

90. Cycle de vie du développement de systèmes d'information informatisés (SII)

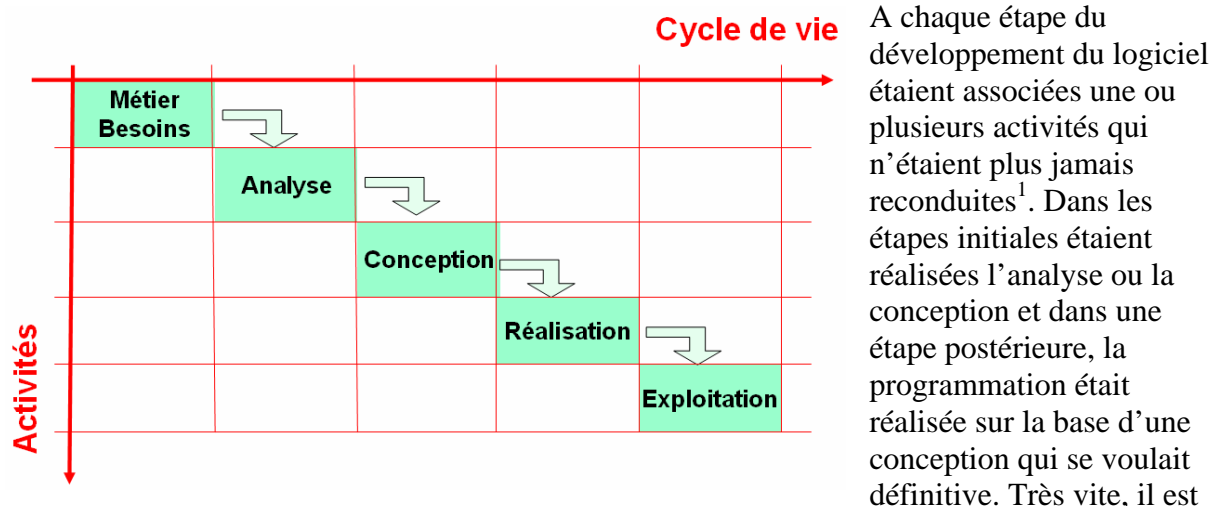
1 Préambule

Le cycle de vie permet de passer de l'idée d'un logiciel à son utilisation régulière et ensuite à son obsolescence en parcourant des étapes ou phases de développement. Ces étapes ou phases peuvent être comparées à celle par lesquelles passe l'être humain : la naissance, l'enfance, l'adolescence, l'âge adulte, la vieillesse et la mort pour finir.

Les méthodes n'étant pas normalisées, le découpage du cycle de vie est souvent propre à chaque méthode et le nom de ces portions du cycle de vie sont multiples ; nous trouvons couramment les termes d'étapes ou de phases.

Les méthodes classiques, telles que Merise, utilisent le terme d'étape ; les méthodes orientées objet, de part l'unification d'UP, ont adopté le terme de phase.

Historiquement, chaque étape ou phase était parcourue une et une seule fois à l'image de l'humain qui passe à l'enfance à l'âge adulte en passant par l'adolescence sans retour possible.



apparu que la linéarité d'une telle démarche ne permettait pas le principe de rétroaction chère à la systémique ; en l'occurrence, ce n'était qu'à la programmation ou à livraison du produit final que des défauts d'analyse ou de conception apparaissaient. Les incidences financières et autres de la découverte très tardive d'erreurs d'analyse ou de conception ont conduit à la perte de nombreux projets, services informatiques ou autres prestataires de service.

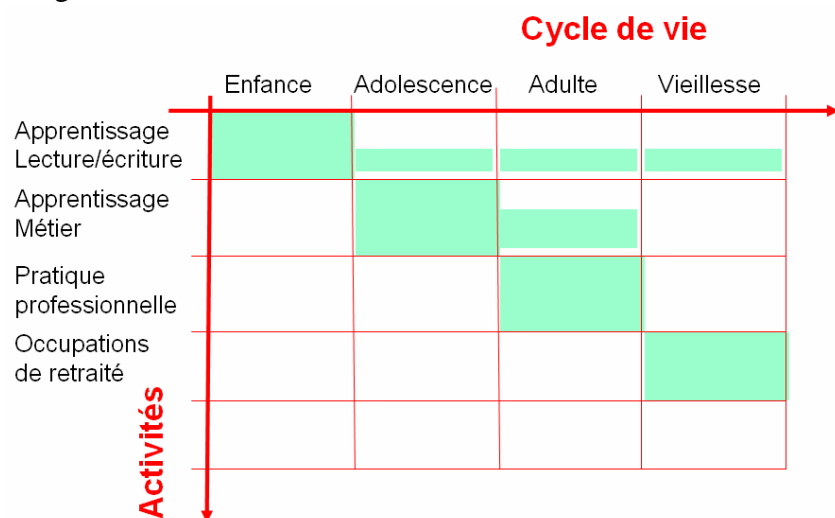
¹ Une boucle de rétroaction chère à la systémique était possible depuis l'étape subséquente. En fait chaque étape valide l'étape qui la précède immédiatement.

