

Support de cours Initiation au développement sur Sage ERP X3 V12

Module 2 langage et actions sur champs



Licence Pro SIL Parcours ERP - IUT Paris 5

Cursus 3 jours ; **Module 2, 2^{ème} jour**

Rédacteur : Sébastien DE BOTTON
Consultant / Formateur sur SAGE ERP X3

Objectif du module 2

Connaitre les bases du langage et les actions sur champs dans SAGE ERP X3 V12





Plan du cours

1. Introduction
2. Les variables
3. Conditions / itérations
4. Gestion des messages
5. Gestion des traces
6. Actions sur champs
7. Eclipse
8. Eclipse debug
9. Continuation Exercice

1.Introduction

- Accès aux traitements :
 - Editeur de traitements :
 - Développement / Dictionnaire des traitements / Traitements
 - Ecran contenant un nom de traitement :
 - Clic droit sur le champ → Editeur (ex : dictionnaire des écrans)
- Barre d'outils :
 - Boutons compilation, sauvegarde, exécution, etc.
 - Raccourcis compilation (F7), sauvegarde(Ctrl+S), exécution (F5), etc.
- Structure du traitement :
 - Actions sur modèle : INICRE, MODIF, etc.
 - Actions Champs : AM_ITMREF, C_NBLIG, etc.
- Remarque : L'éditeur de traitement Eclipse peut aussi être utilisé pour développer en L4G

2. Les variables : déclaration

Classes de variables (visibilité)	Types	Nom de variable	Dimension
Global Local	Integer Shortint Decimal Char Date Clbfile Blbfile	Variable_name	(nb) ou (nb1..nb2)
Local	File	Table_name	
Local	Mask	Masque_name	

A savoir :

- Les variables sont initialisées à la déclaration
- L'instruction 'Raz' ré-initialise une variable

2. Les variables : déclaration

► Déclaration de variables dimensionnées

- Déclaration d'un tableau constitué de 8 entiers:
 - Local integer TABINT(8) → indice de 0 à 7
 - Local integer TABINT(1..8) → indice de 1 à 8
- Déclaration d'un tableau de 8 dates :
 - Local date TABDAT(8) → indice de 0 à 7
 - Local date TABDAT(1..8) → indice de 1 à 8
- Déclaration d'une chaîne de 8 caractères (par défaut 30):
 - Local char LOCCHAR(8)
- Déclaration d'un tableau de 5 chaînes de 8 caractères :
 - Local char LOCCHAR(8)(5) → indice de 0 à 4
 - Local char LOCCHAR(8)(1..5) → indice de 1 à 5

► Initialisation :

- Integer, shortint, decimal : 0
- Date : [0/0/0]
- Char : « »

2. Les variables : opérations


- Opération sur chaînes :
 - Chaîne1 + Chaîne2 : concatène
 - Chaîne1 - Chaîne2 : concatène en insérant un espace
- Opérations sur numériques :
 - Num1 += Num2 : ajout de num2 à num1
 - Idem pour -, *, /
- Opération sur les dates :
 - Date1 - Date2 : donne le nombre de jour entre les deux dates
 - Date +/- Num : donne la date plus Num jours

2. Les variables : Gestion des tables

- Ouverture d'une table (file)
 - Instruction :
 - Local File NOMTABLE [ABV]
 - Utilisation :
 - [F:ABV]NOMCHAMP = VALEUR
 - A savoir :
 - Ouverture multiple possible sous différentes abréviations
 - Possibilité d'indiquer un filtre de sélection
 - Possibilité d'indiquer un ordre de sélection
 - Pour tester si une table est déjà ouverte :
 - Clalev([F:ABV]) renvoie 1 si la table est ouverte, 0 sinon
- Fermeture d'une table
 - Instruction :
 - Close local File [F:ABV]

2. Les variables : Gestion des écrans

- Ouverture d'un écran (mask) :
 - Instruction :
 - Local Mask NOMMASK [ABV]
 - Utilisation :
 - [M:ABV]NOMCHAMP = VALEUR
 - A savoir :
 - Les menus locaux sont exprimés en integer
 - Les variables de bas de tableaux sont en shortint
 - Les autres champs numériques sont en décimal
 - Pour tester si un écran est déjà ouvert :
 - Clalev([M:ABV]) renvoie 1 si l'écran est ouvert, 0 sinon
- Fermeture d'un écran :
 - Instruction :
 - Close local Mask [M:ABV]



