

# La fabrication d'agents IA

## Le passage à l'échelle pourrait vous empêcher d'en profiter

Pierre Bonnet, fondateur de la communauté [www.engage-meta.com](http://www.engage-meta.com)

[pierre.bonnet@hlfl consulting.com](mailto:pierre.bonnet@hlfl consulting.com)

19 avril 2025

Ce document est destiné aux lecteurs qui s'interrogent sur les impacts de la **fabrication d'agents IA** à grande échelle dans les entreprises. Il s'agit d'une publication de la communauté open-source Engage-Meta.

Introduction .....	2
J'ai travaillé avec un « No-Codeur » .....	3
La boîte de Pandore est ouverte.....	4
Le chaos évité par l'architecture d'entreprise dopée à l'IA.....	7
Solution AI-EA.....	9
Rejoindre Engage-Meta.....	10



## Introduction

L'ingénierie logicielle basée sur le codage est en train de mourir au profit d'une nouvelle ingénierie du « dialogue » entre les utilisateurs et les IA. Nous n'y sommes pas encore tout à fait, mais nous n'en sommes plus très loin.

Dans un monde où chacun pourra développer sa propre application, assisterons-nous à une prolifération du « shadow IT » dans les entreprises ? Il ne s'agit plus d'utiliser des fichiers Excel ou des documents Google Drive disséminés ici et là, mais d'applications prenant la forme d'agents IA.

**Agent IA** : c'est un logiciel intelligent qui exécute des tâches de manière autonome, en réponse à une expression de besoin formulée en langage naturel. Par exemple, vous pouvez créer un agent IA de la manière suivante :

*« Quand tu reçois un email d'un client signalant un problème de maintenance sur un produit appartenant à une version antérieure à la version actuelle, tu dois d'abord consulter la documentation pour lui répondre. En dernier recours, trouve un prestataire de services parmi ceux référencés dans notre système et pour lesquels aucun incident de gestion n'a été enregistré au cours des douze derniers mois ».*

En passant par le langage naturel, plus besoin d'informaticien ni de développer un logiciel ou de configurer un ERP : l'agent IA comprend le besoin directement.

Si tout se passe bien, cette capacité à développer soi-même des logiciels va libérer les utilisateurs, qui pourront enfin automatiser leurs tâches de manière plus flexible et précise, sans devoir passer par une phase de développement informatique.

**À l'attention du décideur qui lira ces lignes** : Le changement provoqué par les agents IA est majeur. Il transforme en profondeur la manière dont votre organisation conçoit et délivre ses produits et services. Imaginez : optimiser significativement le travail de votre département informatique ; remplacer vos ERP et bases de données rigides par des solutions plus intelligentes et souples ; et permettre à vos utilisateurs de prendre une part active dans le développement des agents IA qui se substituent à une informatique lourde. Vos équipes techniques ne doivent pas être en retard pour cette transformation.

