

Le Cycle ADM : le moteur méthodologique

Architecture d'Entreprise

Le Cycle ADM : le moteur méthodologique

Architecture d'Entreprise

I — LE CAP

« TOGAF fournit [...] un 'toolkit' — il ne s'agit pas de mettre en œuvre 100 % des bonnes pratiques mais d'importer celles qui consolident les faiblesses de vos propres pratiques. » — Philippe Desfray et Gilbert Raymond, *TOGAF en pratique — Modèles pour l'architecture d'entreprise*, Dunod

« Pas d'excellence sans caractère. »

Vidéo d'introduction

 Télécharger cette leçon en PDF

L'architecture comme un tour de France

Imaginez qu'on vous confie la responsabilité de tracer un Tour de France. Pas une course improvisée au GPS, mais un Tour classique, étape par étape, avec ses cols, ses plaines, ses contre-la-montre, ses jours de repos. Les organisateurs du Tour le font depuis plus d'un siècle. Ils ont un plan, des étapes, des rôles. Chaque étape a un départ, une arrivée, des concurrents, un classement intermédiaire. Le Tour ne réussit pas parce qu'il improvise à chaque étape : il réussit parce que la structure générale existe avant la première pédalée.

L'Architecture Development Method (ADM) est exactement cela pour l'architecture d'entreprise. C'est un tour structuré, avec ses phases, ses livrables, ses points de contrôle. L'Open Group l'a défini formellement à partir des pratiques de ses membres — grands comptes internationaux qui ont partagé ce qui marchait et ce qui échouait. Ce n'est pas une théorie : c'est une méthode éprouvée par des centaines de cycles complets.

L'ADM n'est pas un *waterfall*. C'est un **cycle itératif**. Il y a des phases dans un ordre nominal, mais on peut revenir en arrière, décaler, exécuter en parallèle selon les besoins du contexte. Cette itérativité est sa force : elle permet d'absorber les changements, les surprises, les arbitrages politiques sans devoir tout recommencer. Elle est aussi sa complexité : mal comprise, elle devient une boucle sans fin où l'on n'arrive jamais à boucler un livrable.

L'ADM est composé d'une **Phase Préliminaire** (que nous avons traitée en L2) et de **huit phases principales** (de A à H), articulées autour d'une **gestion continue des exigences** qui irrigue l'ensemble. Chaque phase a des intrants, des activités, des livrables, des sorties. Chaque transition entre phases est un point de validation. Le cycle complet — Phase A → Phase H — dure en pratique entre **six mois et deux ans**, selon l'ampleur de la transformation et la maturité de l'organisation (TOGAF, *Introduction*, p. 6).

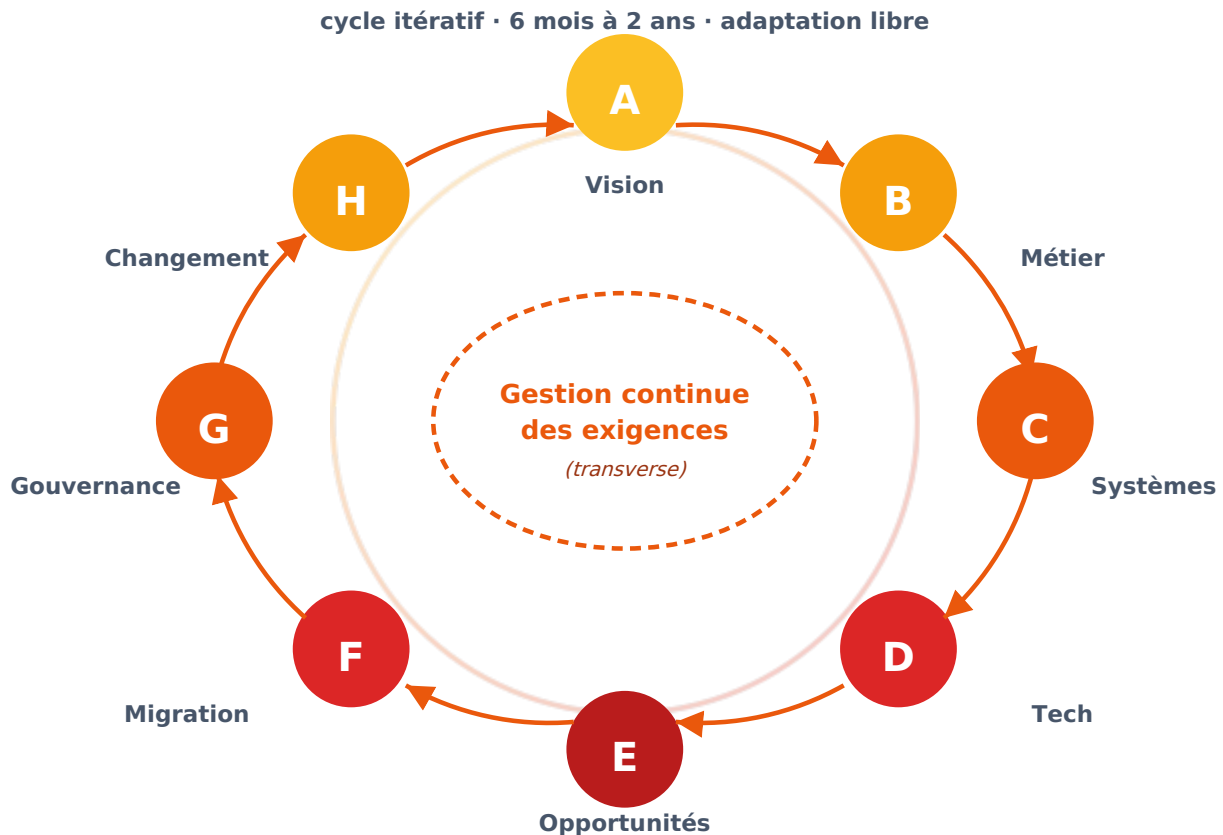
Le but de cette leçon n'est pas de mémoriser les noms des phases. Le but est de comprendre **pourquoi l'ADM fonctionne comme un cycle**, et **comment en tirer parti sans se laisser piéger par sa rigidité apparente**. Voici le cycle ADM complet :



