Oracle Hospitality Integration Platform

Reservations Credit Card Payment Methods

Release 24.3
G18109-01
November 2024

Oracle Hospitality Integration Platform Reservations Credit Card Payment Methods, Release 24.3

G18109-01

Copyright © 2024, 2024, Oracle and/or its affiliates.

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, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that 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 embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. 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®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside 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, Epyc, and the AMD 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

1 Business Context

2 Prerequisites

3 Recommended Tools

4 Environment

5 Configurations

6 Workflow of Use Case

7 Limitations and Constraints

8 Anti-patterns

9 References

10 Key Terminology

11 FAQ

Preface

Oracle Hospitality Integration Cloud Service and OPERA Cloud Foundation users are authorized to access the following modules and features:

- Oracle Hospitality Integration Platform including Oracle Hospitality Developer Portal and Hospitality REST APIs.

The Oracle Hospitality Developer Portal enables users to discover Oracle Hospitality APIs, subscribe to Oracle Hospitality APIs, and to get the necessary information to consume them.

The Oracle Hospitality Developer Portal's APIs page provides information about published Oracle Hospitality APIs. Here you can find and evaluate Oracle Hospitality APIs to use with your applications.

After you discover the Oracle Hospitality APIs that you want to use, register an application and then register those APIs to your application.

To view more details about the Oracle Hospitality Developer Portal, on the user menu drop-down, click the drop-down, and select **About Developer Portal**.

Purpose

This guide describes best practices to determine the credit card payment method data needed to successfully create and update hotel reservations based on an integrated Oracle Payment Interface (OPI) at the property level.

Audience

This guide is intended for customers and partners who develop applications with the Oracle Hospitality Integration Platform.

Documentation

Oracle Hospitality product documentation is available on the Oracle Help Center at https://docs.oracle.com/en/industries/hospitality/.

Revision History

Date	Description of Change
November 2024	Initial publication

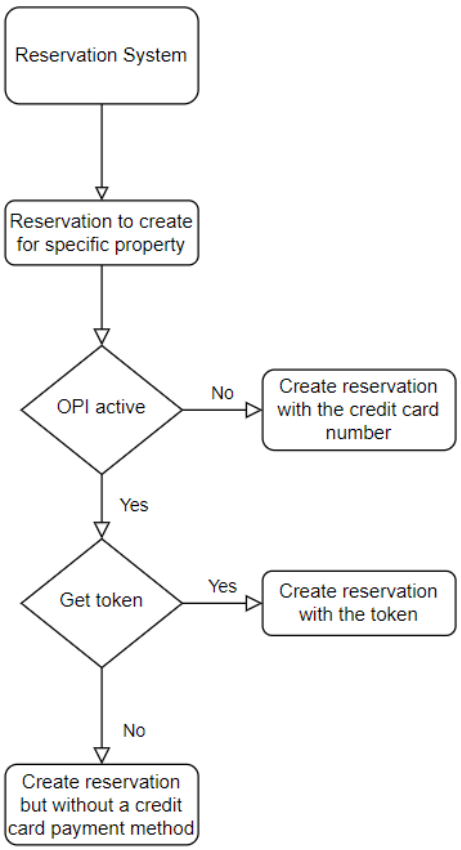
1 Business Context

This guide describes best practices to determine the credit card payment method data needed to successfully create and update hotel reservations based on an integrated Oracle Payment Interface (OPI) at the property level.

Many hotels with OPERA Cloud are using the Oracle Payment Interface (OPI) to integrate with Payment Service Providers (PSP) to obtain tokens for the credit card number on reservations, and then process authorizations and payments to collect revenue. Therefore, it is important to know how best to handle credit card data as the payment method to successfully create/ update reservations that can utilize this process.

> **Note:**
>
> API usage and capabilities are subject to change. For more information on API versioning, see the Versioning topic in the Oracle Hospitality Integration Platform User Guide.
>
> For the most comprehensive and up-to-date information, including additional query parameters and detailed descriptions, always refer to the latest API specifications in GitHub. This will ensure that you are accessing the most current and complete set of features available.

