

Development file

Data Acquisition System of equipment

6/07/2015



Application created by .

tel : . fax : .

Part 1

Heading

Part 1 Heading

ECOLOGISTICS

Implementation document

Service SV_ECOLOG

Service of equipment data
acquisition

Part 2

Project

Part 2 Project

Project

General Information

SV_Ecolog	
Repertory	C:\Mes Projets\SV_Ecolog
Associated analysis	C:\Mes Projets\SV_Ecolog\SV_Ecolog.ana\SV_Ecolog.wda

Gabarit	190 Gensteel
---------	--------------

Project

List of project's elements

	Name	Wording
	FEN_test	
	COL_ProcéduresGlobales	
Statistics		

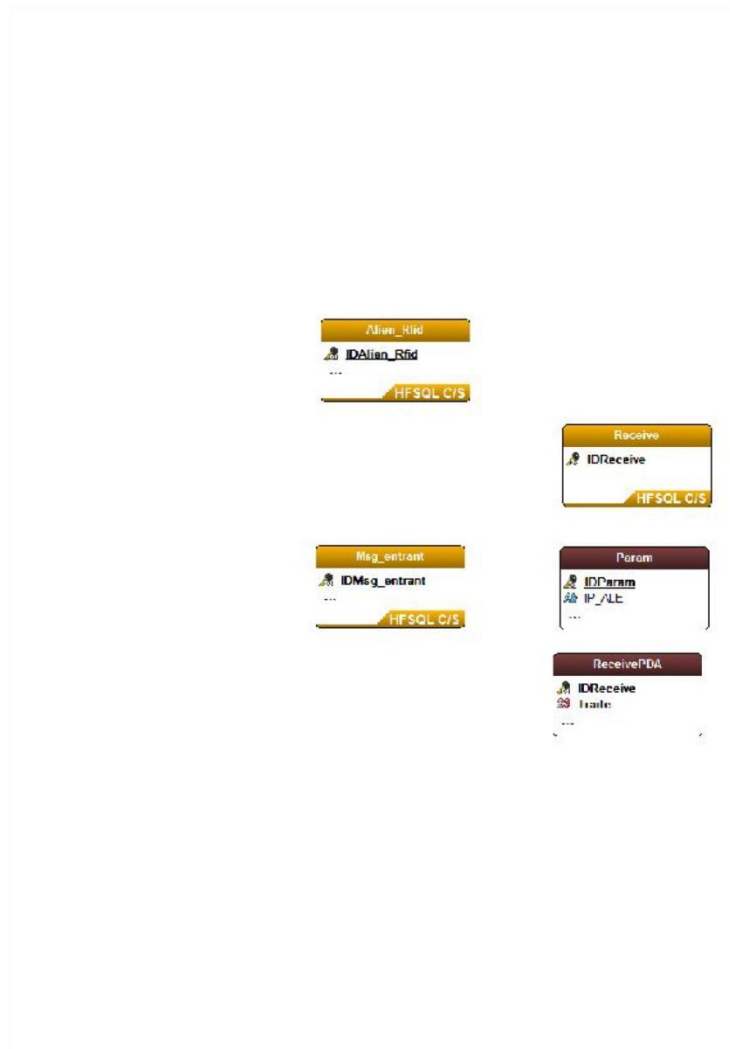
Number of windows :	1
Number of collections of procedures :	1

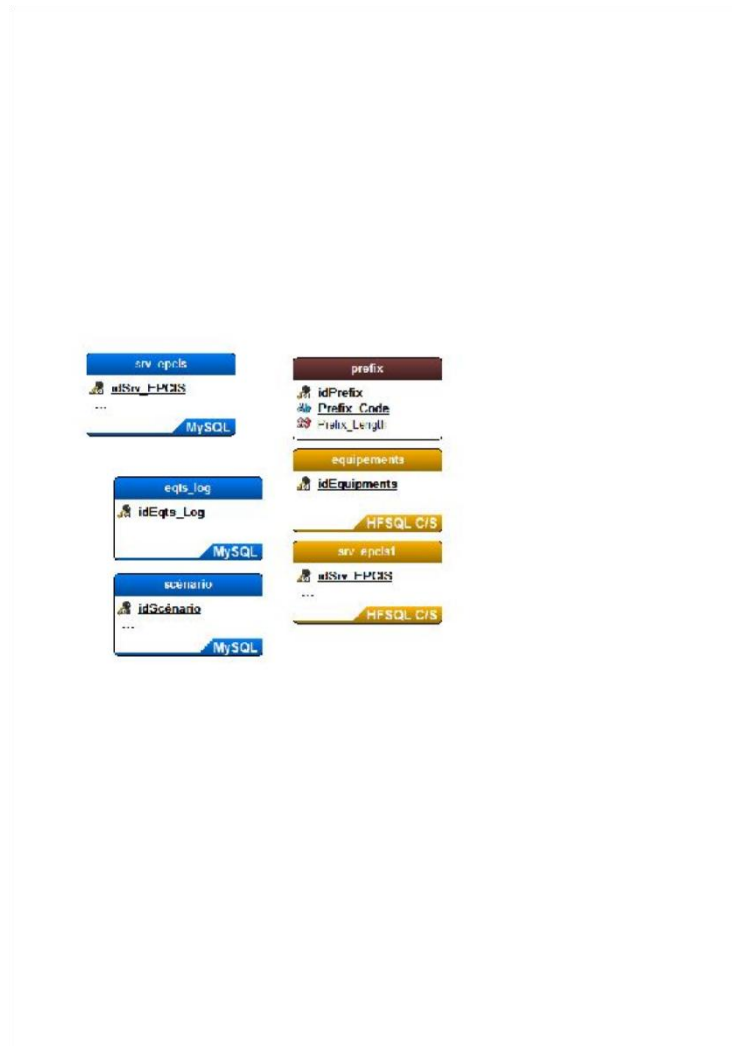
Part 3

Analysis

Analyse

Graphe





Analysis

General Information

SV_Ecolog.wda	
Repertory	C:\Mes Projets\SV_Ecolog\SV_Ecolog.ana\
Repertory of data files	Répertoire de l'application

N°generation	Nb. files	Nb. rubrics	Nb. links	Nb. connections	Nb. groups
25	11	57	0	3	0

