# Crystal Reports

## Using ODBC XML Data with Crystal Reports

## Overview

This document discusses Extensible Markup Language (XML) concepts, provides sample XML scripting, and illustrates the best methods to report off XML in Crystal Reports (CR) using ODBC. The focus of this document is working with ODBC connections to XML data but Crystal Reports 10 and later can connect to XML using the Crystal Reports Native XML driver. A link for downloading the Native XML driver is provided. This document applies to Crystal Reports 8.5 and later.

## Contents

# Introduction

Crystal Reports 8.5 and later allows you to report off XML data. XML is a markup language much like HTML (Hyper Text Markup Language). However, XML and HTML were designed with different intentions.

HTML was designed to display data and concentrate on how data appears. However, XML was designed to describe data and concentrate on what the data is. While HTML contains predefined tags, XML allows you to create customized tags for the structure of the XML document. Each element is contained between opening and closing tags.

# Before You Begin

## Validate the XML

Before you begin, ensure that the XML is valid and without errors. Opening the XML as a local XML file through an Internet Explorer browser 5.5 and higher does not guarantee that the XML has a valid schema or structure. For this reason, it is strongly recommended that you test the XML.

Microsoft offers a utility on their support site called the XML Validator.

Download the XML Validator from the following link:

http://msdn.microsoft.com/archive/default.asp?url=/archive/en-us/samples/internet/xml/xml_validator/default.asp

The XML Validator will verify that the XML is well formed, valid and error free. Also, this utility indicates elements such as parent/child tags and any attributes associated to elements within the XML and the XML hierarchy structure. You will be able to correctly identify any Table or Row hints for the CR XML driver to use. Table and Row hints are further discussed later in this document.

The XML Validator can be used with both XML schemas and Data Type Definitions (DTDs) and allows you to verify unsecured URL XML, and to paste any particular XML to test.

### Steps to Test the XML

1. Click the link Microsoft's site to the XML Validator provided above.
2. Click the Demo link. The XML Validator appears. (If available, you are able to download the utility as well.)
3. Enter the XML path or URL path. Or, click the **Paste** button to enter the XML.
4. Ensure that the **Validation** check box is selected.
5. Click the **Validate** button.

6. Correct the XML if an error message appears.

7. Repeat the steps if necessary.

Upon completing these steps, you will receive a message stating, "Your XML is well formed and is validated" or an error message related to the XML.

Below the message "Your XML is well formed and is validated", there is parser information for the XML and any parent/child tags or elements sorted by hierarchy. If any attributes are found, this will also be listed according to hierarchy level. This information indicates the XML version and if the XML is Unicode compliant. Based on this information, you can identify any Table or Row hints.

# Downloading the CR XML Native Driver

New to Crystal Reports 10 is the Native XML driver. With Crystal Reports XI, the Native XML driver automatically installs with a complete product installation. With Crystal Reports 10, you can download the Native XML driver by clicking "Download Windows JDBC, XML and DB2 Unicode drivers" at

http://www.businessobjects.com/products/downloadcenter/crystalreports.asp

After downloading, consult the included documentation (*install.pdf* and *new_drivers.pdf*) for installation assistance.

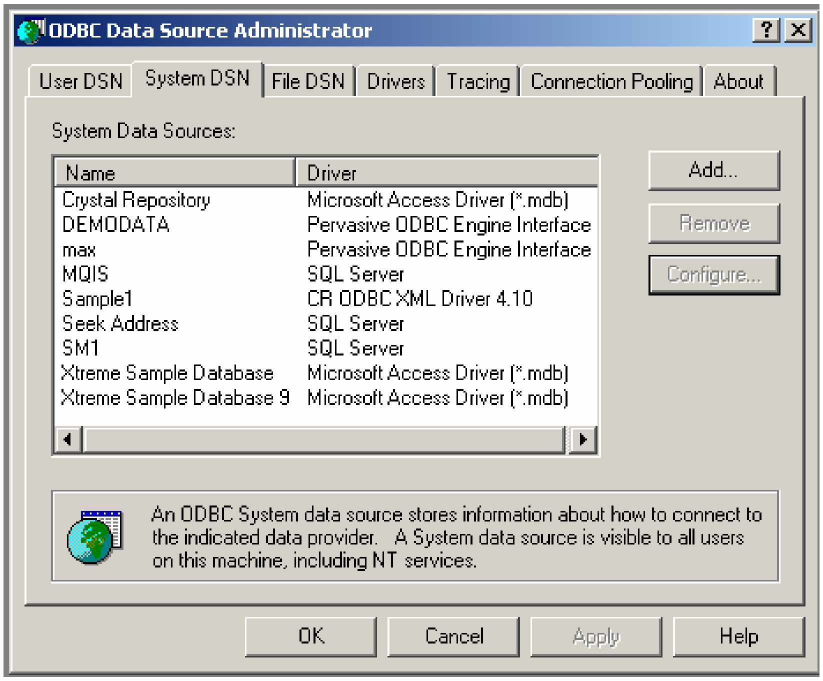
# Configuring the CR XML ODBC Driver

Crystal Reports retrieves data from XML documents using the CR XML ODBC driver, included with Crystal Reports 8.5 and later. Before XML data can be retrieved, an ODBC data source must be configured using this ODBC driver.

| | |
|---|---|
| CAUTION | Do not name the XML file "Test.xml" or with the suffix "...test.xml". An ODBC error will occur. For more information regarding this issue, refer to the knowledge base article,c2011224, on our support site at:<br>http://support.businessobjects.com/search |

## Steps to configure the CR XML ODBC driver in CR 9

1. Open the **ODBC Data Source Administrator**, usually found in the **Windows Control Panel**.

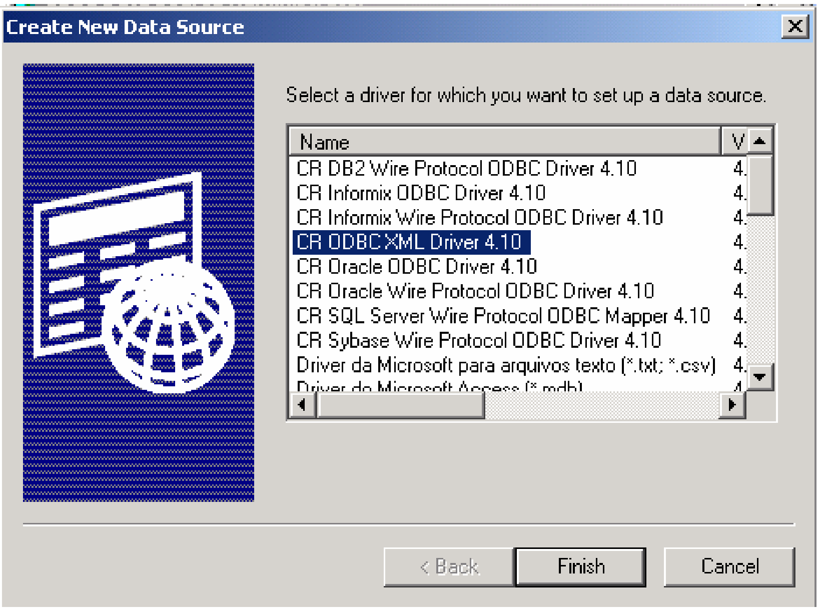


System Data Sources:

| Name | Driver |
|---|---|
| Crystal Repository | Microsoft Access Driver (*.mdb) |
| DEMODATA | Pervasive ODBC Engine Interface |
| max | Pervasive ODBC Engine Interface |
| MQIS | SQL Server |
| Sample1 | CR ODBC XML Driver 4.10 |
| Seek Address | SQL Server |
| SM1 | SQL Server |
| Xtreme Sample Database | Microsoft Access Driver (*.mdb) |
| Xtreme Sample Database 9 | Microsoft Access Driver (*.mdb) |

An ODBC System data source stores information about how to connect to the indicated data provider. A System data source is visible to all users on this machine, including NT services.