Pour corriger les travers de la démarche linéaire sont apparus des modèles dits en spirales où les risques, quels qu'ils soient, sont constamment traités au travers de bouclages successifs ; chaque spire confirme et affine les spires précédentes en menant des activités de même nature successivement. L'analyse ou la conception ne sont plus effectuées dans une seule phase ou étape mais sont conduites en tant qu'activités qui se déroulent sur de multiples phases.

Progressivement le cycle de développement ne s'est plus décliné en étapes ou phases assimilées à des activités mais en phases de maturité du logiciel dans lesquelles, à la limite, chaque activité est conduite si nécessaire. Par exemple, dans les phases initiales il y aura une importante activité d'analyse mais néanmoins un peu de programmation pour confirmer les choix architecturaux d'analyse ; et dans les phases finales, il y aura surtout de la programmation mais néanmoins aussi de l'analyse pour les besoins qui n'avaient pas été estimés prioritaires.

Pour illustrer notre propos et en revenant à la métaphore de l'humain du début, nous pouvons considérer que l'apprentissage de la lecture se fait à l'école primaire pendant l'enfance ; ainsi la phase « enfance » comporte l'activité « apprentissage de la lecture » et cet apprentissage n'est plus jamais reconduit.

Dans la vision d'une démarche de développement linéaire, l'individu qui ne sait pas lire à l'adolescence restera analphabète toute sa vie ; dès lors, cet individu a peu de chance de pouvoir suivre une formation professionnelle et le risque est très grand de le voir se marginaliser.



Dans une vision de développement itérative, la société prend conscience des risques induits par l'analphabétisme ; dès lors, et pour autant que cela soit nécessaire, l'activité « apprentissage de la lecture » n'est plus restreinte à la seule enfance mais à toutes les autres phases du développement de l'individu y compris la

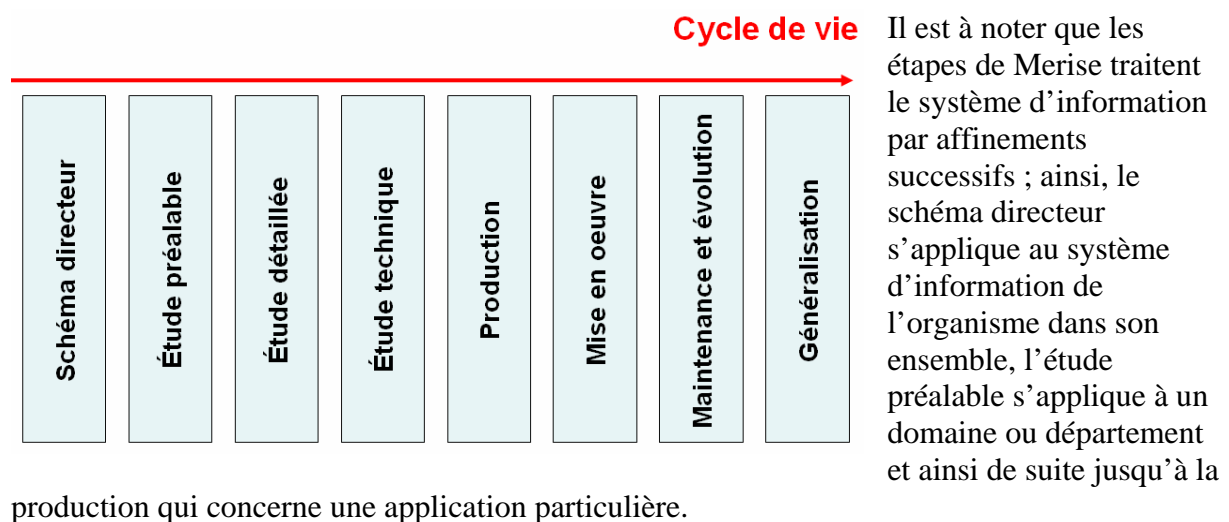
vieillesse. Un vieillard qui ne sait ni lire, ni écrire risque de se refermer sur lui-même ; l'apprentissage de la lecture et de l'écriture lui donnera des moyens supplémentaires de communiquer, particulièrement utiles, lorsque sa mobilité se trouve affectée par la vieillesse. Il en est de même pour la formation professionnelle, si elle est prioritairement le fait de l'adolescence, elle doit pouvoir être reconduite à l'âge adulte ; c'est tout l'aspect très actuel du perfectionnement ou de la reconversion professionnelle.

