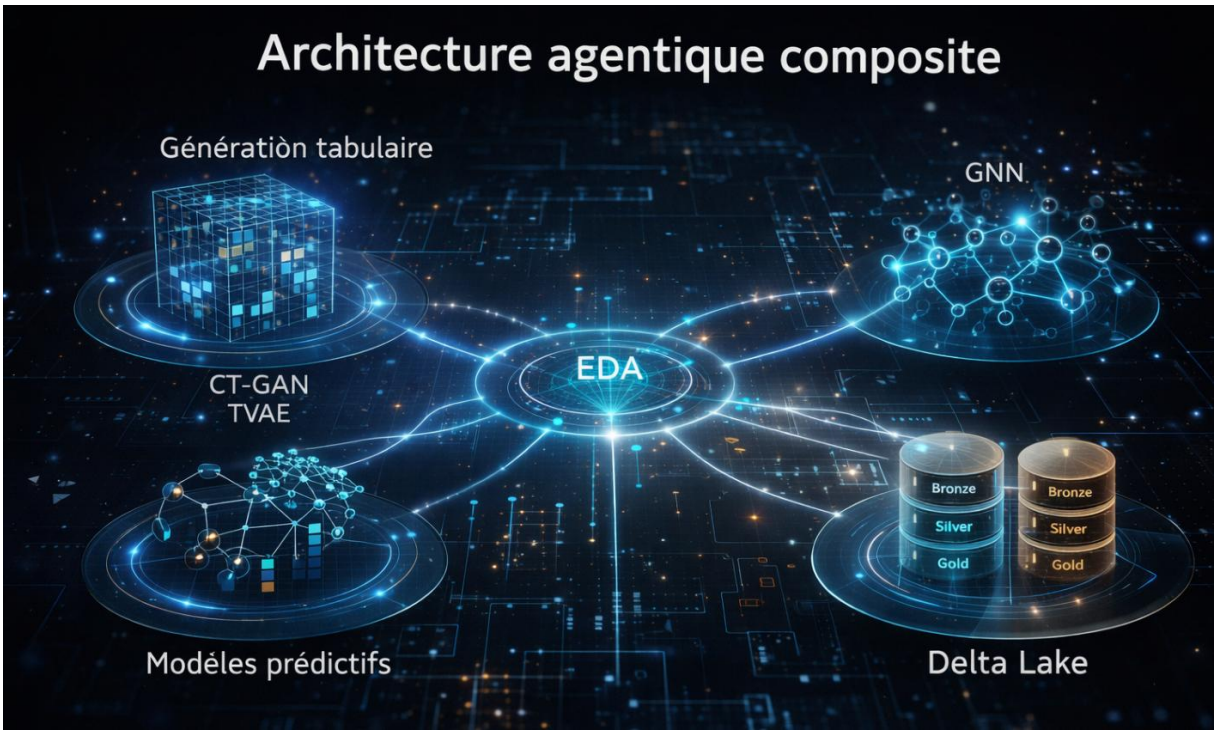


# Au-delà du paradigme LLM-centré : architecture agentique composite pour les jumeaux numériques en environnement régulé

***De la composition algorithmique à l'agent industriel : pourquoi l'IA agentique des environnements régulés ne commence pas par un LLM***

*Jérôme Vetillard / Twingital Institute, avril 2026*

## Résumé

Le discours dominant sur l'IA agentique repose aujourd'hui sur une assimilation implicite : un agent serait un grand modèle de langage (LLM) enrichi par des outils, de la mémoire conversationnelle et des mécanismes d'orchestration. Cette représentation est opératoire pour de nombreux cas d'usage conversationnels, documentaires ou orientés productivité. Elle devient en revanche insuffisante dès lors que l'on considère des environnements régulés caractérisés par des données tabulaires structurées, des dynamiques temporelles, des exigences de calibration probabiliste, des contraintes de traçabilité et des obligations de reproductibilité.

Cet article défend une thèse circonscrite mais ferme : dans les contextes industriels ou cliniques à forte intensité décisionnelle, l'agentique viable ne peut généralement pas être fondée sur un LLM comme noyau computationnel unique. Elle requiert une composition hétérogène d'algorithmes spécialisés, une mémoire de domaine persistante et structurée, ainsi qu'un substrat événementiel assurant coordination, auditabilité et actualisation continue. Dans cette configuration, le LLM conserve un rôle important,

**Jérôme Vetillard**
CTO | VP R&D | Chief Product Officer | AI-Powered Healthcare & Life Sciences Products | Compliance by Design
PhD AgroParisTech | CPO MIT Sloan | Exec MBA IE Business School & Brown University
Twingital-institute / Twingital-ventures : twingital-ventures.com

mais délimité : interprétation contextuelle, médiation linguistique, explicitation des sorties et interaction avec l'humain.