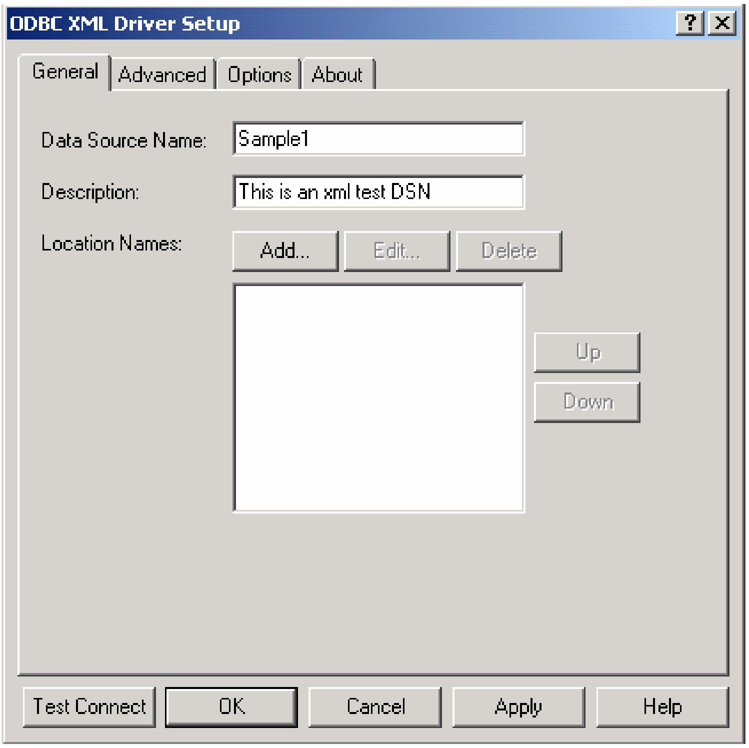
2. Click **Add** to create a new data source. From the list of available ODBC drivers, select **CR ODBC XML Driver 4.10**. Click **Finish**.



Select a driver for which you want to set up a data source.

| Name | V |
|---|---|
| CR DB2 Wire Protocol ODBC Driver 4.10 | 4. |
| CR Informix ODBC Driver 4.10 | 4. |
| CR Informix Wire Protocol ODBC Driver 4.10 | 4. |
| CR ODBC XML Driver 4.10 | 4. |
| CR Oracle ODBC Driver 4.10 | 4. |
| CR Oracle Wire Protocol ODBC Driver 4.10 | 4. |
| CR SQL Server Wire Protocol ODBC Mapper 4.10 | 4. |
| CR Sybase Wire Protocol ODBC Driver 4.10 | 4. |
| Driver da Microsoft para arquivos texto (*.txt; *.csv) | 4. |
| Driver do Microsoft Access (*.mdb) | 4 |

3. The **ODBC XML Driver Setup** dialog box appears.



Data Source Name: Sample1

Description: This is an xml test DSN

In the **ODBC XML Driver Setup** dialog box, left-click the **General** tab and enter the following:

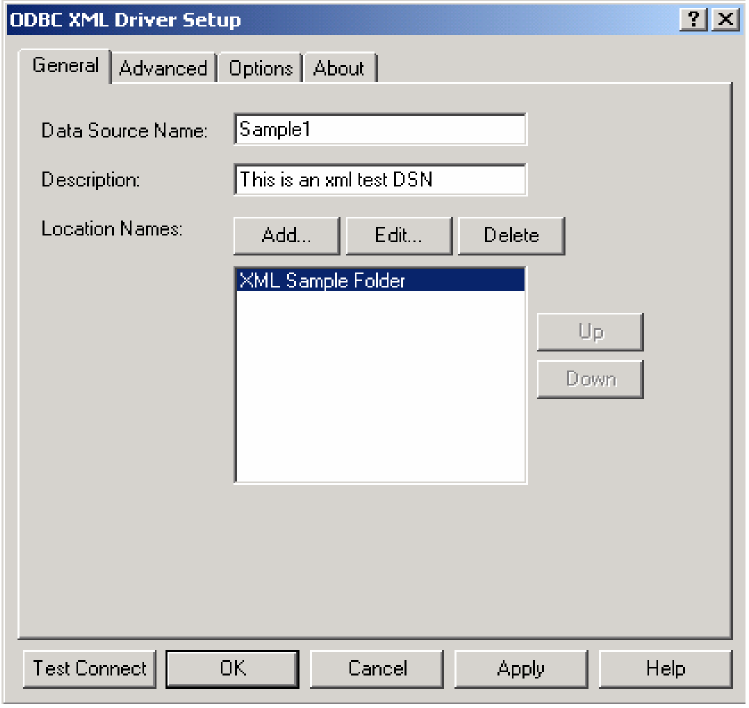
**Data Source Name**: Enter the name you would like to assign to this data source.

**Description**: May be left blank or a simple description can be added for this data source.

**Location**: Contains the path or URL to the folder containing your XML document(s).

| | |
|---|---|
| **NOTE** | For the **Location** field, ensure that the folder containing the XML files is specified, and not the XML files. |

## General Tab

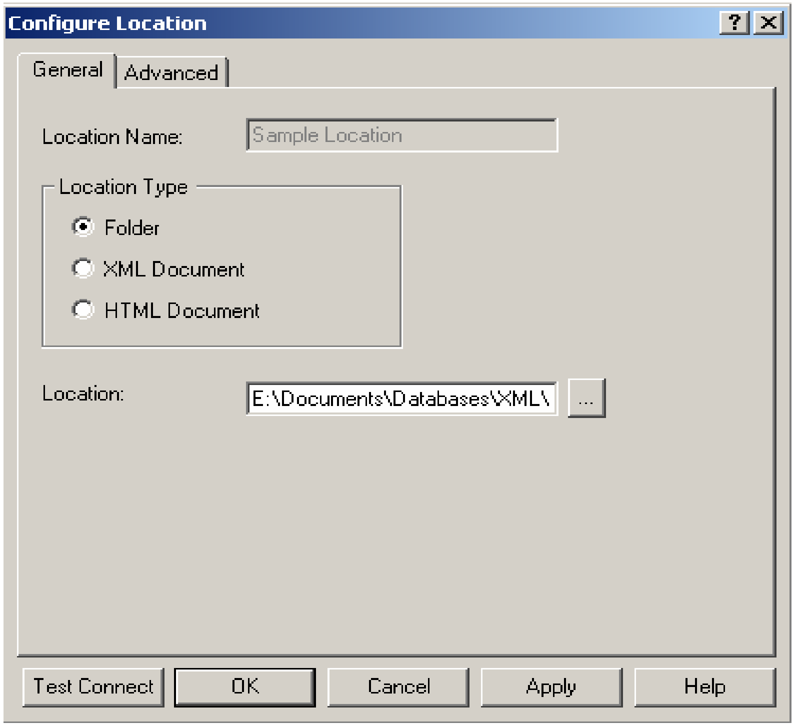


Data Source Name: Sample1

Description: This is an xml test DSN

Location Names: XML Sample Folder

Once you click **Add** or **Edit** you will be prompted with the **Configure Location** window.



