

## Document de conception du projet SIRIUS

Projet	SIRIUS
Document	Document de conception
Version	2.0
Date	24 septembre 2003
Référence	CO1

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Objectif global du logiciel . . . . .	2
1.2	Documents de référence . . . . .	2
1.3	Historique des versions du document . . . . .	2
1.4	Statut du document . . . . .	2
<b>2</b>	<b>Architecture globale du système</b>	<b>4</b>
<b>3</b>	<b>Documents de conception</b>	<b>5</b>
3.1	Choix entre Merise et UML . . . . .	5
3.2	Dictionnaire de données . . . . .	5
3.3	Modèle Conceptuel des Données (MCD) . . . . .	6
3.4	Liste des tables de la base de données . . . . .	12
3.5	Liste des index, vues, ou procédures stockées . . . . .	12
3.6	Base de données distribuée . . . . .	13
<b>4</b>	<b>Spécifications détaillées du système</b>	<b>14</b>
4.1	Liste des écrans . . . . .	14
4.1.1	EC1 : Lancement application . . . . .	14
4.1.2	EC2 : Inscription . . . . .	15
4.1.3	EC3 : Emploi du temps . . . . .	19
4.1.4	EC4 : Gestion des groupes . . . . .	20
4.1.5	EC5 : Barre des icônes . . . . .	20
4.2	Liste des états . . . . .	21
4.2.1	ET1 : Etats des étudiants . . . . .	21
4.2.2	ET1 : Etats des groupes . . . . .	21
4.2.3	ET1 : Etats des emplois du temps . . . . .	22

# Chapitre 1

## Introduction

### 1.1 Objectif global du logiciel

Le projet SIRIUS est un projet open source de développement d'une application de gestion de la scolarité au niveau des établissements de l'enseignement supérieure au maroc. Initié par l'université Hassan II Mohammedia en partenariat avec l'association OSIM<sup>1</sup>, ce projet a pour ambition de créer une synergie entre les compétences et les ressources au sein des universités et écoles marocaines avec les développeurs libres de part le monde.

### 1.2 Documents de référence

Rapport de spécification du projet SIRIUS référence SP1 version 1.0.

### 1.3 Historique des versions du document

Version	Date	Auteur(s)	Apport
1.0	16/6/2003	Tarik Fdil	Création du document
2.0	24/9/2003	Tarik Fdil	Nouveaux écrans, modification schéma base de données

### 1.4 Statut du document

**Vérification :**

Statut	non vérifié
Vérificateur(s)	
Date vérification	

---

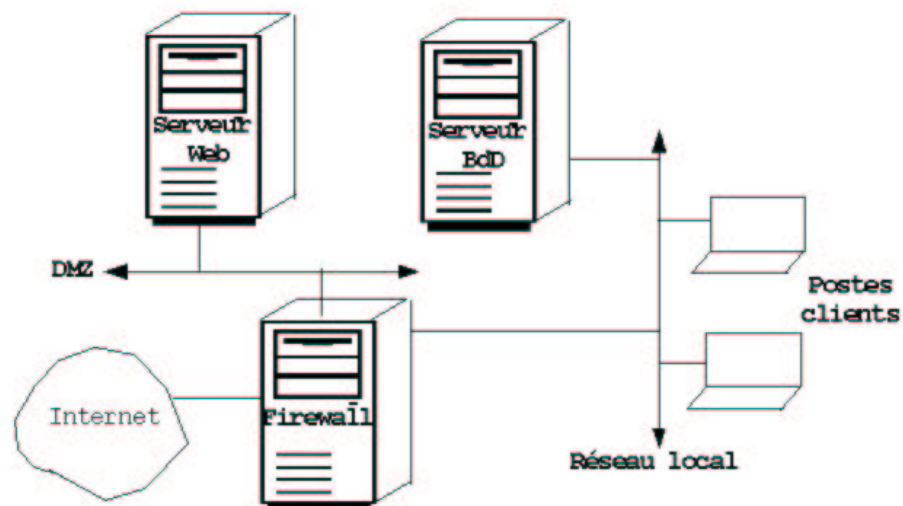
<sup>1</sup> Association Open Source In Morocco <http://www.linux-maroc.org>

**Validation :**

Statut	non validé
Valideur(s)	
Date validation	

## Chapitre 2

# Architecture globale du système



L'architecture choisie est une architecture à trois tiers avec un serveur base de données faisant office de serveur d'application. Un serveur web répond aux requêtes internet. Les postes clients sur le réseau local peuvent travailler en client serveur ou en mode intranet.

Le firewall relié à la DMZ, au réseau local et au réseau internet fait office de routeur et de filtre contrôlant le trafic et protégeant le réseau local et le serveur base de données contre toute tentative d'intrusion.