Le cycle de vie du développement de SII est aussi appelé cycle chronologique de développement de SII par certains auteurs comme [SJBB-00]. Pour notre part, nous utilisons le terme de cycle de vie en référence à la méthode Merise qui est une des premières à avoir proposé le découpage du développement en 3 cycles : cycle de vie, cycle d'abstraction et cycle de décision.

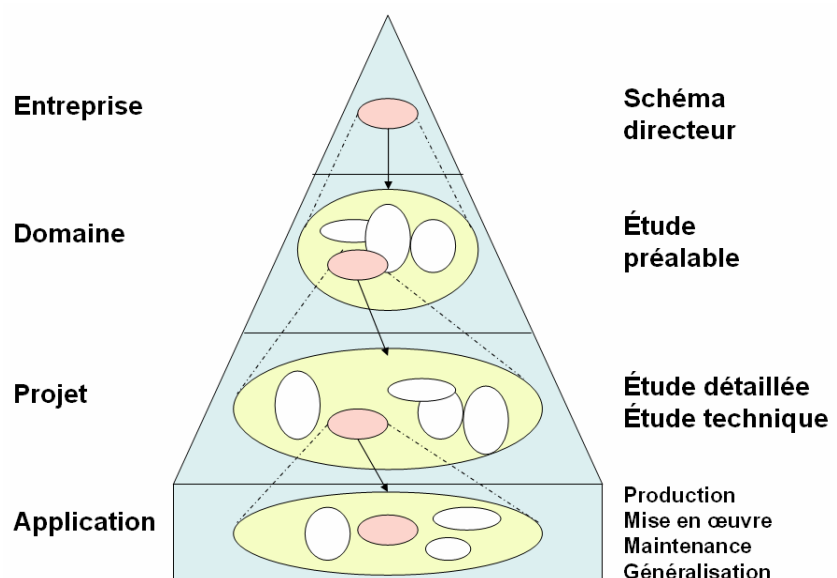
2 Cycle de vie en cascade (Merise)

Le cycle de vie dit de la « cascade » date de 1970, il est l'œuvre de Royce. Il s'agit d'un modèle linéaire. Toute étape est supposée n'avoir de rétroaction que sur l'étape qui la précède. L'activité d'une étape se réalise avec les résultats fournis par l'étape précédente ; ainsi, chaque étape sert de contrôle du travail effectué lors de l'étape précédente. L'utilisateur attend le déroulement complet du cycle de vie du logiciel pour vérifier, lors de la dernière étape, l'adéquation entre ses exigences et le produit livré.

A titre d'exemple, nous reproduisons ci-après, les étapes du cycle de vie² présentées dans l'ouvrage « La méthode Merise Tome 3 – Gamme opératoire » de A. Rochfeld et J. Moréjon publié en 1989. [MERISE-3].



Merise symbolise cette gradation des étapes par une pyramide ; au sommet de la pyramide se trouve le plus général c'est-à-dire l'entreprise et le schéma directeur et à la base de la pyramide se trouve le plus spécialisé, en l'occurrence, l'exploitation d'une application particulière.

