

IBM Net.Commerce
Version 3.1.2 and 3.2



IBM Net.Commerce Integration Kit

Customer Service Integration Kit

Sample Program



IBM Net.Commerce

Version 3.1.2 and 3.2

IBM Net.Commerce

Customer Service Integration Kit – Sample Program

Note:

Before using this information and the product it supports, read the information in “Appendix A: Notices and Trademarks” on page 16.

First Edition (August 1999)

This edition applies to version 1 of the Customer Service Integration Kit – Sample Program and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright International Business Machines Corporation 1999. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

1. Introduction.....	1
2. Before You Begin	1
2.1. Assumptions	2
2.2. Prerequisites	2
2.3. Unpack the Customer Service Kit.....	2
3. Decide Where to Integrate	2
4. Integrate With an Existing Store.....	2
4.1. Create a Customer Service Link	3
4.2. Determine the Shopper's Current Activity	3
4.3. Access Shopper Profile Information	4
4.4. Access Shopping Basket Information	6
4.5. Invoke the Shopper Profile and Shopping Basket SQLs.....	8
4.6. Sixth Avenue File Mapping to the Net.Commerce Directory Structure	10
4.7. Provide Integration Information to Store Administrators	11
5. Integrate with a Store Creator Store Model	11
5.1. Create a Customer Service Link	12
5.2. Determine the Shopper's Current Activity	12
5.3. Provide Integration Information to Store Administrators	14
6. Files and Technical Notes.....	15
Appendix A: Notices and Trademarks.....	16

Tables

Table 1: CS_SQL() Information5

Table 2: CS_ShoppingBasket SQL() Information.....7

Table 3: Sixth Avenue Integration File Mapping 10

1. Introduction

This document is intended for any Independent Software Vendor (ISV) who wishes to integrate their customer service application or service with IBM Net.Commerce. The Customer Service Kit (CSK) shows you how to integrate your application or service with a Net.Commerce online store to offer customer service functionality to shoppers from any location in the store. By following the instructions in this kit, a Customer Service Representative (CSR) may become familiar with the information available to assist Net.Commerce shoppers, and create a demo to market their integration solution to merchants who use Net.Commerce. Your integration may involve simply creating a link to your application or service from a store. Additionally, your integration may involve accessing the shopper-specific information available in this kit for the purpose of channeling shopper questions, via your own customized implementation.

The Net.Commerce Customer Service Integration Kit consists of sample programs that show how to integrate a customer service application with Net.Commerce using Windows NT, AIX, or the Solaris operating environment. The sample programs have not been tested under all conditions and may contain errors. IBM does not guarantee or imply reliability, serviceability, or function of these programs. The Integration Kit is provided “AS IS” with no warranty of any kind.

You can customize the demo included in this kit to invoke your application or service’s application programming interface and provide the demo to a merchant so that they can use it as a model to integrate your solution with their existing store. Also, you can use the instructions and samples in this kit to integrate your application or service with an existing Net.Commerce Store Creator store model, from which a merchant may create a new store that automatically links to your application or service. Then, a Store Administrator acting on behalf of a merchant may use your input to integrate their existing store with your application or service, or create a new store from a Net.Commerce store model which you have already integrated with your application or service. This kit includes a set of customized HTML pages and macros for the Net.Commerce sample Metropolitan mall store, Sixth Avenue, and some sample template macros for the store models.

2. Before You Begin

To use this information, you must be familiar with the general structure of a Net.Commerce Web site and store, including the name of the Net.Commerce database, the database administrator’s user ID and the password. You also should be familiar with the Net.Commerce database schema: the names and contents of relevant tables and columns in the database. This information is available in the IBM document, *Net.Commerce Commands, Tasks, Overridable Functions and Database Tables*, distributed as file name `dbtofcmd.pdf` with Net.Commerce. You should also be familiar with Structured Query Language (SQL) and JavaScript.

2.1. Assumptions

This implementation of the CSK assumes the following:

- You wish to have a customer service link from the footer of each page in a store.
- Your customer service application or service has Web access.

2.2. Prerequisites

You need to have installed IBM Net.Commerce 3.1.2, or later, on Windows NT, AIX, or the Solaris Operating Environment. You also need to install the sample Metropolitan mall with the Sixth Avenue store. Instructions on how to install the mall are detailed in the Net.Commerce online help in the section “Install the Database for the Metropolitan Mall.

