



HAL
open science

Vers la généralisation de manipulations distantes et collaboratives d'instruments de haute technologie

Christophe Gravier

► **To cite this version:**

Christophe Gravier. Vers la généralisation de manipulations distantes et collaboratives d'instruments de haute technologie. domain_stic.inge. Université Jean Monnet - Saint-Etienne, 2007. Français. NNT: . tel-00287637

HAL Id: tel-00287637

<https://theses.hal.science/tel-00287637>

Submitted on 12 Jun 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ JEAN MONNET DE SAINT-ÉTIENNE

Discipline : Génie Informatique, Informatique

Présentée et soutenue publiquement le 27 novembre 2007

par

Christophe GRAVIER

Ingénieur ISTASE – Maître ès Sciences

Vers la généralisation de manipulations distantes et collaboratives d'instruments de haute technologie

Directeur de thèse :

Gérard NOYEL

Co-Directeur de thèse :

Jacques FAYOLLE

Composition du Jury :

JEAN-PIERRE ELLOY	Professeur à l'École Centrale Nantes	<i>Rapporteur</i>
PATRICK PRÉVOT	Professeur à l'INSA de Lyon	<i>Rapporteur</i>
BERNARD CORNU	Professeur au CNED	<i>Président</i>
CAROLINE RIZZA	Maître de Conférences à Télécom Paris	
DJAMAL BENSLIMANE	Professeur à l'Université Claude Bernard Lyon 1	
RICHARD CHBEIR	Maître de Conférences à l'Université de Bourgogne	
GÉRARD NOYEL	Professeur à l'Université Jean Monnet	<i>Directeur</i>
JACQUES FAYOLLE	Maître de Conférences à l'Université Jean Monnet	<i>Co-directeur</i>

Remerciements

Ce travail a été réalisé au sein du laboratoire des *Dispositifs et Instrumentation en Opto-électronique et Micro-ondes* (DIOM) de l'université Jean MONNET de Saint-Etienne et a été financé par le Conseil Général de la Loire.

C'est emprunté d'une grande émotion que je tiens à remercier tout ceux qui m'ont entouré pendant ces trois ans. Comme Victor Hugo le disait, « les mots manquent aux émotions », je n'imagine donc pas que mes mots parviendront à exprimer toute ma gratitude : je m'efforcerai seulement de l'exprimer au mieux.

Je souhaite remercier en tout premier lieu Gérard Noyel et Jacques Fayolle qui ont dirigé cette thèse dans la continuité de mon stage de Master Recherche. Merci à eux pour m'avoir fait découvrir la Recherche quotidiennement, tout au long de ces trois années, en orientant mes travaux aux bons moments, tout en tirant partie de ma formation d'ingénieur.

Je suis particulièrement reconnaissant à Jean-Jacques Rousseau, directeur du laboratoire DIOM, pour m'avoir accueilli au sein du laboratoire.

Je n'aurais pas pu mener à bien ces travaux sans les encouragements de Bernard Bayard, Maître de Conférences au laboratoire DIOM, qui n'a jamais refusé de prodiguer ses précieux conseils, appuyés de discussions stimulantes.

Je désire sincèrement remercier Jean-Pierre Elloy et Patrick Prévost pour me faire l'honneur de montrer le vif intérêt qu'ils portent à ce travail en acceptant spontanément d'être rapporteur.

Mes pensées vont aussi à Bernard Cornu, Caroline Rizza, Djamel Benslimane et Richard Chbeir pour avoir accepté de participer au jury de mon travail, ainsi que pour les

discussions passionnantes qui ont pu avoir lieu ce jour là grâce à la pertinence de leur questions.

Je ne peux oublier l'aide précieuse de mes collègues de travail, Jérémy Lardon et Mikaël Ates. Je les remercie également vivement pour le temps qu'ils ont pu consacrer à redonner un peu de rigueur à ma plume ainsi que pour les très bons moments passés ensemble.

Mes remerciements incluent également mes amis qui ont partagé mes moments de détente, complément obligatoire au bon déroulement de cette thèse : Anne-Sophie, Blandine, Christophe, Claire, Frédéric et Liam. Je les remercie chaleureusement d'autant que certains sont allés jusqu'à se plier au fastidieux exercice de la chasse aux fautes d'orthographe.

Je remercie Thomas Rouiller pour l'incroyable patience et disponibilité dont il a fait preuve lors de mon initiation aux principes et au fonctionnement de l'analyseur vectoriel de réseaux. Merci aussi à Nadège Ollier et Éric Verney qui ont très spontanément accepté de pratiquer des téléTPs lors de leurs séances d'enseignement de travaux pratiques, et pour la grande confiance et détermination dont ils ont fait preuve dans la mise en place de ces séances.

Un grand MERCI aux collègues du laboratoire que j'ai pu cotoyer, à la machine à café ou au déjeuner, pour les bons moments de détente et l'atmosphère très conviviale qui règne au laboratoire. Merci donc à Alain (les deux!), Bruno, Damien, Didier, Évangéline, François, Harouna, Isabelle, Jean Pierre (notre maître ès café, entre autres talents) et bien sûr Stéphane. Je remercie les doctorants du DIOM pour le moment de sport que nous attendions, toujours impatiemment, tous les jeudis soirs.

Je voudrais également souligner le travail et l'enthousiasme des stagiaires que j'ai pu encadrer pendant ces trois ans : Vo Nhat Vinh, Ousmane Farikou et Philippe Oliveira Da Silva.

Je remercie également Luc Jorat, directeur des études, ainsi qu'Alain Mure-Ravaud, responsable des relations internationales, pour la confiance qu'ils m'ont accordée dans le cadre des enseignements que j'ai assuré pour les élèves ingénieurs de l'Istase.

Je n'oublierai pas les aides reçues du personnel administratif de l'Istase : Alice Nepveu-Barrieux, Pascal Grail, Véronique Maritan, Maryse Martinet, Sophie Pinatel, Françoise Pompey, Sandrine Rasclé et Gaétane Zummo, dont le tiramisu n'a d'égal que le clafouti.

Je tiens également à remercier Christian Forissier pour sa bonne humeur et sa disponibilité, durant le temps que nous avons travaillé ensemble sur le réseau informatique de l'école. Merci aussi à Jérôme Da Rugna, maître de conférence à l'Istase, pour son aide et son soutien dans mes tâches d'administration des serveurs de l'école.

Enfin, je ne pourrais terminer mes remerciements sans m'adresser à ma petite famille, qui me soutient et me supporte avec amour au quotidien, à savoir Christelle, mon épouse, et mes deux fils, Lucas et Paul.

Table des matières

Remerciements	3
Introduction	17
I Vers des télé-opérations collaboratives et réutilisables	21
Introduction	23
I.1 Enjeux de la télé-manipulation de dispositifs	24
I.1.1 Généralités	24
I.1.2 Pourquoi piloter à distance	25
I.1.3 Usages associés	26
I.1.4 Bilan	35
I.2 La formation pratique à distance	36
I.2.1 Théories de l'apprentissage et travaux pratiques à distance	36
I.2.2 Les télé-TPs	39
I.2.3 Apprentissage Collaboratif Assisté par Ordinateur	46
I.2.4 Bilan	52
I.3 Etat de l'art	56
I.3.1 Projets existants	56
I.3.2 Etude bibliographique	57
I.3.3 Bilan	65
Conclusion	69
II Généraliser les dispositifs en ligne	71
Introduction	73
II.1 Généraliser la chaîne logicielle de télé-TPs	74
II.1.1 Maillon entre ordinateur local et instrument	74
II.1.2 Maillon entre ordinateur distant et ordinateur local	76
II.1.3 Interface Homme Machine	78
II.1.4 Bilan	79

II.2 IHM et connectivité à l'intergiciel	80
II.2.1 Ontologies et Génie Logiciel	80
II.2.2 Formaliser l'IHM et la connectivité à l'intergiciel	87
II.2.3 Limitations actuelles	104
II.2.4 Bilan	108
II.3 Implémentation de la plate-forme et de l'intergiciel	109
II.3.1 Généraliser le lien entre IHM et l'intergiciel	109
II.3.2 Protocole applicatif pour l'architecture distribuée	112
II.3.3 Choix de l'intergiciel	115
II.3.4 Serveur d'applications pour les composants métiers	120
II.3.5 Points remarquables de l'implémentation	124
Conclusion	129
III Orchestrer l'accès au dispositif _____	131
Introduction	133
III.1 Éléments de contexte	134
III.1.1 Le besoin de régulation des accès collaboratifs	134
III.1.2 Représentation des éléments issus de la collaboration	136
III.1.3 Évaluation de l'approche	139
III.1.4 Bilan	142
III.2 Politique de collaboration	145
III.2.1 Un ensemble de règles comme politique de partage	145
III.2.2 Exprimer les politiques d'accès avec SWRL	147
III.2.3 Autres exemples de politiques d'accès	148
III.2.4 Bilan	152
III.3 Implémentation du système d'orchestration	153
III.3.1 Architecture complète	153
III.3.2 De l'importance de pouvoir modifier la politique en vigueur	156
III.3.3 Couplage de l'unité de modération avec la plate-forme de télé-TP	159
III.3.4 Problèmes d'implémentation rencontrés	160
III.3.5 Bilan	165
Conclusion	167
IV Évaluation et devenir des télé-TPs _____	169
Introduction	171

IV.1	Évaluation de la plate-forme de télé-TP	172
	IV.1.1 Conditions d'évaluation	172
	IV.1.2 Évaluation de l'interface homme machine de l'analyseur	175
	IV.1.3 Enquête de satisfaction	177
	IV.1.4 Bilan	180
IV.2	Perspectives	181
	IV.2.1 Devenir de la plate-forme	181
	IV.2.2 Défis à venir pour les télé-TPs	183
	Conclusion	189
Conclusion _____		191
Annexes _____		195
A	Détail de la répartition de notre bibliographie _____	197
B	Extrait des spécifications de OWL _____	201
B.1	Property Characteristics	202
	B.1.1 TransitiveProperty	202
	B.1.2 SymmetricProperty	202
	B.1.3 FunctionalProperty	203
	B.1.4 inverseOf	203
	B.1.5 InverseFunctionalProperty	204
B.2	Property Restrictions	204
	B.2.1 allValuesFrom, someValuesFrom	205
	B.2.2 Cardinality	206
	B.2.3 hasValue [OWL DL]	206
C	Travaux pratiques sur l'analyseur vectoriel de réseaux _	209
	C.1 Sujet du travail pratique	210
	C.2 Questionnaire distribué aux étudiants	213
	C.3 Dépouillement du questionnaire	220
Bibliographie _____		229

Table des figures

I.1	Le trèfle du pilotage distant de dispositifs : des enjeux et des acteurs pluriels	27
I.2	Des EOA aux EIAH en passant par les EIAO puis STI : évolution de l'usage de l'ordinateur dans l'enseignement de 1960 jusqu'à nos jours. . .	32
I.3	Enseignement assisté par ordinateur et théories d'apprentissage.	38
I.4	Taxonomie des travaux pratiques assistés par ordinateur.	39
I.5	Les laboratoires locaux.	40
I.6	Les laboratoires locaux assistés par ordinateur.	40
I.7	Les laboratoires distants.	41
I.8	Les laboratoires virtuels.	42
I.9	Les laboratoires hybrides.	43
I.10	Les apprenants communiquent entre eux autour du problème posé par le formateur, d'après (Akkouche, 1996).	48
I.11	Positionnement de notre travail : l'expérimentation collaborative assistée par l'ordinateur (ExCAO).	54
I.12	Le tryptique de l'ExCAO : une pluridisciplinarité indéniable.	54
I.13	Fenêtre temporelle de notre étude : répartition par année des publications utilisées.	58
I.14	Répartition par origine géographique des publications.	59
I.15	Répartition par discipline scientifique du dispositif contrôlé à distance. . .	60
I.16	Répartition par solution logicielle du dispositif contrôlé à distance.	62
I.17	Les outils de communications utilisés pour permettre le dialogue entre apprenants.	64
II.1	Réutilisabilité des maillons de la chaîne logicielle des télé-TPs.	74
II.2	Modèle en trois couches du web sémantique : sérialisation (syntaxe), description (données), logique.	83
II.3	Pile du Web Sémantique (Semantic Web Stack), d'après Tim Berners-Lee du W3C (Berners-Lee, 2000).	84
II.4	Un instrument fictif pour illustration de notre démarche.	88

II.5	Première vision de l'IHM d'un instrument à partir de l'instrument fictif de la figure II.4	89
II.6	Exemples d'instruments de télécommunications, électronique et micro-électronique directement visés par notre ontologie	90
II.7	D'autres domaines scientifiques nécessitent des instruments à façade. Exemple ici en chimie avec un détecteur à chimiluminescence.	90
II.8	Ontologie d'une façade deux dimensions d'un instrument de mesure. En jaune les classes primitives, en orange les classes définies.	91
II.9	Relations entre concepts de notre ontologie : filiations et relations sémantiques.	92
II.10	L'ontologie « mère » est importée par chacune des ontologies des façades de dispositif.	95
II.11	L'atelier d'édition de la façade d'un instrument.	97
II.12	La chaîne d'édition d'une ontologie d'une façade d'instrument.	99
II.13	Analyseur vectoriel de réseaux <i>Anritsu 37397A</i> utilisé au laboratoire DIOM.	100
II.14	Testeur sous pointe permettant la caractérisation d'un composant au laboratoire DIOM.	101
II.15	Banc de mesure complet (AVR et DST) du laboratoire.	101
II.16	Application résultante, permettant d'utiliser l'AVR Anritsu à distance, de manière réaliste : fidélité de la représentation et utilisation (écran tactile).	102
II.17	Le principe du « WebSwitch » : un sélecteur d'entrées/sorties pour éviter de multiplier les connexions et déconnexions durant une session de télé-TP.	103
II.18	Le « WebSwitch » : photographie d'un dispositif « fait maison ». Schéma de principe à la figure II.17.	103
II.19	Instrument fictif modifié : ajout du bouton OK : fin de la bijection widget actionné / fonctionnalité identifiée.	104
II.20	Aiguillage classique des évènements dans une application.	111
II.21	Aiguillage dans la plate-forme de télé-TPs : les widgets sérialisent les évènements eux même sur l'intergiciel.	112
II.22	Cycle de vie des messages sur l'intergiciel.	113
II.23	Diagramme de séquence au format UML d'une phase « commande-acquittement-réponse ».	115
II.24	Remote Procedure Call	116
II.25	Fonctionnement des OOM (Object Oriented Middleware).	117
II.26	Fonctionnement des MOM (Message Oriented Middleware).	118
II.27	Utilisation des MOM dans la plate-forme de télé-TPs	120
II.28	Envoi d'une commande sur la plate-forme de télé-TPs.	123

II.29 Retour des messages d'acquiescement et résultat de la commande sur la plate-forme de télé-TPs.	123
II.30 Indicateur de téléprésence développé pour la plate-forme.	124
II.31 L'étudiant peut appauvrir ou enrichir la représentation de la façade suivant le degré de compétence qu'il estime être le sien (voir la figure II.8 pour le concept Level).	125
III.1 Ontologie des éléments de contexte propres à une situation de télé-TP collaboratif.	136
III.2 L'arbre de la règle « admin only » qui donne la main à l'administrateur qui la demande.	147
III.3 Implémentation du SBC orchestrant les accès dans les séances de télé-TP collaboratifs.	154
III.4 La politique d'orchestration peut être modifiée au cours du télé-TP pour satisfaire l'objectif pédagogique du moment.	158
III.5 Couplage asynchrone entre notre SBC et la plate-forme de télé-TP.	159
IV.1 La carte embarquant les composants à tester par les étudiants pour le télé-TP.	172
IV.2 Le laboratoire distant avec PC local, analyseur vectoriel de réseau et composants à caractériser.	173
IV.3 Paillasse proposée aux étudiants en salle de TP à la place de la présence physique de l'instrument.	173
IV.4 Visualisation du dépouillement par heuristiques: Comment les étudiants perçoivent-ils l'ergonomie de l'appareil réel?	176
IV.5 Répartition des réponses des étudiants à la question: « Pour vous, quel est l'intérêt de pouvoir dialoguer avec les autres personnes de la manipulation à travers la fenêtre de discussion? » (question 4)	178
IV.6 Appareil utilisé pour le télé-TP physico-chimique.	182
IV.7 Transparence de localisation des instruments d'un banc de mesure.	184
IV.8 Transparence de localisation des dispositifs associé au principe de basculement des télé-TPs d'un banc de mesure à un autre.	185
IV.9 La convergence télé-TPs et LMS pour une gestion des télé-TPs dans les scénarii d'apprentissage en ligne.	188

Liste des tableaux

I.1	Forces et faiblesses des manipulations distantes et virtuelles.	44
I.2	Les différentes configurations d'ACAO par le couplage espace et temps .	49
I.3	Les principaux projets d'ExAO recensés au niveau mondial.	56
II.1	Concepts hébergés dans notre ontologie des façades d'instruments.	93
II.2	Propriétés liant sémantiquement les concepts de la façade entre eux. . . .	94
II.3	Différence entre la structure d'un programme se connectant à un intergiciel synchrone et celle se connectant à un intergiciel asynchrone.	119
III.1	Concepts pris en compte dans une stratégie de partage d'un appareil im- pliqué dans un télé-TP.	138
III.2	Relations entre les éléments de contexte qui déterminent une session de télé-TP.	140
III.3	Évaluation de notre ontologie à la lumière des critères de Grüber	143
III.4	Politiques de partage: exemples d'autres orchestrations possibles utilisant notre modèle (tableau non exhaustif).	151
IV.1	Résultats par heuristique concernant l'évaluation de l'IHM de l'analyseur vectoriel de réseau Anritsu	176

Introduction

La « société de l'information », souvent citée comme une troisième révolution industrielle, repose sur l'émergence des Sciences et Technologies de l'Information et de la Communication. Pour la France, c'est un secteur important en terme d'emplois (environ 668 000 informaticiens dans l'Hexagone en 2006). De plus, l'information tend à être accessible partout (55 % des foyers français sont équipés d'un ordinateur, d'après le portail de la société de l'information de l'État¹). L'État français contribue d'ailleurs à cet élan via, par exemple, le programme ADELE (ADministration ELEctronique 2004/2007) piloté par l'ADAE (Agence pour le Développement de l'Administration Électronique).

Il s'agit également d'une considération économique puisque les industriels sont des acteurs majeurs de l'ère du numérique. En effet, dans un contexte de concurrence internationale forte, les entreprises n'hésitent pas à externaliser leurs activités dans des sites hors du territoire. On parle alors d'« entreprise étendue ». De ce fait, alors que les systèmes informatiques ont longtemps été cantonnés à des fonctions de calculs et de stockage de données, ils sont aujourd'hui l'outil incontournable pour assurer la communication et le pilotage de l'entreprise.

Les industries font alors face à un problème épineux : il est plutôt simple de partager de l'information, mais il est beaucoup plus ardu de piloter des procédés de production à distance. En effet, les parcs de dispositifs de production sont hétérogènes, tant du point de vue de leurs interfaces que de leur commandabilité. Cela ne facilite pas la mise en place de standards. Par ailleurs, le flux vidéo qui permettrait d'obtenir un retour visuel pour des interventions à distance est impossible en pratique car la bande passante nécessaire pour un tel outil est rarement disponible dans les sites décentralisés de production.

Le contrôle de procédés distants est un enjeu socio-économique encore plus large puisqu'il relève également de problématiques d'enseignement pour les instituts supérieurs de formations. À travers divers appels à projets, nationaux ou régionaux, les écoles d'ingénieurs ont développé des campus numériques, qui permettent de former des élèves ingénieurs à distance. À l'ISTASE², cela s'est traduit par la mise en place d'un campus

1. <http://www.internet.gouv.fr>

2. Institut Supérieur des Techniques Avancées de Saint-Etienne

numérique nommé « Eden3 » (Enseignement à Distance en Ecole d'ingénieurs en 3^{ème} année). Là aussi au sein d'un contexte concurrentiel fort, les écoles d'ingénieurs se doivent de proposer des catalogues de formations de plus en plus riches. Ces offres concernent les enseignements traditionnels, comme les cours magistraux et les travaux dirigés, sans ignorer pour autant les mutations pédagogiques actuelles, en particulier la résolution de problèmes accompagnés de cas. Ces mêmes catalogues s'inscrivent également dans les démarches de VAE³ ou encore de DIF⁴. Les enseignements pratiques sont néanmoins de grands absents de ces dispositifs, car il n'est pas aisé de construire un système hébergeant des travaux pratiques à distance : c'est souvent très coûteux et éloigné de l'expérience que pourrait vivre un apprenant face à l'instrument réel.

Que ce soit dans le cadre industriel ou celui de la formation, la télé-opération de dispositifs demeure donc une problématique sous-explorée. En effet, il s'agit d'un sujet pluridisciplinaire et transversal, faisant appel à de nombreuses thématiques pour construire une chaîne de manipulations du dispositif contrôlé, jusqu'à l'application distante pilotant l'instrument. Il convient alors d'avoir des compétences sur le matériel exploité à distance (électronique, mécanique, chimie, physique, ...), mais aussi en informatique et en génie logiciel pour formaliser le système de communication et le mettre en place. Sans oublier les domaines des réseaux et de la sécurité globale, car les appareils sont onéreux et les données souvent confidentielles.

Le laboratoire des Dispositifs et Instrumentation en Optoélectronique et Micro-ondes (DIOM) de l'Université de Saint-Etienne explore ce vaste problème. C'est à partir d'une expérience dans ce domaine, via le projet « WebAnalyzer » (Bayard *et al.*, 2001), que nous avons structuré notre réflexion. Nous avons travaillé sur l'élaboration d'outils indépendants de l'instrument piloté, dans un contexte collaboratif. Envisager un passage au travail collaboratif sur tout dispositif technologique évolué, implique inévitablement une nouvelle approche logicielle indépendante du type de matériel. C'est l'objet de ce travail, dont un des objectifs est de créer un prototype permettant de déployer de nouveaux démonstrateurs avec le minimum de développements logiciels requis. Cette approche est une avancée dans la problématique du pilotage industriel distant, et plus fondamentalement dans les interfaces homme-machine et des ontologies logicielles. Ce mémoire reflète les tenants et aboutissants d'une telle réflexion et vise ainsi à asseoir les connaissances et compétences du laboratoire DIOM dans ce domaine.

Le premier chapitre de ce mémoire sera consacré aux généralités et à l'étude bibliographique. Nous présenterons d'abord les enjeux du pilotage distant partagé, pour les

3. Validation des Acquis par l'Expérience

4. Droit Individuel à la Formation

laboratoires, les industries et les instituts supérieurs de formation. Une étude bibliographique viendra ensuite compléter cette vision, permettant de mettre en perspective les objectifs de la télé-opération avec l'existant.

Le deuxième chapitre traitera des possibilités de factorisation des développements logiciels de chaque maillon composant la chaîne logicielle d'une manipulation distante. Cette présentation sera associée au compte rendu de l'implémentation d'un prototype, qui permet de minimiser les développements logiciels pour la mise en ligne de formations pratiques à distance et collaboratives.

Ensuite, le troisième chapitre soulèvera et proposera des solutions au problème de la collaboration au sein de travaux techniques partagés. En effet, comme plusieurs utilisateurs sont susceptibles de piloter en même temps un unique dispositif, il convient de mettre en place une certaine orchestration de la collaboration en vue de maximiser l'expérience d'apprentissage. Notre objectif sera alors de formaliser les situations de collaborations pour permettre de modéliser les orchestrations adaptées à chaque besoin.

Pour conclure, le chapitre quatre mettra en évidence les tests du prototype réalisé auprès d'étudiants dans le cadre de formations pratiques, ainsi que son devenir. Enfin, nous évoquerons les nombreux défis à venir pour les travaux pratiques à distance.

Vers des télé-opérations collaboratives et réutilisables

Sommaire

Introduction	23
I.1 Enjeux de la télé-manipulation de dispositifs	24
I.1.1 Généralités	24
I.1.2 Pourquoi piloter à distance	25
I.1.2.1 Le facteur financier	25
I.1.2.2 S'affranchir de contraintes	26
I.1.3 Usages associés	26
I.1.3.1 Les collaboratoires	27
I.1.3.2 Les contrôles de procédés industriels	30
I.1.3.3 La formation pratique à distance	31
I.1.4 Bilan	35
I.2 La formation pratique à distance	36
I.2.1 Théories de l'apprentissage et travaux pratiques à distance	36
I.2.1.1 Précisions linguistiques	36
I.2.1.2 Apprendre à distance par la pratique	37
I.2.2 Les télé-TPs	39
I.2.2.1 Taxonomie des travaux pratiques assistés par ordinateur.	39
I.2.2.2 Télé-TPs réels et télé-TPs virtuels: que choisir?	43
I.2.3 Apprentissage Collaboratif Assisté par Ordinateur	46
I.2.3.1 Définitions	47
I.2.3.2 ACAO: Pourquoi et Comment.	50
I.2.4 Bilan	52
I.3 Etat de l'art	56
I.3.1 Projets existants	56
I.3.2 Etude bibliographique	57
I.3.2.1 Cadre de l'étude	57

I.3.2.2	Analyse	58
I.3.3	Bilan	65
Conclusion	69

CHAPITRE I

Vers des télé-opérations collaboratives et réutilisables

Introduction

Ce premier chapitre posera les bases de compréhension du travail présenté dans ce mémoire. Tout d'abord, nous présenterons les tenants et aboutissants du contrôle distant de ressources. Ensuite, nous donnerons les définitions et les caractéristiques du pilotage distant de ressources à finalité d'apprentissage. Cet état de l'art dressera les grandes tendances passées et actuelles de ce type de procédés. Volontairement restreinte aux situations de formations ouvertes et à distance (FOAD), puisqu'il s'agit de la toile de fond utilisée tout au long de notre travail, cette étude bibliographique mettra en évidence les forces et faiblesses actuelles de ces systèmes. Cette analyse justifiera alors nos problématiques de recherche et les enjeux sous-jacents dans le domaine des Environnements Informatiques pour l'Apprentissage Humain (EIAH).

I.1 Enjeux de la télé-manipulation de dispositifs

I.1.1 Généralités

Depuis ses débuts, l'outil informatique n'a cessé de révolutionner notre façon de travailler et de communiquer. L'outil informatique a en effet le don de nous conduire vers de nouveaux comportements. Une nouvelle fois aujourd'hui, l'informatique démontre son aptitude à transformer, voire à métamorphoser un domaine, notamment celui du contrôle de dispositifs de haute technologie. En effet, l'ordinateur rend possible le pilotage local d'une machine. Cet usage, associé à celui d'Internet, permet d'envisager cette interaction à distance.

Que ce soit dans le cas des pilotages réels de dispositifs ou de simulations, les relations entre hommes et machines ont évolué. Tout d'abord, le contact n'est plus physique mais s'effectue via le média informatique (nous sommes « assistés par l'ordinateur »).

Cette séparation est intrinsèquement vérifiée dans le cas où les personnes et les machines ne se trouvent plus au même endroit, voire au même moment. Elle l'est aussi dans le cas des simulations, car les machines sont « virtualisées » ou « digitalisées » au profit d'algorithmes émulant leur comportement, qui correspond à un modèle plus ou moins précis et validés, préalablement établi, minimisant les risques technologiques. Nous pouvons citer l'exemple des simulateurs de vol qui présentent une imitation de la réalité gérée par l'ordinateur. Le dispositif piloté est de plus en plus transparent pour l'homme. La relation entre homme et machine est donc directement tributaire de la performance du média informatique qui assiste la manipulation.

Au delà de l'aspect potentiellement intrusif de l'ordinateur, il est important de comprendre que l'innovation n'est pas seulement technique (Blanc, 2004). Nous entendons que l'informatique permet d'associer plusieurs personnes à plusieurs machines, à l'inverse de la relation bijective qui pouvait exister auparavant. En effet, la relation quasi exclusive qui caractérisait la perception que nous avions des systèmes automatisés est maintenant beaucoup plus ouverte à l'interopérabilité. Il s'agit de mettre en relation des procédés hétérogènes mais également des collaborations entre des personnes, dans l'optique d'atteindre un objectif partagé. On désigne ce domaine scientifique par Travail Collaboratif Assisté par Ordinateur (TCAO¹). Le TCAO recouvre les « études des outils et techniques des collecticiels ainsi que leurs effets sociaux, psychologiques et organisationnels »,

1. Computer Supported Collaborativer Work (CSCW) en anglais.

selon Wikipédia².

Que ce soit par la remise en question des interactions entre homme et machine, à travers la socialisation des dispositifs par l'interopérabilité des procédés ou encore par le TCAO, les enjeux du pilotage d'instruments distants sont pluriels, et les applications nombreuses. Afin d'illustrer les finalités possibles de ce travail, nous allons nous attacher à décrire les enjeux qui poussent au développement du pilotage distant de ressources.

I.1.2 Pourquoi piloter à distance

I.1.2.1 Le facteur financier

L'achat d'instruments de haute technologie présente bien souvent l'inconvénient d'un coût élevé par rapport à sa fréquence d'utilisation. Il impose également la présence d'un personnel qualifié sur site, disposant de l'expertise suffisante pour assurer une gestion autonome de son parc de machines. L'alternative, consistant à déplacer le dispositif, implique des difficultés techniques ainsi que des problèmes de responsabilité (assurance par exemple) ou encore de sécurité, sans parler des risques liés au transport. La solution couramment utilisée consiste donc à déplacer les utilisateurs sur le site où se trouve la plate-forme technologique (par exemple les industriels en cas de prestation de recherche ou encore les étudiants dans le cadre d'apprentissage).

Cela présente cependant des problèmes pratiques (par exemple les demandes administratives d'accès aux bâtiments hébergeant les plates-formes technologiques), on imagine facilement que le pilotage d'une ressource à distance offre un gain financier : coûts directs (déplacement, hébergement, ...) mais aussi des coûts masqués (temps principalement). Il permet également d'économiser les efforts investis dans les démarches administratives pour les transformer en travail directement productif. Évidemment, ces gains découlent de l'abolition des distances impliquées par la télé-opération, mais demandent des efforts conséquents en termes de développements technologiques.

Dans notre travail, les instruments concernés correspondent essentiellement aux appareils des laboratoires de recherche dont les caractéristiques communes sont des coûts d'achat et d'entretien élevés, un encombrement important et une relative fragilité de fonctionnement et de transport. Un analyseur vectoriel de réseaux dont nous sommes équipés au laboratoire DIOM constitue un bon exemple. Cet appareil de caractérisation électromagnétique travaille dans une bande de fréquence allant jusqu'à des dizaines de gigahertz (domaine des hyperfréquences). Cet appareil de précision est coûteux et com-

2. <http://fr.wikipedia.org/wiki/TCAO>

plexe en terme d'utilisation (son coût de 150 k€ à l'achat justifie à lui de se pencher sur la faisabilité de télé-opérations à distance). Ainsi peut-il constituer un bon démonstrateur d'outils de production évolués. La mutualisation de plates-formes technologiques industrielles est en effet maintenant à l'ordre du jour.

I.1.2.2 S'affranchir de contraintes

Cependant, le pilotage distant de dispositifs **ne se limite pas à une nécessité économique**.

D'autres facteurs conduisent à la mise en place de télé-opérations. Le problème posé par le déplacement des personnes ou des dispositifs est fondamental. Les locaux des laboratoires ne sont pas toujours adaptés à la réception de tels groupes (Bayard *et al.*, 2001, 2004 ; Taboy, 2006). Les incapacités à conduire des manipulations locales sont la conséquence de diverses difficultés :

- *Sécurité* des personnes et des biens, respect des contrats d'assurance contractés par le laboratoire ou encore du règlement intérieur du laboratoire,
- *Observabilité* de la manipulation dans le cas où le nombre de personnes ne permet plus d'appréhender la manipulation *in situ* correctement,
- *Dangerosité* de la manipulation qui pousse à mettre de la distance entre les personnes et le lieu d'expérimentation,
- *Accessibilité* des personnes souffrant d'un handicap physique aux locaux hébergeant la manipulation (Colwell *et al.*, 2002),
- *Disponibilité* des personnes.

Ainsi, le pilotage distant ne se borne pas seulement à résoudre des considérations financières, et cette liste non exhaustive d'incapacités illustre bien d'autres aspects justifiant le pilotage à distance. De plus, il présente d'autres avantages, comme la facilité de répétition, ce qui peut apporter un intérêt dans le cadre d'une exploitation statistique, ou encore une certaine liberté dans l'organisation de la charge de travail des personnels.

I.1.3 Usages associés

Les télé-opérations permettent souvent de mener des manipulations qui n'auraient pas pu avoir lieu autrement, indépendamment des ressources financières considérées. Bien sûr, les tenants et aboutissants de tels procédés concernent un large panel d'acteurs, parties prenantes dans la réalisation de plates-formes technologiques pilotées à distance. Ces entités sont toutes liées par des relations différentes, que l'on trouve résumées dans la figure I.1.

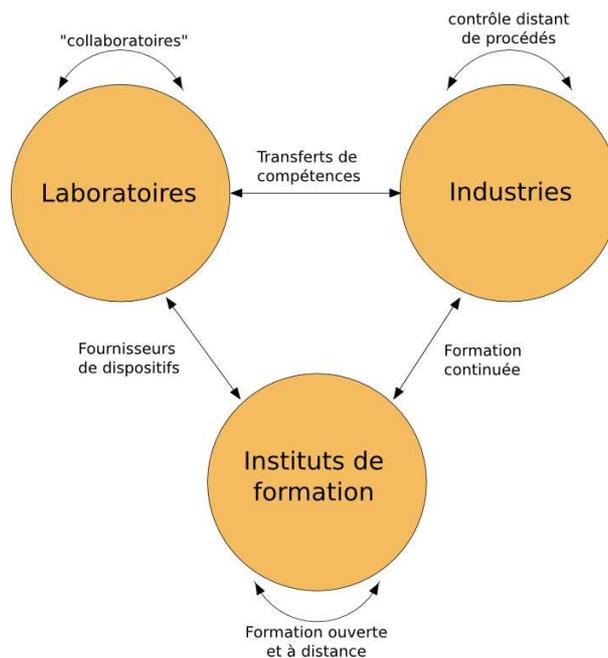


FIG. I.1 – *Le trèfle du pilotage distant de dispositifs : des enjeux et des acteurs pluriels*

Nous allons ici détailler les enjeux perçus par chaque entité, ainsi que les motifs qui poussent celles-ci à entretenir le développement du pilotage distant de dispositifs.

I.1.3.1 Les collaboratoires

... ou la diminution des distances entre les chercheurs

Le terme collaboratoire correspond à la contraction des mots collaboration et laboratoire³. La définition originelle, perdurant aujourd'hui, définit un collaboratoire comme suit :

« [a] center without walls, in which the nation's researchers can perform their research without regard to geographical location, interacting with colleagues, accessing instrumentation, sharing data and computational resources, and accessing information in digital libraries. » (Cerf, 1993)

Abolir les distances entre les chercheurs, pour l'échange d'idées, favoriser la dissémination des résultats ou encore l'événementiel, nécessitent des outils de communication. Des outils informatisés tels que la messagerie électronique, Internet et les portails de partage de l'information, créés à la fin des années 1990, ont permis de renforcer l'émergence de groupes de recherche géographiquement éloignés.

3. En fait le terme collaboratoire dont nous faisons état ici est la traduction de l'anglicisme « collaboratories » proposé par William Wulf en 1989 dans un rapport non publié (Wulf, 1989) et repris sous la forme d'une publication en 1993 (Cerf, 1993) et plus tard Par Finholt (Finholt & Olson, 1997)

Les universitaires ne se sont d'ailleurs pas trompés à ce sujet, et ont engagé des réflexions en ce sens. Les universités sont conscientes des bouleversements que peuvent induire ces nouvelles formes d'organisation (Finholt & Olson, 1997). Sur un plan pratique, il est vrai que, par exemple, les portails de dissémination de données (mise à disposition des versions électroniques de publications scientifiques) ont révolutionné les modes de communication autour des résultats scientifiques.

Il ne faut surtout pas limiter les collaboratoires aux outils de communication entre chercheurs. Les collaboratoires recouvrent également les outils *collaboratifs* de production scientifique, et interviennent, non seulement lors de la publication des résultats, mais également durant leur construction. D'ailleurs, en 1996, Kouzes (Kouzes *et al.*, 1996) souligne que les collaboratoires se doivent :

- d'être socialement acceptés par les chercheurs pour être efficaces,
- d'être robustes pour limiter les frustrations et le rejet technologique,
- d'utiliser des outils déjà existants (comme l'exemple de la messagerie électronique cité plus haut), mais également des outils qui restent à être imaginés !

Le rôle joué par la télé-opération dans les collaboratoires

Après dix ans de développement de collaboratoires et des outils inhérents, d'autres changements peuvent être soulignés.

Tout d'abord, les problématiques scientifiques sont de plus en plus interdisciplinaires. Les chercheurs sont de plus en plus contraints à consulter les secteurs scientifiques voisins, pour pouvoir comprendre les problèmes posés. Cette rupture culturelle passe ainsi par la mise en réseau de chercheurs sur des thématiques connexes, et bien souvent, en pratique, par la mise en place de collaboratoires (ou groupes de recherche).

Les chercheurs éprouvent le besoin de partager leurs connaissances, savoir-faire et résultats. Dès lors, des efforts conséquents se sont portés sur les commodités de recherche des données scientifiques (moteurs de recherche pour les publications basés sur l'indexation du contenu, les méta-données, ou encore la recherche sémantique). Toutefois, dix ans plus tard, on s'aperçoit que les outils informatiques n'ont pas changé et qu'en définitive, ils se sont bornés à faire évoluer les outils déjà existants.

Nous pensons que la télé-opération constitue véritablement un nouvel outil favorisant les échanges au sein des collaboratoires. Même si des tentatives sporadiques ont eu lieu (Agarwal *et al.*, 1998), elles n'ont pas réellement été suivies, à cause des limitations des solutions existantes que nous exposerons plus tard (§I.3.2). Ainsi, le pilotage de ressources

reste une question ouverte pour les laboratoires, dans le cadre de leur réflexion autour des collaboratoires. Un des buts du travail présenté ici est de contribuer à l'avènement de cet outil pour l'expérimentation distribuée au sein de collaboratoires. En effet, nous pouvons imaginer tous les usages qui en découlent :

- échanges de savoir-faire (formations entre chercheurs) : dans le cadre de « transferts de connaissances »⁴ d'un laboratoire à l'autre mais aussi dans le cas de chercheurs partagés entre deux laboratoires, et lorsqu'il n'existe qu'une seule manipulation réelle et opérationnelle.
- dissémination des résultats : actuellement, il est habituel de fournir des liens hypertextes dans les publications pour établir des liens vers des contenus électroniques disponibles sur Internet. Nous pourrions imaginer augmenter l'impact démonstratif d'un tel lien si celui-ci avait pour cible, non pas un contenu électronique, mais un applicatif lançant une manipulation distante, permettant au lecteur de valider expérimentalement une idée. Il serait alors possible de montrer interactivement une expérimentation.
- partage de ressources : nous pourrions imaginer des centres mutualisant l'hébergement de ressources onéreuses, dans des locaux adaptés, financés grâce aux phénomènes d'économie d'échelle. Les dispositifs seraient accessibles à chaque laboratoire comme s'ils étaient hébergés par celui-ci, mais sans les contraintes de compétences et coûts d'entretien.

Au delà du simple échange d'informations, il s'agit de donner les moyens de produire de la recherche de manière partagée. En ce sens, cela renforce la philosophie intrinsèque des collaboratoires. Les liens entre laboratoires pourraient devenir plus profonds puisque le dialogue serait plus riche.

Nous nous sommes attachés dans ce paragraphe à décrire les collaboratoires, le lien réflexif tissé sur l'entité « laboratoire » de la figure I.1. Cependant, les laboratoires partagent ce besoin de « télé-piloter » des dispositifs avec les industries.

Les points de synergies les plus évidents sont les prestations de recherche. Cependant l'entreprise actuelle a également besoin de télé-piloter ses propres moyens de production pour assurer sa rentabilité et sa pérennité.

4. Bien que la connaissance s'acquiert (ou plutôt se développe), à défaut d'être réellement « transmise », nous emploierons quand même l'abus de langage « transferts de compétences » dans la suite de ce mémoire puisqu'il s'agit de l'expression la plus employée pour dénoter, entre autre, le développement des connaissances pour l'industrie à partir de la capitalisation des connaissances des laboratoires de recherche. Par ailleurs, nous n'évoquerons pas ici le problème de la certification (du transfert) des compétences. Cet aspect incombe, à notre sens, à un tiers afin de garantir la crédibilité de l'ensemble, un peu à l'image du contrôle technique automobile, opéré par un tiers, et non pas par un garagiste.

I.1.3.2 Les contrôles de procédés industriels

Pour l'entreprise étendue

Une réflexion identique peut être menée en considérant à présent les applications industrielles de nos travaux. Depuis quelques années, le secteur des nouvelles technologies a beaucoup évolué et son implantation dans l'entreprise est désormais incontournable. Le point de départ a certainement été le système d'information des entreprises : collecte et gestion de données diverses et variées (catalogue de vente en ligne, gestion de la paye, ...).

Depuis quelques années, les frontières de l'entreprise sont devenues floues. Il est commun de parler d'entreprise étendue (Mariotti, 2005). De plus, les nouvelles technologies constituent un levier stratégique à l'épanouissement des industries. Les nouvelles fonctionnalités techniques progressent chaque jour au point de concurrencer voire de se substituer aux formes classiques de l'organisation du travail. Le nomadisme et le télé-travail sont deux facettes de cette évolution.

Pour ces mêmes raisons, l'évolution des technologies nous autorise à ouvrir de nouveaux chantiers. Il nous semble qu'aujourd'hui un aspect de cette évolution réside dans le pilotage distant, au travers d'Internet, des moyens de production de l'entreprise. Ceci soulève d'innombrables questions auxquelles il ne nous appartient pas d'apporter une réponse ici mais dont il faut avoir conscience pour comprendre le contexte de nos travaux :

- Quels partenariats mettre en œuvre pour assurer la réussite de l'évolution du mode de fonctionnement de l'entreprise ?
- Quelles valeurs ajoutées ce type de projets peut-il apporter dans le management (réponse innovante au problème de la délocalisation, réactivité améliorée, ouverture de nouveaux partenariats, ...) ?
- Quel(s) modèle(s) économique(s) pour assurer la pérennité de l'entreprise au regard de l'évolution continue des technologies ?
- Quelle(s) influence(s) cela peut-il avoir sur le fonctionnement même de l'entreprise ou de la société en général ?

Outre renforcer la capillarité des processus dans les entreprises étendues, le contrôle de procédés distants, comme dans le cas de collaboratoires, permet de rentabiliser les acquisitions via des prestations pour le monde industriel. Ces prestations se destinent aux principaux usages suivant : recherche industrielle, maintenances préventives et curatives, ou formation continue.

En effet, il vient naturellement à l'esprit le besoin de maintenance dans des entreprises de production. Il peut être hautement intéressant pour une telle catégorie d'entreprises

de pouvoir intervenir à distance sur une chaîne de production, soit en cas de problème (télé-opérations de maintenances curatives), soit pour de l'optimisation du contrôle (en cas de télé-opérations de maintenances préventives). En effet, les maintenances physiques requièrent souvent des frais et un délai inhérent au déplacement des intervenants. On peut quelquefois éviter cela via le pilotage distant, si le problème est bénin et ne demande pas une intervention physique complexe *in situ*.

Pour le transfert de compétences

Un autre besoin poussant les industries à la télé-opération et à son utilisation réside dans les prestations de recherche. Celles-ci constituent des actions ponctuelles, permettant au monde industriel d'avoir accès aux plates-formes technologiques et au savoir-faire associé des laboratoires universitaires. Les industriels ne sont alors plus des producteurs mais des consommateurs de la télé-opération. Les formations continues permettent aux industriels, via les acteurs de formations, de proposer à leurs salariés une actualisation de leurs compétences sur des matériels de haute technologie. En effet, la complexité d'utilisation de ces appareils nécessite en général une longue phase d'apprentissage qui n'est pas toujours possible sur site en raison de la disponibilité des matériels. De plus, le déplacement d'une personne en formation sur site, engendre des coûts de fonctionnement parfois non négligeables pour l'entreprise. La charge financière est un facteur qui incite les industries à frapper à la porte des laboratoires, pour la formation ou la maintenance des dispositifs.

Il va sans dire que cette possibilité offerte au monde de l'industrie se cristallise autour de la notion de transfert de compétences. En effet, les relations qu'entretiennent les entreprises et laboratoires, parfois par l'entremise d'un institut supérieur de formation telle une école d'ingénieurs, visent à *transformer les résultats scientifiques en savoir-faire industriel*. Les prestations de recherche et les formations continues participent à cet échange. De la même façon que le pilotage distant de procédés participe au développement des prestations de recherche et des formations continues, on peut ainsi établir que le pilotage distant favorise le transfert de compétences des laboratoires vers le monde industriel.

I.1.3.3 La formation pratique à distance

De l'usage de l'ordinateur dans la formation

L'usage de l'ordinateur dans la formation initiale a débuté par un rôle de simple outil, avant d'être de plus en plus présent, et de devenir un environnement pédagogique. Sans vouloir entrer dans les détails d'une chronologie exhaustive, nous amenons ici des précisions sur son emploi progressif, pour comprendre le rôle que l'ordinateur joue aujour-

d'hui dans l'expérimentation à distance.

Nous pouvons esquisser trois périodes que nous qualifions de majeures concernant l'usage de l'ordinateur dans les formations (figure I.2). Évidemment, il est très difficile d'apporter des dates précises concernant les révolutions des utilisations de l'ordinateur car les usages sont « glissants » et les nouveautés se présentent rarement sous la forme d'une rupture brutale. Le lecteur souhaitant une granularité plus fine trouvera plus de détails dans (Bruillard, 1997) et (Peraya *et al.*, 2002).

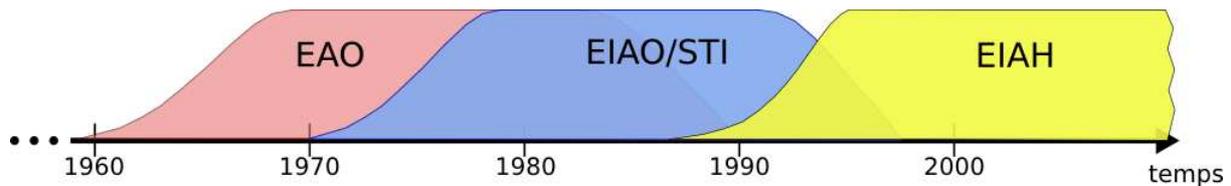


FIG. I.2 – Des EAO aux EIAH en passant par les EIAO puis STI : évolution de l'usage de l'ordinateur dans l'enseignement de 1960 jusqu'à nos jours.

Ainsi, nous mettons en avant trois principales périodes pour ces cinquantes dernières années :

- EAO⁵ : Enseignement Assisté par Ordinateur
- EIAO/STI⁶ : Enseignement Intelligemment Assisté par Ordinateur/Systèmes de Tutorat Intelligent
- EIAH : Environnement Informatique pour l'Apprentissage Humain

Les premières utilisations de l'ordinateur dans l'enseignement débutent vers le début des années 1960, pour ensuite prendre de plus en plus part à l'activité d'apprentissage (nous ne faisons pas ici débat de la différence entre l'usage de l'ordinateur pour l'enseignement et l'usage de l'ordinateur pour l'apprentissage). À cette époque, l'ordinateur est vu comme un support, souvent utilisé comme calculateur. Son principal rôle est de satisfaire des opérations élémentaires, à l'initiative des apprenants ou des maîtres. Bien entendu, les interactions qui en découlent ne procurent qu'une faible génération de savoirs : l'ordinateur est seulement un outil permettant d'accélérer le calcul.

Dans les salles de travaux pratiques, c'est l'émergence de l'Expérimentation Assistée par Ordinateur (ExAO). Là où l'EAO traite de l'enseignement assisté par l'ordinateur de manière générale, l'ExAO en est une sous-discipline consacrée plus précisément à l'utilisation de l'ordinateur pour les travaux pratiques. Les travaux remarquables de Nonnon et son « appareil robot » en 1972, participent grandement à l'élan de l'usage de l'ordinateur pour les enseignements pratiques. On retrouve Nonnon au milieu des années 80 avec

5. traduit de l'anglais « Computer-Aided Instruction » (CAI)

6. traduit de l'anglais « Intelligent Computer Aided Instruction » (ICAI) pour le terme EIAO et « Intelligent Tutoring Systems » (ITS) pour le terme STI

le principe de la lunette cognitive, une métaphore de la vision à travers des lunettes, où chaque verre amène une vision différente sur la manipulation observée : un verre de lunette permet de contrôler les paramètres d'une expérimentation, tandis que l'autre permet de visualiser l'impact des changements sur l'expérimentation elle-même.

Dès 1970, les EIAO et les STI débutent timidement leur apparition dans les enseignements traditionnels, mais ne s'imposent vraiment que dix ans plus tard, vers le début des années 1980⁷. Ces systèmes sont à cette époque l'objet de recherche en intelligence artificielle (IA). En effet, nous sommes alors dans une période où l'IA prétend pouvoir, à terme, modéliser le comportement humain. Pour réaliser cela, les principaux modèles reposent sur la notion de mimétisme, qui vise à simuler le comportement humain par la reproduction de ses faits et gestes. Si l'on applique ce principe à l'enseignement, les chercheurs en IA ont l'idée de construire des systèmes où l'ordinateur devient la copie du tuteur enseignant, le détenteur du savoir. La finalité est d'obtenir des systèmes où l'ordinateur remplit le rôle du tuteur, déchargeant celui-ci d'une partie de la tâche. L'ultime objectif est de constituer une bijection entre un tuteur et un apprenant pour personnaliser la formation (puisque l'on ne peut généralement pas supporter le coût d'un enseignant à temps plein par élève).

Néanmoins, les travaux en IA n'aboutissent pas complètement et leurs résultats apportent frustrations et désillusions à la fin des années 80, début des années 90. Parallèlement, on assiste à l'émergence des EIAH, portés par un nouveau courant en sciences cognitives appliquées à l'éducation. Nous aurons par ailleurs l'occasion de revenir sur cette théorie de l'apprentissage plus tard (§I.2.1.2). Ce qu'il faut comprendre ici, c'est le rôle que joue l'ordinateur aujourd'hui dans les systèmes modernes d'éducation. L'ordinateur n'est plus seulement détenteur d'un savoir qu'il prodigue au coup par coup, comme c'était le cas pour les EIAO/STI, mais il essaie au contraire de favoriser les situations où l'apprenant va lui-même construire son savoir, à partir d'observations, de questions ouvertes et de mises en situation. Pour cela, il essaie de faire naître et de cultiver la curiosité de l'apprenant, lequel devient beaucoup plus actif dans la construction de son savoir, puisqu'il est alors placé au centre de la pédagogie.

En conséquence, l'usage de l'ordinateur pendant ces cinquante dernières années démontre que les systèmes d'éducation assistée par ordinateur deviennent de plus en plus performants et centrés sur l'apprenant. On est passé du modèle centré ordinateur, au modèle centré utilisateur apprenant. Aujourd'hui, ces systèmes poussent les apprenants à se poser des questions. Plutôt que de subir un transfert de savoirs, le média informatique est là pour fournir et entretenir ces situations d'apprentissage.

7. Encore une fois, il n'est pas aisé de dresser des contours nets à de telles évolutions. L'objectif est simplement de situer ces évolutions les unes par rapport aux autres dans le temps.

Néanmoins, ces environnements favorisent l'enseignement par le contenu, et moins par la pratique. Dans la lignée de l'évolution actuelle, nous pensons que l'utilisation de la télé-opération dans les formations est d'une importance capitale pour les acteurs de formations. Un enseignement pratique est le prolongement d'un enseignement traditionnel, particulièrement lorsque l'objectif de la formation est de faire prendre conscience à l'apprenant de la réalité, par lui même. C'est ce que nous proposons d'argumenter dans le paragraphe suivant.

Des EIAH pour travaux pratiques : une importance capitale pour les acteurs de formations

L'ensemble des acteurs publics et privés de la formation supérieure en Sciences de l'Ingénieur et Technologie est aujourd'hui désireux de partager des ressources numériques existantes, des outils, des expériences, etc. L'objectif majeur est d'obtenir une large diffusion sur Internet de ressources numériques capables de renforcer la qualité et la visibilité des formations offertes par les acteurs de formations. Il apparaît clairement que l'ensemble des supports de formation des établissements a vocation à être proposé et partagé via les nouvelles technologies, tant au sein de la communauté universitaire (au sens large, y compris les grandes écoles) qu'avec les partenaires industriels (§I.1.3.2).

Ainsi, la grande majorité des acteurs de formations propose des modules à distance ou encore des diplômes à distance dans leur catalogue de formations. C'est également le cas à l'ISTASE depuis 1998 avec le campus numérique Enseignement à Distance en École d'Ingénieur 3^{ème} année (Eden3) (Fayolle, 2004). Ce campus numérique est animé par une approche pragmatique, dans le sens où il ne vise pas à adapter les contenus existants pour la formation distante, mais privilégie une relation sous la forme de tutorat entre un enseignant et un élève ingénieur. Ainsi, ce programme démontre, depuis près de dix ans, que l'association entre un tuteur enseignant et un portail pédagogique conduit au bon encadrement d'un élève ingénieur, lequel suit les modules proposés par la formation à distance. Un tel dispositif offre une large part aux supports de cours numériques et à l'interaction asynchrone pour des enseignements dirigés, que ce soit sous la forme d'exercices ou de projets.

Malheureusement, l'une des caractéristiques des formations en ingénierie et technologie est le besoin d'apprentissages pratiques sur des dispositifs d'instrumentation évolués et réels. Dans le contexte actuel de la formation à distance, la plupart des supports de formations élude cette problématique de formation pratique, ou la reporte à des sessions présentiels.

Sans prétendre remplacer ou nier l'utilité de ces sessions présentiels, il nous paraît important d'étudier de manière globale dans quelle mesure nous pouvons apporter des solutions ouvrant la voie d'une formation à distance pratique, c'est-à-dire utilisant des appareillages au travers d'Internet, (sans opposer la formation à distance et la formation en présentiel (Lukosch *et al.*, 1999)). Pour éclaircir ce point, citons quelques exemples de dispositifs qui pourraient utilement être pilotés à distance dans un contexte de formation :

- En mécanique : usinage grande vitesse, prototypage rapide, machines à commandes numériques, bancs de mesures et de contrôles dimensionnels (Chiculita & Frangu, 2002)
- En télécommunications : analyseurs hyperfréquences, antennes (Bayard *et al.*, 2002)
- En optique : bancs de caractérisation optique (Esche, 2002)
- En logistique : contrôle de chaîne de production
- En robotique : robot télé-opérable (Bicchi *et al.*, 2001)
- En micro-électronique : circuits intégrés, programmation FPGA, systèmes embarqués (Alamo *et al.*, 2002)
- En chimie : spectrographe (Cedazo *et al.*, 2006)
- Et certainement d'autres champs disciplinaires. . .

Ce sont autant de domaines technologiques pour lesquels la FOAD ne favorise pas l'enseignement pratique et qui pourraient largement trouver avantage dans l'émergence d'infrastructures et plates-formes génériques pour la mise en place assistée de travaux pratiques distants.

I.1.4 Bilan

Cette section a présenté les tenants et aboutissants du pilotage distant de ressources. En effet, nous avons vu que la télé-manipulation n'est pas seulement une problématique financière, mais qu'elle apporte une véritable réponse technologique pour gérer des situations inextricables autrement. Les parties prenantes composent un trio : laboratoires, industries et instituts supérieurs de formation. Il a été exposé les buts propres à chacun, ainsi que les relations tissées entre chacun des acteurs.

Par la suite, nous allons plus spécifiquement nous concentrer sur les possibilités de formations offertes par la télé-opération de ressources de haute technologie, envers des laboratoires, des industriels ou encore des étudiants. À la lumière de ces explications, nous pourrons ensuite, à travers un état de l'art, exposer le contexte scientifique ainsi que les objectifs de nos travaux.

I.2 La formation pratique à distance

I.2.1 Théories de l'apprentissage et travaux pratiques à distance

I.2.1.1 Précisions linguistiques

Arrêtons nous sur le terme « laboratoire distant ». Il provient en fait de la traduction du terme anglais « remote laboratory », qui couvre les sessions de travaux pratiques effectuées à distance. Cet abus de langage appartient à la communauté scientifique française concentrée sur cette démarche visant à apporter la paille au plus près des étudiants qui ne peuvent pas se déplacer (cela vaut aussi pour les enseignants!). La littérature sur le sujet, majoritairement anglaise, n'apporte pas beaucoup de facilité à la traduction de ce terme en français (d'où une première traduction « laboratoires distants »). Néanmoins, s'il convient d'offrir un nom concis et plus approprié en langue française pour les travaux pratiques dans le cadre de formation à distance, nous nous rallions à la proposition de Benmohamed qui suggère le terme de « télé-TPs »⁸. Comme l'auteur l'explique :

« [...] nous avons préféré le terme télé[-]TP à celui de « e-TP ». Les termes : « télélaboratoires » [...], « expérimentations à distance » [...] ou encore « laboratoires en ligne » [...] indiquent clairement l'implication de la notion de la distance. Le terme « e-TP » n'implique pas forcément la distance de la même manière que la e-formation, où le « e » désigne la formation par la voie « électronique », c'est-à-dire médiatisée par l'informatique. » (Benmohamed, 2006)

Par ailleurs, nous conservons la préoccupation de porter nos résultats comme des éléments de réponses aux problématiques industrielles (§I.1.3.2: contrôles de procédés distants) et aux laboratoires (§I.1.3.1: collaboratoires, prestations de recherche). Nous parlerons alors d'eInstrumentation, puisqu'il s'agit historiquement du nom associé à ce travail depuis son origine. En fait, la rigueur voudrait que l'on donne les deux dimensions (la distance et l'assistance par ordinateur) au terme Instrumentation pour obtenir « télé-e-Instrumentation », de la même façon que nous serions obligé d'employer « télé-e-TPs » lorsque l'instrumentation est appliquée au domaine de la formation. Fort heureusement nous ne manquerons pas d'utiliser les abus de langages suivants : « télé-TPs » pour parler de travaux pratiques à distance et de « eInstrumentation » pour faire référence au travail défendu par ce mémoire, puisqu'il couvre plus largement le spectre concernant la manipulation distante de ressources que ce soit pour les industries, les laboratoires ou les instituts supérieurs de formation comme les écoles d'ingénieurs.

8. TP pour Travaux Pratiques

I.2.1.2 Apprendre à distance par la pratique

Les cours magistraux, les travaux dirigés ou les projets, se prêtent à l'enseignement à distance. Il en va tout autrement des enseignements pratiques, néanmoins nécessaires aux élèves pour acquérir des savoir-faire. En effet, dans les étapes d'apprentissage, des travaux reconnus en sciences cognitives ont déjà démontré que les apprenants débutent en comparant les nouvelles situations auxquelles ils font face à des exemples connus. Ensuite, et seulement ensuite, ils sont capables de bâtir des règles par rapport à l'expérience d'apprentissage qu'ils ont vécu. Anderson décrit ainsi cette transformation :

« this is principal means by which knowledge transitions from a declarative form (encoding of examples) to a procedural form (production rules) » (Anderson *et al.*, 1997)

Bien sûr, nous sommes tentés de dresser la comparaison avec nos propres expériences : voir, entendre, toucher (manipuler), mais encore, sentir, entrent dans les « règles » qui nous permettent de reconnaître des expériences passées.

Au final, qu'apporte un enseignement pratique à l'apprenant ?

Une étude européenne (Sere *et al.*, 1998) explique qu'il s'agit d'un enjeu essentiel pour les élèves puisqu'il vise à :

- Apprendre à faire le lien entre la théorie et les phénomènes réels,
- Acquérir des compétences expérimentales et une démarche scientifique,
- Entretenir la motivation et la curiosité,
- Participer au développement personnel,
- Contribuer à la construction des comportements sociaux.

Pour l'enseignant il s'agit, d'une part, de se procurer des situations pédagogiques différentes du cadre traditionnel de la classe, et d'autre part, se constituer un outil alternatif d'évaluation des compétences et des connaissances des élèves.

Dans un tel contexte, nul doute que l'ordinateur a son rôle à jouer, et qu'il enrichit les expériences d'apprentissage (Latchman *et al.*, 1999) . Nous avons par ailleurs déjà illustré les rôles successifs joués par l'ordinateur ces dernières années (figure I.2). Néanmoins, alors que celui-ci s'est imposé dans les séances de travaux pratiques, les universités arrivent péniblement à inclure ces travaux pratiques dans leur catalogue de formation à distance, et ce depuis le tout début de l'émergence des laboratoires distants (Alhalabi

et al., 2000).

Dans la section précédente, nous avons exposé l'avènement des EIAH (figure I.2). Nous avons alors évoqué une nouvelle théorie d'apprentissage supportant les EIAH et les désillusions des recherches en intelligence artificielle ayant conduit à une remise en question des STI. Ce courant se nomme « constructivisme » : il favorise les interactions entre élèves, tuteurs et systèmes dans le but de susciter l'envie d'apprendre. Ce courant, issu des sciences cognitives, doit son avènement au retour des méthodes introspectives d'apprentissage. Il s'agit de conduire l'élève à se poser des questions, et non plus à subir l'enseignement. Cette pratique apporte *peut-être* moins de réponses théoriques à l'élève, mais lui permet de développer son esprit critique et sa façon d'appréhender les notions nouvelles. Le constructivisme fait suite au béhaviorisme. Le béhaviorisme est axé sur l'observation des comportements et voit dans l'apprentissage un changement de comportement qui dure (théorie de Pavlov début du XX^{ème} siècle) repris par Skinner (Tavris et al., 1999). Le béhaviorisme s'intéresse seulement aux interactions entre l'environnement et le comportement des apprenants. Les apprenants sont considérés comme des « boîtes noires », que l'on ne peut ouvrir. En reprenant la chronologie présentée précédemment, le passage du béhaviorisme au constructivisme correspond au déclin des STI en faveur des EIAH, comme l'illustre la figure I.3.

