



---

## 1. Introduction

De nos jours, l'informatique mobile est devenue une réalité grâce aux avancées des technologies mobiles (*appareils et applications mobiles*) et à celles des réseaux sans fil (*Wifi, Hertzians, ZigBee, etc.*) [7, 17]. Les appareils mobiles disponibles sur le marché sont de plus en plus petits, accessibles à bon prix, plus puissants et munis de fonctionnalités très diversifiées (stockage, traitement de données, communication, transmission, etc.). Ils offrent, grâce à leur portabilité, la possibilité d'accéder ou de communiquer avec une autre source de données (Internet, réseaux locaux, mobiles, etc.) à tout moment. Ils peuvent aussi acquérir ou traiter des données pour des domaines très variés (acquisition de données climatologiques grâce aux réseaux de capteurs, comme décrite par les auteurs dans [2]). Les applications mobiles, sur les appareils mobiles, permettent l'accès à des données publiques (météo, trafic routier, cours de la bourse) ou privées (données bancaires, agenda, dossier médical, bookmarks) (Cf. [15]). Celles à usage professionnelle permettent de collaborer (accès ou partage de données relatives à une tâche exécutée par un groupe de personnes).

Cependant, les environnements mobiles diffèrent des environnements traditionnels de plusieurs façons au vu des caractéristiques des médias sans fil et des unités mobiles. Les mécanismes de gestion des données dans de tels systèmes doivent s'accommoder aux contraintes fortes inhérentes à l'environnement matériel et logiciel utilisé.

La problématique première dans cette gestion est certainement la disponibilité des données, parce que les terminaux mobiles sont souvent déconnectés et les réseaux mobiles n'offrent généralement qu'un faible débit, une couverture partielle du territoire et des coûts de communication relativement élevés. C'est pourquoi la réplication des données peut s'avérer utile en environnements mobiles comme mentionné dans [1, 3, 6, 27]. Elle a pour objectif d'améliorer la disponibilité des données et de faire face aux problèmes de couverture réseau dans certaines zones. Cependant, une telle stratégie impose aussi *une gestion de la cohérence des différentes copies de données (répliques) et leur intégration ou fusion en une seule copie*. En effet, les différentes copies d'une donnée répliquée (divergentes à un instant) doivent converger vers un même état dès que possible pour éviter d'éventuelles incohérences dans le système. Certaines protocoles de réplication [9], [25], [29], [34], etc. peuvent être caractérisées avec le modèle utilisé pour propager les mises à jour. Alors que d'autres [1], [4], [8], etc. s'adaptent aux exigences de l'utilisateur (par exemple besoin en termes de cohérence et de sécurité) ou au contexte de l'environnement (bande passante disponible, position géographique de l'utilisateur, etc.) pour gérer la réplication et la cohérence des données.

En ce qui concerne la gestion des transactions mobiles, certains traitements de données (transactions) se font en local, sur les terminaux mobiles ; d'autres font appel à des ressources distantes. Ce qui peut avoir une incidence sur la gestion des transactions et la cohérence des données. Il est donc nécessaire que les mécanismes d'ordonnancement des transactions des applications mobiles tiennent en compte ces contraintes. Dans la littérature, des propositions (Kangaroo, MDSTPM, Moflex, Clustering, Prewrite, Semantics-based, etc.) tentent d'apporter des solutions à ces problèmes. Elles considèrent dans la plupart des cas un modèle d'exécution bien précis (uniquement sur serveurs fixes ou distribuée sur les unités mobiles).

Dans cet article, nous proposons un état de l'art sur les protocoles de réplication et les mécanismes de gestion des transactions utilisés dans la gestion des données en environnements mobiles pour assurer une haute disponibilité du service (données et traitements)

et gérer les accès concurrents malgré les déconnexions et la mobilité.

Nous commençons par donner les architectures possibles et les caractéristiques d'un système mobile. Ensuite, nous proposons une classification des protocoles de réplication proposées en environnements mobiles dans la sous-section 3.1. Dans la sous-section 3.2, nous présentons les mécanismes de gestion des transactions mobiles. Enfin, la Section 4 conclut le papier.

## 2. Architectures et Caractéristiques

### 2.1. Architectures d'un environnement mobile

Le mode de fonctionnement d'un système mobile donne lieu globalement à deux architectures possibles : une architecture en *mode infrastructure* et une architecture en *mode ad hoc*.