2.3. Unpack the Customer Service Kit

This kit contains several sample HTML files and Net.Data macros for Sixth Avenue. It also contains sample template macros for the Net.Commerce store models. Unpack the kit file, `NetC_CustomerServiceKit_1_0.zip` into a temporary directory.

3. Decide Where to Integrate

Depending upon the target audience for your customer service integration, you may want to integrate your customer service application with either the Net.Commerce Metropolitan mall or with a Net.Commerce Store Creator store model.

Integrating your customer service application with the Metropolitan mall will provide a sample that merchants can use when enabling their own, existing Web storefront with customer service functionality. Use this integration method if you intend to sell your application or service to existing Net.Commerce customers with existing Web storefronts.

Integrating your customer service application with a Store Creator store model will automatically provide any new stores created using that store model with links to your customer service application. Use this integration method if you intend to sell your application or service to new Net.Commerce customers who have not yet created their Web storefronts.

You may choose to implement either, or both, of these integration methods depending on your business needs and target market.

4. Integrate With an Existing Store

Net.Commerce stores shopper information in database tables when a shopper registers with a store or mall and orders products. The database that loads the Metropolitan mall is called `demomall`. Several Net.Data macros display the Net.Commerce shopping flow pages that collect and display shopper profile and shopping basket information. The shopping flow consists of category, product, order and shopping basket pages. You can use the samples in this kit to create a demo for your customer service application or service. You can also customize the samples in this kit to include your application or service’s information for a specific store. This kit shows

you how integrate your customer service application or service with Sixth Avenue to do the following:

- Create a customer service link
- Determine the shopper's current activity
- Access shopper profile information
- Access shopping basket information

Note: See section 4.6, "Sixth Avenue File Mapping to the Net.Commerce Directory Structure," for the directory paths of files that you modify when using this kit.

4.1. Create a Customer Service Link

Most Net.Commerce stores use one footer file that is appended to all pages in the store. The sample Sixth Avenue footer file in this kit, `foot_6th.htm`, shows you how to modify the HTML to include your customer service logo and a link to your application or service.

To create a customer service link, do the following:

1. In a text or HTML editor, open the sample footer file `foot_6th.htm`
2. In the following sample link (text that needs to be modified is shown here in **boldface**):

```
<A HREF=http://w3.ibm.com?url=" + escape(loc) + "&" +  
escape(param) + "&" + escape(shopping_bkt) + ">ADD CS LOGO</A>
```

- a. Modify the URL to that of your customer service application or service.
- b. Modify the text to include your customer service logo.

3. Save the file.

4.2. Determine the Shopper's Current Activity

You may want to know which page the shopper was viewing or what activity the shopper was attempting to perform when they clicked on your customer service link. Append the current URL as a name-value pair to your customer service application URL. A CSR can then view a page simultaneously with the shopper.

Alternatively, you could use the URL to channel the customer service request to specific destinations. The URL contains the Net.Commerce command name and the parameters being used to service the page that the shopper is currently viewing. For example, if a shopper requests assistance while viewing their order information page (the URL contains the `OrderDisplay` command), you can use this information to direct the shopper to an order management specialist. Using JavaScript, you can quickly and easily locate the page in the store from which the shopper requests assistance.

The following is an example of how to retrieve the shopper's location in the shopping flow:

1. In a text or HTML editor, open the file `foot_6th.htm`
2. In the following sample, the JavaScript shown in **boldface** provides the current URL:

```
<SCRIPT Language="JavaScript">  
  
var loc= window.location.href;  
var param='$(CSURL)';
```



```

var shopping_bkt='$(CSORDERS)';
document.writeln("<A HREF=http://w3.ibm.com?url=" + escape(loc) + "&" +
escape(param) + "&" + escape(shopping_bkt) + ">ADD CS LOGO</A>");

</SCRIPT>

```

4.3. Access Shopper Profile Information

Shopper profile information helps CSRs learn more about shoppers while they are interacting with shoppers and allows for more personalized interaction. You can append the shopper profile information to the customer service URL as a name-value pair as shown in the `foot_6th.htm` sample.

