



IBM Software Group

# WebSphere® Message Broker V6

## *List Function Enhancements*



@business on demand.

© 2006 IBM Corporation  
Updated October 6, 2006

This presentation discusses the List Function Enhancements for WebSphere Message Broker Version 6.

## Agenda

- List function enhancements
  - ▶ EXISTS
  - ▶ SINGULAR
  - ▶ CARDINALITY
- Summary and references



This topic will discuss the three new list function enhancements for WebSphere Message Broker Version 6. This presentation assumes you have a general understanding of the WebSphere Message Broker product.

## Section

# *List function enhancements*



This section discusses the technical details of the List Function Enhancements implemented in WebSphere Message Broker Version 6.

## List function

- **EXISTS**
  - ▶ TRUE or FALSE
  - ▶ Are there any elements in a list?
- **SINGULAR**
  - ▶ TRUE or FALSE
  - ▶ Is there exactly one element in a list?
- **CARDINALITY**
  - ▶ INTEGER counter
  - ▶ How many elements are in a list?



WebSphere Message Broker V6 provides you with new functionality for lists. For example, you might want to know if any elements exist in a list, whether there is only one element in the list, or how many elements exist in a list with multiple elements. In previous releases, there was no simple, efficient way to determine these characteristics.

## List function limitations in previous releases

- SINGULAR did not exist
- EXISTS could only be used in conjunction with SELECT
- CARDINALITY could only be used with message field references



Before Version 6, the SINGULAR function did not exist. The EXISTS function was limited to be used only with the SELECT statement. CARDINALITY could only be used with message field references.

## List function syntax

```
>--EXISTS--(--ListExpression--)-->>  
>--SINGULAR--(--ListExpression--)-->>  
>--CARDINALITY--(--ListExpression--)-->>
```

where *ListExpression* is any expression which returns a list

- Any SELECT expression,
- LIST constructor
- Field reference having the array indicator [ ]



Version 6 introduces a new function called SINGULAR, which can tell you whether or not there is exactly one element in a list. Version 6 also enhances the previously existing CARDINALITY and EXISTS functions. CARDINALITY can not only operate on a message tree but now can also operate directly on a SELECT result. And conversely, the EXISTS function in Version 6 can operate directly on a message tree. These 3 functions can operate on any SELECT expression, any LIST constructor, or any field references having the array indicator.

## List functions examples

```
-- Build test message
SET OutputRoot.XML.Data.Source.F1[1] = 'F11';
SET OutputRoot.XML.Data.Source.F1[2] = 'F12';
SET OutputRoot.XML.Data.Source.F1[3] = 'F13';

-- Test the exists function on field references
SET OutputRoot.XML.Data.Field.Exists0False =
  EXISTS(OutputRoot.XML.Data.Source.Nonexistent[]); -- doesn't exist
SET OutputRoot.XML.Data.Field.Exists1True =
  EXISTS(OutputRoot.XML.Data.Source[]); -- exists
SET OutputRoot.XML.Data.Field.Exists2True =
  EXISTS(OutputRoot.XML.Data.Source.F1[]); -- exists

-- Test the singular function on field references
SET OutputRoot.XML.Data.Field.Singular0False =
  SINGULAR(OutputRoot.XML.Data.Source.Nonexistent[]); -- not singular
SET OutputRoot.XML.Data.Field.Singular1True =
  SINGULAR(OutputRoot.XML.Data.Source[]); -- one Source
SET OutputRoot.XML.Data.Field.Singular2False =
  SINGULAR(OutputRoot.XML.Data.Source.F1[]); -- many F1
```



Here are examples using the EXISTS function and the SINGULAR function. You can pause this presentation if you want more time to review the examples.

## List functions examples (cont.)

```
-- Build test message
SET OutputRoot.XML.Data.Source.F1[1] = 'F11';
SET OutputRoot.XML.Data.Source.F1[2] = 'F12';
SET OutputRoot.XML.Data.Source.F1[3] = 'F13';

-- Test the cardinality function on field references
SET OutputRoot.XML.Data.Field.Cardinality0 =
  CARDINALITY(OutputRoot.XML.Data.Source.Nonexistent[]); -- zero!
SET OutputRoot.XML.Data.Field.Cardinality1 =
  CARDINALITY(OutputRoot.XML.Data.Source[]); -- one Source
SET OutputRoot.XML.Data.Field.Cardinality3 =
  CARDINALITY(OutputRoot.XML.Data.Source.F1[]); -- three F1

-- Test the singular function on message select results
SET OutputRoot.XML.Data.MessageSelect.Singular0False =
  SINGULAR(SELECT F.* FROM OutputRoot.XML.Data.Source.F1[] AS F
           where F = 'XX'); -- no 'XX' values!
```



Here are examples using the **CARDINALITY** function and the **SINGULAR** function.

The **CARDINALITY** function returns the number of elements in a list; the **SINGULAR** function returns a **BOOLEAN** value indicating whether a list contains exactly one element.

You can pause this presentation if you want more time to review the examples.



## Section

# ***Summary and references***



The last portion of the presentation contains a summary and references.

## Summary

- List function enhancements
  - ▶ EXISTS
  - ▶ SINGULAR
  - ▶ CARDINALITY
- Examples



You have seen the enhancements to the list function. The EXISTS function is new for Version 6. The SINGULAR and CARDINALITY functions have been enhanced to allow you more programming flexibility and efficiency. You have also seen examples using the new and enhanced list functions.

## References

- WebSphere Message Broker library:

<http://www-306.ibm.com/software/integration/wbimessagebroker/library/>

- WebSphere Message Broker Information Center:

<http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp>



# Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	MQSeries	Tivoli
IBM (logo)	Cloudscape	Informix	OS/390	WebSphere
e/logo/business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2006. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.