Troubleshoot PostgreSQL Flexible server Provisioning Issues

Last updated by | Terence Mills | Mar 17, 2023 at 4:28 PM PDT

Issue

This troubleshooting guide will help to troubleshoot Azure Database for PostgreSQL Flexible server provisioning issues.

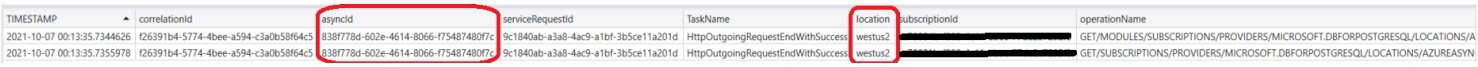
Investigation/Analysis

Get the error message from the Customer along with Tracking ID or Correlation Id (Correlation ID is preferred)

Sample Error Message : "message": "An unexpected error occured while processing the request. Tracking ID: '9c1840ab-a3a8-4ac9-a1bf-3b5ce11a201d'"

1.In Kusto under connections, go to ARMProd. Then Replace Timestamp, Subscription ID, Correlation ID or ServiceRequestID (Tracking ID from the Error message) and get the ASYNCID from the below queries

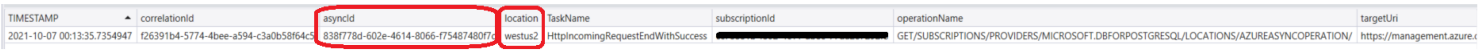
```
//Get ASYNC ID from the below - Replace Timestamp, Subscription ID, Correlation ID and get the asyncId
HttpOutgoingRequests
| where TIMESTAMP > ago(7d)
//| where TIMESTAMP > datetime(2021-10-01 23:20) and TIMESTAMP < datetime(2021-10-04 23:25)
//| where subscriptionId =~ "customersubscriptionid"
| where serviceRequestId =~ '9c1840ab-a3a8-4ac9-a1bf-3b5ce11a201d' //Tracking ID from Customer
//| where correlationId =~ 'b1b44599-9443-4438-803b-9ae2984ffae5'
| where operationName contains "AZUREASYNCOPERATION"
| extend location = extract('locations/([a-z|A-Z|0-9|]+)', 1, targetUri, typeof(string))
| extend asyncId = extract('azureAsyncOperation/([a-z|A-Z|0-9|-]+)', 1, targetUri, typeof(string))
| where isnotempty( asyncId) //and location contains "AddRegionHere"
| project TIMESTAMP, correlationId, asyncId, serviceRequestId,TaskName, location, subscriptionId, oper
```



TIMESTAMP	correlationId	asyncId	serviceRequestId	TaskName	location	subscriptionId	operationName
2021-10-07 00:13:35.7344626	f26391b4-5774-4bee-a594-c3a0b58f64c5	838f778d-602e-4614-8066-f754874 80f7c	9c1840ab-a3a8-4ac9-a1bf-3b5ce11a201d	HttpOutgoingRequestEndWithSuccess	westus2	[redacted]	GET/MODULES/SUBSCRIPTIONS/PROVIDERS/MICROSOFT.DBFORPOSTGRESQL/LOCATIONS/A
2021-10-07 00:13:35.7355978	f26391b4-5774-4bee-a594-c3a0b58f64c5	838f778d-602e-4614-8066-f754874 80f7c	9c1840ab-a3a8-4ac9-a1bf-3b5ce11a201d	HttpOutgoingRequestEndWithSuccess	westus2	[redacted]	GET/SUBSCRIPTIONS/PROVIDERS/MICROSOFT.DBFORPOSTGRESQL/LOCATIONS/AZUREASYN

