DÉVELOPPEZ 10 FOIS PLUS VITE

WINDEV - Concepts
Version 23 - (1) 10-2017

Visitez régulièrement le site **www.pcsoft.fr**, espace téléchargement, pour vérifier si des mises à jour sont proposées.

Adresse e-mail du Support Technique Gratuit : **supportgratuit@pcsoft.fr.**

Conseil : Pour recevoir directement des mises à jour intermédiaires et des conseils d'utilisation, abonnez-vous à la **LST** (revue trimestrielle + DVD), en français.

Cette documentation n'est pas contractuelle. Des modifications ont pu intervenir dans le logiciel depuis la parution de ce manuel. Veuillez consulter l'**aide en ligne**.

Tous les noms de produits ou autres marques cités dans cet ouvrage sont des marques déposées par leurs propriétaires respectifs.
© PC SOFT 2017 : Aucune reproduction intégrale ou partielle du présent ouvrage sur quelque support que ce soit ne peut être effectuée sans l'autorisation expresse de PC SOFT.

Concepts

Dans quel ordre lire les manuels ?

WINDEV est un puissant outil de développement d'applications Windows, qui fournit en standard tous les outils nécessaires à la conception et à la réalisation d'applications.

Pour une formation rapide et efficace à WINDEV, nous vous conseillons l'ordre d'apprentissage suivant :

1 Lecture des "Concepts".
Ce manuel présente les principaux concepts nécessaires à la création d'une application WINDEV performante. Certains concepts sont suivis d'une partie "Pratique" détaillant certaines fonctionnalités de l'éditeur.

2 "Auto-formation" (livre + exercices)
Le guide d'auto-formation permet une première approche "pratique" de WINDEV. Vous pouvez ainsi vous familiariser avec les principaux éditeurs de WINDEV.

3 Test des exemples
Testez les différents exemples livrés avec WINDEV dans les domaines qui vous intéressent (gestion commerciale, planning, ...)

L'aide en ligne, accessible par Internet à l'adresse http://doc.pcsoft.fr ou installée avec WINDEV permet de trouver rapidement la syntaxe d'une fonction du WLangage, d'obtenir une aide sur l'interface, ... Pour chaque thème de programmation, vous trouverez une description de la fonctionnalité associée et la liste des fonctions du WLangage correspondantes.

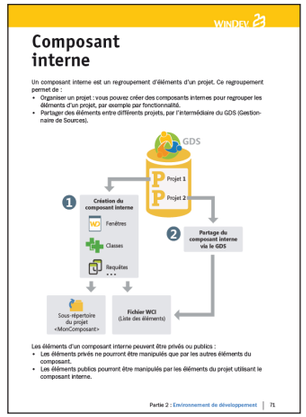
Remarque : En cas de divergence entre les manuels et l'aide en ligne, suivez les instructions de l'aide en ligne.

Nous vous souhaitons une agréable prise en main de WINDEV.

Organisation du manuel

Ce manuel présente les principaux concepts nécessaires à la création d'une application WINDEV performante. Certains concepts sont suivis d'une partie "Pratique" détaillant certaines fonctionnalités de l'éditeur.

Deux types de pages sont donc présents dans ce manuel :



Composant interne

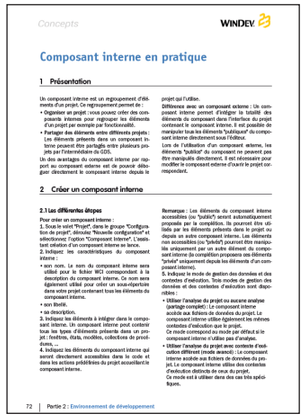
Un composant interne est un regroupement d'éléments d'un projet. Ce regroupement permet de :
- Organiser un projet : vous pouvez créer des composants internes pour regrouper les éléments d'un projet, par exemple par fonctionnalité.
- Partager des éléments entre différents projets, par l'intermédiaire du GDS (Gestionnaire de Sources).

GDS
Projet 1
Projet 2
1 Création du composant interne
Fenêtres
Classes
Requêtes
2 Partage du composant interne via le GDS
Sous-répertoire du projet «MonComposant»
Fichier WCI (Liste des éléments)

Les éléments d'un composant interne peuvent être privés ou publics :
- Les éléments privés ne pourront être manipulés que par les autres éléments du composant.
- Les éléments publics pourront être manipulés par les éléments du projet utilisant le composant interne.

Partie 2 : Environnement de développement | 71

Page concept



Concepts

Composant interne en pratique

1 Présentation

Un composant interne est un regroupement d'éléments d'un projet. Ce regroupement permet de :
- Organiser un projet : vous pouvez créer des composants internes pour regrouper les éléments d'un projet par exemple par fonctionnalité.
- Partager des éléments entre différents projets : Les éléments présents dans un composant interne peuvent être partagés entre plusieurs projets par l'intermédiaire du GDS.

Un des avantages du composant interne par rapport au composant externe est de pouvoir déboguer directement le composant interne depuis le projet qui l'utilise.

Différence avec un composant externe : Un composant interne permet d'intégrer la totalité des éléments du composant dans l'interface du projet contenant le composant interne. Il est possible de manipuler tous les éléments "publiques" du composant interne directement sous l'éditeur.

Lors de l'utilisation d'un composant externe, les éléments "publics" du composant ne peuvent pas être manipulés directement. Il est nécessaire pour modifier le composant externe d'ouvrir le projet correspondant.

2 Créer un composant interne

2.1 Les différentes étapes

Pour créer un composant interne :
1. Sous le volet "Projet", dans le groupe "Configuration de projet", déroulez "Nouvelle configuration" et sélectionnez l'option "Composant interne". L'assistant création d'un composant interne se lance.
2. Indiquez les caractéristiques du composant interne :
- son nom. Le nom du composant interne sera utilisé pour le fichier WCI correspondant à la description du composant interne. Ce nom sera également utilisé pour créer un sous-répertoire dans votre projet contenant tous les éléments du composant interne.
- son libellé.
- sa description.
3. Indiquez les éléments à intégrer dans le composant interne. Un composant interne peut contenir tous les types d'éléments présents dans un projet : fenêtres, états, modèles, collections de procédures, ...
4. Indiquez les éléments du composant interne qui seront directement accessibles dans le code et dans les actions prédéfinies du projet accueillant le composant interne.

Remarque : Les éléments du composant interne accessibles (ou "publics") seront automatiquement proposés par la complétion. Ils pourront être utilisés par les éléments présents dans le projet ou dans les autres composants internes. Les éléments non accessibles (ou "privés") pourront être manipulés uniquement par un autre élément du composant interne (la complétion proposera ces éléments "privés" uniquement depuis les éléments d'un composant interne).
5. Indiquez le mode de gestion des données et des contextes d'exécution. Trois modes de gestion des données et des contextes d'exécution sont disponibles :
- Utiliser l'analyse du projet ou aucune analyse (partage complet) : Le composant interne accède aux fichiers de données du projet. Le composant interne utilise les mêmes contextes d'exécution que le projet.
Ce mode correspond au mode par défaut si le composant interne n'utilise pas d'analyse.
- Utiliser l'analyse du projet avec contexte d'exécution différent (mode avancé) : Le composant interne accède aux fichiers de données du projet. Le composant interne utilise des contextes d'exécution distincts de ceux du projet.
Ce mode est à utiliser dans des cas très spécifiques.

72 | Partie 2 : Environnement de développement

Page de mise en pratique

Sommaire

Partie 1 - Notions de base

Projet et Analyse 9
Cycle de développement d'une application 14
IHM : Les fenêtres 16
Fenêtres modales, non modales ou MDI 17
Fenêtre interne 19
Modèles de fenêtres 20
Les différents types de champs standard 34
Modèles de champs 49
Edition : Les états 50
Modèles d'états 51

Partie 2 - Environnement de développement

Les éditeurs de WINDEV 55
Tableau de bord du projet 59
WINDEV/WEBDEV/WINDEV Mobile : 100% compatibles 60
Configuration de projet 61
Génération multiple 62
Perso-dossiers :
Organisez votre projet 63
Gestionnaire de sources (GDS) 64
Composant interne 71
Composant externe 74
Modes de génération 82
Centre de Suivi de Projets 87
Gestion des exigences 88
Gestion des tâches 89
Gestion des suggestions et des incidents 90
Gestion des règles métier 91

Partie 3 - Bases de données

Analyse : Structure de la base de données 95
Les différents types de fichiers accessibles 104
HFSQL Classic 106
HFSQL Client/Serveur 107
Cluster HFSQL Client/Serveur 108
HFSQL : les fichiers créés physiquement 109
Associer les champs et les données 110
Les requêtes 112
Les requêtes intégrées 113

Les champs Table/Zone Répétée 114
Réplication universelle 116
Architecture 3-tiers 117

Partie 4 - Concepts avancés
RAD RID 121
Le modèle UML 122
Groupware utilisateur 129
Applications multilingues 137
Optimiser une application 143
Livrer "Etats et Requêtes" 145
MCU : Macro Code Utilisateur 146
Les différents tests disponibles 148
Le débogueur 149
Tests unitaires/Tests automatiques 151
Tests unitaires sur l'exécutable 152

Partie 5 - Installation
Le Framework WINDEV 161
Les différents modes d'installation d'une application 163
Surveillez vos applications 172

Partie 6 - Communication
Gestion des emails 175
Transfert de fichiers par FTP 177
Gestion des sockets 179
WINDEV et la téléphonie 181
Les Webservices 182

Partie 7 - Annexes
Composants livrés avec WINDEV 185
Exemples livrés avec WINDEV 189

DÉVELOPPEZ 10 FOIS PLUS VITE

PARTIE 1

Notions de base

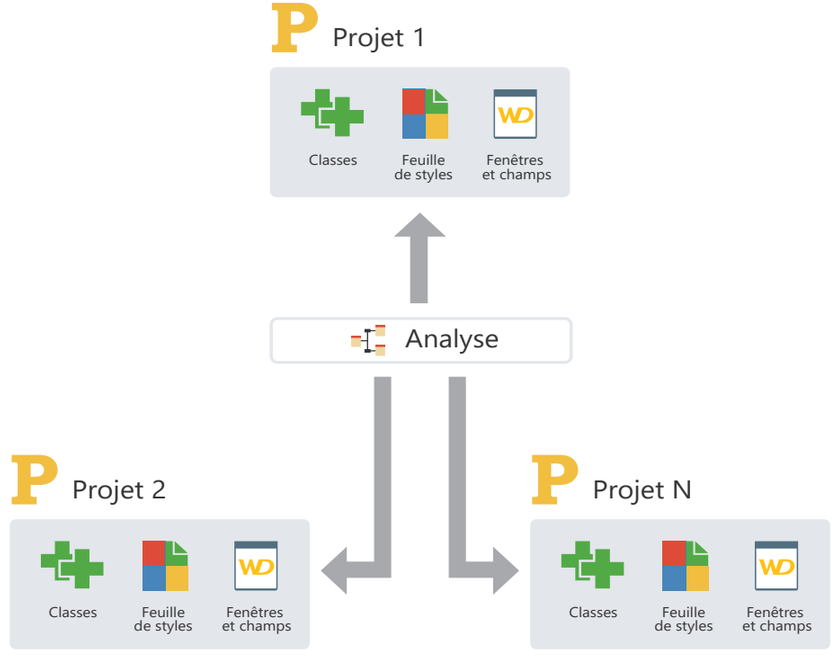
Projet et Analyse

Le développement d'une **Application Windows** en utilisant WINDEV repose sur deux éléments principaux : le Projet et l'Analyse.

Un **Projet WINDEV** est un ensemble d'éléments : fenêtres, états, champs, classes, composants, ... L'assemblage de ces éléments permet de réaliser une application Windows.

Une **Analyse WINDEV** correspond au schéma des données utilisé par l'application. L'analyse regroupe les différentes tables (aussi appelées fichiers de données) utilisées par l'application.

Une application est construite à partir d'un projet.
Un projet est généralement associé à une analyse.
Une analyse peut être associée à un ou plusieurs projets.



Le projet en pratique

1 Présentation

La première étape de la description d'une application consiste à créer un projet.

Lors de la création d'un projet, plusieurs questions vous sont posées afin que votre projet corresponde à toutes vos attentes. Toutes les caractéristiques du projet spécifiées lors de la création du projet pourront être modifiées par la suite.

Remarque : Les analyses sont présentées en détail dans le chapitre "Analyse : Structure de la base de données", page 95.

2 Création du projet

Pour créer un projet :

1. Cliquez sur parmi les boutons d'accès rapide de WINDEV. La fenêtre de création d'un nouvel élément s'affiche : cliquez sur "Projet". L'assistant de création d'un projet se lance.

2. Spécifiez les différentes options du projet. Les principales options sont les suivantes :

- ***le nom et l'emplacement.*** Ces options ne seront pas modifiables. Le projet correspond à un fichier ".WDP". Tous les éléments associés au projet seront créés dans le répertoire spécifié.
- ***le type de projet.*** De nombreuses possibilités sont offertes : application, composant, application Java, ... Ces options sont détaillées plus tard. En règle générale, un projet permet de générer une application (exécutable).
- le ***système d'exploitation*** et le mode d'exécution (pour les exécutables).
- ***si le projet va être manipulé par plusieurs développeurs.*** Il est possible d'utiliser le gestionnaire de sources (GDS) pour partager le projet.
- ***la charte de programmation.*** Cette charte permet de préfixer automatiquement les variables, les noms de champs, de fenêtres, ...
- ***la charte graphique***.
- ***les langues gérées.*** Ces langues seront proposées par défaut dès qu'une option pourra être traduite dans un champ, une fenêtre, un état, ...

3. Indiquez ***si le projet doit utiliser une base de données ou non***. Si oui, la base de données peut exister ou être nouvelle.

4. Validez l'assistant. Le projet créé devient le projet en cours.

Remarques :

- Si vous avez demandé la création d'une base de données, l'assistant correspondant se lance automatiquement.
- L'ensemble des informations fournies peut être modifié dans la description du projet. Pour afficher la fenêtre de description du projet, sous le volet "Projet", dans le groupe "Projet", cliquez sur "Description".

3 Tableau de bord et graphe du projet

3.1 Tableau de bord

Tout responsable de projet cherche à avoir une vision globale et synthétique de l'état d'avancement des projets qu'il dirige.

Tout responsable qualité veut connaître le nombre et l'importance des bogues des projets et en suivre l'évolution.

Le responsable fonctionnel cherche à savoir quelles sont les évolutions demandées par les utilisateurs.

Le développeur cherche à lancer directement et rapidement les éléments du projet, les codes les plus utilisés, ...

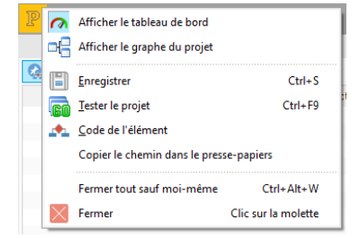
Le tableau de bord permet de répondre à tous ces souhaits. Le tableau de bord propose différents indicateurs sous forme de "Widgets". Ces Widgets sont paramétrables par utilisateur : il est possible d'en ajouter, supprimer ou encore de les redimensionner.

Il est possible par exemple d'afficher les statistiques sur le projet (nombre de fenêtres, ...), le résultat des différents audits (audit statique, dynamique, ...), ...

Pour afficher le tableau de bord de votre projet :

1. Affichez le menu contextuel du projet dans la barre des documents ouverts.

2. Sélectionnez l'option "Afficher le tableau de bord".



Pour plus de détails, consultez le paragraphe “Tableau de bord du projet”, page 59.

3.2 Graphe du projet

L'éditeur de projet permet de visualiser de manière graphique les différents éléments du projet (fenêtres, états, requêtes, ...) et leurs enchaînements.

Pour afficher le graphe de votre projet :

1. Affichez le menu contextuel du projet dans la barre des documents ouverts.

2. Sélectionnez l'option "Afficher le graphe du projet".

3.3 Les éléments du projet

Le projet est composé de fenêtres, d'états, de requêtes, de collections de procédures, ...

Pour connaître la liste des éléments d'un projet, sous le volet "Projet", dans le groupe "Projet", cliquez sur "Liste des éléments".

Cette option permet de :

- ajouter à votre projet des éléments appartenant à des projets accessibles depuis votre poste. Les fichiers correspondants ne seront pas déplacés dans le répertoire de votre projet.
- supprimer des éléments de votre projet. Les fichiers correspondants ne sont pas supprimés physiquement.

Pour rechercher rapidement un élément de votre projet, utilisez la combinaison de touches "CTRL + E" depuis n'importe quel éditeur.

4 Manipulations du projet

Voici les principales manipulations pouvant être réalisées sur un projet :

- Archiver un projet.
- Restaurer un projet.
- Dupliquer un projet.
- Copier ou supprimer un projet.
- Renommer un projet.

Pour plus de détails sur ces fonctionnalités, consultez l'aide en ligne.

5 Dossier du projet

WINDEV permet d'imprimer différents dossiers détaillant l'ensemble des éléments (fenêtre, état, fichier de données, rubrique, ...) du projet. Pour imprimer ces dossiers :

- sous le volet "Accueil", dans le groupe "Général", déroulez l'icône et sélectionnez l'option "Imprimer le dossier du projet". Le dossier peut contenir l'ensemble des caractéristiques du projet.
- sous le volet "Accueil", dans le groupe "Général", cliquez sur l'icône . Le dossier est alors constitué uniquement des caractéristiques de l'élément (fenêtre, état, requête, ...) en cours.

5.1 Type de dossier

Plusieurs types de dossier sont proposés :

- Dossier Analyse / Fichier de données / Requêtes : Contient les informations sur l'analyse, les fichiers de données et les requêtes du projet.
- Dossier IHM (Interface Homme Machine) : Contient uniquement la représentation des fenêtres et des états, ainsi que leur enchaînement.
- Dossier Code : Contient l'ensemble des traitements de tous les objets du projet.
- Dossier Technique : Contient les traitements et les descriptions techniques de tous les objets du projet.
- Dossier Complet : Contient toutes les informations du projet. Ce dossier correspond à la fusion de tous les types de dossiers.

Type d'édition du dossier

Le dossier peut au choix être :

- imprimé,
- exporté vers un document RTF, vers un fichier XML,
- exporté vers un fichier HTML (un navigateur internet doit être installé sur le poste en cours),
- exporté vers un fichier texte.

5.2 Zones d'impression

Si des représentations graphiques (graphe du projet, MLD, diagrammes UML, ...) doivent être imprimées, il est nécessaire de configurer les zones d'impression **avant** d'imprimer le dossier.

Pour spécifier les zones d'impression de la représentation graphique en cours :

1. Sous le volet "Affichage", dans le groupe "Edition", cochez l'option "Zones d'impression". Des bordures représentant les zones imprimables dans le dossier apparaissent sous l'éditeur en cours.

2. Réduisez l'affichage de la représentation graphique (option "Zoom" du volet "Affichage" ou [Ctrl] + Roulette de la souris) pour visualiser l'ensemble du graphe.

3. A l'aide de la souris :

- déplacez ces bordures à l'endroit désiré en maintenant le bouton de la souris enfoncé (la couleur du curseur de la souris est noire).
- choisissez le nombre de pages sur lesquelles la représentation graphique doit être imprimée (le curseur de la souris se transforme en double-flèche Nord-Ouest / Sud-Est).

Si le format d'impression est modifié (passage de A4 à A3 par exemple dans les propriétés de l'imprimante), il est possible d'adapter le format des pages du dossier.

Pour cela :

1. Définissez des zones d'impression.

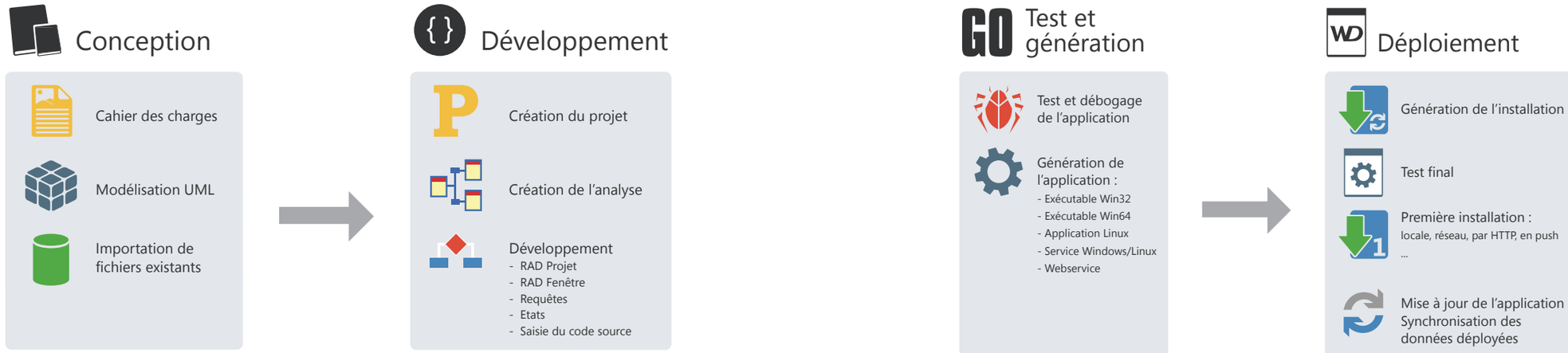
2. Affichez les options de mise en page du dossier : sous le volet "Accueil", dans le groupe "Général", déroulez l'icône et sélectionnez l'option "Mise en page du dossier".

3. Cochez l'option "Adapter la taille des zones d'impression des documents ouverts".

4. Lancez l'impression du dossier.

Cycle de développement d'une application

WINDEV couvre la totalité du cycle de développement d'une application :



Détail des différentes phases :

Phase de conception : Il est possible de concevoir une application à partir d'un simple cahier des charges, d'une modélisation UML des traitements ou même à partir de fichiers de données préexistants.

Phase de développement : La création du projet et de l'analyse est réalisée à l'aide d'assistants très complets. Le développement peut être effectué en mode RAD (Rapid Development Application) avec génération automatique du code et des IHM ou être le résultat d'une création manuelle des différents éléments du projet.

Phase de tests et génération : WINDEV offre toute une panoplie d'outils de tests automatiques pour garantir la fiabilité des applications et assurer la non-régression entre les phases de développement.

Phase de déploiement : Le déploiement d'une application WINDEV peut se faire de nombreuses façons : téléchargement, via un réseau local (y compris en mode Push) ou via Internet. Dans tous les cas, les fichiers de données HFSQL seront automatiquement mis à jour en fonction des évolutions de l'application.

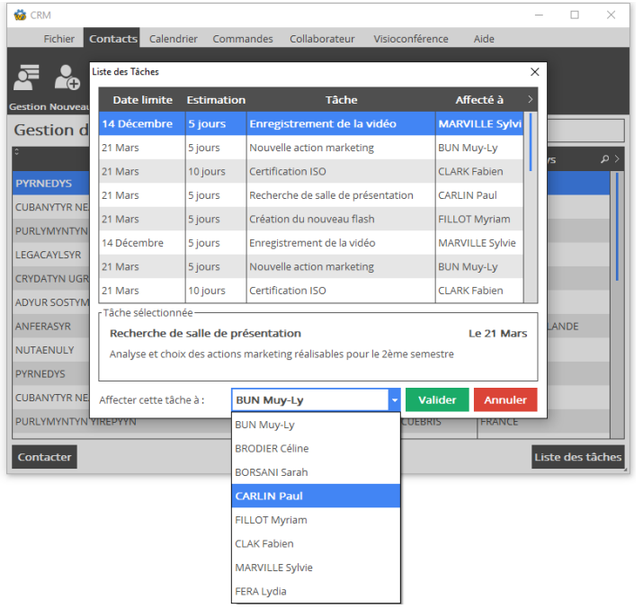
IHM : Les fenêtres

Les fenêtres constituent la base de l'IHM (Interface Homme Machine) d'une application.

WINDEV est constitué d'un éditeur de fenêtres particulièrement évolué permettant de réaliser facilement et rapidement tous les types d'IHM possibles.

De nombreuses fonctionnalités permettent d'obtenir simplement des applications intuitives et agréables d'utilisation :

- des champs puissants et variés,
- un mécanisme d'ancrages permettant à l'IHM de s'adapter automatiquement à la taille de l'affichage,
- un système de compilation de l'IHM avec détection des erreurs (titres vides, libellés non traduits, chevauchements, ...),
- ...



Fenêtres modales, non modales ou MDI

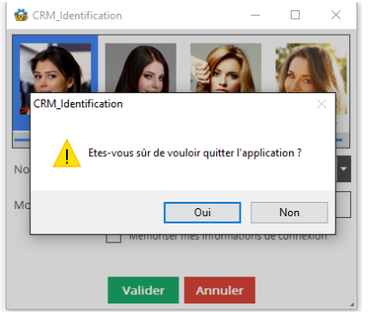
Les applications créées avec WINDEV peuvent utiliser plusieurs types de fenêtres (et mélanger chacun de ces types).

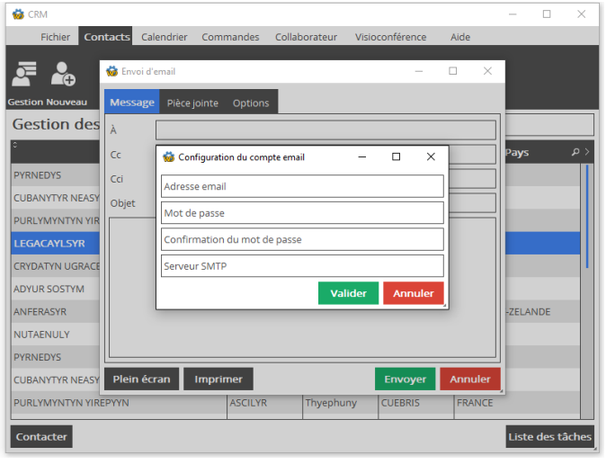
Fenêtres modales

Une fenêtre modale s'ouvre en avant-plan de l'application et bloque l'accès aux autres fenêtres déjà ouvertes (s'il y en a). Les fenêtres modales sont généralement utilisées comme :

- fenêtres principales d'une application,
- boîtes de dialogue dont la validation provoque une action.

Avec WINDEV, lorsqu'une fenêtre modale s'ouvre, une ombre grise apparaît sur les fenêtres en arrière-plan : c'est le GFI ou Grisage des Fenêtres Inaccessibles.

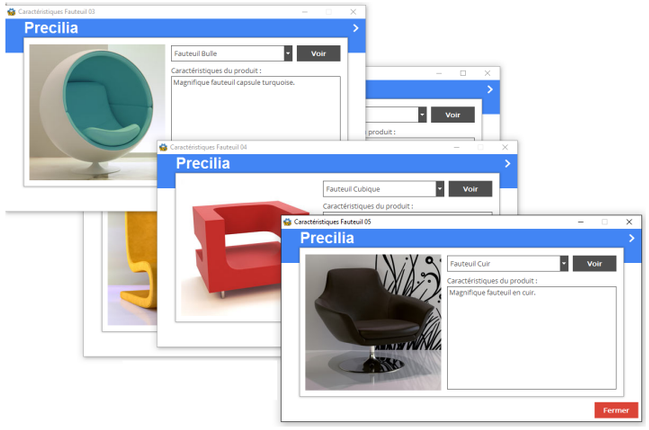




Fenêtres non modales

Une fenêtre non modale s'ouvre en parallèle des autres fenêtres de l'application. Chacune des fenêtres reste accessible et manipulable. Les fenêtres non modales servent généralement :

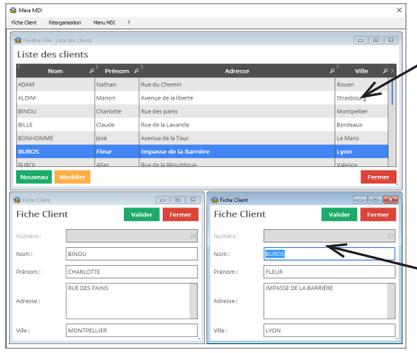
- de palettes d'outils,
- de boîtes de dialogue dont la modification affecte directement la fenêtre principale.



Dans cet exemple, chaque fenêtre de description d'un polo permet d'ouvrir la fenêtre de description d'un autre polo. Toutes les fenêtres ouvertes restent accessibles.

Fenêtres MDI

Dans une application MDI, toutes les fenêtres de l'application sont affichées dans une seule et même fenêtre principale : la fenêtre mère MDI. Les fenêtres des applications MDI peuvent donc être de deux types :



Fenêtre mère MDI : Toutes les autres fenêtres de l'application sont affichées dans cette fenêtre. Cette fenêtre permet à l'utilisateur de manipuler le menu principal de l'application.

Fenêtre fille MDI : Fenêtre de l'application affichée dans la fenêtre mère. C'est dans ces fenêtres que l'utilisateur pourra saisir des informations.

Fenêtre interne

Le champ Fenêtre interne permet d'inclure une fenêtre (et son code) dans une autre fenêtre. A l'exécution, la fenêtre interne sera dynamiquement fusionnée à la fenêtre hôte.

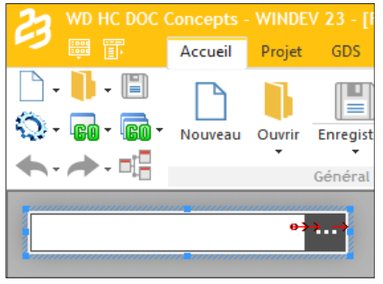
1. Création d'une fenêtre interne.

Pour créer une fenêtre interne :

- Cliquez sur parmi les boutons d'accès rapide.
- La fenêtre de création d'un nouvel élément s'affiche : cliquez sur "Fenêtre" puis sur "Fenêtre interne".

Une fenêtre interne est une fenêtre spécifique qui ne comporte ni barre de titre, ni menu.
Dans cette fenêtre, tous les types de champs peuvent être utilisés.

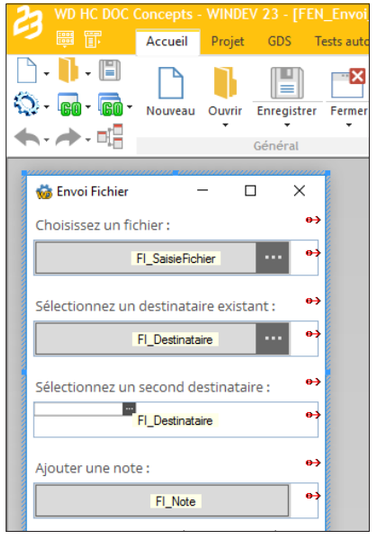
Dans l'exemple, une fenêtre interne de sélection de fichiers est définie.



2. Utilisation d'une fenêtre interne.

Pour utiliser une fenêtre interne :

- Créez un champ de type fenêtre interne.
- Dans la fenêtre de description du champ, sélectionnez la fenêtre interne à utiliser.
- Validez la fenêtre de description du champ.



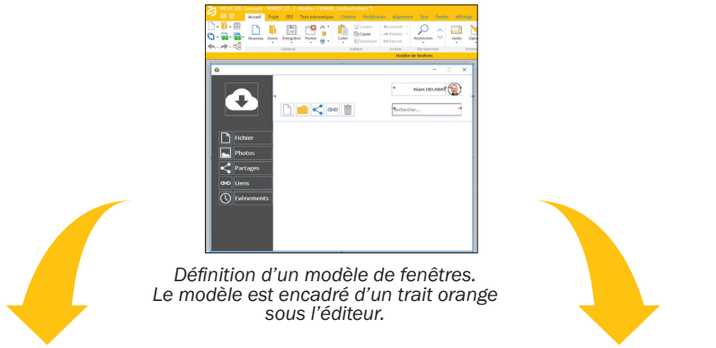
Remarques :

- Il est possible de modifier par programmation la fenêtre interne utilisée dans le champ "Fenêtre interne".
- Limitations : La zone d'accueil est rectangulaire et aucune surcharge n'est possible. Pour réaliser des surcharges, il est conseillé d'utiliser des modèles de champs.

Modèles de fenêtres

WINDEV permet de créer des modèles de fenêtres. Ces modèles contiennent tous les éléments graphiques communs à l'ensemble des fenêtres de votre application.
Les modifications effectuées dans un modèle de fenêtres sont automatiquement reportées sur toutes les fenêtres utilisant ce modèle.

Un modèle de fenêtres facilite le respect de la charte graphique d'une application.



Définition d'un modèle de fenêtres.
Le modèle est encadré d'un trait orange sous l'éditeur.

Utilisation du modèle dans plusieurs fenêtres.
Les éléments appartenant au modèle sont identifiés par un carré jaune.

Remarque : La programmation associée aux éléments du modèle peut être directement réalisée dans le modèle.
Les caractéristiques des éléments peuvent être désolidarisées du modèle. Par exemple, désolidariser la position d'un champ du modèle pour positionner le champ ailleurs tout en conservant les autres évolutions sur le champ (code, style, ...). On parle alors d'**héritage de champ**. Dans ce cas, les éléments sont identifiés par un carré bleu.

Les fenêtres en pratique

Une fenêtre a pour but d'afficher, de consulter et de saisir des informations. Ces informations peuvent provenir de fichiers de données d'une analyse, de fichiers externes, de requêtes, ...

WINDEV propose plusieurs solutions pour créer les fenêtres d'un projet :

- Créer une fenêtre vierge avec l'assistant.
- Construire l'ensemble des fenêtres de l'application à partir de la description de l'analyse.
- Créer une fenêtre à partir de la description de l'analyse (avec son code ou non).
- Créer des fenêtres standard.
- Créer des fenêtres basées sur un modèle, ...

Quelle que soit la méthode utilisée, la fenêtre pourra être modifiée après sa création : il sera possible d'ajouter, modifier ou supprimer des champs, de modifier les caractéristiques de la fenêtre.

Pour plus de détails sur la manipulation des champs d'une fenêtre, consultez "Les champs en pratique", page 39.

Ce chapitre traite les sujets suivants :

- "Création d'une fenêtre", page 21.
- "Manipulations simples sur une fenêtre depuis l'éditeur", page 22.
- "Principales caractéristiques d'une fenêtre", page 22.
- "Fenêtre et image", page 24.
- "Ouvrir une fenêtre "popup"", page 26.
- "Grisage automatique des fenêtres", page 27.
- "Fenêtre interne", page 27.
- "Modèles de fenêtres", page 28.
- "Traitements associés aux fenêtres", page 28.
- "Les menus", page 29.
- "La barre de message", page 33.

1 Création d'une fenêtre

WINDEV permet de créer plusieurs types de fenêtres :

- Fenêtres vierges, sans aucun champ.
- Fenêtres créées par le RAD / RID.
- Fenêtres internes. Ces fenêtres pourront être utilisées dans des champs de type "Fenêtre Interne".
- Fenêtres basées sur un modèle de fenêtres.
- Fenêtres importées à partir d'une application non WINDEV.

Ce paragraphe détaille uniquement la création de fenêtres vierges, internes ou basées sur un modèle. Pour plus de détails, consultez l'aide en ligne.

1.1 Création de fenêtres vierges

Pour créer une fenêtre vierge :

1. Cliquez sur parmi les boutons d'accès rapide. La fenêtre de création d'un nouvel élément s'affiche : cliquez sur "Fenêtre" puis sur "Fenêtre". L'assistant de création d'une fenêtre se lance.
2. Choisissez le type "Vierge".
3. Spécifiez le gabarit de la fenêtre.
4. Validez la création de la fenêtre. Une fenêtre vierge est créée sous l'éditeur. La fenêtre de sauvegarde apparaît automatiquement.
5. Spécifiez le titre et le nom de la fenêtre. La fenêtre correspond à un fichier ".WDW". Par défaut, ce fichier sera créé dans le répertoire principal du projet. Ce nom sera utilisé pour manipuler la fenêtre.
6. Validez l'enregistrement de la fenêtre.
7. Créez les champs dans la fenêtre.

Remarque : Les principales caractéristiques des fenêtres sont détaillées dans l'aide en ligne. Les différents types de champs sont présentés dans le chapitre "Les champs en pratique", page 39.

1.2 Fenêtre interne

Une fenêtre interne est une fenêtre sans barre de titre, ni barre de message qui sera directement incluse dans les autres fenêtres de votre application grâce au champ "Fenêtre interne".

Ainsi, il est possible d'inclure une fenêtre (et son code) dans une autre fenêtre.

Pour créer une fenêtre interne :

1. Cliquez sur parmi les boutons d'accès rapide. La fenêtre de création d'un nouvel élément s'affiche : cliquez sur "Fenêtre" puis sur "Fenêtre". L'assistant de création d'une fenêtre se lance.
2. Sélectionnez l'onglet "Fenêtre interne".
3. Sélectionnez le type de fenêtre et le gabarit as-

socié.
4. Validez. La fenêtre de sauvegarde apparaît automatiquement.

1.3 Fenêtre basée sur un modèle

WINDEV permet également de créer une fenêtre basée sur un modèle de fenêtres. Dans ce cas, il suffit de sélectionner le modèle de fenêtres voulu.

Les modèles de fenêtres contiennent tous les éléments graphiques et le code communs à l'ensemble des fenêtres de votre application.

Pour créer une fenêtre basée sur un modèle :

1. Cliquez sur parmi les boutons d'accès rapide. La fenêtre de création d'un nouvel élément s'affiche : cliquez sur "Fenêtre" puis sur "Fenêtre". L'assistant de création d'une fenêtre se lance.
2. Sélectionnez l'onglet "Basée sur un modèle".
3. Sélectionnez le modèle voulu et le gabarit associé.
4. Validez. La fenêtre de sauvegarde apparaît automatiquement.

Les modifications effectuées dans un modèle de fenêtres sont automatiquement reportées sur toutes les fenêtres utilisant ce modèle.

Un modèle de fenêtres permet de respecter la charte graphique définie pour une application.

2 Manipulations simples sur une fenêtre depuis l'éditeur

L'éditeur de fenêtres permet de réaliser les manipulations suivantes sur les fenêtres :

- Ouvrir une fenêtre sous l'éditeur : Il suffit de double-cliquer sur le nom de la fenêtre affiché dans le volet "Explorateur de projet".
- Modifier la taille d'une fenêtre.
- Sauvegarder et copier une fenêtre.
- Importer et exporter une fenêtre.
- Modifier l'ordre de navigation.

Pour plus de détails sur ces manipulations, consultez l'aide en ligne.

3 Principales caractéristiques d'une fenêtre

3.1 Caractéristiques

Une fenêtre peut accueillir tous les types de champs disponibles. Une fenêtre peut :

- être ***redimensionnable*** : l'utilisateur pourra agrandir ou diminuer la taille de la fenêtre. Selon les caractéristiques d'ancrage définies pour chacun des champs, la position des champs dans la fenêtre pourra varier lors de ce redimensionnement (pour plus de détails sur l'ancrage, consultez "Ancrage des champs", page 45).
- avoir une ***image de fond*** : cette image est affichée en fond de la fenêtre et les champs viennent se superposer à cette image. Il est également possible de définir des caractéristiques de redimensionnement grâce au mode 9 images. Pour plus de détails, consultez "Fenêtre et image", page 24.
- être ***détourée*** : la fenêtre aura ainsi une forme originale (fenêtre en forme de cercle, d'ordinateur, ...). Pour plus de détails, consultez le paragraphe "Fenêtre détourée", page 26.
- avoir un ***menu déroulant*** : ce menu permettra aux utilisateurs d'accéder plus rapidement aux principales fonctionnalités de l'application. Pour créer un menu déroulant dans une fenêtre, sous le volet "Fenêtres", dans le groupe "Barres et menus", déroulez "Menu principal" et sélectionnez l'option "Ajouter le menu principal". La première option apparaît dans la fenêtre. Les options peuvent être manipulées grâce au menu contextuel des options. Les options de menu peuvent ensuite être manipulées par programmation. Pour plus de détails, consultez "Les menus", page 29.
- avoir une ***barre de message*** : cette barre de message affichera les messages d'aide associés aux différents champs de la fenêtre. Cette barre de message permet également d'afficher des informations supplémentaires. Pour plus de détails, consultez "La barre de message", page 33.
- être ***multilingue*** : toutes les langues définies pour la fenêtre seront disponibles pour tous les champs de la fenêtre. Une fenêtre peut être associée à plus de langues que le projet (cas par exemple des fenêtres partagées entre plusieurs projets). Pour plus de détails, consultez le chapitre "Applications multilingues", page 137.
- se ***griser*** automatiquement si elle devient inaccessible. Cette fonctionnalité s'appelle GFI pour Grisage des Fenêtres Inaccessibles. Pour plus de détails, consultez le paragraphe "Grisage automatique des fenêtres", page 27.
- autoriser le ***drag and drop***. Pour plus de détails sur le drag and drop et sa programmation, consultez l'aide en ligne.
- autoriser la ***modification de l'interface*** par l'utilisateur final. Pour plus de détails, consultez l'aide en ligne.

Remarque : Pour forcer une action sur une fenêtre, il suffit de définir un bouton de cette fenêtre en mode "Exécution automatique". Pour plus de détails, consultez le paragraphe "Exécution automatique d'un bouton", page 47.

3.2 Fenêtre de description

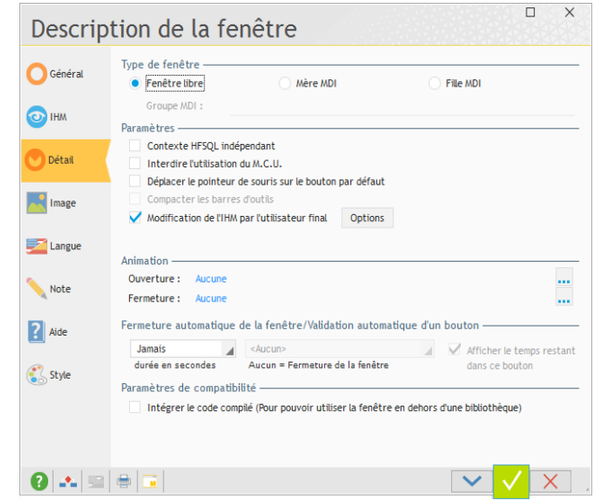
La fenêtre de description est une fenêtre à onglets, regroupant toutes les caractéristiques paramétrables de la fenêtre.

Remarque : Il est également possible de visualiser et/ou de modifier les caractéristiques d'une fenêtre dans le modifieur. Pour plus de détails, consultez l'aide en ligne.

3.3 Affichage des caractéristiques

Pour afficher la fenêtre de description :

- soit double-cliquez sur le fond de la fenêtre.
- soit sélectionnez l'option "Description" du menu contextuel de la fenêtre (clic droit de la souris).
- soit sélectionnez la fenêtre et utilisez le raccourci clavier [Alt] + [Entrée].



3.4 Caractéristiques par onglet

Ce paragraphe présente rapidement les différentes catégories de caractéristiques affichées par onglet.

Pour plus de détails sur chacune des caractéristiques présentes dans les différents onglets, consultez l'aide en ligne.

Onglet Général

L'onglet "Général" permet de spécifier le nom de la fenêtre et son titre.

Onglet IHM

L'onglet "IHM" permet de définir les différents paramètres de l'interface de la fenêtre :

- visibilité de la fenêtre,
- position de la fenêtre,
- gestion du clic droit, du curseur de la souris,
- association à un menu contextuel, ...

Onglet Détail

L'onglet "Détail" permet de définir les différents paramètres de la fenêtre :

- modication de l'interface par l'utilisateur final,
- animation,
- fermeture automatique.

Onglet Image

Cet onglet permet de paramétrer :

- l'image de fond de la fenêtre,
- l'image de la poignée de redimensionnement,
- l'icône de la fenêtre.

Onglet Langues

Cet onglet permet de connaître les langues gérées par la fenêtre, ainsi que la langue affichée sous l'éditeur. Il est possible d'ajouter, de modifier ou de supprimer des langues.

Onglet Notes

L'onglet "Notes" permet de décrire le fonctionnement détaillé de la fenêtre. Ces informations seront imprimées dans les dossiers de programmation (dossier du projet, dossier de la fenêtre, ...).

Onglet Aide

L'onglet "Aide" permet d'associer un fichier d'aide contextuelle à la fenêtre. Ce fichier d'aide contient les pages d'aide contextuelle affichées par chaque champ.

Onglet Style

L'onglet "Style" permet de définir le style de la fenêtre :

- Présence de la barre de titre.
- Présence des différentes icônes.
- Présence de la barre de message, ...

4 Fenêtre et image

4.1 Présentation

Pour optimiser le look de vos écrans, WINDEV permet d'utiliser :

- des couleurs de fond. De nombreuses couleurs standard peuvent être utilisées. La couleur de fond est configurée dans l'onglet "Style" de la fenêtre de description de la fenêtre.
- des images dans vos fenêtres. Ces images sont utilisées en image de fond de la fenêtre. Vos champs sont superposés sur cette image. La configuration de l'image de fond est réalisée dans l'onglet "Image" de la fenêtre.

Remarques :

- Si nécessaire, WINDEV offre également la possibilité d'utiliser le look et le thème XP/Vista dans vos fenêtres. Pour plus de détails, consultez l'aide en ligne.
- Pour personnaliser vos fenêtres, vous pouvez également utiliser un gabarit ou un modèle de fenêtres.

4.2 Configuration de l'image de fond

La configuration est effectuée dans l'onglet "Image" de la description de la fenêtre.

Vous pouvez sélectionner directement le fichier correspondant à l'image de fond ou sélectionner une image du catalogue. Les principaux formats d'image pouvant être affichés sont les suivants : BMP, JPEG, TIFF ou GIF.

Il est également possible de réaliser directement un Drag and Drop du fichier image présent dans l'explorateur vers la fenêtre WINDEV.

Les options possibles sur l'image sont :

- Gestion des images multilingues.
- Mode d'affichage.
- Position.
- Transparence.
- Symétrie.
- Mode 9 images.
- Détourage de la fenêtre selon l'image de fond.

Gestion des images multilingues

Une image de fond différente peut être affichée selon la langue d'exécution du programme. Les caractéristiques d'affichage peuvent être spécifiques à chaque langue.

Pour définir une image de fond multilingue :

1. Cliquez sur l'icône présente à droite du nom de l'image de fond.
2. Sélectionnez l'option "Multi-langues".
3. Pour chaque langue définie dans votre fenêtre, choisissez l'image appropriée.
4. Validez. Le libellé "Valeur multilangue" apparaît dans le champ de définition des images de fond.

Mode d'affichage

Une image peut être affichée dans une fenêtre selon plusieurs modes d'affichage. Ces modes d'affichage permettent d'obtenir différents effets :

- **100%** : l'image est affichée à 100% à partir du coin haut gauche du champ Image.
- **Centré** : l'image est affichée à 100% et centrée dans le champ Image.
- **Etiré** : l'image est déformée pour occuper tout l'espace du champ Image.
- **Répété** : l'image est affichée à 100% et répétée dans le champ Image pour occuper tout l'espace du champ.
- **Homothétique** : l'image est agrandie proportionnellement pour être affichée entièrement dans le champ Image.
- **Homothétique étendu** : l'image est proportionnellement agrandie pour que le plus petit côté de l'image soit affiché entièrement dans le champ Image.
- **Homothétique centré** : l'image est homothétique et centrée dans le champ Image.
- **Homothétique centré étendu** : l'image est étendue et centrée dans le champ Image.
- **Homothétique sans agrandissement** : si l'image peut être affichée entièrement dans la fenêtre, elle est affichée à 100%. Si l'image est plus grande que la fenêtre, elle est automatiquement réduite proportionnellement pour être affichée entièrement.
- **Homothétique centré sans agrandissement** : l'image est centrée et homothétique sans agrandissement.

L'option "Affichage Haute Qualité" permet en mode homothétique d'améliorer la qualité de l'image lorsque celle-ci est réduite par rapport à sa taille initiale.

Position

Ce paramètre permet de définir la position du point haut gauche de l'image par rapport à la surface de la fenêtre.

Ces options seront conservées même si l'image est modifiée par programmation (propriété ***..Image***).

Transparence

Les options de transparence de l'image de fond sont les suivantes :

- **Magenta transparent** : Le magenta présent dans l'image sera transparent à l'affichage sous l'éditeur comme en exécution.

Transparence : aucune | Magenta transparent

- **Point haut gauche** : tous les points ayant la même couleur que celle du point haut gauche de l'image seront transparents.

Ces options seront conservées même si l'image est modifiée par programmation (propriété ***..Image***).

Symétrie

Les options de symétrie permettent de changer l'orientation de l'image contenue dans le fichier image. Ces options seront conservées même si l'image est modifiée par programmation (propriété ***..Image***).

Image originale :

Les options sont les suivantes :

Symétrie verticale	Symétrie horizontale	Symétrie horizontale et verticale

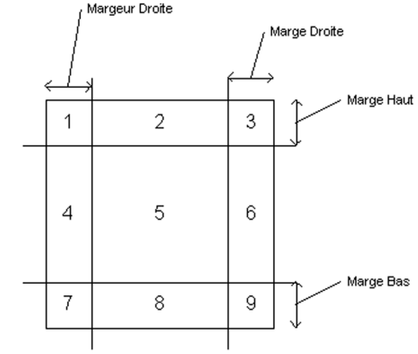
Mode 9 images

Si cette option est sélectionnée, le mode 9 images sera utilisé sur l'image de fond de la fenêtre.

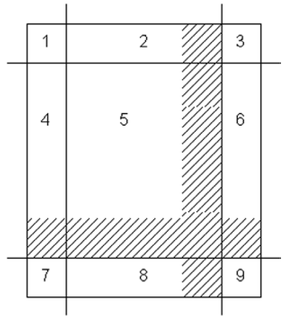
Le mode 9 images consiste à découper l'image de fond en 9 zones.

Pour chaque zone, il est possible de définir si la partie de l'image correspondant à la zone doit être affichée ou non (il suffit de cocher le nom de la zone pour l'afficher). Si la partie de l'image n'est pas affichée, elle sera remplacée par la couleur de fond de la fenêtre.

Les marges permettent de définir la taille de chaque zone.



Lors du redimensionnement de la fenêtre, seules les zones 2, 4, 6, 8 et 5 changent de taille pour occuper tout l'espace :



Le mode 9 images permet de gérer le fonctionnement de chaque zone lors du redimensionnement de la fenêtre.

Pour chaque zone :

- soit en répétant l'image qui compose la zone,
- soit en étirant l'image qui compose la zone.

4.3 Fenêtre détourée

WINDEV permet de détourer les fenêtres. Les fenêtres auront ainsi des formes originales (fenêtre en forme de cercle, d'ordinateur, ...).

La forme de la fenêtre est définie par une image. Dans cette image, utilisez une couleur spécifique pour définir les zones transparentes de l'image (par exemple, la couleur "Magenta Clair"). Toutes ces zones seront invisibles lors de l'affichage de la fenêtre en exécution : la fenêtre sera alors détourée.

Attention : Les bordures de cette image doivent être les plus nettes possible (pas de dégradé, ...) afin de ne pas obtenir d'effets d'affichage lors de l'aperçu ou de l'ouverture de la fenêtre détourée.

La configuration du détourage de la fenêtre est effectuée dans l'onglet "Image" de la fenêtre de description de la fenêtre.

5 Ouvrir une fenêtre "popup"

5.1 Présentation

WINDEV permet d'ouvrir des fenêtres de type "popup".

Une fenêtre "popup" correspond à une fenêtre de taille réduite s'ouvrant au-dessus de la fenêtre en cours. Généralement, cette fenêtre permet à l'utilisateur de réaliser une sélection. Cette fenêtre renvoie donc à la fenêtre principale la valeur sélectionnée par l'utilisateur. Cette fenêtre se ferme automatiquement dès qu'un clic gauche de souris est effectué (à l'intérieur ou non de la fenêtre popup).

Par exemple :

Remarque : Les caractéristiques d'une fenêtre popup sont les suivantes :

- ouverture par défaut sous le champ qui ouvre la fenêtre (ou au-dessus ou à gauche du champ si la place disponible sous le champ est insuffisante). La position d'ouverture peut être modifiée par programmation.
- fermeture automatique dès que la fenêtre n'a plus le focus ou dès l'appel de la fonction ***Ferme***.
- seule l'icône de fermeture apparaît dans la barre de titre de la fenêtre (si la barre de titre a été définie dans le style).

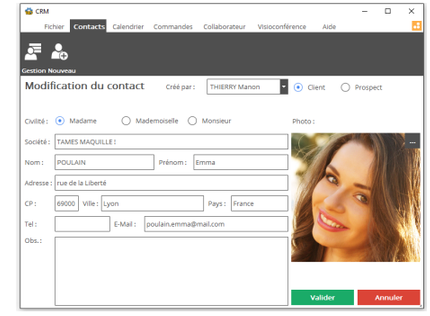
6 Grisage automatique des fenêtres

Votre application ouvre plusieurs fenêtres simultanément et vous ne savez plus quelle fenêtre est celle à utiliser ?

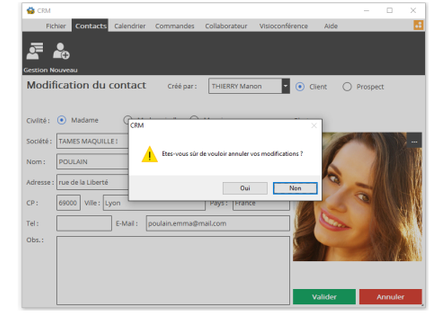
La ou les fenêtres inactives peuvent se griser automatiquement. Ainsi, vous devinez instinctivement la fenêtre à utiliser.

Cette fonctionnalité s'appelle GFI pour Grisage des Fenêtres Inaccessibles.

Par exemple :



Etape 1 : Vous modifiez des informations concernant un client puis vous cliquez sur le bouton "Valider".



Etape 2 : Une fenêtre vous demande de confirmer ou d'annuler ces modifications. La fenêtre principale (en arrière-plan) est devenue inaccessible. Elle s'est grisée automatiquement. Vous savez immédiatement quelle fenêtre doit être utilisée.

