



# TD SYNnex DynamoDB AWS Version 1.0

Step by Step Guide



The information below includes detailed pre-deployment requirements, an in depth step by step guide for the AWS DynamoDB v1.0 Click to Run deployment, and post deployment steps that will need to be considered.

[Page 2: Infrastructure Requirements.](#)

[Page 3: Solution Overview.](#)

[Page 6: AWS DynamoDB v1 Click to Run Deployment.](#)

[Page 9: Post Deployment Activities.](#)

## DynamoDB v2.0 Pre-Requisites

- ✓ Infrastructure Requirements – No resources in AWS need to be pre-provisioned
  - *Make sure your data model is defined and the table name will be unique within the Region* – Table names are unique within an AWS region of your account, so you have to make sure the one used is not already taken, also a primary key needs to be defined before the table gets created
  - *Define the local indexes if they will be used* – Unlike Global Secondary Indexes, local indexes can only be defined during the table creation so it's not possible to do any changes post deployment
  - *Check your Account Limits:* Each AWS account has a specific amount of limits on how many of each resource is allowed per region. Those limits are known as quotas, and many of them are “soft limits”, meaning they can be increased by contacting AWS. The following resources need to have available quota before deployment is started. If the requirements are not met the user can either delete old objects in his account, chose another AWS Region or ask AWS for a soft limit increase

Resource	Default quota
Cloud trails (always deployed to us-east-1)	5
Number of DynamoDB tables per region	256

- *Create Role*: When the user starts the purchasing process he will be sent to the Role creation page. This creates a temporal permission set that allows resource creation inside his AWS account. The least-privilege concept is applied here and this role will be deleted automatically after a successful deployment. The user must have his AWS credentials and the MFA device at hand (if two-factor authentication is used). This step is always required

1- Enter your AWS Account Number

2- Click to Create Stack

Click to Create

3- Validate Stack

Click to Validate

## Solution Overview:

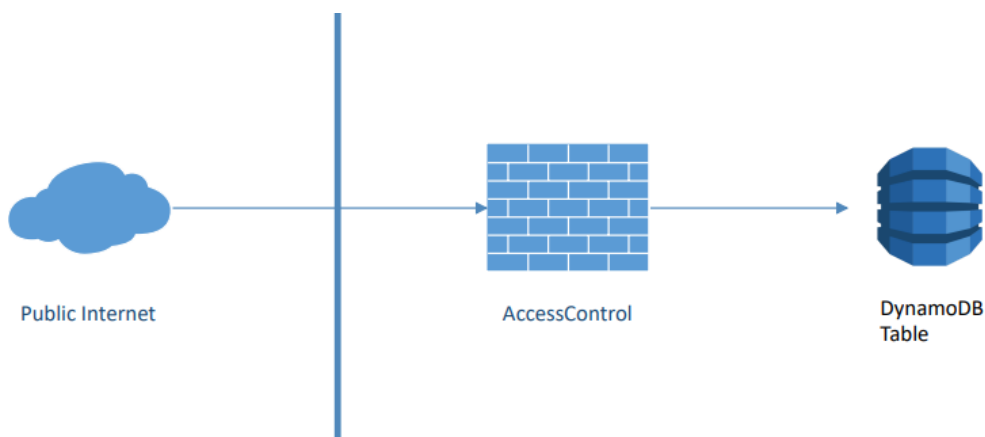
*“Amazon DynamoDB is a key-value and document database that delivers single-digit millisecond performance at any scale. Its fully managed, multi-region, multi-active, durable database with build-in security, backup and restore and in-memory caching for internet-scale applications”. DynamoDB can handle more than 10 trillion request per day and can support peaks of more than 20 million requests per second.*

**Parameters & Inputs** (a more detailed overview is shown on the next pages):

- Select the AWS Region to deploy to
- Input a Name for your DynamoDB table
- Input your Primary Key
- Input your Primary Key Type

- Choose whether to use a Sort Key or not
- Input your Sort Key Name(if used)
- Input your Sort key Type (if used)
- Select the Pricing Model (provisioned, pay per request)
- Choose the read/write IOPS if provisioned model is used
- Choose if to enable Streams on the table
- Select the Stream Type that will be used (if Streams are enabled)
- Select to add up to 3 Global Secondary Indexes
- Select to add up to 3 Local Secondary Indexes
- Put a Name on your Local/Global Indexes (if using any)
- Select a different key for each of your Local/Global Indexes (if using any)
- Choose the Key Type for your Local/Global Indexes (if using any)
- Choose the read/write IOPS if provisioned model is used for your Global Indexes only (if using any)
- Choose the projection type of your Local/Global Indexes (if using any)
- Choose the non-key attributes projected into your Local/Global Indexes (if using any)

## Deployment Architecture:



## AWS DynamoDB v2 Click to Run Deployment

### AWS DynamoDB v2 Deployment and Considerations

Purchase the AWS DynamoDB v2 Click to Run Solution through StreamOne Marketplace and proceed to the Digital Locker to configure and deploy the solution.