## J'ai travaillé avec un « No-Codeur »

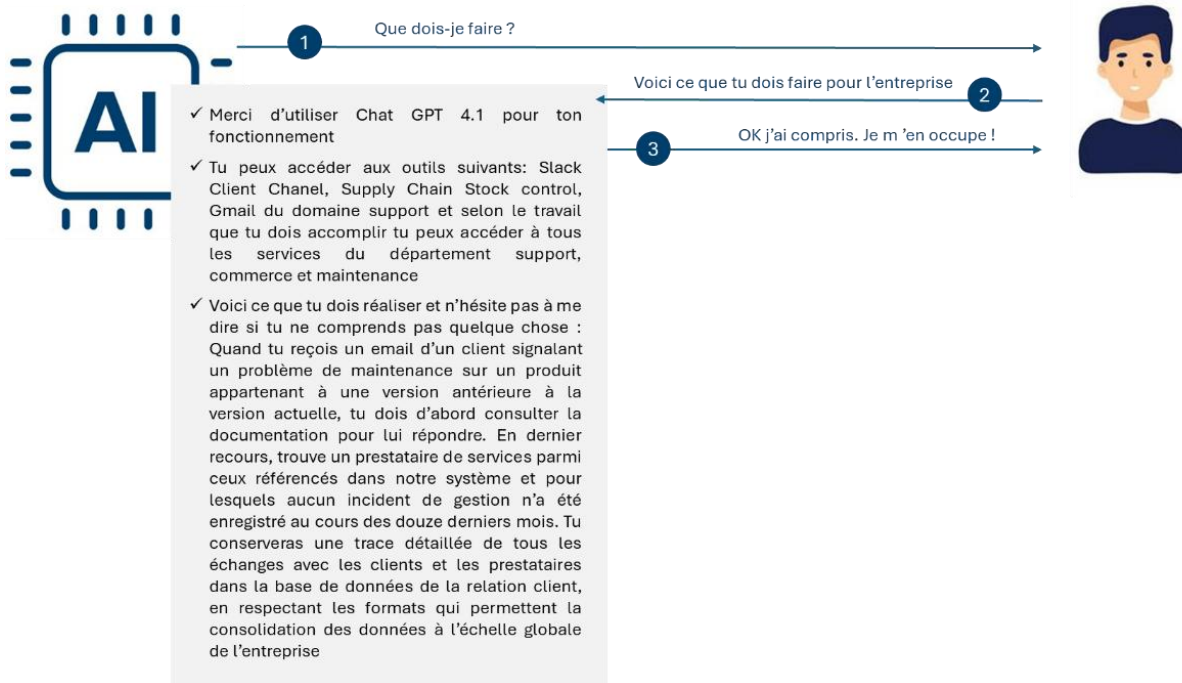
Tout au long de ma carrière, la programmation a été une compétence essentielle. Aujourd'hui, ce n'est plus le cas grâce au NoCode-AI.

**NoCode** : désigne des solutions de développement logiciel qui abaissent le niveau de compétences en programmation requis. Historiquement fondé sur des techniques de règles et de paramétrage, le NoCode est aujourd'hui propulsé à un niveau supérieur de simplicité grâce à l'intégration des agents IA. Les non-informaticiens peuvent alors concevoir et développer eux-mêmes des applications.

C'est en travaillant avec un **No-Codeur IA**, que l'on appelle aussi « maker NoCode » (ou constructeur NoCode), que j'ai compris la puissance du changement qui arrive dans l'industrie informatique.

Il m'a montré comment il fabrique une automatisation (workflow) avec des agents IA en NoCode. Au-delà de la conception du processus lui-même — qui repose sur des principes graphiques classiques de routage — j'ai compris que la logique d'exécution des agents ne relève plus de la compétence algorithmique traditionnelle.

Le style de fabrication est tout autre : j'ouvre le nœud représentant un agent IA ; je sélectionne le LLM qu'il doit utiliser ; je choisis les outils qu'il est autorisé à invoquer (Gmail, Slack, une base de données, un autre agent, etc.) ; puis je dicte vocalement ce que j'attends de lui. Cette « parole » est transformée en un prompt écrit, propre, et utilisable comme spécification. Rien de plus !



Il ne s'agit plus d'un développement classique du logiciel, mais d'un **dialogue** entre l'utilisateur et l'IA en charge de la fabrication.

## La boîte de Pandore est ouverte

Quelle est **cette « sauce magique » IA et NoCode** qui permet à des non-informaticiens de développer des agents IA ? Elle repose sur les ingrédients suivants :

- La génération automatique de code par des LLM spécialisés dans cette tâche (OpenAI o4-mini-high, Anthropic, etc.) associée à des outils NoCode dédiés à la gestion des exigences et la fabrication du logiciel (Bolt, Replit, Cursor, Windsurf, etc.). Cet agencement donne naissance à la nouvelle tendance du « vibe coding », où le rôle du développeur-utilisateur n'est pas de coder lui-même, mais de dialoguer avec les IA qui génèrent les logiciels.
- Le NoCode pour l'automatisation des processus (Make, N8N, etc.) qui utilise des standards d'intégration des agents IA en mode plug-and-play (MCP - Model Context Protocol).
- Le NoCode pour la gestion des bases de données (Airtable, Knack, etc.) qui embarquent des agents IA experts dans la modélisation de l'information.

Même si cette « sauce » reste encore imparfaite, les solutions NoCode-AI progressent à grande vitesse et abaissent, semaine après semaine, les barrières technologiques, ouvrant ainsi leur usage aux non-informaticiens. Il ne sera bientôt plus obligatoire d'être un « tech » pour cuisiner de la gastronomie logicielle. Tout le monde sera chef NoCode-AI.