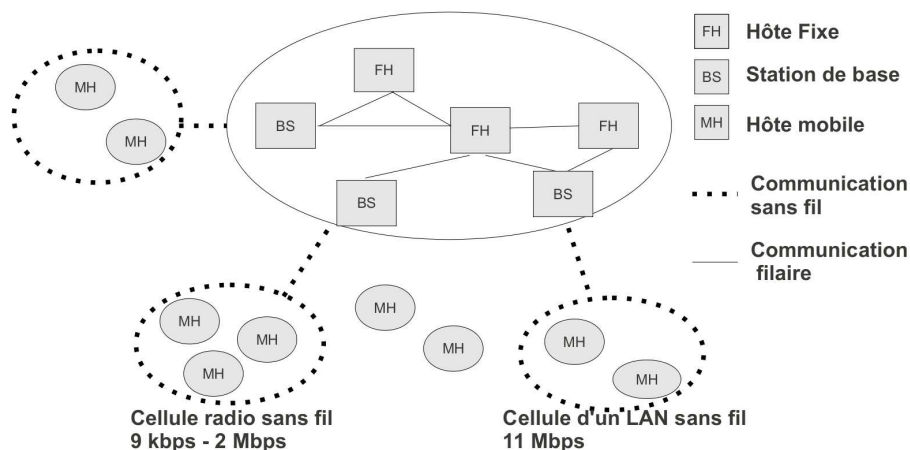


Figure 1. Architecture d'un environnement mobile en mode Infrastructure

#### Mode infrastructure

Le mode infrastructure d'un système mobile se présente comme décrit sur la Figure 1 et comprend deux entités distinctes [19, 32] : une *partie fixe* et une *partie mobile*. La partie fixe (Fixed Network) est principalement constituée par des stations fixes (serveurs, stations de travail, etc.) qui communiquent via un réseau filaire classique à haut débit. Certaines de ses stations fixes sont appelées stations de base (Base Stations) ou stations de support de base (Mobile Support Stations). Elles sont munies d'interface sans fil leur permettant de communiquer avec la partie mobile. Les stations de base ont des fonctions de routage et de commutation mais peuvent aussi supporter des applications qui participent à la gestion des données dans le réseau. La partie mobile, quant à elle, est uniquement composée de stations mobiles (hôtes mobiles ou unités mobiles). Celles-ci utilisent des canaux sans fil pour communiquer entre elles ou pour se connecter aux stations de base qui jouent le rôle de passerelle (Gateway) entre la partie filaire et la partie non filaire de l'infrastructure réseau. Elles peuvent se déplacer librement au sein d'une zone géo-

graphique appelée *cellule*. Chaque cellule est gérée par une station de base particulière et une unité mobile ne peut être connectée à un instant donné qu'à une seule station de base. Le processus de passage d'une cellule à une autre est assuré par un protocole appelé *Handover* et est rendu transparent aux mobiles.

### Mode ad hoc

Le système mobile ad hoc est construit de manière spontanée et ne comprend que des unités dotées d'interface sans fil. Les communications peuvent se faire de manière directe entre toute paire d'unités appartenant à l'environnement, sans avoir recours à une station centrale : on parle ainsi d'unités autonomes dans la gestion des données et des communications.

Dans la suite, nous nous référerons dans notre étude à une architecture généralisée d'un système mobile où les fonctionnements en mode infrastructure et en mode ad hoc pourraient être observés simultanément. La Figure 1 est un exemple d'un tel environnement mobile.

## 2.2. Caractéristiques d'un environnement mobile

L'environnement mobile est caractérisé par la mobilité des appareils, les communications sans fil et les contextes/conditions de travail très imprévisibles. Dans [19, 28], les auteurs ont proposé une étude assez détaillée des caractéristiques citées ci-dessus. Nous nous en inspirons dans la suite de cette section.

### Les appareils mobiles

Les appareils mobiles sont de nature diverses (laptops, notebooks, PDAs, capteurs, téléphones portables, etc.) et ne comportent pas forcément la même configuration. Elles sont en général très légers et de petite taille. Les limitations sont nombreuses surtout du point de vue énergétique avec l'usage de batteries. Cette énergie électrique demeure une ressource critique, en dépit des efforts de la technologie mobile (exemple : les machines portables de dernière génération) en termes de capacité de stockage et de traitement. Enfin, l'unité mobile est moins fiable et moins sécurisée que l'unité statique à cause de sa très forte vulnérabilité aux pannes, aux vols et aux pertes.

