

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

Suivi des versions-révisions et des validations du document.			
<p>Ce document annule et remplace tout document diffusé de version-révision antérieure.</p> <p>Dès réception de ce document, les destinataires ont pour obligation de détruire les versions-révisions antérieures, toutes les copies, et de les remplacer par cette version.</p> <p>Si les versions-révisions antérieures sont conservées pour mémoire, les destinataires doivent s'assurer qu'elles ne peuvent être confondues avec cette présente version-révision dans leur usage courant.</p>			
Version.	Date.	Auteurs.	Création, modification ou validation.
A	23 nov. 2003.	JPD.	Création.
B	24 mar. 2004	JPD.	Mise à jour pour les canaux.

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

1 Tables

1.1 Table des matières

1	Tables	2
1.1	Table des matières	2
1.2	Table des illustrations	3
2	Références	4
2.1	Glossaire	4
2.2	Ressources	4
3	Introduction	5
3.1	Objet du document	5
3.2	Audience	5
3.3	Pré-requis	5
4	Interaction avec l'environnement	6
4.1	Description	6
4.2	Paramètres	6
4.3	Particularités	6
4.3.1	Compilation	6
4.3.2	Exécution	6
4.4	Application Program Interfaces	6
5	Choix techniques	7
5.1	Communication par messages	7
5.1.1	Principe de la communication	7
5.1.2	Empaquetage des paramètres	9
5.1.3	Format d'un message	9
5.2	Pilotes de communication	14
5.3	Gestion des canaux de communication	15
5.3.1	Principe	16
5.3.2	Propriétés d'un canal	16
5.3.3	Méthodes d'un canal	17
6	Modèle de données	20
7	Composants techniques	21

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

1.2 Table des illustrations

Diagramme 1 – Principe de la communication via <i>Up ! Network</i>	8
Tableau 2 – Glossaire du modèle physique des données publiques du module <i>Up ! Network</i>	9
Tableau 3 – Glossaire des messages d' <i>Up ! Network</i>	13
Diagramme 4 – Processus de communication entre le client et le serveur	19
Diagramme 5 – Modèle physique des données publiques du module <i>Up ! Network</i>	20
Tableau 6 – Glossaire du modèle physique des données publiques du module <i>Up ! Network</i>	20
Tableau 7 – Composants techniques du module	22

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

2 Références

2.1 Glossaire

Liste des définitions des termes employés.	
Ce tableau recense tous les termes, les concepts particuliers ainsi que les abréviations employés dans ce document.	
Terme, concept, abrégé.	Définition du terme, du concept ou de l'abréviation.
Marshalling	Voir page 9.
Unmarshalling	Voir page 9.

2.2 Ressources

Liste des documents applicables et en référence.		
Un document est applicable à partir du moment où son contenu est validé et que l'activité ou le projet fait partie de son périmètre d'application. Il est obligatoire d'appliquer son contenu.		
Un document est en référence à partir du moment où son contenu n'est pas validé ou que l'activité ou le projet ne fait partie de son périmètre d'application. Il est recommandé d'appliquer son contenu mais cela n'est pas obligatoire.		
Un document applicable est indiqué par A1, A2, A3 , etc. Un document en référence est indiqué par R1, R2, R3 , etc.		
Index.	Nom du document.	Commentaire.
A1	UpComp-Plan Qualité-000005	Méthode documentaire.
A2	UpComp-Plan Qualité-000006	Processus de management de projet.
A3	UpComp-Plan Qualité-000046	Méthode de spécification technique d'un module.
A4	UpComp-UpsNet-000002	Plan documentaire du projet.
A5	UpComp-UpsVm-000003	Plan de programmation.
A6	UpComp-UpsVm-000004	Programmation en C-- .
R7	http://www.up-comp.com	Site Internet d' Up ! Application System .
A8	UpComp-UpsOms-000003	Spécification technique du module UpsOms .
A9	UpComp-UpsKrn-000003	Spécification technique du module UpsKrn .

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

3 Introduction

3.1 Objet du document

L'objet de ce document est de décrire le contenu technique du module logiciel **Up ! Network** pour le projet **Up ! Application System**.

Ce document est rédigé et approuvé par la **Maîtrise d'Oeuvre (MOE)**.

3.2 Audience

Ce document s'adresse aux :

- **Directeurs de projets et chefs de projets.**
Pour la compréhension du module technique.
- **Ingénieurs de développement.**
Pour savoir comment est conçu le module technique.

Pour aider ces personnes à remplir le document **Spécification technique d'un module**, leur manager et la cellule de support projet se tiennent à leur disposition.

3.3 Pré-requis

Le pré-requis est la connaissance des documents suivants :

- **Méthode documentaire** [A1].
- **Processus de management de projet** [A2].
- **Méthode de spécification technique d'un module** [A3].

Nous rappelons que tous les documents applicables ou référencés pour le projet **Up ! Application System** sont tracés dans le **Plan documentaire** [A4].

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

4 Interaction avec l'environnement

4.1 Description

L'objet du module logiciel *Up ! Network* est de

- Fournir un adaptateur client pour les appels de service sur le réseau.
Toutes les fonctions utilisant implicitement ou explicitement l'opérateur @ utilisent cet adaptateur.
- Fournir un adaptateur serveur pour les services sur le réseau.
Toutes les fonctions désignées par l'opérateur @ sont appelées par cet adaptateur.
- Fournir des services de communication.
Ils sont indépendants du moyen de communication retenue.
- Mettre en oeuvre :
 - Les interfaces.
Il y a *Canal*.
 - Les types.
Il y a *Serveur*.

4.2 Paramètres

Tous les paramètres sont documentés sur le [Site Internet d'Up ! Network](#) [R7].

4.3 Particularités

4.3.1 Compilation

Néant.

