

---

# Un Mécanisme d'Adaptation Guidé par le Contexte en Utilisant une Représentation par Objets

---

Manuele Kirsch Pinheiro

Laboratoire LSR – IMAG, Équipe SIGMA

[Manuele.Kirsch-Pinheiro@imag.fr](mailto:Manuele.Kirsch-Pinheiro@imag.fr)

<http://www-lsr.imag.fr/Les.Personnes/Manuele.Kirsch-Pinheiro/>





# Sommaire

- Motivations
- Objectifs
- Notion de contexte
- Une représentation par objets du contexte
- Processus de filtrage : vue globale
  - Modèle d'accès progressif
  - Modèle de contenu et de profil
  - Processus de filtrage : les étapes
- Résultats
- Conclusions et perspectives



# Motivations



- Utilisation des nouvelles technologies
  - Liberté de l'utilisateur
    - Accès « *anywhere, anytime* »
    - Utilisateur nomade
  - Contraintes des technologies mobiles
    - Dispositifs mobiles : capacités d'affichage et de mémoire limitées, autonomie limitée en termes d'énergie...
    - Réseaux sans fil : faible débit, déconnexions fréquentes...
  - Utilisation dans des situations non prévues par les systèmes
    - Accessibilité des systèmes sur le Web



# Motivations

- Besoin d'adaptation au contexte
  - But des **systèmes sensibles au contexte**
  - Adaptation des services, du **contenu** ou de la présentation au contexte d'utilisation
  - Exemples d'applications : guide touristique, « geonotes »...
- Problèmes
  - La notion de contexte est limitée à la **localisation** et aux capacités des **dispositifs**
  - Utilisateur pas ou peu impliqué dans le processus d'adaptation



# Motivations

## Problèmes

- Notion de contexte limitée aux **aspects physiques**
  - Localisation, dispositif...
- Insuffisant pour les applications collaboratives
  - Les **collecticiels** permettent à un groupe de collaborer pour atteindre un objectif
  - Utilisateurs sont engagés dans un processus de collaboration
- **Prise en compte des aspects physiques et collaboratifs**



# Motivations

## Problèmes

- La prise en compte des **préférences** de l'utilisateur est limitée
  - Peu de systèmes sensibles au contexte en tiennent compte
  - Les préférences de l'utilisateurs sont **déconnectées du contexte** d'utilisation
- **Prise en compte des préférences en fonction du contexte d'utilisation**



# Objectifs

- **Représentation** de la notion de contexte
  - Aspects **physiques et collaboratifs**
- **Adaptation** de contenu → Filtrage guidé par le contexte
  - Prise en compte des **préférences** de l'utilisateur pour un contexte donné
- Application aux **collecticiels** sur le Web
  - Adaptation de l'information de **conscience de groupe**



# Collecticiels sur le Web et Conscience de groupe

## ■ Collecticiels sur le Web

- ❑ But : collaboration à travers le Web pour atteindre un objectif commun
- ❑ Potentiellement accessibles par des utilisateurs nomades
  - Évolution vers les  **systèmes sensibles au contexte**
- ❑ Chaque utilisateur participe à un processus de collaboration
- ❑ Utilisateurs ont besoin de se maintenir informés des activités et de l'état général du groupe
  - **conscience de groupe**





# Collecticiels sur le Web et Conscience de groupe

- **Conscience de groupe (*awareness*)**
  - la connaissance qu'un utilisateur a de son groupe, de ses collègues et de leurs activités
  - constitue un contexte pour les activités individuelles
    - permet d'évaluer les actions individuelles par rapport au groupe



# Plan

- ✓ Motivations
- ✓ Objectifs
- Notion de contexte
- Une représentation par objets du contexte
- Processus de filtrage : vue globale
  - Modèle d'accès progressif
  - Modèle de contenu et de profil
  - Processus de filtrage : les étapes
- Résultats
- Conclusions et perspectives

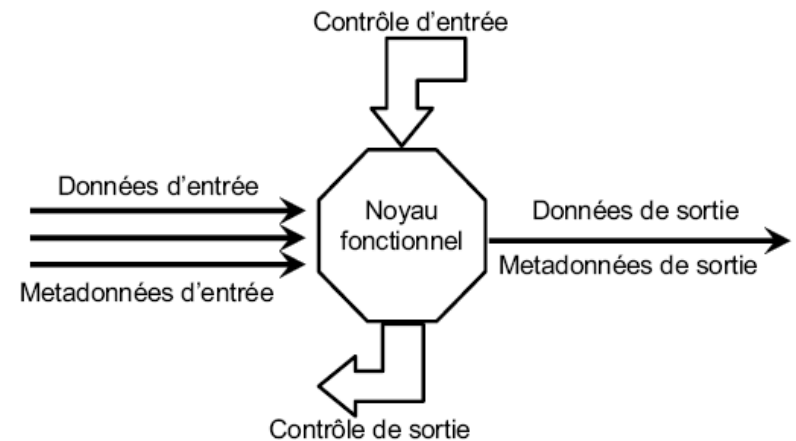
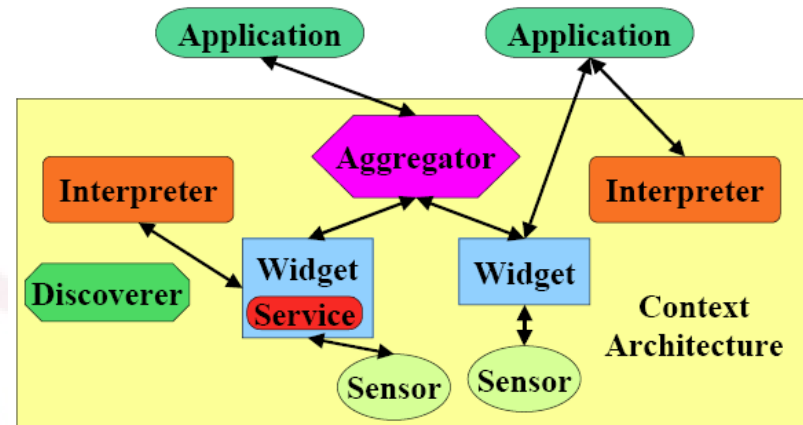


# Notion de Contexte

- Pas de définition précise ...
  - Plusieurs propositions → trop larges, conceptuelles, manque de formalismes exploitables
  - Définition adoptée (Dey, 2000)
    - *tous les **éléments** d'information qui peuvent être utilisés pour **caractériser la situation** d'une entité : personne, endroit, ou objet considéré(e) comme pertinent(e) pour **l'interaction entre l'utilisateur et l'application***
- Impossibilité d'énumérer *a priori* la totalité des éléments
- Acquisition incomplète ou imprécise

- Modèles d'acquisition
  - *Context Toolkit* (Dey, 2000)
    - *widgets, interpreters, aggregators*
  - *Contexteur* (Rey, 2004)
  - Séparer la couche d'acquisition de l'application
  - Problèmes : imprécision, informations incomplètes

## Modèles





# Notion de Contexte

## Modèles

- Modèles de représentation
  - XML, CC/PP, RDF... (Lemlouma, 2004)
  - Ontologies (Bucur, 2005), (Alarcón, 2003)
    - Définition d'un **vocabulaire**
    - Grand pouvoir d'**expression** selon le langage
    - **Limitations** par rapport à la définition des **contraintes** dans les relations
    - Souvent difficiles à comprendre et à manipuler
  - Peu nombreux, considèrent souvent une acquisition parfaite



