

Net.Commerce pour NT, AIX et Solaris



Intégration

Version 3.2

Net.Commerce pour NT, AIX et Solaris



Intégration

Version 3.2

Remarque

Avant d'utiliser ce manuel et le produit associé, prenez connaissance des informations générales figurant à la section "Remarques".

LE PRESENT DOCUMENT EST LIVRE EN L'ETAT. IBM DECLINE TOUTE RESPONSABILITE, EXPRESSE OU IMPLICITE, RELATIVE AUX INFORMATIONS QUI Y SONT CONTENUES, Y COMPRIS EN CE QUI CONCERNE LES GARANTIES DE QUALITE MARCHANDE OU D'ADAPTATION A VOS BESOINS.

Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- <http://www.ibm.fr> (serveur IBM en France)
- <http://www.can.ibm.com> (serveur IBM au Canada)
- <http://www.ibm.com> (serveur IBM aux Etats-Unis)

© Copyright IBM France, 1999. Tous droits réservés.

© Copyright International Business Machines Corporation 1999. Tous droits réservés.

Table des matières

Intégration de SAP R/3 à Net.Commerce	1
Intégration de Net.Commerce à SAP R/3	1
Création d'un ID utilisateur Net.Commerce dans SAP R/3	2
Migration de données SAP vers Net.Commerce	3
Configuration des extensions de fonctions non prioritaires SAP	3
Configuration SAP R/3 requise pour l'exécution de Net.Commerce	6
Fichiers d'intégration SAP	6
Paramètres BAPI SAP R/3 utilisés par Net.Commerce	7
Test du modèle de code SAP	8
MQSeries et Net.Commerce	9
Utilisation de MQSeries pour l'intégration d'un système existant à	
Net.Commerce	10
Deux scénarios d'intégration de MQSeries	10
Installation de MQSeries avec Net.Commerce	11
Configuration d'une interface MQSeries client+serveur à serveur	11
Configuration d'une interface MQSeries de client autonome à serveur	12
Compilation du modèle de code d'intégration MQSeries	12
Fichiers modèle utilisés pour l'intégration avec MQSeries	13
Validation du canal MQSeries et test des connexions	16
Utilisation d'une fonction de substitution pour intégrer Net.Commerce à	
MQSeries	17
Configuration de Net.Commerce pour les messages MQSeries en entrée	18
MQCLIENT : Table du client MQSeries	20
MQXACT : Table des transactions MQSeries	20
MQXACTLOG : Table du journal des transactions MQSeries	21
Test de la fonctionnalité du flux de messages MQSeries	21
CICS et Net.Commerce	22
Intégration d'un système CICS avec Net.Commerce	24
Configuration matérielle et logicielle requise pour l'intégration CICS	24
Installation de l'environnement client CICS	24
Compilation du modèle de code d'intégration CICS	25
Fichiers modèle utilisés pour l'intégration CICS	26
Validation et test de la connexion	29
Utilisation d'une fonction de substitution pour intégrer Net.Commerce avec CICS	29
Configuration de Net.Commerce pour le trafic CICS en entrée	30
CCICSXACTLOG : Table du journal des transactions CICS	33
Test de la fonctionnalité du trafic en sortie vers CICS	33
IMS et Net.Commerce	34
Intégration d'un système IMS avec Net.Commerce	35
Configuration requise pour l'intégration à IMS	35
Installation de l'environnement client IMS	36
Compilation de modèle de code d'intégration IMS	36
Fichiers modèle utilisés pour l'intégration à IMS	37
Validation et test de la connexion IMS	39
Utilisation d'une fonction de substitution pour intégrer Net.Commerce avec IMS	39
IMSXACTLOG : Table du journal des transactions IMS	40
Adaptation du fichier de configuration Net.Commerce pour l'intégration d'IMS	41
Test de la fonctionnalité du trafic IMS	41
EDI et Net.Commerce	41
Utilisation d'EDI avec Net.Commerce	42
Obtention d'un compte IBM Global Services	43
Création et remplissage des tables de base de données EDI Net.Commerce	44
EDIOUT : Table des messages en sortie EDI	44

EDIADDR : Table des adresses EDI	45
EDILAYOUT : Table de présentation EDI	46
EDI_ENVELOP : Table des enveloppes EDI	46
Configuration de l'activité EXT_ORD_PROC pour le support EDI	47
Modification et installation du fichier ncei_servers.ini/ncei_servers.conf pour le support EDI	48
Remarques	51

Intégration de SAP R/3 à Net.Commerce

Si vous utilisez (SAP) R/3 version 3.1G pour le fonctionnement interne de votre entreprise, vous pouvez l'intégrer à Net.Commerce pour créer un World Wide Web prenant en charge les transactions mondiales en ligne. Net.Commerce automatise et organise le processus de vente sur Internet ; il offre les avantages ci-après.

1. Il permet de soulager le système SAP. Une copie du catalogue SAP est conservée dans Net.Commerce et mise à jour régulièrement, ce qui permet de libérer une grande partie de l'utilisation du système SAP.
2. Il protège le système SAP des accès directs tentés par les clients Internet. Le serveur Net.Commerce agit en tant que pare-feu entre les deux systèmes et protège le système SAP.
3. Les transactions du système SAP sont initialisées uniquement lorsque le client affiche ou envoie une commande. Ceci permet de minimiser le nombre de transactions traitées sur le système SAP.
4. La structure et la présentation du catalogue peut être améliorée grâce à Net.Commerce et offrir un résultat plus esthétique que le catalogue SAP R/3 à plat. Par exemple, grâce à Net.Commerce, vous pouvez utiliser des images ou des animations pour améliorer et organiser l'accès au catalogue SAP.

L'intégration de SAP à Net.Commerce suppose l'utilisation des éléments suivants :

- Catalogue : Le catalogue SAP est transféré dans Net.Commerce puis synchronisé normalement. Les informations sur les clients SAP sont insérées dans la base de données Net.Commerce et mises à jour régulièrement.
- Traitement des commandes : Net.Commerce interroge SAP pour toute information relative aux prix, aux remises, à la disponibilité, aux livraisons et aux taxes. Ceci, en temps réel, avant d'afficher l'information au client. La commande est directement créée dans le système SAP et déclenche les processus de tarification, de crédit, de livraison et de facturation de SAP.
- Etat : L'état d'une commande peut être mis à jour dans Net.Commerce, et permettre ainsi aux clients de suivre leur commande sur Internet.
- Intégration de SAP R/3 à Net.Commerce

Intégration de Net.Commerce à SAP R/3

Pour permettre l'intégration de Net.Commerce à votre système SAP (Systems, Applications and Products in Data Processing) R/3, exécutez les procédures suivantes :

1. Si vous souhaitez utiliser les modèles de données pour remplir la base de données Net.Commerce, installez IDES (International Demonstration and Education System) sur votre machine SAP R/3 version 3.1G.
2. Installez le kit d'outils RFC (Remote Function Call) sur la machine hébergeant le serveur Net.Commerce.
Ce kit fait partie du progiciel d'installation de SAP R/3.

Remarque : Si vous installez l'interface de SAP (SAPGUI), RFC fait partie de l'installation.

3. Vérifiez que les machines SAP et Net.Commerce peuvent communiquer via TCP/IP. Pour cela, vous devez soit installer SAPGUI sur le poste de travail, soit copier les entrées du fichier de services ci-après à partir d'un poste sur lequel l'interface SAPGUI est installée, vers le fichier de services :

 [WINDIR]\system32\drivers\etc\services

AIX **SUN** /etc/services

Vous pouvez ensuite tester la connexion à l'aide de l'interface de SAP (SAPGUI) ou de son utilitaire niping.

4. Sur le système SAP, créez un ID utilisateur Net.Commerce CPIC (Common Programming Interface - Communications).
Cet ID sera utilisé par Net.Commerce pour se connecter au système SAP.
5. Créez un utilitaire pour extraire les informations SAP (Material) dans un format pouvant être importé dans Net.Commerce.
6. A l'aide de l'utilitaire d'importation globale de Net.Commerce, importez les informations SAP dans les tables produits de la base de données Net.Commerce.
7. Ecrivez une routine pour importer les données clients SAP dans Net.Commerce.
Dans les exemples du kit SAP fournis avec Net.Commerce, la colonne SHLOGIN de la table SHOPPER est utilisée pour stocker le contenu du champ SAP **PartnNumb** (numéro de partenaire). Selon la façon dont votre base de données SAP a été personnalisée, vous pouvez être amené à créer, dans Net.Commerce, de nouvelles tables qui contiendront certaines informations relatives aux clients.
8. Vous pouvez également modifier la base de données Net.Commerce afin qu'elle prenne en compte les informations supplémentaires qu'il est possible d'extraire de la base de données SAP.
9. Ecrivez un programme SQL pour insérer les données clients SAP dans les tables clients de Net.Commerce.
10. Vous pouvez également modifier la bibliothèque SAPInteg.dll sur le système Net.Commerce pour changer la façon dont les paramètres BAPI SAP sont appelés par Net.Commerce.

Remarque : Il est également possible de tester le modèle de code fourni avec Net.Commerce tel que, avant de le modifier en vue de son utilisation dans un environnement de production.

Création d'un ID utilisateur Net.Commerce dans SAP R/3

Pour que Net.Commerce puisse communiquer avec SAP R/3, le système R/3 doit contenir un ID utilisateur Net.Commerce. C'est cet "utilisateur" qui se connecte à Net.Commerce et communique avec lui. Pour le créer, procédez comme suit :

1. Lancez l'interface SAPGUI et connectez-vous au système IDES (International Demonstration and Education System) R/3 V3.1G en tant que client 800.
2. Sélectionnez **Administration** dans le menu Tools.
3. Dans le menu Maintien User de l'écran d'administration, sélectionnez **Users**.
4. Dans la zone **User**, entrez un nouvel ID (par exemple, "commerce".)
5. Sélectionnez **Create** dans le menu Users.
6. Entrez un mot de passe initial et définissez la période de validité en indiquant des dates dans les zones **Valid from** et **Valid to**.
7. Dans la fenêtre Authorization Profiles, entrez **SAP_ALL** et **SAP_NEW**. (Dans un environnement de production, vous pouvez sélectionner le niveau de droits d'accès à accorder.)
8. Dans la fenêtre User Type, assurez-vous que la case **Dialog** est cochée.
9. Vous pouvez, si nécessaire, entrer des informations complémentaires dans les autres écrans et zones du programme.

10. Quittez l'interface SAPGUI et connectez-vous à nouveau en utilisant cette fois l'ID utilisateur que vous venez de créer. Vous serez invité à définir un nouveau mot de passe.

Remarque : Le nouvel ID utilisateur doit avoir accès aux fonctions CPIC (Common Programming Interface - Communications).

Migration de données SAP vers Net.Commerce

Lorsque vous intégrez Net.Commerce au système SAP R/3, la principale source de données subsiste sur ce dernier système et est importée vers le système Net.Commerce.

Les informations des catalogues, de même que celles relatives aux clients et au matériel, doivent être synchronisées (mises en accord) entre les deux systèmes. Les données des nouveaux clients Net.Commerce doivent être entrées directement dans le système SAP, puis importées dans le système Net.Commerce. Vous ne pouvez pas créer les utilisateurs dans Net.Commerce pour ensuite propager les données correspondantes vers le système SAP.

Pour que des données SAP puissent migrer vers la base de données Net.Commerce, vous devez modifier leur format afin de les rendre compatibles avec le système Net.Commerce. Si vous procédez régulièrement à la mise à jour de la base de données Net.Commerce à partir du catalogue SAP, vous devez créer un utilitaire qui assure la conversion des données, puis exécute l'utilitaire d'importation globale Net.Commerce pour insérer les données dans la base Net.Commerce.

Net.Commerce comprend un fichier appelé `bus2bus.in`, qui contient des modèles de données du système IDES (International Demonstration and Education System) SAP R/3 3.1G, dans un format qui peut être importé dans Net.Commerce à l'aide de l'utilitaire d'importation globale. Ce fichier se trouve dans le répertoire suivant :

 [rép_principal_Net.Commerce]\adt\samples\SAP\testing

  [rép_principal_Net.Commerce]/adt/samples/SAP/testing

Vous pouvez l'utiliser pour comparer les formats de données entre les deux systèmes. Vous pouvez également l'utiliser comme modèle de données pour créer un premier utilitaire préparant vos données de catalogue à l'importation dans Net.Commerce, et un second préparant vos données client à l'insertion dans la base de données Net.Commerce.

Dans ce fichier, ainsi que dans les DLL d'intégration SAP, le champ SAP Material est mis en correspondance avec la colonne PRNBR de la table PRODUCT Net.Commerce. Ce champ et cette colonne restent uniques sur l'ensemble du système. Dans les DLL, le champ SAP PartnNumb (numéro de partenaire) est également mis en correspondance avec la colonne SHLOGID de la table SHOPPER Net.Commerce.

Configuration des extensions de fonctions non prioritaires SAP

Le module Intégration SAP de Net.Commerce contient deux fichiers DLL compilés, quatre fichiers source C++ modifiables, sept fichiers écrits en C et 6 fichiers "include".

WIN OrderDisplay.dll

AIX libOrderDisplay.a

SUN libOrderDisplay.so

appelle la fonction BAPI SimulateOrder dans SAP.

Il s'agit d'une instruction non modifiable.

WIN SAPInteg.dll

AIX libSAPInteg.a

SUN libSAPInteg.so

affiche et crée la commande dans Net.Commerce.

Dans certains cas, cette DLL peut être utilisée telle quelle, mais comme les systèmes SAP R/3 sont pour la plupart personnalisés, vous devrez certainement améliorer ses fonctions.

Les quatre fichiers source, sim_ord_proc.cpp, ext_ord_proc.cpp, SAPInteg.cpp et tasks_api.cpp se trouvent dans le répertoire suivant :

WIN [rép_principal_Net.Commerce]\adt\samples\SAP\api

AIX **SUN** [rép_principal_Net.Commerce]/adt/samples/SAP/api

Vous pouvez les modifier afin de personnaliser la façon dont Net.Commerce appellera la fonction BAPI SAP pour simuler et créer les commandes, respectivement.

Les autres fichiers source, order_create.c, order-util.c, saputils.c, rfcutil.c, saputil.c, order-simulate.c et readini.c, ainsi que les fichiers d'en-tête, readini.h, rfcutil.h, SAPtasks.h, SAPapi.pch, et SAPInteg.h, se trouvent dans le même répertoire. Il n'est pas obligatoirement nécessaire de les modifier.

Pour configurer les extensions de la fonction non prioritaire SAP, procédez comme suit :

1. Si vous souhaitez modifier puis compiler l'API, vous devrez installer le kit d'outils RFC sur le poste de travail que vous utilisez. Dans ce cas, le répertoire de bibliothèques du SDK RFC doit être inclus dans le chemin et le fichier de services doit contenir toutes les entrées SAP.
2. Modifiez les quatre fichiers C++, ou bien créez vos propres versions de ces fichiers.
3. Un fichier makefile se trouve dans le même répertoire que les fichiers source. Modifiez ce fichier afin qu'il pointe sur le répertoire approprié du SDK RFC.
4. Tapez l'instruction suivante à partir de la ligne de commande :
db2 connect to database
db2 -tvf [nomfichier]
pour exécuter le script SQL :

WIN register.cmd

AIX register.sql

SUN register.sun.db2.sql

Ce script remplace l'instruction par défaut OrderDisplay par une nouvelle version, qui se trouve dans le fichier suivant :

WIN OrderDisplay.dll

AIX libOrderDisplay.a

SUN libOrderDisplay.so

qui appelle la BAPI de simulation de commandes de SAP.
Elle enregistre également le fichier suivant :

WIN SAPInteg.dll

AIX libSAPInteg.a

SUN libSAPInteg.so

afin qu'il exécute les fonctions SimulateOrdProc et ExtOrdProc.

(Pour enregistrer les fonctions, vous pouvez utiliser les fonctions de gestion des activités de l'administrateur Net.Commerce, mais pour remplacer l'instruction, vous devez entrer les informations manuellement dans la table de base de données CMDS.) Dans le cas d'une base de données Oracle, exécutez le script suivant :

WIN register.nt.ora.sql

AIX register.aix.ora.sql

SUN register.sun.ora.sql

Remarque : Pour les opérations ci-après, IDES doit être installé sur le système SAP. Les chemins d'accès cités sont les chemins par défaut. Remplacez-les par les chemins de votre choix, si nécessaire.