Schéma

Le cycle ADM — Phases A à H + gestion continue

Itératif, pas séquentiel



Premier principe — L'ADM est un cycle, pas un waterfall

La confusion la plus coûteuse au sujet de l'ADM consiste à le lire comme un processus en cascade : on finit la phase A, on passe à B, puis C, etc. Ce n'est pas le cas. L'ADM est un cycle où chaque phase peut être ré-exécutée, raffinée, ajustée en fonction des découvertes faites en aval.

Le schéma canonique de l'ADM ressemble à un sablier ou à une roue : on entre par la Phase Préliminaire, on traverse A → B → C → D → E → F → G → H, et on revient en arrière pour un nouveau cycle. À chaque cycle, l'architecture devient plus précise, plus alignée, plus implémentable. Les exigences ne sont pas figées : elles sont gérées en continu, depuis la Phase A jusqu'à la Phase H, et continuent même au-delà.

Pourquoi cette structure cyclique ? Pour deux raisons que Desfray et Raymond soulignent dans *TOGAF en pratique*. Premièrement, l'architecture n'est pas un état final mais un **processus d'amélioration continue** : chaque cycle affine ce que le précédent a appris. Deuxièmement, les exigences métier évoluent pendant que l'architecture se construit : un nouveau directeur arrive, un concurrent sort un produit disruptif, une réglementation change. Un waterfall pur livrerait une architecture obsolète le jour de sa remise. Un cycle absorbe ces changements sans devoir tout reprendre.

L'itération n'est pas un désordre. Chaque cycle a un **périmètre** (un sujet précis qu'on traite dans le cycle courant) et un **degré d'engagement** (jusqu'à où on pousse les phases dans ce cycle). Un cycle peut couvrir toute l'organisation sur les phases A et B seulement (cadrage métier), puis un autre cycle peut approfondir la phase C sur un sous-domaine (architecture des données d'un processus spécifique). Les combinaisons sont infinies.

Et toi ? Quand tu mènes un projet d'architecture dans ton organisation, est-ce que tu boucles

explicitement un cycle complet avant d'enchaîner, ou est-ce que tu avances en rafales sans jamais fermer la boucle ? Quel est l'effet sur ta crédibilité quand le COMEX te demande « où en es-tu » ?

Deuxième principe — Les exigences sont au centre, pas en sortie

L'une des erreurs classiques consiste à traiter les exigences comme un livrable de début de projet — un document qu'on produit en Phase A et qu'on oublie ensuite. L'ADM pose au contraire que les exigences sont un objet vivant qui irrigue toutes les phases.