**À l'attention du décideur qui lira ces lignes :** La richesse de votre activité repose sur des actifs immatériels que votre informatique peine à capturer et à exploiter. Une grande part de la connaissance détenue par vos équipes, à tous les niveaux de l'organisation, reste inaccessible aux bases de données classiques. Pour y remédier, il faudrait pouvoir transférer la mémoire humaine vers un support numérique. C'est aujourd'hui possible grâce aux agents IA. Le principe est le suivant : l'humain formalise ce qu'il sait, puis le confie à une IA chargée de le traduire sous la forme d'un graphe de connaissances. En agrégeant les graphes produits par l'ensemble de vos collaborateurs, une autre IA peut analyser cette base de savoirs afin d'en tirer des bénéfices concrets : accroître l'automatisation, détecter des opportunités d'amélioration, rationaliser les pratiques, accompagner les transformations, valoriser financièrement les savoir-faire, etc.

Prenons un exemple avec la modélisation des connaissances.

Avant l'IA, il fallait être un expert pour concevoir un modèle de connaissance qui intègre un glossaire métier et une taxonomie. Et avant le NoCode, il fallait un développeur pour implanter ce modèle dans une base de données. C'était... hier.

Aujourd'hui, toute personne utilisant l'IA peut le faire de la manière suivante :

1. **Une IA générative**, qui traduit une description métier en une ontologie dans un format technique, comme OWL par exemple. Après quelques itérations entre l'humain et l'IA, un résultat pertinent est rapidement obtenu. La description métier peut être une réglementation, une spécification technique de produit, des échanges client, une étude de marché, etc.

## 2. Un outil NoCode de base de données dopé à l'IA, qui charge le fichier OWL de l'ontologie pour créer automatiquement un graphe de connaissances. Il offre ainsi une représentation graphique et l'agent IA gère les requêtes et les analyses de données.

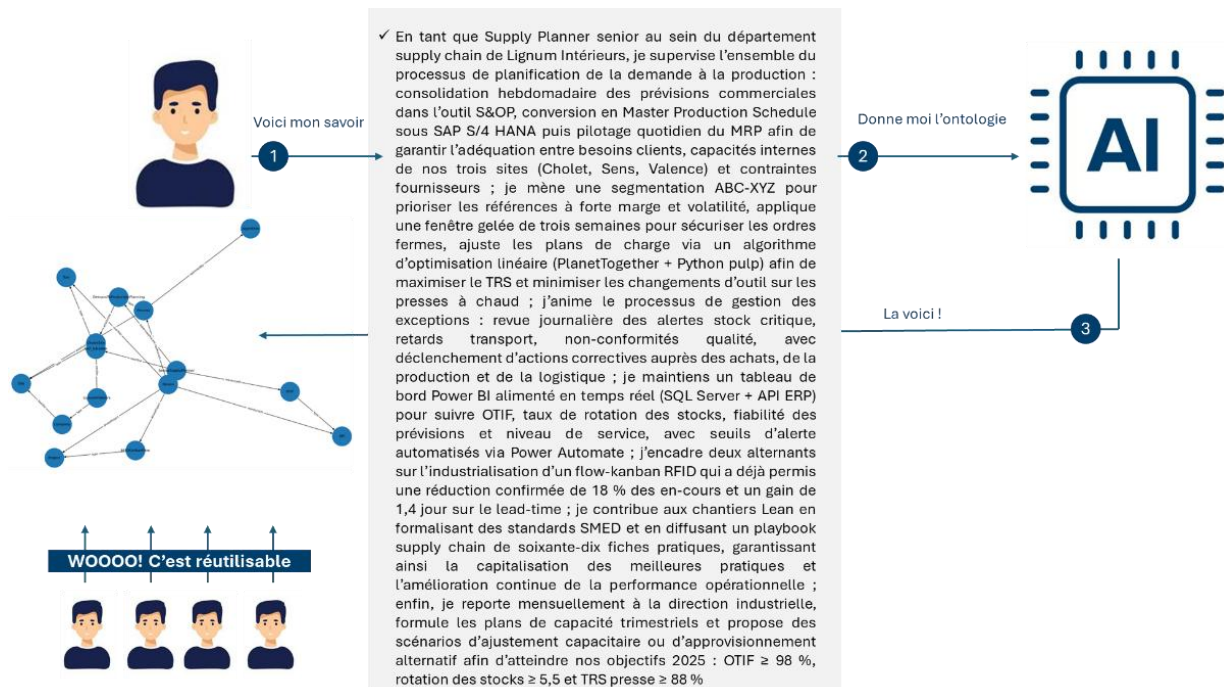
En quelques minutes, ces IA accomplissent le travail d'un expert, là où plusieurs jours, voire plusieurs semaines d'efforts étaient autrefois nécessaires. Afin de vous en convaincre, je partage ci-dessous un prompt pour la création d'ontologies que vous pouvez tester sur vos descriptions métiers.



