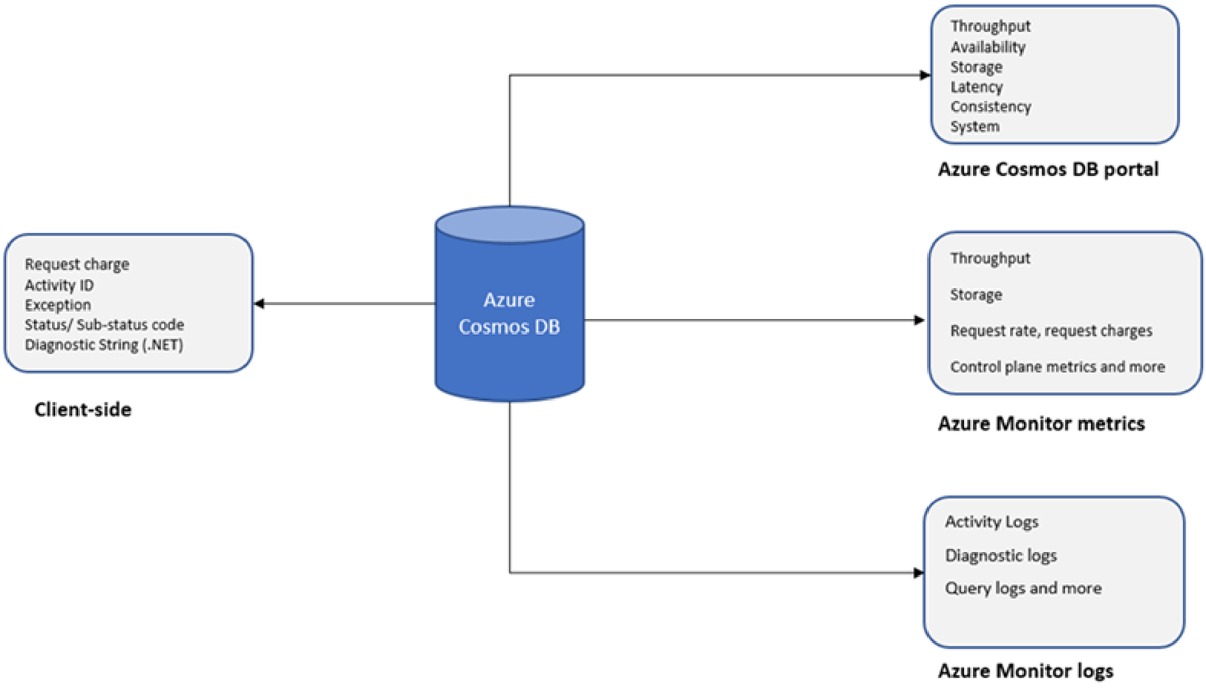
Module 11: Monitor and troubleshoot an Azure Cosmos DB SQL API solution

Measure performance in Azure Cosmos DB SQL API

Understand Azure Monitor

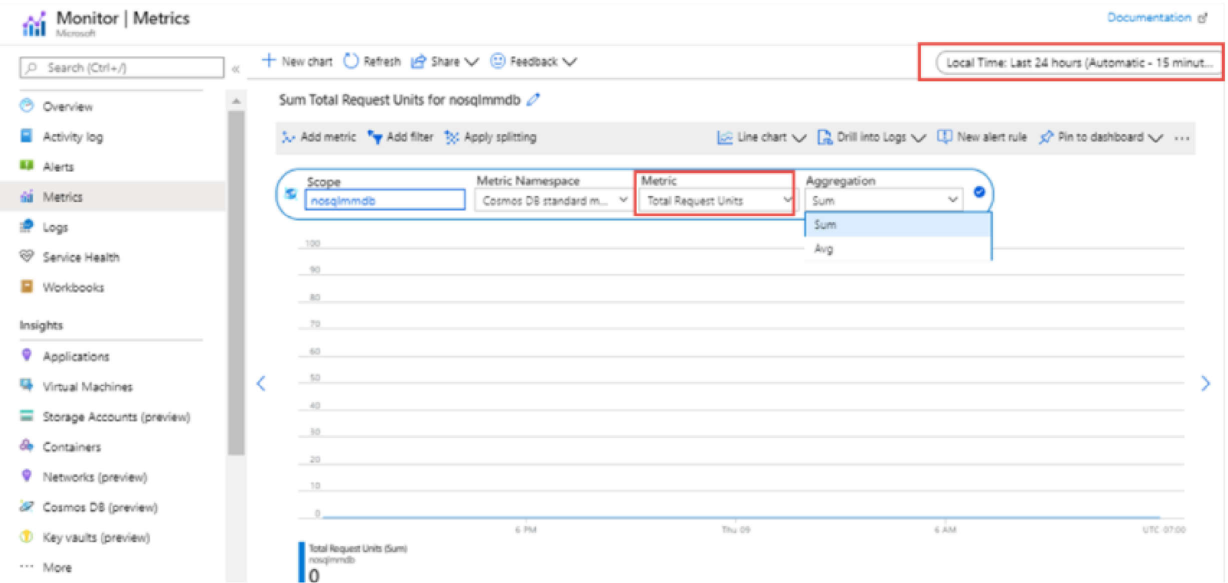
Azure Monitor is used to monitor the Azure resource availability, performance, and operations metrics.



Measure throughput

The Total Request Units metric can then be used to analyze those operations with the highest throughput.

View the Total Request Unit metrics:

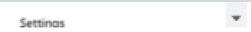


Local Time: Last 24 hours (Automatic - 15 minut...

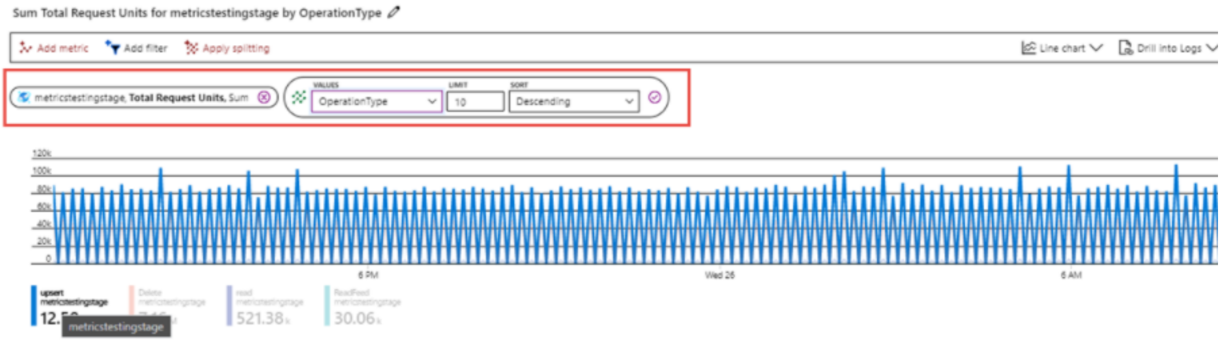
Sum Total Request Units for nosqlmmdb

Scope: nosqlmmdb
Metric Namespace: Cosmos DB standard m...
Metric: Total Request Units
Aggregation: Sum

Total Request Units (Sum)
nosqlmmdb
0



Filter the Total Request Units further:


Sum Total Request Units for metricstestingstage by OperationType
Add metric
Add filter
Apply splitting
Line chart
Drill into Logs
metricstestingstage, Total Request Units, Sum
120k
100k
80k
60k
40k
20k
0
6 PM
Wed 26
6 AM
upsert metricstestingstage
12.5
Delete metricstestingstage
read metricstestingstage
521.38
ReadFeed metricstestingstage
30.06
metricstestingstage

