

Développement d'extensions en version Ax

Voici un petit guide de développement d'une extension (fichier wdl) pour les versions Ax.

Les wdl de Windev sont l'équivalent des dll de Windows : ce sont des bibliothèques de ressources, contenant tout ce que peut contenir un projet windev, et pouvant être appelées par tout programme windev. Elles permettent d'ajouter des fonctionnalités infinies à Gestan, dont l'exécution peut rester conditionnelle.

Sur demande, vous pouvez obtenir les codes-source de différentes extensions.

Conditions préalables

Pour développer des extensions, un engagement de confidentialité doit avoir été signé avec ICS-Informatique.

Toute extension développée pour Gestan A1 et suivantes doit :

- être réalisée selon les [normes de développement](#);
- être archivée sur le GDS mis à disposition par ICS;
- être identifiée selon les modalités précisées par ICS;
- faire l'objet d'une documentation complète publiée sur le manuel.

Dans le cas où l'extension est payante, elle embarque le système de gestion de licences développé par ICS. Les factures de licences sont émises par le développeur et adressées directement à son client, et le n° de licence est délivré par ICS.

Actuellement, Gestan est développé sous Windev 2024, les extensions doivent aussi être développées en version 2024 ou inférieure.

Ressources

Pour développer des extensions, sont nécessaires les éléments suivants :

- L'analyse Gestan, c'est à dire la description des fichiers et des rubriques de la base de données;
- Le gabarit Gestan VAXx, afin de donner une unité de design;
- Le set d'icônes utilisables dans Gestan, pour l'unité de design.

Vous trouverez ces ressources sur la [page dédiée](#).

Création du projet

Créez un projet de type bibliothèque.

La bibliothèque doit se nommer GESTAN_[nom wdl].wdl, par exemple GESTAN_PROSTEP.wdl.

Langues du projet

Comme Gestan, les extensions sont traduites en deux langues : Espagnol et Anglais (dans cet ordre).

Description du projet

Langues du projet

Langue	Nombre	Monétaire	Heure	Date	Durée	Divers
Français 1	☒ Utiliser les paramètres définis dans les options linguistiques du système d'exploitation					
Espagnol	☐ Utiliser les paramètres suivants :					
Anglais	Séparateur de décimales : ,					
	Séparateur de milliers :					
	Format négatif : -1.1					
	Rétablir les valeurs par défaut					

Exemples

Exemple positif : 12 354,26

Exemple négatif : -12 354,26

☐ Langue par défaut en exécution

Pour les éléments actuellement ouverts sous l'éditeur, les modifications effectuées sur les langues du projet ne seront prises en compte que lors de la prochaine ouverture de ces éléments.

☐ Sélectionner automatiquement la langue du système au démarrage de l'application

Code d'initialisation du projet (de la wdl)

Allez dans le code d'initialisation de projet de la WDL, faites un copier-coller du code ci dessous et apportez-y les adaptations nécessaires pour votre projet.

Code d'initialisation de la wdl

SI EnModeTest() ALORS

```

    // Déclarez en externe les fichiers de Gestan que vous allez utiliser
    (si vous n'avez pas copié l'analyse)
    EXTERNE PRODUIT, MACHINE, JOURNALAPP, LIBTRADUIT, CONTACT, PARAMLIB,
    CNX, PROJET, CONSO, FACTURE, TACHE, ADRESSEPOST, gf_Renvoyer_AuthKey

    // Déclarez en externe les fonctions de Gestan que vous allez appeler
    dans votre code
    EXTERNE gf_USER_AlimenterListe
    // Les variables qui servent à Gestan, et qui sont utilisées aussi dans
    l'extension
    gCdUser                                est une chaîne
    // Code de l'utilisateur connecté
    gbAdmin                                est un entier
    // 0 = user simple, 1= admin gestan
    gIniFile                                est une chaîne
    // Nom complet du fichier de paramétrage
    gOptFile                                est une chaîne
    // Nom complet du fichier des options de paramétrage pour la base
    // Fichier de paramétrage spécifique à Gestan Cloud
    gCloudFile                                est une chaîne      =
    ComplèteRep(SysRep(srAppDataCommun))+ "GESTAN\gestan_cloud.ini"
    gRepGestan                                est une chaîne
    // Chemin du répertoire Gestan
    gDHConnect                                est une chaîne      =
    DateHeureSys()    // Date Heure de connexion
    gNumVersion                                est un entier
    // Numéro de version de Gestan (2=Standard ou 3=Pro)
    // N° de version de l'exécutable (x.15.0A.00)
    gVersionExe                                est une chaîne      =
    EnModeTest() ? "x.15.08" SINON ExeInfo(exeVersion)[[À 7]]
    gNumRev                                est un entier      =
    EnModeTest() ? "MT" SINON ExtraitChaîne(ExeInfo(exeVersion),4,".")
    gbPatch                                est un booléen
    gSupport                                est un entier
    // 0=rien, 1=ICS, 2=Partenaires
    gNbDecEcr                                est un entier
    gNbDecQteProd, gNbDecTVA, gNbDecPrxProd    sont des entiers
    gNbHeuresParJournee                        est un réel
    gLST_Fichiers_Externes                    est une chaîne
    gCléRegistre                                est une chaîne
    gbBibChargée_COMPTA                        est un booléen
    gNBUnitésBib                                est un entier      = 20
    // Nombre d'unités de bibliothèques
    gVersionWindev                                est une chaîne      = "29.0"
    // A changer si la version de Windev change (29 = Windev 2024)
    gDestinationPDF                            est un entier
    gNumDatabase                                est un entier sur 8
    // Numéro de la database connectée
    xxNumPostit                                est un entier sur 8