5. Copiez les fichiers d'images à partir du répertoire

WIN [rép_principal_Net.Commerce]\adt\samples\SAP\testing
vers le répertoire

[rép_principal_Net.Commerce]\html\fr_FR\bus2bus2\images.

AIX **SUN** [rép_principal_Net.Commerce]/adt/samples/SAP/testing

vers le répertoire [rép_principal_Net.Commerce]/html/fr_FR/bus2bus2/images

6. Copiez le fichier sap-commerce.ini à partir du répertoire

WIN [rép_principal_Net.Commerce]\adt\samples\SAP\testing vers le répertoire
[rép_principal_Net.Commerce]\bin.

AIX **SUN** [rép_principal_Net.Commerce]/adt/samples/SAP/testing vers le
répertoire [rép_principal_Net.Commerce]\bin.

La fonction modèle SAP lit normalement ce fichier à partir du répertoire local, mais vous pouvez la modifier de sorte qu'elle le lise à partir d'un autre répertoire, si vous le souhaitez.

7. Installez le modèle Office Window à l'aide de l'Administrateur Net.Commerce.

8. Démarrez le serveur Net.Commerce à partir du répertoire :

WIN [rép_principal_Net.Commerce]\bin

AIX **SUN** [rép_principal_Net.Commerce]/bin

Pour cela, tapez l'instruction suivante à partir de la ligne de commande :

WIN srvrctrl - \ibm\www\html\srvrctrl.ini.

AIX **SUN** srvrctrl - /ibm/www/html/srvrctrl.ini.

9. Ouvrez le navigateur à l'adresse suivante :

http://<serveur_utilisé>/bus2bus2.

10. Enregistrez et approuvez un utilisateur SAP.

11. Pour tester la connexion, connectez-vous au site Web en tant qu'utilisateur SAP et testez le produit et les pages de commande.

Configuration SAP R/3 requise pour l'exécution de Net.Commerce

Vous devez disposer de la configuration suivante pour intégrer Net.Commerce à un système SAP R/3 :

- Le système SAP doit exécuter R/3 version 3.1G.
- Si vous souhaitez utiliser les modèles de données inclus dans la boîte à outils d'extensions SAP Net.Commerce, le système IDES (International Demonstration and Education System) doit être installé sur le système SAP.
- Les fonctions RFC (appel de fonction à distance) de SAP doivent être installées sur le système Net.Commerce. Vous les trouverez sur le CD de présentation du programme d'installation SAP.
- Les systèmes SAP et Net.Commerce doivent pouvoir communiquer via TCP/IP.

Fichiers d'intégration SAP

Les fichiers suivants peuvent être utilisés pour intégrer Net.Commerce à SAP R/3 :

Dans le répertoire :

WIN [rép_principal_Net.Commerce]\adt\samples\SAP\Testing

AIX **SUN** [rép_principal_Net.Commerce]/adt/samples/SAP/Testing

bus2bus.in	Version des modèles de données IDES dont le format permet l'importation dans Net.Commerce à l'aide de l'utilitaire d'importation globale.
sap-commerce.ini	Modèle de fichier d'initialisation utilisé dans les tests de l'intégration SAP à Net.Commerce.
	Fichiers image utilisés pour créer des pages exemple lors du test de l'intégration SAP.
WIN register.cmd WIN register.nt.ora.sql AIX register.sql AIX register.aix.ora.sql SUN register.sun.db2.sql SUN register.sun.ora.sql	Fichiers scripts de base de données qui enregistrent la fonction non prioritaire et l'instruction d'intégration SAP dans la base de données.

Dans le répertoire :

WIN [rép_principal_Net.Commerce]\adt\samples\SAP\api

AIX **SUN** [rép_principal_Net.Commerce]/adt/samples/SAP/api

sim_ord_proc.cpp	Fichier source, écrit en C++, qui peut être modifié pour l'intégration des fonctions non prioritaires de Net.Commerce au paramètre BAPI SAP de "simulation".
ext_ord_proc.cpp	Fichier source, écrit en C++, qui peut être modifié pour l'intégration des fonctions non prioritaires de Net.Commerce au paramètre BAPI SAP de "création".
SapInteg.cpp	Contient des fonctions de support communes utilisées par les fichiers ext_ord_proc.cpp et sim_ord_proc.cpp.
tasks_api.cpp	Contient des fonctions de support communes utilisée par les fichiers ext_ord_proc.cpp et sim_ord_proc.cpp.

order_create.c	WIN AIX Fichiers source, écrits en C, compilés avec les fichiers source C++ pour créer le fichier de bibliothèque modifié SAPInteg, utilisé pour personnaliser l'intégration de Net.Commerce et du système SAP R/3. En général, ces fichiers ne sont pas modifiés. SUN Fichiers source, écrits en C, compilés pour créer le fichier de bibliothèque SAPc, utilisé pour personnaliser l'intégration entre Net.Commerce et le système SAP R/3. En général, ces fichiers ne sont pas modifiés.
order-util.c	
saputils.c	
rfcutil.c	
saputil.c	
order-simulate.c	
readini.c	
readini.h	WIN AIX Fichiers d'inclusion qui sont compilés avec les fichiers source C++ pour créer le fichier de bibliothèque modifié SAPInteg, utilisé pour personnaliser l'intégration entre Net.Commerce et le système SAP R/3. En général, ces fichiers ne sont pas modifiés.
rfcutil.h	
SAPtasks.h	
SAPapi.pch	SUN Fichiers d'inclusion compris dans la compilation des fichiers de bibliothèque libSAPInteg.so et libSAPc.so.
SAPInteg.h	
WIN makefile.nt AIX makefile.aix SUN makefile.sun	Fichier makefile utilisé dans la compilation du modèle de code pour le test de l'intégration de SAP à Net.Commerce.

Dans le répertoire :

WIN [rép_principal_Net.Commerce]\bin

AIX **SUN** [rép_principal_Net.Commerce]/bin

WIN SAPInteg.dll AIX libSAPInteg.a SUN libSAPInteg.so SUN libSAPc.so	Modèle de fichier de bibliothèque compilé, contenant la fonction utilisée pour invoquer les fonctions BAPI SAP. Vous pouvez utiliser ce fichier pour intégrer SAP à Net.Commerce mais, si la base de données SAP a été personnalisée, vous devrez le modifier et le recompiler à partir des fichiers source et des fichiers d'inclusion répertoriés ci-dessous.
WIN OrderDisplay.dll AIX libOrderDisplay.a SUN libOrderDisplay.so	Fichier de bibliothèque compilé qui invoque l'activité SimulateOrder, qui doit être mise en oeuvre dans Net.Commerce pour permettre l'intégration de SAP. Ce fichier ne doit pas être modifié.

Paramètres BAPI SAP R/3 utilisés par Net.Commerce

Net.Commerce communique avec le système SAP R/3 par les paramètres BAPI suivants :

- BAPI_CUSTOMERORDER_CREATE
- BAPI_CUSTOMERORDER_SIMULATE

Les références partenaires des clients (**RéfPart**) doivent être créées sur le système SAP, puis insérées dans la base de données Net.Commerce. Vous ne pouvez pas utiliser les paramètres BAPI pour exécuter ces opérations.

Remarque : Vous pouvez encore personnaliser Net.Commerce en créant des macros et des fonctions non prioritaires qui peuvent être mises en correspondance avec d'autres fonctions BAPI.

Test du modèle de code SAP

Pour tester le modèle de code de l'intégration SAP dans un environnement hors production, procédez comme indiqué ci-dessous.

Remarques :

- Pour pouvoir utiliser ce code, le système IDES (International Demonstration Educational System) doit être installé sur le système SAP.
 - Les fichiers modèles se trouvent dans le répertoire [rép_principal_Net.Commerce]/adt/samples/SAP.
 - Il n'est pas nécessaire de les compiler si vous n'avez pas l'intention de modifier le modèle de code. Une version compilée du fichier d'intégration libSAPInteg.so se trouve dans le répertoire [rép_principal_Net.Commerce]/bin. Si vous utilisez le fichier compilé, ne tenez pas compte des étapes 1 à 3 ci-dessous.
1. Installez le kit d'outils RFC sur votre poste de travail. Le répertoire de bibliothèques du SDK RFC doit obligatoirement être inclus dans le chemin et le fichier de services doit contenir toutes les entrées SAP.
 2. Modifiez le fichier makefile afin qu'il pointe sur le répertoire approprié du SDK RFC.
 3. Compilez le code.
 4. Créez un utilisateur SAP sur le système SAP.
 5. Le fichier de la base d'enregistrement contient un script qui vous permet d'enregistrer tant l'instruction que la bibliothèque de fonctions. Connectez-vous à la base de données et lancez le script de la base d'enregistrement. Pour cela, tapez l'instruction suivante :

WIN register

AIX db2 -tvf register.sql

SUN db2 -tvf register.sun.db2.sql

Remarque : Si vous utilisez une base de données Oracle, lancez le script de la base de données Oracle.

6. Copiez le fichier [rép_principal_Net.Commerce]/adt/samples/sap/Testing/bus2bus.in dans le répertoire [rép_principal_Net.Commerce]/nc_schema/db2/bus2bus.
7. Copiez les fichiers image à partir du répertoire [rép_principal_Net.Commerce]/adt/samples/sap/Testing, dans le répertoire [rép_principal_Net.Commerce]/html/en_US/bus2bus2/images.
8. Copiez le fichier [rép_principal_Net.Commerce]/adt/samples/sap/Testing/sap-commerce.ini dans le répertoire [rép_principal_Net.Commerce]/bin.
9. Editez le fichier d'initialisation :
WIN sap-commerce.ini
AIX SUN sap-commerce.conf
afin d'entrer les paramètres système SAP (nom d'hôte TCP/IP, numéro du système, destination SAP et les paramètres client), ainsi que l'ID utilisateur et le mot de passe SAP. Vous trouverez des détails supplémentaires sur les instructions dans les commentaires contenus dans le fichier.
10. Utilisez la page Modèles et tutoriel de l'Administrateur Net.Commerce pour installer le modèle de magasin Office Window.
11. Démarrez le serveur Net.Commerce à partir du répertoire [rép_principal_Net.Commerce]/bin. Pour cela, tapez l'instruction suivante :

```
svrctrl -i /opt/GoServer/usr/internet/server_root/pub/svrctrl.conf.
```

Remarque : Par défaut, la fonction du modèle SAP lit le fichier sap-commerce.ini dans le répertoire local. Vous pouvez la modifier pour qu'elle le lise dans un autre répertoire.

12. Ouvrez le navigateur à l'adresse suivante `http://<serveur_utilisé>/bus2bus2`.
13. Dans Net.Commerce, enregistrez et approuvez un client dont l'ID de connexion (shlogid) est identique à celui d'un client valide (numéro de partenaire) sur le système SAP.
14. Pour revenir à la base de données d'origine, utilisez la page Modèles et tutoriel de l'Administrateur Net.Commerce.

MQSeries et Net.Commerce

Utilisez le modèle de code fourni avec MQSeries en tant que logiciel intermédiaire pour intégrer Net.Commerce en tant que système frontal de l'application ou du processus que vous utilisez. Les messages sortants transmis par Net.Commerce au système cible sont traités en remplaçant certaines fonctions livrées par défaut avec Net.Commerce et en les acheminant vers la file d'attente des messages MQSeries. Les messages issus du système cible sont traités par un agent Net.Commerce (worker), invoqué de façon périodique par le démon d'arrière-plan Net.Commerce pour lire la file d'attente des messages MQSeries.

Utilisez les scripts fournis pour créer des environnement et des tables de base de données, ainsi que du code source et des fichiers makefile que vous pouvez compiler pour créer les fichiers suivants :

- **WIN** test_mq.exe

AIX **SUN** test_mq

Fichier exécutable qui envoie et reçoit un message test.

- **WIN** receiver.exe

AIX **SUN** receiver

Modèle de fichier exécutable sur le poste MQSeries qui, lorsqu'il s'exécute, vide la file d'attente des messages dans un fichier journal. Vous pouvez utiliser ce fichier pour vérifier si les messages de Net.Commerce parviennent normalement au système existant. Pour la production, utilisez vos applications existantes pour recevoir et traiter les messages.

- **WIN** ncei-mq.dll

AIX libncei_mq.a

SUN libncei_mq.so

Modèle de fichier DLL qui gère les messages sortants en remplaçant une activité de transaction de Net.Commerce. Plus précisément, il remplace une fonction de Net.Commerce définie par défaut afin d'exécuter l'activité Net.Commerce **AUDIT_REG** sans logique applicative et il envoie un message à MQServer sur un système existant.

- **WIN** ncei_mqdrv.dll et ncei_mqdrv.lib, modèle de fichier et de bibliothèque DLL contenant les objets MQService.

- **AIX** libncei_mqdrv.a

SUN libncei_mqdrv.so

Modèle de bibliothèque partagée qui contient les objets MQService.

- **WIN** ncei_mqserver.dll et ncei_mqserver.lib

AIX libncei_mqserver.a

SUN libncei_mqserver.so

Modèle de fichier DLL gérant les messages entrants et fichier bibliothèque utilisé pour l'édition de liens (linkedit). Ces fichiers sont particulièrement utilisés pour montrer comment les messages de la file d'attente MQSeries peuvent être reçus et traités par Net.Commerce. Comme pour le modèle de messages sortants, celui des messages entrants ne contient aucune logique applicative.

Utilisation de MQSeries pour l'intégration d'un système existant à Net.Commerce

Cette procédure explique comment établir une connexion entre le système existant et un système Net.Commerce utilisant un logiciel intermédiaire (middleware) MQSeries pour l'échange de messages entre les deux systèmes. Elle indique également comment utiliser le modèle de code fourni avec Net.Commerce Pro

PRO pour remplacer l'activité par défaut **AUDIT_REG** de Net.Commerce, qui contrôle le flux de messages relatifs aux ID des clients, par une fonction que vous pouvez modifier pour que ces messages soient traités par une application du système existant. Pour intégrer pleinement votre système existant avec Net.Commerce utilisant MQSeries pour l'émission et la réception de messages entre les deux systèmes, vous devez créer et enregistrer vos propres fonctions d'interaction avec les applications en place sur le système existant.

Remarque : La gestion des messages en entrée et celle des messages en sortie sont facultatives. Vous n'êtes pas obligé de les configurer ou de les utiliser toutes les deux.

1. Installez les deux serveurs MQSeries ou un serveur et un client, en fonction du scénario de production que vous avez choisi.
2. Compilez le modèle de code Net.Commerce.
3. Validez le canal et testez les connexions.
4. Configurez Net.Commerce de façon à remplacer la fonction par défaut associée à l'activité **AUDIT_REG**. Le modèle de code d'intégration MQSeries permet d'utiliser les messages en sortie issus de Net.Commerce dans le système existant.
5. Configurez Net.Commerce de sorte qu'il accepte les messages en entrée issus du système existant.
6. Testez la fonctionnalité du flux de messages entre Net.Commerce et le système existant.

Deux scénarios d'intégration de MQSeries

Il existe deux manières de configurer Net.Commerce pour qu'il communique avec votre système existant en utilisant MQSeries comme logiciel intermédiaire.

Dans le premier scénario, Net.Commerce est installé sur le même poste qu'un serveur MQSeries. Le système cible dispose d'un autre serveur MQSeries, et les deux serveurs MQSeries communiquent via une connexion de serveur à serveur à l'aide d'un canal de messages. Ce canal exige que les canaux RCV et SRV soient définis comme canaux de connexion utilisés par le canal de messages.

Dans le second scénario, Net.Commerce utilise un client partiel MQSeries, MQI (MQSeries Interface), pour communiquer via un serveur MQSeries en mode client-serveur sur le système existant. Dans ce cas, vous devez installer le canal MQI. Pour ce faire, définissez les canaux SRVCONN et CLNTCONN comme canaux de connexion utilisés par le canal MQI.