VALUES: OperationType

LIMIT: 10

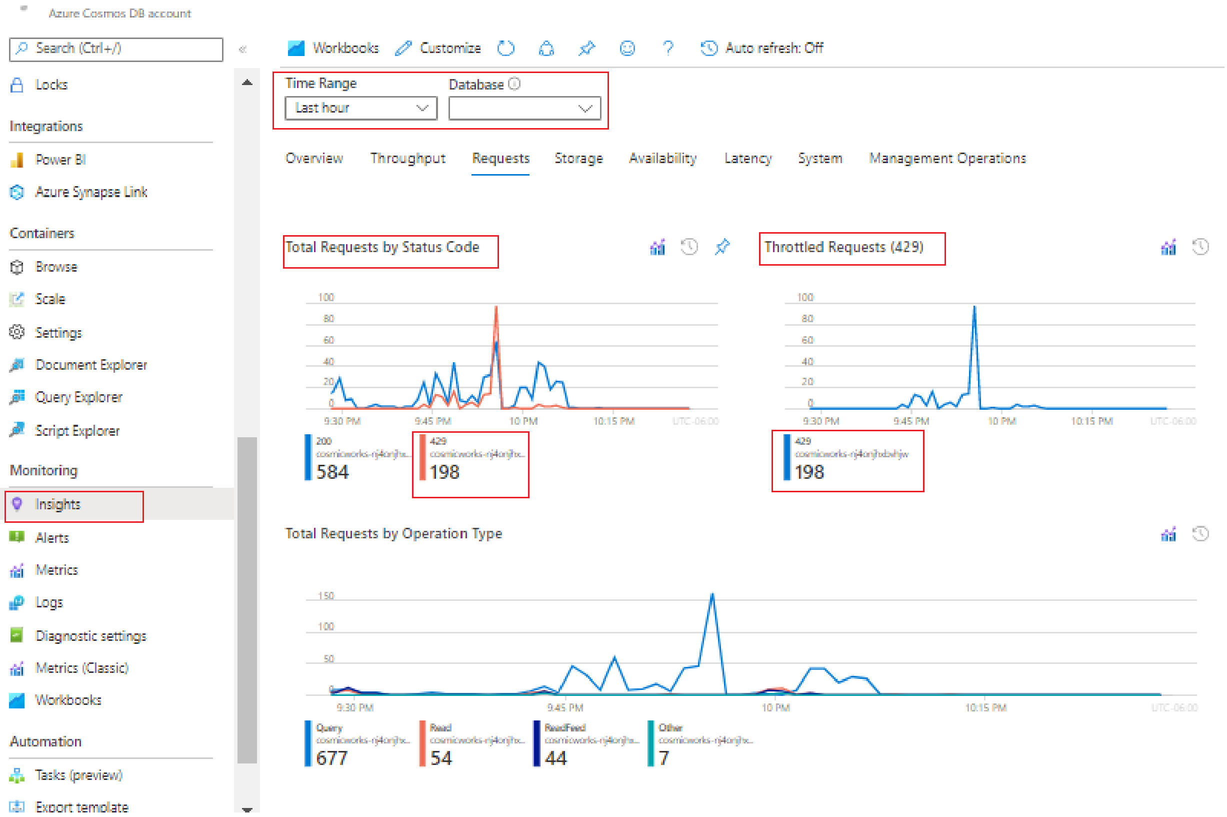
SORT: Descending

Observe rate-limiting events

There are three main reasons why we get a 429 exception:

- Request rate is large.
- The request did not complete due to a high rate of metadata requests.
- The request did not complete due to a transient service error.

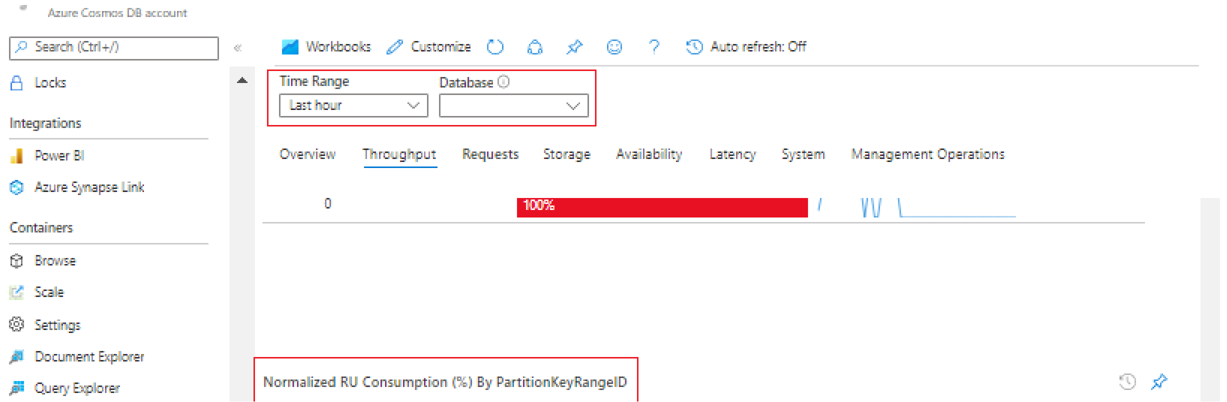
Review the Insights-Request charts for 429s



Time Range: Last hour

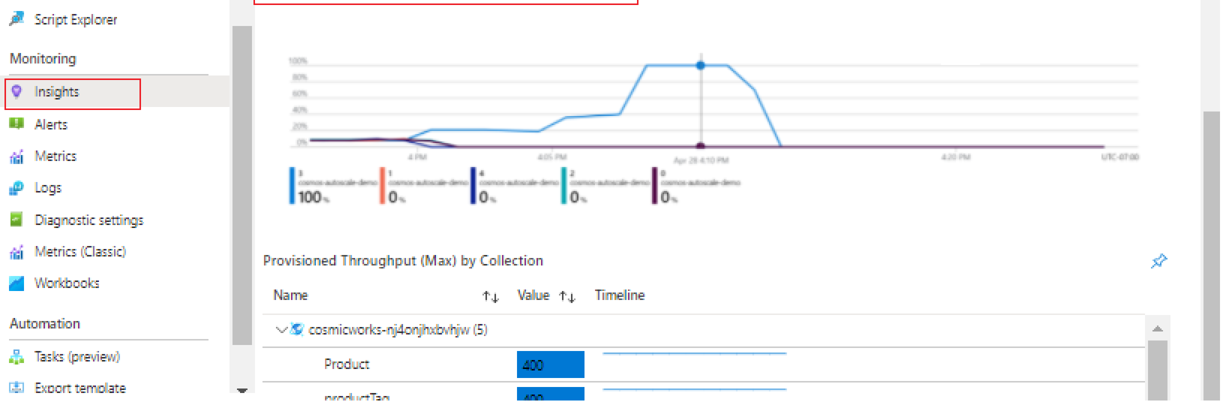
Database:

Review the Insights-Request charts for hot partitions



Time Range: Last hour

Database:



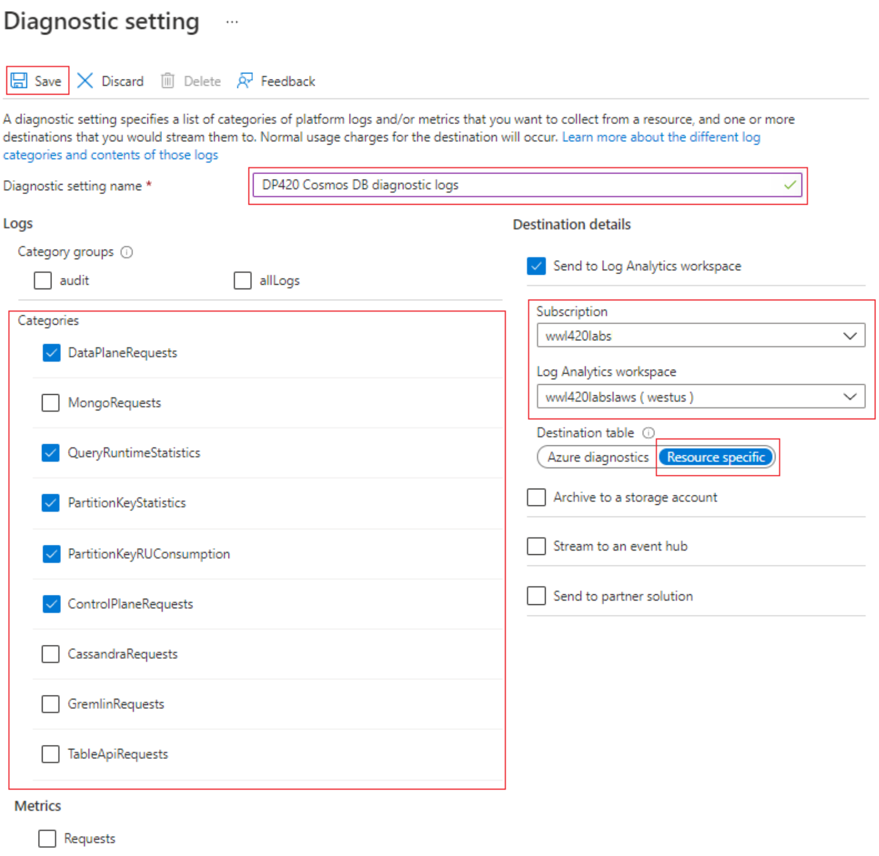
Provisioned Throughput (Max) by Collection

Name ↑↓	Value ↑↓	Timeline
cosmicworks-nj4onjhxbvhjw (5)		
Product	400	
productTag	400	

Query logs

Diagnostics settings are used to collect Azure Diagnostic Logs produced by Azure resources. These logs provide detailed resource operational data.

Create Azure Cosmos DB diagnostics settings



Diagnostic setting ...

Save Discard Delete Feedback

A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a resource, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. Learn more about the different log categories and contents of those logs

Diagnostic setting name * DP420 Cosmos DB diagnostic logs

Logs

Category groups

□ audit □ allLogs

Categories

☑ DataPlaneRequests
□ MongoRequests
☑ QueryRuntimeStatistics
☑ PartitionKeyStatistics
☑ PartitionKeyRUConsumption
☑ ControlPlaneRequests
□ CassandraRequests
□ GremlinRequests
□ TableApiRequests

Metrics

□ Requests

Destination details

☑ Send to Log Analytics workspace

Subscription: wwl420labs

Log Analytics workspace: wwl420labslaws (westus)

Destination table: Azure diagnostics [✓ Resource specific]

□ Archive to a storage account
□ Stream to an event hub
□ Send to partner solution

Query that returns the count and the total request charged of the different Azure Cosmos DB operation types in the last hour.

```
kusto
AzureDiagnostics
| where TimeGenerated >= ago(1h)
| where ResourceProvider=="MICROSOFT.DOCUMENTDB" and
Category=="DataPlaneRequests"
| summarize OperationCount = count(),
TotalRequestCharged=sum(todouble(requestCharge_s)) by OperationName
| order by TotalRequestCharged desc


CDBDataPlaneRequests
| where TimeGenerated >= ago(1h)
| summarize OperationCount = count(),
TotalRequestCharged=sum(todouble(RequestCharge)) by OperationName
| order by TotalRequestCharged desc
| order by TotalRequestCharged desc
```

