eCommerce User Guide

LS Nav – 2013 (7.1)
Contents
1 Introduction ........................................................................................................................................... 1
  1.1 System Requirements .................................................................................................................... 2
2 Functionality Overview ..................................................................................................................... 3
  2.1 The Menu Suite ............................................................................................................................. 4
3 Step by Step eCommerce Setup ......................................................................................................... 5
4 Web Store Setup in LS Nav ................................................................................................................ 6
  4.1 Designating a Web Store ................................................................................................................ 6
  4.2 Currency and Price Display in the eCommerce Web Store ......................................................... 7
    4.2.1 Currency Code ......................................................................................................................... 7
    4.2.2 The Currency Symbol ........................................................................................................... 8
    4.2.3 Price Display in the eCommerce Web Store ........................................................................ 9
    4.2.4 Web Store Display Locale ................................................................................................... 10
5 LS Nav and LS Omni .......................................................................................................................... 11
  5.1.1 Creating a nopProduct ............................................................................................................ 13
  5.2 Product Default Values ................................................................................................................ 16
    5.2.1 Default Values Hierarchy ..................................................................................................... 19
    5.2.2 Primary Variant Dimension ................................................................................................. 22
    5.2.3 Web Store Presentation Type ............................................................................................... 24
    5.2.4 nopProduct Web Store Description .................................................................................... 26
  5.3 Product Prices ............................................................................................................................... 27
  5.4 Images for nopProduct and Attributes ..................................................................................... 28
  5.5 Images for Item Categories and Product Groups ......................................................................... 36
  5.6 Jobs .................................................................................................................................................. 37
  5.7 Sales Orders .................................................................................................................................... 37
    5.7.1 Shipping Requirements ......................................................................................................... 38
    5.7.2 Out of Stock Orders ............................................................................................................. 39
    5.7.3 Cancelling Orders ................................................................................................................ 39
6 User Buying Goods in the Web Store ............................................................................................... 40
7 eCommerce on a Mobile ..................................................................................................................... 47
  7.1 Front Page and Login .................................................................................................................... 47
  7.2 Products for eCommerce on a Mobile .......................................................................................... 48
     Many Products ............................................................................................................................... 48
     One Product .................................................................................................................................... 49
7.3 Shopping Cart ............................................................................................................................................. 50
7.4 Purchasing Items and Viewing Orders on a Mobile POS ................................................................. 51
8 Appendix A – Web Services – Requests and Responses ............................................................. 52
  8.1 WI_NC_CALCULATE_BASKET – Request ......................................................................................... 52
  8.2 WI_NC_CALCULATE_BASKET – Response ...................................................................................... 53
  8.3 WI_NC_CREATE_ORDER – Request .................................................................................................. 55
  8.4 WI_NC_CREATE_ORDER – Response .............................................................................................. 56
  8.5 WI_NC_ORDER_STATUS – Request ................................................................................................ 57
  8.6 WI_NC_ORDER_STATUS - Respond ................................................................................................. 57
1 Introduction

With the LS eCommerce solution it is now possible to create a web store based on LS Nav. This web store can be accessed from computers, pads and mobile phones. The LS eCommerce solution, from here on often referred to as the eCommerce solution, splits into two parts, the standard LS Nav solution and the LS Omni eCommerce Web Store Solution, from here on often referred to as LS Omni, or LS Omni Web Store. The latter solution supports the display and appearance of the web store, but the LS Nav solution handles all retail configuration and Back Office functionality of the store. The focus of this guide is on the LS Nav part and, where needed, the interaction between LS Nav and the LS Omni eCommerce solution.

The LS Retail eCommerce solution meets the increasing demand for online shopping. It offers management of in-store, mobile commerce and online store systems. Offers can be presented online and in-store. The solution makes it possible to have one solution on different operating systems.

This user guide assumes that the user has basic knowledge of LS Nav functionality. The aim of the guide is to fill in the additional detail to the LS Nav and LS Omni systems needed to administrate the web store.

The following document has been added to the standard release documents and is a useful supplement to this document:

- eCommerce Data Mapping Details LS Omni and LS Nav 2013 (7.1)

The Omni part of the system is documented in a separate User Guide and Release Notes. We recommend that you also read these documents for better understanding of the system as a whole:

- LS Omni eCommerce 1.0 Install and User Guide
- LS Omni eCommerce 1.0 Release Notes

The above documents are available on http://partners.lsretail.com for registered users.
### 1.1 System Requirements

**Operating systems supported**
- Windows 7 / 8, Windows Server 2008 / 2012
- IIS 7.0 or above must be installed
- ASP.Net 4.5 and MVC 4 must be installed on the IIS machine.
  - [http://www.asp.net/mvc/mvc4](http://www.asp.net/mvc/mvc4)

**Databases supported**
- SQL Server 2005 or above

**Browsers supported**
- Internet Explorer 6 and above
- Firefox 2.0 and above
- Chrome 1.x
- Safari 2.x
2 Functionality Overview

The basics of the interaction between Nav and the LS Omni eCommerce Web Store part of the solution can be described in the following overview:

All the LS Omni eCommerce Web Store administrative work, both configurative and retail-specific, takes place in Nav. The structure of the Web Store Products is different from the structure of the Nav Retail Item. A set of buffer tables containing configuration and product information is generated by Nav. These tables form a layer between Nav and the LS Omni eCommerce web store. The data in these tables is replicated between the two system databases, as seen on the image. Please note that NOP (NOPCommerce) is often used when referring to the LS Omni eCommerce solution. For example nopProduct, nop order number, nop database and so on.
2.1 The Menu Suite

The Web Integration part of the LS Retail Menu Suite contains pages and tables that handle all configuration specific for the LS Nav part of the LS eCommerce solution, at LS Retail, Web Integration.

![Web Integration Diagram]
3 Step by Step eCommerce Setup

The purpose of this chapter is to give the user a quick overview of the tasks needed to get the LS eCommerce solution up and running. A detailed description of each step is given further along in the document.