Pour plus de détails sur l'installation des clients et des serveurs, sur les connexions et la définition de canaux, reportez-vous à la documentation MQSeries.

Installation de MQSeries avec Net.Commerce

Il existe deux manières d'utiliser MQSeries comme moyen de traitement des messages échangés entre un système Net.Commerce et un système dorsal séparé. Dans les deux cas, le système dorsal contient un serveur MQSeries. Dans un premier cas de figure, le système Net.Commerce comporte également un serveur MQSeries, tandis que dans l'autre, il possède un client autonome MQI (MQ Interface).

Choisissez l'interface que vous utiliserez sur le système Net.Commerce : client autonome MQI ou client+serveur MQSeries.

Configuration d'une interface MQSeries client+serveur à serveur

Si vous décidez d'installer Net.Commerce et un serveur MQSeries sur la même machine, vous pouvez configurer l'interface MQSeries client+serveur à serveur. Effectuez les opérations suivantes dans Net.Commerce :

1. Tapez l'instruction suivante à partir d'une ligne de commande pour installer un Gestionnaire de files d'attente :
`crtmqm /q [nom_Gestionnaire_file]`
2. Tapez l'instruction suivante à partir d'une ligne de commande pour installer un environnement de files d'attente par défaut :
`runmqsc <amqscoma.tst >amqscoma.log`
3. Tapez l'instruction suivante à partir d'une ligne de commande pour installer un environnement pour Net.Commerce :
`runmqsc <ncci_lcl.tst >ncci_lcl.log`

WIN Si le serveur MQSeries du système sur lequel est installée l'application dorsale fonctionne sur une machine Windows NT, procédez comme suit :

1. Tapez l'instruction suivante à partir d'une ligne de commande pour installer un Gestionnaire de files d'attente :
`crtmqm /q [nom_Gestionnaire_file]`
2. Tapez l'instruction suivante à partir d'une ligne de commande pour installer un environnement de files d'attente par défaut :
`runmqsc <amqscoma.tst >amqscoma.log`
3. Tapez l'instruction suivante à partir d'une ligne de commande pour installer un environnement pour Net.Commerce :
`runmqsc <ncci_hst.tst >ncci_hst.log`

390 Si le serveur MQSeries du système sur lequel est installée l'application dorsale fonctionne sur une machine OS/390, procédez comme suit :

1. Créez un fichier membre similaire à la définition `neci_hst.tst` dans le fichier `thlqual.SCSQPROC`, où `thlqual` correspond au qualificateur de niveau supérieur utilisé par votre installation MQSeries OS/390.
2. Assurez-vous que les files d'attente sont disponibles et que les définitions sont correctes. Pour plus de détails sur la syntaxe et la configuration de MQSeries, reportez-vous à la documentation MQSeries.

Pour plus de détails sur l'installation de MQSeries, reportez-vous à la documentation correspondante.

Configuration d'une interface MQSeries de client autonome à serveur

Si vous décidez d'installer Net.Commerce et un serveur MQSeries sur des machines distinctes, vous pouvez configurer l'interface MQSeries de client autonome à serveur. Effectuez les opérations suivantes dans Net.Commerce :

1. Installez le client autonome MQSeries.
2. Définissez la connexion client à un Gestionnaire de files d'attente en ajoutant la variable d'environnement `MQSERVER` à votre environnement système, comme suit :
`MQSERVER = [nom_canal_défini_dans_serveur]/TCP/[adresse_IP_du_serveur]`
Remarque : **AIX** **SUN** Il est tenu compte de la distinction majuscules/minuscules dans la valeur TCP.

N'oubliez pas de redéfinir le canal pour le client/serveur sur le système où est installée l'application dorsale. Pour effectuer cette opération, reportez-vous au fichier modèle Net.Commerce `neci_MQClient.tst`.

Pour plus de détails sur l'installation de MQSeries, reportez-vous à la documentation correspondante.

Compilation du modèle de code d'intégration MQSeries

Net.Commerce Pro **PRO** contient un modèle de code que vous pouvez compiler pour générer des programmes exécutables et des DLL destinés à tester une connexion de Net.Commerce vers un système existant à l'aide de MQSeries. Vous pouvez également modifier ce code pour installer votre propre logique applicative et remplacer les fonctions par défaut de Net.Commerce que vous souhaitez voir traiter sur votre système existant.

Pour compiler les fichiers, procédez comme suit :

1. Editez les fichiers :
WIN `makefile.nt`
AIX `makefile.aix`
SUN `makefile.sun`
dans les sous-répertoires `...MQSeries\API` et `...MQSeries\objects` pour vous assurer que :
 - les zones **NETCPATH** et **MQI_PATH** sont correctes.
 - **NETCPATH** définit le chemin d'installation de Net.Commerce.
 - **MQI_PATH** définit le chemin d'installation de MQSeries sur le système Net.Commerce (et non pas sur le système existant).

- Sélectionnez la bibliothèque MQSeries appropriée pour linkedit
Utilisez mqm.lib pour une connexion MQSeries serveur à serveur.
Utilisez mqmc.lib pour une connexion MQSeries client autonome à serveur.
- Exécutez le fichier makefile à partir du répertoire indiqué ci-après (et non pas ceux qui se trouvent dans les sous-répertoires de ce répertoire, puisque leur exécution est déclenchée par le fichier makefile principal) :

WIN [rép_principal_Net.Commerce]\adt\samples\ncei\MQSeries

AIX **SUN** [rép_principal_Net.Commerce]/adt/samples/ncei/MQSeries

Tapez l'instruction suivante :

WIN nmake -f makefile.nt

AIX make -f makefile.aix

SUN make -f makefile.sun

Les fichiers suivants doivent être créés dans le sous-répertoire ...MQSeries/bin :

WIN ncei_mq.dll WIN ncei_mq.lib AIX libncei_mq.a SUN libncei_mq.so	Fonctions non prioritaires de transactions Net.Commerce
WIN ncei_mqdrv.dll WIN ncei_mqdrv.lib AIX libncei_mqdrv.a SUN libncei_mqdrv.so	Objets Net.Commerce MQ Services
WIN ncei_mqserver.dll AIX libncei_mqserver.a SUN libncei_mqserver.so	Modèle de fichier DLL gérant les messages entrants
WIN ncei_mqserver.lib	Fichier bibliothèque utilisé pour linkedit
WIN receiver.exe AIX SUN receiver	Exécutable récepteur (prend les messages en file d'attente et les écrit dans un fichier journal)
WIN test_mq.exe AIX SUN test_mq	Exécutable (envoie et reçoit un message test).

Fichiers modèle utilisés pour l'intégration avec MQSeries

Les fichiers répertoriés ci-après sont fournis avec Net.Commerce Pro **PRO** afin de faciliter l'intégration du produit avec le système existant, MQSeries étant utilisé comme messagerie interprogramme entre les deux systèmes. Après l'installation de Net.Commerce, vous trouverez ces fichiers dans les répertoires suivants :

WIN [rép_principal_Net.Commerce]\adt\samples\ncei\MQSeries\

AIX **SUN** [rép_principal_Net.Commerce]/adt/samples/ncei/MQSeries/

Nom du fichier	Sous-répertoire	Objectif
WIN makefile.nt AIX makefile.aix SUN makefile.sun		Compilation des fichiers WIN receiver.exe, test_mq.exe, ncei_mq.dll, ncei_mq.lib, ncei_mqdrv.dll et ncei_mqdrv.lib AIX receiver, test_mq, libncei_mq.a et libncei_mqdrv.a SUN receiver, test_mq, libncei_mq.so, libncei_mqdrv.so
WIN ncei_go.cmd AIX SUN ncei_go.sh		Script exécuté sur le serveur MQSeries du système existant. Démarre l'initialisateur de canal, le programme d'écoute de files d'attente et le moniteur de déclenchement.
ncei_hst.tst		Script exécuté sur le serveur MQSeries du système hôte. Configure l'environnement pour Net.Commerce. Ce script est utilisé dans un scénario de serveur à serveur.
ncei_lcl.tst		Script exécuté sur le serveur MQSeries de la machine où est installé Net.Commerce. Ce script permet de configurer l'environnement pour Net.Commerce. Il est utilisé dans un scénario de serveur à serveur.
ncei_mq.db2.sql		Script DB2 qui configure et remplace la fonction TASK_REG et gère la macro d'erreur.
ncei_mq.ora.sql		Script Oracle qui configure et remplace la fonction TASK_REG et gère la macro d'erreur.
ncei_mqserver.db2.sql		Script DB2 qui configure les groupes ncei_servers pour Net.Commerce.
ncei_mqserver.ora.sql		Script Oracle qui configure les groupes ncei_servers pour Net.Commerce.
ncei_mqxacts.db2.sql		Script DB2 qui crée les tables de l'objet de transaction Net.Commerce MQSeries.
ncei_mqxacts.ora.sql		Script Oracle qui crée les tables de l'objet de transaction Net.Commerce MQSeries.
ncei_mqclient.tst		Script exécuté sur le serveur MQSeries du système existant afin de définir les types de canal SRVCONN et CLNTCONN. Il est utilisé dans un scénario de client à serveur.

Nom du fichier	Sous-répertoire	Objectif
WIN README.TXT AIX SUN README		Fichier texte contenant des informations détaillées sur la configuration de la connexion MQSeries, l'exécution des scripts et la compilation du code exemple.
WIN makefile.nt AIX makefile.aix SUN makefile.sun	api	Fichier makefile pour : WIN ncei_mq.dll AIX libncei_mq.a SUN libncei_mq.so Bibliothèque DLL modèle contenant une fonction de substitution pour AUDIT_REG .
mq_audit_reg.cpp	api	Code source de la fonction de substitution contenue dans les fichiers : WIN ncei_mq.dll AIX libncei_mq.a SUN libncei_mq.so qui exécute l'activité AUDIT_REG . Vous pouvez utiliser ce code comme modèle ou le modifier afin de créer vos fonctions de substitution personnelles.
ncei_mqapi.pch	api	Fichier d'en-tête précompilé pour les fichiers DLL.
mqMall_err_reg.d2w	macro/fr_FR	Macro d'erreur modèle pour les messages d'erreur MQSeries.
msregadd.inc	macro/fr_FR	Fichiers include utilisés par la macro mqMall_err_reg.d2w
msregform.inc	macro/fr_FR	
msreglabels.inc	macro/fr_FR	
WIN makefile.nt AIX makefile.aix SUN makefile.sun	objects	Fichier makefile qui crée les objets ncei_mqdrv et ncei_mqserver, plus WIN test_mq.exe AIX SUN test_mq Exemple de programme exécutable qui peut émettre et recevoir un message de test.
mq_service.cpp	objets	Code source d'objet utilisé dans les fichiers DLL.
mq_service.hpp	objets	Fichier d'en-tête pour l'objet de service MQSeries.
ncei_mqserver.cpp	objets	Code source d'objet utilisé pour gérer le trafic de transactions entrant Net.Commerce MQSeries.
ncei_mqserver.hpp	objets	Fichier d'en-tête pour le gestionnaire de trafic entrant.
ncei_mqxact.cpp	objets	Code source pour la gestion de l'objet de transaction Net.Commerce MQSeries.

Nom du fichier	Sous-répertoire	Objectif
ncei_mqxact.hpp	objects	Fichier d'en-tête pour les objets de transaction.
receiver.c	objects	Code source pour receiver.exe, petit programme exécutable qui retire les messages de la file d'attente et les écrit dans un fichier journal texte.
test_mq.cpp	objects	Code source pour WIN test_mq.exe AIX SUN test_mq

Validation du canal MQSeries et test des connexions

Une fois que vous avez installé MQSeries sur le système existant et sur le système Net.Commerce, puis configuré le Gestionnaire de files d'attente et l'environnement, procédez comme suit :

1. Si le serveur MQSeries du système existant se trouve sur une machine Windows NT, exécutez le script ncei_go.cmd. L'initialisateur de canal, le programme d'écoute de files d'attente et le moniteur de déclenchement sont alors démarrés. Vous pouvez également les lancer en tapant les trois instructions suivantes à partir de la ligne de commande :

```
start runmqchi
start runmqslr -t tcp
start runmqtrm
```

Pour d'autres plateformes, consultez la documentation MQSeries.
2. Pour vérifier la connexion, accédez à l'interpréteur de commandes MQSeries en tapant runmqsc à partir de la ligne de commande, et lancez un test par écho (PING) sur le canal éloigné en tapant ping chl('NCEI.TO.HOST'). Vous devez recevoir un accusé de réception du message (ping) sur l'initialisateur de canal du système éloigné. Vous pouvez également tester la file d'attente et le canal à l'aide du fichier suivant (l'un des fichiers compilés à partir du modèle de code) :

WIN test_mq.exe

AIX SUN test_mq

Remarque : Une fois que vous avez compilé l'exécutable récepteur (la compilation s'effectue dans le sous-répertoire MQSeries\bin de Net.Commerce) et que vous l'avez placé, sur le système existant, dans le répertoire du moniteur de déclenchement, ce dernier appelle le programme. Le programme récepteur extrait les messages de la file d'attente LEGACY.LOCAL et les écrit dans le fichier journal message.log qui se trouve dans le même répertoire. Si des erreurs sont détectées, le programme récepteur ajoute les informations d'erreur au fichier error.log. Le programme recherche le fichier ncei.cnf dans son répertoire en cours et le lit pour trouver le nom du Gestionnaire de files d'attente et le nom de la file d'attente locale. Vous trouverez des informations relatives à l'emplacement du programme dans le fichier script de Net.Commerce ncei_hst.tst. Vous pouvez également étudier le contenu du fichier source receiver.c pour obtenir plus de détails sur le programme récepteur.

Utilisation d'une fonction de substitution pour intégrer Net.Commerce à MQSeries

Vous pouvez utiliser les fichiers DLL d'intégration MQSeries que vous avez compilés pour remplacer la fonction par défaut de Net.Commerce qui pilote l'activité AUDIT_REG. Cette activité gère l'ID du client lors de son enregistrement. L'utilisation du modèle de code implique également l'installation d'un modèle de macro d'erreur pour remplacer la macro d'erreur qui, par défaut, est associée à cette activité. Pour remplacer la fonction par défaut, procédez comme suit :

1. Copiez les fichiers suivants :

WIN ncei_mq.dll et ncei_mqdrv.dll

AIX libncei_mq.a et libncei_mqdrv.a

SUN libncei_mq.so et libncei_mqdrv.so

dans le répertoire suivant :

WIN [rép_principal_Net.Commerce]\bin

AIX **SUN** [rép_principal_Net.Commerce]/bin .

2. Copiez le modèle macro mqMail_err_reg.d2w du sous-répertoire MQSeries macro dans le répertoire

WIN [rép_principal_Net.Commerce]\macro\demomail

AIX **SUN** [rép_principal_Net.Commerce]/macro/demomail

3. Faites une copie de sauvegarde des fichiers msreglabels.inc, msregform.inc et msregadd.inc, présents dans le répertoire

WIN [rép_principal_Net.Commerce]\macro\ncsamples

AIX **SUN** [rép_principal_Net.Commerce]/macro/ncsamples

et remplacez-les par leurs homonymes que vous trouverez dans le sous-répertoire MQSeries macro.

La macro et ses fichiers include gèrent les messages d'erreur émanant de l'appel de fonction.

4. Ajoutez les entrées suivantes

- MQ_QMANAGER <nom gestionnaire file d'attente>
- MQ_REMOTEQ LEGACY.LOCAL

au fichier d'initialisation de Net.Commerce.

WIN ncommerce.ini

AIX **SUN** ncommerce.con

1. Exécutez le script DB2 ncei_mq.db2.sql (ou ncei_mq.ora.sql dans le cas d'une base de données Oracle) pour insérer les informations appropriées dans la base de données Net.Commerce. Ce script utilise la valeur 8880 dans la colonne REFNUM de la table OFS. Si cette valeur est déjà utilisée, modifiez préalablement le script afin qu'il utilise une autre valeur disponible. Ce script accomplit les tâches suivantes :

- Il insère, dans la table OFS, une entrée correspondant à MQ_Audit_Reg, fonction que vous utiliserez à la place de la fonction par défaut qui gère l'activité AUDIT_REG.
- Il insère une entrée dans la table TASK_MER_OF pour associer l'activité AUDIT_REG à la fonction MQ_Audit_Reg.
- Il met à jour la table MACROS pour y enregistrer l'activité AUDIT_REG_ERROR.

