

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.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.

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.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	Codifications.....	7
5.1.1	Action.....	7
5.1.2	Etats des actions.....	7
5.1.3	Types de répertoires.....	8
5.1.4	Suffixe des adaptateurs.....	8
5.2	Extension du compilateur.....	8
5.2.1	Extension d'un fichier source.....	9
5.2.2	Générateur.....	9
5.2.3	Extension du compilateur.....	10
5.3	Services de compilation.....	11
5.3.1	Services génériques.....	11
5.3.2	Services de compilation.....	13
5.3.3	Services de génération.....	13
5.3.4	Procédures et fonctions de rappel.....	16
6	Modèle de données.....	17
7	Composants techniques.....	18

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

1.2 Table des illustrations

Tableau 1 – Actions possibles sur les fichiers source	7
Tableau 2 – Etat des actions possibles sur les fichiers source	8
Tableau 3 – Types de répertoires.....	8
Tableau 4 – Types de répertoires.....	8
Diagramme 5 – Modèle physique des données publiques du module <i>Up ! Compiler</i>	17
Tableau 6 – Glossaire du modèle physique des données publiques du module <i>Up ! Compiler</i>	17
Tableau 7 – Composants techniques du module	18

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.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.

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-UpsCp1-000002	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-UpsKrn-000003	Plan d'écriture d'un module.

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

3 Introduction

3.1 Objet du document

L'objet de ce document est de décrire le contenu technique du module logiciel **Up ! Compiler** 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 : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

4 Interaction avec l'environnement

4.1 Description

L'objet du module logiciel *Up ! Compiler* est de piloter la production automatique de logiciels :

- **Extraction.**
Il y a :
 - Les *Interface Repositories* de *Common Object Request Broker Architecture (CORBA)*.
 - Les *Type Libraries* de *Component Object Module (COM)*.
- **Conversion.**
Il y a :
 - Les *Interface Definition Language (IDL)* de *Common Object Request Broker Architecture (CORBA)*.
 - Les *Interface Definition Language (IDL)* de *Component Object Module (COM)*.
 - Les classes et paquets de *Java*.
- **Génération.**
Il y a :
 - Les fichiers en *Up ! 5GL* des modules construits avec *Up ! Workbench*.
 - Les fichiers en *C--* des modules écrits en *Up ! 5GL*.
- **Assemblage.**
Les fichiers en *C++* pour une plate-forme technique cible.

4.2 Paramètres

Les paramètres *Up ! Compiler* sont documentés sur le site *Internet d'Up ! Application System* [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 : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

5 Choix techniques

5.1 Codifications

5.1.1 Action

Voici les actions possibles sur les fichiers source du tableau noir :

Valeur.	Sémantique.
AS_AnalyserClasseJava.	Analyse le fichier source de la classe ou du paquet Java .
AS_AnalyserIdModule.	Analyse de l' Interface Definition Language (IDL) d'un module.
AS_AnalyserInterfaceComposant.	Analyse l'interface d'un composant.
AS_AnalyserInterfaceComposantPourDependance.	Analyse l'interface des composants du module pour calculer les dépendances.
AS_AnalyserInterfaceDesComposants.	Analyse l'interface des composants du module.
AS_AnalyserInterfaceDesComposantsPourDependance.	Analyse l'interface des composants pour calculer les dépendances.
AS_AnalyserInterfaceModule.	Analyse l'interface du module.
AS_AnalyserSourceComposant.	Analyse du fichier source du composant.
AS_AnalyserSourceComposantPourInterface.	Analyse le fichier source du composant pour générer l'interface.
AS_AnalyserSourceModule.	Analyse du fichier source d'un module.
AS_ExtraireIdModule.	Extraction de l' Interface Definition Language (IDL) d'un module.
AS_GenererInterfaceComposant.	Génère les fichiers sources de l'interface d'un composant.
AS_GenererInterfaceDesComposants.	Génère l'interface des composants du module.
AS_GenererInterfaceModule.	Génère l'interface du module.
AS_GenererSourceComposant.	Génère les fichiers sources d'un composant.
AS_GenererSourceDesComposants.	Génère les fichiers sources des composants du module.
AS_GenererSourceModule.	Génère le source du module.
AS_PreprocesserClasseJava.	Pré-processing d'une classe Java .
AS_PreprocesserIdModule.	Pré-processing de l' Interface Definition Language (IDL) d'un module.