Net.Commerce stores shopper profile information in the database tables Shopper Profile (SHOPPER) and Shopper Address Book (SHADDR) when a shopper registers with a store. This kit includes the shopping flow macros for Sixth Avenue with the sample `CS_SQL()` SQL that retrieves shopper profile information from the Net.Commerce database for customer service purposes. The `CS_SQL()` function builds a CSURL variable and inserts the results of the SQL in the desired format.

You can access shopper profile information from anywhere in the shopping flow by implementing the sample SQL in each of the Sixth Avenue shopping flow macros. See section 4.6, “Sixth Avenue File Mapping to the Net.Commerce Directory Structure” for a complete list of the macros that display the shopping flow pages. You can also customize `CS_SQL()` in each of the shopping flow macros to include additional shopper profile information from the database, or to remove unnecessary information from the sample. Your additions are limited only to the information stored in the Net.Commerce database.

The statement `@CS_SQL()` must be included in the main `HTML_INPUT`, or `HTML_REPORT` section of each macro to invoke `CS_SQL()`. See section 4.5, “Invoke the Shopper Profile and Shopping Basket SQLs” for an example of invoking this function. If you replace `CS_SQL()` with a different SQL, you must add a call to execute the SQL statement you added. If you replace the CSURL variable in the JavaScript sample shown in section 4.2, “Determine the Shopper’s Current Activity,” you must also update the line `var param='$(CSURL)';` by replacing CSURL with the name of your new variable.

CS_SQL() retrieves the following shopper profile information in the given sequence:

Table 1: CS_SQL() Information

Database Field Name	Description
satitle	Shopper's title (Not available, Mr., Mrs., Ms., or Dr.)
safname	First name
salname	Last name
saaddr1	Address line 1
saaddr2	Address line 2
sacity	City
sastate	State
sacntry	Country

The following is a sample of CS_SQL():

```
%function(dtw_odbc) CS_SQL(){
    select satitle,safname, salname, saaddr1,saaddr2, sacity, sastate,sacntry
    from shopper,shaddr
    where shrfnbr=sashnbr and saadrflg='P' and shlogid='$(SESSION_ID)'
%REPORT{
    @DTW_assign(CSURL,"")
    %ROW{
        @DTW_Concat("title=",V1,CSURL)
        @DTW_Concat(CSURL,"&fname=",CSURL)
        @DTW_Concat(CSURL,V2,CSURL)
        @DTW_Concat(CSURL,"&lname=",CSURL)
        @DTW_Concat(CSURL,V3,CSURL)
        @DTW_Concat(CSURL,"&addr1=",CSURL)
        @DTW_Concat(CSURL,V4,CSURL)
        @DTW_Concat(CSURL,"&addr2=",CSURL)
        @DTW_Concat(CSURL,V5,CSURL)
        @DTW_Concat(CSURL,"&city=",CSURL)
        @DTW_Concat(CSURL,V6,CSURL)
        @DTW_Concat(CSURL,"&state=",CSURL)
        @DTW_Concat(CSURL,V7,CSURL)
        @DTW_Concat(CSURL,"&country=",CSURL)
        @DTW_Concat(CSURL,V8,CSURL)
    }
%}
%MESSAGE{100:{ %} :continue %}
%}
```

Note: If you are integrating with a Store Creator store model, the information above applies to the three navigation bar macros, instead of the shopping flow macros.

4.4. Access Shopping Basket Information

Shopping basket information can help CSRs solve order problems while shoppers are placing orders. You can append shopping basket information as name-value pairs to the customer service URL.

Shopping basket information is stored in the following Net.Commerce database tables: Shopping Cart (SHOPPINGS), Product (PRODUCT), Merchant Profile, (MERCHANT), Shopper (SHOPPER), Shopper Address Book (SHADDR), and Shipping Association (SHIPTO). This kit includes the shopping flow macros for 6ixth Avenue with the sample SQL `CS_ShoppingBasket()` that retrieves shopping basket information from the Net.Commerce database. `CS_ShoppingBasket()` builds a `CSORDERS` variable with the results of the SQL.