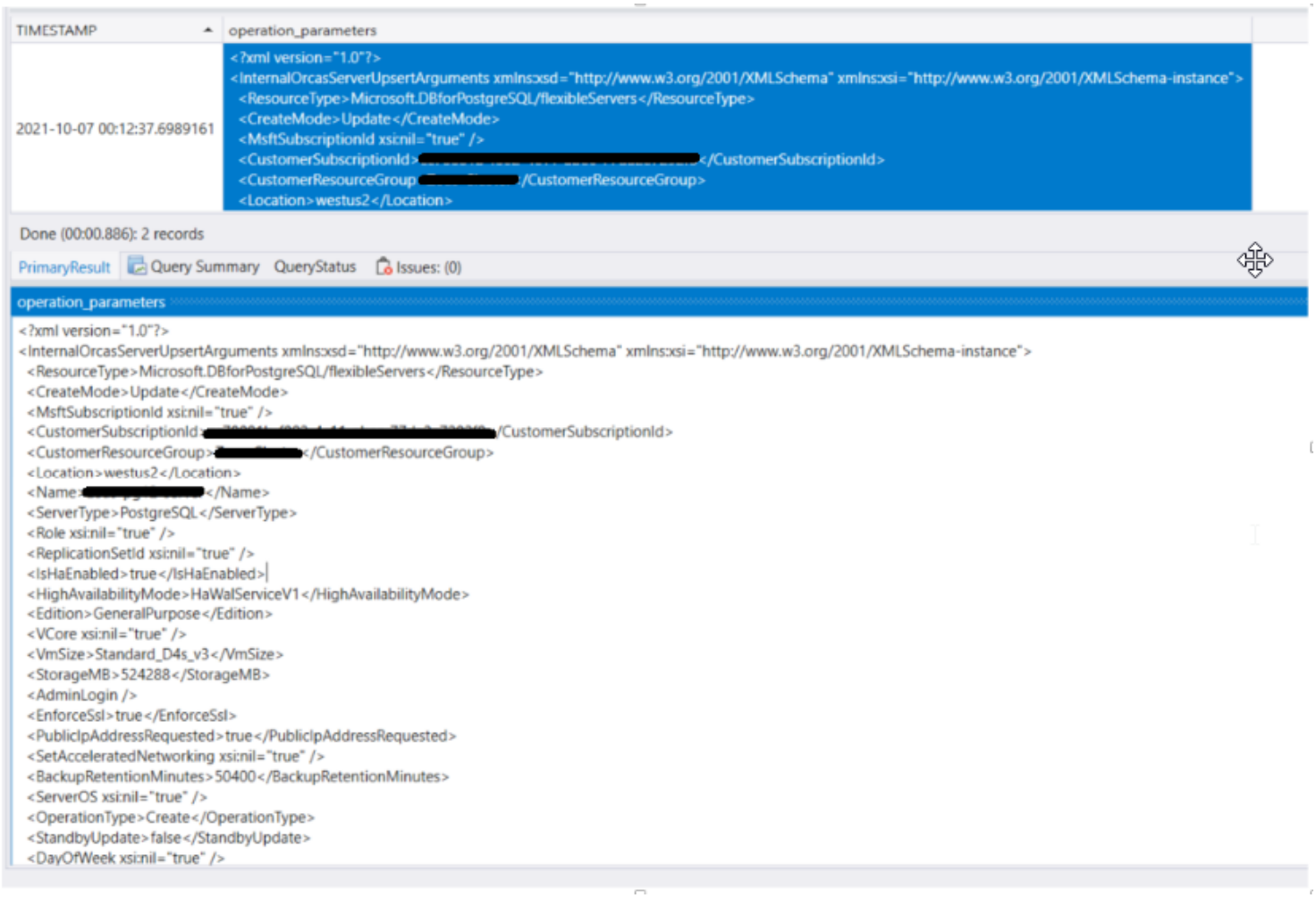
2. You can get the same information from below query also.

```
//Get ASYNC ID from the below - Replace Timestamp, Subscription ID, Correlation ID and get the asyncId f
HttpIncomingRequests
| where TIMESTAMP > ago(7d)
//| where TIMESTAMP > datetime(2021-10-01 23:20) and TIMESTAMP < datetime(2021-10-04 23:25)
//| where subscriptionId =~ "customersubscriptionid"
| where serviceRequestId =~ '9c1840ab-a3a8-4ac9-a1bf-3b5ce11a201d' //Tracking ID from Customer
//| where correlationId =~ 'b1b44599-9443-4438-803b-9ae2984ffae5'
| where operationName contains "AZUREASYNCOPERATION"
| extend location = extract('locations/([a-z|A-Z|0-9|]+)', 1, targetUri, typeof(string))
| extend asyncId = extract('azureAsyncOperation/([a-z|A-Z|0-9|-]+)', 1, targetUri, typeof(string))
| where isnotempty( asyncId) //and location contains "AddRegionHere"
| project TIMESTAMP, correlationId, asyncId, serviceRequestId,TaskName, location, subscriptionId, operat
```



