya Sutskever*) – ou nous pourrions simplement rester coincés ici pendant un certain temps.

## Montez dans la pile avec du vélin

Si les agents d'IA veulent progresser dans la hiérarchie, les équipes ont besoin de meilleurs moyens pour tester, évaluer et affiner leurs flux de travail.

C'est là qu'intervient [le vélin](#) .

À l'heure actuelle, la plupart des développements d'IA reposent sur des essais et des erreurs : peaufiner les invites, ajuster la logique et espérer le meilleur. Mais à mesure que vos flux de travail deviennent plus complexes (en particulier au niveau L2+), le débogage devient un cauchemar. Un appel d'outil erroné, une mauvaise étape de récupération, et tout se brise sur trois niveaux.

Vellum fournit [des tests unitaires et de workflow de bout en bout](#) puissants pour rendre les itérations plus rapides et plus efficaces. Que vous affiniez la logique des agents ou que vous testiez des cas extrêmes, un cadre flexible peut vous aider à progresser de manière fiable dans la pile L0-L4.

[Prenez rendez-vous](#) pour discuter avec l'un de nos experts en IA et découvrez comment Vellum peut vous aider à progresser dans la hiérarchie.

**Anita Kirkovska**Responsable de la croissance fondatrice [in](#)

Experte en IA avec une solide expérience en ML, spécialisée dans la formation GenAI et LLM. Ancienne boursière Fulbright, elle dirige la croissance et l'éducation chez Vellum, aidant les entreprises à créer et à faire évoluer leurs produits d'IA. Elle mène des évaluations LLM et écrit abondamment sur les meilleures pratiques en matière d'IA, permettant aux chefs d'entreprise de favoriser une adoption efficace de l'IA.

**Nico Finelli**Création de GTM [in](#)

Nico est un Data Scientist spécialisé dans le machine learning et l'IA, avec un don pour créer des solutions innovantes. En tant que fondateur et dirigeant de GTM, il comble le fossé entre l'expertise technique et la stratégie de mise sur le marché.

**David Vargas**Ingénieur Fondateur Full Stack [in](#)

Ingénieur fondateur Full Stack chez Vellum, David Vargas est diplômé du MIT (2017) avec une expérience dans une startup de série C et en tant qu'ingénieur open source indépendant. Il a créé des outils de réflexion au sein de sa société, SamePage, et se concentre désormais sur la conception de la prochaine ère d'outils de réflexion basés sur l'IA chez Vellum.

## Les meilleurs conseils en matière d'IA – directement dans votre boîte de réception

-  Dernières actualités, astuces et techniques en matière d'IA
-  Conseils spécifiques pour vos cas d'utilisation de l'IA
-  Pas de spam

“

Chaque numéro regorge de ressources, d'outils et d'idées précieuses qui nous aident à garder une longueur d'avance dans le développement de l'IA. Nous avons découvert des stratégies et des cadres qui ont augmenté notre efficacité de 30 %, ce qui en fait un incontournable pour toute personne travaillant dans le domaine.

”

**Marina Trajkovska**

Responsable de l'ingénierie

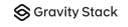


C'est une excellente newsletter. Le contenu est très utile, même lorsque je suis occupé, je les lis.



Jérémy Hicks

Architecte de solutions



### Articles Similaires

→ Voir Plus



GUIDES

GPT 4.5 est là : mieux, mais pas...

27 FÉVRIER 2025



GUIDES

Comment les capacités des...

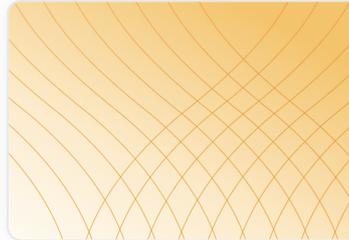
18 FÉVRIER 2025



GUIDES

Claude 3.7 Sonnet : Peut-il réelleme...

30 JANVIER • 5 MINUTES 2025



GUIDES

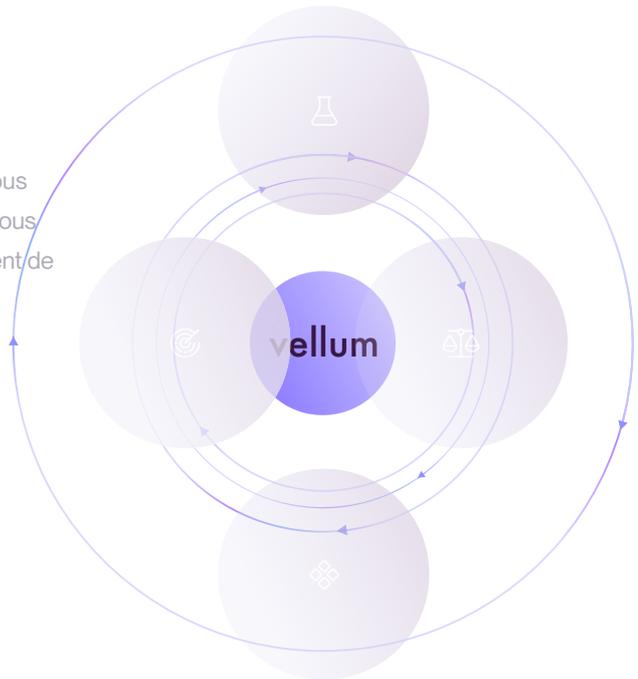
Décomposer le processus de...

24 JANVIER • 10 MINUTES 2025

# Expérimenter, évaluer, déployer, répéter.

Le développement de l'IA ne s'arrête pas une fois que vous avez défini votre système. Découvrez comment Vellum vous aide à gérer l'intégralité du cycle de vie du développement de l'IA.

- Incitation
- Orchestration
- Évaluation
- Récupération
- Déploiement
- Surveillance



Créez des systèmes d'IA auxquels vous pouvez faire confiance

RESSOURCES

- Études De Cas
- Modèles De Raisonnement
- Guides
- Mises À Jour Des Produits
- Comparaison De Modèles
- Documentation
- Classement LLM
- Outils Gratuits
- Bulletin
- Cas D'utilisation

PRODUITS

- Ingénierie Rapide
- Récupération De Documents
- Orchestration
- Évaluations
- Déploiements
- Surveillance
- Kit De Développement Logiciel (SDK)

ENTREPRISE

- Blog
- Carrières
- Contactez-Nous
- Conditions D'utilisation
- Politique De Confidentialité

RÉSEAUX SOCIAUX

- LinkedIn
- Gazouillement
- Youtube