Béhaviorisme et constructivisme sont opposés dans la majorité des écrits scientifiques, et ce depuis longtemps (Jonassen & Wang, 1993). Plusieurs éléments séparent ces deux approches. Le plus important réside dans la façon dont les apprenants acquièrent leurs connaissances. Le béhaviorisme vise à transmettre passivement, à l'initiative de l'enseignant, des connaissances aux élèves, qui devront reproduire ensuite les comportements exposés par le tuteur. A l'inverse, le constructivisme vise à pousser les élèves à découvrir les connaissances par eux-même, via le dialogue et le questionnement.

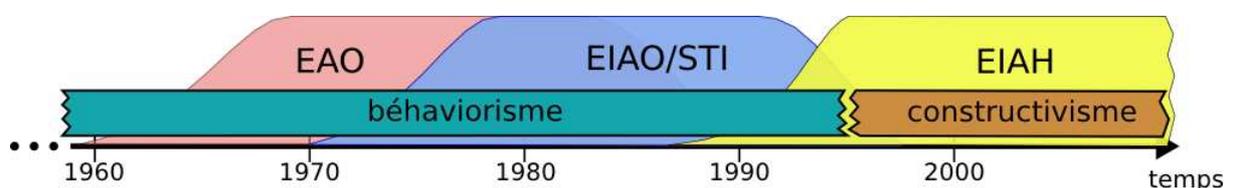


FIG. I.3 – Enseignement assisté par ordinateur et théories d'apprentissage.

Nous sommes actuellement dans une période où le courant dominant est le constructivisme, et c'est précisément pour cette raison que les formations ressentent ce besoin crucial de travaux pratiques à distance. En effet, comment imaginer une pédagogie, pour les disciplines technologiques, laissant une large place à la découverte par l'apprenant,

sans travaux pratiques? Il est certain que les formations professionnalisantes et les formations continues demandent aux enseignants de transmettre des savoir-faire, en plus des connaissances théoriques, et que ce transfert de compétences ne peut s'effectuer que via le support de travaux pratiques.

I.2.2 Les télé-TPs

Les télé-TPs sont issus de divers domaines de l'enseignement technologique, mais n'ont pas tous la même finalité. Le constat le plus simple consiste à comparer les télé-TPs impliquant des instruments réels et les télé-TPs impliquant une simulation : ce ne sont pas les mêmes tenants et aboutissants pédagogiques, pas plus que les mêmes solutions technologiques. Par ailleurs, il existe deux catégories de TP assistés par ordinateur, comme l'illustre la figure I.4. Nous allons détailler chacune des possibilités et établir les situations auxquelles chacune est la plus appropriée.

I.2.2.1 Taxonomie des travaux pratiques assistés par ordinateur.

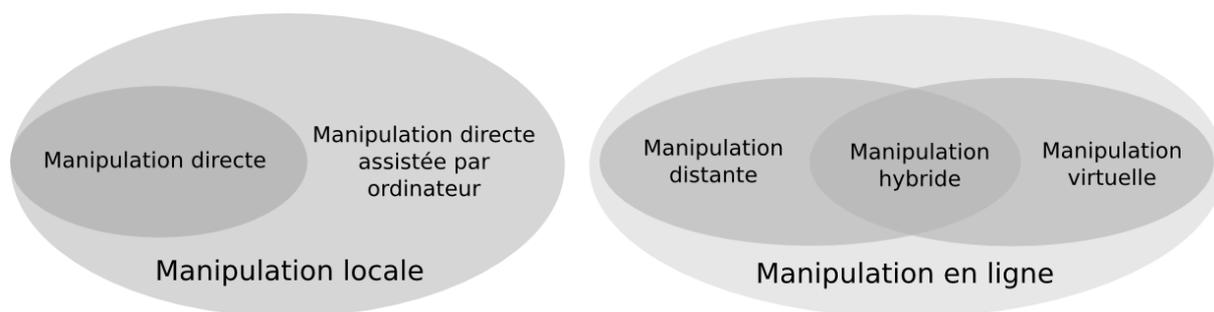


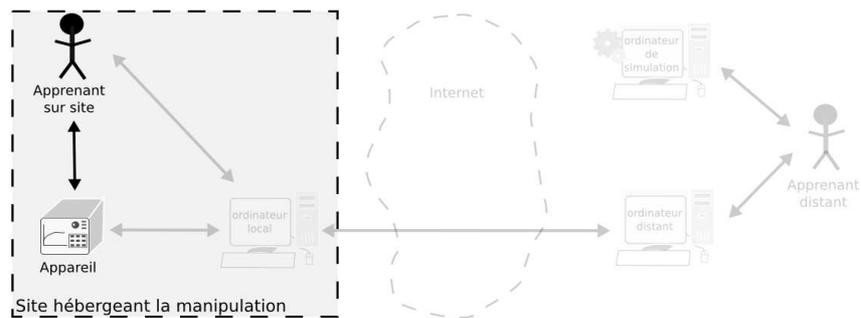
FIG. I.4 – *Taxonomie des travaux pratiques assistés par ordinateur.*

Il est souvent difficile de trouver dans la littérature une certaine homogénéité des termes employés. Par exemple, très récemment encore, le terme « instruments virtuels » était employé lorsque la manipulation était simplement en ligne, sans être une simulation (Canfora *et al.*, 2004). Nous nous proposons donc de faire un rapide tour d'horizon des catégories de travaux pratiques que l'on peut identifier. Il est à noter que nos travaux rejoignent en partie les récentes propositions de Ma (Ma & Nickerson, 2006).

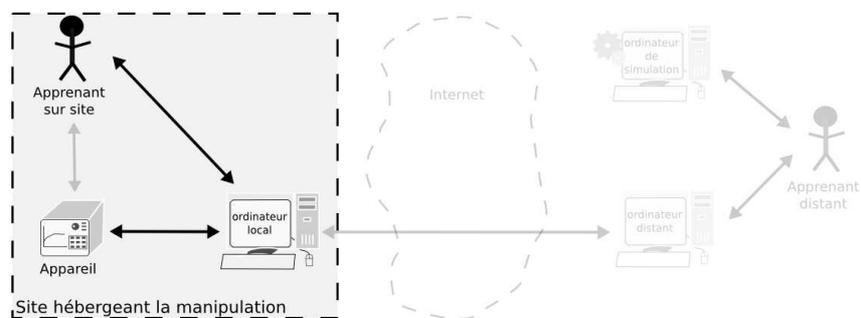
Laboratoires locaux assistés ou non par l'ordinateur

La façon la plus naturelle de conduire une expérimentation à but d'enseignement est bien sûr de la réaliser *in situ*. Lorsque l'ordinateur était encore absent des salles de travaux pratiques, la manipulation était directe (figure I.5).

On appelle « laboratoire local » cette situation où aucun média ne joue l'interface entre l'homme et l'expérimentation. Bien entendu, cette façon de procéder existe toujours, car

FIG. I.5 – *Les laboratoires locaux.*

toutes les expérimentations ne nécessitent pas forcément l'emploi d'un ordinateur, mais elle tend à disparaître. En effet, l'ordinateur apporte bien souvent une nouvelle dimension à la manipulation considérée (visualisation, capacité d'utilisation, adaptation, ...). On parle alors de « laboratoire local assisté par ordinateur » (figure I.6).

FIG. I.6 – *Les laboratoires locaux assistés par ordinateur.*

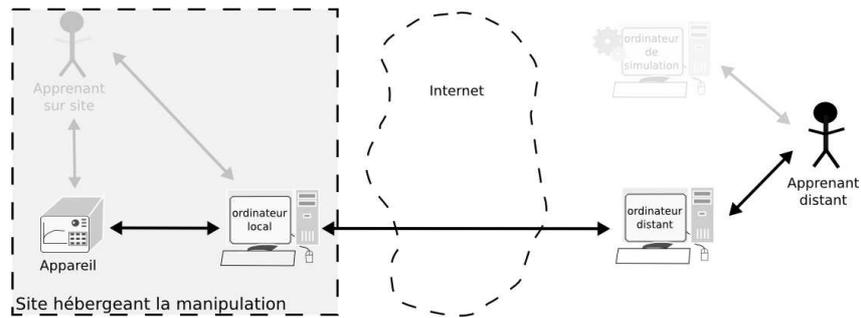
La manipulation est directe dans le sens où il est encore possible d'agir physiquement avec les dispositifs, car l'ordinateur ne sait pas ou ne peut pas gérer entièrement la manipulation. Une panne peut également surgir mais la présence au même endroit et en même temps des dispositifs et des apprenants permet de surmonter ce genre de problème.

Ces deux premiers types de laboratoires, très proches, composent la famille des manipulations locales.

Laboratoires distants

Les laboratoires distants concernent les travaux pratiques menés à distance sur des dispositifs réels (figure I.7).

L'enjeu est beaucoup plus complexe que précédemment, car les apprenants, les tuteurs ainsi que les dispositifs ne sont pas tous au même endroit, ce qui diminue grandement la communication entre chacun des acteurs. En effet, un élève dans sa chambre d'étudiant, pratiquant un télé-TP, aura plus tendance à se sentir esseulé par rapport à la même pratique dans un laboratoire local. L'idée que nous voulons défendre, est que les laboratoires

FIG. I.7 – *Les laboratoires distants.*

distants ne sont pas les remplaçants systématiques des laboratoires locaux. En effet, même si les étudiants plébiscitent les laboratoires distants face aux laboratoires locaux (Corter *et al.*, 2004), il existe des situations où les laboratoires locaux ne sont pas envisageables (dangerosité, accessibilité, temps, finances, ...). Les laboratoires distants constituent alors une des alternatives possibles pour palier à l'impossibilité d'un TP *in situ*.

Aktan résume ainsi le fait que les laboratoires locaux sont les mieux placés pour conduire des expérimentations à finalité d'apprentissage :

« We recognize that no remote system will provide exactly the same feel that is obtained by students sitting in the laboratory. However, SBBT⁹ attempts to project the in-laboratory experience. It allows the student to work on real equipment (no virtual reality here!) and learn about real issues in controller design (no idealized simulations!). The control engineering experience gained by the student is pedagogically the same as the traditional laboratory experience. » (Aktan *et al.*, 1996).

soulignant ainsi que rien ne peut remplacer l'expérience d'apprentissage réelle et *in situ* d'un laboratoire local pour l'enseignement pratique.

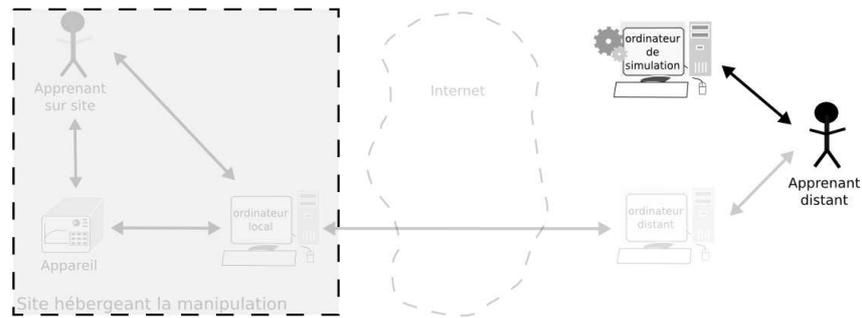
Du point de vue de l'architecture technologique, dans la quasi majorité des approches, l'ordinateur est relié localement au dispositif mais également sur Internet. C'est alors lui qui fait office de relai entre les commandes de l'utilisateur distant et le système local.

Laboratoires virtuels

Une autre alternative au laboratoire local, lorsque celui-ci n'est pas envisageable, est de virtualiser le dispositif. Il s'agit d'une approche déjà ancienne, dont le rôle a déjà été discuté (Smith & Pollard, 1986).

En effet, plutôt que de construire toute une architecture logicielle sur Internet, connec-

9. SBBT signifie « Second Best to Being There », que l'on pourrait traduire par « Deuxième meilleur choix après être sur place »

FIG. I.8 – *Les laboratoires virtuels.*

tant le LAN¹⁰ de l'étudiant au LAN du laboratoire, on peut imaginer établir un algorithme simulant le comportement du dispositif. De ce fait, la connexion à Internet n'est plus forcément nécessaire puisque l'on peut imaginer embarquer un client lourd sur l'ordinateur de l'élève. Bien sûr, dans le cas de clients légers et semi-légers, où l'algorithme est exécuté sur un serveur central, l'apprenant simule de la même façon son expérience.

Ainsi, la connexion avec l'appareil est complètement rompue, puisque le logiciel, guidé par son algorithme, remplace le dispositif réel. Évidemment, construire de tels algorithmes demande de s'appuyer sur des modèles représentant la manipulation réelle (difficile de modéliser très exactement la réalité dans toute sa complexité et ses aléas). Du coup, les implémentations proposées incluent forcément le biais induit par le concepteur du modèle correspondant à l'instrument. De plus, s'atteler à modéliser l'entière réalité d'un dispositif est quasi impossible, du fait de sa complexité. Par conséquent, l'arbre des fonctionnalités est « élagué » pour diminuer cette complexité et donc la granularité de la modélisation de la réalité. En ce sens, et pour continuer à créer des passerelles entre théories d'apprentissage et télé-TPs, les laboratoires virtuels constituent probablement un exemple de la métaphore de l'étagage de Brunner car il diminue le degré de liberté de l'apprenant. En effet, la manipulation virtuelle, puisque basée sur un modèle ne présentant pas la même granularité que la réalité, élude les fonctionnalités moins importantes de son modèle pour satisfaire l'objectif pédagogique. Néanmoins, ce sont tous ces détails qui, combinés, forment le caractère pseudo-aléatoire d'une expérimentation. La question est donc de savoir quel degré de réalité on souhaite atteindre pour une simulation, tout en sachant que l'on ne pourra jamais atteindre un degré de modélisation correspondant à la réalité.

Laboratoires hybrides

Enfin, il existe une autre classe de laboratoires visant à rallier le meilleur des deux mondes (distants et virtuels) : il s'agit des laboratoires hybrides. Ces laboratoires sont en partie composés de manipulations sur des dispositifs distants et réels, mais ils comportent

10. Local Area Network

des simulations, issues d'une étape de modélisation, comme pour les laboratoires virtuels.

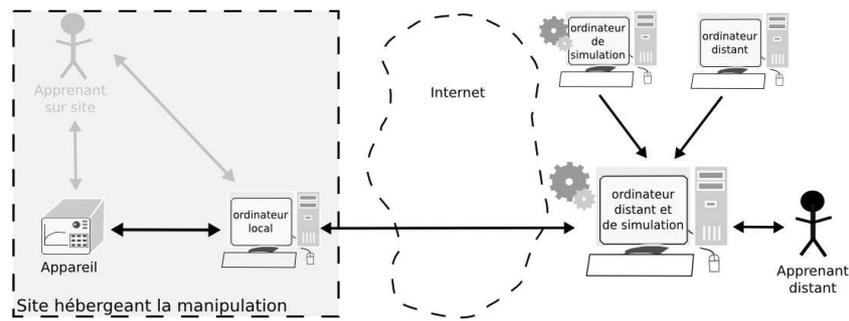


FIG. I.9 – *Les laboratoires hybrides.*

Cela correspond à une catégorie de TP où l'on manipule un ensemble réel de dispositifs que l'on fait ensuite interopérer par simulations, dans l'optique d'une intégration à large échelle difficilement reproductible en laboratoire. Imaginons des travaux pratiques dans le domaine des réseaux informatiques. Un exemple de configuration hybride serait de permettre aux étudiants de configurer des routeurs à distance, et ensuite de les interopérer avec d'autres routeurs cette fois-ci virtuels pour créer un plus grand réseau (pour éviter de fournir plusieurs dizaines de routeurs par apprenants si l'objectif du TP est de simuler un MAN¹¹). Ainsi, ce laboratoire sera qualifié d'hybride : un des nœuds du réseau serait un réel routeur configuré à distance par l'apprenant, et l'intégration d'un MAN, pour passer à l'échelle, serait simulée.

Les manipulations distantes, virtuelles ou hybrides se regroupent dans la famille des manipulations distantes que nous appelons les manipulations en ligne¹².

I.2.2.2 Télé-TPs réels et télé-TPs virtuels : que choisir ?

L'un de nos objectifs consiste à porter les manipulations en laboratoire à la porte des étudiants via Internet. On est tenté de dire que deux possibilités s'offrent à nous : des manipulations distantes et des manipulations virtuelles. Au delà des différences technologiques, on assiste bien souvent à un clivage pédagogique entre manipulation distante et manipulation réelle. Au final, notre mémoire se situe dans le domaine des télé-TPs. Néanmoins, notre propos n'est pas ici d'expliquer en quoi les télé-TPs sont supérieurs aux TP virtuels, mais plutôt de montrer les forces, faiblesses et complémentarités de chacun. Le but sous-jacent est de montrer les situations d'utilisation qui conviennent le mieux à chacune des solutions. En effet, il s'agit d'une question que nous nous sommes posée au début de ce travail : les instruments étant onéreux, vaut-il mieux établir une manipulation

11. Metropolitan Area Network

12. Encore une fois nous nous référons à un anglicisme répandu dans notre communauté scientifique : « Online Laboratories » dans le cas présent

virtuelle à travers une déclinaison de modèles d'utilisation, ou bien mettre en place un réel contrôle distant de ces instruments alors mutualisés ?

Pour répondre à cette question, nous avons résumé dans le tableau I.1 les forces et faiblesses de chaque approche :

	Avantages	Inconvénients
Laboratoires Virtuels	<ul style="list-style-type: none"> - Passage à l'échelle - Manipulation à taille humaine (infiniment petit ou grand) - Palier la dangerosité - Maîtrise du temps (en cas d'inertie du dispositif) - Minimise le risque technologique - Manipulation trop onéreuse - Préséance à un TP 	<ul style="list-style-type: none"> - Manipulation « aseptisée » - Complexité des modèles - Complexité des algorithmes - Se substitue parfois à l'apprenant - Pose le problème de la validation et de la qualité des modèles - Isolement des apprenants
Laboratoires Distants	<ul style="list-style-type: none"> - Conscience de la réalité de la manipulation - Développer une base d'instruments connus - Acquérir des connaissances pratiques - Illustrer les limites de la théorie - Plus facile à mémoriser - Favorise les situations de communication - Mutualisation 	<ul style="list-style-type: none"> - Compétences nécessaires - Robustesse - Coût - Gestion du temps d'utilisation

TAB. I.1 – Forces et faiblesses des manipulations distantes et virtuelles.

Les manipulations virtuelles sont, par nature, basées sur des modèles dont la complexité est fonction de la granularité que l'on veut atteindre dans la simulation de la réalité. En ce sens, elles conduisent forcément à des situations que nous qualifierons d'« aseptisées », c'est-à-dire trop proches des modèles théoriques. Il est difficile de reproduire tous les aléas qui composent une manipulation réelle. À l'inverse, les manipulations distantes procurent des situations réelles, et tendent à faciliter l'apprentissage, puisqu'elles se rapprochent plus de l'approche constructiviste (Aktan *et al.*, 1996).

Alors que les laboratoires virtuels préparent la manipulation pour les apprenants (par exemple, en offrant une palette de signaux électroniques à disposition pour une manipulation), les laboratoires distants requièrent de la part des élèves de savoir contrôler, certes à distance, les dispositifs générateurs de signaux, pour créer le signal souhaité. Dans cet exemple, savoir générer un signal spécifique en configurant un instrument, plutôt que simplement choisir un type de signal dans un panel prédéfini, exige de l'apprenant un degré de compétence plus important.

Un autre élément en faveur des laboratoires distants découle des perturbations et aléas des manipulations réelles. En effet, dans le cadre d'une manipulation réelle à distance, il est fréquent que le biais induit par la pratique (les connexions physiques, la température ambiante, le taux d'humidité, l'usure des dispositifs, etc.) fassent s'éloigner les résultats expérimentaux des résultats théoriques. Du coup, cela amène les élèves à se poser des questions sur les conditions de l'expérimentation, à confronter leurs résultats et à échanger entre eux sur le problème rencontré. Le fait de faire réfléchir les élèves par rapport à un problème rencontré, et de les faire dialoguer pour comprendre et acquérir du savoir, s'inscrit complètement dans la démarche constructiviste. De plus, en tout état de cause, ce sont des situations qui ont les plus faibles chances d'apparaître dans des manipulations virtuelles, puisque la simulation est justement basée sur les modèles théoriques que les étudiants ne font que vérifier (leurs résultats sont alors beaucoup plus booléens : on vérifie ou non le modèle théorique).

Néanmoins, il ne faut pas croire que le monde des travaux pratiques assistés par ordinateur soit si manichéen. En effet, il existe bien entendu des situations où les laboratoires virtuels sont un véritable outil pédagogique, au delà du simple substitut (par exemple dans le cas d'un simulateur de vol). Une simulation peut très bien trouver sa place dans une pré-séance aux travaux pratiques, puisqu'elle permet aux étudiants d'obtenir une idée de la manipulation qu'ils devront effectuer, mais en mettant de côté dans un premier temps les biais imputables aux manipulations réelles (Hodge *et al.*, 2001). Nous pensons également aux manipulations à grande échelle. Par exemple, étudier un réseau informatique métropolitain est possible en simulation, mais pas dans la réalité des salles de travaux pratiques informatiques. Par extension, certaines manipulations ne sont tout simplement pas à l'échelle humaine, que ce soit l'infiniment grand (comme en astronomie, climatologie, ...) ou l'infiniment petit (physique quantique par exemple). Les laboratoires virtuels apportent alors une véritable réponse, là où les laboratoires locaux ou distants ne peuvent le faire. Bien entendu, il existe d'autres situations où les laboratoires virtuels sont à leur avantage : dans le cas de manipulations dangereuses (en chimie par exemple) ou encore lorsque la manipulation réelle est trop onéreuse sans partage d'utilisation pour être considérée comme possible.

Parallèlement, même si certains perçoivent les laboratoires distants comme la « solution ultime » (Alhalabi *et al.*, 2001), il convient d'ajouter que les manipulations réelles à distance ne sont pas exemptes de lacunes. En effet, intégrer, maintenir et conduire des manipulations distantes demande un haut niveau de compétences, dans plusieurs domaines, au tuteur enseignant. De plus, il est certain que les manipulations distantes ont une fréquence d'occurrence de problèmes techniques plus importante durant les séances d'enseignement, freinant ou aidant du coup les situations d'apprentissage : les aléas techniques mettant hors service ces manipulations sont beaucoup plus importants que dans le cas de manipulations virtuelles. De plus, ces manipulations sont onéreuses, et ne sont pas toujours facilement reproductibles (Albu *et al.*, 2004).

Chacune des solutions possède ses forces et faiblesses, mais nous sommes tentés de penser que les manipulations distantes sont à privilégier, lorsque cela est possible. D'ailleurs, comment éviter de citer Nonnon :

« nous réaffirmerons notre conviction que pour développer l'esprit scientifique, il faut placer nos étudiants dans un environnement de laboratoire réel. Un environnement qui leur permette de pratiquer l'investigation scientifique de manière autonome, à partir de leurs propres questions ou hypothèses. » (Nonnon, 2005)

Par conséquent, nous pensons qu'il en va de même pour l'enseignement pratique à distance : il faut favoriser la manipulation distante par rapport à la manipulation virtuelle, lorsque cela est possible. A notre sens, lorsque l'on souhaite introduire de l'enseignement pratique dans les formations à distance, il convient de se poser les questions suivantes :

- La manipulation est-elle réalisable à l'échelle d'un laboratoire ?
- L'expérience pilotée à distance, mais ayant lieu *in situ*, est-elle sans risque ?
- La conduite de l'expérience à distance n'induit-elle pas un surcoût important en dehors de l'accès Internet ?
- Le temps de réaction du dispositif est-il compatible avec l'apprentissage ?

Si l'on répond quatre fois oui, il est probable qu'une manipulation distante insistera plus sur les objectifs pédagogiques d'apprentissage qu'une manipulation virtuelle.

I.2.3 Apprentissage Collaboratif Assisté par Ordinateur

Nous essayons donc de répondre à la question de la conduite des travaux pratiques à distance, par un pilotage distant mais réel de dispositifs.

Il faut noter qu'établir une relation entre un unique étudiant et la manipulation distante prive de la dimension sociale inter-apprenants. En effet, si on compare cela aux séances de travaux pratiques *in situ*, on se rend compte qu'un élément déterminant manque à notre situation d'apprentissage. Dans un laboratoire local (assisté par ordinateur ou non), les apprenants sont en contact les uns avec les autres. Si on veut pousser plus loin la métaphore des travaux pratiques dans le cadre de la formation à distance, il convient d'étudier les possibilités offertes par l'Apprentissage Collaboratif Assisté par Ordinateur (ACAO)¹³. Nous verrons également ce que l'ACAO apporte aux situations d'apprentissage, et donc aux apprenants. Nous étudierons ensuite les possibilités d'ACAO pour les situations de formation par la pratique.

I.2.3.1 Définitions

L'étude des interactions entre élèves dans les situations expérimentales n'est pas nouvelle. En 1982, Mangin écrivait déjà :

« This is interesting to notice that learners think that taking group decision to solve problems is part of the exercise. » (Mangin, 1982).

Par certains côtés, cela porte à croire que les étudiants voient l'accomplissement de leur expérimentation, non seulement dans la bonne conduite de celle-ci, mais également dans l'entraide et la résolution commune du même problème posé à chacun. La dimension collaborative est donc essentielle pour la professionnalisation des apprenants et de leurs comportements.

Bien sûr, l'ACAO partage beaucoup de notions avec la communauté du Travail Collaboratif Assisté par Ordinateur (TCAO)¹⁴. Certes, le domaine du TCAO est un peu éloigné du domaine de l'éducation, car il s'applique plus particulièrement à étudier le partage de documents et de connaissances au sein d'une organisation (bien souvent une entreprise). D'ailleurs, on parle quelque fois de « peer-help » (Greer *et al.*, 1998) dans de tels systèmes, ce qui a quand même tendance à rapprocher les deux mondes. De plus, ces deux domaines partagent la notion clé de collaboration, et les technologies de génie logiciel associées.

Mais posons la question : « Qu'apporte concrètement l'ACAO aux situations d'apprentissage ? »

13. Le terme anglais que l'on retrouve dans la littérature est « Computer Supported Collaborative Work » (CSCL)

14. Nous pensons une nouvelle fois qu'il est important de reporter ici le terme anglais correspondant : Computer Supported Collaborative Work (CSCW)

Comme le soulignait Stahl encore très récemment, l'ACAO est né dans les années 1990, en réaction à l'aberration constatée des logiciels éducatifs de l'époque, lesquels forçaient l'élève à apprendre en tant qu'individu isolé (Stahl *et al.*, 2006). Il est entendu ici que l'Internet fait sa révolution pour l'éducation en permettant aux apprenants d'entrer en relation et de discuter de leurs expériences. Bien sûr, cela implique le risque de dérives de la communication entre apprenants, laquelle peut alors véhiculer d'autres informations (majoritairement sociales) que celles propres à l'apprentissage.

Une première définition de l'ACAO réside dans l'aptitude des apprenants à communiquer entre eux autour d'un problème, alors que jusque là ils ne communiquaient qu'avec un tuteur, et le problème posé jouait le rôle de support. Dans ce cas le formateur ne communique pas avec les apprenants de façon unilatérale. C'est la proposition faite par Akkouche, dont nous reprenons une illustration (figure I.10).

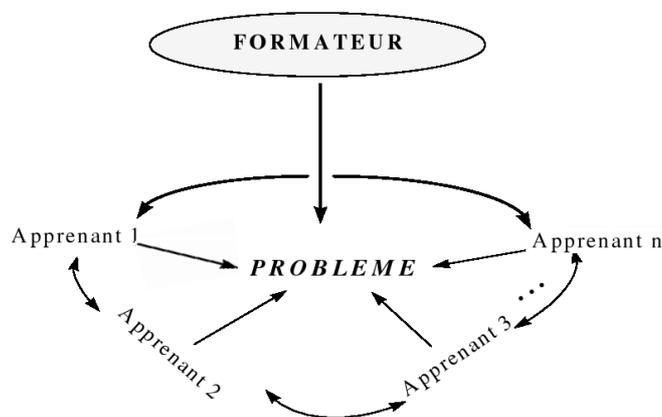


FIG. I.10 – Les apprenants communiquent entre eux autour du problème posé par le formateur, d'après (Akkouche, 1996).

Nous avons cependant besoin de pousser un peu plus la réflexion pour comprendre la nature des outils informatiques associés aux ACAO. En effet, il a été mis en exergue que l'ACAO produit ses propres artefacts pour médiatiser les collaborations au sein de groupes d'apprenants. La définition la plus reconnue de l'ACAO, et qui va dans ce sens, est celle de Koschmann, qui insiste sur le fait qu'il s'agit d'une activité partagée :

« CSCL is a field of study centrally concerned with meaning and the practices of meaning-making in the context of joint activity, and the ways in which these practices are mediated through designed artifacts. » (Koschmann, 2002).

Il est cependant dommage que la définition de l'ACAO ne donne pas une place plus importante à l'éducation, étant donné qu'il s'agit du cœur du problème posé. En effet, l'ACAO est inextricablement lié aux nouvelles techniques d'enseignement (ou devrions-nous dire aux nouvelles techniques d'apprentissage?), comme le souligne Stahl :

« In their penultimate sentence, Hakkarainen, Lipponen, and Järvelä correctly

point out that CSCL researchers have a complex challenge because the educational use of new information/communication technologies is inextricably bound up with new pedagogical and cognitive practices of learning and instruction. » (Stahl, 2002)

Au regard de cette citation et de nos convictions, nous souhaitons apporter notre propre vision sur la définition de l'ACAO :

« On parle d'apprentissage collaboratif assisté par ordinateur lorsque la collaboration a supplanté la coopération au sein d'un groupe dont les individus sont éloignés par la distance ou le temps, et communiquent à l'aide d'outils informatiques synchrones ou asynchrones. Le groupe construit alors ses propres connaissances, par l'observation et le dialogue, ainsi que par la volonté d'entraide des membres qui le composent, faisant ainsi profiter chaque individu des connaissances acquises par la communauté. »

Il nous incombe d'apporter quelques précisions sur cette définition. Nous voulons particulièrement insister sur le fait que l'ACAO est composé de collaborations, au delà de simples coopérations. Dillenbourg explique très simplement la différence entre collaboration et coopération :

« In cooperation, partners split the work, solve sub-tasks individually and then assemble the partial results into the final output. In collaboration, partners do the work 'together' » (Dillenbourg, 1999).

Ainsi, la collaboration vise à l'entraide pour résoudre un problème commun (c'est en ce sens que l'on rejoint totalement la communauté du TCAO), même si la collaboration peut également inclure des épisodes d'émulation dans les sessions de formations. À l'inverse, la coopération est une simple répartition des tâches face à un problème trop complexe pour être mené individuellement. Par exemple, construire une maison entre plus dans un processus coopératif tandis que gagner un match de football relève de la collaboration.

Comme nous l'avons exprimé dans notre définition, le temps et l'espace sont les deux principaux paramètres qui différencient une situation d'ACAO d'une autre. On se retrouve alors dans des situations synchrones ou asynchrones, et centralisées ou distribuées. Le couplage de ces paramètres conduit au tableau suivant :

	Même temps	Temps différent
Même endroit	synchrone centralisée	asynchrone centralisée
Endroit différent	synchrone distribuée	asynchrone distribuée

TAB. I.2 – *Les différentes configurations d'ACAO par le couplage espace et temps*

Chaque configuration nécessite malheureusement des outils différents. Par exemple, concernant la distribution en temps des apprenants, les outils synchrones diffèrent des outils asynchrones. L'exemple le plus utilisé d'outils asynchrones est le forum de discussion. A l'opposé, l'utilisation de la messagerie instantanée reste l'illustration la plus répandue d'outil synchrone. Généralement, les outils synchrones offrent une communication plus riche et soutenue. Pour l'exemple de la messagerie instantanée, cet outil est en effet perçu comme un média permettant un degré de participation plus important de la part des étudiants, d'après l'étude de Hrastinski :

« [...] when comparing students that adopted the IM¹⁵ system with those that did not it was found that the adopters operated with a higher level of participation. » (Hrastinski, 2006).

Cette étude prolonge la vision de Nicholson qui avait déjà montré que les étudiants ressentent mieux la présence d'autrui avec la messagerie instantanée (Nicholson, 2002). Les exemples allant en ce sens sont nombreux et la messagerie instantanée est presque systématiquement employée dans nos systèmes collaboratifs synchrones actuels (Veerman *et al.*, 2004). Cette volonté de faire sentir aux individus du groupe qu'ils ne sont pas seuls est dénotée par le terme « conscience de groupe » en TCAO. Le but est d'immerger les apprenants dans une communauté pour abolir les distances (combattre l'isolement social) mais également favoriser l'approche constructiviste pour les situations d'apprentissage.

I.2.3.2 ACAO : Pourquoi et Comment.

Le retour des étudiants

Nous avons dit précédemment que l'ACAO est né lorsque les élèves étudiaient de manière esseulée, par la distance ou le temps. Il est vrai que les étudiants sont demandeurs d'interaction, et donc des systèmes collaboratifs, qui leur permettent d'entretenir leur réseau social. C'est le résultat de l'étude expérimentale menée cette année par Dewiyantia :

The finding revealed that distance learners appreciate the opportunities to work collaboratively. They show positive experiences and are quite satisfied with collaborative learning (Dewiyantia *et al.*, 2007).

D'autres vont jusqu'à parler de « besoin social », qui intrinsèquement favorise la communication entre apprenants (Münzer, 2007). Paradoxalement, cette même étude démontre la faible, voir l'indifférence au média et aux technologies en jeu pour l'ACAO. En d'autres termes, il apparaît que la technologie importe peu, du moment qu'elle permet une situation d'apprentissage collaboratif. Ce qui importe, ce sont les possibilités d'interactions entre les personnes formées. En effet, les étudiants dénigrent souvent les dispositifs d'enseignement à distance à cause de la pauvreté des interactions sociales offertes.

15. IM se rapportent à « Instant Messaging »

Cependant, une ambiguïté actuelle réside dans l'évaluation de la pertinence de l'emploi de l'ACAO. En effet, il est aujourd'hui encore difficile de définir l'impact d'un tel système, tant nous manquons d'outils d'évaluation. Même si la réponse tend à l'affirmative, le débat a commencé avec une argumentation basée sur le fait que l'ordinateur biaise le débat ou le rend incomplet (Hightower & Sayeed, 1995). D'autres études justifient l'emploi d'ACAO. Prenons l'exemple d'Hobman, qui a confronté les situations d'apprentissage dans 39 groupes différents, dont 20 étaient des groupes dans lesquels les étudiants étaient face à face, et 19 utilisaient un système collaboratif de communication :

« There was more process and relationship conflict in CMC¹⁶ groups compared to FTF¹⁷ groups on Day 1. However, this difference disappeared on Days 2 and 3. There was no difference between CMC and FTF groups in the amount of task conflict expressed on any day. » (Hobman, 2002).

Cela tend à justifier, en partie, l'intérêt prononcé des recherches pour l'ACAO. Cette étude montre aussi que collaborer est un objectif d'apprentissage. Néanmoins, cela n'explique pas complètement l'efficacité de l'ACAO. Pour comprendre cela, nous allons nous pencher sur les différents types de communication au sein de ces systèmes, afin d'appréhender les retombées sur les comportements des individus dans de telles situations d'apprentissage.

La communication dans les situations d'ACAO

Le lecteur aura compris que l'objectif de l'ACAO est d'essayer de reproduire les interactions sociales entre les apprenants éloignés géographiquement ou temporellement. Au delà des techniques informatiques mises en jeu, il faut comprendre que cet objectif est lié aux sciences cognitives de l'éducation. Il est difficile de parler d'ACAO sans parler des types de communication qui ont lieu au sein d'un groupe d'apprenants distribué. En effet, c'est le résultat de ces études qui permet de choisir les outils informatiques à mettre en œuvre.

(Moore, 1989) identifie principalement trois types d'interactions au sein d'un système collaboratif éducatif (pour l'apprenant) :

- apprenant-contenu
- apprenant-enseignant
- apprenant-apprenant

C'est cette dernière catégorie d'interactions qui est propre aux systèmes d'ACAO. En outre, lorsque l'on traite la communication au sein d'un groupe, on inclut bien sûr

16. « Computer-Mediated Communication », soit « communication assistée par ordinateur » en français.

17. « Face To Face », traduisible par « face à face » en français.

les individus qui le composent. Il est depuis longtemps reconnu que la composition des groupes a un impact sur l'étude des systèmes d'ACAO. On comprend aisément que les liens émotionnels et sociaux qui entourent les apprenants modifient les communications au sein du groupe, et de ce fait la qualité du travail effectué (Johnson & Johnson, 1987).

Northrup puis Kreijns (Northrup, 2001 ; Kreijns *et al.*, 2002) ont ainsi mis en évidence deux types de messages échangés au sein d'un groupe :

- *messages « orientés tâches »* : envoyés pour faire avancer la connaissance du groupe ou interagir à propos du problème rencontré,
- *messages « non orientés tâches »* : vecteurs d'une communication sociale entre apprenants, ces messages ne sont pas en relation avec la tâche proposée au groupe.

Ces deux auteurs ajoutent que les messages « sociaux » sont majoritaires dans les historiques de communication des groupes qu'ils ont étudiés. Cela renforce l'idée que les étudiants plébiscitent l'ACAO dans l'enseignement distant pour sortir de l'isolement qu'ils ressentent lorsqu'ils sont seuls à apprendre à distance. Cependant, ce biais dans les communications, qui a peu à voir avec l'apprentissage lui même, est néanmoins perçu comme un bien pour l'apprentissage par la communauté de l'ACAO. Il renforce la métaphore du laboratoire local car il reproduit en ligne les interactions sociales qui ont lieu dans la réalité de sessions de travaux pratiques *in situ* (discussions, confrontations de résultats, entraide, émulation, ...)

I.2.4 Bilan

Nous avons montré ce que l'ACAO tentait d'apporter aux apprenants, principalement la « sociabilité » des enseignements traditionnels lorsque ceux-ci s'effectuent à distance. Nous pensons que c'est encore plus vrai dans le cadre de l'expérimentation à distance, où les étudiants confrontent forcément les résultats de leurs manipulations, et échangent autour de leurs compétences et connaissances. Nous sommes convaincus que ce dialogue est nécessaire pour garantir une plus grande efficacité des enseignements pratiques assistés par ordinateur et à distance. Nous sommes, par contre, persuadés que ces outils ne doivent pas se limiter à supporter les échanges verbaux, comme peut le faire la messagerie instantanée par exemple. Sans remettre en cause l'emploi de celle-ci, nous pensons qu'il est nécessaire d'introduire des moyens de piloter collaborativement les dispositifs, pour obtenir une interaction beaucoup plus riche, puisqu'elle ne se limitera pas seulement aux échanges verbaux mais permettra des actions conjointes, au même moment. Cela va donc au delà du simple visionnage de la manipulation faite par l'apprenant, puisque la situation expérimentale est alors non plus conduite par un individu mais par l'ensemble des apprenants.

Néanmoins, lorsque plusieurs personnes participent au même enseignement de manière collaborative, il devient compliqué de déterminer le degré d'implication de chacun. Généralement une telle évaluation se base sur le nombre de messages envoyés par les outils synchrones ou encore le nombre de sessions ouvertes pour les outils asynchrones (Ellis, 2003). Il est cependant dangereux de baser l'évaluation de la participation de chaque individu sur ces simples critères : les différents échanges n'ont pas pour unique but de remplir l'objectif donné, mais aussi d'entretenir les interactions sociales au sein du groupe. En conséquence, le rôle du tuteur dans la manipulation collaborative distante reste prépondérant (Faltin *et al.*, 2002).

Sur la base des ces observations, nous avons choisi de mener des travaux sur la mise en place d'un outil permettant de réaliser des manipulations :

- à distance
- réutilisables
- collaboratives
- les plus réalistes possible

Il reste difficile de s'enfermer dans une communauté scientifique, lorsque le sujet est pluridisciplinaire. Cette caractéristique est d'ailleurs soulignée par la nécessité du réinvestissement des théories de l'apprentissage en sciences humaines et sociales dans le domaine des EIAH (Tchounikine, 2002a), même s'il s'agit d'un « problème de reconceptualisation beaucoup plus complexe » (Tchounikine, 2002b).

Nous ne pouvons bien sûr pas approfondir ce problème dans le cadre de notre étude. Nous nous concentrerons donc sur les propriétés des plates-formes technologiques à mettre en œuvre et les implémentations à proposer. Il est à noter que d'autres problématiques, spécialement celles relatives à la sécurité des systèmes d'informations et aux Interfaces Homme-Machine (IHM), sont au cœur des activités de recherche au sein de notre équipe.

Par conséquent, les principaux domaines auxquels ce travail appartient sont le génie logiciel, plus précisément la croisée de l'EIAH, de l'ExAO et du TCAO (plus spécifiquement l'ACAO). Plus particulièrement, nos travaux viennent combler le vide induit par la collision des domaines de l'ExAO (qui n'est pas nécessairement collaborative) et de l'ACAO (qui n'est pas nécessairement pratique). Nous parlerons alors d'ExCAO pour « Expérimentation Collaborative Assistée par Ordinateur », comme illustré sur la figure I.11. Ce néologisme que nous revendiquons décrit parfaitement la « niche scientifique » à laquelle ce travail se réfère.

Pour les dispositions technologiques à prendre, la réussite d'un système d'ExCAO passe par le fait que l'élève est totalement impliqué dans le travail collaboratif. Cela re-

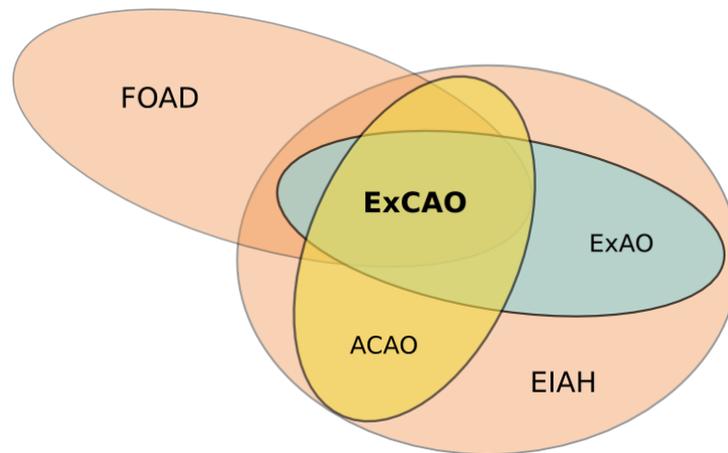


FIG. I.11 – Positionnement de notre travail : l'expérimentation collaborative assistée par l'ordinateur (ExCAO).

quiert des efforts communs entre communautés de l'ACAO et des Interfaces Hommes Machines (Romano *et al.*, 2005).

D'autres publications vont elles aussi dans ce sens, en mettant l'accent sur la grande complexité des EIAH, qui provient de la transdisciplinarité de ce domaine. Nous avons essayé de regrouper ces domaines dans la figure I.12.

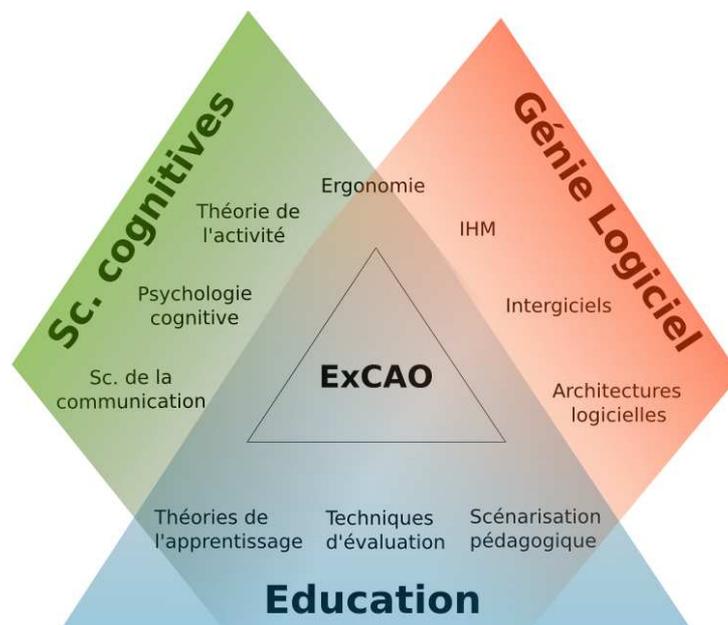


FIG. I.12 – Le tryptique de l'ExCAO : une pluridisciplinarité indéniable.

Ce sont plus particulièrement l'ExAO et les EIAH qui seront les dénominateurs communs des références que nous explorerons dans la section suivante. Nous nous attacherons à établir un état de l'art des solutions techniques existantes, en terme de projets reconnus

ou de publications scientifiques, dans l'optique d'effectuer une étude plus précise sur les propriétés de réutilisabilité et de collaboration des EIAH pour l'ExCAO à distance.

I.3 Etat de l'art

I.3.1 Projets existants

Nous entendons ici dresser la liste des principaux projets identifiés dans la communauté scientifique en terme d'ExCAO. Bien sûr, cette liste ne se veut pas exhaustive, mais vise les principaux projets, avec leur philosophie et les partenaires impliqués. Tous ces projets ont pour objectif de rendre accessibles des travaux pratiques sur Internet. Néanmoins, ils ne voient pas les enjeux des télé-TPs de la même façon ou sous le même angle, il en résulte alors une approche originale quasi systématique.

Pour plus de clarté et de concision, nous avons résumé nos recherches sur ces projets dans le tableau I.3.1 :

Nom	Financement	Originalités
eMerge	Projet européen 2002-2004	<ul style="list-style-type: none"> - Sécurité d'accès - Plannification des accès (réservation de ressources)
ILab	MIT and Microsoft Research	<ul style="list-style-type: none"> - Minimiser le temps d'intégration et d'administration de télé-TPs - Partager le plus largement possible les télé-TPs à travers le monde
Telelabs	Inconnu	<ul style="list-style-type: none"> - Volonté de mesure les performances des apprenants - Optimiser l'organisation entre temps tutoré et temps non tutoré accordés aux apprenants, tout en conservant un apprentissage optimal
PEARL	Inconnu	<ul style="list-style-type: none"> - Outils de communication entre apprenants - Accessibilité des manipulations aux handicapés physiques
Cyberlab	Projet européen ReLAX (Remote LAboratory eXperimentation)	<ul style="list-style-type: none"> - Proposition d'un modèle économique basé sur un fournisseur de services d'expérimentations, médiateur entre utilisateurs et instituts disposant de ressources de hautes technologies

TAB. I.3 – Les principaux projets d'ExAO recensés au niveau mondial.

Les projets recensés sont les suivants : eMerge¹⁸, IILab¹⁹, Telelabs,²⁰ PEARL (Practical Experimentation by Accessible Remote Learning)²¹ et Cyberlab²². On remarque que l'Union Européenne fait un effort significatif pour les télé-TPs, et que les très grandes universités étrangères sont également présentes (MIT pour les États-Unis et l'Open University pour le Royaume-Uni).

I.3.2 Etude bibliographique

I.3.2.1 Cadre de l'étude

Ces projets ainsi que des développements plus concentrés ont abouti à de nombreuses publications, sur lesquelles nous allons nous baser pour établir notre étude bibliographique. Cette étude couvre au final 42 publications²³ relatant la mise en œuvre de laboratoires distants.

Il est tout d'abord important de noter les critères qui nous ont conduit à retenir ces références. Nous avons pris soin d'éviter de citer plus d'une publication faisant état du même projet, dans le but évident de ne pas fausser les résultats et l'analyse de l'existant en terme de répartition. De plus, nous avons limité nos recherches aux dix dernières années, même si nous avons cherché à privilégier les références les plus récentes (8 seulement sont antérieures à 2001, 2001 compris). Par la suite, nous ne relaterons que les chiffres obtenus au regard des caractéristiques qui nous ont semblé pertinentes, à l'exclusion du détail du classement effectué. Le lecteur pourra néanmoins recourir à l'annexe (A) pour le détail de la classification réalisée.

18. <http://www.emerge-project.net>, partenaires (en anglais pour plus d'homogénéité) : University Bordeaux 1, University of Koblenz-Landau, University of Applied Sciences Münster, Technological Education Institute of West Macedonia, University Autonoma of Madrid, Chalmers University of Technology, Graduate Center of Technology (Unik), Czech Technical University in Prague, et "Petru Maior" University Tg-Mures

19. <http://ilab.mit.edu/ServiceBroker/>, Massachusetts Institute of Technology (MIT)

20. <http://telerobot.mech.uwa.edu.au/>, Faculty of Engineering, Computing and Mathematics, The University of Western Australia, School of Mechanical Engineering

21. <http://iet.open.ac.uk/pearl/>, Open University

22. <http://www.cyberlab.org/>, Norwegian University of Science and Technology

23. les références sont les suivantes : Afshari & Payandeh (1999) ; Agarwal *et al.* (1998) ; Aktan *et al.* (1996) ; Alamo *et al.* (2002) ; Ammari & Ben Hadj Slama (2006) ; Bauchspiess *et al.* (2003) ; Bayard *et al.* (2001) ; Bicchi *et al.* (2001) ; Canfora *et al.* (2004) ; Carnevali & Buttazzo (2003) ; Casini *et al.* (2001) ; Cedazo *et al.* (2006) ; Chang *et al.* (2006) ; Chiculita & Frangu (2002) ; Choy *et al.* (2000) ; Colwell *et al.* (2002) ; Delamer & Lastra (2006) ; Dixon *et al.* (2002) ; Dobrogowski *et al.* (2007) ; Esche (2002) ; Faltin *et al.* (2002) ; Ferrero *et al.* (2003) ; Guimaraes *et al.* (2003) ; Gustavsson (2002) ; Hoyer *et al.* (2004) ; Hua & Ganz (2003) ; Hsu *et al.* (2000) ; Kikuchi *et al.* (2001) ; Knight & DeWeerth (1997) ; Lelevé *et al.* (2004) ; Martínez Carreras *et al.* (2005) ; Piguet *et al.* (2002) ; Rohrig & Jochheim (1999) ; Sanchez *et al.* (2004) ; Schwartz & Dunkin (2000) ; Trevelyan (2003) ; Tuttas & Wagner (2001) ; Viedma *et al.* (2005) ; Yan *et al.* (2006) ; Yang *et al.* (2004) ; Zhu (2006) ; Zimmer *et al.* (2003)

Les critères que nous allons observer sont :

- La répartition par année,
- L’origine géographique des publications,
- Le domaine scientifique auquel appartient la manipulation proposée,
- Les solutions logicielles utilisées dans les développements,
- L’usage de l’ACAO dans les plates-formes rencontrées.

I.3.2.2 Analyse

Diagramme de répartition

Par année de publication Dans le but de positionner la fenêtre temporelle dans laquelle notre étude bibliographique s’inscrit, nous avons illustré dans la figure I.13 la répartition temporelle des publications auxquelles nous faisons référence.

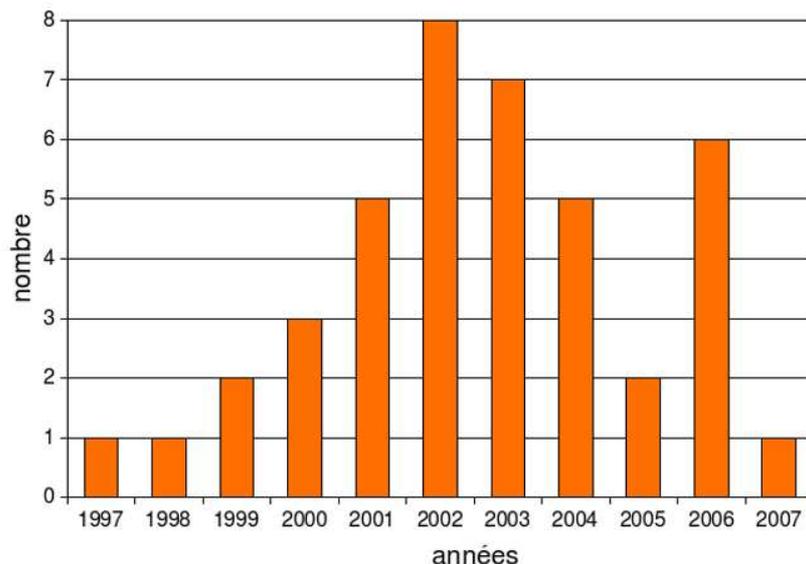


FIG. I.13 – Fenêtre temporelle de notre étude : répartition par année des publications utilisées.

On remarque que la répartition se concentre autour des années 2002 et 2003, ce sont les deux années antérieures au début de ce travail. Une deuxième observation consiste à relever le pic observé en 2006. Le lecteur peut penser qu’il s’agit d’un phénomène lié à l’écriture de ce mémoire (étape bibliographique en début de travail de thèse). En vérité, il n’en est rien, le seul facteur temporel dans la recherche bibliographique était de ne pas mettre en jeu des publications relatant des développements de plus de dix ans. Nous tendons à l’inverse à expliquer ce pic par un regain d’intérêt pour les manipulations distantes. Nous voyons d’ailleurs apparaître seulement aujourd’hui des publications faisant un état de l’art du domaine, comme Ma & Nickerson (2006), ce qui tend à soutenir l’idée que ce domaine scientifique dresse un bilan avant de prendre un nouvel élan.

Par origine géographique Nous dressons ensuite une représentation géographique de l'origine des publications retenues. Pour identifier le pays d'origine d'une publication, nous avons étudié l'université d'appartenance des auteurs. Lorsque des co-auteurs étaient issus d'universités de pays différents, nous avons attribué la publication aux deux pays (c'est la raison pour laquelle le total dans notre figure dépasse 42). Nos résultats sont résumés dans la figure I.14.

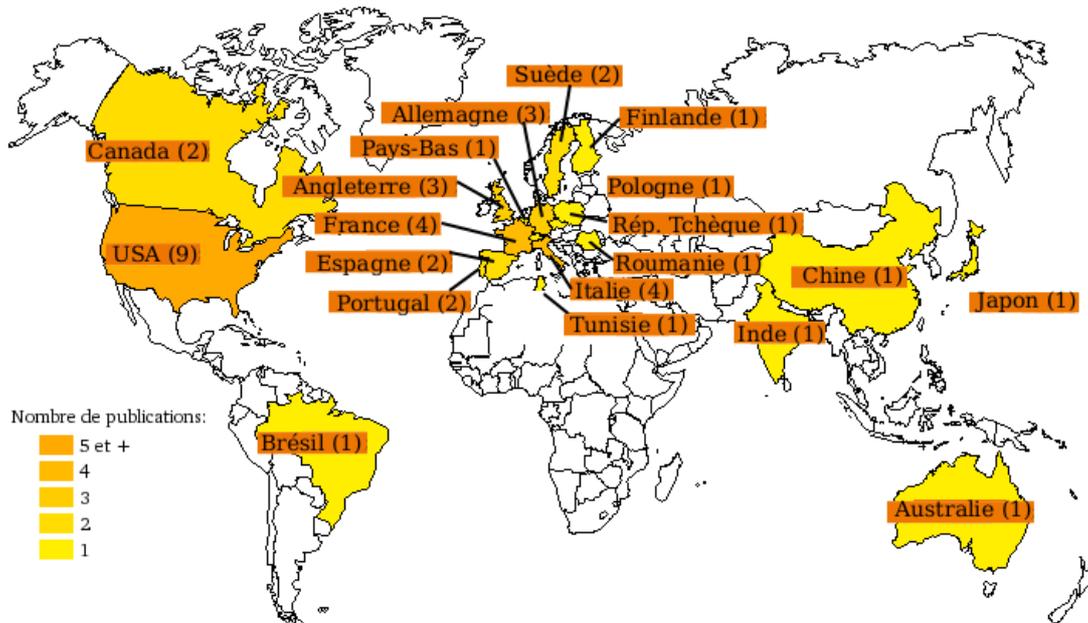


FIG. I.14 – Répartition par origine géographique des publications.

On remarque instantanément que la manipulation distante n'est pas un objectif restreint à une seule zone géographique, la problématique intéresse des chercheurs répartis sur toute la planète. Bien entendu, les États-Unis ainsi que l'Europe (l'Italie et la France en tête) figurent parmi les pays dont les chercheurs sont les plus actifs dans ce domaine. Chaque continent s'intéresse ainsi à ce sujet, le continent africain dans une moindre mesure puisque nous avons trouvé seulement une référence tunisienne.

Par domaine scientifique Nous établissons ici une nouvelle classification de références avant toute analyse. Il s'agit de ranger celles-ci par rapport au domaine scientifique auquel appartient le dispositif manipulé. Bien qu'il soit vrai que beaucoup de publications traitent en fait de travaux pratiques au sujet d'un pendule inversé (Chang *et al.*, 2006 ; Zhu, 2006 ; Sanchez *et al.*, 2004 ; Dixon *et al.*, 2002), le reste du panel étudié est en fait très hétérogène en ce qui concerne le domaine auquel il s'applique (figure I.15).

Il est à noter que par « pluridisciplinaire », nous entendons les références faisant état d'une même architecture ayant été adaptée à plusieurs dispositifs, significativement différents les uns des autres en termes de domaines scientifiques. Une première lecture nous révèle que les domaines de l'automatisme, la robotique, l'électronique et la micro-

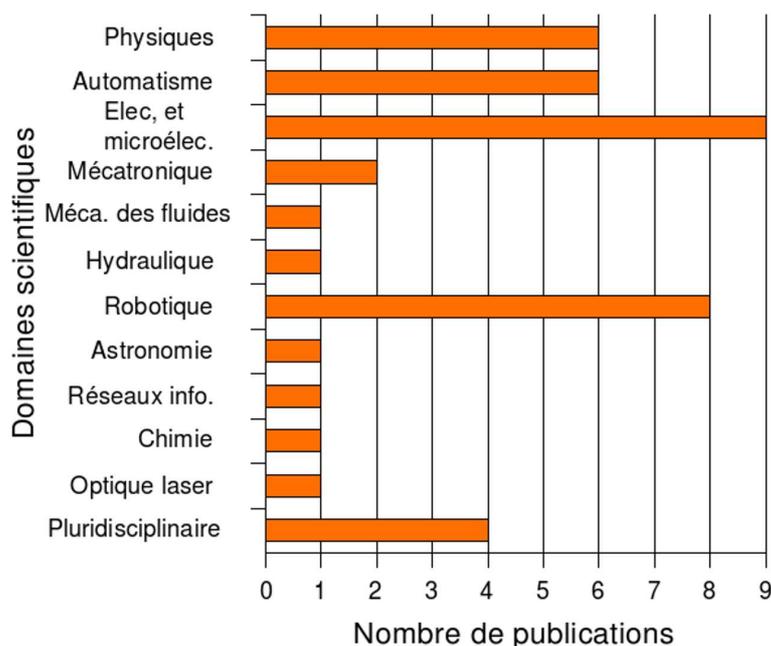


FIG. I.15 – Répartition par discipline scientifique du dispositif contrôlé à distance.

électronique ainsi que la physique (on retrouve ici les exemples relatifs aux pendules inversés) ont une longueur d’avance sur les autres domaines. Nous expliquons cela par le besoin important qu’ont les instituts supérieurs de formation, particulièrement les écoles d’ingénieurs, de dispenser des enseignements pratiques dans ces domaines. Aussi, il est intrinsèquement plus aisé de mener une manipulation à distance dans le cadre de la robotique qu’en chimie ou en astronomie par exemple.

Nous nous sommes contentés jusqu’ici de caractériser l’échantillon que nous avons identifié parmi les publications scientifiques de ces dix dernières années. Bien entendu, le but d’une étude bibliographique ne s’arrête pas là. Notre intérêt étant de construire un système de manipulation distante pour la formation, nous allons nous pencher sur les technologies dominantes dans les solutions logicielles mises en œuvre, et nous essaierons de déterminer si ces solutions suivent les recommandations des théories actuelles pour l’apprentissage par la pratique. Enfin, nous préciserons les limites des systèmes actuels.

Technologies mises en œuvre

Du point de vue technologique, il convient de différencier les deux extrémités de la chaîne de communication : c’est-à-dire d’une part l’interface entre l’ordinateur situé dans le laboratoire et l’appareil lui-même, et d’autre part, l’interface logicielle offerte aux utilisateurs.

Concernant l’interface entre le dispositif réel et l’ordinateur jouxtant la manipulation,

nous pouvons classer les technologies en trois classes principales :

- Logiciels dédiés et propriétaires déjà utilisés dans les universités avant la mise en place. Les deux logiciels souvent cités sont MatLab (avec simulink) et LabView (avec datasocket). Ils représentent une part non négligeable de notre étude (10 publications sur un total de 42). L'inconvénient de ces solutions réside dans le fait qu'elles permettent rarement la réutilisation des développements pour de nouveaux projets, et qu'elles requièrent une compétence supplémentaire pour l'architecte des laboratoires distants, à savoir la maîtrise desdits logiciels.
- Les publications restent très obscures sur les moyens employés pour lier les appareillages à l'ordinateur local. Il peut s'agir de connexions directes TCP²⁴/IP²⁵, ou encore de l'usage de bibliothèques propriétaires comme DAQ²⁶ de la société National Instrument permettant d'utiliser les pilotes des périphériques PXI²⁷. Il s'agit en fait des publications dont nous n'avons pas pu établir avec certitude, par manque d'informations, la technologie employée. Cette catégorie de communications représente 17 publications sur les 42.
- Les autres publications ont été regroupées car elles présentent toutes l'utilisation originale d'un langage ou d'un logiciel, dont l'idée n'a pas été reprise ailleurs. On remarque principalement des solutions basées sur Visual Basic et plus récemment Microsoft .Net, ou encore C/C++. Il est aussi fait état d'une utilisation de python (Chiculita & Frangu, 2002).