Les serveurs web et base de données tourneront sous GNU/Linux. Le SGBD choisi est postgresql. Les postes clients peuvent indifféremment être sous GNU/Linux ou sous windows sous toutes ses versions. Le firewall est préférable sous openBSD mais il peut très bien être sous GNU/Linux.

## Chapitre 3

# Documents de conception

### 3.1 Choix entre Merise et UML

Le choix adopté pour spécifier ce projet est le formalisme MERISE. Cela n'empêchera pas que l'on puisse utiliser UML pour spécifier certaines parties du projet.

### 3.2 Dictionnaire de données

Le dictionnaire initial n'a pas été respecté et a été corrigé à plusieurs niveaux. Il faudrait maintenant extraire à partir du schéma de la base de données le nouveau dictionnaire de données et le transcrire ici. L'avantage d'un dictionnaire est d'uniformiser les appellations de champs dans une base de données et leur donner une sémantique universelle.

#### Codification des champs en arabe

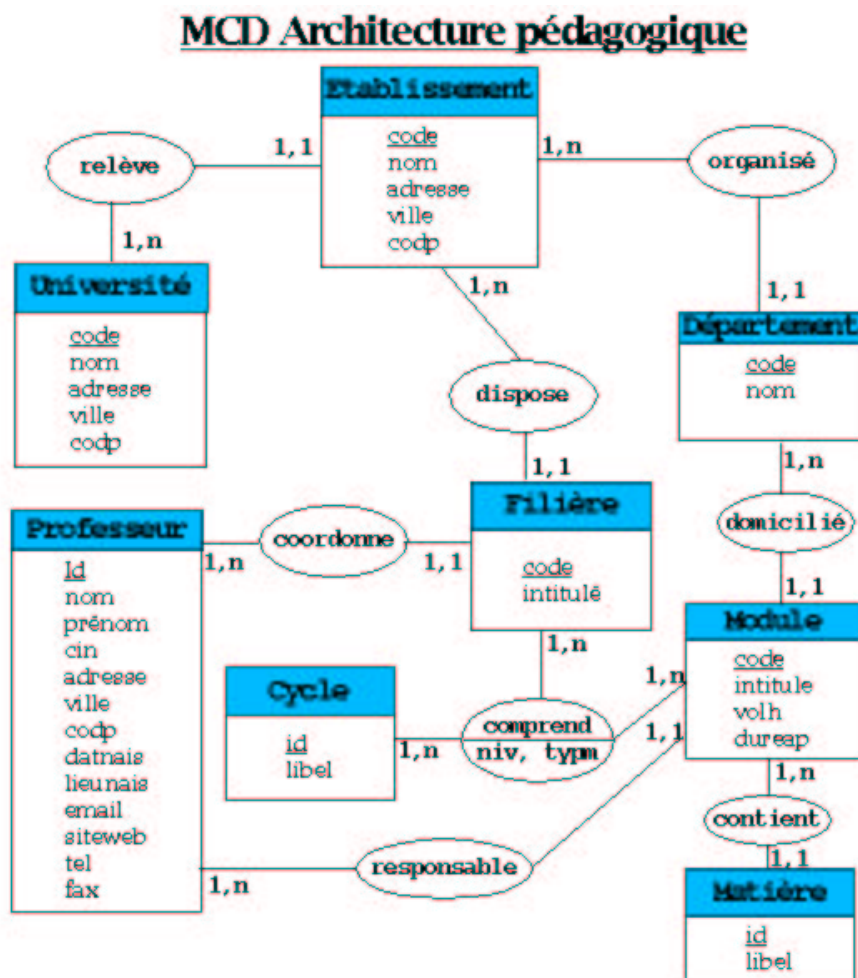
L'encodage utilisé pour les champs en arabe est Unicode UTF-8. Le nom du champ contenant le texte en arabe est codé avec le même nom que le champ correspondant en français avec en plus un suffixe '\_ar'. Par exemple 'prenom\_ar' signifie que le champ contient le prénom en arabe encodé en utf-8.

#### Codification des clefs 'étrangères'

Il est préférable que les clefs étrangères ait le même nom que leur correspondant dans les tables associés. Ceci afin de faciliter l'écriture des requêtes de jointure.

