

Nouveautés NS-DK

Table Of Contents

Nouveautés	1
Nouveautés.....	1
Nouveautés de NS-DK 12.0.....	1
Nouveautés de NS-DK 12.00	1
Nouvelle boîte de modification des propriétés de tous les contrôles de la fenêtre. (Couleurs, police, ancrage...).	1
Sauvegarde centralisée des signets (bookmarks) et indication visuelle des librairies en contenant.....	3
Amélioration de la vitesse d’affichage dans la fenêtre de LOG.	4
Mise en avant des warnings C4700, (variables utilisées sans avoir été initialisées) ...	4
Résumé des warnings de compilation après la génération.	4
Meilleure utilisation du multi-processeur pour la compilation.	5
Meilleure robustesse de NsDebug (plus de plantage sur les pointeurs incorrects)	5
Nouvelles polices plus lisibles dans l’outil	5
Nouvelles icônes s’inspirant du « flat design »	5
Amélioration de la génération C.	5
Nouvel entier INT(0)	6
Support de Windows 11	6
Meilleure cohérence des noms des nouveaux contrôles.	6
Nouveau mot clé #location dans le NCL. Permettant de faciliter les traces applicatives.....	6
Nouveau mot clé #null dans le NCL. Permettant de tester proprement les pointeurs, en évitant les warning	7
Affichage plus rapide de la grille.	7
Les boîtes à onglets ne s'affichent plus en mode "Escalier".....	7
Enrichissement de la boîte About	7
Bases de données (NS-DK 12.00)	8
Support de Windows 11	9
Évolution des librairies (NS-DK 12.00).....	12
Exemples d’utilisation (NS-DK 12.00)	13
Nouveautés de NS-DK 11.0.....	13
Nouveautés de NS-DK 11.00	13
Nouveautés dans l'outil NSDESIGN (NS-DK 11.00).....	14
Bases de données (NS-DK 11.00)	15
Évolution du comportement des applications (NS-DK 11.00)	16
Évolution des librairies (NS-DK 11.00).....	17
Exemples d’utilisation (NS-DK 10.00)	18
Avec la version 11, NS-DK peut être livrée en 32 ou en 64 bits.....	19
Pourquoi passer au 64 bits ?.....	19
Avantages du système 64 bits	20
Historique des versions 64 bits chez Microsoft (x86).....	20
Changement dans les développement des applications pour fonctionner en 32 et 64 bits.	21

Limitations :	21
Nouveautés de NS-DK 10.0	21
Nouveautés de NS-DK 10.00	21
Nouveautés dans l'outil NSDESIGN (NS-DK 10.00)	22
Bases de données (NS-DK 10.00)	22
Évolution du comportement des applications (NS-DK 10.00)	22
Évolution des librairies (NS-DK 10.00)	23
Feuilles de style (NS-DK 10.00)	23
Outil Nat Bot de robotisation (NS-DK 10.00)	26
Exemples d'utilisation (NS-DK 10.00)	26
Nouveautés de NS-DK 8.00	27
Nouveautés de NS-DK 8.00	27
Nouveautés dans l'outil NSDESIGN (NS-DK 8.00)	27
Bases de données (NS-DK 8.00)	29
Évolution des librairies (NS-DK 8.00)	29
Exemples d'utilisation (NS-DK 8.00)	29
Nouveautés de NS-DK 7.00	30
Nouveautés de NS-DK 7.00	30
Nouveautés dans l'outil NSDESIGN (NS-DK 7.00)	30
Bases de données (NS-DK 7.00)	31
Evolution du langage NCL (NS-DK 7.00)	31
Evolution des librairies (NS-DK 7.00)	32
Exemples d'utilisation (NS-DK 7.00)	32
Nouveautés de NS-DK 6.00	33
Nouveautés de NS-DK 6.00	33
Bases de données (NS-DK 6.00)	33
Evolutions du langage NCL (NS-DK 6.00)	33
Agent Nat System pour QuickTest Professional (NS-DK 6.00)	34
Exemples d'utilisation (NS-DK 6.00)	34
Nouveautés de NS-DK 5 SP2	34
Nouveautés de NS-DK 5 SP2	34
Compatibilité avec Windows Vista (NS-DK 5 SP2)	34
Aide en ligne Nat System	35
Evolution de l'interface de NS-Design	39
Sécurité des documents XML (NS-DK 5 SP2)	43
Appel de fonctions utilisateur (NS-DK 5 SP2)	43
Troncature de chaînes (NS-DK 5 SP2)	44
Evolutions du langage NCL	44
Composant NS-PREX (NS-DK 5 SP2)	47
Nouvelles fonctions et instructions	47
Nouveautés de NS-DK 5.00	48
Nouveautés de NS-DK 5.00	48
Environnement technologique	49
Nouvelle interface de NS-Design	50

Nouveautés NS-DK

Suppression des limitations historiques.....	67
Evolution du langage NCL	67
Contrôles graphiques et affichage	72
Les fichiers NSDKCFG.INI et NSDKLOC.INI (NS-DK 5.00).....	77
Génération et déboguage.....	78
Nouvelles fonctions et instructions	79
Index	83

Nouveautés

Nouveautés

[Nouveautés de NS-DK 12.00](#)

[Nouveautés de NS-DK 11.00](#)

[Nouveautés de NS-DK 10.00](#)

[Nouveautés de NS-DK 8.00](#)

[Nouveautés de NS-DK 7.00](#)

[Nouveautés de NS-DK 6.00](#)

[Nouveautés de NS-DK 5 SP2](#)

[Nouveautés de NS-DK 5.00](#)

Nouveautés de NS-DK 12.0

Nouveautés de NS-DK 12.00

NS-DK 12.00 intègre de nouvelles fonctionnalités permettant d'optimiser les applications développées avec cet outil. Nat System répond ainsi aux exigences technologiques et techniques des utilisateurs.

[Nouveautés dans l'outil NSDESIGN](#)

[Bases de données](#)

[Évolution du comportement des applications](#)

[Évolution des librairies](#)

[Exemples d'utilisation](#)

Amélioration du support du 64 bits.

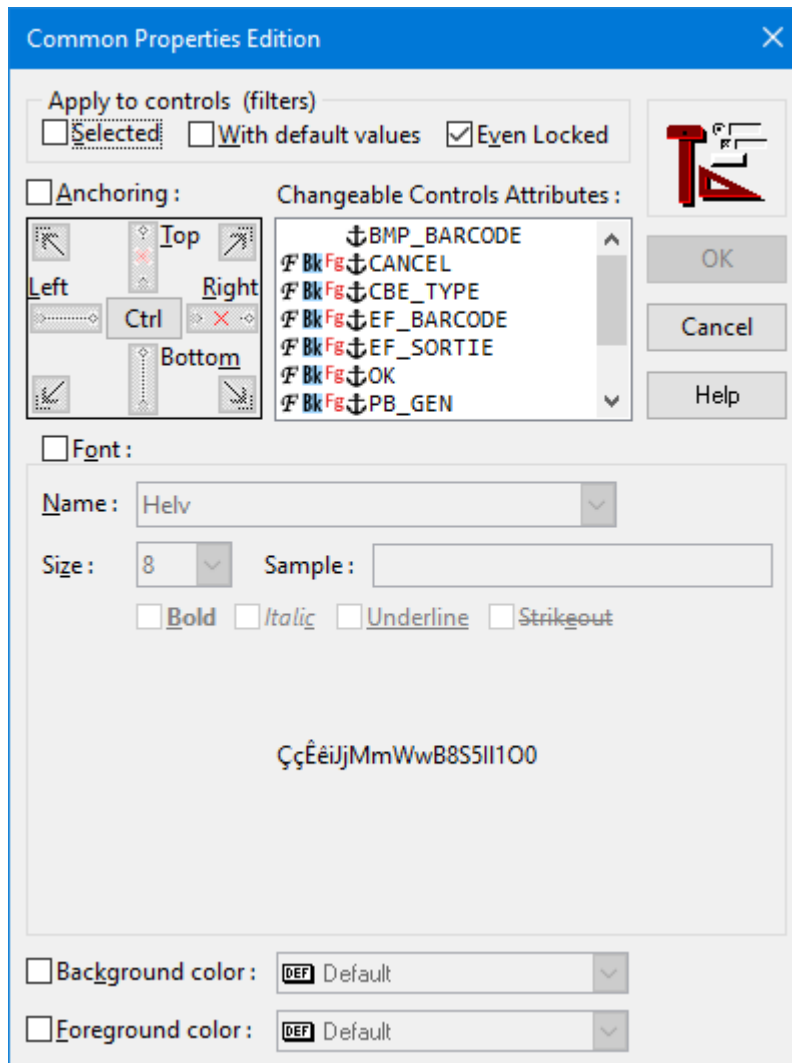
Nouveautés dans l'outil NSDESIGN (**NS-DK 12.00**)

Nouvelle boîte de modification des propriétés de tous les contrôles de la fenêtre. (Couleurs, police, ancrage...).

Quand on clique sur la nouvelle icône "Common Attributes":



Une nouvelle boîte de dialogue apparaît:



Cette boîte de dialogue permet de modifier certaines des propriétés de tous les contrôles de la fenêtre.

- l'Ancrage
- La police de caractère
- la couleur d'avant plan.
- la couleur d'arrière plan.

Utilisation

1. ouvrir une boîte de dialogue dans NS-Design de NS-DK.
2. sélectionner les contrôles que l'on veut affecter (si pas de contrôle sélectionné, tous les contrôles de la fenêtre seront affectés)
3. cliquer sur la nouvelle icône "Common attributes"

Cases à cocher. "Apply to controls (filters)"

- "Selected" pour changer uniquement les contrôles qui sont sélectionnés, sinon tous les contrôles de la fenêtre seront affectés.
- "With Default Value" pour modifier seulement ceux qui ont les valeurs par défaut.
- Si "Even Locked" est coché, même les contrôles verrouillés (par Ctrl-F7) peuvent être modifiés.

Pour l'ancrage :

Cocher la case "Anchoring"

La liste "Changeable Controls Attributes" permet de voir les propriétés et la liste des contrôles modifiables.

Si on veut changer l'ancrage, il faut cliquer sur les coins (flèches obliques) ou les barres verticales et horizontales.

Pour la police de caractères

Cochez la case "Font"

Choisir la police désirée.

Pour les couleurs d'avant plan

Cochez la case ForeGround Color

Choisir la couleur désirée.

Pour les couleurs d'arrière plan

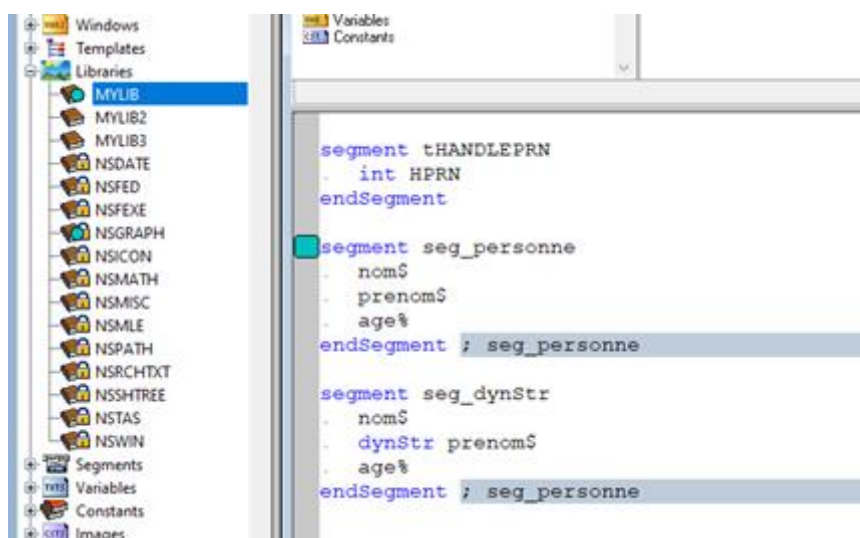
Cochez la case BackGround Color

Choisir la couleur désirée.

Sauvegarde centralisée des signets (bookmarks) et indication visuelle des librairies en contenant

Avec NS-DESIGN, il est possible de placer des bookMarks (ou onglets) dans le code NCL. (Ctrl-F12 dans l'éditeur de librairie NCL, ou par menu contextuel)

Ces bookmarks sont maintenant sauvés et une indication visuelle dans la liste des librairies permet de retrouver facilement les librairies NCL comportant ces bookmark.



Sur cet exemple, on peut voir qu'il y a un signet sur la définition du segment SEG_PERSONNE et que deux librairies NCL (MYLIB et NSGRAPH) ont des bookmarks ou signets.

Cette évolution permet de retrouver facilement l'endroit où l'on travaillé précédemment. Il suffit de double cliquer sur une librairie pour l'ouvrir et accéder directement à l'endroit où est placé le bookMark.

Rappel F12 permet de passer au bookmark suivant.

Amélioration de la vitesse d'affichage dans la fenêtre de LOG.

Jusqu'alors, si on avait activé l'option permettant l'affichage des détails de la génération dans la fenêtre de log, il y avait une désynchronisation entre la fenêtre ligne de commande du compilateur, et la fenêtre de LOG de NS-DK.

Maintenant, la vitesse dans la fenêtre de LOG a été accélérée et des deux affichages sont synchrones). Ce qui diminue notablement le temps de génération d'un projet.

Mise en avant des warnings C4700, (variables utilisées sans avoir été initialisées)

Ces warnings, générés par le compilateur Microsoft, sont importants et doivent être pris en considération. Maintenant, une petite icône rouge apparaît devant la ligne incriminée du fichier.

```

ICompiling wdialog.c...
CL EXE /c /W3 /Gs /Zp /DWIN32 /RTC1 /MDd /I D:\PRJ\PLIBSCPP\TRUNK\TEST\TRUC\GEN /FoD:\PRJ\PLIBSCPP\TRUNK\TES
Compilateur d'optimisation Microsoft (R) C/C++ version 19.34.31937 pour x86
Copyright (C) Microsoft Corporation. Tous droits réservés.

```

```

wdialog.c
% Compilation: D:\PRJ\PLIBSCPP\TRUNK\TEST\TRUC\GEN\wdialog.c(13719) : warning C4700: variable locale 'l' non initialisée utilisée

```

Si on double clique sur cette ligne, on ouvre le fichier C généré. Il est alors facile de retrouver le NCL correspondant

Résumé des warnings de compilation après la génération.

Il peut être intéressant d'avoir d'un coup d'œil un résumé des warning générés par le compilateur C.

```

----- Warning resume -----
Warning C4018 : '>' : incompatibilité signed/unsigned (2)
Warning C4022 : '%' : incompatibilité de pointeur pour le paramètre réel 7 (1)
Warning C4047 : 'fonction' : les niveaux d'indirection de 'NS_PTR *' et de 'NS_LONG *' sont différents (1)
Warning C4101 : '%' : variable locale non référencée (35)
Warning C4244 : '=' : conversion de 'NS_LONG' en 'NS_INT', perte possible de données (5)
% Warning C4700 : variable locale % non initialisée utilisée (1)

```

Voici un exemple de résumé d'un projet comportant de très nombreux warnings. (Dont un warning C4700 et 35 variables non référencées)

Meilleure utilisation du multi-processeur pour la compilation.

Une nouvelle option a été ajoutée au générateur. Elle permet, quand on utilise un compilateur récent (Visual Studio 2010 et supérieur) de lancer la compilation de plusieurs fichiers C en même temps.



La valeur 0 permet de compiler tous les fichiers C générés en même temps. Ce qui réduit le temps de compilation

Meilleure robustesse de NsDebug (plus de plantage sur les pointeurs incorrects)

La nouvelle architecture en C++ de nos produits permet une bien meilleure robustesse.

Dans cette optique il y avait un bug dans NS-DEBUG qui a pu être corrigé :

Si on faisait un watch d'une variable de type pointeur, et que ce pointeur n'était pas encore affecté ou utilisé, on pouvait avoir un plantage, car à l'évaluation le pointeur référençait un emplacement mémoire interdit de lecture.

Maintenant, seul un message apparaît indiquant que le pointeur n'est plus valide :

```
seg_personne(pt).nom$:<Access Violation Exception>
```

Nouvelles polices plus lisibles dans l'outil

Dans l'outil NS-DESIGN de nombreuses fenêtres ont été remaniées afin de les rendre le texte plus lisible. La polie de caractère "Segoe UI,9" est utilisée.

Nouvelles icônes s'inspirant du « flat design »

Les icônes de NS-DK ont été changées pour correspondre davantage à la tendance « Flat Design » actuelle. Avec moins de dégradés

Amélioration de la génération C.

Cette amélioration concerne principalement la gestion des sous/caractères uniques dans une chaîne de caractères.

Lors de l'utilisation de la syntaxe NCL qui permet d'isoler un caractère

```
myString$ (5) = '*'
; ou
ch = myString$(5)
```

Le code C généré est bien plus simple et plus performant.

Nouvel entier INT(0)

Cet entier a la même taille qu'un pointeur. C'est-à-dire qu'il fera 4 octets en 32 bits, et 8 octets en 64 bits.

Cet entier peut être utilisé dans certains cas où une portabilité totale entre les entiers et les pointeurs est requise.

Support de Windows 11

Sur Windows 11, Seule la version de 12 de NS-DK fonctionne correctement. En effet des différences d'affichages ont été constatés avec Windows 11, particulièrement pour les Scroll-bar qui ont complètement changé d'aspect.

Une nouvelle option dans le fichier NSLIB.INI, permet de choisir entre les configurations d'affichage des scroll Bar

```
; Set to false to display old look scrollBar controls.
;UseWindowsScrollBars=true
```

Par défaut, sur Windows, les scroll bar affichées sont celles de Windows. Si, avec Windows 11

Meilleure cohérence des noms des nouveaux contrôles.

Lors de la création d'un nouveau contrôle ou bien de la copie, dans une fenêtre, les noms générés sont maintenant normalisés.

(PB_*, EF_*, ST_* ...=

Nouveau mot clé #location dans le NCL. Permettant de faciliter les traces applicatives.

Si dans une librairie NCL ou depuis le code d'un événement, vous voulez mettre une trace avec le nom de la fonction/instruction ou le nom du contrôlé et de l'événement, le mot clef #location le fait automatiquement.

exemples

```
instruction testTraceFromLib
  local u%
  u% = 10
```

```
NS_TRACE "From my library" && #location && "u=" && u%  
endInstruction ; testTraceFromLib
```

Affichera dans la trace:

NSDK: From my library I.MYLIB.TESTTRACEFROMLIB(145) u= 10

et depuis l'événement EXECUTED du push button PB_TEST_TRACE de la fenêtre WDIALOG

```
; EVENT executed  
ns_trace ("From event :" && #location)
```

Affichera dans la trace:

NSDK: From event : W.WDIALOG.PB_TEST_TRACE.EXECUTED(3)

les N° entre parenthèses correspondent aux N° de ligne dans la librairie et dans l'événement.

Nouveau mot clé #null dans le NCL. Permettant de tester proprement les pointeurs, en évitant les warning

Si vous utilisez des pointeurs, il est plus "propre" de les initialiser à la valeur **#null** plutôt qu'avec 0, cela assure mieux la compatibilité, avec le 64 bits et avec la génération java.

Affichage plus rapide de la grille.

Si vous activez l'option "Display Grid" depuis NS-DESIGN, son affichage a été considérablement accéléré.

Les boîtes à onglets ne s'affichent plus en mode "Escalier"

Cet ancien mode d'affichage a été abandonné pour la version 12.

Enrichissement de la boîte About

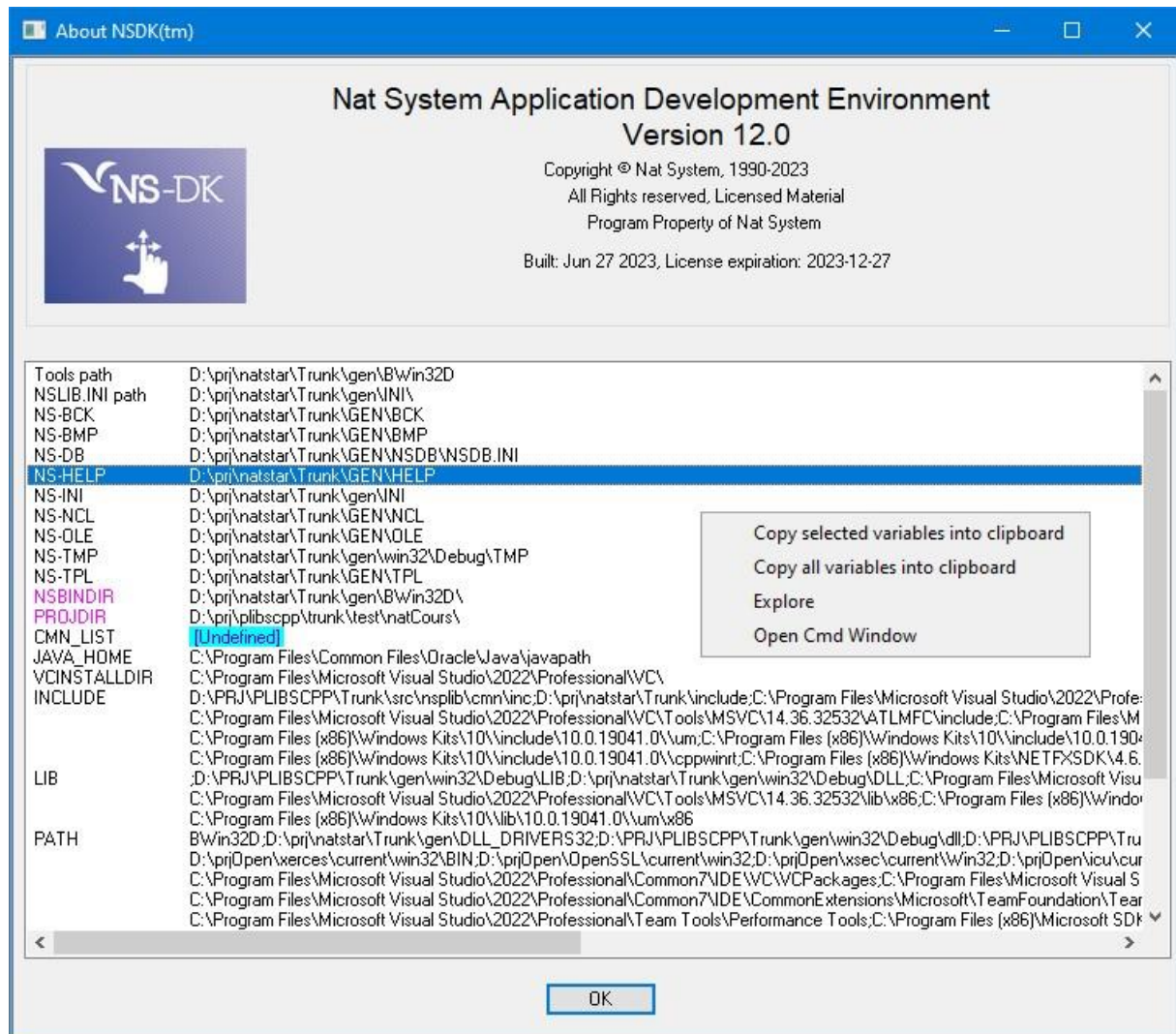
Cette boîte affiche maintenant de nombreuses informations sur les variables d'environnement.

On peut y voir:

- L'emplacement du fichier NSLIB.INI
- Toutes les variables commençant par NS-
- les variables définies dans la boîte "options directories"

- les variables PROJDIR (répertoire du projet) , NSBINDIR (répertoire des binaires de NS-DESIGN), JAVA_HOME, VCINSTALLDIR (emplacement du compilateur), CL, LINK, INCLUDE, LIB, PATH

Exemple



On peut voir que :

- les pseudo variables d'environnement (spécifiques NS-DESIGN) sont en rose.
- La variable CMN_ST définie dans la boîte "Option Directories" n'est pas définie (bleue)
- Le clic droit de souris ouvre un menu contextuel, qui permet de copier les lignes dans le presse papier, d'ouvrir un explorateur ou une ligne de commande dans le répertoire spécifié.

Bases de données (NS-DK 12.00)

NS-DK 12 propose le support des nouvelles versions des bases de données présentées dans le tableau ci-dessous.

