

JTC DATA Load

- 1) Purpose of the requirement: Load fact and dimension raw data to mongoDB database in order to produce aggregated reports.
- 2) Scope of the requirement:
 - 2.1 Load fact data in FactStore collection
 - 2.2 Load products data in DimProduct collection
 - 2.3 Load dates data in DimDates collection
 - 2.4 Load store data in DimStores collection
- 3) Tool Used
MongoDB , Talend, notepad
- 4) Constraints : The dimension data is given in sql script rather in csv or json ,
2 Approaches are thought of to overcome this ,
 - 4.1) we need to either prepare csv or json data by replacing/modifying text in notepad
 - 4.2) We can execute the script to any open source db(my sql in this case) and extract json files to load in MongoDB.

Note: approach 2 is taken to avoid special characters , date formatting issues.

5)Talend jobs

There are two job created in talend under folder loadDimAndFactDatatoMongo to achieve the objective

5.1)loadDimData

This job uses the tMongoDBBulkLoad component to load dimension data to mongo.

a)bulkLoadProduct

- Bulk loads the collection DimProduct in MongoDB

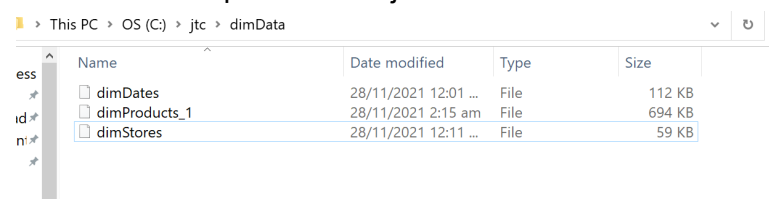
b)bulkDimDates

- Bulk Loads collection DimDates in MongoDB

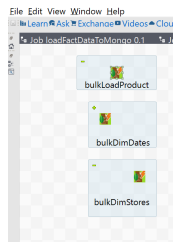
c)bulkDimStores

- Bulk loads collection DimSores in MongoDB

Source files are present in c:/jtc/dimDataFolder



Name	Date modified	Type	Size
dimDates	28/11/2021 12:01 ...	File	112 KB
dimProducts_1	28/11/2021 2:15 am	File	694 KB
dimStores	28/11/2021 12:11 ...	File	59 KB



5.2) loadfactdatatomongo

5.2.1 root_archive

FactStore raw files are present in c:/jtc/2009.zip folder and this module extract them to c:/jtc/2009

The directory are structured as year->month->weekdays inside c:/jtc/2009 folder and raw data files are also archived so we need to do one more unarchive process on the list of files

5.2.2 root_file_list

This component prepares the list of files under directory c:\2009 recursively

5.2.3 staging_unarchive

This component unarchive the file from c:/2009 and create same directory structure under c:\staging

5.2.4 fact_file_read

This component take the name form file from staging_unarchive and read the file and map it to schema

5.2.5 fact_file_write

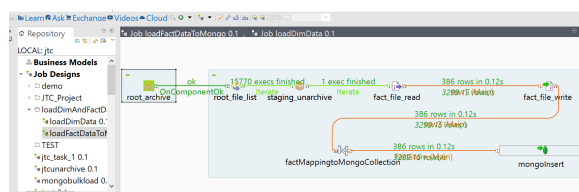
As per the requirement we need to write the pipe separated files in a directory \dataextract. This component read the line from previous process and create new | separated files with the same directory structure under c:\dataextract

5.2.6 factMappingtoMongoCollection

This component map the schema of file structure to mongo collection

5.2.7 mongoinsert

This component insert the lines to mongo collection FactStore



6)Mongo Queries

6.1) Quantity on hand by Product category

db.FactStore.aggregate

```
([{"$lookup":
  {"from": "DimProducts",
   "localField":"ProductID",
   "foreignField":"productid",
```

```

    "as": "avgOnHandQty_details"}},
    {
      "$unwind": "$avgOnHandQty_details"
    },
    {
      "$group": {
        "_id": {
          "productcategoryname": "$avgOnHandQty_details.productcategoryname"
        },
        "avgnHandQty": { "$avg": "$OnHandQty" } },
      $addFields:
        {round_avgnHandQty: {$round: ["$avgnHandQty"]}}
    })
  ).pretty()

