**TurnTo® LINK Integration**

Version 15.1.0



Table of Contents

1. Summary 1-3

2. Component Overview 2-4

2.1 Functional Overview 2-4

2.2 Use Cases 2-4

2.3 Limitations, Constraints 2-4

2.4 Compatibility 2-4

2.5 Privacy, Payment 2-4

3. Implementation Guide 3-6

3.1 Setup 3-6

3.2 Configuration 3-6

3.2.1 Setting up your site 3-6

3.2.2 Create TurnTo Custom Object 3-7

3.2.3 Set up your Catalog Feed Job 3-7

3.2.4 Set up your Historical Feed Job 3-8

3.2.5 Upload a Historical Feed 3-9

3.3 Custom Code 3-9

3.3.1 Adding the TurnTo Widgets to your pages 3-9

3.3.2 Item Teaser: 3-10

3.3.3 Post Purchase Widget: 3-10

3.4 External Interfaces 3-11

3.5 Testing 3-11

4. Operations, Maintenance 4-12

4.1 Data Storage 4-12

4.2 Availability 4-12

4.3 Support 4-12

5. User Guide 5-13

5.1 Roles, Responsibilities 5-13

5.2 Business Manager 5-13

5.3 Storefront Functionality 5-13

6. Known Issues 6-14

7. Release History 7-15

# Summary

The TurnTo® LINK integration makes the process of deploying TurnTo® on your site painless.

This LINK integration contains a cartridge with pipelines for exporting catalog and order information. Once the cartridge is deployed, these pipelines are used to setup background jobs that automate the synchronization of data with the TurnTo® system.

The cartridge also contains a pipeline that can be included on the on the product page which allows you to statically embed the TurnTo® Q&A and Reviews functionality.

Finally, the cartridge contains a pipeline that sets up TurnTo’s full-page Checkout Chatter™ feature. This feature allows site visitors to see what people are buying and why.

It is assumed that anyone installing this LINK integration is an existing TurnTo® customer.

# Component Overview

## Functional Overview

The TurnTo® LINK integration consists of five components. The historical order component exports historical customer orders to a file in a TurnTo® defined format. The catalog feed component periodically sends a feed to TurnTo® containing the entire product catalog. The Q&A component allows for including the TurnTo® Q&A module statically on the product page. The Reviews component allows for including the TurnTo® Reviews module statically on the product page. Finally, the last component configures a new page for viewing the full-page Checkout Chatter® feature.

The historical customer order export is a job that is setup to run only once. The exported file can then be manually uploaded to the TurnTo® system. This data allows TurnTo® to determine which customers have previously purchased products.

The catalog feed keeps TurnTo® up-to-date with any changes to your catalog. It is setup to run twice a day.

The Q&A component allows you to statically include TurnTo® questions and answers on your product page by simply including a template.

The Reviews component allows you to statically include TurnTo® reviews on your product page by simply including a template.

The Checkout Chatter® component allows you to easily setup a webpage to display to customer what other customers are buying and why. The layout of the page is pinboard-style.

## Use Cases

Site needs to export catalog and order information to be sent to TurnTo® in order to use the product. This cartridge supports these uses cases by allowing user’s to easily create background jobs to accomplish both of this tasks

User wants to ask a question about a product and get answers from previous customers or customer service. This cartridge supports this use case by easing the process of placing the TurnTo® Q&A module on the product page.

User wants to see reviews for a product or submit a review for a product. This cartridge supports this use case by easing the process of placing the TurnTo® Reviews module on the product page.

User wants to see what people are buying and why. This cartridge supports this use case by automatically setting up a page for users to view the full-page Checkout Chatter™.

## Limitations, Constraints

Use of the TurnTo® service requires a contract with TurnTo Networks. Please contact [sales@turnto.com](mailto:sales@turnto.com) for any questions.

## Compatibility

Available since Demandware 2.11.5

## Privacy, Payment

Customer purchases and email addresses will be shared with TurnTo®. This information is required for the TurnTo® application to function properly. TurnTo does not use any credit card data. TurnTo does not share the customer data with anyone and will remove the data in its entirety on request.