1. **Ensure that LS Nav 2013 (7.1) has been installed.**

2. **Ensure that the LS Omni eCommerce solution is installed and configured**, see install and configuration guide for eCommerce on the LS Retail Partner Portal:


3. **Set up your Web Store in LS Nav**, see chapter below, *Web Store Setup in LS Nav*.

4. **Make sure that there are retail items in the web store** via the Distribution Group containing the web store, see chapter on nopProduct.
   a. Items are distributed to the web store the same way as in a traditional LS Nav store.

5. Open **LS Retail > WI Integration > Administrations > WI Setup > Setup** and run the action **Create/Update Data for Replication**, (see the chapter on nopProduct). By running this process you populate the buffer tables mentioned above with information about the web store’s nopProducts based on the retail items that have been distributed to the web store.

6. **Ensure that jobs and web services exist in NAV.** The Web services setup is found in an Appendix in this document. The jobs, subjobs and relevant setup can either be configured manually or generated automatically by running the Page **WI Default Data** (no. 10012868). See separate document about LS Omni eCommerce Data Mapping Details.

7. **Set up your Distribution Locations.** The From and To Locations for the jobs generated in previous step need to contain the correct location information. Remember to test these locations.

8. **Run the job E-COMMERCE, NORMAL.** It might take several minutes to run this job depending on the number of retail items in the web store and number of related images.
4 Web Store Setup in LS Nav

4.1 Designating a Web Store

A store is selected to be a Web Store via WI Setup in LS Retail > Web Integration > Administration > Setup.

As for any other web integration in LS Nav, WI in Use needs to be check-marked. On the ECommerce fast tab, the Web Store Code, Web Store POS Terminal, Web Store Staff ID and Web Store Customer No. need to be filled in for a successful setup – that is:

- **Web Store Code** is used to assign a store to be the web store. The web store is selected from the list of all stores.

- **Web Store POS Terminal** is used to assign a terminal to the web store. It is selected from the list of POS terminals assigned to the store. The Web Store POS Terminal must be of the Terminal Type **POS Terminal**. If a terminal of the type Mobile POS is selected, an error message is displayed.

- **Web Store Customer No.** is used to assign a customer to the web store so that standard NAV Sales Orders can be created when sales takes place in the web store. This customer should only be used for this purpose.

The **Web Store Last Action Entry No:** holds the reference entry for the Scheduler job ECOMMERCE, SCHED that is discussed in more detail in the chapters Creating a nopProduct and Jobs below. This refers to an entry number in the Preaction table or the Action table depending on the Scheduler setup for the job.

The **Web Store Display Local** field is used for currency definition. It is explained in detail in the following chapter.
4.2 Currency and Price Display in the eCommerce Web Store

The LS Omni Web Store must be provided with detailed information on currency from LS Nav.

💡 In the first versions of the solution, only one type of currency is allowed in the Web Store of the eCommerce solution.

4.2.1 Currency Code

The currency that is used to display prices in the web store is the currency that the Currency Code selected on the General tab of the store card refers to: LS Retail > BackOffice > Store Management > Lists > Stores, select the web store.

If a currency code is not selected on the store card, the currency according to the LCY Code from the General Ledger Setup is used.
4.2.2 The Currency Symbol

The placement of the currency symbol can be decided at the LS Retail > BackOffice > Store Management > Lists > Retail Currencies, select Currency > POS Terminal fast tab, and edit the field POS Currency Symbol.
4.2.3 Price Display in the eCommerce Web Store

The placement of the Currency Symbol in prices in the web store can be decided at LS Retail BackOffice > Store Management > Lists > Retail Currencies, select Currency > POS Terminal fast tab.

![Currency Symbol Placement Configuration](image)

For example if the price is 100 EUR and Placement Of Currency Symbol = After the Amount, then the web store displays 100 € and if one chooses Placement Of Currency Symbol = Before the Amount then the amount is displayed as € 100.

When it has been decided which stores are Web Stores, the path LS Retail > Web Integration > Administration > Setup is entered.
4.2.4 Web Store Display Locale

The **Web Store Display Locale** is set at **WI Setup** at **LS Retail > Web Integration > Administration > WI Setup > Setup**.

The field **Web Store Display Locale** decides the currency used in the Web Store. It is a code consisting of 2 lower case letters and 2 upper case ones connected with a “.”. By default this field is empty. An example of this value would be **en-US** or **Ja-Jp** representing English speaking US and Japanese speaking Japan respectively.
5   LS Nav and LS Omni

Under the group **LS Retail > Web Integration > Administration > Web Store** there are several features. This group handles all settings and configurations for the nopProducts for the Omni eCommerce Web Store that are irrelevant to the associated standard LS Retail Item.

This section describes the creation and modification of a nopProduct, and the interaction between it and the associated Retail Item.

As concerns the LS Nav infrastructure and functionality, the LS Omni eCommerce Web Store is no different from any other store with regard to manipulating a Retail Item. In LS Nav, the Distribution Groups decide in which stores an Item is sold. A Retail Item can be sold in more than one store.
5.1.1 Creating a nopProduct

The information related to the nopProducts is kept in a set of tables that form a layer between the standard LS Nav solution database and the LS Omni eCommerce Solution database, referred to as the nopDatabase from here on. The information from these tables is then replicated to the LS Omni Database. Hence, when you want to create or update a nopProduct, relevant entries in the middle layer data tables need to be created or modified accordingly. Under the group **LS Retail > Web Integration > Administration > Web Store > Data Tables**, the administrator can view the middle layer data tables.

The nopProduct information can come from two data sources, from the associated Retail Item, or the **LS Retail > Web Integration > Administration > Web Store > Product Default Values** page. The functionality of the **Product Default Values** page is discussed in detail in next section.