### 3.3 Modèle Conceptuel des Données (MCD)

MCD domaine architecture pédagogique



#### Description des entités

**Etablissement** Cette entité stocke les informations relative à l'établissement faculté ou école supérieure.

**Université** Contient les informations relatives à l'université.

**Filière** C'est l'entité qui stocke le code et le libellé des filières types ou propres à un établissement.

**Module** C'est l'entité décrivant un module d'enseignement.

**Matiere** c'est l'entité incluant les matières enseignées dans un module.

**Professeur** est l'entité incluant les informations sur les enseignants.

**Cycle** décrit les cycles d'enseignement possibles : DEUG/DEUP, LF/LP, etc.

#### **Description des relations**

**relève** chaque établissement relève d'une université.

**organisé** chaque établissement est organisé en départements.

**dispose** chaque établissement dispose d'un certain nombre de filières.

**domicilié** chaque module est domicilié au niveau d'un département.

**contient** chaque module contient des matières.

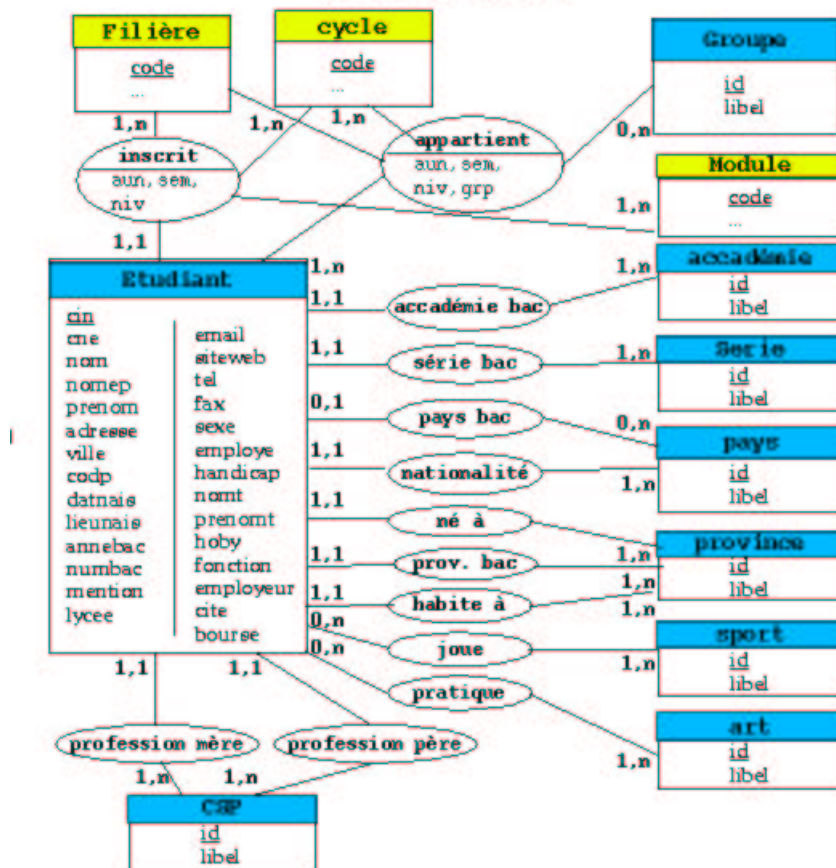
**comprend** une filière contient des modules déterminés en fonction du cycle d'enseignement, de la filière et du niveau. Un module est inclu dans un cycle soit en tant que module majeur, complémentaire ou transversal.

**coordonne** une filière est coordonnée par un professeur.

**responsable** un module a pour responsable un professeur.

## MCD domaine inscription

### MCD Inscription



NB. "\*" dans les propriétés d'une entité signifie que celle-ci est définie dans un autre MCD

## Description des entités

**Etudiant** Inclut toutes les informations relatives à un étudiant.

**Groupe** Les étudiants peuvent être regroupés dans des groupes de TD, TP, section, etc. Ces types de groupe sont stockés dans cette entité. On peut créer autant de types de groupes qu'on veut.

**pays** Entité contenant les codes/libellés des pays. Nous utiliserons le code téléphonique international comme code pays.

**seriebac** série du baccalauréat : sciences, lettres, mathématiques, etc.

**accadémie** liste les accadémies du royaume où la session du bac de l'étudiant est passée.

**serie** c'est la série du baccalauréat de l'étudiant : sciences, lettres, etc.

**province** code/libellé de la province du royaume.

**csp** catégorie socio-professionnelle des parents.

**sport** sports pratiqués.

**art** activités artistiques pratiquées.

### **Description des relations**

**inscrit** L'inscription d'un étudiant s'effectue dans des modules pour une filière dans un cycle donné. Elle est effective pour une année universitaire, un semestre et un niveau donnés.

**appartient** Un étudiant peut appartenir à plusieurs groupes de manière tout à fait dynamique. Cette appartenance est valable pour une année universitaire, un semestre, un niveau et un numéro de groupe donnés.

**nationalité** bien qu'il puisse y avoir des personnes avec plusieurs nationalités, pour des raisons de simplicité nous supposons que chaque étudiant à une seule nationalité.