Vous pouvez également recourir au Gestionnaire de site Net.Commerce pour enregistrer et gérer des activités, des macros et des fonctions.

Remarque : Le modèle de code d'intégration MQSeries ne contient pas de logique applicative. Il se limite à l'échange de messages de test entre Net.Commerce et les systèmes existants. Pour intégrer les deux systèmes sur un serveur Net.Commerce de production, vous devez écrire et implémenter vos propres fonctions.

Configuration de Net.Commerce pour les messages MQSeries en entrée

Voici comment utiliser le modèle de code d'intégration MQSeries pour tester l'aptitude de Net.Commerce à recevoir des messages du système existant :

Remarque : Si vous utilisez déjà le pool de serveurs dorsaux ncei_servers, ignorez les étapes 2 à 5, car elles traitent de la configuration de ce pool pour l'objet agent (worker) MQSeries Net.Commerce.

1. Copiez le fichier

WIN ncei_mqserver.dll

AIX libncei_mqserver.a

SUN libncei_mqserver.so

dans le répertoire [rép_principal_Net.Commerce]\bin.

Il s'agit de l'un des fichiers qui ont été créés lorsque vous avez compilé le modèle de code d'intégration MQSeries.

2. Ajoutez l'entrée suivante ncei_servers_1 18000/tcp # ncei_servers au fichier Services se trouvant dans le répertoire

WIN [WINDIR]\system32\drivers\etc

AIX **SUN** /etc

3. A partir d'une ligne de commande Windows, exécutez le script DB2 Universal Database ncei_mqserver.db2.sql pour créer les tables d'intégration MQSeries dans Net.Commerce. Pour ce faire, tapez l'instruction suivante :

db2 -tvf ncei_mqserver.db2.sql

Si vous utilisez une base de données Oracle, exécutez le script ncei_mqserver.ora.sql à partir d'une invite Oracle.

4. Editez les champs suivants

DBNAME - Nom de la base de données Net.Commerce

DBINST - Nom de l'instance de base de données

DBPASS - Mot de passe d'accès à la base de données (tel qu'il a été chiffré au moyen de l'instruction nc3_crypt)
dans le fichier

WIN ncei_servers.ini

AIX **SUN** ncei_servers.conf

afin d'y inclure les informations sur la base de données tirées du fichier d'initialisation principal de Net.Commerce

WIN ncommerce.ini

AIX **SUN** ncommerce.conf.

Un modèle de fichier de configuration ncei_servers est fourni dans le répertoire adt\samples\ncei.

5. Copiez le fichier de configuration ncei_servers ainsi modifié dans le répertoire qui contient le fichier

WIN srvrctrl.ini

AIX SUN srvrctrl.conf.

Par défaut, il s'agit du répertoire principal des documents du serveur Web.

Remarque : Si vous utilisez déjà

WIN ncei_servers.ini

AIX SUN ncei_servers.conf,

éditez simplement le fichier existant, dans le répertoire principal des documents du serveur Web, plutôt que d'utiliser la version fournie comme modèle.

6. L'étape suivante consiste à activer l'agent (worker) ncei_mqserver pour Net.Commerce. Pour ce faire, à partir de la ligne de commande Windows, exécutez le script ncei_mqserver.db2.sql (ou, pour une base de données Oracle, exécutez le script ncei_mqserver.ora.sql à partir d'une invite Oracle). Vous pouvez également taper les instructions suivantes à partir d'une ligne de commande DB2 :

WIN

```
insert into cmds values (201, 'ncei_mqserver.dll', 'IBM', 'NCEI',  
'Worker_MQServer', 1.0, 'Worker_MQServer', 1, NULL);  
insert into pool_cmd values ((select refnum from pools where name =  
'ncei_servers'), 201);
```

AIX

```
insert into cmds values (201, 'libncei_mqserver.a', 'IBM', 'NCEI',  
'Worker_MQServer', 1.0, 'Worker_MQServer', 1, NULL);  
insert into pool_cmd values ((select refnum from pools where name =  
'ncei_servers'), 201);
```

SUN

```
insert into cmds values (201, 'libncei_mqserver.so', 'IBM', 'NCEI',  
'Worker_MQServer', 1.0, 'Worker_MQServer', 1, NULL);  
insert into pool_cmd values ((select refnum from pools where name =  
'ncei_servers'), 201);
```

où 201 est un numéro de référence choisi arbitrairement, qui peut être changé si nécessaire pour éviter les conflits.

7. A partir de la ligne de commande Windows, exécutez le script suivant pour créer les nouvelles tables de base de données MQCLIENT, MQXACT et MQXACTLOG :
db2 -tvf ncei_mqxact.db2.sql
Remarque : Si vous utilisez une base de données Oracle, exécutez le script ncei_mqxact.ora.sql à la place.
8. A partir d'une ligne de commande DB2, tapez les instructions suivantes pour insérer les informations dans la table MQCLIENT :
insert into mqclient values (1, '[Nom_Gestionnaire_File]',
'[Nom_File_pour_Trafic_Entrée]');
insert into mqclient values (2, '[Nom_Gestionnaire_File]',
'[Nom_File_pour_Trafic_Sortie]');
où le nom du Gestionnaire de files d'attente et les noms des files d'attente correspondent aux noms que vous avez définis.
9. A partir d'une ligne de commande DB2, tapez les instructions suivantes pour insérer les informations MQSeries dans la table MQXACT :
insert into mqxact values (1, 'A', 'GET', NULL);
insert into mqxact values (2, 'A', 'PUT', NULL);

MQCLIENT : Table du client MQSeries

La table MQCLIENT contient des informations relatives au Gestionnaire de files d'attente et aux noms de files d'attente des messages utilisés pour intégrer un système existant à Net.Commerce, à l'aide de MQSeries. La table inclut deux lignes, une pour les files d'attente du trafic entrant et l'autre pour celles du trafic sortant. Toutes deux utilisent le même gestionnaire. Les informations sont entrées manuellement par l'administrateur de site lors de la création d'un support pour les messages MQSeries, et sont utilisées par le démon NCEI pour traiter les messages entrants et sortants.

Nom de la colonne	Type de colonne	Description de colonne
MQREFNUM	INTEGER NOT NULL	Référence de la ligne. Lors de l'installation d'un support MQSeries, une valeur égale à 1 est entrée pour le nom de la file d'attente entrante et une valeur égale à 2 pour le nom de la file d'attente sortante. Il s'agit de la clé primaire.
MQQMNAME	CHAR (48) NOT NULL	Nom du gestionnaire de files d'attente installé en même temps que l'environnement MQSeries.
MQQNAME	CHAR (48) NOT NULL	Nom de la file d'attente.

MQXACT : Table des transactions MQSeries

La table MQXACT contient des informations relatives aux transactions définies pour l'intégration de NCEI et MQSeries. L'état d'une transaction MQSeries peut être actif ou inactif. L'action exécutée peut être "GET" ou "PUT". Pour définir ces actions, insérez manuellement les valeurs dans la table lors de la configuration de Net.Commerce en vue de l'utilisation de MQSeries pour communiquer avec un système existant.

Nom de la colonne	Type de colonne	Description de colonne
MXREFNUM	INTEGER NOT NULL	Référence de la file de messages. Il s'agit d'une clé associée qui fait référence à la colonne MQREFNUM de la table MQCLIENT.
MXSTATUS	CHAR (1) NOT NULL	Etat de la transaction. Les valeurs sont les suivantes : A - Actif I - Inactif
MXACT	CHAR (3) NOT NULL	Nom de l'action. Les valeurs sont les suivantes : PUT GET
MXMERREFNUM	INTEGER	Référence du commerçant. Une valeur NULL signifie que l'intégration MQSeries se fait au niveau du site. Il s'agit d'une clé associée qui fait référence à la colonne MERRFNBR de la table MERCHANT.

MQXACTLOG : Table du journal des transactions MQSeries

La table MQXACTLOG contient un enregistrement de chaque transaction exécutée à l'aide des messages transmis par la file d'attente MQSeries. L'état d'une entrée du journal des transactions peut être Ouvert, En cours, Terminé ou Erreur.

Nom de la colonne	Type de colonne	Description de colonne
MLREFNUM	INTEGER NOT NULL	Référence de la transaction dans le journal. Il s'agit de la clé primaire.
MLSTATUS	CHAR(1)	Etat de la transaction. Les options disponibles sont les suivantes : O - Ouverte W - En cours C - Terminée E - Erreur
MLMQREFNUM	INTEGER	Numéro de la file d'attente. Cette zone contient la même valeur que celle utilisée dans la colonne MQREFNUM de la table MQCLIENT.
MLMERREFNUM	INTEGER	Référence du commerçant. Cette zone contient la même valeur que celle utilisée dans la colonne MERRFNBR de la table MERCHANT. Une valeur NULL signifie que l'opérateur de transaction fonctionne au niveau du site.
MLCRTSTMP	TIMESTAMP	Horodatage de la création de la transaction.
MLLUPSTMP	TIMESTAMP	Horodatage de la dernière mise à jour de la transaction.
MLERROR	CHAR (254)	Message qui s'affiche si une erreur s'est produite au cours de la transaction.
MLMSG	LONG VARCHAR	Contenu du message.
MLFIELD1	VARCHAR (254)	Réservé à la personnalisation du commerçant ou du centre commercial.
MLFIELD2	VARCHAR (254)	Réservé à la personnalisation du commerçant ou du centre commercial.

Test de la fonctionnalité du flux de messages MQSeries

Après avoir installé le canal MQSeries et la fonction de substitution, vous pouvez tester le flux des messages entrants émis par Net.Commerce en direction du système existant. Pour ce faire, procédez comme suit :

1. Démarrez le serveur Net.Commerce en activant la trace de débogage. A partir du répertoire bin de Net.Commerce, tapez l'instruction suivante :

WIN `svrctrl -i [chemin_complet_vers_fichier_configuration_svrctrl]\svrctrl.ini`

AIX **SUN** `svrctrl -i
[chemin_complet_vers_fichier_configuration_svrctrl]/svrctrl.conf`

2. Ouvrez un navigateur Web à l'URL suivant :
<hôte_utilisé>/demoma11/basema11.htm.
3. Cliquez sur le bouton **Enregistrement** pour ouvrir le formulaire d'enregistrement.
4. Entrez les informations souhaitées dans le formulaire d'enregistrement.
5. Cliquez sur **OK**. L'activité sollicitée, **AUDIT_REG**, appelle maintenant la fonction **MQ_Audit_Reg**.

Si la connexion fonctionne correctement et que les messages sont transmis, l'appel de fonction **MQ_Audit_Reg** apparaît dans le fichier journal de trace et un message est reçu par le serveur MQSeries éloigné sur le système existant.

Pour tester le flux des messages entrants émis par le système existant en direction de Net.Commerce, procédez comme suit :

1. Utilisez
 test_mq.exe
  test_mq
sur le système existant pour insérer (**PUT**) un message dans la file d'attente NCEI.LOCAL.
2. Consultez la table MQXACTS de Net.Commerce. Le message doit y être consigné.

CICS et Net.Commerce

Net.Commerce Pro fournit un modèle de code que vous pouvez utiliser pour intégrer Net.Commerce en tant que système frontal d'un serveur de transactions CICS (Customer Information Control System) s'exécutant sur un système existant. Les messages sortants transmis par Net.Commerce au système existant sont traités en remplaçant certaines fonctions livrées par défaut avec Net.Commerce et en les redirigeant vers le serveur existant via l'interface EPI (External Presentation Interface) de CICS. Les messages issus du système existant sont traités par un agent Net.Commerce (worker), invoqué de façon périodique par le démon d'arrière-plan Net.Commerce NCEI.

Remarque : Les fichiers fournis contiennent un modèle de code, c'est-à-dire qu'ils fournissent un exemple d'intégration de Net.Commerce avec CICS. Ils ne contiennent aucune logique applicative et utilisent l'interface EPI de CICS. Lorsque vous créez vos applications pour la transmission de données entre les deux systèmes, vous pouvez également utiliser l'interface ECI (External Call Interface).

Ce modèle de code ajoute une zone de numéro CICS (CICS Number) dans le formulaire d'enregistrement du centre commercial de démonstration de Net.Commerce. Il utilise la colonne SHCHANNS2 de la table SHOPPER de Net.Commerce (par défaut, une question piège). Ce numéro CICS permet de comparer la zone du "nom de famille" dans le "MENU" de transaction CICS du serveur CICS. Une fois le modèle de code installé et testé, Net.Commerce annule la transaction si la valeur de la zone "nom de famille" dans Net.Commerce ne correspond pas à celle de la même zone dans le "MENU" CICS. Cette procédure est la suivante :

- L'activité Net.Commerce **AUDIT_REG** s'assure que les valeurs des zones SALNAME, SAADDR1, SACITY, SASTATE, SACNTRY et SAZIPC de la table SHOPPER ne sont pas définies par NULL. Ceci force le testeur à remplir ces

zones. La zone SALNAME (nom de famille) et la nouvelle zone SHCHAANS2 (numéro CICS), qui peut être vide, sont utilisées pour la vérification des transactions.

- La fonction CICS de Net.Commerce valide la zone SHCHAANS2 (numéro CICS), qu'elle soit vide ou non. Si elle n'est pas vide, la fonction invoque une transaction sur le serveur CICS, à l'aide de l'interface EPI CICS, pour extraire le nom de famille associé au numéro CICS à partir du "MENU" de transaction CICS.
- Si le nom de famille extrait du serveur CICS correspond à celui entré sur le formulaire d'enregistrement Net.Commerce, une page indiquant de l'opération a abouti s'affiche. Sinon, c'est une page d'erreur qui est générée.
- La fonction consigne alors la transaction dans la table CICSXACTLOG de Net.Commerce (table de consignation des transactions CICS).

Des scripts sont fournis pour vous permettre de créer des environnement et des tables de base de données, ainsi que du code source et des fichiers makefile que vous pouvez compiler pour créer les fichiers suivants :

- **WIN** ncei_cicsepi.dll
AIX libncei_cicsepi.a
SUN libncei_cicsepi.so
Modèle de fichier de bibliothèque qui utilise l'interface EPI de CICS pour communiquer avec un serveur CICS. Il permet d'exécuter l'activité Net.Commerce **AUDIT_REG** à la place de la fonction Net.Commerce par défaut. Pour cela, il n'utilise pas de logique applicative mais transmet un message au serveur CICS.
- **WIN** ncei_cicssserver.dll
Modèle de programme DLL qui traite les messages entrants et ncei_cicssserver.lib, bibliothèque utilisée pour l'édition de liens (linkedit).
- **AIX** libncei_cicssserver.a
SUN libncei_cicssserver.so
Modèle de bibliothèque partagée qui traite les messages entrants.
- **WIN** ncei_cicsxact.lib
AIX libncei_cicsxact.a
SUN libncei_cicsxact.so
Modèle de programme de bibliothèque qui contient des objets pour le traitement des transactions CICS et pour la consignation des transactions.
- **AIX** **SUN** cicsdaem
Démon qui s'exécute en arrière-plan et qui traite les connexions, les ressources et l'exécution des transactions CICS.
- **AIX** **SUN** daemcInt
Programme qui peut être utilisé pour tester le démon CICS et pour l'arrêter.
- **AIX** libncei_cicssdk.a
SUN libncei_cicssdk.so
Fichier de bibliothèque qui traite l'interface vers le serveur CICS sur le système existant.

Intégration d'un système CICS avec Net.Commerce

Cette procédure explique comment établir une connexion entre le serveur CICS installé sur votre système existant et un système Net.Commerce. Elle indique également comment utiliser le modèle de code fourni avec Net.Commerce pour remplacer l'activité par défaut **AUDIT_REG** de Net.Commerce, qui contrôle le flot de messages relatifs aux ID des clients. La nouvelle fonction peut être modifiée pour que ces messages soient traités sur le système existant. Pour intégrer pleinement votre système existant avec Net.Commerce, vous devez créer et enregistrer vos propres fonctions d'interaction avec les applications en place sur le système existant.

