

Vers une Recherche Coopérative dans les Systèmes de Recherche d'Informations

Amos A. DAVID
CRIN (Centre de Recherche en Informatique de Nancy)
BP 239, Vandoeuvre Cedex,
FRANCE
email: adavid@loria.fr

Mots-clés: recherche coopérative, modélisation de l'utilisateur, personnalisation de réponses, communication inter-processus, inter-opérabilité

Résumé

Notre objectif dans le cadre de cette recherche est de permettre à deux utilisateurs de mener ensemble une recherche d'informations, sur des ordinateurs distants quelle que soit l'architecture de ces ordinateurs, et en temps réel. Ces travaux se présentent sous deux axes: d'une part, la modélisation de l'utilisateur dans un système de recherche d'informations (SRI) pour étudier la façon de représenter un utilisateur de manière explicite dans un SRI, les connaissances sur l'utilisateur à représenter, comment acquérir ces connaissances et comment les exploiter; d'autre part, la communication inter-processus dans les SRI pour permettre la recherche coopérative. Dans cet axe, nous étudions les outils logiciels nécessaires et les apports de la connaissance sur un utilisateur pour un autre dans ce contexte de recherche coopérative. Le prototype BIRDS que nous avons développé, un SRI qui a été utilisé pour un objectif pédagogique, nous permet actuellement d'expérimenter nos concepts pour une recherche coopérative d'informations mettant en jeu deux utilisateurs.

Summary

Our objective within the framework of this research studies is to allow two users to conduct information research together, on distant machines with different architecture and in real time. These studies cover two research axis: the first axe concerns user modelling in information research systems (IRS) to study the explicit representation of a user in an IRS, the knowledge on the user to represent, how to acquire this knowledge and how to exploit it; the second axe concerns inter-process communication in IRS to allow cooperative information research. In this axe, we are studying the necessary software tools and the contribution of the knowledge acquired on a user for the other user in the context of cooperative information research. The prototype BIRDS that we developed, an IRS, that was already used for a pedagogical objective, allows us to experiment our concepts for cooperative information research.

I. Introduction

Nous présentons dans ce papier nos travaux afin de rendre les systèmes de recherche d'informations encore plus efficaces et plus accessibles pour les utilisateurs. Notre objectif dans le cadre de cette recherche est de *permettre à deux utilisateurs de mener ensemble une recherche d'informations, permettre aux utilisateurs de mener la recherche sur des ordinateurs distants quelle que soit l'architecture de ces ordinateurs, et mener la recherche en temps réel.*

Ces travaux se présentent sous deux axes de recherche. Le premier axe porte sur la mo-

délimitation de l'utilisateur dans un SRI [RUL95]. Pour cet axe, les problèmes que nous étudions concernent la façon de représenter un utilisateur de manière explicite dans le SRI, ou encore les connaissances sur l'utilisateur que l'on veut représenter dans le système, comment acquérir ces connaissances et comment les exploiter. Pour que deux utilisateurs puissent mener fructueusement ensemble une recherche d'information, nous croyons qu'une connaissance mutuelle des deux utilisateurs est nécessaire. Le SRI pourra fournir des informations sur un utilisateur à la demande de l'autre utilisateur. Cela nécessite une représentation explicite des utilisateurs. Nous reviendrons sur cet axe dans la section II.

Le deuxième axe de recherche porte sur la communication inter-processus dans les SRI pour permettre la recherche coopérative de deux utilisateurs. Les problèmes de cet axe nous amènent à étudier les outils logiciels nécessaires à la communication inter-processus pour permettre la recherche coopérative. Nous étudions également les apports des connaissances sur un utilisateur pour un autre utilisateur (ou un expert humain) dans un contexte d'une recherche coopérative. Cet axe est développé dans la section III.

II. Représentation explicite de l'utilisateur dans un SRI

II.1 Pourquoi une représentation explicite?

