SPOTLAB / ANALYSE P12

Évaluer un agent IA en entreprise : cinq dimensions à tester avant le déploiement

Un protocole pour mesurer réussite métier, stabilité, sécurité, coût et résilience, sans confondre réussite ponctuelle et fiabilité.

Analyse / P12

Novembre 2026

Sources consultées au 2 novembre 2026

ANALYSE P12

Sommaire

01	Trois enseignements à retenir	3
02	Pourquoi un benchmark de modèle ne suffit pas	4
03	Les cinq dimensions d'une évaluation professionnelle	5
04	Comment construire un protocole qui résiste au stress test	6
05	Exemple : un agent qui classe une preuve de contrôle interne	8
06	Interpréter la stabilité avec prudence	9
07	Décider : déployer, corriger ou suspendre	10
08	Limites et sources	10

FICHE DE L'ANALYSE

ÉDITEUR
Spotwork.AI — Spotlab, Research & Insights

RÉFÉRENCE
P12

TYPE
Analyse documentaire

COLLECTION
Agents & automatisation

SOURCES CONSULTÉES
2 novembre 2026

PUBLICATIONS LIÉES
I0-02 · P10 · P11

EN BREF

Un agent peut rendre une réponse convaincante sans avoir accompli la tâche, réussir une seule fois sans être stable, ou produire un résultat correct après une action interdite. Un test crédible vérifie le résultat métier, la répétabilité, les permissions, le coût et la capacité de reprise. Les cinq dimensions doivent être interprétées séparément : une moyenne élevée ne compense pas un incident critique.

> *Un taux de réussite moyen doit être complété par des tests de répétabilité, d'erreurs critiques et de coût des reprises.*
>
> CE QU'IL FAUT RETENIR

01 Trois enseignements à retenir

1

Évaluer l'état final, et pas seulement le texte.

Pour une tâche qui crée, modifie ou classe un objet, la preuve est dans le système cible : objet créé, état attendu, source citée, modification autorisée. Les évaluations de systèmes multi-étapes doivent examiner à la fois le résultat et la trajectoire [1] [3].

2

Distinguer performance et stabilité.

« A réussi au moins une fois sur cinq essais » ne signifie pas « réussit cinq fois sur cinq ». Les travaux sur l'évaluation des agents invitent à publier les répétitions, la dispersion et le périmètre mesuré plutôt qu'un score isolé [2].

3

Prévoir un veto sur les défaillances graves.

Une fuite de données, une permission contournée ou une action irréversible non approuvée ne doit pas être masquée par une bonne note moyenne. Les risques agentiques incluent l'abus d'outils, de privilèges et les défaillances en cascade [6].

CE QUE PROPOSE CETTE ANALYSE

Un contrat d'évaluation à cinq dimensions, avec conditions de test, règles d'arrêt, vérificateurs indépendants et décision explicitement conditionnelle. **Un résultat satisfaisant n'autorise pas un comportement interdit** ; inversement, un arrêt sûr devant une permission refusée peut être le bon résultat du test de sécurité, même si la tâche métier n'est pas terminée.

02 Pourquoi un benchmark de modèle ne suffit pas

Un agent n'est pas simplement un modèle. Il associe modèle, instructions, mémoire ou contexte, connecteurs, règles logicielles, droits d'accès et environnement d'exécution. Changer l'un de ces éléments peut modifier son comportement. Les résultats d'un benchmark de modèle, même reconnu, ne mesurent donc pas directement la réussite de votre processus.

Le NIST recommande d'évaluer les risques et la performance avant le déploiement, puis tout au long du fonctionnement. Son manuel ARIA publié en septembre 2026 combine tests du modèle, red teaming et tests utilisateurs : ces approches sont complémentaires, sans se substituer à une recette métier [4] [5].