Nous proposons d'appeler **architecture agentique composite** ce triptyque formé par : (1) un ensemble d'agents spécialisés hétérogènes, (2) une mémoire de domaine stratifiée et versionnée, et (3) une orchestration événementielle assurant la cohérence dynamique du tout. Nous introduisons également le concept d'**asymétrie de persistance fonctionnelle** pour désigner une propriété observée dans certains jumeaux numériques en ingestion continue : les couches d'un lakehouse de type Medallion, conçues canonquement selon un axe de maturation qualitative des données, peuvent aussi acquérir une différence de durée de vie fonctionnelle qui autorise, sous conditions, une lecture structurale en termes de mémoire de travail, mémoire épisodique et mémoire sémantique au sens de Tulving [9]. Cette lecture ne vaut ni comme identité ontologique ni comme propriété universelle du Medallion ; elle vaut comme cadre d'interprétation architecturale pour une classe spécifique de systèmes.

Les propositions sont illustrées par deux terrains d'application : la plateforme TweenMe et le programme PREDICARE/Sentinelle IA, mobilisés ici comme instances de mise en œuvre et non comme preuves générales suffisantes à eux seuls. Le repositionnement proposé est double. Il est d'abord épistémologique : assimiler l'agent à un LLM revient souvent à confondre couche d'interface et couche de calcul. Il est ensuite stratégique : dans les environnements régulés, le besoin principal n'est pas un modèle à télécharger, mais une plateforme de composition algorithmique capable de construire, gouverner et faire évoluer des systèmes hétérogènes de calcul, de mémoire et d'action.

**Mots-clés :** IA agentique · architecture agentique composite · essaim algorithmique · jumeau numérique · Delta Lake · asymétrie de persistance fonctionnelle · composition algorithmique · architecture événementielle · mémoire de domaine · AI Act

## 1. Introduction

Dans un article précédent consacré à l'architecture événementielle comme complément essentiel de l'IA agentique [1], nous avons soutenu que la question centrale n'était pas seulement celle des capacités cognitives internes d'un agent, mais celle de son insertion dans un milieu informationnel, technique et organisationnel. L'Event-Driven Architecture (EDA) y apparaissait comme le substrat assurant l'ancrage temporel, la délégation graduée, le découplage inter-systèmes, l'auditabilité et la coordination distribuée.

Cette première analyse portait donc sur le **milieu d'action** de l'agent. Elle laissait ouverte une question plus fondamentale : **de quelle substance computationnelle un système agentique industriel est-il effectivement composé ?**

**Jérôme Vetillard**
CTO | VP R&D | Chief Product Officer | AI-Powered Healthcare & Life Sciences Products | Compliance by Design
PhD AgroParisTech | CPO MIT Sloan | Exec MBA IE Business School & Brown University
Twingital-institute / Twingital-ventures : twingital-ventures.com

Le marché contemporain répond le plus souvent par une formule simple : un agent serait un LLM capable d'appeler des outils, de raisonner en plusieurs étapes, de consulter des ressources externes et de conserver un certain contexte de travail. Cette représentation a une efficacité pragmatique réelle. Elle permet de concevoir rapidement des assistants documentaires, des agents de support, des copilotes applicatifs ou des systèmes de planification légère. Elle ne doit donc ni être caricaturée ni être rejetée en bloc.

Elle devient toutefois insuffisante lorsque l'on s'intéresse à des domaines où l'agent doit non seulement dialoguer, mais aussi produire, exploiter, gouverner ou superviser des calculs spécialisés sur des structures de données non linguistiques : données tabulaires cliniques, séries temporelles physiologiques, graphes moléculaires, modèles de survie, cohortes synthétiques, contraintes d'allocation territoriale, trajectoires de risque, indicateurs médico-économiques. Dans de tels contextes, l'agentique ne peut plus être pensée comme une simple extension du paradigme conversationnel.

La thèse défendue ici est la suivante : **dans les environnements régulés à forte composante tabulaire, temporelle et décisionnelle, le système agentique pertinent prend généralement la forme d'une architecture composite articulant algorithmes spécialisés hétérogènes, mémoire de domaine persistante et orchestration événementielle**. Le LLM y conserve une place majeure, mais il n'en constitue ni l'unique substance ni nécessairement le centre de gravité computationnel.

Cette thèse n'a pas vocation à décrire tous les agents possibles. Elle vise une classe déterminée de systèmes, en particulier les jumeaux numériques opérant dans des environnements cliniques, industriels ou territoriaux. L'enjeu n'est donc pas de nier l'utilité des frameworks LLM-centriques, mais d'identifier les conditions dans lesquelles leur paradigme devient structurellement insuffisant.