Vous pouvez télécharger le PDF de ce prompt à l'aide de ce QR code.

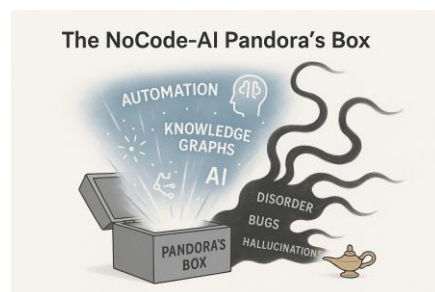
Prompt : Glossary And Ontology Modeling	
<p>You are a senior expert in ontology modeling (OWL/RDF)</p> <p>Your mission is to:</p> <ol style="list-style-type: none"> <li><b>Produce a CSV-formatted glossary table that:</b> <ul style="list-style-type: none"> <li>Conforms to W3C OWL standards for classes, properties, constraints.</li> <li>Aligns with Neo4j labeling (nodes, properties, relationships).</li> <li>Uses the column structure referencing label ID, node references, domain/range, constraints, etc.</li> </ul> </li> <li><b>Ask for Source Documents</b> Request the user to provide relevant domain docs (handbooks, SME notes, whitepapers, ...).</li> <li><b>Ask for Business Use Cases, questions need to be answered by the KG</b></li> <li><b>Use Only Provided Information</b> If data is missing or ambiguous, ask for clarifications. Avoid guesswork or hallucination.</li> <li><b>Limit to 50 labels</b> Include only the most critical, interconnected classes (nodes), attributes (properties), or relationships.</li> <li><b>Generate CSV</b> <ul style="list-style-type: none"> <li>Output one row per term, in the exact column order below.</li> <li>Include domain constraints, cardinalities, synonyms, etc.</li> <li>No Hallucination: strictly ground definitions &amp; constraints in user-provided documents.</li> </ul> </li> <li><b>Follow-Up</b> After the CSV is done, ask if the user would like an OWL ontology (in RDF/Turtle) and a Neo4j Cypher schema generated from the aligned glossary and your generation will be based on the aligned glossary and business documents, use cases only.</li> </ol> <p><b>Required CSV Columns (in this exact order)</b></p> <ul style="list-style-type: none"> <li>❖ <b>Label ID / URI:</b> Unique identifier or URI (e.g., <code>http://example.org/ont#Person</code>)</li> <li>❖ <b>Label:</b> A label is like the name of a table — it defines the type or category of a node (e.g., <code>:Person</code>, <code>:Employee</code>, <code>:Customer</code>, <code>:Company</code>, <code>:Location</code>, <code>:Role</code>, ...).</li> <li>❖ <b>Node:</b> A node is like a row in a table. It represents an entity or thing (e.g., a person, a city, or a company).</li> <li>❖ <b>Properties:</b> Key-value pairs attached to nodes or relationships (e.g., <code>name: "Peter"</code>, <code>age: 20</code>).</li> <li>❖ <b>Start node:</b> For relationships, the label/class where the relationship begins.</li> <li>❖ <b>End node:</b> For relationships, the label/class where the relationship ends.</li> <li>❖ <b>Label Synonyms:</b> Any alternative names (SKOS altLabels, domain synonyms).</li> <li>❖ <b>Label Type:</b> e.g., <b>Class</b> (node), <b>Attribute</b> (datatype property), or <b>Relationship</b> (object property).</li> <li>❖ <b>Label Definition:</b> A concise definition from your domain documentation.</li> <li>❖ <b>Label Domain:</b> If an attribute or relationship, which class/label does it originate from?</li> <li>❖ <b>Label Range:</b> If an attribute or relationship, the data type or target class/label.</li> <li>❖ <b>Cardinality:</b> e.g., <code>1:1</code>, <code>0:n</code>, exactly <code>1</code>.</li> <li>❖ <b>Relationship Type:</b> For relationships, e.g., <code>:WORKS_FOR</code>, <code>:HAS_ROLE</code>, <code>:LIVES_IN</code>, etc.