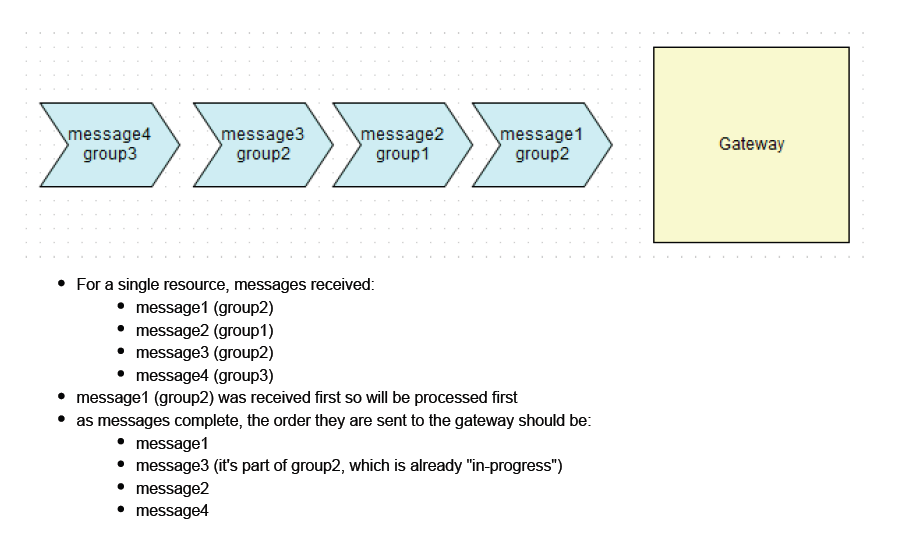
**Resource Scheduler Assignment**

This assignment is about arranging messages in group order and send to Gateway. The Group of messages will be sent first based on group which read first by the program as is depicted in following diagram.



**Configuration:**

To setup the number of resources I have created config file ResourceConfig.xml in config folder.

ResourceConfig contain location of resource which will be picked by program and sent to gateway.

For example in this project I have define four resource location where user can put file containing messages to deliver as mentioned below.

<Resources>

<Directory>resources/resources1</Directory>

<Directory>resources/resources2</Directory>

<Directory>resources/resources3</Directory>

<Directory>resources/resources4</Directory>

</Resources>

I have also set format of data which program will use to deliver messages.

Groupid|MessageId|Message|Status

2|1|GROUP2 Message1|

1|1|GROUP1 Message1|

3|1|GROUP3 Message1|

I have attached my sample data file. But as you can see based on requirement now all messages of group 2 will deliver first then group 1 and at last group 3.



So you just need to drop your file in any one of resource location and program will pick it process it and send to gateway in specified sequence.

**Development**:

There are four main component of this program.

**1: MessageLoader** this class reads files from location of resources mentioned in configuration file and for each resource I have created new thread.

**2:MessageContainer** this class is responsible for holding all information before it deliver to gateway. To contain information I used LinkedHashMap which is indexed by group id (key = groupid of message) where the list of messages are store in queue format as LinkedBlockingQueue.

Map<Long, MessageQueue> messageHolder = new LinkedHashMap<Long, MessageQueue>();

MessageQueue = LinkedBlockingQueue

The main reason of LinkedHashMap selection Is because its data structure is hash table and linked list implementation as in this problem we want to deliver the message based on their group sequence.

And the next of using LinkedBlockingQueue is to traverse the list of message as entered and as a queue that additionally supports operations that wait for the queue to become non-empty when retrieving an element, and wait for space to become available in the queue when storing an element.

**3.MessageSender** This class is responsible of sending message to gateway. It simply reads the messages from LinkedBlockingQueue in a same sequential order as these were inserted with help of LinkedHashMap.

**4:ResourceScheduler** This is the main class which kickoff loader and sender processes with the help of thread executor service. For simplicity I have selected newFixedThreadpool with pool of two thread one for each process.