Analysis

Dictionary of rubrics

Rubric	Type	Size	Key	Key	Used by
Action	Chaîne unicode	45			equipements
Autre_Data	Chaîne	100			Receive
	Chaîne	100			ReceivePDA
BiszStepID	Entier sur 4 octets				equipements
BizDispositionID	Entier sur 4 octets				equipements
BizDispositionName	Chaîne unicode	45			equipements
BizLobGLN					<Inutilisée>
BizLocGLN	Chaîne unicode	100			equipements
BizLocID	Chaîne unicode	100			equipements
BizLocName	Chaîne unicode	100			equipements
BizStepName	Chaîne unicode	45			equipements
CleComp1	Clé composée	8		○	equipements
Comment	Image (mémo binaire)				scénario
Contenu	Mémo texte				Msg_entrant

idSrv_EPCIS

IP_Address

Rubric	Type	Size	Key	Key	Used by...
Contenut					<Inutilisée>
Data_In	Mémo texte				Receive
	Chaîne	50			ReceivePDA
Data_Out	Mémo texte				Receive
	Chaîne	50			ReceivePDA
DATE	Date et Heure			○	Alien_Rfid
dns	Chaîne unicode	100		○	srv_epcis
	Chaîne unicode	100		○	srv_epcis1
EPCIS_URL_Capture	Chaîne unicode	100			equipements
Equipement_Type	Entier sur 4 octets			○	equipements
GPS	Chaîne	50			Receive
	Chaîne	50			ReceivePDA
IDAlien_Rfid	Identifiant automatique (4 octets)		○		Alien_Rfid
idEPCIS	Entier sur 4 octets			○	eqts_log
	Entier sur 4 octets			○	equipements
idEqts_Log	Identifiant automatique (4 octets)		○		eqts_log
idEquipement	Entier sur 4 octets			○	Receive
	Entier sur 4 octets			○	ReceivePDA
idEquipments	Entier sur 4 octets			○	eqts_log
	Identifiant automatique (4 octets)		○		equipements
IDMsg_entrant	Identifiant automatique (8 octets)		○		Msg_entrant
IDParam	Identifiant automatique (8 octets)		○		Param
idPrefix	Identifiant automatique (4 octets)		○		prefix
IDReceive					

Rubric	Type	Size	Key	Key	Used by...
IDReceive	Identifiant automatique (8 octets)		○		Receive
	Identifiant automatique (8 octets)		○		ReceivePDA
	IdScénario				<Inutilisée>
	Entier sur 4 octets			○	Receive
	Entier sur 4 octets			○	eqts_log
	Identifiant automatique (4 octets)		○		scénario
	Entier sur 4 octets			○	ReceivePDA
	Entier sur 4 octets			○	equipements
	Identifiant automatique (4 octets)		○		srv_epcis
	Identifiant automatique (4 octets)		○		srv_epcis1
	Chaîne unicode	45		○	srv_epcis
IP_Adresse	Chaîne unicode	45		○	equipements
	Chaîne unicode	45		○	srv_epcis1
XML					

	Chaîne	50		<input type="radio"/>	Alien_Rfid
IP_ALE	Chaîne	50		<input type="radio"/>	Receive
Mode	Chaîne	50		<input type="radio"/>	ReceivePDA
	Chaîne	50			Param
Model	Entier non signé sur 1 octet				Receive
Name	Entier non signé sur 1 octet				ReceivePDA
Nr_Antennas	Chaîne unicode	45			equipements
NR_READPOINT	Chaîne unicode	45			scénario
NR_SCENARIO	Entier sur 4 octets				equipements
NumFabr	Entier sur 4 octets				Param
Port_Nr	Entier sur 4 octets				Param
Prefix_Code	Chaîne	50		<input type="radio"/>	Alien_Rfid
Prefix_Length	Entier sur 4 octets				equipements
RaedPointGLN	Chaîne unicode	45		<input type="radio"/>	prefix
ReadPointID	Entier sur 4 octets				prefix
ReadPointName	Chaîne unicode	100			equipements
Rfid_Attenuation	Entier sur 4 octets				equipements
TimeStamp	Chaîne unicode	100			equipements
Timing	Entier sur 4 octets				equipements
TimèStamp	Date et Heure			<input type="radio"/>	eqts_log
	Entier sur 4 octets				equipements
Traite	Date et Heure				Receive
	Date et Heure				ReceivePDA
	Booléen			<input type="radio"/>	Msg_entrant
Triggers mode	Booléen			<input type="radio"/>	Alien_Rfid
URL_EPCIS	Booléen			<input type="radio"/>	Receive
	Booléen			<input type="radio"/>	ReceivePDA
	Entier sur 4 octets				equipements
	Chaîne	50			Receive
	Chaîne	50			ReceivePDA
	Image (mémo binaire)				eqts_log

Analysis

Connections

Connection		Data sources	User	Provider	Tm.	Tm.
CnxEcolog		127.0.0.1:4900	admin	WinDevClientServeurHF	30	30
CnxEcolog						
Utilisée par...	Msg_entrant	Alien_Rfid		Receive		
	equipements	srv_epcis1				
MySQL_Ecolog_workbench		127.0.0.1	root	WinDevMySQL	30	30
Connexion à '127.0.0.1'						
Utilisée par...	eqts_log	scénario		srv_epcis		
Interop		127.0.0.1	root	WinDevMySQL	30	30
Connexion à '127.0.0.1'						

Tm. Conn*: Timeout de connexion

Tm. Exec*: Timeout d'exécution

Analysis

Files and rubrics

Abrévi	Id.	N°gén	Taille	+ de	Espac	Répliq	Jnl.	Jnl.	Sécuri	Crypt.	Crypt.	Crypt.	Compr	Type
Abrévi	Id.	N°gén	Taille	+ de	Espac	Répliq	Jnl.	Jnl.	Sécuri	Crypt.	Crypt.	Crypt.	Compr	Type

Alien_Rfid		○	5	124										HFSQL Client/Serveur
eqts_log		○	1	32										Accès natif / Autre accès OLE DB

equipements		<input type="radio"/>	8	1732										HFSQL Client/Serveur
Msg_entrant		<input type="radio"/>	2	26	<input type="radio"/>					<input type="radio"/>	<input type="radio"/>	<input type="radio"/>		HFSQL Client/Serveur
Param		<input type="radio"/>	1	76	<input type="radio"/>					<input type="radio"/>	<input type="radio"/>	<input type="radio"/>		HFSQL Classic
prefix		<input type="radio"/>	1	109									<input type="radio"/>	HFSQL Classic
Receive		<input type="radio"/>	4	306	<input type="radio"/>					<input type="radio"/>	<input type="radio"/>	<input type="radio"/>		HFSQL Client/Serveur
ReceivePDA		<input type="radio"/>	4	392	<input type="radio"/>					<input type="radio"/>	<input type="radio"/>	<input type="radio"/>		HFSQL Classic
scénario		<input type="radio"/>	1	104										Accès natif / Autre accès OLE DB
srv_epcis		<input type="radio"/>	2	298										Accès natif / Autre accès OLE DB
srv_epcis1		<input type="radio"/>	3	307										HFSQL Client/Serveur

