

Le raisonnement à partir de cas

Notes des étudiants du M2IA 2016-2017, reprises et complétées par Amélie Cordier

1. Introduction

1. Définitions

Le raisonnement à partir de cas est un paradigme de l'intelligence artificielle qui consiste à utiliser les solutions de problèmes passés, déjà résolus, dans le but de résoudre de nouveaux problèmes. La réutilisation peut consister en la simple ré-application de la solution précédente, ou bien en l'application d'une solution « adaptée » pour mieux répondre aux spécificités du problème.

2. Les fondements historiques

Le RàPC prend ses racines dans les travaux de Schank, Minsky et Abelson.

[Marvin Minsky](#) (psychologue) : 1975, grand père fondateur de l'IA (décédé en janvier 2016, RIP), très provocateur dans sa façon de voir l'IA. Sa première grande contribution : [concept de frame](#).

Selon Minsky, notre pensée est organisée en schéma.

“Pour toutes les tâches légèrement complexes de notre quotidien, on a des schémas.” → *Aller au restaurant, attendre d'être rentré, être placé... Chez Bob Grand* « révolutionnaire » : il a une approche peu conventionnelle de l'IA et pousse les chercheurs et ses collaborateurs à aller « au-delà » des évidences.

[Roger Schank](#) (spécialiste, et précurseur en Intelligence Artificielle) invente les [Memory Organisation Packets](#). Son hypothèse va plus loin que celle de Minsky : on organise les schémas par paquet, et quand on a besoin de parcourir le schéma, on ouvre le paquet et on l'explore (travaux réalisés entre 1975 et 1982¹).

[Janet L. Kolodner](#), auteur de *“[Inside Case-Based Reasoning](#)”* en 1983, introduit pour la première fois le terme de « *case-based reasoning* » (raisonnement à partir de cas). Son ouvrage est un ouvrage de synthèse des travaux de l'époque, dans lequel elle s'appuie fortement sur les travaux de Minsky et Schank.

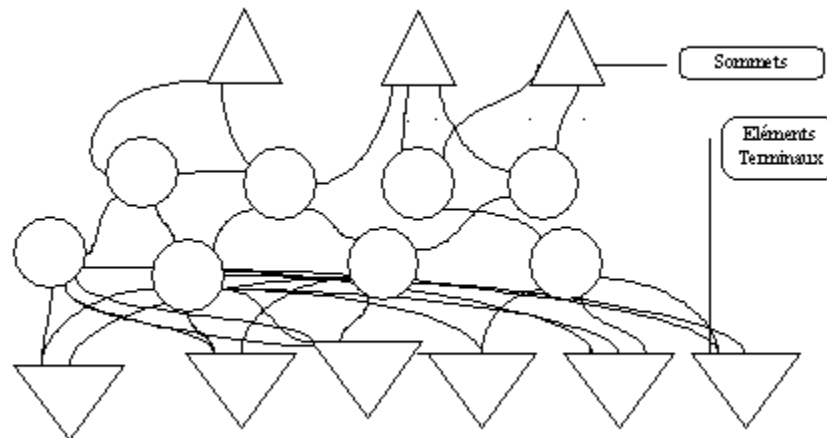
En 1994, Aamodt et Plaza publient un article désormais célèbre dans lequel ils proposent un « cycle » du raisonnement à partir de cas composé de 4 étapes

1 Schank, Roger. *Dynamic Memory: A Theory of Learning in Computers and People*. New York: Cambridge University Press, 1982.

(retrive, reuse, revise and retain). C'est le cycle le plus souvent utilisé pour expliquer comment fonctionne le RàPC.

3. Quelques concepts importants

Les « frames »



Dans le modèle de frames de Minsky, les pensées sont organisées en « schémas ». On se réfère à ces schémas pour résoudre des tâches et organiser notre mémoire.

