

# **Rapport final**

## **Prototype d'intégration sur UniVerse**

4/18/2008

Éric Barrette et Alexandre Voyer

## Table des matières

1. Rappel du projet .....	5
1.2 Description du projet .....	5
1.3 Objectif du projet .....	5
1.4 Technologies utilisées .....	6
1.4.1 Serveur PR1ME.....	6
1.4.2 IBM UniVerse .....	6
1.4.3 BASIC .....	7
1.4.4 Base de données Information.....	7
1.4.5 Windows Server 2003.....	7
1.4.6 Microsoft C#.....	7
1.4.7 UniVerse ODBC Driver .....	7
1.4.8 Outils utilisés.....	7
1.5 Appréciation générale.....	8
2. Concepts de base UniVerse.....	9
2.1 Fichiers .....	9
2.2 Dictionnaires.....	9
2.2.1 Types de champs.....	9
2.3 Vocabulaire.....	10
2.4 Catalogue.....	11
3. Estimation de temps pour faire la conversion .....	12
3.1 Des données .....	12
3.2 Des écrans .....	12
4. Recommandations.....	13
Bibliographie .....	14
Annexes.....	15
Échéancier révisé.....	15
Document sur la configuration du serveur UniVerse .....	15

Document de migration des dictionnaires et des données .....	15
Document de création d'une connexion ODBC.....	16
Annexe « A » .....	17
Semaine du 6 janvier 2008 .....	17
Semaine du 13 janvier 2008 .....	17
Mercredi 16 janvier .....	17
Semaine du 20 janvier 2008 .....	17
Mercredi 23 janvier .....	17
Jeudi 24 janvier .....	17
Semaine du 27 janvier 2008 .....	18
Mercredi 30 janvier .....	18
Semaine du 3 février 2008 .....	18
Mardi 5 février .....	18
Mercredi 6 février .....	18
Semaine du 10 février 2008 .....	19
Mercredi 13 février .....	19
Semaine du 17 février 2008 .....	19
Mercredi 20 février .....	19
Semaine du 24 février 2008 .....	19
Mercredi 27 février .....	19
Semaine du 2 mars 2008.....	20
Mercredi 5 mars.....	20
Semaine du 9 mars 2008.....	20
Mercredi 12 mars.....	20
Semaine du 16 mars 2008.....	20
Mardi 18 mars.....	20
Mercredi 19 mars.....	21
Semaine du 23 mars 2008.....	21
Lundi 24 mars .....	21
Mardi 25 mars.....	21
Mercredi 26 mars.....	21

Semaine du 30 mars 2008 .....	22
Mercredi 2 avril.....	22
Jeudi 3 avril.....	22
Semaine du 6 avril 2008.....	22
Semaine du 18 avril 2008.....	22
Annexe « A » .....	23
Téléchargement de l'application .....	23
Installation sur le serveur.....	23
Vérification de l'installation .....	25
Services .....	25
Test de connexion.....	25
Erreurs communes.....	26
Configuration sur serveur UniVerse .....	27
Créer un compte.....	27
Annexe « C ».....	28
Procédure pour la migration des dictionnaires.....	28
Procédure pour la migration des données.....	29
Annexe « D » .....	30
Installation du Pilote ODBC .....	30
Configuration de l'environnement UniVerse .....	30
Configuration de l'application ODBC Admin .....	31
Faire la connexion dans Visual Studio.....	31

## 1. Rappel du projet

### 1.2 Description du projet

Le projet, qui réunissait l'Université du Québec en Outaouais (UQO) et la Société de transports de l'Outaouais (STO), visait à examiner la possibilité de migrer un serveur PR1ME vers un environnement IBM UniVerse. Cela se présentait comme étant la suite logique car, historiquement, une compagnie du nom de VMark avait acheté la technologie PR1ME pour ensuite la renommer à UniVerse. Quelques années après, IBM a acquis Vmark. C'était donc l'environnement idéal pour procéder à une migration.

Ce prototype est donc une étape cruciale, puisque le serveur PR1ME qui appartient à la STO contient de l'information sensible sur la paie et les employés de l'organisation. Il serait fâcheux si le serveur arrêtait de fonctionner puisqu'il n'existe peu ou pas de support pour ce qui est des pièces physiques appartenant au serveur.