3. Provide the ASYNC ID from the step 1 in the below query in request_id column to find the operation parameters to get the details about the operation.

```
//Get the Operational Parameters for the server creation
MonOrcasBreadthRp
| where TIMESTAMP > ago(7d)
//| where TIMESTAMP > datetime(2021-10-04 23:20) and TIMESTAMP < datetime(2021-10-04 23:25)
| where request_id =~ "838f778d-602e-4614-8066-f75487480f7c"
//| where correlation_id =~ 'e80b02ed-2a57-416e-98de-06921e04ffad'
| where operation_parameters !=''
| project TIMESTAMP, operation_parameters
```

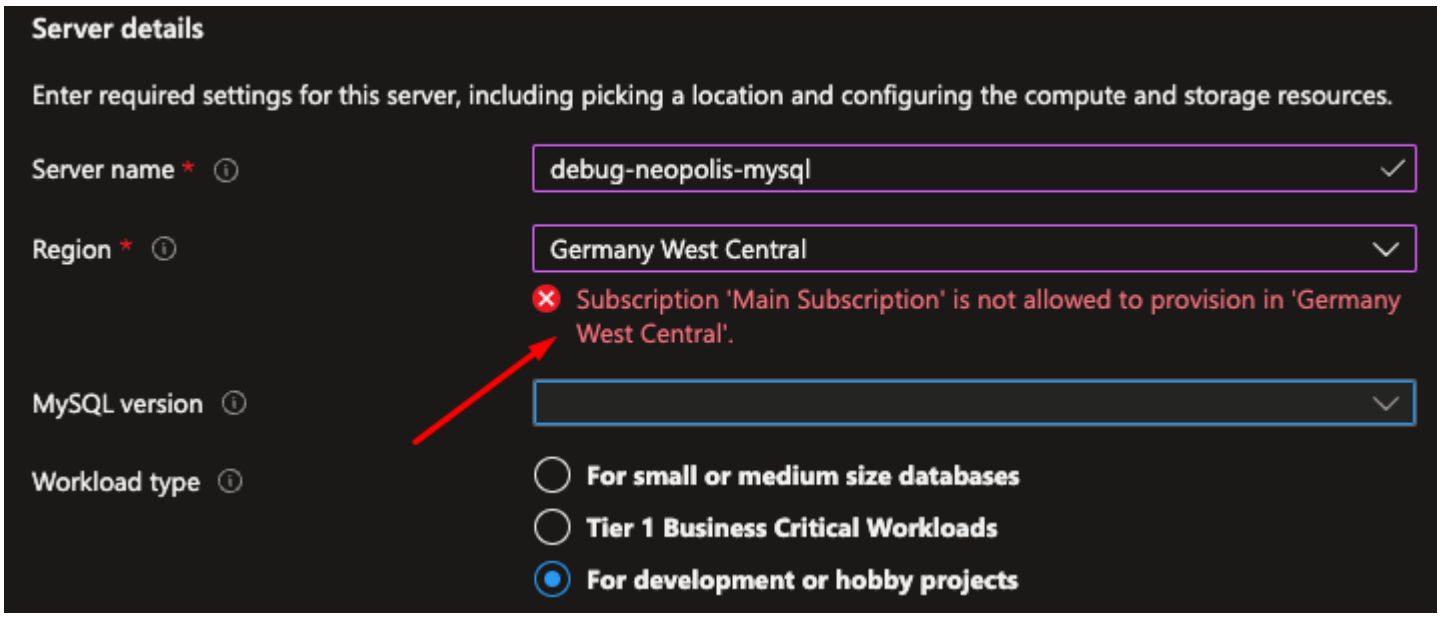


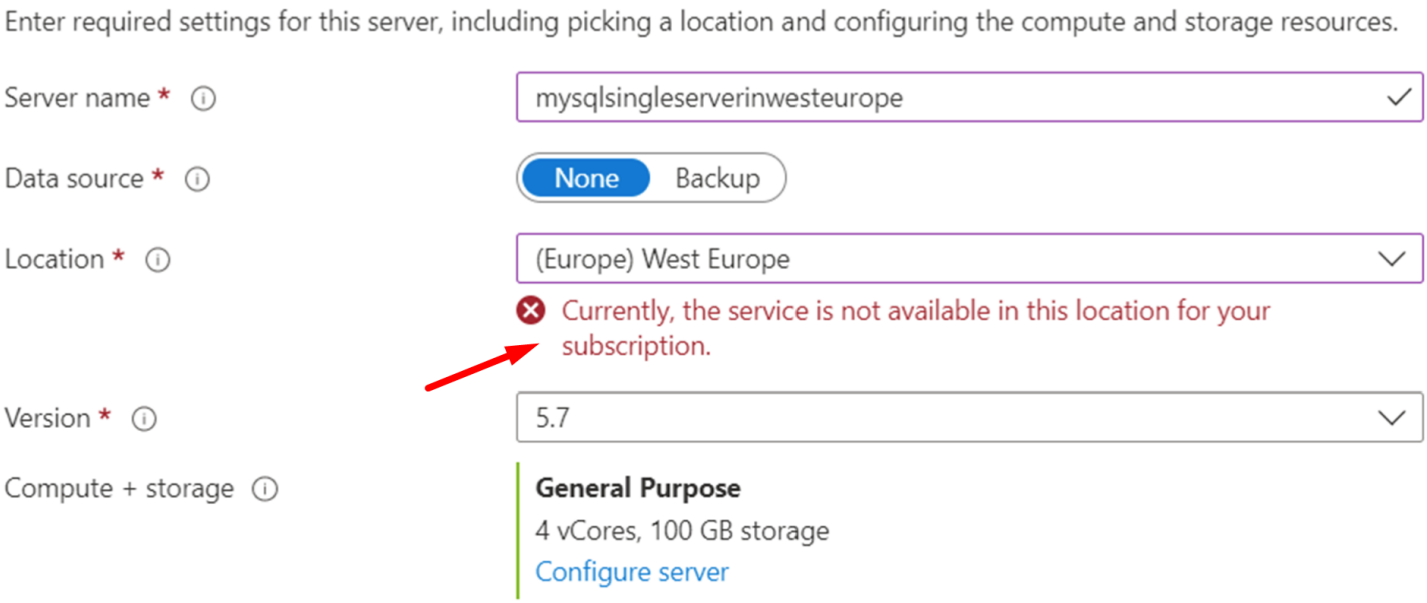
Note in case ASYNC ID is not available

In case ASYNC ID is not obtained running the Kusto queries from action items 1 or 2, this indicates that the ARM request never made it to the backend and therefore the data field 'request_id' is unavailable for the error to be further tracked down. This could have assorted reasons, such as discontinued or unavailable SKU in the region, wrong parameter selection, or the current capacity constraint (Spring 2022).