Les fonctionnalités du GFI sont disponibles au niveau du projet ou personnalisables au niveau de chaque fenêtre.

Il est également possible d'agir sur le GFI par programmation.

Pour plus de détails, consultez l'aide en ligne.

7 Fenêtre interne

Les fenêtres internes permettent de partager dynamiquement une même partie d'interface au sein d'une ou de plusieurs applications.

L'interface à utiliser plusieurs fois est créée dans une fenêtre de type "Fenêtre Interne".

Cette interface est utilisée dans les différentes fenêtres de votre application grâce au champ de type "Fenêtre interne".

Remarque : La fenêtre à fusionner peut être issue d'un composant.

Pour plus de détails, consultez l'aide en ligne.

8 Modèles de fenêtres

WINDEV permet de créer des modèles de fenêtres. Ces modèles contiennent tous les éléments graphiques et le code communs à l'ensemble des fenêtres de votre application.

Les modifications effectuées dans un modèle de fenêtres sont automatiquement reportées sur toutes les fenêtres utilisant ce modèle.

Un modèle de fenêtres permet de respecter la charte graphique définie pour une application.

L'utilisation de modèles de fenêtres dans vos applications permet de :

- simplifier la création des fenêtres de l'application.
- simplifier la mise en page des fenêtres de l'application.
- simplifier la mise à jour de la charte graphique de l'application.

Un modèle de fenêtres se crée aussi simplement qu'une fenêtre. Pour créer une fenêtre en utilisant un modèle, il suffit de choisir l'option "basée sur un modèle".

Par défaut, toute modification effectuée dans le modèle est reportée dans les fenêtres l'utilisant. Il est cependant possible dans une fenêtre spécifique de gérer des cas particuliers en surchargeant les éléments du modèle.

Pour plus de détails, consultez l'aide en ligne.

9 Traitements associés aux fenêtres

9.1 Traitements gérés par défaut

WINDEV gère par défaut les traitements suivants (ordre d'apparition sous l'éditeur de code) :

- **Déclarations globales** :
  Déclaration des variables globales à la fenêtre
- **Fin d'initialisation** :
  Exécuté lors de l'ouverture de la fenêtre, avant le code de prise de focus. Le traitement d'initialisation des champs de la fenêtre est exécuté après ce code.
- **Fermeture** :
  Exécuté lors de la fermeture de la fenêtre. Remarque : une reprise de saisie sur un des champs de la fenêtre dans ce code empêche la fenêtre de se fermer.
- **Demande de mise à jour de l'affichage :**
  Permet de regrouper le code de mise à jour des champs de la fenêtre : contenu des champs, état (visible, grisé, ...), libellé à mettre à jour, ...
  Ce traitement est lancé sur demande grâce aux fonctions :
  - ***DemandeMiseAJourIHM*** : Exécute le traitement à la fin du traitement en cours (lancement asynchrone).
  - ***ExécuteMiseAJourIHM*** : Exécute immédiatement le traitement.
- **Affectation de la propriété ..Valeur (fenêtre interne uniquement)**
  Exécuté lors de l'utilisation de la propriété ..Valeur en affectation sur le champ Fenêtre Interne.
- **Récupération de la propriété ..Valeur (fenêtre interne uniquement)**
  Exécuté lors de l'utilisation de la propriété ..Valeur en lecture sur le champ Fenêtre Interne.

9.2 Traitements optionnels

Plusieurs traitements optionnels peuvent être gérés. Il suffit de :
1. Cliquer sur les "..." présents dans le traitement "Autres traitements" de l'éditeur de code.
2. La fenêtre des traitements optionnels s'affiche.
3. Sélectionner le ou les traitements optionnels à ajouter et valider.
4. Les traitements sélectionnés sont automatiquement ajoutés à la suite des traitements proposés par défaut.

Il est ainsi possible de gérer par exemple :

- la prise et la perte de focus,
- la modification d'un des champs de la fenêtre,
- la modification de la taille de la fenêtre,
- le survol de la fenêtre par la souris,
- le bouton gauche enfoncé, relâché, avec double-clic,
- le bouton droit enfoncé, relâché, avec double-clic,
- la roulette de la souris, ...

10 Les menus contextuels

Un menu contextuel peut être ouvert par l'utilisateur grâce à un clic droit de la souris. Ce menu permet aux utilisateurs d'accéder plus facilement aux fonctionnalités spécifiques d'un champ d'une fenêtre ou d'une fenêtre.

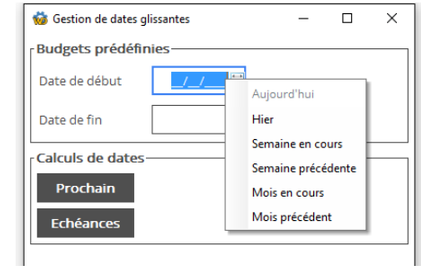
WINDEV permet de gérer deux types de menu contextuel :

- **les menus contextuels système** gérés par Windows. Ces menus contextuels s'affichent lors d'un clic droit sur certains types de champs (champ de saisie, table, ...). Pour plus de détails, consultez le paragraphe "Menu contextuel système par défaut", page 30.
- **les menus contextuels personnalisés** gérés par WINDEV. Ces menus sont créés sous l'éditeur de fenêtres et peuvent être manipulés par programmation.

WINDEV permet de créer des menus contextuels personnalisés et d'associer ces menus contextuels :

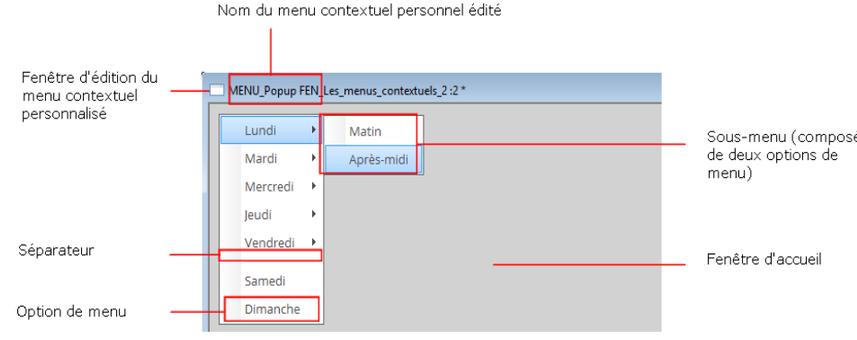
- soit à un ou plusieurs champs d'une fenêtre.
- soit à une fenêtre.

Par exemple :



10.1 Fenêtres d'édition des menus contextuels

WINDEV permet de créer et de modifier des menus contextuels personnalisés dans une fenêtre d'édition particulière. Pour mieux comprendre la gestion des menus contextuels personnalisés, voici le vocabulaire lié à cette fenêtre d'édition :



10.2 Fonctionnalités disponibles

WINDEV permet de :

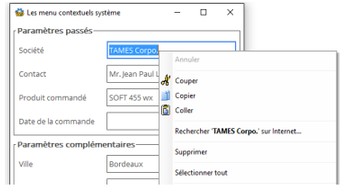
- Créer un menu contextuel personnalisé.
- Associer un menu contextuel personnalisé existant à un ou plusieurs éléments (fenêtre ou champs).
- Editer un menu contextuel personnalisé.
- Afficher la fenêtre de description d'un menu contextuel personnalisé.
- Supprimer un menu contextuel personnalisé.

Pour plus de détails, consultez l'aide en ligne.

10.3 Menu contextuel système par défaut

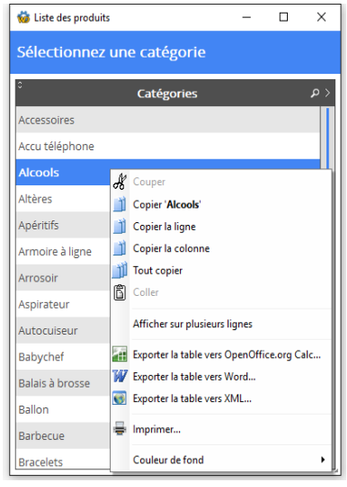
Les types de champ suivants ont un menu contextuel système par défaut (affiché si aucun menu contextuel personnalisé n'est associé au champ) :

- un champ de saisie :



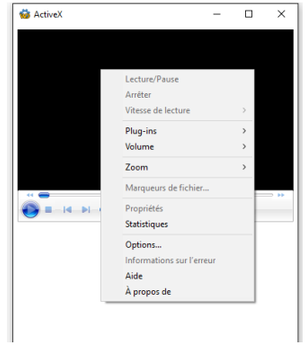
Pour plus de détails sur la traduction de ce menu, consultez l'aide en ligne.

- une colonne de table :



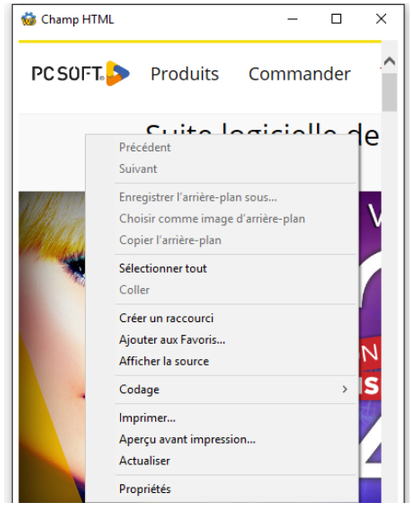
Pour plus de détails sur la traduction de ce menu, consultez l'aide en ligne.

- un champ ActiveX :



Remarque : Le menu contextuel système affiché dépend de l'ActiveX associé au champ.

- un champ HTML :



Remarque : Le menu contextuel système affiché est celui d'Internet Explorer. Il dépend de l'objet sélectionné dans la page (image, texte, lien, ...).

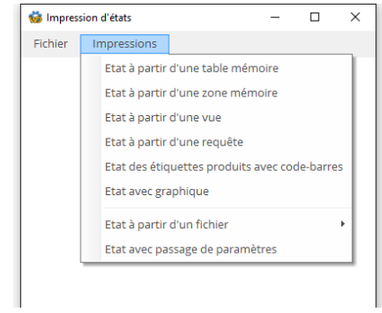
Ce menu contextuel par défaut peut être désactivé aussi bien depuis l'éditeur que par programmation.

11 Les menus principaux

11.1 Présentation

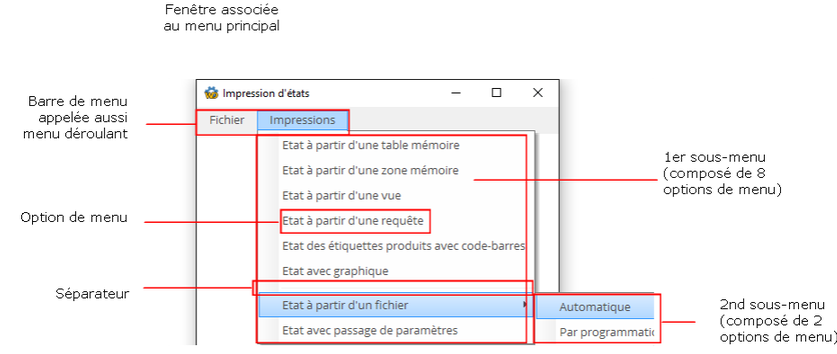
Un menu principal est obligatoirement associé à une fenêtre. Ce menu est affiché sous la barre de titre de la fenêtre sous WINDEV. En règle générale, les applications contiennent une fenêtre menu. Cette fenêtre contient uniquement un menu principal et permet d'accéder aux autres fenêtres de l'application.

Par exemple :



11.2 Vocabulaire associé aux menus principaux

Pour manipuler simplement un menu principal, il est nécessaire de connaître le vocabulaire suivant :



11.3 Fonctionnalités disponibles

WINDEV permet de :

- Créer un menu principal.
- Editer un menu principal.
- Supprimer un menu principal.

Pour plus de détails, consultez l'aide en ligne.

Cas particulier : Menu utilitaire

WINDEV permet d'insérer automatiquement un menu "utilitaire" (de type "?") dans votre application. Sous le volet "Fenêtre", dans le groupe "Barres et menus", déroulez "Menu principal" et sélectionnez l'option "Ajouter le menu '?'".

12 Les options de menu

12.1 Présentation

Un menu (contextuel personnalisé ou principal) est composé d'une ou de plusieurs options et sous-options. Chaque option permet d'exécuter un code WLangage.

Les manipulations des options de menu sous l'éditeur sont identiques que le menu soit un menu principal ou un menu contextuel personnalisé.

Pour manipuler les options de menu, il suffit d'éditer le menu concerné.

Pour éditer (ou afficher sous l'éditeur) :

- le menu principal d'une fenêtre :
  - ouvrez la fenêtre concernée sous WINDEV.
  - sélectionnez une des options de menu. Les options et les sous-options s'affichent automatiquement.
- un menu contextuel personnalisé d'une fenêtre : sous le volet "Fenêtre", dans le groupe "Barres et menus", déroulez "Menus contextuels" et sélectionnez l'option "<NomMenuContextuel>".

12.2 Manipuler les options de menu sous l'éditeur

L'éditeur de fenêtres permet de réaliser très simplement les manipulations suivantes :

- Ajouter une option de menu.
- Ajouter un séparateur.
- Ajouter un sous-menu.
- Supprimer une option de menu.

Les options de menu peuvent également être directement manipulées sous l'éditeur de fenêtres et leurs caractéristiques sont modifiables grâce à une fenêtre de description.

La fenêtre de description des options de menu permet par exemple de :

- modifier le libellé de l'option.
- cocher ou décocher une option de menu.
- associer une image à une option de menu.
- associer un raccourci clavier à une option de menu.

Il est également possible d'associer du code WLangage à une option de menu. Seules les options de menu ne possédant pas de sous-menu peuvent lancer un traitement WLangage.

Pour plus de détails, consultez l'aide en ligne.

12.3 Manipuler les options de menu par programmation

La manipulation d'une option de menu est identique que le menu soit un menu contextuel ou un menu principal.

Pour manipuler par programmation une option de menu, utilisez la notation :

- ***"<NomOption>"*** si l'option de menu est manipulée depuis la fenêtre.
- ***"<NomFenêtre>.<NomMenu>.<NomOption>"*** si l'option est manipulée depuis une autre fenêtre.

13 La barre de message

Il est possible d'afficher dans la barre de message d'une fenêtre :

- un ***message d'information lié au champ en cours***. Ce message est saisi dans l'onglet "Aide" de la fenêtre de description du champ. Le message sera affiché dans la barre de message lorsque l'utilisateur sélectionnera le champ.
- un ***message d'information spécifique***. Ce message est affiché par programmation grâce à la fonction ***Message***.
- une ***jauge de progression***. Cette jauge est affichée par programmation grâce à la fonction ***Jauge***.

Pour afficher simultanément diverses informations, il est possible de diviser la barre de message en plusieurs cases (option "Insérer une case" dans le menu contextuel de la barre de message).

13.1 Fenêtre de description d'une barre de message

La fenêtre de description d'une barre de message permet de :

- ajouter et/ou supprimer des cases.
- modifier l'ordre des cases.
- spécifier pour chaque case les informations à afficher.

Pour plus de détails, consultez l'aide en ligne.

13.2 Type d'une case de la barre de message

Le type de la case permet d'afficher :

- La jauge de progression. Cette jauge sera exécutée lors de l'utilisation de la fonction ***Jauge***.
- Le message d'aide associé au champ ayant le focus. Ce message d'aide est défini dans l'onglet "Aide" du champ ou par programmation grâce à la propriété ***..Message***.
- La date du poste en cours.
- L'heure du poste en cours.
- La position du curseur dans le champ de saisie en cours (option Colonne et Ligne du Caret).
- L'état de la touche [MAJ], indiquant si la touche de verrouillage des majuscules est activée ou non.
- L'état de la touche [INSER], indiquant si la touche [Inser] est activée ou non.
- Le nom de l'utilisateur actuel dans une application utilisant le groupware utilisateur.

Deux options spécifiques sont également disponibles :

- Par programmation : Les informations affichées dans la barre de message sont spécifiées par programmation (fonction ***Message*** par exemple).
- Traitement de mise à jour : Le traitement "A chaque mise à jour" est automatiquement associé à la case de la barre de message. Ce traitement est alors exécuté régulièrement. Ce traitement peut par exemple contenir les éléments à afficher dans la case.

13.3 Caractéristiques d'une case de la barre de message

La fenêtre de description de la barre de message permet également de définir les différentes caractéristiques d'affichage des cases de la barre de message.

- La position de la case dans la barre de message.
- L'image de fond de la case.
- La couleur de fond de la case.
- La largeur de la case.
- Si la case est en saisie.
- L'ancrage de la case (son fonctionnement lors du redimensionnement de la fenêtre).

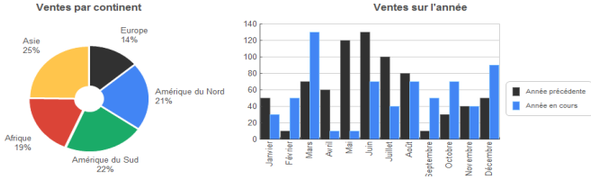
Toutes ces caractéristiques peuvent être manipulées par programmation grâce aux propriétés disponibles sur les cases des barres de message.

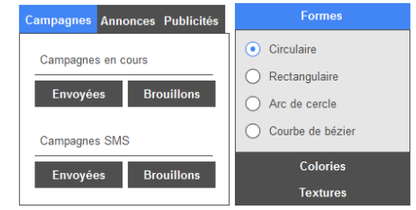
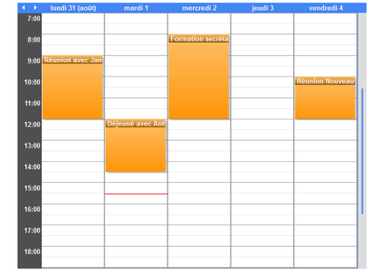
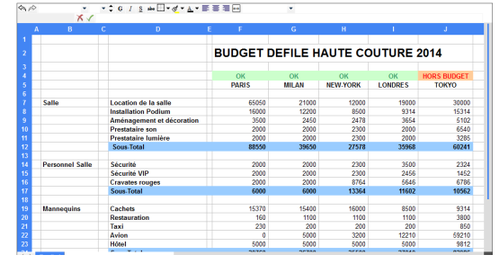
Les différents types de champs standard

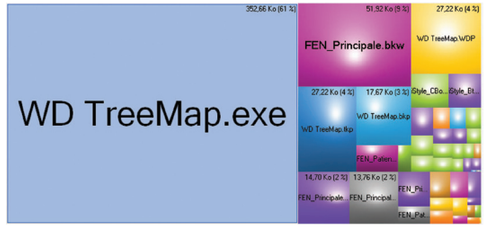
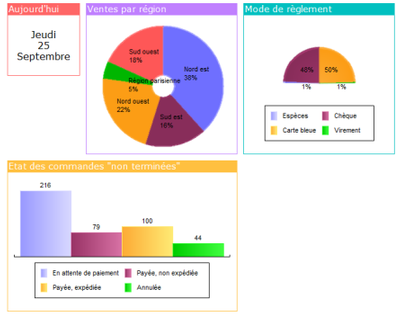
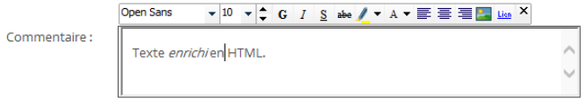
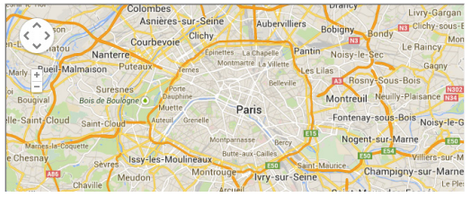
WINDEV met à votre disposition de nombreux champs livrés en standard. Voici comment les utiliser en fonction de l'opération à réaliser :

Vous voulez...	Utilisez un champ...
Afficher du texte, un titre, ... Un libellé simple. Un libellé **RTF** avec des enrichissements.	Libellé
Lire un fichier PDF	Lecteur PDF
Afficher et modifier un document Word.	Traitement de texte
Sélectionner une valeur parmi une liste (Pays, Ville, ...) Civilité : Monsieur / Madame / Mademoiselle Civilité : Monsieur Civilité : Monsieur / Madame / Mademoiselle	Sélecteur, Combo, Liste
Sélectionner plusieurs valeurs parmi une liste (les destinataires d'un message, ...) Carrosserie : Berline, Coupé, Monospace, Décapotable Carrosserie : Berline, Camping-Car, Coupé, 4x4, Monospace, Break, Décapotable	Interrupteur, Liste
Sélectionner une ou plusieurs valeurs parmi une liste graphique (Trombinoscope, ...) Anna, Charlotte, Daniela, Manon	Liste Image
Sélectionner une valeur dans une plage prédéfinie Volume :	Potentiomètre, Potentiomètre rotatif Spin, Range Slider

Sélectionner une date septembre 2017 L M M J V S D 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 1 2 3 4 5 6 7 8 Aujourd'hui	Calendrier
Afficher une image (Photo, ...), une animation ou utiliser des fonctions de dessin	Image
Afficher le contenu d'un fichier de données en table (liste de clients, détail d'une commande, ...). Nom et Prénom / Téléphone / Photo BORSANI Sarah — Tel. : 01 02 03 04 05 / Mob. : 01 02 03 04 05 / Fax : 01 02 03 04 05 BRODIER Céline — Tel. : 01 02 03 04 05 / Mob. : 01 02 03 04 05 / Fax : 01 02 03 04 05 BUN Muy-Ly — Tel. : 01 02 03 04 05 / Mob. : 01 02 03 04 05 / Fax : 01 02 03 04 05	Table
Répéter des champs dans une fenêtre (Catalogue produits avec la photo, ...) Bora Bora — Polynésie Française — 3952 € Bora Bora — Polynésie Française — 3053 € Sud de Ari — Maldives — 3009 € Malé — Maldives — 1972 € Malé — Maldives — 1899 € Miami — Floride — 2699 € Punta Cana — République Mahé — Les Seychelles Phukets — Thaïlande	Zone répétée

Afficher une vidéo, une animation 	Web Caméra Multimédia Conférence
Saisir des informations 	Champ de saisie
Programmer une action dans une fenêtre (afficher une autre fenêtre, lancer une impression, ...) 	Bouton
Afficher des informations hiérarchisées (répertoire) 	Arbre Table hiérarchique
Afficher un histogramme, une courbe, un camembert, ... 	Graphe
Afficher une progression 	Jauge

Regrouper les champs par thème et afficher les thèmes un par un 	Onglet Boîte à outils Tiroir
Afficher un code-barres 	Code-barres
Afficher des événements sur une période de temps 	Agenda
Réaliser des tableaux statistiques 	Tableau croisé dynamique Tableur

Représenter des données de manière pondérée 	TreeMap
Afficher les éléments importants sous forme de tableau de bord 	Tableau de bord
Saisir ou afficher des données HTML 	Champ de saisie HTML Champ HTML
Permettre la sélection d'intervalle ou de zoom	Range slider
Afficher une carte, un plan, un itinéraire 	Carte
Editer une image	Editeur d'images
Afficher un contrôle ActiveX	ActiveX

Les champs en pratique

WINDEV propose de nombreux types de champs, facilement intégrables dans vos fenêtres. Ces champs peuvent être créés directement par le volet "Création" du ruban WINDEV.

Tous les champs peuvent bien sûr être manipulés par programmation.

1 Créer un champ

1.1 Les différents types de champs d'une fenêtre

Les champs disponibles sous l'éditeur de fenêtres de WINDEV sont les suivants :

- Champ de saisie,
- Libellé,
- Bouton,
- Image,
- Editeur d'images,
- Sélecteur,
- Interrupteur,
- Liste,
- Combo,
- Liste image,
- Boîte à outils,
- Tiroir,
- Web Caméra,
- Champ Conférence,
- Champ OLE,
- ActiveX,
- Forme,
- Code-barres,
- Champ HTML,
- Superchamp,
- Agenda,
- Planning,
- Notation,
- Tableau croisé dynamique,
- Tableau de bord,
- Table,
- Tableur,
- Traitement de texte,
- Lecteur de PDF,
- Diagramme de Gantt,
- Table hiérarchique,
- Arbre,
- Organigramme,
- Jauge,
- Zone répétée,
- Zone multiligne,
- Ascenseur,
- Potentiomètre,
- Spin,
- TimeLine,
- Onglet,
- Ruban,
- Champ fenêtre interne,
- Barre d'outils,
- Séparateur,
- Graphe,
- Carte,
- Carrousel,
- Cube,
- Widget,
- Xaml,
- Calendrier,
- Fenêtre interne,
- Modèle de champs,
- Champ Disposition.

Tous les champs peuvent bien sûr être manipulés par programmation.

Pour plus de détails sur chaque type de champ, leurs principales caractéristiques, ainsi que leur programmation en WLangage, consultez l'aide en ligne.

1.2 Créer un nouveau champ

Pour créer un champ :

1. Sélectionnez le type de champ à créer grâce à l'icône correspondante dans le volet "Création" du menu de WINDEV.

2. La forme du nouveau champ apparaît sous le curseur de la souris.

3. Cliquez dans la fenêtre à la future position de création du champ. Le champ est automatiquement créé.

Remarque : D'autres modes de création sont disponibles. Pour plus de détails, consultez l'aide en ligne.

1.3 Créer un champ associé à une rubrique

Tous les champs d'une fenêtre peuvent être associés à une rubrique de fichier de données, SAUF :

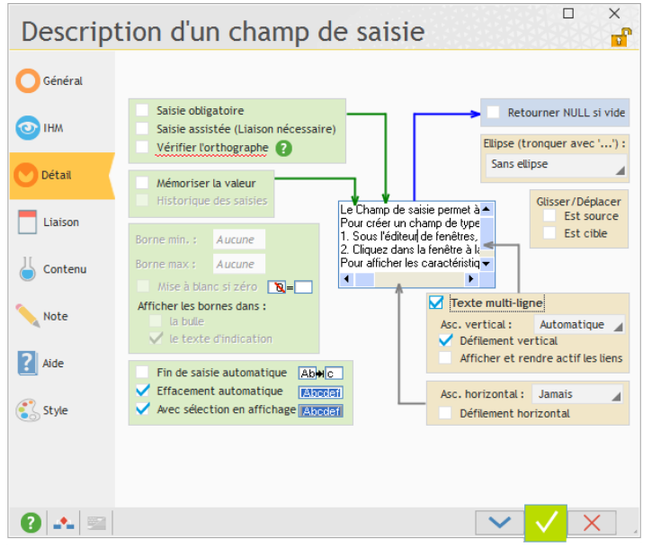
- Bouton
- Forme
- Onglet
- Barre d'outils
- Carte
- Ruban
- Séparateur
- TimeLine
- Tableau de Bord
- Xaml
- Carrousel
- Cube
- Champ HTML
- Champ Disposition

Pour créer un champ associé à une rubrique (ou récupérer une rubrique), plusieurs méthodes sont disponibles :

- Drag and Drop depuis le volet "Analyse".
- Utilisation de l'onglet "Fichier" de la fenêtre de description du champ.

2 Caractéristiques d'un champ

Pour tous les champs d'une fenêtre, il est possible d'afficher la fenêtre de description. Cette fenêtre est une fenêtre à onglets, regroupant toutes les caractéristiques paramétrables d'un ou de plusieurs champs.



Remarque : Il est également possible de visualiser et/ou de modifier les caractéristiques d'un ou de plusieurs champs dans le modifieur. Pour plus de détails, consultez l'aide en ligne.

2.1 Affichage des caractéristiques

Pour afficher la fenêtre de description d'un champ :

- soit double-cliquez sur le champ.
- soit sélectionnez l'option "Description" du menu contextuel du champ (clic droit de la souris).
- soit sélectionnez le champ et utilisez le raccourci clavier [Alt] + [Entrée].

Remarques :

- Il est possible d'afficher la fenêtre de description pour un ensemble de champs sélectionnés. Seules les caractéristiques communes aux différents champs sélectionnés seront affichées.
- Plusieurs fenêtres de description peuvent être affichées simultanément. Chaque fenêtre de description affiche alors les caractéristiques d'un ou de plusieurs champs.

2.2 Caractéristiques par onglet

Ce paragraphe présente rapidement les différentes catégories de caractéristiques affichées par onglet.

Pour plus de détails sur chacune des caractéristiques présentes dans les différents onglets, consultez l'aide contextuelle des fenêtres de description.

Onglet Général

L'onglet "Général" permet de spécifier le nom du champ et toutes les caractéristiques d'affichage du champ (libellé, masque de saisie, ...).

Onglet IHM

L'onglet "IHM" permet de définir les différents paramètres de l'interface du champ :

- Etat initial du champ à l'ouverture de la fenêtre
- Visibilité du champ
- Taille du champ
- Cadrage, ...

Onglet Détail

L'onglet "Détail" permet de définir les différents paramètres du champ :

- paramètres de saisie
- Drag and Drop, ...

Le contenu de cet onglet varie beaucoup en fonction du type de champ en cours.

Onglet Liaison

Cet onglet permet de sélectionner la rubrique (d'un fichier de données ou d'une rubrique) à laquelle le champ est relié. Selon l'enregistrement en cours, le contenu de la rubrique reliée sera affiché dans le champ.

La liaison peut être mono-fichier ou multi-fichier.

Onglet Contenu

Cet onglet est disponible uniquement pour les champs des fenêtres.

L'onglet "Contenu" permet de définir :

- soit le contenu initial du champ (pour les champs de saisie uniquement).
- soit la source de données utilisée pour remplir le champ (pour les listes, les combos et les tables uniquement).

Onglet Notes

L'onglet "Notes" permet de décrire le fonctionnement détaillé du champ. Ces informations seront imprimées dans les dossiers de programmation (dossier du projet, dossier de la fenêtre, ...).

Onglet Aide

L'onglet "Aide" permet de saisir et de configurer tous les types d'aides associées au champ. Un champ peut avoir :

- Une bulle d'aide, affichée lors du survol du champ.
- Un message d'aide, affiché dans la barre de message lors de l'entrée dans le champ.
- Une fenêtre d'aide avec un titre spécifique, affiché lors de l'utilisation de la touche F1.

Un champ peut être également associé à une page d'aide spécifique, présente dans un fichier d'aide au format CHM. Cette page d'aide sera affichée :

- soit sous forme d'aide contextuelle (bulle d'aide affichée sur demande), si un fichier d'aide a été défini pour la fenêtre.
- soit sous forme de page d'aide. La fonction ***WHelp*** devra être utilisée pour afficher la page d'aide.

Onglet Style

L'onglet "Style" permet de définir le style des différents éléments du champ. Cet écran permet de :

- Modifier l'aspect d'un champ : il suffit alors de sélectionner l'élément du champ à modifier, puis ses caractéristiques de style. Seul l'aspect du champ en cours est modifié.
- Appliquer un style existant au champ : il suffit de sélectionner le style à appliquer.
- Créer ou modifier un style : le style créé ou modifié est mis à jour pour tous les champs du projet utilisant ce style.

3 Manipuler les champs sous l'éditeur

3.1 Les poignées des champs

Lorsqu'un champ vient d'être créé ou lorsqu'il est sélectionné, il s'affiche avec des poignées. Les poignées permettent de :

- visualiser la taille d'un champ,
- modifier la taille d'un champ.

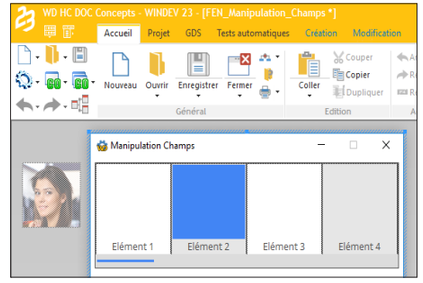
Les couleurs des poignées permettent de connaître les actions réalisables :

- Les ***poignées noires*** permettent de visualiser et de redimensionner un champ.
- Les ***poignées blanches*** permettent uniquement de visualiser un champ. Par exemple, lors de la sélection de plusieurs champs, les poignées blanches indiquent qu'il est impossible de modifier la taille des champs sélectionnés. Ce type de poignées est aussi affiché lorsqu'une fenêtre est en lecture seulement pour le développeur : les champs ne peuvent pas être modifiés.
- Les ***poignées grises*** indiquent que le premier champ sélectionné lors d'une sélection multiple. Ce champ sera le champ de référence.

3.2 Manipulations disponibles

L'éditeur de fenêtres permet de :

- Sélectionner un ou plusieurs champs (par exemple à l'aide du lasso)
- Regrouper des champs sélectionnés
- Déplacer un champ
- Modifier le libellé d'un champ
- Afficher une bulle d'aide avancée lors du survol d'un champ, contenant : le nom du champ, sa position, sa taille, son ancrage, son état initial.
- Utiliser un champ hors fenêtre. Ce champ sera visible uniquement en édition et ne sera pas visible en exécution.



Remarque : Un champ "hors fenêtre" permet par exemple de :

- conserver un champ d'une fenêtre sans le visualiser lors de l'exécution. Il est ainsi possible d'utiliser le contenu du champ en programmation (comme une variable).
- afficher de nouveaux champs dans une fenêtre lors de la modification de la taille de cette fenêtre.

4 Alignement des champs

L'alignement des champs permet d'obtenir des interfaces "professionnelles" et harmonieuses. Pour vous aider à concevoir des interfaces standard, WINDEV vous propose les outils suivants :

- les règles.
- la grille.
- le correcteur d'interface en temps réel (proposé automatiquement lors d'un positionnement d'un champ).
- le correcteur d'interface avancé.
- les options d'alignement.
- le zoom paramétrable.

4.1 Les règles

Sous l'éditeur de fenêtres, il est possible de faire apparaître des règles d'alignement.

Dans ces règles, des guides magnétiques sont affichés : tout champ approché d'un guide est automatiquement "attiré" par celui-ci.

Ce magnétisme permet d'utiliser les guides pour faciliter le positionnement, l'alignement ou le redimensionnement des champs de la fenêtre.

Afficher les règles

Pour afficher les règles, sous le volet "Affichage", dans le groupe "Aide à l'édition", cochez l'option "Règles". Deux types de guides peuvent alors être utilisés :

- les guides, permettant d'aligner et de disposer les champs à l'intérieur des fenêtres.
- les guides de bordure, permettant de déterminer une bordure de taille identique de chaque côté de la fenêtre.

Remarque : L'option "Guide visible" du menu contextuel des règles permet de visualiser la fenêtre avec ou sans ses guides. Visibles ou non, les guides sont toujours actifs : un champ déplacé vers un repère est automatiquement accolé à celui-ci.

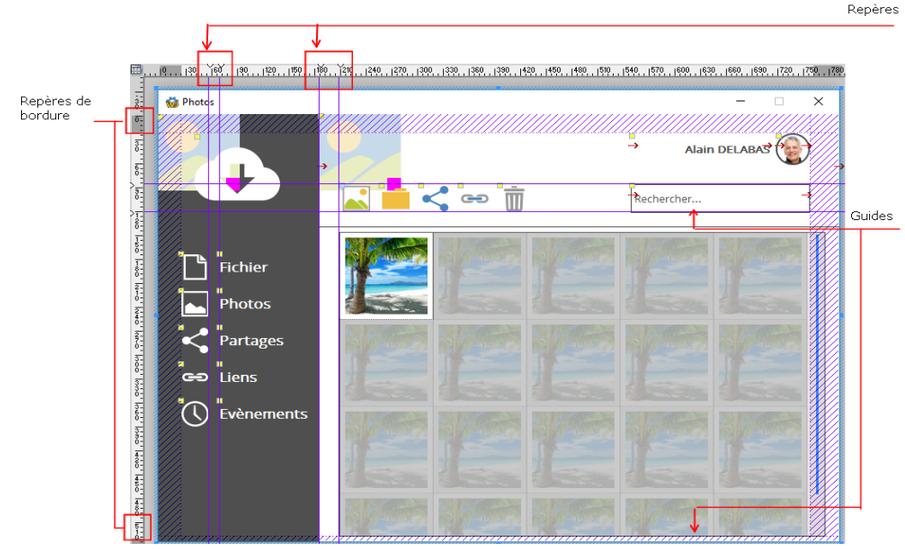
Manipuler les repères

Les repères sont facilement manipulables à l'aide de la souris dans l'éditeur. Lorsque la souris survole un repère, le curseur se transforme en une double flèche.

4.2 Grille magnétique

La grille magnétique permet de placer des repères (verticaux et horizontaux) dans la fenêtre. Les champs créés viennent se placer contre ces repères, comme si les champs et les repères étaient magnétisés. La grille magnétique permet ainsi d'aligner les champs selon les repères.

Par exemple, dans le cas d'une fenêtre :



4.3 Correcteur d'interface en temps réel

Lors du déplacement d'un champ dans une fenêtre, le correcteur d'interface en temps réel affiche automatiquement des guides. Ces guides sont magnétisés et permettent d'aligner le champ déplacé avec les champs présents dans la fenêtre.

4.4 Correcteur d'interface avancé

Le correcteur d'interface avancé est un outil permettant d'harmoniser la disposition des champs dans les différentes fenêtres d'une application.

Le correcteur d'interface avancé propose pour la fenêtre en cours ou uniquement pour les champs sélectionnés, d'appliquer des règles de présentation d'interface issues du standard Windows : alignement des champs, normalisation de la taille des boutons, ...

4.5 Les options d'alignement

Pour aligner plusieurs champs, WINDEV met à votre disposition :

- soit un alignement prédéfini (disponible sous le volet "Alignement" ou sous le volet "Modification", dans le groupe "Alignement").
- soit un alignement personnalisé (disponible sous le volet "Alignement", dans le groupe "Autres alignements").

L'alignement personnalisé permet d'utiliser des propriétés spécifiques d'alignement. Lors de l'alignement personnalisé, il est possible de définir :

- ***l'alignement horizontal ou vertical*** : les champs peuvent être alignés à gauche en fonction du début du champ ou du début de la zone de saisie. Ils peuvent aussi être alignés à droite en fonction de la fin du champ ou de la fin de la zone de saisie.
- ***l'espacement uniforme*** : l'espacement entre les champs est adapté pour être le même entre chaque champ (horizontalement ou verticalement).
- ***la taille*** : la taille des champs sélectionnés est automatiquement adaptée pour obtenir des champs de même hauteur et/ou de même largeur.
- ***le champ de référence*** : pour l'alignement et la taille des champs, le champ de référence est au choix :
  - le premier champ sélectionné (celui qui a les poignées blanches).
  - le dernier champ sélectionné.
  - le plus grand champ sélectionné.
  - le champ situé le plus en haut à gauche des champs sélectionnés.

5 Options d'édition des champs dans une fenêtre

Les options d'édition permettent de paramétrer :

- les options d'affichage.
- le magnétisme des champs.
- les opérations de clic et de double-clic sur les champs.
- les options de sélection des champs.
- la position de la fenêtre à son ouverture.

Ces options d'affichage permettent de personnaliser la manipulation des champs sous l'éditeur de fenêtres.

Pour afficher ces options, sous le volet "Affichage", cliquez sur l'icône de regroupement du groupe "Options".

5.1 Rendre visible / invisible un champ

Pour rendre visible / invisible un champ sous l'éditeur de fenêtres :
1. Sélectionnez le champ sous l'éditeur.
2. Sous le volet "Modification", dans le groupe "Aide à l'édition", cochez/décochez l'option "Visible (éditeur)".

5.2 Verrouiller / Déverrouiller un champ

Pour verrouiller / déverrouiller un champ sous l'éditeur de fenêtres :
1. Sélectionnez le champ sous l'éditeur.
2. Sous le volet "Modification", cochez/décochez l'option "Verrouillé".

Un champ verrouillé est un champ qui ne peut pas être sélectionné au lasso. Il est intéressant par exemple de verrouiller un onglet. Lorsque plusieurs sélections au lasso des champs affichés dans l'onglet sont effectuées, l'onglet n'est pas sélectionné automatiquement.

Toutes les autres manipulations sont possibles sur un champ verrouillé.

5.3 Encadrer ou non un champ

Pour encadrer ou non un champ sous l'éditeur de fenêtres :
1. Sélectionnez le champ sous l'éditeur.
2. Sous le volet "Modification", cochez/décochez l'option "Encadré".

L'encadrement d'un champ permet de visualiser plus facilement l'ensemble des champs présents dans une fenêtre (notamment pour les champs image).

6 Ancrage des champs

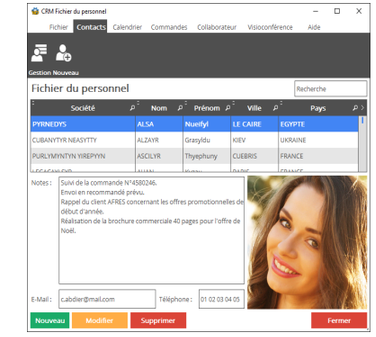
Les fenêtres d'une application WINDEV peuvent être redimensionnables à l'exécution. Grâce au mécanisme d'ancrage, la taille et la position des champs s'adaptent automatiquement lors d'un redimensionnement de la fenêtre.

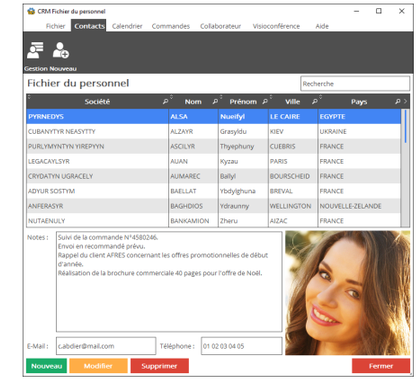
La gestion de l'ancrage d'un champ est constituée de deux paramètres :

- **l'ancrage du champ** : permet de définir la modification à appliquer au champ en fonction du changement de la taille de la fenêtre. Le champ peut se déplacer vers la gauche et ou vers le bas, s'agrandir en largeur et/ou en hauteur, ...
- **le taux d'ancrage** : permet de gérer le pourcentage de déplacement ou d'étirement du champ. Grâce à ce paramètre, il est possible par exemple de :
  - laisser un champ centré en largeur dans la fenêtre quelle que soit sa taille.
  - agrandir proportionnellement certains champs de la fenêtre.

Ces paramètres peuvent être configurés :

- sous l'éditeur de fenêtres.
- par programmation grâce aux propriétés WLangage.





6.1 Ancrer un champ sous l'éditeur

Pour ancrer un champ :
1. Sélectionnez un ou plusieurs champs à ancrer.
2. Définissez l'ancrage du / des champs :

- soit dans l'onglet "IHM" de la fenêtre de description du champ (option "Description" du menu contextuel du champ).
- soit grâce à l'option "Ancrage" du menu contextuel (clic droit) du champ.

3. Dans cette fenêtre, choisissez le type d'ancrage à utiliser. Les options les plus courantes sont présentées sous forme d'icônes :

Aucun ancrage : Le champ n'est pas modifié lorsque la taille de la fenêtre est modifiée. Aucune option d'ancrage n'est sélectionnée.

Largeur : Le champ s'étire vers la droite lorsque la fenêtre s'agrandit. Pour faire varier la vitesse d'étirement du champ par rapport à la fenêtre, utilisez le taux d'ancrage en largeur.

Droite : Le champ se déplace vers la droite lorsque la fenêtre s'agrandit. Pour que le champ se déplace plus ou moins vite, utilisez le taux d'ancrage à droite.

Centré en largeur : Le champ reste centré en largeur dans la fenêtre quelle que soit la largeur de la fenêtre.

Hauteur : Le champ s'étire vers le bas lorsque la hauteur de la fenêtre s'agrandit. Pour faire varier la vitesse d'étirement du champ par rapport à la fenêtre, utilisez le taux d'ancrage en hauteur.

Largeur et Hauteur : Le champ s'étire vers la droite et vers le bas lorsque la fenêtre s'agrandit. Pour faire varier la vitesse d'étirement du champ par rapport à la fenêtre, utilisez le taux d'ancrage en hauteur et le taux d'ancrage en largeur.

Hauteur et Droite : Le champ s'étire vers le bas et se déplace vers la droite lorsque la fenêtre s'agrandit.
Pour que le champ se déplace plus ou moins vite, utilisez le taux d'ancrage à droite.
Pour faire varier la vitesse d'étirement du champ par rapport à la fenêtre, utilisez le taux d'ancrage en hauteur.

Répartition Verticale : Cette option n'est disponible que lorsque plusieurs champs sont sélectionnés. Les options d'ancrage permettent alors de conserver une répartition verticale des champs sélectionnés lorsque la fenêtre s'agrandit en hauteur.

Bas : Le champ se déplace vers le bas lorsque la fenêtre est agrandie vers le bas. Pour que le champ se déplace plus ou moins vite, utilisez le taux d'ancrage en bas.

Largeur et Bas : Le champ s'étire vers la droite et se déplace vers le bas lorsque la fenêtre s'agrandit.
Pour que le champ se déplace plus ou moins vite, utilisez le taux d'ancrage en bas.
Pour faire varier la vitesse d'étirement du champ par rapport à la fenêtre, utilisez le taux d'ancrage en largeur.

Droite et Bas : Le champ se déplace vers la droite et vers le bas lorsque la fenêtre s'agrandit. Pour que le champ se déplace plus ou moins vite, utilisez le taux d'ancrage en bas et le taux d'ancrage à droite.

Centré horizontal en bas : Le champ reste centré en largeur dans la fenêtre quelle que soit la largeur de la fenêtre. Cependant, le champ est ancré en bas et se déplace vers le bas lorsque la fenêtre s'agrandit.

Pour que le champ se déplace plus ou moins vite, utilisez le taux d'ancrage en bas.

Centré en hauteur : Le champ reste centré en hauteur dans la fenêtre quelle que soit la hauteur de la fenêtre.

Répartition Horizontale : Cette option n'est disponible que lorsque plusieurs champs sont sélectionnés. Les options d'ancrage permettent alors de conserver une répartition horizontale des champs sélectionnés lorsque la fenêtre s'élargit.

Centré vertical à droite : Le champ reste centré en hauteur dans la fenêtre quelle que soit la hauteur de la fenêtre.

Cependant, le champ est ancré à droite et se déplace vers la droite lorsque la fenêtre est élargie. Pour que le champ se déplace plus ou moins vite, utilisez le taux d'ancrage à droite.

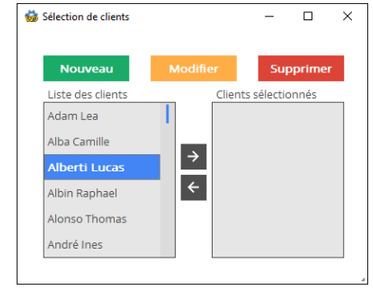
Centré : Le champ reste centré en hauteur et en largeur dans la fenêtre quelle que soit la taille de la fenêtre.

4. Définissez si nécessaire les différents taux d'ancrage.
5. Validez. Si l'option "Ancrage" est activée (sous le volet "Affichage", dans le groupe "Voir (tous les champs)"), les signes d'ancrage apparaissent automatiquement dans le champ (flèches rouges).

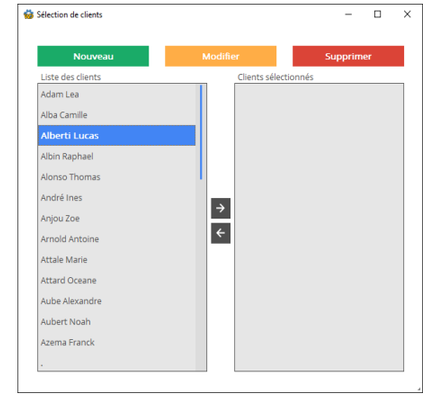
6.2 Exemple de paramétrage de l'ancrage des champs

Pour illustrer les possibilités d'ancrage des champs dans les fenêtres, voici un exemple détaillé.

Fenêtre originale :



Fenêtre agrandie :



Pour obtenir ce résultat :
1. Sélectionnez les boutons [<-] et [->] : ces champs doivent être centrés en hauteur et en largeur (option).

2. Sélectionnez les deux listes : ces champs doivent être ancrés à droite, en largeur et en hauteur (options "à droite", "en largeur" et "en hauteur" cochées).
3. Sélectionnez les 3 boutons : ces champs doivent s'étirer à droite proportionnellement (option).

Remarque : Par programmation, il est possible de configurer l'ancrage des champs grâce à diverses propriétés WLangage.

7 Exécution automatique d'un bouton

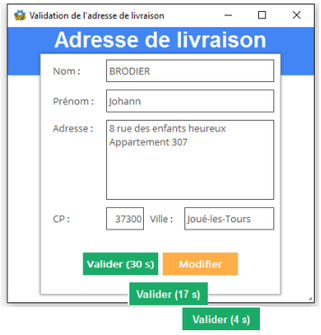
Pour forcer une action sur une fenêtre, il suffit de définir un bouton de cette fenêtre en mode "Exécution automatique". Le principe est très simple, si au bout d'une certaine durée, aucun bouton de la fenêtre n'a été utilisé, le traitement du bouton "Exécution automatique" sera automatiquement exécuté. En exécution, la durée est automatiquement décrémentée. A zéro, le code du bouton est automatiquement exécuté.

7.1 Mise en place

Pour exécuter automatiquement un bouton d'une fenêtre :
1. Déterminez quel bouton de votre fenêtre doit être exécuté automatiquement.
2. Dans la description de la fenêtre (onglet "Détail" de la fenêtre de description), indiquez :

- le temps avant l'exécution automatique du bouton.
- le bouton à exécuter.

3. En exécution, le temps restant est automatiquement affiché.



7.2 Programmation

Il est possible également de définir un bouton de validation automatique grâce à la fonction ***DélaiAvantFermeture***.

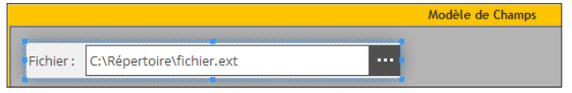
7.3 FAA : Fonctionnalité Automatique de l'Application

Cette fonctionnalité est également directement accessible à l'utilisateur. Il peut demander à ce qu'un bouton de l'application s'exécute automatiquement au bout d'une durée spécifique.

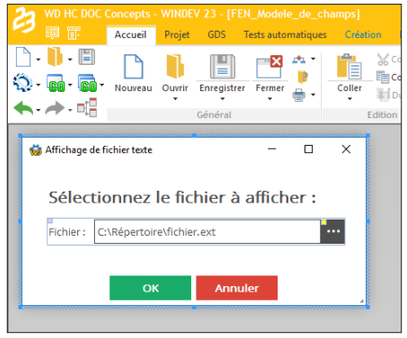
Modèles de champs

WINDEV permet de créer des modèles de champs. Un modèle de champs est un ensemble de champs, réutilisable dans plusieurs fenêtres.
Les modifications effectuées dans un modèle de champs sont automatiquement reportées sur toutes les fenêtres utilisant ce modèle.
Un modèle de champs permet de :

- regrouper un ensemble de champs dans un but précis.
- rendre les champs indépendants de la fenêtre les accueillant.



Définition d'un modèle de champs.
Le modèle est encadré d'un trait vert sous l'éditeur.



Utilisation du modèle dans une fenêtre.
Les éléments appartenant au modèle sont entourés en bleu et identifiés par un carré jaune.

Pour créer une fenêtre utilisant un modèle de champs, créez un champ de type "Modèle de champs".
Remarque : La programmation associée aux éléments du modèle peut être directement réalisée dans le modèle.
Les caractéristiques des éléments peuvent être désolidarisées du modèle. Par exemple, désolidariser la position d'un champ du modèle pour positionner le champ ailleurs tout en conservant les autres évolutions sur le champ (code, style, ...). On parle alors d'**héritage de champ**. Dans ce cas, les éléments sont identifiés par un carré bleu.

Edition : Les états

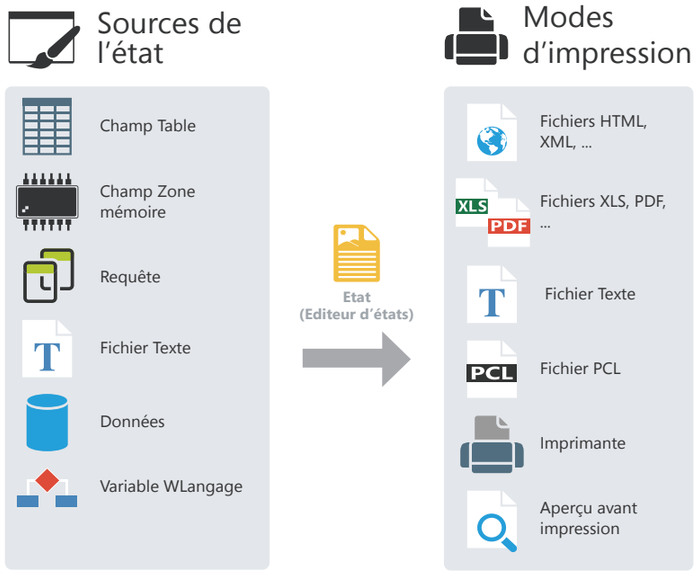
Un état permet d'obtenir une vue personnalisée d'informations : données saisies dans la base de données, données présentes dans un fichier texte, données provenant d'un champ Table, ...
Un état peut être affiché à l'écran, enregistré dans un fichier HTML, directement édité sur une imprimante, ...

Un état offre la possibilité de résumer et de synthétiser les données.
Il est possible de :

- regrouper des données.
- trier des données selon n'importe quel critère.
- réaliser des calculs, des moyennes ou des statistiques ou même réaliser des graphes.

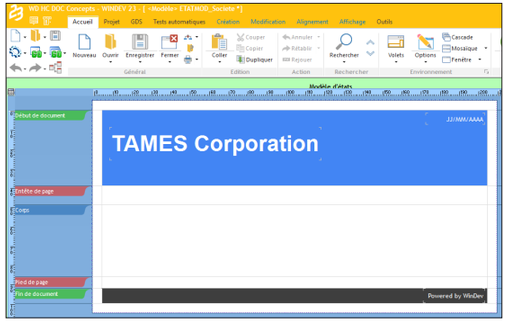
Le schéma ci-dessous présente une définition simplifiée d'un état :

- les données à imprimer sont issues d'une source de données (fichier décrit dans une analyse, vue HFSQL, requête, zone mémoire ou un fichier texte).
- l'état regroupe, trie et met en forme les données.
- l'exécution de l'état peut être dirigée vers l'écran, un fichier HTML, RTF, ... ou vers une imprimante.