### 1.3 Objectif du projet

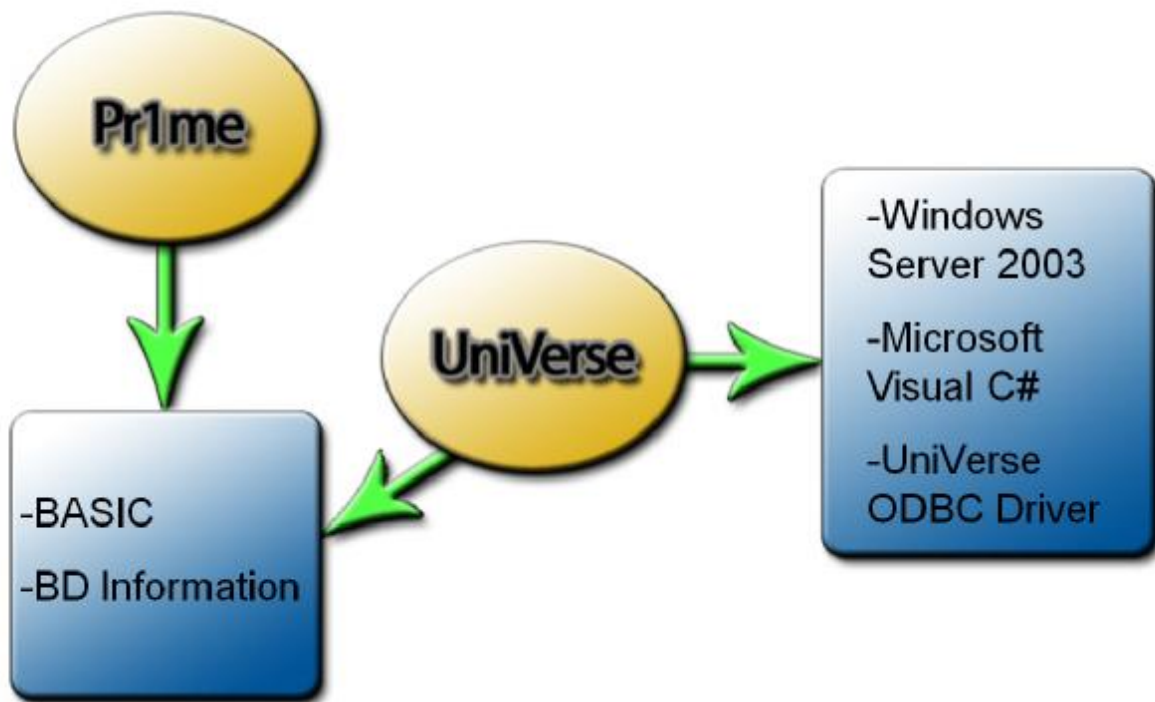
Notre projet consistait donc, en premier lieu, à prouver qu'il était faisable de transférer sans problème l'information du serveur PR1ME à un environnement UniVerse.

En second lieu, il fallait implémenter des outils qui permettraient d'automatiser le processus de transferts de fichiers pour permettre une migration rapide et efficace des bases de données.

En troisième lieu, il fallait trouver des alternatives d'écrans, car les écrans qui résident sur le serveur PR1ME à la STO ne sont pas transférables dans l'architecture IBM UniVerse puisque l'application S1MPLE qui gère actuellement ces écrans est compilée et nous n'avons pas accès au code source; il est donc impossible de le compiler sur UniVerse.

En quatrième lieu, il fallait concevoir une estimation du temps pour effectuer la conversion des données et des écrans, si possible.

## 1.4 Technologies utilisées



Voici une courte liste des technologies qui ont été utilisées au cours du projet et une courte description de chacune d'elles.

### 1.4.1 Serveur PR1ME

Technologie utilisée et supportée de 1972 à 1992. C'était aussi un de premiers serveurs à offrir la possibilité de compiler et d'exécuter plusieurs types de codes sous un même serveur. La STO utilise ce serveur pour exécuter des applications en Fortran et en Basic. Ce serveur appartient à la STO et nous avons eu accès à des écrans et à une partie des données entreposées sur le serveur pour concevoir notre prototype.

### 1.4.2 IBM UniVerse

Environnement qui roule sous un serveur Windows 2003 et qui est à la base de notre prototype. C'est dans cet environnement que nous avons conçu le prototype.

### **1.4.3 BASIC**

Langage de programmation utilisé pour coder les applications et les sous-routines existantes reliées au module de la paie que la STO aura à transférer sur un nouveau serveur. Nos applications sont aussi programmées dans ce langage.

### **1.4.4 Base de données Information**

Base de données utilisée dans le serveur PR1ME et le serveur IBM UniVerse. Cette base de données supporte les champs multi valeurs, ce qui est essentiel à notre projet.