Use Case	Description	Operation IDs
Reservation Payment Methods when OPI is not in use at the property.	Reservation system is ready to create or update a reservation to the property and will have the credit card number in the request.	getHotelInterfaces getOperaSettings getPaymentMethodsLOV postReservation
Reservation Payment Methods when OPI is in use at the property.	Reservation system is ready to create or update a reservation to the property and will have the token in the request.	getHotelInterfaces getOperaSettings getPaymentMethodsLOV postOpenPaymentTokenExchange postReservation

Journey to create Reservations with Credit Card Payment Method	
	• Partner application is prepared to create a reservation for the guest. • The property is selected and guest provides their credit card number for payment method and guarantee on the reservation. • Partner has fetched knowledge for this property that OPI is not active. • Reservation is created with the guest credit card number as the payment method. OR • Partner has fetched knowledge for this property that OPI is active. • Partner fetches a token for the guest's credit card number directly from the PSP or from OPI Cloud. • Reservation is created with that token as the payment method. OR • Partner has fetched knowledge for this property that OPI is active. • Partner can't fetch a token for the guest's credit card number directly from the PSP or from OPI Cloud. • Reservation is created without a credit card number or token as the payment method.

2
Prerequisites

- Knowledge of Reservation Systems to successfully create reservations. For more information, refer to the CRS Implementation Guide.
- OPERA Cloud is striving to not have credit card numbers for use (that is, PCI data); therefore, it is expected that OPI will be active with most property's to utilize tokenization.
- Payment Service Providers (PSP) must be certified with Oracle for OPI functionality.

3
Recommended Tools

The below software tools are useful when following the steps outlined in this guide.

Tool	Description	Links
Github	A Github repository containing both Oracle Hospitality REST API specifications and accompanying Postman Collections.	Github and Postman Collections Github
Postman Collections	In the postman-collections folder in Github, you can download and use the Postman Collections associated to the use cases explained in this guide.	Oracle Hospitality APIs Overview Postman

4
Environment

The following environments and modules must be available to perform the Business cases with the Property APIs that are described in this guide.

Oracle Platform/Service	Description	Minimum Version
OPERA Cloud Services	The customer must have a subscription to OPERA Cloud Foundation	With OPI on premise - 21.5.0.0 With OPI Cloud - 23.1.0.0
Oracle Hospitality Integration Platform (OHIP)	For customers, OHIP is included in the subscription to OPERA Cloud Foundation. Partners need a subscription to Oracle Hospitality Integration Cloud Service	The latest released version. OHIP is a single version product.
Oracle Payment Interface (OPI)	The customer can have OPI on premise or OPI Cloud.	20.3 OPI (Premise) 23.1 OPI Cloud

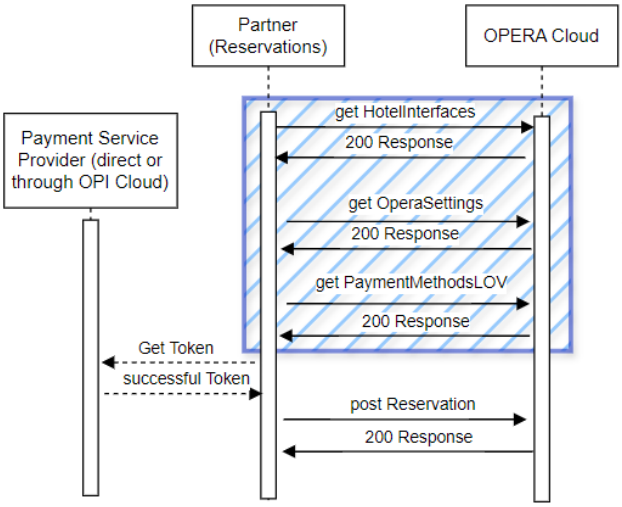
5 Configurations

1. To start calling Property APIs, you must perform the following tasks:
   - Accessing the Oracle Hospitality Integration Platform Developer Portal
   - Creating an Integration User
   - Registering an Application

6

Workflow of Use Case

Sequence Diagram



1. Determine if OPI is in use at the property
2. Determine if OPI Cloud is active at the property
3. Fetch the CardTypes and PaymentMethods for use
4. Fetch the token if needed
5. Create the reservation with the PaymentMethod

Configuration
API calls to retrieve data to determine payment use

Description of Steps

Steps	Description of Steps	Operation Samples

1	The Partner queries hotelId for the active Hotel Interfaces to determine if a property has an Oracle Payment Interface (OPI) active. This configuration does not change often, and it is recommended to cache the verification result and revalidate only if issues emerge around the reservation payment methods.
	If the response includes an interfaceType=EFT and *activeFlag* = true for the property, it indicates the following:
	• OPI is active.
	• Tokenization is active.
	• OPI communicates to the Payment Service Providers (PSP) to retrieve tokens.
	• A clear credit card number can NOT be sent in the Property APIs.
	• Tokens are needed in the Property APIs.
	If the 200 response is empty of data, this indicates the following:
	• OPI is not active.
	• Tokenization is not active.
	• A clear credit card number can be sent in the Property APIs.
		getHotelInterfaces
		GET {{HostName}}/int/config/v1/hotelInterfaceTypes?includeInactive=false&interfaceTypes=EFT&hotelIds={{HotelId}}
		200 Response with active EFT (OPI):
		