### Les réseaux sans fil

Les réseaux sans fil sont caractérisés par des débits très variables, une connectivité intermittente, etc. (Cf. Section 1). La qualité de la communication sans fil varie selon la technologie utilisée. Les auteurs de [7] proposent une classification des différentes technologies sans fil existantes sur le marché.

Le tableau 1 résume les caractéristiques principales des unités mobiles et des canaux de

<i>Caractéristiques du médium sans fil</i>	<i>Caractéristiques d'une unité mobile</i>
Faible largeur de bande passante	Petite taille et assez légère
Déconnexions fréquentes	Petit écran
Variabilité des besoins en bande passante	Durée de vie des batteries limitée
Déconnexions volontaires ou involontaires	Risque de vol, perte, accidents
Diffusion naturelle dans chaque cellule	Ressources CPU et de stockage variables

**Tableau 1.** *Caractéristiques du médium sans fil et d'une unité mobile [30]*

communications sans fil.

### Contextes et Conditions de travail

Le système reste très hétérogène et requiert la conception d'algorithmes distribués efficaces pour assurer une bonne gestion des données. Il se singularise par quatre modes de fonctionnement : *connecté, partiellement déconnecté, déconnecté et en veille*. Le passage d'un mode à un autre peut être fortement causé par des contraintes sur l'unité mobile ou sur le réseau sans fil. Par conséquent, la qualité du service peut varier à tout moment du fait d'un changement de conditions de travail (économie d'énergie électrique, bande passante faible, surcharge ect.).

---

## 3. Gestion des données en environnements mobiles

### 3.1. Protocoles de réplication des données

La nature de certains équipements utilisés dans un système mobile influe beaucoup sur les performances du système. L'état de l'art sur la question [3, 6, 18] montre que le fonctionnement du système (disponibilité des données, temps de réponse du système, diminution du coût de la communication, etc.) repose essentiellement sur les mécanismes de réplication mis en place. Le but de la réplication est d'optimiser les performances du système en termes de temps d'accès, de traitement de données et la survie des informations critiques, etc. Cela en offrant une très haute disponibilité, surtout au niveau des unités mobiles, qui représentent les points sensibles pour le bon fonctionnement du système.

*Plusieurs stratégies de réplication optimistes*, combinées avec des protocoles de gestion de la cohérence des répliques, ont été proposées [3, 8, 10, 9, 12, 13, 23, 29, 33, 34]. Elles couvrent beaucoup de domaines d'applications (Base de données, Systèmes de fichiers, Gestion de Documents, etc.). Certaines de ces propositions peuvent être classées suivant le modèle de propagation des mises à jour (*client-serveur, pair-pair et hybride*), alors que d'autres sont dites *adaptatives*.

#### Les systèmes de réplication client-serveur

Les systèmes de réplication client-serveur répliquent les données sur un ensemble de serveurs. Ils utilisent un cache local au niveau des mobiles pour permettre les traitements en mode déconnecté. Cependant, dans ce modèle, les mobiles sont toujours obligés d'établir une connexion avec un serveur pour l'accès aux données ou la mise à jour des données. Le serveur se charge ensuite de propager les mises à jour aux autres hôtes (serveurs ou clients). Ces connexions sont souvent *coûteuses en termes de temps, notamment dans un environnement mobile géographiquement étendu*. Par ailleurs, *une convergence rapide des répliques nécessite des mécanismes efficaces de gestion des mises à jour concurrentes, lorsque le nombre de répliques devient important*. Enfin, signalons que les serveurs peuvent constituer des goulots d'étranglement ou points de défaillance du fait d'une surcharge de trafic (mises à jour ou accès aux données).

