

CSCW : une Bibliographie

1 Si Vous êtes pressés

2 Objectif

bibliographie de CSCW + documentation de notre démarche : support de la création d'applications CSCW par le Model Driven Engineering
La mobilité

3 Ce qu'est le CSCW

3.1 Vue d'ensemble

présentation générale

3.2 //Paragraphe 1

apport

contraintes

4 Eléments constitutifs d'une application CSCW

4.1 Vue d'ensemble

aspects fonctionnels et non fonctionnels

5 Architecture des applications CSCW

5.1 Vue d'ensemble

6 Les applications CSCW et la mobilité

6.1 Vue d'ensemble

Travail coopératif capilaire, et autres

6.2 Les problèmes

[Vaisman02] présente, dans son rapport de DEA, un recensement des problèmes posés par la mobilité dans les applications de CSCW.

Premièrement, la mobilité introduit la nécessité d'affiner la dichotomie synchrone/asynchrone [David 01].

[McGEE 98], [RIDDLE 98], identifient les problèmes suivants :

- Collaboration et coordination des acteurs mobiles
- Cohérence et validité des informations (connection par intermitence, décentralisation)
- Hétérogénéité des protocoles de communication

- Contraintes d'interfaces
- Solutions spécifiques à l'awareness [TANG01].

Plusieurs limitations existent dans les usages du CSCW en mobilité :

- WYSIWYS en mobilité, très couteux en ressources de communications,
- Canal audio/video, idem (+ contraintes de gigue et de garantie de débit),
- Distance entre acteurs - coût d'usage.

On distingue trois types d'objets mobiles communicants :

- objets autonomes - téléphones, PDA,
- objets embarqués d'environnement - non mobiles, mais peuvent être bougés (exemple : bornes d'information),
- objets passifs - Etiquettes RFID.

Les échanges qui ont lieu se passent :

- entre acteurs
- entre acteurs et objets
- entre objets

Les principaux éléments du contexte concernant la mobilité sont :

- qui - acteurs ou objets,
- où - localisation,
- quand - historique.

6.3 Apport de AMF-C pour le TCAO Capillaire

[Vaisman02] présente les apports du modèle AMF-C pour le TCAO Capillaire.

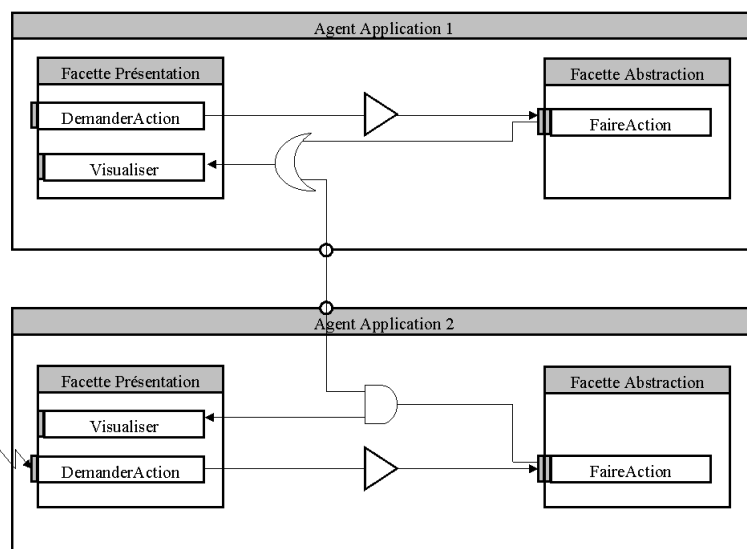
Les problématiques liées à la mobilité identifiées, pour lesquelles l'auteur propose une solution, sont :

1. Réplication
2. Synchronisation des répliques
3. Déconnexion volontaire
4. Déconnexion involontaire
5. Mobilité
6. Référencement, appropriation et utilisation des périphériques

6.3.1 Réplication

La réplication de données ou de traitements permet d'augmenter la tolérance aux fautes par la redondance. Elle permet également la parrallélisation des traitements. Il est nécessaire de conserver la cohérence des différentes répliques, et donc de gérer leur mise à jour de manière adéquate.

La figure suivante montre un exemple de représentation de la gestion de réplication avec AMF-C :



On observe

que la mise à jour par les administrateurs est différente pour chacun des deux agents concernés. Ce mécanisme est particulièrement adapté à ce qu'on appelle le WISIWYS (What I See Is What You See) relâché, c'est à dire lorsque la vue de deux utilisateurs sur une situation collaborative est couplée, mais non identique.