L'Open Group appelle cela la **gestion continue des exigences** (*Requirements Management*). Ce n'est pas une phase — c'est un processus transverse qui dure toute la durée du cycle. Chaque phase A → H produit ou raffine des exigences. Chaque revue de phase valide ou rejette des exigences. Chaque décision d'architecture est tracée jusqu'à une exigence explicite.

Pourquoi cette obsession de la traçabilité ? Parce qu'elle est ce qui distingue une architecture décisionnelle d'une architecture décorative. Une architecture décorative décrit un état futur idéal sans l'ancrer dans les exigences réelles du métier — elle reste sans défenseur quand le budget se serre. Une architecture décisionnelle relie chaque composant à une exigence nommée, à un sponsor identifié, à un indicateur mesurable. Quand le budget se serre, ce n'est pas l'architecture qui se contracte — c'est l'arbitrage qui devient possible : on coupe ce qui n'est adossé à aucune exigence vive, on protège ce qui est adossé à une exigence critique.

La contrefaçon à éviter est précisément l'inverse : une architecture où les exigences sont implicites, dispersées entre la tête des architectes et quelques slides éparses. C'est l'architecture qui flotte. C'est l'architecture qui meurt à la première vague.

***Et toi ?** Pour ton dernier livrable d'architecture, combien d'exigences sont explicitement tracées dans un document validé ? Et pour chaque composant du modèle cible, peux-tu remonter à l'exigence métier qui l'a justifié ?*

Troisième principe — Chaque phase a un livrable nommé, pas une étape “plus tard”

Le deuxième piège classique consiste à traiter les phases comme des étapes administratives dont les livrables peuvent être remis “plus tard”. L'ADM exige au contraire qu'à chaque phase corresponde un livrable nommé, validé, archivé.

Le schéma TOGAF liste pour chaque phase A → H les intrants (*Inputs*), les activités (*Steps*), les livrables (*Deliverables*), et les sorties (*Outputs*). Cette nomenclature n'est pas un détail bureaucratique. C'est ce qui rend possible la traçabilité, la gouvernance, et la capitalisation entre cycles. Sans livrables nommés, chaque cycle est une expérience à chaque fois nouvelle — et la pratique AE ne progresse pas.

L'illustration concrète est la Phase A (Vision). Le livrable central est le *Statement of Architecture Work* — un document d'une dizaine de pages qui pose le périmètre, les parties prenantes, les critères de succès, les contraintes, les hypothèses. Sans ce document, la Phase B (Architecture Métier) démarre dans le flou. Sans Phase B documentée, la Phase C (Architecture des Données) démarre dans le flou. Sans Phase C documentée, la Phase D (Architecture Applicative) démarre dans le flou. Et ainsi de suite, jusqu'à ce que l'équipe ne sache plus pourquoi elle produit ce qu'elle produit.

La parade n'est pas de produire tous les livrables en avance — c'est de les produire **au moment où la phase se**

termine, dans leur version minimale viable, validés par l'instance de gouvernance. Un *Statement of Architecture Work* de trois pages qui existe vaut mieux qu'un document de vingt pages qui n'existe pas.

Et toi ? Pour ton dernier cycle d'architecture, combien de phases ont produit leur livrable nommé en temps voulu ? Et combien ont glissé vers "on l'a fait à l'oral" ?

Quatrième principe — Le *tailoring* est une compétence, pas un écart

Le tailoring — l'adaptation de l'ADM au contexte — est la compétence la plus sous-estimée de la pratique AE. Bien appliqué, il distingue les architectes matures des techniciens qui récitent TOGAF comme un catéchisme.

L'ADM propose un cadre nominal. Mais chaque organisation a ses contraintes : taille, culture, cycle budgétaire, maturité numérique, profil de risque. Un cycle ADM qui s'exécute tel quel, sans adaptation, produit presque toujours un livrable inadapté au contexte. Un *tailoring* qui néglige les bonnes pratiques du cadre produit un livrable qui ne ressemble plus à de l'architecture d'entreprise.

La compétence de *tailoring* consiste à **identifier ce qui, dans le cadre, est essentiel à la discipline AE, et ce qui peut être adapté sans perte de substance**. Les phases A → H sont essentielles — elles structurent le cycle. La gestion des exigences est essentielle — elle garantit la traçabilité. Mais le nombre exact d'itérations par phase, le périmètre des livrables, le formalisme des revues, la cadence des comités : tout cela se négocie avec le contexte.