		`{`
		`"hotelInterfaces": [`
		`{`
		`"interfaceId": {`
		`"id": "{{Id}}",`
		`"type": "InterfaceId"`
		`},`
		`"hotelId": "{{HotelId}}",`
		`"interfaceName": "OPI EFT",`
		`"interfaceType": "Eft",`
		`"logo": "{{Logo}}",`
		`"machineName": "{{Machine}}",`
		`"status": "Running",`
		`"activeFlag": true`
		`}`
		`],`
		`"links": []`
		`}`
		
		200 Response without active EFT (OPI):
		
		`{`
		`"hotelInterfaces": [],`
		`"links": []`
		`}`

Steps	Description of Steps	Operation Samples
2	If OPI is active and in use, the Partner can query if OPI Cloud is active. With 200 response where value=Y, it is known that OPI Cloud is available. This is optional, but it will help determine if the Partner can use the postOpenPaymentTokenExchange in Step 4 to get a token for the credit card number. It is not needed if OPI is not active at the property (determined from Step 1) or the Partner already has a token from the relevant PSP (when OPI is active at the property determined from Step 1).	getOperaSettings GET {{HostName}}/ent/config/v1/settings?hotelId={{HotelId}}&parameterNameWildCard=OPI 200 Response: { "activeFunctions": 143, "groups": [{ "appSettings": [{ "shortDescription": "Activate OPI Cloud Credit Card Processing", "description": "Activate OPI Cloud Credit Card Delivery and Disable IFC Controller for Credit Card Processsing.", "wildCardMatch": true, "name": "OPI_CREDIT_CARD_PROCESSING", "displayName": "OPI Cloud Credit Card Processing", "type": "Parameter", "hotelId": "{{HotelId}}", "sequence": 1, "editAllowed": true, "value": "Y", "scope": "P", "valueType": "Yn", "levelType": "Hotel", "levelCode": "{{HotelId}}", "subscriptionCountEligible": true }, { "shortDescription": "Activate OPI Cloud Credit Card Processing", "description": "Activate OPI Cloud Credit Card Delivery and Disable IFC Controller for Credit Card Processsing", "wildCardMatch": true, "name": "OPI_TOKEN_PROXY_SERVICE", "displayName": "OPI

Steps	Description of Steps	Operation Samples
		Cloud Credit Card Processing", "type": "Parameter", "hotelId": "{{HotelId}}", "sequence": 1, "editAllowed": true, "scope": "P", "valueType": "Yn", "levelType": "Hotel", "levelCode": "{{HotelId}}", "subscriptionCountEligible": true }], "groupName": "Credit Card", "groupDisplayName": "Credit Card" }], "links": [] }

Steps	Description of Steps	Operation Samples

3	Partner queries hotelId for available cardTypes and paymentMethods. A successful response contains the Card Types and Payment Methods in use at the property. The "parameterValue" data of the "parameterName"=CardType along with the "parameterValue":"Y" for the "parameterName": "ResvPaymentYN" indicates the cardType and the "code" indicates the paymentMethod that can be used in a post reservation (Step 5). This configuration does not change often, and it is recommended to cache the result and revalidate only if issues emerge around the reservation payment methods.	getPaymentMethodsLOV
		GET {{HostName}}/lov/v1/listOfValues/hotels/{{HotelId}}/paymentMethods?includeInactiveFlag=false
		200 Response:
		
		{
		"listOfValues": {
		"items": [
		{
		"flexfields": [
		{
		"parameterName": "CardType",
		"parameterValue": "CP"
		},
		{
		"parameterName": "ChipPinYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "ResvPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "BillingPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "DepositPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "ARPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "CompPaymentYN",