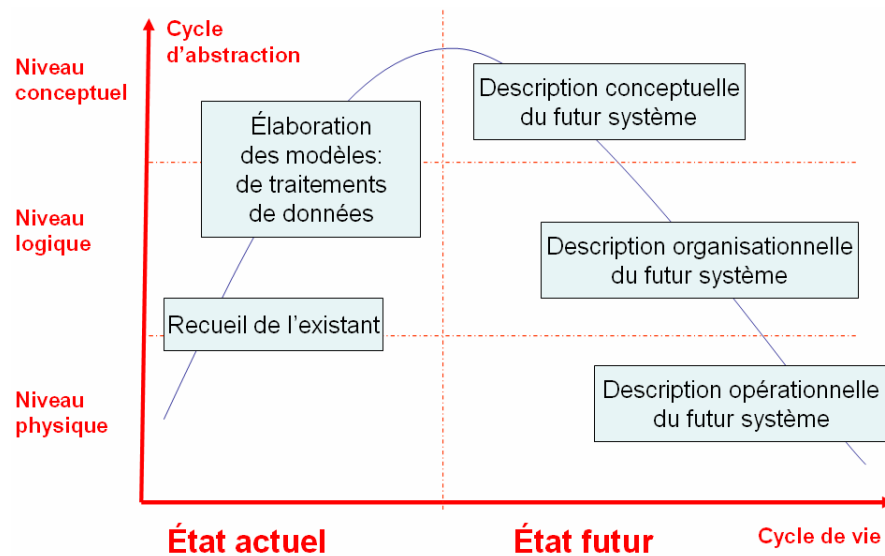


² Le cycle de vie tel que l'ouvrage le décrit et le présente graphiquement était déjà défini lors de l'élaboration de la méthode en 1978/1979

En 1978, Merise introduisait déjà les prémisses de différentiation entre découpage temporel du cycle de vie et activités menées à chaque étape; cette différentiation a été illustrée par un schéma devenu célèbre : « la courbe du soleil ».

Sur l'axe horizontal se trouve le cycle de vie sous forme d'état du système d'information et sur l'axe vertical, le cycle d'abstraction, se trouve la représentation du système.

Dans cet espace à 2 dimensions sont représentés les différentes activités³, respectivement modèles, à réaliser.



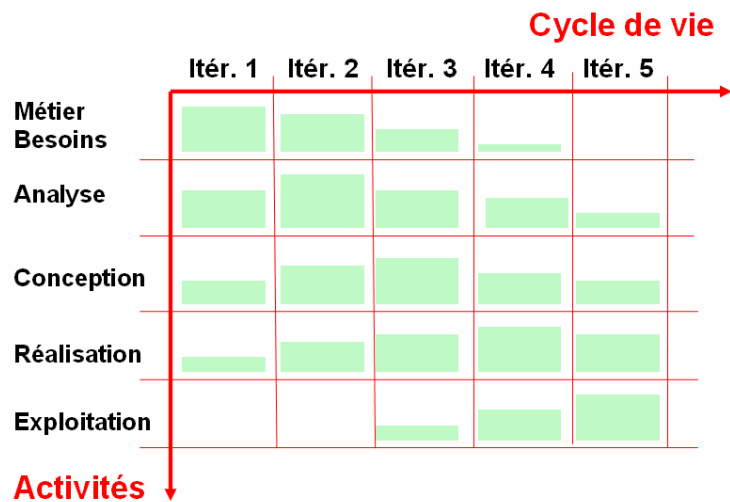
Appliquée à mauvais escient, la rigidité du modèle en cascade de Merise, a mené de nombreux projets à l'échec ; et, cette « courbe du soleil » a été caricaturée par certains comme le lancement du projet au lever du jour et son aboutissement au crépuscule donc bien après que l'on en eut l'utilité. En effet, appliquée sans

discernement, le principe de la courbe du soleil comporte le risque de passer trop de temps à décrire l'état actuel au détriment de la réalisation du nouveau système ; donc de fournir une solution désuète et en retard.

³ Uniquement par souci de rigueur mais néanmoins avec le souci de ne pas perturber le lecteur, nous nous devons de mentionner que Merise qualifiait ces activités de « phases » ; dans la terminologie originale de Merise, nous avons donc le couple :

- étapes pour le découpage temporel ;
- phases pour les activités menées dans les différentes étapes.