# Implementation Guide

## Setup

This TurnTo® LINK integration contains one cartridge named int\_turnto. The cartridge has two pipelines named TurnTo and TurnToExport. The TurnTo pipeline contains three entry points: GetProductQA, GetProductReviews and ShowChatter. The TurnToExport pipeline consists of two entry points: ExportHistoricalOrders and ExportCatalog. The entry points in TurnToExport should only be used to setup background jobs.

The ExportHistoricalOrders entry point exports all of the customer orders that have ever been placed. The export file, named “exportOrder.txt”, is written to a TurnTo folder within the Import/Export folder. This file should be used to manually upload the historical orders, as described in the TurnTo® documentation (<http://www.turnto.com/docs/demandware>). Historical orders only need to be uploaded once.

The ExportCatalog entry point exports all of the products from the catalog. The exported data is then automatically pushed to the TurnTo® system. A temporary file, named “exportCatalog.txt”, is created in a TurnTo folder within the Import/Export folder. The file is left on the files system after the job is finished. The file is overwritten each time the export job is run

## Configuration

This section describes the configuration of TurnTo® on Demandware®.

### Setting up your site

1. Create a TurnTo account on [www.turnto.com/register](http://www.turnto.com/register" \t "_new) if you don't already have one.   
  
2. Make sure your TurnTo contact has added your site to the turnto.com system.   
  
3. Make sure your TurnTo contact has made you a TurnTo account manager of your site. Once you are a manager of your site you will see a new navigation area on the left with your site name and links to your settings, reporting and moderation.   
  
4. Add int\_turnto to site cartridge path  
     - In Demandware Business Manager, Click Sites in the Administration section  
     - Click "Manage Sites" link  
     - Click the name of your site. For instance, SiteGenesis.  
     - Fill in the catridges field: int\_turnto:storefront.   
 **Note: replace storefront with your store cartridge**     - Click apply.  
  
5. Add int\_turnto to business manager cartridge path  
     - In Demandware Business Manager, Click Sites in the Administration section  
     - Click "Manage Sites" link  
     - Click the "Business Manager" link  
     - Fill in the catridges field: int\_turnto:storefront.   
 **Note: replace storefront with your store cartridge**  
     - Click Apply.

### Create TurnTo Custom Object

A number of the TurnTo® pipelines require the creation of a custom object named TurnTo. Follow the steps below to create and configure the object.

1. In Demandware Business Manager, Click Site Development in the Administration section
2. Click the Custom Object Definitions link
3. Click New
4. Fill in the following information:  
     
   ID: TurnTo  
   Key Attribute: TurnTo  
   Name: TurnTo  
   Data Replication: Not replicable  
   Storage Scope: Site
5. Click Apply
6. Click the Attribute Definitions tab
7. Create Attribute Definitions with the listed IDs by clicking the New button.

- authKey (Value Type String)  
- siteKey (Value Type String)  
- url (Value Type String)  
- staticURL (Value Type String)  
- jobTimeout (Value Type Integer)  
- wsTimeout (Value Type Integer)  
- version (Value Type String)

1. Click the Attribute Grouping tab
2. Type “config” for the ID and “Config Items” for the Name.
3. Click Add
4. Click the Edit link on the row with an ID of config.
5. Assign authKey, siteKey, url, staticURL, and jobTimeout to the Attribute Group   
   The Custom Object definition is now complete. Now you’ll need to create a new Custom Object using the Object Definition you just created.
6. Click Custom Objects in the Site side bar navigation menu
7. Click Custom Object Editor
8. Select “TurnTo” from the drop-down menu and click the New button
9. Fill in the following values:  
     
   authKey: <YOU AUTH KEY>  
   TurnTo: 1  
   siteKey: <YOUR SITE KEY>  
   url: http://www.turnto.com  
   staticURL: http://static.www.turnto.com  
   jobTimeout: 500000  
   wsTimeout: 7000  
   version: 4.3 **(OR desired TurnTo version)**  
    **NOTE: <YOUR AUTH KEY> and <YOUR SITE KEY> should be replaced with your actual siteKey and authKey. These values can be obtained from your TurnTo® representative.**

