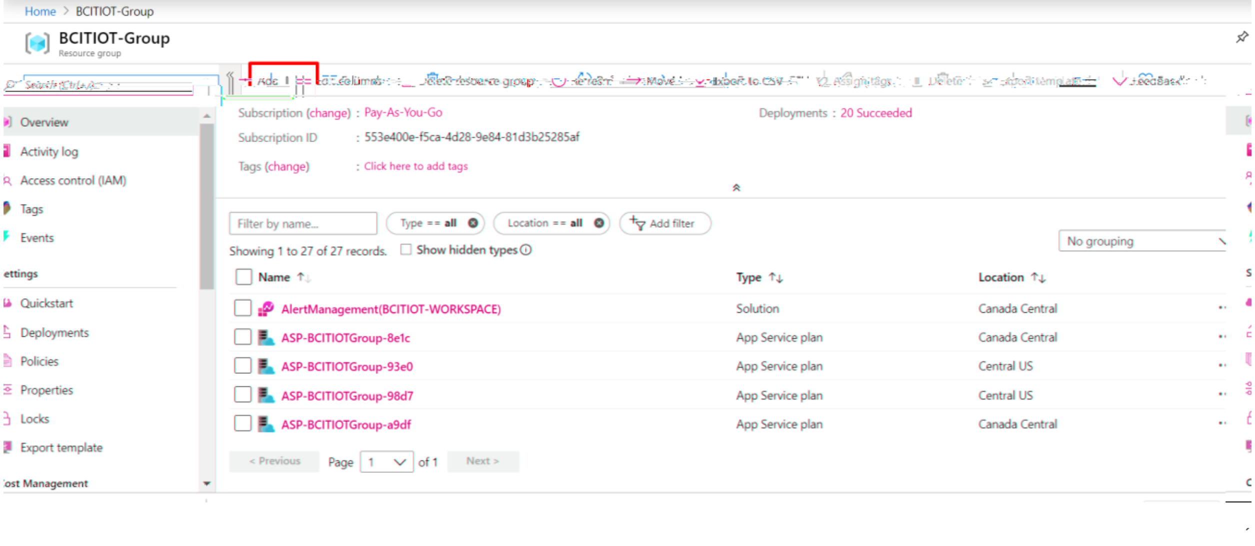
Exporting all data sent to IoT Central into a Cosmos DB (SQL-like database)

Get an Event Hub



Home > BCITIOT-Group

BCITIOT-Group
Resource group

Add | Edit columns | Delete resource group | Refresh | Move | Export to CSV | Assign tags | Delete | Export template | Feedback

Subscription (change)	: Pay-As-You-Go
Subscription ID	: 553e400e-f5ca-4d28-9e84-81d3b25285af
Tags (change)	: Click here to add tags

Deployments : 20 Succeeded

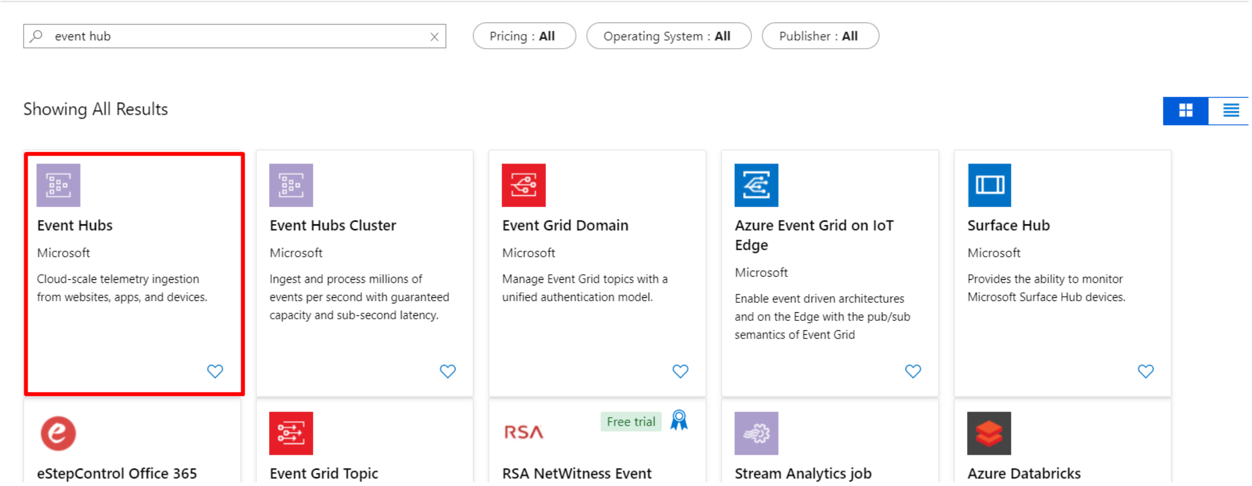
Filter by name... | Type == all | Location == all | Add filter

Showing 1 to 27 of 27 records. □ Show hidden types

No grouping

Name	Type	Location
AlertManagement(BCITIOT-WORKSPACE)	Solution	Canada Central
ASP-BCITIOTGroup-8e1c	App Service plan	Canada Central
ASP-BCITIOTGroup-93e0	App Service plan	Central US
ASP-BCITIOTGroup-98d7	App Service plan	Central US
ASP-BCITIOTGroup-a9df	App Service plan	Canada Central