Il est relativement difficile d'établir une taxonomie concise et complète des technologies employées pour piloter ne serait-ce que localement les dispositifs. Nous expliquons cela par le fait que les dispositifs sont seulement utilisables via une programmation logicielle en utilisant une bibliothèque propriétaire propre au constructeur de l'appareil. Dans d'autres cas, le dispositif est de « fabrication maison » : il suit rarement un standard ouvert autorisant l'interopérabilité.

A l'inverse, nous allons voir que le segment des interfaces logicielles soulève beaucoup moins d'ambiguïtés, et il est plus facile de dresser la répartition des technologies. C'est ce que nous avons d'ailleurs illustré dans la figure I.16²⁸.

Bien que les applications clientes basées sur le langage Java et les applications Internet basées sur des « applets Java » ne sont pas toujours catégorisées de la même façon dans

24. Transmission Control Protocol

25. Internet Protocol

26. Data AcQuisition

27. PCI eXtension for Instrumentation

28. Le lecteur aura noté que la somme des publications annotées ici ne fait pas 42 mais 60. En réalité, certaines architectures proposées couplent plusieurs technologies. C'est par exemple le cas de (Hoyer *et al.*, 2004) qui couple Java et VRML, ou encore de (Canfora *et al.*, 2004) qui couple Java, C++ et Corba dans la même architecture. De telles architectures vont donc au crédit de chacune des technologies.

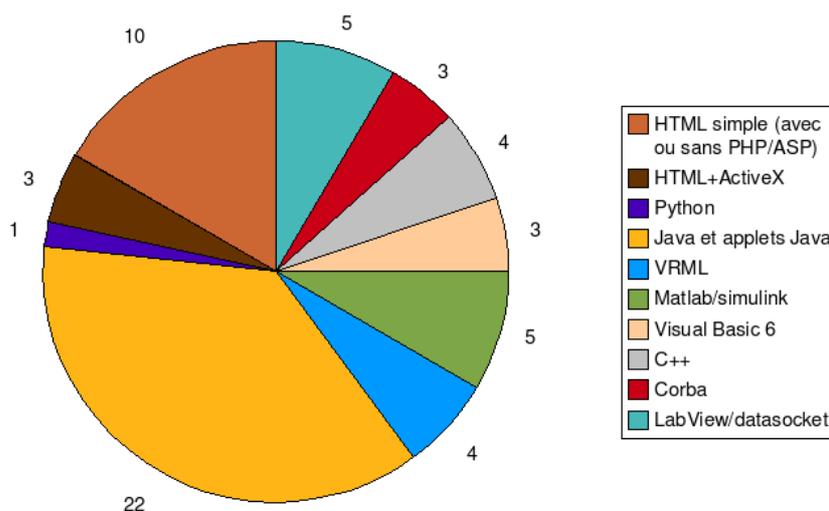


FIG. I.16 – Répartition par solution logicielle du dispositif contrôlé à distance.

des études bibliographiques, nous pensons que les applets Java se rapprochent plus des applications Java que des applications HTML. Néanmoins, le discours rencontré dans la très grande majorité des publications utilisant les applets java incite à penser que seul un navigateur est nécessaire pour assurer l'accès aux solutions logicielles basées sur des applets Java. Nous pensons que cela n'est pas exact : il est obligatoire de télécharger la machine virtuelle Java (plus connue sous l'anglicisme JVM) pour une applet, et c'est le même prérequis, pour un applicatif dit lourd. L'installation du logiciel dans le cadre d'un client lourd peut d'ailleurs être automatisée grâce à la technologie Java Web Start de Sun Microsystems. Nous considérons donc les applets Java comme des clients lourds, au même titre que les applications « stand alone » Java. Par opposition, les applications exclusivement HTML, avec ou sans langage de script comme CGI, PHP, perl ou encore ASP, sont considérées comme les réels clients légers, puisqu'elles ne nécessitent qu'un navigateur, sans modules additionnels. L'emploi de clients légers a un impact sur l'ergonomie du dispositif représenté. En effet, des formulaires Web ne sont pas à même de proposer des interfaces fidèles à la réalité des dispositifs pilotés.

On constate, outre la domination commune des logiciels propriétaires permettant les contrôles de travaux pratiques (LabView et MatLab) et de Java, que le reste des solutions repose sur des choix technologiques hétérogènes. Cela sous-entend que peu d'efforts de réutilisabilité dans le déploiement de laboratoires distants ont été réalisés, ne facilitant donc pas le déploiement de nouvelles manipulations distantes à partir des anciennes. Les seuls projets faisant état de plusieurs laboratoires distants pour la même architecture logicielle suggèrent une forte ré-ingénierie concernant le développement logiciel.

On observe qu'il est possible de construire un laboratoire distant avec pratiquement tous les langages de programmation, mais cela nécessite à chaque fois que le concepteur du laboratoire distant dispose de compétences affirmées en génie logiciel, mais également dans le domaine scientifique auquel appartient la manipulation. Il ne faut pas non plus oublier que ce même concepteur se doit d'avoir les connaissances éducatives nécessaires à l'observation d'une pédagogie adaptée à l'enseignement pratique proposé. À travers cette analyse, nous vérifions la réalité du tryptique de l'ExCAO proposé précédemment (figure I.12), dont les composantes principales sont le génie logiciel, les sciences cognitives et les sciences de l'éducation.

L'ExCAO dans notre bibliographie

Nous avons vu précédemment que le constructivisme était la théorie qui favorisait les expériences d'apprentissage par l'interaction des apprenants (I.2.1.2). Nous avons alors présenté l'ACAO et ses répercussions, ce qui nous a amené à former le néologisme EX-CAO (Expérimentation Collaborative Assistée par Ordinateur), concernant l'apprentissage collaboratif dans le cas des travaux pratiques menés à distance.

Nous nous sommes naturellement posé les questions suivantes pour les références rencontrées :

- Parmi tous ces déploiements de télé-TPs, lesquels embarquent l'aspect collaboratif?
- Parmi ceux qui le supportent, quels sont les outils synchrones mis en œuvre?
- Parmi ces télé-TPs que nous qualifierons de « collaboratifs », comment le partage d'une ressource entre plusieurs utilisateurs distants est-il géré?

Tout d'abord, seulement 12 publications sur les 42 utilisent des outils de communication entre apprenants.²⁹

Sur ces 12 publications, nous avons recherché les outils informatiques permettant aux apprenants de vivre en commun leur expérience d'apprentissage. Le résultat est synthétisé dans la figure I.17.

Contrairement à ce que l'on pourrait supposer, la messagerie instantanée ne tient pas le haut du pavé : c'est le son (via des outils de conférence, VoIP³⁰ ou encore la prise de son embarquée dans des caméras vidéos) qui détrône le dialogue en mode texte. De

29. Il s'agit de (Aktan *et al.*, 1996 ; Agarwal *et al.*, 1998 ; Carnevali & Buttazzo, 2003 ; Colwell *et al.*, 2002 ; Faltin *et al.*, 2002 ; Ferrero *et al.*, 2003 ; Kikuchi *et al.*, 2001 ; Piguet *et al.*, 2002 ; Martínez Carreras *et al.*, 2005 ; Zhu, 2006 ; Hua & Ganz, 2003 ; Hoyer *et al.*, 2004).

30. « Voice over IP », c'est-à-dire voix sur IP

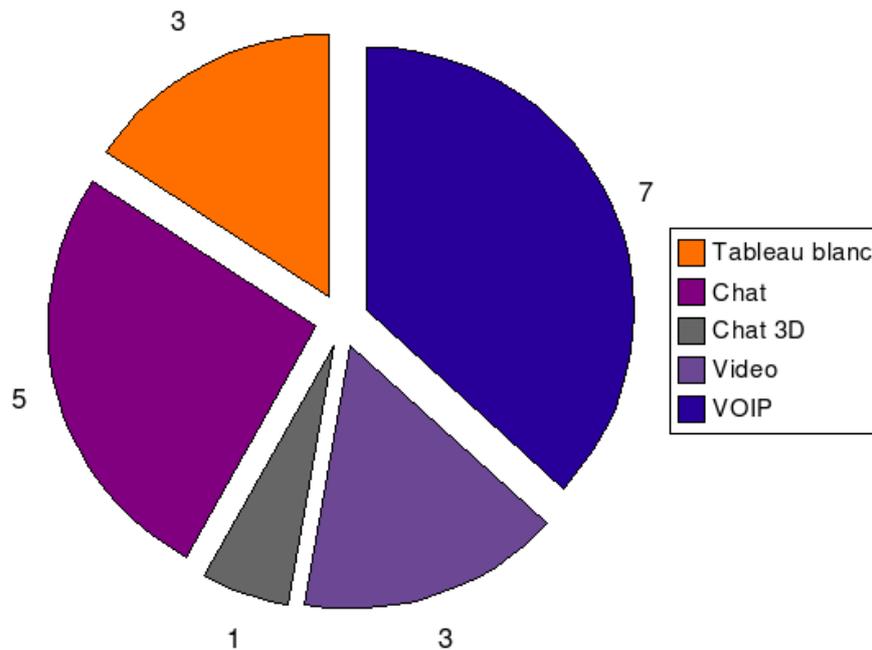


FIG. I.17 – Les outils de communications utilisés pour permettre le dialogue entre apprenants.

manière générale, les solutions retenues se focalisent sur la fourniture de la communication mais très peu sur l’immersion sociale. Une seule publication propose par exemple d’utiliser un univers en trois dimensions dans lequel l’utilisateur est représenté par un avatar (Hoyer *et al.*, 2004). Ainsi, la métaphore qui viserait à reproduire virtuellement une salle de travaux pratiques n’est pour le moment que très peu développée. C’est dommageable puisqu’elle favoriserait les échanges entre apprenants qui se sentiraient probablement beaucoup plus impliqués dans la situation expérimentale que devant un formulaire Web. Cette dernière remarque montre que les solutions collaboratives pour les travaux pratiques favorisent peu la collaboration pour la manipulation elle-même, mais seulement le dialogue entre utilisateurs.

En effet, dans le cadre de travaux pratiques et plus largement des systèmes collaboratifs synchrones dans lesquels une seule ressource est partagée parmi plusieurs utilisateurs, nous nous situons dans un environnement paradoxalement mi-concurrentiel/mi-collaboratif. Les apprenants visent à circonvenir les problèmes pratiques qui leur sont posés (chacun tentera donc de s’accaparer le dispositif distant), mais ils doivent également s’entraider, confronter leurs résultats, etc. puisque certains problèmes se résolvent plus facilement à plusieurs (chacun passera par un stade où il souhaitera collaborer avec autrui). Comment cela se traduit-il dans notre panel?

Très peu de manipulations distantes à but de formation sont collaboratives. Nous sommes pourtant fortement convaincus que fournir des outils ne suffit pas à rendre une manipulation collaborative ! Il faut, en plus, que la ressource pilotée soit elle-même accessible de manière collaborative, c'est-à-dire partagée, et donc que chacun puisse identifier l'auteur d'une action. En effet, piloter un instrument tour à tour, dans un ordre séquentiel, relève peu de la collaboration. À l'inverse, piloter à plusieurs et *simultanément* une même ressource, dénote un caractère collaboratif beaucoup plus prononcé. C'est ici que nous rejoignons exactement les problèmes du domaine scientifique du TCAO, qui parle alors de conscience de groupe (comme nous l'avons évoqué précédemment (§ I.1.1)).

Nous n'avons recensé que trois publications allant dans ce sens. Il s'agit de Sanchez *et al.* (2004) ; Hoyer *et al.* (2004) et Trevelyan (2003) qu'il convient de classer en deux catégories. Tout d'abord Sanchez *et al.* (2004) et Hoyer *et al.* (2004) traitent partiellement le problème des collecticiels pour les travaux pratiques. Ils proposent de régler simplement le problème des accès concurrentiels ou collaboratifs en verrouillant la ressource pour l'utilisateur qui en fait la demande, à l'exclusion des autres donc, en se calquant sur la règle bien connue d'ordonnancement des processus « First In, First Out » (FIFO). Trevelyan (2003) va plus loin puisqu'il permet à tout le monde de visualiser sur son interface logicielle le résultat des actions d'autrui (lorsqu'une action est menée, tout le monde en observe le résultat). Il est par contre dommage que la proposition n'inclut pas la conscience de groupe, puisqu'il est impossible dans cette solution de déterminer l'auteur de l'action à partir de son résultat.

Le constat est donc particulièrement sévère pour toutes les publications que nous avons recensées : l'aspect collaboratif de la manipulation elle-même (au delà de la communication entre apprenants) est largement sous-exploité, alors qu'il s'agit, au même titre que le dialogue entre apprenants, d'une composante de l'ACAO. En effet, alors que le propos du constructivisme est de sociabiliser l'apprentissage, comment envisager des formations pratiques à distance efficaces si celles-ci sont dénuées de collaboration ?

I.3.3 Bilan

Nous dressons ici le bilan de notre étude bibliographique. Celle-ci est basée sur les publications relatives aux manipulations distantes à but de formation des 10 dernières années. En dehors des doublons (publications faisant référence à la même solution logicielle), nous avons recensé 42 références. Nous avons ensuite décrit plus précisément notre panel (répartition par domaine scientifique, par année de publication ou encore par technologie mise en œuvre, dans le but de caractériser et d'appréhender la nature des publications retenues). Notre analyse a ensuite consisté à établir les limites des solutions existantes, d'une part en termes de génie logiciel, d'autre part en termes d'apprentissage collaboratif,

puisque l'ACAO est une proposition supportée par le constructivisme, le courant dominant actuellement les recherches en théories de l'apprentissage.

Tout d'abord les publications présentent un panel large de possibilités d'implémentation, mais nécessitent l'utilisation de la programmation logicielle pour proposer une interface aux apprenants. De plus, cette approche limite fortement la réutilisabilité des applications, puisque l'emploi de la programmation logicielle requiert une ré-ingénierie lourde, et une double compétence de l'architecte (le constructeur du télé-TP), en terme de génie logiciel et concernant le domaine scientifique considéré, voire une triple orientation de ses compétences si on inclut la dimension pédagogique.

De plus, les télé-TPs proposés jusqu'à aujourd'hui prennent peu en compte la nécessité d'une collaboration entre apprenants, même si cela va à contre-courant des théories actuelles de l'apprentissage. C'est d'autant plus frappant que le très faible échantillon de publications ayant implémenté un peu de collaboration se focalise seulement sur l'aspect communication entre apprenants, et pas du tout sur les politiques de partage d'un instrument entre plusieurs apprenants. En effet, si les accès étaient totalement concurrentiels, la politique de partage de la ressource serait évidente : elle se bornerait à des mécanismes d'exclusion mutuelle où la section critique serait la phase de manipulation (protection d'une ressource dans un environnement concurrentiel), à l'image de ce que l'on connaît déjà en ordonnancement des processus. A l'inverse, nous avons expliqué que, certes les accès ont un caractère concurrentiel (chacun veut terminer son exercice de télé-TP) mais il s'agit également de manipuler à plusieurs pour échanger des savoirs (puisque des problèmes sont plus facilement solvables s'ils sont traités de manière collaborative). En ce sens, il serait donc intéressant d'envisager des télé-TPs dont la manipulation elle-même permettrait de supporter la conscience de groupe (déterminer facilement les auteurs de chaque action) et où la politique de partage ne serait pas strictement séquentielle mais à l'inverse évoluerait *avec* les situations de collaboration. Ce n'est en effet pas à la pédagogie de subir la plate-forme logicielle, mais bien à la plate-forme logicielle de s'adapter à la pédagogie retenue !

Enfin, il est certain que les solutions présentées ne visent pas forcément à reproduire le plus fidèlement possible les interactions entre un apprenant et une machine. Certes, une petite moitié de notre panel choisit la vidéo pour reproduire la réalité, mais est-ce suffisant pour que l'apprenant soit capable de clarifier la situation, le jour où il se retrouvera devant la machine physique ? Ne faudrait-il pas plutôt doter les interfaces hommes machines de suffisamment de vraisemblance pour qu'elles se rapprochent le plus possible au dispositif réel, puisqu'il importe aux étudiants, non seulement de confronter les modèles théoriques

à la pratique, mais aussi de développer des compétences ?

Durant notre thèse, nous avons orienté nos recherches de façon à répondre aux questions suivantes :

- Comment garantir facilement une réutilisation aisée des développements pour laboratoires distants ?
- Peut-on confier cette tâche à quelqu'un qui ne sait pas programmer ?
- Comment faire pour que les apprenants puissent communiquer, mais aussi partager une ressource de manière collaborative, au delà du simple aspect coopératif ? (collaboration et coopération s'entendent au sens de Dillenbourg (1999) comme expliqué au § I.2.3.1)
- Que faut-il mettre en œuvre pour que la manipulation soit la plus proche de la réalité possible ?

CHAPITRE I

Vers des télé-opérations collaboratives et réutilisables

Conclusion

Dans ce chapitre nous avons dressé le portrait actuel de l'Expérimentation Assistée par Ordinateur, domaine scientifique soutenu par trois acteurs majeurs: laboratoires, industries et instituts supérieurs de formation.

Les tenants et aboutissants de la manipulation distante de ressources se traduisent par différentes configurations envisageables pour l'architecte d'un télé-TP: laboratoires distants, virtuels ou encore hybrides. Cet architecte se doit néanmoins de respecter les théories actuelles issues des sciences humaines et sociales (principalement l'approche « constructiviste » qui a été abordée dans ce mémoire) pour garantir l'efficacité a priori de son système. Néanmoins, il est très ardu de mener à bien la construction de manipulations distantes puisque ce même architecte doit être doté d'une triple compétence: en génie logiciel, dans le domaine de la manipulation considérée et également en pédagogie.

Par ailleurs, la pluridisciplinarité des télé-TPs collaboratifs ne favorise pas la construction d'une étude bibliographique: ACAO, TCAO, IHM, EIAH, ExAO ou sciences cognitives sont autant de domaines susceptibles d'héberger des télé-TPs. Nous avons alors nommé la croisée de ces domaines scientifiques par l'acronyme ExCAO, pour Expérimentation Collaborative Assistée par Ordinateur.

Lors de notre étude bibliographique, nous avons principalement relevé deux problèmes dans l'existant. Tout d'abord les faibles possibilités de factorisation en terme de développement logiciel pour la mise à disposition de téléTPs, visant à diminuer le plus possible le niveau de compétences requis à l'intégration logicielle. De plus, nous avons évoqué l'inexistence de modalités adaptées permettant la mise en place d'une réelle collaboration au sein des télé-TPs (au delà de la communication entre apprenants). Les deux prochains chapitres viseront respectivement à faire état de nos résultats en terme de généralisation des architectures de téléTPs et de l'embarquement de l'aspect collaboratif dans celles-ci.

Généraliser les dispositifs en ligne

Sommaire

Introduction	73
II.1 Généraliser la chaîne logicielle de télé-TPs	74
II.1.1 Maillon entre ordinateur local et instrument	74
II.1.2 Maillon entre ordinateur distant et ordinateur local	76
II.1.3 Interface Homme Machine	78
II.1.4 Bilan	79
II.2 IHM et connectivité à l'intergiciel	80
II.2.1 Ontologies et Génie Logiciel	80
II.2.1.1 Qu'est-ce qu'une ontologie	80
II.2.1.2 Représenter une ontologie en génie logiciel	82
II.2.1.3 Motivation du choix des ontologies	84
II.2.2 Formaliser l'IHM et la connectivité à l'intergiciel	87
II.2.2.1 Formalisation des concepts et relations	87
II.2.2.2 Edition des ontologies	96
II.2.2.3 Exemples d'ontologies	99
II.2.3 Limitations actuelles	104
II.2.4 Bilan	108
II.3 Implémentation de la plate-forme et de l'intergiciel	109
II.3.1 Généraliser le lien entre IHM et l'intergiciel	109
II.3.2 Protocole applicatif pour l'architecture distribuée	112
II.3.3 Choix de l'intergiciel	115
II.3.4 Serveur d'applications pour les composants métiers	120
II.3.5 Points remarquables de l'implémentation	124
Conclusion	129

CHAPITRE II

Généraliser les dispositifs en ligne

Introduction

Le chapitre précédent a mis en évidence deux faiblesses des solutions logicielles actuelles. Ces deux points, à savoir la difficulté de réutilisation des briques logicielles, et l'absence de possibilité d'apprentissage collaboratif, semblent être les clés d'un développement important et rapide des télé-TPs. Ce chapitre traite de la première de ces deux problématiques, c'est-à-dire ce que nous avons identifié comme factorisable dans la chaîne logicielle de télé-TPs. Nous exposerons tout d'abord un moyen de rendre le travail plus simple (en terme de compétences requises) et plus court (en terme de temps d'intégration). Ensuite, nous présenterons une réflexion sur les briques logicielles mises en place, ainsi que les efforts de développement réalisés dans le cadre de la mise en œuvre d'une plate-forme expérimentale.

II.1 Généraliser la chaîne logicielle de télé-TPs

Pour identifier les éléments factorisables de différentes chaînes logicielles de télé-TPs existantes, étudions leurs architectures qui ont été présentées auparavant (figure I.7). À partir de cette illustration, nous pouvons identifier les trois maillons qui composent la chaîne logicielle reliant les apprenants aux dispositifs distants. Ces trois maillons correspondent aux interfaces entre chaque élément de la chaîne logicielle des télé-TPs, à savoir :

- le maillon entre le PC jouxtant la manipulation et le dispositif lui-même. Ce maillon constitue un pilotage local couramment utilisé en EXAO,
- le maillon entre le PC distant et le PC local. Il s’agit de l’intergiciel (middleware), à savoir la messagerie applicative entre les deux applications s’exécutant sur les deux ordinateurs,
- le maillon entre l’apprenant et le PC distant, que nous avons déjà nommé au chapitre précédent « Interface Homme-Machine » (IHM).

Toutefois, cette figure doit être complétée en identifiant clairement le potentiel de factorisation de chaque maillon (figure II.1, avec en vert ce que nous estimons factorisable, en orange ce qui ne l’est que partiellement).

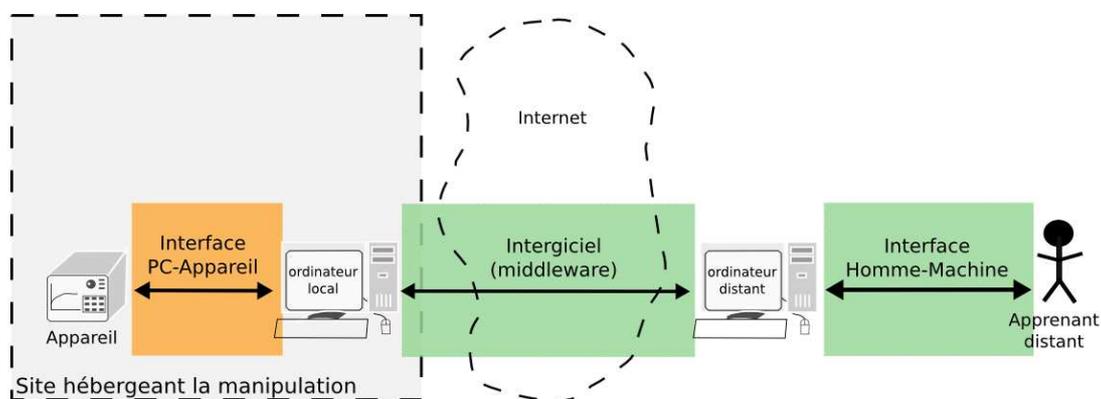


FIG. II.1 – Réutilisabilité des maillons de la chaîne logicielle des télé-TPs.

II.1.1 Maillon entre ordinateur local et instrument

Établir l’interface entre PC local et instrument implique de maîtriser les standards et normes d’accès aux instruments. Ceci est très complexe du fait de l’hétérogénéité et du nombre de technologies à piloter mais également des interfaces matérielles permettant de s’interfacer avec les instruments : PXI, USB, GPIB, ...

On peut néanmoins comprendre que certains instruments, de constructeurs différents ou de séries différentes, proposent les mêmes fonctionnalités et qu’il est donc dommageable de proposer une bibliothèque d’interfaçage par instrument. Il est, par conséquent,

logique de proposer une bibliothèque d'interfaçage par *classe* d'instrument, dans le but d'économiser des développements à même finalité, mais pour des dispositifs différents. C'est ce que proposent les technologies VISA¹ et IVI². Ces dernières visent à rendre les pilotes d'accès aux instruments les plus indépendants possibles de l'instrument piloté, en regroupant les dispositifs en catégories.

Malheureusement, ces nouveaux protocoles souffrent pour nous de plusieurs limitations. Tout d'abord, ce sont des standards très jeunes au regard de l'âge de l'instrumentation assistée par ordinateur. En outre, ils imposent des instruments respectant ces nouvelles normes. Il est clair que ceci exclut les systèmes existants³, même si des tentatives comme (Gutterman, 2004) visent à établir des ponts entre « legacy systems » et les standards actuels d'instrumentation. D'autres limitations imposent la prudence sur l'aspect généralisable d'une interface PC/instrument. (Landragin & Schwartz, 2006) en expose trois :

- « il est clair que les résultats ne seront pas identiques entre instruments dont les performances métrologiques sont très différentes, même si chacun semble parfaitement fonctionner dans l'environnement. »
- « il est évident que des modules qui disposent de fonctions spécifiques, uniques dans leur catégorie, bien que disposant de pilotes IVI, n'assurent pas l'interchangeabilité dans le cadre de ces fonctions spéciales. »
- « le surcroît logiciel qu'implique l'adoption du standard IVI est susceptible de ralentir la vitesse du système : un programme de haut niveau est toujours plus lent à s'exécuter que des appels de bas niveau. »

Nous sommes, à notre tour, tentés d'apporter d'autres limitations :

- de nombreux travaux pratiques sont issus de dispositifs « faits maison », dont on ne peut complètement garantir le respect des normes. Or, cela représente un nombre non négligeable de travaux pratiques dans la réalité.
- nous nous focalisons dans ce travail sur le domaine de l'électronique et la micro-électronique, puisqu'il nous est donné la possibilité de manipuler de tels dispositifs au laboratoire DIOM. Néanmoins, nous soulignons notre volonté d'opérer au niveau de travaux pratiques distants, quelqu'en soit le domaine. Il est certain que le domaine de l'électronique fait, malgré certaines limitations, de gros efforts de standardisation d'accès logiciel aux dispositifs. Ces efforts ne sont pas aussi prononcés dans d'autres domaines scientifiques, comme la chimie ou l'astronomie puisque les

1. Virtual Instrument Software Architecture

2. Interchangeable Virtual Instruments

3. L'omniprésence de l'anglais dans le milieu de l'instrumentation nous poussera à utiliser le terme « legacy systems » pour désigner les systèmes existants.

dispositifs contrôlés sont souvent uniques en leur genre et hétérogènes dans leurs commandes.

Le bilan est mitigé au niveau des possibilités de généralité de ces interfaces, car nous avons vu qu'en pratique ce lien impose quasi systématiquement un développement logiciel dédié. Ces conclusions nous conduisent à attribuer la couleur orange au lien PC/instrument sur la figure II.1.

II.1.2 Maillon entre ordinateur distant et ordinateur local

Penchons-nous à présent sur l'interface, exclusivement logicielle, entre l'ordinateur distant de l'apprenant et l'ordinateur local pilotant la manipulation. Nous parlerons alors d'intergiciel ou de son anglicisme « middleware ». Il s'agit de véhiculer, via Internet, les commandes ou des requêtes entrées par les apprenants, de l'application distante jusqu'à l'ordinateur local à la manipulation, et inversement transférer les résultats des commandes et requêtes à partir des dispositifs jusqu'aux utilisateurs.

Un intergiciel se définit comme suit (Bernstein, 1996) : c'est un lien entre la couche applicative et son système d'exploitation. Son principal but est d'acheminer les messages entre les applications, de la manière la plus transparente possible pour l'utilisateur (mais également pour le développeur). C'est de cette définition issue du monde de la recherche qu'est né le slogan de Big Blue : « Middleware is everywhere. Can you see it? ». Une interprétation consiste à dire que le middleware est partout (comment faire dialoguer nos applicatifs sinon?) mais qu'on ne le voit pas forcément puisqu'un intergiciel de qualité est transparent, invisible.

Il convient de considérer cette maxime en étant conscient de l'objectif : donner un comportement de collecticiels aux télé-TPs. En effet, la commandabilité distante, dans notre idée défendue au chapitre précédent, vise à permettre à plusieurs personnes d'accéder à un dispositif distant au même moment. Du coup, l'intergiciel a pour fonction le transport des messages, du dispositif vers tous les apprenants.

L'intergiciel a donc la charge de sérialisation et de désérialisation des données sur le canal de discussion. Il existe deux approches radicalement différentes : les intergiciels orientés objets et les intergiciels orientés messages. En fait, la littérature ne fait référence qu'aux termes anglais correspondants : « Object-Oriented Middleware » (OOM) et « Message-Oriented Middleware » (MOM). La différence se situe dans la méthode d'accès à un service distant. Les OOMinstancient un représentant local de l'objet distant⁴ et l'uti-

4. on parle alors de « proxy »

lisent comme s'il s'agissait de l'objet distant (la sérialisation et la désérialisation sont alors assurées par ledit représentant). A l'inverse, les MOM n'exposent pas directement des moyens d'accès aux opérations sur l'objet distant mais s'appuient sur une messagerie applicative pour échanger des messages quant aux opérations à effectuer de part et d'autre du lien.

Du fait de sa mission de transport et livraison d'informations, ainsi que de sérialisation/désérialisation, il est tout à fait correct de percevoir l'intergiciel comme une couche logicielle insérée entre couches applicatives et systèmes d'exploitation, comme le propose Bernstein (Bernstein, 1996). L'intergiciel est donc, sous sa forme de MOM ou OOM, un élément essentiel dans les applications distribuées. Il est forcément utilisé dans tous les domaines nécessitant des architectures distribuées. En voici quelques exemples :

- les micro-noyaux dans le domaine des systèmes d'exploitations (Tanenbaum & Woodhull, 2006),
- les jeux en ligne (Bharambe *et al.*, 2002),
- les applications Internet (Malan *et al.*, 1997),
- le nomadisme, en particulier les applications sur téléphones mobiles (Yoneki, 2003),
- ...

Le but n'est pas d'être exhaustif mais de montrer que l'intergiciel est la pierre angulaire des applications distribuées. Il faut mettre cela en perspective avec le fait que nos télé-TPs sont forcément basés sur une architecture distribuée : les apprenants ne se connectent pas tous à partir du même ordinateur et nous avons besoin de mécanismes pour garantir l'intégrité des messages véhiculés sur l'intergiciel.

En conséquence, s'il est choisi correctement, un intergiciel peut satisfaire n'importe quelle situation de téléTP. Selon nous, les principales qualités que se doit d'avoir un intergiciel pouvant servir de base à une architecture de télé-TPs collaboratifs sont les suivantes :

- Asynchronisme dans la livraison de la réponse. En effet, alors qu'un intergiciel synchrone est caractérisé par un appel bloquant pour la méthode de réception de message (tant qu'aucune réponse n'est reçue, le processus du programme est en attente), un intergiciel asynchrone permet de conserver la main sur son application même si le message n'est pas encore réceptionné. Cela peut par exemple être intéressant pour permettre aux étudiants d'exploiter une courbe, alors que l'instrument réalise une opération parfois longue.
- Robustesse et disponibilité pour satisfaire l'objectif de passage à l'échelle. L'Ex-CAO, à l'inverse de l'expérimentation *in situ*, ne limite aucunement le nombre de

personnes connectées à l'expérimentation. En vérité, rien n'empêche qu'un grand nombre d'apprenants soit connecté au même téléTP, au même moment, via Internet. Il faut cependant être vigilant à la montée en charge et à la robustesse des noeuds logiciels supportant la répartition de l'architecture.

- Garantie d'ordonnement qui assure que les messages seront réceptionnés dans l'ordre dans lequel ils ont été émis. C'est une garantie obligatoire pour un intergiciel pilotant un instrument : l'ordre des commandes est capital pour le bon fonctionnement de la manipulation, voire l'intégrité du dispositif contrôlé.

Étant donné ces caractéristiques, il nous est logique d'attribuer la couleur verte pour le maillon représentant l'intergiciel des télé-TPs sur la figure figure II.1.

II.1.3 Interface Homme Machine

Nous avons vu dans le chapitre précédent que, quelle que soit la technologie logicielle retenue par les architectes de télé-TPs, très peu de plates-formes sont partagées entre différents dispositifs. Cela signifie qu'il n'existe pas vraiment de prédisposition à la réutilisabilité de ces architectures en terme d'IHM. Vu que les solutions existantes proposant un accès collaboratif sont rares, il en résulte que les IHM des télé-TPs actuels ne peuvent être réutilisées en l'état pour offrir des télé-TPs collaboratifs pour des parcs de dispositifs hétérogènes.

Dans le but de diminuer le temps d'intégration d'un télé-TPs, nous nous penchons sur les possibilités de factorisation des IHM de télé-TPs collaboratifs. En fait, que doit fournir une IHM de télé-TP ? À cette question nous répondons les fonctionnalités suivantes :

- Visuel fidèle au dispositif réel,
- Interactivité la plus proche de l'interactivité du dispositif technologique,
- Connectivité à l'intergiciel pour l'envoi et la réception des commandes/résultats ainsi que des messages à caractère collaboratif (actions d'autrui et messages texte, audio ou vidéo),
- Richesse de la bibliothèque des contrôles graphiques (ou « widgets⁵ »),
- Assemblage et construction assistés, dans le cas où un développement logiciel est lourd (onéreux et long),
- Indépendance au classes de dispositif piloté (les contrôles assemblés représentent l'IHM d'une manipulation mais pris un à un, rien ne doit être spécifique à un dispositif particulier).

5. D'après Wikipédia, « widget est un mot-valise formé des mots window (fenêtre) et gadget, ce qui signifierait donc gadget de fenêtre », et plus précisément il désigne « un élément de base d'une interface graphique (liste déroulante, bouton, etc.) que l'on peut désigner également par le terme calqué de l'anglais contrôle » (<http://fr.wikipedia.org/wiki/Widget>)

En pratique, les invariants d'un télé-TP à l'autre sont la connectivité à l'intergiciel, et l'assemblage de l'interface. À l'inverse, le visuel et l'interactivité dépendent de la vision que l'architecte a de la manipulation. La richesse de cette représentation dépend de la maturité de la bibliothèque de contrôles graphiques. Un de nos objectifs est alors de proposer une bibliothèque de contrôles graphiques, embarquant nativement la connectivité à un intergiciel asynchrone, tout en permettant une construction assistée de l'IHM basée sur ces contrôles graphiques, puisque l'interactivité et l'indépendance aux dispositifs pilotés en dépendent. En tout état de cause, cette réflexion se traduit par une coloration en vert de ce lien sur la chaîne logicielle des télé-TPs (figure II.1).

II.1.4 Bilan

Il existe manifestement des invariants entre différents développements de solutions logicielles de télé-TP. Dans la suite de ce mémoire, nous allons voir comment nous décrivons l'IHM et la connectivité à l'intergiciel. Bien entendu, nous proposerons les concepts et modèles permettant d'atteindre ce but mais également les technologies logicielles retenues.

Nous illustrerons cela par l'intégration d'une plate-forme que nous avons mise en oeuvre, associée à un cas d'utilisation sur un appareil du laboratoire DIOM.

II.2 IHM et connectivité à l'intergiciel

Suite à l'apport des ontologies dans le domaine médical, il nous a semblé très pertinent de l'introduire dans les définitions des interfaces, c'est ce point que nous allons développer, expliquer et argumenter ici.

II.2.1 Ontologies et Génie Logiciel

II.2.1.1 Qu'est-ce qu'une ontologie

Le petit Robert définit ainsi une ontologie :
« Ontologie (n.f.) : Partie de la métaphysique qui s'applique à l'être en tant qu'être, indépendamment de ses déterminations particulières. »

Le petit Robert ajoute d'ailleurs que cette définition nous est donnée en héritage par Sartre, dans son essai philosophique « L'Être et le Néant, essai d'ontologie phénoménologique ». En effet, le terme « ontologie » est issu de la philosophie. Épistémologiquement, le terme ontologie provient du grec, et l'on pourrait le traduire par « la science d'être » ou encore « l'étude de l'existant ».

Littéralement, l'ontologie est donc la science qui étudie l'être ou l'existant. En philosophie, l'ontologie vise à décrire les concepts et les relations de base de notre monde. C'est pourquoi Sartre souligne qu'il s'agit d'une partie de la métaphysique.

Ce concept a ensuite été repris en informatique et plus précisément en génie logiciel. Nous verrons que ce concept présente des singularités et des nuances selon qu'il est employé en philosophie ou en informatique, quand bien même l'idée conductrice demeure la même. Dans un souci de clarté, nous ferons référence à la philosophie en parlant de « l'ontologie » (dénotant la science étudiant l'être et l'existant) et au génie logiciel en parlant « des ontologies » (au pluriel).

Dans une première approche, nous pouvons donc dire que le but des ontologies est de décrire les choses, plutôt que de se limiter à les nommer. En fait, le but d'une ontologie est de fournir les moyens à l'homme et la machine de se comprendre. Effectivement, si l'ontologie définit un ensemble de concepts propres à l'activité humaine, lisible par la machine, on obtient une structure sémantique interprétable par hommes et machines.

Aux premiers abords, les ontologies paraissent s'assimiler à un vocabulaire, puisque qu'elles énoncent des concepts et leurs définitions. Par exemple, on pourrait définir dans une ontologie le concept « couleur », qui serait l'ensemble des valeurs suivantes : rouge, vert, bleu. Si l'on fait de même pour tous les concepts dont nous avons besoin dans une application, on obtient alors un dictionnaire de concepts, exprimés avec précision.

Néanmoins, il est primordial de comprendre qu'une ontologie est plus qu'un vocabulaire :

« Une ontologie peut prendre différentes formes, mais elle inclura nécessairement un vocabulaire de termes et une spécification de leur signification. Cette dernière inclut des définitions et une indication de la façon dont les concepts sont reliés entre eux, les liens imposant collectivement une structure sur le domaine et contraignant les interprétations possibles des termes. » (Uschold, 1998)

En effet, une ontologie définit également les relations entre chacun des concepts définis. La définition la plus consensuelle et largement acceptée dans la littérature reste celle de Grüber :

« In the context of computer and information sciences, an ontology is a specification of a conceptualization. That is, it specifies the concepts, ideas, relations, abstractions, and so forth in an objective form. The intent is to clarify the meaning, enabling shared understanding. » (Gruber *et al.*, 2007)

C'est en ce sens que le mot se démarque de son origine philosophique :

« [The word "ontology"] refers to the subject of existence. It is also often confused with epistemology, which is about knowledge and knowing. In the context of knowledge sharing, I use the term ontology to mean a specification of a conceptualization. That is, an ontology is a description (like a formal specification of a program) of the concepts and relationships that can exist for an agent or a community of agents. This definition is consistent with the usage of ontology as set-of-concept-definitions, but more general. And it is certainly a different sense of the word than its use in philosophy. » (Gruber, 1993)

Cette différence est due à l'utilisation que l'architecte de système informatique a des ontologies : alors que la philosophie vise à peindre le monde en général, les ontologies en informatique visent un domaine applicatif particulier et concis. Les ontologies sont donc une représentation d'une conceptualisation partagée et consensuelle, dans un domaine particulier. La définition de Grüber est celle très largement adoptée par les chercheurs en Intelligence Artificielle, Web sémantique et Génie Logiciel.

Néanmoins, le but d'une ontologie n'est pas de représenter l'existence dans son intégralité ! En effet, construire une structure sémantique pour modéliser le sens de concepts et leurs relations impose de se limiter aux propriétés essentielles des concepts appréhendés (Bouaud *et al.*, 1994). Évidemment ces propriétés sont trop nombreuses dans le monde réel pour être sémantiquement structurées dans une ontologie. Guarino affine donc la définition de Grüber en soulignant qu'une ontologie est une spécification rendant

partiellement compte d'une conceptualisation (Guarino & Giaretta, 1995). Il faut choisir le niveau de détail, c'est-à-dire la granularité du modèle que l'on souhaite structurer sémantiquement dans une ontologie. Ce degré d'approximation dénote l'« engagement ontologique » pris par le designer de l'ontologie, subséquent à l'objectif visé.

II.2.1.2 Représenter une ontologie en génie logiciel

Une fois les ontologies définies, il nous appartient de les exprimer sous la forme d'un fichier informatique. Dans l'exemple précédent, proposant le concept de « couleur », définir ce concept n'est pas suffisant. Il faut déterminer quelle(s) relation(s) le concept de couleur entretient avec les autres concepts déclarés. Par exemple, si l'on a déclaré préalablement le concept « voiture », on peut alors définir la relation déterminant le fait qu'une voiture possède une couleur. Ainsi la phrase « La voiture est de couleur rouge » en langage naturel est traduisible pour la machine sous la forme d'un triplet sujet, objet, prédicat. La proposition devient alors :

$$\{Couleur, Voiture, EstPeinteEn\}$$

Cette notation peut également s'écrire sous la forme $EstPeinteEn(Voiture, Couleur)$, si bien que les trois formes suivantes sont équivalentes :

$$\left\{ \begin{array}{l} \{Couleur, Voiture, EstPeinteEn\} \\ EstPeinteEn(Voiture, Couleur) \\ \forall Voiture, \exists Couleur / EstPeinteEn(Voiture, Couleur) \end{array} \right. \quad (II.1)$$

En fait, du point de vue informatique, ce genre de triplet (sujet, objet, relation) n'est pas propre aux ontologies, puisqu'il existe déjà dans la sous-couche descriptive nommée RDF (Resource Description Framework). D'après wikipédia⁶,

« Resource Description Framework (RDF) est un modèle de graphe destiné à décrire de façon formelle les ressources Web et leurs métadonnées, de façon à permettre le traitement automatique de telles descriptions. Développé par le W3C, RDF est le langage de base du Web sémantique. Une des syntaxes (sérialisation) de ce langage est RDF/XML. »

RDF est justement un moyen pour exprimer ce triplet. Bien souvent, on utilise XML (Extensible Markup Language) comme une enveloppe de RDF, puisqu'XML favorise la sérialisation. Par ailleurs, RDF est employé dans beaucoup de domaines informatiques, en dehors des ontologies, par exemple :

- RSS (Really Simple Syndication),

6. http://fr.wikipedia.org/wiki/Resource_Description_Framework

- SOAP (Simple Object Access Protocol) / WSDL (Web Service Definition Language),
- UML (Unified Modeling Language),
- DCMI (Dublin Core Metadata Initiative),
- UDDI (Universal Description, Discovery and Integration),
- XUL (XML-based User interface Language).

On peut donc dire que RDF est un modèle de syntaxe sur lequel le web sémantique s'appuie pour créer des sur-couches propres à chaque domaine listé précédemment, et dont XML assure généralement la sérialisation (figure II.2).

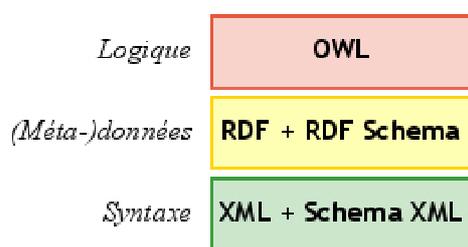


FIG. II.2 – Modèle en trois couches du web sémantique : sérialisation (syntaxe), description (données), logique.

Concernant les ontologies, le format le plus répandu, issu du W3C (World Wide Web Consortium) se nomme OWL (Ontology Web Language). OWL s'appuie souvent sur RDF/XML pour assurer sa sérialisation. Néanmoins, OWL peut aussi être stocké dans une base de données (on parle alors de « triplestore »). Alors que nous devrions utiliser le terme de « OWL/XML » (le cas échéant), nous commettrons l'abus de langage de parler de OWL (voire « d'OWL »), de la même façon que le terme RDF employé dans la littérature fait en réalité référence à « RDF/XML ».

OWL a été créé car RDF ne permet pas de formaliser toute la logique descriptive nécessaire aux applications sémantiques, puisque RDF est principalement un schéma de base. Plus précisément, par rapport à OWL, RDF, d'après (Lapique, 2006), ne permet pas :

- d'exprimer que deux concepts sont disjoints. Par exemple, les concepts homme et femme sont disjoints : une « chose »⁷ ne peut pas être à la fois un homme et une femme.
- de créer des classes⁸ par combinaison ensembliste d'autres classes (intersection, union, complément). Par exemple, on veut construire la classe Personne comme

7. par référence à owl:thing propre au schéma OWL

8. Le mot « classe » sera parfois employé au lieu du mot « concept ». Il convient de garder à l'esprit qu'il ne s'agit pas tout à fait des mêmes notions puisqu'une classe est ce qu'on peut appeler la « réalisation d'un concept. »

l'union disjointe des classes des hommes et des femmes.

- de définir des restrictions sur le nombre d'occurrences de valeurs que peut prendre une propriété. Par exemple, on ne peut pas dire qu'une personne a exactement deux parents.
- de définir certaines caractéristiques des propriétés : transitivité (par exemple : est-PlusGrandQue), unicité (par exemple : estLePèreDe), propriété inverse (par exemple : mange est la propriété inverse de estMangéPar).

C'est justement ce que OWL permet, et c'est en ce sens que OWL permet d'écrire des ontologies : l'expressivité offerte vise non seulement à décrire les concepts comme en RDF, *mais également les relations entre ces concepts!* D'ailleurs, l'empilement des outils sérialisation, description de données, description de relations et application de logique est résumé par le W3C par la pile technologique du Web, périodiquement mise à jour, dont une vue assez précise est reproduite par la figure II.3⁹.

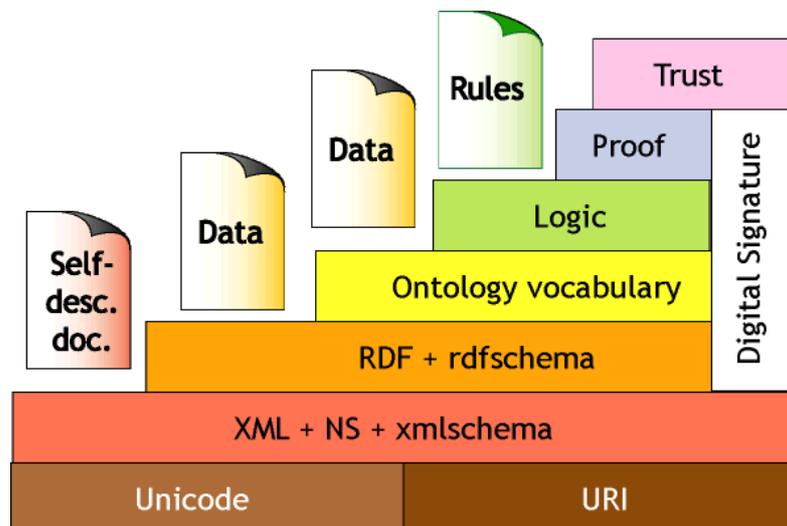


FIG. II.3 – Pile du Web Sémantique (*Semantic Web Stack*), d'après Tim Berners-Lee du W3C (*Berners-Lee, 2000*).

II.2.1.3 Motivation du choix des ontologies

Un des principaux avantages des systèmes basés sur des ontologies est qu'ils visent à être compréhensibles à la fois par la machine et par l'homme. Cette approche est souvent appelée « conceptual modeling » (Guarino, 2006). Plus généralement, on parle de Systèmes à Bases de Connaissances¹⁰ (SBC). Au delà de l'aspect purement communicationnel entre homme et machine, les SBC sont utilisés la plupart du temps pour des raisons d'interopérabilité en différents systèmes. En effet, lorsque chacun emploie un vocabulaire et

9. URI: Uniform Resource Identifier

10. ou « Knowledge Based Systems » dans la littérature (KBS)

des relations entre concepts propres à son domaine, il devient difficile de faire interopérer deux visions différentes de la même chose. On peut donc dire que les ontologies sont la formalisation d'un consensus autour d'une structure sémantique commune, pour des applicatifs interconnectés.

Les SBC interviennent alors pour représenter la carte des concepts et leurs relations, et la sérialiser dans un fichier informatique. C'est le cas, par exemple, pour les systèmes informatiques à grande échelle (Oberle *et al.*, 2006). À travers cet exemple, il est clair que lire des lignes de code ne suffit pas à comprendre la finalité d'un système, d'autant plus que l'on parle de systèmes à grande échelle. En outre, les architectes de ces applications doivent faire face à une grande hétérogénéité des informations. Les SBC leur permettent alors de décrire de manière claire, précise et concise (dans une approche « ni plus, ni moins ») les informations qu'ils gèrent. D'autres exemples de SBC appartiennent à la communauté des Systèmes Multi-Agents (SMA). D'autres applications existent dans le domaine des « Content Management Systems » (CMS) (Snae & Brueckner, 2007).

Dans tous les cas, les SBC ont pour finalité de réduire la difficulté d'intégration et d'utilisation des logiciels: alors que le degré de compétences techniques pour intégrer un SBC augmente (compétences en génie logiciel mais aussi connaissance du domaine considéré), la complexité du problème adressé diminue puisque les outils basés sur les concepts identifiés sont plus intuitifs et accessibles que les langages de programmation. Nous assistons alors aujourd'hui à la montée en puissance des systèmes d'informations basés sur des ontologies (« Ontology-driven Information Systems » (Bottazzi & Ferrario, 2005)).

D'après (Guarino, 1998), les apports des SBC pour l'ingénierie des systèmes d'information sont :

- Réutilisation, partage par plusieurs logiciels,
- Fiabilité dans l'approche consistant à automatiser les vérifications logicielles,
- Spécification, car dresser une ontologie permet de dresser le périmètre de ses besoins,
- Interopérabilité, puisque les ontologies deviennent de plus en plus le format d'échange des cartographies de concepts des systèmes d'information hétérogènes.

Selon nous, le critère d'interopérabilité est un atout majeur pour notre domaine. En effet, pour pouvoir garantir des solutions logicielles de télé-TPs, il faut veiller à proposer le couplage le plus faible possible entre les briques logicielles de la plate-forme et la manipulation assistée par ordinateur elle-même.

Citons l'exemple de (Yan *et al.*, 2006) qui base une architecture de télé-TPs sur des Web Services ainsi qu'une représentation XML des instruments. La finalité de cette approche, très élégante au demeurant, est clairement proche de la nôtre. Il s'agit de promouvoir le couplage faible (« loose-coupling ») des logiciels et interlogiciels, tout en facilitant l'intégration des IHM des clients de la plate-forme de télé-TPs, par une description assistée de celle-ci. Ce qui diffère dans notre vision, c'est que nous percevons l'IHM et l'interlogiciel des télé-TPs comme appartenant à la couche « logique » et non pas à la couche « métadonnées » (description des données). Ces couches ont été présentées précédemment (figure II.2).

En effet, pour nous, les questions qui seront celles des télé-TPs de demain, voire d'aujourd'hui (Gravier *et al.*, 2006a) et (Gravier *et al.*, 2006b) sont :

- Comment proposer une IHM adaptée aux compétences de l'apprenant au regard des actions qu'il mène ?
- Comment évaluer un apprenant à partir des actions qu'il a mené durant la session de télé-TPs ?
- Comment aider l'apprenant défaillant ?
- Comment fournir la transparence de localisation des instruments et le déménagement d'un télé-TP vers un autre instrument géographiquement éloigné en cours des manipulations ?
- Comment construire des manipulations mettant en oeuvre plusieurs instruments (bancs de mesure) ?
- Comment assurer et sécuriser l'accès à plusieurs au dispositif ?

Il est certain que les SBC, impliquant une surcouche logique à la couche descriptive, aident en cela. En effet, s'il est possible de construire une ontologie des IHM des clients de laboratoires distants, il devient alors envisageable d'utiliser ces mêmes concepts pour raisonner, puisque l'on se place au niveau logique de la pile du web sémantique. À l'inverse, avec seulement des fichiers descriptifs des IHM, on se borne à de la journalisation et non pas à une inférence sur les actions menées. Si l'on est capable de produire des algorithmes basés sur des concepts représentant les IHM des clients logiciels des télé-TPs, on peut alors apporter une réponse aux trois premières questions soulevées précédemment. Toute la différence se situe dans le fait que l'on peut faire des déductions en se basant sur des concepts, alors que c'est impossible en employant seulement des descriptions au sens de RDF.

Concernant les deux dernières questions, les ontologies permettent de fournir le niveau d'abstraction nécessaire pour s'affranchir de la connaissance de la ressource instrumentale pilotée. Ceci signifie que le système n'a pas besoin de savoir à quel instrument la manipula-

tion en cours est relative, pas plus que son emplacement géographique. Idéalement, les ontologies permettent de capter la connaissance nécessaire et de la produire de manière informatique, ce qu'une enveloppe descriptive niveau XML/RDF ne peut suffire à représenter. On trouve alors dans OWL un moyen d'expression sémantique suffisant pour subvenir à l'interopérabilité des télé-TPs.

Bien sûr, toutes les ontologies ne sont pas du même niveau. Dans le cadre de ce mémoire, nous nous limiterons à la distinction suivante (Guarino, 1998) :

- *les ontologies de haut niveau*, partagées entre plusieurs domaines (on parle encore de « core ontologies »),
- *les ontologies de domaine*, propres à un domaine considéré.

En fait, les ontologies de domaine s'attachent à décrire les éléments d'un domaine précis, dont le périmètre est clairement délimité. A l'inverse des ontologies de haut niveau, les ontologies de domaine n'ont de signification que dans le domaine considéré. On peut souvent observer que les ontologies de domaine sont le résultat d'un phénomène de capillarité (ou « top-down ») des ontologies de haut niveau pour un domaine donné.

En conclusion, nous pensons que les challenges présents et futurs des télé-TPs reposent sur des problèmes de sémantique, et qu'il convient de les traiter en tant que tel, dans le périmètre du domaine considéré. En effet, la surcouche logique fournie par OWL au dessus de XML/RDF permet de répondre aux attentes des EIAH pour télé-TPs de demain, attentes alimentées par les théories de l'apprentissage d'aujourd'hui, par les enseignants et par les apprenants eux-même. La suite de cette section consiste à présenter nos ontologies permettant de décrire les concepts que nous identifions dans l'IHM des clients logiciels des télé-TPs ainsi que les relations que ces concepts nourrissent entre eux.

II.2.2 Formaliser l'IHM et la connectivité à l'intergiciel

II.2.2.1 Formalisation des concepts et relations

Pour représenter l'IHM d'un dispositif, avec ni plus ni moins de détails que nécessaire, nous avons besoin d'une représentation des connaissances de notre domaine (spécialisation de notre conceptualisation (Gruber, 1993) des IHM de télé-TPs). C'est le rôle que OWL va jouer ici pour nous.

Pour comprendre notre approche, prenons le cas du dispositif fictif de la figure II.4, comme exemple.

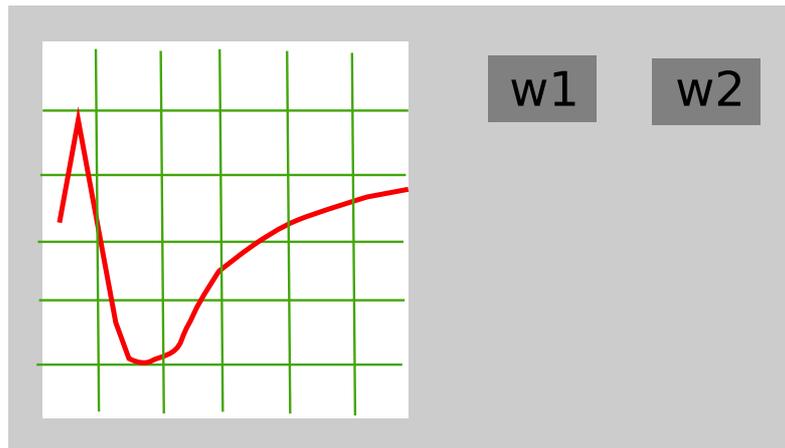


FIG. II.4 – *Un instrument fictif pour illustration de notre démarche.*

Imaginons que cet instrument fictif et basique offre les fonctionnalités suivantes :

- Affichage des données sous la forme d'une courbe en coordonnées cartésiennes sur action du widget « w1 »,
- Affichage des données sous la forme d'une courbe en coordonnées polaires sur action du widget « w2 ».

Nous dégageons de l'observation les concepts suivants pour notre exemple :

- l'IHM de l'instrument est composée de plusieurs widgets et d'une zone représentant des données sous plusieurs formes
- Les widgets se trouvent être des boutons dans ce cas précis
- Les widgets sont à des positions définies
- Les widgets ont une certaine taille
- Les textes éventuels se représentent dans une certaine police de caractères avec une certaine taille

Bien entendu, ces concepts ne sont pas indépendants. Certains sont des ressources permettant d'apporter des propriétés précises à un premier concept. Les widgets sont liés à leur localisation par une relation fonctionnelle, que l'on pourrait nommer « hasLocation » par exemple. L'ensemble de ces observations est résumé sur la figure II.5. L'objectif d'une ontologie demeure d'être partagée entre plusieurs acteurs du même domaine, voire de domaines différents. Comme le domaine du génie logiciel fait un usage important de l'anglais et que notre objectif est de distribuer notre ontologie, nous nous sommes attachés à nommer nos concepts et relations dans la langue de Shakespeare.

De telles observations sont, bien sûr, seulement valables pour des instruments disposant d'un panneau de commandes à deux dimensions. Notre propos vise dans ce mémoire les

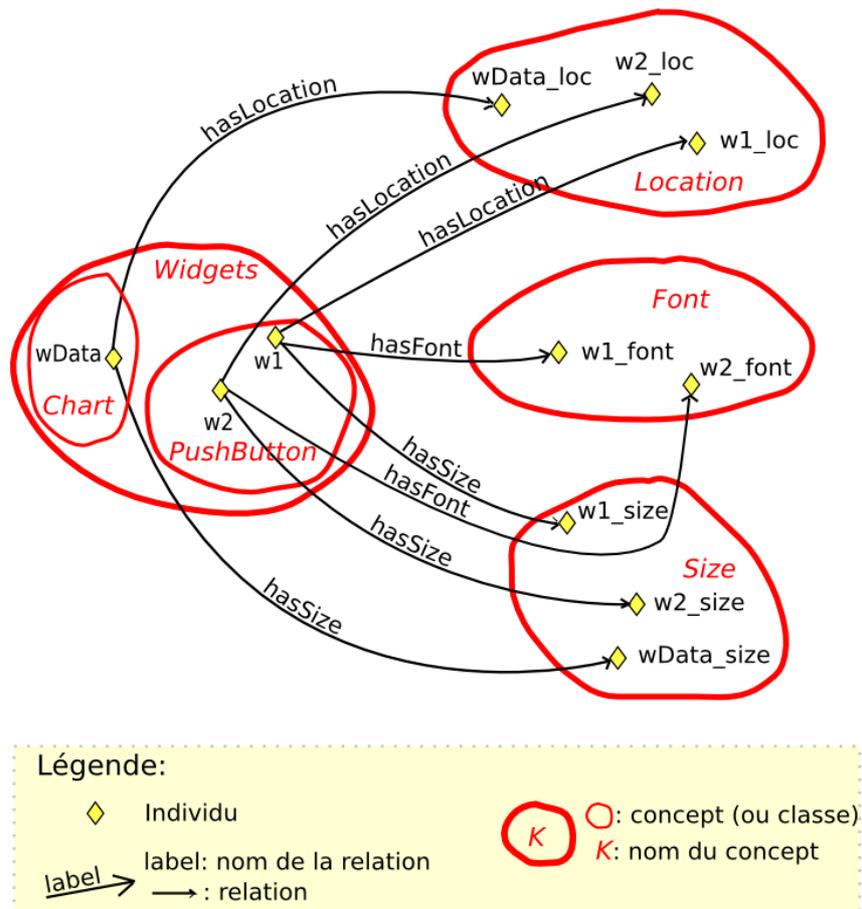


FIG. II.5 – Première vision de l'IHM d'un instrument à partir de l'instrument fictif de la figure II.4

dispositifs pour l'électronique et la micro-électronique, et plus précisément les¹¹ :

- Oscilloscopes
- Analyseurs de spectres
- Analyseurs logiques
- Générateurs de signaux
- Dispositifs de tests pour la vidéo et le son
- Dispositifs de diagnostics de ligne

Puisque l'aspect de ces instruments a une importance pour le travail que nous menons, la figure II.6 fournit un exemple de façade de ces dispositifs pour chaque famille.

Par ailleurs, d'autres dispositifs peuvent être indirectement concernés par nos travaux. Il s'agit d'instruments appartenant à des domaines différents mais dont le principe d'une façade reste valable. Nous illustrons cette idée par un détecteur à chimiluminescence¹² que l'on rencontre en chimie, cf. figure II.7.

11. La liste est probablement non exhaustive, puisque nous considérons les domaines plus que connexes



FIG. II.6 – Exemples d'instruments de télécommunications, électronique et micro-électronique directement visés par notre ontologie



FIG. II.7 – D'autres domaines scientifiques nécessitent des instruments à façade. Exemple ici en chimie avec un détecteur à chimiluminescence.