In case the subscription, region or SKU tier is restricted, customers will likely see a message in Azure Portal deployment page informing the restriction and preventing them from proceeding with the deployment. This will not be the case if they use Terraform or any form of scripted deployment. They won't be seeing the error

message below and will get a generic error message in their automation tool. For cases like this, please request the customers to attempt the deployment via portal, particularly if you believe the underlying issue is related to the ongoing capacity constraint and you need to fetch the error to document it upon ICM creation.

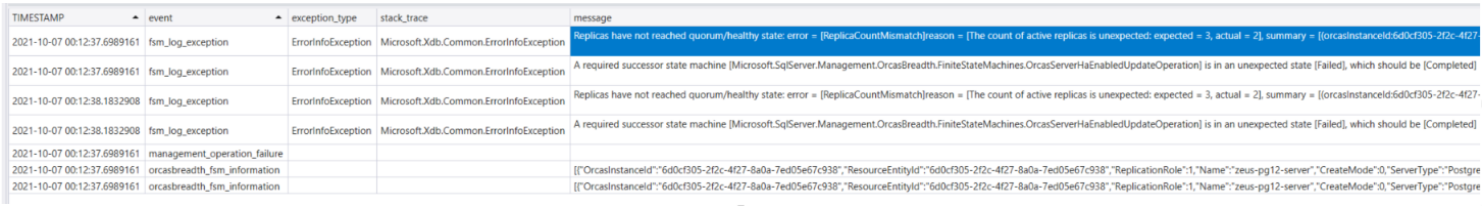




If customers complain that they should be able to deploy or modify resources via automation and the portal is not the most adequate way for them to carry out their devops tasks, please confidently advise them that the capacity constraint is of temporary nature and the fastest way to detect whether the issue is related to any sort of restriction is to test it via portal.