Espaces *: Complétion des chaînes par des espaces
Jnl. Lect/Ecr *: Journalisation des lectures et écritures

Jnl. Ecr *: Journalisation des écritures
Sécurité *: Mode sécurité renforcée

Alien_Rfid

Files and rubrics

General Information

Alien_Rfid	Alien_Rfid
Nom sur disque	Alien_Rfid.FIC
Connexion	CnxEcolog

Rubrics of Alien_Rfid file

	Wording	Type	Size	Key	Key	Way	Default val.
IDAlien_Rfid	Identifiant de Alien_Rfid	Identifiant automatique (4 octets)		<input type="radio"/>		<input checked="" type="checkbox"/>	

NumFabr	Numfabr	Chaîne	50	○	■		
DATE	DATE	Date et Heure (aaaammjjhhmmssccc)			○	■	0000000000000
IP_Adresse	Ip_adresse	Chaîne	50		○	■	
Traite	Traite	Booléen			○	■	0

eqts_log

Files and rubrics

General Information

eqts_log	eqts_log (partagé)
-----------------	--------------------

Connexion	Connexion à '127.0.0.1'
------------------	-------------------------

Rubrics of eqts_log file

	Wording	Type	Size	Key	Key	Way	Default val.
idEqts_Log	idEqts_Log	Identifiant automatique (4 octets)		○		■	
XML	XML	Image (mémo binaire)					
TimeStamp	TimeStamp	Date et Heure (aaaammjjhhmmssccc)			○	■	
idEquipments	idEquipments	Entier sur 4 octets			○	■	
idEPCIS	idEPCIS	Entier sur 4 octets			○	■	
idScenario	idScenario	Entier sur 4 octets			○	■	

equipments

Files and rubrics

General Information

equipments	equipements
-------------------	-------------

Name on disc	equipements.fic
---------------------	-----------------

Connection	CnxEcolog
-------------------	-----------

Rubrics of equipments file

	Wording	Type	Size	Key	Key	Way	Default val.
idEquipments	idEquipments	Identifiant automatique (4 octets)		○		■	
idScenario	idScenario	Entier sur 4 octets			○	■	
idEPCIS	idEPCIS	Entier sur 4 octets			○	■	

Libellé		Type	Taille	Clé	Clé	Sens	Val. défaut
BizLocName	BizLocName	Chaîne unicode	100				
BizLocID	BizLocID	Chaîne unicode	100				
BizLocGLN	BizLocGLN	Chaîne unicode	100				
ReadPointName	ReadPointName	Chaîne unicode	100				
ReadPointID	ReadPointID	Entier sur 4 octets					
RaedPointGLN	RaedPointGLN	Chaîne unicode	100				
Equipement_Type	Equipement_Type	Entier sur 4 octets			○	■	
Model	Model	Chaîne unicode	45				
IP_Address	IP_Address	Chaîne unicode	45		○	■	
Port_Nr	Port_Nr	Entier sur 4 octets					
Nr_Antennas	Nr_Antennas	Entier sur 4 octets					
Triggers mode	Triggers mode	Entier sur 4 octets					
Timing	Timing	Entier sur 4 octets					
Action	Action	Chaîne unicode	45				
BiszStepID	BiszStepID	Entier sur 4 octets					
BizStepName	BizStepName	Chaîne unicode	45				

BizDipositionID	BizDipositionID	Entier sur 4 octets					
BizDispositionName	BizDispositionName	Chaîne unicode	45				
EPCIS_URL_Capture	EPCIS_URL_Capture	Chaîne unicode	100				
Rfid_Attenuation	Rfid_Attenuation	Entier sur 4 octets					
CleComp1	CleComp1	Clé composée : idScenario+ReadPointID	8		○	■	

Msg_entrant

Files and rubrics

General Information

Msg_entrant	Msg_entrant
Nom sur disque	Msg_entrant.fic
Connexion	CnxEcolog

Rubriques du fichier Msg_entrant

	Wording	Type	Size	Key	Key	Way	Default val.
IDMsg_entrant	Identifiant de Msg_entrant	Identifiant automatique (8 octets)		<input type="radio"/>		<input checked="" type="checkbox"/>	
Contenu	Contenu	Mémo texte					
Traite	Traite	Booléen			<input type="radio"/>	<input checked="" type="checkbox"/>	0

Param

Files and rubrics

General Information

Param Param

Name on disc Param.fic

Rubrics of Param file

	Wording	Type	Size	Key	Key	Way	Default val.
IDParam	Identifiant de Param	Identifiant automatique (8 octets)		○		■	
IP_ALE	Ip_ale	Chaîne	50				
NR_SCENARIO	Nr_scenario	Entier sur 4 octets					0
NR_READPOINT	Nr_readpoint	Entier sur 4 octets					0

prefix

Files and rubrics

General Information

prefix | prefix (importé)

Nom on disc | prefix.FIC

Rubrics on prefix file

	Wording	Type	Size	Key	Key	Way	Default val.
idPrefix	idPrefix	Identifiant automatique (4 octets)		<input type="radio"/>		<input checked="" type="checkbox"/>	
Prefix_Code	Prefix_Code	Chaîne unicode	45	<input type="radio"/>		<input checked="" type="checkbox"/>	
Prefix_Length	Prefix_Length	Entier sur 4 octets				<input type="checkbox"/>	

Receive

Files and rubrics

General Information

Receive	Receive
Name on disc	Reeceive.fic
Connection	CnxEcolog

Rubrics of Receive file

	Wording	Type	Size	Key	Key	Way	Default val.
IDReceive	Identifiant de Receive	Identifiant automatique (8 octets)		<input type="radio"/>		<input checked="" type="checkbox"/>	
Mode	Mode	Entier non signé sur 1 octet					0
Data_In	Data_in	Mémo texte					
Data_Out	Data_out	Mémo texte					
TimùeStamp	Timùestamp	Date et Heure (aaaammjjhhmmssccc)					0000000000000
Traite	Traite	Booléen			<input type="radio"/>	<input checked="" type="checkbox"/>	0
IdScénario	Idscénario	Entier sur 4 octets			<input type="radio"/>	<input checked="" type="checkbox"/>	0
idEquipement	Idequipement	Entier sur 4 octets			<input type="radio"/>	<input checked="" type="checkbox"/>	0
GPS	Gps	Chaîne	50				
Autre_Data	Autre_data	Chaîne	100				
IP_Adresse	Ip_adresse	Chaîne	50		<input type="radio"/>	<input checked="" type="checkbox"/>	
URL_EPCIS	Url_epcis	Chaîne	50				