4.3.2 Exécution

Néant.

4.4 Application Program Interfaces

Néant.

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

5 Choix techniques

5.1 Communication par messages

Cette section décrit comment la communication inter-module est réalisée par **Up ! Network**.

Les différentes alternatives possibles sont les suivantes :

- Utiliser les **Remote Procedure Call (RPC)**.
- Utiliser un mécanisme d'appel distant propriétaire.

La seconde alternative a été retenue pour la raison suivante :

- **Indépendance par rapport aux moyens de communication.**
Ces **Application Program Interfaces (API)** sont la propriété de **Sun Microsystem**. Elles ne sont pas indépendantes du moyen de communication.
- **Portabilité.**
Ces **Application Program Interfaces (API)** ne sont pas portables sur tous les systèmes d'exploitation.
- **Opérations de bas niveau.**
Des opérations de bas niveau sont nécessaires pour synchroniser le contenu des entrepôts publics. Elles sont réalisées à la demande de **Up ! Object Management System**.

Compte-tenu de la troisième raison, il n'est pas possible de changer de choix.

5.1.1 Principe de la communication

La communication via **Up ! Network** s'effectue par messages envoyés dans le schéma client-fournisseur. Le client est l'adaptateur client d'**Up ! Network** qui se charge de créer le message de requête. Le fournisseur est l'adaptateur serveur d'**Up ! Network** qui se charge de créer le message de réponse.

Une fois le message constitué, un moyen de communication est employé. A ce jour, cela peut être :

- **Les socquettes sur Tcp-Ip.**
Elles permettent de faire communiquer deux machines reliées par un réseau **Tcp-Ip**.
Le pilote est **Up ! Tcp-Ip**.
- **Microsoft Network.**
Il permet de faire communiquer deux machines exploitées par **Windows**.
Le pilote est **Up ! Microsoft Network**.
- **Les tuyaux nommés.**
Ils permettent de faire communiquer deux applications sur la même machine.
Le pilote est **Up ! Named pipes**.

Voici un diagramme synthétisant le principe de la communication :



Spécification technique du module

Diffusion restreinte

Date rédaction :
7 avril 2004.

Date validation :

Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc

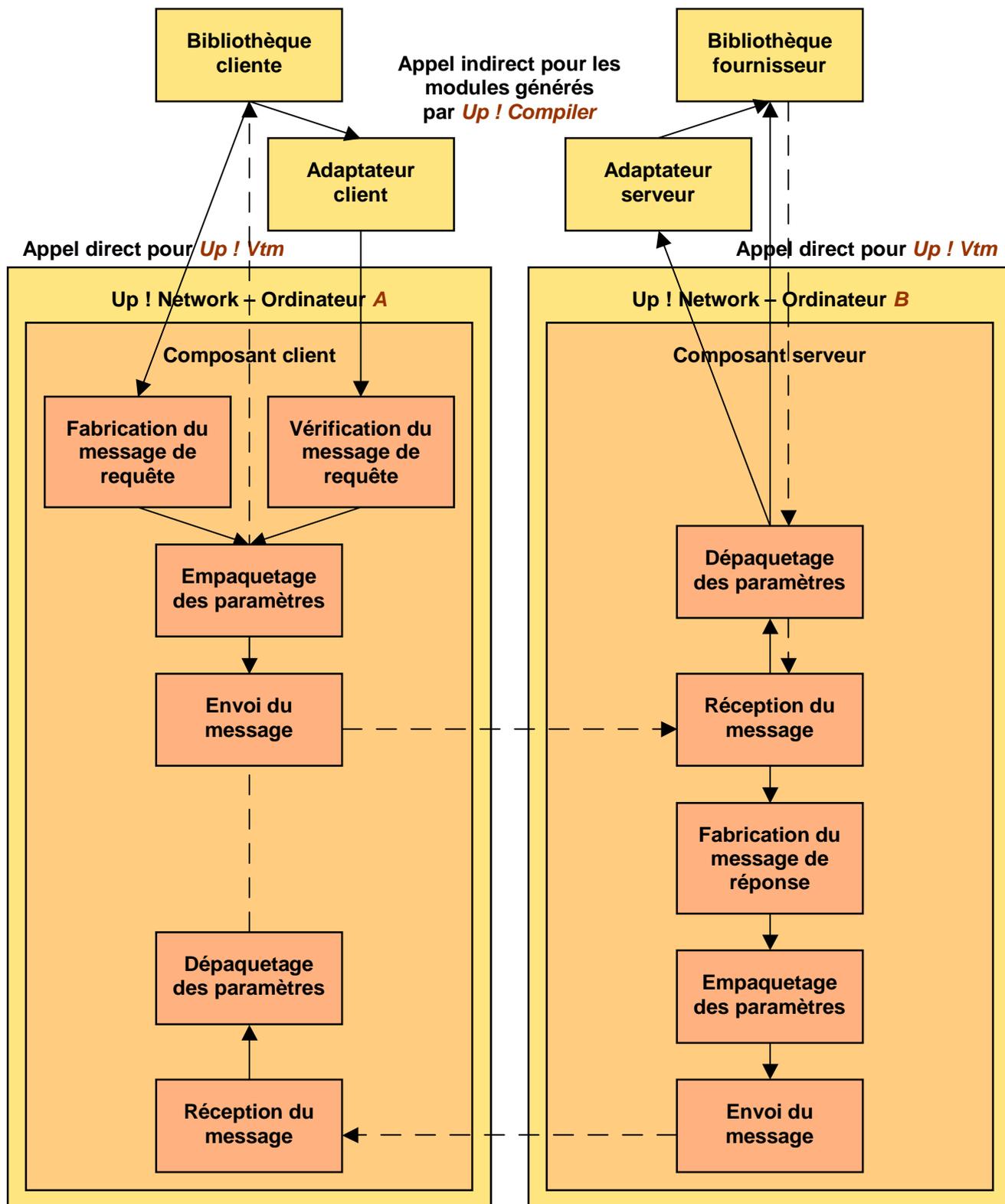


Diagramme 1 – Principe de la communication via Up! Network

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

5.1.2 Empaquetage des paramètres

& L'empaquetage des paramètres est réalisé via la procédure d'exportation d'*Up ! Object Management System*. Cette opération est appelée **marshalling** dans les standards *Corba* ou *DCom*.