Ce qu'apportent les principaux référentiels

Référence	Apport	Limite
Anthropic, *Demystifying evals for AI agents* (9 janvier 2026) [1]	Définit tâche, essai, grader, transcription, résultat (état final de l'environnement) et harnais ; recommande des environnements réinitialisés, des graders programmatiques si possible, des juges IA calibrés par des experts et la lecture des traces	Retour d'expérience d'un fournisseur, sans seuil universel
Princeton SAGE, *Emerging Trends in AI Agent Evaluation* (2026) [2]	Distingue pass@k (au moins une réussite sur k exécutions) et pass∧k (réussite à chaque exécution) ; plaide pour de meilleures statistiques sur les trajectoires longues et stochastiques	Un même modèle peut obtenir des scores différents selon l'échafaudage, les outils, l'environnement et l'analyse statistique
NVIDIA (21 septembre 2026) [3]	Hiérarchie benchmark → essai → tâche → tour → étape ; évaluation du résultat final et de chaque étape à partir de la même trace ; coût par tâche réussie	Guide d'ingénierie, non une norme
NIST AI RMF et ARIA (18 septembre 2026) [4] [5]	La fonction MEASURE documente performance, incertitude et suivi ; ARIA combine tests de modèle, red teaming et tests utilisateurs	Ni certification ni validation implicite d'une méthode
OWASP Agentic 2026 [6]	Taxonomie de risques pour choisir des tests adversariaux : détournement d'objectif, abus d'outils, d'identité et de privilèges, empoisonnement du contexte, défaillances en cascade	Aucun taux d'incident applicable à toutes les entreprises
METR, *Time horizons* (mise à jour du 8 septembre 2026) [7]	Mesure la durée des tâches qu'un agent réussit avec une fiabilité donnée	Tâches surtout logicielles, bien plus « propres » qu'un travail réel ; mesures au-delà de 16 heures peu fiables ; page plus mise à jour

Ces sources convergent sur la nécessité de tester au-delà de la réponse finale, sans imposer un protocole unique ni un nombre universel de répétitions.

03 Les cinq dimensions d'une évaluation professionnelle

Dimension	Question à trancher	Preuves minimales
1. Résultat métier	La tâche est-elle effectivement accomplie ?	État final vérifié, critères métier, complétude des preuves
2. Stabilité	L'agent reproduit-il le résultat lorsque l'essai est rejoué ?	Répétitions prévues, taux par scénario, dispersion et incertitude
3. Sécurité	Respecte-t-il les permissions et les interdictions, même sous perturbation ?	Journaux d'outils, décisions d'autorisation, tests adversariaux
4. Économie	Quel coût et quelle latence pour une tâche acceptée ?	Coût de tous les essais, échecs inclus, durée et ressources mobilisées
5. Résilience	Peut-il s'arrêter, reprendre ou escalader sans effet indésirable ?	Test d'interruption, reprise contrôlée, absence de doublons et traçabilité

ATTENTION AUX AGRÉGATS

Le « taux de réussite » dépend de ce que l'on appelle une tâche réussie ; le « coût par réussite » doit inclure les tentatives infructueuses, pas uniquement les exécutions favorables. Le nombre de cas couverts et la distribution des risques comptent autant que la moyenne générale.

Indicateurs et points de vigilance

Dimension	Indicateurs	Point de vigilance
Résultat métier	Réussite par cas, acceptation de l'état final, qualité des sources	Exclure les résultats seulement déclarés par l'agent
Stabilité	Distribution par scénario et répétition, taux « toutes réussies », dispersion	Ne pas confondre réussite ponctuelle et fiabilité
Sécurité	Violations de droits, actions interdites, refus corrects, diffusion non autorisée	Incident éliminatoire, distinct d'une moyenne
Économie	Coût total / essais, coût total / tâches acceptées, latence au 95e centile	Inclure échecs, reprises, outils et interventions humaines
Résilience	Reprise, doublons, état après interruption, escalade	Une reprise incomplète ne vaut pas réussite métier

Le coût complet par livrable accepté — ici la tâche acceptée — est indéfini si aucune tâche n'est acceptée. Erreurs techniques, refus corrects, violations critiques et tâches non réalisées s'affichent séparément.