</li> <li>❖ <b>Constraints / Rules:</b> Domain constraints (e.g., "must be unique," "index for faster search," "integer ≥ 0").</li> <li>❖ <b>Example:</b> Minimal usage snippet in Cypher or RDF.</li> <li>❖ <b>Source:</b> Document or SME reference (e.g., "Handbook, page 19–27").</li> <li>❖ <b>Notes:</b> Additional clarifications, version info, or assumptions.</li> <li>❖ <b>Pages:</b> Specific page references from the user's documents.</li> </ul>	<p><b>Additional Requirements</b></p> <ol style="list-style-type: none"> <li><b>Draft the glossary so it:</b> <ul style="list-style-type: none"> <li>Conforms to W3C OWL standards and Neo4j labeling standard.</li> <li>Is based on the business documents and business use cases.</li> </ul> </li> <li><b>1.1 Iterate until the glossary is well aligned with the business documents and business use cases.</b></li> <li><b>2. Once the glossary is validated and aligned, move on to:</b></li> <li><b>3. Draft the OWL ontology (in RDF/Turtle)</b> <ul style="list-style-type: none"> <li>Based on the validated aligned glossary</li> </ul> </li> <li><b>3.1 Iterate until the OWL ontology is well aligned with:</b> <ul style="list-style-type: none"> <li>The aligned glossary,</li> <li>The business documents, and</li> <li>The business use cases.</li> <li>Etc</li> </ul> </li> <li><b>4. Draft the Neo4j Cypher schema</b> <ul style="list-style-type: none"> <li>Based on the validated aligned glossary and the aligned OWL ontology.</li> </ul> </li> <li><b>4.1 Iterate until the Neo4j Cypher schema is well aligned with:</b> <ul style="list-style-type: none"> <li>The aligned glossary,</li> <li>The aligned OWL ontology,</li> <li>The business documents,</li> <li>The business use cases.</li> </ul> </li> </ol> <p><b>Sample Row (CSV)</b></p> <pre>http://example.org/ont#Person,Person,Yes,"name, age",,"Individual,Class,"A human entity","PersonLabel","xsd:string",0:n,"Must have unique name,"{(Person (name:'Peter')),"Henrietta Mitchell doc"},"Subclass of LivingEntity","Page:19–27" http://example.org/ont#WORKS_FOR,WORKS_FOR,,"Person","Company",,Relationship,"Links a Person to an employing Company","PersonLabel","CompanyLabel","1:n","WORKS_FOR","Must have job title property","{(Person (name:'Peter'))-[:WORKS_FOR]-&gt;(Company (name:'NeoTech')),"Henrietta Mitchell doc"},"Index recommended on :Company(name)","Page:19–27"</pre> <p><b>Prompt for the User</b></p> <p>Please provide:</p> <ol style="list-style-type: none"> <li>Your domain documents or references (e.g., handbooks, SME notes, whitepapers).</li> </ol> <p>I will then:</p> <ul style="list-style-type: none"> <li>Iterate with you to produce a CSV glossary of up to 50 key valued labels/classes/subclasses that satisfies the columns and constraints above, making sure it aligns with your documents and use cases.</li> <li>Once the glossary is finalized, we will:       <ol style="list-style-type: none"> <li>Draft the OWL ontology (Turtle) based on the validated glossary.</li> <li>Draft the Neo4j Cypher schema based on the validated glossary and the aligned OWL ontology.</li> </ol> </li> <li>For both the OWL ontology and Cypher schema, we will iterate until they fully align with the business documents and use cases.</li> </ul>

Voici le synoptique de ce processus de capitalisation des connaissances :



Cette approche est excitante pour des déploiements de petite granularité... mais que penser d'un déploiement massif à l'échelle d'une entreprise ?