Remarque : La gestion des messages en entrée et celle des messages en sortie sont facultatives. Vous n'êtes pas obligé de les configurer ou de les utiliser toutes les deux.

1. Installez et configurez votre environnement client CICS.
2. Validez et testez la connexion.
3. Compilez le modèle de code Net.Commerce fourni dans les fichiers source.
4. Configurez une fonction de substitution de façon à établir un trafic en sortie entre Net.Commerce et le système existant.
5. Configurez Net.Commerce de sorte qu'il accepte le trafic en entrée à partir du système existant.
6. Testez la fonctionnalité du flot de messages entre Net.Commerce et le système existant.

Configuration matérielle et logicielle requise pour l'intégration CICS

Pour pouvoir utiliser le code exemple fourni afin d'intégrer Net.Commerce avec un serveur de transactions CICS sur un système existant, vous devez disposer de la configuration suivante :

- IBM Net.Commerce Version 3.1 PRO
- IBM DB2 Software Development Kit (même si vous utilisez une base de données Oracle)
- IBM Transaction Server for Windows NT (seule la partie client du client-serveur CICS est requise pour le code exemple)

Installation de l'environnement client CICS

Vous pouvez choisir d'accéder à votre hôte CICS de l'une des manières suivantes :

- Utilisation d'un client CICS pour accéder au système existant via APPC. Vous devez, dans ce cas, installer le composant Microsoft SNA Server.
- Utilisation de Transaction Server pour NT comme routeur. Pour ce faire, vous devez établir une connexion TCP/IP vers Transaction Server et la machine sur laquelle celui-ci est installé doit également comporter Microsoft SNA Server.

Pour des instructions détaillées sur l'installation et la configuration du serveur et du client CICS, reportez-vous à la documentation de ces produits.

Pour configurer l'environnement du client CICS, procédez comme suit :

1.  Mettez à jour le fichier `cicsccli.ini` sur l'ordinateur client (Net.Commerce) pour faire en sorte que le client reconnaisse le serveur CICS. Une entrée type, pour l'utilisation de TCP/IP, doit avoir le format suivant :

Server = [nom de serveur CICS choisi arbitrairement]
Description = [description choisie arbitrairement pour le serveur]
Protocol = TCPIP
NetName = [nom hôte du serveur CICS]
Port = [numéro de port]

Remarque : La zone NetName doit contenir l'adresse IP du serveur et le numéro de port doit être celui de la machine sur laquelle ce serveur est installé.

2. **AIX SUN** Ajoutez la variable d'environnement suivante :
CICS_HOSTS=[adresse IP du serveur CICS]
3. **WIN** Pendant que Net.Commerce est en cours d'exécution, lancez le client CICS. Pour ce faire, cliquez sur l'icône **Démarrage du client**, dans le dossier CICS pour Windows NT, ou tapez l'instruction suivante à partir d'une invite Windows :
[Client CICS]\bin\cicscli.exe

Remarque : **AIX SUN** Pour les instructions relatives au démarrage du client CICS, consultez la documentation CICS.

Compilation du modèle de code d'intégration CICS

Net.Commerce Pro contient un modèle de code que vous pouvez compiler pour générer des exécutables et des DLL/bibliothèques partagées destinés à tester une connexion de Net.Commerce vers un serveur CICS en place sur votre système existant. Vous pouvez également modifier ce code pour installer votre propre logique applicative et remplacer les fonctions par défaut de Net.Commerce que vous souhaitez voir traiter sur votre système existant.

Pour compiler les fichiers, procédez comme suit :

1. Editez les fichiers

WIN makefile.nt

AIX makefile.aix

SUN makefile.sun

dans les sous-répertoires

WIN [rép_principal_Net.Commerce]\adt\samples\ncei\CICS_EPI\api

AIX SUN [rép_principal_Net.Commerce]/adt/samples/ncei/CICS_EPI/api
et

WIN [rép_principal_Net.Commerce]\adt\samples\ncei\CICS_EPI\objects

AIX SUN [rép_principal_Net.Commerce]/adt/samples/ncei/CICS_EPI/objects
pour vous assurer que les champs NETCPATH et CICS_PATH sont corrects.
NETCPATH définit le chemin d'installation de Net.Commerce.
CICS_PATH définit le chemin d'installation du client CICS sur le système Net.Commerce.

2. Exécutez le fichier makefile à partir du répertoire

WIN [rép_principal_Net.Commerce]\adt\samples\ncei\CICS_EPI

AIX SUN [rép_principal_Net.Commerce]/adt/samples/ncei/CICS_EPI
(et non pas ceux qui se trouvent dans les sous-répertoires de ce répertoire, puisque leur exécution est déclenchée par ce fichier makefile).
Tapez l'instruction suivante :

```

WIN nmake -f makefile.nt
AIX make -f makefile.aix
SUN make -f makefile.sun

```

Les fichiers suivants doivent être créés dans le sous-répertoire

```

WIN [rép_principal_Net.Commerce]\adt\samples\ncei\CICS_EPI\bin
AIX SUN [rép_principal_Net.Commerce]/adt/samples/ncei/CICS_EPI/bin
:
```

WIN ncei_cicsepi.dll WIN ncei_cicsepi.lib AIX libncei_cicsepi.a SUN libncei_cicsepi.so	Fonction non prioritaire de transaction Net.Commerce
WIN ncei_cicsxact.lib AIX libncei_cicsxact.a SUN libncei_cicsxact.so	Objets de transactions CICS
WIN ncei_cicsserver.dll AIX libncei_cicsserver.a SUN libncei_cicsserver.so	Modèle de programme de bibliothèque prenant en charge le trafic en entrée
AIX libncei_cicssdk.a SUN libncei_cicssdk.so	Modèle de programme de bibliothèque prenant en charge l'interface vers le serveur CICS sur le système existant
AIX SUN cicsdaem	Programme du démon exécuté en arrière-plan et prenant en charge les connexions et les ressources CICS, ainsi que l'exécution des transactions.
AIX SUN daemclnt	Programme pouvant être utilisé pour tester ou arrêter le démon CICS (cicsdaem).

Fichiers modèle utilisés pour l'intégration CICS

Les fichiers répertoriés ci-après sont fournis avec Net.Commerce Pro **PRO** afin de faciliter l'intégration du produit avec le serveur CICS sur votre système existant. Après l'installation de Net.Commerce, vous trouverez ces fichiers dans les répertoires suivants :

```

WIN [rép_principal_Net.Commerce]\adt\samples\ncei\CICS_EPI\
AIX SUN [rép_principal_Net.Commerce]/adt/samples/ncei/CICS_EPI/

```

Nom du fichier	Sous-répertoire	Objectif
WIN makefile.nt AIX makefile.aix SUN makefile.sun		Compilation des fichiers WIN ncei_cicsepi.dll, ncei_cicsserver.dll, ncei_cicsserver.lib, ncei_cicsxact.lib AIX libncei_cicsepi.a, libncei_cicsserver.a, libncei_cicsxact.a, libncei_cicssdk.a, cicsdaem, daemcInt SUN libncei_cicsepi.so, libncei_cicsserver.so, libncei_cicsxact.so, libncei_cicssdk.so, cicsdaem, daemcInt
ncei_cics.db2.sql		Script DB2 qui configure et remplace la fonction AUDIT_REG et gère la macro d'erreur.
ncei_cics.ora.sql		Script Oracle qui configure et remplace la fonction AUDIT_REG et gère la macro d'erreur.
ncei_cicsserver.db2.sql		Script DB2 qui configure les groupes ncei_servers pour Net.Commerce.
ncei_cicsserver.ora.sql		Script Oracle qui configure les groupes ncei_servers pour Net.Commerce.
ncei_cicsxacts.db2.sql		Script DB2 qui crée les tables de l'objet de transaction Net.Commerce CICS.
ncei_cicsxacts.ora.sql		Script Oracle qui crée les tables de l'objet de transaction Net.Commerce CICS.
WIN README.TXT AIX SUN README		Fichier texte contenant des informations détaillées sur la configuration de la connexion entre CICS et Net.Commerce, l'exécution des scripts et la compilation du code exemple.
WIN makefile.nt AIX makefile.aix SUN makefile.sun	api	Fichier makefile pour la compilation. WIN ncei_cicsepi.dll AIX libncei_cicsepi.a SUN libncei_cicsepi.so Bibliothèque modèle contenant une fonction de substitution pour AUDIT_REG .

Nom du fichier	Sous-répertoire	Objectif
cics_audit_reg.cpp	api	Code source de la fonction de substitution contenue dans les fichiers : WIN ncei_cicsepi.dll AIX libncei_cicsepi.a SUN libncei_cicsepi.so qui exécute l'activité AUDIT_REG . Vous pouvez utiliser ce code comme modèle ou le modifier afin de créer vos fonctions de substitution personnelles.
ncei_cicsapi.pch	api	Fichier d'en-tête précompilé pour le fichier de bibliothèque.
cicsMall_err_reg.d2w	macro/fr_FR	Macro d'erreur modèle pour les messages d'erreur CICS.
msregadd.inc	macro/fr_FR	Permet d'inclure les fichiers utilisés par la macro cicsMall_err_reg.d2w
msregform.inc	macro/fr_FR	
msreglabels.inc	macro/fr_FR	
WIN makefile.nt AIX makefile.aix SUN makefile.sun	objects	Fichier makefile qui crée la bibliothèque modèle. WIN ncei_cicsserver.dll, ncei_cicsserver.lib, ncei_cicsxact.lib, AIX libncei_cicsserver.a, libncei_cicsxact.a SUN libncei_cicsserver.so, libncei_cicsxact.so
ncei_cicslog.cpp	objects	Code source objet pour le journal des transactions CICS.
ncei_cicslog.hpp	objects	Fichier d'en-tête pour le journal des transactions CICS.
ncei_cicsserver.cpp	objects	Code source d'objet utilisé pour gérer le trafic de transactions entrantes CICS pour Net.Commerce.
ncei_cicsserver.hpp	objects	Fichier d'en-tête pour le gestionnaire de trafic entrant.
ncei_xaction.cpp	objects	Code source code pour la gestion de l'objet de transaction Net.Commerce CICS.
ncei_xaction.hpp	objects	Fichier d'en-tête pour l'objet de transaction IMS.
ncei_cicsobjs.pch	objects	Fichier d'en-tête précompilé pour tous les objets CICS.

Nom du fichier	Sous-répertoire	Objectif
AIX SUN c3270.c c3270.h cesn.c cesn.h auxiliar.c auxiliar.h menu.c menu.h errorlog.c errorlog.h daemclnt.c cicsdaem.c cesf.c cesf.h	objects	Fichiers source et fichiers d'en-tête utilisés pour la compilation des programmes et des bibliothèques. Certains fichiers comportent des commentaires dont vous pouvez vous inspirer pour modifier les fichiers.

Validation et test de la connexion

Pour valider et tester la connexion entre le serveur CICS en place sur le système existant et le client CICS installé sur la machine Net.Commerce, vous avez seulement besoin de démarrer un terminal CICS. Toute transaction autorisée sur la machine serveur est alors accessible à partir de la fenêtre du client.

Utilisation d'une fonction de substitution pour intégrer Net.Commerce avec CICS

Vous pouvez utiliser les fichiers de bibliothèque d'intégration CICS que vous avez compilés pour remplacer la fonction par défaut de Net.Commerce qui pilote l'activité **AUDIT_REG**. Par défaut, cette activité gère l'ID du client lors de son enregistrement. L'utilisation du modèle de code implique également l'installation d'un modèle de macro d'erreur pour remplacer la macro d'erreur qui, par défaut, est associée à cette activité. Pour remplacer la fonction par défaut, procédez comme suit :

1. Copiez les fichiers suivants dans le répertoire

WIN

 [rép_principal_Net.Commerce]\bin

AIX SUN

 [rép_principal_Net.Commerce]/bin :

WIN

 ncei_cicsepi.dll

AIX

 libncei_cicsepi.a

SUN

 libncei_cicsepi.so

AIX

 libncei_cicssdk.a

SUN

 libncei_cicssdk.so

AIX SUN

 cicsdaem

AIX SUN

 daemclnt

2. Exécutez le script DB2 ncei_cics.db2.sql (ou ncei_cics.ora.sql dans le cas d'une base de données Oracle) pour insérer les informations appropriées dans la base de données Net.Commerce. Ce script accomplit les tâches suivantes :

- Il insère, dans la table **OFS**, une entrée correspondant à **CICS_Audit_Reg**, fonction que vous utiliserez à la place de la fonction par défaut qui gère l'activité **AUDIT_REG**.
 - Il insère une entrée dans la table **TASK_MER_OF** pour associer l'activité **AUDIT_REG** à la fonction **CICS_Audit_Reg**.
 - Il met à jour la table **MACROS** pour y enregistrer l'activité **AUDIT_REG_ERROR**.
3. Copiez le modèle de macro `cicsMall_err_reg.d2w` du sous-répertoire `CICS_EPI/macro/fr_FR` vers le répertoire
- WIN** [rép_principal_Net.Commerce]\macro\fr_FR\demomall
- AIX SUN** [rép_principal_Net.Commerce]/macro/fr_FR/demomall.
4. Faites une copie de sauvegarde des fichiers `include msreglabels.inc`, `msregform.inc` et `msregadd.inc`, présents dans le répertoire
- WIN** [rép_principal_Net.Commerce]\macro\fr_FR\ncsamples
- AIX SUN** [rép_principal_Net.Commerce]/macro/fr_FR/ncsamples
- et remplacez-les par leurs homonymes que vous trouverez dans le sous-répertoire
- WIN** [rép_principal_Net.Commerce]\adt\samples\ncei\CICS_EPI/macro\fr_FR
- AIX SUN** [rép_principal_Net.Commerce]/adt/samples/ncei/CICS_EPI/macro/fr_FR.
- La macro et ses fichiers `include` gèrent les messages d'erreur émanant de l'appel de fonction. Les nouveaux fichiers `include` diffèrent de la version par défaut en ce qu'ils contiennent le champ "CICS Number".
5. Créez un mot de passe CICS et chiffrez-le à l'aide de l'instruction `nc_crypt` suivante, saisie à partir d'une invite :
- `nc3_crypt -e [mot de passe]`
- Cette instruction renvoie une chaîne que vous devez mémoriser et utiliser à l'étape suivante.
6. Editez également les champs suivants dans le fichier
- WIN** `ncommerce.ini` ou
- AIX SUN** `ncommerce.conf`
- afin de préciser certaines informations sur le serveur CICS :
- CICS_SERVER** - Nom du serveur de transactions CICS
- CICS_USER** - Nom d'utilisateur CICS
- CICS_PASSWORD** - Mot de passe de l'utilisateur CICS, indiqué sous la forme chiffrée obtenue à l'étape 4
- AIX SUN CICS_PORT** - Numéro du port sur lequel `cicsdaem` prend en charge les demandes. Ce numéro doit également être ajouté à `/etc/services`. Le numéro par défaut est 20000.

Remarque : Le modèle de code d'intégration CICS ne contient pas de logique applicative. Il se limite à l'échange de messages de test entre Net.Commerce et les systèmes existants. Pour intégrer les deux systèmes sur un serveur Net.Commerce de production, vous devez écrire et implémenter vos propres fonctions.

Configuration de Net.Commerce pour le trafic CICS en entrée

Voici comment utiliser le modèle de code d'intégration CICS pour tester l'aptitude de Net.Commerce à recevoir des messages du système CICS existant :

Remarque : Si vous utilisez déjà le pool de serveurs dorsaux ncei_servers, ignorez les étapes 2 à 5, car elles traitent de la configuration de ce pool pour l'objet agent (worker) CICS Net.Commerce.

1. Copiez les fichiers suivants dans le répertoire

WIN [rép_principal_Net.Commerce]\bin

AIX **SUN** [rép_principal_Net.Commerce]\bin.

Il s'agit de fichiers qui ont été créés lorsque vous avez compilé le modèle de code d'intégration CICS.