Location Name: Sample Location

Location Type

● Folder

○ XML Document

○ HTML Document

Location: E:\Documents\Databases\XML\

In the **Configure Location** dialog box under the **General** tab, enter the following information:

**Location Name:** Provide a name for the location of the XML file.

**Location Type:** This is either the Folder, File or HTML file in which the XML is located.

**Location:** Browse to the location of the XML source by clicking the **...** button or enter the URL.

| | |
|---|---|
| **NOTE** | If a file name is not specified in the URL (for example, http://www..../xml_samples), Crystal Reports prompts you to specify the table name. That is, the file name without the ".xml" extension. |

| | |
|---|---|
| **NOTE** | If a file name is not specified in the path (for example, c:\yourxmlfiles), the Database Expert displays each XML file from the directory c:\yourxmlfiles as a table. |

## Advanced Tab

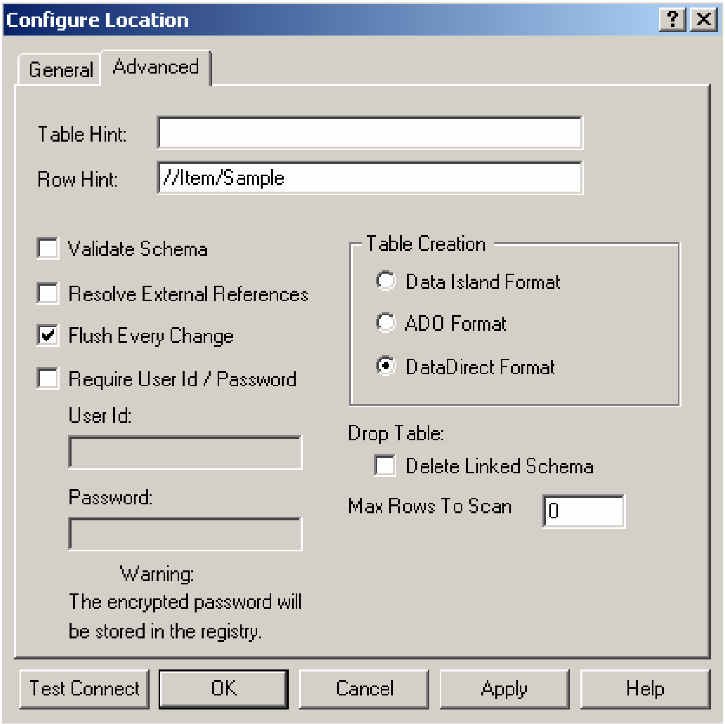


Table Hint:

Row Hint: //Item/Sample

☐ Validate Schema

☐ Resolve External References

☑ Flush Every Change

☐ Require User Id / Password

User Id:

Password:

Warning:
The encrypted password will be stored in the registry.

Table Creation

○ Data Island Format

○ ADO Format

● DataDirect Format

Drop Table:

☐ Delete Linked Schema

Max Rows To Scan 0

In the **ODBC XML Driver Setup** dialog box, left-click the **Advanced** tab and enter the following:

**Table Hint**: Enter the table hint needed to interpret the XML document, if applicable.

**Row Hint**: Enter the row hint needed to interpret the XML document, if applicable.

| NOTE | Table and Row Hints are discussed, at length, later in this document. |
|---|---|

**Validate Schema**: This option is unselected by default. This option enables the CR XML ODBC driver to check the XML document to ensure that the data it contains matches its schema (data structure). If unselected, the XML document will be processed regardless. It is recommended that this option be checked.

**Resolve External References**:

This option is unselected by default, this option will enable the CR XML ODBC driver to use files used by the XML document, such as DTDs, Schemas, Entities, and Notations. It is recommended that this option be checked.

**Flush Every Change:** This allows for the XML file to have flushing of the twig (an extension of the TREE\Branch) up to and including the current element, enabled. Flushing is done in the XML code.

**Require User Id / Password**:

This option is checked by default. This option prompts for a user ID and password when connecting to this data source. Unless there is security placed on the XML document(s), this item should be unselected.

**Table Creation:** This option determines the style of XML file that is produced when this data source is used as a means to create XML documents. This option has no effect when retrieving data and reporting off XML documents.

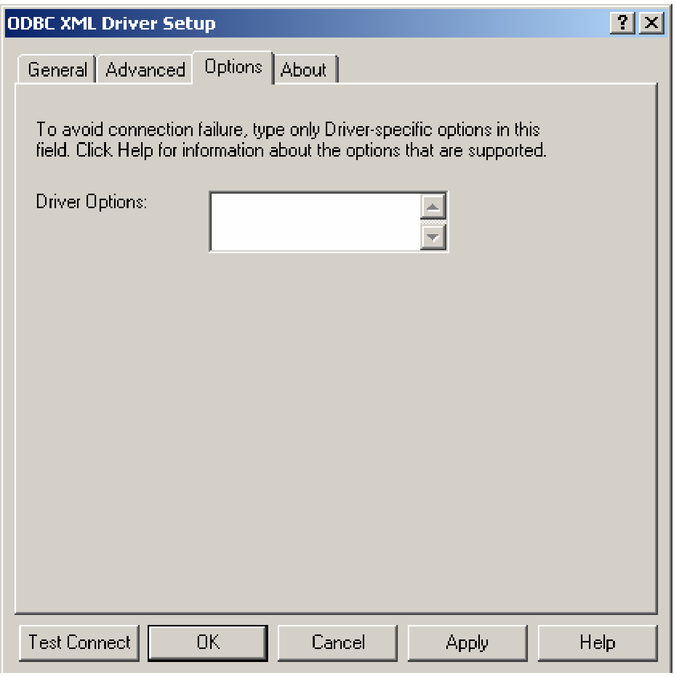
**International Sort Order:**

This option affects the order that records are retrieved when issuing an ORDER BY clause in the SELECT statement passed to this data source. For more information on this option, please see the CR XML ODBC Online help, via the Help button.

**Delete Linked Schema**: This option does not directly affect reporting. For more information on this option, please see the CR XML ODBC Online help, via the Help button.

**Logging:** This option does not directly affect reporting. For more information on this option, please see the CR XML ODBC Online help by pressing the Help (F1) button.

### Options Tab



To avoid connection failure, type only Driver-specific options in this field. Click Help for information about the options that are supported.

