

Known Issues for Oracle Blockchain Platform on Oracle Cloud Infrastructure

Learn about the issues you may encounter when using Oracle Blockchain Platform and how to work around them.

Topics:

- [Storage and OCPU Scaling Not Supported](#)
- [Hyperledger Fabric SDKs Deprecated](#)
- [Incorrect Storage Value Reported in REST API](#)
- [Some Fields Not Returned by Verify Blocks REST API](#)
- [Can't View Ledger Details for a Channel](#)
- [Rich History Configuration Not Saved](#)
- [Go Chaincode Version Compatibility](#)
- [Node.js Chaincode Version Compatibility](#)
- [Unchanged On-Hold Balances After Minting and Burning Approvals](#)

Issues Initially Reported in Previous Releases:

- [Account Limit Error When Upgrading Go Chaincodes](#)
- [Unexpected Token Error When Debugging on Microsoft Windows 11](#)
- [Required Go Syntax for No Daily Account Limits](#)
- [REST Proxy Timeout Errors When Using Postman Collections](#)
- [Function List in Visual Studio Code Includes Two Methods Incorrectly](#)
- [App Builder CLI Is Uninstalled If Prerequisites Check Fails](#)
- [No Error When a Fractional NFT Transaction Uses Incorrect Number of Decimal Places](#)

- [Cannot Run Fiat Money Token or Balance Transfer Sample From the Developer Tools Page](#)
- [Go Deployment Error When Using Blockchain App Builder](#)
- [Initialization Failure in TypeScript Chaincode Generated by Blockchain App Builder](#)
- [Gateway Timeouts When Using the XA Library](#)
- [Cannot Run Token Chaincode in Debug Mode on Microsoft Windows](#)
- [Event Size Limits](#)
- [User IDs That Contain a Colon \(:\) Can't Be Used in REST API Calls](#)
- [CORS Header Not Returned for Invalid Credentials](#)
- [Orderers Status Error When Creating Channels](#)
- [Debugging in Visual Studio Code on Microsoft Windows](#)
- [Multi-Organization Environments and Blockchain App Builder](#)
- [Scaling Only Works on One Component at a Time](#)
- [Incorrect Operating System Clock May Result in Rejected Requests](#)
- [Blockchain Applications Aren't Working as Expected Due to an Older Client SDK](#)
- [The Network's Oracle Blockchain Platform Instances Can't Manage Revoked Third-Party Certificates](#)
- [Founder's Channel List Contains Incorrect Created by Information and Edit Channel Organizations Option Isn't Available](#)
- [ImplicitMeta Policy Isn't Supported by Oracle Blockchain Platform](#)
- [Channel Creator Can't Update the Channel's Configuration](#)
- [Setting blocktolive to 0 in instantiateChaincode Endpoint Not Supported in REST API](#)
- [Peer Fails to Pull Private Data from Another Peer](#)
- [Channel Creator Organization and Channel Policy Settings Inconsistency](#)
- [Exported and Imported File Incompatibility](#)
- [Chaincode Name Requirements](#)
- [Date and Time Picker Behavior](#)

Storage and OCPU Scaling Not Supported

Scaling storage or OCPU capacity is not supported on Oracle Blockchain Platform version 25.4.1 instances. If you attempt to use the REST API to scale storage or OCPU capacity, an `UnsupportedOperationException` error message is returned.

Hyperledger Fabric SDKs Deprecated

The Hyperledger Fabric SDKs are now deprecated in favor of the Fabric Gateway client API. It might still be possible to use the Hyperledger Fabric SDKs to develop applications for Oracle Blockchain Platform, but the Fabric Gateway client API is preferred.

Incorrect Storage Value Reported in REST API

The `storageUsedInTBs` value returned by the Get Blockchain Platform Information REST API endpoint always returns 0 for Oracle Blockchain Platform version 25.4.1 instances.

Workaround: There is no workaround for this issue.