**série bac** le bac obtenu par l'étudiant est d'une série donnée.

**accadémie bac** académie où le bac a été obtenu.

**pays bac** relation utile surtout si le baccalauréat est étranger.

**né à** province où est né l'étudiant.

**prov. bac** la province où le bac a été obtenu.

**habite à** province où habite les parents de l'étudiant.

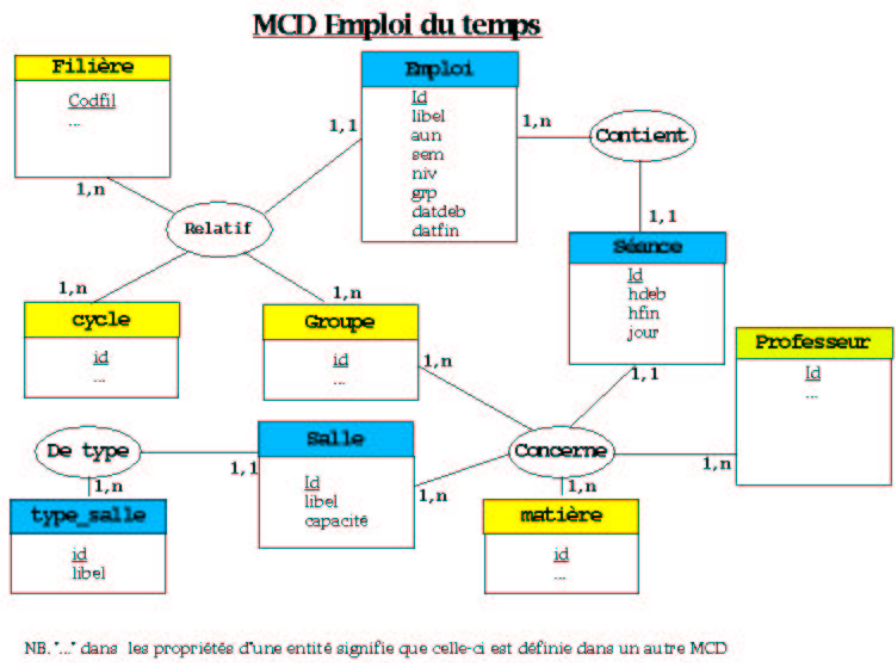
**joue** liste des sports pratiqués par l'étudiant.

**pratique** liste des activités artisitiques de l'étudiant.

**profession père** catégorie socio-professionnelle du père.

**profession mère** catégorie socio-professionnelle de la mère.

## MCD domaine Emploi du temps



### Description des entités non décrites dans le mcd précédent

**emploi** est l'entité qui consigne les informations sur les emplois du temps. Un emploi du temps est le plan, par tranche de temps, d'occupation des salles par les groupes d'étudiants et par les enseignants pour enseigner leurs matières. Un emploi du temps peut être limité dans le temps, par exemple, on peut créer un emploi spécial Ramadan.

**seance** est l'entité regroupant les séances composant un emploi du temps. Une séance a une heure de début et une heure de fin, elle a lieu un jour de la semaine.

**salle** est la salle où se déroule un cours.

**type\_salle** c'est soit amphithéâtre, salle de td, salle de tp, etc.

### Description des relations

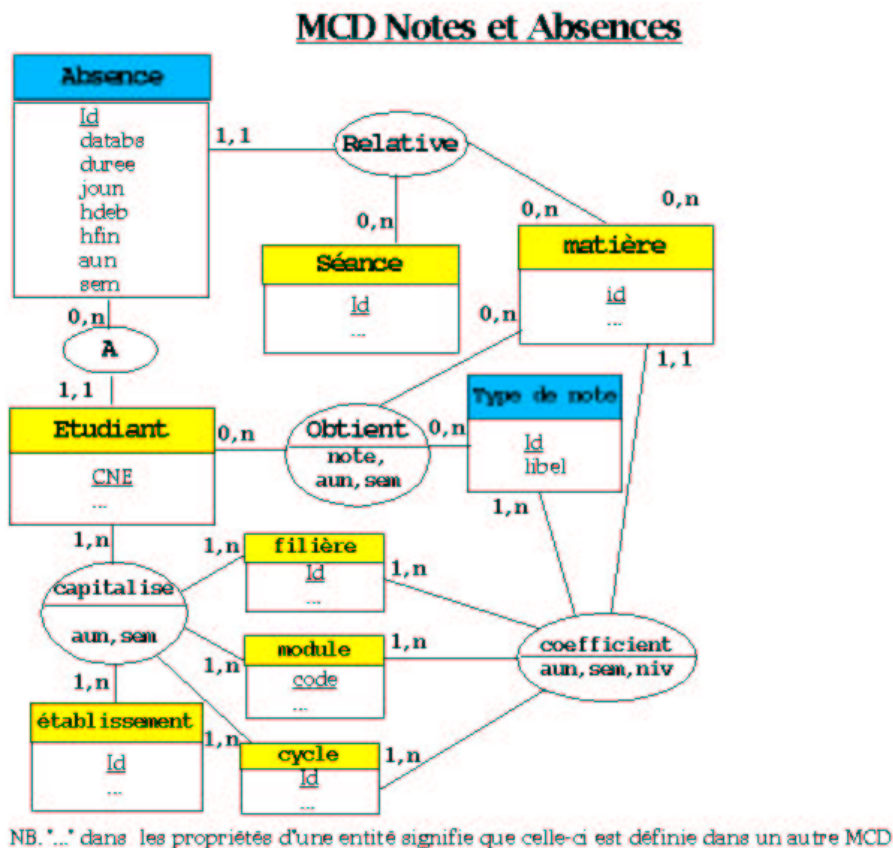
**relatif** un emploi est défini pour une filière, cycle et un groupe d'étudiants.

