



Loonar Compass

Guide de l'utilisateur

version 0.9.2 · octobre 2026

Loonar Mariusz Głowacki
ul. Gombrowicza 12, 73-110 Stargard
loonar.eu · mariusz@loonar.eu

Sommaire

1. [Présentation de Loonar Compass](#)
2. [Configuration requise](#)
3. [Installation](#)
4. [Fenêtre de l'application et navigation](#)
5. [Paramètres et langue](#)
6. [Conversation avec l'assistant](#)
7. [Travail avec le modèle Enterprise Architect](#)
8. [Exemples d'utilisation](#)
9. [Collaboration avec la fenêtre EA ouverte](#)
10. [Compass dans Enterprise Architect](#)
11. [Éditeur de diagrammes](#)
12. [Envoyer des diagrammes vers Miro](#)
13. [Documents : Word, Excel, images, Confluence](#)
14. [Modèles de documents](#)
15. [Mémoire de l'assistant](#)
16. [Validation des modifications et journal des modifications](#)
17. [Conversations : historique, recherche, travail en arrière-plan](#)
18. [Données, sécurité et confidentialité](#)
19. [Résolution des problèmes](#)
20. [Signalement de bugs et nouvelles versions](#)
21. [Mise à jour et désinstallation](#)
22. [Compass Hub \(service d'équipe\)](#)
23. [Mode de contrôle pour les pipelines CI](#)
24. [Licence, marques et contact](#)

1. Présentation de Loonar Compass

Loonar Compass est une application de bureau autonome pour Windows : un assistant IA pour travailler avec un modèle Enterprise Architect (Sparx Systems). Vous lui écrivez en langage courant, et il :

- **lit le modèle** (paquetages, éléments, diagrammes, connecteurs, attributs et opérations, valeurs étiquetées et mots-clés) et décrit ce qu'il contient,
- **analyse le modèle** : impact de la modification d'un élément sur le reste du modèle, endroits où les opérations sont utilisées, éléments sans connecteurs,
- **modifie le modèle** : crée des paquetages, des diagrammes, des éléments et des connecteurs, ajoute des attributs, des opérations, des valeurs étiquetées et des mots-clés, dispose les éléments sur les diagrammes, les déplace, les retire d'un diagramme ou les supprime du modèle, les colore, ajoute des légendes, copie des paquetages et en crée des baselines,
- **tient le glossaire du projet EA** (termes avec leur définition et leur type) ainsi que les **scénarios des éléments** (étapes de l'acteur et du système),
- **dessine** des diagrammes à partir d'une description, d'une photo, d'un croquis ou d'un fichier joint,
- **affiche les diagrammes** sous forme d'images dans le chat, tels qu'Enterprise Architect les dessine,
- **prépare des documents** à partir du contenu du modèle : Word, Excel, PowerPoint, PDF, images PNG et pages Confluence, y compris à partir de modèles de documents,
- **lit des documents** : PDF, Word, PowerPoint, Visio et Excel, sur votre disque ou dans la bibliothèque SharePoint de l'équipe,
- **suit les instructions de modélisation de l'équipe** conservées dans le référentiel EA lui-même, afin que tous ceux qui travaillent avec ce référentiel appliquent les mêmes règles,
- **retient les points convenus** d'une conversation à l'autre (conventions de modélisation, préférences, corrections).

L'application utilise via Internet un modèle de langage d'IA, avec votre propre clé API : par défaut Claude Haiku 5.5 (Anthropic) ; dans les paramètres, vous pouvez choisir un autre modèle Claude ou un modèle OpenAI (chapitre 5). Elle se connecte au modèle EA de deux façons : elle le lit directement (dans une base PostgreSQL, SQL Server, MySQL/MariaDB ou Oracle, ou dans le fichier .qea/.qeax) et elle y écrit par l'intermédiaire d'Enterprise Architect, avec le compte utilisateur du modèle indiqué dans les paramètres lorsque le modèle en possède un. Avec un fichier de modèle que vous avez ouvert dans EA, les modifications passent par votre fenêtre EA ; sinon, EA est lancé en arrière-plan.

Loonar Compass fonctionne comme une fenêtre autonome, lancée depuis le menu Démarrer, ou comme une fenêtre ancrée dans Enterprise Architect lui-même, qui travaille sur le projet ouvert dans cette fenêtre EA (chapitre 10).

L'interface et les réponses de l'assistant sont disponibles en quatre langues : polonais, anglais, allemand et français (chapitre 5). L'application est gratuite, y compris pour un usage commercial (chapitre 24).

Sans fenêtre, Compass contrôle aussi le modèle dans les pipelines CI (chapitre 23), et le programme distinct Compass Hub met les modèles de l'équipe à la disposition des personnes et des outils (chapitre 22).

2. Configuration requise

Prérequis. La configuration prise en charge exige les conditions suivantes :

1. **Le modèle est un fichier de projet EA .gea ou .geax, ou se trouve dans une base de données PostgreSQL, SQL Server, MySQL/MariaDB ou Oracle.** Les anciens formats de fichier (.eap, .eapx, .feap) ne sont pas pris en charge ; Enterprise Architect convertit un fichier .eap/.eapx en .gea par un transfert de projet (Settings → Model → Transfer → Full Project Transfer).
2. **Dans un modèle en base de données, la sécurité utilisateur est activée** (dans Enterprise Architect : Settings → Security → Enable Security), en mode « Require User Lock to Edit ». Dans un fichier de modèle, la sécurité est facultative : sans elle, l'assistant n'a besoin d'aucun compte EA. L'application se connecte au modèle en tant qu'utilisateur précis et pose des verrous pendant l'enregistrement ; lorsque la sécurité est désactivée, les verrous sont ignorés et cette configuration n'est pas prise en charge. Dans l'application, vous l'indiquez dans la carte **Modèle** des paramètres avec les options « La sécurité est activée dans le modèle (comptes utilisateurs EA) » et « Le modèle exige un verrou pour modifier (Require User Lock to Edit) » (chapitre 5).
3. **Un modèle dont la sécurité est activée contient un compte utilisateur pour l'assistant** (Settings → Security → Users), avec un mot de passe et des droits de modification. Le nom du compte est libre, par exemple `claud` ou `assistant`, mais le compte doit exister. Vous saisissez son identifiant et son mot de passe dans la carte **Modèle** des paramètres, avec le fichier ou la base du modèle.

Élément	Détails
Système	Windows 10 ou Windows 11 (64 bits)
Enterprise Architect	Installé sur le même ordinateur, avec une licence valide. L'application le lance en arrière-plan ; pendant ce temps, il occupe une place de licence. Compass dans EA (chapitre 10) travaille dans votre fenêtre EA.
Modèle EA	Fichier de projet EA .qea ou .qeax (seul son chemin est nécessaire) ou référentiel dans une base PostgreSQL avec la sécurité utilisateur activée (informations nécessaires : serveur, port, nom de la base, compte et mot de passe de la base). Un fichier .qea est destiné à une seule personne ; un fichier .qeax à une petite équipe qui partage le fichier. Pour un usage en entreprise, donnez à Compass un rôle PostgreSQL avec le seul droit SELECT sur les tables EA (<code>ea_*</code>) et réglez le chiffrement de la connexion sur « Obligatoire, certificat vérifié » (chapitre 5). Enterprise Architect enregistre avec ce même compte : pour un modèle que l'assistant doit aussi modifier, il faut en plus INSERT, UPDATE et DELETE sur ces tables, jamais les droits de superutilisateur. Un référentiel sur SQL Server, MySQL/MariaDB ou Oracle : les mêmes informations, avec la même règle pour le compte (chapitre 5, « SQL Server, MySQL et Oracle ») ; l'enregistrement exige une édition d'EA qui prend en charge les référentiels SGBD.
Pilote ODBC (SQL Server, MySQL, Oracle)	Uniquement pour un modèle sur SQL Server, MySQL/MariaDB ou Oracle : le pilote ODBC 64 bits de l'éditeur de la base. Pour SQL Server, Windows dispose déjà de son propre pilote « SQL Server » ; le pilote « ODBC Driver 18 for SQL Server » (ou 17) de Microsoft est préférable et il est utilisé en premier lorsqu'il est installé.
Compte utilisateur du modèle	Pour un modèle en base de données ou un fichier de modèle dont la sécurité est activée : identifiant et mot de passe d'un utilisateur défini dans le modèle EA. De préférence un compte distinct réservé à l'assistant, afin que ses modifications et ses verrous soient visibles séparément et n'entrent pas en conflit avec les vôtres.
Clé API	Une clé API Claude de la console Anthropic (Anthropic Console) ou, si vous choisissez le fournisseur OpenAI, une clé API OpenAI de la plateforme OpenAI (chapitre 5).
Internet	Connexion HTTPS à <code>api.anthropic.com</code> (ou à <code>api.openai.com</code> lorsque le fournisseur est OpenAI), directement ou via le proxy défini dans Windows (chapitre 18).
Confluence (facultatif)	Confluence Cloud ou Data Center, uniquement pour la publication de pages par l'assistant : adresse Confluence, jeton et clé de l'espace (chapitre 13). Sans cela, les pages sont tout de même créées, et vous collez le code vous-même.

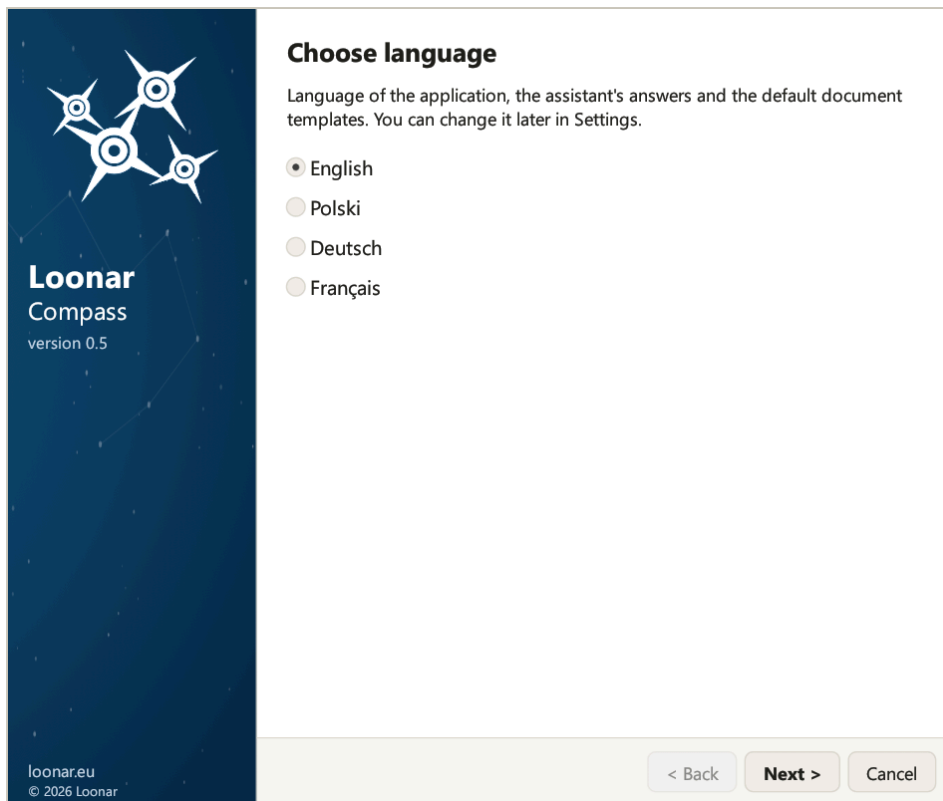
Miro (facultatif)	Uniquement pour envoyer des diagrammes vers un tableau Miro : un jeton d'accès d'une application Miro (chapitre 12).
Microsoft Office (facultatif)	Uniquement pour les fichiers PDF créés par l'assistant : Word convertit un document en PDF, PowerPoint une présentation (chapitre 13). Les fichiers Word et PowerPoint eux-mêmes sont créés sans Office.
SharePoint (facultatif)	Uniquement pour les documents de la bibliothèque SharePoint de l'équipe : une inscription d'application dans Microsoft Entra ID avec un secret client (chapitre 13).
Node.js (facultatif)	Uniquement pour les serveurs MCP lancés avec <code>npx</code> , par exemple les serveurs prêts à l'emploi Azure DevOps et Files (chapitre 7).
Logiciel de messagerie (facultatif)	Pour signaler des bugs et s'abonner aux informations sur les nouvelles versions : un logiciel de messagerie défini par défaut dans Windows, par exemple Outlook.

Coûts. L'application elle-même est gratuite. L'utilisation du modèle d'IA est facturée par le fournisseur choisi (Anthropic ou OpenAI) selon la consommation, sur le compte auquel appartient la clé API.

3. Installation

Lancez le fichier `LoonarCompass-Setup-0.9.exe`. L'assistant d'installation vous guide à travers les étapes avec les boutons **Suivant** et **Précédent**. Des droits d'administrateur ne sont nécessaires que si vous installez l'application pour tous les utilisateurs de l'ordinateur.

Avertissement Windows SmartScreen. Le programme d'installation n'est pas encore signé numériquement. Si Windows affiche « Windows a protégé votre ordinateur », cliquez sur **Informations complémentaires**, puis sur **Exécuter quand même**. Lorsque le contrôle intelligent des applications (Smart App Control) est activé, Windows bloque les applications non signées : attendez alors la version signée.



La première page de l'assistant d'installation : le choix de la langue

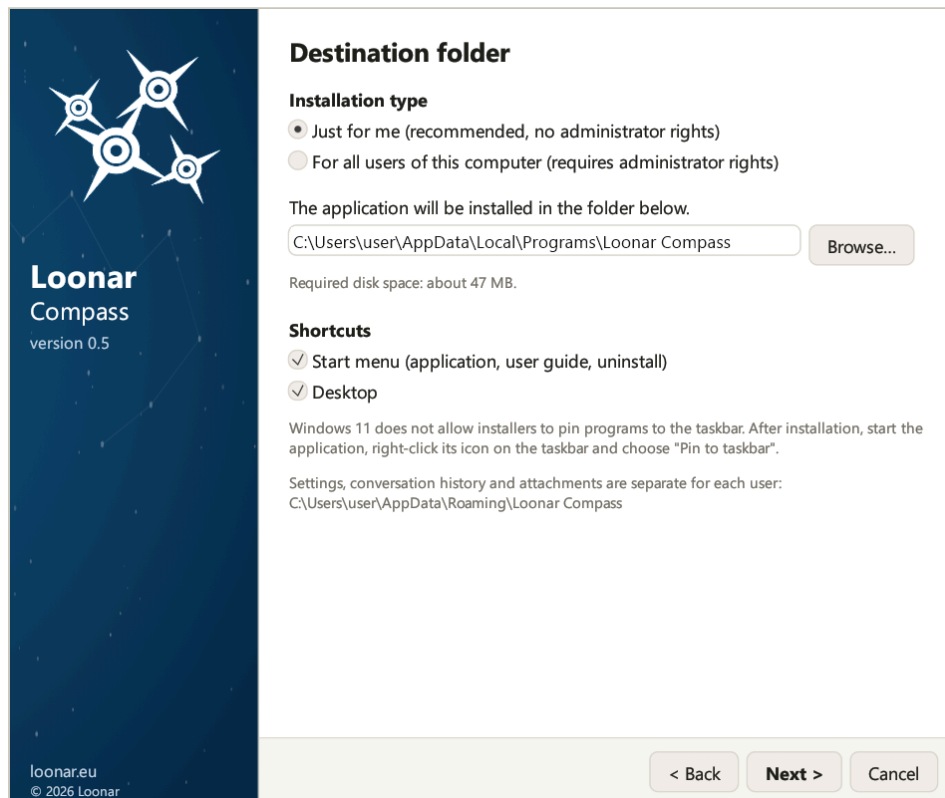
1. **Langue.** La première page demande la langue : polonais, anglais, allemand ou français. Lors d'une nouvelle installation, l'anglais est présélectionné ; lors d'une mise à jour, la langue enregistrée dans vos paramètres. La langue choisie s'applique à l'assistant d'installation, à l'application et aux réponses de l'assistant ; vous pourrez la modifier plus tard dans les paramètres (chapitre 5).
2. **Accueil.** Les prérequis et la liste des éléments à préparer avant l'installation.
3. **Nouveautés.** Lors d'une mise à jour : les modifications depuis la version installée. Lors d'une nouvelle installation : les principales modifications de la version actuelle.
4. **Contrat de licence.** Cochez « J'accepte les conditions du contrat de licence ».
5. **Statistiques d'utilisation.** L'option « Envoyer des statistiques d'utilisation anonymes (recommandé) » est cochée. La page explique ce qui est envoyé et ce qui ne l'est jamais ; vous pouvez changer d'avis plus tard dans la vue « Statistiques d'utilisation » (chapitre 18). Les données que l'assistant demandait auparavant (clé d'API, modèle, Enterprise Architect) se saisissent désormais après le premier démarrage, dans la vue « Paramètres » (chapitre 5).
6. **Dossier de destination.** Le type d'installation (décrit ci-dessous), le dossier du programme (le bouton **Parcourir...** permet d'en choisir un autre) et les raccourcis : menu Démarrer (application, guide de l'utilisateur, désinstallation) et bureau. L'option « **Ajouter Loonar Compass à Enterprise Architect** » installe Compass dans EA (chapitre 10) ; redémarrez EA après l'installation.
7. **Prêt pour l'installation.** Le récapitulatif du type d'installation, du dossier, des raccourcis et de la configuration. Cliquez sur **Installer** ; la barre de progression affiche le pourcentage et l'étape en cours.
8. **Fin.** Si « Lancer Loonar Compass » est coché, l'application s'ouvre après un clic sur **Terminer**.

Type d'installation

Sur la page « Dossier de destination », vous choisissez pour qui vous installez l'application :

- « **Pour moi uniquement (recommandé, sans droits d'administrateur)** » : le programme est installé dans le dossier `%LOCALAPPDATA%\Programs\Loonar Compass`. Windows ne demande pas de droits d'administrateur.
- « **Pour tous les utilisateurs de cet ordinateur (nécessite des droits d'administrateur)** » : le programme est installé dans le dossier `%ProgramFiles%\Loonar Compass`, où seuls les administrateurs peuvent modifier les fichiers. Windows ne demande des droits d'administrateur qu'après un clic sur **Installer**. Si vous ne les accordez pas, rien n'est installé.

Quel que soit le type d'installation, les paramètres et les secrets sont toujours enregistrés pour l'utilisateur Windows connecté. Dans une installation pour tous les utilisateurs, chacun a ses propres paramètres, conversations et mots de passe.



Le type d'installation, le dossier du programme et les raccourcis

Barre des tâches. Windows 11 ne permet pas aux programmes d'installation d'épingler des programmes à la barre des tâches. Lancez l'application, cliquez avec le bouton droit sur son icône dans la barre des tâches et choisissez « Épingler à la barre des tâches ».

Mise à jour depuis une version antérieure

Loonar Compass s'appelait auparavant Loonar EA Assistant : les paramètres, les conversations, la mémoire, les modèles de documents et les mots de passe de Loonar EA Assistant sont repris automatiquement.

- Une ancienne version installée pour vous uniquement est supprimée pendant l'installation (fichiers du programme, raccourcis et entrée dans les paramètres de Windows) ; vos données sont conservées.
- Si l'ancienne version a été installée pour tous les utilisateurs, la page « Prêt pour l'installation » l'indique et propose le bouton **Supprimer l'ancienne version (droits d'administrateur)** ; vous pouvez aussi la supprimer plus tard dans les paramètres de Windows (Applications). Vos paramètres et vos conversations sont conservés.

- Lors d'une mise à jour depuis la version 0.1 ou 0.2, l'assistant d'installation présélectionne la langue enregistrée dans les paramètres, ou l'anglais si aucune langue n'y figure : vérifiez ce choix avant de cliquer sur **Suivant**.
- Un changement de langue dans le programme d'installation ne traduit ni la mémoire ni les modèles de documents. Pour les traduire, conservez la langue actuelle dans le programme d'installation, puis changez-la dans la carte « Langue et validation des modifications » des paramètres (chapitre 5).
- Le mot de passe de la base, que les anciennes versions conservaient dans le fichier de paramètres, est transféré dans le Gestionnaire d'identification Windows, et les fichiers aux noms polonais du dossier des données reçoivent des noms anglais (par exemple `audit_log.json1` et `templates`).

4. Fenêtre de l'application et navigation

Sur le côté gauche de la fenêtre se trouve une **barre de navigation** bleu marine avec des icônes. Lorsque vous survolez une icône avec la souris, son nom apparaît. Un clic change de vue.

Icône	Contenu
Chat	La liste des conversations et la conversation avec l'assistant (chapitres 6 et 17)
Paramètres	Modèle d'IA, modèle EA, Enterprise Architect, langue et mode de validation, nouvelles versions, Confluence, Jira, SharePoint, agents d'événements, serveurs MCP, hub d'équipe, Miro (chapitre 5)
Mémoire de l'assistant	Les points que l'assistant a retenus d'une conversation à l'autre (chapitre 15)
Modèles de documents	Les modèles de documents que l'assistant remplit avec le contenu du modèle EA (chapitre 14)
Journal des modifications du modèle	Les enregistrements de l'assistant dans le modèle, avec export vers Excel (chapitre 16)
Nouveautés	La liste des fonctionnalités et des limitations connues de chaque version
Signaler un bug	Formulaire de signalement de bug avec images (chapitre 20)
Statistiques d'utilisation	Ce que l'application envoie de façon anonyme, le résumé du jour, le journal des envois des 30 derniers jours, la désactivation et un nouvel identifiant d'installation (chapitre 18)
Soutenir le projet (icône de cœur, en bas de la barre, au-dessus de « À propos »)	Soutien volontaire au développement de l'application, liens vers GitHub et loonar.eu
À propos (en bas de la barre)	Version, propriétaire de l'application, liens (politique de confidentialité, code source sur GitHub, loonar.eu), marques, licence et composants open source (chapitre 24)

Un « ! » orange sur les icônes Paramètres, Mémoire de l'assistant et Modèles de documents signifie qu'après un changement de langue, la traduction n'est pas encore terminée (chapitre 5).

Soutenir le projet. Loonar Compass est gratuit, y compris pour un usage commercial, et le restera. S'il vous fait gagner du temps, vous pouvez soutenir son développement par une contribution volontaire : le bouton **Soutenir sur suppi.pl** ouvre <https://suppi.pl/loonar>. La contribution finance notamment un certificat de signature de code. Vous pouvez aussi aider en signalant des bugs et des idées, en recommandant l'application ou en donnant une étoile au projet sur GitHub ; la vue contient les liens vers le code source sur GitHub et vers loonar.eu.

Vue du chat

- **La liste des conversations** à gauche est regroupée en « Aujourd'hui », « Hier », « 7 derniers jours » et « Plus ancien ». Vous commencez une nouvelle conversation avec le bouton **+**. La conversation sélectionnée est mise en surbrillance. Un clic droit sur une conversation permet de la supprimer (chapitre 17).
- **L'en-tête de la conversation** affiche son titre, le bouton **Modèle : ...** avec le modèle de la conversation (chapitre 17) et, pendant que l'assistant travaille, l'indicateur « L'assistant travaille ».
- **Messages** : les vôtres sont dans une bulle beige, les réponses de l'assistant portent l'icône de l'application. Chaque message indique l'heure (les plus anciens aussi la date).
- **L'écran d'une nouvelle conversation** affiche le logo, une indication de ce que vous pouvez faire et le modèle EA qu'utilisera la conversation, avec les boutons **Choisir un autre modèle...** et **Confirmer le modèle et commencer** (chapitre 17).
- **La carte de message** en bas : le champ de texte, le bouton de collage, le bouton **+** (choisir dans le modèle), avec Claude aussi les commutateurs  (raisonnement) et  (code dans le bac à sable), le modèle et le mode de validation, ainsi que les boutons ronds d'envoi et d'arrêt (chapitre 6). Sous la carte, un rappel : « Les réponses et les modifications sont préparées par un modèle d'IA et peuvent contenir des erreurs. Vérifiez-les avant de les valider. »

5. Paramètres et langue

La vue **Paramètres** comporte douze cartes : **Modèle d'IA** et **Langue et validation des modifications** dans la colonne de gauche, **Modèle** et **Enterprise Architect** dans celle de droite, puis en dessous **Confluence**, **Jira**, **SharePoint**, **Agents d'événements (Compass dans EA)**, **Serveurs MCP**, **Équipe (Compass Hub)**, **Miro** et **Nouvelles versions (newsletter)**. Dans Compass ouvert dans Enterprise Architect, les cartes **Modèle** et **Enterprise Architect** n'apparaissent pas, car le modèle est le projet ouvert dans cette fenêtre EA (chapitre 10). Vous enregistrez chacune séparément avec le bouton situé sous la carte, et l'application teste aussitôt les données saisies. Le résultat apparaît sous le bouton : en vert avec une description ou en rouge avec la cause de l'erreur.



Settings

Each group is saved separately; after saving, the application immediately checks that the details work. Leave a password or key field empty to keep the current value.

1. Claude

Model

Max tokens

Workspace ID

Claude API key

To answer, the app sends the conversation, attachments, remembered notes, templates, model query results and diagram images to Anthropic (Claude API) using your key. Loonar does not receive this data.

Save and test key

4. Language and change approval

Language

Approval mode

ask - asks before every save

Save

2. Model

☒ PostgreSQL database ☐ EA file (.qea, .qeax)

Server

Port

Database name

Database user

Database password

Connection encryption

TLS when the server offers it, otherwise an unencrypted connection.
Administrators can enforce TLS for all clients with hostssl entries in pg_hba.conf.

Save and test connection

3. Enterprise Architect

Path to EA.exe ...

