TDF Tech 2016
Support de cours

Dossier spécial TDF TECH 2016

Le document que vous avez entre les mains n'a pas la finalité de remplacer les documentations techniques ou commerciales livrées avec WINDEV, WEBDEV ou WINDEV Mobile.

Ce document n'est pas un cours d'auto-formation, mais uniquement un support "papier" de la présentation technique que vous avez suivie.

Ce document a été réalisé par l'équipe de la LST.

TDF TECH 2016 SOMMAIRE

Retrouvez dans ce support de cours les sujets abordés lors du TDF Tech 2016. Le code source des différentes applications et projets indiqués dans ce support de cours est présent sur le DVD du TDF Tech 2016.

4 Sommaire

8 PC SOFT à votre service

10 WINDEV
Présentation rapide de WINDEV et de ses fonctionnalités

12 WEBDEV
Présentation rapide de WEBDEV et de ses fonctionnalités

14 WINDEV Mobile
Présentation rapide de WINDEV Mobile et de ses fonctionnalités

18 Environnement : 10 nouveautés utiles
En version 21, les éditeurs de WINDEV, WEBDEV et WINDEV Mobile disposent de nouvelles fonctionnalités pratiques.
Voici 10 fonctionnalités, nouvelles ou non, que vous devez connaître !

20 Le GDS : l'outil indispensable
Le GDS est un gestionnaire de sources élaboré qui permet de sauvegarder les sources, les historiques, les versions,...
Primordial lors du développement en équipe, le GDS s'avère également très utile si vous travaillez seul.
Voici 5 avantages à utiliser le GDS !

22 Le Centre de Suivi de Projets
Le Centre de Suivi de Projets permet de gérer les tâches, les incidents, les exigences, les retours clients...
Cet outil offre de nombreuses fonctionnalités utiles au quotidien.
Cet article présente 3 fonctionnalités immédiatement utiles.

24 Éditeur d'états : 5 nouveautés utiles

Dans la majorité des applications, les éditions ont une place importante : facture, formulaire, plaquette...
En version 21, l'éditeur d'états évolue et vous propose de nouvelles fonctionnalités utiles au quotidien dans vos applications.

26 HFSQL : 12 conseils pour optimiser vos accès

HFSQL est la base de données disponible en standard sur WINDEV, WEBDEV et WINDEV Mobile.
Voici 12 conseils efficaces pour optimiser vos accès à une base HFSQL.

30 Générateur de gabarits : votre charte graphique

Le générateur de gabarits vous permet de créer votre propre charte graphique aux couleurs de votre entreprise ou de celles de vos clients.
Cette fonctionnalité est particulièrement utile lors du développement de nombreux projets pour unifier la charte graphique des applications.

32 21 fonctions WLangage utiles au quotidien

Le WLangage est un langage riche qui dispose de nombreuses fonctions utiles au quotidien.
Le WLangage vous permet de développer plus vite, en ayant un code plus compact.
Voici une sélection de fonctions, nouvelles ou méconnues, qui vous seront bientôt indispensables.

36 PCSCloud : vos solutions Cloud

PCSCloud propose des plateformes CLOUD prêtes à l'emploi pour répondre à vos différents besoins : partager une base de données HFSQL, fournir des sites et des webservices, fournir des installations d'applications WINDEV, partager un GDS et des Centres de Contrôle, héberger un serveur de télémétrie.

38 Télémétrie : des statistiques précises sur vos applications

À partir de la version 21, WINDEV propose des statistiques avancées sur l'utilisation de vos applications WINDEV grâce à la télémétrie.
Avec la télémétrie, vous connaissez tout de vos applications déployées et vous pouvez facilement les améliorer.

TDF TECH 2016 SOMMAIRE (SUITE)

42 Le champ Fil : connectez vos champs

Mettre en place des IHM intuitives et dynamiques pour construire des workflow, des plans ou des diagrammes, c'est très facile grâce au champ Fil.
Le champ Fil vous permet de mettre en place un lien visuel entre 2 champs.

44 PIU (Personnalisation de l'interface par l'utilisateur)

Quoi de plus frustrant pour un utilisateur qu'une IHM mal adaptée : une zone de texte trop petite, des boutons indispensables non mis en avant, ...
En version 21, l'utilisateur final a la possibilité de personnaliser complètement l'IHM directement depuis l'exécutable, en quelques clics.
Et bien entendu, le développeur peut récupérer ces modifications pour les mettre en place dans une prochaine mise à jour.

46 Le champ Note repositionnable

Le champ Note repositionnable permet d'intégrer dans vos applications un pense-bête toujours visible.
Ce champ est très pratique pour avoir toujours à portée, dans l'application, un «mémento» sur un champ, un client...

48 RAD Onglet dynamique

En version 21, un nouveau type de RAD fait son apparition : il s'agit du RAD «Onglet dynamique» (ou RAD «Ruban»).
La principale caractéristique de ce RAD est qu'il génère des fenêtres internes utilisées dans des onglets dynamiques.
La navigation est effectuée à l'aide d'un ruban, automatiquement généré.
Ce RAD est un «RAD Fenêtre».

50 Le champ Tableau croisé dynamique (Cube ROLAP)

Le champ Tableau croisé dynamique est un champ indispensable pour vos décisionnaires.
Il présente de manière synthétique toutes les statistiques nécessaires à la prise de décisions.
Avec WINDEV, en quelques clics, vous créez un champ Tableau Croisé dynamique complet et performant.

52 Graphes : les astuces de Marc pour faire des graphes au top

WINDEV permet de créer facilement et en standard des graphes élaborés sur vos données : courbe, aire, secteur, beignet, histogramme, etc.
Voici quelques nouveautés de la version 21 sur les graphes ainsi que quelques astuces pour améliorer le rendu.

54 Responsive Web Design : méthodologie

En partant d'un existant ou d'une nouvelle page, une bonne méthode permet de réaliser rapidement une page Responsive Web Design fonctionnant dans les différents types de périphériques : mobile, tablette ou bureau (desktop).

59 Le Dynamic Serving

Le «Dynamic serving» permet d'accéder à deux pages différentes (une page pour PC, une page pour mobile) à partir d'une même URL.
C'est une solution alternative (voire complémentaire) au «Responsive Web Design» pour rendre un site traditionnel «mobile friendly».
Avec le «Dynamic serving», il n'est pas nécessaire de modifier les pages existantes : il suffit d'en rajouter pour les mobiles.

60 Nouveautés mobiles

WINDEV Mobile vous permet de développer pour tous les terminaux mobiles : tablettes et téléphones, sous iOS, Android, Universal Windows 10 App, ...
La version 21 apporte son lot de nouveautés bien pratiques.
En voici quelques-unes.

62 Universal Windows 10 App

WINDEV Mobile vous permet de créer, à partir du même code source, des applications pour iOS, Android et Universal Windows 10 App (nouveau en version 21).

64 De WINDEV à WINDEV Mobile

Passer d'une application Windows à une application mobile (Android, iOS...), c'est possible !
Il suffit de bien suivre les différentes étapes et de savoir comment traiter les différences de comportement.
Suivez le guide !

67 De WINDEV à WEBDEV

Passer une application Windows en une application Web, c'est également possible !
Les étapes sont sensiblement les mêmes que pour le passage en mobile.
Suivez le guide (à nouveau) !

70 Schéma : de WINDEV à WINDEV Mobile ou WEBDEV

Le passage d'une application WINDEV en WINDEV Mobile et WEBDEV suit les mêmes étapes.
Voici un schéma récapitulatif des différentes étapes présentées précédemment.

72 De WINDEV à WINDEV Mobile ou WEBDEV : les points à contrôler

Si le passage d'une application WINDEV à une application WEBDEV ou WINDEV Mobile est rapide, il existe cependant différents points qui nécessitent des adaptations.
En effet, certaines fonctionnalités diffèrent ou ne sont pas disponibles selon la plateforme.
Voici une liste (non exhaustive) des principaux points à contrôler.

PC SOFT À VOTRE SERVICE

N'hésitez pas à nous contacter : une journée de consulting est toujours rentable pour vos développements.

Support Technique Gratuit

Le site du support technique gratuit (www.pcsoft.fr/st) met gratuitement à votre disposition un ensemble d'informations destinées à vous aider à mieux développer :

- les FAQ
- les téléchargements
- les forums développeurs professionnels (News Group)

Vous avez une question ? Contactez le Support Technique Gratuit PC SOFT !

- Avec l'utilitaire RequêteAuST (média conseillé). Votre requête sera généralement traitée sous 48 heures.
- Par téléphone au 04.67.03.17.17 (préparez votre numéro de série et votre question svp) ou par courrier.

Envoi par email : merci de ne pas attacher de fichiers de taille supérieure à 10 Mo sans l'accord préalable de votre destinataire. Sinon votre fichier sera refusé par le système.

Assistance Directe

Un complément efficace du Support Technique Gratuit

Tout projet important ou stratégique doit bénéficier d'une "Assistance Directe" !

L'Assistance Directe permet d'une part de bénéficier d'un contact téléphonique immédiat avec un ingénieur spécialisé, et d'autre part de choisir les sujets à traiter, et ce pour une durée que vous définissez vous-même. Cela vous permet de résoudre immédiatement les sujets qui vous posent problème.

Le contact téléphonique est garanti sous 16 heures ouvrables, ce qui vous permet une meilleure efficacité.
Ce service est disponible sur abonnement.

Contactez le service commercial pour plus d'informations (Fabrice CHAMBON au 04.67.032.032).

Consulting

Profitez de l'expérience d'un ingénieur de PC SOFT sur votre site

Un consulting PC SOFT peut être délégué chez vous, pour une durée de 1 à 5 jours, par un ingénieur PC SOFT connaissant à la fois votre domaine et les outils que vous utilisez.
L'ingénieur PC SOFT répond directement à vos questions, travaille sur votre projet réel et votre configuration réelle. Vous obtenez les réponses précises qui vous font gagner un temps précieux, et vous assurent des bonnes orientations technologiques et méthodologiques.

Notre conseil ? Commandez 1 jour de consulting pour 100 jours de développement.

Et, chaque année, prenez l'habitude de faire venir (juste 2 jours) chez vous un ingénieur PC SOFT, pour profiter d'une expertise de toutes les nouveautés.
Sur votre projet, dans vos locaux, en votre présence, l'ingénieur PC SOFT valide l'analyse, vous montre des nouveautés utiles dans votre cas de figure, vous propose des optimisations de votre code, valide vos modes d'utilisation, ...
Vous serez ravi de l'efficacité de son intervention ! Budgétez d'ores et déjà la prochaine intervention !

Un assistanat vous fait gagner des journées de développement !

Quelques exemples de sujets traités :

- Simplification du code et mise en place de nouvelles techniques : HFSQL Client/Serveur, Sockets, Webservices, Composants, Patchs, Configurations, ...
- Transformation de modules développés avec WINDEV en Webservices.
- Optimisation de traitements avant la recette de l'application, validation d'analyse avant de débuter une application.

Formation

Séminaires de formation pour WINDEV, WEBDEV et WINDEV MOBILE

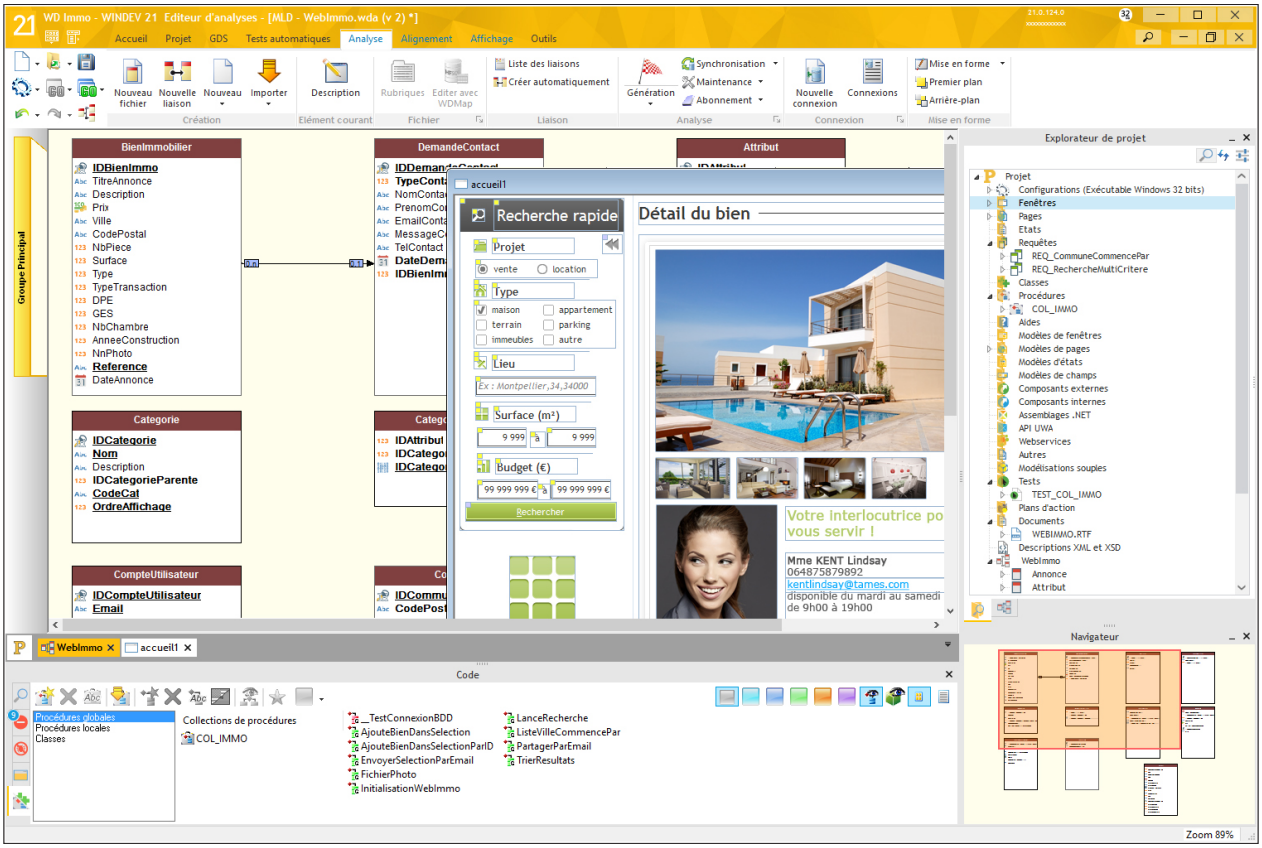
PC SOFT organise à Paris, chaque semaine, des séminaires de formation à WINDEV, WEBDEV et WINDEV Mobile, de différents niveaux : Prise en main, Perfectionnement, Expert, Client/Serveur, ...

Ces séminaires, animés par des ingénieurs PC SOFT expérimentés (compétence assurée !), permettent de découvrir et de maîtriser WINDEV, WEBDEV et WINDEV Mobile selon un plan efficace.
Consultez le calendrier des séminaires joint (le détail des stages et les contenus pédagogiques sont disponibles sur notre site Web : www.pcsoft.fr), vous trouverez LA formation adaptée à votre expérience et à votre besoin.

Suivre une formation organisée par PC SOFT est toujours rentable : c'est se donner les moyens de maîtriser WINDEV et WEBDEV dans les meilleurs délais, ou d'en découvrir la face cachée, encore plus puissante ! Inscrivez-vous vite! Ces séminaires peuvent également être organisés dans vos locaux, en France et à l'étranger.
Le montant de ces formations est déductible du "1,5 % Formation".

Dossier spécial TDF TECH 2016

Présentation



WINDEV

Présentation rapide de WINDEV et de ses fonctionnalités

WINDEV permet de développer simplement tout type d'applications Windows et Linux dans les domaines de la gestion, de l'industrie, du médical, etc.

Présentation

Les applications développées peuvent inclure l'accès à des bases de données.
WINDEV propose un puissant moteur de base de données : HFSQL.

Il est conseillé d'utiliser HFSQL pour vos applications afin d'obtenir les meilleures performances dans vos traitements de fichiers. Une version HFSQL Client/Serveur est aussi disponible.

L'environnement de développement intégré de WINDEV se compose de différents éditeurs :
• Un éditeur de projet permettant de visualiser le tableau de bord du projet.
• Un éditeur d'analyses permettant l'accès à tout type de bases de données (HFSQL, HFSQL Client/Serveur, xBase, MySQL**, AS/400*, Oracle*, SQL Server*, Access**, ODBC, OLE DB, ...).
• Un éditeur UML permettant une modélisation objet de vos données et traitements.
• Un éditeur d'IHM avec correcteur d'interface en temps réel.
• Un éditeur de code évolué (avec assistants, vérification du code saisi, coloration syntaxique, aide contextuelle, ...) incluant un puissant débogueur.
• Un éditeur de requêtes pour les sélections d'enregistrements dans les fichiers.
• Un éditeur d'états.
• Un éditeur d'aide.
• Un éditeur d'installation.
• Un éditeur de dossier technique.
* accès natif optionnel, ** accès natif inclus.

Les outils

Des outils facilitant le développement sont également fournis (WDMAP, WDSQL, ...) ainsi que de nombreux exemples et assistants réutilisables.

Des Centres de Contrôle permettent la gestion du cycle de vie de vos applications ainsi que leur administration.

Une aide en ligne vous guide tout au long du développement de votre application. Cette aide inclut également une aide à la correction d'erreurs.
Une version Internet et actualisée de l'aide en ligne est disponible à l'adresse "http://doc.pcsoft.fr".

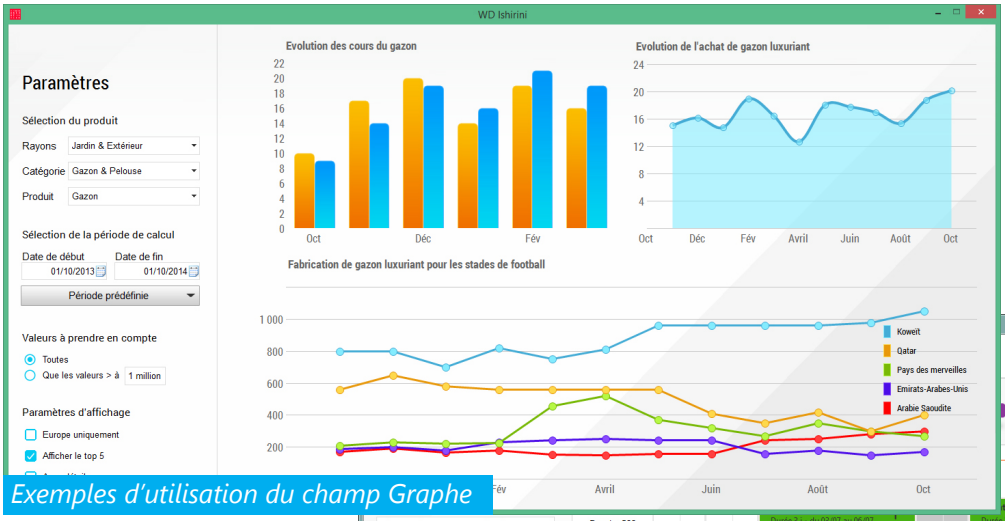
Les différents éditeurs seront utilisés, depuis la définition d'une analyse jusqu'à l'installation, en passant par toutes les phases du développement (création des fenêtres, des états, des traitements...), sans oublier les sauvegardes.

Les bases de données

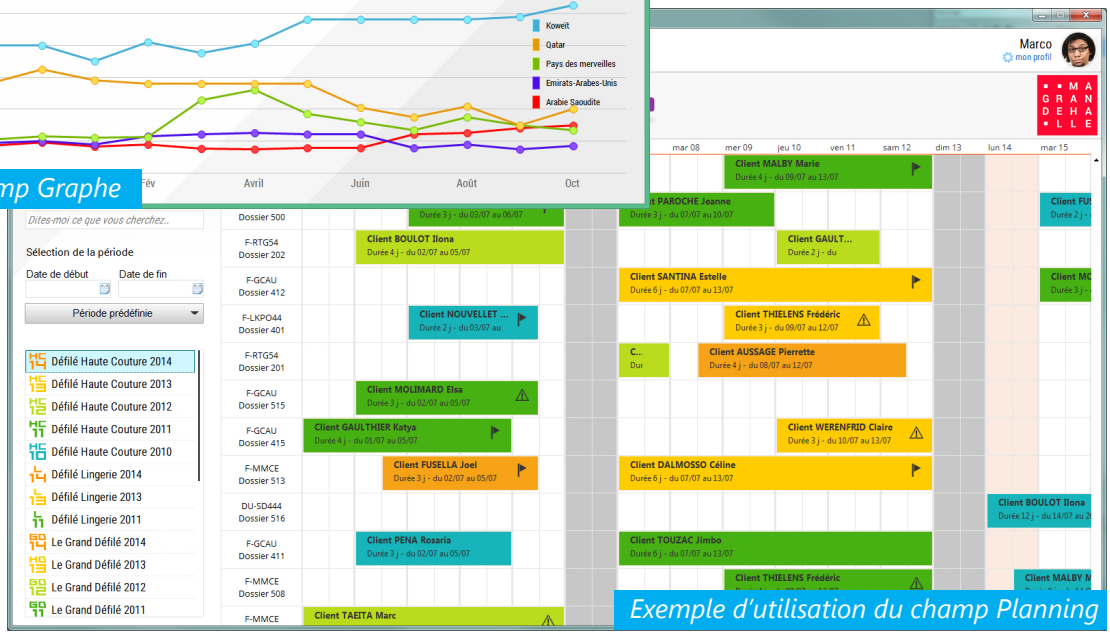
Le moteur HFSQL sera utilisé pour les fichiers de données des applications. La base de données HFSQL est disponible en mode Classic ou Client/Serveur et est librement diffusable avec les applications WINDEV.

L'utilisation d'autres moteurs de bases de données est également possible sur le même principe.

■

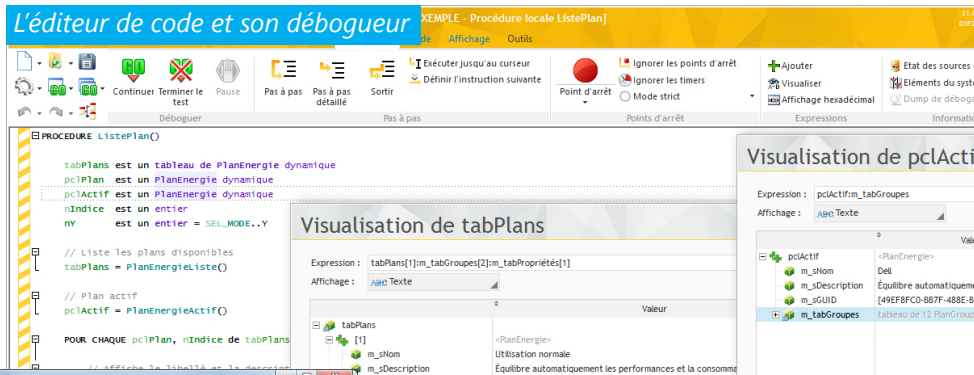


Exemples d'utilisation du champ Graphe

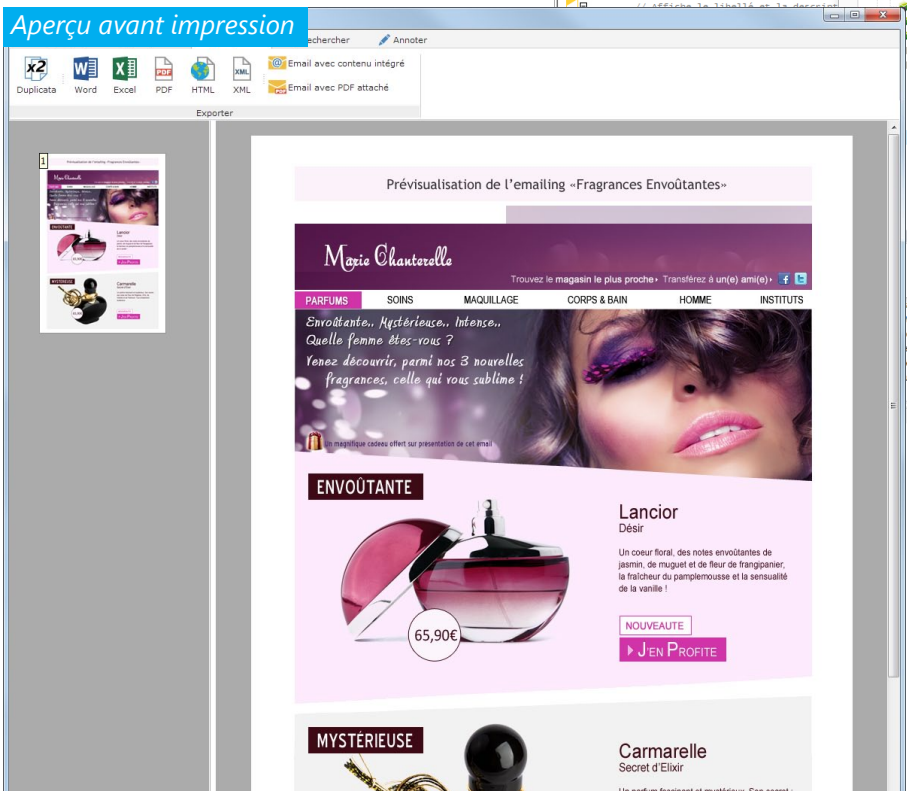


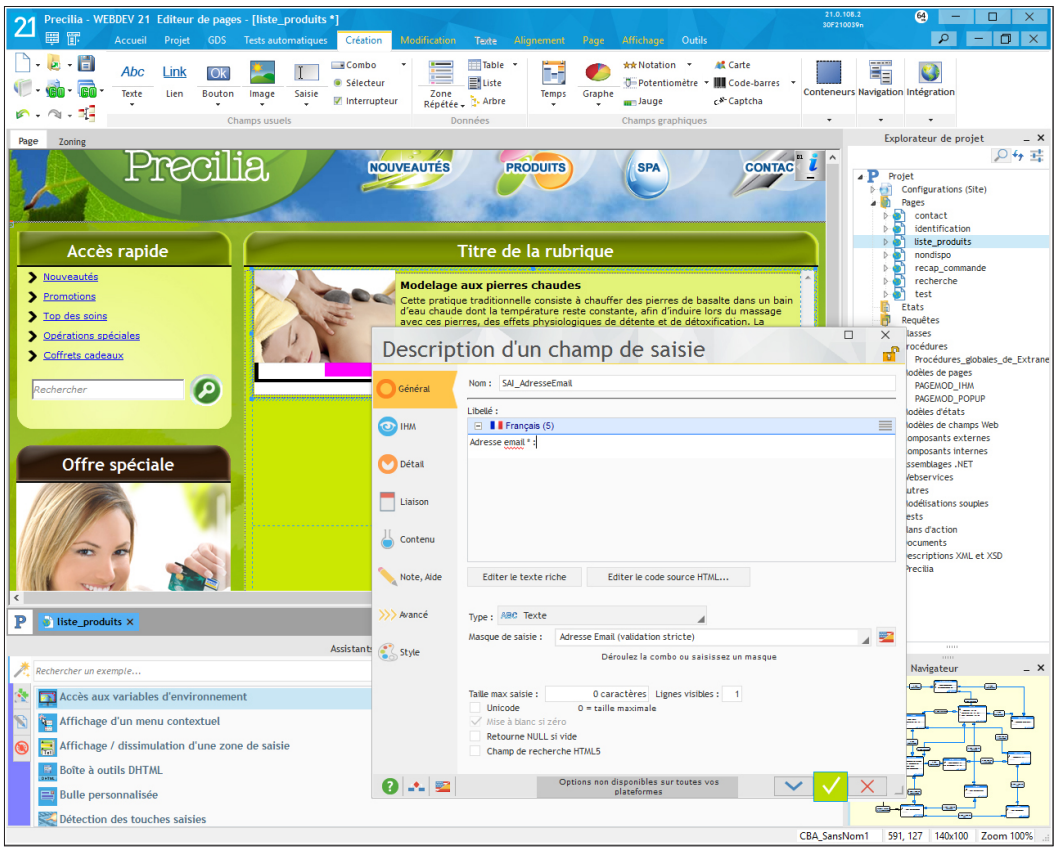
Exemple d'utilisation du champ Planning

L'éditeur de code et son débogueur



Aperçu avant impression





WEBDEV

Présentation rapide de WEBDEV et de ses fonctionnalités

WEBDEV est un AGL (Atelier de Génie Logiciel) orienté développement de sites Intranet et Internet.
WEBDEV permet de développer tout type de sites dynamiques incluant l'accès à des bases de données. Il permet aussi de développer des sites semi-dynamiques et statiques ou PHP.

Tous les sites sont compatibles avec tous les navigateurs du marché (Chrome, Edge, Firefox, ...).

Présentation de WEBDEV

WEBDEV est composé de différents éditeurs :
• Un éditeur de projet permettant de visualiser le tableau de bord du projet.
• Un éditeur de pages.
• Un éditeur d'analyses permettant l'accès à tout type de bases de données (HFSQL, HFSQL Client/Serveur, MySQL**, xBase, AS/400*, Oracle*, SQL Server*, Access**, OLE DB, ...).
• Un éditeur de code évolué (avec assistants, vérification du code saisi, coloration syntaxique, aide contextuelle, ...) incluant un puissant débogueur.
• Un éditeur de styles incluant police, couleur, position, etc.
• Un éditeur de requêtes pour les sélections d'enregistrements dans les fichiers.
• Un éditeur d'états.
• Un éditeur de dossier technique reprenant intégralement toutes les informations techniques de votre projet.
• Un éditeur d'installation.
* accès natif optionnel, ** accès natif inclus

L'éditeur d'installation assure la mise en place des sites créés, ainsi que leur maintenance. Cet outil permet de réaliser une installation sur le serveur via FTP ou par média physique (CD, Zip, etc.).
Des outils facilitant le développement sont également fournis (WDMAP, WDOUTIL, WDSQL, etc.) ainsi que de nombreux exemples et assistants réutilisables.

Les principaux éditeurs sont utilisés, depuis la définition d'une analyse jusqu'à l'installation, en passant par toutes les phases du développement (création de pages, traitements, états, etc.), sans oublier les sauvegardes.

Fonctionnalités

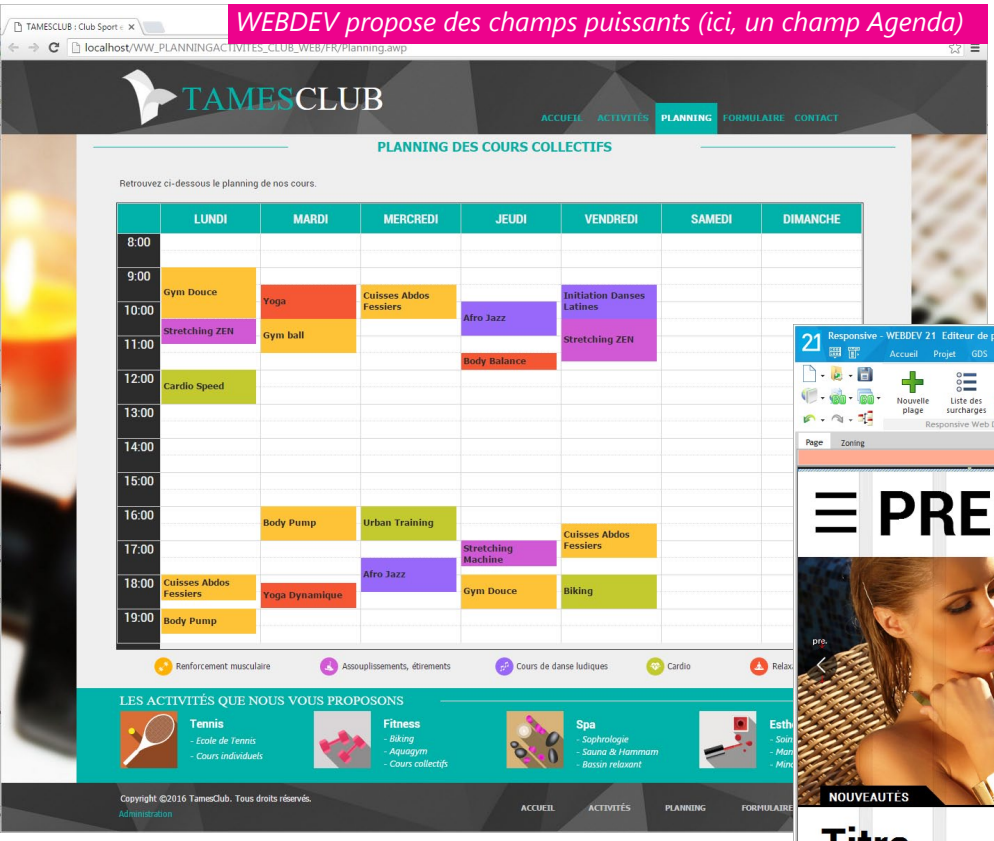
WEBDEV dispose de nombreuses fonctionnalités qui simplifient la création de sites :
• la gestion automatique des **standards HTML5 et CSS3** permettent la mise en place de sites actuels (effets, animations...),
• les **ambiances et palettes** offrent un look soigné et une cohérence graphique au site,
• le **Responsive Web Design** et sa grille de positionnement permettent de créer des sites "mobile friendly" (dont la mise en page est adaptée aux appareils mobiles),
• les **zones d'édition** permettent de créer des architectures complexes de pages en quelques clics,
• ...

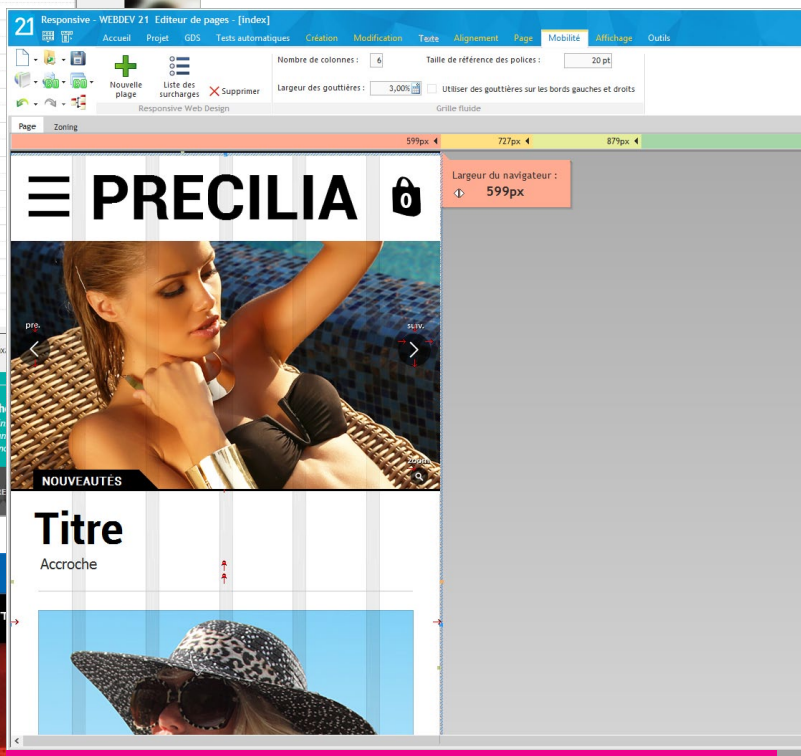
Hébergement

WEBDEV permet d'héberger les sites chez un hébergeur, en interne (via un serveur d'application) ou dans le cloud des applications PC SOFT (www.pcscloud.net).

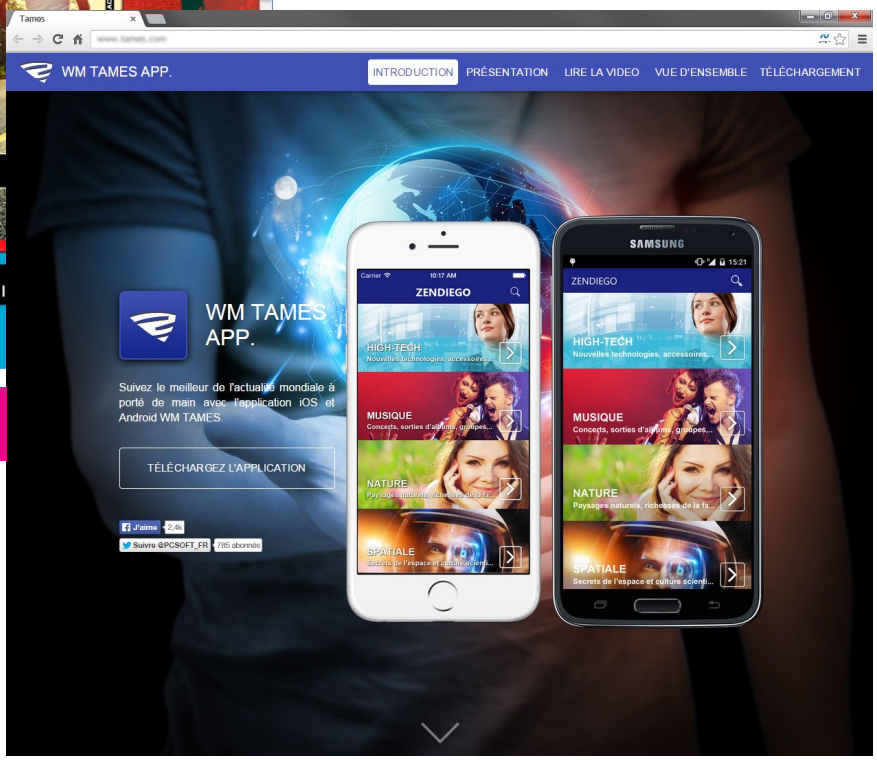
■

WEBDEV propose des champs puissants (ici, un champ Agenda)

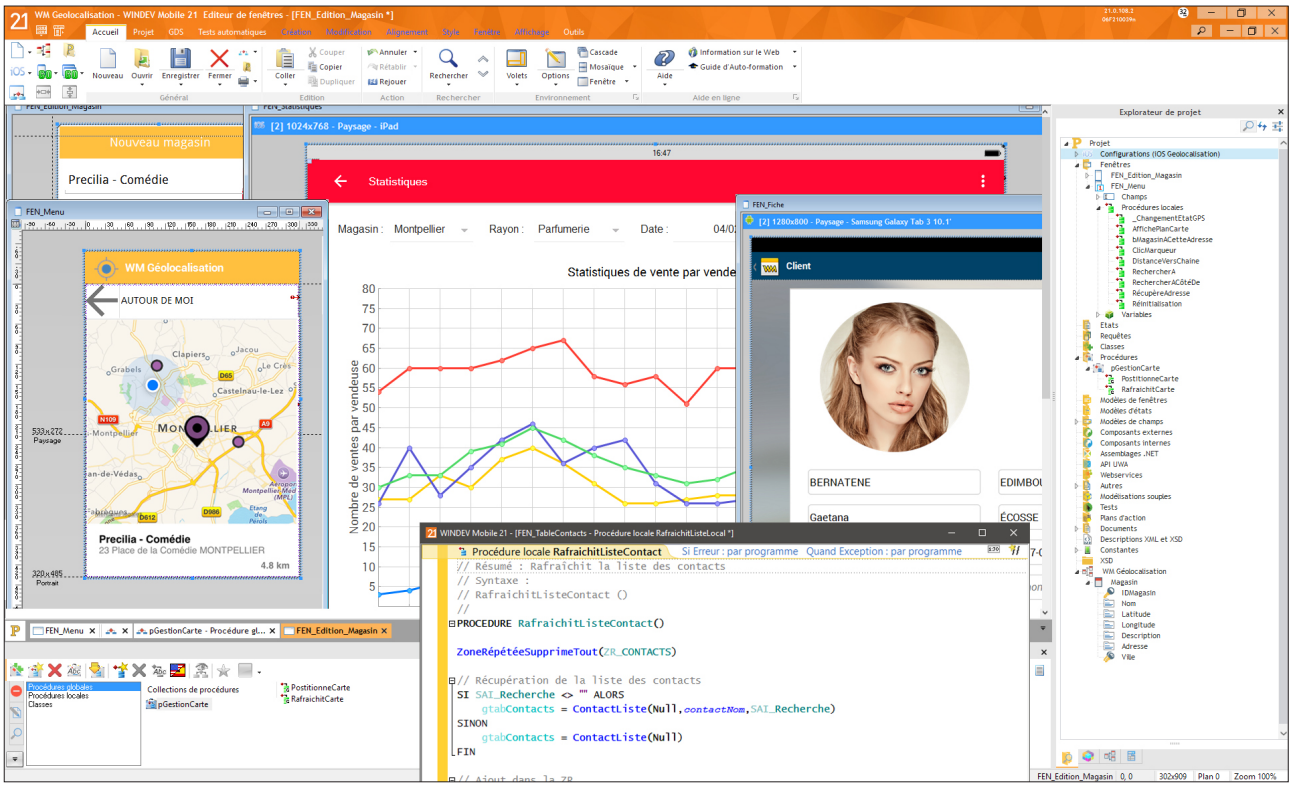




Responsive Web Design : aperçu du rendu directement en édition



Des looks actuels, sans être graphiste, grâce aux ambiances et palettes de couleurs



WINDEV MOBILE

Présentation rapide de WINDEV Mobile et de ses fonctionnalités

WINDEV Mobile permet de développer simplement tout type d'applications destinées à être utilisées sur un Smartphone, un téléphone ou une tablette Android, un téléphone Windows Phone 10, un iPhone ou un iPad et la plupart des terminaux mobiles (Pocket PC, Symbol, Psion, ...).