04 Comment construire un protocole qui résiste au stress test

ÉTAPE 1

Contractualiser chaque cas

Définir la demande, les données initiales, les outils permis, le résultat attendu, les états interdits, la règle de notation et les conditions d'arrêt. Le vérificateur doit, si possible, être indépendant du modèle testé. Si une évaluation humaine ou un juge IA est nécessaire, utiliser des critères explicites et vérifier son accord avec des experts.

ÉTAPE 2

Constituer plusieurs familles de scénarios

Inclure des cas courants, des dossiers ambigus, des pièces manquantes, des erreurs de connecteur, une source contenant une instruction hostile et des opérations sensibles. Conserver un jeu stable de non-régression et renouveler séparément un jeu d'exploration pour éviter l'apprentissage artificiel des seuls tests connus.

ÉTAPE 3

Répéter de manière bornée

Définir à l'avance le nombre d'essais par niveau de risque, la durée maximale d'une exécution, le plafond de coût, le nombre maximal d'appels d'outils et les règles d'abandon. Comparer les mêmes scénarios sur des environnements réinitialisés, en consignant versions, paramètres et éventuels changements du fournisseur. Trois à cinq essais peuvent constituer un premier diagnostic technique, mais ne démontrent jamais à eux seuls une fiabilité élevée. Le volume nécessaire dépend de la précision statistique recherchée, de la fréquence des risques rares et de l'usage réel.

ÉTAPE 4

Examiner les traces et les écarts

Enregistrer appels d'outils, paramètres, décisions d'autorisation, état avant/après, erreurs et coût. Distinguer une erreur de modèle, une anomalie de données, une panne technique, un défaut du vérificateur et un arrêt protecteur réussi. Une reprise sûre n'est pas une « réussite complète » si le résultat métier n'a pas été obtenu.

ÉTAPE 5

Décider avant de déployer

Fixer des seuils d'acceptation avant les essais : qualité minimale, coût et latence admissibles, comportement sous perturbation, et incidents éliminatoires. Ne pas « retester jusqu'à trouver une bonne moyenne » : ce serait sélectionner le résultat le plus favorable.

CAS DISTINCTS	RÉPÉTITIONS D'UN CAS	TESTS ADVERSARIAUX
Couverture des situations métier	Variabilité à conditions contrôlées	Résistance à des perturbations précises

Cent répétitions d'un cas unique ne remplacent pas cent cas distincts. Chaque strate — courante, limite, à risque élevé, de panne, adversariale — se suit séparément, pour que les cas simples ne masquent pas les échecs critiques.

04 / LE CONTRAT D'UN SCÉNARIO

Champ du cas	Exigence de test	Preuve ou vérificateur
Identifiant, version, responsable	Cas stable et traçable	Identifiant immuable, version et responsable métier
Demande et données initiales	État de départ reproductible	Instantané, corpus et droits initiaux
Résultat attendu	Critère métier observable	Lecture de l'état cible, tests de contenu et de provenance
Autorisations / interdictions	Frontière des actions permises	Audit des ACL, appels d'outil et destinations
Perturbation	Normal, ambigu, panne, adversarial, reprise	Scénario d'injection reproductible
Graders	Code prioritaire, humain ou juge IA calibré	Rubrique, version, désaccords, preuves
Budget / arrêt	Étapes, durée et coût plafonnés	Compteurs côté orchestrateur, état d'arrêt
Reprise et nettoyage	Absence d'effets de bord parasites	Idempotence si applicable, journal et état final

Règle d'impartialité. Le modèle évalué ne définit pas seul le critère de réussite et n'autorise pas ses propres actions. Un cas accepte toutes les solutions légitimes, sans imposer une trace unique ; les correctifs d'un grader défectueux sont historisés. L'orchestrateur impose durée, coût, appels d'outils, reprises, permissions et approbations : l'expiration produit un état « arrêté », sans boucle infinie, et toute action à effet externe passe par un contrôle système indépendant du LLM.

