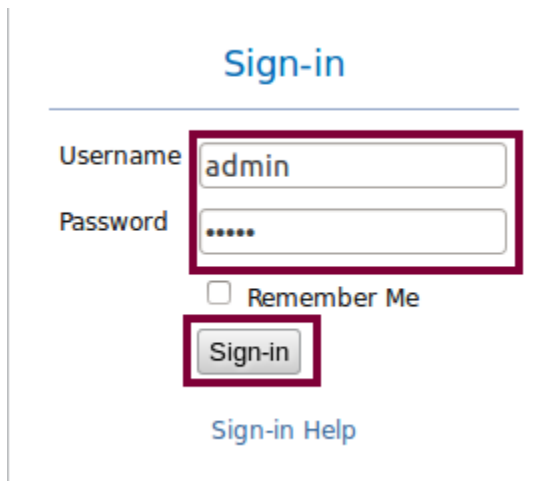


Following tutorial will guide you through basic configurations of Wihidum CEP in a step by step process.

## Create Agent Broker

Before creating the bucket to filter stock quotes it is essential to have a Carbon broker. Since In this example we are going to use agent Broker, it is needed to create a broker with type agent. To do that

1. Start CEP Server.
2. Login as admin.

A screenshot of a web-based sign-in form. At the top, the text "Sign-in" is displayed in blue. Below it, there are two input fields: "Username" with the text "admin" and "Password" with masked characters "\*\*\*\*\*". A checkbox labeled "Remember Me" is positioned below the password field. A "Sign-in" button is located below the checkbox. At the bottom of the form, there is a link labeled "Sign-in Help". The entire form is enclosed in a light gray border.

1. In the Configure menu you can find a Menu item called "Broker" and under that you can see sub menu 'Add' and click on that.
2. You will get a page with header "Create a New Broker" and you need to enter following details in that form to create a agent broker.

Broker Name : localAgentBroker

Type	: agent
URL	: tcp://localhost:7611
Authenticator URL	: ssl://localhost:7711
User Name	: admin
Password	: admin

3. Finally click on Add Broker button and you will get the added broker to the list of available brokers.

Signed-in as: admin@localhost | [Sign-out](#) | [Docs](#) | [About](#)

Home > Configure > Broker > Add

### Create a New Broker

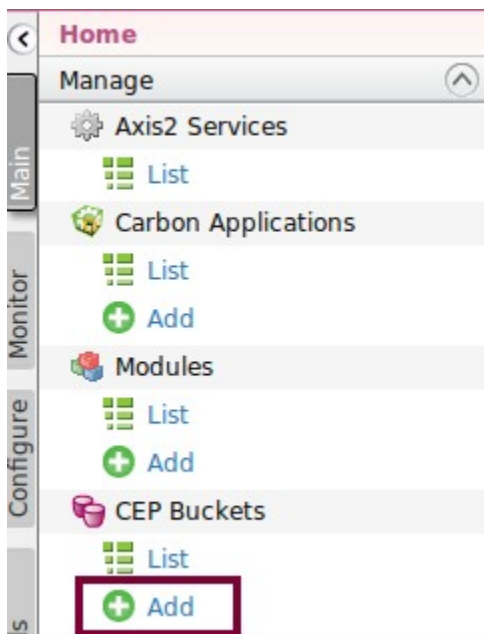
Enter Broker Details

Broker Name*	localAgentBroker
Broker Type*	agent
Receiver URL*	tcp://localhost:7611
Authenticator URL	ssl://localhost:7711
User Name*	admin
Password*	*****

Add Broker

## Create Bucket with Wihidum

To create a bucket use Add menu item under CEP Buckets in the Main menu. Bucket creation form has three major sections. Basic information, Input and query and loadbalancer. How to fill those sections is described below.



## Section 1 : Basic Information

Use the following information to fill the basic information section as shown in the below screenshot.

Bucket Name (Name of the bucket) : KPIAnalyzer  
Description (Description about the bucket) : Notifies when a user purchase more than 3 items  
Engine Provider(CEP Runtime engine to be used) : SiddhiCEPRuntime [Choose from the drop down]

Persistence snapshot time interval in minutes : 0  
Enable distributed processing : true

**Add Bucket**

**Bucket Information**

Bucket Name\* : KPIAnalyzer

Bucket Description : Notifies when a user purchase more than 3 items

CEP Runtime\* : SiddhiCEPRuntime

**Backend Runtime Configuration**

Persistence snapshot time interval in minutes : 0

Enable distributed processing : true

## Section 2 : Inputs

This section is used to define the inputs CEP will receive. To add an input click on Add Input link and then use following details. Screen shot is provided below for your convenience.