On comprend donc que d'autres domaines scientifiques soient directement intéressés par nos résultats, puisque les dispositifs propres à ces domaines sont constitués d'une façade, à l'image des instruments présentés à la figure II.6.

Par extension de notre première approche de la figure II.5, nous avons construit une ontologie, la plus simple possible pour l'IHM d'un instrument. Cette ontologie est présentée par la figure II.8.

suivants : électronique, micro-électronique, télécommunications, réseaux, optique,...

12. « La chimiluminescence est un phénomène de réaction chimique ayant pour conséquence la production de lumière. Une réaction de ce type est l'oxydo-réduction du luminol (3-aminophthalhydrazide) par l'eau oxygénée, par exemple, ou un quelconque hydroxyde. » Wikipédia, <http://fr.wikipedia.org/wiki/Chimiluminescence>

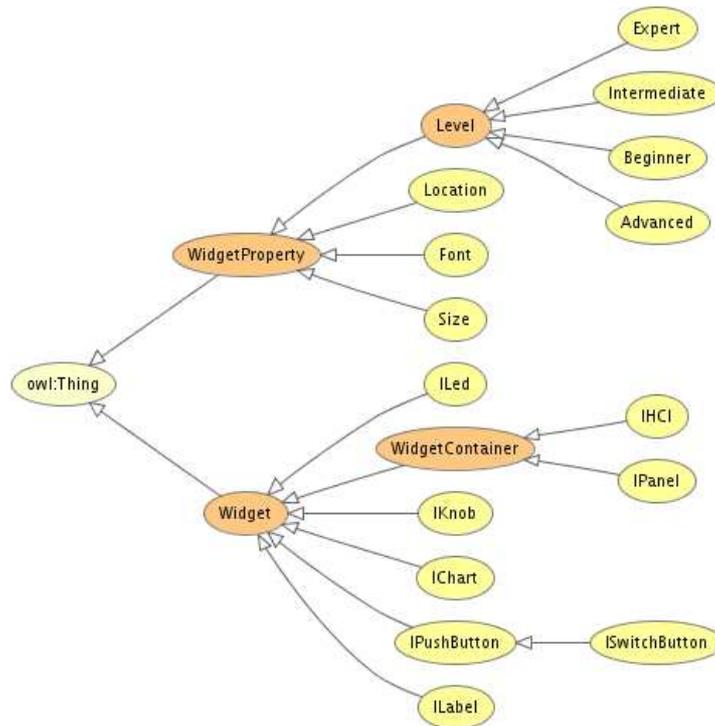


FIG. II.8 – Ontologie d'une façade deux dimensions d'un instrument de mesure. En jaune les classes primitives, en orange les classes définies.

Si l'on se remémore la définition et l'utilité d'une ontologie comme présentées précédemment (c.f. II.2.1), on se souvient qu'une ontologie est plus qu'une carte de concepts, dans le sens où elle représente également les relations entretenues entre ces concepts. Dans notre exemple, on identifie, par exemple, que les concepts « Widgets » et « IPushButton » sont liés par une relation de filiation, puisque IPushButton « est un »¹³ élément graphique. D'autre part, « Widgets » et « Size » sont, eux aussi, liés sémantiquement par le fait qu'un élément graphique possède une taille. Nous avons choisi de nommer cette relation « hasSize ». Les relations existantes sont présentées dans la figure II.9.

Avant d'entrer dans le détail des relations, attardons nous sur les concepts présentés à la figure II.8. Ces concepts sont représentés par des classes. Nous avons représenté en jaune les classes primitives et en orange les classes définies. Les classes primitives sont reconnaissables par l'absence de condition nécessaire et suffisante permettant de justifier l'appartenance d'un individu¹⁴ à cette classe. Il s'agit en fait de classe servant de base à la structure sémantique. À l'inverse, les classes définies sont les classes dotées de conditions nécessaires et suffisantes. Un individu vérifiant ces conditions appartient *donc* à cette classe. Bien sûr, les conditions nécessaires et suffisantes sont exprimées en fonction de concepts déjà existants dans l'ontologie (les classes primitives!). L'assemblage basé sur

13. L'expression « est un » est accompagné de guillemets pour faire allusion à la relation « is a » défini en OWL-DL, dans le cadre de relation de filiation (« subsumption relationships »)

14. En web sémantique, l'individu est au concept (ou classe), ce que l'instance est à une classe en programmation orienté objet, à savoir une *réalisation* de la classe.

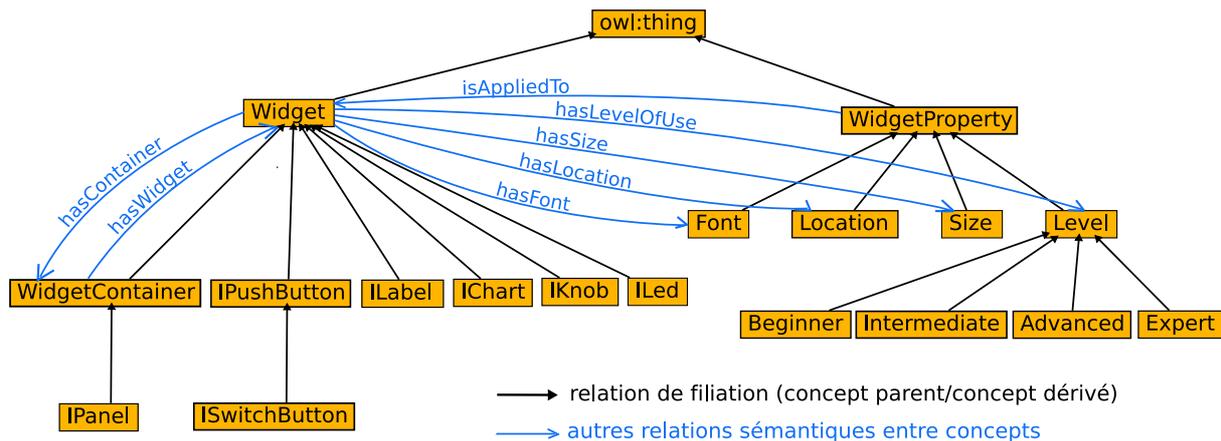


FIG. II.9 – Relations entre concepts de notre ontologie : filiations et relations sémantiques.

les classes primitives pour aboutir à la conceptualisation de classes définies est d'ailleurs comparé au jeu de Lego par (Rector *et al.*, 2004).

Il existe d'autres critères relatifs aux concepts en OWL que nous ne détaillerons pas ici, puisque ce n'est pas l'objet de ce mémoire. Le lecteur averti doit déjà connaître les propriétés de disjonction, closure, restriction, hypothèse d'ouverture par défaut, cardinalité, ... La lecture des spécifications de OWL publié par le W3C est une très bonne lecture pour appréhender ces concepts (Bechhofer *et al.*, 2004). Une fraction de ces spécifications, la plus significative pour ce mémoire, est rapportée en annexe B.

Un simple graphe tel que la figure II.9 permet de visualiser la carte sémantique des concepts et leurs relations en un seul coup d'œil mais ne permet malheureusement pas de montrer toutes leurs propriétés et toutes les relations mises en jeu. Deux tableaux ont pour objectif de remplir cette demande. Dans ceux-ci, les opérateurs « *allValuesFrom* » (respectivement « *someValuesFrom* ») en OWL sont les équivalents de « n'importe quel » (respectivement « au moins un ») en langage naturel.

Le tableau II.1 présente alors le détail des concepts de notre ontologie. Dans ce tableau, la colonne « Type » emploie les lettres D (respectivement P) pour signifier une classe définie (respectivement primitive).

Le tableau II.2 vient en complément puisqu'il précise les propriétés des relations entre les concepts avec lesquels s'assemblent les classes définies. Nous renvoyons le lecteur à l'annexe B pour l'explication des propriétés émargées par ce tableau.

On constate que notre ontologie est plus qu'une liste de classes. Nous avons pris soin de contraindre le domaine que nous décrivons de la manière la plus précise, mais également la plus ouverte possible, pour pouvoir faciliter l'extension. En effet, nous avons choisi de

Nom de la classe	Type	Classe parente	Restrictions	Disjointe de
Widget	D	owl:thing	hasFont allValuesFrom Font hasSize allValuesFrom Size hasLocation allValuesFrom Location hasLevelOfUse allValuesFrom Level	WidgetProperty
IChart	P	Widget	is-a Widget	Ilabel, Iled, IPushButton, IKnob
ILabel	P	Widget	is-a Widget	IChart, Iled, IPushButton, IKnob
ILed	P	Widget	is-a Widget	IChart, ILabel, IPushButton, IKnob
IKnob	P	Widget	is-a Widget	IChart, ILabel, Iled, IPushButton
IPushButton	P	Widget	is-a Widget	IChart, ILabel, Iled, IKnob
ISwitchButton	P	IPushButton	is-a IPushButton	∅
WidgetContainer	D	Widget	hasWidget allValuesFrom Widget	∅
IPanel	P	WidgetContainer	is-a WidgetContainer	∅
IHCI	P	WidgetContainer	is-a WidgetContainer	∅
WidgetProperty	D	owl:thing	isAppliedTo allValuesFrom Widget	Widget
Font	P	WidgetProperty	is-a WidgetProperty	Level, Location, Size
Level	D	WidgetProperty	Beginner or Intermediate or Advanced or Expert	Font, Location, Size
Location	P	WidgetProperty	is-a WidgetProperty	Font, Level, Size
Size	P	WidgetProperty	is-a WidgetProperty	Font, Level, Location
Beginner	P	Level	is-a Level	Intermediate, Advanced, Expert
Intermediate	P	Level	is-a Level	Beginner, Advanced, Expert
Advanced	P	Level	is-a Level	Beginner, Intermediate, Expert
Expert	P	Level	is-a Level	Beginner, Intermediate, Advanced

TAB. II.1 – Concepts hébergés dans notre ontologie des façades d'instruments.

Nom de la propriété	Domaine	Portée	Fonctionnelle	Fonctionnelle inverse	Symétrique	Transitive	Propriété inverse
isAppliedTo	WidgetProperty	Widget	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasContainer	Widget	WidgetContainer	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	$(hasContainer)^{-1} = hasWidget$
hasFont	Widget	Font	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasLevelOfUse	Widget	Level	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasLocation	Widget	Location	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasSize	Widget	Size	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasWidget	WidgetContainer	Widget	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	$(hasWidget)^{-1} = hasContainer$

TAB. II.2 – Propriétés liant sémantiquement les concepts de la façade entre eux.

suivre une approche pragmatique. Deux possibilités s’offrent au concepteur d’ontologie au moment de sa construction :

- il choisit de bâtir et utiliser ses ontologies dans un domaine d’application fermé
- il permet le plus possible à autrui d’étendre son ontologie et cherche à imposer son ontologie pour le domaine considéré (et ainsi éviter d’obtenir plusieurs ontologies représentant le même domaine mais de manière différente)

Le premier choix, dans le cas où le domaine est fermé, permet d’obtenir une ontologie la plus complète et fidèle possible par rapport à l’objectif applicatif à remplir. À l’inverse, les restrictions sémantiques de type « closure » ont l’inconvénient de limiter les possibilités d’extension sans remettre en cause l’ontologie de base.

La seconde possibilité vise à être beaucoup plus permissive. On accepte de diminuer la précision sémantique de son ontologie pour favoriser la possibilité d’extension. Bien entendu, moins l’ontologie a de contraintes, plus il est aisé d’ajouter des concepts et relations à celle-ci. Néanmoins, diminuer les restrictions et les concepts a un impact sur le degré de représentation de la réalité. C’est bien souvent un problème de compromis.

Nous avons retenu la seconde solution, tout en veillant à fournir le minimum de sens permettant à l’ontologie de représenter une façade en deux dimensions. Ce choix a été guidé par le souhait de proposer une ontologie extensible pour satisfaire toutes les façades à deux dimensions que nous pourrions rencontrer (ne perdons pas de vue que notre objectif est de proposer une solution pour la mise en ligne de n’importe quel TP, du moment qu’il se présente en deux dimensions). De plus, le propre d’une ontologie est de faire partager sa

vision (une « spécialisation d'une conceptualisation » comme le dit Grüber) à autrui, dans le but d'asseoir un cadre commun de dénomination (vocabulaire) et sémantique (relations et propriétés) pour l'interopérabilité. Une ontologie est à notre sens intrinsèquement faite pour être partagée, en accord avec la vision de (Drummond, 2005).

Dans cette approche pragmatique, les propriétés de type « données » complètent notre ontologie. Par exemple, un « ILabel » possède un texte (affecté via la propriété « hasText », un « IPushButton » a une image pour son état normal (« hasImageNormal ») ainsi qu'une image lorsqu'il est enfoncé (« hasImagePressed »). Notre ontologie complète au format OWL est accessible en ligne à l'adresse :

<http://dev.istase.fr/satin/r1ab/eINSTv5.owl>

En pratique, cette ontologie n'est pas dupliquée autant de fois qu'il existe d'appareils dont nous reproduisons la façade. En effet, elle est seulement importée comme base dans chacun des fichier OWL contenant les individus de la façade décrite. Il existe donc une seule ontologie hébergeant les concepts et leurs relations (une telle ontologie est appelée « ontologie mère » (Benmohamed, 2006)). Ensuite, une ontologie par instrument est créée, permettant de décrire les individus (ou instances) des classes décrites dans l'ontologie mère (les éléments graphiques). Ce processus est illustré à la figure II.10.

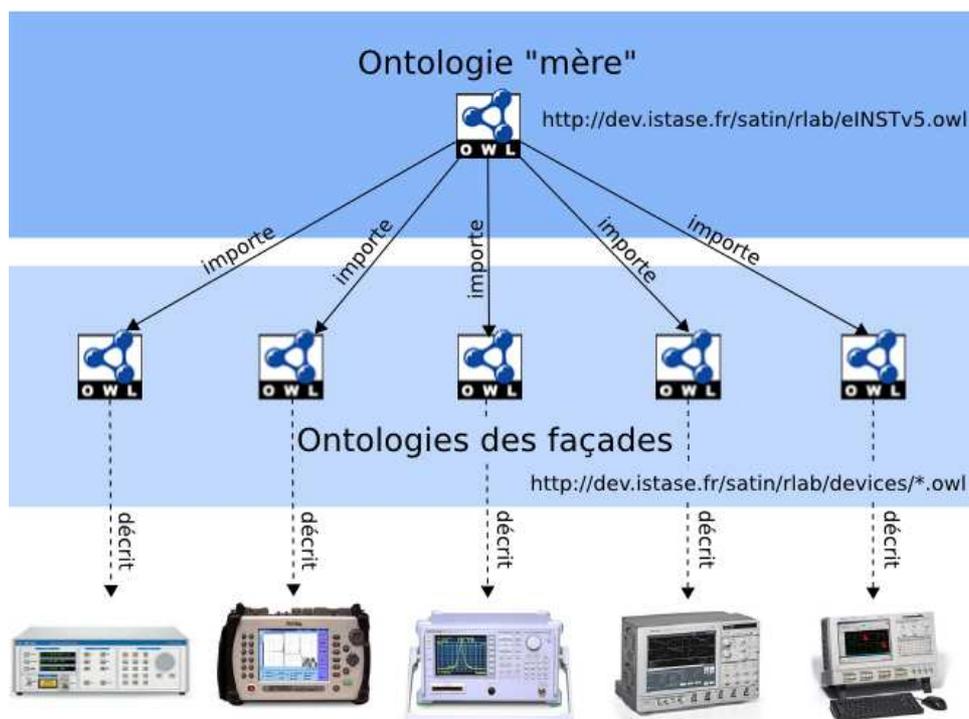


FIG. II.10 – L'ontologie « mère » est importée par chacune des ontologies des façades de dispositif.

Par ailleurs, nous voulons également rendre possible des manipulations impliquant des instruments qui ne présentent pas à première vue une représentation en deux dimensions

mais en trois dimensions. Nous pouvons citer l'exemple du banc d'antennes dans le domaine des télécommunications que nous employons au laboratoire DIOM. Nous avons rendu ce dispositif opérable à distance dans le cadre de nos travaux. Dans ce cas précis, il faut adapter une représentation à trois dimensions en deux dimensions. En réalité, nous ne conservons que les dimensions qui sont capitales à la compréhension de la manipulation. Par exemple, dans le cadre d'instruments en électronique et micro-électronique, nous avons délaissé la profondeur de l'appareil, puisque la façade frontale en deux dimensions est suffisante pour représenter le plus fidèlement possible l'appareil (dans les deux dimensions qu'offre l'écran de l'ordinateur). Il en va de même pour un banc d'antennes : les antennes ne se déplacent que d'un point à un autre sur un rail, tout en étant asservies à un moteur : nous pouvons donc ignorer la dimension suivant laquelle l'antenne ne se déplacera jamais, et obtenir simplement une représentation en deux dimensions. Il faut néanmoins convenir qu'un tel passage de trois à deux dimensions implique des compromis : la vision du concepteur a alors un impact très important sur la représentation du télé-TP.

II.2.2.2 Edition des ontologies

Atelier de façade

La première possibilité pour écrire le fichier OWL d'une façade d'instrument est bien sûr l'édition directe dans un fichier texte. Ce n'est malheureusement pas la façon la plus aisée de procéder. Il devient vite fastidieux et compliqué de produire un fichier OWL (rappelons que ce format s'appuie sur RDF, lequel n'est pas très « lisible » par l'homme directement).

Fort heureusement, il existe plusieurs éditeurs de fichiers OWL sur le marché dont voici les principaux :

- SWOOP¹⁵ de l'Université du Maryland
- Protégé¹⁶ de l'Université de Stanford
- SemanticWorks¹⁷ de Altova (*commercial*)

Nous pensons que Protégé constitue un choix judicieux, du fait de son utilisation répandue qui garantit évolution et documentation. Par ailleurs, Protégé est très ouvert en termes de développement de plugins. On peut donc affirmer que de nombreux outils existent, et il n'est pas rare d'avoir le luxe de pouvoir choisir entre différentes solutions pour réaliser une opération sur l'ontologie en cours d'édition.

15. <http://code.google.com/p/swoop/>

16. <http://www.stanford.us/protege>

17. http://www.altova.com/products/semanticworks/semantic_web_rdf_owl_editor.html

Néanmoins, quels que soient les degrés de maturité et de complétude de ces outils, ils ne sont intrinsèquement pas adaptés au domaine de l'ontologie éditée. L'utilisateur novice à la conception d'ontologie ne sait pas facilement s'adapter à l'outil. Il convient de mettre cela en rapport avec notre objectif de fournir une solution diminuant le plus possible le degré de compétences requis pour amener un TP en ligne. Si l'utilisation des ontologies implique une augmentation des savoir-faire requis, elle ne constitue pas une solution viable au regard des objectifs que nous nous sommes fixés. Néanmoins, les ontologies fournissent un haut degré de sémantique des objets décrit. Cette propriété *de facto* nous permet d'envisager un logiciel permettant d'éditer notre propre ontologie. Un tel logiciel permet de rendre transparent l'usage des ontologies, tout en apportant une édition « WYSIWYG »¹⁸. Nous avons nommé cet utilitaire « Atelier IHM » dont une capture est disponible à la figure II.11.

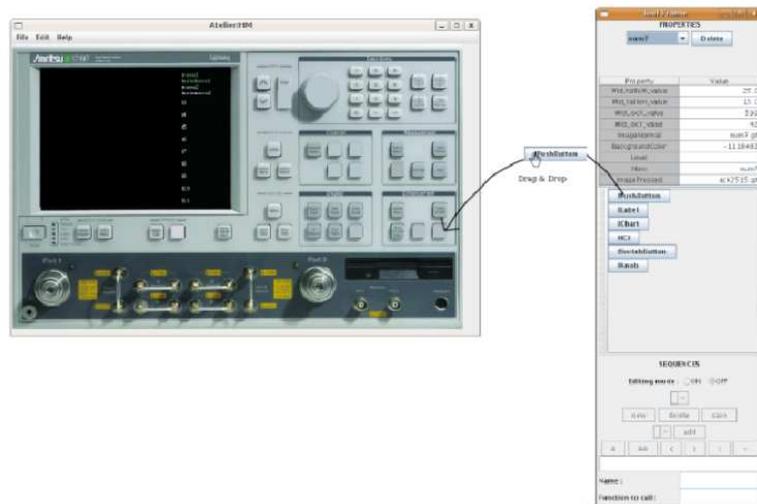


FIG. II.11 – L'atelier d'édition de la façade d'un instrument.

L'éditeur, écrit en Java dans le cadre de notre travail, produit une archive qui sera interprétée par un client ayant en charge l'affichage de la façade à l'exécution (voir II.3.4). Il est important de comprendre que le logiciel créé parcourt une ontologie pour présenter la façade d'un instrument, de manière **indépendante** à l'instrument considéré. En d'autres termes, le client n'a pas connaissance du dispositif dont il instancie une façade déportée. L'archive se trouve être au format JAR (Java ARchive), contenant :

- Les médias (images et sons) utilisés par les widgets (par exemple l'image d'un bouton ou encore le son joué lorsque l'on atteint le pas de rotation d'un bouton rond)

¹⁸. « What You See Is What You Get » (WYSIWYG) est un terme propre au domaine des ateliers de génie logiciel. Il décrit le fait que ce qui est affiché lors de l'édition d'une interface fournit en principe le même résultat à l'exécution.

- l'ontologie elle-même de la façade de l'instrument, au format OWL, important l'ontologie mère disponible en ligne, comme expliqué précédemment à la figure II.10

Il faut ajouter que le logiciel ne connaît pas la composition de l'ontologie. La barre d'outils créée sur la droite est le résultat de l'interprétation des classes filles de la classe widget. Le code n'est donc pas dépendant de l'état de formalisation (forcément mouvant) de l'ontologie. En d'autres termes, cette propriété est très importante puisqu'elle autorise l'enrichissement de l'ontologie par de nouveaux widgets, sans pour autant modifier et recompiler le code de l'atelier IHM. En effet, les widgets sélectionnés pour instanciation dans la barre d'outils sont instanciés via la bibliothèque de réflexion Java, permettant de parcourir les classes chargées par leur nom et de créer des instances à la volée. Au delà de l'instanciation des éléments graphiques, un panneau permet également de rentrer les propriétés desdits éléments aisément, et cela y compris pour le néophyte.

Pré-création de l'ontologie d'une façade d'un instrument

On comprend néanmoins qu'un tel outil montre ses limites lorsque le nombre de widgets à créer est élevé. Nous parlons d'abord des limites temporelles. Rappelons ici qu'un de nos objectifs était de diminuer le temps d'intégration d'un TP dans notre solution de télé-TPs. Concrètement, notre volonté était de descendre en dessous d'une demie-journée pour l'intégration de l'IHM de la façade déportée ainsi que l'intégration à l'intergiciel. Nous sommes aujourd'hui capables de tenir cet objectif, même en traitant des cas non triviaux. Par exemple, lorsque le nombre de boutons sur une façade d'instrument est important, il devient vite fastidieux de déterminer la position de chaque bouton, et de créer les images associées, puisque l'on vise une fidélité la plus précise possible de la façade de l'instrument.

Nous avons alors imaginé un procédé original pour accélérer la qualification des éléments graphiques courants. Ce procédé utilise le traitement d'image, et plus particulièrement la détection de contour. Le principe en est trivial et débute par une prise de vue de la façade de l'instrument. Bien sûr une telle prise de vue ne va pas sans problème d'éclairage et d'aberrations à cause de l'objectif. Il convient alors d'utiliser le matériel et un protocole adéquats. L'image acquise devient l'image de fond de l'interface finale. De plus, un algorithme tout à fait classique de détection de contour nous permet de mettre en évidence les formes susceptibles d'être des contrôles graphiques usuels (les plus courants sont les boutons et les LED¹⁹, ainsi que les rectangles, disques et ellipses ayant une taille relative par rapport à la taille totale de l'image). L'algorithme construit alors l'ontologie correspondant à son interprétation.

Malheureusement, le résultat n'est jamais parfait. En effet, il est souvent difficile,

19. Light-Emitting Diode

en se basant seulement sur la forme, de différencier différents contrôles graphiques (par exemple quelle différence de forme entre un bouton et un bouton à deux états?). De plus, l'algorithme peut éventuellement oublier des formes, ou présenter de faux positifs, c'est-à-dire identifier des zones comme des contrôles graphiques mais qui en fait n'en sont pas. Fort heureusement, il demeure possible de reprendre le fichier produit en l'important dans l'atelier IHM, pour une édition plus poussée et la génération de l'archive JAR. Il est clair que le but de cet algorithme n'est pas d'être parfait, mais de simplifier un travail fastidieux, dans le but de diminuer le temps d'intégration.

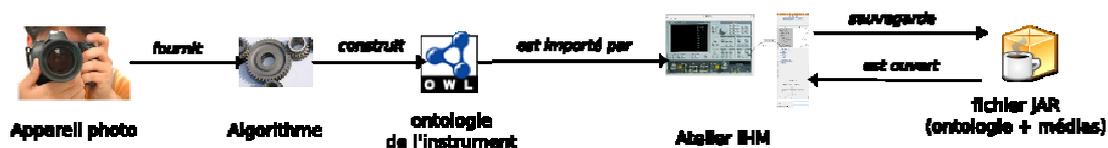


FIG. II.12 – La chaîne d'édition d'une ontologie d'une façade d'instrument.

L'approche est à notre sens innovante pour les ateliers logiciels. En effet, il s'agit d'une application où l'on dispose du résultat graphique à atteindre, alors que bien souvent les IHM sont à imaginer, dans le respect de recommandations ergonomiques. Par cet algorithme, notre chaîne d'édition de l'atelier de génie logiciel s'en trouve modifiée (figure II.12). Cette singularité nous permet d'accélérer la construction de la façade et renforce la métaphore de l'instrument (au sens des IHM), puisqu'elle renforce la fidélité de représentation de la ressource à piloter. L'IHM distante se base en effet sur une photo de l'appareil à piloter.

II.2.2.3 Exemples d'ontologies

Analyseur de réseaux

L'analyseur vectoriel de réseaux (par la suite AVR) est un type d'appareil que nous connaissons très bien au laboratoire DIOM. Récemment encore, des travaux scientifiques menés au DIOM mettent en exergue le fait que les analyseurs vectoriels de réseaux sont indissociables des mesures hyperfréquences. Ces travaux présentent ainsi les analyseurs de réseaux :

« C'est en 1967 que la société américaine *Hewlett Packard* développa le premier analyseur de réseau, le HP8410. Celui-ci permettait d'extraire du dispositif sous test les paramètres de dispersion, grandeurs aujourd'hui indissociables de la mesure hyperfréquences (Kahrs, 2003). Ces appareils très utilisés en laboratoire fonctionnent grâce à des réflectomètres à deux coupleurs. Ces derniers acheminent un signal électromagnétique vers le dispositif à analyser et récupèrent le signal réfléchi par celui-ci. Deux coupleurs sont utilisés pour

copier dans deux canaux parallèles le signal émis, qui fait office de signal de référence, et le signal réfléchi. Pour caractériser complètement le DST²⁰, l'appareil doit pouvoir remonter au rapport des amplitudes ainsi qu'à la différence des phases. Or ces opérations ne sont pas simples pour des signaux de fréquence élevée et variable. L'analyseur procède donc à une série d'échantillonnages pour convertir les données en signaux de fréquences intermédiaires fixes et plus basses. Le point important est de garder le même rapport d'amplitudes et la même différence de phases lors de la conversion.

Les analyseurs vectoriels de réseau sont aujourd'hui disponibles auprès des principaux leaders mondiaux du test et de la mesure, qu'ils soient japonais comme *Anritsu*, américains comme *Agilent Technologies* ou bien allemands comme *Rohde & Schwarz...* ». (Rouiller, 2006)



FIG. II.13 – Analyseur vectoriel de réseaux Anritsu 37397A utilisé au laboratoire DIOM.

Au laboratoire DIOM, nous utilisons principalement un analyseur *Anritsu*, modèle 37397A (figure II.13). Son utilisation conduit à la détermination expérimentale des paramètres de dispersions²¹ du DST (Pozar, 2005). Une photographie de notre testeur sous pointe est présente à la figure II.14. Cet AVR présente une plage de travail de 40 MHz jusqu'à 65 GHz, permettant ainsi d'effectuer une caractérisation par mesure sous pointe. Le banc complet de mesure comprend donc l'AVR et le DST, comme illustré à la figure II.15.

Nous avons réalisé l'ontologie de cet appareil en regard de notre ontologie mère présentée précédemment. Le fichier OWL ainsi produit est interprété par une application Java (dont

20. Dispositif Sous Test

21. ou « paramètres S », souvent notés S_{ij}

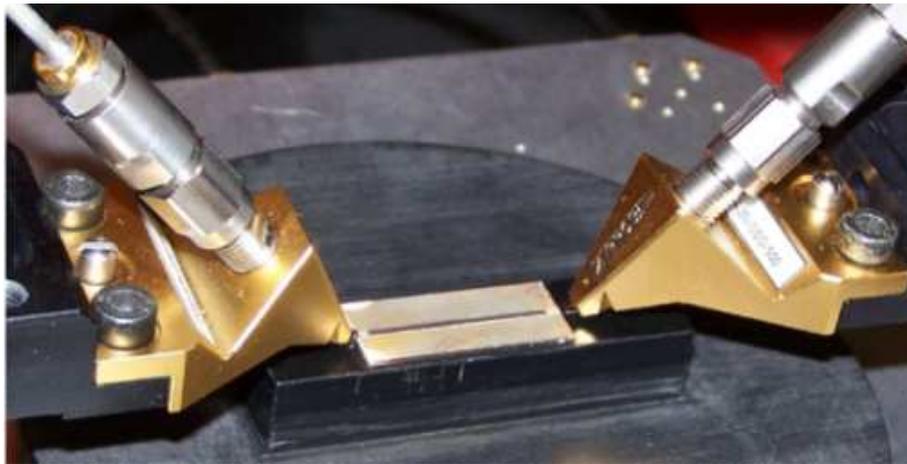


FIG. II.14 – *Testeur sous pointe permettant la caractérisation d'un composant au laboratoire DIOM.*

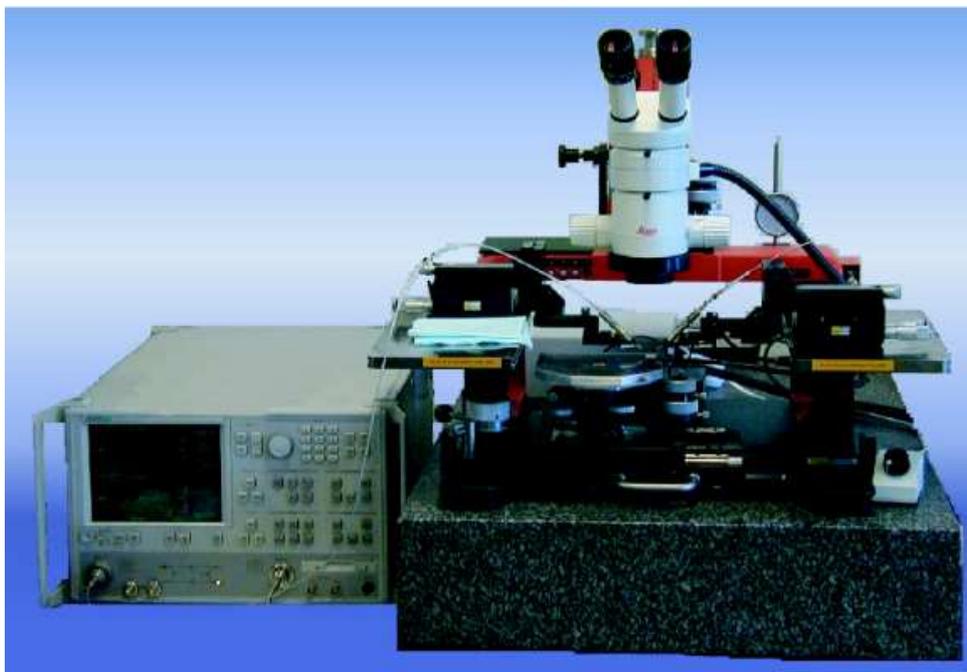


FIG. II.15 – *Banc de mesure complet (AVR et DST) du laboratoire.*

l'interconnexion à l'intergiciel est présentée au chapitre suivant).

Le résultat, s'interfaçant via un écran tactile pour plus de réalisme, a été photographié et reproduit à la figure II.16.

« *Web Switch* » et autres exemples

Un autre appareil a été rendu pilotable par la description de son ontologie, laquelle est exploitée par notre applicatif. Il s'agit d'un multiplexeur de voies, nommé « *Web Switch* » au sein du laboratoire. Il s'agit d'un dispositif « fait maison ». Ce dispositif



FIG. II.16 – Application résultante, permettant d'utiliser l'AVR Anritsu à distance, de manière réaliste : fidélité de la représentation et utilisation (écran tactile).

permet, par l'intermédiaire d'une carte d'interfaçage, de choisir parmi les composants embarqués. Il s'agit en fait d'un sélecteur d'entrées/sorties. À l'origine, comme les télé-TPs requièrent plusieurs dispositifs, ce sélecteur a été créé pour basculer d'un composant à l'autre, sans avoir à détacher l'ancien composant et rattacher physiquement le composant à caractériser. Le « Web Switch » permet justement de pouvoir choisir, parmi les composants embarqués sur une carte raccordée, celui qui sera sollicité. Ce schéma de principe est illustré à la figure II.17.

Une photographie du « WebSwitch » est également disponible à la figure II.18.

De la même façon que pour l'AVR Anritsu 37397A, nous avons établi l'ontologie de cet instrument au format OWL. Notre client charge alors cette ontologie et instancie dynamiquement les « IWidgets » correspondants.

Bien entendu, nous travaillons sur d'autres équipements pour renforcer notre approche. Ainsi, nous travaillons sur un télé-TP de chimie (mélange de fluides), et un télé-TP mettant en œuvre un banc d'antennes (déplacement des antennes sur un rail). Ce sont à nos yeux des télé-TPs très stimulants, dans le sens où l'hétérogénéité de leur domaine nous permet de montrer une force de notre modèle : l'indépendance aux instruments pilotés. Le télé-TP impliquant le banc d'antenne est particulièrement avancé puisqu'il est fonctionnel, cependant quelques tests restent à effectuer avant d'être déployé en production.

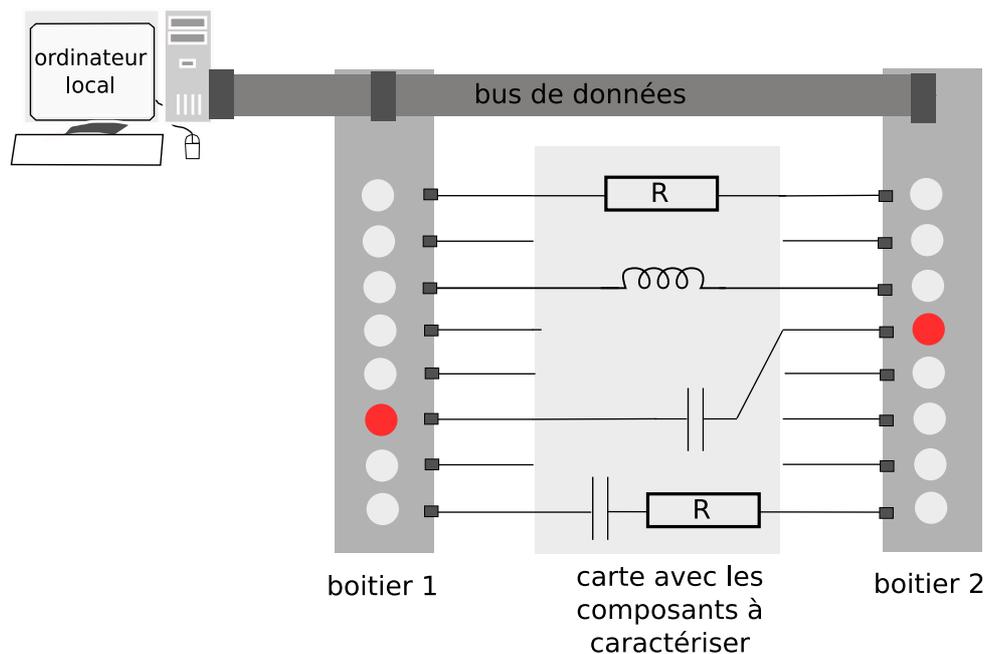


FIG. II.17 – Le principe du « WebSwitch » : un sélecteur d'entrées/sorties pour éviter de multiplier les connexions et déconnexions durant une session de télé-TP.

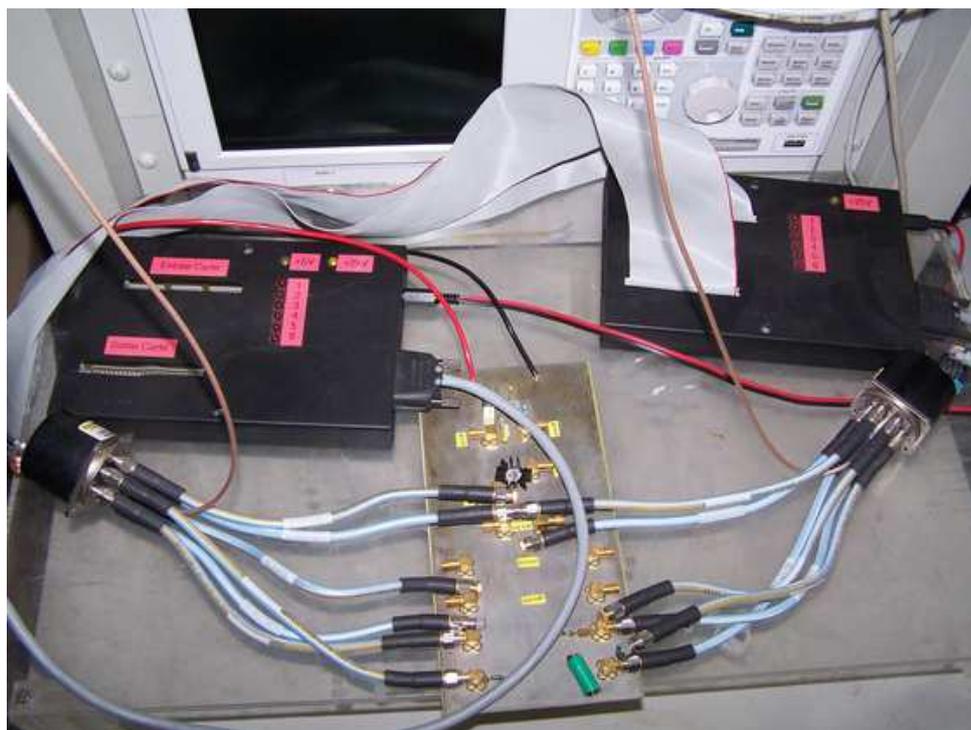


FIG. II.18 – Le « WebSwitch » : photographie d'un dispositif « fait maison ». Schéma de principe à la figure II.17.

À notre connaissance, il s'agit de la seule plate-forme de télé-TPs qui embarque une couche d'abstraction suffisante pour que l'appliquatif logiciel soit commun à *tous* les télé-TPs mis en œuvre. En effet, quel que soit le type d'appareil, du moment que l'on peut établir une façade déportée en utilisant notre bibliothèque de widgets, il sera possible

de l'intégrer à notre système, sans écrire une seule ligne de programme en dehors de l'ontologie associée à la façade.

II.2.3 Limitations actuelles

Notre approche pragmatique nous pousse à livrer une ontologie concise et fonctionnelle. L'objectif étant de faciliter l'extension de l'ontologie de façade des instruments. Il est vrai que la liste des widgets disponibles est loin d'être exhaustive. Celle-ci est complétée au fur et à mesure que le besoin de nouveaux widgets se fait sentir durant la phase de déploiement de nouveaux télé-TPs.

Un autre problème vient s'ajouter à celui exposé précédemment. Lorsque l'on examine l'instrument fictif de la figure II.4, on s'aperçoit qu'il existe une bijection entre action sur un élément de l'interface et déclenchement d'une fonctionnalité. Cette propriété est très commode, puisque chaque évènement est ainsi envoyé à l'instrument distant via l'intergiciel. Cette propriété est également vérifiée pour les deux dispositifs présentés précédemment : l'AVR Anristu et notre « WebSwitch ».

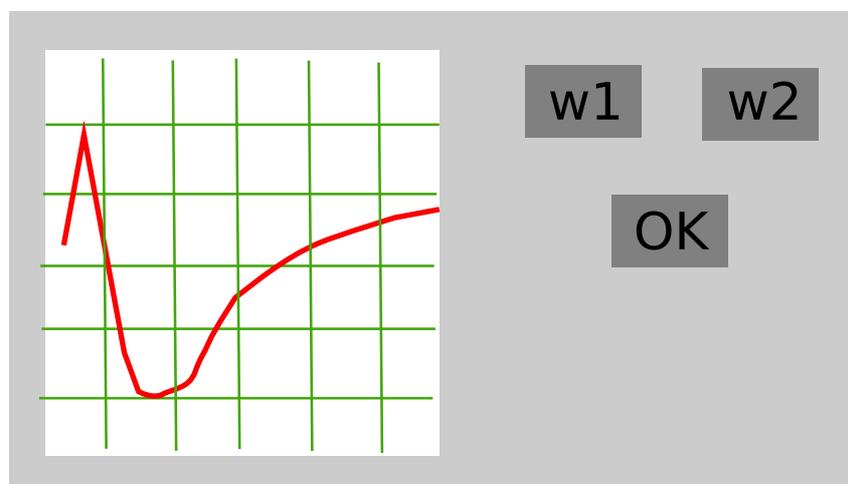


FIG. II.19 – *Instrument fictif modifié: ajout du bouton OK: fin de la bijection widget actionné / fonctionnalité identifiée.*

Modifions légèrement le dispositif fictif de la figure II.4 pour le transformer comme présenté à la figure II.19. La seule différence se situe dans l'ajout d'un bouton « OK ». Le comportement précédent était :

- Affichage des données sous la forme d'une courbe sur action du widget « w1 »,
- Affichage des données sous la forme d'une abaque de Smith sur action du widget « w2 ».

Et il devient :

- Affichage des données sous la forme d'une courbe sur action du widget « w1 » **puis** du bouton « OK »,
- Affichage des données sous la forme d'une abaque de Smith sur action du widget « w2 » **puis** du bouton « OK ».

En d'autres termes, le bouton « OK » vient valider la sélection du mode d'affichage. Le fait d'appuyer sur « w1 » ou « w2 » ne suffit donc plus à identifier la fonctionnalité déclenchée. La propriété de bijection widgets/fonctionnalités n'est plus vérifiée : dans l'illustration proposée, un widget ne correspond plus à une et une seule fonctionnalité pour un widget donné. Par contre, à une séquence d'action correspond une fonctionnalité.

On peut dire que les exemples de l'instrument fictif, de l'AVR Anritsu et du « WebSwitch » sont par chance des situations où toutes les séquences correspondant à une fonctionnalité sont atomiques : chaque séquence n'est composée que de l'actionnement d'un seul widget.

Bien sûr, rien n'empêche d'éluder le problème d'un point de vue fonctionnel. À chaque action, on envoie l'évènement correspondant à l'instrument distant par l'intergiciel. A lui de traiter s'il s'agit d'une fonctionnalité atteinte ou de mémoriser celle-ci pour pouvoir identifier une séquence donnée parmi toutes celles possibles. Un tel raisonnement est logique puisque l'on repousse le développement logiciel propre à l'interaction avec l'instrument sur le PC local relié physiquement au dispositif.

D'un autre côté, ce code devient spécifique à chaque fois qu'il convient de maintenir des séquences d'actions. Par ailleurs, comment anticiper des éléments de réponse aux challenges futurs des télé-TPs sans savoir identifier les séquences d'utilisation associées aux actions menées par l'utilisateur ? Par exemple, comment déterminer le degré d'expertise d'un utilisateur, à partir de son cheminement dans un scénario pédagogique, sachant qu'un scénario est basé sur les fonctionnalités atteintes, et non pas sur une stricte séquence d'actions. Plusieurs séquences d'utilisation différentes peuvent mener au même résultat : il convient de modéliser tous les chemins possibles permettant de déclencher la fonctionnalité correspondante. Identifier les fonctionnalités peut également faciliter la conservation de l'intégrité du dispositif (existe-t-il des exclusions mutuelles entre fonctionnalités qui conduisent soit à une dangerosité accrue soit à l'endommagement du matériel si elles ne sont pas respectées?).

Bien sûr, établir la correspondance entre actions et fonctionnalités demande la coo-

pération entre concepteurs du télé-TP et développeurs de la solution logicielle (Borchers, 2001). La difficulté majeure étant de pouvoir modéliser toutes les possibilités dans leur ordre chronologique (Gravier & Fayolle, 2006a). Par exemple, si λ_1 est une **séquence d'utilisation** composée des widgets w_1 , w_2 , w_3 et w_4 , λ_1 , pouvant s'écrire comme suit :

$$\lambda_1 = w_1 \& (w_2 | w_3) \& \& w_4 \quad (\text{II.2})$$

Il faut alors lire : « la séquence λ_1 est réalisée si et seulement si le widget w_1 est actionné, suivi ou précédé soit du widget w_2 , soit de w_3 , et cette action est suivie de l'utilisation du widget w_4 ».

Les opérateurs pour les séquences d'utilisation sont ceux de base de l'algèbre de Boole adaptés à notre domaine :

- $\&$: coordination sans ordre (la séquence « A & B » est satisfaite si et seulement si A est actionné puis B, mais également si B est actionné juste avant A, sans autre intermédiaire).
- $\&\&$: coordination ordonnée (la séquence « A &\& B » est satisfaite si et seulement si A puis B sont utilisés, et donc à l'exclusion de l'expression commutée « B suivi de A », qui s'exprime B &\& A).
- $|$: ou (inclusif) (la séquence « A | B » est satisfaite si et seulement si A est sollicité ou si B est sollicité, ou si A et B sont tous les deux sollicités).
- \oplus : ou exclusif (la séquence « A \oplus B » est satisfaite si et seulement si A est utilisé en l'absence de B ou si B est utilisé en l'absence de A).

Dans ces conditions, on retrouve les réflexes classiques de simplification des expressions booléennes : la séquence λ_1 est équivalente à la séquence λ_2 laquelle est équivalente à λ_3 , avec λ_2 :

$$\lambda_2 = (w_2 | w_3) \& w_1 \& \& w_4 \quad (\text{II.3})$$

et λ_3 :

$$\lambda_3 = (w_3 | w_2) \& w_1 \& \& w_4 \quad (\text{II.4})$$

Bien entendu, il reste possible d'introduire des opérateurs plus complexes, comme par exemple « l'inversion d'ordre chronologique »²², pour enrichir la sémantique permettant d'exprimer une séquence d'utilisation par rapport à une autre. Par exemple, en regard

22. On notera ici $\overleftarrow{\lambda_1}$ la séquence impliquant les mêmes widgets que la séquence λ_1 : il s'agit de la séquence λ_1 « jouée » à l'envers.

des déclarations précédentes :

$$\overleftarrow{\lambda}_1 = w_4 \& \& w_1 \& (w_2 | w_3) \quad (\text{II.5})$$

$$= w_4 \& \& (w_2 | w_3) \& w_1 \quad (\text{II.6})$$

Une autre possibilité d'enrichissement pourrait être la négation d'utilisation d'un widget. En effet, il est intéressant de pouvoir nommer « n'importe quel widget sauf », au lieu de donner la liste exhaustive des widgets possibles.

D'autres possibilités résident dans les relations que les séquences entretiennent entre elles. Par exemple, il peut être avantageux d'être en mesure d'exprimer le fait que la séquence λ_4 est incluse dans λ_1 , et par conséquent :

$$\lambda_4 = w_1 \& (w_2 | w_3) \Rightarrow \lambda_1 = \lambda_4 \& \& w_4 \quad (\text{II.7})$$

La notion de sous ensemble appliquée au domaine des séquences d'utilisation favorise la différence entre *séquences* et *actions*. Elle permet de montrer qu'une séquence atomique (insécable), donc une action, peut être un sous-ensemble de plusieurs séquences d'utilisation.

Il reste néanmoins la nécessité de transcrire cette approche dans l'applicatif. Comment modéliser une séquence d'utilisation, de la manière la plus simple possible pour la personne élaborant un télé-TP ?

Nous trouvons une nouvelle fois dans les ontologies une réponse à ce dilemme. Nous avons spécifié les concepts et leurs relations pour une façade d'instrument, de manière à faciliter et généraliser leur construction, sans avoir à compiler une ligne de code. Il est possible d'utiliser cette base de connaissances pour étendre l'ontologie en y intégrant la notion de séquence.

L'approche la plus pertinente est d'utiliser un langage permettant d'établir des règles logiques à partir d'une ontologie. Comme nous le verrons plus tard (III.3), il existe de tels langages dans le monde du web sémantique. Néanmoins, l'étude de ces langages nous montrera que leur conception souffre de plusieurs limitations, particulièrement en ce qui concerne les opérateurs logiques (appelés « built-in » dans ces langages). L'opérateur $\&$ est en réalité le seul à disposition, ce qui ne nous permet pas aujourd'hui d'implémenter notre formalisme de séquence d'utilisation d'un instrument. C'est la raison pour laquelle les séquences ne sont pas encore implémentées dans la solution actuelle, et cela constitue une limitation qui demande une réingénierie des langages à base de règles (extension de

leur expressivité) pour être levée.

II.2.4 Bilan

Nous avons montré dans cette partie qu'il était possible, avec un peu de recul et une approche pragmatique, de factoriser un grand nombre de façades d'instruments pour plusieurs domaines scientifiques. Ces façades sont représentées en utilisant une ontologie, hébergeant les concepts clés, leurs relations et leurs propriétés. L'objectif sous-jacent reste de diminuer le degré de compétences et le temps requis pour créer un nouveau télé-TP. Cette approche permet d'éviter de faire appel à des compétences en développement logiciel, puisqu'un atelier de génie logiciel propre au domaine permet de construire l'ontologie pour l'instrument concerné. Par la suite, un client générique (c'est-à-dire indépendant aux instruments) instancie la façade à partir de la lecture de l'ontologie correspondant à cette façade. Ce client est le même pour tous les télé-TPs, et est en charge d'assurer la connectivité à l'intergiciel, lui aussi partagé entre tous les télé-TPs mis en ligne. **En résumé, l'ontologie produite est un fichier de déploiement d'un nouvel instrument dans la plate-forme de télé-TPs, sans qu'aucun autre ajout ne soit nécessaire.** L'avantage visé étant bien sûr d'éviter au concepteur d'un télé-TP de se soucier de l'intergiciel. La suite de ce mémoire présente donc la plate-forme et l'intergiciel développés au cours de notre thèse.

II.3 Implémentation de la plate-forme et de l'intergiciel

Nous nous sommes focalisés jusqu'ici sur la formalisation des éléments entrant en compte dans une plate-forme de télé-TPs. Cette partie débattre de l'implémentation d'un prototype réalisé dans le cadre de notre travail, au regard des spécifications établies dans la section précédente.

II.3.1 Généraliser le lien entre IHM et l'intergiciel

Faire correspondre les classes de l'ontologie aux classes de l'implémentation.

Les widgets participant à la façade d'un télé-TP sont ceux présentés par l'ontologie mère à la figure II.8. Tous les widgets présents dans l'ontologie correspondent à une classe écrite en langage Java. Ces derniers sont rassemblés dans une seule bibliothèque au format JAR. Il est important d'appréhender l'existence d'une correspondance entre les propriétés et relations de chacune des classes et leur équivalent sous la forme d'une méthode publique dans la classe Java. Illustrons cette idée par un exemple : le widget « ILabel » possède la propriété « hasText » d'après l'ontologie délivrée à la section précédente. Cette propriété dans la classe « ILabel » de l'ontologie trouve son équivalent nommé « setText » dans la classe « ILabel » de la bibliothèque de widget. Cette correspondance est importante puisque c'est elle qui nous permettra plus tard de modifier une propriété de widget par un appel non explicite. Cet appel passe par la recherche d'une méthode publique ayant pour signature la concaténation des chaînes « set » et du « nom de la propriété » et les arguments adéquats, grâce à la bibliothèque de réflexion Java (mécanisme d'introspection).

L'utilisation de la bibliothèque Java procurant l'introspection est primordiale pour le caractère générique de notre approche. En effet, nous recherchons des méthodes publiques à partir de leur signature dans la structure du programme pour les instancier. Cette façon de procéder permet de lancer n'importe quelle méthode à partir de son nom, et donc de s'affranchir d'un long bloc d'instructions conditionnelles « if ... else if ... else », qui présente également l'inconvénient de devoir être modifié, lorsqu'il faut rajouter une situation possible à notre bloc conditionnel.

Pour bien comprendre, positionnons-nous du côté du PC pilotant l'instrument (à l'entrée des commandes et l'extraction des résultats). Si le résultat d'une commande implique de changer le texte du « ILabel » nommé par l'identifiant « label_w1 », le message envoyé au client distant comporte forcément la modification à apporter à l'interface gra-

phique pour refléter le changement. C'est précisément à la réception de ce message que le risque de perte de genericité existe. Le premier réflexe du développeur consiste à proposer des embranchements conditionnels suivant la modification à apporter à l'interface. Il faudrait donc autant d'embranchements conditionnels que de widgets à traiter, multiplié par le nombre de propriétés potentiellement modifiées, widget par widget.

On comprend bien que cette façon de procéder n'est ni élégante, ni simple à maintenir. D'autant plus que multiplier le nombre de lignes de code augmente le risque de bogues. En effet, si l'on rajoute une propriété (et sa méthode publique) au widget, il faut modifier ce bloc d'instructions conditionnelles autant de fois que la propriété est susceptible d'apparaître. Évidemment, il faut par ailleurs créer autant de blocs qu'il existe d'instances de widget, et en rajouter un à chaque fois que l'on ajoute un widget à l'interface. On est alors loin d'une approche maintenable et réutilisable facilement.

La bibliothèque d'introspection Java permet de complètement gommer le problème, puisque l'on peut rechercher des pointeurs sur les classes, méthodes, propriétés, et instances, et ainsi modifier soit la structure du programme, soit les instances sans connaître *a priori* la signature de la méthode à invoquer. L'algorithme se réduit alors à l'expression suivante :

procédure recevoir (**E** widget_id : **Chaîne de caractères** ; **E** widget_property : **Chaîne de caractères** ; **E** property_value : **Chaîne de caractères**)

Déclaration widget : IWidget; widgetClass, propertyClass : Classe; propertyMethod : Methode; methodeName : **Chaîne de caractères**

début

Remarque : Récupération de l'instance du widget concerné

widget ← self.getWidgetById(widget_id);

Remarque : Identification de la classe du widget

widgetClass ← widget.getClasse();

Remarque : Identification de la classe de la propriété

propertyClass ← property_value.getClasse();

Remarque : Récupération d'un pointeur sur la méthode portant la signature :

methodeName ← "set"+widget_property

propertyMethod ← widgetClass.getMethode(methodeName,propertyClass);

Remarque : Exécution de la méthode trouvée, sur l'instance du widget considéré, avec pour argument celui passé en paramètre d'entrée

```
propertyMethod.invoke(widget, property_value);
```

fin

Ces lignes de pseudo-code remplacent avantageusement le code précédemment proposé puisqu'elles sont concises et n'ont pas à être modifiées si une propriété est ajoutée au widget dans l'ontologie et à la classe Java associée, ou si une nouvelle instance d'un widget est ajoutée à la façade.

Connectivité à l'intergiciel

Généraliser la connectivité à un intergiciel, pour éviter le développement spécifique, passe par la modification du comportement natif des widgets. D'ordinaire, lorsque des widgets sont utilisés, l'évènement est envoyé au niveau de la couche applicative supérieure. Ensuite, celle-ci sollicite les écouteurs d'évènements qui sont abonnés pour être notifiés en cas d'occurrence de ce type d'évènement (figure II.20).

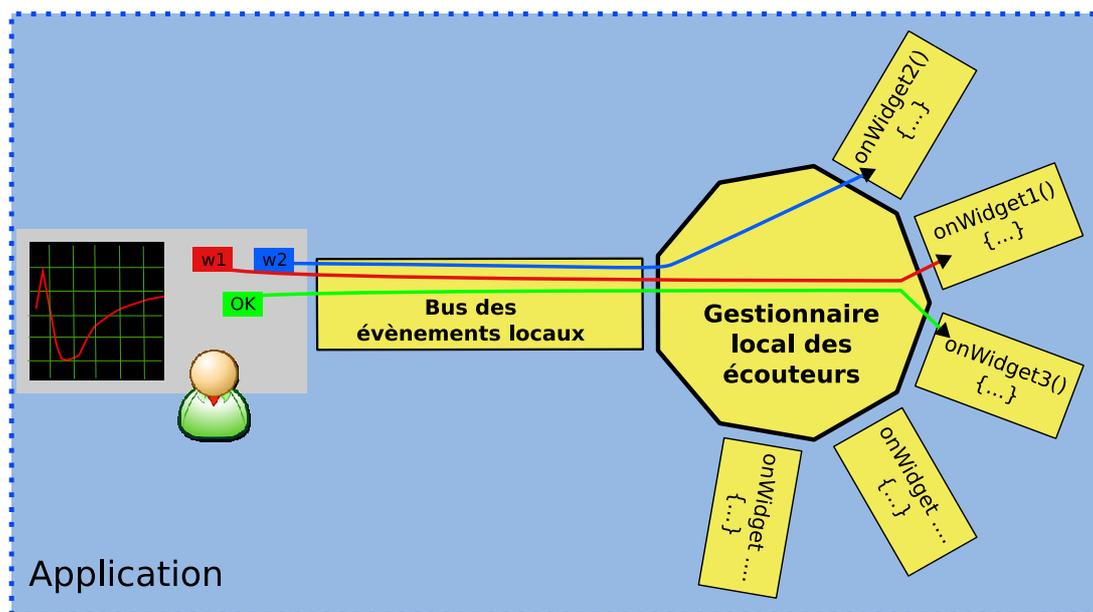


FIG. II.20 – Aiguillage classique des évènements dans une application.

Ce mécanisme d'aiguillage n'est plus valable dans le cas d'une application distribuée. En effet, les écouteurs ne sont plus internes à l'application mais embarqués dans des instances d'une application distante. Il faut alors surcharger chacun des écouteurs pour sérialiser le message sur l'intergiciel considéré, et ce pour chaque écouteur de chaque widget.

Pour contourner le problème, nous avons modifié ce comportement natif des widgets. Au lieu de notifier une couche applicative supérieure, l'évènement est **inhibé** localement et

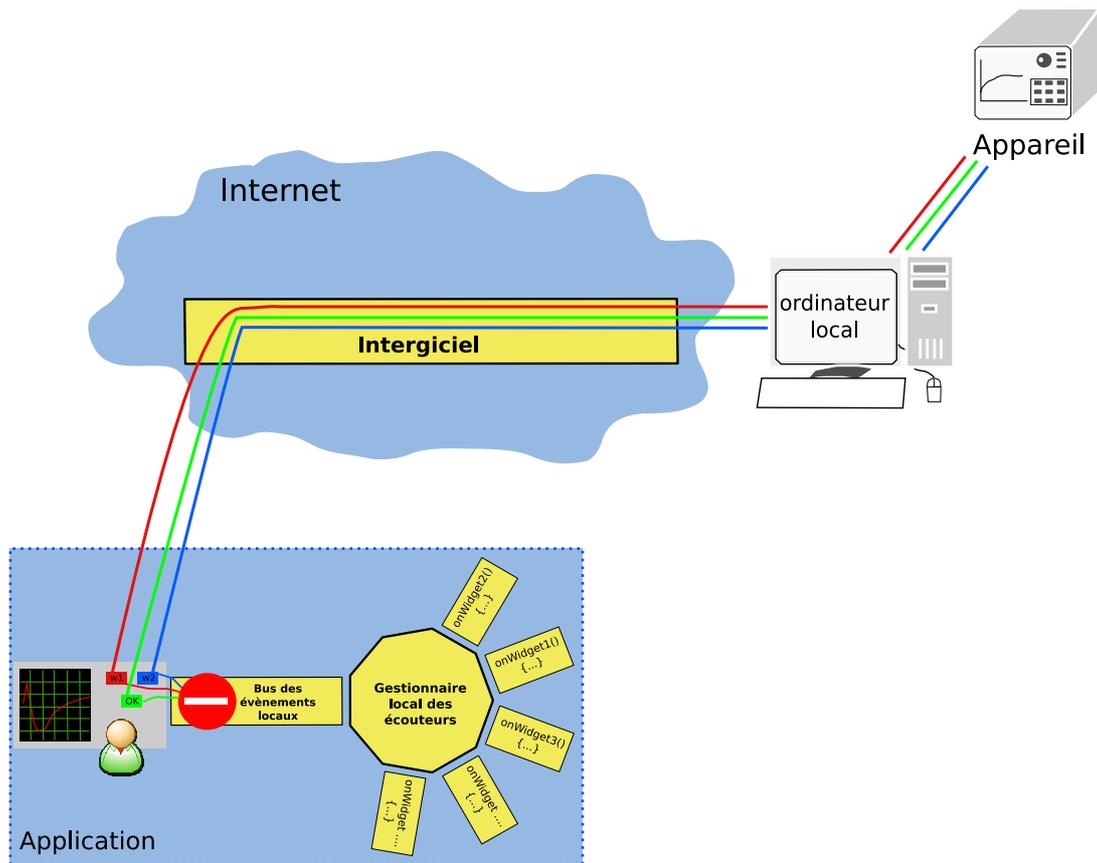


FIG. II.21 – Aiguillage dans la plate-forme de télé-TPs : les widgets sérialisent les évènements eux même sur l’intergiciel.

acheminé sur l’intergiciel, directement par le widget lui-même (figure II.21). Cela présente l’avantage qu’aucune ligne de code n’est nécessaire pour sérialiser et désérialiser des messages entre les applications. Autrement dit, les widgets de notre bibliothèque envoient directement les évènements qu’ils génèrent au PC du laboratoire, sans intervention d’un développeur ! Construire l’ontologie d’une façade apporte donc, non seulement une interface graphique sans coder une ligne, mais fournit également la connectivité à l’intergiciel (et donc au PC distant), sans programmer. L’adresse de l’intergiciel à utiliser est simplement passée en argument du programme et les widgets savent alors sur quel canal de discussion sérialiser les messages. Cela implique qu’aucun développement n’est nécessaire pour envoyer les messages jusqu’à l’instrument. En d’autres termes, nous avons respecté la généralisation des maillons de la chaîne logicielle des télé-TPs que nous présentions à la figure II.1.