# Une Représentation par Objets du Contexte

- Proposition d'un modèle extensible et centré sur les collecticiels
- Représentation des entités et de leurs relations par des classes et des associations (à la UML)
- Identification des concepts de base
  - 5 points de vue : *espace*, *outil*, *temps*, *communauté*, et *processus*
  - Identification des classes d'**éléments de contexte** et de leurs **relations**



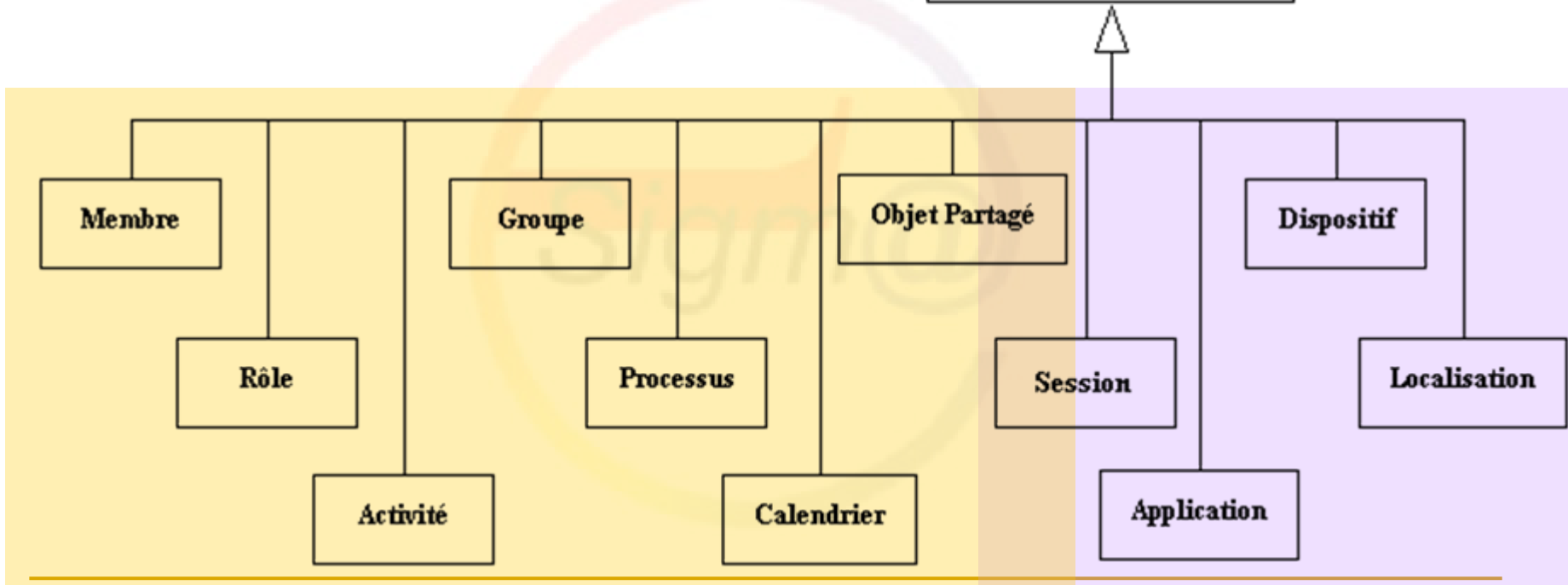
# Une Représentation par Objets

## Concepts

Points de Vue	Concepts	Classes	
Espace	localisation	Localisation	Contexte Physique
Outil	dispositif application	Dispositif Application	
Temps	calendrier	Calendrier	Contexte Collaboratif
Communauté	groupe rôle utilisateur	Groupe Rôle Membre	
Processus	processus activité objet partagé	Processus Activité Objet Partagé	

# Une Représentation par Objets

## Classes









# Une Représentation par Objets

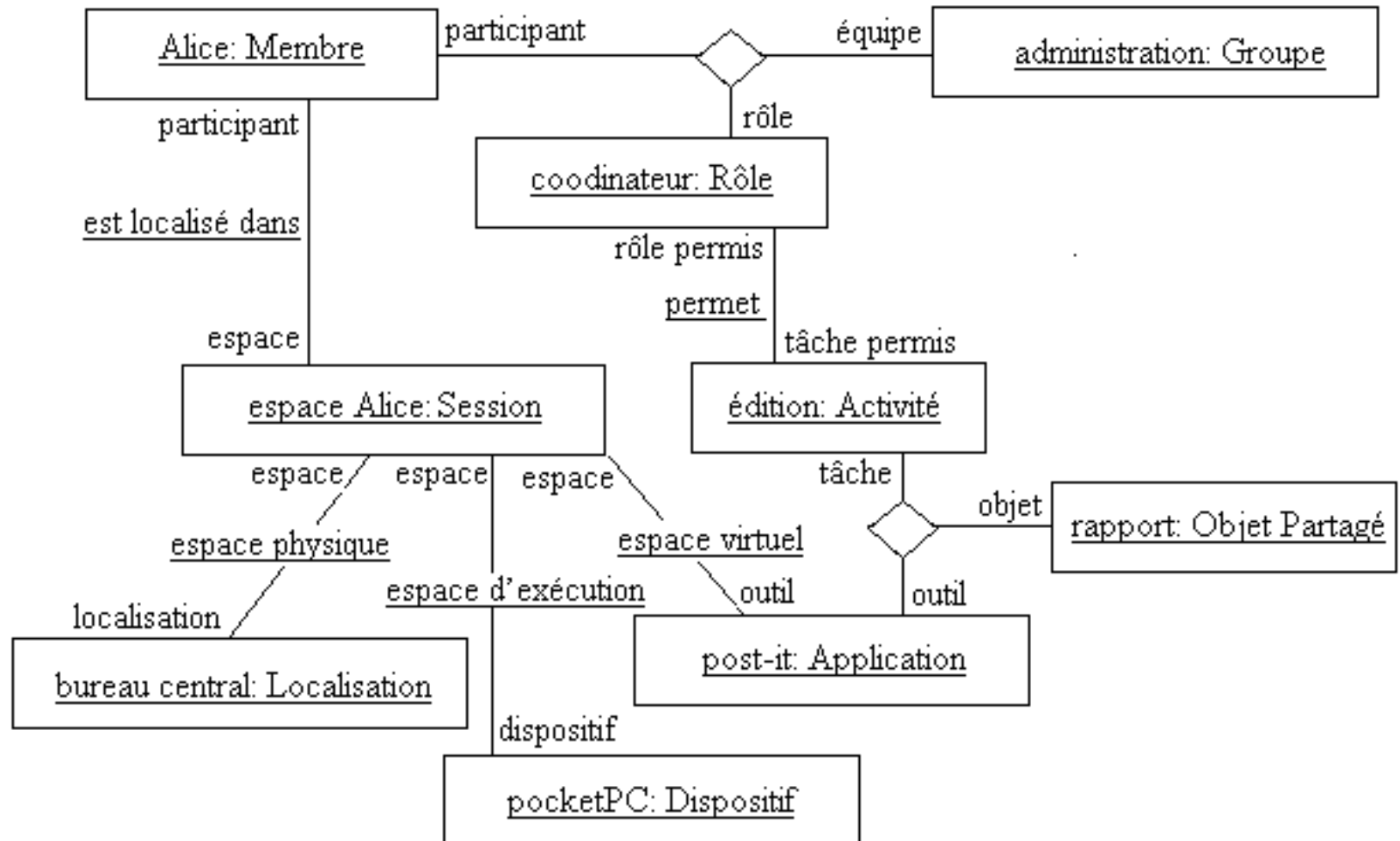
## Exemple

- Alice...
  - Coordinatrice du groupe d'administration
  - une application de *post-it* électroniques
  - Accès depuis un ordinateur de poche