Two Codeunits are linked to the process of creating and updating a nopProduct. A codeunit that creates and updates the middle layer tables to be replicated to the LS Omni nopDatabase, and another codeunit that clears all information from these middle layer tables.

The codeunit (10012865) **WI NC Test** deletes all entries for all the nopProducts in the middle layer tables. That is; all the data that is replicated into the NOP database. The codeunit can only be accessed as the action **LS Retail > Web Integration > Administration > WI Setup > Setup > Clear Data for Replication.**
The rest of this section deals with the codeunit that creates and updates the nopProduct data in the middle layer tables.

The middle layer tables are empty in the Demo Data. The user needs to run the codeunit that creates the nopProduct data to populate the tables prior to running the normal job.

The nopProduct data is generated via codeunit (10012864), **WI NC Mgt**. This codeunit can been run as follows:

1. As the action, **Create/Update Data for Replication**, from **LS Retail > Web Integration > Administration > WI Setup > Setup**.
By running the codeunit here via the action Create/Update Data for Replication, all nopProducts in the web store are considered. That is; the codeunit loops through all Retail Items and for each item assigned to the store's distribution group, associated nopProduct data is created or updated. This is done both with regard to new or changed Retail Item data, and with regard to change in nopProduct configuration in Product Default Values. That is; the codeunit writes into the middle layer tables, the data of which is then replicated into the nopDatabase. The user gets a message when the codeunit run has finished. The idea is that on setup the administrator runs the codeunit from here to create the initial group of nopProducts.

2. Using Scheduler job, creating actions on nopProducts that are new or where data has changed. When the Codeunit is run by a Scheduled job, it updates the middle layer tables by the following actions/pre-actions according to the WI Setup, Web Store Last Action Entry No: on the underlying retail item data tables:

   (27) Item
   (5401) Item Variant
   (10000704) Item Distribution
   (10001412) Extended Variant Dimensions
   (10001413) Extended Variant Values
   (10001414) Item Variant Registration

   Note, that changes in the Product Default Values create pre-actions on the Item table. For more information on the scheduled job, see the section on Job.
3. From the Retail Item Card, only updating the nopProduct that mirrors the selected Retail Item. Settings in Product Default Values related to this Retail Item are also considered here.

The middle layer tables are only populated with information relevant to the products in the web store. That is; the data load in them is minimal.

All data in the middle layer tables need to be replicated into the nopDatabase. To see the data change in the webpage, you might need to Clear Cache or Restart the Application in the nop Administrative tool.

5.2 Product Default Values

Under the group LS Retail > Web Integration > Administration > Web Store you find the Product Default Values page:

On this page, you can set a default value hierarchy for nopProducts. That is; one can:

1. Set default values for a nopProduct on an Item, Product Group or Item Category level
2. Select primary Variant Dimension for a nopProduct
3. Select default values for Web Store Presentation Type

The following subsections will describe these three configuration hierarchies in detail.

The following image shows the Product Default Values.

![Product Default Values](image)

The first seven fields in the first column of the Product Default Values card make the primary key of the table. Their values are discussed in detail in the following subsections. Below are comments on the values that are important for the remaining fields:

- **IsShipEnabled**: If this field is marked as “false”, you are not able to ship the product
- **AdditionalShippingCharge**: Can be added to the shipping price if this product is more expensive to ship than others
- **IsTaxExempt**: Used to decide if we want to calculate taxes or not, this is not supported in LS Nav
- **TaxCategoryId**: Used, not supported in LS Nav, should be set to Zero
- **DisplayStockAvailability**: If check-marked, stock availability is shown on homepage
- **DisplayStockQuantity**: If check-marked, stock quantity is shown on homepage
- **MinStockQuantity**: The quantity for which you would like to trigger an event like disable buy button
- **LowStockActivityId**:  
  - Nothing = 0  
  - DisableBuyButton = 1  
  - Unpublish = 2
- **BackOrderModeId**:  
  - No Backorder = 0  
  - AllowQtyBelow0 = 1  
  - AllowQtyBelow0AndNotifyCustomer = 2
- **OrderMinimumQuantity**: Minimum quantity that the customer can buy, default value should be 1
- **OrderMaximumQuantity**: Maximum quantity that the customer can buy, default value should be 100
- **DisableBuyButton**: Indicates whether the buy button should be disabled on the page or not
- **DisableWishlistButton**: Indicates whether the wish list button should be disabled on the page or not
- **AvailableForPreOrder**: Is the customer allowed to order the item if it is out of stock? Not supported in LS Nav now.
- **CallForPrice**: Price is not shown on the page, the customer gets a message that he needs to call the store for price
- **Weight, Length, Width and Height**: Information about the size of the package
- **Published**: Check-marked or not; check-marked to publish the product
- **Deleted**: If check-marked, the product is marked as deleted
- **ProductTemplateId**: Variant in Grid, Single Product Variant
5.2.1 Default Values Hierarchy

In this section we will describe the hierarchy that sets default values for nopProducts/LS Retail Items.

The primary key in the page's underlying table is “Item No., Item Sub Code, Item Category Code, Product Group Code, and Primary Variant Dimension”. All these fields can be empty, and if the table is empty, an entry with all key values empty is created in open page. Hence, when editing these fields in an entry, you get a message that a record is being renamed.

The value of the last field in the key, “Primary Variant Dimension” has no relevance to the hierarchy described here. However, it plays a key role in the second hierarchy mentioned above. Let us state the hierarchy from bottom down in the following table.