**contient** chaque emploi est constitué de plusieurs séances.

**concerne** une séance d'emploi du temps concerne une salle, une professeur, une matière et un groupe (td, tp, etc.).

**de type** une salle est soit un amphithéâtre, une salle de tp, td, etc.

## MCD domaine absences et notes



### Description des entités non décrites dans les mcd précédents

**type de note** cela peut être 'note de td', 'note orale', 'note examen final', 'note contrôle continu', etc. Cette entité permet de prendre en compte le type d'évaluation que le responsable du module veut appliquer dans les matières relevant de ce module.

**absence** cette entité consigne toutes les absences d'un étudiant.

### Description des relations

**obtient** un étudiant obtient une note par type de note et par matière.

**a** un étudiant à zéro ou plusieurs absences.

**relative** une absence est relative à une séance d'emploi du temps. Impliquer l'entité 'matière' dans la relation ou stocker les heures de débuts et fin dans l'entité peut paraître redondant car toutes ces informations sont stockées

dans l'entité 'séance'. En fait l'explication de cette redondance c'est que l'emploi du temps peut évoluer dans le temps et changer et nous ne gardons par l'historique des changements d'un emploi du temps. Par conséquent pour garder l'historique des absences, il faut stocker les informations pertinentes dans l'entité absence.

**coefficient** chaque matière a un coefficient dépendant du type de note, de la filière, du module, du cycle et du niveau. Eventuellement, le coefficient peut varier d'une année universitaire ou d'un semestre à l'autre.

**capitalise** un étudiant peut avoir capitalisé des modules dans un établissement durant une année universitaire/semestre dans une filière et cycle donnés.

### 3.4 Liste des tables de la base de données

Les tables issues des entités sont les suivantes : etablissement, filiere, universite, departement, module, matiere, cycle, professeur, etudiant, accademie, groupe, serie, pays, province, sport, art, csp, emploi, seance, salle, typsalle, typnote, absence.

Les tables issues des relations sont les suivantes :

**inscrit** relation entre etudiant, filière et cycle.

**suit** relation entre etudiant et module.

**comprend** relation entre filière, cycle et module

**appartient** relation entre etudiant, cycle et groupe

**joue** relation entre etudiant et sport.

**pratique** relation entre etudiant et art.

**note** relation entre etudiant, typnote et matière.

**capitalise** relation entre etudiant, etablissement, filiere, module et cycle.

**coefficient** relation entre matiere, typnote, filiere, module et cycle.

### 3.5 Liste des index, vues, ou procédures stockées

Toutes les clefs des tables donnent lieu automatiquement à un index créé par le SGBD. Toutes les clefs étrangères importées dans une table doivent donner lieu à la création d'un index ayant pour nom 'clef\_etrangere\_ndx'. Nous allons présenter ici les index créés automatiquement par un script au moment de la création des tables en dehors des deux types d'index cités précédemment.

**idref\_code\_etab\_ndx** index clef étrangère code établissement dans dépa.

**cin\_ndx** index de recherche par cin dans la table étudiant

**nom\_ndx** index de recherche par nom dans la table étudiant

### 3.6 Base de données distribuée

La base de données est créée au niveau du serveur de chaque établissement. Cependant, elle est conçue pour être distribuée au niveau de tous les établissements de toutes les universités qui auront adopté SIRIUS. Pour cela nous distinguons deux types de tables : les tables communes et identiques entre tous les établissements et les tables propres à chaque établissement :

Tables communes	Tables spécifiques
université	filiere
etablissement	departement
etudiant	module
cycle	matiere
accademie	groupe
service	emploi
pays	seance
province	salle
sport	typsalle
art	typnote
csp	absence
capitalise	inscrit
professeur	suit
	comprend
	appartient
	joue
	pratique
	note
	coefficient