Steps	Description of Steps	Operation Samples
		"parameterValue": "N" }, { "parameterName": "CheckPaymentYN", "parameterValue": "N" }, { "parameterName": "CashPaymentYN", "parameterValue": "N" }, { "parameterName": "DirectBillYN", "parameterValue": "N" }, { "parameterName": "TrxCode", "parameterValue": "CP" }, { "parameterName": "AuthDuringStayYN", "parameterValue": "Y" }, { "parameterName": "AuthAtCheckinYN", "parameterValue": "Y" }, { "parameterName": "PostAllowedYN", "parameterValue": "N" }, { "parameterName": "RoundFactorYn", "parameterValue": "N" }, {

Steps	Description of Steps	Operation Samples
		"parameterName": "GuestLedgerDepositsYn", "parameterValue": "N" }, { "parameterName": "DepositFoliosYn", "parameterValue": "N" }, { "parameterName": "DepositReceiptsYn", "parameterValue": "N" }, { "parameterName": "AuthStlmtAtCheckOutYn", "parameterValue": "N" }, { "parameterName": "FiscalPaymentPayloadEligibilityYn", "parameterValue": "N" }], "code": "CP", "name": "Chip and Pin", "description": "Chip and Pin", "active": true }, { "flexfields": [{ "parameterName": "CardType" }, { "parameterName": "ChipPinYN", "parameterValue": "N" }, { "parameterName": "ResvPaymentYN",

Steps	Description of Steps	Operation Samples

		<pre>"parameterValue": "Y"
		},
		{
		"parameterName": "BillingPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "DepositPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "ARPaymentYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "CompPaymentYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "CheckPaymentYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "CashPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "DirectBillYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "TrxCode",
		"parameterValue": "9000"
		},</pre>

Steps	Description of Steps	Operation Samples

		<pre>{
		"parameterName": "AuthDuringStayYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "AuthAtCheckinYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "PostAllowedYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "RoundFactorYn",
		"parameterValue": "Y"
		},
		{
		"parameterName": "GuestLedgerDepositsYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "DepositFoliosYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "DepositReceiptsYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "AuthStlmtAtCheckOutYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "FiscalPaymentPayloadEligibilityYn",</pre>

Steps	Description of Steps	Operation Samples

		<pre>"parameterValue": "N"
		}
		],
		"code": "CA",
		"name": "Cash",
		"description": "Cash",
		"active": true
		},
		{
		"flexfields": [
		{
		"parameterName":
		"CardType",
		
		"parameterValue": "AX"
		},
		{
		"parameterName":
		"ChipPinYN",
		
		"parameterValue": "Y"
		},
		{
		"parameterName":
		"ResvPaymentYN",
		
		"parameterValue": "Y"
		},
		{
		"parameterName":
		"BillingPaymentYN",
		
		"parameterValue": "Y"
		},
		{
		"parameterName":
		"DepositPaymentYN",
		
		"parameterValue": "Y"
		},
		{
		"parameterName":
		"ARPaymentYN",
		
		"parameterValue": "Y"
		},
		{
		"parameterName":
		"CompPaymentYN",</pre>

Steps	Description of Steps	Operation Samples
		"parameterValue": "N" }, { "parameterName": "CheckPaymentYN", "parameterValue": "N" }, { "parameterName": "CashPaymentYN", "parameterValue": "N" }, { "parameterName": "DirectBillYN", "parameterValue": "N" }, { "parameterName": "TrxCode", "parameterValue": "90025" }, { "parameterName": "AuthDuringStayYN", "parameterValue": "Y" }, { "parameterName": "AuthAtCheckinYN", "parameterValue": "Y" }, { "parameterName": "PostAllowedYN", "parameterValue": "Y" }, { "parameterName": "RoundFactorYn", "parameterValue": "N" }, {

Steps	Description of Steps	Operation Samples
		"parameterName": "GuestLedgerDepositsYn", "parameterValue": "N" }, { "parameterName": "DepositFoliosYn", "parameterValue": "N" }, { "parameterName": "DepositReceiptsYn", "parameterValue": "N" }, { "parameterName": "AuthStlmtAtCheckOutYn", "parameterValue": "N" }, { "parameterName": "FiscalPaymentPayloadEligibilityYn", "parameterValue": "N" }], "code": "AX", "name": "AMEX", "description": "American Express", "active": true }, { "flexfields": [{ "parameterName": "CardType" }, { "parameterName": "ChipPinYN", "parameterValue": "N" }, { "parameterName": "ResvPaymentYN",