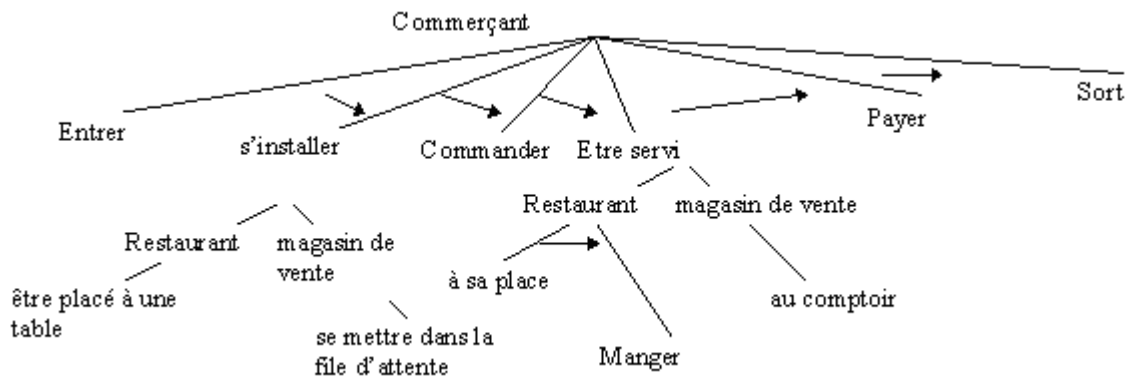
Dans ce modèle, les sommets sont toujours « vrai » (il s'agit de concepts de haut niveau), et les éléments terminaux correspondent à des éléments de la vie réelle, ce sont des concepts de bas niveau.

Lorsque les schémas sont mobilisés pour conduire un raisonnement, l'ordre des étapes suivies peut changer. Des adaptations de schémas peuvent être nécessaires. Pour cela, on utilise en priorité le raisonnement logique, et, si ça ne suffit pas, d'autres formes de raisonnements.

Minsky a des propos assez virulents sur le sujet, et défend notamment le fait que la seule logique classique ne peut pas suffire pour représenter le monde, et qu'il faut penser à d'autres modes de représentation des connaissances.

Les MOPs (entre 1975 et 1982).

MOPs : Memory Organisation Packets. Ils permettent de regrouper des schémas en paquet pour naviguer plus simplement entre les schémas.



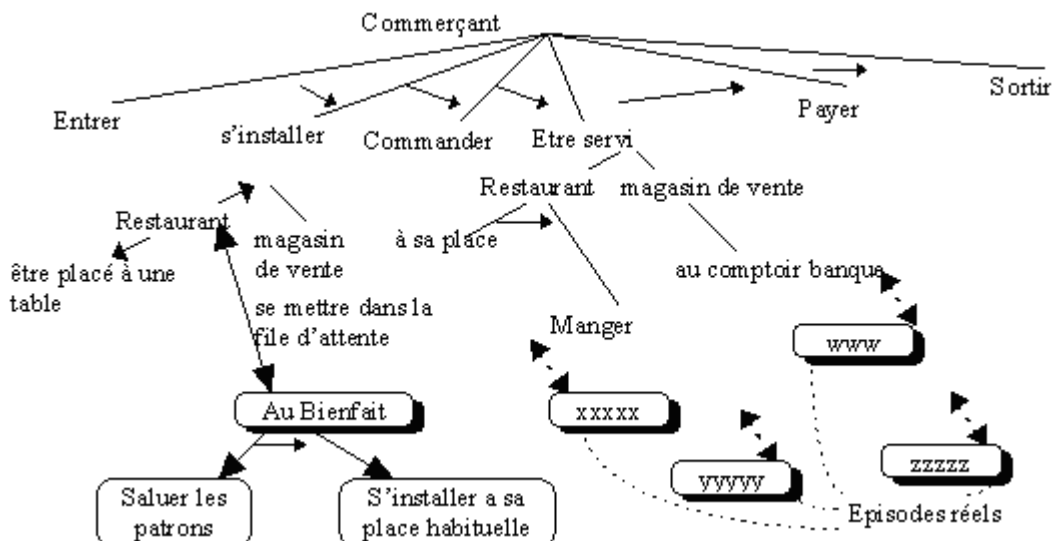
Les paquets sont « ouverts » au fur et à mesure, lorsqu'on en a besoin.