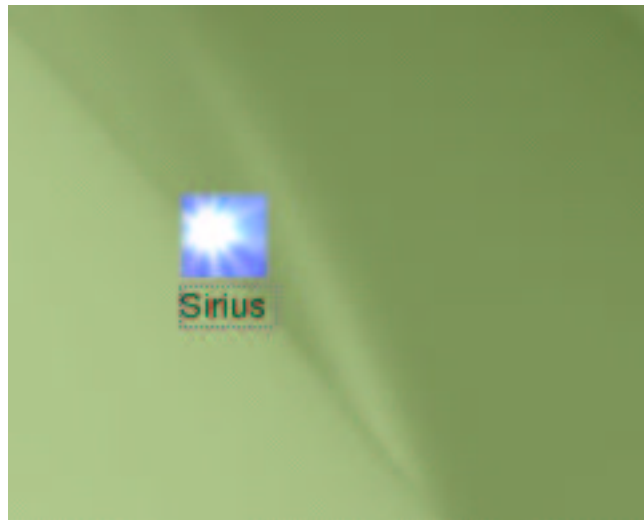
Les tables communes sont synchronisées automatiquement entre tous les sites hébergeant SIRIUS, tout changement fait au niveau d'un établissement doit être reflété au niveau de tous les autres. Par exemple, un établissement qui inscrit un étudiant, ce dernier est automatiquement pris en compte par les systèmes d'information de tous les autres établissements. Les autres tables, sont spécifiques à un établissement et contiennent des données évoluant indépendamment des autres établissements. La synchronisation des tables communes entre les établissements s'effectuera en utilisant des fichiers au format XML et des démons échangeant et intégrant les données au niveau de chaque base. La fréquence de mise à jour des données ainsi que la procédure exacte d'échange sera décrite dans une version ultérieure de ce même document.

## Chapitre 4

# Spécifications détaillées du système

### 4.1 Liste des écrans

#### 4.1.1 EC1 : Lancement application



Une icône sur le bureau permet de lancer l'application.



Le fenêtre qui s'affiche présente la boîte de dialogue d'authentification dans Sirius.

#### 4.1.2 EC2 : Inscription

L'onglet "informations personnelles" permet de saisir les données personnelles de l'étudiant.

Sirius 0.1

File Edit Déplacement Paramétrage Help

Informations personnelles Parents Baccalauréat Informations complémentaires Parcours pédagogique Inscription

Nom du père:  prénom du père:

Nom du mère:  Prénom de la mère:

Nom tuteur:  Prénom tuteur:  N° CIN du tuteur:

Adresse:

Commune:  Province ou préfecture:

Ville:  Code postal:

Tel:  Fax:

Email:  Site web:

Profession Père:  Profession Mère:

Etudiant Inscription 24.09.2003

L'onglet "Parents" permet de saisir les données concernant les parents et le tuteur de l'étudiant.

Sirius 0.1

File Edit Déplacement Paramétrage Help

Informations personnelles Parents Baccalauréat Informations complémentaires Parcours pédagogique Inscription

Année d'obtention:  سنة 1 للحصول عليها

Série du bac:  شعبة 1 لبا كلوريا

Province:  لاقليم

Mention:  لميزة 1

Accadémie:  لأكا ديمية

Pays:  لبلد 1

Lycée:  لسا سوية 1

Etudiant Inscription 24.09.2003

L'onglet "Baccalauréat" permet de saisir toutes les données relatives au baccalauréat obtenu par l'étudiant.

Fonction(s) vous travaillez:  Organisme employeur:   
☐ En cas d'handicap, cochez ici: N° carte d'handicap:   
 Sport  
☐ haut\_niveau ☐ collectif ☐ individuel  
 Art  
☐ musique ☐ danse ☐ theatre  
 Note:   
 Autre activité artistique :   
 Bourse  
☐ Boursier ☐ Demi\_boursier ☐ Non\_boursier  
 Résidez-vous en Cité Universitaire ou en Internat?   
 Laquelle ou lequel?

Etudiant **Projet Sirius** Inscription 24.09.2003

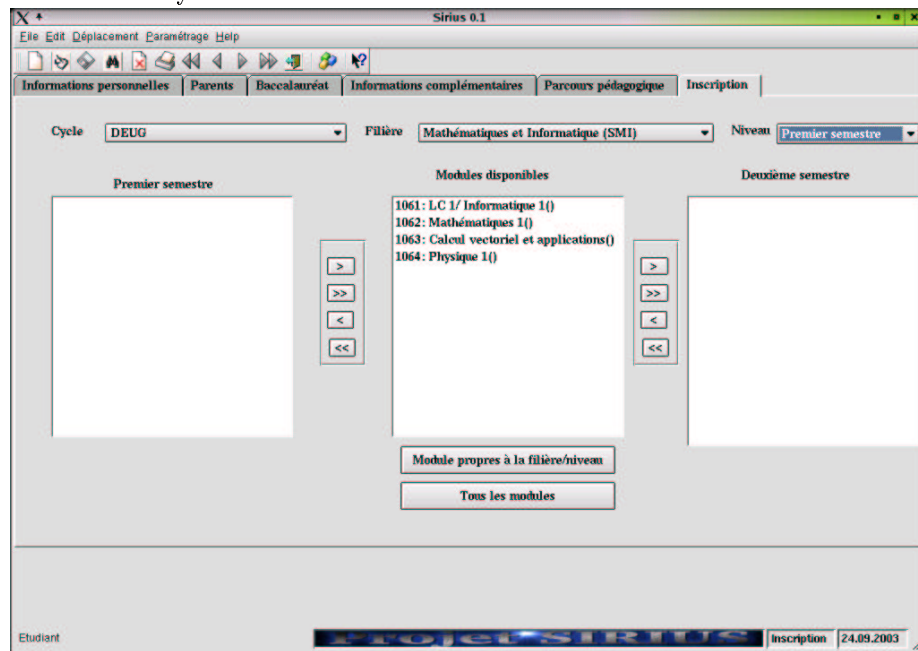
L'onglet "Informations complémentaires" permet d'enregistrer les activités sportives ou artistiques pratiquées par l'étudiant, l'organisme employeur éventuel, l'état de la bourse, etc.

