Chain-of-Shards :
chaînes privées imbriquées avec finalité publique sélective

Résumé

Chain-of-Shards part d'un problème simple : les blockchains publiques sont immuables, mais chaque transaction y consomme un espace soumis au marché du gas. Chain-of-Shards déplace le volume transactionnel dans des chaînes privées imbriquées, où l'application exécute à très haut débit et paie principalement le coût du compute. La chaîne produit ses preuves dans son domaine privé, selon son profil de déploiement, puis publie sur Ethereum les checkpoints qui doivent devenir interopérables, auditables ou consommables par un contrat. Le gas est ainsi réservé aux moments de portée publique : bridge, règlement ou audit.

1 Introduction

Le trilemme de la blockchain est souvent présenté comme l'un des principaux défis structurels des réseaux blockchain : il met en tension la sécurité, la scalabilité et la décentralisation. Bitcoin et Ethereum se tiennent du côté sûr et décentralisé : chacun peut les vérifier, mais ils restent lents et très coûteux. Un second groupe privilégie la rapidité et le faible coût. Les cas de BNB Smart Chain et de Base sont éclairants. Leur interopérabilité est souvent confondue avec la décentralisation, or ce sont deux choses différentes. Ces chaînes sont en grande partie exploitées par une seule entreprise ou un petit ensemble de validateurs. Aujourd'hui, en somme, « rapide et interopérable » signifie presque toujours « rapide et centralisé ».

La propriété la plus précieuse d'une blockchain publique est aussi sa ressource la plus coûteuse : rendre un état immuable. Sur Ethereum, chaque opération inscrite entre dans un marché du gas, donc dans une logique d'enchère, de congestion et de coût global. Chain-of-Shards conserve cette immuabilité pour les moments où elle apporte une valeur externe, tout en déplaçant l'exécution quotidienne dans des chaînes privées rapides et vérifiables.

Cette distinction rejoint les travaux récents sur les rollups, les validiums, les bridges zero-knowledge et la disponibilité des données [1–3]. Elle précise aussi un point central de l'interopérabilité : un fait prouvé hors chaîne acquiert une portée différente lorsqu'il devient lisible sur la couche publique par un contrat de destination. Nos récents travaux de recherche ont formalisé cette frontière avec les états *Local*, *Proved* et *Checkpointed* [4].

Chain-of-Shards est une nouvelle génération de blockchain. Un shard est une chaîne privée appartenant à une famille imbriquée, rattachée à la même couche d'ancrage. Ensemble, ces chaînes forment un tissu polycentrique : chacune conserve son profil de déploiement et sa cadence de publication. Une seule racine publique (seed) peut porter de vastes familles de chaînes privées. Chaque chaîne s'exécute de manière autonome, paie principalement le compute de son domaine (un coût prévisible et faible), puis publie sur Ethereum uniquement les hauteurs qui doivent devenir interopérables, auditables ou consommables par un contrat.

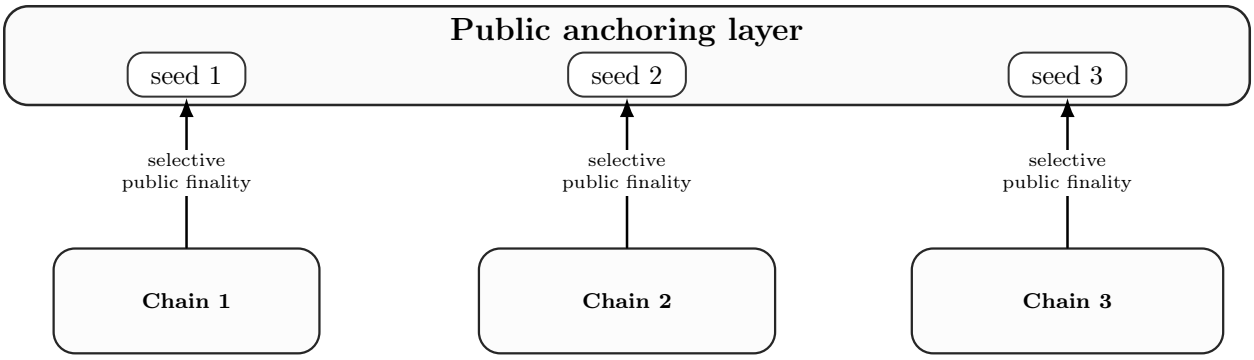

FIGURE 1 – Chaque chaîne part d'une seed publique, s'exécute dans son domaine privé et publie des checkpoints sur Ethereum lorsque certaines hauteurs doivent prendre une portée publique.

