Configure the connector to be compliant with UDMI

The Universal Device Management Interface (UDMI) provides high-level specifications for the management and operation of physical Internet of Things systems. This data is usually exchanged with cloud entities that can maintain "digital twins" or "shadow devices" in the cloud. It is nominally used with Googe's Cloud IoT Core, and as an architecture, it can be applied to any data set or hosting setup. In addition, the architecture provides provisions for basic telemetry ingestion, such as the flow of data points from IoT devices.

In this tutorial we'll see step by step how to implement UDMI using the connectors Framework.

Cloud Setup

Create the registry

1. Go to the GCP console from the left side navigation pane find the Big Data section then choose lot Core service.



2. Create a device registry.

noT Core	Registries	+ CREATE REGISTRY
Filter registries		<u> </u>
Registry ID 🕇		

3. Give your registry a name and select the most suited region for you.

	 Create a registry
efine how devices in thi egistry, you can start ad	s registry will send data to Cloud IoT Core. After you create your ding devices to it. <u>Learn more</u>
Registry propert	ies
Registry propert Registry ID udmi-registry	your registry. 3-255 characters. Start with a letter. You can also
Registry propert Registry ID udmi-registry Permanent identifier for y include numbers and the	/our registry. 3-255 characters. Start with a letter. You can also following characters: + . % ~
Registry propert Registry ID udmi-registry Permanent identifier for y include numbers and the Region	/our registry. 3-255 characters. Start with a letter. You can also following characters: + . % ~

4. On the PubSUb section create a new topic where events will be published by IoT Core devices.

oud IoT Core routes dev essages to different top the messages. <u>Learn m</u>	vice messages to Cloud Pub/Sub for aggregation. You pics and subfolders in Cloud Pub/Sub based on the ty <u>nore</u>	ı can pe of
Select a Cloud Pub/Sub to	opic	
Device telemetry events w	vill be published to this topic by default.	
	OPIC	
SHOW ADVANCED OP	TIONS	
CREATE CANCEL		
Create a topic	hat you can use it in your device registry.	
Create a topic Add a topic to Pub/Sub so th Topic ID *	hat you can use it in your device registry.	
Create a topic Add a topic to Pub/Sub so th Topic ID * udmi-telemetry Topic name: projects/btiblot	hat you can use it in your device registry. core/topics/udmi-telemetry	
Create a topic Add a topic to Pub/Sub so th Topic ID * udmi-telemetry Topic name: projects/btiblot Encryption	hat you can use it in your device registry.	
Create a topic Add a topic to Pub/Sub so th Topic ID * udmi-telemetry Topic name: projects/btibiot Encryption © Google-managed key No configuration required	hat you can use it in your device registry.	
Create a topic Add a topic to Pub/Sub so th Topic ID * udmi-telemetry Topic name: projects/btibiot Encryption Google-managed key No configuration required Customer-managed key Manage via Google Cloud	hat you can use it in your device registry. Image: core/topics/udmi-telemetry IKey Management Service	

5. Add another topic for device state. Click on show advanced config

Device state topic (optional)

Device state data will be published to your selected topic on a best-effort basis, as well as to the default MQTT state topic (if your devices use MQTT protocol). Learn more

Select a Cloud Pub/Sub topic	
projects/btibiotcore/topics/udmi-state	•

6. Then hit create

rou can auu more cerunicates arter you ve createu uns registry.





CANCEL

Create a service account

1. From the left navigation bar select IAM & Admin



2. Then Service Accounts



3. And hit create



Service accounts for project "BtibloTCore"

A service account represents a Google Cloud service identity such as code running on Compute Engine V

4. Then give your service account a name and a description.



5. Choose the Cloud IoT Admin (you can add conditions to give permission on a specific resources recommended). and click on Continue.