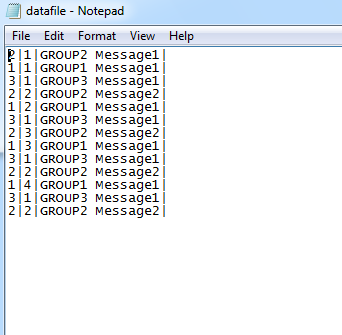
**Note:** Due to time constraint I only displayed the output on command prompt. And it will also delete feed file you put in resource folder so please keep copy of your original file.

**Build:**

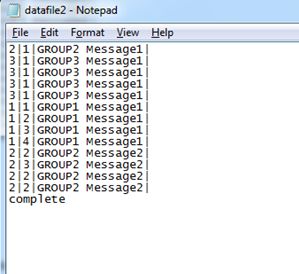
I have copied RescourceScheduler source into Src folder in with eclipse project to import in any eclipse environment where Exe folder contain executable jar which with all other configuration required.

**Execution Sample:**

Sample data in first file

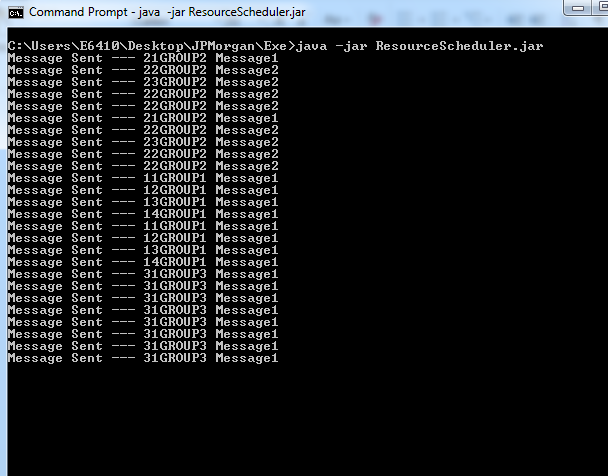


So the sequence of message should be group2 -> group1 -> group3

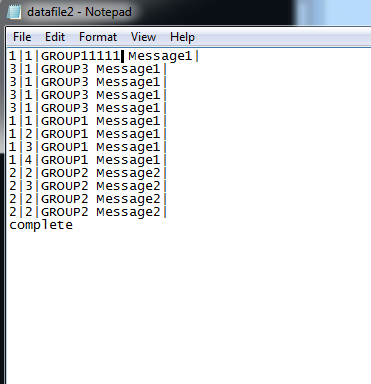


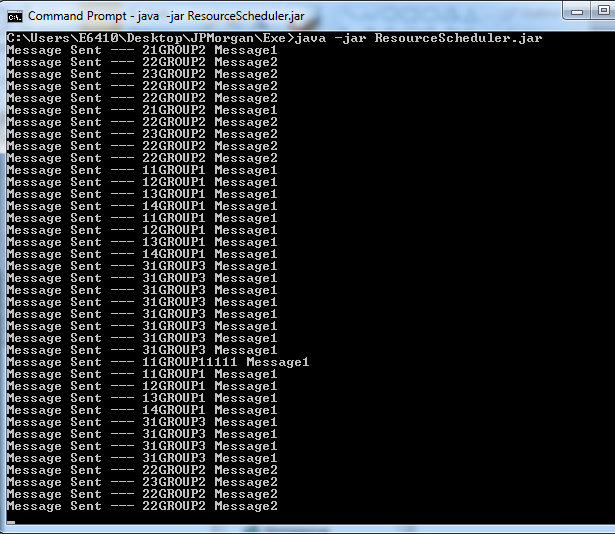
Here in second sample file we have group 2 -> group 3 -> group 1 sequence but because message is not complete program keeps the sequence of last resource until message is not complete.

So the sequence of message should be group2 -> group1 -> group3 but at the end as the message is complete program will clear the linked hash table records



Next time when user try it will not keep same sequence as shown below.





Hope you will like this.

Thanks