1. **Select an available AWS Region:** This is the region where the solution will be deployed. Some regions need to be enabled first on the target account before deploying to them, a warning message will appear in such case.

Location

---

Select AWS region

Select an available AWS Region

2. **Select the name of the DynamoDB table:** The Database table which is a collection of items. It needs to be unique within your region and you can have the same table in multiple regions for High Availability for example. This field is mandatory

Basic Information

---

Name of the DynamoDB Table

3. **Select the primary key:** The key used to uniquely identify each item inside the table. If not combined with a sort key only this partition key will be used. This field is mandatory.

Primary Key

4. **Select the type of your primary key:** Choose between the 3 options. This field is mandatory

Primary key type

String

Binary  
 String  
 Number

5. **Choose if to add a sort key:** The primary key apart from a single partition key can also be composed of a partition and sort key. If using a sort key choose the key and its type

Has sort key?

Sort Key

Sort key type

String

6. **Select the pricing Model:** There are two different options to select from. One is using provisioned IOPS for predictable workloads, the second one offers unlimited on demand capacity

Pricing Model

PROVISIONED

PROVISIONED  
PAY\_PER\_REQUEST

7. **Select the required read/write throughput:** Those fields will be only available for the Provisioned pricing Model. You need to predict the workload your application will have and configure the amount of strongly consistent reads and writes per second. The more capacity will be provisioned the bigger the monthly costs will be. Those value can be adjusted later on as the application grows or can even be controlled by autoscaling mechanisms

Read Capacity per second

Write Capacity per second

1

1

8. **Add secondary indexes if you require them:** Up to 3 global and 3 local secondary indexes can be added (max 6 in total). The configuration fields are identical for both types of indexes, except for the throughput values. The types of indexes differ on how they work inside the table and how data is stored, please refer to official AWS documentation to understand the difference when choosing which type to use.

Add a Secondary Global index?

Add a Secondary Local index?

9. **Select a name for every selected index:** Set those fields only if you choose to add a new index

Name of first global secondary index in the table

**10. Select a key for every selected index:** Set those fields only if you choose to add a new index

The key of first global secondary index in the table

**11. Select a type for every index key:** Set those fields only if you choose to add a new index

Type of first global secondary index key

String

Binary

String

Number

**12. Select the required read/write throughput for the index:** Use those fields only if using the Provisioned pricing model and when configuring global indexes.

Read Capacity per second

Write Capacity per second

**13. Select the Projection type of the index:** Which attributes will be copied into the index. The “INCLUDE” option will show an additional field where the user inputs the fields he wants

First global secondary index key projection type

INCLUDE

Non-key attributes that will be projected into the index

Null

**14. Add another index if needed:** Once you have configured the previous index choose if you want to add another one (max 3 of every type)

Add a second Secondary Global Index?

Add a Secondary Local Index?

-- Proceed to the next page to for post-deployment steps --

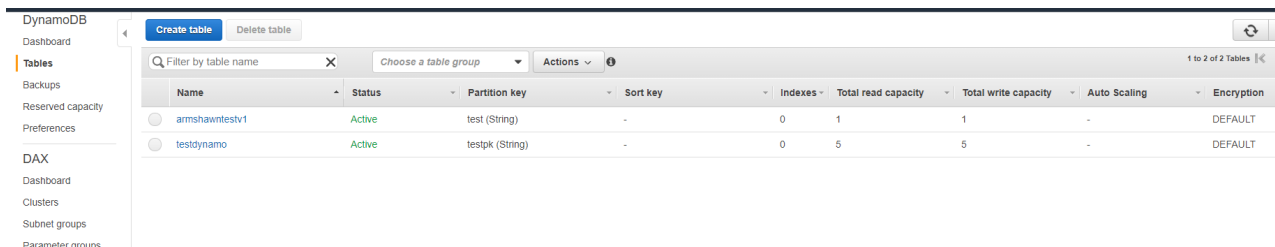
## Post-Deployment activities for AWS DynamoDB v2:

- Verifying the state of your instance.
- Connecting to your database.

## Post Deployment Activities

### Verifying the state of your table.

When the DynamoDB table is first deployed, its state will change to “Active” after some seconds. If the instance doesn’t start there might have been a problem during startup, so either check the deployment logs or contact our support. To verify the state, go to your AWS console. Once logged in, select the service “DynamoDB”. Go to “Tables”. Your table should appear there, with the name you have selected during the deployment process. The state will be displayed in the “Status” column.



Name	Status	Partition key	Sort key	Indexes	Total read capacity	Total write capacity	Auto Scaling	Encryption
armshawntestv1	Active	test (String)	-	0	1	1	-	DEFAULT
testdynamo	Active	testpk (String)	-	0	5	5	-	DEFAULT

### Connecting to your Database.

To access the table and preform CRUD operations you will require an IAM user that has valid long term programmatical access keys if connecting from outside or configure an IAM role if connecting from inside the AWS Cloud. If you are having problems connecting to DynamoDB you need to review the IAM policies of the corresponding user, group or role. Please note that IAM allows specific read only or write permissions to all or individual tables, for more information please consult the AWS documentation.