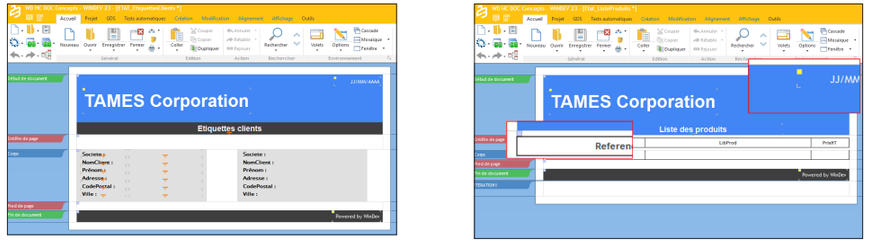


Modèles d'états

Dans une société, bien souvent, les impressions utilisent un look et une mise en page unifiés : date en haut à droite dans un format spécifique, pied de page avec heure d'impression et nom du fichier, logo en haut à gauche, ...
Les modèles d'états permettent d'uniformiser simplement la mise en page de vos états.



Définition d'un modèle d'états sous l'éditeur d'états.



Utilisation du modèle dans différents états.
Les éléments appartenant au modèle sont identifiés par un carré jaune.
Les éléments du modèle surchargés sont identifiés par un carré bleu.

Remarque : La programmation associée aux éléments du modèle peut être directement réalisée dans le modèle.
Les caractéristiques des éléments peuvent être désolidarisées du modèle. Par exemple, désolidariser la position d'un champ du modèle pour positionner le champ ailleurs tout en conservant les autres évolutions sur le champ (code, style, ...). On parle alors d'**héritage**. Dans ce cas, les éléments sont identifiés par un carré bleu.

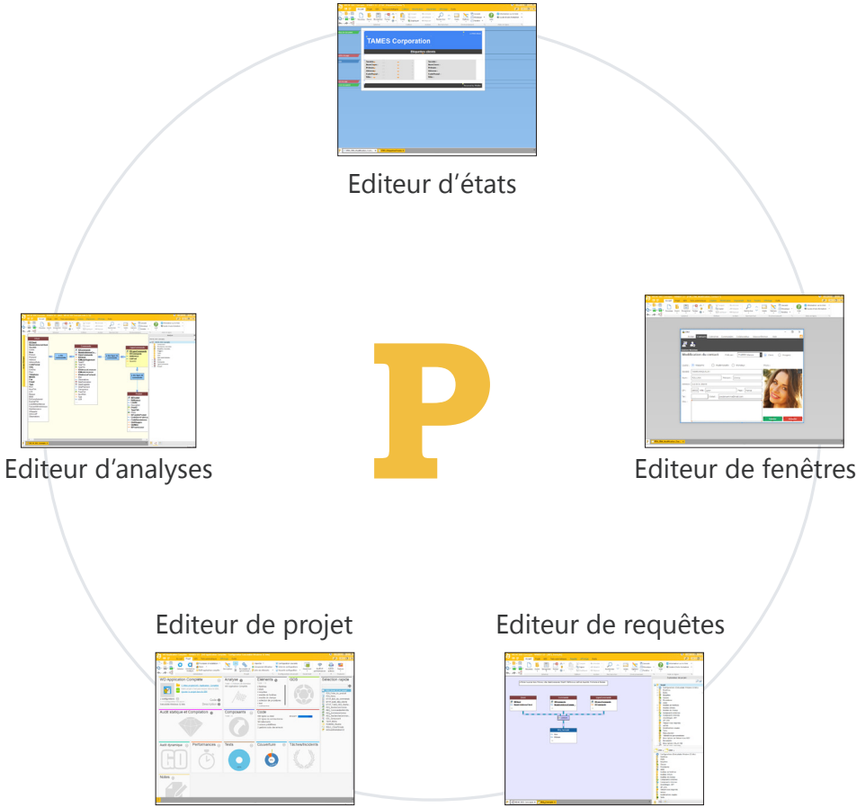
DÉVELOPPEZ 10 FOIS PLUS VITE
PARTIE 2
Environnement de développement

Les éditeurs de WINDEV

Pour manipuler un projet, WINDEV propose de nombreux éditeurs intégrés :

- éditeur de projet.
- éditeur d'analyses.
- éditeur de modèles UML
- éditeur de requêtes.
- éditeur de fenêtres.
- éditeur d'états.
- ...

Ces éditeurs permettent de manipuler facilement les différents éléments du projet.



Des éditeurs conviviaux en pratique

1 Introduction

WINDEV est architecturé autour d'éditeurs, adapté à chacun des besoins du développeur, unifié dans un environnement unique :

- éditeur de projet.
- éditeur d'analyses.
- éditeur de modèles UML.
- éditeur de requêtes.
- éditeur de fenêtres.
- éditeur d'états.
- éditeur de code source, débogueur et compilateur.
- éditeur d'aide.
- éditeur de modélisation.
- éditeur de documents.
- éditeur d'images.

Ces différents éditeurs vont permettre de créer les différents éléments (fenêtres, états, base de données, programmes, ...) manipulés par l'application.

2 Les différents éditeurs

2.1 Editeur de projet

L'éditeur de projet a deux fonctionnalités principales :

- visualisation rapide de l'ensemble des éléments d'un projet (principalement fenêtres, états et requêtes).
- visualisation et création des enchaînements entre les fenêtres et les états.

Voici quelques caractéristiques de l'éditeur de projet :

- Editeur WYSIWYG ("What You See Is What You Get"). Vous visualisez directement tous les éléments de votre projet.
- Gestion du projet à l'aide d'un tableau de bord.
- Création et visualisation des enchaînements entre plusieurs éléments (fenêtres, états, ...) de votre projet à l'aide du graphe du projet.
- Interaction des différents volets avec les différents éléments du projet.

2.2 Editeur d'analyses

L'éditeur d'analyses permet de décrire les caractéristiques et la structure des fichiers de données.

L'éditeur d'analyses supporte la méthode Merise (MCD et MLD).

L'éditeur d'analyses permet principalement deux types de descriptions d'une base de données :

- description directe de l'analyse liée au projet (appelée aussi Modèle Logique des Données (MLD))
- description du Modèle Conceptuel des Données (MCD) puis génération automatiquement de l'analyse.

Voici quelques caractéristiques de l'éditeur d'analyses :

- Editeur WYSIWYG ("What You See Is What You Get"). Vous visualisez directement tous les fichiers de données et les liaisons de votre projet.
- Création et description des fichiers de données et de leurs rubriques.
- Mise en forme automatique des liaisons de l'analyse.
- Informations sur les liaisons par des bulles d'aide.
- Récupération simplifiée de la description d'une base de données externe (SQL Server, Oracle, ...).
- Description indépendante des fichiers de données.
- Cryptage des fichiers de données.
- Modification automatique des fichiers de données en cas de modification de la structure des fichiers.
- Vérification de l'intégrité référentielle.
- Génération automatique de l'analyse.
- Zoom sur l'éditeur d'analyses.
- Insertion de commentaires dans le graphe de l'analyse.
- Interaction des différents volets avec les différents éléments de l'analyse.

2.3 Editeur UML

Le langage UML est un langage graphique permettant de :

- représenter le système d'information étudié sous forme d'objets.
- générer la structure objet de l'application (squelette de l'application sous forme de classes objet) correspondant au système d'information étudié.

L'éditeur d'UML permet principalement :

- de décrire directement un ou plusieurs diagrammes UML.
- de construire un modèle UML par rétro-analyse de votre projet.

Pour plus de détails sur la création d'un modèle UML, consultez le chapitre "Le modèle UML", page 122.

Voici quelques caractéristiques de l'éditeur d'UML :

- Création et description de diagrammes UML.
- Mise en forme automatique des liaisons.
- Génération automatique d'une classe ou d'un ensemble de classes à partir d'un diagramme de classes.
- Zoom sur l'éditeur d'UML.
- Insertion de commentaires dans un modèle UML.
- Interaction des différents volets avec les différents éléments du modèle UML.

2.4 Editeur de requêtes

L'éditeur de requêtes permet de créer automatiquement des requêtes sur les fichiers de données. La programmation est ainsi simplifiée : fenêtres, tables, combos, états, ... pourront s'appuyer sur les requêtes.

Un assistant permet de créer une requête : il suffit de choisir les rubriques à intégrer et de saisir les conditions de sélection à l'aide de l'assistant. La requête est automatiquement générée (en code SQL optimisé) et peut être testée immédiatement.

Vous trouverez toutes les informations concernant les requêtes dans le "Guide de l'éditeur d'états et de requêtes".

Voici quelques caractéristiques de l'éditeur de requêtes :

- Editeur WYSIWYG ("What You See Is What You Get"). Vous visualisez directement la requête et son résultat.
- Création simplifiée d'une requête à l'aide d'un assistant.
- Génération automatique du code SQL de chaque requête.
- Possibilité de tester immédiatement les requêtes.
- Zoom sur l'éditeur de requêtes.

2.5 Editeur de fenêtres

L'éditeur de fenêtres permet de décrire les caractéristiques de l'interface utilisateur de votre projet.

De nombreux gabarits (feuilles de styles associées à la fenêtre) et types de fenêtres sont proposés. Ils permettent d'améliorer l'ergonomie de vos fenêtres et de rendre vos applications conviviales.

Pour plus de détails sur la création d'une fenêtre, consultez le chapitre "Les fenêtres en pratique", page 21.

Voici quelques caractéristiques de l'éditeur de fenêtres :

- Editeur WYSIWYG ("What You See Is What You Get"). Vous visualisez directement votre fenêtre telle qu'elle apparaîtra à l'utilisateur.
- Création et description des fenêtres et de leurs champs.
- "Drag and Drop" permettant de copier, de coller ou de déplacer des champs d'une fenêtre à une autre.
- Gabarits de fenêtres et styles de champs prédéfinis.
- Présence de catalogues d'icônes permettant d'associer des images aux champs.
- Correcteur d'interface en temps réel permettant de simplifier le positionnement des champs.
- Ancrage des champs dans la fenêtre.
- Saisie des libellés des champs directement dans la zone de travail de l'éditeur.
- Gestion de l'aide contextuelle dans les fenêtres.
- Zoom sur l'éditeur de fenêtres.
- Interaction des différents volets avec les différents éléments de l'éditeur.

2.6 Editeur d'états

L'éditeur d'états permet de créer très simplement des états à imprimer.

Vous trouverez toutes les informations concernant la création d'états imprimés dans le "Guide de l'éditeur d'états et de requêtes".

Voici quelques caractéristiques de l'éditeur d'états :

- Editeur WYSIWYG ("What You See Is What You Get"). Vous visualisez directement les états tels qu'ils seront à l'impression.
- Création simplifiée d'un état sans une seule ligne de code.
- "Drag and Drop" permettant de copier, de coller ou de déplacer des champs d'un état à un autre.
- Gabarits d'états et styles de champs prédéfinis.
- Correcteur d'interface en temps réel permettant de simplifier le positionnement des champs.
- Utilisation d'un formulaire en fond d'un état.
- Création d'étiquettes multi-colonnes.
- Edition d'un état au format HTML (pour le publier sur Internet par exemple), ou au format RTF (pour l'éditer avec un traitement de texte par exemple).
- Zoom sur l'éditeur d'états.

2.7 Editeur de code

L'éditeur de code permet de saisir tous les traitements en WLangage (le langage de programmation intégré à WINDEV). Il permet par exemple de saisir le code source :

- des champs,
- des fenêtres,
- des états,
- des procédures locales et globales,
- des classes et des méthodes, ...

Voici quelques caractéristiques de l'éditeur de code :

- Mise en forme automatique des informations saisies.
- Complétion automatique.
- Glossaire sur les fonctions.
- Détection immédiate des erreurs de frappe et aide à la correction.
- Compilation incrémentale.
- Visualisation des différents traitements d'une fenêtre, d'un champ ou d'un état du projet.
- Insertion de traitements spécifiques à l'utilisation de la souris ou du clavier.
- Gestion de l'historique des modifications du code avec possibilité de revenir en arrière.
- Ouverture de plusieurs fenêtres de code (pour effectuer des comparaisons de code par exemple).
- Zoom sur l'éditeur de code.
- Interaction des différents volets avec les différents éléments de l'éditeur.

Pour plus de détails sur l'éditeur de code, nous vous recommandons de lire l'aide en ligne.

2.8 Editeur d'aide

L'éditeur d'aide permet de créer un ensemble de pages d'aide (fichier "CHM") documentant tout ou partie de votre projet.

Pour chaque fenêtre de votre projet, une page d'aide est créée. Ces pages permettent de décrire les différentes manipulations pouvant être réalisées par les utilisateurs.

Voici quelques caractéristiques de l'éditeur d'aide :

- Editeur WYSIWYG ("What You See Is What You Get"). Vous visualisez directement les pages d'aide de votre projet.
- Compilation des pages d'aide et création d'un CHM.
- Création d'un index et d'un sommaire.
- Insertion d'images.
- Insertion de liens entre les pages d'aide.
- Possibilité d'affichage des titres de pages dans un bandeau.
- Zoom sur l'éditeur d'aide.

2.9 Editeur d'images

L'éditeur d'images permet d'éditer les images et icônes du projet et de ses élements.

Voici quelques caractéristiques de l'éditeur d'images :

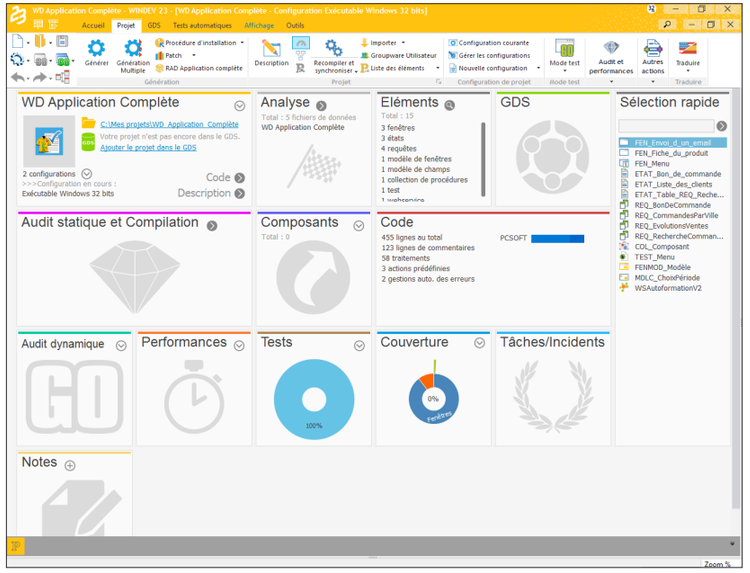
- Recadrage.
- Redimensionnement.
- Symétrie, rotation.
- Gestion des textes.
- Gestion des calques.
- Accès aux images du catalogue d'images.
- Conservation de la transparence.

Tableau de bord du projet

Le tableau de bord du projet est un élément indispensable à la gestion de projets WINDEV. Le tableau de bord permet d'avoir une vision globale et synthétique de l'état d'avancement d'un projet.

Le tableau de bord du projet est composé de différents indicateurs sur le contenu du projet :

- statistiques sur le projet,
- incidents,
- tâches,
- état des tests automatiques,
- résultat des différents audits,
- liste des éléments extraits du GDS (Gestionnaire de Sources),
- résultat des plans d'actions (intégration continue), ...



Les différents éléments de ce tableau de bord sont présentés sous forme de Widgets. Ces Widgets peuvent être paramétrés, déplacés, activés, désactivés, ... Il est possible d'ajouter de nouveaux indicateurs.

La configuration du tableau de bord est sauvegardée par utilisateur. La configuration du tableau de bord est la même pour tous les projets d'un même utilisateur.

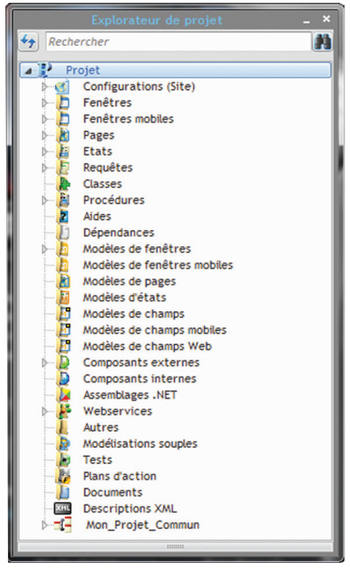
WINDEV/WEBDEV/WINDEV Mobile : 100% compatibles

Les projets créés actuellement sont souvent multi-cibles.
Par exemple, pour un ERP destiné à fonctionner sous Windows, il est fort probable qu'en plus de l'application qui sera le socle de la solution applicative, des commerciaux seront équipés de Mobiles iOS ou Android, les magasiniers utiliseront des terminaux mobiles pour la gestion des stocks et qu'un site Intranet et un site Internet seront mis en place.

Tous les éléments, hormis l'IHM (pages et fenêtres) sont 100% compatibles et partageables entre des projets WINDEV, WEBDEV et WINDEV Mobile.

Il est ainsi possible de partager des collections de procédures ou des classes par exemple entre plusieurs projets.

Quel que soit le produit avec lequel un projet a été créé, il est possible de l'ouvrir avec les autres produits.



Lorsque vous ouvrez un projet dans un produit différent de celui qui a permis de le créer, un assistant s'affiche, permettant de créer une **configuration de projet** spécifique au produit utilisé.

Par exemple, si un projet WINDEV est ouvert avec WEBDEV, vous pouvez créer une configuration de projet nommée "Site", permettant de regrouper tous les éléments nécessaires au site WEBDEV.

Dans les phases d'utilisation de l'environnement, il est possible de visualiser depuis chaque environnement les éléments de chaque cible. Un projet sous WINDEV affiche les vignettes des pages WEBDEV et les fenêtres WINDEV Mobile par exemple. Cliquer sur une page WEBDEV depuis l'éditeur de projet WINDEV ouvre la page WEBDEV (WEBDEV doit être installé sur le poste).

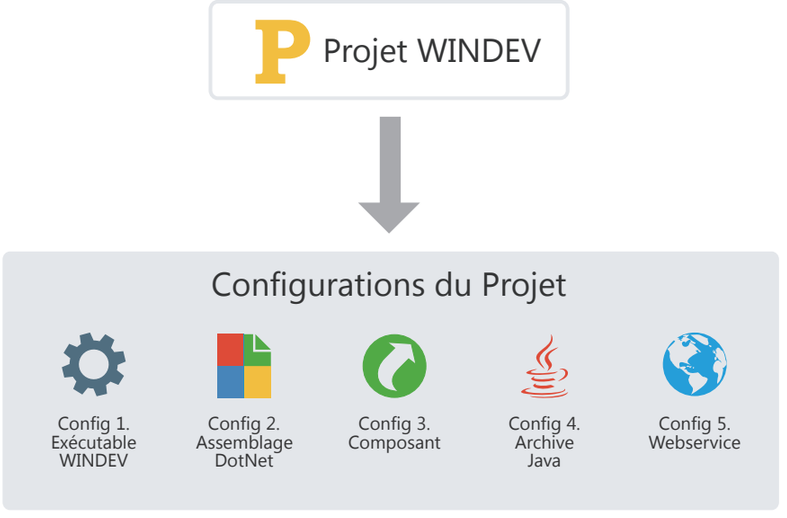
Remarque : Si vous possédez WINDEV et WINDEV Mobile, il est possible de manipuler les configurations WINDEV Mobile directement sous WINDEV. Vous pouvez ouvrir les fenêtres mobiles, les modifier, les tester, ...

Configuration de projet

Les configurations de projet permettent de créer à partir d'un même projet plusieurs "cibles" différentes.
Vous pouvez ainsi créer à partir d'un même projet par exemple :

- des exécutables qui ne contiennent pas les mêmes éléments, qui ont des noms différents, ...
- différents composants
- des Webservices
- des assemblages DotNet

A tout moment, vous pouvez travailler sur une configuration spécifique : les éléments n'appartenant pas à cette configuration apparaissent grisés dans le graphe de projet.



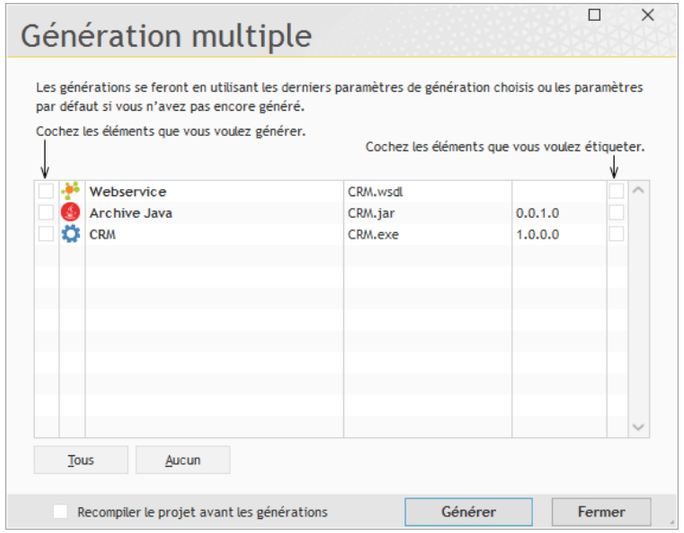
Il est possible de générer en une seule opération toutes les configurations d'un projet (ou uniquement certaines) grâce à la génération multiple.

Génération multiple

Les configurations de projet permettent de définir simplement les différentes "Cibles" de votre projet. Vous pouvez ainsi définir pour un même projet, des exécutables, bibliothèques, Webservices, assemblages DotNet, ...

Pour générer le résultat de chaque configuration, vous pouvez bien entendu sélectionner une à une chaque configuration et générer le programme correspondant.

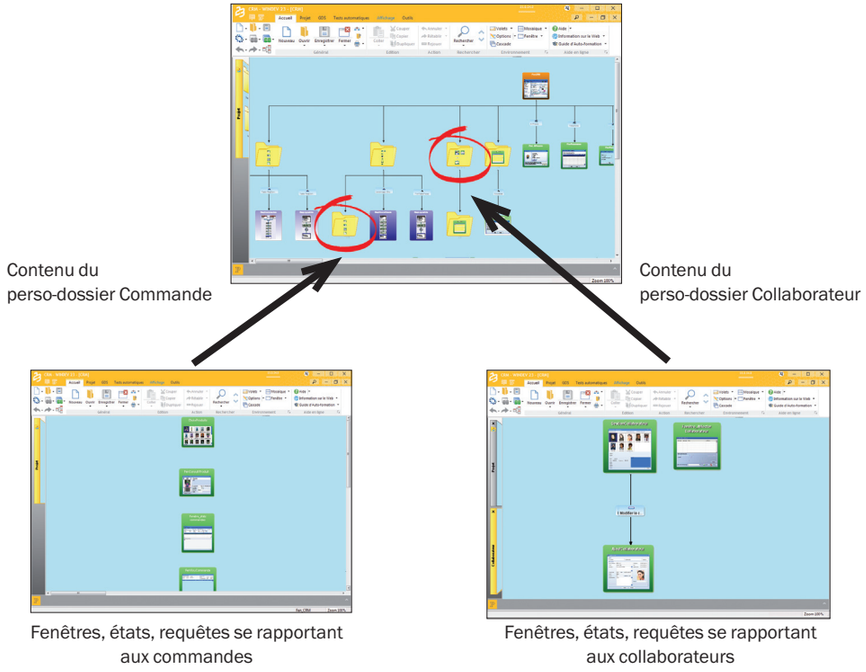
Un autre moyen plus rapide existe : **la génération multiple**. Vous sélectionnez en une seule opération les configurations à générer et le résultat est immédiat.



Pour lancer une génération multiple, sous le volet "Projet", dans le groupe "Génération", cliquez sur "Génération Multiple".

Perso-dossiers : Organisez votre projet

Tous les éléments qui composent un projet sont listés dans le volet "Explorateur de Projet". Par défaut, les éléments sont triés en fonction de leur type : fenêtres, états, classes... Dans les projets de taille importante, il est souvent plus pertinent de regrouper les éléments qui ont trait à la même fonctionnalité : gestion du stock ou traitement des commandes par exemple.
Pour cela, il suffit de créer des "perso-dossiers" dans l'arborescence de l'explorateur de projet et de glisser les différents éléments dans ces dossiers.
Des éléments peuvent être communs à plusieurs "perso-dossiers".
Il est ainsi plus simple de travailler sur une partie de l'application.



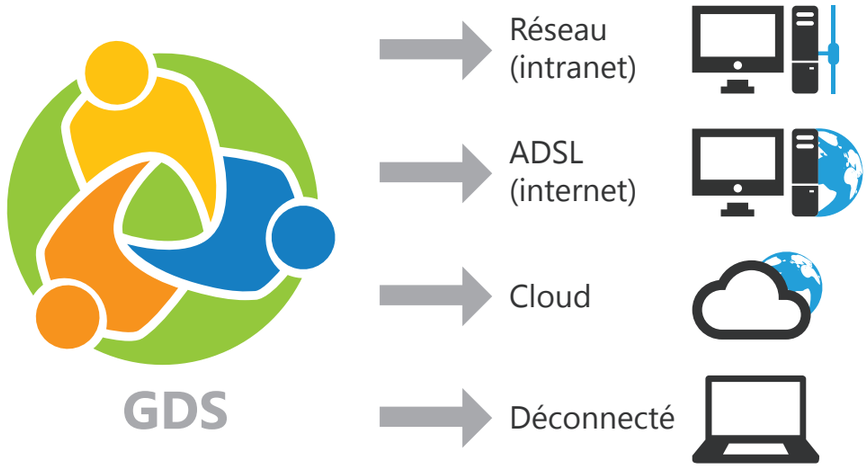
Remarques :

- Pour ranger, organiser vos projets, mais aussi pour partager un ensemble d'éléments entre différents projets, WINDEV met également à votre disposition les composants internes.
- Les "perso-dossiers" sont affichés dans le graphe du projet et dans le volet "Explorateur du projet".

Gestionnaire de sources (GDS)

Présentation

Pour simplifier vos développements en équipe, WINDEV propose un gestionnaire de sources. Ce gestionnaire de sources permet à plusieurs développeurs de travailler simultanément sur le même projet et de partager aisément des éléments entre différents projets.

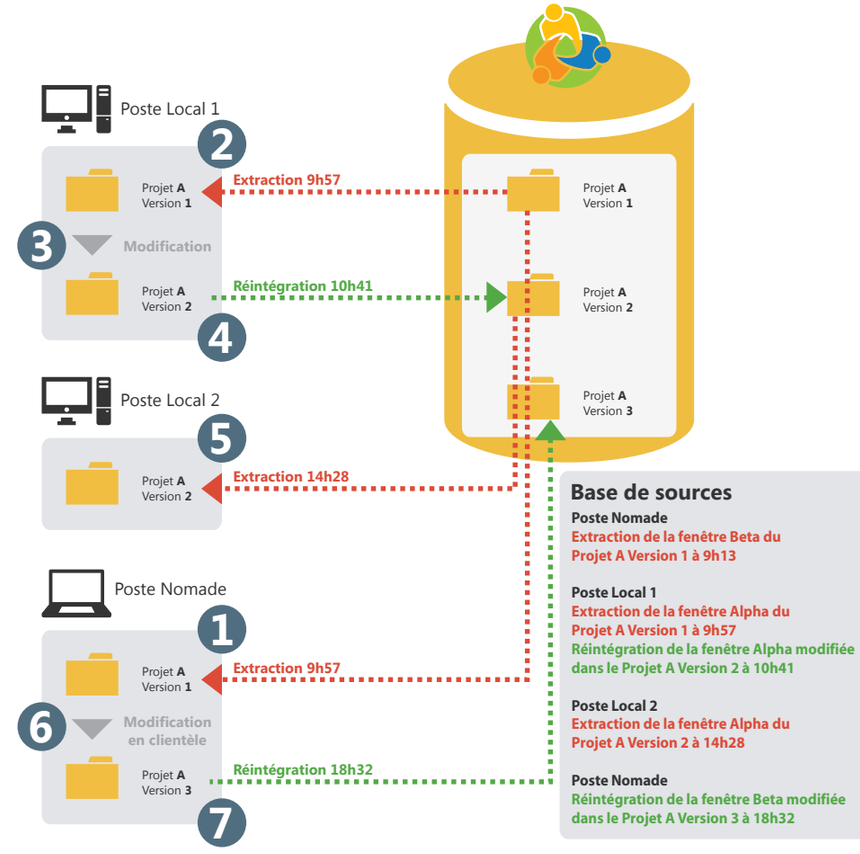


Il est possible de partager les éléments présents dans le GDS :

- via un réseau local,
- via Internet,
- via le Cloud,
- en mode déconnecté. Dans ce cas, les éléments sur lesquels un travail spécifique devra être effectué seront extraits du GDS par exemple lors de la connexion du portable au système général.

Fonctionnement du Gestionnaire de Sources

Voici un exemple d'utilisation du Gestionnaire de Sources :



Si un élément (fenêtre, état, ...) du projet est extrait, ce même élément ne pourra pas être extrait une seconde fois.
Après modification des éléments extraits, il est nécessaire de réintégrer ces éléments pour que le projet de base prenne en compte ces modifications. En effet, la base de sources conserve un historique de tous les éléments du projet depuis leur création.

A chaque réintégration d'un élément, le numéro de version du projet de base est incrémenté de 1.

Partager un projet en pratique

Un développement informatique important nécessite la participation de plusieurs développeurs. Ces développeurs doivent travailler sur un projet WINDEV unique en partageant les différentes ressources (requêtes, classes, ...) manipulées.

Pour partager un projet, vous pouvez utiliser :

- le gestionnaire de sources.
- le groupware développeur. Son utilisation est conservée pour compatibilité avec les versions précédentes. Pour plus de détails, consultez l'aide en ligne (mot-clé : "Groupware développeur").

1 Le Gestionnaire de Sources

1.1 Présentation

WINDEV innove dans la gestion du travail en groupe avec le gestionnaire de sources (appelé également GDS).

Totalement intégré à l'environnement, le gestionnaire de sources (GDS) permet :

- un développement performant en équipe,
- l'historique des modifications et des versions,
- l'automatisation de la sauvegarde des sources des membres de l'équipe développement.

Dimensionné pour des équipes de 1 à 100 développeurs, le GDS favorise et normalise sans contrainte les échanges entre développeurs (même en développant seul, le GDS est utile car il contient l'historique de vos applications).

Le GDS utilise une base de sources : procédures, classes, fenêtres, états, composants, analyses, ...

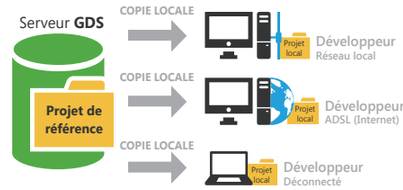
Cette base peut être installée à votre choix sur un serveur (en mode HFSQL Classic ou HFSQL Client/Serveur) ou sur un poste du réseau dans un répertoire partagé.

Le GDS permet un fonctionnement connecté en local et à distance, via Internet ou une liaison HTTP ou HTTPS. Il est ainsi possible de travailler sur un projet depuis une agence ou depuis un site client sans crainte de perte des modifications effectuées.

Le GDS permet également un fonctionnement déconnecté (train, avion, ...).

1.2 Principe d'utilisation

Installation



Tous les éléments du projet sont enregistrés dans la base de sources (sur le serveur). Cette opération est effectuée à la création du projet ou lors de l'importation d'un projet existant dans le Gestionnaire de Sources.

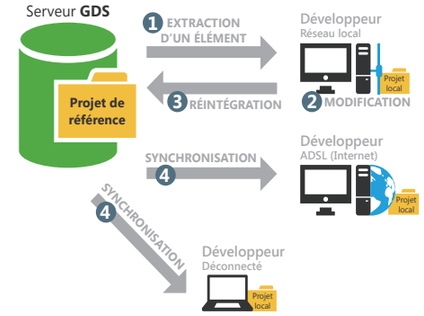
Chaque développeur utilisant le Gestionnaire de Sources récupère une copie du projet en local.

Utilisation

Pour travailler sur un élément du projet (fenêtre, état, ...), le développeur doit extraire l'élément de la base de sources, le modifier, puis le réintégrer.

Si l'élément est déjà extrait, le développeur peut demander à la personne ayant extrait de réintégrer l'élément (par l'intermédiaire de la messagerie intégrée).

Pour profiter des modifications effectuées, les autres développeurs doivent synchroniser leur projet local avec le projet de référence (présent dans la base de sources).



Conseils

Les sources de vos applications sont primordiales.

Ces sources doivent être traitées avec l'attention qu'elles méritent !

Voici donc une série de conseils qui s'appliquent au serveur qui hébergera vos sources :

- Utilisez un serveur dédié avec un disque de taille confortable (au moins 200 Go).
- Utilisez plutôt le Gestionnaire de Sources (GDS) en mode Client/Serveur.
- Les disques durs peuvent avoir des problèmes physiques : utilisez si possible un système RAID I sur votre serveur (plusieurs disques stockant les mêmes informations en double).
- Protégez l'alimentation de votre serveur par un onduleur.
- Faites des sauvegardes régulières de la base de sources (1 sauvegarde par semaine au minimum).
- Placez le serveur dans une zone "sécurisée", en utilisant un firewall.

2 Créer une base de sources pour le GDS

2.1 Présentation

Pour partager un projet grâce au gestionnaire de sources, il est nécessaire de créer une base de sources. Cette base de sources doit être créée une seule fois sur un poste serveur.

Le système d'exploitation et le système de fichiers sur lequel est installée la base du GDS doivent savoir gérer les fichiers de plus de 4 Go.

Il est possible de créer la base de sources :

- dans un répertoire partagé sur le réseau : tous les utilisateurs du GDS doivent avoir tous les droits sur ce répertoire. La base de sources sera au format HFSQL Classic. Il suffit de préciser le répertoire réseau où la base de sources doit être créée.
- sur un serveur HFSQL Client/Serveur : la base de sources sera au format HFSQL Client/Serveur. Dans ce cas, il est nécessaire de préciser :
  - le serveur
  - la base
  - le port utilisé
  - le nom et le mot de passe d'un administrateur de la base HFSQL.

2.2 Quand créer une base de sources ?

La base de sources doit être créée une seule fois.

WINDEV permet de créer cette base de sources à différents moments :

- lors de l'installation de WINDEV.
- lors de la création d'un projet utilisant le GDS.
- lors de l'importation d'un projet dans le gestionnaire de sources.
- directement dans l'administrateur du GDS.

Lorsque la base de sources est créée, il est possible d'importer dans cette base de sources tous les projets WINDEV à partager.

2.3 Sauvegardes

Il est conseillé de faire régulièrement des sauvegardes de la base de sources du GDS. Ces sauvegardes peuvent être effectuées grâce à l'administrateur du GDS.

3 Configurer le projet pour travailler avec le GDS

Pour utiliser un projet existant avec le gestionnaire de sources, quelques manipulations sont nécessaires.

3.1 Ajouter un projet dans le GDS

Pour ajouter un projet local dans le GDS :
1. Ouvrez votre projet sous WINDEV.
2. Sous le volet "GDS", dans le groupe "Projet", cliquez sur "Ajouter le projet".
3. Dans l'assistant qui s'affiche, sélectionnez la base de sources à utiliser.
4. Dans l'assistant qui s'affiche, sélectionnez la base de sources à utiliser. Indiquez si vous voulez utiliser :

- une base de sources située sur un partage réseau (base de sources au format HFSQL Classic). Donnez le répertoire de la base de sources (poste du réseau ou répertoire partagé).
- une base de sources située sur un serveur HFSQL Client/Serveur. La base de sources sera au format HFSQL Client/Serveur. Dans ce cas, il est nécessaire de préciser :
  - le serveur
  - la base
  - le port utilisé
  - le nom et le mot de passe d'un administrateur de la base HFSQL.
- une base de sources GDS Drive. Indiquez les renseignements d'identification à votre GDS Drive : email, mot de passe et équipe.
- une base de sources située dans PCS Cloud. Indiquez le nom de la plateforme cloud utilisée et le projet à ouvrir.

Remarque : Si vous n'avez pas encore créé de base de sources, cliquez sur le bouton "Créer une base".
5. Validez l'assistant. Le projet est ajouté au GDS.

3.2 Partage de ressources

Vous avez importé dans le gestionnaire de sources un premier projet. Ce projet contient des éléments partagés avec d'autres projets (classes, fenêtres, procédures, feuilles de styles, analyse, ...).

Le partage est une notion importante du GDS. Différentes méthodes sont disponibles pour effectuer le partage. Pour plus de détails, consultez l'aide en ligne.

4 Travailler avec le GDS

4.1 Options d'un projet concernant le GDS

Plusieurs options permettent de configurer un projet manipulé avec le Gestionnaire de Sources. Ces options sont regroupées dans l'onglet "GDS" de la description du projet (option "Description" sous le volet "Projet"). Ces options sont cochées par défaut.

- **Proposer la récupération de la dernière version des éléments à l'ouverture du projet :**
  Cette option permet lors de l'ouverture d'un projet présent dans le GDS, de proposer la récupération de la dernière version des différents éléments du projet.
  Par défaut, la dernière version des éléments est automatiquement récupérée.

- **Proposer la réintégration des éléments à la fermeture du projet** :
  Cette option permet d'afficher lors de la fermeture du projet la liste de tous les éléments actuellement extraits, afin de réintégrer certains ou tous ces éléments.
  Par défaut, à la fermeture du projet, les éléments extraits ne sont pas réintégrés.

- **Extraire/Réintégrer automatiquement le projet :**
  Cette option permet une gestion automatique du "fichier projet". Si cette option est cochée, le fichier projet est extrait uniquement si l'action réalisée le nécessite. Lorsque l'action sur le projet a été effectuée, le fichier projet est automatiquement réintégré.
  Cette option permet de ne plus avoir la gestion "Maître / Invité" sur le projet. Il est également possible d'activer cette option sous le volet "GDS", dans le groupe "Projet", en déroulant "Maître / Invité" et en sélectionnant "Gérer automatiquement l'extraction du projet".

4.2 Extraire un élément

Les différents modes d'extraction

Le GDS propose deux modes d'extraction des éléments du projet :

- **le mode classique** : si vous affichez un élément du GDS non extrait, un panneau indique que cet élément doit être extrait pour être modifié. Vous pouvez extraire immédiatement l'élément (bouton d'extraction présent dans le panneau).
- **le mode automatique** : si vous tentez de modifier un élément du GDS non extrait, le GDS propose automatiquement d'extraire cet élément. Après la validation de l'extraction, l'élément peut être modifié.
  Remarque : ce mode est déconseillé lors d'une utilisation du GDS avec une connexion Internet lente.

Pour changer le mode d'extraction :
1. Sous le volet "Accueil", dans le groupe "Environnement", déroulez "Options" et sélectionnez l'option "Options générales de WINDEV".
2. Dans l'onglet "Général", l'option "Extraction des éléments à la première modification" permet de passer tous les prochains projets ouverts en mode automatique.

Ouvrir un élément du projet pour modifier ses caractéristiques

Pour modifier les caractéristiques d'un élément d'un projet géré par le GDS :
1. Extrayez l'élément du Gestionnaire de sources.
2. Sélectionnez le mode d'extraction de l'élément. Le mode d'extraction peut être :

- ***exclusif*** : personne ne pourra extraire cet élément jusqu'à sa réintégration. L'élément pourra uniquement être extrait pour test.
- ***pour test*** : l'élément pourra être modifié mais les modifications ne pourront pas être réintégrées.
- ***multiple*** : l'élément pourra également être extrait par d'autres utilisateurs. Dans ce cas, lors de la réintégration, il sera possible de visualiser les différences entre les différentes versions de l'élément.

3. Validez. L'élément est ouvert. La barre de titre indique que l'élément est extrait.

4.3 Réintégrer un élément

Les éléments extraits du Gestionnaire de sources apparaissent entourés d'un trait rouge sous l'éditeur de projet.

Pour réintégrer un élément, il suffit de sélectionner l'option "Réintégrer" dans le menu contextuel de l'élément (dans le graphe du projet ou dans le volet "Explorateur de projet").

Lors de la réintégration d'un élément, un écran s'affiche permettant d'effectuer les actions suivantes avant de réintégrer l'élément :

- connaître les modifications effectuées.
- comparer l'élément de la base de sources avec l'élément local (extrait).
- accéder à l'historique de l'élément dans la base de sources.

Il est possible de réintégrer les modifications effectuées dans l'élément, tout en conservant l'élément extrait (option "Garder le fichier extrait").

4.4 Modes de gestion du projet

Deux modes de gestion du projet sont disponibles avec le GDS :

- Gestion du projet en mode Maître / Invité.
- Gestion du projet en mode automatique (par défaut).

Maître et invité

Le Gestionnaire de sources considère 2 types d'utilisateurs :

- le maître : le maître est l'utilisateur qui a placé le projet dans le gestionnaire de sources.
- les invités : les invités sont les développeurs qui manipulent le projet depuis le gestionnaire de sources.

Il n'est pas obligatoire d'être connecté en mode maître sur un projet. Le mode "Maître" est nécessaire uniquement pour :

- modifier les caractéristiques du projet et réintégrer ces modifications dans la base de sources.
- réintégrer tous les éléments pour faire l'exécutable et le programme d'installation de l'application.

Pour passer du mode maître au mode invité, sous le volet "GDS", dans le groupe "Projet", déroulez "Maître / Invité" et sélectionnez l'option "Devenir invité sur le projet (et tout réintégrer)".

Dans ce cas, le gestionnaire de sources propose la réintégration de tous les éléments du projet (y compris le fichier .WDP).

Pour passer du mode invité au mode maître, sous le volet "GDS", dans le groupe "Projet", déroulez "Maître / Invité" et sélectionnez l'option "Devenir maître sur le projet".

Attention : Modification des options du projet :

Tous les utilisateurs du projet (qu'ils soient maîtres ou invités) peuvent modifier les caractéristiques du projet (première fenêtre du projet, animation, charte de programmation, ...). Ces modifications pourront être réintégrées dans le gestionnaire de sources uniquement par le maître du projet.

Les modifications faites par un invité seront perdues lors de la mise à jour du projet depuis la base de données.

Mode automatique

Avec le mode automatique, le fichier projet est extrait uniquement si l'action réalisée le nécessite (quel que soit l'utilisateur). Lorsque l'action sur le projet a été effectuée, le fichier projet est automatiquement réintégré.

Le mode automatique permet de ne plus avoir la gestion "Maître/Invité" sur le projet.

5 Travailler en mode déconnecté avec le GDS

Le Gestionnaire de Sources permet très simplement de travailler en mode déconnecté ou nomade.

Ce mode permet par exemple à un développeur utilisant un portable de continuer à travailler sur un projet présent dans la base de sources tout en étant déconnecté de la base de source.

Le principe est simple :

- avant la déconnexion, sous le volet "GDS", dans le groupe "Autres actions", déroulez "Travail à distance" et sélectionnez l'option "Se déconnecter pour une utilisation nomade". Il est conseillé d'extraire avant la déconnexion les différents éléments qui vont être modifiés (ainsi, ces éléments seront "déjà extraits" pour les autres utilisateurs). Vous pourrez travailler en local sur votre projet. Les différents éléments du projet sont manipulables directement.
- à la reconnexion, sous le volet "GDS", dans le groupe "Autres actions", déroulez "Travail à distance" et sélectionnez l'option "Se reconnecter et synchroniser". Il suffit alors de réintégrer les éléments modifiés.

Pour plus de détails, consultez l'aide en ligne.

6 Administrateur du GDS

L'administrateur du GDS permet de manipuler directement les différents projets inclus dans le gestionnaire de sources.

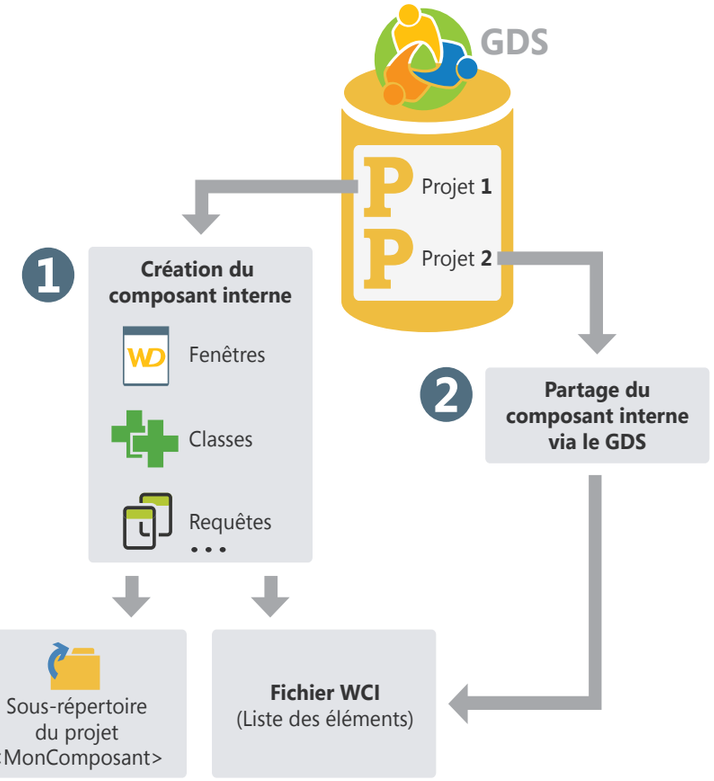
Il permet par exemple de :

- gérer les bases de sources (création, connexion à une base de sources).
- gérer les branches.
- gérer les fichiers et les répertoires présents dans un projet de la base de sources (ajouter, supprimer, renommer, ... des fichiers et des répertoires).
- gérer les différents fichiers de la base de sources (extraction, réintégration, partage, ...).
- lancer certains outils (options, maintenances, ...).
- restaurer une version d'un projet.
- voir les historiques et les différences de versions de sources.
- annuler les extractions (en mode administrateur).
- purger une base, la sauvegarder, la restaurer, ...
- rajouter des fichiers de tout type dans la base (.doc, .xls, .pdf, ...).

Composant interne

Un composant interne est un regroupement d'éléments d'un projet. Ce regroupement permet de :

- Organiser un projet : vous pouvez créer des composants internes pour regrouper les éléments d'un projet, par exemple par fonctionnalité.
- Partager des éléments entre différents projets, par l'intermédiaire du GDS (Gestionnaire de Sources).



Les éléments d'un composant interne peuvent être privés ou publics :

- Les éléments privés ne pourront être manipulés que par les autres éléments du composant.
- Les éléments publics pourront être manipulés par les éléments du projet utilisant le composant interne.

Composant interne en pratique

1 Présentation

Un composant interne est un regroupement d'éléments d'un projet. Ce regroupement permet de :

- **Organiser un projet** : vous pouvez créer des composants internes pour regrouper les éléments d'un projet par exemple par fonctionnalité.
- **Partager des éléments entre différents projets** : Les éléments présents dans un composant interne peuvent être partagés entre plusieurs projets par l'intermédiaire du GDS.

Un des avantages du composant interne par rapport au composant externe est de pouvoir déboguer directement le composant interne depuis le projet qui l'utilise.

Différence avec un composant externe : Un composant interne permet d'intégrer la totalité des éléments du composant dans l'interface du projet contenant le composant interne. Il est possible de manipuler tous les éléments "publiques" du composant interne directement sous l'éditeur.

Lors de l'utilisation d'un composant externe, les éléments "publics" du composant ne peuvent pas être manipulés directement. Il est nécessaire pour modifier le composant externe d'ouvrir le projet correspondant.

2 Créer un composant interne

2.1 Les différentes étapes

Pour créer un composant interne :

1. Sous le volet "Projet", dans le groupe "Configuration de projet", déroulez "Nouvelle configuration" et sélectionnez l'option "Composant Interne". L'assistant création d'un composant interne se lance.

2. Indiquez les caractéristiques du composant interne :

- son nom. Le nom du composant interne sera utilisé pour le fichier WCI correspondant à la description du composant interne. Ce nom sera également utilisé pour créer un sous-répertoire dans votre projet contenant tous les éléments du composant interne.
- son libellé.
- sa description.

3. Indiquez les éléments à intégrer dans le composant interne. Un composant interne peut contenir tous les types d'éléments présents dans un projet : fenêtres, états, modèles, collections de procédures, ...

4. Indiquez les éléments du composant interne qui seront directement accessibles dans le code et dans les actions prédéfinies du projet accueillant le composant interne.

Remarque : Les éléments du composant interne accessibles (ou "public") seront automatiquement proposés par la complétion. Ils pourront être utilisés par les éléments présents dans le projet ou depuis un autre composant interne. Les éléments non accessibles (ou "privés") pourront être manipulés uniquement par un autre élément du composant interne (la complétion proposera ces éléments "privés" uniquement depuis les éléments d'un composant interne).

5. Indiquez le mode de gestion des données et des contextes d'exécution. Trois modes de gestion des données et des contextes d'exécution sont disponibles :

- **Utiliser l'analyse du projet ou aucune analyse (partage complet)** : Le composant interne accède aux fichiers de données du projet. Le composant interne utilise également les mêmes contextes d'exécution que le projet.
  Ce mode correspond au mode par défaut si le composant interne n'utilise pas d'analyse.
- **Utiliser l'analyse du projet avec contexte d'exécution différent (mode avancé)** : Le composant interne accède aux fichiers de données du projet. Le composant interne utilise des contextes d'exécution distincts de ceux du projet.
  Ce mode est à utiliser dans des cas très spécifiques.
- **Utiliser une analyse spécifique** : Le composant interne accède à ses propres fichiers de données. Le composant interne utilise des contextes d'exécution distincts de ceux du projet.
  Ce mode correspond au mode par défaut si le composant utilise une analyse.
  Dans ce cas, il est nécessaire d'indiquer l'analyse utilisée par le composant interne. Il est également possible de créer directement une nouvelle analyse. Cette analyse sera associée au composant interne.

6. Validez la création du composant interne. A tout moment, il sera possible de :

- Modifier les caractéristiques du composant interne grâce à la fenêtre de description du composant interne.
- Manipuler le composant interne et ses éléments.

Astuce : Un composant interne n'a pas de code de déclaration des globales. Il suffit d'utiliser une collection de procédures pour effectuer l'initialisation du composant interne.

2.2 Composant interne et analyse : cas de l'autonomie totale

Un composant interne peut être lié à sa propre analyse. Dans ce cas, le projet accueillant le composant interne peut avoir plusieurs analyses :

- l'analyse du projet.
- l'analyse du ou des composants internes. Cette analyse est définie lors de la création du composant interne. Elle peut être également sélectionnée depuis la fenêtre de description du composant interne.

Les éléments générés par le RAD pour le composant interne seront générés dans le répertoire du composant interne. Bien entendu, si l'analyse du composant interne est modifiée, une nouvelle génération par le RAD proposera automatiquement de générer uniquement les éléments correspondants aux modifications.

3 Partager des composants internes (via le GDS)

Les composants internes peuvent être partagés entre projets grâce au GDS.

Pour partager un composant interne via le GDS, il est nécessaire que le projet contenant le composant interne soit dans le GDS. Il est possible :

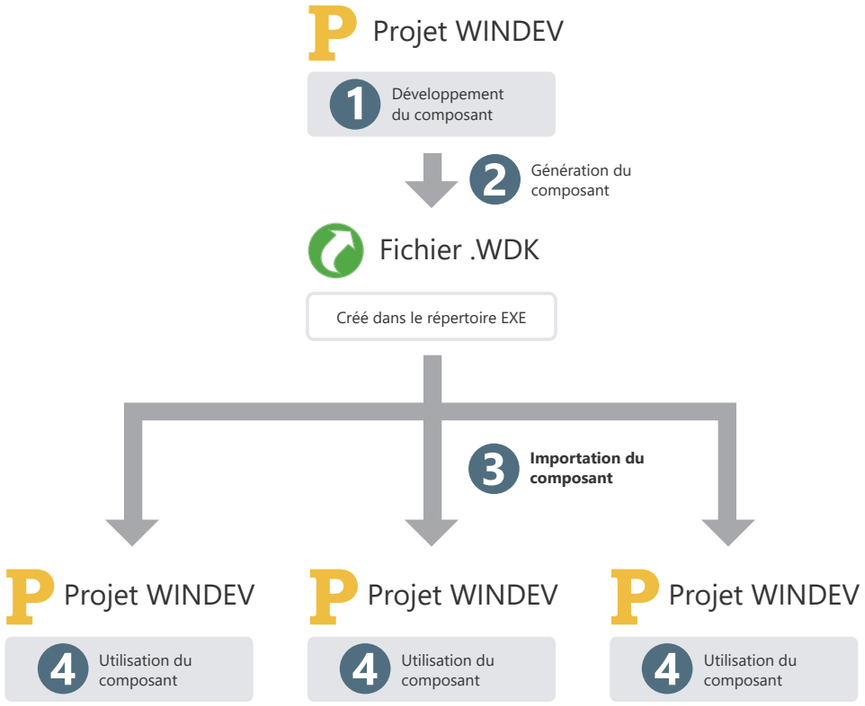
- soit de créer le composant interne à partir d'un projet présent dans le GDS. Le composant interne sera automatiquement présent dans le GDS.
- soit d'importer dans le GDS un projet contenant un ou plusieurs composants internes. Les composants internes seront alors automatiquement intégrés dans le GDS.

Composant externe

Un composant externe est un ensemble d'éléments WINDEV : fenêtres, états, analyse, ... Cet ensemble d'éléments effectue une fonctionnalité précise. Par exemple, un composant peut correspondre à une des fonctionnalités suivantes :

- Envoi de fax,
- Envoi d'emails,
- ...