Desfray et Raymond résument la chose ainsi : « *TOGAF fournit un 'toolkit' — il ne s'agit pas de mettre en œuvre 100 % des bonnes pratiques mais d'importer celles qui consolident les faiblesses de vos propres pratiques* ». C'est la définition même du *tailoring* : un choix actif de ce qui consolide l'existant, pas une exécution servile du cadre.

La contrefaçon à éviter : le *tailoring* par ignorance — celui qui consiste à sauter des phases parce qu'on ne sait pas les exécuter, en prétendant que "c'est du *tailoring*". Ce n'est pas du *tailoring*. C'est de l'amateurisme qui se pare des atours de la méthode.

Et toi ? Sur ton dernier cycle ADM, quels éléments as-tu adaptés au contexte ? Et quels éléments as-tu sauté "parce qu'on n'avait pas le temps" ? La différence entre les deux est ta maturité de praticien.

Cinquième principe — Un cycle complet prend du temps, et c'est précisément ce qui le rend robuste

Le dernier piège à déjouer est celui de l'impatience : croire qu'on peut mener un cycle ADM complet en trois mois, parce que "le marché n'attend pas". C'est confondre vitesse et précipitation.

L'Open Group évalue la durée réaliste d'un cycle ADM complet entre **six mois et deux ans** (TOGAF, *Introduction*, p. 6). Ce n'est pas une fourchette théorique. C'est la médiane observée dans les organisations qui mènent des cycles aboutis. Pourquoi ? Parce qu'un cycle complet implique de traverser la Phase Préliminaire (cadrage), les phases A à H (production d'architectures As-Is, cible, de transition), les phases F et G (planification et gouvernance de l'implémentation), sans compter les itérations et les validations de gouvernance.

L'impatience produit deux types de dégâts. Premièrement, les **cycles avortés** — on déclare un cycle "complété" alors que seules les phases A et B ont vraiment été menées ; les phases C à H sont restées à l'état d'ébauche. Le résultat : une architecture cible sans implémentation, ou une implémentation sans architecture de transition. Deuxièmement, les **cycles compressés** — on tente de mener les phases en parallèle pour gagner du temps, ce qui produit des livrables

bâclés et des incohérences entre couches.

La parade n'est pas la lenteur par principe. C'est la **planification réaliste dès la Phase Préliminaire** : annoncer au sponsor que le cycle prendra X mois, négocier les jalons en conséquence, protéger les fenêtres de production contre les interruptions opérationnelles. C'est cette discipline de cadrage temporel qui distingue les architectes matures des pompiers permanents.

***Et toi ?** Sur ton dernier cycle annoncé à ton sponsor, quelle durée as-tu promise ? Et quelle durée a-t-il réellement fallu ? L'écart entre les deux est ton indicateur le plus fiable de crédibilité.*

Application — Ma roadmap ADM 90 jours

À rédiger séance à séance. Trois livrables précis, datés, à afficher et à confier à deux personnes de confiance.

1. **Mon périmètre de cycle** — Le sujet précis que je traite dans mon prochain cycle ADM. Ni trop large (toute l'organisation), ni trop étroit (un seul composant). Un périmètre qui peut être cadré en 90 jours.
2. **Mon cadrage temporel** — Pour chaque phase que je mène réellement (pas toutes les phases nominales), une durée estimée et une date butoir. Justification du *tailoring* appliqué.
3. **Mes trois exigences critiques** — Les trois exigences métier qui doivent absolument être couvertes par ce cycle. Avec leur sponsor nommé et leur indicateur de succès.

Ce document de trois pages est ton mon livrable de Phase A. C'est lui que tu présenteras au sponsor avant de commencer.

Encadré Au lecteur / À l'animateur

Question d'ouverture (à poser à voix haute avant la lecture) : « Combien de cycles ADM as-tu menés jusqu'à maintenant dans ton organisation ? Combien en as-tu bouclés ? »

Profils à observer chez l'apprenant :

- **Le récitant qui sait l'ADM par cœur mais ne l'a jamais pratiqué.** Énumère les phases, cite les livrables, n'a jamais vu un *Statement of Architecture Work* validé. *Question* : « Quel est ton dernier livrable de Phase A ? »
- **L' impatient qui confond vitesse et précipitation.** Promet des cycles de trois mois, livre des architectures sans implémentation. *Question* : « Combien de temps a duré ton dernier cycle complet ? »
- **Le rigoriste qui applique TOGAF sans *tailoring*.** Livre des livrables inadaptés au contexte parce qu'il a voulu suivre le cadre à la lettre. *Question* : « Quels éléments du cadre as-tu adaptés au contexte ? »
- **Le démineur qui improvise chaque cycle comme une urgence nouvelle.** N'a aucune capitalisation, refait les mêmes erreurs à chaque cycle. *Question* : « Quelle trace laisses-tu après chaque cycle ? »