### Set up your Catalog Feed Job

**NOTE: The catalog feed includes links to product images. By default, the demandware image store is used. If you use a third party to store your images you’ll need to modify ExportCatalog.ds in the int\_turnto cartridge to point to your image store.**

1. Make sure you have installed the TurnTo Demandware Integration into your Demandware cloudbox.   
  
2. Login to your Demandware Business Manager   
  
3. Click the Operations link in the Administration section   
  
4. Click Job Schedules   
  
5. Click the New button to create a new Job   
  
6. Fill in the following information:   
    Name: TurnTo Catalog Export   
    Execution Scope: Sites   
    Pipeline: TurnToExport   
    Startnode: ExportCatalog   
    Recurring Interval: [checked]   
    Every: 12 Hours   
  
7. Click Apply   
  
8. Click Sites tab   
  
9. Check the box next to the site you want to export catalog data from.   
  
10. Click apply

### Set up your Historical Feed Job

**NOTE: The catalog feed includes links to product images. By default, the demandware image store is used. If you use a third party to store your images you’ll need to modify ExportHistoricalOrders.ds in the int\_turnto cartridge to point to your image store.**

1. Make sure you have installed the TurnTo Demandware Integration into your Demandware cloudbox.   
  
2. Login to your Demandware Business Manager   
  
3. Click the Operations link in the Administration section   
  
4. Click Job Schedules   
  
5. Click the New button to create a new Job   
  
6. Fill in the required information   
    Name: TurnTo Historical Order Export   
    Execution Scope: Sites   
    Pipeline: TurnToExport   
    Startnode: ExportHistoricalOrders   
  
7. Click Apply   
  
8. Click Sites tab   
  
9. Check the box next to the site you want to export catalog data from.   
  
10. Click apply   
  
11. Click General tab   
  
12. Click Run

### Upload a Historical Feed

The historical feed that was created in 3.2.3 must be manually uploaded to the TurnTo system.

1. Download the file generated in the previous section. It is located in the Import/Export section of Demandware in the TurnTo folder.   
  
2. Go to your TurnTo settings area on turnto.com and click on the "Upload Feed" navigation.   
  
3. Follow the instructions to upload the file that was generated from your system.

## Custom Code

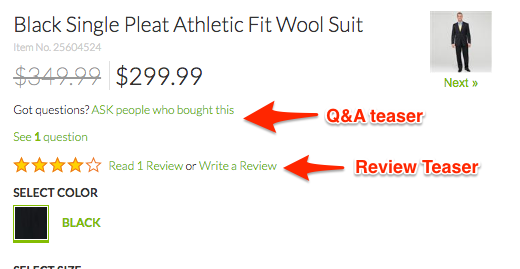
This section describes the custom code that will need to be added to your storefront. There are three different setup types that TurnTo® supports: overlay, dynamically embedded, and statically embedded. This document only describes how to implement statically embedded Q&A. For more information refer to http://www.turnto.com/docs or contact your TurnTo® representative.

### Adding the TurnTo Item Teaser and Statically Embedded Content

This section requires you have Demandware UX Studio setup and linked to your site. See <https://xchange.demandware.com/docs/DOC-1936> for details.  
  
*1. Log in to [www.turnto.com](http://www.turnto.com/" \t "_new) and click on****Settings****beneath your site name in the left navigation. Copy your "SiteKey" from the Manage Site area. This key will be used in turnToConfig javascript objects to identify you to our systems.   
  
2. In Demandware UX Studio, open the storefront template: storefront/catridge/templates/default/components/header/htmlhead.isml   
  