Année	Etablissement	Département	Spécialité	Niveau
2001/2002	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
2002/2003	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Etudiant **Projet Sirius** Inscription 24.09.2003

L'onglet "Parcours pédagogique" est un écran temporaire permettant de saisir l'historique pour des étudiants qui redoublent leur première année et qui rentrent

dans le nouveau système.



L'inscription pédagogique permet d'afficher d'entrée tous les modules normalement devant être suivis par un étudiant. Les flèches permettent alors de sélectionner ou désélectionner les modules effectivement suivis par l'étudiant. En effet, il se peut qu'il ait déjà capitalisé un module et il n'est pas alors obligé de le suivre.

### 4.1.3 EC3 : Emploi du temps

Année universitaire  Semestre

Filière  Cycle  Niveau  Module

Type de groupe  N° de groupe

	jour	HD	HF	salle	element	TD/TP	Enseignant
1							
2							
3							
4							
5							
6							
7							

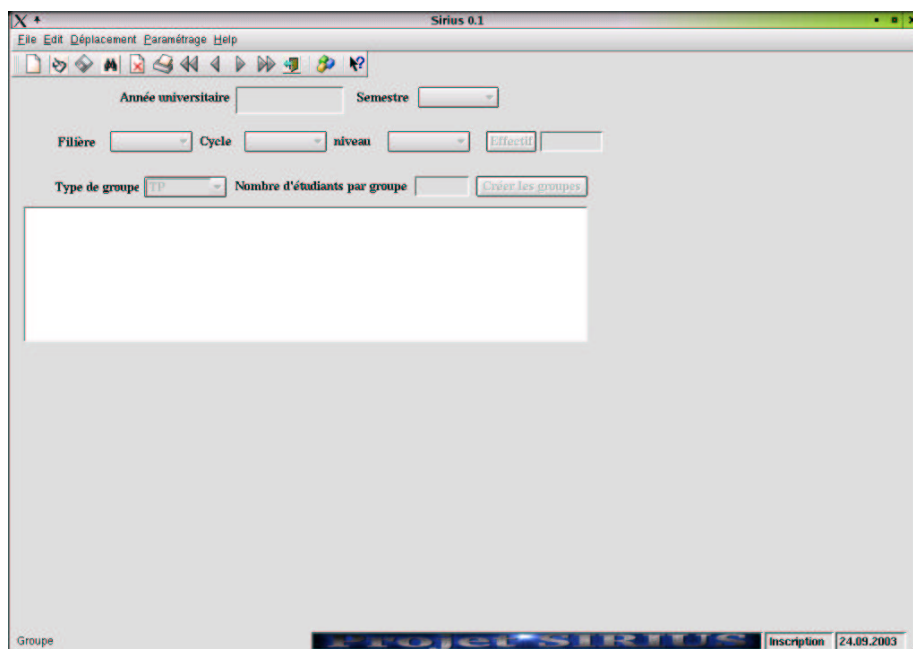
Emploi

PROJET SIRIUS

Inscription 24.09.2003

L'écran "Emploi du temps" permet de saisir les emplois du temps.

#### 4.1.4 EC4 : Gestion des groupes



La notion de groupes dans Sirius est dynamique. Un établissement peut définir autant de type de groupe qu'il le souhaite. Il y a bien évidemment les types groupes classiques : section, TD ou TP mais on peut très bien définir des groupes spécifiques pour un séminaire particulier par exemple. L'écran ci-dessus permet de créer automatiquement ces groupes en répartissant les étudiants selon le nombre fixé par l'utilisateur.

