

Kanban, Lean, Agile et Scrum : Évolution et concepts

Au début était un simple tableau des tâches de la semaine à effectuer, puis ce tableau Kanban s'est transformé en philosophie Lean complète, puis l'Agilité pour les logiciels et enfin Scrum pour tous les projets [1, 3, 5, 6]. Découvrez l'évolution historique des concepts Lean, Agile et Scrum, leurs différences, objectifs et bénéfices pour les équipes et projets.

Évolution historique

Lean, Agile et Scrum sont des termes clés dans la gestion de projets modernes, en particulier dans le développement logiciel. Ces concepts, bien qu'interconnectés, ont des origines et des applications distinctes. Cet article retrace leur évolution historique, du Kanban au Lean nés dans l'industrie automobile au Japon, à l'Agilité pour les logiciels, jusqu'à Scrum, un cadre adapté à divers projets.

Kanban : La visualisation des tâches

Dans les années 1940-1950, Kanban (terme japonais pour "carte" ou "panneau") émerge au sein du *Toyota Production System* (TPS). Utilisant des cartes physiques pour signaler les besoins en production, Kanban visualise les tâches et limite le travail en cours pour éviter le surstock [3]. Dans les années 2000, David J. Anderson adapte Kanban au développement logiciel, utilisant des tableaux pour optimiser les flux de travail [4].

Lean : Une philosophie d'optimisation

Également issu du TPS dans les années 1950, Lean se développe comme une philosophie de gestion visant à **minimiser les gaspillages** tout en maximisant la valeur client [1]. Lean identifie huit gaspillages : attente, défauts, transport, mouvement, surstock, surproduction, traitement inutile et talents inexploités. Des outils comme 5S et **Just-in-time** optimisent les processus. Eric Ries popularise Lean dans l'entrepreneuriat avec *Lean Startup* [2].

Agile : Flexibilité pour le logiciel

Dans les années 1990, face à des méthodes rigides, les développeurs créent le **Manifeste Agile** (2001) [5]. Il prône quatre valeurs :

- **Individus et interactions** plutôt que processus et outils.

- **Logiciel fonctionnel** plutôt que documentation exhaustive.
- **Collaboration avec le client** plutôt que négociation contractuelle.
- **Adaptation au changement** plutôt que suivi d'un plan rigide.

Agile favorise une approche itérative et incrémentale, avec des livraisons fréquentes de petites parties fonctionnelles. Initialement pour le logiciel, Agile s'étend à la gestion de projet et au marketing.

Scrum : Un cadre pour tous les projets

Dans les années 1990, Scrum émerge comme un **cadre Agile** structuré, formalisé dans les années 2000 [6]. Organisé en **sprints** (itérations courtes), Scrum définit des rôles clairs :

- **Product Owner** : représente les intérêts du client.
- **Scrum Master** : garantit l'application des pratiques Scrum.
- **Équipe de développement** : équipes multidisciplinaires autonomes.

Scrum utilise des outils comme le **backlog**, les **réunions quotidiennes** (Daily Scrum) et les **revues de sprint** pour une progression rapide et une amélioration continue.

Différences clés

- **Kanban** : Méthode visuelle de gestion des flux, limitant le travail en cours pour optimiser la productivité [3].
- **Lean** : Philosophie axée sur l'élimination des gaspillages et l'optimisation de la valeur, applicable à divers secteurs [1].
- **Agile** : Philosophie itérative centrée sur la flexibilité et la collaboration, conçue pour le logiciel mais adaptable [5].
- **Scrum** : Cadre structuré dans l'Agile, avec rôles, événements et artefacts pour gérer des projets complexes [6].

Kanban initie la visualisation des flux, Lean généralise l'optimisation, Agile adapte ces principes au logiciel, et Scrum structure leur application.

Bénéfices de Kanban, Lean, Agile et Scrum

L'adoption de ces concepts apporte :

- **Réduction des coûts** : Moins de défauts et meilleure gestion des risques [2].
- **Retour sur investissement rapide** : Livraisons fréquentes de fonctionnalités prêtes à l'emploi [5].
- **Productivité accrue** : Jusqu'à 2-3 fois supérieure grâce à l'élimination des gaspillages et à la gestion des flux [3].

- **Visibilité et contrôle** : Suivi précis via des tableaux Kanban ou des sprints Scrum [4, 6].
- **Flexibilité** : Adaptation continue aux changements [5].
- **Qualité supérieure** : Validation à chaque étape pour un produit robuste [1].
- **Amélioration continue** : Retours et ajustements via des rétrospectives ou analyses de flux [3, 6].

Autres cadres : SAFe et FAST

- **SAFe (Scaled Agile Framework)** : Cadre pour appliquer l'Agilité à grande échelle, adapté aux grandes organisations. Il structure la coordination entre plusieurs équipes Agile via des rôles, des processus et des cycles synchronisés comme les *Program Increments* [7].
- **FAST (Fluid Scaling Technology)** : Approche Agile légère pour les grandes entreprises, favorisant des cycles de livraison rapides et une coordination fluide sans la lourdeur de SAFe [8].

Conclusion

De Kanban, simple outil de visualisation, à Lean, philosophie d'optimisation, puis à l'Agilité pour le logiciel et enfin Scrum pour divers projets, ces approches forment une évolution cohérente. SAFe et FAST étendent ces principes aux grandes organisations. Comprendre leurs différences permet de choisir la meilleure stratégie pour maximiser la valeur client et l'efficacité des équipes.

Références

1. Womack, J. P., & Jones, D. T. (2003). *Lean Thinking: Banish Waste and Create Wealth in Your Corporation*. Simon & Schuster.
2. Ries, E. (2011). *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. Crown Business.
3. Ohno, T. (1988). *Toyota Production System: Beyond Large-Scale Production*. Productivity Press.
4. Anderson, D. J. (2010). *Kanban: Successful Evolutionary Change for Your Technology Business*. Blue Hole Press.
5. Beck, K., et al. (2001). *Manifesto for Agile Software Development*. Disponible sur : <http://agilemanifesto.org/>
6. Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Disponible sur : <https://scrumguides.org/>
7. Leffingwell, D., et al. (2021). *SAFe 5.0 Distilled: Achieving Business Agility with the Scaled Agile Framework*. Addison-Wesley.
8. FAST Agile. (2023). *Fluid Scaling Technology Overview*. Disponible sur : <https://fast-agile.com/>

Scrum : cadre, principe, règle, framework



Product Backlog
Global ToDo list
Made by PO

Sprint Backlog
Sprint Planning
ToDo NOW list
Selected by team

SPRINT (1 to 4 weeks)
Daily Scrum (10mn every day)

Artefact
Sprint Review
Completed features
Partial testable product
MVP - shippable parts

Product