```

```
// Pour le dragndrop des postit vers l'agenda
  gPriv                                     est un tableau associatif
d'entiers
  gbJournalConnexion                       est un booléen
  gbEffacerSaint                           est un booléen
  gbNoTimeStamp                           est un booléen
// Pour bypasser la mise à jour de USMOD & DHMOD
  gModeConnect                             est un entier
  gbEnCloud                               est un booléen
// Vrai si on est en cloud
  gCouleurAppli                           est un tableau de 3 entiers
  tablo_LICENCE                           est un tableau associatif
d'entiers
  gCodeAdminCloud                         est une chaîne
  gRefExt                                 est une chaîne
// Pour utilisation de wdl externes
  gDescToolBar                             est une chaîne
// Pour l'utilisation de toolbars en provenance de wdl externes
  gSysRepMesDocs                           est une chaîne
// Répertoire MesDocuments (U en Cloud)
  gbNoDéconnect                           est un booléen
// pour ne pas déconnecter l'instance qui a demandé une déconnexion générale
  gCodeExtension                           est une chaîne
  gbRuban                                 est un booléen
  gNbBoutonsMétier                         est un entier
DateHeureParDéfaut("19000101","0000")
// Constantes diverses
CONSTANTE
  COUL_Bleu                               = 14249279
  COUL_BleuClair                           = 16250347
  COUL_BleuCredir                           = 14526242 // rvb(34 167 221)
  Coul_BleuLigne                           = 16438176
  COUL_Brun                                = 5014704 // RVB(176,132,76)
  COUL_GrisClair                           = 16314090
  COUL_GrisMoyen                           = 12632256
  COUL_Jaune                               = 10092543
  COUL_Kaki                                = 10070188
  COUL_OrangeClair                         = 9884415
  COUL_OrangeFoncé                         = 37119
  COUL_Rouge                               = 7643
  COUL_RougeClair                          = 7434721
  COUL_Vert                                = 905472
  COUL_VertClair                           = 4122527 // RVB(159,231,62) anc :1687808
  COUL_VertFoncé                           = 1344030 // 1994280
  COUL_Rose                                = 16764159
  COUL_OrangeLeger                         = 13434879
  COUL_LivingCoral                         = 7368191
  COUL_VertFerie                           = 11337689
  COUL_BleuWeekEnd                         = 16314087
// Couleurs de Windows 8 (utilisées dans les widgets, notamment)
  COUL8_bleuClair                         = 14852379 // rvb(27,161,226)
```

```
COUL8_bleuMoyen      = 15697965 // 45,136,239
COUL8_bleu           = 12473610  // 10,85,190
COUL8_vert           = 40960 // 0,160,0
COUL8_violine        = 11141283 // 163,0,170
COUL8_violet         = 11810905 // 89 56 180
COUL8_orange         = 2574803 // 211,73,39
COUL_ToastOK         = VertClair
COUL_ToastKO         = RougeClair
COUL_OK              = 65280
COUL_KO              = 255
MSG_RIEN_IMPRIMER    = 1
MSG_SUPPRIMER        = 2
MSG_ERRINATTENDUE    = 3
MSG_FICHIERVERIDE    = 4
MSG_PARAMKO          = 5
MSG_DUPLIQUER        = 6
MSG_APPELINCORRECT   = 7
MSG_OPSIMPOSSIBLE    = 8
MSG_OPERATIONADMIN   = 9
MSG_SUPPRIMER_SELECTION = 10
MSG_PBFHSQL          = 11
MSG_DROITSWINDOWS    = 12
MSG_PRIVILINSUFF     = 13
MSG_RIENAAFF         = 14
MSG_VERSIONENTREPRISE = 15
MODE_CLASSIC         = 0
MODE_CS_LAN          = 1
MODE_CS_WAN          = 2
GPSSWD = "*****"
```

FIN

// Déclaration éventuelle de fichiers présents dans l'analyse de Gestan
mais pas dans celle de l'extension

LOCAL

NomAna est une chaîne="E:\Mes
Projets\GESTAN_15\Gestan.ana\Gestan.wdd"

```
HDéclare("PAYS",NomAna)
HDéclare("PARAMLIB",NomAna)
HDéclare("CONTACT",NomAna)
HDéclare("ADRESSEPOST",NomAna)
```

