



Web Dynpro Best Practices: How to Configure the JCo Destination Settings

Jochen Guertler



Copyright

© Copyright 2004 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, and Informix are trademarks or registered trademarks of IBM Corporation in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

Index and Table of Contents

Index and Table of Contents	3
Introduction	4
General Information	4
Parameter: Maximum Connections	4
Example:.....	5
Calculation:	6
Result:.....	7
Parameter: Maximum Pool Size	7
Example:.....	7
Parameter: Connection Timeout.....	8
Parameter: Maximum Wait Time	8
General Remark.....	9
User Authentication is: Defined User.....	9
Metadata Destinations	9
The Author	10

Introduction

Using the adaptive RFC model to get access to an SAP R/3 backend within a Web Dynpro application needs the configuration of JCo destinations. To get the best performance and optimized resource consumption (especially on systems with heavy user load), you have to make sure that these JCo destinations are configured correctly.

For general documentation about configuration of the JCO destination settings, please consult the following SDN paper: [How to Use the Web Dynpro Content Administrator](#).

This article explains the details.

General Information

The JCo destination parameters depend on the way the Web Dynpro application has been designed and how it is used – for example, one user works with one application in one window, with multiple applications in multiple windows, or, in the case of the portal, multiple applications in one window. Therefore, there is no general answer. Also, it depends on how the user is authenticated (SSO Ticket or defined user).

We will now discuss the case in which the SSO ticket version is used. This is the standard configuration in production systems, whereas typically, defined users are used in test and development systems. The following assumes a production system only (using SSO ticket authentication). Also, the following recommendations are only valid for application data destinations (not for metadata destinations). At the end, you will also find a recommendation for metadata destinations.

Parameter: `Maximum Connections`

Basically, the administrator of the system has to determine what the maximum amount of connections per user should be. This needs to be determined for each Application Data destination (also called logical system).

The information needed to determine this is:

1. Which Web Dynpro applications will be used by a user or role?

2. For each of these applications, determine which application data destinations (or logical systems) are required.
3. Each application should specify whether it only uses a single connection to the backend, or whether multiple connections are required. By default, each application uses one connection.
4. How many applications can be launched simultaneously in a single user session?

With this information, the proper settings can be determined for each application destination.

Example:

1a) User in role 'manager' requires 3 different applications, let's say 'App A', 'App B', 'App C'

1b) User in role 'employee' requires 4 different applications, let's say 'App C', 'App D', 'App E', 'App F'

2+3 a) 'App A': 1 Destination: 'WD_LISA_MODELDATA_DEST' Max Connections required per App Instance: 1 (default)

2+3 b) 'App B': 1 Destination: 'WD_MONA_MODELDATA_DEST' Max Connections required per App Instance: 1 (default)

2+3 c) 'App C': 1 Destination: 'WD_LISA_MODELDATA_DEST' Max Connections required per App Instance: 2

2+3 d) 'App D': 1 Destination: 'WD_LISA_MODELDATA_DEST' Max Connections required per App Instance: 1 (default)

2+3 e) 'App E': 1 Destination: 'WD_LISA_MODELDATA_DEST' Max Connections required per App Instance: 2

2+3 f) 'App F': 2 Destinations: 'WD_LISA_MODELDATA_DEST' Max Connections required per App Instance: 2, 'WD_MONA_MODELDATA_DEST' Max Connections required per App Instance: 1 (default)

4a) User 'manager' can open up to 3 browser windows and uses the 3 different page configurations:

- Page 'M1': 'App A' and 'App B' => 1 Connection for 'WD_LISA_MODELDATA_DEST'; 1 Connection for 'WD_MONA_MODELDATA_DEST'
- Page 'M2': 'App B' and 'App C' => 2 Connections for 'WD_LISA_MODELDATA_DEST'; 1 Connection for 'WD_MONA_MODELDATA_DEST'
- Page 'M3': 'App C' => 2 Connections for 'WD_LISA_MODELDATA_DEST'

4b) User 'employee' can open up to 1 browser window and uses the 5 different page configurations:

- Page 'E1': 'App C' => 2 Connections for 'WD_LISA_MODELDATA_DEST'
- Page 'E2': 'App C' and 'App D' => 3 Connections for 'WD_LISA_MODELDATA_DEST'
- Page 'E3': 'App C' and 'App E' => 4 Connections for 'WD_LISA_MODELDATA_DEST'
- Page 'E4': 'App C' and 'App D' and 'App E' => 5 Connections for 'WD_LISA_MODELDATA_DEST'
- Page 'E5': 'App F' => 2 Connections for 'WD_LISA_MODELDATA_DEST'; 1 Connection for 'WD_MONA_MODELDATA_DEST'

Calculation:

For each user and destination, determine the page requiring the maximum number of connections and multiply by the maximum number of concurrent browser windows:

- User 'manager', 'WD_LISA_MODELDATA_DEST' : 2 Connections x 3 (browser windows) = 6
- User 'manager', 'WD_MONA_MODELDATA_DEST' : 1 Connections x 3 (browser windows) = 3
- User 'employee', 'WD_LISA_MODELDATA_DEST' : 5 Connections x 1 (browser windows) = 5
- User 'employee', 'WD_MONA_MODELDATA_DEST' : 1 Connections x 1 (browser windows) = 1

Result:

The maximum number of connections of each destination multiplied with a safety factor of 2 is the minimum number to use for the parameter "maximum connections."