## 2. Clarification terminologique

Une partie importante de la confusion contemporaine tient à l'usage instable du terme « agent ». Il convient donc de distinguer plusieurs niveaux analytiques.

### 2.1 Composant algorithmique spécialisé

Nous appelons **composant algorithmique spécialisé** un modèle ou une procédure dont la structure mathématique est adaptée à une classe déterminée de tâches : génération tabulaire, classification supervisée, analyse de survie, modélisation sur graphes, détection d'anomalies, optimisation, interprétation linguistique. Un CT-GAN, un modèle Fine & Gray, un XGBoost, un GNN et un LLM relèvent tous de cette catégorie, mais n'ont ni la même nature ni les mêmes garanties.

**Jérôme Vetillard**
CTO | VP R&D | Chief Product Officer | AI-Powered Healthcare & Life Sciences Products | Compliance by Design
PhD AgroParisTech | CPO MIT Sloan | Exec MBA IE Business School & Brown University
Twingital-institute / Twingital-ventures : twingital-ventures.com

### 2.2 Agent spécialisé

Nous appelons ici **agent spécialisé** un composant algorithmique encapsulé dans une capacité d'action située. Il reçoit certains types d'entrées, produit certains types de sorties, s'inscrit dans des politiques d'activation, publie ou consomme certains événements, et interagit avec une mémoire de domaine. Dans cette acception, un agent n'est pas nécessairement un LLM. Ce choix terminologique dépsychologise délibérément la notion d'agent : ce qui importe n'est pas une apparence de conversation ou d'intentionnalité, mais une fonction située dans un système de transitions, d'états et de dépendances.

### 2.3 Système agentique composite

Nous appelons **système agentique composite** l'ensemble coordonné de plusieurs agents spécialisés hétérogènes, couplés à une mémoire partagée et à une infrastructure de coordination. C'est à ce niveau que l'on peut parler d'architecture agentique au sens fort.

### 2.4 Jumeau numérique

Enfin, nous appelons **jumeau numérique** une classe particulière de systèmes composites visant à maintenir une représentation opérationnelle, évolutive et exploitable d'un référent réel, individuel ou collectif : patient, cohorte, molécule, infrastructure, territoire. Tous les agents ne sont donc pas des jumeaux numériques. En revanche, certains jumeaux numériques avancés peuvent être compris comme des systèmes agentiques composites.

Cette clarification permet d'éviter deux erreurs symétriques : réduire tout agent à un chatbot outillé, ou décréter qu'un agent digne de ce nom devrait déjà être un jumeau numérique complet.

## 3. Le biais LLM-centré : genèse et portée

L'assimilation contemporaine entre agent et LLM n'est pas le résultat d'une démonstration théorique. Elle tient davantage à une conjonction historique entre capacités techniques nouvelles, logique de démonstration de marché et héritage conceptuel du chatbot.

Premièrement, les LLM ont produit un choc cognitif légitime. Leur capacité à résumer, reformuler, planifier, coder, questionner ou synthétiser a créé une impression de généralité fonctionnelle. Cette impression n'est pas entièrement illusoire, mais elle a souvent été extrapolée au-delà de son domaine de validité. La confusion entre la capacité

à **parler d'un domaine** et la capacité à **calculer dans un domaine** constitue l'un des ressorts principaux du biais LLM-centré.

Deuxièmement, la facilité de prototypage a joué un rôle décisif. Il est possible de construire en quelques heures un agent conversationnel capable d'enchaîner plusieurs appels d'outils et de produire une démonstration convaincante. En comparaison, la constitution d'un pipeline validé de génération tabulaire, de modélisation prédictive calibrée ou de simulation contrefactuelle exige des temps de développement, de validation et de documentation beaucoup plus longs. Le marché a donc privilégié les architectures qui maximisent le **time-to-demo**, en confondant parfois maturité de démonstration et maturité architecturale [2].

Troisièmement, le concept d'agent a été réinterprété à partir du paradigme du chatbot. Au lieu de penser l'agent comme un système situé dans des flux d'événements, muni d'états, de politiques d'action, de dépendances mémorielles et de capacités hétérogènes, on l'a souvent pensé comme un interlocuteur rendu plus autonome. Cette translation est compréhensible. Elle n'est pas neutre. Elle a conduit à survaloriser la cohérence discursive comme critère principal d'intelligence opératoire.