Create a query that returns a timechart graph for all successful (status 200) and rate limited (status 429) request in the last hour. The requests will be aggregated every 10 minutes.

```
kusto
AzureDiagnostics
| where TimeGenerated >= ago(1h)
| where ResourceProvider=="MICROSOFT.DOCUMENTDB" and
Category=="DataPlaneRequests"
| summarize requestcount=count() by statusCode_s, bin(TimeGenerated, 10m)
| render timechart


CDBDataPlaneRequests
| where TimeGenerated >= ago(2h)
| summarize requestcount=count() by StatusCode, bin(TimeGenerated, 10m)
| render timechart
```

Monitor responses and events in Azure Cosmos DB SQL API

Review common response codes

Status Code	Name	Description
200	OK	List, Get, Replace, Patch, Query -> The operation was successful.
201	Created	The operation was successful.
204	No Content	The delete operation was successful.
304	Not Modified	The document requested wasn't modified since the specified eTag value in the If-Match header. The service returns an empty response body.
400	Bad Request	The JSON body is invalid. Check for missing curly brackets or quotes.
403	Forbidden	The operation couldn't be completed because the storage limit of the partition has been reached.
404	Not Found	The document no longer exists, that is, the document was deleted.
408	Request timeout	The operation did not complete within the allotted amount of time. This code is returned when a stored procedure, trigger, or UDF (within a query) does not complete execution within the maximum execution time.
409	Conflict	The `id` provided for the new document has been taken by an existing document.
413	Entity Too Large	The document size in the request exceeded the allowable document size in a request.
429	Too many requests	The collection has exceeded the provisioned throughput limit. Retry the request after the server specified retry after duration. For more information, see request units.
500	Internal Server Error	The operation failed because of an unexpected service error. Contact support.
503	Service Unavailable	The operation couldn't be completed because the service was unavailable. This situation could happen because of network connectivity or service availability issues. It's safe to retry the operation. If the issue persists, contact support.

Understand transient errors

We can identify and troubleshoot Azure Cosmos DB service unavailable exceptions when our request returns status code 503.

- Required ports are blocked: Verify that the following ports are enabled for the SQL API.

Connection mode	Supported protocol	Supported SDKs	API/Service port
Gateway	HTTPS	All SDKs	SQL (443)
Direct	TCP	.NET SDK, Java SDK	When using public/service endpoints: ports in the 10000 through 20000 range. When using private endpoints: ports in the 0 through 65535 range

- Client-side transient connectivity issues

```
TransportException: A client transport error occurred: The request timed out
while waiting for a server response.
(Time: xxx, activity ID: xxx, error code: ReceiveTimeout [0x0010], base error:
HRESULT 0x80131500
```

- Service Outage: Check the Azure status page to see if there's an ongoing issue.

Review rate limiting errors

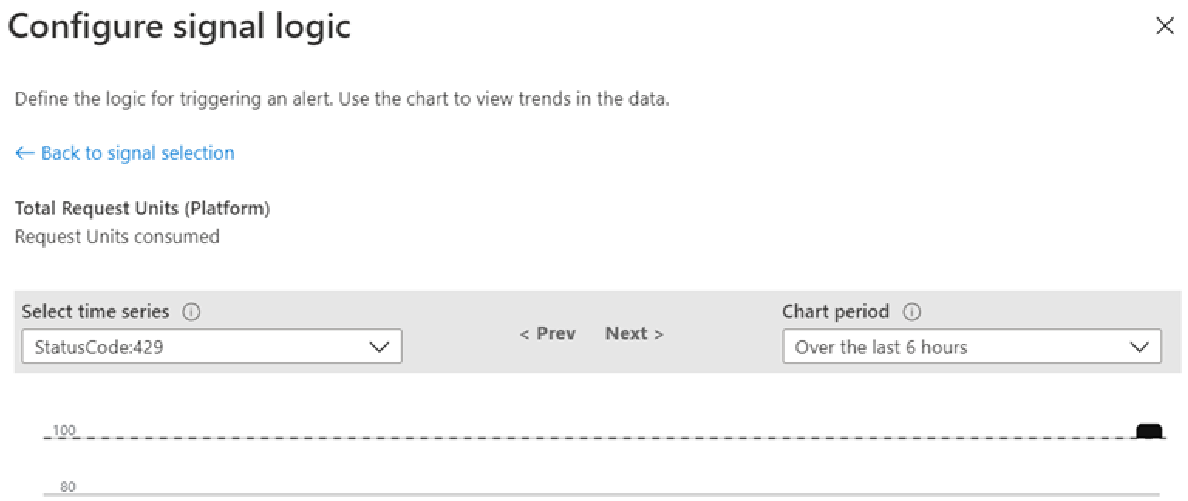
Requests return status code 429 for the exception request rate too large status code, indicating that your requests against Azure Cosmos DB are being rate-limited.

```
kusto
AzureDiagnostics
| where TimeGenerated >= ago(24h)
| where Category == "DataPlaneRequests"
| summarize throttledOperations = dcountif(activityId_g, statusCode_s ==
429), totalOperations = dcount(activityId_g), totalConsumedRUPerMinute =
sum(todouble(requestCharge_s)) by databaseName_s, collectionName_s,
OperationName, requestResourceType_s, bin(TimeGenerated, 1min)
| extend averageRUPerOperation = 1.0 * totalConsumedRUPerMinute /
totalOperations
| extend fractionOf429s = 1.0 * throttledOperations / totalOperations
| order by fractionOf429s desc
```

Rate-limiting due to transient service error: Retrying the request is the only recommended solution.

Configure Alerts

Azure Cosmos DB uses the Azure Monitor Service to set up and send alerts.



Configure signal logic

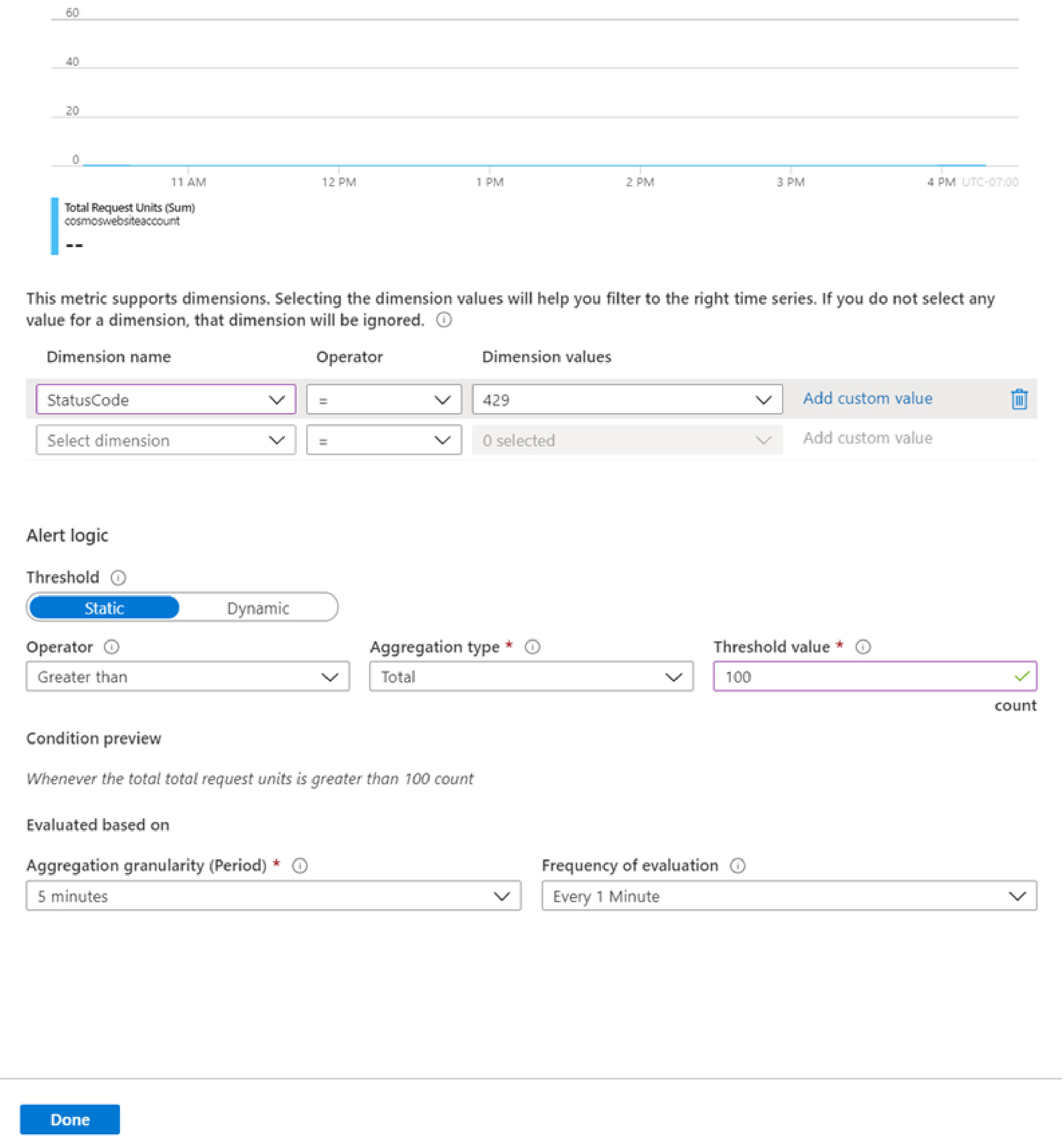
Define the logic for triggering an alert. Use the chart to view trends in the data.