Versions des bases de données	Windows 32 bits	Windows 64 bits
DB2 11	NSW2DB211.DLL	NSW4DB211.DLL
Oracle 19c	NSW2OR19.DLL	NSW4OR19.DLL
ODBC 3.51	NSW2ODC3.DLL	NSW4ODC3.DLL
SYBASE 16	NSW2S16.DLL	
MYSQL 8	NSW2MY8.DLL	NSW4MY8.DLL
Informix 11	NSW2INF11.DLL	
PostgreSQL 14	NSW2PG14.DLL	NSW4PG14.DLL
PostgreSQL 15	NSW2PG15.DLL	NSW4PG15.DLL
Maria DB 10	NSW2MARIA10.DLL	NSW4MARIA10.DLL

Évolution du comportement des applications ([NS-DK 12.00](#))

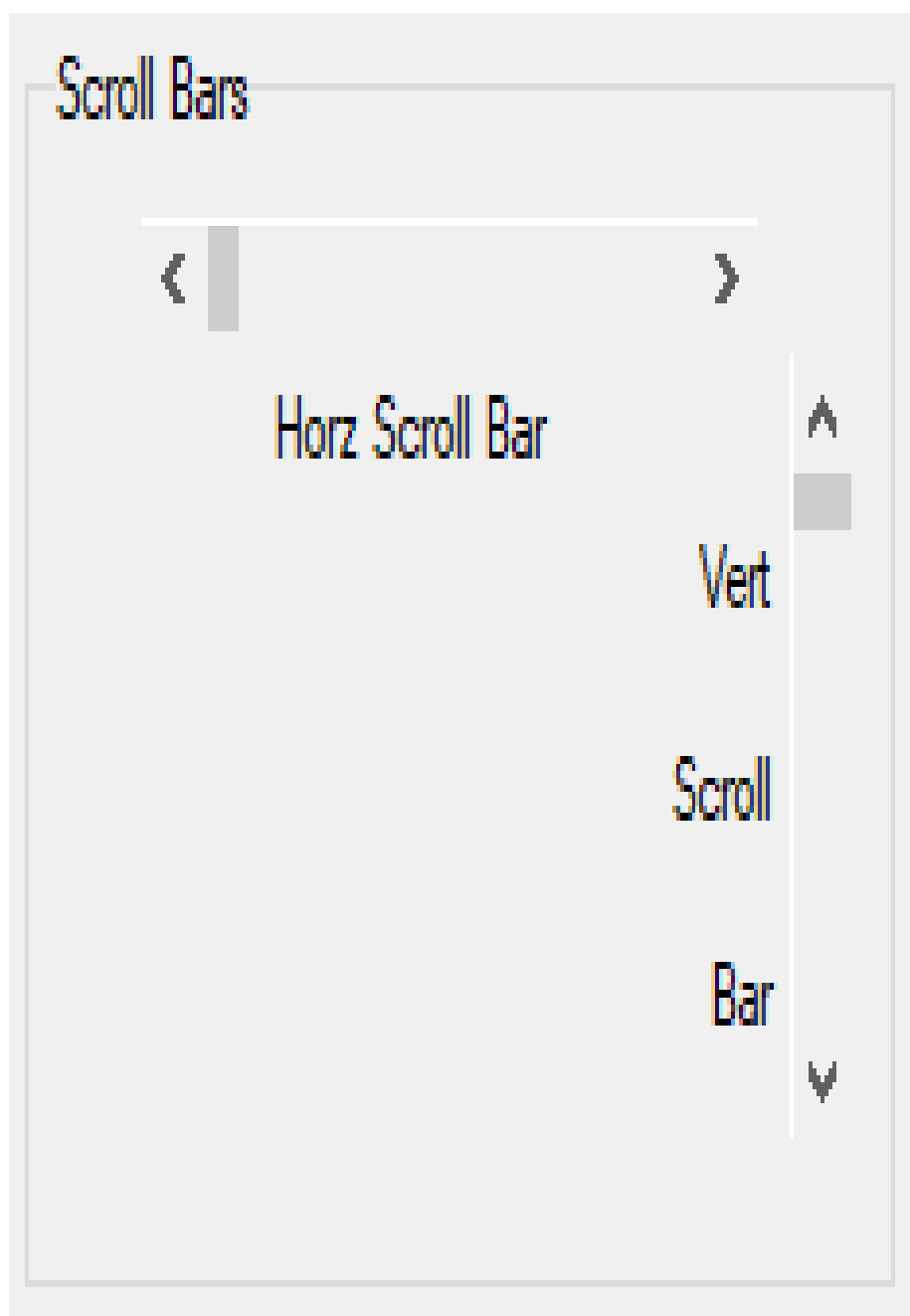
Support de Windows 11

Sur Windows 11, Seule la version de 12 de NS-DK fonctionne correctement. En effet des différences d'affichages ont été constatés avec Windows 11, particulièrement pour les Scroll-bar qui ont complètement changé d'aspect.

Microsoft a introduit une nouveauté , qui fait que les scroll bar ne sont pas vraiment visibles par défaut, elles ne le deviennent que si la souris passe au dessus. Ce qui peut gêner la visibilité et ne pas aider à reconnaître ce type de contrôle du premier coup.

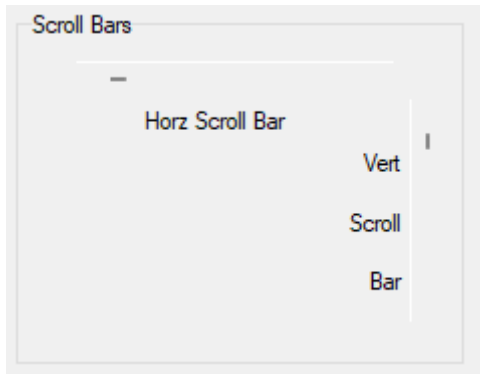
Avec Windows 10

Aspect des contrôles ScrollBar sous Windows 10

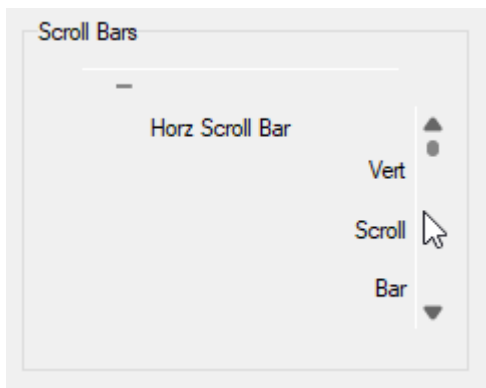
**Avec Windows 11**

Aspect des contrôles ScrollBar sous Windows 11 (Avec NS-DK 12) par défaut
On peut constater que l'aspect des scroll bars est minimaliste,

Nouveautés NS-DK



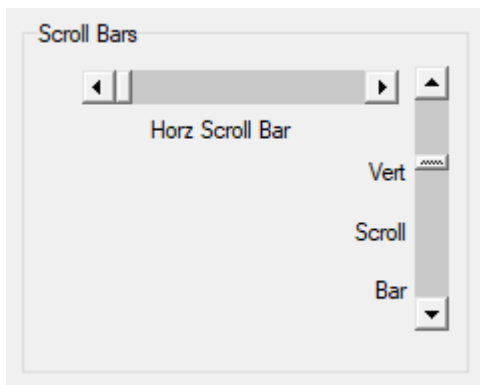
Si on passe la souris sur une scroll bar (ici la verticale), celle-ci apparaît plus clairement.



Pour palier à ce changement et aider l'utilisateur à retrouver un aspect habituel aux scroll bar, Un nouveau paramètre a été rajouté au fichier NSLIB.INI

```
; Set to false to display old look scrollBar controls.  
;UseWindowsScrollBars=True
```

En passant ce paramètre à False, on retrouve un ancien mode d'affichage des scroll bars. Ce qui améliore leur visibilité.



Nouvelle librairie nclLOGR

Cette librairie permet de gérer la génération de fichier LOG, en utilisant toute la puissance de LOG4J/LOG4CXX

Vous pouvez utiliser dans votre code applicatif la nouvelle librairie nclLogr.NCL (remplaçante de NSLOGR.NCL) pour ajouter vos propres logs.

Voir l'aide en ligne de NSLOGR

Amélioration de l'API NS_TRACE.

Avec NS-DK 12.00 on peut maintenant utiliser la bibliothèque log4cxx pour tracer avec NS-TRACE

Recherche du fichier NSLIB.INI

La liste des répertoires où on cherche le fichier NSLIB.INI a été enrichie, maintenant le fichier est cherché dans l'un des répertoires suivant :

1. Répertoire courant. (.)
2. Répertoire du programme
3. (NS-CFG) pour Windows or (NS_CFG) sinon
4. (NS-INI) pour Windows or (NS_INI) sinon, en enlevant le nom de fichier de terminaison si présent.
5. Windows seulement: (LOCALAPPDATA)exeName
6. Windows seulement: (APPDATA)exeName
7. Windows seulement: (ALLUSERSPROFILE)exeName
8. (Répertoire du programme)..\ini
9. Tous les répertoires contenus dans la variable d'environnement PATH)

Évolution des librairies (NS-DK 12.00)

Librairie NSDYNCOLLEC

La nouvelle librairie NSDYNCOLLEC permet de gérer des collections de pointeurs. Cette collection peut croître dynamiquement, et possède de nombreuses fonctionnalités. (tri, libération automatique)

Librairie NSMISC

fonction FGetSizeEx% renvoyant un INT(8) pour lire les tailles de fichier > 2 gigas.

fonction getTmpDir\$, permettant de spécifier un répertoire temporaire spécifique à l'applicatif courant.

fonction cleanTmpDir\$, pour effacer le contenu de ce répertoire temporaire.

fonctions concatPath1\$, concatPath2\$, concatPath3\$, concatPath4\$ pour concaténer simplement des répertoires et des noms de fichiers, sans se soucier des séparateurs intermédiaires.

Librairie NSMAPS

Nouvelle API nsMaps_Map_Clear() pour vider une map sans la détruire.

Librairie NSWIN

Ajout d'une nouvelle fonction nsWriteLnConsole% () pour écrire dans la console depuis une application graphique.. ceci permet de faire, avec NS-DK, un programme purement console, ou bien qui peut-être à la fois graphique et écrire dans la console.

Librairie nclLOGR

Cette librairie permet de gérer la génération de fichier LOG, en utilisant toute la puissance de LOG4J/LOG4CXX

Vous pouvez utiliser dans votre code applicatif la nouvelle librairie nclLogr.NCL (remplaçante de NSLOGR.NCL) pour ajouter vos propres logs.

Librairie nsUnicod

Ajout d'une nouvelle fonction UTF8toAnsiEx\$ permettant de Convertir une chaîne UTF8 avec des caractères décomposés en une chaîne ANSI (précomposée)

Exemples d'utilisation (NS-DK 12.00)

Nouveaux exemples disponibles :

NSDYNCOLLEC : Utilisation de la bibliothèque de gestion de tableaux dynamiques.

NSBARCODE : démonstration de l'appel du composant Zint qui sert à créer tous types de barCode et QR Code

Exemples enrichis

NSMAP : Utilisation de la nouvelle API : NSMAPS_MAP_CLEAR.

Voir le fichier SAMPLES.PDF livré.

Nouveautés de NS-DK 11.0

Nouveautés de NS-DK 11.00

NS-DK 11.00 intègre de nouvelles fonctionnalités permettant d'optimiser les applications développées avec cet outil. Nat System répond ainsi aux exigences technologiques et techniques des utilisateurs.

[Nouveautés dans l'outil NSDESIGN](#)

[Bases de données](#)

[Évolution du comportement des applications](#)

[Évolution des librairies](#)

[Exemples d'utilisation](#)

[Support de la Nouvelle cible Windows 64 bits](#)

Nouveautés dans l'outil NSDESIGN (NS-DK 11.00)

Déclaration automatique des variables.

Quand on écrit du code NCL, il n'est plus nécessaire de remonter en haut de la fonction ou de l'événement pour déclarer une variable locale, il suffit de taper le nom de la nouvelle variable, par exemple MACHIN% et d'enfoncer les touches Shift-Alt-L.

Support de l' INT (8)

Un nouvel entier 64 bit est disponible à l'utilisation, Ces nouveaux entiers INT(8) , permettent de travailler sur de très grands nombres entiers. Leurs valeurs vont de -9223372036854775808 à 9223372036854775807.

Ils permettent de faire des opérations sur des grands nombres sans poser les problèmes d'approximation que l'on peut trouver si on utilise des réels.

On peut par exemple déclarer

```
local i8% (8)
i8% = 200000000000000000L
```

Auto-complétion

L'auto complétion a été grandement améliorée en version 11. notamment les performances et la complétion de qualificatifs dynamiques.

Noter que si l'on passe la souris sur un élément affiché dans la liste de l'auto-complétion, une bulle d'aide descriptive s'affiche.

Accès facilité aux Fonctions/Instructions des librairies

Dans NS-DESIGN, on peut maintenant faire un double-Click sur un nom de fonction/Instruction tout en tenant enfoncée la touche Ctrl.

Ce qui permet d'accéder directement à la fonction. (Action plus simple que Shift-F1 / Shift-F5)

Dans cet exemple touche Ctrl + Double-Click sur CONTROLNAME\$ affiche directement le contenu de la fonction.

Nouveau template Richedit

Ce nouveau contrôle : permet la lecture/écriture des fichiers RTF, affichage et modification du texte formaté, contrôle de l'orthographe (A partir de Windows 10).

Voir le sample Richedit livré avec NS-DK 11

Nouveau

template

NSCHWV

Ce template, basé sur Edge Chromium, en remplacement de NSIEWEB, devenue obsolète, permet d'afficher et de piloter des pages WEB à l'intérieur des applications NS-DK.

Amélioration de la boîte About.

Désormais, cette boîte retailable, affiche le répertoire du binaire ns-Design (tool path), le contenu des variables d'environnement PROJDIR, NS-*, INCLUDE, LIB et PATH

Grep dans les fichiers

Nouveau paramétrage.

Un nouveau flag a été rajouté dans le fichier NSLIB.INI. il permet de choisir si on active ou non par défaut, le mode "expression régulières" dans le GREP

[DevTools]

; set to TRUE to active the Regular expression mode by default in the GREP
;defaultRegExp=False

Copie des Résultats dans le clipBoard.

Il est maintenant possible de sélectionner plusieurs lignes de résultat et de les copier dans le presse-papier, par ctrl-insert ou ctrl-C

Alignement des contrôles.

Amélioration de la boîte d'alignement des contrôles (accessible par F9). il est maintenant possible de préciser les distances entre les contrôles lors d'un alignement.

Nouvelle propriété "Hidden"

Nouvelle propriété "Hidden" qui permet de préciser qu'un contrôle sera caché à l'ouverture de la fenêtre

Support des images au format PNG

NS-DK 11 supporte maintenant les nouveau format PNG.

Souris

Le Clic sur le troisième bouton (milieu) de la souris et maintenant reconnu, ce bouton porte le N° 3. (MOU_TERTIARY%)

Bases de données (NS-DK 11.00)

NS-DK 11 propose le support des nouvelles versions des bases de données présentées dans le tableau ci-dessous.

Versions des bases de données	Windows 32 bits	Windows 64 bits
DB2 11	NSW2DB211.DLL	NSW4DB211.DLL
Oracle 18	NSW2OR18.DLL	NSW4OR18.DLL
ODBC 3.51	NSW2ODC3.DLL	NSW4ODC3.DLL

SYBASE 16	NSW2S16.DLL	
MYSQL 8	NSW2MY8.DLL	NSW4MY8.DLL
Informix 11	NSW2INF11.DLL	
PostgreSQL 10	NSW2PG10.DLL	NSW4PG10.DLL
SqlAnywhere 12	NSW2SA120.DLL	
Maria DB 10	NSW2MARIA10.DLL	NSW4MARIA10.DLL

Évolution du comportement des applications (**NS-DK 11.00**)

Nouveau runtime en C++

Historiquement NS-DK a toujours utilisé les bibliothèques portables pour Windows écrites en Pascal.

Ceci a toujours constitué un frein à son développement. Avec cette version NS-DK fonctionne avec deux nouvelles versions des Bibliothèques portables pour Windows

- Les Bibliothèques Portables C++ 32 bits iso-fonctionnelles des Bibliothèques Pascal.
- Les Bibliothèques Portables C++ 64 bits permettant de produire des Applications 64 bits Windows.

Ce portage améliore la vitesse d'exécution des applications et ouvre la porte à de nombreux développements futurs.

Souris

Le Clic sur le troisième bouton de la souris et maintenant reconnu, ce bouton porte le N° 3. (MOU_TERTIARY%)

Molette

La gestion de la molette de la souris a été optimisée, les scroll dans les listes et les sheet-box sont maintenant bien plus rapides.

Amélioration de l'API NS_TRACE.

Avec NS-DK 11.00 on peut maintenant tracer en temps réel. La trace est directement visible dans la fenêtre du logiciel DebugView et est beaucoup plus rapide que la trace dans un fichier.

Pour cela il suffit de définir la variable NS-TRACE et la fixer à DBGVIEW au lieu d'un fichier texte :

```
SET NS-TRACE=DBGVIEW
```

Idem pour la trace DB-TRACE

```
SET NS-DBTRACE=DBGVIEW
```

Pour télécharger DebugView, sur le site de SysInternals.

<https://docs.microsoft.com/en-us/sysinternals/downloads/debugview>

Voir l'aide en ligne de NS-TRACE

Évolution des librairies (NS-DK 11.00)

Librairie NSWIN.NCL

Amélioration des API de gestion des Multi écran.

En mode multi-écrans, la nouvelle fonction `nsGetMonitorInfosEx%` permet de connaître sur chaque écran, la position de la barre des tâches, la taille de la partie utilisable de l'écran ainsi que le nom du device associé.

Nouvelles API de gestion du registre,

- L'API `BinaryReadRegistryKey%` pour lire les données binaires dans le registre.
- Les APIs `RegistrySubKeysList%` et `RegistryValuesList%` pour lire en une fois les collections de clés ou de valeurs depuis le registre.

Librairie NSSOAP

Les données binaires peuvent désormais être transportées sous leur forme originale, donc sans avoir besoin de les encoder en base64 (ce qui augmente la taille des données à transmettre). Ceci est possible grâce à l'utilisation du mécanisme MTOM (Message Transmission Optimization Mechanism) associé à XOP (XML-binary Optimized Packaging).

Le mécanisme est activable en NCL grâce à l'instruction :

```
NS_SOAPHTTP_SET_MTOMENABLED
```

Librairie NSHTTP et NSSOAP

Des instructions NCL ont été ajoutées pour gérer les cas où les serveurs Web qui gèrent l'authentification ne transmettent les informations nécessaires pour que les applications puissent se connecter.

Ce sont les instructions :

```
NS_SOAPHTTP_SET_USERNAME
```

```
NS_SOAPHTTP_SET_PASSWORD
```

```
NS_SOAPHTTP_SET_AUTHENTICATION
```

Librairie NSJSON

Ajout de la fonction `NSJSON_ISNULL` pour gérer les champs à null. Exemple : `text$ = '{"nom": null}'`

Librairie NSRCHTXT.NCL

Apis permettant de gérer les fichiers texte enrichi au format RTF, de les lire et de les afficher dans le template `RICHEDIT`, ou bien de les sauvegarder depuis le template vers un fichier dans ce format.

voir l'exemple "RICHEDIT" fourni avec NS-DK 11, et qui utilise la plupart des fonctionnalités de cette librairie.

Librairie NSAUTOM

La librairie Automation propose des nouvelles API pour convertir les string BSTR au format COM ou au format UTF16 vers les DynStr Nat System et inversement.

Librairie NSSHTREE

La nouvelle API STV_LINE_FROM_NODE% permet de retrouver le N° de ligne d'une sheetBox à partir d'un nœud

Librairie NSMAPS

Cette librairie permet de gérer une MAP de pointeur ou de String dynamiques

Une map est utilisable comme un tableau, à la différence que l'indexation se fait à l'aide d'une string, et non pas d'un entier.

Elle peut être utilisée par, par exemple, pour mémoriser une liste de personnes, indexée par leur N° INSEE

Librairie et composant NSPREX

Ce composant permet, depuis un navigateur WEB :

- de charger une DLL NS-DK (ou NatStar) et d'appeler plusieurs instructions en leur passant des paramètres.
- de lancer un exécutable NS-DK ou NatStar en lui passant des paramètres.
- d'envoyer un message privé à un programme déjà lancé en lui passant des informations.

La librairie NSPREX.NCL permet aux applications NS-DK de réceptionner des messages envoyés par les navigateurs WEB.

Librairie NSCHW et composant nsChromium

Ce composant, basé sur Edge Chromium vient en remplacement de NSIEWEB, basé sur Internet Explorer, devenue obsolète.

Librairie NSTAS

Ajout d'une nouvelle fonction TAS_DBOX_GETDIRAMEEX pour sélectionner un répertoire avec une nouvelle interface.

Exemples d'utilisation (NS-DK 10.00)

Nat System vous propose de nouveaux exemples d'utilisation (disponibles dans le dossier /samples) vous permettant d'expérimenter les nouvelles fonctionnalités proposées :

- NATBOT : Prise en main complète du robot de test.
- NSJSON : Exemple d'utilisation de la librairie jSon
- NSSTYLES : Exemple d'utilisation de la Feuille de style

Voir le fichier SAMPLES.PDF livré

Nouvelle cible Windows 64 bits



Avec la version 11, NS-DK peut être livrée en 32 ou en 64 bits.

Pour ce faire, l'intégralité des composants des produits Nat System a été portée en 64 bits et les outils (NS-DK, NatStar, NatExtend) peuvent maintenant tourner en mode 64 bits sous Windows.

Ces version 64 bits de NS-DK / NatStar peuvent maintenant générer des applications ou des DLL Windows au format 32 bits ou au format 64 bits.

La cible 64 bits est fournie avec un ensemble de règle de portage des applications.



Renseignez-vous auprès de Nat System pour obtenir les version 64 bits des outils et du run-time NS-DK.

Pourquoi passer au 64 bits ?

- Tous les nouveaux PC sont livrés avec un processeur 64 bits.
- Le 64 bits offre de bien meilleures performances du processeur et de gestion de la mémoire.
- De plus en plus d'applications et librairies externes sont livrées dans cette architecture (Driver de Base de données, Tuxedo...).

- A terme, le 32 bits disparaîtra

Avec cette cible, Nat System assure la viabilité des applicatifs client pour de nombreuses années sur les futures versions de Windows.

La cible 64 bits est l'assurance pour le futur

Avantages du système 64 bits

Mémoire adressable

Les applications 32 bits peuvent adresse 4 gigas de mémoire maximum.

Les applications 64 bits peuvent gérer 16 exa octets (16 milliards de giga octets)

Meilleure performance, intégration à l'architecture 64 bits qui se généralise.

Pérennité des processeurs et applications 64 bits.

Un peu d'assembleur : les processeur passent de 8 registres généraux 32 bits à 16 registres généraux 64 bit, bien meilleur pour l'optimisation

les opération sont plus rapides car basée sur des entiers à 8 octets.

Historique des versions 64 bits chez Microsoft (x86)

Les systèmes suivants existent en version 64 bits :

Windows XP, (2005)
Windows Server 2003, Windows Server 2008,
Windows Vista,
Windows 7
Windows 8.
Windows 10

Ces systèmes 64 bits ne supportent plus les sous système 16 bits

Il s'est écoulé environ 18 ans entre la sortie de Windows 95 (premier 32 bits) et la fin du support 16 bits.

Changement dans les développement des applications pour fonctionner en 32 et 64 bits.

Nat System fournit une charte de développement pour les application client, afin qu'elles soient compatible 32 et 64 bits.

Entre autre, Il ne doit plus avoir de confusion entre les INT(4) et les pointeurs. en 32 bits la taille était la même mais plus en 64 bit.

Limitations :

L'intégralité des modules des produits Nat System a été porté en 64 bits. Seul l'ancien composant NS-DB n'a pas été porté sur cette cible.



Nouveautés de NS-DK 10.0

Nouveautés de NS-DK 10.00

NS-DK 10.00 intègre de nouvelles fonctionnalités permettant d'optimiser les applications développées avec cet outil. Nat System répond ainsi aux exigences technologiques et techniques des utilisateurs.

[Nouveautés dans l'outil NSDESIGN](#)

[Bases de données](#)
[Evolution du comportement des applications](#)
[Evolution des librairies](#)
[Feuilles de Style](#)
[Nouvel outil de robotisation](#)
[Exemples d'utilisation](#)

Nouveautés dans l'outil NSDESIGN (**NS-DK 10.00**)

Fenêtre_Grep

L'option par défaut "No regular Expression est activée"

Embellisseur de code

Une nouvelle action est disponible dans l'éditeur NCL : l'embellisseur de code. Cette action est lancée par le raccourci Ctrl+I.

Son paramétrage d'effectue via la boîte de paramétrage de l'éditeur NCL.

Auto-complétion

L'auto complétion a été grandement améliorée en version 10.

Undo Ctrl-Z

Le Undo de l'éditeur peut se faire maintenant par Ctrl+Z, auparavant, il fallait taper Ctrl+Shift+Z.

Amélioration du contrôle Picture Button

Ce contrôle a été amélioré , il peut être multi-actions avec menu contextuel.

Voir le sample PICTBUT livré avec NS-DK 10

Bases de données (**NS-DK 10.00**)

NS-DK 10 propose le support des nouvelles versions des bases de données présentées dans le tableau ci-dessous.