Ex : Trajet Lyon → BNF. Décomposition en paquet :

- Premier paquet : Lyon → Paris (train, vélo, covoit', etc...)
- Second paquet : Paris → BNF (métro, vélo, voiture, etc..)

La création de plusieurs schémas est moins coûteuse que l'adaptation systématique de schémas... Cela dit, la réutilisation de schémas est fréquente car elle représente un gain en mémoire.

Ci-dessous, un exemple d'implémentation dans le cadre d'une résolution de problème



Pour pouvoir résoudre un problème, il est nécessaire d'être capable de décrire la situation en terme d'indicateurs, de fouiller la mémoire à l'aide de descripteurs pour trouver des schémas correspondants au problème et d'essayer d'appliquer ce schéma au problème. Si le schéma n'est plus applicable, il est nécessaire d'adapter le schéma.

2. Raisonnement à partir de cas (RàPC ou Case Based Reasoning CBR)

Depuis la publication du livre de Janet Kolodner, la communauté de recherche sur le raisonnement à partir de cas s'est structurée : il existe plusieurs ressources disponibles, gérées par une communauté rassemblée autour d'une conférence annuelle (ICCBR) et d'un ensemble d'activités (point d'entrée : [CBR Wiki](#)).

Il existe au moins 6 laboratoires en France dans lesquels des équipes dédient une partie de leurs recherches au RàPC (Paris, Nancy, Lyon, Besançon).

1. Pourquoi le CBR a eu du succès ?

Dans les années 1990, l'IA a rencontré un grand succès industriel, en grande partie grâce aux systèmes experts (systèmes à base de règles). Malheureusement, à cause des difficultés à modéliser les connaissances, mais surtout à maintenir les systèmes, l'engouement s'est vite éteint.

Exemple de système ayant eu du succès à cette époque : [MYCIN](#), système de diagnostic.

Les systèmes experts requièrent une modélisation relativement exhaustive du monde sur lequel on souhaite raisonner. Or, tout modéliser est loin d'être possible. À l'inverse, dans le cas du raisonnement à partir de cas, on mémorise les expériences de résolution de problèmes au fur et à mesure qu'on les rencontre (diagnostic, planification, aide à la décision, conception, etc.). Certes, au démarrage, le système est moins performant, mais il gagne en compétences au fur et à mesure, et il est moins difficile à maintenir. De plus, le RàPC est très adapté à de nombreux domaines (listés juste avant), et correspond bien à une de nos façons de raisonner... beaucoup de gens se l'approprient donc assez facilement.

2. Notion de base de cas.

Base de cas : collection de cas de résolution du même problème.