ReceivePDA

Files and rubrics

General Information

ReceivePDA	ReceivePDA
Name on disc	ReceivePDA.fic

Rubrics of ReceivePDA file

	Wording	Type	Size	Key	Key	Way	Default val.
IDReceive	Identifiant de Receive	Identifiant automatique (8 octets)		<input type="radio"/>		<input checked="" type="checkbox"/>	
Mode	Mode	Entier non signé sur 1 octet					0
Data_In	Data_in	Chaîne	50				
Data_Out	Data_out	Chaîne	50				
TimeStamp	Timestamp	Date et Heure (aaaammjjhhmmssccc)					
Traite	Traite	Booléen			<input type="radio"/>	<input checked="" type="checkbox"/>	0
IdScénario	Idscénario	Entier sur 4 octets			<input type="radio"/>	<input checked="" type="checkbox"/>	0
idEquipement	Idequipement	Entier sur 4 octets			<input type="radio"/>	<input checked="" type="checkbox"/>	0
GPS	Gps	Chaîne	50				
Autre_Data	Autre_data	Chaîne	100				
IP_Adresse	Ip_adresse	Chaîne	50		<input type="radio"/>	<input checked="" type="checkbox"/>	
URL_EPCIS	Url_epcis	Chaîne	50				

scenario

Files and rubrics

General information

scenario	scénario (partagé)
Connection	Connexion à '127.0.0.1'

Rubrics of scenario file

	Wording	Type	Size	Key	Key	Way	Default val.
idScénario	idScénario	Identifiant automatique (4 octets)		<input type="radio"/>		<input checked="" type="checkbox"/>	
Name	Name	Chaîne unicode	45				
Comment	Comment	Image (mémo binaire)					

srv_epcis

Files and rubrics

General Information

srv_epcis | srv_epcis (partagé)

Connection | Connexion à '127.0.0.1'

Rubrics of srv_epcis file

	Wording	Type	Size	Key	Key	Way	Default val.
idSrv_EPCIS	idSrv_EPCIS	Identifiant automatique (4 octets)		<input type="radio"/>		<input checked="" type="checkbox"/>	
IP_Address	IP_Address	Chaîne unicode	45		<input type="radio"/>	<input checked="" type="checkbox"/>	
dns	dns	Chaîne unicode	100		<input type="radio"/>	<input checked="" type="checkbox"/>	

srv_epcis1

Files and rubrics

General Information

srv_epcis1	srv_epcis (partagé)
Nom sur disque	srv_epcis1.fic
Connexion	CnxEcolog

Rubrics of srv_epcis1 file

	Wording	Type	Size	Key	Key	Way	Default val.
idSrv_EPCIS	idSrv_EPCIS	Identifiant automatique (4 octets)		<input type="radio"/>		<input checked="" type="checkbox"/>	
IP_Address	IP_Address	Chaîne unicode	45		<input type="radio"/>	<input checked="" type="checkbox"/>	
dns	dns	Chaîne unicode	100		<input type="radio"/>	<input checked="" type="checkbox"/>	

Part 4

WinDev window

Largeur	800
---------	-----

FEN_test

General Information

FEN_test	
----------	--

Physical file	C:\Mes Projets\SV_Ecolog\FEN_test.wdw
Date of creation	13/03/2015 11:01:28
Date of the latest modifictaion	7/04/2015 07:51:57
Title	FR: test
Type of window	Free window
Open Mode	Centred compared to the mother

Note for the file

FR:
FR:

Position X	0
Hauteur	600

Position Y	0
------------	---

Visible	<input type="radio"/>	Barre de titre	<input type="radio"/>	Menu système	<input type="radio"/>
Icône réduction	<input type="radio"/>	Icône agrandissement	<input type="radio"/>	Icône aide ctx.	
Barre de message		Détourer le fond		Mode 9 images	
Redimensionnable	<input type="radio"/>	Avec grip	<input type="radio"/>	Grip dans image	
Déplace/fond		Sauve position	<input type="radio"/>	Ascenseur auto.	
Gestion grande police	<input type="radio"/>	Menu look XP	<input type="radio"/>	Thème XP	
Ctx. HFSQL indép.					
Etat initial	Actif				
Clic droit					
Curseur survol	0				

Menu contextuel

Fichier aide ctx.

Opacité	100
Type cadre	Double
Transparence Vista	Non
Transparence gérée par Vista	Non
Valide auto.	0
Bouton auto.	
Lib. + tps restant	Oui
Groupe MDI	
Pos. X Menu	6
Pos. Y Menu	28
Mode menu ctx.	<Aucun>
M.C.U.	Oui
Anim. Ouverture	0
Anim. Fermeture	0
Durée Anim. Ouverture	100
Durée Anim. Fermeture	50

Image	
Image fond	FR:
Icône	
Image poignée	C:\Mes Projets\SV_Ecolog\Gensteel_Resize.gif

Adapte Taille contenu	Non
Assombri quand inactif	Oui
Déplace souris sur bouton par défaut	Non
Compact. barres outils	Non
Fichier parcouru	
Rub. parcourue	

Statistics of the window


	Bouton	1		
---	--------	---	--	--

Nombre d'éléments dans la fenêtre : 1

FEN_test

Information on the fields

Fenêtre : FEN_test

 Bouton

Position	Position Y	Largeur	Hauteur	Plan	Visib I	Etat initial	Touc	Type bouton
123	172	80	24	0	<input type="radio"/>	Actif	<input type="radio"/>	

BTN_SansNom1

Normal

Fenêtre : FEN_test

 Bouton : BTN_SansNom1

FR:
FR:

Libellé	FR: &Bouton
Ordre navigation	1
Curseur survol	0
On/Off	Non
Recopie Option menu flèche	Non

Note for the file

Numéro aide	0
Retrait gauche	0
Réduction Ruban	Ne pas réduire le bouton

Opacité	100						
Animée par défaut	Oui						
Mode ellipse	<Aucun>						
Type flèche	<Aucun>						
Mode menu ctx.	<Aucun>						
Nb. états image	1						
Image fond	FR: C:\Mes Projets\SV_Ecolog\Gensteel_Btn_Std.png						
Nb. états fond	5						
Etirement de l'image de fond	Largeur et hauteur						
Nb. étapes animation	6						
Largeur Halo	0						
Hauteur Halo	0						
Ancrage	Droite						
Taux anc. droite	500	Taux anc. bas	1000	Taux anc. largeur	1000	Taux anc. hauteur	1000

FEN_test

Code

Global Declarations of FEN_test

```
PROCEDURE MaFenêtre()
```

FEN_test

Fields code

Click on BTN_SansNom1

```
cd est une chaîne ="00806141411234567896"  
Traitement_Msg()
```