Driver Options:

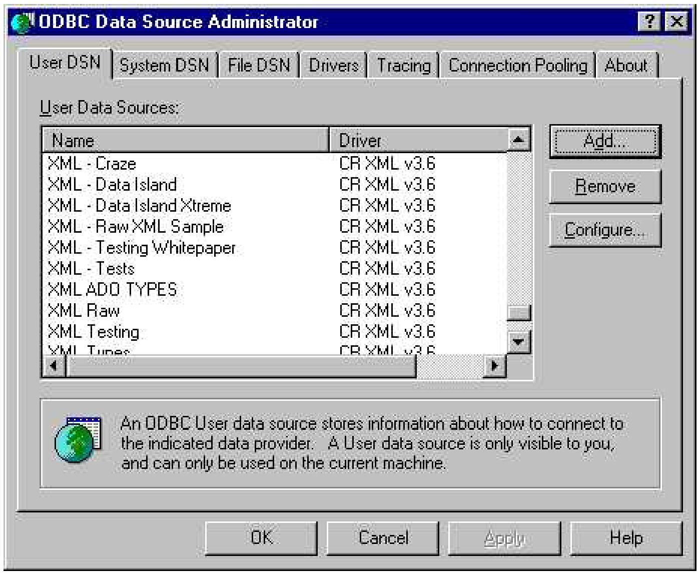
In the **ODBC XML Driver Setup** dialog box, under the **Options** tab is the **Driver Options**.

The **Driver Options** does not directly affect reporting. For more information on this option, please refer to the CR XML ODBC Online help by pressing the **Help** (F1) button.

## Steps to configure the CR XML ODBC driver in CR 8.5

To configure the CR XML ODBC driver:

**8.** Open the **ODBC Data Source Administrator**, usually found in the Windows Control Panel.

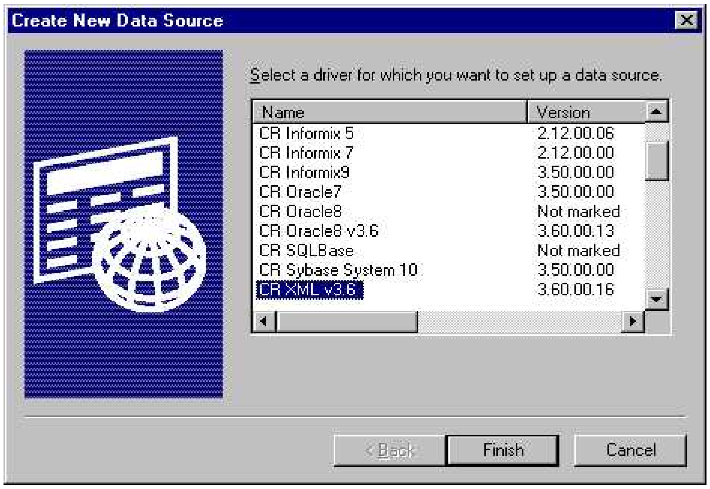


User Data Sources:

| Name | Driver |
|---|---|
| XML - Craze | CR XML v3.6 |
| XML - Data Island | CR XML v3.6 |
| XML - Data Island Xtreme | CR XML v3.6 |
| XML - Raw XML Sample | CR XML v3.6 |
| XML - Testing Whitepaper | CR XML v3.6 |
| XML - Tests | CR XML v3.6 |
| XML ADO TYPES | CR XML v3.6 |
| XML Raw | CR XML v3.6 |
| XML Testing | CR XML v3.6 |
| XML Types | CR XML v3.6 |

An ODBC User data source stores information about how to connect to the indicated data provider. A User data source is only visible to you, and can only be used on the current machine.

**9.** Click **Add** to create a new data source. From the list of available ODBC drivers that appears, select **CR XML v3.6**. Click **Finish**.

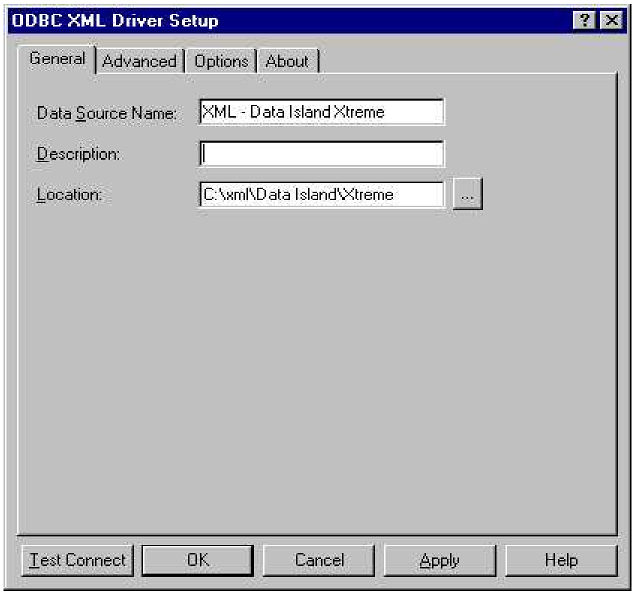


Select a driver for which you want to set up a data source.

| Name | Version |
|---|---|
| CR Informix 5 | 2.12.00.06 |
| CR Informix 7 | 2.12.00.00 |
| CR Informix9 | 3.50.00.00 |
| CR Oracle7 | 3.50.00.00 |
| CR Oracle8 | Not marked |
| CR Oracle8 v3.6 | 3.60.00.13 |
| CR SQLBase | Not marked |
| CR Sybase System 10 | 3.50.00.00 |
| CR XML v3.6 | 3.60.00.16 |

**10.** The **ODBC XML Driver Setup** dialog box appears.

## General Tab



Data Source Name: XML - Data Island Xtreme

Description:

Location: C:\xml\Data Island\Xtreme

In the **ODBC XML Driver Setup** dialog box, left-click the **General** tab and enter the following:

**Data Source Name**: Enter the name you would like to assign to this data source.

**Description**: May be left blank or a simple description can be added for this data source.

**Location**: Contains the path or URL to the folder containing your XML document(s).

| | |
|---|---|
| CAUTION | For the **Location** field, ensure that the folder containing the XML files is specified, and not the XML files. |

## Advanced Tab

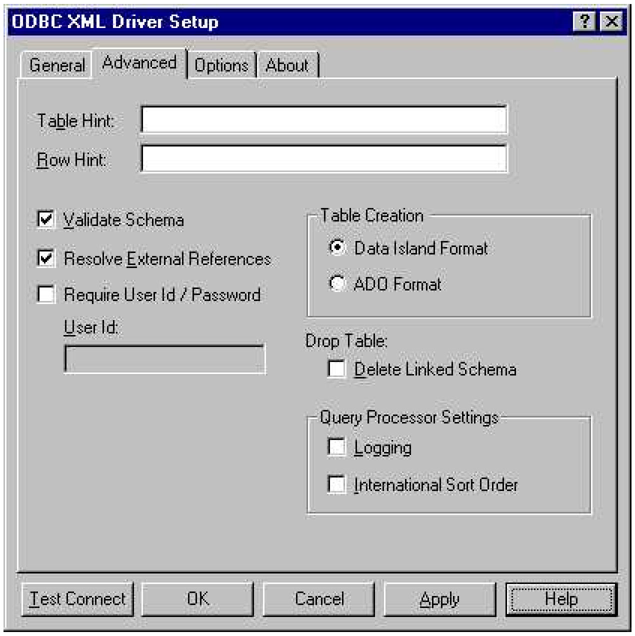


Table Hint:

Row Hint:

☑ Validate Schema

☑ Resolve External References

☐ Require User Id / Password

User Id:

Table Creation

● Data Island Format

○ ADO Format

Drop Table:

☐ Delete Linked Schema

Query Processor Settings

☐ Logging

☐ International Sort Order

In the **ODBC XML Driver Setup** dialog box, left-click the **Advanced** tab and enter the following:

**Table Hint**: Enter the table hint needed to interpret the XML document, if applicable.

**Row Hint**: Enter the row hint needed to interpret the XML document, if applicable.

| | |
|---|---|
| **NOTE** | Table and Row Hints are discussed, at length, later in this document. |

**Validate Schema**: This option is unselected by default. This option enables the CR XML ODBC driver to check the XML document to ensure that the data it contains matches its schema (data structure). If unselected, the XML document will be processed regardless. It is recommended that this option be checked.