3 Cycle de vie itératif

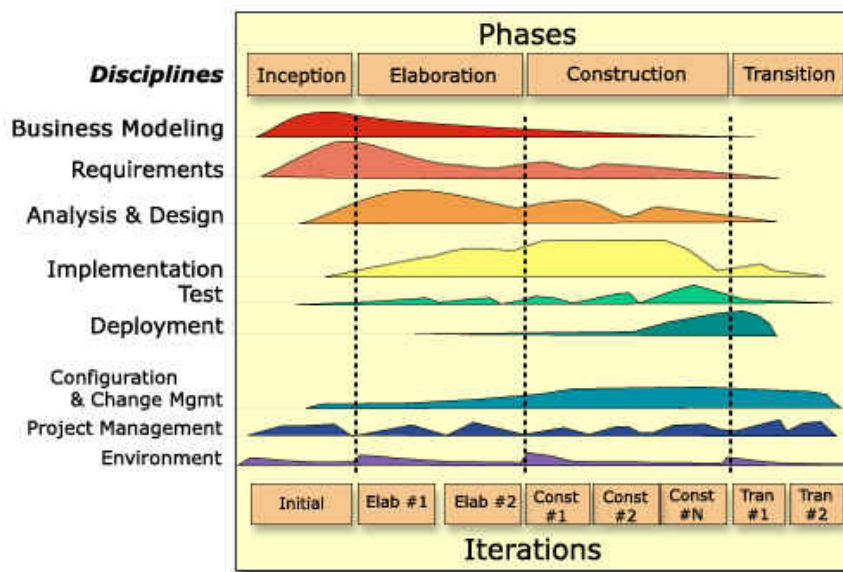


Le cycle de vie itératif consiste à mener la plupart des activités de production du logiciel successivement dans chacune des phases du cycle de vie; la ou les premières itérations se concentreront sur la compréhension du métier et des besoins mais, incluront néanmoins aussi de l'analyse et de la conception et dans une moindre mesure une réalisation de quelques prototypes qui seront mis en œuvre pour s'assurer de la justesse des

choix architecturaux. Le choix des premiers éléments du logiciel à réaliser doit être dicté par les risques les plus critiques encourus par le projet ; la réalisation de ces premiers éléments doit donner l'assurance que les risques identifiés sont maîtrisés ou maîtrisables.

Pour garantir la maîtrise des risques, le cycle de vie en cascade s'accompagne de la réalisation d'artefacts de validation des choix architecturaux ; ainsi, en plus des différents modèles graphiques, sont réalisés des maquettes et prototypes soumis pour validation aux utilisateurs. De même lorsque le cadre du projet le permet, la mise en exploitation du logiciel est effectuée progressivement, au fur et à mesure de l'avancement des itérations ; ainsi, les utilisateurs peuvent quittancer le travail réalisé une partie du travail avant la livraison finale.

Les différentes itérations peuvent être nommées en tant que phases ou faire partie de phases englobantes. A titre d'illustration, nous reproduisons deux diagrammes « Phases & activités », le premier issu de la méthode RUP⁴ et le deuxième issu de la méthode CDM⁵.



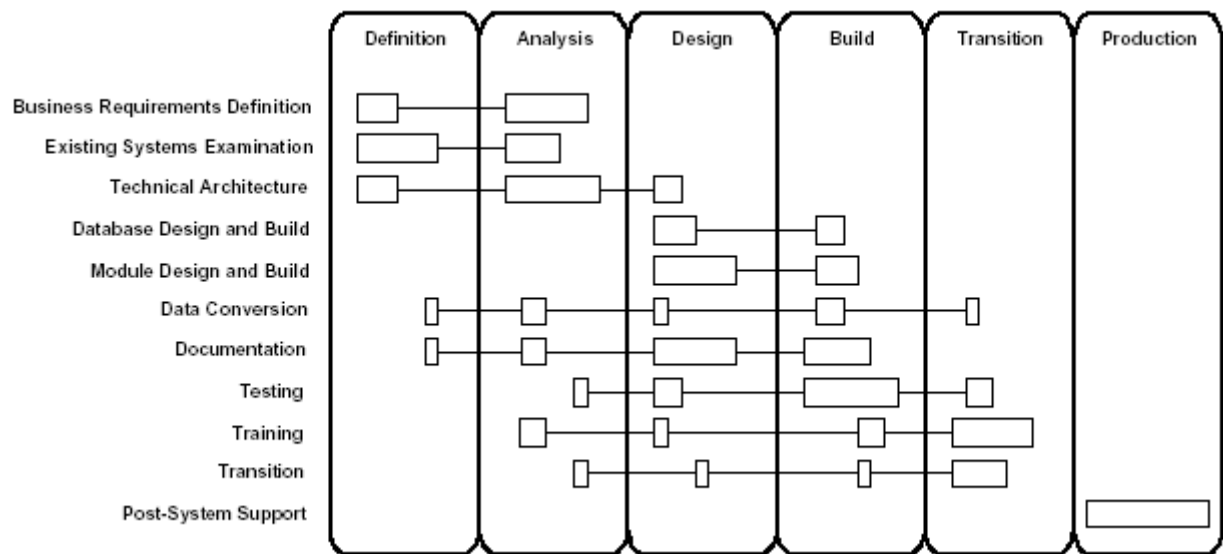
Il est à noter que le Processus unifié et RUP utilisent le terme de discipline en lieu et place d'activité ; ils réservent le terme « activité » pour décrire le travail à réaliser à l'intérieur de phases et disciplines.