*Les synchroniseurs de données (mail, agenda électronique, etc.) ou de fichiers* et les *gestionnaires de configuration* constituent des exemples populaires de systèmes de réplication client-serveur utilisés en environnements mobiles. Les synchroniseurs de données (ActiveSync, iSync, HotSync ...) et de fichiers (Microsoft Briefcase, Power Merge, Windows File Synchronizer, Unisson ou 3DM) [9, 13, 15] permettent de synchroniser ou de fusionner des données entre une station mobile et un serveur central. Les gestionnaires de configuration [9], tels que CVS (*Concurrent Versions System*), Subversion ou encore

ClearCase, permettent à des utilisateurs de partager des documents et de les éditer de manière collaborative. Les accès concurrents sont gérés par un système central avec le paradigme copier-modifier-fusionner. La fusion [9, 15] des copies est réalisée par des outils dédiés, suivant le type du fichier : pour les fichiers textes diff3 ou rcs Merge, XMLmerge pour les fichiers XML, etc.

### Les systèmes de réplication pair-à-pair

Les systèmes de réplication pair-à-pair tentent d'accroître l'accès aux données et la convergence des répliques pour prendre en compte certaines spécificités du système mobile (déconnexion du mobile, bande passante faible). Cependant, avec les communications pair-à-pair, une convergence des répliques est presque impossible au delà d'une certaine limite [9, 29] : donc la réplication des données passe difficilement à l'échelle. Bayou [29] et Rumor [25] sont des exemples de systèmes de réplication pair-à-pair. Le premier est une plateforme de réplication de base de données mobiles qui adopte une stratégie de réplication optimiste [9] (*read any / write all*) pour permettre à l'utilisateur d'accéder aux données disponibles sur n'importe quel serveur du système. Rumor quant à lui est un système de fichiers qui utilise un modèle pair-à-pair pour les communications (accès aux données et trafic de mise à jour). En plus Rumor opère à un niveau applicatif, ce qui lui permet de minimiser les coûts de réplication et d'avoir une forte portabilité.

### Les systèmes de réplication hybrides

Les systèmes de réplication hybrides tels que Roam [12] qui est basé sur le modèle WARD (*Wide Area Replication Domain model*) combinent les modèles client-serveur et pair-à-pair dans la gestion des données. Une réplication pair-à-pair est utilisée entre les machines serveurs et clients. Ces systèmes essaient de contourner les inconvénients des systèmes pair-à-pair en établissant une connectivité forte entre tous les serveurs. Le modèle hybride permet d'augmenter la disponibilité des données, tout en assurant une propagation rapide des mises à jour (minimise la divergence des répliques.). Cependant, une augmentation très rapide du nombre de répliques et du nombre de mises à jour conflictuelles entraînent une convergence très difficile à atteindre [9].

### Les systèmes de réplication adaptatifs

D'autres travaux, comme [1], [4], [8], [10], [23], [33], ont aussi analysé le problème de la réplication des données en environnements mobiles et proposent l'utilisation d'algorithmes dynamiques pour gérer la réplication et la cohérence des données dans ce type de système. Le modèle de réplication appelé CReam qui place l'utilisateur au centre du système en lui permettant de choisir la quantité de ressources qu'il met à disposition est abordé dans [1]. Dans [23], les auteurs proposent un algorithme de réplication active en fonction des profils utilisateurs (séquences de mobilité des utilisateurs), des modèles d'accès et de lecture/écriture, et reconfigure de manière active les répliques pour les ajuster aux changements de comportement de l'utilisateur et à l'état du système. Cet algorithme est basé sur le modèle « *primary copy* » et sur le protocole « *read-one-write-all* » pour le contrôle de la cohérence. Cependant, son implémentation doit prendre en compte plusieurs problèmes telles que le maintien de l'historique des accès et les profils utilisateurs, la sélection de la copie primaire (site primaire), la limite du nombre de répliques, etc. Un mécanisme de réplication en fonction de la nature des données manipulées dans le système mobile est abordé dans [10]. Selon sa taille, sa structure, son niveau de confidentialité, une donnée pourrait nécessiter un ensemble de ressources (ex : bande passante, espace de stockage, cryptage, etc.) pour être répliquée.

Globalement, les algorithmes de réplication en environnements mobiles requièrent des techniques de réconciliation des copies. Cependant *une gestion automatique des conflits devient très difficile lorsque le nombre de copies devient très important*. C'est dans ce sens que la mise en oeuvre d'outils dédiés à cette tâche fait l'objet d'une attention accrue dans la recherche. IceCube [9] est un exemple d'outil de réconciliation de répliques dans un système mobile.