Le concept d'agent, déjà établi de longue date dans la littérature multi-agents [3], s'est ainsi trouvé recentré autour du LLM comme noyau cognitif supposé universel. Les frameworks contemporains ont cristallisé cette identification en proposant une abstraction où « agent » signifie souvent, en pratique, « un LLM qui décide quel outil appeler ». Ces facteurs expliquent la domination du paradigme. Ils n'en démontrent pas l'universalité.

## 4. Limites du paradigme LLM-centré dans les environnements régulés à forte composante structurée

Il ne s'agit pas ici de soutenir qu'un LLM serait incapable de contribuer à des systèmes complexes. Il s'agit de montrer que, dans certains contextes, il ne constitue pas à lui seul le support computationnel adéquat des fonctions critiques dont dépend la validité opérationnelle du système.

### 4.1 Génération tabulaire et fidélité structurelle

Un LLM peut décrire un jeu de données, commenter des statistiques descriptives, suggérer des transformations, voire piloter un outil externe de génération synthétique. En revanche, la production de cohortes synthétiques tabulaires fidèles à une distribution réelle repose sur des mécanismes spécialisés distincts de ceux des modèles de langage. Elle implique la modélisation de distributions jointes, de dépendances conditionnelles, d'interactions entre variables catégorielles et continues, ainsi que des protocoles de validation adaptés.

Dans ce domaine, la question n'est pas seulement de générer des exemples plausibles, mais de préserver de manière contrôlée des propriétés statistiques pertinentes pour les tâches aval. Cela suppose des approches dédiées telles que CT-GAN [4], TVAE, copules gaussiennes ou autres architectures spécialisées. Le point important n'est donc pas qu'un LLM serait incapable de participer à un tel pipeline, mais qu'il n'en constitue ni le mécanisme mathématique central ni la source principale de validité.

### 4.2 Prédiction supervisée sur données tabulaires et cliniques

Dans de nombreux problèmes structurés, notamment cliniques ou médico-économiques, les meilleures performances opérationnelles demeurent souvent obtenues par des familles de modèles spécialisées : gradient boosting, forêts, modèles de survie, architectures temporelles dédiées, modèles sur graphes selon les cas. La littérature comparative évolue, mais les travaux disponibles montrent qu'il reste difficile de soutenir qu'un LLM, pris comme moteur principal, constituerait aujourd'hui le meilleur choix général pour des tâches tabulaires typiques [5].

Le point décisif n'est d'ailleurs pas seulement la performance brute. C'est aussi la possibilité de spécifier clairement les variables, de contrôler les jeux de features, d'inspecter les comportements du modèle, de documenter les pipelines d'entraînement et de produire des sorties compatibles avec des procédures d'évaluation clinique ou réglementaire. Dans ce cadre, demander à un LLM de porter seul une tâche de prédiction structurée revient souvent à mobiliser une capacité de médiation linguistique là où la tâche requiert avant tout une mécanique statistique explicite.

### 4.3 Calibration, incertitude et utilisabilité réglementaire

Dans un environnement régulé, une prédiction n'est utile que si son régime d'incertitude peut être caractérisé, testé et documenté. Les exigences opérationnelles portent moins sur la fluidité discursive du système que sur la qualité de ses sorties mesurables : calibration, discrimination, stabilité, sensibilité, spécificité, robustesse, dérive, reproductibilité.

Les LLM peuvent contribuer à l'explication de l'incertitude ou à la médiation de sorties probabilistes calculées ailleurs. En revanche, ils ne doivent pas être confondus avec la source de probabilités réputées calibrées et auditables. Autrement dit, ils peuvent **interpréter** un régime de confiance, mais ne doivent pas en être présumés le fondement statistique par simple effet d'interface.

### 4.4 Traçabilité et gouvernance des transformations

Dans un cadre régi par des exigences de documentation, de traçabilité et de gestion des mises à jour, le problème architectural ne se réduit pas à l'appel d'un modèle performant. Il implique aussi la provenance des données, les règles de transformation, le

versionnement des modèles, la documentation des choix de conception, les procédures de validation et la capacité à reconstituer une chaîne de décision.

À ce niveau, le centre de gravité du système se déplace vers la composition algorithmique, la mémoire de domaine et l'orchestration. Les cadres normatifs contemporains, qu'il s'agisse du règlement européen sur l'intelligence artificielle [6] ou des cadres de gestion des risques comme le NIST AI RMF [7], ne prescrivent pas une architecture donnée. Ils rendent néanmoins certaines propriétés architecturales plus plausibles et plus gouvernables que d'autres, notamment en matière de journalisation, de supervision humaine, de documentation et de contrôle des transformations.

## 5. Vers une architecture agentique composite

### 5.1 Principe général

Une architecture agentique composite repose sur trois éléments indissociables : un ensemble d'agents spécialisés hétérogènes, une mémoire de domaine persistante et une orchestration événementielle assurant la cohérence dynamique du tout.

