Hash Map as a Service

In Service broker two parts has involved.

1. Hash broker

2.Hash Client to consume the Has Broker Service.

Deploy a custom service broker.

1. The `cf-haash-broker` implements a Hash Map as a Service

2. Deploy the Service Broker as an app to Cloud Foundry

3. Register the Broker with the Cloud Controller

4. Make the single plan in the catalog “public”

5. Create an instance of your service6. Push and bind a test app to your service instance

7. Test the app’s interaction with the service

- Push the broker as an app to Cloud Foundry using `cf`
- Use the name: `<first-initial><last-initial>-haash-broker`
- The file you need to deploy is `build/libs/haash-broker-0.0.1-SNAPSHOT.jar`.
- Register the broker with the cloud controller.
- You must first login as the admin user to run this command, using the password from the Elastic Runtime tile in Ops Manager,
- from the credentials tab for UAA/admin.
- For the service broker name, use your initials <first initial><last initial>-broker.
- The username & password you need to create the service broker are in: `src/main/resources/application.properties`
- Next, you need to make the plan public: You will need to figure out the service and plan names.
- You can do this by running the following command:
- Verify you can see your new service in the marketplace.

- There is a `cf` command for this.
- Create a new instance of your service using the CF commands.
- Name the instance `Murali-Tirupati-haash-service`.
- Running the Haash Client

```
public class DashboardClient {

private final String id = "Haash-XX";

private final String redirectURI = "http://example.com";

private final String secret = "secret";
```

- Change into the `cf-haash-client` directory.
- There is no need to build the Haash Client application. It is already built for you and the jar you need to deploy is in the `pre-built` directory.
- Create a manifest for your Haash Client.
- Be sure to bind the Haash Client to the service instance you created above.
- Push the Haash Client to Cloud Foundry and name it: `Murali—Tirupati-haash-client`
- To access your client you need to be able to send HTTP GET and PUT requests.
- Access your haash client to be sure it is working and using your haash-service •
- Using `curl`, you can add key,value pairs using:

```
$ curl <your haash-client>/HaaSh/<key> -d '<some value>' -X PUT
```

• Now retrieve the values:

```
$ curl <your haash-client>/HaaSh/<key>
```