#### 4.1.5 EC5 : Barre des icônes

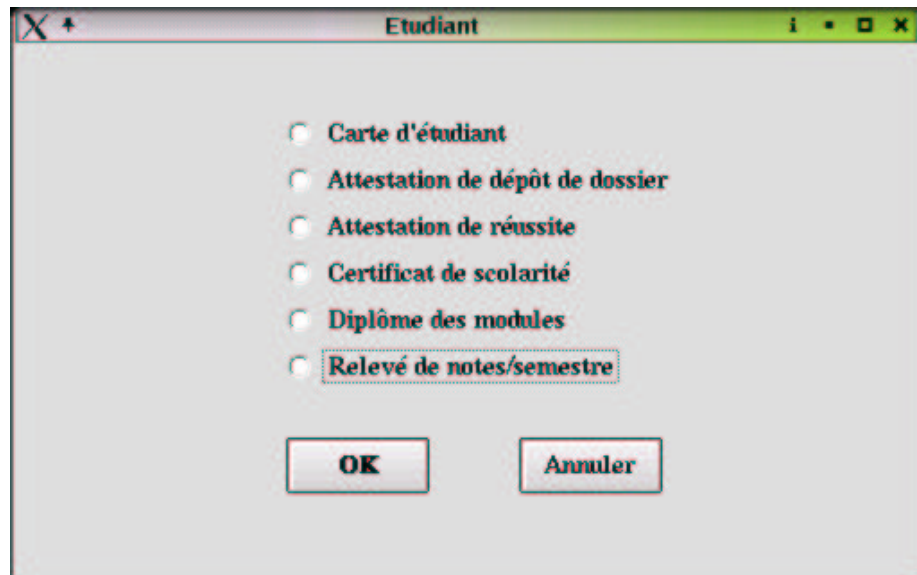


L'utilisation de cette barre des icônes permet de simplifier l'utilisation de l'application. Dans tout les écrans on utilise cette même barre soit pour créer un nouvel enregistrement, ou pour le modifier, le sauvegarder, le supprimer, rechercher ou

se déplacer parmi les résultats de la recherche.

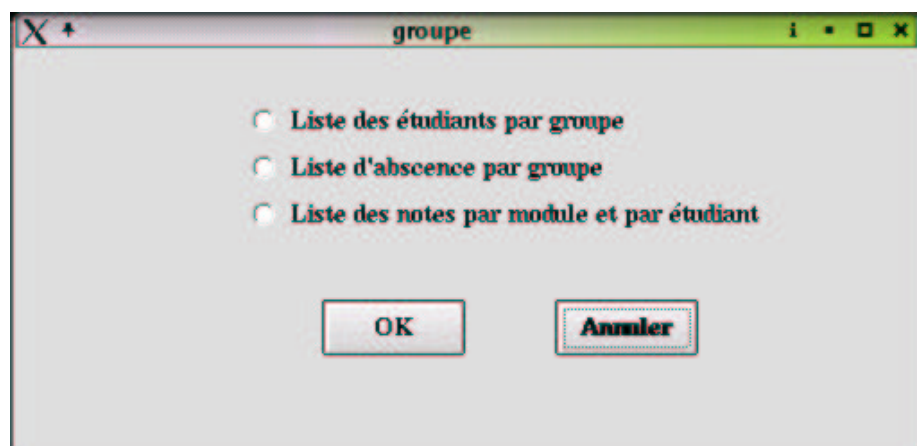
## 4.2 Liste des états

### 4.2.1 ET1 : Etats des étudiants



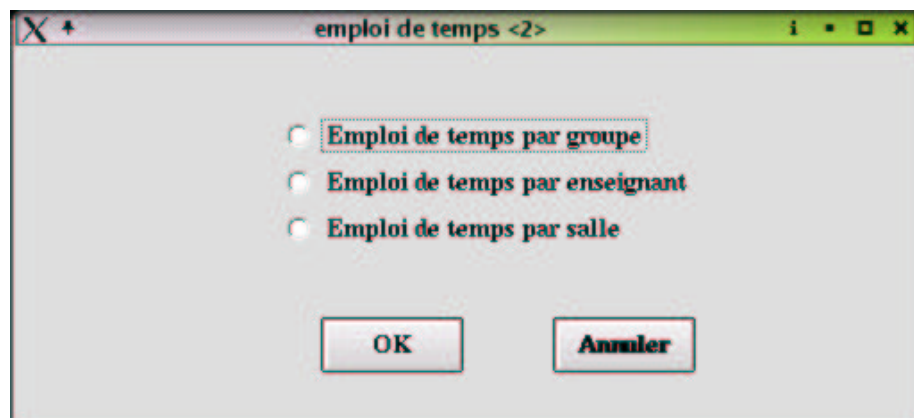
Cette fenêtre lancée à partir de l'écran "Etudiant" permet de sélectionner l'état à imprimer.

### 4.2.2 ET1 : Etats des groupes



Cette fenêtre lancée à partir de l'écran "Groupe" permet de sélectionner l'état à imprimer.

#### 4.2.3 ET1 : Etats des emplois du temps



Cette fenêtre lancée à partir de l'écran "Emploi" permet de sélectionner l'état à imprimer.