Cette structure déplace le problème de l'orchestration de prompts vers l'ingénierie de composition. Le cœur de la difficulté n'est plus simplement de faire raisonner un modèle en plusieurs étapes, mais de faire coopérer des classes de calcul différentes, opérant sur des objets différents, avec des garanties différentes, dans un régime de persistance et de traçabilité compatible avec les exigences du domaine.

### 5.2 Hétérogénéité réelle et non seulement narrative

Une distinction doit être posée entre deux formes d'hétérogénéité.

Dans les architectures multi-agents purement LLM, l'hétérogénéité est souvent principalement narrative : plusieurs instances d'un même substrat computationnel sont spécialisées par prompt, rôle ou consigne. Cette approche a sa valeur. Elle permet une division du travail conversationnel ou symbolique.

Dans une architecture composite au sens fort, l'hétérogénéité est **mathématique et computationnelle**. Les composants ne diffèrent pas seulement par leur instruction, mais par la nature des problèmes qu'ils traitent et par les garanties qu'ils peuvent fournir. Un générateur tabulaire, un modèle de survie, un réseau sur graphes, un moteur d'optimisation, un module de calibration et un LLM n'occupent pas des rôles rhétoriques différents sur un même substrat. Ils appartiennent à des régimes de calcul distincts.

### 5.3 Exemple de répartition fonctionnelle

Dans un contexte de jumeau numérique en santé, on peut distinguer plusieurs familles d'agents spécialisés.

Des **agents générateurs** ont pour fonction de produire des populations synthétiques préservant, dans une mesure évaluée, certaines propriétés statistiques de la cohorte réelle. Des architectures telles que CT-GAN ou TVAE remplissent cette fonction.

Des **agents prédictifs** ont pour fonction de produire des prédictions calibrées sur des variables de résultat clinique ou organisationnel. Ils peuvent relever de modèles de gradient boosting, de modèles de survie, de modèles bayésiens ou d'autres approches adaptées au problème.

Des **agents de deep learning spécialisé** opèrent sur des structures que le langage n'épuise pas : graphes moléculaires, séries temporelles complexes, représentations sous contraintes, etc. Les GNN en toxicologie moléculaire ou certaines familles de réseaux informés par contraintes cliniques relèvent de ce niveau.

Enfin, un **agent d'interprétation et d'interface**, souvent fondé sur un LLM, consomme les sorties structurées des agents précédents pour les rendre intelligibles à des utilisateurs humains, contextualiser des alertes, reformuler des résultats, expliciter des limites ou adapter le registre d'énonciation au destinataire.

Le point central est ici fonctionnel : le LLM ne disparaît pas, mais il cesse d'être présumé constituer à lui seul le moteur computationnel du système.

## 6. La mémoire de domaine : au-delà du contexte conversationnel

### 6.1 Un essaim sans mémoire n'est qu'une chaîne de traitement

Les agents spécialisés décrits ci-dessus ne peuvent pas fonctionner en isolation. Un générateur synthétique suppose une cohorte source. Un modèle prédictif suppose des états consolidés et versionnés. Un interprète linguistique suppose un contexte de trajectoire, d'historique ou de comparaison.

L'architecture composite requiert donc une **mémoire de domaine**, entendue non comme simple stockage de données, mais comme l'ensemble des structures par lesquelles un système conserve, rend accessibles, transforme et réutilise des informations pertinentes au fil du temps.

Cette mémoire comprend au minimum : données brutes et horodatages, états consolidés, features construites, sorties intermédiaires, modèles entraînés et leurs versions, artefacts de validation, traces décisionnelles, métadonnées de provenance et informations de qualité.

### 6.2 Le Medallion canonique : un axe de maturation qualitative

L'architecture Medallion de type Bronze / Silver / Gold repose canoniquement sur un axe de maturation qualitative des données : brut, nettoyé, orienté exploitation. Cet axe n'est

pas en soi un axe de temps. Dans un entrepôt analytique classique, on peut parfaitement avoir du Gold très récent et du Bronze très ancien. Il faut donc éviter l'erreur naïve consistant à assimiler immédiatement Medallion et mémoire temporelle.

### 6.3 Asymétrie de persistance fonctionnelle

Dans une classe plus spécifique de systèmes, en particulier les jumeaux numériques en ingestion continue, une propriété supplémentaire peut toutefois apparaître. Nous proposons de nommer **asymétrie de persistance fonctionnelle** la différence de durée de vie opérationnelle des couches, différence qui se superpose alors partiellement à l'axe canonique de qualité.