### 3.2. Gestion de la concurrence

Comme toute application manipulant des données, les applications mobiles doivent pouvoir gérer des accès concurrents et simultanés à une même donnée par différents utilisateurs. La mobilité n'est pas sans conséquence pour la mise en place d'un support transactionnel adéquat.

Les mécanismes transactionnels sont utilisés dans les SGBD (Systèmes de Gestion de Base de Données) pour assurer la cohérence des données en cas d'accès concurrents : gestion de comptes bancaires, réservation de places d'avion, etc. Cependant, certaines caractéristiques des applications mobiles rendent caduques les mécanismes classiques d'ordonnancement des transactions et doivent être prises en compte dans la conception de nouvelles solutions. Des études sur les effets et les impacts de la mobilité sur les modèles transactionnels traditionnels ont été proposées dans [15, 19]. Ces études confirment le besoin de transactions et de mécanismes d'ordonnement adéquats dans les environnements mobiles.

Une transaction mobile peut être définie comme une transaction distribuée dont certaines parties s'exécutent sur des sites statiques et d'autres sur des unités mobiles. Cependant, même si le contexte des applications mobiles est voisin des BDs réparties, certaines caractéristiques essentielles lui sont propres. En particulier, les déplacements dans l'espace ne doivent pas provoquer une rupture des transactions en cours ; les déconnexions et leur durée ne doivent pas peser sur la performance du système. En effet, avec la mobilité, une transaction peut être initiée dans un lieu et se terminer dans un autre lieu. Les déconnexions fréquentes mettent en avant le problème de fiabilité qui devient une exigence primaire du traitement transactionnel mobile.

La déconnexion conduit à la création d'une nouvelle version de données sur l'UM (unité mobile) dont l'intégration doit être assurée par le mécanisme d'ordonnancement des transactions. L'isolation des transactions qui se traduit par la sérialisabilité et qui garantit la cohérence dans les SGBD, est très difficile à prendre en compte au-delà d'un certain délai, surtout avec des temps de déconnexion non bornés. *La rupture de l'isolation entraîne une intégration difficile des versions créées par les transactions concurrentes*. Certains travaux [15, 14] mettent en avant la nécessité de revoir les propriétés transactionnelles (Atomicité, Cohérence, Isolation et Durabilité) et de les adapter aux contraintes des environnements mobiles (déplacement des utilisateurs, déconnexions et changement de réseau de communication). Ainsi, pour répondre aux exigences des systèmes mobiles, de nombreux travaux cherchent à améliorer le support des transactions mobiles. Une analyse approfondie des propositions existantes a été faite dans [14, 15, 19].

Ces études considèrent que la plateforme d'exécution des transactions mobiles est constituée d'hôtes fixes (FHs) ou mobiles (MHs), qui peuvent être des serveurs ou des clients. Les MHs peuvent être de différente nature : des PDAs, des PC portables, etc. Aussi, il n'y a pas de contrainte sur le modèle de données (relationnelle, objet, etc.), ni sur la nature du système de gestion de données. Cependant, les MHs doivent avoir des capacités de stockage de données et doivent pouvoir réaliser les fonctions classiques d'un SGBD.

La suite de cette section décrit les principaux travaux proposés dans la littérature pour la gestion de la concurrence en environnements mobiles. Ces travaux sont regroupés en deux familles en fonction du type de distribution des transactions : exécution sur les serveurs fixes uniquement, ou exécution sur les serveurs fixes et les unités mobiles.

### Les systèmes multibases sur serveurs fixes

Dans cette première famille de travaux (Moflex [20], Pre-serialization [21], Kangaroo [26], MDSTPM,...), le support transactionnel mobile est supposé être basé sur un système multibases (ensemble de bases de données manipulées à l'aide d'un langage commun) sur des serveurs fixes. Les transactions sont donc exécutées sur la partie fixe du réseau. Ainsi, *les propriétés ACID sont respectées dans la plupart des cas*. En effet, cette partie fixe offre des garanties en termes de communication permanente et de ressources suffisantes pour une exécution correcte et sans interruption des transactions. Les protocoles généralement utilisés pour le contrôle de concurrence et la validation sont classiques (ex : approche pessimiste et verrouillage en deux phases). Une transaction peut être validée même si l'unité mobile est déconnectée, et les messages ou résultats en attente sont délivrés à la prochaine reconnexion. Cependant, ces techniques posent un problème de dépendance forte de l'UM à la partie fixe du réseau pour l'exécution d'une transaction. Ainsi, en mode déconnecté (passage dans une zone d'ombre par exemple), le mobile demeure dans un état inactif jusqu'à ce que la connexion soit rétablie, ce qui entraîne un *problème de fiabilité et de disponibilité du service*. Cette dépendance peut aussi s'avérer coûteuse. En effet, lorsque le coût de communication est élevé ou que le débit est bas, l'envoi des transactions aux serveurs et la réception de leurs résultats peut présenter un coût élevé, surtout pour des transactions longues ou nécessitant beaucoup d'échanges de données entre les serveurs et l'unité mobile.