& Le dépaquetage des paramètres est réalisé via la procédure d'importation d'*Up ! Object Management System*. Cette opération est appelée **unmarshalling** dans les standards *Corba* ou *DCom*.

Pour plus de précisions, merci de se référer à **Spécification technique du module UpsOms** [A8].

5.1.3 Format d'un message

Le type des messages est *UpsNet.ProtocoleUpsNet*. Le format est le suivant :

- **En-tête du message.**
Il contient les champs suivants qui sont tous obligatoires :

Entité.	Description.
Correction.	Numéro de correction d' <i>Up ! Virtual Technical Machine</i> de l'émetteur.
EstBigEndian.	Si <i>Vrai</i> , la plate-forme de l'émetteur est dans le format <i>Big endian</i> pour l'ordre des octets des entiers.
NbBits.	Nombre de bits d'adressage de la plate-forme cible – 16, 32 ou 64.
NumeroServeur.	Numéro de serveur de l'émetteur. A défaut, 0.
PageDeCode.	Page de code du système d'exploitation.
Revision.	Numéro de révision d' <i>Up ! Virtual Technical Machine</i> de l'émetteur.
Token.	Identificateur du message. Le corps du message dépend de sa valeur.
Version.	Numéro de version d' <i>Up ! Virtual Technical Machine</i> de l'émetteur.

Tableau 2 – Glossaire du modèle physique des données publiques du module *Up ! Network*

- **Identificateur du message.**
Les identificateurs du message sont les suivants :

Identificateurs.	Module concerné.	Description.
KN_Reponse	Tous.	Message de réponse à une requête.
KN_MonterSegment	UpsOms.	Monte un segment de mémoire.
KN_AllouerSegmentMobile	UpsOms.	Alloue un nouveau segment de mémoire mobile.
KN_AllouerCache	UpsOms.	Alloue un nouveau cache.
KN_LibererCache	UpsOms.	Libère un cache préalablement alloué.
KN_OuvrirSession	UpsOms.	Ouvre une session à distance.
KN_FermerSession	UpsOms.	Ferme une session à distance.
KN_Connecter	UpsOms.	Connexion au serveur.
KN_Deconnecter	UpsOms.	Déconnexion du serveur.
KN_ArretServeur	UpsOrb.	Demande l'arrêt du serveur.



Spécification technique du module

Date rédaction :
7 avril 2004.

Diffusion restreinte

Date validation :

Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc

Identificateurs.	Module concerné.	Description.
KN_UtiliserObjet	UpsOms.	Spécifie qu'objet local a une référence vers un objet distant.
KN_NePlusUtiliserObjet	UpsOms.	Spécifie qu'objet local n'a plus une référence vers un objet distant.
KN_LibererObjet	UpsOms.	Libère l'espace mémoire utilisé par l'objet distant.
KN_RetasserType	UpsOms.	Compacte la mémoire utilisée par les objets du type.
KN_RechercherAdresse	UpsOms.	Recherche une adresse d'un bloc mémoire et pose un verrou dessus.
KN_LibererVerrou	UpsOms.	Libère un verrou sur un bloc mémoire.
KN_RechercherImageAvant	UpsOms.	Recherche l'image avant d'un objet distant.
KN_AllouerTransaction	UpsOms.	Alloue une transaction distante
KN_ConserverImageAvant	UpsOms.	Conserve l'image avant d'un objet distant.
KN_TenterDeValiderTransaction	UpsOms.	Vérifie s'il est possible de valider la transaction à distance au regard des contraintes d'intégrité.
KN_ValiderTransaction	UpsOms.	Valide la partie distante de la transaction.
KN_InvalidierTransaction	UpsOms.	Invalide la partie distante de la transaction.
KN_VerrouillerTransaction	UpsOms.	Verrouille toutes les transactions en préparation à la validation distribuée.
KN_DeVerrouillerTransaction	UpsOms.	Déverrouille toutes les transactions en préparation à la validation distribuée.
KN_Photographier	UpsOms.	Pose une photographie sur les entrepôts distants.
KN_LireConfiguration	UpsOrb.	Fait lire la configuration au serveur local Up ! Object Request Broker parce qu'elle vient d'être modifiée.
KN_AjouterNoeud	UpsOrb.	Ajoute un nœud à la topologie.
KN_SupprimerNoeud	UpsOrb.	Supprime un nœud de la topologie.
KN_DemarrerNoeud	UpsOrb.	Démarré un nœud de la topologie.
KN_ArrêterNoeud	UpsOrb.	Arrête un nœud de la topologie.
KN_AjouterServeur	UpsOrb.	Ajoute un serveur un nœud.
KN_SupprimerServeur	UpsOrb.	Supprime un serveur d'un nœud.
KN_StatistiquesServeur	UpsOrb.	Envoie au serveur père les statistiques d'utilisation.
KN_AjouterModule	UpsOrb.	Ajoute un module au serveur.
KN_SupprimerModule	UpsOrb.	Supprime un module d'un serveur.
KN_ChercherModule	UpsOrb.	Cherche un serveur disponible qui gère le module demandé.
KN_ChercherServeur	UpsOrb.	Cherche un serveur disponible.
KN_PrendreJetonLicence	UpsSec.	Prends un jeton sur une licence.
KN_LacherJetonLicence	UpsSec.	Lâche un jeton sur une licence.

