

Multi-byte Character Support

Oracle Banking Payments

Release 14.7.4.0.0

[June 2024]



---

# Contents

- 1. INTRODUCTION .....3
- 1.1 BACKGROUND .....3
- 1.2 APPROACH.....3

---

# 1. Introduction

Oracle Banking Payments Solution supports multi-byte characters by means of setting session-level character length semantics during Payments installation using the NLS\_LENGTH\_SEMANTICS parameter of Oracle Database. Most of the application code is written without explicit length semantics (BYTE or CHAR) and, hence, adapts to the session-level semantics during compilation at the time of Payments installation.

## 1.1 Background

NLS\_LENGTH\_SEMANTICS parameter allows you to specify the length of a column datatype/PLSQL variable in terms of characters (CHAR) instead of the default BYTE. Typically, you would need this if you were to deal with (in PL/SQL code) and store data that contains multi-byte characters such as Japanese or Mandarin characters.

Oracle strongly recommends that you do not set NLS\_LENGTH\_SEMANTICS parameter on instance level (i.e. in the server parameter file) but only use it to alter session-level semantics as necessary. From MOS note 144808.1:

*“Oracle advises to use explicit CHAR semantics in the SQL or PL/SQL syntax  
OR*

*To make sure your application does an "ALTER SESSION SET  
NLS\_LENGTH\_SEMANTICS=CHAR;" when connecting if CHAR semantics is required but the  
semantic is not defined explicit in SQL.”*

Based on this note and other available documentation for the parameter, we recommend an approach in the next section which should be followed in sites where multi-byte character support is required.



SR with oracle mentioned below is feasible. (SR 3-16364463401)

## 1.2 Approach

In situations where PL/SQL code would interact with multi-byte character data and the same kind of data would be stored in Payments tables, follow below approach:

Please note the approach is suggested for 12c Database version only.

1. Complete the database installation and set up as per the Oracle Banking Payments Database Best Practices for your Payments release.
2. Immediately after creating the Payments application user (schema), create a logon trigger as the SYS user to alter each new session so that the session-level length semantics are maintained as CHAR for the user at all times. For example, if the schema name was PAYMENTS and the trigger was named "PAYMENTS\_LOGON", the trigger would look like:

```
CREATE OR REPLACE TRIGGER SYS.PAYMENTS_LOGON
AFTER LOGON ON PAYMENTS.SCHEMA
BEGIN
    EXECUTE IMMEDIATE 'alter session set nls_length_semantics=CHAR';
END;/
```

3. Proceed with the rest of the Payments installation as per the installation documents.

In situations where an operation results in Payments objects being invalidated, make sure the objects are re-compiled either:

4. by the PAYMENTS schema user using one of the following options:

- a. Manually using ALTER... COMPILER statements
- b. SYS.UTL\_RECOMP.RECOMP\_PARALLEL procedure
- c. SYS.UTL\_RECOMP.RECOMP\_SERIAL procedure

Or,

5. by a user with appropriate privileges on SYS.UTL\_RECOMP package (such as the SYS user) **using one of the following options only:**

- d. SYS.UTL\_RECOMP.RECOMP\_PARALLEL procedure
- e. SYS.UTL\_RECOMP.RECOMP\_SERIAL procedure

Oracle Banking Payments strongly advises against manually compiling PAYMENTS schema objects (using ALTER...COMPILE) as a non-Payments application user when length semantics of CHAR are being enforced. Such an attempt may change the NLS\_LENGTH\_SEMANTICS parameter setting of the PL/SQL objects to BYTE and this may result in runtime exceptions within the application, such as:

“ORA-06502- PL/SQL: numeric or value error: character string buffer too small”.



If the PAYMENTS schema is ever re-compiled using SYS.DBMS\_UTILITY.COMPILE\_SCHEMA in an environment that is using CHAR semantics, then it must be ensured that for the user executing the procedure, the session-level NLS\_LENGTH\_SEMANTICS parameter is set to CHAR. This is because while SYS.UTL\_RECOMP.x procedures re-use existing PL/SQL object settings, SYS.DBMS\_UTILITY.COMPILE\_SCHEMA compiles the objects with the NLS settings of the invoking user's session and may alter the PL/SQL object settings of the objects (including length semantics).



Oracle Banking Payments  
Multi-byte Character Support  
[June] [2024]  
Version 14.7.4.0.0

Oracle Financial Services Software Limited  
Oracle Park  
Off Western Express Highway  
Goregaon (East)  
Mumbai, Maharashtra 400 063  
India

Worldwide Inquiries:  
Phone: +91 22 6718 3000  
Fax: +91 22 6718 3001  
[www.oracle.com/financialservices/](http://www.oracle.com/financialservices/)

Copyright © [2017], [2024], Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

**U.S. GOVERNMENT END USERS:** Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.