Versions des bases de données	Drivers
DB2 10	nsw2db210.dll
Oracle 12.1	NSnnOR121.DLL

nn correspond au numéro de version de l'interface que vous avez installé : w2 pour Windows 32 bits,

Évolution du comportement des applications (**NS-DK 10.00**)

- Les Combo Box

Toutes les combo-box sont maintenant real-time scrolling, ce qui facilite leur utilisation.

- Les fenêtres de type MDI

Le menu MDiWindow voit la capacité de sa liste passer de 9 à 18 fenêtres avant de proposer l'item de menu "More" qui affiche la boîte de dialogue contenant toutes les fenêtres ouvertes.

- Les Feuilles de style

Il est maintenant possible de modifier l'aspect des applications sans les regénérer.

Évolution des librairies (NS-DK 10.00)

Librairie NSJSON.NCL

Une nouvelle librairie NS-JSON est livrée. Elle sert à lire les fichiers de type JSON.

Librairie NSWIN.NCL

Nouvelle Api nsMonitorFromWindow pour savoir sur quel écran est affichée une fenêtre NS-DK, dans le mode multi-écrans.

Librairie NSFEXE.NCL

StartExecuteEx supporte maintenant environ 5000 caractères pour le deuxième paramètre qui contient les arguments à passer à l'exécutable.

Librairie NSMISC.NCL

Support du code page 8859-15.

NSDK apporte la possibilité de convertir dans cette page de code. Elle est notamment utilisée pour la transmission de données aux administrations.

Librairie NSSHTREE.NCL

Renvoie le N° de la première ligne visible : FUNCTION STV_GET_TOPLINE% (POINTER SHEET)

Renvoie le N° de la dernière ligne visible : Function STV_GET_BOTTOMLINE% (POINTER SHEET)

Feuilles de style (NS-DK 10.00)

NS-DK 10 est livré avec la possibilité de modifier l'aspect des applications existantes sans les regénérer.

Pour cela, il faut créer ou modifier le fichier nsStyles.xml (le fichier NSLIB.INI peut également être utilisé).

Le fichier nsStyles.xml est placé dans le même répertoire que le fichier NSLIB.INI et se compose comme suit :

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
  <nsstyles>
    <nsstyle name="Default" type="<ctrlType>">
      <property name="foreground" type="Color" force="true"
default="16">White</property>
      <property name="background" type="Color"
default="16">Color1</property>
      <property name="backgroundDisable" type="Color"
default="16">Color1</property>
      <property name="font" type="Font" force="true"
default="Helv;8">Times New Roman,10</property>
    </nsstyle>

    <nsstyle name="Default" type="<ctrlType>">
      ....
      ....
      ....
    </nsstyle>

    <nsstyle name="Default" type="SHEET">
      <property name="FOREGROUND" type="Color" force="false"
default="-1">LIGHTRED</property>
      <property name="TITLE_FORECOLOR" type="Color" force="false"
default="-1">LIGHTGREEN</property>
      <property name="R1C2_FORECOLOR" type="Color" default="-
1">GREEN</property>
      <property name="R1C1_FORECOLOR" type="Color" default="-
1">BLUE</property>
      <property name="R2C2_FORECOLOR" type="Color" default="-
1">WHITE</property>
      <property name="R2C1_FORECOLOR" type="Color" default="-
1">LIGHTRED</property>
      <property name="BACKGROUND" type="Color" force="false"
default="-1">LIGHTBLUE</property>
      <property name="TITLE_BACKCOLOR" type="Color" force="false"
default="-1">YELLOW</property>
      <property name="R1C1_BACKCOLOR" type="Color" default="-
1">GREEN</property>
      <property name="R1C2_BACKCOLOR" type="Color" default="-
1">BLUE</property>
      <property name="R2C1_BACKCOLOR" type="Color" default="-
1">WHITE</property>
      <property name="R2C2_BACKCOLOR" type="Color" default="-
1">LIGHTRED</property>
      <property name="font" type="Font" force="true" default="Helv;8">Times
New Roman,10</property>
      <property name="TITLE_FONT" type="Font" force="true"
default="Helv;8">Times New Roman,10</property>
    </nsstyle>
  </nsstyles>

```

Dans cet exemple les contrôles standards peuvent être de type :

- RadioButton
- CheckBox
- PushButton
- EntryField
- StaticText

- MLE
- GroupBox
- ListBox
- ComboBoxEdit
- ComboBox

On peut préciser le nom d'une couleur ou bien Color1, color2... qui sont les couleurs redéfinies dans la palette (fichier NSLIB.INI).

Deux comportements sont possibles :

- si force = false, on ne change la couleur que si la couleur est la couleur par défaut.
- si force = true on change la couleur dans tous les cas.

Exemples :

Cette ligne permet de modifier la couleur d'avant plan :

```
<property name="foreground" type="Color" force="true"
default="16">Red</property>
```

Ne pas modifier la couleur par défaut (16) : on va la remplacer par le rouge (Red).

Cette ligne permet de modifier la couleur d'arrière plan :

```
<property name="background" type="Color" default="16">White</property>
```

On va la remplacer par le blanc (White).

Cette ligne permet de modifier la couleur d'arrière plan. des contrôles en mode disable :

```
<property name="backgroundDisable" type="Color"
default="16">White</property>
```

On va la remplacer par le blanc (White).

Cette ligne permet de modifier la police de caractères :

```
<property name="font" type="Font" force="true" default="Helv;8">Times New
Roman,10</property>
```

Contrôles tabulaires (Sheet-Box et LDATA)

Avec les contrôles tabulaires, les propriétés sont plus nombreuses et correspondent à celles de la boîte de propriété.

```
<property name="FOREGROUND" type="Color" force="false" default="-
1">LIGHTRED</property>
<property name="TITLE_FORECOLOR" type="Color" force="false" default="-
1">LIGHTGREEN</property>
<property name="R1C2_FORECOLOR" type="Color" default="-1">GREEN</property>
<property name="R1C1_FORECOLOR" type="Color" default="-1">BLUE</property>
<property name="R2C2_FORECOLOR" type="Color" default="-1">WHITE</property>
<property name="R2C1_FORECOLOR" type="Color" default="-
1">LIGHTRED</property>
<property name="BACKGROUND" type="Color" force="false" default="-
1">LIGHTBLUE</property>
<property name="TITLE_BACKCOLOR" type="Color" force="false" default="-
1">YELLOW</property>
<property name="R1C1_BACKCOLOR" type="Color" default="-1">GREEN</property>
<property name="R1C2_BACKCOLOR" type="Color" default="-1">BLUE</property>
<property name="R2C1_BACKCOLOR" type="Color" default="-1">WHITE</property>
```

```
<property name="R2C2_BACKCOLOR" type="Color" default="-
1">LIGHTRED</property>
<property name="font" type="Font" force="true" default="Helv;8">Times New
Roman,10</property>
<property name="TITLE_FONT" type="Font" force="true" default="Helv;8">Times
New Roman,10</property>
```

FICHIER NSLIB.INI

Ce fichier peut également être modifié pour :

- changer la couleur de fond des fenêtres
- changer la palette de couleur.

Il se compose ainsi :

```
; System Color
;
[ColorPalette]
DialogBackground=37,37,37

[DefPalette]
; presque noir.
Color1 = 37,37,37
; jaune pale
Color2 = 255,255,186
; vert
Color3 = 120,186,0
; jaune vif
Color4 = 255,255,0
; rouge sombre
color5 = 122,8,31
```

Il est conseillé de partir des feuilles de style existantes (samples NSSTYLES) et de les modifier.

Outil Nat Bot de robotisation (NS-DK 10.00)

NS-DK 10 est livré avec un nouvel outil de robotisation composé de plusieurs modules et permet de :

- Espionner une application Nat System (NS-DK / NatStar) afin de connaître facilement le nom des fenêtres et des contrôles.
- Enregistrer un scénario de test, simplement en utilisant une application déjà écrite, y incorporer des checkPoint, et le sauver au format XML.
- Transformer ce scénario en script NCL.
- Incorporer ce NCL dans l'outil de développement pour l'adapter et le modifier.
- Rejouer ce script pour piloter et tester l'application.

Voir Nat Bot, présentation générale

Exemples d'utilisation (NS-DK 10.00)

Nat System vous propose de nouveaux exemples d'utilisation (disponibles dans le dossier /samples) vous permettant d'expérimenter les nouvelles fonctionnalités proposées :

- NATBOT : Prise en main complète du robot de test.
- NSJSON : Exemple d'utilisation de la librairie jSon
- NSSTYLES : Exemple d'utilisation de la Feuille de style

Voir le fichier SAMPLES.PDF livré

Nouveautés de NS-DK 8.00

Nouveautés de NS-DK 8.00

NS-DK 8.00 intègre de nouvelles fonctionnalités permettant d'optimiser les applications développées avec cet outil. Nat System répond ainsi aux exigences technologiques et techniques des utilisateurs.

[Nouveautés dans l'outil NSDESIGN](#)

[Bases de données](#)

[Evolution des Librairies](#)

[Exemples d'utilisation](#)

Nouveautés dans l'outil NSDESIGN (NS-DK 8.00)

Grep

Sauvegarde de la liste des chaînes cherchées, ainsi que du dernier résultat de recherche.

Support du mode « Only in comments ».

Fermeture de la fenêtre de grep par appui de la touche Echap.

Démarrage du Grep par appui de la touche Entrée.

Suppression d'une ligne de résultats par appui sur la touche Suppr.

On ne peut plus lancer de recherche si le champ "Find" est vide.

Voir Fenêtre_Grep.

Browser de ressources

La liste des contrôles de chaque fenêtre est maintenant triée par ordre Alphabétique.

Check du NCL

Les messages d'erreur "Invalid Identifier" affichent maintenant quel est l'identificateur invalide.

Création de d'une nouvelle configuration de build

Les répertoires par défaut des configuration de génération sont maintenant (PROJDIR)\GEN, et ce, pour toutes les ressources créées

Gestion des n° de version des excutables et DLL générés

Il est maintenant facile de créer et d'intégrer un fichier de ressources .RC. via la boîte de génération du projet.

Voir le nouveau sample COLORS et la documentation sur la boîte de génération.

Changement des événements par défaut.

Quand on double clique sur un contrôle pour arriver à sa fenêtre d'événements, le premier sélectionné est maintenant pertinent.

Auparavant l'INIT était sélectionné, mais depuis la version 8, ce sont :

- SELECTED pour les Check Box, les radio-boutons, les scroll bar, les combo, comboEdit et les list box.
- CHANGED pour les entry Field.

Ce changement permettra d'éviter de coder l'INIT par inadvertance.

Boîte About

La boîte about affiche maintenant des informations supplémentaires :

- le répertoire où s'exécute le binaire NSDSGN.EXE
- le répertoire où a été lu le fichier NSLIB.INI

Ouverture de projet

Quand on ouvre un projet, la fenêtre des ressources est maintenant affichée.

CONTRÔLES

- SheeBox

Une nouvelle colonne a été ajoutée, elle permet de spécifier le nom d'une colonne.

L'appel de la nouvelle API STV_GET_COLUMN_FROM_NAME% permet de retrouver le numéro de la colonne.

Voir Paramétrage du la Sheet Box et Apis de nom de colonne

Une Nouvelle propriété "Line separator a été ajoutée, elle permet de mettre des lignes horizontales de séparation dans la sheet box.

La boîtes de propriétés détaillées de la Sheet box est retailable, la largeur des colonnes,s'ajustant automatiquement.

- Push Bouton

Le paramètre BackGround permet maintenant de paramétrer la couleur d'arrière plan des push bouton.

- Entry Field et combo Edit

Dans la propriété "Format" a été rajouté le format de date français : dd/mm/yyyy

- Group Box

Une nouvelle propriété "Colored Frame", permet de colorer le cadre du group box suivant la couleur d'avant plan du texte.

Bases de données (NS-DK 8.00)

NS-DK propose le support des nouvelles versions des bases de données présentées dans le tableau ci-dessous.

Une nouvelle base de donnée est supportée, il n'agit de MariaDB, base dérivée de mySql.

Versions des bases de données	Drivers
Oracle 112	NSnnOR112.DLL
PostGres 9.1	NSnnPG91.DLL
MariaDB	NSnnMARIA10.DLL

Pour la documentation de la base de données mariaDB, reportez-vous à la documentation de mySQL.

nn correspond au numéro de version de l'interface que vous avez installé : w2 pour Windows 32 bits

Évolution des librairies (NS-DK 8.00)

Librairie NSFED.NCL

Cette librairie comprend 2 nouvelles fonctions :

- MLE_GETDYNSTR
- MLE_SETDYNSTR

Elles permettent de lire et d'écrire directement une DYNSTR dans une Window de type Edit.Librairie NWXML.NCL

NWX_DOCUMENT_GETFROMDYNSTR

Elle permet de lire un document XML stocké dans une string dynamique.

Librairie NSMISC.NCL

NS_TRACE_DS, nouvelle API de trace utilisant des strings dynamiques.

Librairie NSWIN.NCL

- nsGetMonitorsCount%
- nsGetMonitorInfos%

Ces nouvelles APIS prennent en charge la gestion des écrans multiples

Librairie NSSHTREE.NCL

Gestion des Noms de colonne

Gestion des séparateurs de ligne

Documentation étendue.

Exemples d'utilisation (NS-DK 8.00)

Nat System vous propose de nouveaux exemples d'utilisation (disponibles dans le dossier /samples) vous permettant d'expérimenter les nouvelles fonctionnalités proposées :

- MULTSCREEN: démonstration de la gestion des écrans multiples.
- SHEETBOX : exemple modifié avec démonstration des nouveautés, gestion des Nom de colonnes avec GET_COLUMN_FROM_NAME%
- INIMGR : exemple d'utilisation de la librairie de gestion des fichiers .INI
- COLORS : exemple d'utilisation des paramétrage de couleurs sur la fenêtre et les contrôles. Ce sample offre aussi un exemple de gestion de versionning d'exécutable.

Voir le fichier SAMPLES.PDF livré.

Nouveautés de NS-DK 7.00

Nouveautés de NS-DK 7.00

NS-DK 7.00 intègre de nouvelles fonctionnalités permettant d'optimiser les applications développées avec cet outil. Nat System répond ainsi aux exigences technologiques et techniques des utilisateurs.

[Nouveautés dans l'outil NSDESIGN](#)

[Bases de données](#)

[Evolutions du langage NCL](#)

[Evolution des librairies](#)

[Exemples d'utilisation](#)

Nouveautés dans l'outil NSDESIGN (NS-DK 7.00)

Browser de ressources

L'appui des touches Alt-F5 permet de rafraîchir la liste des ressources. cette option peut être utile si des ressources du projet ont été déplacées alors que le projet est toujours ouvert dans NSDESIGN.

Fenetre de resources

Sélection des contrôles.

Meilleure Sélection / Désélection multiple des contrôles en design.

Voir Sélection_multiple

Grep

Support du mode « exclude comments »

Voir Fenêtre_Grep

Fenêtre de Log

Dans la fenêtre de Log, ajout de la date, de l'heure et du temps de génération.

Voir Fenêtre_de Log

Simplification des créations de ressources

Lors de leur création, les Listbox sont en mode Realtime-Scroll par défaut et les boîtes de dialogue sont en mode " Shell Position" (centrée à l'écran).

Amélioration de la boîte de sélection des polices de caractère

Cette boîte est maintenant plus grande et permet de choisir la chaîne de caractères qui sera affichée dans l'exemple.

Voir Choix de la police_de_caractères

Bitmaps dans les onglets

Il est maintenant possible d'afficher des bitmap dans les onglets.

Voir Insérer des bitmaps dans les onglets

Bases de données (NS-DK 7.00)

NS-DK propose le support des nouvelles versions des bases de données présentées dans le tableau ci-dessous.

Une nouvelle base de donnée est supportée, il s'agit de PostGres, base embarquée.

Versions des bases de données	Drivers
Oracle 112	NSnnOR112.DLL
PostGres 9.1	NSW2PG91.DLL

nn correspond au numéro de version de l'interface que vous avez installé : w2 pour Windows 32 bits, w4 pour Windows 64 bits.

Evolution du langage NCL (NS-DK 7.00)

Mot-clé CONTROL

On peut désormais utiliser le mot clé CONTROL à la place du nom du contrôle pour écrire du code générique : soit le contrôle CHECKBOX CHB_TEST, dans l'événement Selected, lorsqu'on voulait tester l'état coché ou non, on faisait jusqu'à présent :

```
if CHB_TEST = CHECKED%
```

à partir de la version 7, on peut remplacer avantageusement cette ligne par :