2 Définition d'une chaîne

Une chaîne commence par une seed publique. La seed est le point de départ que tout le monde peut relire : elle stocke l'identité de la chaîne et liste ses règles d'exécution. Elle donne une origine stable à la chaîne.

À partir de cette seed, la chaîne produit son historique. À chaque hauteur, elle calcule un engagement qui résume son état : solde, journal ou racine d'état.

Le profil de déploiement décrit la manière dont la chaîne vit réellement. Il répond à quatre questions simples : qui valide la chaîne, comment l'état se récupère, qui gouverne les évolutions et comment l'ordre des blocs est produit. Cette lecture évite de réduire la décentralisation à une seule étiquette. Deux chaînes peuvent partager le même ancrage public et se comporter très différemment une fois en service.

Chaque chaîne décide quelles hauteurs doivent devenir publiques. Les autres hauteurs peuvent rester privées, utiles et prouvées sans être inscrites sur Ethereum. Une chaîne peut donc publier souvent pour servir des bridges ou des règlements rapides, ou publier rarement pour ne rendre publics que les faits décisifs. Toutes conservent une origine vérifiable et une relation de preuve claire.

3 *Zero-knowledge recursive state*

La vitesse vient d'une séparation simple : la chaîne privée avance bloc après bloc, tandis que les preuves zero-knowledge donnent une vérification compacte de ce qui a été exécuté. À mesure que la chaîne avance, le validateur produit des preuves de transition. Chaque preuve condense l'exécution dans un objet court, vérifiable et rattaché à la seed.

Le caractère récursif donne à cette preuve une portée plus forte. Une preuve engage les preuves et les engagements précédents : la hauteur courante porte alors la continuité cryptographique de la chaîne. Un observateur vérifie le résultat courant avec un coût de lecture très faible, tout en gardant une relation claire avec l'historique déjà prouvé.

C'est cet état que le schéma nomme *zero-knowledge recursive state*. Il se situe après l'exécution privée et avant la publication sur Ethereum. Il prépare le passage public sans l'imposer : la preuve donne la validité, le checkpoint donne la portée publique.

4 Les trois états

Nos récents travaux de recherche sur l'interopérabilité cross-chain ont formalisé trois états *Local*, *Proved* et *Checkpointed* [4]. Ces états forment une progression cumulative : un engagement local existe lorsque l'exécution le fixe. Il devient prouvé lorsqu'une preuve hors chaîne justifie la transition d'état. Il devient checkpointé lorsqu'une transaction inscrit la preuve et le rend consommable sur Ethereum.

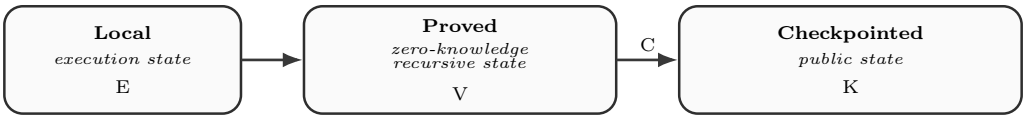

FIGURE 2 – Les trois états et la lecture opérationnelle E -> V -> C -> K.

Concrètement, une hauteur prouvée possède une justification cryptographique compacte. Un observateur peut vérifier cette justification sans rejouer toute l'exécution. Lorsque cette hauteur est checkpointée, le checkpoint public porte l'identité de la chaîne, la hauteur et l'engagement consommable sur Ethereum. À ce stade, l'engagement devient un fait public rattaché à la chaîne.

Cette séparation est déterminante pour les contrats et les bridges. Une preuve hors chaîne établit la validité cryptographique d'un résultat dans le domaine privé. Un contrat public consomme ce résultat au moment où le checkpoint correspondant devient lisible sur Ethereum. La frontière est nette : la preuve donne la validité, le checkpoint donne la portée publique.