Steps	Description of Steps	Operation Samples
		"parameterValue": "Y" }, { "parameterName": "BillingPaymentYN", "parameterValue": "Y" }, { "parameterName": "DepositPaymentYN", "parameterValue": "N" }, { "parameterName": "ARPaymentYN", "parameterValue": "Y" }, { "parameterName": "CompPaymentYN", "parameterValue": "N" }, { "parameterName": "CheckPaymentYN", "parameterValue": "N" }, { "parameterName": "CashPaymentYN", "parameterValue": "N" }, { "parameterName": "DirectBillYN", "parameterValue": "Y" }, { "parameterName": "TrxCode", "parameterValue": "96674" },

Steps	Description of Steps	Operation Samples

		<pre>{
		"parameterName": "AuthDuringStayYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "AuthAtCheckinYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "PostAllowedYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "RoundFactorYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "GuestLedgerDepositsYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "DepositFoliosYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "DepositReceiptsYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "AuthStlmtAtCheckOutYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "FiscalPaymentPayloadEligibilityYn",</pre>

Steps	Description of Steps	Operation Samples

		<pre>"parameterValue": "N"
		}
		],
		"code": "DB",
		"name": "Direct Bill",
		"description": "Direct Bill",
		"active": true
		},
		{
		"flexfields": [
		{
		"parameterName": "CardType",
		"parameterValue": "DS"
		},
		{
		"parameterName": "ChipPinYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "ResvPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "BillingPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "DepositPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "ARPaymentYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "CompPaymentYN",</pre>

Steps	Description of Steps	Operation Samples
		"parameterValue": "N" }, { "parameterName": "CheckPaymentYN", "parameterValue": "N" }, { "parameterName": "CashPaymentYN", "parameterValue": "N" }, { "parameterName": "DirectBillYN", "parameterValue": "N" }, { "parameterName": "TrxCode", "parameterValue": "90004" }, { "parameterName": "AuthDuringStayYN", "parameterValue": "Y" }, { "parameterName": "AuthAtCheckinYN", "parameterValue": "Y" }, { "parameterName": "PostAllowedYN", "parameterValue": "N" }, { "parameterName": "RoundFactorYn", "parameterValue": "N" },

Steps	Description of Steps	Operation Samples

		{
		"parameterName": "GuestLedgerDepositsYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "DepositFoliosYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "DepositReceiptsYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "AuthStlmtAtCheckOutYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "FiscalPaymentPayloadEligibilityYn",
		"parameterValue": "N"
		}
		],
		"code": "DS",
		"name": "Discover",
		"description": "Discover",
		"active": true
		},
		{
		"flexfields": [
		{
		"parameterName": "CardType",
		"parameterValue": "DC"
		},
		{
		"parameterName": "ChipPinYN",
		"parameterValue": "N"
		},

Steps	Description of Steps	Operation Samples

		<pre>{
		"parameterName": "ResvPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "BillingPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "DepositPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "ARPaymentYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "CompPaymentYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "CheckPaymentYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "CashPaymentYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "DirectBillYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "TrxCode",</pre>

Steps	Description of Steps	Operation Samples
		"parameterValue": "90005" }, { "parameterName": "AuthDuringStayYN", "parameterValue": "Y" }, { "parameterName": "AuthAtCheckinYN", "parameterValue": "Y" }, { "parameterName": "PostAllowedYN", "parameterValue": "N" }, { "parameterName": "RoundFactorYn", "parameterValue": "N" }, { "parameterName": "GuestLedgerDepositsYn", "parameterValue": "N" }, { "parameterName": "DepositFoliosYn", "parameterValue": "N" }, { "parameterName": "DepositReceiptsYn", "parameterValue": "N" }, { "parameterName": "AuthStlmtAtCheckOutYn", "parameterValue": "N" },

Steps	Description of Steps	Operation Samples

		{
		"parameterName": "FiscalPaymentPayloadEligibilityYn",
		"parameterValue": "N"
		}
		],
		"code": "DC",
		"name": "Diner Club",
		"description": "Diner Club",
		"active": true
		},
		{
		"flexfields": [
		{
		"parameterName": "CardType",
		"parameterValue": "VA"
		},
		{
		"parameterName": "ChipPinYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "ResvPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "BillingPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "DepositPaymentYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "ARPaymentYN",
		"parameterValue": "N"
		},

Steps	Description of Steps	Operation Samples

		<pre>{
		"parameterName": "CompPaymentYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "CheckPaymentYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "CashPaymentYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "DirectBillYN",
		"parameterValue": "N"
		},
		{
		"parameterName": "TrxCode",
		"parameterValue": "90002"
		},
		{
		"parameterName": "AuthDuringStayYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "AuthAtCheckinYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "PostAllowedYN",
		"parameterValue": "Y"
		},
		{
		"parameterName": "RoundFactorYn",</pre>