Topic( topic to events be received): phone.retail.store/1.2.0  
Broker Name (Broker to be used) : localAgentBroker

### Mapping

Stream (Name of the event stream) : phoneRetailStream  
Query Event Type : Tuple  
Input Mapping Type : Tuple Mapping

Properties (these properties will be extracted from the received tuple event and fed to the CEP engine)

Name : brand

Input Name : brand  
Input Data Type : payload data  
Type : String

Name : quantity  
Input Name : quantity  
Input Data Type : payload data  
Type : Integer

Name : totalPrice  
Input Name : totalPrice  
Input Data Type : payload data  
Type : Integer

Name : buyer  
Input Name : buyer  
Input Data Type : payload data  
Type : String

 Add Input

Add Input

Topic Name \*

phone.retail.store/1.2.0

Broker Name

localBroker

Mapping

Stream\*

phoneRetailStream




Query Event Type

Tuple

Input Mapping Type

Tuple Mapping

Properties

Name	Input Name	Input Data Type	Type	Actions
brand	brand	payloadData	java.lang.String	 Delete
quantity	quantity	payloadData	java.lang.Integer	 Delete
totalPrice	total	payloadData	java.lang.Integer	 Delete

Name: buyer

Input Name: buyer

Input Data Type: Payload Data

Type: String

Add

Add Input

## Section 3 : Queries

This section is used to define the queries which will run on inputs and define outputs. To add a query click on Add query link and use following information. Screen shot is provided below for your convenience.

Query Name (To identify the query) : KPIQuery

Expression : from phoneRetailStream[quantity > 3]  
                  insert into highPurchaseStream  
                  buyer, brand, quantity, totalPrice;

Query Deploying IP List

IP : 192.168.1.2

Output(Define the output)

Topic : high.purchase.buyers/1.5.0  
Broker Name : 192.168.1.2  
Output Mapping : Tuple Mapping

Tuple Mapping

Meta Data

Name : buyer	value of : buyer	Type : String
--------------	------------------	---------------

Payload data

Name : brand	value of : brand	Type : String
Name : quantity	value of : quantity	Type : Integer
Name : purchasePrice	value of : totalPrice	Type : Integer

[+ Add query](#)

**Query**

Query name\*

KPIQuery

Expression\*

```
from phoneRetailStream[quantity > 3]
insert into highPurchaseStream
buyer, brand, quantity, totalPrice;
```

**Query Deploying IP List**

IP	Actions
192.168.1.2	Delete

IP\*

Add

**Output**

Topic\*

high.purchase.buyers/1.5.0

Broker Name

192.168.1.2

Output Mapping

Tuple Mapping

**Tuple Mapping**

Meta Data

Name	Value Of	Type	Actions
buyer	buyer	java.lang.String	Delete

Name :

Value Of :

Type:

String

Add

Correlation Data

No Correlation Data properties Defined

Name :

Value Of :

Type:

Integer

Add

Payload Data

Name	Value Of	Type	Actions
brand	brand	java.lang.String	Delete
quantity	quantity	java.lang.Integer	Delete
totalPrice	totalPrice	java.lang.Integer	Delete

Name :

Value Of :

Type:

Integer

Add

Add Query

## Section 3 : LoadBalancer configurations


This section is used to define load balancer configurations. To add a configuration click on Add LB Configuration link and use following information. Screen shot is provided below for your convenience.

LoadBalancer IP : 192.168.1.4


Output Nodes


IP : 192.168.1.2 Port : 9443

### LoadBalancer Configuration

Select all in this page | Select none  Delete

No LB Config Defined


Select all in this page | Select none  Delete


 Add LB Configuration

Loadbalancer

LoadBalancer IP\*

Output Nodes

IP	Port	Actions
192.168.1.2	9443	 Delete

IP\*  Port\*  

Add LB Configuration

Save

Cancel

Using above instructions you can create and save a bucket.