Some Fields Not Returned by Verify Blocks REST API

The following fields are not returned by the Verify Blocks REST API endpoint on Oracle Blockchain Platform version 25.4.1 instances: `verifyResult` and `firstBlockNum`.

Workaround: There is no workaround for this issue.

Can't View Ledger Details for a Channel

When you click a block on the ledger page for a channel, the transaction details are displayed. There is an intermittent issue where sometimes the transaction details are not displayed when you click a block.

Workaround: Refresh the browser page and then click the block again to show the transaction details.

Rich History Configuration Not Saved

When you click **Save** in the Configure Rich History dialog box, the UI indicates that the global configuration was saved successfully; however, when you try to view the rich history configuration, it is not available because it was not actually saved. This is an intermittent issue.

Workaround: Retry saving the rich history configuration.

Go Chaincode Version Compatibility

For version 25.4.1 of Oracle Blockchain Platform, the version of Go that is used was updated to 1.24.4. If you attempt to deploy or upgrade a chaincode that is not compatible with Go 1.24.4, the process might fail. Before you upgrade or deploy a chaincode that is written in Go, ensure that the chaincode can build with Go 1.24.4.

You can use the following commands on the contents of the chaincode folder on a system where Go 1.24.4 is installed to check for errors:

```
go mod tidy
```

```
go mod vendor
```

Currently deployed chaincodes are not affected by this behavior and will continue to work as expected after you upgrade to Oracle Blockchain Platform 25.4.1.

Node.js Chaincode Version Compatibility

For version 25.4.1 of Oracle Blockchain Platform, the version of Node.js that is used was updated to 22.14.0. If you attempt to deploy or upgrade a chaincode that is not compatible with Node.js 22.14.0, the process might fail. Before you upgrade or deploy a chaincode that is written in Node.js, ensure that the chaincode can build with Node.js 22.14.0.

Currently deployed chaincodes are not affected by this behavior and will continue to work as expected after you upgrade to Oracle Blockchain Platform 25.4.1.

Unchanged On-Hold Balances After Minting and Burning Approvals

Minting and burning approvals use the hold mechanism, which takes an operation ID as an input. The methods `getOnHoldBalanceWithOperationId` and `getOnHoldDetailsWithOperationId` return the hold amount that is associated with a specified operation ID. For standard hold operations that use the `executeHold` and `releaseHold` methods, the hold amount changes after the method completes. For example, if you request a hold with operation ID of `first` for 100 tokens, after approval (using the `executeHold` method), the hold amount is 0. The `getOnHoldBalanceWithOperationId` and `getOnHoldDetailsWithOperationId` methods will return 0 as the hold amount for the operation ID `first`.

This behavior does not apply to minting or burning approvals. For example, if you make a request to mint 100 tokens with the operation ID of `second`, after approval (using the `approveMint` method), the hold amount is still 100. The `getOnHoldBalanceWithOperationId` and `getOnHoldDetailsWithOperationId` methods will return 100 for the operation ID `second`.

In the case of minting or burning, the approval represents a request to mint or burn that amount, but no transfer occurs at the approval stage. Because the minting and burning approvals use the same internal hold mechanism as standard hold operations, the query methods return the value that is stored in the hold key-value pair, which is not changed after minting or burning approvals.

Workaround: You can use the `getOnHoldBalanceWithOperationId` and `getOnHoldDetailsWithOperationId` methods to verify the requested mint or burn

amounts, but be aware of the differences between those operations and standard hold operations.

Issues Initially Reported in Previous Releases:

Account Limit Error When Upgrading Go Chaincodes

If you upgrade chaincode that was generated by a previous version of Blockchain App Builder in the Go language using the extended Token Taxonomy Framework standard to version 25.2.2, you might see the following error when you try to complete a transaction:

```
account has exceeded the limit for maximum number of transfers -1
```

This error also occurs on the Go version of the chaincode that is specific to the wholesale CBDC model. This error occurs because the previous version of Blockchain App Builder used the value `-1` to represent no daily account limits, but in version 25.2.2 there is no default value and the absence of that key/value pair indicates that the account does not have daily limits.