< Previous Page 1 of 1 Next >

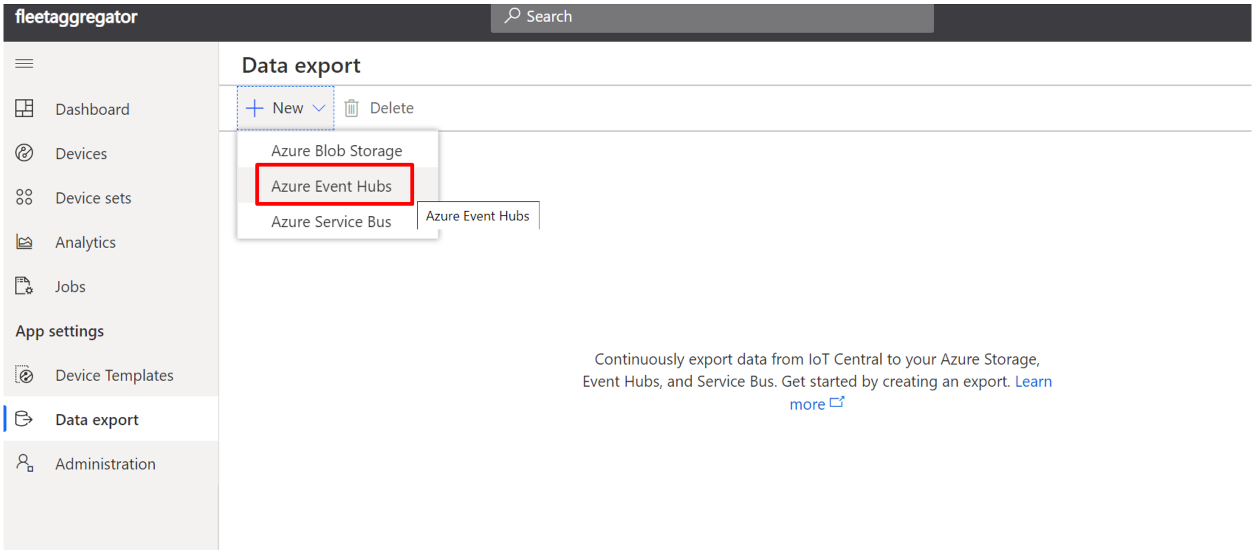


event hub | Pricing : All | Operating System : All | Publisher : All

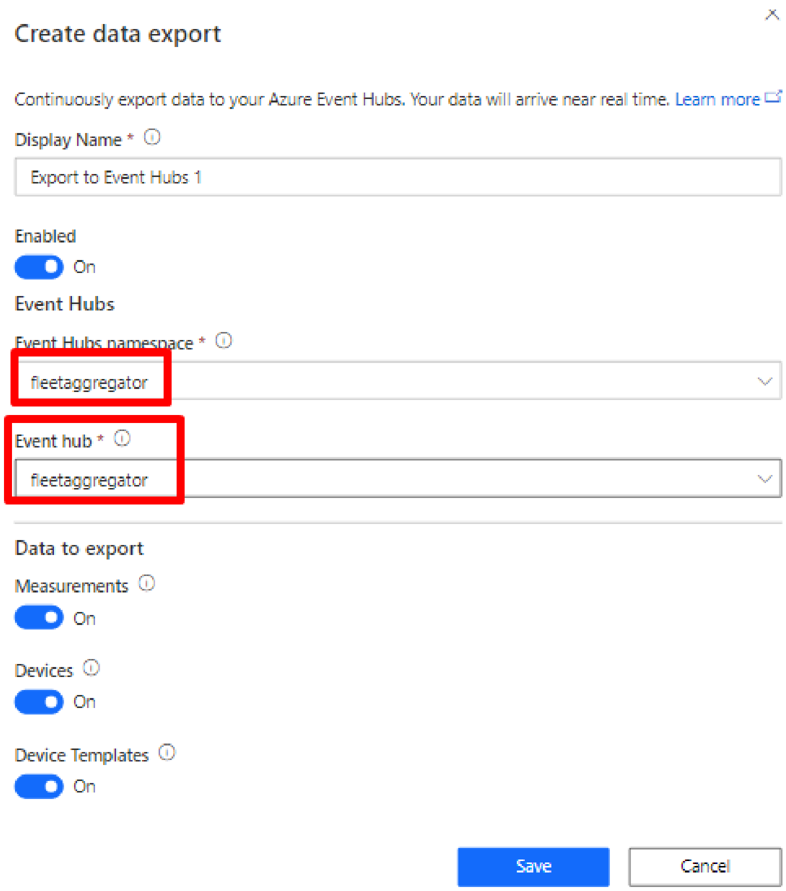
Showing All Results

Event Hubs Microsoft Cloud-scale telemetry ingestion from websites, apps, and devices.	**Event Hubs Cluster** Microsoft Ingest and process millions of events per second with guaranteed capacity and sub-second latency.	**Event Grid Domain** Microsoft Manage Event Grid topics with a unified authentication model.	**Azure Event Grid on IoT Edge** Microsoft Enable event driven architectures and on the Edge with the pub/sub semantics of Event Grid	**Surface Hub** Microsoft Provides the ability to monitor Microsoft Surface Hub devices.
eStepControl Office 365	**Event Grid Topic**	Free trial **RSA NetWitness Event**	**Stream Analytics job**	**Azure Databricks**

Start exporting your data to the Event Hubs you just made



Continuously export data from IoT Central to your Azure Storage, Event Hubs, and Service Bus. Get started by creating an export. Learn more



Create data export

Continuously export data to your Azure Event Hubs. Your data will arrive near real time. Learn more