// Déclaration mots de passe des fichiers utilisés par l'extension

```
HPasse("*",GPSSWD)
HPasse(PARAMAPPLI,GPSSWD)
HPasse(USER,GPSSWD)
HPasse(PARAMPLUS,GPSSWD)
HPasse(CIVILITE,GPSSWD)
HPasse("PAYS",GPSSWD)
HPasse("PARAMLIB",GPSSWD)
HPasse("CONTACT",GPSSWD)
HPasse("ADRESSEPOST",GPSSWD)
```

```

// Si on a besoin de CNX
HDéclareExterne(gf_Renvoyer_Répertoire("ADD")+"CNX.fic","CNX",GPSSWD)

// Initialisation du répertoire et du code user, pour l'usage en mode
test
gRepGestan="C:\GESTAN_15\TEST\"
gCdUser ="EN"
HFerme(" *")
HSubstRep("?",gRepGestan+"DATABASE")
// Déclaration des triggers sur les fichiers de l'extension et les
fichiers externes
HDécritTrigger(Z43_PROSPECT,"HAJOUTE,HMODIFIE,HSUPPRIME","Horodate",hTrigger
Avant)
HDécritTrigger(EMPLACEMENT,"HAJOUTE,HMODIFIE,HSUPPRIME","Horodate",hTriggerA
vant)
HDécritTrigger(CONTACT,"HAJOUTE,HMODIFIE,HSUPPRIME","Horodate",hTriggerAvant
)
//[etc...]

FIN

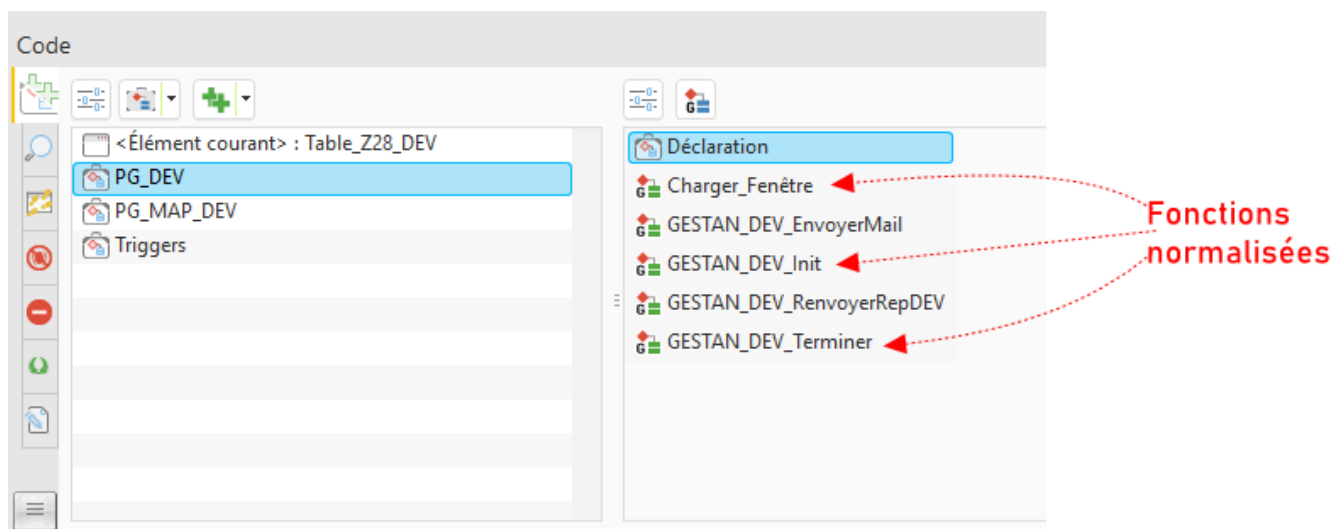
```

Procédures globales (obligatoires)

Dans le projet, créez tout d'abord une collection PG_[nom wdl], contenant trois procédures globales.

- Charger_Fenêtre : pour charger les fenêtres
- [nom wdl]_Init : procédure d'initialisation de l'extension
- [nom wdl]_Terminer : procédure de terminaison de l'extension

Nous allons voir le contenu de ces fonctions ci-dessous.



Dans cet exemple, PG_DEV contient Charger_Fenêtre, GESTAN_DEV_Init, et GESTAN_DEV_Terminer

Vous pouvez aussi mettre dans cette collection les autres procédures globales qui seraient propres à cette extension. Nommez-les [nom wdl]_[nom procédure], comme par exemple GESTAN_DEV_EnvoyerMail ou GESTAN_DEV_RenvoyerRepDev (cela évite les conflits de noms de procédure avec d'éventuelles autres extensions Gestan).

Si vous souhaitez copier-coller des fonctions globales de Gestan, mettez-les dans une collection de procédure PG_MAP_[nom wdl], par exemple PG_MAP_PROSTEP (MAP pour "Mise au point").

Procédure ChargerFenêtre

Cette procédure est toujours identique. Copiez le code ci-dessous.

Procédure ChargerFenêtre

```
// Résumé : Ouvre une fenêtre depuis une extension
// Syntaxe :
// Charger_Fenêtre (<StrParam>)
//
// Paramètres :
// StrParam : Nom de la fenêtre, éventuellement avec des paramètres séparés
// par TAB
// Valeur de retour :
// Aucune
//
PROCÉDURE Charger_Fenêtre(StrParam)

NomFenetre  est une chaîne    = ExtraitChaîne (StrParam,1)
P1          est une chaîne    = ExtraitChaîne (StrParam,2)

SI P1 DANS ("",EOT) ALORS Ouvre (NomFenetre) SINON Ouvre (NomFenetre,P1)
```

Procédure [nom wdl]_Init

Cette procédure est exécutée au moment du chargement de la bibliothèque par Gestan.