```
if CONTROL = CHECKED%
```

Ainsi, on pourra profiter de la généricité du code au cas où on renommerait le contrôle ou bien lors d'un copier-coller.

Cette syntaxe ne fonctionne que sous les événements des contrôles, et est particulièrement utile avec les Check Box et les Radio-boutons.

Evolution des librairies (NS-DK 7.00)

Librairie NSINIM.NCL

Une nouvelle librairie NS-INIM est livrée et sert à lire et écrire les fichiers de type INI.

Elle permet une plus grande portabilité par rapport aux fonctions GetProfileString du Kernel32 de Windows.

Librairie NSSHTREE.NCL

Supporte de nouvelles Apis :

- SET/GET Line/Column/cell BEHAVIOR : individualisation des comportements pour les colonnes, les lignes, cellules. (Edit in place, visible ou non, colonne retaillable ou non, possibilité de n'afficher que la première colonne pour les nœuds ayant des fils)
- STV_NOUPDATE/STVUPDATE APIs pour accélérer l'affichage pour les modifications multiples dans la Sheet-Box
- STV_SETCOLUMRELIEF : modification du relief pour une colonne donnée.
- STV_SETFOCUS : donne le focus à une cellule donnée..

Librairie NSMISC.NCL

Support du code page 8859-15.

NSDK 7.0 apporte la possibilité de convertir dans cette page de code. Elle est notamment utilisée pour la transmission de données aux administrations.

Librairie NSWIN.NCL

Nouvelle Api nsGetTickCount% pour pouvoir gérer les durées au centième de seconde près.

Exemples d'utilisation (NS-DK 7.00)

Nat System vous propose de nouveaux exemples d'utilisation (disponibles dans le dossier /samples) vous permettant d'expérimenter les nouvelles fonctionnalités proposées :

- WEBMETEO : démonstration du client des Services Web, consultation de la météo aéronautique
- DRAGDROP : Démonstration de Drag And Drop depuis Windows vers une application NSDK.
- CLASSIF – Exemple de gestion d'onglets dynamiques avec des images dans les onglets.
- ADDRBOOK_XML : Exemple moderne de gestion de fiches : Look windows moderne, ancrage, sauvegarde des fiches au format XML, Sheet-Box et onglets.

- SHEETBOX : exemple modifié avec démonstration des nouveautés, gestion des SET/GET LINE/COL/CELL BEHAVIOR, et STV_NOUPDATE, STV_UPDATE

Voir le fichier SAMPLES_V7.PDF livré.

Nouveautés de NS-DK 6.00

Nouveautés de NS-DK 6.00

NS-DK 6.00 intègre de nouvelles fonctionnalités permettant d'optimiser les applications développées avec cet outil. Nat System répond ainsi aux exigences technologiques et techniques des utilisateurs.

[Bases de données](#)

[Evolutions du langage NCL](#)

[Agent Nat System pour QuickTest Professional](#)

[Exemples d'utilisation](#)

Bases de données (NS-DK 6.00)

NS-DK propose le support des nouvelles versions des bases de données présentées dans le tableau ci-dessous.

Versions des bases de données	Drivers
Oracle 11g	NSnnOR11.dll
MySQL 5.0	NSnnMY50.DLL
SQL Server 2008	NSnnMS80.DLL

nn correspond au numéro de version de l'interface que vous avez installé : w2 pour Windows 32 bits, w4 pour Windows 64 bits.

Evolutions du langage NCL (NS-DK 6.00)

Librairie NSMLE

La librairie NSMLE permet de manipuler le texte dans les contrôles ou les fenêtres de type MLE.

Librairie NSXMLE

La librairie XMLE permet de manipuler le contrôle XMLE correspondant à un éditeur de texte étendu.

Librairie NSDYNCTR

La nouvelle librairie NSDYNCTR permet de manipuler dynamiquement vos contrôles.

Librairie NWXSEC

La nouvelle librairie NWXSEC permet de sécuriser vos documents XML en autorisant l'utilisation de signatures numériques et la mise en forme canonique des documents XML.

Librairie NXML

Deux nouvelles fonctions NWX_XSLTRANSFORMFILE_ERRMSG et NWX_XSLTRANSFORM_ERRMSG améliorent la transformation des fichiers XML.

Agent Nat System pour QuickTest Professional (NS-DK 6.00)

Nat System propose, en partenariat avec Hewlett-Packard, la possibilité de tester facilement vos applications développées avec NS-DK.

QuickTest Professional © est une marque déposée de Hewlett Packard.

L'agent Nat System pour QTP est une extension payante des outils de développement Nat System.

Exemples d'utilisation (NS-DK 6.00)

Nat System vous propose de nouveaux exemples d'utilisation (disponibles dans le dossier /samples) vous permettant d'expérimenter les nouvelles fonctionnalités proposées :

- DYNAMICCTRL : création dynamique de contrôles,
- NSTEXTPAD : démonstration du nouveau composant XMLE,
- NSSAX : librairie NS-SAX, pour la manipulation des fichiers XML volumineux.

Nouveautés de NS-DK 5 SP2

Nouveautés de NS-DK 5 SP2

NS-DK 5 SP2 intègre de nouvelles fonctionnalités permettant d'optimiser les applications développées avec cet outil. Nat System répond ainsi aux exigences technologiques et techniques des utilisateurs.

[Compatibilité avec Windows Vista](#)

[Aide en ligne Nat System](#)

[Evolution de l'interface de NS-Design](#)

[Sécurité des documents XML](#)

[Appel de fonctions utilisateur](#)

[Troncature de chaînes](#)

[Evolutions du langage NCL](#)

[Composant NS-PREX](#)

[Nouvelles fonctions et instructions](#)

Compatibilité avec Windows Vista (NS-DK 5 SP2)

Nat System garantit la compatibilité de NatStar 5 SP2 avec le système d'exploitation Windows Vista en environnement de développement et en environnement client.

Aide en ligne Nat System

Aide en ligne Nat System (**NS-DK 5 SP2**)

Pour parfaire la compatibilité avec la plate-forme Vista qui ne supporte plus les fichiers au format .HLP, Nat System fournit désormais une aide en ligne au format CHM (Compiled HTML Help Format).

Ce format étant du HTML compilé, les fichiers occupent beaucoup moins d'espace mémoire que les anciens fichiers d'aide (.HLP et .CNT).

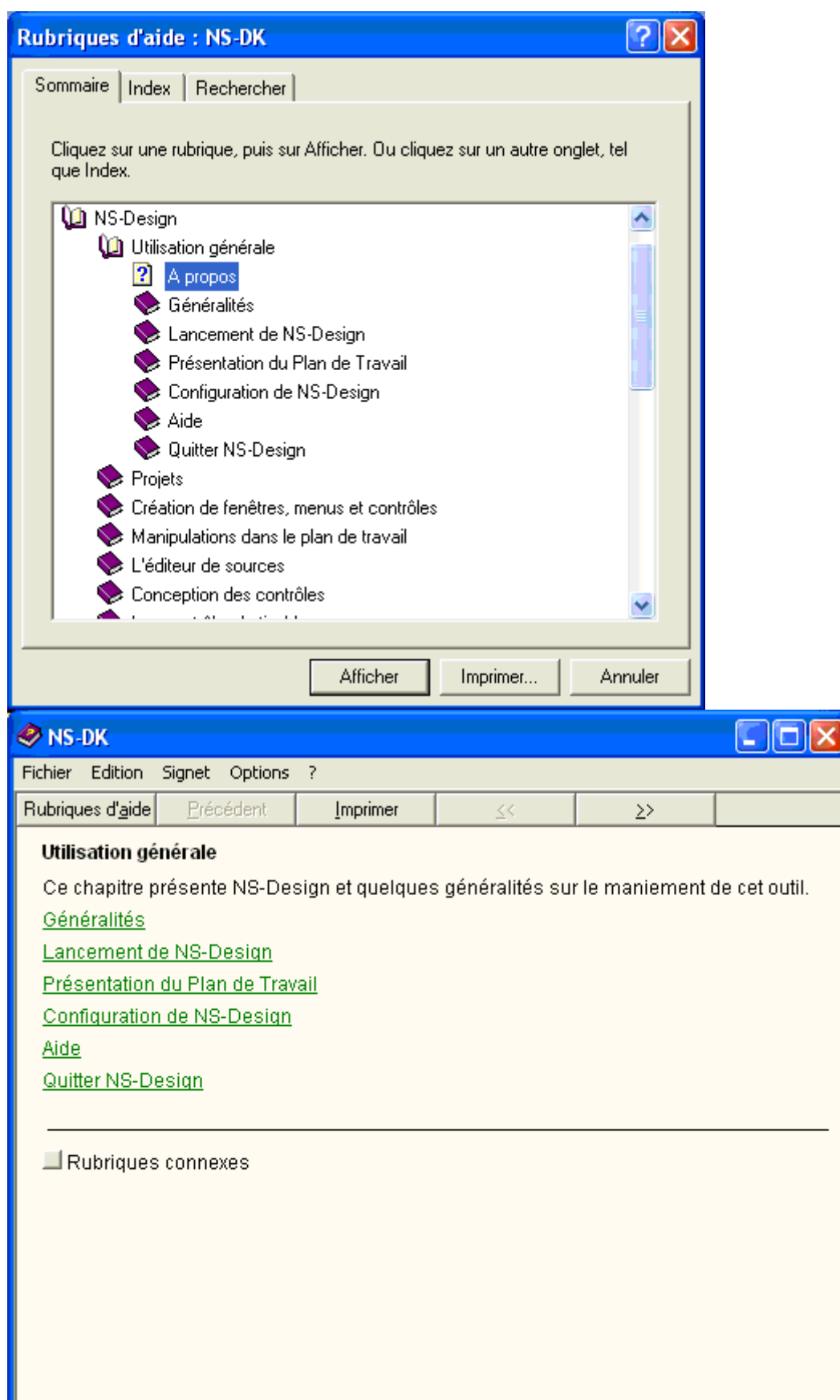
L'aide en ligne au format .CHM doit impérativement être localisé sur le poste client de chaque utilisateur.

Les fichiers .CHM sont uniquement compatibles avec les technologies Internet Explorer et ActiveX. Ils ne sont pas accessibles sur Netscape Navigator ou sur des plates-formes non-Windows (Macintosh, UNIX, ...) car ces derniers ne supportent pas ActiveX.

[Mise en page](#)

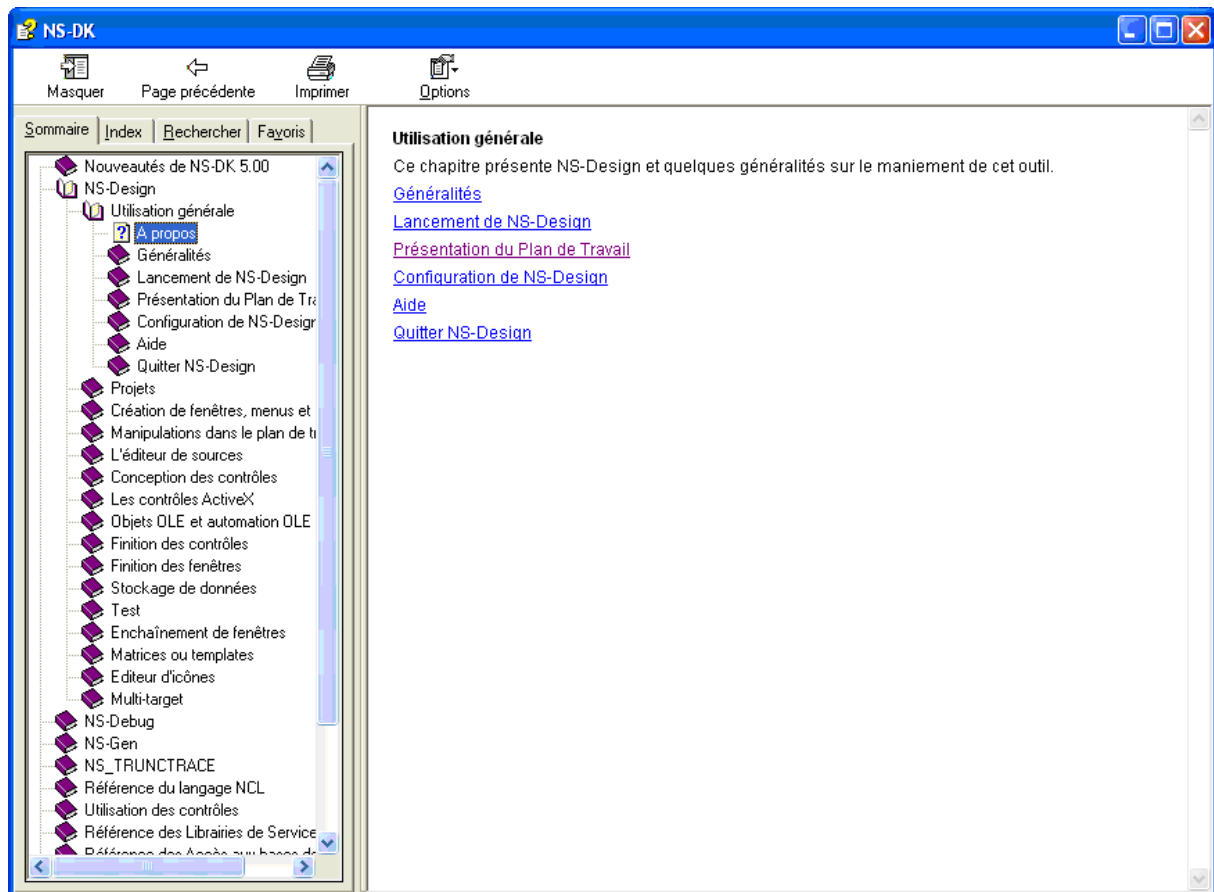
Mise en page (**NS-DK 5 SP2**)


La mise en page dans le format .CHM facilite l'utilisation de l'aide. Dans l'ancienne aide en ligne, la table des matières, l'index et les champs de recherche étaient disposés dans une fenêtre séparée des rubriques d'aide.




Désormais, les différents composants de l'aide en ligne sont affichés dans une fenêtre unique.

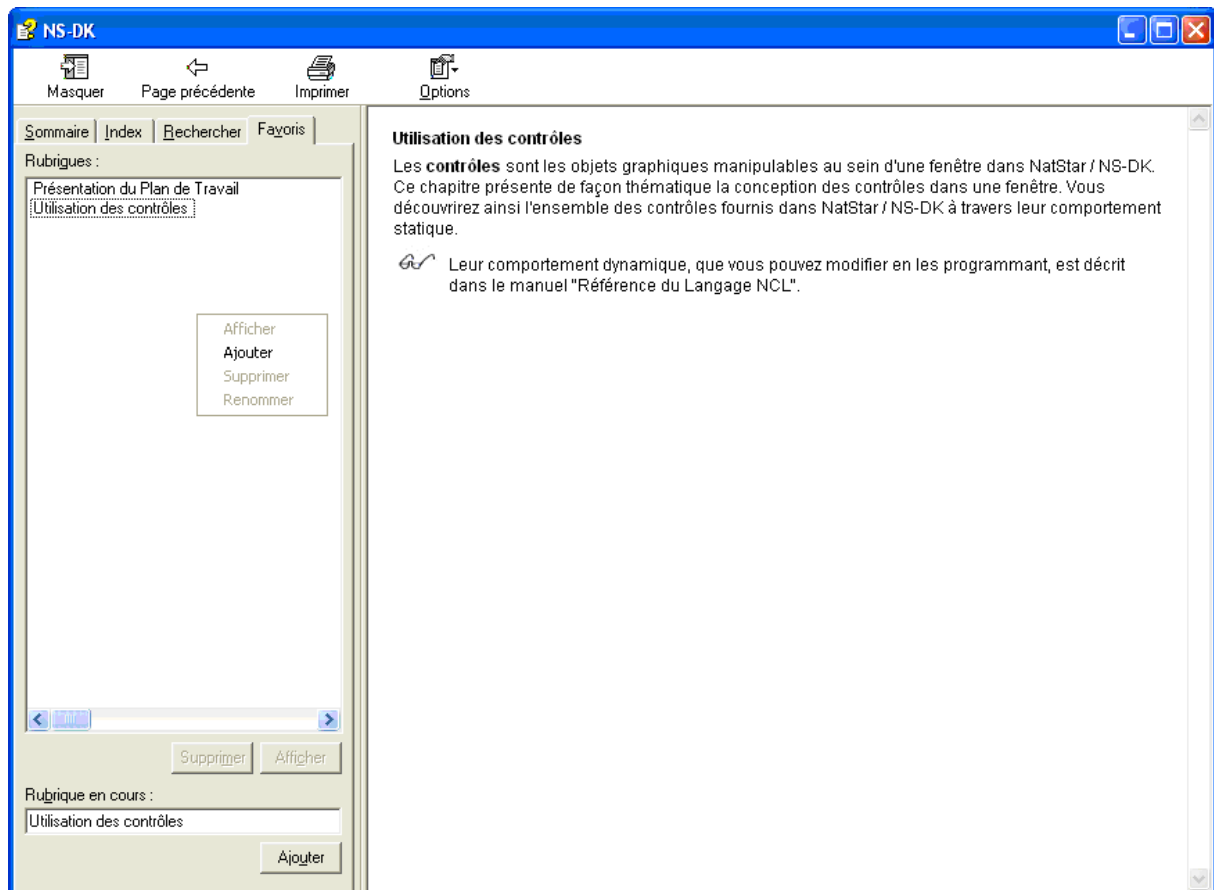
Nouveautés NS-DK



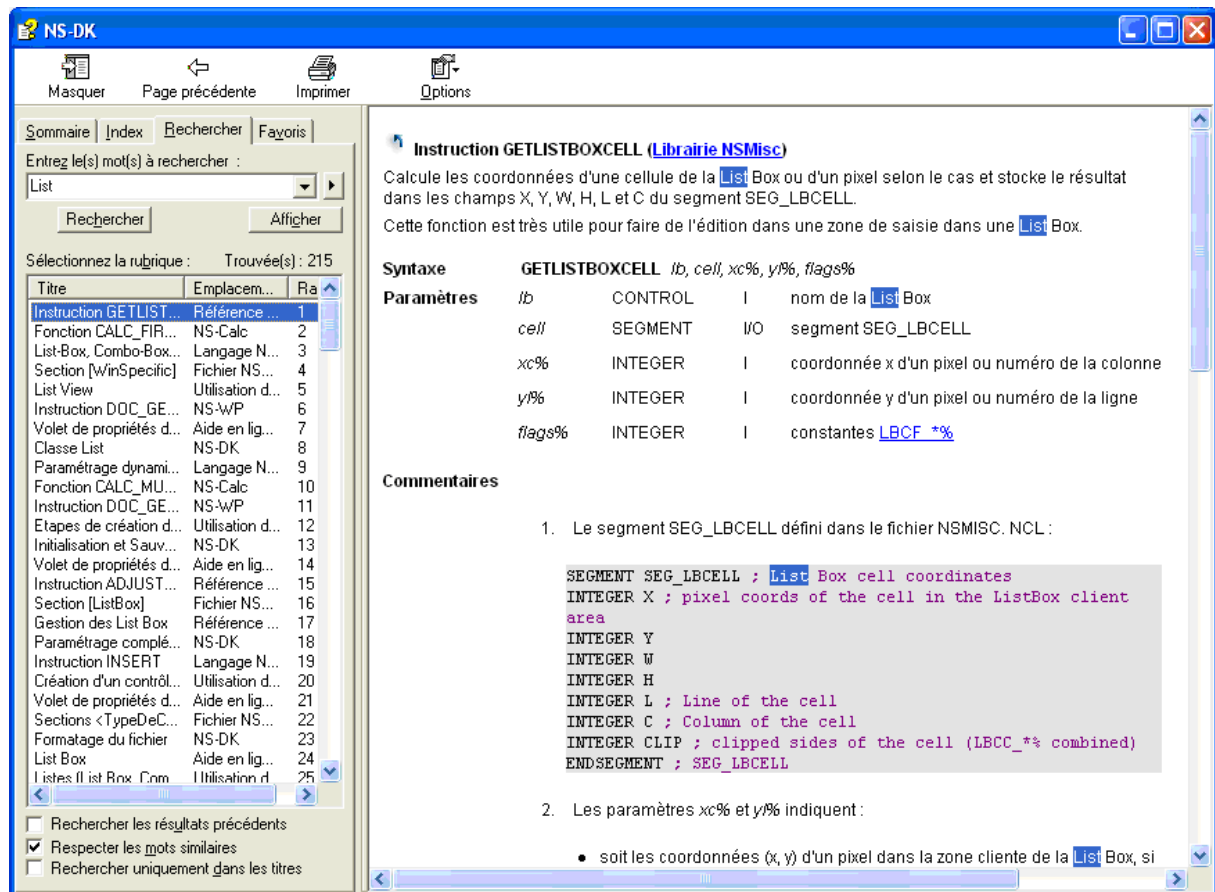
Vous pouvez masquer le panneau de gauche en activant le bouton  **Masquer** et le

réafficher avec le bouton  **Afficher**.

Un nouvel onglet Favoris vous permet de créer des signets sur des pages auxquelles vous souhaitez accéder ultérieurement.



L'onglet Rechercher améliore les critères de recherche en proposant de nouvelles fonctionnalités ou en les rendant plus visibles : Rechercher uniquement dans les titres, Rechercher les résultats précédents et Respecter les mots similaires.



Evolution de l'interface de NS-Design

Evolution de l'interface de NS-Design (NS-DK 5 SP2)

L'interface de NS-Design évolue pour améliorer l'utilisation de l'outil.

[Projet .XNP](#)

[Volet de propriétés](#)

[Browser de ressources](#)

[Navigation dans l'éditeur NCL](#)

[Visualisation de l'ancrage des contrôles](#)

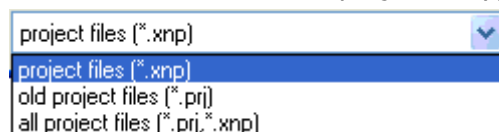
[Transparence des images](#)

[Fenêtre Log](#)

[Compilation](#)

Projet .XNP (NS-DK 5 SP2)

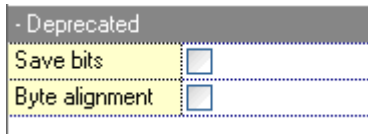
Lors de l'ouverture d'un projet, le type de projet proposé par défaut est désormais .XNP.



Volet de propriétés (NS-DK 5 SP2)

Le volet de propriétés des contrôles graphiques est situé à droite du plan de travail de NS-Design. Il est désormais redimensionnable (à l'aide de la souris) pour mieux visualiser les différentes caractéristiques de l'élément sélectionné.

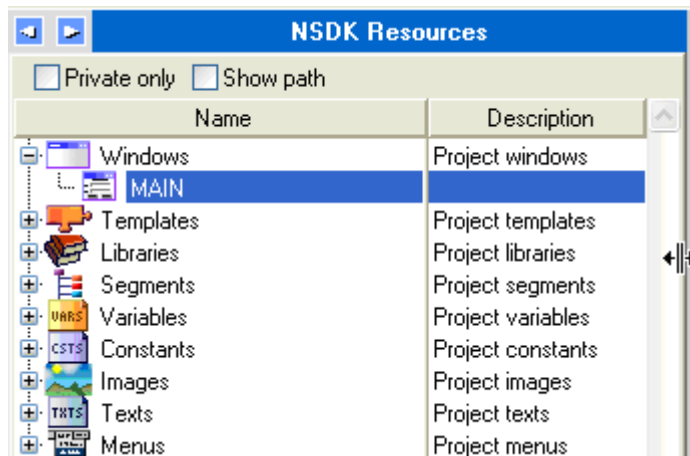
Par ailleurs, deux propriétés des fenêtres sont désormais obsolètes Save bits et Byte alignment.



La propriété Save Bits était utilisée pour accélérer le réaffichage des fenêtres quand les cartes graphiques étaient particulièrement lentes. Par ailleurs, la propriété Byte alignment était utile principalement sur les anciennes cartes graphiques OS/2. Ces deux propriétés sont donc désactivées.

Browser de ressources (NS-DK 5 SP2)

Le browser de ressources est lui aussi désormais redimensionnable.



Navigation dans l'éditeur NCL (NS-DK 5 SP2)

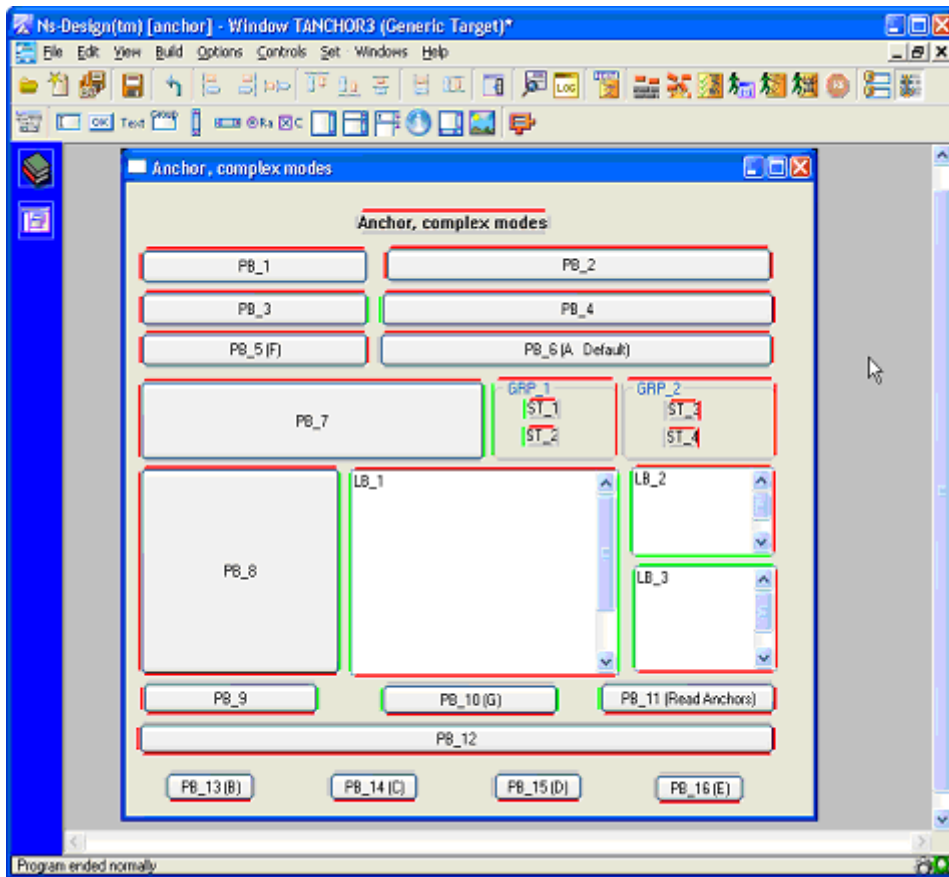
Les touches fléchées [PageUp] et [PageDown] associées à la touche [Ctrl] permettent de naviguer plus facilement dans les blocs d'instruction (INSTRUCTION ... ENDINSTRUCTION, IF ... ENDIF, ...) du code NCL.

Visualisation de l'ancrage des contrôles (NS-DK 5 SP2)

Une nouvelle icône dans la barre d'outils Display Anchors (ou le menu Set/Display Anchors) permet d'afficher l'ancrage des contrôles.

L'affichage de l'ancrage apparaît selon les modalités suivantes :

- rouge, si le côté est ancré en mode fixe sur un bord,
- vert, s'il est ancré en mode proportionnel sur les bords,
- gris, s'il n'y a pas d'ancrage.



Transparence des images (NS-DK 5 SP2)

De nouvelles propriétés de transparence ont été intégrées pour les contrôles Bitmap.

- Transparency	
BMP Transparency	Fixed Color
Corner pixel	Default
Transp Color	
Subst. Color	Default

BMP Transparency

Opaque

Sélectionner pour n'avoir aucun impact sur l'apparence de la bitmap.

Corner Pixel

Sélectionner pour choisir le coin de la bitmap à utiliser en transparence de la bitmap.

Fixed Color

Sélectionner pour choisir une couleur précise.

Corner pixel

Si vous avez choisi Corner Pixel dans BMP Transparency, sélectionnez dans le champ Corner Pixel, le coin de la bitmap dont le pixel sera utilisé pour la transparence:

Top Left : utilise le coin supérieur gauche

Bottom Left : utilise le coin inférieur gauche

Top Right : utilise le coin supérieur droit

Bottom Right : utilise le coin inférieur droit

Default : utilise la couleur par défaut définie pour le projet (en général, le pixel le plus en haut à gauche).

Opaque : aucun impact sur l'apparence de la bitmap.

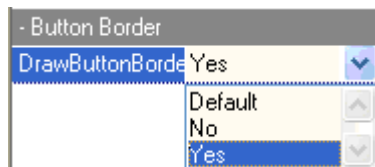
Transp Color

Si vous avez choisi Fixed Color dans BMP Transparency, sélectionnez dans le champ Transp color, la couleur qui sera utilisée pour la transparence.

Subst Color

Couleur de transparence : Couleur qui va remplacer la couleur transparente. Par défaut la couleur de fond de la fenêtre (ou du bouton dans le cas où le bouton est dessiné).

Par ailleurs, la propriété DrawButtonBorder si la bitmap est de type Push-Button permet de redessiner les contours de l'image.



Default : valeur par défaut de l'application de redessin /ou non des boutons de type bitmaps.

No : pas de dessin du contour du bouton

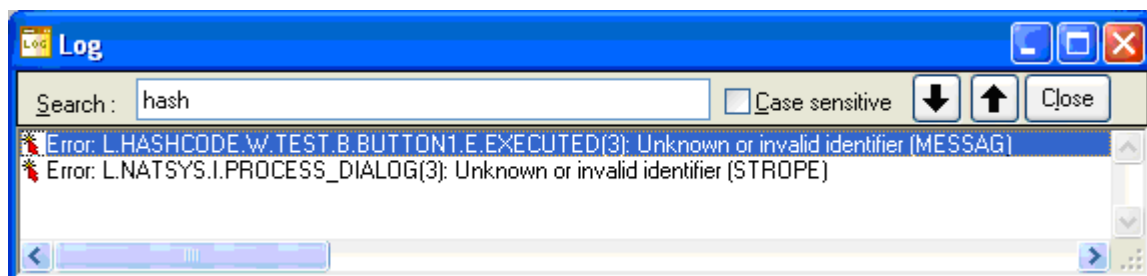
Yes : dessin du contour du bouton sous réserve que les conditions suivantes soient respectées :

Dans ce mode si les champs Bitmap Released, Bitmap Pressed, Bitmap Disabled ne sont pas tous renseignés ou sont identiques le dessin du contour est effectué, sinon on ne fait rien.

Fenêtre Log (NS-DK 5 SP2)

En activant les touches [Ctrl]+F ou le menu contextuel Find, un panneau de recherche apparaît en haut de la fenêtre Log. Ce panneau contient notamment un champ de saisie vous permettant de rechercher une chaîne de caractères dans la liste des erreurs.

En cochant Case sensitive, la recherche respecte la casse de la chaîne de caractères saisie. Les deux flèches vous permettent d'effectuer la recherche vers le haut ou vers le bas.



Activer la touche [Enter] pour commencer la recherche. Le bouton Close ou la touche [Esc] ferme le panneau de recherche.

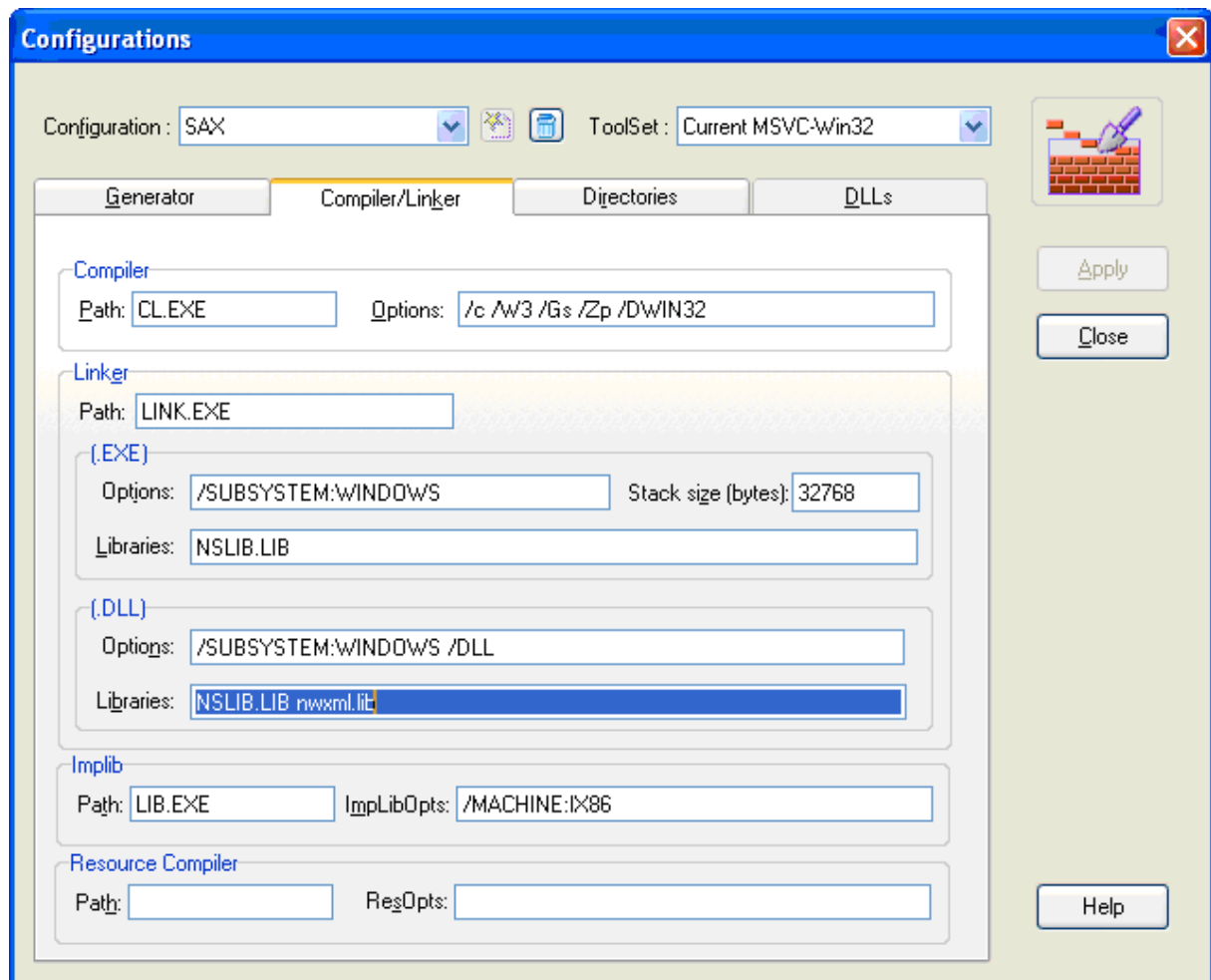
Compilation (NS-DK 5 SP2)

Dans la boîte de configuration (menu Options\Build Configurations ...), il est désormais possible dans les champs Libraries de définir un nom de fichier.

Exemple :