### **1.4.5 Windows Server 2003**

Système d'exploitation sur lequel notre prototype est conçu.

### **1.4.6 Microsoft C#**

Langage de programmation qui pourrait être utilisé lors de la conception d'écrans Windows. C'est avec celui-ci que nous avons fait des tests sommaires.

### **1.4.7 UniVerse ODBC Driver**

Pilote pour établir une connexion entre Microsoft C# et l'environnement UniVerse.

### **1.4.8 Outils utilisés**

#### ***1.4.8.1 IBM UniVerse Serveur***

Environnement installé sur le serveur, préférablement Windows Server 2003 mais également supporté par des systèmes comme Windows XP. Cet environnement permet à des utilisateurs de s'y connecter et d'interagir avec le système.

#### *1.4.8.2 IBM Dynamic Connect*

Cet outil permet aux utilisateurs de se connecter sur l'environnement UniVerse. Il remplace Telnet qui est généralement déconseillé par IBM puisqu'il peut causer certains problèmes comme la session qui se termine abruptement.

#### *1.4.8.3 IBM ODBC Admins*

Cet outil permet à l'administrateur du système de créer des connexions ODBC, ce qui permettra par la suite à des programmes comme Visual Studio de se connecter à la base de données Information du serveur UniVerse.

### **1.5 Appréciation générale**

Tout en nous donnant du fil à retordre, il est important de noter que ce projet nous a fait sortir des sentiers battus et nous a donné la possibilité de pouvoir travailler avec des technologies qui sont en dehors des cours qui nous ont été enseignés à l'UQO. Ce fut une expérience des plus enrichissantes car nous savions que ce prototype serait une étape clé pour une entreprise bel et bien réelle de la région. Nous sommes reconnaissants envers l'UQO et la STO de nous avoir donné la chance de leur montrer que nous avons la détermination et la débrouillardise nécessaires pour en venir à conclure le projet qui nous était assigné.



## 2. Concepts de base UniVerse

### 2.1 Fichiers

Dans l'environnement UniVerse les fichiers et les dossiers sont considérés comme étant des fichiers. Un fichier a donc un « file type » qui permet de créer des dossiers (si le type de fichier est de 1 ou 19) ou de créer des fichiers (si le type est entre 2 et 18). Nous n'entrerons pas dans les détails, mais il est cependant important de savoir, à cette étape-ci, que la structure des tables de la base de données est aussi emmagasinée dans des fichiers.

Il est aussi bon à savoir que d'après les standards, tous les noms de fichiers ne devraient pas comporter d'espaces ni de lettres majuscules.

### 2.2 Dictionnaires

Les dictionnaires sont des fichiers qui agissent comme des tables. En fait, chaque dictionnaire est composé de deux fichiers. Le premier fichier étant celui qui décrit tout les champs et leur structure respective. Le deuxième fichier représente les données qui sont lié à cette table.

Chaque dictionnaire est composé de trois types de champs qui seront expliqués plus bas.

#### 2.2.1 Types de champs

##### 2.2.1.1 Données

Les champs de type données représentent des champs réels qui sont présents dans le fichier où les données sont stockées. Que ce soit des champs de type « Integer », « VarChar » ou « String », UniVerse n'en tient pas compte, il suffit de déclarer ces champs comme Data.

### *2.2.1.2 Imaginaires*

Les champs de type imaginaire sont des champs qui n'existent pas dans le fichier de données. Souvent ils sont composés de sous-routines ou de codes directement injectés dans les champs. Par exemple, si vous avez deux champs de type données « Nom » et « Prenom » qui stockent les noms et les prénoms des employés, il pourrait y avoir un champ imaginaire « NomPrenom » qui ferait la concaténation du nom et du prénom de ces employés. Cela peut être utile si vous affichez souvent la concaténation du nom et du prénom et peut sauver plusieurs lignes de codes redondantes.

### *2.2.1.3 Phrases*

Les champs de type phrases consistent principalement d'une énumération d'autres champs existants. Cela réduit la longueur d'une requête SQL et permet de réutiliser ces champs s'ils sont inscrits à plusieurs endroits dans l'application.

### *2.2.1.4 Association de champs (Multi valeurs)*

La notion multi valeurs, quoique très difficile à comprendre, est extrêmement utile. Elle vous fera fort probablement penser à une requête du type « JOIN » en SQL. Dans un dictionnaire, en UniVerse, il est possible que dans une même table, un ID soit associé à une séquence de champs prédéfinis dans la structure de la table.

