

# XML Tool

---

This tool is designed to create, email, and print special job-documents that you can customize. It works by creating **XML** files, which may use **XSL** templates to format documents.

## About XML

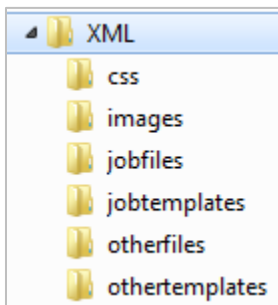
- **XML** stands for e**X**tensible **M**arkup **L**anguage.
- **XML** is designed to transport and store data.
- **XML** data is stored in plain text format that can be shared by different applications.

## About XSL Transformations (XSLT)

- **XSL** stands for **E**xtensible **S**tylesheet **L**anguage.
- **XSLT** stands for XSL **T**ransformations. Most modern web browsers support transformation of XML to HTML (or other languages) through XSLT.

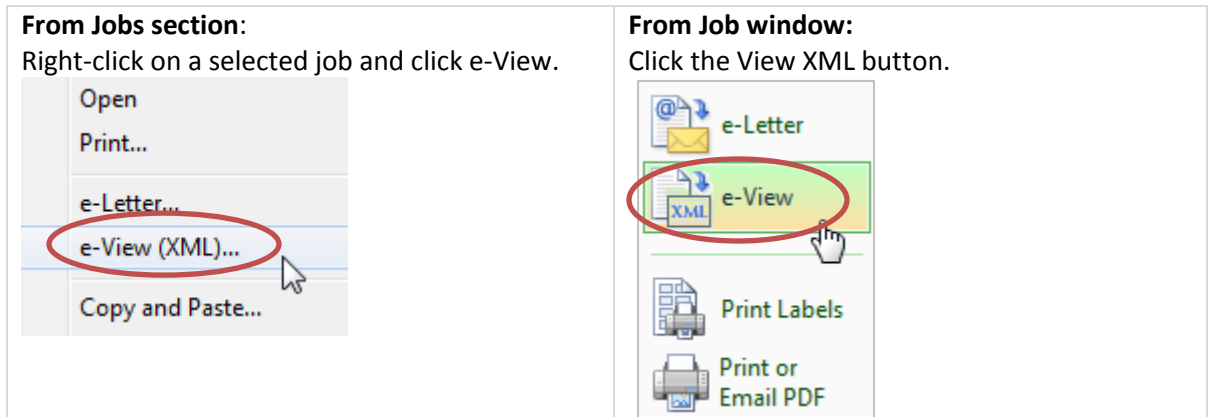
## Get Ready - Verify XML Folder

- After installing this tool, be sure the XML folder and its sub-folders are added to your Server/Printer's Plan folder.

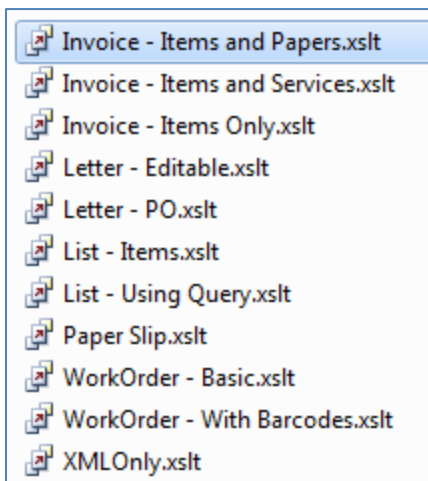


## Test - Create XML Files in Printer's Plan

You can create XML files from two locations:



Printer's Plan will ask you to select a template (XSLT file).



After you select a template, Printer's Plan will:

- Create an XML file in the "jobfiles" folder.
- Name the file as: Job Type + Job Number + Template Name + ".xml".  
Example: Order123 – Invoice – Items and Papers.xml.
- Include a reference to the template name in the xml file.
- Open the document (formatted XML file) in your web browser where you can email and print.

**Exception:** If you select the "XMLOnly" template or type a file name that is not listed, Printer's Plan will not add a template name to the xml file. When you open the file, you will see unformatted xml data.

**Suggestion:** Try each template and print each document for **one job**.

## Customize

After trying all templates, modify the following files for your company's needs. This document does not explain how to modify them.

1. **Image** file: XML\images\logo.gif.  
Create your company's image file and replace logo.gif.
2. **Stylesheet** file: XML\css\plan.css.  
This file defines how to display HTML elements. You may use it as is or change font name, size, color, etc.
3. **Template** files: XML\jobtemplates\\*.xslt.

## Sample XML File

```

<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="..\jobtemplates\Invoice - Items and Papers.xslt"?>
<Jobs>
<Header>
  <DocName>Order2427.xml</DocName>
  <DocHead>Order2427</DocHead>
</Header>
<Job>
  <JobName>Order</JobName>
  <JobNo>2427</JobNo>
  .....
  <Customer>
    <CusNo>32</CusNo>
    <Name>Abel Laboratories</Name>
    <Address>3370 Peartree Street Suite 300</Address>
    .....
  </Customer>
  <Buyer>
    <ContactNo>3</ContactNo>
    <First>Mark</First>
    <Last>Sturm</Last>
    .....
  </Buyer>
  <Shipto>
    <ShiptoNo>10</ShiptoNo>
    <Company>Abel Laboratories</Company>
    <Address>1825 Charles Street Suite 400</Address>
    .....
  </Shipto>
  <CSR>
    <EmpNo>102</EmpNo>
    <Name>Paul</Name>
    .....
  </CSR>
  <SalesRep>
    <EmpNo>202</EmpNo>
    <Name>Paul</Name>
    .....
  </SalesRep>
  <Item>
    <ItemID>4279</ItemID>
    <Description>Letterhead</Description>
    <Service id="1" name="Prep">
      <SerName>CTP Plate</SerName>
      <ActualTime>0</ActualTime>
      .....
    </Service>
    <Service id="2" name="PAPER">
      <CatName>Writing</CatName>
      <SerName>24# Classic Laid</SerName>
      <RunSize>8.5 x 11</RunSize>
      <FinSize>8.5 x 11</FinSize>
      <Color>White-Warm</Color>
      <Side1>Black + 1 Color</Side1>
      .....
    </Service>
  </Item>
  .....
</Job>
</Jobs>

```

## Sample XSL File

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <html>
      <head>
        <title><xsl:value-of select="Jobs/Document"/> </title>
        <meta http-equiv="content-type" content="text/html; charset=UTF-8"/>
        <link rel="stylesheet" type="text/css" href="../css/plan.css"></link>
      </head>

      <body>
        <xsl:for-each select="Jobs/Job">
          <!-- logo and invoice info -->
          .....
          <!-- bill to and ship to -->
          .....
          <!-- more job info -->
          .....
          <!-- Items and Services -->
          <xsl:for-each select="Item">
            <xsl:for-each select="Service">
              .....
            </xsl:for-each>
            .....
          </xsl:for-each>
          .....
          <!-- job totals -->
          .....
          <!-- footer -->
          .....
        </xsl:for-each>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```