Service Architecture Leveraging Tuxedo (SALT)

Release Notes

12*c* Release 2 (12.2.2)

April 2016

ORACLE

Oracle Service Architecture Leveraging Tuxedo (SALT) Release Notes, 12*c* Release 2 (12.2.2)

Copyright © 2006, 2016 Oracle and/or its affiliates. All rights reserved.

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.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

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 fail-safe, 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.

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

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.

Contents

SALT 12*c* Release 2 (12.2.2) Release Notes

About This SALT Release 2

SALT 12c Release 2 (12.2.2) 2

Upgrade Considerations 3

SALT Installation Prerequisites 3

SALT Platform Support 4

Interoperability Considerations 4

Known Issues 4

Web Services 4

Interoperability 6

See Also 6

SALT 12*c* Release 2 (12.2.2) Release Notes

Date: April 2016

Table 1 Revision History

Revision Date	Summary of Change
April 2016	GA Release

The following topics are discussed in this section:

- About This SALT Release
- Upgrade Considerations
- SALT Installation Prerequisites
- SALT Platform Support
- Interoperability Considerations
- Known Issues

About This SALT Release

SALT 12c Release 2 (12.2.2)

What's New and Improved

This release introduces the following features:

Custom SOAP Headers

This feature enables use application specific headers in SOAP messages when accessing Tuxedo services using SOAP/http. This feature leverages the existing `tpsetcallinfo()` and `tpgetcallinfo()` APIs to place/retrieve data into/from the SOAP header for inbound/outbound directions.

- Inbound (Oracle Tuxedo services exposed as web services, or native web services).
- Outbound (invoking web services as Oracle Tuxedo services, or external web services).

For more information, see `tpsetcallinfo()`, `tpgetcallinfo()`, *Enabling the SALT Configuration Tool, Custom SOAP Headers* in the SALT Configuration Guide.

RECORD Buffer Type Support

`RECORD` buffer type is now supported in SALT. `RECORD` buffer type is a flexible way to correctly and completely represent COBOL copybook records.

`RECORD` buffers are used in SALT to expose mainframe transactions and ART Transactions as Web services and to access Web services from these transactions.

For more information, see *Using Oracle Tuxedo Service Metadata Repository for SALT*, and *Tuxedo-to-XML Data Type Mapping for Oracle Tuxedo Services*.

Mainframe Transaction Publisher for SALT

This release of SALT makes it easier for transactions running in IBM mainframe's CICS/IMS environments to be exported as Web services (SOAP/http or RESTful Web services). In addition, such mainframe transactions can also access external Web services using SOAP/http or RESTful API. Such integration is completely configuration driven and no code development is needed.

SALT includes tools, such as `wscobolcvt`, which allow to import COBOL copybook and create metadata repository artifacts, WSDL and other required configuration to expose mainframe transactions as Web services. Similarly, tools allow import of a WSDL for external Web service and create required artifacts for mainframe transactions to access external Web services.

For more information see, `wscobolcvt`, *SALT Mainframe Transaction Publisher*, and *XML-to-Tuxedo Data Type Mapping for External Web Services*.

New Tuxedo Services Console

- This release of SALT replaces existing SALT configuration tool with a new Tuxedo Services Console to export, import Tuxedo services. This new console provides following major functionality:
- Services Management: add/edit/delete Tuxedo service definitions in the Tuxedo metadata repository.
- Configure Web services (SOAP and REST): enables Oracle Tuxedo services to be accessed as SOAP or REST services and enables Tuxedo applications to access external SOAP/REST Web services.
- Integrate with mainframe transactions: enables mainframe transactions to be accessed as Web services (SOAP or REST), or enables mainframe transactions to access external Web services.

For more information, see `TMADMSVR` and SALT Mainframe Transaction Publisher.

Single-Sign-on with Oracle Access Manager

This release of SALT integrates with Oracle Access Manager (OAM) seamlessly, with automatic detection of the OAM authentication server configuration and handling of OAM tokens. This achieves single sign-on at the HTTP level for support of secure SOAP and REST access.

For more information see *Configuring Oracle Tuxedo Web Services*, Configuring Security Features.