Le Bronze reçoit les signaux bruts, événements et observations élémentaires. Ces objets conservent une utilité technique durable pour l'audit, le replay ou la réconciliation. Leur **fonction cognitive directe** dans le fonctionnement courant du jumeau est cependant souvent brève : une fois consolidés, ils cessent généralement d'être les unités principales de décision.

Le Silver correspond à des états consolidés, nettoyés, structurés, comparables entre eux. Ces objets demeurent opératoires plus longtemps. Ils soutiennent les mises à jour des profils, les recalculs de features, les réévaluations de trajectoire, les comparaisons longitudinales.

Le Gold rassemble des artefacts plus stabilisés : cohortes validées, modèles versionnés, indicateurs consolidés, seuils calibrés, connaissances dérivées du domaine. Ces objets ont généralement la plus grande persistance fonctionnelle. Ils s'accumulent, se révisent lentement et constituent le socle de la mémoire longue du système.

Nous ne prétendons pas que cette asymétrie soit universelle. Nous soutenons qu'elle apparaît dans certains systèmes réunissant au moins trois conditions : ingestion continue, objet persistant dont l'identité s'accumule dans le temps, et différence marquée de durée de vie fonctionnelle entre couches.

### 6.4 Lecture structurale en termes de mémoire

Lorsque ces conditions sont réunies, une lecture structurale peut être proposée à partir des travaux de Tulving [9] et de Baddeley [10].

La couche Bronze peut être comprise comme occupant, dans l'économie cognitive du système, une fonction dominante proche d'une **mémoire de travail** ou d'un tampon de traitement : forte proximité avec la perception, utilité immédiate, faible durée de vie fonctionnelle directe malgré une persistance technique pour audit.

La couche Silver peut être lue comme une forme de **mémoire épisodique opérationnelle** : elle maintient des états consolidés, contextualisés temporellement,

exploitables pour suivre des trajectoires et reconstruire des séquences de transformation.

La couche Gold peut être comprise comme une **mémoire sémantique de domaine** : elle cristallise des régularités, des modèles, des seuils et des connaissances relativement décontextualisées par rapport à l'épisode brut d'acquisition.

Cette correspondance ne vaut pas comme identité ontologique entre psychologie cognitive et architecture de données. Elle vaut comme **homologie fonctionnelle** portant sur le rôle dominant des couches dans l'économie du système. C'est une distinction importante. Les humains aiment confondre analogie et preuve, puis s'étonnent de fabriquer des métaphores qui mordent.

### 6.5 Conséquence architecturale

Dans les systèmes où cette asymétrie est effectivement présente, une économie architecturale significative apparaît. Le pattern Medallion ne sert plus seulement à organiser la qualité des données ; il devient aussi le support principal de la mémoire du jumeau. Autrement dit, la maturation qualitative de la donnée et sa cristallisation progressive en connaissance peuvent relever, sous conditions, d'un même dispositif architectural observé sous deux angles différents.

Cette proposition ne dispense pas de penser des mécanismes complémentaires de consolidation, d'oubli sélectif, de hiérarchisation ou de gouvernance des représentations. Elle indique simplement qu'il n'est pas toujours nécessaire d'ajouter une couche mémorielle entièrement séparée pour obtenir une mémoire de domaine opératoire.

### 6.6 Différence avec la mémoire de session d'un LLM

La différence avec le contexte d'un LLM est ici qualitative. Le contexte conversationnel est une mémoire de session, bornée, linéaire, essentiellement textuelle. Une mémoire de domaine est stratifiée, multimodale, versionnée, persistante, réinscriptible, et adossée à des politiques de qualité, de sécurité et d'audit. L'une sert l'interaction. L'autre soutient la continuité opératoire du système.

## 7. L'orchestration événementielle comme logique de circulation

L'architecture composite ne peut fonctionner sans mécanisme de coordination. C'est la fonction du substrat événementiel.

Les événements jouent ici plusieurs rôles. Ils déclenchent des traitements. Ils relient des composants sans couplage excessif. Ils rendent visibles les transitions du système. Ils

assurent la traçabilité temporelle. Ils permettent une réactivité graduée selon le type de signal, le niveau de criticité et la politique de délégation.

Dans un jumeau numérique, un événement brut peut alimenter la couche Bronze. Sa consolidation peut produire un état Silver actualisé. Cet état peut à son tour déclencher une réévaluation d'un risque, la mise à jour d'un artefact Gold, puis éventuellement l'émission d'une alerte interprétée par un LLM ou transmise à un opérateur humain. L'EDA n'est donc pas seulement un bus de communication. Elle constitue la logique de circulation des états, des déclencheurs et des décisions dans un système agentique composite.