Display Name * : Export to Event Hubs 1

Enabled: On

Event Hubs

Event Hubs namespace * : fleetaggregator

Event hub * : fleetaggregator

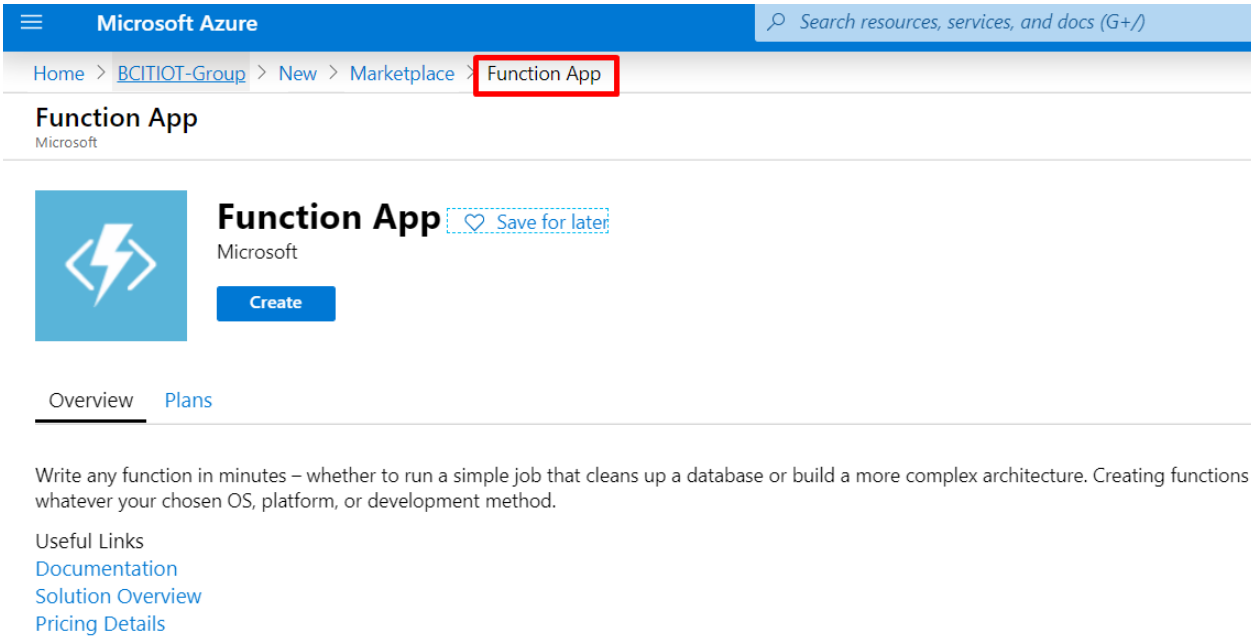
Data to export

Measurements: On

Devices: On

Device Templates: On

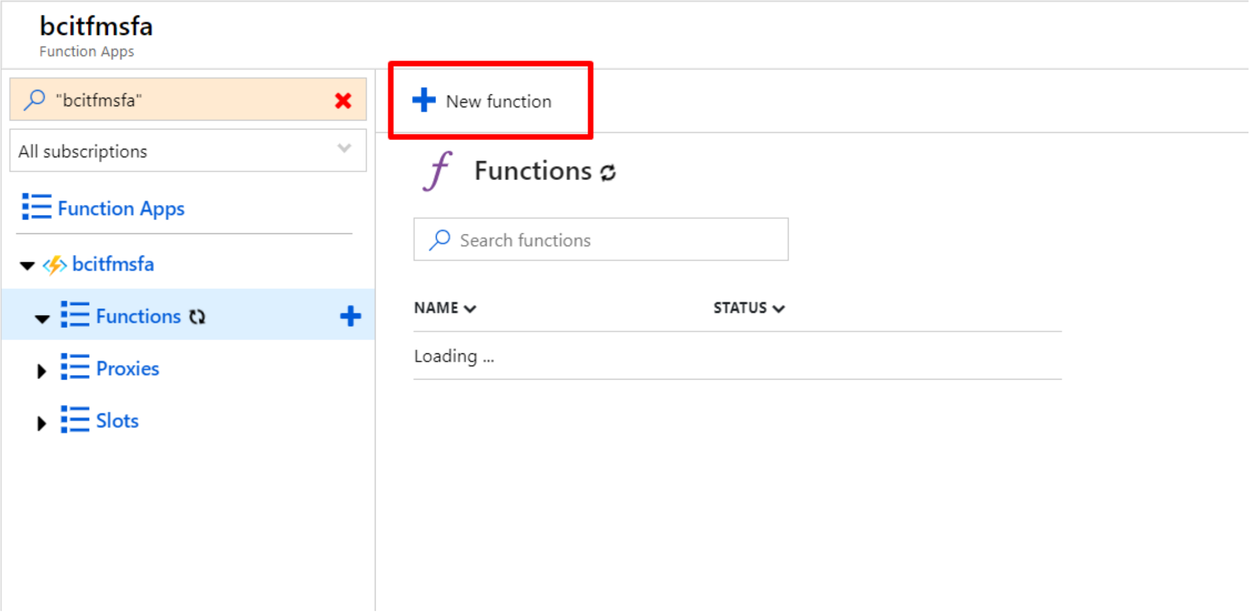
Get a FunctionApp service

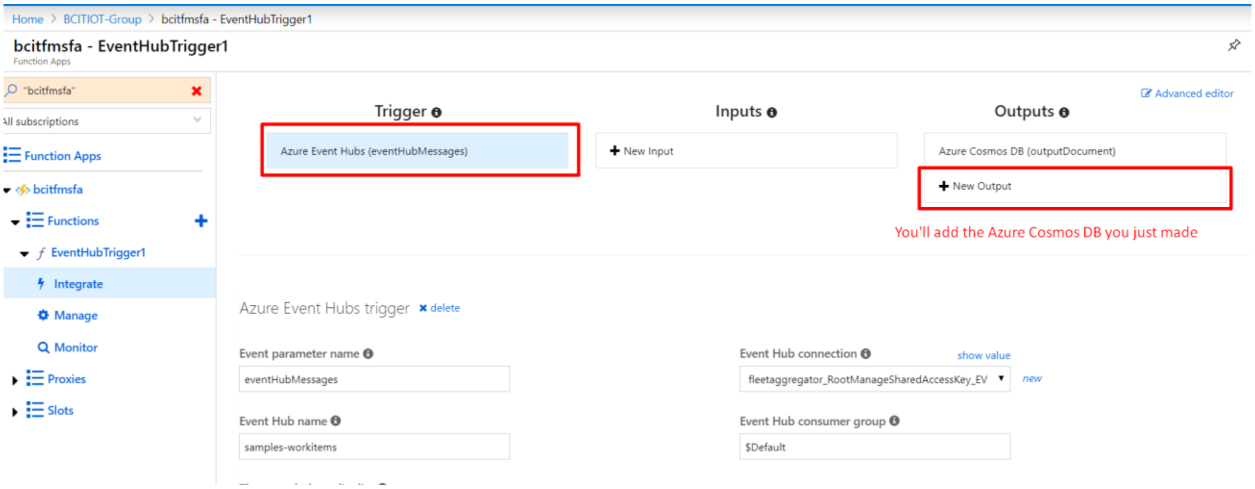


Microsoft Azure
Search resources, services, and docs (G+/)
Home > BCITIOT-Group > New > Marketplace > Function App
Function App
Microsoft
Function App
Save for later
Microsoft
