

Livre blanc de la théorie du modèle Alioth

Le modèle de données d'entreprise du point de vue de l'ontologie commutative : fondements mathématiques,
invariants structurels et vérification machine

Version v2.1 · octobre 2026 · Hangzhou Yuqi Technology Co., Ltd.

Implémentation correspondante : modèle Alioth v10.0.44 (publication vérifiée le 2026-10-19) · plateforme AliothStudio

Sommaire

- Résumé
- Domaine de problème et objectifs de conception
- Catégorie monoïdale symétrique et symétrie
- Espace ontologique : produit de coordonnées et orthogonalité
- Dimension temporelle : ordre total des états et involution miroir
- Complétude des cinq domaines
- Algèbre des relations structurelles
- Externalisation des scalaires et mesure de conservation
- Point de vue de la théorie des modèles : théorie et modèles multi-espaces de noms
- Vérification machine et boucle d'ingénierie
- Conclusion

1. Résumé

Alioth est un modèle de données d'entreprise dont le support physique est l'héritage de tables PostgreSQL et dont le cadre sémantique est l'ontologie commutative (Commutative Ontology). Le présent livre blanc expose ses fondements mathématiques : le comportement économique y est formalisé comme un morphisme d'une

catégorie monoïdale symétrique, les transactions réversibles forment un **groupoïde** (groupoid) muni d'un endofoncteur symétrique involutif ; chaque entité métier occupe une coordonnée unique dans l'espace produit orthogonal constitué de la scène (Scene), du facteur (Factor) et de la fonction (Function), la chaîne d'état principale fournissant une quatrième dimension de temps logique discret. Les propriétés structurelles du modèle — conservation symétrique, orthogonalité des coordonnées, couverture des cinq domaines, acyclicité des relations, additivité de la mesure — sont énoncées sous forme de douze invariants exécutables, chacun ancré à une porte qualité d'ingénierie ; cinq métathéorèmes centraux ont déjà été vérifiés par machine avec le prouveur de théorèmes Lean 4. La même théorie a été validée par induction sur des déploiements en production dans six espaces de noms sectoriels.

2. Domaine de problème et objectifs de conception

Le dilemme classique du modèle de données d'entreprise est le suivant : la sémantique métier se disperse dans la conception empirique de plusieurs centaines de tables, faute de critère objectif permettant de décider « ce qu'est un objet métier légitime » ; l'intégration de plusieurs domaines métier se fait par copier-coller et le modèle n'est pas réutilisable ; la correction repose sur la discipline de configuration plutôt que sur une garantie structurelle. Les objectifs de conception d'Alioth sont les suivants :

Objectif	Signification
Coordonnées sémantiques	Toute entité métier peut être localisée par une coordonnée unique, la coordonnée elle-même portant la sémantique du domaine
Invariants structurels décidables	Chaque propriété du modèle est un prédicat mécaniquement vérifiable, et non une convention documentaire
Théorie définie une fois, réuti - lisée sur plusieurs domaines	L'intégration d'un nouveau domaine métier ne modifie pas la structure du modèle (zéro DDL) ; elle ne fait qu'attribuer une interprétation aux coordonnées
Théorie vérifiable par machine	Les métathéorèmes centraux peuvent être revérifiés indépendamment par un prou - veur de théorèmes

3. Catégorie monoïdale symétrique et symétrie

3.1 Catégorie monoïdale symétrique

En prenant les agents économiques pour objets et les transactions pour morphismes, on obtient une catégorie $\mathcal{E}$: la composition des transactions est une chaîne de transactions et l'élément neutre est la transaction nulle. Notons que $\mathcal{E}$ est une catégorie et non un groupe — une transaction quelconque n'est pas réversible, et l'on ne peut donc pas la présenter comme un « groupe d'agents économiques » (l'ensemble des morphismes entre objets n'admet pas d'opération binaire globale).

3.2 Symétrie : foncteur involutif sur le groupoïde noyau

Le noyau réversible $\text{core}(\mathcal{E})$ de $\mathcal{E}$ est le groupoïde formé de toutes les transactions réversibles. L'énoncé rigoureux du principe de symétrie est : il existe un endofoncteur involutif

$$S : \text{core}(\mathcal{E}) \rightarrow \text{core}(\mathcal{E})^{\text{op}}, S^2 = \text{id}$$

autrement dit, toute transaction réversible admet une transaction inverse miroir, et le miroir du miroir est la transaction elle-même. L'involutivité garantit que la relation de miroir n'a aucune préférence d'orientation — c'est là le sens exact de l'énoncé « toute transaction admet une transaction inverse symétrique ». Le noyau au niveau fonctionnel (l'application involutive est une bijection auto-inverse) est déjà prouvé par machine (cinq théorèmes dont `Symmetry.Involutive.injective_surjective`).