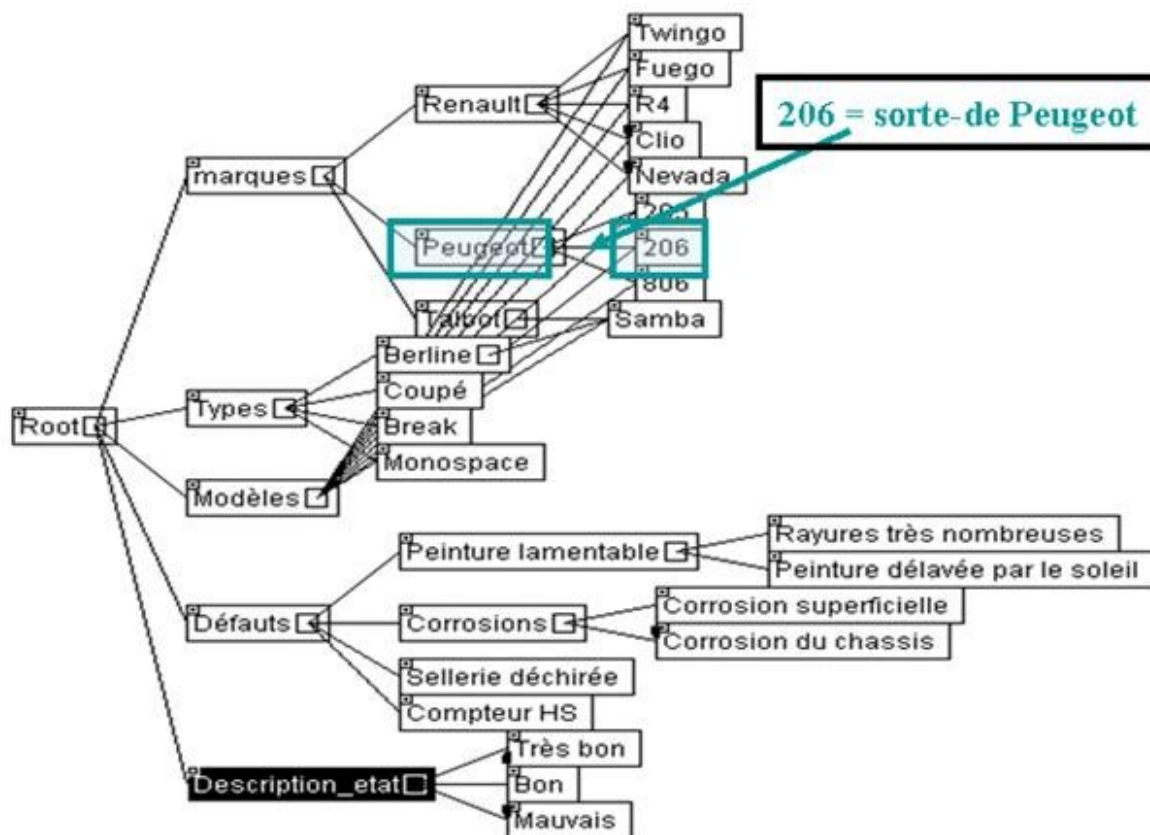
Ex : une base de cas résolus de vente de voitures.

Cas : description d'un problème et sa solution

- problème : ensemble des descripteurs (en général, sous forme de couple attribut-valeur, mais d'autres structures sont possibles) du problème
- solution : valeur ou ensemble d'étapes (idem, attributs, descripteurs, etc.).

Un cas est une expérience de résolution de problèmes. C'est aussi la description d'un problème et de sa solution.

E.g : livre de recette de cuisine = base de cas



Ontologie des descripteurs d'une voiture = couple {attribut, valeur}.

Une ontologie est un ensemble de termes reliés entre eux par des relations vérifiées. Relations les plus classiques : héritage (sorte-de, est-un...), composition