4. Use the below query to find if there are any error messages.

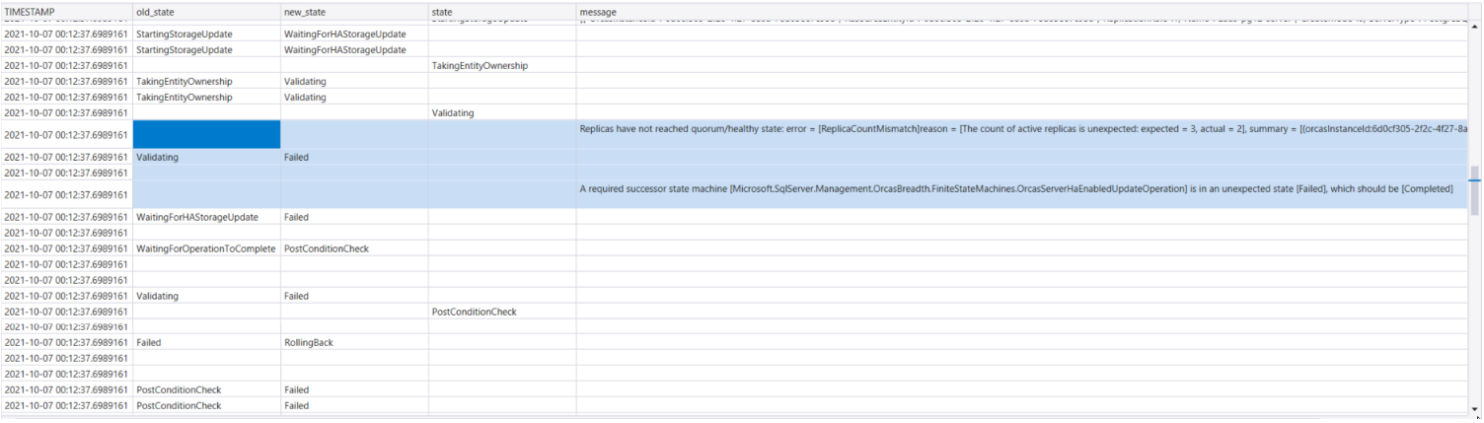
```
//Get the error messages
MonOrcasBreadthRp
| where TIMESTAMP > ago(7d)
//| where TIMESTAMP > datetime(2021-10-04 23:20) and TIMESTAMP < datetime(2021-10-04 23:25)
| where request_id =~ "838f778d-602e-4614-8066-f75487480f7c"
//| where correlation_id =~ '443fbf97-75e9-4cd7-a73e-6a8b5f543d02'
//| where event != "orcasbreadth_fsm_information"
| where event != "orcasbreadth_fsm_information" and (event contains "cancel" or event contains "timeout" or ev
| project TIMESTAMP, event, exception_type, stack_trace, message, operation_type, error_code, error_severity,
```



TIMESTAMP	event	exception_type	stack_trace	message
2021-10-07 00:12:37.6989161	fsm_log_exception	ErrorInfoException	Microsoft.Xdb.Common.ErrorInfoException	Replicas have not reached quorum/healthy state: error = [ReplicaCountMismatch]reason = [The count of active replicas is unexpected: expected = 3, actual = 2], summary = [{orcasInstanceId:6d0cf305-2f2c-4f27-
2021-10-07 00:12:37.6989161	fsm_log_exception	ErrorInfoException	Microsoft.Xdb.Common.ErrorInfoException	A required successor state machine [Microsoft.SqlServer.Management.OrcasBreadth.FiniteStateMachines.OrcasServerHaEnabledUpdateOperation] is in an unexpected state [Failed], which should be [Completed]
2021-10-07 00:12:38.1832908	fsm_log_exception	ErrorInfoException	Microsoft.Xdb.Common.ErrorInfoException	Replicas have not reached quorum/healthy state: error = [ReplicaCountMismatch]reason = [The count of active replicas is unexpected: expected = 3, actual = 2], summary = [{orcasInstanceId:6d0cf305-2f2c-4f27-
2021-10-07 00:12:38.1832908	fsm_log_exception	ErrorInfoException	Microsoft.Xdb.Common.ErrorInfoException	A required successor state machine [Microsoft.SqlServer.Management.OrcasBreadth.FiniteStateMachines.OrcasServerHaEnabledUpdateOperation] is in an unexpected state [Failed], which should be [Completed]
2021-10-07 00:12:37.6989161	management_operation_failure			
2021-10-07 00:12:37.6989161	orcasbreadth_fsm_information			[{"OrcasInstanceId":"6d0cf305-2f2c-4f27-8a0a-7ed05e67c938","ResourceEntityId":"6d0cf305-2f2c-4f27-8a0a-7ed05e67c938","ReplicationRole":1,"Name":"zeus-pg12-server","CreateMode":0,"ServerType":"Postgre
2021-10-07 00:12:37.6989161	orcasbreadth_fsm_information			[{"OrcasInstanceId":"6d0cf305-2f2c-4f27-8a0a-7ed05e67c938","ResourceEntityId":"6d0cf305-2f2c-4f27-8a0a-7ed05e67c938","ReplicationRole":1,"Name":"zeus-pg12-server","CreateMode":0,"ServerType":"Postgre