Create
Overview
Plans

Write any function in minutes – whether to run a simple job that cleans up a database or build a more complex architecture. Creating functions whatever your chosen OS, platform, or development method.

Useful Links
Documentation
Solution Overview
Pricing Details





You'll add the Azure Cosmos DB you just made

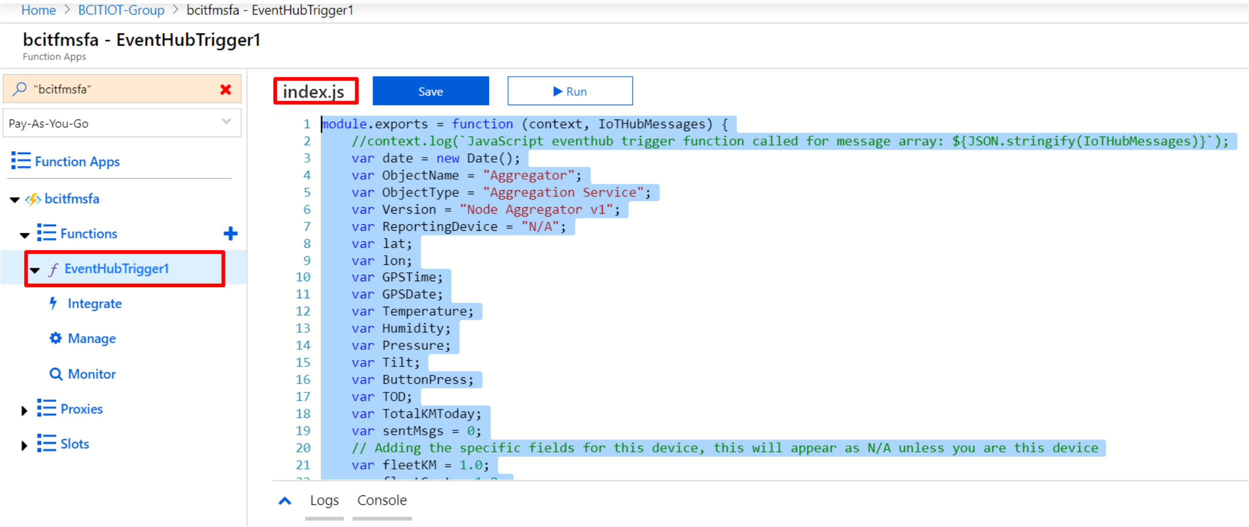
Azure Event Hubs trigger ✖ delete

Event parameter name: eventHubMessages

Event Hub connection: fleetaggregator_RootManageSharedAccessKey_EV (show value, new)