Elle renvoie Vrai ou Faux à Gestan, selon qu'elle est exécutée correctement ou pas. Elle est appelée avec une chaîne vide en paramètre, pour pouvoir traiter le message d'erreur éventuel.

Exemple de fonction _Init (extension PROSTEP)

```
PROCÉDURE GESTAN_PROSTEP_Init(MsgErr="")

// Identification de l'extension
gCodeExtension = "ICS-
INFORMATIQUE"+TAB+"GESTAN_PROSTEP"+TAB+"43"+TAB+"1815"+TAB+"A1"+TAB...
+"PAYS|CONTACT|MSG_TYPE|ADRESSEPOST|PARAMPLUS|USER|PRODUIT|MACHINE|LIBTRADUI
T|CNX|PARAMAPPLI|CIVILITE"+TAB...
```

```
"Z43_PROCESS|Z43_PROSPECT"

// -----
// -- Fonctions normalisées - Ne pas modifier sous ce trait --
NomExt, Préfix, wVersionWDL,wVersionSCR, Lst_Fichiers sont des chaînes

// Vérification de la version de l'extension (anti-downgrade)
NomExt      = ExtraitChaîne(gCodeExtension,2)
Préfix      = ExtraitChaîne(gCodeExtension,3)
wVersionWDL = ComposantInfo ("",
ciVersion,gf_Renvoyer_Répertoire("ADD")+(NomExt+".wdl"))
gf_PARAMPLUS("L","Z"+Préfix+"_VERSION","",wVersionSCR,"")
SI wVersionSCR="" ALORS
    gf_PARAMPLUS("E","Z"+Préfix+"_VERSION","",wVersionWDL,"Version de
l'extension "+NomExt)
SINON
    SI wVersionWDL<wVersionSCR ALORS
        MsgErr=ChaîneConstruit("La version de l'extension %1 installé sur ce
poste (%2) est inférieure à la version en cours (%3). Procédez à la mise à
jour.", NomExt, wVersionWDL, wVersionSCR)
        RENVOYER Faux
    FIN
FIN

// Déclaration des fichiers spécifiques à l'extension
Lst_Fichiers = ExtraitChaîne(gCodeExtension,7)
SI Lst_Fichiers<>"" ALORS
    POUR TOUTE CHAÎNE nmFic DE Lst_Fichiers SÉPARÉE PAR "|"
        SI PAS HDéclare(nmFic,NomExt+".wdd",GPSSWD,nmFic,hWDL) ALORS
Erreur(HErrreurInfo());RENVoyer Faux
        // Assignment des fichiers, et création si inexistant
        SI HSubstRep(nmFic,ComplèteRep(gRepGestan)+"DATABASE") ALORS
            QUAND EXCEPTION DANS
                SI PAS HCréationSiInexistant(nmFic,GPSSWD) ALORS
Erreur(ChaîneConstruit("Impossible de créer %1",nmFic));RENVoyer Faux
            FAIRE
                ExceptionActive()
                SI ExceptionInfo(errCode)=70016 ALORS
                    SI PAS HModifieStructure(nmFic,hmsNormal,Null) ALORS
                        MsgErr = "Une erreur de format de fichier a été
dé détectée et n'a pas pu être réparée."+RC+HErrreurInfo()
                        RENVOYER Faux
                    SINON
                        Info(ChaîneConstruit("Un des fichiers de l'extension
%1 avait changé de structure, et a été modifié correctement. Vous pourrez
redémarrer Gestan et utiliser à nouveau l'extension.",NomExt))
                    FIN
                SINON
                    MsgErr = HErrreur()
                    RENVOYER Faux
                FIN
            FIN
        FIN
    FIN
    MsgErr = HErrreur()
    RENVOYER Faux
FIN
```

```

        FIN
    FIN
    FIN
    // Déclaration obligatoire pour la prise en compte dans l'écran de
réindexation des fichiers=
    gLST_Fichiers_Externes+=[RC]+Remplace(Lst_Fichiers,"|",RC)
    FIN
    // -- Fin des fonctions normalisées --
    // -----

// Ajout des menus de l'extension dans Gestan
QUAND EXCEPTION DANS
    // Ajout de la fenêtre spécifique
    SI gbRuban ALORS
        EXTERNE Ruban1,mnuRub_ContactPlus
        // Ajout de la fenêtre de paramétrage
MenuAjouteOption("mnuRub_Paramétrage_application.Paramétrage_Extensions","MN
U_EXT_PROSTEP_param","Étapes prospection", "Paramétrer_Addon",
"FI_PARAMPLUS_PROSTEP|Extension PROSTEP")
        // Ajout du menu spécifique (option dans le bouton StatPlus)
        Ruban1.ContactPlus..Visible=Vrai
        MenuAjouteOption("mnuRub_ContactPlus","MNU_EXT_PROSTEP","Étapes
prospection", Charger_Fenêtre,"Table_Z43_PROSPECT")
        {"MNU_EXT_PROSTEP", indChamp}..Image = "prostep_16_1.png"
        mnuRub_ContactPlus.OptAjoutContactPlus..Visible=Faux
    SINON
        // Ajout de la fenêtre de paramétrage
MenuAjouteOption("Menu._Menu.mnuOutils.Paramétrage_application.Paramétrage_a
ddons", "MNU_EXT_PROSTEP_param", "Étapes prospection", "Paramétrer_Addon",
"FI_PARAMPLUS_PROSTEP|Extension PROSTEP")
        {"MNU_EXT_PROSTEP_param", indChamp}..Image = "prostep_16_1.png"
        // Ajout du menu spécifique (option dans le menu Stats)
        MenuInsèreOption("Menu._Menu.mnuContacts",
"mnuAction_commerciale", "MNU_EXT_PROSTEP", "Étapes prospection",
Charger_Fenêtre,"Table_Z43_PROSPECT")
        {"MNU_EXT_PROSTEP", indChamp}..Image = "prostep_16_1.png"
    FIN
    FAIRE
        MsgErr=ChaîneConstruit("Erreur dans la mise en place du menu de
l'extension %1",NomExt+RC+ErreurInfo(errComplet))
        RENVOYER Faux
    FIN

// La procédure a été effectuée OK : renvoyer Vrai
RENVoyer Vrai