← Back to signal selection

Total Request Units (Platform)
Request Units consumed

Select time series: StatusCode:429

Chart period: Over the last 6 hours



This metric supports dimensions. Selecting the dimension values will help you filter to the right time series. If you do not select any value for a dimension, that dimension will be ignored. ⓘ

Dimension name	Operator	Dimension values		
StatusCode	=	429	Add custom value	🗑
Select dimension	=	0 selected	Add custom value	

Alert logic

Threshold ⓘ

● Static ○ Dynamic

Operator ⓘ: Greater than

Aggregation type * ⓘ: Total

Threshold value * ⓘ: 100 ✓ count

Condition preview

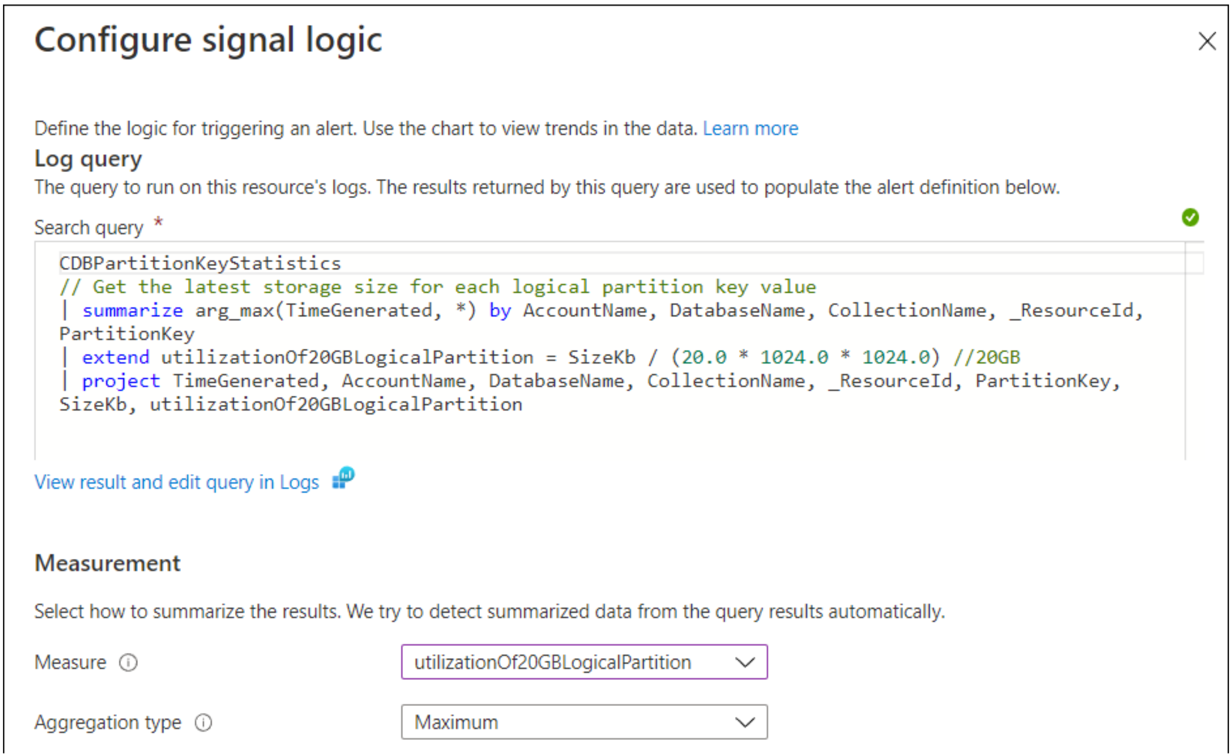
Whenever the total total request units is greater than 100 count

Evaluated based on

Aggregation granularity (Period) * ⓘ: 5 minutes

Frequency of evaluation ⓘ: Every 1 Minute

Here's an example of an alert that will trigger if the storage for a logical partition key exceeds 70% of the 20 GB limit (has more than 14 GB of storage)



Configure signal logic

Define the logic for triggering an alert. Use the chart to view trends in the data. Learn more

Log query

The query to run on this resource's logs. The results returned by this query are used to populate the alert definition below.

Search query *

```
CDBPartitionKeyStatistics
// Get the latest storage size for each logical partition key value
| summarize arg_max(TimeGenerated, *) by AccountName, DatabaseName, CollectionName, _ResourceId,
PartitionKey
| extend utilizationOf20GBLogicalPartition = SizeKb / (20.0 * 1024.0 * 1024.0) //20GB
| project TimeGenerated, AccountName, DatabaseName, CollectionName, _ResourceId, PartitionKey,
SizeKb, utilizationOf20GBLogicalPartition
```

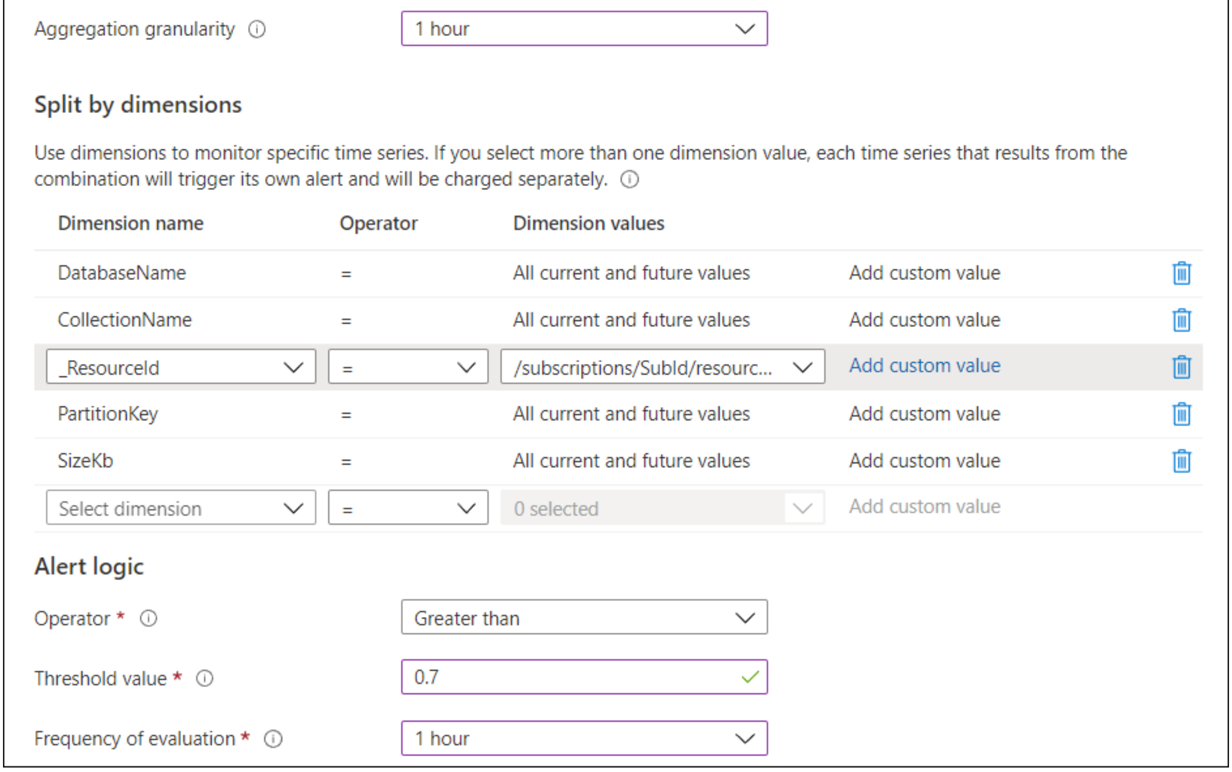
View result and edit query in Logs

Measurement

Select how to summarize the results. We try to detect summarized data from the query results automatically.

Measure ⓘ: utilizationOf20GBLogicalPartition

Aggregation type ⓘ: Maximum



Aggregation granularity ⓘ 1 hour

Split by dimensions

Use dimensions to monitor specific time series. If you select more than one dimension value, each time series that results from the combination will trigger its own alert and will be charged separately. ⓘ

Dimension name	Operator	Dimension values		
DatabaseName	=	All current and future values	Add custom value	🗑
CollectionName	=	All current and future values	Add custom value	🗑
_ResourceId	=	/subscriptions/SubId/resourc...	Add custom value	🗑
PartitionKey	=	All current and future values	Add custom value	🗑
SizeKb	=	All current and future values	Add custom value	🗑
Select dimension	=	0 selected	Add custom value	

