1. **Business Challenge / Requirement**

Every organization implements CRM as a strategy that integrates concepts of data mining, and data warehousing in order to support an organization’s decision-making process to retain long- term and profitable relationships with its customers.

futureCart Inc. is a hypothetical leading retail company with omni presence in India with more than 5000 retail stores and hypermarkets across and e-commerce in the country.

Company has set of a dedicated team to handle after sales customer care services which handles customer complaints and resolves them to increase customer retention, loyalty and decrease customer attrition.

**Below is an abstract of end to end process:**

* Company has multiple call centers across India to handle the customer requests.
* Customers can reach to the care team over different communication modes: Calls or Chat or Email.
* CCR(customer care representative) will register the complaint with the all the provided details which is called a ‘case’.
* A case can have status open or closed.
* Each case can belong to a category and sub-category. This category and sub-category will determine case priority. Depending on the priority key, CCR has an SLA in hours to close the case within the SLA hours.
* Once case is closed, customer is sent a survey link to rate overall process.
* Customer can take a survey or leave it unattended. He can fill scores from 1-10 on various questions around how his experience was during customer care process.
* Survey response is captured for that particular case.

This complete CRM process generate data and company wants to do some analysis around this data and capture below KPIs to further enhance and optimize the CRM process.

**KPIs (Both on real time data and batch processed data)**

* Total numbers of cases
* Total open cases in last 1 hour
* Total closed cases in last 1 hour
* Total cases priority wise
* Total positive/negative responses in last 1 hour
* Total number of surveys in last 1 hour
* Total open cases in a day/week/month
* Total closed cases in a day/week/month
* Total positive/negative responses in a day/week/month
* Total number of surveys in a day/week/month

1. **The goal of the Project**

Below are some of the high level technical and non-technical goals for this project :

* Get overall understanding of CRM domain.
* Learn **fundamentals** & standards of ETL and data warehousing.
* Real time and batch ingestion of data from multiple source systems to big data storages like hive/Hbase/HDFS using Kafka and Spark.
* Data cleansing/wrangling/transformation using Hive and Spark.
* Lambda architecture where we want to process data in both batch and real-time
* Reporting KPIs using dashboards.

1. **Use – cases**
2. **Data Flow Architecture / Process Flow**

* High Level Architecture :

A screenshot of a social media post

Description automatically generated

* Low level Data flow diagram:

A picture containing screenshot

Description automatically generated

**Explaining the tech stacks in the low level diagram.**

Batch Processing :

Mysql(RDBMS) 🡪 Spark or Sqoop 🡪Hive (Landing) 🡪 hive(final)

* Companies generally store transactional data in RDBMS because they provide faster read and write operations and ACID.
* Two bring them to Hadoop ecosystem we can either use Sqoop, which is a specialized tool to import/export data between RDBMS and HDFS.
* Sqoop provides a faster way of copying data in parallel by mentioning the number of mappers which would copy the data in parallel from RDBMS table to HDFS/Hive tables.
* If data size is not very huge then we can also use Spark to read RDBMS tables using JDBC and load to a dataframe and to a hive table subsequently.
* Hive is best fitted for a datalake/data warehouse solutions where company wants to ingest data from multiple data sources in batch fashion and do some analysis and generate reports on the historical data. It support ANSI SQL like language (Hive query language) which is widely used in data warehouse world by the developers.
* With data stored in hive tables, we can perform complex transformation in huge data set using hive queries or spark.
* Any reporting tool (e.g. tableau) can connect to hive tables and can be used to generate reports.

**Real-time Processing :**

Realtime-processing

^

||

Real-time Json 🡪 produced to Kafka 🡪 Consumed in Kafka 🡪load to Cassandra

* We may want to process the transactional data or stream of events in real time to get quicker insights or detecting patterns and take business decisions immediately.
* There can be different types of real time data sources. E.g. RDBMS tables, json files in a dir, json files from hitting an API etc.
* In a typical real-time processing we ingest the continuous data feed to a storage and at the same time we process the events by applying business rules to get the insights or trigger any other event.

Read/ingest streams:

* Kafka is an scalable stream processing solution to ingest/process data in real time.

It provide a unified, high-throughput, low-latency platform for handling real-time data feeds. It provides wide range of libraries using which we can connect to multiple data sources and process the data as well.

* Apache Cassandra is a highly scalable, high-performance distributed NoSQL database designed to handle large amounts of data ingestion across many commodity servers, providing high availability with no single point of failure

Process the streams:

* Spark Streaming is an extension of the core Spark API that enables scalable, high-throughput, fault-tolerant stream processing of live data streams.

It can connect to Kafka and process the streams with complex algorithms and push back the processed stream to back to Kafka.