WIN ncei_cicsserver.dll

AIX libncei_cicsserver.a

SUN libncei_cicsserver.so

AIX libncei_cicssdk.a

SUN libncei_cicssdk.so

AIX **SUN** cicsdaem

AIX **SUN** daemcInt

2. Ajoutez l'entrée suivante au fichier Services se trouvant dans le répertoire

WIN [WINDIR]\system32\drivers\etc ou

AIX **SUN** /etc :

ncei_servers_1 [votre_socket_de_pools]/tcp # ncei_servers

3. A partir d'une ligne de commande, exécutez le script DB2 Universal Database ncei_cicsserver.db2.sql pour créer les tables d'intégration CICS dans Net.Commerce. Pour ce faire, tapez l'instruction suivante :

db2 -tvf ncei_cicsserver.db2.sql

Si vous utilisez une base de données Oracle, exécutez le script ncei_cicsserver.ora.sql à partir d'une invite Oracle.

4. Editez les champs suivants du fichier

WIN ncei_servers.ini

AIX **SUN** ncei_servers.conf

afin d'y inclure les informations sur la base de données tirées du fichier d'initialisation principal de Net.Commerce

WIN ncommerce.ini

AIX **SUN** ncommerce.conf.

Un modèle de fichier de configuration ncei_servers est fourni dans le répertoire adt\samples\ncei.

DBNAME - Nom de la base de données Net.Commerce

DBINST - Nom de l'instance de base de données

DBPASS - Mot de passe d'accès à la base de données (tel qu'il a été chiffré au moyen de l'instruction nc3_crypt)

5. Copiez le fichier de configuration ncei_servers ainsi modifié dans le répertoire qui contient le fichier

WIN svrctrl.ini

AIX **SUN** svrctrl.conf.

Par défaut, il s'agit du répertoire principal des documents du serveur Web.

Remarque : Si vous utilisez déjà

WIN ncei_servers.ini

AIX **SUN** ncei_servers.conf,

éditez simplement le fichier existant, dans le répertoire principal des documents du serveur Web, plutôt que d'utiliser la version fournie comme modèle.

6. Editez le fichier de configuration de contrôle du serveur

WIN `svrctrl.ini`

AIX **SUN** `svrctrl.conf`

en ajoutant `ncci_servers` à l'instruction de configuration du pool de contrôle, comme suit :

WIN `CONTROL_POOL_CONFIG [chemin ncci_servers]\ncci_servers`

AIX **SUN** `CONTROL_POOL_CONFIG [chemin ncci_servers]/ncci_servers`

7. Créez un mot de passe CICS et chiffrez-le à l'aide de l'instruction `nc_crypt` suivante, saisie à partir d'une invite :

`nc3_crypt -e [mot de passe]`

Cette instruction renvoie une chaîne que vous devez mémoriser et utiliser à l'étape suivante.

8. Editez également les champs suivants dans le fichier

WIN `ncci_servers.ini`

AIX **SUN** `ncci_servers.conf`

afin de préciser certaines informations sur le serveur CICS :

`CICS_SERVER` - Nom du serveur de transactions CICS

`CICS_USER` - Nom d'utilisateur CICS

`CICS_PASSWORD` - Mot de passe de l'utilisateur CICS, indiqué sous la forme chiffrée obtenue à l'étape 7

AIX **SUN** `CICS_PORT` - Numéro du port utilisé par le démon CICS pour prendre en charge les demandes CICS provenant de Net.Commerce

9. L'étape suivante consiste à activer l'agent (worker) `ncci_cicsserver` pour Net.Commerce. Pour ce faire, tapez l'instruction suivante sur une ligne de commande DB2 :

WIN `insert into cmds values (202, 'ncci_cicsserver.dll', 'IBM', 'NCEI', 'Worker_CICSServer', 1.0, 'Worker_CICSServer', 1, NULL);`
`insert into pool_cmd values ((select refnum from pools where name = 'ncci_servers'), 202);`

AIX `insert into cmds values (202, 'libncci_cicsserver.a', 'IBM', 'NCEI', 'Worker_CICSServer', 1.0, 'Worker_CICSServer', 1, NULL);`
`insert into pool_cmd values ((select refnum from pools where name = 'ncci_servers'), 202);`

SUN `insert into cmds values (202, 'libncci_cicsserver.so', 'IBM', 'NCEI', 'Worker_CICSServer', 1.0, 'Worker_CICSServer', 1, NULL);`
`insert into pool_cmd values ((select refnum from pools where name = 'ncci_servers'), 202);`

où 202 est un numéro de référence choisi arbitrairement, qui peut être changé si nécessaire pour éviter les conflits.

10. A partir d'une invite, exécutez le script suivant pour créer la nouvelle table de base de données `CICSXACTLOG` :

`db2 -tvf ncci_cicsxact.db2.sql`

Remarque : Si vous utilisez une base de données Oracle, exécutez le script `ncci_cicsxact.ora.sql` à la place.

CCICSXACTLOG : Table du journal des transactions CICS

La CICSXACTLOG contient les données relatives aux transactions définies pour l'intégration de NCEI et CICS Transaction Server. L'état d'une transaction CICS peut être actif ou inactif. Ses actions peuvent être définies comme "GET" ou "PUT". Pour les définir, insérez manuellement les valeurs dans la table lors de la configuration de Net.Commerce en vue de la communication avec un serveur CICS sur un système existant.

Nom de la colonne	Type de colonne	Description de colonne
CICSXREFNUM	INTEGER NOT NULL	Référence de la file de messages. Il s'agit d'une clé primaire.
CICSXSTATUS	CHAR(1) NOT NULL	Etat de la transaction. Les valeurs sont les suivantes : A- Actif I- Inactif
CICSXACT	CHAR (3) NOT NULL	Nom de l'action. Les valeurs sont les suivantes : PUT GET
CICSMERREFNUM	INTEGER	Référence du commerçant. Une valeur NULL signifie que l'intégration CICS se fait au niveau du site. Il s'agit d'une clé associée qui fait référence à la colonne MERRFNBR de la table MERCHANT.

Test de la fonctionnalité du trafic en sortie vers CICS

Après avoir mis en place la fonction de substitution CICS, vous pouvez tester le flot de messages en sortie entre Net.Commerce et le système existant. Pour ce faire, procédez comme suit :

1. Démarrez le serveur Net.Commerce en activant la trace de débogage. A partir du répertoire bin de Net.Commerce, tapez l'instruction suivante :
`svrctrl -i [chemin_complet_vers_fichier_configuration_svrctrl]`
2.   Démarrez le démon CICS en tapant l'instruction suivante à partir de la ligne de commande :
`cicsdaem [nom de région CICS] [numéro du port de prise en charge]`
3.   Testez le démon en tapant l'instruction suivante à partir de la ligne de commande :
`daemclnt TEST 0`
4. Ouvrez un navigateur Web à l'URL suivant :
`<votre_hôte>/demomall/basemall.htm`.
5. Cliquez sur le bouton **Enregistrement** pour ouvrir le formulaire d'enregistrement.
6. Entrez les informations souhaitées dans le formulaire d'enregistrement.
7. Cliquez sur **OK**. L'activité sollicitée, **AUDIT_REG**, appelle maintenant la fonction **CICS_Audit_Reg**.

Si la connexion en sortie fonctionne comme prévu et que les messages sont transmis normalement, l'appel de la fonction **CICS_Audit_Reg** doit être enregistré dans le journal de trace et un message doit également être consigné dans la table CICSXACTLOG.

Pour tester le flot des messages entrants émis par le système existant à destination de Net.Commerce, vérifiez la table CICSXACTLOG dans Net.Commerce. Le message doit y être consigné.

IMS et Net.Commerce

Net.Commerce Pro **PRO** fournit un modèle de code que vous pouvez utiliser pour intégrer Net.Commerce à une application existante, à l'aide de code généré par l'application (IMS) Web Studio.

Vous pouvez configurer Net.Commerce de sorte que les transactions soient envoyées au système existant en remplaçant certaines fonctions livrées par défaut avec Net.Commerce, et en les acheminant à un serveur IMS.

Remarque : Il s'agit d'un modèle de code qui illustre l'intégration de Net.Commerce à IMS, sans logique applicative. L'exemple utilise la transaction universelle "Annuaire téléphonique" d'IMS.

IMS Web Studio génère un ensemble de fichiers C++ contenant des définitions et des implémentations de classes C++ qui accompagnent une transaction IMS. Pour la transaction, les classes sont des dérivées de la source MFS comme suit :

- Une classe C++ est générée pour chaque page logique de la transaction. Il s'agit de sous-classes des classes IMS Web, HWSTranIn et HWSTranOut.
- Le nom de la classe générée est celui de la page logique. Il peut être remplacé par l'application IMS Web.
- Les zones d'une page logique deviennent des attributs (ou membres de données) de la classe générée pour la page logique.
- Le nom d'un attribut généré est celui de la zone. Il peut être remplacé par l'application IMS Web.
- Des méthodes d'accès sont générées pour chaque attribut. Les méthodes GET et SET sont générées pour les attributs de pages logiques d'entrée. Seules les méthodes GET sont générées pour les attributs de pages logiques de sortie.

En outre, pour les pages logiques d'entrée, les classes générées suivent les méthodes suivantes :

- setUser, pour définir l'ID utilisateur RACF
- setGroup, pour définir le nom de groupe RACF
- setHost, pour définir le nom de l'hôte TCP/IP
- execute, pour envoyer la demande au système IMS et renvoyer une des classes générées pour les pages logiques de sortie.

Lorsque vous utilisez le modèle de code pour intégrer Net.Commerce à IMS, le code peut remplacer la fonction par défaut de Net.Commerce qui permet de contrôler l'activité **AUDIT_REG**, qui elle-même contrôle l'enregistrement d'un nouvel utilisateur. Elle vérifie que les valeurs des zones de la table SALNAME, SAADDR1, SACITY, SASTATE, SACNTRY et SAZIPC de la table SHOPPER ne sont pas définies par NULL.

Le modèle de code crée une transaction qui compare la zone SAZIPC (code postal) de la page d'enregistrement de Net.Commerce à celle extraite de l'annuaire téléphonique IMS, en utilisant la zone Nom IMS pour la recherche. Si la recherche aboutit, les codes postaux sont identiques. En revanche, si les conditions ne sont pas réunies, Net.Commerce annule la transaction. La fonction consigne l'événement dans la table IMSXACTLOG de la base de données Net.Commerce.

Le kit d'intégration IMS de Net.Commerce contient plusieurs scripts qui peuvent être utilisés pour créer des environnements et des tables de base de données, ainsi qu'un code source et des fichiers makefile qui peuvent être compilés pour créer les fichiers suivants :

- **WIN** ncei_ims.dll

- **AIX** libncei_ims.a

- **SUN** libncei_ims.so

Modèle de fichier DLL qui utilise l'interface de messagerie IMS pour communiquer avec un serveur IMS. Il permet d'exécuter l'activité Net.Commerce **AUDIT_REG** à la place de la fonction Net.Commerce par défaut. Pour cela, il n'utilise pas de logique applicative mais transmet un message au serveur.

- **WIN** ncei_imsxact.lib

- **AIX** libncei_imsxact.a

- **SUN** libncei_imsxact.so

Modèle de programme de bibliothèque qui contient des objets pour le traitement des transactions IMS et pour la consignment des transactions.

Intégration d'un système IMS avec Net.Commerce

Cette procédure explique comment établir une connexion entre le serveur IMS installé sur votre système existant et un système Net.Commerce. Elle indique également comment utiliser le modèle de code fourni avec Net.Commerce pour remplacer l'activité par défaut **AUDIT_REG** de Net.Commerce, qui contrôle le flot de messages relatifs aux ID des clients, par une fonction que vous pouvez modifier pour que ces messages soient traités sur le système existant. Pour intégrer pleinement votre système existant avec Net.Commerce, vous devez créer et enregistrer vos propres fonctions d'interaction avec les applications en place sur le système existant.

1. Installez et configurez votre environnement client IMS.
2. Compilez le modèle de code Net.Commerce.
3. Validez et testez la connexion.
4. Configurez Net.Commerce de façon à remplacer la fonction par défaut associée à l'activité **AUDIT_REG**. Le modèle de code d'intégration IMS permet d'établir un trafic de Net.Commerce vers le système existant.
5. Testez le flot de messages entre Net.Commerce et le système existant.

Configuration requise pour l'intégration à IMS

Pour pouvoir utiliser le code exemple fourni afin d'intégrer Net.Commerce à un serveur de transactions IMS sur un système existant, vous devez disposer de la configuration suivante:

- IBM Net.Commerce version 3.0 PRO

- IBM DB2 Software Development Kit (même si vous utilisez une base de données Oracle)
- IBM IMS Web Studio
- Connexion IBM IMS TCPIP OTMA
- Composant d'exécution du serveur Web IBM IMS

Les trois derniers composants sont accessibles à partir du site Web IMS d'IBM.
Pour plus de détails, reportez-vous à la documentation de votre serveur.

Installation de l'environnement client IMS

Pour configurer l'environnement du client IMS, procédez comme suit :

1. Téléchargez IMS Web Studio, ITOC (IMS TCPIP OTMA Connection) et le programme d'exécution IMS Web à partir du site IBM IMS Web, à l'URL suivant :
`http://www.software.ibm.com/data/ims.`
2. Installez Web Studio sur la machine de développement.
3. Installez le programme d'exécution IMS Web sur la machine Net.Commerce.
4. Installez ITOC sur le système hôte IMS.
5. Prenez note des informations suivantes lors de l'installation, car vous devrez les entrer dans le fichier de configuration Net.Commerce pour définir les instructions propres à IMS : hôte, numéro de port, magasin de données, ID du groupe et ID utilisateur.

Remarque : L'installation de l'environnement client IMS requiert le téléchargement et l'installation du logiciel à partir du site IBM IMS Web. Consultez votre documentation IMS et les informations qui se trouvent sur le site Web pour plus de détails sur l'installation et la configuration.

Compilation de modèle de code d'intégration IMS

Net.Commerce Pro **PRO** contient un modèle de code que vous pouvez compiler pour générer des exécutables et des DLL destinés à tester une connexion de Net.Commerce vers un serveur IMS en place sur votre système existant. Vous pouvez également modifier ce code pour installer votre propre logique applicative et remplacer les fonctions par défaut de Net.Commerce que vous souhaitez voir traiter sur votre système existant.

Pour compiler les fichiers, procédez comme suit :

1. Editez les fichiers :

 `makefile.nt`
 `makefile.aix`
 `makefile.sun`

situés dans les sous-répertoires ...IMS_EPI\API et ...IMS_EPI\objects pour vous assurer que les zones NETCPATH et IMS_PATH sont correctes.
NETCPATH définit le chemin d'installation de Net.Commerce.
IMS_PATH définit le chemin d'installation d'IMS Web Studio.
2. Exécutez le fichier makefile à partir du répertoire

`[rêp_principal_Net.Commerce]\adt\samples\ncei\IMS` (et non pas ceux qui se trouvent dans les sous-répertoires de ce répertoire, puisque leur exécution est déclenchée par le fichier makefile principal). Tapez l'instruction suivante :

WIN nmake -f makefile.nt
AIX make -f makefile.aix
SUN make -f makefile.sun

Les fichiers suivants doivent être créés dans le sous-répertoire ...IMS\bin :

WIN ncei_ims.dll AIX libncei_ims.a SUN libncei_ims.so	Modèle de programme DLL qui remplace une fonction Net.Commerce par défaut et utilise l'interface de messagerie IMS pour communiquer avec un serveur IMS.
WIN ncei_imsxact.lib AIX libncei_imsxact.a SUN libncei_imsxact.so	Objets de traitement et de journalisation des transactions IMS

Fichiers modèle utilisés pour l'intégration à IMS

Les fichiers répertoriés ci-après sont fournis avec Net.Commerce afin de faciliter l'intégration du produit au serveur IMS sur votre système existant. Après l'installation de Net.Commerce, vous trouverez ces fichiers dans les répertoires suivants :

WIN [rép_principal Net.Commerce]\adt\samples\ncei\IMS\
AIX **SUN** [rép_principal Net.Commerce]/adt/samples/ncei/IMS/

Nom du fichier	Sous-répertoire	Objectif
WIN makefile.nt AIX makefile.aix SUN makefile.sun		Compile les fichiers : WIN ncei_ims.dll AIX libncei_ims.a SUN libncei_ims.so et WIN ncei_imsxact.lib AIX libncei_imsxact.a SUN libncei_imsxact.so
ncei_ims.db2.sql		Script DB2 qui configure et remplace la fonction TASK_REG et gère la macro d'erreur.
ncei_ims.ora.sql		Script Oracle qui configure et remplace la fonction TASK_REG et gère la macro d'erreur.
ncei_imsxacts.db2.sql		Script DB2 qui crée les tables de l'objet de transaction Net.Commerce IMS.
ncei_imsxacts.ora.sql		Script Oracle qui crée les tables de l'objet de transaction Net.Commerce IMS.