```

output

```

{
  "_id" : {
    "productcategoryname" : "Cameras and camcorders "
  },
  "avgnHandQty" : 20.67926907486834,
  "round_avgnHandQty" : 21
},

/* 2 */
{
  "_id" : {
    "productcategoryname" : "Cell phones"
  },
  "avgnHandQty" : 41.16605977663188,
  "round_avgnHandQty" : 41
},

/* 3 */
{
  "_id" : {
    "productcategoryname" : "Computers"
  },
  "avgnHandQty" : 21.73863693874828,
  "round_avgnHandQty" : 22
},

/* 4 */
{
  "_id" : {
    "productcategoryname" : "Games and Toys"
  },

```

```

        "avgnHandQty" : 75.91057615846165,
        "round_avgnHandQty" : 76
    },

    /* 5 */
    {
        "_id" : {
            "productcategoryname" : "Home Appliances"
        },
        "avgnHandQty" : 20.28480982976129,
        "round_avgnHandQty" : 20
    },

    /* 6 */
    {
        "_id" : {
            "productcategoryname" : "TV and Video"
        },
        "avgnHandQty" : 20.413036681950814,
        "round_avgnHandQty" : 20
    },

    /* 7 */
    {
        "_id" : {
            "productcategoryname" : "Music, Movies and Audio Books"
        },
        "avgnHandQty" : 18.86934555493405,
        "round_avgnHandQty" : 19
    },

    /* 8 */
    {
        "_id" : {
            "productcategoryname" : "Audio"
        },
        "avgnHandQty" : 21.75158076962575,
        "round_avgnHandQty" : 22
    }
}

```

6.2) Quantity on hand by Store type

```

db.FactStore.aggregate([
    {"$lookup":
        {"from": "DimStores",
         "localField": "StoreID",

```

```

        "foreignField": "storeid",
        "as": "avgOnHandByStore_details"}
    },
    { "$unwind": "$avgOnHandByStore_details"
    ,{"$group":
        { "_id":
            { "storetype": "$avgOnHandByStore_details.storetype"},
            "avgnHandQty":{ "$avg": "$OnHandQty"}

        }
    }
    ,{
        $addFields:{round_avgnHandQty: {$ceil:["$avgnHandQty"]},
        }
    }
    ]).pretty()

```

Results

```

{
  "_id" : {
    "storetype" : "Online"
  },
  "avgnHandQty" : 46.90951924191797,
  "round_avgnHandQty" : 47
},

/* 2 */
{
  "_id" : {
    "storetype" : "Store"
  },
  "avgnHandQty" : 21.584948602576453,
  "round_avgnHandQty" : 22
},

/* 3 */
{
  "_id" : {
    "storetype" : "Catalog"
  },
  "avgnHandQty" : 58.059337655733536,
  "round_avgnHandQty" : 59
},

/* 4 */
{
  "_id" : {
    "storetype" : "Reseller"
  },
  "avgnHandQty" : 36.1727101592396,
  "round_avgnHandQty" : 37
}

```

7)Performance

Index are created for mongo collections
 FactStore - index on StoreID and ProductID
 dimProducts- index on productid

dimStores- index on storeid

8) Future Improvement

- 1) Clean up of folder and fact collection on the start of process
- 2) Partitioning for fact data
- 3) Single job design for end to end
- 4) Log management (number of files ,lines in files)
- 5) Referential integrity
- 6)

9) Setup

Copy the jtc folder to c: drive , copy 2009.zip file in c:\jtc folder
Extract the project to talend from C:\jtc\jtcTalendProject