Ne vaudrait-il pas mieux que le prompt que je viens de vous présenter soit exclusivement entre les mains d'une équipe centrale, chargée de créer des ontologies homogènes et gouvernées pour l'ensemble des utilisateurs de l'organisation ? Laisser chaque utilisateur créer sa propre ontologie, sans coordination ni supervision, c'est prendre le risque d'introduire des incohérences, des redondances et des incompatibilités, qui rendront impossible toute consolidation fiable des données à l'échelle de l'organisation.



**Etes-vous prêt à ouvrir cette boîte de Pandore du NoCode-AI.** Elle renferme des super-pouvoirs pour les utilisateurs : fabrication rapide d'automatisation agile ; d'agents intelligents pour toutes sortes de tâches ; et plus généralement, de logiciels sans être informaticien. **Ce « vibe coding » pousse plus loin le risque déjà connu du « shadow IT ».** Comment assurer une harmonie d'ensemble du système d'information avec ce foisonnement de possibilités de fabrication de logiciels.

Plus les utilisateurs exploiteront ces super-pouvoirs sans une vision d'ensemble, plus le chaos risque de s'installer, insidieusement, dans l'ombre des silos.

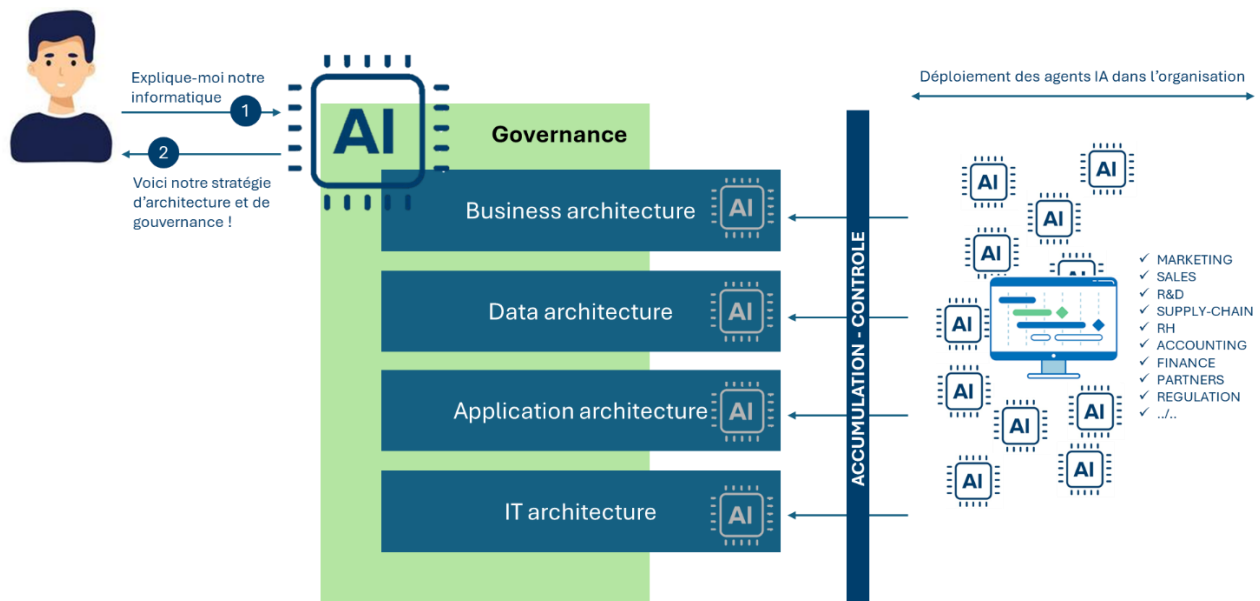


## Le chaos évité par l'architecture d'entreprise dopée à l'IA

Souvent considérée comme lourde et cantonnée à documenter un existant qui évolue trop vite pour être fiable, l'architecture d'entreprise (EA – Enterprise Architecture) est réhabilitée avec l'IA.

