

Oracle® MICROS Symphony

Extensibility Compatibility Guide



Release 19.2.x or Later
G10969-01
July 2024

ORACLE®

Copyright © 2010, 2024, 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 software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then 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.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface	iv
1 Introduction	1-1
Scope of Document	1-1
2 General Implications	2-1
Compatibility	2-1
Maintaining Compatibility with Recompile	2-1
3 Urgency and Timing	3-1
4 Configuring for Multiple Versions	4-1
Extension Applications (EMC)	4-1
5 Implementation Specific Implications	5-1
DataStoreClient	5-1
GetBusinessDateTime	5-1

Preface

An Oracle MICROS Symphony POS architecture change beginning with version 19.2, affects applications built against the extensibility API DLLs. Extensibility developers must take action to maintain compatibility.

Purpose

This document provides details of what extensibility app developers need to do to maintain compatibility with Symphony 19.2.x or later.

Audience

This document is intended for extensibility developers for Symphony.

Customer Support

To contact Oracle Customer Support, access My Oracle Support at the following URL:

<https://support.oracle.com>

When contacting Customer Support, please provide the following:

- Product version and program/module name
- Functional and technical description of the problem (include business impact)
- Detailed step-by-step instructions to re-create
- Exact error message received and any associated log files
- Screenshots of each step you take

Documentation

Oracle MICROS product documentation is available on the Oracle Help Center at <http://docs.oracle.com/en/industries/food-beverage/>

Revision History

Date	Description of Change
July 2024	Initial publication.

1

Introduction

A Symphony POS architecture change beginning with version 19.2 affects applications built against the extensibility API DLLs.

Scope of Document

This document serves as a guidance for extensibility developers on actions required to maintain compatibility with Symphony 19.2 or later, including:

- General implications of the Symphony architecture change.
- Actions required to maintain compatibility.
- How to configure specific extensibility apps for multiple Symphony versions.
- Implementation specific implications

2

General Implications

Simphony POS architecture changed beginning with version 19.2, adding functionality for Simphony Transaction Services (STS) Gen2, for both on premise and Cloud environments. This change affected Extensibility API forwards and backwards compatibility.

Any Simphony version 19.2 or later Extensibility solutions need to be built against Simphony Core DLL's:

- *Ops.dll*
- *PosCommonClasses.dll*
- *PosCore.dll*

Compatibility

Extensibility solutions are neither forwards nor backwards compatible with 19.2. Specifically:

1. A Simphony 19.2 Extensibility solution is **not** backwards compatible. It will not run on Simphony Client 19.1, or 18.2 and earlier.
2. Any existing Simphony Extensibility solution (built against Simphony 19.1, 18.2 or earlier) is **not** forwards compatible. It will not run on a Simphony 19.2 client.

Maintaining Compatibility with Recompile

A recompile of existing extensibility solutions is necessary against the 19.2 DLLs to maintain compatibility with Simphony 19.2. In many cases, this is sufficient.

If *DataStoreClient* or *GetBusinessDateTime* are referenced in the code, then code changes are required. See the section later in this document detailing implementation specific changes when **more** than just a recompile is necessary.

3

Urgency and Timing

Simphony Enterprise Cloud can be updated to Simphony version 19.2 or later without an updated client extensibility solution in place. However, updated client extensibility solutions **must** be in place before Simphony **clients** are updated to 19.2 or later. A Simphony client is able to stay on a previous version for days or weeks after Simphony Enterprise Cloud is updated to 19.2. This provides some flexibility for the development and/or deployment of updated extensibility solutions.

4 Configuring for Multiple Versions

Symphony allows simultaneous support for new and old extension applications, as described here. EMC offers the functionality needed to maintain version specific Extension Application Content (aka Symphony Extensibility DLLs).

Extension Applications (EMC)

EMC provides configuration for extension DLLs by version. It is through this configuration screen where it will be important to specify both the new and old versions, one for Symphony versions prior to 19.2, and one for 19.2 and later. Pictured is a sample of such a configuration.

	#	Zoneable Key	Description	Option Bits	Disk File Name	Target	Version
	1	NewtonSoft.Json	. NewtonSoft.Json	0000000000000000 ...	Newtonsoft.Json.dll	FF1FFFFF ...	(any) - (any) ...
	2	VSDC	. VSDC DLL	0000000000000000 ...	VSDC.dll	FF1FFFFF ...	(any) - 19.1 ...
	3	VSDC_19_2	. VSDC DLL	0000000000000000 ...	VSDC.dll	FF1FFFFF ...	19.2 - (any) ...
	4	VSDCExtension	. VSDCExtension	0000000000000000 ...	VSDCExtension.dll	FF1FFFFF ...	(any) - 19.1 ...
	5	VSDCExtension_19_2	. VSDCExtension	0000000000000000 ...	VSDCExtension.dll	FF1FFFFF ...	19.2 - (any) ...

5

Implementation Specific Implications

Simphony Extensibility internal *DataStoreClient* (*OpsContext.DataStore*) is deprecated and should no longer be used with Simphony 19.2.

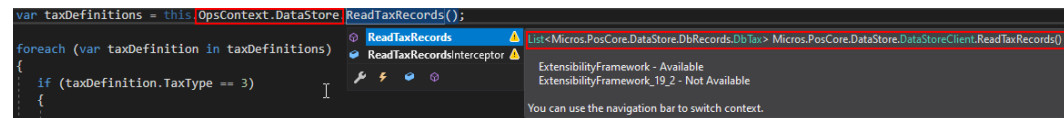
Source needs to be updated to make use of the Extensibility *DataStoreClient* only (namespace *Micros.PosCore.Extensibility.DataStore*). Sample code is provided below.

Functionality around *OpsContext.GetBusinessDateTime()* has changed and needs to be updated. Sample code is provided below.

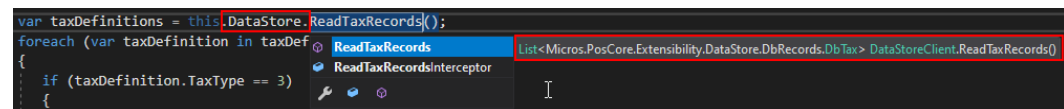
DataStoreClient

This sample below is based on a single Visual Studio solution that includes two Simphony Extensibility projects, one built against an earlier Simphony version, the second built against Simphony 19.2.

Deprecated use of Simphony Extensibility internal *DataStoreClient*:



Correct use of Simphony Extensibility *DataStoreClient*:



GetBusinessDateTime

Change in the Signature of the *GetBusinessDateTime()* method:

GetBusinessDateTime() method in earlier Simphony versions:

```
public Micros.EGatewayHandlerCommon.BusinessDateTimes GetBusinessDateTime(FinancialCalendarData calendarData);
public decimal GetCheckSubtotalAsDecimal()
```

GetBusinessDateTime() method in Simphony 19.2:

```
public BusinessDateTimes GetBusinessDateTime(PosCore.Extensibility.DataStore.DbRecords.FinancialCalendarData calendarData);
[Obsolete("GetBusinessDateTime is using internal types that you may not be able to access, please use the Micros.PosCore.Extensibility.DataStoreClient.ReadTaxRecords() method")]
public Micros.EGatewayHandlerCommon.BusinessDateTimes GetBusinessDateTime(PosCore.DataStore.DbRecords.FinancialCalendarData calendarData);
public List<PosCore.Extensibility.DataStore.DbRecords.FinancialCalendarData> GetBusinessDateTimesSelection(string reportPrivilegeID, bool includeAllRecords)
```