### Les systèmes distribués sur unités mobiles

Comme indiqué ci-dessus, les modèles basés sur les systèmes multibases posent un problème de dépendance forte de l'unité mobile à la partie fixe du réseau. Une deuxième classe de supports transactionnels mobiles (Pro-motion [22], Clustering [24], IOT [31], etc.), basée sur une exécution distribuée des transactions mobiles (possibilité d'une exécution complétée de la transaction au niveau de l'unité mobile), permet d'éviter ce problème. Une partie de la transaction est exécutée sur l'unité mobile, alors que l'autre est traitée sur le réseau fixe. Ce mécanisme offre plus *d'autonomie, de disponibilité, et de flexibilité*, en permettant au mobile de pouvoir faire des opérations localement, même en mode déconnecté et de propager les modifications faites après reconnexion au système central. La plupart des propositions prennent en compte la déconnexion et la mobilité des unités, en intégrant des modules intermédiaires au niveau des stations de base (*proxy, agent ou manager*) qui leur permettent de suivre la localisation des mobiles, de gérer correctement leurs absences lors des déconnexions et de coordonner l'exécution normale des transactions. Certaines propositions essayent de réduire les coûts de communication en acceptant des traitements en environnement faiblement connecté, en organisant les accès aux données suivant la localisation géographique des unités, la sémantique des données, et les besoins des utilisateurs et des applications en termes de cohérence.

Cependant, cette classe est caractérisée par un *relâchement des propriétés ACID* (l'isolation en particulier). Pour gérer la cohérence des données, *des variantes des protocoles de validation et de contrôle de concurrence* [5, 11] sont proposées, ou bien *des protocoles dits optimistes* sont adoptés. Ces protocoles autorisent la manipulation simultanée et concurrente de différentes versions d'une même donnée. Néanmoins, ils requièrent des



phases de synchronisation et de réconciliation pour assurer une cohérence à terme du système.

---

## 4. Conclusion

Dans cet article nous avons analysé les travaux existants dans le domaine de la réplication de données et la gestion de la concurrence dans les environnements mobiles.

La réplication a pour objectif d'augmenter la disponibilité des données et d'améliorer la performance des applications exécutées sur les mobiles. Cependant, une gestion de la convergence à terme des différentes copies est requise. Les problèmes de connectivité et de déconnexion en environnements mobiles augmentent la possibilité de divergence des copies. Les travaux existants proposent des solutions qui permettent de réduire cette divergence et tentent de faciliter la réconciliation des répliques.

En cas d'accès concurrents aux données, il est nécessaire de mettre en place un mécanisme qui garantit la cohérence des données. C'est le rôle des systèmes de gestion de transactions dans les SGBD (systèmes de gestion de bases de données). Mais la transposition de ces systèmes aux environnements mobiles se heurte aux problèmes de déconnexions fréquentes des terminaux et à leur mobilité. Les déconnexions peuvent obliger le système à un relâchement des propriétés ACID des transactions, impactant sur la cohérence globale du système. La mobilité peut constituer un frein à la prise en compte des transactions longues (qui débutent à un endroit et se terminent à un autre endroit). Les solutions proposées dans la littérature répondent assez bien aux problèmes de la mobilité. Pour le problème de déconnexion, les travaux existants essaient de réduire son impact sur la cohérence des données sans sacrifier la performance du système (une cohérence stricte a pour effet de réduire considérablement la concurrence).

La réconciliation des copies est une question importante qui nécessite une connaissance approfondie des données gérées (structure, sémantique, etc.) et des traitements qui leur sont associés. L'automatisation d'un tel processus est nécessaire pour faciliter le travail des agents mobiles et pour éviter tout risque d'incohérence néfaste au système.