EA user account

EA account password

Save and test EA login

Loonar Compass 0.4 · Loonar Mariusz Głowacki, ul. Gombrowicza 12, 73-110 Stargard · loonar.eu · mariusz@loonar.eu

Paramètres : chaque carte avec son propre enregistrement et son test (capture d'une version antérieure : les cartes étaient numérotées, la carte « Modèle d'IA » s'appelait « Claude » et le compte EA se trouvait dans la carte Enterprise Architect)

Carte	Champs	Ce que vérifie le test
Modèle d'IA	Fournisseur, Modèle, Effort, Jetons max., ID de l'espace de travail (workspace) et Adresse de Claude (Claude uniquement), Adresse du jeton OAuth2 (avec l'ID client OAuth2, la portée et le secret client) et la clé API du fournisseur choisi : Clé API Claude ou Clé API OpenAI (détails ci-dessous). Sous les champs, une note indique ce qui est envoyé au fournisseur (chapitre 18).	« Enregistrer et tester la clé » vérifie la clé par une courte requête au modèle choisi
Modèle	Au choix : Base PostgreSQL, SQL Server, MySQL ou Oracle (serveur, port, nom de la base, compte de la base, mot de passe de la base conservé dans le Gestionnaire d'identification Windows, chiffrement de la connexion ; détails ci-dessous) ou Fichier EA (.qea, .qeax) (Fichier du modèle ; le bouton ... ouvre la sélection de fichier). En dessous, le compte EA du modèle : la case « La sécurité est activée dans le modèle (comptes utilisateurs EA) », le compte utilisateur EA et le mot de passe du compte EA (sans cette case, les deux champs sont désactivés et EA ouvre le modèle sans connexion), ainsi que l'option « Le modèle exige un verrou pour modifier (Require User Lock to Edit) » : si elle est activée, l'assistant verrouille les paquetages et éléments qu'il modifie, puis libère les verrous ; sans elle, il ne verrouille rien.	Base : « Enregistrer et tester la connexion » se connecte, compte les paquetages, éléments et diagrammes et, pour PostgreSQL, indique si la connexion est chiffrée et avertit si le compte de la base est un superutilisateur. Fichier : « Enregistrer et tester le fichier » l'ouvre en lecture seule et compte les paquetages, éléments et diagrammes. Le test enregistre aussi un paquetage de test dans le modèle, le vérifie et le supprime aussitôt
Code source	Dossiers de code source : clones locaux de dépôts que l'assistant lit sans demander (lister, rechercher, lire des fichiers ; lecture seule), séparés par « ; » ; le bouton ... ajoute un dossier. Pour tout autre dossier l'assistant demande d'abord. La carte est aussi visible dans l'onglet EA.	« Enregistrer et vérifier les dossiers » vérifie que les dossiers existent
Enterprise Architect	Chemin d'EA.exe (le bouton ... ouvre la sélection de fichier). Le compte EA se règle dans la carte Modèle, car il	« Enregistrer et tester l'accès à EA » ouvre le modèle dans EA (pour un fichier de modèle que vous avez ouvert : dans votre

	appartient au fichier ou à la base du modèle.	fenêtre EA), se connecte avec le compte de la carte Modèle, enregistre un paquetage de test dans le modèle, le vérifie et le supprime aussitôt. Le démarrage d'EA prend de quelques secondes à plus de dix secondes.
Langue et validation des modifications	Langue de l'application ; mode de validation : ask, plan ou auto (chapitre 16)	Enregistrement uniquement
Nouvelles versions (newsletter)	Adresse e-mail et consentement à recevoir des informations sur les nouvelles versions, avec une note sur l'utilisation de l'adresse et un lien vers la politique de confidentialité (chapitre 20)	Prépare un message dans le logiciel de messagerie
Confluence	Adresse (commençant par https://), e-mail (Cloud uniquement), jeton, clé de l'espace, ID de la page parente (chapitre 13)	Se connecte à Confluence avec le jeton indiqué et vérifie l'accès à l'espace
Miro	Jeton d'accès d'une application Miro avec les autorisations boards:read et boards:write (chapitre 12)	Se connecte à Miro et compte les tableaux visibles avec le jeton
Jira	Adresse Jira, e-mail (Cloud uniquement), jeton, clé de projet par défaut, certificat client, mot de passe du certificat (chapitre 7). L'adresse doit commencer par https:// .	« Enregistrer et tester la connexion Jira » se connecte à Jira et vérifie le projet
SharePoint	Adresse du site, Bibliothèque de documents (vide = Documents), ID du locataire (annuaire), ID d'application (client), Secret client (chapitre 13)	« Enregistrer et tester la connexion SharePoint » se connecte à Microsoft Graph et compte les éléments de la bibliothèque
OneDrive	ID d'application (client) d'une inscription dans Entra ID comme client public (« Applications mobiles et de bureau », URI de redirection http://localhost , autorisations déléguées Files.ReadWrite et User.Read), Locataire (organizations , consumers pour un compte personnel, ou votre locataire)	« Enregistrer, se connecter et tester OneDrive » ouvre une fois la connexion Microsoft dans le navigateur et compte les éléments du dossier racine ; ensuite l'assistant liste, lit et (chaque fois avec votre accord) envoie
Google Drive	ID client et code secret d'un client OAuth de type « Application de bureau	« Enregistrer, se connecter et tester Google Drive » ouvre une

	» dans un projet Google Cloud avec l'API Google Drive activée	fois la connexion Google dans le navigateur ; ensuite l'assistant liste, lit (aussi Google Docs, Sheets et Slides) et (chaque fois avec votre accord) envoie
Équipe (Compass Hub)	Adresse du hub, jeton, Ce modèle dans le hub (facultatif) (chapitre 22)	« Enregistrer et tester la connexion au hub » se connecte au hub et affiche votre compte, votre rôle et la politique de l'équipe
Serveurs MCP	Liste de vos propres serveurs MCP, enregistrée dans le fichier <code>mcp_servers.json</code> à côté des paramètres, la liste Ajouter un serveur prêt à l'emploi et le champ Jeton d'un serveur MCP (chapitre 7)	« Enregistrer et tester les serveurs MCP » se connecte à chaque serveur et affiche le nombre de ses outils
Agents d'événements (Compass dans EA)	Règles du fichier <code>event_agents.json</code> (chapitre 10)	« Enregistrer les agents d'événements » vérifie les règles et affiche le nombre d'agents

Un champ de mot de passe ou de clé laissé vide conserve la valeur actuelle. La mention **enregistré** à côté d'un champ indique que la valeur est déjà mémorisée ; **ABSENT** signifie qu'elle manque encore.

Valeurs par défaut. Lorsqu'il n'existe encore aucun paramètre, les champs Serveur, Nom de la base et Compte utilisateur EA ainsi que les mots de passe sont vides (le champ Serveur affiche l'indication grise « p. ex. 127.0.0.1 ») ; le port est 5432, le compte de la base `postgres`, le chemin d'EA `C:\Program Files\Sparx Systems\EA\EA.exe`, le mode de validation ask, le fournisseur Anthropic (Claude), le modèle `claude-haiku-5-5`, l'effort « par défaut du modèle » et Jetons max. 10000. Les paramètres existants ne sont pas modifiés.

Jetons max. Si l'assistant interrompt son travail avec un message de réponse tronquée, augmentez cette valeur (par exemple à 20000). La limite ne coûte rien en soi : vous payez le texte réellement généré.

Le mot de passe de la base ne doit pas contenir de point-virgule (;), car Enterprise Architect ne peut pas le transmettre dans sa chaîne de connexion. L'adresse Confluence doit commencer par `https://` ; `http://` n'est accepté que pour un serveur installé sur le même ordinateur.

Le verrouillage des paquetages et éléments lors de l'enregistrement dépend de l'option « Le modèle exige un verrou pour modifier (Require User Lock to Edit) » de la carte Modèle : cochez-la lorsque « Require User Lock to Edit » est activé pour le modèle dans EA.

Modèle d'IA : fournisseur, modèle et effort

La carte **Modèle d'IA** détermine quel modèle de langage répond dans les conversations :

Champ	Signification
Fournisseur	Anthropic (Claude) (par défaut) ou OpenAI .
Modèle	Un champ de texte avec une liste ▼. Claude : <code>claude-haiku-5-5</code> (par défaut), <code>claude-sonnet-5-5</code> , <code>claude-opus-5-5</code> , <code>claude-fable-5-1</code> . OpenAI : <code>gpt-5.1</code> , <code>gpt-5</code> , <code>gpt-5-mini</code> . Vous pouvez aussi saisir l'identifiant d'un autre modèle du fournisseur choisi.
Effort	par défaut du modèle (par défaut), low , medium , high , xhigh ou max . Un effort plus élevé donne des réponses et un travail sur le modèle plus approfondis, mais consomme plus de jetons et de temps, et coûte donc plus cher ; un effort plus bas est plus rapide et moins cher. Avec OpenAI, l'effort correspond au reasoning effort du modèle.
Jetons max.	Limite de longueur d'une réponse du modèle (voir ci-dessous).
ID de l'espace de travail (workspace)	Claude uniquement, facultatif : l'identifiant d'un espace de travail (workspace) de la console Anthropic, par exemple <code>wrkspc_...</code> .
Adresse de Claude, Adresse du jeton OAuth2	Facultatif : Claude par Azure AI Foundry ou une passerelle d'entreprise, et connexion par OAuth2 au lieu d'une clé API (voir « Connexion au service d'IA » ci-dessous).
Point de terminaison	OpenAI uniquement. Vide = OpenAI. Vous pouvez saisir l'adresse d'un autre serveur compatible avec l'API OpenAI : Azure OpenAI (<code>https://<ressource>.openai.azure.com/openai/v1</code> ; la clé est envoyée dans l'en-tête <code>api-key</code>), Ollama (<code>http://localhost:11434/v1</code>), LM Studio (<code>http://localhost:1234/v1</code>) ou une passerelle d'entreprise. Les serveurs locaux fonctionnent sans clé, et <code>http://</code> n'est autorisé que pour localhost. Dans le champ Modèle, saisissez le nom du modèle (sous Azure : le nom du déploiement).
Clé API Claude / Clé API OpenAI	La clé du fournisseur choisi. La clé Claude se crée dans la console Anthropic (Anthropic Console). La clé OpenAI se crée sur la plateforme OpenAI : <code>platform.openai.com</code> → API keys → Create new secret key . Les clés sont conservées dans le Gestionnaire d'identification Windows, chacune séparément.

Enregistrer et tester la clé vérifie la clé par une courte requête au modèle choisi. Les données nécessaires aux réponses sont envoyées uniquement au fournisseur choisi : à `api.anthropic.com` ou à `api.openai.com` (chapitre 18). Si vous ne modifiez rien dans cette carte, l'application fonctionne comme avant : avec Claude Haiku 5.5 et l'effort par défaut du modèle. Les modèles plus grands et un effort plus élevé coûtent plus cher ; les tarifs sont publiés par Anthropic et OpenAI.

Changement de langue

L'application fonctionne en quatre langues : **polonais, anglais, allemand et français**. Vous choisissez la langue dans le programme d'installation (anglais par défaut) et la modifiez dans la carte « **Langue et validation des modifications** » :

1. Choisissez une langue dans la liste **Langue**.

2. Cliquez sur **Enregistrer**. L'interface passe à la nouvelle langue, et l'assistant y répond à partir du message suivant.
3. L'application transpose vos contenus dans la nouvelle langue : **la mémoire de l'assistant et vos modèles sont traduits par le modèle d'IA**.

Seules la mémoire et les modèles de documents sont traduits. **L'historique des conversations n'est jamais traduit** : les conversations restent dans la langue dans laquelle vous les avez menées.

Pas de connexion au modèle d'IA. La traduction nécessite une clé API fonctionnelle. En l'absence de connexion, un « ! » orange apparaît sur les icônes **Paramètres**, **Mémoire de l'assistant** et **Modèles de documents**, et l'info-bulle explique que les contenus ne sont pas encore traduits. La traduction reprend au prochain démarrage de l'application, ou immédiatement après un clic sur **Traduire maintenant** dans la carte « Langue et validation des modifications » ; l'enregistrement des autres cartes ne la relance pas.

Lorsque vous enregistrez la langue, Compass règle aussi le correcteur orthographique d'EA sur cette langue, si EA a un dictionnaire pour elle. EA ne fournit que des dictionnaires anglais ; pour les autres langues, Compass vérifie lui-même les textes (chapitre 7, « Orthographe dans la langue de Compass »).

Chiffrement de la connexion à la base

Le champ **Chiffrement de la connexion** de la carte « Modèle » (option « Base PostgreSQL ») détermine si la lecture du modèle par l'assistant passe par TLS :

Réglage	Effet
Préféré (par défaut)	TLS si le serveur le propose, sinon une connexion non chiffrée. Fonctionne avec tous les serveurs.
Obligatoire	TLS est obligatoire : sans TLS, l'application ne se connecte pas. Le certificat du serveur n'est pas vérifié.
Obligatoire, certificat vérifié	TLS est obligatoire, et le certificat du serveur est vérifié, avec le nom du serveur, dans le magasin de certificats Windows (y compris les certificats d'entreprise).

- Le test de connexion indique si la connexion est chiffrée : « La connexion est chiffrée (TLS). » ou « La connexion n'est PAS chiffrée : le serveur ne propose pas TLS. ».
- Avec la vérification du certificat, le nom saisi dans le champ **Serveur** doit correspondre au nom figurant dans le certificat du serveur.
- L'administrateur de la base peut imposer TLS à tous les clients avec des entrées `hostssl` dans le fichier `pg_hba.conf`.
- Le test avertit aussi lorsque le compte de la base est un superutilisateur. Ces droits ne sont pas nécessaires : utilisez un compte ordinaire disposant des droits de lecture et d'écriture sur les tables du modèle (l'assistant ne fait que lire la base, mais Enterprise Architect enregistre les modifications avec ce même compte).
- Le réglage concerne la lecture du modèle par l'assistant. La connexion d'Enterprise Architect lui-même à la base (lors des enregistrements) suit les paramètres d'EA.
- Avec SQL Server, le réglage est transmis au pilote ODBC (« Obligatoire » : chiffré, le certificat n'est pas vérifié ; « Obligatoire, certificat vérifié » : chiffré et vérifié). Avec MySQL et Oracle, réglez le chiffrement

dans le pilote ou dans `db.odbc` (ci-dessous).

SQL Server, MySQL et Oracle

Outre PostgreSQL et les fichiers EA, Compass lit les référentiels EA sur **SQL Server, MySQL/MariaDB et Oracle** : dans la fenêtre autonome, dans le hub d'équipe (chapitre 22) et en mode de contrôle (chapitre 23). Dans la carte **Modèle**, choisissez **SQL Server, MySQL** ou **Oracle** et remplissez les mêmes champs que pour PostgreSQL ; le port prend la valeur habituelle de cette base (1433, 3306, 1521), sauf si vous l'avez modifié vous-même. Le modèle d'une conversation (chapitre 17) peut lui aussi être une telle base.

- **Pilote ODBC.** Compass se connecte par le pilote ODBC 64 bits de l'éditeur de la base et le trouve tout seul. Pour SQL Server, il prend « ODBC Driver 18 for SQL Server », puis 17, et en dernier le pilote « SQL Server » propre à Windows. Pour MySQL, le pilote ODBC de MySQL ou de MariaDB ; pour Oracle, le pilote du client Oracle (« Oracle in ... »). Si aucun n'est installé, le test indique les pilotes présents.
- **Votre propre chaîne de connexion.** Lorsque la connexion demande quelque chose que la carte ne propose pas (par exemple un nom TNS Oracle ou des options du pilote), indiquez une chaîne de connexion ODBC complète dans `app.properties` sous `db.odbc`, sans le mot de passe : Compass ajoute le mot de passe depuis le Gestionnaire d'identification. Exemple : `db.odbc=Driver={ODBC Driver 18 for SQL Server};Server=sql01,1433;Database=ea;UID=compass;Encrypt=yes`.
- **Lecture seule.** La lecture se fait dans une transaction toujours annulée (rollback) : Compass lui-même ne modifie rien dans la base. Un compte avec SELECT sur les tables EA (`t_*`) suffit pour lire ; quand Enterprise Architect enregistre avec le même compte (SQL Server, ci-dessous), il lui faut aussi INSERT, UPDATE et DELETE sur ces tables.

Enregistrement. Les modifications d'un tel modèle sont enregistrées par Enterprise Architect, comme avec PostgreSQL. Votre édition d'EA doit prendre en charge les référentiels SGBD.

- **SQL Server** : Compass construit lui-même la connexion propre à EA (fournisseur OLE DB SQLOLEDB), à partir des champs de la carte Modèle. Testez-la avec « Enregistrer et tester l'accès à EA ».
- **MySQL et Oracle** : EA les atteint par une source de données ODBC. Indiquez la connexion d'EA dans `app.properties` sous `ea.conn` : le texte d'un raccourci de connexion EA, par exemple `ea_lab --- DBType=0;Connect=Provider=MSDASQL.1;Data Source=<DSN ODBC>;...`. Sans `ea.conn`, utilisez Compass dans Enterprise Architect (chapitre 10) pour les modifications : il enregistre par votre fenêtre EA, qui fonctionne avec tout référentiel ouvert dans EA.

Dans `app.properties`, le choix s'écrit `model.source=sqlserver`, `mysql` ou `oracle` (`postgres` et `file` comme avant), avec `db.host`, `db.port`, `db.name`, `db.user` et `db.sslmode`.

Connexion au service d'IA : OAuth2 et autres adresses

Certaines entreprises donnent accès au modèle d'IA par leur propre passerelle ou par Microsoft Azure au lieu d'une clé API. La carte **Modèle d'IA** a des champs pour cela :

Champ	Signification
Adresse de Claude	Claude uniquement : vide = Anthropic ; Azure AI Foundry ou une passerelle d'entreprise : son adresse de base, par exemple <code>https://<ressource>.services.ai.azure.com/anthropic</code> . Elle doit commencer par <code>https://</code> (<code>http://</code> seulement sur cet ordinateur).
Adresse du jeton OAuth2	Facultatif, à la place d'une clé API : Compass obtient un jeton à cette adresse (identifiants client OAuth2, client credentials) et l'envoie à la place de la clé, pour Claude comme pour OpenAI. Pour Microsoft Entra ID : <code>https://login.microsoftonline.com/<tenant>/oauth2/v2.0/token</code> .
ID client OAuth2, Portée OAuth2 (scope), Secret client OAuth2	Affichés quand l'adresse du jeton est remplie : l'ID et le secret de l'application enregistrée chez le fournisseur, et la portée, par exemple <code>https://cognitiveservices.azure.com/.default</code> . Le secret est conservé dans le Gestionnaire d'identification Windows.

Les champs sont enregistrés sous `claude.url` , `ai.oauth.token_url` , `ai.oauth.client_id` et `ai.oauth.scope` dans `app.properties` . Testez-les avec **Enregistrer et tester la clé**.

6. Conversation avec l'assistant

Conversations

+

Search conversations...

Today

In a new package "Web s..."

In a new package "Online sho..."

In a new package "Web store", draw a BPMN diagram

Order placed by customer

Payment started

Choose payment method

Authorise payment

Confirm payment

Payment confirmed

Ship order

Notify customer

Order completed

Save as...Open full sizeOrder_process.png

15:06

I created the package **Web store** in the model root and drew the BPMN diagram "**Order process**" in it. The diagram is below.

The process runs left to right:

1. **Order placed by customer** (start event).

2. **Payment** is an embedded subprocess, with its steps drawn inside it: **Choose payment method** (user task), **Authorise payment** and **Confirm payment** (service tasks), bracketed by the start event **Payment started** and the end event **Payment confirmed**.

3. **Ship order** (service task).

4. **Notify customer** (send task).

5. **Order completed** (end event).

All elements are connected with sequence flows. The layout is clean. The text in "Choose payment method" runs slightly past the edge of its box.

15:06

Write a message, drag in files or paste an image (Ctrl+V)...

Drag files into the window to attach them

claude-haiku-5-5 · mode ask

Answers and changes are prepared by an AI model (Claude by Anthropic) and may contain errors. Review them before you approve.

Ready.

Une conversation avec l'image d'un diagramme : la liste des conversations à gauche, le chat à droite, la carte de message en bas

Écrire

Vous saisissez votre message dans la **carte de message** en bas de la fenêtre. La carte contient le champ de texte, le bouton de collage, le bouton  (Choisir dans le modèle, décrit ci-dessous), avec Claude aussi les commutateurs  et  (« Raisonnement et calculs » ci-dessous), le nom du modèle et le mode de validation actuel, ainsi que des boutons ronds : **Envoyer** et, pendant que l'assistant travaille, **Arrêter**.

Touche	Action
Entrée	Envoie le message
Maj + Entrée	Nouvelle ligne dans le message
↑ dans un champ vide	Rappelle votre dernier message de cette conversation ; les appuis suivants remontent aux précédents, ↓ revient
Ctrl + V	Colle du texte et, lorsque le presse-papiers contient une image ou des fichiers, les ajoute en pièces jointes

Le champ de message s'agrandit avec le texte. Les fautes d'orthographe sont soulignées en rouge à l'aide du dictionnaire du système Windows, dans la lanque définie dans les paramètres du système. Cliquez avec

le bouton droit sur un mot souligné et choisissez **Ajouter « ... » au dictionnaire** : le mot est ajouté au dictionnaire de Windows, et une expression polonaise l'est sous toutes ses formes fléchies (singulier et pluriel, du nominatif au vocatif). Dans le même menu, **Ajouter « ... » au glossaire du projet EA** demande à l'assistant d'ajouter le terme au glossaire du projet (chapitre 7).

Choisir dans le modèle

Le bouton  à côté du bouton de collage ouvre **Choisir dans le modèle**. Saisissez une partie du nom d'un élément, d'un paquetage ou d'un diagramme dans le champ « Nom d'un élément, d'un paquetage ou d'un diagramme » ; avec un champ vide, vous voyez les « Paquetages principaux du modèle : ». Un clic sur un résultat l'ajoute au message sous forme de référence, par exemple «Élément: Commande (Class, id 26)». C'est la même référence que celle du menu contextuel d'EA **Ajouter à la conversation Compass** (chapitre 10) : l'assistant sait ainsi exactement de quel objet il s'agit.

Décris «Élément: Commande (Class, id 26)» et indique les diagrammes sur lesquels il figure.

Pièces jointes

Vous pouvez joindre jusqu'à 5 fichiers à un message, chacun de 10 Mo au maximum. Méthodes : faites glisser les fichiers dans la fenêtre (depuis l'Explorateur ou depuis un navigateur), collez-les depuis le presse-papiers ( + ) ou utilisez le bouton de collage de la carte de message.

Type de fichier	Ce que voit l'assistant
Images (PNG, JPG, GIF, WebP)	L'image : photo d'un tableau blanc, croquis, capture d'écran. Dans le chat, les fichiers GIF et WebP apparaissent sous forme de carte de fichier, et non d'image.
PDF	Le document entier, chaque page ; y compris un scan sans couche de texte, dont le modèle lit les pages comme des images
Word (DOCX)	Le texte du document
PowerPoint (PPTX, PPTM)	Chaque diapositive avec son texte et ses notes
Visio (VSDX, VSDM)	Chaque page : les formes avec leur texte et les connexions entre elles (A → B), par exemple un processus à modéliser dans EA
Excel (XLSX, XLSM)	Chaque feuille avec les valeurs de ses cellules
Fichiers texte (BPMN, XML, CSV, JSON, code...)	Le contenu du fichier ; le fichier doit être encodé en UTF-8
ZIP	Les fichiers qu'il contient, chacun selon son type : au plus 20 fichiers et 30 Mo au total ; un ZIP contenu dans un ZIP n'est pas décompressé
Anciens formats DOC, PPT, XLS, VSD	Ils ne sont pas lus : ouvrez le fichier dans Word, PowerPoint, Excel ou Visio et enregistrez-le dans le nouveau format (DOCX, PPTX, XLSX, VSDX)
Autres fichiers binaires	Ils ne sont pas lus : l'assistant voit seulement le nom du fichier et une mention indiquant que son contenu ne peut pas être transmis

Pendant que l'assistant travaille

- Sous les messages apparaît la **carte des étapes** : ce que l'assistant fait en ce moment, combien de temps cela prend et le résultat de chaque étape. Pendant le travail, la carte est développée ; une fois le travail terminé, elle se réduit à une ligne ; un clic sur l'en-tête de la carte la développe à nouveau. En français, la carte affiche un court résultat traduit (par exemple « lignes : 12 », « enregistré dans le modèle EA », « image prête ») ; le résultat détaillé, rédigé en anglais, ne s'affiche que dans l'interface anglaise.
- Vous pouvez **continuer à écrire** : le message suivant est mis en file d'attente et l'assistant en tiendra compte à l'étape suivante, sans interrompre son travail.
- Le bouton **Arrêter** interrompt le travail après l'étape en cours. Une étape déjà commencée (par exemple un enregistrement dans EA) va jusqu'à son terme.
- Une réponse compte **au plus 24 étapes**. Si l'assistant n'a pas terminé, il s'arrête pour éviter une boucle sans fin ; poursuivez dans un nouveau message.

The screenshot shows the Loonar Compass interface. On the left is a sidebar with icons for conversations, settings, and other functions. The main area is titled "In a new package 'Online shop', draw a BPMN diagram" and shows a list of steps the assistant is performing. The steps are numbered 1 through 9, with the current step being 9. The steps include analyzing and planning, reading the model database, and getting the diagram from EA. The interface also shows a search bar for conversations, a status bar at the bottom, and a footer with contact information.

Conversations +

Search conversations...

No saved conversations

In a new package "Online shop", draw a BPMN diagram

Claude is working

In a new package "Online shop", draw a BPMN diagram of the order process: the customer places an order, payment as an embedded subprocess (choose method, authorise, confirm), then shipping and a customer notification. Show me the diagram at the end.

14:49

☰ Claude is working · step 9 collapse

14:49:14 ✓ Claude is analyzing and planning (step 1) (3 s) → next: db_query, 365 response tokens

14:49:17 ✓ Reading the model database: SELECT package_id, name, parent_id, ea_guid FROM t_package WHERE name ILIKE '%online%' OR name ILIKE '%shop%' OR parent_id IS NULL ORDER BY package_id (0 s) → package_id | name | parent_id | ea_guid 80 | Online shop | 1 | {295F53C2-78B5-4a31-A4DB-172CB0837B05}

14:49:17 ✓ Claude is analyzing and planning (step 2) (2 s) → next: db_query, 219 response tokens

14:49:19 ✗ Reading the model database: SELECT o.object_id, o.name, o.object_type, o.stereotype, o.package_id FROM t_object o WHERE o.package_id IN (SELECT package_id FROM t_package WHERE package_id=8... (0 s) → one query at a time (no semicolons inside)

14:49:19 ✓ Claude is analyzing and planning (step 3) (1 s) → next: db_query, db_query, 201 response tokens

14:49:20 ✓ Reading the model database: SELECT o.object_id, o.name, o.object_type, o.stereotype, o.package_id FROM t_object o WHERE o.package_id = 80 ORDER BY 1 (0 s) → object_id | name | object_type | stereotype | package_id 652 | Order placed by customer | Event | StartEvent | 80 653 | Payment | Activity | Activity | 80 654 | Payment started | Event | StartEvent | ...

14:49:20 ✓ Reading the model database: SELECT d.diagram_id, d.name, d.diagram_type, d.package_id FROM t_diagram d WHERE d.package_id = 80 (0 s) → diagram_id | name | diagram_type | package_id 71 | Order process | Analysis | 80

14:49:20 ✓ Claude is analyzing and planning (step 4) (1 s) → next: ea_show_diagram, 134 response tokens

14:49:22 🛑 Getting diagram 71 from EA as an image (2 s)

Add a message: Claude will take it into account while working...

📎 Drag files into the window to attach them

claude-haiku-5-5 · mode ask

Answers and changes are prepared by an AI model (Claude by Anthropic) and may contain errors. Review them before you approve.

🛑 Claude is getting a diagram from EA...

Loonar Compass 0.3 · Loonar Mariusz Głowacki, ul. Gombrowicza 12, 73-110 Stargard · loonar.eu · mariusz@lonnar.eu

Les étapes de l'assistant pendant son travail

Évaluer les réponses et signaler depuis le chat

Sous chaque réponse de l'assistant se trouvent les boutons 👍 et 👎. Après 👎, choisissez un motif : « Réponse fausse ou imprécise », « Pas ce que j'ai demandé », « Trop lent » ou « Une erreur dans le modèle ». Le même menu contient « **Signaler un problème avec cette réponse** » : il ouvre le formulaire de signalement (chapitre 20) avec l'objet rempli et la réponse signalée jointe. Les évaluations vont uniquement dans les statistiques d'utilisation, si elles sont activées, et aident à décider quoi améliorer en premier.

Lorsque vous demandez quelque chose que l'assistant ne sait pas faire (par exemple un format de fichier ou une intégration non pris en charge), il le dit clairement et propose ce qu'il peut faire à la place. Sous une telle réponse apparaît le bouton «  **Demander cette fonction** » : il ouvre le formulaire de signalement avec l'objet « Demande de fonction : ... », où vous décrivez votre besoin. La demande arrive sur GitHub comme idée.

Questions, plan et sous-agents

- **Questions de clarification.** Lorsqu'un point important n'est pas clair avant une modification importante (quel packaging, quelle notation, quel nommage, quelle portée), l'assistant ne devine pas : il affiche dans le chat un petit formulaire « Questions pour vous », avec des choix ou un champ pour une réponse courte. Un choix peut être remplacé par votre propre texte dans le champ « Autre chose... ». **Envoyer les réponses** les envoie comme votre message suivant, qui commence par « Mes réponses : », et l'assistant poursuit.
- **Plan d'un travail plus important.** Pour un travail de trois étapes ou plus, l'assistant affiche son plan dans la carte des étapes, par exemple «  Plan : 1 sur 4 faits », avec les tâches marquées ✓ (faite), ► (en cours) et ○ (en attente). Le plan est mis à jour au fil du travail ; dans Compass dans EA, il apparaît aussi dans System Output (chapitre 10).
- **Sous-agents.** Pour une analyse plus vaste, l'assistant peut confier une tâche ciblée à un sous-agent : **impact** (ce qu'affecte une modification prévue), **review** (un packaging, un brouillon ou des modifications prévues confrontés aux bonnes pratiques et aux instructions de l'équipe) ou **document** (un projet de texte d'une section de document). Le sous-agent ne fait que lire le modèle, dans son propre contexte, et renvoie un résultat compact, de sorte que la conversation reste courte. L'étape affiche « Sous-agent (impact) : ... ».

Raisonnement et calculs

Deux commutateurs de la carte de message ne fonctionnent qu'avec Claude (avec OpenAI, ils ne s'affichent pas) ; le réglage est mémorisé.

-  **Afficher le raisonnement de l'assistant dans les étapes.** Un résumé du raisonnement du modèle apparaît dans les étapes sous «  Raisonnement de l'assistant ». Le modèle raisonne de toute façon ; le commutateur ne fait que l'afficher. L'ampleur du raisonnement reste réglée par l'effort de la carte **Modèle d'IA** (chapitre 5).
- **Σ Code dans le bac à sable** (désactivé par défaut). L'assistant peut exécuter du code dans le bac à sable d'Anthropic pour des calculs et l'analyse de données, par exemple les totaux d'une estimation ou les statistiques d'un export du modèle. Son résultat apparaît comme l'étape « Code dans le bac à sable ». Les données que l'assistant y transmet vont à Anthropic, comme la conversation (chapitre 18).
-  **Web** (désactivé par défaut). L'assistant peut chercher sur le web et lire des pages, par exemple la documentation d'un produit ou un article dont vous donnez l'adresse. Avec Claude, c'est Anthropic qui cherche et lit. Avec OpenAI, Compass lit lui-même uniquement les pages dont vous avez donné l'adresse dans la conversation (sans recherche) ; il ne lit jamais d'adresses de cet ordinateur ni du réseau de l'entreprise. Le paramètre `claude.web_domains` (domaines séparés par des virgules) restreint les sites autorisés. L'administrateur du hub peut désactiver la fonction pour tous (`"web_access": false` dans la politique d'équipe) ou limiter les domaines (`"web_allowed_domains"`). Chaque recherche et chaque page apparaissent dans les étapes de la réponse.