- 'WD_LISA_MODELDATA_DEST' : $6 \times 2 = 12$ connections
- 'WD_MONA_MODELDATA_DEST' : $3 \times 2 = 6$ connections

Remark: the parameter "maximum connections" can be reduced if the number of connections per user is to be limited. But this may lead to severe error messages in the applications and is typically not the ideal way to improve performance or scalability. Use the parameter "PoolSize" instead!

Parameter: `Maximum Pool Size`

The parameter `Maximum Pool Size` is used to specify how many connections should be left open for the current user, even if they are not being used actively.

A pooled RFC connection is allocated much faster than a new connection. Therefore, increasing the pool size will optimize the time needed to get a new connection. On the other hand, if a connection is pooled, it is not available for other users. Typically, ABAP backend systems define an upper limit for the number of RFC connections. Therefore, if this number has been fully allocated by pooled connections, then new users requesting a new connection may not succeed, because of the maximum number of connections allowed by the ABAP system having already been consumed.

It is therefore necessary to weigh out the scalability requirements for the J2EE system and the ABAP backend systems. This value can also be further optimized in combination with the property `Connection Timeout`. (See further below for more details.)

Typically, the maximum pool size is kept very low and may even be set to '0.' In case you have substantially less concurrent J2EE users than the number of concurrent RFC connections to an ABAP system, then the pool size can be increased to improve performance.

Example:

ABAP backend system 'LOU' allows 500 simultaneous RFC connections

ABAP backend system 'VRE' allows 300 simultaneous RFC connections

From the above calculation:

- 6 connections to 'WD_LISA_MODELDATA_DEST' are possible per user
- 3 connections to 'WD_MONA_MODELDATA_DEST' are possible per user

Under the assumption that the two destinations point to different ABAP backends, then a calculation must be made for each ABAP backend

- 'WD_LISA_MODELDATA_DEST' => System 'LOU' => concurrent J2EE users: 30 multiplied with nr. of connections (6) = 180, with 500 allowed => poolsize ~ 2
- 'WD_MONA_MODELDATA_DEST' => System 'VRE' => concurrent J2EE users: 30 multiplied with nr. of connections (3) = 90, with 300 allowed => poolsize ~ 3

Under the assumption that the two destinations point to the same ABAP backends, then a calculation must be made only for one ABAP backend

- 'WD_LISA_MODELDATA_DEST' => System 'LOU'
- 'WD_MONA_MODELDATA_DEST' => System 'LOU' => System 'LOU' => concurrent J2EE users: 30 multiplied with nr. of connections (6+3) = 270, with 500 allowed => poolsize ~ 1

These are the minimum values possible. They can be increased if the parameter connection timeout is decreased appropriately.

Parameter: Connection Timeout

The connection timeout defines how long the pool waits (in seconds) before a destination, which is in the range between 1 and the maximum pool size, is closed by the pool. This allows closing pooled connections after a certain amount of time if the user is no longer active. A typical value here is 10 - 30 seconds. After this time, the additional time needed to fetch a new connection is generally not noticeable by a user of a Web Dynpro application.

Parameter: Maximum Wait Time

The maximum wait time defines how long you can wait (in seconds) before you get an exception after trying to get a destination, even though the maximum destinations are open. In a high-user load scenario, it makes sense to increase this number appropriately (20-60); otherwise, the request may timeout very fast because no new RFC connection can be acquired. In low-user load

scenarios, this number can typically be reduced to a very low number (1-5 seconds), because waiting generally will not lead to success, therefore the error message should be issued as fast as possible.

General Remark

All of the above numbers are highly dependent on the system landscape as well as the user load, as well as the number and connection demand of deployed Web Dynpro applications. Therefore, the examples above only illustrate the process used to obtain the optimal values. Therefore, your personal judgment should also be taken into account when determining the final values.

User Authentication is: `Defined User`

In case you are in a test or a development scenario, then it is often necessary to use a specific ABAP user, regardless of which user is currently logged onto the J2EE engine. In this case, you specify a defined or named user. In this case, the maximum connections as well as the pool size as determined above must be multiplied by the number of concurrent users expected. Here are some typical values for this scenario (again, these numbers must be adapted to your landscape!):

- Maximal Pool Size 50
- Maximum Connections 100
- Connection Timeout 10
- Maximum Wait Time 10

Metadata Destinations

Because this configuration is not used directly by an application, but rather by the AdaptiveRFC framework, this configuration is typically straight-forward and should be set as follows:

- Maximal Pool Size 1
- Maximum Connections 5
- Connection Timeout 60
- Maximum Wait Time 10

The Author

Jochen Guertler studied computer science at the university in Karlsruhe and joined SAP in 1998. After two years with SAP Markets he joined 2001 the Web Dynpro Java Runtime team, where he is responsible for the Web Dynpro server platform, the Web Dynpro Content Administrator, and especially for the integration with the SAP Enterprise Portal".