

LES FONDEMENTS DE L'INGÉNIERIE DES ENVIRONNEMENTS NUMÉRIQUES D'APPRENTISSAGE

Gilbert Paquette

RÉSUMÉ

L'ingénierie des environnements numériques d'apprentissage (IENA) est une méthodologie d'ingénierie pédagogique visant la réalisation des environnements numériques d'apprentissage (ENA). Elle peut se concrétiser par différentes méthodes spécifiques¹ et se situe à l'intersection de trois domaines du savoir : le design pédagogique scientifique, l'ingénierie des systèmes d'information et l'ingénierie des connaissances. Par rapport aux théories sur l'apprentissage et l'enseignement, le design pédagogique peut être vu comme une forme d'ingénierie ayant pour but d'améliorer les pratiques éducatives. L'ingénierie des systèmes d'information examine pour sa part les ENA comme des systèmes particuliers de traitement de l'information. Quant à l'ingénierie des connaissances, elle vise la modélisation des connaissances qui constituent l'objet même de l'apprentissage. Ce chapitre définit ces trois domaines méthodologiques et présente leurs contributions à la définition d'une méthode d'ingénierie des ENA. En conclusion, on évoque la problématique que soulève l'implantation des pratiques de l'IENA dans les systèmes éducatifs.


MOTS-CLÉS

Design pédagogique; génie logiciel; génie cognitif; méthode systématique; science des systèmes.

1 Une méthodologie est vue ici comme une classe de méthodes, une sorte de boîte à outils où chaque outil est une méthode et où l'ensemble des outils appartient à la même catégorie.

Certaines méthodes d'ingénierie pédagogique sont présentées brièvement au chapitre 2 et le chapitre 4 est consacré à un exemple détaillé d'une méthode d'IENA, soit la méthode d'ingénierie des systèmes d'apprentissage (MISA).

Introduction

 chapitre présente un survol général des fondements méthodologiques de l'ingénierie des environnements numériques d'apprentissage (IENA).

L'évolution rapide des moyens d'apprentissage sur Internet met en évidence la multiplicité des décisions à prendre au moment de la conception d'un environnement numérique d'apprentissage (ENA). Sur le plan pédagogique, les possibilités extrêmement variées rendues possibles par les technologies de l'information et de la communication constituent un premier argument pour une méthodologie adaptée à la conception des ENA.

Un deuxième argument justifie le recours à une telle méthodologie. Un ENA est un ensemble de logiciels plus ou moins interreliés, certains conçus spécifiquement pour l'apprentissage d'un domaine donné, d'autres réutilisant des logiciels à usage général, donnant accès à des documents numérisés et à des services de communication. Le développement de tels systèmes accessibles sur Internet ne peut plus être traité comme une œuvre artisanale, en marge des méthodes du génie logiciel et des sciences de l'information.

Enfin, troisième argument, il faut aussi mettre en évidence l'importance accrue de la gestion des connaissances. De plus en plus, on reconnaît les connaissances et l'expertise comme le capital le plus précieux d'un individu ou d'une organisation. Cette préoccupation a présidé, au cours des années 1970 et 1980, à l'essor des systèmes experts. Depuis, les techniques et les outils informatiques ont évolué, les systèmes à base de connaissances s'intégrant dans divers dispositifs techniques. Le Web, qui accueille maintenant la plupart des nouvelles formations, est devenu social et sémantique, facilitant les interactions des apprenants et des apprenantes dans leur traitement des connaissances et des compétences. Les méthodes d'extraction, de formalisation et de traitement des connaissances, autrement dit l'« ingénierie des connaissances » ou « génie cognitif », sont, par voie de conséquence, au cœur de la conception des environnements d'apprentissage dont le but est l'acquisition de connaissances et de compétences.

La première section de ce chapitre traite de la notion de « méthodologie systémique » qui caractérise cette méthodologie particulière qu'est l'IENA. Celle-ci hérite principalement des processus et des principes de trois domaines méthodologiques particuliers décrits dans les trois sections suivantes : le design pédagogique scientifique (section 2), l'ingénierie des systèmes d'information (section 3) et l'ingénierie des connaissances (section 4). La conclusion aborde brièvement la problématique d'implantation d'une méthode d'IENA.

1 Les méthodes systémiques

Située à l'intersection du design pédagogique, de l'ingénierie des systèmes d'information (ou génie logiciel) et de l'ingénierie des connaissances (ou génie cognitif) dont elle hérite ses principales propriétés (figure 3.1), l'IENA se présente comme une méthodologie systémique particulière, vouée à la résolution des problèmes de conception, de réalisation et de diffusion de systèmes, de dispositifs ou d'environnements numériques d'apprentissage, ceux-ci faisant majoritairement appel aux technologies numériques. Nous en décrivons dans cette section les particularités.

1.1 L'IENA comme méthodologie systémique

L'IENA se fonde sur la science des systèmes (Le Moigne, 1995; Simon, 1981) qui définit la notion de système comme un ensemble d'éléments en interaction dynamique, organisés en fonction d'un but.

Comme cela a été souligné au chapitre 2, cette définition très large réfute l'accusation injustifiée que l'on fait parfois aux méthodes systémiques pour leur caractère soi-disant « mécaniste » qui nierait la possibilité d'environnements d'apprentissage ouverts misant sur l'autonomie et l'agentivité des apprenants et des apprenantes. S'il est vrai que les premiers systèmes d'apprentissage étaient fondés sur des approches d'enseignement programmé transmissif (Skinner, 1954), ce n'est plus le cas dans les ENA ouverts qui misent sur l'autonomie des participants et des participantes. Dans ces environnements, les éléments en interaction dynamique peuvent être une communauté d'acteurs et de ressources, spécifiés *a priori*, ou se constituant en tout ou en partie de façon dynamique en cours de diffusion, pour atteindre un but qui est l'acquisition de nouvelles connaissances et de compétences, autrement dit l'apprentissage.

Une méthode d'IENA propose un ensemble d'objets pédagogiques à construire, de tâches pour les réaliser et de principes d'opération à appliquer en soutien à la conception d'un ENA. Au moment de sa diffusion, ce type d'environnement prend la forme d'un système essentiellement numérique à l'usage des apprenants et apprenantes ainsi que des facilitateurs et facilitatrices (formateurs, formatrices, enseignants, enseignantes, personnes tutrices, etc.). En d'autres termes, l'IENA est un système méthodologique qui vise à développer d'autres systèmes que sont les ENA.

Plus précisément, l'IENA, en tant que méthodologie, doit fournir aux concepteurs et aux conceptrices qui l'utilisent une ou plusieurs « méthodes », celles-ci

étant définies comme des systèmes prescriptifs, plutôt que descriptifs, qui regroupent à la fois des processus, des systèmes conceptuels et surtout des principes méthodologiques (des stratégies) qui servent à guider l'activité humaine dans un domaine (Paquette, 2002).

Une méthode d'IENA se définit donc comme une méthode systémique soutenant l'analyse, la conception, la réalisation et la planification de la diffusion des environnements numériques d'apprentissage, intégrant les concepts, les processus et les principes du design pédagogique, du génie logiciel et du génie cognitif.

1.2 L'IENA comme méthodologie systémique pour le design pédagogique

L'ingénierie pédagogique, en particulier l'IENA, doit d'abord et avant tout être abordée comme un processus complexe de résolution de problèmes, tel que défini dans les sciences cognitives (Newell et Simon, 1972) et souvent étudié comme tel en sciences de l'éducation (Romiszowski, 1981; Reigeluth, 1983, Tennyson, 1990; Merrill, 1994).

Abordé du point de vue des sciences cognitives, le modèle général de la résolution de problèmes invite à caractériser une méthode d'IENA comme un processus systémique comportant :

- un état final de la solution, soit l'ENA à définir;
- un état initial, soit une définition plus ou moins précise du problème de formation auquel l'ENA doit apporter une réponse;
- des opérateurs, ou des processus, qui permettront de transformer la définition initiale du problème de formation en des descriptions de plus en plus précises jusqu'à la définition concrète de l'ENA, à sa réalisation et à sa mise en opération.

La méthodologie systémique a été appliquée pour résoudre des problèmes dans des domaines aussi divers que l'économie, l'architecture, le design de produits et l'éducation. Elle se compose de cinq phases ou processus principaux guidés par des principes méthodologiques :

- *la définition du problème*, c'est-à-dire l'identification la plus précise possible des caractéristiques et des contraintes de la solution recherchée (l'état final), des données ou de la situation actuelle (l'état initial);
- *l'analyse du problème*, consistant à générer des choix possibles pour l'élaboration d'une solution;

- *l'élaboration d'un plan de solution*, soit l'identification des opérations, des étapes, des phases ou des moyens par lesquels on pourra transformer la situation actuelle pour atteindre la situation visée;
- *l'application ou l'implantation du plan de solution*, soit l'assemblage des éléments du plan pour produire des situations de plus en plus proches de la situation visée et l'implantation de celle-ci;
- *l'évaluation de la solution et la révision*, c'est-à-dire, d'une part, la vérification que la solution obtenue correspond bien à la solution recherchée et, d'autre part, l'examen de la solution obtenue aux fins de réutilisation dans la solution d'autres problèmes.

Polya (1957) a qualifié ce processus général de résolution de problèmes d'« heuristique », c'est-à-dire guidé par des principes méthodologiques qui ne garantissent pas le succès, mais qui offrent de « bonnes » chances de solution. L'application de tels principes peut se concrétiser par une suite de questions que l'on peut se poser à chacune des phases, telles que les suivantes :

- À la phase de la définition du problème : Quel est le but? Que cherche-t-on exactement? L'inconnue est-elle décomposable? Si oui, en quelles parties? Quelles sont les données du problème?
- À la phase de l'analyse du problème : Quelles sont les conditions ou les relations entre les données et l'inconnue qui permettraient de passer de l'une à l'autre? Quelles en seront les étapes?
- À la phase de l'élaboration d'un plan de solution : Le problème est-il décomposable en problèmes plus simples? Connaît-on la solution d'un problème analogue? Le plan doit-il procéder des données vers l'inconnue par approximations successives ou, au contraire, partir du but en utilisant un raisonnement régressif, en se demandant de proche en proche de quoi on a besoin, jusqu'à revenir aux données du problème?

Cette méthodologie générale de résolution de problèmes est utile pour guider le processus d'ingénierie pédagogique, en particulier celui de l'IENA, car elle invite à décomposer les problèmes de formation, qui sont rarement simples, en problèmes plus faciles d'accès, pour lesquels une solution pourra être élaborée plus rapidement et ensuite combinée avec d'autres solutions partielles. Cette méthodologie invite également à distinguer l'élaboration du plan (la conception d'un devis pédagogique ou d'un plan de cours) de sa mise en œuvre, à savoir la réalisation des différents éléments du matériel pédagogique, le choix des supports médiatiques ainsi que l'intégration d'outils et de moyens de communication. Elle invite enfin à prévoir une phase d'évaluation, qui peut engendrer un plan de révision, avant de passer à l'implantation définitive du système d'apprentissage.

