Google Cloud

API BaaS

Query Language

Proprietary + Confidential

Query language basics

There are 2 parts to the query language used for BaaS API requests:

Path to the collection – `https://apibaas-trial.apigee.net/<org>/<app>/users`

Query language statement – after ?ql=

Guidelines:

Any values specified in the query must be enclosed in single-quotes

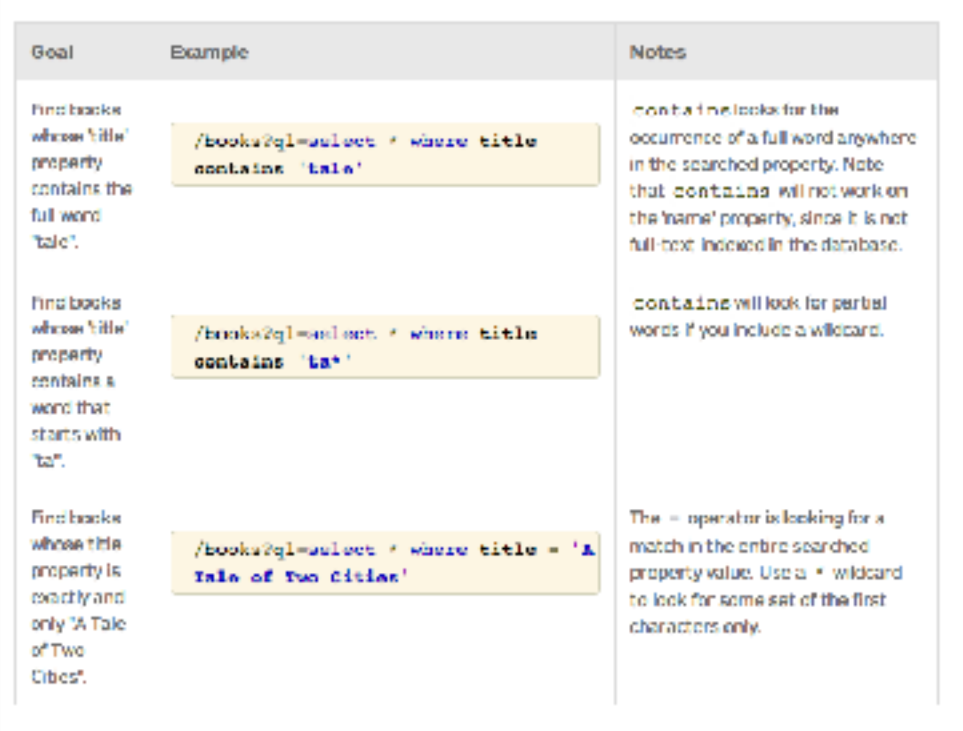
No more than 3 parameters in your queries to achieve optimal performance.

If your query starts with select * where, you can omit that part of the statement and just use the condition in the query.

Example:

```
https://apibaas-trial.apigee.net/your-org/your-app/users?limit=5&ql=status='active' and age lt 40 order
    by name asc
```

Query language examples



Goal	Example	Notes
Find books whose 'title' property contains the full word 'tale'.	`/books?ql=select * where title contains 'tale'`	`contains` looks for the occurrence of a full word anywhere in the searched property. Note that `contains` will not work on the 'name' property, since it is not full-text indexed in the database.
Find books whose 'title' property contains a word that starts with 'ta'.	`/books?ql=select * where title contains 'ta*'`	`contains` will look for partial words if you include a wildcard.
Find books whose title property is exactly and only 'A Tale of Two Cities'.	`/books?ql=select * where title = 'A Tale of Two Cities'`	The `=` operator is looking for a match in the entire searched property value. Use a `*` wildcard to look for some set of the first characters only.

Order By, Limit & Cursor

Order by - You can return query results that are sorted in the order you specify. Use the order by clause to specify the property to sort by, along with the order in which results should be sorted. The syntax for the clause is as follows:

```
order by <property_name> asc | desc
```

Limit - When your query might return more results than you want to display to the user at once, you can use the limit parameter with cursors or API methods to manage the display of results. By default, query results are limited to 10 at a time. You can adjust this by setting the limit parameter to a value you prefer. Max number is 1000.

Cursor - An encoded representation of the query position pointing to a set of results. To retrieve the next set of results, pass the cursor with your next call for most results.

Matrix queries

Matrix Queries allow you to use embedded query strings in the URL.

These are Needed if you want to use multiple filters at multiple levels.

This becomes very handy for filtering connected entities.