# Une Représentation par Objets

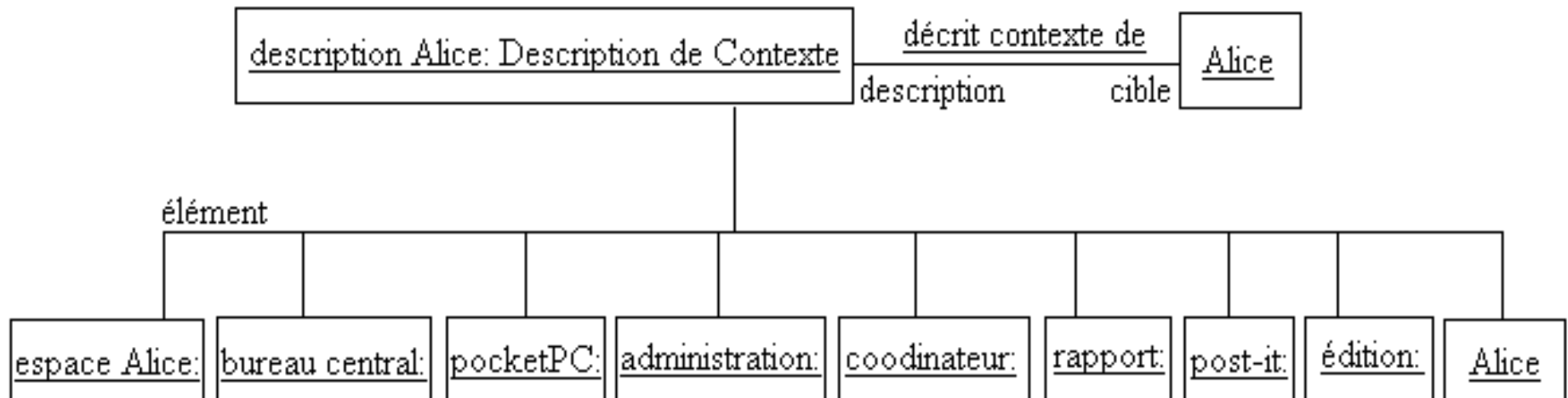
## Exemple





# Une Représentation par Objets

## Exemple





# Plan

- ✓ Motivations
- ✓ Objectifs
- ✓ Notion de contexte
- ✓ Une représentation par objets du contexte
- **Processus de filtrage : vue globale**
  - Modèle d'accès progressif
  - Modèle de contenu et de profil
  - Processus de filtrage : les étapes
- **Résultats**
- **Conclusions et perspectives**



# Processus de filtrage

## Vue globale

- **Objectif** : adaptation du contenu
  - Réduire et organiser l'information délivrée
  - Information de conscience de groupe
- Processus guidé par le **contexte d'utilisation** et par les **préférences** de l'utilisateur pour ce contexte
  - L'utilisation des collecticiels sur le Web se fait majoritairement dans des situations plus ou moins récurrentes



# Processus de filtrage

## Vue globale

- Utilisation de 4 modèles à objets :
  - Modèle du contexte
  - Modèle d'accès progressif
    - Organisation de l'information en différents niveaux
  - Modèle de contenu
    - Description de l'information de conscience de groupe
  - Modèle de profil
    - Description des préférences de l'utilisateur
- Processus en 2 étapes
  - Sélection des profils
  - Application modèle d'accès progressif



# Plan

- ✓ Motivations
- ✓ Objectifs
- ✓ Notion de contexte
- ✓ Une représentation par objets du contexte
- ✓ Processus de filtrage : vue globale
  - Modèle d'accès progressif
  - Modèle de contenu et de profil
  - Processus de filtrage : les étapes
- Résultats
- Conclusions et perspectives





# Modèle d'Accès Progressif

- Un utilisateur n'a pas **tout le temps** besoin d'accéder à **toute** l'information disponible
- **Entité Masquable (EM)**
  - Un ensemble d'au moins 2 éléments
- **Représentations d'Entité Masquable (REM)**
  - Sous-ensembles ordonnés d'éléments de l'EM
- **Stratification**
  - Organisation d'une EM en une séquence de REM



# Modèle d'Accès Progressif

## Opérations

- L'accès progressif repose sur les opérations
  - Masquer / Dévoiler
  - Retourner / Avancer
- Masquer / Dévoiler
  - donne accès à la REM de niveau inférieur / supérieur
- Retourner / Avancer
  - donne accès aux éléments qui appartiennent **uniquement** au niveau inférieur / supérieur

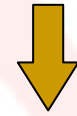
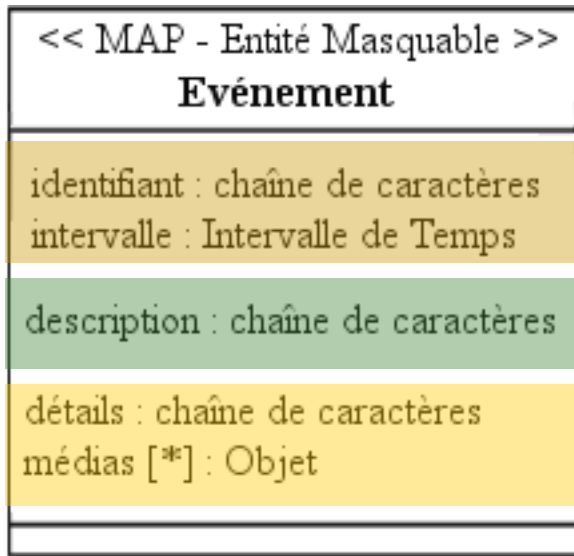


# Modèle d'Accès Progressif

## Opérations

### Stratification $S_1$

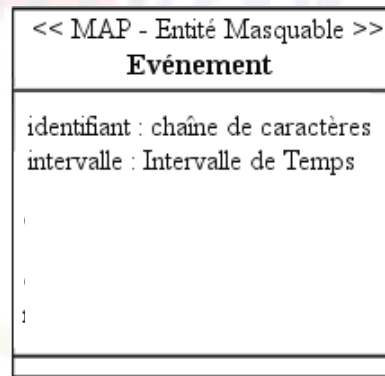
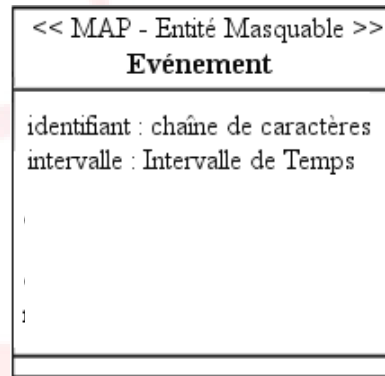
### Dévoiler / Masquer