2. Les variables : Variables Globales

- Diverses variables globales :
 - [V]GUSER : Code utilisateur
 - [V]GNOMUSER : Nom utilisateur
 - [V]GLANGUE : Code langue courante
 - [V]GREP : Contexte en cours
 - [V]GFONCTION : Fonction courante
 - [V]GACTION : Action courante
 - [V]GTRACE : Nom du fichier de trace
 - [V]GOK : Communiquer avec le superviseur
 - [V]GERR : Type de message d'erreur



2. Les variables : les variables système

- fstat : statut de retour d'une instruction sur table (0 = ok)
- noline : numéro de ligne courant dans un tableau déroulant
- mkstat : statut d'erreur en saisie de zone (0 = ok)
- status : statut de retour suite à une saisie
- nomap : application courante

3. Conditions / Itérations

► IF

If expr1

[*instructions*]

[**Elsif** exprn

[*instructions*]]

[**Else**

[*instructions*]]

EndIf

► CASE

Case expr_choix

[When expr_v : [*instructions*]]

[When expr_v1,expr_v2 : [*instructions*]]

[When Default : [*instructions*]]

EndCase

3. Conditions / Itérations

➤ WHILE

Réalise une instruction tant qu'une condition est vraie

While expr
instructions

Wend

➤ REPEAT

Réalise une instruction jusqu'à réalisation d'une condition

Repeat
instructions

Until expr

➤ FOR

For variable_num = valeur_init **To** valeur_fin [**Step** pas]

instructions

Next [variable_num]

4. Gestion des messages

➤ Gestion des erreurs :

➤ ONERRGO :

- permet le débranchement en cas d'erreur vers l'étiquette déclarée dans le traitement. Permet de gérer la suite à exécuter après une erreur programme.

➤ RESUME :

- Retour au traitement initial après l'instruction à l'origine de l'erreur par l'ordre **Resume**

➤ Exemple :

```
Onerrgo ERREUR  
    instructions ...
```

```
Onerrgo  
End
```

```
$ERREUR  
Infbox("Erreur dans traitement"-errp-"en ligne"-errl  
Resume
```

4. Gestion des messages

- Afficher un message d'information (1 bouton):
 - Call MESSAGE(TEXTE) From GESECRAN
- Afficher un message d'erreur (1 bouton):
 - Call ERREUR(TEXTE) From GESECRAN
 - Call ERREUR(TEXTE,TIT) From GESECRAN
 - Call ERREND(TEXTE) From GESECRAN
- Afficher un message d'avertissement (2 boutons)
 - Call AVERTIR(TEXTE,OK) From GESECRAN
- Afficher une boîte de dialogue (2 boutons)
 - Call OUINON(TEXTE,OK) From GESECRAN

5. Gestion des traces

- Ouverture d'une trace
 - Call OUVRE_TRACE(« Titre de la trace ») From LECFIC
- Ecriture dans la trace
 - Message d'erreur :
 - Call ECR_TRACE(« Erreur ! », 1) From GESECRAN
 - Message d'information
 - Call ECR_TRACE(« Message », 0) From GESECRAN
- Fermeture de la trace
 - Call FERME_TRACE From LECFIC
- Lecture de la trace
 - Call LEC_TRACE from LECFIC

6.Actions sur champs

- Permet d'intervenir lors de la saisie de l'utilisateur
- On peut définir pour chacun des champs une liste d'action à exécuter
- Variables intéressantes :
 - Mkstat, nolign, fstat

Code écran SOH1 SOH1 Intitulé Gestion des commandes

◆ Général ◆ Champs

	Champ	Bloc	Pos	Intitulé	Col	Type	Menu	Long	Saisie	Mode
1	BPCINV	1	1	Client facture	1	BPC			Saisi	
2	BPINAM	1	1,2			NAM			Affiché	
3	BPCPYR	1	2	Tiers Payeur	1	BPR			Saisi	
4	BPCGRU	1	3	Client groupe	1	BPC			Saisi	
5	BPAADD	1	4	Adresse livraison	1	BPD			Saisi	
6	BPDNAM	1	4,2			NAM			Affiché	
7	PJT	2	1	Affaire	1	A		20	Saisi	
8	CNTNAM	2	2	Personne à contacter	1	NAM			Invisible	
9	PJTFLT	2	2	Filtre affaire	1	A		3	Invisible	