Le triptyque devient alors clair : des agents spécialisés hétérogènes, opérant sur une mémoire structurée, coordonnés par un substrat événementiel. C'est cette articulation qui rend possible un jumeau numérique réellement opératoire.

## 8. Pourquoi le paradigme download-and-deploy est insuffisant

Une grande partie du marché repose sur une logique simple : sélectionner un modèle pré-entraîné, le fine-tuner ou l'encapsuler, puis l'intégrer à un framework d'orchestration. Cette logique fonctionne dans de nombreux cas. Elle devient insuffisante lorsque l'enjeu ne porte plus sur l'accès à un modèle, mais sur la **composition validée d'un système hétérogène de calcul**.

Les composants critiques de l'architecture composite ne sont souvent pas des modèles de fondation généralistes importables tels quels. Ils doivent être construits, paramétrés, validés et documentés à partir des distributions propres au domaine, du schéma de données, des contraintes métier et des exigences d'évaluation.

Le défi principal n'est donc plus de choisir un modèle, mais d'assurer la compatibilité, l'enchaînement, la validation croisée, le versionnement et la traçabilité de plusieurs classes de modèles au sein d'une même plateforme opératoire.

C'est ce que nous appelons ici un problème de **composition algorithmique**.

## 9. Terrains d'illustration : TweenMe et PREDICARE

Les considérations précédentes peuvent être illustrées par deux systèmes issus de nos travaux. Ils sont mobilisés ici comme **terrains d'implémentation** et non comme démonstration générale autosuffisante de la thèse.

### 9.1 TweenMe comme plateforme de composition

TweenMe peut être décrit comme une plateforme de génération de jumeaux numériques fondée sur la composition d'algorithmes spécialisés calibrés sur les données d'un domaine. Le pipeline général comprend : ingestion et profilage de cohorte, génération synthétique, validation statistique, construction de modèles prédictifs, couche d'interprétation, puis déploiement sur une infrastructure de coordination.

Dans le cadre de l'étude OCTOPUS en oncologie, le pipeline a obtenu un score TSTR élevé sur le protocole retenu. Un tel résultat ne suffit pas, à lui seul, à conclure à une indistinguabilité statistique générale entre cohorte réelle et cohorte synthétique. Il constitue en revanche un indice fort de **fidélité opérationnelle** pour les tâches aval considérées dans ce cadre d'évaluation. Le point important, pour la présente discussion, n'est pas la célébration d'un chiffre isolé, mais le fait que ce type de résultat repose sur une chaîne de composition et de validation, et non sur le téléchargement d'un modèle unique.

### 9.2 PREDICARE comme instanciation territoriale

Le programme PREDICARE/Sentinelle IA permet quant à lui d'illustrer l'architecture composite à l'échelle d'un territoire de santé. L'idée centrale n'est pas de déployer un unique modèle prédictif, mais d'organiser plusieurs jumeaux spécialisés ou sous-systèmes orientés vers des fonctions différentes : évolution individuelle, dynamique démographique, allocation territoriale, anticipation d'événements, efficience médico-économique.

Ce terrain n'est pas invoqué ici comme preuve exhaustive d'efficacité clinique finale. Il est mobilisé comme **matérialisation architecturale** de la thèse : un tel système ne peut pas être pensé comme un chatbot enrichi. Il exige un couplage entre calcul spécialisé, mémoire de domaine et orchestration événementielle.

## 10. Discussion et limites

Plusieurs précautions s'imposent.

Premièrement, cette thèse ne vaut pas pour tous les agents. Il existe des classes d'agents pour lesquelles le LLM peut légitimement demeurer le composant central : assistance documentaire, support conversationnel, workflows bureautiques, médiation logicielle, coordination légère de tâches.

Deuxièmement, l'opposition entre LLM et modèles spécialisés ne doit pas être absolutisée. Dans de nombreux systèmes réels, le LLM peut agir comme interface de pilotage, couche de supervision secondaire, mécanisme de traduction inter-modèles ou support d'explicitation. La question n'est donc pas son exclusion, mais sa juste localisation dans l'architecture.

Troisièmement, la notion d'asymétrie de persistance fonctionnelle mérite un approfondissement théorique et empirique. Elle doit être testée sur d'autres configurations que celles qui l'ont inspirée, et ne doit pas être étendue mécaniquement à tout lakehouse ou à tout jumeau numérique.

Quatrièmement, la lecture structurale inspirée de Tulving doit être maniée avec prudence. Elle ne vaut pas comme importation naïve d'une psychologie de la mémoire humaine dans l'architecture des données. Elle propose un cadre fonctionnel d'interprétation dont l'intérêt tient à sa capacité explicative et non à une prétention d'identité forte entre les deux registres.