Alert logic

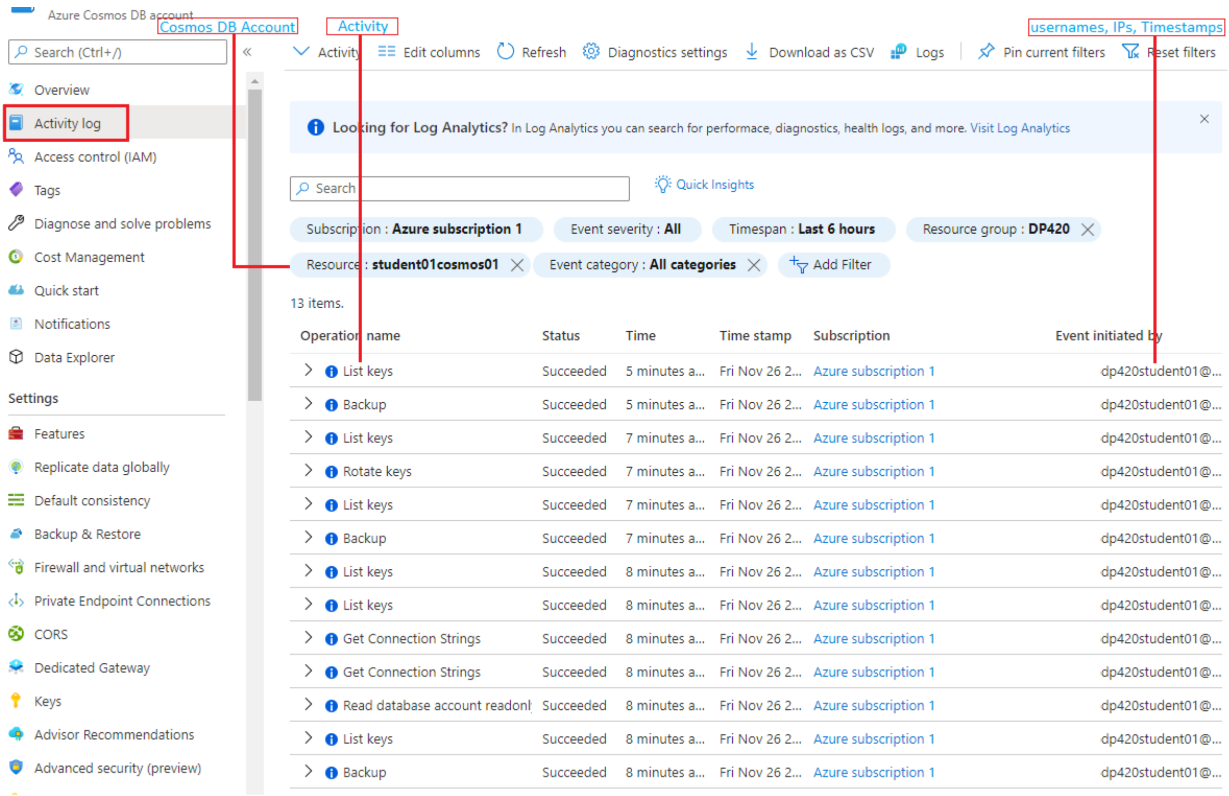
Operator * ⓘ Greater than

Threshold value * ⓘ 0.7

Frequency of evaluation * ⓘ 1 hour

Audit security

Activity logs, which are automatically available, contain all write operations (PUT, POST, DELETE) for your Cosmos DB resources except read operations (GET).



Looking for Log Analytics? In Log Analytics you can search for performance, diagnostics, health logs, and more. Visit Log Analytics

Subscription : **Azure subscription 1** Event severity : **All** Timespan : **Last 6 hours** Resource group : **DP420**

Resource : **student01cosmos01** Event category : **All categories**

13 items.

Operation name	Status	Time	Time stamp	Subscription	Event initiated by
List keys	Succeeded	5 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
Backup	Succeeded	5 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
List keys	Succeeded	7 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
Rotate keys	Succeeded	7 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
List keys	Succeeded	7 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
Backup	Succeeded	7 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
List keys	Succeeded	8 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
List keys	Succeeded	8 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
Get Connection Strings	Succeeded	8 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
Get Connection Strings	Succeeded	8 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
Read database account readonl	Succeeded	8 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
List keys	Succeeded	8 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...
Backup	Succeeded	8 minutes a...	Fri Nov 26 2...	Azure subscription 1	dp420student01@...

Implementing backup and restore for Azure Cosmos DB SQL API

Evaluate periodic backup

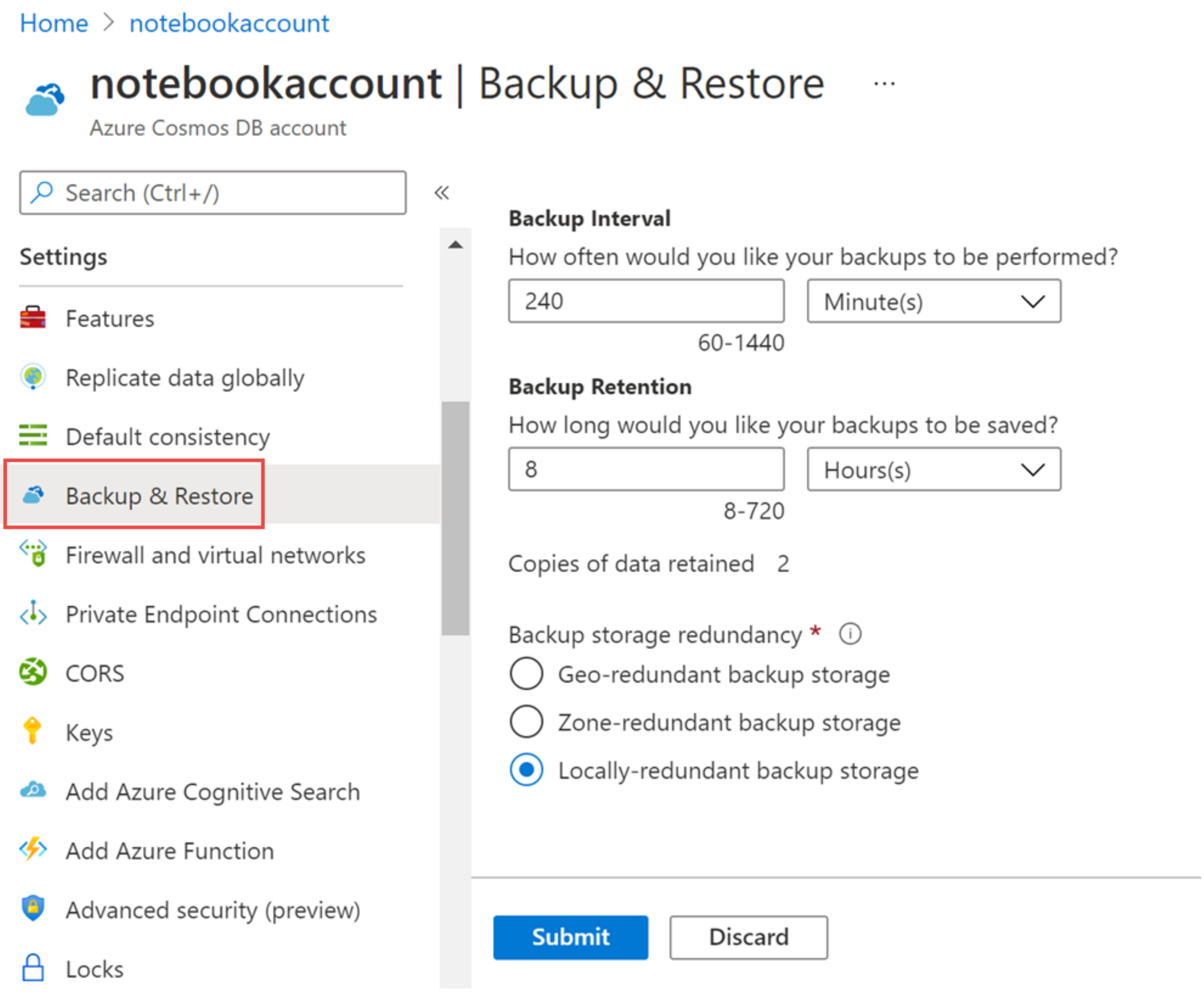
Azure Cosmos DB takes automatic backups of your data at regular periodic intervals.

Backup Storage Redundancy

- Geo-redundant
- Zone-redundant
- Locally redundant

Change the default backup interval and retention period

- Backup Interval
- Backup Retention
- Backup storage redundancy


Home > notebookaccount

notebookaccount | Backup & Restore

Azure Cosmos DB account

Backup Interval

How often would you like your backups to be performed?

240 Minute(s)

60-1440

Backup Retention

How long would you like your backups to be saved?

8 Hours(s)

8-720

Copies of data retained 2

Backup storage redundancy *

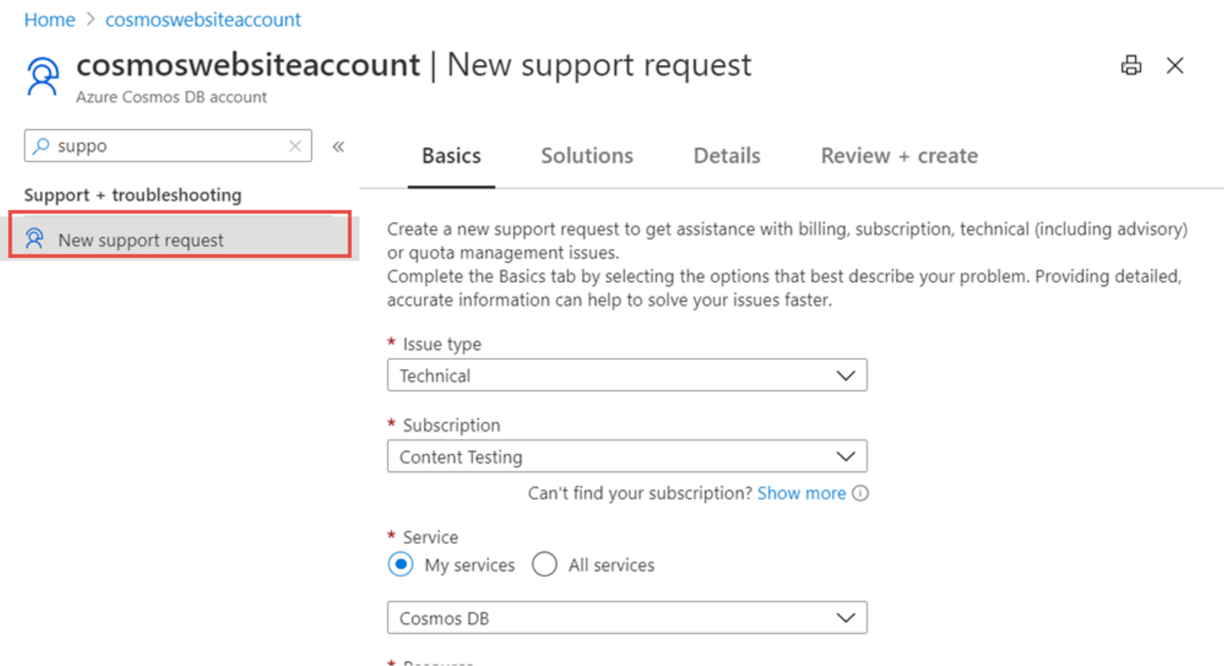
○ Geo-redundant backup storage

○ Zone-redundant backup storage

● Locally-redundant backup storage

To request to restore a backup

- Open a request ticket or call the Azure support team.