Phases et disciplines du RUP

⁴ RUP : Déclinaison commerciale d'IBM du Processus unifié (UP)

⁵ CDM : Méthode de développement de logiciel du constructeur Oracle

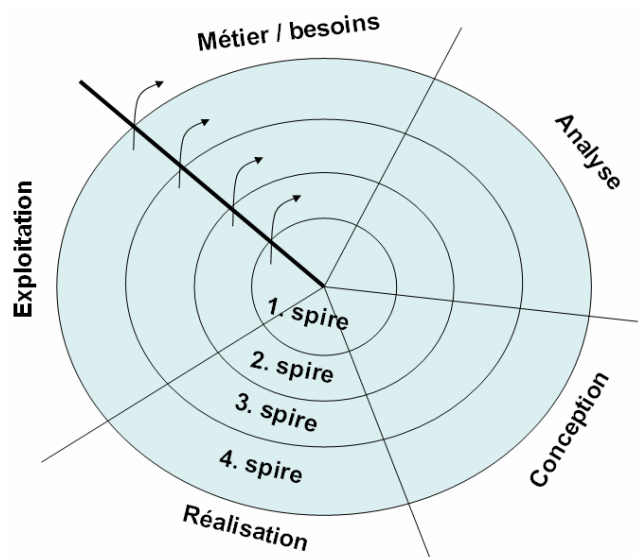
De son côté, CDM utilise le terme de « processus » en lieu et place d'activité ; un « processus » de CDM est un enchaînement de « tâches » réalisées sur plusieurs phases.



Phases, processus et tâches de CDM-Classic

Remarque : Formellement, UP, RUP et CDM préconisent une mise à disposition du logiciel opérationnel aux utilisateurs seulement lors de la phase terminale. Ainsi, pour de nombreuses méthodes, les itérations ne concernent que les phases précédant la mise en exploitation ; les itérations servent essentiellement à maîtriser les risques inhérents à la construction du logiciel et non aux risques induits par d'éventuels changements organisationnels liés à son utilisation.

4 Cycle de vie en spirale



Le cycle de vie en spirale est une démarche hautement itérative. La construction du SII se réalise par spires de fonctionnalités successives. Les premières couches, comme les premières itérations du cycle itératif, servent à réduire les risques identifiés, et simultanément à fournir les prototypes et/ou solutions opérationnelles offrant aux utilisateurs les services minimaux attendus. Les couches suivantes doivent permettre d'élargir les services offerts pour atteindre progressivement la couverture complète des besoins tout en assurant la transition avec les éléments réalisés antérieurement.

Chaque boucle de la spirale peut être menée comme un mini projet s'appuyant sur un cycle de vie itératif avec ses phases et activités⁶ comme nous l'avons vu précédemment pour le RUP.

Selon la nature du projet, une planification relativement poussée est réalisée en amont de la mise en œuvre de la première boucle; ensuite, seul un affinement de planification est réalisée avant chaque boucle. L'affinement de planification a trait essentiellement à la prise en compte des enseignements retirés des boucles précédentes ; que ce soit au niveau de la gestion des risques ou de la satisfaction des utilisateurs.

A l'inverse, pour un projet exploratoire qui n'a pas de caractère stratégique, la planification peut se faire uniquement en amont de chaque boucle en fonction des retours des boucles antérieures.

Un cycle de vie en spirale n'est pas la panacée universelle ; il comporte ses risques. Nous pensons, en priorité, au risque de faire des choix architecturaux dans les premières spires qui s'avèrent inappropriés pour supporter les exigences à satisfaire dans les spires à venir. En fait le cycle en spirale, quand il se traduit par une mise en exploitation du logiciel à chaque spire, est une forme de pari sur l'avenir.

5 Conclusion

Entre un cycle purement en cascade et un cycle en spirale, de multiples manières de faire sont implantées par les différentes méthodes.

Très certainement que dans notre monde économique où la capacité d'adaptation des organismes et de leur système d'information aux changements devient primordiale, le modèle en cascade n'est plus envisageable. Les modèles itératifs ou en spirales sont plus appropriés à prendre en compte les incidences de changements subis par tout projet entre son initialisation et sa finalisation.

Remarque :

Pour tout organisme très hiérarchisé et qui est peu sujet aux changements environnementaux, les bases de Merise sont certainement toujours pertinentes.

⁶ Discipline dans la terminologie UP ou RUP