Présentation

Les applications développées peuvent accéder à des bases de données.
WINDEV Mobile inclut un puissant moteur de base de données : HFSQL Mobile.

Il est conseillé d'utiliser HFSQL pour vos applications afin d'obtenir les meilleures performances dans vos traitements de fichiers. Une version HFSQL Client/Serveur est aussi disponible.

WINDEV Mobile permet également d'accéder aux bases de données SQLite pour les applications Android.

Environnement

L'environnement de développement intégré de WINDEV Mobile se compose de différents éditeurs :

- Un éditeur de projet permettant de visualiser le tableau de bord du projet.
- Un éditeur d'analyses permettant l'accès à tout type de bases de données (HFSQL, HFSQL Client/Serveur, ...).
- Un éditeur UML permettant une modélisation objet de vos données et traitements.
- Un éditeur d'IHM avec correcteur d'interface en temps réel.
- Un éditeur de code évolué (avec assistants, vérification du code saisi, coloration syntaxique, aide contextuelle) incluant un puissant débogueur
- Un éditeur de requêtes pour les sélections d'enregistrements dans les fichiers.
- Un éditeur d'états.
- Un éditeur d'installation.
- Un éditeur de dossier technique reprenant intégralement toutes les informations techniques de votre projet.

Des outils facilitant le développement sont également fournis ainsi que de nombreux exemples et assistants réutilisables.

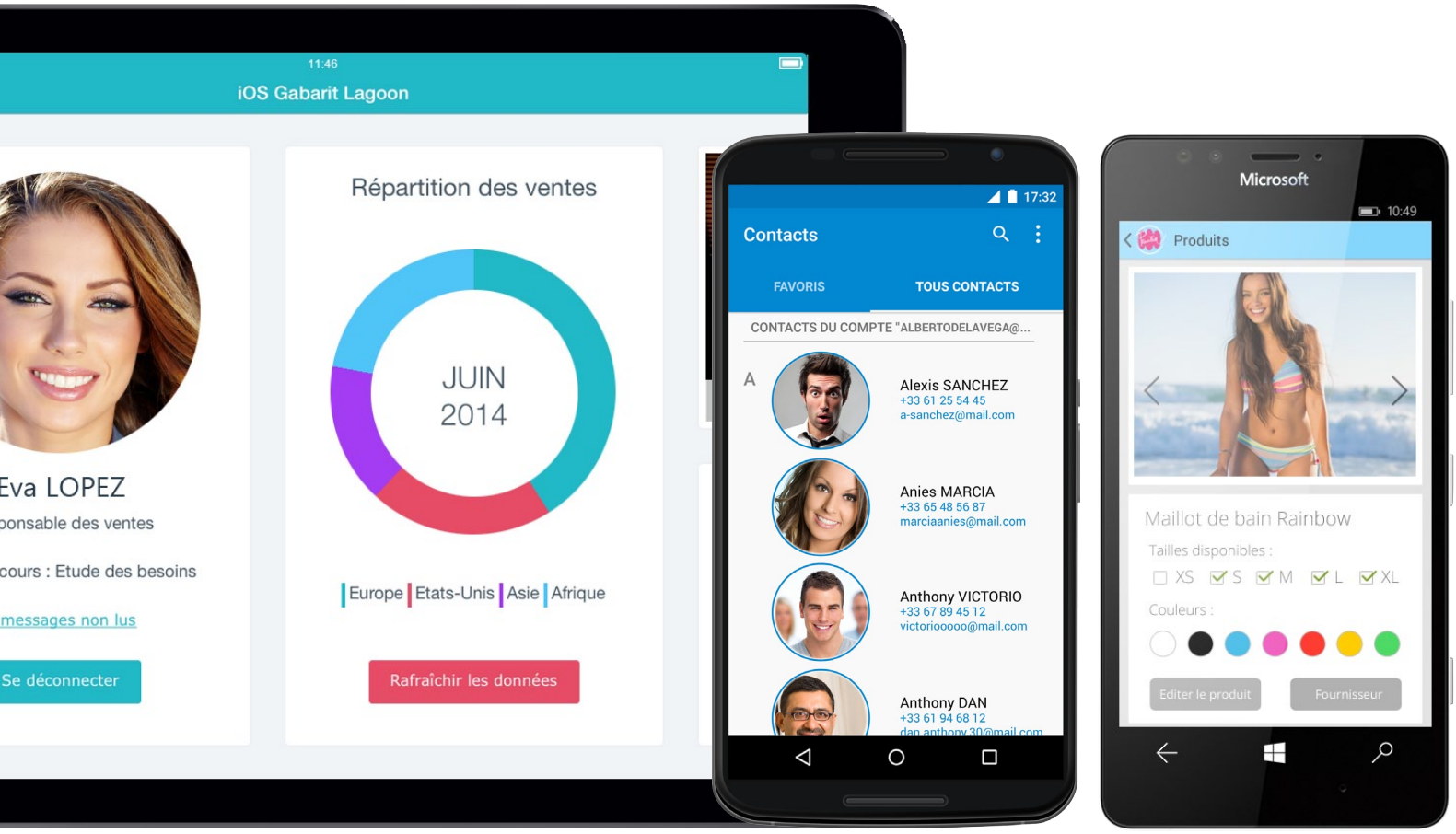
Des Centres de Contrôle permettent la gestion du cycle de vie de vos applications ainsi que leur administration.

Une aide en ligne vous guide tout au long du développement de votre application. Cette aide inclut également une aide à la correction d'erreurs. Une version Internet et actualisée de l'aide en ligne est disponible à l'adresse "http://doc.pcsoft.fr".

Le moteur HFSQL Mobile sera utilisé pour les fichiers de données de l'application.

L'utilisation d'autres moteurs de bases de données est également possible sur le même principe.

■



Vous réalisez des applications mobiles (téléphone et tablette) pour Android, iOS, Universal Windows 10 App, Windows CE, ...

Dossier spécial TDF TECH 2016

Support de cours

ENVIRONNEMENT : 10 NOUVEAUTÉS UTILES !

En version 21, les éditeurs de WINDEV, WEBDEV et WINDEV Mobile disposent de nouvelles fonctionnalités pratiques.
Voici 10 fonctionnalités, nouvelles ou non, que vous devez connaître !

1 Gestion des configurations

Une configuration de projet permet de définir un type de génération spécifique pour le projet: Windows 32 bits, Windows 64 bits, Linux, Android, iOS, etc.

La mise en place de différentes configurations dans un même projet facilite le partage des éléments (fenêtres, collections de procédures, etc.) entre plusieurs plateformes.

Pour créer une nouvelle configuration, vous pouvez maintenant **dupliquer une configuration existante**. La nouvelle configuration contient ainsi les mêmes éléments: particulièrement utile pour créer une configuration iOS dans un projet Android par exemple.

Depuis la fenêtre de gestion des configurations, vous sélectionnez la configuration "source" et vous cliquez sur le bouton "Dupliquer".

Il vous suffit ensuite d'ouvrir la description de cette copie pour modifier son nom et sa cible.

Autre fonctionnalité bien pratique: depuis le menu contextuel d'un élément de l'explorateur de projet, vous pouvez **fixer les configurations dans lesquelles l'élément est disponible**. Il est ainsi inutile de se positionner dans chaque configuration pour inclure ou exclure les éléments.

2 Nommage des langues

Par défaut, 48 langues "standards" sont disponibles dans les produits: français, anglais, allemand, chinois, japonais, ...

Vous disposez également de 16 langues personnalisées si besoin: par exemple, si votre application est utilisée en Tanzanie, au Kenya ou en Ouganda, vous pourriez afficher votre application en Kiswahili (la langue officielle) !

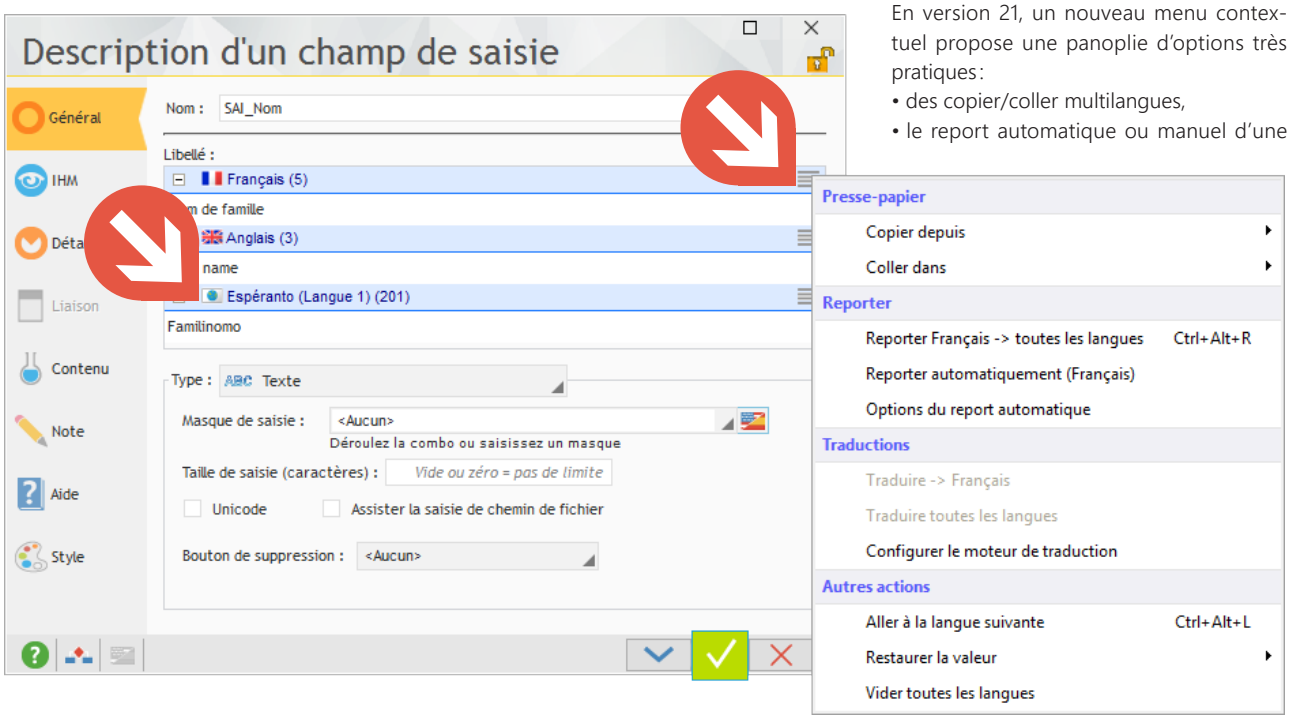
Pour ajouter une langue à votre projet, affichez la description du projet, onglet "Langues" et cliquez sur le bouton "Ajouter".
Si vous sélectionnez une langue personnalisée, vous pouvez saisir son libellé directement depuis la fenêtre d'ajout.
Si vous souhaitez modifier le libellé d'une langue personnalisée déjà présente dans le projet, positionnez-vous sur l'onglet "Divers", option "Nom de la langue".

La dénomination de la langue personnalisée sera utilisée partout: lors de la complétion de la fonction WLangage ***Nation***, dans les fenêtres de descriptions des champs, etc.

3 Option de langues

Dans la fenêtre de description des champs, pour tous les éléments d'IHM (libellé, texte d'indication, bulle, etc.), les différentes langues du projet sont proposées.

En version 21, un nouveau menu contextuel propose une panoplie d'options très pratiques:
• des copier/coller multilangues,
• le report automatique ou manuel d'une



langue vers les autres (pour éviter d'avoir des textes par défaut),
• l'accès direct au moteur de traduction (Google Traduction par exemple, voir le paragraphe suivant),
• ...

4 Google traduction

Pour traduire vos IHM facilement, vous avez le choix du moteur à utiliser :
• l'application WDDixio (livrée avec WDTrad),
• une application tierce,
• un site Web,
• Google Traduction (une clé API est requise).

Pour sélectionner le moteur de traduction à utiliser, affichez les options générales de WINDEV, onglet "Traductions".

L'utilisation d'un moteur de traduction, accessible directement depuis l'éditeur, permet de traduire vos IHM directement, dès la conception !
Vous avez alors une première version de l'application plus rapidement.
Libre à vous ensuite d'extraire les ressources grâce à l'outil WDMsg pour les faire traduire par un cabinet de traduction.

5 Relancer le dernier test

Lorsque vous réalisez une phase de débogage, tous les éléments sur lesquels vous vous arrêtez (point d'arrêt, pause, etc.) s'ouvrent dans l'éditeur.

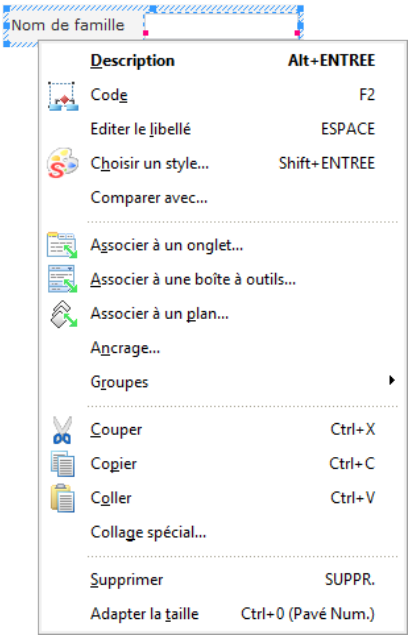
Parfois, après avoir corrigé un problème dans une fenêtre, vous aimeriez refaire un Go de la fenêtre initialement testée, sans avoir à vous repositionnez dessus.

L'option "Relancer le dernier test" (disponible dans le sous-menu du bouton "Go de fenêtre") permet de refaire un go de fenêtre sur la dernière fenêtre testée, même s'il ne s'agit pas de la fenêtre courante.

6 Les raccourcis

Dans toute application, l'utilisation de raccourcis est souvent un avantage pour gagner en rapidité.
WINDEV, WEBDEV et WINDEV Mobile foisonnent de raccourcis pratiques sur les actions usuelles.

En version 21, tous les raccourcis sont affichés dans les menus contextuels.



C'est beaucoup plus pratique pour les apprendre et donc les réutiliser !

7 Boutons ronds

En développement mobile, les boutons ronds sont devenus incontournables. Plébiscité par les recommandations "Material Design" de Google, le bouton rond flottant identifie l'action principale de la fenêtre en cours.

Pour faire un bouton rond, depuis l'éditeur de fenêtres de WINDEV Mobile, c'est très simple : il suffit de le dropper depuis le ruban.

Les boutons ronds sont également disponibles pour WINDEV : dans la description du bouton, onglet "Style", positionnez-vous sur l'élément "Cadre extérieur" et sélectionnez "Rond".
Vous pouvez également appliquer le style de gabarit "BTN_Flottant" (clic droit, "Choisir un style...").

8 Recherche évoluée

L'onglet de recherche dans les éditeurs permet d'effectuer des recherches très précises :
• le code qui contient "client" sans tenir compte de la casse,
• les dernières modifications faites par Marc depuis 2 semaines,
• ...

Une nouvelle option de recherche vous permet de retrouver des éléments (fenêtre, champs, classes, ...) qui correspondent à un ou plusieurs critères. Il est possible par exemple de rechercher :
• tous les champs Bouton dont le nom contient "Plus" pour vérifier leur bulle d'aide,
• toutes les classes qui utilisent le fichier HFSQL "Client",
• ...

Vous pouvez rechercher sur quatre critères différents, parmi plus de quarante critères possibles.

9 Comparer deux champs

Dans certains cas, il peut être pratique de comparer 2 champs situés dans 2 fenêtres différentes.

Par exemple, si vous utilisez un modèle de champs, il peut être pratique de comparer les surcharges effectuées sur les 2 modèles !

Pour comparer deux champs situés dans une même fenêtre, c'est très simple : sélectionnez les champs à comparer et dans le menu contextuel (clic droit), sélectionnez l'option "Comparer les deux éléments".

Pour comparer des éléments situés sur des fenêtres différentes :
• sélectionnez le premier champ,
• sélectionnez l'option "Comparer avec ..." dans le menu contextuel (clic droit),
• suivez l'assistant pour sélectionner l'élément de comparaison. Vous pouvez choisir une fenêtre à partir de son emplacement physique (sur le disque) : il est ainsi possible de comparer un élément d'une fenêtre d'un autre projet.

10 Installation automatique du SDK Android

Pour compiler et générer des applications Android depuis WINDEV Mobile, le poste de développement doit disposer :
• du SDK Android,
• de Gradle (à partir de la version 21 de WINDEV Mobile). Gradle est l'outil de construction d'applications utilisé par Google pour générer les binaires Android.

Auparavant, pour mettre à jour le SDK Android, il fallait lancer l'application "SDK Manager" et lancer le téléchargement des modules souhaités.

En version 21, WINDEV Mobile vous fait gagner des clics. Sur le premier plan de l'assistant de génération, vous pouvez demander à WINDEV Mobile de télécharger et installer automatiquement le SDK Android et Gradle.

■

LE GDS : L'OUTIL INDISPENSABLE

Le GDS est un gestionnaire de sources élaboré qui permet de sauvegarder les sources, les historiques, les versions,...
Primordial lors du développement en équipe, le GDS s'avère également très utile si vous travaillez seul.
Voici 5 avantages à utiliser le GDS !

Préambule

Si vous ne connaissez pas le GDS ni son fonctionnement, nous vous invitons à consulter les pages de documentation à son sujet avant de lire cet article : "http://doc.pcsoft.fr/?2038001".

1 Historique : ne perdez plus vos modifications

La principale caractéristique d'un gestionnaire de sources est l'accès à l'historique de l'élément. Chaque réintégration crée une ligne d'historique.

Pour accéder à l'historique d'un élément, vous avez plusieurs solutions :
• depuis le ruban de l'éditeur, sous le volet "GDS", dans le groupe "Elément en cours", cliquez sur le bouton "Historique".
• depuis l'administrateur du GDS, sélectionnez un élément puis utilisez l'option "Historique" du menu contextuel ou le bouton "Historique" du ruban (volet "Fichiers").

Depuis l'éditeur de code, vous avez également accès aux modifications de l'élément. Pour cela, vous devez afficher la date et l'auteur de chaque ligne de code (raccourci F6).
En cliquant sur le lien qui correspond à la date de réintégration de la ligne de code, une popup [**Fig. 1**] vous permet de :
• voir le commentaire de réintégration correspondant,
• afficher la modification du code,
• afficher toutes les modifications (ihm et code) de l'élément en cours par rapport à la version,
• accéder à l'historique complet de l'élément courant.

La fenêtre d'historique d'un élément affiche pour chaque ligne :
• le développeur qui a effectué la modification,
• la date de modification,
• la version interne (VI) utilisée,
• le commentaire de réintégration.

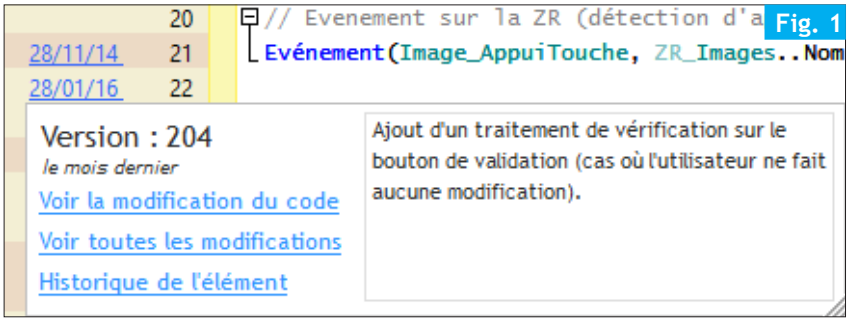


Fig. 1

Pour une ligne d'historique, il est possible de :
• revenir à cette version,
• récupérer l'historique (pour test),
• enregistrer l'historique,
• comparer / fusionner avec la version courante de l'élément.

Le "diff" : comparer et récupérer

Tout l'intérêt d'un historique est de pouvoir comparer l'élément courant avec une de ses versions d'historique, pour :
• voir ce qui a changé et qui expliquerait une différence de comportements,
• reporter les propriétés ou lignes de code de la version d'historique.

Les étiquettes : des marqueurs utiles

Une étiquette ( ) symbolise la création d'une version client ou la création d'une branche. En un coup d'œil, vous voyez quelle version de l'élément a été intégrée dans la version déployée.

Depuis l'historique du projet, vous pouvez ajouter vous-même une étiquette sur une ligne d'historique.

Les étiquettes permettent également de créer une branche a posteriori.

2 Gestion des droits : contrôlez les accès

La gestion des droits d'accès aux éléments du GDS permet de limiter les accès (et donc les modifications) pour chaque développeur et pour chaque élément du GDS.

Un droit est associé :
• à un compte de connexion (un développeur),
• à un groupe (uniquement si la base du GDS est en mode HFSQL Client/Serveur).

Il est possible de définir de droits :
• sur un répertoire,
• sur un fichier.

La gestion des droits est réalisée depuis l'administrateur du GDS.

Définir un droit

La définition des droits d'un élément s'effectue dans la fenêtre des propriétés de l'élément : sélectionnez l'élément et cliquez sur "Propriétés" dans le menu contextuel. Le volet "Droits" affiche les droits définis de l'élément.

Par défaut, "tout le monde a tous les droits".

Pour ajouter un droit (ou une restriction), il suffit de cliquer sur le bouton "Ajouter" et de sélectionner la portée du droit : tout le monde, un groupe, un utilisateur.
Une ligne est alors ajoutée à la table des droits : sélectionnez l'utilisateur (ou le groupe) et modifiez ses droits. Il existe différents types de droits.

Types de droits

1. Contrôle total : l'utilisateur peut réaliser toutes les opérations.
2. Lecture : l'utilisateur peut lire l'élément (récupérer une version et extraire pour test). Si un utilisateur n'a pas le droit de lecture sur un répertoire, les fichiers du répertoire sont invisibles.
3. Écriture : l'utilisateur peut ajouter, modifier ou supprimer l'élément.
4. Modifier les droits : l'utilisateur peut modifier les permissions des autres utilisateurs.

3 Branches : mettez en place des versions de vos applications

Les branches GDS permettent de gérer en parallèle plusieurs versions d'une application. Par exemple, une version de l'application est diffusée en clientèle : elle est dans une branche. Le tronc commun contient la future version de l'application.
Vous pouvez continuer de travailler sur la version en clientèle et même reporter des corrections de bugs sur cette version à partir du tronc commun.

Créer une branche

1. Pour créer une branche, sous le volet "GDS", dans le groupe "Projet", déroulez "Branches" et sélectionnez "Créer une branche". La branche est alors créée à partir de la version en cours des éléments.
2. Une branche peut être créée automatiquement lorsque vous ouvrez un projet dans une version supérieure de WINDEV (par exemple, si vous ouvrez un projet 20 avec une version 21).
3. Une branche peut être créée a posteriori, depuis l'historique du projet. Dans l'administrateur du GDS, ouvrez le menu contextuel sur le répertoire du projet et sélectionnez "Historique". Vous pouvez alors créer une branche à partir d'une ligne d'historique via le bouton "Créer une branche".

Reporter une modification

Pour récupérer une modification depuis une branche, sous le volet "GDS", dans le groupe "Projet", déroulez "Branches" et sélectionnez "Récupérer des modifications depuis une branche".

L'assistant de report de modifications s'affiche. Pour chaque élément différent (ajouté, modifié ou supprimé), il est possible de choisir l'action à effectuer :
• ne rien faire,
• ajouter dans le projet,
• ...

4 Partage d'éléments : mutualisez vos créations

L'un des avantages majeurs du GDS est de pouvoir partager un élément entre plusieurs projets :
• une fenêtre,
• une collection de procédures,
• un modèle de fenêtres ou de champs,
• un composant interne,
• ou même une analyse !

Cette fonctionnalité présente plusieurs avantages :
• les modifications sont reportées automatiquement dans tous les projets qui partagent l'élément,
• cet élément est modifiable directement dans chacun des projets.

Partager un élément

Pour partager un élément, depuis l'éditeur, sous le volet "GDS", dans le groupe "Projet", cliquez sur "Importer depuis le GDS".

La fenêtre qui s'ouvre permet de :
• rechercher un élément par son nom,
• parcourir l'arborescence du GDS.

Lorsque vous sélectionnez un élément, la liste des partages est affichée.
Pour partager l'élément, il vous suffit de cliquer sur le bouton "Partager" (en bas de la fenêtre).

Le bouton "Importer une copie" permet de récupérer une copie locale de l'élément dans le projet : l'élément n'est alors pas partagé.

Remarque
Il est également possible de partager un élément depuis l'administrateur du GDS : il suffit d'utiliser le bouton ou d'effectuer un Drag and Drop de l'élément (fenêtre, état, etc.) depuis la table sur le projet (dans l'arbre).

5 Politique de réintégration : la contrainte positive

Une politique de réintégration définit des règles qui doivent être respectées pour que le développeur soit autorisé à réintégrer des éléments d'un projet.

Définir une politique

Pour définir une politique de réintégration :
• depuis l'éditeur, sous le volet "GDS", dans le groupe "Base GDS", déroulez "Gérer" et sélectionnez "Politique de réintégration".
• depuis l'administrateur du GDS, ouvrez le menu contextuel d'un répertoire de projet et sélectionnez l'option "Politique de réintégration".

Les règles sont découpées en 3 thèmes :
• Général : aucune erreur/warning/information, commentaire obligatoire...
• Tests automatiques : présence de tests automatiques, pas de tests en erreur.
• Métrique de code : taux de commentaires.

Fonctionnement

À la réintégration d'un élément, si un critère n'est pas respecté, une fenêtre [**Fig. 2**] alerte l'utilisateur du non-respect de la politique de réintégration et la réintégration n'est pas effectuée.

Il est possible d'ignorer les règles de la politique à la réintégration (option "Ces règles peuvent être ignorées"). Cela permet de passer outre les restrictions, notamment pour des éléments de test. Dans ce cas, l'utilisateur peut réintégrer son élément (en saisissant un commentaire facultatif sur cette réintégration "forcée").

Notre astuce
Si vous permettez d'ignorer la politique de réintégration, nous vous conseillons de forcer la saisie d'un commentaire. Vous conservez ainsi un suivi et évitez d'éventuels abus.

■

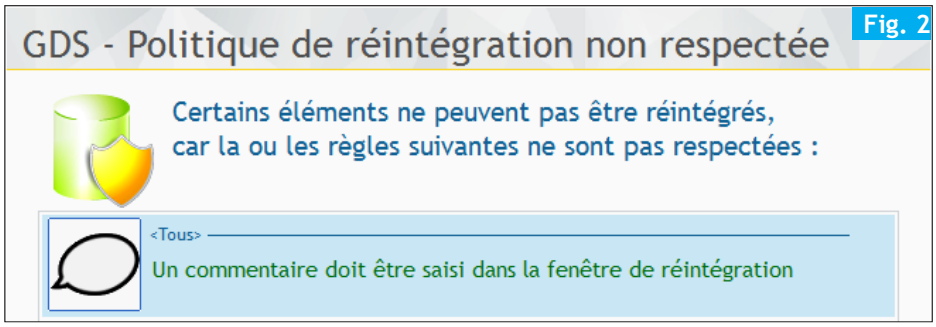


Fig. 2

LE CENTRE DE SUIVI DE PROJETS

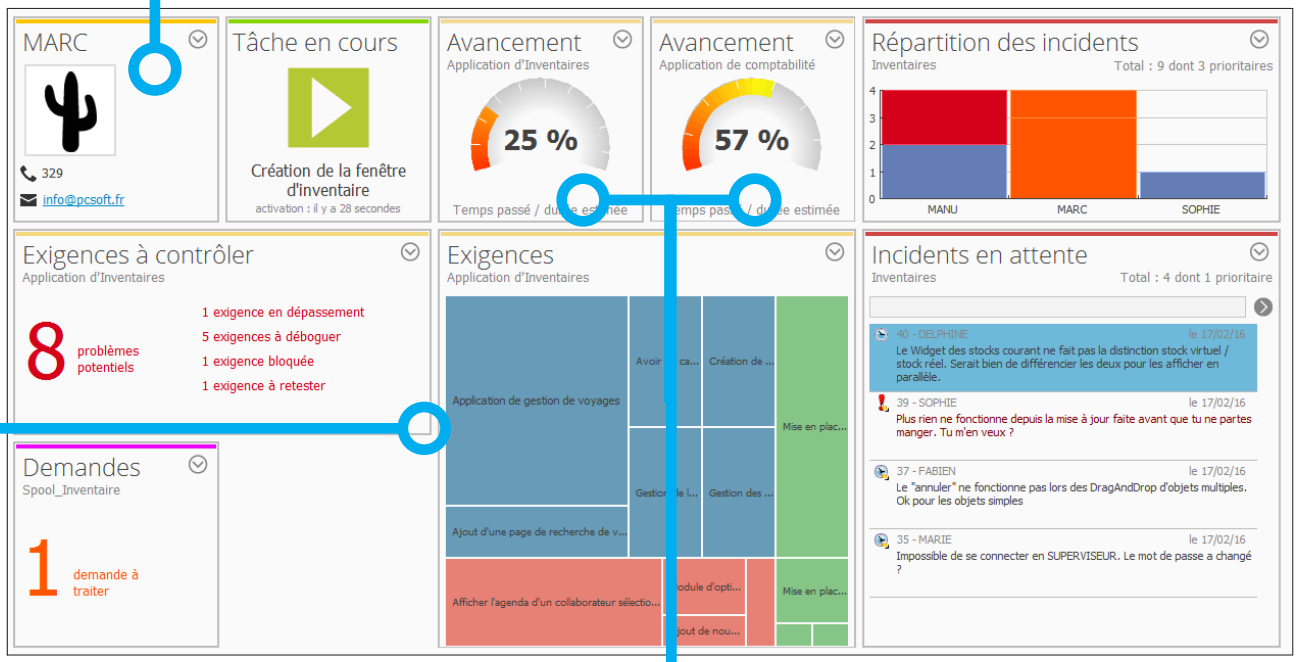
Le Centre de Suivi de Projets permet de gérer les tâches, les incidents, les exigences, les retours clients...
Cet outil offre de nombreuses fonctionnalités utiles dans votre travail quotidien.
Cet article présente 3 fonctionnalités immédiatement utiles.

1 Un tableau de bord adapté au rôle de chacun

Le Centre de Suivi de Projets propose un tableau de bord personnalisable qui vous permet d'ajouter les widgets qui vous intéressent. Que vous soyez développeur, chef de projet ou membre du service qualité, vous trouverez forcément les widgets qui vous seront utiles et qui vous offriront une vision globale sur votre travail et celui de vos collègues.

Personnalisation totale

Comme tout tableau de bord WINDEV, ce tableau de bord est entièrement personnalisable : vous affichez les widgets que vous souhaitez et vous les disposez à votre convenance.



Des widgets adaptés à chacun

Le tableau de bord du Centre de Suivi de Projets propose différents widgets :
- la tâche en cours (pour en changer rapidement),
- l'avancement courant d'une liste d'exigences,
- un compteur d'incidents,
- la répartition des incidents par développeur,
- les incidents en attente,
- les exigences (à contrôler et en cours de réalisation).

Plusieurs instances d'un même widget

Bien entendu, vous avez la possibilité d'afficher un même widget plusieurs fois, en modifiant sa source (depuis le bouton ⌄).

Par exemple :
- le widget "Avancement" est basé sur une liste d'exigences,
- les widgets relatifs aux incidents (comme le "Compteur") sont basés sur un lot d'incidents donné.

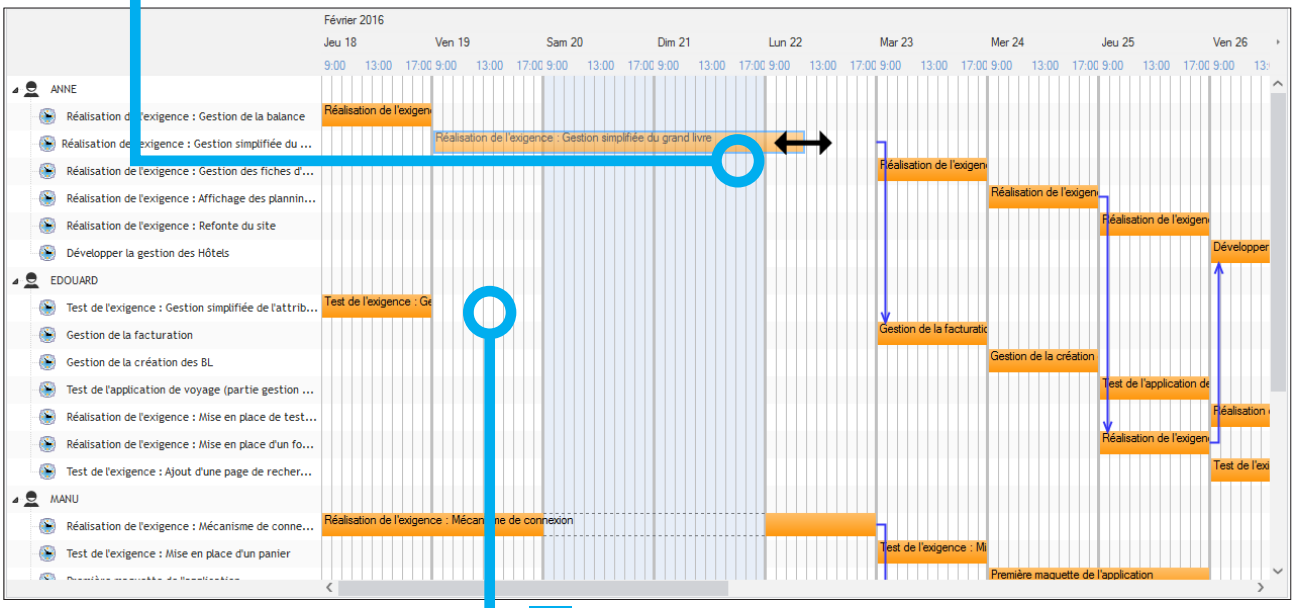
Vous pouvez tout à fait afficher 2 fois le widget "compteur" en sélectionnant un lot d'incidents différent.

2 Un diagramme de Gantt interactif

Le diagramme de Gantt permet de visualiser, pour chaque développeur, l'enchaînement de ses tâches et éventuellement ses temps morts. Directement depuis le diagramme, vous avez la possibilité d'ajuster au mieux chaque tâche : déplacement, réaffectation, ajustement de la durée, modification des dépendances...

Ajuster la durée

Une tâche sous-évaluée ? Aucun problème !
Directement depuis le diagramme, vous pouvez augmenter la durée de la tâche pour avoir un aperçu réel du nouveau planning.



Combler les trous

Un développeur en attente d'une autre tâche, sans rien à faire ? Récupérez une tâche indépendante et affectez-la directement au développeur.

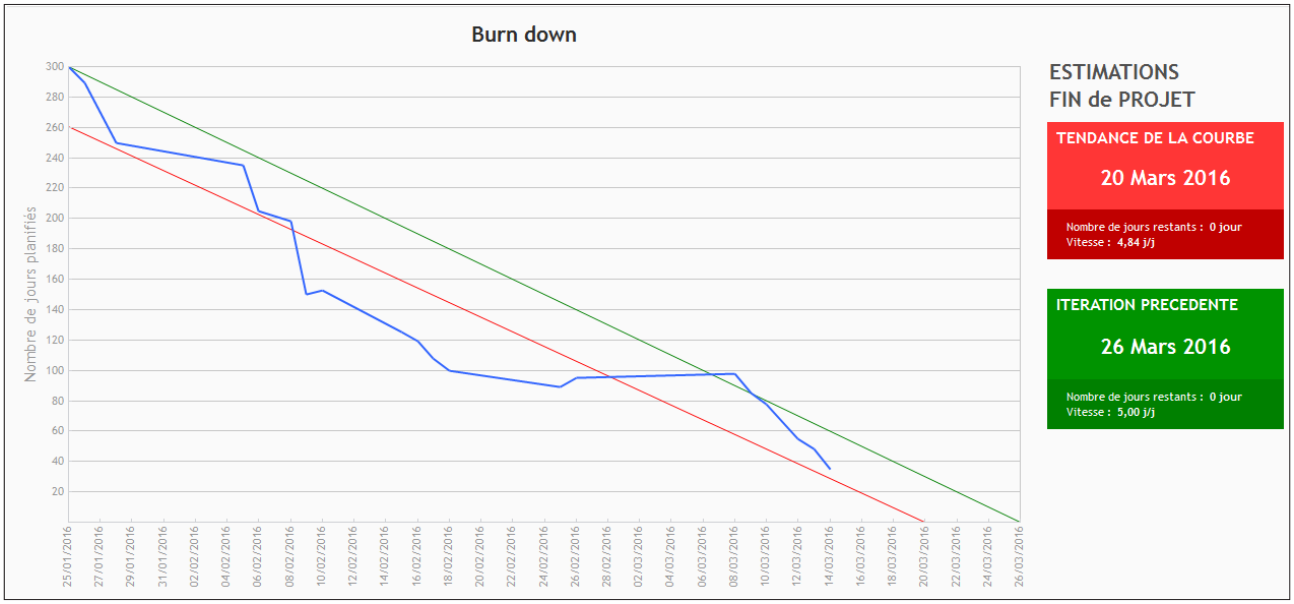
3 Une courbe d'avancement (Burndown Chart) pour estimer au mieux

La courbe d'avancement (nommé "Burndown chart" en méthode agile) vous permet d'estimer au mieux la fin d'un projet en fonction de l'avancement réel courant ainsi que des projets précédents. Pour afficher une courbe d'avancement, sélectionnez une liste d'exigences et cliquez sur "Courbe d'avancement" (dans le groupe "Gestion").

Dans la courbe d'avancement, l'axe horizontal représente le temps et l'axe vertical le nombre de jours planifiés pour toute l'équipe.
La courbe représente donc la quantité de travail qui reste à fournir pour terminer le projet.

La droite rouge représente la tendance de la courbe. La pente de cette droite correspond à la vitesse de l'équipe. Cette tendance permet d'estimer la date de fin du projet.

La droite verte reprend la pente de la courbe d'une itération précédente. Cette courbe permet de voir si l'équipe va plus vite ou moins vite et permet également d'obtenir une date de fin incluant l'expérience précédente (donc plus réaliste).



EDITEUR D'ÉTATS : 5 NOUVEAUTÉS UTILES

Dans la majorité des applications, les éditions ont une place importante : facture, formulaire, plaquette...
En version 21, l'éditeur d'états évolue et vous propose de nouvelles fonctionnalités utiles au quotidien dans vos applications.

Projet d'illustration

Le projet utilisé pour illustrer ce sujet est l'exemple "WD TCD et Etats TDF21" (présent sur le DVD).

1 Ascenseur visuel

Si votre état est particulièrement grand, il peut être pratique d'utiliser l'ascenseur visuel.
L'ascenseur visuel vous permet de visualiser une miniature de l'état et donc de vous positionner très facilement dans l'état.

Pour activer l'ascenseur visuel depuis l'éditeur d'états, dans le ruban, sous le volet "Affichage", dans le groupe "Visualisation", cochez l'option "Ascenseur visuel".