<table>
<thead>
<tr>
<th>Importance</th>
<th>Item No.</th>
<th>Item Sub Code</th>
<th>Item Category Code</th>
<th>Product Group Code</th>
<th>Primary Variant Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Value</td>
<td>Value</td>
<td>The Value related to the “Item No.”</td>
<td>The Value related to the “Item No.”</td>
<td>Possibly some value related to “Item No.”</td>
</tr>
<tr>
<td>2</td>
<td>Value</td>
<td>Empty</td>
<td>The Value related to the “Item No.”</td>
<td>The Value related to the “Item No.”</td>
<td>Possibly some value related to “Item No.”</td>
</tr>
<tr>
<td>3</td>
<td>Empty</td>
<td>Empty</td>
<td>Value</td>
<td>Some Value related to the “Item Category Code”</td>
<td>Empty</td>
</tr>
<tr>
<td>4</td>
<td>Empty</td>
<td>Empty</td>
<td>Value</td>
<td>Empty</td>
<td>Empty</td>
</tr>
<tr>
<td>5</td>
<td>Empty</td>
<td>Empty</td>
<td>Empty</td>
<td>Empty</td>
<td>Empty</td>
</tr>
</tbody>
</table>

*LS Nav assumes that the last line, no 5, exists. Remember this line is initially created on page open.*

The **Product Group Code** has no meaning if the **Item Category Code** has no value. Also, if the **Item No.** is not empty, these item values are filled in accordingly by default on creating an entry.

**Example:** **Item Sub Code** can take on any value of the item’s variant framework.
The subcodes for Item 40020, Linda skirt, have the variant dimensions COLOUR and SIZE, hence the possible values for “Item Sub Code” are the colors BLACK, GREEN, ORANGE, PINK, RED and YELLOW, and the sizes 34, 36, 38, 40, 42, 44 and 46. Hence, if you for example want to remove yellow skirts, or skirts size 46 in all colors you can create a template to do this by selecting appropriate item and subcode, and then setting the field Published as FALSE.
The next two images show how it is displayed when the product has the **Primary Variant Dimension** set to as **STYLE:**
Example: If you want free shipping for all items in women's clothing, you could make a template with Item Category Code CLOTHING and Product Group Code WOMEN-S and set the value of IsFreeShipping to TRUE.

5.2.2 Primary Variant Dimension

In this section, we will describe the configurations of the nopProduct’s Primary Variant Dimension. Towards the user, the Primary Variant Dimension has to do with the display of the product in the web store.

Example: Let us consider a nopProduct the associated Retail Item of which has three variant dimensions, COLOR, SIZE and STYLE. The following image shows how it is displayed as it has no Primary Variant Dimension.
The primary structural change between nop3 and nop3.1 is that a product does not necessarily need to have a primary variant.

The default is that a nopProduct does not have a Primary Variant Dimension.

If you want a nopProduct to have a primary variant, it is selected in the following way in the Product Default Values page:

1. Create/or edit a line for the Retail Item you have in mind.
2. Drill down into the Primary Variant Dimension field. A Lookup opens that displays the available variant dimensions according to the Retail Item’s Variant Framework.

On creating a new entry with empty Primary Variant Dimension, the default value for ProductTemplateId is Variant in Grid. However, if a nopProduct has a Primary Variant Dimension selected, it needs to have selected ProductTemplateId = Single Product Variant. Hence, when a Product Variant Dimension is selected, the ProductTemplateId is updated accordingly.

On creating a new entry in Product Default Values with a nonempty Item No. field, the fields for Product Group Code, Item Category Code and Variant Framework Code get the value from the item. If another entry exists for this item with Primary Variant Dimension selected, the new line gets the same value for that field. That is; two entries for same item need to have the same Primary Variant Dimension.

5.2.3 Web Store Presentation Type

The Web Store Presentation Type decides how a nopProduct’s attribute value (NAV variant dimension) is displayed in the web store.

You can set a default value for a nopProduct’s attribute value in all variant frameworks by creating an entry in the Product Default Values page with all key values empty except the Primary Variant Dimension field which is used here to define the attribute value. In the case where all primary key values are empty and you open the lookup for the Primary Variant Dimension, the lookup displays all possible values from table Extended Variant Dimensions (10001412).

Example: The framework in the following picture states that if a nopProduct has attribute value (that is; the Retail Item’s variant framework contains the following dimensions)
COLOR, its value will be selected via a drop down in the web store
SIZE, its value will be selected via Radio Buttons in the web store
STYLE, its value will be selected via Check Boxes in the web store

Example: If a nopProduct has attributes, but none of them stated in the Product Default Values page (the associated Retail Item has a variant framework that does not include these dimensions), then the attribute values are selected via its Web Store Presentation Type if there is an entry in Product Default Values for the item. Otherwise it is selected via the Web Store Presentation Type for the default framework (the one with all the values in the primary key empty).

In the image below we see an entry for item 52010. Its variant framework is FUEL, which does not have Web Store Presentation Type default value defined. Hence, its attributes are selected via a Radio Button. The entry for item 30085 is similar. Its variant framework SODA does not include COLOUR, SIZE or STYLE, hence its attributes are selected via checkbox.
On creating new entry, the **Web Store Presentation Type** is assigned the value of the **Web Store Presentation Type** in default framework with all the values in the primary key empty.

### 5.2.4 `nopProduct` Web Store Description

On the Retail Item Card, descriptions of each item can be added in an HTML editor. The editor is opened by selecting the Item HTML icon on the Navigate tab of a Retail Item Card for the item in question.
In the editor, the text is entered and previewed:

5.3 Product Prices

In NAV, prices are set on item or item’s variant level on the Retail Item card, the Pricing tab, via LS Retail > BackOffice > Product Design > Retail Items. Products in the web store are displayed with the item price except possibly in one case. This is when the item has one variant dimension, and that dimension has been selected as the Primary Variant Dimension. Hence, the item is displayed on variant level and the product is displayed with the variant price, if any. Otherwise it is displayed with the item price.
5.4 Images for nopProduct and Attributes

In LS Nav, images can be added to a Retail Item and its variants. This is done on the page (Retail Image Link List (99009082)). To access this page for Retail Item or Variants, start by opening the Retail Item card via **LS Retail > BackOffice > Product Design > Retail Items.** Then go to **Navigate > Master Data.** Adding an image to:

- **Retail Item** - Click on the Image icon. Then the Retail Image Link List page opens. There you can add image to the Retail Item. If you want to add more than one image, the **Display Order** decides the order in which they are displayed in the web stores. For example the one with the lowest **Display Order** will be in focus when the item is accessed in the web store.