1.3 L'analyse de la stratégie employée lors de la conception, un système à construire

Les problèmes de conception présentent des similitudes, que l'on travaille dans le domaine de l'architecture, de l'ingénierie d'un système physique, du logiciel ou de l'ingénierie pédagogique. Dans tous les cas, la solution a ceci de systémique qu'elle se présente comme un système à construire devant satisfaire certaines contraintes, très peu définies au départ, qui devront être spécifiées à la phase initiale, puis précisées tout au long du processus. En observant des concepteurs et des conceptrices à l'œuvre pour résoudre des problèmes de design de différents types, Goël et Pirolli (1989) ont identifié un certain nombre de connaissances stratégiques invariantes qu'ils utilisent au cours du processus de résolution et qui tiennent compte de la complexité de ce type de problèmes :



exercer une activité intense de structuration et de restructuration du problème.

Développer plusieurs modèles possibles du système, se concrétisant dans des devis, des schémas fonctionnels, des prototypes.

Comme il n'existe pas de « bonne » ou de « mauvaise » réponse, effectuer l'évaluation continue de la valeur d'une solution ou d'un élément de solution; cette évaluation est cyclique et s'effectue par approximations successives.

Préciser progressivement les contours du système tout en essayant de conserver une certaine latitude.

Décomposer le problème en modules ayant des intersections et des liens plus ou moins élaborés.

Passer des buts abstraits initiaux aux spécifications concrètes finales par une série d'approximations de plus en plus précises jusqu'à produire le système qui constitue la solution au problème initial.

Utiliser abondamment des systèmes symboliques ou graphiques pour décrire les résultats intermédiaires.

Ces principes s'appliquent également à la conception pédagogique à laquelle la méthodologie de l'IENA est destinée. Ils s'appliquent aussi aux trois méthodologies qui s'intègrent dans l'IENA (figure 3.1) : design pédagogique scientifique, ingénierie des systèmes d'information (ou génie logiciel) et ingénierie des connaissances (ou génie cognitif), que nous allons examiner dans les trois prochaines sections.

2 Le design pédagogique scientifique

On peut attribuer l'origine du design pédagogique en tant que domaine scientifique à John Dewey (1900) qui, au début du siècle dernier, réclamait le dévelo-

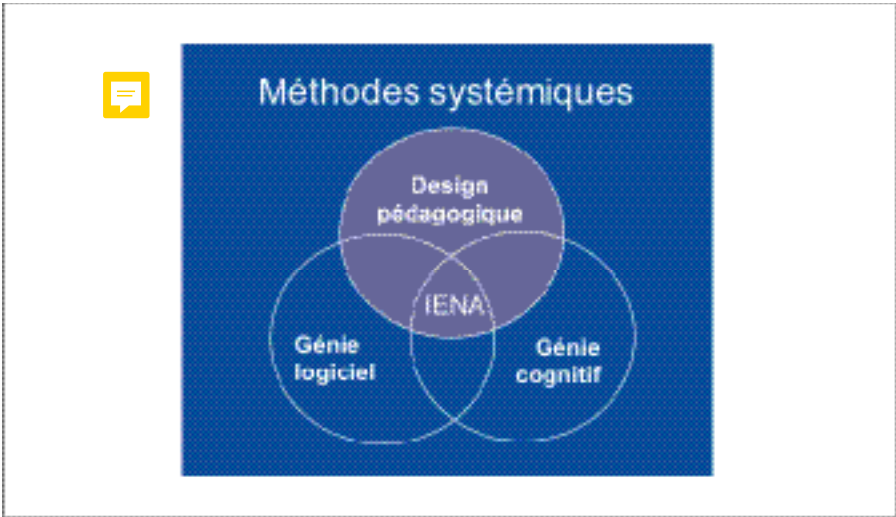


FIGURE 3.1 L'IENA à l'intersection de trois méthodologies proposant des méthodes systémiques

pement d'une « science charnière » entre les théories de l'apprentissage et les pratiques éducatives. Son appel fut largement entendu au début des années 1960, période qui marque l'émergence d'une nouvelle discipline appelée « technologie éducative », principalement sous l'influence des travaux de B. F. Skinner, Jerome Bruner et David Ausubel. Skinner (1954) a proposé une approche scientifique de la pédagogie en la distinguant des travaux sur l'apprentissage, tout en développant le premier modèle d'intervention pédagogique fondé sur le « conditionnement opérant ». Ce modèle validé sur une base empirique a donné lieu au milieu des années 1950 au développement de l'enseignement programmé. Contrairement à l'approche behavioriste de Skinner, Bruner (1966) a proposé une théorie fondée sur la pédagogie de la découverte et l'identification de stades de développement intellectuel. Par ailleurs, Ausubel (1968) a construit un modèle pédagogique qui a mené à des méthodes de présentation de contenus fondées sur les structures cognitives de l'apprenant.

Par la suite, les années 1970 et 1980 ont permis une floraison de travaux visant la construction de « théories de l'enseignement ». On assista alors au développement prolifique d'un appareil théorique et conceptuel incluant une approche cybernétique de l'enseignement (Landa, 1976), l'énoncé de conditions internes et externes d'apprentissage (Gagné, 1970), l'identification de stratégies pédagogiques fondées sur les théories structuralistes de l'apprentissage (Scandura, 1973), le développement d'une théorie cognitiviste de l'enseignement au moyen

d'activités de recherche (Collins et Stevens, 1983), l'élaboration d'une théorie de l'analyse des composantes des stratégies pédagogiques (Merrill, 1983) ainsi que des travaux sur une théorie de l'élaboration de l'enseignement pour l'analyse et le design des tâches de conception pédagogique (Reigeluth et Rodgers, 1983). Ces travaux et d'autres menés par la suite par une diversité de chercheurs et de chercheuses ont été publiés dans les quatre volumes de la série *Instructional-design theories and models* dirigée par Reigeluth (1983, 1999; Reigeluth et Carr-Chellman, 2009; Reigeluth *et al.*, 2017).

Sur la base de ces différents travaux qui ont fourni un solide socle de connaissances qui continue à s'étendre, le design pédagogique, que l'on pouvait désormais qualifier de *scientifique*, se présente aujourd'hui comme l'ensemble des connaissances permettant de comprendre, d'améliorer et d'appliquer des méthodes d'enseignement favorisant l'apprentissage. Par rapport aux théories développées en psychologie de l'apprentissage, le design pédagogique peut être vu comme une forme d'ingénierie visant à améliorer les pratiques éducatives et, ainsi, à optimiser l'apprentissage. Sa relation à la psychologie de l'apprentissage est analogue à celle des méthodes de l'ingénierie par rapport aux sciences physiques, ou à celle de la médecine par rapport aux sciences de la vie.

Comme on l'a indiqué au chapitre 1, les productions résultant de la pratique du design pédagogique se présentent sous la forme d'un ensemble de plans et de devis, à valeur de prescriptions favorisant l'apprentissage, plutôt que d'une description des processus d'apprentissage. Suivant l'ampleur du problème de formation traité, les plans et les devis regroupant ces prescriptions pourront se déployer sur plusieurs niveaux, par exemple : le plan de formation, le programme de formation, le cours, le module et l'activité d'apprentissage.

En tant que discipline faisant partie du domaine des sciences de l'éducation, et plus spécifiquement de celui de la technologie éducative², le design pédagogique scientifique vise à produire des connaissances favorisant une qualité optimale des plans et des devis de l'environnement d'apprentissage en résultant. En ce sens, toute méthode d'IENA se situe dans le cadre du design pédagogique scientifique qui offre de solides fondements reposant sur une somme impressionnante de travaux toujours en développement et méritant d'être pleinement exploités.

Aujourd'hui, il apparaît toutefois nécessaire de renouveler la méthodologie de design pédagogique. Le chapitre 2 a mis en évidence des critiques parfois justifiées formulées par de nombreux artisans du domaine à l'égard de certaines

2 Voir le chapitre 1.

méthodes de design pédagogique. Plus spécifiquement, le défi se situe tant dans la prise en compte des connaissances et des compétences nouvelles requises de la société du savoir, que dans l'opérationnalisation des éléments théoriques dans une méthode à la fois efficace et flexible, tirant parti des multiples possibilités d'apprentissage qu'offrent maintenant les technologies numériques.

3 L'ingénierie des systèmes d'information

L'IENA est une méthodologie qui a pour but de produire des ENA, qui sont des environnements (systèmes ou dispositifs) d'apprentissage faisant appel aux technologies numériques (ou technologies d'information) utilisées dans la majorité des ressources mises à la disposition des apprenants et des apprenantes, et possiblement par d'autres personnes ayant pour rôle de faciliter l'apprentissage. Un ENA est donc, en particulier, un système d'information.

L'ingénierie des systèmes d'information est une méthodologie systémique pour l'analyse, la conception, la réalisation et l'implantation des « systèmes d'information », lesquels intègrent en général plusieurs agents logiciels et agents humains en interaction. Le terme « génie logiciel », pris de façon stricte, correspond plutôt à la construction de ressources pédagogiques informatisées, bien qu'il ait fini par s'étendre à tous les types de systèmes d'information.

3.1 L'évolution des méthodes de conception des systèmes d'information

L'ingénierie des systèmes d'information a réussi à vaincre progressivement la tendance artisanale qui refuse l'usage de toute méthodologie. On reconnaîtra certainement l'analogie entre ce domaine et celui de l'IENA dans la citation suivante (non exempte d'humour) où le terme « génie logiciel » est remplacé par « ingénierie pédagogique », le terme « programmeurs » par « concepteurs » et le terme « code informatique » par « plans et devis des ENA ».

Les premiers agents de résistance aux approches modernes d'**ingénierie pédagogique**, ce sont les **concepteurs** créatifs et solitaires. [...] Ils sont plongés dans leur métier, mais trouvent difficile d'expliquer ce qu'ils font ou de documenter leur travail. [...] Ils considèrent leur travail comme une expérience artistique stimulante. Ils font peu confiance à l'assistance d'autres **concepteurs** ou à de la documentation ou des pratiques qui ne concernent pas directement leurs travaux. Les **plans et devis des ENA** qu'ils élaborent sont uniques, élégants et généralement incompréhensibles à d'autres, ce qui explique pourquoi on les considère comme indispensables. (Firesmith, 1993, p. 1) [traduction et adaptation libres]

La programmation artisanale s'avérait inadéquate pour vaincre la complexité croissante des systèmes d'information, ce pour quoi elle a fait place aux méthodes de génie logiciel. Les systèmes d'information modernes exigent une analyse fine des besoins de qui en fera usage, une interopérabilité des logiciels et une maintenance étalée sur de longues périodes. De plus en plus, la taille, la complexité et le caractère vital des systèmes d'information dépassent tout simplement la capacité de compréhension d'un programmeur unique. Le but premier de l'ingénierie des systèmes d'information est donc de fournir des processus et des outils permettant de produire des systèmes corrects, efficaces, extensibles, flexibles, modifiables, portables, fiables, réutilisables, validables et compréhensibles.