Pour tester l'ascenseur visuel dans notre exemple, ouvrez l'état "ETAT_Plaquette".

Remarque: vous pouvez également bénéficier de cet ascenseur visuel dans l'éditeur de code. Il s'agit d'un paramètre de l'éditeur de code.
Dans la fenêtre d'option de l'éditeur de code, onglet "Code", vous pouvez modifier le type d'ascenseur pour un ascenseur visuel (qui s'affichera au choix à droite ou à gauche).

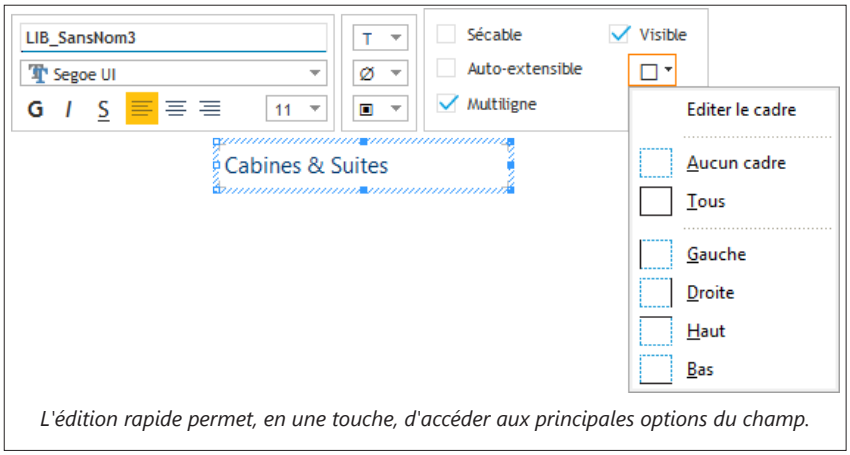
2 Édition rapide des champs

Pour éditer un champ d'un état, vous pouvez, au choix, ouvrir la description du champ ou utiliser les options du ruban (volet "Modification").

Pour aller encore plus vite et avoir accès aux principales options du champ, l'éditeur d'états propose l'édition rapide des champs, activable avec le raccourci "R".

L'édition rapide vous permet de modifier :
- le nom du champ,
- la police (nom, taille, graisse, souligné, etc.),
- le cadre,
- les couleurs (texte, fond, cadre),
- les options de mise en forme (sécable, multiligne, etc.),
- ...

Bien entendu, les options sont contextuelles au type de champ : pour un champ Image par exemple, vous aurez accès au chemin de l'image et à son mode d'affichage.



L'édition rapide permet, en une touche, d'accéder aux principales options du champ.

3 PDF en fond (vectoriel)

Pour imprimer et remplir un formulaire, vous pouvez afficher un PDF en fond d'état.

Dans la description de l'état, onglet "Fond", il suffit de sélectionner le fichier PDF comme "Image de fond". Vous pouvez également choisir d'afficher toutes les pages du PDF ou seulement une page donnée.

En version 21, les PDF en fond d'état sont automatiquement de meilleure qualité : ils sont maintenant dessinés de manière vectorielle. La qualité de ces impressions est donc optimale.

4 Génération au format PDF/A

Le PDF/A est un format PDF standardisé par l'ISO (norme ISO 19-005).

Cette norme garantit que le PDF sera toujours affichable et imprimable, exactement avec la même mise en page, même dans 50 ans.

Ce type de PDF embarque dans le fichier PDF tous les éléments nécessaires à son affichage ou impression : par exemple, toutes les polices sont systématiquement incluses.

Il s'agit d'un format destiné à l'archivage électronique à long terme.

Ce format PDF oblige à respecter certaines limitations (par exemple, la transparence est interdite).

Pour vous aider à mettre en place des états qui respectent ce format, l'éditeur d'états peut afficher une erreur d'IHM si un élément

de votre état ne respecte pas les limitations du format PDF/A : dans la description de l'état, onglet "Options", cochez simplement l'option "Activer les erreurs spécifiques à l'export PDF/A-1b".

5 Champ Signature

En version 21, les PDF générés par l'éditeur d'états peuvent être signés. Pour signer un état, il suffit d'ajouter un champ Signature dans l'état (depuis le ruban, comme n'importe quel champ).

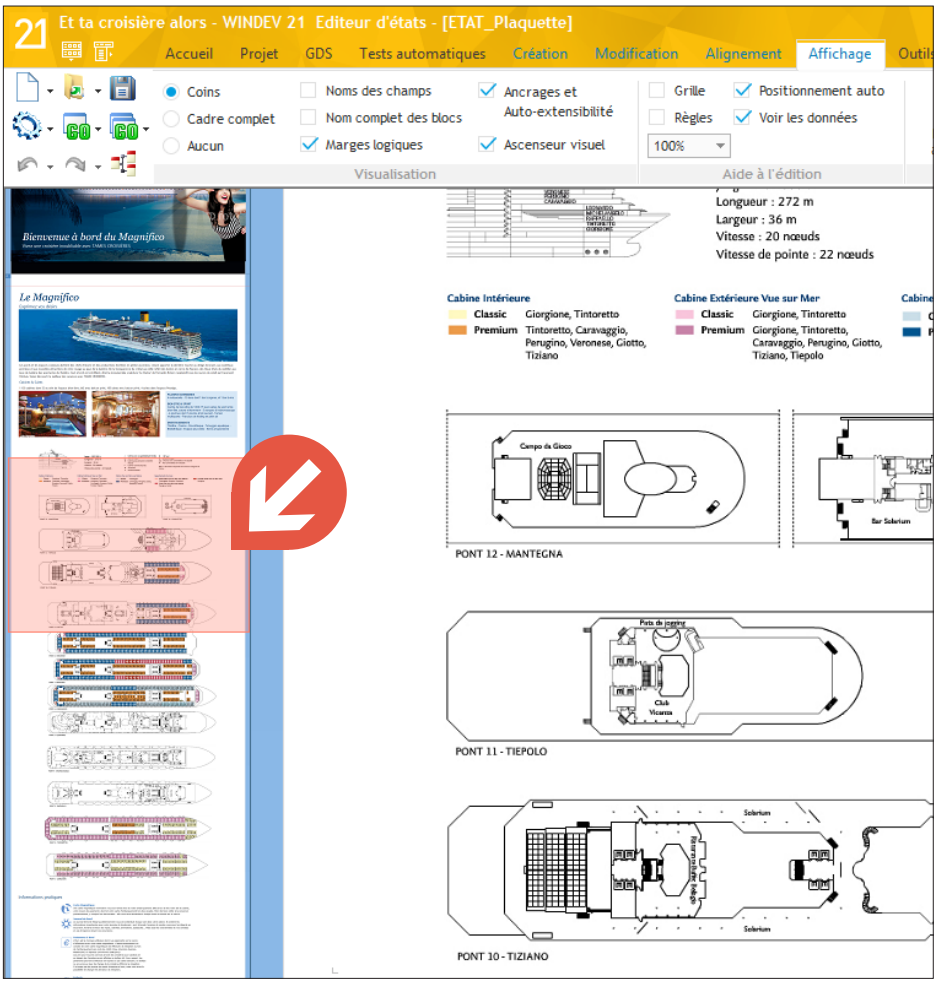
Il existe 2 types de signatures :
• la signature de certification, qui prouve l'identité de l'auteur du document et garantit que le document n'a pas été modifié après avoir été signé.
• la signature d'approbation qui permet d'ajouter des informations supplémentaires dans le document (des annotations, des pièces jointes, etc.). **Attention** : il s'agit bien d'ajout et non de modification du document original.

Dans les deux cas, vous devez sélectionner le certificat qui sera utilisé pour signer l'état. Ce certificat doit être renseigné :
• dans la fenêtre de description du champ Signature, onglet "Général",
• ou par programmation à l'aide de la propriété ***..Certificat***.

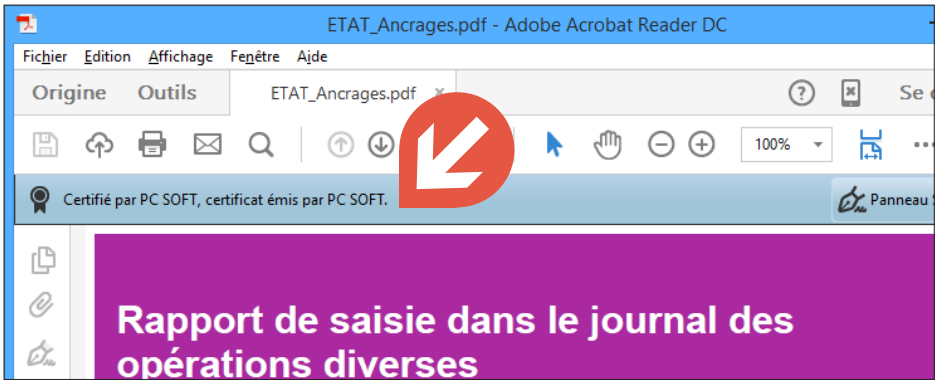
Pour obtenir un certificat, vous avez 2 possibilités :
• acheter un certificat à un organisme reconnu (Symantec, Comodo, Thawte, GeoTrust...). Cette solution apporte plus de confiance à l'utilisateur mais a un coût. Dans ce cas, l'organisme s'engage sur votre identité et garantit que le certificat vous appartient bien.
• créer vous-même votre propre certificat (voir page 61). Cette solution est moins sécurisée (car l'identité n'est pas vérifiée par un organisme tiers) mais est plus simple à mettre en œuvre.

Important : la signature numérique des factures sera obligatoire en France à partir du 1er Janvier 2017. Pensez-y !

■



L'ascenseur visuel offre un confort de navigation sur les états dont la hauteur excède la hauteur de l'éditeur / de l'écran.



Sous Acrobat Reader, la présence de signatures est mise en avant par un bandeau spécifique.

HFSQL : 12 CONSEILS POUR OPTIMISER VOS ACCÈS

HFSQL est la base de données disponible en standard sur WINDEV, WEBDEV et WINDEV Mobile.
Voici 12 conseils efficaces pour optimiser vos accès à une base HFSQL.

1 Passez en HFSQL Client/Serveur

Si votre application est utilisée par plusieurs utilisateurs, utilisez HFSQL Client/Serveur au lieu de HFSQL Classic.

Le serveur HFSQL va donner une priorité aux demandes.

Par exemple, si le serveur se rend compte qu'un traitement prend du temps, il va exécuter en priorité les requêtes unitaires.
De ce fait, vos applications ne sont pas pénalisées par un éventuel gros traitement (batch).

Avec HFSQL Client/Serveur, en plus de la performance, vous bénéficiez également de la sécurité.

Une application multi-utilisateur, dont les données transitent par le réseau, n'est pas à l'abri de micrococupures par exemple.

En mode Client/Serveur, la commande est envoyée entièrement au serveur et c'est le serveur HFSQL qui réalise l'écriture effective.

L'intégrité de votre base de données n'est plus dépendante du réseau.

Notre astuce
Par programmation, vous pouvez tout à fait modifier la priorité d'une application (et donc de ses accès) via la fonction WLangage ***HPriorité***. Plus la priorité est haute, plus le serveur traitera les demandes de l'application en premier.

```
// Positionne une priorité forte
// pour la connexion courante
HPriorité(gcnxMaConnexion, 10)
```

2 Réduisez le nombre d'allers-retours

Lors d'un accès distant, avec une connexion Internet par exemple, la latence (durée du ping) prend souvent du temps.
Cette latence est présente systématiquement à chaque aller-retour.
Si vous avez 1 seconde de latence, c'est 1 seconde de perdue à chaque aller-retour !

Plus vous allez réduire le nombre d'allers-retours, plus vos accès à la base seront rapides.

Remarque : cette latence est beaucoup plus faible sur un réseau local, mais elle est toujours présente.

La meilleure façon (et la plus simple) pour réduire le nombre d'allers-retours est d'utiliser des requêtes.
En effet, l'utilisation d'une requête va générer un seul aller-retour au serveur alors que l'utilisation des fonctions WLangage ***Hxxx*** ou de l'instruction ***POUR TOUT*** directement sur les fichiers de données va générer plusieurs allers-retours.

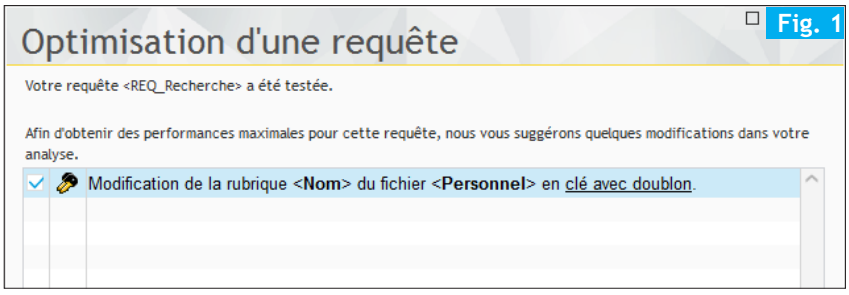
3 Recalculez les statistiques régulièrement

HFSQL se sert de statistiques sur les index pour déterminer la meilleure clé à utiliser en fonction des **données réellement contenues dans le fichier de données**.

Lorsque beaucoup d'ajouts ou de modifications sont faits sur un fichier, ces statistiques ne reflètent plus la réalité : HFSQL peut alors utiliser la meilleure clé théorique mais pas forcément la plus discriminante.

En recalculant régulièrement les statistiques des index de vos fichiers de données, vous êtes sûrs que HFSQL utilise toujours la clé la plus discriminante (et donc, la plus rapide).

Pour recalculer les statistiques d'un fichier, vous pouvez :
• utiliser la fonction ***HAccélèreVitesse*** (ou ***HStatCalcule***),
• demander au serveur HFSQL de recalculer automatiquement ces statistiques de façon régulière. Depuis le Centre de Contrôle HFSQL, dans le volet de visualisation du serveur, cliquez sur le bouton "Configuration du serveur". Cochez ensuite l'option "Activer le calcul automatique des diverses statistiques sur les clés des fichiers de données du serveur" de l'onglet "Général".



L'optimiseur de requêtes détecte les clés manquantes dans vos fichiers de données et propose des solutions pour que l'exécution de la requête soit optimale.

HFSQL®

Remarque: la réindexation d'un fichier de données (fonction ***HRéindexe***) met également à jour les statistiques du fichier de données.

4 Définissez des clés utiles

Pour vous aider à définir les clés nécessaires dans votre base de données, n'hésitez pas à utiliser l'optimiseur de requêtes.

L'optimiseur de requêtes analyse les requêtes créées avec l'éditeur de requêtes et indique les clés à ajouter pour en optimiser l'exécution.

Pour lancer l'optimiseur de requêtes, depuis le ruban, sous le volet "Projet", dans le groupe "Audits et performances", déroulez 'Audit d'édition" et sélectionnez "Optimiser les requêtes du projet". L'optimiseur analysera alors toutes les requêtes du projet.

Remarque: si vous souhaitez analyser une requête précise, ouvrez-la dans l'éditeur de requêtes puis, sous le volet "Requête", dans le groupe "Analyser", cliquez sur "Optimiser la requête".

L'optimiseur de requêtes analyse la requête et déduit les clés qu'il faudrait rajouter dans le fichier pour optimiser la requête [**Fig. 1**].

Vous pouvez ajouter les nouvelles clés directement depuis l'optimiseur de requêtes.

EXPLAIN

Si une requête n'est pas aussi rapide que prévue, n'hésitez pas à utiliser le EXPLAIN.

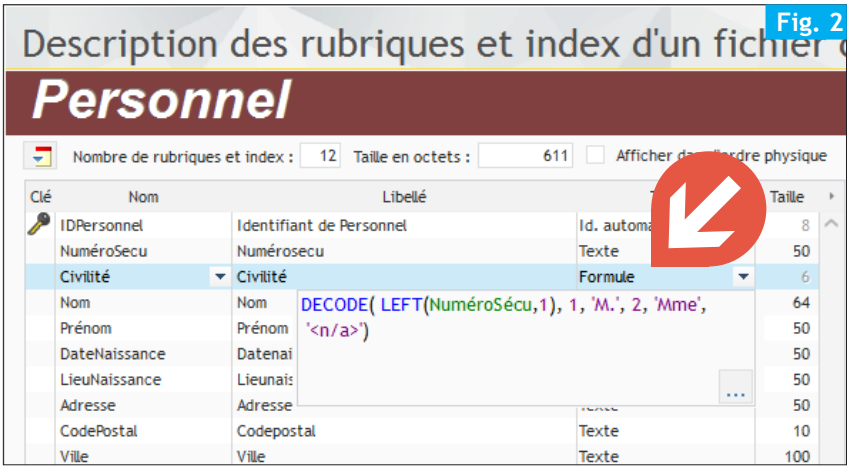
Le mot-clé EXPLAIN permet de récupérer le plan d'exécution qui est utilisé par le moteur HFSQL pour exécuter une requête.

Ce plan d'exécution dépend des données réelles de la base de données.

Pour tester le EXPLAIN sur une requête:
- ouvrez le Centre de Contrôle HFSQL,
- positionnez-vous sur la base de données accédée par la requête,
- dans le ruban, groupe "Requête SQL", cliquez sur le bouton "Nouvelle requête",
- collez le texte SQL de votre requête,
- dans le ruban, groupe "Autres actions", cliquez sur le bouton "Afficher le EXPLAIN".

Le Centre de Contrôle HFSQL affiche une représentation visuelle du résultat du EXPLAIN. En cliquant sur chaque élément, vous aurez plus de détails sur le parcours réalisé et les clés utilisées.

Fig. 2



Une rubrique de type "Formule" permet d'obtenir automatiquement un résultat d'opérations. Ici, la civilité est calculée selon le numéro de sécurité sociale.

5 Rubriques calculées (formule)

Les rubriques calculées permettent d'éliminer tous les traitements accessoires de vos applications.

Les rubriques calculées permettent par exemple d'éviter les "calculs" a posteriori:
- pour déterminer la rubrique "Nom et Prénom" (concaténation des rubriques "Nom" et "Prénom"),
- pour fixer le prix TTC,
- pour calculer la surface d'une pièce en fonction de sa largeur et de sa longueur.

Pour créer une rubrique calculée, il vous suffit d'ajouter une rubrique de type Formule dans votre fichier de données: vous devez alors spécifier la formule de calcul [**Fig. 2**].

Dans cette formule, vous pouvez utiliser et combiner:
- des fonctions SQL,
- des fonctions WLangage (pour des formules encore plus complexes).

Vous pouvez également demander à stocker le résultat du calcul dans le fichier de données. Si vous stockez la valeur, l'écriture dans le fichier sera ralentie (puisqu'il faut calculer la valeur à l'écriture de l'enregistrement), mais les lectures seront accélérées.

Enfin, vous avez la possibilité d'indexer la rubrique calculée pour effectuer des recherches.

HFSQL : 12 CONSEILS POUR OPTIMISER VOS ACCÈS (SUITE)

Fichiers HFSQL & iTunes

HFSQL est bien entendu disponible pour vos applications iOS (iPhone / iPad). Lors de la génération de l'application, vous pouvez demander à ce que les fichiers HFSQL puissent être récupérés et modifiés depuis iTunes : dans l'étape "Configuration", il suffit de cocher l'option "Autoriser le partage de fichiers avec iTunes ("iTunes File Sharing")".

Notre astuce

Vous pouvez indexer une rubrique calculée même si elle n'est pas stockée dans le fichier de données. Il est ainsi possible de faire des recherches sur un calcul qui n'a aucun intérêt à être stocké (par exemple, la surface d'une pièce).

Avec les rubriques calculées, le calcul sera toujours identique, quelle que soit l'application qui y accède, puisque c'est HFSQL qui s'en occupe.

Note : il est également possible d'utiliser des vues matérialisées pour réaliser des calculs "par anticipation" et ainsi améliorer les temps de réponse.

6 Optimisez le matériel

L'optimisation des accès passe aussi par une optimisation matérielle : **ne sous-évaluez pas votre serveur** !

Voici quelques points auxquels il faut prêter attention :
• des disques durs dédiés et rapides (en RAID si possible pour la sécurité),
• un processeur multicœurs,
• un système 64 bits (conseillé),
• beaucoup de RAM (64 Go pour un bon serveur est une valeur habituelle).

Pour des performances optimales, évitez d'ajouter d'autres services et applications sur la machine. Votre serveur de bases de données doit **rester uniquement un serveur de bases de données**.

Pour éviter certaines lenteurs, pensez à vérifier que **votre antivirus** n'analyse pas les fichiers HFSQL (.fic, .ndx, .mmo et .ftx). Dans le cas contraire, à chaque accès aux fichiers de données, l'analyse de l'antivirus va pénaliser les performances.

Enfin, évitez d'installer deux instances de serveurs HFSQL sur un même serveur physique. Dans le cas contraire, les ressources physiques (notamment le disque) devront être partagées, ce qui réduira forcément les performances.

7 Déportez les traitements sur le serveur

Si votre application contient des traitements qui agissent uniquement sur la base de données (le calcul de statistiques internes par exemple), utilisez des **procédures stockées** pour déporter ces traitements sur le serveur.

Le traitement sera plus rapide car il bénéficiera de la puissance du serveur et aucun aller-retour réseau ne sera généré.

De plus, si le traitement doit être exécuté régulièrement, vous pouvez définir une tâche planifiée directement sur le serveur HFSQL :
• soit par programmation via la fonction WLangage ***HAjouteTâche***,
• soit depuis le Centre de Contrôle HFSQL.

Si vous souhaitez avoir plus d'informations sur les procédures stockées (création, déploiement, etc.), consultez l'aide en ligne : "http://doc.pcsoft.fr/?3044360".

8 Limitez vos requêtes au strict nécessaire

Pour réduire la bande passante consommée et donc accélérer les performances de vos requêtes, limitez les données retournées par la requête : récupérez **uniquement les rubriques dont vous avez besoin**.

De plus, si vous savez que les données sur lesquelles une requête est basée n'ont pas changé, **ne réexécutez pas la requête inutilement**. Réutilisez l'instance de la requête déjà exécutée !
Pour savoir si une requête a déjà été exécutée, utilisez la fonction WLangage ***HFichierExiste***.

```
// Si la requête qui liste
// les statistiques de la veille
// n'a pas encore été lancée
SI HFichierExiste(...
 REQ_StatistiquesHier)=Faux ALORS
  // Exécute la requête
  HExécuteRequête(...
    REQ_StatistiquesHier)
FIN

// Parcourt les résultats
POUR TOUT REQ_StatistiquesHier
  ...
FIN
```

9 Gérez un cache des données figées

Si votre application utilise souvent certaines données figées (des paramètres par exemple), **évitez de relire inutilement ces informations qui ne sont pas modifiées**.

Lisez les informations nécessaires (au démarrage de l'application par exemple) et stockez-les dans des variables ou conservez-les dans une requête.

10 POUR TOUT / HLitSuivant

Pour parcourir un fichier de données (ou une requête), vous avez plusieurs possibilités en WLangage,

HFSQL®

notamment :
• la syntaxe ***POUR TOUT***,
• les fonctions ***HLitPremier*** et ***HLitSuivant***.

Lorsque vous utilisez la syntaxe ***POUR TOUT***, HFSQL fait automatiquement une sauvegarde et une restauration de contexte : c'est une sécurité très intéressante mais qui peut prendre du temps.

Si vous êtes sûr de ne pas avoir besoin de cette restauration de contexte, les fonctions ***HLitPremier / HLitRecherchePremier*** et ***HLitSuivant*** peuvent accélérer le traitement.

11 Supprimez les lectures inutiles

Si vous souhaitez uniquement savoir si un enregistrement existe ou non, utilisez la fonction WLangage ***HRecherche*** (ou ***HRecherchePremier***) plutôt que ***HLitRecherche*** (ou ***HLitRecherchePremier***).

En effet, ***HRecherche*** ne lit pas les données mais **parcourt uniquement l'index** du fichier de données (pour savoir si l'enregistrement demandé existe).

```
// Recherche et lit l'enregistrement
HLitRecherchePremier(Produit,
  Référence, "REF-005")
SI HTrouve(Produit) ALORS
 ...
FIN

// Recherche l'enregistrement
// uniquement
HRecherchePremier(Produit,
  Référence, "REF-005")
SI HTrouve(Produit) ALORS
  ...
FIN
```

12 Localisation et .REP

Il est possible d'intégrer des fichiers de données HFSQL directement dans la bibliothèque (.WDL) d'un exécutable. De ce fait, lors de l'accès à un fichier de données par programmation, ce fichier est d'abord recherché dans l'exécutable. Si cette recherche ne trouve pas les fichiers HFSQL intégrés, ceux-ci sont alors recherchés à l'emplacement spécifié par la fonction WLangage ***HChangeRep***.
De ce mécanisme découlent deux optimisations importantes pour toutes les applications qui exploitent traditionnellement les données sur disque :

1. La principale optimisation concerne la recherche des fichiers de données dans l'exécutable. En fonction du nombre d'éléments du projet, la recherche des fichiers de données peut être coûteuse en temps. Il est donc possible d'indiquer à HFSQL Classic qu'il ne doit pas rechercher les fichiers HFSQL dans l'exécutable, grâce à la fonction ***HChangeLocalisation*** :

```
// Modifie le mode de recherche des
// fichiers de données
// (hDisque: les fichiers sont
// recherchés sur le disque)
HChangeLocalisation("*", hDisque)
```

Le gain n'est pas significatif pour les projets ayant une centaine d'éléments. Il devient en revanche très intéressant dès que le projet possède au moins 1000 éléments.

2. La seconde optimisation concerne la conservation de la localisation des données utilisées via le fichier .REP.
Dans le cas d'une application utilisant plusieurs jeux de données, l'accroissement progressif de la taille du .REP peut impacter les temps d'accès.

Il est donc possible d'optimiser les accès en supprimant régulièrement du .REP les emplacements obsolètes et/ou en inhibant sa gestion par ***HGèreRep*** lorsque c'est possible.

■

Restez informé, suivez nos blogs !
Le Support Technique publie régulièrement des astuces sur le blog de PC SOFT: "http://blogs.pcsoft.fr/".

Une section de ce blog est dédiée tout spécialement à HFSQL et ses performances : "http://blogs.pcsoft.fr/billets.awp?blog=hfsql_performances".

GÉNÉRATEUR DE GABARITS : VOTRE CHARTE GRAPHIQUE

Le générateur de gabarits vous permet de créer votre propre charte graphique aux couleurs de votre entreprise ou de celles de vos clients. Cette fonctionnalité est particulièrement utile lors du développement de nombreux projets pour unifier la charte graphique des applications.

Rappels

Un gabarit, c'est quoi ?

Un gabarit WINDEV (ou WINDEV Mobile) contient toutes les couleurs, polices et images qui sont utilisées dans les styles des champs.

Pour chaque type de champ (libellé, saisie, bouton...), un gabarit propose différents styles.

Pour la majorité, ces styles sont communs à chaque gabarit : si vous êtes amené à changer le gabarit d'une fenêtre (ou de tout un projet), WINDEV appliquera automatiquement le style de même nom disponible dans le nouveau gabarit.
De cette façon, vous conservez une cohérence dans l'IHM.

Comment appliquer un gabarit ?

À la création puis dans la fenêtre de description du projet, vous pouvez sélectionner le gabarit par défaut du projet.
Ce gabarit sera automatiquement sélectionné lorsque vous créerez une nouvelle fenêtre.
Ce gabarit par défaut peut être modifié depuis la fenêtre de description du projet, onglet "Style".

Pour appliquer un gabarit, vous avez plusieurs possibilités :

1. Le gabarit doit être appliqué à toutes les fenêtres du projet : dans le ruban, sous le volet "Projet", dans le groupe "Autres actions", cliquez sur le bouton "Appliquer un gabarit au projet". Choisissez le gabarit ainsi que les paramètres d'application et validez.

2. Le gabarit doit être appliqué à une fenêtre uniquement : dans le ruban, sous le volet "Fenêtre", dans le groupe "Styles et gabarits", cliquez sur le bouton "Appliquer un gabarit". La fenêtre d'application du gabarit est la même que pour l'application globale au projet. Vous pouvez également modifier le gabarit d'une fenêtre depuis sa description, onglet "Style", option "Gabarit en cours".

3. Le gabarit doit être appliqué dynamiquement, c'est-à-dire par programmation, à toutes les fenêtres ou à une fenêtre spécifique : il suffit d'utiliser la fonction WLangage ***ChangeGabarit***, en précisant :
• l'élément à modifier (nom de la fenêtre, ...),
• le gabarit à appliquer.

```
// Utilisation du gabarit "Cochet"
// pour la fenêtre en cours
ChangeGabarit(MaFenêtre..Nom,
  "Cochet.WDY")
```

Le générateur de gabarits

Le générateur de gabarits vous permet de créer un gabarit WINDEV ou WINDEV Mobile complet, en personnalisant les couleurs, les polices et les images.

Pour lancer le générateur de gabarits, depuis le ruban, sous le volet "Outils", dans le groupe "Gabarits", cliquez sur "Générateur de gabarits".

Attention : aucun projet ne doit être ouvert sous l'éditeur car le générateur effectue de nombreuses opérations (compilations, synchronisation des styles et des modèles,...).

La génération du gabarit se déroule en 3 étapes. N'hésitez pas à consulter la documentation en ligne pour plus détails sur la génération d'un gabarit, étape par étape : "http://doc.pcsoft.fr/?2016024".

1 Générez les sources de votre gabarit

La première étape dans la génération d'un gabarit consiste à créer le répertoire "source" du gabarit.

Vous définissez le nom du gabarit à générer et l'assistant crée automatiquement le répertoire source, contenant toutes les images et tous les modèles (de fenêtres, d'états...) nécessaires à la génération du gabarit final.

2 Modifiez vos images

Avant de générer réellement le gabarit, vous pouvez personnaliser les fichiers source, c'est-à-dire :
• les différentes images utilisées pour définir le style des champs dans le gabarit,
• les modèles de fenêtres,
• les modèles d'états.

Pour connaître le rôle de chaque image et savoir comment les modifier correctement, nous vous invitons à consulter les pages de documentation suivantes :
• "http://doc.pcsoft.fr/?2016025" (WINDEV),
• "http://doc.pcsoft.fr/?2016026" (WINDEV Mobile).

3 Générez votre gabarit

Lorsque vous avez modifié tous les modèles et images souhaités, relancez le générateur de gabarits depuis l'éditeur et sélectionnez l'option "Générer un gabarit personnalisé".

Sélectionnez ensuite le gabarit à générer. Si votre gabarit n'est pas listé automatiquement, vous pouvez le sélectionner manuellement sur disque à partir du bouton "Charger un gabarit depuis le disque...".

L'assistant se déroule ensuite en 6 étapes, permettant de personnaliser les couleurs de :
• texte,
• fond,
• cadre,
• menu,
• état,
• ...

Pour personnaliser une couleur, il suffit de cliquer sur le carré pour ouvrir le sélecteur de couleurs.

Lorsque vous avez terminé de personnaliser

les couleurs du gabarit, cliquez simplement sur le bouton "Terminer" (la flèche verte) dans la dernière étape de l'assistant.

Le gabarit est généré et automatiquement copié dans le sous-répertoire "\Personnel\Gabarits\<WD/WM>\" du produit.

Le gabarit peut alors être immédiatement utilisé dans vos projets.
Vos gabarits personnalisés sont listés en premier dans les fenêtres de choix de gabarits.

Nos astuces

Se conformer aux images initiales

Si vous remplacez une image, celle-ci doit avoir exactement les mêmes dimensions et le même contenu.

Par exemple, l'image "Btn_Menu.png" correspond à une planche de 5 états (5 boutons en largeur) avec 6 animations (6 boutons en hauteur).

Si vous changez cette image, assurez-vous que votre image correspond à une planche de 5 par 6.

Laisser le générateur créer les états / animations

Pour les images qui doivent être en 5 états ("Pict_Close_16_5" par exemple), il est possible de demander au générateur de travailler pour vous.

Par exemple, si vous disposez d'une image unique (dans son état "normal") et que vous souhaitez avoir une variante légère de l'image pour chaque état, voici une astuce bien pratique :

• créez un répertoire "5 etats" dans le répertoire source du gabarit (s'il n'existe pas encore),

• copiez votre image, qui doit porter exactement le nom attendu (par exemple "Pict_Close_16_5.png").

À la génération du gabarit, le générateur va automatiquement créer les 4 états manquants en appliquant des effets légers sur chacun (comme le fait le catalogue d'images lorsque vous sélectionnez une image pour un bouton).

Par exemple, l'image ✕ sera générée comme cela : ✕✕✕✕✕.

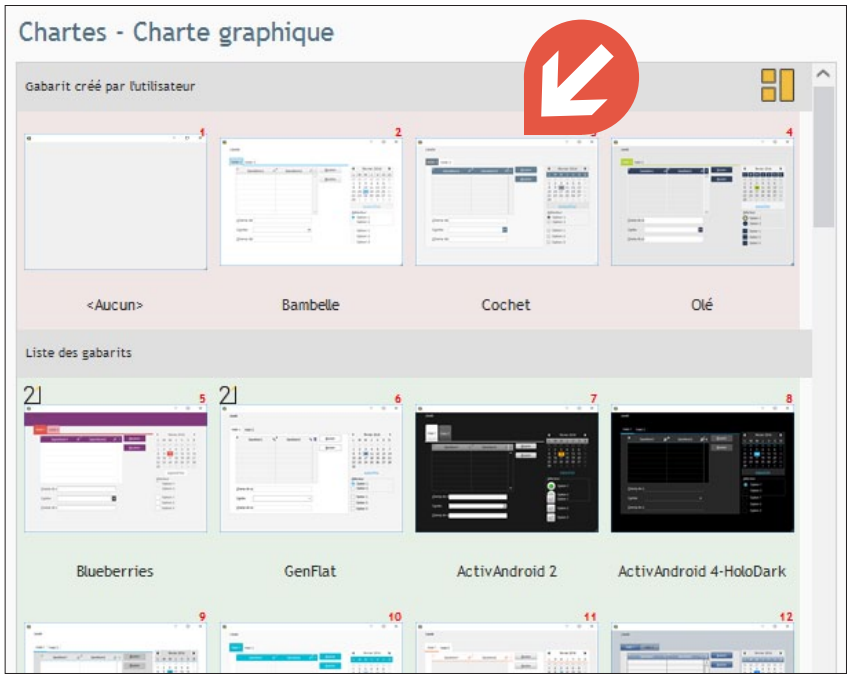
De la même manière, pour les images qui doivent être en 5 états et animées ("Btn_Media_OK" par exemple), il suffit de copier l'image simple (sans état, sans animation) dans un répertoire "5 etats_anim".

Modèles et images

Pour éditer un modèle de fenêtres ou d'états, droppez le fichier .wdt ou .wte directement dans WINDEV ou WINDEV Mobile, sans projet ouvert.

Si vous souhaitez ajouter une image à votre modèle, copiez cette image dans le répertoire "Fond fenêtre" (pour un modèle de fenêtres) ou "Etat" (pour un modèle d'états) du répertoire source du gabarit.

■



21 FONCTIONS WLANGAGE UTILES AU QUOTIDIEN

Le WLangage est un langage riche qui dispose de nombreuses fonctions utiles à tous vos projets. Le WLangage vous permet de développer plus vite, en ayant un code plus compact.
Voici une sélection de fonctions, nouvelles ou méconnues, qui vous seront bientôt indispensables.

Projets d'illustration

Le projet utilisé pour illustrer ce sujet est "WD WLangage TDF21" (présent sur le DVD).

Passer d'une chaîne à un tableau

La fonction WLangage ***ChaîneVersTableau*** permet de découper des parties d'une chaîne selon un séparateur donné et de mettre chaque partie dans un tableau.

```
sMaChaîne est une chaîne = ...
  "1,45,3,56,2,42,51,36"
tabMonTableau est un tableau
  d'entier
// Découpe la chaîne pour mettre
// chaque nombre dans le tableau
ChaîneVersTableau(sMaChaîne,
  tabMonTableau, ",")
```

La fonction WLangage ***TableauVersChaîne*** est la fonction réciproque : elle permet de concaténer tous les éléments d'un tableau dans une chaîne avec un séparateur donné.

```
tabMonTableau est un tableau
 d'entier = [1,45,3,56,2,42,51,36]
sMaChaîne est une chaîne
// Concatène les éléments du tableau
// en les séparant par ","
sMaChaîne = TableauVersChaîne(...
  tabMonTableau, ",")
```

Ces deux fonctions sont très utiles si vous avez besoin de trier des éléments contenus dans une chaîne : il vous suffit de transformer la chaîne en tableau, de trier le tableau (via la fonction WLangage ***TableauTrie***) puis de recréer une chaîne à partir du tableau trié.

Les fonctions WLangage ***ChaîneVersTableau*** et ***TableauVersChaîne*** fonctionnent également avec un tableau de structures.

```
// Structure contenant 2 membres
STPersonne est une Structure
        sNom est une chaîne
        sPrénom est une chaîne
FIN
// Tableau de structures
tabPersonnes est un tableau de
        STPersonne
// Chaîne qui contient une liste de
// personnes (Nom<TAB>Prénom)
// séparées par RC
sListe est une chaîne = ...
  "DUPONT" +TAB+ "Jean" +RC+
  "DURAND" +TAB+ "Paul"
// Remplit le tableau de structures
// à partir de la chaîne
ChaîneVersTableau(sListe,
  tabPersonnes, RC, TAB)
```

Échanger des éléments d'un tableau

La fonction WLangage ***EchangeLigne*** (ou ***TableauEchangeLigne***) permet d'échanger deux lignes dans un tableau à deux dimensions.

```
MonTableau est un tableau de
        3 par 2 chaînes
MonTableau[1,1] = "A"
MonTableau[1,2] = "a"
MonTableau[2,1] = "B"
MonTableau[2,2] = "b"
MonTableau[3,1] = "C"
MonTableau[3,2] = "c"
// Échanger la ligne 1 avec la ligne 3
EchangeLigne(MonTableau, 1, 3)
// Le tableau contient les éléments :
// "C", "c"
// "B", "b"
// "A", "a"
```

Cette fonction vous évite de déclarer une variable intermédiaire pour faire l'échange.
Remarque : si votre tableau est un tableau à 1 dimension et contient uniquement des types simples (entiers, chaînes,...), vous pouvez utiliser l'opérateur d'échange **<=>** (inférieur égal supérieur).

```
tabMonTableau est un
  tableau de chaînes = ...
    ["Marc","Manu","Loreleï"]
// Echange Marc / Manu
tabMonTableau[1] <=>
  tabMonTableau[2]
// Le tableau contient les éléments :
// "Manu", "Marc", "Loreleï"
```

Créer un fichier à une taille donnée

La fonction WLangage ***fChangeTaille*** permet de changer la taille d'un fichier externe.

```
// Crée un fichier
soit nIdentifiant = fCrée(...
  "C:\MonFichier.tmp")
// Change la taille du fichier à 64 Ko
fChangeTaille(nIdentifiant, 64*1024)
// Ferme le fichier
fFerme(nIdentifiant)
```

Cette fonction peut être utile pour **réserver l'espace disque** nécessaire avant le téléchargement d'un gros fichier par exemple : vous créez le fichier local à la taille attendue et vous le téléchargez.

Surveiller un répertoire

La fonction WLangage ***fSurveilleRépertoire*** permet d'être averti (via une procédure callback) lors de la modification du contenu d'un répertoire (création, modification, renommage ou suppression de fichiers).
Cette fonction attend en paramètres :
• le répertoire à surveiller,
• la procédure callback à exécuter en cas de modification,
• les changements sur lesquels la procédure

doit être exécutée,
• la prise en compte des sous-répertoires.

```
// Surveille le répertoire
// "C:\Mes Images\"
// Toute suppression d'un fichier
// dans le répertoire
// (ou un sous-répertoire) entraînera
// l'appel de la fonction
// ControleSuppression
fSurveilleRépertoire(...
  "C:\Mes Images\",
  ControleSuppression,
  fsSuppressionFichier, Vrai)
```

Remarque : la fonction WLangage ***fSurveilleFichier*** fonctionne sur le même principe et permet de détecter la modification d'un fichier donné.

Obtenir un identifiant unique

La manipulation d'identifiant unique est un problème récurrent en programmation.