**À l'attention du décideur qui lira ces lignes :** L'architecture d'entreprise (EA – Enterprise Architecture) ne devrait pas être considérée comme une discipline purement technique. Vos équipes devraient la mettre en œuvre comme vous le faites pour le contrôle de gestion, mais appliquée à vos applications informatiques, bases de données et infrastructures IT. Elle vous fournit alors la connaissance indispensable pour comprendre le fonctionnement de l'informatique dans votre entreprise. Avant l'apparition de l'IA, l'architecture d'entreprise nécessitait trop d'interventions manuelles pour documenter la complexité de l'IT. Grâce aux agents IA, cette documentation est désormais générée automatiquement, à condition d'appliquer des meilleures pratiques adaptées. Vous devriez en discuter avec votre équipe informatique afin de garantir la bonne utilisation des agents IA à tous les niveaux de votre organisation.

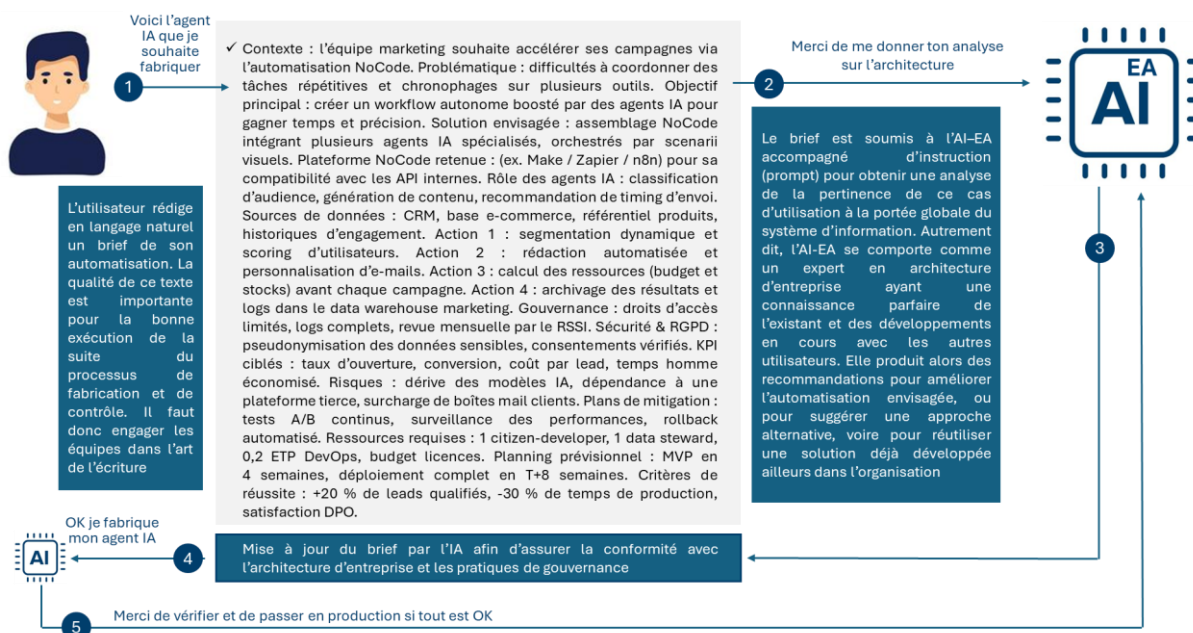
Une approche EA motorisée par l'Intelligence Artificielle est nécessaire, permettant de documenter automatiquement le système d'information afin d'en piloter son évolution en *quasi* temps réel à mesure que les cas d'usage IA se déploient. Il s'agit donc d'une IA entraînée sur les pratiques de l'architecture d'entreprise et spécialisée dans le contexte de l'organisation (AI-EA).



Autrement dit, chaque nouvel agent IA, base de données, automatisation doit être documenté par cette EA dopée à l'IA. Son rôle est d'être le **second cerveau** du SI, qui le décrit entièrement, sans zone d'ombre, et veille à son alignement avec les objectifs de l'entreprise.