Steps	Description of Steps	Operation Samples

		"parameterValue": "N"
		},
		{
		"parameterName": "GuestLedgerDepositsYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "DepositFoliosYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "DepositReceiptsYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "AuthStlmtAtCheckOutYn",
		"parameterValue": "N"
		},
		{
		"parameterName": "FiscalPaymentPayloadEligibilityYn",
		"parameterValue": "N"
		}
		],
		"code": "VA",
		"name": "VISA",
		"description": "VISA",
		"active": true
		},
		{
		"flexfields": [
		{
		"parameterName": "CardType",
		"parameterValue": "MC"
		},
		{
		"parameterName": "ChipPinYN",

Steps	Description of Steps	Operation Samples
		"parameterValue": "Y" }, { "parameterName": "ResvPaymentYN", "parameterValue": "Y" }, { "parameterName": "BillingPaymentYN", "parameterValue": "Y" }, { "parameterName": "DepositPaymentYN", "parameterValue": "Y" }, { "parameterName": "ARPaymentYN", "parameterValue": "N" }, { "parameterName": "CompPaymentYN", "parameterValue": "N" }, { "parameterName": "CheckPaymentYN", "parameterValue": "N" }, { "parameterName": "CashPaymentYN", "parameterValue": "N" }, { "parameterName": "DirectBillYN", "parameterValue": "N" }, {

Steps	Description of Steps	Operation Samples
		"parameterName": "TrxCode", "parameterValue": "90001" }, { "parameterName": "AuthDuringStayYN", "parameterValue": "Y" }, { "parameterName": "AuthAtCheckinYN", "parameterValue": "Y" }, { "parameterName": "PostAllowedYN", "parameterValue": "Y" }, { "parameterName": "RoundFactorYn", "parameterValue": "N" }, { "parameterName": "GuestLedgerDepositsYn", "parameterValue": "N" }, { "parameterName": "DepositFoliosYn", "parameterValue": "N" }, { "parameterName": "DepositReceiptsYn", "parameterValue": "N" }, { "parameterName": "AuthStlmtAtCheckOutYn",

Steps	Description of Steps	Operation Samples

		<pre>"parameterValue": "N"
		                    },
		                    {
		                        "parameterName":
		"FiscalPaymentPayloadEligibilityYn",
		
		"parameterValue": "N"
		                    }
		                ],
		                "code": "MC",
		                "name": "Master Card",
		                "description": "Master
		Card",
		                "active": true
		            }
		        ],
		        "lovName": "PaymentMethods",
		        "itemCount": 8
		    }
		}</pre>

Steps	Description of Steps	Operation Samples

4	If OPI is not active at the property, skip to Step 5.
	If OPI is active at the property, Partners and customers can work with the relevant Payment Service Provider (PSP) for the property to get the token for the credit card number **directly** (not via OPI) from the PSP. Once you have the token from the PSP, use that token in the post Reservation reservationPaymentMethods section (step 5).
	If Step 2 determined OPI Cloud is active, there is the OPI API available through OHIP, post OpenPaymentTokenExchange, allowing you to request the token in exchange of the credit card number. A successful response will provide the token, cardType, last 4 of card number, and expiryDate to then use in the post Reservation reservationPaymentMethods section (step 5).
	There is also an intermediate solution that may be available when OPI is active at the property, but OPI Cloud is not in use. You can still call the postOpenPaymentTokenExchange API if the PSP supports it and an OPI Cloud has been provisioned for the property. You must ask if this option is available for you in the Partner Slack Channel.
		postOpenPaymentTokenExchange
		POST {{HostName}}/tokenExchange/v1/hotels/{{hotelCode}}/tokens
		Request Body:
		{
		"expiryDate": "2912",
		"pan": "4444333322221111"
		}
		200 Response:
		{
		"cardType": "VA",
		"expiryDate": "2912",
		"pan": "XXXXXXXXXXXX1111",
		"token": "8444048903241827"
		}