L'objectif principal d'un SRI est de donner des réponses pertinentes à l'utilisateur. La pertinence d'une réponse concerne l'exactitude de la réponse par rapport à la demande formulée par l'utilisateur [ING92]. Mais la pertinence d'une réponse concerne également l'adéquation de la réponse à l'utilisateur, par exemple l'adéquation de la réponse au niveau de connaissance de l'utilisateur et à ses préférences. C'est ce deuxième type de pertinence que nous étudions.

Pour mieux comprendre ce que nous avons défini comme objectif à réaliser dans les SRI, nous allons prendre la collection d'informations d'une agence de tourisme comme exemple. Prenons l'exemple d'un client qui garde la même agence pour organiser ses voyages. Pour son prochain voyage, il veut des informations pour se rendre dans les îles Caraïbes. Si le client s'adresse à une hôtesse de l'agence qui *le connaît bien*, les informations que l'hôtesse donnera au client ne seront pas les mêmes que celles données par une hôtesse ne le connaissant pas. Par exemple, l'hôtesse qui connaît bien le client sait quelles sont les îles où le client a déjà passé des vacances, les problèmes rencontrés, les satisfactions obtenus par le client lors de ses séjours dans ces îles. En bref, l'hôtesse possède des connaissances spécifiques sur le client en question.

Dans la même agence, on peut avoir d'autres types de client. Par exemple on peut avoir un client qui part en voyage pour la première fois ou un client qui a déjà beaucoup voyagé mais qui veut se rendre dans les îles Caraïbes pour la première fois. Chacun de ces types d'utilisateur a besoin d'informations bien spécifiques à son cas.

Si nous devons mettre en place un SRI pour fournir les informations aux clients de l'agence, il est indispensable d'avoir une représentation explicite de ces clients [ING92, pg 23]. Des informations exactes ne suffisent pas. Une information exacte ne correspond pas forcément à la préférence du client. Et même la préférence d'un client peut être ce qu'un autre client déteste. D'où la nécessité d'une représentation explicite de chaque utilisateur.

II.2 Quoi représenter?

Nous avons adopté un modèle cognitif pour représenter l'utilisateur. Ce modèle permet de représenter le niveau de connaissance de l'utilisateur et ses préférences. Le modèle est basé sur les différentes phases cognitives d'un utilisateur en situation d'apprentissage, car comme le dit

Cluzeau-Ciry[CLU88], «les demandes des non-spécialistes correspondent au besoin culturel d'un autodidacte». Nous pouvons identifier quatre phases cognitives d'un utilisateur en situation d'apprentissage [DAV91]:

La phase d'observation: Dans cette phase, l'apprenant prend connaissance du domaine étudié par le processus d'observation. Pour un utilisateur en recherche d'information, cette phase correspond à une simple consultation. Par rapport à notre agence de voyage, on peut présenter des séquences vidéos à l'utilisateur.

La phase d'abstraction élémentaire: Dans cette phase, l'apprenant acquiert le vocabulaire du domaine, désigne des objets et commence ainsi un processus élémentaire d'abstraction. Pour un utilisateur en recherche d'information, cette phase correspond à des demandes en information précises ou des demandes thématiques. Par rapport à notre agence de voyage, le client peut demander ce que signifie le terme «formule d'un voyage».

La phase de symbolisation et de raisonnement: Dans cette phase, l'apprenant emploie un vocabulaire spécialisé du domaine, il est capable de développer des idées abstraites et de raisonner avec des connaissances déjà acquises, soit pour exprimer ses connaissances, soit pour résoudre des problèmes. Cette phase correspond pour un utilisateur en recherche d'information, à des demandes connotatives. Par rapport à notre agence de voyage, le client peut demander des informations précises sur une formule de voyage avec ses préférences.

La phase de créativité: Dans cette phase, l'apprenant exploite ses connaissances pour en développer d'autres qui ne lui ont pas été transmises. Pour un utilisateur en recherche d'information, cette phase correspond à des jugements émis par l'utilisateur, par exemple pour exprimer ses préférences à la suite d'une analyse des réponses proposées par le système. Par rapport à notre agence de voyage, le client peut donner à l'hôtesse de l'agence ses préférences en fonction de ses analyses des voyages précédents.