```
@(envVar) responsefile.rsp
```


Ce fichier contient la liste des librairies d'import. Si la variable d'environnement (paramètre envVar) n'est pas précisée, le fichier est recherché dans le répertoire racine du projet (PROJDIR).



Sécurité des documents XML (NS-DK 5 SP2)

Nat System propose la nouvelle librairie NWXSEC. Elle permet de sécuriser vos documents XML en utilisant des signatures numériques et la mise en forme canoniques.

Appel de fonctions utilisateur (NS-DK 5 SP2)

Nat System fournit une nouvelle librairie NSFCTUTIL permettant de définir des fonctions utilisateur.

Ces fonctions de contrôles seront appelées :

- Après une vérification sur du code NCL (touche [F8]) appartenant à une librairie ou à un évènement.
- Avant et/ou après le lancement de la construction de l'application (Build).
- A partir de l'éditeur de code NCL (deux fonctions).
- Quelque soit le contexte (édition NCL, édition scr, compilation ...) (deux fonctions utilisateurs globales).

Troncature de chaînes (NS-DK 5 SP2)

Nat System présente NS-TruncTrace, un nouvel outil qui permet de résoudre les problèmes liés aux troncatures de chaînes lors de l'exécution d'un programme généré avec NS-DK.

Il fonctionne avec deux modes de recherche :

- Mode sans recompilation : lorsqu'une troncature de chaîne est détectée, affichage d'une fenêtre de warning, et/ou génération d'un message d'information dans le fichier de trace spécifié par la variable d'environnement NS-TRACE.
- Mode avec recompilation : lorsqu'une troncature de chaîne est détectée, génération dans le fichier de trace d'un message d'information précisant la localisation de la troncature (fichier/numéro de ligne).

Pour plus d'informations, reportez-vous au "Manuel d'utilisation de NS-TruncTrace".

Evolutions du langage NCL

Evolutions du langage NCL (NS-DK 5 SP2)

NS-DK 5 SP2 propose de nouvelles API permettant de faire évoluer le langage NCL.

[Gestion des menus](#)

[Composant Custom Control IE](#)

[Fichier NSLIB.INI](#)

[Bulles d'aide](#)

[Requêtes HTTP](#)

[Evolution de la librairie NSMisc](#)

[Evolution de la librairie NWXML](#)

Gestion des menus (NS-DK 5 SP2)

Une nouvelle instruction MNU_SET_STATUSTEXT (librairie NSCUST) permet de modifier le texte de la barre d'état.

Une nouvelle constante MENU_SEPARATOR_TOOLBAR% permet de spécifier un séparateur dans la barre d'état. Pour cela, chaque séparateur doit avoir dans sa boîte Modify menu item, le champ Bitmap For spécifié à Toolbar ou Both.

Composant Custom Control IE (NS-DK 5 SP2)

Pour faciliter la gestion du Custom Control IE, Nat System y implémente des paramètres dynamiques.

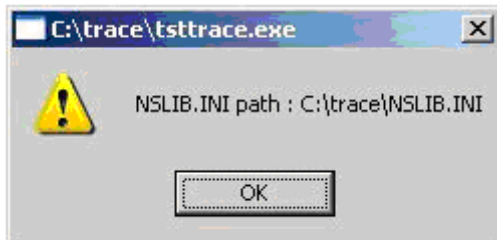
Fichier NSLIB.INI

Fichier NSLIB.INI (NS-DK 5 SP2)

Lecture du fichier NSLIB.INI (NS-DK 5 SP2)

La lecture du fichier NSLIB.INI évolue pour permettre d'afficher la localisation du fichier lu par l'exécutable (dans une boîte de message), ainsi que les sections et paramètres utilisés (dans le fichier trace).

Pour connaître la localisation du fichier NSLIB.INI lu par l'exécutable, positionner la variable d'environnement NS_SHOW_INI à YES OU TRUE. Un message d'information s'affiche.



Par ailleurs, si on active le mode trace à la compilation d'un projet les sections et les paramètres lus et utilisés dans le fichier NSLIB.INI sont indiqués dans le fichier trace.

```
NSDK:12:03:19
*****
NSDK:12:03:19 NSGENTRACE-exe { Exe : C:\trace\TSTTRACE.EXE}
NSDK:12:03:19
*****
NSDK:12:03:19 NSGENTRACE-ini { Nslib.ini file location :
  C:\trace\NSLIB.INI }
NSDK:12:03:19 USED TOPICNAMES {
NSDK:12:03:19     [System]
NSDK:12:03:19         RespectWindowSettings=False
NSDK:12:03:19     [Entry.Translations]
NSDK:12:03:19         ToggleInsert=F11
NSDK:12:03:19         ToggleInsert=Insert
NSDK:12:03:19     [List.Translations]
NSDK:12:03:19         PreviousLine=Down
NSDK:12:03:19         NextLine=Up
NSDK:12:03:19     [Frame.Translations]
NSDK:12:03:19         Help=F1
NSDK:12:03:19     [Dialog.Translations]
NSDK:12:03:19         ToggleSelect=F10
NSDK:12:03:20     [Window]
NSDK:12:03:20         HideOnClose=True
NSDK:12:03:20         BroadcastCheck=True
NSDK:12:03:20     [Combo]
NSDK:12:03:20         ExecuteDefPush=True
NSDK:12:03:20         FilterButtonDowns=True
NSDK:12:03:20     [Entry]
NSDK:12:03:20         InsertMode=True
NSDK:12:03:20         MouseAutoSelect=True
NSDK:12:03:20     [WinFontSubstitutes]
NSDK:12:03:20         Helv=MS Sans Serif
NSDK:12:03:20     [PMFontSubstitutes]
NSDK:12:03:20         MS Sans Serif=Helv
NSDK:12:03:20     [PMSpecific]
NSDK:12:03:20         MsgQueueSize=100
NSDK:12:03:20     [WinSpecific]
```

```

NSDK:12:03:20      3DDialogs=True
NSDK:12:03:20      UseCtl3D=False
NSDK:12:03:20      MoveWithOwner=True
NSDK:12:03:20      UseLookXp=trueNSDK:12:03:20
      IsTabsSelectedAfterInit=True
NSDK:12:03:20      [Entry]
NSDK:12:03:20      InsertMode=True
NSDK:12:03:20      MouseAutoSelect=True
NSDK:12:03:20      [Miscellaneous]
NSDK:12:03:20      Zooming=True
NSDK:12:03:20      BmpZooming=True
NSDK:12:03:20      NoMenuPictures=False
NSDK:12:03:20      NoMenuBarPictures=True
NSDK:12:03:20 } USED TOPICNAMES
NSDK:12:03:20
*****
NSDK:12:03:20
*****

```

Evolution de la section [Winspecific] (NS-DK 5 SP2)

Dans la section [Winspecific], le paramètre UseLookXP évolue et un nouveau paramètre UseClipChildren est intégré.

```
UseLookXP= True | False
```

Permet d'avoir un aspect Windows XP ou Vista sur l'interface utilisateur. Positionnée à False, cette option permet d'avoir le look Windows 2000 dans les applications utilisant le runtime Nat System. A True, elle laisse Windows dessiner les contrôles. Dans ce cas l'aspect est vraiment celui de Windows, mais on n'a pas la maîtrise totale de ce qui est dessiné en termes de couleur et de relief. Si les couleurs et relief des contrôles de l'application sont des informations utilisateurs, il est préférable de positionner ce flag à False.

```
UseClipChildren= True | False
```

Dans le cas où on utilise l'ancrage dans une boîte de dialogue retaillable et que certains de ses contrôles se repeignent mal (par exemple des Custom Controls), on peut corriger le problème en mettant le flag useClipChildren=False. Cette option impacte l'application entière, contrairement au paramètre dynamique CLIENT.CLIPCHILDREN qui ne concerne que le pseudo-contrôle CLIENT d'une fenêtre.

Bulles d'aide (NS-DK 5 SP2)

Il est désormais possible d'afficher une bulle d'aide sur plusieurs lignes en insérant les caractères "#13" dans le libellé.

Exemple :