6. Create a key and choose json format then hit Create.



7. A json file will be dwonloaded containing the access key for this service account save it we will need it later.

•	Labels	Service account admins role
\$	Settings	Grant users the permission to administer this service accor
0	Privacy & Security	Private key saved to your computer
۲	Cryptographic Keys	Create key (optional)
	Identity-Aware Proxy	Download a file that contains the private key. Store the f can't be recovered if lost. However, if you are unsure wh bibliotcore-33685b43555e json allows access to your cloud resources, so store it securely. Learn more
	Roles	for now.
Ξ	Audit Logs	Keys
	Ť	Key (0 202651-402552-7316-40-70-40-40-470-40-40-470-40-40-470-40-40-40-40-40-40-40-40-40-40-40-40-40
		+ CREATE KEY
		DOWS CAMPE
		DURE CARCEL
	1.2	
<1		
	btibiotcore-33685json ^	

8. Type this commands to generate a public/private keys.

```
openssl req -x509 -newkey rsa:2048 -keyout rsa_private.pem -nodes -days 36500 -out rsa_cert.pem -
subj "/CN=unused"
```

```
openssl pkcs8 -topk8 -inform PEM -outform DER -in rsa_private.pem -nocrypt > rsa_private_pkcs8
```

These commands will generate 3 files we will need 2 of them:

- rsa_cert.pem.
- rsa_private_pkcs8.

Connector Setup

Connection setup

1. Go to the GooglelotCore palette and drag and drop the connector to BtibService ExterbalConnectors RealtimeConnector

11 O 🗵 🔇 My Network	-	Name	Enabled	Status	Fault Cause	Last Attempt	Last Success	Last Failure
PlatformServices	^	GoogleIoTCoreConnector	false	{ok}		null	null	null
👻 🔥 BtibService		1						
Event		/						
🕨 膧 Time Range Definition								
Px Properties								
Log								
Relection								
Resource								
🕨 💱 Api Controller								
🕨 👗 Model								
Propagation								
E Strategy								
📼 🎉 External Connectors								
🕨 🎉 Time Series Connector								
🕨 🎉 Realtime Connector								
🕨 🏅 Model Connector								
🕨 🚶 File Connector								
D L L L								
* Palette								
🖿 🗶 🖉 🛔 btibGoogleIoTCore	-							
▶ 🍿 GoogleloTCoreConnector								
Extensions								
Alarm								

2. Copy the keys to your station



3. Fill the connection information

=	Google Cloud Platform StibloTCore -			Q Search products	and resources
ŵ	IoT Core Registries CREATE F	EGISTRY			
Ŧ	Filter registries				
	Registry ID 🕇		Region	Protocol	Telemetry Pub/Sub topics
	btibtest		europe-west1	MQTT, HTTP	projects/btibiotcore/topics/events
	udmi-registry		europe-west1	MQTT, HTTP	projects/btibiotcore/topics/udmi-telemetry
Pr	roperty Sheet				
ŵ	GoogleIoTCoreConnector (Goog	le Io T Core Connect	tor)		
•	\ominus Log Ext	System Log Ext			
	🗎 Status	{ok}			
	Fault Cause				
	Enabled Enabled	🛑 false 🗸 🗸			
	🗎 Last Attempt	null			
	🗎 Last Success	null			
	🗎 Last Failure	null			
	🗎 Use Data Retention	🛑 false 🔍 🗸			
	🗎 Data Retention Duration	+00072h 00m 00s	a a		
	🗎 Data Send Retry Duration	+00000h 01m 00s			
	` Messages Thread Pool Size	5			
•	🖑 Advanced Config	Advanced Config			
	🗎 Commands Topic	Commands Topic	-		
	📔 State Update Delay	1000			
	Project Id 📔	btibiotcore			
	🗎 Registry Id	udmi-telemetry			
	🗎 Region	europe-west1			
	📔 Service Account Json	•••••			
	Public Key	file:^rsa_cert.p	em		•
	Private Key	file:^rsa_privat	e_pkcs8		-

4. Enable the connector you should see the last attempt and last success values populated.

Property Sheet	
n GoogleIoTCoreConnecto	r (Google Io T Core Connector)
Log Ext	System Log Ext
🗎 Status	{ok}
Fault Cause	
Enabled	🕒 true 🔽
🗎 Last Attempt	18-May-2020 11:00 AM CEST
Last Success	18-May-2020 11:00 AM CEST

