

Notification

Table of contents

1 Notification for a New Partition.....	2
2 Notification for a Set of Partitions.....	3

In HCatalog 2.0 we introduce notifications for certain events happening in the system. This way applications such as Oozie can wait for those events and schedule the work that depends on them. The current version of HCatalog supports two kinds of events:

- Notification when a new partition is added
- Notification when a set of partitions is added

No additional work is required to send a notification when a new partition is added: the existing `addPartition` call will send the notification message. This means that your existing code, when running with 0.2, will automatically send the notifications.

1 Notification for a New Partition

To receive notification that a new partition has been added, you need to follow these three steps.

1. To start receiving messages, create a connection to a message bus as shown here:

```
ConnectionFactory connFac = new ActiveMQConnectionFactory(amqurl);
Connection conn = connFac.createConnection();
conn.start();
```

2. Subscribe to a topic you are interested in. When subscribing on a message bus, you need to subscribe to a particular topic to receive the messages that are being delivered on that topic.

- The topic name corresponding to a particular table is stored in table properties and can be retrieved using following piece of code:

```
HiveMetaStoreClient msc = new HiveMetaStoreClient(hiveConf);
String topicName = msc.getTable("mydb",
    "myTbl").getParameters().get(HCatConstants.HCAT_MSGBUS_TOPIC_NAME);
```

- Use the topic name to subscribe to a topic as follows:

```
Session session = conn.createSession(true, Session.SESSION_TRANSACTED);
Destination hcatTopic = session.createTopic(topicName);
MessageConsumer consumer = session.createConsumer(hcatTopic);
consumer.setMessageListener(this);
```

3. To start receiving messages you need to implement the JMS interface `MessageListener`, which, in turn, will make you implement the method `onMessage(Message msg)`. This method will be called whenever a new message arrives on the message bus. The message contains a partition object representing the corresponding partition, which can be retrieved as shown here:

```

@Override
public void onMessage(Message msg) {
    // We are interested in only add_partition events on this table.
    // So, check message type first.

    if(msg.getStringProperty(HCatConstants.HCAT_EVENT).equals(HCatConstants.HCAT_ADD_PARTITION_EVENT))
    {
        Object obj = (((ObjectMessage)msg).getObject());
    }
}

```

You need to have a JMS jar in your classpath to make this work. Additionally, you need to have a JMS provider's jar in your classpath. HCatalog uses ActiveMQ as a JMS provider. In principle, any JMS provider can be used in client side; however, ActiveMQ is recommended. ActiveMQ can be obtained from: <http://activemq.apache.org/activemq-550-release.html>

2 Notification for a Set of Partitions

The example code below illustrates how to send a notification when a set of partitions has been added.

```

HiveMetaStoreClient msc = new HiveMetaStoreClient(conf);

// Create a map, specifying partition key names and values
Map<String,String> partMap = new HashMap<String, String>();
partMap.put("date", "20110711");
partMap.put("country", "*");

// Mark the partition as "done"
msc.markPartitionForEvent("mydb", "mytbl", partMap, PartitionEventType.LOAD_DONE);

```

To receive this notification, the consumer needs to do the following:

1. Repeat steps one and two from above to establish the connection to the notification system and to subscribe to the topic.
2. Receive the notification as shown in this example:

```

HiveMetaStoreClient msc = new HiveMetaStoreClient(conf);

// Create a map, specifying partition key names and values
Map<String,String> partMap = new HashMap<String, String>();
partMap.put("date", "20110711");
partMap.put("country", "*");

// Mark the partition as "done"
msc.markPartitionForEvent("mydb", "mytbl", partMap, PartitionEventType.LOAD_DONE);

```

If the consumer has registered with the message bus and is currently live, it will get the callback from the message bus once the producer marks the partition as "done".

Alternatively, the consumer can ask explicitly for a particular partition from the metastore. The following code illustrates the usage from a consumer's perspective:

```
// Enquire to metastore whether a particular partition has been marked or not.
boolean marked = msc.isPartitionMarkedForEvent("mydb", "mytbl", partMap,
    PartitionEventType.LOAD_DONE);

// Or register to a message bus and get asynchronous callback.
ConnectionFactory connFac = new ActiveMQConnectionFactory(amqurl);
Connection conn = connFac.createConnection();
conn.start();
Session session = conn.createSession(true, Session.SESSION_TRANSACTED);
Destination hcatTopic = session.createTopic(topic);
MessageConsumer consumer = session.createConsumer(hcatTopic);
consumer.setMessageListener(this);

public void onMessage(Message msg) {

    MapMessage mapMsg = (MapMessage)msg;
    Enumeration<String> keys = mapMsg.getMapNames();

    // Enumerate over all keys. This will print key value pairs specifying the particular
    // partition
    // which was marked done. In this case, it will print:
    // date : 20110711
    // country: *

    while(keys.hasMoreElements()){
        String key = keys.nextElement();
        System.out.println(key + " : " + mapMsg.getString(key));
    }
    System.out.println("Message: "+msg);
}
```

Notification is enabled by default. To disable notification, you need to leave `hive.metastore.event.listeners` blank or remove it from `hive-site.xml`.