```

Cette fonction-type comporte trois parties :

- l'identification de l'extension

- la fonction de contrôle de version et de déclaration éventuelle des fichiers
- la fonction de mise en place des menus

Ces trois parties sont détaillées ci-dessous.

Procédure [nom wdl]_Terminer

Cette procédure est exécutée au moment du téléchargement de la bibliothèque par Gestan.

Elle renvoie Vrai ou Faux, selon qu'elle est exécutée correctement ou pas.

Elle supprime les menus qui avaient été ajoutés, annule les déclarations, et ferme les fichiers utilisés.

Ci-dessous la fonction GESTAN_PROSTEP_Terminer, de l'extension GESTAN_PROSTEP.

Procédure [nom wdl]_Terminer

```
PROCÉDURE GESTAN_PROSTEP_Terminer()

// Identification de l'extension
gCodeExtension = "ICS-
INFORMATIQUE"+TAB+"GESTAN_PROSTEP"+TAB+"43"+TAB+"0245"+TAB+"A0"+TAB...
+"PAYS|CONTACT|MSG_TYPE|ADRESSEPOST|PARAMPLUS|USER|PRODUIT|MACHINE|LIBTRADUI
T|CNX|PARAMAPPLI|CIVILITE"+TAB...
+"Z43_PROCESS|Z43_PROSPECT"

// - Suppression du menu de paramétrage
MenuSupprime("MNU_EXT_PROSTEP_param")

// Suppression du menu spécifique
SI gbRuban ALORS
    // Pour le menu spécifique, il faut effacer le bouton ContactPlus et le
    menu d'ajout d'option
    EXTERNE mnuRub_ContactPlus
    mnuRub_ContactPlus.OptAjoutContactPlus.Visible=Faux
    mnuRub_ContactPlus.Visible=Faux
FIN

MenuSupprime("MNU_EXT_PROSTEP")

// -----
// -- Fonctions normalisées - Ne pas modifier sous ce trait --
// Fermeture des fichiers et annulation des déclarations
Lst_Fichiers est une chaîne = ExtraitChaîne(gCodeExtension,7)
SI Lst_Fichiers<>"" ALORS
    POUR TOUTE CHAÎNE nmFic DE Lst_Fichiers SÉPARÉE PAR "|"
        HFerme(nmFic)
        HAnnuleDéclaration(nmFic)
    FIN
FIN
```

```
// Mise à jour de la liste des fichiers
gLST_Fichiers_Externes = Remplace(RC+gLST_Fichiers_Externes,
Remplace(Lst_Fichiers,"|",RC), "")
FIN

// Fin des fonctions normalisées

RENOYER Vrai
```

Dans cette procédure, il faut :

- recopier le code d'identification de l'extension
- coder l'effacement des menus spécifiques

Pour l'identification de l'extension, c'est simplement le code ci-dessous.

```
// Identification de l'extension
gCodeExtension = "ICS-
INFORMATIQUE"+TAB+"GESTAN_PROSTEP"+TAB+"43"+TAB+"0245"+TAB+"A0"+TAB...
+"PAYS|CONTACT|MSG_TYPE|ADRESSEPOST|PARAMPLUS|USER|PRODUIT|MACHINE|LIBTRADUI
T|CNX|PARAMAPPLI|CIVILITE"+TAB...
+"Z43_PROCESS|Z43_PROSPECT"
```

Pour la suppression du/des menus spécifique, cela dépend de la mise en place. Ci-dessous l'exemple de PROSTEP.

```
// - Suppression du menu de paramétrage
MenuSupprime("MNU_EXT_PROSTEP_param")

// Suppression du menu spécifique
SI gbRuban ALORS
    // Pour le menu spécifique, il faut effacer le bouton ContactPlus et le
    menu d'ajout d'option
    EXTERNE mnuRub_ContactPlus
    mnuRub_ContactPlus.OptAjoutContactPlus.Visible=Faux
    mnuRub_ContactPlus.Visible=Faux
FIN

MenuSupprime("MNU_EXT_PROSTEP")
```

Ecran de paramétrage

Si vous avez besoin de paramètres pour vos extensions, utilisez pour cela une fenêtre interne de dimension 777*404, qui sera appelée par l'écran standard de paramétrage de Gestan (voir ci-dessous) :

Cette fenêtre est redimensionnable : pensez à gérer les ancrages de votre fenêtre interne.