- **Variants** - Select **Variants** from the **Variants**-drop down. The Item Variants page opens. There you select **Navigate > Images** and the Retail Image Link List page opens.

  The page allows the user to select more than one image, but the one used in these cases needs to have **Display Order = 0**, which is the default value. If more than one image has Display Order = 0 the first one in the list is used.

It is assumed that the user is selecting image for the first variant dimension group.
Example: You want to add an image to Retail Item 40020, Linda Skirt that has variant dimension one as COLOUR and variant dimension two as SIZE. Below we add a link to images for all variants that represent yellow skirts.

1. On the Retail Item Card at LS Retail > BackOffice > Product Design/Sales Marketing > Lists > Sales > Retail Items, select the Retail Item 40020, Linda Skirt.
2. On the Retail Item Card on the Home tab, select the Variants button. A list of variants for the product appears. Select one of the variants representing a yellow skirt.
3. An image for the variant can be selected by going to the Retail Image Link page, accessed on the Navigate tab, Images button.

4. To enter an image for a variant, press the New button on the Real Image Link List card. In an empty line in the Picture ID column you can access an image in the database and select the one you want to use. The selection will be valid when you go to another line but the image will only appear on the card the next time you enter it.

Note that when you enter the image initially it is assumed that the user is selecting an image for the first variant dimension. Example: Let's add an image to the variant Skirt Linda YELLOW/34, Variant Code 000. Then the system suggests that the user adds the same image to all variants in the dimension YELLOW.
5. If the user deletes the image from one of the variants, the system suggests that the image is deleted for all the variants in the dimension.

A nopProduct that is an associated Retail Item has variants. The images linked to the variants are used to visualize the nopProduct attributes that correspond to the first variant dimension in the Retail Items variant framework. The images are displayed in the following way:

**nopProduct with no attributes:** The images selected for the item are displayed as before. The one with the lowest **Display Order** is in focus, and the other ones are displayed smaller, according to the display order.
nopProduct with attributes: If there are images linked to the item's variant, the ones linked to the first dimension in the variant framework are assigned to the relevant attribute. These images are then also linked to the nopProduct behind the scenes in LS Nav. The result in the web store is that the images are displayed as for a nopProduct with no attributes, (see the example above), but when an attribute value in the corresponding first variant dimension is selected, the image in focus changes to the one that is linked to that value.
Example: Suite Davi-s Professional Wear (Retail Item 40060) has COLOUR as its first variant dimension. Hence if one selects a value for the attribute COLOUR, the image in focus changes. Here, BLUE has been selected and the image of blue suits is in focus:
In the example below, however, BLACK has been selected and the image with black suite are in focus:

![Image of a man wearing a black suit]

Variants of the product:

- **COLOUR**: BLACK
- **SIZE**: 44, 46, 48, 50, 52
- **STYLE**: S01, S02

**Variant nopProduct with Primary Variant**: As for nopProduct with no attributes, the images linked to the Retail Item and the attributes are displayed at the top of the page in the web store alongside the general information about the nopProduct. Below, each value of the **Primary Variant Dimension** gets its section. If the **Primary Variant Dimension** is not selected as the first variant dimension, and there have been selected images for that dimension in NAV, the image will change when different values are selected.
**Example:** Here the **Primary Variant Dimension** is selected to be STYLE and the first variant dimension in the framework is COLOUR. That is, they are not the same. In the following image we see that for the **Primary Variant Dimension**-value STYLE S01, the color BLUE has been selected. Hence, the image for the attribute has changed to blue suit. Note, that you do not see all the images for the attributes as for the nopProduct itself.
5.5 Images for Item Categories and Product Groups

The user can add an image for **Item Categories** and **Product Groups**. This is done on the same page (Retail Image Link List (99009082)) that the user adds an image to a Retail Item and its variants. To open the Retail Image Link List and link image to:

**Item Category:** Go to LS Retail > BackOffice > Product Design > Item Categories. The Retail Item Categories page opens. This page can also be accessed via the Retail Item Card by drilling down to Item Category Code field. Then go to Navigate > Images.

**Product Groups:** Go to LS Retail > BackOffice > Product Design > Product Groups. The page Retail Product Groups opens. This page can also be accessed via the Retail Item Card by drilling down to the Product Group Code field. Then go to Navigate > Images.

In the Retail Image Link List you add an image link in the same way as the links to Images and Variants.

The page allows the user to select more than one image, but the one used in these cases needs to have Display Order = 0, which is the default value. If more than one image has Display Order = 0 the first in the list is used.
5.6 Jobs

There are three jobs for eCommerce and two of them are generated with WI Default Data page 10012868.

<table>
<thead>
<tr>
<th>Job Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-COMMERCE, ACTION</td>
<td>E-Commerce by Action replic.</td>
</tr>
<tr>
<td>E-COMMERCE, NORMAL</td>
<td>E-Commerce Normal replic.</td>
</tr>
<tr>
<td>E-COMMERCE, SCHED</td>
<td>E-Commerce Scheduler Job</td>
</tr>
</tbody>
</table>

**E-COMMERCE, NORMAL** is a normal job used in the setup process to replicate data to the web store and can be repeated as needed.

**E-COMMERCE, ACTION** replicates data if an action has been created on the "WI NC"-tables that are a set of buffer tables.

**E-COMMERCE, SCHED** is run by Scheduler. It runs the codeunit that updates the "WI NC"-tables with regard to the Web Store Last Action Entry No: on the LS Retail > Web Integration > WI Setup > Setup.

In the memo file eCommerce Data Mapping Details LS OMNI and LS NAV there are further information on the Jobs and Subjobs to use in eCommerce.