Pièges d'animation :

- Ne pas laisser la discussion dériver vers la mécanique des phases (A, B, C...). Rappeler que la leçon porte sur **les principes qui sous-tendent le cycle**, pas sur le détail opérationnel.
- Ne pas confondre *tailoring* et paresse. Un *tailoring* se justifie, se documente, se défend. Une improvisation se constate, se subit, se regrette.

Si profil d'apprenant ≠ praticien AE : ouvrir sur l'équivalent dans le rôle de l'apprenant. Tout projet structurant

traverse des phases analogues — cadrage, conception, livraison, transition, gouvernance. Adapter les exemples.

Ce que la leçon 4 vous demande

Ouvrez votre dossier d'architecte à la page suivante. Préparez :

- **Un périmètre précis** pour votre prochain cycle d'architecture ou de projet structurant.
- **Une durée estimée, jalonnée** par les phases que vous mènerez réellement.
- **Trois Exigences critiques** avec leur sponsor nommé et leur indicateur de succès.

Sources de cette leçon : TOGAF Standard 10^e (ADM, ADM Techniques) · Desfray & Raymond, *TOGAF en pratique*

Leçon suivante : **L4 — La Vision (Phase A) : éviter les choix hâtifs.**

Quiz — 18 questions pour valider votre compréhension

Trois séries de six questions. Prenez un crayon, écrivez vos réponses dans l'espace prévu. Score indicatif : 14/18 minimum pour valider la leçon.

A — COMPREHENSION

A1 Qu'est-ce que l'ADM ? Quelles sont les huit phases principales ? Quel est le rôle de la gestion continue des exigences ?

A2 Pourquoi l'ADM est-il un cycle et non un waterfall ? Quel est l'intérêt de cette cyclicité ?

A3 Qu'est-ce que la « gestion continue des exigences » ? Pourquoi est-elle transverse plutôt qu'un livrable de phase ?

A4 Qu'est-ce que le « tailoring » dans TOGAF ? Quels sont les niveaux d'adaptation possibles ?

A5 Pourquoi le cycle ADM dure-t-il en pratique entre six mois et deux ans ? Quels facteurs font varier cette durée ?

A6 Quels sont les livrables canoniques de chaque phase ADM (A, B, C, D, E, F, G, H) ?

B — AUTO-EVALUATION

B1 Combien de phases d'un cycle ADM as-tu réellement traversées sur ton dernier projet ? Lesquelles ont été escamotées ?

B2 As-tu déjà pratiqué la gestion continue des exigences ? Combien d'exigences as-tu tracées sur ton dernier projet ?

B3 As-tu pratiqué le tailoring TOGAF, ou as-tu appliqué le cadre tel quel ? Quel a été l'effet ?

B4 Sur tes cinq derniers projets, combien ont duré plus de 12 mois ? Combien ont été interrompus avant la fin du cycle ?

B5 Combien de livrables TOGAF canoniques as-tu produits sur ton dernier cycle ? Lesquels as-tu improvisés ?

B6 Sur ton dernier cycle ADM, combien de tours complets as-tu réalisés avant de considérer l'architecture cible stabilisée ?

C — ENGAGEMENTS A 30/45/60 JOURS

C1 Identifier le livrable canonique de chaque phase (A → H) de mon prochain cycle, et l'adapter à mon contexte. (Échéance : J+7.)

C2 Définir le rythme de mon cycle : revue trimestrielle, gouvernance mensuelle, ajustement annuel. (Échéance : J+14.)

C3 Identifier 3 exigences critiques pour ce cycle, avec leur sponsor nommé et leur indicateur de succès. (Échéance : J+21.)

C4 Présenter le livrable Phase A (3 pages) au sponsor et obtenir son accord avant de démarrer. (Échéance : J+30.)

C5 Documenter explicitement le *tailoring* appliqué (1 page) — ce qui est conservé tel quel, ce qui est adapté, ce qui est écarté. (Échéance : J+14.)

C6 Premier livrable de phase produit et validé par l'instance de gouvernance dans les 30 jours. (Échéance : J+30.)