II.3.2 Protocole applicatif pour l’architecture distribuée

Bien entendu, sérialiser un message sur un intergiciel implique un transport de bout en bout et une désérialisation du message. En fait, un client Java, indépendant des instru-

ments²³, construit la façade logicielle de l'instrument à partir de l'ontologie.

Lorsqu'un utilisateur actionne un widget, un message est sérialisé sur l'intergiciel (de l'utilisateur vers l'instrument ou de l'instrument vers *tous* les utilisateurs²⁴) et un processus en trois phases est déclenché pour acheminer le message à destination (figure II.22).

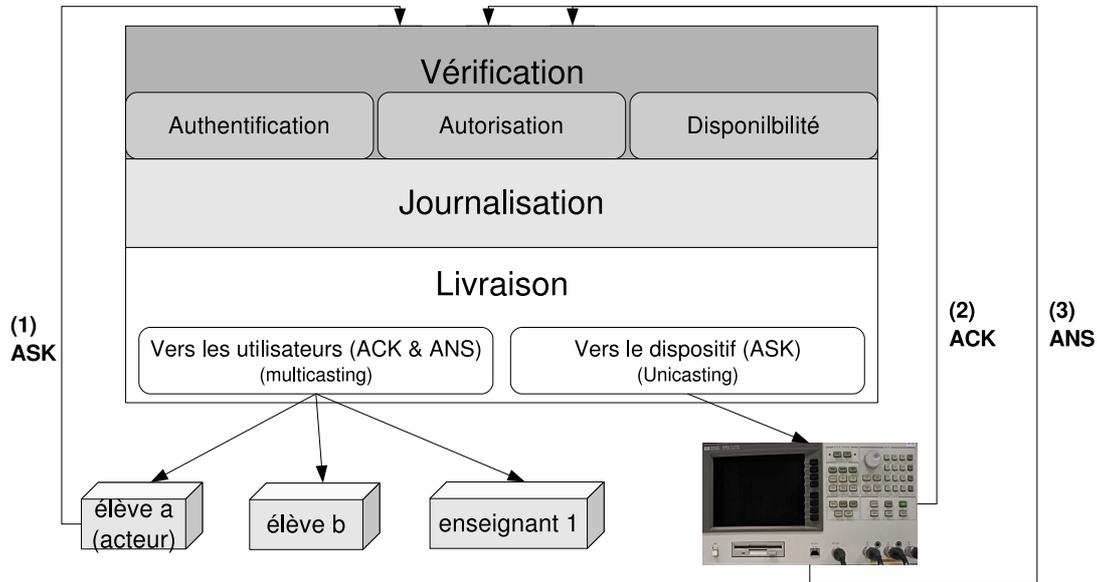


FIG. II.22 – Cycle de vie des messages sur l'intergiciel.

Une première couche applicative est responsable de l'authentification et l'autorisation d'accès de l'auteur du message véhiculé. L'authentification est effectuée avec un annuaire LDAP²⁵. Une fois l'identité vérifiée, cette même couche logicielle associe un rôle à l'utilisateur, en regard de ses appartenances à des groupes d'individus dans l'annuaire. Cette association permet de donner ou non l'accès à la publication de messages (être authentifié ne suffit pas, il faut appartenir au groupe des utilisateurs des télé-TPs). Ce type d'association est un modèle bien connu en sécurité, puisqu'il s'agit de RBAC²⁶. En effet, nous n'associons pas des droits pour des opérations à des individus ou des groupes, mais des rôles endossés dans l'application. Ensuite, ce sont les appartenances aux groupes qui déterminent le rôle joué par l'utilisateur dans l'application. Cette couche logicielle écrite en Java se base sur la bibliothèque JAAS²⁷.

23. Ce qualificatif est important : le client ne connaît pas l'instrument qu'il pilote, il existe une totale décorrélation entre l'applicatif et l'instrument distant, c'est justement ce qui permet d'assurer une généralisation de cette solution logicielle à plusieurs télé-TPs.

24. Il convient ici de se rappeler de notre détermination à proposer une approche collaborative. En ce sens, tous les messages de l'instrument sont envoyés à tous les utilisateurs, de façon à ce que chacun puisse être averti des actions menées durant la session de télé-TP

25. Lightweight Directory Access Protocol

26. Role Based Access Control

27. Java Authentication and Authorization Service, <http://java.sun.com/products/jaas/>

La seconde couche logicielle de l'intergiciel couvre la journalisation des messages. Tous les messages (autorisés ou rejetés) sont journalisés dans une base de données externe. Nous utilisons le SGBD²⁸ libre et ouvert PostgreSQL²⁹.

Enfin, une dernière couche logicielle est responsable de la livraison des messages applicatifs aux instances logicielles du télé-TP de l'apprenant, des enseignants et de l'appareil distant. Il est important de souligner que le dispositif distant ne peut être atteint que par ces couches logicielles, les accès directs ne sont pas possibles. En effet, seul l'intergiciel, au niveau réseau, c'est-à-dire TCP/IP, est autorisé à accéder au PC local pilotant l'appareil. Cela est rendu possible par un jeu de pare-feu (le PC local du laboratoire se trouvant au delà de la zone démilitarisée). De tels filtres basés sur l'authentification et l'autorisation, associés à un contrôle niveau réseau, permettent de réduire les risques d'une utilisation frauduleuse de l'appareil.

Nous classifions les messages reliés sur l'intergiciel en trois catégories. Nous associons à chacun une clé, qui agit comme représentation du performatif associé. Les différentes clés des trois catégories sont les suivantes :

- ASK, pour « ASKing ». Cette clé de message est utilisée dans notre protocole de communication lorsqu'un widget a été utilisé. Il s'agit de relayer la commande à l'instrument pour exécution.
- ACK, pour « ACKnowledgment ». Une commande envoyée à l'instrument distant est susceptible d'occuper une fenêtre temporelle importante (par exemple un déplacement d'antenne ou une acquisition de nombreux points). A chaque réception d'un message de type ASK, l'application renvoie, par réflexe programmé, un message ACK pour acquitter de la bonne réception du message. Ce type de message a en réalité, un rôle plus évolué : il est envoyé à **tous** les participants du télé-TP. Il véhicule également une information concernant l'identité de l'auteur du message ACK à l'origine. De ce fait, le retour effectué auprès des participants permet non seulement de prendre connaissance qu'une action a été menée, mais également de savoir qui l'a réalisée. C'est un aspect très important puisqu'il supporte la conscience de groupe. Comme une unique couleur est attribuée à chaque participant à sa connexion, l'acquiescement graphique est peint de la couleur de l'auteur de la commande (l'auteur du message ASK d'origine). En pratique, cela se traduit par le recouvrement total du widget sollicité par la couleur de l'auteur de l'action. Cet acquiescement coloré perdure tant que la commande envoyée n'a pas retourné de résultats. De ce fait, pendant toute la durée de l'acquiescement il n'est plus possible d'envoyer des messages

28. Système de Gestion de Base de Données

29. <http://www.postgresql.org/>

à l'instrument.

- ANS, pour « ANSwering ». Il s'agit du message de l'instrument vers les participants correspondant à la réponse au stimulus provoqué par une commande reçue au préalable (donc par un message de type « ASK »).

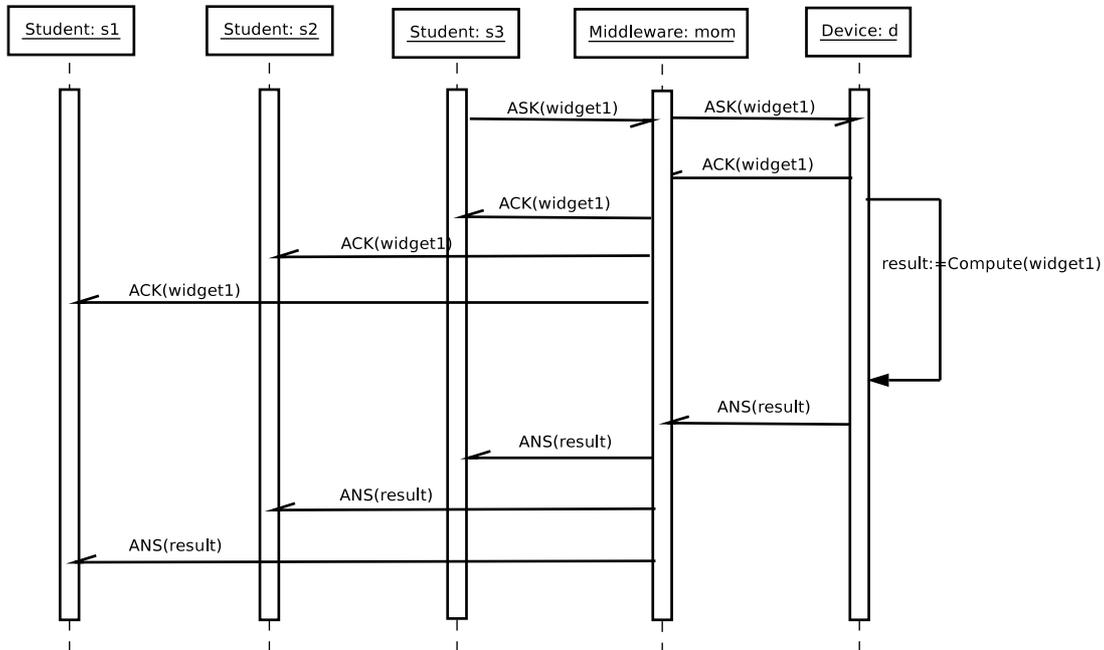


FIG. II.23 – Diagramme de séquence au format UML d'une phase « commande-acquittement-réponse ».

À l'inverse des messages utilisant le performatif ASK (lequels sont seulement envoyés d'un utilisateur vers l'instrument par le truchement de l'intergiciel), les messages utilisant les performatifs ACK et ANS sont envoyés à tous les participants à la manipulation. Nous l'avons dit, cela permet de fournir la conscience de groupe à cette application collaborative distribuée. L'exemple d'une phase « commande-acquittement-réponse » est fourni à la figure II.23, via un diagramme de séquence UML³⁰. Dans cette figure, tous les appels sont asynchrones (demi-pointe de flèche sur le message). Le seul appel synchrone se trouve être le message récursif de l'instrument exécutant la commande reçue (nommé « Compute » sur notre figure).

II.3.3 Choix de l'intergiciel

Un intergiciel possible dans le cadre de l'instrumentation distance est le très répandu RPC³¹. RPC est un moyen de déclencher à distance une opération sur un serveur d'applications. Il permet alors de manipuler un objet (au sens de la programmation orientée

30. Unified Modeling Language

31. Remote Procedure Call

objet) et en ce sens permettrait à la plate-forme de télé-TPs d'exposer des objets « instrument » pour la manipulation distante. Son fonctionnement est illustré à la figure II.24.

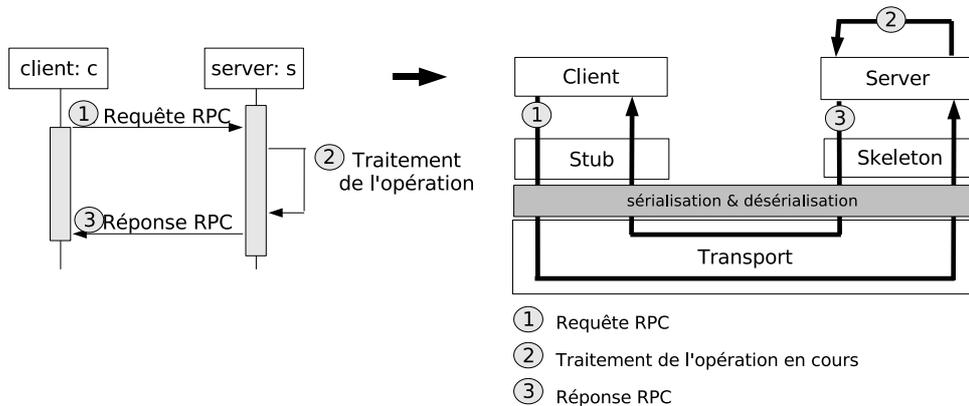


FIG. II.24 – *Remote Procedure Call*

Étant donné le critère impératif de multi-plateforme pour l'instrumentation distante, RMI³² (qui ajoute l'orientation objet à la notion RPC) a été étudié. En effet, RMI est issu du monde Java (par conséquent multi-plateforme). De ce fait, une première idée est de construire un objet côté serveur accédant à l'instrument et permettant à des clients autorisés de l'invoquer à distance.

Néanmoins, RMI atteint certaines limitations (Tannenbaum, 1988) handicapantes dans le cadre de l'instrumentation distante. En réalité, une implémentation de RPC (soit une bibliothèque implémentant la notion RPC comme, par exemple, RMI cité plus haut), bloque le déroulement de l'exécution du programme lorsque l'on utilise l'instruction « recevoir ». Or, lorsque le client a envoyé sa commande via la fonction « envoyer() » en utilisant RMI, il est probable que l'instruction suivante sera « recevoir() » pour obtenir le résultat de la requête émise avec la commande « envoyer() ». Le problème est que l'instruction « recevoir() » est bloquante ! En d'autres termes, tant que le client ne reçoit pas de données, l'exécution du programme est bloquée sur la ligne « recevoir() ». Cela est la définition même d'un intergiciel synchrone.

Pour les télé-TPs, cela peut se révéler être un lourd handicap : il est légitime de penser qu'une opération demandée à l'instrument peut nécessiter beaucoup de temps pour s'effectuer (de quelques secondes à plusieurs minutes pour un banc d'antennes de télécommunications) (Gravier & Fayolle, 2005). Il faut donc pouvoir offrir des possibilités d'interaction entre le système et l'utilisateur durant tout le traitement de l'opération que celui-ci aura sollicitée. A titre d'exemple, il doit être possible pour l'utilisateur de grossir la courbe affichée ou encore de consulter des aides en ligne via l'application. Ces fonction-

32. Remote Method Invocation

nalités doivent pouvoir être utilisées en dépit de l'attente de la livraison d'un résultat. De plus, RPC a originellement été construit dans l'optique d'une communication point à point et, de ce fait, ce modèle n'est pas adapté pour un contexte d'utilisation avec plus de 2 utilisateurs : il n'est pas possible avec RPC de joindre deux appels en un seul. Aussi, si le processus client devient faux (panne, erreur de transport, ...), il n'y a aucun moyen de récupération (de par le fait que la réponse n'est délivrée qu'en un seul transfert, quelle que soit la taille des données). Il serait de plus impossible d'envisager une implémentation sous forme d'un collecticiel du fait que les communications sont exclusivement point à point.

Ces limitations de RPC ont amené la plate-forme de télé-TPs à prendre en compte l'existence de solutions asynchrones. L'intergiciel asynchrone est un intergiciel dans lequel l'instruction « recevoir() » n'est pas bloquante, elle n'empêche pas de continuer l'exécution du programme. Étant donné que, de manière générale, les instruments ne sont accessibles que via des protocoles propriétaires, une solution plus proche du monde industriel a été recherchée. Ainsi, une alternative en terme d'intergiciel asynchrone est CORBA³³. CORBA est en fait un OOM³⁴. Le principe de fonctionnement d'un OOM est illustré (basiquement) sur la figure II.25.

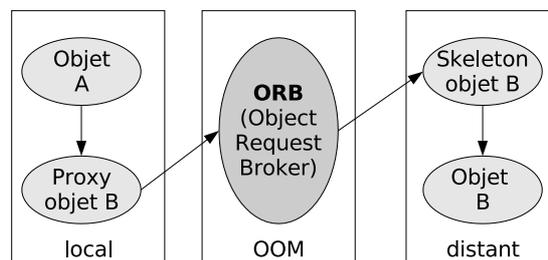


FIG. II.25 – Fonctionnement des OOM (Object Oriented Middleware).

CORBA s'apparente fortement à RPC mais un courtier (ORB³⁵) s'interpose entre l'objet local et l'objet distant (ou objet adressé). En réalité, le but des OOM, comme de RPC, est de proposer au développeur la manipulation d'objet distant comme s'il avait une existence dans l'applicatif client, via un représentant local (« proxy ») de l'objet en question. A l'inverse de RPC, cela signifie que le développeur n'a même pas besoin de savoir où se situe cet objet (l'ORB se charge de fournir la localisation de l'objet, on appelle ce mécanisme la transparence de localisation³⁶).

33. Common Object Request Broker Architecture, <http://www.corba.org/>

34. Object Oriented Middleware, soit en français un « intergiciel orienté objet », mais c'est l'anglisme qui est en vigueur

35. Object Request Broker, soit en français « courtier en requête d'objets », mais, une nouvelle fois, c'est l'anglisme qui est en vigueur

36. « localization transparency »

Les OOM sont historiquement des intergiciels synchrones. Nous avons déjà présenté précédemment les inconvénients d'un tel type d'intergiciel. Cependant, CORBA propose un autre modèle nommé « Event Channels » permettant de délivrer des messages via un canal d'évènements, et ce de manière asynchrone. Ceci constitue le mode « publish-register-notify » de CORBA. Toutefois, de nombreux problèmes liés à la sérialisation et au franchissement de pare-feu ont été soulevés lors de l'emploi d'une implémentation d'une plate-forme avec CORBA. De plus, même si CORBA tend à proposer une approche de plus en plus asynchrone (Siegel, 1999), il n'en reste pas moins qu'il fut originellement totalement synchrone. Il lui est difficile de faire abstraction de cet héritage historique.

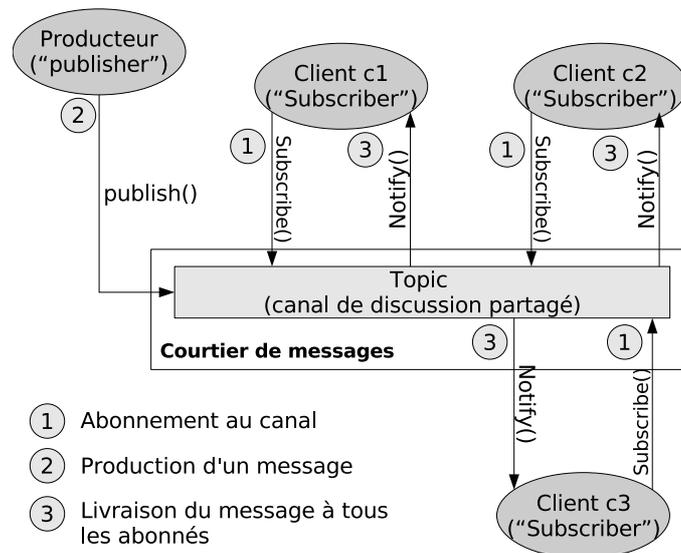


FIG. II.26 – Fonctionnement des MOM (Message Oriented Middleware).

Une alternative dans le champ des intergiciels asynchrones est le standard JMS³⁷ du monde J2EE³⁸. JMS introduit un modèle de communication basé sur la structure de messages : les MOM³⁹ (Oki *et al.*, 1993), (Maffeis, 1997). En effet, à l'inverse des OOM ou l'information échangée est un objet, les échanges dans les MOM sont basés sur des messages qui impliqueront une action sur l'objet local comme illustré sur la figure II.26. Ce modèle repose sur le mode de couplage « Publier/S'abonner »⁴⁰ (Eugster *et al.*, 2000), (Eugster *et al.*, 2001).

Suivons les étapes illustrées dans la figure II.26 :

- tous les clients intéressés par des évènements communs *s'abonnent* à un canal de

37. Java Messaging Service, <http://java.sun.com/products/jms/>

38. Java 2 Enterprise Edition

39. Message Oriented Middleware

40. Plus connu sous sa formulation anglaise: « Publish/Subscribe »

- discussion partagé (via la méthode de l'API⁴¹ d'un MOM « subscribe() »),
- un *Producteur* (« Publisher ») émet un message sur le canal de discussion,
- le *Courtier de messages* (« Message Broker ») va redistribuer le message publié sur le canal à tous les abonnés.

Il est important de noter que les MOM procurent à l'architecture qu'ils supportent des mécanismes de (Belissard *et al.*, 1999) :

- *Asynchronisme* dans la livraison de la réponse,
- *Robustesse* et *disponibilité* pour satisfaire l'objectif de passage à l'échelle,
- *Garantie d'ordonnement* qui assure que les messages seront réceptionnés dans l'ordre dans lequel ils ont été émis.

Enfin, il est très important de noter que la principale différence permettant de qualifier un intergiciel comme synchrone ou asynchrone réside dans l'aspect bloquant de l'instruction « recevoir() », comme énoncé plus haut.

<i>Cas d'un intergiciel synchrone</i>	<i>Cas d'un intergiciel asynchrone</i>
<pre> procédure executerCommande (E widget_id: Chaîne de caractères;) Déclaration msgEnvoye, msgRecu: Message début msgEnvoye ← creerMessage(widget_id); self.send(msgEnvoye); Remarque : Cette instruction est un appel bloquant (synchrone) msgRecu ← self.recevoir(); écrire(« Message reçu: »+msgRecu) fin </pre>	<pre> procédure executerCommande (E widget_id: Chaîne de caractères;) Déclaration msgEnvoye: Message début msgEnvoye ← creerMessage(widget_id); self.send(msgEnvoye); écrire(« message envoyé! ») fin procédure surReponse (E widget_id: Chaîne de caractères;) Déclaration msgRecu: Message début Remarque : Non bloquant car appelée à la réception d'un nouveau message (l'intergiciel se charge de la scrutation). msgRecu ← self.lireMessage(); écrire(« Message reçu: »+msgRecu) fin </pre>

TAB. II.3 – Différence entre la structure d'un programme se connectant à un intergiciel synchrone et celle se connectant à un intergiciel asynchrone.

Il devient alors aisé d'imaginer le concept de MOM appliqué à notre plate-forme de télé-TPs (Gravier & Fayolle, 2006b). Cela est illustré dans la figure II.27. L'implémentation

41. Rappel: Application Program Interface

JMS choisie est JORAM⁴².

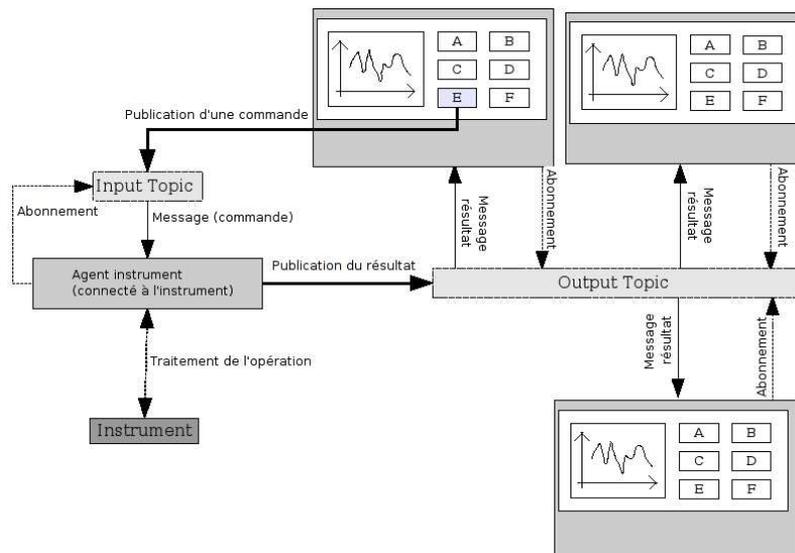


FIG. II.27 – Utilisation des MOM dans la plate-forme de télé-TPs

En réalité, nous ne faisons que suivre les recommandations de (Knublauch, 2004), mais en lieu et place d’une interconnexion via les technologies JSP ou Web Service comme originellement proposé par Knublauch, nous utilisons JMS pour obtenir l’accès au middleware asynchrone et surtout pour bénéficier de l’emploi d’un courtier de messages pour assurer l’aspect collaboratif de la téléopération.

En fait, l’indépendance aux dispositifs couplée à l’embarquement de la connectivité à l’intergiciel est rendue possible par la nature même de nos widgets propriétaires (les « IWidgets »). En effet, nos widgets se composent de trois blocs :

- Représentation graphique : composant(s) Java dérivé(s) de la bibliothèque Swing,
- Connectivité à l’intergiciel : méthodes publiques accessibles par introspection,
- Fidélité de représentation : méthodes privées (pour le multimédia (sons et images), ou pour le comportement des éléments graphiques comme le degré de résistance des potentiomètres lors de leur utilisation)

II.3.4 Serveur d’applications pour les composants métiers

Le mécanisme « Producteur/Abonnement » n’est pas suffisant pour satisfaire tous les besoins de la plate-forme de télé-TPs. C’est la raison pour laquelle un serveur d’applications est couplé à l’intergiciel asynchrone. Cela permet de fournir des mécanismes intégrés au serveur d’application, pour des briques logicielles hébergées, tels que la gestion :

- des *sessions* des utilisateurs,

42. Java Open Reliable Asynchronous Messaging, <http://www.objectweb.org/>

- de la *charge réseau* (mise en grappe, répartition de la charge, parallélisme de traitement, haute disponibilité),
- des *connexions ouvertes* sur un catalogue externe de données.

De plus, le choix d'un serveur d'applications J2EE permet d'implémenter des briques métiers (« EJB »⁴³) tirant pleinement avantage des mécanismes intégrés au serveur J2EE, tels que la gestion :

- des *accès concurrentiels*,
- de la *persistance des objets*,
- des *transactions*,
- du *passage à l'échelle*,
- de la *transparence de localisation*.

Choisir une implémentation d'un serveur d'applications

En ce qui concerne le choix de l'implémentation des serveurs d'applications, il est clair que le monde J2EE offre un très large panel de solutions. Plus précisément, et seulement pour lister les principaux acteurs du marché, on peut choisir une implémentation parmi :

- *Websphere d'IBM*⁴⁴,
- *JBoss de RedHat*⁴⁵,
- *JOnAS d'ObjectWeb*⁴⁶,
- *Geronimo de la fondation Apache*⁴⁷

Le choix exercé s'est porté sur le serveur d'application JOnAS⁴⁸. La première raison est que JORAM a été choisi comme implémentation de la norme JMS. Comme JORAM est issu du consortium ObjectWeb⁴⁹, l'orientation légitime est de choisir le serveur d'applications proposé par le même consortium. En fait, même si JORAM peut être distribué indépendamment de JOnAS, il est également intégré à défaut dans le paquetage de JOnAS, ce qui constitue également un atout indéniable. De plus, les tests effectués sur JOnAS le présentent comme un serveur très puissant et robuste. Aussi, les communautés d'utilisateurs construites autour de JOnAS et JORAM sont vraiment imbriquées et liées entre elles (les développeurs de JORAM interviennent sur JOnAS et vice-versa). Cela constitue des garanties de collaboration et d'intégration éprouvées entre ces deux types de serveurs. JOnAS apporte, par ailleurs, la garantie de la certification Sun J2EE 1.4, en ayant passé

43. Entreprise Java Bean

44. <http://www-306.ibm.com/software/websphere/>

45. www.jboss.org

46. jonas.objectweb.org

47. geronimo.apache.org

48. Java Open Application Server

49. <http://www.objectweb.org/>

plusieurs milliers de tests menés de manière indépendante⁵⁰.

Concernant notre plate-forme, plusieurs EJB⁵¹ développés durant cette thèse permettent de faire fonctionner la plate-forme de télé-TPs :

- *EJB Session* jouant le rôle de représentant des autres EJB non directement exposés,
- *EJB Entity* pour journaliser les accès auprès de la base de données hébergé par PostgreSQL, et pour extraire les statistiques d'utilisation,
- *EJB Message* pour publier une commande sur le canal de discussion de l'intergiciel asynchrone auquel l'instrument est abonné.

En réalité, il est très important de noter que la commande n'est publiée (i.e. l'instrument sollicité) que lorsque le client a été authentifié via le module J2EE JAAS⁵² au travers de l'annuaire d'authentification LDAP. Les EJB sont protégés via leurs fichiers de déploiement sur le serveur d'applications. Un « marqueur » de sécurité permet de spécifier quels sont les rôles (et par conséquent les groupes d'utilisateurs) autorisés à utiliser ledit EJB (le grain employé est la méthode). Nous avons déjà souligné précédemment (II.3.2) que JAAS établit alors la correspondance entre les rôles autorisés et les groupes d'un annuaire LDAP.

Au final, la figure II.28 ainsi que la figure II.29 dressent l'architecture complète du principe de fonctionnement de l'implémentation réalisée. Les messages retournés par l'instrument parcourent le même cheminement que les messages reçus par celui-ci, à savoir les phases d'authentification, de journalisation et de publication du message sur le canal de discussion partagé.

Pour une plus grande visibilité du logiciel développé, le lecteur pourra se rendre sur le site Internet du projet que nous avons mis en place :

<http://diom.istase.fr/satin/einst>

On y retrouve les grandes idées conductrices de nos travaux et surtout une vidéo de la plate-forme en fonctionnement⁵³.

Journalisation des actions menées

Le principe de journalisation des actions menées durant la session de télé-TP est primordiale pour deux raisons.

50. certification SUN J2EE 1.4, version de JOnAS 4.3.1, obtenue fin janvier 2005

51. Entreprise Java Bean

52. Java Authentication and Authorization Service

53. http://diom.istase.fr/satin/einst/einst_demo.avi

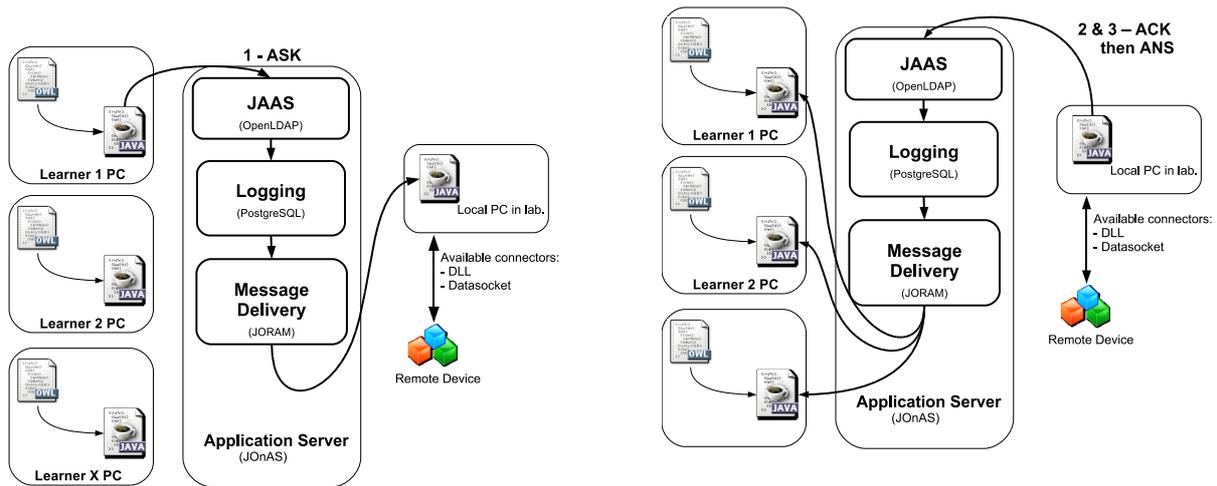


FIG. II.28 – Envoi d’une commande sur la plate-forme de télé-TPs.

FIG. II.29 – Retour des messages d’acquiescement et résultat de la commande sur la plate-forme de télé-TPs.

Tout d’abord, il est clair que tous les élèves et professeurs ne se connectent pas tous en même temps à la plate-forme. Par conséquent, il faut pouvoir amener le nouvel arrivant à la situation présente. Pour résoudre ce problème, deux possibilités a priori :

- Être en mesure de dresser un instantané de l’état de l’appareil, et ainsi reproduire cet état sur l’instance du nouvel arrivant,
- Rejouer les actions passées dans le même ordre afin d’arriver au même état de l’interface.

Bien que la première solution semble la plus élégante, nous pensons qu’elle n’est pas la plus aisée, et encore moins la plus profitable. En effet, pouvoir dresser une photographie de l’état d’un appareil à un instant donné, impose de connaître les capacités et les éléments du dispositif. Or cela est contraire à notre objectif : nous visons une plate-forme indépendante de l’instrument !

De plus, l’historique, même s’il est conséquent, n’est pas lourd à l’exécution. Il s’agit généralement de modifier la couleur d’un bouton et dans le cas le plus défavorable d’afficher une courbe. Nous avons donc opté pour la seconde solution : l’historique des réponses envoyées par un instrument aux clients et transmis au nouvel arrivant. C’est ainsi que les instances des nouveaux arrivants rejoignent l’état en cours. Par ailleurs, il est intéressant de noter que le tout premier utilisateur créant le télé-TP induit déjà une arrivée tardive : l’instrument a déjà effectué des actions dans sa phase d’initialisation, et il convient de répercuter leurs résultats sur la façade du créateur de la session de télé-TP. En d’autres termes, les arrivées tardives sont gérées par un « rejeu » de l’historique des messages.

Un second avantage de la journalisation est la possibilité de sérialisation du journal dans un fichier. L’historique de la séance de télé-TP ainsi sauvegardé, il devient possible,

via le développement de l'applicatif correspondant, de rejouer la manipulation plus tard. En d'autres termes, il est offert aux élèves de rejouer un réel TP, qui a eu lieu dans le passé. L'étudiant peut alors jouer sur la vitesse de lecture, aller en arrière ou en avant. Une simple journalisation devient alors un formidable outil pédagogique. Nous aurons l'occasion dans les chapitres suivants de revenir sur les effets pédagogiques de tels outils IV.1.

II.3.5 Points remarquables de l'implémentation

Indicateur de téléprésence

D'autres outils viennent renforcer notre implémentation. Tout d'abord, il est possible de visualiser instantanément la couleur associée à chaque personne dans une zone de l'écran (ainsi que la personne ayant la main sur l'appareil, mais cet aspect fera l'objet du chapitre suivant). Le jeu de couleurs (limité à 15 couleurs) a déjà été évoqué dans l'établissement de la conscience de groupe. L'usage du jeu de couleur facilite en effet la correspondance entre les actions réalisées et les auteurs de ces actions (voir figure II.30).

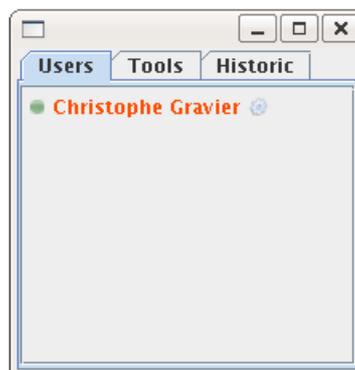


FIG. II.30 – *Indicateur de téléprésence développé pour la plate-forme.*

Adaptation de la façade envers l'apprenant

Par ailleurs, il est possible pour un étudiant de volontairement épurer la façade de l'appareil piloté. En effet, l'appareil piloté peut, de prime abord, présenter une certaine austérité pour le néophyte. Nous avons néanmoins veillé à préciser pour chaque widget, le degré de compétence requis pour son usage, et représenté cette connaissance dans notre ontologie (figure II.8).

De ce fait, l'étudiant peut, de lui-même, choisir le niveau de représentation (débutant, intermédiaire, avancé, expert) associé aux compétences qu'il pense être les siennes. Le résultat du dégradé d'une façade de dispositif est représenté à la figure II.31. La qualification des widgets (quel niveau pour quel widget) incombe au producteur du télé-TP

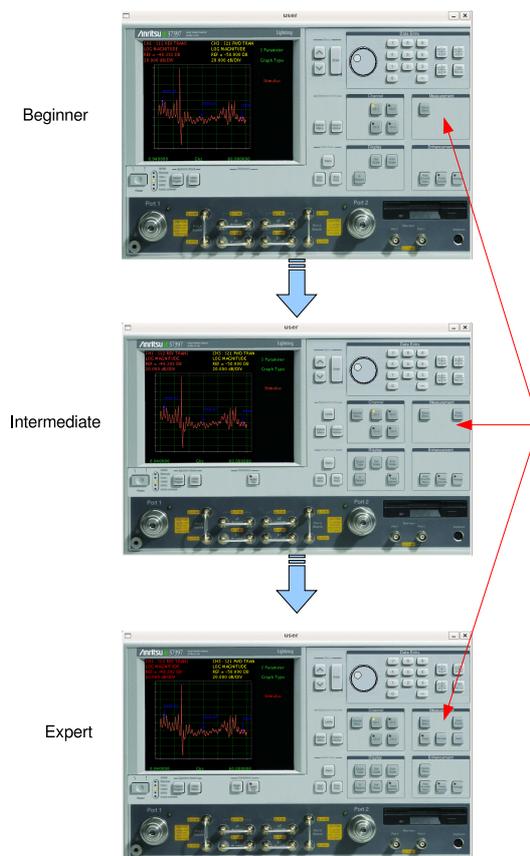


FIG. II.31 – L'étudiant peut appauvrir ou enrichir la représentation de la façade suivant le degré de compétence qu'il estime être le sien (voir la figure II.8 pour le concept Level).

ayant les compétences requises pour exploiter le dispositif. C'est également lui qui produit l'ontologie du dispositif à l'aide de l'atelier de génie logiciel.

Bien entendu, un tel mécanisme gagnerait à être adapté en fonction des actions de l'utilisateur. On pourrait par exemple imaginer que les actions de l'utilisateur (bonnes ou mauvaises) conditionnent le degré de complexité de l'interface. Un tel projet est conséquent et soulève de nombreuses questions. Nous nous bornerons donc à le présenter dans le paragraphe relatif aux futurs travaux (voir IV.2.2), avec une implémentation possible, puisque nous avons déjà réfléchi à la question, mais qu'aucune implémentation n'a été produite.

Messagerie instantanée

Nous avons évoqué le fait que les messages d'acquittements et de résultats de commandes sont publiés vers toutes les instances de tous les utilisateurs de la session de télé-TP. Nous avons également ajouté que les modifications de la façade consécutives à la réception de ces messages impliquaient une coloration suivant la couleur associée à l'auteur de ces messages. Cela nous permet d'avoir conscience instantanément des actions d'autrui (il s'agit de la notion de « conscience de groupe » définie dans le domaine du Travail Collaboratif Assisté par Ordinateur).

Néanmoins, cela ne permet pas de supporter suffisamment de collaboration entre les apprenants pour favoriser l'émulation, le partage de savoir et de connaissances, de résultats, l'entretien des relations sociales propres à un groupe, ... (I.2.3.2). Nous avons alors évoqué le besoin d'outils de communication entre les utilisateurs pour maximiser le gain pédagogique d'une séance de télé-TP (I.2.3.1). Pour assurer la communication, nous nous appuyons sur un serveur de messagerie instantanée et de voix sur IP. Il s'agit du protocole XMPP⁵⁴ du système de messagerie instantanée Jabber⁵⁵. Une très bonne définition est celle de Wikipédia⁵⁶ : « XMPP est à Jabber ce que HTTP⁵⁷ est au Web ».

En dehors de l'extraordinaire robustesse de XMPP, deux raisons nous poussent à retenir ce protocole : il est ouvert et supporté par l'IETF⁵⁸. XMPP est par exemple le standard retenu par Google pour « GoogleTalk », son service de messagerie instantanée. Ce service que nous avons apporté à notre plate-forme de télé-TPs repose sur l'implémentation Open-Fire⁵⁹. Par ailleurs, le client jabber Jabbin⁶⁰ (puisque'il faut user d'un client indépendant à la plate-forme de télé-TP pour se connecter au serveur Jabber), permet d'utiliser XMPP

54. eXtensible Messaging and Presence Protocol, basé sur le format XML

55. <http://wiki.jabberfr.org/>

56. http://fr.wikipedia.org/wiki/Extensible_messaging_and_presence_protocol

57. HyperText Transfer Protocol

58. Internet Engineering Task Force

59. <http://www.igniterealtime.org/projects/openfire/index.jsp>

60. www.jabbin.com/fr/

pour relayer des messages vocaux et donc offrir la fonctionnalité de voix sur IP aux apprenants et tuteurs durant le télé-TP. Les messages sont cryptés sur un canal sécurisé (SSL) et l'authentification est, à l'instar de la plate-forme de télé-TP, effectuée sur un serveur LDAP. Cela permet aux utilisateurs de ne mémoriser qu'un seul mot de passe.

CHAPITRE II

Généraliser les dispositifs en ligne

Conclusion

Ce chapitre a principalement démontré qu'il est possible de factoriser les développements logiciels lourds des télé-TPs. Nous avons proposé une approche et les implémentations associées pour tendre vers une plate-forme de télé-TP indépendante des TP qu'elle héberge. Autrement dit, nous visons à factoriser tout ce qui est commun à chaque télé-TP : intergiciel et construction de la façade déportée.

Cette approche passe par la conception d'une ontologie des façades d'instruments. Ces façades sont constituées d'éléments graphiques (widgets) embarquant la connectivité à l'intergiciel. Cela signifie que le simple fait de les instancier suffit à envoyer et recevoir des messages à l'utilisation. C'est autant de développement ôté à l'architecte d'un télé-TP.

La plate-forme implémentée⁶¹ s'appuie sur un intergiciel asynchrone (produit du consortium ObjectWeb : JORAM). L'intergiciel relaie les messages via un serveur d'application (JOnAS), lequel est responsable des phases d'authentification/autorisation (OpenLdap) et de la journalisation des messages (PostgreSQL). Il est certain qu'à l'heure des choix technologiques, l'accent a été mis sur les standards ouverts et les implémentations libres correspondantes. Ces différentes briques logicielles forment une plate-forme collaborative. En effet, plusieurs utilisateurs peuvent se connecter simultanément à la plate-forme et suivre les actions menées en temps réel, tout en connaissant l'auteur de chaque action (de par l'association d'une couleur à chaque personne). Ainsi, l'implémentation offre des mécanismes de téléprésence et de conscience de groupe.

Il ne faut néanmoins pas sous-estimer l'aspect collaboratif : nous avons vu au premier chapitre que la réussite des logiciels pour l'enseignement pratique à distance passe par la communication entre les utilisateurs. Il demeure cependant un aspect volontairement occulté jusque là : comment partager et segmenter le temps d'accès aux dispositifs, lorsque les utilisateurs sont plusieurs face à un seul dispositif ? La réponse à cette question fera l'objet du chapitre suivant.

⁶¹. dont nous rappelons qu'une vidéo de démonstration en utilisation est disponible à l'adresse <http://diom.istase.fr/satin/einst/>

Orchestrer l'accès au dispositif

Sommaire

Introduction	133
III.1 Éléments de contexte	134
III.1.1 Le besoin de régulation des accès collaboratifs	134
III.1.2 Représentation des éléments issus de la collaboration . . .	136
III.1.3 Évaluation de l'approche	139
III.1.4 Bilan	142
III.2 Politique de collaboration	145
III.2.1 Un ensemble de règles comme politique de partage	145
III.2.2 Exprimer les politiques d'accès avec SWRL	147
III.2.3 Autres exemples de politiques d'accès	148
III.2.4 Bilan	152
III.3 Implémentation du système d'orchestration	153
III.3.1 Architecture complète	153
III.3.2 De l'importance de pouvoir modifier la politique en vigueur	156
III.3.3 Couplage de l'unité de modération avec la plate-forme de télé- TP	159
III.3.4 Problèmes d'implémentation rencontrés	160
III.3.5 Bilan	165
Conclusion	167

Orchestrer l'accès au dispositif

Introduction

La problématique de l'apprentissage collaboratif assisté par ordinateur requiert des logiciels collaboratifs (et non coopératifs) au sens de Dillenbourg (Dillenbourg, 1999). Les apprenants sont amenés à réaliser conjointement un objectif partagé. Cette vision est nuancée par l'usage d'un seul dispositif commun à tous les utilisateurs : l'environnement produit est alors tantôt collaboratif, tantôt concurrentiel. C'est d'ailleurs cet aspect qui pousse à déporter la façade d'un instrument, au lieu de fournir un simple retour vidéo. Le retour vidéo ne peut en effet présenter l'interaction originelle, pas plus que la détermination visuelle de l'auteur d'une commande (aucune possibilité de « personnalisation » d'une action).

Dès lors, sur quels critères s'appuyer pour désigner la personne prioritaire pour l'accès à l'instrument (tandis que les autres regardent)? Comment départager deux personnes qui demandent l'accès exclusif à la ressource distante, alors que nous sommes dans un cadre collaboratif (et donc pas complètement concurrentiel)?

III.1 Éléments de contexte

III.1.1 Le besoin de régulation des accès collaboratifs

Notre modèle de télé-TP et le prototype correspondant, exposés au chapitre précédent, présentent une limitation importante en terme de collaboration. Certes, lorsqu'une commande est envoyée par un utilisateur, tous les participants aux télé-TPs sont notifiés de celle-ci, via un acquittement coloré, permettant d'identifier l'auteur de l'action. Néanmoins, nous n'avons pas décrit comment l'accès aux commandes était orchestré. En effet, permettre à tous les utilisateurs de mener des actions en même temps aurait pour effet une diminution de la conscience de groupe, puisque cela conduirait à l'augmentation de la concurrence au sein de celui-ci. Ainsi, nous pensons que l'absence de régulation dans les accès à l'instrument ne favorise pas la compréhension. En effet, si plusieurs utilisateurs conduisent des actions en même temps, sans rapport entre elles, il devient difficile de comprendre la logique dans la suite d'actions entreprises. Cela peut même remettre en cause l'intégrité de l'instrument (en cas de commandes incompatibles par exemple).

Si l'on se penche sur l'existant, il est clair que l'accès concurrentiel à une ressource distante n'est pas une problématique nouvelle en génie logiciel. Dans le domaine de l'ordonancement des processus, les réponses apportées sont multiples : modèles FIFO¹, FILO², dit du « tourniquet »³, ...

Néanmoins, ces approches présentent des limites si l'on tente de les appliquer à notre domaine. En effet, les situations pédagogiques varient au cours d'une séance de télé-TP, cette variation se traduit par différentes séquences d'utilisation de la plate-forme, et donc se répercute sur les modalités de partage de la ressource distante. Cela signifie qu'une séance d'enseignement pratique est rythmée par différentes phases, pas nécessairement connues à l'avance. Par exemple, l'enseignant peut souhaiter donner plus de temps de manipulation aux personnes les plus en difficulté, ou à celles qui ont eu le moins de temps de manipulation, ou à l'inverse aux derniers arrivés afin de leur permettre de rattraper leur retard. Les exemples de situations sont pléthores, et nous ne pouvons pas dresser une liste exhaustive ici.

Les besoins d'orchestration de la collaboration au sein des télé-TPs sont très différents d'une séance à l'autre, et dépendent aussi de la pédagogie mise en place par le tuteur enseignant. C'est alors que se dressent les limites des modèles actuels. Un algorithme gérant

1. « First In, First Out », autrement dit « premier arrivé, premier servi »
2. « First In, Last Out », autrement dit « premier arrivé, dernier servi »
3. ou « round robin »

l'accès à une ressource ne change pas au cours du temps, ce sont les mêmes conditions d'accès quelque soit le contexte d'exécution. Dans ces conditions, il conviendrait de fournir autant d'algorithmes décrivant une orchestration qu'il peut exister de modalités de partage de l'instrument.

Ce n'est pas une solution envisageable car les situations peuvent varier énormément d'un tuteur ou d'une séance à l'autre, ce qui conduit à l'explosion du nombre de développements nécessaires. De plus, il est certain que les pédagogies sont en constante évolution, ce qui signifie qu'un algorithme spécifiant le partage d'un instrument pour une pédagogie actuelle ne sera pas forcément réutilisable pour les formes d'enseignement futures.

Nous désirons apporter une nouvelle vision du partage d'une ressource distante. Nous estimons qu'une approche plus efficace consiste à prendre en compte les éléments du contexte dans le but de prendre une décision sur la priorité entre plusieurs utilisateurs faisant la demande d'accès au dispositif distant. Dans le cadre de notre travail, les éléments de contexte sont issus de la manipulation collaborative. En d'autres termes, nous pensons que le plus logique est de positionner la politique de partage à mettre en place en fonction de la pédagogie mise en œuvre. Il s'agit ainsi de mettre en adéquation la politique décrivant les modalités de partage de la ressource distante par rapport à la pédagogie, et non l'inverse!

Pour obtenir un tel modèle, nous nous basons sur la réflexion suivante. Certes, nous avons vu que les modalités de partage d'un instrument exigibles par les tuteurs enseignants de télé-TPs collaboratifs sont nombreuses. Néanmoins, les éléments qui définissent ces modalités de partage sont souvent les mêmes. L'approche que nous défendons est donc de formaliser ces éléments de contexte, récurrents dans la plupart des politiques de partage.

Pour bien comprendre, nous pouvons citer, comme éléments revenant dans de nombreuses politiques d'orchestration, l'heure de connexion, le temps de connexion, le temps cumulé de manipulation, le rôle de l'utilisateur, le temps d'attente, etc. Prenons un exemple simple : à l'aide de deux politiques de partage très différentes : une première donnant la priorité aux derniers arrivés et une seconde offrant l'avantage d'accès aux premiers arrivés. Dans les deux cas, c'est à chaque fois la notion d'heure de connexion qui importe pour déterminer lequel des utilisateurs a priorité sur l'autre.

Notre idée a donc été de formaliser un certain nombre de ces éléments les plus récurrents, pour ensuite construire, à la demande, des politiques de partage adaptées au besoin de la pédagogie à mettre en place. Cette approche a l'avantage de permettre

la construction personnalisée de politiques de partage adaptées à chaque situation, du moment que l'élément de contexte entrant en compte dans l'énonciation de la politique a été modélisé au préalable, et donc que le système parvient à l'interpréter.

Nous représentons les éléments de contexte dans une ontologie, puisque c'est intrinsèquement le rôle d'une ontologie que de capturer sous un certain formalisme, les concepts et relations d'un domaine donné. Ces éléments sont bien sûr ceux sur lesquels s'appuie une politique de régulation d'actions d'un dispositif éloigné pour une séance de télé-TP. Grüber n'a-t-il pas exposé cette année que les « killer applications »⁴ des ontologies se trouvent être à l'intersection du Web social et sémantique, qu'il appelle alors « Collective Knowledge Systems » (Gruber, 2007)? La solution que nous proposons dans ce chapitre permet de s'affranchir du problème de développement d'une politique de partage au besoin, par l'utilisation d'une ontologie. Nous visons donc à ce que la mise en place d'une politique de partage se réduise à l'édition d'une ontologie, et non plus au développement d'une nouvelle « verrue applicative » pour l'unité d'orchestration de la collaboration au sein des télé-TPs collaboratifs.

III.1.2 Représentation des éléments issus de la collaboration

Tous les concepts identifiés sont représentés par la figure III.1. Il s'agit donc ici d'une ontologie de domaine.

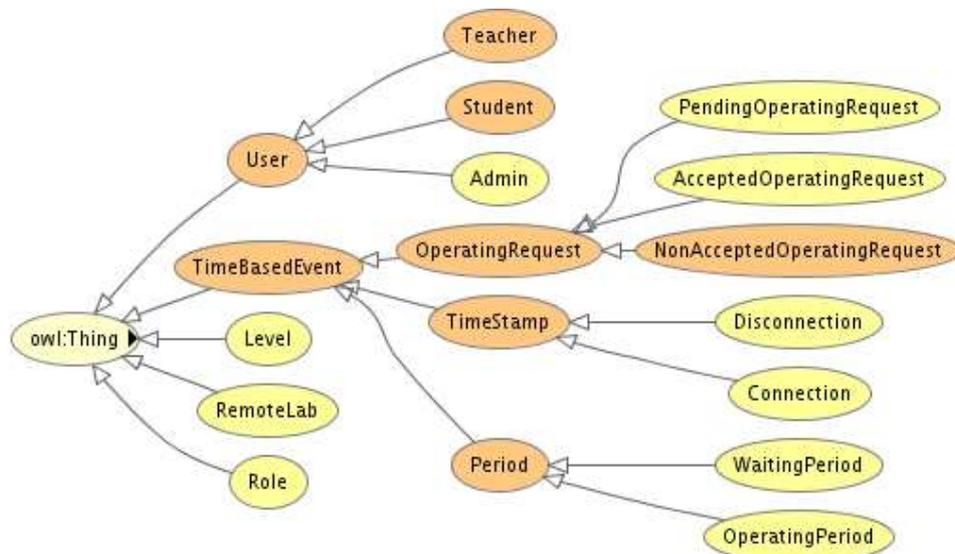


FIG. III.1 – Ontologie des éléments de contexte propres à une situation de télé-TP collaboratif.

4. Une « killer application » dans le jargon du génie logiciel qualifie une application qui justifie par elle-même l'emploi d'une technologie. Par exemple le World Wide Web est une « killer application » du réseau Internet.

Ces concepts sont issus de l'observation et de la réponse à la question suivante : quels éléments participent aux politiques gouvernant le partage des dispositifs pour la mise en place de collaborations entre apprenants durant un télé-TP ? Cette ontologie ne se prétend pas exhaustive (nous verrons quelles en sont les extensions possibles dans la partie III.1.3).

Nous pouvons répartir les éléments de contexte que nous avons identifiés en trois grandes catégories :

1. le *rôle* de l'utilisateur du télé-TP⁵, que ce soit l'étudiant, le tuteur enseignant, ou l'administrateur du télé-TP (qui est bien souvent le tuteur enseignant lui-même, mais ce n'est pas une obligation, et l'on observe quelques entorses à cette généralité dans la pratique).
2. le *niveau de compétence* (d'un utilisateur apprenant). Il s'agit du concept importé de l'ontologie vue au chapitre précédent. Précédemment le niveau de compétence était le degré d'expertise requis pour obtenir l'affichage d'un élément graphique. Ici ce même concept permet de modéliser le degré de connaissances d'un utilisateur. On rappellera simplement que les différentes valeurs possibles sont : Beginner, Intermediate, Advanced, Expert. Des travaux sont en cours (Gravier *et al.*, 2006a) pour rendre ce niveau d'utilisation dynamique (évolue positivement ou négativement suivant les actions menées par l'apprenant), et ainsi promouvoir une adaptation du scénario pédagogique suivi par l'étudiant en fonction de l'évolution de ses compétences.
3. les *éléments qui dépendent du temps*, comme par exemple le temps de connexion d'un utilisateur, le temps cumulé d'attente, le temps cumulé en ayant endossé le rôle d'opérateur, ...

Voici le tableau des concepts⁶ actuellement présents dans notre ontologie (Les colonnes et leurs significations demeurent les mêmes que celles employées dans le tableau II.1).

Bien sûr, ce tableau ne représente que les concepts, sans les relations liant ces concepts entre eux. Pour prendre connaissance de celles-ci, il est possible de consulter l'ontologie complète sur Internet⁷. Le lecteur peut également consulter le tableau III.2.

Comme pour l'ontologie de la façade d'un instrument, nous nous sommes appliqués à construire cette ontologie de manière pragmatique, c'est-à-dire en visant la simplicité et la concision. La finalité est de favoriser sa réutilisation et son évolution. Or, de la même façon que le langage structurant les ontologies se doit de ne pas être trop complexe (Rousset *et al.*, 2004), nous pensons que les ontologies elles-mêmes doivent rester

5. Alors que les classes « Student » et « Teacher » sont disjointes, elles ne le sont pas de la classe « Administrator », puisqu'un administrateur peut très bien être un enseignant ou, pourquoi pas, un étudiant.

6. À l'exclusion de « Level » déjà défini précédemment, ce concept représentant le niveau de compétences de l'apprenant à un instant *t*.

7. <http://dev.istase.fr/satin/rlab/collaborativev4.swrl.owl>

Nom de la classe	Type	Classe parente	Restrictions	Disjointe de
RemoteLab	P	owl:thing	hasOperator allValuesFrom User hasParticipants some User	User, TimeBasedEvent, Role, Level
Role	D	owl:thing	Administrate Study Teach	Level, RemoteLab, TimeBasedEvent, User
TimeBasedEvent	D	owl:thing	TimeStamp or Period or OperatingRequest	RemoteLab, Level, User, Role
OperatingRequest	D	TimeBasedEvent	AcceptedOperatingRequest or NonAcceptedOperatingRe- quest or PendingOperatingRequest	Period, TimeStamp
AcceptedOpera- tingRequest	P	OperatingRequest	is-a OperatingRequest	NonAccepted- OperatingRequest, PendingOpera- tingRequest
NonAccepted- OperatingRequest	D	OperatingRequest	is-a OperatingRequest not AcceptedOperatingPeriod not PendingOperatingRequest	AcceptedOpera- tingPeriod, PendingOpera- tingRequest
Period	D	TimeBasedEvent	is-a TimeBasedEvent OperatingPeriod or WaitingPeriod	TimeStamp, OperatingRequest
OperatingPeriod	P	Period	is-a Period	WaitingPeriod
WaitingPeriod	P	Period	is-a Period	OperatingPeriod
TimeStamp	D	TimeBasedEvent	is-a TimeBasedEvent Connection or Disconnection	Period, OperatingRequest
User	D	owl:thing	Student or Teacher or Admin	RemoteLab, Level, Role, TimeBasedEvent
Admin	P	User	is-a User hasRole has Administrate	∅
Student	D	User	is-a User hasRole has Study	Teacher
Teacher	D	User	is-a User hasRole has Teach	Student

TAB. III.1 – Concepts pris en compte dans une stratégie de partage d'un appareil impliqué dans un télé-TP.

concises (mais précises) pour passer à l'échelle et être réutilisées. Nous pensons donc qu'il est important de représenter le domaine considéré avec suffisamment d'exactitude, mais sans tomber dans le piège de l'exhaustivité. En effet, une description sémantique formalisant un domaine de manière beaucoup trop fidèle réduit les possibilités de réutilisation. Conserver cette balance entre exactitude et réutilisation ne va pas sans compromis. C'est en ce sens que nous pensons que l'évaluation d'une ontologie ne doit pas se focaliser sur la vérification de son intégrité (des raisonneurs existent à cet effet⁸!), mais à l'inverse se concentrer sur la pertinence des choix effectués pour équilibrer au mieux l'ontologie dans ce rapport entre précision et potentiel de réutilisation.

III.1.3 Évaluation de l'approche

Quelle que soit l'ontologie, quelques principes de base permettent d'éviter des erreurs de conceptualisation handicapantes. Les ontologies sont une manière de fournir une modélisation basée sur des déclarations et des descriptions de concepts. Cela signifie que leur but premier est de représenter les « choses »⁹ d'un domaine donné. Il est donc primordial de vérifier si une ontologie réussit à capturer les connaissances que nous avons d'un domaine, dans un fichier informatique compréhensible par la machine. Tandis que les raisonneurs se focalisent sur la vérification de la consistance sémantique en se basant sur la syntaxe, une évaluation humaine permet d'appréhender la qualité de la représentation effectuée, en terme de granularité. Le principe de base est de décrire le domaine « ni plus, ni moins ». Tous les concepts, les relations ou les propriétés doivent revêtir un caractère primordial à l'appréhension du domaine considéré pour figurer dans l'ontologie. Le superflu est à proscrire. Il faut même aller plus loin : une ontologie ne doit poser que les déclarations les plus importantes (et omettre les moins significatives, même si elles sont correctes). En effet, plus l'ontologie embarque de vocabulaire, plus elle présente de risques de conflits avec des définitions futures. L'ontologie ne doit donc pas être complète mais suffisante pour le domaine¹⁰. Cette volonté de limiter l'expression d'une ontologie à son strict nécessaire pour ne pas mettre en péril les extensions futures, est connue sous le nom d'engagement ontologique minimal¹¹. Il traduit également l'accord passé entre deux systèmes pour l'utilisation d'une sémantique, s'appuyant sur une granularité de description partagée, capturée dans une ontologie.

Malheureusement, comme le souligne Grüber depuis le début de ses travaux sur les

8. Le raisonneur le plus connu et celui que nous avons employé est « Racer », (RenamedABox and Concept Expression Reasoner), <http://www.racer-systems.com/>

9. allusion au concept racine de toute ontologie : « owl:thing »

10. On retrouve ici l'approche pragmatique que nous avons mené pour nos deux ontologies : façade d'un instrument au chapitre précédent et ontologie des éléments de politique d'accès dans le présent chapitre

11. et plus précisément son équivalent anglo-saxon « minimal ontology commitment »

Nom de la propriété	Domaine	Portée	Fonctionnelle	Fonctionnelle inverse	Symétrique	Transitive	Propriété inverse
hasParticipant	RemoteLab	User	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasControlOver	User	RemoteLab	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	$(hasControlOver)^{-1} = hasOperator$
hasAccepted-OperatingRequest	User	AcceptedOperatingRequest	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasDisconnected	User	Disconnexion	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasRole	User	Role	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasPending-OperatingRequest	User	PendingOperatingRequest	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasOperating-Period	User	OperatingPeriod	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasOperator	RemoteLab	User	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	$(hasOperator)^{-1} = hasControlOver$
hasNonAccepted-OperatingRequest	User	NonAccepted-OperatingRequest	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasConnection	User	Connection	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset
hasWaitingPeriod	User	WaitingPeriod	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	\emptyset

TAB. III.2 – Relations entre les éléments de contexte qui déterminent une session de télé-TP.

ontologies (Gruber, 1993), nous faisons des choix pour représenter le domaine. Les choix favorisent certains aspects du domaine au détriment d'autres. Le design d'une ontologie force aux compromis. C'est la raison pour laquelle l'évaluation d'une ontologie ne garantit pas sa pertinence, mais argumente plutôt sur les choix de représentations effectués. Finalement, c'est un bon moyen pour le concepteur de s'interroger sur ses propres choix.