Workaround: Use the following methods to set new default values for account limits. If you do not pass any values for the optional `max_daily_amount` or `max_daily_transaction` parameters, the key/value pairs will be removed so that the account will work correctly with version 25.2.2.

```
SetMaxDailyAmount(token_id string, org_id string, user_id string,  
max_daily_amount ...float64)
```

```
SetMaxDailyTransactionCount(token_id string, org_id string, user_id  
string, max_daily_transactions ...float64)
```

Unexpected Token Error When Debugging on Microsoft Windows 11

When you use the Blockchain App Builder debug function in Visual Studio Code running on Microsoft Windows 11, the following error message is displayed:

```
Unexpected token ':'"admin"' in expression or statement.
```

When this error occurs, chaincode deployment and debugging fail.
Workaround: There is no workaround for this issue.

Required Go Syntax for No Daily Account Limits

In chaincode generated by Blockchain App Builder that uses the extended Token Taxonomy Framework standard and the Go programming language, if you do not want to set the daily account limits when using the `CreateAccount` method, you must pass the empty value `"` and not `"{}"` for the `daily_limits` parameter. The correct parameters for the Go version of the `CreateAccount` method look similar to the following example:

```
["CentralBank","cb","fungible",""]
```

This Go version of the chaincode specific to the wholesale CBDC model also has this requirement.

REST Proxy Timeout Errors When Using Postman Collections

For the Postman collections generated by Blockchain App Builder, which run transactions by using the Oracle Blockchain Platform REST proxy, the default timeout value is set to 6000, as shown in the following example:

```
{
  "chaincode": "WholesaleCBDCConfidential",
  "args": [
    "getTokenAuditors"
  ],
  "timeout": 6000,
  "sync": true,
  "peer": "org-xyz-abc.blockchain.ocp.oraclecloud.com:20009",
  "transientMap": {
    "args": "[]"
  }
}
```

In some cases the REST proxy processing can time out even though the transaction is actually being processed by Oracle Blockchain Platform, which results in the following response:

```
{
  "returnCode": "Failure",
  "error": "failed to invoke chaincode: Client Status Code: (5)
TIMEOUT. Description: request timed out or been cancelled",
  "result": null
}
```

Workaround: Increase the timeout value based on your specific use case.

Function List in Visual Studio Code Includes Two Methods Incorrectly

When you use the confidential transactions feature with the Visual Studio Code version of Blockchain App Builder, two methods are shown in the function list on the Execute pane: `executeHoldTokensSender` and `executeHoldTokensReceiver`. These two methods work correctly only when used with the two-phase commit API of the REST proxy, which the Visual Studio Code version of Blockchain App Builder does not support. Do not use these two methods in the Visual Studio Code version of Blockchain App Builder.

App Builder CLI Is Uninstalled If Prerequisites Check Fails

When you upgrade the Blockchain App Builder command-line interface (CLI) by installing or reinstalling the latest version, if the prerequisites check fails, the existing Blockchain App Builder CLI is uninstalled.

Workaround: You can either reinstall the older version of the Blockchain App Builder CLI, or update your system to meet the prerequisites and then install the newer version of the Blockchain App Builder CLI.

No Error When a Fractional NFT Transaction Uses Incorrect Number of Decimal Places

When you use Blockchain App Builder to work with fractional NFTs that use the ERC-1155 standard, if a transaction amount uses a larger number of decimal places than the maximum number of decimal places in the specification file, an error is not indicated. Instead, the actual transaction amount is truncated to the number of decimal places defined in the specification file. Additionally, even though the sender's and receiver's balances are truncated to the specified number of digits, the `transactedAmount` value in the transaction history shows the untruncated value.

Workaround: Validate a transaction amount before creating a transaction to ensure that the transaction amount does not use a larger number of decimal places than what was specified in the input specification file for the chaincode.