## **2.3 Vocabulaire**

Toutes les commandes dans l'univers UniVerse sont entreposées dans un fichier. Ce fichier est appelé « VOC », ou communément appelé le vocabulaire. L'environnement UniVerse est flexible puisqu'il laisse à l'utilisateur l'option de modifier n'importe quel de ses commandes internes et d'en ajouter d'autres, au besoin.

## 2.4 Catalogue

Le catalogue dans UniVerse sert à entrer des raccourcis de commandes. Il est très utile dans le sens que les sous-routines peuvent utiliser ces raccourcis et n'ont jamais besoin de savoir où résident les applications en question si elles ont besoin de faire appel à ces fonctions.

## 3. Estimation de temps pour faire la conversion

### 3.1 Des données

À la suite du développement d'un parseur de dictionnaires et de données, nous en sommes venus à la conclusion que pour convertir l'ensemble des dictionnaires et leurs données qui lui sont associés qui nous ont été fournies, il faudrait moins d'une semaine (37,5 heures) de travail. Cela dit, il est à noter que chacune des sous-routines liées aux champs imaginaires devraient être par la suite évaluées et compilées dans l'environnement UniVerse. Ces sous-routines devraient aussi être soumises à une série de tests pour s'assurer que le code est fonctionnel.

### 3.2 Des écrans

Pour ce qui est des écrans, il faudrait possiblement plus d'une semaine pour développer un écran dit complexe. Pour ce qui est des écrans simples qui servent uniquement à afficher des données directement tirées du dictionnaire, il serait possible de faire ceux-ci en quelques heures à partir du programme ENTER.

Notre estimation se base principalement sur l'application ENTER écrite en BASIC qui est exécutée directement sur le serveur. Pour ce qui est des écrans programmés en C#, il nous est impossible de donner une estimation concrète car ceux-ci présentent plusieurs problèmes causés principalement par les champs imaginaires et lors de la sauvegarde ou de la mise à jour des champs.

## 4. Recommandations

Nous avons deux recommandations en ce qui concerne le développement futur du projet.

Premièrement, il serait réaliste de considérer de faire la migration complète des données qui sont entreposées sur l'architecture du serveur PR1ME vers une architecture IBM UniVerse pour s'assurer que les données sensibles ne puissent être perdues à tout moment. Nous sommes aussi convaincus que la transition des données peut se faire d'une façon simple et efficace. Ces données pourraient aussi, par la suite, être transformées vers une base de données MySQL/MSSQL/Oracle si le client décidait de refaire son système sous une autre architecture.

Deuxièmement, il serait intéressant d'étudier la possibilité de développer des services Web, ce qui pourrait donner une perspective intéressante au projet en cours et lui ajouter des possibilités dans le futur.

## Bibliographie

*Comprendre les données multi valeurs*, <http://en.wikipedia.org/wiki/MultiValue> .

*Life, the universe, and everything*, <http://www.mannyneira.com/universe/> .

*International IBM U2 User Group*, <http://www.u2ug.org/> .

*Writing your first IBM UniVerse application*,  
[http://212.241.202.162/cms/learnerpack/articles/first\\_uv\\_app.pdf](http://212.241.202.162/cms/learnerpack/articles/first_uv_app.pdf) .

## Annexes

Vous trouverez, annexés à ce rapport, un document portant sur l'échéancier complet du projet suivi de trois documents sur les procédures à suivre pour faire la migration d'un PR1ME à un serveur UniVerse.

### Échéancier révisé

Vous trouverez notre échéancier complet avec un sommaire du travail qui a été exécuté au courant de la session. Cela vous donnera une vision globale de l'envergure du projet et du temps que nous avons passé sur chaque étape clef du projet.

### Document sur la configuration du serveur UniVerse

Ce document contient de l'information très utile pour les personnes qui ont peu ou pas d'expérience avec UniVerse. Il explique comment configurer son serveur de A à Z et comment avoir un serveur qui est prêt en très peu de temps.

### Document de migration des dictionnaires et des données

Ce document traite de la migration des dictionnaires et des données liées à ceux-ci. Avec ce guide en main, il vous sera possible de migrer vos dictionnaires en toute aisance d'esprit avec nos parseurs de dictionnaires et de leurs données respectives.

## Document de création d'une connexion ODBC

Ce document aborde les connexions avec la base Information de UniVerse. Le guide, quoique sommaire, vous donnera une idée de la bonne façon d'interagir avec Information.