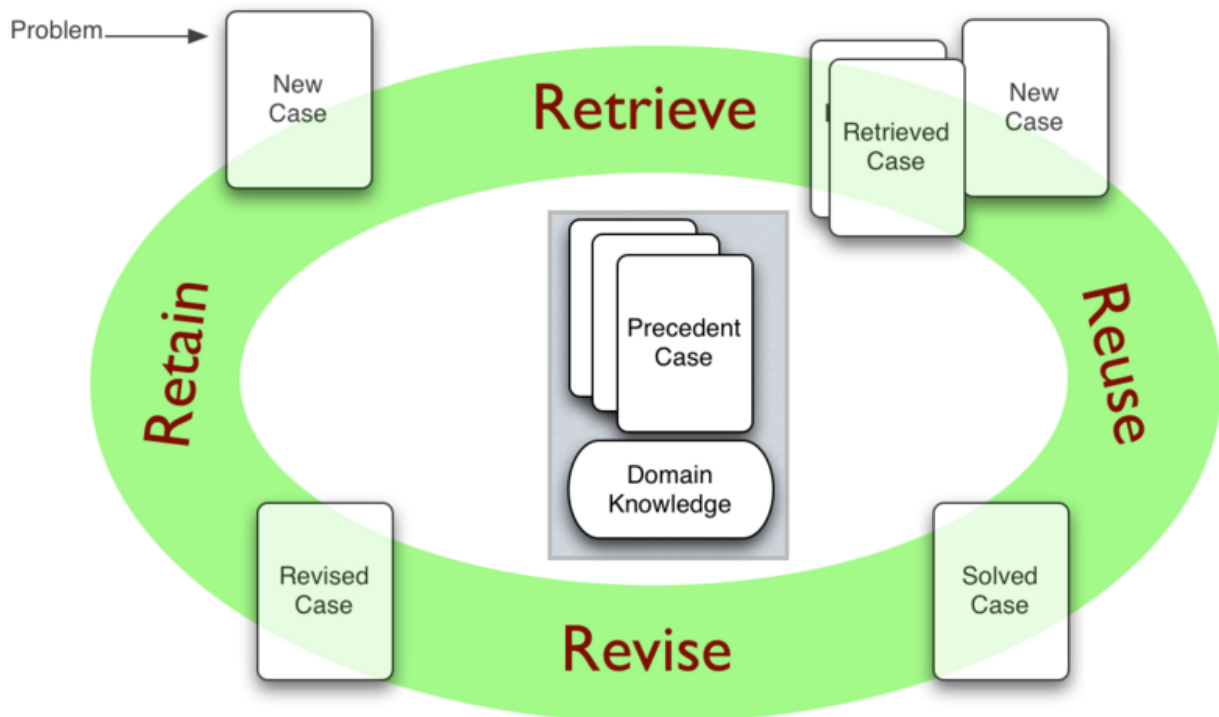
(est-composé-de, composant-de). En général, l'ontologie permet d'exprimer un vocabulaire pour décrire un domaine précis. Souvent, les descripteurs des cas sont choisis dans l'ontologie.

En RàPC, il ne s'agit pas de décrire le monde mais de décrire des relations qui sont vraies entre les termes utilisés pour les valeurs de descripteurs.

3. Le cycle de RàPC

En 1994, Aamodt & Plaza, dans leur publication : [Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches](#), tentent de formaliser le RàPC.

Le schéma quasi-original est donné ci-dessous.



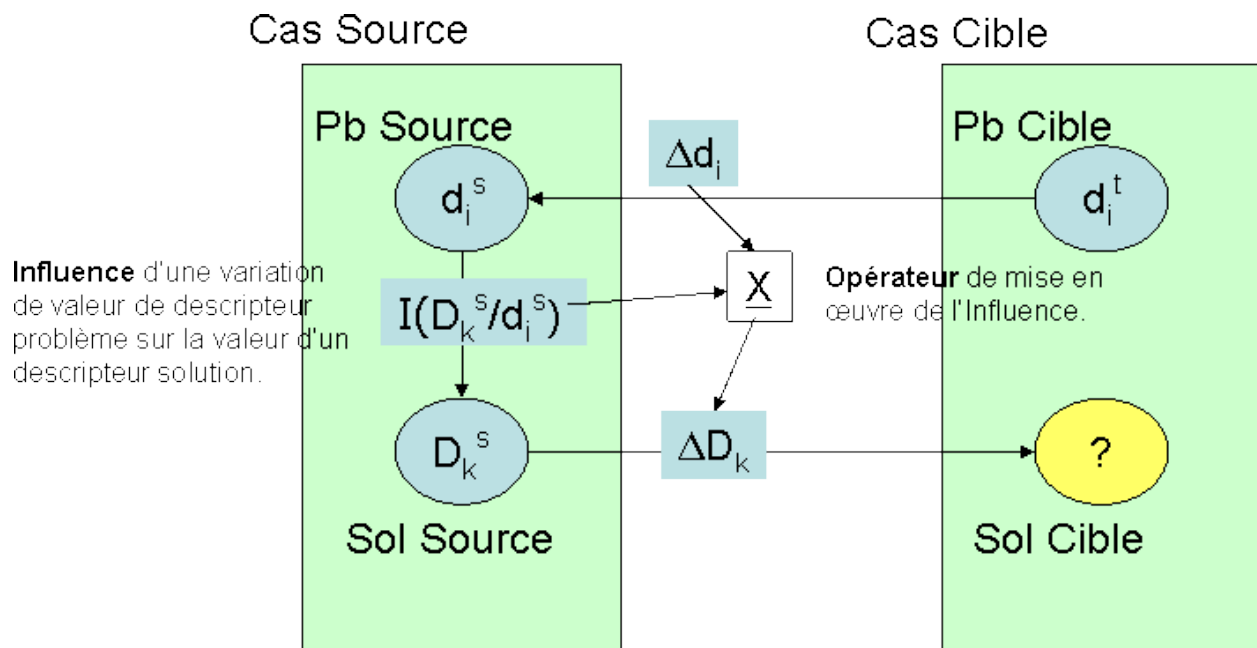
Le cycle du RàPC est repris et appliqué dans plusieurs outils, tels que [Remind](#), ou [plus récemment, jColibri](#).