---

## 5. Bibliographie

- [1] ZEINA TORBEY AND NADIA BENNANI AND DAVID COQUIL AND LIONEL BRUNIE, « CReaM: User-Centric Replication Model for Mobile Environments », *International Workshop on "Mobile P2P Data Management, Security and Trust (M-PDMST 2010)" in conjunction with the 11th International Conference on Mobile Data Management (MDM 2010)*, page 348-353, 2010.
- [2] LEVENT GÜRGEN AND CLAUDIA RONCANCIO AND CYRIL LABBÉ AND VINCENT OLIVE, « Gestion de données de capteurs », *Ingénierie des Systèmes d'Information*, vol. 1, n° 14, 2009.
- [3] HAGEN HÖPFNER AND ESSAM MANSOUR AND DANIELA NICKLAS, « Review of Data Management Mechanisms on Mobile Devices », *it - Information Technology*, vol. 51, n° 2, page 79-84, 2009.
- [4] ZOUARI, MOHAMED AND ANDRÉ, FRANÇOISE AND SEGARRA, MARIA-TERESA, « Support d'adaptation dynamique et distribuée dans la conception de systèmes de réplication de données », *Proceedings of the 5th French-Speaking Conference on Mobility and Ubiquity Computing*, page 61-64, 2009.
- [5] NOUALI-TABOUDJEMAT, NADIA AND DRIAS, HABIBA, « A policy-based context-aware approach for the commitment of mobile transactions », *Proceedings of the 8th international conference on New technologies in distributed systems*, page 1-11, 2008.
- [6] EVAGGELIA PITOURA AND PANOS K. CHRYSANTHIS, « Caching and Replication in Mobile Data Management », *IEEE Data Eng. Bull.*, n° 30, page 13-20, 2007.
- [7] PH.D ANN CAVOUKIAN, « Technologies de communication sans fil : protection de la vie privée et sécurité », *Commissaire à l'information et à la vie privée/Ontario*, 2007.