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :

Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc

Identificateurs.	Module concerné.	Description.
KN_ProcedureSansParametre	UpsNet.	Appel d'une procédure sans paramètre.
KN_ProcedureAvecParametre	UpsNet.	Appel d'une procédure avec paramètre.
KN_FonctionEnumereCourtSansParametre	UpsNet.	Appel d'une fonction retournant un <i>TypUpsVmUnsignedChar</i> sans paramètre.
KN_FonctionEnumereCourtAvecParametre	UpsNet.	Appel d'une fonction retournant un <i>TypUpsVmUnsignedChar</i> avec paramètre.
KN_FonctionEnumereLongSansParametre	UpsNet.	Appel d'une fonction retournant un <i>TypUpsVmUnsignedShort</i> sans paramètre.
KN_FonctionEnumereLongAvecParametre	UpsNet.	Appel d'une fonction retournant un <i>TypUpsVmUnsignedShort</i> avec paramètre.
KN_FonctionEntierSansParametre	UpsNet.	Appel d'une fonction retournant un <i>TypUpsVmLong</i> sans paramètre.
KN_FonctionEntierAvecParametre	UpsNet.	Appel d'une fonction retournant un <i>TypUpsVmLong</i> avec paramètre.
KN_FonctionReelSansParametre	UpsNet.	Appel d'une fonction retournant un <i>TypUpsVmDouble</i> sans paramètre.
KN_FonctionReelAvecParametre	UpsNet.	Appel d'une fonction retournant un <i>TypUpsVmDouble</i> avec paramètre.
KN_FonctionObjetSansParametre	UpsNet.	Appel d'une fonction retournant un objet sans paramètre.
KN_FonctionObjetAvecParametre	UpsNet.	Appel d'une fonction retournant un objet avec paramètre.
KN_MethodeSansParametre	UpsNet.	Appel d'une méthode procédurale sans paramètre.
KN_MethodeAvecParametre	UpsNet.	Appel d'une méthode procédurale avec paramètre.
KN_MethodeEnumereCourtSansParametre	UpsNet.	Appel d'une méthode retournant un <i>TypUpsVmUnsignedChar</i> sans paramètre.
KN_MethodeEnumereCourtAvecParametre	UpsNet.	Appel d'une méthode retournant un <i>TypUpsVmUnsignedChar</i> avec paramètre.
KN_MethodeEnumereLongSansParametre	UpsNet.	Appel d'une méthode retournant un <i>TypUpsVmUnsignedShort</i> sans paramètre.
KN_MethodeEnumereLongAvecParametre	UpsNet.	Appel d'une méthode retournant un <i>TypUpsVmUnsignedShort</i> avec paramètre.
KN_MethodeEntierSansParametre	UpsNet.	Appel d'une méthode retournant un <i>TypUpsVmLong</i> sans paramètre.
KN_MethodeEntierAvecParametre	UpsNet.	Appel d'une méthode retournant un <i>TypUpsVmLong</i> avec paramètre.
KN_MethodeReelSansParametre	UpsNet.	Appel d'une méthode retournant un <i>TypUpsVmDouble</i> sans paramètre.
KN_MethodeReelAvecParametre	UpsNet.	Appel d'une méthode retournant un <i>TypUpsVmDouble</i> avec paramètre.



Spécification technique du module

Date rédaction :
7 avril 2004.

Diffusion restreinte

Date validation :

Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc

Identificateurs.	Module concerné.	Description.
KN_MethodeObjetSansParametre	UpsNet.	Appel d'une méthode retournant un objet sans paramètre.
KN_MethodeObjetAvecParametre	UpsNet.	Appel d'une méthode retournant un objet avec paramètre.
KN_ConstructeurSansParametre	UpsNet.	Appel d'un constructeur sans paramètre.
KN_ConstructeurAvecParametre	UpsNet.	Appel d'un constructeur avec paramètre.
KN_Destructeur	UpsNet.	Appel d'un destructeur sur l'objet distant.
KN_MethodeAffecter	UpsNet.	Appel de la méthode = sur l'objet distant.
KN_MethodeEgal	UpsNet.	Appel de la méthode == sur l'objet distant.
KN_MethodeDifferent	UpsNet.	Appel de la méthode != sur l'objet distant.
KN_MethodeCloner	UpsNet.	Appel de la méthode Cloner sur l'objet distant.
KN_MethodeExporter	UpsNet.	Appel de la méthode Exporter sur l'objet distant.
KN_MethodeImporter	UpsNet.	Appel de la méthode Importer sur l'objet distant.
KN_MethodeLirePropriete	UpsNet.	Appel de la méthode LirePropriete sur l'objet distant.
KN_MethodeEcrirePropriete	UpsNet.	Appel de la méthode EcrirePropriete sur l'objet distant.
KN_MethodeIncrementerDecrementerPropriete	UpsNet.	Appel de la méthode IncrementerDecrementerPropriete sur l'objet distant.
KN_MethodeSupprimerPropriete	UpsNet.	Appel de la méthode SupprimerPropriete sur l'objet distant.
KN_MethodeEnumererProprietes	UpsNet.	Appel de la méthode EnumererProprietes sur l'objet distant.
KN_MethodeEnumererMethodes	UpsNet.	Appel de la méthode EnumererMethodes sur l'objet distant.
KN_MethodeAllouerRessource	UpsNet.	Appel de la méthode AllouerRessources sur l'objet distant.
KN_MethodeLibererRessource	UpsNet.	Appel de la méthode LibererRessources sur l'objet distant.
KN_MethodeComposer	UpsNet.	Appel de la méthode = sur l'objet distant avec la composition d'une opération.
KN_MethodeAffecterTypeParametre	UpsNet.	Appel de la méthode = d'un paramètre d'un type pour un objet distant.
KN_MethodeEgalTypeParametre	UpsNet.	Appel de la méthode == d'un paramètre d'un type pour un objet distant.
KN_MethodeDifferentTypeParametre	UpsNet.	Appel de la méthode != d'un paramètre d'un type pour un objet distant.
KN_MethodeContrainte	UpsNet.	Appel de la méthode Contraine sur l'objet distant.
KN_Exporter	UpsOms.	Exportation en local d'un objet distant.