La fonction WLangage ***DonneGUID*** retourne un GUID, c'est-à-dire un identifiant globalement unique.

```
clMonObjet est un CObjet
// Donne un identifiant unique
// à l'objet
clMonObjet.Identifiant = ...
  DonneGUID()
```

L'algorithme utilisé vous garantit que, quelle que soit la machine ou quelles que soient les conditions de l'appel, la fonction ne retournera jamais deux fois la même valeur.

Le WLangage dispose d'une autre fonction similaire lorsque vous avez besoin d'identifiants uniques mais seulement sur la durée d'exécution de votre application : ***DonneIdentifiant***.

Comparer des fichiers grâce au hash

Le hachage d'un fichier permet de réaliser un résumé condensé d'un fichier (ou d'une chaîne de caractères), comme une empreinte. Cette empreinte est appelée "hash".

La fonction WLangage ***HashFichier*** permet de calculer le hash d'un fichier.

```
// Calcule le hash du fichier
sHash est une chaîne = ...
  HashFichier(HA_MD5_128,
  "C:\Mes documents\MonFichier.txt")
```

Cette fonctionnalité est très pratique pour savoir si deux fichiers sont identiques (ou si un fichier n'a pas été modifié) : il suffit de comparer les hash des deux fichiers (ou d'utiliser la fonction WLangage ***HashVérifieFichier*** pour comparer directement un fichier à un hash donné).

Les fonctions WLangage ***HashChaîne*** et ***HashVérifieChaîne*** permettent de manipuler le hash d'une chaîne.

Ajouter une liste de fichiers dans une archive

Le WLangage propose plusieurs fonctions pour manipuler facilement des archives.

Si vous souhaitez ajouter simultanément plusieurs fichiers à une archive, la fonction WLangage ***zipAjouteListeFichier*** est faite pour vous !

```
sListeFichier est une chaîne
// Liste des fichiers à ajouter
// à l'archive
sListeFichier =
  "C:\Mes Documents\CGV.pdf"
sListeFichier += [RC] +
  "C:\Produits\Nouveautés.pdf"
sListeFichier += [RC] +
  "C:\Mes Images\Bonus.jpg"
...
// Ajoute tous les fichiers à l'archive
zipAjouteListeFichier("MonZip",
  sListeFichier)
```

Important : la fonction ***zipAjouteListeFichier*** est beaucoup plus performante qu'une suite d'appels à la fonction ***zipAjouteFichier***. Privilégiez donc l'utilisation de la fonction ***zipAjouteListeFichier*** .

Signer un fichier avec un certificat

Si vous distribuez un fichier et que vous souhaitez garantir à vos clients que le fichier est bien le vôtre et qu'il n'a pas été modifié quand ils le recevront, vous devez signer votre fichier à l'aide d'un certificat.

Pour cela, le WLangage dispose de fonctions ***CertificatXXX*** simples et pratiques :
• ***CertificatSigneFichier*** qui permet de signer un fichier.

```
// Fichier à signer
sMonFichier est une chaîne = ...
  "C:\MonFichier.pdf"
// Choix du certificat à utiliser parmi
// les certificats de Windows
MonCertificat est un Certificat = ...
  CertificatSélecteur()
// Signe le fichier avant transmission
bufSignature est un Buffer =...
  CertificatSigneFichier(...
    sMonFichier, MonCertificat)
```

• ***CertificatVérifieFichierSignatureXXX*** qui permet de vérifier l'origine et l'intégrité d'un fichier (à partir d'un buffer ou d'un fichier).

```
// Vérifie la signature
SI CertificatVérifieFichierSignature-
```

21 FONCTIONS WLANGAGE (SUITE)

```
Buffer(sMonFichier, bufSignature)
        <> certificatOk ALORS
  // La signature est incorrecte
  ...
FIN
```

Récupérer la liste des pays

La fonction WLangage ***DonneListePays*** renvoie la liste de tous les pays du monde, avec pour chaque pays :
• le nom du pays,
• ses différents codes ISO,
• le code correspondant pour la fonction ***Nation***.

```
// Récupère la liste des pays
// (libellé et constante nationXXX
// associée)
sListePays est une chaîne = ...
  DonneListePays(...
    lpLibellé+lpNation)
```

```
// Parcourt les pays
POUR TOUTE CHAÎNE sUnPays
    DE sListePays SEPAREE PAR RC
  // Ajoute le pays à la combo
  ListeAjoute(COMBO_Pays,
  ExtraitChaîne(sUnPays,1) +
  gLien(ExtraitChaîne(sUnPays,2)))
FIN
```

Créer une forme polygonale

La fonction WLangage ***FormePolygone*** permet de transformer un champ Forme en une forme libre, définie par une suite de points. Associée au champ Fil, cette fonctionnalité permet de dessiner à peu près n'importe quel diagramme, organigramme ou workflow.

```
tabPoints est un tableau de Point
// Définit les points délimitant
// la forme
Ajoute(tabPoints, [500,0])
Ajoute(tabPoints, [600,400])
Ajoute(tabPoints, [1000,500])
Ajoute(tabPoints, [600,600])
Ajoute(tabPoints, [500,1000])
Ajoute(tabPoints, [400,600])
Ajoute(tabPoints, [0,500])
Ajoute(tabPoints, [400,400])
// Applique la forme au champ
FormePolygone(...
  FRM_Polygone,tabPoints)
```

Attention
La forme polygonale est incluse dans le champ Forme initial.
Les positions X et Y sont donc relatives à la taille du champ et sont exprimées en 1/1000.
Par exemple :
• le point situé au centre du bord haut a pour coordonnées "500, 0".
• le point central a pour coordonnées "500, 500".

Instruction *COMPILE SI* : votre nouvel allié des projets multi-configurations !

La compilation conditionnelle (instruction ***<COMPILE SI>***) permet de définir des portions de code qui seront compilées uniquement pour une configuration ou un type de configuration donné :

```
<COMPILE SI
TypeConfiguration=Android>
  // Code à compiler en Android
  ...
<SINON>
 // Code à compiler
 // dans les autres plateformes
  ...
<FIN>
```

La compilation conditionnelle permet d'éviter l'affichage d'erreurs de compilations comme dans le code-cible conditionnel : les subtilités propres à chaque plateforme (Android, iOS) peuvent être codées sans provoquer d'erreurs de compilation dans les autres configurations.

Si vous devez mettre en place des spécificités de code par configuration, privilégiez la compilation conditionnelle.

Le code-cible conditionnel est intéressant pour la création d'un composant qui doit être utilisé dans différentes configurations. Dans ce cas, il est en effet nécessaire que TOUT le code soit compilé à la génération du composant, pour qu'il puisse être exécuté quelle que soit la plateforme d'exécution.

Notre astuce

La compilation conditionnelle autorise également l'utilisation de constantes comme condition de compilation.
Vous pouvez :
• définir une constante (et sa valeur) dans un ***COMPILE SI*** "classique" (dont la condition concerne une configuration ou un type de configuration)

```
-- Initialisation du projet
// Définit la valeur de la constante
// MODE_INTERNE selon la
// configuration courante
<COMPILE SI Configuration =
                "Application Mailing">
 CONSTANT MODE_INTERNE = Faux
<FIN>
```

```
<COMPILE SI Configuration =
        "Application Robot">
 CONSTANT MODE_INTERNE = Vrai
<FIN>
...
```

• puis utiliser la valeur de la constante directement dans un ***COMPILE SI***.

```
// En mode interne, compile le code
<COMPILE SI MODE_INTERNE>
...
<FIN>
```

Si vous définissez différentes constantes, vous pouvez également combiner ces constantes comme condition de compilation.

```
// Le code sera compilé en mode
// service et pas en mode interne
<COMPILE SI MODE_SERVICE ET
        PAS MODE_INTERNE>
  ...
<FIN>
```

Le Saviez-Vous ?

Toutes plates-formes confondues (Windows, Linux, Android, iOS, Universal Windows 10 App, etc.), le WLangage comporte **plus de :**

- **3500** fonctions,
- **800** propriétés,
- **250** types de variable et mots.

Pensez toujours à rechercher dans l'aide en ligne : la fonction dont vous avez besoin existe déjà certainement !

L'aide en ligne, c'est par là : "http://doc.pcsoft.fr/".

Manipuler une planche d'images

Les fonctions ***ChargeImage*** et ***ChargePlancheImage*** permettent de charger des images en mémoire. Ces images peuvent ensuite être manipulées ultérieurement via la fonction WLangage ***gImage***.

En version 21, la fonction WLangage ***gImageExiste*** permet de savoir si une image (ou une planche d'images) a déjà été chargée en mémoire.

```
// Charge la planche
// d'images IDIMAGELOGO
// si elle n'a pas encore été chargée
SI gImageExiste(IDIMAGELOGO) =
                    Faux ALORS
  // Charge l'image
  ChargePlancheImage(...
    IDIMAGELOGO,
    "Logos.png")
FIN
// Utilise l'image chargée
LIB_Titre = "Notre logo : " +
  gImage(IDIMAGELOGO, 1)
```

Manipuler des enregistrements via JSON

Si vous développez des webservices REST qui envoient des données en JSON, voici 3 fonctions WLangage complémentaires et très utiles :

• ***HEnregistrementVersJSON*** permet de convertir, en un appel, un enregistrement d'un fichier de données HFSQL en une structure JSON prête à l'emploi :

```
// Récupère les rubriques de
// l'enregistrement Produit en cours
// au format JSON
sJSON est une chaîne = ...
 HEnregistrementVersJSON(Produit)
```

La variable ***sJSON*** contient alors une chaîne de ce type :

```
{ "Produit":
{ "Reference":"REF-42",
"Libelle":"La r\u00e9ponse ultime",
 "IDProduit":1, "PrixHT":51 }}
```

• ***HImporteJSON*** permet de convertir un fichier HFSQL complet depuis le format JSON :

```
// Complète le fichier Produit avec
// les enregistrements du fichier
// JSON
HImporteJSON(Produit,
  "C:\Temp\Mesproduits.json")
```

• ***HExporteJSON*** permet de convertir un fichier HFSQL complet vers le format JSON :

```
// Exporte le fichier Produit complet
// au format JSON
HExporteJSON(Produit,
  "C:\Temp\Mesproduits.json")
```

Associer une application à une extension

Si votre application utilise des fichiers aux extensions personnalisées, il peut être pratique d'associer ces extensions directement à votre application.
De cette façon, le double-clic sur un fichier lancera automatiquement votre application.

La fonction WLangage ***SysAssocieExtension*** vous permet de réaliser cette opération en une seule ligne code.

```
// Associe l'extension .tames à
// l'application courante
// Lors d'un double-clic sur un fichier
// .tames, l'application sera lancée
// avec le fichier en ligne de
// commande
// Cette association est effective
// uniquement pour l'utilisateur
// courant
SysAssocieExtension(...
  ".tames",
  ExeInfo(exeNom)+
    " ""%1""",
  sysUtilisateurCourant)
```

Notre conseil
Utilisez cette fonctionnalité directement lors de l'installation de l'application, en ajoutant un plugin à WDSetup par exemple.

Construire une trace

La fonction WLangage ***ChaîneConstruit*** permet de formater facilement une chaîne en fonction de paramètres.
La fonction WLangage ***TraceConstruit*** permet de formater directement une ligne de trace sans avoir à appeler successivement ***ChaîneConstruit*** puis ***Trace***.

```
// Affiche une trace construite
TraceConstruit(...
  "Produit courant : %1 (id = %2)",
  Produit.Libellé, Produit.IDProduit)
```

Remarque : de la même manière, les fonctions WLangage ***InfoConstruit*** et ***ErreurConstruit*** permettent de formater directement une boîte d'information et d'erreur.

Exporter un agenda/planning/rendez-vous au format iCalendar

Les fonctions WLangage ***AgendaVersiCalendar***, ***PlanningVersiCalendar*** et ***RendezVousVersiCalendar*** permettent d'exporter le contenu d'un champ Agenda, d'un champ Planning ou une liste de rendez-vous au format standard iCalendar.

Vous pouvez ensuite utiliser le fichier généré, par exemple :

• en pièce jointe d'un email pour envoyer des rendez-vous à vos clients,
• sur votre site Web pour que vos clients puissent télécharger les dates d'événements à venir,
• ...

```
// Exporte les rendez-vous de ce
// mois-ci au format iCalendar
sListeRDV est une chaîne = ...
  AgendaVersiCalendar( ...
    AGD_Agenda,
    PremierJourDuMois(),
    DernierJourDuMois())
```

Remarque : le WLangage propose également les fonctions réciproques ***iCalendarVersAgenda***, ***iCalendarVersPlanning*** et ***iCalendarVersRendezVous*** pour remplir un champ Agenda, un champ Planning ou un tableau de ***RendezVous*** à partir d'un fichier iCalendar.

■

PCSCLOUD : VOS SOLUTIONS CLOUD

PCSCloud propose des plateformes CLOUD prêtes à l'emploi pour répondre à vos différents besoins :

- partager une base de données HFSQL,
- fournir des sites et des webservices,
- fournir des installations d'applications WINDEV,
- partager un GDS et des Centres de Contrôle,
- héberger un serveur de télémétrie.

Les différents types de plateformes

PCSCloud propose actuellement 3 types de plateformes correspondant à des usages différents :

- les plateformes d'exploitation sont destinées à la mise en production (de bases, de sites,...),
- les plateformes de développement sont destinées au développement d'applications,
- les plateformes de télémétrie sont destinées aux relevés télémétriques des applications WINDEV.

1 Les plateformes d'exploitation

Une plateforme exploitation permet la mise en exploitation d'applications WINDEV ou WEBDEV. Les plateformes exploitation peuvent ainsi héberger :

- des sites WEBDEV,
- des webservices WINDEV,
- des bases de données HFSQL Client/Serveur,
- des installations d'applications WINDEV.

Chacune de ces fonctionnalités peut être activée (ou non) sur la plateforme. Il est possible d'activer toutes les fonctionnalités sur une même plateforme ou de les répartir pour mettre en place une architecture plus souple (une plateforme frontale pour un site et une plateforme pour la base de données par exemple).

Paiement à l'usage

Les plateformes d'exploitation ont un tarif horaire.
Les heures d'utilisation des plateformes sont facturées uniquement si la plateforme est active, autrement dit uniquement quand les applications sont disponibles et accessibles. De même, seul l'espace de stockage occupé est facturé.
Les équipes de développement ou de production n'ont ainsi pas à se préoccuper des problèmes de capacité de stockage ou du matériel.

Licences

Dans ces plateformes, les licences sont incluses : la licence Windows Server comme les licences du serveur d'applications WEBDEV. Les mises à jour intermédiaires et majeures sont également incluses : vous êtes tranquilles !

Important
Toutes les licences sont sans limite : le nombre de sites, de webservices, de bases de données... est illimité.

Quelle plateforme choisir ?

Les plateformes d'exploitation peuvent varier en terme de puissance. Il est ainsi possible de moduler la puissance en fonction de la période de l'année ou de la montée en charge d'un projet.

La puissance de la plateforme augmente pour chaque type de plateforme : Bronze, Argent, Or, Platine ou Diamant. Il est possible à tout moment de changer la puissance de la plateforme. Les caractéristiques de ces plateformes sont consultables sur le site "www.pcscloud.net".

Les plateformes Bronze sont destinées aux tests et à la mise en place initiale du projet. Les autres puissances sont destinées à la mise en production. Lors de la création d'une plateforme, il est donc généralement conseillé de choisir une plateforme Bronze. Mais ensuite ?

Ensuite, la puissance du projet en production peut être estimée :

- en fonction de l'expérience des équipes de développement ou des caractéristiques des plateformes,
- en fonction de la charge relevée sur la ou les plateformes (des graphiques permettent de voir les consommations mémoire et CPU des plateformes),
- via le calculateur en ligne du site "www.pcscloud.net" en répondant à différentes questions sur l'objectif de la plateforme.

Important
Attention à ne pas surévaluer votre besoin de puissance. En effet, sur un serveur classique, ce choix est définitif. En raisonnement Cloud, il est préférable de monter en puissance en fonction du besoin. De même, en cas de baisse saisonnière de consommation, il est intéressant de diminuer la puissance de la plateforme.

2 Les plateformes de développement

Une plateforme de développement permet de partager et d'utiliser à distance un Gestionnaire de Sources (GDS) et des Centres de Contrôle depuis WINDEV, WEBDEV ou WINDEV Mobile.

La plateforme de développement permet également de partager un dictionnaire de traduction (utilisé par WDTrad).

Paiement selon le nombre de développeurs et de traducteurs

La plateforme de développement initiale est destinée à 1 développeur.
Il est ensuite possible d'ajouter à tout moment des accès pour des développeurs ou des traducteurs supplémentaires.

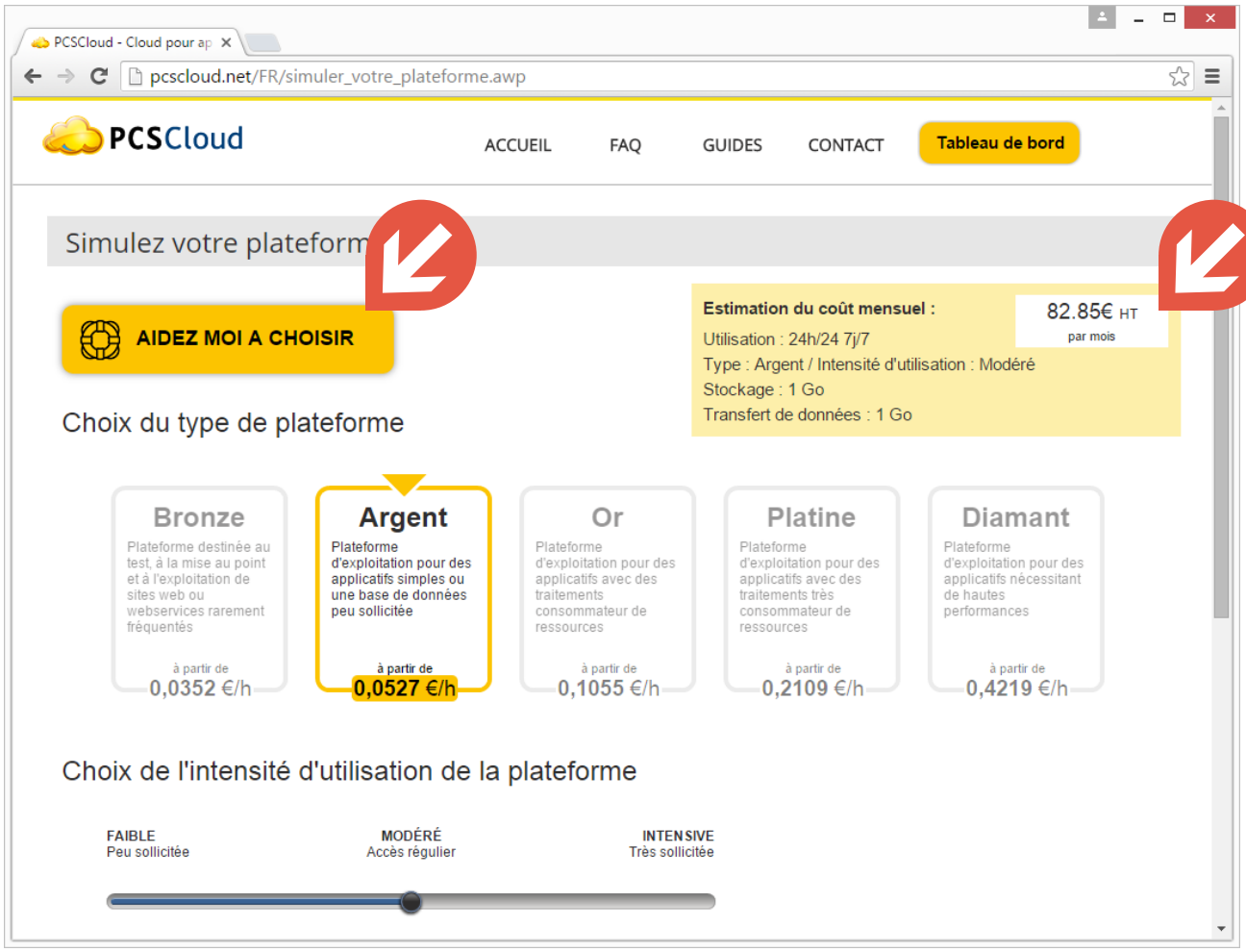
La facturation est mensuelle et proportionnelle au nombre de développeurs/traducteurs.

Partage d'un Gestionnaire de Sources (GDS) et des Centres de Contrôle

Grâce aux plateformes de développement, les équipes peuvent virtualiser leurs projets. Les projets sont toujours accessibles, de n'importe où !

SIMULER VOTRE PLATEFORME
Estimation à titre indicatif

Si vous ne savez pas quelle plateforme choisir, un calculateur vous aide à choisir la plateforme la plus adaptée à vos besoins : il suffit de répondre à quelques questions et vous avez une estimation du coût mensuel !



Et puisque les Centres de Contrôle sont également gérés, les développeurs distants ont accès à toutes les ressources de gestion : messagerie, gestion des exigences, gestion des incidents, planning, etc.

Partage d'un dictionnaire de traduction

Lors d'une traduction d'une application WINDEV, WEBDEV ou WINDEV Mobile, les traductions peuvent être mémorisées dans un dictionnaire. Cela permet ensuite de :

- traduire automatiquement les ressources identiques dans les autres applications ou les versions futures,
- proposer des traductions cohérentes par rapport aux traductions précédentes.

L'utilisation d'un dictionnaire de traduction centralisé permet également de :

- traduire dans une même langue plusieurs interfaces tout en évitant la traduction multiple de ressources identiques,
- traduire automatiquement les prochains projets avant de les fournir aux traducteurs pour limiter les coûts de traduction,
- utiliser des traducteurs indépendants ou des cabinets pour compléter une traduction,
- être propriétaire du dictionnaire de traduction et ainsi de pouvoir changer de traducteur.

3 Les plateformes de télémétrie

Les plateformes de télémétrie disposent d'un serveur de télémétrie accessible par Internet. Le serveur peut gérer plusieurs applications.

Un serveur de télémétrie ?

À partir de la version 21, WINDEV propose des statistiques avancées sur l'utilisation des applications WINDEV grâce à la Télémétrie.

La télémétrie permet aux équipes de développement de tout connaître des applications déployées, ce qui permet de les améliorer facilement.

Vous retrouvez plus de détails sur la télémétrie dans l'article dédié pages 36 à 39.

Une tarification mensuelle simple

La plateforme de télémétrie est facturée au mois, quel que soit le nombre d'utilisateurs, quel que soit le nombre d'applications.

La seule limite est le nombre d'actions collectées par mois pour la plateforme. Au-delà de ce seuil (actuellement 5.000.000 d'actions !), un supplément est appliqué par tranche de million d'actions supplémentaires.

Il est possible de suivre le nombre d'actions en cours soit :

- depuis le tableau de bord PCSCloud de la plateforme,
- depuis le tableau de bord de la télémétrie de WINDEV.

■

TÉLÉMÉTRIE : DES STATISTIQUES PRÉCISES SUR VOS APPLICATIONS

À partir de la version 21, WINDEV propose des statistiques avancées sur l'utilisation de vos applications WINDEV grâce à la télémétrie.
Avec la télémétrie, vous connaissez tout de vos applications déployées et vous pouvez facilement les améliorer.

Présentation

Pour utiliser la télémétrie dans une application, il suffit de l'activer dans l'éditeur de WINDEV, de recompiler l'application et de la déployer sur les postes clients.

La télémétrie fonctionne automatiquement, en tâche de fond : vos utilisateurs ne voient aucune différence.

Automatiquement, la télémétrie collecte les informations sur l'utilisation de l'application : nombre d'utilisateurs, jours de connexion, résolution et nombre d'écrans, actions effectuées (nombre et temps de traitement), etc. Toutes ces données sont automatiquement envoyées à un serveur de télémétrie.

Le serveur de télémétrie agrège toutes les données et consolide les informations dans une base HFSQL.

Après quelques jours/semaines/mois, vous pouvez consulter les résultats directement dans WINDEV, dans le tableau de bord de la télémétrie.

Installation du serveur de télémétrie

Pour utiliser la télémétrie dans vos applications, il est nécessaire de disposer d'un serveur de télémétrie. Deux solutions sont disponibles :
• installer un serveur de télémétrie sur une machine de votre parc,
• utiliser un serveur de télémétrie de PCSCloud.

Installer un serveur

Pour installer un serveur de télémétrie, vous devez lancer l'installation du Serveur Réservé.

Le Serveur Réservé est une version spéciale du Serveur d'Application WEBDEV qui est utilisée pour faire fonctionner différents modules :
• le SILO (Superviseur d'Infrastructures LOgicielle),
• le serveur de réplication,
• le webservice d'accès aux bases tierces,
• le serveur de télémétrie.

L'installation du serveur réservé est disponible dans le sous-répertoire "\Install\WDServeurR\" de WINDEV.
Suivez simplement l'assistant d'installation.

Dans l'étape "Télémétrie : serveur HFSQL", vous devez indiquer :
• les informations de connexion au serveur HFSQL dans lequel toutes les données de la télémétrie seront collectées (nom/adresse et port),
• l'utilisateur administrateur (et son mot de passe),
• le nom de la base de données dédiée à la télémétrie ("TLM" par défaut),
• l'utilisateur dédié à la télémétrie (et son mot de passe). Si cet utilisateur n'existe pas, l'assistant va le créer automatiquement et lui donner les droits nécessaires sur la base.

À la fin de l'installation, l'assistant fournit les paramètres de connexion au serveur de télémétrie. Ces informations devront être renseignées dans WINDEV, gardez-les précieusement.

Votre serveur est maintenant prêt à l'emploi.

Plateforme PCSCloud

Pour faciliter l'utilisation de la télémétrie, vous pouvez utiliser une plateforme de télémétrie de PCSCloud. Il s'agit d'un nouveau type de plateforme.

Depuis le tableau de bord de votre compte PCSCloud, il suffit de :
• créer une nouvelle plateforme de type télémétrie,
• donner un nom à la plateforme,
• sécuriser les accès si besoin, en limitant les adresses IP autorisées. Ce paramétrage est très utile si vous connaissez l'adresse IP de tous les postes clients.
• définir la politique de mise à jour (comme pour toutes les plateformes PCSCloud).

Après validation, la plateforme est créée (le temps de création est de quelques minutes).

Le Saviez-Vous ?

A partir de la version 210051f, le tableau de bord de la télémétrie propose un affichage des **statistiques générales du serveur**. L'onglet "Matériel" est très intéressant puisqu'il regroupe les statistiques matériels (système d'exploitation, nombre et résolutions d'écran, puissance, ...) de tous les utilisateurs.
Cela permet de mieux cibler les nouvelles IHM des nouveaux projets.

Retrouvez plus de détails sur PCSCloud dans l'article pages 34-35.

Configuration

L'activation de la télémétrie pour un projet se fait au niveau de chaque configuration.

En effet, vous pouvez tout à fait activer la télémétrie sur la configuration principale de votre application (celle utilisée par vos clients) mais pas sur vos autres configurations (qui correspondent à des applications "automates" par exemple).

Dans la description de la configuration, affichez l'onglet "Télémétrie" et cochez la case "Activer la télémétrie". Vous pouvez alors :
• indiquer les informations de connexion (serveur "personnalisé" ou plateforme PCSCloud),
• définir le nom de l'application qui apparaîtra dans les rapports de la télémétrie.
• choisir d'activer la télémétrie automatiquement ou manuellement.

Notre astuce
Par défaut, la télémétrie est activée automatiquement : les données seront collectées dès que les utilisateurs se serviront de l'application.
Mais vous pouvez tout à fait activer la télémétrie par programmation via la fonction WLangage ***TélémétrieActive***.
L'activation manuelle peut être pratique si vous ne souhaitez pas utiliser la télémétrie sur certaines parties de l'application (pour des raisons de confidentialité par exemple).

La fonction ***TélémétrieParamètre*** permet de configurer les paramètres de la télémétrie : la taille du buffer de collecte, les délais de transmission, ...

Exécutable / Installation

À la génération de l'exécutable, vous pouvez choisir d'activer ou non la télémétrie (et éventuellement, modifier les informations de connexion au serveur).

Si vous choisissez d'activer la télémétrie, une étape spécifique "Télémétrie" sera affichée dans l'assistant d'installation de votre application (étape 3/3 de "Langues et messages").

Vous pourrez alors choisir d'afficher un message à l'utilisateur au moment de l'installation, qui l'informera que la télémétrie est activée pour cette application.
Il est également possible de proposer à l'utilisateur de désactiver la télémétrie (s'il ne souhaite pas que des données soient collectées depuis son poste).

Visualisation

Le tableau de bord

Pour afficher l'éditeur de télémétrie, sous le volet "Projet", dans le groupe "Performances et audit", cliquez sur "Télémétrie".

Vous avez directement accès à tous les projets sur lesquels la télémétrie est active mais également à des statistiques globales sur le serveur.
Ce tableau de bord (voir page 38) est accessible même si aucun projet n'est ouvert dans WINDEV.

Remarque : si vous visualisez les statistiques d'un projet qui n'est pas ouvert dans votre éditeur, certaines fonctionnalités ne seront pas disponibles : éléments non utilisés, accès aux éléments, télémétrie d'un élément donné, ...

Le tableau de bord affiche les données recueillies sur une période donnée : aujourd'hui, hier, 7 derniers jours, ... Vous pouvez également choisir une période personnalisée.

Le tableau de bord est composé de différents onglets par défaut :
• l'onglet "Synthèse" présente un résumé de l'utilisation de l'application : nombre d'utilisateurs, éléments les plus utilisés, les moins utilisés....
• l'onglet "Utilisation" affiche les statistiques d'utilisation sur les éléments pour une période donnée : éléments les plus utilisés, les moins utilisés, traitements les plus exécutés, les plus longs, ...
• l'onglet "Utilisateurs" affiche le nombre d'utilisateurs de l'application.
• l'onglet "Matériel" contient des détails sur les configurations physiques des utilisateurs (système d'exploitations, RAM, résolution et nombre d'écrans, ...).
• l'onglet "Anomalies" répertorie toutes les erreurs rencontrées lors de l'utilisation de l'application (erreurs WLangage les plus fréquentes, ...).

Pour chaque élément affiché dans les différents widgets, vous pouvez, depuis le menu contextuel :
• afficher l'élément dans l'éditeur,
• afficher les statistiques détaillées de cet élément. Il est ainsi possible d'obtenir le détail des champs et menus les plus et moins utilisés dans la fenêtre.

Lorsque vous affichez les statistiques détaillées d'un élément, un onglet dédié à cet élément s'ajoute aux 5 onglets par défaut et une nouvelle option s'active dans le ruban : le "Go" en mode télémétrie (voir page 39).

Ce "Go" vous permet d'avoir un aperçu en exécution des éléments utilisés.

Ce mode de visualisation permet de voir d'un coup d'œil les champs les plus utilisés avec pour chacun leur temps de traitement moyen.

Autre avantage : vous pouvez naviguer dans votre application pour avoir un aperçu de la télémétrie de chaque fenêtre.

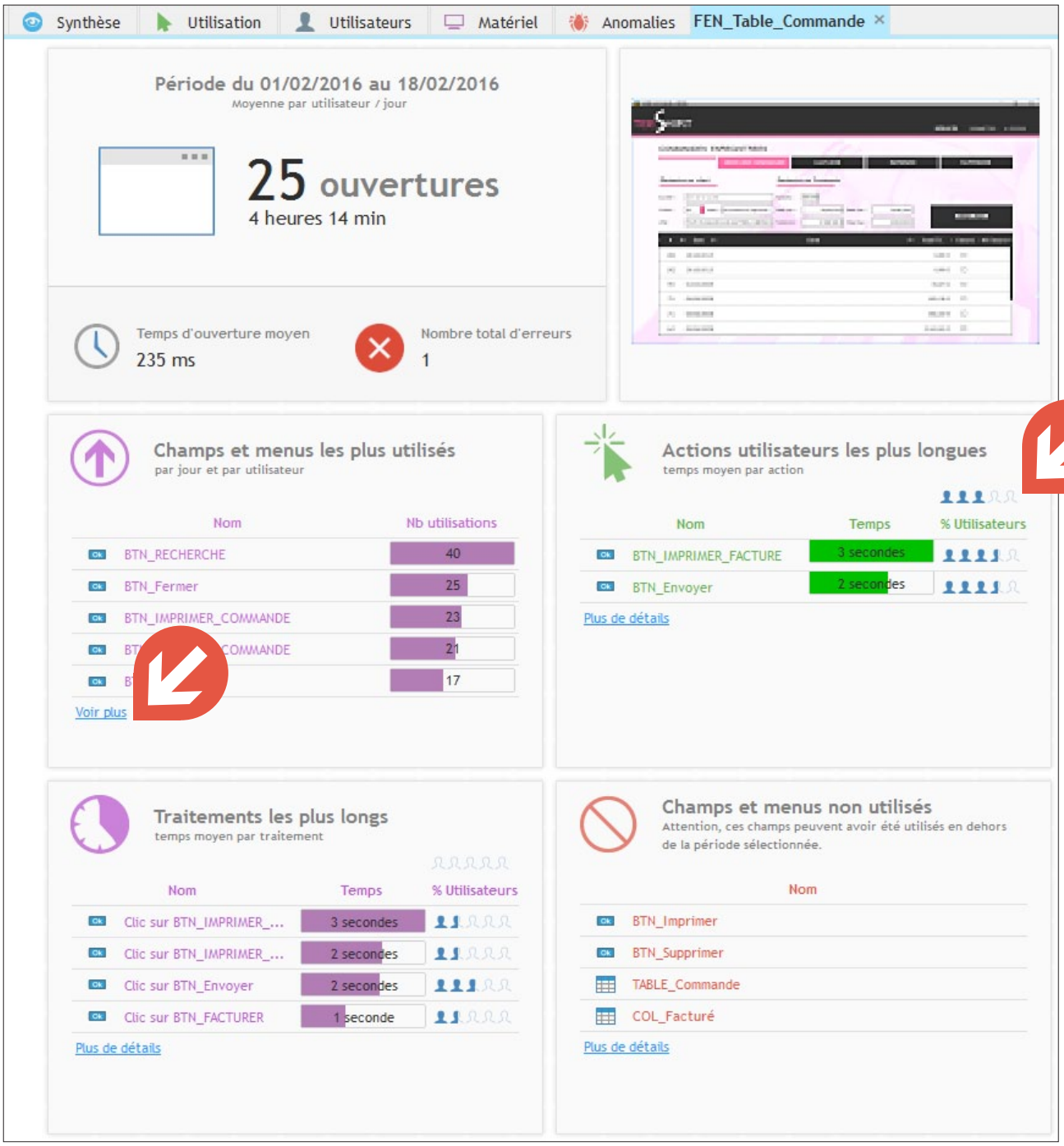
■

TÉLÉMÉTRIE : TABLEAU DE BORD

Le tableau de bord vous permet d'avoir une vision synthétique :
• des utilisateurs, toute application confondue (nouveauté de la version 210051f),
• des statistiques, actions, utilisateurs... d'une configuration de projet,
• des actions, traitements... d'une fenêtre donnée.

Le tableau de bord est découpé en différents volets, chacun regroupant plusieurs widgets dédiés.
Par défaut, certains widgets affichent uniquement les 5 premiers éléments (5 éléments les plus utilisés, 5 éléments les moins utilisées, etc.). Le lien "Voir plus" permet d'afficher la liste complète des éléments pour le widget dans un nouveau volet.

L'indicateur "% Utilisateurs" (disponible sur certains widgets) vous permet de connaître le pourcentage d'utilisateurs concernés par le traitement ou l'élément.
Un bouton vous permet de filtrer rapidement les lignes correspondant à un "% Utilisateurs" donné : c'est très pratique si vous souhaitez concentrer vos efforts sur les éléments manipulés par une majorité d'utilisateurs. ■

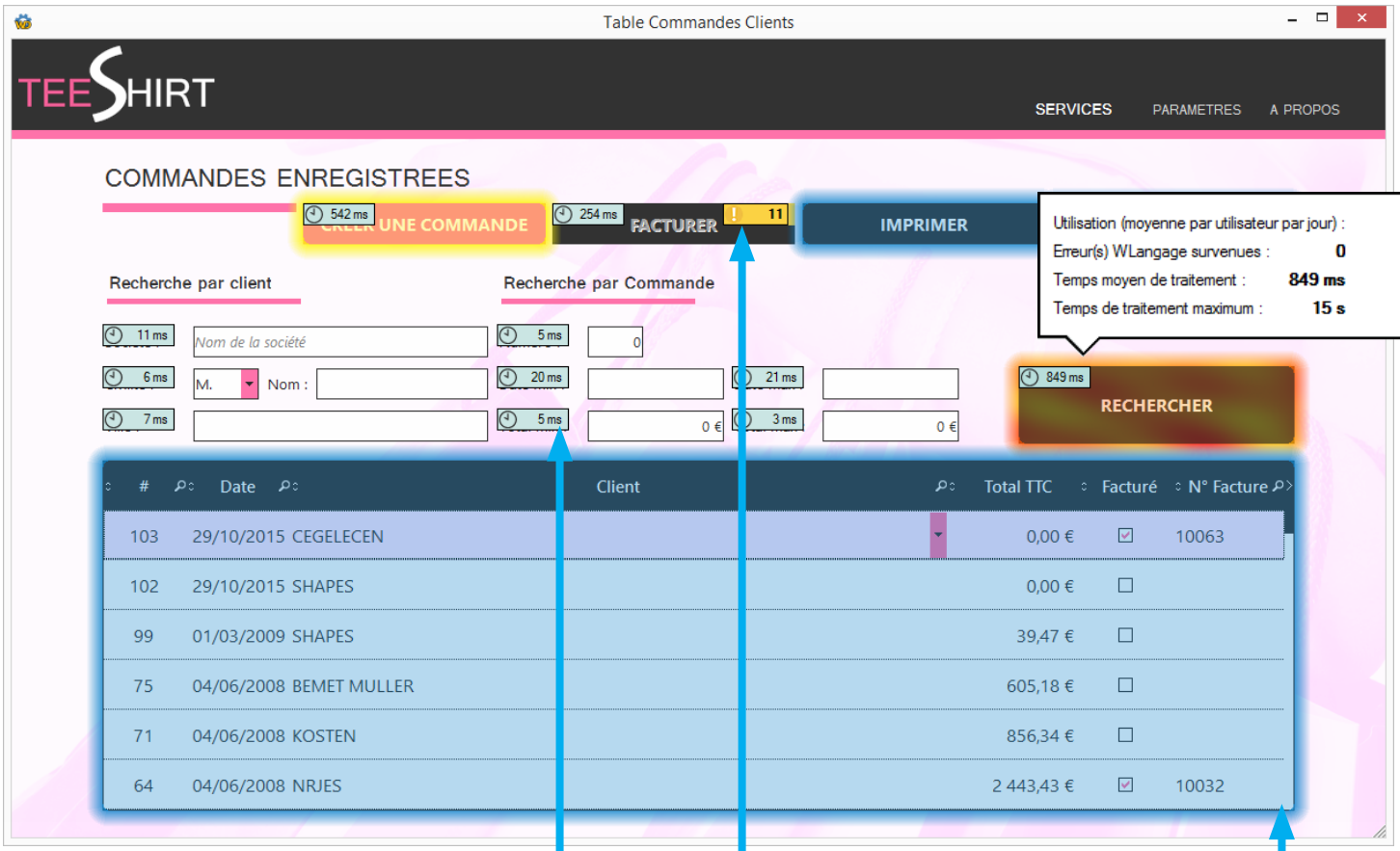


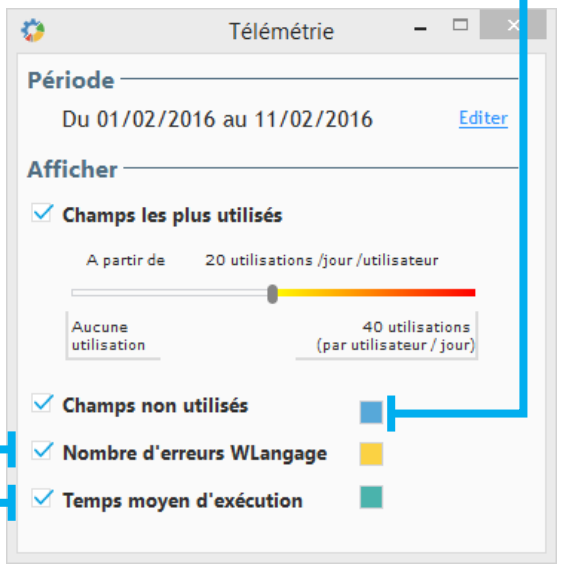
LE "GO TÉLÉMÉTRIE"

Le "Go télémétrie"
Cette fonctionnalité vous permet d'avoir un aperçu visuel des champs réellement utilisés par vos utilisateurs.