Type	Action	Intitulé	Exécution
1	Contrôle	CTLBPRCPY	Contrôle tiers/société
2	Contrôle	STD	Action de déroutement
3	Contrôle	SPE	Action de déroutement
4	Après_zone	TRTADRBPC	Traitement adresse c
5	Après_zone	STD	Action de déroutement
6	Après_modif	STD	Action de déroutement
7	Bouton2	SAIADRBPC	Adresse

Descriptif	Valeur
1	ABREV "ADB3"
2	BPRNUM VALEUR
3	ECRAN "ADB3"
4	SITE [M:SOH0]SALFCY
5	

6.Actions sur champs

Exercice

- Annexe langage sur les actions sur champs.
- Exercice N°3

7.Eclipse

Créer un projet

- Menu « File | New | Project ... »
 - Sélectionner « SAFE X3 Project »
 - « Next > »
 - Donner un nom pour le projet (ex: myserver_DEMO)
 - Renseigner les informations demandées (idem client X3)
 - Folder (ex: DEMO)
 - Hostname (ex: http://sage-vs1)
 - Port (ex: 2187)
 - X3 User et X3 Password
 - Language (ex: FRA, ENG, ...)
 - User et Password (authentification OS)
 - Help
 - ➔ Ne saisir que si l'info n'est pas présente dans le fichier solution.xml
- « Finish »



7.Eclipse

Ajouter des traitements

- Sélectionner le projet
- Menu « File | Open SAFE X3 Source file... »
 - Sélectionner dans la liste
 - Filtrer la liste
 - Saisir directement le nom d'un traitement
- Organiser en créant des dossiers
 - Clic droit sur un projet, puis « New | Folder »
 - Sélectionner les fichiers puis 2 options :
 - Déplacement à la souris dans le dossier
 - Clic droit menu « Move... »



7.Eclipse

Les vues

- Project Explorer : Arborescence des projets
- Navigator : Arborescence des fichiers
- Outline : Sous-programmes, Fonctions et traitements liés au traitement en cours d'édition
- Files properties : Infos de GESADC (ADICTRT)
- Locked Files : Fichiers verrouillés (*.LCKsrc)
- History : Historique local des modifications et/ou historique lié à un gestionnaire de sources (SVN, CVS, ...)
- Help : Aide dynamique
- Bookmarks : Visualiser ses repères dans les sources
- Problems : Erreurs de compilations, alertes sur la syntaxe
- Error Log : Erreurs, Informations liés au fonctionnement des plugins Eclipse
- Progress : Affiche les tâches d'arrière plan en cours.
- Search : Résultat des recherches dans les fichiers.
- Console : même possibilité que la calculatrice X3.

7.Eclipse

L'éditeur - 1

- Coloration syntaxique modifiable au menu **Window | Preferences** puis **SAFE X3 Studio | Editor**
- Multi-fenêtrage
- Passage en plein écran (Ctrl + M)
- Complétion (Ctrl + Espace) sur les mots clés, les variables globales, les variables systèmes
- Complétion sur les champs des abréviations (Tables et Masques) déclarées dans le traitement (Local File, Local Mask)
- Auto-fermeture des caractères fonctionnant par paires : ", ', (et [
- Bulle d'aide (tables, masques, variables globales, menus locaux)
- Templates de code et possibilité d'ajouter ses propres templates dans **Window | Preferences** puis **SAFE X3 Studio | Editor | Templates**