Dans la réalisation du modèle, nous observons les activités de l'utilisateur dans ces phases. Nous observons également les termes utilisés dans chacune des phases. Pour une plus grande efficacité, ces observations doivent avoir lieu dans le temps et ne pas être limitées à une seule session d'utilisation du système.

II.3 Comment exploiter les connaissances sur l'utilisateur?

Nous travaillons sur deux approches pour l'exploitation des connaissances sur l'utilisateur [RUL95]. La première approche concerne l'exploitation par le système pour mieux comprendre ce que veut l'utilisateur et adapter la réponse à son niveau de connaissance (le niveau déduit par le système). La deuxième approche concerne la présentation d'une synthèse d'information sur l'utilisateur à un expert ou à un autre utilisateur, pour mieux orienter le processus de la recherche d'information. C'est cette deuxième approche qui nécessite une recherche coopérative et les outils de communication inter-application que nous développons dans la section III.

Une troisième approche d'exploitation possible, intéressant pour les chercheurs qui travaillent sur l'apprentissage machine, concerne l'analyse des observations faites pendant la médiation d'un expert. Le système acquiert des connaissances sur l'utilisateur. L'expert qui joue un rôle de médiation propose des solutions. Les critères durables des solutions proposées par l'expert qui satisfont la plupart des utilisateurs peuvent être intégrés comme de nouvelles connaissances dans le système.

III. Communication inter-application

Nous presentons ci-dessous les définitions des termes que nous utilisons dans le cadre de ce travail.

Un processus, une application: Nous considérons un *processus* p comme un ensemble de processus élémentaires p_j fonctionnant d'une manière coordonnée. Une *application* est pour nous un ensemble de processus. Le déroulement de l'application est directement ou indirectement, partiellement ou totalement contrôlé par un utilisateur.

Un serveur, un serveur d'application, un serveur de localisation: Un *serveur* est pour nous un processus particulier dont le rôle est de permettre l'échange d'informations entre les processus. Un *serveur d'application* (SA) est un processus serveur attaché à une application, lui assurant la communication avec les moyens de communication du système informatique. Son rôle principal est de réceptionner des messages destinés à l'application et les communiquer à l'application pour le traitement nécessaire. Un *serveur de localisation* (SL) a pour rôle de tenir à jour la liste des processus et applications disponibles, et leurs caractéristiques.

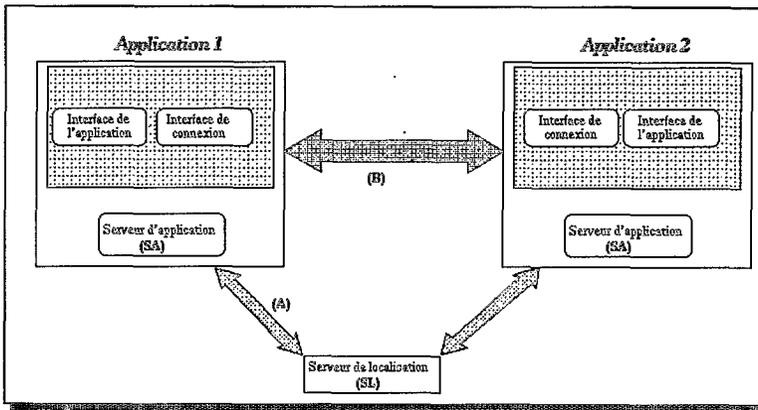


Figure 1. Architecture de la communication inter-application

Une interface de connexion: L'interface de connexion est un module de l'application qui permet de réaliser la connexion et le contrôle de la connexion avec un autre utilisateur. Il réalise le protocole de connexion et de déconnexion d'un utilisateur et traite la fin d'une session de l'utilisateur.