Part 5

Collection of procedures

COL_ProcéduresGlobales

Code

Global Procedure AI_Value__Extract

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//[ <Résultat> = ] AI_Value__Extract (<AI>, <Code128>)
// //
Paramètres :
// AI : <indiquez ici le rôle de AI> // Code128
// : <indiquez ici le rôle de Code128> // valeur
de retour :
// chaîne ANSI : // Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
//
PROCEDURE AI_Value__Extract(AI,Code128)
```

```
sValRet est une chaîne = ""
sAi est une chaîne = ai
TANTQUE Taille (sai)
<2sai ="0"+sai FIN sAi
= "("+sai+"")
```

```
c est un caractère P est un
entier bSortie est un boolean
= False
```

```
P = Position(Code128,sai)
IF P>0 THEN
  P+=4
  TANTQUE PAS bSortie=Milieu(Code128,P,1)
    IF Asc(c) >47 AND Asc(c) <(58) THEN
      sValret += c
      P++
    ELSE bSortie
      = True
  END
END
```

END

REVOYER(*sva*lret)

Global Procedure Base_to_Decimal__Convert_

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//[ <Résultat> = ] Base_to_Decimal__Convert_ (<Base>, <value>)
// //
Paramètres :
// Base : <indiquez ici le rôle de Base> //
value : <indiquez ici le rôle de value> //
valeur de retour :
// entier sur 8 octets : // Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
//
PROCEDURE Base_to_Decimal__Convert_(Base , value)
```

nvalret est un entiersur8bytes *nRes*
est un entier = *value* *nExp* est un
entier =0 *ch* est un entier *t* est un
entier = Taille(*value*)

TANTQUE *t*>0

```
  SELON Base
    CAS 2,8
      ch = NumériqueVersChaîne(Milieu(value,t,1))
    CAS 16
      SELON Majuscule(NumériqueversChaîne(Milieu(value,t,1)))
        CAS "0" ch
          = 0
        CAS "1"
          ch = 1
        CAS "2"
          ch = 2
        CAS "3"
          ch = 3
        CAS "4"
          ch = 4
```

```
CAS "5"  
  Ch = 5  
CAS "6"  
  Ch = 6  
CAS "7"  
  Ch = 7  
CAS "8"  
  Ch = 8  
CAS "9"  
  Ch = 9  
CAS "A"  
  Ch = 10  
CAS "B"  
  Ch = 11  
CAS "C"  
  Ch = 12  
CAS "D"  
  Ch = 13  
CAS "E"  
  Ch = 14  
CAS "F"  
  Ch = 15  
  
AUTRE CAS  
  
  FIN  
AUTRE CAS  
  
FIN  
  
  nvalret += ((Puissance(base,nexp)) *  
ch) t-nexp++ END  
  
REVOYER(nvalret)
```

Global Procedure Caract_Delete_

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//[ <Résultat> = ] Caract_Delete_ (<str>, <car>)
// //
Paramètres : //
str : <indiquez
ici le rôle de
str>
// car : <indiquez ici le rôle de car> //
Valeur de retour :
// chaîne ANSI : // Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
//
PROCEDURE Caract_Delete_(str,car)

svalret est une chaîne = ""
i est un entier

FOR i = 1 TO Taille(str)
  IF Milieu(str,i,1) <>car THEN
    svalret+= Milieu (str,i,1) END
END
REVOYER (svalret)
```

Global Procedure Clear_Alien

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//Clear_Alien ()
// //
Paramètres : //
Aucun
// Valeur de retour :
// Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
//// Procédure automatique :
// La procédure est exécutée automatiquement, après le code d'initialisation du projet, avec un différé de 1 seconde
// Elle sera répétée en boucle, en attendant 10 minutes entre chaque appel
```

```
// Chaque appel suivant exécute une seule fois la procédure, sans timer
//
```

```
PROCEDURE Clear_Alien()
HSupprimeTout (Alien_Rfid)
```

Global Procedure Code128_To_GRAI96__Convert

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//[ <Résultat> = ] Code128_To_GRAI96__Convert (<cb>)
// //
Paramètres :
// cb : <indiquez ici le rôle de cb> //
Valeur de retour :
// chaîne ANSI : // Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
//
PROCEDURE Code128_To_GRAI96__Convert(cb)
svalRet est une chaîne = ""

surn1 est une chaîne = "urn:epc:id:grai:" sUrn2
est une chaîne = "urn:epc:tag:grai-96:3."

sval1 sont des chaînes
tprefix est un entier = Prefix_Company__Size (Milieu(cb,6,12))

sval1 = Milieu(cb,6,tprefix)+"."+Milieu(cb,6+tprefix,5)+"."+SansEspace(Milieu(cb,12+tprefix,20))
svalRet = surn1+sval1

REVOYER(svalret)
```

Global Procedure Code128_To_SGTIN96__Convert

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
```

```

//[ <Résultat> = ] Code128_To_SGTIN96__Convert (<cb>)
// //
Paramètres :
// cb : <indiquez ici le rôle de cb> //
Valeur de retour :
// chaîne ANSI : // Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
//
PROCEDURE Code128_To_SGTIN96__Convert(cb)

```

```

svalRet est une chaîne = "" urn1 est
une chaîne = "urn:epc:id:sgtin:" urn2 est
une chaîne = "urn:epc:tag:sgtin-96:3."
nHeader est un entier = 48 nFilter
est un entier = 3 nPartition est
un entier = 5 sCompany est une
chaîne sIndicator est une chaîne
sSerial est une chaîne

```

```

sval1,sval2,sval3,sval4,sval5 sont des chaînes
p est un entier = (Position (cb,"21",17))-1
cb = "("+Gauche(cb,2)+")"+Milieu(cb,3,p-
2)+"(21)+Milieu(cb,p+3,10) sGtin est une chaîne
=AI_Value__Extract(01,cb) Tsgtin est un entier = Taille(sgtin)
sSerialNumber est une chaîne = NumériqueversChaîne(Val(AI_Value__Extract(21,cb))) sval1 =
urn1 + Milieu(sGtin,2,7)+"."+Gauche(sGtin,1)+Milieu(sGtin,9,Tsgtin-9)+"."+sSerialNumber sval2
= urn2 + Milieu(sGtin,2,7)+"."+Gauche(sGtin,1)+Milieu(sGtin,9,Tsgtin-9)+"."+sSerialNumber
scompany = Decimal_to_Base__Convert(2,Val(ExtraitChaîne(ExtraitChaîne(sval1,5,":"),1,":"),24)
sIndicator = Decimal_to_Base__Convert(2,Val(ExtraitChaîne(sval1,2,":"),20) sSerial =
Decimal_to_Base__Convert(2,Val(ExtraitChaîne(sval1,3,":"),38)
sval3 = Decimal_to_Base__Convert(2,nheader,8)+ Decimal_to_Base__Convert(2,nfilter,3)+ Decimal_to_Base__Convert(2,npartition,3)+
scompany
+ sindicator+sserial