3.15.25.02

Process ▼ +

Type de gestion ▼

Modif ▼

Suppression ▼

☒ Logguer les changements d'étapes

Étapes ↔

Etape 1	Durée 1	0	Etape 7	Durée 7	0
Etape 2	Durée 2	0	Etape 8	Durée 8	0
Etape 3	Durée 3	0	Etape 9	Durée 9	0
Etape 4	Durée 4	0	Etape 10	Durée 10	0
Etape 5	Durée 5	0	Etape 11	Durée 11	0
Etape 6	Durée 6	0	Etape 12	Durée 12	0

Appliquer ✓
Valider ✓
Annuler ✕

Pour afficher le numéro de version :

NomWDL est une chaîne="GESTAN_CHRONSTAT.wdl" // Mettre ici le nom de l'extension, ne pas modifier en dessous

LaVersion est une chaîne = ComposantInfo ("",
ciVersion,gf_Renvoyer_Répertoire("ADD")+NomWDL)

LibVersion = LaVersion

LibVersion..Bulle = ChaîneConstruit("Version de l'extension :
%1",LaVersion)

SI PAS EnModeTest() ALORS

REQ est une Source de Données

MaRequete est une chaîne = [

SELECT VERSION FROM

GSTN_ADDON_AX AS NUMVERSION WHERE NOMADDON='%1'

]

MaRequete = ChaîneConstruit(MaRequete, NomWDL)

SI HExécuteRequêteSQL(REQ, hRequêteDéfaut, MaRequete) ALORS

SI REQ.NUMVERSION>LaVersion ALORS

LibVersion..Couleur=RougeClair

LibVersion..Bulle += RC+ChaîneConstruit("La version %1 est
disponible.",REQ.NUMVERSION)

FIN

HLibèreRequête(REQ)

FIN

FIN

Ainsi, cette fenêtre sera accessible depuis le menu d'administration (donc seulement aux administrateurs Gestan).

Les paramètres d'extension sont stockés dans le fichier PARAMPLUS. Ce fichier permet de stocker tout type de paramètre. Il fonctionne classiquement avec un code paramètre, un code user éventuel, et une valeur de paramètre.

Les paramètres utilisés par vos bibliothèques doivent commencer par le préfixe Zxx_, xx étant votre n° d'addon communiqué par ICS. Tout autre code est susceptible d'être écrasé par Gestan.

Détail de la fonction [nom wdl]_Init

Identification de l'extension

Voici un exemple :

```
// Identification de l'extension
gCodeExtension = "ICS-
INFORMATIQUE"+TAB+"GESTAN_PROSTEP"+TAB+"43"+TAB+"0245"+TAB+"A1"+TAB+"PAYS | CO
NTACT | MSG_TYPE | ADRESSEPOST | PARAMPLUS | USER | PRODUIT | MACHINE | LIBTRADUIT | CNX | PAR
AMAPPLI | CIVILITE"+TAB+"Z43_PROCESS | Z43_PROSPECT"
```

Cette chaîne est construite comme suit :

- le nom du développeur (ici, ICS-INFORMATIQUE)
- le nom de l'extension, sans ".wdl" (ici, GESTAN_PROSTEP)
- le suffixe de fichier attribué par ICS, même si l'extension n'utilise pas de fichiers (ici, 43)
- le code de l'extension (attribué par ICS au développeur - ici 0245)
- la version de Gestan pour laquelle l'extension a été développée (ici A1)
- la liste des fichiers de Gestan auxquels l'extension accède, séparés par des pipes.
- la liste des fichiers spécifiques à l'extension (tous préfixés par Zxx_)

Dans Gestan, la liste des fichiers et de leurs version sert de référence. Par exemple, si en version A2, seul le fichier ECRITURE a été modifié, l'extension, qui n'utilise pas le fichier ECRITURE, pourra fonctionner même si elle est en version A1 et que Gestan est en A2.

Fonction de contrôle de version et de déclaration éventuelle des fichiers

Cette partie est normalisée, vous n'avez rien à faire

La partie de test de l'erreur 70016 (SI ExceptionInfo(errCode)=70016 ALORS, etc.) est utile dans le cas où vous avez changé les structures de vos fichier Zxx : elle permet d'en déclencher la remise au format automatique via une fonction de Gestan "core".

Fonction de mise en place des menus

Cette fonction nécessite un peu d'attention pour son fonctionnement correct. Elle doit fonctionner pour l'option "menu classique", ou pour l'option "menu ruban".