You can access shopping basket information from anywhere in the shopping flow by implementing the sample SQL in each of the 6ixth Avenue shopping flow macros. See section 4.6, “6ixth Avenue File Mapping to the Net.Commerce Directory Structure,” for a complete list of the macros that display the shopping flow pages. You can also customize `CS_ShoppingBasket()` in each of the shopping flow macros to include additional shopping basket information from the Net.Commerce database, or to remove unnecessary information collected by the sample `CS_ShoppingBasket()`. For example, if you wish to retrieve only items from pending orders, edit the where clause in the SQL to say `and ststat=P`. Your additions are limited only to the information stored in the Net.Commerce database.

The statement `@CS_ShoppingBasket()` must be included in the main `HTML_INPUT`, or `HTML_REPORT` section of the macro to invoke `CS_ShoppingBasket()`. See section 4.5, “Invoke the Shopper Profile and Shopping Basket SQLs” for an example of invoking this function. If you replace `CS_ShoppingBasket()` with a different SQL, you must add a call to execute the SQL. If you replace the `CSORDERS` variable in the JavaScript sample, you must also update the line `var shopping_bkt='$(CSORDERS)';` by replacing `CSORDERS` with the name of your new variable.

CS_ShoppingBasket() retrieves the following shopping basket information about all orders placed by the shopper, in the given sequence, from the database:

Table 2: CS_ShoppingBasket SQL() Information

Database Field Name	Description
mestname	Name of the merchant.
sbprice	Price of the shipment.
sbpcur	Currency value for the shipment price.
prnrbr	SKU number.
prfnbr	Product reference number.
prsdsc	Product description.
sanick	Nickname of the person the shipment will be delivered to.
stquant	Quantity of the item purchased.
ststat	State of the shipment.

Note: The start of a new order is identified by the `order_num` variable which is the first name in the name-value pairs. This is the unique order number and allows the CSR to differentiate shopping basket information between different orders.

The following is an example of CS_ShoppingBasket():

```
%function(dtw_odbc) CS_ShoppingBasket(){
  select distinct mestname, sbprice, sbcpcur, prnbr, prrfnbr, prsdesc,
  sanick, stquant, ststat
  from SHOPPINGS, product, merchant, shopper, shaddr, shipto
  where (sbmenbr=merfnbr) and (sbprnbr=prrfnbr) and
        (sbshnbr=shrfnbr) and (shlogid = '$(SESSION_ID)') and
        (shrfnbr=sashnbr) and (saadrflg = 'P')
        and (stsanbr=sarfnbr) and
        stprnbr=prrfnbr and stshnbr=shrfnbr
%REPORT{
  @DTW_assign(CSORDERS, "")
  %ROW{
    @DTW_Concat(CSORDERS, "&order_num=", CSORDERS)
    @DTW_Concat(CSORDERS, ROW_NUM, CSORDERS)
    @DTW_Concat(CSORDERS, "&merchant=", CSORDERS)
    @DTW_Concat(CSORDERS, V1, CSORDERS)
    @DTW_Concat(CSORDERS, "&price=", CSORDERS)
    @DTW_Concat(CSORDERS, V2, CSORDERS)
    @DTW_Concat(CSORDERS, "&currency=", CSORDERS)
    @DTW_Concat(CSORDERS, V3, CSORDERS)
    @DTW_Concat(CSORDERS, "&SKU=", CSORDERS)
    @DTW_Concat(CSORDERS, V4, CSORDERS)
    @DTW_Concat(CSORDERS, "&product_num=", CSORDERS)
    @DTW_Concat(CSORDERS, V5, CSORDERS)
    @DTW_Concat(CSORDERS, "&desc=", CSORDERS)
    @DTW_Concat(CSORDERS, V6, CSORDERS)
    @DTW_Concat(CSORDERS, "&sent_to=", CSORDERS)
    @DTW_Concat(CSORDERS, V7, CSORDERS)
    @DTW_Concat(CSORDERS, "&quantity=", CSORDERS)
    @DTW_Concat(CSORDERS, V8, CSORDERS)
    @DTW_Concat(CSORDERS, "&state=", CSORDERS)
    @DTW_Concat(CSORDERS, V9, CSORDERS)
  }
  %}
  %}
  %MESSAGE{100:{ %} :continue %}
  %}
}
```

Note: If you are integrating with a Store Creator store model, the information above applies to the three navigation bar macro files, instead of all the shopping flow macros.