**Resolve External References**:

This option is unselected by default, this option will enable the CR XML ODBC driver to use files used by the XML document, such as DTDs, Schemas, Entities, and Notations. It is recommended that this option be checked.

**Require User Id / Password**:

This option is checked by default. This option prompts for a user ID and password when connecting to this data source. Unless there is security placed on the XML document(s), this item should be unselected.

**Table Creation:** This option determines the style of XML file that is produced when this data source is used as a means to create XML documents. This option has no effect when retrieving data and reporting off XML documents.

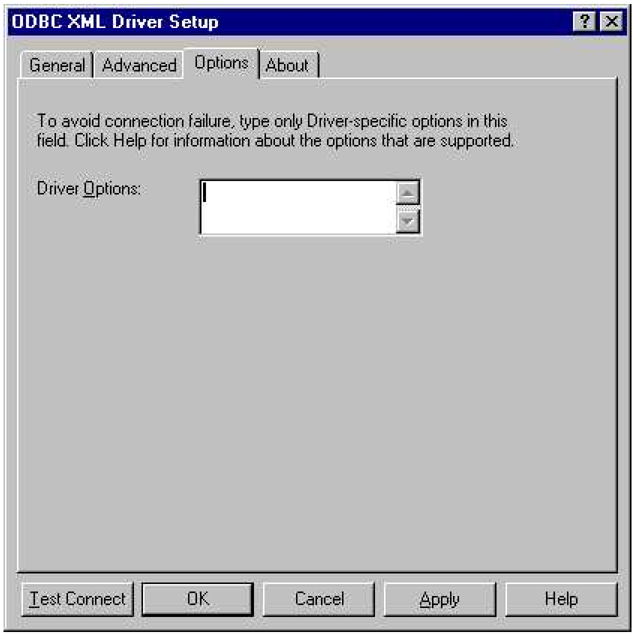
**International Sort Order:**

This option affects the order that records are retrieved when issuing an ORDER BY clause in the SELECT statement passed to this data source. For more information on this option, please see the CR XML ODBC Online help, via the Help button.

**Delete Linked Schema**: This option does not directly affect reporting. For more information on this option, please see the CR XML ODBC Online help, via the Help button.

**Logging:** This option does not directly affect reporting. For more information on this option, please see the CR XML ODBC Online help by pressing the Help (F1) button.

### Options Tab



To avoid connection failure, type only Driver-specific options in this field. Click Help for information about the options that are supported.

Driver Options:

In the **ODBC XML Driver Setup** dialog box, under the **Options** tab is the **Driver Options**.

The **Driver Option** does not directly affect reporting. For more information on this option, please see the CR XML ODBC Online help by pressing the Help (F1) button.

## XML Formats That Crystal Reports Can Report Off

Crystal Reports, via the CR XML ODBC and native drivers, can report off 3 different XML formats:

- Raw
- ADO
- Data Island.

These three formats use a hierarchical structure and all formats use opening and closing XML tags to define each data element.

Crystal Reports, via the CR XML native driver only, can also report off:

- XSD files

### Raw XML

Unlike ADO and Data Island formats, Raw XML does not explicitly define a schema (data structure). The CR XML ODBC driver will apply a structure for the XML document based entirely on the XML data contained within that document.

Example:

```
<Sample_Raw_XML_Document>
  <record>
        <MovieName>Star Wars</MovieName>
        <LeadActor>Mark Hamill</LeadActor>
        <YearReleased>1977</YearReleased>
        <MovieDescription>Set "a long long time ago in a
        galaxy far, far away", Star Wars set the standard
        for science fiction films.</MovieDescription>
  </record>
  <record>
        <MovieName>The Terminator</MovieName>
        <LeadActor>Arnold Schwarzenegger</LeadActor>
        <YearReleased>1984</YearReleased>
        <MovieDescription>This film placed Arnold
        Schwarzenegger in the A list of Hollywood
        actors.</MovieDescription>
  </record>
 </Sample_Raw_XML_Document>
```

| | |
|---|---|
| **NOTE** | This sample XML script can be pasted into a normal text editor and saved as an XML file. |

## ADO XML

ADO XML documents contain the data schemas (structures) followed by the actual data.

Fields for this XML code are defined within the <s:AttributeType> tags in the first block of code. Each field is primarily defined with a name and a data type. Secondary attributes, such as field precision, are also included. Data for these fields is contained within the <rs:data> tags.

Example:

```
<xml xmlns:s="uuid:BDC6E3F0-6DA3-11d1-A2A3-
00AA00C14882" xmlns:dt="uuid:C2F41010-65B3-11d1-A29F-
00AA00C14882" xmlns:rs="urn:schemas-microsoft-com:rowset"
xmlns:z="#RowsetSchema">
  <s:Schema id="RowsetSchema">
    <s:ElementType name="row" content="eltOnly">
          <s:AttributeType name="MOVIENAME" rs:number="1"
          rs:nullable="true" rs:writeunknown="true">
                <s:dataType dt:type="string"
                dt:maxLength="25" rs:precision="25" />
         </s:AttributeType>
         <s:AttributeType name="LEADACTOR" rs:number="2"
          rs:nullable="true" rs:writeunknown="true">
                <s:dataType dt:type="string"
                dt:maxLength="30" rs:precision="30" />
         </s:AttributeType>
         <s:AttributeType name="YEARRELEASED"
            rs:number="3" rs:nullable="true"
            rs:writeunknown="true">
                 <s:dataType dt:type="string"
                   dt:maxLength="4" rs:precision="4" />
         </s:AttributeType>
         <s:extends type="rs:rowbase" />
    </s:ElementType>
  </s:Schema>

  <rs:data>
      <z:row MOVIENAME="Star Wars" LEADACTOR="Mark
         Hamill" YEARRELEASED="1977" />
      <z:row MOVIENAME="The Terminator"
         LEADACTOR="Arnold Schwarzenegger"
         YEARRELEASED="1984" />
  </rs:data>
</xml>
```

| | |
|---|---|
| **NOTE** | This sample XML script can be pasted into a normal text editor and saved as an XML file. |

## Data Island XML

Data Island XML differs from ADO XML in that the data schema (structure) is held in a separate XML Data Definition document. The primary XML data document holds the actual data and references the Data Definition document. The Data Definition document should reside in the same location as the primary XML data document.