5.7 Sales Orders

After Order Confirmation and Payment confirmation in NOP, a standard Sales Order is created in LS Nav with the web service WI_NC_CREATE_ORDER. The Customer can see the Order Status in NOP using the web service WI_NC_ORDER_STATUS.

When a purchase takes place in the web store, a standard Nav Sales Order is created. This Sales Order is created with **Nav Document No.** which is a code-variable, but in the web store it has a NOP Order Number which is an integer. So to find the Orders in LS Nav, access it and see its status and shipping status on the NOP Order Number. We have the page Web Store Sales Documents under Web Integration at LS Retail > Web Integration > Lists > Web Store Sales Documents.
When a Sales Order is posted, a Retail Posted Sales Invoice is created on a new Document No. and the original Sales Order does not exist anymore. This information is updated on the page and the action “Sales Document” shows the relevant sales documents for the selected line.

**Order Status:**

- **Pending:** When the Sales Order is created it gets the Order Status *Pending* by default. A Sales Order that has been released can get the Order Status *Pending* if it is reopened.
- **Processing:** When the Sales Order is released, it gets the Order Status *Processing*.
- **Complete:** A Sales Order that is posted gets the Order Status *Complete*.
- **Canceled:** If a Sales Order is canceled it gets the status *Canceled*.

**Shipping Status:**

- **Not required:** If shipping is not required.
- **Not shipped:** None of the items in the order have been shipped.
- **Partially:** Some of the items in the Sales Order have been shipped and some not.
- **Shipped:** All items in the order have been shipped.
- **Delivered:** All the items in the order have been shipped and delivered.

Initially the order has the status Pending but when Sales Order has been created from the basket in LS Nav using web-services, it will enter the status *Not Shipped; Partially Shipped*, if a part of the order has been shipped; *Shipped* and *Delivered*, only if the order has reached the Order Status: *Complete*.

### 5.7.1 Shipping Requirements

Standard NAV functionality is used for shipping. Retail Sales Order is created in NAV.
5.7.2 Out of Stock Orders

It is possible to buy variant items in the web store even though it is not available in stock. It is assumed that it will be in stock when the order is ready for delivery. If that is not the case then it is the Back Office's role to contact the buyer with further information. It is also possible not to allow this and a setting TRUE/FALSE in the code decides which is used.

This means that Out of Stock Orders are allowed for products with variants in eCommerce. This is handled in the Codeunit WINCProductVariantAttrComb_p.AllowOutOfStockOrders:= TRUE; and can be changed there to FALSE if another approach is preferred.

5.7.3 Cancelling Orders

1. If the customer wants to cancel orders, it is not done in the web store directly. It has to be done in the back-end so the customer contacts the Head Office or Store by e-mail or phone and the staff on that end does the cancellation manually in the system.
2. What the customer can do at all times is to view the status of the order and stages. Therefore, it is possible to check the status at all times, for instance if the order has actually been cancelled.
3. When viewing an earlier order it is possible to reorder the same item(s).
Chapter 6 - User Buying Goods in the Web Store

6 User Buying Goods in the Web Store

The basic functionality from the user’s side can be described with this scenario:

1. The user logs on to the Web Store.

2. The Web Store's page appears offering several options to look up or buy goods that can be displayed, for instance by Product Group as defined in LS Nav. Pictures can be added to Product Groups, Item Categories and to Item Variants in LS Nav for this purpose.
3. The user makes a purchase.

The item is either available or Out of Stock.

If it is out of stock it is not added to the Shopping Cart.

4. If the item(s) the user wants to buy are available, they are added to the shopping cart and a notification tells you they are added. If you look at the status of your shopping cart, you can see how many items have been added and, on mouseover, see what is there.
5. When the shopping is concluded, the user goes to the shopping cart or directly to **Checkout**.

6. At the shopping cart the user can perform several actions:
   
a. Remove one or more items from the shopping cart, marking them and selecting: **Update shopping cart**.
b. Continue shopping by selecting the corresponding button.
c. Change the quantity of the items.
Note that it is possible that latest discounts are not checked and a warning stating this may appear.

6. In the checkout process, information about the user is entered or called, including personal information and payment information.

7. When the Shipping Address is entered, the options available are to change it from the address for the user or use the one you have entered for the user.
8. In next step the shipping method is selected.

![Shipping Method](image1)

9. The next step offers payment methods.

![Payment Method](image2)

The basket calculator is called. The corresponding activity in LS Nav is explained in an earlier chapter.

💡 Basket Calculations are triggered in LS Nav using WI_NC_CALCULATE_BASKET

- Verifies Stock Status before payment
- Calculates final prices
- Price calculations are the same as in Mobile POS
10. Necessary payment information is shown in next step. This is text that can be edited from the admin panel.
11. In the final stage you are asked to confirm the order and given details on it.

At all stages you can go back to correct some part of the order.

Once the Order has been confirmed, a message appears to verify your order and give access to the order's details.

At this stage a sales order has been created in LS Nav as described earlier in this document.
7 eCommerce on a Mobile

The functionality of eCommerce on a Mobile is the same as in a Web Store even though the customer interface is bound to be different as shown in the following screenshots.