- [8] ABAWAJY, J. H. AND DERIS, M. MAT AND OMAR, M., « A novel data replication and management protocol for mobile computing systems », *Mobility. Informations. Systems.*, vol. 2, n° 1, page 3–19, 2006.
- [9] GÉRALD OSTER, « Réplication Optimiste et Cohérence des Données dans les Environnements Collaboratifs Répartis », *Thèse de Doctorat : Université Henri Poincaré - Nancy I*, 2005.
- [10] BELOUED, ABDELKRIM AND GILLIOT, JEAN-MARIE AND SEGARRA, MARIA-TERESA AND ANDRÉ, FRANÇOISE, « Dynamic data replication and consistency in mobile environments », *DSM '05 : Proceedings of the 2nd international doctoral symposium on Middleware*, page 1-5, 2005.
- [11] PATRICIA SERRANO-ALVARADO AND CLAUDIA RONCANCIO AND MICHEL E. ADIBA AND CYRIL LABBÉ, « Modèles, architectures et protocoles pour transactions mobiles adaptables » *Ingénierie des Systèmes d'Information*, vol. 10, n° 5, page 95-121, 2005.
- [12] DAVID RATNER AND PETER L. REIHER AND GERALD J. POPEK, « Roam : A Scalable Replication System for Mobility », *MONET*, vol. 9, n° 5, page 537-544, 2004.
- [13] LINDHOLM, TANCRED, « A three-way merge for XML documents », *Proceedings. DocEng*, 2004.
- [14] PATRICIA SERRANO-ALVARADO AND CLAUDIA RONCANCIO AND MICHEL E. ADIBA, « A Survey of Mobile Transactions », *Distributed and Parallel Databases*, vol. 16, n° 2, page 193-230, 2004.
- [15] GUY BERNARD AND JALEL BEN-OTHTMAN AND LUC BOUGANIM AND GÉRÔME CANALS AND SOPHIE CHABRIDON AND BRUNO DEFUDE AND JEAN FERRIÉ AND STÉPHANE GANÇARSKI AND RACHID GUERRAOUÏ AND PASCAL MOLLI AND PHILIPPE PUCHERAL AND CLAUDIA RONCANCIO AND PATRICIA SERRANO-ALVARADO AND PATRICK VALDURIEZ, « Mobilité et bases de données : état de l'art et perspectives », *TSI : Technique et science informatiques*, vol. 22, n° 4, 2003.
- [17] S MADRIA, B BHARGAVA, M MOHANIA, S BHOWMICK « Data and Transaction Management in a Mobile Environment », *Mobile Computing : Implementing Pervasive Information and Communication Technologies*, 2003.
- [18] UGUR ÇETINTEMEL AND PETER J. KELEHER, « Light-Weight Currency Management Mechanisms in Mobile and Weakly-Connected Environments », *Distributed and Parallel Databases*, vol. 11, n° 1, page 53-71, 2002.
- [19] N.N.TABOUDJEMAR, N. BADACHE, « Transactions Management in Mobile Systems », *Revue d'Information Scientifique et Technique*, vol. 11, n° 1, page 11-32, 2001.
- [20] KYONG-I KU AND YOO-SUNG KIM, « Moflex Transaction Model for Mobile Heterogeneous Multidatabase Systems », *RIDE*, page 39-46, 2000.
- [21] RAVI A. DIRCKZE AND LE GRUENWALD, « A pre-serialization transaction management technique for mobile multidatabases », *MONET*, vol. 5, n° 4, page 311-321, 2000.
- [22] GARY D. WALBORN AND PANOS K. CHRYSANTHIS, « Transaction Processing in PRO-MOTION », page 389-398, 1999.
- [23] SHIOW-YANG WU AND YU-TSE CHANG, « An Active Replication Scheme for Mobile Data Management », *Database Systems for Advanced Applications, Proceedings of the Sixth International Conference on Database Systems for Advanced Applications (DASFAA), April 19-21, Hsinchu, Taiwan*, page 143-150, 1999.
- [24] EVAGGELIA PITOURA AND BHARAT K. BHARGAVA, « Data Consistency in Intermittently Connected Distributed Systems », *IEEE Trans. Knowl. Data Eng.*, vol. 11, n° 6, page 896-915, 1999.
- [25] RICHARD G. GUY AND PETER L. REIHER AND DAVID RATNER AND MICHAL GUNTER AND WILKIE MA AND GERALD J. POPEK, « Rumor : Mobile Data Access Through Optimistic Peer-to-Peer Replication », *ER Workshops*, page 254-265, 1998.
- [26] DUNHAM, MARGARET H. AND HELAL, ABDELSALAM AND BALAKRISHNAN, SANTOSH, « A mobile transaction model that captures both the data and movement behavior », *Mob. Netw. Appl.*, vol. 2, n° 2, page 149-162, 1997.
- [27] JIM GRAY AND PAT HELLAND AND PATRICK E. O'NEIL AND DENNIS SHASHA, « The Dangers of Replication and a Solution », *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, Montreal, Quebec, Canada, June 4-6, 1996*, page 173-182, 1996.
- [28] A. BAGGIO, « Environnements Mobiles : Etude et Synthèse Bibliographique », *Mémoire de DEA, INRIA Rocquencourt - SOR Project*, 1995.
- [29] ALAN DEMERS AND KARIN PETERSEN AND MIKE SPREITZER AND DOUGLAS TERRY AND MARVIN THEIMER AND BRENT WELCH, « The Bayou Architecture : Support for Data Sharing among Mobile Users », 1994.
- [30] GEORGE FORMAN AND JOHN ZAHORIAN, « The Challenges of Mobile Computing », *IEEE Computer*, vol. 27, n° 4, page 38-47, 1994.
- [31] QI LU AND MAHADEV SATYANARAYANAN, « Isolation-Only Transactions for Mobile Computing », *Operating Systems Review*, vol. 28, n° 2, page 81-87, 1994.
- [32] EVAGGELIA PITOURA AND BHARAT BHARGAVA, « Dealing with Mobility : Issues and Research Challenges », 1993.
- [33] D. BARBARA AND H. GARCIA-MOLINA, « Replicated Data Management in Mobile Environments : Anything New Under the Sun ? », *Technical Report, Stanford University*, n° 2, 1993.
- [34] MAHADEV SATYANARAYANAN AND JAMES J. KISTLER AND PUNEET KUMAR AND MARIA E. OKASAKI AND ELLEN H. SIEGEL AND DAVID AND C. STEERE, « Coda : A Highly available File System for a Distributed Workstation Environment », *IEEE Transactions on Computers*, vol. 39, page 447–459, 1990.