Chaque champ est mis en évidence par un halo lumineux dont la couleur indique le degré d'utilisation : du jaune (moins utilisé) au rouge (très utilisé).
Le champ le plus utilisé de la fenêtre est mis en évidence par un halo animé (ici, il s'agit du bouton "RECHERCHER").

Au survol d'un champ, une bulle de télémétrie indique :
- l'utilisation moyenne par utilisateur par jour,
- le nombre d'erreurs WLangage survenues,
- le temps moyen de traitement,
- le temps de traitement maximum.





Grâce à la télécommande, vous pouvez modifier les paramètres d'affichage :
• les champs utilisés,
• le nombre minimal d'utilisations par jour et par utilisateur pour déterminer qu'un champ est utilisé,
• les champs non utilisés (affichés par un halo bleu),
• le nombre d'erreurs WLangage générées par le champ,
• le temps moyen d'exécution du traitement du champ.

LE CHAMP FIL : CONNECTEZ VOS CHAMPS

Mettre en place des IHM intuitives et dynamiques pour construire des workflow, des plans ou des diagrammes, c'est très facile grâce au champ Fil. Le champ Fil vous permet de mettre en place un lien visuel entre 2 champs.

Le champ Fil

Le champ Fil permet de relier visuellement 2 champs. Un fil peut être :
• composé d'un ou de plusieurs segments.
• rectiligne, à angles droits ou libres.

Lors du déplacement d'un des champs attachés au fil (en édition, par programmation ou en mode PIU, voir article pages 42-43), le fil suit automatiquement la position des champs reliés.

Créer un fil

Le champ Fil peut être créé :
• en édition. Depuis le ruban, sous le volet "Création", dans le groupe "Ajuster", cliquez sur "Fil entre 2 champs". Si vous avez préalablement sélectionné 2 champs dans la fenêtre, le fil est automatiquement créé entre ces 2 champs.
• par programmation, via la fonction WLangage ***ChampCréé***.
• en mode "PIU" par l'utilisateur final.

Paramètres du fil

Le champ Fil dispose de plusieurs propriétés, modifiables en édition ou par programmation :
• les champs de départ et d'arrivée,
• la forme des extrémités (carré, flèche...),
• les points d'attache de départ et d'arrivée (position où le fil démarre et arrive sur un champ),
• les points de passage (positions par lesquelles le fil passera obligatoirement).

```
MonFil est un Champ
MonFil <-
  ChampCrée(sNomFil, typFil)

// Champ de départ et d'arrivé
MonFil..ChampDépart = ...
  IMG_Pile1
MonFil..ChampArrivée = ...
  INT_Interrupteur1
```

```
// Formes des extrémités
MonFil..FormeExtrémitéDépart =
  formeAucune
MonFil..FormeExtrémitéArrivée =
  formeRond

// Points d'attache
MonFil..PointDAttacheDépart = ...
  ptmilieudroite
MonFil..PointDAttacheArrivée = ...
  PointDAttacheAjoute(...
    INT_Interrupteur1, 0, 44)

// Couleur du fil
MonFil..Couleur = RVB(241,190,65)

// Couleur de fond
// (pour le rond à l'extrémité)
MonFil..CouleurFond = ...
  RVB(241,190,65)
```

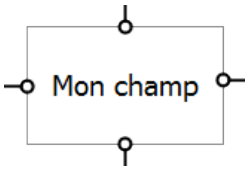
Comprendre les points d'attache

Par défaut, WINDEV choisit le point d'ancrage le mieux adapté en fonction de la position des champs de départ et d'arrivée.

Mais vous pouvez tout à fait forcer la position de ces points d'attache à l'aide des propriétés ***..PointDAttacheDépart*** et ***..PointDAttacheArrivée***.

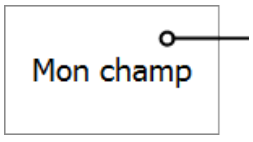
La position des points d'attache peut être définie :
• via une des constantes ***ptXXX*** disponibles. Ces constantes permettent d'indiquer sur quel bord sera fixé le fil (par exemple, ***ptBasCentre*** pour fixer le point d'attache au bord bas). Le fil sera toujours attaché au centre de ce bord.

```
// Le point d'attache de départ est
// le centre du bord bas
FIL_MonFil..PointDAttacheDépart=...
  ptBasCentre
```

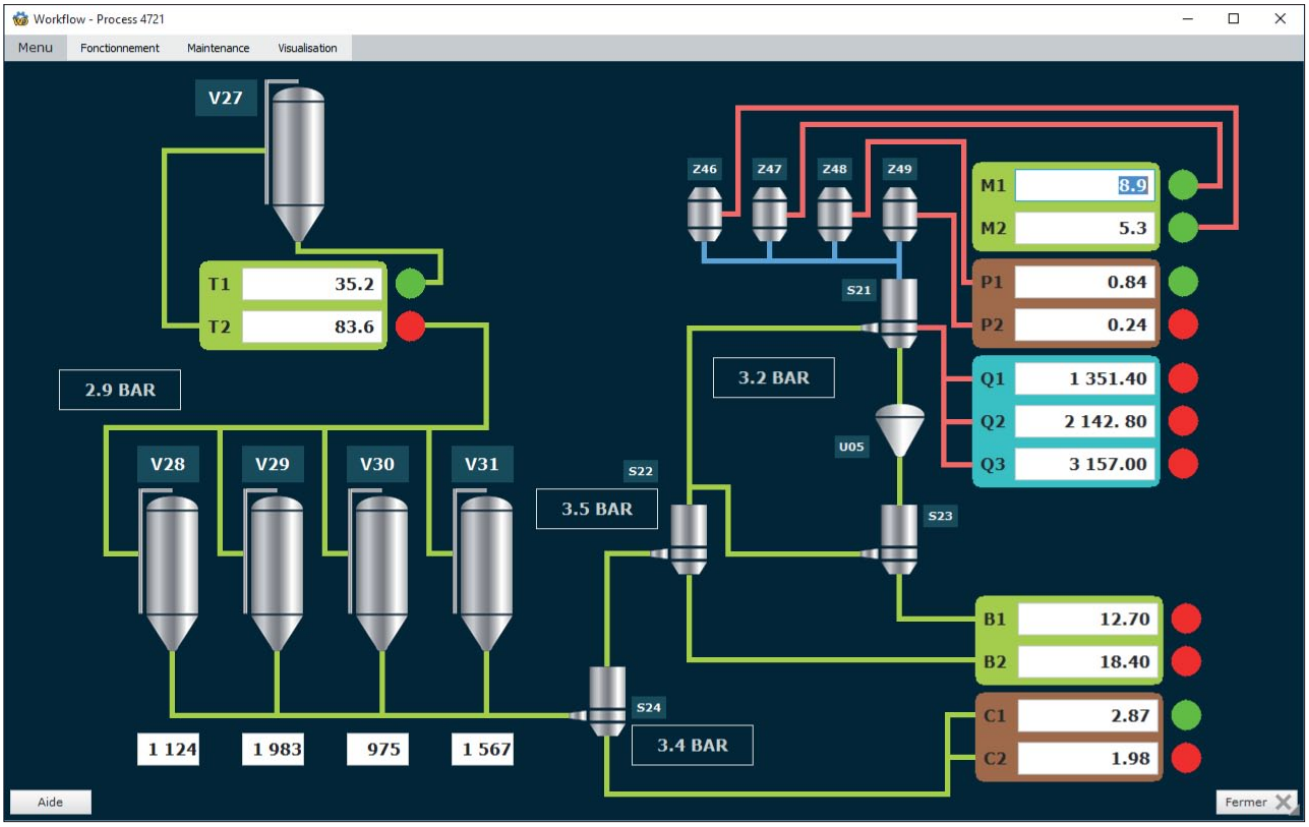


• via un identifiant de point d'attache. La fonction WLangage ***PointDAttacheAjoute*** permet en effet de définir un point d'attache personnalisé pour un champ. Cela permet de positionner le point d'attache ailleurs que sur le centre d'un bord.

```
// Définit le point d'attache
// au champ de départ :
// - à 75 % en largeur
// - à 25 % en hauteur
nIdPointAttacheDépart = ...
  PointDAttacheAjoute(
  BTN_MonChamp, 75, 25)

// Modifie le point d'attache du fil
FIL_MonFil..PointDAttacheDépart=...
  nIdPointAttacheDépart
```



La personnalisation des points d'attache est particulièrement utile si les champs reliés ont un visuel spécifique et que les fils doivent se fixer à un endroit très précis (pour respecter le visuel).



Comprendre les points de passage

Par défaut, le champ Fil relie les 2 champs au mieux, en utilisant 3 points de passage.

Mais vous pouvez définir par programmation, pour chaque champ Fil, des points de passage par lesquels le champ Fil passera obligatoirement.

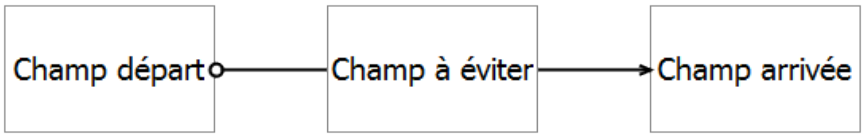
Il suffit pour cela de renseigner la propriété ..***PointDePassage*** du champ Fil.
Il est possible de fournir :
• un tableau de variables de type ***Point*** contenant les coordonnées des points de passage,
• une chaîne de caractères contenant les différentes coordonnées séparées par des virgules,
• une chaîne vide ("") pour supprimer tous les points de passage.

Attention
Les coordonnées des points de passage sont des coordonnées **relatives** à la position de départ du fil. De plus, comme ces points de passage sont relatifs au champ de départ, si ce champ de départ est déplacé (par l'utilisateur final par exemple), les points de passage se déplaceront en même temps que le champ.

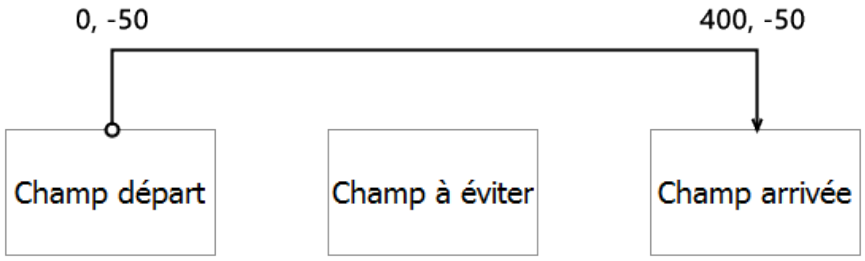
```
// Tableau qui contiendra
// tous les points de passage
// du champ Fil
tabPointsDePassage est un
        tableau de Points

// Ajoute 2 points de passage
// au tableau
Ajoute(tabPointsDePassage,
  [0, -50])
Ajoute(tabPointsDePassage,
  [400, -50])

// Modifie les points de passage
// du champ Fil
FIL_Fil..PointDePassage = ...
  tabPointsDePassage

// Force les points d'attache
FIL_Fil..PointDAttacheDépart = ...
  ptHautCentre
FIL_Fil..PointDAttacheArrivée = ...
  ptHautCentre
```

Notre conseil
Lorsque vous définissez des points de passage dans votre IHM, fixez également les points d'attache des champs reliés par le fil. Cela permet de forcer le fil à prendre le chemin souhaité et d'être sûr que le fil ne s'attachera pas de façon inattendue.

■



*Par défaut, WINDEV applique le **plus court chemin** entre le départ et l'arrivée du fil.*
Selon les IHM, il est possible qu'on veuille "éviter" des champs, via des points de passage.



Les positions des points de passage sont relatives au départ du fil.
En donnant une coordonnée Y négative par exemple, vous forcez le fil à aller "vers le haut".

PIU (PERSONNALISATION DE L'INTERFACE PAR L'UTILISATEUR)

Quoi de plus frustrant pour un utilisateur qu'une IHM mal adaptée : une zone de texte trop petite, des boutons indispensables non mis en avant...
En version 21, l'utilisateur final a la possibilité de personnaliser complètement l'IHM directement depuis l'exécutable, en quelques clics.
Et bien entendu, le développeur peut récupérer ces modifications pour les mettre en place dans une prochaine mise à jour.

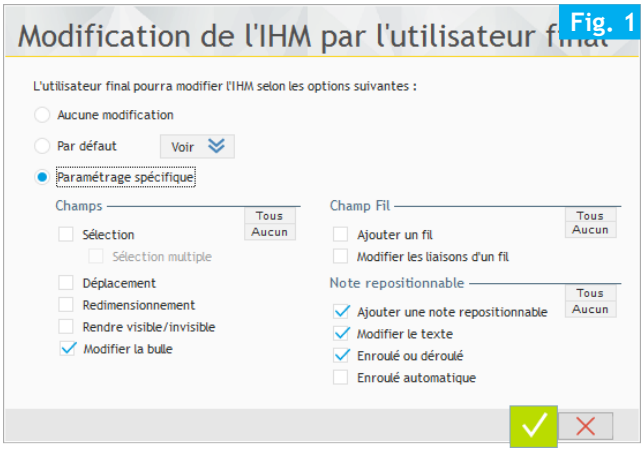


Fig. 1

Projet d'illustration

Le projet utilisé pour illustrer ce sujet est l'exemple "WD PIU et MCU TDF21" (présent sur le DVD).

Mise en place et configuration

La personnalisation de l'interface par l'utilisateur final est une option activée par défaut pour toutes les fenêtres des projets WINDEV.

Cette option peut être paramétrée :
• au niveau du projet (description du projet, onglet "Avancé") [**Fig. 1**],
• au niveau de la fenêtre (description de la fenêtre, onglet "Détail").

Vous pouvez choisir de :
• désactiver la modification,
• appliquer les paramètres par défaut (pour le projet) / appliquer les paramètres du projet (pour les fenêtres),
• personnaliser les modifications possibles pour les champs, les champs Fil et les notes repositionnables.

Passer en mode modification

Lorsque la modification de l'interface par l'utilisateur final est activée, un sous-menu "Personnaliser l'interface" spécifique apparaît dans le menu contextuel de la fenêtre.

À partir de ce menu, l'utilisateur peut passer en mode modification : il est alors possible de déplacer, redimensionner, cacher les champs de la fenêtre, ajouter des fils ou éditer les bulles d'aide.

Lorsque les modifications sont terminées, l'option "Fin du mode modification" permet de repasser en exécution.

Le menu contextuel du mode édition permet également à l'utilisateur de :
• revenir à la configuration initiale (c'est-à-dire la configuration définie par le développeur),
• charger et sauver une configuration.

La sauvegarde de la configuration se fait dans un fichier ".wdmodif".

Ce fichier est utile pour partager ses modifications avec un collègue ou les fournir au développeur de l'application pour qu'elles soient intégrées dans une version ultérieure.

Le développeur peut alors récupérer tout ou partie des modifications pour adapter au mieux les modifications.

MCU

La modification d'interface permet à l'utilisateur de modifier la position et les dimensions des champs.

Le MCU permet à l'utilisateur final de saisir du code WLangage pour ajouter une fonctionnalité : il peut par exemple modifier la couleur d'un champ.

Par défaut, le MCU n'est pas activé : dans la fenêtre de description du projet, onglet "Avancé", cliquez sur "Macro Code Utilisateur" et sélectionnez "Oui : Autoriser la saisie de Macro Code Utilisateur (MCU)".
Remarque : il est possible d'interdire l'utilisation du MCU dans une fenêtre en cochant l'option adéquate dans la fenêtre de description, onglet "Détail".

En exécution, le raccourci "Windows + F2" permet d'ouvrir la fenêtre du MCU.
L'utilisateur peut alors modifier le rendu du bouton tout simplement, en 2 lignes de code.

```
// Le bouton doit être rouge
MoiMême..CouleurFond = ...
  RougePastel
```

Modification d'IHM en mode Go

Lorsque vous testez votre projet ou une fenêtre via un Go, vous avez également la possibilité de passer en mode modification.

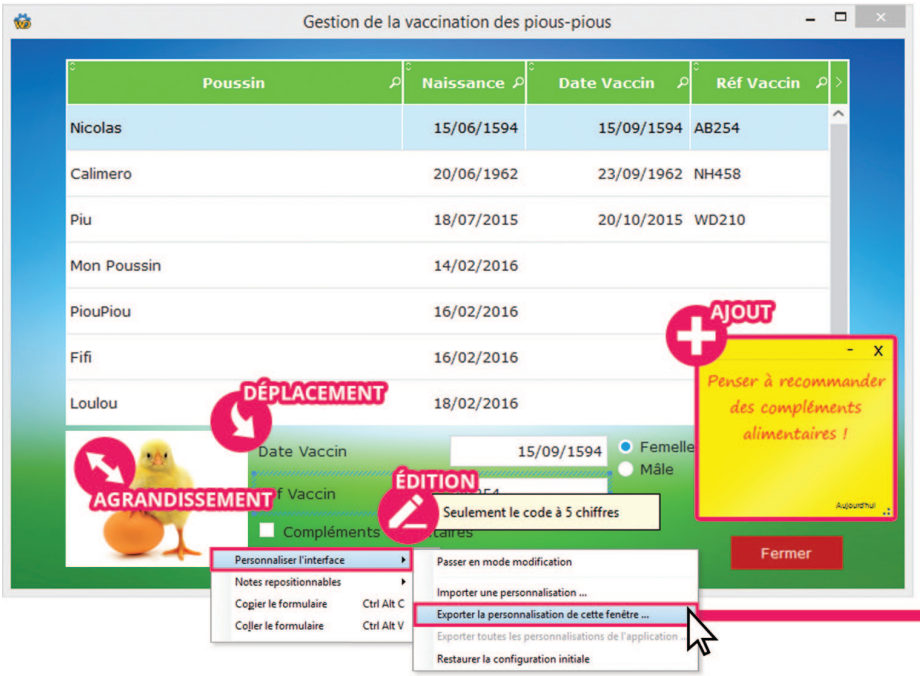
Cette fonctionnalité est particulièrement utile pour mettre au point l'IHM, avec des vraies données qui ne peuvent pas être affichées par le Live Data de WINDEV (par exemple, des données provenant d'un webservice).

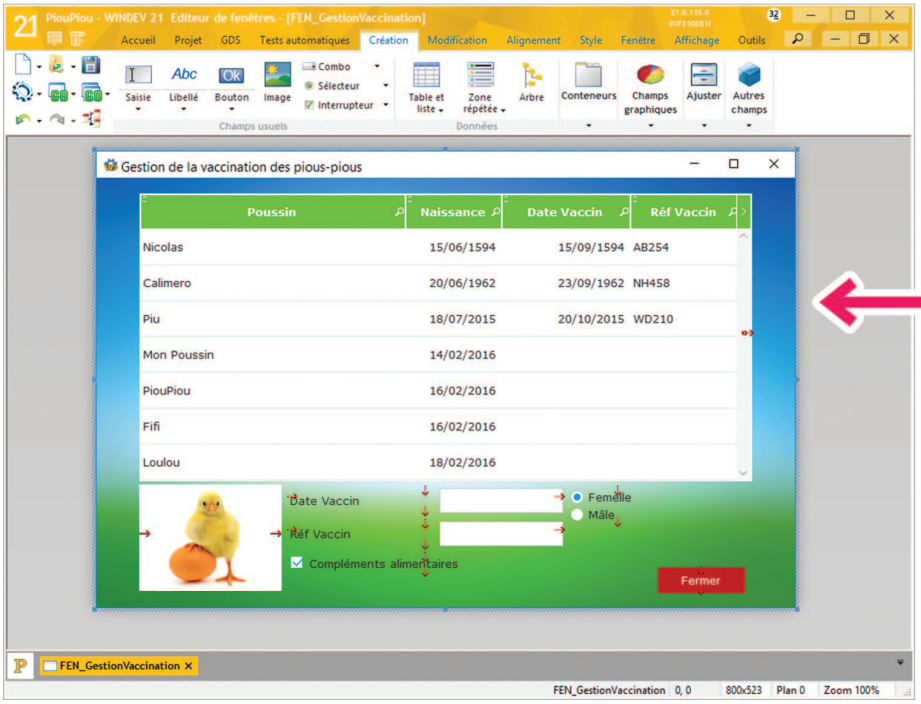
À la fin du test, lors du retour dans l'éditeur, WINDEV vous propose automatiquement de récupérer les modifications réalisées pendant la phase de test (exactement comme si vous importiez un fichier ".wdmodif").

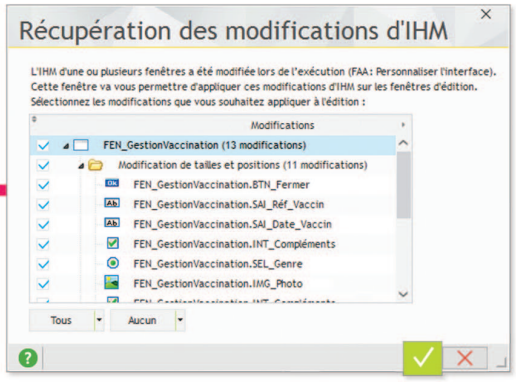
■

Les fonctions WLangage FenEd

Si vous souhaitez avoir la main sur les modifications d'interface réalisées par l'utilisateur, le WLangage propose une famille de fonctions complète ***FenEdXXX*** ("http://doc.pcsoft.fr/?1000021854"). Pour avoir un aperçu de ces fonctions et de leur utilisation, n'hésitez pas à consulter l'exemple "WD Objets Graphiques" (livré en standard avec WINDEV).
L'article sur les notes repositionnables (pages suivantes) utilise également certaines de ces fonctions.







Depuis l'exécutable, l'utilisateur final peut personnaliser l'interface de la fenêtre : depuis le menu contextuel de la fenêtre (clic droit), option "Personnaliser l'interface .. Passer en mode modification".

La modification d'interface permet d'agir sur les champs visibles. Il est possible de :
- *déplacer un champ,*
- *redimensionner un champ,*
- *éditer la bulle d'aide d'un champ.*

L'utilisateur peut également ajouter des notes repositionnables (voir l'article à ce sujet pages 44-45).

Lorsque ses modifications sont terminées, l'utilisateur peut exporter ses modifications pour les transmettre au développeur de l'application (via un fichier .wdmodif).

Le développeur de l'application peut alors importer le fichier .wdmodif et sélectionner les modifications qu'il souhaite intégrer à la fenêtre.

LE CHAMP NOTE REPOSITIONNABLE

Le champ Note repositionnable permet d'intégrer dans vos applications un pense-bête toujours visible. Ce champ est très pratique pour avoir toujours à portée, dans l'application, un "mémento" sur un champ, un client, ...

Projet d'illustration

Le projet utilisé pour illustrer ce sujet est l'exemple "WD SynchroNotes TDF21" (présent sur le DVD).

Le champ Note Repositionnable

Le champ Note repositionnable permet d'afficher un "pense-bête" dans une fenêtre.
Il permet à l'utilisateur final d'afficher des informations importantes sur un client, le fonctionnement d'un champ, etc.

La création d'une note repositionnable dans une application, c'est une simple FAA (Fonctionnalités Automatiques des Applications) : il suffit à l'utilisateur d'ouvrir le menu contextuel de la fenêtre et de sélectionner "Notes repositionnables .. Créer une note repositionnable".

La note peut ensuite être déplacée n'importe où dans la fenêtre. Elle peut également être réduite.

Cette note est complètement personnalisable :
- contenu,
- couleur de texte,
- couleur de fond,
- taille du texte,
- ...

Import / export

Le menu "Notes repositionnable" du menu contextuel de la fenêtre permet à l'utilisateur final de :
- exporter les notes de la fenêtre courante,
- exporter toutes les notes de l'application,
- importer des notes (depuis un fichier généré par l'export).

Attention : les notes exportées peuvent uniquement être réimportées dans la même fenêtre de la même application.

Un champ avant tout

Le champ Note repositionnable est un champ et peut donc être créé :
- directement depuis l'éditeur de fenêtres (dans le groupe "Autres champs" du volet "Création").
- par programmation (via la fonction ***ChampCrée***).

Pour désactiver ou activer les notes repositionnables par programmation sur une fenêtre, il est possible d'utiliser la propriété ***..FenEditionMode*** avec la constante ***feAjouteNoteRepositionnable***.
La manipulation des notes repositionnables peut également être désactivée en édition, pour tout le projet ou pour une fenêtre donnée (voir page 42).

Synchroniser des notes entre plusieurs postes utilisateurs

Les notes sont locales à une application et à un utilisateur.
Mais un pense-bête utile, il peut être intéressant de le partager entre tous les utilisateurs d'une même application !

L'exemple "WD SynchroNotes TDF21" propose un composant interne "SynchroNotes" dont le but est de synchroniser toutes les notes de l'application, pour tous les utilisateurs.

Principe

À la fermeture de la fenêtre, les notes présentes dans la fenêtre sont exportées et mémoriser dans un buffer.
Ce buffer est stocké dans un fichier de données HFSQL.
À l'ouverture de la fenêtre, les notes sont réimportées à partir du buffer stocké préalablement.

Mise en œuvre

Dans un premier temps, il est nécessaire d'exporter les notes.
La fonction WLangage ***FenEdSauve*** permet de récupérer dans un buffer (ou un fichier) toutes les modifications d'interface réalisées par l'utilisateur.
Cette fonction peut être utilisée uniquement si la fenêtre est en mode édition : il est donc nécessaire de forcer le passage en mode édition via la fonction WLangage ***FenEdActive***.

```
// Passe en mode édition
FenEdActive(MaFenêtre, Vrai)

// Mémorise les modifications
// sur les notes
bufNote est un Buffer = FenEdSauve(
  MaFenêtre, enMémoire,
  feModeNoteRepositionnable)

// Fin du mode édition
FenEdActive(MaFenêtre, Faux)
```

Le buffer contenant les modifications doit ensuite être enregistré en base. Les modifications étant propres à une fenêtre d'un projet, le buffer est mémorisée par rapport à une clé composée "Nom du projet + nom de la fenêtre".

```
// Recherche les notes pour
// cette fenêtre
HLitRechercheDebut(...
  SynchroNotes,
  NomProjetNomFenetre,
  [ProjetInfo(piNomProjet),
    MaFenêtre..Nom],
  hBlocageEcriture)
// Prépare un nouvel enregistrement
SI HTrouve() = Faux ALORS
  HRAZ(SynchroNotes)
FIN

// Remplit les rubriques
AVEC SynchroNotes
  .NomProjet = ...
    ProjetInfo(piNomProjet)
```

```
  .NomFenetre = ...
    MaFenêtre..Nom
          .SynchroNotes = bufNote
FIN

// Enregistre les notes
HEnregistre(SynchroNotes)
```

Pour recharger et réafficher les notes, il suffit de faire les opérations inverses : récupérer le buffer contenant les notes et les appliquer à la fenêtre avec la fonction WLangage ***FenEdCharge***.

```
// Recherche les notes pour
// cette fenêtre
SI HLitRechercheProchain(...
  SynchroNotes,
  NomProjetNomFenetre,
  [ProjetInfo(piNomProjet),
   MaFenêtre..NomComplet]) ALORS
  // Récupère les modifications
  bufNote est un Buffer = ...
    SynchroNotes.SynchroNotes
  // Charge les modifications
  // dans la fenêtre
  FenEdCharge(...
    MaFenêtre..NomComplet,
    bufNote)
FIN
```

Utilisation

Pour faciliter la réutilisation du partage de notes, un composant interne a été utilisé. L'avantage du composant interne dans ce cas est qu'il dispose d'une analyse spécifique (utile dans notre cas pour mémoriser les notes).

La fonction ***SynchroNotesParamètre*** permet de paramétrer la connexion à utiliser pour la mémorisation des notes.
Cette fonction attend en paramètres :
• soit les paramètres de la connexion (serveur, base, utilisateur, mot de passe et port),
• soit une variable de type ***Connexion***.

```
// Définit la connexion pour
// la synchronisation des notes
SynchroNotesParamètre(...
  gcnxMaConnexionHFCS)
```

Pour demander à mémoriser les notes d'une fenêtre, il suffit d'utiliser la fonction ***SynchroNotesCharge*** dans le code de déclaration de la fenêtre.

Et pour la synchronisation des notes en fermeture ? Et bien, aucun appel à prévoir ! C'est en effet la fonction ***SynchroNotesCharge*** qui se charge d'ajouter le traitement de sauvegarde en fermeture, à l'aide de la propriété ***..Traitement***.

```
// Branche le traitement de
// sauvegarde à la fermeture
MaFenêtre..Traitement[trtFermeture]
+= SynchroNotesSauve
```

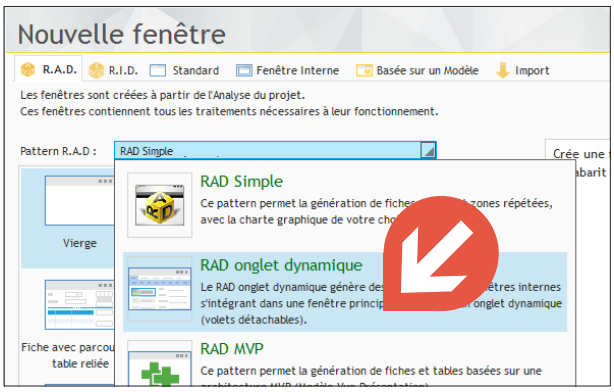
Cette syntaxe permet d'ajouter un appel à la fonction ***SynchoNotesSauve*** au traitement "Fermeture de la fenêtre".

■

RAD ONGLET DYNAMIQUE

En version 21, un nouveau type de RAD fait son apparition : il s'agit du RAD "Onglet dynamique" (ou RAD "Ruban"). La principale caractéristique de ce RAD est qu'il génère des fenêtres internes utilisées dans des onglets dynamiques. La navigation est effectuée à l'aide d'un ruban, automatiquement généré. Ce RAD est un "RAD Fenêtre".

Le projet "WD RAD Onglet Dynamique TDF21" (présent sur le DVD) est un projet vierge contenant uniquement une analyse "Client-Commande-Produit" classique. N'hésitez pas à vous en servir pour tester le RAD "Onglet dynamique".

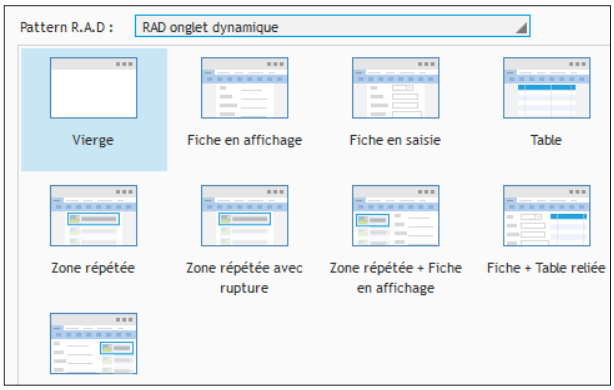


1 RAD Onglet dynamique : un RAD Fenêtre

Le RAD Onglet dynamique est un RAD Fenêtre, c'est-à-dire qu'il est disponible depuis la fenêtre de création d'une nouvelle fenêtre.

Il ne s'agit pas d'un RAD complet (qui génère toutes les fenêtres pour tous les fichiers de l'analyse). Vous choisissez vous-mêmes les fenêtres nécessaires, avec la configuration la plus adaptée au rendu attendu.

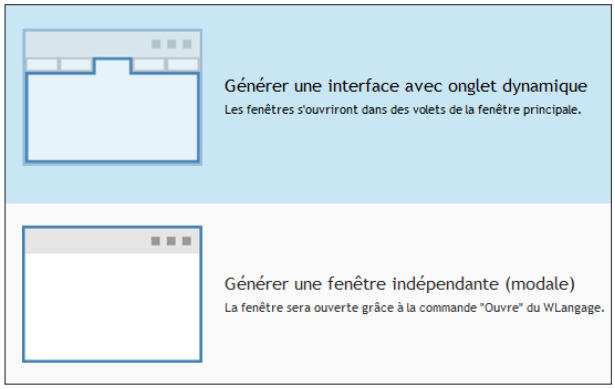
Depuis l'onglet "R.A.D." de la fenêtre "Nouvelle fenêtre", sélectionnez le pattern "RAD onglet dynamique".



2 Choix du type fenêtre

Le RAD Onglet dynamique propose 8 fenêtres qui couvrent l'ensemble des besoins :

- une fiche (en affichage ou en saisie),
- une table (pour afficher toutes les données, effectuer des recherches, etc.),
- une zone répétée, pour un affichage plus visuel,
- une zone répétée avec rupture, utile si les données peuvent être regroupées selon un critère,
- une zone répétée avec fiche associée, pour visualiser directement le détail d'un enregistrement,
- une fiche avec une table ou zone répétée reliée (utile pour afficher une commande et les lignes de commande associées par exemple).



3 Mode de génération

Quel que soit le type de fenêtre que vous choisissez, la première étape de l'assistant consiste à sélectionner le mode de génération de la fenêtre.

Deux modes sont disponibles :

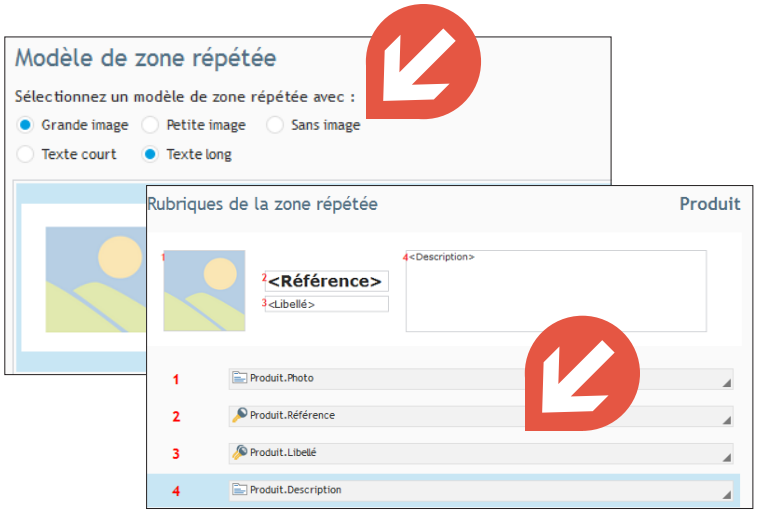
1. "Générer une interface avec onglet dynamique".
Dans ce mode, WINDEV va générer une fenêtre principale ("FEN_Onglet") contenant un champ Onglet dynamique. La fenêtre RAD générée sera en fait une fenêtre interne qui s'ouvrira comme un volet dynamique.

2. "Générer une fenêtre indépendante (modale)".
Dans ce mode, WINDEV va générer une fenêtre indépendante, comme un RAD fenêtre "classique".

4 Choix du fichier

Une fenêtre RAD peut être basée sur :
• un fichier de données,
• une requête,
• une vue (SQL ou matérialisée).

En fonction de vos fichiers de données, il peut être intéressant de passer par une requête ou une vue pour regrouper des données complémentaires (fichier Client et fichier Adresse par exemple).
Cela permet d'offrir une IHM plus compréhensible et intéressante à vos utilisateurs.



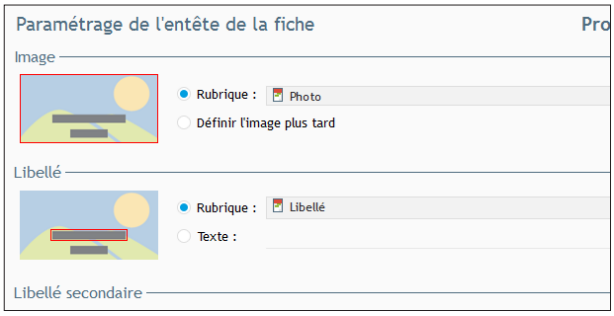
5 Modèle de zone répétée

Si vous choisissez une fenêtre utilisant une zone répétée, vous pouvez choisir différents modèles de présentation, chacun s'adaptant mieux à un type de contenu.

Par exemple, un modèle avec une image et un texte long sera plus adapté à l'affichage de produits que de clients.

Pour choisir le modèle le plus adapté :
• commencez par choisir si le champ doit afficher une image (grande ou petite) ainsi que la taille du texte principal,
• sélectionnez le modèle qui vous semble le mieux convenir à votre fichier de données.

Selon le modèle sélectionné, l'assistant vous proposera de définir, pour chaque champ, la rubrique source. Par défaut, WINDEV détecte les champs les plus appropriés (par exemple pour l'image, la première rubrique "mémo image" trouvée dans le fichier).

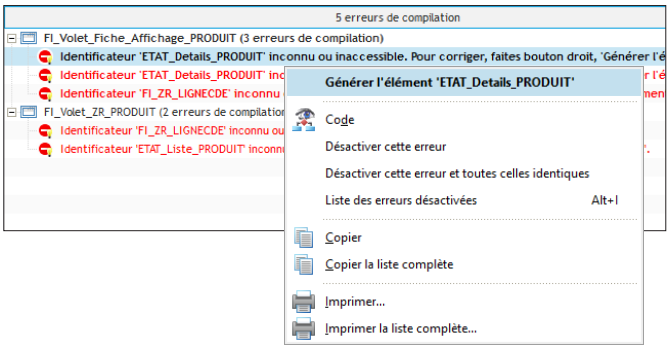


6 Fenêtre fiche

La fenêtre fiche se décompose en deux sections : l'entête et le corps.

L'entête est une section mise en avant et contenant généralement les informations qui identifient le mieux l'enregistrement. Pour un produit par exemple, cet entête contiendra la photo du produit, sa référence et son libellé ; pour un client, cet entête contiendra plutôt son identité (nom et prénom) et sa société.

Le corps de la fiche contiendra ensuite toutes les autres rubriques que vous souhaitez afficher.

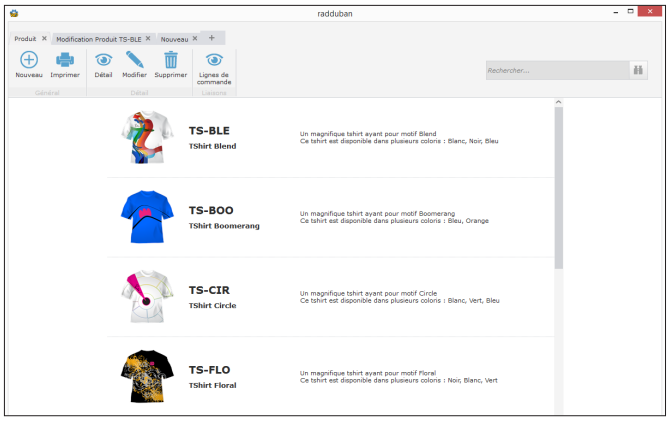


7 Erreurs interactives

À la génération d'une fenêtre, le RAD va automatiquement créer des boutons pour imprimer le contenu courant (la fiche ou la liste) et accéder aux fenêtres des fichiers liés (par exemple, depuis la fiche d'un client, il est possible d'accéder à la liste de ses commandes).

Le code de ces boutons va générer une erreur du type "*Identificateur 'xxx' inconnu ou inaccessible. Pour corriger, faites bouton droit, 'Générer l'élément*'".

Vous pouvez donc, au choix :
• générer l'élément manquant directement depuis le volet des erreurs,
• adapter ou supprimer le code du bouton généré.



8 Résultat final

Le RAD Onglet dynamique vous permet de générer des fenêtres à l'IHM soignée, correspondant à une bonne base d'application.

L'utilisation du champ Onglet dynamique offre plusieurs avantages à l'utilisateur final :
• afficher 2 fiches côte à côte pour les comparer ou recopier des informations de l'une vers l'autre,
• afficher simultanément plusieurs fenêtres (ou volets) complètement,
• ...

■

LE CHAMP TABLEAU CROISÉ DYNAMIQUE (CUBE ROLAP)

Le champ Tableau croisé dynamique est un champ indispensable pour vos décisionnaires. Il présente de manière synthétique toutes les statistiques nécessaires à la prise de décisions.
Avec WINDEV, en quelques clics, vous créez un champ Tableau Croisé dynamique complet et performant.