Les critères de base de l'évaluation ontologique, définis par Grüber (Gruber, 1993 ; Gruber *et al.*, 2007), sont les suivants :

- clarté
- cohérence
- extensibilité
- biais d'encodage minimal
- engagement ontologique minimal

L'évaluation d'une ontologie n'est pas triviale ni instantanée, puisqu'il n'existe pas de mesure numérique pour ces critères.

Nous pensons que notre ontologie satisfait le critère de clarté, sur la base des deux observations suivantes :

- toutes les classes et toutes les propriétés sont documentées en langage naturel,
- les classes sont définies par l'utilisation de restrictions nécessaires et suffisantes, ce qui tend à favoriser le formalisme.

Le critère de cohérence exige qu'aucune inférence ne puisse impliquer des assertions contredisant l'ontologie de départ. Nous n'en avons pas trouvé, que ce soit par l'utilisation d'un raisonneur ou par simple déduction, ce qui ne garantit malheureusement pas qu'il n'y en ait aucune.

L'extensibilité est le troisième critère à évaluer. Nous proposons ici encore une approche simple et pragmatique, à finalité fonctionnelle. Les concepts de base hébergés dans notre ontologie définissent et qualifient les utilisateurs, les rôles endossés par ceux-ci et le concept de télé-TP (classe « RemoteLab » dans l'ontologie). De plus, cette base est enrichie de concepts dépendant du temps pour qualifier les états relatifs à la situation de collaboration pour chacun des utilisateurs (temps d'attente cumulé, heure d'arrivée, ...). Nous avons voulu rendre ces déclarations indépendantes les unes des autres, et introduire un maximum de disjonctions entre les concepts pour lutter contre l'hypothèse d'un monde ouvert (« Open World Assumption »¹²). Le fait de présenter une ontologie en suivant une

12. Si l'on prend l'exemple suivant : « je suis guitariste », et posons ensuite la question : « suis-je un batteur ? ». Dans l'hypothèse d'un monde fermé je ne peux être les deux à la fois, dans l'hypothèse du monde ouvert, je peux avoir la double compétence.

approche pragmatique nous permet donc de satisfaire, dans la mesure du possible, au critère d'extensibilité.

La quatrième critère est le biais d'encodage minimal. Nous pensons que l'ontologie proposée n'est pas idéale. En effet, il évalue le biais introduit par l'implémentation. Dans notre ontologie, nous utilisons la notion de temps. Et le temps demeure une « chose » difficile à modéliser dans une ontologie, parce qu'il dépend du domaine auquel on fait référence (Hobbs & Pan, 2004). Par conséquent, nous avons dû utiliser le type de donnée « entier » pour représenter une période de temps (dans le sens « une certaine quantité de temps »). Ce n'est probablement pas le meilleur moyen de représenter cela, mais c'est certainement le plus pratique et surtout le plus réalisable en ce qui concerne l'implémentation.¹³

Nous avons également introduit une part de biais d'encodage à cause de l'implémentation du moteur de règles (cela sera expliqué plus tard III.3). Nous verrons alors que la réalité de l'implémentation des systèmes à base de connaissances peut avoir un impact sur la représentation du domaine (et donc favoriser un biais d'encodage).

Nous pensons, à l'inverse, avoir respecté le dernier critère. Il s'agit de l'engagement ontologique minimal défini précédemment. Nous pensons nous être engagés dans une approche décrivant suffisamment le domaine pour valider le principe, sans être allés trop loin dans le niveau de détails. Par ailleurs, comme Guarino le souligne (Guarino, 1998), ce critère suggère également de minimiser l'impact des connaissances du domaine du concepteur de l'ontologie. En effet, le choix des mots représentant un concept demeure fondamental.

Le tableau III.3 résume ainsi les qualités et faiblesses de notre ontologie, à la lumière des cinq critères de Grüber.

III.1.4 Bilan

L'évaluation est une question importante pour la modélisation conceptuelle, et plus particulièrement celle des ontologies. En effet, la connaissance que nous avons d'un domaine, et donc la représentation que l'on s'en fait, est volatile, puisqu'elle évolue avec le temps. La meilleure représentation d'un domaine à un instant donné peut rapidement devenir obsolète, voire erronée. C'est la raison pour laquelle les ontologies de domaines sont rapidement remodelées : les domaines évoluent souvent, et les ontologies qui les décrivent se

13. Il est à noter que la bibliothèque utilisée dans l'implémentation s'est vu récemment enrichie des opérateurs temporels adéquats, ce qui n'était pas le cas au moment de notre implémentation.

Critères	Qualités	Défauts
Clarté	<ul style="list-style-type: none"> – Conditions nécessaires et suffisantes si possible – Concepts, relations et propriétés sont documentés 	
Cohérence	<ul style="list-style-type: none"> – Aucune inférence identifiée contredisant les axiomes de l'ontologie de base 	
Extensibilité	<ul style="list-style-type: none"> – Concise – Approche pragmatique – Peu de concepts interdépendants 	
Biais d'encodage minimal		<ul style="list-style-type: none"> – L'implémentation a forcé à des solutions de contournement modifiant le design original de l'ontologie – les limitations du moteur de règles impliquent aussi des modifications du design original de l'ontologie
Engagement ontologique minimal	<ul style="list-style-type: none"> – L'ontologie vise à définir la théorie la plus minimaliste – La majorité du vocabulaire employé ne dépend pas du domaine 	<ul style="list-style-type: none"> – Quelques termes nécessitent néanmoins la lecture des descriptions en langage naturel pour être compris

TAB. III.3 – *Évaluation de notre ontologie à la lumière des critères de Grüber*

doivent de suivre ces évolutions.

Nous pensons avoir expliqué que, pour nous, l'objectif de l'évaluation d'une ontologie n'est pas de vérifier l'intégrité d'une ontologie (les raisonneurs s'en chargent) mais d'évaluer le potentiel d'évolution présenté par l'ontologie. Ce potentiel est évalué à l'aide de différents critères et nous avons choisi ici les plus connus et les plus largement acceptés, ceux de Grüber. Au delà de la prise de conscience des forces et faiblesses de notre ontologie, nous pensons qu'une telle étude vise à estimer ce que l'on pourrait nommer la « durabilité »¹⁴ d'une ontologie. En conclusion, nous pensons fournir un modèle pragmatique (car à finalité d'implémentation) qui permet d'adapter les modalités d'accès au sein de manipulations en fonction des situations pédagogique.

14. double sens : dureté (robustesse) et durée (longévité)

III.2 Politique de collaboration

III.2.1 Un ensemble de règles comme politique de partage

La section précédente décrit l'ontologie sur laquelle l'aspect collaboratif de la plateforme de télé-TP repose. Nous identifions des éléments du partage d'une ressource dans les situations d'apprentissage collaboratif assisté par ordinateur. L'utilisation de cette ontologie n'a pas encore été exposée : ce sera l'objet de la présente section.

Rappelons pourquoi nous avons opté pour une représentation conceptuelle de notre problème : la liste des politiques de partage du dispositif lors d'un télé-TP est très importante (et sujette à des ajouts en permanence), mais les éléments dont elles dépendent ne sont pas si nombreux. Notre objectif est alors de construire un système permettant de mettre en place une politique de partage de l'instrument distant adaptée à l'expérience d'apprentissage que l'enseignant veut délivrer aux étudiants.

En utilisant l'ontologie précédemment décrite, nous pouvons construire des règles, s'appuyant sur les concepts décrits afin de modéliser les conditions d'accès à la ressource distante, c'est-à-dire les conditions qui permettent à un utilisateur, à un moment donné, de déloger l'utilisateur opérateur de la télé-opération. Ainsi, au fur et à mesure que la base de connaissance évolue¹⁵, les conditions restent les mêmes, mais comme le contexte change, l'individu prioritaire à un moment donné, ne l'est plus forcément l'instant suivant, même si les conditions régulant l'accès n'ont pas changé. Nous verrons également qu'il peut être extrêmement intéressant de faire évoluer les conditions d'accès au cours de la session de télé-TP, pour respecter un certain rythme dans la pédagogie employée.

Une politique d'orchestration des accès est décrite dans notre travail comme un ensemble de règles, qui s'appuient sur les classes (spécification des concepts) décrites dans notre ontologie. Ces classes sont les concepts que nous identifions comme récurrents dans les politiques décrivant les modalités de partage d'instrument.

Pour réaliser des règles avec des classes issues d'une ontologie, l'emploi de SWRL¹⁶ est une des solutions possibles. En fait, comme SWRL est le langage qui combine OWL et RuleML¹⁷, il s'agit du choix le plus naturel par rapport à ce que nous avons déjà bâti.

15. nous parlons bien ici de l'ensemble des éléments réalisant les classes de l'ontologie : les *instances* des classes

16. Semantic Web Rule Language

17. Rule Markup Language

SWRL est supporté par un groupe de travail du W3C¹⁸. SWRL :

« allows users to write Horn-like rules expressed in terms of OWL concepts to reason about OWL individuals. The rules can be used to infer new knowledge from existing OWL knowledge bases » (O'Connor *et al.*, 2005a).

Les connaissances issues d'une inférence sont appelées « assertions ». Les deux types d'assertions possibles sont les suivantes :

- ajout d'instance de propriété,
- appartenance à une classe.

Le système orchestrant les accès utilise de telles assertions pour déterminer, à partir des éléments connus, qui sera l'opérateur du télé-TP, pour un instant donné et lorsque la personne en effectue la demande. Chacune de ces règles est composée d'un antécédent et d'une conséquence, qui agrègent un ensemble potentiellement vide d'atomes¹⁹.

Illustrons notre propos d'un exemple concret. Construisons la politique d'orchestration qui pourrait s'énoncer en langage naturel comme suit :

Si un utilisateur endosse le rôle d'administrateur pour le télé-TP, alors lui et lui seul est autorisé à prendre la main sur le dispositif distant (les autres regardent)

Nous ferons référence à cette politique par la suite sous l'appellation « admin only ». Une telle règle se représente aisément sous la forme d'un arbre. La figure III.2 représente cet arbre pour la politique « admin only » (Nous nous appuyons ici sur la représentation proposée par le tutoriel de RuleML²⁰).

Cette structure peut également être traduite en Datalog²¹ comme suit :

```
<Implies>
  <head>
    <Atom>
      <Rel>hasNewOperator</Rel>
      <Ind>RemoteLab</Ind>
      <Var>User</Var>
    </Atom>
  </head>
  <body>
```

18. <http://www.w3.org/Submission/SWRL/>

19. propositions insécables, d'où leur nom

20. <http://www.ruleml.org/papers/tutorial-ruleml-20050513.html>

21. Datalog est un langage bien connu dans le domaine de la programmation. Il fut créé pour interroger des bases, via la description logique des relations entre les différents types d'informations hébergés dans une base de données.

```

<And>
  <Atom>
    <Rel>hasRole</Rel>
    <Var>User</Var>
    <Ind>Administrate</Ind>
  </Atom>
  <Atom>
    <Rel>hasPendingOperatingRequest</Rel>
    <Var>User</Var>
  </Atom>
</And>
</body>
</Implies>

```

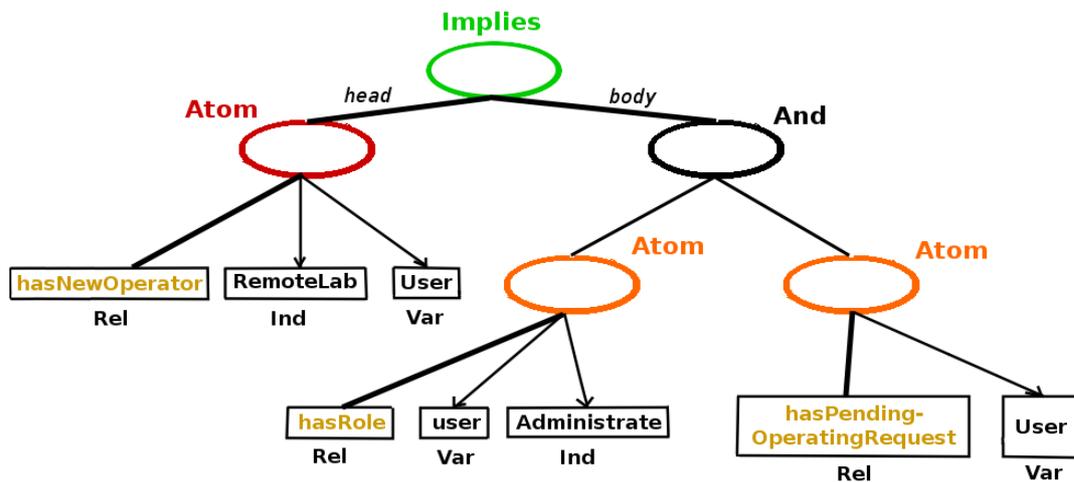


FIG. III.2 – L'arbre de la règle « admin only » qui donne la main à l'administrateur qui la demande.

III.2.2 Exprimer les politiques d'accès avec SWRL

Néanmoins, il est certain que l'expression précédente en Datalog ne suffit pas. Nous voulons en effet tirer profit de l'usage des ontologies. Puisque nous utilisons des concepts, et plus précisément les classes d'une ontologie qui représentent ces concepts, il est important de retenir un langage qui exploite les ontologies au format OWL. De par ce fait, SWRL est un bon choix, puisqu'il permet d'exprimer des règles en fonction des classes et des individus d'une ontologie au format OWL.

Cependant, il faut comprendre que SWRL suit la logique de Horn : il ne permet d'exprimer une règle seulement par une suite de conjonctions. Fort heureusement, nous allons

voir que cette approche est satisfaisante pour l'expression des règles courantes. Il n'en est pas toujours ainsi, et nous verrons que cet aspect de SWRL devient une limitation pour l'expression des règles d'orchestration des accès dans des télé-TPs (III.3.4)

Notre premier exemple (« admin only ») ne contenant que des conjonctions, nous pouvons aisément transcrire la règle Datalog précédente en SWRL comme suit :

$$\begin{aligned} \text{Def_hasOperator} : & \text{RemoteLab}(?y) \wedge \text{User}(?x) \wedge \text{hasRole}(?x, \text{Administrate}) \\ & \wedge \text{hasPendingOperatingRequest}(?x, ?r) \Rightarrow \text{hasNextOperator}(?y, ?x) \end{aligned} \quad (\text{III.1})$$

Cette expression implémentable en SWRL représente l'accès préemptif au dispositif pour un administrateur en faisant la demande.

Nous basons nos règles régissant les accès sur les qualifications des utilisateurs, en plus d'inférer la propriété « hasNextOperator », (c'est-à-dire l'utilisateur ayant obtenu l'accès au télé-TP après inférence). En conséquence, l'ensemble des règles précédentes (III.1), contenant seulement la règle déterminant le prochain opérateur, ne suffit pas. Il faut lui adjoindre les règles qualifiant les utilisateurs pour obtenir une expression complète d'une politique de partage (III.2)

$$\left\{ \begin{array}{l} \text{Def_isAdmin} : \text{User}(?x) \wedge \text{hasRole}(?x, \text{Administrate}) \Rightarrow \text{Admin}(?x) \\ \text{Def_hasOperator} : \text{RemoteLab}(?y) \wedge \text{Admin}(?x) \\ \wedge \text{hasPendingOperatingRequest}(?x, ?r) \Rightarrow \text{hasNextOperator}(?y, ?x) \end{array} \right. \quad (\text{III.2})$$

Sous cette forme, les politiques d'orchestration des accès peuvent être exprimées en SWRL, avec un fichier SWRL par orchestration. Par exemple, la règle « Def_hasOperator » dans la politique précédente (III.2) est consultable en ligne à l'adresse : <http://dev.istase.com/satin/rlab/policies/adminonly.swrl.owl>

III.2.3 Autres exemples de politiques d'accès

Nous avons décrit un exemple trivial d'orchestration des accès : celui où seul les administrateurs peuvent être opérateurs du télé-TPs. Malheureusement, cette situation ne satisfait pas toutes les situations d'apprentissage collaboratif : les apprenants font plus

que regarder, ils doivent pouvoir également manipuler et communiquer.

Dans le but de satisfaire ces autres situations, de nouvelles conventions de partage doivent être envisagées. Prenons l'exemple d'orchestration pouvant s'exprimer ainsi en langage naturel :

On donne le rôle d'opérateur à l'apprenant sur demande, si et seulement si le temps cumulé pendant lequel ils ont été opérateurs est plus petit que celui de l'opérateur courant. Cette restriction ne s'applique pas aux enseignants et administrateurs, lesquels jouissent constamment d'un accès préemptif sur tous les autres utilisateurs.

Il s'agit très certainement un exemple moins trivial que la politique « admin only ». Nous ferons référence à cette politique par la suite comme « student minus operating time, admins/teachers bypass ».

La première tâche à accomplir pour construire sa propre politique de partage en SWRL, est de transformer la politique exprimée en langage naturel comme un ensemble de règles, utilisant des conjonctions entre les concepts de notre ontologie. C'est ainsi que « student minus operating time, admins/teachers bypass » peut être résumé par l'ensemble de règles suivant :

$$\left\{ \begin{array}{l}
 \text{Def_hasAdminOperator} : \text{RemoteLab}(?y) \wedge \text{Admin}(?x) \\
 \wedge \text{hasPendingOperatingRequest}(?x,?r) \Rightarrow \text{hasNextOperator}(?y,?x) \\
 \\
 \text{Def_hasTeacherOperator} : \text{RemoteLab}(?y) \wedge \text{Teacher}(?x) \\
 \wedge \text{hasPendingOperatingRequest}(?x,?r) \Rightarrow \text{hasNextOperator}(?y,?x) \\
 \\
 \text{Def_hasStudentOperator} : \text{RemoteLab}(?rlab) \wedge \text{hasNextOperator}(?rlab,?s1) \\
 \wedge \text{Student}(?s2) \wedge \text{hasPendingOperatingRequest}(?s2,?r) \\
 \wedge \text{hasOperatingPeriod}(?s1,?op1) \wedge \text{hasOperatingPeriod}(?s2,?op2) \\
 \wedge \text{hasDuration}(?op1,?d1) \wedge \text{hasDuration}(?op2,?d2) \\
 \wedge \text{swrlb} : \text{lessThan}(?d2,?d1) \Rightarrow \text{hasNextOperator}(?rlab,?s2)
 \end{array} \right. \quad (\text{III.3})$$

De même, il est possible de consulter en ligne le fichier SWRL correspondant : <http://dev.istase.com/satin/rlab/policies/studentminusoptime.swrl.owl>

Toutes nos modalités d'accès au sein de télé-TPs ont été éditées avec le logiciel

Protégé (Knublauch, 2004), et plus précisément avec le module « SWRLTab »²² (O'Connor *et al.*, 2005a) qui permet d'éditer les règles préparées pour SWRL au sein de Protégé.

À titre d'illustration, les autres orchestrations que nous avons développées au format SWRL se trouvent en ligne : <http://dev.istase.com/satin/rllab/policies/>. L'idée n'est pas de fournir un ensemble complet de politiques d'accès qui satisferait toutes les situations d'apprentissage collaboratif. Nous voulons à l'inverse montrer qu'il est possible d'exprimer des orchestrations plutôt complexes de manière simple et concise, et de déployer ces politiques via SWRL. Il devient possible, de la même façon, que chacun construise sa propre stratégie de partage, dans le but de satisfaire ses besoins d'utilisation.

Par exemple, nous avons envisagé un autre cas de figure : le contrôle est donné à l'utilisateur si et seulement si c'est la première fois qu'il demande la main. Nous ferons référence par la suite à cette orchestration par « student first time, admins/teachers bypass ».

Pour bien comprendre que notre contribution ne se limite en aucun cas à peu de situations d'apprentissage collaboratif, nous donnons d'autres exemples de règles de partage dans le tableau III.4. Il convient de garder à l'esprit que le tableau n'est en aucun cas exhaustif, et qu'il s'appuie seulement sur le faible nombre de concepts hébergés dans notre ontologie des éléments dont dépend les accès collaboratifs au sein de télé-TPs.

Rien n'empêche de construire des règles plus complexes, ou encore des politiques qui combinent plusieurs politiques présentées dans le tableau III.4. Par exemple, la dernière présentée (« round robin ») pourrait trier les utilisateurs à partir de leur niveau de compétences, plutôt que par ordre alphabétique. Encore une fois, nous ne pensons pas qu'une politique soit meilleure qu'une autre, mais que celle à employer dépend de l'expérience d'apprentissage collaboratif que l'on veut offrir aux apprenants en regard de la pédagogie employée.

Il est également intéressant de noter que, même si l'ontologie sur laquelle se basent les politiques d'orchestration est volontairement concise, le nombre de politiques pouvant s'exprimer à partir de celle-ci est déjà grand. Moyennant d'éventuelles extensions (ajouts de classes décrivant des concepts), l'ontologie des éléments de la collaboration dans un télé-TP peut mener à un très grand nombre d'expressions, soit autant de politiques de partage utilisables.

Un autre point remarquable, réside dans le fait que les concepts employés par les règles SWRL ne sont pas tellement spécifiques au domaine des télé-TPs. Ce sont des

22. <http://protege.cim3.net/cgi-bin/wiki.pl?SWRLTab>

Alias	Description
Admin only	Seuls les administrateurs ont accès au rôle opérateur.
Student minus operating time	Quand un étudiant demande la main, il devient opérateur si et seulement si il a été moins souvent opérateur que l'opérateur courant (en temps cumulé).
Student first time	Quand un étudiant demande la main, il devient opérateur si et seulement si c'est la première fois qu'il la demande.
Best students	Quand un étudiant demande la main, il devient opérateur si et seulement si son niveau de compétences reconnu est plus haut que l'opérateur courant.
Worst students	Quand un étudiant demande la main, il devient opérateur si et seulement si son niveau de compétences reconnu est plus faible que l'opérateur courant.
First connected	Quand un étudiant demande la main, il devient opérateur si et seulement si il s'est connecté avant l'opérateur courant.
The more you had it, the more you will have it.	Quand un étudiant demande la main, il devient opérateur si et seulement s'il a été plus souvent opérateur que l'opérateur courant (non pas en temps cumulé, mais en nombre de demandes acceptées pour le rôle opérateur).
The less you had it, the more you will have it	Quand un étudiant demande la main, il devient opérateur si et seulement si il a été moins souvent opérateur que l'opérateur courant (non pas en temps cumulé, mais en nombre de demandes acceptées pour le rôle opérateur).
Exclude heavy disconnected users	Quand un étudiant demande la main, il devient opérateur si et seulement si il a déconnecté de la session plus de 5 fois pour la session de télé-TP en cours.
Balance from ratio	Quand un étudiant demande la main, il devient opérateur si et seulement si le rapport $r = \text{OperatingPeriod}/\text{WaitingPeriod}$ est plus faible que le même rapport pour l'opérateur en cours.
You must have waited	Quand un étudiant demande la main, il devient opérateur si et seulement si il est entré dans la session de télé-TP il y a plus de 5 minutes.
Alphabetical round robin	Les utilisateurs bénéficient d'un accès exclusif de quinze minutes. Durant ces quinze minutes, il ne peuvent pas être détrônés par d'autres utilisateurs (sauf par des enseignants et des administrateurs qui jouissent d'un accès préemptif). Les utilisateurs prennent leur tour dans l'ordre alphabétique de leur identifiant.

TAB. III.4 – Politiques de partage : exemples d'autres orchestrations possibles utilisant notre modèle (tableau non exhaustif).

concepts plutôt propres aux systèmes distribués (connexions, utilisateurs) dans lesquels l'intégrité d'une ressource partagée se détermine par un accès concurrentiel (OperatingRequest, PendingOperatingRequest, ...). Cette observation nous porte à croire que notre façon de modéliser les accès collaboratifs pourrait tout à fait être reproduite dans le domaine du Travail Collaboratif Assisté par Ordinateur.

Il est vrai que quelle que soit la ressource (un tableur, un emploi du temps, un tableau blanc, ...), le même genre de politique de partage est nécessaire pour assurer le travail collaboratif de manière générale, car il est lui aussi rythmé par différentes phases d'utilisation, nécessitant autant de formes de collaboration entre utilisateurs. Nous pensons alors que pourrait naître de nos travaux une ontologie, non plus de domaine, mais une ontologie de haut niveau, formalisant la connaissance que nous avons aujourd'hui du travail collaboratif. Une telle ontologie, associée à nos approches mariant SWRL et OWL pour satisfaire l'orchestration de la collaboration, permettrait de fournir une plate-forme gérant les conflits d'accès à une ressource, indépendamment de la ressource contrôlée. De ce fait, l'écriture de nouvelles conditions de partage n'impliquerait que l'écriture d'un fichier SWRL (et non le développement d'une nouvelle application métier). Cette approche « bottom-up » pourrait permettre la mise à disposition de plus d'outils collaboratifs synchrones, dotés d'un degré plus fort de collaboration que le simple verrouillage d'une ressource. Une telle idée pourrait tout à fait être un axe de développement futur.

III.2.4 Bilan

Jusqu'ici nous avons démontré qu'il était possible de formaliser les politiques de partage d'une ressource dans un système distribué, et plus spécifiquement dans le cas d'un télé-TP. Dans la section suivante, nous nous attacherons à développer deux axes de l'implémentation. Tout d'abord nous présenterons la plate-forme implémentée permettant de confronter la réalité de l'implémentation à la théorie de formalisation que nous avons développé précédemment. Ensuite, une utilisation originale de SWRL sera introduite : la possibilité de basculer d'un ensemble de règles à un autre, sans remettre en question la connaissance capturée jusque là.

III.3 Implémentation du système d'orchestration

L'objectif de cette section est de montrer comment mettre en pratique le modèle présenté précédemment. En effet, exploiter des ontologies demande la mise en place d'un Système à Base de Connaissance (SBC). C'est ce SBC qui est chargé de maintenir l'intégrité des individus contenus dans l'ontologie, et d'intégrer les nouvelles connaissances produites par son moteur d'inférence sous la forme d'assertions.

III.3.1 Architecture complète

La façon dont un SBC acquiert de nouvelles connaissances demeure une des clés de sa performance. Nous voyons principalement deux possibilités :

- la connaissance est transférée par une source externe vers le système qui la capte,
- le système infère de nouvelles connaissances à partir de celles qu'il détient.

Dans notre domaine, nous avons besoin de renseigner le SBC des faits ayant lieu durant la télé-opération. La section précédente expose que cette connaissance est décrite par le temps d'attente cumulé par utilisateur, des horodatages divers (heure de connexion, déconnexion)... Par la suite, le système infère, au regard de la politique d'orchestration en cours, si l'utilisateur qui demande la main se la voit accordée ou non.

Au départ, Protégé nous permet de créer le fichier OWL présenté précédemment (figure III.1). Protégé nous permet également de créer les politiques d'accès se basant sur notre ontologie, via le module SWRLTab. Par la suite, ce fichier SWRL est téléchargé par notre unité de collaboration, qui a la charge de répartir plusieurs utilisateurs qui réclament la main au même moment. Une vue complète du SBC implémenté s'appuyant sur des briques logicielles du Web sémantique répandues est reproduite à la figure III.3.

Les mécanismes de collaboration et la base de connaissances associée sont hébergés par l'unité logique de collaboration. Il s'agit d'une application Java qui :

- permet de fournir les connexions entre les sessions de télé-TPs et les politiques d'orchestrations des accès,
- embarque toutes les connaissances relatives à l'état des sessions de télé-TPs en cours,
- met à jour la base de connaissances à chaque nouvelle action sur un télé-TP,
- détermine, suivant la politique de partage en place, si un utilisateur demandant la main sur le laboratoire distant est exaucé ou non.

Dans notre approche, le SBC implémenté est commun à toutes les sessions de télé-TPs, mais il existe autant d'instances d'unités logiques de modération des accès qu'il existe de

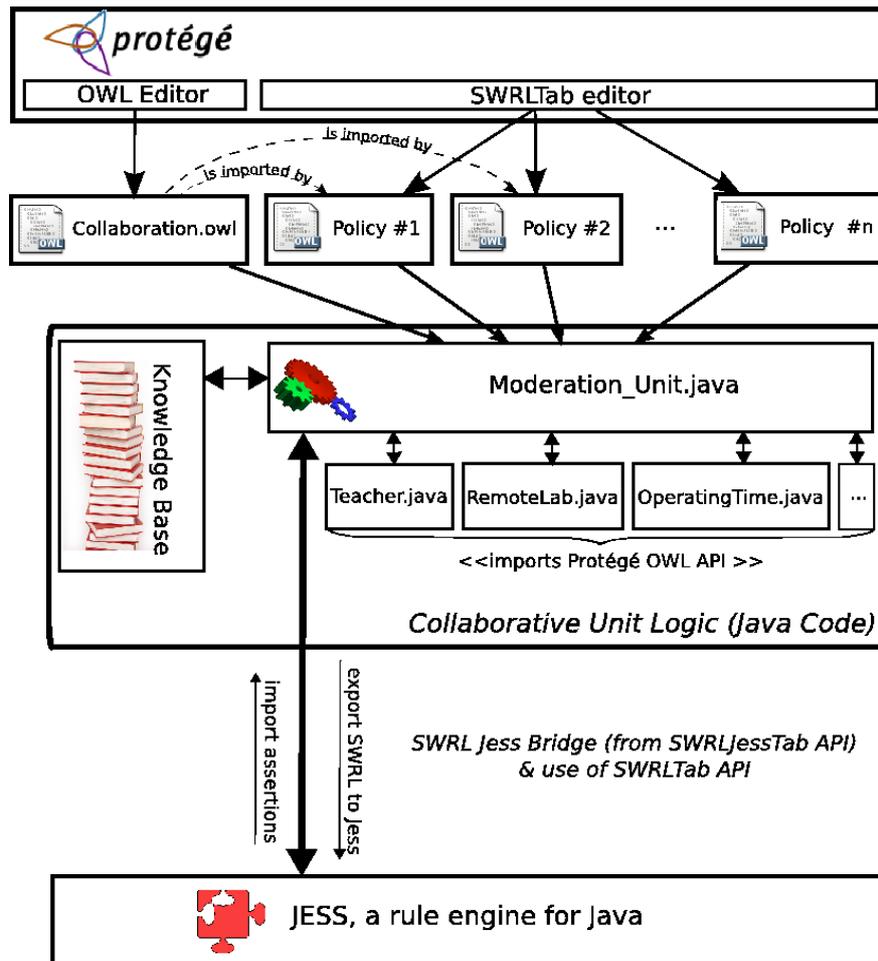


FIG. III.3 – Implémentation du SBC orchestrant les accès dans les séances de télé-TP collaboratifs.

sessions d'un télé-TP. Le SBC télécharge, à partir de son URL²³, une politique d'accès à mettre en place pour la session considérée. Le SBC met à jour sa base de connaissances en accord avec les actions menées durant la séance de télé-TP. Lorsqu'un utilisateur demande la main, la requête est transmise à l'unité logique de collaboration. Celle-ci lance une inférence qui détermine, à partir de la base de connaissances et des règles d'accès en vigueur, si l'utilisateur se voit attribuer la main ou non sur le télé-TP. Quel qu'en soit le résultat, la base de connaissances est mise à jour (nombre de demandes pour obtenir le rôle d'opérateur, nombre de fois que cette demande a été acceptée/refusée (le cas échéant), les temps cumulés, ...)

L'algorithme prépondérant de l'unité logique de collaboration correspond au traitement déclenché lorsqu'un utilisateur demande le rôle d'opérateur sur le télé-TP. Cet al-

23. Uniform Resource Locator

gorithme est le suivant :

procédure askHand (**E** login : chaîne de caractère ; **E** remoteLab : RemoteLab)

Déclaration date : Date; user : User; pOpReq : PendingOperatingRequest; accept : booléen

début

date ← now();

***Remarque** : Mise à jour de la base de connaissances (temps d'attente et temps cumulé pour l'opérateur en cours)*

self.updateKB();

***Remarque** : Positionne l'utilisateur comme demandant la main dans la base de connaissances*

user ← self.getUser(login);

pOpReq ← createPendingOperatingRequest(date);

u.setHasPendingOperatingRequest(pOpReq);

***Remarque** : S'il y a déjà un opérateur, on le met en concurrence avec l'utilisateur demandant la main*

si remoteLab.hasHasOperator() **alors**

***Remarque** : Exporte la base de connaissances OWL+SWRL et lance l'inférence*

exportKbToRuleEngineAndInfer();

***Remarque** : on statue suivant le résultat de l'inférence*

si remoteLab.getHasOperator() == login **alors**

accept ← vrai;

finsi

sinon

***Remarque** : sinon on applique les politiques à défaut de la politique de partage*

accept ← reasoningFromInitPolicies(login);

finsi

***Remarque** : Mise à jour post-traitement de la base de connaissances (Nombre d'acceptations ou de rejets pour l'utilisateur ayant demandé la main)*

postAskHand(login,accept);

retourner accept

fin

Le lecteur peut voir une différence dans la façon de prendre une décision suivant si le télé-TP a déjà un opérateur ou non. Lorsqu'un utilisateur demande la main sur un télé-TP sans opérateur, il ne doit pas forcément l'avoir à *défaut*. Par exemple, on ne veut

peut-être pas que des étudiants (même si personne n'a la main) se voient confier un accès à un télé-TP dès qu'ils en font la demande. Il appartient à la politique d'accès de définir la décision à prendre dans une telle configuration. Plutôt que d'ajouter des règles pour traiter ce cas spécial, des propriétés de classe « RemoteLab » permettent de statuer. Ce sont des individus de l'ontologie qui servent comme des variables d'initialisation pour une décision dans les cas à défaut (aucun opérateur à déloger).

Notre but ultime est de donner la possibilité à chacun d'exprimer sa propre politique de partage, de la manière la plus simple et compréhensible possible.

Le code Java correspondant à cet algorithme est présent dans un système de développement collaboratif en ligne (le logiciel que nous utilisons est « subversion »²⁴). Les sources sont consultables à l'adresse suivante :

<http://ist-bessat.univ-st-etienne.fr/svn/collab/ModerationUnit>²⁵

III.3.2 De l'importance de pouvoir modifier la politique en vigueur

Il apparaît malheureusement qu'une seule politique d'accès ne peut pas être en vigueur pour la totalité de la même session de télé-TP : il existe forcément un rythme dans l'apprentissage, qui fait que les besoins d'orchestration des accès varient au cours d'un même télé-TP. Prenons l'exemple d'une séance où l'on montre dans un premier temps aux apprenants comment se servir d'un appareil, et ensuite on les laisse manipuler. Il est évident qu'une unique politique ne peut modéliser ce besoin de collaboration. On aura plutôt tendance à réaliser une politique pour le premier type de partage (très ressemblant à « admin only »), et ensuite à mettre en vigueur, à la place de la précédente politique, une nouvelle orchestration autorisant les apprenants à téléopérer.

Cet exemple simple illustre la nécessité de basculer d'une politique à l'autre, en cours de télé-TP. C'est à l'unité logique de collaborations présentée précédemment (figure III.3) qu'il appartient de proposer la méthode de basculement d'une politique de partage à une autre.

Au niveau applicatif, cela se traduit par la reconstruction de l'instance de la classe

24. <http://subversion.tigris.org/>

25. L'accès est pour le moment restreint comme notre proposition est en cours de soumission auprès d'un journal scientifique international. Pour consulter le code il vous faudra entrer le nom d'utilisateur « reviewer » et le mot de passe « dfg340 ». Les sources seront publiques une fois que notre publication sera effective.

« OWLModel » (de l'API SWRLTab incluse dans Protégé). L'implémentation a néanmoins montré que changer les règles SWRL tout en conservant la base de connaissances acquise jusque-là n'avait jamais été envisagé. En effet, charger un nouveau fichier SWRL à partir d'une URL impliquait la destruction des individus dans la base de connaissances par l'API (nouvelle instanciation d'un objet de type « OWLModel », plutôt qu'une mise à jour des individus SWRL pour l'instance en cours). Par conséquent, une première implémentation a montré que toutes les connaissances acquises jusqu'ici étaient perdues : le modèle en mémoire était effectivement rechargé, c'est ce qui conduisait à la perte de tous les individus précédemment créés depuis le chargement initial. La conséquence directe était que toutes les actions (horodatage de connexion, temps d'activité, d'attente, ...) étaient effacées de la base de connaissances !

Nous avons travaillé avec le développeur de l'API SWRL, de l'université de Stanford. Cette relation a abouti à la création d'une nouvelle méthode dans l'API de Protégé, pour permettre le basculement d'un ensemble de règles au format SWRL vers un autre ensemble (différent) sans aucune perte de la connaissance acquise jusque-là.

La méthode Java ajoutée porte la signature suivante :

```
public static void replaceImps(edu.stanford.smi.protege.owl.model.OWLModel);
```

(paquetage edu.stanford.smi.protege.owl.swrl.model.SWRLFactory.java package de l'API Protégé)

Plus d'explications sont accessible en ligne à l'adresse :

<http://protege.cim3.net/cgi-bin/wiki.pl?SWRLFactoryFAQ#nid8GX>

Nous pensons que cette modification est primordiale pour les SBC, puisqu'elle permet de pouvoir dorénavant adapter les règles régissant les inférences à une situation donnée. Nous pouvons désormais choisir les règles que doit suivre une inférence, tout en sachant que l'inférence suivante pourra en suivre d'autres, mais que cela ne remettra pas en cause la connaissance acquise jusque là. Même si cela ne sert pas tous les types de SBC, cela pourrait se révéler crucial pour des systèmes nécessitant d'adapter leurs règles d'une inférence à l'autre. Nous voyons cela comme une solution élégante et rapide pour la mise en place d'une reconfiguration d'applications au regard du contexte. Il est en effet possible, à l'image de notre plate-forme, de commander aux applications de modifier leur inférence suivant les éléments captés à partir du contexte.

Pour notre domaine des télé-TPs, prenons l'exemple de la séance de télé-TP suivant :

- une première période permet à l'enseignant de montrer aux étudiants ce qu'ils peuvent faire. La politique de partage en vigueur est alors « admin only »
- Ensuite, les étudiants sont invités à procéder à des tests par eux-même. Comme l'enseignant veut voir tout le monde tester au moins une fois l'appareil à distance, il favorise les arrivées tardives pour s'assurer que tout le monde participe. La politique en vigueur devient alors « student first time, admins/teachers bypass » (En remplacement de « admin only »).
- Enfin, l'enseignant assigne aux apprenants les exercices à réaliser pour leur session de télé-TP. Comme chaque étudiant aura le même exercice que les autres, tous vont utiliser le télé-TP de manière collaborative. Dans l'optique de limiter les inégalités, l'enseignant va mettre en place une nouvelle politique de partage favorisant les personnes ayant eu la main le moins de temps (« student minus operating time, admins/teachers bypass »).

Dans une telle session, le basculement d'un fichier SWRL à l'autre (et donc d'une politique de partage à l'autre) est clairement nécessaire pour proposer une modération des accès adaptée à l'objectif pédagogique du moment (figure III.4).

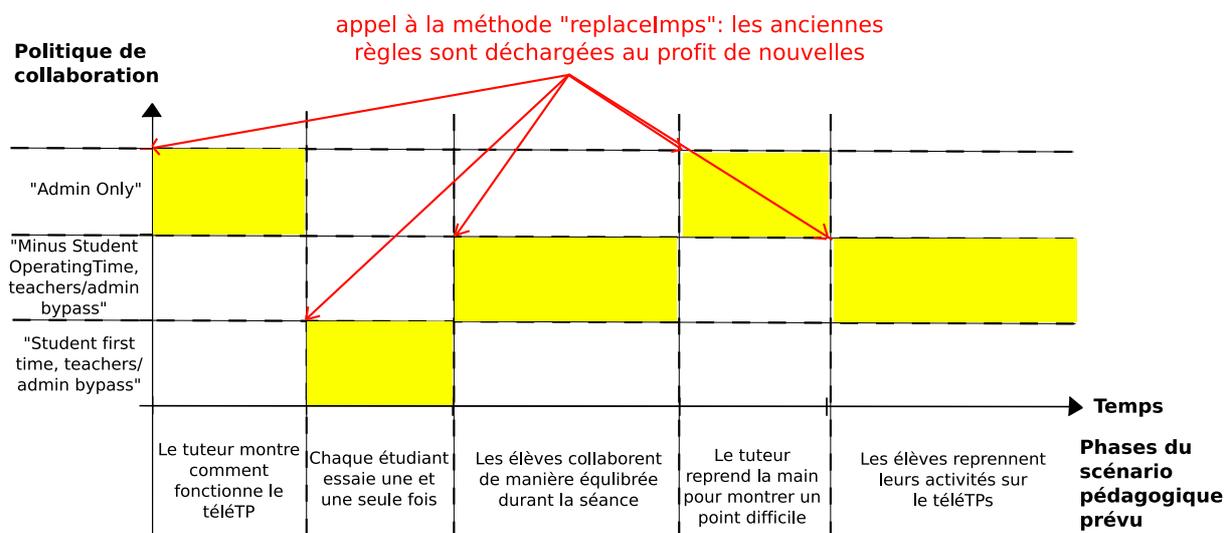


FIG. III.4 – La politique d'orchestration peut être modifiée au cours du télé-TP pour satisfaire l'objectif pédagogique du moment.

Le scénario, utilisé ici à titre d'illustration, a été développé avec le langage Java et la bibliothèque de Protégé permettant de manipuler des bases de connaissances au format SWRL. Il est possible de consulter le code source correspondant à la classe :

```
fr.istase.satin.einst.collabenv.demo.DemoModeration1.java
```

dans notre gestionnaire de développement collaboratif dont le lien a été donné précédemment.

III.3.3 Couplage de l'unité de modération avec la plate-forme de télé-TP

Il est nécessaire de positionner le SBC par rapport à notre plate-forme collaborative et générique de télé-TPs présentée au chapitre précédent. En fait, le SBC s'abonne à la messagerie applicative comme un participant du télé-TP et l'on introduit dans le protocole de communication propre à notre intergiciel asynchrone (présenté en II.3.2) les trois nouveaux performatifs :

- ASK_OP, pour demander une inférence qui déterminera si l'utilisateur (dont l'identifiant est passé en paramètre) demandant le rôle d'opérateur est exaucé ou non,
- ANS_OP_OK, pour signifier que l'utilisateur est devenu opérateur,
- ANS_OP_NOK, pour signifier que l'utilisateur s'est vu refuser le rôle d'opérateur.

Le couplage asynchrone entre notre plate-forme de télé-TP et notre SBC se représente facilement (figure III.5) puisque le SBC est perçu comme un abonné à la messagerie applicative.

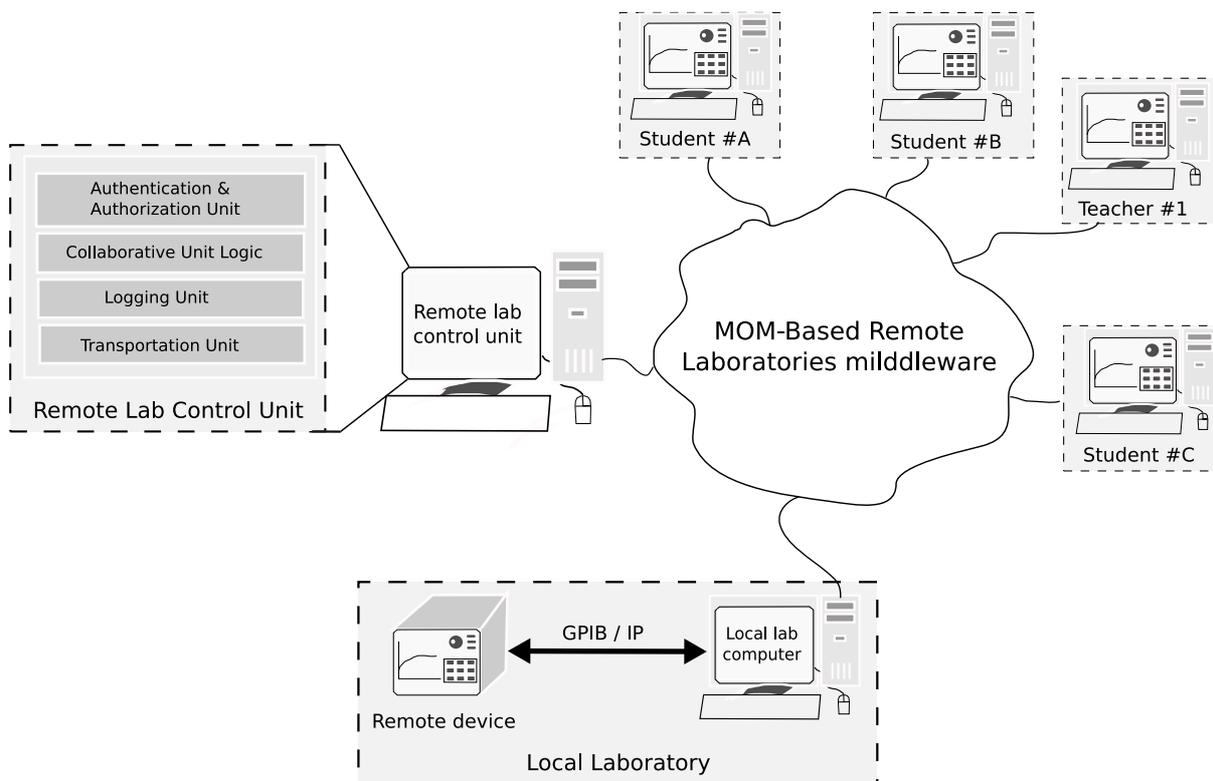


FIG. III.5 – Couplage asynchrone entre notre SBC et la plate-forme de télé-TP.

SWRL demeure une solution puissante, puisque nous pouvons exprimer la majorité des politiques d'accès de notre domaine (voir un extrait de ces règles au tableau III.4). Il existe cependant certaines politiques de partage que nous ne pouvons pas formaliser à

cause de certaines limitations actuelles de SWRL. Ces lacunes feront l'objet de l'exposé de la section suivante.

III.3.4 Problèmes d'implémentation rencontrés

Le manque d'implémentations natives pour des inférences basées SWRL

Le marché des langages sémantiques de règles est encore immature puisqu'il commence à peine à percer le monde de l'industrie, et que de nombreux problèmes de recherche demeurent à résoudre.

Nous utilisons JESS²⁶ comme moteur d'inférence pour manipuler notre base de connaissances au format OWL et nos règles sont écrites en SWRL. Cette volonté résulte d'un choix par défaut : fin 2006, JESS était le seul moteur d'inférence sachant traiter le SWRL de manière suffisamment juste pour envisager une phase d'implémentation.

Cependant, JESS ne traite pas le format SWRL de manière native. JESS se contente de traduire les règles SWRL dans son langage propriétaire de règles. Il faut donc utiliser un « pont » entre SWRL et JESS pour que la communication entre ces deux unités du SBC soit réalisable²⁷.

Depuis, un logiciel très connu dans le monde du web sémantique, RacerPro²⁸, a intégré la prise en charge de SWRL. Cependant, il s'agit du même constat que pour JESS : Racer convertit le SWRL que le développeur lui fournit en nRQL²⁹, le langage propriétaire de requêtes de Racer.

Paradoxalement, alors que SWRL tend à devenir un standard du Web sémantique, aucune implémentation de moteur d'inférences, à notre connaissance, ne présente de support natif de SWRL.

SWRL ne peut exprimer que des règles sous forme normale disjunctive

Martin O'Connor introduit ainsi SWRL :

« In common with many other rule languages, SWRL rules are written as antecedent-consequent pairs. In SWRL terminology, the antecedent is referred to as the rule body and the consequent is referred to as the head. The head

26. de Sandia National Laboratories, <http://herzberg.ca.sandia.gov/jess/>.

27. Nous utilisons celui livré avec Protégé, SWRLJessTAB, <http://protege.cim3.net/cgi-bin/wiki.pl?SWRLJessTab>

28. de Racer Systems, <http://www.racer-systems.com>

29. new Racer Query Language

and body consist of a conjunction of one or more atoms. At present, SWRL does not support more complex logical combinations of atoms. » (O'Connor *et al.*, 2005b)

SWRL est donc très souvent présenté, à juste titre, comme un langage suivant la logique de Horn, puisque les règles sont exprimées par des conjonctions (notées \wedge ou quelque fois « et »). SWRL exclut donc d'autres opérateurs potentiellement très pratiques d'un point de vue sémantique, comme par exemple les opérateurs « ou » (\vee) et « ou exclusif » ($\vee\vee$).

Pour notre domaine, cela signifie exprimer toutes les politiques d'accès par un ensemble de règles sous une forme normale disjonctive (FND)³⁰. Par exemple, essayons de formaliser la politique de partage qui pourrait s'énoncer ainsi en langage naturel :

Les étudiants qui n'ont jamais eu le statut d'opérateur ou qui attendent depuis plus de 10 minutes bénéficient d'un accès préemptif, les administrateurs et les enseignants ont quant à eux toujours un accès inconditionnel.

Cela se transpose naturellement comme suit :

$$\left\{ \begin{array}{l} \text{Def_hasAdminOperator} : \text{RemoteLab}(?y) \wedge \text{Admin}(?x) \\ \wedge \text{hasPendingOperatingRequest}(?x,?r) \Rightarrow \text{hasNextOperator}(?y,?x) \\ \\ \text{Def_hasTeacherOperator} : \text{RemoteLab}(?y) \wedge \text{Teacher}(?x) \\ \wedge \text{hasPendingOperatingRequest}(?x,?r) \Rightarrow \text{hasNextOperator}(?y,?x) \\ \\ \text{Def_hasStudentOperator} : \text{RemoteLab}(?rlab) \wedge \text{Student}(?s1) \wedge \\ ((\text{hasOperatingPeriod}(?s1,?req) \wedge \text{hasDuration}(?op1,?d1) \wedge \text{swrlb} : \text{equal}(?d1,0)) \\ \vee (\text{hasWaitingOperatingPeriod}(?s1,?wait1) \wedge \text{hasDuration}(?wait1,?d2) \\ \wedge \text{swrlb} : \text{greaterThan}(?d2,600))) \Rightarrow \text{hasNextOperator}(?rlab,?s1) \end{array} \right. \quad (\text{III.4})$$

Comme SWRL ne sait pas formaliser la disjonction, il faut exploser la règle « Def_has-StudentOperator » en deux règles disjointes. On obtient alors très logiquement le système suivant :

30. c'est-à-dire une suite de disjonctions de conjonctions

$$\left\{ \begin{array}{l}
\text{Def_hasAdminOperator} : \text{RemoteLab}(?y) \wedge \text{Admin}(?x) \\
\wedge \text{hasPendingOperatingRequest}(?x,?r) \Rightarrow \text{hasNextOperator}(?y,?x) \\
\\
\text{Def_hasTeacherOperator} : \text{RemoteLab}(?y) \wedge \text{Teacher}(?x) \\
\wedge \text{hasPendingOperatingRequest}(?x,?r) \Rightarrow \text{hasNextOperator}(?y,?x) \\
\\
\text{Def_hasStudentOperatorA} : \text{RemoteLab}(?rlab) \wedge \text{Student}(?s1) \wedge \text{hasOperatingPeriod}(?s1,?req) \wedge \text{hasDuration}(?op1,?d1) \wedge \text{swrlb} : \text{equal}(?d1,0) \\
\Rightarrow \text{hasNextOperator}(?rlab,?s1) \\
\\
\text{Def_hasStudentOperatorB} : \text{RemoteLab}(?rlab) \wedge \text{Student}(?s1) \wedge \text{hasWaitingOperatingPeriod}(?s1,?wait1) \wedge \text{hasDuration}(?wait1,?d2) \wedge \text{swrlb} : \text{greaterThan}(?d2,600) \\
\Rightarrow \text{hasNextOperator}(?rlab,?s1)
\end{array} \right. \quad (\text{III.5})$$

Il s'agit certes d'une limitation qui n'empêche pas l'implémentation de la politique sous la forme d'un fichier SWRL. Néanmoins, sans parler des risques d'erreurs lors de la transformation et de l'exhaustivité de la tâche, cela handicape grandement la lisibilité d'une politique de partage.

Être et ne pas être

De la même façon, SWRL n'offre pas la possibilité d'exprimer la négation, généralement notée \neg (quelquesfois \sim). Ce peut être une limitation plus importante que la précédente, car elle peut non seulement nuire à la lisibilité d'un fichier SWRL mais également prohiber l'écriture d'une politique de partage sans apporter des modifications sur la structure de la base de connaissances sur laquelle s'appuient les règles. Cela signifie que SWRL peut forcer le remaniement de l'ontologie de départ.

Par exemple, examinons la politique de partage que l'on énoncerait ainsi en langage naturel :

Si un utilisateur n'est pas un étudiant, il ne peut pas être un opérateur.

À l'aide de la négation, on formalise alors cette politique d'orchestration comme suit :

$$\text{RemoteLab}(?rlab) \wedge \text{User}(?u) \wedge \neg(\text{Student}(?u)) \Rightarrow \text{hasNextOperator}(?rlab,?u) \quad (\text{III.6})$$

Il convient alors de transformer cette expression en une FND, excluant toute négation, pour pouvoir être implémentée en SWRL. C'est pourtant difficile puisque l'opérateur de

négarion (que l'on pourrait écrire quand bien même il n'existe pas encore) aurait besoin de charger toute la base de connaissances pour procéder à l'opération de négation. En fait, nous ne sommes pas en présence d'un simple opérateur arithmétique (ajouter deux dates, soustraire des valeurs, comparer des chaînes, ...) du même esprit que ceux qui existent déjà en SWRL (et dont d'autres restent à développer). Il s'agit d'un opérateur logique car l'évaluation de l'appartenance à une classe pour un individu n'est pas triviale pour l'ordinateur : elle implique de connaître et comprendre la sémantique structurant la base de connaissance dans les sous-couches applicatives. Nous pensons par ailleurs que les opérations menées sur les clauses d'une règle sémantique doivent rester au niveau des relations entre les règles pour demeurer visibles, et non pas être enfouies dans un opérateur. Il est cependant vrai que ce problème est partagé entre tous les langages de règles, et non spécifique à SWRL.

SWRL ne permet pas la révocation de connaissances

Un moteur d'inférence produit généralement deux types d'assertions : ajout d'une valeur pour une propriété donnée ou appartenance à une classe pour un individu. À propos des assertions de propriétés, ces assertions positionnent une valeur pour une propriété qui était préalablement vide. Il s'agit d'une inférence qui permet de déduire la valeur de cette propriété.

Dans le cas où la propriété n'est pas nulle, la nouvelle assertion vient s'ajouter à la première. Par conséquence, comme la propriété a deux valeurs là où il ne devrait y en avoir qu'une au maximum, n'importe quel raisonneur déclarerait l'inconsistance de la base de connaissances.

Or, nous avons montré un usage original d'un moteur d'inférence impliquant une modification à la volée les règles de l'inférence. Cela a pour conséquence directe d'ajouter une valeur pour la propriété « `hasNextOperator` » à chaque fois qu'un utilisateur se voit donner le rôle d'opérateur, alors que l'ancienne valeur n'est pas supprimée. En d'autres termes, il n'est pas possible d'indiquer clairement à SWRL que les assertions issues d'une règle peuvent impliquer la révocation (ou non) de connaissances déjà acquises.

De par le manque d'un mécanisme de révocation, un nouveau prédicat³¹ est ajouté à chaque fois qu'un utilisateur délègue un autre du rôle d'opérateur. Au final, la propriété fonctionnelle « `hasOperator` » possède deux valeurs distinctes pour la même propriété fonctionnelle. Cela implique intrinsèquement l'inconsistance de l'ontologie.

31. il s'agit de : `hasOperator(?rlab,?user_login)`

Pour remédier à cela, nous avons mis en place le palliatif suivant : la règle ne met pas à jour la propriété « `hasOperator` », mais « `hasNextOperator` », toujours pour le domaine de la classe « `RemoteLab` » et de portée la classe « `User` ». L'algorithme de l'unité logique de collaboration est alors responsable de :

- la révocation de l'ancienne valeur de la propriété « `hasNextOperator` »,
- cloner la valeur de la propriété « `hasNextOperator` » dans « `hasOperator` »,
- supprimer la valeur de la propriété tampon « `hasNextOperator` ».

Ce n'est évidemment pas une solution élégante ou satisfaisant les critères de passage à l'échelle que de dupliquer une classe à chaque fois que des assertions suivant une inférence sont susceptibles de mener à la révocation de l'ancienne connaissance d'une propriété. C'est par contre la seule option qui n'entraîne pas une remise en cause profonde des fondations de SWRL et de ses implémentations.

Tests d'existence : OWL omet les valeurs positionnées à « null » en RDF

Un autre problème rencontré réside dans le fait que SWRL ne permet pas de tester la nullité d'une propriété. RDF accepte la valeur « null » pour une propriété dont l'instance de la valeur associée n'est pas encore connue. Cependant, à l'inverse de RDF sur lequel il se base, le format OWL ne prévoit pas cette possibilité. OWL propose une vue de l'esprit différente : si une valeur est absente, c'est qu'elle n'est pas connue, il n'y a nul besoin de le préciser explicitement.

La conséquence directe est qu'une valeur nulle n'est pas visible dans la base de connaissances (elle est seulement déduite de par son absence). Aussi, une règle comme :

« Si un télé-TP n'a pas d'opérateur et que quelqu'un demande la main, il est toujours exaucé »

pourrait se traduire ainsi :

$$\begin{aligned}
 &RemoteLab(?rlab) \wedge hasOperator(?rlab,?dummy) \wedge isNull(?dummy) \wedge User(?u) \\
 &\wedge hasPendingOperatingRequest(?u) \Rightarrow hasNextOperator(?rlab,?u) \quad (III.7)
 \end{aligned}$$

Cela ne peut pas s'écrire en SWRL. En réalité, il est seulement possible d'écrire son propre opérateur « `isNull` », lequel irait explorer la base de connaissances au niveau RDF (donc descendre dans une sous couche de l'applicatif) pour évaluer la valeur de l'objet considéré. Il faudrait alors bien sûr réécrire un opérateur pour chaque type d'objet à tester, puisque les signatures des opérateurs sont fortement typées en SWRL.

III.3.5 Bilan

Cette section a présenté l'implémentation de notre système à base de connaissance, pour l'orchestration de sessions d'apprentissage collaboratif. Nous avons exposé comment nous parvenons à formaliser les éléments qui déterminent une politique de partage d'une ressource pour notre architecture distribuée de télé-TPs. Ces politiques, formalisées par des règles, sont embarquées dans des fichiers au format SWRL. Ces fichiers sont alors interprétés par un moteur d'inférences. Celui-ci utilise les règles ainsi formalisées pour déterminer si l'utilisateur qui en fait la demande se voit ou non attribuer le rôle d'opérateur. Cela peut éventuellement se faire au détriment de l'utilisateur qui était jusqu'alors l'opérateur du télé-TP.

Par ailleurs, nous avons montré l'importance cruciale de basculer d'un fichier SWRL à l'autre (donc d'une politique à l'autre) : il s'agit de donner les moyens au concepteur d'un télé-TP de rythmer la séance en regard des objectifs pédagogiques de chaque phase d'apprentissage. Il n'est pas simple dans une première approche d'implémenter un tel mécanisme puisque l'API Protégé ne le permettait pas. Des efforts ont donc été fournis pour arriver à une implémentation élégante, satisfaisant les objectifs de départ. Enfin, nous avons également pointé certaines limites que nous avons rencontrées dans l'emploi de SWRL.

Orchestrer l'accès au dispositif

Conclusion

Lorsque plusieurs apprenants sont connectés sur le même télé-TP, ils ne font face qu'à un seul instrument. Il faut donc établir des règles de partage. Certes, les accès sont quelques fois simplement concurrentiels dans le cadre d'une stricte émulation. Néanmoins, si la pédagogie est tournée en faveur de l'approche constructiviste, nous savons que les apprenants ont besoin de dialoguer entre eux et de manipuler ensemble pour confronter leurs résultats ou surmonter un problème commun.

La collaboration revêt plusieurs formes, selon la phase d'apprentissage dans laquelle se trouvent les apprenants. Quelques fois il s'agit de fournir un accès à l'instrument équitable en temps d'accès, d'autres fois il s'agit de privilégier les apprenants les plus en retard. On peut par ailleurs imaginer de privilégier les apprenants les plus en avance, dans le but de libérer la place à court terme pour les personnes rencontrant le plus de difficultés.

Quoiqu'il en soit, il est difficile de lister toutes les configurations possibles. Il est par contre plus aisé d'identifier les éléments qui composent ces politiques de régulation des accès dans les télé-TPs collaboratifs.

Par conséquent, nous avons choisi de formaliser les éléments entrant en jeu dans la collaboration entre apprenants. Les spécifications des concepts correspondants sont hébergés dans une ontologie au format OWL. Le langage SWRL nous permet alors de construire des règles, s'appuyant sur ces concepts, pour définir une politique de partage. Bien entendu, comme les phases d'apprentissage évoluent, il est possible de basculer d'une politique d'orchestration à l'autre sans remettre en cause la base de connaissances construite jusque-là.