5. Below Kusto can help to find the steps involved in Provisioning operation and its status.

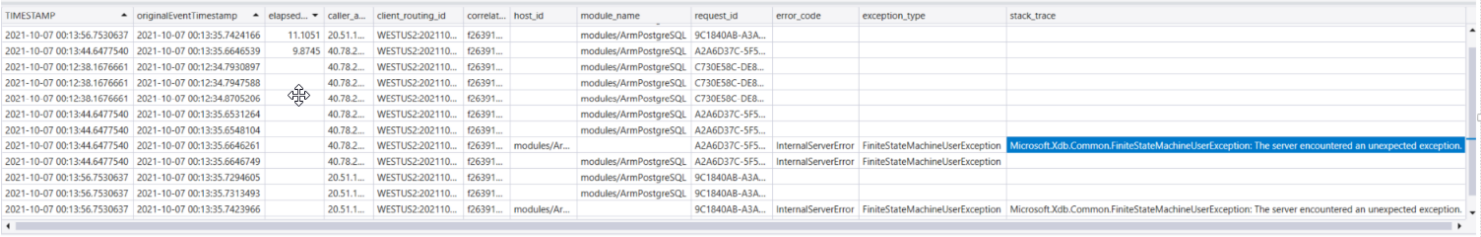
```
//Get the state and Error messages
MonOrcasBreadthRp
| where TIMESTAMP > ago(7d)
//| where TIMESTAMP > datetime(2021-10-04 23:20) and TIMESTAMP < datetime(2021-10-04 23:25)
| where request_id =~ "838f778d-602e-4614-8066-f75487480f7c"
| project TIMESTAMP, old_state, new_state, state,  message, error_message, error_number | order by TIMES
```



TIMESTAMP	old_state	new_state	state	message
2021-10-07 00:12:37.6989161	StartingStorageUpdate	WaitingForHAStorageUpdate		
2021-10-07 00:12:37.6989161	StartingStorageUpdate	WaitingForHAStorageUpdate		
2021-10-07 00:12:37.6989161			TakingEntityOwnership	
2021-10-07 00:12:37.6989161	TakingEntityOwnership	Validating		
2021-10-07 00:12:37.6989161	TakingEntityOwnership	Validating		
2021-10-07 00:12:37.6989161			Validating	
2021-10-07 00:12:37.6989161				Replicas have not reached quorum/healthy state: error = [ReplicaCountMismatch]reason = [The count of active replicas is unexpected: expected = 3, actual = 2], summary = [{orcasInstanceId:6d0cf305-2f2c-4f27-8a
2021-10-07 00:12:37.6989161	Validating	Failed		
2021-10-07 00:12:37.6989161				
2021-10-07 00:12:37.6989161				A required successor state machine [Microsoft.SqlServer.Management.OrcasBreadth.FiniteStateMachines.OrcasServerHaEnabledUpdateOperation] is in an unexpected state [Failed], which should be [Completed]
2021-10-07 00:12:37.6989161	WaitingForHAStorageUpdate	Failed		
2021-10-07 00:12:37.6989161				
2021-10-07 00:12:37.6989161	WaitingForOperationToComplete	PostConditionCheck		
2021-10-07 00:12:37.6989161				
2021-10-07 00:12:37.6989161				
2021-10-07 00:12:37.6989161	Validating	Failed		
2021-10-07 00:12:37.6989161			PostConditionCheck	
2021-10-07 00:12:37.6989161				
2021-10-07 00:12:37.6989161	Failed	RollingBack		
2021-10-07 00:12:37.6989161				
2021-10-07 00:12:37.6989161				
2021-10-07 00:12:37.6989161	PostConditionCheck	Failed		
2021-10-07 00:12:37.6989161	PostConditionCheck	Failed		