## **Annexe « A »**

### *Échéancier révisé*

#### **Semaine du 6 janvier 2008**

- Rencontre initiale pour le projet.
- Décision de prendre le projet de conversion d'un système PR1ME à UniVerse.

#### **Semaine du 13 janvier 2008**

##### **Mercredi 16 janvier**

- 10 h – 13 h: Première rencontre avec la STO.
- 14 h – 18 h: Questions et discussion dans le local de M. Iglewski.

#### **Semaine du 20 janvier 2008**

##### **Mercredi 23 janvier**

- 9 h – 11 h: Rencontre avec Mr. Iglewski.
- 11h – 17h: Travail sur UniVerse
  - Recherche de documentation
  - Tentatives d'installation

##### **Jeudi 24 janvier**

- 10h – 12h: Rencontre avec la STO.

## Semaine du 27 janvier 2008

### Mercredi 30 janvier

- 9h – 10h: Recherche préliminaire.
- 10h – 11h: Rencontre avec M. Iglewski et M. Gagnon.
- 11h – 12h: Installation dans le local, information sur les nouvelles machines/serveur.
- 12h – 18h: Travail, recherche d'autres manuels que ceux d'IBM qui sont trop approfondis.

## Semaine du 3 février 2008

### Mardi 5 février

- 9h – 17h: Élaboration du wiki, recherche sur les VOC/CATALOG/SUBROUTINES.

### Mercredi 6 février

- 10h – 11h : Rencontre avec M. Iglewski et M. Gagnon.
- 11h – 18h :
  - Entrer en contact avec Marc L'italien et établir une liste de questions techniques pour un rendez-vous futur.
  - Commencer à développer un parseur pour les dictionnaires.

## Semaine du 10 février 2008

### Mercredi 13 février

- 9h-10h: Pré-rencontre dans le local.
- 10h-11h: Rencontre avec M. Iglewski et M. Gagnon.
- 11h-12h: Rencontre avec Marc L'italien.
- 12h-19h:
  - Mettre un sommaire de la rencontre sur le Wiki.
  - Paramétrage serveur UniVerse sous Windows Server 2003.
  - Élaboration d'un parseur de « listing » de dictionnaire à dictionnaire UniVerse.
  - Recherche d'alternatives à S1MPLE.

## Semaine du 17 février 2008

### Mercredi 20 février

- 8h-18h:
  - Finition du parseur de « listing » de dictionnaire à un dictionnaire.
  - Élaboration d'un parseur qui prend les données Comma Separated Values (CSV) et les multi valeurs et les insérer dans le dictionnaire.
  - Recherche d'alternatives à SIMPLE.

## Semaine du 24 février 2008

### Mercredi 27 février

- 8h-17h:
  - Élaboration du document de progression.

- Travail sur le parseur qui prend les données CSV et les multi valeurs et les insérer dans le dictionnaire.

## Semaine du 2 mars 2008

### Mercredi 5 mars

- Travailler sur le parseur de dictionnaires.
- Entrer en communication avec un consultant d'IBM UniVerse au Royaume-Uni.
- Étudier son programme "ENTER" pour concevoir des écrans UniVerse.

## Semaine du 9 mars 2008

### Mercredi 12 mars

- Compléter le parseur de dictionnaires.
- Travailler sur le parseur de données.
- Étudier les possibilités de faire des interfaces en Visual Basic .NET au lieu d'ENTER/SIMPLE.
- Mettre à jour le Wiki avec nos programmes.

## Semaine du 16 mars 2008

### Mardi 18 mars

- Améliorer nos connaissances de l'environnement Visual Studio et de l'utilisation de la librairie disponible avec UniVerse.
- Mise à jour du wiki.

### Mercredi 19 mars

- Compléter une ou deux interfaces VB communiquant avec UniVerse.

### Semaine du 23 mars 2008

#### Lundi 24 mars

- Nous avons mis près de 12 heures à essayer de déchiffrer et de démêler toutes les différences entre UniData, UniDB2 et UniOLEDB. Nous avons par la suite choisi d'y aller avec UniOLEDB car UniDB2 ne semble pas avoir encore été mis à jour (Le "Fix Pack #1" devrait sortir sous peu, d'après IBM.)
- Préparation du document pour la présentation orale.

#### Mardi 25 mars

- Finition du fichier PowerPoint pour la présentation orale.

#### Mercredi 26 mars

- Développement d'un MDI et des deux interfaces en C# .NET 2005. Nous parvenons à 'afficher les écrans mais ne pouvons pas effectuer certains changements sur ceux-ci.