6.3.2 Synchronisation des répliques

Deux stratégies peuvent être adoptées pour la synchronisation des données : optimiste [Brun98]. La stratégie pessimiste vise à conserver les données identiques au cours du temps, et donc à empêcher l'apparition d'incohérences. La stratégie optimiste vise à corriger les incohérences quand elles apparaissent - ces corrections étant relativement coûteuses en ressources. La deuxième est adaptée aux applications asynchrones.

6.3.3 Déconnexion volontaire

La déconnexion explicite d'une des répliques permet la mise en place de mécanismes qui permettent, à la prochaine connexion d'effectuer la mise à jour. Ces mécanismes sont constitués de la mémorisation des états successifs de la donnée.

Ceci suppose qu'au moment de la déconnexion, l'état des données soit cohérent.

6.3.4 Déconnexion involontaire

La possibilité de déconnexions involontaires (coupures réseaux, etc.), et d'erreurs dans la synchronisation des données, rend indispensable la gestion des données desynchronisées.

Par conséquent, lors de chaque appel de méthode sur une réplique, le système doit effectuer la vérification de la cohérence des données, la synchronisation éventuelle, et seulement après peut exécuter l'opération.

La mémorisation de l'ensemble des manipulations sur chaque réplique est donc indispensable. Pour éviter les journaux de taille excessive, une purge régulière doit être réalisée, par exemple après chaque synchronisation.

6.3.5 Mobilité

Les agents peuvent être mobiles dans le réseau. Ceci suppose la disponibilité de la machine cible vis à vis de ces agents mobiles, ainsi que la mise à disposition de services d'outils et de mobilité.

Le déplacement peut être accompagné d'une duplication temporaire de l'agent, qui est alors présent sur la cible et sur la source, ce qui permet de ne pas interrompre son exécution.

Il est possible également de ne répliquer que des facettes d'un agent.

6.3.6 Référencement, appropriation et utilisation des périphériques

L'utilisation d'objets mobiles rend imaginable l'exploitation de ressources locales de l'environnement, qui doivent être accessibles sans configuration manuelle.

6.4 Autre problème

La distribution des fonctions d'un système applicatif entre plusieurs terminaux et/ou serveurs pose un problème majeur de sécurité. En effet, les sessions, les documents, peuvent avoir un caractère de confidentialité.

En particulier, dans le cas d'applications supportant des connexions non fiables au serveur - c'est à dire mettant en oeuvre des clients lourds - aucune garantie n'est apportée au système en ce qui concerne les actions menées dans les terminaux : les processus peuvent tout à fait être modifiés par un utilisateur malveillant, de sorte à tromper soit le système, pour contourner les mécanismes de sécurité existants, soit les autres utilisateurs, pour obtenir des informations de manière induue.

Ces problèmes dépassent largement le cadre de notre travail, et constituent en eux-mêmes un thème de recherche à part entière.

7 Modélisation d'une application CSCW

7.1 Vue d'ensemble

les approches existantes [CSCW_pattern] présente un catalogue de patterns ayant été implémentés dans des applications collaboratives.

les modèles existants

8 Perspectives du CSCW

8.1 Vue d'ensemble

visions pour le futur

9 Pour notre travail

10 Bibliographie

- [**Brun98**] Brun-Cottan Georges, «Cohérence de données répliquées partagées par un groupe de processus coopérant à distance», Thèse de Doctorat, Université Pierre et Marie Curie - Paris VI, 1998, cité par [Vaisman02].
- [**McGEE 98**], McGee D.R., Cohen P., «Exploring Handheld, Agent-based, Multimodal Collaboration», Handheld CSCW, Seattle USA, 1998, cité par [Vaisman02].
- [**RIDDLE 98**], Riddle W., «Palmtop Collaboration : Problems and A Few Solutions», Handheld CSCW, Seattle USA, 1998, cité par [Vaisman02].
- [**TANG01**], Tang J.C., Yankelovich N., Begole J., «ConNexus to Awarenex : Extending awareness to mobile users», In Actes de la conférence CHI 2001, Seattle, Washington, 2001, cité par [Vaisman02].
- [**Vaisman02**] TCAO Capillaire : apports du modèle AMF-C, Gérard VAISMAN, Rapport de DEA, Laboratoire ICTT, 2002.