```

```

svalret = sval1+";"+sval2+";"+sval3+";"+sval4+";"+sval5

```

```

RENVoyer (svalret)

```

Global Procedure Code128_To_SSCC96__Convert

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//[ <Résultat> = ] Code128_To_SSCC96__Convert (<cb>)
// //
Paramètres :
// cb : <indiquez ici le rôle de cb> //
Valeur de retour :
// chaîne ANSI : // Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
//
PROCEDURE Code128_To_SSCC96__Convert(cb)
    sValRet est une chaîne = ""
    sUrn1 est une chaîne = "urn:epc:id:sscc:" sUrn2
    est une chaîne = "urn:epc:tag:sscc-96:3."
    sVal1 sont des chaînes
    tprefix est un entier = Prefix_Company__Size(Milieu(cb,4,12))
    sVal1 = Milieu(cb,4,tprefix)+"."+Milieu(cb,3,1)+Milieu(cb,4+tprefix,20-(tprefix+4))
    sValRet = surn1+sval1

REVOYER(svalret)
```

Global Procedure Code_Hexa_To_SGTIN96__Convert

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//[ <Résultat> = ] Code_Hexa_To_SGTIN96__Convert (<code>)
// //
Paramètres :
// code : <indiquez ici le rôle de code> //
Valeur de retour :
// chaîne ANSI : // Aucune
// //
Exemple :
```



```
// Indiquez ici un exemple d'utilisation.
//
PROCEDURE Code_Hexa_To_SGTIN96__Convert(code)

svalRet est une chaîne = "urn:epc:id:sgtin:" svalBin
est une chaîne

svalbin = Hexa_to_Binaire__Convert(code)
svalRet += NumériqueVersChaîne(Base_to_Decimal__Convert(2,Milieu(svalbin,15,24)), "07d")
svalret += "."+NumériqueVersChaîne(Base_to_Decimal__Convert(2,Milieu(svalBin,39,20)), "06d")
svalret += "."+Base_to_Decimal__Convert_(2,Milieu(svalBin,59,38))
RENVOYER (svalRet)
```

Global of Procedure Code_Hexa_To_SSCC96__Convert

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//[ <Résultat> = ] Code_Hexa_To_SSCC96__Convert (<Code>)
// //
Paramètres :
// Code : <indiquez ici le rôle de Code> //
valeur de retour :
```

```
// chaîne ANSI : // Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
//
PROCEDURE Code_Hexa_To_SSCC96__Convert(Code)

sValRet est une chaîne = "urn:epc:id:sscc:" sValBin
est une chaîne

sValBin = Hexa_to_Binaire__Convert(code)
sValRet += NumériqueVersChaîne(Base_to_Decimal__Convert(2,Milieu(sValBin,15,24)), "07d")
sValRet += "."+Base_to_Decimal__Convert_(2,Milieu(sValBin,39,34))

REVOYER (sValRet)
```

Global Procedure FinPgm

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//FinPgm ()
// //
Paramètres :
// Aucun
// valeur de retour :
// Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
///// Traitement automatique des erreurs : RENVOYER
//

PROCEDURE FinPgm()
ExeTermine(ExeDonnePID())
```

Global Procedure Key_Eqt_Compute

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
```

```
//[ <Résultat> = ] Key_Eqt__Compute (<sc>, <eq>)  
// //  
Paramètres :  
// sc : <indiquez ici le rôle de sc> eq  
// : <indiquez ici le rôle de eq>  
// Valeur de retour :  
// chaîne ANSI : // Aucune  
// //  
Exemple :  
// Indiquez ici un exemple d'utilisation.  
//  
PROCEDURE Key_Eqt__Compute(sc,eq)
```

```
svalret est une chaîne = NumériqueVersChaîne(sc,"04d")+NumériqueVersChaîne(eq,"04d")  
REVOYER(svalret)
```

Global Procedure Lecture_Alien

```
// Résumé : <indiquez ici ce que fait la procédure>  
// Syntaxe :  
//Lecture_Alien ()  
// //  
Paramètres : //  
Aucun  
// Valeur de retour :  
// Aucune  
// //  
Exemple :  
// Indiquez ici un exemple d'utilisation.  
//// Traitement automatique des erreurs : exécuter le bloc de code CAS ERREUR  
//  
// Traitement automatique des exceptions : appeler la procédure 'FinPgm'  
// // Procédure  
automatique :  
// La procédure est exécutée automatiquement, après le code d'initialisation du projet, avec un différé de 10 secondes  
// Elle s'exécutera dans un thread (sans avoir besoin d'appeler la fonction ThreadExecute), sans utilisation de HFSQL  
// Elle sera répétée en boucle, en attendant 10 secondes entre chaque appel
```

```
//
PROCEDURE Lecture_Alien()

SectionCritiqueDébut("ALIEN")
bPrompt is boolean = False
Resultat est une chaîne
sLigne est une chaîne
sligneGen est une chaîne i
est un entier

HFilter(equipements,Equipement_Type,3)
HLitPremier (equipements,Equipement_Type)

TANTQUE PAS HEnDehors
  gsAlien_Ip=equipements.IP_Address
  gsAlien_Login="alien" gsAlien_Paswword="password"
  gsAlien_FileNameResut="ResultAlien1.txt"
  gnAlien_RF_Attenuation=equipements.Rfid_Attenuatio
  n IF Ping(gsAlien_Ip,1000) THEN

    gclMyCIsReader:InitOnNetwork (gsAlien_Ip,23)
    gclMyCIsReader:Connect()
    IF gclMyCIsReader:Login (gsAlien_Login, gsAlien_Paswword) THEN
      gclMyCIsReader:set_RFAttenuation (gnAlien_RF_Attenuation)
      Resultat = gclMyCIsReader:SendReceive("get TagList",bPrompt) END
      gclMyCIsReader:Disconnect()

  END
  i =
  1
  bRFID_Present est un boolean

  sligneGen = resultat
  HRAZ(Receive)
  sligne = ExtraitChaîne(slignegen,i,RC)
  TANTQUE SansEspace(sLigne) <>EOT i++
    bRFID_Present = False gsMsgRet=
    Milieu(sligne,5,30) gsMsgRet =
    Caract_Delete_(gsMsgRet,"") gsMsgRet
    = Caract_Delete_(gsMsgRet,"")

    HLitRecherche(Alien_Rfid,NumFabr,gsMsgRet)
    IF PAS HTrouve THEN
      IF (Val(gsMsgRet) >0) THEN HRAZ(Alien_Rfid)
        Alien_Rfid.NumFabr=gsMsgRet
```