Cannot Run Fiat Money Token or Balance Transfer Sample From the Developer Tools Page

A new version of the Fiat Money Token sample is included with Oracle Blockchain Platform 22.4.2. The updated chaincode has a different package ID than the previous version. If you deployed the Fiat Money Token sample, after upgrading to Oracle Blockchain Platform 22.4.2 the sample will no longer be shown as deployed on the

Developer Tools page in the console and you will not be able to run any operations from the Developer Tools page for the sample.

The same behavior occurs with the Balance Transfer sample, which was updated for Oracle Blockchain Platform 22.3.2 to work with the atomic transactions function.

Workaround: Install the sample chaincode again on the peers, which will install the newer version of the chaincode with the new package ID. For each channel where the sample was deployed, upgrade the chaincode to the new package. After the chaincode is upgraded, the status and operations of the sample on the Developer Tools page will function normally.

Go Deployment Error When Using Blockchain App Builder

You might encounter an error similar to the following when deploying a Go chaincode project by using Blockchain App Builder:

```
ERROR RunCommand: spawn modd ENOENT
ERROR RunCommand: Error in Chaincode deployment
[+] Running 3/3
  Container ca.example.com Stopped 0.4s
  Container orderer.example.com Stopped 0.6s
  Container peer0.org1.example.com Stopped 0.6s
ERROR RunCommand: Error in Chaincode deployment, process exit with code
1
```

Workaround:

macOS

1. Run the following command:

```
env G0111MODULE=off go get `github.com/cortesi/modd/cmd/modd`
```

2. Retry the deployment.

Microsoft Windows

1. Use `go install` to run the following command:

```
go install github.com/cortesi/modd/cmd/modd@latest
```

2. Use `go get` to run the following commands:

```
SET G0111MODULE=auto
go get `github.com/cortesi/modd/cmd/modd`
```

3. Retry the deployment.

Initialization Failure in TypeScript Chaincode Generated by Blockchain App Builder

If you generate TypeScript chaincode with Blockchain App Builder version 22.4.2 or earlier, you might see errors similar to the following text when you deploy or run the chaincode locally or remotely:

Local environment:

```
INFO: Error in initalizing chaincode. Error : failed to initialize
chaincode Error: Failed to
initialize the chaincode Error: endorsement failure during invoke.
response: status:500 message:"error
in simulation: failed to execute transaction
e22ba18c00ecbd3135cdb509e08667cf6d5d9e79c4217b73492b5bb50836d58d: could
not launch chaincode
testagain:v0: error building chaincode: error building image: failed to
get chaincode package for
external build: could not get legacy chaincode package 'testagain:v0':
open
/var/hyperledger/production/chaincodes/testagain.v0: no such file or
directory
```

Remote environment:

INFO:

```
===== Started Initializing Chaincode =====
```

```
ERROR: {
  "Error": "Chaincode Deployment failed. Error in initializing
chaincode: Status Code: 400,
Error Message: {\"returnCode\": \"Failure\", \"error\": \"failed to invoke
chaincode:
Transaction processing for endorser
[jasfounderdec5-oabcs1-ams.blockchain.ocp.oraclecloud.com:20010]:
Chaincode status Code:
(500) UNKNOWN. Description: error in simulation: failed to execute
transaction
947bbaf2feccc39cdf53bd7a07cd17f15d682a5a4ee6e3c3e63dec6346b0394e: error
sending: chaincode
stream terminated\", \"result\": null}"
}
```

Workaround: To work around this behavior for existing chaincodes, complete the steps at the following support page:

[Deployment of TypeScript Chaincode in Both Local and Remote OBP Environment Fail](#)

To avoid this behavior in new chaincodes, upgrade to Blockchain App Builder 22.4.3 or later. You can download Blockchain App Builder from the **Developer Tools** tab on the Blockchain App Builder pane of the Oracle Blockchain Platform console.

