Report on Old Batch Job & Database

This report contains: - Information to batch job, which is created using JAVA 8, Issues & bugs in batch and fixes implemented for it, Database details and structure of tables used by our applications.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Content** | **Page No.** |
| 1 | Batch Job Details | 2 |
| 2 | Existing Issues and bugs within Batch Job | 3 |
| 3 | Fixes and ideas to address the Issues | 4 |
| 4 | Database details | 5 |
| 5 | Table Structure | 6 |

**Batch Job Details**

* The Existing batch job is written using Java version 8 LTS (8 Long term support) released in 2014 and current java SDK is using update released around 2018-2019.
* The code to the batch job can be accessed via remote server pstrwaaa9192. This is windows server which contains only code.
* The batch jobs are deployed to host server pstrlaaa9426.
* We have 5 separate java batch files for STA as named below. Each running for a particular table used by our STA applications.
  + BotDatalakeJob.java----for tables
    - BOTFEEDBACK
  + OTVJob.java----for tables
    - OTV\_IXNS
  + RealTimeSTAJob.java----for tables
    - REALTIMEPHRASES\_EXT
    - REALTIMEPHRASES\_INT
    - REALTIMETOPICS\_EXT
    - REALTIMETOPICS\_INT
  + STADatalakeJob.java----for tables
    - CLOUD\_STA\_IXNS
    - PSENTIMENTPHRASES
    - NSENTIMENTPHRASES
    - HIST\_TOPICS\_IXNS
    - SILENCETIMETOPICS
    - HOLDTIMETOPICS
    - IVRTOPICS

**Issues Within Batch Job**

* The current batch job process does not bring in all desired conversations present within Genesys cloud. This might be due to how the filtering logic is implemented.
* The current batch job uses Version 171.0.0 of SDK Library provided by Genesys cloud to make API calls and get data.
* This version of SDK does not provide us with all data/KVP’s like Empathy Score.
* The workaround for this is to make exclusive call to Genesys API with leads to repetition and redundancy ultimately affecting the performance of the batch process and further increasing the time taken to finish the process.
* Upgrading to the next version of SDK leads to unforeseen errors.
* The Error handling for the program is not sufficient and this makes it difficult to find and solve errors and bugs in the program.
* The Conditional Logic implemented is complex in nature and not modular enough. Changes to a certain part in program affects other parts.
* The Development Environment (IDE) used to develop code is legacy software and does not have modern tools or features like syntax highlighting, code formatting, GitHub extension, other extensions.

**Fixes and ideas to address the Issues**

* SDK Version 217 incompatible with program. To fix the issue where batch job had failed due to unidentified reasons, we reverted to using older version of SDK i.e. 171.0.0.
* A screenshot of a computer

  Description automatically generatedThis version of SDK is stable but does not provide necessary data like empathy score. Below is the workaround implemented for this.
* To cover all the issues the batch job can be newly written in Python which will use the latest version of SDK provided by Genesys cloud.
* This will also allow us to add additional KVP’s into our database.
* The structure of database and tables can be refined along with this to incorporate all data.
* Python is an advanced programming language which has inbuilt Data analytics capabilities which can be used in future to perform tasks like value prediction and classification.

**Database details**

A] The KVP’s including Acoustics data elements are dispersed into 5 Separate tables:-

1. CLOUD\_STA\_IXNS
2. SILENCETIMETOPICS
3. HOLDTIMETOPICS
4. NSENTIMENTPHRASES
5. PSENTIMENTPHRASES

B] Current VOICE STA DASHBOARD uses data from HIST\_TOPICS\_IXNS table (6). There is a mismatch between KVP’s available in these tables. For instance, 2 & 3 do not have LOB and 6 does not have silenceTimeDuration.

C] VOYA PAL DASHBOARD uses BOTFEEDBACK (7) table. It currently only has 5 KVP’s i.e. 5 columns. New KVP’S like Sentiment Score need to be added.

* STARTDATE
* CONVERSATIONID
* COMMENTS
* LOB
* DEVICETYPE
* INTERACTIONREASON

**Table Structure**

Table 1: -

A screenshot of a computer

Description automatically generated

Table 2: -

A screenshot of a computer

Description automatically generated

Table 3: -

A screenshot of a computer

Description automatically generated

Table 4: -

A screenshot of a computer

Description automatically generated

Table 5: -

A screenshot of a computer

Description automatically generated

Table 6: -

A screenshot of a computer

Description automatically generated

Table 7: -

A screenshot of a computer

Description automatically generated