Design For Importer and Data Store

- for ComScore Code Challenge Section 1

Author: Ran Zhao

Email: kevin2robinli@yahoo.com

Contents

1. Summary

1.1 Development Environment and Input Format

The design is to use Java programming language to implement the requirement. The development jdk version **jdk1.8.0_0131**.

The IDE is Eclipse. Version control tool is Git and code repository is Github.

The input sample data format is ".txt".

Java io / nio is needed to read/write into txt file.

1.2 Design Concept

The design for the data store follows "**Column Oriented storage**" concept. Basically, after parsing the input sample, different columns will be split out and stored independently. This way it stores the data is similar as PostgreSQL.

1.3 Error Handling

Since the importer and data store has many i/o features, error handling mechanism is included in the design. It has both Java default error handing and customized error handing.

2. Read/Parse Input, Generate Data Store

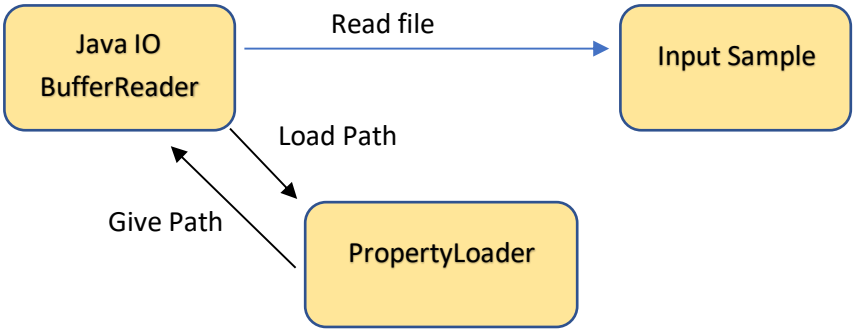
There are two parts in the workflow:

2.1 Read Input Sample And Parse data

2.1.1 Read input

To make the READ feature flexible and easier to manage, the input file path is stored in a property file which keep all input source.

The design is to use a property loader to load the property and get the input file path. Then use Java IO BufferReader to read the file.



2.1.2 Parse Input Sample

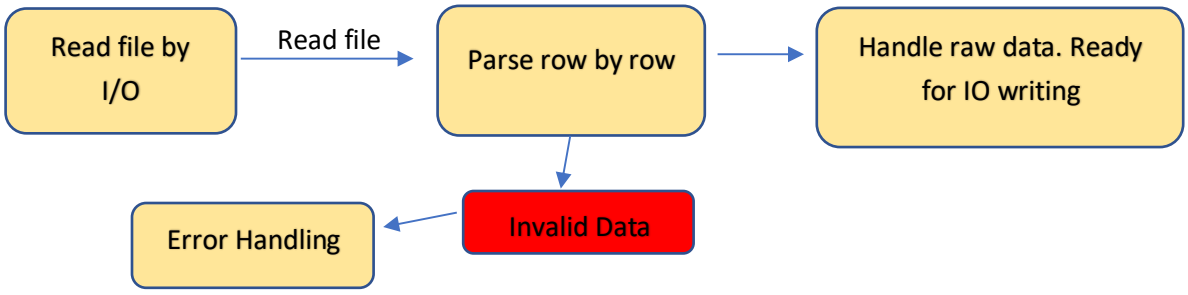
After read the input file in a FileStream, system needs to parse the content, handle and content, and split the information by column.

When parse the content, based on the requirement, the information needs to be separated by "|". So for each input row, each data between two "|" is a single data unit. And for all rows, the single data unit at same index can consist a "column".

For example, row "stb1|the matrix|warner bros|", there are 3 data units can be separated, and the for all rows, 3 columns can be defined.

When the input file contains invalid data, customize error handling. i.e, in the design, the TITLE column max length is 64 char. If it over this max length, will throw "DataOverMaxLenghtException".

Below is workflow of parsing information



2.2 Write Data and Create Data Store

2.2.1 Create data store folder and data store files

When IO is ready to write. Create a folder named with input Sample's name. Then create data store files in that folder. Each data column stores the "Column" defined in Read part. For example, below input will have 3 date store files, one for STB, one for TITLE, and one for Provider.

stb1|the matrix|warner bros|
stb1|the matrix|warner bros|
stb1|the matrix|warner bros|

2.2.2 Create HashRecord file

After creating all data store files, create one more file named "HashRecord". This table is to save the hashcode to check record unique. Based on requirement, STD + TITLE + DATE is unique in the data store.

When write record into data store files, check new record's hashcode in HashRecord file. If it doesn't exist, means this record can be inserted. If it already exists, the old record in data store files need to be overwrite.

All Data Store files and Hash file are in ".txt" format.

3. Workflow

The whole workflow as below: