

Data Format Description Language (DFDL) message modeling In Websphere Message Broke

Sandipan Chakraborty

May 08, 2013

This article shows you how to use the new DFDL support in WebSphere Message Broker V8 to model, create, run, and test messages in both Text and Comma Separated Value (CSV) formats.

Introduction

DFDL is a XML-based standard that you can use to define the structure of formatted data independently from the data format itself. The following new function in IBM® WebSphere® Message Broker V8 (hereafter called Message Broker) enables you to create, run, and test DFDL models:

- DFDL Parser and domain
- DFDL Schema Creation Wizards
- DFDL Schema Editor to model text and binary data formats
- DFDL Test Perspective to test your DFDL schema files

Creating a DFDL CSV message model

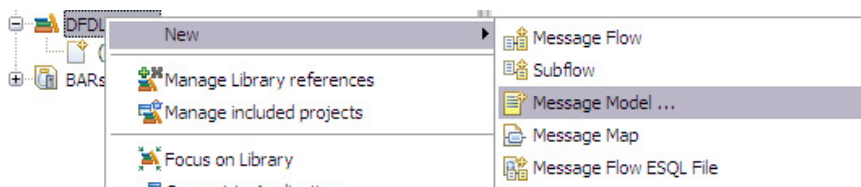
Here is some sample CSV data with a header record, several detail records, but no trailer record. Each row is delimited by newline, and fields within each row are delimited by commas:

```
EMP_NM, EMP_ID, EMP_DEPT
Sandi, 1234, ABC
John, 0987, JPQ
Elex, 7834, MNO
```

Here are the steps to creating a DFDL CSV message model:

1. Create a library named `DFDLModelLib`, right click-on it, and select **New => Message Model**:

Figure 1



2. Select **CSV text** and click **Next**:

Figure 2

New Message Model

Create a new message model file
Select the message model type or format

XML

- ☐ **SOAP XML** XML data for use in Web Services.
- ☐ **Other XML** All other XML data.

Text and binary

- ☒ **CSV text** Comma Separated Values data, a delimited text format commonly used as an export format by spreadsheets and databases.
- ☐ **Record-oriented text** Text data formats where delimited fields are grouped into records.
- ☐ **COBOL** Data for COBOL programs
- ☐ **C** Data for C programs
- ☐ **Other text or binary** All other text or binary data formats.

Enterprise Information Systems

- ☐ **SAP** Data from SAP systems including IDoc and BAPI
- ☐ **Siebel** Data from Siebel systems
- ☐ **PeopleSoft** Data from PeopleSoft
- ☐ **JD Edwards** Data from JD Edwards systems

Other

- ☐ **CORBA IDL** Data from CORBA
- ☐ **Database record** Records from relational databases
- ☐ **MIME** Data for extended email format
- ☐ **IBM supplied** Predefined data format

3. Choose how you want to create your message model. Select option one, which creates a DFDL schema with the default element name (you can change it later). Click **Next**:

Figure 3

CSV text

Choose how you would like to create your CSV message model.

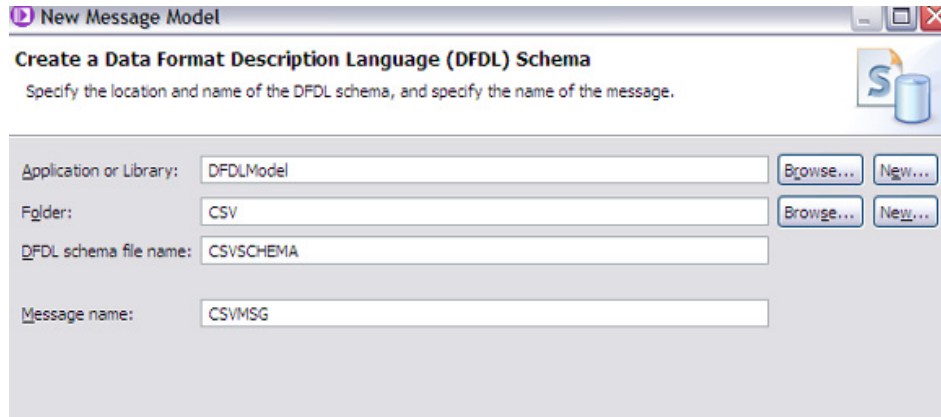
WebSphere Message Broker requires a message model in order to parse, serialize and validate CSV data. A message model also speeds up development of your message broker applications by enabling ESQL content assist and graphical maps.

- ☒ **Create a DFDL schema file using this wizard to guide you.**
- ☐ Create an empty DFDL schema file, I will model my data using the DFDL schema editor
- ☐ Import or replace the IBM supplied DFDL schema property defaults for CSV.