Mise en place de la fenêtre de paramétrage de l'extension

S'il y a une fenêtre de paramétrage pour l'extension, l'écran de paramétrage doit être disponible dans :

- outils→paramètres de l'application→paramétrage des extensions→ l'option pour votre extension dans le menu classique
- outils→bouton paramètres→paramétrage des extensions→l'option pour votre extension dans le menu classique

****Pour le menu classique : ****

```
// Ajout de l'option dans le menu
MenuAjouteOption("Menu._Menu.mnuOutils.Paramétrage_application.Paramétrage_addons", "MNU_EXT_PROSTEP_param", "Étapes prospection", "Paramétrer_Addon", "FI_PARAMPLUS_PROSTEP|Extension PROSTEP")
// Eventuellement, une image pour cette option
{"MNU_EXT_PROSTEP_param", indChamp}..Image = "prostep_16_1.png"
```

Pour le menu ruban :

```
// Ajout de la fenêtre de paramétrage
EXTERNE Ruban1,mnuRub_ContactPlus
MenuAjouteOption("mnuRub_Paramétrage_application.Paramétrage_Extensions","MNU_EXT_PROSTEP_param","Étapes prospection", "Paramétrer_Addon", "FI_PARAMPLUS_PROSTEP|Extension PROSTEP")

// Eventuellement, une image pour cette option
{"MNU_EXT_PROSTEP_param", indChamp}..Image = "prostep_16_1.png"
```

On a :

- **MNU_EXT_PROSTEP_param** est un nom arbitraire que vous choisissez pour votre menu de paramétrage
- **Étapes prospection** est le libellé qui sera affiché pour l'option
- **Paramétrer_Addon** est le nom de la procédure d'appel (ne pas modifier)
- **FI_PARAMPLUS_PROSTEP|Extension PROSTEP** est la concaténation du nom de la fenêtre de paramétrage, et du titre que vous voulez lui voir affiché.

Mise en place des fenêtres de l'extension

Il y a différents cas possibles, qui donnent lieu à un codage différent.

**** __ L'ajout dans un menu __ ****

```
// Ajout du menu spécifique (option dans le menu Stats)
MenuInsèreOption("Menu._Menu.mnuContacts", "mnuAction_commerciale", "MNU_EXT_PROSTEP", "Étapes prospection", Charger_Fenêtre,"Table_Z43_PROSPECT")
```

```
{ "MNU_EXT_PROSTEP", indChamp }..Image = "prostep_16_1.png"
```

Avec le menu classique, ce code va ajouter une option dans le menu "Contacts" de Gestan, juste avant l'option "Action Commerciale", option qui va ouvrir la table Table_Z43_PROSPECT.

Pour le menu ruban le codage est un peu différent : il faut d'abord activer le bouton ContactPlus, puis ajouter le menu.

Cela donne :

```
EXTERNE Ruban1,mnuRub_ContactPlus
// Ajout du menu spécifique (option dans le bouton StatPlus)
Ruban1.ContactPlus..Visible=Vrai
MenuAjouteOption("mnuRub_ContactPlus","MNU_EXT_PROSTEP","Étapes
prospection", Charger_Fenêtre,"Table_Z43_PROSPECT")
{ "MNU_EXT_PROSTEP", indChamp }..Image = "prostep_16_1.png"
mnuRub_ContactPlus.OptAjoutContactPlus..Visible=Faux
```

**** __ L'ajout d'un menu sur le bouton "Plus" __ ****

C'est le cas d'un menu déroulant qui n'est pas repris dans le menu ruban. Pour le menu classique, rien ne change :

```
// On insère une option dans le menu Stat de Gestan
MenuInsèreOption("Menu._Menu.mnuStatistiques",
"mnuSuivi_des_transformations", "MNU_EXT_CHRONSTAT", "Séries statistiques",
Charger_Fenêtre, "Table_Z1_SERIESTAT")
```

En revanche, pour le menu ruban, on active un bouton "+", ici StatP+, on ajoute l'option dans le menu lié au bouton, et on efface l'option qui sert à l'ajout. Cela donne :

```
// On ajoute une option au bouton StatPlus
Ruban1.StatsPlus..Visible=Vrai
MenuAjouteOption("mnuRub_StatsPlus", "MNU_EXT_CHRONSTAT", "Séries
statistiques", Charger_Fenêtre, "Table_Z1_SERIESTAT")
mnuRub_StatsPlus.OptAjoutStatPlus..Visible=Faux
```

**** __ L'ajout dans le menu "métier" __ ****

Dans le cas du menu classique, on affiche le menu métier et on ajoute l'option. Cela donne :

```
MenuInsèreMenu(MENU._Menu.mnuMétier, "Pour_ajout_possible", "MNU_EXT_DEV",
"Développement")
{ "MNU_EXT_DEV", indChamp }..Image = "gear_blue_16_1.png"
Menu._Menu.mnuMétier..Visible=Vrai
MenuAjouteOption("MNU_EXT_DEV", "MNU_EXT_DEV_devs", "Développements",
Charger_Fenêtre, "Table_Z28_DEV")
MenuAjouteOption("MNU_EXT_DEV", "MNU_EXT_DEV_versions", "Versions",
Charger_Fenêtre, "Table_Z28_VERSION")
MenuAjouteOption("MNU_EXT_DEV", "MNU_EXT_DEV_domaines", "Domaines",
Charger_Fenêtre, "Table_Z28_DOMAINE")
```

```
{ "MNU_EXT_DEV_devs", indChamp }.Image = "gear_blue_16_1.png"
```