UDMI Setup

UDMI specification uses custom format for messages exchanged between the cloud and the device

Device Telemetry schema:

- Schema https://github.com/faucetsdn/daq/blob/master/schemas/udmi/pointset.json
- Example:

```
{
   "version": 1,
   "timestamp": "2018-08-26T21:39:29.364Z",
   "points": {
        "reading_value": {
            "present_value": 21.30108642578125
        }
   }
}
```

Device State:

- Schema https://github.com/faucetsdn/daq/blob/master/schemas/udmi/state.json
- Example:

```
{
  "version": 1,
  "timestamp": "2018-08-26T21:39:29.364Z",
  "system": {
    "make_model": "ACME Bird Trap",
   "firmware": {
     "version": "3.2a"
   },
    "last_config": "2018-08-26T21:49:29.364Z",
    "operational": true,
    "statuses": {
     "base_system": {
       "message": "Tickity Boo",
       "category": "device.state.com",
        "timestamp": "2018-08-26T21:39:30.364Z",
        "level": 600
     }
   }
  },
  "pointset": {
    "points": {
      "return_air_temperature_sensor": {
       "units": "Celsius",
        "status": {
         "message": "Invalid sample time",
         "category": "device.config.validate",
         "timestamp": "2018-08-26T21:39:28.364Z",
          "level": 800
       }
      }
```

} } }

Device Commands

- Schema https://github.com/faucetsdn/daq/blob/master/schemas/udmi/config.json
- Example:

```
{
  "version": 1,
  "timestamp": "2018-08-26T21:39:29.364Z",
  "system": {
    "min_loglevel": 500
 },
 "gateway": {
    "devices": {
      "AHU-123": {
               "protocol": "bacnet",
        "local_id": "327412"
      }
    }
 },
  "pointset": {
    "points": {
      "nexus_sensor": {
        "fix_value": 21.1
      }
   }
 }
}
```

1. Go to the advanced config slot on the connector.



2. On the message value template put the udmi template below

```
{
   "version": 1,
   "timestamp": "$(timestamp)",
   "points": {
        "$(pointId)": {
            "present_value": $(pointValue)
        }
   }
}
```

}

```
3. On the state template use.
```

```
{
 "version": 1,
  "timestamp": "$(timestamp)",
  "system": {
   "make_model": "$(deviceName)",
    "firmware": {
     "version": "1"
   },
    "statuses": {
     "base_system": {
       "message": "$(deviceStatus)",
       "category": "device.state",
       "timestamp": "$(timestamp)"
     }
   }
  },
  "pointset": {
   "points": {
      "$(pointId)": {
       "status": {
         "message": "$(pointStatus)",
         "category": "point.state",
         "timestamp": "$(timestamp)"
       }
     }
   }
 }
}
                             {
```

4. For commands we use this template and SFormat to extract the json information needed.

```
{
   "version": 1,
   "pointset": {
      "points": {
        "nexus_sensor": {
            "value": 21.1
        }
      }
   }
}
```

Command Device Id		()
🗎 Command Point Id	<pre>{json('pointset.points.firstKey')}</pre>	1
Command Action	OVERRIDE	1
Command Value	<pre>{json('pointset.points.firstKeyValue.value')}</pre>	0
Command Duration	0	0

Testing configuration

1. Create a new device and add a device ext.



2. On the device registry verify that the device provisioned successfully.

Registry ID: udmi-registry

europe-west1

Devices are things that connect to the internet directly or through a gateway. Learr

