Data Distribution Plan

DBST 663 Group 6

University of Maryland University College

Data Distribution Plan

# Describe the Database Distribution

In our hospital database, there are 6 tables. We will create two different databases called “Hospdba” and “Hospdbb”, and we will create a link between these two. We will be utilizing the horizontal fragmentation strategy for one table, the “Visits” table. We will create 6 tables in the first database, and then 1 table, the fragmented one, in the second database.

# Identify Tables for Fragmentation

We have chosen to split our “Visits” table with horizontal fragmentation. The next section will provide the rationale behind this decision.

# Provide Rationale for the Fragmentation

The rationale behind horizontally fragmenting the “Visits” table is due to the nature of this table having significantly more entries than any of the other tables. Distributing the load of this table across two databases will provide enhanced query speeds and reduce network latency.

# Describe 5 Distributed Views and Queries

Here are 5 distributed views and queries based upon real-world requirements needed from the hospital.

1. The view will be a union between the “Patients” tables on “Hospdba” and “Hospdbb”.
2. The first query will select all data from the “Patients” table, joined with the “Visits” and “Provider” tables where the “Provider” specialty equals “Radiology”.
3. The second query will select all data from the “Patients” table, joined with the “Provider”, “Visits”, “Prescriptions”, and “Medications” tables where the “Medications” name equals “Penicillin” and the “Visits” date field is equal to or greater than “Jan 1, 2017”.
4. The third query joins together the “Hospital” table, “Visits” table, and “Patients” table to show how many visits a CMS Region has had with the last patient seen, sorted by the number of visits.
5. The fourth query joins together the “Patients” table with the “Visits” table, and a subquery will be performed to determine how many visits a patient has had, in order by visit count.

# Describe the Updates Plan

In our database, the most likely time needed to update data would be when a doctor needs to go back in for adding additional notes to a visit. We will provide several updates to the “Visits” table, specifically the notes column. The updates will be based on the Patients\_ID, Providers\_ID, and Hospital\_CCN\_ID.

# Describe the Retrieval Strategy

For our project, we will be creating a few views to pull together the fragmented data. These views will be the primary source for our distributed queries detailed in the second above. Outside of the fragmented “Visits” table, retrieval of all remaining data will be normal.