Après le lancement d'une application, l'application commence par s'enregistrer dans le SL (A). Une application qui veut communiquer avec une autre sélectionne l'application par l'interface de connexion. Toutes les informations nécessaires à la communication directe (B) entre les applications sont disponibles après l'établissement de la connexion. Tous les messages sont réceptionnés par les serveurs d'application.

Les niveaux de communication: Conceptuellement, nous avons adopté une communication inter-application par couches [COR91]: couche communication, couche gestion et couche application. La couche communication est fortement liée au système d'exploitation et à leurs fonctions de communication inter-processus. La couche gestion sert de lien entre la couche communication et la couche application. Les outils de cette couche assurent les transformations nécessaires aux données reçues des autres applications et aux données à transmettre aux autres applications. La couche application sert de lien entre les interfaces de l'application et la couche de gestion. Nous avons

adopté cette méthode de communication par couche pour permettre une simplification des tâches de développement des applications. En effet, ce qu'intègre le développeur à son application est constitué des outils de la couche application. A noter qu'il y a d'autres propositions qui reposent sur quatre ou cinq couches [WAR95].

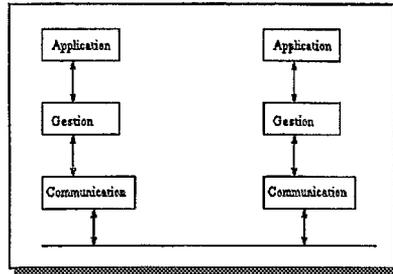


Figure 2. Les niveaux de communication

La communication inter-application repose sur le concept de communication par message. Nous ne nous intéressons pas pour le moment à l'échange de données importantes directement par les outils de communication que nous avons développés. Les données de grandes tailles sont censées être accessibles par toutes les applications impliquées. Nous utilisons par exemple des fichiers sur un réseau local pour notre prototype. Tous les messages entre les applications sont de la forme *<commande> [liste de paramètres]*. Les commandes sont traduites par l'application destinataire. Les outils du niveau communication que nous avons développés reposent sur l'utilisation de sockets [GAB92]. Ces outils sont maintenant disponibles dans les langages C, Small-Talk et Tcl/Tk et se présentent actuellement sous forme de bibliothèques de fonctions.

IV. Modèles de communication et types d'application

Une recherche coopérative nécessite la mise en place de moyens de dialogue entre les deux utilisateurs. Le dialogue passe obligatoirement par les moyens de communications disponibles. Nous présentons dans la suite de cette section notre étude sur les modèles de communication et leurs applications dans notre cadre de travail.

IV.1 Modèles de communication

On peut noter deux modèles ou types de protocole de communication entre des applications: une communication asynchrone et une communication synchrone

IV.1.1. Un protocole de communication asynchrone

Dans un protocole de communication asynchrone, les applications des deux utilisateurs en communication se déroulent d'une manière asynchrone. Ce protocole ressemble à un système de producteur-consommateur où le producteur produit à son rythme et le consommateur consomme à son rythme également.

Nous nous sommes servi de ce type de protocole pour un projet européen FORCE qui concerne la gestion de la commercialisation des fruits et légumes. Il y a trois principaux acteurs dans ce système: la coopérative vendeur qui représente les agriculteurs producteurs, le distributeur acheteur, et le technicien contrôleur qui vérifie la qualité des produits proposés au marché. La proposition des produits à vendre se fait au rythme de l'agriculteur. Le technicien vérifie les produits

à son rythme également et notifie au producteur son jugement sur le produit. L'acheteur scrute les produits pour choisir ceux qui l'intéressent. Il notifie au producteur son intérêt pour le produit. Dans le prototype que nous avons réalisé, la communication entre le producteur et l'acheteur se fait d'une manière asynchrone car les prix et les négociations sont contrôlés par la coopérative. Mais on peut envisager un cas où l'acheteur serait amené à négocier directement avec le producteur et cela en temps réel. Dans ce cas nous serons dans un contexte de communication synchrone.