6. If you know the region and correlation ID, You can use the below query also to find the error message

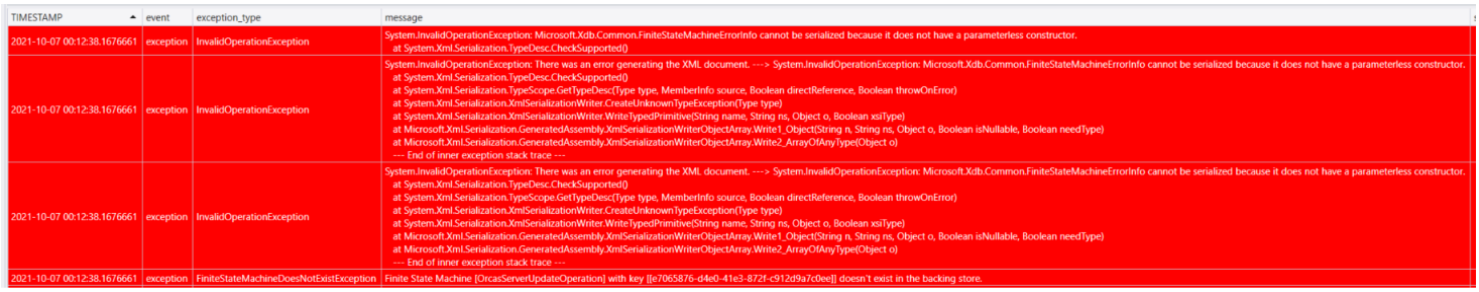
```
//Get the Error messages
MonOrcasBreadthResourceProvider
| where TIMESTAMP > ago(7d)
//| where TIMESTAMP > datetime(2021-10-04 23:20) and TIMESTAMP < datetime(2021-10-04 23:25)
| where correlation_id =~ 'f26391b4-5774-4bee-a594-c3a0b58f64c5'
| project TIMESTAMP, originalEventTimestamp, elapsed_time_milliseconds, caller_address, client_routing_i
```



TIMESTAMP	originalEventTimestamp	elapsed...	caller_a...	client_routing_id	correlat...	host_id	module_name	request_id	error_code	exception_type	stack_trace
2021-10-07 00:13:56.7530637	2021-10-07 00:13:35.7424166	11.1051	20.51.1...	WESTUS2:202110...	f26391...		modules/ArmPostgreSQL	9C1840AB-A3A...			
2021-10-07 00:13:44.6477540	2021-10-07 00:13:35.6646539	9.8745	40.78.2...	WESTUS2:202110...	f26391...		modules/ArmPostgreSQL	A2A6D37C-5F5...			
2021-10-07 00:12:38.1676661	2021-10-07 00:12:34.7930897		40.78.2...	WESTUS2:202110...	f26391...		modules/ArmPostgreSQL	C730E58C-DE8...			
2021-10-07 00:12:38.1676661	2021-10-07 00:12:34.7947588		40.78.2...	WESTUS2:202110...	f26391...		modules/ArmPostgreSQL	C730E58C-DE8...			
2021-10-07 00:12:38.1676661	2021-10-07 00:12:34.8705206		40.78.2...	WESTUS2:202110...	f26391...		modules/ArmPostgreSQL	C730E58C-DE8...			
2021-10-07 00:13:44.6477540	2021-10-07 00:13:35.6531264		40.78.2...	WESTUS2:202110...	f26391...		modules/ArmPostgreSQL	A2A6D37C-5F5...			
2021-10-07 00:13:44.6477540	2021-10-07 00:13:35.6548104		40.78.2...	WESTUS2:202110...	f26391...		modules/ArmPostgreSQL	A2A6D37C-5F5...			
2021-10-07 00:13:44.6477540	2021-10-07 00:13:35.6646261		40.78.2...	WESTUS2:202110...	f26391...	modules/Ar...		A2A6D37C-5F5...	InternalServerError	FiniteStateMachineUserException	Microsoft.Xdb.Common.FiniteStateMachineUserException: The server encountered an unexpected exception.
2021-10-07 00:13:44.6477540	2021-10-07 00:13:35.6646749		40.78.2...	WESTUS2:202110...	f26391...		modules/ArmPostgreSQL	A2A6D37C-5F5...	InternalServerError	FiniteStateMachineUserException	
2021-10-07 00:13:56.7530637	2021-10-07 00:13:35.7294605		20.51.1...	WESTUS2:202110...	f26391...		modules/ArmPostgreSQL	9C1840AB-A3A...			
2021-10-07 00:13:56.7530637	2021-10-07 00:13:35.7313493		20.51.1...	WESTUS2:202110...	f26391...		modules/ArmPostgreSQL	9C1840AB-A3A...			
2021-10-07 00:13:56.7530637	2021-10-07 00:13:35.7423966		20.51.1...	WESTUS2:202110...	f26391...	modules/Ar...		9C1840AB-A3A...	InternalServerError	FiniteStateMachineUserException	Microsoft.Xdb.Common.FiniteStateMachineUserException: The server encountered an unexpected exception.