Nom du fichier	Sous-répertoire	Objectif
README.TXT		Fichier texte contenant des informations détaillées sur la configuration de la connexion entre IMS et Net.Commerce, l'exécution des scripts et la compilation du code exemple.
WIN makefile.nt AIX makefile.aix SUN makefile.sun	api	Fichier makefile pour : WIN ncei_ims.dll AIX libncei_ims.a SUN libncei_ims.so, Bibliothèque DLL modèle contenant une fonction de substitution pour AUDIT_REG .
ims_audit_reg.cpp	api	Code source de la fonction de substitution contenue dans les fichiers : WIN ncei_ims.dll AIX libncei_ims.a SUN libncei_ims.so qui exécute l'activité AUDIT_REG . Vous pouvez utiliser ce code comme modèle ou le modifier afin de créer vos fonctions de substitution personnelles.
ncei_imsapi.pch	api	Fichier d'en-tête précompilé pour le fichier DLL.
imsMall_err_reg.d2w	macro/fr_FR	Macro d'erreur modèle pour les messages d'erreur IMS.
WIN makefile.nt AIX makefile.aix SUN makefile.sun	objects	Fichier makefile qui crée la bibliothèque modèle. WIN ncei_imsxact.lib AIX libncei_imsxact.a SUN libncei_imsxact.so
ncei_imslog.cpp	objects	Code source d'objet pour le journal des transactions IMS.
ncei_imslog.hpp	objects	Fichier d'en-tête pour le journal des transactions IMS.
ncei_xaction.cpp	objects	Code source pour la gestion de l'objet de transaction Net.Commerce IMS.
ncei_xaction.hpp	objects	Fichier d'en-tête pour l'objet de transaction IMS.
ncei_imsobjs.pch	objects	Fichier d'en-tête précompilé pour tous les objets IMS.
Dfsivf1.mfs	objects	Fichier source IMS utilisé par IMS Web Studio pour créer des fichiers C++ qui définissent et formatent la transaction.

Nom du fichier	Sous-répertoire	Objectif
HWSLPG01.cpp	objects	Fichiers source et en-têtes C++ générés par IMS Web Studio à partir du fichier source Dfsivf1.mfs, contenant les données de formatage et de définition de la transaction.
HWSLPG02.cpp		
HWSLPG01.hpp		
HWSLPG02.hpp		

Validation et test de la connexion IMS

Une fois que vous avez installé IMS Web Studio et l'environnement d'exécution, vous pouvez tester la connexion IMS en procédant comme suit :

1. Soumettez le fichier DSFIVF1.MFS à IMS Web Studio (inclus dans le kit d'intégration IMS de Net.Commerce) pour créer un programme CGI et une page HTML. Web Studio génère automatiquement ces fichiers à partir du fichier source.
2. Placez l'exécutable dans le répertoire cgi-bin du serveur Web (et non dans le répertoire cgi-bin de Net.Commerce).
3. Ouvrez la page HTML générée, à l'aide d'un navigateur.

Si la connexion est valide, le programme CGI appelle la transaction "Répertoire téléphonique" sur l'hôte IMS.

Utilisation d'une fonction de substitution pour intégrer Net.Commerce avec IMS

Vous pouvez utiliser les fichiers DLL d'intégration IMS que vous avez compilés pour remplacer la fonction par défaut de Net.Commerce qui pilote l'activité **AUDIT_REG**. Cette activité gère l'ID du client lors de son enregistrement. L'utilisation du modèle de code implique également l'installation d'un modèle de macro d'erreur pour remplacer la macro d'erreur qui, par défaut, est associée à cette activité. Pour remplacer la fonction par défaut, procédez comme suit :

1. Copiez les fichiers suivants :
 - WIN** ncei_ims.dll et ncei_imsxact.lib
 - AIX** libncei_ims.a et libncei_imsxact.a
 - SUN** libncei_ims.so et libncei_imsxact.so
 dans le répertoire suivant :
 - WIN** [rép_principal_Net.Commerce]\bin
 - AIX** **SUN** [rép_principal_Net.Commerce]/bin.
2. Exécutez le script DB2 ncei_ims.db2.sql (ou ncei_ims.ora.sql dans le cas d'une base de données Oracle) pour insérer les informations appropriées dans la base de données Net.Commerce. Ce script accomplit les tâches suivantes :
 - Il insère, dans la table **OFS**, une entrée correspondant à **IMS_Audit_Reg**, fonction que vous utiliserez à la place de la fonction par défaut qui gère l'activité **AUDIT_REG**.
 - Il insère une entrée dans la table **TASK_MER_OF** pour associer l'activité **AUDIT_REG** à la fonction **IMS_Audit_Reg**.
 - Il met à jour la table **MACROS** pour y enregistrer l'activité **AUDIT_REG_ERROR**.

3. Exécutez le script `ncei_imsxacts.db2.sql` (ou `ncei_imsxacts.ora.sql` dans le cas d'une base de données Oracle) pour créer la table de transactions. Cette table consigne les réussites et les échecs des transactions IMS.
4. Copiez le modèle de macro `imsMa11_err_reg.d2w` du sous-répertoire IMS `macro\fr_FR` vers le répertoire

WIN [rép_principal_Net.Commerce]\macro\fr_FR\demoma11

AIX **SUN** [rép_principal_Net.Commerce]/macro/fr_FR/demomall

Remarque : Le modèle de code d'intégration IMS ne contient pas de logique applicative. Il se limite à l'échange de messages de test entre Net.Commerce et les systèmes existants. Pour intégrer les deux systèmes sur un serveur Net.Commerce de production, vous devez écrire et implémenter vos propres fonctions.

IMSXACTLOG : Table du journal des transactions IMS

La IMSXACTLOG contient les données relatives aux transactions définies pour l'intégration de NCEI et IMS Transaction Server. L'état d'une transaction IMS peut être actif ou inactif. L'action exécutée peut être "GET" ou "PUT". Pour les définir, insérez manuellement les valeurs dans la table lors de la configuration de Net.Commerce en vue de la communication avec un serveur IMS sur un système existant.

Nom de la colonne	Type de colonne	Description de colonne
CLREFNUM	INTEGER NOT NULL	Référence de la transaction. Il s'agit d'une clé primaire.
CLSTATUS	CHAR(1) NOT NULL	Etat de la transaction. Les valeurs sont les suivantes : O- Ouverte W- En cours C- Terminée E- Erreur Cette colonne forme un index lorsqu'elle est associée à CLCCREFNUM et CLMERREFNUM.
CLCCREFNUM	INTEGER	Non utilisée pour l'instant.
CLMERREFNUM	INTEGER	Référence du commerçant.
CLCRTSTAMP	TIMESTAMP NOT NULL	Horodatage de la création de la transaction.
CLLUPSTMP	TIMESTAMP	Horodatage de la dernière mise à jour de la transaction.
CLERROR	CHAR (254)	Message d'erreur.
CLMSG	LONG VARCHAR	Contenu du message.
CLFIELD1	VARCHAR (254)	Réservé à la personnalisation par le commerçant.
CLFIELD2	VARCHAR (254)	Réservé à la personnalisation par le commerçant.

Adaptation du fichier de configuration Net.Commerce pour l'intégration d'IMS

Pour que Net.Commerce reconnaisse la connexion au système IMS, vous devez entrer les instructions suivantes dans le fichier de configuration :

WIN ncommerce.ini

AIX **SUN** ncommerce.conf

(qui se trouve dans le répertoire principal des documents du serveur Web). Utilisez les valeurs adoptées lors de l'installation des composants IMS.

- **IMS_HOST** - Adresse IP qualifiée complète du système hôte IMS
- **IMS_PORT** - Port de communication sur lequel ITOC est installé
- **IMS_DATASTORE** - Emplacement des données
- **IMS_GROUP** - Niveau de contrôle d'accès
- **IMS_USER** - ID de l'utilisateur autorisé à appeler la transaction IMS

Test de la fonctionnalité du trafic IMS

Après avoir mis en place la fonction de substitution IMS, vous pouvez tester le flot de messages entre Net.Commerce et le système existant. Pour ce faire, procédez comme suit :

1. Démarrez le serveur Net.Commerce en activant la trace de débogage. A partir du répertoire bin de Net.Commerce, tapez l'instruction suivante :
`svrctr1 -i [chemin_complet_vers_fichier_configuration_svrctr1]`
2. Ouvrez un navigateur Web à l'URL suivant :
`<votre_hôte>/demomall/basemall.htm`.
3. Cliquez sur le bouton **Enregistrement** pour ouvrir le formulaire d'enregistrement.
4. Entrez les informations souhaitées dans le formulaire d'enregistrement, y compris un nom et un code postal valides, enregistrés dans la base de données IMS.
5. Cliquez sur **OK**. L'activité sollicitée, **AUDIT_REG**, appelle maintenant la fonction **IMS_Audit_Reg**.

Si la connexion fonctionne comme prévu et que les messages sont transmis normalement, l'appel de la fonction **IMS_Audit_Reg** doit être enregistré dans le journal de trace et un message doit également être consigné dans la table IMSXACTLOG.

EDI et Net.Commerce

Vous pouvez configurer Net.Commerce de sorte qu'il génère des données de commande au format approprié pour être traduites en message EDI (Electronic Data Interchange) 850 (Ordre d'achat). Cette configuration entraîne le remplacement d'une fonction Net.Commerce par défaut. Les informations doivent être envoyées à IBM Global Services (IGS), auprès de qui vous devez ouvrir un compte comprenant des boîtes aux lettres. IGS traduit alors les données au format message EDI 850 dont le contenu est formaté en zones que vous avez préalablement configurées pour les adapter à votre activité. Cette procédure est la suivante :

- Pour préparer la traduction EDI des données de la commande, souscrivez un ou plusieurs comptes auprès d'IGS, installez et configurez une ou plusieurs boîtes aux lettres d'échange d'informations, configurez les tables de la base de données Net.Commerce pour indiquer au démon d'intégration EDI comment formater les données, remplacez la fonction par défaut de Net.Commerce qui permet d'exécuter l'activité EXT_ORD_PROC par une fonction d'intégration EDI intitulée EDIExtOrdProc.
- Lorsqu'un utilisateur passe une commande, l'information est écrite dans une table de la base de données d'intégration EDI. Le démon interroge la base de données à intervalles réguliers, que vous pouvez définir, pour rechercher de nouvelles commandes. Les commandes sont groupées, écrites dans un fichier dans un format compréhensible par IGS, chiffrées et envoyées à IGS sous une forme convenue par avance, telle que FTP ou Expedite. IGS convertit les données au format message EDI 850 et envoie le résultat à votre système, à vos distributeurs ou à toute destination qui aura été préalablement convenue. Le système cible doit être prêt à recevoir des messages EDI.

L'intégration EDI peut être configurée au niveau du centre commercial ou du magasin. Vous pouvez décider de disposer d'autant de boîtes aux lettres que nécessaire, en fonction du nombre de commerçants, de groupes de clients ou des combinaisons commerçants-groupes de clients. Toutefois, pour qu'un message EDI soit généré, le client qui passe la commande doit être enregistré.

Net.Commerce permet aussi de traiter un message d'accusé de réception fonctionnel de l'IGS, qui serait comparable à un message EDI 997.

Utilisation d'EDI avec Net.Commerce

Procédez comme suit pour installer et configurer le support EDI (Electronic Data Interchange) en vue de son utilisation avec Net.Commerce :

1. Procurez-vous un compte IBM Global Services.
2. Créez et remplissez les tables EDI dans la base de données Net.Commerce.
3. Créez quatre répertoires sur votre système pour le stockage des types de fichier suivants :
 - Messages en entrée
 - Messages en sortie
 - Messages d'erreur
 - Messages d'archive
4. Configurez le logiciel de communication choisi en accord avec IGS. Il assurera l'envoi des fichiers de votre répertoire de sortie vers votre boîte aux lettres, ainsi que la réception des fichiers de cette boîte aux lettres vers votre répertoire d'entrée.
5. Configurez l'activité EXT_ORD_PROC.
6. Modifiez et installez le fichier ncei_servers.ini/ncei_servers.conf.
7. Configurez le fichier de configuration de contrôle du serveur :

WIN `svrctrl.ini`

AIX **SUN** `svrctrl.conf`

en ajoutant la ligne suivante (y compris la virgule du début) à la fin de l'instruction CONTROL_POOL_CONFIG :

WIN `,<chemin ncei_servers>\ncei_servers`

AIX **SUN** `,<chemin ncei_servers>/ncei_servers.`

Remarque : Si vous utilisez déjà un démon d'intégration dorsal avec un système existant, cette ligne figure déjà dans le fichier

WIN srvrctrl.ini

AIX **SUN** srvrctrl.conf.

Ne l'ajoutez pas une seconde fois.

8. Si vous voulez envoyer des messages EDI en mode test (l'indicateur TEST étant activé), ajoutez l'instruction suivante au fichier

WIN ncommerce.ini

AIX **SUN** ncommerce.conf :

EDI_TESTMODE TEST

Si l'indicateur TEST est activé, IGS n'enverra pas le message à la boîte aux lettres de destination. Si cette instruction n'est pas spécifiée, les messages sont envoyés avec l'indicateur PRODUCTION.

9. Vous pouvez éventuellement définir le nombre maximal de commandes à inclure dans une enveloppe. Editez le fichier

WIN ncei_servers.ini

AIX **SUN** ncei_servers.conf

pour que le nombre maximal de commandes à envoyer simultanément soit défini dans l'instruction EDI_AMX_DOCUMENT_COUNT. Si aucune valeur n'est indiquée, la valeur par défaut est 100.

10. Ajoutez l'entrée suivante au fichier services de votre système, le cas échéant. Choisissez des numéros de port qui ne sont pas utilisés.

ncei_servers_1 18000/tcp # ncei_servers

Remarque : Si vous utilisez plusieurs processus pour l'EDI, vous devez ajouter une entrée séparée pour chacun d'eux (ncei_servers_1, ncei_servers_2, etc.).

11. Exécutez le script ncei_ediserver.db2.sql (ou ncei_ediserver.ora.sql dans le cas d'une base de données Oracle). Celui-ci ajoute une ligne à la table **POOLS** pour définir un pool de serveurs dorsaux.

Remarque : La première ligne exécutable de ce script contient une instruction insert qui permet de créer le pool. Si vous avez déjà défini un pool de serveurs dorsaux pour intégrer Net.Commerce avec un système existant, supprimez la première ligne de ce script à l'aide d'un éditeur de texte avant de l'exécuter.

12. Redémarrez le contrôleur de serveur Net.Commerce.

Obtention d'un compte IBM Global Services

Pour envoyer des bons de commande sous forme de messages EDI à partir de Net.Commerce, vous devez d'abord obtenir un compte IBM Global Services (IGS). IGS assure la conversion des messages provenant de la commande Net.Commerce en messages EDI 850 (bons de commande). Ces services fournissent également les boîtes aux lettres nécessaires à la réception des messages issus de Net.Commerce et leur transmission à IGS.

1. Lorsque vous prenez contact avec IBM Global Services, veillez à préciser que vous utilisez Net.Commerce et que vous avez besoin de services de conversion intraréseau ("in-network"). Indiquez comment vous prévoyez de transférer les fichiers de et vers IGS, et décrivez la ou les boîtes aux lettres de destination des messages, ainsi que la manière dont ces messages doivent être convertis.

Remarque : Vous aurez besoin d'une adresse par association centre commercial/commerçant ou commerçant/groupe de clients pour laquelle vous souhaitez permettre l'envoi de messages EDI.