Tableau 1 – Actions possibles sur les fichiers source

5.1.2 Etats des actions

Voici les états des actions sur un fichier source :

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

Valeur.	Sémantique.
EA_AFaire.	Action à faire.
EA_Fait.	Action faite.
EA_NonProgramme.	Action non programmée.

Tableau 2 – Etat des actions possibles sur les fichiers source

5.1.3 Types de répertoires

Voici les types de répertoires permettant d'exploiter la configuration d'installation :

Valeur.	Sémantique.
TR_Aucun.	Pas de répertoire particulier.
TR_Generation.	Répertoire des fichiers générés.
TR_IdlCorba.	Répertoire des Interface Definition Language (IDL) de Common Object Request Broker Architecture (CORBA) .
TR_IdIDCom.	Répertoire des Interface Definition Language (IDL) de Component Object Module (COM) .
TR_Source.	Répertoire des fichiers sources.
TR_Temporaire.	Répertoire des fichiers temporaires.

Tableau 3 – Types de répertoires

5.1.4 Suffixe des adaptateurs

Voici les suffixes des adaptateurs :

Valeur.	Sémantique.
SA_Aucun.	Pas de suffixe particulier.
SA_ClientCorba.	Adaptateur client Corba .
SA_ClientDCom.	Adaptateur client Com .
SA_ClientJava.	Adaptateur client Java .
SA_ClientUpsNet.	Adaptateur client Up ! Network .
SA_ServeurUpsNet.	Adaptateur serveur Up ! Network .
SA_Dictionnaire.	Adaptateur dictionnaire.
SA_ServeurCorba.	Adaptateur serveur Corba..
SA_ServeurDCom.	Adaptateur serveur Com .
SA_ServeurJava.	Adaptateur serveur Java .

Tableau 4 – Types de répertoires

5.2 Extension du compilateur

Le compilateur peut être étendu via un ou plusieurs modules secondaires dont les caractéristiques sont les suivantes, dans la limite de **CO_NbGenerateurs1Max** extensions :

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

- L'interface de données doit être une extension de **Cp2XDonnees**. Ceci permet de spécifier à **Up! Compiler** la taille de l'espace mémoire privé nécessaire pour l'extension de chaque concept de base. Quand aucun espace mémoire n'est pas nécessaire, la taille à spécifier est **0**. Elle est inutile à ce jour.
- L'interface de traitements doit être une extension de **Cp2XTraitements**. Ceci permet de spécifier à **Up! Compiler** les fonctions de rappels nécessaires pour étendre les algorithmes de compilation. Quand une fonction de rappel n'est pas nécessaire, la fonction de rappel doit être **UpsKrn.AppellImpossibleApiNull**.

5.2.1 Extension d'un fichier source

Si l'extension d'**Up! C / C++ Generator** est en mesure de manipuler des fichiers source qui sont des extensions d'**Up! 5GL**, alors les **Application Program Interfaces (API)** suivantes doivent être renseignées :

- **UpsCp2X.CalculerModuleRattachement**.
Calcule le module de rattachement du fichier source.
- **UpsCp2X.ControlerAppartenanceSource**.
Contrôle si le paquet de définitions est bien rattaché au fichier source.
- **UpsCp2X.EffectuerActionSurSource**.
Effectue une action sur un fichier source.
- **UpsCp2X.EffectuerPremiereAction**.
Effectue la première action sur le fichier source i.e. l'action par défaut.
- **UpsCp2X.ExecuterAction**.
Exécute l'action demandée sur le fichier source.
- **UpsCp2X.IIExistePrincipale**.
Retourne **Vrai** si le fichier source comporte la fonction principale.
- **UpsCp2X.LireInterface**.
Lit l'interface d'un fichier source.
- **UpsCp2X.LireListeModulesImportes**.
Lecture de la liste des modules importés.
- **UpsCp2X.ReallouerDependances**.
Réalloue les dépendances d'un fichier source.
- **UpsCp2X.RechercherFichier**.
Recherche du fichier cible en tenant compte des répertoires et des cibles de génération.