Un composant externe peut être redistribué à d'autres développeurs WINDEV (gratuitement ou non). Ces développeurs pourront ainsi intégrer simplement la fonctionnalité proposée par le composant externe dans leur application sans pour autant avoir accès au code source correspondant (s'il n'a pas été lui aussi distribué). Le composant externe sera ensuite intégré à l'application et distribué avec elle.



Composant externe en pratique

1 Présentation

1.1 Définition

Un composant externe est un ensemble d'éléments WINDEV : fenêtres, états, analyse, ... Cet ensemble d'éléments effectue une fonctionnalité précise. Par exemple, un composant externe peut correspondre à une des fonctionnalités suivantes :

- Envoi de fax
- Envoi d'emails, ...

Remarques :

- Pour optimiser l'organisation de vos projets, il est possible d'utiliser des composants internes. Pour plus de détails, consultez "Composant interne en pratique", page 72.
- Dans la suite de ce chapitre, le terme "composant" signifie "composant externe".

Un composant externe peut être redistribué à d'autres développeurs WINDEV (gratuitement ou non). Ces développeurs pourront ainsi intégrer simplement la fonctionnalité proposée par le composant dans leur application. Le composant sera donc intégré à l'application et distribué avec l'application.

Lors de la création du composant, l'auteur spécifie :

- la provenance des éléments du composant. Le développeur peut construire un composant à partir d'un projet WINDEV lié à un projet WEBDEV et/ou un projet WINDEV Mobile.
- les éléments du composant visibles ou non par l'utilisateur du composant. Les éléments visibles seront accessibles dans le graphe du projet ou par programmation.
- le mode d'utilisation du composant (comment utiliser le composant, les paramètres permettant d'accéder aux fonctionnalités du composant, ...). Une documentation succincte (à compléter) est automatiquement générée et associée au composant.

Remarque : Des exemples de composants sont livrés en standard avec WINDEV. Ces composants sont accessibles depuis la fenêtre de bienvenue (option "Ouvrir un exemple").

1.2 De quoi est constitué un composant externe ?

Un composant est défini et généré à partir d'un projet WINDEV existant. Il peut être ensuite intégré dans d'autres projets WINDEV.

Tous les éléments du composant sont totalement indépendants du projet dans lequel le composant est intégré. Le composant peut avoir sa propre analyse, ses fenêtres, ses états, ses fichiers de données, ... Lors de la création de la génération du composant, il suffit d'indiquer si ces éléments pourront être manipulés ou non dans le projet WINDEV intégrant ce composant.

Le composant est constitué de trois fichiers :

<NomComposant>.WDK	***Fichier composant***. Contient tous les éléments du composant. Ce fichier est nécessaire à l'intégration du composant dans un projet. Ce fichier doit également être livré dans la procédure d'installation de l'application utilisant le composant.
<NomComposant>.WDI	***Description du composant*** (uniquement pour le développement d'applications). Ce fichier est nécessaire à l'intégration du composant dans un projet. Ce fichier **ne doit pas** être livré dans la procédure d'installation de l'application utilisant le composant.
<Nom du composant>.WDO (fichier optionnel)	***Fichier optionnel*** Fichier au format texte contenant la liste des différents éléments livrés en plus avec le composant (fichiers de données, fichiers .INI, ...). Consultez "Le fichier WDO", page 80.

1.3 Que contient un composant externe ?

Un composant contient :

- les différents éléments à redistribuer (fenêtres, états, ...). Ces éléments peuvent être accessibles ou non lors de l'importation du composant dans un projet WINDEV.
- une brève description du composant.

- une aide permettant de ré-utiliser ce composant. Cette aide est générée à partir des commentaires de code.

Remarque : Par défaut, si un composant utilise une analyse, des fichiers de données, ... les fonctions WLangage de gestion HFSQL manipuleront ces éléments dans un contexte indépendant. Ces paramètres peuvent être modifiés dans les options avancées du composant. Pour plus de détails, consultez l'aide en ligne.

2 Créer et générer un composant externe

La création d'un composant doit être effectuée à partir d'un projet contenant tous les éléments nécessaires au fonctionnement du composant. ***Il est donc conseillé d'utiliser un projet spécifique pour créer chaque composant.***

La création d'un composant se fait en plusieurs étapes :

1. Développement des éléments du composant.
2. Création du composant.
3. Définition des options avancées du composant.
4. Génération du composant.

Le composant peut alors être distribué et ré-utilisé.

3 Distribuer un composant externe

3.1 Présentation

Lorsque le composant a été créé, testé et généré, il peut être mis à disposition des développeurs. Plusieurs méthodes sont possibles :

- ***utilisation directe du composant***
  Exemple : le composant est directement utilisé sur le poste du développeur qui a créé le composant.
- ***distribution simple***, en fournissant directement les fichiers nécessaires.
  Exemple : le composant est destiné à être utilisé à l'intérieur d'une même société, par plusieurs développeurs. Dans ce cas, les fichiers nécessaires peuvent être copiés sur un disque réseau.
- ***procédure d'installation du composant*** (avec une procédure d'installation du composant identique à celle utilisée pour les applications).
  Exemple : Ce mode de distribution est conseillé si le composant est destiné à être vendu ou à être distribué à grande échelle, avec des mises à jour régulières.

3.2 Utilisation directe du composant

Le composant est créé et utilisé sur le même poste.

Lors de l'importation du composant dans un projet WINDEV, il suffit de sélectionner le fichier WDI correspondant au composant. Ce fichier est disponible dans le répertoire EXE du projet composant.

Attention : Si le composant utilise des fichiers spécifiques (fichiers de données, fichiers texte, ...), il est nécessaire de créer un fichier <NomComposant>.WDO après la génération du composant. Ce fichier contient la liste des fichiers externes (fichiers de données, fichiers texte, ...) utilisés par le composant.

Ces fichiers référencés dans le fichier <NomComposant>.WDO seront automatiquement recopiés dans le répertoire EXE du projet utilisant le composant WINDEV.

3.3 Distribution simple d'un composant

La distribution simple d'un composant consiste à fournir uniquement les fichiers nécessaires au fonctionnement du composant par simple copie. Les fichiers sont copiés dans un répertoire spécifique. Ce répertoire peut par exemple être présent sur un serveur réseau.

Lors de l'importation du composant dans un projet WINDEV, il suffit de sélectionner le fichier WDI correspondant au composant dans le répertoire de distribution.

Pour distribuer un composant, il est nécessaire de fournir :

- Les fichiers automatiquement générés par WINDEV (fichier <NomComposant>.WDK et fichier <NomComposant>.WDI).
- Si nécessaire, les fichiers spécifiques manipulés par le composant (fichiers de données, fichiers d'initialisation, ...) ainsi que le fichier <NomComposant>.WDO.
  Le fichier texte <NomComposant>.WDO contient la liste des fichiers qui doivent être fournis avec le composant. Pour plus de détails sur ce fichier, consultez le paragraphe "Le fichier WDO", page 80.

3.4 Distribution grâce à une procédure d'installation

La diffusion des composants à l'aide d'une procédure d'installation consiste à fournir un programme d'installation aux utilisateurs du composant WINDEV. Ce programme installe tous les fichiers nécessaires à l'utilisation du composant dans le répertoire spécifié par l'utilisateur.

Ce mode d'installation permet de gérer automatiquement :

- le fichier WDO et l'installation des fichiers spécifiques (fichiers de données, ...) utilisés par le composant.
- l'installation d'outils spécifiques (MDAC, driver ODBC pour HFSQL, ...)
- la mise à jour automatique des fichiers de données utilisés par le composant (si nécessaire).
- la désinstallation du composant.

Pour proposer une procédure d'installation pour un composant :

1. Créez la procédure d'installation du composant : sous le volet "Projet", cliquez sur "Procédure d'installation".
2. Dans les différents plans de l'assistant, spécifiez :

- le support d'installation du composant
- les langues proposées dans la procédure d'installation
- le répertoire d'installation par défaut du composant
- les modules optionnels à installer. Il est possible de modifier la liste des fichiers à distribuer avec le composant. Le fichier WDO sera automatiquement créé à partir de cette liste.
- ...

Par défaut, les fichiers nécessaires à l'installation d'un composant sont créés dans le sous-répertoire INSTALL COMPO du projet.

Rappel : Lors de la création de l'installation d'une application, les fichiers nécessaires sont créés dans le répertoire INSTALL du projet.

4 Utiliser un composant externe dans une application

Un composant peut être ré-utilisé à tout moment dans n'importe quelle application WINDEV : il suffit de connaître le répertoire des fichiers du composant et de l'importer dans le projet. L'importation du composant peut également se faire via le GDS.

Lorsqu'une nouvelle version d'un composant est disponible, il suffit d'installer les nouveaux fichiers de cette mise à jour dans le répertoire d'installation du composant (selon le mode d'installation utilisé). Pour plus de détails sur les différents modes d'installation d'un composant, consultez "Distribuer un composant externe", page 76.

Si le composant a été publié et importé via le GDS, les mises à jour sont proposées à l'ouverture du projet utilisant le composant.

La création de l'exécutable (utilisant le composant) se fait de la même manière que pour une application classique WINDEV. Les DLLs nécessaires au fonctionnement du composant sont automatiquement détectées.

5 Déploiement d'une application utilisant un composant externe

5.1 Présentation

Le déploiement d'une application utilisant un ou plusieurs composants se fait de la même façon que le déploiement d'une application classique (option "Procédure d'installation" sous le volet "Projet").

Lors du déploiement d'une application utilisant un composant, les fichiers suivants sont automatiquement installés sur le poste de l'utilisateur :

- <NomComposant>.WDK,
- <NomApplication>.EXE,
- les fichiers nécessaires au fonctionnement du composant et de l'application.

5.2 Mise à jour des composants et des exécutables déployés

Lors de la mise à jour d'un composant et/ou d'un exécutable déployé, il est possible d'installer sur le poste de l'utilisateur final :

- soit l'exécutable de l'application, le composant et les fichiers nécessaires à leur fonctionnement.
- soit uniquement l'exécutable de l'application et les fichiers nécessaires au fonctionnement de l'application.
- soit uniquement le composant et les éventuels fichiers nécessaires à son fonctionnement.

Pour plus de détails sur la gestion de la compatibilité entre l'application et son composant, consultez l'aide en ligne.

Pour mettre à jour un composant en clientèle, deux possibilités existent :

1. La recompilation du projet hôte

Recompilez le projet qui utilise le composant et redistribuez l'application avec le composant. Dans ce cas, aucun problème de version ou de compatibilité ne sera rencontré. La recompilation est nécessaire dans les cas suivants :

- De nouvelles fonctions ont été ajoutées dans le composant et doivent être prises en compte.
- Les paramètres de certaines procédures ont été modifiés.
- Incompatibilité de la nouvelle version avec les anciennes versions du composant.
- ...

2. Distribution directe du fichier .WDK

Fournissez directement une mise à jour du composant (fichier .WDK) en clientèle sans recompilation du projet.

Cette possibilité s'applique en général lorsque :

- La nouvelle version sert à corriger un ou des dysfonctionnements d'une version antérieure.
- De nouvelles fonctions ont été ajoutées dans le composant mais ne sont pas nécessaires au bon fonctionnement de l'application.

6 Modifier un composant externe

6.1 Présentation

A tout moment, un composant créé et généré peut être modifié. Cette modification peut correspondre :

- soit à l'ajout d'éléments dans le composant.
- soit à la suppression d'éléments du composant.
- soit à la modification des droits sur les éléments du composant.
- soit à la modification d'un des éléments du composant.

Dans tous les cas, après cette modification, le composant doit être re-généré pour prendre en compte les modifications effectuées.

6.2 Les différents types de compatibilité

La gestion de la compatibilité est liée aux versions d'un composant. En effet, si des modifications ont été apportées au composant, les applications qui utilisent le composant en déploiement peuvent rencontrer des problèmes d'exécution si elles ne sont pas en phase avec ce composant.

La gestion de la compatibilité est une étape importante dans la modification d'un composant. Il existe 2 types de compatibilité :

- la ***compatibilité descendante*** : la version du composant utilisée (fichier .WDK) pour compiler les applications utilisant ce composant doit être toujours supérieure ou égale à la version actuellement utilisée en déploiement.
  Cette compatibilité est automatiquement gérée. Si le composant installé avec l'application est d'une version inférieure à celle utilisée par l'application, l'application refusera de fonctionner.
- la ***compatibilité ascendante*** : l'utilisation d'une nouvelle version d'un composant avec les projets compilés avec une ancienne version peut être interdite. Il est donc nécessaire de recompiler les projets pour utiliser correctement la nouvelle version du composant.

7 Caractéristiques avancées d'un composant

7.1 Documentation automatique

Un composant doit absolument être accompagné d'une documentation pour permettre sa réutilisation.

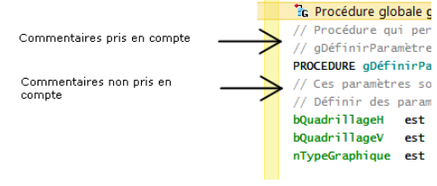
WINDEV simplifie la réalisation de cette documentation en proposant :

- ***une présentation générale du composant***. Cette présentation générale est saisie lors de la génération du composant. Lors de l'utilisation du composant dans une application WINDEV, cette présentation sera visible lors du survol du composant affiché dans le volet "Assistants, Exemples et Composants". Pour plus de détails, consultez l'aide en ligne.
- ***une génération automatique de la documentation technique*** à partir des commentaires insérés dans le code des éléments du composant.

Quels sont les commentaires de code pris en compte ?

Les commentaires pris en compte automatiquement pour la documentation du composant sont :

- Les commentaires placés en tête des procédures WLangage



- Les commentaires placés en tête des traitements suivants :
  - Code d'initialisation des fenêtres,
  - Code d'initialisation des états,
  - Code d'initialisation des classes,
  - Code d'initialisation des collections de procédures.

A quel moment la documentation est-elle générée ?

La documentation du composant est générée lors de la première génération du composant (option "Générer" sous le volet "Projet").

Lors de cette génération de la documentation :

- les commentaires existants dans le code sont utilisés pour créer la documentation.
- si aucun commentaire n'existe, WINDEV documentera automatiquement les éléments accessibles du composant en précisant les paramètres attendus en entrée et en sortie de chaque élément. Les commentaires correspondants sont automatiquement créés dans les différents éléments.

Remarque : A chaque re-génération du composant, il est possible de re-générer la documentation associée au composant (bouton "Regénérer" dans la fenêtre "Historique du composant").

Attention : Si vous utilisez le bouton "Regénérer", les modifications effectuées sous l'assistant de génération seront supprimées.

Comment disposer de la documentation d'un composant ?

La présentation générale du composant et la documentation technique du composant sont disponibles :

- lors de l'intégration d'un composant dans un projet : sous le volet "Projet", dans le groupe "Projet", déroulez "Importer" et sélectionnez l'option "Importer un composant externe".
- à tout moment, lors d'un double-clic sur l'icône du composant présent dans le volet "Explorateur de projets" ou par l'option "Description" du menu contextuel du composant.

La **documentation spécifique à un élément** du composant peut être obtenue en double-cliquant sur cet élément (volet "Explorateur de projet" ou graphe du projet) ou en appuyant sur la touche [F2] depuis le code de cet élément.

7.2 Visibilité d'un élément d'un composant

Lors de la création d'un composant, il est possible de définir les éléments du composant qui seront accessibles ou non par l'utilisateur du composant.

- Si l'élément est accessible, l'utilisateur du composant verra cet élément dans la liste des éléments du projet. Il pourra manipuler ces éléments par programmation (comme s'il s'agissait d'un élément créé dans le projet).
  Remarque : Le code de cet élément n'est cependant pas visible.
- Si l'élément n'est pas accessible, l'utilisateur ne saura même pas que cet élément existe.

Attention : Selon le mode de déclaration des éléments du projet (classe, Collection de procédures, ...), les éléments accessibles ou non peuvent varier.

7.3 Le fichier WDO

Lors de la génération du composant, deux fichiers sont automatiquement créés dans le répertoire EXE du projet en cours :

<NomComposant>.WDK	Contient tous les éléments à redistribuer (fenêtres, états, ...) avec le composant
<NomComposant>.WDI	Interface du composant. Ce fichier contient : - une aide pour l'utilisation du composant lors de sa réintégration - les éléments à l'utilisation du composant dans le projet (information de compilation, ...)

Lors de la diffusion du composant, ces deux fichiers doivent être redistribués.

Si le composant utilise des éléments supplémentaires (fichiers de données, ...), il est nécessaire d'ajouter dans le répertoire EXE du projet :

- un fichier <NomComposant>.WDO : ce fichier contient la liste des fichiers externes (fichiers de données, fichiers texte, ...) utilisés par le composant. Ces fichiers doivent être livrés et installés avec le composant.
- les fichiers devant être redistribués avec le composant. Ces fichiers peuvent être placés selon une sous-arborescence spécifique. Dans ce cas, le code du composant doit gérer l'accès à ces fichiers selon cette même arborescence.

Qu'est-ce que le fichier WDO ?

Le fichier <NomComposant>.WDO est un fichier au format TXT, pouvant être créé et modifié à tout moment. Ce fichier peut par exemple être créé et modifié avec l'éditeur de texte standard de Windows, NotePad.

Ce fichier contient la liste des fichiers externes (fichiers de données, fichiers texte, ...) utilisés par le composant et devant être livrés et installés avec le composant. Ces fichiers doivent être recopiés dans le répertoire EXE des projets utilisant le composant.

Ce fichier ".WDO" peut contenir :

- soit le ***nom complet du fichier***.
  Par exemple : C:\Composants\ComposantSélecteur\EtatInitial.INI
- soit le ***nom du fichier***. Ce fichier sera recherché dans le répertoire en cours du composant. Par exemple : EtatInitial.INI
- soit un ***nom de fichier utilisant un chemin relatif***. Les syntaxes possibles sont :
  - Répertoire\NomFichier.xxx pour indiquer un sous répertoire du répertoire en cours
  - .\NomFichier.xx pour indiquer le répertoire en cours
  - ..\NomFichier.xx pour indiquer le répertoire parent Par exemple : \ComposantSélecteur\EtatInitial.INI

Ce fichier sera utilisé lors de l'intégration du composant dans le projet. Les chemins précisés dans le WDO doivent correspondre aux chemins où les fichiers sont installés sur le poste de développement du composant.

Lors de l'intégration du composant, l'arborescence spécifiée dans le WDO sera conservée et reproduite dans le répertoire EXE du projet. Pour plus de détails, consultez le paragraphe "Utiliser un composant externe dans une application", page 77.

Exemple : Le composant "Code Postal" utilise un fichier de données "Cedex" (fichiers Cedex.fic et Cedex.ndx). Dans le projet de création du composant, ce fichier de données est présent dans le répertoire EXE du projet.

Pour que le composant soit livré et installé avec le fichier de données, il faut créer le fichier WDO dans le répertoire EXE du projet de création du composant. Ce fichier doit contenir les lignes suivantes :

```
.\CEDEX.Fic
.\CEDEX.NDX
```

Diffusion d'un composant avec WDO

Pour diffuser un composant utilisant un fichier WDO :

- Si vous n'utilisez pas de procédures d'installation pour le composant, il est nécessaire de fournir :
  - le fichier WDK,
  - le fichier WDI,
  - le fichier WDO,
  - tous les fichiers nécessaires référencés dans le fichier WDO.
- Si vous utilisez une procédure d'installation pour le composant, le fichier WDO sera automatiquement créé lors de la création de la procédure d'installation du composant.
  Il suffit pour cela dans l'assistant de :
  1. Demander la modification des fichiers à installer (option "Modifier la liste des fichiers à installer" dans les "modules additionnels").
  2. Sélectionner les fichiers supplémentaires à installer.
  Le fichier WDO sera automatiquement créé et installé avec le composant.

Modes de génération

Au-delà des applications Windows, WINDEV vous permet de générer de nombreux autres types de projets.

Applications

Les applications sont le mode de génération le plus fréquemment utilisé. Les applications construites avec WINDEV peuvent cibler les plateformes d'exécution suivantes : Windows (en 32 ou 64 bits) et Linux.

Services

WINDEV permet de générer des services pour Windows (en 32 ou 64 bits) et pour Linux (sous Linux, les services sont habituellement appelés des daemons). Un service est une forme particulière d'application, lancée en même temps que le système d'exploitation et ne comportant pas d'IHM.

Bibliothèques et patchs

Une bibliothèque est un fichier unique rassemblant plusieurs éléments d'un projet WINDEV : des fenêtres, des états, etc. Il est possible de générer des bibliothèques autonomes pouvant être utilisées par d'autres applications ainsi que des patchs correctifs pour une application en clientèle évitant d'avoir à réinstaller l'application complète pour une correction mineure.

Composants externes

Les composants externes sont des briques applicatives permettant de partager une ou plusieurs fonctionnalités spécifiques entre différentes applications. Un composant externe généré avec WINDEV peut également être utilisé dans un projet WEBDEV ou WINDEV Mobile.

Webservices SOAP ou REST

A partir d'un projet WINDEV, il est possible de générer un Webservice (également appelé service Web XML). Un Webservice expose un ensemble de fonctions (généralement une ou plusieurs collections de procédures) et les rend accessibles à travers le Web (ou un réseau privé) en utilisant les protocoles standard de communication HTTP, SOAP et REST.
Remarque : Un Webservice doit être déployé sur un Serveur d'Application WEBDEV pour être utilisable.

Applications et DLL d'assemblage DotNet

WINDEV permet la génération d'applications et d'assemblages pour la plateforme DotNet de Microsoft. Un assemblage DotNet généré avec WINDEV pourra être importé dans une application DotNet développée dans un autre langage d'une façon similaire à un composant externe entre deux applications WINDEV.

Applications et Applets Java

WINDEV permet de générer des applications Java ainsi que des Applets. Les applications Java peuvent ensuite être exécutées sur n'importe quelle machine possédant une machine virtuelle Java, quel que soit son système d'exploitation. Il est possible de cette façon de construire des applications fonctionnant indifféremment sous Windows, Mac OS ou même Solaris par exemple.
Une Applet Java est une application destinée à fonctionner à l'intérieur d'un navigateur Web.

Applications Windows Store Apps

WINDEV permet de générer des applications Windows Store Apps. Ces applications peuvent être utilisées en mode tuile ou distribuées via Windows Store.

Création de l'exécutable en pratique

Lorsque votre application est totalement décrite (fenêtre, états, ...), il est nécessaire de créer le programme exécutable correspondant.

Le programme exécutable (fichier ".EXE") pourra être directement exécuté : il ne sera pas nécessaire de lancer l'application depuis WINDEV.

Lorsque le programme exécutable est créé, il est possible de déployer l'application chez vos utilisateurs finaux en créant le programme d'installation.

1 Créer le programme exécutable d'une application WINDEV

Pour créer un programme exécutable :

1. Sélectionnez si nécessaire la configuration de projet correspondant à l'exécutable à créer :

- Configuration de projet pour un exécutable Windows 32 bits
- Configuration de projet pour un exécutable Windows 64 bits
- Configuration de projet pour un exécutable Linux

2. Sous le volet "Projet", cliquez sur "Générer".

3. L'assistant de création de l'exécutable s'ouvre.

4. Spécifiez :

- les ***options générales*** de l'exécutable (nom de l'exécutable, icône du programme exécutable, Splash screen ...).
- le ***mode de fonctionnement*** de l'exécutable :
  - *Multi-instance* : Ce programme pourra être lancé plusieurs fois depuis un même poste : plusieurs instances seront ouvertes en même temps.
  - *Mono-instance* : Ce programme pourra être lancé une seule fois depuis un même poste : une seule instance sera ouverte en même temps. Si le développeur tente de lancer une autre instance du programme, l'instance déjà en cours sera activée.
- le ***mode d'utilisation de la bibliothèque***. La bibliothèque est optionnelle. Elle peut être créée dans un fichier séparé (extension ".WDL") ou intégrée dans l'exécutable.
  L'intégration de la bibliothèque dans le programme exécutable a l'avantage de n'avoir qu'un seul fichier à installer ou à mettre à jour, mais cela augmente la taille du fichier ".EXE" créé.
- le ***message d'erreur*** qui sera affiché si une erreur de programmation survient.
- l'utilisation ***des mises à jour par Patch***. Il suffira alors de copier le fichier correspondant au Patch dans le répertoire de l'exécutable.
- les langues gérées par l'exécutable.
- les ***fichiers contenus dans la bibliothèque*** (si elle est utilisée).
- l'intégration des composants utilisés dans le projet à l'exécutable.
- ***l'emplacement des fichiers de données du groupware utilisateur*** (uniquement si votre application gère le groupware utilisateur).
- ***l'emplacement des fichiers de données*** à utiliser par l'exécutable.
- les ***informations de version***. Ces informations seront affichées dans l'onglet "Version" de la fenêtre des propriétés de l'exécutable (accessibles depuis l'explorateur Windows). Le ***numéro de version*** peut être automatiquement ***incrémenté*** à chaque création d'exécutable.
- le mode d'utilisation du framework WINDEV (utilisation du framework propre à l'application ou du framework commun à toutes les applications WINDEV installées sur le poste).
- le manifeste à utiliser (pour une exécution sous Windows Vista et supérieur uniquement).
- s'il est nécessaire de ***réaliser une sauvegarde du projet***.

5. Validez la description de l'exécutable. L'exécutable est automatiquement généré selon les informations fournies. Il est possible de lancer directement l'exécutable après sa création.

Application Java en pratique

1 Présentation

WINDEV permet de générer des applications Java sans pour autant connaître le langage Java. Tout le développement de l'application se fait en WLangage, comme pour toute application créée avec WINDEV.

1.1 Qu'est-ce qu'une application Java ?

Une application JAVA est entièrement compilée en JAVA. Si cette application manipule une base de données, cette base doit disposer d'un driver JDBC (MySQL, Oracle,...). Ce driver est spécifique à chaque base de données (et peut être trouvé sur le site de l'éditeur de la base de données).

L'accès à une base de données HFSQL se fait nativement (uniquement sous Windows et Linux) via les fonctions SQL du WLangage. Aucun pilote supplémentaire n'est requis.

Une application Java correspond à un fichier ".jar" et nécessite le runtime Java pour pouvoir être exécutée.

Pour compiler (en Java) un projet Java réalisé avec WINDEV, il est nécessaire d'installer sur le poste de développement le J2SDK de Oracle (version 1.4.2 ou supérieure nécessaire). Une version est disponible sur le CD d'installation de WINDEV. Il est également possible de télécharger ce programme sur le site de Oracle.

1.2 Pourquoi développer une application Java avec WINDEV ?

Vous possédez une application WINDEV qui fonctionne sous Windows et vous voulez la rendre accessible quel que soit le système d'exploitation ? Générez une application Java et ainsi les utilisateurs travaillant sous un système MAC OS X par exemple pourront également utiliser votre application.

Vous pourrez ainsi partager vos applications réseau.

1.3 Principe de génération d'une application Java avec WINDEV

Comme indiqué ci-dessus, il n'est pas nécessaire de connaître le langage Java pour développer une application Java avec WINDEV. Vous développez votre application en WLangage comme toute autre application WINDEV et lors de la compilation du projet, les fenêtres et les fonctions du WLangage sont automatiquement converties en Java.

Remarque : le nombre de fonctions WLangage utilisables dans un projet Java est limité dans cette version. Ce nombre de fonctions est amené à croître régulièrement.

2 Fonctionnalités pour un projet Java

2.1 Création d'un projet Java

La création d'un projet Java est effectuée de la même façon que tout autre projet WINDEV :

1. Cliquez sur parmi les boutons d'accès rapide de WINDEV. La fenêtre de création d'un nouvel élément s'affiche : cliquez sur "Projet".

2. L'assistant de création de projet se lance.

3. Dans un des premiers écrans de l'assistant, vous pouvez choisir le type de génération de votre projet. Il suffit de sélectionner "Java" et de suivre l'assistant.

Par défaut, un projet WINDEV pour Java correspond à une configuration spécifique du projet. Il est possible de modifier à tout moment le type de génération du projet. Il suffit de créer une nouvelle configuration de projet.

Dès que le projet est configuré comme un projet Java, tous les outils disponibles avec WINDEV sont automatiquement paramétrés pour le projet Java :

- Le RAD fenêtre ou le RAD application complète est spécifique à une base de données manipulée par une application Java.

- Le compilateur indique si votre projet contient des fonctions ou champs non utilisables en Java. Un message apparaît en violet dans le volet "Erreurs de compilation".
- La création de l'archive permet de générer un fichier ".jar" dans le répertoire EXE du projet.
- Le module d'installation manipule uniquement les fichiers nécessaires au déploiement de l'application Java.

Remarque : il est conseillé de créer un nouveau projet WINDEV pour développer une application Java plutôt que de transformer une application WINDEV en Java.

2.2 RAD JAVA

Comme sur tout projet associé à une analyse, vous pouvez utiliser le RAD pour générer les fenêtres de votre application.

Le RAD prend en compte toutes les spécificités de la génération Java (champs, traitements, fonctions utilisables dans une application WINDEV générée en Java).

Quel que soit le type de fenêtre à générer, le premier écran de l'assistant permet de spécifier le type de connexion à la base de données :

- HFSQL (sous Windows et Linux) : l'assistant classique de création de fenêtres est proposé.
- JDBC : vous pouvez alors spécifier les paramètres de connexion à la base de données utilisant le driver JDBC.

2.3 Test d'un projet Java sous l'éditeur

Lors du test d'un projet Java ( ), le projet est compilé et lancé sous Windows. Le projet n'est pas compilé en Java.

Pour exécuter l'application Java, il est nécessaire de lancer le fichier ".Jar" présent dans le répertoire EXE de votre projet.

2.4 Compiler un projet Java sous WINDEV

La compilation d'un projet Java permet de :

- connaître les erreurs de programmation présentes dans le code WLangage
- visualiser les champs, traitements, fonctions WLangage non disponibles en Java. Ces erreurs apparaissent en violet dans le volet "Erreurs de compilation".
  Pour plus de détails sur les éléments WINDEV non disponibles dans une application Java, consultez l'aide en ligne.

Remarque : comme indiqué dans la partie "Qu'est-ce qu'une application Java ?", page 85, il est nécessaire d'installer le J2SDK de Sun sur le poste de développement pour compiler votre projet en Java.

2.5 Créer une archive Java

WINDEV permet de :

- créer et compiler sous un compilateur Java les différents fichiers Java nécessaires à l'application
- créer une archive Java directement exécutable :
  - une application Java classique,
  - une applet Java,
  - une application Java Web Start.
- utiliser les fenêtres du projet depuis vos propres fichiers sources Java. Dans ce cas, il est possible de générer le projet Eclipse correspondant.

Pour créer l'archive Java, il suffit de lancer la création de l'exécutable : sous le volet "Projet", dans le groupe "Génération", cliquez sur "Générer". Un assistant vous guide alors dans la création de l'archive Java.

2.6 Déployer une application Java

Comme pour toute application, WINDEV permet de créer un programme d'installation pour une application Java. Cette procédure d'installation n'est actuellement disponible que pour Windows. Le programme d'installation sera un exécutable et permettra d'installer les fichiers nécessaires ainsi que le runtime Java (JRE). Pour les autres plateformes, il est actuellement nécessaire de déployer vos fichiers manuellement.

Pour lancer l'assistant de création d'installation, sous le volet "Projet", dans le groupe "Génération", cliquez sur "Procédure d'installation".

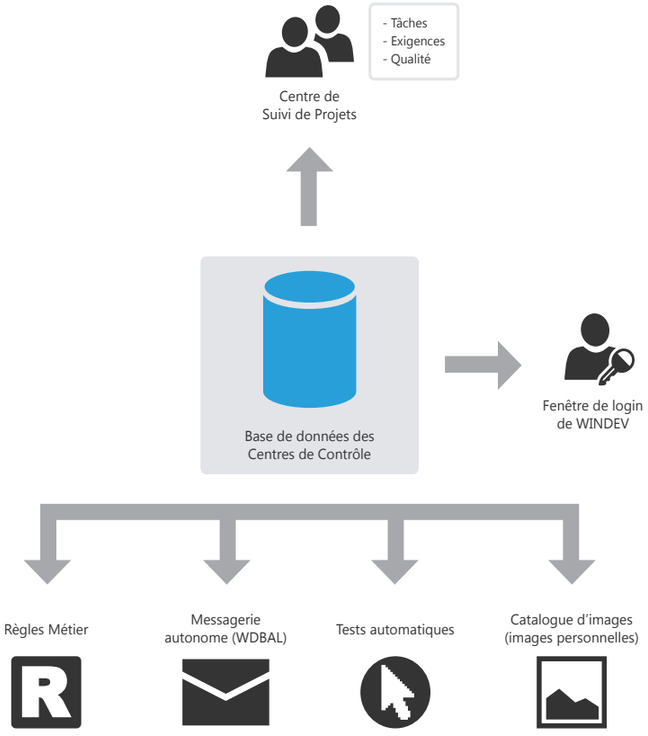
Pour plus de détails, consultez l'aide en ligne (mot-clé : "Java").

Centre de Suivi de Projets

Pour gérer le cycle de vie de vos projets de développement, WINDEV vous propose d'utiliser le Centre de Suivi de Projets. Le Centre de Suivi de Projets permet de :

- Gérer les exigences d'un projet,
- Gérer le suivi d'un projet (planning des tâches à réaliser),
- Gérer les bugs et les évolutions signalés par vos utilisateurs sur un projet.

Le Centre de Suivi de Projets utilise une base de données (HFSQL Classic ou Client/Serveur) : la base de données des Centres de Contrôle. Cette base de données est partagée par de nombreux outils disponibles dans WINDEV :



Remarque : Lors de l'installation de WINDEV, le programme d'installation propose :

- soit de créer la base des Centres de Contrôle (base de données du Centre de Suivi de Projets). Cette base sera automatiquement créée au format HFSQL Classic dans le répertoire spécifié.
- soit de partager une base de données des Centres de Contrôle existante.

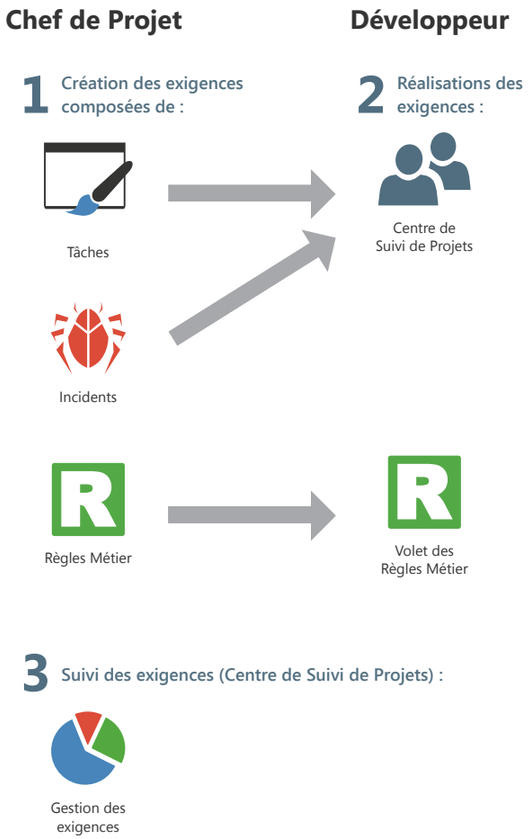
Gestion des exigences

Une exigence est un besoin sur ce qu'un produit (ou un service) devrait faire.

Le Centre de Suivi de Projets permet à un chef de projet de gérer les exigences d'un projet de développement.

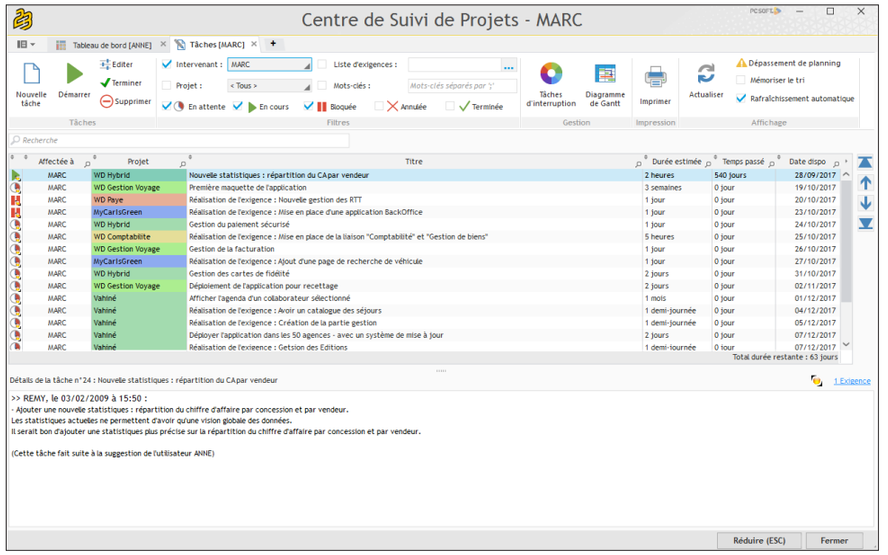
Il suffit pour cela de :
- définir les différents intervenants du projet.
- définir les exigences (avec les différents éléments qui leur sont associés).

Chaque développeur effectue les différentes tâches qui lui sont affectées.
Le chef de projet peut à tout moment suivre l'état d'avancement du projet.



Gestion des tâches

Le Centre de Suivi de Projets permet aux différents intervenants d'un projet de gérer leur planning des tâches. Ces tâches peuvent être liées à des exigences et correspondre à plusieurs projets.



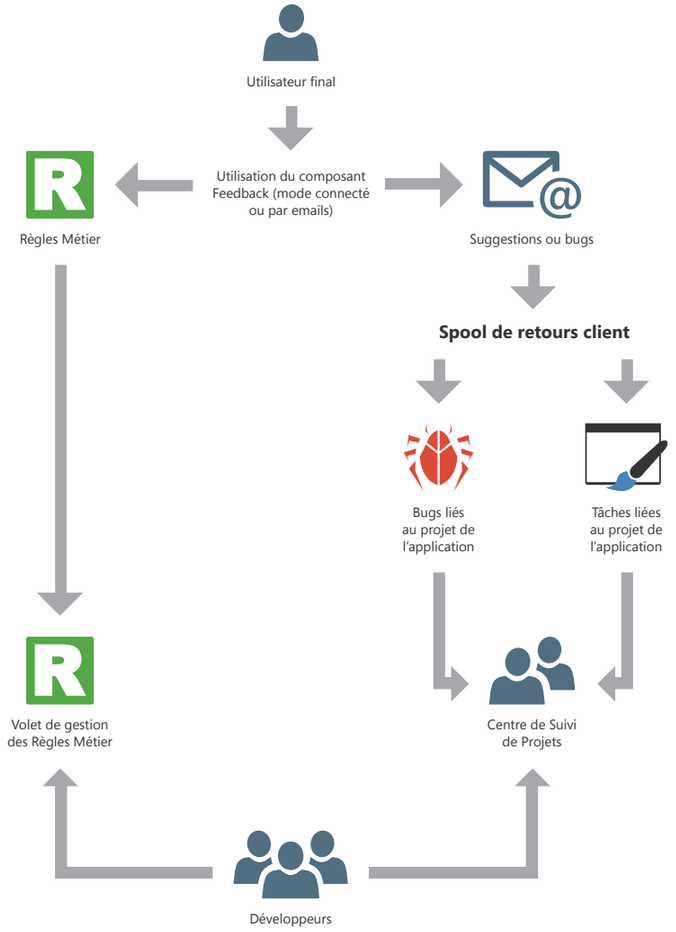
Après avoir listé l'ensemble des tâches d'un projet, le Centre de Suivi de Projets s'occupe de tout. La saisie du temps passé est quasi automatique, elle ne demande aucune action spécifique et ne génère aucune contrainte particulière.

Lors de l'ouverture du projet concerné, le Centre de Suivi de Projets demande ou indique la tâche en cours. Dès qu'une tâche est réalisée, il suffit d'indiquer que cette tâche est terminée et de spécifier la nouvelle tâche.

Une tâche peut être reliée à un élément du projet (fenêtre, état, ...). Chaque fois que l'élément concerné est ouvert, le temps passé sur cet élément est décompté et mémorisé dans le Centre de Suivi de Projets. Inversement, depuis la liste des tâches, il est possible d'ouvrir automatiquement l'élément correspondant à la tâche que vous voulez réaliser.
Chaque développeur peut également visualiser directement sa liste de tâches dans le volet "Centre de Suivi de Projets" présent sous l'éditeur.

Gestion des suggestions et des incidents

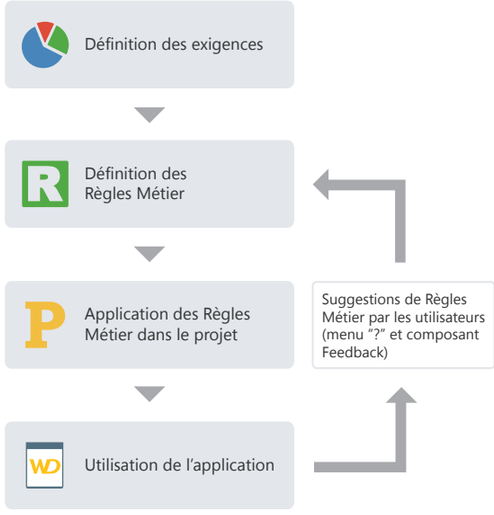
Les Centres de Contrôle permettent également de gérer les retours client sur vos applications.
Si votre application contient un menu d'aide automatique (menu '?' avec l'option "Envoyer des suggestions ..."), les utilisateurs de votre application peuvent simplement vous envoyer des suggestions, des incidents ou des règles métier.



Remarque : en cas d'erreur fatale pendant l'exécution de l'application, un rapport d'erreur peut être automatiquement généré et envoyé au développeur en utilisant le composant "Feedback" livré en standard avec WINDEV.

Gestion des règles métier

WINDEV permet de gérer des règles métier. Une règle métier permet de définir un mode opératoire précis ou de préciser un traitement particulier. Par exemple : le calcul d'un taux de TVA avec ses particularités, les règles de changement de statut d'un client, la formule de calcul d'un montant de frais de port, une commission commerciale, un taux de remise, un coefficient de vétusté, ...



Une règle métier peut être simple ou élaborée.

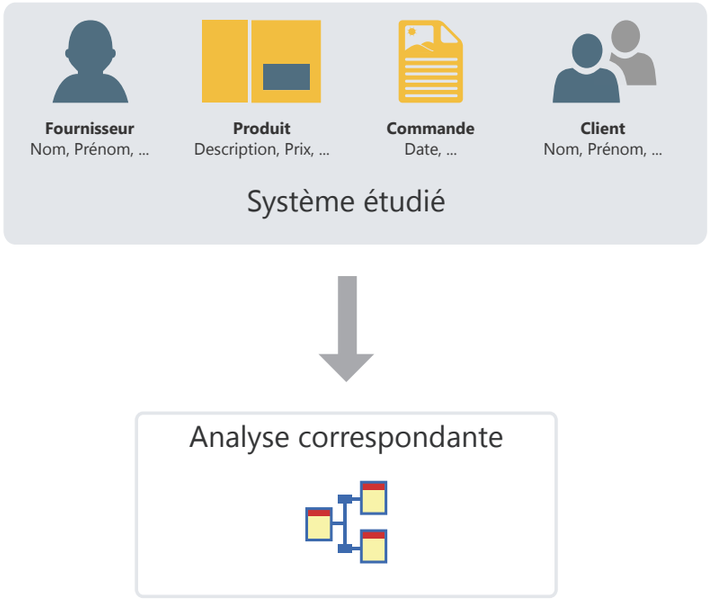
Les règles métier peuvent provenir :
- du cahier des charges (correspondant aux exigences).
- de suggestions faites par les utilisateurs de l'application.

Pendant le développement, les règles métier associées au projet sont affichées directement dans le volet "Règles métier" de l'environnement de développement. Ce volet affiche le nombre d'éléments du projet auxquels les règles s'appliquent et le pourcentage de règles actuellement réalisé.

DÉVELOPPEZ 10 FOIS PLUS VITE
PARTIE 3
Bases de données

Analyse : Structure de la base de données

Lorsqu'un projet WINDEV, WEBDEV ou WINDEV Mobile utilise des fichiers de données, ce projet doit être associé à une analyse. Une analyse permet de décrire les structures de données (fichiers, rubriques, ...) utilisées dans votre projet.



L'éditeur d'analyses permet de créer très simplement une analyse.
L'analyse d'un projet WINDEV correspond au MLD (Modèle Logique des Données). Toute la structure et l'organisation des données sont décrites : les données sont regroupées par fichier de données. Chaque fichier de données contient plusieurs données appelées rubrique.

Dans l'analyse, il est possible de lier la description d'un fichier de données à un type de fichier de données (HFSQL, Oracle, ...).

L'analyse en pratique

1 Présentation

Lorsqu'un projet utilise des fichiers de données, ce projet doit être associé à une analyse. Une analyse permet de décrire les structures de données (fichiers, rubriques, ...) utilisées dans votre projet.

Pour décrire une analyse, deux méthodes sont possibles :

- **1ère méthode** : Création directe de l'analyse (c'est-à-dire un Modèle Logique des Données (MLD)).
- **2ème méthode** : Création du Modèle Conceptuel des Données (MCD), puis création automatique de l'analyse à partir du MCD. Pour plus de détails, consultez l'aide en ligne.

Ce chapitre présente la première méthode.

2 Création d'un MLD (Modèle Logique des Données)

Dans WINDEV, les termes "MLD" et "Analyse" sont utilisés indifféremment pour définir la structure de la base de données associée à un projet.

2.1 Créer une analyse (ou MLD)

Pour créer un MLD :

1. Cliquez sur parmi les boutons d'accès rapide. La fenêtre de création d'un nouvel élément s'affiche : cliquez sur "Données" puis sur "Analyse". L'assistant de création de l'analyse est lancé.

2. Spécifiez :

- ***le nom et le répertoire de l'analyse***. L'analyse correspond à un fichier ".WDA". Par défaut, ce fichier sera créé dans le répertoire de l'analyse du projet (répertoire <Nom du projet>.ANA). Ce répertoire doit être accessible en lecture et en écriture.
- ***le libellé de l'analyse*** décrivant succinctement le sujet de l'analyse.
- ***si l'analyse doit être associée au projet en cours***.
- le type de la base de données manipulée par le projet.

3. La création du premier fichier de données est automatiquement proposée.

4. Créez l'ensemble des éléments (fichiers de données, rubriques et liaisons) de votre analyse.

2.2 Ajouter un fichier de données dans une analyse

L'analyse permet de décrire la structure des fichiers de données utilisés par le projet. Un fichier de données présent dans l'analyse peut être :

- Un nouveau fichier de données.
- Un fichier de données prédéfini, livré avec WINDEV.
- Un fichier de données importé depuis une base existante (utilisant par exemple un format spécifique).

Les paragraphes suivants détaillent comment créer un fichier de données.

Pour créer un nouveau fichier de données :

1. Sous le volet "Analyse", dans le groupe "Création", cliquez sur "Nouveau fichier". L'assistant de création des fichiers de données se lance.

2. Sélectionnez l'option "Créer une nouvelle description d'un fichier de données".

3. Spécifiez :

- ***le nom du fichier de données***. Ce nom est le nom logique du fichier de données. Il sera utilisé pour manipuler le fichier de données.
- ***le libellé du fichier de données*** résumant succinctement le sujet du fichier de données.
- ***la représentation d'un enregistrement*** dans le fichier de données. Cette représentation permet d'améliorer la lisibilité des questions posées lors de la description des liaisons. Cette option doit être précédée d'un article indéfini (UN ou UNE).
- si le fichier de données possède une ***rubrique de type "Identifiant automatique"***. La valeur de cette rubrique est unique pour chaque enregistrement et est automatiquement calculée par WINDEV.
- ***le type de la base de données associée au fichier***. Selon le type choisi, ce fichier de données sera manipulé par le moteur HFSQL, par un pilote OLE DB ou par un des accès natifs de WINDEV (SQL Server, Oracle, ...).

4. Le fichier de données créé devient le fichier en cours. La fenêtre de description des rubriques du fichier s'ouvre automatiquement. Vous pouvez décrire les rubriques du fichier de données.

Pour importer une description de fichier de données prédéfinie :

1. Sous le volet "Analyse", dans le groupe "Création", cliquez sur "Nouveau fichier". L'assistant de création des fichiers de données se lance.

2. Sélectionnez l'option "Sélectionner une description parmi des fichiers de données prédéfinis".

3. Choisissez le fichier de données prédéfini à créer (il est possible d'en sélectionner plusieurs). Ce fichier de données sera importé dans l'analyse en cours. Ce fichier de données pourra être modifié par la suite.

4. Sélectionnez les rubriques du fichier de données à conserver. Ces rubriques pourront être modifiées par la suite.

5. Le fichier de données importé est automatiquement inséré dans l'analyse en cours.

Remarque : Pour modifier un fichier de données ou les rubriques d'un fichier de données :

1. Sélectionnez le fichier de données.

2. Dans le menu contextuel, sélectionnez l'option "Description du fichier de données" ou l'option "Description des rubriques".

Pour importer une description de fichier existante, deux solutions sont possibles :

Solution 1 : Depuis l'éditeur d'analyses

1. Sous le volet "Analyse", dans le groupe "Création", cliquez sur "Nouveau fichier". L'assistant de création des fichiers de données se lance.

2. Sélectionnez l'option "Utiliser des fichiers de données d'une base existante", puis sélectionnez le type de la base de données.

3. Spécifiez la base de données source contenant les descriptions à importer et le type de cette base de données source. Selon le type choisi, renseignez les informations demandées.

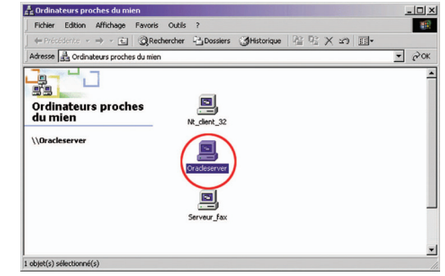
4. Sélectionnez les tables ou les fichiers de données dont la description doit être importée et validez.

5. Le fichier de données importé est automatiquement inséré dans l'analyse en cours.

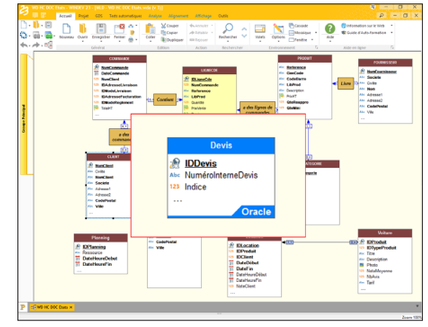
Solution 2 : Depuis l'explorateur de Windows

"Droppez" directement la description du fichier de données de l'explorateur Windows vers l'éditeur d'analyses.

Par exemple :



Ici, une base de données Oracle, sur le serveur. Il suffit de faire glisser le nom de la base ...



... pour en transférer la description sous l'éditeur d'analyses.

2.3 Créer une rubrique

Pour créer une rubrique :
1. Double-cliquez sur le fichier de données dans lequel la rubrique doit être créée. La fenêtre de description des rubriques du fichier de données s'ouvre.
2. Cliquez sur la première ligne vide du tableau des rubriques.
3. Spécifiez le nom, le libellé et le type de la rubrique directement dans le tableau.
4. Indiquez dans la partie droite de l'écran les détails concernant la nouvelle rubrique (type, taille, valeur par défaut, sens de parcours, ...).
5. Indiquez dans la partie basse de l'écran les détails concernant les informations partagées.
6. Validez la description de la rubrique.

Remarque : il est également possible de créer une rubrique à partir des méta-types proposés par WINDEV. Pour cela, il suffit de cliquer sur le bouton "+ M" à droite du tableau. La liste des méta-types disponibles apparaît.

2.4 Créer une liaison

Il est possible de créer différents types de liaison entre les fichiers. Pour plus de détails sur les liaisons, consultez le paragraphe "Caractéristiques des liaisons définies dans une analyse", page 98.

Pour créer une liaison :
1. Sous le volet "Analyse", dans le groupe "Création", cliquez sur "Nouvelle liaison". Le curseur de la souris se transforme en crayon.
2. Sélectionnez les deux fichiers de données à relier. La fenêtre de description de liaison s'ouvre automatiquement.
3. Pour définir les cardinalités entre les deux fichiers de données :
- soit sélectionnez directement les cardinalités parmi celles proposées (0,1 ; 1,1 ; 0,N ; 1,N).
- soit répondez aux questions posées. Les cardinalités seront mises à jour automatiquement.

4. Pour décrire des cardinalités avancées, cochez l'option "Afficher les cardinalités avancées" et répondez aux questions posées.
5. Saisissez le libellé de la liaison en décrivant succinctement le sujet de la liaison.
6. Spécifiez les clés à relier.
7. Précisez les règles d'intégrité. Ces règles permettent d'assurer la cohérence des données lors de la modification ou lors de la suppression d'une des clés de la relation.
8. Validez. La liaison est automatiquement créée.

3 Caractéristiques des liaisons définies dans une analyse

3.1 Fichier propriétaire et fichier membre

Dans le cas de liaison entre deux fichiers de données, il existe un fichier ***propriétaire*** et un fichier ***membre*** :
- le fichier propriétaire est propriétaire de la clé.
- le fichier membre est un membre des fichiers de l'analyse contenant une copie de la clé.

Pour gérer la liaison entre deux fichiers de données, la clé du fichier propriétaire est copiée dans le fichier membre.

Par exemple, la clé du fichier Fournisseur est copiée dans chaque enregistrement du fichier Produit. Plusieurs enregistrements du fichier Produit peuvent avoir la même clé du fichier Fournisseur :
- le fichier propriétaire est Fournisseur,
- le fichier membre est Produit.