2. IBM Global Services vous fournira les informations nécessaires pour chaque adresse. Retenez bien ces informations, car vous en aurez besoin pour configurer votre logiciel de communication et la base de données Net.Commerce.
 - Informations sur l'adresse de l'émetteur
 - Informations sur l'adresse du destinataire
 - Instance d'application
3. Après avoir obtenu votre compte IBM Global Services et reçu toutes les informations utiles, commencez à mettre en place le support EDI en créant et remplissant les tables EDI dans la base de données Net.Commerce.

Création et remplissage des tables de base de données EDI Net.Commerce

Pour créer et remplir les tables de base de données Net.Commerce nécessaires au support EDI, procédez comme suit :

1. Exécutez le script ncei_editab.db2.sql (ou ncei_editab.ora.sql dans le cas d'une base de données Oracle) pour créer quatre nouvelles tables de base de données Net.Commerce : EDIOUT, EDIADDR, EDILAYOUT et EDI_ENVELOP.
2. Modifiez le script ncei_ediaddr.db2.sql (ou ncei_ediaddr.ora.sql dans le cas d'une base de données Oracle). Vous devez ajouter une ligne pour chaque association commerçant/groupe de clients/centre commercial nécessitant une boîte aux lettres séparée. Inspirez-vous de l'exemple de ligne figurant dans le script pour déterminer le format à adopter pour chaque ligne.
Remarque : Si vous laissez la zone EASGNBR (numéro de groupe de clients) à blanc, tous les messages du commerçant concerné seront envoyés à la même boîte aux lettres, indépendamment du groupe auquel les clients appartiennent. Si vous laissez la zone EAMRENBR (référence du commerçant) à blanc, tous les messages du centre commercial seront envoyés à la même boîte aux lettres, indépendamment du commerçant.
3. Exécutez le script ncei_ediaddr.db2.sql (ou ncei_ediaddr.ora.sql) pour remplir la table des adresses EDI, EDIADDR.
4. Exécutez le script ncei_ediaddkey.db2.sql (ou ncei_ediaddkey.ora.sql) pour ajouter une ligne à la table KEYS contenant le numéro de référence EDI.

EDIOUT : Table des messages en sortie EDI

La table EDIOUT contient des informations relatives aux commandes qui sont prêtes à être envoyées ou déjà envoyées à IBM Global Services en vue d'une transmission via EDI. Chaque ligne de la table contient des informations concernant une seule commande.

Nom de la colonne	Type de colonne	Description de colonne
EDORNBR	INTEGER NOT NULL	Référence de la commande. Il s'agit d'une clé qui fait référence à la colonne ORRFNBR de la table ORDERS. Il s'agit d'un index unique également.

Nom de la colonne	Type de colonne	Description de colonne
EDSTAT	INTEGER NOT NULL	Etat de la transmission EDI de la commande. Les options sont les suivantes : 1 - Prête à être envoyée 2 - Envoyée 3 - Erreur lors de l'envoi 4 - Conversion réussie 5 - Echec de la conversion 6 - Réception de 977 ACK réussie 7 - Echec du traitement de 977 ACK
EDECNBR	INTEGER	Référence de l'enveloppe EDI dans laquelle le lot est envoyé. Il s'agit d'une clé associée qui fait référence à ECRFNBR dans la table EDI_ENVELOPE.
EDRECCNT	INTEGER	Nombre d'enregistrements dans la zone LAYOUT.
EDLAYNAME	CHAR (8)	Nom de la zone LAYOUT dans l'enregistrement .DOC.
EDDOCTAG	CHAR (64)	Marque de document dans l'enregistrement .DOC.
EDSENDER	CHAR (24)	Nom de l'expéditeur dans l'enregistrement .DOC.
EDRECEIVER	CHAR (24)	Nom du destinataire dans l'enregistrement .DOC.
EDDOC	CHAR (254)	Enregistrement .DOC de la commande.

EDIADDR : Table des adresses EDI

La table EDIADDR contient des informations d'adressage sur le site, les commerçants et les groupes de clients auxquels peuvent être affectées des boîtes aux lettres distinctes pour la transmission des commandes EDI. Chaque ligne de la table contient des informations d'envoi et de réception sur une paire commerçant-groupe de clients. Si la zone correspondant au groupe de clients contient la valeur NULL, un commerçant peut envoyer toutes ses commandes vers une seule boîte aux lettres, que le client qui a passé la commande appartienne ou non à un groupe de clients. Si les zones correspondant au groupe de clients et au commerçant contiennent la valeur NULL, un gestionnaire de site peut envoyer toutes les commandes d'un site vers une seule boîte aux lettres, quel que soit le commerçant à qui le client a passé la commande.

Nom de la colonne	Type de colonne	Description de colonne
EAMENBR	INTEGER	Référence du commerçant. Il s'agit d'une clé associée qui fait référence à la colonne MERFNBR de la table MERCHANT. Il s'agit d'un index unique également.

Nom de la colonne	Type de colonne	Description de colonne
EASGNBR	INTEGER	Référence du groupe de clients. Il s'agit d'une clé associée qui fait référence à la colonne SGRFNBR de la table SHOPGRP. Il s'agit d'un index unique également.
EASENDER	CHAR (24)	Nom de l'expéditeur logique, fourni par IBM Global Services.
EARECEIVER	CHAR (24)	Nom du destinataire logique, fourni par IBM Global Services.
EALAYOUT	CHAR (8)	Nom de la présentation utilisée

EDILAYOUT : Table de présentation EDI

La table EDILAYOUT contient les enregistrements d'une commande Net.Commerce, qui représentent un bon de commande dans un format convertible par IBM Global Services en message EDI 850. Chaque ligne contient des informations sur un élément spécifique d'une commande donnée.

Nom de la colonne	Type de colonne	Description de colonne
ELORNBR	INTEGER	Référence de la commande. Il s'agit d'une clé associée qui fait référence à la colonne ORRFNBR de la table ORDERS. Si elle est accompagnée de ELRECNAME et ELRECSEQ, elle constitue un index.
ELRECNAME	CHAR (10)	Nom de l'enregistrement. S'il est accompagné de ELORNBR et ELRECSEQ, il constitue un index.
ELRECCOUNT	INTEGER	Nombre d'enregistrements physiques dans ELRECVALUE.
ELRECSEQ	INTEGER	Numéro d'ordre utilisé pour formater le message envoyé à IBM Global Services. S'il est accompagné de ELORNBR et ELRECNAME, il s'agit d'un index.
ELRECVALUE	VARCHAR (2000)	Partie du bon de commande. Les informations relatives au bon de commande se subdivisent en données, qui sont comptabilisées dans ELRECCOUNT et mises en ordre dans ELRECSEQ.

EDI_ENVELOP : Table des enveloppes EDI

La table EDI_ENVELOP contient des informations sur le conteneur virtuel où est envoyé un lot de messages. Chaque ligne de la table contient des informations sur un seul lot de bons de commandes.

Nom de la colonne	Type de colonne	Description de colonne
ECRFNBR	INTEGER	Référence de l'enveloppe. Il s'agit d'un index unique.

Nom de la colonne	Type de colonne	Description de colonne
ECDOTCOUNT	INTEGER	Nombre de points.
ECRECCOUNT	INTEGER	Nombre d'enregistrements.
ECAPPINST	CHAR (17)	Instance de l'application
ECFIL	CHAR (254)	Enregistrement de début pour un lot de bons de commande.
ECTRL	CHAR (254)	Enregistrement de fin pour un lot de bons de commande.

Configuration de l'activité EXT_ORD_PROC pour le support EDI

Net.Commerce comprend un fichier dont la fonction (EDIExtOrdProc) peut se substituer à l'activité par défaut EXT_ORD_PROC et router les messages vers le répertoire de sortie. Ce fichier s'appelle :

 nc3_ediapi.dll

 libnc3_ediapi.a

 libnc3_ediapi.so

Pour configurer Net.Commerce afin qu'il utilise cette fonction, effectuez les opérations suivantes :

1. Exécutez le script ncei_ediapi.db2.sql (ou ncei_ediapi.ora.sql si vous disposez d'une base de données Oracle). Une ligne est ajoutée à la table OFS (fonctions non prioritaires) et la fonction d'intégration EDIExtOrdProc est enregistrée.
2. Dans le **Gestionnaire de site**, sélectionnez **Gestion des activités**.
3. Dans la liste **Type d'activité**, sélectionnez **Traitement**. Une liste s'affiche dans le cadre inférieur.
4. Cliquez sur **EXT_ORD_PROC** dans la liste Nom de l'activité.
5. Si vous voulez changer la portée de l'activité, sélectionnez **Centre commercial** ou **Magasin** dans la liste **Portée** et cliquez sur **Sauvegarder**.
6. Sur la barre des tâches, cliquez sur **Affectation des activités**. Le formulaire Affectation d'une activité s'affiche.
7. Vérifiez que la zone **Type d'activité** contient la valeur **Traitement** et que **EXT_ORD_PROC** apparaît dans la zone Nom de l'activité. Si ce n'est pas le cas, retournez à la fenêtre Gestion des activités et répétez les étapes ci-dessus.
8. Sélectionnez votre magasin dans la liste ou, si vous êtes l'administrateur du site et que vous activez le support EDI pour l'ensemble du site, sélectionnez **Centre commercial**.
9. Dans le formulaire Affectation de fonction non prioritaire, cliquez sur **EDIExtOrdProc**.
10. Cliquez sur **Mettre à jour**. Un message s'affiche, indiquant que l'activité EXT_ORD_PROC sera maintenant réalisée par la fonction EDIExtOrdProc.

Modification et installation du fichier ncei_servers.ini/ncei_servers.conf pour le support EDI

Le fichier de configuration qui contrôle le démon d'intégration dorsal est le suivant :

WIN ncei_servers.ini

AIX **SUN** ncei_servers.conf.

Puisque ce fichier est utilisé uniquement en cas d'intégration de Net.Commerce avec la messagerie EDI ou un système existant, il n'est pas opérationnel initialement et vous devez donc le localiser, le modifier et le placer dans le répertoire approprié. Pour configurer le fichier, procédez comme suit :

1. Ouvrez le fichier à l'aide d'un éditeur de texte. Si vous n'utilisez pas encore le démon d'intégration dorsal, vous trouverez ce fichier de configuration dans le répertoire

WIN [NetCommerce3]\adt\samples\ncei

AIX **SUN** [NetCommerce3]/adt/samples/ncei.

2. Remplacez la mention <chemin_installation_Net.Commerce> par le véritable chemin d'accès complet au répertoire principal de Net.Commerce.
3. Entrez respectivement la clé chiffrée du commerçant et le nom d'hôte dans les zones MERCHANT KEY et MS_HOSTNAME. Ces informations figurent dans le fichier

WIN ncommerce.ini

AIX **SUN** ncommerce.conf,

situé dans le répertoire principal des documents du serveur Web.

4. Si le chemin d'accès au journal de trace et le niveau de journalisation par défaut ne vous conviennent pas, entrez les valeurs de votre choix dans les instructions MS_LOGPATH et MS_LOGLEVEL, respectivement.
5. Entrez respectivement le nom de la base de données, celui de l'instance de la base et le mot de passe d'accès chiffré dans les instructions DBNAME, DBINST et DBPASS. Ces informations figurent dans le fichier

WIN ncommerce.ini

AIX **SUN** ncommerce.conf.

6. Remplissez les zones DB_RETRY_LIMIT et DB_RETRY_INTERVAL. Les valeurs de ces paramètres peuvent être reprises telles quelles à partir du fichier

WIN ncommerce.ini

AIX **SUN** ncommerce.conf

ou modifiées.

7. Au besoin, modifiez les valeurs des paramètres BACKGROUND_CYCLE_TIME et BACKGROUND_INTER_JOB_TIME. Exprimée en secondes, la valeur du paramètre BACKGROUND_CYCLE_TIME fixe l'intervalle d'interrogation périodique de la base de données par le démon dorsal en vue de détecter la présence de nouveaux messages. BACKGROUND_INTER_JOB_TIME définit la durée, en secondes, qui s'écoule entre les travaux d'un cycle d'interrogation donné.
8. Rendez active l'instruction EDI_APP_INSTANCE (qui était auparavant en commentaire) et indiquez l'instance d'application EDI qui vous a été communiquée par IBM Global Services.

9. Rendez actives les instructions de définition de l'emplacement des répertoires de stockage des messages en entrée, en sortie, d'erreur et d'archive. Indiquez pour chacun d'eux le chemin d'accès approprié.
10. Si vous voulez modifier le nombre maximal de commandes à inclure dans une enveloppe (la valeur par défaut étant 100), ajoutez une instruction EDI_MAX_DOCUMENT_COUNT et affectez-lui la valeur souhaitée.
11. Sauvegardez le fichier et placez-le dans le répertoire contenant le fichier de configuration du contrôleur de serveur

WIN `svrctrl.ini`

AIX **SUN** `svrctrl.conf`.

Par défaut, il s'agit du répertoire principal des documents du serveur Web.

Remarques

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Cela ne signifie pas qu'IBM ait l'intention de les y annoncer. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM EMEA Director of Licensing
IBM Europe Middle-East Africa
our Descartes La Défense 5
2, avenue Gambetta 92066 Paris-La Défense CEDEX
France

Pour le Canada, veuillez adresser votre courrier à :

IBM Director of Commercial Relations
IBM Canada Ltd.
3600 Steeles Avenue East
Markham, Ontario
L3R 9Z7
Canada

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation
P.O. Box 12195
3039 Cornwallis Road
Research Triangle Park, NC 27709-2195
U.S.A.

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

Le présent document peut contenir des informations sur des produits d'autres sociétés et des références aux sites Internet desdites sociétés. IBM ne fournit aucune garantie d'aucune sorte liée au contenu, à l'exhaustivité ou à l'utilisation de telles informations.

Ce produit est basé sur le protocole SET et est en cours de test de compatibilité, mais à la date de conception de ce produit, il n'a pas encore obtenu la certification de compatibilité SET de SET Secure Electronic Transaction LLC.



Ce produit contient des composants RSA.

Marques

Les termes qui suivent sont des marques d'International Business Machines Corporation dans certains pays :

IBM	AIX	IBM Payment Server
DB2	DB2 Universal Database	DB2 Extensions
DATABASE 2	IMS	Net.Data
MQSeries	CICS	

Microsoft, Windows, Windows NET et le logo Windows sont des marques de Microsoft Corporation.

Lotus et Domino sont des marques de Lotus Development Corporation dans certains pays.

Adobe et Acrobat sont des marques d'Adobe Systems Incorporated.

Pentium est une marque d'Intel Corporation dans certains pays.

Netscape et Netscape Communicator sont des marques de Netscape Communications Corporation dans certains pays.

Java, et toutes les marques et logos incluant Java, sont des marques de Sun Microsystems, Inc.

SET Secure Electronic Transaction, SET et le logo SET sont des marques de SET Secure Electronic Transaction LLC. Toute utilisation des marques sans licence écrite fournie par SET Secure Electronic Transaction LLC est strictement interdite.

Les images de cartes de paiement, les marques et noms commerciaux indiqués dans ce produit ne doivent être utilisés que par des commerçants autorisés par le propriétaire de la marque de carte de paiement à accepter des règlements au moyen de cette carte.

D'autres sociétés sont propriétaires des autres marques, noms de produits ou logos qui pourraient apparaître dans ce document.

Informations sur la licence

Lorsque vous faites l'acquisition de la licence du logiciel Net.Commerce, vous recevez une licence pour le logiciel IBM Net.Commerce pour AIX ou pour le logiciel IBM Net.Commerce pour Windows NT. Vous recevez également une licence pour les versions des logiciels suivants :

- IBM DB2 Universal Database
- Domino** Go Webserver
- IBM DB2 Extensions

Ces licences ne vous autorisent à utiliser ces produits qu'en association avec le logiciel Net.Commerce pour lequel vous avez acquis la licence.

Si vous souhaitez utiliser les logiciels Oracle et Domino Go Web Server ou Netscape Enterprise Server 3.0 plutôt que les logiciels DB2 et Domino Go Webserver, fournis sur le CD-ROM Net.Commerce, vous devez vous procurer des licences pour ces logiciels.

IBM