En 2005 - Ramos et al. publient une revue des travaux « récents » sur le RàPC : *Retrieve Reuse Revise Retain (élaboration, mémorisation, réutilisation, révision, mémorisation)*. Les recherches ont naturellement encore progressé depuis.

En résumé, voici les différentes étapes que nous décrivons en détails plus loin.

- **Élaboration** : il s'agit de passer du problème « vaguement décrit » dans le monde réel à un problème suffisamment bien formalisé pour qu'il soit exploitable par un système informatique.
- **Remémoration** : à partir de la description élaborée dans la phase précédente, rechercher dans la base de cas des problèmes similaires (en utilisant des mesures de similarité).
- **Réutilisation** : appliquer la solution du problème retrouvé au problème en cours de résolution.
- **Révision** : évaluer la solution apportée, et la corriger en cas de résultat insatisfaisant. Le rôle de l'utilisateur (ou d'une forme de feedback de l'environnement) est crucial dans cette étape.
- **Mémorisation** : on met la nouvelle solution dans la base de cas, étendant ainsi sa couverture.

Le carré d'analogie :



Le carré d'analogie est une métaphore du fonctionnement du RàPC et s'appuie sur un postulat de base (énoncé pour la première fois par McSheery) selon lequel « des problèmes similaires sont des problèmes qui ont une solution similaire ».

Dans le carré d'analogie :

β → symbolise la dépendance entre le problème et la solution (exemple : le kilométrage d'une voiture influence son prix de vente).

α → symbolise les différences entre source (cas dont on va s'inspirer) et cible (par exemple : 2000 kilomètres de différence entre les kilométrages de deux voitures aura un impact de $2000 * 0,01€$ sur le prix).

En général, on adjoint une base de connaissances à la base de cas.

Différence entre le problème source et le problème cible.

“Deux clio rouges, exactement identiques sauf leurs kilométrages (une à 15 000 l'autre à 22 000. A priori celle à 15 000 serait vendue plus cher”.

4. Les phases du raisonnement à partir de cas

Phase d'élaboration

On cherche à formaliser le problème. L'enjeu est de taille. Il faut regrouper un maximum d'informations pour faciliter les étapes suivantes.

Phase de remémoration

On recherche un problème similaire dans la base de cas. Un bon problème est évidemment un problème dont la solution va nous permettre de résoudre le problème courant. Pendant cette phase, on utilise des mesures de similarité. L'approche la plus classique consiste à faire une somme pondérée de critères sur les attributs descripteurs de cas, mais il existe des approches plus riches (graphes, approches statistiques, etc.). Bien entendu, l'idéal est de pouvoir s'intéresser à la similarité des solutions, et non à celle des problèmes.