Cette architecture permet une gestion plus fine des coûts. La chaîne produit ses preuves dans son domaine privé. Les applications gardent leur cadence interne, puis publient les hauteurs nécessitant un ancrage public : un jeu publiera une racine de tournoi, un système financier publiera un règlement, une chaîne d'entreprise publiera une preuve d'audit, un bridge publiera la hauteur utile à une sortie ou à un transfert inter-domaine.

5 *Checkpointability*

La *checkpointability* répond à une question pratique : la hauteur privée sur laquelle un participant agit serait-elle acceptée par le chemin public si elle était checkpointée maintenant ? Cette question a une valeur économique directe. La chaîne privée concentre l'exécution et la preuve dans son domaine ; Ethereum intervient lorsque la hauteur prend une portée publique. La *checkpointability* donne au participant le moyen de savoir, avant de payer le gas de publication, que la hauteur qu'il utilise possède un passage public admissible.

Nos récents travaux de recherche sur la *checkpointability* ont formalisé cette lecture avec quatre signaux : E pour la hauteur éligible, V pour la preuve vérifiée, C pour l'appel public accepté par simulation fidèle et K pour le checkpoint confirmé [5].

$$\mathbf{E} \longrightarrow \mathbf{V} \longrightarrow \mathbf{C} \longrightarrow \mathbf{K}$$

Un utilisateur qui agit sur une chaîne privée n'a pas besoin de publier immédiatement son action. Il lui suffit de vérifier la preuve. Si cette preuve est valide, il sait que cette hauteur pourra être ancrée publiquement à tout moment. Il peut donc continuer à opérer dans le domaine privé, reporter le paiement du gas, ou laisser un autre participant publier plus tard le checkpoint qui rendra l'état interopérable. Les actifs restent vérifiables dans le domaine privé, puis deviennent mobilisables sur Ethereum une fois le checkpoint correspondant confirmé.

La *checkpointability* donne à la chaîne privée un lien permanent avec Ethereum. Une hauteur prouvée peut rester privée tant que nécessaire, tout en étant publiable sur demande. C'est cette garantie de « publiabilité » qui permet d'opérer en confiance dans le domaine privé.

6 Immuabilité

Un engagement Chain-of-Shards suit un cycle lisible de bout en bout. L'application produit d'abord une transition dans son domaine privé. La chaîne calcule un nouvel engagement, l'inscrit dans son historique local et rattache ce résultat à la seed. À ce stade, le fait existe pour les participants de la chaîne et s'insère dans sa cadence locale.

Le cycle avance ensuite vers la preuve. Le validateur produit une preuve zero-knowledge, puis lie cette preuve à la bonne chaîne par la clé de vérification. La chaîne obtient alors un résultat prouvé : l'état local

possède une justification cryptographique compacte. Dans une construction récursive, la preuve engage les preuves et engagements précédents, sans avoir besoin de tout rejouer. Cette étape porte la finalité locale. Elle donne aux validateurs et aux observateurs une manière de vérifier le résultat avec un coût de lecture très faible.

La publication constitue le troisième temps. La chaîne sélectionne une hauteur, prépare l'appel de checkpoint attendu, puis publie sur Ethereum. Le checkpoint transforme l'engagement prouvé en fait public. Un contrat de bridge, un système de règlement ou un auditeur peut alors consulter le checkpoint public, rattacher la publication à la bonne chaîne, vérifier la preuve et consommer le résultat publié.

Une application peut rester rapide dans son domaine privé, garder ses preuves disponibles, puis rendre public le résultat qui porte une valeur externe. La racine publique (seed) devient le lieu de coordination, la chaîne privée reste le lieu de production, la preuve fait le lien entre les deux. Chain-of-Shards obtient ainsi une architecture où l'état circule par engagements, preuves et checkpoints.

7 Profils de déploiement

Le profil de déploiement rend la décentralisation observable. Il montre comment la chaîne est validée, récupérée, gouvernée et ordonnée.

Tableau 1 – Lecture opérationnelle du profil de déploiement.

Axe	**Question rendue vérifiable**
Validateurs	Qui exécute, prouve, signe et relaie les étapes de la chaîne.
Récupération	Où vivent l'état récupérable, les données de reconstruction et les preuves utiles.
Gouvernance	Qui autorise les mises à jour, rotations de clés, changements de paramètres et évolutions du chemin public.
Ordonnancement	Comment l'ordre local est produit, alterné, attesté et vérifié.