* Processed streams can be used for getting insights/detecting patterns or any other ML/Data science activities in real time.

1. **Data sources :**

**We have three types of data sources:**

1. Static/Dimension tables in Mysql
2. History data of 6 days for cases and survey events in mysql table
3. Real time data for current date for cases and survey events in json files

**Data sources for batch Processing :**

**MYSQL :**

We have below tables in mysql which can act like dimension tables:

DB\_Name : trainer\_db

**Tables** :

* retailcart\_calendar\_details (Calendar details for the company)
* futurecart\_call\_center\_details (call center details for the company)
* futurecart\_case\_category\_details (category details of a case event)
* futurecart\_case\_country\_details (country details)
* futurecart\_case\_priority\_details (priority details of a case)
* futurecart\_employee\_details (employee details of the company)
* futurecart\_product\_details (product details of the company)
* futurecart\_survey\_question\_details (question details for the survey)

We can import these tables to Hadoop by any of the following ways :

* Directly to a Hive table
* Import as a file in hdfs location first and load to a hive table.
* Use JDBC connection in spark to read from mysql table and write to a hive table.

**Data sources for real-time processing :**

**pwd**

/mnt/bigdatapgp/edureka\_921625/project2/data/realtime/

If we run realtime\_simulator.py, it generates json data for case and survey every 5 seconds to simulate a realtime data generation.

**Survey json location :** /mnt/bigdatapgp/edureka\_921625/project2/data/realtime/survey

**Case json location :**

/mnt/bigdatapgp/edureka\_921625/project2/data/realtime/case

**Json formats :**

Generated Json files will have below naming convention

<data type>\_data\_<epochtimestamp>.json

Data type can be ‘case’ or ‘survey’

**Sample file names :**

survey\_data\_1592422939.json

case\_data\_1592422939.json

**Case json format :**

[

{

"status": "Open",

"category": "CAT3",

"sub\_category": "SCAT14",

"last\_modified\_timestamp": "2020-06-17 18:42:19",

"case\_no": "600999",

"create\_timestamp": "2020-06-17 18:42:19",

"created\_employee\_key": "240604",

"call\_center\_id": "C-116",

"product\_code": "9829787",

"country\_cd": "PR",

"communication\_mode": "Chat"

},

{

"status": "Open",

"category": "CAT3",

"sub\_category": "SCAT14",

"last\_modified\_timestamp": "2020-06-17 18:42:19",

"case\_no": "601000",

"create\_timestamp": "2020-06-17 18:42:19",

"created\_employee\_key": "215285",

"call\_center\_id": "C-114",

"product\_code": "12457101",

"country\_cd": "EE",

"communication\_mode": "Call"

}

]

**Survey Json format:**

[

{

"Q1": 9,

"Q3": 1,

"Q2": 8,

"Q5": 3,

"Q4": "N",

"case\_no": "600991",

"survey\_timestamp": "2020-06-17 19:42:04",

"survey\_id": "S-500014"

},

{

"Q1": 8,

"Q3": 9,

"Q2": 1,

"Q5": 1,

"Q4": "N",

"case\_no": "600992",

"survey\_timestamp": "2020-06-17 19:42:04",

"survey\_id": "S-500015"

}

]