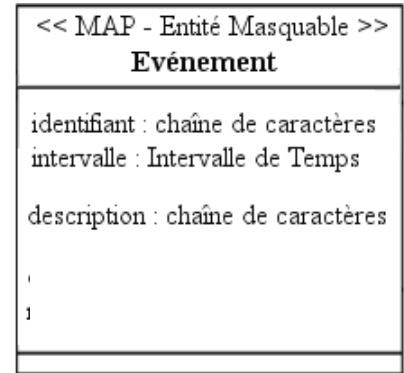
$REM_1$

$REM_2$

$REM_3$



Dévoiler



Masquer



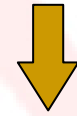
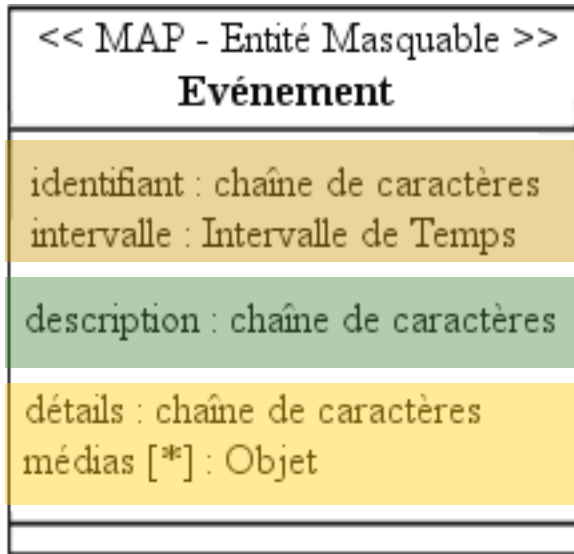


# Modèle d'Accès Progressif

## Opérations

Stratification  $S_1$

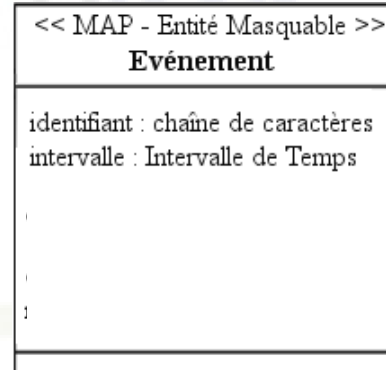
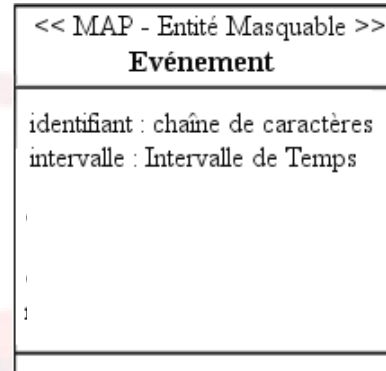
Avancer / Retourner



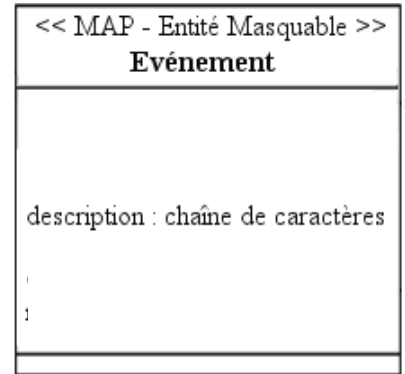
REM<sub>1</sub>

REM<sub>2</sub>

REM<sub>3</sub>



Avancer



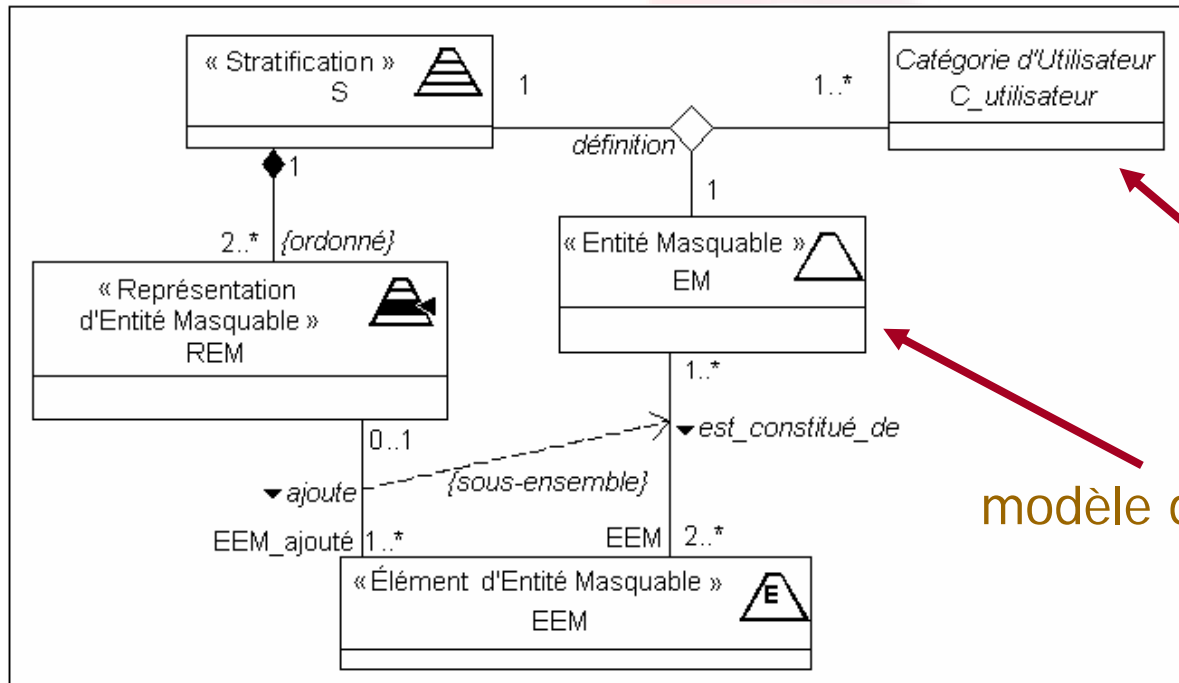
Reculer



# Modèle d'Accès Progressif

## Modèle UML

- Le MAP est défini comme un **diagramme de classes UML** générique qui exprime les règles de bonne formation des stratifications pour une EM donnée