Home > cosmoswebsiteaccount

cosmoswebsiteaccount | New support request

Azure Cosmos DB account

Create a new support request to get assistance with billing, subscription, technical (including advisory) or quota management issues.
Complete the Basics tab by selecting the options that best describe your problem. Providing detailed, accurate information can help to solve your issues faster.

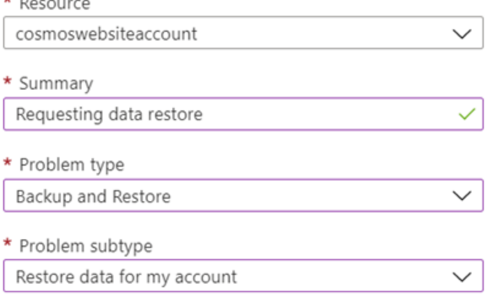
* Issue type: Technical

* Subscription: Content Testing

Can't find your subscription? Show more

* Service: ● My services ○ All services

Cosmos DB



* Resource

cosmoswebsiteaccount

* Summary

Requesting data restore

* Problem type

Backup and Restore

* Problem subtype

Restore data for my account

Consider restoring a backup when you...

- Delete the entire Azure Cosmos DB account.
- Delete one or more Azure Cosmos DB databases.
- Delete one or more Azure Cosmos DB containers.
- Delete or modify the Azure Cosmos DB items within a container. This specific case is typically - referred to as data corruption.

Costs of Extra backups:

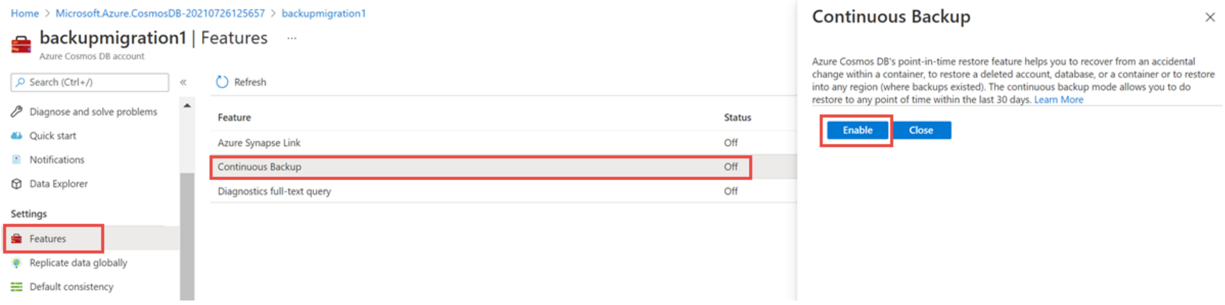
- Two backups included with the account for free.
- Extra backups will be charged on a region-based backup-storing pricing.

Manage your own backups:

- Azure Data Factory
- Change feed

Configure continuous backup and recovery

When using the continuous backups mode, backups are continuously taken in every region where the Azure Cosmos DB account exists.



Feature	Status
Azure Synapse Link	Off
Continuous Backup	Off
Diagnostics full-text query	Off

Continuous Backup

Azure Cosmos DB's point-in-time restore feature helps you to recover from an accidental change within a container, to restore a deleted account, database, or a container or to restore into any region (where backups existed). The continuous backup mode allows you to do restore to any point of time within the last 30 days. Learn More

Backup Storage Redundancy

- Locally redundant by default
- Zone-redundant when using Availability zones

Change backup options

- Only option is to enable Continuous Backups
- Once set on a new or existing account can not be changed

Continuous backup mode charges

- Backup storage space
- A separate charge will be added every time a restore is started.

Limitations when using the continuous backup mode

- Azure Cosmos DB accounts using customer-managed keys are not supported.

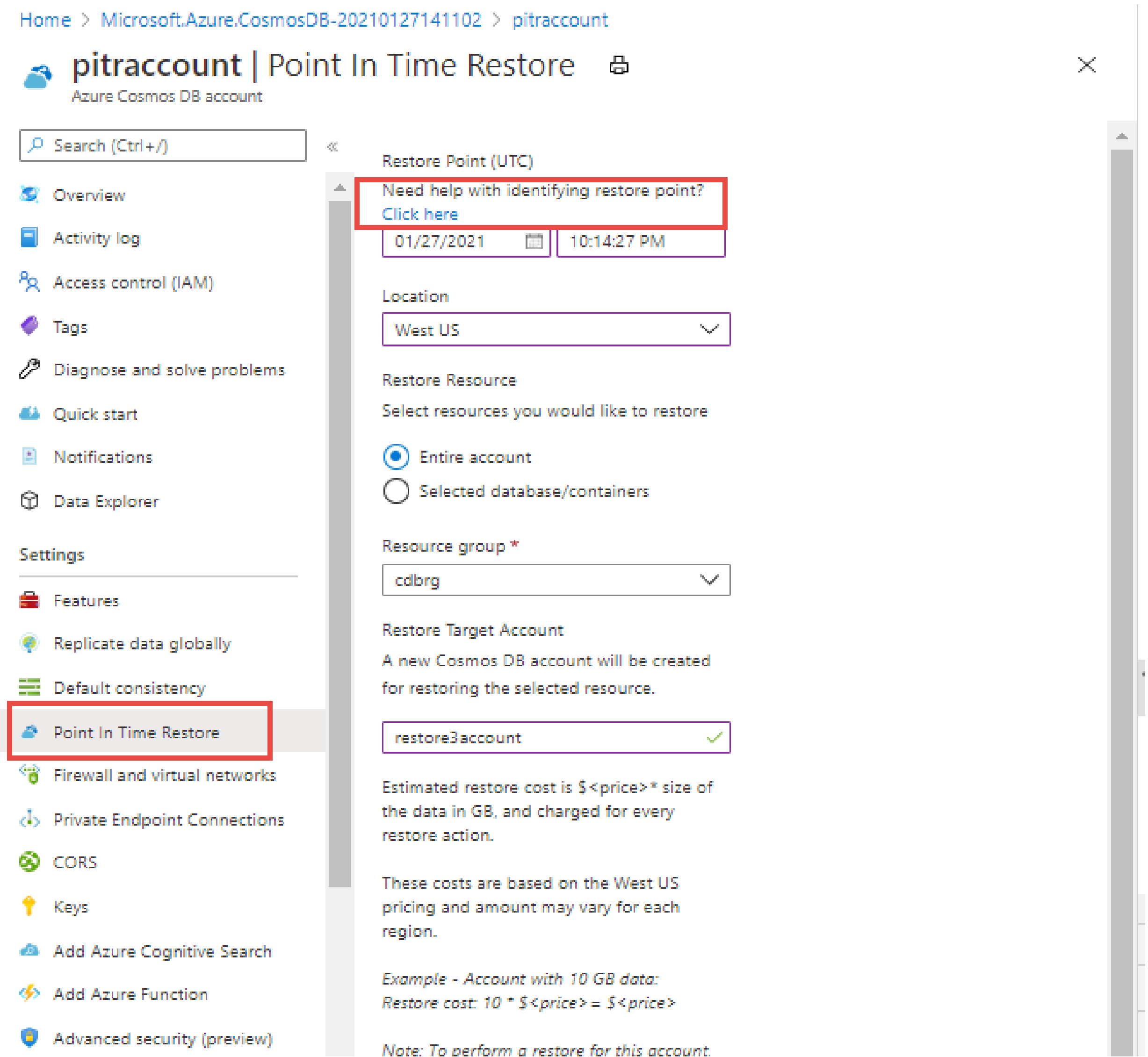
- Multi-region write accounts not supported.
- You can't restore an account into a region where the source account did not exist.
- The retention period is 30 days and can't be changed.
- Can't modify or delete IAM policies when restore is in progress.
- Accounts that create unique indexes after the container is created are not supported.
- Point in time restore always restores to a new Azure Cosmos DB account.
- Collection's consistent indexes may still be rebuilding after completing the restore.
- Since TTL container properties are restored with the restore process, restores must be for timestamps before TTL properties were added to a container. This timestamp will prevent data from being deleted right after the restore.
- Azure Synapse Link and continuous backup mode can't coexist in the same database account.

Perform a point-in-time recovery

Point-in-time recovery will allow you to choose any timestamp within the up to 30-days backup retention period and restore a combination of Azure DB containers, databases, or the accounts.