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :

Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc

Identificateurs.	Module concerné.	Description.
KN_Importer	UpsOms.	Importation en local d'un objet distant.
KN_RemettreDansImport	UpsOms.	Remise d'octets dans le tampon d'importation.
KN_LireAdresseType	UpsNet.	Lecture de l'adresse d'un objet représentant un type.
KN_LireAdresseException	UpsNet.	Lecture de l'adresse d'un objet représentant une exception.
KN_LireAdresseEntrepot	UpsNet.	Lecture de l'adresse d'un objet représentant un entrepôt.
KN_LireAdresseFile	UpsNet.	Lecture de l'adresse d'un objet représentant une file d'attente.
KN_LireAdresseModule	UpsNet.	Lecture de l'adresse d'un objet représentant un module.
KN_LireVariable	UpsNet.	Lecture de la valeur d'une variable distante.
KN_EcrireVariable	UpsNet.	Ecriture de la valeur d'une variable distante.
KN_LireEstInstanceDe	UpsNet.	Lecture de la propriété EstInstanceDe d'un objet distant.
KN_IncrementerDecrementerVariable	UpsNet.	Incrémente ou décrémente une variable distante.
KN_DetruireChampObjet	UpsNet.	Destruction des champs d'un objet distant.
KN_ExtraireEnumereCourt	UpsKrn.	Extrait une valeur d'un énuméré sur TypUpsVmUnsignedChar d'un objet énuméré distant.
KN_ExtraireEnumereLong	UpsKrn.	Extrait une valeur d'un énuméré sur TypUpsVmUnsignedShort d'un objet énuméré distant.
KN_ExtraireEntier	UpsKrn.	Extrait une valeur d'un entier sur TypUpsVmLong d'un objet entier distant.
KN_ExtraireReel	UpsKrn.	Extrait une valeur d'un réel sur TypUpsVmDouble d'un objet réel distant.
KN_LireValeurVariable	UpsKrn.	Lecture de la valeur d'une variable partagée.
KN_LireEntreeAnnuaireGlobal	UpsOrb.	Lit une entrée de l'annuaire global.
KN_EcrireEntreeAnnuaireGlobal	UpsOrb.	Ecrit une entrée de l'annuaire global.
KN_SupprimerEntreeAnnuaireGlobal	UpsOrb.	Supprime une entrée de l'annuaire global.
KN_ModifierEntreeAnnuaireGlobal	UpsOrb.	Modifie une entrée de l'annuaire global.
KN_LireArborescenceAnnuaireGlobal	UpsOrb.	Lit un niveau d'arborescence de l'annuaire global.
KN_TransfererObjet	UpsOrb.	Transfère un objet encapsulant une ressource vers l'annuaire global.
KN_Exception	Tous.	Envoie une erreur lors de l'interprétation d'un message.

Tableau 3 – Glossaire des messages d'Up! Network

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

- **Corps du message.**
Il s'agit de union appelée **Origine** qui est spécialisée par identificateur de message. Pour plus de précisions, merci de se référer au fichier **upsnet.e**.

- **Paramètres du message.**
Les paramètres sont exportés dans un complément joint au message. Les paramètres sont à destination d'un appel identifié par :

- Le nom du module.
- Les numéros de version, de révision et de correction d'application du module.
- L'index de l'appel.
- L'index du co-appel.

Pour plus de précision sur l'index des appels dans les interfaces de traitements, merci de se référer à **Spécification technique du module UpsKrn** [A9].

a

L'index du co-appel est particulièrement important puisqu'il permet d'exporter ou d'importer les paramètres d'une procédure, d'une fonction ou d'une méthode. Il permet d'identifier l'appel jumeau à la procédure, la fonction ou la méthode faisant partie de la bibliothèque dictionnaire du module. Cet appel jumeau énumère les paramètres d'une procédure, d'une fonction ou d'une méthode ainsi que leurs caractéristiques.

Pour plus de précision sur l'index des appels dans les interfaces de traitements, merci de se référer à **Spécification technique du module UpsKrn** [A9].

- **Résultat du message de requête.**
Le résultat du message peut être :
 - Un énuméré de type **TypUpsVmUnsignedChar**.
 - Un énuméré de type **TypUpsVmUnsignedShort**.
 - Un entier de type **TypUpsVmLong**.
 - Un réel de type **TypUpsVmDouble**.
 - Un objet.
 - Un message d'erreur sous forme de chaîne de caractère.
Pour le message d'identification **KN_Exception**.
 - Un complément dans un tampon spécifique.
Pour les messages de **Up ! Objet Management System** uniquement.

5.2 Pilotes de communication

Pour offrir ses services, **Up ! Network** s'appuie sur des pilotes de communication au comportement indépendant de leur technologie. A ce jour, il y a trois pilotes :

- **Up ! Microsoft Network.**
Pour une communication via **Netbui**.
- **Up ! Transmission Control Protocol.**
Pour une communication via **Transmission Control Protocol / Internet Protocol (TCP / IP)**.
- **Up ! Named Pipes.**
Pour une communication via **Named pipes**.