Example:

```
/users/ed/friends;ql=“location eq new york”/achievements?ql="level eq mayor”
```

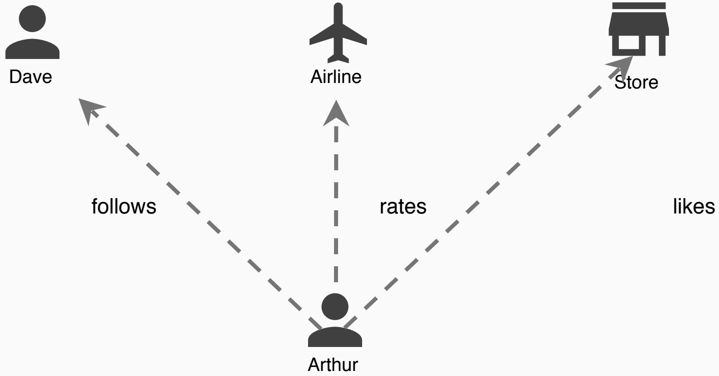
Entities and relationships

- Relationships: typically a verb could be used to create a connection between any two entities.

- Entities not necessarily be of the same type (user, store etc.)

- Relationships are unidirectional

Example: The “likes” relationship from Arthur to Store, does not create a “liked by” relationship from the Store towards Arthur automatically.



Entities & relationships

Entities are connected using relationships

To create a connection, the entity being connected to can be specified by

- UUID
- Collection and name property

```
curl -X POST https://apibaas-trial.apigee.net/<org>/<app>/<connecting_collection> /
<connecting_entity>/<relationship>/<connected_collection>/<connected_entity>
```

To retrieve a connection

```
curl -X GET https://apibaas-trial.apigee.net/<org>/<app>/<collection>/<entity>/<relationship>
```

Assets

Asset can be any type of file. It can be used to hold data objects like image, text, video or audio.

An asset can be uploaded to a collection

An asset can also be associated with an entity. Only 1 asset can be attached to an entity.

To create a collection *items* with 2 entities milk and bread

```
curl -X POST "https://apibaas-trial.apigee.net/
<org>/sandbox/items" -d '[{"name":"milk",
"price":"3.25"}, {"name":"bread",
"price":"2.50"}]'
```

To assign an image bread.jpeg to an entity *bread*

```
curl -X POST -F name='bread.jpeg' -F
file=@bread.jpeg 'https://apibaas-
trial.apigee.net/<org>/sandbox/items/bread
```

To retrieve an image, check the content-type on the entity’s file-metadata

```
curl -X GET -H 'Accept: image/jpeg' "https://
apibaas-trial.apigee.net/<org>/sandbox/items/
bread"
```

Assets

Assets can also be organized into folders. For example all item images can be linked to a folder “images”

To create a folder “images”:

```
          curl -X POST 'https://apibaas-trial.apigee.net/<org>/sandbox/folders' -d '{"name":
"images", "owner": ”<valid user uuid>","path": "/images"}’
```

To assign an asset to a folder:

```
          curl -X POST 'https://apibaas-trial.apigee.net/<org>/sandbox/folders<folder uuid>/
assets/<asset uuid>'
```

Counters

Counters can be used to track statistics for an application.

They can be incremented/decremented by any value or reset to 0.

To create a counter poll.yes and increment by 1:

```
curl -X POST https://apibaas-trial.apigee.net/<org>/sandbox/events -d '{"timestamp":0,
"counters" : { "poll.yes":1}}'
```

Timestamp is specified as UNIX timestamp. A value of 0 assigns the current time

It may take up to 30 seconds for the counter value to be updated.

To retrieve the value of the counter:

```
curl -X GET https://apibaas-trial.apigee.net/<org>/sandbox/counters?counter=poll.yes
```

Counters

Counters can be created and incremented hierarchically using dot notation.

e.g. poll, poll.yes, poll.no

Total number of people who took the poll:

```
curl -X GET https://apibaas-trial.apigee.net/<org>/sandbox/counters?counter=poll
```

Calling API BaaS from Edge

Service callout or Target endpoint – Sample flow:

- Pull client credentials out of KVM (you’re not even thinking of storing them in code, right? ☺)
- Pull API BaaS URL out of KVM to parameterize between prod, test, dev, etc.
- Use HTTPS service callout to get the token using the credentials
- Store token in an object cache (you may need to reuse it for other API BaaS calls)
- Then use HTTPS service callout to retrieve data from API BaaS
- Extract the data from API BaaS response

For more complex sequences

- script callout in JS, Java, Python

Could also use Node.js

THANK YOU