7.Eclipse

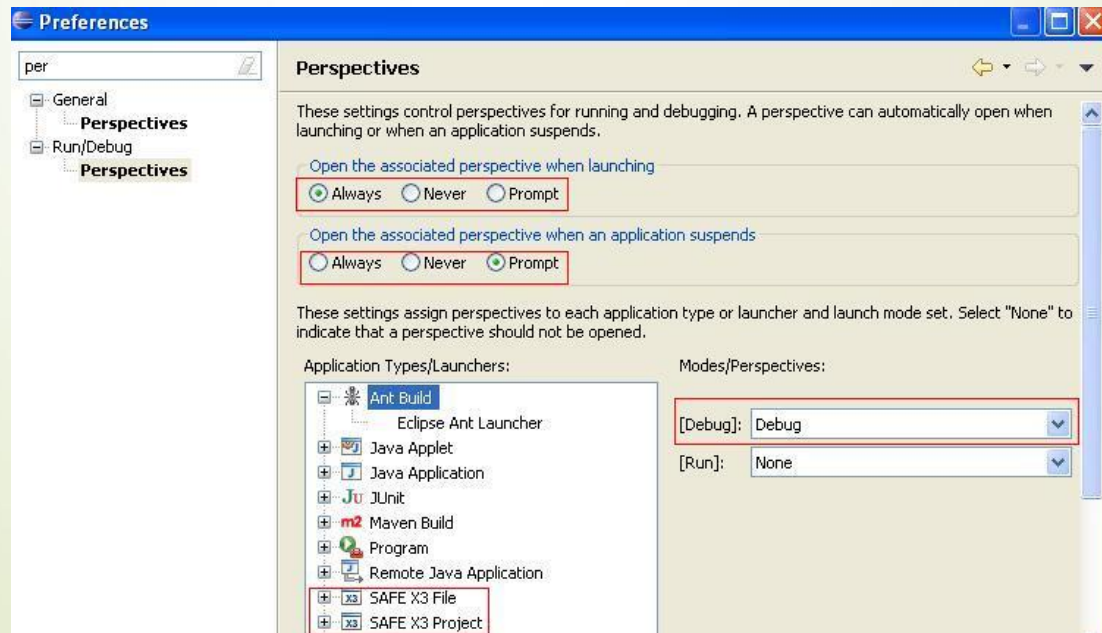
L'éditeur - 2

- Commenter / dé-commenter (Ctrl + /) une ou plusieurs lignes à la fois
- Indentation (Tab) d'une ou plusieurs lignes à la fois
- Atteindre les étiquettes, les procédures et fonctions (y compris celles déclarées dans d'autres traitements) selon la position actuelle du curseur (F3) ou Ctrl + click
- Détection des erreurs de syntaxe à la saisie
- Navigation dans l'historique des déplacements multi-fichiers (Alt + flèche gauche et Alt + flèche droite)
- Historique local des modifications (Clic droit, puis **Compare With | Local History...**)
- Recherche multi-fichiers (Ctrl + H) dans les fichiers du workspace, d'un projet, ...1

8. Eclipse Le debugger


Les pré-requis

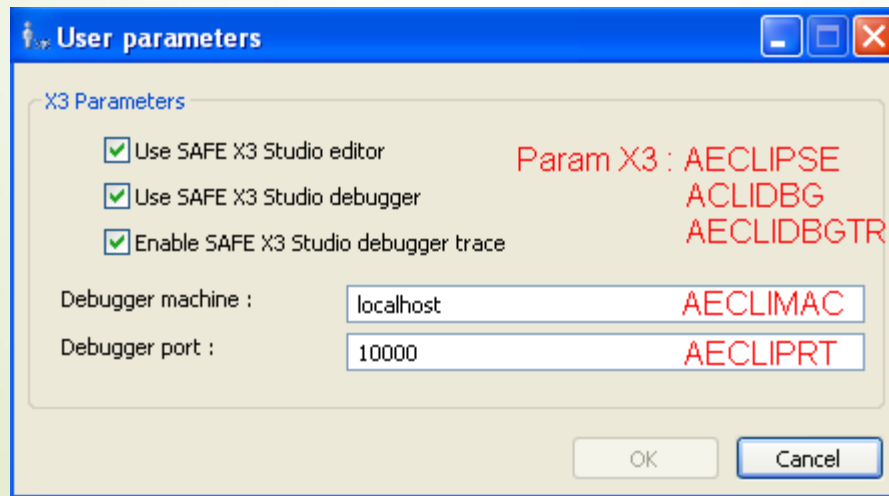
- Version majeur 6.
- Moteur 16r217 minimum.
- Maintenance 1849 (Patch 17 sur ERP V6).
- Client CS CLI.104 minimum.
- Paramètre X3 AECLIDBG à positionner à oui.
- Paramètres Perspectives Eclipse, propriété [Debug] doit être renseigné à « Debug » pour SAFE X3 File et SAFE X3 Project.