```
OK.tooltip = "Première Ligne "&#13;" - Deuxième Ligne"&#13;" - Troisième
Ligne"&#13;" - Quatrième Ligne"
```

Quand la bulle d'aide s'affiche sur une seule ligne le texte reste centré. Si la bulle d'aide est sur plusieurs lignes, le texte est aligné à gauche.

Requêtes HTTP (NS-DK 5 SP2)

Pour faciliter l'utilisation des requêtes HTTP, de nouvelles fonctions et instructions permettent de manipuler des données de type DynStr :

- NS_HTTP_GET_HEADEREX
- NS_HTTP_GETLINEEX
- NS_HTTP_POSTEX
- NS_HTTP_READEX

Evolution de la librairie NSMisc (NS-DK 5 SP2)

- La librairie NSMisc est complétée par de nouvelles fonctionnalités permettant de gérer des fichiers texte dépassant la limite des 255 caractères (T_READLNEX\$ et T_WRITELNEX).
- La fonction F_LoadFile\$ qui retourne la totalité du contenu d'un fichier, indiqué en paramètre, dans une seule chaîne de caractères de type DynStr. Ainsi, les sauts de ligne sont présents comme caractères de contrôle.
- L'instruction F_SaveFile qui permet de sauvegarder du texte dans un fichier.
- La fonction GETLONGENV\$ qui retourne la valeur d'une variable d'environnement dans un type DYNSTR.

Evolution de la librairie NWXML (NS-DK 5 SP2)

La librairie NWXML est complétée par de nouvelles fonctionnalités permettant de gérer des chaînes de caractère de dimension variable de type Dynstr : NWX_NODE_OUT_DYNSTR et NWX_DOCUMENT_OUT_DYNSTR.

Composant NS-PREX (NS-DK 5 SP2)

Nat System propose un nouveau composant de communication inter-technologies, permettant, depuis un navigateur Web d'appeler une application NatStar (sous forme de DLL ou d'exécutable) en lui passant des paramètres.

Ce composant fonctionne sur tous les navigateurs du marché (Internet Explorer, Firefox...), sans ActiveX et sans toucher aux paramètres de sécurité du système client.

Ce composant permet de charger une DLL NatStar et d'appeler plusieurs instructions en leur passant des paramètres. Par ailleurs, il permet également de lancer un exécutable NatStar.

Ce composant est d'une grande simplicité d'utilisation. Une seule ligne de JavaScript permet de lancer l'appel au composant NS-PREX. Et, la description des exécutables et DLL à lancer se trouve dans un fichier .INI unique afin de simplifier le JavaScript.

Pour plus d'informations, reportez-vous à la documentation correspondante.

Nouvelles fonctions et instructions

Nouvelles fonctions et instructions (NS-DK 5 SP2)

[Librairie NSHelp](#)

[Librairie NSMisc](#)

[Librairie NSCust](#)

[Librairie NSHttp](#)

[Librairie NWXml](#)

Librairie NSHELP (NS-DK 5 SP2)

Pour rappel, la librairie NSHelp fournit les fonctions et instructions permettant d'inclure et d'exploiter dans une application NatStar l'aide en ligne générée sous forme de fichiers .HLP ou .CHM.

NatStar 5 SP2 implémente une nouvelle constante HLPMODE_CHM% permettant de charger une aide au format CHM. et une instruction et ses constantes associées pour gérer l'ouverture de l'aide : HLPHTMLHELPTYPKEY, NS_HH_*

Librairie NSMisc (NS-DK 5 SP2)

Ces deux nouvelles fonctionnalités permettent de gérer des fichiers texte en utilisant des DYNSTR : T_READLNEX\$ et T_WRITELNEX.

Librairie NSCUST (NS-DK 5 SP2)

La librairie NSCUST propose l'instruction MNU_SET_STATUSTEXT qui permet de modifier le texte de la barre d'état.

Librairie NSHTTP (NS-DK 5 SP2)

La librairie NSCUST propose les nouvelles instructions suivantes permettant de gérer le type DynStr.

- NS_HTTP_GET_HEADEREX renvoie le header d'une réponse http sous forme de DynStr.
- NS_HTTP_GETLINEEX renvoie une ligne du flux HTTP sous forme de DynStr.
- NS_HTTP_POSTEX envoie une requête HTTP en utilisant la méthode POST.
- NS_HTTP_READEX retourne le nombre d'octets lus.

Librairie NWXML (NS-DK 5 SP2)

La librairie NWXML propose la fonction NWX_NODE_OUT_DYNSTR qui permet d'exporter le contenu d'un nœud vers une chaîne dynamique et NWX_DOCUMENT_OUT_DYNSTR.

Nouveautés de NS-DK 5.00

Nouveautés de NS-DK 5.00

NS-DK 5.00 répond aux besoins des utilisateurs qui souhaitent, tout en conservant la simplicité du produit, bénéficier d'évolutions pertinentes qui assureront la pérennité des applications.

[Environnement technologique](#)

[Nouvelle interface de NS-Design](#)

[Suppression des limitations historiques](#)

[Evolution du langage NCL](#)

[Contrôles graphiques et affichage](#)

[Les fichiers NSDKCFG.INI et NSDKLOC.INI](#)

[Génération et déboguage](#)

[Nouvelles fonctions et instructions](#)

Environnement technologique

Environnement technologique (**NS-DK 5.00**)

L'environnement principal de NS-DK reste Windows 32 bits.

Cependant, une ouverture est proposée vers le client graphique de Linux. Une application NS-DK peut donc s'exécuter sur un poste Linux moyennant quelques contraintes de portabilité.

De même, côté serveur pour ceux qui utilisent également NS-DK comme outil de développement de leurs batchs les plates-formes 64 bits seront supportées.

Nous souhaitons offrir une alternative à Windows pour des "petits" serveurs grâce à Linux et MySQL

NS-DK supporte de nouvelles bases de données MySQL, Oracle 10G et SQL Server 2005.

[Windows](#)

[Unix](#)

[Linux](#)

[Base de données](#)

Windows (**NS-DK 5.00**)

Les plates-formes Windows supportées sont :

- Windows XP SP1 et SP2,
- Windows 2003,
- Windows 2000,
- Windows NT 4.0,
- Windows 98 SE.

Unix (**NS-DK 5.00**)

Les cibles serveurs Unix sont désormais disponibles en 32 et 64 bits. Les serveurs disponibles sont :

- Aix 5.2 en 32 et 5.3 en 64 bits,
- Hp-UX 11.1 PA-Risc 2 en 32 et 64 bits,
- Sun Solaris 10 en 32 et 64 bits.

Linux (**NS-DK 5.00**)

NS-DK 5.00 dispose de deux cibles Linux Red Hat 3.0 pour processeur Intel. Une cible serveur 32 bits permettant de faire des batchs et une cible graphique Linux permettant de faire tourner une application NS-DK affichant également des fenêtres.

Les deux bases supportées pour cet environnement sont Oracle (9.2 et 10G) et MySQL.

Base de données (**NS-DK 5.00**)

Les bases de données supportées nativement sont :

- Oracle 9i Release 2 (9.2), 10G (10.1) et 10.2
- DB2 8.1 et DB2 9.
- Sybase 12.5.3 et 15
- MySQL 4.1
- SQL Server 2000 et 2005.

De plus, un driver ODBC 3.51 est mis à disposition. Ce driver permet l'accès à toutes bases de données disposant d'un driver ODBC. Il peut être notamment utilisé pour accéder à MS-Access comme alternative de remplacement à NS-DBR ou MySQL comme base de développement ou base légère embarquée.

Nouvelle interface de NS-Design

Nouvelle interface de NS-Design (**NS-DK 5.00**)

Nat System propose une toute nouvelle interface de l'outil de développement NS-DK. Cette interface permet une meilleure navigation entre les ressources d'un projet, ainsi qu'une plus grande visibilité des propriétés des contrôles.

[Nouvelle organisation du plan de travail](#)

[Browser de ressources](#)

[Le volet de propriétés](#)

[Nouvel éditeur de fenêtre](#)

[Editeur NCL](#)

[Macros](#)

[Nouveaux menus](#)

[Nouveaux raccourcis clavier](#)

[Nouvelle barre d'outils](#)

[Modification de certaines boîtes de dialogues](#)

[Paramétrage d'un projet](#)

[Génération, compilation et édition de liens](#)

Nouvelle organisation du plan de travail (**NS-DK 5.00**)

Le plan de travail contient vos fenêtres de travail. C'est désormais une zone MDI (Multiple Document Interface). Vous pouvez donc ouvrir simultanément plusieurs fenêtres dans le plan de travail et basculer facilement de l'une à l'autre. Vous pouvez travailler avec plusieurs objets en même temps en utilisant un grand nombre de commandes, qui varient en fonction du type de fenêtre.

Le plan de travail est décomposé en 4 grandes zones :

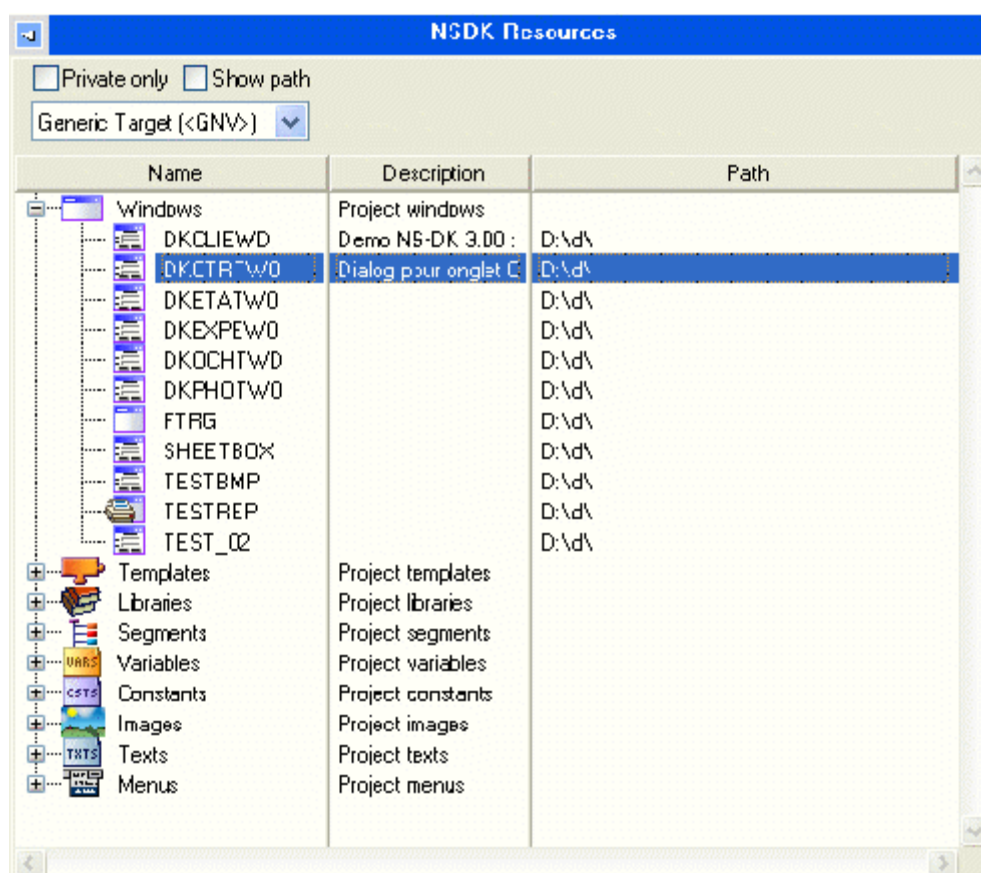
- La barre de menu avec ses deux [barres d'outils](#) en haut
- Le [browser de ressources](#) à gauche
- Le [volet de propriétés](#) des contrôles graphiques à droite
- La zone de travail qui permet [l'édition des fenêtres](#) et des [scripts NCL](#) (bibliothèques et événements).

Browser de ressources (NS-DK 5.00)

Le browser de ressource présente en permanence une vision arborescente des ressources de votre projet. Ce browser est escamotable afin de rendre disponible un espace de travail complet.

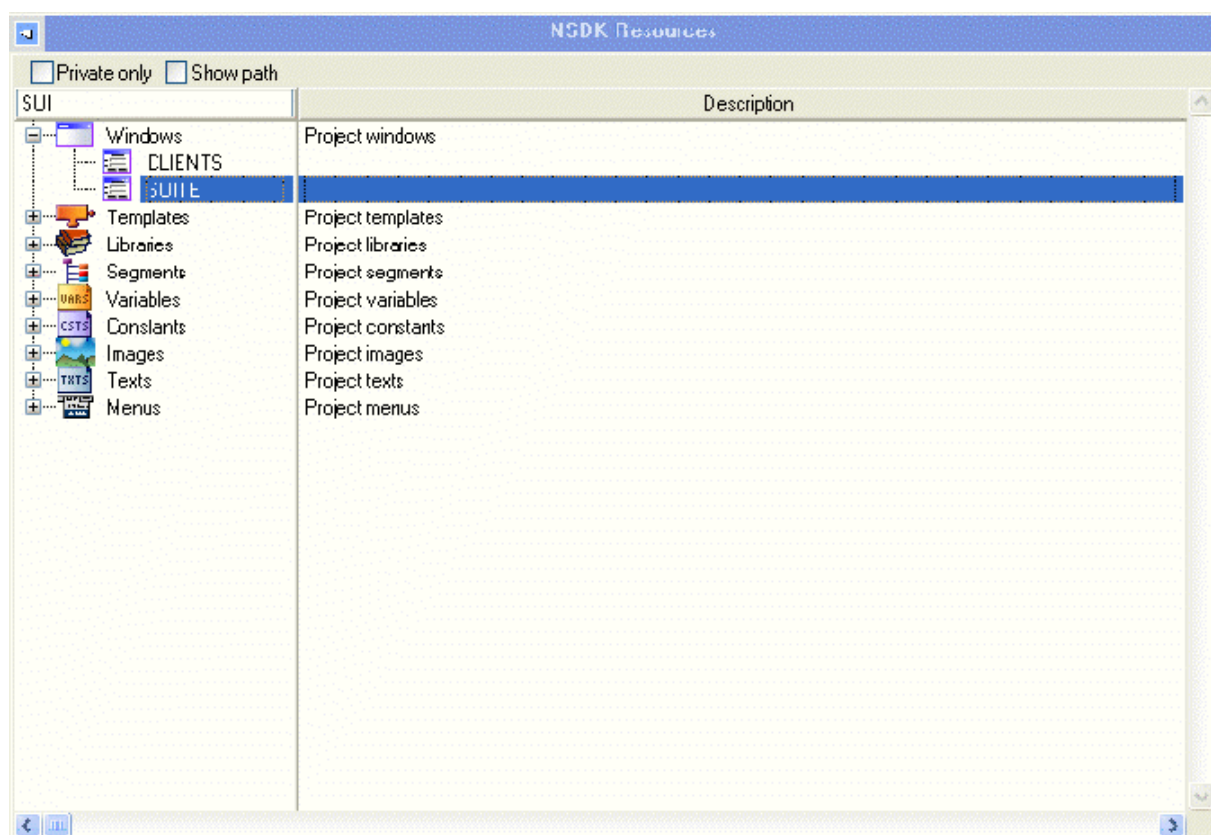
Les ressources sont organisées suivant leur type. Les regroupements disponibles sont légèrement différents de ceux des anciennes versions de NS-DK. La liste est la suivante :

- Windows
- Templates
- Libraires
- Segments
- Constantes
- Variables
- Images qui regroupent BMP, ICO et PTR et ajoutent la gestion des ressources de types JPEG et GIF
- Textes
- Menus externes : un nouveau type de ressources spécifiques à cette version



Il sera possible de limiter l'affichage exclusivement aux ressources interne au projet (celle qui sont modifiables). De plus une iconographie différente (ajout d'un petit cadenas) permettra une identification rapide des ressources non modifiables.

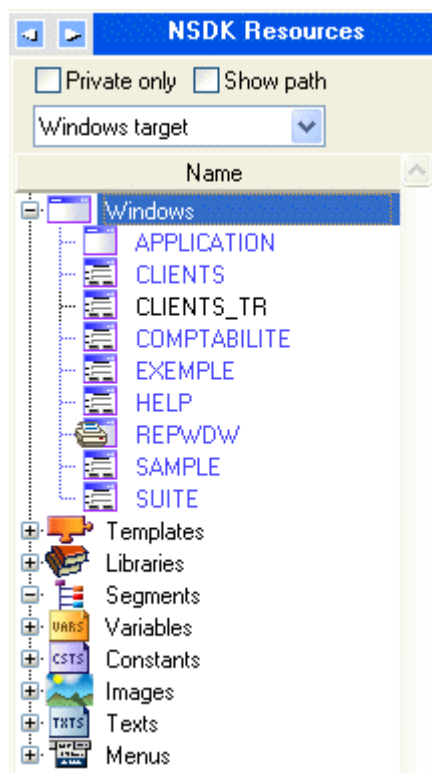
Pour rechercher une ressource, cliquez sur la colonne Name en activant en même temps la touche [Ctrl] et saisissez le nom.



Les touches [+] et [-] du clavier permettent d'ouvrir ou fermer les arborescences.


Les différents types d'images Icons, Pointers et Bitmaps ont été rassemblés dans le type Images.

Si votre projet est multi-target, une liste déroulante vous permet dans le browser de ressources de sélectionner les ressources correspondant à une target spécifique. Les ressources des autres targets sont affichées en bleu.



Le volet de propriétés (NS-DK 5.00)

Pour ouvrir le volet de propriétés, sélectionner un élément dans le plan de travail (contrôle, fenêtre, ...) et :

- double-cliquer dessus avec le bouton droit de la souris.
- activer le bouton Properties  de la barre d'outils.
- sélectionner le menu Edit / Properties.

- Main	
Name	MANUFACTUREF
Description	
Tooltip	
Input	
Mode	
Type	Default
Max. length	
Characters	'0'..'9'
Format	0
Tab to	
Justification	Left
- Layout	
X	213
Y	117
Width	200
- Behavior	
Uppcase	<input checked="" type="checkbox"/>
Skip blanks	<input checked="" type="checkbox"/>
No blanks	<input checked="" type="checkbox"/>
Auto-tab	<input checked="" type="checkbox"/>
Condition	
Hide text	<input checked="" type="checkbox"/>
Locked text	<input checked="" type="checkbox"/>
Disabled	<input checked="" type="checkbox"/>
Auto-scroll	<input checked="" type="checkbox"/>
- Deprecated	
Output	
- Look	
Shadow	Dark (sunken)
Margin	<input checked="" type="checkbox"/>
- Font	MS Sans Serif;8
Name	MS Sans Serif
Size	8
Bold	0
Italic	0
Underscore	0
Strikeout	0
Foreground color	0
Background color	0



Le volet de propriétés des contrôles graphiques est situé à droite du plan de travail de NS-Design. Elle présente les différentes caractéristiques du contrôle sélectionné. Ainsi, vous ne pouvez l'ouvrir qu'une fois et visualiser les propriétés des différents contrôles de

vosre fenêtre en les sélectionnant dans la liste déroulante. L'affichage des propriétés au moyen du volet de propriétés est donc plus opérationnel que les boîtes Info de chaque contrôle.

Cinq types principaux de propriétés sont présentés : Main, Events, Layout, Behavior, Look :

- Main intègre les propriétés principales de l'élément : nom, description, type du contrôle ou de la fenêtre, texte de présentation,...
- Events présente les événements. Ceux implémentés sont indiqués par un coche. En cliquant sur un des événements INIT, HELP, GETFOCUS, ... on déclenche l'ouverture de la fenêtre Events appropriée.
- Layout permet de disposer le contrôle dans la fenêtre en indiquant sa position verticale, horizontale et sa dimension en largeur et en hauteur. Une Combo-box éditable permet d'agrandir ou de réduire les dimensions rapidement.
- Behavior rassemble les propriétés comportementales : désactivation du contrôle, absence de focus, mode d'appel du contrôle, ...
- Look réunit les propriétés de présentation visuelle du contrôle : police, taille de la police, texte en gras, italique, souligné, couleur du texte, ...

Vous pouvez retrouver d'autres propriétés en fonction du type de contrôle sélectionné : Deprecated, Miscellaneous, ...

L'icône  indique que la fenêtre est verrouillée, et n'est donc pas modifiable. L'icône  indique que la fenêtre n'est pas verrouillée.

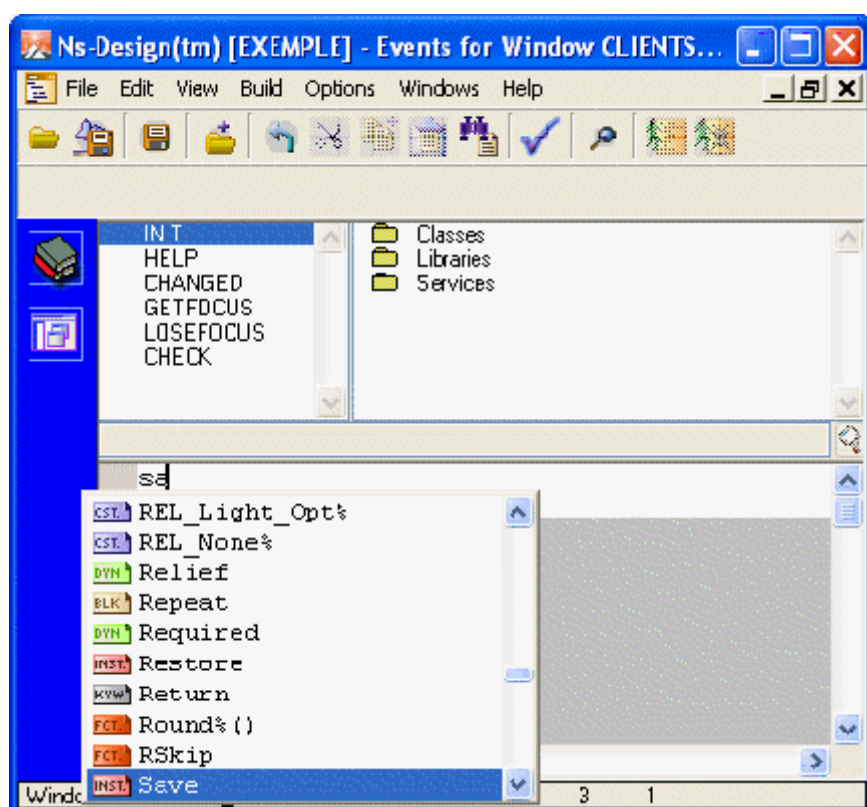
Nouvel éditeur de fenêtre (**NS-DK 5.00**)

Le nouvel éditeur de fenêtre de NS-DK remplace la boîte de dialogue affichant les propriétés d'un contrôle par un browser de propriétés sur la partie droite. Ce browser est repliable afin d'agrandir l'espace de travail.

La propriété ToolTip (bulle d'aide) d'un contrôle devient accessible dans le browser de propriétés.

Editeur NCL (**NS-DK 5.00**)

L'éditeur de script NCL (Librairies et événements) évolue : il permettra de faire de l'auto-complétion sur les verbes du NCL en activant les touches [Ctrl]+[barre d'espace].



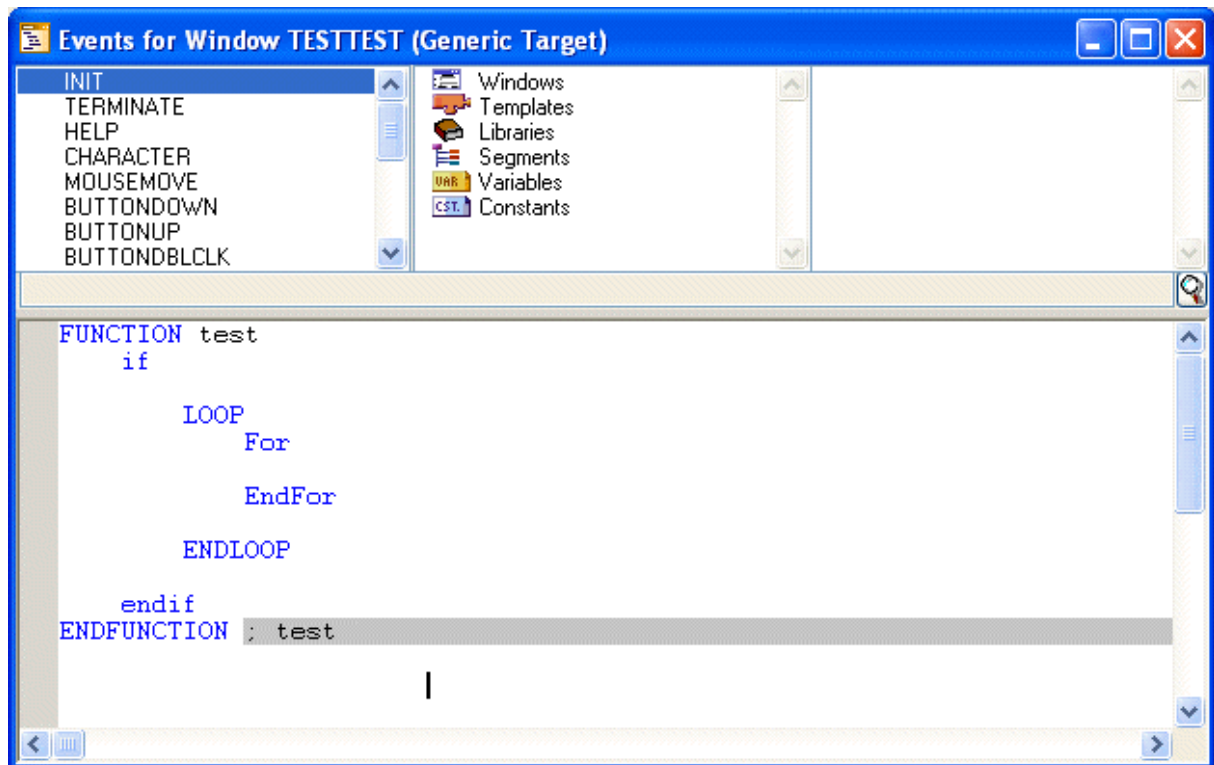
Par ailleurs, l'auto-complétion s'effectue également sur les instructions END* en fonction des instructions disposées avant : EVALUATE, FUNCTION, IF, INSTRUCTION, LOOP, SEGMENT, EVALUATE/WHERE, WHILE, FOR, INDEXES.

|| L'auto-complétion ne fonctionne pas avec le bloc REPEAT/UNTIL.

Cette auto-complétion s'effectue lorsqu'on quitte la ligne (flèches haut/bas, touche [PageUp], touche [PageDown], clic souris hors de la ligne courante contenant le END, touche [Return]...).

L'auto-complétion des instructions END* respecte l'indentation et les majuscules de l'instruction qu'elle termine.

Les instructions ENDFUNCTION et ENDINSTRUCTION sont suivies d'un point virgule et du nom de la fonction/instruction.



Exemple :

En intégrant l'exemple suivant dans l'éditeur de source de NatStar :

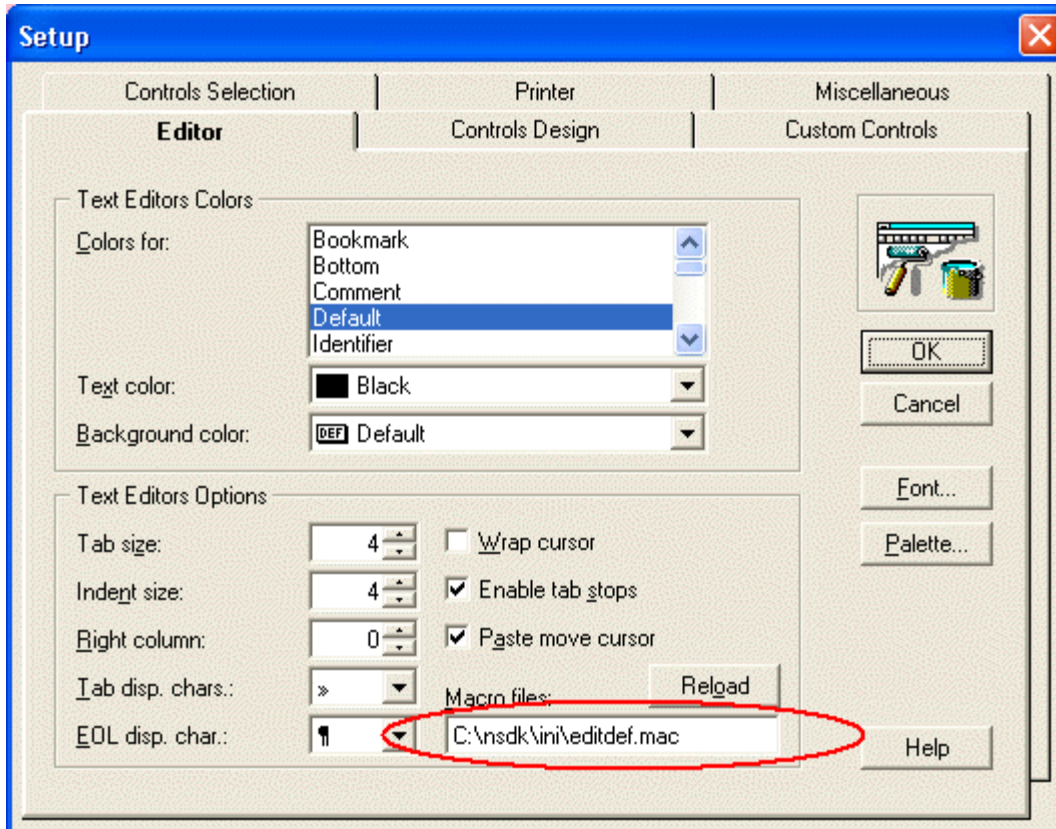
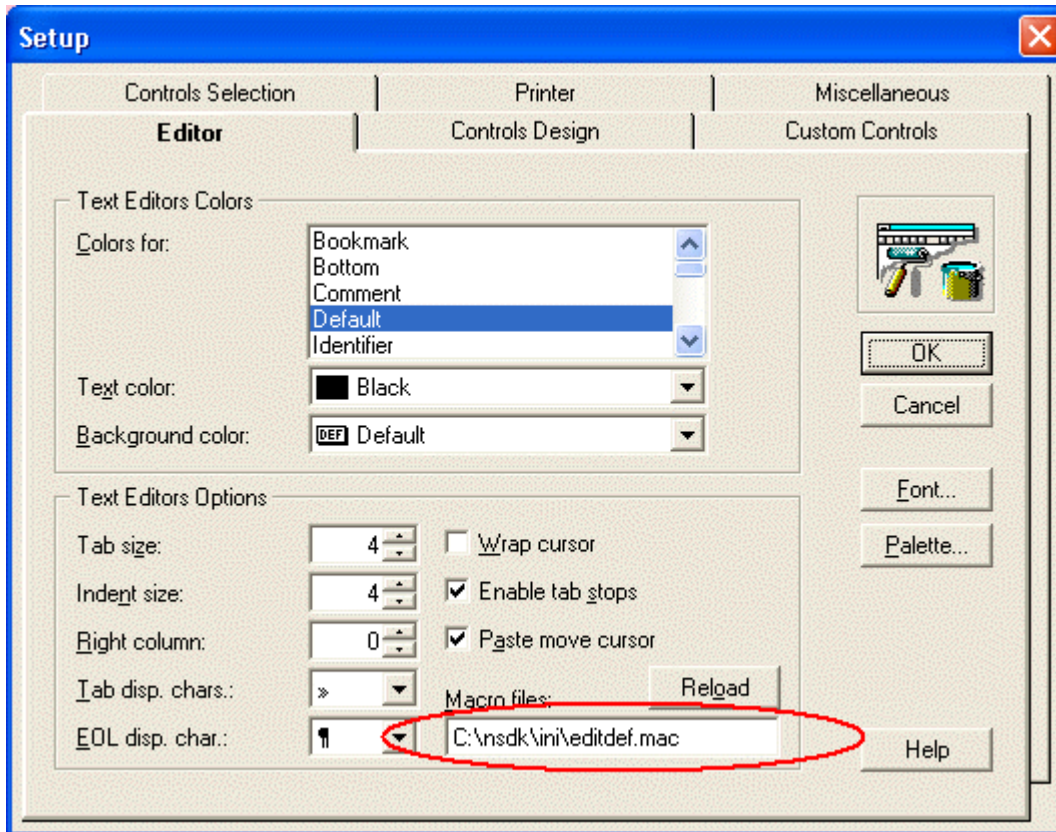
```
Function Toto$
  while
    Evaluate
      WHERE
        for
          bidule
          end
          end
          end
          end
          end
  end
```

Puis, en quittant la ligne contenant la dernière instruction END, toutes les instructions END sont transformées (si on sort d'une ligne contenant une instruction END qui n'est pas la dernière, seul ceux qui la précèdent sont mises à jour) et on obtient :

```
Function Toto$
  while
    Evaluate
      WHERE
        for
          bidule
        endfor
      ENDWHERE
    EndEvaluate
  endwhile
EndFunction ; Toto$
```

Macros (NS-DK 5.00)

Il est également possible de paramétrer des macros pour les séquences de code qui se répètent souvent. Ces macros doivent être définies dans un fichier d'extension .mac. Le chemin de ce fichier est paramétré au niveau de l'onglet Option/Setup/Editor.



Dans ce fichier plusieurs macros peuvent être définies et doivent respecter la syntaxe suivante. Un exemple de fichier macros (editdef.mac) est livré avec NS-DK (présent dans le répertoire INI lors de l'installation).

Une fois ce fichier installé et son chemin paramétré au niveau de l'onglet Option/Setup, pour créer un en-tête de fonction par exemple, on tape cf dans l'éditeur :

```

: Library EDITMAC.NCL
: Created 20/10/06
cf

```

Curseur sur la fin de la ligne et appuie sur les touches [Ctrl]+[Shift]+[Espace] et on obtient :