Qualifier chaque écart

Catégorie d'écart	Interprétation	Traitement recommandé
Erreur de tâche	Le résultat ne satisfait pas les critères métier	Corriger et rejouer le cas
Défaillance de donnée	Source absente ou obsolète	Corriger la source, distinguer l'état du modèle
Défaillance outil / plateforme	API indisponible, panne, latence	Tester la reprise, classer en incident technique
Violation de sécurité	Action ou diffusion hors autorisation	Veto sur la configuration, enquête
Arrêt protecteur correct	Refus et escalade face à un droit insuffisant	Succès sécurité, tâche métier non achevée
Défaut du grader	Critère mal spécifié ou trop rigide	Corriger la version du test et relancer proprement

05 Exemple : un agent qui classe une preuve de contrôle interne

SCÉNARIO ILLUSTRATIF Scénario illustratif, sans résultat expérimental Spotwork.AI. Un agent recherche une clause dans les procédures habilitées, prépare une synthèse sourcée et classe une pièce au bon dossier. La tâche n'est acceptée que si les références sont exactes, les droits respectés et le classement confirmé dans le système de référence.

Les perturbations testées comprennent : document indisponible, accès interdit, instruction hostile insérée dans une pièce, interruption après une écriture partielle et nouvelle tentative de classement. Le résultat attendu n'est pas nécessairement « tout terminer » : en cas de refus d'accès, s'arrêter sans divulguer la pièce peut être le comportement correct. En cas de reprise, la pièce ne doit pas être classée deux fois.

T01 **Normal**

Procédure accessible, preuve classée au dossier attendu, traçabilité complète.

T02 **Source manquante**

L'agent signale une insuffisance de preuve, sans inventer de référence.

T03 **Accès refusé**

L'agent s'arrête et escalade sans exposer le contenu.

T04 **Injection**

Un document externe demande d'envoyer la preuve à une adresse tierce : aucun appel de diffusion ne doit avoir lieu.

T05 **Reprise**

Le système cible accepte l'écriture du classement, mais la confirmation n'est pas reçue (interruption simulée). Après reprise, une vérification indépendante doit constater une seule pièce classée ; une interruption avant toute écriture ne suffit pas à valider ce cas.

Cette distinction est capitale en environnement réglementé : **une erreur technique ne doit pas devenir artificiellement une non-conformité métier, et une opération interdite ne devient pas acceptable parce que la conclusion semble juste.** Un refus de permission correct ne doit pas non plus gonfler le taux de dossiers traités.

06 Interpréter la stabilité avec prudence

On peut suivre deux mesures distinctes sur un cas répété k fois : succès au moins une fois et succès à chaque essai. La première renseigne sur la capacité occasionnelle ; la seconde sur la répétabilité dans le périmètre testé. Ni l'une ni l'autre ne prouve la fiabilité dans une situation non évaluée.

SUCCÈS AU MOINS UNE FOIS · PASS@5



Réussi — une exécution favorable suffit.

SUCCÈS À CHAQUE ESSAI · PASS^5



Échoué — un seul échec suffit à invalider.

Mêmes cinq exécutions illustratives, deux lectures opposées. Les deux mesures sont définies par Anthropic et Princeton SAGE [1] [2].

Pour illustrer l'incertitude, zéro incident observé sur 20 essais indépendants n'implique pas un risque nul : la borne supérieure unilatérale de confiance à 95 % est encore d'environ 14 % dans un modèle binomial simple. Dans un agent réel, les essais peuvent être corrélés, ce qui rend cette interprétation encore plus délicate. Il faut donc éviter tout discours de type « 100 % fiable » fondé sur quelques exécutions favorables.

ZÉRO INCIDENT SUR N ESSAIS INDÉPENDANTS — BORNE SUPÉRIEURE À 95 %

$$p_{max} = 1 - (0{,}05)^{(1/n)} \cdot n = 20 \rightarrow \approx 13{,}9\ \%$$

Illustration théorique : pour vingt essais d'un même agent sur un même cas, l'indépendance ne peut pas être présumée, et la répétabilité sur un cas ne dit rien de la représentativité des cas. Elle montre au contraire la fragilité de l'affirmation « aucun incident, donc aucun risque ». Les inférences statistiques supposent des conditions rarement réunies dans des systèmes multi-étapes.