## **Semaine du 30 mars 2008**

### **Mercredi 2 avril**

- Préparation et pratique pour la présentation du 3 avril 2008. Installation de l'environnement sur le serveur de l'école et configuration dans le but de la présentation.

### **Jeudi 3 avril**

- Présentation du projet final au jury et aux superviseurs.

## **Semaine du 6 avril 2008**

- Préparation du rapport final du projet.
- Mise à jour du wiki.

## **Semaine du 18 avril 2008**

Compléter et corriger le rapport final pour ensuite le remettre au superviseur du projet.

## Annexe « A »

### *Configuration serveur UniVerse*

#### Téléchargement de l'application

Avant de débiter le processus d'installation, il faut premièrement procéder au téléchargement de l'environnement serveur d'essai UniVerse. Pour se faire, vous devez aller dans la section *Trials and Demos* de la page officielle de UniVerse ( <http://www-306.ibm.com/software/data/u2/universe/> ). Sur cette page vous allez devoir choisir le système d'exploitation sous lequel vous voulez installer le serveur, dans notre cas nous avons opté pour *U2 Personal Editions (UniVerse 10.2 Personal Edition including Clients)* qui est le choix idéal pour un environnement Windows 2003.

Finalement vous devez sélectionner *Windows 2000, Windows Server 2003, Windows XP: IBM UniVerse Personal Edition* qui pèse une vingtaine de mégaoctets. L'autre fichier d'environ 300 mégaoctets consiste de l'application client.

#### Installation sur le serveur

Il faut tout d'abord extraire le fichier téléchargé préalablement dans un dossier temporaire. Ce dossier devrait contenir un fichier intitulé *INSTALL.EXE* qui lancera l'installation du serveur IBM UniVerse.

La première fenêtre vous demande la langue de l'installation. Pour éviter des erreurs de traductions nous avons choisi l'anglais (English).

La prochaine fenêtre vous annonce que vous devriez fermer les applications Windows en cours, ce que nous vous conseillons fortement de faire. Ensuite une fenêtre avec les *Terms and Conditions* vous sera présentée.

La fenêtre suivante est la plus importante, le module d'installation vous demande dans quel dossier vous devriez installer IBM UniVerse. Nous vous recommandons de choisir un dossier **sans espaces** cela vous permettra d'éviter des erreurs dans le futur. Dans notre cas, nous avons décidé de rester avec les dossiers par défaut soit "C:\IBM\UV" et "C:\IBM\unishared". Après avoir fait *Next* le système devrait, en principe, vous avertir que les dossiers n'existent pas et qu'il va devoir les créer.

Ensuite, le système demande s'il doit utiliser le *National Language Support* pour UniVerse. Nous avons décidé de ne pas l'installer puisque nous n'en aurons pas besoin au cours de ce prototype.

Pour ce qui est du *Console Code Page* nous vous conseillons fortement de choisir l'écran qui vous ai conseillé par défaut, dans notre cas *PC850*, si vous ne savez pas lequel choisir. Si vous avez un écran dont vous connaissez le code, choisissez celui dans la liste qui vous convient.

Le prochain écran vous annoncera les services qui seront installés. Nous vous conseillons de garder les 3 services.

Pour le reste des écrans, il vous suffit de faire *Next* car ce sont des paramètres de Windows.

L'installation vous avertira qu'IBM à installer une *Master Key* à des fins d'encryptages. Pour les fins du prototype nous ne la changerons pas mais dans un cas futur il faudrait cependant envisager de la changer pour une *site specific Master Key* pour améliorer la sécurité.

Félicitation, l'installation devrait être terminée! Vous n'avez pas besoin de redémarrer votre système.



## Vérification de l'installation

### Services

Pour s'assurer que l'installation s'est bien passée il faut tout d'abord aller dans le menu démarrer -> Programmes -> IBM U2 -> UniVerse. Si vous avez suivis les instructions, ce dossier devrait être situé a cet endroit et sélectionner *UniVerse Control*.

Une fenêtre vous montrant que tous les services sont en marche devrait apparaitre. S'ils ne sont pas en marche, il faut les démarrer en cliquant *Start all services*.

### Test de connexion

Maintenant, il faut essayer de se connecter au serveur.

Pour vous connecter, il vous suffit d'ouvrir une fenêtre *Command Prompt* (Démarrer -> Exécuter -> cmd). Dans cette fenêtre, entré la commande suivante:

*telnet localhost* suivit d'un retour de charriot.