3.3 Lois de conservation

Soient une attribution de dotation et une évaluation de droits ; la loi de conservation exige que les droits échangés par les deux parties du miroir soient de valeur égale : $v(M) = v(D)$, de sorte que la dotation totale du système reste invariante sous toute composition de transactions réversibles (théorème `Lean Conservation.endowment_conservation`). Au niveau de l'implémentation, la conservation n'est pas une déclaration mais un témoignage pièce par pièce : la classe de base des pièces justificatives renseigne, pour chaque pièce, le solde d'ouverture et le solde de clôture, rendant « état antérieur + variation nette = état postérieur » mécaniquement vérifiable sur chaque ligne ; les pièces d'entrée/sortie de stock et les pièces d'écriture financière héritent de la même classe de base — la conservation physique et la conservation financière sont isomorphes.

4. Espace ontologique : produit de coordonnées et orthogonalité

4.1 Êtres vivants et foncteur de coordonnées

Toutes les entités métier dotées d'un cycle de vie descendent par héritage de la table racine `zc_id_lifecycle` (la publication courante v10.0.44 compte 1022 tables isahl et 1204 arêtes d'héritage, dont 374 tables feuilles de cycle de vie ; les coordonnées tridimensionnelles totalisent actuellement 645 entrées : scènes 109 + facteurs 118 + fonctions 418). Trois clés étrangères constituent le foncteur de coordonnées :

$$\chi = (\text{dk_scene}, \text{dk_factor}, \text{dk_function}) : \text{Ob}(\mathcal{L}) \rightarrow S \times F \times U$$

S = scène métier (Where & When) F = facteur métier (Who & What) U = fonction métier (How)

Les trois dimensions forment une **catégorie produit** — chacune prend ses valeurs et évolue indépendamment, ce qui est le sens rigoureux de l'orthogonalité. Toute entité de cycle de vie occupe un point unique dans l'espace ontologique. À l'intérieur d'une dimension, les clés étrangères de catégorie organisent un ordre partiel forestier ; le code de dimension `code = code catégorie || symbole[numéro]` (jeu de 42 caractères) est l'adresse hiérarchique dans la forêt, attribuée automatiquement par déclencheur au moment de l'écriture.

4.2 Instantané quaternaire

L'état n'est pas porté par une colonne de l'entité, mais rattaché par une table de relation. La chaîne d'état principale est un ensemble totalement ordonné (évolution unidirectionnelle, par exemple naissance→mort) et constitue un temps logique discret ; `status_date` la projette sur le temps physique. L'instantané quaternaire de l'entité au nœud d'état t est :

$$\Sigma(e, t) = (\text{scene}, \text{factor}, \text{function}, \text{status}) \in S \times F \times U \times St$$

Les quatre dimensions sont toutes orthogonales ; les trois premières sont des dimensions spatiales, l'état est la dimension temporelle.

5. Dimension temporelle : ordre total des états et involution miroir

L'ensemble des tranches temporelles est partagé en deux moitiés par le « maintenant » : celles qui ont eu lieu sont des événements (event, faits passés), celles qui n'ont pas eu lieu sont des tâches (task, à-faire futurs). Les deux sont le miroir l'un de l'autre, de part et d'autre du présent, d'un même concept de tranche temporelle ; l'achèvement d'une tâche fait basculer la tranche (task → event) et la trace de l'action est consignée par le support opération (operation). Le plan (plan) se déploie de part et d'autre de l'axe temporel au moyen de deux ponts symétriques (groupe plan→event et groupe plan→task), organisant l'intention intertemporelle en un

point d'ancrage unique ; l'avancement est une référence scalaire et non une entité, et le diagramme de Gantt est la projection naturelle du système de planification, non un modèle autonome.

6. Complétude des cinq domaines

L'appartenance à un domaine se déduit par la chaîne de catégories de la dimension facteur :

```
domain = a_type_ ◦ ck_category ◦ dk_factor : Ob(ℒ) → {L, C, G, P, F, ~}
L Personne (sujet)  C Contenant (médium · réceptacle)  G Marchandise (objet)  P Lieu
(médium · emplacement)  F Finance (processus et information)  ~ joker
```

Un instantané est complet si et seulement si l'image de l'application de domaine couvre les cinq domaines : $\text{domain}(\Sigma) \supseteq \{L, C, G, P, F\}$ — aucune rupture de symétrie n'est tolérée. La complétude est un **prédicat décidable** (décision de couverture d'un ensemble fini, réalisable par machine dans l'instance Lean), vérifié instantané par instantané par le raisonneur de la phase de génération ; les domaines manquants sont complétés par fermeture vers le haut le long de l'ordre partiel d'héritage, et la terminaison comme l'idempotence de cette complétion sont prouvées par machine (`Completion.isAncestor_height_lt`, `closure_idempotent`).