Scenarios:

- Restore deleted account
- Restore data of an account in a particular region
- Recover from an accidental write or delete operation within a container with a known restore timestamp
- Restore an account to a previous point in time before the accidental delete of the database
- Restore an account to a previous point in time before the accidental delete or modification of the container properties


Home > Microsoft.Azure.CosmosDB-20210127141102 > pitraccount
pitraccount | Point In Time Restore
Azure Cosmos DB account
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Quick start
Notifications
Data Explorer
Settings
Features
Replicate data globally
Default consistency
Point In Time Restore
Firewall and virtual networks
Private Endpoint Connections
CORS
Keys
Add Azure Cognitive Search
Add Azure Function
Advanced security (preview)

Restore Point (UTC)

Need help with identifying restore point? Click here

01/27/2021 10:14:27 PM

Location: West US

Restore Resource

Select resources you would like to restore

● Entire account

○ Selected database/containers

Resource group *: cdbrg

Restore Target Account

A new Cosmos DB account will be created for restoring the selected resource.

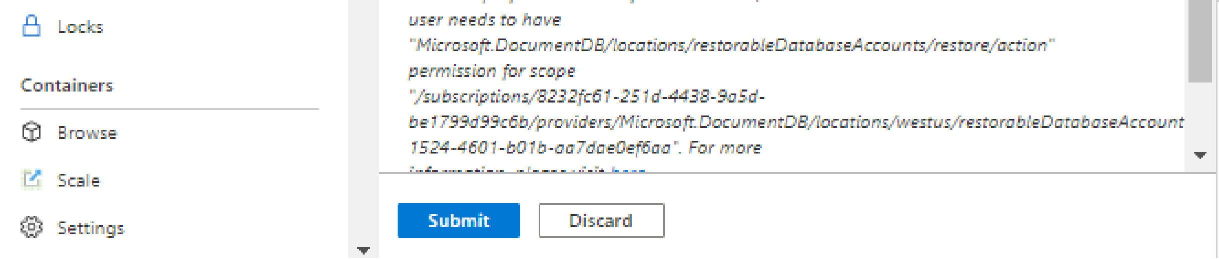
restore3account

Estimated restore cost is $<price>* size of the data in GB, and charged for every restore action.

These costs are based on the West US pricing and amount may vary for each region.

Example - Account with 10 GB data:
*Restore cost: 10 * $<price> = $<price>*

Note: To perform a restore for this account,



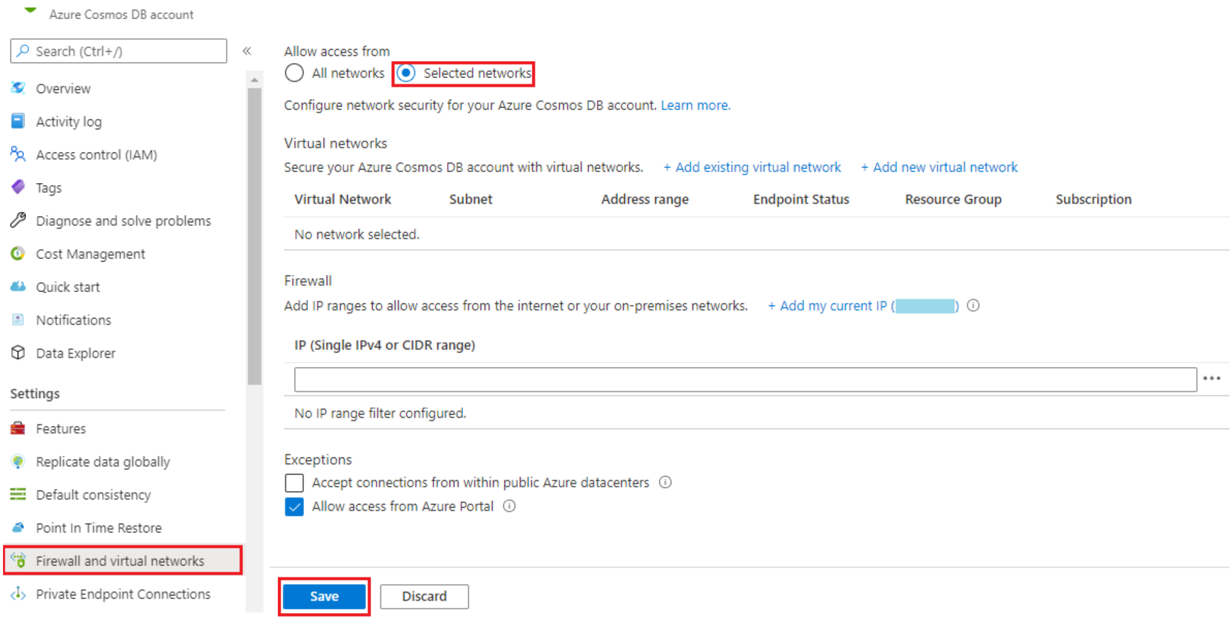
user needs to have "Microsoft.DocumentDB/locations/restorableDatabaseAccounts/restore/action" permission for scope "/subscriptions/8232fc61-251d-4438-9a5d-be1799d99c6b/providers/Microsoft.DocumentDB/locations/westus/restorableDatabaseAccount 1524-4601-b01b-aa7dae0ef6aa". For more

Implement security in Azure Cosmos DB SQL API

Implement network-level access control

Azure Cosmos DB supports IP-based access controls for inbound firewall support.

Configure an IP firewall by using the Azure portal



Allow access from
○ All networks ● Selected networks

Configure network security for your Azure Cosmos DB account. Learn more.

Virtual networks

Secure your Azure Cosmos DB account with virtual networks. + Add existing virtual network + Add new virtual network

Virtual Network	Subnet	Address range	Endpoint Status	Resource Group	Subscription
No network selected.					

Firewall

Add IP ranges to allow access from the internet or your on-premises networks. + Add my current IP ([redacted])

IP (Single IPv4 or CIDR range)

No IP range filter configured.

Exceptions

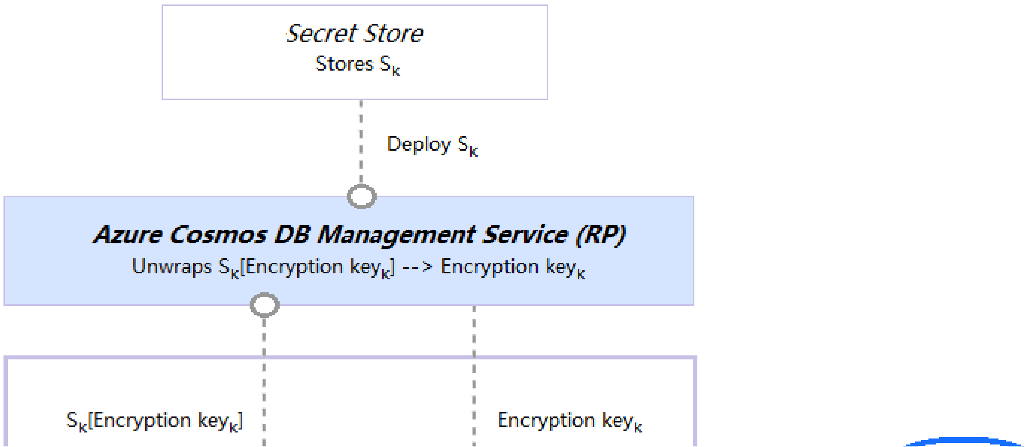
☐ Accept connections from within public Azure datacenters

☑ Allow access from Azure Portal

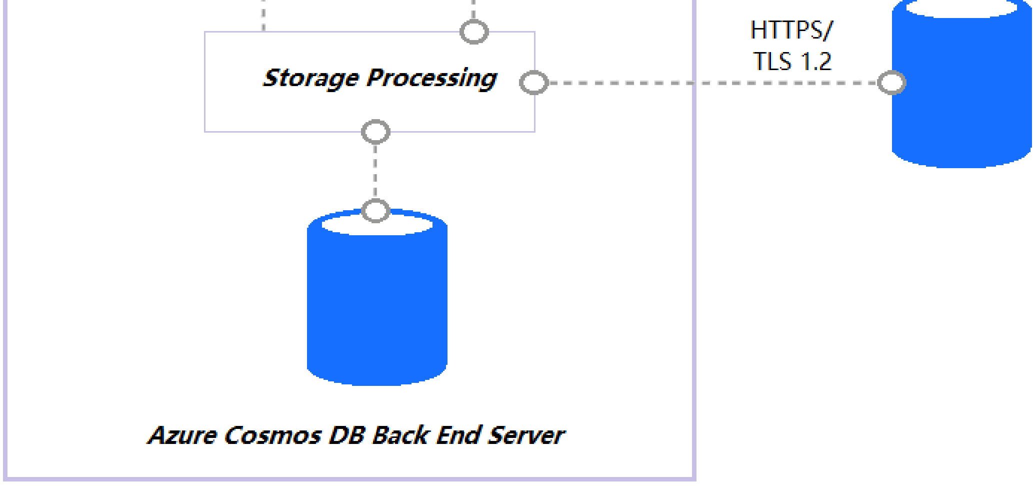
Review data encryption options

Azure Cosmos DB now uses encryption at rest for all its databases, backups, and media. When Azure Cosmos DB data is in transit, or over the network, that data is also encrypted.