7.1 Front Page and Login
### 7.2 Products for eCommerce on a Mobile

**Many Products**

![Soft Drinks](image)

- **Lemonade, Ola-s 0.5 L**
  - 1.42€

- **Soda**
  - 1.16€

- **Soda Guarana 33 CL**
  - 1.28€

- **Soda Lime 0.5 L**
  - Price not visible
One Product

<table>
<thead>
<tr>
<th>Register</th>
<th>Log in</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shopping cart (0)</td>
<td>Wishlist (0)</td>
</tr>
</tbody>
</table>

Skirt Linda Professional Wear

Skirt Linda Professional Wear
7.3 Shopping Cart

![Shopping cart interface]

**Skirt Linda Professional Wear (BLACK)**
SIZE: 34
Price: **101.48€**
Qty.: 2

Total: **202.96€**

Update qty
7.4 Purchasing Items and Viewing Orders on a Mobile POS

Orders

Order Number: 71
Order status: Pending
Order Date: 8/22/2013 11:57:14 AM
Order Total: 101.48€
Details

Order Number: 62
Order status: Complete
Order Date: 8/13/2013 4:13:00 PM
Order Total: 140.00€
Return Item(s) Details

Product(s)

Skirt Linda Professional Wear (BLACK)
SIZE: 34
Price: 101.48€
Quantity: 1
Total: 101.48€
Re-order

Sub-Total: 101.48€
Shipping: 0.00€
Tax: 20.30€
Order Total: 101.48€
8 Appendix A – Web Services – Requests and Responses

The following screenshots show the eCommerce specific web services’ requests and responses.

8.1 WI_NC_CALCULATE BASKET – Request
### 8.2 WI_NC_CALCULATE BASKET – Response

![Image of WS Request Setup window](image)

<table>
<thead>
<tr>
<th>Level</th>
<th>Node Type</th>
<th>Source Type</th>
<th>Source</th>
<th>Node Name</th>
<th>Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top</td>
<td>Text</td>
<td>Request_ID</td>
<td>Response</td>
<td>Request_ID</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Text</td>
<td>Response_Code</td>
<td>Response_Code</td>
<td>Response_Code</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Text</td>
<td>Response_Text</td>
<td>Response_Text</td>
<td>Response_Text</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Text</td>
<td>Response_Body</td>
<td>Response_Body</td>
<td>Response_Body</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Table</td>
<td>MobileTransaction</td>
<td>MobileTransaction</td>
<td>MobileTransaction</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>Id</td>
<td>Id</td>
<td>Id</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>StoreId</td>
<td>StoreId</td>
<td>StoreId</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>TerminalId</td>
<td>TerminalId</td>
<td>TerminalId</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>StaffId</td>
<td>StaffId</td>
<td>StaffId</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>TransactionType</td>
<td>TransactionType</td>
<td>TransactionType</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>EntryStatus</td>
<td>EntryStatus</td>
<td>EntryStatus</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>ReceiptNo</td>
<td>ReceiptNo</td>
<td>ReceiptNo</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>TransactionNo</td>
<td>TransactionNo</td>
<td>TransactionNo</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>TransDate</td>
<td>TransDate</td>
<td>TransDate</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>CurrencyCode</td>
<td>CurrencyCode</td>
<td>CurrencyCode</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>CurrencyFactor</td>
<td>CurrencyFactor</td>
<td>CurrencyFactor</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>BusinessTAXCode</td>
<td>BusinessTAXCode</td>
<td>BusinessTAXCode</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>CustomerId</td>
<td>CustomerId</td>
<td>CustomerId</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>CustDiscGroup</td>
<td>CustDiscGroup</td>
<td>CustDiscGroup</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>MemberCardNo</td>
<td>MemberCardNo</td>
<td>MemberCardNo</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>MemberPriceGroupCode</td>
<td>MemberPriceGroupCode</td>
<td>MemberPriceGroupCode</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>MemberContactNo</td>
<td>MemberContactNo</td>
<td>MemberContactNo</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>ManualTotalDiscPercent</td>
<td>ManualTotalDiscPercent</td>
<td>ManualTotalDiscPercent</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>ManualTotalDiscAmount</td>
<td>ManualTotalDiscAmount</td>
<td>ManualTotalDiscAmount</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Table</td>
<td>MobileTransactionsLine</td>
<td>MobileTransactionsLine</td>
<td>MobileTransactionsLine</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>Id</td>
<td>Id</td>
<td>Id</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>StoreId</td>
<td>StoreId</td>
<td>StoreId</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>TerminalId</td>
<td>TerminalId</td>
<td>TerminalId</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>LineNo</td>
<td>LineNo</td>
<td>LineNo</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>EntryStatus</td>
<td>EntryStatus</td>
<td>EntryStatus</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>TransactionNo</td>
<td>TransactionNo</td>
<td>TransactionNo</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>LineType</td>
<td>LineType</td>
<td>LineType</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>Number</td>
<td>Number</td>
<td>Number</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>Barcode</td>
<td>Barcode</td>
<td>Barcode</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>CurrencyCode</td>
<td>CurrencyCode</td>
<td>CurrencyCode</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>CurrencyFactor</td>
<td>CurrencyFactor</td>
<td>CurrencyFactor</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Field</td>
<td>VariantCode</td>
<td>VariantCode</td>
<td>VariantCode</td>
<td></td>
</tr>
<tr>
<td>Field Type</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Barcode</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CurrencyCode</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CurrencyFactor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VariantCode</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UomId</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NetPrice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Price</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quantity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DiscountAmount</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DiscountPercent</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NetAmount</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TaxAmount</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TaxProductCode</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TaxBusinessCode</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ManualPrice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CardOnCostNo</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ManualDiscountPercent</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ManualDiscountAmount</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DiscInfoLine</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TotalDiscInfoLine</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ItemDescription</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VariantDescription</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UomDescription</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TenderDescription</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TransDate</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ExternalId</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MobileTransDiscountLine</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Id</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LineNo</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DiscountType</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PeriodicDiscType</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PeriodicDiscGroup</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DiscountPercent</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DiscountAmount</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## 8.3 WI_NC_CREATE_ORDER – Request

![Image of WS Request Setup]