3. Add the following TurnTo code to the bottom of the template file you opened in step 2.*

|  |
| --- |
| *<iscomment>Include TurnTo</iscomment>*  *<isscript>*  *TTpid = null*  *if(pdict.Product != null)*  *{*  *TTpid = pdict.Product.isVariant() ? pdict.Product.masterProduct.ID : pdict.Product.ID;*  *}*  *</isscript>*  *<isif condition="${TTpid != null}">*  *<script type="text/javascript"><!--*  *var TurnToItemSku = "${TTpid}";*    *//--></script>*  *<script type="text/javascript"><!--*  *var TurnToItemSku = "${TTpid}";*    */\*\**  *\**  *\* This is a way to implement a teaser function that returns the html of the actual teaser.*  *\* This is called with the parameter TurnToItemData, which contains the item counts.*  *\* The clicks are calling clickQaTabFromTeaser() in this example.*  *\**  *\*/*  *function staticItemTeaserDisplay(data) {*  *if (typeof data == "undefined") {*  *// do nothing*  *return;*  *}*  *var iteaserhtml = '<div id="TT2ILTbox">Got questions? <span id="TT2ILTbutton-holder"><a class="TT2ILTbutton TurntoItemTeaserClick" href="javascript:clickQaTabFromTeaser()"><span>ASK people who bought this</span></a></span>';*  *iteaserhtml += '<div id="TT2ILTcount-line">';*  *if (data.counts.q > 0 || data.counts.a > 0) {*  *iteaserhtml += '<p><a class="TurntoItemTeaserClick TurnToIteaSee" href="javascript:clickQaTabFromTeaser()">See ';*  *if (data.counts.q > 0) {*  *iteaserhtml += '<strong>' + data.counts.q + '</strong>' + (data.counts.q > 1 ? ' questions' : ' question');*  *}*  *if (data.counts.a > 0) {*  *iteaserhtml += ' <strong>' + data.counts.a + '</strong>' + (data.counts.a > 1 ? ' answers' : ' answer');*  *}*  *iteaserhtml += '</a></p>';*  *}*  *iteaserhtml += '</div></div>';*  *return iteaserhtml;*  *}*    *function clickQaTabFromTeaser() {*  *var qaTab = jQuery('#tab-6');*  *qaTab.click();*  *window.scrollTo(0,qaTab.parent().offset().top);*  *}*    */\*\**  *\* Optional: Only needed if you intend on using reviews*  *\* This is a way to implement a teaser function that returns the html of the actual teaser.*  *\* This is called with the parameter TurnToItemData, which contains the item counts.*  *\* The clicks are calling clickReviewTabFromTeaser() in this example.*  *\*/*  *function staticReviewsTeaserDisplay(clazz, data) {*  *if (typeof data == "undefined") {*  *// do nothing*  *return;*  *}*  *var clazzNam = clazz || "TurnToReviewsTeaser";*    *var iteasers = jQuery("." + clazzNam);*    *// round the average rating to the nearest tenth*  *var rating = Math.round((data.counts.ar + 0.25) \* 100.0) / 100.0;*  *rating = rating.toString();*  *var decimal = parseInt(rating.substring(2, 3))*  *rating = rating.substring(0, 1) + "-" + (decimal >= 5 ? '5' : '0')*    *var html = '<div><div class="TT2left TTratingBox TTrating-'+rating+'"></div>'*  *+'<div class="TTratingLinks" style="padding: 2px 0 0 5px;">'*  *+' <a class="TTreadReviews" href="javascript:void(0)">Read '+data.counts.r+' Review'+(data.counts.r == 1 ? '' : 's')+'</a> or <a class="TTwriteReview" href="javascript:void(0)">Write a Review</a>'*  *+'</div><div class="TTclear"></div>'*  *+'</div>';*  *iteasers.html(html);*    *iteasers.find('.TTreadReviews').click(function(){clickReviewTabFromTeaser();TurnTo.itemReviewTeaserClick();});*  *iteasers.find('.TTwriteReview').click(function(){clickReviewTabFromTeaser();TurnTo.itemReviewTeaserClick({writeReview: true})});*  *}*    *function clickReviewTabFromTeaser() {*  *var reviewsTab = jQuery('#tab-5');*  *reviewsTab.click();*  *window.scrollTo(0,reviewsTab.parent().offset().top);*  *}*  *//--></script>*  *<script type="text/javascript"><!--*  *if(document.location.href.indexOf('orderconfirmation') < 0){*  *var turnToConfig = {*  *siteKey: "lJxfHSMDnb7W5fRsite",*  *localProxyUrl:"${URLUtils.absStatic('/tra/turntoproxy.html')}",*  *setupType: "staticEmbed",*  *reviewsSetupType: "staticEmbed",*  *iTeaserFunc: staticItemTeaserDisplay,*  *reviewsTeaserFunc: staticReviewsTeaserDisplay*  *};*  *(function() {*  *var tt = document.createElement('script'); tt.type = 'text/javascript'; tt.async = true;*  *tt.src = document.location.protocol + "//static.www.turnto.com/traServer4\_3/trajs/" + turnToConfig.siteKey + "/tra.js";*  *var s = document.getElementsByTagName('script')[0]; s.parentNode.insertBefore(tt, s);*  *})();*  *}*  *//--></script>*  *<script type="text/javascript" src="//static.www.turnto.com/sitedata/lJxfHSMDnb7W5fRsite/v4\_3/${TTpid}/d/itemjs"></script>*  *<link href="${URLUtils.staticURL('/css/turnto.css')}" type="text/css" rel="stylesheet" />*  *</isif>* |