Un système d'information (en particulier un ENA), est un objet artificiel (un artefact) construit en créant une représentation de ses « cas d'utilisation » par un ensemble de personnes et de ressources en interaction, permettant de capter, de mémoriser, de traiter et de rendre disponible en tout ou en partie cette activité sous une forme compréhensible finale. Comme le note un praticien du génie logiciel :

L'une des difficultés majeures de l'ingénierie des systèmes d'information est que les dits systèmes sont des objets plus ou moins immatériels. Personne n'a jamais vu un système d'information et ce système n'est visible qu'à partir de ses représentations : documentation, modèles, etc. [...] Il faut s'assurer que plusieurs interlocuteurs décrivant le même objet vont bien arriver à la même représentation formelle, ce qui pose le problème de la fidélité des techniques de modélisation. (Bouchy, 1994, p. 27)

L'ingénierie des systèmes d'information est une méthodologie qui modélise les interactions des personnes qui y jouent un rôle, leurs processus, leurs ressources et leurs produits de même que les principes d'opération de l'environnement où ces éléments interagissent. Elle est globale, constituant un langage commun à des utilisateurs et à des utilisatrices multiples et à des métiers divers. Elle est dynamique, s'intéressant à l'évolution des états des objets du système à travers divers processus. Elle définit clairement les biens livrables, les produits qui résultent de ces différents processus de conception. Elle repose sur un certain nombre de principes d'opération qui tiennent compte des caractéristiques des activités humaines de design, notamment des liens d'influence enchevêtrés plutôt que linéaires, impliquant de fréquents retours en arrière.

3.2 Les phases en ingénierie des systèmes d'information

Les méthodes d'ingénierie des systèmes d'information, telles que le génie logiciel orienté objet (*Object-Oriented Software Engineering* [OOSE]; Jacobson,

1992) et le processus rationnel unifié (*Rational Unified Process* [RUP]; Quatrani, 2002), prévoient toutes un découpage méthodologique en modules, en phases ou en étapes de conception qui correspondent à des processus qui jalonnent les travaux dans le temps sans toutefois leur imposer une exécution linéaire. On y trouve notamment les processus suivants :

- *L’observation critique, l’état des lieux ou le diagnostic.* Ce processus doit conduire à des recommandations précises et bien étayées quant à la possibilité de construire ou de modifier un système numérique existant. Le produit principal est un diagnostic résultant d’une réflexion sur un nouveau système dont l’équivalent n’existe pas dans l’organisation, ou sur la réingénierie ou la validation d’un système existant en regard d’une nouvelle problématique qui se pose à l’organisation.
- *Les orientations générales, l’établissement des principes d’opération.* En partant d’un diagnostic préliminaire, on étudie les stratégies possibles et on propose un ou plusieurs scénarios généraux d’architecture du système. Ce processus permet notamment de définir des principes d’orientation décrivant le choix des types de ressources à intégrer, des activités et des personnes y jouant un rôle et que l’on doit soutenir, ainsi que des moyens à mettre à leur disposition.
- *La définition préliminaire et l’appel d’offres.* Dans ce processus, on définit, sur la base des orientations générales, une ou plusieurs solutions opérationnelles, tenant compte des offres de produits sur le marché. On y trouvera notamment une formalisation des orientations générales, une première modélisation de la situation cible, l’étude de scénarios contradictoires de mise en œuvre et leur évaluation menant à des recommandations pour la conception du système.
- *L’architecture du système d’information.* Sur la base de la solution retenue, on traitera les problèmes de composition, d’organisation, de structuration du système informatique, du choix des types de ressources et des caractéristiques de l’exploitation du futur système. Cette phase conduit naturellement à la production d’une maquette ou d’un prototype illustrant le mieux possible le fonctionnement général du futur système. Elle se termine dès que les utilisateurs et utilisatrices ainsi que les clients et clientes ont validé, à partir d’un prototype, les choix technologiques clés inclus dans le devis d’architecture.
- *La programmation du système.* La phase de programmation consiste à transformer les spécifications du logiciel provenant de la phase d’architecture, en produits matériels qui, une fois assemblés, deviennent une application informatique. Dans certains cas, l’emploi d’outils de génie logiciel de haut niveau, comme l’« architecture dirigée par les modèles » (*model driven architecture* [MDA]; Kleppe *et al.*, 2003), peut diminuer ou même supprimer la codification traditionnelle dans un langage de programmation. Dans tous les cas, contraire-

ment aux premiers développements de l'informatique, la programmation est un point de passage du développement. Lorsqu'elle est nécessaire, elle demande désormais un effort limité par rapport à l'ensemble des phases de développement du logiciel.

- *La simulation et la qualification du système.* La simulation ou la conception des jeux d'essai et de tests doit permettre de lancer la qualification du système, c'est-à-dire sa validation par la mise à l'essai. La qualification consiste à vérifier l'adéquation du logiciel réalisé aux spécifications du plan d'architecture et aux caractéristiques des services attendus par les personnes qui en feront usage.
- *L'implantation du système.* La phase d'implantation prépare la mise en place du système informatique dans son contexte d'utilisation. Il s'agit non seulement d'installer le nouveau système, mais aussi de préparer les structures d'accueil des logiciels, ainsi que des éléments du matériel soutenant les bases d'information et des procédures de gestion et d'exploitation qui assureront la qualité de son opération dans l'organisation.
- *L'exploitation du système.* Ce processus correspond à la finalité de l'ensemble de la démarche d'ingénierie d'un système d'information et constitue sa mise à l'épreuve dans la réalité. Il est guidé par un plan d'évolution prévoyant généralement une période de rodage à la suite de laquelle on doit s'attendre à une révision de certaines des fonctionnalités du système. Par la suite, on assurera de façon continue à la fois l'observation et la collecte de données sur le système en activité dans son environnement cible, l'analyse et la synthèse de ces observations et la gestion des changements à apporter pour assurer la maintenance du système.

3.3 Les méthodes agiles dans la réalisation des systèmes d'information

De plus en plus, les méthodes agiles, comme *extreme programming* (XP) ou celle qui est proposée par Larman et Basili (2003), se sont imposées en ingénierie des systèmes d'information en réaction aux projets informatiques qui appliquent à la lettre les phases et les étapes présentées plus haut, et dont les délais de réalisation s'avéraient parfois interminables. Tout en tempérant l'excès de méthodologie et la rigidité des planifications à long terme, les méthodes agiles ne proposent pas un retour à la programmation artisanale d'antan. Inspirées des valeurs fondamentales du *Manifeste pour le développement Agile de logiciels* (Beck et al., 2001), ces méthodes proposent notamment les principes suivants extraits du site de ce manifeste³ :

3 <http://agilemanifesto.org/iso/fr/principles.html>



- Accueillez positivement les changements de besoins, même tard dans le projet.

 [...]

- Livrez fréquemment un logiciel opérationnel avec des cycles de quelques semaines à quelques mois et une préférence pour les plus courts.
- Les utilisateurs ou leurs représentants et les développeurs doivent travailler ensemble quotidiennement tout au long du projet.

[...]

- Un logiciel opérationnel est la principale mesure d'avancement.

[...]

- Les meilleures architectures, spécifications et conceptions émergent d'équipes autoorganisées.
- À intervalles réguliers, l'équipe réfléchit aux moyens de devenir plus efficace, puis règle et modifie son comportement en conséquence.

3.4 Les contributions de l'ingénierie logicielle à l'IENA

Nous appliquons le terme « ingénierie », plutôt que « design », aux méthodes d'IENA lorsqu'elles s'inspirent des méthodes de conception des systèmes d'information, en les adaptant à la conception des environnements d'apprentissage.

Une méthode d'IENA prévoira une série de phases de développement inspirée à la fois du design pédagogique scientifique et de l'ingénierie des systèmes d'information :

- On appliquera des processus de conception s'exécutant largement en parallèle et par itérations successives appelées « livraisons ».
- La méthode décrira précisément les produits (« biens livrables ») de ces processus et leur contribution au devis général de l'ENA.
- Elle valorisera une élaboration progressive de l'architecture de l'ENA plutôt que le développement hâtif et artisanal des éléments du matériel pédagogique, tout en tenant compte des principes agiles de conception.
- Enfin, la méthode préparera soigneusement la mise en place ainsi que la diffusion des ENA et leur évaluation.

4 L'ingénierie des connaissances

L'IENA, en tant que méthodologie proposant des méthodes pour la réalisation des ENA, vise à soutenir l'apprentissage et donc l'acquisition de connaissances et de

compétences chez les apprenants. En conséquence, il est capital qu'une méthode d'IANA intègre une méthode d'ingénierie des connaissances et offre aux concepteurs d'ENA des outils de modélisation des connaissances et des compétences.

L'ingénierie des connaissances, parfois qualifiée de « génie cognitif », est à la fois une discipline et une méthodologie; elle s'est développée dans la foulée de l'évolution des systèmes d'information de plus en plus capables d'effectuer un traitement des connaissances. À partir des grandes lignes de cette évolution, il importe d'établir les processus et les principes de l'ingénierie des connaissances, les outils numériques utiles pour réaliser la nécessaire modélisation des connaissances dont les usagers et les usagères doivent disposer et, finalement, les apports de cette discipline à l'ingénierie pédagogique des ENA.

4.1 L'évolution des systèmes à base des connaissances

L'ingénierie des connaissances s'est d'abord développée dans la foulée des applications des systèmes experts et de l'intelligence artificielle à partir des années 1960. Les premiers systèmes pouvant être qualifiés d'« experts » se sont attaqués à des domaines complexes de l'expertise humaine dans le but de créer des logiciels rendant cette expertise disponible à un plus grand nombre de personnes (Hayes-Roth *et al.*, 1983; Waterman, 1986). Le premier système expert, DENDRAL, faisait appel à une importante base de connaissances spécialisée dans l'analyse des données dans le domaine de la chimie. D'autres systèmes experts furent aussi développés dans des domaines comme le diagnostic médical (MYCIN), le calcul symbolique (MACSYMA) ou les relations entre dépôts de minéraux (PROSPECTOR).