5.2.2 Générateur

Si l'extension d'**Up! Compiler** est en mesure de mettre en oeuvre les primitives de génération, alors les **Application Program Interfaces (API)** suivantes doivent être renseignées :

- **UpsCp2X.CalculerFichierObjet**.
Calcule le nom du fichier objet résultant de la génération du fichier source.
- **UpsCp2X.ChercherCible**.
Recherche la cible de génération identifiée par son mot-clé.
- **UpsCp2X.DebuterBatchCompilation**.
Débute le fichier batch de compilation.

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

- ***UpsCp2X.ExecuterBatchCompilation.***
Exécute le fichier batch de compilation.
- ***UpsCp2X.Generer.***
Effectue la génération du fichier source.
- ***UpsCp2X.Preprocesser.***
Effectue le pré-processing du fichier source.
- ***UpsCp2X.TerminerBatchCompilation.***
Termine le fichier batch de compilation.

5.2.3 Extension du compilateur

Si l'extension d'**Up ! Compiler** est un compilateur prenant en charge une technologie tierce, alors les **Application Program Interfaces (API)** suivantes doivent être renseignées :

- ***UpsCp2X.AnalyserParametres.***
Analyse les paramètres de compilation.
- ***UpsCp2X.AnalyserDependances.***
Recherche les modules importés d'un fichier source pour analyser ses dépendances.
- ***UpsCp2X.AnalyserDependances2.***
Programme l'analyse des dépendances d'un fichier source.
- ***UpsCp2X.AnalyserParametres2.***
Analyse les paramètres spécifiques au fichier source pour le compilateur cible.
- ***UpsCp2X.AnalyserSource.***
Programme l'analyse du fichier source.
- ***UpsCp2X.AnalyserSource2.***
Au cours de l'analyse du fichier source, exécute l'action demandée.
- ***UpsCp2X.AnalyserSourceComposantOuModule.***
Analyse le fichier source du composant ou du module.
- ***UpsCp2X.AnalyserSourceComposantOuModule2.***
Au cours de l'analyse le fichier source du composant ou du module, vérifie que toutes les dépendances sont disponibles.
- ***UpsCp2X.AnalyserSourceComposantOuModule3.***
Suite à l'analyse le fichier source du composant ou du module, efface le fichier issu du pré-processing.
- ***UpsCp2X.CalculerNumeroSemantique2.***
Calcule le numéro de l'analyseur sémantique.
- ***UpsCp2X.CalculerProfondeur.***
Calcule la priorité du fichier source d'après la priorité des modules importés. Usuellement, il y a un appel récursif à ***UpsCp1.CalculerProfondeur.***
- ***UpsCp2X.DebuterAnalyseSource.***
Affecte les méthodes de rappel pour l'analyseur d'un fichier source écrit dans le langage d'une technologie tierce.
- ***UpsCp2X.DebuterAnalyseSource2.***
Débute une succession d'analyse.

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

- ***UpsCp2X.DebuterAnalyseSource3.***
Importe les modules standards correspondant à la technologie tierce au cours du début de l'analyse.
- ***UpsCp2X.DebuterLireCaractereFichier.***
Débute la lecture d'un fichier source en mode caractère.
- ***UpsCp2X.EmpilerSource.***
Déclare un nouveau fichier source sur le tableau noir et met le en attente. Version simplifiée.
- ***UpsCp2X.EmpilerSource2.***
Déclare un nouveau fichier source sur le tableau noir et met le en attente. Version étendue.
- ***UpsCp2X.EstUneClasseJava.***
Retourne **Vrai** si le fichier source correspond à une classe ou un paquet **Java**.
- ***UpsCp2X.EstUneldl.***
Retourne **Vrai** si le fichier source correspond à une **Interface Definition Language (IDL)**.
- ***UpsCp2X.ImporterModulesStandards.***
Importe les modules standards en **Up ! 5GL** correspondant à la technologie tierce.
- ***UpsCp2X.ImporterModulesStandards2.***
Importe les définitions du module importé en les déclarant au niveau de l'analyseur lexical.
- ***UpsCp2X.InterfaceModuleDisponible.***
Retourne l'interface du module si elle est disponible.
- ***UpsCp2X.InvalidierUsageInterface.***
Recherche le fichier source correspondant à l'interface dont il faut invalider le statut.
- ***UpsCp2X.LireInterfaceModuleImporte.***
Lit l'interface du module importé.
- ***UpsCp2X.MarquerAnalyserInterfaceModule.***
Marque qu'il faut analyser l'interface du module.
- ***UpsCp2X.PreprocesserInterfaceModuleOuClasse.***
Calcule le fichier cible du pré-processing.
- ***UpsCp2X.SetNumeroLexical.***
Fixe le numéro de l'analyseur lexical pour l'analyseur sémantique.
- ***UpsCp2X.SetNumeroSyntaxique.***
Fixe le numéro de l'analyseur syntaxique pour l'analyseur sémantique.
- ***UpsCp2X.TerminerAnalyseSource.***
Suite à l'analyse du fichier source, libère les ressources du fichier source.
- ***UpsCp2X.VerifierGenerer.***
Vérifie que la génération est possible.