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

L'interface de traitements d'un pilote de communication est une extension de *TypUpsNetXTraitements* décrite dans le fichier *upsnet.e*. Voici les *Application Program Interfaces (API)* de cette interface :

- **Initialiser.**
Initialise le pilote.
- **InitialiserCanal.**
Initialise le canal pour une nouvelle communication.
- **InitialiserCanalClient.**
Initialise le canal pour un client accepté par le scrutateur.
- **InitialiserCanalEnLocal.**
Initialise le canal pour travailler en local et non en réseau.
- **InitialiserCanalOrbLocal.**
Initialise le canal pour *Up ! Object Request Broker* local.
- **InitialiserCanalScrutateur.**
Initialise le canal pour la tâche *Scrutateur*.
- **EnregistrerParametresCanalOrb.**
Enregistre les paramètres d'*Up ! Object Request Broker*.
- **EstEnLocal.**
Retourne *Vrai* si le canal opère en local.
- **LireParametresCanal.**
Lit les paramètres du canal

5.3 Gestion des canaux de communication

Cette section décrit comment la gestion des canaux de communication est réalisée par *Up ! Network*.

Les différentes alternatives possibles sont les suivantes :

- Utiliser les canaux du **run-time** du langage **C**.
- Utiliser un mécanisme de flux propriétaire.

La seconde alternative a été retenue pour la raison suivante :

- **Limitation des flux du run-time du langage C.**
Ils s'appliquent uniquement aux socquettes et aux tuyaux nommés. Le mécanisme n'est pas ouvert pour être étendu à d'autres supports et n'est pas compatible avec les flux.
- **Sûreté de fonctionnement.**
Ces *Application Program Interfaces (API)* ne comportent pas de mécanisme suffisant visant à contrôler la sûreté de fonctionnement.
- **Portabilité.**
Ces *Application Program Interfaces (API)* ne sont pas complètement portables dans leur sémantique.

Compte-tenu de la première raison, il n'est pas possible de changer de choix étant donné de la montée en puissance d'*Internet* et de la nécessité de l'aisance de programmation pour ces technologies mouvantes.

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

5.3.1 Principe

Les canaux sont des services de haut niveau de gestion de connexion, de lecture et d'écriture d'informations sur une ligne de communication et indépendamment de celui-ci.

Les supports gérés à ce jour sont :

- Les socquettes.
- Les tuyaux nommés.
- La communication via **Netbeui**.

Un canal est modélisé par le type **TypUpsNetCanal**. Les services de haut niveau reposent sur les fonctions de bas niveau suivantes, que le support doit mettre en oeuvre au travers de l'interface **TypUpsNetMethodesCanal** :

- **AccepterUnClient.**
Accepte un nouveau client.
- **CalculerPlusPetitCanal.**
Compare les paramètres des canaux pour trouver un canal non utilisé.
- **Connecter.**
Connecte le client à un serveur.
- **Copier.**
Copie les informations du canal.
- **Decoder.**
Décode les champs entiers du canal.
- **Detruire.**
Détruit le canal de communication.
- **EcouterClients.**
Configure le canal pour traiter un certain nombre de clients au plus.
- **EnregistrerParametres.**
Enregistre les paramètres d'un nœud.
- **Exporter.**
Exporte le canal de communication.
- **Importer.**
Importe le canal de communication.
- **MethodesFlux.**
Ensemble des méthodes des flux.
- **FormaterParametres.**
Formate les paramètres du canal.
- **RecupererInformationsSession.**
Récupère les informations de session du client.

5.3.2 Propriétés d'un canal

Les services de haut niveau énuméré prennent en compte les caractéristiques suivantes des canaux, auxquelles il faut ajouter celles offertes par les flux :

- **Un nom de protocole.**
Mot-clé identifiant le module mettant en oeuvre le canal.

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

- **Une adresse.**
Identifiant du destinataire de la communication.

5.3.3 Méthodes d'un canal

Voici les services de haut niveau offert par un canal, auxquels il faut ajouter ceux offerts par les flux :

- **AccepterUnClient.**
Accepte un nouveau client.
- **AllouerCanal.**
Alloue un port de communication dans une plage réservée.
- **CalculerPlusPetitCanal.**
Compare les paramètres des canaux pour trouver un canal non utilisé.
- **CanauxEgaux.**
Retourne **Vrai** si les caractéristiques des canaux sont identiques.
- **Connecter.**
Connecte le client à un serveur.
- **CopierCanal.**
Copie les informations du canal.
- **DecoderAdresse.**
Décode une adresse complète de type **Url** en un canal.
- **DecoderCanal.**
Décode les champs entiers du canal.
- **DeplacerCanal.**
Déplace les informations d'un canal d'un descripteur à l'autre.
- **Detruire.**
Détruit le canal de communication.
- **EcouterClients.**
Configure le canal pour traiter un certain nombre de clients au plus.
- **EnregistrerParametresCanal.**
Enregistre les paramètres du nœud.
- **EnregistrerParametresCanalOrb.**
Enregistre les paramètres d'**Up ! Object Request Broker**.
- **EstEnLocal.**
Retourne **Vrai** si le canal opère en local.
- **Exporter.**
Exporte le canal de communication.
- **FormaterAdresseCanal.**
Formate le libellé de l'adresse pour un message d'erreur.
- **InitialiserCanal.**
Initialise le canal pour une nouvelle communication.
- **InitialiserCanalClient.**
Initialise le canal pour un client accepté par le scrutateur.

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

- **InitialiserCanalEnLocal.**
Initialise le canal pour travailler en local et non en réseau.
- **InitialiserCanalOrbLocal.**
Initialise le canal pour **Up ! Object Request Broker** local.
- **InitialiserCanalScrutateur.**
Initialise le canal pour la tâche **Scrutateur**.
- **LireCanalMachine.**
Lit le canal pour communiquer avec la machine
- **Importer.**
Importe le canal de communication.
- **LireParametresCanal.**
Lit les paramètres du canal
- **OuvrirUnCanal.**
Ouvre un canal pour pouvoir communiquer.
- **RecupererInformationsSession.**
Récupère les informations de session du client.