Ŧ	Enter exact device ID
	Device ID
	TestDevice_p0RzKdjc3YrALoHVaiQqF

Cloud IoT Core documentation

- Nav	-2	Property Sheet			
AL C X My Network		GoogleIoTCorePointExt (Google Io	T Core Point Ext)		
-		Advanced Config	Advanced Config		
▶ 🖳 Bacnet Comm		🗎 Status	{ok}		
Monitor		Fault Cause			
X Tuning Policies		Enabled	🔵 true 🔍		
TestDevice		Device Query	<pre>slot: neql:traverse n:parentDevice-></pre>	-	• (
Points		🗎 Trigger On Value Change Only	🔵 true 🔍		
√ (N) testPoint		谢 Can Write	🔵 true 🔍		
Config Miggielo TCoreDeviceExt Miggielo TCo					
	-				
Palette Solution Solution					
GoogleloTCoreConnector					
Extensions					
GoogleloTCoreDeviceExt					
WW Googlelo I CorePointExt					

4. To view messages we should create a subscription to the telemetry and state topics. On the Google console go to the **pubsub service subscription** create.



5. Give the subscription a name then choose the telemetry topic and click on Create.

*	Pub/Sub	← Create subscription					
P	Topics	A subscription directs messages on a topic to subscribers. Messages can be pushed to subscribers immediately or subscribers can cull message as needed.					
=	Subscriptions	Subscription ID *					
0	Snapshots	udmi-telemetry-sub					
		Subscription name: projects/btibiotcore/subscriptions/udmi-telemetry-sub					
		Select a Cloud Pub/Sub topic * projects/btiblotcore/topics/udmi-telemetry					
		Delivery type 🝘					
		Pull					
		O Push					
		Subscription expiration @					
		O Expire after this many days of inactivity (up to 365)					
		A subscription is inactive if there is no subscriber activity such as open connections, active pulls, or successful pushes.					
		Never expire					
		The subscription will never expire no matter the activity.					
		Acknowledgement deadline					
		Deadline time is from 10 seconds to 600 seconds					
		10 Seconds					
		Message retention duration @					
		Duration is from 10 minutes to 7 days Plane Hours Minutes					
		Retain acknowledged messages					
		When enabled, acknowledged messages are retained for the message retention duration specified above. This increases message storage fees. Learn more					
		apeanea asore. The nervases neerge arrage read, <u>secont nerv</u>					
		Dead lettering					
		Enable dead lettering Subscriptions may configure a maximum number of delivery attempts. When a message					
		cannot redelivered, it is republished to the specified dead letter topic.					
		CREATE					