La représentation d'une telle liaison est la suivante :

3.2 Les cardinalités

Les cardinalités permettent de dénombrer les liens entre les fichiers de données.

La cardinalité est définie en fonction des réponses aux deux questions suivantes :
1. Pour chaque enregistrement du fichier de données, à combien d'enregistrements de l'autre fichier de données **au minimum** cet enregistrement est-il relié ?

La réponse fournit la première partie de la cardinalité (***cardinalité minimale***) :
- si la réponse est "aucun", la cardinalité est du type 0,X.
- si la réponse est "un seul", la cardinalité est du type 1,X.

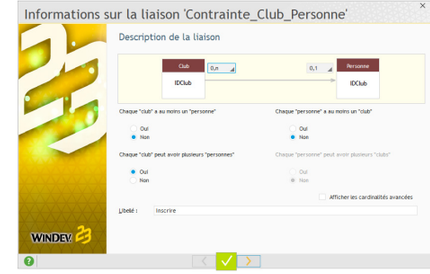
2. Pour chaque enregistrement du fichier de données, à combien d'enregistrements de l'autre fichier de données **au maximum** cet enregistrement est-il relié ?

La réponse fournit la deuxième partie de la cardinalité (***cardinalité maximale***) :
- si la réponse est "un seul", la cardinalité est du type X,1.
- si la réponse est "plusieurs", la cardinalité est du type X,N.

La réponse à ces deux questions définit la cardinalité qui peut être de type : 0,1 ; 0,N ; 1,1 ; 1,N

La description des cardinalités est fondamentale : elle est la base du respect de l'intégrité référentielle de la base de données.

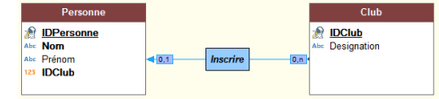
Pour plus de clarté, ces deux questions peuvent être décomposées en quatre questions. Par exemple, pour décrire une liaison entre le fichier Club et le fichier Personne :



- chaque "Club" a au moins un "Personne" : Oui/Non ?
- chaque "Club" peut avoir plusieurs "Personne" : Oui/Non ?
- chaque "Personne" a au moins un "Club" : Oui/Non ?
- chaque "Personne" peut avoir plusieurs "Club" : Oui/Non ?

Exemple de cardinalités :

Cet exemple présente deux cardinalités différentes :



- ***Cardinalité 0,1*** : Une personne peut être inscrite à un seul club de sport. Cette personne n'est pas obligée d'être inscrite à un club de sport.
- ***Cardinalité 0,N*** : Un club peut avoir aucune ou plusieurs personnes inscrites.

3.3 Les cardinalités avancées

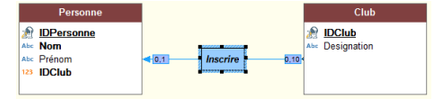
Il est possible de définir avec plus de précision les cardinalités. On parle alors de **cardinalités avancées**.

Ces cardinalités permettent de préciser exactement la cardinalité minimale et la cardinalité maximale.

Pour définir les cardinalités avancées, cochez l'option "Affichez les cardinalités avancées" dans la description de la liaison.

Exemple de cardinalités avancées :

Cet exemple présente deux cardinalités différentes :



- ***Cardinalité 0,1*** : Une personne peut être inscrite à un seul club de sport. Cette personne n'est pas obligée d'être inscrite à un club de sport.
- ***Cardinalité 0,10*** : Un club peut avoir aucune ou jusqu'à 10 personnes inscrites.

3.4 L'intégrité référentielle

L'intégrité référentielle d'une base de données correspond au respect des contraintes impliquées par les liaisons entre les fichiers de données.

L'intégrité référentielle consiste à vérifier par exemple que :

- si un enregistrement est supprimé dans le propriétaire, les enregistrements correspondants dans les membres sont également supprimés,
- si un enregistrement est ajouté dans un membre, il existe un enregistrement correspondant dans le propriétaire,
- si un enregistrement est modifié dans un propriétaire, la clé unique n'est pas modifiée, ...

La vérification de l'intégrité référentielle dépend de la nature de la liaison entre les fichiers de données.

Pour plus de détails sur le contrôle de l'intégrité référentielle, consultez l'aide en ligne.

3.5 Les différents types de liaisons

Les liaisons entre fichiers de données peuvent être de plusieurs types :

- parallèle
- facultative
- de complément
- partagée
- complexe

Liaison parallèle

Dans le cas d'une liaison parallèle, ***chaque enregistrement*** d'un fichier de données (Produit) est relié à ***un enregistrement*** dans un autre fichier de données (Information), et ***réciproquement***.

Pour gérer une liaison parallèle, l'identifiant du fichier Produit est copié dans le fichier Information. Cet identifiant est également clé unique dans le fichier Information.

Le fichier propriétaire est alors Produit et le fichier membre est Information.

Remarque : Ce type de liaison est rare car les deux fichiers peuvent être regroupés en un seul.

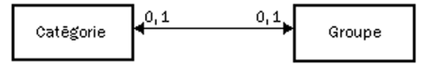
Le parallélisme des enregistrements dans le fichier est respecté si les opérations suivantes sont exécutées simultanément sur les deux fichiers :

- création d'un enregistrement
- suppression d'un enregistrement
- réindexation avec compactage

Liaison facultative

Dans le cas d'une liaison facultative :

- ***Chaque enregistrement d'un fichier de données*** (Catégorie) ***n'a aucun enregistrement ou un seul enregistrement*** associé dans un autre fichier de données (Groupe).
- ***Chaque enregistrement*** de l'autre fichier de données (Groupe) ***n'a aucun enregistrement ou un seul enregistrement*** associé dans le premier fichier de données (Catégorie).

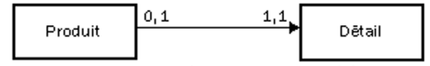


Une liaison facultative est réalisée en copiant l'identifiant de chacun des fichiers dans l'autre fichier.

Liaison de complément

Dans le cas d'une liaison de complément :

- ***Chaque enregistrement*** d'un fichier de données (Produit) ***n'a aucun enregistrement ou un seul enregistrement*** associé dans un autre fichier de données (Détail).
- ***Chaque enregistrement*** de l'autre fichier de données (Détail) a ***nécessairement un enregistrement*** associé dans le premier fichier de données (Produit).



Les liaisons de complément sont assez fréquentes. Elles sont notamment utilisées lorsqu'un enregistrement peut avoir des compléments d'informations optionnels.

Pour gérer une liaison de complément, la clé du fichier Produit est copiée dans le fichier Détail. Pour assurer la cardinalité maximale de 1, elle reste clé unique. L'unicité de cette clé interdit l'insertion de plus d'un enregistrement dans Détail pour un enregistrement de Produit.

Le fichier propriétaire est Produit, le fichier membre est Détail.

Liaison partagée

Dans le cas d'une liaison partagée, un même enregistrement dans un fichier de données (Fournisseur) peut être partagé par plusieurs enregistrements dans un autre fichier de données (Produit).

Pour gérer une liaison partagée, la clé du fichier Fournisseur est copiée dans le fichier Produit. Elle devient clé multiple pour accélérer la vérification de l'intégrité.

Le fichier propriétaire est Fournisseur, le fichier membre est Produit.

Selon la cardinalité, il est possible de distinguer quatre types de liaisons partagées :

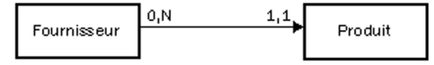
- Liaison partagée de cardinalité 0,n - 0,1
- Liaison partagée de cardinalité 0,n - 1,1
- Liaison partagée de cardinalité 1,n - 0,1
- Liaison partagée de cardinalité 1,n - 1,1

Liaison partagée de cardinalité 0,N - 0,1

Avec ce type de liaison :

- un propriétaire peut ne pas avoir de membre (un fournisseur n'a pas obligatoirement de produit).
- un membre peut ne pas avoir de propriétaire (un produit n'a pas obligatoirement de fournisseur).

Liaison partagée de cardinalité 0,N - 1,1



Avec ce type de liaison :

- un propriétaire peut ne pas avoir de membre (un fournisseur peut avoir aucun produit).
- chaque membre a un seul propriétaire (chaque produit a un seul fournisseur).

Remarque : Ce type de liaison est fréquent. WINDEV permet de créer automatiquement les fenêtres permettant de gérer les fichiers de données liés par une liaison de type 0,N - 1,1.

Liaison partagée de cardinalité 1,N - 0,1

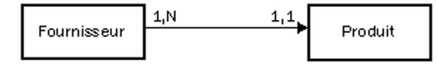
Avec ce type de liaison :

- chaque propriétaire a au moins un membre (un fournisseur a au moins un produit).
- un membre peut ne pas avoir de propriétaire (un produit peut avoir aucun fournisseur).

Liaison partagée de cardinalité 1,N - 1,1

Avec ce type de liaison :

- chaque propriétaire a au moins un membre (chaque fournisseur a au moins un produit).
- chaque membre a un seul propriétaire (chaque produit a un seul fournisseur).



Liaison complexe

Dans le cas d'une liaison "complexe", il est nécessaire de gérer un fichier de liaison, appelé ***fichier de relation***.

Le fichier de relation sera automatiquement créé. Il contiendra une clé unique composée des deux clés des fichiers de données reliés.

Le fichier de relation peut également contenir des informations spécifiques à la liaison.

Une liaison complexe se décompose alors en deux liaisons de type partagé.

Exemple de liaison complexe

Une commande (fichier Commande) peut contenir un ou plusieurs produits. Un produit (fichier Produit) peut être utilisé dans plusieurs commandes.

Dans ce cas, il faut avoir un fichier de liaison (fichier LigneCommande).

Le fichier LigneCommande contient :

- une clé unique composée des clés de Produit et de Commande.
- le nombre de produits commandés.

Les liaisons entre les fichiers de données sont les suivantes :

4 Manipulations possibles sur une analyse

WINDEV permet de réaliser les manipulations suivantes sur une analyse :

- Dupliquer/Copier un MLD : La duplication d'un MLD permet de disposer de deux MLD identiques de nom différent.
- Supprimer un MLD.
- Renommer un MLD.
- Associer un MLD à un projet.
- Agrandir ou réduire l'affichage d'un MLD sous l'éditeur.
- Déplacer l'affichage du MLD sous l'éditeur.

WINDEV permet de réaliser les manipulations suivantes sur un fichier de données présent dans l'éditeur d'analyses :

- Dupliquer/Copier un fichier de données.
- Supprimer un fichier de données.
- Renommer un fichier de données.

WINDEV permet de réaliser les manipulations suivantes sur une rubrique d'un fichier de données présent dans l'éditeur d'analyses :

- Dupliquer/Copier une rubrique.
- Supprimer une rubrique.
- Renommer une rubrique.

Pour plus de détails, consultez l'aide en ligne.

5 Génération de l'analyse (MLD)

La génération de l'analyse intervient avant la phase de programmation et après la phase de description des fichiers de données. Cette génération permet de :

- valider les modifications effectuées sur l'analyse (MLD).
- créer les modules nécessaires à la programmation.
- mettre à jour automatiquement les fichiers de données si nécessaire.

Tant que l'analyse (le MLD) n'a pas été générée, la description de l'analyse (les fichiers de données) ne pourra pas être utilisée dans le projet.

Pour lancer la génération de l'analyse, sous le volet "Analyse", dans le groupe "Analyse", cliquez sur "Génération".

La génération comporte trois étapes :

- Vérification des modifications effectuées et génération des fichiers de la description de l'analyse.
- Modification automatique des fichiers de données accessibles (fichiers présents dans le répertoire "EXE" du projet).
- Synchronisation du projet.

6 Gestion des versions de l'analyse

WINDEV permet de gérer les différentes versions de l'analyse :

1. Sous le volet "Analyse", dans le groupe "Analyse", déroulez "Génération".
2. Sélectionnez l'option "Gestion des versions".
3. Dans l'assistant, indiquez si vous voulez travailler sur l'analyse en cours ou sur une analyse spécifique.

Remarque : lorsqu'une analyse est endommagée, il est possible de sélectionner directement l'analyse endommagée pour restaurer par exemple une des versions précédentes.

Toutes les versions de l'analyse sont visualisées sous forme d'un graphique. Le carré jaune indique le numéro de version de l'analyse, le libellé à droite rappelle la date de la génération.

Entre deux versions figurent des "petits rectangles" qui signifient :

- un rectangle : il y a eu peu de modifications effectuées entre les deux versions.
- plusieurs rectangles : il y a eu plusieurs modifications effectuées entre les deux versions.

Pour visualiser le détail des modifications effectuées entre deux versions : Double-cliquez sur la ligne contenant les "petits rectangles" ou cliquez sur le bouton [Modifications].

Les options disponibles dans le gestionnaire de versions sont :

- Restauration d'une analyse.
- Annulation de la dernière génération.
- Remettre à 1 le numéro de version.

6.1 Restauration d'une analyse

Pour restaurer une analyse à partir du gestionnaire de versions :

1. Sélectionnez la version à restaurer.
2. Il est possible de :

- soit restaurer la version dans le répertoire spécifié. Une analyse indépendante est créée correspondant à la version choisie. Cette option est sélectionnée par défaut.
- soit écraser l'analyse en cours.

3. Validez. La restauration est effectuée.

Attention : Les fichiers de données correspondant à une analyse de numéro de version supérieur à la version restaurée ne pourront plus être ouverts. Il faut alors également restaurer des fichiers de données correspondant à la version de l'analyse restaurée ou supprimer les fichiers de données existants pour qu'ils soient recréés.

6.2 Annulation de la dernière génération

L'annulation de la dernière génération de l'analyse permet de restaurer l'analyse dans son état avant la dernière génération. Les modifications effectuées depuis ne sont pas appliquées.

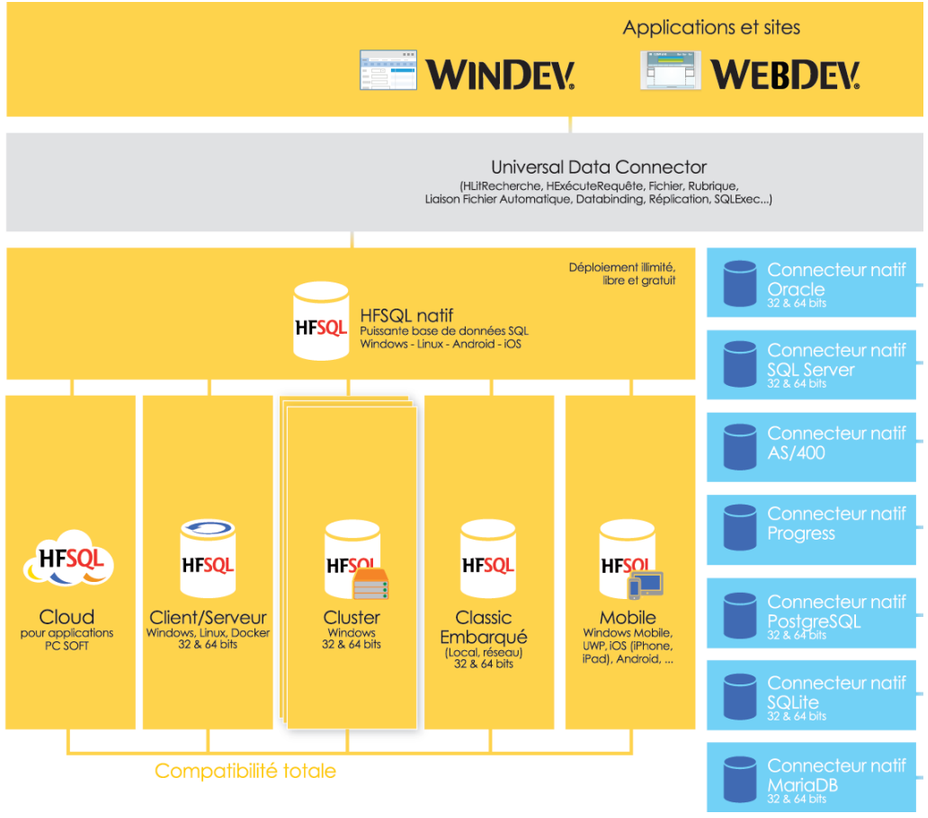
6.3 Remettre à 1 le numéro de la version

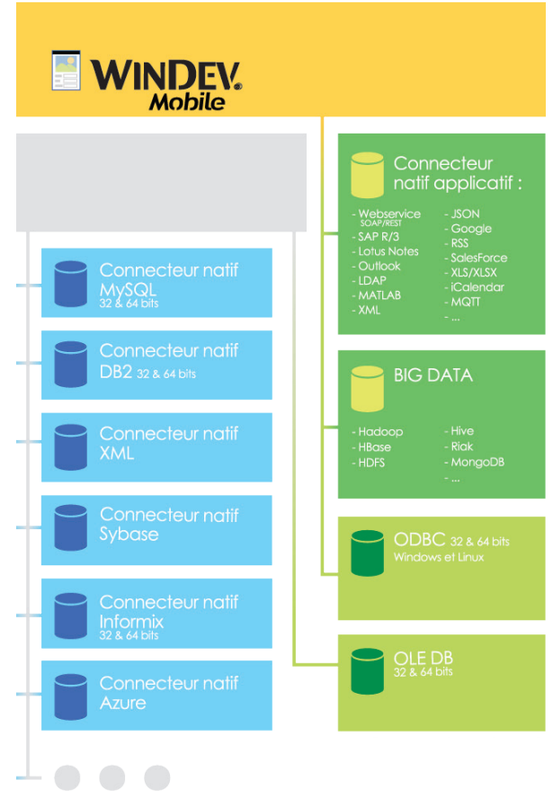
Il est possible de remettre le numéro de génération de l'analyse à "1" : sous le volet "Analyse", dans le groupe "Analyse", déroulez "Génération" et sélectionnez l'option "Remettre à 1 la version de l'analyse".

Dans ce cas le numéro de version de l'analyse est remis à un. Aucune action spécifique n'est réalisée sur les fichiers de données.

Les différents types de fichiers accessibles

WEBDEV, WINDEV et WINDEV Mobile proposent un accès simple à la majorité des bases de données du marché.



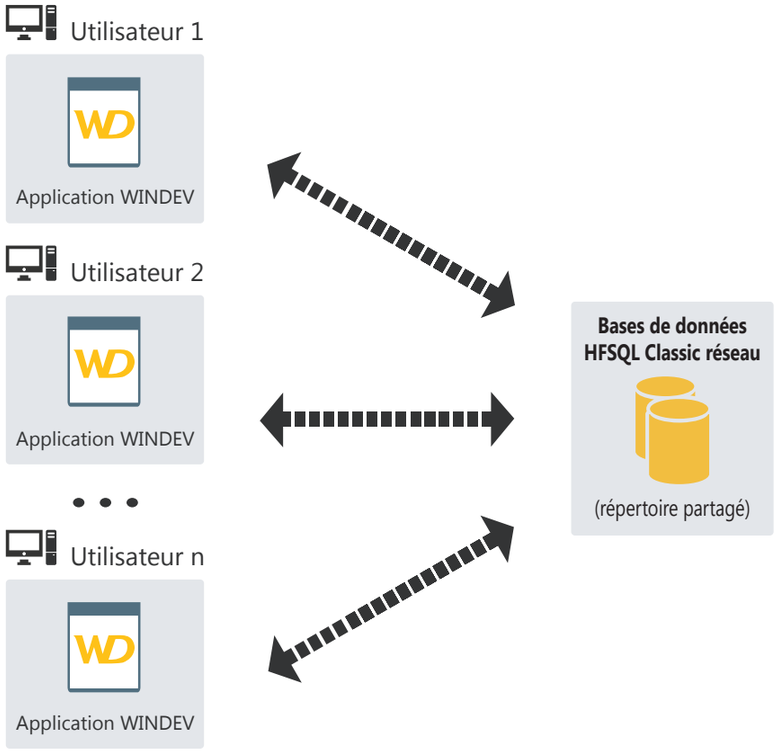


HFSQL Classic

HFSQL Classic est un puissant système de gestion de bases de données relationnelles (SGBD/R) utilisable en monoposte ou en réseau local. HFSQL Classic ne requiert pas l'installation d'un serveur de bases de données.

Les caractéristiques de HFSQL Classic sont les suivantes :

- Une application HFSQL Classic peut être exécutée sur différents postes utilisateurs répartis en réseau local.
- Les fichiers de données sont présents dans un répertoire accessible par les postes utilisateurs (par exemple, un répertoire partagé sur le réseau). Chaque poste utilisateur accède physiquement aux fichiers de données.
- Les traitements (requête, lecture/ajout dans un fichier de données, ...) sont réalisés sur chaque poste utilisateur.

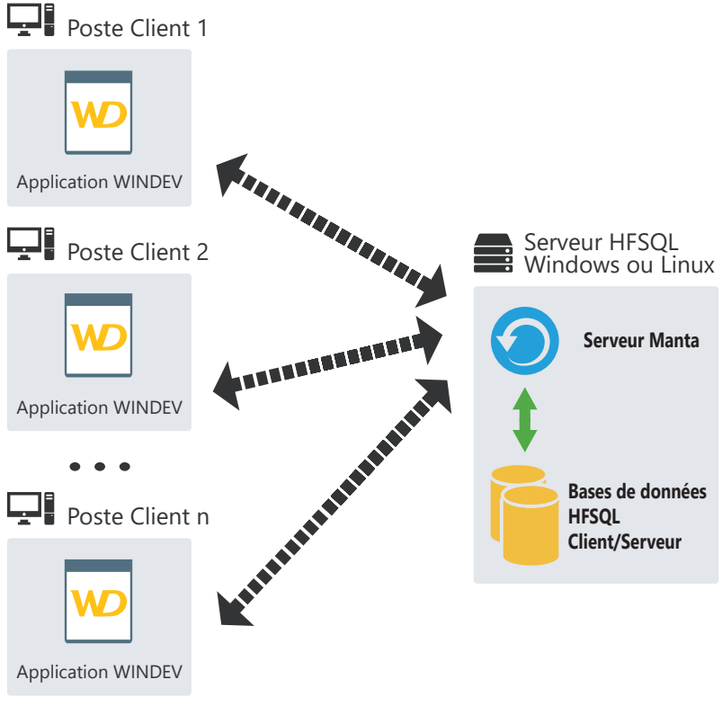


HFSQL Client/ Serveur

HFSQL Client/Serveur est un puissant système de gestion de bases de données relationnelles (SGBD/R) en mode Client/Serveur.

Les caractéristiques de HFSQL Client/Serveur sont les suivantes :

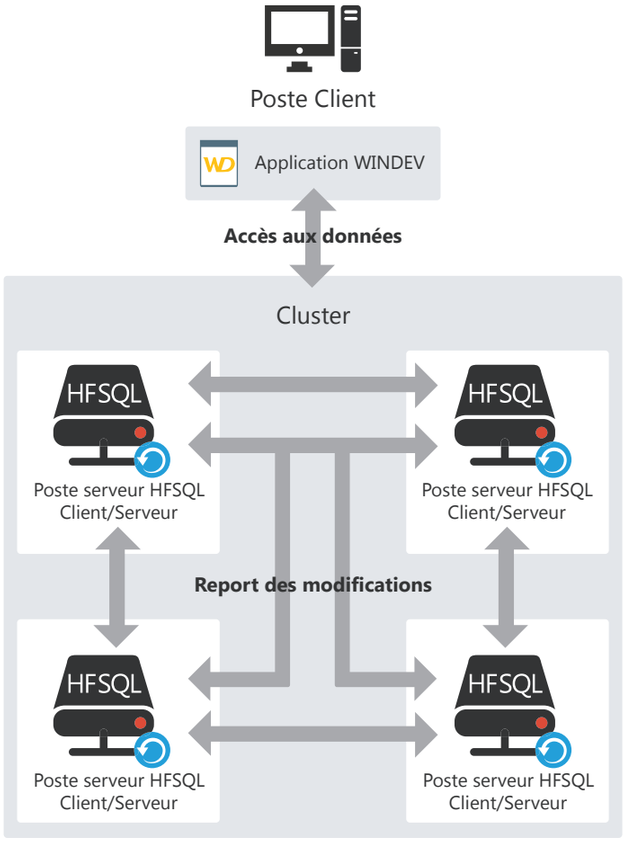
- Une application HFSQL Client/Serveur peut être exécutée sur différents postes utilisateur (appelés machines clientes) qui communiquent à travers le réseau avec le serveur de base de données.
- Les fichiers de données sont présents sur un poste serveur. Seul le serveur accède physiquement aux fichiers de données.
- L'ensemble des traitements (requête, lecture/ajout dans un fichier de données, ...) est réalisé sur le serveur.



Cluster HFSQL Client/Serveur

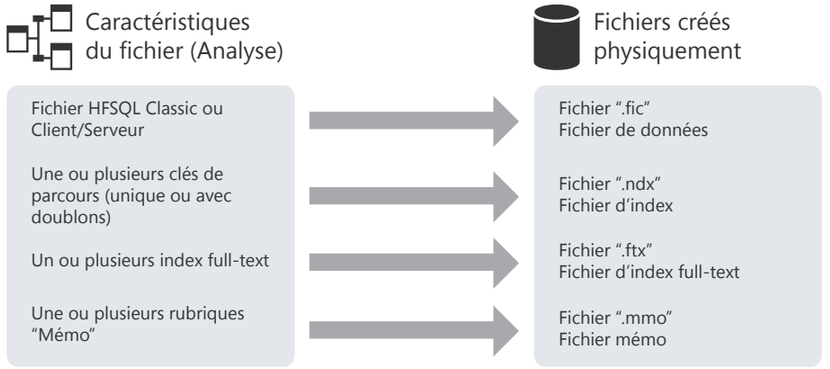
HFSQL Cluster est une extension du modèle de base de données de HFSQL Client/Serveur. Dans un cluster de bases de données, les différents serveurs HFSQL contiennent tous une copie des bases de données et sont synchronisés en temps réel.

- La charge en lecture peut être équilibrée entre différents serveurs.
- La configuration physique peut évoluer sans interruption pour les postes clients.
- En cas de crash d'un des serveurs, le client est automatiquement redirigé vers un serveur opérationnel.



HFSQL : les fichiers créés physiquement

L'éditeur d'analyses permet de décrire la structure des fichiers de données.
Selon les informations saisies sous l'éditeur d'analyses, différents fichiers sont créés physiquement.



Remarque : Ce schéma présente uniquement les principaux fichiers créés. D'autres fichiers spécifiques peuvent être créés si le fichier de données utilise la journalisation, les transactions ou encore la réplication.

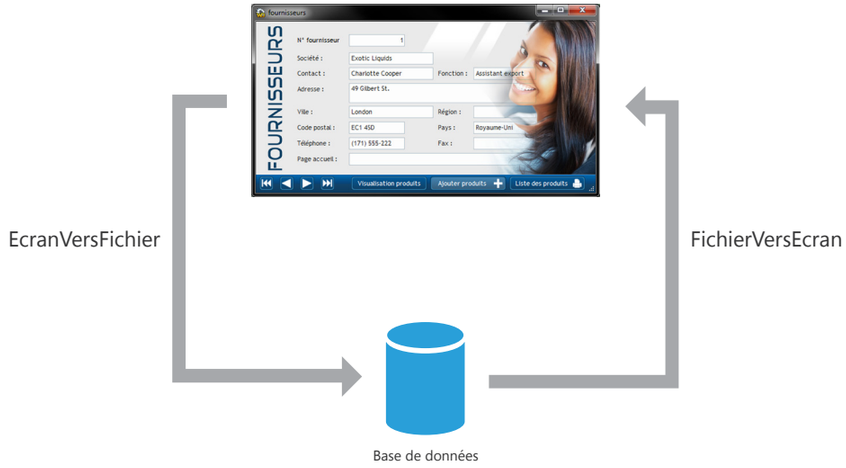
Associer les champs et les données

Une fenêtre peut afficher des informations provenant :
- d'une base de données : les champs sont directement liés aux rubriques des fichiers de données ou des requêtes disponibles dans la base de données.
- de variables présentes dans le code de l'application (variables globales à la fenêtre ou au projet ou paramètres passés à la fenêtre).

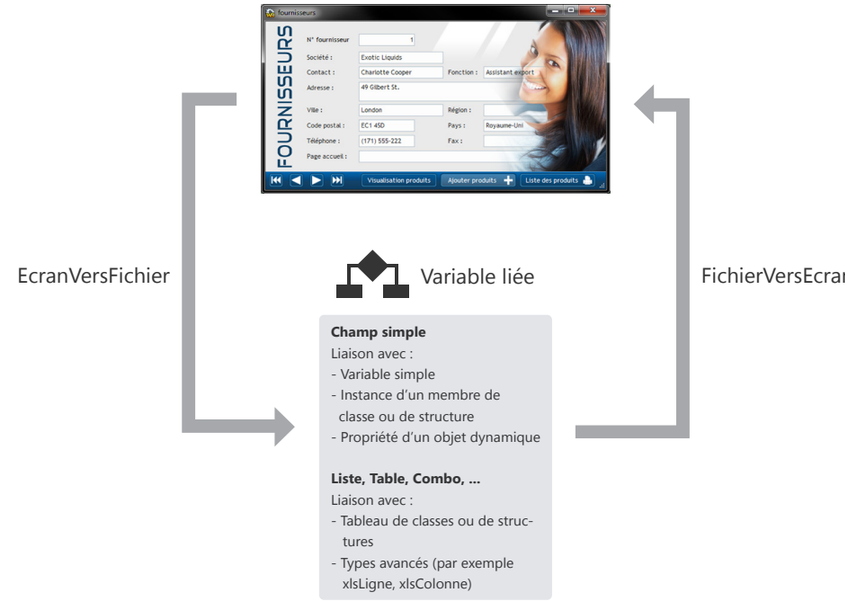
Pour afficher ces informations dans une fenêtre, il est nécessaire de lier les champs de cette fenêtre avec :
- les différentes rubriques de la base de données.
- les différentes variables WLangage disponibles.

La méthode d'affichage et de récupération des informations est très simple :
- La liaison d'un champ avec une rubrique ou une variable est définie sous l'éditeur de fenêtres, lors de la description du champ (onglet "Liaison").
- La fonction ***EcranVersFichier*** permet de mettre à jour soit l'enregistrement, soit la variable avec les données présentes à l'écran.
- La fonction ***FichierVersEcran*** permet de mettre à jour les données affichées à l'écran avec soit les informations enregistrées dans le fichier de données, soit les informations enregistrées dans la variable.

Liaison champ / rubrique



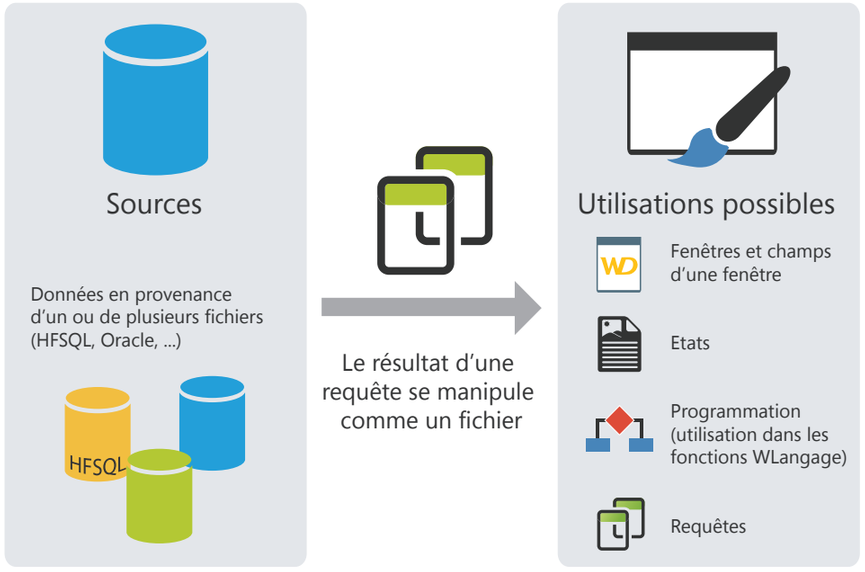
Liaison champ / variable



Les requêtes

Une requête sert à interroger une base de données pour visualiser, insérer, modifier ou supprimer des données. La structure de la requête définit les données manipulées. Une requête peut interroger un ou plusieurs fichiers de données.

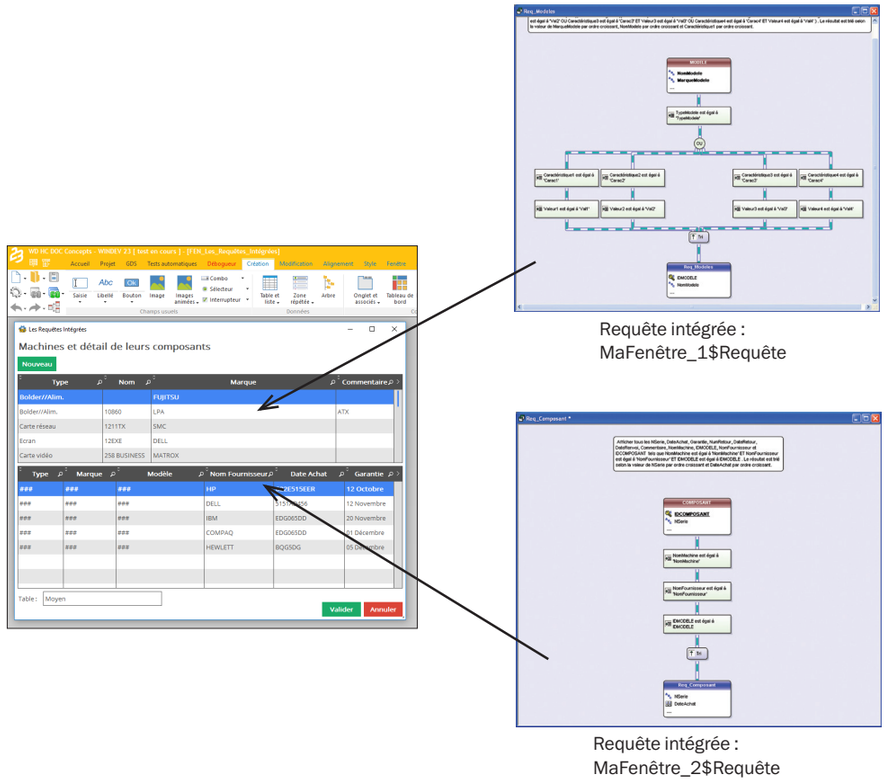
L'éditeur de requêtes permet de créer très simplement des requêtes, sans avoir à programmer.



Remarque : En programmation, une requête peut être manipulée de la même façon qu'un fichier de données. En particulier, elle peut être associée à un champ d'affichage (un champ Table par exemple) qui présentera les données retournées par la requête.

Les requêtes intégrées

Les champs d'une fenêtre peuvent être liés à un fichier de données ou à une requête existante. Il est également possible de lier ces champs à une requête créée lors de la conception du champ.



Requête intégrée : MaFenêtre_1$Requête

Requête intégrée : MaFenêtre_2$Requête

Dans ce cas, la requête est intégrée à la fenêtre. Elle est présente dans le fichier WDW correspondant à la fenêtre. Si le fichier WDW est copié (dans un autre projet par exemple), les requêtes intégrées utilisées par cette fenêtre seront également copiées.

Les champs Table/ Zone Répétée

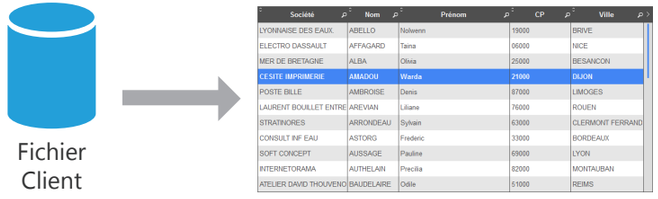
Les champs Table/Zone répétée peuvent être utilisés pour afficher un ensemble d'informations (par exemple, le contenu d'un fichier de données). Le contenu de ces champs peut être issu de trois sources différentes :

- Champs Table/Zone Répétée "Fichier accès direct",
- Champs Table/Zone Répétée "Mémoire",
- Champs Table/Zone Répétée "Fichier chargé en mémoire".

Remarque : Ces trois modes de remplissage vont être détaillés dans cette page pour le champ Table. Les mêmes concepts s'appliquent au champ Zone Répétée.

Champ Table "Fichier accès direct"

Un champ Table fichier de type accès direct permet d'afficher directement les données provenant d'un fichier de données ou d'une requête. La structure du fichier de données a été décrite sous l'éditeur d'analyses et les données ont été saisies par exemple dans l'application. Un parcours du fichier de données permet d'afficher les données dans le champ Table. Pour chaque ligne affichée, le fichier de données est lu : l'enregistrement lu est affiché dans une ligne du champ Table.



Société	Nom	Prénom	CP	Ville
LYONNAISE DES EAUX	ABELLO	Nolwenn	19000	BRIVE
ELECTRO DASSAULT	AFFAGARD	Taina	06000	NICE
MER DE BRETAGNE	ALBA	Olivia	25000	BESANCON
CESITE IMPRIMERIE	AMADOU	Warda	21000	DIJON
POSTE BILLE	AMBROISE	Denis	87000	LIMOGES
LAURENT BOUILLET ENTRE	AREVIAN	Liliane	76000	ROUEN
STRATINORES	ARRONDEAU	Sylvain	63000	CLERMONT FERRAND
CONSULT INF EAU	ASTORG	Frederic	33000	BORDEAUX
SOFT CONCEPT	AUSSAGE	Pauline	69000	LYON
INTERNETORAMA	AUTHELAIN	Precilia	82000	MONTAUBAN
ATELIER DAVID THOUVENO	BAUDELAIRE	Odile	51000	REIMS

Les données affichées non liées au fichier de données ne sont pas conservées lors des rafraîchissements d'affichage du champ Table (en cas de scrolling par exemple).

Il est possible de limiter les enregistrements affichés dans le champ en utilisant un filtre (fonction ***HFiltre*** utilisée dans le code d'initialisation du champ).

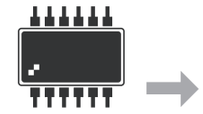
Plusieurs fonctions du WLangage permettent de manipuler les champs de type "Table fichier". Ces fonctions commencent toutes par "Table".

Pour ajouter un enregistrement dans le champ Table, ajoutez l'enregistrement dans le fichier de données HFSQL correspondant (fonction ***HAjoute***), puis ré-affichez le champ Table avec la fonction ***TableAffiche***.

Pour supprimer un enregistrement dans le champ Table, supprimez l'enregistrement dans le fichier de données HFSQL correspondant (fonction ***HSupprime***), puis ré-affichez le champ Table avec la fonction ***TableAffiche***.

Champ Table "Mémoire"

Un champ Table mémoire permet d'afficher directement des données chargées en mémoire. Les données sont ajoutées dans le champ Table par programmation (par exemple avec la fonction ***TableAjouteLigne***).

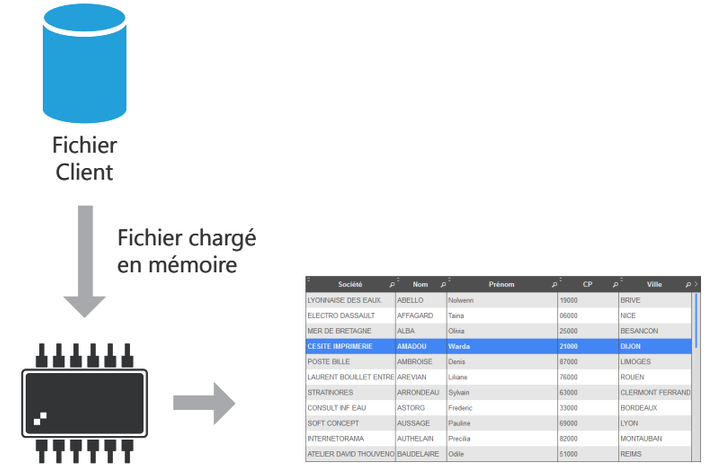


Société	Nom	Prénom	CP	Ville
LYONNAISE DES EAUX	ABELLO	Nolwenn	19000	BRIVE
ELECTRO DASSAULT	AFFAGARD	Taina	06000	NICE
MER DE BRETAGNE	ALBA	Olivia	25000	BESANCON
CESITE IMPRIMERIE	AMADOU	Warda	21000	DIJON
POSTE BILLE	AMBROISE	Denis	87000	LIMOGES
LAURENT BOUILLET ENTRE	AREVIAN	Liliane	76000	ROUEN
STRATINORES	ARRONDEAU	Sylvain	63000	CLERMONT FERRAND
CONSULT INF EAU	ASTORG	Frederic	33000	BORDEAUX
SOFT CONCEPT	AUSSAGE	Pauline	69000	LYON
INTERNETORAMA	AUTHELAIN	Precilia	82000	MONTAUBAN
ATELIER DAVID THOUVENO	BAUDELAIRE	Odile	51000	REIMS

Les données étant présentes en mémoire, le champ Table permet toutes les manipulations sur les données (tri sur n'importe quelle colonne, recherche dans les colonnes, ...).

Champ Table "Fichier chargé en mémoire"

Les champs Table fichier chargé en mémoire combinent les avantages des champs Table fichier à accès direct et ceux des champs Table mémoire.
Le champ Table est lié au fichier de données, mais le contenu du fichier de données est totalement chargé en mémoire. Le tri et la recherche sont disponibles sur toutes les colonnes.
Les données non liées au fichier de données sont conservées lors de la manipulation de l'ascenseur (colonne de type Interrupteur par exemple).
Les enregistrements du fichier de données étant chargés en mémoire, ce type de champ est conseillé pour les fichiers de données de moins de 100 000 enregistrements (pour éviter une saturation mémoire).



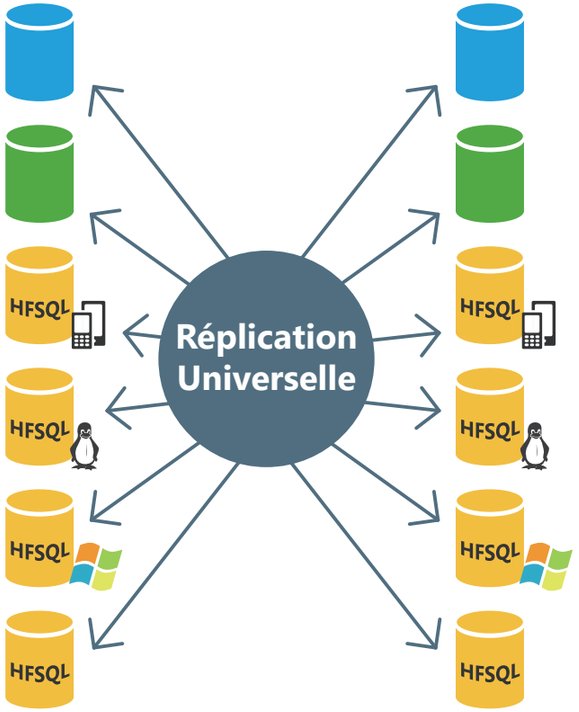
Société	Nom	Prénom	CP	Ville
LYONNAISE DES EAUX	ABELLO	Nolwenn	19000	BRIVE
ELECTRO DASSAULT	AFFAGARD	Taina	06000	NICE
MER DE BRETAGNE	ALBA	Olivia	25000	BESANCON
CESITE IMPRIMERIE	AMADOU	Warda	21000	DIJON
POSTE BILLE	AMBROISE	Denis	87000	LIMOGES
LAURENT BOUILLET ENTRE	AREVIAN	Liliane	76000	ROUEN
STRATINORES	ARRONDEAU	Sylvain	63000	CLERMONT FERRAND
CONSULT INF EAU	ASTORG	Frederic	33000	BORDEAUX
SOFT CONCEPT	AUSSAGE	Pauline	69000	LYON
INTERNETORAMA	AUTHELAIN	Precilia	82000	MONTAUBAN
ATELIER DAVID THOUVENO	BAUDELAIRE	Odile	51000	REIMS

Remarque : Les différents modes de remplissage (mémoire, fichier, fichier chargé en mémoire) sont disponibles pour les champs Liste, Combo, Table, Zone répétée, ...

Réplication universelle

La réplication universelle permet de maintenir à jour des bases de données de formats identiques ou différents (HFSQL, Oracle, SQL Server, ...). Il est par exemple possible de réaliser une synchronisation entre une base de données HFSQL Classic et une base de données Oracle.

La réplication universelle utilise un modèle centralisé : toutes les bases de données se synchronisent avec une base de données maître. La base de données maître répercute ensuite les modifications vers les autres bases de données.



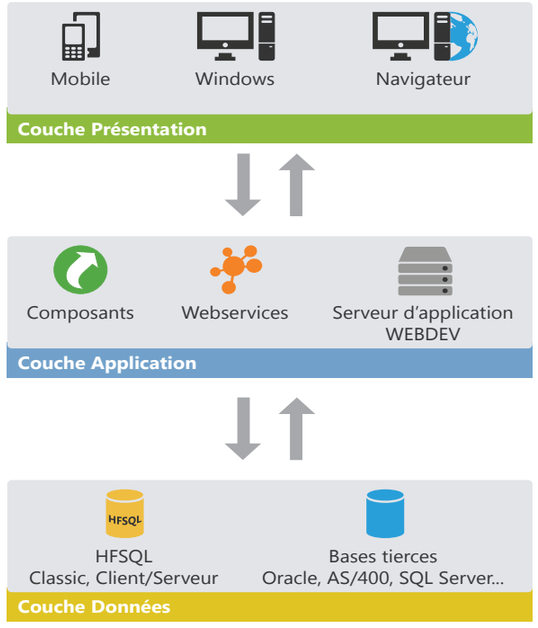
Il est possible d'adapter la synchronisation à des cas particuliers. Par exemple, il est possible de récupérer uniquement les enregistrements concernant un produit spécifié ou les enregistrements réalisés à une certaine date, de gérer les conflits, d'afficher une fenêtre de paramétrage, ...
Ces adaptations doivent être effectuées par programmation grâce à la fonction ***HRplProcédureFiltre***.

Architecture 3-tiers

L'architecture 3-tiers (le terme "tier" vient de l'anglais et signifie "niveau") est un modèle d'architecture d'application.
Son principe de base consiste à séparer trois couches logicielles contenues dans une application :

- la couche présentation : correspondant à l'affichage,
- la couche traitements ou application : correspondant aux processus métiers de l'application,
- la couche d'accès aux données persistantes.

Cette séparation a pour but de rendre indépendante chacune des couches afin de faciliter la maintenance et les évolutions futures de l'application. Elle assure une sécurité plus importante car l'accès à la base de données n'est autorisé que par la couche de traitements. Elle a également l'avantage d'optimiser le travail en équipe et le développement multi-cibles.



L'architecture 3-tiers est totalement compatible avec le développement d'applications en utilisant WINDEV.

DÉVELOPPEZ 10 FOIS PLUS VITE
PARTIE 4
Concepts avancés

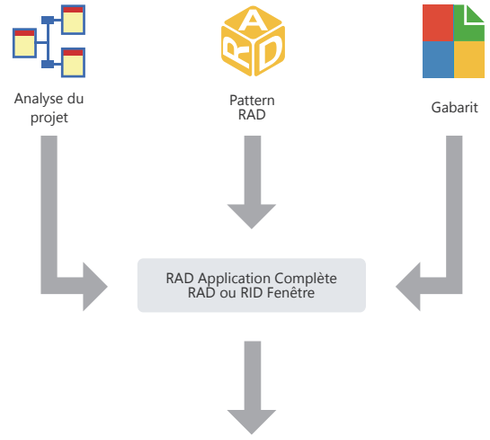
RAD
RID

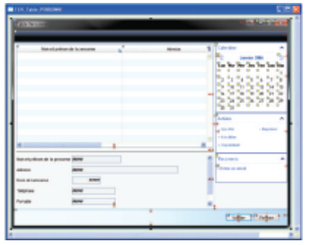
Le RAD (Rapid Application Development) et le RID (Rapid graphical Interface Design) permettent de créer des fenêtres à partir :

- de l'analyse liée au projet,
- des patterns RAD standard ou personnalisés,
- des gabarits.

En génération RAD, les fenêtres générées contiennent tout le code nécessaire à leur fonctionnement. Ces fenêtres peuvent être testées immédiatement, avec les données présentes sur le poste de développement.

En génération RID, les fenêtres générées contiennent uniquement les champs liés aux rubriques de l'analyse. Tout le code nécessaire au fonctionnement de ces fenêtres reste à la charge du développeur. Seul le code nécessaire au fonctionnement des éléments annexes du pattern est ajouté. Vous pouvez directement saisir votre code personnalisé.



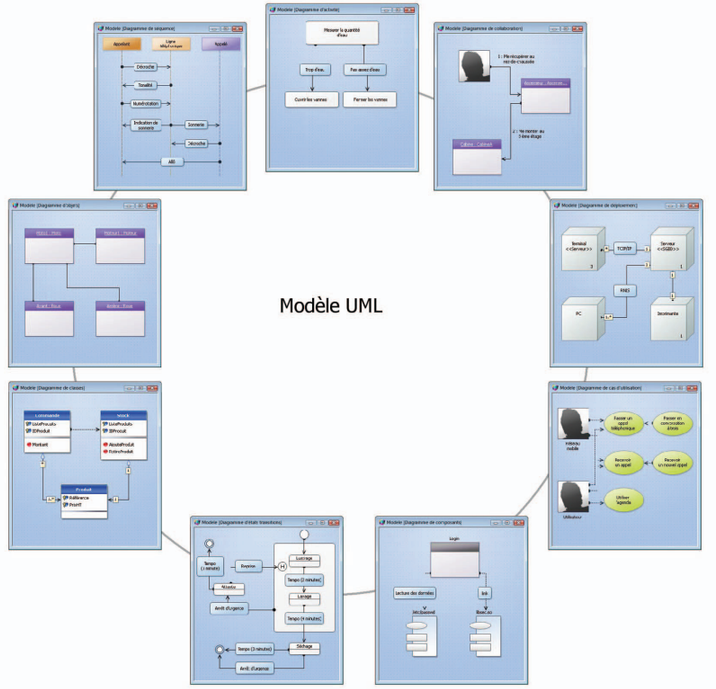


Exemple de fenêtre générée par le RAD ou le RID

Le modèle UML

WINDEV permet de créer neuf types de modèle UML :

- ***Diagramme de classes*** : décrit de manière générale la structure d'un système.
- ***Diagramme de cas d'utilisation*** : représente les fonctionnalités du système du point de vue de l'utilisateur.
- ***Diagramme d'objet*** : représente un ensemble d'objets et leurs relations à un moment donné.
- ***Diagramme de composants*** : décrit l'architecture physique et statique d'une application informatique.
- ***Diagramme d'activité*** : représente le comportement d'une méthode ou le déroulement d'un cas d'utilisation.
- ***Diagramme de séquences*** : représente l'ordre chronologique des messages envoyés et reçus par un ensemble d'objets.
- ***Diagramme de collaboration*** : présente l'organisation structurelle des objets qui envoient et reçoivent des messages.
- ***Diagramme d'états-transitions*** : présente un automate à états finis.
- ***Diagramme de déploiement*** : montre la répartition des matériels (les nœuds) utilisés dans un système et l'association des programmes exécutables à ces matériels.



Le modèle UML en pratique

1 Présentation

L'objectif principal d'une équipe de développement est de créer des applications optimisées, capables de satisfaire les besoins en constante évolution de leurs utilisateurs.

La modélisation d'une application permet de spécifier la structure et le comportement attendus d'un système, de comprendre son organisation, de déceler les possibilités de simplification et de réutilisation et de gérer les risques encourus.

Un modèle est la simplification de la réalité. Il permet de mieux comprendre le système à développer.

Un diagramme est la représentation graphique d'un ensemble d'éléments qui constituent un système. Pour visualiser un système sous différentes perspectives, le langage UML (Unified Modeling Language) propose neuf diagrammes, représentant chacun un état du système.

WINDEV permet de créer ces neuf types de modèle UML :

- Diagramme de classes
- Diagramme de cas d'utilisation
- Diagramme d'objet
- Diagramme de composants
- Diagramme d'activité
- Diagramme de séquences
- Diagramme de collaboration
- Diagramme d'états-transitions
- Diagramme de déploiement

Ce chapitre ne fait que présenter de manière globale la modélisation UML. Pour plus de détails sur le langage UML, nous vous conseillons de consulter des ouvrages spécifiques.

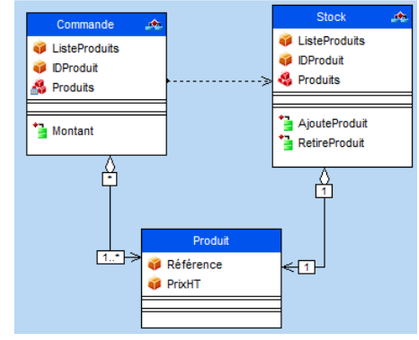
2 Les différents diagrammes UML

2.1 Diagramme de classes

Un diagramme de classes permet de modéliser la structure d'un système grâce à des classes et à des relations entre ces classes.

Les diagrammes de classes sont les diagrammes les plus courants dans la modélisation des systèmes orientés objet.

Par exemple, le diagramme suivant présente la gestion des stocks :

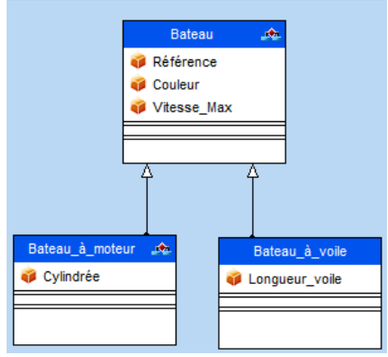


Un diagramme de classes est composé des éléments suivants :

- ***Classe*** : représente les structures de l'application. Chaque classe est divisée en trois compartiments :
  - *le nom de la classe* indique ce que la classe est et non ce qu'elle fait.
  - *les attributs de la classe* donnent les caractéristiques de la classe.
  - *les opérations de la classe* représentent les actions possibles sur la classe.