adaptation à différents utilisateurs

modèle de données



# Plan

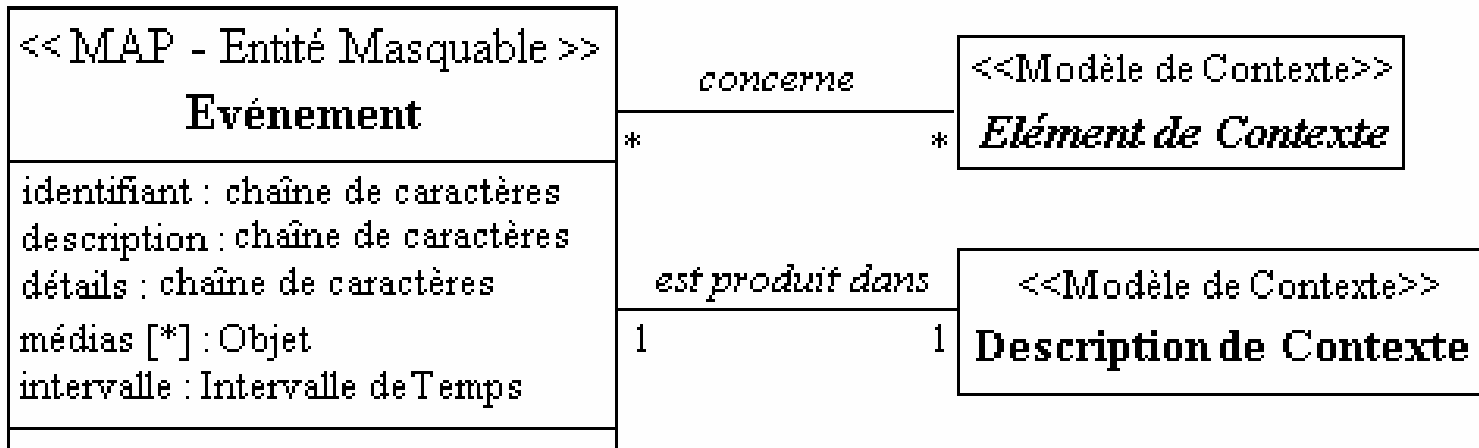
- ✓ Motivations
- ✓ Objectifs
- ✓ Notion de contexte
- ✓ Une représentation par objets du contexte
- ✓ Processus de filtrage : vue globale
  - ✓ Modèle d'accès progressif
  - Modèle de contenu et de profil
  - Processus de filtrage : les étapes
- Résultats
- Conclusions et perspectives



# Modèle de Contenu et de Profil



- **Modèle de contenu**
  - Description du contenu à adapter → **conscience de groupe**
  - **Événements**
    - un ensemble d'informations relatives à une action réalisée dans le cadre du travail collaboratif sur un sujet donné
    - Les **entités masquables** pour le modèle d'accès progressif





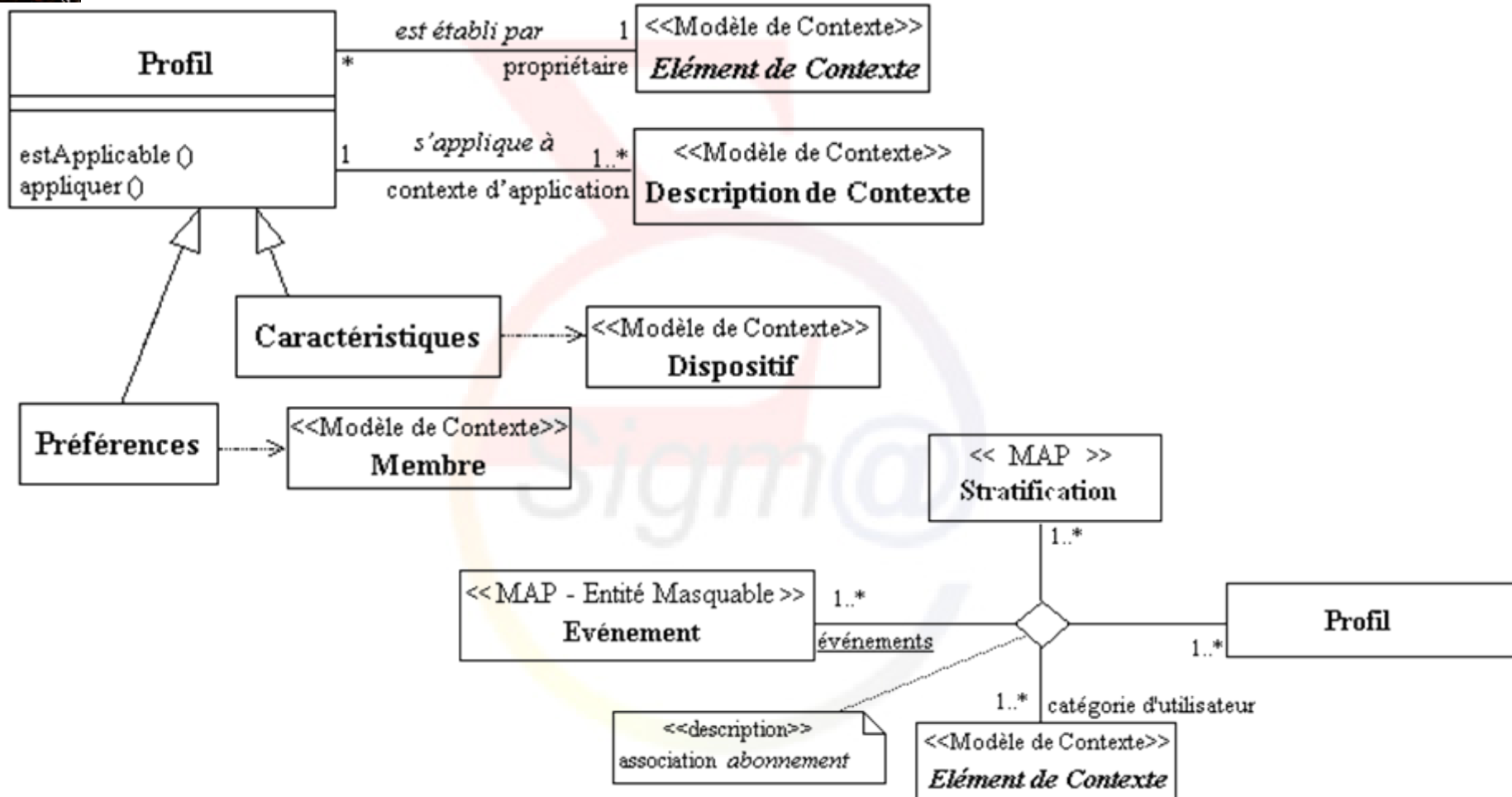
# Modèle de Contenu et de Profil

- Modèle de profil
  - Représentation des **préférences** et des contraintes que le système doit satisfaire pour un contexte donné
  - « *Règles* » de filtrage
    - Définition des événements pertinents
    - Organisation des événements
  - Description des *contextes potentiels*





# Modèle de Contenu et de Profil





# Plan

- ✓ Motivations
- ✓ Objectifs
- ✓ Notion de contexte
- ✓ Une représentation par objets du contexte
- ✓ Processus de filtrage : vue globale
  - ✓ Modèle d'accès progressif
  - ✓ Modèle de contenu et de profil
  - Processus de filtrage : les étapes
- Résultats
- Conclusions et perspectives



# Processus de Filtrage

## Les étapes

- Filtrer l'information de **conscience de groupe**
- Le filtrage se fait en **2 étapes** :
  - **1ère étape** : sélection des profils valables
    - *Comparaison entre les contextes d'application et le contexte courant de l'utilisateur*
  - **2ème étape** : filtrage et organisation des événements
    - *Filtrage des événements auxquels sont abonnés les profils*
    - *Organisation en niveaux de détail selon les stratifications*
- **Résultat** :
  - un ensemble limité et organisé d'informations qui sera délivré à l'utilisateur



# Processus de Filtrage

## 1ère étape

- But : sélectionner, parmi les profils disponibles, les profils valides par rapport au contexte courant de l'utilisateur
- Comparaison (*matching*) entre les contextes d'application et le contexte courant
  - Relation de « sous-contexte »
    - graphe → un objet et tous les objets auxquels il est associé forment un graphe
  - Opérations *Contains* et *Equals*