6. Do the same for the state topic subscription

-	Pub/Sub	Create subscription						
	Topics	A subscription directs messages on a topic to subscribers. Messages can be pushed t						
-	Subscriptions	subscribers immediately, or subscribers can pull messages as needed.						
_		Subscription ID *						
0	Snapshots	Subscription parties in this interver (automatications (using laters out)						
		Delana elevit Del Del ante el a						
		projects/btibiotcore/topics/udmi-state						
		Delivery type						
		Pull						
		O Push						
		Subscription expiration @						
		 Expire after this many days of inactivity (up to 365) 						
		A subscription is inactive if there is no subscriber activity such as open connections, activ						
		public or successful pushes.						
		pulls, or successful pushes.						
		puil, of source provides. (e) Never explore The subscription will never explore on matter the activity. Acknowledgement deadline Describe transition form 10 accorded to 500 accord.						
		 Public of acceleration Public of the subscription will never expire no matter the activity. Acknowledgement deadline Deadline time is from 10 seconds to 600 seconds Seconds 						
		Point of the state of the						
		puid, of a constraints.						
		Puil, of a cost produce. Prove experiment of the subscription will never expire no matter the activity. Acknowledgement deadline Deadline time is from 10 seconds to 600 seconds To Seconds Message retention duration Cost Seconds Duration is from 10 minutes to 7 days						
		Power conference of the states. New exactly The subscription will never expire no matter the activity. Acknowledgement deadline Deadline time is from 10 seconds to 600 seconds Deadline time is from 10 seconds to 600 seconds Message retention duration Deadline is from 10 minutes to 7 days Deadline is from 10 minutes to 7						
		Point of the states. New early The subscription will never expire no matter the activity. Acknowledgement deadline Deadline time is from 10 seconds to 600 seconds 10 Seconds Message retention duration Duration is from 10 minutes to 7 days Duration is from 10 minutes to 7 days Deadline time story with the story of the stor						
		Pundi, or your contract. New exactly The subscription will never expire no matter the activity. Acknowledgement deadline Desdine time is from 10 seconds to 600 seconds Desdine time is from 10 seconds to 600 seconds Message retention duration Duration is from 10 minutes to 7 days Dur						
		Puil, or verte scale Never scale The subscription will never expire no matter the activity. Acknowledgement deadline Desdine time is from 10 seconds to 600 seconds 10 Seconds Message retention duration Duration is from 10 minutes to 7 days Duration is from 10 minutes to 7 days Comparing the state of the second						
		Puid, or produce. Prove scale The subscription will never expire no matter the activity. Acknowledgement deadline @ Deadline time is from 10 seconds to 600 seconds Descline time is from 10 seconds to 600 seconds Descline time is from 10 seconds to 600 seconds Descline time is from 10 seconds to 600 seconds Descline time is from 10 seconds to 600 seconds Descline time is from 10 minutes to 7 days Descline time acknowledged messages are retained for the message retention duration secolified above. This increases message storage frees. Learn more Dead lettering						
		Pure reaction Were reaction The subscription will never expire no matter the activity. Acknowledgement deadline Dessline time is from 10 seconds to 600 seconds Dessline time is from 10 seconds to 600 seconds Message retention duration Duration is from 10 minutes to 7 days Duration is from 10 minutes to 7 days Duration is from 10 minutes to 7 days Message retention duration Duration is from 10 minutes to 7 days Message retention duration Duration is from 10 minutes to 7 days Message retention duration Duration is from 10 minutes to 7 days Message retention duration Duration is from 10 minutes to 7 days Duration is from 10 minutes to 7 days Duration is from 10 minutes to 7 days Message retention duration Duration is from 10 minutes to 7 days Duration is from 10 minut						
		pung, or configure of mainter in the activity.						
		Pure variable Provide a construction will never expire no matter the activity. Acknowledgement deadline Provide a construction of a seconds Deadline time is from 10 seconds to 600 seconds Message retention duration Provide a construction of a seconds Message retention duration Provide a construction of a seconds Message retention duration Provide a construction of a seconds Message retention duration Provide a construction of a seconds Message retention duration Provide a construction of a seconds Message retention duration Provide a construction of a seconds Message retention duration Provide a construction of a seconds Message retention duration Provide a construction of a seconds Message retention duration Provide a construction of a seconds Message retention duration						
		Pure reaction Provide the subscription will never expire no matter the activity. Acknowledgement deadline Deadline time is from 10 seconds to 600 seconds Desdine time is from 10 seconds to 600 seconds Desdine time is from 10 seconds to 600 seconds Dustion is from 10 minutes to 7 days Dustion is from 10 minutes Dustion is from 10						

7. Now go to the Niagara point and change the value and the state.

My Network	Property Sheet					
TestDevice	🕔 testPoint (Numeric Writabl	le)				
Alarm Source Info	Facets	units=null,precision=1,min=-inf,max=+inf 📎 🕓 🔹				
- Deints	Proxy Ext	null				
 testPoint 	— Out	- {null} 0 def				
Proxy Ext	In1	- {null}				
InfoSource	— In2	- (null) ¥ - (null) ¥ - (null) ¥ - (null) ¥ - (null) ¥				
GoogleIoTCorePointExt	- In3					
Virtual	— In4					
Alarms	— In5					
Schedules	— In6					
Trend Logs	💻 In' 🌉 Override	× •				
Config	Ini Override Value					
GoogleIoTCoreDeviceExt	- In: Override Duration Re	▼				
InfoSource	In:	¥				
Apps	— In	OK Cancel				
Station Info Source	- In12	- {null} ¥				