Que sait faire l'assistant ? L'assistant connaît les fonctionnalités de l'application grâce à la liste des nouveautés. Demandez-lui : « Que sais-tu faire dans cette application ? ».

7. Travail avec le modèle Enterprise Architect

Écrivez comme à un collègue qui connaît EA. Désignez les paquetages et les diagrammes par leur nom. L'assistant désigne les objets du modèle par leur nom, leur paquetage et leur GUID. Voici quelques exemples de demandes ; des exemples classés par objectif figurent au chapitre 8.

Lecture

Quels paquetages y a-t-il dans le modèle et combien de diagrammes contient chacun d'eux ?

Décris ce que représente le diagramme « Processus de commande » et indique les éléments sans connecteurs.

Affiche le diagramme « Cas d'utilisation de la boutique ».

Liste les attributs et les opérations de la classe « Commande » avec leurs paramètres.

Le diagramme affiché apparaît dans le chat sous forme d'image. En dessous se trouvent les boutons **Enregistrer sous...** et **Ouvrir en taille réelle**.

Analyse d'impact d'une modification

L'assistant part de l'élément indiqué et suit les relations en profondeur dans le modèle (jusqu'à 6 niveaux, dans un sens ou dans les deux, éventuellement uniquement selon certains types de relations). Pour chaque élément concerné, il indique la relation par laquelle l'impact arrive, le chemin depuis l'élément de départ et les diagrammes sur lesquels figure l'élément. Vous pouvez obtenir le résultat sous forme de description, d'image, de document Word ou de classeur Excel.

Qu'est-ce qui changera si nous modifions le composant « Gestion des paiements » ? Vérifie 3 niveaux dans les deux sens et donne le résultat dans Excel.

Où les opérations sont utilisées

Pour une classe, un composant ou une interface, l'assistant trouve où ses opérations (services) sont appelées : sur les diagrammes de séquence (messages vers l'élément ou sa ligne de vie) et d'activité (actions appelant l'opération), avec le paquetage et l'appelant. Il indique aussi les opérations qui ne sont utilisées sur aucun diagramme.

Où sont utilisées les opérations de l'interface « API des commandes » ? Lesquelles ne sont utilisées nulle part ?

Dessin

Crée le paquetage « Boutique » sous le paquetage « Model » et dessines-y un diagramme de cas d'utilisation : le client parcourt le catalogue, ajoute un produit au panier, passe une commande et paie. Ajoute les acteurs et de courtes descriptions des cas.

Redessine le processus de la photo jointe sous forme de diagramme BPMN dans le paquetage « Processus ».

Extrais les exigences du document joint, crée-les dans le paquetage « Exigences » et affiche-les sur un diagramme d'exigences.

Sous-processus BPMN

- **Intégré (embedded)** : les étapes du sous-processus sont ses éléments enfants et sont dessinées dans son rectangle sur le même diagramme.
- **Réduit (collapsed)** : les étapes se trouvent sur un diagramme séparé, et un double-clic sur le sous-processus ouvre ce diagramme. L'assistant peut aussi lier un sous-processus (ou un autre élément) à un diagramme existant en tant que diagramme composite.
- L'assistant ajoute de nouvelles étapes à un sous-processus existant.

Dans le processus « Traitement des réclamations », remplace la tâche « Vérification » par un sous-processus intégré à trois étapes : vérifier les documents, vérifier la garantie, prendre une décision.

Modifications d'un modèle existant

Déplace les exigences FR01–FR05 dans le paquetage « Catalogue et panier ».

Sur le diagramme « Processus de commande », colore en vert les tâches exécutées par le système et en jaune les tâches manuelles.

Supprime du diagramme « Accueil » la cellule menant au diagramme « Archives ».

Ajoute un diagramme « Accueil » avec des tuiles-liens (navigation cells) vers tous les diagrammes de cas d'utilisation, quatre par ligne.

Renomme le diagramme « Brouillon » en « Processus de retour » et ajoute-lui une courte description.

Agrandis le pool « Entrepôt » sur le diagramme « Processus d'expédition » pour que toutes les tâches y tiennent.

L'assistant modifie aussi la position et la taille d'un élément sur un diagramme sans l'en retirer (les connecteurs sont conservés), retire un élément d'un diagramme en le laissant dans le modèle, et change le nom, la description et le stéréotype d'un diagramme existant.

Mots-clés et légendes

L'assistant définit les **mots-clés** des éléments (champ Keywords dans EA) et crée une **légende de diagramme avec coloration automatique**, comme dans EA : les éléments (selon le mot-clé, le stéréotype, le statut ou le type) et les connecteurs (par exemple selon le stéréotype) reçoivent une couleur de la légende, et la légende affiche la valeur, la couleur et le libellé.

Marque les exigences avec le mot-clé « MVP » ou « plus tard » et ajoute sur le diagramme une légende qui les colore selon ce mot-clé.

Lignes des connecteurs

- **Style de ligne** (comme Line Style dans EA) : direct, automatique, ligne libre, orthogonal carré, orthogonal arrondi, arbre et latéral ; pour les connecteurs sélectionnés ou pour tous ceux du diagramme.
- **Tracé d'un connecteur** : l'assistant fait passer la ligne par les points de passage indiqués et les points de contact avec les côtés des éléments, par exemple d'après un croquis sur une capture d'écran.
- **Masquage de connecteurs** sur un seul diagramme (et réaffichage) : la relation reste dans le modèle et sur les autres diagrammes, ce qui permet d'alléger un diagramme chargé sans supprimer de relations.

Sur le diagramme « Architecture », passe tous les connecteurs en orthogonal et masque les dépendances vers le composant « Connexion ».

Copie de paquetage et baseline

- **La copie de paquetage** crée un duplicata du paquetage avec tout son contenu.
- **Une baseline** est une version enregistrée d'un paquetage dans EA. Avant une suppression ou une restructuration importante d'un paquetage existant, l'assistant a pour consigne de créer une baseline. Ce n'est pas une garantie : pour les paquetages importants, demandez-lui explicitement une baseline et travaillez en mode **ask**.
- **La restauration** se fait dans Enterprise Architect : cliquez avec le bouton droit sur le paquetage, choisissez **Manage Baselines**, sélectionnez la baseline et utilisez **Compare/Restore**. L'assistant ne restaure pas lui-même les baselines, car la restauration par automatisation fige EA.

Glossaire du projet EA

L'assistant ajoute des termes au glossaire du projet EA (Project Glossary) : un terme avec sa définition et son type, par défaut le type **Glossary**. Un terme polonais est ajouté sous toutes ses formes fléchies (singulier et pluriel, du nominatif au vocatif), de sorte qu'EA met en évidence chacune de ses formes dans les notes. Vous ajoutez un terme en le demandant dans la conversation ou depuis le champ de message : cliquez avec le bouton droit sur un mot et choisissez **Ajouter « ... » au glossaire du projet EA**. Si l'assistant ne connaît pas la définition ou le type du terme, il vous les demande.

Ajoute au glossaire du projet le terme « commande » : demande d'un client pour la livraison de produits choisis.

Scénarios des éléments

L'assistant crée, modifie et supprime les scénarios des éléments, par exemple des cas d'utilisation. Ce sont des scénarios structurés, visibles dans EA dans l'onglet **Scenarios** de l'élément, avec des étapes exécutées par l'acteur ou par le système.

Ajoute au cas d'utilisation « Passer une commande » le scénario principal : le client confirme le panier, le système calcule le montant de la commande, le client choisit le paiement, le système enregistre la commande.

Instructions de modélisation de l'équipe

Le règlement de l'équipe pour un référentiel (nommage, types d'éléments et stéréotypes, structure des paquetages, diagrammes, tagged values obligatoires, contrôles de qualité) peut se trouver **dans le référentiel EA lui-même** : dans les notes d'un élément Artifact portant le stéréotype «CompassInstructions», nommé « Loonar Compass – modelling instructions ». Tous ceux qui ouvrent le référentiel avec Compass travaillent selon les mêmes règles, et l'historique des modifications est celui d'EA (audit, baselines, contrôle de version).

- L'assistant reçoit les instructions dans chaque réponse et les suit comme conventions du référentiel.
- **Assistant de démarrage** : lorsque le référentiel n'a pas encore d'instructions, l'assistant l'analyse (types d'éléments et stéréotypes, relations, diagrammes, paquetages, tagged values, statuts, exemples de noms) et présente un projet de règles. Il ne les enregistre qu'avec votre accord.
- **Modifications** : l'assistant remplace les instructions ou leur ajoute une règle, par exemple une correction que vous avez donnée. Chaque enregistrement est une modification du modèle et passe par la validation habituelle (chapitre 16). L'élément est créé dans le premier paquetage sous la racine du modèle, sauf si vous indiquez un autre paquetage.
- Les instructions comptent au plus 20 000 caractères. Vous pouvez aussi modifier vous-même les notes de l'élément dans EA.

Prépare des instructions de modélisation pour ce référentiel.

Ajoute aux instructions : les activités commencent par un verbe.

Les instructions du référentiel valent pour tous ceux qui travaillent avec lui ; la mémoire de l'assistant (chapitre 15) contient vos propres points sur cet ordinateur.

Contrôle de la qualité du modèle

D'une seule commande, l'assistant contrôle un paquetage avec ses sous-paquetages. Le contrôle est effectué par le programme, et non par le modèle d'IA : rien n'est deviné. Contrôles intégrés :

- noms par défaut et vides (par exemple Class1, Activity 2),
- éléments sans description,
- éléments orphelins (sans relations et sur aucun diagramme),
- noms en double dans un paquetage et relations en double,
- relations d'un élément avec lui-même et avec des éléments inexistants,
- relations qu'ArchiMate 3.2 n'autorise pas (voir ci-dessous),
- paquetages vides et diagrammes vides.

Chaque constat a une gravité : **erreur**, **avertissement** ou **information**. L'assistant n'apporte de corrections qu'avec votre accord.

Contrôle la qualité du paquetage Commandes.

Règles de l'organisation. Vous écrivez vos propres règles dans le fichier `governance.json`, placé à côté de `app.properties` (le fichier de paramètres du dossier de données de l'utilisateur). S'il n'existe pas et que Compass est connecté à un hub d'équipe, les règles proviennent de la politique de l'équipe (chapitre 22). Exemple :

```
{ "naming": [ { "applies_to": "Component", "pattern": "*Service", "message": "Un nom de composant se termine par Service" } ],  
  "required_tags": [ { "applies_to": "«ArchiMate_ApplicationComponent»", "tag": "Owner" } ],  
  "allowed_relations": [ { "connector_type": "Realisation", "source": "Component", "target": "Interface" } ] }
```

- **naming** : les noms des éléments du type indiqué doivent correspondre au motif (`pattern`) ; `message` est le texte du constat.
- **required_tags** : un élément du type indiqué doit avoir la tagged value indiquée.
- **allowed_relations** : un type de relation figurant ici n'est autorisé qu'entre les extrémités indiquées ; les types de relation qui n'y figurent pas ne sont pas restreints.

Dans `applies_to` , vous indiquez un type d'élément ou un «stéréotype» ; les jokers `*` et `?` fonctionnent.

Relations ArchiMate 3.2. Lorsqu'une relation et ses deux extrémités ont des stéréotypes ArchiMate (par exemple «ArchiMate_ApplicationComponent», «ArchiMate_DataObject»), le contrôle applique aussi les règles de relations d'ArchiMate 3.2, sans aucune configuration. Il ne signale que les violations certaines, comme erreurs ; les combinaisons qu'une relation dérivée peut rendre valides ne sont pas signalées. Par exemple :

- une relation d'accès (access) doit aboutir à un élément de structure passive (objet métier, objet de données, artefact...) et partir d'un élément de comportement ou de structure active,
- une relation d'influence doit aboutir à un élément de motivation,
- les relations de déclenchement (triggering), de flux (flow) et de service (serving) ne relient pas d'éléments de structure passive ou de motivation,
- une affectation (assignment) va d'une structure active vers un comportement (ou d'un nœud vers un artefact),
- une spécialisation relie deux éléments du même type,
- un élément de motivation ne réalise qu'un autre élément de motivation.

Les jonctions (junction) et les regroupements (grouping) ne sont pas contrôlés. Le contrôle en continu de Compass dans EA (chapitre 10) applique les mêmes règles. Pour désactiver les règles ArchiMate, indiquez `"archimate": false` dans `governance.json` .

Vérifie dans le paquetage Paysage applicatif quelles relations ArchiMate 3.2 n'autorise pas.

Analyse des écarts existant / cible

L'assistant compare deux paquetages, par exemple l'état actuel et l'état cible, et indique ce qui est **nouveau**, ce qui est **à retirer** et ce qui a **changé** (stéréotype, description, relations). Vous pouvez enregistrer le résultat dans Excel.

Compare le paquetage « Boutique existant » avec le paquetage « Boutique cible » et enregistre les écarts dans Excel.

Chaîne des exigences : modèle, code, tests, tickets

Lorsque vous comparez des exigences au modèle (section 8.5) et indiquez aussi un dossier de code source, l'assistant (l'outil `trace_requirements`) suit chaque exigence plus loin : exigence → élément du modèle → code → test → clés de tickets. Il lit le dossier en lecture seule.

- **Code** : les fichiers qui déclarent un type nommé comme l'élément du modèle tracé, et les fichiers source qui citent l'identifiant de l'exigence.
- **Tests** : les fichiers de test (dans un dossier de tests ou nommés comme `OrderTest.java`, `order.spec.ts`, `test_order.py`) qui citent l'identifiant de l'exigence ou portent sur un type de ce code.
- **Tickets** : les clés telles que `ARCH-123` dans l'exigence et dans les éléments tracés (nom, alias, notes, tagged values).

Chaque exigence reçoit un statut : **complète**, ou rompue : **absente du modèle**, **pas de code** ou **pas de tests**.

Suis les exigences de C:\Documents\BRD.docx à travers le modèle jusqu'au code et aux tests de C:\repo\shop : où la chaîne se rompt-elle ?

CMDB et modèle

L'assistant compare les éléments de configuration (CI) d'une CMDB au modèle (l'outil `cmdb_compare`). Il ne fait que lire : rien n'est écrit ni dans le modèle ni dans la CMDB.

- **Données** : un export CSV ou JSON, par exemple un export de liste de ServiceNow ou le JSON de sa Table API (`{"result": [...]}`), donné comme chemin ou pièce jointe, ou les données qu'un serveur MCP de la CMDB a renvoyées dans la conversation. Les colonnes telles que le nom (name), sys_id, la classe (class), le responsable (owned_by) et le statut (install_status) sont reconnues par leurs en-têtes.
- **Association** : par l'identifiant CMDB conservé dans une tagged value de l'élément (`sys_id` ou `CMDB id`), sinon par le nom.
- **Éléments** : par défaut les composants, nœuds, équipements, environnements d'exécution ainsi que les composants applicatifs, nœuds, logiciels système et services ArchiMate. Vous pouvez indiquer d'autres types ou stéréotypes et limiter la comparaison à un paquetage.
- **Rapport** : seulement dans la CMDB, seulement dans le modèle, et responsable (tagged value Owner) ou statut différent.

Compare l'export CMDB C:\Exports\cmdb_ci_appl.csv avec les composants du modèle.

Récupère les CI applicatifs dans ServiceNow et compare-les avec le modèle.

La seconde demande nécessite un serveur MCP pour ServiceNow sur la carte **Serveurs MCP** (voir « Vos propres serveurs MCP » ci-dessous). Sur demande, l'assistant ajoute au modèle les éléments, responsables ou identifiants manquants (avec votre accord) ou enregistre le rapport dans Excel. Dans un pipeline, l'option `--cmdb` effectue la même comparaison (chapitre 23).

Lecture de documents

L'assistant lit les documents de votre ordinateur que vous nommez dans la conversation (l'outil `read_document`), dans les mêmes formats que les pièces jointes (chapitre 6) : PDF (chaque page), PowerPoint .pptx (diapositives avec notes), Visio .vsdx (pages, formes avec leur texte et connexions A → B), Excel .xlsx (feuilles et cellules), Word .docx, HTML et fichiers texte. Un fichier qui n'est pas une pièce jointe de la conversation fait d'abord l'objet d'une demande. Un PDF sans couche de texte (un scan) ne peut pas être lu ainsi : joignez-le à la conversation, et le modèle lit ses pages comme des images. Les anciens formats (.doc, .ppt, .xls, .vsd) doivent d'abord être enregistrés dans le nouveau format. Les documents de la bibliothèque SharePoint de l'équipe se lisent de la même façon (chapitre 13).

Lis le processus dans C:\Documents\Reclamations.vsd et modélise-le sous forme de diagramme BPMN dans le paquetage Processus.

Les exigences pour la comparaison avec le modèle (section 8.5) et pour la chaîne des exigences peuvent aussi provenir de fichiers PDF, PowerPoint, Visio et Excel, dans la conversation comme en mode de contrôle (--requirements , chapitre 23).

Tickets Jira

Une fois la carte **Jira** remplie (chapitre 5), l'assistant lit les tickets par clé ou par recherche JQL et les utilise comme contexte : il dessine des diagrammes d'après la description d'un ticket (story) ou vérifie que le modèle couvre ses exigences. **Toujours avec votre accord dans le panneau de validation**, il crée des tickets, ajoute des commentaires et des liens vers des éléments du modèle.

L'assistant **modifie aussi les tickets existants** : résumé, description et étiquettes (ajout et retrait), par exemple il corrige les critères d'acceptation de plusieurs tickets d'un coup. Une story ou un cas d'utilisation avec un **epic comme parent** est créé par le champ parent sur Jira Cloud et par le champ **Epic Link**, qu'il trouve lui-même, sur Jira Server et Data Center. Pour une série de tickets, cliquez dans le panneau de validation sur **Approuver tous les tickets Jira jusqu'à la fermeture de Compass** : les tickets et modifications suivants passent sans demander.

Champ de la carte « Jira »	Que saisir
Adresse Jira	Cloud : <code>https://entreprise.atlassian.net</code> ; Data Center : l'adresse du serveur de l'entreprise. Elle doit commencer par <code>https://</code> .
E-mail (Cloud uniquement)	Cloud : l'e-mail du compte Atlassian. Avec un jeton d'accès personnel (Data Center), laissez vide.
Jeton	Cloud : le jeton API du compte (créé comme pour Confluence, chapitre 13). Data Center : un jeton d'accès personnel. Le jeton est conservé dans le Gestionnaire d'identification Windows.
Clé de projet par défaut	Par exemple <code>ARCH</code> : le projet où les nouveaux tickets sont créés lorsque vous n'en indiquez pas d'autre.
Certificat client, Mot de passe du certificat	Seulement si le serveur de l'entreprise exige un certificat client en plus du jeton (mTLS ; sans lui, le serveur ferme la connexion) : le fichier <code>.p12.pfx</code> (bouton « ... ») et son mot de passe, conservé dans le Gestionnaire d'identification Windows. Un fichier PEM avec le certificat et la clé privée convient aussi.

Dessine un diagramme de séquence d'après le ticket ARCH-123.

Ajoute au ticket ARCH-123 un lien vers l'élément « Panier ».

Vérification d'un enregistrement

Après chaque enregistrement, Compass relit le modèle et le compare à la demande : les objets créés existent-ils, les nouvelles relations ont-elles les extrémités demandées, les nouveaux éléments sont-ils dans le paquetage demandé, les noms, notes, stéréotypes et tagged values modifiés ont-ils les nouvelles

valeurs, et les objets supprimés ont-ils disparu. Les différences sont transmises à l'assistant, qui doit les signaler au lieu de dire « terminé ».

L'enregistrement est **tout ou rien** : s'il échoue à mi-chemin, les objets déjà créés sont de nouveau supprimés. La suppression d'un élément qui a des relations demande toujours votre accord (aussi en mode **auto**), et le panneau de validation énumère ces relations.

Toujours avec une demande et tous les détails. L'écriture de fichiers (export XML, documents issus de modèles de documents EA) et l'import XML demandent toujours votre accord, y compris en mode **auto**, et le panneau affiche le chemin complet. Seuls les chemins complets sur un disque local sont acceptés : les partages réseau ainsi que les dossiers système, des programmes et des données d'application sont refusés. Un identifiant doit être un nombre. L'enregistrement d'un point dans la mémoire de l'assistant, l'enregistrement d'un modèle de document ainsi que l'enregistrement et l'exécution d'un script EA demandent aussi toujours votre accord, avec le texte ou le code complet (les textes de plus de 20 000 caractères sont refusés) ; un script s'exécute exactement tel qu'il a été affiché, avec ses inclusions **!INC**. Les actions de contrôle de version sur un paquetage et les envois vers SharePoint demandent aussi toujours votre accord. Quand l'assistant veut lire un fichier qui n'est pas une pièce jointe de la conversation (un document d'exigences, un dossier de code source), il vous le demande d'abord.

Vos propres serveurs MCP

Vos propres serveurs MCP donnent à l'assistant des outils supplémentaires, par exemple un système de tickets, un dépôt ou une base de connaissances de l'entreprise. Vous listez les serveurs dans le fichier `mcp_servers.json` à côté du fichier de paramètres et vous les testez dans la carte **Serveurs MCP**. Un serveur peut être un programme local (`command` et `args` , communication par stdio) ou un service HTTP (`url` et `headers` , Streamable HTTP) :

```
{ "servers": [{ "name": "hub", "url": "http://127.0.0.1:8730/mcp", "headers": { "Authorization": "Bearer ${secret:hub_token}" } }, { "name": "files", "command": "npx", "args": [ "-y", "@modelcontextprotocol/server-filesystem", "C:/docs" ] ] }
```

- `${secret:NOM}` est une entrée du Gestionnaire d'identification Windows et `${env:NOM}` une variable d'environnement ; les jetons n'ont donc pas à figurer dans le fichier.
- Chaque appel d'un outil demande votre accord, sauf si le serveur a `"trusted": true`. Ne l'indiquez que pour les serveurs auxquels vous faites confiance : il ne suffit pas que le serveur déclare lui-même qu'un outil ne fait de lire.
- Les clés et mots de passe propres à Compass (les clés des fournisseurs d'IA, les mots de passe et jetons de la base, d'EA, de Confluence, de Jira et de Miro) ne peuvent pas être utilisés dans `${secret:...}`. La seule exception est `${secret:hub_token}` dans les en-têtes du point MCP propre au hub d'équipe (l'adresse du hub des Paramètres avec le chemin `/mcp`), comme dans l'exemple ci-dessus. Enregistrez les jetons des autres serveurs sous leurs propres noms dans le Gestionnaire d'identification Windows.
- Un programme local (`command`) ne reçoit que les variables d'environnement Windows de base (PATH, TEMP, les dossiers du profil utilisateur, les paramètres de proxy...) et les variables listées dans son `"env"`.
- `"command": "npx"` fonctionne : Compass lance `npx.cmd` (de même pour `npm`, `pnpm` et `yarn`). Les serveurs lancés avec `npx` nécessitent Node.js sur l'ordinateur.
- **Connexion autre que par un en-tête.** Un serveur HTTP peut avoir `"auth"` au lieu d'un en-tête `Authorization : { "type": "basic", "user": ..., "password": ... }` pour un nom d'utilisateur et un mot de passe, ou `{ "type": "oauth2", "token_url": ..., "client_id": ..., "client_secret": ...,`

"scope": ...} pour des identifiants client OAuth2 : Compass obtient un jeton à l'adresse `token_url` et l'envoie comme `Bearer`. Conservez les mots de passe et les secrets dans le Gestionnaire d'identification sous la forme `${secret:mcp_...}`.

- Le bouton **Enregistrer et tester les serveurs MCP** se connecte à chaque serveur et affiche le nombre de ses outils.

Serveurs prêts à l'emploi. Sur la carte **Serveurs MCP**, choisissez un serveur dans **Ajouter un serveur prêt à l'emploi** et cliquez sur **Ajouter** : son entrée est ajoutée au texte de `mcp_servers.json`, où vous la complétez avant d'enregistrer la carte. Un nom déjà utilisé reçoit un numéro.

Serveur	Ce qu'il faut compléter
Azure DevOps	Un serveur local lancé avec <code>npx</code> : remplacez <code>YOUR-ORGANIZATION</code> par votre organisation. Vous vous connectez avec votre compte Microsoft.
GitHub	Le serveur distant <code>https://api.githubcopilot.com/mcp/</code> : enregistrez un jeton d'accès personnel GitHub (personal access token) sous <code>mcp_github</code> .
ServiceNow / CMDB	Remplacez <code>YOUR-SERVICENOW-MCP-SERVER</code> par l'adresse du serveur MCP approuvé dans votre entreprise et enregistrez son jeton sous <code>mcp_servicenow</code> .
Files (a local folder)	Un serveur local lancé avec <code>npx</code> qui lit un dossier : remplacez <code>C:/docs</code> par votre dossier.
Compass Hub	Proposé seulement lorsque l'adresse du hub est renseignée sur la carte Équipe (Compass Hub) : le serveur MCP du hub avec votre jeton du hub, marqué <code>"trusted": true</code> .

Jeton d'un serveur MCP. Saisissez un nom qui commence par `mcp_` (lettres, chiffres et `_`, par exemple `mcp_github`) et le jeton, puis cliquez sur **Enregistrer et tester les serveurs MCP**. Le jeton est enregistré dans le Gestionnaire d'identification Windows ; dans `mcp_servers.json`, vous l'utilisez sous la forme `${secret:mcp_github}`. Le même champ enregistre le mot de passe ou le secret client d'un serveur avec `"auth"`, par exemple :

```
{
  "servers": [
    {
      "name": "wiki",
      "url": "https://mcp.example.com/mcp",
      "auth": {
        "type": "oauth2",
        "token_url": "https://login.example.com/oauth2/token",
        "client_id": "compass",
        "client_secret": "${secret:mcp_wiki}",
        "scope": "mcp.read"
      }
    }
  ]
}
```

Scripts EA

L'assistant énumère les groupes de scripts et les scripts du modèle, lit un script, enregistre un script (JScript, VBScript ou JavaScript ; un groupe manquant est créé) et l'exécute. **L'enregistrement et l'exécution d'un script demandent votre accord.** Une fois un script approuvé, les scripts suivants de même structure (le même code avec d'autres données) ne redemandent pas, et le bouton **Approuver tous les scripts jusqu'à la fermeture de Compass** supprime les questions sur les scripts jusqu'à la fermeture de l'application, y compris en mode **auto**, car un script peut tout modifier dans le modèle ; le panneau de validation affiche le code complet. Le script s'exécute dans la fenêtre EA ouverte ou, s'il n'y en a pas, dans la session EA en arrière-plan propre à Compass. `Repository` et `Session.Output` fonctionnent comme dans EA, et les inclusions `!INC Groupe.Script` sont résolues à partir du modèle. Si un autre modèle est ouvert dans EA, l'exécution s'arrête avec un message.

Liste les scripts du modèle, puis exécute le script « Rapport.Notes manquantes ».

Technologies MDG et API d'EA

- **Technologies MDG** (l'outil `mdg_technologies`) : les profils réellement utilisés par le modèle, et combien de fois ; les fichiers de technologies d'EA et les vôtres ; les technologies importées dans le modèle. Pour une technologie (par exemple ArchiMate3, BPMN2.0, SysML1.5), l'assistant liste ses stéréotypes avec la métaclasse et ce à quoi ils s'appliquent, et pour un stéréotype ses tagged values avec le type, les valeurs autorisées et la valeur par défaut. Il s'en sert avant de créer des éléments avec des stéréotypes d'une technologie et avant de remplir des tagged values.
- **L'API d'EA** (l'outil `ea_api`) : l'interface d'automatisation d'EA, lue dans la bibliothèque de types `EA.tlb` de l'EA installé sur cet ordinateur (le chemin d'EA.exe dans les Paramètres) : classes, propriétés (en lecture ou en lecture/écriture), méthodes avec leurs paramètres et énumérations. L'assistant la consulte avant d'écrire un script EA, afin que les noms des propriétés et des méthodes correspondent à votre version d'EA.

Quels stéréotypes d'ArchiMate3 notre modèle utilise-t-il, et quelles tagged values a «ArchiMate_ApplicationComponent» ?