Event Hub name: samples-workitems

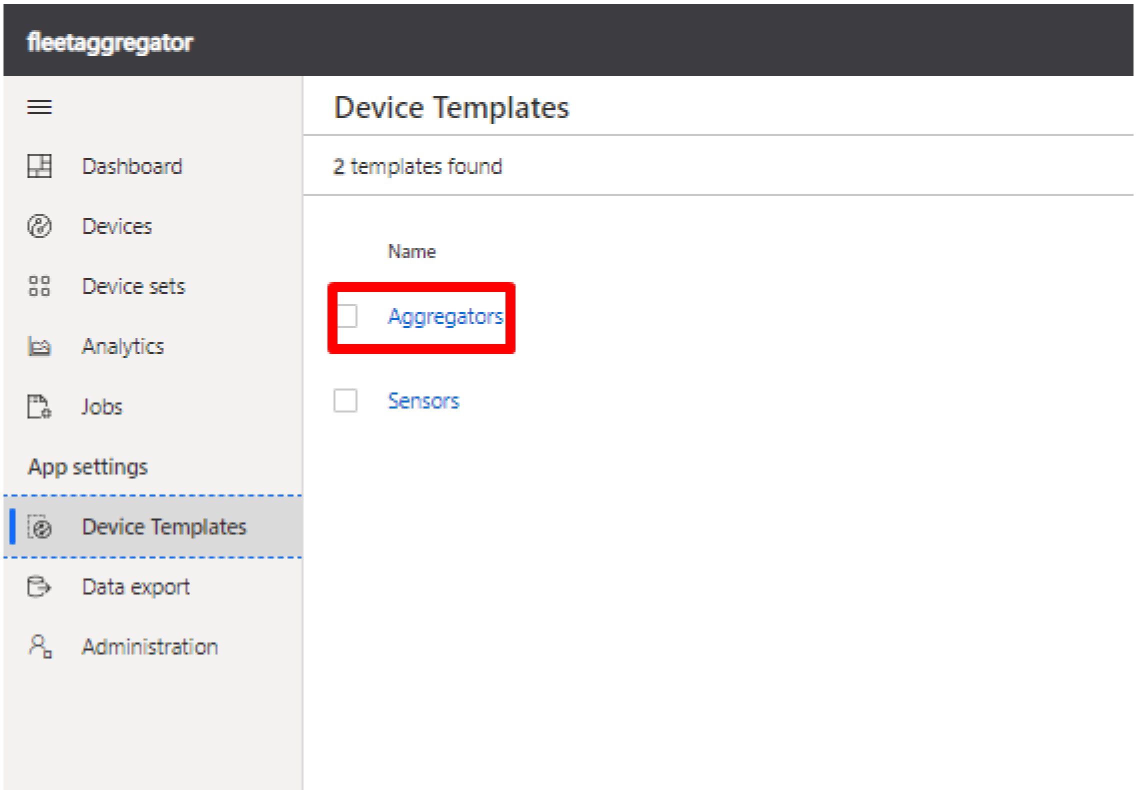
Event Hub consumer group: $Default



```
module.exports = function (context, IoTHubMessages) {
    //context.log(`JavaScript eventhub trigger function called for message array: ${JSON.stringify(IoTHubMessages)}`);
    var date = new Date();
    var ObjectName = "Aggregator";
    var ObjectType = "Aggregation Service";
    var Version = "Node Aggregator v1";
    var ReportingDevice = "N/A";
    var lat;
    var lon;
    var GPSTime;
    var GPSDate;
    var Temperature;
    var Humidity;
    var Pressure;
    var Tilt;
    var ButtonPress;
    var TOD;
    var TotalKMToday;
    var sentMsgs = 0;
    // Adding the specific fields for this device, this will appear as N/A unless you are this device
    var fleetKM = 1.0;
```

```
l                                        l

  g  h h       C Ci   f l  C f   h   CL  K ePh  djh  C C
CCCC  f   h      j  Md dVf l  Ch h     eC  ljjh Ci   f l  C fd  hgCi  C h  djhCd  d  C
  MVR     l jli  L  K ePh  djh       C
CCCC d Cgd hC C h CGd h   C
CCCC d CRe hf  d hC C Djj hjd     C
CCCC d CRe hf    hC C Djj hjd l  CVh  l fh  C
CCCC d C h  l  C C   ghCDjj hjd   C    C
CCCC d C h    l jGh l fhC C   D  C
CCCC d C d  C
CCCC d C    C
CCCC d CJSV l h C
CCCC d CJSVGd h C
CCCC d C h  h d   h C
CCCC d CK  lgl   C
CCCC d CS h    h C
CCCC d C l   C
CCCC d CE     S h   C
CCCC d C RG C
CCCC d C   d NP  gd  C
CCCC d C h  P j C C  C
CCCC  CDggl jC  hC  hflilfCilh g Ci  C  l Cgh lfh C  l C l  Cd  hd Cd C  DC   h  C
   Cd hC    Cgh lfhC
CCCC d Ci  h NPC C    C
CCCC d Ci  h     C C    C
C
CCCL  K eP   djh  i  Hdf   h  d hC  C C
CCCCCCCf   h     j  S l  l jC  C d C h  djhC hfhl hg      C
CCCCCCCf   h     j MVR     l jli   h  djh C     C    C
C

CCCCCCCC  f   h     j  Md dVf l  Ch h    eC  ljjh Ci   f l  Cfd  hgCi  C h  djhC
d  d  C  MVR     l jli  L  K eP   djh      C
CCCCCCCCRe hf  d hC C h  djh Re hf  d h C
CCCCCCCCRe hf    hC C h  djh Re hf    h C
CCCCCCCC h  l  C C h  djh  h  l   C
CCCCCCCC h    l jGh lfhC C h  djh  h    l jGh lfh C
CCCCCCCC d C C h  djh   fd l     d    C
CCCCCCCC   C C h  djh   fd l          C
CCCCCCCCJSV l hC C h  djh JSV l h C
CCCCCCCCJSVGd hC C h  djh JSV l h C
CCCCCCCC h  h d   hC C h  djh  h  h d   h C
CCCCCCCCK  lgl  C C h  djh K  lgl   C
CCCCCCCCS h    hC C h  djh S h    h C
CCCCCCCC l  C C h  djh  l   C
CCCCCCCCE     S h  C C h  djh E     S h   C
CCCCCCCC RGC C h CGd h     O fd hV  l j   C
CCCCCCCC    NP  gd C C    djh    d NP  gd  C
CCCCCCCC h  P j C C h  d h  h  P j  C
CCCCCCCC  C ggl jC  hC   flilfCilh g Ci  C  l Cgh lfh C  l C l  Cd  hd Cd C  DC
   h  C   C  hC  l Cgh l hC
CCCCCCCCi h  NPC C h  dj  i hh NP C
CCCCCCCCi h      C C h   jh i hh      C
C
CCCC   C
C
CCCC d C     C C C
C
CCCCCCCC Re hf  d h  CRe hf  d h C
CCCCCCCC Re hf    h  CRe hf    h C
CCCCCCCC  h  l    C h  l   C
CCCCCCCC  h    l jGh lfh  C h    l jGh lfh C
CCCCCCCC  d   C d  C
CCCCCCCC      C    C
CCCCCCCC JSV l h  CJSV l h C
CCCCCCCC JSVGd h  CJSVGd h C
CCCCCCCC  h  h d   h  C h  h d   h C
CCCCCCCC K  lgl    CK  lgl   C
CCCCCCCC S h    h  CS h    h C
CCCCCCCC  l    C l   C
CCCCCCCC E     S h    CE      S h   C
CCCCCCCC  RG  C RG C
CCCCCCCC    d NP  gd   C   d NP  gd  C
CCCCCCCC  h  P j   C h  P j  C
CCCCCCCC i hh NP C Ci hh NP C
CCCCCCCC i hh      C Ci hh      C
CCCC  C
C
CCCC    l C l  C  l  C    C      C e hf Cl CdC lfh  Ci   d  hgC d  C
CCCCCCCf   h     j  RG  C
CCCCCCCf   h     j MVR     l jli         C     C    C
C
CCCCf   h   el gl j        G f  h  C CMVR     l jli          C
C
CCCCf   h   g  h   C
  C
```