Par la suite, les applications se sont davantage diversifiées en incluant la reconnaissance des textes et des images, la traduction automatique et la robotisation. Les systèmes experts ont aussi été utilisés dans des domaines moins spécialisés, s'intégrant de plus en plus aux autres systèmes d'information et leur apportant une intelligence par l'intégration d'expertises articulées autour d'une « base de connaissances ». On a préféré alors parler de systèmes à base de connaissances, d'agents conseillers ou d'agents intelligents (Paquette et Roy, 1990).

Les termes « données », « informations » et « connaissances » marquent trois stades de l'évolution des systèmes d'information. Au début, les grands systèmes informatisés de gestion des organisations permettaient le traitement de grandes quantités de données surtout numériques et parfois symboliques. Au siècle dernier, dès le début des années 1980, on prend conscience de plus en plus des interrelations entre les données et on commence à parler plutôt de traitement des

informations à l'aide de grandes bases de données relationnelles construites sous la direction d'un ou d'une architecte informatique, à partir des informations fournies par un ou une spécialiste de contenu.

Le passage au traitement des connaissances a permis depuis de gérer, de façon intégrée, non seulement les connaissances factuelles (données et informations), mais aussi les connaissances plus abstraites décrivant les concepts, les procédures, les règles, les méthodes, les savoir-faire et les modèles intéressant une communauté de personnes ou une organisation. L'acquisition, la modélisation, le traitement et la communication des connaissances deviennent alors des processus essentiels pour constituer une mémoire des savoirs d'un domaine, développer des méthodes pour l'exploiter et rendre ainsi disponibles des moyens de perfectionner ses connaissances ou de les rendre disponibles à d'autres.

Cela a mené l'ingénierie des connaissances au cœur de l'évolution vers une nouvelle génération de l'Internet et vers le Web social sémantique, succédant aux premières phases de l'Internet qui ne diffusait alors que des pages d'information.

Internet, en devenant un immense réservoir de ressources d'information et de connaissances⁴ avec ses milliards de pages, en relation avec plus de quatre milliards de personnes qui en font usage sur tous les continents, est désormais incontournable comme cadre d'apprentissage. Mais cet immense réservoir de ressources pose un défi énorme à ceux et à celles qui doivent trouver des ressources adaptées à leurs propres besoins d'apprentissage, comme formateurs ou formatrices ou encore pour leurs activités de conception d'un ENA. Malgré l'efficacité des moteurs de recherche sur le Web, les ressources qui s'y trouvent sont décrites essentiellement par des mots-clés, de sorte qu'il faut multiplier et raffiner les recherches pour obtenir quelques centaines, voire des milliers de ressources qu'il faudrait consulter une à une. Leur combinaison est alors laissée à la personne qui en fait usage, car on utilise une très faible partie des capacités d'un ordinateur, celui-ci servant essentiellement de présentateur de textes, de graphiques ou de fichiers audio ou vidéo.

4 Cette base de connaissances que constitue Internet a connu une croissance extrêmement rapide. En décembre 2011, au cours de cette seule année, on recensait 555 millions de sites Web dont 300 millions nouveaux sites. En 2019, on en compte plus de 1,6 milliard dans le monde. En 2011, 800 millions de personnes utilisaient Facebook. On en trouve actuellement 2,3 milliards, soit le quart de la population mondiale! YouTube compte mensuellement 1,9 milliard d'utilisateurs et d'utilisatrices et Instagram, 1 milliard par mois en 2019. Toutes ces personnes contribuent à augmenter la quantité des informations disponibles sur le réseau des réseaux.

Au début des années 2000, le fondateur du Web et actuel directeur du W3C, Tim Berners-Lee, et ses collègues James Hendler et Ora Lassila, (2001) ont proposé des moyens de représenter les connaissances traitées dans les pages et les documents du Web. En décrivant la sémantique des ressources Web au-delà de leur syntaxe, des agents informatiques allaient pouvoir traiter les informations du Web au moyen de recherches plus intelligentes, en offrant des services plus ciblés qu'auparavant parce que basés sur le sens, sur les connaissances derrière les mots et les textes. Cette évolution du Web a été qualifiée de Web sémantique. Pour la réaliser, il aura fallu faire appel à l'ingénierie des connaissances.

Un concept central du Web sémantique est le « Web de données liées », dans lequel ce sont les schémas de données derrière les pages elles-mêmes qui sont mis en réseau, plutôt que simplement les pages par lien Internet. À titre d'exemple, l'entreprise française Antidot⁵ a enrichi une base de connaissances de 43 720 monuments historiques de France au moyen de six autres sources d'information préexistantes ayant chacune été construite indépendamment : la liste des 3 065 gares ferroviaires, celle des 301 stations du métro parisien, les données du code officiel géographique de l'Institut national de la statistique et des études économiques (INSEE), une banque de 122 828 photos de monuments, les descriptions des monuments dans Wikipédia et le service de géolocalisation PlaceFinder qui permet d'afficher les lieux sur une carte géographique. Toutes ces informations ont été reliées entre elles sur le plan des concepts pour former un énorme graphe de données liées. Elles peuvent être consultées de façon intégrée dans un portail Web extrêmement utile pour les touristes et les intervenants de l'industrie touristique.

Cet exemple illustre le principe essentiel du Web sémantique qui est de soutenir un Web intégré sur le plan des connaissances plutôt qu'un Web limité à la simple présentation des informations. La figure 3.2 illustre le concept des données liées à l'œuvre dans l'exemple des monuments historiques.

À gauche dans cette figure, plusieurs données sans lien entre elles sont présentées et correspondent à différentes bases de données mises en relation. Chaque donnée (par exemple, une photo, une gare ou un article Wikipédia) a un ou plusieurs auteurs ou auteures appartenant à une organisation et traite d'un ou de plusieurs sujets dans une hiérarchie de termes. Dans le Web non sémantique, ces données ne communiquent pas entre elles.

5 Voir le site de l'entreprise à <https://www.antidot.net>

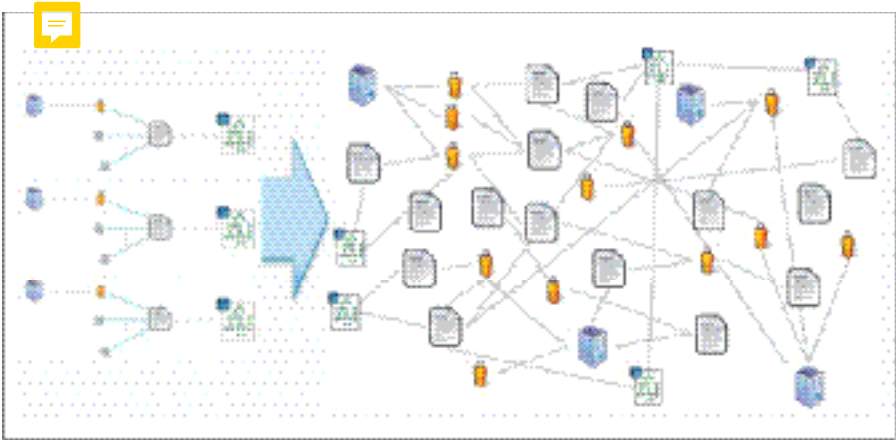


FIGURE 3.2 Des données séparées extraites de trois sources aux données liées, base du Web sémantique

À droite de la flèche, on présente un graphe de connaissances les reliant par des liens. Ce graphe relie non seulement les documents, les connaissances qui s’y trouvent, mais aussi les personnes et les organisations. Tous ces types d’objets sont représentés sur le Web par leur adresse Internet. Chaque lien du graphe représente une propriété reliant un objet à d’autres objets ou à une valeur (un nombre, un sujet, une date, etc.) appartenant possiblement à une autre base de données alimentant d’autres pages Web.

Cette façon de faire permet de répondre à des requêtes intégrant des termes et des objets qui appartiennent à plusieurs sources de données, par exemple « je cherche une photo et un article sur une cathédrale qualifiée de monument historique, située en Alsace, à proximité de la gare de Strasbourg ».

Le Web sémantique opère dans un « monde ouvert ». Contrairement aux premiers systèmes d’intelligence artificielle (systèmes experts, traducteurs de textes, systèmes de reconnaissance de formes), les systèmes du Web sémantique ne sont pas fermés sur une spécialité restreinte utilisant un vocabulaire fixe. Le Web sémantique permet de traiter une variété de concepts et de terminologies qui évoluent régulièrement et séparément, alimentés par un nombre croissant d’intervenants et d’intervenantes sur le Web.

Le cœur du Web sémantique est constitué d’ontologies (Allemang et Hendler, 2011), un type de modèles des connaissances décrit selon une logique de des-

cription qui définit un schéma structurel aux données qui en sont des instances. Pour construire les ontologies du Web sémantique, il faut appliquer des méthodes d'ingénierie des connaissances⁶.

4.2 Les types de connaissances et de modèles

Au cours de cette évolution des systèmes d'information à base de connaissances, on a vu apparaître un nouveau métier, « ingénieur ou ingénieure de la connaissance », ainsi qu'une nouvelle méthodologie, l'ingénierie des connaissances (Vogel, 1988; Ermine, 1993), laquelle étudie les méthodes et les pratiques de développement des systèmes à base de connaissances (Hart, 1997; McGraw et Harbisson-Briggs, 1989).

La notion de représentation des connaissances occupe une place centrale en ingénierie des connaissances, qui a pour but la construction de modèles des connaissances, y compris les ontologies du Web sémantique. Pour échanger, transmettre, transformer ou acquérir des connaissances, il est nécessaire de pouvoir les représenter sur un support externe à notre cerveau, sous une forme transmissible et traitable. Pour cela, on se doit d'utiliser un langage de représentation. C'est ce que nous faisons lorsque nous écrivons et ce pour quoi les langues ont été créées par les différentes civilisations comme langage de représentation. Cependant, les langues naturelles sont parfois inadéquates pour décrire la réalité. Par exemple, la description d'une pièce musicale en langue française prendrait rapidement l'allure d'un roman-fleuve : « jouer la note do de la troisième octave pendant un demi-temps, puis la note ré dièse... ».

Divers types de langages visuels sont utilisés en ingénierie des connaissances. Cela est vrai des divers types de schémas, de diagrammes et de graphes utilisés dans la représentation des connaissances pour l'apprentissage. En voici quelques exemples :

- *Les arbres sémantiques* sont largement utilisés en éducation pour catégoriser des concepts et en fournir des exemples. Le format graphique met en évidence les liens hiérarchiques entre les concepts et leurs exemples, formant un modèle des connaissances.
- *Les cartes conceptuelles* prennent aussi la forme d'un graphe, mais qui n'est pas nécessairement de nature hiérarchique. Dans une carte conceptuelle, les nœuds du graphe représentent les concepts et les liens, la relation entre deux

⁶ Depuis quelques années, des ontologies pour la conception des environnements d'apprentissage ont été développées; certaines d'entre elles sont présentées au chapitre 7.

concepts. Cette fois, la nature de chaque lien est précisée par un mot inscrit sur le trait qui représente graphiquement le lien.