Contrôle de version des paquetages

Lorsque des paquetages du modèle sont sous contrôle de version dans EA (SVN, Git, TFS ou un partage de fichiers), l'assistant lit leur état (l'outil `version_control_status`) : la configuration, le fichier, qui a extrait chaque paquetage, le dernier enregistrement et le dernier chargement. Sur demande, il effectue un **check out**, un **check in** (avec un commentaire), un **get latest**, un **put latest** ou un **resync** d'un paquetage. Chacune de ces actions **demande toujours votre accord**, dans tous les modes. Le paquetage doit déjà être sous contrôle de version dans EA (Package → Version Control → Add Package to Version Control) ; l'assistant ne met pas en place le contrôle de version.

Quels paquetages sont extraits, et par qui ? Fais un check out du paquetage Orders.

Autres opérations sur le modèle

- **Disposition automatique du diagramme** : direction vers le bas, la droite, le haut ou la gauche, et espacement.
- **XMI** : export d'un paquetage vers un fichier XMI et import d'un paquetage depuis un fichier (l'import remplace les GUID, sauf si vous demandez de les conserver).
- **Documents liés** des éléments, créés à partir d'un texte.
- **Couloirs (swimlanes)** d'un diagramme.
- **Inversion d'une relation** (échange de ses extrémités).
- **Modification d'un paquetage** : nom, notes, version, statut et stéréotype.
- **Document à partir d'un modèle de document EA** (DOCX, PDF ou RTF), par exemple le modèle « Model Report », comme Publish → Report Builder dans EA.

Dispose automatiquement le diagramme « Architecture » vers la droite et génère un DOCX à partir du modèle « Model Report » pour le paquetage Boutique.

Orthographe dans la langue de Compass

L'outil de l'assistant `spell_check` vérifie les notes des éléments et des diagrammes, les étapes des scénarios et le glossaire du projet avec le dictionnaire orthographique de Windows dans la langue de Compass (le correcteur d'EA n'a que des dictionnaires anglais). Les noms des éléments et les termes du glossaire comptent comme des mots connus. L'assistant propose des corrections et ne corrige qu'avec votre accord. Si Windows n'a pas de dictionnaire pour la langue, ajoutez la langue dans Paramètres Windows → Heure et langue → Langue. Dans Compass dans EA, les notes d'un élément modifié sont vérifiées en continu (lignes «  Orthographe : ... » dans System Output, chapitre 10).

Vérifie l'orthographe des notes du paquetage Boutique.

Bon à savoir

- **Le premier appel à EA prend de quelques secondes à plus de dix secondes**, car l'application lance Enterprise Architect en arrière-plan et ouvre le modèle. Les enregistrements et récupérations de diagrammes suivants prennent moins d'une seconde. Après 10 minutes sans appel, EA en arrière-plan se ferme et libère la licence. La fermeture de l'application ferme aussi sa session EA. Compass dans EA (chapitre 10) ne lance pas EA en arrière-plan : il enregistre par votre fenêtre EA.
- **Verrous.** Lorsque l'option « Le modèle exige un verrou pour modifier » est activée dans la carte Modèle, l'assistant verrouille un objet à son nom pendant l'enregistrement, puis lève le verrou. Il ne modifie pas un objet verrouillé par un autre utilisateur ; le message indique qui verrouille l'élément (compte ou groupe, depuis quand), son paquetage et son diagramme.
- **La suppression est définitive.** La suppression d'un paquetage supprime tout son contenu. Pour les modèles importants, utilisez le mode **ask** (chapitre 16), servez-vous des baselines et faites des sauvegardes de la base.
- **Enregistrement tout ou rien.** Si l'une des opérations d'un enregistrement échoue, les objets créés plus tôt par cet enregistrement sont de nouveau supprimés, de sorte qu'il ne reste rien d'inachevé dans le modèle ; le message indique la cause.
- **Vérifiez le résultat.** L'assistant peut se tromper : mal lire une photo, choisir un autre type d'élément ou disposer un diagramme de façon peu lisible. Demandez-lui d'afficher le diagramme après les modifications et d'apporter des corrections.
- **Les grands diagrammes** sont enregistrés par l'assistant en plusieurs étapes. C'est normal.

8. Exemples d'utilisation

Ce chapitre rassemble des exemples de demandes classés par objectif. Ils ont été essayés sur un modèle d'exemple comportant un paquetage « Shop domain », un processus « Order process » et un diagramme « Domain model » ; dans votre modèle, utilisez les noms de vos propres paquetages, diagrammes et éléments. L'assistant n'exige pas de formulation exacte : ce qui compte, ce sont les noms des objets et ce que vous attendez. Le détail des fonctions se trouve au chapitre 7, et les règles de validation des modifications au chapitre 16.

8.1 Interroger le modèle

La lecture du modèle ne change rien et ne demande aucun accord. Compass lit le modèle par la base de données ou le fichier EA, et l'assistant répond à partir de cette lecture en désignant les objets par leur nom, leur paquetage et leur GUID. L'analyse d'impact suit les relations en profondeur dans le modèle et indique aussi les diagrammes où figurent les éléments touchés.

Quelles classes se trouvent dans le paquetage Shop domain et quelles relations ont-elles ?

Qu'est-ce qui sera touché si nous supprimons la classe Product ? Indique les éléments et les diagrammes.

Ce que vous voyez : la carte des étapes avec les lectures du modèle, puis la réponse dans la conversation. Pour une question d'impact, chaque élément est accompagné de la relation par laquelle l'impact arrive, du chemin depuis l'élément de départ et de la liste des diagrammes. Aucune fenêtre de validation n'apparaît.

Astuce : ajoutez « donne le résultat dans Excel » si la liste est longue : vous obtenez un classeur au lieu d'un texte dans la conversation.

8.2 Créer et modifier le modèle

L'assistant crée des paquetages, des diagrammes, des éléments et des relations, et modifie ceux qui existent : noms, notes, tagged values, position sur un diagramme. Il n'enregistre dans EA qu'après votre accord (en mode **ask** avant chaque enregistrement, en mode **plan** avant le premier enregistrement d'une réponse ; chapitre 16).

Crée un paquetage Benchmark contenant un diagramme de classes Catalogue avec les classes Category et Item ; Category agrège Item.

Dans le processus Order process, ajoute l'activité Check fraud entre Payment et Ship order.

Ajoute à toutes les classes du paquetage Shop domain la tagged value Owner de valeur Team A.

Ce que vous voyez : au-dessus du champ de message, le panneau « **L'assistant veut enregistrer des modifications dans le modèle** » avec la description des modifications et une partie « **Avant → après :** » : ce qui est créé, ce qui change de quelle valeur à quelle valeur (champ : ancienne → nouvelle valeur) et ce qui est supprimé. Pour la troisième demande, vous y voyez la modification de chaque classe. **Valider** enregistre les modifications ; **Refuser** ne change rien et vous permet de dire à l'assistant ce qu'il doit faire autrement.

- **Vérification après chaque enregistrement.** Compass relit le modèle et le compare à la demande (les objets existent-ils, les relations ont-elles les extrémités demandées, les nouvelles valeurs sont-elles enregistrées). L'assistant ne rapporte que ce que le modèle confirme et signale les écarts au lieu de dire « terminé ».
- **Tout ou rien.** Si un enregistrement échoue en cours de route, les objets déjà créés sont de nouveau supprimés.
- **La suppression d'un élément ayant des relations** demande toujours votre accord, y compris en mode **auto**, et le panneau énumère ces relations.

Astuce : pour une modification en masse comme la troisième demande, lisez « Avant → après » avant de valider : chaque modification y est visible.

8.3 Annuler des modifications

Chaque enregistrement de l'assistant est consigné dans un journal d'annulation (la « machine à remonter le temps ») avec les opérations qui l'inversent. L'annulation supprime les objets créés et rend leurs valeurs précédentes aux champs modifiés, aux tagged values, aux positions et aux déplacements. Elle porte sur les enregistrements de l'assistant dans ce modèle ; les modifications faites à la main dans EA n'en font pas partie.

Annule la dernière modification.

Montre l'historique des modifications de l'assistant dans ce modèle.

Ce que vous voyez : pour l'historique, une liste des enregistrements, du plus récent au plus ancien, avec ce qui peut être annulé. Pour une annulation, un panneau de validation « Annuler les derniers enregistrements de l'assistant (...) » avec la liste des enregistrements (il apparaît dans tous les modes) ; après validation, Compass annule les enregistrements et vérifie le résultat dans le modèle. S'il n'y a rien à annuler, l'assistant indique qu'il n'y a rien à annuler dans ce modèle.

Les suppressions, les imports et la mise en page automatique des diagrammes ne peuvent pas être inversés par des opérations. Avant un tel enregistrement, Compass crée dans EA une baseline du paquetage concerné. Vous la restaurez dans Enterprise Architect : cliquez avec le bouton droit sur le paquetage, choisissez **Manage Baselines**, sélectionnez la baseline et utilisez **Compare/Restore**.

Astuce : vous pouvez annuler plusieurs enregistrements à la fois (« Annule les trois dernières modifications ») ; demandez d'abord l'historique pour voir ce que l'annulation va couvrir.

8.4 Qualité du modèle et règles d'équipe

Le contrôle de qualité est exécuté par le programme, pas par le modèle d'IA, donc rien n'est deviné : noms par défaut et vides, éléments sans description, éléments orphelins, doublons, relations vers des éléments inexistants, relations non autorisées en ArchiMate 3.2, paquetages et diagrammes vides, ainsi que les règles du fichier `governance.json` (ou de la politique d'équipe dans le hub). L'orthographe des notes est vérifiée avec le dictionnaire Windows de la langue de Compass.

Vérifie la qualité du paquetage Shop domain et énumère les problèmes les plus importants.

Vérifie l'orthographe des notes dans tout le modèle.

Exemple de fichier `governance.json` court (il se trouve à côté de `app.properties`) qui exige la tagged value Owner sur chaque classe :

```
{"required_tags":[{"applies_to":"Class","tag":"Owner"}]}
```

Ce que vous voyez : un résumé du contrôle et des constats avec une gravité : **erreur**, **avertissement** ou **information**. Pour l'orthographe : le nombre de textes et de mots vérifiés et la liste des fautes possibles avec des suggestions. Une fenêtre de validation n'apparaît que lorsque vous demandez des corrections. Si Windows n'a pas de dictionnaire pour la langue de Compass, ajoutez la langue dans Paramètres Windows → Heure et langue → Langue.

Astuce : après le contrôle, demandez de corriger un groupe de constats à la fois, par exemple « Complète les descriptions manquantes des classes du paquetage Shop domain ». L'enregistrement passe par la validation et la vérification habituelles.

8.5 Un document d'exigences face au modèle

L'assistant lit des exigences dans un document (Word `.docx`, PDF, PowerPoint `.pptx`, Visio `.vsdx`, Excel `.xlsx`, `.txt`, `.md`, une pièce jointe de la conversation, une page Confluence par son ID ou un texte collé) et les met en regard du modèle. Une exigence est une ligne ou une ligne de tableau qui commence par un identifiant, par exemple W1, R.1.2, REQ-12, FR-01. Le rapprochement est fait par le programme : il cherche les éléments portant cet identifiant (dans le nom, l'alias, les notes, les tagged values), les relations de traçabilité et les mots proches. Les candidats sont évalués par l'assistant.

Compare les exigences du document C:\Documents\BRD.docx avec le modèle : qu'est-ce qui est couvert et qu'est-ce qui manque ?

Compare les exigences de la page Confluence 123456 avec le modèle et enregistre la matrice de traçabilité dans Excel.

Crée les relations de traçabilité pour les exigences couvertes.

Ce que vous voyez : pour un fichier qui n'est pas une pièce jointe de la conversation, d'abord la question de savoir si l'assistant peut le lire ; puis un résumé « Traçabilité : ... exigences – couvertes ..., seulement des candidats ..., NON couvertes ..., même identifiant mais contenu différent ... » et la liste des exigences avec l'un des quatre statuts : **couverte**, **candidats**, **NON COUVERTE** ou **MÊME ID, CONTENU DIFFÉRENT**. Une liste distincte s'intitule « Éléments du modèle auxquels aucune exigence ne se réfère ». L'assistant signale aussi les exigences qui contredisent le modèle. La matrice Excel arrive sous forme de carte de fichier avec les boutons **Ouvrir** et **Enregistrer sous...**. Les relations de traçabilité (Realisation ou Dependency «trace») ne sont créées qu'après votre accord, dans le panneau de validation habituel.

Astuce : une page Confluence peut être lue une fois la carte **Confluence** des Paramètres remplie (chapitre 5). Le rapprochement est le plus fiable lorsque l'identifiant de l'exigence figure dans le nom, l'alias, la note ou la tagged value de l'élément ; sans lui, il ne reste que des candidats. Vous pouvez limiter le périmètre à un paquetage : « ...avec le paquetage Shop domain ». Ajoutez un dossier de code source pour suivre les exigences jusqu'au code, aux tests et aux tickets (chapitre 7, « Chaîne des exigences »).

8.6 Le modèle face au code

L'assistant lit un dossier de code (par exemple un clone local d'un dépôt) en lecture seule et le compare au modèle : classes, interfaces, énumérations et records avec leurs champs (Java, Kotlin, C#, TypeScript, Rust,

Python), tables et colonnes issues du SQL (`CREATE TABLE`) et opérations d'API (`operationId` d'OpenAPI) par rapport aux classes, attributs, tables et opérations du modèle.

Compare le modèle avec le code du dossier C:\repo\shop – que manque-t-il dans le modèle et que manque-t-il dans le code ?

Ce que vous voyez : d'abord la question de savoir si l'assistant peut lire le dossier, puis un résumé « Modèle ↔ code : ... fichiers source » et les sections **Dans le code, pas dans le modèle, Dans le modèle, pas dans le code, Champs différents (code vs attributs du modèle), Tables de base de données et Opérations API (OpenAPI operationId) absentes du modèle.** Aucune validation d'enregistrement n'apparaît, car rien n'est écrit : ni dans le code ni dans le modèle.

Astuce : limitez la comparaison à un paquetage (« ...avec le paquetage Shop domain ») si le dossier contient de nombreux services, et demandez un rapport dans Excel. Ajouter au modèle les classes manquantes est une demande distincte, qui passe par la validation habituelle.

8.7 Estimer une modification

À partir de l'impact d'une modification dans le modèle, l'assistant établit une première estimation : les éléments à modifier, les éléments liés à adapter, les éléments plus éloignés à vérifier et les diagrammes à mettre à jour deviennent des lignes de WBS en langage fonctionnel simple, avec des heures selon le type d'élément. S'y ajoutent les tests, l'analyse et la documentation, et une marge de risque.

Estime la modification de la classe Product : que faut-il changer et combien de temps cela prendra-t-il ?

Enregistre cette estimation dans Excel.

Ce que vous voyez : un tableau WBS avec des lignes telles que « Modification : Product », « Adaptation : ... », « Vérifier l'impact sur : ... », « Mettre à jour le diagramme : ... », « Tests de la modification », « Analyse et documentation » et « Marge de risque », un total en heures et en jours-personne, « Heures par type » et « Risques » (par exemple un élément ayant beaucoup de relations, sans description, ou une modification touchant de nombreux paquetages). Le fichier Excel est une carte de fichier avec les boutons **Ouvrir** et **Enregistrer sous....**

Astuce : c'est une base issue du modèle, pas une offre : ajustez les heures avec la connaissance de l'équipe. Les normes horaires se trouvent dans le fichier `estimate.json`, à côté du fichier de paramètres ; s'il manque, les normes par défaut s'appliquent.

8.8 Documents et exports

L'assistant prépare des documents Word à partir du modèle et de ses diagrammes (chapitre 13). Enterprise Architect génère aussi un rapport à partir d'un de ses modèles de documents EA (DOCX, PDF ou RTF) et exporte des paquetages en XML. Le rapport et l'export XML demandent toujours votre accord, y compris en mode **auto**.

Prépare un document Word décrivant le paquetage Shop domain avec le diagramme Domain model.

Génère un rapport à partir du modèle EA Model Report pour le paquetage Shop domain dans le fichier C:\Rapports\shop.docx.

Exporte le paquetage Shop domain dans le fichier XMI C:\Export\shop.xml.

Ce que vous voyez : pour le document Word, une carte de fichier avec les boutons **Ouvrir** et **Enregistrer sous...** Pour le rapport et le XML, un panneau de validation avec une description (« documents issus de modèles EA », « paquetages exportés en XML ») et le chemin complet du fichier, puis, après l'enregistrement, le message « Rapport enregistré : ... » ou « XML enregistré : ... » avec le chemin ; Compass vérifie que le fichier a bien été créé.

Astuce : le rapport doit avoir l'extension `.docx`, `.pdf` ou `.rtf`, et le fichier XML `.xml` ou `.xmi`. Indiquez un chemin complet sur un disque local : les partages réseau ainsi que les dossiers système, des programmes et des données d'application sont refusés. Un modèle portant le nom indiqué doit exister dans EA ; sinon EA ne crée pas le rapport et l'assistant le signale.

8.9 Diagrammes

Outre le dessin de nouveaux diagrammes, l'assistant améliore ceux qui existent : il les met en page automatiquement (vers le bas, la droite, le haut ou la gauche, avec un espacement donné), ajoute des couloirs (swimlanes) et crée des documents liés (linked documents) aux éléments à partir d'un texte donné.

Mets en page automatiquement le diagramme Domain model de gauche à droite.

Ajoute au diagramme Order process les couloirs Ventes et Entrepôt.

Ajoute à la classe Order un document lié avec le texte : Commande client – en-tête et lignes.

Ce que vous voyez : le panneau de validation habituel avec la description de la modification, puis, après l'enregistrement, la vérification du modèle. Si le même modèle est ouvert dans EA, le diagramme modifié s'actualise de lui-même dans votre fenêtre (chapitre 9).

Astuce : après la modification, demandez « Montre le diagramme Domain model » : vous le voyez sous forme d'image dans la conversation et jugez si la mise en page est lisible. Une mise en page automatique ne peut pas être inversée par des opérations ; Compass crée donc d'abord une baseline du paquetage (section 8.3).

8.10 Scripts EA

L'assistant écrit des scripts EA (JScript, VBScript ou JavaScript) et les exécute dans le modèle : ils affichent des données, vérifient quelque chose ou modifient de nombreux objets à la fois. Un groupe de scripts manquant est créé automatiquement. Le script s'exécute dans la fenêtre EA ouverte ou, s'il n'y en a pas, dans une session EA distincte en arrière-plan.

Crée et exécute un script EA en JScript qui affiche les noms de tous les paquetages du modèle.

Ce que vous voyez : d'abord un panneau de validation pour l'enregistrement du script (groupe, nom, langage et code complet), puis une question distincte sur l'exécution : les deux apparaissent à chaque fois, y compris en mode **auto**, car un script peut tout modifier dans le modèle. Le script s'exécute exactement tel qu'il a été affiché, avec ses inclusions `!INC`. Le résultat (ce que le script a écrit via `Session.Output`) arrive dans la conversation sous la forme « Sortie du script ... ». Si un autre modèle est ouvert dans EA, l'exécution se termine par un message.

Astuce : avant de valider l'exécution, lisez le code du script dans la fenêtre de validation, surtout s'il doit modifier quelque chose. Commencez par des scripts en lecture seule.

8.11 Jira et Confluence

Une fois les cartes **Jira** et **Confluence** des Paramètres remplies (chapitre 5), l'assistant lit les tickets et les pages comme contexte : il vérifie quels éléments du modèle un ticket concerne, dessine des diagrammes à partir de sa description ou compare les exigences d'une page au modèle (section 8.5).

Lis le ticket ARCH-123 et dis quels éléments du modèle il concerne.

Crée dans Jira une tâche : renommer l'élément Class1 du paquetage Shop domain.

Ce que vous voyez : à la lecture, la lecture du ticket dans la carte des étapes et une réponse avec la liste des éléments. La création de tickets, de commentaires et de liens vers des éléments du modèle demande **toujours** votre accord dans le panneau de validation, dans tous les modes. La publication d'une page dans Confluence demande elle aussi toujours l'accord et indique clairement lorsqu'elle remplace une page existante (chapitre 13).

Astuce : indiquez la clé du ticket et le nom du paquetage. Si vous ne précisez pas de projet, le nouveau ticket est créé dans le projet par défaut de la carte Jira.

8.12 Travailler en équipe (Compass Hub)

Avec un hub d'équipe (chapitre 22 ; carte **Équipe (Compass Hub)** des Paramètres), vous pouvez d'abord **proposer** une modification du modèle à l'équipe : l'assistant n'enregistre rien dans le modèle, il envoie au hub un ensemble de modifications avec une description et un aperçu « avant → après ». L'équipe le commente dans la vue web, et un architecte ou un administrateur l'approuve ou le rejette. Personne, pas même un administrateur, ne peut approuver son propre ensemble de modifications. Vous appliquez un ensemble approuvé dans Compass.

Propose à l'équipe une modification à relire : ajoute la classe Warehouse au paquetage Shop domain.

Montre les ensembles de modifications approuvés et applique CH-...

Ce que vous voyez : après la proposition, le message « Ensemble de modifications pour l'équipe : ... » avec son numéro (CH-...) et, dans la vue web du hub, une nouvelle ligne dans l'onglet **Change sets**. À l'application : le panneau de validation habituel et la vérification après l'enregistrement ; le hub marque l'ensemble comme appliqué. Seul un ensemble approuvé qui concerne ce modèle peut être appliqué, et une seule fois : Compass le réserve d'abord dans le hub et signale le résultat après l'enregistrement.

Vue web. Les personnes sans Enterprise Architect s'y connectent avec un nom d'utilisateur et un mot de passe fournis par l'administrateur du hub (adresse `https://<hub>/` ; `http://` seulement sur le même ordinateur). Les onglets portent des noms anglais : **Ask** (questions sur le modèle), **Search** (recherche d'éléments), **Model check** (contrôle du modèle selon les règles de l'équipe), **Gaps** (écarts existant / cible), **Audit trail** (journal des modifications de l'équipe), **Team notes** (notes partagées) et **Change sets** (modifications proposées avec commentaires et décision). Dans l'onglet **Ask**, vous saisissez une question avec vos propres mots, par exemple « Quels systèmes utilisent la passerelle de paiement ? ». Là, l'assistant ne fait que lire le modèle : il n'a accès ni aux fichiers, ni à Jira, ni à Confluence, ni aux serveurs MCP, et ne peut rien proposer ni enregistrer. Il répond à une question à la fois (une question compte au plus 2000 caractères), et les questions sont consignées dans le journal du hub.

Astuce : donnez à la proposition un titre qui permette au lecteur de comprendre le périmètre de la modification sans ouvrir les détails. Un ensemble de modifications appartient à un modèle : vous ne

pouvez l'appliquer que dans un Compass relié au même modèle du hub.

8.13 Outils supplémentaires issus de serveurs MCP

Vos propres serveurs MCP donnent à l'assistant des outils supplémentaires : un système de tickets, un dépôt, une base de connaissances ou le serveur MCP du hub lui-même. Vous déclarez les serveurs dans le fichier `mcp_servers.json` (chapitre 7) ou en ajoutez un prêt à l'emploi (Azure DevOps, GitHub, ServiceNow / CMDB, Files, Compass Hub), et vous les vérifiez dans les Paramètres, sur la carte **Serveurs MCP**, avec le bouton **Enregistrer et tester les serveurs MCP**.

Via le serveur MCP hub, énumère les sources de modèles de l'équipe.

Ce que vous voyez : dans la carte des étapes, l'appel de l'outil du serveur, puis son résultat dans la réponse. Chaque appel d'un outil demande votre accord, sauf si le serveur porte `"trusted": true` ; ne l'indiquez que pour les serveurs auxquels vous faites confiance.

Astuce : nommez dans la demande le serveur de `mcp_servers.json` (« via le serveur MCP hub... ») pour que l'assistant sache lequel utiliser. Conservez les jetons sous leurs propres noms dans le Gestionnaire d'identification de Windows (`secret:nom`), pas dans le fichier ; les clés et mots de passe propres à Compass ne peuvent pas y être utilisés (chapitre 7).

8.14 Compass dans Enterprise Architect

Lorsque Compass est ancré dans EA (chapitre 10), il surveille le modèle pendant que vous travaillez et n'a besoin d'aucune demande. Les résultats apparaissent dans l'onglet **Loonar Compass** de la fenêtre **System Output** :

- **Contrôle en continu** : après le tracé d'une relation et après la modification d'un élément (nom, stéréotype, tagged values), une ligne «  Contrôle du modèle : ... » apparaît, selon les mêmes règles qu'à la section 8.4.
- **Orthographe** : les notes d'un élément modifié sont vérifiées dans la langue de Compass et donnent «  Orthographe : ... ».
- **Agents d'événements** : vos propres règles dans le fichier `event_agents.json` (carte **Agents d'événements (Compass dans EA)**) déclenchent l'assistant après un événement dans EA. Ses remarques commencent par .

Exemple de règle qui vérifie le propriétaire et la description après la création d'un composant :

```
{ "agents": [ { "name": "Component owner", "event": "element_created", "applies_to": "Component",  
  "prompt": "Check that the new component has an owner and a description; suggest what to fill in." } ] }
```

Ce que vous voyez : dans System Output, une ligne telle que «  Component owner : ... » avec les remarques de l'assistant. L'agent **ne fait que lire le modèle** : il n'a accès ni aux fichiers, ni à Jira, ni à Confluence, ni aux serveurs MCP, et ne peut rien proposer ni enregistrer. Un même agent réagit au même objet au plus une fois par minute. Un nouvel élément n'est vérifié qu'après sa première modification, car EA lui donne d'abord le nom Class1.

Astuce : rédigez des demandes d'agent courtes et concrètes et limitez-les avec `applies_to` (un type, un «stéréotype» ou `*`). Après avoir modifié le fichier, cliquez sur **Enregistrer les agents d'événements**. Un agent peut aussi surveiller les suppressions et les arrêter (section 8.22).

8.15 Contrôler le modèle dans un pipeline CI

Le mode de contrôle exécute le contrôle du modèle, la comparaison avec le code et avec la CMDB, et la chaîne des exigences sans fenêtre et sans modèle d'IA, puis se termine par un code de sortie compris par un pipeline (chapitre 23).

```
LoonarCompass.exe --check --model model\shop.qea --rules governance.json --code src --requirements docs\exigences.md --junit junit.xml --fail-on error --lang fr | Out-Host
```

Ce que vous voyez : un rapport Markdown sur la sortie standard, qui commence par le résultat, PASSED ou FAILED, suivi des sections du contrôle du modèle, du modèle et du code, et des exigences. Dans le pipeline, le fichier JUnit montre chaque groupe de constats comme un test, en échec lorsque sa gravité atteint ou dépasse `--fail-on`. Le code de sortie vaut 0 lorsque le contrôle est réussi, 1 en présence de tels constats et 2 pour des arguments erronés ou un modèle illisible.

Astuce : commencez par `--fail-on error` et gardez les avertissements dans le rapport ; passez à `warning` lorsque le modèle est propre. Gardez `governance.json` dans le dépôt et transmettez-le avec `--rules`, pour que le pipeline et les personnes appliquent les mêmes règles.

8.16 Comparer la CMDB et le modèle

L'assistant compare les éléments de configuration d'une CMDB (un export ou les données d'un serveur MCP) aux composants du modèle et montre ce qui manque de chaque côté et où le responsable ou le statut diffère (chapitre 7, « CMDB et modèle »).

Compare l'export CMDB C:\Exports\cmdb_ci_appl.csv avec les composants du modèle.

Récupère les CI applicatifs dans ServiceNow et compare-les avec le modèle.

Ce que vous voyez : pour un fichier qui n'est pas une pièce jointe de la conversation, d'abord la question de savoir si l'assistant peut le lire ; avec un serveur MCP pour ServiceNow, l'appel de son outil (avec une demande d'accord, sauf si le serveur est de confiance). Puis un résumé « CMDB ↔ modèle : ... éléments de configuration, ... éléments du modèle, associés ... » et les sections **Dans la CMDB, absent du modèle**, **Dans le modèle, absent de la CMDB** et **Responsable ou statut différent**. Rien n'est enregistré sans votre accord.

Astuce : conservez l'identifiant CMDB (`sys_id`) dans une tagged value des éléments : les entrées restent associées même après un renommage. Limitez la comparaison à un paquetage ou indiquez les types d'éléments lorsque le modèle contient plus que la CMDB.

8.17 Tickets Jira et pages Confluence depuis le hub

Lorsque l'administrateur a configuré `integrations` dans `hub.json` (chapitre 22), les architectes et les administrateurs créent des tickets Jira et des pages Confluence directement depuis la vue web du hub, sans Compass et sans EA. Le hub utilise son propre compte de service.