  Par exemple, la classe Stock contient l'attribut *ListeProduits*. Cette classe regroupe aussi les opérations *AjouteProduit* et *RetireProduit*. Ces opérations sont applicables aux instances de la classe.

  Remarque : Le langage UML définit trois niveaux de visibilité pour les attributs et les opérations :
  - ***Public*** : l'élément est visible par toutes les autres classes.
  - ***Protégé*** : l'élément est visible par la classe elle-même et par les sous-classes.
  - ***Privé*** : l'élément est visible uniquement par la classe.
- ***Relation*** : décrit le comportement des classes entre elles. Trois types de relations existent :
  - *Association* : Relation structurelle entre classes. Par exemple, la classe *Commande* est liée aux classes *Produit* et *Client*. Un Client peut passer plusieurs Commandes. Une commande est constituée de plusieurs produits. Une commande doit obligatoirement contenir au moins un produit.
  - *Dépendance* : Relation d'utilisation qui établit que les instances d'une classe sont reliées aux instances d'un autre élément. Par exemple, la classe *Commande* utilise la classe *Stock* : avant d'ajouter un produit dans une commande, il est nécessaire de vérifier que ce produit est en stock.
  - *Généralisation* : Relation entre une classe générale (parent) et une classe spécifique (enfant) qui en dérive. Par exemple, les classes *Bateau à voile* et *Bateau à moteur* sont dérivées de la classe *Bateau*.



- ***Paquetages*** : permettant de diviser et d'organiser la représentation du diagramme (de la même manière que les répertoires organisent les fichiers). Chaque paquetage peut contenir des classes et des relations.

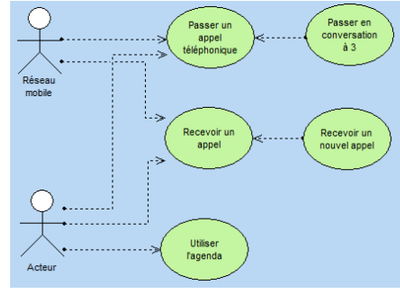
Grâce à la génération d'un diagramme de classes, il est possible de créer la structure des classes WINDEV utilisées dans votre application.

2.2 Diagramme de cas d'utilisation

Un diagramme de cas d'utilisation permet de visualiser le comportement d'un système de telle sorte que :

- l'utilisateur puisse comprendre comment utiliser chaque élément.
- le développeur puisse implémenter ces éléments.

Par exemple, il est possible de décrire le comportement d'un téléphone portable à l'aide d'un diagramme de cas d'utilisation.

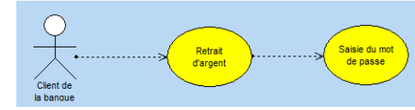


Un diagramme de cas d'utilisation est composé des éléments suivants :

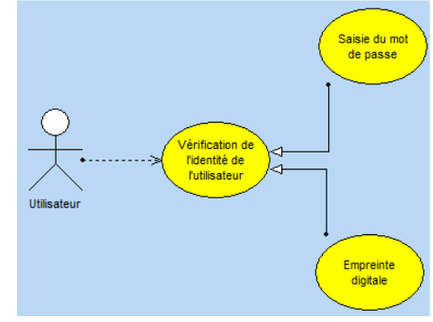
- ***Acteur*** : représente le rôle des utilisateurs de l'application. Par exemple, une personne qui travaille dans une banque sera le *Gestionnaire de prêts*. Si cette personne à un compte dans cette banque, elle jouera aussi le rôle de *Client*.
- ***Cas d'utilisation*** : décrit une séquence d'actions exécutées par l'application. Par exemple, *Passer une commande*, *Saisir une facture*, *Créer une nouvelle fiche client*, ...

  Un cas d'utilisation décrit ce que fait une application mais ne précise pas comment l'application le fait.
- ***Relation*** : décrit le comportement des acteurs par rapport aux cas d'utilisation. Trois types de relations existent :
  - *Association* : Relation structurelle entre deux éléments reliés.
  - *Dépendance* : Relation qui établit qu'un élément en utilise un autre. Par exemple, le *Client* d'une banque peut retirer de *l'argent* dans un distributeur automatique. Dans ce cas, l'action *Retrait d'argent* dépend du *Client*.

    Pour pouvoir retirer de l'argent, le Client doit préalablement saisir son code confidentiel. Dans ce cas, l'action *Retrait d'argent* dépend de la *Saisie du mot de passe*.



  - *Généralisation* : Relation permettant d'organiser les éléments selon une hiérarchie. Par exemple :
    - l'acteur *Client* peut être de deux types : *Client individuel* ou *Client d'entreprise*.
    - la vérification de l'identité d'un utilisateur peut être effectuée de deux façons : saisie du mot de passe ou vérification de l'empreinte digitale.



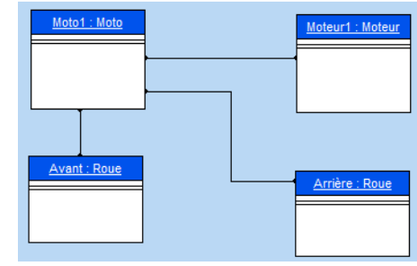
- ***Paquetage*** : divise et organise la représentation du diagramme (de la même manière que les répertoires organisent les fichiers). Chaque paquetage peut contenir des acteurs et des cas d'utilisation.

2.3 Diagramme d'objets

Un diagramme d'objets représente un ensemble d'objets et leurs relations à un moment donné.

Un diagramme d'objets est utilisé pour montrer un contexte (avant ou après une interaction entre objets par exemple).

Par exemple, le diagramme suivant présente une partie de la structure générale des motos :



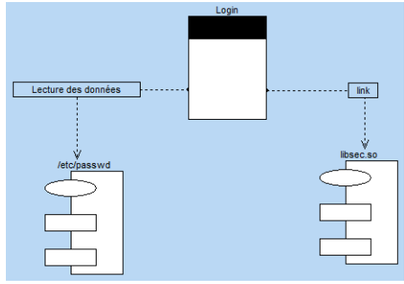
Un diagramme d'objets est composé des éléments suivants :

- ***objet*** : représente une instance d'une classe. Remarque : Si un diagramme de classes est ouvert, il est possible de créer un objet à partir d'une classe présente dans ce diagramme (drag and drop à partir du volet "Analyse UML").
- ***objet composite*** : représente de manière visuelle un objet formé d'autres objets. Par exemple : une fenêtre qui contient des ascenseurs, des boutons, ...
- ***lien*** : représente les relations entre les différents objets.

2.4 Diagramme de composants

Un diagramme de composants décrit l'architecture physique et statique d'une application informatique. Par exemple : fichiers sources, librairies, exécutables, ...

Par exemple, le diagramme suivant présente le fonctionnement d'un programme permettant de se loguer en mode texte sous Unix.



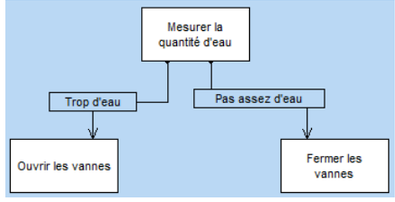
Un diagramme de composants est composé des éléments suivants :

- ***module*** : représente les différents éléments physiques constituant une application informatique. Par exemple : un fichier, une librairie, ... Un module peut être représenté :
  - soit par une ***spécification*** qui montre l'interface du module. Cette spécification peut être ***générique*** dans le cas de classes paramétrables.
  - soit par son ***corps*** qui présente l'implémentation du module.
- ***tâche*** : représente un composant ayant son propre flot (thread) de contrôle.
- ***programmes principaux*** de l'application informatique.
- ***sous-programmes*** : regroupent les procédures et les fonctions qui n'appartiennent pas à des classes.

2.5 Diagramme d'activité

Un diagramme d'activité représente le comportement d'une méthode ou le déroulement d'un cas d'utilisation.

Par exemple, le diagramme suivant présente le déroulement d'un barrage :



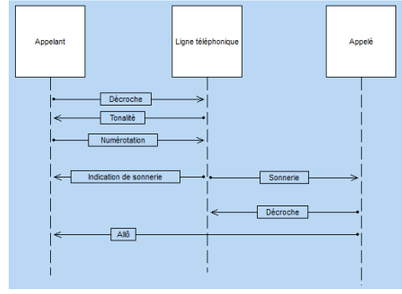
Un diagramme d'activité est composé des éléments suivants :

- ***activité*** : représente une étape particulière dans l'exécution d'un mécanisme. Par exemple : "Etablir un devis", "Ouvrir la fenêtre", ...
- ***barre de synchronisation*** : permet de synchroniser les différentes activités :
  - soit en indiquant les activités à effectuer avant une certaine activité. Par exemple : "Appuyer sur l'embrayage" et "Enclencher une vitesse" avant de "Relâcher l'embrayage".
  - soit en indiquant les activités à effectuer en parallèle.
- ***objet*** : permet de rattacher des activités à l'objet qui réalise ces activités. Par exemple, les activités "Commander" et "Payer" sont rattachées à l'objet "Client"; les activités "Enseigner", "Contrôler les connaissances" sont rattachées à l'objet "Enseignant".
- ***émission de signal*** : représente l'envoi d'un signal vers un objet.
- ***attente de signal*** : représente l'attente d'un signal en provenance d'un objet.
- ***transition*** : représente le passage d'une activité terminée à une autre. Par exemple : "Trop d'eau", "Assez d'argent", ...

2.6 Diagramme de séquence

Un diagramme de séquence représente l'ordre chronologique des messages envoyés et reçus par un ensemble d'objets.

Par exemple, le diagramme suivant représente le début d'une communication téléphonique :



Un diagramme de séquence est composé des éléments suivants :

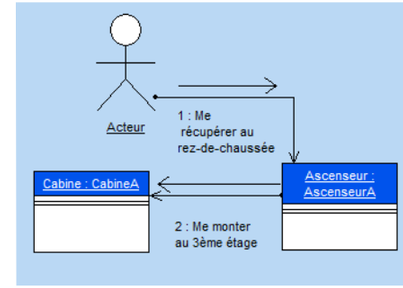
- ***objet*** : représente les différents objets utilisés. Chaque objet est représenté par un carré surmontant une ligne en pointillé. Cette ligne représente la durée de vie de l'objet. Par exemple : "Appelant", "Appelé", ...
- ***période d'activation d'un objet*** : sur la ligne de vie d'un objet, il est possible d'insérer des périodes d'activation de l'objet. Ces périodes représentent les moments où l'objet est actif.

- ***message*** : représente, grâce à des flèches horizontales, les messages échangés entre les différents objets. Ces flèches sont orientées de l'émetteur du message vers le destinataire. L'ordre d'envoi des messages est donné par la position des flèches sur l'axe vertical. Par exemple : "Décroche", "Sonnerie", ...

2.7 Diagramme de collaboration

Un diagramme de collaboration présente l'organisation structurelle des objets qui envoient et reçoivent des messages.

Par exemple, le diagramme suivant montre l'utilisation d'un ascenseur par une personne :



Un diagramme de collaboration est composé des éléments suivants :

- ***objet*** : représente les différents objets utilisés.
- ***acteur*** : représente un élément externe du système. Par exemple une personne.
- ***message*** : représente les messages échangés entre les différents objets.

2.8 Diagramme d'états-transitions

Un diagramme d'états-transitions présente un automate à états finis. Il permet ainsi de décrire les changements d'états d'un objet ou d'un composant.

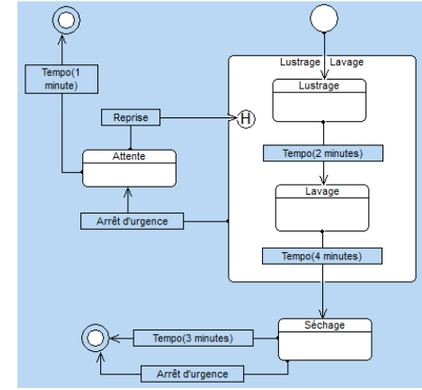
Un état se caractérise par sa durée et sa stabilité.

Une transition représente le passage instantané d'un état vers un autre.

Une transition est déclenchée :

- soit par un événement.
- soit automatiquement lorsqu'aucun événement déclencheur n'est spécifié.

Par exemple, le diagramme suivant présente les différentes étapes d'une machine à laver les voitures :



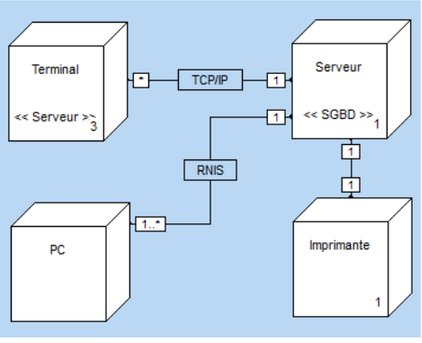
Un diagramme d'états-transitions est composé des éléments suivants :

- ***état*** : représente la valeur des attributs d'un objet à un instant donné.
- ***état initial*** : représente l'état au démarrage du système.
- ***état final*** : représente l'état dans lequel se trouve le système à la fin du fonctionnement.
- ***super-état*** : permet de structurer le diagramme en indiquant plusieurs niveaux de distinction entre les états.
- ***historique*** : représente le dernier état actif d'un super-état.
- ***souche*** : permet de symboliser les états contenus dans un super-état. Il est ainsi possible de relier ces états à d'autres états n'appartenant pas au super-état.
- ***transition*** : représente le passage d'un état à un autre.

2.9 Diagramme de déploiement

Un diagramme de déploiement montre la répartition physique des matériels (les nœuds) utilisés dans un système et l'association des programmes exécutables à ces matériels.

Par exemple, le diagramme suivant présente les différents matériels utilisés dans une entreprise :



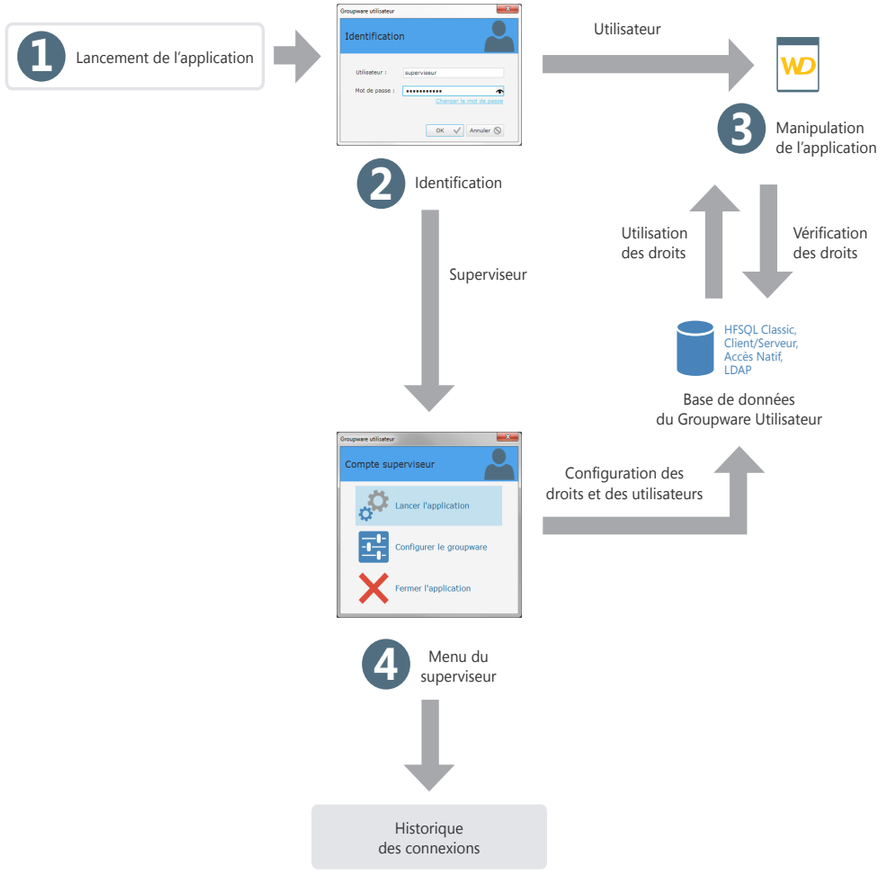
Un diagramme de déploiement est composé des éléments suivants :

- ***classe de nœuds*** : représente une classe de ressource matérielle. Par exemple : un serveur, un PC, une imprimante, ...
- ***instance d'un nœud*** : représente une ressource matérielle. Par exemple : le serveur numéro 3, l'imprimante numéro 7, ...
- ***connexion*** : décrit le support de communication entre deux nœuds. Par exemple : liaison RNIS ou TCP/IP.

Groupware utilisateur

Le Groupware Utilisateur permet d'inclure simplement une gestion des utilisateurs et de leurs droits dans une application WINDEV.
Deux types d'utilisateurs sont pris en compte :

- utilisateur simple, utilisant directement l'application.
- superviseur, pouvant configurer les utilisateurs et leurs droits.



Le groupware utilisateur en pratique

1 Présentation

Une application nécessite la définition du rôle des différents intervenants. Il est souvent nécessaire de prévoir différents niveaux d'accès selon les fonctions de l'utilisateur.

En effet, tous les utilisateurs n'ont pas les mêmes fonctions, ni les mêmes besoins. Leur domaine d'action doit donc être plus ou moins limité.

Prenons un exemple simple : lors de la mise en place d'une gestion commerciale, l'application propose les fonctionnalités suivantes :

- Consultation du tarif,
- Modification du tarif,
- Saisie des commandes,
- Saisie des clients.

Selon l'utilisateur les accès possibles sont différents. Quelques exemples :

- les secrétaires peuvent consulter le tarif et créer des commandes
- les commerciaux peuvent consulter le tarif et établir les commandes, gérer de nouveaux clients.
- les directeurs commerciaux ont accès à toutes les options.

Pour gérer ces niveaux d'accès dans vos applications WINDEV, il suffit d'intégrer le groupware utilisateur à votre application. Quelques clics de souris suffisent à transformer une application classique en une application gérant de nombreux niveaux d'accès. La mise en place est instantanée.

Lors de l'exécution de l'application, le responsable pourra créer des utilisateurs (identifiés par leur login et leur mot de passe) et leur donner accès à l'une ou l'autre des fonctionnalités de l'application.

2 Comment fonctionne le groupware utilisateur ?

Une application utilisant le groupware utilisateur a deux niveaux d'utilisation :

- le niveau utilisateur
- le niveau superviseur

2.1 Le niveau utilisateur

L'utilisateur se connecte à l'application grâce à une fenêtre de login et accède aux fonctionnalités qui lui sont autorisées.

Remarque : Le groupware utilisateur peut permettre de se connecter en utilisant un annuaire LDAP.

2.2 Le niveau superviseur

Le superviseur se connecte à l'application par une fenêtre de login et accède à un menu avancé lui permettant de configurer les utilisateurs et leurs droits, ou bien de lancer l'application.

La fenêtre de configuration permet au superviseur de :

- lancer l'application.
- configurer le groupware.

La configuration du groupware consiste à créer des utilisateurs, des groupes et d'associer des utilisateurs aux groupes, à gérer les droits de chaque utilisateur (ou groupe) sur chacune des fenêtres de l'application. Les droits sont définissables au niveau des options de menu, des groupes de champs et des champs. Le superviseur peut rendre ces éléments grisés, inactifs ou invisibles. Ces configurations sont prioritaires sur toute configuration effectuée dans le programme.

Pour plus de détails sur la configuration du groupware par le superviseur consultez le paragraphe "Paramétrer l'application utilisant le groupware utilisateur", page 134.

Remarque : Toutes les informations concernant les utilisateurs et leurs droits sont stockées dans des fichiers de données au format HFSQL (Classic ou Client/Serveur).

3 Mise en place du groupware utilisateur

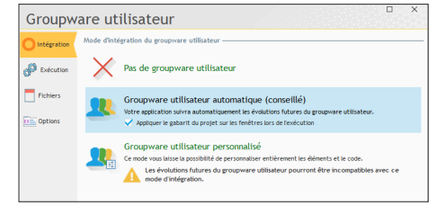
3.1 Ajouter le groupware utilisateur dans une application

Pour mettre en place le groupware utilisateur dans une application WINDEV, une seule option suffit : sous le volet "Projet", dans le groupe "Projet", cliquez sur "Groupware Utilisateur". La fenêtre de paramétrage du groupware utilisateur permet de définir :

- Le mode d'intégration du groupware utilisateur.
- Le type de lancement du groupware utilisateur.
- L'emplacement des fichiers de données du groupware utilisateur.
- Le mode d'identification de l'utilisateur.

Mode d'intégration du Groupware Utilisateur

Plusieurs modes d'intégration du groupware utilisateur sont disponibles :



- **Groupware utilisateur automatique** : toute la programmation du groupware utilisateur est automatiquement intégrée à votre application. Aucune personnalisation n'est possible. Le gabarit du projet est automatiquement appliqué à toutes les fenêtres utilisées par le groupware utilisateur. Ce mode d'intégration permet de suivre automatiquement les éventuelles évolutions du groupware utilisateur.
- **Groupware utilisateur personnalisé** : toute la programmation du groupware utilisateur est automatiquement intégrée à votre application via un composant interne. ***Toutes les fenêtres nécessaires à la gestion du groupware utilisateur (login et gestion des utilisateurs) sont automatiquement ajoutées à votre projet***. Vous pouvez ainsi personnaliser totalement toutes les fenêtres utilisées par le groupware utilisateur.

Le détail des différentes fenêtres du Groupware Utilisateur est présenté dans l'aide en ligne.

Lancement du Groupware Utilisateur

Deux modes de lancement du groupware utilisateur sont disponibles :

- ***Lancement automatique*** : La fenêtre de login du groupware utilisateur est lancée avant toute autre fenêtre de l'application. Le code d'initialisation du projet est exécuté après l'ouverture et la validation de la fenêtre de login.
- ***Lancement manuel*** : La fenêtre de login ne sera ouverte que si la fonction ***gpwOuvre*** est utilisée. Cette option permet d'exécuter le code d'initialisation du projet par exemple, avant d'ouvrir la fenêtre de login. Il est ainsi possible par exemple d'afficher une fenêtre demandant la langue d'exécution de l'application.

Login automatique en mode test

Vous pouvez indiquer le login et le mot de passe à utiliser en mode "Test automatique". Ces informations seront utilisées si un test automatique est réalisé sur l'application.

Fichiers de données du Groupware Utilisateur

Le groupware utilisateur utilise plusieurs fichiers de données spécifiques. Ces fichiers de données permettent notamment de gérer les utilisateurs, les groupes, les droits et les éléments de l'application.

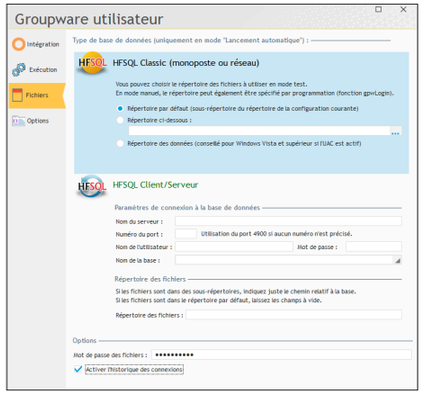
Le groupware utilisateur peut être utilisé :

- soit avec des fichiers de données HFSQL Classic. Dans ce cas, l'onglet "Fichiers" permet de spécifier si nécessaire l'emplacement des fichiers de données.
- soit avec des fichiers de données HFSQL Client/Serveur. Dans ce cas, l'onglet "Fichiers" permet de définir les paramètres d'accès à la base de données HFSQL Client/Serveur, ainsi que le répertoire des fichiers.

Le mot de passe des fichiers de données

Par défaut, les fichiers de données du groupware utilisateur ont un mot de passe spécifique. Ce mot de passe est : "PCSGPW2001".

Pour changer ce mot de passe, saisissez le nouveau mot de passe dans l'onglet "Fichiers" de la fenêtre de description du groupware utilisateur. Les caractères saisis apparaissent sous forme d'étoiles.



Remarque : Ce mot de passe est utilisé par exemple lors de la ré-indexation ou de l'ouverture des fichiers de données avec l'outil WDMAP.

Pour plus de détails sur ces fichiers de données et leur configuration (répertoire d'installation, ...), consultez l'aide en ligne.

Identification

Le groupware utilisateur propose plusieurs types d'identification des utilisateurs :

- Identification gérée par le groupware utilisateur de WINDEV.
- Identification à partir d'un annuaire LDAP.
- Identification de Windows.

Identification gérée par le groupware utilisateur

Les login et mot de passe sont enregistrés dans les fichiers de données du groupware utilisateur.

Identification à partir d'un annuaire LDAP

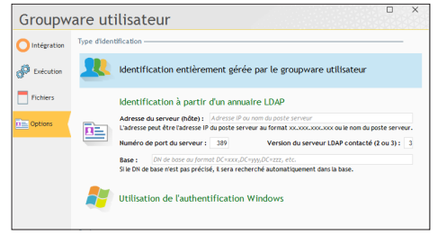
Si votre client utilise un annuaire LDAP au sein de son entreprise, il est possible de brancher le groupware utilisateur sur cet annuaire. Il suffit de préciser dans l'onglet "Identification" les caractéristiques de l'annuaire LDAP (serveur, port, base, ...). Ainsi le compte et le mot de passe présent dans l'annuaire LDAP seront automatiquement demandés à l'utilisateur.

Deux modes sont disponibles :

- **Sans gestion des droits** : Dans ce cas, aucun fichier de données du groupware ne sera créé. Si le lancement est automatique, la fenêtre de login demandera à l'utilisateur de s'identifier. Si l'utilisateur est bien enregistré dans l'annuaire LDAP, l'application se lancera, sinon elle se fermera. Il n'est pas possible à un superviseur de paramétrer les droits sur les fenêtres.
- **Avec gestion des droits** : Seuls les utilisateurs présents dans l'annuaire LDAP pourront se connecter. Le paramétrage des droits sur les fenêtres est disponible.

Pour plus de détails, consultez l'aide en ligne.

Remarque : Le groupware utilisateur fonctionne avec Active Directory. Il ne fonctionne pas avec openLDAP.



Authentification Windows

Dans ce cas, les login et mot de passe correspondent à ceux définis pour utiliser Windows.

Remarque : il est possible d'utiliser une authentification unique. Dans ce cas, le login et le mot de passe seront demandés une seule fois par session.

3.2 Tester l'application

Lors du test d'une application gérant le groupware utilisateur, la première fenêtre apparaissant est automatiquement la fenêtre de login (quelle que soit la première fenêtre définie dans votre application).

Par défaut, un seul utilisateur existe : le superviseur. Pour se connecter en tant que superviseur, il suffit de saisir les informations suivantes dans la fenêtre de login :

- Nom : SUPERVISEUR
- Mot de passe : SUPERVISEUR

Vous pouvez alors soit tester votre application, soit configurer le fonctionnement du groupware.

Remarques :

- Pour que la première fenêtre de votre application ne soit pas la fenêtre de login, sélectionnez l'option "Lancement manuel" dans l'onglet "Exécution" des options du Groupware Utilisateur. Il suffira d'utiliser la fonction du WLangage ***gpwOuvre*** pour ouvrir la fenêtre de login.
- Les fenêtres de gestion du groupware utilisateur sont exécutées avant le code d'initialisation du projet.
- La première fenêtre de votre application définie dans le projet sera lancée après la fenêtre de login (lorsque l'utilisateur n'est pas le superviseur).
- **Pour ne pas lancer le groupware utilisateur lors des tests de l'application**, il suffit de sélectionner l'option "Pas de groupware utilisateur" dans l'onglet "Intégration" des options du groupware utilisateur.
  Si vous rebranchez le groupware utilisateur par la suite, les fichiers de données utilisés précédemment pour le groupware utilisateur ne seront pas effacés.

3.3 Créer l'exécutable WINDEV

Lors de la création de l'exécutable gérant le groupware utilisateur (option "Générer" sous le volet "Accueil"), un écran spécifique au groupware utilisateur est affiché.

Vous pouvez ainsi définir l'emplacement des fichiers de données du groupware utilisateur.

3.4 Installer une application utilisant le groupware utilisateur

L'installation d'une application utilisant le groupware utilisateur se fait comme pour toute application : sous le volet "Projet", il suffit de cliquer sur "Procédure d'installation".

Un écran spécifique au groupware utilisateur permet de configurer l'emplacement des fichiers de données propres au groupware utilisateur.

Remarques :

- Si vous avez paramétré les fichiers de données du groupware utilisateur pour votre client, il est nécessaire de les sélectionner lors de la préparation de l'installation. Il est donc nécessaire de personnaliser la liste des fichiers de l'installation.
- Si vous n'avez pas paramétré les fichiers de données du groupware, lors du premier lancement de l'application, seul l'utilisateur Superviseur existera dans l'application.

Conseil : Si vous ne configurez pas les différents niveaux d'utilisation du groupware utilisateur pour vos clients, il est conseillé de leur fournir un document reprenant tous les champs de vos fenêtres pour obtenir une configuration optimisée.

4 Paramétrer l'application utilisant le groupware utilisateur

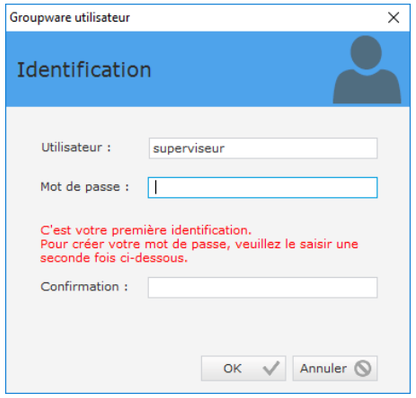
Lorsque l'application est déployée, le paramétrage du groupware utilisateur est réalisé par le superviseur de l'application.

Ce paramétrage consiste à créer des utilisateurs, des groupes d'utilisateurs et à leur attribuer pour chaque champ de chaque fenêtre de l'application certains droits. Ces droits consistent à rendre visible, invisible ou grisé des champs, des groupes de champs, des options de menus.

Pour paramétrer le groupware utilisateur, il suffit de :

1. Lancer l'application (exécutable ou test depuis WINDEV) et se connecter en tant que superviseur :

- Nom : SUPERVISEUR
- Mot de passe : SUPERVISEUR



2. Choisir l'option "Configurer le groupware".

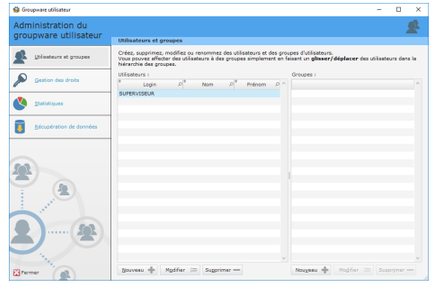
3. La fenêtre de gestion du groupware utilisateur s'affiche. Cette fenêtre permet de :

- gérer les utilisateurs et les groupes
- gérer les droits
- afficher les statistiques
- migrer les données provenant d'un groupware utilisateur version 17 et antérieur.

4.1 Gérer les utilisateurs

La gestion des utilisateurs consiste à :

- créer des utilisateurs,
- créer des groupes,
- associer des utilisateurs à des groupes.



La création d'un nouvel utilisateur consiste à donner les informations suivantes :

- le nom de l'utilisateur.
- le prénom de l'utilisateur.
- le login de l'utilisateur (obligatoire). Ce login correspond à l'identifiant de l'utilisateur lors de sa connexion à l'application.
- le mot de passe de l'utilisateur. Ce mot de passe n'est pas obligatoire et peut être saisi par l'utilisateur lui-même lors de sa première connexion (cochez l'option correspondante).

Il est possible de définir l'utilisateur comme étant un superviseur de l'application.

Il est bien entendu possible de modifier ou de supprimer un utilisateur. La suppression d'un utilisateur permet soit de supprimer entièrement l'utilisateur, soit de supprimer uniquement ses droits.

Les utilisateurs peuvent être rassemblés en groupes.

Pour associer un utilisateur à un groupe, il suffit de réaliser un "Drag and Drop" de l'utilisateur vers le groupe choisi.

Un utilisateur peut être associé à plusieurs groupes.

Remarques :

- Il est conseillé de changer le mot de passe du superviseur dès la première utilisation.
- Si vous utilisez un annuaire LDAP, vous avez la possibilité d'importer les utilisateurs présents dans l'annuaire pour pouvoir gérer les droits de ces utilisateurs.

4.2 Gérer les droits

Pour chaque utilisateur (ou groupe d'utilisateurs), le superviseur peut gérer leurs droits sur les différents éléments des fenêtres du projet.

Pour chaque association utilisateur / fenêtre, il est possible de définir un état spécifique pour tous les éléments de la fenêtre.

Les éléments gérés dans les fenêtres, les fenêtres internes et les modèles de fenêtres sont les suivants :

- les champs
- les groupes de champs
- les options de menu

Les états disponibles pour chaque élément sont les suivants :

- ***Défaut*** : le comportement de l'élément correspond au comportement par défaut, défini dans l'application.
- ***Inactif*** : l'élément est affiché mais toute saisie est impossible.
- ***Grisé*** : l'élément est affiché mais grisé. Toute saisie est impossible.
- ***Invisible*** : l'élément n'est pas affiché.

4.3 Afficher les statistiques

Pour chaque application, il est possible d'obtenir des statistiques d'utilisation selon l'utilisateur, pour une période spécifique.

4.4 Migrer des données

A partir de la version 18, le groupware utilisateur a évolué. Les fichiers de données utilisés par le groupware utilisateur ne sont plus les mêmes.

L'administrateur du groupware utilisateur peut migrer les données existantes au nouveau format et ainsi récupérer automatiquement les données utilisées dans des versions précédentes. Cette migration peut être effectuée quel que soit le format des données : HFSQL Classic, HFSQL Client/Serveur, Accès Natif.

5 Conseils pour une application gérant le groupware utilisateur

5.1 Utilisation de groupes de champs

Pour simplifier le paramétrage de la gestion du groupware utilisateur en fonction des utilisateurs, il est conseillé de regrouper les champs par groupe de champs.

Il est par exemple possible de créer dans vos fenêtres des groupes de champs correspondant aux champs à afficher pour un type d'utilisateur.

La possibilité d'associer un champ à plusieurs groupes de champs augmente les combinaisons disponibles.

Ces groupes de champs peuvent être créés dans votre application uniquement en vue d'une gestion du groupware, sans programmation spécifique de votre part.

5.2 Visibilité des champs

Lors du développement de votre application, vous pouvez définir les caractéristiques de visibilité des différents éléments de vos fenêtres :

- soit lors de la description de l'élément (fenêtre à 7 onglets)
- soit par programmation (propriétés ***..Etat*** ou ***..Visible***)

Lors de la configuration du groupware utilisateur, le superviseur peut définir d'autres caractéristiques de visibilité. Les caractéristiques définies par le superviseur sont prioritaires.

Par exemple, un bouton permet de rendre un champ actif. Or ce champ a été grisé par le superviseur. Votre code ne sera pas pris en compte et le champ ne sera pas actif.

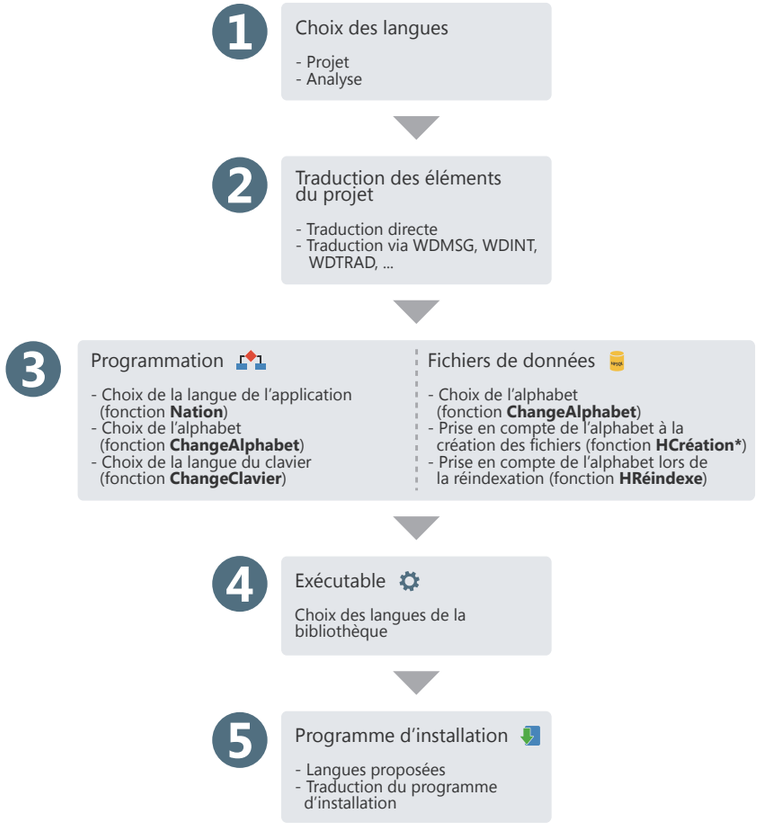
5.3 Définition des droits

Pour obtenir une définition des droits correspondant aux fonctionnalités de votre application, il est conseillé :

- soit d'effectuer la configuration des droits voulue par l'utilisateur de votre application avant de créer la procédure d'installation. Il suffit alors d'ajouter les fichiers de données du groupware lors de la création de l'installation de l'application.
- soit de livrer un dossier de programmation permettant de donner les noms des champs, des groupes de champs et des options à gérer selon le niveau d'utilisation de l'application.

Applications multilingues

Une application multilingue est une application qui pourra être diffusée dans plusieurs langues. WINDEV prend en compte les différentes langues de l'application à toutes les étapes du développement d'une application.
Pour développer une application multilingue, les principales étapes sont les suivantes :



Applications multilingues en pratique

1 Présentation

Une application multilingue pourra être diffusée dans plusieurs langues. Les différentes langues de l'application seront prises en compte lors des différentes étapes du développement.

Pour développer une application multilingue, les étapes sont les suivantes :
1. Choix des langues gérées par le projet et l'analyse.
2. Saisie des différents éléments du projet (fenêtre, code, ...) dans les différentes langues du projet.
3. Détermination de la langue du projet par programmation.
4. Gestion des alphabets spécifiques dans les fichiers de données.
5. Création de l'exécutable et du programme d'installation.

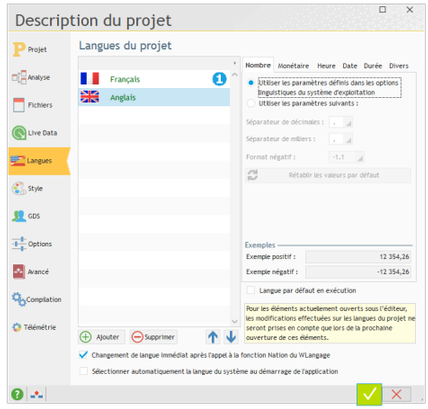
Remarques :

- Si le système d'exploitation du poste en cours gère plusieurs langues (Hébreux, Arabe, Grec, ...), lors de la saisie des traductions dans ces langues, l'alphabet correspondant sera automatiquement utilisé.
- Si votre application est multilingue, il est nécessaire de gérer cette fonctionnalité dans les fenêtres du groupware utilisateur et les fenêtres de gestion automatique des erreurs HFSQL. Pour plus de détails, consultez l'aide en ligne.
- La gestion de l'Unicode est disponible dans les fichiers de données HFSQL, les champs des fenêtres et le code WLangage.

2 Choix des langues gérées par le projet et l'analyse

2.1 Langues gérées par le projet

Les différentes langues gérées par le projet sont définies dans la description du projet :
1. Sous le volet "Projet", dans le groupe "Projet", cliquez sur "Description". La fenêtre de description du projet s'affiche.
2. Dans l'onglet "Langues", il est possible d'ajouter ou de supprimer les langues gérées par le projet.



Les différentes langues sélectionnées seront proposées pour toutes les informations des différents éléments ou objets pouvant être traduites (libellés des champs, options de menus, messages d'aide associés à un champ, ...).

Lors de la modification des différentes langues du projet, les modifications seront automatiquement prises en compte :

- pour tout nouvel élément ou objet créé sous l'éditeur de WINDEV,
- pour tout élément ou objet ouvert sous l'éditeur de WINDEV.

La langue principale correspond à la langue utilisée par défaut en exécution.

Options linguistiques

L'onglet "Langues" de la description du projet permet également de définir toutes les options spécifiques à une langue : nombre, monétaire, date, ...

Par défaut, les paramètres utilisés sont ceux définis dans les options linguistiques de Windows (accessibles depuis le panneau de configuration de Windows).

Lors de la création d'un champ de saisie ou d'une colonne de table affichant des données de type numérique (nombre, monétaire, date, heure, ...), le masque de saisie utilisé sera automatiquement le masque défini dans les options de langue du projet. Cette option est disponible dans les fenêtres.

En exécution, lorsqu'un champ de saisie ou une colonne de table a pour masque "Défini par le projet", le masque de saisie / d'affichage s'adaptera automatiquement selon les options choisies dans le projet pour la langue affichée en exécution.

Remarque : Les options linguistiques permettent également de définir le sens d'écriture et l'alphabet utilisé (option "divers").

Programme exécutable

Lors de la création du programme exécutable, il est possible de spécifier la ou les langues prises en compte dans l'exécutable. Il est ainsi possible de créer par exemple, un programme exécutable différent pour chaque langue gérée par le projet.

Cette fonctionnalité permet de diminuer la taille de l'exécutable.

2.2 Langues gérées par l'analyse

Si votre projet utilise une analyse, le choix des différentes langues gérées par l'analyse se fait directement sous l'éditeur d'analyses. En effet, une même analyse peut être partagée entre différents projets ne proposant pas tous les mêmes langues. Ainsi, le nombre de langues défini pour l'analyse peut être supérieur à celui défini pour le projet.

Pour configurer les différentes langues gérées par l'analyse :
1. Affichez l'analyse de votre projet sous l'éditeur.
2. Affichez la fenêtre de description de l'analyse : sous le volet "Analyse", dans le groupe "Elément courant", cliquez sur "Description".
3. Sélectionnez l'onglet "International" de la fenêtre de description de l'analyse.

Les différentes langues configurées dans l'analyse seront proposées :

- lors de la configuration des informations partagées des rubriques. Vous pouvez saisir la description des champs liés aux rubriques (options, libellés, ...) dans les différentes langues gérées par l'analyse. Lors de la génération d'un "RAD application complète" ou d'un RAD fenêtre, ces informations seront automatiquement prises en compte pour toutes les langues communes à l'analyse et au projet.
- pour les informations imprimées dans le dossier de l'analyse (notes du fichier de données ou de la rubrique).
- pour les informations gérées par "Etats et Requêtes". En effet, si le logiciel "Etats et Requêtes" est livré avec votre application WINDEV, il est possible de traduire le nom du fichier de données et le nom de chaque rubrique. Ainsi, l'utilisateur pourra utiliser les noms correspondant à sa langue lors de la création de ses états et requêtes. Ces informations peuvent être saisies dans l'onglet "Etats et Requêtes" de la description de fichier de données et de la description de rubriques.

2.3 Langues gérées par les différents éléments du projet

Par défaut, les différents éléments du projet (fenêtres, états, code, classes, collections de procédures, ...) gèrent les mêmes langues que le projet dans lequel ils ont été créés.

Il est possible qu'un élément gère plus de langues que le projet (par exemple si l'élément est partagé entre plusieurs projets gérant des langues différentes).

3 Saisie de l'interface dans les différentes langues

Lorsque les différentes langues gérées par l'application ont été sélectionnées, il est nécessaire de saisir toutes les informations affichées par l'application dans ces différentes langues.

Pour saisir l'interface dans différentes langues, plusieurs éléments doivent être pris en compte :

- Le mode de traduction choisi.
- Les langues gérées (langues utilisant un alphabet spécifique ou non)
- L'utilisation de fenêtres spécifiques (groupware utilisateur, gestion automatique des erreurs HFSQL, aperçu avant impression)
- Les messages affichés par programmation.

3.1 Mode de traduction choisi

WINDEV propose plusieurs modes de traduction :

- Traduction automatique des interfaces à l'aide d'un outil de traduction spécifique, présent sur le poste de développement. Cette traduction est réalisée directement sous l'éditeur de WINDEV.
- Export de tous les messages à traduire à l'aide de WDMSG et réintégration après traduction.

Traduction des interfaces

Pour chaque élément, différentes zones multilingues sont affichées dans les fenêtres de description de l'élément. Ces zones multilingues permettent de saisir les informations dans les différentes langues gérées par le projet.

Pour traduire ces informations depuis WINDEV :
1. Affichez les options de WINDEV pour configurer les options de traduction du logiciel utilisé : sous le volet "Accueil", dans le groupe "Environnement", déroulez "Options" et sélectionnez "Options générales de WINDEV".
2. Dans l'onglet "Traduction", il est possible de définir :

- l'outil de traduction par défaut (et les modalités de transmission du texte à traduire/traduit).
- les langues source et destination.

3. Si un outil de traduction est précisé, dans chaque élément contenant des informations à traduire, un bouton "Traduire" sera affiché. Ce bouton permettra de traduire le texte sélectionné en utilisant l'outil de traduction spécifié.
4. Pour effectuer la traduction, il suffit de sélectionner le texte à traduire et cliquer sur le bouton "Traduire".

Export et réintégration des informations à traduire

L'outil WDMSG, disponible séparément, permet d'extraire tous les messages d'un projet (libellé des champs, ...), puis de les réintégrer après traduction.

Contactez le service commercial de PC SOFT pour plus de détails sur les conditions d'utilisation de ce produit.

3.2 Langues utilisant un alphabet spécifique

Si votre application gère des langues utilisant des alphabets spécifiques (Grec, Russe, ...), il est nécessaire de saisir la traduction des différents messages multilingues en utilisant ces alphabets spécifiques.

WINDEV permet de gérer automatiquement l'utilisation des alphabets spécifiques sous l'éditeur.

En effet, dès que le caret (curseur de saisie de la souris) est positionné dans une zone de saisie d'une langue utilisant un alphabet spécifique, la langue d'entrée (alphabet utilisé par le clavier) correspondante est automatiquement activée.

Ainsi, si vous saisissez un libellé dans la zone Russe de la description du libellé, l'alphabet utilisé par le clavier sera automatiquement le russe.

Rappel : Pour utiliser des alphabets spécifiques, il est nécessaire d'installer les fichiers correspondants aux alphabets voulus dans les options régionales de Windows (panneau de configuration).

3.3 Traductions des fenêtres spécifiques

Gestion des erreurs HFSQL

Par défaut, les fenêtres de gestion des erreurs HFSQL sont fournies en anglais et en français. Pour les traduire dans une autre langue, il est nécessaire de :

1. Intégrer les fenêtres d'erreur par défaut dans votre projet. Les fenêtres sont fournies (avec leur code WLangage) à titre d'exemple dans le répertoire \Programmes\Données\Fenêtres prédéfinies\HFSQL - Fenêtres assistance automatique.
2. Personnaliser la gestion des erreurs pour utiliser les fenêtres HFSQL de gestion des erreurs (fonction ***HSurErreur***).
3. Traduire les différents messages (voir paragraphe précédent).

Gestion automatique des erreurs

Par défaut, les fenêtres de gestion automatique des erreurs sont fournies en anglais et en français. Pour les traduire dans une autre langue, il est nécessaire de :
1. Intégrer les fenêtres d'erreur par défaut dans votre application. Ces fenêtres sont fournies (avec leur code WLangage) à titre d'exemple, dans le répertoire \Programmes\Donnees\Fenêtres prédéfinies\Assertion - Erreur - Exception.
2. Traduire les différents messages (voir paragraphe précédent).

Groupware utilisateur

Par défaut, le groupware utilisateur est livré en anglais et en français.

Pour traduire une application utilisant le groupware utilisateur, il est nécessaire de choisir une intégration personnalisée lors du paramétrage du groupware utilisateur dans votre application. Rappel : Pour paramétrer le groupware utilisateur, sous le volet "Projet", dans le groupe "Projet", cliquez sur "Groupware Utilisateur".

Lors de l'intégration personnalisée, les différents éléments du groupware utilisateur sont intégrés via un composant interne.

Il ne restera plus qu'à traduire ces éléments lors de la traduction du projet.

Remarque : Si nécessaire, lors du paramétrage du groupware utilisateur, dans l'onglet "Exécution", sélectionnez l'option "Lancement manuel" : vous pourrez ainsi afficher une fenêtre de sélection de la langue de l'application avant de lancer la fenêtre de login.

Aperçu avant impression

Par défaut, la fenêtre d'aperçu avant impression est disponible en anglais et français.

Pour traduire la fenêtre d'aperçu avant impression, il est nécessaire d'intégrer les fenêtres correspondant à l'aperçu avant impression dans votre application. Ces fenêtres sont fournies (avec leur code WLangage), dans le répertoire \Programmes\Donnees\Fenêtres prédéfinies\Apercu. Pour plus de détails, consultez l'aide en ligne.

Il ne restera plus qu'à traduire ces fenêtres.

Menus système

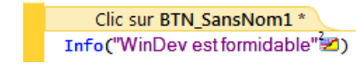
Pour les champs de saisie ou les champs de type Table, WINDEV propose un menu contextuel par défaut. Ce menu permet à l'utilisateur de réaliser simplement diverses opérations.

Pour traduire ces menus, il est nécessaire de posséder WDINT.

3.4 Traduction des messages présents dans le code WLangage

Dans votre code, diverses fonctions du WLangage vous permettent de communiquer avec l'utilisateur au moyen de chaînes de caractères. Ces messages doivent être eux aussi traduits dans les différentes langues de votre application.

Pour traduire une chaîne de caractères saisie dans l'éditeur de code :
1. Sélectionnez la chaîne de caractères à traduire.
2. Dans le menu contextuel de la sélection, choisissez l'option "Traduire". La fenêtre de traduction des messages s'affiche.
3. Saisissez la traduction et validez. La ligne de code apparaît sous l'éditeur de code :



Le drapeau indique que des traductions existent pour cette chaîne de caractères. Pour afficher la fenêtre de traduction, il suffit de cliquer sur ce drapeau.

Le chiffre (2 dans notre exemple) indique le nombre de traductions saisies pour cette chaîne de caractères.

Remarque : Pour transformer tous les messages de votre code en messages multilingues, sous le volet "Code", dans le groupe "Langues", déroulez "Traduire les chaînes" et sélectionnez l'option "Convertir les chaînes simples en messages multilingues".

4 Choix de la langue par programmation

Une application multilingue est une application qui pourra être diffusée dans plusieurs langues. L'utilisateur pourra choisir la langue d'exécution de l'application. Il est possible par exemple :

- de demander la langue d'exécution au premier lancement de l'application
- de prévoir une option (option de menu ou bouton par exemple) permettant à l'utilisateur de changer de langue en cours d'exécution de l'application.

5 Gérer des alphabets spécifiques dans des fichiers de données HFSQL

Lorsqu'une application permet d'afficher ou d'enregistrer des données dans des fichiers de données HFSQL, ces données sont saisies dans la langue en cours.

Si l'application est utilisée dans un pays utilisant des alphabets latins, aucune gestion spécifique n'est nécessaire.

Si l'application est utilisée dans un pays utilisant un alphabet spécifique, il est nécessaire de gérer cet alphabet également au niveau des fichiers de données. Ainsi, les recherches, les tris, ... seront effectués en fonction de l'alphabet choisi.

6 Gestion de l'Unicode et des alphabets spécifiques

WINDEV propose deux modes de gestion :

- **le mode mono alphabet** : Dans ce mode, si votre application gère des langues utilisant des alphabets spécifiques (grec, coréen, ...), il est possible de changer par programmation :
  - l'alphabet utilisé par toutes les polices écran (fonction ***ChangeAlphabet***). Cet alphabet pourra également être utilisé pour les données enregistrées dans les fichiers HFSQL. Pour plus de détails, consultez l'aide en ligne (mot-clé : Multilingue).
  - la langue du clavier manipulé par les utilisateurs de l'application (fonction ***ChangeClavier***).
- **le support complet de Unicode** : Ce mode permet de gérer les alphabets non latins. Dans ce mode, les fonctions ***ChangeAlphabet*** et ***ChangeClavier*** sont inutiles. Le changement d'alphabet est réalisé automatiquement selon les options de langue définies dans la description du projet. Il est également possible de mélanger les différents alphabets.

La configuration du mode de gestion de l'Unicode est réalisée dans les options de la configuration en cours (onglet "Unicode" de la fenêtre de description de la configuration).

Important

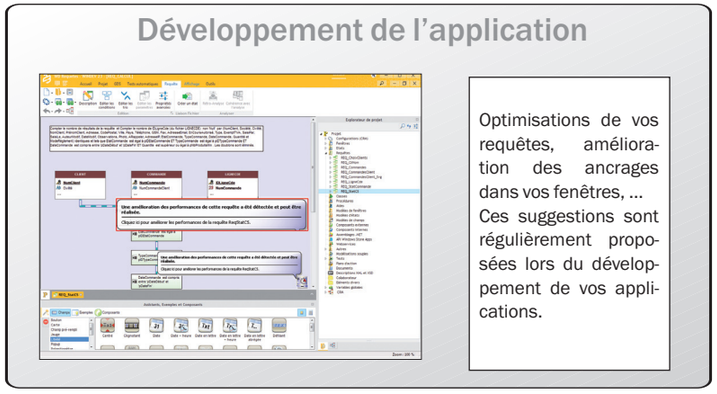
Si vous choisissez de passer vos projets en mode UNICODE, la cohabitation ANSI / Unicode génère toutefois des contraintes : il faut traduire les données lors du passage d'un système de codage à l'autre. L'échange de chaînes texte avec l'extérieur de l'application doit être adapté pour prendre en compte la nouvelle représentation mémoire des chaînes (en Unicode, 1 caractère est codé sur 2 octets alors qu'en ANSI il est codé sur 1 seul octet) : appel d'API, lecture ou écriture sur disque pour échange de données, lecture ou écriture de sockets contenant des chaînes, ...