- *Les réseaux sémantiques* ou *graphes entités-associations* (Chen, 2002) sont utilisés pour représenter un ensemble de connaissances sous forme de propositions, comme dans l'exemple de la figure 3.3 où les entités (ici des personnes et des concepts comme sexe et âge) sont les nœuds du graphe et les liens, des relations entre ces entités.

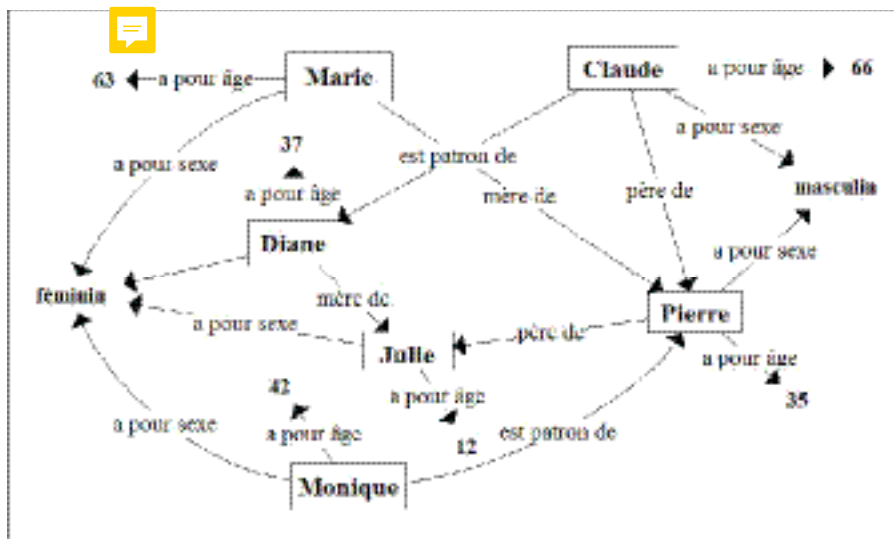


FIGURE 3.3 Un exemple de graphe entité-association

- *Les ordinogrammes* servent à représenter de façon graphique des algorithmes, soit une liste structurée d'actions. Contrairement aux représentations précédentes, qui mettent en évidence des systèmes conceptuels, ils décrivent des connaissances procédurales.
- *Les diagrammes causaux* servent à décrire des liens d'influence ou de cause à effet entre différents facteurs qui peuvent mener à prendre des décisions. Organisés sous forme d'arbres de décision, ils permettent d'établir un ensemble de règles de décision ou de principes stratégiques, lesquels sont à la base des systèmes experts.
- *Les modèles orientés objet* sont à la base de la représentation des systèmes informatiques, comme ceux qui sont développés en utilisant le langage UML (*Unified Modeling Language*) qui est devenu un standard en modélisation des logiciels orientés objet (Booch *et al.*, 2005).

Un langage de représentation des connaissances doit être suffisamment général pour exprimer les divers types de connaissances et de modèles des connaissances dans les différents domaines du savoir dont peut traiter un ENA. Depuis longtemps, les travaux en sciences de l'éducation distinguent des types de connaissances. L'intention derrière cette classification est d'associer des stratégies pédagogiques selon les types de connaissances visés par l'apprentissage.

Dans les travaux de Merrill (1994), Romiszowski (1981), Tennyson (1988) et West *et al.*, (1991) notamment, on distingue en général quatre types de connaissances⁷ : les faits, les concepts, les procédures et les principes.

- Les *faits* sont des assertions qu'une association entre une date et un événement existe, ou qu'un attribut d'un objet a une certaine valeur.
- Les *concepts* sont des classes d'objets, d'événements ou de symboles partageant un certain nombre de caractéristiques communes.
- Les *procédures* sont des ensembles d'étapes permettant de conduire des actions.
- Les *principes* sont des liens de cause à effet dans un processus ou encore des savoirs sur quand et pourquoi utiliser les concepts, les règles et les principes d'un domaine.

Règle générale, dans un modèle des connaissances, on trouve plusieurs de ces types de connaissances, qui donnent lieu à une catégorisation des modèles selon le type de connaissances qui y domine. Le tableau 3.1 en présente des exemples.

4.3 L'ingénierie des connaissances : une nouvelle méthodologie

L'ingénierie des connaissances est une méthodologie regroupant des méthodes, chacune proposant un processus et des principes d'exécution regroupant des opérations comme l'identification des connaissances, leur explicitation, leur représentation et leur formalisation dans un langage symbolique ou graphique facilitant leur utilisation subséquente.

De façon typique, l'ingénieur ou l'ingénieure de la connaissance consulte la documentation du domaine pour identifier les connaissances. Il procède au moyen de discussions avec un ou plusieurs spécialistes du contenu qui détiennent l'exper-

7 Romiszowski (1981) mentionne que ce sont là les quatre catégories d'information utilisées par Williams (1977) pour étendre les taxonomies de Bloom (1956) et de Gagné (1970) et qu'elles ressemblent également aux catégories d'information utilisées auparavant par Horn (1969) comme base de ce que ce dernier nomme « cartographie de l'information » (*information mapping*).

tise qu'il désire modéliser. À l'aide de méthodes d'interview systématique ayant pour but l'acquisition des connaissances, il raffine de plus en plus sa représentation du domaine par étapes successives jusqu'à la capture dans un langage visuel qui sera par la suite intégré, par exemple dans un ENA.

TABLEAU 3.1 Les types de connaissances et de modèles

Modèles de connaissances	Type de connaissances	Exemples
Modèles factuels	La majorité des connaissances sont des faits	Un tableau des marques d'ordinateurs avec leurs propriétés. Les traces d'exécution d'une procédure de traitement d'un formulaire. Un réseau de liens de parenté entre des personnes.
Modèles conceptuels	La majorité des connaissances sont des concepts	Une automobile, ses sous-systèmes et leurs composantes. La taxonomie du règne animal de Linné. Un ensemble de figures géométriques classifiées par types avec leurs composantes.
Modèles procéduraux	La majorité des connaissances sont des procédures	Le déroulement d'une réunion indiqué dans l'ordre du jour avec ses points de discussion et ses sous-points. Une procédure de calcul de l'impôt sur le revenu. Un plan de réalisation d'un projet en équipe. Une boucle de rétroaction dans le fonctionnement d'un thermostat.
Modèles prescriptifs	La majorité des connaissances sont des principes	Une liste de vérification de règles permettant de choisir une maison en fonction de ses besoins. Les lois de la gravitation. La théorie de l'évolution des espèces. Les principes qui régissent le diagnostic médical.
Modèles méthodologiques	Aucun type de connaissances n'est majoritaire. Il s'agit d'un modèle procédural, traitant de concepts comme intrants ou produits, régis par des principes et des personnes.	Un processus ou une méthode de gestion de projets. Un processus de diagnostic du mauvais fonctionnement d'une automobile. Une méthode de conception et de réalisation d'un logiciel. Une méthode d'ingénierie pédagogique. Un processus de rédaction d'un ouvrage en équipe précisant les rôles et les productions de chacun.

La technique de modélisation visant la représentation des connaissances à acquérir dans un environnement d'apprentissage, présentée à la figure 3.4, se compose (liens C) de cinq sous-processus, chacun ayant ses intrants et ses produits (liens IP). Elle est régie (lien R) par des principes généraux de progression qui déterminent quand mettre en œuvre tel ou tel processus. Au départ, un problème de représentation des connaissances est fourni à la personne ou à l'équipe qui conçoit le modèle. Le but, les données et les contraintes de ce problème guideront le développement du modèle. Le résultat final est un modèle des connaissances qui apporte une solution au problème initial, respectant le but de la modélisation et ses contraintes.

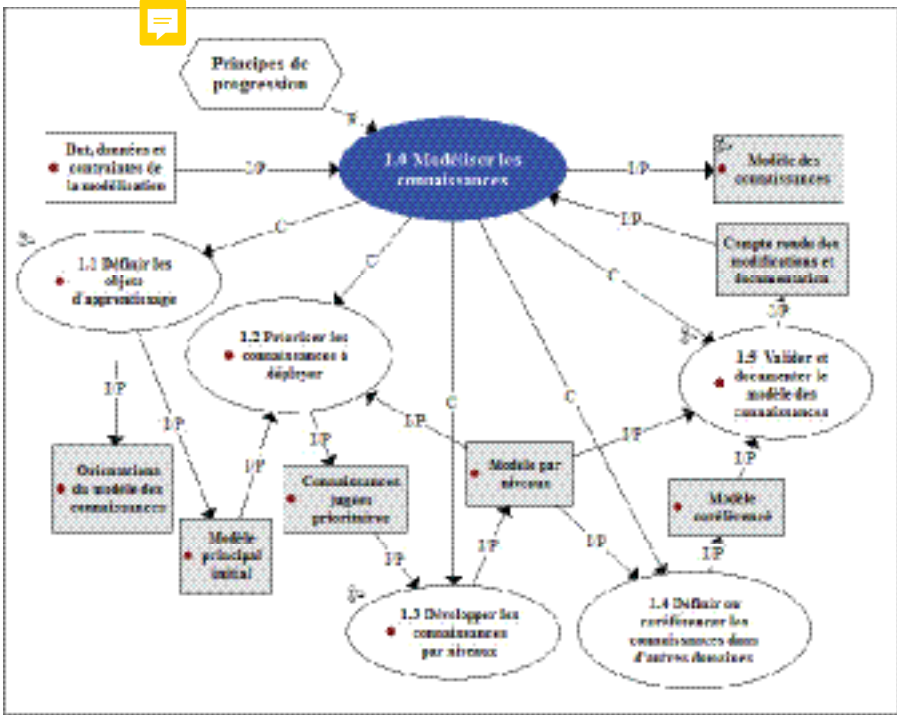


FIGURE 3.4 La technique de modélisation des connaissances visées dans un ENA

La technique *Modéliser les connaissances* se compose (lien C) des cinq sous-processus⁸ suivants :