- **Search** → cliquez sur un élément → **Create Jira issue** : saisissez le résumé ; le ticket reçoit les détails de l'élément.
- **Change sets** → ouvrez un ensemble de modifications → **Create Jira issue for this change set** : un ticket par ensemble de modifications.
- Choisissez un paquetage dans l'arborescence → **Model check** → **Publish this package to Confluence** : saisissez le titre (vide : le nom du paquetage et du modèle).

Ce que vous voyez : « Created ARCH-... » avec le lien vers le ticket ; dans l'ensemble de modifications, une ligne « Jira: ARCH-... » ; pour Confluence, « Created: » ou « Updated: » avec un lien vers la page. Chaque action est consignée à votre nom dans l'onglet **Audit trail**.

Astuce : la page Confluence est un résumé textuel du paquetage (éléments avec leurs descriptions et relations, sans diagrammes) remplacé à chaque publication : modifiez le modèle, pas la page. Pour une page avec des diagrammes, publiez depuis Compass (chapitre 13).

8.18 Instructions de modélisation de l'équipe

Les règles de l'équipe pour un référentiel sont conservées dans le référentiel lui-même, dans les notes d'un Artifact «CompassInstructions» (chapitre 7, « Instructions de modélisation de l'équipe »). L'assistant les suit dans chaque réponse, pour tous ceux qui ouvrent le référentiel.

Prépare des instructions de modélisation pour ce référentiel.

Ajoute aux instructions : les activités commencent par un verbe.

Ce que vous voyez : pour la première demande, l'analyse du référentiel dans les étapes (« Instructions de modélisation : ... »), puis un projet de règles sous des titres (nommage, types d'éléments et stéréotypes, structure des paquetages, diagrammes, relations, tagged values, contrôles de qualité) et la question de savoir s'il faut l'enregistrer. L'enregistrement, ainsi que la règle ajoutée par la seconde demande, passent par le panneau de validation habituel comme une modification de l'élément « Loonar Compass – modelling instructions ».

Astuce : après une correction que vous avez donnée dans la conversation, demandez à l'assistant de l'ajouter aux instructions : elle vaut alors pour toute l'équipe, et pas seulement pour vous. Les préférences personnelles ont leur place dans la mémoire de l'assistant (chapitre 15).

8.19 Documents : PDF, PowerPoint, Visio, Excel

L'assistant lit les documents que vous joignez ou que vous nommez : PDF, Word, PowerPoint avec les notes, Visio avec les formes et les connexions, Excel avec les feuilles (chapitre 7, « Lecture de documents »).

Lis le processus dans C:\Documents\Reclamations.vsd et modélise-le sous forme de diagramme BPMN dans le paquetage Processus.

Compare les exigences du PDF joint avec le paquetage Shop domain.

Ce que vous voyez : pour un fichier qui n'est pas une pièce jointe, d'abord la question de savoir si l'assistant peut le lire, puis l'étape « Lecture du document ... ». Pour le dessin Visio, l'assistant propose les éléments et les flux à partir des formes et des connexions ; l'enregistrement passe par la validation habituelle.

Astuce : un PDF scanné ne contient pas de texte : joignez-le à la conversation au lieu d'indiquer son chemin, et le modèle lit les pages comme des images. Enregistrez d'abord les anciens fichiers .doc, .ppt, .xls et .vsd dans le nouveau format.

8.20 Présentations et PDF

L'assistant crée des présentations PowerPoint à partir du modèle (diapositive de titre, diapositives avec des puces et des diagrammes) ainsi que des documents Word, y compris au format PDF (chapitre 13).

Prépare une présentation de 5 diapositives sur le packaging Shop domain, avec le diagramme Domain model, aussi en PDF.

Prépare une description du processus Order process sous forme de document PDF.

Ce que vous voyez : l'étape « Préparation de la présentation ... » et une carte de fichier avec les boutons **Ouvrir** et **Enregistrer sous...** ; avec le PDF, une seconde carte pour le fichier PDF. Le PDF est créé par Microsoft PowerPoint ou Word sur cet ordinateur.

Astuce : sans Microsoft Office, la présentation et le document sont tout de même créés en PPTX et DOCX ; seul le PDF manque, et l'assistant le signale.

8.21 Travaux plus importants : questions, plan, sous-agents

Avant une modification importante, l'assistant demande ce qui n'est pas clair, affiche son plan et peut confier des analyses à des sous-agents (chapitre 6, « Questions, plan et sous-agents »).

Modélise le processus de retour : un diagramme BPMN, les classes du domaine et les cas d'utilisation.

Avant de modifier le composant Payment, fais analyser l'impact par un sous-agent.

Ce que vous voyez : un formulaire « Questions pour vous » (par exemple quel packaging, quelle notation), avec des choix et le champ « Autre chose... ». Après **Envoyer les réponses**, le plan «  Plan : 0 sur 4 faits » dans les étapes, mis à jour avec ✓ ► ○ au fil du travail. Le travail d'un sous-agent apparaît comme l'étape « Sous-agent (impact) : ... », suivie de son résultat compact.

Astuce : si vous connaissez déjà les réponses, mettez-les dans la demande (« dans le packaging Returns, en BPMN 2.0 ») : l'assistant pose alors moins de questions.

8.22 Un gardien qui arrête une suppression

EA consulte un agent d'événements doté de `"mode": "guard"` avant qu'un objet soit supprimé, et cet agent peut arrêter la suppression (chapitre 10, « Agents d'événements »).

```
{ "agents": [ { "name": "Keep approved", "event": "element_deleting", "mode": "guard", "applies_to": "Requirement", "prompt": "Block deleting a requirement whose status is Approved." } ] }
```

Ce que vous voyez : lorsque vous supprimez une exigence approuvée dans EA, le message « Loonar Compass a arrêté cette suppression : » avec la raison, et l'exigence reste en place. Dans System Output, les lignes «  Keep approved : bloqué - ... » ou «  Keep approved : autorisé ».

Astuce : un gardien répond en quelques secondes ; s'il ne décide pas dans les 40 secondes (ou ne le peut pas, par exemple sans IA), la suppression est arrêtée, afin qu'un objet protégé ne soit jamais perdu par accident. Rédigez des demandes de gardien qui disent clairement quand bloquer.

8.23 Contrôle de version des paquetages

Pour les paquetages sous contrôle de version dans EA, l'assistant indique qui a extrait quoi, et il extrait et archive les paquetages (chapitre 7, « Contrôle de version des paquetages »).

Quels paquetages sont extraits, et par qui ?

Fais un check in du paquetage Orders avec le commentaire « Retours ajoutés ».

Ce que vous voyez : pour la première demande, la liste des paquetages sous contrôle de version avec leur configuration, leur fichier et leur état (« checked out to ... » ou « checked in »). Pour la seconde, un panneau de validation « Contrôle de version : check_in – paquetage Orders », qui apparaît dans tous les modes.

Astuce : placez d'abord le paquetage sous contrôle de version dans EA ; l'assistant ne travaille qu'avec les paquetages qui y sont déjà.

8.24 SharePoint

Une fois la carte **SharePoint** remplie (chapitre 13), l'assistant lit les documents de la bibliothèque de l'équipe et y envoie les documents et présentations qu'il a créés.

Liste les documents du dossier Exigences sur SharePoint et compare le BRD le plus récent avec le modèle.

Envoie la présentation que tu as préparée vers SharePoint, dossier Revues.

Ce que vous voyez : la liste du dossier, puis le texte du document lu comme à la section 8.19. Pour l'envoi, un panneau de validation « Envoyer ... vers SharePoint : ... » avec le site, la bibliothèque et le dossier ; il apparaît dans tous les modes.

Astuce : seuls les fichiers créés par l'assistant dans la conversation (dans son dossier des pièces jointes) peuvent être envoyés, et non n'importe quel fichier de votre disque.

8.25 Power BI et GraphQL depuis le hub

Le hub d'équipe fournit les modèles sous forme de tables plates et par une API GraphQL en lecture seule, pour Power BI, Excel et d'autres outils (chapitre 22, « GraphQL et tables pour Power BI »).

```
POST https://hub.example.com/api/sources/shop-db/graphql
{"query":"{ elements(type: \"Component\", limit: 20) { name stereotype connectors { type target { name } } } }"}"
```

Ce que vous voyez : une réponse JSON avec les composants et leurs relations. Dans Power BI : **Obtenir les données** → **Web** → **Avancé**, l'adresse `https://<hub>/api/sources/<id>/table/elements` et l'en-tête `Authorization` avec la valeur `Bearer <jeton API>`.

Astuce : demandez à l'administrateur du hub votre propre jeton API (onglet **Users**) ; tous les rôles peuvent lire.

Conseils pour de bonnes demandes

- **Nommez exactement les paquetages et les diagrammes**, tels qu'ils figurent dans le modèle (de préférence entre guillemets). Si deux objets portent le même nom, ajoutez le paquetage.

- **Dites s'il faut modifier ou seulement proposer.** « Propose, mais n'enregistre pas encore » donne une description des modifications sans enregistrement.
- **Pour une modification plus importante, demandez d'abord un aperçu** et travaillez en mode **plan** : le panneau de validation montre tout le « Avant → après » avant le premier enregistrement.
- **Demandez l'enregistrement du résultat dans Excel ou Word** lorsqu'il doit vivre en dehors de la conversation : listes, matrices et estimations.
- **Pour les modifications risquées, utilisez le mode ask** (Paramètres → « Langue et validation des modifications »), demandez une baseline du packaging et vérifiez le diagramme après les modifications.

9. Collaboration avec la fenêtre EA ouverte

Si vous avez **le même modèle** ouvert dans Enterprise Architect, l'assistant collabore avec votre fenêtre EA :

- **il sait ce que vous avez ouvert et sélectionné** : quel diagramme est ouvert et quels éléments sont sélectionnés sur le diagramme ou dans le navigateur de projet. Vous pouvez donc écrire « cet élément » ou « les tâches sélectionnées »,
- **il ouvre des diagrammes** dans votre fenêtre EA et **indique des éléments** dans le navigateur de projet,
- **après un enregistrement, il actualise lui-même** les diagrammes et paquetages modifiés dans votre fenêtre, sans appuyer sur **F5**.

Décris l'élément que j'ai sélectionné et montre sur quels autres diagrammes il apparaît.

Ouvre dans mon EA le diagramme « Processus de commande » et indique dans le navigateur le paquetage « Exigences ».

Lorsque la fenêtre EA n'est pas ouverte ou affiche un autre modèle, l'assistant fonctionne normalement, et vous verrez les modifications après avoir actualisé le paquetage dans EA (**F5**).

La collaboration décrite ici concerne l'application autonome. Compass lancé dans Enterprise Architect lui-même travaille toujours sur le projet ouvert dans sa fenêtre EA (chapitre 10).

Les lectures d'un fichier EA ouvert dans une fenêtre EA passent par cette fenêtre, car le fichier sur le disque peut ne pas encore contenir vos dernières modifications.

10. Compass dans Enterprise Architect

Loonar Compass fonctionne aussi à l'intérieur d'Enterprise Architect, comme une fenêtre ancrée à côté du navigateur de projet et des diagrammes. Vous y travaillez comme dans l'application autonome : la conversation, les pièces jointes, la validation des modifications, la mémoire, les modèles de documents et les documents fonctionnent de la même façon. L'application autonome, lancée depuis le menu Démarrer, fonctionne aussi comme avant.

Installation et lancement

1. Dans le programme d'installation, sur la page « Dossier de destination », cochez « **Ajouter Loonar Compass à Enterprise Architect** ». Cela fonctionne avec toute installation d'Enterprise Architect, y compris une version d'essai. Redémarrez EA après l'installation.
2. Ouvrez un projet dans Enterprise Architect : un fichier ou une base de données.
3. Dans le ruban **Start**, groupe **Add-Ins**, cliquez sur **Loonar Compass** (le bouton se trouve aussi sous **Specialize**). Le bouton n'est actif que lorsqu'un projet est ouvert.
4. Compass s'ouvre comme une fenêtre ancrée dans EA, au même endroit que la fenêtre Properties. Vous pouvez la faire glisser ailleurs, la détacher ou l'ajouter à un groupe d'onglets ; EA retient l'endroit où vous l'avez placée.

« **LoonarCompass – Error** » dans la gestion des compléments. Installé pour un seul utilisateur (par défaut), le complément n'est enregistré que pour cet utilisateur, et Windows n'utilise pas ces enregistrements dans les programmes lancés en tant qu'administrateur. Lancez Enterprise Architect normalement, pas avec « Exécuter en tant qu'administrateur » ; si EA doit s'exécuter en tant qu'administrateur, installez Compass pour tous les utilisateurs de l'ordinateur.

Projet et conversations

- Dans EA, Compass travaille toujours sur le projet ouvert dans cette fenêtre EA : un fichier ou n'importe quelle base de données. Il lit le modèle par l'intermédiaire d'Enterprise Architect lui-même ; il n'a donc besoin ni des paramètres du modèle, ni de mots de passe, ni du choix d'un modèle de conversation. Ouvrir un autre projet dans EA fait passer Compass à ce projet ; sans projet ouvert, Compass ne fonctionne pas.
- Vous choisissez la conversation dans la liste déroulante en haut de la fenêtre. La liste n'affiche que les conversations de ce projet. Le bouton + commence une nouvelle conversation, et le crayon renomme celle qui est sélectionnée.
- Les icônes des vues se trouvent en bas de la fenêtre ; dans une fenêtre étroite, les pages défilent horizontalement et verticalement.
- **Ctrl** + **V** colle des images, des fichiers et du texte, comme dans l'application autonome.
- Les paramètres dans EA n'ont pas de cartes Modèle et Enterprise Architect. Les cartes Modèle d'IA, Langue et validation des modifications, Confluence, Miro, Jira, SharePoint et Équipe (Compass Hub) fonctionnent comme dans l'application autonome (chapitre 5).

Modifications, diagrammes et références aux objets

- Les enregistrements passent par la même fenêtre EA : les modifications apparaissent aussitôt dans le navigateur de projet et sur les diagrammes. La validation des modifications fonctionne comme dans

l'application autonome (chapitre 16).

- **Afficher le diagramme** et **Modifier le diagramme** ouvrent le diagramme dans EA lui-même, au lieu d'afficher une image dans le chat.
- Cliquez avec le bouton droit sur un élément, un paquetage ou un diagramme (dans le navigateur de projet ou sur un diagramme) et choisissez **Specialize** → **Ajouter à la conversation Compass**. Une référence à l'objet est ajoutée au message, par exemple «Paquetage: Boutique (id 5)», et l'assistant sait de quel objet il s'agit. Le bouton ➦ de la carte de message ajoute les mêmes références en cherchant dans le modèle (chapitre 6, « Choisir dans le modèle »).

Décris «Paquetage: Boutique (id 5)» et propose les exigences manquantes.

Contrôle en continu

Lorsque Compass est ancré dans EA, il contrôle les modifications pendant votre travail selon les mêmes règles que le contrôle du modèle (celles intégrées et celles de `governance.json` ou du hub). Le contrôle démarre après le dessin d'une relation et après la modification d'un élément (nom, stéréotype, tagged values). Les constats apparaissent dans l'onglet **Loonar Compass** de la fenêtre System Output, chacun une seule fois, sous la forme «  Contrôle du modèle : ... ». Un nouvel élément n'est contrôlé qu'après sa première modification, car EA le nomme d'abord Class1.

La liste des conversations de l'onglet ne montre que celles du projet ouvert dans cette fenêtre. Le menu ... à côté de la corbeille propose **Conversations d'autres modèles...** (par exemple après le déplacement du projet d'une base de données vers un fichier : **Continuer ici** crée une copie de la conversation rattachée à ce projet ; l'original reste avec son modèle, et les notes de mémoire de l'ancien modèle peuvent être rattachées au nouveau), **Exporter la conversation...** et **Importer une conversation...** (un fichier `.lcchat`, chapitre 17).

Agents d'événements

La carte **Agents d'événements (Compass dans EA)** contient des règles selon lesquelles l'assistant réagit à ce qui se passe dans EA. Les règles se trouvent dans le fichier `event_agents.json` :

```
{"agents":[{"name":"Component owner","event":"element_created","applies_to":"Component",
  "prompt":"Check that the new component has an owner and a description; suggest what to fill in."}]}
```

- Événements (`event`), 18 au total :
 - éléments : `element_created`, `element_changed`, `element_deleting` ;
 - connecteurs : `connector_created`, `connector_changed`, `connector_deleting` ;
 - diagrammes : `diagram_created`, `diagram_changed`, `diagram_deleting` ;
 - paquetages : `package_created`, `package_changed`, `package_deleting` ;
 - attributs et méthodes : `attribute_created`, `attribute_deleting`, `method_created`, `method_deleting` ;
 - le glossaire du projet : `glossary_term_created`, `glossary_term_deleting` .

Dans `applies_to`, indiquez un type d'élément, «stéréotype» ou `*` (les jokers `*` et `?` fonctionnent).

- Les agents fonctionnent tant que Compass est ouvert dans son onglet d'EA. L'assistant lit le modèle et répond dans l'onglet **Loonar Compass** de System Output (les lignes commencent par ). **Il ne fait que lire le modèle** : il n'a accès ni aux fichiers, ni à Jira, ni à Confluence, ni aux serveurs MCP, et ne peut rien proposer ni enregistrer.

- Le même agent réagit au même objet au plus une fois par minute.
- Le bouton **Enregistrer les agents d'événements** enregistre les règles et affiche le nombre d'agents.

Gardiens. Par défaut, un agent conseille après l'événement ("mode": "advise"). Avec "mode": "guard", permis uniquement pour les événements *_deleting, EA consulte Compass avant de supprimer l'objet. Le gardien répond ALLOW ou BLOCK avec une raison. Avec BLOCK, EA affiche « Loonar Compass a arrêté cette suppression : » avec la raison et annule la suppression. Si le gardien ne décide pas dans les 40 secondes ou ne peut pas décider, la suppression est arrêtée avec un message. System Output affiche « 🍷 ... : autorisé » ou « 🍷 ... : bloqué - ... ».

```
{
  "agents": [
    {
      "name": "Keep approved",
      "event": "element_deleting",
      "mode": "guard",
      "applies_to": "Requirement",
      "prompt": "Block deleting a requirement whose status is Approved."
    }
  ]
}
```

Agents de l'équipe. Avec un hub d'équipe (chapitre 22), les agents de la politique de l'équipe (event_agents) fonctionnent dans le Compass dans EA de chacun, à côté des agents de cet ordinateur. Leurs noms portent la mention « (team) ».

Orthographe en continu. Les notes d'un élément modifié sont vérifiées dans la langue de Compass ; les remarques apparaissent dans System Output sous la forme « 🍷 Orthographe : ... ».

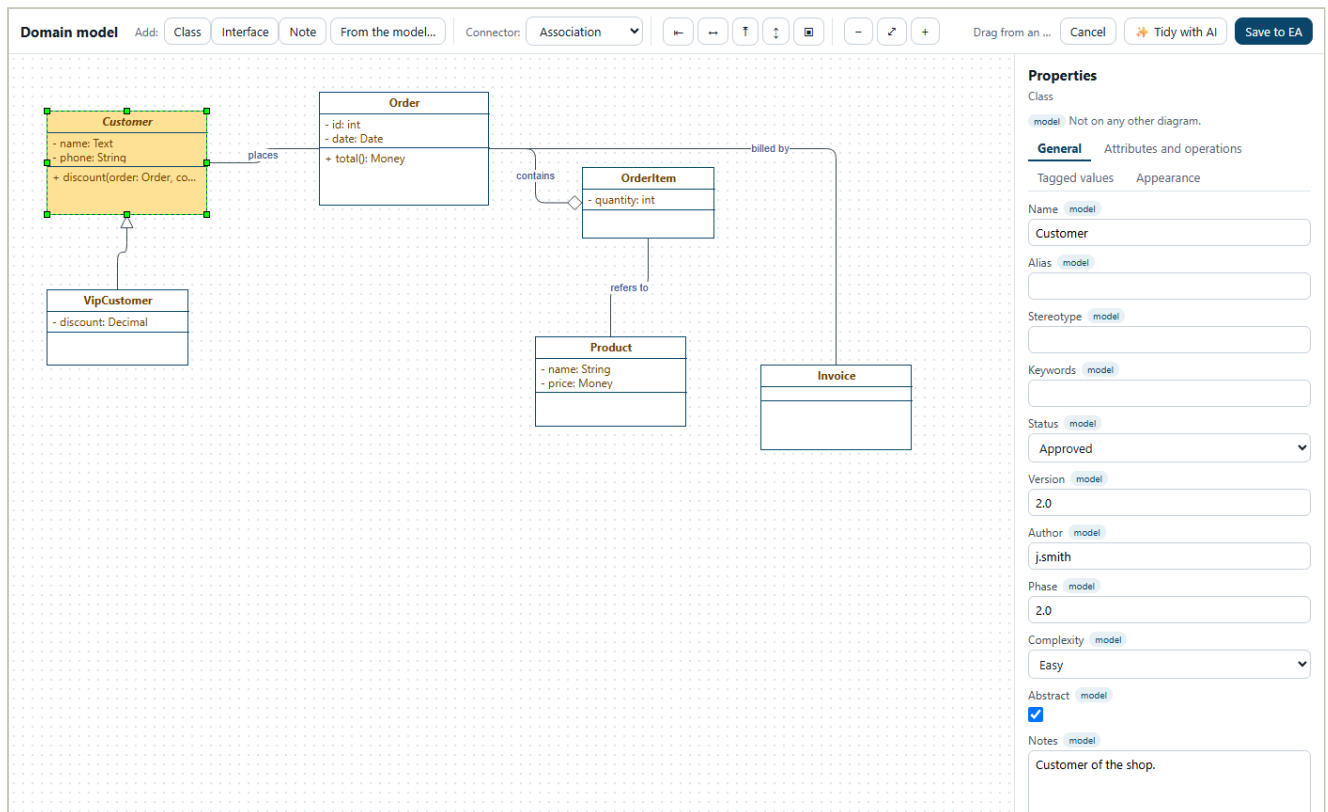
Étapes de l'assistant dans System Output

La fenêtre **System Output** d'EA comporte un onglet **Loonar Compass** : vos messages et chaque étape de l'assistant (lectures du modèle, enregistrements dans EA, résultats, le plan « 📄 Plan : ... » d'un travail plus important), avec l'heure. Pendant ce temps, le chat de la fenêtre Compass n'affiche que la dernière étape, sur une seule ligne.

Fermeture. Compass se ferme avec Enterprise Architect et enregistre la conversation. Après la fermeture du projet dans EA, Compass demande d'ouvrir un projet.

11. Éditeur de diagrammes

Chaque diagramme d'EA affiché dans la conversation a un bouton **Modifier le diagramme**. Il ouvre le diagramme dans une fenêtre d'éditeur où vous le modifiez à la souris. L'éditeur travaille sur le modèle, pas sur une image : ce que vous enregistrez arrive dans EA par les mêmes opérations que celles de l'assistant, avec le panneau d'approbation et le journal des modifications (chapitre 16). L'éditeur a besoin de WebView2, qui fait partie de Windows 10 et 11. Dans Compass dans EA, le bouton **Modifier le diagramme** ouvre le diagramme dans Enterprise Architect lui-même (chapitre 10).



L'éditeur de diagrammes : la barre d'outils, le diagramme et le panneau Propriétés

Dessiner

- **Déplacer, redimensionner et aligner** les éléments à la souris et avec les boutons d'alignement ; **Ctrl** + **Z** et **Ctrl** + **Y** annulent et rétablissent.
- **Ajouter** : les boutons dépendent du type de diagramme (classes, composants, cas d'utilisation, activités, BPMN, machine à états, déploiement, paquetages, séquence). Un nouvel élément va à un emplacement libre près du milieu de la vue ; tapez tout de suite son nom.
- **Relier** : faites glisser d'un élément à un autre. Le type de relation se choisit dans la liste **Connecteur**.
- **Depuis le modèle...** place dans le diagramme un élément qui existe déjà dans le modèle (recherche par nom, type ou paquetage). Ses relations avec les éléments du diagramme apparaissent après l'enregistrement.
- Un double-clic sur un élément ou une relation le renomme.

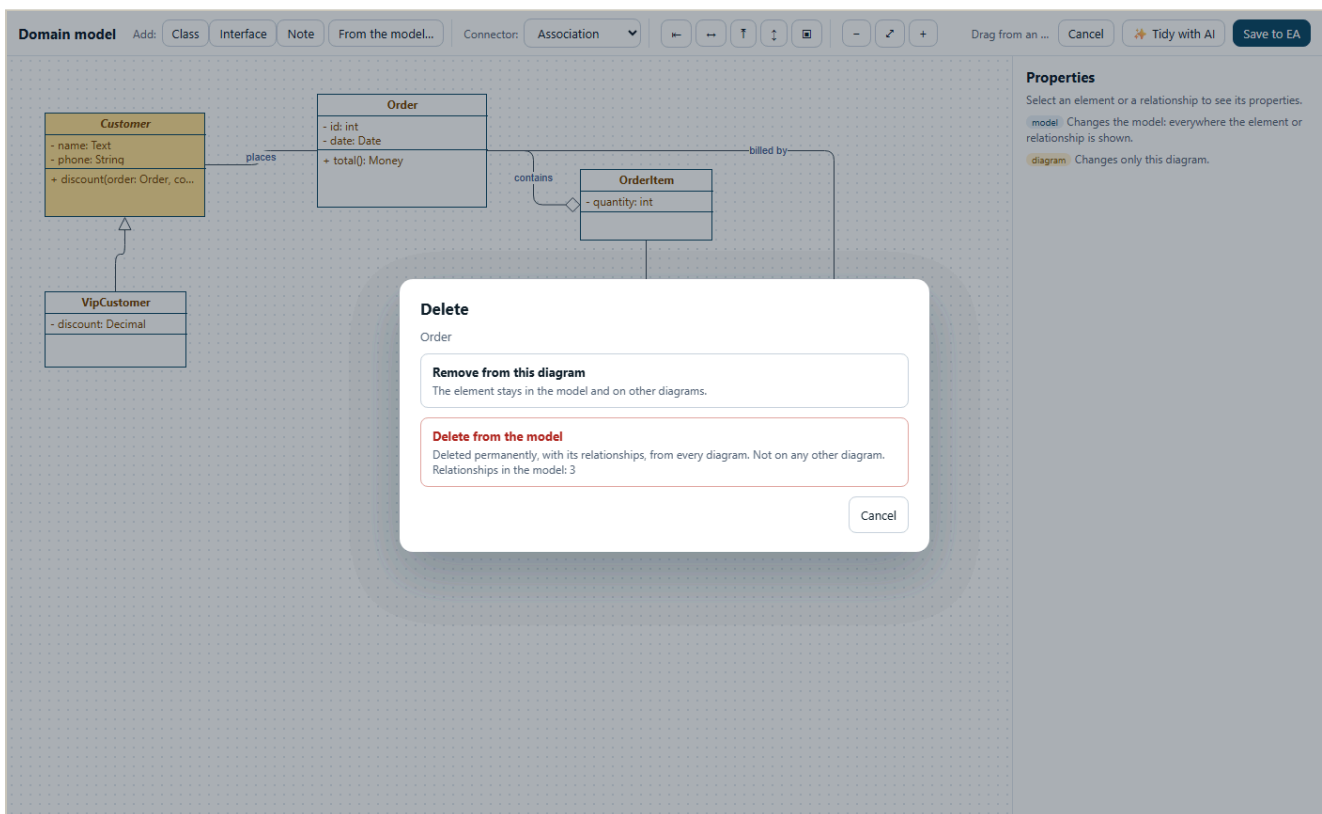
Propriétés

Le panneau **Propriétés** montre l'élément ou la relation sélectionné. Chaque champ porte la marque **modèle** (la modification vaut partout où l'objet apparaît) ou **diagramme** (uniquement ce diagramme).

- **Général** : nom, alias, stéréotype, mots-clés, statut, version, auteur, phase, complexité, abstrait et notes ; le paquetage, le GUID et les dates en lecture seule. Pour une relation : nom, stéréotype, direction, multiplicités et rôles des deux extrémités, notes.
- **Attributs et opérations** (classes, interfaces, composants) : ajouter, modifier (visibilité, nom, type) et supprimer ; une nouvelle opération prend ses paramètres sous la forme **nom : Type, autre : Type** .
- **Valeurs étiquetées** : modifier, ajouter et retirer ; quand le profil donne les valeurs permises (par exemple les types de tâches BPMN), vous choisissez dans une liste.
- **Apparence** : la couleur de remplissage dans ce diagramme ou la couleur par défaut de l'élément, et le style de ligne d'une relation dans ce diagramme.

Supprimer

La touche **Suppr** retire sans demander ce que vous avez ajouté dans l'éditeur. Pour un objet déjà présent dans le modèle, l'éditeur demande : **Retirer de ce diagramme** (il reste dans le modèle et dans les autres diagrammes) ou **Supprimer du modèle** (définitivement, avec ses relations, de tous les diagrammes). Il indique dans quels autres diagrammes figure l'élément et combien de relations il a.