Azure Cosmos DB at rest and in transit encryption implementation



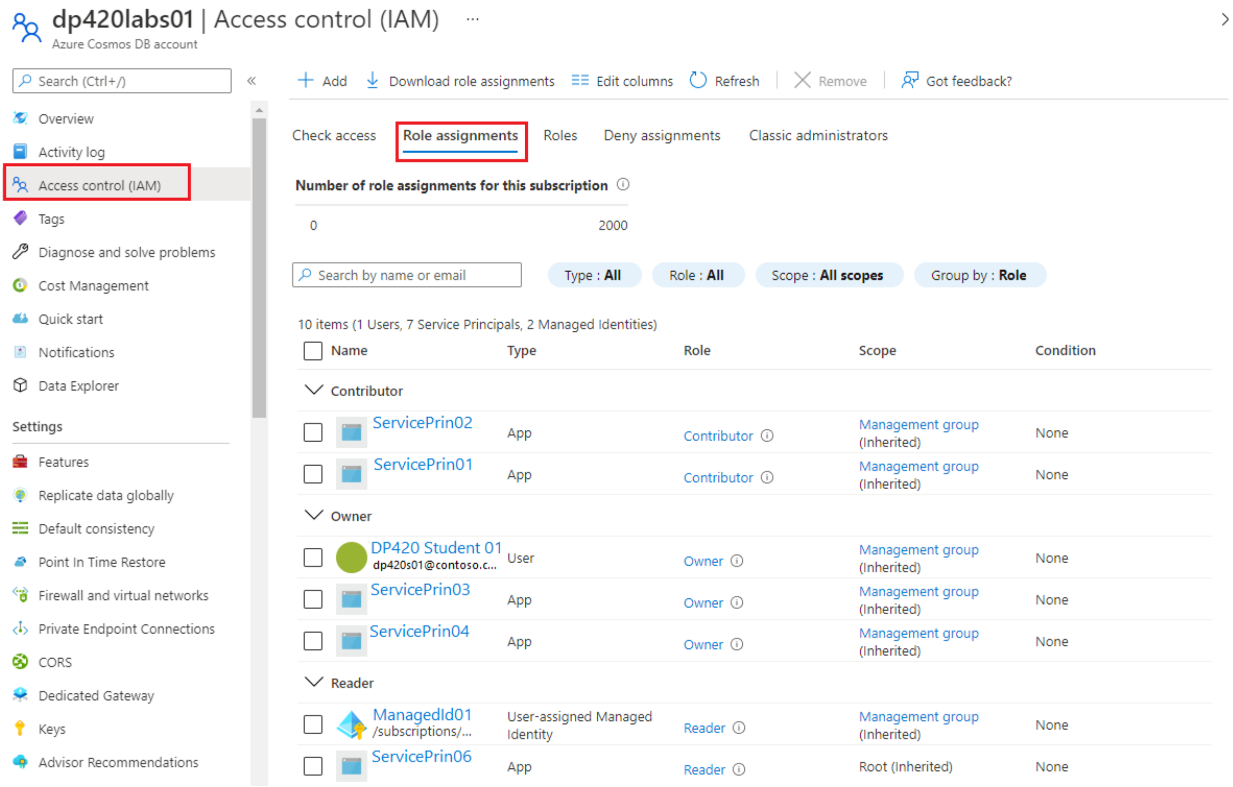
Secret Store
Stores S_K
Deploy S_K
Azure Cosmos DB Management Service (RP)
Unwraps S_K[Encryption key_K] --> Encryption key_K
S_K[Encryption key_K]
Encryption key_K



Use role-based access control (RBAC)

Azure role-based access control (RBAC) is provided in Azure Cosmos DB to do common management operations.

Built-in role	Description
DocumentDB Account Contributor	Can manage Azure Cosmos DB accounts.
Cosmos DB Account Reader	Can read Azure Cosmos DB account data.
Cosmos Backup Operator	Can submit a restore request for Azure portal for a periodic backup enabled database or a container. Can modify the backup interval and retention on the Azure portal. Can't access any data or use Data Explorer.
CosmosRestoreOperator	Can do restore action for Azure Cosmos DB account with continuous backup mode.
Cosmos DB Operator	Can provision Azure Cosmos accounts, databases, and containers. Can't access any data or use Data Explorer.



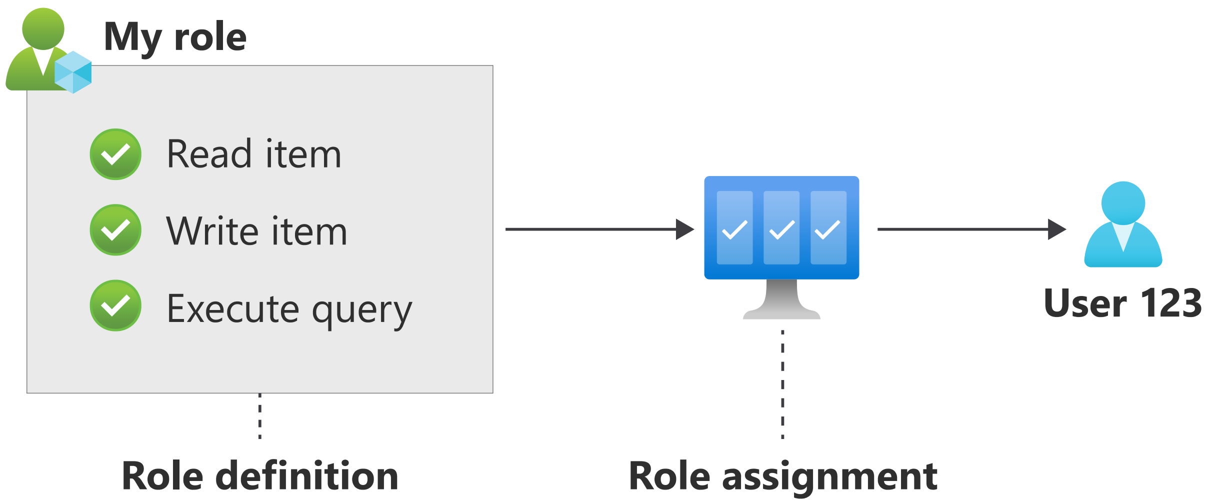
Number of role assignments for this subscription ⓘ

0 2000

10 items (1 Users, 7 Service Principals, 2 Managed Identities)

□ Name	Type	Role	Scope	Condition
Contributor				
□ ServicePrin02	App	Contributor ⓘ	Management group (Inherited)	None
□ ServicePrin01	App	Contributor ⓘ	Management group (Inherited)	None
Owner				
□ DP420 Student 01 dp420s01@contoso.c...	User	Owner ⓘ	Management group (Inherited)	None
□ ServicePrin03	App	Owner ⓘ	Management group (Inherited)	None
□ ServicePrin04	App	Owner ⓘ	Management group (Inherited)	None
Reader				
□ ManagedId01 /subscriptions/...	User-assigned Managed Identity	Reader ⓘ	Management group (Inherited)	None
□ ServicePrin06	App	Reader ⓘ	Root (Inherited)	None

Custom roles provide users a way to create Azure role definitions with a custom set of resource provider operations.



Access account resources using AAD

Access account resources using AAD allows you to authenticate your data requests with an Azure Active Directory (Azure AD) identity.

Permission model

- Microsoft.DocumentDB/databaseAccounts/sqlDatabases/containers/*
- Microsoft.DocumentDB/databaseAccounts/sqlDatabases/containers/items/create
- Microsoft.DocumentDB/databaseAccounts/sqlDatabases/containers/items/*
- Microsoft.DocumentDB/databaseAccounts/readMetadata
- Microsoft.DocumentDB/databaseAccounts/sqlDatabases/containers/executeStoredProcedure

To use Azure Cosmos DB RBAC with the SDK, you'll no longer pass the primary key. You'll pass an instance of a TokenCredential class.

```
TokenCredential servicePrincipal = new ClientSecretCredential(
    "<azure-ad-tenant-id>",
    "<client-application-id>",
    "<client-application-secret>");
CosmosClient client = new CosmosClient("<account-endpoint>", servicePrincipal);
```

In situations where you want to force clients to connect to Azure Cosmos DB through RBAC exclusively, you have the option to disable the account's primary/secondary keys. When doing so, any incoming request using either a primary/secondary key or a resource token will be actively rejected.

In [ ]:

```
az ad sp create-for-rbac --name "cosmosclientapp" --sdk-auth
```

This will generate a service principal

{ "clientId": "xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx", "clientSecret": "xxxxxxxxxxxxxxxxxxxxxxxxxxxx", "subscriptionId": "b895a719-7034-411a-9944-ff196d1f450f", "tenantId": "xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx", "activeDirectoryEndpointUrl": "https://login.microsoftonline.com", "resourceManagerEndpointUrl": "https://management.azure.com/", "activeDirectoryGraphResourceId": "https://graph.windows.net/", "sqlManagementEndpointUrl": "https://management.core.windows.net:8443/", "galleryEndpointUrl": "https://gallery.azure.com/", "managementEndpointUrl": "https://management.core.windows.net/" }

In [ ]:

```
az cosmosdb sql role definition list --account-name cosmos-dp-420-sql-provisioned --resourc
```

In [ ]:

```
az cosmosdb sql role assignment create `
az cosmosdb sql role assignment create
    --account-name cosmos-dp-420-sql-provisioned `
    --resource-group rg-dp-420 `
    --scope "/" `
    --principal-id "b4a58102-a65b-42a9-ac32-93676fc8d3a9" `
    --role-definition-id "00000000-0000-0000-0000-000000000001"
```

In [ ]:

```
using Microsoft.Azure.Cosmos;
using Azure.Identity;
using Azure.Core;

TokenCredential servicePrincipal = new ClientSecretCredential(
    "72f988bf-86f1-41af-91ab-2d7cd011db47", // <azure-ad-tenant-id>
    "b4a58102-a65b-42a9-ac32-93676fc8d3a9", // <client-application-id>
    "U-DWDXOOu7BVym7BHTPqsKxPNUPQDsHRfd");  // <client-application-secret>
CosmosClient client = new CosmosClient(documentEndpoint, servicePrincipal);
```

Demo teardown

In [ ]:

```
await database.DeleteAsync();
```