4.5. Invoke the Shopper Profile and Shopping Basket SQLs

To invoke the SQL function defined, ensure that the statement @CS_SQL(), @CS_ShoppingBasket(), or your own customized SQL function is included in the main HTML_INPUT or HTML_REPORT section of the macro.

The following is an example (shown in **boldface**) of invoking `CS_SQL()`:

Also, to pass the information retrieved from the SQLs processed above, change the JavaScript section in foot_6th.htm. As shown below the param and shopping_bkt parameters contain the information saved in CSURL and CSORDERS parameters, respectively:

```
<SCRIPT Language="JavaScript">

var loc= window.location.href;
var param='$(CSURL)';
var shopping_bkt='$(CSORDERS)';
document.writeln("<A HREF=http://w3.ibm.com?url=" + escape(loc) + "&" +
escape(param) + "&" + escape(shopping_bkt) + ">ADD CS LOGO</A>");

</SCRIPT>
```

4.6. Sixth Avenue File Mapping to the Net.Commerce Directory Structure

The table below describes the files included in this kit and the directories in which they are located for the sample Sixth Avenue integration.

Table 3: Sixth Avenue Integration File Mapping

File Name	Page Description	Metropolitan Mall Directory
HTML Pages		
foot_6th.htm	Sixth Avenue footer page	NetCommerce3/html/en_US/demomall
footmall.htm	Metropolitan mall footer page	NetCommerce3/html/en_US/demomall
Macros		
cat_clo1.d2w	Sixth Avenue clothing category	NetCommerce3/macro/en_US/category
cat_clo2.d2w	Sixth Avenue clothing category	NetCommerce3/macro/en_US/category
cat_clo3.d2w	Sixth Avenue clothing category	NetCommerce3/macro/en_US/category
cat_com1.d2w	Sixth Avenue computer category	NetCommerce3/macro/en_US/category
cat_com2.d2w	Sixth Avenue computer category	NetCommerce3/macro/en_US/category
cat_com3.d2w	Sixth Avenue computer category	NetCommerce3/macro/en_US/category
cat_ha1.d2w	Sixth Avenue hardware category	NetCommerce3/macro/en_US/category
cat_ha2.d2w	Sixth Avenue hardware category	NetCommerce3/macro/en_US/category
cat_ha3.d2w	Sixth Avenue hardware category	NetCommerce3/macro/en_US/category
tempclo2.d2w	Sixth Avenue clothing products	NetCommerce3/macro/en_US/product
tempclo3.d2w	Sixth Avenue clothing products	NetCommerce3/macro/en_US/product
tempclot.d2w	Sixth Avenue clothing products	NetCommerce3/macro/en_US/product
tempcomp.d2w	Sixth Avenue computer products	NetCommerce3/macro/en_US/product
temphard.d2w	Sixth Avenue hardware products	NetCommerce3/macro/en_US/product
tempite2.d2w	Sixth Avenue products	NetCommerce3/macro/en_US/product
tempitem.d2w	Sixth Avenue products	NetCommerce3/macro/en_US/product
orderspp.d2w	Pending order details	NetCommerce3/macro/en_US/ncsample
shptodsp.d2w	Shipping details	NetCommerce3/macro/en_US/ncsample
shptolst.d2w	Prepare orders	NetCommerce3/macro/en_US/ncsample

4.7. Provide Integration Information to Store Administrators

To integrate your customer service application with an existing Net.Commerce store, you must provide the necessary information to Net.Commerce Store Administrators so that they can customize existing HTML pages and macros that comprise their shopping flow. The following is a list of the information you must provide to Store Administrators so that they can integrate with your application:

- Your application URL and logo
- Customized versions of `CS_SQL()` and `CS_ShoppingBasket()` that contain information specific to your application
- Your customized version of the footer page containing the JavaScript required to process your application URL and logo, and any modifications to the `CSURL` and `CSORDERS` variables
- Instructions on how to implement your customized integration
- Any description of your extended customer service solution

Note: You may also want to ensure that Store Administrators are aware that this integration depends on the link to your customer service application or service being intact.