Gateway Timeouts When Using the XA Library

When you use the XA Java library with Oracle Blockchain Platform, multiple blockchain transactions can be combined into a global transaction that is controlled by an external transaction manager. If the blockchain transactions take longer than the gateway timeout value of 60 seconds to complete, then the transaction status can become nondeterministic on the transaction manager side and the Oracle Blockchain Platform side.

Workaround: There is no workaround for this issue. In this case you must manually resolve the transaction.

Cannot Run Token Chaincode in Debug Mode on Microsoft Windows

If you use Blockchain App Builder version 22.2.1 or earlier, you cannot run token chaincode in debug mode on Microsoft Windows.

Workaround: Upgrade to the latest version of Blockchain App Builder. If you are unable to upgrade, complete the following steps:

1. Open the `chaincode/.vscode/task.json` file for editing.
2. The sixth line of the `task.json` file includes the `command` key. Remove the following string from the line:

```
-p '${workspaceFolder}'
```

For example, the line in the `task.json` file before editing:

```
"command": "ochain debug -p '${workspaceFolder}' \"[{\\\\"user\\\\"} : \\\\"admin\\\\" , \\\\"orgId\\\\" : \\\\"Org1MSP\\\\"}]\" -v v8",
```

After:

```
"command": "ochain debug \"[{\\\\"user\\\\"} : \\\\"admin\\\\" , \\\\"orgId\\\\" : \\\\"Org1MSP\\\\"}]\" -v v8",
```

Event Size Limits

Starting with version 22.2.1, by default the maximum payload size of an event is limited to 50 KB. Any events larger than the maximum payload size will be dropped. The Oracle DevOps team can modify this parameter by request. If you expect to subscribe to events where the payload will be larger than 50 KB, open a Service

Request (SR) in My Oracle Support to request a larger maximum event size. For more information, see [Subscribe to an Event](#) in the REST API documentation.

User IDs That Contain a Colon (:) Can't Be Used in REST API Calls

Oracle Blockchain Platform allows you to associate a user ID that contains a colon (:) with a REST proxy enrollment. However, that user ID cannot be used in REST API calls when basic authentication (user name and password) is used.

Workaround: Ensure that all users associated with REST proxy enrollments do not have a colon (:) in their user IDs.

CORS Header Not Returned for Invalid Credentials

Customer applications that invoke REST proxy transactions will not receive the Cross-Origin Resource Sharing (CORS) header (the `Access-Control-Allow-Origin` header) in the response if the credentials that were sent in the request are invalid, incorrect, or expired.

Orderers Status Error When Creating Channels

When creating a channel, you might see the following error:

```
Failed to create the channel with error: aborted
Please check the orderers status.
```

Workaround: Try to create the channel again. This is an intermittent issue.

Debugging in Visual Studio Code on Microsoft Windows

On Microsoft Windows, you might encounter an error similar to the following while debugging chaincode projects in Visual Studio Code:

```
dlv: failed to install dlv(github.com/go-delve/delve/cmd/dlv@latest):
Error: Command failed:
C:\Program Files (x86)\Go\bin\go.exe get -x github.com/go-delve/
delve/cmd/dlv@latest
# get https://proxy.golang.org/github.com/go-delve/delve/cmd/dlv/@v/list
# get https://proxy.golang.org/github.com/@v/list
# get https://proxy.golang.org/github.com/go-delve/@v/list
# get https://proxy.golang.org/github.com/go-delve/delve/cmd/@v/list
# get https://proxy.golang.org/github.com/go-delve/delve/@v/list
```

```
# get https://proxy.golang.org/github.com/@v/list: 410 Gone (0.420s)
# get https://proxy.golang.org/github.com/go-delve/delve/cmd/@v/list:
410 Gone (1.040s)
# get https://proxy.golang.org/github.com/go-delve/@v/list: 410 Gone
(1.062s)
# get https://proxy.golang.org/github.com/go-delve/delve/cmd/dlv/@v/
list: 410 Gone (1.066s)
# get https://proxy.golang.org/github.com/go-delve/delve/@v/list: 200
OK (1.448s)
go: found github.com/go-delve/delve/cmd/dlv in github.com/go-delve/
delve v1.8.3C:\Users\\go\pkg\mod\github.com\go-
delve\delve@v1.8.3\service\debugger\debugger.go:28:2:found packages
native (proc.go) and
your_operating_system_and_architecture_combination_is_not_supported_by_d
elve(support_sentinel.go) in C:\Users\Asus\go\pkg\mod\github.com\go-
delve\delve@v1.8.3\pkg\proc\native
```

There is no workaround for this error at this time.