<table>
<thead>
<tr>
<th>Level</th>
<th>Node Type</th>
<th>Source Type</th>
<th>Source</th>
<th>Node Name</th>
<th>Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>ReqID</td>
<td>Text</td>
<td>Request_ID</td>
<td>Request_ID</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ReqBody</td>
<td>Text</td>
<td>Request_body</td>
<td>Request_body</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Basket_Holder_Buffer</td>
<td>Table</td>
<td>WI_Basket_Holder_Buffer</td>
<td>WI_Basket_Holder_Buffer</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Contact_No.</td>
<td>Field</td>
<td>Contact_No.</td>
<td>Contact_No.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Currency_Code</td>
<td>Field</td>
<td>Currency_Code</td>
<td>Currency_Code</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Web_Transaction_GUID</td>
<td>Field</td>
<td>Web_Transaction_GUID</td>
<td>Web_Transaction_GUID</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Basket_Price</td>
<td>Field</td>
<td>Basket_Price</td>
<td>Basket_Price</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Full_Name</td>
<td>Field</td>
<td>Full_Name</td>
<td>Full_Name</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Ship_to_Address</td>
<td>Field</td>
<td>Ship_to_Address</td>
<td>Ship_to_Address</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Ship_to_Address_2</td>
<td>Field</td>
<td>Ship_to_Address_2</td>
<td>Ship_to_Address_2</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Ship_to_City</td>
<td>Field</td>
<td>Ship_to_City</td>
<td>Ship_to_City</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Ship_to_Post_Code</td>
<td>Field</td>
<td>Ship_to_Post_Code</td>
<td>Ship_to_Post_Code</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Ship_to_County</td>
<td>Field</td>
<td>Ship_to_County</td>
<td>Ship_to_County</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Ship_to_Country/Region</td>
<td>Field</td>
<td>Ship_to_Country/Region</td>
<td>Ship_to_Country/Region</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Bill_to_Address</td>
<td>Field</td>
<td>Bill_to_Address</td>
<td>Bill_to_Address</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Bill_to_Address_2</td>
<td>Field</td>
<td>Bill_to_Address_2</td>
<td>Bill_to_Address_2</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Bill_to_City</td>
<td>Field</td>
<td>Bill_to_City</td>
<td>Bill_to_City</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Bill_to_Country</td>
<td>Field</td>
<td>Bill_to_Country</td>
<td>Bill_to_Country</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Bill_to_Post_Code</td>
<td>Field</td>
<td>Bill_to_Post_Code</td>
<td>Bill_to_Post_Code</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Bill_to_Country/Region</td>
<td>Field</td>
<td>Bill_to_Country/Region</td>
<td>Bill_to_Country/Region</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Phone_No.</td>
<td>Field</td>
<td>Phone_No.</td>
<td>Phone_No.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Mobile_Phone_No.</td>
<td>Field</td>
<td>Mobile_Phone_No.</td>
<td>Mobile_Phone_No.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>E-Mail</td>
<td>Field</td>
<td>E-Mail</td>
<td>E-Mail</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Shipping_Agent_Code</td>
<td>Field</td>
<td>Shipping_Agent_Code</td>
<td>Shipping_Agent_Code</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Price_Include_YAT</td>
<td>Field</td>
<td>Price_Include_YAT</td>
<td>Price_Include_YAT</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>WNCOrId</td>
<td>Field</td>
<td>WNCOrId</td>
<td>WNCOrId</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Basket_Line_Buffer</td>
<td>Table</td>
<td>WI_Basket_Line_Buffer</td>
<td>WI_Basket_Line_Buffer</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Id</td>
<td>Field</td>
<td>Id</td>
<td>Id</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Line_No.</td>
<td>Field</td>
<td>Line_No.</td>
<td>Line_No.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Item_No.</td>
<td>Field</td>
<td>Item_No.</td>
<td>Item_No.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Variant_Code</td>
<td>Field</td>
<td>Variant_Code</td>
<td>Variant_Code</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Quantity</td>
<td>Field</td>
<td>Quantity</td>
<td>Quantity</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Price</td>
<td>Field</td>
<td>Price</td>
<td>Price</td>
<td></td>
</tr>
</tbody>
</table>
8.4 WI_NC_CREATE_ORDER – Response

<table>
<thead>
<tr>
<th>Level</th>
<th>Node Type</th>
<th>Source Type</th>
<th>Source</th>
<th>Node Name</th>
<th>Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Top</td>
<td>Text</td>
<td>Response</td>
<td>Response</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ReqID</td>
<td>Text</td>
<td>Request_ID</td>
<td>Request_ID</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ResCode</td>
<td>Text</td>
<td>Response_Code</td>
<td>Response_Code</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ResText</td>
<td>Text</td>
<td>Response_Text</td>
<td>Response_Text</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Normal</td>
<td>Text</td>
<td>Response_Body</td>
<td>Response_Body</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Normal</td>
<td>Table</td>
<td>WI Basket Header Buffer</td>
<td>WI_Basket_Header_Buffer</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Store No.</td>
<td>Store_No.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Contact No.</td>
<td>Contact_No.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Basket Price</td>
<td>Basket_Price</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Web Transaction GUID</td>
<td>Web_Transaction_GUID</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Document Type</td>
<td>Document_Type</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Normal</td>
<td>Table</td>
<td>WI Basket Line Buffer</td>
<td>WI_Basket_Line_Buffer</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Line No.</td>
<td>Line_No.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Item No.</td>
<td>Item_No.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Variant Code</td>
<td>Variant_Code</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Product Variant ID</td>
<td>Product_Variant_ID</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Quantity</td>
<td>Quantity</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Price</td>
<td>Price</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>GUID</td>
<td>GUID</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Response Code</td>
<td>Response_Code</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Normal</td>
<td>Field</td>
<td>Response Text</td>
<td>Response_Text</td>
<td></td>
</tr>
</tbody>
</table>

![WS Request Setup](image-url)

Chapter 8 - Appendix A – Web Services – Requests and Responses 56
8.5 WI_NC_ORDER_STATUS – Request

![Image of WI_NC_ORDER_STATUS Request]

8.6 WI_NC_ORDER_STATUS - Respond

![Image of WI_NC_ORDER_STATUS Respond]