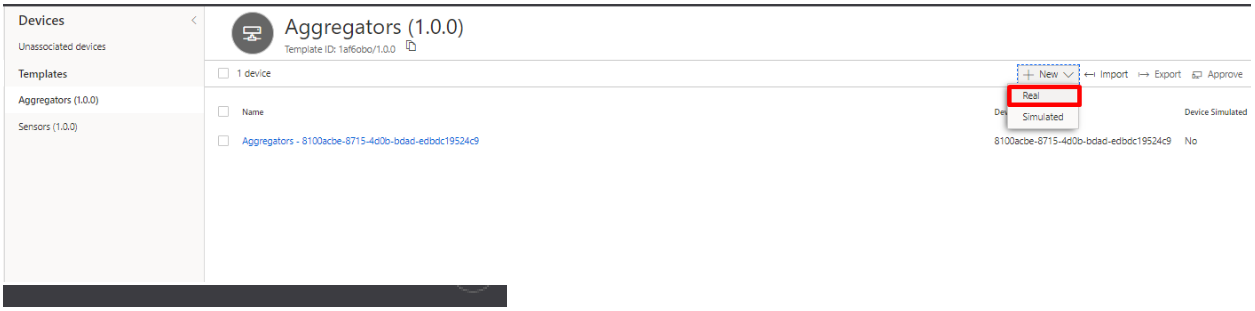
Provision a device to be the aggregator in IoT Central



Device Templates

2 templates found

Name
□ Aggregators
□ Sensors

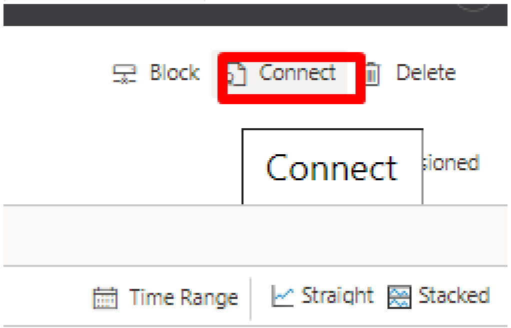


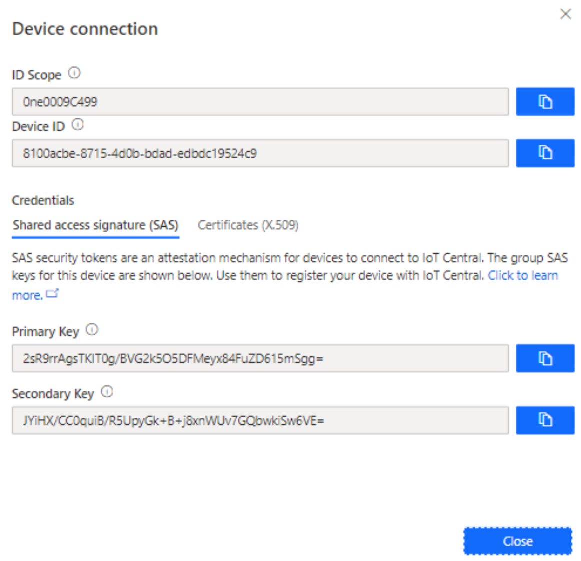
Aggregators (1.0.0)

Template ID: 1af6obo/1.0.0

□ 1 device

□ Name	Dev	Device Simulated
□ Aggregators - 8100acbe-8715-4d0b-bdad-edbdc19524c9	8100acbe-8715-4d0b-bdad-edbdc19524c9	No





Device connection

ID Scope ⓘ

0ne0009C499

Device ID ⓘ

8100acbe-8715-4d0b-bdad-edbdc19524c9

Credentials

Shared access signature (SAS) | Certificates (X.509)

SAS security tokens are an attestation mechanism for devices to connect to IoT Central. The group SAS keys for this device are shown below. Use them to register your device with IoT Central. Click to learn more.

Primary Key ⓘ

2sR9rrAgsTKIT0g/BVG2k5O5DFMeyx84FuZD615mSgg=

Secondary Key ⓘ

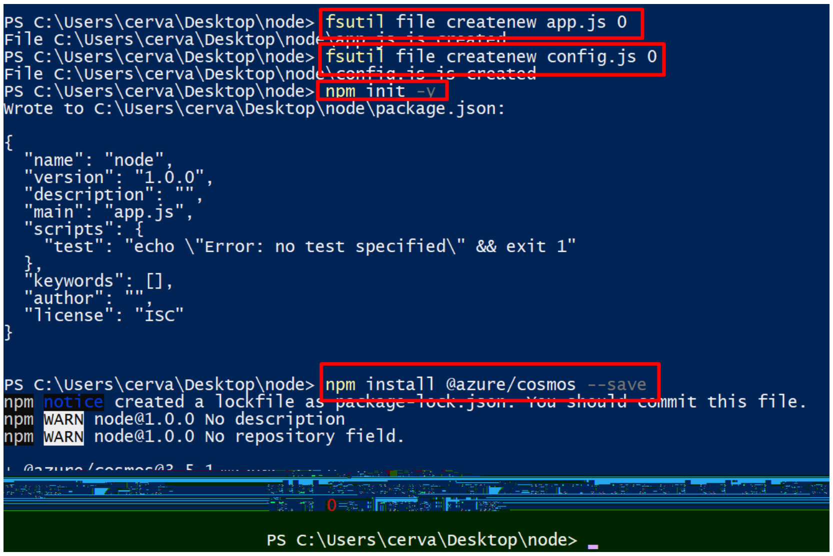
JYiHX/CC0quiB/R5UpyGk+B+j8xnWUv7GQbwkiSw6VE=