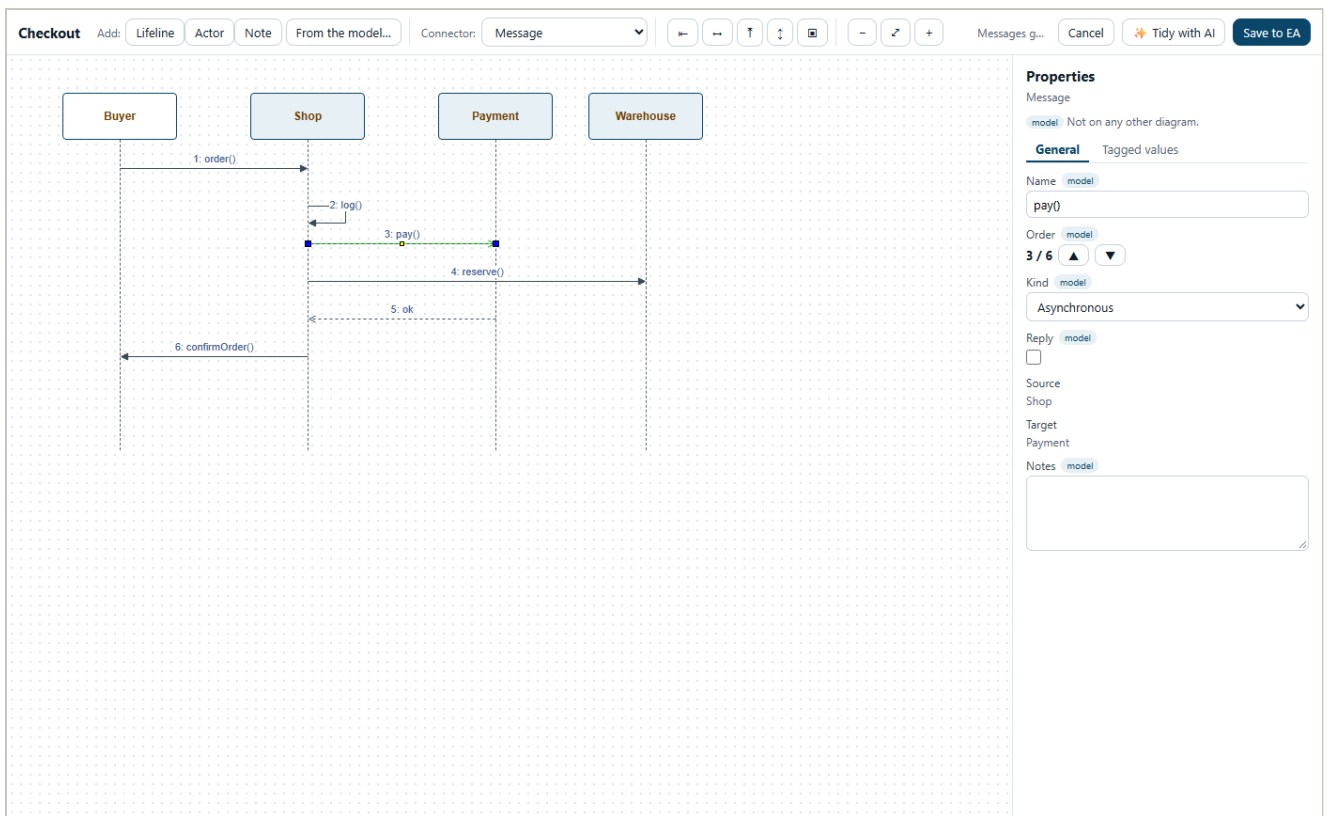


Supprimer : de ce diagramme ou du modèle, avec les autres diagrammes de l'élément

Diagrammes de séquence

Les lignes de vie ont un en-tête et une ligne pointillée ; elles ne se déplacent qu'horizontalement. Les messages vont de haut en bas dans leur ordre, avec leurs numéros, comme EA les dessine. Changez l'ordre du message sélectionné avec ▲ ▼ dans le panneau ou **Alt + ↑** / **Alt + ↓**. Un nouveau message (faites glisser d'une ligne de vie à une autre) se place juste après le message sélectionné. Dans le panneau,

vous choisissez le type (synchrone ou asynchrone), s'il s'agit d'une réponse, et vous pouvez le nommer d'après une opération du destinataire.



Un diagramme de séquence : messages numérotés, ordre modifié avec ▲ ▼

Enregistrer

- **Enregistrer dans EA** enregistre les modifications et ferme l'éditeur ; le résultat apparaît dans la conversation. **Annuler** le ferme sans enregistrer.
- **Réorganiser avec l'IA** enregistre les modifications et demande à l'assistant, dans la conversation, de disposer le diagramme pour qu'il se lise bien, sans rien ajouter, supprimer ni renommer.

12. Envoyer des diagrammes vers Miro

Un diagramme d'EA peut aller sur un tableau Miro sous forme de formes et de connecteurs modifiables, dans un cadre portant le nom du diagramme. Seulement d'EA vers Miro : ce qui est modifié sur le tableau ne revient pas dans le modèle.

Configuration, une seule fois :

1. Dans Miro, ouvrez les paramètres de votre profil → **Your apps** → **Create new app**. Miro demande d'abord de créer une **Developer team** (un bac à sable gratuit pour vos applications) : acceptez ses conditions et créez-la. Dans une organisation au plan Enterprise, c'est l'administrateur de l'organisation qui la met en place.
2. Donnez un nom à l'application (par exemple Loonar Compass) et **laissez « Expire user authorization token » décoché** : un jeton qui expire cesse de fonctionner au bout d'une heure.
3. Sous **Permissions**, cochez `boards:read` et `boards:write`.
4. Cliquez sur **Install app and get OAuth token** et choisissez l'organisation et l'équipe dont l'application peut utiliser les tableaux (pour un premier test, la Developer team). Copiez le jeton.

5. Collez-le dans les Paramètres, carte « **Miro** », puis cliquez sur **Enregistrer et tester la connexion à Miro**. Le jeton est conservé dans le Gestionnaire d'identification de Windows.

Les diagrammes ne peuvent aller que sur les tableaux de l'équipe où l'application est installée ; pour une autre organisation, installez-la aussi là-bas (avec le même bouton).

Envoi :

1. Cliquez sur **Envoyer vers Miro** sous un diagramme dans la conversation.
2. Choisissez le tableau (celui utilisé la dernière fois est proposé en premier ; le champ de recherche filtre par nom).
3. La miniature montre ce qui est déjà sur le tableau (en gris) et où ira le diagramme (un cadre bleu, rouge s'il recouvrait quelque chose). L'application trouve elle-même un emplacement libre ; faites glisser le cadre pour en choisir un autre.
4. Cliquez sur **Envoyer vers Miro**. Une fois l'envoi terminé, la conversation reçoit un message avec un lien qui ouvre le tableau sur le nouveau cadre.

Les diagrammes de séquence arrivent sous forme de lignes de vie avec des messages numérotés dans leur ordre d'EA (réponses en pointillé). Renvoyer le même diagramme place un nouveau cadre à côté du précédent : rien de ce qui a été modifié sur le tableau n'est écrasé.

13. Documents : Word, Excel, images, Confluence

L'assistant prépare des documents à partir du contenu du modèle et de la conversation. Le fichier terminé apparaît dans le chat sous forme de carte avec des boutons ; il est enregistré dans le dossier `attachments` des données utilisateur (chapitre 18).

Type	Contenu	Boutons dans le chat
Word (DOCX)	Titres, paragraphes, listes, tableaux, diagrammes d'EA et images dessinées auparavant dans la conversation	Ouvrir, Enregistrer sous...
Excel (XLSX)	Plusieurs feuilles, en-têtes avec filtre et première ligne figée, nombres enregistrés comme nombres ; par exemple une liste d'éléments, une matrice, le résultat d'une analyse d'impact	Ouvrir, Enregistrer sous...
PowerPoint (PPTX)	Une présentation au format 16:9 : une diapositive de titre, puis des diapositives avec un titre, des puces sur deux niveaux et/ou un diagramme d'EA ou une image dessinée dans la conversation	Ouvrir, Enregistrer sous...
PDF	Un document Word ou une présentation au format PDF, créés par Microsoft Word ou PowerPoint sur cet ordinateur	Ouvrir, Enregistrer sous...
Image (PNG)	Une matrice (par exemple des cas d'utilisation et des exigences), un tableau, un graphique, un schéma. L'assistant voit le résultat et peut le corriger.	Enregistrer sous..., Ouvrir en taille réelle
Page Confluence	Le code de la page pour l'éditeur de source, ainsi que les diagrammes et images dans un dossier <code><page>_images</code>	Copier le code de la page, Ouvrir le dossier des images, Enregistrer sous...

Prépare un document Word décrivant le packaging « Boutique » : liste des cas d'utilisation avec leurs descriptions et le diagramme de cas d'utilisation.

Dessine sous forme d'image la matrice des cas d'utilisation et des exigences du packaging « Boutique ».

Présentations et PDF

- **PowerPoint** : l'assistant construit la présentation (16:9) à partir du modèle et de la conversation : une diapositive de titre avec un sous-titre, puis des diapositives avec un titre, au plus six puces environ sur deux niveaux et/ou une image : un diagramme d'EA ou une image dessinée dans la conversation. Avec des puces, l'image se place à droite.
- **PDF** : sur demande, la présentation est aussi enregistrée en PDF par Microsoft PowerPoint, et un document Word par Microsoft Word (en PDF seul, ou en DOCX et PDF). Office travaille de façon

invisible en arrière-plan. Sans Office, le PPTX ou le DOCX est tout de même créé, et l'assistant indique que le PDF n'a pas pu l'être.

Prépare pour le comité de pilotage une présentation des modifications du packaging « Boutique », avec le diagramme Domain model, aussi en PDF.

SharePoint

L'assistant travaille avec une bibliothèque de documents du site SharePoint de l'équipe (l'outil `sharepoint`) : il liste un dossier, lit le texte d'un document (PDF, Word, PowerPoint, Visio, Excel) et envoie un fichier qu'il a créé dans cette conversation (un document, un classeur, une présentation). **Chaque envoi demande votre accord**, dans tous les modes ; les fichiers situés ailleurs sur votre disque ne peuvent pas être envoyés.

Champ de la carte « SharePoint »	Ce qu'il faut saisir
Adresse du site	Par exemple <code>https://societe.sharepoint.com/sites/Architecture</code> . Vide : l'assistant n'utilise pas SharePoint.
Bibliothèque de documents	Le nom de la bibliothèque ; vide = Documents
ID du locataire (annuaire)	Par exemple <code>contoso.onmicrosoft.com</code> ou un GUID
ID d'application (client)	L'ID de l'inscription d'application dans Microsoft Entra ID
Secret client	Le secret de cette inscription d'application ; conservé dans le Gestionnaire d'identification Windows

Inscription d'application. Demandez à votre administrateur Microsoft 365 une inscription d'application dans Microsoft Entra ID avec un secret client et l'autorisation **d'application** Microsoft Graph `Sites.Selected`, accordée à ce site (recommandé), ou `Sites.ReadWrite.All`. Compass se connecte en tant que cette application (identifiants client OAuth2) sur `login.microsoftonline.com` et travaille par Microsoft Graph (`graph.microsoft.com`). Enregistrez la carte avec **Enregistrer et tester la connexion SharePoint** : elle affiche « Enregistré. Connecté à SharePoint : ... éléments dans la bibliothèque. ».

Lis le document Exigences/BRD.docx depuis SharePoint et compare-le avec le modèle.

Pages Confluence

L'assistant prépare une page au format de l'éditeur de source Confluence (storage XHTML) : titres, tableaux, listes, table des matières, panneaux (info, remarque, avertissement, astuce, section dépliable), statuts, tickets Jira, mentions de personnes et blocs de code. Les diagrammes d'EA et les images sont ajoutés en pièces jointes.

Collage manuel (sans paramètres Confluence) :

1. Cliquez sur **Copier le code de la page** sur la carte de la page.
2. Dans Confluence, ouvrez la page en modification et l'éditeur de source (Insérer → Éditeur de source), collez le code et validez.
3. Cliquez sur **Ouvrir le dossier des images** et ajoutez les images à la page en pièces jointes.

Publication par l'assistant. Une fois la carte « **Confluence** » remplie, l'assistant publie la page à votre demande via l'API REST : il crée une nouvelle page sous la page parente ou met à jour une page existante (désignée par son ID ou son titre). Les images sont envoyées en pièces jointes ; celles qui existent déjà reçoivent une nouvelle version. Les diagrammes sont exportés à nouveau depuis EA à chaque publication, si bien qu'une nouvelle publication actualise la page après des modifications du modèle. L'assistant se souvient des pages qu'il a publiées.

La publication nécessite toujours une validation, y compris en mode **auto**. Avant l'envoi de la page à Confluence, un panneau avec les boutons **Valider** et **Refuser** apparaît. Il indique s'il s'agit d'une nouvelle page ou si la publication **REMPACERA** le contenu d'une page existante (même titre ou même ID) ; dans ce cas, la version précédente reste dans l'historique de la page.

Champ de la carte « Confluence »	Que saisir
Adresse Confluence	Sur les serveurs de l'entreprise (Data Center), par exemple <code>https://confluence.societe.fr</code> ; dans le cloud (Cloud) <code>https://societe.atlassian.net/wiki</code> . L'adresse doit commencer par <code>https://</code> ; <code>http://</code> n'est accepté que pour un serveur installé sur le même ordinateur.
E-mail (Cloud uniquement)	Pour Confluence Cloud : l'e-mail du compte. Pour Data Center, laissez vide.
Jeton	Data Center : jeton d'accès personnel (Profil → Personal Access Tokens). Cloud : jeton API du compte. Le jeton est conservé dans le Gestionnaire d'identification Windows.
Espace (clé)	La clé de l'espace, par exemple <code>ARCH</code>
Page parente (ID)	Facultatif : l'ID de la page sous laquelle les nouvelles pages sont créées
Certificat client, Mot de passe du certificat	Seulement si le serveur de l'entreprise exige un certificat client en plus du jeton (mTLS ; sans lui, le serveur ferme la connexion) : le fichier <code>.p12.pfx</code> (bouton « ... ») et son mot de passe, conservé dans le Gestionnaire d'identification Windows. Un fichier PEM avec le certificat et la clé privée convient aussi.

Confluence Cloud : d'où viennent les informations.

1. **Jeton** : il se crée sur votre compte Atlassian, pas dans Confluence. Ouvrez `https://id.atlassian.com/manage-profile/security/api-tokens` (compte Atlassian → Security → API tokens). Atlassian peut d'abord demander une confirmation par un code envoyé par e-mail.
2. Cliquez sur **Create API token** (le simple, pas **Create API token with scopes**), donnez-lui un nom, par exemple Loonar Compass, et choisissez sa validité : une semaine par défaut, un an au maximum. Cliquez sur **Create** et copiez le jeton : Atlassian ne l'affiche qu'une fois. Quand il expire, créez-en un nouveau et collez-le dans la carte « Confluence ».
3. **E-mail** : l'adresse du compte Atlassian auquel appartient le jeton.
4. **Adresse** : votre site suivi de `/wiki` , par exemple `https://votre-site.atlassian.net/wiki` .

5. **Clé de l'espace** : elle figure dans l'adresse de chaque page de l'espace, après `/spaces/` (par exemple `.../wiki/spaces/PM/...` donne `PM`), et dans les **paramètres de l'espace**.
6. **ID de la page parente** (facultatif) : le nombre dans l'adresse de la page, après `/pages/` (par exemple `.../pages/65818/...` donne `65818`).

Enregistrez la carte avec le bouton **Enregistrer et tester la connexion à Confluence**.

Prépare une page Confluence décrivant le processus « Traitement des réclamations » avec son diagramme et publie-la dans l'espace.

14. Modèles de documents

Un modèle indique à l'assistant à quoi doit ressembler un document : ses sections, leur contenu, des indications et des exemples. L'assistant le remplit avec le contenu du modèle EA, de la conversation, de Jira, de Confluence et du code, puis prépare une page Confluence, un document Word ou la description d'un ticket Jira. Vous pouvez avoir autant de modèles que vous voulez - un, quelques-uns ou cent - et chaque utilisateur a les siens. Compass ne fournit pas de modèles prêts à l'emploi : vous ajoutez ceux qu'utilise votre équipe.

Ajouter des modèles

- ✚ **Ajouter depuis des fichiers...** - choisissez un ou plusieurs fichiers : Markdown (.md), texte, Word (.docx), HTML, PDF, PowerPoint ou Excel. Les titres, listes et tableaux de Word et HTML sont conservés ; PDF, PowerPoint et Excel donnent leur texte. Avec un seul fichier, le formulaire du modèle s'ouvre ; plusieurs fichiers sont enregistrés directement, et vous définissez ensuite leur outil et leur groupe avec **Propriétés**.
- ✚ **Ajouter depuis Confluence** - collez l'adresse d'une page (avec `/pages/ID` ou `pageId=ID`) ou son id. La page exemple devient un modèle pour Confluence.
- ✚ **Nouveau modèle** - un formulaire vide ; vous écrivez le contenu en Markdown.
- Vous pouvez aussi demander à l'assistant de faire un modèle à partir d'un document exemple (voir plus bas).

Propriétés d'un modèle

Champ	Signification
Nom	C'est par lui que vous désignez le modèle dans la conversation.
Description	Ce qu'est le document et quand l'utiliser - l'assistant choisit le modèle d'après elle.
Outil	Confluence, Jira, Word ou tout outil.
Groupe	Un nom libre, p. ex. un niveau de documentation (général, métier, technique), un projet ou une équipe. La liste est classée par outil, puis par groupe.
Modèle IA	Modèle des paramètres ou Modèle plus puissant (demande d'abord) : quand vous demandez un document selon ce modèle, Compass propose un modèle IA plus puissant du même fournisseur (p. ex. Claude Sonnet au lieu de Haiku) pour cette réponse seulement, en indiquant combien de fois il coûte plus ; la réponse suivante utilise de nouveau le modèle des paramètres.
Contenu	Markdown : titres <code>#</code> , listes, tableaux. Les lignes commençant par <code>></code> sont des indications et des exemples pour l'assistant - elles ne vont pas dans le document.

Un modèle est un fichier Markdown dans le dossier `templates`. Ses propriétés sont dans ses premières lignes ; vous pouvez donc aussi préparer des fichiers vous-même et les déposer dans le dossier :

- > *Modèle: Document technique - architecture, interfaces, données, exigences non fonctionnelles*
- > *Cible: Confluence*
- > *Groupe: Niveaux de documentation*
- > *Modèle IA: plus fort*

1. Contexte architectural

- > *Systèmes autour de la solution. Exemple : ...*

La vue « Modèles de documents »

- **Utiliser** insère dans la conversation une demande de document selon le modèle (page Confluence, document Word ou ticket Jira - selon l'outil du modèle). Complétez-la et envoyez-la.
- **Propriétés** ouvre le formulaire avec le nom, la description, l'outil, le groupe, le modèle IA et le contenu.
- **Modifier** ouvre le fichier du modèle dans le Bloc-notes.
- **Supprimer** (deux clics) déplace le modèle dans le dossier `bin` du dossier des modèles - vous pouvez l'y récupérer.
- **Ouvrir le dossier des modèles** ouvre le dossier `templates` dans l'Explorateur.

Les modèles fournis par les versions précédentes (BRD, description de processus et autres) que vous n'avez pas modifiés vont dans le dossier `bin` après la mise à jour ; ceux que vous avez modifiés restent.

Comment l'assistant écrit un document

Dans la conversation, nommez le modèle ou son groupe :

Prépare une page Confluence selon le modèle « Niveau technique » pour les changements de PRJ-123 et les diagrammes du paquet Commandes.

- L'assistant liste d'abord vos demandes sur le document dans toute la conversation (sections, chaque exigence citée, sources, images, format) et lit toutes les sources indiquées, y compris les images jointes aux tickets Jira.
- Il suit les titres du modèle dans leur ordre ; chaque exigence a sa propre entrée avec son numéro et le texte de la source ; ce qu'il ne sait pas, il le marque « à compléter ».
- Compass ne crée pas un document auquel manque une section du modèle ou dans lequel une demande n'a pas sa place - l'assistant doit d'abord le compléter. Dans sa réponse, il montre la liste : demande → section.

Un modèle à partir d'un document exemple

L'assistant demande toujours votre accord avant d'enregistrer un modèle, aussi en mode **auto**, et affiche tout le texte.

Crée un modèle à partir du document joint : garde la structure des sections et ajoute ce que chacune doit contenir.

15. Mémoire de l'assistant

L'assistant retient les points convenus d'une conversation à l'autre : conventions de modélisation, vos préférences, corrections et faits concernant les projets. Il les connaît dans chaque nouvelle conversation. Un point sur le même sujet est mis à jour au lieu d'être dupliqué, et les points obsolètes sont supprimés.

L'assistant enregistre lui-même les points importants et le signale en une phrase dans sa réponse. Vous pouvez aussi le lui demander explicitement en commençant par « retiens... » :

Retiens : nous numérotons les exigences FR01, FR02... et les gardons dans le paquetage « Exigences ».

Retiens que, sur les diagrammes BPMN, nous colorons les tâches système en vert.

Les points retenus sont communs à tous les modèles. Si vous voulez qu'un point ne s'applique qu'à un seul modèle, précisez-le, par exemple « retiens cela uniquement pour ce modèle » : le point n'est alors utilisé que dans les conversations avec ce modèle, et la vue « Mémoire de l'assistant » l'indique par la mention « Uniquement pour le modèle ... ».

L'enregistrement d'un point demande toujours votre validation, dans tous les modes, et le panneau affiche son texte complet (chapitre 16). L'enregistrement n'interrompt pas le travail de l'assistant sur la tâche en cours. Le contenu complet de la mémoire est envoyé au modèle d'IA avec chaque requête (chapitre 18).

La vue « Mémoire de l'assistant »

- La liste des points avec leur date.
- **Supprimer** efface le point sélectionné.
- **Ouvrir le dossier de la mémoire** ouvre le dossier `memory` dans l'Explorateur. Les points y sont des fichiers Markdown, et l'index se trouve dans le fichier `MEMORY.md`.

16. Validation des modifications et journal des modifications

Vous définissez le mode dans la vue **Paramètres**, carte « Langue et validation des modifications ». Il détermine si et quand l'application demande votre accord avant un enregistrement dans EA (l'enregistrement d'un point dans la mémoire de l'assistant et celui d'un modèle de document demandent toujours votre accord). La lecture du modèle et l'affichage des diagrammes ne nécessitent jamais d'accord. Le mode actuel est aussi visible dans la carte de message.

Mode	Quand il demande l'accord	Pour qui
ask	Avant chaque enregistrement	Travail sur un modèle important ; contrôle total
plan	Une fois, avant le premier enregistrement d'une réponse de l'assistant. Les enregistrements suivants de cette réponse se font sans nouvelle demande, sauf la suppression et le déplacement d'objets ainsi que la copie de paquetages, qui demandent toujours une nouvelle validation.	Tâches plus importantes, lorsque le premier enregistrement montre la direction
auto	Seulement pour les actions qui demandent toujours un accord (ci-dessous)	Modèle de test ou travail facile à annuler

Lorsqu'un accord est nécessaire, le panneau « **L'assistant veut enregistrer des modifications dans le modèle** » apparaît au-dessus du champ de message, avec une description des modifications (combien de paquetages, diagrammes, éléments, connecteurs, ce qui est supprimé) et les boutons **Valider** et **Refuser**. Pour une suppression, un déplacement ou une copie de paquetage, le panneau donne le nom de chaque objet concerné (lignes SUPPRESSION, DÉPLACEMENT et COPIE). Un enregistrement refusé ne modifie pas le modèle ; l'assistant est informé du refus et peut proposer autre chose.

rejetés avant l'envoi à EA (par exemple en cas de conflit de version), n'y figurent pas. La vue **Journal des modifications du modèle** affiche les 200 dernières entrées, de la plus récente à la plus ancienne : résultat, heure, compte EA, base et description des modifications. Le bouton **Exporter vers Excel** enregistre l'ensemble du journal sous forme de classeur XLSX, par exemple pour une revue des modifications en équipe ; **Ouvrir le fichier journal** ouvre le fichier dans le Bloc-notes.

17. Conversations : historique, recherche, travail en arrière-plan

- **Nouvelle conversation** : le bouton + au-dessus de la liste des conversations. Chaque conversation est enregistrée et réapparaît après le redémarrage de l'application.
- **Groupes** : les conversations sont regroupées en « Aujourd'hui », « Hier », « 7 derniers jours » et « Plus ancien ».
- **Recherche** : le champ « Rechercher dans les conversations... » filtre la liste par titre et par contenu des messages.
- **Renommer** : cliquez sur le crayon à côté de la conversation sélectionnée, saisissez un nom ; **Entrée** enregistre, **Échap** annule.
- **Supprimer** : cliquez avec le bouton droit sur une conversation de la liste et choisissez **Supprimer la conversation**. Son fichier est déplacé dans le dossier `history\bin` (chapitre 18) : vous pouvez la récupérer en remettant le fichier dans `history` pendant que l'application est fermée. Une conversation dans laquelle l'assistant travaille ne peut pas être supprimée avant la fin de son travail.
- **Longues conversations** : avant qu'une requête n'approche 100 000 jetons (Claude Haiku 5.5 facture une requête plus longue 5× en totalité), la partie ancienne de la conversation est remplacée pour l'assistant par un résumé écrit par le modèle, et les 8 messages les plus récents restent mot pour mot. Le résumé est enregistré avec la conversation et utilisé dans les réponses suivantes ; dans le chat, la ligne « L'assistant voit la conversation ci-dessus sous forme de résumé » marque où commence le texte complet. L'historique affiché reste complet. Lors d'un très long travail avec les outils (par exemple des dizaines de fichiers lus), le serveur d'Anthropic retire lui-même les plus anciens résultats d'outils de la vue du modèle ; les 6 plus récents restent et l'assistant relit un fichier si nécessaire.
- **Plusieurs conversations à la fois** : pendant que l'assistant travaille, vous pouvez commencer une nouvelle conversation ou passer à une autre. Le travail commencé se poursuit en arrière-plan.
- **Modèle de la conversation** : chaque conversation travaille avec son propre modèle EA, et une nouvelle conversation commence par la confirmation de ce modèle. L'écran d'une nouvelle conversation affiche le modèle par défaut défini dans les paramètres. Le bouton **Choisir un autre modèle...** ouvre la fenêtre « Modèle de cette conversation », où vous choisissez **Par défaut, depuis les paramètres** ou **Autre modèle** : **Base PostgreSQL** (serveur, port, nom de la base, compte de la base, mot de passe, chiffrement de la connexion) ou **Fichier EA (.qea, .qeax)**, avec le compte EA et les options de sécurité comme dans la carte Modèle. **Tester la connexion** ouvre le modèle, compte les paquets, éléments et diagrammes, puis enregistre un paquetage de test qu'il vérifie et supprime aussitôt. Le bouton **Confirmer le modèle et commencer** fixe le modèle de la conversation : il ne peut plus être modifié ensuite, et le bouton **Modèle : ...** de l'en-tête l'affiche en lecture seule. Pour travailler avec un autre modèle, commencez une nouvelle conversation. Changer le modèle par défaut dans les paramètres ne modifie pas les conversations en cours. Les mots de passe d'une connexion différente de celle des paramètres sont conservés dans le Gestionnaire d'identification Windows. Dans Compass dans EA, le modèle de la conversation est toujours le projet ouvert dans EA (chapitre 10).
- **Fichier EA déposé sur la fenêtre** : un fichier .qea ou .qeax déposé sur le chat devient le modèle de la conversation au lieu d'une pièce jointe ; dans une conversation qui contient déjà des messages, une

nouvelle conversation commence avec ce fichier.

- **Conversations des versions précédentes** : au premier démarrage de la version 0.5 ou plus récente, elles reçoivent une copie du modèle défini dans les paramètres ; elles continuent donc à travailler avec ce modèle même si les paramètres changent par la suite. Le survol d'une conversation dans la liste affiche son modèle.
- **Continuer avec un autre modèle** : dans la fenêtre séparée, vous changez le modèle d'une conversation avec le bouton « Modèle : ... » de l'en-tête ; à partir de ce point, le chat affiche la ligne « À partir d'ici : modèle ... », et l'assistant sait que les messages précédents concernaient un autre modèle. Dans l'onglet EA, utilisez « Conversations d'autres modèles... » (chapitre 10).
- **Export et import** : clic droit sur une conversation → **Exporter la conversation...** l'enregistre avec ses pièces jointes dans un fichier `.lcchat` (sans comptes ni mots de passe : seulement le chemin du fichier ou le nom de la base et du serveur) ; l'icône d'import à côté de + lit un tel fichier comme nouvelle conversation avec le modèle ouvert dans cette fenêtre.
- **Coût d'une réponse** : la dernière étape de chaque réponse, « 💰 Coût de cette réponse », indique les appels au modèle, les jetons d'entrée (avec la part lue depuis le cache), les jetons de sortie et le coût en USD aux tarifs du modèle. Chaque appel va aussi dans le journal `usage\<jour>.json1` du dossier de données (des chiffres seulement, aucun contenu).