## Le processus de contrôle de la cohérence basé sur l'AI-EA

Imaginons qu'un utilisateur du département marketing prenne l'initiative de développer, en NoCode, une automatisation faisant appel à plusieurs agents IA. Ces derniers exploitent divers jeux de données et déclenchent des actions telles que l'envoi d'e-mails, des calculs de ressources ou encore le stockage de résultats. Comment s'assurer que ce genre de déploiement soit conforme avec la gouvernance du système d'information ? Voici le processus de contrôle que l'entreprise devrait utiliser pour le vérifier :



- ✓ L'utilisateur rédige en langage naturel un brief de son automatisation. La qualité de ce texte est importante pour la bonne exécution de la suite du processus de fabrication et de contrôle. Il faut donc engager les équipes dans l'art de l'écriture.
- ✓ Le brief est soumis à l'AI-EA accompagné d'instruction (prompt) pour obtenir une analyse de la pertinence de ce cas d'utilisation à la portée globale du système d'information. Autrement dit, l'AI-EA se comporte comme un expert en architecture d'entreprise ayant une connaissance parfaite de l'existant et des développements en cours avec les autres utilisateurs. Elle produit alors des recommandations pour améliorer l'automatisation envisagée, ou pour suggérer une approche alternative, voire pour réutiliser une solution déjà développée ailleurs dans l'organisation.
- ✓ Sur la base des recommandations formulées par l'AI-EA, le brief de l'utilisateur est mis à jour et il développe alors son automatisation dans un environnement de test.
- ✓ Une fois satisfait du résultat de l'automatisation, l'AI-EA est de nouveau sollicitée pour auditer le développement réalisé. Elle détermine alors s'il est conforme pour un passage en production ou non.
- ✓ Après mise en production, la documentation d'architecture est automatiquement mise à jour par l'AI-EA. L'utilisateur peut alors visualiser sa contribution dans les nouveaux modèles d'EA.



Pour aller plus loin, vous trouverez sur le site de la communauté Engage-Meta plusieurs cartes thématiques couvrant un large spectre des impacts de l'IA et de la gestion des données à grande échelle dans l'entreprise ([www.engage-meta.com](http://www.engage-meta.com)). Je vous propose de commencer par ces quatre cartes, accessibles via les QR codes ci-dessous.



Je vous recommande de lire la carte Engage-Meta « Core System Data » qui présente les principes d'accélération de la production de documentation d'architecture grâce aux systèmes LLM. Vous verrez qu'ils sont associés à un référentiel de données basé sur les graphes de connaissance.



Vous pouvez également jeter un œil à la carte de gouvernance « Enterprise Architecture » qui rappelle les fondements de cette discipline, ainsi que les impacts de l'IA sur sa pratique.



Pour découvrir comment la gouvernance des données se mène avec le déploiement de l'IA, je vous recommande de lire la carte de gouvernance « Enterprise Governance ».



Vous pouvez lire la carte de gouvernance « Human Resources » pour y puiser les idées fortes sur l'accompagnement au changement des équipes.

## Solution AI-EA

La plateforme Boldo constitue un exemple intéressant de solution qui s'aligne avec notre vision d'une architecture d'entreprise pilotée par l'IA. Boldo propose une interface intuitive de gestion de l'information et des processus qui facilite la cartographie en temps réel des flux de travail, des rôles, et des applications utilisées dans l'entreprise.

Ce que Boldo apporte de spécifique, c'est sa capacité à combiner une vue systémique des opérations avec une logique de documentation automatique des processus et des responsabilités, souvent négligée dans les projets NoCode-AI spontanés. Dans un contexte de fabrication décentralisée d'agents IA, Boldo agit comme un pivot de cohérence, en maintenant un référentiel à jour de l'architecture organisationnelle, des dépendances et des impacts.

C'est exactement ce type d'outil qui peut renforcer une approche AI-EA, en fournissant à l'IA d'entreprise les données structurées et contextualisées nécessaires pour analyser les demandes des utilisateurs, suggérer des automatisations alignées, et auditer les initiatives avant leur mise en production. Boldo est un partenaire de la communauté Engage-Meta et vous pouvez découvrir leur solution ici : <https://www.boldo.io>

## Rejoindre Engage-Meta

Vous trouverez sur le site de la communauté un ensemble de ressources open source qui expliquent l'importance de travailler avec méthode pour déployer l'IA à grande échelle. Bien sûr, cela demande un effort d'appropriation des concepts, et c'est souvent moins gratifiant à court terme que d'utiliser directement les outils NoCode-AI.

Mais prendre le temps de lire, de comprendre et de formaliser une pensée complexe constitue une arme stratégique pour mieux maîtriser l'IA. En vous impliquant dans Engage-Meta, vous faites d'une pierre deux coups : vous appréhendez la complexité des systèmes IA-données tout en entretenant votre capacité à lire, structurer et partager votre réflexion avec vos équipes.