R

Pull all the data you want via a Node.js service

R

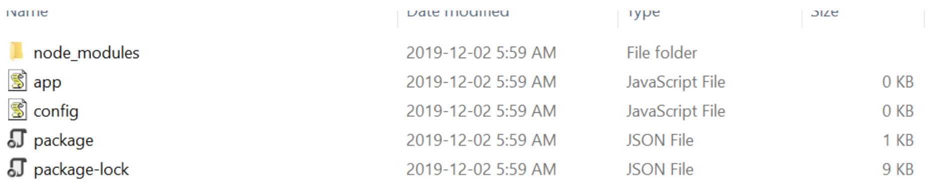
Set up a node project with dependencies to work with Azure



```
PS C:\Users\cerva\Desktop\node> fsutil file createnew app.js 0
File C:\Users\cerva\Desktop\node\app.js is created
PS C:\Users\cerva\Desktop\node> fsutil file createnew config.js 0
File C:\Users\cerva\Desktop\node\config.js is created
PS C:\Users\cerva\Desktop\node> npm init -y
Wrote to C:\Users\cerva\Desktop\node\package.json:

{
  "name": "node",
  "version": "1.0.0",
  "description": "",
  "main": "app.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}


PS C:\Users\cerva\Desktop\node> npm install @azure/cosmos --save
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN node@1.0.0 No description
npm WARN node@1.0.0 No repository field.

+ @azure/cosmos@3.5.1

                          PS C:\Users\cerva\Desktop\node> _
```

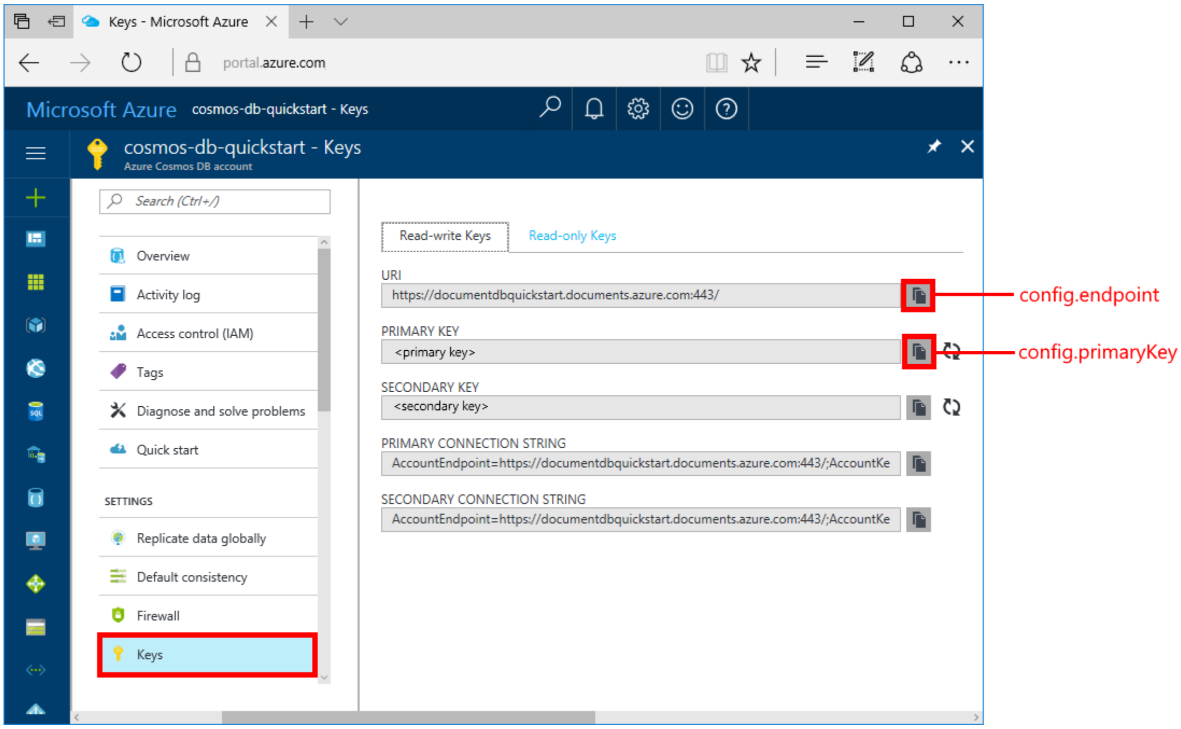
- ○
- ○

- ○
- ○



Name	Date modified	Type	Size
node_modules	2019-12-02 5:59 AM	File folder	
app	2019-12-02 5:59 AM	JavaScript File	0 KB
config	2019-12-02 5:59 AM	JavaScript File	0 KB
package	2019-12-02 5:59 AM	JSON File	1 KB
package-lock	2019-12-02 5:59 AM	JSON File	9 KB

Set your app's configurations



URI
https://documentdbquickstart.documents.azure.com:443/

PRIMARY KEY
<primary key>

SECONDARY KEY
<secondary key>

PRIMARY CONNECTION STRING
AccountEndpoint=https://documentdbquickstart.documents.azure.com:443/;AccountKe