Projet d'illustration

Le projet utilisé pour illustrer ce sujet est l'exemple "WD TCD Et Etats TDF21" (présent sur le DVD).

Le champ Tableau croisé dynamique (TCD)

Le tableau croisé dynamique ("TCD"), ou Cube ROLAP, permet aux décisionnaires d'avoir visuellement toutes les statistiques dont ils ont besoin pour prendre les décisions.

Le champ Tableau croisé dynamique peut être rempli :
• directement par binding sur un fichier de données,
• par une requête,
• par programmation.

En fonction de la quantité de données, le remplissage d'un tableau croisé dynamique peut être relativement long.

Il peut donc être intéressant d'optimiser ce remplissage.

Dans notre exemple, le tableau croisé dynamique affiche le chiffre d'affaires et le pourcentage de remplissage de chaque croisière par région du monde, nom de la croisière, nom du bateau et par année, trimestre, mois, semaine.

Ce tableau croisé dynamique est directement lié à une requête : cette requête est donc réexécutée à chaque fois que l'utilisateur demande à remplir le champ.

Pour optimiser le chargement des données, plusieurs possibilités sont disponibles.

Fonctions TCDSauve et TCDCharge

La première astuce, la plus simple à mettre en œuvre, consiste à mémoriser le contenu "calculé" du tableau croisé dynamique puis à le charger aux lancements suivants.

Pour cela, le WLangage dispose de 2 fonctions :
• ***TCDSauve*** permet de mémoriser le contenu d'un champ Tableau croisé dynamique dans un fichier,
• ***TCDCharge*** permet de charger un tableau croisé dynamique à partir d'un fichier.

```
gsNomFichier est une chaîne

// Définit le fichier de mémorisation
gsNomFichier = fRepExe() + ["\"] +
  TCD_Stats..NomComplet +
  ".wdolap"

// Tente de charger le fichier
SI TCDCharge(TCD_Stats,
        gsNomFichier)=Faux ALORS

  // Calcul complet
  TCDCalculeTout(TCD_Stats)

FIN

// Mémorise le contenu
TCDSauve(TCD_Stats, gsNomFichier)
```

Avec cette méthode, après le remplissage initial, tous les autres affichages du champ seront immédiats.
Le chargement est immédiat parce que les données sont figées dans le fichier.

Or, en base, les données évoluent.

Pour mettre à jour les données chargées depuis un précédent remplissage, il suffit d'utiliser la fonction WLangage ***TCDCalculeMiseAJour***.

```
// Tente de charger le fichier
SI TCDCharge(TCD_Stats,
gsNomFichier) ALORS
  // Mise à jour des données
  TCDCalculeMiseAJour(TCD_Stats)
SINON
  // Calcul complet
  TCDCalculeTout(TCD_Stats)
FIN
```

Cette méthode accélère grandement le remplissage du champ. Cependant la mise à jour des données via ***TCDCalculeMiseAJour*** va réexécuter et reparcourir la requête pour ajouter les nouvelles données.

Pour gagner encore en performances, il est possible de passer par une vue matérialisée.

Vue matérialisée

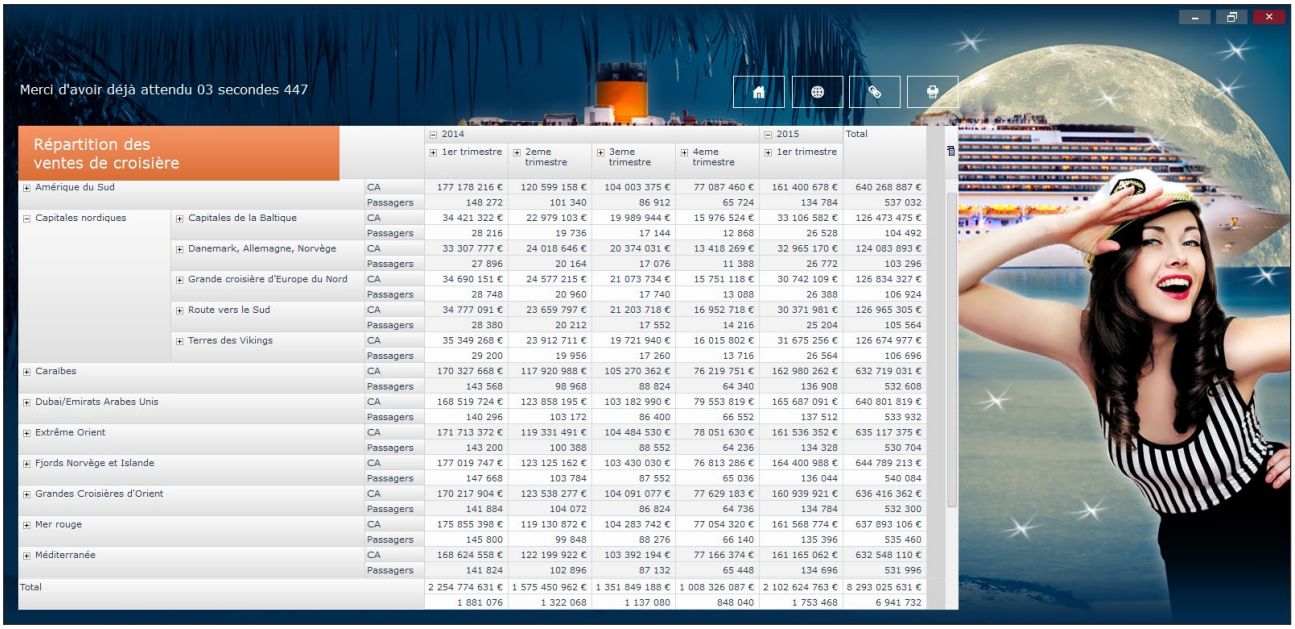
Une vue matérialisée permet de sauver le résultat d'une requête directement dans un fichier de données.

Dans notre cas, il suffit de reprendre la même requête pour créer une vue matérialisée.

L'avantage de la vue matérialisée par rapport à la requête est qu'elle est toujours disponible : il n'y a jamais besoin de la réexécuter pour afficher le tableau croisé dynamique.

Et pour les mises à jour, il suffit de rafraîchir la vue régulièrement, par exemple via une tâche planifiée.
Ainsi, seul le serveur effectue le traitement et non le poste client.
Le temps de rafraîchissement est donc imperceptible pour l'utilisateur final !

Notre conseil
L'utilisation d'une vue matérialisée ne remplace pas la sauvegarde par fichier.
Vous pouvez tout à fait combiner les 2 solutions : le chargement initial sera accéléré de



Répartition des ventes de croisière

			2014				2015	Total
			1er trimestre	2eme trimestre	3eme trimestre	4eme trimestre	1er trimestre	
Amérique du Sud		CA	177 178 216 €	120 599 158 €	104 003 375 €	77 087 460 €	161 400 678 €	640 268 887 €
		Passagers	148 272	101 340	86 912	65 724	134 784	537 032
Capitales nordiques	Capitales de la Baltique	CA	34 421 322 €	22 979 103 €	19 989 944 €	15 976 524 €	33 106 582 €	126 473 475 €
		Passagers	28 216	19 736	17 144	12 868	26 528	104 492
	Danemark, Allemagne, Norvège	CA	33 307 777 €	24 018 646 €	20 374 031 €	13 418 269 €	32 965 170 €	124 083 893 €
		Passagers	27 896	20 164	17 076	11 388	26 772	103 296
	Grande croisière d'Europe du Nord	CA	34 690 151 €	24 577 215 €	21 073 734 €	15 751 118 €	30 742 109 €	126 834 327 €
		Passagers	28 748	20 960	17 740	13 088	26 388	106 924
	Route vers le Sud	CA	34 777 091 €	23 659 797 €	21 203 718 €	16 952 718 €	30 371 981 €	126 965 305 €
		Passagers	28 380	20 212	17 552	14 216	25 204	105 564
	Terres des Vikings	CA	35 349 268 €	23 912 711 €	19 721 940 €	16 015 802 €	31 675 256 €	126 674 977 €
		Passagers	29 200	19 956	17 260	13 716	26 564	106 696
Caraïbes		CA	170 327 668 €	117 920 988 €	105 270 362 €	76 219 751 €	162 980 262 €	632 719 031 €
		Passagers	143 568	98 968	88 824	64 340	136 908	532 608
Dubaï/Emirats Arabes Unis		CA	168 519 724 €	123 858 195 €	103 182 990 €	79 553 819 €	165 687 091 €	640 801 819 €
		Passagers	140 296	103 172	86 400	66 552	137 512	533 932
Extrême Orient		CA	171 713 372 €	119 331 491 €	104 484 530 €	78 051 630 €	161 536 352 €	635 117 375 €
		Passagers	143 200	100 388	88 552	64 236	134 328	530 704
Fjords Norvège et Islande		CA	177 019 747 €	123 125 162 €	103 430 030 €	76 813 286 €	164 400 988 €	644 789 213 €
		Passagers	147 668	103 784	87 552	65 036	136 044	540 084
Grandes Croisières d'Orient		CA	170 217 904 €	123 538 277 €	104 091 077 €	77 629 183 €	160 939 921 €	636 416 362 €
		Passagers	141 884	104 072	86 824	64 736	134 784	532 300
Mer rouge		CA	175 855 398 €	119 130 872 €	104 283 742 €	77 054 320 €	161 568 774 €	637 893 106 €
		Passagers	145 800	99 848	88 276	66 140	135 396	535 460
Méditerranée		CA	168 624 558 €	122 199 922 €	103 392 194 €	77 166 374 €	161 165 062 €	632 548 110 €
		Passagers	141 824	102 896	87 132	65 448	134 696	531 996
Total			2 254 774 631 €	1 575 450 962 €	1 351 849 188 €	1 008 326 087 €	2 102 624 763 €	8 293 025 631 €
			1 881 076	1 322 068	1 137 080	848 040	1 753 468	6 941 732

même que la mise à jour.
De cette façon, vous optimisez encore l'affichage !

N.B. : vous retrouverez dans la LST 103 un article sur les différentes méthodes de remplissage d'un champ Tableau croisé dynamique, avec leurs avantages et inconvénients.

Impression

À partir de la version 21, l'impression d'un champ Tableau croisé dynamique est disponible depuis le menu contextuel du champ, option "Imprimer...".

Le champ s'imprime tel qu'il est affiché, c'est-à-dire que les lignes et colonnes enroulées restent enroulées à l'impression.

Personnaliser l'impression

Vous avez la possibilité de personnaliser l'impression du TCD : pour respecter la charte graphique de l'entreprise par exemple (look, format, logo, informations légales...).

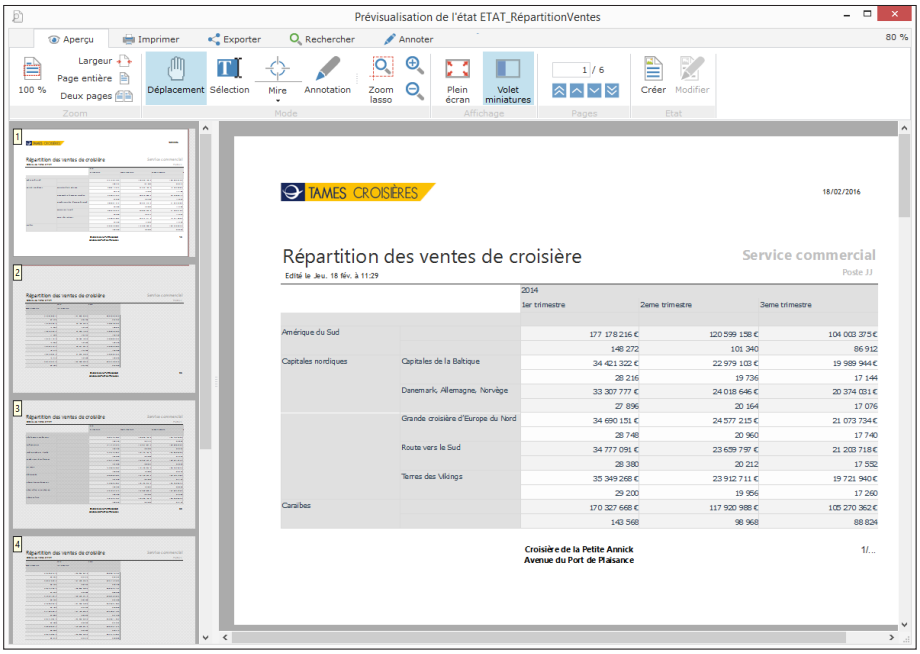
Il suffit de créer un nouvel état de type "Etat Tableau croisé" ayant pour source le champ Tableau croisé dynamique.

Vous pouvez alors le personnaliser comme vous l'entendez, par exemple en utilisant un modèle d'états.

Rappel : un modèle d'états permet de partager du code et des champs entre plusieurs états.

Bien entendu, si vous créez un état ayant pour source un champ Tableau croisé dynamique, cet état sera utilisé lors de l'utilisation de la FAA "Imprimer...".

■



L'impression du tableau croisé dynamique est réalisée par défaut dans un état prédéfini. Mais vous pouvez tout à fait personnaliser l'état utilisé (via un modèle d'états par exemple).

FAA du champ Tableau croisé dynamique

Comme pour tous les champs des applications WINDEV, vos utilisateurs disposent de nombreuses FAA (Fonctionnalités Automatiques des Applications) sur le champ Tableau croisé dynamique :

- redimensionner les colonnes et les lignes,
- rechercher dans les cellules (Ctrl + F),
- déplier et replier les colonnes et les lignes,
- supprimer et réafficher les colonnes et les lignes,
- ajouter des regroupements "date" (semaine/mois/trimestre),
- réorganiser les entêtes des lignes et de colonnes,
- exporter le contenu vers un document Excel,
- mémoriser le contenu courant et le recharger,
- imprimer le champ,
- colorer des cellules,
- ...

GRAPHES : LES ASTUCES DE MARC POUR FAIRE DES GRAPHES AU TOP

WINDEV permet de créer facilement et en standard des graphes élaborés sur vos données : courbe, aire, secteur, beignet, histogramme, etc.
Voici quelques nouveautés de la version 21 sur les graphes ainsi que quelques astuces pour améliorer le rendu.

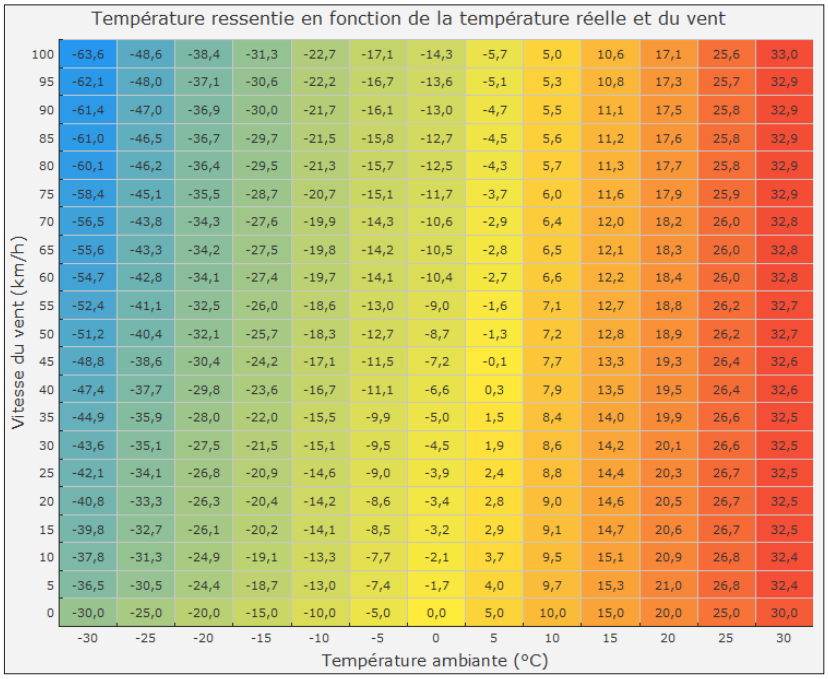
Projet d'illustration

Le projet utilisé pour illustrer ce sujet est l'exemple "WD Graphes TDF21" (présent sur le DVD).
L'exemple "WD Graphes" (livré en standard avec WINDEV) présente tous les types de graphes disponibles, dont le graphe Heatmap.

Nouveautés 21

Le graphe Heatmap

Un graphe "Heatmap" (carte de chaleur) permet de représenter en 2 dimensions des données qui ont 3 axes de valeurs.
Par exemple :
• le chiffre d'affaires selon la date et le lieu,
• la température ressentie selon la vitesse du vent et la température réelle,
• le nombre de ventes par taille et catégorie de vêtements,
• ...



	-30	-25	-20	-15	-10	-5	0	5	10	15	20	25	30
100	-63,6	-48,6	-38,4	-31,3	-22,7	-17,1	-14,3	-5,7	5,0	10,6	17,1	25,6	33,0
95	-62,1	-48,0	-37,1	-30,6	-22,2	-16,7	-13,6	-5,1	5,3	10,8	17,3	25,7	32,9
90	-61,4	-47,0	-36,9	-30,0	-21,7	-16,1	-13,0	-4,7	5,5	11,1	17,5	25,8	32,9
85	-61,0	-46,5	-36,7	-29,7	-21,5	-15,8	-12,7	-4,5	5,6	11,2	17,6	25,8	32,9
80	-60,1	-46,2	-36,4	-29,5	-21,3	-15,7	-12,5	-4,3	5,7	11,3	17,7	25,8	32,9
75	-58,4	-45,1	-35,5	-28,7	-20,7	-15,1	-11,7	-3,7	6,0	11,6	17,9	25,9	32,9
70	-56,5	-43,8	-34,3	-27,6	-19,9	-14,3	-10,6	-2,9	6,4	12,0	18,2	26,0	32,8
65	-55,6	-43,3	-34,2	-27,5	-19,8	-14,2	-10,5	-2,8	6,5	12,1	18,3	26,0	32,8
60	-54,7	-42,8	-34,1	-27,4	-19,7	-14,1	-10,4	-2,7	6,6	12,2	18,4	26,0	32,8
55	-52,4	-41,1	-32,5	-26,0	-18,6	-13,0	-9,0	-1,6	7,1	12,7	18,8	26,2	32,7
50	-51,2	-40,4	-32,1	-25,7	-18,3	-12,7	-8,7	-1,3	7,2	12,8	18,9	26,2	32,7
45	-48,8	-38,6	-30,4	-24,2	-17,1	-11,5	-7,2	-0,1	7,7	13,3	19,3	26,4	32,6
40	-47,4	-37,7	-29,8	-23,6	-16,7	-11,1	-6,6	0,3	7,9	13,5	19,5	26,4	32,6
35	-44,9	-35,9	-28,0	-22,0	-15,5	-9,9	-5,0	1,5	8,4	14,0	19,9	26,6	32,5
30	-43,6	-35,1	-27,5	-21,5	-15,1	-9,5	-4,5	1,9	8,6	14,2	20,1	26,6	32,5
25	-42,1	-34,1	-26,8	-20,9	-14,6	-9,0	-3,9	2,4	8,8	14,4	20,3	26,7	32,5
20	-40,8	-33,3	-26,3	-20,4	-14,2	-8,6	-3,4	2,8	9,0	14,6	20,5	26,7	32,5
15	-39,8	-32,7	-26,1	-20,2	-14,1	-8,5	-3,2	2,9	9,1	14,7	20,6	26,7	32,5
10	-37,8	-31,3	-24,9	-19,1	-13,3	-7,7	-2,1	3,7	9,5	15,1	20,9	26,8	32,4
5	-36,5	-30,5	-24,4	-18,7	-13,0	-7,4	-1,7	4,0	9,7	15,3	21,0	26,8	32,4
0	-30,0	-25,0	-20,0	-15,0	-10,0	-5,0	0,0	5,0	10,0	15,0	20,0	25,0	30,0

Pour manipuler ce graphe particulier, le WLangage dispose de fonctions spécifiques :
• ***grHMAjouteDonnées*** permet d'ajouter une valeur dans le graphe pour une position (X et Y) définie.

```
// Ajoute la valeur 42 à la position
// 1;1
grHMAjouteDonnée(GRF_Graphe,
  1, 1, 42)
```

• ***grHMRécupèreDonnée*** permet de récupérer la valeur définie à une position donnée.

```
// Récupère la valeur aux
// coordonnées 1;1
nValeur est un entier = ...
  grHMRécupèreDonnée(...
    GRF_Graphe, 1, 1)
```

• ***grHMCouleur*** permet de spécifier les couleurs des différentes valeurs. Les couleurs sont définies par palier. Pour les valeurs se situant entre 2 paliers, un dégradé linéaire est appliqué par défaut. Ce comportement peut toutefois être modifié à l'aide de la fonction ***grParamètre***.
La fonction ***grHMSupprimeCouleur*** permet de supprimer toute définition de couleurs dans le graphe préalablement effectuée.

```
// Supprime les couleurs
// précédemment définies
grHMSupprimeCouleur(GRF_Graphe)
// Ajoute les couleurs par palier
grHMCouleur(GRF_Graphe, -42,
  RougePastel)
grHMCouleur(GRF_Graphe, 0,
  JaunePastel)
grHMCouleur(GRF_Graphe, 42,
  VertPastel)
```

• les fonctions ***grHMLibelléX*** et ***grHMLibelléY*** permettent de définir le libellé d'une catégorie sur l'axe des abscisses et ordonnées.

Le graphe Secteur multiniveau (SMN) : excentrer les parts

Apparu en version 20, le graphe de type Secteur multiniveau est un graphe Secteur contenant plusieurs niveaux de données hiérarchiques.

L'intérêt de ce type de graphe est double :
• d'une part, il permet l'affichage concis et hiérarchisé d'un très grand nombre de données,
• d'autre part, ce type de graphe est interactif : le clic sur une part permet de rentrer dans le détail des sous-niveaux de cette part.

En version 21, la fonction ***grSMNPartExcentrée*** permet d'excentrer une part ou toutes les parts du graphe.

```
// Excentre la part "Asie"
// de 40 pixels
grSMNPartExcentrée(...
  GRF_Graphe, "Asie", 40)
// Force le dessin du graphe
```

```
// pour appliquer les modifications
grDessine(GRF_Graphe)
```

Remarque : cette fonctionnalité est également disponible pour les graphes Secteur, Hémicyle et Donut via la fonction WLangage ***grSecteurPartExcentrée***.

Astuces pour améliorer le rendu

Par défaut, les graphes générés par WINDEV sont ergonomiques et agréables à utiliser.

Mais avec quelques petites astuces utilisées à bon escient, il est possible d'améliorer encore le rendu des graphes.

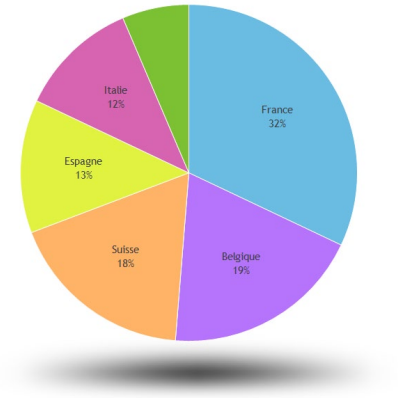
Attention : les astuces présentées ici ne conviennent pas forcément à tous les graphes. Elles doivent être utilisées avec parcimonie.

Graphe Secteur : éviter l'effet de flottement

Une des particularités des graphiques Secteur (Hémicycle, Beignet, Secteur multiniveau) est la suivante : le rendu ne repose pas sur des axes.

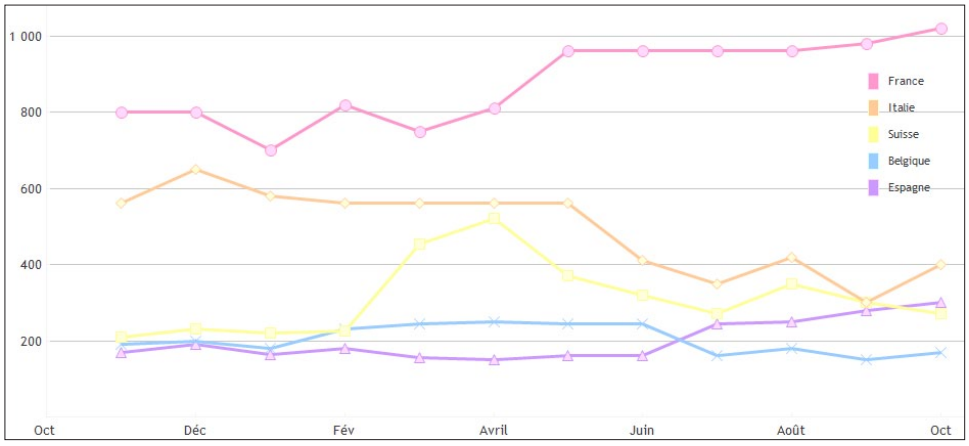
Selon les interfaces graphiques, le graphe peut alors sembler être juste "posé" dans l'interface, "flottant", sans lien avec le reste de l'IHM.

Pour supprimer cette impression de flottement, l'idée présentée ici est de simuler une "limite" au champ :
• en positionnant une image de fond de graphe (un fond hachuré par exemple). Dans ce cas, pensez à positionner un cadre autour du champ (pour forcer la délimitation).
• en ajoutant une ombre sous le graphe. L'ombre permet de donner du relief au graphe.



Graphe Courbe : aérer le contenu

Un graphe Courbe contient beaucoup d'informations : les axes et leurs graduations, la légende et les courbes.



Pour faciliter la lecture des informations essentielles, l'idée est d'aérer au maximum le graphe :

1. Cacher les axes

Si le but du graphe est d'afficher une tendance, c'est-à-dire si les valeurs ne sont pas primordiales, utilisez la fonction ***grAxeVisible*** pour cacher les axes. Cette fonction cache à la fois les axes et leurs valeurs.

Si les valeurs des axes sont importantes, vous pouvez "cacher" les axes en leur donnant la même couleur que le fond sur lequel le champ est posé. De cette façon, seul le trait symbolisant l'axe sera caché, les valeurs seront toujours visibles.

La couleur des axes peut être modifiée soit en édition (onglet "Style", élément "Couleur des axes), soit par programmation via la fonction ***grCouleur*** (avec la constante ***grCouleurAxe***).

2. Afficher uniquement les graduations nécessaires

Sur un axe où les valeurs ont une notion sémantique (par exemple, les mois d'une année), il n'est pas forcément nécessaire d'afficher toutes les graduations (une graduation sur 2 ou 3 suffit à identifier tous les mois).

Le pas des graduations peut être modifié soit en édition (onglet "Général", option "Graduation") soit par programmation via la fonction ***grGraduation***.

3. Inclure la légende dans le graphe

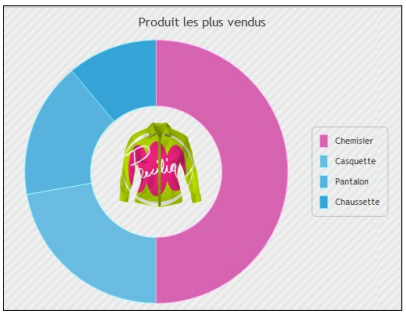
Dans un graphe Courbe, la légende est souvent importante et ne peut pas être cachée. Mais la zone occupée par la légende réduit d'autant la largeur du graphe.

Pour donner plus de places aux courbes, vous pouvez passer la légende en "positionnement manuel" puis la positionner n'importe où dans le champ Graphe :
• dans la description du champ Graphe, onglet "Détail", cliquez sur le bouton "Editer la légende du graphe" puis sélectionnez "Positionnement manuel",
• ouvrez le menu contextuel du champ Graphe et sélectionnez l'option "Editer la taille et position des zones internes". Le champ Graphe passe en mode "édition".
• sélectionnez la zone légende et déplacez-la où vous le souhaitez (de préférence près d'un bord du champ).
• sélectionnez la zone graphe et agrandissez-la pour occuper toute la largeur.

Mettre en avant une série

Pour aider à la compréhension du graphe, il peut être intéressant de mettre en avant une valeur importante (par exemple la portion la plus importante ou la courbe la plus significative) ou la valeur sélectionnée par l'utilisateur.

Pour un graphe Secteur, voici 2 possibilités de mise en évidence :
• excentrer la part voulue, via la fonction ***grSecteurPartExcentrée*** (ou ***grSMNPartExcentrée*** pour un graphe SMN).
• appliquer une couleur "vive" sur la part voulue et des tonalités opposées pour les autres parts.



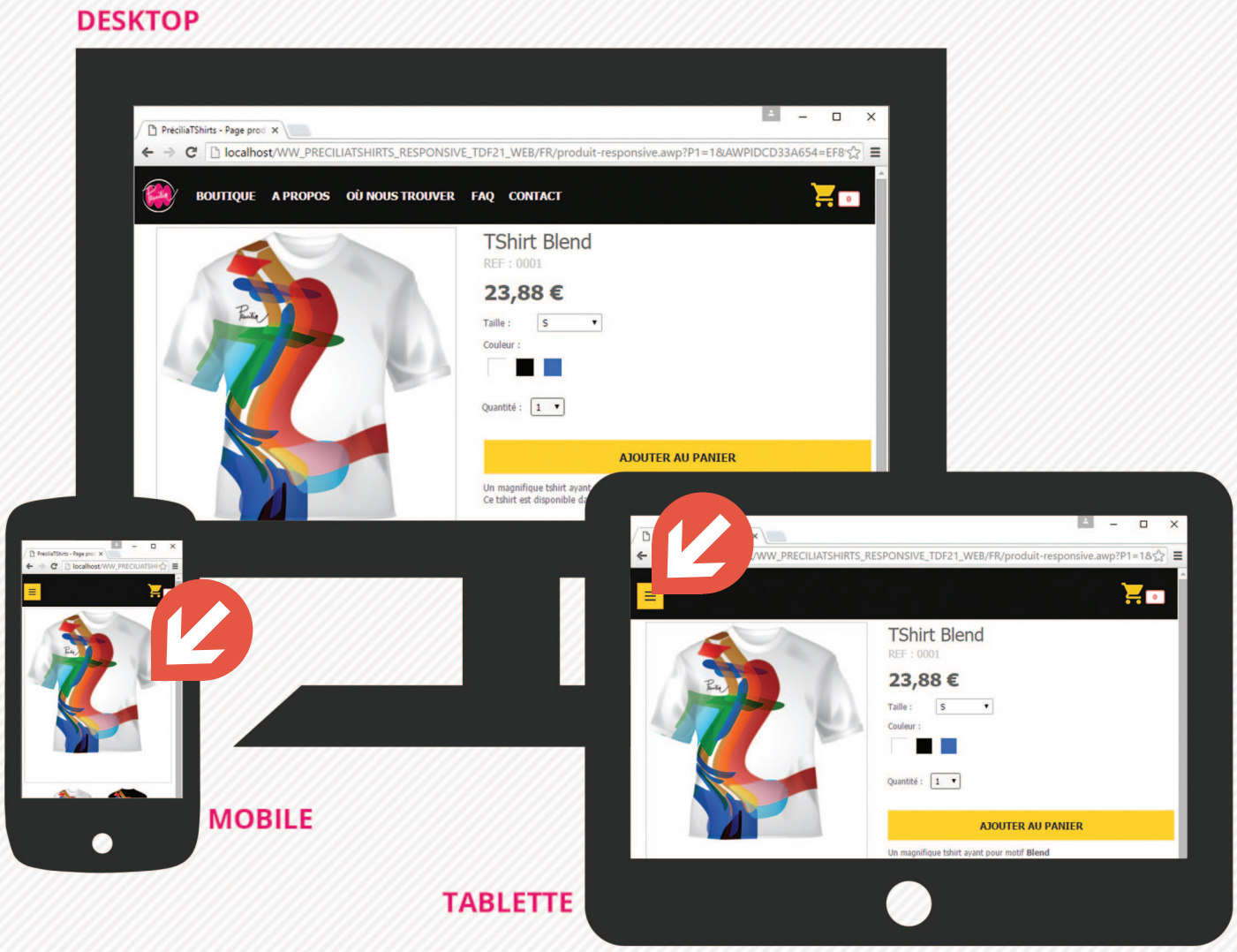
Pour un graphe Courbe, la fonction ***grEpaisseurTrait*** permet de modifier l'épaisseur d'une courbe, la mettant ainsi en avant par rapport aux autres.

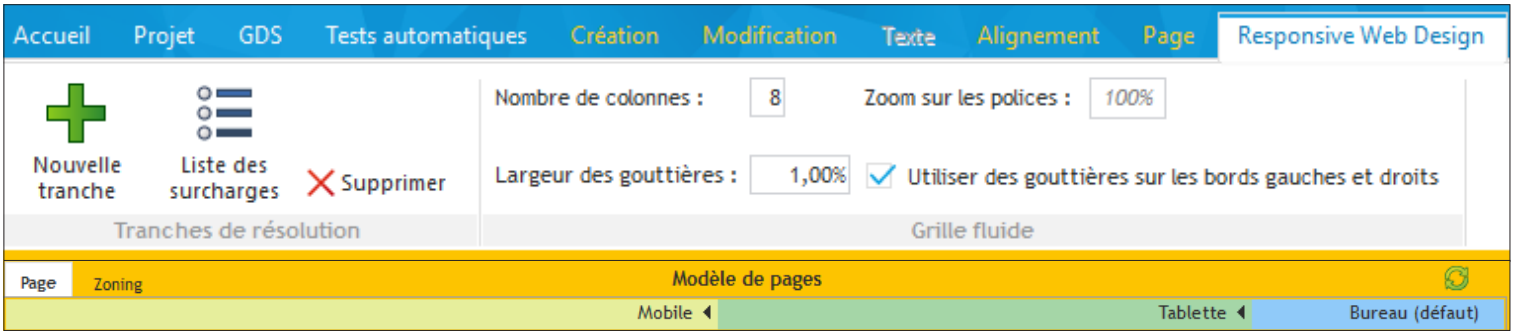
■

RESPONSIVE WEB DESIGN MÉTHODOLOGIE

En partant d'un existant ou d'une nouvelle page, une bonne méthode permet de réaliser rapidement une page Responsive Web Design fonctionnant dans les différents types de périphériques : mobile, tablette ou bureau (desktop).

Un même site sur les différents types de périphériques.
Dans l'affichage "tablette", le menu laisse la place au bouton "hamburger" qui déroule le menu.
Dans l'affichage "mobile", la mise en page a été complètement modifiée pour un affichage plus en hauteur.





Projet d'illustration

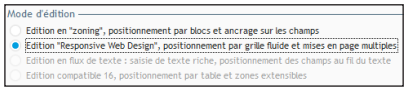
Le projet utilisé pour illustrer ce sujet est l'exemple "WW_PreciliaTShirts_Responsive_TDF21" (présent sur le DVD).

Commençons par les modèles de pages

Le développement d'une page commence souvent à partir d'un modèle de pages. Ce modèle peut être :
• vierge,
• sélectionné parmi les modèles prédéfinis de WEBDEV,
• un modèle existant dans le projet.

Dans le cas d'un modèle responsive sur un projet existant, plusieurs cas peuvent se présenter :
1. Les modèles de pages responsive proposés par défaut ne correspondent pas au besoin. Il est alors préférable de partir d'un modèle vierge (plutôt que de supprimer des éléments).

2. Un modèle WEBDEV est disponible, mais n'est pas responsive. Il est alors possible de le changer en modèle responsive directement depuis sa description. Le modèle existant servant alors de référence pour la version "bureau".



3. Vous disposez d'un modèle existant, mais il n'est pas responsive. Il est alors conseillé d'enregistrer une copie de ce modèle dans le projet pour en créer une version responsive, sans pour autant associer toutes les pages du projet à ce modèle. En effet, sur un projet important, le passage en responsive s'effectue généralement par groupes de pages, au fil du temps.

Des zones ?

Le principe des zones de mises en page est disponible lors de la création de pages responsive. Il est ainsi conseillé de structurer les pages et les modèles responsive (pied de page, zone de contenu, etc.).

Dans l'exemple "WW_PreciliaTShirts_Responsive_TDF21", pour respecter une mise en page similaire, le modèle de pages contient donc :
• une zone d'entête contenant le menu et le panier,
• une zone de contenu,
• un pied de page.

Tout comme une page classique, les zones disposent de possibilités de mise en page : couleur ou image de fond, mode d'affichage, etc.

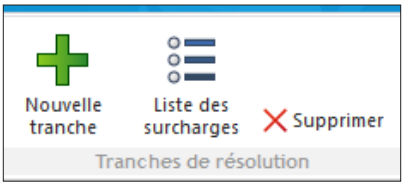
Spécificité : les pages et les modèles de pages responsive n'ont pas de hauteur. En cas de réalisation d'une zone "pied de page", la zone sera affichée non pas en bas du navigateur, mais sous le contenu. De même, l'espace entre le contenu et le pied de page ne sera pas figé puisque les champs peuvent avoir des hauteurs différentes selon les tranches.

1 Concept 1 : Les tranches

Les tranches représentent les résolutions des matériels les plus courants.

Par défaut, 3 tranches sont définies :
• bureau (pour les sites affichés sur un écran d'ordinateur de bureau),
• tablette,
• mobile.

Il est possible de modifier ces tranches, de les supprimer ou d'en ajouter.



Changer de tranche

Un clic sur une tranche permet de visualiser directement la page en cours dans cette tranche.

Certaines propriétés des champs peuvent êtres redéfinies / surchargées dans une tranche donnée (voir "Surcharges et menu contextuel "Responsive Web Design"").

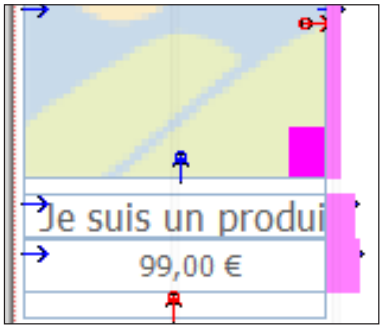
Redimensionnement intermédiaire

Pour tester le comportement intermédiaire des champs entre deux tranches, il est possible en édition d'utiliser la poignée de redimensionnement.

Cette poignée permet de redimensionner la taille du navigateur sous l'éditeur afin de visualiser dynamiquement l'adaptation des champs à une taille du navigateur donnée.

Pour diminuer ou augmenter la taille du navigateur sous l'éditeur, il suffit de cliquer sur la double flèche et de déplacer la souris.

Il est alors important d'être particulièrement vigilant aux tailles minimales des champs, ainsi qu'aux risques de superpositions !

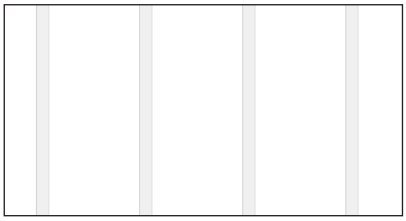


Ici : il faut trouver une solution, car en exécution les champs de la zone répétée ne pouvant pas être plus grands que la zone répétée, il y aura un affichage "non maîtrisé" dépendant du navigateur.

RESPONSIVE WEB DESIGN MÉTHODOLOGIE (SUITE)

2 Concept 2 : la grille

C'est une spécificité de l'édition de pages responsive. Les pages possèdent une grille en colonnes permettant de positionner les champs dans la page.



Cette grille est à la fois :
• une aide au positionnement des champs,
• le reflet de l'effet des ancrages sur la taille en largeur des champs.
Elle n'a aucune existence réelle dans la page Web.

La grille peut se désactiver lors du déplacement des champs en utilisant la touche SHIFT pour positionner précisément un champ.

3 Concept 3 : du bureau au mobile

C'est la **règle d'or à appliquer lors de la création d'une page ou d'un modèle responsive** : configurez d'abord l'affichage en mode "Bureau", puis en mode "Tablette" et enfin en mode "Mobile".

En effet, les modifications de positionnement ou de propriétés personnalisables par tranches s'effectuent en cascade dans cet ordre. Ainsi, une modification appliquée dans la tranche tablette sera automatiquement reportée dans la tranche mobile.

4 Outil 1: surcharges et menu contextuel "Responsive WebDesign"

Principe

Le principe de l'édition des propriétés des champs est très simple :
• par défaut, les propriétés des champs sont globales (par exemple le masque de saisie d'un champ),
• certaines propriétés des champs sont personnalisables par tranche, avec un mécanisme de cascade de la tranche "Bureau" vers la tranche "Mobile" (par exemple le nombre de colonnes d'un champ Zone répétée).