Et pour le menu ruban :

```
EXTERNE Ruban1,btnRub_Metier1,mnuRub_Métier1
Ruban1.Métier..Visible=Vrai
gNbBoutonsMétier++
{"btnRub_Metier"+(gNbBoutonsMétier),indChamp}..Visible =Vrai
{"btnRub_Metier"+(gNbBoutonsMétier),indChamp}..Libellé="Dev"
{"btnRub_Metier"+(gNbBoutonsMétier),indChamp}..Image="gear_blue_32_1.png"
MenuAjouteMenu("mnuRub_Métier"+(gNbBoutonsMétier), "MNU_EXT_DEV",
"Développement")
MenuAjouteOption("MNU_EXT_DEV", "MNU_EXT_DEV_devs", "Développements",
Charger_Fenêtre, "Table_Z28_DEV")
MenuAjouteOption("MNU_EXT_DEV", "MNU_EXT_DEV_versions", "Versions",
Charger_Fenêtre, "Table_Z28_VERSION")
MenuAjouteOption("MNU_EXT_DEV", "MNU_EXT_DEV_domaines", "Domaines",
Charger_Fenêtre, "Table_Z28_DOMAINE")
{"MNU_EXT_DEV_devs", indChamp}..Image = "gear_blue_16_1.png"
```

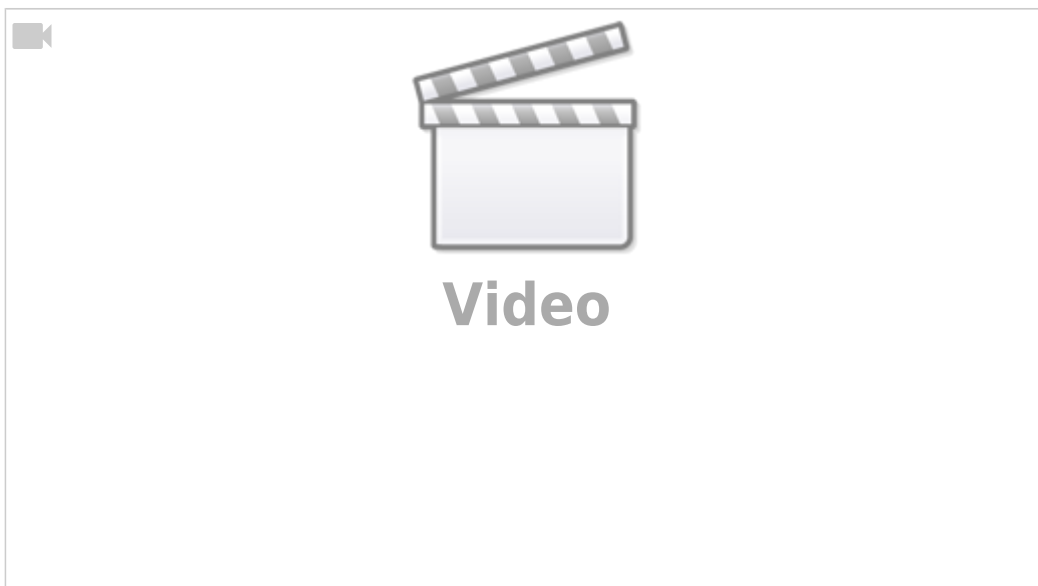
Utiliser des fichiers externes dans une extension

Par exemple, vous voulez utiliser les fichiers PAYS et PARAMLIB de Gestan. Vous avez alors trois options :

- Avec un hDéclare en référençant l'analyse de Gestan
- Via un hDéclareExterne
- En copiant-collant les fichiers que vous voulez utiliser dans l'analyse de extension :====

Mais ça, c'était avant !

Le plus simple à partir de la version 2024 de Windev est de fonctionner en mode multi-analyse, et d'intégrer l'analyse de Gestan.



Version de compilation



Création de bibliothèque

Informations de version de la bibliothèque

Décrivez les informations sur la bibliothèque.

Société : ICS-Informatique

Description : Gestion des étapes de prospection

Version : 3.A1.05.07

Copyright : ICS-Informatique 2025

Si vous le souhaitez, la version sera automatiquement incrémentée lors des prochaines créations de la bibliothèque.

☐ Incrémenter automatiquement la version à chaque création

WINDEV 2024

Précédent Terminer Suivant

Pensez à indiquer la bonne version au moment de la compilation de la WDL, ici, 3.A1.05.07.

Si A1 n'est pas précisé, Gestan refusera de charger la bibliothèque.

Autres articles "Développement"

[Développement d'extensions en version Ax](#)

[Intégrer une pièce dans Gestan depuis une extension](#)

[Ressources](#)

[Rédiger des pages d'aide sur le wiki](#)

From:

<https://manuel.gestan.fr/> - **Le manuel de Gestan**

Permanent link:

<https://manuel.gestan.fr/wiki/partenaires/dev/devextax?rev=1783713552>

Last update: **2026/07/10 21:59**