Point à côté d'une conversation	Signification
● orange	L'assistant travaille dans cette conversation (ou attend votre accord pour enregistrer)
● vert	L'assistant a terminé son travail en arrière-plan, et vous n'avez pas encore consulté le résultat

Les enregistrements dans EA provenant de différentes conversations s'exécutent les uns après les autres, pas simultanément. EA en arrière-plan ne garde qu'un seul modèle ouvert à la fois : passer d'une conversation à une autre qui utilise un modèle différent le rouvre, ce qui prend de quelques secondes à environ 25 secondes. Lorsque le chat a défilé vers le haut, le cercle avec une flèche au-dessus du champ de message vous ramène à la fin de la conversation.

18. Données, sécurité et confidentialité

Où se trouvent vos données

Données	Emplacement
Programme	Le dossier d'installation : %LOCALAPPDATA%\Programs\Loonar Compass (pour vous uniquement) ou %ProgramFiles%\Loonar Compass (pour tous les utilisateurs)
Paramètres (app.properties , sans aucun secret), historique des conversations (history ; conversations supprimées dans history\bin), pièces jointes et documents créés : Word, Excel, PowerPoint, PDF, PNG, pages Confluence et leurs dossiers d'images <page>_images (attachments), mémoire de l'assistant (memory , index MEMORY.md), modèles de documents (templates), journal des modifications (audit_log.json1), pages Confluence publiées (confluence_pages.json)	Le dossier des données utilisateur : %APPDATA%\Loonar Compass . Chaque utilisateur Windows de l'ordinateur a le sien.
Clé API Claude, clé API OpenAI, mot de passe du compte EA, mot de passe de la base, jetons Confluence et Miro, secret client SharePoint, secret OAuth2 du service d'IA (ai_oauth_client_secret), jetons et secrets des serveurs MCP (mcp_...)	Gestionnaire d'identification Windows, entrées « loonar-compass » (protégées par le système)
Instructions de modélisation de l'équipe	Dans le référentiel EA lui-même (un Artifact «CompassInstructions», chapitre 7), et non sur votre ordinateur

L'historique des conversations est enregistré sous forme de fichiers ordinaires dans votre profil Windows : il est protégé par votre compte Windows, mais l'application ne le chiffre pas.

Ce qui est envoyé au fournisseur du modèle d'IA

Pour que l'assistant puisse répondre, chaque requête envoie au fournisseur de modèle d'IA choisi (Anthropic ou OpenAI, carte Modèle d'IA), avec votre clé API : la conversation, les fichiers joints, le contenu

complet de la mémoire de l'assistant, les instructions de modélisation du référentiel, la liste des modèles de documents avec leurs descriptions, les résultats des requêtes de l'assistant sur le modèle (noms et descriptions des paquetages, éléments, diagrammes) et les images des diagrammes. La base entière n'est pas envoyée, seulement ce que l'assistant a demandé. La chaîne de connexion de la fenêtre EA ouverte n'est jamais transmise au modèle. Après un changement de langue, le contenu de la mémoire et des modèles à traduire est également envoyé. Les données vont directement au fournisseur choisi (api.anthropic.com ou api.openai.com) ; l'autre fournisseur ne reçoit rien. Loonar ne reçoit pas ces données. Avec `claude.url` (chapitre 5), les requêtes vont à cette adresse, par exemple votre ressource Azure AI Foundry ou une passerelle d'entreprise.

Autres destinations possibles de vos données, uniquement si vous les configurez ou les activez :

Où	Quoi
Bac à sable d'Anthropic (commutateur <code>Σ</code> , chapitre 6)	Le code que l'assistant exécute et les données qu'il lui transmet, pour des calculs et l'analyse de données. Ils vont à Anthropic, aux mêmes conditions que la conversation. Désactivé par défaut.
SharePoint (Microsoft Graph, chapitre 13)	Les noms des dossiers et documents que l'assistant liste, les documents qu'il lit et les fichiers dont vous validez l'envoi. Connexion sur <code>login.microsoftonline.com</code> , données sur <code>graph.microsoft.com</code> .
Adresses de jetons OAuth2 (<code>ai.oauth.token_url</code> , <code>"auth"</code> des serveurs MCP)	Uniquement l'ID client, le secret client et la portée (scope), pour obtenir un jeton ; aucun contenu du modèle ni de la conversation.
Serveurs MCP, Jira, Confluence, le hub d'équipe	Comme avant : ce que l'assistant envoie à leurs outils, tickets, pages et au journal d'audit (chapitres 7, 13 et 22).

Avant d'utiliser l'application sur un modèle d'entreprise, vérifiez si les règles de votre organisation et vos contrats avec vos clients autorisent la transmission du contenu du modèle à un service d'IA externe, et quelles conditions de traitement des données s'appliquent à votre compte chez le fournisseur choisi (Anthropic ou OpenAI).

Droits de l'assistant

- L'assistant lit la base dans une session en lecture seule, et un fichier de modèle en lecture seule, avec une limite de durée des requêtes ; les fonctions du serveur qui lisent des fichiers ou agissent sur d'autres sessions sont bloquées. Toutes les modifications passent par Enterprise Architect, avec le compte EA du modèle (de la carte Modèle ou du modèle de la conversation) ; les droits et verrous de ce compte s'y appliquent donc. Compass dans EA lit et enregistre le modèle par la fenêtre EA dans laquelle il est ouvert, avec le compte que vous avez utilisé pour vous connecter dans EA.
- Dans Confluence, l'assistant agit avec les droits du jeton de la carte « Confluence » et ne publie qu'après votre validation.
- L'assistant n'a accès à aucun fichier sur le disque en dehors de ceux que vous lui joignez vous-même et de ses propres dossiers de mémoire et de modèles ; il ne lit tout autre fichier (un document d'exigences, un dossier de code source) qu'après vous l'avoir demandé. Les images qu'il dessine ne peuvent lire aucun fichier sur le disque ni sur le réseau.
- **Protection contre les instructions cachées** (prompt injection). Un document, un ticket, une page ou une note du modèle peut contenir un texte rédigé pour piloter l'assistant. L'assistant reçoit des règles

de sécurité dans chaque requête ; les résultats des documents, de Jira, de Confluence, des serveurs MCP, de la CMDB et du code sont marqués comme des données, et non comme des instructions ; une réponse qui reprend mot pour mot les instructions internes de l'assistant est retenue (« La réponse reprenait les instructions internes de l'assistant et a été retenue. »). Les enregistrements passent toujours par votre validation (chapitre 16) : vérifiez chaque modification avant de la valider.

Statistiques d'utilisation

Sauf si vous les désactivez, Loonar Compass envoie des statistiques anonymes : à chaque démarrage un identifiant d'installation aléatoire (créé sur cet ordinateur, non dérivé de l'ordinateur ni de vous) avec la version de l'application, la version de Windows et la langue, et toutes les heures un résumé de la journée : les types de tâches et leur issue, combien de fois chaque fonction de l'assistant a été utilisée et a échoué, les évaluations des réponses, ce que l'assistant n'a pas su faire, les validations, les vues ouvertes, les jetons et le type de configuration. Ne sont **jamais** envoyés le contenu des conversations, les noms ou identifiants du modèle, les chemins de fichiers, les adresses de serveurs, les messages d'erreur ni les noms d'utilisateurs. Le service qui reçoit les données n'enregistre pas les adresses IP ; seul l'auteur voit les données, et seuls des résumés de l'ensemble des installations sont publiés (lien « Statistiques publiées »). Les données sont conservées 13 mois.

Au premier démarrage, l'application affiche un avis avec les boutons **Laisser activé**, **Désactiver** et **Afficher les détails**. La vue « **Statistiques d'utilisation** » (barre de navigation) montre exactement ce qui a été compté aujourd'hui (« Aujourd'hui jusqu'ici ») et ce qui a été envoyé ces 30 derniers jours, en JSON avec un bouton **Copier**. Vous pouvez y décocher « Envoyer des statistiques d'utilisation anonymes (toutes les heures) » (les jours non encore envoyés sont alors supprimés) ou donner à l'installation un **Nouvel identifiant**, après quoi les jours suivants ne peuvent plus être reliés aux précédents.

Une organisation peut désactiver les statistiques pour tous : avec la variable d'environnement `LOONAR_TELEMETRY=0`, une stratégie de registre (modèle ADMX du paquet pour administrateurs) ou la stratégie du hub d'équipe (`"telemetry": false`). Rien n'est alors compté ni envoyé.

Connexions et chiffrement

- **API du modèle d'IA (Anthropic ou OpenAI, ou l'adresse de `claude.url`), Confluence, Jira, SharePoint, Miro et adresses de jetons OAuth2** : uniquement via HTTPS (`http://` n'est accepté que pour un serveur installé sur le même ordinateur).
- **Base du modèle** : la connexion peut être chiffrée avec TLS, selon le réglage de la carte Modèle (chapitre 5). Les administrateurs peuvent imposer TLS à tous les clients avec des entrées `hostssl` dans `pg_hba.conf` . La connexion d'Enterprise Architect lui-même à la base suit les paramètres d'EA.
- **Proxy** : l'application utilise le proxy défini dans les paramètres Internet de Windows ainsi que les variables d'environnement `HTTPS_PROXY` et `ALL_PROXY` . Les scripts PAC et l'authentification NTLM/Kerberos sur le proxy ne sont pas pris en charge.
- **Erreurs temporaires de l'API du modèle d'IA** (surcharge 529 et autres erreurs 5xx, limite de requêtes 429) : l'application réessaie automatiquement.

19. Résolution des problèmes

Symptôme	Que faire
L'application s'ouvre sur la vue « Paramètres »	Des paramètres ou des secrets manquent. Remplissez les cartes et enregistrez chacune avec son bouton « Enregistrer et tester... ».
« Le fichier de paramètres ... contient une erreur »	Une valeur du fichier <code>app.properties</code> est incorrecte ; le message indique laquelle. Corrigez la valeur dans la carte concernée et enregistrez de nouveau la carte.
Test de la clé : « L'API Claude a refusé la clé API (401) » ou « L'API OpenAI a refusé la clé (401) »	La clé est erronée ou révoquée. Collez la clé actuelle du fournisseur choisi (console Anthropic ou plateforme OpenAI) dans la carte « Modèle d'IA ».
« L'API OpenAI ne connaît pas le modèle ... (404) »	L'identifiant du modèle dans la carte « Modèle d'IA » est erroné, ou votre compte OpenAI n'a pas accès à ce modèle. Choisissez un modèle dans la liste ▼.
« ... une réponse qui ne provient pas de l'API (un proxy ou un pare-feu bloque peut-être api.anthropic.com) »	Un proxy ou un pare-feu bloque la connexion. Demandez à votre service informatique d'autoriser api.anthropic.com (ou api.openai.com lorsque le fournisseur est OpenAI) en HTTPS. Les scripts PAC et l'authentification NTLM/Kerberos sur le proxy ne sont pas pris en charge (chapitre 18).
« L'API Claude est surchargée ou temporairement indisponible » ou « La limite de requêtes de l'API Claude est atteinte (429) »	L'application a déjà réessayé automatiquement. Patientez un instant et renvoyez votre message.
Test du fichier : « cela ne ressemble pas à un fichier de modèle Enterprise Architect »	Choisissez un fichier de projet .qea ou .qeax créé par Enterprise Architect 16 ou une version ultérieure. Les anciens fichiers .eap/.eapx doivent d'abord être convertis en .qea dans EA (transfert de projet).
Test de la base : pas de connexion	Vérifiez le serveur, le port, le nom de la base, le compte et le mot de passe, et assurez-vous que le serveur PostgreSQL accepte les connexions depuis cet ordinateur.
« La connexion n'est PAS chiffrée : le serveur ne propose pas TLS » ou échec de la vérification du certificat	Demandez à l'administrateur de la base d'activer TLS. En mode « Obligatoire, certificat vérifié », le certificat du serveur doit être approuvé dans le magasin de certificats Windows et correspondre au nom saisi dans le champ Serveur ; sinon, choisissez « Obligatoire » ou « Préféré ».
Le test de la base avertit que le compte est un superutilisateur	Les droits de superutilisateur ne sont pas nécessaires. Demandez à l'administrateur un compte ordinaire disposant des droits de lecture et d'écriture sur les tables du modèle : l'assistant ne fait que lire la base, mais Enterprise Architect enregistre les modifications avec ce même compte.

« EA n'a pas répondu dans les 300 s »	Le plus souvent, un compte ou un mot de passe d'utilisateur du modèle erroné (EA attend alors sur une fenêtre de connexion invisible) ou aucune licence disponible. Corrigez les données dans la carte Modèle (compte EA, données de la base) et utilisez « Enregistrer et tester l'accès à EA » dans la carte Enterprise Architect.
Le premier enregistrement ou le premier diagramme prend du temps	C'est le démarrage d'Enterprise Architect en arrière-plan (de quelques secondes à plus de dix secondes). Les appels suivants sont rapides.
« La réponse du modèle a été tronquée à la limite de ... jetons »	Augmentez « Jetons max. » dans la carte « Modèle d'IA » ou demandez que la tâche soit effectuée en plusieurs parties plus petites.
« L'assistant a effectué 24 étapes sans terminer »	L'assistant s'est arrêté pour éviter une boucle sans fin. Poursuivez dans un nouveau message ou divisez la tâche en parties plus petites.
Les modifications ne sont pas visibles dans Enterprise Architect	Si le même modèle est ouvert dans EA, l'assistant l'actualise lui-même. Sinon, actualisez le paquetage dans le navigateur de projet (F5) ou rouvrez le diagramme.
« Impossible de verrouiller l'objet »	L'objet est verrouillé par un autre utilisateur du modèle ; le message indique lequel. Levez le verrou dans EA ou demandez à son propriétaire de le faire.
« ! » orange sur les icônes Paramètres, Mémoire de l'assistant, Modèles de documents	Après un changement de langue, la traduction attend une connexion au modèle d'IA. Vérifiez la clé API dans la carte « Modèle d'IA » et cliquez sur « Traduire maintenant » dans la carte « Langue et validation des modifications ».
La publication dans Confluence échoue	Vérifiez la carte « Confluence » avec le bouton « Enregistrer et tester la connexion à Confluence » : adresse (https://), jeton, clé de l'espace et, pour Cloud, également l'e-mail. Vous pouvez aussi coller la page manuellement avec le bouton « Copier le code de la page ».
Après une baseline, il faut revenir à une version antérieure du paquetage	Dans EA : clic droit sur le paquetage → Manage Baselines → Compare/Restore.
L'assistant affirme ne pas savoir faire quelque chose	Demandez : « Que sais-tu faire dans cette application ? ». Si la fonctionnalité figure dans la liste de la vue « Nouveautés », dites-le-lui explicitement et demandez-lui de réessayer.
Le bouton Loonar Compass manque dans le ruban Start d'EA ou est inactif	Relancez le programme d'installation en cochant « Ajouter Loonar Compass à Enterprise Architect », puis redémarrez EA. Le bouton n'est actif que lorsqu'un projet est ouvert (chapitre 10).
Impossible de faire glisser des fichiers dans la fenêtre	L'application a été lancée en tant qu'administrateur. Fermez-la et lancez-la avec le raccourci habituel.

« aucun pilote ODBC pour sqlserver (ou mysql, oracle) n'est installé sur cet ordinateur »	Installez le pilote ODBC 64 bits de l'éditeur de la base (pour SQL Server : « ODBC Driver 18 for SQL Server ») et testez à nouveau la carte Modèle. Le message indique les pilotes installés. Vous pouvez aussi indiquer la chaîne de connexion complète dans <code>db.odbc</code> (chapitre 5).
« Word n'est pas installé sur cet ordinateur » ou « PowerPoint n'est pas installé sur cet ordinateur » lors de la création d'un PDF	Le PDF est créé par Microsoft Office. Le fichier DOCX ou PPTX est bien là : ouvrez-le et enregistrez-le vous-même en PDF, ou installez Office.
Connexion d'entreprise au hub : « La connexion a échoué »	« ... n'appartient à aucun groupe du hub » : demandez à l'administrateur de vous ajouter à l'un des groupes du hub. « Un utilisateur local du hub porte le même nom » : l'administrateur renomme ou supprime cet utilisateur local. « La connexion a expiré ou n'a pas commencé ici » : recommencez depuis la page de connexion. « La connexion avec le compte d'entreprise n'est pas disponible pour l'instant » : le hub ne joint pas le fournisseur ; l'administrateur vérifie <code>issuer</code> et <code>hub.log</code> (chapitre 22).
SharePoint : « Microsoft Graph a répondu 403 : ... »	L'inscription d'application n'a pas d'autorisation pour ce site : l'administrateur accorde <code>Sites.Selected</code> pour le site (ou <code>Sites.ReadWrite.All</code>) comme autorisation d'application, avec le consentement de l'administrateur. 401 : vérifiez le locataire, l'ID client et le secret client (il a peut-être expiré).
Windows SmartScreen avertit au lancement du programme d'installation	Le programme d'installation n'est pas encore signé. Cliquez sur « Informations complémentaires », puis sur « Exécuter quand même ». Si le contrôle intelligent des applications (Smart App Control) est activé, les applications non signées sont bloquées : attendez la version signée.
<code>LoonarCompass.exe --check</code> n'affiche rien dans PowerShell, ou <code>\$LASTEXITCODE</code> est vide	Compass est un programme à fenêtre : PowerShell ne l'attend donc pas. Redirigez sa sortie : <code>... Out-Host</code> (chapitre 23).
Un serveur MCP prêt à l'emploi lancé avec <code>npx</code> ne démarre pas	Installez Node.js (il fournit npx) et redémarrez Compass. Vérifiez l'organisation ou le dossier dans l'entrée du serveur.
Impossible de se connecter à la vue web du hub	Vérifiez le nom d'utilisateur et le mot de passe. Après 20 tentatives erronées depuis une même adresse, le hub la refuse pendant 10 minutes. Un administrateur peut définir un nouveau mot de passe dans l'onglet Users (chapitre 22).
Message d'erreur inattendue, l'application se ferme	Les détails sont enregistrés dans le fichier <code>%TEMP%\loonar-compass-crash.log</code> . Signalez le problème (chapitre 20) en joignant ce fichier.
L'installation échoue car un fichier est utilisé par un autre processus	Compass (ou EA avec l'onglet Compass) est en cours d'exécution. Sur la page d'erreur, cliquez sur Fermer Compass

et réessayer, ou fermez-le vous-même et cliquez sur **Réessayer** – sans recommencer l'installation.

20. Signalement de bugs et nouvelles versions

Signaler un bug

1. Cliquez sur l'icône **Signaler un bug** dans la barre de navigation ou, pour une réponse précise, sur  → « **Signaler un problème avec cette réponse** » (chapitre 6).
2. Choisissez comment envoyer le signalement : « **Ticket public sur GitHub (recommandé)** » ou « **E-mail à l'auteur (privé)** ».
3. Saisissez l'objet et la description : ce que vous faisiez, ce qui s'est passé, ce que vous attendiez. La description est obligatoire ; vous pouvez y coller le texte de l'erreur depuis le chat. Pour un ticket GitHub, l'objet et la description sont publics : n'y mettez pas d'informations confidentielles.
4. Ajoutez des images, idéalement des captures d'écran avec le problème marqué : glissez-les dans la fenêtre, collez-les depuis le presse-papiers (**Ctrl** + **V**) ou choisissez-les sur le disque avec + **Ajouter une image**.
5. Options : « **Joindre les informations techniques (public)** » ajoute la version, Windows, le modèle d'IA, le mode de validation, le nombre d'étapes et l'identifiant d'installation, sans mots de passe ni clés. Pour un signalement depuis une réponse, « **Joindre la réponse signalée** » ajoute votre question, la réponse et les étapes de l'assistant ; seul l'auteur les voit. « **Afficher ce qui est joint** » montre exactement ce qui sera envoyé. Une adresse e-mail pour la réponse est facultative.
6. Cliquez sur **Envoyer le signalement**. Le ticket GitHub est créé tout de suite, et l'application affiche son lien, où vous suivez ce qu'il devient. Pour l'e-mail, un message prêt s'ouvre dans votre messagerie avec les images en pièces jointes ; vérifiez-le et envoyez-le.

Dans un ticket GitHub (dépôt github.com/loonar-eu/loonar-compass), le titre, la description et, si vous laissez l'option activée, les informations techniques sont publics. Les captures d'écran, la réponse signalée et votre adresse e-mail vont dans un dépôt privé que seul l'auteur lit. Un signalement envoyé par e-mail n'est pas publié. Les signalements servent uniquement à répondre et à améliorer l'application et sont conservés jusqu'à 2 ans après la clôture du dossier. Avant l'envoi, retirez des captures les données personnelles et les informations confidentielles. Les détails figurent dans la politique de confidentialité, liée dans le formulaire ; le lien « **Tickets et idées** » mène à la liste de tous les signalements.

Nouvelles versions (newsletter)

Dans la carte « **Nouvelles versions (newsletter)** » des paramètres, saisissez votre adresse e-mail et cochez la case « je souhaite recevoir des informations sur les nouvelles versions » : un court e-mail à la sortie d'une nouvelle version, avec une description des modifications et l'endroit où la télécharger, sans publicité. Votre choix est mémorisé dans les paramètres, et l'inscription comme la désinscription sont envoyées à l'auteur par un message du logiciel de messagerie.

Votre adresse n'est utilisée que pour les informations sur les nouvelles versions, sur la base de votre consentement, jusqu'à son retrait : décochez la case et enregistrez, ou écrivez un e-mail. Responsable du traitement : Loonar Mariusz Głowacki, mariusz@loonar.eu. La carte contient un lien vers la politique de confidentialité.

21. Mise à jour et désinstallation

Mise à jour. Fermez l'application et lancez le programme d'installation de la nouvelle version. L'assistant d'installation affiche la page « Nouveautés de la version ... » avec les modifications depuis la version installée, conserve le type et le dossier de l'installation existante, ainsi que vos paramètres, l'historique des conversations, les pièces jointes, la mémoire, les modèles et le journal des modifications. Pour une mise à jour depuis l'ancienne version, voir le chapitre 3.

Avis de nouvelle version. Toutes les heures, Compass vérifie sur GitHub s'il existe une version plus récente et affiche alors une barre en haut de la fenêtre : **Nouveautés et téléchargement** ouvre la page de la version, **Pas maintenant** masque la barre jusqu'à la version suivante. Dans EA, l'avis apparaît une fois dans l'onglet « Loonar Compass » de la fenêtre System Output. Compass ne télécharge ni n'installe jamais rien tout seul, et la requête ne contient aucun identifiant de l'installation. Désactivez la vérification dans Paramètres → Nouvelles versions (newsletter) ou avec `app.update_check=false` ; l'administrateur du hub peut la désactiver pour tous (`"update_check": false` dans la politique d'équipe).

Fermeture de l'application. La fermeture de la fenêtre ferme aussi la session Enterprise Architect lancée en arrière-plan par l'application, de sorte que la licence EA est libérée immédiatement.

Désinstallation. Fermez l'application. Utilisez le raccourci « Désinstaller Loonar Compass » du menu Démarrer ou l'entrée « Loonar Compass » dans les paramètres de Windows (Applications → Applications installées). Pour une installation destinée à tous les utilisateurs, Windows demande des droits d'administrateur. Le programme de désinstallation supprime les fichiers du programme, les raccourcis et l'entrée système, en affichant la progression sur une barre. Vos données sont conservées, sauf si vous cochez :

- « Supprimer les paramètres, les conversations, les pièces jointes, la mémoire de l'assistant, les modèles de documents et le journal des modifications » : le dossier `%APPDATA%\Loonar Compass` est alors supprimé en entier,
- « Supprimer la clé API, les mots de passe et les jetons du Gestionnaire d'identification » : tous les secrets, y compris le jeton Confluence et le mot de passe de la base.

Pour une installation destinée à tous les utilisateurs, seules vos propres données sont supprimées ; celles des autres utilisateurs restent dans leur profil.

22. Compass Hub (service d'équipe)

Compass Hub est un service d'équipe facultatif, exécuté sur un ordinateur ou un serveur de l'équipe. Il fournit une politique commune (règles du contrôle du modèle et mode d'approbation minimal), un journal d'audit central des modifications de tous les postes, des notes partagées, des ensembles de modifications à relire, une vue web des modèles pour les personnes sans Enterprise Architect, des tickets Jira et des pages Confluence créés depuis la vue web, des rapports Word et Excel des paquetages, la connexion avec le compte d'entreprise (OpenID Connect), des agents d'événements pour toute l'équipe, une API GraphQL en lecture seule et des tables pour Power BI, et un serveur MCP (lecture seule) pour d'autres outils d'IA. Sans hub, Compass fonctionne seul, comme avant.

Pour l'utilisateur

Vous recevez l'adresse du hub et votre jeton API de l'administrateur (onglet **Users** de la vue web). Saisissez-les dans la carte « **Équipe (Compass Hub)** » de la vue Paramètres : **Adresse du hub** (elle doit commencer par `https://` ; `http://` seulement sur cet ordinateur), **Jeton** et, facultativement, **Ce modèle dans le hub** (l'id de la source, c'est-à-dire le modèle du hub qui correspond au vôtre), puis cliquez sur **Enregistrer et tester la connexion au hub**. Le test affiche votre compte, votre rôle et la politique de l'équipe. Avec un hub :

- la politique d'équipe `min_approval` ne peut que durcir le mode de validation local (du plus souple : `auto`, `plan`, `ask`) ;
- les règles du contrôle du modèle du hub s'appliquent s'il n'y a pas de `governance.json` local (chapitre 7) ;
- chaque modification enregistrée va dans le journal d'audit de l'équipe ; l'envoi se fait en arrière-plan, donc un hub hors service n'arrête pas votre travail, et le dernier problème est affiché sur la carte.

Installer le hub

Le hub est un programme à part entière, `compass-hub`, pour Windows et Linux. Il lit les modèles directement (une base PostgreSQL, SQL Server, MySQL/MariaDB ou Oracle, ou des fichiers `.qea/.qeax`) : le serveur n'a donc pas besoin d'Enterprise Architect. Téléchargements :

- **Windows** : `CompassHub-0.9.0-windows-x64.zip` avec `compass-hub.exe`, un exemple de `hub.json` et un README.
- **Docker** : l'image `ghcr.io/loonar-eu/compass-hub:0.9.0`, ou construisez-la vous-même dans le dossier racine du dépôt : `docker build -f hub/Dockerfile -t compass-hub .`

Dans une console (Windows ou Linux). Le hub tourne jusqu'à la fermeture de la console :

```
compass-hub hub.json
```

Comme service Windows, dans une console lancée en tant qu'administrateur. Le service « CompassHub » démarre aussitôt et avec Windows ; vous l'arrêtez dans Services ou avec `sc stop CompassHub`, et le supprimez avec `--service-uninstall` :

```
compass-hub --service-install C:\Hub\hub.json
compass-hub --service-uninstall
```

Dans Docker. Placez `hub.json` avec `"listen": "0.0.0.0:8730"` dans le dossier de données, que le conteneur voit comme `/data` :

```
docker run -d --name compass-hub -p 8730:8730 -v /srv/compass-hub:/data \
-e LOONAR_SECRET_CLAUDE_API_KEY=... ghcr.io/loonar-eu/compass-hub:0.9.0
```

- Le conteneur s'exécute sous l'utilisateur d'id 10001 : le dossier de données doit être accessible en écriture pour lui, car le hub y écrit `hub.json`, le journal d'audit, les notes, les ensembles de modifications et `hub.log`.
- Modèles : des fichiers `.qea/.qeax` dans le dossier de données, avec les chemins vus depuis le conteneur (par exemple `model.file=/data/models/shop.qea` dans l'`app.properties` de la source), ou une base de données. L'image contient unixODBC pour SQL Server, MySQL et Oracle ; le pilote ODBC Linux de l'éditeur (par exemple le paquet `msodbcsql18` de Microsoft) doit être ajouté dans une image construite sur celle-ci. PostgreSQL et les fichiers EA n'ont besoin de rien d'autre.

`LoonarCompass.exe --hub C:\Hub\hub.json` fonctionne toujours : l'application de bureau exécute alors le hub sans fenêtre. Le hub écrit son journal dans `hub.log`, à côté de `hub.json`. `compass-hub --version` affiche la version.