8 Ces sous-processus sont décrits de façon plus détaillée dans Paquette (2002) aux pages 129 à 166.

- Le premier sous-processus vise à identifier les objets d'apprentissage. Pour ce faire, on définit les orientations du modèle des connaissances (types de connaissances à privilégier, ampleur et type du modèle à élaborer (factuel, conceptuel, procédural, prescriptif ou méthodologique), sources des connaissances à utiliser), ces éléments étant eux-mêmes sélectionnés en fonction du but de la représentation des connaissances et de l'usage futur du modèle. Sur cette base, on développe un modèle initial des connaissances en identifiant un petit nombre de connaissances et de liens.
- Le deuxième sous-processus consiste à orienter le développement du modèle en donnant un ordre de priorités à certaines connaissances qualifiées de prioritaires. Ce sont les connaissances pour lesquelles les publics cibles auxquels le modèle est destiné ont un besoin d'apprentissage. Ces connaissances seront explicitées dans le sous-processus suivant.
- Le troisième sous-processus vise à déployer le modèle initial en associant un modèle plus détaillé à chaque connaissance prioritaire, obtenant ainsi des sous-modèles de niveau 2. Puis, on fera appel de nouveau au deuxième sous-processus pour déterminer les connaissances principales de ces sous-modèles. Celles-ci seront à leur tour développées en leur associant des modèles descendants, donc de niveau 3. On procédera ainsi jusqu'à ce que le domaine de connaissances soit jugé suffisamment élaboré et précis en regard du but et des contraintes initiales du projet de modélisation.
- Le quatrième sous-processus vise à utiliser le modèle en cours de développement pour décrire les connaissances d'un autre domaine appelé « codomaine » qui est représenté dans un autre modèle que celui des connaissances. En contexte d'apprentissage, un codomaine pourrait être, par exemple, le matériel pédagogique fourni dans l'ENA. Un autre codomaine pourrait être le scénario pédagogique. Il s'agit ici d'associer, à certains objets du modèle des connaissances, des codomaines (par exemple, les chapitres d'un livre, les modules d'un système logiciel ou les unités d'apprentissage d'un cours), un sous-ensemble des connaissances du modèle qui en définit le contenu.
- Le cinquième et dernier sous-processus consiste à mettre à l'essai le modèle auprès de personnes représentatives de celles auxquelles il est destiné. Cette validation portera à la fois sur l'exactitude, la cohérence et la complétude du modèle en regard du but et des contraintes initiales. Elle permettra à la fois de réviser le modèle, ses sous-modèles et ses comodèles (représentant les codomaines y étant associés), de produire un compte rendu des modifications apportées et de documenter le modèle par une note qui en décrira le contenu et les articulations.

Les *principes de progression* représentés à la figure 3.4 constituent la structure de contrôle de la méthode de modélisation. Ces principes généraux sont complétés par des principes spécifiques pour chacun des sous-processus qui ne sont pas présentés ici. On distingue trois groupes de principes généraux de progression :

- *Les principes de parallélisme et d’itération.* Une fois le premier sous-processus (1.1) terminé, on avance en parallèle dans les trois sous-processus suivants. On priorise des connaissances du modèle principal (1.2), puis on développe les modèles descendants qui correspondent aux connaissances principales (1.3), et enfin on coréfère les objets d’un ou de plusieurs autres domaines (1.4) à l’aide du modèle par niveaux. On répète ensuite le développement en parallèle de ces trois sous-processus jusqu’à un nombre de niveaux jugé satisfaisant. C’est alors que l’on exécute le dernier sous-processus de validation et de documentation du modèle (1.5); la documentation peut également être développée à l’intérieur du cycle de modélisation par niveaux.
- *Les principes de non-engagement et de retour en arrière.* La progression dans un processus n’implique pas un engagement définitif. On se réserve le droit de revenir modifier le modèle à quelque niveau que ce soit. Dans certains cas, il faudra toutefois aller plus loin et revenir effectuer des changements de structure. Par exemple, le développement par niveaux du modèle peut remettre en question le type de modèle sélectionné au début et conduire à reconstruire le modèle sur une nouvelle base.
- *Les principes de transition et d’arrêt.* Chaque sous-processus n’est pas une condition préalable au suivant au sens strict. Il doit cependant être suffisamment avancé pour que le sous-processus suivant puisse s’amorcer raisonnablement. Ce degré d’avancement suffisant est défini par les principes de transition et d’arrêt suivants, lesquels visent à établir quand on doit quitter un sous-processus pour en entreprendre un autre ou terminer le processus de modélisation :
 - On démarre le sous-processus « identifier les objets d’apprentissage » lorsque le problème de représentation a été suffisamment défini, c’est-à-dire lorsque l’on connaît le but de la modélisation, les données qui nous sont accessibles et les contraintes des publics cibles, l’ampleur prévue du modèle et le type de connaissances et de liens à représenter.
 - On passe aux trois sous-processus suivants lorsque les connaissances les plus essentielles ou utiles ont été déterminées, que le type du modèle principal a été défini et qu’un modèle principal initial a été construit.
 - On passe au dernier sous-processus, « valider et documenter le modèle », lorsque le modèle par niveaux et son association aux codomaines semblent satisfaire le but et les contraintes initiales.

- On termine la modélisation lorsque le modèle et ses sous-modèles ont été validés, révisés et documentés.

4.4 Les langages et les outils de modélisation des connaissances

Une méthode d'ingénierie des connaissances utilise un langage et des outils de modélisation des connaissances. Il est possible de réaliser des cartes conceptuelles ou des modèles des connaissances sans recourir à des outils informatisés, en les traçant directement sur une feuille de papier. On peut aussi les construire avec des logiciels comme Visio et PowerPoint ou des logiciels de dessin. Cependant, les logiciels de modélisation visuels offrent des avantages évidents : facilité de création et de modification du modèle, possibilité de stockage, modes variés de visualisation de la carte, fonctions de recherche, possibilité de création de sous-cartes, rattachement de textes, d'images ou de clips vidéo aux nœuds et aux liens, construction collaborative d'un modèle à distance, évaluation automatisée des modèles, conversion des cartes en liste textuelle, en tableau ou dans un format informatique comme XML (*Extensible Markup Language*) pour l'intégration du modèle dans un ENA.

Le nombre de logiciels de construction de modèles ou de cartes de connaissances de toutes sortes est en progression constante (Basque et al., 2004⁹). En voici quelques exemples :

- *Belvedere*. Outil développé par l'équipe d'Alain Lesgold et Dan Suthers (Learning and Resource Development Center, University of Pittsburgh). Il propose une typologie de trois liens et une typologie de trois objets et permet d'élaborer, en plus de cartes de connaissances en réseaux, des cartes argumentatives ou causales soutenant une démarche d'expérimentation scientifique (*scientific inquiry*).
- *Cmap Tools*. Outil développé à l'Institute for Human & Machine Cognition ([IHMC], University of West Florida) (Windows et Mac). Il permet la création de cartes de connaissances structurées en réseau, sans proposer de typologies de liens et de connaissances. Il permet d'attacher des documents de différents formats (vidéo, texte, image, etc.) aux connaissances.
- *Inspiration*. Développé par Inspiration Software (Oregon) (Windows et Mac). Ce logiciel permet de construire des représentations graphiques en réseau et en arbre, en utilisant des connaissances et des liens non typés. On peut choisir, dans une « librairie », des icônes pour symboliser les connaissances et modifier

9 Voir également le site www.mind-mapping.org

leurs attributs graphiques. Il peut aussi intégrer dans la carte des documents de différents formats (vidéo, texte, image, etc.).

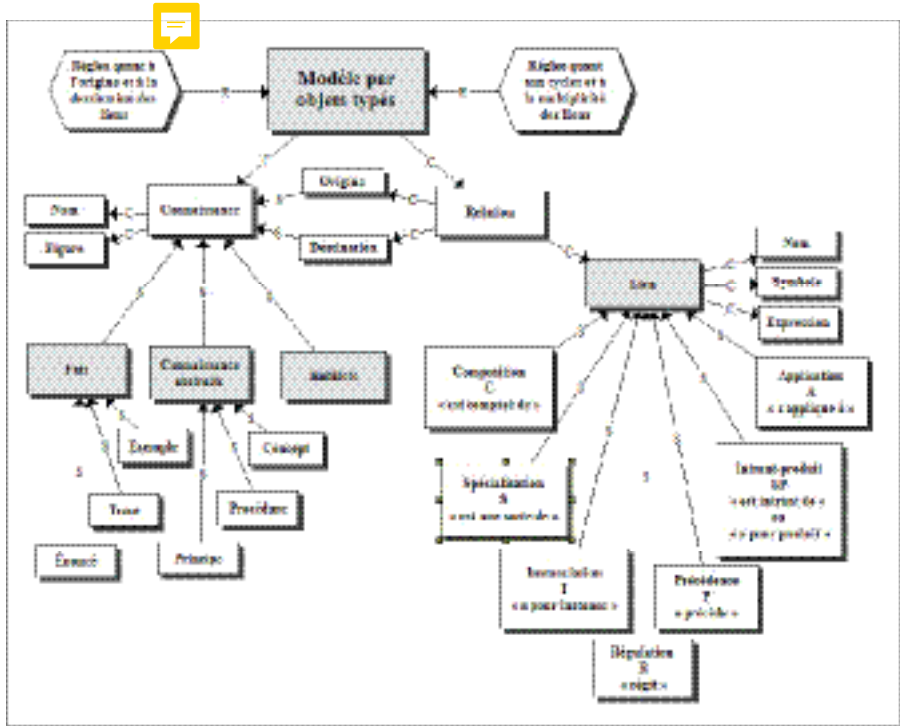
- *Model-It*. Outil développé au Center for Highly Interactive Computing in Education (University of Michigan). Il permet de construire une carte représentant des objets (connaissances) sous forme d'icônes ou d'images qu'on peut choisir à son gré. On définit ensuite des variables pour chaque connaissance représentée et les relations causales entre les connaissances. On peut par la suite simuler le comportement dynamique du système selon les variables et les paramètres de la relation causale qu'on aura définis (fonction Play).
- *MOT*, *MOTplus* et *GMOT*. Trois générations d'outils développés par l'équipe de Gilbert Paquette à l'Institut LICEF (Université TÉLUQ) (Windows et Mac) (Paquette, 2002, 2010). MOT est l'acronyme de « modélisation par objets typés ». Ce logiciel offre un ensemble de formes graphiques pour distinguer les types de connaissances (concept, processus, principe et fait) et de liens (composition, spécialisation, précédence, intransitif/produit, régulation et instanciation, application). Les outils MOT permettent également d'attacher divers fichiers médias aux nœuds de la carte et d'exporter la carte dans un autre fichier, dans une base de données ou dans un fichier XML.