5. Integrate with a Store Creator Store Model

The technique for integrating with a Net.Commerce Store Creator store model is similar to integrating with an existing store, except that only one macro for each store model must be modified to complete the integration. There are three types of store models from which Store Administrators can create their stores: One Stop Shop, Personal Delivery, and Business-to-Business. Each store model is generated by a separate set of template macros. One of the template macros creates a navigation bar that appears on each page of the store. The navigation bar macros in this kit already include modifications to Net.Data macros that integrate with your customer service solution, and only require that you include your customer service link to complete the integration. SQL statements and JavaScript statements append the data retrieved from processing the SQLs as parameters to the customer service URL.

Once you have integrated with the store models, Store Administrators only need to overwrite their existing navigation bar macros with your customized versions and create their new, integrated stores. For more information about the store models, see the “Store Creator” section in the Net.Commerce online help.

To integrate with a store model, you must create a customer service link. You can further customize the existing JavaScript and SQLs which integrate the following service options with the store models:

- Determine the shopper’s current activity
- Access shopper profile information (See section 4.3, “Access Shopper Profile Information”)
- Access shopping basket information (See section 4.4, “Access Shopping Basket Information”)

5.1. Create a Customer Service Link

Store model macros contain a navigation bar that is appended to all pages of the newly created store. This bar appears at the bottom, top or side of the store pages. The sample navigation bar macros in this kit, `nav_bottom.htm`, `nav_side.htm`, and `nav_top.htm`, contain JavaScript in which you can include your customer service logo and a link to your application or service.

To create the customer service link JavaScript, do the following:

1. In a text editor or HTML editor, open the following files from the following directories:

- One Stop Shop - `nav_side.d2w`
`NetCommerce3/SmartGuide/OneStop`
- Personal Delivery - `nav_top.d2w`
`NetCommerce3/SmartGuide/PDelivery/nav_top.d2w`
- Business-to-Business - `nav_bottom.d2w`
`NetCommerce3/SmartGuide/BtoB/nav_bottom.d2w`

2. In the following sample link (text that needs to be modified is shown here in **boldface**):

```
<A HREF=http://w3.ibm.com?url=" + escape(loc) + "&" +  
escape(param) + "&" + escape(shopping_bkt) + ">ADD CS LOGO</A>
```

- a. Modify the URL to that of your customer service application or service.
- b. Modify the text to include your customer service logo.

3. Save the file.

Note: For every link selected from the navigation page, the following JavaScript keeps the navigation frame and the customer service logo current. This must be in all the navigation template macros:

```
%HTML_REPORT {  
<SCRIPT Language="JavaScript">  
function reloadframe() {  
    parent.navbar.location.reload(true);  
    return true;  
}  
</SCRIPT>
```

5.2. Determine the Shopper's Current Activity

You may want to know which page the shopper was viewing or what activity the shopper was attempting to perform when they clicked on your customer service link. Append the current URL as a name-value pair to your customer service application URL. A CSR can then view the page simultaneously with the shopper. The navigation bar macros in this kit already include JavaScript which lets you quickly and easily locate the page in the store from which the shopper requests assistance. If you wish to customize the Shopper Profile and Shopping Basket SQLs, you may also need to edit this JavaScript.

To customize this JavaScript, do the following:

1. In a text editor or HTML editor, open the following files from the specified directories:
 - **One Stop Shop** - `nav_side.d2w` in
`NetCommerce3/SmartGuide/OneStop`
 - **Personal Delivery** - `nav_top.d2w` in
`NetCommerce3/SmartGuide/PDelivery/nav_top.d2w`
 - **Business-to-Business** - `nav_bottom.d2w` in
`NetCommerce3/SmartGuide/BtoB/nav_bottom.d2w`
2. In the following samples, the JavaScript shown in **boldface** provides the current URL for each of the store models used:

a. **One Stop Shop**

```
<SCRIPT Language="JavaScript">  
var loc= parent.main.location.href;  
var param='$(CSURL)';  
var shopping_bkt='$(CSORDERS)';  
document.writeln("<A HREF=http://w3.ibm.com?url=" + escape(loc) +  
"&" +  
escape(param) + "&" + escape(shopping_bkt) + "><FONT SIZE=1><B>Add  
CS Logo</FONT/></B></A>");  
</SCRIPT>
```