IV.1.2. Un protocole de communication synchrone

Un protocole de communication synchrone permet de réaliser un dialogue entre deux ou plusieurs utilisateurs. Ce type de protocole est nécessaire dans des situations où les utilisateurs doivent coopérer pour résoudre un problème. Les types de problème à résoudre qui nécessitent un dialogue entre deux ou plusieurs utilisateurs sont nombreux. La négociation entre un producteur et un acheteur, l'assistance d'un expert dans un domaine pour rechercher des informations, le guidage d'un élève par son professeur dans un processus d'apprentissage, la résolution d'un problème par deux élèves, les jeux à deux ou plusieurs, etc. sont quelques exemples où la communication de type synchrone est nécessaire.

Le prototype BIRDS que nous présentons dans la section VI emploie le type de communication synchrone. Nous présentons dans la section suivante comment les deux types de protocole de communication peuvent servir dans un contexte d'apprentissage, ou dans un contexte de recherche d'information.

IV.2 Les modèles de communication et les types d'application

Observation du déroulement d'un processus d'apprentissage ou de recherche d'information

Un avantage principal d'une communication inter-application, en mode communication asynchrone est de permettre à un utilisateur d'observer le déroulement de la session d'un autre utilisateur sans intervenir dans le déroulement. Ainsi, un professeur pourra observer le travail d'un élève et un élève pourra observer comment un problème est résolu par le professeur ou par un autre élève. Il en est de même pour un expert et un utilisateur en recherche d'information, ou entre deux utilisateurs.

Participation au déroulement d'un processus d'apprentissage ou de recherche d'information

L'observation du déroulement d'un processus de résolution d'un problème ne suffit pas dans certaines situations. Dans ces cas, l'intervention en temps réel sur le processus par le professeur ou par l'expert est nécessaire. L'intervenant agit sur le déroulement au moment précis du besoin. Ce type de d'application nécessite une communication de type synchrone.

V. Les types de contrôle en communication inter-application

Un système de communication inter-application doit protéger aussi bien les utilisateurs que les objets des applications. Nous distinguons actuellement deux principaux types de contrôle: des contrôles au niveau de connexion sur l'application d'un utilisateur et des contrôles au niveau des objets de l'application de l'utilisateur.

Contrôle sur la connexion: Un contrôle au niveau de la connexion doit permettre à un utilisateur de se protéger contre une connexion non autorisée, même si l'utilisateur s'est enregistré dans le système de communication inter-application. Le protocole de connexion (demande et autorisation) est réalisé par l'interface de contrôle.

Contrôle sur les objets d'une application: Un objet d'une application peut être:

- local-privé* : l'objet est accessible uniquement par l'application
- local-commun* : l'objet est accessible par les applications qui sont en communication
- global-privé* : l'objet est accessible par toutes les applications mais seule l'application qui l'a créé peut le modifier.
- global-commun* : l'objet est créé par une application mais il peut être modifié par n'importe laquelle des applications en communication.