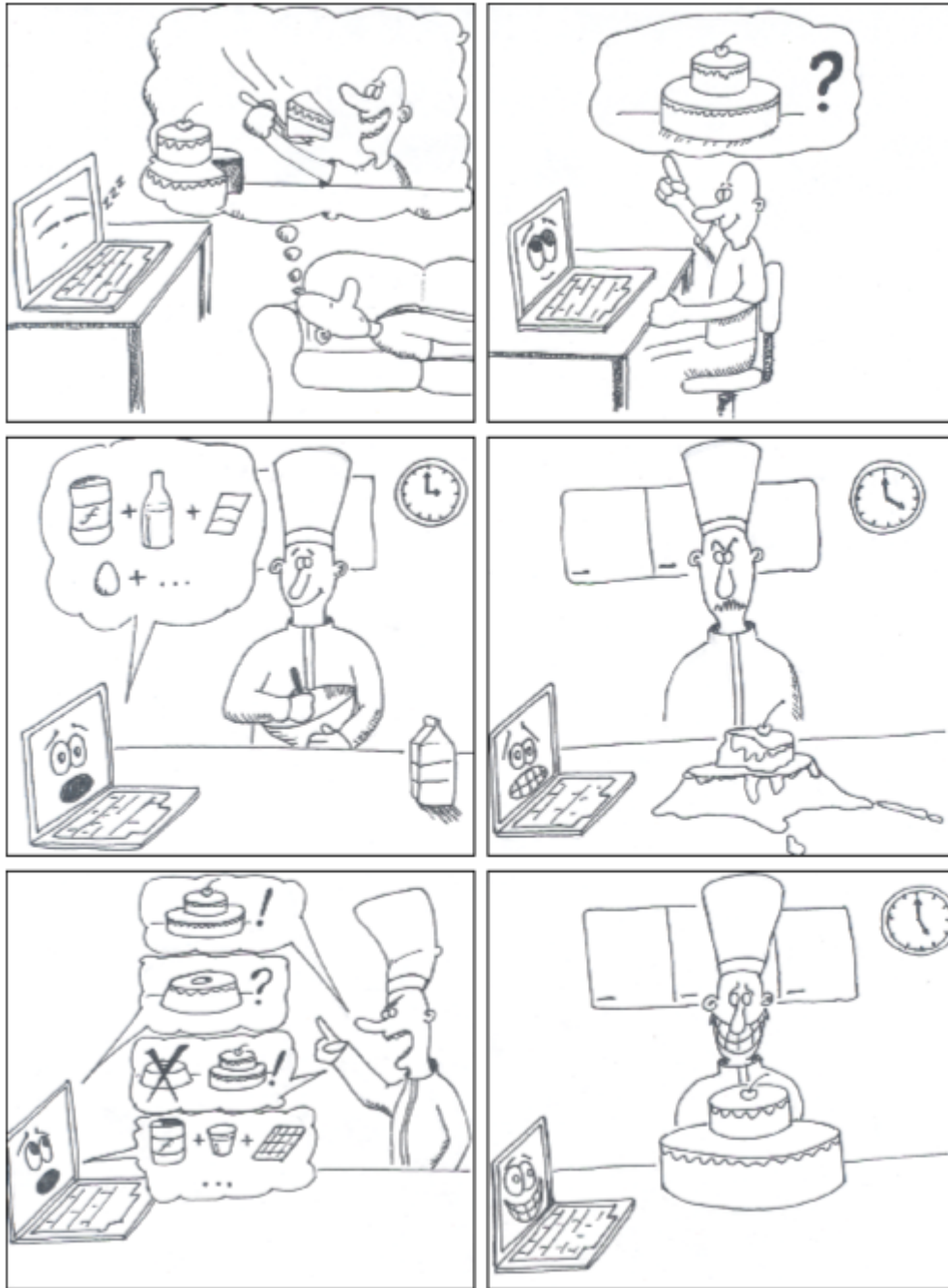
Phase de réutilisation

Cette phase est fréquemment appelée « phase d'adaptation ». Elle consiste à adapter la solution du cas retrouvé pour que cela s'applique au problème en cours de résolution. Il existe plusieurs approches de l'adaptation : adaptation numérique, adaptation-guided retrieval, etc.