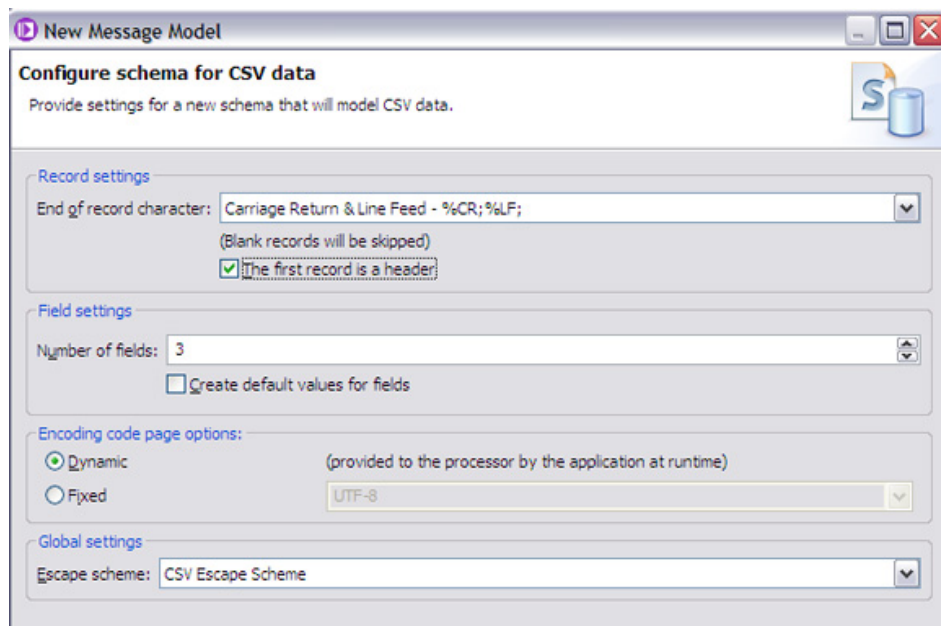
	A	B	C	D	E
1	Year	Make	Model	Description	Price
2	2009	SK Inc	MBTk7	4293cc, V8	53880.00
3	2010	Hans On	DFDL	3000cc straight 6	31395.00
4	2010	AOD corp	MB8	4163cc, V8	51435.00

Export

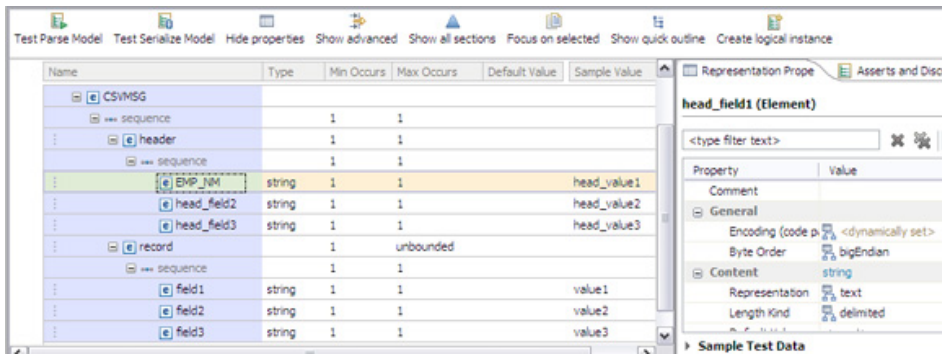
4. Provide the details as shown below. Create the csv folder in DFDLModelLib, and then click **Next**:

Figure 4

5. Provide details about the number of fields in the CSV message that is being modeled -- in this case three. Select **First Record Is Header**, and select **End of record character** to specify that each record is separated by newline. Click **Finish**:

Figure 5

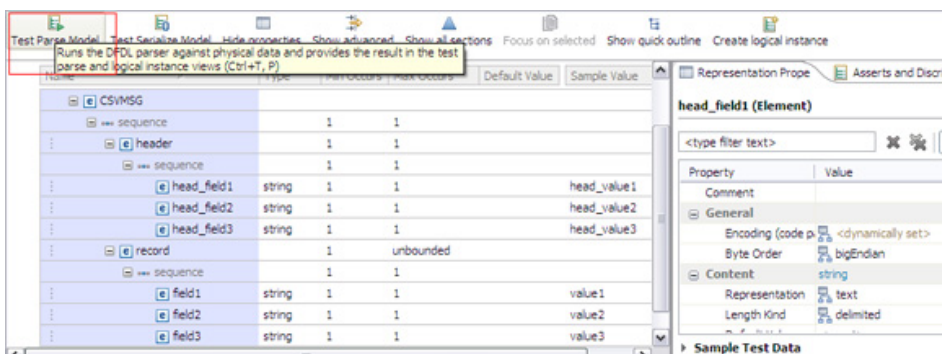
6. Here is the modeled CSV message. You can edit the Header or Record field name -- as shown below, the Header field name has been changed to EMP_NM:

Figure 6

Testing the DFDL CSV message model

In a Message Broker V8 DFDL message model, you can immediately test and validate a message model with sample data:

1. Click on **Test Parse Model**:

Figure 7

2. Provide the path of the sample CSV file and click **OK**:

Figure 8

Message
Select message for testing. [More...](#)
Message name:* CSVMSG

Parser Input
Select content to be parsed against schema.
☐ Content from 'DFDL Test - Serialize' view
☒ Content from a data file
Input file name:* /DFDLModel/csvmsg.txt [Browse...](#)

Specify runtime configuration.

Runtime encoding options
Provide runtime values for properties which have been configured in the model to be dynamically set. [More...](#)
Encoding (code page): UTF-8
Floating point format: IEEE Non-Extended
Byte order: ☐ Little endian ☒ Big endian

Runtime validation
☐ Validate data against schema [More...](#)

3. The sample CSV file was successfully parsed by the DFDL CSV parser:

Figure 9

DFDL Test - Parse: Runs the DFDL parser with the provided physical input data and selected message, and updates the logical instance view with the result of the parse.
Status: Parsing completed: Tue Apr 02 12:30:00 IST 2013

Input
Data: /DFDLModel/csvmsg.txt [Browse...](#) Message: CSVMSG (/DFDLModel/CSV/CSVSCHEMA.xsd)

Parsed Input

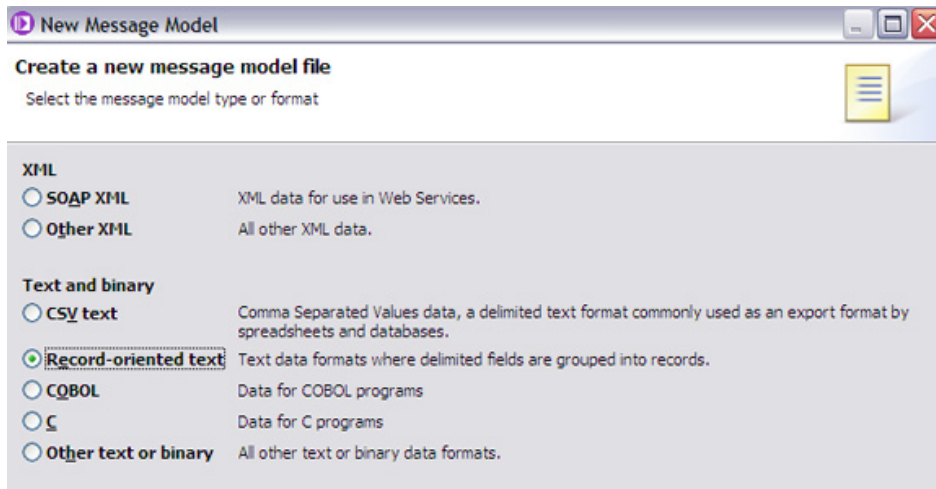
Characters
1 EMP_NM EMP_ID EMP_DEPT
2 Sandi 1234 ABC
3 John 0987 JPQ
4 Alex 7834 MNO

Creating a DFDL Text (TDS) message model

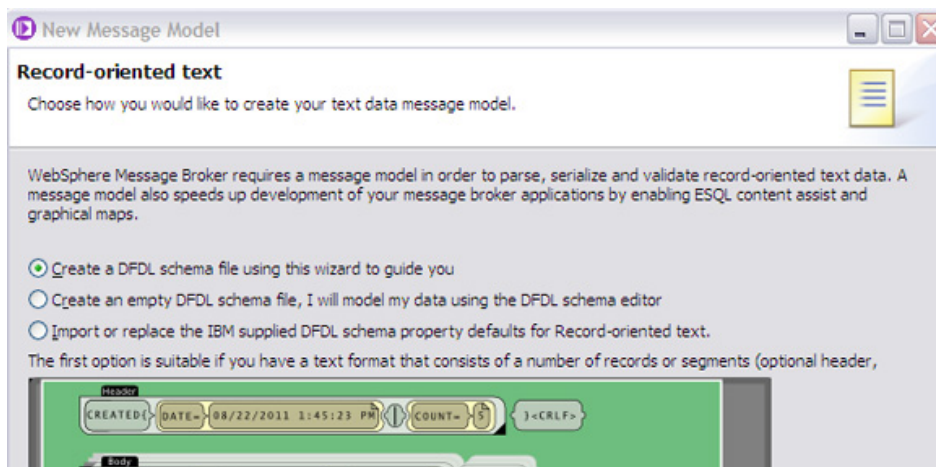
Here is some sample Text (TDS) data. It is comma separated, and it uses **h***, **b***, and **t*** as initiator for header, body, and trailer records respectively. The body data is in bold, along with the initiator **"Add["** and the terminator **"]"**:

```
h*head_val1,head_value2,head_value3
b*body_value1,Add[Add1,Add2],body_value3
t*trailer_value1,trailer_value2,trail_value3
```

1. Right-click on **DFDLModelLib** and select **New => Message Model**.
2. On the Message Model screen, select **Record-Oriented text** and click **Next**:

Figure 10

3. Choose how you want to create your Text message model. Select option one, which creates a DFDL schema with the default element name (you can change if later), and then click **Next**:

Figure 11

4. Provide the details as shown below. Create the `Text` folder in `DFDLModelLib`, and then click **Next**:

Figure 12

The screenshot shows the 'New Message Model' dialog box with the 'Create a Data Format Description Language (DFDL) Schema' tab selected. The dialog prompts the user to specify the location and name of the DFDL schema and the name of the message. The fields are filled with the following values:

- Application or Library: DFDLModel
- Folder: Text
- DFDL schema file name: TextSchema
- Message name: TextMsg

Buttons for 'Browse...' and 'New...' are available for the Application or Library, Folder, and DFDL schema file name fields.

5. This screen defines initiators for the header, body, and trailer; the field separators (commas in this case), and the number of fields in the header, body and trailer. When done, click **Next**:

Figure 13

The screenshot shows the 'New Message Model' dialog box with the 'Configure schema for data formatted as records and fields' tab selected. The dialog prompts the user to provide settings for the new DFDL schema that represent record-oriented data. The settings are as follows:

- Record settings:**
 - End of record character: Carriage Return & Line Feed - %CR;%LF;
 - (Blank records will be skipped)
 - ☒ The first record is a header
 - ☒ The last record is a trailer
- Header fields:**
 - Header initiator: h*
 - Number of fields: 3
- Field settings:**
 - ☒ Separated by: , - %#44; (UTF-8: 0x2C) (UTF-16: 0x002C)
 - ☐ Fixed length
 - ☐ All fields have an initiator
 - ☐ Create default values for fields

6. Here is the modeled text message:

Figure 14

Test Parse Model Test Serialize Model Hide properties Show advanced Show all sections Focus on selected Show quick outline Create logical instance

▼Messages

A message is a global element that models an entire document of data.

Name	Type	Min Occurs	Max Occurs	Default Value	Sample Value
⊖ a					
⊖ sequence		1	1		
⊕ header		1	1		
⊖ body		1	unbounded		
⊖ sequence		1	1		
⊖ body_elem1	string	1	1		body_value1
⊖ body_elem2	string	1	1		body_value2
⊖ body_elem3	string	1	1		body_value3
⊕ trailer		1	1		
Add a Local Element					

7. You still need to modify the message model to accommodate the highlighted text in the body (Add [Add1, Add2]). To model the highlighted text, click on **Type** for **body_elem2**, select **Anonymous**, and then add the local elements **add_elem1** and **add_elem2**. Also provide the initiator and terminator details for body_elem2 shown below:

Figure 15

Name	Type	Min Occurs	Max Occurs	Default Value
⊖ a				
⊖ sequence		1	1	
⊕ header		1	1	
⊖ body		1	unbounded	
⊖ sequence		1	1	
⊖ body_elem1	string	1	1	
⊖ body_elem2		1	1	
⊖ sequence		1	1	
⊖ add_elem1	string	1	1	
⊖ add_elem2	string	1	1	
⊖ body_elem3	string	1	1	
⊕ trailer		1	1	
Add a Local Element				

<type filter text>

Property	Value
Comment	
General	
Encoding (code page)	<dynamic>
Byte Order	<dynamic>
Content	
Length Kind	delimited
Occurrences	
Min Occurs	1
Max Occurs	1
Delimiters	
Initiator	Add[
Terminator]

Testing the DFDL Text (TDS) message model

- 1. Click **Test Parse Model** and provide the file path for the sample text data:

Figure 16

2. The sample Text file was successfully parsed by the DFDL Parser:

Figure 17

Conclusion

WebSphere Message Broker V8 now enables you to model, parse, and test DFDL message models. You can develop and test model definitions in the WebSphere Message Broker Toolkit without deploying to a running broker instance. The DFDL Parser provides an efficient and open-standards-based way to access, parse, and transform data in many different formats, with a focus on text and binary data.