Vous devriez voir ce message:

*Welcome to the IBM UniVerse Telnet Server. Enter user name:*

Vous **devez** vous identifier avec un compte Windows qui à des droits administrateur et qui a un mot de passe. Cette étape est primordiale.

A des fins de test nous avons créé un compte Windows avec des droits d'administrateur sur notre ordinateur portable avec les informations suivantes:

*Nom d'utilisateur:* sto

*Mot de passe:* sto

Pour pouvoir accéder le système UniVerse avec un compte valide.

## Erreurs communes

**Problème:** La connexion Telnet semble ne pas répondre et j'utilise l'anti-virus NOD32.

**Solution:** Il faut supprimer NOD32, IBM semble dire que cela se limite à être la seule solution.

**Problème:** J'essaie de m'identifier avec un compte administrateur qui n'a pas de mot de passe.

**Solution:** Il faut aller mettre un mot de passe à votre usager, car UniVerse ne permet pas qu'un compte n'a pas de mot de passe.

## Configuration sur serveur UniVerse

### Créer un compte

Maintenant que le serveur est installé, il est temps de créer un compte qui nous servira à entreposer les modules de paie qui nous ont été fournis préalablement.

Pour se faire, il faut s'identifier via Telnet comme client du système UniVerse. Si vous ne savez pas comment, veuillez vous référer à la section « Test de connexion ».

Une fois cela fait, le système devrait vous demander le nom du compte ou le chemin vers le compte ("Account name or path()"). Avant de remplir ce champ il faut avoir créé un dossier où nous allons entreposer les applications IBAS. Chaque dossier créé va représenter une application ou un ensemble d'applications distinctes. Par exemple, il serait possible de créer un dossier Paie et un dossier Employé qui feraient tout deux affaire avec UniVerse mais qui feraient la gestion de processus différent.

Après avoir donné le chemin (qui devrait être sans espace, par exemple, dans notre cas nous utiliserons C:\IBM\UV\STO) valide UniVerse vous avertira que le dossier n'est pas configuré pour UniVerse et vous demandera de choisir la "saveur" (flavor) du dossier en question. Pour la transition nous avons décidé d'utiliser PICK (option 3) puisque cela semble être la technologie la plus stable et est très semblable avec PRIME.

Après avoir choisi cette option, vous êtes fin prêt à vous lancer dans l'aventure UniVerse!

## Annexe « C »

### *Migration des dictionnaires et des données*

Avant de commencer, il est important que la valeur qui sépare les données soit un "~" car les points-virgules ou les virgules peuvent être utilisés dans les *Field Definition* ce qui rendrait la tâche du parseur moins évidente.

### Procédure pour la migration des dictionnaires

Cette étape décrit pas à pas la procédure à utiliser pour parser un dictionnaire. Une fois ces étapes terminées vous serez en mesure de visualiser la structure du dictionnaire.

1. Compiler l'application LOAD.DICT en exécutant la commande suivante :
  - a. BASIC [Nom du fichier] LOAD.DICT
2. Prendre les deux fichiers, qui consistent des champs de type "I" (Imaginaire) et "D" (Données), et les mettre dans un seul fichier en commençant par les champs de type "D" suivis des champs de type "I".
3. Exécuter la commande suivante pour démarrer le parseur :
  - a. RUN [Nom du fichier] LOAD.DICT
4. Le programme devrait vous demander d'entrer la location du dictionnaire à parser, entrer le chemin qui mène à votre dictionnaire. Il est conseillé d'avoir un chemin et un nom de fichier sans espaces. (ex: C:\dictionnaire.txt)
5. L'application vous demandera ensuite le nom du dictionnaire que vous voulez lui donner dans UniVerse. Nous vous recommandons de garder les noms de fichier en majuscule. Vous devez aussi spécifier le même nom qu'utilise le fichier de données.
6. Si tout s'est bien passé un message de réussite apparaîtra avec le nombre de champs qui a été inséré dans le dictionnaire.

Le *listing* du dictionnaire est accessible via la commande suivante :

```
LIST DICT [Nom du dictionnaire]
```

## Procédure pour la migration des données

Cette étape vous permettra de migrer les données des anciens dictionnaires qui résident sur l'environnement PR1ME vers l'environnement UniVerse.