```

    Alien_Rfid.DATE =Today()+Now()
    Alien_Rfid.Traite=1
    Alien_Rfid.IP_Adresse = gsAlien_Ip
    HAjoute(Alien_Rfid)
    //Hraz(Receive)
    Receive.Data_In+=Alien_Rfid.NumFabr+";"
//
//    Receive.TimùeStamp = Alien_Rfid.DATE
//    Receive.Mode = 2
//    receive.traite =0
//    receive.IP_Adresse= Alien_Rfid.IP_Adresse
//    Receive.URL_EPCIS= url__epcis(Alien_Rfid.IP_Adresse)
//hajoute(Receive)

    END
    END
    sligne = ExtraitChaîne(slignegen,i,RC)
FIN
Receive.TimùeStamp = Alien_Rfid.DATE
Receive.Mode = 2
Receive.Traite =0
Receive.IP_Adresse= Alien_Rfid.IP_Adresse
Receive.URL_EPCIS= URL__EPCIS(Alien_Rfid.IP_Adresse)
IF SansEspace(Receive.Data_In) <>" THEN HAjoute(Receive)
HLitSuivant(equipements,Equipement_Type)
END
HDésactiveFiltre(equipements)
SectionCritiqueFin ("ALIEN")
CAS ERREUR:

```

Global Procedure Logger

```

// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//Logger (<ind>, <txt>)
// //
Paramètres :
// ind : <indiquez ici le rôle de ind> //
txt : <indiquez ici le rôle de txt> //
valeur de retour :
// Aucune
// //
Exemple :

```

```
// Indiquez ici un exemple d'utilisation.
//
PROCEDURE Logger(ind, txt )
sligne est une chaîne

nFic est un entier = fOuvre("log.txt", foCréationSiInexistant+foLectureEcriture)
IF nFic>0 THEN
  IF ind = 1 THEN sligne = "OK" + DateVersChaîne(Today) + ""+
    HeureVersChaîne(Now, "HH:MM:SS") + txt
  ELSE sligne = "KO" + DateVersChaîne(Today) + " " + HeureVersChaîne(Now, "HH:MM:SS") +
    txt END
  fEcritLigne(nFic, sligne)
  fFerme(nFic)
END
```

Global Procedure Main_Code128_TO_EPCGen2__Convert

```
PROCEDURE Main_Code128_TO_EPCGen2__Convert(Code128)
svalRet est une chaîne = ""

IF Position(Code128, "00") = 1 THEN svalRet = Code128_To_SCC96__Convert(code128)
IF Position(Code128, "01") = 1 THEN svalRet = ExtraitChaîne(Code128_To_SGTIN96__Convert(code128), 1, ";")
IF Position(Code128, "8003") = 1 THEN svalRet = Code128_To_GRAI96__Convert(Code128)

REVOYER(svalRet)
```

Global Procedure Main_HEXa_TO_EPCGen2__Convert

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//[ <Résultat> = ] Main_HEXa_TO_EPCGen2__Convert (<CodeHexa>)
// //
Paramètres :
// CodeHexa : <indiquez ici le rôle de CodeHexa> //
Valeur de retour :
// chaîne ANSI : // Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
PROCEDURE Main_HEXa_TO_EPCGen2__Convert (CodeHexa)
svaRet est une chaîne = ""

IF Position(Codehexa,"31") >0 THEN svaRet = Code_Hexa_To_SCC96__Convert(Codehexa)
IF Position(Codehexa,"30") >0 THEN svaRet = Code_Hexa_To_SGTIN96__Convert(codehexa)

//END

REVOYER(svaRet)
```

Global procedure Prefix_Company__Size

```
// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//[ <Résultat> = ] Prefix_Company__Size (<Code>)
// //
Paramètres :
// Code : <indiquez ici le rôle de Code> //
Valeur de retour :
// entier : // Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
//
PROCEDURE Prefix_Company__Size(Code)
nvaRet est un entier = 7
bSortie est un boolean =
```

```
False sCle est une chaîne i
est un entier = Taille(code)
TANTQUE PAS bSortie AND (i>0)
  scle = Gauche(code,i)
  HLitRecherche(prefix,Prefix_Code,scle)
  IF HTrouve THEN nvalret =
    prefix.Prefix_Length
    bsortie = True
  END
  i--
FIN
REVOYER (nvalret)
```

Global Procedure Traitement_Msg


```
PROCEDURE Traitement_Msg()
i est une entier = 1
ideqts, sUrl est une
chaîne sCleIn sont des
chaîne
HDésactiveFiltre(Receive)
Hfiltre(Receive,Traite,0)
HLitPremier(Receive,Traite)
TANTQUE PAS HEndehors
  //if SansEspace(Receive.Data_Out) = "" then
  SELON Receive.Mode
    CAS 1// 128
      sCleIn = ExtraitChaîne(Receive.Data_In,1,";")
      TANTQUE Taille (sCleIn) >10
        IF Position(sCleIn,RC) >0 THEN sCleIn =
          Milieu(sCleIn,3,Taille(sCleIn)-2)
        END
        Receive.Data_Out += Main_Code128_TO_EPCGen2__Convert(sCleIn)+";"
        i ++
        sCleIn = ExtraitChaîne(Receive.Data_In,i,";")
      END
    CAS 2// HEXA
      //receive.Data_Out= Main_HEXATO_EPCGen2__Convert (receive.Data_In)
      sCleIn = ExtraitChaîne(Receive.Data_In,i,";")
      TANTQUE Taille (sCleIn) >10
        IF Position(sCleIn,RC) >0 THEN sCleIn =
          Milieu(sCleIn,3,Taille(sCleIn)-2)
        END
        Receive.Data_Out += Main_HEXATO_EPCGen2__Convert(sCleIn)+";"
        i++
        sCleIn = ExtraitChaîne(Receive.Data_In,i,";")
      END
    CAS 3// URI
      Receive.Data_Out = Receive.Data_In
    AUTRE CAS
  FIN
  Receive.Traite=1
  HModifie(Receive)
```

```

//end
ideqts = Receive.IP_Adresse
IF SansEspace(ideqts)="" THEN ideqts = Key_Eqt__Compute(Receive.IdScénario,Receive.idEquipement)
HLitRecherche(equipements,IP_Address,ideqts)
IF HTrouve THEN
    sUrl = equipements.EPCIS_URL_Capture
    IF XML__Compute(ideqts,sUrl) THEN
        Receive.URL_EPCIS = sUrl
        HModifie(Receive)
        DélaiAvantFermeture(150)
        Info("ok")
        DélaiAvantFermeture()
    END
END
HLitSuivant(Receive,Traite)
FIN
HDésactiveFiltre(Receive)