The code snippet above will add the reviews and Q&A teasers to the page. It also loads TurnTo’s main javascript file. The screenshot below shows the teasers that are added

**

*4. Open template /default/product/producttopcontent.isml. Add the “Q and A” tab by pasting the following code in the place where you’d like the tab to appear (see screenshot in step 5):*

*<div class="tab">*

*<input type="radio" name="css-tabs" id="tab-6" class="tab-switch">*

*<label for="tab-6" class="tab-label">Q & A</label>*

*<div class="tab-content"><isinclude template="product/components/turntoqa"/></div>*

*</div>*

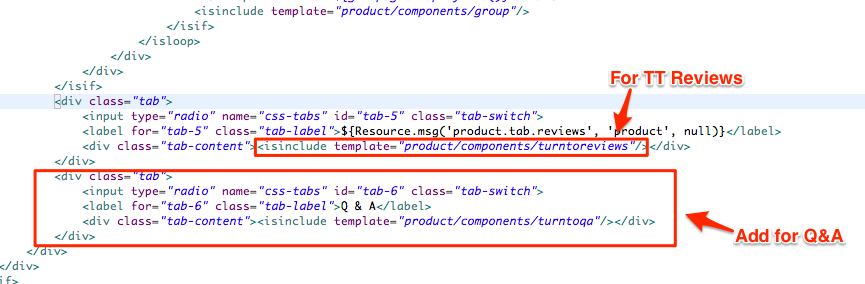
*5. Optional. Replace the built-in reviews with TurnTo reviews by replacing this line:*

*<isinclude template="product/components/reviews"/>*

*with this line:*

*<isinclude template="product/components/turntoreviews"/>*

*Example of code in IXML:*

**

*6. Add the teasers to the product page. Open templates/default/product/productcontent.isml and past the following code in the position where you'd like the teasers to appear.*

*<isif condition="${!isQuickView}">*

*<div class="TurnToItemTeaser"></div>*

*<isif condition="${pdict.CurrentHttpParameterMap.format.stringValue == 'ajax'}">*

*<script type="text/javascript">*

*jQuery('.TurnToItemTeaser').html(staticItemTeaserDisplay(TurnToItemData));*

*</script>*

*</isif>*

*<!-- Optional.  Only needed if you plan on using reviews from TurnTo -->*

*<div class="TurnToReviewsTeaser"></div>*

*<isif condition="${pdict.CurrentHttpParameterMap.format.stringValue == 'ajax'}">*

*<script type="text/javascript">*

*jQuery('.TurnToReviewsTeaser').html(staticReviewsTeaserDisplay(null, TurnToItemData));*

*</script>*

*</isif>*

*</isif>*

*7. The css for the TurnTo widget is in int\_turnto/cartridge/static/default/css/turnto.css. Feel free to modify it to your liking.   
  
8. Flush Business Manager cache  
     - In Demandware Business Manager, Click Sites in the Administration section  
     - Click "Manage Sites" link  
     - Click the "Business Manager" link  
     - Click the "Cache" tab  
     - Invalidate all caches by click the "Invalidate" buttons.  
  