# Processus de Filtrage

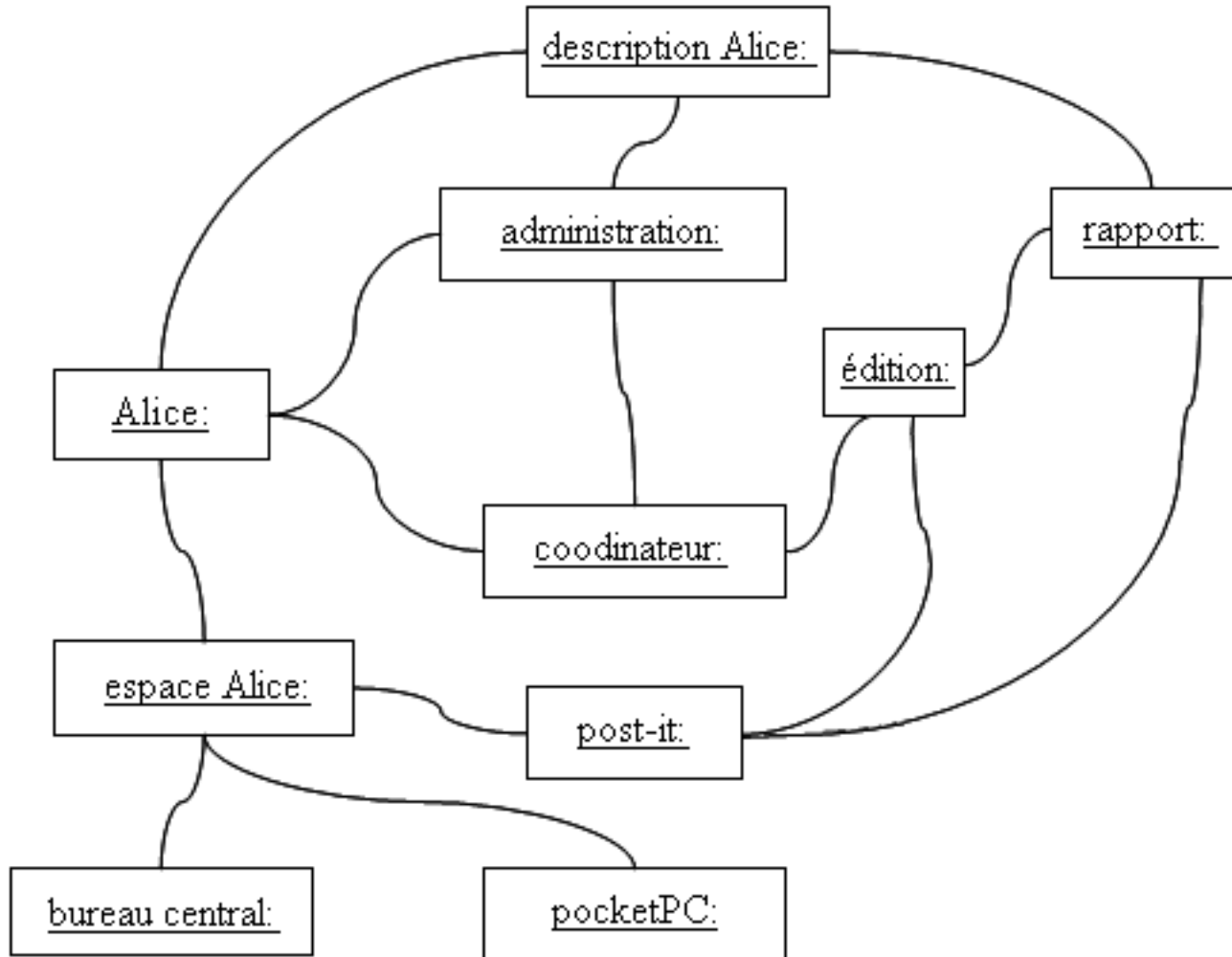
## 1ère étape - Opérations

- *Equals* - si deux instances ont le même contenu
  - $O$  *equals*  $O'$  si et seulement si
    - **Objets** :  $(\forall i) O.v_i = O'.v_i$
    - **Tuples** : si  $E$  connecte les  $O_1$  et  $O_2$ , et  $E'$  connecte  $O'_1$  et  $O'_2$ , alors  $E$  *equals*  $E'$  si  $O_1$  *equals*  $O'_1$  et  $O_2$  *equals*  $O'_2$ .
- *Contains* - si un graphe contient tous les éléments de l'autre
  - $C$  *contains*  $C'$  si et seulement si
    - **Objets** : pour chaque objet  $N'$  appartenant à  $C'$  (noté  $N'C'$ ), il existe un objet  $N$  appartenant à  $C$  (noté  $NC$ ) tel que  $NC$  *equals*  $N'C'$
    - **Tuples** : pour toute tuple  $E'$  de  $C'$  (notée  $E'C'$ ), il existe au moins une tuple  $E$  de  $C$ , (notée  $EC$ ) telle que  $EC$  *equals*  $E'C'$
- Variations :
  - Utilisation d'un seuil : % des éléments égaux
  - Utilisation des ontologies pour les valeurs de variables (dictionnaire)
  - Calcul de distance sémantique