7. Using the query below you can find the exception messages. This query is helpful when investigating Capacity constraint issues.

```
//Get the Error messages
MonOrcasBreadthRpExceptions
| where TIMESTAMP > ago(7d)
//| where TIMESTAMP > datetime(2021-10-04 23:20) and TIMESTAMP < datetime(2021-10-04 23:25)
| where request_id =~ "838f778d-602e-4614-8066-f75487480f7c"
//| where message !contains "Finite State Machine [OrcasResourceSpecificationCreateOperation] with key"
| project TIMESTAMP, event, exception_type, message, stack_trace, level
```



TIMESTAMP	event	exception_type	message
2021-10-07 00:12:38.1676661	exception	InvalidOperationException	System.InvalidOperationException: Microsoft.Xdb.Common.FiniteStateMachineErrorInfo cannot be serialized because it does not have a parameterless constructor. at System.Xml.Serialization.TypeDesc.CheckSupported()
2021-10-07 00:12:38.1676661	exception	InvalidOperationException	System.InvalidOperationException: There was an error generating the XML document. ---> System.InvalidOperationException: Microsoft.Xdb.Common.FiniteStateMachineErrorInfo cannot be serialized because it does not have a parameterless constructor. at System.Xml.Serialization.TypeDesc.CheckSupported() at System.Xml.Serialization.TypeScope.GetTypeDesc(Type type, MemberInfo source, Boolean directReference, Boolean throwOnError) at System.Xml.Serialization.XmlSerializationWriter.CreateUnknownTypeException(Type type) at System.Xml.Serialization.XmlSerializationWriter.WriteTypedPrimitive(String name, String ns, Object o, Boolean xsiType) at Microsoft.Xml.Serialization.GeneratedAssembly.XmlSerializationWriterObjectArray.Write1_Object(String n, String ns, Object o, Boolean isNullable, Boolean needType) at Microsoft.Xml.Serialization.GeneratedAssembly.XmlSerializationWriterObjectArray.Write2_ArrayOfAnyType(Object o) --- End of inner exception stack trace ---
2021-10-07 00:12:38.1676661	exception	InvalidOperationException	System.InvalidOperationException: There was an error generating the XML document. ---> System.InvalidOperationException: Microsoft.Xdb.Common.FiniteStateMachineErrorInfo cannot be serialized because it does not have a parameterless constructor. at System.Xml.Serialization.TypeDesc.CheckSupported() at System.Xml.Serialization.TypeScope.GetTypeDesc(Type type, MemberInfo source, Boolean directReference, Boolean throwOnError) at System.Xml.Serialization.XmlSerializationWriter.CreateUnknownTypeException(Type type) at System.Xml.Serialization.XmlSerializationWriter.WriteTypedPrimitive(String name, String ns, Object o, Boolean xsiType) at Microsoft.Xml.Serialization.GeneratedAssembly.XmlSerializationWriterObjectArray.Write1_Object(String n, String ns, Object o, Boolean isNullable, Boolean needType) at Microsoft.Xml.Serialization.GeneratedAssembly.XmlSerializationWriterObjectArray.Write2_ArrayOfAnyType(Object o) --- End of inner exception stack trace ---
2021-10-07 00:12:38.1676661	exception	FiniteStateMachineDoesNotExistException	Finite State Machine [OrcasServerUpdateOperation] with key [[e7065876-d4e0-41e3-872f-c912d9a7c0ee]] doesn't exist in the backing store.

Mitigation

Azure Database for PostgreSQL Flexible server provision can be due to different issues. Review the Error message and act accordingly. If you are unable to resolve the error message, Please file an ICM.

Public Doc Reference

https://docs.microsoft.com/en-us/azure/postgresql/flexible-server/quickstart-create-server-portal

Root Cause Classification

CRUD PostgreSQL Flexible Server/CRUD