Édition globale

L'édition des propriétés générales des champs s'effectue par la fenêtre des 7 onglets ou par le modifieur du champ.
Seules les propriétés ayant l'icône suivante dans la fenêtre d'édition sont définies par tranche :

Il est donc possible de modifier les propriétés générales d'un champ depuis n'importe quelle tranche.

Personnalisation par tranche

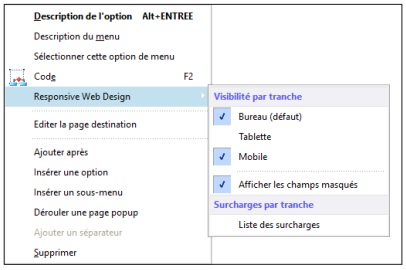
Les options personnalisables par tranches peuvent être modifiées :
• depuis la fenêtre des 7 onglets pour les options ayant l'icône :

• depuis le menu contextuel "Responsive WebDesign" sur le champ concerné.

La modification s'effectue alors selon la tranche en cours d'édition et se propage à la ou les tranches suivantes (dans l'ordre Bureau, Tablette, Mobile).

Visibilité des champs par tranche

La visibilité est une surcharge "spécifique". Il est possible de définir depuis le menu contextuel la visibilité d'un champ dans toutes les tranches. C'est la seule surcharge qui ne se propage pas aux tranches suivantes.



Position des champs par tranche

Il est possible de déplacer un même champ dans les tranches. Dans la page "produit_responsive" de l'exemple "WW_PreciliaTShirts_Responsive_TDF21", la tranche "Mobile" est organisée en hauteur, alors que les tranches "Bureau" et "Tablette" sont en largeur.

5 Outil 2 : barre de navigation

Le champ Barre de navigation permet de créer une zone de menu s'adaptant facilement à la taille du navigateur.
Le champ Barre de navigation peut contenir tout type de champs.

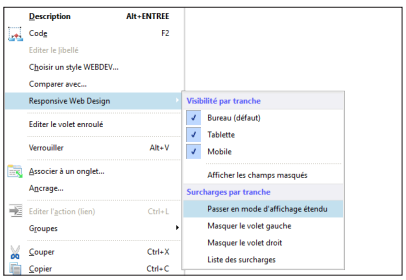
Important : il n'est pas nécessaire que tous les champs de l'entête du site soient placés dans la barre de navigation. En revanche, en mode "mobile", la partie déroulée ne pourra pas être plus grande que le champ Barre de navigation.

3 zones

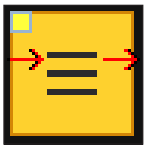
La barre de navigation est composée de 3 zones : gauche, centre, droite. Ces 3 zones peuvent être masquées ou affichées dans chaque tranche.

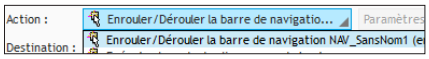
La zone "Centre" est la zone la plus importante et est destinée à contenir le menu de l'application.

Par défaut, la zone "Gauche" contient un bouton ayant une action spécifique permettant d'enrouler ou de dérouler le menu contenu dans la zone centrale dans le mode d'affichage "Compact". Le bouton peut bien entendu être déplacé ou changé de zone (généralement pour aller dans le zone "droite").



Ce bouton est appelé bouton "Hamburger" dans l'aide en ligne (en rapport à son aspect graphique). Ce bouton peut être remplacé, c'est son action qui importe : "enrouler / dérouler la barre de navigation".





Il est également possible d'effectuer cette opération par programmation grâce à la propriété WLangage navigateur ***..Enroulé***.

Compact ou étendu

Le champ Barre de navigation dispose de 2 modes d'affichage qui lui sont propres :

• le mode Etendu : les 3 zones sont affichées par défaut. Le mode étendu permet de gérer l'affichage dans la tranche "Bureau" par défaut.

• le mode Compact : la zone "Centre" peut être déroulée (lors de l'appui sur un bouton). Le mode compact permet de gérer par défaut l'affichage dans les tranches "Tablette" et "Mobile".

Dans l'exemple "WW_PreciliaTShirts_Responsive_TDF21", le menu étant "petit", le mode étendu pourrait être utilisé pour la tranche "Tablette" en plus de la tranche "Bureau". Seule la tranche "Mobile" utiliserait alors le mode "Compact". C'est vous qui choisissez le mode du menu selon les tranches.

Attention : pour que le fonctionnement en mode "Compact" de la barre de navigation soit possible, il est obligatoire que le menu soit un menu CSS.

Dans l'exemple "WW_PreciliaTShirts_Responsive_TDF21", le projet d'origine contient un menu "Images". Lors de la création du modèle en mode Responsive, il a donc été nécessaire de :

• passer le menu en mode CSS (depuis la description du champ),

• supprimer l'option "superposable" du menu (spécifique au menu "Images").

6 Outil 3 : menu CSS

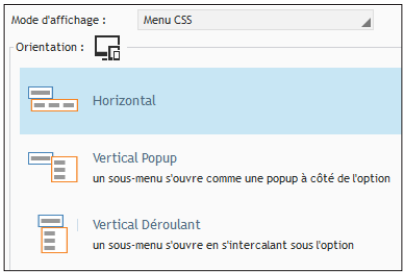
Le menu CSS et la gestion de l'orientation

L'intérêt du menu CSS dans le cas d'une page responsive est qu'il dispose de 3 modes d'affichage :

• horizontal : c'est le mode d'affichage idéal pour la tranche "Bureau",

• vertical popup : c'est un mode d'affichage pratique en cas de nombreuses options. C'est un mode d'affichage qui peut être utile pour la tranche "Bureau" ou "Tablette".

• vertical déroulant : c'est le mode d'affichage idéal pour la tranche "Mobile".



L'orientation du menu CSS étant paramétrable par tranche, il est ainsi possible de proposer un même menu dans les différentes tranches tout en ayant un affichage adapté.

Édition du menu CSS

Pour éditer le menu CSS lorsque la barre de navigation est en mode "Compact", il est nécessaire de passer par le champ Barre de navigation pour provoquer l'édition du menu.

Il est ensuite possible de personnaliser l'affichage du champ Menu.

Important : l'exemple "WW_PreciliaTShirts_Responsive_TDF21" contient uniquement le menu dans la zone centrale et un logo, mais il serait possible d'avoir d'autres champs. Tous les champs présents dans la zone centrale de la barre de navigation seront alors présents dans le volet en mode compact. Si des champs doivent être masqués, ils peuvent l'être comme pour les autres tranches.

Surcharges d'options

Pour surcharger une option ou une propriété d'un champ, il suffit de se placer dans la tranche la plus grande où la modification sera effectuée (elle sera automatiquement propagée sur les tranches inférieures) et de faire la modification. La modification est alors affichée dans la liste des surcharges de la tranche.

Dans l'exemple "WW_PreciliaTShirts_Responsive_TDF21", dans la tranche "Tablette", il pourrait ainsi être choisi de surcharger l'option "FAQ" pour ne pas l'afficher dans cette tranche (par exemple si la page destination n'est pas disponible dans cette tranche ou pour laisser plus de place pour les autres options).

7 Outil 4 : gestion de la taille des champs

Par défaut, aucune largeur minimale n'est définie pour les champs, à l'exception du champ Bouton (dont la largeur minimale est fixée à la largeur en édition). Si le navigateur se réduit, le champ se réduit proportionnellement.

Pour fixer une taille minimale, dans la fenêtre de description du champ, onglet "IHM", définissez la largeur minimale souhaitée :

• pas de taille minimale,

• taille en édition,

• valeur personnalisée saisie directement.

8 Outil 5 : gestion des ancrages

Le mécanisme des ancrages dans une page responsive est différent de celui d'une page "classique". En effet, dans une page "classique", les ancrages permettent de définir le comportement lors de l'agrandissement de la page.

Dans une page responsive, les ancrages permettent de définir le comportement lors de l'agrandissement ou la réduction de la largeur de la page. Ce changement de largeur est réparti par les colonnes de la grille.

Ancrages au niveau de la page

Pour définir les ancrages dans une page responsive, depuis le menu contextuel de la page, sélectionnez l'option "Ancrage". La fenêtre de description des ancrages de la page apparaît.

Les options disponibles au niveau de la page sont les suivantes :

• utiliser la taille d'édition : cette option permet de fixer la largeur ou la hauteur de la page. La hauteur ou la largeur prise en compte est celle spécifiée dans l'onglet "IHM" de la fenêtre de description de la page.

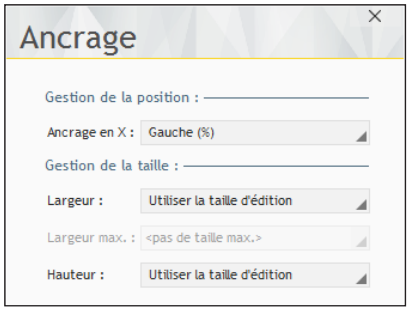
• adapter à la grille : grâce à cette option, la page se contracte ou s'étire selon la taille du navigateur.

• adapter à la grille sans agrandissement : si cette option est sélectionnée, la page se contracte et s'étire au maximum de la taille d'édition (hauteur ou largeur définie dans l'onglet "IHM" de la description de la page).

RESPONSIVE WEB DESIGN MÉTHODOLOGIE (SUITE)

Ancrages au niveau des champs

Les ancrages au niveau des champs sont plus précis que ceux au niveau de la page.



Pour définir les ancrages au niveau des champs, depuis le menu contextuel du champ, sélectionnez l'option "Ancrage". La fenêtre de description des ancrages du champ apparaît.

Les options disponibles sont les suivantes :

1. Ancrage en X

• Gauche : l'espace à gauche du champ (par rapport au champ précédent ou au bord de la page) est conservé proportionnellement lors de la modification de la largeur de la page.
• Droite : l'espace à droite du champ (par rapport au champ précédent ou au bord de la page) est conservé proportionnellement lors de la modification de la largeur de la page.
• Centre : cette option place le champ au centre de la page. Cette option peut être utilisée pour un seul champ uniquement. Si deux champs sont présents sur la même ligne, le centrage est ignoré.

2. Largeur

• Adapter à la grille (option par défaut) : cette option permet de conserver les proportions existantes sous l'éditeur lorsque la page s'étire ou se contracte.
• Utiliser la taille d'édition : cette option permet de fixer la largeur du champ quelle que soit la taille de la page.
• Adapter au contenu : cette option permet de réduire la largeur du champ au minimum pour que son contenu soit visible.

3. Largeur max

Cette option permet de définir une largeur maximale si la largeur est adaptée à la grille ou adaptée au contenu. Ainsi, le champ pourra avoir une largeur limite. Les options disponibles sont :

• pas de taille max : la largeur du champ n'est pas limitée.
• taille en édition : la largeur maximale du champ sera celle définie en édition.

4. Hauteur

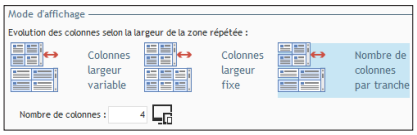
• Utiliser la taille d'édition : cette option permet de fixer la hauteur du champ quelle que soit la taille de la page.
• Adapter au contenu : cette option permet de réduire la hauteur du champ au minimum pour que son contenu soit visible.

9 Outil 6 : zones répétées

Le champ Zone répétée est l'élément fondamental de l'affichage de données dans une page responsive.
Dans une page responsive, il est possible de définir le nombre de colonnes par tranche.

Dans l'exemple "WW_PreciliaTShirts_Responsive_TDF21", il a ainsi été possible de définir :

• 3 colonnes pour la zone répétée dans la tranche "Bureau" (avec un bandeau à droite),
• 3 colonnes pour la zone répétée dans la tranche "Tablette",
• 2 colonnes pour la zone répétée dans la tranche "Mobile".



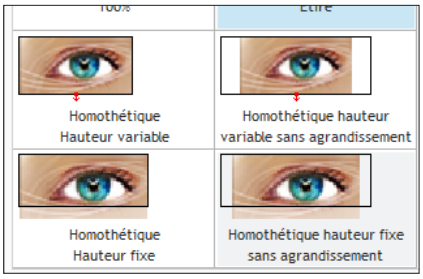
Ensuite, grâce aux ancrages sur les champs de la zone répétée, il est possible de définir le comportement pour les cas intermédiaires.

Dans l'exemple "WW_PreciliaTShirts_Responsive_TDF21", il a été choisi de ne pas augmenter les éléments mais plutôt de les espacer pour conserver une taille constante quel que soit l'affichage.

10 Outil 7 : modes d'affichage des images

WEBDEV 21 dispose de nouveaux modes d'affichage des champs Image qui sont particulièrement utiles dans le cas d'une page responsive :

• Homothétique à hauteur variable (avec ou sans agrandissement),
• Homothétique à hauteur fixe (avec ou sans agrandissement).



Ces 4 nouveaux modes d'affichage, sans retour au serveur, permettent de redimensionner des images côté navigateur et donc d'avoir une même image pour différentes tranches (même en cas de dimensions différentes).

Note : il est possible de surcharger une image dans une tranche pour des questions d'optimisation principalement, mais aussi pour gérer des cadrages spécifiques.

Plusieurs avantages

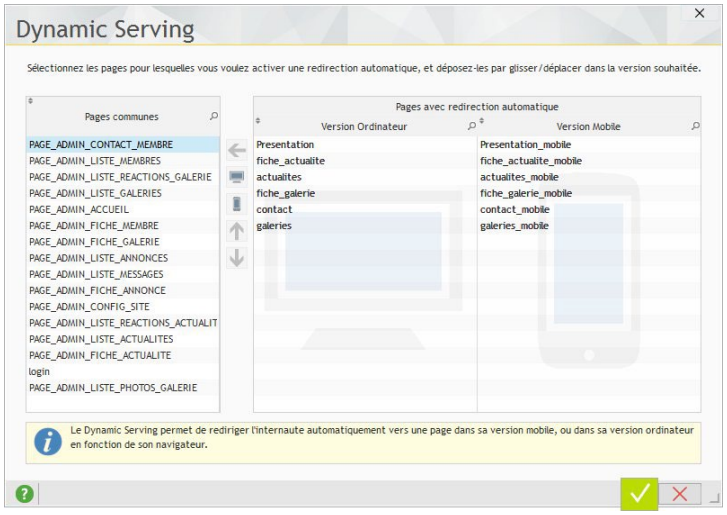
Les cas d'utilisations d'images dans ces nouveaux modes homothétiques sont multiples :

• il est possible d'utiliser une même image pour des dimensions différentes (par exemple dans la zone répétée de la page boutique, mais aussi dans la page produit),
• la largeur de l'image peut être adaptée en fonction de la taille de largeur de la page,
• en cas de redimensionnement dans une tranche, l'image restera cohérente (elle ne sera pas déformée).

■

LE DYNAMIC SERVING

Le "Dynamic serving" permet d'accéder à deux pages différentes (une page pour PC, une page pour mobile) à partir d'une même URL.
C'est une solution alternative (voire complémentaire) au "Responsive Web Design" pour rendre un site traditionnel "mobile friendly".
Avec le "Dynamic Serving", il n'est pas nécessaire de modifier les pages existantes : il suffit d'en rajouter pour les mobiles.



La configuration du Dynamic Serving est très simple : il suffit de mettre en vis à vis la page "PC" et la page "Mobile" associée dans la liste des "Pages avec redirection "automatique".

Les sites "mobile friendly"

La navigation depuis des terminaux mobiles est de plus en plus fréquente et devient plus importante que la navigation depuis un ordinateur traditionnel.

Google a donc mis en avant dans son classement (ranking) les sites compatibles avec les terminaux mobiles.

Pour cela, Google fournit un test en ligne permettant de tester la compatibilité mobile d'un site Web (on parle de site "mobile friendly") : "https://www.google.com/webmasters/tools/mobile-friendly/?hl=fr".

Ce test permet notamment de tester :
• la lisibilité des textes sur un mobile,
• la proximité des liens pour une utilisation tactile,
• le contenu tel qu'il est vu par le robot de Google (Googlebot),
• etc.

Google fournit également un site reprenant un ensemble de conseils pour les sites mobiles : "https://developers.google.com/webmasters/mobile-sites/?hl=fr".

Le "Responsive Web Design" est une des solutions pour réaliser un site "mobile friendly", mais la mise en place est souvent longue en cas d'existant. WEBDEV 21 propose une autre solution technique pour qu'un site devienne mobile friendly en ajoutant spécifiquement des pages mobiles : le "Dynamic Serving".

Le Dynamic Serving

Le "Dynamic serving" est une technique dans laquelle la même adresse (URL) mène à 2 pages différentes : une page pour PC, une page pour mobile.

Selon le matériel (PC ou mobile) qui accède à la page, la page adéquate sera affichée.

C'est une solution alternative (ou complémentaire) au "Responsive Web Design" pour rendre un site traditionnel "mobile friendly".

Avec le "Dynamic serving", il n'est pas nécessaire de modifier les pages existantes : il suffit d'en rajouter pour les mobiles (qui peuvent être des pages "Responsives").

Mise en œuvre

Principe

Comme indiqué précédemment, le but est d'avoir dans le projet 2 pages pour chaque page :
• une page mobile ("contact_mobile" dans l'exemple "WW_PreciliaTShirts TDF21"),
• une page "classique" ("contact" dans l'exemple "WW_PreciliaTShirts TDF21") correspondant à la page actuelle.

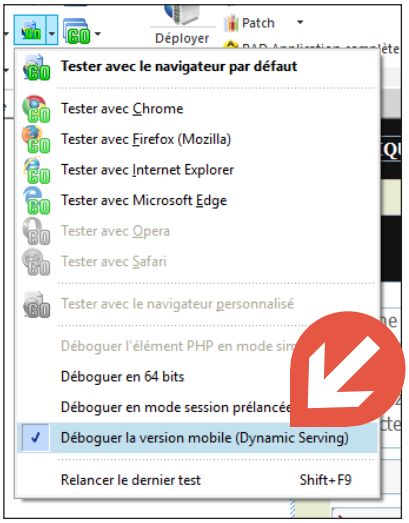
La première étape consiste donc généralement à :
• enregistrer une copie de la page existante pour en faire une page mobile,
• dissocier le modèle ou utiliser un modèle mobile spécifique,
• définir la taille de l'écran cible dans la fenêtre de description de la page,
• mettre en page le contenu de la page mobile.

Configurer le Dynamic Serving

Une fois les deux pages disponibles, il suffit d'indiquer au serveur d'application WEBDEV 21 les équivalences de pages mobiles et "ordinateur / bureau".
Pour cela, une interface dédiée est disponible dans le volet "Projet" : dans le groupe "Web", déroulez "Référencement" et sélectionnez "Dynamic Serving".

Tester le site

Puisque désormais une même URL peut représenter deux pages, WEBDEV 21 propose une option permettant de tester spécifiquement la version mobile en "Go".



■

NOUVEAUTÉS MOBILE

WINDEV Mobile vous permet de développer pour tous les terminaux mobiles : tablettes et téléphones, sous iOS, Android, Universal Windows 10 App, ...
La version 21 apporte son lot de nouveautés bien pratiques.
En voici quelques-unes.

Projet d'illustration

Le projet utilisé pour illustrer ce sujet est l'exemple "WM AirB & Cook TDF21" (présent sur le DVD).

1 Identification Facebook

De plus en plus d'applications mobiles proposent de s'identifier à l'aide d'un compte Facebook.
C'est une fonctionnalité très pratique : cela évite à l'utilisateur de créer un nouveau compte, avec un nouveau mot de passe à retenir.
De plus, l'identification par Facebook permet de renseigner automatiquement les informations principales du compte (nom, avatar, etc.).

Important
Bien entendu, l'identification par Facebook nécessite d'autoriser l'application à accéder aux informations du réseau social.
Une fenêtre spécifique [**Fig. 1**] est affichée par l'application Facebook pour autoriser l'application courante à accéder aux informations.

Avant toute autre opération, il est nécessaire de déclarer (et ouvrir) une connexion au service Facebook à l'aide d'une variable de type ***fbSession***. Il est nécessaire de renseigner :
• l'identifiant et la clé secrète de l'application. Pour savoir comment récupérer ces informations, consultez la documentation associée : "http://doc.pcsoft.fr/?1000021926".
• les permissions d'accès (information du profil, emails, informations sur les amis...).

```
// Paramètre la session Facebook
FBSession est une fbSession
FBSession.AppID = "<id>"
FBSession.AppSecret = "<secret>"
FBSession.Permission = ...
  [fbInfoProfil, fbEmail, fbInfoAmis]
// Ouvre la session
FBOuvreSession(FBSession)
```

La récupération des informations du compte peut ensuite être effectuée par la fonction WLangage ***FBRécupèreInformationUtilisateur***.

```
// Récupération des informations
// du compte Facebook
// - l'identifiant
sId = ...
 FBRécupèreInformationUtilisateur(
  FBSession, fbIdentifiant)
// - le nom
sNomAffichage = ...
 FBRécupèreInformationUtilisateur(
  FBSession, fbNom)
// - l'email de connexion
sEmail = ...
 FBRécupèreInformationUtilisateur(
  FBSession, fbEmail)
```

Pour récupérer les amis Facebook, la manipulation est un peu plus complexe.
En effet, il n'existe pas de fonctions WLangage pour récupérer directement les amis.
Il est donc nécessaire de passer par la fonction ***FBRequête***.
Cette fonction vous permet d'exécuter n'importe quelle requête auprès du réseau social et donc de récupérer toutes les informations que vous souhaitez ... comme les amis.

```
tabData est un tableau de
  STFacebookDonnéesUtilisateur

// URL d'interrogation des amis
sURL = "https://graph.facebook.
com/me/friends?limit=5000"

// Interroge l'URL
bufJSON = ...
  FBRequête(FBSession, sURL)
// Convertit la réponse pour
// la traiter
sJSON = UTF8VersChaîne( bufJSON)
```

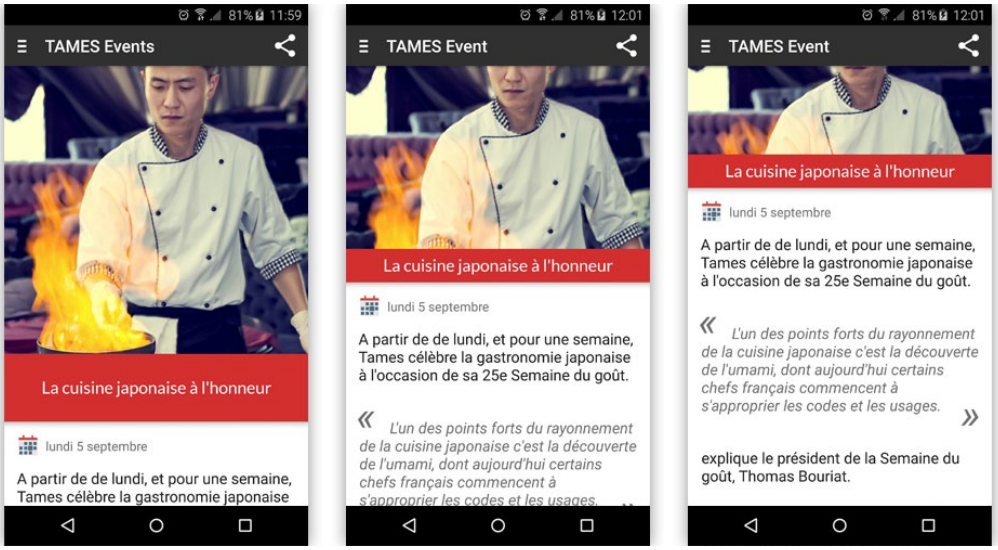
Fig. 1

```
// Récupère un tableau d'utilisateurs
tabData = ...
  FacebookParseJsonAmis(sJSON)
```

La procédure ***FacebookRécupèreAmis*** de l'exemple est une fonction générique que vous pouvez réutiliser dans vos applications mobiles.

2 Images rondes

Pour appliquer un effet "image ronde" à un champ Image, c'est très simple : dans la description du champ Image, onglet "Style", il suffit de sélectionner l'élément "Cadre" et de choisir un cadre rond !



La mise en place de l'effet parallaxe se fait en quelques clics et apporte une touche de dynamisme et de fluidité à l'application. Ne vous en privez pas !

Vous pouvez même choisir un cadre "Personnalisé" et faire ce que vous voulez (par exemple un arrondi sur deux coins uniquement...) !

3 GPS Low energy

Le GPS Low Energy permet à l'application d'être "prévenue" lorsque le téléphone bouge de façon notable.

Cette fonctionnalité permet d'avoir des informations constantes sur la position, sans vider la batterie du téléphone (comme le ferait l'activation classique du GPS, qui teste la position en permanance).

Les fonctions WLangage ***géoSuiviActive*** et ***géoSuiviProcédure*** permettent respectivement d'activer le suivi GPS et de définir la procédure à appeler à chaque changement notable de position :

• sous iOS, le système appelle la procédure au minimum tous les 500 m ou toutes les 5 minutes.

• sous Android, les changements dépendent également de la vitesse ; cela peut varier de toutes les 30 secondes à toutes les 10 minutes.

```
// Active le suivi
géoSuiviActive(ProcédureActivation)

// Définit la procédure de suivi.
// ProcédureSuivi sera appelé à
// chaque changement notable de
// position
géoSuiviProcédure(ProcédureSuivi)

// Cette procédure recevra en
// paramètres la position courante
PROCEDURE ProcédureSuivi(
  mPosition est une géoPosition)
// Traitement sur la position
...
```

Important : pour utiliser cette fontionnalité, le module "Google Play services" doit être installé dans le SDK Android.

4 Effet parallaxe

L'effet parallaxe est un terme générique qui correspond au déplacement de différents champs à différentes vitesses et sur différents niveaux. Cet effet permet de donner très simplement un effet de profondeur dans vos fenêtres.

Pour créer un effet parallaxe dans vos fenêtres, ouvrez la description d'un champ, onglet "IHM" et cliquez sur le bouton "Parallaxe : aucun".

Plusieurs effets par défaut sont disponibles :

• Aucun parallaxe : le champ se déplace en hauteur selon le déplacement de l'ascenseur de la fenêtre.

• Effet Material Design sans réduction : le champ se déplace en hauteur proportionnellement au déplacement de l'ascenseur de la fenêtre.

• Effet Material Design avec réduction : le champ se réduit en hauteur proportionnellement au déplacement de l'ascenseur de la fenêtre.

• Champ fixe : le champ est fixe, il ne se déplace pas dans la fenêtre. Il est toujours visible lors du déplacement de l'ascenseur de la fenêtre.

• Personnaliser l'effet : vous saisissez le taux de parallaxe Y (vitesse de déplacement du champ) et le taux de parallaxe Hauteur (vitesse de réduction du champ) qui vous conviennent.

Dans la popup de choix de l'effet, la partie droite offre un aperçu du rendu du paramétrage courant.

Bon à savoir

L'effet de parallaxe peut également être défini par programmation via les propriétés ***..TauxParallaxeHauteur*** et ***..TauxParallaxeY***.

5 Le champ Planning

Le champ Planning permet d'afficher et de manipuler des plannings automatiquement. Il permet de gérer l'affichage de ressources multiples.

Pour remplir un champ Planning, vous pouvez utiliser :

• le binding, directement avec un fichier de données, une requête ou même une variable,

• les fonctions WLangage ***PlanningXXX***.

Depuis la fenêtre de description du champ, onglet "Détail", vous pouvez paramétrer complètement le champ :

• le nombre de jours à afficher,

• le mode d'affichage des ressources,

• les masques d'affichage,

• la granularité,

• les dimensions des zones,

• les possibilités de saisie et modifications de l'utilisateur final,

• la plage horaire à afficher,

• ...

Bien entendu, comme pour les autres champs, le look du champ Planning peut être entièrement personnalisé depuis la fenêtre de description, onglet "Style".

■

UNIVERSAL WINDOWS 10 APP

WINDEV Mobile vous permet de créer, à partir du même code source, des applications pour iOS, Android et Universal Windows 10 App (nouveau en version 21).

Windows 10

Présentation

Windows 10 regroupe deux modes de fonctionnement :
• le mode desktop / bureau, utilisé sur les PC. Ce mode est utilisé pour les applications de bureau "classiques".
• le mode "Tuiles" qui permet d'utiliser des "Universal Windows 10 App".

Ce mode de fonctionnement exécutant des applications "Universal Windows 10 App" se retrouve sur les tablettes Surface sous Windows 10 et les smartphones sous Windows 10 Phone.

En créant une application "Universal Windows 10 App" avec WINDEV Mobile 21, cette application peut donc s'exécuter :
• sur des smartphones Windows Phone 10,
• sur des tablettes Windows 10,
• sur des ordinateurs de bureau dans la partie "tuile".

Base de données : HFSQL

Les applications WINDEV Mobile 21 "Universal Windows 10 App" peuvent exploiter les bases de données HFSQL :
• en version Classic (embarquée sur le périphérique),
• en version Client/Serveur.

Gestion des périphériques : utilisation combinée d'ancrages et d'agencements

Pour que les applications WINDEV Mobile 21 "Universal Windows 10 App" fonctionnent sur les différents périphériques du marché, il suffit de combiner deux fonctionnalités standards :
• les ancrages, qui permettent de définir la taille des champs en fonction de la taille d'écran,
• les agencements, qui permettent d'organiser les champs en fonction d'un périphérique (par exemple une tablette ou un boîtier Continuum).

Lors de l'exécution sur un périphérique, l'application utilise l'agencement le plus proche des spécificités du périphérique. Ensuite, les ancrages sont appliqués pour s'adapter au mieux au périphérique.

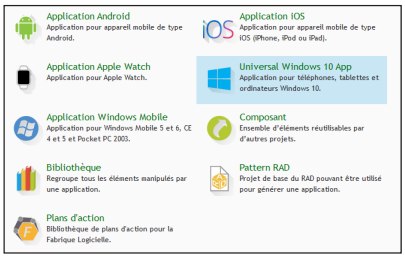
Il est possible de changer par programmation l'agencement utilisé avec la fonction ***FenChangeAgencement*** du WLangage.

Développement / Prérequis

Pour développer une application en mode Universal Windows 10 App, les prérequis sont les suivants :
• le poste de développement doit être sous Windows 10,
• un SDK Windows 10 spécifique doit être installé ("https://dev.windows.com/fr-fr/downloads/windows-10-sdk").
Consultez l'aide en ligne pour plus d'informations sur le sujet : "http://doc.pcsoft.fr/?1410086574".
• Le poste de développement doit être passé en mode "développeur". Le mode développeur permet d'installer des applications sans passer par le store. Le mode développeur s'active par le menu "Pour les paramètres de développeur" de Windows 10.
• Même en mode développeur, il est nécessaire que l'application soit signée avec un certificat. En phase de test, ce certificat peut être "autosigné". En phase de production, ce certificat doit être acheté à un organisme reconnu. L'assistant de génération de l'application "Windows 10" vous permet de sélectionner un certificat déjà installé ou de créer un certificat spécifique.

Création d'un projet ou d'une configuration

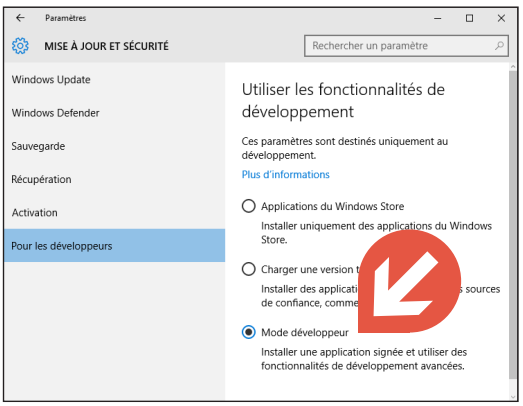
Comme pour une application iOS, Android ou autre, il est possible de créer :
• un projet spécifique pour Universal Windows 10 App,
• une nouvelle configuration sur un projet existant.

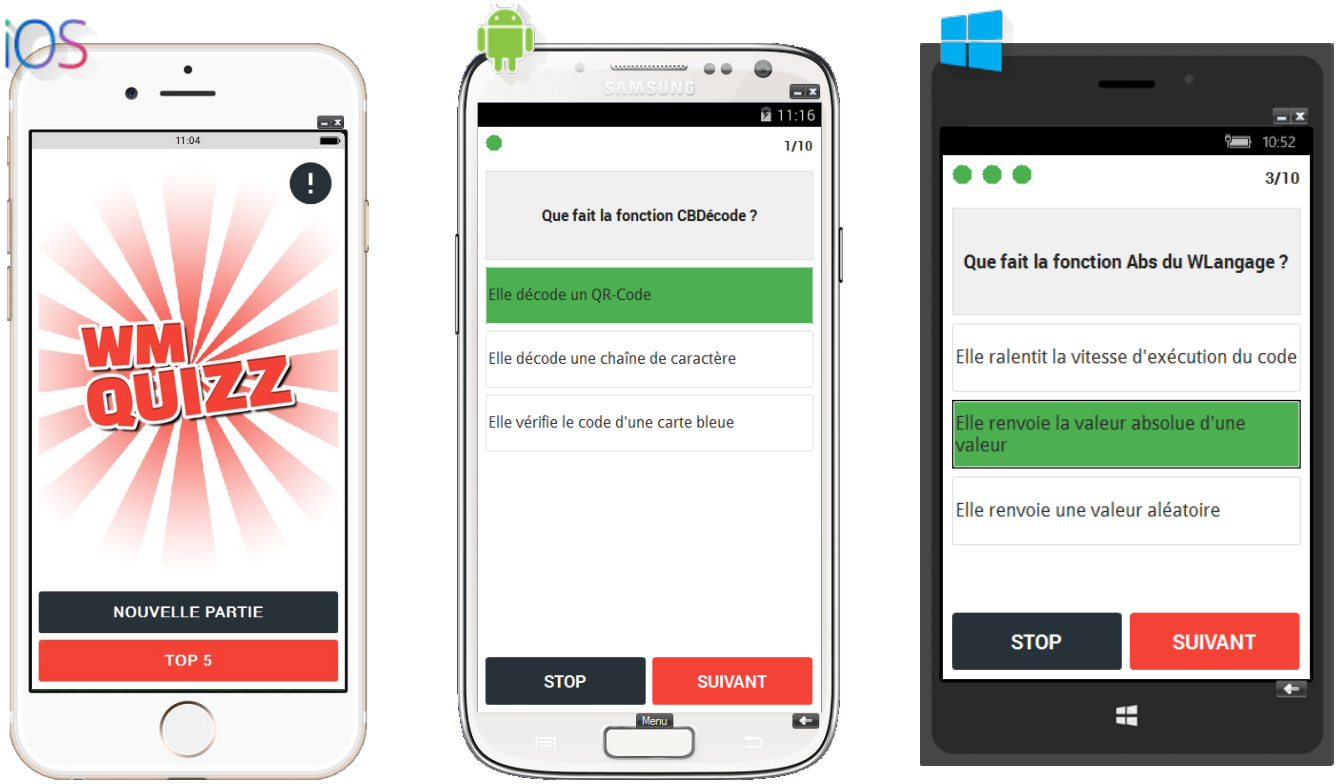


Ensuite, le développement se fait comme pour une application mobile classique (agencements, programmation WLangage, etc.).

Génération

Comme souvent au niveau des applications mobiles, l'assistant de génération propose quelques spécificités :
• le chemin du SDK doit être renseigné,
• les permissions nécessaires à l'application. doivent être fixées. Ces permissions sont automatiquement détectées par WINDEV Mobile 21 mais peuvent être adaptées à des cas spécifiques (utilisation de fonctionnalités en code natif par exemple).





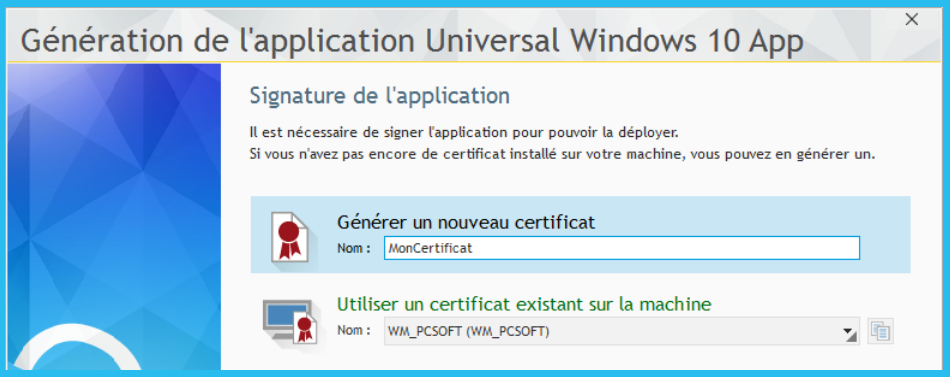
WM Quizz : testez vos connaissances sur les produits PC SOFT via une application disponible sur iOS, Android et Universal Windows 10 Apps !
(exemple livré en standard avec WINDEV Mobile)

Génération d'un certificat autosigné

L'exemple "WD Signature Exécutable TDF21" (livré sur le DVD) est un petit assistant utilitaire qui vous permet de créer rapidement et simplement un certificat autosigné.

Les certificats générés peuvent être utilisés pour signer :
• une application (Windows, Universal Windows 10 App, ...),
• une installation (autonome ou avec mise à jour réseau),
• un état,
• ...

Cet assistant reprend les étapes présentées dans l'article sur la signature d'exécutables de la LST 96 (pages 26-27).



L'application Universal Windows 10 App doit obligatoirement être signée via un certificat.
Vous avez la possibilité de générer un nouveau certificat ou d'utiliser un certificat de la machine.

DE WINDEV À WINDEV MOBILE

Passer d'une application Windows à une application mobile (Android, iOS...), c'est possible !
Il suffit de bien suivre les différentes étapes et de savoir comment traiter les différences de comportement.
Suivez le guide !

1 Préparer le projet WINDEV

Première étape pour une transition sereine vers le mobile : **séparez tout le code métier de l'IHM** dans votre application.

Cette étape est sans doute la plus importante mais c'est également la plus complexe selon l'importance de votre projet.

Attention : avant de vous lancer dans cette étape, n'hésitez pas à créer une branche GDS de votre projet en cours et à faire le découpage sur le tronc commun. Vous aurez ainsi la possibilité de mettre à jour votre application à partir de la branche.
Toutefois, n'oubliez pas de reporter toute correction faite dans l'application "originale" dans la nouvelle application "en cours de découpage" (pour éviter de perdre des corrections ou des nouvelles fonctionnalités dans le projet "découpé").

2 Créer le projet WINDEV Mobile

Sous WINDEV Mobile, créez un nouveau projet et suivez les étapes de l'assistant.

Si votre application WINDEV utilise une base de données, demandez à ce que l'application WINDEV Mobile utilise la même analyse.

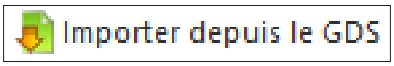
Si votre application WINDEV est dans un GDS (et nous vous le conseillons fortement), ajoutez le projet WINDEV Mobile dans le même GDS.

Lorsque l'assistant d'ajout au GDS vous le demandera, indiquez que vous souhaitez partager l'analyse avec l'autre projet.

3 Récupérer le code métier

Pour récupérer le code métier de l'application WINDEV, c'est très facile : il vous suffit de partager les classes et les collections de procédures depuis l'autre projet.

Depuis le ruban, sous le volet "GDS", dans le groupe "Projet", cliquez sur "Importer depuis le GDS". Une fenêtre de sélection s'affiche.



Positionnez-vous sur le répertoire du projet WINDEV, sélectionnez les éléments à partager et cliquez sur "Partager".