Les prototypes de ces **Application Program Interfaces (API)** sont décrits dans le fichier **upsnet.e**.

Voici le processus de communication entre un client et un serveur basé sur ces **Application Program Interfaces (API)** :



Spécification technique du module

Date rédaction :
7 avril 2004.

Diffusion restreinte

Date validation :

Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc

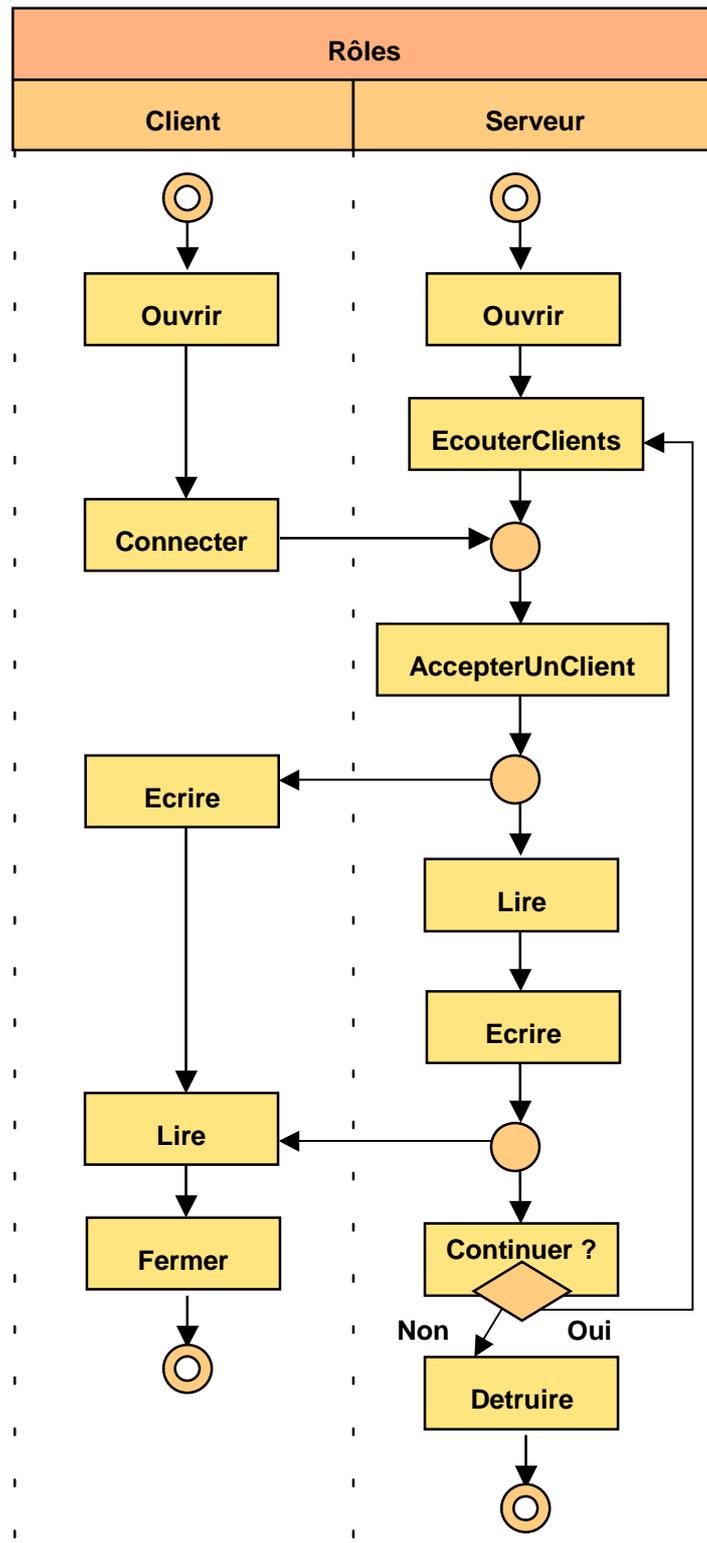


Diagramme 4 – Processus de communication entre le client et le serveur

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

6 Modèle de données

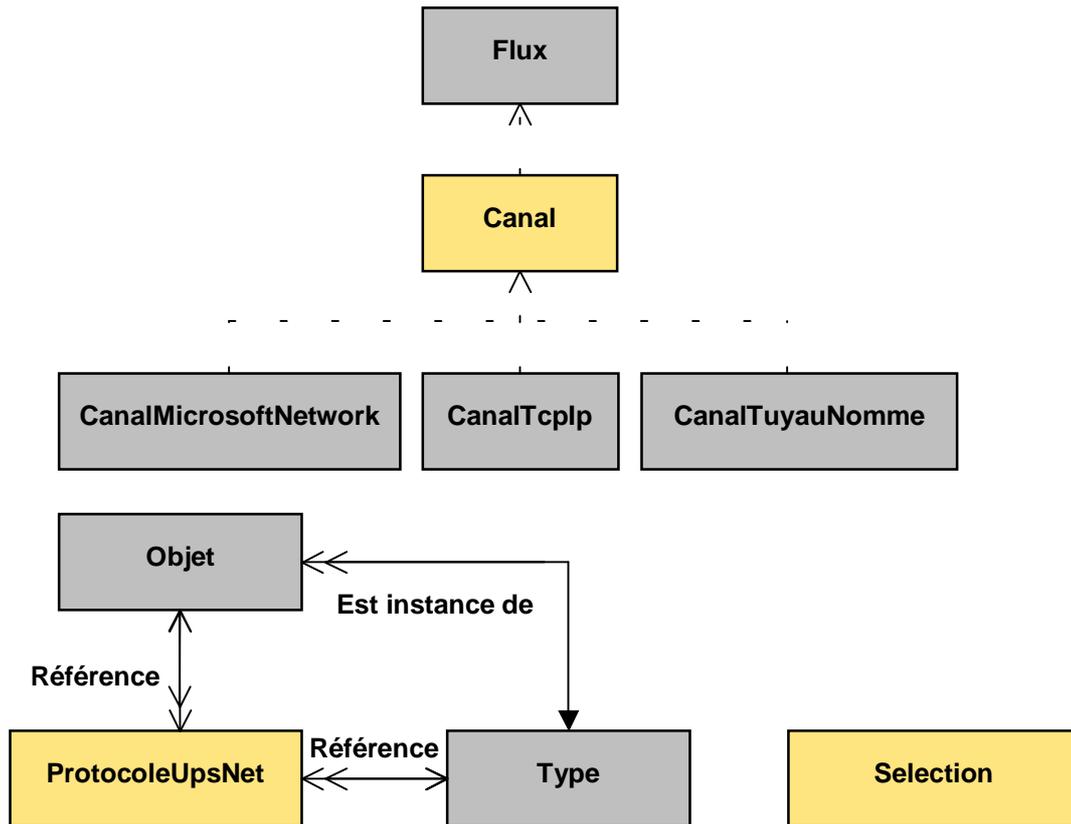


Diagramme 5 – Modèle physique des données publiques du module *Up ! Network*

Toutes les entités suivantes sont décrites dans le fichier **upsnet.e** :

Entité.	Description.
Canal.	Canal d'échange d'informations.
CanalMicrosoftNetwork.	Canal d'échange d'informations via <i>Microsoft Netbui</i> .
CanalTcplp.	Canal d'échange d'informations via <i>Transmission Control Protocol</i> .
CanalTuyauNomme.	Canal d'échange d'informations via <i>Named Pipes</i> .
Flux.	Flux de lecture ou d'écriture d'informations.
ProtocoleUpsNet.	Description du protocole <i>Up ! Network</i> .
Selection.	Description d'une sélection pour l'empaquetage des paramètres.

Tableau 6 – Glossaire du modèle physique des données publiques du module *Up ! Network*

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

7 Composants techniques

Le module **Up ! Network** pour le projet **Up ! Application System** est constitué des composants suivants :

Fichiers du module.	
<ul style="list-style-type: none"> Fichier upsnet.e – Définition des interfaces des bibliothèques gérées par Up ! Network. Fichier upsnet.h – En-tête privé de Up ! Network. Fichier upsnet.def – Exportation des symboles pour Windows. 	
Composants.	upsnet0.
Description.	
Interface entre Up ! Network et Up ! Module .	
Fichiers.	
<ul style="list-style-type: none"> Fichier upsnet0.cpp – Fichier source. Fichier upsnet0.h – En-tête privé de upsnet0.cpp. Fichier upsnet0.e – En-tête protégé de upsnet0.cpp. 	
Composants.	upsnet1.
Description.	