Upgrade Considerations

If a previous SALT release is installed, it must be uninstalled before installing SALT 12c Release 2 (12.2.2). SALT 12c Release 2 (12.2.2) cannot co-exist with a previous SALT installation on the same Oracle Tuxedo installation.

For information, see *Migrating from SALT 1.1 Application* in the SALT Configuration Guide.

SALT Installation Prerequisites

Before installing SALT 12*c* Release 2 (12.2.2), you must ensure that Oracle Tuxedo 12c Release 2 (12.2.2) is installed.

SALT 12c Release 2 (12.2.2) is included in Oracle Tuxedo 12c Release 2 (12.2.2) installer. For more information, see the Oracle Tuxedo 12c Release 2 (12.2.2) Installation Guide.

SALT Platform Support

SALT 12*c* Release 2 (12.2.2) supports the same platforms as Oracle Tuxedo 12c Release 2 (12.2.2).

Interoperability Considerations

SALT 12*c* Release 2 (12.2.2) is compatible with (and supports), most industry-standard Web service development toolkits. For more information, see *Interoperability Considerations* in the SALT Interoperability Guide.

Known Issues

The following sections describe known problems in SALT 12*c* Release 2 (12.2.2). Entries include a description of the problem, and a workaround or solution where appropriate.

Each problem is listed by the BugDB number.

- Web Services
- Interoperability

Web Services

BugDB Number	Description and Workaround or Solution	Found In
22978257	**Problem: Invoking an external Web Service using RECORD/COBOL results in accuracy issue with decimal fields.** When the Web Service interface contains elements mapped to COBOL decimal types such as pic s9(18) the conversion will remove the decimal portion or right part of the decimal point, for example 123.34 will convert to 123.	SALT 12.2.2
	Platforms: All	
	Workaround: None, this will be corrected in RP01l	

BugDB Number	Description and Workaround or Solution	Found In
22978268	**Problem: Using the Jolt Test page in the Tuxedo Services Console, fields with more than 1 occurrence are partially supported.** Fields with more a 'count' attribute of 2 or more cannot currently be tested fully as only the first occurrence is used for capturing input, and also only the first occurrence is displayed in the returned result.	SALT 12.2.2
	Platforms: All	
	Workaround: Modify the generated code to send the expected occurrences, and process the responses, by using the item number parameter in the setters and getters. This will be corrected in RP01.	
22995695 22994222 22988290 22986213	**Problem: Issue with mapping of element or attribute names in COBOL.** When setting up SALT to perform external Web Services invocations through Mainframe Transaction Integrator, element or attribute names in Web Services may be incompatible with COBOL naming conventions. For example elements can be COBOL keywords, or contain characters invalid in COBOL (starting with underscore). This prevents generated COBOL copybooks from compiling.	SALT 12.2.2
	Platforms: All	
	Workaround: Modify Web Service interface to remove the above mentioned restrictions.	

Interoperability

BugDB Number	Description and Workaround or Solution	Found In
21414741	**Problem: Interoperability issue with WLS 12c JAX-WS external Web Services built without specifying `elementFormDefault`.** This affects the invocation of external Web Services from Oracle Tuxedo on WLS 12c onward. JAX-WS uses "`elementFormDefault=unqualified`" as default, while Oracle SALT uses "`qualified`". This causes parsing exceptions on the server (WLS) side, for example: `"<Client Request doesn't pass Service's Schema Validation com.sun.istack.SAXParseException2; org.xml.sax.SAXParseException; cvc-complex-type.2.4.a: Invalid content was found starting with element 'ns0:arg0'. One of '{arg0}' is expected ..."`	SALT 12.1.2
	Platforms:	
	Workaround: Generate the JAX-WS service with setting `elementFormDefault`, preferably "`qualified`" for the best interoperability with Oracle SALT.	

See Also

- SALT 12c Release 2 (12.2.2) Product Overview
- SALT 12c Release 2 (12.2.2) Administration Guide
- SALT 12c Release 2 (12.2.2) Configuration Guide
- SALT 12c Release 2 (12.2.2) Programming Guide
- SALT 12c Release 2 (12.2.2) Reference Guide