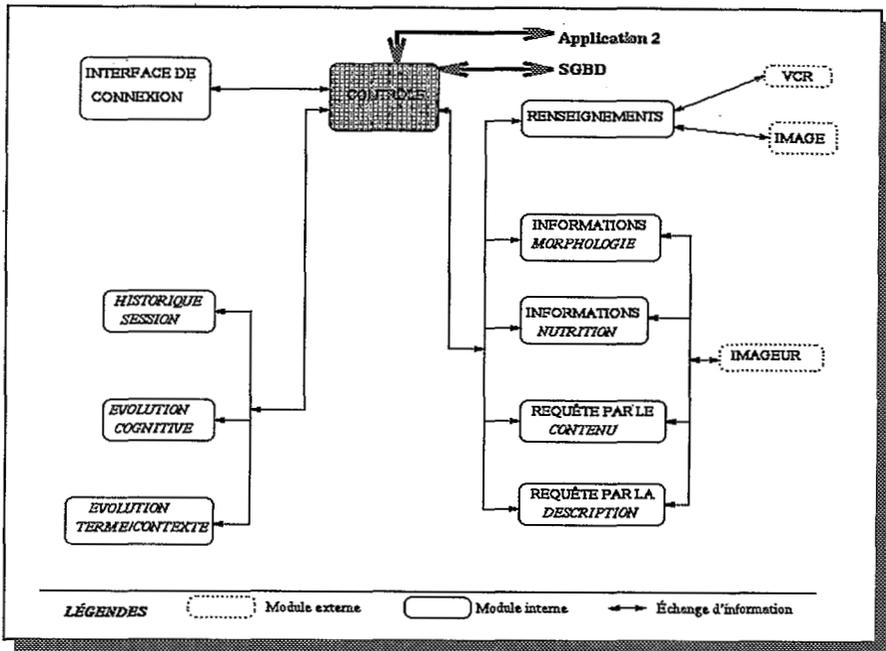


Figure 3. Les modules de BIRDS

VI. Application dans le prototype BIRDS

Le prototype BIRDS est un SRI multimedia qui a été utilisé pour une application pédagogique. Dans ce contexte pédagogique, l'objectif est de «permettre aux élèves de cultiver leur capacité à associer les caractéristiques morphologiques des oiseaux avec leurs modes de nourritures». Il a été expérimenté par des élèves de 6^e et 5^e à Nancy en France.

Pour une application en recherche d'information, le système contient des informations sur les oiseaux de jardin de la Grande Bretagne. Les informations portent sur les habitats de ces oiseaux, leurs morphologies, leurs nourritures et leurs modes de déplacement. Le système est prévu pour tout type de public. Il intègre un modèle de l'utilisateur.

Le prototype est composé de quatre principaux modules: le module d'interface, le module de gestion d'interaction, le module de domaine et le module de l'utilisateur.

Le module de gestion d'interaction est composé de cinq sous-modules qui permettent à l'utilisateur d'utiliser le système dans les quatre phases cognitives (cf. II.2). Le sous-module de pilotage du vidéodisque permet de réaliser les activités de la phase d'observation. L'utilisateur

peut visualiser les séquences d'images sur les oiseaux. Le sous-module d'illustration permet à l'utilisateur de poser des questions simples portant sur des termes uniques. C'est la phase d'abstraction élémentaire.