8. On the telemetry subscription you should see your message.

Ξ	Google Cloud Platform	S• BibloTCore 👻	Q Search products and re	Messages				
12	Pub/Sub	CREATE SNAPSHOT	③ REPLAY MESSAGES					
P	Topics	udmi-telemetry-sub		Click Pull to view mes Select Enable ACK m be pulled at a time. Cl	ssages and temporarily delay n essages and then click ACK ne lick Pull again to retrieve more	ressage delivery to other subscribers. at to the message to permanently prevent mess messages from the backlog. Use this option ca	sage delivery to other subscri autiously in production enviro	bers. Only a few messages w iments. If you miss the
-	Subscriptions			acknowledgement de	adline (10 seconds), the mess	age will be sent again it no other subscribers of	This subscription acknowled	ged the message. Learn more
8	Snapshots	Unacked message count		RALL Exclusion Exclusions				
				Filter table				0
				Publish time 🕇	Attribute keys	Message body	Body JSON keys	Ack
				May 18, 2020, 12:14:38 PM	deviceld	("version": 1, "timestamp": "2020-05-1	version simestamp	Deadline exceeded
		11.28 11.30 11.35 11.40 11.48 11.50 11.58 12.914 12.05	12:10 12:19 12:20	May 18, 2020, 12:18:54 PM	deviceId	("version": 1, "timestamp": "2020-05-1	version timestamp	Deadline exceeded
		Subscription name projectu ptolocio u uzborojsticonu udmi-skemetry-sub. 10		May 18, 2020, 12:19:44 PM	deviceId deviceNumid deviceRegistryId deviceRegistryLocation projectid	('version': 1, 'timestamp': '2020-05- 18712:10:44.0624:02.00', 'points': ('testPoint_iORX6cs:bogeC4iomsJeWQ': ('peasent_value': 11.00.).))	version timestamp points.testPoint_kOPpKeou	Deadline exceeded
		Topic name		4				
		projects/biblotone/topics/udmi-telemetry.						

9. The same for the state subscription

		Messages				
Pub/Sub	← Subscription details / EXT ● WEW MESSAGES	Click Pall to view me Select Enable ACK m	ssages and temporarily delay ressages and then click ADK r	message delivery to other subscribers. exit to the message to permanently prevent me	ssage delivery to other subscr	ibers. Only a few messages will
Subscriptions	udmi-state-sub	be pulled at a time. O acknowledgement d	lick Pull again to retrieve mon eadline (10 seconds), the mes	e messages from the backlog. Use this option sage will be sent again if no other subscribers	cautiously in production enviro of this subscription acknowled	nments. If you miss the lged the message. <u>Learn more</u>
Snapshots	Unakled message count	FULL Enable ack me	ssages			Θ
		Publish time	Attribute keys	Message body	Body JSON keys	Ack 🕈
		May 18, 2020, 12:14:38 PM	deviceId	("version": 1, "timestemp": "2020-05-1	version timestamp	Deadline exceeded
		May 18, 2020, 12:18:54 PM	deviceId	('version': 1, 'timestamp': '2020-05-1	version timestamp	Deadline exceeded
	In the part part property process that the part of the part process p	May 18, 2020, 12:19:44 PM	deviced deviced deviceSegativid deviceSegativid deviceSegativid deviceSegativid deviceSegativid deviceSegativid	('version'), 1'metaterity' 2020-5- 1911;19:44.01-6200', 'vysterit, (' metat.model', Testfolevick, Timmare', ('version'), 1'statese? ("boas,systerit,' ('message', 'mil, 'statese? ('workshi', 'mestage', 'mil, 'statese? ('workshi', 'mestage', 'mil, 'statese?', 'workshi', 'mestage', 'mil, 'statese?', 'workshi', 'mestage', 'mil, 'statese?', 'workshi', 'mestage', 'mil, 'statese?', 'statese?', 'mil, g?, ' 'statese?', 'statese?',	version Smertamp system Smeare version 9900m cales back, sy portext control takes, py portext control takes, py portext control takeborr, portext control takeborr, portext control takeborr,	Deadline exceeded
	Subocription expiration Subocription will never aspin.	-				

Congratulations you finished the tutorial for more information check the connector documentation.