Note d'honnêteté théorique : la couverture des cinq domaines, en tant qu'assertion sur les échanges réels, est un postulat et non un théorème — son statut est celui des lois de conservation en physique ; elle s'appuie sur la falsifiabilité et sur les preuves inductives issues de plusieurs espaces de noms (voir section 9), et le présent modèle ne la déguise pas en proposition démontrable.

7. Algèbre des relations structurelles

7.1 Ordre partiel d'héritage

L'ensemble des tables et la relation d'héritage forment un ordre partiel en graphe orienté acyclique $(\mathcal{T}, \leq)$: la racine est la table d'objets abstraits, le plus grand sous-arbre métier est la chaîne de cycle de vie ; les tables feuilles sont les éléments minimaux de l'ordre partiel et le seul point de chute légal pour l'écriture de données ; l'héritage multiple fait de cette structure un DAG et non un arbre. L'extension du modèle ne peut être qu'une croissance par cône inférieur de l'ordre partiel (enregistrer un sous-type et le rattacher par héritage) ; la duplication autonome de tables est interdite — la structure conserve ainsi une source de vérité unique.

7.2 Familles de colonnes : projections typées

Les colonnes d'entité sont réparties en familles selon leur préfixe, chaque famille étant une projection typée vers un système d'héritage déterminé : **dk_*** (coordonnées ontologiques), **fk_*** (associations à un être vivant), **qk_*** (pointeurs d'espace de valeurs), **sk_*** (unités), **ck_*** (catégories), **tk_*** (étiquettes), **lk_*** (niveaux), **tpl_id** (pont instance→exemple). Toutes sont unifiées en bigint — une colonne de référence est un pointeur d'un univers de types unique, la distinction de type étant portée par le préfixe (la catégorie cible).

7.3 Familles de relations : relations fonctionnelles et span

Les tables de relation **r_*** (ref_right unique) encodent une relation fonctionnelle (1:N, chaque enfant appartenant à exactement un parent) ; les tables de pont **rr_*** encodent un span (relation arbitraire M:N) ; les colonnes **fk_*** encodent une fonction N:1. La composition de relations est la composition de morphismes ; les relations d'imbrication et de hiérarchie exigent l'invariant DAG (chaîne d'ancêtres sans cycle, prouvé par machine `isAncestor_irrefl`), et le rollup est une fonctionnelle de sommation sur la fermeture transitive.

7.4 Les six quadrants : phase × niveau d'abstraction

Le préfixe du code de fonction porte la structure produit {création, conception, réalisation} × {exemple, instance} (six quadrants) ; la classification des entités est calculée mécaniquement à partir du préfixe du code par un homomorphisme dérivé, limitée à ces six paires, toute écriture manuelle étant interdite. L'axe exemple/instance est aussi l'axe de propriété : les exemples sont des actifs publics de l'organisation (une fois référencés, ils ne peuvent être supprimés et leurs versions ne peuvent être effacées), les instances sont des productions privées (librement gérées par leur propriétaire) ; les seuls flux valides entre quadrants sont deux flux unidirectionnels, du privé vers le public.

7.5 Arbitrage des concepts dérivés : l'échantillon n'entre pas dans la structure

Arbitrage ontologique (« ALIOTH_ONTOLOGY_SPEC » § 8.5.11, octobre 2026) : l'échantillon (样品) est un concept dérivé du lot/de la branche — l'entité produit ne doit ni ajouter de colonne d'identification d'échantillon, ni créer de sous-table produit pour les échantillons ; la sémantique d'échantillon se dérive toujours via les enregistrements de lot/branche et les ponts de classification/relation. C'est l'application la plus récente du principe de « source unique de vérité structurelle » : toute sémantique dérivable des coordonnées et relations existantes ne doit pas se sédimenter en nouveau point structurel. Le même mois, la famille de concepts goods a été redécoupée au niveau du modèle (4 groupes d'identifiants renommés), et le runtime, les seeds et les artefacts dérivés de la plateforme ont convergé dans le même lot — une démonstration complète du mécanisme de propagation « le modèle au centre, le code comme produit ».