Example:

```
<?xml version="1.0" ?>
<XML_DATA_ISLAND_SAMPLE xmlns="x-schema:DD_16394.xml">
  <_Record>
     <MOVIENAME>Star Wars</MOVIENAME>
     <LEADACTOR>Mark Hamill</LEADACTOR>
     <YEARRELEASED>1977</YEARRELEASED>
  </_Record>
  <_Record>
     <MOVIENAME>The Terminator</MOVIENAME>
     <LEADACTOR>Arnold Schwarzenegger</LEADACTOR>
     <YEARRELEASED>1984</YEARRELEASED>
  </_Record>
</XML_DATA_ISLAND_SAMPLE>
```

All data is held within the <XML_DATA_ISLAND_SAMPLE> tags, where XML_DATA_ISLAND_SAMPLE is the name of the primary XML data document. The opening <XML_DATA_ISLAND_SAMPLE> tag also specifies the Data Definition document that provides the schema information for the data, **DD_16394.xml** in this case. The code for this document is shown below.

```
<Schema xmlns="urn:schemas-microsoft-com:xml-data"
  xmlns:dt="urn:schemas-microsoft-com:datatypes">
  <ElementType name="XML_DATA_ISLAND_SAMPLE"
     content="eltOnly" order="many">
       <element type="_Record" />
  </ElementType>
  <ElementType name="_Record" content="eltOnly"
     order="many">
       <element type="MOVIENAME" />
     <element type="LEADACTOR" />
     <element type="YEARRELEASED" />
  </ElementType>
  <ElementType name="MOVIENAME" content="textOnly"
   dt:type="string" />
  <ElementType name="LEADACTOR" content="textOnly"
   dt:type="string" />
  <ElementType name="YEARRELEASED" content="textOnly"
   dt:type="i2" />
</Schema>
```

| NOTE | • These sample XML scripts can be pasted into a normal text editor and saved as XML files.<br>• Data Island schemas define data types for fields but do not define field precisions (like String length). |
|---|---|

### XSD

XSD files are models for describing and validating the structure of an XML environment. XSD files point to data sources. Crystal Reports can connect to these files natively.

## XML Data Types

The following table lists the data types used in XML and the data types as they appear in Crystal Reports.

| NOTE | These data types can be specified for ADO and Data Island XML formats. Since Raw XML does not include schema (data structure) information, data types cannot be specified for Raw XML format. |
|---|---|

| XML Data Type | Description | Crystal Reports Data Type |
|---|---|---|
| BOOLEAN | 0 or 1, where 0 = "false" and 1 ="true" | BOOLEAN |
| STRING (255 characters or less) | Character data up to 255 characters in length | STRING |
| STRING (256 characters or | Character data over 255 characters in length | MEMO |

| | | |
|---|---|---|
| more) | | |
| CURRENCY | Money data, with a decimal precision of 2 | NUMBER |
| DATE | Date in a subset ISO 8601 format, without the time data. For example: "1994-11-05". | DATE |
| FIXED.x.y | Same as "number" but no more than x=14 digits to the left of the decimal point, and no more than y=4 to the right. | NUMBER |
| FLOAT | Positive or negative value, values range from 2.2250738585072014E-308 to 1.7976931348623157E+308 | NUMBER |
| I1 | Integer represented in one byte. A number, with optional sign, no fractions, no exponent. For example: "1, 127, -128". Values range from –128 to 127. | NUMBER |
| I2 | Integer represented in one word (2 bytes). A number, with optional sign, no fractions, no exponent. For example: "1, 703, -32768". Values range from –32768 to 32767. | NUMBER |
| I4 | Integer represented in 4 bytes. A number, with optional sign, no fractions, no exponent. For example: "1, 703, -32768, 148343, -1000000000". Values range from –2147483648 to 2147483647. | NUMBER |
| INT | Number, with optional sign, no fractions, and no exponent. | NUMBER |
| NUMBER | Number, with no limit on digits; can potentially have a leading sign, fractional digits, and optionally an exponent. Values have same range as most significant number, R8, 1.7976931348623157E+308 to 2.2250738585072014E-308 | NUMBER |

| | | |
|---|---|---|
| R4 | Real number, with no limit on digits; can potentially have a leading sign, fractional digits, and optionally an exponent. Values range from 3.40282347E+38F to 1.17549435E-38F. | NUMBER |
| R8 | Same as "float." Real number, with no limit on digits; can potentially have a leading sign, fractional digits, and optionally an exponent. Values range from 1.7976931348623157E+308 to 2.2250738585072014E-308. | NUMBER |
| TIME | Time in a subset ISO 8601 format, with no date and no time zone. For example: "08:15:27" | TIME |
| UI1 | Unsigned integer. A number, unsigned, no fractions, no exponent. For example: "1, 25 | NUMBER |
| UI2 | Unsigned integer, two bytes. A number, unsigned, no fractions, no exponent. For example: "1, 255, 65535". | NUMBER |
| UI4 | Unsigned integer, four bytes. A number, unsigned, no fractions, no exponent. For example: "1, 703, 3000000000". | NUMBER |

## Table and Row Hints

XML documents are set up in a hierarchical structure. They begin with parent tags containing child tags, which contain child tags of those tags, and so on. However, the CR XML ODBC driver is designed to understand only one hierarchy level. As a result, table and row hints may be needed to indicate which hierarchy level represents the table and which hierarchy level represents the rows.

Here is an example Raw XML code that illustrates the use of hints.

```
<?xml version="1.0" encoding="UTF-8" ?>
    <Inventory>
    <Books>
      <Fiction>
        <Item>
          <CODE>16-041</CODE>
          <CATEGORY>Horror</CATEGORY>
          <RELEASE_DATE>1998-03-07</RELEASE_DATE>
          <TITLE>IT</TITLE>
          <SALES>127853</SALES>
      </Item>
        <Item>
          <CODE>16-048</CODE>
          <CATEGORY>Science Fiction</CATEGORY>
          <RELEASE_DATE>1998-04-21</RELEASE_DATE>
          <TITLE>Contact</TITLE>
          <SALES>375298</SALES>
      </Item>
        <Item>
          <CODE>16-105</CODE>
          <CATEGORY>Classics</CATEGORY>
          <RELEASE_DATE>1998-05-10</RELEASE_DATE>
          <TITLE>Diary of Samuel Pepys</TITLE>
          <SALES>297311</SALES>
      </Item>
    </Fiction>
    <Non-Fiction>
        <Item>
          <CODE>17-041</CODE>
          <CATEGORY>HTML</CATEGORY>
          <RELEASE_DATE>1998-03-07</RELEASE_DATE>
          <TITLE>Instant HTML</TITLE>
          <SALES>127853</SALES>
      </Item>
        <Item>
          <CODE>17-048</CODE>
          <CATEGORY>Scripting</CATEGORY>
          <RELEASE_DATE>1998-04-21</RELEASE_DATE>
          <TITLE>Instant JavaScript</TITLE>
          <SALES>375298</SALES>
      </Item>
        <Item>
          <CODE>17-105</CODE>
          <CATEGORY>ASP</CATEGORY>
          <RELEASE_DATE>1998-05-10</RELEASE_DATE>
          <TITLE>Instant Active Server Pages</TITLE>
          <SALES>297311</SALES>
      </Item>
    </Non-Fiction>
  </Books>
</Inventory>
```

| | |
|---|---|
| **NOTE** | This sample XML script can be pasted into a normal text editor and saved as an XML file. |

By examining this code, it is clear that the <Item> tags should represent the rows. Within the <Item> tags, <CODE>, <CATEGORY>, <RELEASE_DATE>, <TITLE>, and <SALES> are field tags that contain the data. <Item> can be identified as the row hierarchy level by using row hints.