Si vous choisissez de passer vos projets en mode UNICODE, un assistant mesurant l'impact du passage à Unicode est déclenché sur le projet lors du passage à la norme Unicode.

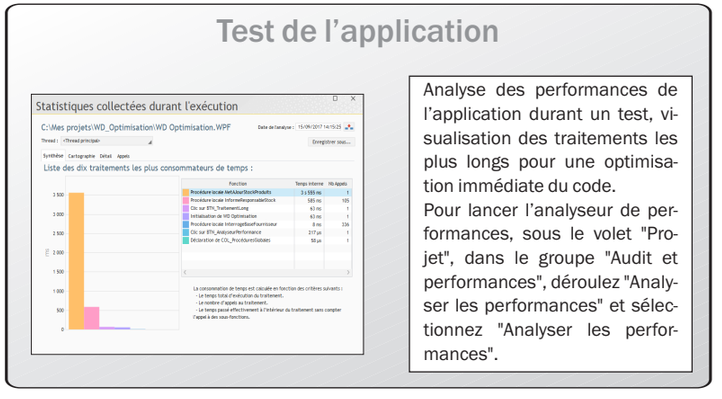
Optimiser une application

Pour optimiser vos applications, WINDEV propose plusieurs méthodes :

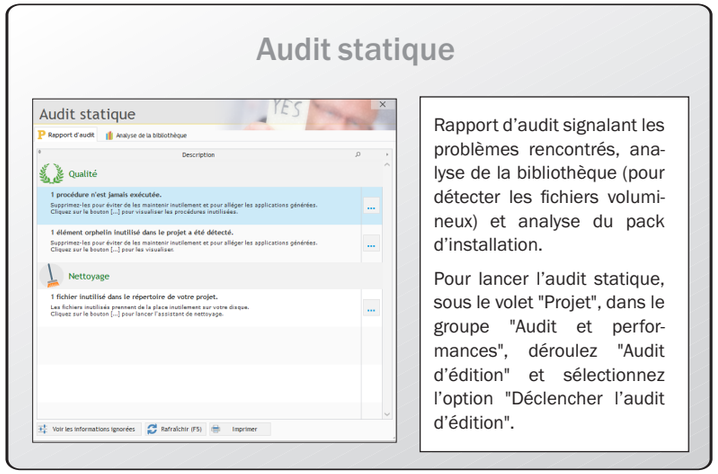
- **Les Aides Automatiques au Développement (AAD)** qui suggèrent des améliorations de code ou d'interface lors du développement de l'application.



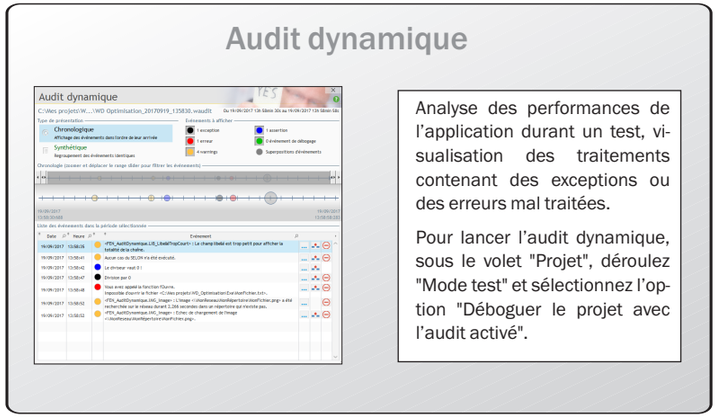
- **L'analyseur de performances** qui permet de visualiser rapidement les traitements les plus gourmands en temps. Il suffit ensuite d'optimiser le traitement voulu.



- **L'Audit statique du projet** qui signale les problèmes potentiels de programmation (groupes vides, éléments orphelins, etc.) ainsi que les fichiers volumineux pouvant entraîner des temps de chargement anormalement longs.

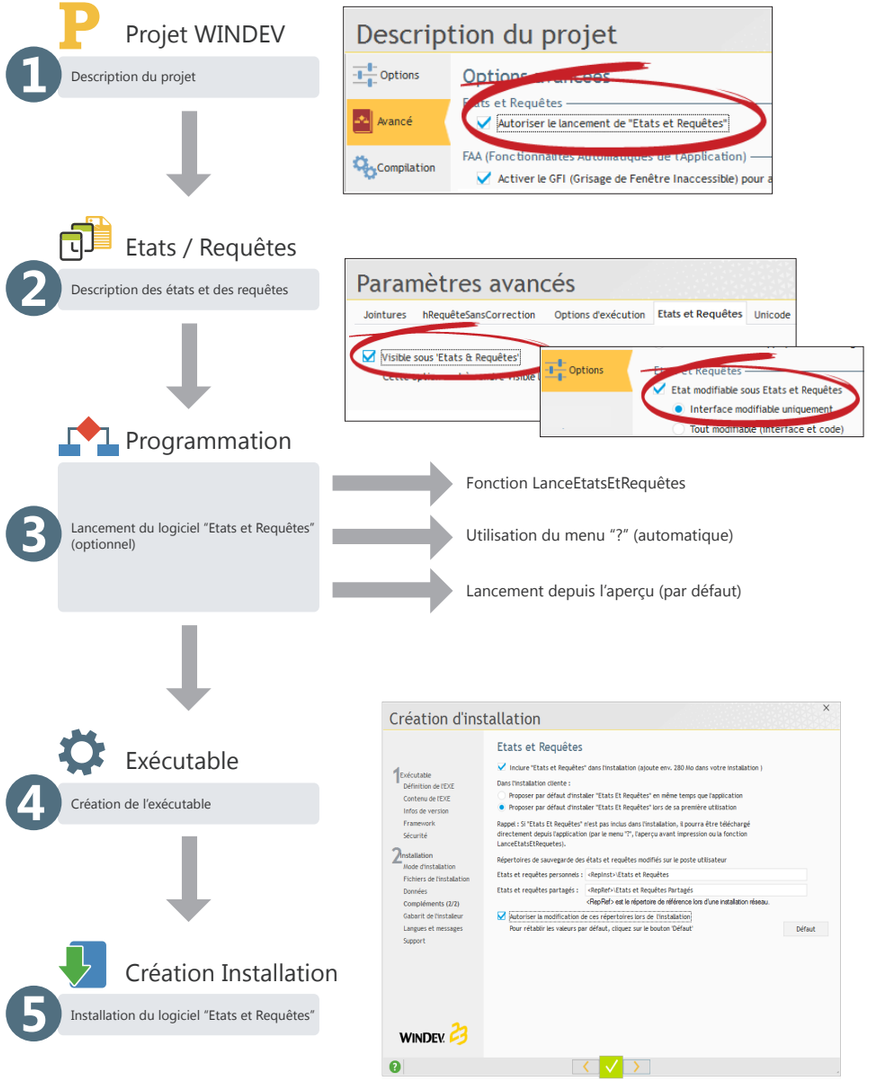


- **L'Audit dynamique du projet** qui détectera les pertes de ressources, les erreurs mal traitées, les lenteurs de l'application dans son environnement de production.



Livrer "Etats et Requêtes"

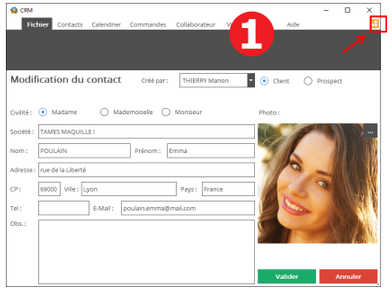
Avec le logiciel "Etats et Requêtes" librement redistribuable avec vos applications, les utilisateurs pourront personnaliser selon leurs souhaits les états que vous leur proposez. Bien sûr, seuls les états que vous aurez choisis seront personnalisables.



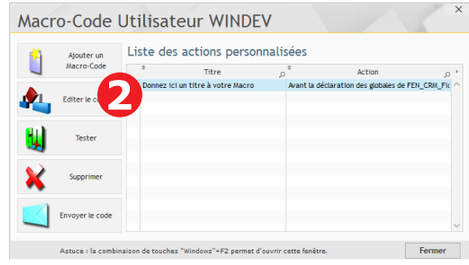
MCU : Macro Code Utilisateur

Le Macro Code Utilisateur (MCU) permet à l'utilisateur final de saisir directement du code WLangage et le greffer sur une application existante.

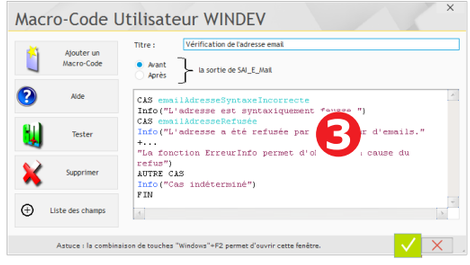
Considérons le cas suivant :
Une fenêtre de saisie d'une fiche client nécessite la présence d'une adresse email, mais ne vérifie pas son format. L'utilisateur souhaite au moins vérifier la présence du caractère "@" dans l'adresse.



Une icône située en haut de la fenêtre permet à l'utilisateur de personnaliser les traitements.



La fenêtre de configuration du MCU apparaît. L'utilisateur peut ajouter une nouvelle action pour cette fenêtre. Dans ce cas, l'utilisateur ajoute la vérification de l'adresse email.



L'utilisateur saisit le code WLangage. L'aide du WLangage peut être lancée grâce à la touche F1.

Pour activer cette fonctionnalité (qui est désactivée par défaut), il suffit d'indiquer que le projet accepte le MCU :

- soit dans la description du projet,
- soit lors de la création de l'exécutable.

Macro Code Utilisateur : MCU en pratique

1 Qu'est-ce que le MCU ?

Le MCU est une option permettant aux utilisateurs avancés de réaliser leurs propres traitements dans l'application. L'utilisateur final peut saisir du code WLangage et le greffer sur une application existante.

Il est possible de faire remonter au développeur les traitements ajoutés afin que ceux-ci soient intégrés dans les sources de l'application.

De nombreuses fonctionnalités du WLangage sont accessibles à l'utilisateur et des restrictions peuvent être appliquées par le développeur. Par exemple interdire l'utilisation des fonctions de gestion des fichiers. Afin d'assurer l'intégrité des données, les ordres de modification ou d'ajout d'enregistrement ne sont pas disponibles pour l'utilisateur.

2 Comment activer/désactiver le MCU ?

Il est bien sûr possible de débrancher cette fonctionnalité, par fenêtre ou pour l'application entière si vous le désirez.

Pour désactiver le MCU sur une fenêtre, il faut cocher l'option "Interdire l'utilisation du M.C.U." dans l'onglet "Détail" de la description d'une fenêtre.

Pour gérer le MCU sur le projet :
1. Affichez la fenêtre de description du projet : sous le volet "Projet", dans le groupe "Projet", cliquez sur "Description".
2. Affichez l'onglet "Avancé". En cliquant sur le bouton "Macro Code Utilisateur", vous avez la possibilité d'autoriser la saisie de Macro Code Utilisateur ou de l'interdire pour l'application complète. Si vous l'autorisez, deux options sont alors disponibles :

- Autoriser l'utilisation des ordres HFSQL (HLitPremier, HFiltre, HAjoute, ...).
- Accepter les suggestions faites par les utilisateurs.

Cette deuxième option permet l'envoi d'un email par l'application lorsque l'utilisateur crée un macro code. Le fichier "*.MCU" correspondant à la macro code est envoyé en pièce jointe. Vous récupérez ainsi facilement les suggestions des clients et les intégrer automatiquement dans votre projet ! WINDEV dispose en effet d'une option d'importation des fichiers "*.MCU" (sous le volet "Projet", dans le groupe "Projet", déroulez "Importer" et sélectionnez l'option "Des Macro-Codes Utilisateurs ...").

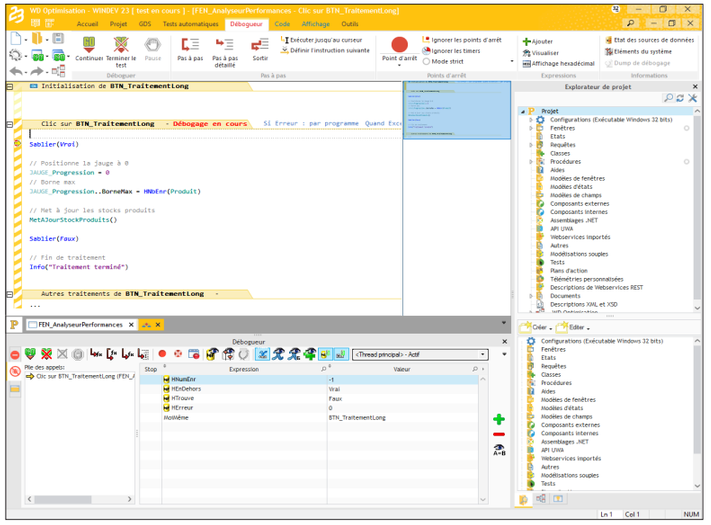
Les différents tests disponibles

WINDEV propose plusieurs méthodes pour tester votre application. Le tableau ci-dessous présente ces différents modes.

Type de test	Description
Test de l'ensemble du projet	Simule le lancement de l'application par l'exécutable. Permet de tester l'application dans son ensemble, même si son développement n'est pas terminé. Dès qu'un problème apparaît dans l'exécution du projet, vous pouvez lancer le débogueur pour connaître et solutionner le problème rencontré.
Test de l'élément en cours	Permet de tester le fonctionnement de cet élément dès que son développement est terminé. Pour les états ou les fenêtres, il est possible de lancer le débogueur dès qu'un problème est rencontré.
Test d'un projet pas à pas (grâce au débogueur)	Permet de lancer le débogueur au lancement de l'application. Cette solution permet de suivre méticuleusement le déroulement de l'application à l'aide du débogueur. Il est également possible de débogueur un exécutable spécifique, un exécutable déjà lancé ou un exécutable lancé sur un autre poste.
Test unitaire	Permet de tester les fenêtres, les procédures et les classes d'une application à différents niveaux de développement. Les scénarios de test sont automatiquement générés en WLangage à partir du test de l'application ou du test d'un élément. Les scénarios peuvent être directement modifiés sous l'éditeur de code.
Test de non-régression	Permet de tester la non-régression d'un élément entre les différentes versions d'une application. Il suffit de sélectionner (ou de créer) le scénario de test à exécuter.

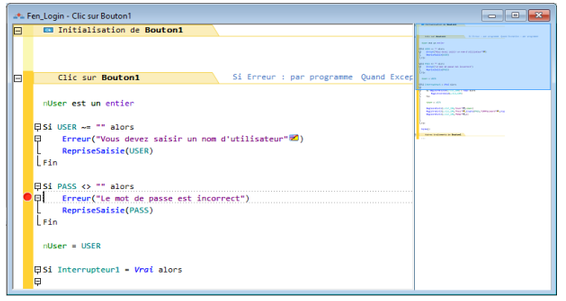
Le débogueur

Pour mettre au point votre code, un débogueur est livré en standard avec WINDEV.
Le débogueur permet une exécution pas à pas de vos applications, en visualisant le code WLangage exécuté, le contenu des variables, ...



Débogage du projet en cours

1. Utilisation d'un point d'arrêt, placé dans l'éditeur de code.
Lorsque la ligne précédée d'un point d'arrêt sera exécutée, le débogueur sera automatiquement lancé.
Pour poser un point d'arrêt, il suffit de cliquer devant la ligne de code : une puce rouge apparaît.



2. Utilisation du mot-clé STOP du WLangage dans le code WLangage
Le débogueur sera automatiquement lancé dès que cette ligne de code sera exécutée.

3. Utilisation de l'option "Tracer le projet"
Le débogueur est lancé en même temps que le test de l'application. L'option "Tracer le projet" est disponible sous le volet "Projet", en déroulant "Mode test".

4. Utilisation des touches [CTRL] + [PAUSE] pendant le test de la fenêtre ou du projet
Le débogueur est lancé lors de la prochaine action effectuée après l'utilisation des touches [CTRL] + [PAUSE].

Débogage d'un exécutable

WINDEV permet également de déboguer directement l'exécutable associé au projet en cours sous WINDEV. Cet exécutable peut :

- être lancé depuis WINDEV, sur le poste de développement.
- être déjà en cours d'exécution sur le poste de développement.
- être lancé depuis WINDEV, sur un poste distant.
- être déjà en cours d'exécution sur un poste distant.

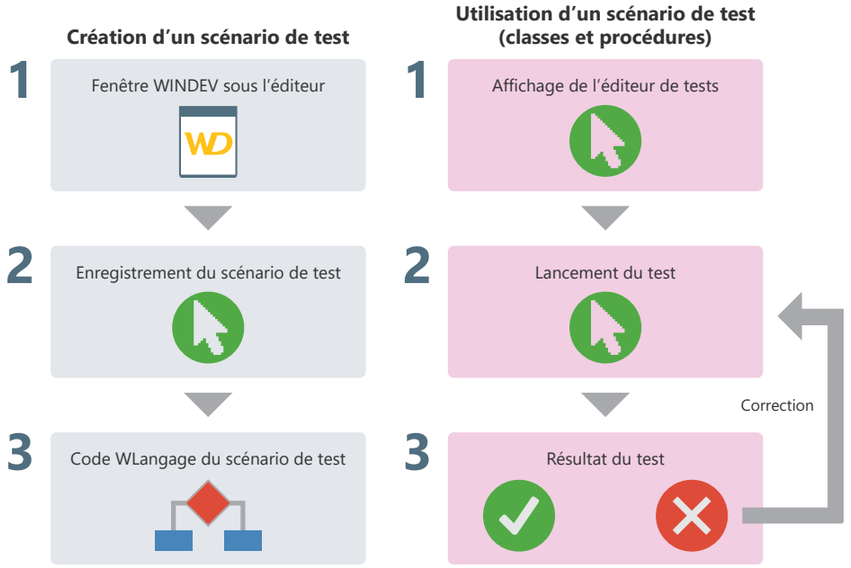
Débogage d'un composant

Le débogage d'un composant est réalisé à partir du projet du composant. Il est possible de :

- Tester le composant depuis le projet du composant.
- Tester le composant en exécutant un autre projet présent sur le poste de développement. Il est ainsi possible de tester le composant à partir du projet qui l'utilise.
- Tester le composant en exécutant un exécutable présent sur le poste de développement. Il est ainsi possible de tester le composant directement à partir de l'exécutable qui l'utilise.

Tests unitaires/ Tests automatiques

Les tests unitaires (appelés également tests automatiques) permettent de tester les fenêtres, les procédures et les classes d'une application à différents niveaux de développement. Il suffit de sélectionner (ou de créer) le scénario de test à exécuter.
Ces scénarios sont générés en WLangage et peuvent être directement modifiés.
Ces scénarios sont regroupés dans l'Editeur de Tests. L'Editeur de Tests analyse le résultat des tests unitaires et calcule le taux de validation de l'application.

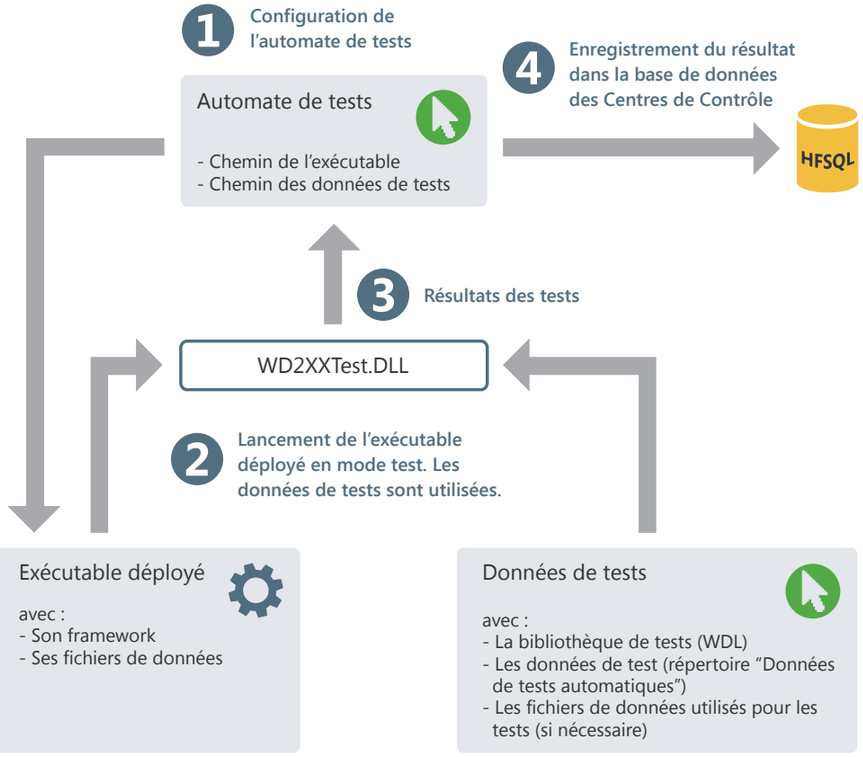


Pour créer un test unitaire sur la fenêtre en cours, sous le volet "Tests automatiques", dans le groupe "Tests", déroulez "Nouveau" et sélectionnez "Enregistrer un nouveau scénario".

Pour créer un test unitaire sur une procédure ou une classe, sélectionnez cette procédure ou cette classe dans le volet "Explorateur de projet" et sélectionnez l'option "Tests unitaires .. Créer un test unitaire" du menu contextuel.

Tests unitaires sur l'exécutable

Il est possible de passer les tests unitaires créés avec WINDEV sur un exécutable déployé, grâce à l'automate de tests.



Test d'une application en pratique

1 Présentation

WINDEV propose plusieurs méthodes pour tester vos applications :

- test de l'ensemble du projet,
- test d'une fenêtre seule,
- test d'une requête seule (consultez le livre "Etats et Requêtes" pour plus de détails),
- test d'un état seul (consultez le livre "Etats et Requêtes" pour plus de détails),
- exécution du projet pas à pas,
- test des performances de votre application,
- test de non-régression/test automatique.

Le test de l'ensemble du projet permet de simuler le lancement de l'application par l'exécutable. Il est ainsi possible de tester l'application dans son ensemble, même si son développement n'est pas terminé. Dès qu'un problème apparaît, vous pouvez lancer le débogueur pour connaître et solutionner le problème rencontré.

Le test d'une fenêtre permet d'exécuter uniquement la fenêtre en cours. Vous pouvez ainsi choisir de tester votre projet à partir d'une fenêtre donnée, ou de tester le fonctionnement d'une fenêtre dès que son développement est terminé. Comme pour le test du projet, il est possible de lancer le débogueur dès qu'un problème est rencontré.

Le test d'une requête seule permet d'exécuter uniquement la requête en cours. Vous pouvez ainsi choisir de tester le fonctionnement d'une requête dès que son développement est terminé.

Le test d'un état seul permet d'exécuter uniquement l'état en cours. Vous pouvez ainsi choisir de tester le fonctionnement d'un état dès que son développement est terminé. Comme pour le test du projet, il est possible de lancer le débogueur dès qu'un problème est rencontré.

L'exécution du projet pas à pas permet de lancer le débogueur au lancement de l'application. Cette solution permet de suivre méticuleusement le déroulement de l'application.

Le test des performances de votre application permet de vérifier et d'optimiser le temps d'exécution de votre application.

Le test de non-régression (ou test automatique) est basé sur l'exécution de scripts. Il permet de vérifier que lors de l'exécution de vos applications, les fonctionnalités existantes sont toujours supportées.

Pour compléter ces différentes méthodes, WINDEV propose également de connaître le "Code coverage" de l'application, c'est-à-dire la mesure de la couverture des tests réalisés sur une application. Pour plus de détails, consultez l'aide en ligne.

2 Tester et déboguer un projet WINDEV (poste de développement)

2.1 Présentation

Le test de l'ensemble du projet WINDEV permet de simuler le lancement de l'application par l'exécutable. Il est ainsi possible de tester l'application dans son ensemble, même si son développement n'est pas terminé. Dès qu'un problème apparaît dans l'exécution du projet, vous pouvez lancer le débogueur pour connaître et solutionner le problème rencontré.

WINDEV présente une palette complète de tests sur une application pouvant être réalisés à partir du poste de développement :

- Test et débogage du projet en cours sous l'éditeur.
- Lancement et débogage de l'exécutable correspondant au projet en cours.
- Débogage d'un exécutable déjà lancé sur le poste de développement.

Remarque : Le test d'un projet peut être lancé quel que soit l'élément en cours sous l'éditeur.

2.2 Tester et déboguer le projet en cours

Pour tester un projet depuis l'éditeur, sous le volet "Projet", déroulez "Mode test" et sélectionnez l'option "Déboguer le projet" (combinaison de touches [Ctrl] + [F9]). L'éditeur est automatiquement réduit en icône et le projet s'exécute.

2.3 Lancer l'exécutable et le déboguer sur le poste de développement

Certains problèmes ne peuvent être reproduits que lors d'un test de l'exécutable en condition d'utilisation réelle. Ce mode de test permet par exemple de déboguer un exécutable après l'avoir installé sur le poste de développement.

Dans ce cas, le code exécuté provient directement des bibliothèques (et non du code présent dans les éléments présents sur le poste). Le code peut donc être déphasé entre le code du projet en cours et le code de l'exécutable : le code déphasé est surligné en rose lors du débogage.

Ce mode de test permet de lancer un exécutable spécifique installé sur le poste de développement et de le déboguer directement.

Pour plus de détails, consultez l'aide en ligne.

2.4 Débogage d'un exécutable déjà lancé sur le poste de développement

Certains problèmes ne peuvent être reproduits que lors d'un test de l'exécutable en condition d'utilisation réelle.

Ce mode de test permet par exemple de trouver un problème qui se produirait uniquement dans l'exécutable (et non en tests).

Le principe est simple :

- Le développeur lance l'exécutable sur son poste et se positionne directement à l'endroit voulu dans l'application.
- Dans WINDEV, le développeur ouvre le projet correspondant à l'exécutable et demande à s'attacher à l'exécutable lancé.
- Le débogueur sous WINDEV permet de suivre l'exécution du programme pas à pas.

Dans ce cas, le code exécuté provient directement des bibliothèques (et non du code présent dans les éléments présents sur le poste). Le code peut donc être déphasé entre le code du projet en cours et le code de l'exécutable : le code déphasé est surligné en rose lors du débogage.

Pour plus de détails, consultez l'aide en ligne.

3 Tester et déboguer un projet WINDEV sur le poste de l'utilisateur

WINDEV offre plusieurs possibilités pour tester et déboguer un projet directement sur le poste de développement. Mais dans certains cas, il est nécessaire de déboguer directement sur le poste de l'utilisateur final (connecté par réseau ou par Internet).

Il est ainsi possible de déboguer, depuis votre bureau de Paris, une application qui s'exécute à Taïwan. Le débogage est effectué sans se déplacer, directement sur la configuration du client.

Deux fonctionnalités sont disponibles :

- Lancement et débogage de l'exécutable sur une machine distante
- Débogage d'une application en cours d'exécution sur une machine distante.

Pour ces deux fonctionnalités, une configuration spécifique de la machine distante est nécessaire.

Remarque : pour déboguer une application WINDEV à distance, l'application WINDEV doit inclure la librairie WD230CPL.DLL dans son framework.

Pour plus de détails, consultez l'aide en ligne.

4 Tester une fenêtre

4.1 Tester la fenêtre depuis l'éditeur

Pour tester une fenêtre depuis l'éditeur :
1. Ouvrez la fenêtre à tester.
2. Cliquez sur parmi les boutons d'accès rapide (ou [F9]). L'éditeur est automatiquement réduit en icône et la fenêtre s'exécute.

Lors du test, l'ensemble des fonctionnalités de la fenêtre pourra être exécuté. Il sera possible par exemple d'ouvrir d'autres fenêtres.

4.2 Arrêter le test d'une fenêtre

Pour arrêter le test, plusieurs méthodes sont possibles :

- 1ère méthode :
  Fermez l'application en cours de test. WINDEV affiche l'éditeur en cours au moment du lancement du test.
- 2ème méthode :
  - Revenez dans l'éditeur avec la barre des tâches ou avec [Alt] + [Tab].
  - Confirmez l'arrêt du test. WINDEV affiche l'éditeur en cours au moment du lancement du test.

5 Tracer un projet

5.1 Principes du débogage

Le débogage d'une application consiste à :

- vérifier le bon fonctionnement d'un traitement,
- comprendre le fonctionnement d'un programme existant,
- vérifier la valeur des variables,
- vérifier le bon fonctionnement de cas particuliers dans une application.

Le débogueur permet de réaliser ces opérations.

Remarque : WINDEV met également à votre disposition divers outils de trace (fenêtre de trace, boîte d'information, ...). Pour plus de détails, consultez le paragraphe "Déboguer sans le débogueur", page 155.

5.2 Présentation du débogueur

Le débogueur permet de tracer les programmes en WLangage afin de faciliter la mise au point de programmes.

Le code source exécuté est visualisé à l'écran. Les différents traitements exécutés sont hiérarchisés dans le volet "Débogueur".

La valeur des variables peut être visualisée :

- individuellement dans la bulle d'aide de survol de chaque variable.
- dans le volet "Débogueur".

5.3 Fonctionnalités du débogueur

Le débogueur permet de :

- connaître la pile des appels,
- visualiser le contenu des variables ou des expressions,
- exécuter pas à pas avec possibilité de sauter des blocs,
- utiliser des points d'arrêt conditionnels,
- modifier le code tout en continuant l'exécution,
- ...

5.4 Déboguer sans le débogueur

Dans certains cas, l'exécution d'un programme avec ou sans le débogueur peut être différente.

En effet, le débogueur introduit des pauses dans l'exécution du traitement, durant lesquelles Windows effectue certaines tâches.

Ainsi, le débogueur ne peut pas être utilisé dans une procédure appelée par timer ou dans le code de l'ascenseur.

Remarque : Pour connaître l'ensemble des limites du débogueur, consultez l'aide en ligne.

Pour déboguer ces applications, il peut être nécessaire par exemple de suivre l'évolution d'une valeur, le passage dans différentes procédures, ...

Ces informations peuvent être :

- affichées à l'écran.
- stockées dans un fichier de trace.

Attention : Si les informations sont affichées à l'écran, elles doivent être affichées uniquement lors des tests de l'application.

Afficher des informations

Deux outils permettent d'afficher des informations :

- ***les boîtes d'information*** : fonction ***Info*** du WLangage.
  Attention : L'affichage d'une boîte d'information est bloquante.
- ***la fenêtre de trace*** : fonction ***Trace*** du WLangage.
  La fenêtre de trace s'affiche en haut à gauche de l'écran, sans interrompre le déroulement du programme.

Gérer l'affichage des informations de débogage

L'affichage à l'écran des informations de débogage est utile uniquement en mode test.

Avant de diffuser une application, il est donc nécessaire de supprimer tout affichage superflu.

Pour éviter tout oubli, il est conseillé de gérer l'affichage des informations de débogage à l'aide d'une procédure globale. Par exemple :

```
PROCEDURE MaTrace(ChaîneATracer)
SI EnModeTest()=Vrai ALORS
      Trace(ChaîneATracer)
FIN
```

Dans ce code, selon le résultat de la fonction ***EnModeTest***, la fenêtre de trace apparaît uniquement lors d'un test de l'application.

Une telle procédure permet de laisser l'appel aux fenêtres de trace dans le code de l'application, sans risque d'apparition en clientèle.

L'appel à cette procédure de trace est identique à l'utilisation de la fonction ***Trace*** :

```
MaTrace("Client : "+ ...
Client.NumClient)
```

Créer un fichier de trace

Lors de traitements longs (traitements Batchs, ...), pour contrôler le bon déroulement du programme, il est nécessaire de conserver une trace physique des traitements effectués (un fichier texte par exemple).

La procédure suivante permet de gérer par exemple l'affichage de la trace :

- soit à l'écran (paramètre /DEBUG en ligne de commande).
- soit dans un fichier texte (mode par défaut).

```
PROCEDURE MaTrace(ChaîneATracer)
CheminFichier est une chaîne
CheminFichier = ...
   fRepDonnéesUtilisateur()+...
   ProjetInfo(piNomProjet) + ".txt"
Fichier est un entier
ModeDebug est un booléen = Faux
SI Position(LigneCommande(),...
        "/DEBUG") > 0 ALORS
      ModeDebug = Vrai
FIN
```

```
SI ModeDebug = Vrai ALORS
    Trace(ChaîneATracer)
SINON
    Fichier = fOuvre(...
     CheminFichier, ...
     foCréationSiInexistant +...
     foEcriture + foAjout)
    SI Fichier <> -1 ALORS
     DateHeureTrace est un DateHeure
     DateTrace est une Date
     HeureTrace est une Heure
     DateHeureTrace = DateHeureSys()
     DateTrace = Madate..PartieDate
    HeureTrace = Madate..PartieHeure
   fEcritLigne(Fichier,...
  DateVersChaîne(DateTrace) +...
  " -
"+ HeureVersChaîne(HeureTrace))
fEcritLigne(Fichier, ChaîneATracer)
    fEcritLigne(Fichier, " ")
    fFerme(Fichier)
  FIN
FIN
```

Remarques :

- Le fichier de trace est créé par défaut dans le répertoire des données de l'utilisateur. Ce fichier a pour nom le nom du projet. Ce fichier contient les informations à tracer durant l'exécution du programme. Les informations sont complétées par la date et l'heure de chaque "Trace". Il est ainsi possible de déterminer un éventuel dysfonctionnement durant le traitement.
- Exemple de contenu du fichier de trace :

```
01/12/2015 - 10:53:25:20
Nom de client : Martin
```


6 Test de performances

6.1 Présentation

L'analyseur de performances est un outil permettant de vérifier et d'optimiser le temps d'exécution de votre application.

Son principe est simple :

Vous testez votre application. Pendant ce test, l'analyseur de performances répertorie toutes les actions effectuées et les traitements correspondants exécutés.

A la fin du test, l'analyseur de performances vous présente :

- les 10 manipulations qui ont pris le plus de temps
- toutes les actions effectuées dans l'application testée, triées par durée (de l'action la plus longue à l'action la moins longue).

Il est alors possible de sélectionner un traitement afin d'analyser les causes de son temps de traitement pour l'optimiser.

6.2 Lancer l'analyseur de performances

Pour lancer l'analyseur de performances, sous le volet "Projet", dans le groupe "Audit et performances", déroulez "Analyser les performances" et sélectionnez l'option "Analyser les performances".

Le projet est alors automatiquement exécuté en mode test. Vous pouvez exécuter le traitement à optimiser dans votre application.

Pour revenir sous l'éditeur, il suffit de fermer votre application.

L'analyseur de performances affiche alors le résultat de l'analyse.

Remarque : Il est conseillé d'utiliser l'analyseur de performances pour optimiser votre application (avant sa diffusion par exemple).

6.3 Lire le résultat de l'analyseur de performances

L'analyseur de performances présente le résultat de l'analyse dans plusieurs onglets :

- l'onglet "Synthèse" présente les dix traitements qui ont pris le plus de temps.
- l'onglet "Cartographie" présente une vision graphique des traitements les plus importants.
- l'onglet "Détail" présente tous les traitements lancés lors du test de l'application (classés du plus long au plus rapide).
- l'onglet "Appels" permet de visualiser le détail des opérations réalisées dans un traitement.

Pour chaque traitement, les informations suivantes sont affichées :

Fonction	Fonction, traitement ou procédure exécutée.
Temps Total	Temps d'exécution de la fonction.
Nb appels	Nombre d'appels effectués à la fonction (procédure ou traitement).
Temps 1 appel	Temps d'exécution d'un appel à la fonction (procédure ou traitement).
% code	Pourcentage de code exécuté hors appel à une fonction WLangage ou à un appel d'une fonction ou une procédure personnelle.
Parent	Traitement qui a appelé la fonction.

Remarques :

- Le libellé "Exécution complète" représente le temps complet de l'exécution du test de l'application réalisé avec l'analyseur de performances.
- Le libellé "Total Fenêtre XXX" représente le temps complet de l'exécution de la fenêtre XXX (de son ouverture à sa fermeture).

6.4 Choisir un traitement à optimiser

Le choix du traitement à optimiser se fait en fonction de plusieurs critères :

- ***le temps d'exécution du traitement***. Les traitements les plus longs doivent bien entendu être optimisés.
- ***le pourcentage de temps passé dans le traitement de la fonction ou de la procédure***. Plus ce pourcentage est important, plus le code peut contenir des traitements pouvant être optimisés.

Remarque : Si le traitement correspond à une fonction WLangage, il est relativement difficile de l'optimiser.

7 Tests de non-régression

7.1 Présentation

Soucieux de la qualité des applications, plusieurs outils de tests sont à votre disposition :

- Le mode test (Go de projet ou Go de fenêtre) qui permet de tester immédiatement une modification dans votre application.
- Les tests automatiques qui permettent d'enregistrer un scénario de test facilement re-jouable.
- L'utilitaire WDTest qui permet par exemple de créer des tests de validation et de non-régression en WINDEV.

Pour automatiser ces tests et augmenter la qualité de vos applications, vous pouvez faire **des tests unitaires automatiques**. Grâce à ces tests, il est encore plus simple de contrôler toutes les fonctionnalités proposées par vos applications.

7.2 Tests automatiques

Chaque test est composé d'un scénario directement éditable dans l'interface du produit. Ce scénario est écrit en WLangage et peut être modifié à n'importe quel moment.

Ces tests peuvent être lancés par exemple avant chaque création d'exécutable pour vérifier le bon fonctionnement d'une application après diverses modifications.

Les éléments suivants peuvent être testés :

- les fenêtres WINDEV.
- les collections de procédures.
- les classes.
- les exécutables WINDEV.

Chaque test est associé à un code WLangage : le scénario du test. Ce scénario est visible sous l'éditeur de code. Le code des tests peut être modifié.

Les tests et le code associé ne sont pas intégrés à l'exécutable et ne sont pas livrés en clientèle. Le nombre de tests d'une application n'a donc aucune incidence sur la taille de l'application livrée en clientèle.

Pour plus de détails, consultez l'aide en ligne (mot-clé : "Test automatique").

7.3 L'utilitaire WDTest

WDTest est un utilitaire permettant d'enregistrer et d'exécuter automatiquement des scripts de test d'applications (WINDEV ou non WINDEV).

Un script de test (appelé également "macro") contient principalement les différentes manipulations (position de la souris, clic de souris et touche du clavier utilisé) effectuées sur une application.

Après avoir été enregistré, le script de test peut être exécuté à volonté. Lorsqu'un script est exécuté, un script de résultat est automatiquement généré. Les différences existantes entre les deux scripts (script d'origine et script de résultat) sont signalées.

Il est ainsi possible par exemple de vérifier le fonctionnement de vos applications WINDEV entre deux mises à jour.

WDTest permet donc d'automatiser les procédures de test de vos applications.

Important : Pour effectuer un test avec succès, il est nécessaire d'utiliser exactement la même configuration :

- sur le poste où le test est créé.
- sur le poste sur lequel le test est exécuté.

Pour plus de détails, consultez l'aide en ligne (mot-clé : "WDTest").

Le Framework WINDEV

Le Framework WINDEV est l'ensemble des librairies (fichiers .DLL) nécessaires au fonctionnement d'une application WINDEV.
Lors de la création de l'exécutable, vous pouvez choisir d'utiliser :

- le Framework à côté de l'exécutable,
- le Framework intégré à l'exécutable,
- le Framework WINDEV commun.

Utilisation d'un Framework à côté de l'exécutable

Dans le cas de l'utilisation d'un Framework à côté de l'exécutable, les librairies sont installées dans le répertoire de chaque application. De cette façon, chaque application WINDEV utilise sa propre version des librairies. Il est possible de renommer les librairies dans le cas de l'utilisation d'un Framework à côté de l'exécutable.

Avantages du Framework à côté de l'exécutable :

- Chaque application peut exploiter une version différente des DLL du Framework.
- La mise à jour du Framework d'une application peut être faite sans perturber le fonctionnement des autres applications.
- Le Framework peut être renommé.

Utilisation du Framework intégré à l'exécutable

Dans le cas de l'utilisation du Framework intégré à l'exécutable, l'exécutable est autonome. Les DLL du Framework ne sont pas extraites au lancement de l'exécutable.

Avantages du Framework intégré à l'exécutable :

- Exécutable autonome.
- Pas d'extraction de DLL du Framework.

Remarque : Selon les fonctionnalités utilisées par l'application, l'utilisation du Framework intégré à l'exécutable peut ne pas être possible.

Utilisation du Framework WINDEV commun

Dans le cas de l'utilisation du Framework commun, les librairies sont installées dans un répertoire commun. Ces librairies sont partagées par toutes les applications WINDEV installées sur le même poste.

Avantages du Framework commun :

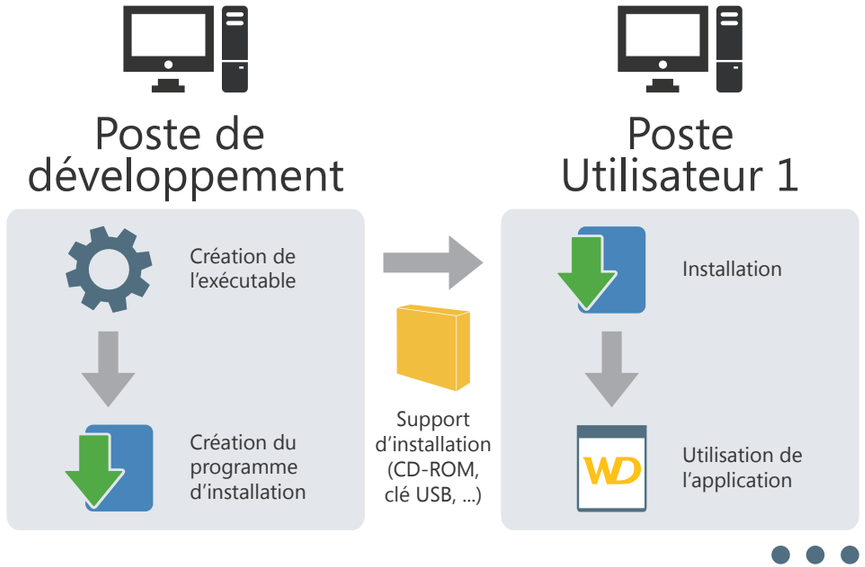
- L'espace disque utilisé par le Framework (environ 30 Mo) est mutualisé entre les applications.
- La mise à jour du Framework ne se fait qu'une fois pour toutes les applications installées.
- Le Framework commun peut être renommé.

Remarque : Dans le cadre de l'utilisation du Framework commun, il est possible de remplacer une ou plusieurs DLL (par exemple si une version particulière est nécessaire).

Les différents modes d'installation d'une application

Installation individuelle

L'installation individuelle consiste à créer un programme d'installation qui sera installé sur chaque poste utilisateur grâce à un support d'installation.

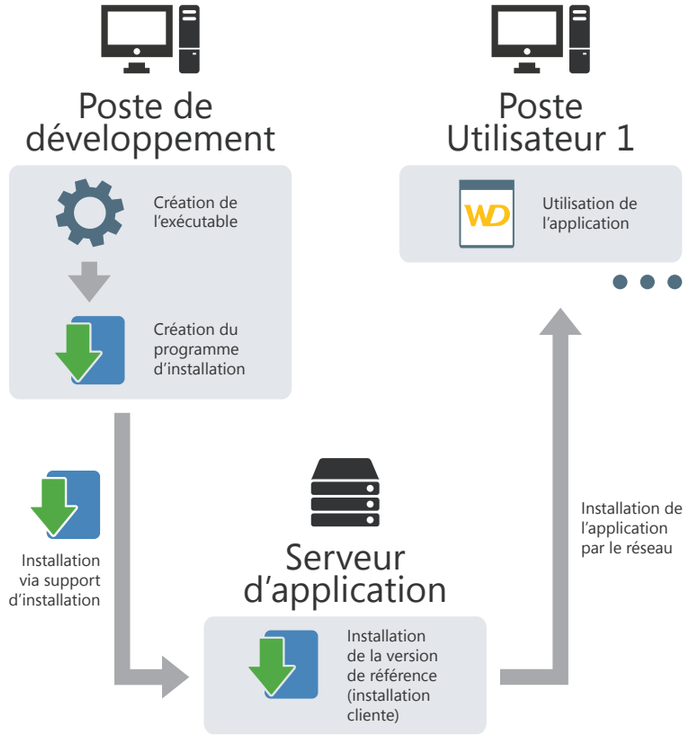


Ce mode d'installation est compatible aussi bien avec :

- des fichiers de données au format HFSQL Classic installés en local sur le poste utilisateur,
- des fichiers de données au format HFSQL Classic installés en réseau et accessibles depuis le poste utilisateur,
- des fichiers de données au format HFSQL Client/Serveur installés sur un serveur.

Installation réseau (avec mise à jour automatique)

L'installation réseau consiste à créer un programme d'installation qui sera installé sur un poste Serveur. Tous les postes utilisateur voulant installer l'application, lanceront directement l'installation de l'application par le réseau.



Si une mise à jour est disponible sur le serveur d'application, la mise à jour de l'application sur le poste utilisateur sera proposée au lancement de l'application.

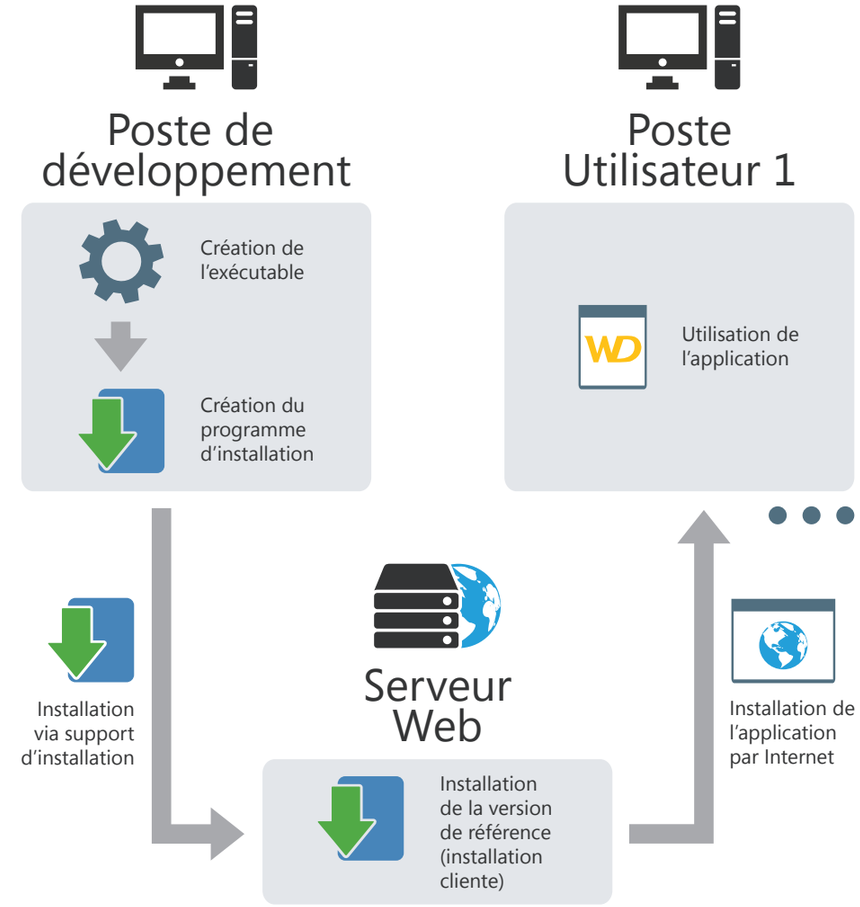
Une installation réseau permet un contrôle fin des postes utilisateurs (versions déployées, mises à jour obligatoires, déconnexion des utilisateurs, etc.)

Ce mode d'installation est compatible aussi bien avec :

- des fichiers de données au format HFSQL Classic installés en local sur le poste utilisateur,
- des fichiers de données au format HFSQL Classic installés en réseau et accessibles depuis le poste utilisateur,
- des fichiers de données au format HFSQL Client/Serveur installés sur un serveur.

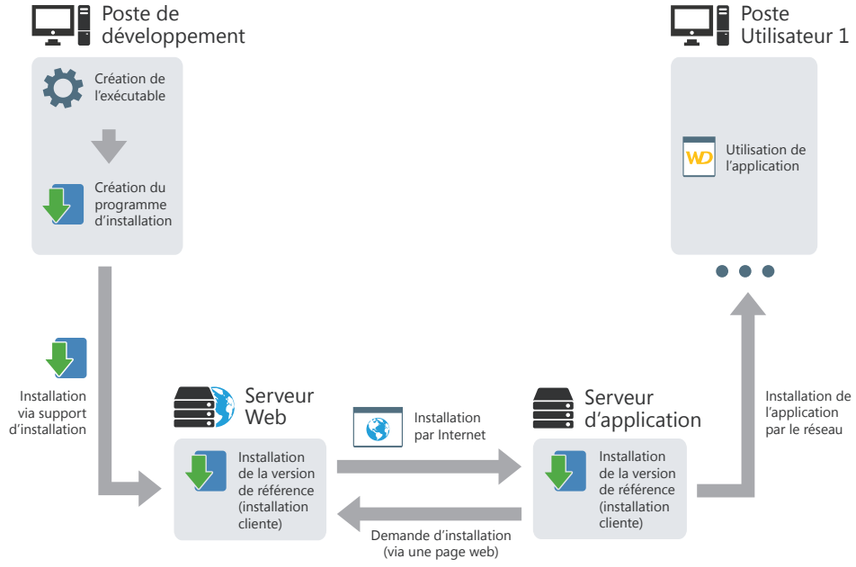
Installation HTTP (avec mise à jour automatique)

L'installation HTTP consiste à créer un programme d'installation qui sera installé sur un Serveur Web.
Tous les postes utilisateur voulant installer l'application, lanceront directement l'installation de l'application par une URL HTTP.
Si une mise à jour est disponible sur le serveur Web, la mise à jour de l'application sur le poste utilisateur sera proposée au lancement de l'application.



Installation multi-sites (avec mise à jour automatique)

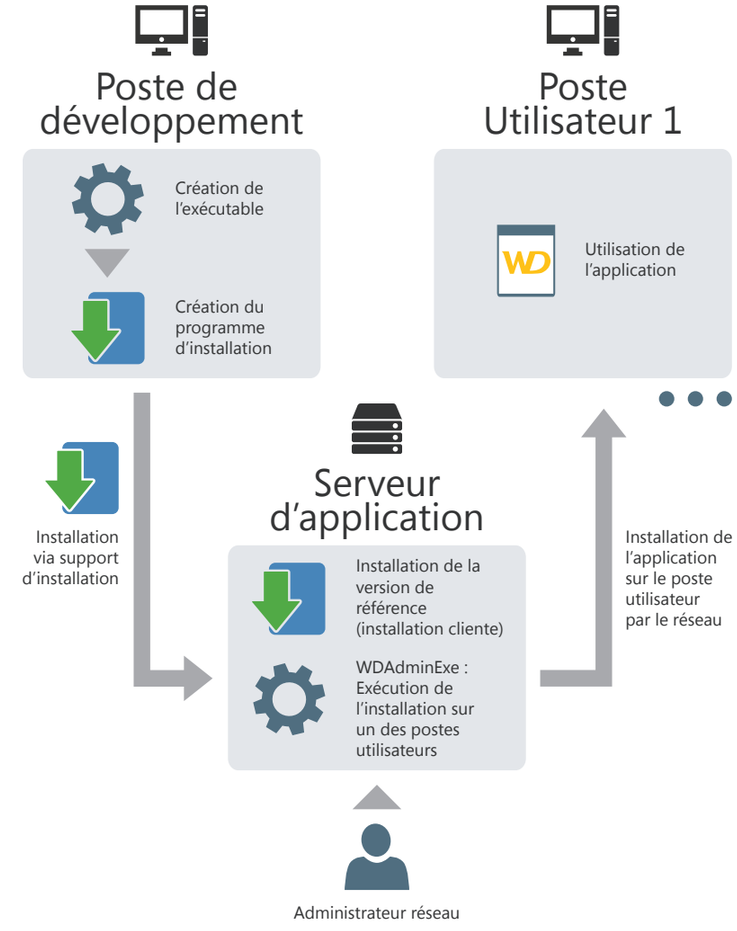
L'installation multi-sites permet d'installer une application sur plusieurs sites différents (un site représentant par exemple une filiale d'une entreprise).
L'installation multi-sites consiste à créer un programme d'installation qui sera installé sur un serveur Web.
Tous les sites voulant installer l'application lanceront directement l'installation de l'application par une URL HTTP. Dans ce cas, le programme d'installation sera alors installé sur un poste Serveur du site.
Tous les postes utilisateurs voulant installer l'application, lanceront l'installation directement par le réseau.
Cette installation combine les avantages de l'installation réseau et de l'installation HTTP en permettant un déploiement à grande échelle et un contrôle fin des installations locales.



Si une mise à jour est disponible sur le serveur Web, la mise à jour de l'application sur le serveur d'application est automatiquement réalisée.

Installation réseau de type Push

L'installation réseau consiste à créer un programme d'installation qui sera installé sur un poste Serveur.
L'installation sera effectuée depuis le poste Serveur, par l'administrateur réseau sur les postes Utilisateur sélectionnés grâce à WD2xxAdminExe.



Une installation en Push peut ensuite être mise à jour soit en Push, soit dans le cadre d'une installation réseau.

Déploiement d'une application WINDEV

Lors du développement d'une application, une des phases importantes consiste à déployer cette application sur les postes des utilisateurs.

WINDEV optimise ce déploiement en offrant un large choix :

- dans les supports d'installation.
- dans les outils disponibles pour créer le programme d'installation.
- dans les différents modes d'installation disponibles (installation monoposte ou réseau).

Ce chapitre décrit rapidement les différents types d'installation proposés par WINDEV et leurs spécificités.