Le profil de déploiement maintient séparés la récupération, la gouvernance et l'ordonnancement, chaîne par chaîne. Cette séparation transforme trois risques en trois surfaces d'attaque distinctes : la rétention de données, la gouvernance et la censure. Elle distingue deux systèmes en apparence identiques, mais qui se comportent très différemment une fois en service.

Certaines hauteurs peuvent rester significatives et prouvées localement sans devenir des faits publics. La finalité publique sélective devient donc un choix explicite : la chaîne décide quand son état doit compter publiquement et servir d'un domaine à l'autre. La cadence des preuves et la cadence des publications restent dissociées, ce qui donne aux applications un contrôle plus fin que l'habituel tout ou rien.

Cette lecture rend ces dimensions mesurables : l'indice de Herfindahl-Hirschmann quantifie la concentration des validateurs, tandis que le coefficient de Nakamoto mesure la robustesse à la censure. La décentralisation devient une mesure objectivable.

8 Finalité

La production d'un bloc est sub-seconde et sa finalité se renforce dans le temps. Dès qu'un bloc n est produit, une preuve zero-knowledge arrive dans un bloc ultérieur $n + x$.

Pour un round donné, chaque validateur exécute :

$$\text{étape } 1 : A, \qquad \text{étape } 2 : B, \qquad \text{étape } 3 : C.$$

Chaque validateur produit une preuve, la signe, puis rattache son étape aux précédentes. L'étape de B engage le résultat de A ; l'étape de C engage le résultat de $A + B$. Le round forme ainsi une séquence courte d'accords explicites et vérifiables.

Chaque nœud refait le même calcul depuis la seed : validateur attendu, étape du round, transactions et preuves associées. Un bloc qui annonce un validateur inattendu, engage une transaction invalide ou rompt l'ordre prévu est rejeté. La sécurité vient de ce consensus.

La finalité atteint sa portée publique lorsqu'un checkpoint inscrit la hauteur sur Ethereum. Elle devient alors lisible et utilisable par un contrat, un bridge, un règlement ou un audit.

9 Conclusion

Chain-of-Shards part d'un enjeu économique simple : toute transaction publiée sur Ethereum entre dans le marché du gas. Ce marché donne une sécurité et une vérifiabilité fortes, mais il devient coûteux lorsqu'il porte tout le volume applicatif. Chain-of-Shards déplace ce volume dans des chaînes privées rapides, où l'application paie principalement le compute et produit ses preuves à sa cadence.

La couche publique garde alors son rôle le plus précieux. Elle intervient lorsque le résultat doit sortir du domaine privé pour servir un bridge, un règlement, un audit, une sortie d'actifs ou un contrat sur Ethereum. Le checkpoint devient le moment de passage. Les gas fees accompagnent l'interopérabilité et les faits qui portent une valeur externe.

Cette séparation donne au modèle sa force : les transactions sont rapides dans le domaine privé, les preuves apportent une vérification cryptographique des transitions, la publication rend les hauteurs choisies interopérables. Chain-of-Shards permet ainsi de traiter un grand volume de transactions en quasi temps réel, tout en garantissant un haut niveau de décentralisation.

Références

[1] Louis Tremblay Thibault, Tom Sarry, and Abdelhakim Senhaji Hafid. Blockchain scaling using rollups : A comprehensive survey. *IEEE Access*, 10 :93039–93054, 2022.

[2] Stefanos Chaliasos, Denis Firsov, and Benjamin Livshits. Towards a formal foundation for blockchain ZK rollups. In *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security*, pages 2714–2728, 2025.

[3] Muhammad Bin Saif, Sara Migliorini, and Fausto Spoto. A survey on data availability in layer 2 blockchain rollups : Open challenges and future improvements. *Future Internet*, 16(9) :315, 2024.

[4] Jonathan Oleszkiewicz and Babu Pillai. Cross-chain interoperability : Local, proved, and checkpointed states, 2026. Manuscript.

[5] Jonathan Oleszkiewicz. Checkpointability : Separating proof availability from publication liveness, 2026. Manuscript.