<Fiction> and <Non-Fiction> both behave as parent tags of <Item> and should represent tables in this XML code. Table hints can be specified

for <Fiction> and <Non-Fiction> but not at the same time since <Fiction> and <Non-Fiction> represent different hierarchies on the same level.

Table and Row Hints can be specified in the CR XML ODBC driver data source setup in the **Advanced** tab:

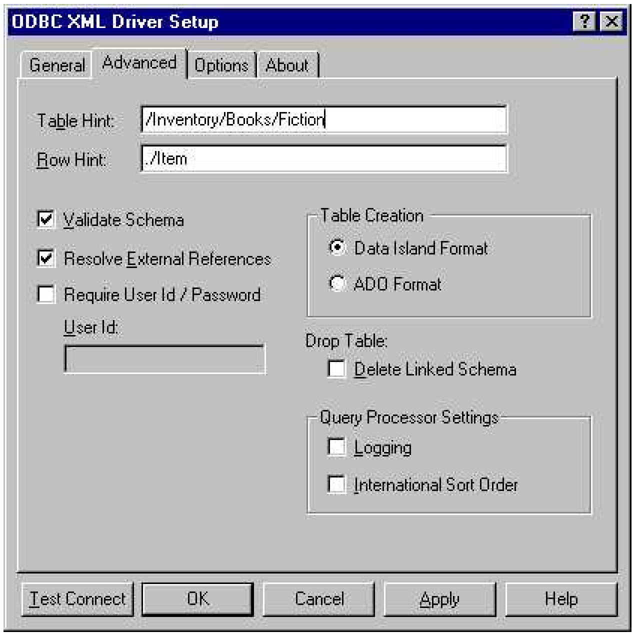


Table Hint: /Inventory/Books/Fiction

Row Hint: ./Item

☑ Validate Schema

☑ Resolve External References

□ Require User Id / Password

User Id:

Table Creation

● Data Island Format

○ ADO Format

Drop Table:

□ Delete Linked Schema

Query Processor Settings

□ Logging

□ International Sort Order

If the <Fiction> hierarchy is to be used, table and row hints can be specified using either of these methods:

- **Method One:**

  Table hint: /Inventory/Books/Fiction

  Row hint: ./Item

  The table hint specifies the <Fiction> hierarchy as the table and the row hint indicates the <Item> tag, within <Fiction>, as the row. Notice that there is a "**.**" at the beginning of the row hint.

| | |
|---|---|
| **NOTE** | A "**.**" is included at the beginning of the row hint to specify that on the <Item> elements from the <Fiction> hierarchy are to be used. |

- **Method Two**:

  Table hint:

  Row hint: /Inventory/Books/Fiction/Item

  The row hint is specified by completely qualifying the <Item> tag.

| | |
|---|---|
| **NOTE** | • The <Non-Fiction> hierarchy for table and row hints is specified in the same way.<br>• Table and Row Hints are case sensitive |

## Additional Table and Row Hint Examples From Online Help

This section provides XML code and example hints. There are 3 Raw XML file samples, followed by 6 XML code samples. This XML code also appears in the Crystal Reports Online Help.

| | |
|---|---|
| **CAUTION** | When changing the hints in the CR XML ODBC driver, the ODBC data source must be logged off and on again (in Crystal Reports Designer) before the new hints can take effect. |

### BookstoreHintSample1.XML:

```
  <Bookstore>
    <Books>
      <Item>
        <CODE>16-041</CODE>
        <CATEGORY>HTML</CATEGORY>
        <RELEASE_DATE>1998-03-07</RELEASE_DATE>
        <TITLE>Instant HTML</TITLE>
        <SALES>127853</SALES>
    </Item>
      <Item>
        <CODE>16-048</CODE>
        <CATEGORY>Scripting</CATEGORY>
        <RELEASE_DATE>1998-04-21</RELEASE_DATE>
        <TITLE>Instant JavaScript</TITLE>
        <SALES>375298</SALES>
    </Item>
      <Item>
        <CODE>16-105</CODE>
        <CATEGORY>ASP</CATEGORY>
        <RELEASE_DATE>1998-05-10</RELEASE_DATE>
        <TITLE>Instant Active Server Pages</TITLE>
        <SALES>297311</SALES>
    </Item>
  </Books>
</Bookstore>
```

### BookstoreHintSample2.XML:

```
  <Bookstore location="Raleigh">
    <Books>
      <Item>
        <CODE>16-041</CODE>
        <CATEGORY>HTML</CATEGORY>
        <RELEASE_DATE>1998-03-07</RELEASE_DATE>
        <TITLE>Instant HTML</TITLE>
        <SALES>127853</SALES>
    </Item>
      <Item>
        <CODE>16-048</CODE>
```

```
            <CATEGORY>Scripting</CATEGORY>
            <RELEASE_DATE>1998-04-21</RELEASE_DATE>
            <TITLE>Instant JavaScript</TITLE>
            <SALES>375298</SALES>
      </Item>
         <Item>
            <CODE>16-105</CODE>
            <CATEGORY>ASP</CATEGORY>
            <RELEASE_DATE>1998-05-10</RELEASE_DATE>
            <TITLE>Instant Active Server Pages</TITLE>
            <SALES>297311</SALES>
      </Item>
       </Books>
    </Bookstore>
```

**BookstoreHintSample3.XML:**

```
  <Bookstore>
     <Books>
        <Item CODE="16-041" CATEGORY="HTML"
           RELEASE_DATE="1998-03-07"
         TITLE="Instant HTML" SALES="127853" />
         <Item CODE="16-048" CATEGORY="Scripting"
           RELEASE_DATE="1998-04-21"
           TITLE="Instant JavaScript" SALES="375298" />
         <Item CODE="16-105" CATEGORY="ASP"
           RELEASE_DATE="1998-05-10"
           TITLE="Instant Active Server Pages"
           SALES="297311" />
     </Books>
   </Bookstore>
```

**Example 1:**

Table Hint:

Row Hint: //Item

Use with file: BookstoreHintSample1.xml

The row nodes are the nodes named Item. The table node is the parent of the row nodes. Use this form only when all of the Item nodes reside under one parent. If some Item nodes have different parents, use a table hint or a more specific row hint to select the set of Item nodes.

| | |
|---|---|
| **NOTE** | Nodes can be thought of as levels or branches in a hierarchy in an XML document. |

**Example 2:**

Table Hint:

Row Hint: /Bookstore/Books/Item

Use with file: BookstoreHintSample1.xml

The row nodes are the nodes named Item. The table node is Books, which is a child of the Bookstore node.

**Example 3:**

Table Hint: /Bookstore/Books

Row Hint:

Use with file: BookstoreHintSample1.xml

The table node is Books, which is a child of the Bookstore node. The row nodes are the children of the Books node. It is assumed that all of the child nodes under the Books nodes have the same name. If the child nodes do not all have the same name, the name of the first child node encountered is used as the row node name. In that case, it would be better to specify both a table and row hint.