5.3 Services de compilation

5.3.1 Services génériques

Les services de génériques, qui peuvent être utilisés par une extension d'**Up ! Compiler**, sont les suivants :

- ***UpsCp1.AdaptateurClientUpsNet.***
Si **Vrai**, il faut générer l'adaptateur client pour **Up ! Network**.

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

- **UpsCp1.AdaptateurDictionnaire.**
Si **Vrai**, il faut générer la bibliothèque dictionnaire de données et de traitements.
- **UpsCp1.AdaptateurServeurUpsNet.**
Si **Vrai**, il faut générer l'adaptateur serveur pour **Up ! Network**.
- **UpsCp1.AllouerSource.**
Alloue un nouveau fichier source.
- **UpsCp1.ComparerFichierInterface.**
Retourne **Vrai** si deux fichiers interfaces sont identiques à l'en-tête près.
- **UpsCp1.Contrainte.**
Si **Vrai**, il faut générer le code de prise en compte des contraintes.
- **UpsCp1.Desindenter.**
Supprime une indentation de l'écriture dans le fichier source généré.
- **UpsCp1.Detail.**
Si **Vrai**, il faut afficher le détail des opérations.
- **UpsCp1.Indenter.**
Indente l'écriture dans le fichier source généré.
- **UpsCp1.FichierLog.**
Fichier journal de la compilation.
- **UpsCp1.FichierModifie.**
Retourne **Vrai** si le fichier source a été modifié après le fichier objet correspondant.
- **UpsCp1.LireExtensionSource.**
Lit l'extension d'un fichier source **Up ! 5GL** spécifique à **Up ! Compiler**.
- **UpsCp1.LireNumeroSemantiqueUpsGr1.**
Numéro de l'analyseur sémantique **Up ! Grammar**.
- **UpsCp1.LireProfondeur.**
Lit la profondeur d'imbrication d'un fichier source.
- **UpsCp1.Menage.**
Si **Vrai**, il faut supprimer les fichiers temporaires.
- **UpsCp1.MiseAuPoint.**
Si **Vrai**, il faut générer le code de mise au point.
- **UpsCp1.ModeLancement.**
Mode de lancement demandé pour le programme exécutable à générer.
- **UpsCp1.Persistant.**
Si **Vrai**, il faut générer le code de persistance.
- **UpsCp1.PrefixeModule.**
Calcule le préfixe d module.
- **UpsCp1.RechercherFichierDansRepertoires.**
Recherche un fichier dans une liste de répertoires.
- **UpsCp1.RepertoireClasses.**
Lit le répertoire des classes **Java** générées.
- **UpsCp1.RepertoireExecutables.**
Lit le répertoire des fichiers exécutables.

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

- ***UpsCp1.RepertoireGeneres.***
Lit le répertoire des fichiers générés.
- ***UpsCp1.RepertoireIdls.***
Lit le répertoire des ***Interface Definition Language (IDL)*** générée.
- ***UpsCp1.RepertoireInterfaces.***
Lit le répertoire des fichiers interfaces.
- ***UpsCp1.RepertoireMessages.***
Lit le répertoire des fichiers messages.
- ***UpsCp1.RepertoireObjets.***
Lit le répertoire des fichiers objets.
- ***UpsCp1.RepertoireSources.***
Lit le répertoire des fichiers sources.
- ***UpsCp1.RepertoireTemporaires.***
Lit le répertoire des fichiers temporaires.