# Processus de Filtrage

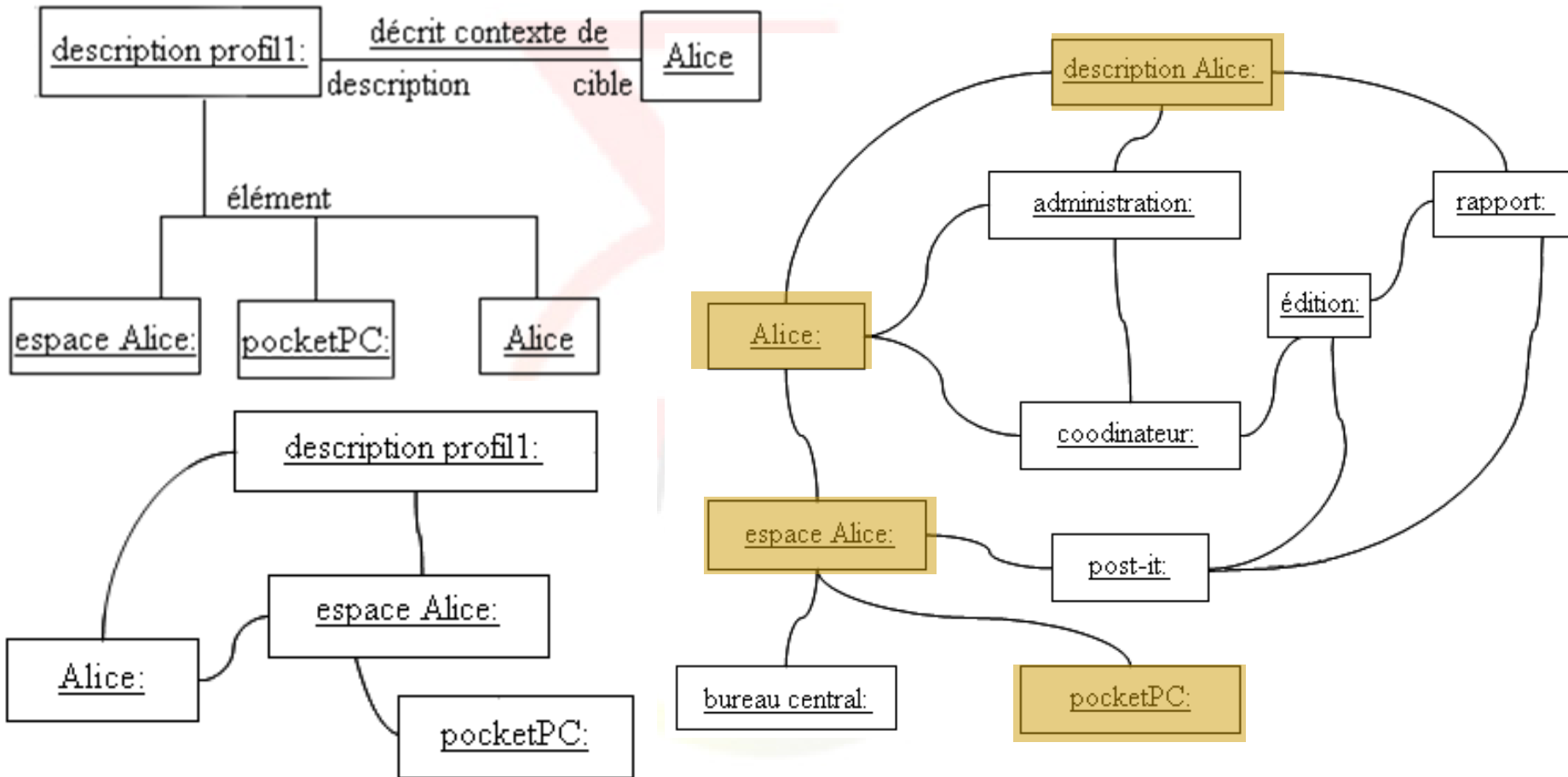
1ère étape





# Processus de Filtrage

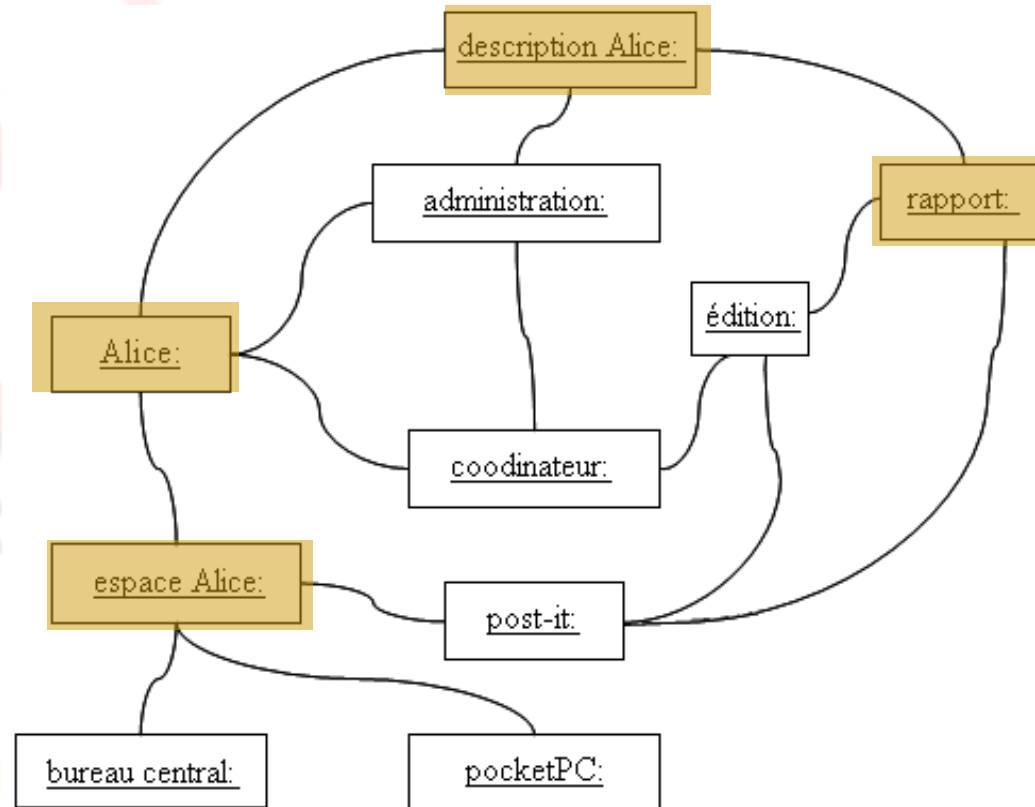
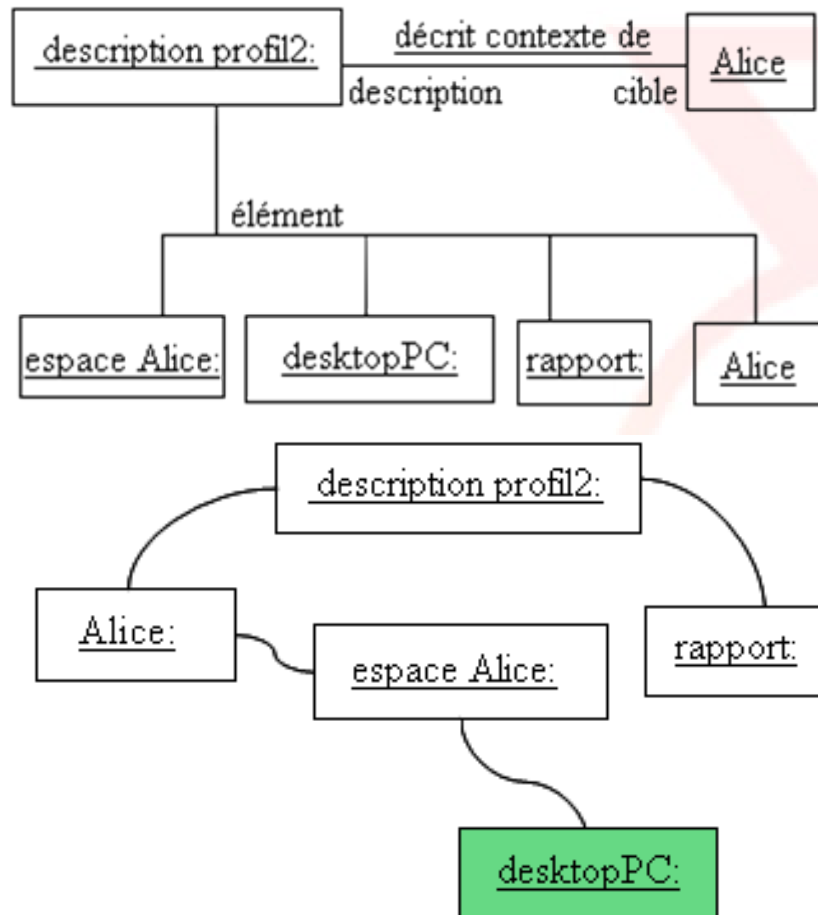
## 1ère étape





# Processus de Filtrage

## 1ère étape







# Processus de Filtrage

## 2ème étape

- Application des stratifications définies par les profils sélectionnés
- Application **successive** de chaque profil
  - Ordonnancement des profils sélectionnés par **ordre de priorité**
  - Pour chaque profil
    - Sélection des **événements** auxquels la classe est abonnée
    - Application des **stratifications**
- **Ordre de priorité par une mesure de similarité**