Le schéma de la figure 3.5 utilise le langage MOT pour représenter de façon synthétique les composantes d'un modèle par objets typés. On peut y lire les informations suivantes :

- Un modèle par objets typés est composé uniquement de connaissances et de relations où chacune des relations se compose d'une connaissance appelée « origine », d'un lien et d'une connaissance appelée « destination ».
- Il y a six sortes de connaissances : trois sont des sortes de faits (exemple, trace, énoncé) et trois sont des sortes de connaissances abstraites (concept, procédure, principe).
- Les habiletés servant à définir les compétences sont des sortes connaissances, celles-ci s'appliquant sur d'autres connaissances.
- Chaque connaissance est identifiée par son nom et une figure géométrique.
- Il y a sept sortes de liens : C (est composé de), S (est une sorte de), I (a pour instance), R (régit), P (précède), I/P (est intransitif de/a pour produit) et A (s'applique à).
- Chaque lien est identifié par un nom, un symbole et une expression qui le désigne uniquement.

On distingue aussi sur le graphe deux groupes de principes ou de règles de grammaire. Un premier groupe inclut les règles quant à l'origine et à la destination

des liens qui déterminent les liens possibles entre deux connaissances données selon leur type. Le second groupe inclut les règles qui présentent les propriétés des relations quant aux cycles et à la multiplicité des liens dans un graphe.



Légende : Lien C (se compose de); lien S (est une sorte de); lien R (régit).

FIGURE 3.5 Un modèle MOT de la notion de modèle par objets typés

Le langage MOT a évolué à travers trois générations d'outils de modélisation. Le plus récent, le multi-éditeur GMOT, a été créé entre 2008 à 2012. Cet outil polyvalent regroupe les principales fonctionnalités de l'éditeur graphique précédent MOTplus et est construit sur des bases technologiques plus récentes.

Comme l'indique la figure 3.6, le multi-éditeur GMOT offre cinq éditeurs de modèles :

- L'éditeur de diagrammes informels permet de créer des cartes de connaissances dont les entités et les liens sont non typés.

- L'éditeur de modèles MOT permet de construire des modèles semi-formels comme celui de la figure 3.3, qui respectent la syntaxe et la sémantique du langage MOT.
- L'éditeur de scénarios permet de créer des modèles de processus multi-acteurs, exécutables dans l'interface Web d'un ENA, où les personnes interagissent entre elles, réalisent les tâches dont elles sont responsables en utilisant des ressources (intrants) de ces tâches pour produire des ressources prévues dans le scénario.
- L'éditeur de graphes RDF/RDFS est fondé sur les langages correspondants élaborés par le W3C et qui sont à la base du Web sémantique ou Web de données liées.
- L'éditeur d'ontologies peut être vu comme une extension du vocabulaire RDFS; les ontologies ainsi construites respectent la grammaire du langage d'ontologies OWL-DL (*Ontology Web Language Description Logic*).

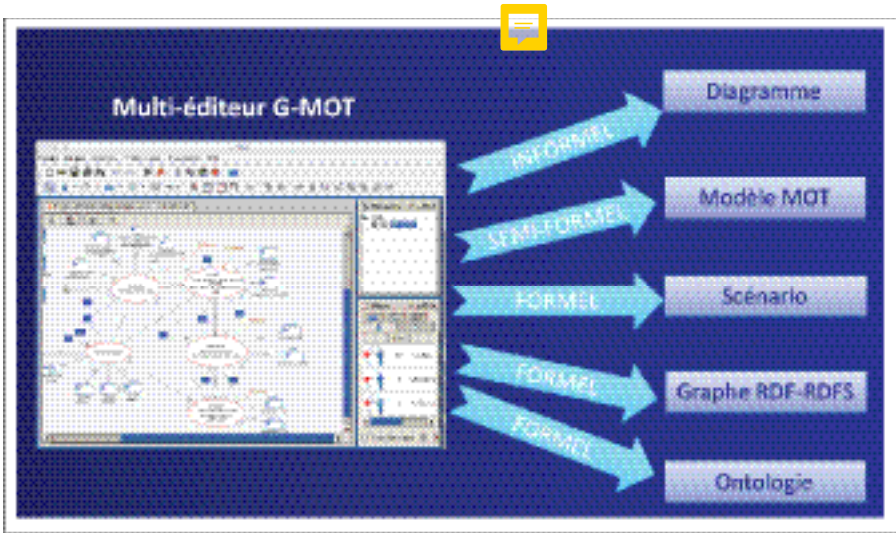


FIGURE 3.6 Les cinq éditeurs du multi-éditeur GMOT

Le langage et les outils de modélisation MOT, MOTplus et GMOT offrent la possibilité d'intégrer dans un même modèle des points de vue différents (déclaratif, procédural, stratégique), ainsi que les principales méthodes de représentation (cartes conceptuelles, algorithmes, arbres de décision, systèmes de règles). Cette généralité permet de construire une variété de modèles à partir d'un petit nombre de types de connaissances et de liens. Ces modèles peuvent être informels (cartes

conceptuelles), semi-formels (en modélisation typée) ou formels (langages RDFS, OWL, scénarios).

4.5 Les contributions de l'ingénierie des connaissances à l'ENEA

On peut voir un apprentissage comme un processus par lequel on passe d'un modèle de ses connaissances internes à un moment donné à un modèle plus évolué de celles-ci à la fin d'un apprentissage.

La représentation des connaissances joue donc un rôle central dans les ENA. Elle peut servir entre autres à diagnostiquer ou à évaluer l'état des connaissances d'un apprenant ou d'une apprenante à tout moment de sa démarche d'apprentissage, à lui communiquer des informations structurées utiles pour son apprentissage; elle peut aussi servir de base à des stratégies d'apprentissage utilisées par la personne elle-même pour améliorer son modèle des connaissances. Cela conduit à plus de stratégies d'utilisation de la modélisation des connaissances, dont les suivantes :

- *Diagnostiquer ou évaluer les connaissances d'une personne.* Ce processus se réalise en trois étapes : 1) identifier les compétences de la personne sur un sujet au moyen de diverses techniques d'évaluation ou d'autoévaluation, 2) les représenter structurellement, puis 3) comparer cette représentation avec un modèle idéal, par exemple celui qui est fourni par des experts du domaine (Jonassen *et al.*, 1993).
- *Structurer le contenu de l'apprentissage.* Le but ici est de fournir une représentation structurée du contenu dans un site Web et/ou de rassembler les éléments du matériel pédagogique en fonction de cette structure. On pourra utiliser diverses méthodes de représentation du contenu sous forme de représentations graphiques, telles que les cartes conceptuelles, les réseaux sémantiques, les cartes causes-effets, les modèles MOT, etc., dans le but d'offrir une vue synthétique de la matière.
- *Apprendre en représentant des connaissances.* Il s'agit d'une stratégie d'apprentissage utilisée par la personne elle-même, seule ou en équipe, dans la découverte d'un domaine (Basque, 2012; Basque et Pudelko, 2010). Typiquement, l'apprenant ou l'apprenante explore le domaine en consultant diverses ressources sur le Web et construit son propre modèle des connaissances qu'il peut ensuite comparer avec ceux d'autres apprenants et apprenantes ou faire valider par une personne connaissant bien le domaine.

L'ingénierie des connaissances permet à la personne qui conçoit un ENA de construire une vue explicite et structurée des connaissances visées qui peut gui-

der la sélection et la structuration du contenu, des activités d'apprentissage et des ressources mis à la disposition des usagers et des usagères de l'ENA. C'est de cette façon que l'on peut intégrer l'ingénierie des connaissances à une méthode d'IANA. L'ingénierie des connaissances est au cœur de plusieurs processus de la méthode où elle sert notamment à :

- construire un modèle des connaissances et des compétences visées dans l'ENA;
- élaborer les scénarios pédagogiques décrivant les activités proposées ainsi que les ressources utilisées et produites dans ces activités;
- définir les modèles médiatiques des ressources Web nécessaires à l'apprentissage;
- spécifier les processus logistiques de diffusion des ENA lorsqu'ils seront mis en œuvre.

Sur un autre plan, l'ingénierie des connaissances nous aidera à définir la méthode d'IANA elle-même. En appliquant une forme d'ingénierie des connaissances à l'activité de conception pédagogique, elle permettra d'identifier les concepts, les processus et les principes constitutifs de la méthode qui seront utiles. Les sources d'expertise seront puisées à la fois dans les théories et les méthodes de design pédagogique en sciences de l'éducation, ainsi que dans les concepts, les processus et les principes du génie logiciel et du génie cognitif.

Conclusion

Ce chapitre a situé l'IANA au confluent du design pédagogique, du génie logiciel et de l'ingénierie des connaissances. Ces trois méthodologies ont été intégrées jusqu'à un certain point dans la méthode d'ingénierie des systèmes d'apprentissage (MISA¹⁰) développée par l'Institut LICEF. Il y aura lieu d'assurer une plus forte intégration, notamment de certains aspects plus récents du design pédagogique scientifique et du génie logiciel fondé sur la science des données. C'est d'ailleurs encore plus vrai pour la plupart des méthodes d'ingénierie pédagogique qui parfois se concentrent uniquement sur une ou deux de ces trois méthodologies.

Par ailleurs, l'IANA doit intégrer les quatre phases principales du cycle de vie d'un ENA présenté à la figure 3.7 avec les personnes qui y jouent un rôle, les processus qui s'enchaînent et dont celles-ci sont responsables. Le concepteur

¹⁰ Voir la discussion du chapitre 4 sur la MISA.

ou la conceptrice pédagogique produit les plans et les devis de l'ENA qui servent d'intrant à sa réalisation par un ou une spécialiste en médiatisation. L'ENA est ensuite mis en diffusion et utilisé par les apprenants et les apprenantes et souvent par un ou plusieurs facilitateurs et facilitatrices. Un ou une gestionnaire de l'ENA s'assure de la maintenance de la qualité de l'ENA en recueillant et évaluant des données d'utilisation, en produisant des demandes pour sa révision au besoin. Ces demandes de révision sont communiquées au concepteur ou à la conceptrice, ce qui amorce un autre cycle conception/réalisation/utilisation/évaluation.