5.3.2 Services de compilation

Les services de compilation, qui peuvent être utilisés par une extension d'***Up ! Compiler***, sont les suivants :

- ***UpsCp1.AnalyserSource.***
Analyse un fichier source.
- ***UpsCp1.ChercherSource.***
Cherche un fichier source sur le tableau noir et crée-le s'il n'existe pas.
- ***UpsCp1.DebuterAnalyseSource.***
Débute l'analyse d'un fichier source.
- ***UpsCp1.LireNomFichierSource.***
Lit le nom du fichier source courant.
- ***UpsCp1.RechercherAction.***
Retourne ***Vrai*** si une action a déjà été programmée pour un fichier source.
- ***UpsCp1.RechercherFichier.***
Recherche un fichier dans une liste de répertoires donnée et pour une extension donnée.
- ***UpsCp1.TerminerAnalyseSource.***
Termine l'analyse d'un fichier source.

5.3.3 Services de génération

Les services de génération, qui peuvent être utilisés par une extension d'***Up ! Compiler***, sont les suivants :

- ***UpsCp1.CalculerBornesTableau.***
Calcule les bornes d'un tableau.
- ***UpsCp1.CalculerCheminHeritage.***
Calcule le chemin menant à une propriété ou une méthode dans un héritage.
- ***UpsCp1.CalculerFichierObjet.***
Calcule le nom du fichier objet cible de la génération du fichier source.

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

- ***UpsCp1.CalculerPrefixeDUnModule.***
Calcul le préfixe d'un module.
- ***UpsCp1.CalculerPrefixeModule.***
Calcule le préfixe du module.
- ***UpsCp1.CalculerPrefixePaquet.***
Calcule le préfixe du paquet.
- ***UpsCp1.CalculerProfondeur.***
Calcule la profondeur du fichier source en fonction de la profondeur de ses fichiers modules importés.
- ***UpsCp1.CalculerTypeMemoireSegmentPrive.***
Calcule le type de mémoire pour le segment privé.
- ***UpsCp1.CalculerTypeMemoireSegmentProtege.***
Calcule le type de mémoire pour le segment protégé.
- ***UpsCp1.CalculerTypeMemoireSegmentPublic.***
Calcule le type de mémoire pour le segment public.
- ***UpsCp1.ChangerExtension.***
Change l'extension d'un fichier.
- ***UpsCp1.ChargerParametresEtOptions.***
Charge les paramètres et les options d'**Up ! Compiler**.
- ***UpsCp1.ControlerAppartenanceModule.***
Contrôle qu'un paquet de définitions est bien rattaché au module de rattachement du fichier source.
- ***UpsCp1.ControlerAppartenanceSource.***
Contrôle qu'un paquet de définitions est bien rattaché au fichier source.
- ***UpsCp1.CreerRepertoire.***
Crée un répertoire en tenant compte de la configuration d'installation.
- ***UpsCp1.DebuterBatchCompilation.***
Débute le fichier batch de compilation.
- ***UpsCp1.Ecrire.***
Ecrit une chaîne de caractères dans le fichier de génération.
- ***UpsCp1.EcrireCommentaire.***
Ecrit une liste de commentaires dans le fichier de génération.
- ***UpsCp1.EcrireEnteteStandard.***
Ecrit l'en-tête de génération dans le fichier de génération.
- ***UpsCp1.EcrireEntier.***
Ecrit une constante entière dans le fichier de génération.
- ***UpsCp1.EcrireReel.***
Ecrit une constante réelle dans le fichier de génération.
- ***UpsCp1.EcrireReelSansSaut.***
Ecrit une constante réelle dans le fichier de génération sans réaliser de sauts de lignes automatiquement.