1. **Dataset Explanation & Schema**

|  |  |  |  |
| --- | --- | --- | --- |
| futurecart\_calendar\_details |  |  |  |
| column Name | Data type | Column description | sample value |
| calendar\_date | date, | calendar date in yyyy-mm-dd format | 2011-02-20 |
| date\_desc | varchar(50), | calendar date in words | Sunday, February 20, 2011 |
| week\_day\_nbr | smallint, | number of day in a week | 2 |
| week\_number | smallint, | week number of the year | 4 |
| week\_name | varchar(50), | week name | Week 04 |
| year\_week\_number | int, | weeknumber with year | 201104 |
| month\_number | smallint, | month number in the year | 1 |
| month\_name | varchar(50), | month name | february |
| quarter\_number | smallint, | quarter number in the year | 1 |
| quarter\_name | varchar(50), | quarter name | Q1 |
| half\_year\_number | smallint, | half year number in the year | 1 |
| half\_year\_name | varchar(50), | half year name | 1st Half |
| geo\_region\_cd | char(2) | geographic region code | US |
|  |  |  |  |
| futurecart\_call\_center\_details |  |  |  |
| column Name | Data type | Column description | sample value |
| call\_center\_id | varchar(10) | Unique identifier for a call center | C-101 |
| call\_center\_vendor | varchar(50) | Vendor company name which is handling the call center | Concentrix |
| location | varchar(50) | Call center location | New york |
| country | varchar(50) | Call center country | US |
|  |  |  |  |
| futurecart\_case\_category\_details |  |  |  |
| column Name | Data type | Column description | sample value |
| category\_key | varchar(10) | Unique identifier for a case category | CAT1 |
| sub\_category\_key | varchar(10) | Unique identifier for a case sub category | SCAT1 |
| category\_description | varchar(50) | Category description | Subscription |
| sub\_category\_description | varchar(50) | Sub category description | Renewal |
| priority | varchar(10) | Priority key | P1 |
|  |  |  |  |
|  |  |  |  |
| futurecart\_case\_country\_details |  |  |  |
| column Name | Data type | Column description | sample value |
| id | int | Unique identifier for a country | 4 |
| Name | Varchar(75) | Country name | India |
| Alpha\_2 | Varchar(2) | Country short name 2 chars | IN |
| Alpha\_3 | varchar(2), | Country short name 3 chars | IND |
|  |  |  |  |
| futurecart\_case\_priority\_details |  |  |  |
| column Name | Data type | Column description | sample value |
| Priority\_key | Varchar(5) | Unique identifier for a case priority | P1 |
| priority | varchar (20), | Priority level | Highest |
| severity | varchar (100) | Severity level | critical |
| SLA | varchar (100) | SLA in HOURS for the priority and severity combination | 1 |
|  |  |  |  |
| futurecart\_employee\_details |  |  |  |
| column Name | Data type | Column description | sample value |
| emp\_key | Int | Unique ID of an employee | 10001 |
| first\_name | varchar | First name | Georgi |
| last\_name | varchar | Last name | Facello |
| email | Varchar | email | Georgi.Facello01@testmail.com |
| gender | Varchar | gender | M |
| ldap | varchar | User id | 5941CF7D |
| hire\_date | Date | Hire date | 2014-04-06 |
| manager | varchar | Manager key | 455246 |
|  |  |  |  |
|  |  |  |  |
| futurecart\_product\_details |  |  |  |
| column Name | Data type | Column description | sample value |
| product\_id | varchar | Unique id for a product | 26355 |
| department | varchar | Department description | GROCERY |
| brand | varchar | Brand description | Private |
| commodity\_desc | varchar | commodity description | COOKIES/CONES |
| sub\_commodity\_desc | varchar | Sub commodity description | SPECIALTY COOKIES |

|  |  |  |  |
| --- | --- | --- | --- |
| futurecart\_survey\_question\_details |  |  |  |
| column Name | Data type | Column description | sample value |
| question\_id | varchar | Unique id for a survey question | Q1 |
| question\_desc | varchar | Question text | How would you rate your overall experience with the customer support process? |
| response\_type | varchar | Response type (scale or options) | Scale |
| range | varchar | Scale range if response type is scale else NA | 1-10 |
| negative\_response\_range | varchar | Scale range to qualify a survey response as negative | 1-4 |
| neutral\_response\_range | Varchar | Scale range to qualify a survey response as neutral | 5-7 |
| positive\_response\_range | varchar | Scale range to qualify a survey response as positive | 8-10 |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| futurecart\_case\_details |  |  |  |
| column Name | Data type | Column description | sample value |
| case\_no | varchar | Unique ID of a case | 2024 |
| create\_timestamp | varchar | Case create timestamp | 2020-04-20 01:01:29 |
| last\_modified\_timestamp | varchar | Case last modified timestamp | 2020-04-20 01:01:29 |
| created\_employee\_key | Varchar | Employee key who created the case | 274649 |
| call\_center\_id | Varchar | Call center id where case is logged and handled | C-104 |
| status | varchar | Current status of the case | Open |
| category | varchar | Category key of the case | CAT1 |
| sub\_category | Varchar | Sub category key of the case | S CAT1 |
|  |  |  |  |
| communication\_mode | Varchar | Mode of communication | Email |
|  |  |  |  |
| country\_cd | Varchar | Country code | PY |
| product\_code | Varchar | Product code | 997719 |

|  |  |  |  |
| --- | --- | --- | --- |
| futurecart\_survey\_details |  |  |  |
| column Name | Data type | Column description | sample value |
| survey\_id | varchar | Unique ID of a survey | S-1000 |
| Case\_no | varchar | Case no for which survey has been filled | 130114 |
| survey\_timestamp | varchar | Survey taken timestamp | 2020-04-20 01:01:29 |
| q1 | Varchar | Q1 response | 2 |
| Q2 | Varchar | Q2 response | 7 |
| Q3 | varchar | Q3 response | 3 |
| Q4 | varchar | Q4 response | N |
| Q5 | Varchar | Q5 response | 7 |