**Example 4:**

Table Hint: /Bookstore [@location = "Raleigh"]/Books

Row Hint: ./Item

Use with file: BookstoreHintSample2.xml

The table node is Books, which is a child of the Bookstore node. Bookstore has a "location" attribute with the value Raleigh. The row nodes are the Item nodes that are children of the Books node.

| | |
|---|---|
| **NOTE** | The next 2 examples show the use of Column Mode Identifiers. See the Crystal Reports Online Help for details. |

**Example 5:**

Table Hint:

Row Hint: //Item \column attr

Use with file: BookstoreHintSample3.xml

The row nodes are named Item. The table node is the parent of the row nodes. The columns are attributes of the row node.

**Example 6:**

Table Hint:

Row Hint: //Item \column child

Use with file: BookstoreHintSample1.xml

The row nodes are the nodes named Item. The table node is the parent of the row nodes. The columns are attributes of the row node.

# Additional Configuration For XML

## Multiple XML Documents Vs Multiple Tables In One XML Document

It is possible to have more than one table in an XML document (see Table and Row Hints). However, if these tables are to be linked in Crystal Reports, it is more efficient to have these tables in separate XML documents (in the same folder on your computer).

Having separate XML documents allows each XML document to act as a table. When you link these multiple tables, Crystal Reports creates a single SQL statement to retrieve records based on the table joins from one ODBC data source.

When multiple tables are contained within one XML document, one ODBC data source needs to be created and configured for each table referenced in the XML document. Hence, multiple ODBC data sources would need to be linked in a report. This would reduce performance since multiple SQL statements need to be passed to ODBC and Crystal Reports would be responsible for filtering records based on the table links.

| | |
|---|---|
| **NOTE** | For more information on working with ODBC data sources in Crystal Reports, please refer to the Crystal Reports User's Guide. |

## Configuration of String and Memo Fields

In CR, fields greater than 255 characters are considered Memo fields. In CR 8.5, the CR XML ODBC driver cannot report off Memo fields.

By default, the CR XML ODBC driver will truncate all Memo fields at 255 characters and read them as String fields. This is done to take advantage of sorting, grouping, conditional formatting, and record selection on these fields in Crystal Reports.

CR 9 can report off Memo fields.

# Connecting Natively to an XML or XSD file

The following prerequisites are required to connect natively to an XML or XSD file:

- Supported Operating systems from .NET include Windows 2000,Windows XP, and Windows CE.

- The Microsoft .Net framework must be installed. (This can be obtained from Microsoft's website).

- Crystal Reports 9 Developer and Advanced Developer editions only.

### Steps to Connect

1. Copy the crdb_a~3.dll from the Crystal Reports 9 Developer CD to the C:\Program Files\Common Files\Crystal Decisions\2.0\Bin folder.

2. Rename this file to crdb_adoplus.dll

3. In Crystal Reports 9, on the **File** menu, click **New**.

4. Select the **As a Blank Report** option button and then click **OK**.

5. In the **Database Expert** dialog box, double-click **Create New Connection**, and then double-click **Database Files**. Browse to your xml file.

| | |
|---|---|
| **NOTE** | When the data or structure has changed you must log off/server and then log back on. The data changes will be seen within the report. |

## Creating XML Files Through The CR XML ODBC Driver

XML documents can be created via the CR XML ODBC driver.  The CR XML ODBC driver can be configured to produce ADO, Data Island or DataDirect XML documents.

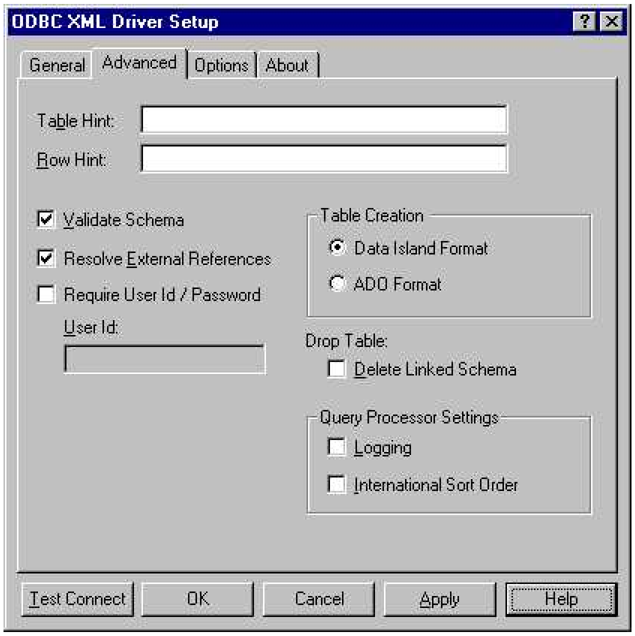


Table Hint:

Row Hint:

☑ Validate Schema

☑ Resolve External References

□ Require User Id / Password

User Id:

Table Creation

● Data Island Format

○ ADO Format

Drop Table:

□ Delete Linked Schema

Query Processor Settings

□ Logging

□ International Sort Order

The Table Creation frame in the **Advanced** tab of the **ODBC XML Driver Setup** dialog box controls the format of the XML document(s) that are created, whether it be Data Island Format or ADO Format. Raw XML cannot be created through the CR XML ODBC driver.

If the XML documents are created by exporting to a CR XML ODBC data source from Crystal Reports, the report requires the proper format to produce a readable XML document. Format the report with the following:

- Only one Details section.
- All fields to be created in the XML document are inserted into the Details section. If all the fields do not fit on one line, reduce the width of the field objects so that all the fields can be contained on one line in the Details section. It does not matter that the data from these fields can only be partially seen.
- Export the report to a pre-configured ODBC data source for the CR XML ODBC driver. An XML document name can be specified at this time.

| NOTE | Although not generally recommended due to slow performance, this method is an effective way of creating simple XML documents. To create more detailed XML documents, it is recommended to use the Crystal Reports Export to XML feature. For more information on this feature, please refer to the Crystal Reports Online Help and the Crystal Reports User’s Guide. |
|---|---|

# Limitations

For information on limitations with the XML 4.10.00.00 ODBC driver refer to the following knowledge base articles:

C2014594 - Cannot save changes to table and row hints in an XML ODBC data source

C2014595 - Err Msg: "The database table <table name> cannot be found..." connecting to XML

C2014596 - Crystal Reports does not recognize some XML field's data types correctly

C2014597 - Changes to XML data are not recognized in Crystal Reports

The knowledge base can be found at:

http://support.businessobjects.com/search

# Finding More Information

For more information on Crystal Reports and XML, please refer to the Crystal Reports User's Guide and the Crystal Reports Online Help.

For information about working with the Native XML driver for Crystal Reports, search for cr_xi_native_xml_driver.pdf at:

http://support.businessobjects.com/search

In addition, search for XML knowledge base articles at:

http://support.businessobjects.com/search

# Contacting Business Objects for Technical Support

We recommend that you refer to the product documentation and that you visit our Technical Support web site for more resources.

**Self-serve Support:**
http://support.BusinessObjects.com/

**Email Support:**
http://support.BusinessObjects.com/support/answers.asp

**Telephone Support:**
http://www.BusinessObjects.com/contact/support.asp

►