Secrets du hub

Les clés, jetons et mots de passe ne figurent pas dans `hub.json`. `compass-hub` lit chacun d'eux dans la variable d'environnement `LOONAR_SECRET_<NOM>` ou dans le fichier indiqué par `LOONAR_SECRET_<NOM>_FILE` (par exemple un secret Docker) ; sous Windows aussi dans le Gestionnaire d'identification du compte sous lequel le hub s'exécute. Pour le service Windows, définissez-les comme variables d'environnement système.

Secret	Usage
<code>LOONAR_SECRET_CLAUDE_API_KEY</code> ou <code>LOONAR_SECRET_OPENAI_API_KEY</code>	Le modèle d'IA de l'onglet Ask
<code>LOONAR_SECRET_JIRA_TOKEN</code>	Le Jira de l'équipe (<code>integrations</code>)
<code>LOONAR_SECRET_CONFLUENCE_TOKEN</code>	Le Confluence de l'équipe (<code>integrations</code>)
<code>LOONAR_SECRET_DB_PASSWORD</code>	Le mot de passe d'une base de modèle (PostgreSQL, SQL Server, MySQL ou Oracle)
<code>LOONAR_SECRET_OIDC_CLIENT_SECRET</code>	Le secret client du hub auprès du fournisseur de la connexion d'entreprise (<code>oidc</code>)

Comptes et connexion

Les personnes se connectent à la vue web (`https://<hub>/` ; `http://` seulement sur le même ordinateur) avec un nom d'utilisateur et un mot de passe. Compass et les clients MCP utilisent un jeton API.

Compte par défaut. Un nouveau hub (un `hub.json` sans utilisateurs) crée le compte `admin` avec le mot de passe `password` (avec un zéro). À la première connexion, le hub demande de changer le mot de passe et n'autorise rien d'autre d'ici là. Faites-le avant que le hub soit accessible depuis d'autres ordinateurs.

- Un mot de passe compte au moins 10 caractères et ne peut être ni le mot de passe par défaut ni le nom d'utilisateur. Le hub ne conserve les mots de passe que sous forme d'empreintes PBKDF2.
- Une connexion dure 12 heures ; **Sign out** y met fin immédiatement.
- Après 20 mots de passe ou jetons erronés depuis une même adresse, cette adresse est refusée pendant 10 minutes.
- **Sign in with a token instead**, sur la page de connexion, permet de se connecter avec un jeton API.

L'onglet **Users** (administrateurs seulement) :

- **Add a user** (ajouter un utilisateur) : le nom, le rôle et un premier mot de passe, que l'utilisateur doit changer à la première connexion.
- **Role** : change le rôle (viewer, architect ou admin).
- **Reset password** : définit un nouveau mot de passe, que l'utilisateur doit changer à la connexion suivante.
- **New API token** : un jeton pour Compass (Paramètres → Équipe (Compass Hub)) et pour les clients MCP. Il n'est affiché qu'une fois, et le jeton précédent de l'utilisateur cesse de fonctionner.
- **Remove** : supprime l'utilisateur. Le hub garde toujours au moins un administrateur, et vous ne pouvez pas vous supprimer vous-même.

La modification ou la suppression d'un utilisateur met fin à ses connexions. Chaque modification des utilisateurs est consignée dans le journal d'audit.

Connexion avec le compte d'entreprise (OpenID Connect)

Les personnes peuvent se connecter à la vue web avec leur compte d'entreprise : AD FS, Microsoft Entra ID, Keycloak, Okta et d'autres fournisseurs OpenID Connect. La page de connexion affiche alors le bouton **Sign in with your company account**. Ajoutez à `hub.json` :

```
"oidc": {
  "issuer": "https://adfs.example.com/adfs",
  "client_id": "<l'ID client fourni par le fournisseur>",
  "redirect_uri": "https://hub.example.com/auth/callback",
  "groups_claim": "groups",
  "roles": { "admin": ["Compass-Admins"], "architect": ["Compass-Architects"], "viewer": ["Compass-Viewers"] },
  "password_login": true
}
```

- **Chez le fournisseur** : inscrivez le hub comme application web confidentielle avec le flux *authorization code*, l'URI de redirection ci-dessus (l'adresse du hub avec `/auth/callback`), les portées `openid profile email` (AD FS : le hub ajoute lui-même `allatclaims` ; la clé `scopes` permet de les modifier) et une revendication `groups` contenant les noms des groupes dans le jeton d'identité. Le secret client est le secret du hub `OIDC_CLIENT_SECRET` (`LOONAR_SECRET_OIDC_CLIENT_SECRET`).
- **Vérifications** : la connexion utilise PKCE et un state à usage unique ; le hub vérifie lui-même le jeton d'identité : la signature RS256 avec les clés du fournisseur, l'émetteur, l'audience, l'expiration et le nonce.
- **Rôles à partir des groupes** : `roles` indique les groupes qui donnent chaque rôle (par défaut Compass-Admins, Compass-Architects et Compass-Viewers) ; `groups_claim` nomme la revendication des groupes (par défaut `groups`). Le rôle est redéfini à chaque connexion, et le plus élevé l'emporte. Une personne qui n'appartient à aucun de ces groupes n'a aucun accès.

- **Nom d'utilisateur :** `name_claim` choisit la revendication (par défaut `upn`, puis `preferred_username`, `email`). Si un utilisateur local du hub porte le même nom, la connexion d'entreprise est refusée jusqu'à ce que l'administrateur renomme ou supprime cet utilisateur.
- **Uniquement la connexion d'entreprise :** `"password_login": false` désactive la connexion par nom et mot de passe. Les jetons API de Compass et des clients MCP continuent de fonctionner. `button` modifie le texte du bouton.
- Chaque connexion d'entreprise est consignée dans le journal d'audit.

Configuration (hub.json)

```
{
  "listen": "127.0.0.1:8730",
  "sources": [
    {
      "id": "shop-db",
      "name": "Shop (PostgreSQL)",
      "properties": "shop-db/app.properties"
    },
    {
      "id": "shop-file",
      "name": "Shop (file)",
      "properties": "shop-file/app.properties"
    }
  ],
  "users": [
    {
      "name": "anna",
      "role": "admin",
      "password_pbkdf2": "pbkdf2-sha256$210000$...."
    },
    {
      "name": "bob",
      "role": "architect",
      "token_sha256": "<SHA-256 du jeton>"
    }
  ],
  "integrations": {
    "jira": {
      "url": "https://entreprise.atlassian.net",
      "user": "<service account>",
      "project": "ARCH",
      "issue_type": "Task"
    },
    "confluence": {
      "url": "https://entreprise.atlassian.net/wiki",
      "user": "<service account>",
      "space": "ARCH",
      "parent": "65818"
    },
    "policy": {
      "min_approval": "ask",
      "ask_min_role": "viewer",
      "ask_per_hour": 20,
      "governance": { ... governance.json ... },
      "event_agents": [
        {
          "name": "Keep approved",
          "event": "element_deleting",
          "mode": "guard",
          "applies_to": "Requirement",
          "prompt": "Block deleting a requirement whose status is Approved."
        }
      ]
    }
  }
}
```

- **listen** : l'adresse et le port ; `127.0.0.1:8730` par défaut, `0.0.0.0:8730` dans Docker. Au-delà d'un seul ordinateur, placez le hub derrière HTTPS (voir « Sécurité du hub »).
- **sources** : les modèles de l'équipe. `properties` est le fichier `app.properties` du modèle concerné (même format que le fichier de paramètres de Compass), relatif à `hub.json`. Un modèle sur SQL Server, MySQL ou Oracle y indique `model.source=sqlserver`, `mysql` ou `oracle` (chapitre 5), avec le mot de passe du secret `DB_PASSWORD` ; le serveur a besoin du pilote ODBC de l'éditeur.
- **users** : vous n'avez pas à les écrire. Omettez `users` pour obtenir l'administrateur par défaut, et gérez les utilisateurs dans l'onglet **Users** : le hub les réécrit dans `hub.json` uniquement sous forme d'empreintes (`password_pbkdf2`, `token_sha256`). Une entrée avec un `"token"` en clair (au moins 16 caractères aléatoires) est toujours acceptée et convertie en `token_sha256` lorsque le hub enregistre les utilisateurs. `compass-hub --hub-token` (ou `LoonarCompass.exe --hub-token`) affiche un nouveau jeton aléatoire et son empreinte. Les noms et les jetons doivent être uniques.
- **integrations** : le Jira et le Confluence de l'équipe (voir ci-dessous) ; les jetons proviennent des secrets du hub.
- **policy** : `min_approval` (`ask`, `plan` ou `auto`), `ask_min_role` (qui peut utiliser **Ask** dans la vue web ; `viewer` par défaut), `ask_per_hour` (questions par utilisateur et par heure ; 20 par défaut), `governance` au format de `governance.json` et `event_agents` : les agents d'événements de l'équipe, au format de la liste `"agents"` d'`event_agents.json`, gardiens compris (chapitre 10). Ils fonctionnent dans le Compass dans EA de chacun, avec la mention « (team) ». Les administrateurs modifient la politique dans l'onglet **Team policy** de la vue web.
- **oidc** : la connexion avec le compte d'entreprise (décrite ci-dessus).

Rôle	Droits
viewer	Lecture : vue web, REST, MCP
architect	Comme viewer, plus l'envoi d'entrées d'audit, la modification des notes d'équipe, la décision sur les ensembles de modifications des autres et la création de tickets Jira et de pages Confluence depuis la vue web
admin	Comme architect, plus la gestion des utilisateurs (onglet Users), la modification de la politique (PUT /api/policy , enregistrée dans hub.json) et la libération d'un ensemble de modifications resté « en cours d'application » après l'arrêt d'un Compass : après vérification du modèle, Not applied – free it (non appliqué, le libérer) ou It is in the model – mark applied (il est dans le modèle, le marquer comme appliqué)

Personne, pas même un admin, n'approuve son propre ensemble de modifications. Un ensemble est appliqué en deux étapes : un Compass le réserve puis, après l'enregistrement, signale le résultat ; il ne peut donc pas être appliqué deux fois.

Jira et Confluence depuis le hub

Avec `integrations` dans `hub.json`, les architectes et les administrateurs disposent de boutons dans la vue web. Le hub travaille avec son propre compte de service : `user` (l'e-mail pour Cloud ; vide pour Data Center avec un jeton d'accès personnel) et les jetons des secrets `JIRA_TOKEN` et `CONFLUENCE_TOKEN`.

- **Create Jira issue** sur un élément (**Search** → élément) : un ticket dans le projet `project`, du type `issue_type` (Task par défaut), avec les détails de l'élément dans sa description.
- **Create Jira issue for this change set** dans un ensemble de modifications : un ticket par ensemble, avec sa description et son aperçu ; le ticket est lié dans l'ensemble de modifications.
- **Publish this package to Confluence** dans le contrôle du modèle (**Model check**) d'un paquetage : une page de texte dans l'espace `space` (sous la page `parent`, si elle est indiquée) avec les éléments, leurs descriptions et les relations. Elle ne contient pas d'images de diagrammes, car le hub n'a pas d'EA. La publication suivante du paquetage remplace la page.

Chacune de ces actions est consignée dans le journal d'audit. L'assistant de **Ask** reste en lecture seule : il ne peut créer ni tickets ni pages.

Rapports Word et Excel

Choisissez un paquetage dans l'arborescence et ouvrez **Model check** : les boutons **Word report of the package** et **Excel report of the package** téléchargent un rapport du paquetage et de ses sous-paquetages : la description, les éléments et les relations. Les rapports ne contiennent pas d'images de diagrammes, car le hub n'a pas d'EA. Chaque téléchargement est consigné dans le journal d'audit.

Vue web

À l'adresse `https://<hub>/` (`http://` seulement sur le même ordinateur), vous vous connectez avec votre nom d'utilisateur et votre mot de passe, ou avec **Sign in with your company account** lorsque la connexion d'entreprise est configurée. Elle affiche l'arborescence des paquetages, la recherche, les détails d'un élément avec l'impact, le contrôle du modèle selon les règles de l'équipe, les écarts, le journal d'audit, les notes d'équipe et les ensembles de modifications, ainsi que, pour les administrateurs, les utilisateurs et l'onglet **Team policy** : un éditeur JSON de la politique (`min_approval`, `governance`, `event_agents`,

`ask_min_role` , `ask_per_hour`), enregistrée dans `hub.json` . Dans l'onglet **Ask**, l'assistant répond aux questions sur le modèle. Il ne fait que lire le modèle : il n'a accès ni aux fichiers, ni à Jira, ni à Confluence, ni aux serveurs MCP, et ne peut rien proposer ni enregistrer ; il répond à une question à la fois.

Sécurité du hub

Le hub parle en HTTP simple. Pour un accès depuis d'autres ordinateurs, placez-le derrière un proxy inverse HTTPS (par exemple IIS ou nginx) et donnez aux utilisateurs l'adresse `https://` . Changez immédiatement le mot de passe du compte par défaut. Protégez `hub.json` et le dossier de données par les droits d'accès aux fichiers. Les requêtes sont limitées en taille et en durée. Après 20 mots de passe ou jetons erronés depuis une même adresse, cette adresse est refusée pendant 10 minutes ; un jeton valide continue de fonctionner. Les erreurs internes ne sont affichées que sous forme d'identifiant ; les détails sont dans `hub.log` . La vue web ne conserve la connexion que pendant la session du navigateur et propose un bouton **Sign out** (se déconnecter).

IIS avec ARR devant le hub : le hub écoute seulement sur cet ordinateur (`"listen": "127.0.0.1:8730"`), et un site IIS avec le certificat transmet vers `http://127.0.0.1:8730/` (URL Rewrite + Application Request Routing). Dans les paramètres de proxy ARR, désactivez la réécriture des en-têtes de réponse : `reverseRewriteHostInResponseHeaders="false"` (Gestionnaire IIS → serveur → Application Request Routing Cache → Server Proxy Settings → décochez « Reverse rewrite host in response headers »). Sinon ARR remplace la redirection vers AD FS ou Entra ID par l'adresse du hub et la connexion d'entreprise échoue. L'en-tête `Authorization` doit passer sans changement (par défaut). Le hub ne fait confiance à `X-Forwarded-For` que depuis un proxy sur le même ordinateur : le blocage après des mots de passe erronés vise alors l'adresse du client, et non tout le monde à la fois. Les connexions échouées et refusées (mot de passe et compte d'entreprise) figurent dans l'onglet **Audit trail** avec le résultat X et l'adresse du client, et l'onglet **Users** liste aussi les comptes d'entreprise avec leur rôle et leur dernière connexion (lecture seule : le rôle se change dans les groupes chez le fournisseur).

Serveur MCP

Le serveur MCP (lecture seule) fonctionne en JSON-RPC sur HTTP POST à l'adresse `/mcp` , avec l'en-tête `Authorization: Bearer <jeton>` . Outils : `list_sources` , `list_packages` , `search_elements` , `element_details` , `impact_analysis` , `model_lint` , `gap_analysis` .

GraphQL et tables pour Power BI

D'autres outils lisent les modèles du hub avec un jeton API (onglet **Users**, en-tête `Authorization: Bearer <jeton>`) ; tous les rôles peuvent lire, et rien ne peut être modifié.

- **GraphQL** : `POST /api/sources/{sourceId}/graphql` avec `{"query": "..."} .` Requêtes : `packages` , `package` , `elements(name, type, stereotype, package, limit)` , `element` , `diagrams` , `diagram` . Les éléments ont `attributes` , `operations` , `taggedValues` , `connectors` (avec `source` et `target`) et `package` . Le schéma en SDL se trouve à `GET /api/graphql/schema` .
- **Tables** : `GET /api/sources/{sourceId}/table/{table}` renvoie une table plate en JSON, pour `elements` , `connectors` , `packages` , `diagrams` , `attributes` ou `tagged_values` .
- **Power BI : Obtenir les données** → **Web** → **Avancé** ; l'adresse d'une table et, dans les en-têtes de la requête HTTP, `Authorization` avec la valeur `Bearer <jeton API>` . Excel et d'autres outils fonctionnent de la même manière.

```
{ elements(type: "Component", limit: 20) { name stereotype taggedValues { name value } connectors { type target { name } } } }
```

Clés d'IA, Jira et Confluence sur le hub

Le hub peut conserver les clés Claude et les comptes Jira et Confluence de l'équipe, pour que les personnes ne les aient pas sur leur ordinateur. Les clés sont des secrets du hub (variables

`LOONAR_SECRET_<NOM>` ou fichiers `LOONAR_SECRET_<NOM>_FILE`), et `hub.json` indique qui utilise laquelle :

```
"ai": {
  "keys": [
    {"for": "anna.architekt@lab.local", "secret": "claude_key_anna"},
    {"for": "role:architect", "secret": "claude_key_team"},
    {"for": "*", "secret": "claude_key_team"}
  ],
  "limits": [
    {"for": "role:viewer", "tokens_per_day": 50000},
    {"for": "*", "tokens_per_day": 300000, "tokens_per_month": 5000000}
  ],
  "models": ["claude-haiku-5-5", "claude-sonnet-5-5"]
},
"policy": {"ai_via_hub": true, "jira_via_hub": true, "confluence_via_hub": true, "web_access": false}
```

- **keys** – une clé pour une personne (nom du hub ou UPN du compte d'entreprise), pour un rôle (`role:viewer`, `role:architect`, `role:admin`) ou pour tous (`*`) ; la première entrée pour la personne l'emporte, puis pour le rôle, puis `*`. Sans clé, la requête est refusée.
- **limits** – jetons (entrée + sortie) par jour et par mois, dans le même ordre. La consommation de chacun s'affiche dans l'onglet **Users** (aujourd'hui et ce mois-ci) ; détails dans `ai_usage.jsonl`.
- **models** – facultatif : seulement ces modèles.
- **Politique** : `"ai_via_hub": true` – Compass interroge Claude via le hub (sans clé propre) ; `"jira_via_hub"` et `"confluence_via_hub"` – Compass utilise Jira et Confluence via le hub, avec le compte de la section `integrations` (un viewer ne fait que lire, chaque modification figure dans l'audit avec le nom de la personne), et le projet et l'espace de l'équipe sont les valeurs par défaut.

Stratégie de l'ordinateur : imposer le hub (stratégie de groupe, Intune)

La politique d'équipe du hub s'applique à ceux qui y sont connectés. Pour qu'elle s'applique à tous, le service informatique définit la **stratégie de l'ordinateur** dans `HKLM\SOFTWARE\Policies\Loonar\Compass` (pour l'ordinateur) ou `HKCU\SOFTWARE\Policies\Loonar\Compass` (pour l'utilisateur), par stratégie de groupe ou Intune. Un utilisateur standard ne peut pas la modifier. Les règles ne font que durcir les paramètres :

- **HubUrl** – l'adresse du hub d'équipe ; le champ des paramètres est verrouillé.
- **RequireHub** = 1 – l'assistant ne fonctionne que connecté au hub et avec sa politique. Si le hub ne répond pas pendant un moment, Compass utilise la dernière politique reçue (au plus une semaine) ; sans elle, l'assistant ne fonctionne pas.
- **WebAccess** = 0 – pas d'accès au web (le bouton  est verrouillé).
- **UpdateCheck** = 0 – pas de vérification quotidienne d'une nouvelle version.
- **MinApproval** = ask, plan ou auto – le mode d'approbation le moins strict.
- **AllowedProviders** – les fournisseurs d'IA autorisés, p. ex. `anthropic` ou `anthropic,openai`.

Connexion avec le compte d'entreprise : dans Paramètres → Équipe (Compass Hub), le bouton **Se connecter avec le compte d'entreprise** ouvre la connexion du hub dans le navigateur (AD FS, Entra ID, Keycloak...) ; ensuite le hub renvoie la session à une adresse à usage unique sur cet ordinateur (127.0.0.1) et Compass la conserve dans le Gestionnaire d'identification à la place d'un jeton. Une session dure 12

heures ; il suffit ensuite de se reconnecter. Les modèles de stratégie de groupe (`LoonarCompass.admx` et les fichiers de langue `.adml`) se trouvent dans le dossier `deploy\policies` du dépôt : copiez-les dans `C:\Windows\PolicyDefinitions` ou dans le magasin central du domaine. Toute personne ayant les droits d'administrateur sur l'ordinateur peut contourner la stratégie, comme pour toute application.

23. Mode de contrôle pour les pipelines CI

Avec `--check`, Loonar Compass contrôle le modèle sans fenêtre et sans modèle d'IA, par exemple dans un pipeline CI après chaque modification du modèle ou du code. Les contrôles sont effectués par le programme : le même modèle donne donc toujours le même résultat, et rien n'est écrit dans le modèle.

```
LoonarCompass.exe --check --model <fichier.qea|app.properties>
[--package <id|nom>] [--rules governance.json] [--code <dossier>]
[--requirements <document>] [--cmdb <export.csv|json>] [--report rapport.md]
[--junit junit.xml] [--fail-on error|warning|info|never] [--lang en|pl|de|fr]
```

Option	Signification
<code>--model</code>	Un fichier EA .qea ou .qeax, lu directement et en lecture seule (Enterprise Architect n'est pas nécessaire), ou le fichier <code>app.properties</code> d'une installation de Compass, par exemple pour un modèle PostgreSQL, SQL Server, MySQL ou Oracle. Sans cette option : les paramètres de l'utilisateur Windows qui lance la commande.
<code>--package</code>	Le paquetage à contrôler (id ou nom), avec ses sous-paquetages. Sans cette option : tous les paquetages racines du modèle.
<code>--rules</code>	Un <code>governance.json</code> avec les règles de l'équipe (chapitre 7). Sans cette option : le <code>governance.json</code> situé à côté de l' <code>app.properties</code> indiqué ou la politique de son hub ; pour un fichier de modèle, seules les règles intégrées.
<code>--code</code>	Un dossier de code source : le modèle est comparé au code (section 8.6) ; avec <code>--requirements</code> , chaque exigence est aussi suivie jusqu'au code et aux tests (chapitre 7, « Chaîne des exigences »).
<code>--requirements</code>	Un document d'exigences : .docx, .pdf, .pptx, .vsdx, .xlsx, .md, .txt ou .html (section 8.5).
<code>--cmdb</code>	Un export de la CMDB, CSV ou JSON (chapitre 7, « CMDB et modèle »).
<code>--report</code>	Enregistre aussi le rapport Markdown dans ce fichier.
<code>--junit</code>	Enregistre un fichier JUnit XML pour le pipeline.
<code>--fail-on</code>	La gravité la plus basse qui fait échouer le contrôle : <code>error</code> (par défaut), <code>warning</code> , <code>info</code> ou <code>never</code> .
<code>--lang</code>	La langue du rapport : <code>en</code> (par défaut), <code>pl</code> , <code>de</code> ou <code>fr</code> .

Ce qui est contrôlé

Partie	Quand	Gravité
Contrôle du modèle : contrôles intégrés, relations ArchiMate 3.2, règles de l'équipe	toujours	comme dans le contrôle du modèle (chapitre 7) : erreur, avertissement ou information
Modèle et code	<code>--code</code>	écarts : avertissement
CMDDB et modèle	<code>--cmdb</code>	écarts : avertissement
Exigences	<code>--requirements</code>	une exigence absente du modèle, qui n'a que des candidats (sans relation de traçabilité) ou dont l'identifiant correspond mais le contenu diffère : avertissement ; avec <code>--code</code> , une exigence sans code ou sans tests : information

Résultat

- **Rapport** : du Markdown sur la sortie standard. Il commence par le résultat, PASSED ou FAILED, et le nombre de groupes d'erreurs et d'avertissements, suivi d'une section pour chaque partie.
- **JUnit** : une suite de tests par partie et un cas de test par groupe de constats. Les groupes dont la gravité atteint ou dépasse `--fail-on` sont des échecs : le pipeline les affiche comme des tests en échec.
- **Code de sortie** : 0 réussi ; 1 constats à partir de `--fail-on` ; 2 arguments erronés, modèle illisible ou fichier impossible à écrire (la cause va sur la sortie d'erreur).

Un programme à fenêtre dans PowerShell. `LoonarCompass.exe` est un programme à fenêtre : PowerShell ne l'attend donc pas. Redirigez sa sortie, par exemple `... | Out-Host` : PowerShell attend alors, affiche le rapport et définit `$LASTEXITCODE`. Le mot de passe d'un modèle en base de données provient de la variable d'environnement `LOONAR_DB_PASSWORD` (une variable secrète du pipeline) et a priorité sur le Gestionnaire d'identification.

GitLab CI

Un job sur un runner Windows (PowerShell) où Loonar Compass est installé pour tous les utilisateurs. Le job exécute le contrôle et publie `junit.xml` comme rapport de tests :

```
compass-check:
  tags: [windows]
  script:
    - '& "C:\Program Files\Loonar Compass\LoonarCompass.exe" --check --model model\shop.qea --rules governance.json --code src --junit junit.xml --report report.md | Out-Host'
    - 'if ($LASTEXITCODE -ne 0) { exit $LASTEXITCODE }'
  artifacts:
    when: always
    paths: [report.md]
    reports:
      junit: junit.xml
```

GitHub Actions

Un job sur un runner Windows auto-hébergé où Loonar Compass est installé pour tous les utilisateurs. Le job exécute le contrôle et téléverse `junit.xml` et le rapport comme artefact, qu'une étape de rapport JUnit de votre choix peut afficher comme résultats de tests :

```
jobs:
  compass-check:
    runs-on: [self-hosted, windows]
    steps:
      - uses: actions/checkout@v4
      - name: Model check
        shell: pwsh
        run: |
          & "C:\Program Files\Loonar Compass\LoonarCompass.exe" --check --model model\shop.qea --
rules governance.json --code src --junit junit.xml --report report.md | Out-Host
          exit $LASTEXITCODE
      - name: Publish the results
        if: always()
        uses: actions/upload-artifact@v4
        with:
          name: compass-check
          path: |
            junit.xml
            report.md
```

Pour un modèle PostgreSQL, indiquez à `--model` un `app.properties` sans mots de passe et définissez `LOONAR_DB_PASSWORD` : dans GitLab comme variable CI/CD masquée, dans GitHub à partir d'un secret (`env: LOONAR_DB_PASSWORD: ${ secrets.EA_DB_PASSWORD }`).

24. Licence, marques et contact

Licence. Loonar Compass est un logiciel gratuit (freeware) : l'utilisation est gratuite, y compris à titre professionnel et commercial, sans limite de durée ni de territoire ; la modification de l'application est interdite. La licence contient aussi des règles concernant les résultats de l'IA, les données et les composants open source. Le texte intégral se trouve dans la vue **À propos** et dans le fichier `LICENSE.txt` du dossier d'installation ; seule la version polonaise de la licence fait foi. La liste des fonctionnalités et des limitations connues de cette version se trouve dans la vue **Nouveautés**.

Confidentialité et mentions légales. Politique de confidentialité : <https://loonar.eu/fr/privacy/>. Mentions légales : <https://loonar.eu/fr/legal/>.

Composants open source. L'application contient des bibliothèques et des polices open source ; chacune conserve sa propre licence. Leur liste se trouve dans le fichier `THIRD-PARTY-NOTICES.txt` du dossier d'installation ; dans la vue **À propos**, le bouton **Afficher les licences des composants** ouvre ce fichier.

Soutien. Si l'application vous est utile, vous pouvez soutenir son développement par une contribution volontaire : vue **Soutenir le projet** ou <https://suppi.pl/loonar>.

Contact. Propriétaire de l'application : Loonar Mariusz Głowacki, ul. Gombrowicza 12, 73-110 Stargard. E-mail : mariusz@loonar.eu, site : <https://loonar.eu>, code source : <https://github.com/loonar-eu/loonar-compass>. Le plus simple pour signaler un bug est d'utiliser la vue **Signaler un bug**.

Loonar Compass 0.9. Sparx Systems et Enterprise Architect sont des marques déposées de Sparx Systems Pty Ltd. Claude et Anthropic sont des marques d'Anthropic, PBC. OpenAI et GPT sont des marques d'OpenAI. Atlassian, Confluence et Jira sont

des marques d'Atlassian. Microsoft, Windows, Word, Excel, PowerPoint, Visio, SharePoint, SQL Server, Entra ID, Power BI, Azure, Azure DevOps et GitHub sont des marques du groupe Microsoft. Oracle et MySQL sont des marques déposées d'Oracle et/ou de ses sociétés affiliées ; MariaDB est une marque de MariaDB plc ; Okta est une marque d'Okta, Inc. GitLab est une marque de GitLab Inc., ServiceNow de ServiceNow, Inc. et Docker de Docker, Inc. PostgreSQL est une marque déposée de la PostgreSQL Community Association. Loonar Compass est un produit indépendant, sans lien avec ces sociétés, qui ne l'approuvent ni ne le parrainent.