# Processus de Filtrage

## 2ème étape

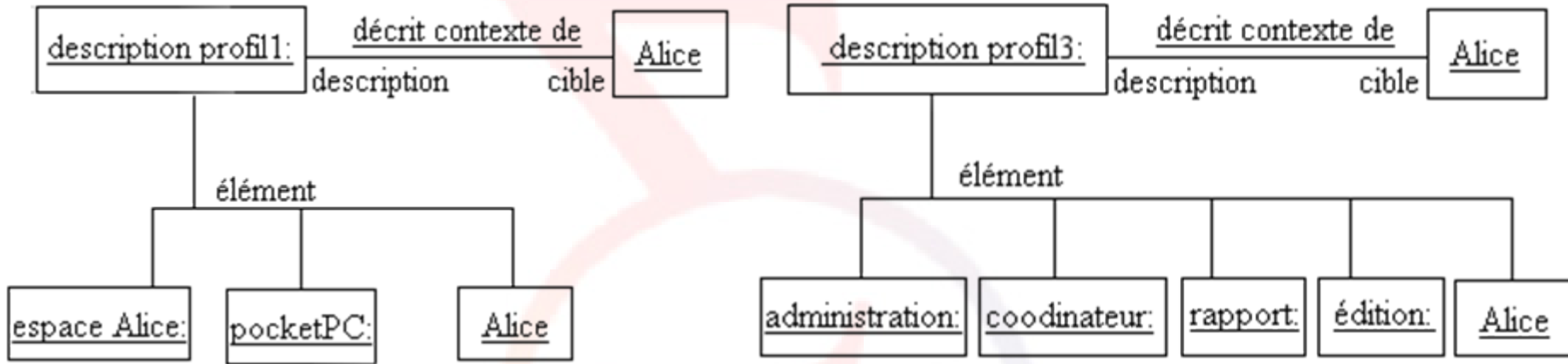
- Mesure de similarité (*Sim*)
  - Proportion d'éléments du contexte de l'utilisateur qui ont des équivalents (*Equals*) dans le contexte d'application
  - But : favoriser les profils les plus spécifiques au détriment des profils plus généraux

$$Sim(C_u, C_p) = x, x \in \mathbb{R}, x \in [0, 1]$$

- $x = 1$  si tout élément de  $C_u$  possède un élément égal (*equals*) en  $C_p$
- $x = |X| / |C_u|$  où  $X = \{x \mid x \text{ equals } y, x \in C_u, y \in C_p\}$

# Processus de Filtrage

## 2ème étape



$$Sim(C_{Alice}, C_{profil1}) \approx 0,3$$

< Alice,  $C_{profil1}$ , **even1**,  $S_1$  >

$S_1 = \{ \{ \text{identifiant, intervalle} \}, \{ \text{description} \}, \{ \text{détails, médias} \} \}$

$$Sim(C_{Alice}, C_{profil3}) \approx 0,6$$

< Alice,  $C_{profil3}$ , **even2**,  $S_3$  >

$S_3 = \{ \{ \text{identifiant, description} \}, \{ \text{détails, médias} \} \}$



# Plan

- ✓ Motivations
- ✓ Objectifs
- ✓ Notion de contexte
- ✓ Une représentation par objets du contexte
- ✓ Processus de filtrage : vue globale
  - ✓ Modèle d'accès progressif
  - ✓ Modèle de contenu et de profil
  - ✓ Processus de filtrage : les étapes
- Résultats
- Conclusions et perspectives



# Résultats Provisoires

- Implémentation
  - Modèle de contexte : Système de Représentation de Connaissances par Objets **AROM**
  - Mécanisme de filtrage : canevas **BW-M**
- Tests
  - Divers versions des opérateurs *Contains* et *Equals*
  - Mesure de similarité *Sim*
- Résultats
  - Méthode d'adaptation : cibler et réduire les informations
  - Définition des contexte d'application plus ou moins critique
  - Définition des stratifications : besoin d'interface appropriée



# Conclusions et Perspectives

- **Modèle à objets du contexte**
  - Extensible
  - Facile à comprendre
  - Représentation partielle
- **Mécanisme de filtrage**
  - Participation directe de l'utilisateur
  - Personnalisation des informations (MAP)
  - Personnalisation selon le contexte



# Conclusions et Perspectives

## ■ Perspectives

- ❑ **Modèle de profil** : création et mise à jour automatique des **stratifications**
- ❑ **Opérations** : calcul distance sémantique (*Equals*)
- ❑ Utilisation des stratifications pour le contrôle d'accès à l'information
- ❑ Connexion à des modèles d'acquisition de contexte
- ❑ Tests avec des utilisateurs en situations réelles



# Questions ?







# Références

- **Alarcón**, R., Fuller, D., « *Application design based on work ontology and an agent based awareness server* ». In : Favela, J., Decouchant, D. (Eds.), **9th International Workshop on Groupware: Design, Implementation and Use - CRIWG 2003, LNCS 2806**, Springer-Verlag, **2003**, pp. 314-329.
- **Bucur**, O., Beaume, P., Boissier, O., « *Définition et représentation du contexte pour des agents sensibles au contexte* », **Actes des deuxièmes journées francophones : Mobilité et Ubiquité 2005 (UbiMob'05)**, 31 mai – 3 juin **2005**, Grenoble, France, pp. 13-16.
- **Dey**, A.K., « *Providing Architectural Support for Building Context-Aware Applications* », PhD Thesis, Georgia Institute of Technology, **2000**.
- **Lemlouma** T., « *Architecture de négociation et d'adaptation de Services Multimédia dans des Environnements Hétérogènes* », Thèse de Doctorat, Institut National Polytechnique de Grenoble, Grenoble, France, avril **2004**.
- **Rey**, G., Coutaz, J., « *Le contexteur : capture et distribution dynamique d'information contextuelle* », **Mobilité & Ubiquité'04 (UbiMob'04)**, Nice, France, **2004**. pp. 131-138.



# Résultats Provisaires

## Algorithmes

```
Set ProfileSelection (Member owner) {  
  
    //set of selected profiles is initially empty  
    Set selected = new Set ();  
  
    //get the user's context  
    ContextDescription Cu = getContext (owner);  
    //get user's profiles  
    Set profiles = getProfiles (owner);  
  
    //for each profile, compare each application context with  
    //the user's context  
    while (profiles.hasNext()) {  
        Profile P = profiles.next();  
        Set applicationContext = getContext (P);  
        while (applicationContext.hasNext ()) {  
            ContextDescription Cp = applicationContext.next();  
            //if the user's context contains the application context,  
            //then the profile is selected  
            if (Contains (Cu, Cp))  
                selected.add (P);  
        }  
    }  
  
    //return the selected profiles  
    return (selected);  
}
```



# Résultats Provisaires

## Algorithmes

```
List SelectAndOrder (Set selected, Member owner) {  
  
    //list of information to be delivered, organized by levels  
    List deliverList = new List();  
  
    //order the profiles using the Sim mesure  
    List profiles = order (selected, getContext (owner));  
  
    //for each profile, combine and apply the stratifications by type  
    while (profiles.hasNext ()) {  
        Profile P = profiles.next();  
  
        //combine the stratifications by type (extensional and intensional)  
        Stratification Sext = combine (getExtensionalStratifications (P));  
        Stratification Sint = combine (getIntensionalStratifications (P));  
  
        //get events signed up by the profile  
        Set events = getSignedUpEvents (P);  
  
        //apply the stratifications (first extensional, then intensional)  
        List selectedInfo = apply (Sext, events);  
        selectedInfo = apply (Sint, selectedInfo);  
        //add the organized information  
        deliverList.add (selectedInfo);  
    }  
  
    return deliverList;  
}
```