Related topics

- **Websphere Message Broker resources**

- [WebSphere Message Broker V8 information center](#)
A single Web portal to all WebSphere MQ V7 documentation, with conceptual, task, and reference information on installing, configuring, and using WebSphere Message Broker V8.
- [WebSphere Message Broker product page](#)
Product descriptions, product news, training information, support information, and more.
- [Download free trial version of WebSphere Message Broker](#)
WebSphere Message Broker is an ESB built for universal connectivity and transformation in heterogeneous IT environments. It distributes information and data generated by business events in real time to people, applications, and devices throughout your extended enterprise and beyond.
- [WebSphere Message Broker documentation library](#)
WebSphere Message Broker specifications and manuals.
- [WebSphere Message Broker forum](#)
Get answers to technical questions and share your expertise with other WebSphere Message Broker users.
- [WebSphere Message Broker support page](#)
A searchable database of support problems and their solutions, plus downloads, fixes, and problem tracking.
- [IBM Training course: WebSphere Message Broker V8 Development](#)
This course from IBM Training shows you how to use the components of the WebSphere Message Broker development and runtime environments to develop and troubleshoot message flows that use ESQL, Java, and PHP to transform messages.
- [Youtube tutorial: Integrating Microsoft .NET code in a WebSphere Message Broker V8 message flow](#)
This five-minute youtube tutorial shows you how simple it is to use WebSphere Message Broker V8 to build a message flow that includes Microsoft .NET code. Microsoft Visual Studio is used to build .NET code in C#, which is then integrated into a message flow using Message Broker and an HTTP RESTful interface.

- **WebSphere resources**

- [developerWorks WebSphere](#)
Technical information and resources for developers who use WebSphere products. developerWorks WebSphere provides product downloads, how-to information, support resources, and a free technical library of more than 2000 technical articles, tutorials, best practices, IBM Redbooks, and online product manuals.
- [developerWorks WebSphere application integration developer resources](#)
How-to articles, downloads, tutorials, education, product info, and other resources to help you build WebSphere application integration and business integration solutions.
- [Most popular WebSphere trial downloads](#)
No-charge trial downloads for key WebSphere products.
- [WebSphere forums](#)

Product-specific forums where you can get answers to your technical questions and share your expertise with other WebSphere users.

- [WebSphere demos](#)

Download and watch these self-running demos, and learn how WebSphere products can provide business advantage for your company.

- [WebSphere-related articles on developerWorks](#)

Over 3000 edited and categorized articles on WebSphere and related technologies by top practitioners and consultants inside and outside IBM. Search for what you need.

- [developerWorks WebSphere weekly newsletter](#)

The developerWorks newsletter gives you the latest articles and information only on those topics that interest you. In addition to WebSphere, you can select from Java, Linux, Open source, Rational, SOA, Web services, and other topics. Subscribe now and design your custom mailing.

- [WebSphere-related books from IBM Press](#)

Convenient online ordering through Barnes & Noble.

- [WebSphere-related events](#)

Conferences, trade shows, Webcasts, and other events around the world of interest to WebSphere developers.

- **developerWorks resources**

- [Trial downloads for IBM software products](#)

No-charge trial downloads for selected IBM® DB2®, Lotus®, Rational®, Tivoli®, and WebSphere® products.

- [developerWorks business process management developer resources](#)

BPM how-to articles, downloads, tutorials, education, product info, and other resources to help you model, assemble, deploy, and manage business processes.

- [developerWorks blogs](#)

Join a conversation with developerWorks users and authors, and IBM editors and developers.

- [developerWorks tech briefings](#)

Free technical sessions by IBM experts to accelerate your learning curve and help you succeed in your most challenging software projects. Sessions range from one-hour virtual briefings to half-day and full-day live sessions in cities worldwide.

- [developerWorks podcasts](#)

Listen to interesting and offbeat interviews and discussions with software innovators.

- [developerWorks on Twitter](#)

Check out recent Twitter messages and URLs.

- [IBM Education Assistant](#)

A collection of multimedia educational modules that will help you better understand IBM software products and use them more effectively to meet your business requirements.

© Copyright IBM Corporation 2013

(www.ibm.com/legal/copytrade.shtml)

[Trademarks](#)

(www.ibm.com/developerworks/ibm/trademarks/)