8. Eclipse Le debugger

➤ Les pré-requis


- Paramètres utilisateurs Développeurs Eclipse :
- Se trouvent dans « Fichier > Users Parameters », ainsi que sur cliques de l'icone 
- Les paramètres de cette fenêtre font tous référence aux paramètres utilisateurs X3 chapitre « Superviseur » groupe « DEV » du dossier connecté.

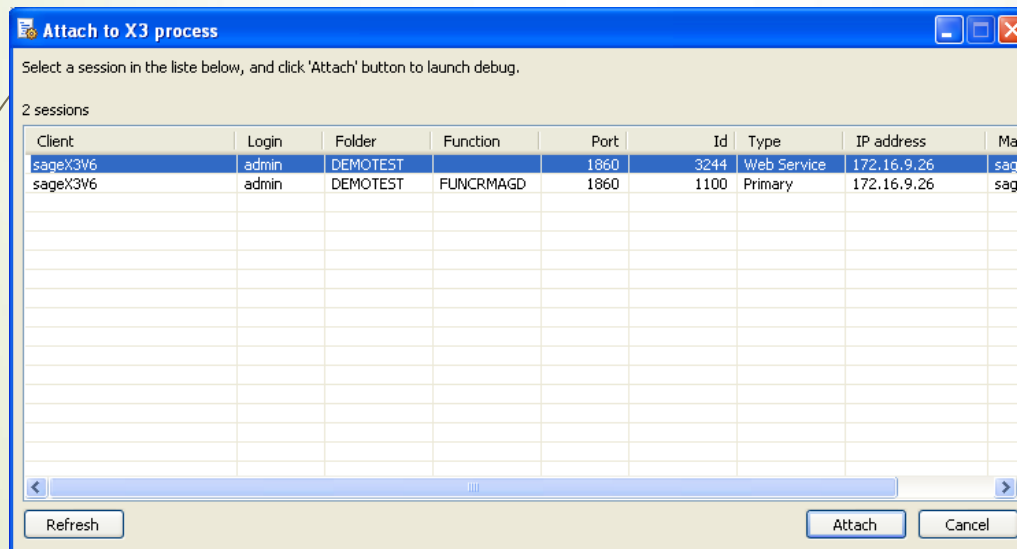


8.Eclipse Le debugger

- Les paramètres
 - AECLIPSE : Active les tunnels vers Eclipse
 - AECLIDBG : Active le debugger Eclipse
 - AECLIMAC : Permet de définir l'adresse de la machine où Eclipse est installé. Par défaut localhost.
 - AECLIPRT : Port de connexion vers le debugger Eclipse. Par défaut ou si égal à 0 alors 10000 est utilisé.

8.Eclipse Le debugger

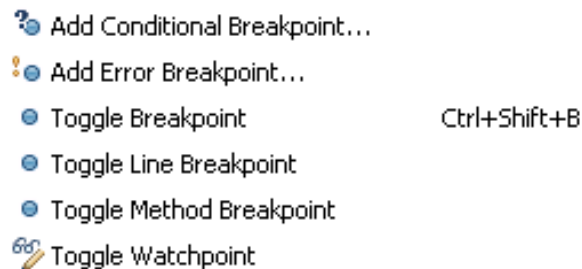
- Passer en mode debug
 - Ouvrez la perspective « Debug »
 - Dbgaff
 - Attachement à un processus en cours d'exécution
 - Cliquez sur icône  ou menu « Project>Attach to process »



Sélectionnez le processus puis cliquez sur « Attach »

8.Eclipse Le debugger

- Passer en mode debug
 - Breakpoints, sur ligne avec ou sans condition
 - Ctrl+Shift+B
 - Menu Run>Toggle Breakpoint
 - Double clic dans la marge
 - Break sur condition « Add Conditional Breakpoint... »
 - Break sur erreur Menu « Add Error Breakpoint »
 - Break sur changement de valeur « Toggle Watchpoint »



9.Actions sur champs

Exercice Continuation

- Annexe langage sur les actions sur champs.
- Exercice N°3

Fin de l'initiation
au langage