Multi-Organization Environments and Blockchain App Builder

In an environment with multiple organizations, you might need to use the console to complete some operations.

To re-deploy the chaincode on the same channel through a participant instance, use the console to deploy the chaincode.

(Hyperledger Fabric v2.2.4) To upgrade the chaincode, use the console and manually approve the chaincode from the participants.

Scaling Only Works on One Component at a Time

You can only scale one node type at a time. For example, you can make add peer nodes and modify existing peer node settings at the same time, but if you also want to increase your storage you must do that separately.

Additionally, you can only scale one peer or OSN at a time - for example you can't add two peers in a single operation.

Incorrect Operating System Clock May Result in Rejected Requests

If the client or SDK's local clock is more than 15 minutes off, the requests from it will be rejected by the peer and orderer. Ensure your local clock is set correctly.

Blockchain Applications Aren't Working as Expected Due to an Older Client SDK

An application can behave unexpectedly if it's using an older version of the client SDK.

Workaround: Read the documentation describing the client SDK updates and modify your applications as needed. For more information, see [Hyperledger Fabric SDKs](#) in the Hyperledger Fabric documentation.

The Network's Oracle Blockchain Platform Instances Can't Manage Revoked Third-Party Certificates

If an Oracle Blockchain Platform network contains either organizations with third-party certificates or Hyperledger Fabric organizations and their certificates are revoked, then the revoked certificates aren't applied to, won't display in, and can't be revoked from the network's Oracle Blockchain Platform instances.

Workaround: Use the native Hyperledger Fabric CLI or SDK to import the organization's certificate revocation list (CRL) file.

Founder's Channel List Contains Incorrect Created by Information and Edit Channel Organizations Option Isn't Available

In a mixed network (where a founder instance and participant instance are running different versions of Oracle Blockchain Platform), the founder's channel list may display the wrong MSP ID for a channel created by a participant. Instead of the participant's MSP ID, the founder's MSP ID is displayed. This may happen after you import the CRL, revoke or apply the CRL, or set an anchor peer on a channel.

The channel's **Edit Channel Organizations** option is only available for the instance displayed in the **Created by** field. If the wrong MSP ID is displayed, then the channel creator can't update the channel organizations.

Workaround: There is no workaround for this issue.

ImplicitMeta Policy Isn't Supported by Oracle Blockchain Platform

If you use the native Hyperledger Fabric CLI or SDK to modify a channel's configuration, some of the configuration settings you specify can't be supported by Oracle Blockchain Platform.

- The native Hyperledger Fabric CLI and SDK use the ImplicitMeta channel policy for readers and writers. When the channel uses these policies, the Oracle Blockchain Platform console can't guarantee that the administrative operations (for example, edit organization) can be successfully processed.

Workaround: Update the readers and writers policies to the Signature policies, and define the policy rules as needed. For more information, see [Access Control Lists \(ACL\)](#) in the Hyperledger Fabric documentation.

- If a channel is using the ImplicitMeta policy type and in the channel configuration you change the `mod_policy` in the groups section to Admins and there is more than one organization in the channel, then you can't use Oracle Blockchain Platform to manage the channel. For example, you can't add new organizations to the channel or change the channel's ACL policy in any way, including restoring its original value.

Workaround: Use the native Hyperledger Fabric CLI or SDK to manage the channel.

Channel Creator Can't Update the Channel's Configuration

When you use the native Hyperledger Fabric CLI or SDK to create a channel, the Creator policy isn't included in the `configtx.yaml` file. Oracle Blockchain Platform requires the Creator policy to allow the channel creator to edit a channel's configuration.