```

Global Procedure URL__EPCIS

```

// Résumé : <indiquez ici ce que fait la procédure>
// Syntaxe :
//[ <Résultat> = ] URL__EPCIS (<IP>)
// //
Paramètres :
// IP : <indiquez ici le rôle de IP> //
Valeur de retour :
// chaîne ANSI : // Aucune
// //
Exemple :
// Indiquez ici un exemple d'utilisation.
//
PROCEDURE URL__EPCIS (IP)

sValRet est une chaîne ="http://localhost:8080/epcis-repository-0.5.0/capture"

HLitRecherche(equipements,IP_Address,IP)
IF HTrouve THEN
    HLitRecherche(srv_epcis1,idSrv_EPCIS,equipements.idEPCIS)
    IF HTrouve THEN
        sValRet="http://" +srv_epcis1.IP_Address+":8080/epcis-repository-0.5.0/capture"
    END
END

```

```

END
END
RENOYER (svalRet)

```

Global Procedure XML__Compute

```

PROCEDURE XML__Compute(idEq, sUrl)

i est un entier = 1
svalue est une chaîne
sLigne est une chaîne
ideqt=""
bvalret est un boolean = True nFic
est un entier
//*****
//
sLigne = "<?xml version="+Caract(34)+"1.0"+Caract(34)+ " encoding="+Caract(34)+"UTF-8"+Caract(34)+"?>"; //FecritLigne(nFic,sLigne) sLigne
+= "<epcis:EPCISDocument xmlns:epcis="+Caract(34)+"urn:epcglobal:epcis:xsd:1"+Caract(34)+
" xmlns:xsi="+Caract(34)+"http://www.w3.org/2001/XMLSchema-instance"+Caract(34)+" creationDate="+Caract(34)+DateVersChaîne(Today, "AAAA-MM-JJ")+ "T"+
HeureVersChaîne(Now, "HH:MM:SS")+ ".016+01:00"+Caract(34)+" schemaVersion="+Caract(34)+"1.0"+Caract(34)+" xmlns:myNs="+Caract(34)+
"http://my.unique.namespace"+Caract(34)+">"; //FecritLigne(nFic,sLigne)
sLigne += "<EPCISBody>" ; //FecritLigne(nFic,sLigne) sLigne +=
"<EventList>" ; //FecritLigne(nFic,sLigne) sLigne += "<ObjectEvent>"
; //FecritLigne(nFic,sLigne)
sLigne += "<eventTime>"+DateVersChaîne(Today, "AAAA-MM-JJ")+ "T"+HeureVersChaîne(Now, "HH:MM:SS")+ "Z</eventTime>" ; //FecritLigne(nFic,sLigne)
sLigne += "<eventTimeZoneOffset>+00:00</eventTimeZoneOffset>" ; //FecritLigne(nFic,sLigne) sLigne += "<epcList>" ; //FecritLigne(nFic,sLigne)
Svalue = ExtraitChaîne(Receive.Data_Out, i, ";")
TANTQUE Taille (svalue) >20 sLigne
    += "<epc>"+svalue+"</epc>";
    i++ //FecritLigne(nFic,sLigne)
    svalue = ExtraitChaîne(Receive.Data_Out, i, ";")
END
sLigne += "</epcList>" ; //FecritLigne(nFic,sLigne)
sLigne += "<action>"+equipements.Action+"</action>"; //FecritLigne(nFic,sLigne) sLigne
+= "<bizStep>"+equipements.BizStepName+"</bizStep>"; //FecritLigne(nFic,sLigne)
sLigne += "<disposition>"+equipements.BizDispositionName+"</disposition>"; //FecritLigne(nFic,sLigne) sLigne
+= "<readPoint>"; //FecritLigne(nFic,sLigne)
sLigne += "<id>"+equipements.ReadPointName+"</id>"; //FecritLigne(nFic,sLigne)
sLigne += "</readPoint>"; //FecritLigne(nFic,sLigne) sLigne += "<bizLocation>";
//FecritLigne(nFic,sLigne)

```

```

sLigne += "<id>"+equipements.BizLocName+"</id>"; //FecritLigne(nFic,sLigne)
sLigne += "</bizLocation>"; //FecritLigne(nFic,sLigne) sLigne +=
"</ObjectEvent>"; //FecritLigne(nFic,sLigne) sLigne += "</EventList>";
//FecritLigne(nFic,sLigne) sLigne += "</EPCISBody>";
//FecritLigne(nFic,sLigne) sLigne += "</epcis:EPCISDocument>";
//fEcritLigne(nFic,sLigne)
//*****
HTTPCréeFormulaire("Form")
HTTPAjouteParamètre("Form","",sLigne)
IF HTTPEnvoieFormulaire("Form",sUrl,httpPost, "", "", "text/xml") THEN
    bvalret=1
END
IF bvalret = 1 THEN repname est une
    chaîne = ("C:\Data")
    fRepCrée(repname)
    sFilename est une chaîne = ComplèteRep(repname) + "Post.log"
    nfic = fOuvre (sFilename,foCréationSiInexistant+foLectureEcriture+foAjout)
    IF nfic>0 THEN
        fEcritLigne(nfic,sLigne
    ) fFerme(nFic) END

END
//*****

//end
REVOYER(bvalret)

```

Global Procedure XML__Post

```

PROCEDURE XML__Post(msg)
bok est un boolean = False

HTTPCréeFormulaire("Form")
HTTPAjouteParamètre("Form","",msg)
IF HTTPEnvoieFormulaire("Form",gsPathEPCIS_Server,httpPost, "", "", "text/xml")
THEN bok = True ELSE bok = False
END
REVOYER(bok)

```

Table of contents

Part 6

Table of contents

Project SV_Ecolog

3 Heading

Part 1

3 ○ Heading

29 ○ Information on the fields
31 ○ Code
32 ○ Fields Code

5 Project

Part 2

5 ○ General Information

5 ○ List of project's elements

34 Collection of procedures

Part 5

34 ○ COL_ProcéduresGlobales

34 ○ Code

7 Analysis

Part 3

7 ○ Graphic

9 ○ General information

9 ○ Dictionnary of rubrics

12 ○ Connections

12 ○ Files and rubrics

13 ○ Alien_Rfid

15 ○ eqts_log

16 ○ equipements

18 ○ Msg_entrant

19 ○ Param

20 ○ prefix

21 ○ Receive

22 ○ ReceivePDA

23 ○ scénario

24 ○ srv_epcis

25 ○ srv_epcis1

27 WinDev window

Part 4

27 ○ FEN_test

27 ○ General Information