```

: Library EDITMAC.NCL
: Created 20/10/06
cf
:
: >> Function [***]
: >> Created [***]1999 by [***]
: >> Modified [***]1999 by [***]
: >> [***]
: >>
: >> Parameters
: >> >> [***]
: >>
: >> Return value
: >> >> [***]
: >>
: >> Local variables declaration
: >>
: >>
Things_Trace "[***] : ***** BEGIN *****"
Things_Trace "[***] : ***** Parameters In =>" & [***] & ";"
: >>
Things_Trace "[***] : ***** Parameters Out=>" & [***] & ";" Return Value =>" & [***] & ";"
Things_Trace "[***] : ***** END *****"

```

Nouveaux menus (NS-DK 5.00)

L'organisation des menus est légèrement revue. Un certain nombre d'action sont rendus disponibles et accessibles plus rapidement comme le renommage et la suppression d'une ressource.

Nouveaux raccourcis clavier (NS-DK 5.00)

Les tableaux suivants récapitulent les nouveaux raccourcis clavier permettant d'exécuter plus rapidement les tâches courantes.

Raccourcis d'éléments du menu File

Raccourci clavier	Description
[Ctrl]+S	Sauvegarde de la fenêtre courante.
[Shift]+[F2]	Sauvegarde toutes les fenêtres ouvertes.
[Ctrl]+[F4] ou [Esc]	Ferme la fenêtre courante.
[Alt]+[F4]	Ferme NS-DK.

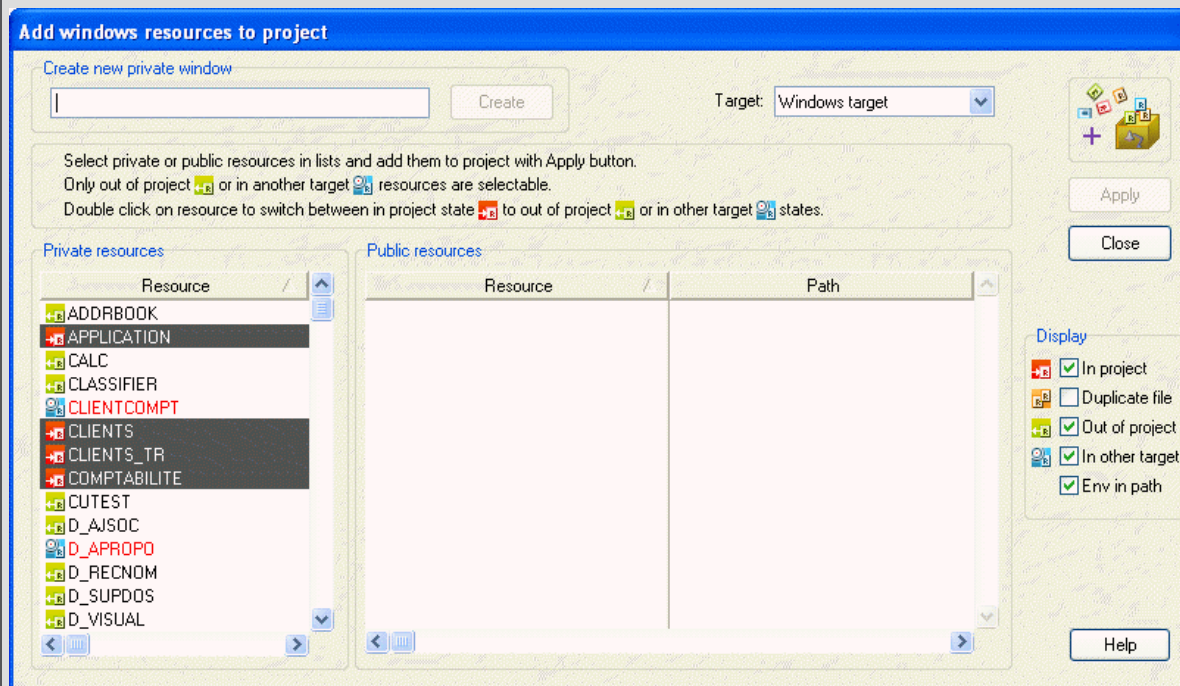
Raccourcis d'éléments du menu Edit

Raccourci clavier	Description
-------------------	-------------

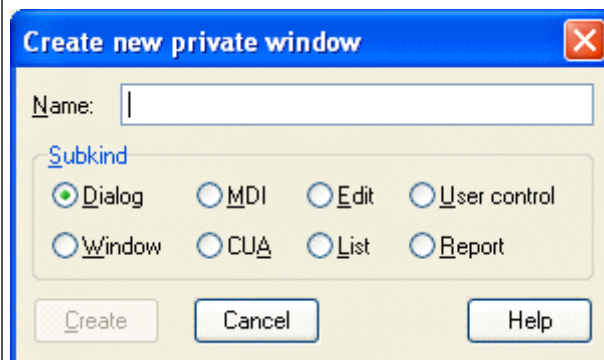
[Alt]+[Backspace]		Annule la dernière action.
[Alt]+[Enter]		Refait la dernière action.
[Shift]+[Del] [Ctrl]+[X]	ou	Coupe l'élément sélectionné.
[Ctrl]+[Ins] [Ctrl]+[C]	ou	Copie l'élément sélectionné.
[Shift]+[Ins] [Ctrl]+[V]	ou	Colle l'élément sélectionné.
[Del]		Efface la sélection.
[F9]		Arrangement des contrôles sélectionnés entre eux.
[Shift]+[F7]		Verrouille ou déverrouille la fenêtre courante, sans toutefois modifier le blocage des contrôles appartenant à la fenêtre
[Ctrl]+[F7]		Verrouille ou déverrouille la totalité des contrôles de la fenêtre courante, sans toutefois modifier le blocage de la fenêtre même.
[F7]		Permet de basculer entre le verrouillage et le déverrouillage des éléments sélectionnés.

Raccourcis d'éléments du menu View

Raccourci clavier	Description
[Ctrl]+[Alt]+L	Ouverture de la vue Libraries.
[Ctrl]+[Alt]+W	Ouverture de la vue Opened Windows qui liste les fenêtres ouvertes dans le browser ressources.
[Alt]+[F8]	Ouverture de la fenêtre Targets Info qui liste les targets de la fenêtre courante.
[Shift]+[Enter]	Ouvre l'éditeur de code.
[Shift]+[Enter]	Ouvre l'éditeur de code.
[Enter]	Edition d'une ressource.
[Del]	Suppression d'une ressource.
[Ins]	Insertion d'une ressource. Ouverture de la boîte Add windows resources to project.



[Shift]+[Enter]	Création d'une ressource. Ouverture de la boîte Create new private window.
[Ctrl]+G	Ouverture de la fenêtre de Grep.



Raccourcis d'éléments du menu Build

Raccourci clavier	Description
[Ctrl]+[Shift]+B	Génération des fichiers source.
[F5]	Lance le projet NS-DK.

Actions de copier/coller

Raccourci clavier	Description
[Alt]+[Backspace] ou [Ctrl]+[Shift]+[Z]	Annuler une action
[Alt]+[Enter] ou [Ctrl]+[Shift]+[A]	Refaire une action.
[Ctrl]+[X] ou [Shift]+Del	Couper
[Ctrl]+[C] ou [Ctrl]+[Ins]	Copier
[Ctrl]+[V] ou [Shift]+[Ins]	Coller
[Ctrl]+[Shift]+[Del]	Couper en ajout à la fin du texte déjà présent dans le presse-papiers
[Ctrl]+[Shift]+[Ins]	Copier en ajout à la fin du texte déjà présent dans le presse-papiers
[Ctrl]+[Shift]+[M]	Pour passer du mode où un Coller déplace le curseur au mode où le curseur reste au début du texte collé

Autres manipulations dans l'éditeur de code

Raccourci clavier	Description
[F8]	Vérification du code
[Ctrl]+[F]	Ouverture de la boîte de recherche
[Ctrl]+[Shift]+[R] ou [Alt]+[Shift]+[F9]	Ouverture de la boîte de remplacement
[F9]	Recherche de l'occurrence suivante
[Shift]+[F9]	Recherche de l'occurrence suivante précédente
[Ctrl]+[F9] et [Ctrl]+[Shift]+[F9]	Recherche du texte sélectionné en bas ([Ctrl]+[F9]) ou en haut ([Ctrl]+[Shift]+[F9]) du code.
[Ctrl]+[Shift]+[G]	Ouverture de la boîte Goto
[Tab]/[Shift]+[Tab]	Indentation avant/arrière de la sélection
[Alt]+[Curseur] et [Alt]+[Shift]+flèches	Sélection rectangulaire

[Ctrl]+[Shift]+[U]	Conversion en majuscule
[Ctrl]+[Shift]+[L]	Conversion en minuscule
[Ctrl]+[F12]	Ajouter/retirer des signets sur les lignes
[F12]	Signet suivant
[Shift]+[F12]	Signet précédent
[Alt]+[F12]	Premier signet
[Alt]+[Shift]+[F12]	Dernier signet
[Ctrl]+[Shift]+[F12]	Effacer tous les signets
[Ctrl]+[Enter]	Couper une ligne sans indentation
[Ctrl]+[ArrowUp] [Ctrl]+[ArrowDown]	et Défilement du texte en gardant le curseur dans la même ligne
[Ctrl]+[Shift]+[Enter]	Recherche des occurrences de type [***]
[Ctrl]+[Espace]	Auto-complétion des verbes NCL.
[Ctrl]+[Shift]+[Espace]	Extension des macros textes
[Ctrl]+[PageUp] [Ctrl]+[PageDown]	et Navigation entre les éléments d'un même bloc de code NCL
[Shift]+[F1]	Recherche l'élément sélectionné dans les librairies NCL et affiche sa syntaxe.
[Ctrl]+[F1]	Affiche l'aide en ligne correspondant à l'élément sélectionné.
[Shift]+[F5]	Atteindre le code d'une instruction ou fonction utilisateur

Nouvelle barre d'outils (NS-DK 5.00)

NS-DK 5.00 propose une nouvelle barre d'outils sans onglets.

Barres d'outils

Modification de certaines boîtes de dialogues (NS-DK 5.00)

Les boîtes de dialogue permettant la définition des ressources suivantes ont été modifiées :

- Constant,
- Segment,
- Variable.

Les boîtes Edit Segment et Edit Variable contiennent maintenant un champ Type permettant de définir le type du segment ou de la variable. Par ailleurs, les champs des segments/variables peuvent être déclarés par référence (champ Reference) pour une utilisation par pointeurs typés.

Edit segment RGBCOLOR

Segment

Name:

Description:

Fields details

Name:

Type: Size:

Dimensions: ☐ Reference

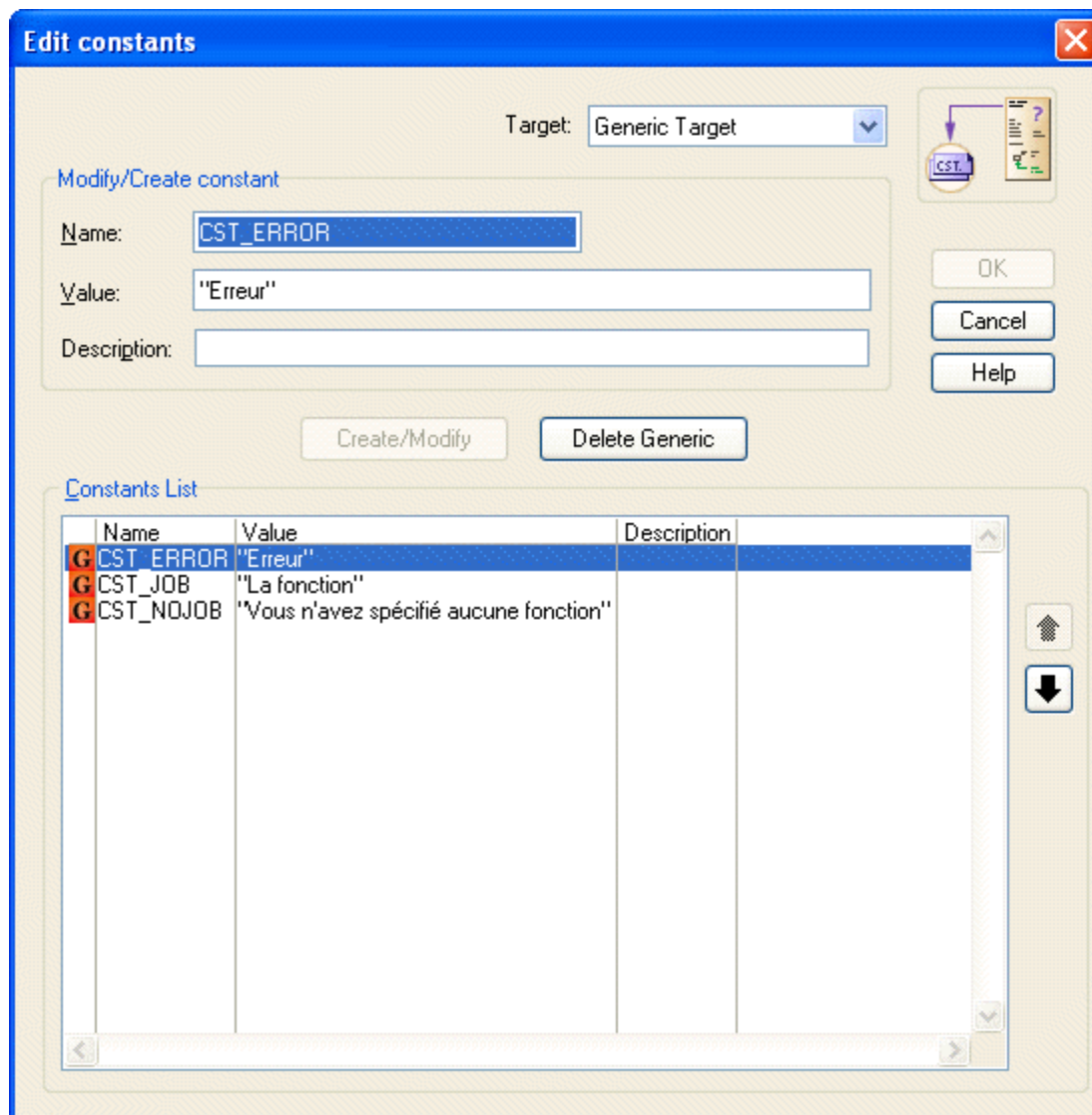
Description:

Fields List

Type	Name	Size	Dims	Description
INT	RED%	(1)[128]		
INT	BLUE%	(1)[20]		
INT	GREEN%	(1)		

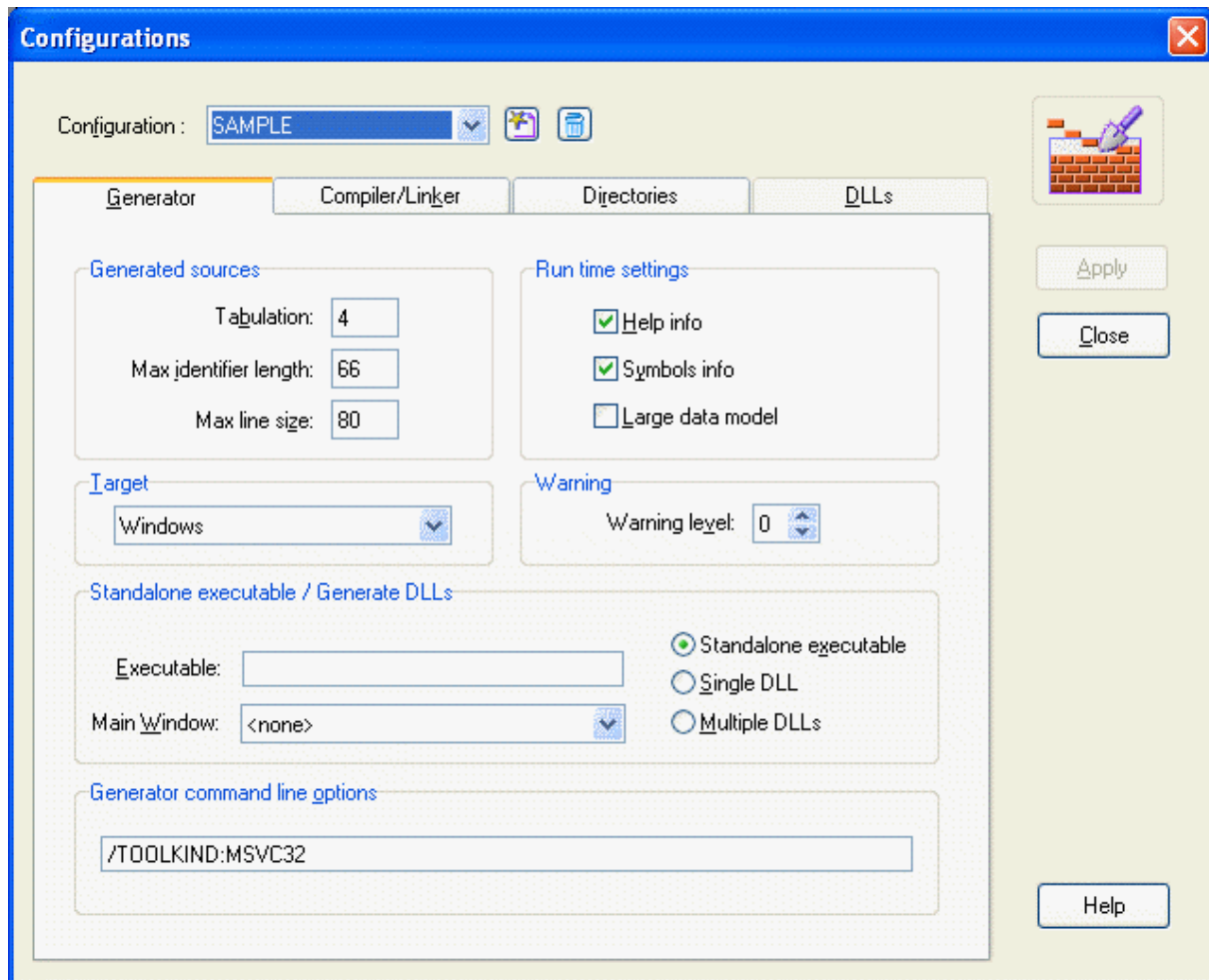
Buttons: OK, Cancel, Indexes..., Help, Append, Insert, Modify, Delete, Up, Down

La boîte Edit Constants a également été modifiée et permet notamment de sélectionner les constantes en fonction de la target.



Les boîtes de dialogues permettant la configuration de l'outil et des projets sont également revues. Ainsi, la boîte Configurations reprend les éléments de l'ancienne boîte de dialogue Project generation (obtenue par le menu Projects/Generation options setup) et ses boîtes associées : Setup generation, Setup compiler, Setup Linker, Setup executable, Setup DLLs, Setup Directories, Setup targets.

La boîte Configurations, appelée depuis le menu Options/Configurations ..., vous permet donc de créer ou sélectionner un fichier de configuration puis de définir pour ce fichier la configuration générale de la construction (onglets Generator, Directories et DLLs) et la configuration des différents outils (onglets Generator et Compiler/Linker). Pour plus d'informations, voir [Paramétrage d'un projet](#) et [Génération, compilation et édition de liens](#)



Paramétrage d'un projet (NS-DK 5.00)

La boîte Configurations, appelée par le menu Options/Configurations ... de NS Design, permet de spécifier plusieurs fichiers de configuration de génération pour un même projet. Un fichier de configuration a pour extension .N_C et contient la configuration choisie pour chacun des outils utilisés pour la construction d'un exécutable (générateur, compilateur, éditeur de liens, compilateur de ressources, etc.), ceux ci étant paramétrables à partir de cette boîte.

Génération, compilation et édition de liens (NS-DK 5.00)

La boîte Build/Run project est obtenue par sélection de l'option Build/Run configuration du menu Build.

Cette boîte de dialogue vous permet de lancer NS-Gen. Celui-ci utilise l'ensemble des renseignements que vous avez fournis au préalable dans la boîte Configurations.

Elle est organisée en plusieurs parties :

- la partie supérieure permet de sélectionner le fichier de configuration à utiliser,
- la deuxième partie contient les différentes opérations que NS-Gen peut effectuer en séquence.

Suppression des limitations historiques

Suppression des limitations historiques ([NS-DK 5.00](#))

Dans les versions précédentes, le fichier de description des projets (.PRJ) était au format binaire afin d'être compatible sur les plates-formes Windows et OS/2 16 bits. Dans NS-DK 5.00, le fichier .PRJ est remplacé par le fichier .XNP au format XML.

[Fichier XNP](#)

[Taille du nom des ressources et nouvelles ressources](#)

[Limitations de 64 Ko](#)

Fichier XNP ([NS-DK 5.00](#))

Les projets NS-DK des versions précédentes (fichier PRJ) sont repris complètement afin de passer au format XML (fichier XNP). L'abandon du format binaire actuel va permettre de dépasser un certain nombre de limites historiques et d'ajouter de nouveaux types de ressources.

Le format XNP contient les fichiers de projet PRJ et de configuration N_C. Le format XML du fichier XNP permet de stocker plusieurs fichiers N_C.

Un fichier N_C contient les paramètres de génération tant pour les fichiers source générés par NS-Gen que pour le compilateur et éditeur de lien à utiliser pour obtenir un exécutable.

Contrairement aux fichiers PRJ qui sont des fichiers binaires, les fichiers XNP sont des fichiers texte et donc peuvent être lus avec un éditeur de texte.

Taille du nom des ressources et nouvelles ressources ([NS-DK 5.00](#))

Le passage au format XML des fichiers de projet permet :

- Le passage à 220 caractères du répertoire d'installation du projet
- Le passage à 26 caractères du nom des SCR, NCL et images
- La possibilité de gérer les ressources GIF et JPEG en plus des BMP

Limitations de 64 Ko ([NS-DK 5.00](#))

La limite de 64 Ko pour les ressources de types NCL (bibliothèques) et SCR (écran) sont supprimées.

Evolution du langage NCL

Evolution du langage NCL ([NS-DK 5.00](#))

NS-DK 5.00 propose de nouvelles API et de nouveaux types permettant de faire évoluer le langage NCL.

[Le type DynStr](#)

[Extension de la fonction POS%](#)

[Extension de l'opérateur STRING](#)
[Extension de la fonction INSERT\\$](#)
[Extension de l'instruction NEW](#)
[Liste Chaînée](#)
[API de Trace](#)
[Modification des services NCL](#)
[Passage de paramètres à un événement](#)
[Evolution de NSWIN](#)

Le type DynStr (NS-DK 5.00)

Les principales fonctionnalités de ce nouveau type sont listées ci-dessous :

- La taille d'une variable de type DynStr n'est pas déterminée. Ceci constitue la principale différence entre une CSTRING et une DynStr.
- Il est désormais possible de manipuler des chaînes d'une taille supérieure à 65 000 caractères de manière transparente.
- L'utilisation du type DynStr est quasi-identique à celle de CSTRING. La majorité des fonctions et instructions manipulant des CSTRING peuvent ainsi également utiliser des DynStr. Les fonctions COPY\$, DELETE\$, INSERT\$, POS%, SKIP, LSKIP, RSKIP, LENGTH, ASC%... et les opérateurs & et && utilisent directement le type natif DynStr sans aucune conversion préalable en CSTRING.
- Un type DynStr sera converti automatiquement en CSTRING en cas d'affectation vers une variable de ce type et vice-versa. Exemple :

```

Local CSTRING A$(2000)
Local DYNSTR B
A$="Test"
B=A$

```

- Pour faciliter les insertions et les mises à jour de données dans les bases Oracle, NS-DK effectue de façon transparente des conversions implicites du type DynStr vers le type CLOB (Character Large Object).
- Enfin, le type DynStr facilite la communication avec les ActiveX ou les Services Web.
- Il est impératif d'utiliser l'instruction NEW pour allouer dynamiquement un segment contenant une variable ou plus de type DYNSTR. L'instruction NEW ajoute dans l'en-tête du bloc mémoire retourné un pointeur permettant à l'instruction NCL standard DISPOSE de libérer la ou les DYNSTR du segment ou du tableau de segment.
- Il est possible d'utiliser une variable de type DYNSTR comme variable hôte dans une requête SQL (SQL_EXEC, SQL_EXECSTR, SQL_EXECCSTR). Cependant, il est totalement exclu d'utiliser des variables de type segment contenant un ou plusieurs champs de type

DYNSTR (même indirectement via un champ de type segment), puisque le segment est échangé comme un bloc binaire avec la base de données, ce qui provoquerait toutes sortes de bug en cas de présence de DYNSTR.

Pour plus d'informations, reportez-vous au manuel "Langage NCL".

Extension de la fonction POS% (NS-DK 5.00)

La fonction POS% évolue pour permettre de retourner la position d'une occurrence de chaîne de caractères dans une autre chaîne de caractères à partir d'une position précise.

Extension de l'opérateur STRING (NS-DK 5.00)

L'opérateur STRING intègre deux nouveaux paramètres entier et base permettant de convertir un nombre décimal en une autre base (2, 8 ou 16).

Extension de la fonction INSERT\$ (NS-DK 5.00)

La fonction INSERT\$ intègre un nouveau paramètre optionnel nb-car-chaîne-dest-suppr permet de définir le nombre de caractères à supprimer dans la chaîne destination lors de l'insertion. Cette option permet essentiellement de remplacer un certain nombre de caractères dans une chaîne.

Extension de l'instruction NEW (NS-DK 5.00)

L'instruction NEW introduit une nouvelle syntaxe permettant d'allouer des tableaux sans taille prédéfinie au préalable.

Liste Chaînée (NS-DK 5.00)

Ce nouveau type permet de construire dynamiquement une liste chaînée de valeur simple ou de segments NCL.

L'allocation d'espace est faite automatiquement en fonction du besoin.

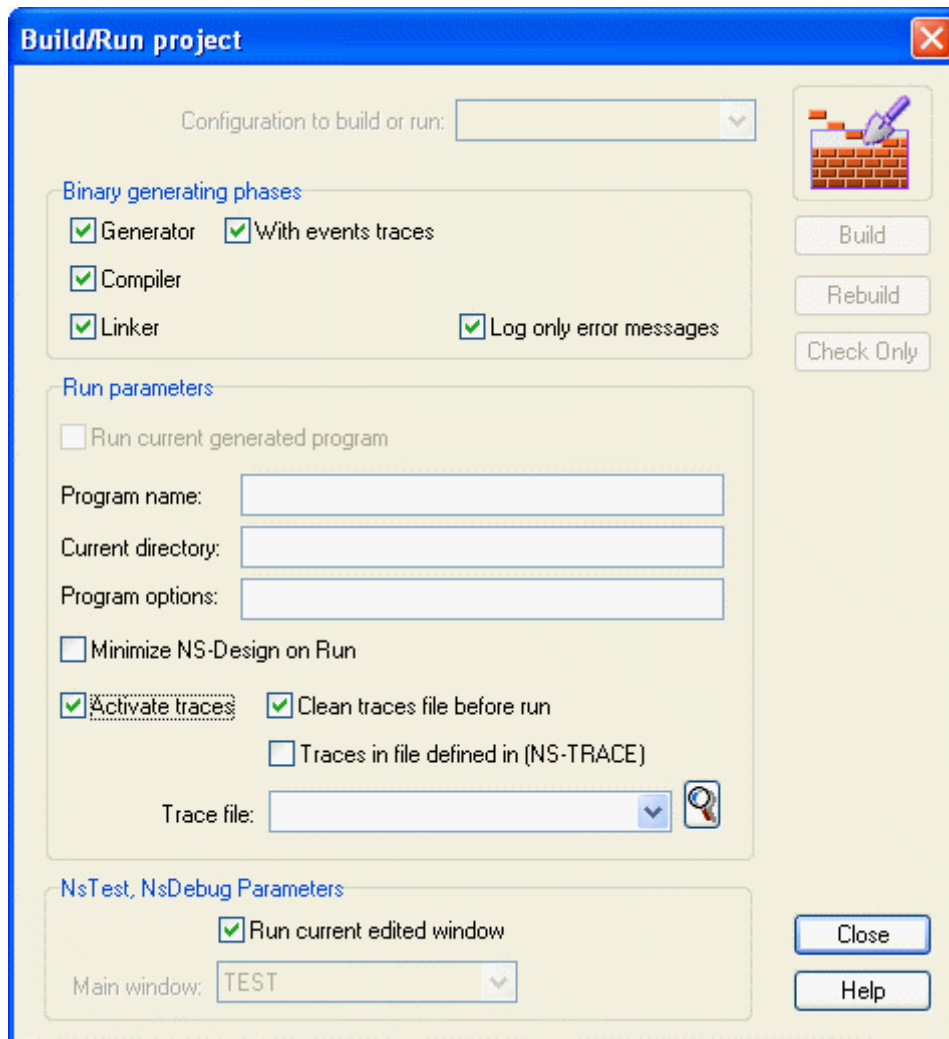
Un élément peut être accédé directement.

API de Trace (NS-DK 5.00)

Il est désormais possible d'ajouter simplement des traces qui seront activées ou désactivées en exploitation.

A l'exécution, NS-DK crée un fichier trace dans lequel une trace peut être créée à l'entrée et à la sortie d'un événement, d'une fonction ou d'une instruction.

Pour générer ces traces, cocher la case With event traces dans la boîte de dialogue Build/Run project (obtenue par le menu Build/Run Configuration). Les traces sont générées dans un fichier texte.



Cocher la case **Activate traces**, pour obtenir les traces à l'exécution du binaire, soit dans le fichier désigné par la variable d'environnement `NS-TRACE`, soit dans le fichier sélectionné dans **Trace file**.

L'utilisateur peut ajouter ces propres traces dans ses programmes NCL par l'appel de l'instruction `NS_TRACE`.

L'instruction `SetNSGenTrace` permet de modifier le formatage des traces.

Pour plus d'informations, reportez-vous à la librairie NSMISC à la rubrique Gestion des traces.

Modification des services NCL (NS-DK 5.00)

Certains services NCL de NS-DK ont des paramètres définis en `INT(4)` alors qu'il s'agit de `POINTER`.

NS-DK 5.00 corrige cette anomalie afin de faciliter les évolutions futures de votre architecture.

Le comportement reste inchangé pour vos applications Windows 32 bits.

Pour plus d'informations, reportez-vous au manuel "Langage NCL".

Passage de paramètres à un événement (NS-DK 5.00)

Le passage à un événement d'un paramètre dont la taille était supérieure à 2 octets nécessitait de le décomposer au préalable en deux mots de 2 octets chacun grâce aux instructions Hiw et Low.

Avec la version 5.00, cette décomposition préalable ne sera plus nécessaire il sera possible d'utiliser la syntaxe :

```
SEND EXECUTED, POINTER TO NOMCONTROL
```

Au lieu de :