4 Récupérer l'IHM

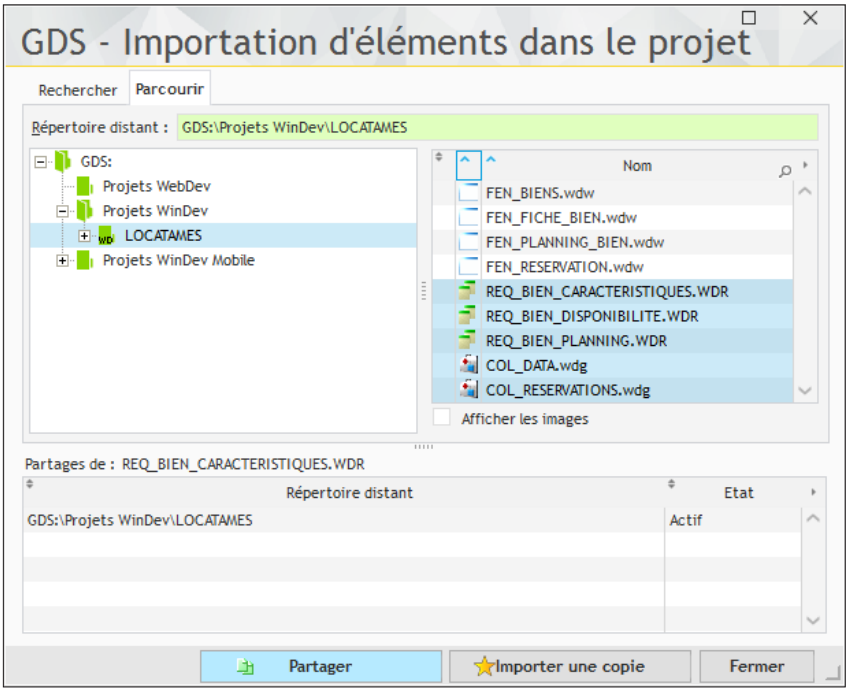
Les IHM mobiles et les IHM Windows ne sont pas tout à fait identiques. Mais ce n'est pas une raison pour jeter votre existant !

Commencez par importer les fenêtres WINDEV : depuis le ruban, sous le volet "Projet", dans le groupe "Projet", déroulez "Importer" et sélectionnez "Des éléments WINDEV et leurs dépendances".

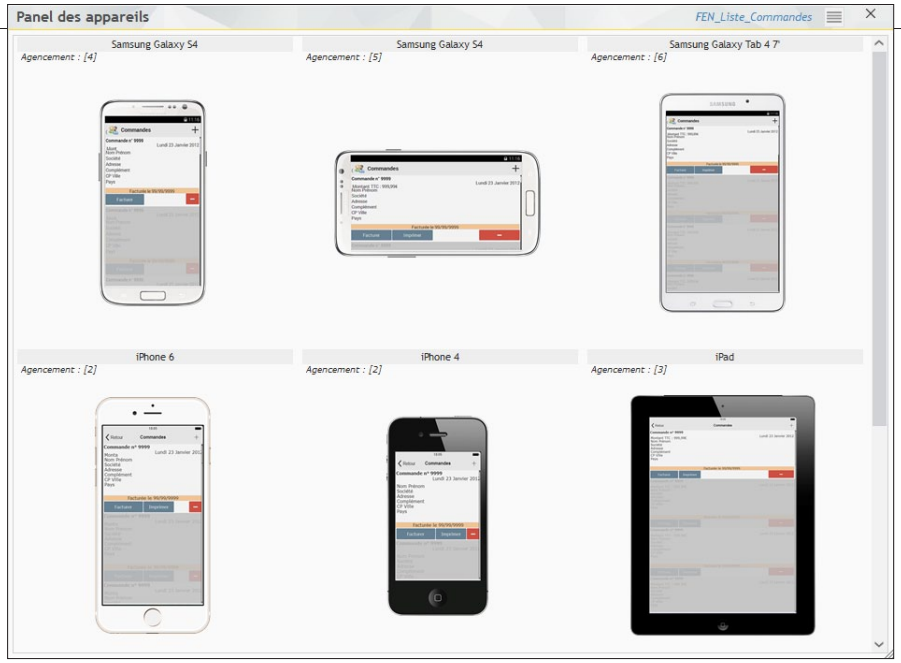
Dans l'assistant qui se lance, sélectionnez le projet WINDEV puis les fenêtres à importer.

WINDEV Mobile importe et convertit les champs WINDEV en WINDEV Mobile.

Tous les traitements des champs et toutes



La fenêtre d'importation du GDS vous permet de récupérer des éléments d'un projet donné. Vous avez le choix entre "Partager" l'élément ou en "Importer une copie". Dans la majorité des cas, préférez le partage à la copie : vos modifications seront ainsi "reportées" dans tous les projets partageant l'élément.



La panel de visualisation est un nouveau mode de visualisation qui permet de vérifier immédiatement que les interfaces sont correctes pour les cibles visées :
• ancrages correctement définis,
• gestion des modes portrait/ paysage,
• fonctionnement sur des écrans de définition différente,
• vérification des agencements (dissociation correcte des tailles, positions et ancrages),
• compatibilité des polices,
• ...

La visualisation est effectuée en temps réel.

Le clic sur un matériel lance la fenêtre de "test" (GO de la fenêtre) afin de pouvoir tester le comportement et de pouvoir comparer entre plusieurs matériels ou plusieurs systèmes.

les procédures locales des fenêtres sont bien entendu conservés.

Si vous utilisiez des champs en WINDEV qui ne sont pas disponibles pour les plateformes mobiles, une erreur d'IHM vous le signale.
Il vous suffit de supprimer le champ et d'adapter la fenêtre.

Bien entendu, la taille des fenêtres devra être adaptée à la plateforme mobile. Et de façon générale, une fenêtre Windows correspondra plus à un affichage en mode paysage sous mobile.

Si vous aviez défini des ancrages en WINDEV, ceux-ci s'appliquent automatiquement au redimensionnement : votre IHM reste correcte quelle que soit sa taille.

L'utilisation des ancrages en mobile présente un énorme avantage : cela permet de gérer les petites différences de résolution à l'exécution entre les différents terminaux.

5 Adapter l'IHM (agencements)

Selon le contenu des fenêtres, les ancrages sur les champs peuvent suffire pour que l'affichage en mode portrait et paysage soit correct.

Mais dans certains cas, il est nécessaire de proposer une organisation des champs différente entre ces 2 modes.

Pour réaliser ce découpage d'affichage, WINDEV Mobile propose une fonctionnalité unique : les agencements.

Les agencements permettent de définir plusieurs dispositions des champs dans une même fenêtre en fonction :
• de la taille,
• de l'orientation,
• de la plateforme d'exécution.

Les agencements sont indispensables pour avoir des IHM spécifiques et soignées !

Créer un agencement

Dans le ruban, sous le volet "Fenêtre", dans le groupe "Agencements", déroulez "Agencements" et sélectionnez "Ajouter des agencements".
Un assistant se lance : il suffit de répondre à quelques questions et WINDEV Mobile détermine automatiquement si des agencements sont nécessaires et lesquels créer.

Ensuite, l'assistant demande si l'application doit s'exécuter sur n'importe quelle taille d'écran ou sur un matériel spécifique.
Le premier choix est le plus souple et généralement le bon.
Le second est utile dans les cas où vous développez pour une flotte homogène d'appareils (généralement dans le cas de développements internes ou spécifiques).
Dans ce cas, les agencements auront automatiquement la taille du matériel ciblé.
Il est ainsi possible de réaliser des IHM très précises sans se préoccuper des ancrages puisque les écrans des appareils mobiles ne changent pas de taille !

Éditer les agencements

Éditer un agencement, c'est exactement comme éditer une fenêtre : vous déplacez les champs, vous en ajoutez...

Mais toute modification faite dans un agencement est automatiquement répercutée dans tous les autres agencements.

Pour qu'un même champ ait un affichage différent entre deux agencements, il est possible de dissocier certaines propriétés : position, taille, ancrage, police et visibilité.

Pour dissocier une propriété d'un champ dans un agencement, il suffit de dérouler l'option "Agencements" du menu contextuel du champ et de cliquer sur l'option voulue.

N'hésitez pas à consulter la documentation pour plus de détails sur les agencements : "http://doc.pcsoft.fr/?9000153".

Le panel de visualisation

Les périphériques mobiles du marché sont très hétérogènes. Avec WINDEV Mobile, les agencements et les ancrages permettent de créer des IHM qui s'adaptent au mieux sur chaque périphérique.

Grâce au panel de visualisation, vous pouvez même tester votre application sur n'importe quel appareil mobile : depuis le ruban, volet "Fenêtre", groupe "Aperçu", cliquez sur "Panel des appareils" (ou utilisez le raccourci "Alt + P").

Par défaut, le panel de visualisation affiche un aperçu pour chaque plateforme déclarée dans le projet.
Si vous souhaitez voir le rendu d'une fenêtre sur un appareil spécifique, ouvrez le menu (bouton en haut à droite) et cliquez sur "Ajouter un appareil".

6 Traiter les différences d'ouverture

La principale différence entre la programmation Windows et la programmation mobile est la gestion de l'ouverture des fenêtres :
• en Windows, les ouvertures de fenêtres sont modales et la fonction WLangage ***Ouvre*** bloque le code appelant. L'ouverture de fenêtre est synchrone.
• en mobile, et quelle que soit la plateforme, les ouvertures de fenêtres ne bloquent pas le code appelant. Elles sont dites asynchrones.

Pour bien marquer la différence, la fonc-

DE WINDEV À WINDEV MOBILE (SUITE)

tion WLangage à utiliser pour ouvrir une fenêtre en mobile est différente : il s'agit de ***OuvreFenêtreMobile***.

Lors du passage au mobile, il est donc nécessaire de revoir les appels à la fonction ***Ouvre*** (qui sont signalés par des erreurs) et de traiter différemment les valeurs de retours.

Voyons le cas sur ce code :

```
-- Code de clic du bouton
"Nouveau produit"
// Ouvre la fenêtre produit en
// création
// Cette fenêtre renvoie la référence
// du produit créé
sRéférence est une chaîne = ...
  Ouvre(FEN_Produit)
```

1. Commencez par remplacer la fonction ***Ouvre*** par la fonction ***OuvreFenêtreMobile***.

2. Coupez le code qui récupère et traite la valeur de retour.

3. Copiez ce code dans le traitement "Fermeture d'une fenêtre fille" de la fenêtre. Ce traitement est appelé dès qu'une fenêtre ouverte par ***OuvreFenêtreMobile*** est refermée. Pour récupérer la valeur renvoyée, utilisez l'expression ***MaFenêtreFille..ValeurRenvoyée***.

```
-- Code de clic du bouton
"Nouveau produit"
OuvreFenêtreMobile(FEN_Produit)

-- Code de fermeture d'une
fenêtre fille de la fenêtre
sRéférence est une chaîne = ...
  MaFenêtreFille..ValeurRenvoyée
```

Remarque : si la fenêtre courante ouvre plusieurs fenêtres filles, utilisez ***MaFenêtreFille..Nom*** pour identifier la fenêtre fille qui vient d'être fermée.

Une première version de votre application mobile est maintenant disponible.
Mais avec WINDEV Mobile, vous pouvez aller beaucoup plus loin, pour enrichir l'application et la rendre plus agréable.

Une fonction n'est pas disponible pour la configuration. Comment adapter mon code ?

Outre les erreurs d'ouverture, il est possible que certaines fonctions, propriétés, types, ... génèrent une erreur de compilation "non disponible dans le framework <Android/iOS>" :
• soit parce qu'elles n'ont pas de raison d'être (les fonctions ***RegistreXXX*** par exemple),
• soit parce que les champs associés ne sont pas disponibles (fonctions ***AgendaXXX*** par exemple),
• soit parce qu'elles seront disponibles ultérieurement.
Il faut prendre chaque erreur au cas par cas pour adapter au mieux. En utilisant la compilation conditionnelle (nouvelle syntaxe ***COMPILE SI***), vous pouvez adapter au mieux un code partagé entre différentes configurations (Windows, Android, iOS, ...).

```
<COMPILE TypeConfiguration =
                          Android>
  // Code à exécuter sous Android
  ...
<SINON>
  // Code à exécuter pour les
  // autres types de configuration
  ...
<FIN>
```

Pour des besoins particuliers ou une fonctionnalité qui ne serait pas encore disponible en WLangage, vous pouvez intégrer du code Java et/ou Objective-C en créant des procédures spécifiques.

7 Enrichir l'application mobile

1. Gestures

Les gestures sont les mouvements de doigts qui sont utilisés en permanence sur les mobiles (pour valider, rafraîchir ou effacer des éléments). Dans vos applications, vous pouvez en proposer à vos utilisateurs.

Il existe des gestures automatiques (comme le scroll et le zoom du champ Image par exemple) qui sont activables par une simple option.

Mais vous pouvez tout à fait gérer n'importe quel type de gesture avec la famille de fonctions WLangage ***GesteXXX***.

2. Sliding menu

Les sliding menus sont des fenêtres coulissantes qui apparaissent par la gauche ou par la droite de l'écran par gesture de l'utilisateur ou appel de la fonction WLangage ***FenCoulissanteVisible***.

Pour définir un sliding menu avec WINDEV Mobile, c'est très facile. Il suffit de :
• créer une fenêtre interne correspondant au menu,
• sélectionner cette fenêtre comme menu depuis la fenêtre de description de la fenêtre.

Il est également possible de modifier dynamiquement le sliding menu par programmation avec les propriétés ***..MaFenêtreCoulissanteGauche*** et ***..MaFenêtreCoulissanteDroite***.

3. Pull to refresh

Le pull to refresh est l'action de faire glisser le contenu d'une table ou d'une zone répétée vers le bas pour provoquer son rafraîchissement. C'est un fonctionnement standard des IHM mobiles.
Pour activer cette fonctionnalité, il suffit de cocher l'option "Rafraîchissement par tirer/relâcher (Pull to refresh)" dans la description du champ, onglet "Détail".

4. Swipe

En version 21, vous pouvez définir comment se comporte le swipe (balayage) sur une zone répétée grâce à une toute nouvelle fenêtre de paramétrage.
Il est par exemple possible de déclarer deux fenêtres internes qui seront utilisées automatiquement pour le swipe à gauche et le swipe à droite.

5. Fonctionnalités spécifiques mobiles

Au lieu d'un champ de recherche texte pour trouver un produit, mettez une recherche de code-barres avec scan intégré dans la fenêtre. C'est une nouveauté de la version 21.

Pour savoir comment mettre en place ces fonctionnalités dans vos fonctionnalités, n'hésitez pas à consulter l'aide en ligne.

■

DE WINDEV À WEBDEV

Passer une application Windows en une application Web, c'est également possible !
Les étapes sont sensiblement les mêmes que pour le passage en mobile.
Suivez le guide (à nouveau) !

1 Préparer le projet WINDEV

Première étape pour une transition sereine vers le Web : tout comme en mobile **séparez tout le code métier de l'IHM** dans votre application (voir les recommandations à ce sujet page 62).

2 Créer le projet WEBDEV

Sous WEBDEV, créez un nouveau projet et suivez les étapes de l'assistant.

Pour convertir une application WINDEV en site Web, le plus simple est de commencer par faire un site dynamique classique.
Ensuite, si besoin, vous pourrez le faire évoluer vers le mode AWP plus facilement.

Si votre application WINDEV utilise une base de données, demandez à ce que l'application WEBDEV utilise la même.

Si votre application WINDEV est dans un GDS (et nous vous le conseillons fortement), ajoutez le projet WEBDEV dans le même GDS. Lorsque l'assistant d'ajout au GDS vous le demandera, indiquez que vous souhaitez partager l'analyse avec l'autre projet.

3 Récupérer le code métier

Comme pour le passage en mobile, récupérez tout le code métier de l'application en partageant les classes et les collections de procédures depuis le projet source WINDEV.

Depuis le ruban, sous le volet "GDS", dans le groupe "Projet", cliquez sur "Importer depuis le GDS". Une fenêtre de sélection s'affiche.

Positionnez-vous sur le répertoire du projet WINDEV, sélectionnez les éléments à partager et cliquez sur "Partager".

4 Récupérer les IHM

Commencez par importer les fenêtres WINDEV : depuis le ruban, sous le volet "Projet", dans le groupe "Projet", déroulez "Importer" et sélectionnez "Des éléments WINDEV ou WINDEV Mobile...".

Dans l'assistant qui se lance, sélectionnez le projet WINDEV puis les fenêtres à importer.

WEBDEV importe et convertit les champs WINDEV en WEBDEV.

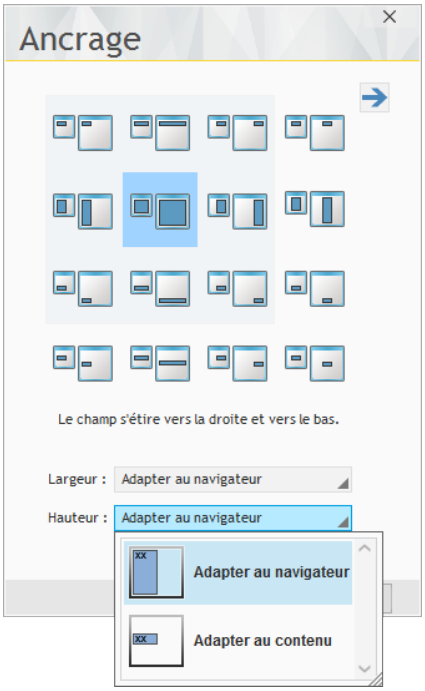
Tous les traitements des champs et toutes les procédures locales des fenêtres sont bien entendu conservés.

Pour créer l'IHM des pages Web, comme en mobile, tout est quasiment automatique.

Notre astuce
Par défaut, à la fin de l'import d'une fenêtre WINDEV, tous les champs sont marqués comme "superposables" (option "Le champ peut être superposé", accessible depuis l'onglet "IHM" de la description du champ). Pour vous faciliter la mise en page, commencez par désactiver ce mode (réactivez-le uniquement si besoin, si le champ doit réellement être superposé). Dans la majorité des cas, vos champs n'ont pas besoin d'être superposés.

5 Adapter l'IHM (ancrages)

Les ancrages en WEBDEV sont complètement différents des ancrages en WINDEV. En WEBDEV, un champ peut :
• utiliser la taille définie en édition (pas d'ancrage),
• adapter sa hauteur/largeur au navigateur,
• adapter sa hauteur/largeur à son contenu.



Comme les ancrages WEBDEV sont différents des ancrages WINDEV, ceux-ci ne sont pas récupérés automatiquement lors de l'import.

Vous devez donc repasser sur les pages pour définir l'ancrage le plus adapté à chaque champ.

6 Traiter les différences d'ouverture

Tout comme lors du passage en mobile, une erreur de compilation indique que la fonction ***Ouvre*** n'est pas disponible en WEBDEV.

Pour ouvrir une page en WEBDEV, vous pouvez utiliser les fonctions ***PageAffiche*** ou ***PageAfficheDialogue***.

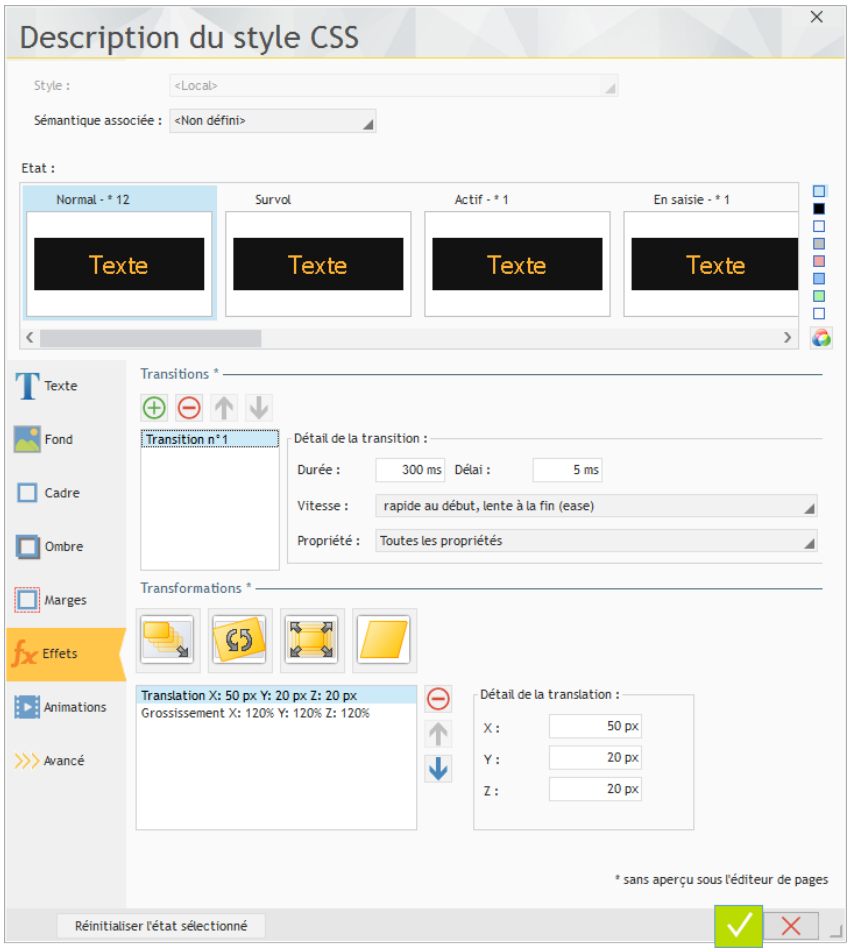
DE WINDEV À WEBDEV (SUITE)

Pour respecter le comportement initial de l'application WINDEV, il convient ici d'utiliser la fonction ***PageAffficheDialogue***: de cette façon, le navigateur va afficher la page par-dessus la page en cours avec un effet de GFI automatique.

Rappel: le GFI (Grisage des Fenêtres Inaccessibles) applique un effet grisé sur les champs de la page appelante (les champs seront visibles en arrière-plan, mais grisés).
Pour renvoyer un résultat depuis une page ouverte par ***PageAffficheDialogue***, il suffit de remplacer l'appel initial à la fonction WLangage ***Ferme*** par ***FermePageDialogue***.

Remarque: la fonction ***PageAffficheDialogue*** n'est pas la seule fonction permettant d'ouvrir une page à partir d'une autre page.
Pour plus d'informations, consultez le tableau de la page suivante.

Et voilà, votre site Web est opérationnel!
Pour aller encore plus loin, il existe des astuces pour rendre votre application Web encore plus riche.



La fenêtre d'édition du style CSS d'un champ permet de personnaliser tous les éléments du champ (texte, fond, cadre, marges, effets, animation, ...) pour chaque état du champ (normal, survol, actif, ...)

7 Enrichir l'application Web

1. Responsive Web Design (RWD)

Par défaut à l'import, les pages créées sont en mode d'édition "zoning".

Mais pour gérer au mieux les différentes tailles de navigateurs, vous pouvez tout à fait passer en mode d'édition Responsive Web Design (ou RWD) apparu en version 21.

Pour plus de détails sur ce mode d'édition et son utilisation, reportez-vous à l'article pages 52-56.

2. Code navigateur

Pour gagner en dynamisme, certains traitements peuvent être déplacés en code navigateur.

De cette façon, tous les traitements qui ne manipulent pas de données (comme la validation des saisies ou les modifications d'affichage) sont réalisés sur le poste client.

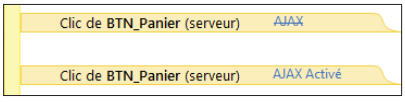
Vous économisez ainsi des allers-retours au serveur: votre site est plus rapide.

3. Ajax

AJAX est la technique qui permet d'appeler des procédures serveur directement depuis du code navigateur.

Pour exécuter du code en AJAX avec WEBDEV, c'est très simple:

• vous pouvez activer ce mode depuis l'éditeur de code sur le traitement "Clic" serveur des boutons et liens. Il suffit de cliquer sur le "AJAX" barré (qui passe alors en "AJAX activé).



• vous pouvez appeler une procédure serveur depuis un code navigateur via la fonction WLangage ***AjaxExécute***. Pensez bien à indiquer que la procédure peut être utilisée en AJAX (bouton "AJAX" allumé en vert dans le bandeau de code de la procédure).

```
// Exécute la procédure ValideAjout
// en AJAX.
// Cette procédure ajoute un
```

Navigateurs

Les pages générées par WEBDEV sont automatiquement compatibles avec tous les navigateurs : Chrome, Firefox... Et même des navigateurs très récents comme Vivaldi (successeur d'Opera) ou Edge (de Microsoft) !

Attention : si vous utilisez encore Internet Explorer, seule la version 11 est encore prise en charge par Microsoft. Le support de toutes les autres versions a pris fin le 12 janvier 2016 ! N'oubliez pas de migrer et de faire migrer vos clients et vos postes utilisateurs.

```
// enregistrement en base (il s'agit
// d'une procédure serveur)
AJAXExécute(ValideAjout, SAI_Titre)
```

Notre astuce

Si la procédure exécutée en Ajax modifie le contenu d'un champ (par exemple, via un ***TableAffiche***), il est possible de demander à WEBDEV de reporter automatiquement les modifications faites sur les champs dans le code serveur : utilisez simplement le paramètre ***ajaxActualiseChamps*** de la fonction ***AJAXExécute***.

4. Animations automatiques

WEBDEV offre de nombreuses transitions et transformations CSS sur les champs.
Ces animations permettent de dynamiser vos IHM et de les rendre plus agréables à utiliser.

Pour appliquer une animation sur un champ, depuis la description du champ, onglet "Style", sélectionnez l'élément "Effets" : vous pouvez alors ajouter différents effets, paramétrables (durée, sens, ...).

Il est également possible de personnaliser encore plus finement les effets et transitions, selon l'état du champ (actif, survol, etc.) : toujours depuis l'onglet "Style", positionnez-vous sur l'élément "Libellé (CSS)" et cliquez sur le bouton "Plus d'options".

■

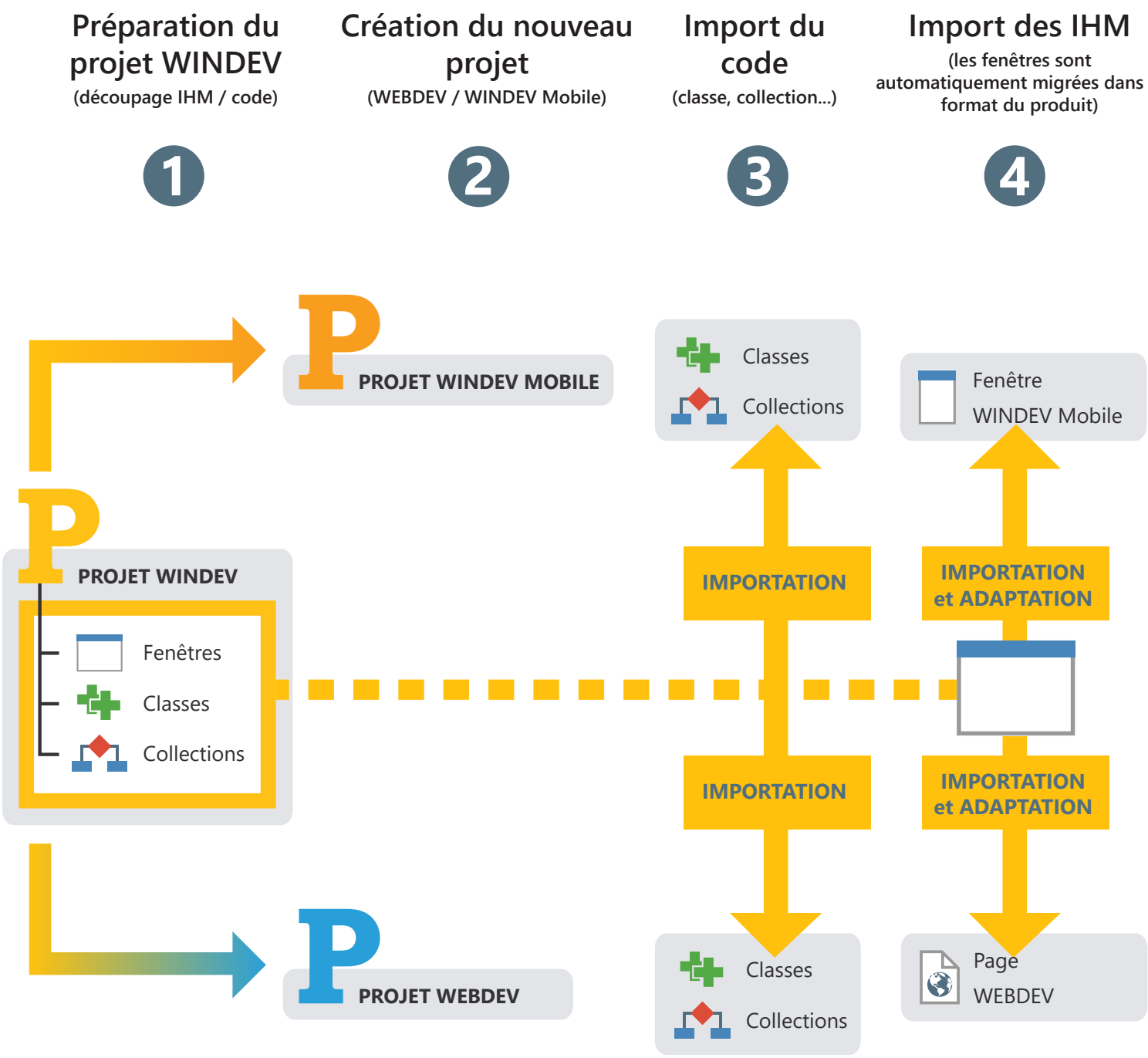
Communication entre deux pages dans un site Web : comparatif

Lors du passage d'une application WINDEV vers WEBDEV, nous avons remplacé les appels à la fonction WLangage ***Ouvre*** par la fonction WLangage ***PageAfficheDialogue***. Cette fonction permet de conserver la logique d'ouverture d'une application WINDEV.
Mais il existe différentes fonctions pour ouvrir et faire "communiquer" des pages WEBDEV.
Voici un tableau récapitulatif des différentes méthodes avec leurs avantages et inconvénients. Pour plus de renseignements sur ces méthodes, consultez les exemples et documents indiqués dans la colonne "Ressources".

	Avantages	Inconvénients	Ressources
Gestion d'une modalité des pages grâce à la fonction ***PageAffiche-Dialogue***	• Simplicité de mise en œuvre lors d'un existant WINDEV • Fonctionnement assez similaire à celui d'une application Windows	• Non référençable (impossible de passer en mode AWP) • Interface lourde • Consommation mémoire	• Exemple "WW_Assistant" (livré en standard) • Aide en ligne • Exemple "eCommerce" (livré en standard) dans la partie "Administration"
Gestion d'une modalité partielle grâce aux Popup et aux cellules (***CelluleAfficheDialogue***, ***PopupAffiche***)	• Rapide pour des modalités simples à 1 ou 2 niveaux (confirmations, saisies)	• Nécessité de découper l'enchaînement des fenêtres en pages + Popup • Partage des popups uniquement par le biais de modèles de pages • Gestion du code serveur pouvant être assez complexe	• LST 89 • LST 91 • Aide en ligne • Exemple "Honolulu" (livré en standard)
Suppression de la modalité et passage en mode "Web" par la fonction ***PageAffiche***	• Logique Web (prise en main plus rapide) • Possibilité d'évolution future vers une page référençable AWP • Consommation mémoire limitée	• Changement d'architecture nécessitant un temps important de développement	• Guide d'autoformation WEBDEV • Aide en ligne

SCHEMA : DE WINDEV À WINDEV MOBILE OU WEBDEV

Le passage d'une application WINDEV en WINDEV Mobile et WEBDEV suit les mêmes étapes. Voici un schéma récapitulatif des différentes étapes présentées précédemment.



Revue des IHM

(correction des ancrages, ajout d'agencements...)

5

Ancrages

Agencements

Gestion des différences de code

(ouverture des fenêtres / pages...)

6

OuvreFenêtreMobile

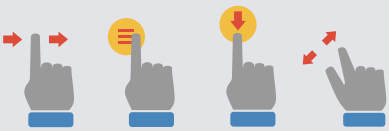
Enrichissement

(ajout de fonctionnalités spécifiques au produit)

7

Gesture, Sliding Menu

Pull To Refresh, Swipe



Ancrages

PageAfficheDialogue

Passage en RWD, Utilisation code navigateur

Ajax, Animation

DE WINDEV À WEBDEV OU WINDEV MOBILE : LES POINTS À CONTRÔLER

Si le passage d'une application WINDEV à une application WEBDEV ou WINDEV Mobile est rapide, il existe cependant différents points qui nécessitent des adaptations. En effet, certaines fonctionnalités diffèrent ou ne sont pas disponibles selon la plateforme.
Voici une liste (non exhaustive) des principaux points à contrôler.

Fonctionnalité	WEBDEV	WINDEV Mobile
Sélection de fichiers (fonction ***fSélecteur***)	Pour manipuler un fichier du poste utilisateur, il est nécessaire de passer par un **champ Upload**. En effet, le fichier ne peut pas être manipulé sans avoir été préalablement récupéré sur le serveur. Pour utiliser un champ Upload, glissez-le simplement dans la page depuis le ruban et renseignez le traitement "Réception des fichiers uploadés".	Le parcours des fichiers du mobile n'est pas possible pour une application Android / iOS. La fonction WLangage ***AlbumSélecteur*** permet cependant de sélectionner un fichier de type image ou vidéo directement dans l'album de l'appareil mobile.
Impression d'états	Si l'impression doit être réalisée sur une **imprimante connectée au serveur Web** (cas d'un Intranet par exemple), une configuration spéciale du serveur est nécessaire (voir la page d'aide à ce sujet : "http://doc.pcsoft.fr/?1011002"). Si l'impression doit être réalisée **pour l'internaute** (impression d'une facture par exemple), il est dans ce cas nécessaire d'imprimer l'état dans un fichier (PDF par exemple) puis d'afficher ou télécharger le fichier sur le poste client.	Sous iOS, l'impression permet uniquement de générer **un fichier PDF**, qui peut ensuite être visualisé avec l'application "viewer" intégrée. Sous Android, seule l'impression par les fonctions du WLangage (***iTrait***, ***iImprimeMot***, etc.) est disponible. La LST 88 propose deux exemples pour : • imprimer directement depuis une application iOS en utilisant "Air Print", • imprimer directement depuis une application Android en utilisant "Google Cloud Print".
Plan de fenêtre	En WEBDEV, la notion de plan n'existe pas. Il est donc nécessaire de repenser l'IHM pour l'adapter au mieux à la logique Web. Vous pouvez par exemple passer par des **pages popup** (une popup par plan). Si l'affichage sous forme de plan ne peut pas être modifié facilement, il existe une astuce pour **simuler des plans** en WEBDEV : il suffit d'utiliser un champ Onglet dont les volets sont invisibles (voir l'article à ce sujet dans la LST 100).	Les plans sont disponibles en WINDEV Mobile et sont donc automatiquement récupérés de l'application WINDEV. Toutefois, pour rester dans une logique mobile, nous vous conseillons de convertir les différents plans en **fenêtres internes**. En utilisant un champ Fenêtre interne, l'utilisateur pourra alors "changer de plan" par une simple gesture.
Menu contextuel	En WEBDEV, la notion de menu contextuel n'existe pas. Il est donc nécessaire d'adapter au mieux l'IHM. Pour remplacer un menu contextuel, vous pouvez facilement passer par une **page popup**.	Les menus contextuels sont disponibles en WINDEV Mobile et sont donc automatiquement récupérés de l'application WINDEV. Cependant, n'hésitez pas à repasser sur les menus contextuels composés de sous-menus. D'une part, la navigation dans des sous-menus est moins pratique en mobile (Android), d'autre part, sous iOS, seules les **options de premier niveau** sont affichées.
Champs	Certains champs WINDEV ne sont pas disponibles dans les autres produits (champ Treemap ou Tableau croisé dynamique par exemple). Dans ce cas, vous pouvez, au choix : • supprimer simplement le champ, • mettre en place une visualisation alternative du champ, soit en codant vous-même en WLangage et en utilisant les champs disponibles, soit en utilisant des ressources externes (JS, CSS, ...).	

LICENCE LST

PROGRAMME EXEMPLE désigne tout logiciel source fourni avec la LST, et en particulier celui qui inclut ce texte.

Le PROGRAMME EXEMPLE est fourni dans un but didactique.

L'utilisation de ce programme s'effectue sous l'entière responsabilité de son utilisateur. La responsabilité de PC SOFT ne pourra en aucun cas être mise en cause si le PROGRAMME EXEMPLE ne fonctionne pas tel que vous l'attendez, ou pour quelque raison que ce soit.

Tout détenteur d'une licence WINDEV 21 et/ou WEBDEV 21 et/ou WINDEV Mobile 21 enregistrée peut utiliser et/ou modifier ce PROGRAMME EXEMPLE en respectant les conditions suivantes :

• Les PROGRAMMES EXEMPLE peuvent être inclus dans des applications sauf mention contraire dans l'article de présentation et/ou le programme fourni.

• Toute mention se rapportant à PC SOFT ou à WINDEV ou à WEBDEV devra être supprimée, afin qu'aucun doute ne puisse subsister dans l'esprit d'un utilisateur final.

• Si les applications sont destinées exclusivement à un usage interne au site physique de la société abonnée à la LST, il n'y a pas de contrainte particulière à l'utilisation du PROGRAMME EXEMPLE dans les applications.

• Si le PROGRAMME EXEMPLE est destiné à être diffusé, à titre gratuit ou payant, par quelque moyen que ce soit, ce PROGRAMME EXEMPLE doit être inclus dans une application dont 90% au moins des fonctionnalités de cette application sont constituées d'éléments autres que des PROGRAMMES EXEMPLE provenant d'une LST.

Il est donc interdit, par exemple, de modifier légèrement un PROGRAMME EXEMPLE et de le diffuser. Il est interdit de copier le contenu de cette LST, en partie ou en totalité, par quelque moyen que ce soit et quel que soit le but. Il est interdit de dupliquer et/ou diffuser et/ou transmettre toute ou partie du CD / DVD.

Une LST est destinée à l'usage unique de la personne qui y est abonnée.

Il est interdit de prêter, louer ou vendre "la LST" (CD et / ou magazine). Il est interdit de diffuser par quelque moyen que ce soit les codes sources accompagnant une LST.

RESTEZ INFORMÉS

Fournissez votre email et recevez des informations régulières en plus de la LST.

La LST fournit tous les trimestres des informations, mises à jour et trucs et astuces sur les différents produits PC SOFT.
Pour obtenir automatiquement des informations entre deux LST, fournissez votre adresse email à PC SOFT. Vous serez ainsi régulièrement prévenu :
• des nouvelles versions disponibles en téléchargement.
• des nouvelles FAQ disponibles sur le site de PC SOFT.
• ...

N'hésitez pas à envoyer un email à PC SOFT (pcsoft@pcsoft.fr) pour indiquer vos coordonnées électroniques. Votre adresse email ne sera utilisée que par PC SOFT.

Le support technique pour ce PROGRAMME EXEMPLE est accessible à travers le service "Assistance Directe" uniquement. Malgré les soins apportés à sa rédaction, ce document n'est pas contractuel. Les copies d'écran sont indicatives. PC SOFT se réserve le droit d'améliorer et de modifier ses produits à tout moment.

ABONNEZ-VOUS !

L16113
Tarif modifiable sans préavis

Bulletin d'abonnement à retourner avec votre règlement à :

PC SOFT
Lettre du Support Technique
BP 44 408
34197 MONTPELLIER Cedex 05
France

Vous pouvez également vous abonner directement sur notre site Internet et profiter d'abonnements supplémentaires. "http://pcsoft.fr/pcsoft/tarifs#LST"

Vos Nom & Prénom : .....
Votre Société : .....
Votre Adresse précise : .....
.....
Code Postal : ..... Ville : .....
Pays : ..... Tel : .....
Adresse Email : .....

Je choisis un abonnement "LST PC SOFT" pour : À partir du numéro : .....

FRANCE Métropolitaine
☐ 1 an - 4 N°+ 4 DVD : 159 euros HT; 190,80 euros TTC
☐ 2 ans - 8 N°+ 8 DVD : 279 euros HT; 334,80 euros TTC

AUTRE (exp. par avion)
☐ 1 an - 4 N°+ 4 DVD : 175 euros HT
☐ 2 ans - 8 N°+ 8 DVD : 299 euros HT

Ci joint mon règlement de : ..... Euros TTC
Note : une facture acquittée est systématiquement adressée.

☐ Je règle par chèque.
J'autorise PC SOFT à débiter sur ma carte VISA/MasterCard la somme de : ..... Euros

☐ Je règle par Carte Bancaire.
Cryptogramme : ☐☐☐
Numéro complet de la carte : ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐
la carte expire : mois ☐☐ année ☐☐☐☐

Signature obligatoire du détenteur de la carte :
Nom du détenteur de la carte :
.....