Steps	Description of Steps	Operation Samples

5	Partner posts the completed reservation to the hotelId with the required reservationPaymentMethods defined in the post Reservation.
	When there is no active OPI at a property, clear text credit card numbers can be sent in the Post Reservation, and they will be accepted/handled. The data is secured with the protocol layer and card number is stored encrypted according to PCI compliance.
	The Post Reservation contains “reservationPaymentMethods” including the following:
	• ‘cardNumber’ — The actual Credit Card Number.
	• ‘cardOrToken’ — The add the value ‘cardNumber’ to indicate a cardNumber exists.
	In the sample Request Body provided here, the result is the following:
	• The VISA payment method is added to the reservation in OPERA Cloud. Within the UI, the "1111" displays for all (as the card’s last 4 digits).
	• Should the OPERA Cloud user have the ‘Sensitive Data Role’ assigned, they have the ability to view the complete card number in OPERA Cloud.
	• Financial transactions are carried out with the credit card number on the reservation.
	When OPI is active at the property, tokens are to be sent in the Post Reservation. If credit card numbers are defined, they are stripped from the reservation.
	The Post Reservation contains the “reservationPaymentMethods” including the following:
	• ‘cardNumber’ — Set as the token value (retrieved from the PSP directly or through OPI Cloud).
	• cardNumberMasked – 12 XXXs and the last 4 of the actual card number.
	• ‘cardOrToken’ — Set as ‘Token’ to identify the cardNumber is a token.
	In the sample Request Body provided here, the result is the following:
		postReservation
		POST {{HostName}}/rsv/v1/hotels/{{HotelId}}/reservations
		Request Body when OPI in not active and credit card number is in use:
		
		{
		"reservations": {
		"reservation": {
		"reservationGuests": {
		"profileInfo": {
		"profileIdList": {
		"id": "{{ProfileId}}",
		"type": "Profile"
		}
		}
		},
		"reservationPaymentMethods": {
		"paymentCard": {
		"cardHolderName": "Test CC Number",
		"cardType": "VA",
		"cardNumber": "4444333322221111",
		"expirationDate": "2029-12-31",
		"cardOrToken": "CardNumber"
		},
		"folioView": 1,
		"paymentMethod": "VA"
		},
		"markAsRecentlyAccessed": true,
		"hotelId": "{{HotelId}}",
		"reservationStatus": "Reserved",
		"roomStay": {
		"guarantee": {
		"onHold": false,
		"guaranteeCode": "6PM"
		},
		"roomRates": {
		"sourceCode": "CAT",
		"numberOfUnits": 1,
		"rates": {
		"rate": {
		"start":

Steps	Description of Steps	Operation Samples

	• The AMEX payment method is added to the reservation in OPERA Cloud. Within the UI, the "1007" displays for all (as the card's last 4 digits).
	• Should OPERA Cloud users have the 'Sensitive Data Role' assigned, they have the ability to view the complete Token in OPERA Cloud.
	• Financial transactions are carried out with the Token on the reservation.
		<pre>"2024-05-01",
		"end":
		"2024-10-30",
		"base": {
		"amountBeforeTax": 150,
		"currencyCode": "USD"
		}
		}
		},
		"start":
		"2024-09-11",
		"marketCode": "CORP",
		"end": "2024-09-12",
		"roomTypeCharged":
		"STDQ",
		"ratePlanCode":
		"IHUBASE1",
		"roomType": "STDQ",
		"pseudoRoom": false
		},
		"guestCounts": {
		"children": 0,
		"adults": 2
		},
		"departureDate":
		"2024-09-12",
		"arrivalDate":
		"2024-09-11"
		}
		}
		}
		}</pre>
		Request Body when OPI is active and token is in use:
		<pre>{
		"reservations": {
		"reservation": {
		"reservationGuests": {
		"profileInfo": {
		"profileIdList": {
		"id":
		"{{ProfileId}}",
		"type": "Profile"
		}
		}
		},
		"reservationPaymentMethods":

Steps	Description of Steps	Operation Samples

		{
		"paymentCard": {
		"cardHolderName": "Test Token",
		"cardType": "AX",
		"cardNumberMasked": "XXXXXXXXXXXX1007",
		"cardNumber": "853230000012007",
		"expirationDate": "2029-12-31",
		"cardOrToken": "Token"
		},
		"folioView": 1,
		"paymentMethod": "AX"
		},
		"markAsRecentlyAccessed": true,
		"hotelId": "{{HotelId}}",
		"reservationStatus": "Reserved",
		"roomStay": {
		"guarantee": {
		"onHold": false,
		"guaranteeCode": "6PM"
		},
		"roomRates": {
		"sourceCode": "CAT",
		"numberOfUnits": 1,
		"rates": {
		"rate": {
		"start": "2024-05-01",
		"end": "2024-10-30",
		"base": {
		"amountBeforeTax": 150,
		"currencyCode": "USD"
		}
		}
		},
		"start": "2024-09-11",
		"marketCode": "CORP",
		"end": "2024-09-12",
		"roomTypeCharged": "STDQ",