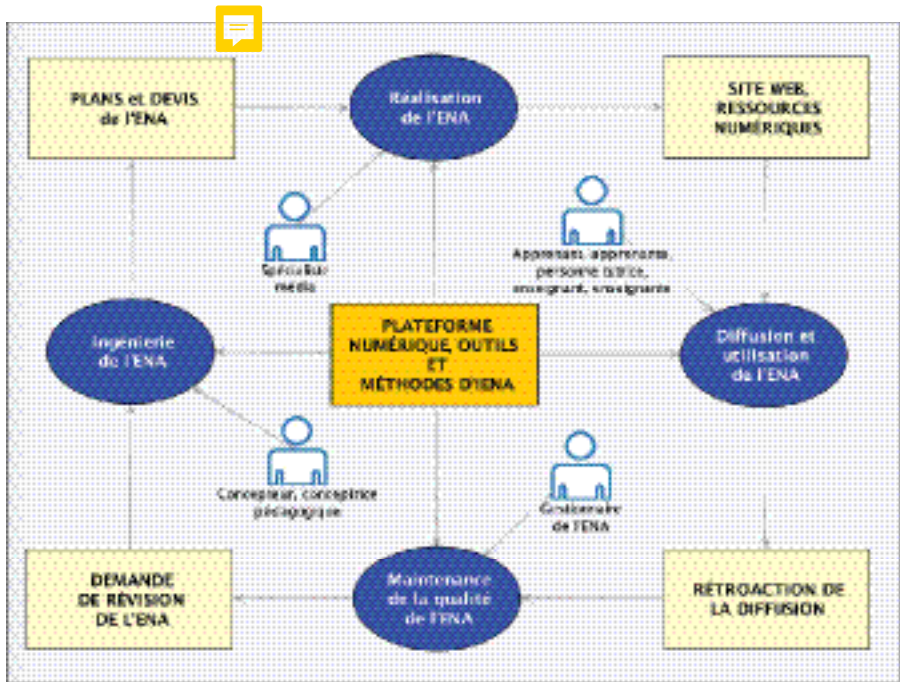


FIGURE 3.7 Le cycle de vie d'un ENA (Paquette, 2014)

La problématique que soulève l'implantation des pratiques de l'ENA dans les systèmes éducatifs a été trop brièvement évoquée à ce jour. Il importe de souligner ici l'énorme défi que représente l'implantation de l'ENA, un défi majeur trop souvent occulté, qui implique d'importantes transformations dans la pratique des établissements d'enseignement et également dans les modes d'apprentissage et d'enseignement.

Pour les établissements, la pratique de l'ENA implique notamment la mise en place d'une infrastructure technologique adéquate, le recrutement et l'intégration


d'équipes de spécialistes, la modification des processus de travail institutionnel pour soutenir une éducation de qualité. L'ampleur de ces transformations explique le retard à tirer pleinement parti des énormes possibilités offertes par les technologies numériques, limitant souvent leur usage à l'imitation de présentations en classe, plus faciles à mettre en œuvre.

Par ailleurs, les personnes enseignantes doivent transformer en profondeur leurs pratiques pédagogiques et se voient forcées de ne plus envisager leur tâche en solo. On ne peut demander à une personne enseignante seule d'intégrer à la fois des compétences en design pédagogique, des compétences informatiques et médiatiques et des compétences en ingénierie des connaissances. Le travail collaboratif d'ingénierie des ENA devient incontournable.

Pour les apprenants et les apprenantes surtout, l'usage des ENA est beaucoup moins confortable que l'écoute de présentations en classe, mais il est potentiellement plus fécond. L'apprentissage requiert davantage d'autonomie et de participation interactive, sans compter les nouvelles compétences informationnelles et techniques requises¹¹. L'ENEA doit tenir compte de nouveaux objectifs et de nouvelles conditions d'apprentissage, indispensables dans une société numérique de plus en plus axée sur les connaissances et les compétences.

11 Voir le chapitre 12 à ce sujet.

Références

- Allemang, D. et Hendler, J. (2011). *Semantic Web for the working ontologist: Effective modeling in RDFS and OWL*. Morgan-Kaufmann/Elsevier.
- Ausubel, D. P. (1968). *Educational psychology: A cognitive view*. Rhinehart & Winston.
- Basque, J. (2012). Apprendre en construisant des cartes de connaissances à l'aide d'un outil logiciel : oui, mais selon quelle technique? Dans *Actes du XIIIe colloque pédagogique de l'Alliance française de São Paulo*. Alliance française de São Paulo. <https://r-libre.teluq.ca/372/> 
- Basque, J. et Pudelko, B. (2010). Modeling for learning. Dans G. Paquette (dir.), *Visual knowledge modeling for semantic web technologies: Models and ontologies* (p. 325-343). IGI Global. <https://r-libre.teluq.ca/433/>
- Basque, J., Rogozan, D. et Pudelko, B. (2004). *Les outils informatisés de construction de cartes de connaissances : des outils pour l'apprenant, le concepteur et le formateur en contexte de téléapprentissage* [communication]. Congrès de l'ACFAS, 4e colloque annuel du CIRTA, Montréal, 10-12 mai 2004. <https://r-libre.teluq.ca/2214/>
- Beck, K., Beedle, M., Bennekum, A. V., Cockburn, A., Cunningham, W. et al. (2001). *Manifeste pour le développement Agile de logiciels*. Site du Manifeste. <http://agilemanifesto.org/iso/fr/manifesto.html>
- Berners-Lee, T., Hendler, J. et Lassila, O. (2001, mai). The semantic Web. *Scientific American*, 284(5), 34-43.
- Bloom, B. S. (1956). *Taxonomy of educational objectives: Book I. Cognitive domain*. Longman.
- Booch, G., Rumbauch, J. et Jacobson, I. (2005). *The unified modeling language user guide* (2^e éd.). Addison-Wesley.
- Bouchy, S. (1994). *L'ingénierie des systèmes d'information évolutifs*. Eyrolles.
- Bruner, J. S. (1966). *Towards a theory of instruction*. Harvard University Press.
- Chen, P. (2002). Entity-relationship modeling: Historical events, future trends, and lessons learned. *Software pioneers: contributions to software engineering*. Springer.
- Collins, A. et Stevens, A. L. (1983). A cognitive theory of inquiry teaching. Dans C. Reigeluth (dir.), *Instructional theories in action: Lessons illustrating selected theories and models* (p. 247-278). Lawrence Earlbaum.

- Dewey, J. (1900). Psychology and social practice. *The Psychological Review*, 7, 105-124.
- Ermine, J. L. (1993). *Génie logiciel et génie cognitif pour les systèmes à base de connaissances. Volume 1. Aspects méthodologiques*. Lavoisier Tec & Doc.
- Firesmith, D. G. (1993). *Object-oriented requirements analysis and logical design*. Wiley.
- Gagné, R. M. (1970). *The conditions of learning* (2^e éd.). Holt, Rhinehart & Winston.
- Goel, V. et Pirolli, P. (1989, printemps). Design within information-processing theory: The design problem space, *AI Magazine*, 10(1), 19-36.
- Hart, A. (1997). *Acquisition du savoir pour les systèmes experts*. Dunod.
- Hayes-Roth, F., Waterman, D. A. et Lenat, D. B. (1983). *Building expert systems*. Addison-Wesley.
- Horn, R. E. (1969). *Information mapping for learning and reference*. Information Resources Inc.
- Jacobson, I. (1992). *Object-oriented software engineering: A use case driven approach*. ACM Press.
- Jonassen, D. H., Beissner, K. et Yacci, M. (1993). *Structural knowledge: Techniques for representing, conveying and acquiring structural knowledge*. Laurence Earlbaum Associates.
- Kleppe, A. G., Warmer, J. B. et Bast, W. (2003). *MDA explained. The model driven architecture: practice and promise*. Addison-Wesley.
- Landa, L. (1976). *Instructional regulation and control: Cybernetics, algorithmization, and heuristics in education*. Educational Technology Publications.
- Larman, C. et Basili, V. R. (2003, juin). Iterative an incremental development: A brief history, agile methods. *Computer Journal, IEEE Computer Society*.
- Le Moigne, J. L. (1995). *Les épistémologies constructivistes*. Presses Universitaires de France.
- McGraw, K. L. et Harbisson-Briggs, K. (1989). *Knowledge acquisition*. Prentice-Hall.
- Merrill, M. D. (1983). Component display theory. Dans C. Reigeluth (dir.), *Instructional theories in action: Lessons illustrating selected theories and models* (p. 279-333). Lawrence Earlbaum.
- Merrill, M. D. (1994). *Principles of instructional design*. Educational Technology Publications.
- Newell, A. et Simon, H. (1972). *Human problem solving*. Prentice-Hall.

- Paquette, G. et Roy, L. (1990). *Systèmes à base de connaissances*. Télé-université du Québec.
- Paquette, G. (2002). *La modélisation des connaissances et des compétences, un langage graphique pour concevoir et apprendre*. Presses de l'Université du Québec.
- Paquette, G. (2010). Ontology-based educational modelling: Making IMS-LD visual. *Technology, Instruction, Cognition and Learning*, 7(3-4), 263-296.
- Paquette, G. (2014). Technology-based instructional design: Evolution and major trends. Dans M. Spector, M. D. Merrill, J. Elen J. et M. J. Bishop (dir.), *Handbook of research on educational communications and technology* (p. 661-671).
- Polya, G. (1957). *Comment poser et résoudre un problème?* Dunod.
- Quatrani, T. (2002). *Visual modeling with rational rose 2002 and UML*. Addison-Wesley Object Technology Serie.
- Reigeluth, C. M. (dir.). (1983). *Instructional-design theories and models: Vol. I. An overview of their current status*. Erlbaum.
- Reigeluth, C. M. (dir.). (1999). *Instructional-design theories and models: Vol. II. A new paradigm of instructional theory*. Erlbaum.
- Reigeluth, C. M., Beatty, B. J. et Myers, R. D. (dir.). (2017). *Instructional-design theories and models: Vol. IV. The learner-centered paradigm of education*. Routledge.
- Reigeluth, C. M. et Carr-Chellman, A. A. (dir.). (2009). *Instructional-design theories and models: Vol. III. Building a common knowledge base*. Routledge.
- Reigeluth, C. M. et Rodgers, C. A. (1980). *The elaboration theory of instruction: Prescriptions for task analysis and design*. *NSPI Journal*, 19(1), 16-26.
- Romiszowski, A. J. (1981). *Designing instructional systems*. Kogan Page London/ Nichols Publishing.
- Scandura, J. M. (1973). *Structural learning I: Theory and research*. Gordon & Breach science Publishers.
- Simon, H. A. (1981). *The sciences of the artificial*. The MIT Press. [Traduction française de J.-L. Le Moigne (1990), *Sciences des systèmes, sciences de l'artificiel*, Dunod.]
- Skinner, B. F. (1954). The science of learning and the art of teaching. *Harvard Educational Review*, 24(2), 86-97.
- Tennyson, R. D. (1990). Cognitive learning theory linked to instructional theory. *Journal of Structured Learning*, 10(3), 249-258.

Vogel, C. (1988). *Génie cognitif*. Masson.

Waterman, D. A. (1986). *A guide to expert systems*. Addison-Wesley.

West, C. K., Farmer, J. A. et Wolff, P. M. (1991). *Instructional design, implications from cognitive science*. Allyn and Bacon.

Williams, R. G. (1977). A behavioural typology of educational objectives for the cognitive domain. *Educational Technology*, 17(6).