L'adaptation-guided retrieval est probablement la sous-partie la plus innovante en RàPC ces dernières années, pour les vrais problèmes de RàPC.

Durant la phase d'adaptation, non seulement on apprend des nouveaux cas, mais également de nouvelles connaissances réutilisables pour l'adaptation (c'est d'ailleurs le grand challenge de la recherche en ce moment).

Exemple :



Phase de révision

La phase de révision est souvent négligée en RàPC. Dans cette phase, on est supposés donner au système du feedback sur la solution (est-ce que ça a fonctionné ? Sinon, pourquoi ? Est-ce que l'on connaît maintenant une solution qui fonctionne ?). Malheureusement, il n'est pas toujours possible de faire cela. En effet, le feedback n'est pas toujours facile à obtenir : soit on ne l'a pas immédiatement, soit on n'a pas de moyen d'évaluer la qualité de la solution.

Exemples de stratégies de révision :

- Humain et réalité (rare) : la solution présentée par la machine est exposée à l'être humain. L'humain teste la solution proposée. Résultat binaire pour que le système corrige les connaissances d'adaptation ayant servi à la construction de la solution (les cas ne sont pas corrigés).
- L'ordinateur teste lui-même la solution (souvent mis en place lorsque cela est possible)

Phase de mémorisation

Cette étape semble triviale (mémoriser un cas et sa solution, c'est simple, une fois que tout le travail est fait). Pour autant, l'étape de mémorisation a donné lieu à de nombreux travaux de recherche : comment optimiser la couverture de la base de cas, comment rendre la base de cas compétente, comment régler le problème de l'oubli pour conserver une base de cas d'une taille raisonnable, etc. Comment mesurer la qualité des cas ? En quoi cela peut-il guider les stratégies de maintenance de la base de cas, etc.

Plusieurs travaux (Smith en particulier) se sont intéressés à cette problématique et ont proposé de nombreuses solutions, notamment l'utilisation de cas prototypiques (cas génériques, construits de toutes pièces à partir de cas existants), ainsi que l'utilisation de classes de cas pour organiser la base de cas et donc accélérer les temps de remémoration.

3. Domaines d'application

- Diagnostic (panne, médical, etc.)
- Recommandations
- Planification (de trajet)
- Médecine
- Formation (recommandation de ressources pédagogiques)
- Automobile
- Cuisine

4. Ressources et liens

http://www-sop.inria.fr/acacia/afia2005/actes/ActesRaPC_Final.pdf

https://www.ibisc.univ-evry.fr/~abchiche/SUPPORTS/M2_ROB/RAPC.pdf

http://liris.cnrs.fr/amille/enseignements/Master_PRO/TIA/RAPC/igc_rapc_Folder/Rapc_Session2_Cas_base_de_cas.html

http://docnum.univ-lorraine.fr/public/SCD_T_2009_0109_BADRA.pdf

<http://liris.cnrs.fr/~rapc/wiki/doku.php/accueil>

<http://www.nextinpact.com/news/101572-partnership-on-ai-cinq-geants-net-unis-pour-promouvoir-intelligence-artificielle.htm>

http://liris.cnrs.fr/amille/enseignements/Master_PRO/TIA/RAPC/igc_rapc_Folder/RAPC_Session1_exercices.html

http://liris.cnrs.fr/amille/enseignements/Master_PRO/TIA/RAPC/igc_rapc_Folder/rapc_session2_exercices.html