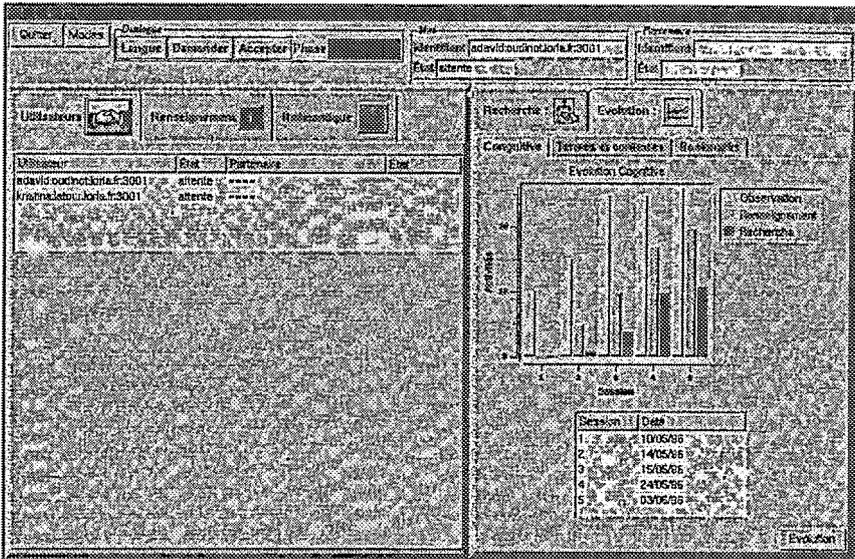


Figure 4. L'interface de BIRDS

Le module de recherche d'images permet à l'utilisateur de rechercher des images. L'utilisateur peut poser des requêtes portant sur les contenus des images ou des requêtes portant sur la description des oiseaux. Dans ce module, il faut que l'utilisateur emploie ses connaissances pour préciser ses demandes. C'est donc la phase de symbolisation et de raisonnement. Le sous-module de création d'un réseau sémantique permet à l'utilisateur d'enregistrer ses propres connaissances ou des annotations personnelles, ce qui correspond à la phase de créativité. Pour une utilisation pédagogique, nous avons intégré un sous-module d'exercices pour proposer des exercices à un élève. Les propositions reposent sur les observations concernant l'évolution de l'élève dans les quatre phases cognitives. Les exercices font évoluer un élève sur d'autres phases cognitives qu'il n'aurait pas encore exploités.

VI.1 Le prototype BIRDS et la recherche coopérative

La Figure 3. montre les modules du prototype BIRDS dans son état actuel. Les modules *historique de session*, *évolution cognitive*, *évolution terme/contexte* concernent le module de l'utilisateur [RUL95]. Les modules *renseignements*, *informations morphologie*, *informations nutrition*, *requête par le contenu*, *requête par la description* concernent le module de gestion de l'interaction. La base descriptive, la base de connaissances et les informations sur l'utilisateur sont gérées par un système de gestion de base de données (SGBD). Le SGBD est un système autonome et externe à BIRDS. Les communications entre BIRDS et le SGBD se font avec les outils que nous avons développés.

Le module VCR, qui est un module externe permet de piloter le lecteur de vidéodisque. La communication entre VCR et BIRDS se fait également avec les outils que nous avons déve-

loppés. BIRDS utilise les informations du VCR pour interroger le SGBD pour avoir des informations sur les oiseaux.

Le résultat d'une requête peut être plusieurs images d'oiseaux. L'imageur (Figure 5) que nous avons développé, qui est aussi un module externe, permet d'afficher tous les images sous forme de palette d'images.

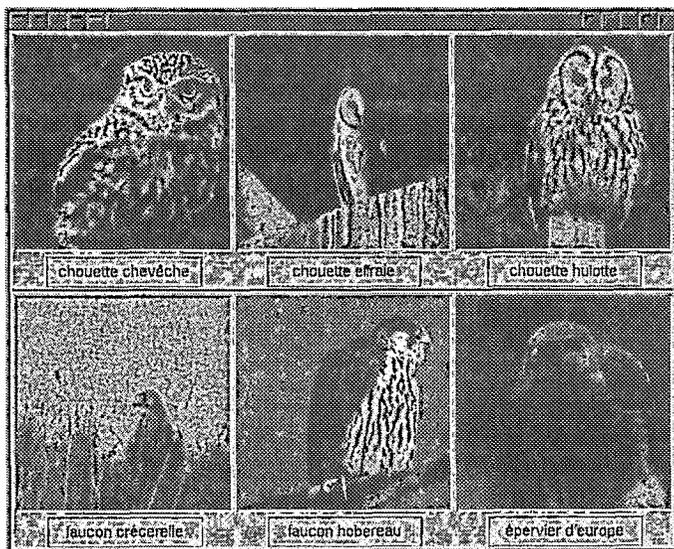


Figure 5. Imageur pour BIRDS

VI.2 Les données multimedia

Un vidéodisque sur les oiseaux

Nous utilisons le vidéodisque BIRDS produit par la BBC, qui contient des séquences d'images sur les oiseaux. Chacune des séquences est dédiée à un oiseau, présentant sa morphologie, son habitat, ses nourritures, et comment il se déplace. Les informations de ce prototype sont réparties en trois catégories:

- les informations spécifiques aux images et aux séquences d'images du vidéodisque (la base descriptive)
- les informations générales sur les oiseaux, indépendantes de la collections des oiseaux du vidéodisque (la base de connaissances)
- les informations sur l'utilisateur (le modèle de l'utilisateur)

VI.3 L'interface de connexion dans BIRDS

L'interface de connexion permet à un utilisateur de demander de se connecter sur l'application d'un autre utilisateur. La liste des utilisateurs enregistrés dans le serveur de localisation est présentée dans la zone «UTILISATEURS». Après la demande de connexion, le partenaire demandé doit accepter la demande pour établir la connexion. Un des utilisateurs peut décider de se déconnecter. Le partenaire doit accepter la demande aussi.