Steps	Description of Steps	Operation Samples

		"ratePlanCode": "IHUBASE1",
		"roomType": "STDQ",
		"pseudoRoom": false
		},
		"guestCounts": {
		"children": 0,
		"adults": 2
		},
		"departureDate": "2024-09-12",
		"arrivalDate": "2024-09-11"
		}
		}
		}
		}

7

Limitations and Constraints

- If your integration is with a property that is using OPI, you are unable to send the clear credit card number as a payment method with the Property APIs. Oracle does not provide any automatic token exchange service. If credit card numbers are received, OPERA Cloud removes/ignores them.
- Tokens must be from the same PSP as in use at the property level to ensure financial transactions can be successful at the property.
- The Property API responses only provide the last 4 digits of the credit card number.

8
Anti-patterns

Anti-Pattern	Category	Rationale	Recommendation
Populating the guest's credit card number as the payment method in the reservation when OPI is active at property.	Functional	Poor customer experience and operational impact.	The credit card data will not be populated and the hotel will not be able to collect deposits or cancel fees for the reservation. The payment method must be provided with a token for the reservation to be considered guaranteed and for the property to process any pre-payments.

9
References

Find additional documentation on Oracle Payment Interface and Oracle Payment Interface Cloud at https://docs.oracle.com/en/industries/hospitality/integration_platforms.html.

10

Key Terminology

Term	Description
Tokenized	The storage of tokens in place of credit card numbers. Accomplished with the Token Exchange process.
Token Exchange	The process of exchanging a credit card number for a token. This process works by replacing a cardholder's Primary Account Number (PAN)/Credit Card with a one-time unique identifier.
Oracle Payment Interface (OPI)	This provides OPERA Cloud users an interface for industry-standard payment acceptance and opens the door to an ecosystem of Payment Service Providers (PSP). Through the set of services offered by OPI, OPERA Cloud can communicate with PSPs for card processing and token exchange transactions. OPI Cloud service also exposes a Token Exchange API through Oracle Hospitality Integration Platform so that partner applications can use the API to exchange the card number with a token for a hotel property that also uses the OPI Cloud Service for card transactions.
Payment Service Providers (PSP)	PSPs process the card data and provide a token through OPI. This token represents the card data and can be used for financial transactions for the property. PSPs must be certified with Oracle for OPI functionality.

11 FAQ

How to activate OPI Cloud Service?

Oracle is responsible for OPI Cloud to be made available. There will be more availability as more PSPs certify for use with OPI Cloud.

When OPI is active, is the full card number stored in OPERA PMS?

No. The card number is stored with the Payment Service Provider. OPERA Cloud will only store the token in place of the card number.

Can the token, in place of the card number, be used for subsequent transactions?

Yes, the token is in place to process payments, but can only be used for the reservation for which it was generated.

Can a token number be manually entered into OPERA Cloud?

No

Can a partner send a token number through to OPERA Cloud?

Yes

Can a partner unmask the Credit Card number?

No

Can a partner fetch a Credit Card number from OPERA Cloud and then tokenize the card and return the token to OPERA Cloud?

No, Credit Card Numbers are not provided by OPERA Cloud.

Can a partner send just the first 4 and last 4 digits of the card only when OPERA Cloud is not tokenized?

No. If a property is processing credit cards for payments, then full PAN is needed.

Does OPI ensure the hotel is PCI Compliant?

Deployment of OPI significantly reduces the requirements during a PCI audit as there will no longer be card data stored. However, it is important to have the entire network and all software following the PCI Security Council Rules.

Can I find more information on the financial transactions, like Deposits and Authorizations?

Yes, Implementation Guides also exist for these areas. *View Deposit Handling, Authorization and Settlements, and Above Property Payments.*

Can we have a PSP send a token to us directly where we send these tokens to OPERA Cloud through OHIP? For example, we are connected to Adyen, but the customer has Wordline. Will my token be recognized?

No. Only if you are using the same PSP as the customer.

Is there a way to find which PSP a customer is connected to through OPI?

Not at this time. You must ask the customer for their Payment Provider.

I don't want the customer OPI to tokenize, but we as a third party system want to tokenize. Is this possible?

Yes, you can do this, but only a token that is from the same PSP is useful with the customer's OPI in OPERA Cloud.