8. Externalisation des scalaires et mesure de conservation

Les valeurs ne sont pas encapsulées dans la ligne d'entité mais externalisées dans un système d'héritage scalaire (valeur + unité + précision) : l'entité détient un pointeur, et une même valeur n'est stockée qu'une seule fois dans toute la base — l'externalisation des objets-valeurs fait de « l'égalité de valeur » une décision mécanique de « pointeur identique » ; toute arithmétique sur une colonne de pointeur est donc dépourvue de sens et doit passer d'abord par une résolution. Les relations d'accompagnement portent une durée de validité ; les statistiques à l'instant t n'évaluent que les relations en vigueur (accompagnement spatio-temporel). Le stock n'est pas un attribut d'entité : c'est une mesure signée additive de faits justificatifs :

$$\mu_t(P, S) = \Sigma \text{ variation nette des pièces (income - outgo)} + \Sigma \text{ rollup des sous-stocks imbriqués}$$

La mesure satisfait l'additivité et la signabilité ; la valeur matérialisée est un cache de la mesure, entretenu de façon incrémentale par un point d'entrée unique. L'inventaire est un événement atomique de correction de la mesure : l'écart entre le comptage physique et la valeur matérialisée (excédent / manquant) est absorbé dans la mesure par une pièce de recalibrage au lieu d'être masqué.

9. Point de vue de la théorie des modèles : théorie et modèles multi-espaces de noms

Résumons dans le langage de la théorie des modèles : la structure du modèle (ordre partiel d'héritage figé + codes de dimension + ensemble d'invariants) est la **théorie** $\mathbb{T}$; chaque déploiement d'espace de noms sectoriel est un **modèle** $\mathcal{M} \models \mathbb{T}$ de $\mathbb{T}$ — la sémantique métier se dépose dans l'interprétation locale des codes de coordonnées, sans aucune bifurcation structurelle. En octobre 2026, une même théorie porte déjà six espaces de noms :

Espace de noms	Applications	Modules	Block	Services	Domaine métier
Alioth	3	11	50	20	La plateforme elle-même
WZ	4	16	90	16	Collaboration logistique externe
AVIC-CAASEC	2	5	65	11	Approbation de projets de recherche
SE	1	6	32	6	Gestion budgétaire
Cosmic-Tools	4	23	56	28	Suite outils/WMS/ERP
MAT	1	0	0	1	Nouveau domaine en cours d'intégration
Total	15	61	293	82	—

L'intégration d'un nouvel espace de noms (par exemple SE et MAT) s'effectue par « localisation par héritage + liaison de coordonnées + assemblage de configuration », sans aucun DDL ni correctif de modèle — la capacité d'expression de la théorie pour de nouveaux domaines métier n'a pas été mise en défaut, ce qui constitue une preuve inductive de l'orthogonalité des coordonnées et de la complétude des cinq domaines.

10. Vérification machine et boucle d'ingénierie

La valeur d'une théorie se juge à son exécutabilité. Les douze invariants mathématiques (conservation, orthogonalité, couverture, acyclicité, monotonie, cohérence des six quadrants, externalisation des valeurs, préservation de la structure, non-suppression du domaine public, pureté de l'héritage, miroir temporel, additivité de la mesure) sont mis en correspondance un à un avec les portes qualité d'ingénierie en vigueur (portes enregistrées actuelles : 229 — 137 en phase commit / 92 en phase push) ; parmi eux, cinq métathéorèmes centraux sont vérifiés par machine avec le prouveur de théorèmes Lean 4 (builds verts depuis le 2026-09-18, zéro dépendance de bibliothèque externe, zéro placeholder de preuve) :

Invariant	Preuve machine
Conservation de la valeur (T3)	Conservation.endowment_conservation
Couverture des 5 domaines (décidabilité)	Domain.complete_iff_image_covers
Acyclicité des relations (corollaire de T1)	Completion.isAncestor_irrefl
Pureté de l'héritage / soundness de la complétion (T4)	Completion.closure_superset / closure_idempotent
Miroir temporel (T2, couche involutive)	Symmetry.Involutive.injective_surjective

Il en résulte une boucle en trois couches : formulation mathématique (le présent livre blanc) → portes qualité d'ingénierie (application continue de prédicats calculables) → preuve machine (revérification indépendante par le prouveur de théorèmes).

11. Conclusion

La thèse fondamentale d'Alioth est la suivante : la correction d'un modèle de données d'entreprise doit reposer sur la structure et les mathématiques, et non sur la discipline et l'expérience. La sémantique coordonnée, les invariants décidables, la théorie définie une fois et réutilisée sur plusieurs domaines, ainsi que les métathéorèmes vérifiables par machine, constituent ensemble la chaîne de preuves complète de cette thèse. Nous accueillons favorablement l'examen, la contestation et l'amélioration de cette théorie par les milieux académique et industriel.

Document complémentaire : « Manuel de construction d'applications AliothStudio (Livre bleu) » ; pratique en ligne : <https://cosmic-tools.shop/> .

© 2025-2026 Hangzhou Yuqi Technology Co., Ltd. (CosmicTools.ltd). Le modèle Alioth est publié sous licence MIT.