Workaround: Manually edit the `configtx.yaml` file to add the Creator policy.

Setting `blocktolive` to 0 in `instantiateChaincode` Endpoint Not Supported in REST API

If you're using the REST API's `instantiateChaincode` endpoint and in the `dataCollectionConfig` you set the `blocktolive` value to 0, then you'll receive the following error: `{"respMsg": "invalid argument"}`.

To prevent purging data from the private database, Hyperledger Fabric requires you to set the `blocktolive` value to 0. However, the Oracle Blockchain Platform REST API doesn't support setting this configuration to 0.

Workaround: Use the console to instantiate the chaincode, and in the Instantiate Chaincode dialog's Private Data Collections section, set the **blocktolive** field to 0.

Peer Fails to Pull Private Data from Another Peer

A peer can fail to pull private data from another peer if a private data collection's `blocktolive` value is less than 10 and its `maxPeerCount` value is less than the total number of peers, not including the endorsing peer. This value is set when you use the

console to create a private data collection definition or use the native Hyperledger Fabric CLI or SDK.

Workaround: Confirm that the `blocktolive` value is set to greater than or equal to 10. Or, confirm that the `maxPeerCount` value is set to no less than the total number of peers, not including the endorsing peer. If needed, you can re-instantiate or upgrade the chaincode to reset these values.

Channel Creator Organization and Channel Policy Settings Inconsistency

You can use the console to create a channel and set your organization's ACL to `ReaderOnly`. After you save the new channel, you can't update this ACL setting from the channel's **Edit Channel Organizations** option.

However, you can use the console's **Manage Channel Policies** option to add your organization to the Writers policy, which overwrites the channel's `ReaderOnly` ACL setting.

Workaround: There is no workaround for this issue.

Exported and Imported File Incompatibility

You can't export and import files (CRLs, certificates, ordering service settings, and peers) between the console and the REST APIs.

Files exported by the console and REST APIs are only compatible for import with the same component. For example, if you export a peer using the console, then you can't import it with the REST API (you can only import it with the console). And if you export a peer with the REST API, then you can't import it with the console (you can only import with the REST API).

Workaround: There is no workaround for this issue.

Chaincode Name Requirements

The Oracle Blockchain Platform chaincode name and version requirements are different than the Hyperledger Fabric requirements. You must use the Oracle Blockchain Platform requirements when you deploy a chaincode from the console or the Hyperledger Fabric client. If you don't follow these requirements when deploying from the Hyperledger Fabric client, then the chaincode may be listed incorrectly in the console.

Workaround: Use the following rules when deploying a chaincode name and version.

- Use ASCII alphanumeric characters, dashes (-), and underscores (_).
- The name must start and end only with ASCII alphanumeric characters. For example, you can't use names like `_mychaincode` or `mychaincode_`.

- Dashes (-) and underscores (_) must be followed with ASCII alphanumeric characters. For example, you can't use names like `my--chaincode` or `my-_chaincode`.
- The name and version can each be up to 64 characters long.
- The chaincode version can also contain periods (.) and plus signs (+).

Date and Time Picker Behavior

The Oracle Blockchain Platform date and time picker doesn't behave as expected. You use the date and time picker to filter items such as log files or ledger activity.

Workaround: Use the following information to help you use the date and time picker.

- If you select a specific time period (for example, **Last day**) and then select it again to re-run the query, the query doesn't re-run. To get the latest information, click the refresh button.
- If you haven't set the time zone on your computer, then when you select the **Custom** option, you must specify the start time and end time in GMT. However, if you set the Timezone Setting to **GMT** in the Preferences (in the console select your instance name, then click Preferences, and then Timezone Setting), the timezone on the console automatically converts to GMT.

Oracle Cloud Known Issues for Oracle Blockchain Platform On Oracle Cloud Infrastructure,

F26722-20

Copyright © 2020, 2026, 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.