```
SEND EXECUTED, Low POINTER, Hiw POINTER TO NOMCONTROL
```

Ce changement de syntaxe est également valable pour les instructions POST, OPEN, CALL...

|| L'ancien mode reste supporté, mais ne fonctionne pas sur une architecture 64 bits.

Pour plus d'informations, reportez-vous à la rubrique "Instruction SEND" du manuel "Langage NCL".

Evolution de NSWIN

Evolution de NSWIN (NS-DK 5.00)

La librairie NSWIN.NCL est complétée par des fonctionnalités propres aux environnements Windows. Les fonctionnalités suivantes sont ajoutées :

- [Gestion de la Registry Windows,](#)
- [Gestion de la position du pointeur souris.](#)

Pour plus d'informations, reportez-vous au chapitre "Librairie NSWIN Windows" du manuel Référence des Librairies de Services

Lecture/écriture dans la base de Registre Windows (NS-DK 5.00)

Le système Windows sauvegarde les données correspondant à sa configuration dans une base de données appelée le Registre.

Le Registre conserve les profils de chaque utilisateur de l'ordinateur ainsi que les données correspondant au matériel du système, aux programmes installés et aux paramètres des propriétés.

L'éditeur de Registre fourni avec Windows se nomme regedit.exe.

L'API du Registre du service NS-WIN permet de contrôler et de modifier le Registre Windows. Cependant, ces fonctions s'adressent exclusivement à des utilisateurs expérimentés. En effet, il n'est pas recommandé de modifier la base de Registre Windows. Nat System vous préconise de laisser les programmes Windows modifier eux-mêmes le Registre lorsque cela est indispensable.

L'utilisation de ces fonctions est particulièrement sensible. En effet, une modification incorrecte de la base de Registre Windows peut endommager gravement votre système. Nat System vous recommande donc avant toute modification d'enregistrer vos données critiques. Si malgré tout, votre système est endommagé la seule solution reste souvent sa réinstallation. Pour plus d'informations sur la base de registre et les éventuels problèmes occasionnés par sa modification, reportez-vous au centre d'aide et de support de Windows.

Pour plus d'informations, reportez-vous au chapitre "Librairie NSWIN Windows" du manuel Référence des Librairies de Services

Gestion de la position du pointeur souris (NS-DK 5.00)

NS-DK intègre deux nouvelles instructions GETCURSORPOS et SETCURSORPOS permettant respectivement de récupérer la position courante du curseur et positionner le curseur à la position demandée.

Pour plus d'informations, reportez-vous au chapitre "Librairie NSWIN Windows" du manuel Référence des Librairies de Services

Contrôles graphiques et affichage

Contrôles graphiques et affichage (NS-DK 5.00)

NS-DK 5.00 intègre de nombreuses nouveautés sur les contrôles graphiques et l'affichage de Nat System va répondre à plusieurs préoccupations clients au niveau des applications NS-DK produites.

- Affichage d'une application NS-DK lors des changements de résolutions des écrans avec [l'ancrage des contrôles](#).
- Module [NS-ScreenExploser](#) permettant de modifier la taille des caractères des contrôles.
 - [Contrôle Liste](#)
 - [Contrôle Menu Externe](#)
 - [Custom Control IE](#)
 - [Template étendu et Custom Control](#)

Ancrage des contrôles

Ancrage des contrôles (NS-DK 5.00)

L'ancrage des contrôles leur permet, lors du redimensionnement de la fenêtre, de s'adapter à la nouvelle taille de la fenêtre et de prendre la position et la taille appropriée.

Par exemple, si un contrôle Push Button est ancré aux bords gauche et supérieur de la fenêtre, il est positionné de façon à toujours rester à la même distance des bords gauche et supérieur de la fenêtre.

|| L'option Size Redraw dans le volet de propriétés de la fenêtre doit être activée pour repositionner / retailer les contrôles à chaque fois que l'utilisateur modifie la taille de la fenêtre.

Pour que l'ancrage fonctionne, la boîte de dialogue ne doit pas posséder de barre de défilement.

[Principe](#)

[Méthodes d'ancrage](#)

[Exemples d'utilisation](#)

[Utilisation](#)

Principe (NS-DK 5.00)

Chaque contrôle possède 4 côtés qui peuvent être ancrés de 3 manières sur son bord de fenêtre (total = $3^4 = 81$ possibilités d'ancrage par contrôle)

- L'ancrage fixe permet de garder constante la distance du contrôle avec le bord de la fenêtre. Si la fenêtre est redimensionnée, la distance entre le contrôle et le bord de la fenêtre reste identique.
- L'ancrage en proportion permet de conserver constant la proportion de la distance du contrôle avec le bord de la fenêtre par rapport à la taille de la fenêtre. Si la fenêtre est redimensionnée, la distance entre le côté du contrôle et le bord est toujours proportionnelle à la taille de la fenêtre.
- Sans ancrage. Le côté du contrôle n'est pas attaché au bord de la fenêtre.

Si un contrôle n'est pas ancré sur les deux cotés opposés, en cas de redimensionnement de la fenêtre, la position de son centre reste fixe en proportion par rapport aux bords de la fenêtre.

Certains contrôles ou attributs de contrôles ne peuvent pas être retaillés : hauteur des Entry-Field, Static-Text, Radio boutons, largeur des contrôles en mode auto-size ...

Méthodes d'ancrage (NS-DK 5.00)

On peut ancrer les contrôles de deux manières :

- Via le paramètre Anchor du volet de propriétés des contrôles (qui permet de choisir entre les neuf types d'ancrage les plus courants).

Type d'ancrage	Description
NS_AS_BOTTOM_LEFT	Ancrage en bas et à gauche (valeur par défaut)
NS_AS_TOP_LEFT	Ancrage en haut et à gauche
NS_AS_TOP_RIGHT	Ancrage en haut et à droite
NS_AS_BOTTOM_RIGHT	Ancrage en bas et à droite
NS_AS_WIDTH_BOTTOM	Ancrage à gauche, à droite et en bas
NS_AS_WIDTH_TOP	Ancrage à gauche, à droite et en haut
NS_AS_HEIGHT_LEFT	Ancrage en haut, en bas et à gauche
NS_AS_HEIGHT_RIGHT	Ancrage en haut, en bas et à droite
NS_AS_WIDTH_HEIGHT	Ancrage en haut, en bas, à gauche et à droite

- Via le paramètre dynamique .ANCHOR (en écriture seule) avec les constantes NS_AS_* positionnées dans la librairie NSMisc, qui permet de choisir entre toutes les possibilités d'ancrage.

Exemples d'utilisation (NS-DK 5.00)

Le contrôle est ancré en bas à gauche, sa taille est fixe

```
CTRL.ANCHOR=NS_AS_BOTTOM_LEFT
```

Le contrôle est ancré en haut à droite, sa taille est fixe

```
CTRL.ANCHOR=NS_AS_TOP_RIGHT
```

Le contrôle est ancré à gauche, à droite et en bas, sa largeur change de la même façon que la celle de la fenêtre

```
CTRL.ANCHOR=NS_AS_WITH_BOTTOM
```

Le contrôle est ancré à gauche, à droite, en haut et en bas, ses largeurs/hauteurs changent de la même façon que la celle de la fenêtre. On applique ce mode en général au list-box

```
CTRL.ANCHOR=NS_AS_WITH_HEIGHT
```

Le contrôle est ancré en bas et à gauche en fixe et à droite en proportion, la position de son bord droit est proportionnelle à la taille de la fenêtre

```
CTRL.ANCHOR=NS_AS_LEFT & NS_AS_RIGHT & NS_AS_BOTTOM
```

Le contrôle est ancré en bas en fixe, à gauche et à droite en proportion, les positions de ses bords droit et gauche ainsi que sa largeur sont proportionnelles à la largeur de la fenêtre.

```
CTRL.ANCHOR=NS_AS_PLEFT & NS_AS_PRIGHT & NS_AS_BOTTOM
```

Attention, cas spécial : Ce contrôle est ancré en bas mais ni à droite ni à gauche. Dans ce cas la largeur du contrôle est fixe et son centre occupe toujours la même position en proportion par rapport aux bords gauches et droits de la fenêtre.

```
CTRL.ANCHOR=NS_AS_BOTTOM
```

Utilisation (NS-DK 5.00)

L'ancrage peut être utilisé principalement pour résoudre deux problèmes.

1. Faire des applications retailables et repositionnables.
2. Faire des applications qui s'adaptent automatiquement à la résolution de l'écran.

Préconisations de développement (NS-DK 5.00)

Exemple : Pour écrire une application qui s'adapte automatiquement à la résolution (800/600 ou 1024/768) :

- Le développeur dessine ses fenêtres en 800/600 et modifie l'ancrage pour les contrôles (par le volet de propriétés ou bien par le qualificateur dynamique ANCHOR),
- Il MAXIMISE sa fenêtre dans l'événement INIT ou par design. Si l'ancrage est bien codé, l'aspect de la fenêtre sera toujours correct et

la quantité d'information affichée ne dépend que de la résolution d'écran.

Modifications dynamiques (NS-DK 5.00)

A n'importe quel endroit de l'application, on peut changer dynamiquement les positions/tailles des contrôles ainsi que l'ancrage, la mise à jour est automatique.

Voir aussi (NS-DK 5.00)

On peut associer l'ancrage à l'instruction `SetAttachedChildFlags` pour faire des fenêtres aux fonctionnalités et à l'aspect moderne.

Remarques (NS-DK 5.00)

Quand la fenêtre devient petite et pour éviter le chevauchement des contrôles, il peut être utile de forcer la dimension minimale d'une fenêtre à l'aide de l'événement `AdjustSizeOrPos`.

Cet événement est généré lorsque l'utilisateur désire modifier la position ou la taille d'une fenêtre. Il permet d'intercepter cette demande et de changer par programme les valeurs des nouvelles positions et tailles.

Dans le cas où on utilise l'ancrage dans une boîte de dialogue retaillable et que certains de ses contrôles se repeignent mal (par exemple des custom controls), on peut corriger le problème en mettant le flag `CLIENT.CLIPCHILDREN` à `FALSE%`.

```
TANCRA4(self%).Client.CLIPCHILDREN=FALSE%
```

Ce paramètre permet essentiellement de résoudre les problèmes de compatibilité avec les versions précédentes. Par défaut en version 5, ce flag est positionné à vrai automatiquement pour les boîtes de dialogue retaillables avec l'option `Size Redraw` cochée.

NS-ScreenExploser (NS-DK 5.00)

Ce module permet par la configuration du fichier `NSLIB.INI` de changer la taille des caractères des différents contrôles NS-DK : les contrôles comme la `CheckBox` ou le `RadioButton` qui ne changent pas de taille avec l'ancrage sont impactés par `ScreenExploser`.

Contrôle Liste (NS-DK 5.00)

NS-DK intègre une version évoluée du contrôle `SheetBox` qui a pour objectif de remplacer la `ListBox` partout où celle-ci est limitée. Par exemple, la `SheetBox` ne connaît pas la limite des 255 caractères.

Ce contrôle évolue avec la version 5.00. Il devient possible de définir par le volet de propriétés :

- Les couleurs du titre

- La couleur des lignes paires et impaires
- La couleur des colonnes paires et impaires
- De figer une ou plusieurs colonnes à gauches
- De définir pour chaque colonne un format de saisie en ligne

Il est de plus possible par programmation de définir :

- La couleur d'une cellule
- La possibilité pour un utilisateur de déplacer par Drag & Drop les colonnes
- La possibilité de trier une colonne par clic sur le titre de la colonne

Pour plus d'informations, reportez-vous au chapitre "Librairie SheetBox NSSHTREE" du manuel "Librairies de Services".

Contrôle Menu Externe (NS-DK 5.00)

NS-DK 5.00 propose le menu externe comme nouvelle ressource.

Les avantages de la gestion dynamique de ressources Menu sont les suivants :

- vous pouvez associer une même définition de barre de menus à plusieurs fenêtres différentes : une ressource Menu est réutilisable, car elle est indépendante d'une fenêtre particulière,
- sans programmation spécifique, vous pouvez disposer d'une barre d'outils synchrone avec les items de menus, c'est-à-dire qu'elle affiche des boutons qui correspondent toujours aux items de menu et déclenchent les mêmes traitements,
- vous pouvez surcharger une barre de menus d'une fenêtre fille à partir de celui de sa fenêtre mère sans dupliquer les items de menu de celle-ci, et donc gérer du MDI (Multiple Document Interface) de façon transparente.

Il ne faut pas confondre cette gestion de menus et la conception de contrôles Menu. Ceux-ci permettent en effet de concevoir des menus de façon complètement statique : les contrôles Menu font partie intégrante de la définition de la fenêtre dans laquelle ils ont été conçus. D'autre part dans ce cas, si vous désirez une barre d'outils associée, vous devez la programmer entièrement.

Custom Control IE (NS-DK 5.00)

NS-DK intègre un nouveau composant Custom Control qui s'appuie sur l'OCX WebBrowser fourni par Microsoft. Il vous permet d'intégrer facilement un navigateur Internet Explorer dans une fenêtre NS-DK.

Ce composant intègre différentes fonctionnalités :

- Passage de paramètres en mode POST ou GET (avec formatage automatique des paramètres en fonction du mode),
- Affichage ou non d'une barre de statut,
- Blocage possible des POPUP,
- Blocage possible du CTRL+N,

- Possibilité de filtrer les pages affichées,
- Analyse du D.O.M. de chaque page,
- Modifications des objets du D.O.M.

Pour plus d'informations sur ce composant et la librairie NSCUSTIE associée, reportez-vous au chapitre Librairie NSCUSTIE

Template étendu et Custom Control (NS-DK 5.00)

Désormais, vous avez la possibilité d'afficher les propriétés d'un template étendu ou d'un Custom Control dans le volet de propriétés, et non plus dans une fenêtre modale, grâce au fichier de modélisation du contrôle au format XML. Ce fichier contient l'ensemble des informations nécessaires à l'affichage des propriétés du contrôle dans le volet des propriétés.

Par ailleurs, un segment de contexte facilite la gestion des données contextuelles et d'exécution dans la programmation du mode Exécution du template étendu ou du contrôle utilisateur. L'accès aux propriétés de ce segment de contexte s'effectue à travers des fonctions de lecture/écriture (`<Name>_Get_<PtyName>`, `<Name>_Set_<PtyName>`).

Pour plus d'informations sur ces nouvelles fonctionnalités, reportez-vous au chapitre 2 du "Guide de Développement".

Les fichiers NSDKCFG.INI et NSDKLOC.INI (NS-DK 5.00)

Le fichier NS-DK.INI, qui permettait de fournir un ensemble d'outils et de paramètres proposés par défaut dans les différentes boîtes de génération, est obsolète. Il est désormais remplacé par deux fichiers :

- NSDKCFG.INI qui correspond à la configuration globale de vos projets.

Le fichier NSDKCFG.INI est livré avec NS-DK. Il a pour vocation d'être partagé sur une installation de NS-DK en réseau par exemple. Le fichier NSDKCFG.INI est normalement modifié par un administrateur de l'installation NS-DK ou un chef de projet.

- NSDKLOC.INI qui correspond à la configuration locale d'un projet.

Le fichier NSDKLOC.INI n'est pas fourni avec NS-DK. C'est un fichier de configuration propre à chaque utilisateur, qui le crée et le modifie.

Nat System vous suggère de faire pointer la variable d'environnement NS-INI sur un répertoire local de l'utilisateur, avec une installation NS-DK éventuellement réseau.

Le fichier NSDKCFG.INI est trouvé automatiquement dans l'installation NS-DK, tandis que le fichier NSDKLOC.INI est créé dans le répertoire local pointé par NS-INI.

Le fichier NSDKCFG.INI est recherché prioritairement dans les répertoires suivants :

1. dans le répertoire pointé par la variable d'environnement NS-INI (donc sans le nom du projet),
2. S'il n'existe pas dans le répertoire pointé par la variable d'environnement NS-INI, il est recherché dans le répertoire /ini de l'installation.
3. S'il n'existe pas dans le répertoire /ini, il est créé dans le répertoire pointé par NS-INI.
4. Si le répertoire pointé par la variable d'environnement NS-INI n'existe pas ou si NS-INI n'est pas défini, il est créé dans le répertoire /ini de l'installation.

Le fichier NSDKLOC.INI est recherché prioritairement dans le répertoire pointé par NS-INI :

5. S'il n'existe pas dans le répertoire pointé par la variable d'environnement NS-INI, il est créé dans le répertoire pointé par NS-INI.
6. Si le répertoire pointé par NS-INI n'existe pas ou si NS-INI n'est pas défini, il est recherché dans le répertoire INI de l'installation.
7. S'il n'existe pas dans le répertoire INI de l'installation, il est créé dans le répertoire INI de l'installation.

Si l'installation de NS-DK est partagée en réseau, et si la variable NS-INI n'est pas définie pour plusieurs utilisateurs, il y a risque de collision sur le fichier NSDKLOC.INI qui devient partagé (aucune protection n'existe sur l'écriture simultanée par plusieurs intervenants). A l'inverse si l'installation de NS-DK est distribuée sur chaque poste, et si la variable NS-INI n'est pas définie, chaque utilisateur va créer un fichier NSDKLOC.INI dans le répertoire INI de l'installation locale de NS-DK.

Génération et déboguage

Génération et débogage (NS-DK 5.00)

NS-DK propose différentes évolutions liées à la génération et au débogage des applications :

- [Limitation](#)
- [Générateur automatique de trace](#)
- L'outil [NS-DebugMem](#) qui permet de détecter les fuites mémoire
- [Optimisation du générateur pour supporter les cibles 64 bits.](#)
- [Taille des noms de DLLs](#)

Limitation des ressources externes (NS-DK 5.00)

Il n'est désormais plus possible d'avoir une configuration de génération permettant la génération de ressources externes.

Par conséquent, lors de la migration d'une application développée dans des versions précédente, le mode ressource externe est ignoré.

Générateur automatique de trace (NS-DK 5.00)

NS-DK 5.00 met à la disposition des développeurs une nouvelle option de génération qui permet de générer automatiquement des traces lors de l'appel d'une fonction/instruction et de sa sortie.

Cette fonctionnalité s'appuiera sur la nouvelle [API de trace](#). Les traces ne seront effectivement actives que lors du positionnement de la bonne variable d'environnement.

Le développeur pourra suivre le déroulement de son application à l'exécution indépendamment de l'utilisation ou non d'autres modules.

NS-DebugMem (NS-DK 5.00)

Cet utilitaire sera mis à disposition dans la version 5.00, il permet de détecter les fuites mémoires lors de l'exécution d'une application NS-DK.

Les fuites mémoires sont détectées indépendamment de leur impact sur l'exécution. Il devient possible de corriger simplement des problèmes de stabilité.

Optimisation pour le 64 bits ([NS-DK 5.00](#))

La génération est modifiée afin de devenir compatible avec les cibles 64 bits.

Certains mécanismes de transtypes automatiques pris en charge par le générateur ont ainsi été améliorés.

Nom des DLLs ([NS-DK 5.00](#))

La taille du nom des DLLs générées n'est plus limitée à 8 caractères.

Nouvelles fonctions et instructions

Nouvelles fonctions et instructions ([NS-DK 5.00](#))

Ce chapitre présente les nouvelles fonctions et instructions intégrées à NS-DK 5.00 pour optimiser le développement.

[Librairie NSWIN](#)

[Librairie NSMisc](#)

[Librairie NSCUSTIE](#)

Librairie NSWIN

[Librairie NSWIN \(\[NS-DK 5.00\]\(#\)\)](#)

Pour rappel, la librairie NSWIN permet d'obtenir des fonctionnalités uniques de Windows, comme l'énumération des fenêtres ouvertes, la manipulation de la base de Registre Windows ou encore l'envoi de messages.

Cette librairie s'adresse à des programmeurs ayant de très bonnes connaissances des mécanismes de gestion interne des fenêtres et des événements de Windows.

[Lecture/écriture dans la base de Registre Windows](#)

[Gestion de la position du pointeur souris](#)

[Attachement des fenêtres](#)

[Lecture/écriture dans la base de Registre Windows \(\[NS-DK 5.00\]\(#\)\)](#)

Le système Windows sauvegarde les données correspondant à sa configuration dans une base de données appelée le Registre.

Le Registre conserve les profils de chaque utilisateur de l'ordinateur ainsi que les données correspondant au matériel du système, aux programmes installés et aux paramètres des propriétés.

L'éditeur de Registre fourni avec Windows se nomme regedit.exe.

L'API du Registre du service NS-WIN permet de contrôler et de modifier le Registre Windows. Cependant, ces fonctions s'adressent exclusivement à des utilisateurs expérimentés. En effet, il n'est pas recommandé de modifier la base de Registre Windows. Nat System vous préconise de laisser les programmes Windows modifier eux-mêmes le Registre lorsque cela est indispensable.

L'utilisation de ces fonctions est particulièrement sensible. En effet, une modification incorrecte de la base de Registre Windows peut endommager gravement votre système. Nat System vous recommande donc avant toute modification d'enregistrer vos données critiques. Si malgré tout, votre système est endommagé la seule solution reste souvent sa réinstallation. Pour plus d'informations sur la base de registre et les éventuels problèmes occasionnés par sa modification, reportez-vous au centre d'aide et de support de Windows.

Fonction CREATEMODIFYREGISTRYKEY%

Fonction READREGISTRYKEY%

Fonction DELETEREGISTRYKEY%

Fonction DELETEREGISTRYVALUE%

Constantes NS_HKEY_**%

Constantes NS_REG_**%

Gestion de la position du pointeur souris (NS-DK 5.00)

Instruction GETCURSORPOS

Instruction SETCURSORPOS

Attachement des fenêtres (NS-DK 5.00)

Avec GetAttachedChildFlags% et SetAttachedChildFlags une fenêtre fille peut disposer de flags.

Les Constantes ACF_**% permettent de désactiver le menu de la fenêtre fille ou de la rendre redimensionnable.

Librairie NSMisc

Librairie NSMisc (NS-DK 5.00)

Pour rappel, la librairie NS MISC est une librairie annexe (Misc = Miscellaneous) contenant de nombreuses fonctions et instructions variées, comme la gestion de fichiers, gestion de disques (répertoires, place disponible, recherche de fichiers), génération de nombres aléatoires, conversions ASCII / EBCDIC, ...

[Gestion des traces](#)

[Gestion de l'ancrage des contrôles](#)

[Tri des tableaux](#)

Nouveautés NS-DK

Affectation d'images aux items de menu

Gestion des traces (NS-DK 5.00)

Instruction SetNSGenTrace

Instruction MyTrace

Instruction NS_TRACE

Segment SEG_NSAGENTTRACE

Constantes GTE_ *%

Gestion de l'ancrage des contrôles (NS-DK 5.00)

Constantes NS_AS_*

Tri des tableaux (NS-DK 5.00)

Instruction NSSORT

Affectation d'images aux items de menu (NS-DK 5.00)

Constantes SMIP_ *%

Librairie NSCUSTIE (NS-DK 5.00)

La librairie NSCUSTIE permet de manipuler le Custom Control IE, qui est un navigateur Internet Explorer intégrable dans une application NS-DK.

Index

A

Affectation d'images aux items de menu 81

Aide en ligne Nat System 35

Ancrage des contrôles 72

API de Trace 69

B

Base de Registre Windows 71

Bases de données supportées par NS-DK 5.00 50

Browser_de_ressources 51

C

Compatibilité avec Windows Vista 34

Contrôle Liste 75

Contrôle Menu Externe 76

Contrôles graphiques et affichage 72

Custom Control IE 76

E

Editeur NCL 56

Environnement technologique 49

Evolution de NSWIN 71

Exemples d'utilisation 74

Exemples V7 32

Extension de l'opérateur STRING 69

Extension de la fonction INSERT\$ 69

Extension de la fonction POS% 69

Extension de NEW 69

F

Fichier XNP 67

G

Génération compilation et édition de liens 66

Génération et débogage 78

Gestion de la position du pointeur souris 72

gestion des écrans multiple. 29

L

Le type DynStr 68

Le volet de propriétés 54

Les fichiers NSDKCFG.INI et NSDKLOC.INI 77

Limitation des ressources externes 78

M

Macros 58

Méthodes d'ancrage 73

Modification de certaines boîtes de dialogues 63

Modification des services NCL 70

Modifications dynamiques 75

N

Nouveautés de NS-DK 10.00 21

Nouveautés de NS-DK 5 SP2 34

Nouveautés de NS-DK 5.00 48

Nouveautés de NS-DK 6.00 33

Nouveautés de NS-DK 7.00 30

Nouveautés de NS-DK 8.00 27

Nouveaux menus 60

Nouveaux raccourcis clavier 60

Nouvel éditeur de fenêtre 56

Nouvelle barre d'outils 63

Nouvelle organisation du plan de travail 50

NS-ScreenExploser 75

P

Paramétrage d'un projet 66

Passage de paramètres à un événement 70

Préconisations de développement 74

Principe 73

R

Remarque 75

S

Suppression des limitations historiques 67

T

Taille du nom des ressources et nouvelles ressources 67

Template étendu et Custom Control 77

Tri des tableaux 81

U

Utilisation 74

V

Voir aussi 75

X

XNP 67