Ce chapitre et le précédent montrent comment nous répondons aux problèmes de généralisation de la construction d'un télé-TP collaboratif. Le chapitre suivant visera à évaluer la plate-forme implémentée et fera l'exposé des principaux défis que devront relever les télé-TPs dans un avenir proche.

Évaluation et devenir des télé-TPs

Sommaire

Introduction	171
IV.1 Évaluation de la plate-forme de télé-TP	172
IV.1.1 Conditions d'évaluation	172
IV.1.2 Évaluation de l'interface homme machine de l'analyseur .	175
IV.1.3 Enquête de satisfaction	177
IV.1.4 Bilan	180
IV.2 Perspectives	181
IV.2.1 Devenir de la plate-forme	181
IV.2.2 Défis à venir pour les télé-TPs	183
Conclusion	189

CHAPITRE IV

Évaluation et devenir des télé-TPs

Introduction

A l'issue des travaux présentés précédemment, nous disposons d'une plate-forme de télé-TPs collaborative, dans laquelle un nouveau télé-TP peut être déployé moyennant l'écriture d'une ontologie au format OWL. Nous avons également exposé la solution que nous avons mise en place pour réguler les accès au télé-TP.

Il est difficile de prétendre avoir obtenu un prototype fonctionnel si celui-ci n'a pas subi de tests en conditions réelles. Nous relaterons donc ici l'utilisation de la plate-forme dans le cadre de télé-TPs d'électronique pour des élèves ingénieurs de deuxième année à l'ISTASE. Après avoir exposé les conditions de l'expérimentation, nous nous attacherons à interpréter les résultats d'un questionnaire soumis aux élèves.

Nous mettrons ensuite en avant les chantiers futurs induits par ces observations, pour notre plate-forme. Enfin, nous présenterons quelques évolutions du domaine des télé-TPs qui nous semblent pertinentes et possibles. En effet, des technologies en devenir permettent de dresser les contours de la métamorphose à venir des télé-TPs, avec l'objectif de résoudre des questions jusque là sans réponse.

IV.1 Évaluation de la plate-forme de télé-TP

IV.1.1 Conditions d'évaluation

Il n'est pas aisé de trouver rapidement des situations d'utilisation pour des tests, à très court terme, de la plate-forme de télé-TPs : il faut pouvoir synchroniser la réalisation du prototype avec les échéances d'enseignement.

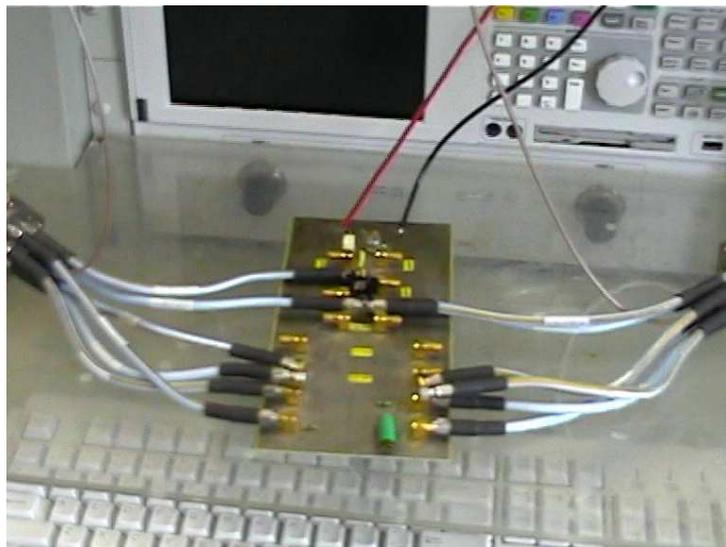


FIG. IV.1 – La carte embarquant les composants à tester par les étudiants pour le télé-TP.

Toutefois, nous avons pu procéder à des tests à échelle réelle dans le cadre de travaux pratiques destinés à des élèves ingénieurs en deuxième année de cycle d'ingénierie à l'ISTASE. Le sujet du TP mettait en avant l'utilisation d'un analyseur vectoriel de réseau pour la caractérisation de composants de base pour les télécommunication (figure IV.1). Le sujet du TP est consultable en annexe C.1.

Les conditions de ces premiers tests étaient les suivantes : les apprenants et les enseignants étaient dans une salle et l'analyseur vectoriel de réseaux dans une pièce distincte, deux étages plus haut (figure IV.2). Les étudiants avaient pour consigne de ne pas interagir directement les uns avec les autres, mais d'utiliser les moyens de communication mis à disposition par la plate-forme. Organisés en binômes, les étudiants se sont partagés en deux équipes utilisant chacune un écran tactile représentant la façade de l'analyseur vectoriel de réseaux du laboratoire. (figure IV.3). Nous n'avons pas voulu remettre en cause l'organisation par binôme car notre expérience nous porte à croire que ce format est très adapté au sujet qui est proposé.



FIG. IV.2 – Le laboratoire distant avec PC local, analyseur vectoriel de réseau et composants à caractériser.



FIG. IV.3 – Paillasse proposée aux étudiants en salle de TP à la place de la présence physique de l'instrument.

Nous sommes conscients que cette configuration de test n'est pas optimale pour tester les capacités de la plate-forme : l'enseignant était dans la même salle que les apprenants, et ceux-ci n'ont pas forcément tous joué le jeu du dialogue via les outils présentés, du fait de leur proximité immédiate. Même si la majorité des étudiants a suivi les règles, le fait que certains aient dialogué de visu avec leurs pairs malgré les consignes prouve la nécessité de collaboration et de télé-présence exprimée dans le premier chapitre de ce mémoire. Il convient de souligner qu'il s'agissait de la toute première expérience conduite dans des conditions réelles. Par conséquent, nous avons opté pour un scénario simplifié dans le but de ne pas trop mettre en péril (risque de panne) les séances prévues par nos collègues enseignants de la discipline. Douze séances de télé-TPs ont été organisées, impliquant au

total 45 étudiants.

Ce qui est fondamental, c'est que l'objectif pédagogique était pluriel. En effet, les étudiants devaient acquérir la compétence de se servir de cet appareil précisément, mais aussi apprendre à travailler ensemble, de manière collaborative, à l'aide des outils disponibles.

Au cours de ces sessions et de nos propres tests, le journal indique que 19 536 commandes ont été relayées sur l'intergiciel à destination de l'instrument.

Pour une évaluation pertinente de l'utilisation, il faut exploiter plus finement ce journal des télé-TPs généré par notre couche applicative de journalisation (un EJB¹ entité). La couche de journalisation, en excluant les actions menées sous nos identifiants, montre que le nombre de messages relayés dans le cadre des télé-TPs (tests de plate-forme exclus) se réduit à 3 409. En moyenne, 155 actions ont été menées par un binôme d'étudiants au cours du télé-TP. C'est largement plus que le nombre d'actions nécessaires pour mener à bien le télé-TP. Cela porte à croire, dans une première approche, que les étudiants ont exploité l'outil de communication mis à leur disposition (les messages relayés représentant des commandes ou des échanges textuels). Mais ce résultat est peut être biaisé par l'aspect ludique que présente un écran tactile. Les étudiants ont probablement été tentés de solliciter l'appareil plus souvent qu'ils ne l'auraient fait avec un écran normal.

Dans tous les cas, cela représente donc 284 commandes par session de télé-TP, soit 95 actions par heure, puisque la durée d'une session était de trois heures. Avec 1,7 requêtes par seconde, en moyenne, le serveur hébergeant la plate-forme est loin d'être exploité au maximum de ses capacités. D'autant que le profil du trafic est très diffus : on n'observe pas de tempêtes de messages. Cela s'explique par le fait que les étudiants enchaînent les actions une à une, que la plate-forme est verrouillée durant toute l'exécution d'une commande, et qu'il n'y ait jamais eu plus de quatre étudiants, soit deux postes de travail, connectés simultanément.

Nous considérons que le facteur de montée en charge est peu significatif au vu de l'usage de la plate-forme. Il est en effet peu probable que plusieurs milliers d'apprenants se connectent au même télé-TP, au même moment. En effet, la plate-forme est répartie entre quatre serveurs physiques permettant d'assurer le passage à l'échelle de l'ensemble.

1. Enterprise Java Bean

IV.1.2 Évaluation de l'interface homme machine de l'analyseur

Avant de tenter d'évaluer la perception qu'ont les utilisateurs de notre plate-forme, nous nous sommes penchés sur les performances de l'analyseur vectoriel de réseau employé. En effet, nous souhaitons la façade de l'instrument, ainsi que ses comportements, les plus réalistes possible, et donc les plus fidèles possible. Nous avons donc probablement reproduit les défauts que cette interface présentait.

Il faut donc être conscient des forces et faiblesses que présente l'interface homme machine de l'appareil utilisé, avant d'évaluer l'interface homme machine présentée par la déportation de la façade de l'instrument sur un écran tactile. Ainsi, nous pourrions clairement distinguer les limites qui incombent à la plate-forme développée dans notre travail, des problèmes inhérents à la nature de l'interface voulue par le constructeur de l'appareil.

Pour cela, nous avons soumis un questionnaire aux étudiants. L'utilisation de questionnaires pour évaluer une interface homme machine est une pratique courante et déjà ancienne (Root & Draper, 1983). Sur les 45 questionnaires distribués, 28 ont été rendus complétés.

La première partie de ce questionnaire concernait justement l'évaluation de l'interface de l'appareil. Pour cela, nous avons utilisé les heuristiques usuelles employées dans l'évaluation ergonomique, à savoir :

- les attributs d'utilisabilité de Nielsen (Nielsen, 1993)
- le questionnaire sur l'usabilité d'un système informatisé de Lewis (Lewis, 1995)
- l'instrument de mesure de satisfaction de Chin (Chin & Diehl, 1988)

Les questionnaires issus de ces trois heuristiques ainsi que leur dépouillement sont présentés en annexe C.2.

Nous avons choisi trois heuristiques pour limiter le biais induit par le choix arbitraire de l'une ou l'autre d'entre elles. Il est très intéressant, dans un sens tout à fait positif, de voir que ces heuristiques amènent à une appréciation globale quasi identique. En effet, alors que le texte des questions, leur nombre et les échelles des réponses varient d'une heuristique à l'autre, le résultat numérique est sensiblement identique (rapport entre le cumul des points attribués selon l'échelle et le nombre de points maximum possible, ramené sous la forme d'un pourcentage). L'observation de trouver trois résultats rapprochés demeure malgré tout peu surprenante, dans la mesure où ces heuristiques mesurent toutes la même chose, à savoir la qualité d'une interface homme machine et ce, via des questions d'ergonomie très proches d'une heuristique à l'autre. Cette observation est mise en évidence dans le

tableau IV.1.

Nom	Nombre de questions	Echelle	Apréciation globale
Nielsen	5	1 à 7	62,04 %
Lewis	19	1 à 7	59,14 %
Chin	21	0 à 9	57,50 %

TAB. IV.1 – Résultats par heuristique concernant l'évaluation de l'IHM de l'analyseur vectoriel de réseau Anritsu

Nous pensons que ce résultat est honorable pour cet appareil, qui est très difficile à prendre en main².

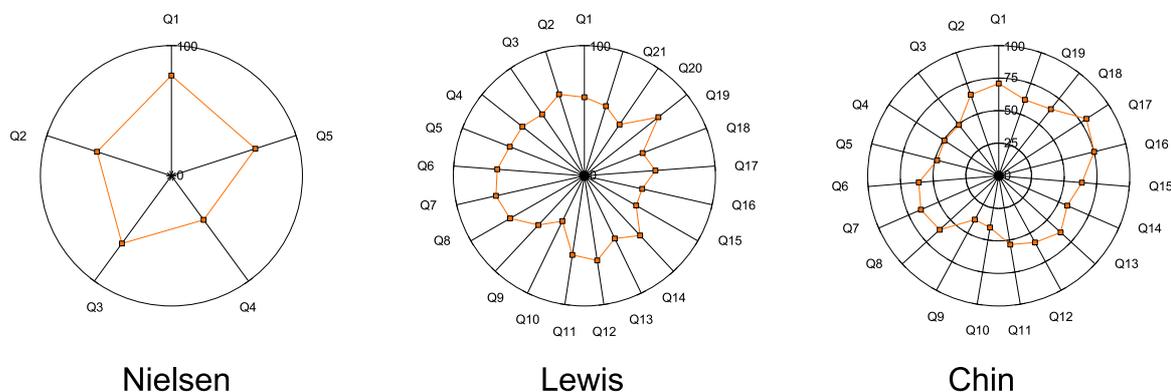


FIG. IV.4 – Visualisation du dépouillement par heuristiques: Comment les étudiants perçoivent-ils l'ergonomie de l'appareil réel?

Au delà d'un simple chiffre, les étudiants ont clairement explicité leur satisfaction par rapport à la prise en main du système (5,39 points sur 7 en moyenne, soit 77,04 % de la notation maximale, et un écart type de 1,01 point sur les 28 réponses recueillies) (question 1). Ils ajoutent qu'ils ont aimé utiliser le système (5,60 sur 7 en moyenne, question 22) et que celui-ci est agréable à l'emploi (5,26 sur 7 en moyenne, question 21).

Les élèves ont également ressenti des points faibles quant à l'utilisation de l'analyseur. Ils lui attribuent une note de 2,68 sur 7 en moyenne au sujet de l'existence de messages d'erreurs (question 14), ainsi qu'une notation moyenne de 3,48 sur une échelle de 9 quant à la pertinence de ces messages (question 34). Même si les élèves ne sont pas tous d'accord sur le critère de pertinence des messages d'erreurs (écart type supérieur à 2 points sur 9 dans les réponses), il est vrai que les messages d'erreurs sont rares et peu compréhensibles pour celui qui n'a pas coutume d'employer le dispositif. Les étudiants ont, par ailleurs, estimé qu'en cas d'erreur il n'est pas facile ni rapide de résoudre le problème rencontré

2. Il s'agit bien de l'évaluation de l'IHM de l'appareil lui-même et non de celle de la plate-forme.

(2,82 sur 7 en moyenne, question 15).

En conclusion, nous avons souhaité la première partie de ce questionnaire centrée sur l'interface de l'instrument lui même, car nous avons visé un réalisme maximum, et donc hérité des forces et lacunes de son interface homme machine. La finalité étant l'apprentissage, il nous paraît évident que l'adéquation entre l'instrument réel et la façade déportée du télé-TP doit être maximisée. Il serait problématique pour les apprenants confrontés à la réalité que les erreurs soient mieux gérées sur la façade déportée que sur l'appareil réel? Si le fait de calquer les erreurs de l'appareil à l'origine du télé-TP nuit à l'apprentissage à court terme (car cela nuit au processus d'apprentissage), une partie des objectifs des télé-TPs demeure de permettre aux apprenants de se construire des compétences sur des appareils réels.

Notre objectif en terme d'évaluation reste néanmoins de mesurer la perception qu'ont les apprenants de notre plate-forme, indépendamment de l'instrument sollicité. Cette étude préalable, pour l'analyseur vectoriel de réseaux concerné, était pourtant nécessaire puisque la qualité de l'interface de l'appareil mis en jeu a forcément un impact sur la compréhension qu'ont les apprenants de la plate-forme.

IV.1.3 Enquête de satisfaction

Nous étudions ici les résultats de la seconde partie de notre questionnaire. Après avoir évalué l'interface de l'analyseur vectoriel de réseaux, il s'agit ici de questions relatives à la plate-forme de télé-TP utilisée et de la conduite de TP des étudiants.

Cette partie du questionnaire (annexe C.3) présente quinze questions à choix fermé, ainsi que deux questions ouvertes. Les résultats seront exprimés ici sous la forme de pourcentage par question. Comme les questions étaient à choix multiples, il arrive que pour certains critères, la somme des pourcentages dépasse cent.

On note tout d'abord que le sujet du télé-TP présente peu de problèmes pour les étudiants, en dehors de ceux qui n'estiment pas avoir les bases théoriques nécessaires ou déclarent n'avoir pas préparé la séance (question 3).

Les étudiants ont tout à fait perçu l'aspect collaboratif de la plate-forme, et y sont favorables. Ils sont, en effet, plus de 82% à estimer qu'un télé-TP collaboratif est une excellente idée (question 5). Ils ont également perçu qu'ils n'étaient pas seuls devant l'appareil mais que d'autres personnes utilisaient le même dispositif en même temps (question 7), ce qui tend à montrer qu'il existe une certaine conscience de groupe au sein de la plate-

forme. Quant à l'intérêt de pouvoir dialoguer lors d'un télé-TP collaboratif (question 4), les résultats sortent dans l'ordre suivant : entraide, confronter les résultats, parler de ses vacances/soirées (*sic!*), comme l'indique la figure IV.5. Il est à noter qu'aucun étudiant n'a trouvé aucun intérêt à la possibilité de dialogue. Enfin, il est intéressant de mesurer que le premier réflexe en cas de problème pour un étudiant est de se tourner vers un pair et non vers le tuteur (question 2).

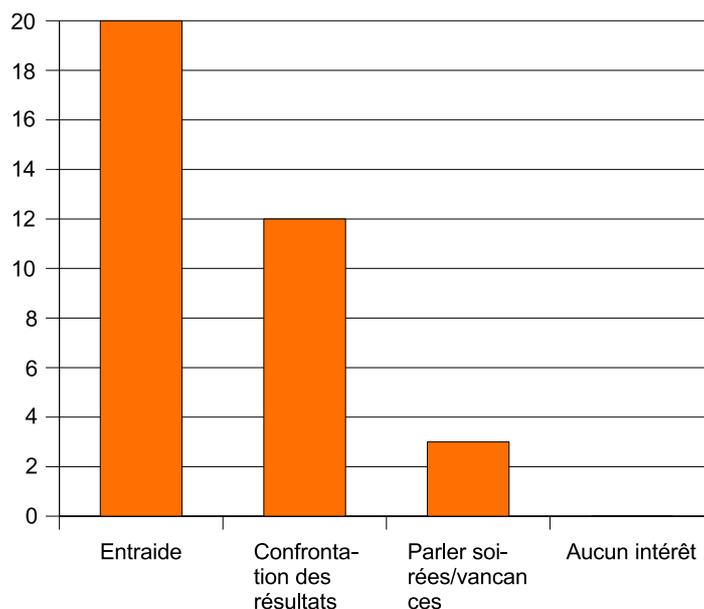


FIG. IV.5 – Répartition des réponses des étudiants à la question : « Pour vous, quel est l'intérêt de pouvoir dialoguer avec les autres personnes de la manipulation à travers la fenêtre de discussion ? » (question 4)

Par rapport à la qualité ressentie de l'apprentissage, les étudiants déclarent à hauteur de 78,57% (question 6) n'avoir pas pâti ou très peu du fait que le TP soit distant et collaboratif. À l'inverse, ils l'ont perçu comme un avantage puisqu'ils sont 64,29% à déclarer que l'utilisation d'une façade déportée a grandement facilité la rédaction de leur compte rendu de télé-TP (question 8). À la même question, 28,57% étudiants sondés estiment que cette aide était précieuse pour la rédaction mais que cela leur a demandé un effort (l'utilité de la plate-forme pour la rédaction du compte rendu est donc au final reconnue à hauteur de 92,86%). Enfin, en montrant l'appareil réel aux étudiants, ceux-ci ont pu se prononcer sur la fidélité de représentation offerte par la plate-forme. 57,14% estiment que la façade déportée est fidèle en tous points à son homologue réelle, alors que 39,29% l'estiment fidèle à quelques exceptions près. Cela porte le pourcentage d'étudiants estimant la façade suffisamment fidèle à la réalité à près de 100%! (question 13).

D'autre part, bien que les ordinateurs portables connectés à l'écran tactile manquaient

un peu de vélocité, la vitesse d'exécution de la plate-forme a été jugée satisfaisante voire très satisfaisante à hauteur de 78,58% des sondés (question 12).

Du point de vue des fonctionnalités atteintes par ce télé-TP (question 14), le bilan est moyen puisque seulement 35,71% des étudiants estiment avoir pu utiliser toutes les fonctionnalités qu'ils souhaitaient. Ce constat est à nuancer puisque viennent s'ajouter les 28,57% des étudiants déclarant que le sujet du TP était restreint à seulement une ou deux fonctionnalités.

En outre, 60,72% des utilisateurs ont volontairement essayé des commandes qui n'étaient pas demandées dans le sujet de télé-TP, juste par curiosité intellectuelle (question 15). Ce nombre est intéressant puisqu'un avantage des télé-TPs face aux simulations est qu'ils n'élaguent pas les fonctionnalités et permettent donc aux apprenants de satisfaire leur curiosité, et par voie de conséquence d'en retirer une certaine connaissance.

Globalement, on peut dire que les volontaires ayant participé à ce tout premier test sont satisfaits de ce qui leur a été proposé. C'est d'autant plus encourageant qu'il s'agit du tout premier test grandeur nature. Les étudiants sont en effet 71,43% à s'estimer à présent capables de se servir de l'instrument réel (alors que le quart restant manque simplement de confiance en soi) (question 10 et 11). Ils sont même 78,57% à se déclarer prêts à refaire le TP devant l'appareil réel (question 9).

Il reste toutefois un bémol à apporter à cette évaluation globalement très favorable des étudiants. Il s'agit de la gestion des erreurs et des problèmes techniques. Bien sûr, nous avons mis en avant le fait que l'interface de l'analyseur reproduite sur la façade déportée présente de grandes lacunes en termes de gestion des erreurs (messages la plupart du temps donnés sous la forme d'un code d'erreur voire même d'un gel de l'écran). Les commentaires libres (question 17) montrent que cela a pu être un frein à la réalisation du télé-TP. Il faut également ajouter à ce problème (propre à l'appareil mis en place) un autre problème propre à notre plate-forme. Il s'agit d'un bogue qui survenait dans des situations particulières lors de la représentation des abaques de Smith. Un abaque en particulier n'a pas pu être affichée pour une question du TP tant que le bogue n'a pas été résolu. Les premiers étudiants à effectuer leur TP sur la plate-forme ont subi l'impact direct de ce problème. Ce souci technique a engendré de la frustration auprès de nombreux étudiants. Bien que ce problème ne remet pas en cause la plate-forme elle-même, mais une seule question du sujet du télé-TP, il a diminué l'efficacité du processus d'apprentissage pour ce télé-TP en particulier.

IV.1.4 Bilan

L'évaluation présentée dans cette partie est double. Elle s'appuie sur un cas concret de mise en pratique, grandeur nature, de notre plate-forme de télé-TPs collaboratifs. Tout d'abord, nous avons mis en avant les forces et faiblesses du matériel impliqué dans la manipulation, à savoir l'analyseur vectoriel de réseaux de marque Anritsu présent au laboratoire DIOM. Il était important de bien connaître les points forts et faibles de notre appareil puisque nous avons calqué le plus possible son interface sur la façade déportée. Nous avons donc transposé de la même façon les points forts et faibles, dans le but de proposer une expérience d'apprentissage la plus réelle possible aux apprenants. Cela a forcément un impact sur la perception générale qu'ont les apprenants de la plate-forme de télé-TPs (et pas seulement de ce télé-TP-ci), particulièrement concernant la gestion des erreurs. D'autant qu'un bogue, propre à nos développements logiciels cette fois, est venu s'ajouter à la gestion maladroite des erreurs proposée par l'interface de l'analyseur de réseaux.

Concernant la plate-forme elle-même, le questionnaire distribué aux étudiants met en évidence le fait que la conscience de groupe est bien perçue par les apprenants. C'est d'autant plus positif que les étudiants emploient les procédés collaboratifs au profit d'une entraide et d'une confrontation de leurs résultats, loin devant les échanges ayant pour but de maintenir leur réseau social (messages néanmoins nécessaires). L'analyse du questionnaire met également en avant les performances satisfaisantes en termes de vitesse d'exécution de la plate-forme, ainsi qu'une grande fidélité de représentation de l'instrument impliqué. Un autre point très satisfaisant est que les étudiants ont exploité le fait de disposer de toutes les commandes d'un véritable appareil pour explorer volontairement des fonctionnalités non requises pour l'exécution du télé-TP.

IV.2 Perspectives

Nous avons dressé le bilan du télé-TP le plus utilisé durant notre travail et évalué, via un questionnaire rempli par les utilisateurs, la performance de notre plate-forme. Dans cette partie, nous allons préciser le devenir de cette plate-forme de test. Nous évoquerons également les conclusions que nous pouvons tirer avec notre recul quant à l'évolution de notre domaine scientifique.

IV.2.1 Devenir de la plate-forme

Robustesse

La liberté temporelle de réalisation d'un télé-TP est un facteur déterminant dans sa mise un place. Comme nous l'avons vu précédemment, le questionnaire montre très clairement que la robustesse et la fiabilité sont les caractéristiques prépondérantes de notre plate-forme. Nous avons également vu qu'un seul dysfonctionnement a eu un impact sur le ressenti des utilisateurs.

Ce dysfonctionnement a fort heureusement été corrigé depuis. Mais dans ces conditions, il est certain que le premier axe à développer pour le télé-TP impliquant l'analyseur de réseaux Anritsu sera de bâtir une série de tests unitaires permettant de placer plus de confiance dans la fiabilité de la manipulation distante.

Les tests effectués impliquaient volontairement un faible éloignement géographique. En effet, il fallait pouvoir bénéficier d'une solution de secours, puisque c'était la toute première fois que la plate-forme était utilisée dans des conditions réelles. Cependant, l'analyseur étant dans les mêmes locaux mais dans une salle éloignée de deux étages, il faut reconnaître que l'impact d'une mise hors service de la plate-forme sur la séance de télé-TP suite à un dysfonctionnement est très important. Il est primordial de prendre conscience de la nécessité d'un énoncé de secours, étant donné les aléas inhérents à l'utilisation d'une plate-forme sollicitant un réseau informatique. Il est certain que la séance de télé-TP demeure tributaire du bon fonctionnement du réseau fournissant l'accès à l'instrument.

Panel d'instruments

L'expérimentation conduite ne suffit pas, à elle seule, à garantir la réutilisabilité de la plate-forme pour d'autres situations de télé-TPs. C'est la raison pour laquelle nous avons particulièrement été attentifs au choix des télé-TPs construits ensuite. En effet, nous avons délibérément effectué le choix ambitieux de retenir un banc d'antennes comme télé-TP à implémenter. Le banc d'antennes consiste en un rail sur lequel se déplacent deux antennes. L'éloignement des antennes et la possibilité de positionner des obstacles entre

celles-ci constitue le sujet d'un TP pour des étudiants en télécommunications. Nous avons choisi cette manipulation car elle est suffisamment différente de celle impliquant l'analyseur vectoriel de réseaux. Le but est de montrer que notre plate-forme, dans l'état exposé dans ce mémoire, est apte à héberger des télé-TPs hétérogènes. Bien sûr, le principe reste toujours celui d'écrire l'ontologie de l'appareil pour le déployer dans la plate-forme.

Toujours dans l'optique de promouvoir l'hétérogénéité des manipulations dont notre plate-forme est capable, nous portons actuellement en ligne une manipulation chimique (figure IV.6).



FIG. IV.6 – Appareil utilisé pour le télé-TP physico-chimique.

Par ailleurs, nous prévoyons la mise à disposition d'une manipulation en opto-électronique, mettant en oeuvre une diode laser, des capteurs, des actionneurs et des instruments de mesure.

Situations d'utilisation à venir

Les offres de formations pratiques *in situ*, présentent autant de possibilités diversifiées de mise en place de télé-TP. Ces nombreuses occasions permettent de pratiquer des sessions de télé-TPs à plus grande échelle, notamment avec des apprenants issus d'autres nationalités.

IV.2.2 Défis à venir pour les télé-TPs

Explications

Au delà du devenir de notre plate-forme, nous pensons que le domaine des télé-TPs, en stagnation depuis cette dernière décennie, présente de nombreuses pistes inexplorées de recherche scientifique. Ce secteur pluridisciplinaire est appelé à évoluer, ne serait-ce qu'à cause des avancées technologiques ayant cours aujourd'hui. Ces changements permettent d'appréhender de nouveaux défis que nous devons relever pour mener les télé-TPs à maturité. En effet, il est vrai que les télé-TPs accusent encore un retard important par rapport à leurs pendants théoriques (cours magistraux et travaux dirigés à distance).

Nous livrons ici les pistes que nous envisageons pour les télé-TPs de demain. Il s'agit certainement d'un exercice délicat (avenir incertain). Néanmoins, nous pensons qu'il est nécessaire que le travail présenté dans ce mémoire s'accompagne d'une vision plus globale. Le recul dont nous disposons sur l'architecture et la construction des télé-TPs, nous amène à faire des propositions d'évolutions technologiques qui nous semblent réalisables dans un laps de temps de l'ordre de quelques années.

Transparence de localisation d'un banc instrumental distribué

Tout comme l'interopérabilité et la standardisation dans le domaine du eLearning ont permis une pratique généralisée de l'enseignement des cours et exercices à distance, l'interopérabilité des télé-TPs ouvre une voie vers de nouvelles perspectives. Un premier défi se présente dès aujourd'hui aux télé-TPs dans l'interopérabilité des instruments. Les manipulations présentées jusqu'ici impliquent un seul et unique instrument. Toutefois, pourquoi ne pas imaginer des télé-TPs impliquant tout un banc d'instruments?

Bien sûr, ce banc peut se trouver dans une même salle de travaux pratiques. Dans ce cas, la configuration diffère peu de celle impliquant un seul instrument.

Cependant, on pourrait imaginer un banc instrumental géographiquement distribué. À titre d'exemple, un premier instrument se trouverait en France, un autre au Portugal et un dernier en Chine, mais tout le banc serait vu comme un bloc unique. L'utilisateur n'aurait finalement aucune raison de connaître l'emplacement physique réel des instruments qu'il utilise (savoir qu'ils sont réels suffit). Disposer d'un banc d'instruments distribué permettrait de composer des télé-TPs à la volée, à partir des instruments disponibles sélectionnés parmi un ensemble préalablement déclaré. Cette approche est schématisée à la figure IV.7

Toutefois, une telle approche présente des contraintes techniques. Avoir un banc de

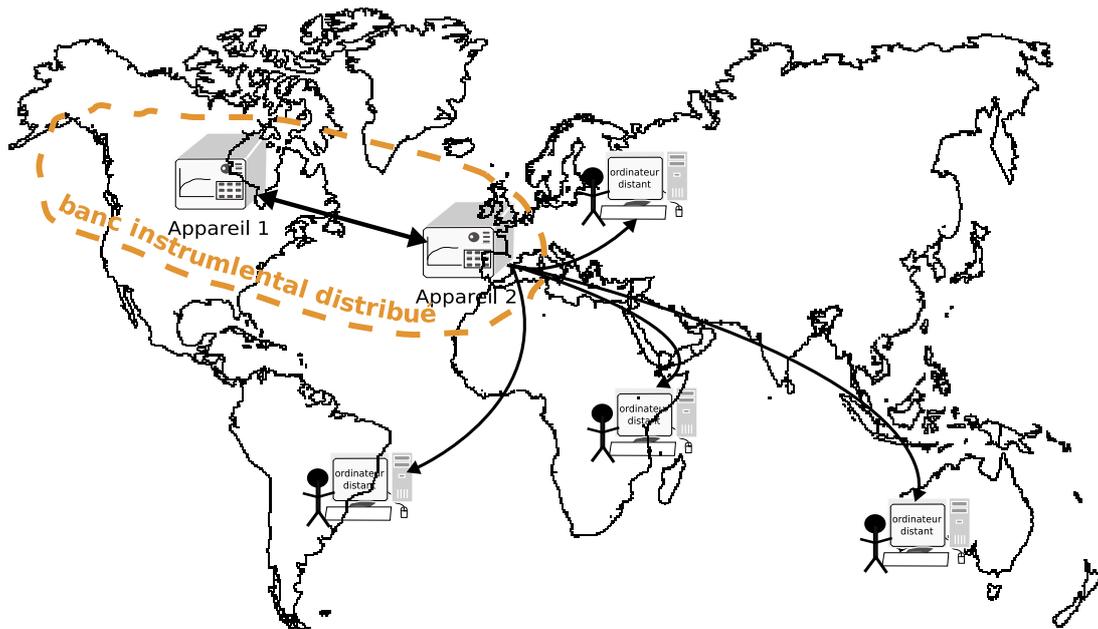


FIG. IV.7 – *Transparence de localisation des instruments d'un banc de mesure.*

mesures distribué implique de faire interopérer les instruments entre eux, ne serait-ce que pour répercuter une action effectuée sur un instrument vers les autres dispositifs composant le banc de mesure. Nous pensons que le Web sémantique pourra apporter des éléments de réponses. Nous ne voyons pas comment une couche logicielle simplement descriptive (généralement XML/RDF) pourrait répondre à ce défi technologique. Par contre nous voyons clairement l'avantage de disposer d'une ontologie pour formaliser les concepts tels que les boutons sollicités (et donc les actions déclenchées sur un instrument). L'action sur un instrument peut être véhiculée aux autres instruments avec lesquels il est interconnecté via Internet. On assiste alors à l'envoi de messages ayant une signification sémantique entre les instruments de manière à appliquer les impacts d'une opération sur un instrument aux autres instruments du banc de mesure.

Substitution des dispositifs

L'usage géographiquement distribué des dispositifs conduit à des problématiques fortes en termes de gestion de ressources. Alors qu'il est simple de mettre en place un planning papier dans des salles de manipulation pour réserver des ressources, cela est beaucoup plus difficile pour des appareils accessibles en ligne. Il convient en effet de recenser tous les instruments en ligne accessibles via une même plate-forme, et surtout d'être en mesure de déterminer si ceux-ci sont en ligne, s'ils sont déjà réservés et pour quelles périodes, ...

Une telle gestion est forcément rendu nécessaire par la multiplicité des acteurs et des ressources mises en jeu. Néanmoins, une gestion optimale des dispositifs de télé-TPs

permettrait de maximiser l'utilisation des télé-TPs. En effet, l'enjeu d'interopérabilité cité précédemment devrait permettre de rendre transparente la localisation d'un dispositif, tout en rendant interopérable les instruments entre eux, via un dialogue structuré à l'avance, sous la forme d'une ontologie. Par conséquent, il serait possible de transposer l'exécution d'un télé-TP à un autre relativement facilement. En d'autres termes, un apprenant pourrait débiter son télé-TP sur un analyseur vectoriel de réseaux et le terminer sur un analyseur équivalent, mais physiquement différent, situé dans un autre secteur géographique. Si l'on est en mesure de transposer un télé-TP d'un dispositif à un autre, et ce de manière transparente pour l'utilisateur, on comprend que l'on peut imaginer d'optimiser les utilisations des appareils onéreux. Un cas (favorable) d'application à titre d'illustration est présenté à la figure IV.8.

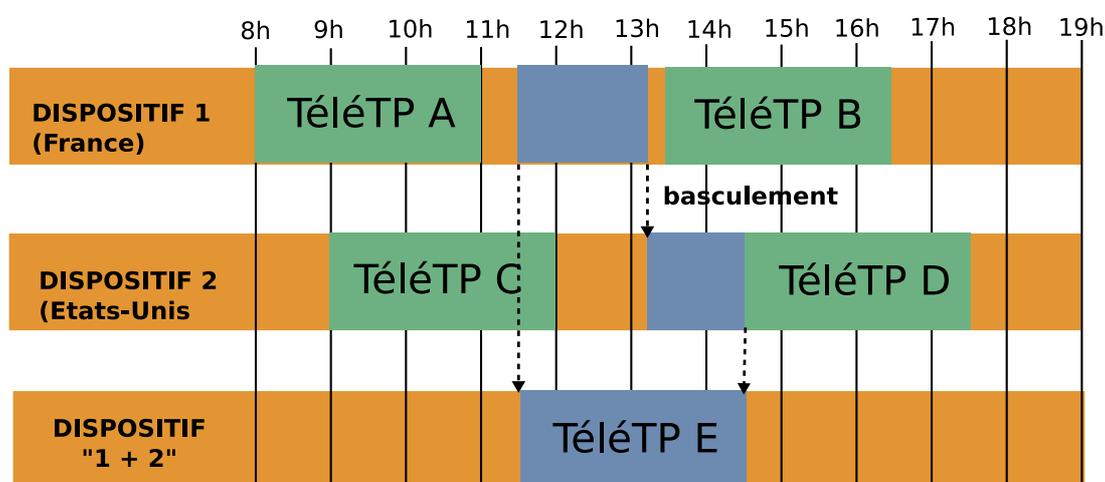


FIG. IV.8 – *Transparence de localisation des dispositifs associé au principe de basculement des télé-TPs d'un banc de mesure à un autre.*

Dans cette figure, on constate que l'utilisation des plages libres respectives des dispositifs notés « dispositif 1 » et « dispositif 2 » ne permettent pas de conduire plus de quatre télé-TPs. Par contre, l'union de ces plages libres permet de conduire le télé-TP noté « TéléTP E », qui n'aurait pu avoir lieu autrement. En effet, si on est capable de basculer l'état d'avancement d'un dispositif à l'autre (par exemple en rejouant les messages reçus par le premier instrument), il devient possible de pratiquer le télé-TP partiellement sur un premier instrument et de le terminer sur un autre.

Il convient toutefois de rester prudent : on ne peut pas multiplier ainsi les dispositifs. Le dispositif « 1+2 » n'est pas à part entière un nouveau dispositif. Une telle approche ne permet donc pas de dupliquer des instruments physiques, mais seulement d'optimiser leurs plages d'utilisation pour maximiser l'emploi de dispositifs souvent très onéreux. On comprend alors les avantages que présente le basculement de télé-TPs s'il est associé à un système efficace de réservation de ressources et à un nombre suffisamment important de dispositifs identiques. Cela n'est probablement pas le cas au sein d'un même laboratoire,

mais pourrait facilement devenir réalité si l'on tenait compte de tous les dispositifs de tous les laboratoires.

Convergence entre télé-TPs et LMS

Comme nous l'avons évoqué précédemment, nombreux sont les instituts supérieurs de formation ayant déjà inclus des formations à distance dans leur catalogue. Des portails électroniques, supports de la pédagogie et de l'entretien des relations tuteurs/apprenants, exposent les contenus pédagogiques sur Internet. Ces portails portent la dénomination de LMS (Learning Management Systems). Un LMS est donc l'outil informatique permettant de gérer et de consulter les supports de cours en ligne. En outre, les LCMS (Learning Content Management Systems) complètent les fonctionnalités des LMS, puisqu'ils offrent aux auteurs la possibilité de faciliter des contenus pédagogiques sur le LMS correspondant. Leur objectif est d'optimiser la réutilisabilité des contenus en ligne par les tuteurs, pour les enseignements classiques comme les travaux dirigés ou les cours magistraux.

Toutefois, ces plates-formes, largement utilisées aujourd'hui, ne sont pas prévues pour héberger des télé-TPs ! Comme nous l'avons vu au premier chapitre, alors que les télé-TPs proposent de mettre en pratique les enseignements théoriques, ils se retrouvent exclus des LMS, faute de technologies logicielles suffisantes. Le fait que les LMS et les télé-TPs s'ignorent n'est pas sans conséquences : toutes les situations d'apprentissage ne sont pas proposées en ligne, et surtout, il ne peut y avoir de suivi mis en place confrontant les résultats des apprenants aux exercices théoriques et leurs performances en expériences pratiques. Nous pensons donc que l'avenir des télé-TPs passera nécessairement par une convergence avec les LMS (et les LCMS sous-jacents).

Les LMS évolués utilisent des méta-données pour la description des ressources pédagogiques. Sans vouloir faire état ici des différences entre chacune des normes concernant les méta-données, nous pensons qu'une possibilité viable (n'excluant pas les autres) réside dans la norme IMS LD (Instructional Management Systems – Learning Design) qui décrit des « unités d'apprentissage » (Koper & Olivier, 2004). IMS LD va plus loin que la simple description des contenus pédagogiques à exposer en ligne puisqu'elle permet de planifier la stratégie d'apprentissage que suivront les apprenants. IMS-LD a été créé par l'« Open University of the Netherlands » (OUNL). En bref, IMS LD permet de modéliser les processus d'apprentissage identifiés par le tuteur. Pour arriver à cette modélisation, le tuteur décrit les contenus ainsi que les rôles associés à chaque acteur pendant les phases d'apprentissage. Cette spécification s'incarne dans un fichier au format XML suivant une grammaire publiée et mise à jour fréquemment³ au format XSD (XML Schema Description).

3. sur le site officiel de IMS LD : <http://www.imsglobal.org/learningdesign/>

IMS LD présente trois déclinaisons :

- « IMS LD level A » qui inclut la modélisation des activités, des environnements (les objets pédagogiques et les services associés pour les consulter) ainsi que les rôles joués par les acteurs pendant les sessions d'apprentissage,
- « IMS LD level B » qui ajoute principalement au niveau précédent la possibilité de poser des contraintes (conditions) sur le scénario suivi par un apprenant. Ce niveau permet aussi le stockage de propriétés relatives aux acteurs.
- « IMS LD level C » qui permet d'exposer le scénario décrit via un mécanisme de notifications.

Le gouffre entre plate-forme de télé-TPs et les LMS provient du fait que les télé-TPs requièrent des développements spécifiques. Ils sont donc stockés dans des plates-formes propriétaires ne suivant pas une norme permettant de mettre en place des scénarii pédagogiques. Dans le cas d'IMS LD, il n'existe pas de plan de leçons propre aux télé-TPs. En outre, il est difficile d'imaginer d'héberger tous les télé-TPs dans un LMS, de la même façon que les cours magistraux ou les travaux dirigés, qui demandent beaucoup moins d'interactivité entre les apprenants et les contenus pédagogiques. Nous pensons donc qu'une solution pragmatique consiste à intéropérer LMS et télé-TPs via un intergiciel présentant un couplage faible. Quelques implémentations de moteurs IMS LD permettent par ailleurs d'apporter des facilités d'accès aux scénarii pédagogiques joués, via des services web, moyennant un transport des messages au format SOAP (Simple Object Access Protocol). Nous voyons dans la déclinaison « IMS LD level C » un moyen d'utiliser des possibilités de notification d'évènements des télé-TPs vers le LMS, pour faire infléchir le scénario proposé à l'apprenant en fonction de ses performances au cours de la télé-opération.

En effet, l'emploi du niveau C de IMS LD permettrait d'informer le moteur de scénarii pédagogiques⁴ de l'évolution (progression ou régression) des apprenants pour une situation d'apprentissage donnée. Il serait alors possible d'adapter dynamiquement le scénario joué au regard des actions menées par l'utilisateur.

Les avantages du couplage entre télé-TPs et LMS sont vraiment nombreux et non négligeables :

- adaptation des scénarii pédagogiques : adapter la difficulté du télé-TP en fonction des actions menées par un apprenant,
- interfaces « souffleuses » : mise en évidence du prochain bouton à employer pour une question donnée, pour accompagner un apprenant dans son utilisation d'un

4. le moteur de scénarii est la brique logicielle déroulant le processus d'apprentissage prévu au départ par le tuteur

dispositif,

- évaluation : connaître la distance entre les actions menées par un étudiant et un référentiel de base (scénario associé au processus d'apprentissage considéré) permet de mesurer l'aptitude de l'apprenant à répondre à une question soulevée dans un télé-TP. Cela peut en tout cas se révéler être un outil plus précis qu'une évaluation subjective par le tuteur basée sur les actions menées par un apprenant pendant une courte période d'observation (le tuteur ne pouvant pas observer tous les apprenants sur la totalité de leur séance).

L'idée majeure vise à présenter les télé-TPs comme un support pédagogique de même niveau que les cours magistraux et les travaux dirigés dans des scénarii d'apprentissage échafaudés par les tuteurs enseignants (figure IV.9).

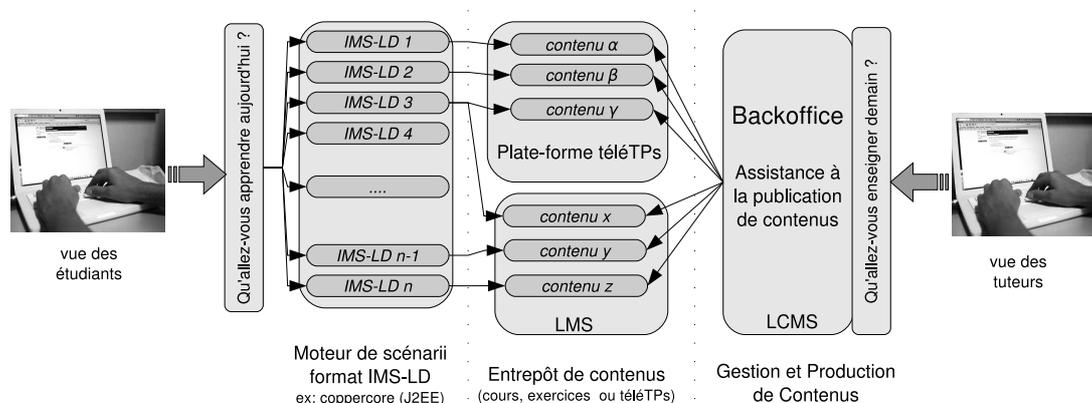


FIG. IV.9 – La convergence télé-TPs et LMS pour une gestion des télé-TPs dans les scénarii d'apprentissage en ligne.

Toutefois, il reste beaucoup de difficultés à résoudre avant d'atteindre une telle maturité dans le mariage entre télé-TPs et LMS. En effet, il faut viser à homogénéiser les accès au moteur de scénarii (non pas au niveau de l'enveloppe transport, puisque XML s'impose déjà, mais au niveau sémantique), c'est-à-dire mettre en corrélation les « Learning Object » de la norme IMS LD⁵ avec les actions menées sur un télé-TP, quel qu'il soit. Nous avons cependant conceptualisé une brique élémentaire en faveur de télé-TPs interopérables.

5. Nous prenons ici l'exemple d'une implémentation supposée qui s'appuierait sur IMS LD, mais les mêmes problèmes surgissent avec d'autres normes concurrentes comme SCORM

CHAPITRE IV

Évaluation et devenir des télé-TPs

Conclusion

Ce chapitre a mis en avant le devenir de la plate-forme ainsi que les défis futurs à relever pour les télé-TPs de manière générale. Nous pensons que ce bilan s'imposait : la fin de ce travail ne rime pas avec l'extinction du prototype, bien au contraire. Cette plate-forme est amenée à se développer à l'avenir : sous la forme d'un collaboratoire au laboratoire DIOM et via les télé-TPs inscrits au catalogue de formations à distance de l'ISTASE. D'autres projets sont actuellement en développement pour offrir un spectre d'applications directes encore plus large.

Ces développements futurs viseront principalement à apporter des réponses aux problématiques des télé-TPs de demain. Il s'agit de ce que nous avons nommé la « transparence de localisation des instruments » ou encore de la convergence entre plates-formes de télé-TPs et LMS. Notre expérience dans ce domaine nous porte à croire que le visage des télé-TPs, figé depuis une décennie, est appelé à profondément changer dans les prochaines années. La multiplication des actions de formation à distance, épaulée par des dispositifs informatiques de plus en plus évolués, va dans le sens du développement des contenus interactifs et collaboratifs et *a fortiori*, les télé-TPs, grands absents jusque là des catalogues de formations à distance, à cause de l'obstacle technologique sous-jacents qui bloquait leur mise en oeuvre. Notre plate-forme apporte à juste titre une réponse concrète à cette problématique.

Conclusion

Ce travail avait pour objectif d’asseoir les connaissances et compétences du laboratoire DIOM dans le domaine du pilotage distant d’appareils de haute technologie. Nous nous sommes particulièrement focalisés sur le domaine des télé-opérations à but de formation, puisque l’ISTASE nous offrait la possibilité de tester nos prototypes dans le cadre de sessions de formations pratiques.

Après avoir rappelé les enjeux de la manipulation distante, l’étude bibliographique menée dans ce domaine nous a conduit à deux observations majeures. Tout d’abord, les développements de télé-TPs sont systématiquement dédiés et demandent un haut niveau de compétences dans plusieurs domaines à ses concepteurs. En d’autres termes, un des obstacles majeurs à la généralisation des manipulations distantes pour la formation pratique repose sur la difficulté à formaliser la chaîne logicielle d’un télé-TP de bout en bout, dans le but d’en factoriser les développements. Par ailleurs, nous avons constaté que la collaboration entre apprenants autour d’un problème commun était un facteur essentiel pour l’acquisition des savoirs. Les apprenants sont généralement enclins à échanger leurs connaissances et compétences pour répondre à une question posée. Favoriser ces échanges est au cœur de la proposition du constructivisme, une approche pédagogique moderne, issue des communautés scientifiques gravitant autour des sciences cognitives. Il faut mettre le constructivisme en confrontation avec l’absence de modèles de télé-TPs offrant un quelconque degré de collaboration. Nous sommes manifestement en présence d’un paradoxe : tandis que les pédagogies poussent à la collaboration entre apprenants pour la découverte de savoirs par eux mêmes, les modèles d’architectures logicielles sous-jacents ne proposent qu’un parfait cloisonnement entre un instrument et un unique apprenant.

Nous avons donc essayé d’apporter une réponse à ces deux problématiques. C’est la raison pour laquelle le second chapitre a permis d’exprimer notre vision du problème de réutilisabilité des développements de télé-TPs. Nous avons essayé de montrer comment les langages du Web sémantique permettent de formaliser de nouveaux modèles d’architectures de télé-TPs, faisant abstraction de la ressource pilotée. Par la même, nous avons vu que l’emploi de modèles se situant au niveau de la couche logique (donc au-dessus de la couche descriptive) de la pile des langages du Web, permet de se préparer à relever les défis des télé-TPs de demain.

Le chapitre trois s'est attaché sur le problème de l'organisation des accès au sein d'une séance de télé-TP collaborative synchrone. Les apprenants participent à la même télé-manipulation, et doivent donc se partager la ressource distante. Il s'agit principalement de déterminer si un utilisateur qui requiert l'accès à la ressource se voit prendre la main sur celle-ci au détriment d'un autre utilisateur. Même si le partage ne repose pas toujours sur la base d'une collaboration stricte, puisqu'il peut parfois être complètement concurrentiel, il demeure des phases d'apprentissage durant lesquelles le besoin d'orchestrer les accès à l'instrument est crucial. Une telle orchestration est pertinente si elle se base sur la pédagogie en oeuvre au moment donné. Comme les pédagogies possibles sont nombreuses et variées, nous apportons un modèle formalisant les éléments entrant en jeu dans les modalités de partage courantes, ainsi qu'un moyen permettant d'agréger ces éléments en une politique des accès, de la manière la plus aisée pour les personnes ne sachant pas programmer.

Afin de mettre à l'épreuve notre modèle d'architecture, nous avons implémenté celui-ci sous la forme d'un prototype, présentant une indépendance à l'instrument piloté et embarquant des outils de collaboration synchrone. La majorité de ces développements sont accessibles via notre gestionnaire de développements collaboratifs en ligne et reposent pour la plupart sur des logiciels libres. Ainsi, nous avons pu tester ce prototype dans le cadre de travaux pratiques distants impliquant un analyseur vectoriel de réseaux. Les élèves ont réagi très positivement à l'architecture proposée. C'est le signe qu'une telle approche peut réellement révolutionner la conduite de travaux pratiques dans le cadre de formations à distance. Il s'agira d'ailleurs de l'objectif à court terme de nos activités, puisque nous travaillons d'ores et déjà sur des manipulations en chimie (mélangeur), télécommunications (banc d'antennes) et optoélectronique (tracé de la caractéristique d'une diode laser).

Comme nous avons pu le constater, le domaine des télé-TPs, en jachère depuis une décennie, semble présenter un regain d'intérêt. Cela se mesure à travers le nombre de publications scientifiques à ce sujet, ou encore l'engouement actuel des demandeurs de manipulations distantes (laboratoires, industries ou encore instituts supérieurs de formation). Il est fort probable que cet élan continuera et que des problèmes spécifiques attireront l'attention de communautés scientifiques existantes. En effet, lorsque les outils de télé-TP auront pris de l'assurance, il faudra nécessairement les rendre interopérables et maximiser le temps d'emploi des instruments (dont le coût n'est probablement pas appelé à baisser). De plus, vu la prolifération des projets de télé-TPs, il faudra pouvoir découvrir et répertorier les plates-formes existantes au sein de communautés, afin d'éviter de réinventer un même télé-TP plusieurs fois. En effet, puisque le coût d'un télé-TP demeure non négligeable, une découverte dynamique des télé-TPs existants présenterait l'avantage de réaliser des économies en termes de temps d'intégration, de compétences

requis et de coûts. Enfin, du point de vue des architectures logicielles, il est probable que les plates-formes de télé-TPs de demain ne pourront continuer d'ignorer les plates-formes d'apprentissage existantes, qui hébergent aujourd'hui les cours magistraux et les travaux dirigés. De ce dernier constat naîtra une convergence naturelle entre télé-TPs et plates-formes proposant des contenus pédagogiques en ligne.

Pour finir, il est important de mettre ces travaux en perspective avec les préoccupations industrielles d'aujourd'hui. En effet, nos résultats sont transposables au secteur économique, en terme de mutualisation de matériels, de partage des compétences et du travail collaboratif, participant ainsi au développement territorial.

Annexes

Détail de la répartition de notre bibliographie

Extrait des spécifications de OWL

Il s'agit d'un extrait des spécifications du langage OWL édictant les différentes caractéristiques et restrictions utilisées pour sémantiquement relier des classes entre elles.

Le document original est accessible en ligne à l'adresse:

<http://www.w3.org/TR/2004/REC-owl-guide-20040210/#BasicDefinitions>, il s'intitule « OWL Web Ontology Language Guide »

Il existe une version traduite de cet extrait à l'adresse:

<http://www.yoyodesign.org/doc/w3c/owl-guide-20040210/>

Cependant, seul le document original (dont l'extrait suivant est issu) fait autorité. C'est la raison pour laquelle nous avons choisi de proposer ici la version originale malgré l'existence d'une traduction.

B.1 Property Characteristics

The next few sections describe the mechanisms used to further specify properties. It is possible to specify property characteristics, which provides a powerful mechanism for enhanced reasoning about a property.

B.1.1 TransitiveProperty

If a property, P , is specified as transitive then for any x , y , and z :

$$P(x,y) \text{ and } P(y,z) \text{ implies } P(x,z)$$

The property `locatedIn` is transitive.

```
<owl:ObjectProperty rdf:ID="locatedIn">
  <rdf:type rdf:resource="&owl;TransitiveProperty" />
  <rdfs:domain rdf:resource="&owl;Thing" />
  <rdfs:range rdf:resource="#Region" />
</owl:ObjectProperty>
```

```
<Region rdf:ID="SantaCruzMountainsRegion">
  <locatedIn rdf:resource="#CaliforniaRegion" />
</Region>
```

```
<Region rdf:ID="CaliforniaRegion">
  <locatedIn rdf:resource="#USRegion" />
</Region>
```

Because the `SantaCruzMountainsRegion` is `locatedIn` the `CaliforniaRegion`, then it must also be `locatedIn` the `USRegion`, since `locatedIn` is transitive.

B.1.2 SymmetricProperty

If a property, P , is tagged as symmetric then for any x and y :

$$P(x,y) \text{ iff } P(y,x)$$

The property `adjacentRegion` is symmetric, while `locatedIn` is not. To be more precise, `locatedIn` is not intended to be symmetric. Nothing in the wine ontology at present prevents it from being symmetric.

```
<owl:ObjectProperty rdf:ID="adjacentRegion">
```

```

    <rdf:type rdf:resource="&owl;SymmetricProperty" />
    <rdfs:domain rdf:resource="#Region" />
    <rdfs:range rdf:resource="#Region" />
</owl:ObjectProperty>

<Region rdf:ID="MendocinoRegion">
    <locatedIn rdf:resource="#CaliforniaRegion" />
    <adjacentRegion rdf:resource="#SonomaRegion" />
</Region>

```

The MendocinoRegion is adjacent to the SonomaRegion and vice-versa. The MendocinoRegion is located in the CaliforniaRegion but not vice versa.

B.1.3 FunctionalProperty

If a property, P, is tagged as functional then for all x, y, and z:

$P(x,y)$ and $P(x,z)$ implies $y = z$

In our wine ontology, hasVintageYear is functional. A wine has a unique vintage year. That is, a given individual Vintage can only be associated with a single year using the hasVintageYear property. It is not a requirement of a owl:FunctionalProperty that all elements of the domain have values. See the discussion of Vintage cardinality.

```

<owl:Class rdf:ID="VintageYear" />

<owl:ObjectProperty rdf:ID="hasVintageYear">
    <rdf:type rdf:resource="&owl;FunctionalProperty" />
    <rdfs:domain rdf:resource="#Vintage" />
    <rdfs:range rdf:resource="#VintageYear" />
</owl:ObjectProperty>

```

B.1.4 inverseOf

If a property, P1, is tagged as the owl:inverseOf P2, then for all x and y:

$P1(x,y)$ iff $P2(y,x)$

Note that the syntax for owl:inverseOf takes a property name as an argument. A iff B means (A implies B) and (B implies A).

```

<owl:ObjectProperty rdf:ID="hasMaker">
    <rdf:type rdf:resource="&owl;FunctionalProperty" />
</owl:ObjectProperty>

```

```
<owl:ObjectProperty rdf:ID="producesWine">
  <owl:inverseOf rdf:resource="#hasMaker" />
</owl:ObjectProperty>
```

Wines have makers, which in the definition of Wine are restricted to Winerys. Then each Winery produces the set of wines that identify it as maker.

B.1.5 InverseFunctionalProperty

If a property, P, is tagged as InverseFunctional then for all x, y and z:

$P(y,x)$ and $P(z,x)$ implies $y = z$

Notice that producesWine in the preceding section is inverse functional. The reason is that the inverse of a functional property must be inverse functional. We could have defined hasMaker and producesWine as follows and achieved the identical effect as the preceding example.

```
<owl:ObjectProperty rdf:ID="hasMaker" />

<owl:ObjectProperty rdf:ID="producesWine">
  <rdf:type rdf:resource="&owl;InverseFunctionalProperty" />
  <owl:inverseOf rdf:resource="#hasMaker" />
</owl:ObjectProperty>
```

Think of the elements of the range in an inverse functional property as defining a unique key in the database sense. owl:InverseFunctional implies that the elements of the range provide a unique identifier for each element of the domain.

In OWL Full, we can tag a DatatypeProperty as inverseFunctional. This permits us to identify a string as a unique key. In OWL DL literals are disjoint from owl:Thing, which is why OWL DL does not permit InverseFunctional to be applied to DatatypeProperty.

B.2 Property Restrictions

In addition to designating property characteristics, it is possible to further constrain the range of a property in specific contexts in a variety of ways. We do this with property restrictions. The various forms described below can only be used within the context of an owl:Restriction. The owl:onProperty element indicates the restricted property.

B.2.1 allValuesFrom, someValuesFrom

We have already seen one way to restrict the types of the elements that make up a property. The mechanisms to date have been global in that they apply to all instances of the property. These next two, `allValuesFrom` and `someValuesFrom`, are local to their containing class definition.

The `owl:allValuesFrom` restriction requires that for every instance of the class that has instances of the specified property, the values of the property are all members of the class indicated by the `owl:allValuesFrom` clause.

```
<owl:Class rdf:ID="Wine">
  <rdfs:subClassOf rdf:resource="&food;PotableLiquid" />
  ...
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasMaker" />
      <owl:allValuesFrom rdf:resource="#Winery" />
    </owl:Restriction>
  </rdfs:subClassOf>
  ...
</owl:Class>
```

The maker of a Wine must be a Winery. The `allValuesFrom` restriction is on the `hasMaker` property of this Wine class only. Makers of Cheese are not constrained by this local restriction.

`owl:someValuesFrom` is similar. If we replaced `owl:allValuesFrom` with `owl:someValuesFrom` in the example above, it would mean that at least one of the `hasMaker` properties of a Wine must point to an individual that is a Winery.

```
<owl:Class rdf:ID="Wine">
  <rdfs:subClassOf rdf:resource="&food;PotableLiquid" />
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasMaker" />
      <owl:someValuesFrom rdf:resource="#Winery" />
    </owl:Restriction>
  </rdfs:subClassOf>
  ...
</owl:Class>
```

The difference between the two formulations is the difference between a universal and existential quantification. Relation Implications `allValuesFrom` For all wines, if they have

makers, all the makers are wineries. `someValuesFrom` For all wines, they have at least one maker that is a winery.

The first does not require a wine to have a maker. If it does have one or more, they must all be wineries. The second requires that there be at least one maker that is a winery, but there may be makers that are not wineries.

B.2.2 Cardinality

We have already seen examples of cardinality constraints. To date, they have been assertions about minimum cardinality. Even more straight-forward is `owl:cardinality`, which permits the specification of exactly the number of elements in a relation. For example, we specify `Vintage` to be a class with exactly one `VintageYear`.

```
<owl:Class rdf:ID="Vintage">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasVintageYear"/>
      <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

We specified `hasVintageYear` to be a functional property, which is the same as saying that every `Vintage` has at most one `VintageYear`. This application of that property to `Vintage` using the cardinality restriction asserts something stronger, that every `Vintage` has exactly one `VintageYear`.

Cardinality expressions with values limited to 0 or 1 are part of OWL Lite. This permits the user to indicate 'at least one', 'no more than one', and 'exactly one'. Positive integer values other than 0 and 1 are permitted in OWL DL. `owl:maxCardinality` can be used to specify an upper bound. `owl:minCardinality` can be used to specify a lower bound. In combination, the two can be used to limit the property's cardinality to a numeric interval.

B.2.3 hasValue [OWL DL]

`hasValue` allows us to specify classes based on the existence of particular property values. Hence, an individual will be a member of such a class whenever at least one of its property values is equal to the `hasValue` resource.