1. Compiler LOAD.REC en exécutant la commande suivante :
  - a. *BASIC [Nom du fichier] LOAD.REC*
2. Exécuter la commande suivante pour démarrer l'application :
  - a. *RUN [Nom du fichier] LOAD.REC"*
3. Le programme devrait vous demander d'entrer la location fichier de donnée a parser, entrer un chemin, préférablement sans espace. (ex: C:\fichier.txt)
4. L'application vous demandera ensuite le nom du fichier que vous voulez lui donner. Nous vous recommandons de garder les des fichiers en majuscule. Vous devez spécifier le même nom qu'utilise le dictionnaire.
5. Si tout s'est bien dérouler un message de réussite apparaîtra avec le nombre de données qui on été insérées dans le fichier.

Le *listing* du fichier de données est accessible via la commande suivante :

```
LIST [Nom du fichier]
```

Noter que seul la clé (@id) est affichée lors d'un affichage par défaut du dictionnaire. Pour afficher d'autres champs il faut entrer la même commande suivit des champs que vous voulez afficher.

```
LIST [Nom du fichier] Champ1 Champ2 Champ3 ...
```

## Annexe « D »

### *Pilote ODBC*

Les pilotes ODBC permettent aux applications de développement d'établir une connexion vers le serveur UniVerse. Cette annexe vous permettra d'installer le logiciel nécessaire pour la connexion, vous apprendrez également comment configurer la connexion ODBC et finalement vous serez capable de visualiser les données de la base de données UniVerse dans l'application Visual Studio.

### Installation du Pilote ODBC

Vous devez tout d'abord avoir téléchargé l'application client d'IBM UniVerse. Démarrer l'installation et choisir « Install IBM UniVerse ODBC Driver ». Veuillez suivre les étapes affichées à l'écran.

### Configuration de l'environnement UniVerse

A cette étape vous avez déjà installé le pilote ODBC de IBM et vous avez déjà vos bases de données installées sur UniVerse. Maintenant suivez les étapes suivantes pour faire la configuration.

1. Ouvrez une session à votre serveur UniVerse
2. Authentifiez vous au compte HS.ADMIN à l'aide de la commande *logto HS.ADMIN*.
3. Lancez l'application HS.ADMIN (UniVerse Server Administration). Tapez *HS.ADMIN*.
4. Choisissez l'option 3 : *Activate access to files in an account*
5. Spécifiez le nom ou le chemin de votre compte que vous voulez donner accès à ODBC Driver
6. Si tout s'est bien passé il n'y a pas de message d'avertissement ou d'erreur.

## Configuration de l'application ODBC Admin

Vous que vous avez donné accès aux bases de données, il est maintenant temps d'offrir l'accès à l'application de développement. Pour se faire il faut tout d'abord configurer l'application *ODBC Admin*. Veuillez suivre les étapes suivante.

1. Ouvrir l'application *ODBC Admin*.
2. Appuyer sur *Add*.
3. Choisissez le type de base de données, dans notre cas il s'agit de *IBM UniVerse ODBC Driver* et appuyer sur *Finish* lorsque terminé.
4. Maintenant entré les informations de connections. Le serveur est de type "localuv". Pour la base de données entre le chemin de votre compte UniVerse OU le nom spécifié. Le nom d'utilisateur et de mot de passe sont ceux de votre compte UniVerse.
5. Vous devez maintenant vous assurer que la connexion est bien établie en appuyant *Test connexion*, si tout s'est bien déroulé un message vous affichera le succès de l'opération.
6. Appuyez sur "Ok" et maintenant vous pouvez fermer *ODBC Admin* ou ajouter de nouvelles connections.

## Faire la connexion dans Visual Studio

Vous êtes alors prêt à utiliser la connexion et la base de données UniVerse à l'intérieur d'application de développement, dans le cas présent nous utiliserons Visual Studio 2005. Veuillez suivre les étapes suivantes pour offrir à Visual Studio la possibilité de communiquer avec le serveur UniVerse au travers des pilotes ODBC.

1. Ouvrir l'environnement de développements de Visual Studio. (Noter qu'il n'est pas nécessaire de créer un nouveau projet).
2. Dans le menu "View" choisir l'option "Server Explorer".

3. Avec le bouton droite de votre souris appuyer sur la fenêtre du "Server Explorer" et choisir "Add Connection".

4. Dans la fenêtre "Chose Data Source" choisir 'IBM UniVerse ODBC Driver" ensuite Choisissez "localuv" "dans Data Provider", un fois fini appuyer sur "Continue".

5. Dans la prochaine fenêtre veuillez entre les informations relatives au compte UniVerse "username" and "password" (Nom d'utilisateur et mot de passe).

6. Vous pouvez maintenant tester la connexion et appuyer sur "Ok" si tout s'est bien dérouler.