9. Flush Site cache  
     - In Demandware Business Manager, Click Sites in the Administration section  
     - Click "Manage Sites" link  
     - Click the name of your site. For instance, SiteGenesis.  
     - Click the "Cache" tab  
     - Invalidate all caches by click the "Invalidate" buttons.*

### Post Purchase Widget:

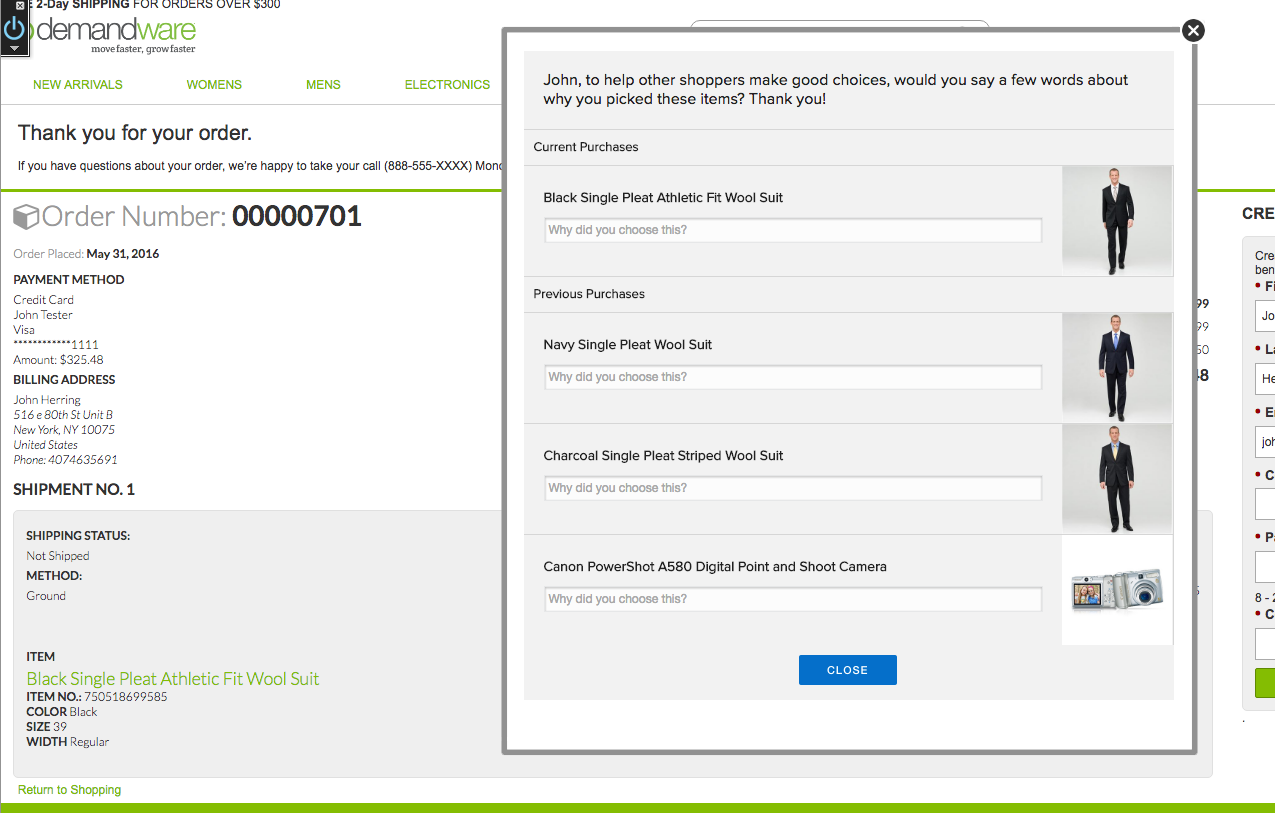
To add the TurnTo Post Purchase Widget to your order confirmation page you need to add the following code to the file: **storefront/templates/default/checkout/confirmation/confirmation.isml**

**NOTE: The post purchase feed includes links to product images. By default, the demandware image store is used. If you use a third party to store your images you’ll need to modify the following code snippet to point to your image store.**