```
<owl:Class rdf:ID="Burgundy">
  ...
  <rdfs:subClassOf>
```

```
<owl:Restriction>
  <owl:onProperty rdf:resource="#hasSugar" />
  <owl:hasValue rdf:resource="#Dry" />
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

Here we declare that all Burgundy wines are dry. That is, their `hasSugar` property must have at least one value that is equal to `Dry`.

As for `allValuesFrom` and `someValuesFrom`, this is a local restriction. It holds for `hasSugar` as applied to Burgundy.

Travaux pratiques sur l'analyseur vectoriel de réseaux

C.1 Sujet du travail pratique

ETUDE SUR L'ANALYSEUR VECTORIEL DE RESEAUX DES LIMITATIONS EN FREQUENCE DE COMPOSANTS PASSIFS ET DE MONTAGES AMPLIFICATEURS

BUT :

Le but de cette manipulation est de montrer les limites fréquentielles de fonctionnement des composants passifs (résistance, capacité, bobine) et d'un exemple de composant actif (amplificateur) lorsque la fréquence d'utilisation devient très importante (supérieure à 100 Mhz).

Vous allez effectuer cette manipulation en utilisant une plate-forme de télé-TP.

I – Quelques informations sur la plate-forme de télé-TPs que vous utilisez

Les deux principales caractéristiques de celle-ci sont, pour ce qui nous concerne ici, la fidélité de représentation de l'instrument ainsi que l'aspect collaboratif de la manipulation.

En effet, il s'agit de vous confronter à distance avec un dispositif réel, tout en reproduisant le plus fidèlement possible son apparence et son comportement.

Vous pourrez d'ailleurs aller vous rendre compte sur place, au début de la séance, de l'allure de l'instrument piloté ainsi que de la manipulation physiquement mise en place (dans un bureau au laboratoire DIOM). Il s'agira également de prendre conscience qu'il s'agit d'un véritable appareil (onéreux) que vous pilotez à distance (il ne s'agit en aucun cas d'une simulation). Notez que vos PCs sont munis d'un écran tactile pour plus de réalisme (reproduire la gestuelle des interactions avec l'instrument que vous auriez pu mener in situ).

De plus, le TP est collaboratif. Cela signifie que vous pouvez vous connecter, via l'application installée sur les PCs de votre paillasse, à plusieurs sur le même instrument. Dès lors, une couleur vous est associée. Chaque action menée par une personne connectée sera suivie d'un acquittement visuel (qui dure le temps de la commande!) de la couleur associée à cette personne. Vous déterminez ainsi qui a fait quoi. Dans le cadre des TPs, la politique en place est de laisser à tous la possibilité d'appuyer sur un bouton. (Il faudra donc travailler entre binômes, et vous entraider).

II – Calibrage de l'analyseur Les composants qui vont être testés sont reliés à l'analyseur via un câble et un adaptateur.

Le calibrage est fait à l'extrémité du câble. On dispose de différents standards (charge adaptée à 50W, circuits ouverts, court-circuits, connecteurs pour liaison directe entre les câbles). Les connecteurs et les câbles utilisés en électronique hyperfréquence méritent un soin particulier lors de leur maniement.

L'appareil a été calibré précédemment et les données ont déjà été chargées dans l'analyseur.

III– Étude de composants passifs Vous trouverez en annexe un rappel de cours

III.1 – La résistance $R = 47 \Omega$

Observer à l'analyseur de réseau l'allure du module de S11 en dB.

Relever dans un tableau la valeur de la fréquence et celle de S11 en dB correspondante sur la bande [40 MHz – 500 Mhz].

Tracer la courbe donnant le module de S11 en fonction de la fréquence. Conclure.

III.2 – Le condensateur $C = 100 \text{ pF}$

Observer à l'analyseur de réseau l'allure de S11 sur l'abaque de Smith. Relever dans un tableau, la fréquence et l'impédance (partie réelle et imaginaire) sur la bande [40 MHz-500 MHz]. Calculer sous excel puis tracer C(f).

Analyse des résultats :

- la valeur de la capacité est-elle toujours constante? Sinon à partir de quelle valeur y a-t-il changement?

- le comportement du composant est-il toujours le même?

III.3 – La bobine à air

On fera le même travail que pour la capacité.

Observer à l'analyseur de réseau l'allure de S11 sur l'abaque de Smith. Relever dans un tableau, la fréquence et l'impédance (partie réelle et imaginaire) sur la bande [40 MHz-500 MHz]. Calculer sous Excel puis tracer L(f).

Analyse des résultats :

- la valeur de l'inductance est-elle toujours constante? Sinon à partir de quelle valeur

y-a-t-il changement ?

- le comportement du composant est-il toujours le même ?

- le modèle de la bobine, habituellement utilisé en électronique, est-il toujours valable ?

IV – Amplificateur intégré Mesurer à l'analyseur vectoriel de réseau les paramètres S11 et S21. Tracer le gain de transfert et le gain en puissance effective en dB en fonction de la fréquence sur la bande [40 M Hz-500 Mhz]. Mesurer à l'analyseur vectoriel de réseau le paramètre S12.

L'amplificateur est-il unilatéral ?

Auteur du sujet: Nadège OLLIER, Maître de conférence à l'Istase.

C.2 Questionnaire distribué aux étudiants

Nom Prénom :

Groupe TP:

Qui suis-je? étudiant enseignant

Questionnaire

Retour d'expérience sur le TP analyseur vectoriel de réseaux Anritsu

Ce questionnaire a pour but d'évaluer la plate-forme d'instrumentation collaborative distante que vous venez d'utiliser. Quelque soit vos réponses (favorables ou défavorables), ce questionnaire n'intervient **pas du tout** dans votre évaluation. Soyez donc francs et honnêtes dans vos réponses!

Vous êtes également invités à remplir cette fiche seul(e). Bien sûr, vous pouvez choisir de ne pas remplir vos noms et prénoms à ce questionnaire pour répondre de façon anonyme.

Partie 1: Le système est-il utilisable? (*heuristique de Nielsen*)

Le système est-il:

n°	Observations	1 Pas du tout d'accord	2	3	4	5	6	7 Tout à fait d'accord
Q1	Facile à prendre en main							
Q2	Efficace à utiliser							
Q3	Facile à se rappeler							
Q4	Exempt d'erreurs							
Q5	Subjectivement satisfaisant							

Par ailleurs, le système est-il : (*heuristique de Lewis*)

N°	Questions	1 – Pas du tout	2 – Un peu	3 – Presque	4 – Correct	5 – Bon	6 – Très bon	7 – Parfait	Non applicable
6	En général, je suis satisfait(e) de la facilité d'utilisation de ce système								
7	Ce système est simple à utiliser								
8	J'ai complété mon travail correctement en utilisant ce système								
9	J'ai été en mesure de compléter rapidement mon TP avec ce système								
10	J'ai complété mon travail efficacement en utilisant ce système								
11	Je me sens à l'aise avec ce système								
12	J'ai eu de la facilité à apprendre comment utiliser ce système								
13	Je crois être devenu(e) rapidement efficace en utilisant ce système								
14	Les messages d'erreur présentés par ce système m'indiquent clairement comment résoudre les problèmes								
15	Lorsque j'ai fait une erreur d'utilisation sur ce système, il m'a été facile et rapide de la corriger								
16	Les outils d'aide disponibles sur ce système (tels que l'aide en ligne, les messages à l'écran et autres informations) sont utiles								
17	J'ai facilement trouvé l'information que je cherchais								
18	L'information fournie avec ce système est facile à comprendre								
19	L'information disponible sur ce système contribue à me soutenir dans la réalisation des tâches et des scénarios								
20	L'organisation de l'information dans les écrans du système est claire								
21	L'interface de ce système est agréable								
22	J'aime utiliser l'interface de ce système								
23	Ce système possède toutes les fonctions et le potentiel correspondant à mes attentes pour ce TP								
24	En général, je suis satisfait(e) de ce système								

Aussi: (heuristique de Chin)

N°	Questions	0 – Nul/Pas du tout	1	2	3	4	5	6	7	8	9 – Parfait
25	Lecture aisée des caractères à l'écran										
26	La couleur simplifie l'exercice										
27	Organisation de l'information										
28	Séquencement des écrans										
29	Les termes sont clairs										
30	La terminologie est propre au domaine										
31	Position des messages à l'écran										
32	Gestion des actions par rapport à la réalité d'utilisation										
33	L'ordinateur informe sur l'avancée de la tâche en cours										
34	Pertinence des messages d'erreurs affichés										
35	Apprentissage intuitif de l'utilisation de l'appareil										
36	Exploration de nouvelles fonctionnalités par curiosité										
37	Facilité à se souvenir des noms de commande										
38	Lancer une action est simple										
39	Présence de message d'aide sur l'écran										
40	Présence de documentation supplémentaire										
41	Vitesse d'exécution d'une commande										
42	Robustesse de l'application										
43	Le système est silencieux										
44	Vous arrivez à corriger vos erreurs facilement										
45	L'interface propose des niveaux d'utilisation différents										

Partie 2: Questionnaire sur la pratique de TP

1- Pour quelle raison avez-vous fait appel à l'enseignant?

1. Problème technique (réseau, erreur, ...)
2. Incompréhension du sujet
3. Valider une étape du TP
4. Demande de complément d'information
5. Autre: ...

2- Premier réflexe en cas de pépin?

1. Dialoguer avec les personnes qui font le même TP que moi
2. Dialoguer avec les personnes ayant déjà fait le TP
3. Demander de l'aide à l'enseignant
4. Chercher de la documentation
5. Autre: ...

3- Quel est le principal problème du sujet de TP:

1. Je ne comprends pas l'énoncé car il me manque des bases théoriques
2. Je ne comprends pas l'énoncé car il est ambigu
3. J'ai mal compris au départ sans m'en rendre compte
4. Je n'ai pas préparé la phase théorique
5. Je ne retrouve pas les résultats théoriques au cours de la manipulation
6. Aucun problème

4- Pour vous, quel est l'intérêt de pouvoir dialoguer avec les autres personnes de la manipulation à travers la fenêtre de discussion:

1. Pouvoir s'entraider en cas de problème
2. Confronter les résultats obtenus
3. Parler de ses vacances ou de la dernière soirée étudiante
4. Aucun intérêt pour moi

5- Que pensez-vous de l'idée de faire un TP à plusieurs?

1. Excellente idée

2. Cela ne change pas grand chose
3. Mauvaise idée

6- Pensez-vous que votre apprentissage ait souffert d'avoir manipulé l'analyseur hyperfréquence à distance au lieu de vous trouver in situ ?

1. Pas du tout
2. Un peu
3. Beaucoup
4. Complètement, je me suis senti(e) perdu(e).

7- Combien de personnes ont travaillé sur l'instrument en même temps que vous ?

1. 0
2. une seule
3. deux
4. trois
5. quatre
6. cinq et plus

8- Faire des mesures via l'ordinateur a-t-il facilité la rédaction de votre compte rendu ?

1. Tout à fait, j'ai pu facilement capturer les courbes par exemple.
2. Un peu, mais j'ai eu un peu de mal à m'y faire
3. J'ai été à l'inverse gêné(e).
4. J'ai été énormément gêné(e), la prochaine fois je préférerais être devant le vrai dispositif pour relever ma courbe point à point.

9- Seriez-vous prêt(e) à refaire le sujet de TP, devant l'instrument cette-fois ci ?

1. Oui
2. Non

10- Pensez-vous que vous seriez directement opérationnel si vous vous trouveriez en face du réel analyseur hyperfréquence ?

1. Oui
2. Non

11- Si non, pourquoi?

1. Piloter à distance l'analyseur ne m'a pas permis d'acquérir suffisamment de connaissance sur son fonctionnement
2. Je n'ai pas compris la théorie du TP
3. Je n'ai pas compris le sujet du TP
4. Je n'ai pas confiance en mes capacités / peur de ne pas me souvenir
5. Autre: ...

12- Comment jugez-vous la vitesse d'exécution de la plate-forme?

1. Très satisfaisante
2. Satisfaisante
3. Peu satisfaisante
4. Décevante
5. Très mauvaise

13- Jugez-vous que l'interface proposée représente le plus fidèlement possible l'analyseur utilisé?

1. Tout à fait
2. Oui, à quelques exceptions près
3. Bof, il subsiste des incohérences
4. Pas du tout

14- Combien de questions¹ n'avez vous pas pu réaliser par manque de fonctionnalité de l'appareil?

1. 0
2. 1 ou 2
3. 3 ou 4
4. 5 et plus

15- Avez-vous essayé d'autres fonctionnalités de l'analyseur, juste par curiosité?

1. Oui, plusieurs

1. Cette question induit du bruit. Il aurait pu être plus pertinent de poser: « Combien de questions *du sujet posé* n'avez vous pas pu réaliser par manque de fonctionnalité de l'appareil »

2. Oui, une seule
3. Oui involontairement, je me suis trompé de bouton
4. Non jamais

16- Quels sont les points qui vous ont le plus satisfait ?

1. -
2. -
3. -

17- Quels sont les points qui vous ont le plus frustré ?

1. -
2. -
3. -

Merci beaucoup d'avoir pris le temps de remplir ce questionnaire !!!

C.3 Dépouillement du questionnaire

Heuristiques	Questions	Etudiant a	Etudiant b	Etudiant c	Anonyme 1	Anonyme 2	Anonyme 3	Etudiant d	Etudiant e	Etudiant f	Etudiant g	Etudiant h	Etudiant i	Etudiant j	Etudiant k	Etudiant l	Etudiant m	Etudiant n	Etudiant o	Etudiant p	Etudiant q	Etudiant r	Etudiant s	Etudiant t	Etudiant u	Etudiant v	Etudiant w	Anonyme 4	Etudiant x	NOMBRE	TOTAL	MAX	MOYENNE	%	ECART TYPE
Nielsen	Q1	6	6	7	5	6	6	6	7	4	4	5	4	4	4	6	5	5	7	5	4	7	5	5	5	7	5	4	7	28	151	196	5,39	77,04	1,01
	Q2	2	5	7	3	5	5	3	6	2	2	5	6	6	5	6	6	6	6	4	2	5	5	6	3	6	3	7	28	117	196	4,18	59,69	1,56	
	Q3	4	3	5	3	5	4	4	5	4	4	5	5	6	4	5	6	6	5	4	4	5	5	6	5	5	4	3	28	126	196	4,5	64,29	1,12	
	Q4	2	6	2	1	5	2	4	3	2	2	5	1	5	3	4	5	6	3	2	2	3	4	5	3	1	6	3	7	28	82	196	2,93	41,84	1,73
	Q5	2	7	6	3	6	5	5	4	4	4	5	5	6	1	6	6	5	5	4	4	5	5	6	7	1	6	3	28	132	196	4,71	67,35	1,62	
		16	27	27	26	23	25	18	17	23	24	15	15	11	25	21	24	24	16	16	23	23	23	29	23	14	29	16	35	608	980	62,04			
Lewis	Q6	3	7	7	4	5	5	5	5	4	5	4	4	5	6	6	5	5	7	4	4	5	6	6	2	5	6	5	6	28	139	196	4,96	70,92	1,32
	Q7	2	7	6	4	5	4	3	3	4	3	4	4	5	6	5	5	5	4	3	3	5	5	6	4	5	5	4	6	28	129	196	4,61	65,82	1,26
	Q8	2	6	7	1	5	4	2	2	2	5	2	1	4	1	2	2	2	4	4	3	5	4	6	3	2	4	5	7	28	97	196	3,46	49,49	1,9
	Q9	2	5	5	1	6	3	5	2	6	2	2	1	1	1	1	1	2	1	1	4	5	5	6	6	4	5	5	7	28	97	196	3,46	49,49	2,11
	Q10	4	5	6	1	6	2	4	2	6	2	5	2	1	1	1	4	4	4	4	1	5	5	5	4	5	6	4	6	28	95	196	3,39	48,47	1,88
	Q11	4	5	7	1	6	3	4	3	3	4	5	2	5	5	7	5	4	4	4	3	5	5	5	3	2	5	2	6	28	120	196	4,29	61,22	1,39
	Q12	3	7	5	2	5	5	5	4	4	4	4	4	5	5	7	5	5	7	4	3	5	5	5	5	1	5	4	6	26	118	182	4,54	64,84	1,37
	Q13	4	4	5	4	5	4	5	3	3	5	4	4	7	6	7	3	3	6	5	2	4	4	5	5	2	6	4	6	28	120	196	4,29	61,22	1,28
	Q14	3	4	6	1	5	3	5	3	3	2	4	5	4	6	7	3	3	5	5	2	4	5	5	4	2	4	4	4	28	75	196	2,68	38,27	1,44
	Q15	4	6	3	1	4	2	4	1	1	3	1	2	1	1	1	2	2	3	2	3	3	4	5	5	3	4	3	6	28	79	196	2,82	40,31	1,63
	Q16	3	4	2	5	4	3	4	3	3	6	5	4	5	1	1	1	4	5	4	6	3	3	6	5	3	5	2	5	28	105	196	3,75	53,57	1,53
	Q17	5	3	6	4	4	4	5	4	4	5	5	4	5	1	1	4	3	4	5	5	3	3	5	2	2	4	4	6	26	106	182	4,08	58,24	1,21
	Q18	6	4	6	4	4	4	4	4	4	4	6	4	4	4	4	6	3	5	5	4	3	4	6	5	3	5	4	6	24	108	168	4,5	64,29	0,91
	Q19	6	5	6	4	4	4	4	4	4	6	6	4	6	4	4	4	4	5	4	5	3	5	5	2	3	4	4	25	100	175	4	57,14	1,13	
	Q20	7	6	7	3	5	6	4	3	6	4	4	3	5	4	5	6	2	7	7	5	4	5	7	7	4	5	5	5	107	168	4,46	63,69	1,53	
	Q21	7	5	7	7	5	5	4	4	4	5	5	7	5	4	5	5	2	7	7	5	4	5	7	4	5	5	3	6	27	142	189	5,26	75,13	1,35
	Q22	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	25	140	175	5,6	80	1,23
	Q23	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	26	118	182	4,54	64,84	1,42
	Q24	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	25	108	175	4,32	61,71	1,05
																													6	2108	3556	59,14			

Heuristique	Questions	Etudiant a	Etudiant b	Etudiant c	Anonyme 1	Anonyme 2	Anonyme 3	Etudiant d	Etudiant e	Etudiant f	Etudiant g	Etudiant h	Etudiant i	Etudiant j	Etudiant k	Etudiant l	Etudiant m	Etudiant n	Etudiant o	Etudiant p	Etudiant q	Etudiant r	Etudiant s	Etudiant t	Etudiant u	Etudiant v	Etudiant w	Anonyme 4	Etudiant x	NOMBRE	TOTAL	MAX	MOYENNE	%	ECART TYPE
	Q25	7	2	6	6	6	6	3	5	7	3	9	9	9	9	9	9	0	6	1	5	4	8	1	8	3	6	6	9	28	152	252	5,43	60,32	2,77
	Q26	6	4	7	9	6	6	4	6	9	4	6	2	9	9	6	2	8	6	4	5	5	8	2	5	6	5	5	9	28	165	252	5,89	65,48	1,97
	Q27	7	5	7	4	5	5	4	3	8	3	6	5	7	9	9	6	6	6	2	4	5	7	3	4	5	6	4	9	28	144	252	5,14	57,14	1,77
	Q28	5	?	6	6	5	5	4	4	8	4	6	5	9	9	0	4	4	5	3	5	5	4	4	5	6	5	4	27	147	243	5,44	60,49	2,1	
	Q29	6	5	6	6	5	6	4	5	7	5	8	8	9	9	8	9	9	5	5	3	5	8	4	3	3	6	4	28	154	252	5,5	61,11	1,94	
	Q30	7	6	6	6	4	5	6	6	8	5	8	8	8	9	8	6	9	5	5	5	5	7	5	2	3	5	5	28	188	252	6	66,67	1,73	
	Q31	8	?	?	?	7	5	4	5	7	5	5	5	8	9	7	5	7	5	7	5	5	4	5	6	6	6	4	26	162	234	6,23	69,23	1,69	
	Q32	7	5	7	9	4	6	6	2	8	4	5	8	9	9	7	3	8	7	6	6	5	3	4	4	5	6	4	28	130	252	5,89	65,48	2,08	
	Q33	6	5	5	8	1	5	1	1	5	5	5	5	5	0	4	3	6	7	5	5	5	5	4	1	4	6	5	28	130	252	4,64	51,59	2,3	
Chin	Q34	4	?	5	4	2	5	1	0	1	6	5	5	0	0	4	2	9	5	5	5	5	6	1	3	4	5	5	27	94	243	3,48	38,68	2,17	
	Q35	6	5	6	4	7	6	6	2	6	5	6	6	6	9	4	6	9	6	4	5	5	4	6	3	5	5	4	28	155	252	5,54	61,51	1,78	
	Q36	7	?	?	?	5	6	6	7	6	5	6	6	6	9	7	5	9	6	5	5	5	3	4	6	3	5	4	25	148	225	5,92	65,78	1,6	
	Q37	4	7	7	4	6	?	?	3	6	5	7	7	7	9	2	3	7	5	3	5	5	5	6	4	4	4	3	27	130	243	4,81	53,5	2,06	
	Q38	5	?	4	6	6	5	5	2	6	4	3	6	3	9	9	6	9	6	6	5	5	6	4	5	7	4	4	27	152	243	5,63	62,55	1,95	
	Q39	6	3	4	2	6	4	0	0	5	4	3	3	3	9	4	3	2	5	6	5	5	5	3	4	2	4	4	28	115	252	4,11	45,63	2,11	
	Q40	8	5	4	2	3	5	0	0	5	6	5	5	3	4	5	4	3	5	6	5	5	5	4	5	2	5	3	28	114	252	4,07	45,24	1,93	
	Q41	6	6	6	3	4	6	7	1	5	5	6	3	3	4	4	3	3	6	4	7	5	6	4	5	6	5	9	137	252	4,89	54,37	1,97		
	Q42	5	?	7	5	6	6	1	0	5	6	5	3	3	0	3	2	9	5	7	5	5	7	5	3	6	6	5	27	116	243	4,3	47,74	2,29	
	Q43	3	5	8	5	8	5	9	9	7	5	8	8	8	4	9	7	9	7	7	5	5	9	3	6	6	6	4	28	182	252	6,5	72,22	2,04	
	Q44	1	6	8	4	6	4	0	1	2	6	3	3	3	0	3	3	1	5	2	5	5	6	8	4	6	6	4	28	120	252	4,29	47,62	2,52	
	Q45	4	4	7	6	7	5	0	2	6	5	3	3	3	?	4	4	9	5	4	?	5	6	5	4	6	7	5	26	131	234	5,04	55,98	1,93	
																													2981	5184		57,5			

: Questions citer dans le texte de ce mémoire
 : Bon score
 : Mauvais score

Questions	Etudiant a	Etudiant b	Etudiant c	Anonyme 1	Anonyme 2	Anonyme 3	Etudiant d	Etudiant e	Etudiant f	Etudiant g	Etudiant h	Etudiant i	Etudiant j	Etudiant k	Etudiant l	Etudiant m	Etudiant n	Etudiant o	Etudiant p	Etudiant q	Etudiant r	Etudiant s	Etudiant t	Etudiant u	Etudiant v	Etudiant w	Anonyme 4	Etudiant x	NOMBRE TOTAL	MAX	MOYENNE	%	ECART TYPE	
1	1	0	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	28	21	28	0,75	75	0,43
2	0	1	0	0	0	1	1	1	0	1	1	1	1	1	1	0	0	1	0	1	0	0	0	0	0	0	0	1	28	7	28	0,25	25	0,43
3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	28	4	28	0,14	14,29	0,35
4	1	0	0	1	0	1	0	1	1	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	28	8	28	0,29	28,57	0,45
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	28	1	28	0,04	3,57	0,19
1	1	1	0	1	0	0	1	1	1	0	1	1	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	28	15	28	0,54	53,57	0,5
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	28	0,04	3,57	0,19	0,19	
3	0	1	1	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	28	15	28	0,54	53,57	0,5
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	28	1	28	0,04	3,57	0,19
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	28	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	27	1	27	0,04	3,7	0,19	
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	0	27	9	27	0,33	33,33	0,47
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	0	27	3	27	0,11	11,11	0,31
4	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	0	27	3	27	0,11	11,11	0,31
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	0	27	3	27	0,11	11,11	0,31
6	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	?	?	?	?	?	?	?	?	?	?	?	1	27	9	27	0,33	33,33	0,47
1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	28	20	28	0,71	71,43	0,45
2	0	1	0	1	0	0	1	1	1	0	1	1	1	1	1	0	1	1	1	1	0	0	1	1	1	1	1	0	28	12	28	0,43	42,86	0,49
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	3	28	0,11	10,71	0,31
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	28	0	0	0
1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	28	23	28	0,82	82,14	0,38
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	2	28	0,07	7,14	0,26
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	3	28	0,11	10,71	0,31
1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	28	15	28	0,54	53,57	0,5
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	28	7	28	0,25	25	0,43
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	5	28	0,18	17,86	0,38
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	1	28	0,04	3,57	0,19
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	28	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	9	28	0,32	32,14	0,47
3	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	10	28	0,36	35,71	0,48
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	9	28	0,32	32,14	0,47
5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	28	0	0	0
1	1	0	1	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	28	18	28	0,64	64,29	0,48
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	28	8	28	0,29	28,57	0,45
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	2	28	0,07	7,14	0,26
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	28	0	0	0
1	1	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	22	28	0,79	78,57	0,41
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	6	28	0,21	21,43	0,41

Questions	Etudiant a	Etudiant b	Etudiant c	Anonyme 1	Anonyme 2	Anonyme 3	Etudiant d	Etudiant e	Etudiant f	Etudiant g	Etudiant h	Etudiant i	Etudiant j	Etudiant k	Etudiant l	Etudiant m	Etudiant n	Etudiant o	Etudiant p	Etudiant q	Etudiant r	Etudiant s	Etudiant t	Etudiant u	Etudiant v	Etudiant w	Anonyme 4	Etudiant x	NOMBRE	TOTAL	MAX	MOYENNE	%	ECART TYPE
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	28	20	28	0,71	71,43	0,45
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	8	28	0,29	28,57	0,45	
11	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	14	3	14	0,21	21,43	0,41	
	2	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	14	3	14	0,21	21,43	0,41	
	3	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	14	1	14	0,07	7,14	0,26	
	4	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	14	6	14	0,43	42,86	0,49	
5	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	14	2	14	0,14	14,29	0,35	
12	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	4	28	0,14	14,29	0,35	
	2	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	28	18	28	0,64	64,29	0,48	
	3	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	4	28	0,14	14,29	0,35	
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	1	28	0,04	3,57	0,19	
	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	1	28	0,04	3,57	0,19	
13	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	28	16	28	0,57	57,14	0,49	
	2	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	28	11	28	0,39	39,29	0,49	
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	1	28	0,04	3,57	0,19	
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	28	0	0	0	
14	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	28	10	28	0,36	35,71	0,48	
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	28	8	28	0,29	28,57	0,45	
	3	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	5	28	0,18	17,86	0,38		
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	5	28	0,18	17,86	0,38		
15	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	11	28	0,39	39,29	0,49	
	2	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	6	28	0,21	21,43	0,41	
	3	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	28	7	28	0,25	25	0,43	
	4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	4	28	0,14	14,29	0,35	

Index des acronymes

- ACAO** Apprentissage Collaboratif Assisté par Ordinateur (*p. 47*)
- ASP** Active Server Pages (*p. 62*)
- AVR** Analyseur Vectoriel de Réseaux (*p. 99*)
- CAI** Computer-Aided Instruction (*p. 32*)
- CGI** Common Gateway Interface (*p. 62*)
- CMC** Computer-Mediated Communication (*p. 51*)
- CMS** Content Management Systems (*p. 85*)
- CORBA** Common Object Request Broker Architecture (*p. 62*)
- CSCL** Computer Supported Collaborative Work (*p. 47*)
- CSCW** Computer Supported Collaborative Work (*p. 24*)
- DAQ** Data AcQuisition (*p. 61*)
- DCMI** Dublin Core Metadata Initiative (*p. 83*)
- DIF** Droit Individuel à la Formation (*p. 18*)
- DIOM** Dispositifs et Instrumentation en Optoélectronique et Micro-ondes (*p. 18*)
- DST** Dispositif Sous Test (*p. 100*)
- EAO** Enseignement Assisté par Ordinateur (*p. 32*)
- EIAH** Environnements Informatiques l'Apprentissage Humain (*p. 23*)
- EIAO** Enseignement Intelligemment Assisté par Ordinateur (*p. 32*)
- EJB** Entreprise Java Bean (*p. 122*)
- Eden3** Enseignement à Distance en Ecole d'Ingénieurs en 3ème année (*p. 18*)
- ExAO** Expérimentation Assistée par ordinateur (*p. 32*)
- ExCAO** Expérimentation Collaborative Assistée par Ordinateur (*p. 53*)
- FIFO** First In, First Out (*p. 65*)
- FND** Forme Normale Disjonctive (*p. 161*)
- FOAD** (Formation Ouverte et A Distance) (*p. 23*)
- FTF** Face To Face (*p. 51*)
- GPiB** General Purpose Interface Bus (*p. 74*)
- HTML** HyperText Markup Langage (*p. 62*)
- HTTP** HyperText Transfer Protocol) (*p. 126*)

- IA** Intelligence Artificielle (*p. 33*)
- ICAI** Intelligent Computer Aided Instruction (*p. 32*)
- IETF** Internet Engineering Task Force (*p. 126*)
- IHM** Interface Homme-Machine (*p. 53*)
- IMS-LD** Instructional Management Systems – Learning Design (*p. 186*)
- IM** Instant Messaging (*p. 50*)
- IP** Internet Protocol (*p. 61*)
- ISTASE** Institut Supérieur des Techniques Avancées de Saint-Etienne (*p. 17*)
- ITS** Intelligent Tutoring Systems (*p. 32*)
- IVI** Interchangeable Virtual Instruments (*p. 75*)
- J2EE** Java 2 Enterprise Edition (*p. 118*)
- JAAS** Java Authentication and Authorization Service (*p. 113*)
- JAR** Java ARchive (*p. 97*)
- JMS** Java Messaging Service (*p. 118*)
- JORAM** Java Open Reliable Asynchronous Messaging (*p. 120*)
- JOnAS** Java Open Application Server (*p. 121*)
- JVM** Java Virtual Machine (*p. 62*)
- KBS** Knowledge Based System (*p. 84*)
- LCMS** Learning Content Management Systems (*p. 186*)
- LDAP** Lightweight Directory Access Protocol (*p. 113*)
- LED** Light-Emitting Diode (*p. 98*)
- LMS** Learning Management Systems (*p. 186*)
- MAN** Metropolitan Area Network (*p. 43*)
- MOM** Message-Oriented Middleware (*p. 76*)
- OOM** Object-Oriented Middleware (*p. 76*)
- ORB** Object Request Broker (*p. 117*)
- OUNL** Open University of the Netherlands (*p. 186*)
- OWL** Ontology Web Language (*p. 83*)
- PCI** Peripheral Component Interconnect (*p. 61*)
- PC** Personal Computer (*p. 74*)
- PHP** Hypertext Preprocessor (acronyme récursif) (*p. 62*)
- PXI** PCI eXtension for Instrumentation (*p. 61*)
- RBAC** Role Based Access Control (*p. 113*)
- RDF** Resource Description Framework (*p. 82*)
- RMI** Remote Method Invocation (*p. 116*)

-
- RPC** Remote Procedure Call (*p. 115*)
- RSS** Really Simple Syndication (*p. 82*)
- Racer** RenamedABox and Concept Expression Reasoner (*p. 139*)
- ReLAX** Remote LAboratory eXperimentation (*p. 56*)
- RuleML** Rule Markup Langage (*p. 145*)
- SBC** Systèmes à Bases de Connaissances (*p. 84*)
- SGBD** Système de Gestion de Base de Données (*p. 114*)
- SMA** Système Multi-Agents (*p. 85*)
- SOAP** Simple Object Access Protocol (*p. 83, 187*)
- STI** Systèmes de Tutorat Intelligent (*p. 32*)
- SWRL** Semantic Web Rule Language (*p. 145*)
- TCAO** Travail Collaboratif Assisté par Ordinateur (*p. 24, 47*)
- TCP** Transmission Control Protocol (*p. 61*)
- TP** Travaux Pratiques (*p. 36*)
- UDDI** Universal Description, Discovery and Integration (*p. 83*)
- UML** Unified Modeling Language (*p. 83, 115*)
- URI** Uniform Resource Identifier (*p. 84*)
- URL** Uniform Resource Locator (*p. 154*)
- USB** Universal Serial Bus (*p. 74*)
- VAE** Validation des Acquis par l'Expérience (*p. 18*)
- VB** Visual Basic (*p. 61*)
- VISA** Virtual Instrument Software Architecture (*p. 75*)
- VRML** Virtual Reality Markup Language (*p. 61*)
- VoIP** Voice over IP (*p. 63*)
- W3C** World Wide Web Consortium (*p. 83*)
- WSDL** Web Service Definition Language (*p. 83*)
- WYSIWYG** What You See Is What You Get (*p. 97*)
- XML** Extensible Markup Language (*p. 82*)
- XMPP** eXtensible Messaging and Presence Protocol (*p. 126*)
- XSD** XML Schema Description (*p. 186*)
- XUL** XML-based User interface Language (*p. 83*)
- nRQL** new Racer Query Language (*p. 160*)

Bibliographie

- AFSHARI K. & PAYANDEH S. **1999**. Toward Implementation of Java/VRML Environment for Planning, Training and Tele-Operation of Robotic Systems. *Dans : 3rd World Multiconference on systemics, Cybernetics and Informatics*. Orlando, Florida, États-Unis : IIS : International Institut of Informatics & Systemics.
- AGARWAL D. A., SACHS S. R. & JOHNSTON W. E. **1998**. The reality of collaboratories. *Computer Physics Communications*, **110**(1-3), 134–141.
- AKKOUCHE I. **1996**. *Télé-enseignement: formation professionnelle à distance et formation coopérante*. Thèse de doctorat, INSA de Lyon.
- AKTAN B., BOHUS C. A., CROWL L. A. & SHOR M. H. **1996**. Distance Learning Applied to Control Engineering Laboratories. *IEEE Transactions on Education*, **39**(3), 320–326.
- ALAMO J. A. D., HARDISSON J., MISHURIS G., BROKKS L., MCCLEAN C., CHANG V. & HUI L. **2002** (18–22 August). Educational Experiments with an Online Microelectronics Laboratory. *Dans : International Conference on Engineering Education – ICEE’02*.
- ALBU M. M., HOLBERT K. E., HEYDT G. T., GRIGORESCU S. D. & TRUSCA V. **2004**. Embedding remote experimentation in power engineering education. *IEEE Transactions on Power Systems*, **19**(1), 139–143.
- ALHALABI B. A., HAMZA M. K., HSU S. & MARCOVITZ D. M. **2000**. Remote labs: an innovative leap in the world of distance education. Orlando, Floride, États-Unis : Information Systems, Analysis and Synthesis – ISAS’00.
- ALHALABI B. A., HAMZA M. K., HSU S. & ROMANCE N. **2001**. Virtual labs vs remote labs: between myth and reality. Oslo, Norway : International Conference on Engineering Education.
- AMMARI AHMED CHIHEB & BEN HADJ SLAMA JALELEDDINE. **2006**. The development of a rmeote laboratory for Internet-based engineering education. *Journal of Asynchronous Learning Networks*, **10**(4).
- ANDERSON J. R., FINCHAM J. M. & DOUGLASS S. **1997**. The role of examples and rules in the acquisition of a cognitive skill. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, **23**(4), 932–945.

- BAUCHSPIESS A., GUIMARAES B. & GOSMANN H. **2003**. Remote experimentation on three coupled water reservoirs. *Pages 572–577 de : IEEE International Symposium on Industrial Electronics.*
- BAYARD B., FAYOLLE J., SAUVIAC B. & NOYEL G. **2001** (29–30 Octobre). Projet Web Analyzer : Instrumentation réelle sur le world web. *Dans : 3ème Colloque sur l'enseignement des technologies des sciences de l'information et des systèmes en EEA – CETISIS'01.*
- BAYARD B., SAUVIAC B. & NOYEL G. **2002** (20–22 Mars). WebAnalyzer : mesures hyperfréquences par internet. *Dans : Journées de Caractérisation Microondes et Matériaux – JCMM'02.*
- BAYARD B., FAYOLLE J., SAUVIAC B. & NOYEL G. **2004** (Septembre). A general and secure corba framework for distant control of instruments. *Dans : IFAC Workshop Internet Based Control Education – IBCE'04.*
- BECHHOFFER S., HARMELEN VAN F., HENDLER J., HORROCKS I., MCGUINNESS D. L., PATEL-SCHNEIDER P. F. & STEIN L. A. **2004**. *OWL Web Ontology Language Reference, W3C Recommendation.* Tech. rept. World Wide Web Consortium.
- BELISSARD L., PALMA N., FREYSSINET A., HERRMANN M. & LACOURTE S. **1999**. *Technical Report 14: Agent Infrastructure: The agent Anytime Anywhere Platform.* Tech. rept. Control and Coordination of Complex Distributed Services C3DS.
- BENMOHAMED HCENE. **2006**. *ICTT@Lab: un environnement informatique pour la génération et l'exécution de scénarios de téléTP.* Thèse de doctorat, ICTT (Laboratoire Interaction Collaborative Téléformation Téléactivités).
- BERNERS-LEE T. **2000**, **mise à jour en 2007**. *W3C Semantic Web Stack.*
- BERNSTEIN P. A. **1996**. Middleware: A model for distributed system services. *Communications of the ACM*, **39**, 86–98.
- BHARAMBE A. R., RAO S. & SESHAM S. **2002**. Mercury: A Scalable Publish-Subscribe System for Internet Games. Bruanschweig, Germany : 1st Workshop on Network and Systems support for games.
- BICCHI A., COPPELLI A., QUATRO F., RIZZO L., TUCHI F. & BALESTRINO A. **2001** (21–26 Mai). Breaking the Lab's Walls Tele-Laboratories at the university of Pisa. *Pages 1903–1908 de : IEEE International Conference on Robotics and Automation 2001*, vol. 2.
- BLANC C. **2004**. *Pour un écosystème de la croissance.* Rapport au premier ministre.
- BORCHERS J.O. **2001**. A pattern approach to interaction design. *Symposium on Designing Interactive Systems*, **15**, 369–378.
- BOTTAZZI E. & FERRARIO R. **2005**. A Path to an Ontology of Organizations. The Netherlands: International EDOC Workshop on Vocabularies, Ontologies and Rules for The Enterprise (VORTE 2005), Center for Telematics and Information Technology.

- BOUAUD J., BACHIMONT B., CHARLET J. & ZWEIGENBAUM P. **1994** (20–23 Septembre). Acquisition and Structuring of an Ontology within Conceptual Graphs. *Pages 1–25 de : ICCS'94 Workshop on Knowledge Acquisition using Conceptual Graph Theory.*
- BRUILLARD E. **1997**. *Les machines à enseigner*. Paris : Hermès.
- CANFORA G., DAPONTE P. & RAPUANO S. **2004**. Remotely accessible laboratory for electronic measurement teaching. *Computer Standards & Interfaces*, **26**(6), 489–499.
- CARNEVALI G. & BUTTAZZO G. **2003** (July9–11). A virtual laboratory environment for real-time experiments. *Pages 3242–3247 de : SISICAS.*
- CASINI M., PRATTICHIZZO D. & VICINO A. **2001**. The Automatic Control Telelab: a Remote Control Engineering Laboratory. Orlando, Florida, États-Unis : 40th IEEE Conference on Decision and Control.
- CEDAZO R., SANCHEZ F. M. & SEBASTIAN J. M. **2006**. Ciclope chemical: a remote laboratory to control a spectrograph. *Dans : Advances in Control Education – ACE'06.*
- CERF V. G. **1993**. *National Collaboratories: Applying Information Technology for Scientific Research*. Washinton, DC : National Academy Press.
- CHANG T., JAROONSIRIPHAN P., BERNHARDT M. & LUDDEN P. **2006**. Web-based command shaping of cobra 600 robot with a swinging load. *IEEE Transactions on Industrial Informatics*, **2**(1), 59–69.
- CHICULITA C. & FRANGU L. **2002** (14–18 Juillet). A web Based Remote Control Laboratory. *Dans : 6th Multiconference on Systemic, Cybernetics and Informatics.*
- CHIN J. P. & DIEHL, V. A. / NORMAN K. L. **1988**. Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface Interface Evaluations.
- CHOY G., PARKER D. R., D'AMOUR J. N. & SPENCER J. L. **2000**. Remote experimentation: a Web-operable two-phase flow experiment. *Pages 2939–2943 de : American Control Conference*, vol. 4.
- COLWELL E., CHETZ C. & SCANLON M. **2002**. Using remote laboratories to extend access to science and engineering. *Computers & Education*, **38**(1-3), 65–76.
- CORTER J. E., NICKERSON JEFFREY V., ESCHE S. K. & CHASSAPIS C. **2004**. Remote versus hands-on labs: A comparative study. Savannah, Georgie, États-Unis : 34th ASEE/IEEE Frontiers in Education Conference.
- DELAMER I. M. & LASTRA J. L. M. **2006**. Service-Oriented Architecture for Distributed Publish/Subscribe Middleware in Electronics Production. *IEEE Transactions on Industrial Informatics*, **2**(4), 281–294.
- DEWIYANTIA SILVIA, BRAND-GRUWELA SASKIA, JOCHEMS WIM & BROERSB NICK J. **2007**. Students' experiences with collaborative learning in asynchronous Computer-Supported Collaborative Learning environments. *Computers in Human Behavior*, **23**, 496–514.

- DILLENBOURG P. **1999**. What do you mean by « collaborative learning »? *Amsterdam, NL: Pergamon, Elsevier Science, Elsevier Science*, 1–16.
- DIXON W. E., DAWSON D. M., COSTIC B. T. & QUEIROZ M. S. D. **2002**. MATLAB-Based Control Systems Laboratory Experience for Undergraduate Students: Towards Standardization and Shared Resources. *IEEE Transactions on Education*, **45**(2), 218–226.
- DOBROGOWSKI W., MAZIEWSKI A. & ZABLOTSKII V. **2007**. Remote teaching experiments on magnetic domains in thin films. *European Journal of Physics*, **28**, 71–83.
- DRUMMOND N. **2005**. *CS646: OWL Tutorial (session 1)*. Tech. rept. Cooperative Ontologies Program.
- ELLIS A. **2003**. Personality type and participation in networked learning environments. *Educational Media International*, **40**(1-2), 101–114.
- ESCHE S. K. **2002** (1–4 Octobre). Remote Experimentation - One building Block in Online Engineering Education. *Dans: ASEE/SEFI/TUB International Colloquium on Global Changes in Engineering Education*.
- EUGSTER P. T., GUERRAOU R. & SVENTEK J. **2000**. Distributed Asynchronous Collections: Abstractions for Publish/Subscribe Interactions. *Lectures notes in Computer Science, Proc. of 14th European Conference on Object-Oriented Programming*, **1850**, 252–276.
- EUGSTER P. T., FELBER P. & GUERRAOU R. **2001**. *The Many Faces of Publish/Subscribe*. Tech. rept. EPFL, Lausanne, Switzerland.
- FALTIN N., BÖHNE A., TUTTAS J. & WAGNER B. **2002**. Distributed Team Learning in an Internet-Assisted Laboratory. Manchester, U.K: International Conference on International Conference on Engineering Education.
- FAYOLLE J. **2004**. A pragmatic approach of e-learning and virtual university experiences and strategic points of view on the virtual campus Eden3. *Information Technologies in Higher Education*, **1**(1), 51.
- FERRERO A., SALICONE S., BONORA C. & PARMIGIANI M. **2003**. ReMLab: a Java-based remote, didactic measurement laboratory. *IEEE Transactions on Instrumentation and Measurement*, **52**(3), 710–715.
- FINHOLT T. A. & OLSON G. M. **1997**. From Laboratories to collaboratories: A new organizational form for scientific collaboration. *Psychological Science*, **8**(1), 28–36.
- GRAVIER C. & FAYOLLE J. **2005**. A Composite Web Service supporting User Context to provide an Adapted Remote Control of High Technological Instruments. *Elsevier Electronic Notes on Theoretical Computer Science*, **146**(1), 17–32.
- GRAVIER C. & FAYOLLE J. **2006a**. E-instrumentation: a collaborative and adaptative HCI for the remote control of devices. Paphos, Chypre: 8th International Conference on Enterprise Information Systems (ICEIS'2006).

- GRAVIER C. & FAYOLLE J. **2006b**. A Generic Distant and Collaborative Executive System in Extended Enterprise. vol. 1. Saint-Etienne, France : 12th IFAC Symposium on Information Control Problems in Manufacturing (INCOM'2006).
- GRAVIER C., FAYOLLE J., NOYEL G., LELEVÉ A. & BENMOHAMED H. **2006a**. Closing the Gap between Remote Labs and Learning Management Systems. Hammamet, Tunisia: the 1st International Conference on E-Learning in Industrial Electronics (ICELIE'2006).
- GRAVIER C., FAYOLLE J., LARDON J. & NOYEL G. **2006b**. A Distributed Online Laboratory System for Distant Learning. Hammamet, Tunisia: The International Conference on Signal-Image Technology and Internet-Based Systems (SITIS'2006).
- GREER J., MCCALLA G., COLLINS J., KUMAR V., MEAGHER P. & VASSILEVA J. **1998**. Supporting Peer Help and Collaboration in Distributed Supporting Peer Help and Collaboration in Distributed Workplace Environments. *International Journal of Artificial Intelligence in Education*, **9**, 159–177.
- GRUBER T. R. **1993**. *Towards Principles for the Design of Ontologies Used for Knowledge Sharing*. Deventer, The Netherlands: Kluwer Academic Publishers.
- GRUBER T. R., GRUNINGER M., HAYES P., MC GUINNESS D. & OBRST L. **2007**. *Ontology Framework Draft*. Tech. rept. OntologySummit2007: Frameworks For Consideration.
- GRUBER T.R. **2007**. Grande Challenges for Ontology Design (or is it Vente?). Invited Presentation to Ontolog, <http://tomgruber.org/writing/challenges-for-ontology-design.htm>.
- GUARINO N. **1998**. Formal Ontology and Information Systems. Trento, Italy: the 1st International Conference on Formal Ontologies in Information Systems (FOIS'98).
- GUARINO N. **2006**. Formal Ontology, Knowledge Representation and Conceptual Modelling: Old Inspirations, Still Unsolved Problems. **4068**.
- GUARINO N. & GIARETTA P. **1995**. Ontologies and Knowledge Bases: Towards a Terminological Clarification. *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing*, IOS Press, Amsterdam, 25–32.
- GUIMARAES E., MAFFEIS A., PEREIRA J., RUSSO B., CARDOZO E., BERGERMAN M. & MAGALHAES M. F. **2003**. REAL: a virtual laboratory for mobile robot experiments. *IEEE Transactions on Education*, **46**(1), 37–42.
- GUSTAVSSON I. **2002** (April17–19). Remote laboratory experiments in electrical engineering education. *Pages 1–5 de: 4th International Caracas Conference on Devices, Circuits and Systems*.
- GUTTERMAN L. **2004** (20–23 Septembre). Integrating VISA, IVI and ATEasy to migrate legacy test systems. *Dans: AUTOTESTCON 2004*.

- HIGHTOWER R. & SAYEED L. **1995**. The Impact of Computer-Mediated Communication Systems on Biased Group Discussion. *Computers in Human Behavior*, **11**(1), 33–44.
- HOBBS J.R. & PAN F. **2004**. An Ontology of Time for the Semantic Web. *ACM Transactions on Asian Language Processing (TALIP): Special issue on Temporal Information Processing*, **3**(1), 66–85.
- HOBMAN E. V. **2002**. The Expression of Conflict in Computer-Mediated and Face-To-Face Groups. *Small Group Research*, **33**(4), 439–465.
- HODGE H., HINTON H. S. & LIGHTNER M. **2001**. Virtual Circuit Laboratory. *Journal of Engineering Education*, **90**(4), 507–511.
- HOYER H., JOCHHEIM A., RÖHRIG C. & BISCHOFF A. **2004**. A Multiuser Virtual-Reality Environment for a Teleoperated Laboratory. *IEEE Transactions on Education*, **47**(1), 121–126.
- HRASTINSKI S. **2006**. Introducing an informal synchronous medium in a distance learning course: How is participation affected? *The Internet and Higher Education*, **9**, 117–131.
- HSU S., ALHALABI B. A. & ILYAS M. **2000**. A Java-Based Remote Laboratory for Distance Learning. International Conference on Engineering Education.
- HUA J. & GANZ A. **2003** (November5–8). Web enabled remote laboratory (R-Lab) Framework. *Page T2C de : 33rd ASEE/IEEE Frontiers in Education Conference*.
- JOHNSON DAVID W. & JOHNSON ROGER P. **1987**. *Learning Together and Alone: Cooperative, Competitive, and Individualistic Learning*. Prentice Hall.
- JONASSEN D. & WANG S. **1993**. Acquiring structural knowledge from semantically structured hypertext. *Journal of Computer-Based-Instruction*, **20**, 1–8.
- KAHRS M. **2003**. Applications of rf and microwave sampling to instrumentation and measurement. IEEE MTT-S International Microwave Symposium Digest.
- KIKUCHI T., KENJO T. & FUKUDA S. **2001**. Remote laboratory for a brushless DC motor. *IEEE Transactions on Education*, **44**(2).
- KNIGHT C. D. & DEWEERTH S. P. **1997**. A Distance Learning Laboratory for Engineering Education. Milwaukee, WI, États-Unis : ASEE Annual Conference & Exposition.
- KNUBLAUCH H. **2004**. Ontology-Driven Software Development in the Context of the Semantic Web: An Example Scenario with Protégé OWL. 1st International Workshop on the Model Driven Semantic Web (MDSW2004).
- KOPER R. & OLIVIER B. **2004**. *Representing the Learning Design of Units of Learning*. Vol. 7. Pages 97–111.
- KOSCHMANN T. **2002**. Dewey’s contribution to the foundations of CSCL research. *Computer support for collaborative learning: Foundations for a CSCL community: Proceedings of CSCL 2002*, 17–22.

- KOUZES R. T., MEYERS J. D. & WULF W. A. **1996**. Collaboratories: Doing science on the Internet. *IEEE Computer*, **29**(8), 40–46.
- KREIJNS K., KIRSCHNER P. A. & JOCHEMS W. **2002**. The Sociability of Computer-Supported Collaborative Learning Environments. *Educational Technology & Society*, **5**(1), 822–837.
- LANDRAGIN J.-P. & SCHWARTZ P. **2006** (7 Mai). *L'instrumentation PXI*.
- LAPIQUE F. **2006**. *Le langage d'ontologie Web OWL*. Tech. rept. École Polytechnique Fédérale de Lausanne, Flash Informatique.
- LATCHMAN H. A., SALZMANN CH. & GILLET D. **1999**. Information Technology Enhanced Learning in Distance and Conventional Education. *IEEE Transactions on Education*, **42**(4), 247–254.
- LELEVÉ A., PRÉVOT P., BENMOHAMED H. & BENADI M. **2004**. Generic e-lab platforms and e-learning standards. Grenoble, France: Computer Aided Learning In Engineering Education – CALIE'04.
- LEWIS J. R. **1995**. IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use. *International Journal of Human-Computer Interaction*, **7**(1), 57–78.
- LUKOSCH S., ROTH J. & UNGER C. **1999** (Septembre). Marrying on-campus teaching to distance teaching. *Dans: 19th World Conference on Open Learning and Distance Education*.
- MA J. & NICKERSON J. V. **2006**. Hands-on, simulated, and remote laboratories: A comparative literature review. *ACM Computing Surveys (CSUR)*, **38**(3), 1–24.
- MAFFEIS S. **1997**. *iBus: The Java Intranet Software Bus*. Tech. rept. TR, Softwired A.G.
- MAGIN D. J. **1982**. Collaborative peer learning in the laboratory. *Studies in Higher Education*, **7**(2), 105–117.
- MALAN G. R., JAHAMIA F. & SUBRAMANIAN S. **1997**. Salamander: A push-based Distributed Substrate for Internet Applications. USENIX Symposium on Internet Technologies and Systems.
- MAO C. & BACON J. **1998**. COBEA: A CORBA-based Event Architecture. New Mexico: 4th USENIX Conference on Object Oriented Technologies and Systems (COOTS).
- MARIOTTI F. **2005**. *Qui gouverne l'entreprise en réseau?* Paris: Presses de Sciences Po.
- MARTÍNEZ CARRERAS M. A., GÓMEZ SKARMETA A. F., MARTÍNEZ GRACIÁ E. & MORA GÓNZALEZ M. **2005**. COLAB: A platform Design for Collaborative Learning in Virtual Laboratories. *Technology enhanced learning*, 95–109.
- MENDELSON E. **1997**. *Introduction to Mathematical Logic, 4th ed.* London: International Thomson Publishing. Page 440.

- MILTERSEN P. B., RADHAKRISHNAN J. & WEGENER I. **2003**. On converting CNF to DNF. *Electronic Colloquium on Computational Complexity*.
- MOORE M. G. **1989**. Three Types of Interaction. *The American Journal of Distance Education*, **3**(2), 1–6.
- MÜNZER S. **2007**. An evaluation of synchronous co-operative learning in the field: The importance of instructional design. *Educational Media International*, **40**(1/2), 91–100.
- NICHOLSON SCOTT. **2002**. Socialization in the virtual hallway: Instant messaging in the asynchronous Web-based distance education classroom. *The Internet and higher education*, **5**(4), 363–372.
- NIELSEN J. **1993**. *Usability Engineering*. Morgan Kaufmann.
- NONNON P. **2005**. L'ExAO: bilan, développements récents et perspectives. *Skholé*, 15–20.
- NORTHRUP P. **2001**. A Framework for Designing Interactivity into Web-based Instruction. *Educational Technology*, **41**(2), 31–39.
- OBERLE D., LAMPARTER S., GRIMM S., VRANDECIC D., STAAB S. & GANGEMI A. **2006**. Towards Ontologies for Formalizing Modularization and Communication in Large Software Systems. *Journal of Applied Ontology*, **1**(2), 163–202.
- O'CONNOR M., KNUBLAUCH H., TU S., GROSOFF B., DEAN M., GROSSO W. & MUSEN M. **2005a**. Supporting Rule System Interoperability on the Semantic Web with SWRL. *Lecture notes in computer science*, **3729**, 974–986.
- O'CONNOR M., KNUBLAUCH H., TU S. & MUSEN M.A. **2005b**. Writing Rules for the Semantic Web Using SWRL and Jess. Madrid, Espagne: 8th International Protégé Conference, Protégé with Rules Workshop.
- OKI B., PFLUEG M., SIEGEL A. & SKEEN D. **1993**. The Information Bus: An Architecture for Extensible Distributed Systems. *Operating Systems review*, **27**(5), 58–68.
- PERAYA D., VIENS J. & KARSENTI T. **2002**. Formation des enseignants à l'intégration pédagogique des TIC, Esquisse historique des fondements, des recherches et des pratiques. *Revue des sciences de l'éducation*, **28**(2), 243–264.
- PIGUET YVES, GILLET DENIS, TOMATIS NICOLA & SIEGWART ROLAND. **2002**. Environment for online experimentation and analysis. *Dans: Educational Applications of Online Robots Workshop*. Washington DC, États-Unis: IEEE International Conference on Robotics and Automation – ICRA'02.
- POZAR D. M. **2005**. *Microwave Engineering, Troisième édition*. John Wiley & Sons, Inc.
- RECTOR A., DRUMMOND N., HORRIDGE M., ROGERS J., KNUBLAUCH H., STEVENS R., WANG H. & WROE C. **2004**. OWL Pizzas: Practical Experience of Teaching

- OWL-DL: Common Errors & Common Patterns. *Pages 63–81 de : Engineering Knowledge in the Age of the SemanticWeb*, vol. 3257. Berlin / Heidelberg : Springer.
- ROHRIG C. & JOCHHEIM A. **1999**. The virtual lab for controlling real experiments via Internet. *Pages 279–284 de : the 2001 American Control Conference*. Kohala Coast Island of Hawaii, Hawaii : IEEE International Symposium on Computer Aided Control System Design – ISCACSD’99.
- ROMANO N. C. JR., SHARDA R. & LUCCA J. **2005**. Computer-supported collaborative learning requiring immersive presence: An introduction. *Information Systems Frontiers*, **7**(1), 5–12.
- ROOT R. W. & DRAPER S. **1983**. Questionnaires as a Software Evaluation Tool. Madrid, Espagne : ACM CHI’83 Conference on Human Factors in Computing Systems.
- ROUILLER T. **2006**. *Isolateur coplanaire 40-60 GHz à couche mince magnétique*. Thèse de doctorat, Laboratoire DIOM, Université Jean Monnet de Saint-Etienne.
- ROUSSET M.C., MCILRAITH S.A., DIMITRIS P. & VAN HARMELEN F. **2004**. Small Can Be Beautiful in the Semantic Web. International Semantic Web Conference.
- SANCHEZ J., DORMIDO S., PASTOR R. & MORILLA F. **2004**. A Java/Matlab-based environment for remote control system laboratories: Illustrated with an inverted pendulum. *IEEE Transactions on Education*, **47**(3), 321–329.
- SCHWARTZ T. L. & DUNKIN B. M. **2000**. Facilitating interdisciplinary hands-on learning using LabView. *International Journal of Engineering Education*, **16**(3), 218–227.
- SERE M.G., LEACH J., NIEDDERER H., PSILLOS D. & TIBERGHEN A. **1998**. *Improving Science Education: Issues and Research on Innovative Empirical and Computer-Based Labwork in Europe*. Tech. rept. La Commission Européenne, action de recherche socio-économique, Recherche sur l’Education et l’Apprentissage.
- SIEGEL J. **1999**. An Overview of Corba 3. Boston, États-Unis : 2nd IFIP International Working Conference: Distributed Application and Interoperable Systems (IDAIS 1999).
- SMITH P. R. & POLLARD D. **1986**. The Role of Computer Simulations in Engineering Education. *Computer Education*, **10**(3), 335–340.
- SNAE C. & BRUECKNER M. **2007**. Ontology-Driven E-Learning System Based on Roles and Activities for Thai Learning Environment. *Interdisciplinary Journal of Knowledge and Learning Objects*, **3**, 1–17.
- STAHL G. **2002**. Rediscovering CSCL. *Pages 169–181 de : KOSCHMANN T., HALL R. & MIYAKE N. (eds), CSCL 2: Carrying forward the conversation*.
- STAHL G., KOSCHMANN T. & SUTHERS D. **2006**. CSCL: An historical perspective. *Cambridge Handbook of the Learning Sciences*, 409–426.
- TABOY J. P. **2006**. A community sharing hands-on centers in engineer’s training. *International Journal of Online Engineering*, <http://www.i-joe.org>, **2**(1).

- TANENBAUM ANDREW S. & WOODHULL ALBERT S. **2006**. *Operating Systems Design and Implementation*. 3/e edn. Prentice Hall.
- TANNENBAUM A. S. **1988**. A Critique of the Remote Procedure Call Paradigm. Vienna, Austria : EUTECO '88.
- TAVRIS C., WADE C., GAGNON A., GOULET C. & WIEDMANN P. **1999**. *Introduction à la psychologie : Les grandes perspectives*. De Boeck Université.
- TCHOUNIKINE P. **2002a**. *Conception des environnements informatiques d'apprentissage : mieux articuler informatique et sciences humaines et sociales*. INRP - MSH - IUFM de Basse Normandie. Pages 203–210.
- TCHOUNIKINE P. **2002b**. *Quelques éléments sur la conception et l'ingénierie des EIAH*. Tech. rept. GDR I3.
- TREVELYAN J. **2003**. Remote Laboratories and Team Skills in Mechatronics. Perth, Australie : Mechatronics and Machine Vision in Practice Conference.
- TUTTAS J. & WAGNER B. **2001**. Distributed Online Laboratories. Oslo, Norvège : International Conference on Engineering Education – ICEE'01.
- USCHOLD M. **1998**. Knowledge level modelling: Concepts and terminology. *Knowledge Engineering Review*, **13**(1).
- VEERMAN A. L., ANDRIESEN J. E. B. & KANSELAAR G. **2004**. Learning through synchronous electronic discussion. *Computer & Education*, **34**(3-4), 269–290.
- VEDMA G., DANCY I. J. & LUNDBERG K. H. **2005**. A Web-based linear-systems iLab. Portland, Oregon, États-Unis : American Control Conference – ACC'05.
- WULF W. A. **1989**. *The national collaboratory*.
- YAN Y., LIANG Y., DU X., SALIAH-HASSANE H. & GHORBANI A. **2006**. Putting Labs Online with Web Services. *IT Professional*, **8**(2), 27–34.
- YANG X., PETRIU D. C., WHALEN T. E. & PETRIU E. F. M. **2004**. A Web-based 3D virtual robot remote control system. *Canadian Conference on Electrical and Computer Engineering*, **2**(2-5), 955–958.
- YONEKI E. **2003**. Mobile Applications with a Middleware System in Publish Subscribe Paradigm. Bern, Switzerland : 3rd Workshop on Applications and Services in Wireless Networks.
- ZHU S. **2006**. The application of an Internet-based laboratory for Distance Learning. Vienne, Autriche : 10th IACEE World Conference on Continuing Engineering Education (WCCEE).
- ZIMMER T., GEOFFROY D., BILLAUD D. & DANTO Y. **2003**. The eLab for electrical engineering education. Marrakech, Maroc : 4th International Conference on Information Technology Based Higher Education and Training.

143 références.