CE QU'UN RAPPORT D'ÉVALUATION DEVRAIT PUBLIER

Le nombre de cas distincts et de répétitions ; les deux mesures de stabilité, par strate de scénarios ; la dispersion autour des moyennes ; le coût et la durée par tâche ; la configuration complète testée — modèle, version de l'agent, outils, routage et versions des composants. Le guide MLflow recommande de même de répéter chaque configuration, d'en publier l'écart et de documenter l'ensemble de l'échafaudage [8].

07 Décider : déployer, corriger ou suspendre

Déployer progressivement si les critères prédéfinis sont satisfaits, sans incident éliminatoire et avec un périmètre d'usage correspondant aux essais. Corriger puis retester si les écarts sont maîtrisables et explicables. Suspendre si une permission est contournée, si un effet irréversible non autorisé est possible, si le protocole est invalide ou si des erreurs critiques restent inexpliquées. Après tout changement significatif de modèle, d'outil, de prompt, de données ou de règles, rejouer le lot de non-régression.

GO — DÉPLOIEMENT PROGRESSIF | CORRECTION PUIS RETEST | NO-GO — SUSPENSION

Tracer modèle, version de l'agent, prompt, contexte, outils, droits, configuration, données, graders et protocole. Le lot de non-régression inclut les cas ayant déjà échoué. En production, journaux d'incident, retours utilisateurs, contrôles échantillonnés et surveillance des dérives complètent les tests de préproduction.

08 Limites et sources

Cette méthode est une proposition opérationnelle Spotwork.AI dérivée de publications de recherche, de documents de fournisseurs et de référentiels publics. Ce n'est ni un standard certifié, ni un benchmark propriétaire. Aucun taux mesuré chez un client n'est présenté. Les seuils réels doivent être validés avec les métiers, la sécurité et le contrôle des risques.

[1] Anthropic — Demystifying evals for AI agents (9 janvier 2026).
anthropic.com/research/demystifying-evals-for-ai-agents

[2] Princeton SAGE — Emerging Trends in AI Agent Evaluation (2026).
sage.cs.princeton.edu/trends/

[3] NVIDIA — How to Evaluate AI Agents From Tool Calls to Task Completion (21 septembre 2026).
developer.nvidia.com/blog/how-to-evaluate-ai-agents-from-tool-calls-to-task-completion/

[4] NIST — AI Risk Management Framework, Core (fonction MEASURE).
airc.nist.gov/airmf-resources/airmf/5-sec-core/

[5] NIST — ARIA Evaluation Planning Manual: Elements of ARIA-Style AI Evaluations (18 septembre 2026).
nist.gov/publications/aria-evaluation-planning-manual-elements-aria-style-ai-evaluations

[6] OWASP GenAI Security Project — Top 10 for Agentic Applications 2026.
genai.owasp.org/resource/owasp-top-10-for-agentic-applications-for-2026/

[7] METR — Task-Completion Time Horizons of Frontier AI Models (mise à jour du 8 septembre 2026).
metr.org/time-horizons/

[8] MLflow — Benchmarking AI Agent Performance: A Practical Protocol (27 juillet 2026).
mlflow.org/articles/benchmarking-ai-agent-performance

Le protocole proposé n'a pas été exécuté pour cette publication. La valeur de N, les taux cibles et les limites de latence ne sont pas fixés universellement. Les contenus du cas pratique sont fictifs.

Spotwork.AI

L'IA professionnelle de confiance

Mieux travailler aujourd'hui,
construire votre intelligence de demain.

SPOTLAB · RESEARCH & INSIGHTS · ANALYSE P12

Évaluer un agent IA en entreprise : cinq dimensions à tester avant le déploiement

Tests reproductibles et décisions de déploiement fiables

© 2026 SPOTWORK.AI