b. **Personal Delivery**

```
<SCRIPT Language="JavaScript">  
if (top.main.length == 0)  
  var loc = top.main.location.href;  
else var loc= top.main.maincat.location.href;  
var param='$(CSURL)';  
var shopping_bkt='$(CSORDERS)';  
document.writeln("<A HREF=http://w3.ibm.com?url=" + escape(loc) +  
"&" +  
escape(param) + "&" + escape(shopping_bkt) + "><FONT SIZE=1><B>Add  
CS Logo</FONT/></B></A>");  
</SCRIPT>
```

c. **Business-to-Business**

```
<SCRIPT Language="JavaScript">  
var loc = top.main.location.href;  
var param='$(CSURL)';  
var shopping_bkt='$(CSORDERS)';  
document.writeln("<A HREF=http://w3.ibm.com?url=" + escape(loc) +  
"&" +  
escape(param) + "&" + escape(shopping_bkt) + "><FONT SIZE=1><B>Add  
CS Logo</FONT/></B></A>");  
</SCRIPT>
```

For the Business-to-Business store model, this JavaScript is also included in the user registration type sections. This will cause the navigation bar to change depending on

what type of user the shopper is. You can copy and paste the JavaScript for additional user types. The following user types already include this JavaScript:

- R - Registered user
 - M - Merchant
 - A - Administrator
 - Else - All others
3. If you want to determine a shopper's location, you must include your customer service logo.
 4. If you want to customize the Shopper Profile and Shopping Basket SQLs, and you rename the shopper profile and shopping basket parameters, you must modify the CSURL and CSORDERS variable names for the store models you wish to customize.
 5. Save the file.

5.3. Provide Integration Information to Store Administrators

To enable Store Administrators to create new stores from Store Creator store models that have been integrated with your customer service solution, you must provide them with all of the customized macros. The following is a list of the information you must provide to Store Administrators:

- Customized store model navigation template macros `nav_side.d2w`, `nav_top.d2w`, and `nav_bottom.d2w`
- Customized versions of `CS_SQL()` and `CS_ShoppingBasket()` that contain information specific to your application
- Any description of your extended customer service solution

6. Files and Technical Notes

The following is a summary of files included in this kit:

File Name	Technical Design Notes
foot_6th.htm	Sixth Avenue footer.
footmall.htm	Metropolitan mall footer.
cat_clo1.d2w	Sixth Avenue clothing category.
cat_clo2.d2w	Sixth Avenue clothing category.
cat_clo3.d2w	Sixth Avenue clothing category.
cat_com1.d2w	Sixth Avenue computer category.
cat_com2.d2w	Sixth Avenue computer category.
cat_com3.d2w	Sixth Avenue computer category.
cat_ha1.d2w	Sixth Avenue hardware category.
cat_ha2.d2w	Sixth Avenue hardware category.
cat_ha3.d2w	Sixth Avenue hardware category.
tempclo2.d2w	Sixth Avenue clothing products.
tempclo3.d2w	Sixth Avenue clothing products.
tempclot.d2w	Sixth Avenue clothing products.
tempcomp.d2w	Sixth Avenue computer products.
temphard.d2w	Sixth Avenue hardware products.
tempite2.d2w	Sixth Avenue products.
tempitem.d2w	Sixth Avenue products.
orderspp.d2w	Pending order details.
shptodsp.d2w	Shipping details
shptolst.d2w	Prepare orders.
nav_side.d2w	One Stop Shop navigation frame.
nav_top.d2w	Personal Delivery navigation frame.
nav_bottom.d2w	Business-to-Business navigation frame.
CustomerService.doc	Customer Service Kit documentation.

Appendix A: Notices and Trademarks

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

This document may contain information about other companies' products, including references to such companies' Internet sites. IBM has no responsibility for the accuracy, completeness, or use of such information.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM International Program License Agreement for Non-Warranted Programs.

COPYRIGHT LICENSE:

This document contains information including sample application programs in source language that illustrate programming techniques on various operating platforms. You may use the sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. The sample programs have not been tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States or other countries or both:

IBM AIX DB2

Net.Data

Microsoft, Windows, and Windows NT are trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries.

Solaris Operating Environment, Java and all Java-based trademarks and logos are trademarks or registered trademark of Sun Microsystems, Inc. in the United States and/or other countries.

Other company, product, and service names, may be the trademarks or service marks of others.

Credit card images, trademarks and trade names provided in this product should be used only by merchants authorized by the credit card mark's owner to accept payment via that credit card.