Add the following code snippet after this line:   
**</isdecorate>**

|  |
| --- |
| <!-- start turnto post purchase widget -->  <script type=*"text/javascript"*>  var turnToConfig = {  siteKey: "lJxfHSMDnb7W5fRsite",  localProxyUrl:"${URLUtils.absStatic('/tra/turntoproxy.html')}",  floatingTeaserStyle:2,  orderConfFlowPauseSeconds:3,  postPurchaseFlow:true,  setupType: "overlay"  };  document.write(unescape("%3Cscript src='" + document.location.protocol + "//static.www.turnto.com/tra4\_3/turntoFeed.js' type='text/javascript'%3E%3C/script%3E"));  </script>  <script type=*"text/javascript"*>  document.write(unescape("%3Cscript src='" + document.location.protocol + "//static.www.turnto.com/traServer4\_3/trajs/" + turnToConfig.siteKey + "/tra.js' type='text/javascript'%3E%3C/script%3E"));  </script>  <script>  TurnToFeed.addFeedPurchaseOrder({orderId:'${pdict.Order.getOrderNo()}',email:'${pdict.Order.getCustomerEmail()}',postalCode: '${pdict.Order.getBillingAddress().getPostalCode()}',firstName: '${pdict.Order.getBillingAddress().getFirstName()}', lastName: '${pdict.Order.getBillingAddress().getLastName()}', deliveryDate: '${dw.util.StringUtils.formatCalendar(new dw.util.Calendar(pdict.Order.getCreationDate()), "yyyy-MM-dd hh:mm:ss")}' });  <isloop items="${pdict.Order.getProductLineItems()}" var="item">  TurnToFeed.addFeedLineItem({title: '${item.getProduct().getName().replace(/'/g, "\\'")}', url: '${dw.web.URLUtils.http('Product-Show', 'pid', item.getProduct().getID()).toString()}',sku:'${item.getProduct().isVariant() ? item.getProduct().getVariationModel().getMaster().getID() : item.getProduct().getID()}', price:'${item.getAdjustedNetPrice().getValue().toString()}',itemImageUrl:'${item.getProduct().getImage("small") != null ? item.getProduct().getImage("small").getAbsURL().toString() : ""}' });  </isloop>  TurnToFeed.sendFeed();  </script>  <!-- end turnto post purchase widget --> |

*After the Post Purchase widget is installed you should see an overlay on the checkout confirmation screen. The screenshot below is an example of what the widget looks like on the SiteGenesis store:*

**

*Flush Site cache  
     - In Demandware Business Manager, Click Sites in the Administration section  
     - Click "Manage Sites" link  
     - Click the name of your site. For instance, SiteGenesis.  
     - Click the "Cache" tab  
     - Invalidate all caches by click the "Invalidate" buttons.*

## External Interfaces

Behind the scenes, the TurnTo® LINK integration uses a web interface to push the catalog feed to the TurnTo® system using an HTTP POST. The details of this interface are described in this document: <http://www.turnto.com/docs/automate_feed>. However, since the integration with this interface is already done in the cartridge, you don’t need to be concerned with the details.

## Testing

*Your existing orders and products can be used to test the export jobs. The jobs only export data; they DO NOT alter any data in your store.  
  
If the steps in section 3.3.1 are followed, the product page will be altered and should be regression tested.  
  
The steps in section 3.3.2 alter the checkout confirmation page, therefore, it should be tested as well.*

# Operations, Maintenance

## Data Storage

*After running the background jobs a folder will be created in the Demandware Import/Export folder named TurnTo. Inside of the TurnTo folder there will be two files: exportOrders.txt and exportCatalog.txt. If desired, these files can be deleted at anytime.*

## Availability

TurnTo® is available 24 hours per day to receive feeds.

## Support

Please report bugs and feature requests to [support@turnto.com](mailto:support@turnto.com).

# User Guide

## Roles, Responsibilities

*After the instructions in section 3 have been followed, the system will maintain itself. However, it is recommended that you periodically ensure that the catalog export job is running without errors.*

## Business Manager

*This LINK integration does not contain any changes to the Demandware® Business Manager.*

## Storefront Functionality

*This LINK integration does not contain any changes to the Demandware® Storefront.*

# Known Issues

*There are no known issues.*

# Release History

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Changes** |
| 1.0.0 | 1/17/2012 | Initial release |
| 15.1.0 | 5/6/2015 | Upgraded to TurnTo version 4.2. Made changes for compatibility with Demandware 15.1.0. |