<ul style="list-style-type: none"> Adaptateur client d'Up ! Network. Encapsulation des messages de service de la sorte à les exposer sous forme d'Application Program Interfaces (API) écrite en C--. 	
Fichiers.	
<ul style="list-style-type: none"> Fichier upsnet1.cpp – Fichier source. Fichier upsnet1.h – En-tête privé de upsnet1.cpp. Fichier upsnet1.e – En-tête protégé de upsnet1.cpp. 	
Composants.	upsnet2.
Description.	
<ul style="list-style-type: none"> Adaptateur serveur d'Up ! Network. Fonction principale d'une tâche Serveur. Fonction principale de la tâche Scripteur. Fonction principale de la tâche Statistiques. 	
Fichiers.	
<ul style="list-style-type: none"> Fichier upsnet2.cpp – Fichier source. Fichier upsnet2.h – En-tête privé de upsnet2.cpp. Fichier upsnet2.e – En-tête protégé de upsnet2.cpp. 	
Composants.	upsnet3.
Description.	
<ul style="list-style-type: none"> Services élémentaires d'Up ! Network applicables à tous les pilotes. Gestion des définitions upsnet.upi. 	
Fichiers.	

	Spécification technique du module	Date rédaction : 7 avril 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsNet-000003-A Spécification technique du module UpsNet.doc		

<ul style="list-style-type: none"> Fichier upsnet3.cpp – Fichier source. Fichier upsnet3.h – En-tête privé de upsnet3.cpp. Fichier upsnet3.e – En-tête protégé de upsnet3.cpp. 	
Composants.	upsnet4.
Description.	
Méthodes de l'interface Canal .	
Fichiers.	
<ul style="list-style-type: none"> Fichier upsnet4.cpp – Fichier source. Fichier upsnet4.h – En-tête privé de upsnet4.cpp. Fichier upsnet4.e – En-tête protégé de upsnet4.cpp. 	
Composants.	upsnet5.
Description.	
Méthodes du type Serveur .	
Fichiers.	
<ul style="list-style-type: none"> Fichier upsnet5.cpp – Fichier source. Fichier upsnet5.h – En-tête privé de upsnet5.cpp. Fichier upsnet5.e – En-tête protégé de upsnet5.cpp. 	
Composants.	upsnet99.
Description.	
Interface entre Up ! Network et Up ! Kernel .	
Fichiers.	
<ul style="list-style-type: none"> Fichier upsnet99.cpp – Fichier source. Fichier upsnet99.h – En-tête privé de upsnet99.cpp. Fichier upsnet99.e – En-tête protégé de upsnet99.cpp. 	

Tableau 7 – Composants techniques du module

Fin de document