Trois types d'interaction sont implantés: l'observation, l'intervention et le déroulement seul. En mode observation, un utilisateur voit seulement ce que fait l'autre utilisateur. En intervention, les deux utilisateurs mènent ensemble la recherche. En autonomie, l'utilisateur mène seul la recherche

VII. Conclusion

Nous présentons en guise de conclusion quelques avantages de ces axes de recherche et notre perspective. Un système de communication inter-application permet de réaliser un système semi-automatique. Le système pourra prendre en charge les traitements automatisables et les autres par un expert humain, et cela en temps réel. Ce type de système aura indiscutablement un effet psychologique sur l'utilisateur. L'utilisateur saura qu'il peut avoir un interlocuteur humain et pas seulement une machine.

Puisque le SRI est semi-automatique, il sera possible de donner aux experts et aux intervenants, des informations à valeur ajoutée pour mieux répondre aux besoins d'un utilisateur. Ces informations à valeur ajoutée peuvent être réalisées par la modélisation de l'utilisateur et par la synthèse des sessions de travail de l'utilisateur. Le suivi de l'utilisateur pour construire son modèle nous permet de faire une synthèse de ses sessions de travail[ING92]. L'expert sera ce que l'utilisateur a fait et surtout comment.

Nous comptons poursuivre ce travail en appliquant l'architecture que nous proposons à des communications de type n-aire, où on pourra avoir plus de deux utilisateurs. Nous nous sommes limité actuellement à des applications binaires, c'est-à-dire au plus deux utilisateurs.

Nous étudions l'application des concepts présentés dans le domaine d'enseignement assisté par ordinateur, communication commerciale avec les outils informatiques courants et les applications d'aide en ligne.

Références

- [AND92] Daniel ANDLER. Introduction aux sciences cognitives. Folio. Gallimard, 1992.
- [CLU88] M. CIRY-CLUZEAU. Typologie des Utilisateurs et des Utilisations d'une banque d'images: Application au projet EXPRIM. ADBS, 25, mai-juin 1988.
- [COR91] DEC, HP, HyperDesk, NCR, ObjectDesign, and SunSoft. The Common Object Request Broker: Architecture and Specification. Number 91.12.1. Object Management Groupe (OMG), December 1991.
- [DAV91] Amos A. DAVID, Odile THIERY . Le prototype BIRDS: Une expérimentation d'un modèle de l'élève dans un environnement d'apprentissage basé sur l'image. *Génie éducatif*, (2):11 – 19, 1991.
- [GAB92] Michel GABASSI, Bertrand DUPOUY. *L'informatique répartie sous UNIX*. Eyrolles, 1992.
- [ING92] Peter INGWERSEN. *Information retrieval interaction*. Taylor Graham, 1992.
- [RAS91] Francois RASTIER. *Sémantique et Recherche Cognitives*. Formes sémiotiques. puf, France, 1991.
- [RUL95] V. RULQUIN, A. DAVID and O. THIERY. Personnalisation des réponses dans un système de recherche d'information. In Actes Journée d'étude sur les systèmes de recherche d'information élaborés, Ile Rousse, Corse, France, 1995.
- [WAR95] J.P. THOMESSE, S. WARGNIER, Q. WU and all. Modélisation et gestion des applications multimedia distribuées. In *Deuxième séminaire IRMA : Interaction Répartie Multimodale pour le Multimedia*, pages 5 – 12, Septembre 1995.