**Relation between different datasets.**

A screenshot of a cell phone

Description automatically generated

**Survey KPIs logic in pivot table :**

below feedback KPIs can be either positive, negative or neutral for a survey if

* Support\_Process\_Feedback
* Employee\_Conversation\_Feedback
* Employee\_Technical\_Feedback
* Overall\_Feedback
* Referral\_Feedback

case

when survey.Q1 >= question\_dim.negative\_lower and survey.Q1 <= question\_dim.negative\_upper then 'Negative'

when survey.Q1 >= question\_dim.neutral\_lower and survey.Q1 <= question\_dim.neutral\_upper then 'Neutral'

when survey.Q1 >= question\_dim.positive\_lower and survey.Q1 <= question\_dim.positive\_upper then 'Positive'

end as Support\_Process\_Feedback,

case

when survey.Q2 >= question\_dim.negative\_lower and survey.Q2 <= question\_dim.negative\_upper then 'Negative'

when survey.Q2 >= question\_dim.neutral\_lower and survey.Q2 <= question\_dim.neutral\_upper then 'Neutral'

when survey.Q2 >= question\_dim.positive\_lower and survey.Q2 <= question\_dim.positive\_upper then 'Positive'

end as Employee\_Conversation\_Feedback,

case

when survey.Q3 >= question\_dim.negative\_lower and survey.Q3 <= question\_dim.negative\_upper then 'Negative'

when survey.Q3 >= question\_dim.neutral\_lower and survey.Q3 <= question\_dim.neutral\_upper then 'Neutral'

when survey.Q3 >= question\_dim.positive\_lower and survey.Q3 <= question\_dim.positive\_upper then 'Positive'

end as Employee\_Technical\_Feedback,

case

when survey.Q4 = 'N' then 'Negative'

when survey.Q4 = 'Y' then 'Positive'

end as Overall\_Feedback,

case

when survey.Q5 >= question\_dim.negative\_lower and survey.Q5 <= question\_dim.negative\_upper then 'Negative'

when survey.Q5 >= question\_dim.neutral\_lower and survey.Q5 <= question\_dim.neutral\_upper then 'Neutral'

when survey.Q5 >= question\_dim.positive\_lower and survey.Q5 <= question\_dim.positive\_upper then 'Positive'

end as Referral\_Feedback

1. **Target Environment**
2. **Problem Statements / Tasks**

High level task is to create a data mart on CRM data with a lambda architecture where we will ingest and process data in both batch and real time.

We also want to enable reporting of KPIs in both batch and real time.

Technical tasks in details :

**Refer data flow and architecture for additional reference:**

* Perform a batch ingestion from mysql to hive tables for static dimensions and historical data for case and survey events.
* Capture new cases and survey events from json files being written to a directort and produce them to a kafka topic.
* Create a consumer application which will consume the incoming messages from the topic and ingest to cassandra tables in real – time.
* Once we have captures both batch and real-time data in stage tables, Perform a data modelling around business KPIs in hive to create facts and dimension.
* Join facts and dimensions and load pivot table
* Create a tableau reports on pivot table
* On the real time feeds, Also develop a real time analysis framework which will create real time KPIs and publish them to a dashboard. For this you might have to join both real time feeds and static mysql tables.

1. **Approach to Solve**
2. Identify tools and technologies
3. Identify data for real time and batch ingestions
4. Build incremental/history ingestion load
5. Model the data to facts and dimensions as per requirement
6. Process the data and load facts and dimensions
7. Create a pivot table for KPI reporting
8. **Considerations / Assumptions**

* Dimensions/static/historicat data are present in Mysql tables. Details shared above.
* All the target tables that we develop should be have ORC storage format.
* All the target tables that we develop should have additional row\_insertion\_dttm column which will store current timestamp .
* All the target tables that we develop should be partitioned on any date (business/sales date) columns if available.
* We can create any topic name with any number of partitions. End goal is to ingest real time sales and price change data from mysql to casssandra.
* We will have real time data for current day in json files for case and survey events.
* There will be a simulator script in python which (if we run it) will start creating jsons file for both case and survey events in a directory.
* If there are multiple surveys for a closed case then we need to consider the survey with the earliest timestamp.
* Answers to survey questions will be divided into negative/neutral/positive responses in final fact table depending on the range for every question available in survey question dimension table.

1. **Deliverables**

* A fact table joined on case and survey events
* A pivot table joined among above fact table and other dimensions.
* Real time analysis framework to monitor KPIs in real time.

1. **Business Benefits**

Afterhis solution is developed, business will have below operations benefits.

* Company can track some important information/KPIs in real time which will enable it to take immediate decisions.
* It will also create a data mart for the company and with this, it can monitor its historical performance and can check what went well and what can be improved in future.
* Data mart can also be used for advanced data science.
* Solution will create a pivot table which can be integrated to any tableau to create reports. Or data analysts/business users can directly query the table.