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

- ***UpsCp1.EcrireSansSaut.***
Ecrit une chaîne de caractères dans le fichier de génération sans réaliser de sauts de lignes automatiquement.
- ***UpsCp1.EcrireSansSautEtBackSlash.***
Ecrit une chaîne de caractères dans le fichier de génération sans réaliser de sauts de lignes automatiquement ni en doublant les caractères **division inversée **.
- ***UpsCp1.EcrireUnCommentaire.***
Ecrit un commentaire unitaire dans le fichier de génération.
- ***UpsCp1.EnumererLibellesOptions.***
Enumère les libellés des options d'**Up ! Compiler**.
- ***UpsCp1.EnumererCheminHeritage.***
Enumère étape par étape le chemin d'un héritage.
- ***UpsCp1.ExecuterBatchCompilation.***
Exécute le fichier batch de compilation.
- ***UpsCp1.ExtraireNomFichier.***
Extrait le nom du fichier du chemin complet.
- ***UpsCp1.FermerFichier.***
Ferme un fichier précédemment ouvert.
- ***UpsCp1.Generer.***
Effectue la génération à partir d'un fichier source.
- ***UpsCp1.IIExisteAppelGlobal.***
Retourne **Vrai** si le fichier source comporte un appel global.
- ***UpsCp1.IIExisteDonneesPrivees.***
Retourne **Vrai** si le fichier source comporte au moins une donnée privée.
- ***UpsCp1.IIExisteDonneesProtegees.***
Retourne **Vrai** si le fichier source comporte au moins une donnée protégée.
- ***UpsCp1.IIExistePrincipal.***
Retourne **Vrai** si le fichier source comporte la fonction principale.
- ***UpsCp1.IIExisteVariableGlobale.***
Retourne **Vrai** si le fichier source comporte une variable globale.
- ***UpsCp1.IIExisteVariableObjet.***
Retourne **Vrai** si le fichier source comporte une variable objet.
- ***UpsCp1.OuvrirFichier.***
Ouvre un fichier en tenant compte de la configuration d'installation.
- ***UpsCp1.PasCroissant.***
Retourne **Vrai** si le pas d'une boucle est croissant.
- ***UpsCp1.Preprocesser.***
Effectue le pré-processing d'un fichier source.
- ***UpsCp1.Recompiler.***
Retourne **Vrai** s'il faut forcer la compilation.
- ***UpsCp1.SauterLigne.***
Saute une ligne dans le fichier de génération.

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

- ***UpsCp1.Souligner.***
Ecrit un commentaire de soulignement dans le fichier de génération.
- ***UpsCp1.TerminerBatchCompilation.***
Termine le fichier batch de compilation.
- ***UpsCp1.VerifierValiditeCommentaire.***
Retourne **Vrai** si la forme d'un commentaire est valide.

5.3.4 Procédures et fonctions de rappel

Les procédures et fonctions de rappel, à utiliser pour les appels récurrents des procédures et des fonctions de rappels d'une extension d'**Up ! Compiler**, sont les suivants :

- ***UpsCp1.AnalyserDependances.***
Analyse les dépendances entre les fichiers source.
- ***UpsCp1.AnalyserSourceComposantOuModule.***
Analyse les dépendances du fichier source.
- ***UpsCp1.CalculerDependance.***
Calcule les dépendances du fichier source courant.
- ***UpsCp1.CalculerNumeroSemantique2.***
Calcule le numéro de l'analyseur sémantique.
- ***UpsCp1.ChercherUnModuleAImporter.***
Enumère le module à importer suivant.
- ***UpsCp1.EffacerInterfaceSource.***
Efface l'interface du fichier source.
- ***UpsCp1.EffectuerActionSurSource2.***
Effectue l'action demandée sur le fichier source ou tout au moins programme-la. L'analyseur sémantique est connu.
- ***UpsCp1.EmpilerAction.***
Empile une action sur le fichier source sans réaliser de contrôles particuliers.
- ***UpsCp1.GenererInterface.***
Génère l'interface du composant ou du fichier source.
- ***UpsCp1.GenererInterfaceDesComposants.***
Génère l'interface des composants du fichier source du module.
- ***UpsCp1.ImporterModulesStandards.***
Importe les modules standards d'**Up ! 5GL**.
- ***UpsCp1.MarquerAnalyserInterfaceModule.***
Marque qu'il faut analyser l'interface du module.
- ***UpsCp1.PreprocesserInterfaceModuleOuClasse.***
Réalise le pré-processing d'une interface d'un module ou d'une classe.
- ***UpsCp1.SourceModifie.***
Retourne **Vrai** si le fichier source a été modifié.
- ***UpsCp1.SupprimerExtension.***
Supprime l'extension du nom d'un fichier source.
- ***UpsCp1.VerifierGenerer.***
Vérifie que la génération est possible.

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

6 Modèle de données

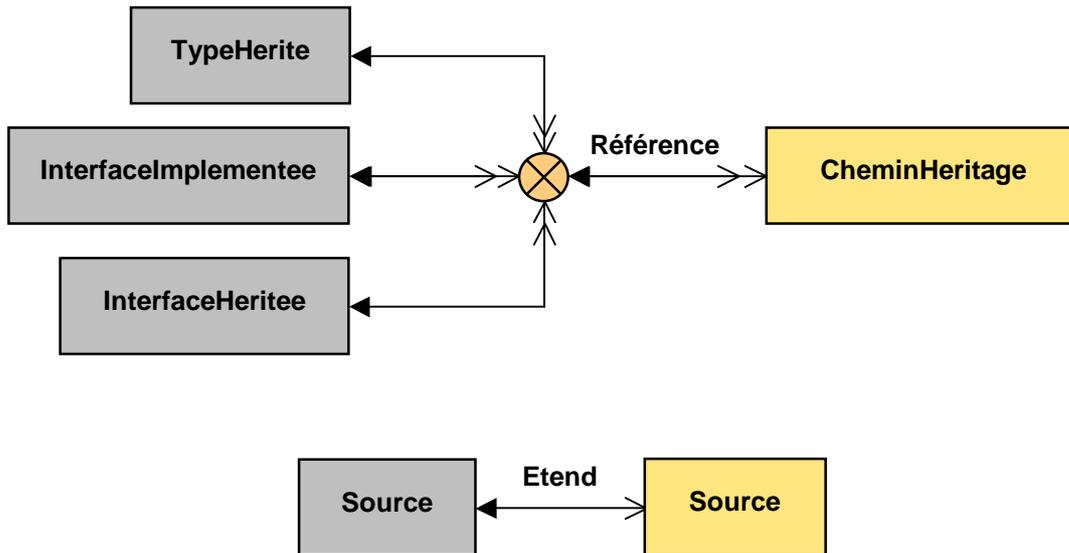


Diagramme 5 – Modèle physique des données publiques du module *Up ! Compiler*

Toutes les entités suivantes sont décrites dans le fichier **upscp1.e** :

Entité.	Description.
CheminHeritage.	Etape d'un chemin d'héritages pour accéder à une propriété ou à une méthode.
InterfaceHeritee.	Interface héritée par une autre interface.
InterfacImplementee.	Interface implémentée par un autre type.
TypeHerite.	Type hérité par un autre type.
Source.	Fichier source.

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

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

7 Composants techniques

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

Fichiers du module.	
<ul style="list-style-type: none"> Fichier uspcp1.e – Définition de <i>Up ! Compiler</i>. Fichier uspcp1.h – En-tête privé de <i>Up ! Compiler</i>. Fichier uspcp1.def – Exportation des symboles pour <i>Windows</i>. 	
Composants.	uspcp1_0.
Description.	
Interface entre <i>Up ! Compiler</i> et <i>Up ! Module</i> .	
Fichiers.	
<ul style="list-style-type: none"> Fichier uspcp1_0.cpp – Fichier source. Fichier uspcp1_0.h – En-tête privé de uspcp1_0.cpp. Fichier uspcp1_0.e – En-tête protégé de uspcp1_0.cpp. 	
Composants.	uspcp1_1.
Description.	
Utilitaires pour le pilotage de compilation.	
Fichiers.	
<ul style="list-style-type: none"> Fichier uspcp1_1.cpp – Fichier source. Fichier uspcp1_1.h – En-tête privé de uspcp1_1.cpp. Fichier uspcp1_1.e – En-tête protégé de uspcp1_1.cpp. 	
Composants.	uspcp1_2.
Description.	
Tableau noir de pilotage de la compilation.	
Fichiers.	
<ul style="list-style-type: none"> Fichier uspcp1_2.cpp – Fichier source. Fichier uspcp1_2.h – En-tête privé de uspcp1_2.cpp. Fichier uspcp1_2.e – En-tête protégé de uspcp1_2.cpp. 	
Composants.	uspcp1_3.
Description.	
Utilitaires pour la génération.	
Fichiers.	
<ul style="list-style-type: none"> Fichier uspcp1_3.cpp – Fichier source. Fichier uspcp1_3.h – En-tête privé de uspcp1_3.cpp. Fichier uspcp1_3.e – En-tête protégé de uspcp1_3.cpp. 	

Tableau 7 – Composants techniques du module

	Spécification technique du module	Date rédaction : 3 février 2004.
	Diffusion restreinte	Date validation :
Référence : UpComp-UpsCp1-000003-A Spécification technique du module UpsCp1.doc		

Fin de document