Pour plus de détails sur la création du programme d'installation, consultez l'aide en ligne.

1 Installation : Tableau récapitulatif

1.1 Support d'installation

En fonction du support d'installation utilisé (Internet, Réseau interne, ...), WINDEV offre plusieurs modes de génération de l'installation.

Pour optimiser votre installation, choisissez le mode de génération conseillé.

		Modes de génération de l'installation proposés par WINDEV			
		Exécutable Auto-Extractible	**Répertoires à la taille de disquettes**	**Répertoire unique**	**CD ROM DVD ROM**
Supports d'installation	**Internet**	X (conseillé)			
	CD ROM / DVD ROM			X	**X** (conseillé) Autorun possible
	Réseau interne	X		**X** (conseillé)	X

1.2 Outils pour créer le programme d'installation

Différents outils sont disponibles pour créer le programme d'installation, le maintenir et le personnaliser.

	Assistant	**WDINST, éditeur d'installation**	**Personnalisation de WDSETUP**
Créer un programme d'installation standard (en français et anglais par défaut)	**X** (conseillé)	X	
Maintenir un programme d'installation	X	X	
Personnaliser un programme d'installation (le programme qui est lancé chez le client)		X	X (installation multi-lingue par exemple)
Mode de lancement	Sous le volet "Projet", dans le groupe "Général", cliquez sur "Procédure d'installation".	Sous le volet "Outils", dans le groupe "Utilitaires", cliquez sur "WDInst".	Projet WDSETUP disponible dans le sous-répertoire "Exemples" du répertoire d'installation de WINDEV.

1.3 Les différents types d'installation

Deux types d'installations sont disponibles dans WINDEV :

- **Installation individuelle** : conseillée pour les applications monopostes.
  Le programme d'installation installe l'application directement sur le poste utilisateur.
- **Installation avec mise à jour automatique** : conseillée pour les applications multi-postes. L'installation avec mise à jour automatique est réalisée en deux étapes :
  - L'application de référence est installée sur le serveur. L'application de référence correspond au programme d'installation de l'application sur les postes utilisateur.
  - L'application finale est installée sur chaque poste utilisateur, en exécutant le programme d'installation présent sur le serveur. Les mises à jour seront ensuite effectuées automatiquement lorsque l'application de référence évoluera sur le serveur.

2 Création du programme d'installation

Installation individuelle

Différents types d'installation sont possibles :

- ***Installation standard***
  Le programme d'installation est indépendant de l'application. Tous les fichiers nécessaires à l'application sont installés sur chacun des postes utilisateurs. Pour prendre en compte une mise à jour, cette mise à jour doit être exécutée sur chacun des postes utilisateurs.
- ***Installation compacte***
  Une installation standard est effectuée. Cependant, le programme d'installation est moins important en taille car il n'utilise pas de DLL spécifiques et certaines options de personnalisation ne sont pas disponibles.

- ***Installation par un package MSI***
  Une installation standard utilisant les fonctionnalités de l'installateur intégré de Windows est effectuée.

Installation avec mise à jour automatique

Différents types d'installation sont possibles :

- ***Installation en réseau local (LAN), avec prise en charge possible du "Push".*** Une première installation de l'application sera réalisée sur le réseau, dans un répertoire partagé. Ce répertoire contiendra le programme d'installation utilisé par les utilisateurs finaux et la version de référence permettant de gérer les mises à jour.
  Cette installation permet de :
  - gérer la mise à jour automatique de l'application installée en réseau.
  - connaître les caractéristiques des utilisateurs connectés à l'application à un moment donné.
  - paramétrer l'emplacement des fichiers nécessaires à l'application. Il est possible de gérer différents cas, par exemple :
    - Installation de tous les fichiers de l'application sur le serveur.
    - Installation de la base de données sur le serveur et des fichiers nécessaires à l'exécution de l'application (".EXE", ".WDL", ".DLL", ...) sur chaque poste utilisateur.
    - Installation d'une partie de la base de données sur le serveur et d'une autre partie sur chaque poste utilisateur. Les fichiers nécessaires à l'exécution de l'application (".EXE", ".WDL", ".DLL", ...) sont également installés sur chaque poste utilisateur.
  - Si la prise en charge du "Push" est utilisée, cette installation permet également de :
    - déployer une nouvelle application sur un réseau depuis un unique poste d'administration.
    - déployer une mise à jour de l'application sur un réseau depuis un unique poste d'administration.
- ***Installation par HTTP (sur un serveur propriétaire ou dans le CLOUD)***. Une première installation de l'application sera réalisée sur un serveur HTTP. Les utilisateurs pourront se connecter à une page Web pour télécharger le programme d'installation de l'application.
  Cette intallation permet notamment de :
  - gérer la mise à jour automatique de l'application installée sur un serveur HTTP. .
  - paramétrer l'emplacement des fichiers nécessaires à l'application. Il est possible de gérer différents cas, par exemple :
    - Installation de tous les fichiers de l'application sur le serveur.
    - Installation de la base de données sur le serveur et des fichiers nécessaires à l'exécution de l'application (".EXE", ".WDL", ".DLL", ...) sur chaque poste utilisateur.
    - Installation d'une partie de la base de données sur le serveur et d'une autre partie sur chaque poste utilisateur. Les fichiers nécessaires à l'exécution de l'application (".EXE", ".WDL", ".DLL", ...) sont également installés sur chaque poste utilisateur.
- ***Installation multisite***. Cette installation est la combinaison d'une installation réseau local et d'une installation HTTP. L'installation HTTP est utilisée pour la mise à jour de l'installation de référence sur le réseau local.

2.1 Créer un programme d'installation

Pour créer un programme d'installation permettant de :

- ***déployer une application WINDEV***, il est possible :
  - soit d'utiliser l'assistant de création du programme d'installation directement depuis WINDEV (sous le volet "Projet", dans le groupe "Génération", cliquez sur "Procédure d'installation"). Cet assistant permet de créer facilement les différents types de programme d'installation.
  - soit d'utiliser l'éditeur d'installation WDINST (sous le volet "Outils", dans le groupe "Utilitaires", cliquez sur "WDInst"). Cet éditeur permet de personnaliser entièrement le programme d'installation.
- ***fournir d'autres fichiers***, il est nécessaire d'utiliser l'assistant de création du programme d'installation lancé depuis l'éditeur d'installation WDINST (option "Fichier .. Nouveau" depuis WDINST).
  Rappel : Pour lancer l'éditeur d'installation WDINST, sous le volet "Outils", dans le groupe "Utilitaires", cliquez sur "WDInst".

2.2 Installer une application WINDEV

Installation standard, compacte ou MSI

Pour installer votre application, il suffit de lancer le programme d'installation de l'application sur le poste utilisateur. Les fichiers nécessaires à l'application s'installent dans le répertoire d'installation spécifié.

Pour installer une mise à jour de votre application, il suffit de lancer le programme d'installation de la mise à jour sur le poste utilisateur. Les fichiers de l'application modifiés par la mise à jour sont remplacés.

Installation avec mise à jour automatique

Pour installer votre application :
1. Lancez votre programme d'installation sur le poste serveur. Les fichiers nécessaires à l'application de référence s'installent dans le répertoire d'installation spécifié. L'application de référence correspond à un programme d'installation client.
2. Exécutez le programme d'installation client à partir de chaque poste utilisateur. Les fichiers nécessaires à l'application client sont installés dans le répertoire d'installation spécifié.

Pour installer une mise à jour de votre application :
1. Lancez votre mise à jour d'installation sur le poste serveur. Les fichiers de l'application de référence modifiés par la mise à jour sont remplacés. Il est possible de sauvegarder ces fichiers remplacés.
2. Lors du lancement de l'application depuis le poste d'un utilisateur, la mise à jour de l'application est automatiquement proposée.

Installation avec mise à jour automatique en mode PUSH

Pour installer votre application en mode PUSH :
1. Lancez votre programme d'installation sur le poste serveur. Les fichiers nécessaires à l'application de référence s'installent dans le répertoire d'installation spécifié. L'application de référence correspond à un programme d'installation client.
2. Le programme d'administration des applications réseau se lance automatiquement.
3. Sélectionnez le poste du réseau sur lequel l'installation doit être effectuée.

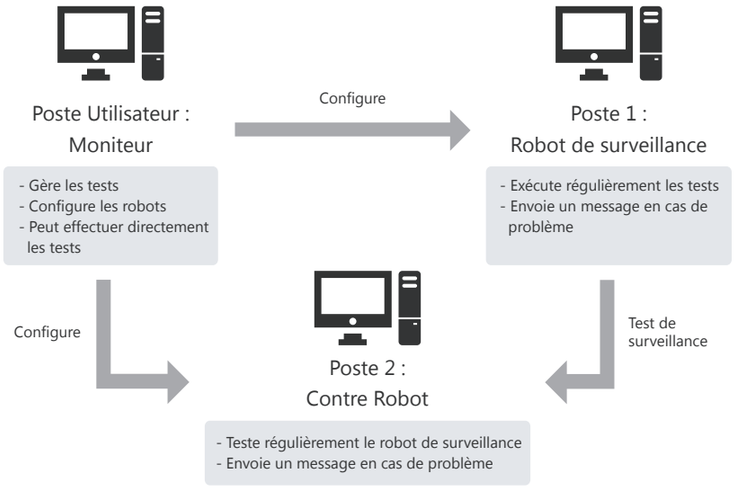
Pour installer une mise à jour de votre application en mode PUSH :
1. Lancez votre mise à jour d'installation sur le poste serveur. Les fichiers de l'application de référence modifiés par la mise à jour sont remplacés. Il est possible de sauvegarder ces fichiers remplacés.
2. Lancez le programme d'administration des applications réseau.
3. Sélectionnez le poste du réseau sur lequel la mise à jour doit être effectuée.

Surveillez vos applications

Une application dépend souvent de nombreux paramètres extérieurs : site Web, Serveur HFSQL, réseau, ... Pour optimiser la gestion des incidents, WINDEV propose d'utiliser un robot de surveillance.
Composé de trois exécutables lancés sur différents postes, le robot de surveillance permet d'exécuter différents tests : vérification de la connectivité Internet, tests réseau, ...
En cas de problèmes lors du passage d'un test, le robot de surveillance peut vous avertir de différentes façons :

- Message envoyé dans la messagerie PC SOFT (WDBAL).
- Email
- Exécution d'un programme tiers.

De plus, une alerte sonore peut être mise en place sur le moniteur.



Types de tests disponibles

PARTIE 6

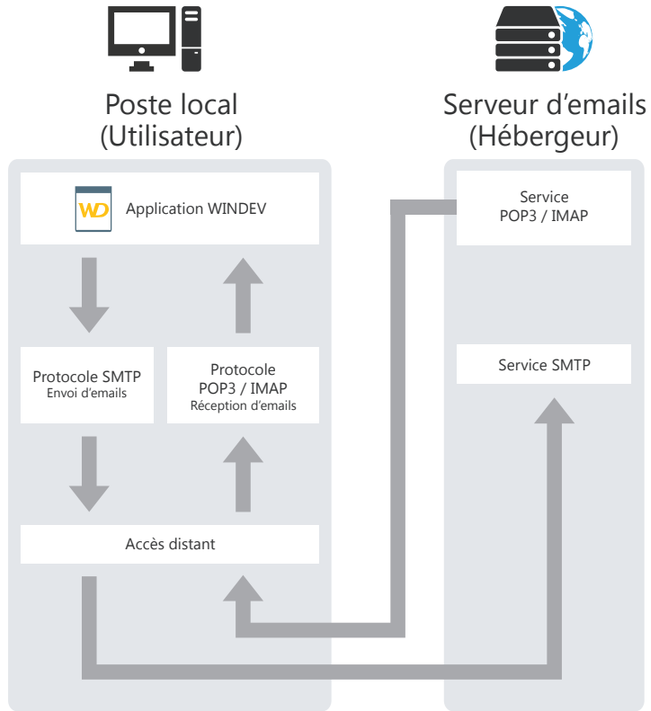
Communication

Gestion des emails

WINDEV permet de gérer les emails grâce à plusieurs méthodes :
- les protocoles POP3 ou IMAP et SMTP (méthode la plus courante)
- l'API "Simple Mail API (appelé aussi SMAPI ou Simple MAPI)"

Protocoles POP3 ou IMAP et SMTP

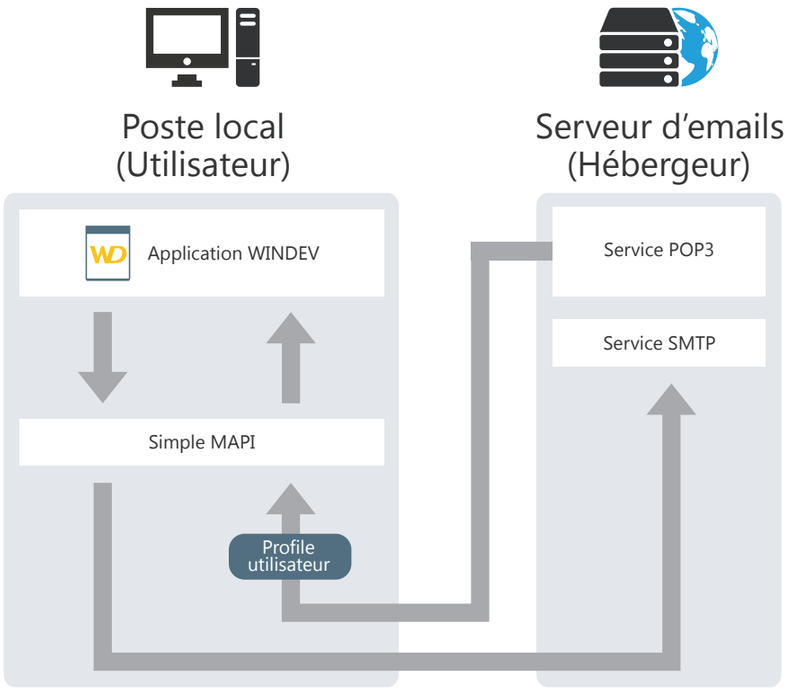
Ces protocoles sont les standards Internet pour la gestion des emails. Ils sont normalisés par les documents RFC 1939, 3501 et 5321. Ces protocoles vous permettent de dialoguer directement avec le serveur, disponible chez votre fournisseur d'accès.



- POP3 (Post Office Protocol) est un protocole de lecture d'emails.
- IMAP (Internet Message Access Protocol) est un protocole de lecture d'emails à distance.
- SMTP (Simple Mail Transfer Protocol) est un protocole d'envoi d'emails.

API "Simple Mail API (appelé aussi SMAPI ou Simple MAPI)"

Ce mode de gestion des emails est utilisé par la plupart des applications Microsoft et principalement MS Exchange.



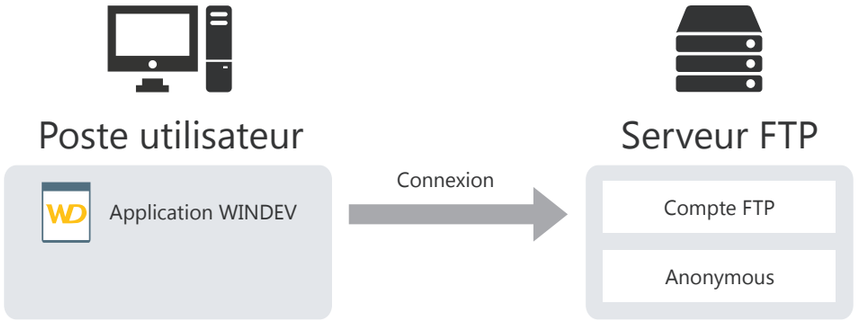
Transfert de fichiers par FTP

Le FTP (File Transfer Protocol) est un protocole de transfert de fichiers d'un site vers un autre site distant. Ce protocole permet d'échanger des fichiers par TCP/IP ou Internet.

Principe d'utilisation

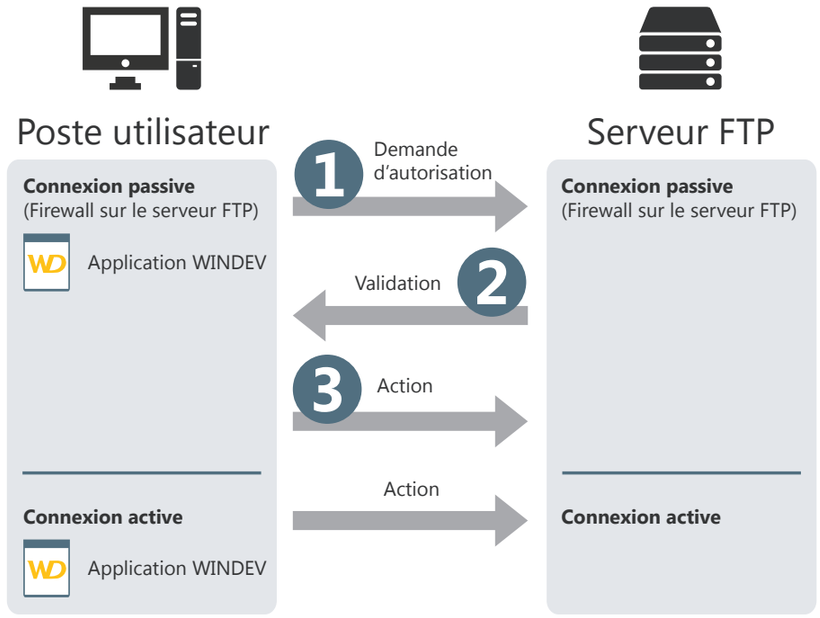
Pour manipuler des fichiers sur un serveur FTP depuis une application WINDEV, il est nécessaire de suivre les règles suivantes :

1. Connexion à un serveur FTP (fonction ***FTPConnecte***).

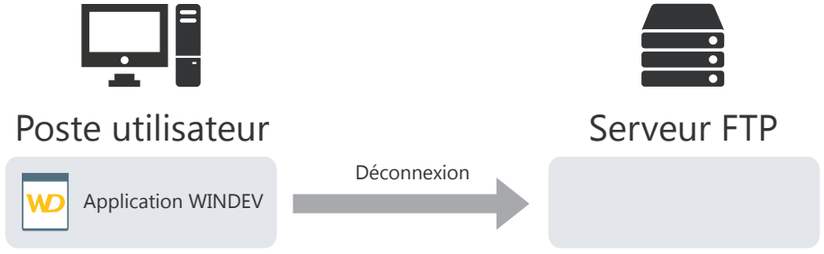


2. Transmission et récupération des fichiers (selon le mode de connexion établi). En cas de connexion passive, l'application WINDEV doit demander l'autorisation au serveur FTP avant chaque manipulation de fichiers sur le serveur. Il est également possible de :

- obtenir des caractéristiques sur des fichiers présents sur un serveur FTP : attributs, taille, ...
- manipuler des fichiers présents sur un serveur FTP : création, suppression, ...
- lister les fichiers d'un répertoire présent sur un serveur FTP en lançant une procédure effectuant un traitement pour chaque fichier trouvé.



3. Fermeture de la connexion avec le serveur (fonction ***FTPDéconnecte***).



Gestion des sockets

WINDEV met à votre disposition des fonctions qui permettent de réaliser une gestion avancée des sockets.
Un socket est une ressource de communication utilisée par les applications pour dialoguer d'une machine à une autre sans se soucier du type de réseau.
Ce mode de communication peut par exemple être utilisé pour établir une communication entre des postes reliés par Internet.

Différentes possibilités

Une application WINDEV peut gérer les sockets selon différents modes :

- ***Application WINDEV Cliente*** : l'application se connecte à un serveur quelconque et échange des données via un socket.
- ***Application WINDEV "Serveur simplifié"*** : l'application WINDEV est un serveur, échangeant des informations via un socket avec un seul poste client.
- ***Application WINDEV "Serveur standard"*** : l'application WINDEV est un serveur, échangeant des informations par sockets avec plusieurs postes clients quelconques.

Principe d'une application cliente

Une application cliente d'un serveur de sockets se connecte à un serveur quelconque pour échanger des informations par l'intermédiaire d'un socket.

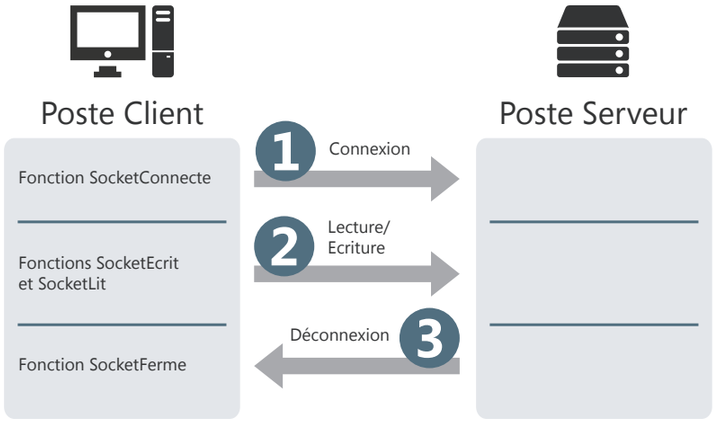
Exemple : Une application WINDEV cliente peut se connecter à un serveur de news standard sur Internet.

Etape 1 : Connexion au serveur
Pour se connecter à un socket du serveur, utilisez la fonction ***SocketConnecte***. Cette fonction permet d'effectuer une demande de connexion au serveur. Le socket est identifié par son port et par son adresse.

Etape 2 : Échange de données
Lorsque deux postes ont connecté leur socket, un flux de communication s'établit entre ces deux postes. Il est alors possible pour ces deux postes de lire et d'écrire des chaînes de caractères sur le socket.
Pour lire et écrire sur le socket du poste serveur, l'application cliente WINDEV doit utiliser les fonctions ***SocketLit*** et ***SocketEcrit***.

Etape 3 : Fin de la communication
Pour terminer la communication, il suffit de fermer le socket depuis le poste client avec la fonction ***SocketFerme***.

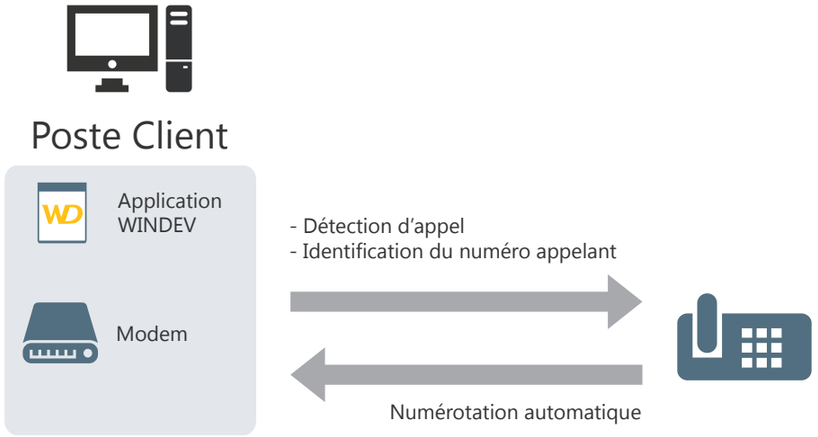
Les différentes étapes peuvent être représentées de la façon suivante :



Remarque : les fonctions SocketXXX permettent aussi de gérer des sockets sécurisés selon le protocole SSL. Il faut pour cela utiliser les fonctions ***SocketCréeSSL*** et ***SocketConnecteSSL***.

WINDEV et la téléphonie

WINDEV permet de gérer simplement les appels téléphoniques entrants et sortants grâce aux fonctions WLangage de téléphonie. Ces fonctions permettent par exemple de gérer une boîte vocale, un répondeur, ... directement dans une application WINDEV.



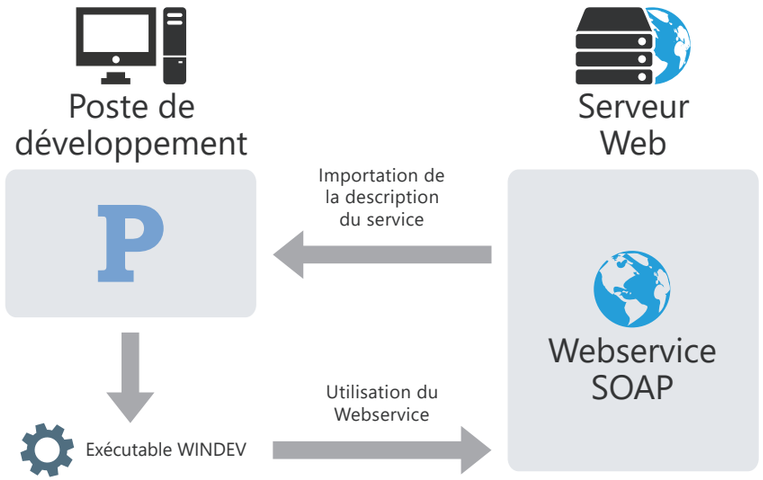
Configuration nécessaire :
Pour pouvoir utiliser les fonctionnalités de la téléphonie, il est nécessaire de posséder :

- un modem.
  Pour enregistrer et jouer des enregistrements, ce modem doit gérer la voix.
  Pour avoir les caractéristiques de la personne qui appelle, le modem doit posséder l'option "Notification du numéro appelant" (caller Id).
- le protocole Tapi 2.0.

Les Webservices

WINDEV vous permet d'utiliser très simplement des Webservices (ou services Web XML). Un Webservice est une application disponible sur le Web. WINDEV permet d'accéder à des Webservices SOAP (Simple Object Access Protocol) comme à des Webservices REST (Representational State Transfer).

Toute application WINDEV peut utiliser un Webservice existant.
Pour utiliser un Webservice SOAP, il suffit de connaître l'adresse du Webservice et d'importer sa description dans l'application.



Pour utiliser un Webservice REST, il suffit de connaître les caractéristiques du Webservice et d'utiliser la fonction ***RESTEnvoie***.

Remarque : il est également possible d'exposer une ou plusieurs collections de procédures d'une application WINDEV sous la forme d'un Webservice SOAP ou REST. Il faut pour cela :

- créer une configuration de projet dédiée à cet usage,
- dans le cas d'un Webservice REST, créer une description du Webservice REST,
- générer le Webservice,
- le déployer sur un Serveur d'Application WEBDEV.

PARTIE 7

Annexes

Composants livrés avec WINDEV

WINDEV est livré en standard avec projets correspondants à des composants. Ces projets contiennent :

- une configuration de projet correspondant au composant.
- une configuration de projet correspondant à l'utilisation du composant.
- un composant interne.

Voici la liste des principaux composants livrés avec WINDEV :

Nom du composant	Description
WD BGInfo	Permet d'écrire des informations sur le fond d'écran d'un ordinateur (adresse IP, Nom de la machine, Date du jour, etc.).
WD CodesPostaux-France	Permet la recherche et la sélection d'une ville sans modifier l'analyse.
WD Communication	Permet de faire communiquer deux applications distinctes. Ces deux applications peuvent être : - Soit sur des postes distants. La communication est effectuée via sockets. - Soit sur le même poste. La communication est effectuée grâce aux messages Windows.
WD Derniersdocuments	Permet de proposer une option de menu permettant d'ouvrir les derniers documents ouverts dans l'application.
WD Ecran Tactile	Permet l'interfaçage d'un clavier tactile pouvant remplacer le clavier standard. Ce clavier permet par exemple d'utiliser une application WINDEV sur un poste ne disposant pas de clavier.
WD Envoi Fax	Permet de gérer l'édition et l'envoi de fax (fonctionne uniquement sous Windows 2000 et Windows XP).
WD Explorer bar	Permet de gérer une barre d'adresse similaire à la barre d'adresse de l'explorateur de Windows Vista.
WD Extraction	Permet d'extraire et de stocker le contenu de divers documents (Open Office, Word 2007, ...) afin de les retrouver grâce à la recherche full-text.
WD Fenêtres furtives	Permet d'afficher des informations temporaires à l'utilisateur sans perturber son travail.

WD FileImpression	Détermine la liste des imprimantes les moins sollicitées pour effectuer un ou plusieurs travaux d'impression.
WD Gestion Logs	Permet de définir, pour une application, un paramétrage des logs d'exécution : - nombre de fichiers de log qui doivent être conservés, - taille maximale par fichier de log, - informations qui doivent être logguées. Cet exemple simule différents traitements pour observer la gestion des fichiers de logs (fichiers .wlog).
WD Glossaire	Permet de classer et de conserver des formulations écrites qu'un client réutilise régulièrement. Il est ainsi possible de proposer en permanence au client un accès à un "Glossaire" qu'il personnalise aisément. Toutes les entrées du glossaire peuvent être classées en catégories arborescentes.
WD Importation de pages HTML	Permet d'importer des pages HTML avec les fonctions WLangage.
WD InfoSysteme	Propose trois catégories de fonctions permettant de rassembler les informations sur le poste en cours, l'utilisateur et le réseau auquel est connecté le poste en cours.
WD Jauges	Permet de gérer automatiquement dans vos projets une jauge simple ou double qui affiche le temps restant ou le nombre d'étapes d'un traitement.
WD JournalEvenement	Permet de : - Lister et lire les événements enregistrés dans le journal des événements de Windows. - Ajouter un événement dans le journal des événements de Windows.
WD Limitation	Permet de limiter une application par différents moyens : - Limitation de durée (Application limitée à 30 jours) - Limitation à 1 poste (Application n'a le droit de s'exécuter que sur un seul poste) - Limitation du nombre d'utilisateurs (Une application ne peut pas être exécutée par plus de n utilisateurs)
WD ModèleImpression	Permet de gérer des profils d'impression.

WD Paramètres Impressions	Permet d'utiliser la fonction iParamètre. Cette fonction permet de récupérer les différents paramètres définis pour une imprimante donnée. L'imprimante utilisée est définie grâce à la fonction iConfigure(Nom de l'imprimante). Ce composant permet de modifier différents paramètres : - L'orientation du papier - L'échelle - La gestion des polices TrueType - Le nombre de copies - La gestion de la couleur (impression couleur ou monochrome) - Le recto-verso - Le style de sortie des feuilles (assemblées par copies ou par numéro de page)
WD Pays du monde	Contient la liste de tous les pays du monde et certaines informations : - code ISO de chaque pays, - liste des régions, états ou pays constituant un pays (avec le décalage horaire pour chaque état, région ou pays), - décalage horaire (GMT), - monnaie principale du pays, - principales langues parlées, - drapeau de chaque pays, - capitale, - indicatif téléphonique.
WD Photocopieur	Permet de réaliser des photocopies de documents à l'aide d'un scanner et d'une imprimante.
WD Popup Graphique	Dans de nombreux logiciels de retouche d'images, il est possible de sélectionner une forme à dessiner, de choisir une épaisseur pour cette forme, et ceci via une combo popup graphique avancée. Ce type de combo popup graphique offre en effet une prévisualisation en temps réel de la forme à l'épaisseur choisie. Ce composant permet d'intégrer une telle interface dans une application WINDEV.
WD Saisie email	Permet de faciliter la saisie des emails.

WD Select RTF	Permet d'effectuer tous types de recherches dans un champ RTF. Vous pouvez par exemple rechercher : - les mots en caractères gras, - les mots en italique, - les mots d'une couleur spécifique, - une chaîne de caractères ayant des mots spécifiques.
WD Signature	Permet d'intégrer facilement dans vos applications WINDEV un champ 'Signature' prêt à l'emploi. Ce champ est notamment très pratique pour les applications qui sont destinées à des terminaux tactiles, TabletPC sous Windows XP, bornes interactives, ...
WD Simplexe	Permet de réaliser des calculs à partir de l'algorithme du Simplexe.
WD Trace Socket	Permet de tracer tout ce qui passe sur une communication via socket, sur une adresse donnée, sur un port donné. Cela permet de comprendre ce qu'il se passe dans un dialogue via socket entre deux applications. Ainsi, dans des applications qui dialoguent via socket, même si les développeurs n'ont pas prévu d'option de trace, il sera possible de voir l'ensemble des échanges effectués sur la socket. Cet exemple peut également permettre de mieux comprendre le fonctionnement des protocoles réseau basés sur les sockets : POP3, SMTP, FTP, HTTP, etc.
WD Sockets	Permet de faire dialoguer facilement deux applications en utilisant des sockets.

Exemples livrés avec WINDEV

Les exemples livrés avec WINDEV présentent de manière didactique les différentes fonctionnalités de WINDEV.
Leur code source est commenté et détaillé.
Ces exemples sont présents dans le sous-répertoire "Exemples" du répertoire d'installation de WINDEV et peuvent être ouverts directement depuis la fenêtre d'accueil de WINDEV.

Différents types d'exemples sont livrés avec WINDEV :

- Exemples complets : ces exemples correspondent à des applications complètes, qui peuvent être utilisées sans adaptation.
- Exemples didactiques : ces exemples illustrent une fonctionnalité spécifique.

Voici les fonctionnalités de quelques exemples livrés avec WINDEV.

Exemples complets

CRM	Permet de gérer des clients, d'établir des commandes et des devis, et de gérer une liste de tâches et un planning.
eCommerce (Back office)	Cet exemple est une application de back-office pour le site eCommerce. Il utilise le webservice eCommerce afin de : - gérer les produits - gérer les commandes - modérer les avis - ...
WD Agenda	Permet de synchroniser des rendez-vous entre vos agendas Outlook, Lotus Notes et Google.
WD Congés et RTT	Gestion complète des congés et des RTT des collaborateurs d'une société. Chaque utilisateur dispose d'un tableau de bord de l'état de ses congés et RTT.

WD Emprunt	Permet de simuler des emprunts et notamment de : - calculer le montant des mensualités à partir du montant emprunté donné - calculer le montant empruntable à partir d'une mensualité donnée - calculer le rapport d'un placement à partir d'un apport mensuel Pour chaque cas, il est possible d'imprimer le tableau d'amortissement correspondant à vos paramètres.
WD Gantt	Présente une utilisation avancée d'un diagramme de Gantt. Permet de créer un diagramme de Gantt complet avec tâches, liaisons, tâches cumulatives et jalons. Les tâches peuvent être éditées dans une fenêtre spécifique. Le diagramme créé peut être sauvegardé dans un fichier pour être rechargé par la suite.
WD Gestion Commerciale	Présente une application de gestion commerciale, "tendance", sous forme d'onglets dynamiques, avec ruban.
WD Gestion Contacts	Permet de gérer des contacts et de leur envoyer des emails. Il est également possible de synchroniser ces contacts avec ceux présents sur le terminal mobile relié au PC et avec ceux de Outlook.
WD Gestion de parc informatique	Permet de gérer un parc informatique. Pour chaque poste du parc, ses caractéristiques, ses composants et ses utilisateurs peuvent être définis. Ces fonctionnalités sont mises en œuvre grâce à l'utilisation de requêtes, d'états et de combos.
WD Gestionnaire de tâches DOTNET	Permet de lister les applications et les processus grâce à des fonctions .NET. Permet d'effectuer différents traitements sur les applications et sur les processus.
WD Graveur	Permet de graver des CD Audio et des CD de données, DVD ou disques Blue-Ray.
WD Graveur Lite	Permet de graver des CD de données, DVD ou disques Blue-Ray.
WD Lecteur RSS	Présente l'utilisation de classes permettant de lire des flux RSS

WD Loupe	Permet de zoomer une partie de l'écran grâce à une loupe visuelle. Il est possible de zoomer jusqu'à 8 fois.
WD Mail	Cette application est un client mail complet. Il s'appuie sur les objets Email. Ce client mail permet de récupérer et d'envoyer des emails en utilisant les protocoles POP, IMAP et SMTP.
WD Messagerie instantanée	Présente les différentes fonctions de gestion des sockets.
WD MotDePasse	Présente un "coffre" à mot de passe. Peut être utilisé pour gérer les mots de passe créés lors de l'utilisation de sites Internet (ou d'applications) ou des mots de passe de la vie courante (digicodes, codes de cartes bancaires, etc.) . Peut également être utilisé comme un générateur de mot de passe pour produire des mots de passe particulièrement sûrs.
WD Multimédia	Permet de lire et de jouer : - des vidéos (AVI, MPEG...), - des fichiers sons (MP3, WAV), - des CD Audio, - des animations Flash.
WD Outlook	Permet de lire et écrire dans les dossiers de Outlook : - Messagerie, - Calendrier, - Contacts, - Tâches. Cet exemple est conçu pour fonctionner avec la version standard de Outlook livrée avec Office.
WD Planning	Permet de réaliser un planning comportant des zones clicables.
WD Proxy HTTP	Cet exemple est un proxy HTTP simplifié. Il est possible d'enregistrer les connexions et de connaître les pages consultées.
WD ProxyPOP3	Cet exemple présente un proxy POP3. Dans cet exemple, le proxy est utilisé pour archiver automatiquement dans une base HFSQL les emails récupérés.

WD Recherche sur Internet	Permet de piloter différents moteurs de recherche sur Internet (Yahoo, Alta Vista, Voilà, Lycos, Excite, Nomade, Google, Euroseek, etc.). Le résultat de la recherche est affiché grâce à un ActiveX.
WD Répondeur Téléphonique	Utilise les fonctionnalités de téléphonie pour la gestion d'un répondeur téléphonique. Ce répondeur est interrogeable et paramétrable à distance grâce à un menu.
WD Scanner - Twain	Permet de : - Prévisualiser une partie d'un document, - Numériser un document (format GIF, JPEG, BMP), - Numériser un document et le placer dans le presse-papiers, - Numériser un document et l'imprimer.
WD Synchronisation de fichiers	Permet de synchroniser deux répertoires de fichiers. Après synchronisation, le répertoire de destination est identique au répertoire source. La comparaison est effectuée sur la présence et la date des fichiers.
WD Téléphonie	Utilise les fonctions téléphonie de WINDEV pour : - Composer un numéro de téléphone, - Détecter et identifier des appels entrants.
WD Télésurveillance	Permet de transmettre des images filmées par une Web Caméra à travers un réseau (Internet, Intranet, etc.).
WD Transfert de fichiers par FTP	Permet de transférer des fichiers : - d'un serveur FTP vers le poste local, - du poste local vers un serveur FTP.
WD Visioconférence	Permet d'entrer en conférence avec un utilisateur distant. Permet d'envoyer un fichier à un correspondant et de dialoguer avec lui.
WD WebCam	Présente l'intégration d'une Web Caméra dans une fenêtre WINDEV. Il est possible de : - Visualiser l'image retransmise par la caméra, - Effectuer une capture instantanée, - Capturer la scène retransmise sous forme d'animation Windows (Fichier "AVI").

WD Zip	Permet de créer et de lire des archives compressées au format "WDZ" (ZIP WINDEV).

Exemples didactiques

WD Active Directory DotNet	Permet la visualisation du contenu d'un Active Directory. Cette visualisation est effectuée grâce à des objets .NET contenus dans l'assembly System.DirectoryServices.
WD Analyseur Explain HFSQL	Représente visuellement le détail du plan d'exécution d'une requête SQL grâce à la commande SQL EXPLAIN.
WD APISystemes	Montre l'utilisation des Apis Windows.
WD Appels JAVA	Illustre l'appel aux méthodes des classes d'une archive JAR grâce à la fonction WLangage JavaExécuteFonction. Montre également le remplissage d'une table mémoire (identique en mode Java et en mode Windows).
WD Application Interactive	Montre l'utilisation des fonctions FenEdXXX dans une application interactive. Cette application permet de gérer une maison connectée (allumer les lumières, connaître la température).
WD Assistant	Gère une fenêtre de type "Assistant".
WD Bluetooth	Cet exemple montre l'utilisation des fonctions Bluetooth : - pour lister les périphériques Bluetooth accessibles, - pour lister les services offerts par un périphérique, - pour se connecter à un périphérique Bluetooth, - pour envoyer un fichier à un périphérique via les fonctions "OBEX" du WLangage.
WD Champ Recherche	Cet exemple montre comment gérer un champ de saisie permettant de réaliser des recherches Full-Text de type "google" dans votre base de données.
WD Changements Gabarits	Cet exemple présente une solution de changement dynamique de gabarits par les utilisateurs.

WD Coloration Recherche	Cet exemple présente comment mettre en évidence les mots recherchés dans les résultats d'une recherche Full-Text.
WD Coloration Syntaxique	Cet exemple propose une classe de base permettant d'effectuer une coloration syntaxique classique sur du texte.
WD ContrôleADistance	Permet de mettre en application le contrôle à distance d'une application WINDEV.
WD Contrôles de saisie	Effectue la vérification des données saisies par l'utilisateur (nom, société, code postal, ville, adresse e-mail, etc.) dans une fenêtre de type formulaire.
WD DataBinding	L'exemple "WD DataBinding" présente comment lier un champ à une variable du WLangage.
WD Découverte MVP - Partie 1	Cet exemple didactique présente la mise en oeuvre d'une architecture MVP (Modèle Vue Présentation) dans un projet.
WD Découverte MVP - Partie 2	Cet exemple didactique présente la mise en oeuvre d'une architecture MVP (Modèle Vue Présentation) dans un projet pour une analyse simple (Client - Commande - Ligne de commande - Produit).
WD Délégué DOTNET	Cet exemple didactique montre comment interagir avec les délégués DOTNET
WD Dessin sur Graphe	Cet exemple repose sur l'utilisation d'un composant interne permettant de dessiner des éléments sur un champ graphe : Zones min/max, Zones diverses, Drapeaux, Plots et trackers de suivi,...
WD Detection Erreurs HFSQL	Présente le module de gestion assistée des erreurs HFSQL : - le fonctionnement par défaut, - les différentes possibilités de personnalisation.
WD DirectX	Cet exemple permet d'utiliser DirectX 9.0 dans vos applications WINDEV.
WD Duplication	Cet exemple didactique montre comment il est possible de dupliquer une base de données sur un serveur.

WD Ecran de veille	Permet de réaliser un économiseur d'écran grâce à la gestion des timers et des événements Windows.
WD Etats	Cet exemple montre les différentes méthodes pour réaliser un état : - impressions basées sur différentes sources de données (requêtes, variables, ...) - impressions basées sur des champs (Table, Tableur, TCD, ...) - impression d'états composés - impressions spécifiques (portrait / paysage, état avec filigrane, état avec code-barres, ...)
WD Exemple de test unitaire	Cet exemple présente l'utilisation de l'éditeur de test unitaire intégré de WINDEV dans deux cas concrets : - le test d'une fenêtre - le test d'une procédure globale
WD Export HTML	Gère des fichiers externes. Présente également la manipulation des fichiers au format HTML.
WD Extraction MIME	Permet d'extraire les pièces contenues dans un email.
WD Fichier externe	Présente comment manipuler des fichiers non décrits dans l'analyse.
WD Génération PowerPoint	Cet exemple présente comment générer une présentation PowerPoint à partir : - de données contenues dans une application WINDEV, telles que les données de tables, de graphes .. - et d'un document PowerPoint .pptx de base.
WD Gestion Auto Erreurs	Cet exemple est un exemple didactique sur la gestion automatique des erreurs. Cet exemple montre le cas des erreurs fatales, non fatales, sur plusieurs niveaux.
WD Gestion des erreurs Java	Cet exemple didactique permet de comprendre le fonctionnement et la mise en œuvre du mécanisme de gestion des erreurs de WINDEV.
WD Gestion Erreurs HFSQL	Cet exemple présente de façon didactique la gestion des erreurs HFSQL (doublon, intégrité). Cet exemple montre comment personnaliser le traitement lorsqu'une erreur est levée.

WD Graphe	Cet exemple didactique présente diverses utilisations du champ graphe. Tous les types de graphe disponibles sont présentés.
WD Grapheur Java	Cet exemple montre comment réaliser des graphes grâce à aux fonctions de dessin. Trois types de graphe sont présentés : histogramme, nuage de points, courbe.
WD Groupware utilisateur	Présente le fonctionnement du Groupware Utilisateur.
WD GroupwareCS	Présente le groupware utilisateur avec une base de données HFSQL Client/Serveur.
WD Hotkey	Permet de définir vos propres raccourcis clavier.
WD ImpRectoVerso	Cet exemple présente une solution d'impression recto-verso qu'il est possible de désactiver, c'est-à-dire que les versos peuvent ne pas être imprimés.
WD Impression de Tree View	Permet d'imprimer un Treeview. Cet exemple utilise une classe "TreeImp" facilement réutilisable pour n'importe quel champ Treeview.
WD JavaMail	Permet de créer une applet, utilisable dans une page Internet. Cette applet permet de lire et envoyer des emails.
WD Jump Lists	JumpList sous Windows 7 Une JumpList est un menu accessible via le menu Démarrer ou l'icône de l'application dans la barre des tâches.
WD Langage Externe	Cet exemple illustre l'utilisation de fenêtres faites avec WINDEV dans une application en C ++ (Visual Studio 2005) et MFC.
WD Lasso	Utilise un lasso de sélection dans un champ Image pour réaliser un zoom dans un champ de prévisualisation.
WD Lien réflexif	Utilise un lien "réflexif" pour gérer des arbres généalogiques : chaque personne a plusieurs parents et plusieurs enfants. Tous ces enregistrements sont stockés dans le même fichier.

WD Limitation de durée d'utilisation	Permet de limiter l'utilisation d'une application à une période donnée (période d'essai).
WD ListeChaînée	Fournit les outils pour construire des listes chaînées à chaînage simple. Une liste à chaînage simple permet de stocker et parcourir de manière ordonnée une collection de valeurs quelconques.
WD Mailing par emails	Permet d'envoyer un "mailing" par Internet.
WD Manipulation du champ Agenda	Exemple simple d'utilisation du champ Agenda. Les fonctionnalités utilisées sont les suivantes : Création d'un rendez-vous, Sauvegarde d'un agenda, Restauration d'un agenda
WD Menu Animé	Cet exemple propose différentes animations de menu.
WD Menu Pure	Cet exemple propose un menu personnalisable au travers d'un modèle de champ.
WD Multi fenêtrage	Permet de comprendre les bases de la gestion du Multi-fenêtrage avec WINDEV.
WD Multitouch	Cet exemple utilise un composant interne permettant d'utiliser les fonctionnalités Windows Touch dans une application WINDEV.
WD OAuth	Cet exemple propose didactiquement de se connecter à Google et Microsoft grâce à la fonction Auth-Identifie et au type OAuth2Paramètre.
WD Objets graphiques	Cet exemple est un éditeur d'objets graphiques. Il est possible de : - déplacer, redimensionner les objets - effectuer des sélections au lasso ou à la souris - couper/copier/coller des objets - supprimer/dupliquer des objets - enregistrer les documents créés et de les imprimer.
WD Observateur Répertoire	Cet exemple montre l'utilisation des delegates .NET (événements .NET) en WLangage.
WD OLE DB	Permet d'accéder à des bases de données externes via OLE DB.

WD OPC	Montre l'implémentation du protocole OPC en WINDEV.
WD OpenGL	Permet d'interfacer OpenGL et WINDEV. OpenGL est une interface logicielle destinée à dessiner des objets tridimensionnels (3D).
WD Organigramme	Cet exemple permet de créer des organigrammes.
WD OSD	OSD signifie : On-Screen Display. C'est le nom donné aux interfaces qui apparaissent sur les écrans d'ordinateur ou des téléviseurs par exemple. Ils servent souvent à effectuer des réglages. Ces types de menus ont comme particularité d'être affichés par-dessus tout ce qui se trouve à l'écran. Cet exemple montre comment réaliser ce type d'interface sous WINDEV.
WD Ouverture Périphérique	Cet exemple permet d'ajouter ou supprimer une application du système d'ouverture automatique pour les périphériques amovibles par programmation.
WD PartageMemoire	Cet exemple montre comment utiliser les fonctions de partage de mémoire.
WD PC Infos	Permet de connaître la langue du poste sur lequel l'application s'exécute.
WD Persistance	Cet exemple montre l'utilisation des fonctions InitParamètre, ChargeParamètre, SauveParamètre. Ces fonctions permettent de configurer la sauvegarde de champs, de variables et de tout autre paramètre.
WD Pilotage de Excel	Permet de piloter Excel via OLE Automation. Toutes les fonctions d'Excel peuvent ainsi être utilisées.
WD Pilotage de IE	Permet de piloter IE via OLE Automation. Toutes les fonctions de IE peuvent ainsi être utilisées.
WD Pilotage de Open Office Base	Cet exemple propose une classe permettant de manipuler facilement une base Open Office sans avoir à aucun moment à manipuler des objets OLE.
WD Pilotage de OpenOffice Calc	Cet exemple didactique illustre le pilotage du tableur OpenOffice.org. Il est en effet possible de créer, ouvrir et manipuler des classeurs ouverts dans OpenOffice.org depuis une application WINDEV.

WD Pilotage de OpenOffice Writer	Cet exemple didactique illustre le pilotage de OpenOffice Writer. Il est en effet possible de créer, ouvrir et manipuler des documents ouverts dans OpenOffice Writer depuis une application WINDEV.
WD Pilotage de Outlook	Permet de piloter Outlook via OLE Automation. Toutes les fonctions de Outlook peuvent ainsi être utilisées.
WD Pilotage de Tableurs	Cet exemple illustre le pilotage du tableur OpenOffice Calc et du tableur Excel, avec un code générique.
WD Pilotage de Word	Permet de piloter Word via OLE Automation. Toutes les fonctions de Word peuvent ainsi être utilisées.
WD Pilotage traitement de texte	Le but de cet exemple est de proposer une interface universelle, permettant de piloter ces applications, sans se soucier de l'application à piloter.
WD PlanningAvecRupture	L'exemple "WD PlanningAvecRupture" propose une solution pour gérer des "ruptures" dans un champ Planning
WD Plugin WDSETUP	Ce projet contient les éléments nécessaires à la réalisation d'un plugin (ou "extension") d'installation pour WDSETUP.
WD POO Simple	Exemple sur la POO avec WINDEV.
WD Presse-Papiers Multiple	Cet exemple gère automatiquement un presse-papiers multiniveaux entre toutes les applications WINDEV lancées sur le poste.
WD Pyramide Des Ages	Montre comment créer un graphique de type "Pyramide des âges" à partir de deux histogrammes.
WD Qui bloque	Signale aux utilisateurs d'une base de données HFSQL en réseau quelle est la personne qui bloque un enregistrement inaccessible.
WD Recherche par clé composée	Présente l'utilisation des filtres et l'utilisation des clés composées afin d'optimiser les temps de recherche sur un fichier de données HFSQL.

WD RechercheRuban	L'exemple "WD RechercheRuban" propose une solution pour rechercher un élément d'un champ Ruban à partir de son libellé et de sa bulle d'aide
WD Réplication Universelle	Montre comment synchroniser les données de différents sites en utilisant la réplication universelle.
WD Requêtes	Permet d'effectuer des calculs, de passer des paramètres ou d'imprimer le résultat d'une requête SQL sur un fichier de données HFSQL.
WD Riak	WD Riak présente un exemple d'utilisation de la base de données Riak en WLangage.
WD Service	Cet exemple illustre la création d'une application de type "service" avec WINDEV.
WD Sharepoint	Cet exemple détaille la méthode d'authentification auprès du Webservice ainsi que l'utilisation de la recherche.
WD Tableau de bord	L'exemple "WD Tableau de bord" est un exemple didactique sur l'utilisation du champ Tableau de bord.
WD TableauCroiséDynamique	Cet exemple illustre l'utilisation du champ Tableau croisé dynamique. Ce champ tableau croisé dynamique permet d'afficher des statistiques dynamiques en effectuant des regroupements. Ces statistiques sont calculées à partir de la base de données.
WD Text-To-Speech	Cet exemple montre comment effectuer une synthèse vocale dans une application WINDEV en utilisant la classe SpeechSynthesizer du framework DOTNET.
WD Timeline	Propose un composant interne permettant de manipuler un modèle de champ 'Timeline'. Une timeline permet d'afficher de façon linéaire une série d'événements qui se sont déroulés sur une période donnée de temps.
WD Trames réseau	L'analyse des trames circulant sur le réseau permet aux administrateurs d'auditer ce dernier et de détecter certaines incohérences. Pour écouter le réseau, l'application se base sur l'API Winsock 2 et les RAW Socket.

WD Transaction	Illustre le fonctionnement des transactions lors du passage d'une commande.
WD Treemap	Utilise le champ TreeMap afin de visualiser le contenu d'un répertoire et la place prise par chaque fichier constituant ce répertoire.
WD Trigger	Permet de saisir les notes de frais des collaborateurs d'une société. Deux modes sont utilisables : - Consultation avec possibilité de modification (lecture/écriture) - Consultation uniquement Ce sont les triggers qui permettent de contrôler l'accès aux fichiers en fonction du mode choisi.
WD Trigger serveur	Cet exemple permet de montrer comment utiliser des triggers serveur.
WD Utilisation de DOTNET	Permet de convertir des devises dans d'autres devises. La base de ce projet (la classe cEuro) est utilisée pour créer un assemblage DotNet.
WD Utilisation des classes DotNet	Permet d'utiliser des classes DOTNET en WLangage. Permet de : - analyser une chaîne de caractères, - configurer les boîtes de dialogue affichées.
WD Utilisation des sockets	Présente l'utilisation des sockets en mode client/serveur. Chaque socket est manipulée au travers d'un thread.
WD Utilisation de Webservices	Projet client : Cet exemple montre comment récupérer, à partir d'un Webservice, des images en fonction de mots-clés. Ce projet est le client qui se connecte au WebService "WD Serveur Webservice". Projet serveur : Cet exemple montre comment créer un Webservice fournissant des images en fonction de mots clés. Ce projet est le Webservice qui est utilisé par le projet WD Client Webservice.
WD WindowsUpdate	Cet exemple montre comment gérer les mises à jour disponibles sur Windows Update.

WD WMI DOTNET	Le WMI (Windows Management Instrumentation) permet une administration exhaustive des applications et des périphériques. Le WMI peut être comparé à une base de données regroupant de multiples informations sur un poste

Des exemples supplémentaires seront livrés avec la Lettre du Support Technique (LST) ou seront disponibles sur notre site (www.pcsoft.fr).