SECONDARY CONNECTION STRING
AccountEndpoint=https://documentdbquickstart.documents.azure.com:443/;AccountKe

Sample config.js connection strings and writing your aggregation code

```
    config

config endpoint   "https://fmscosmos.documents.azure.com:443/"
config key
"0QwJ7UK7BwDh9MJabChx8nei5aj8PKH0jFRTYLQAPyOx813bQCVDRnNL5jRgfcUKQDJ2Q5lfKq5P
taKwzhrjDg=="

config database
  id  "outDatabase"


config container
  id  "MyCollection"


config items

module exports   config
```

Sample app.js

R

```
//@ts-check
      CosmosClient   require '@azure/cosmos'  CosmosClient
      config   require './config'
      url   require 'url'
      endpoint   config endpoint
      key   config key
      databaseId   config database id
      containerId   config container id
      partitionKey     kind  'Hash'  paths   '/Country'
      client       CosmosClient   endpoint  key


/**
 * Application specific variables
 */
    nucleoboard2km
    nucleoboard3km
    date       Date

/**
 * Variables to make sure we transmit the same format as the other devices
 */

    ObjectName   "Aggregator"
    ObjectType   "Aggregation Service"
    Version   "Node Aggregator v1"
    ReportingDevice   "N/A"

  "lat"  ""
  "lon"  ""

    GPSTime   "N/A"
    GPSDate   "N/A"
    Temperature   "N/A"
    Humidity   "N/A"
    Pressure   "N/A"
    Tilt   "N/A"
    ButtonPress   "N/A"
    TOD   date toLocaleString
    TotalKMToday   "N/A"
    sentMsgs   0
// Adding the specific fields for this device, this will appear as N/A unless
you are this device
    fleetKM   1.0
    fleetCost   1.0

/**
 * Sending transmission code to move data to IoT Hub, this will get picked up
by the FunctionApp service
 * The method which actually handles the sending is in the aggregateKms
function
 */









    Mqtt   require 'azure-iot-device-mqtt'  Mqtt
    DeviceClient   require 'azure-iot-device'  Client
    Message   require 'azure-iot-device'  Message
    clientSend   DeviceClient fromConnectionString connectionString  Mqtt

/**
 * Create the database if it does not exist
 */
               createDatabase
          database           client databases createIfNotExists
    id  databaseId

  //console.log(`Created database:\n${database.id}\n`)


/**
 * Read the database definition
 */
               readDatabase
          resource  databaseDefinition           client
```

```
     database databaseId
     read
  //console.log(`Reading database:\n${databaseDefinition.id}\n`)


/**
 * Create the container if it does not exist
 */
               createContainer
          container           client
     database databaseId
     containers createIfNotExists
        id  containerId  partitionKey
        offerThroughput  400

  //console.log(`Created container:\n${config.container.id}\n`)


/**
 * Read the container definition
 */
               readContainer
          resource  containerDefinition           client
     database databaseId
     container containerId
     read
  //console.log(`Reading container:\n${containerDefinition.id}\n`)



/**
 * Cleanup the database and collection on completion
 */
               cleanup
        client database databaseId


/**
 * Exit the app with a prompt
 *        {string} message - The message to display
 */
         exit message
  console log message
  console log 'Press any key to exit'
  process stdin setRawMode
  process stdin resume
  process stdin on 'data'  process exit bind process  0
```

Sample Query Functions

```
/**
 * Calls for the latest transmission received from device NUCLEOBOARD2
 */
```

```
               executeQuery1
    console log

            resources  results           client
     database databaseId
     container containerId
     items query 'SELECT TOP 1 r.ObjectName, r.TotalKMToday, r._ts FROM root
r WHERE r.ObjectName = "NUCLEOBOARD2"  ORDER BY r._ts DESC'
     fetchAll

    console log results
    nucleoboard2km   results 0  "TotalKMToday"




/**
 * Calls for the latest transmission received from device NUCLEOBOARD3
 */
               executeQuery2

            resources  results           client
         database databaseId
         container containerId
         items query 'SELECT TOP 1 r.ObjectName, r.TotalKMToday, r._ts FROM
root r WHERE r.ObjectName = "NUCLEOBOARD3"  ORDER BY r._ts DESC'
         fetchAll


    console log results
    nucleoboard3km    results 0  "TotalKMToday"
    console log




/**
 * This function takes the totalKMs for each device, and adds them together
 * and stores them in variable totalKM. Then, the message is sent.
 */
         aggregateKms

    // This manages message sending

    // Simulate telemetry.
        sentMsgs
        executeQuery1
        executeQuery2





        fleetCost   fleetKM   .54
            message       Message JSON stringify
            ObjectName   ObjectName
            ObjectType   ObjectType
            Version   Version
            ReportingDevice   ReportingDevice

            GPSTime   GPSTime
            GPSDate   GPSDate
            Temperature   Temperature
            Humidity   Humidity
            Pressure   Pressure
            Tilt   Tilt
            ButtonPress   ButtonPress
            TOD       Date   toLocaleString
            TotalKMToday   TotalKMToday
            sentMsgs   sentMsgs
            fleetKM   fleetKM
            fleetCost   fleetCost


        console log 'Sending message: '   message getData

        // Send the message.
        clientSend sendEvent message            err
                err
                console error 'send error: '   err toString

                console log "Message sent at: "
Date   toLocaleString


       3000   // Sends every 3000ms




/**
 * This drives the program and queries the devices, but will enter a
setInterval
 * loop at aggregateKms which runs the transmission every 15000ms
 */
createDatabase
   then       readDatabase
   then       createContainer
   then       readContainer
   then       aggregateKms
   then
    exit `Completed successfully - transmission reports will display as they
are sent...`

         error
    exit `Completed with error ${JSON.stringify(error)}`
```

Run the service