Cinquièmement, l'argument réglementaire ne doit pas être simplifié. Les cadres normatifs n'imposent pas mécaniquement une architecture donnée. En revanche, certaines architectures rendent la conformité, la validation et l'audit beaucoup plus plausibles que d'autres.

Enfin, le champ évolue rapidement. Les capacités des modèles de fondation sur certaines tâches structurées progressent. La thèse défendue ici porte sur l'état actuel du problème et sur les propriétés architecturales requises dans les environnements visés. Même si le centre de gravité computationnel de certains sous-systèmes devait évoluer, la nécessité d'une mémoire de domaine persistante et d'une orchestration événementielle demeurerait.

## 11. Conclusion

Le centre de gravité de l'évaluation d'un système agentique se déplace. Dans un paradigme LLM-centré, la qualité d'un agent tend à se mesurer à la cohérence verbale de son raisonnement : capacité à décomposer un problème, à planifier, à répondre de manière crédible. Dans une architecture composite, l'évaluation se déplace vers la qualité mesurable du calcul, de la mémoire et des transitions : fidélité des cohortes synthétiques pour les usages considérés, calibration des prédictions, robustesse des chaînes de transformation, qualité de la mémoire de domaine, auditabilité, reproductibilité et intégrabilité organisationnelle.

Le besoin stratégique non couvert est donc celui d'une plateforme qui ne commence pas par un LLM comme hypothèse de conception, mais qui soit capable de finir par un LLM lorsque la médiation linguistique devient utile. Une telle plateforme prend en entrée des données de domaine structurées et produit en sortie un système hétérogène de calcul, de mémoire et d'action, déployable sur un substrat événementiel.

Dans les environnements régulés à forte composante tabulaire, temporelle et décisionnelle, l'agentique utile ne se réduit pas à l'orchestration d'un modèle de langage. Elle tend à prendre la forme d'une architecture composite articulant calcul spécialisé,

mémoire de domaine persistante et coordination événementielle. Dans ces environnements, l'agent n'est pas un interlocuteur rendu autonome. C'est un système de calcul, de mémoire et d'action : Un tel système ne se télécharge pas. Il se compose.

*Cet article fait suite à « L'architecture événementielle comme complément essentiel de l'IA agentique » (Twingital Institute, 2026). L'auteur est VP R&D & CPO chez Qualees et fondateur du Twingital Institute.*

**Références**


[1] J. Vetillard, « L'architecture événementielle comme complément essentiel de l'IA agentique : De la cognition déléguée à l'action située, traçable et gouvernable en environnement d'entreprise », Twingital Institute, 2026.

[2] F. P. Brooks, *The Mythical Man-Month: Essays on Software Engineering*, Addison-Wesley, 1975.

[3] M. Wooldridge et N. R. Jennings, « Intelligent Agents: Theory and Practice », *The Knowledge Engineering Review*, vol. 10, no. 2, pp. 115–152, 1995.

[4] L. Xu, M. Skoularidou, A. Cuesta-Infante et K. Veeramachaneni, « Modeling Tabular Data using Conditional GAN », *Advances in Neural Information Processing Systems*, 2019.

[5] L. Grinsztajn, E. Oyallon et G. Varoquaux, « Why do tree-based models still outperform deep learning on typical tabular data? », *Advances in Neural Information Processing Systems*, 2022.

[6] Règlement (UE) 2024/1689 du Parlement européen et du Conseil du 13 juin 2024 établissant des règles harmonisées concernant l'intelligence artificielle.

[7] National Institute of Standards and Technology, « Artificial Intelligence Risk Management Framework (AI RMF 1.0) », NIST AI 100-1, 2023.

[8] J. Vetillard, « Clinically-Informed Neural Networks », note doctrinale de travail, Twingital Institute, 2025.

[9] E. Tulving, « Episodic and semantic memory », in *Organization of Memory*, Academic Press, 1972, pp. 381–403 ; E. Tulving, « How many memory systems are there? », *American Psychologist*, vol. 40, no. 4, pp. 385–398, 1985.

[10] A. D. Baddeley, « Working memory », *Science*, vol. 255, no. 5044, pp. 556–559, 1992 ; A. D. Baddeley, « The episodic buffer: a new component of working memory? », *Trends in Cognitive Sciences*, vol. 4, no. 11, pp. 417–423, 2000.



**Jérôme Vetillard**
CTO | VP R&D | Chief Product Officer | AI-Powered Healthcare & Life Sciences Products | Compliance by Design
PhD AgroParisTech | CPO MIT Sloan | Exec MBA IE Business School & Brown University
Twingital-institute / Twingital-ventures : twingital-ventures.com