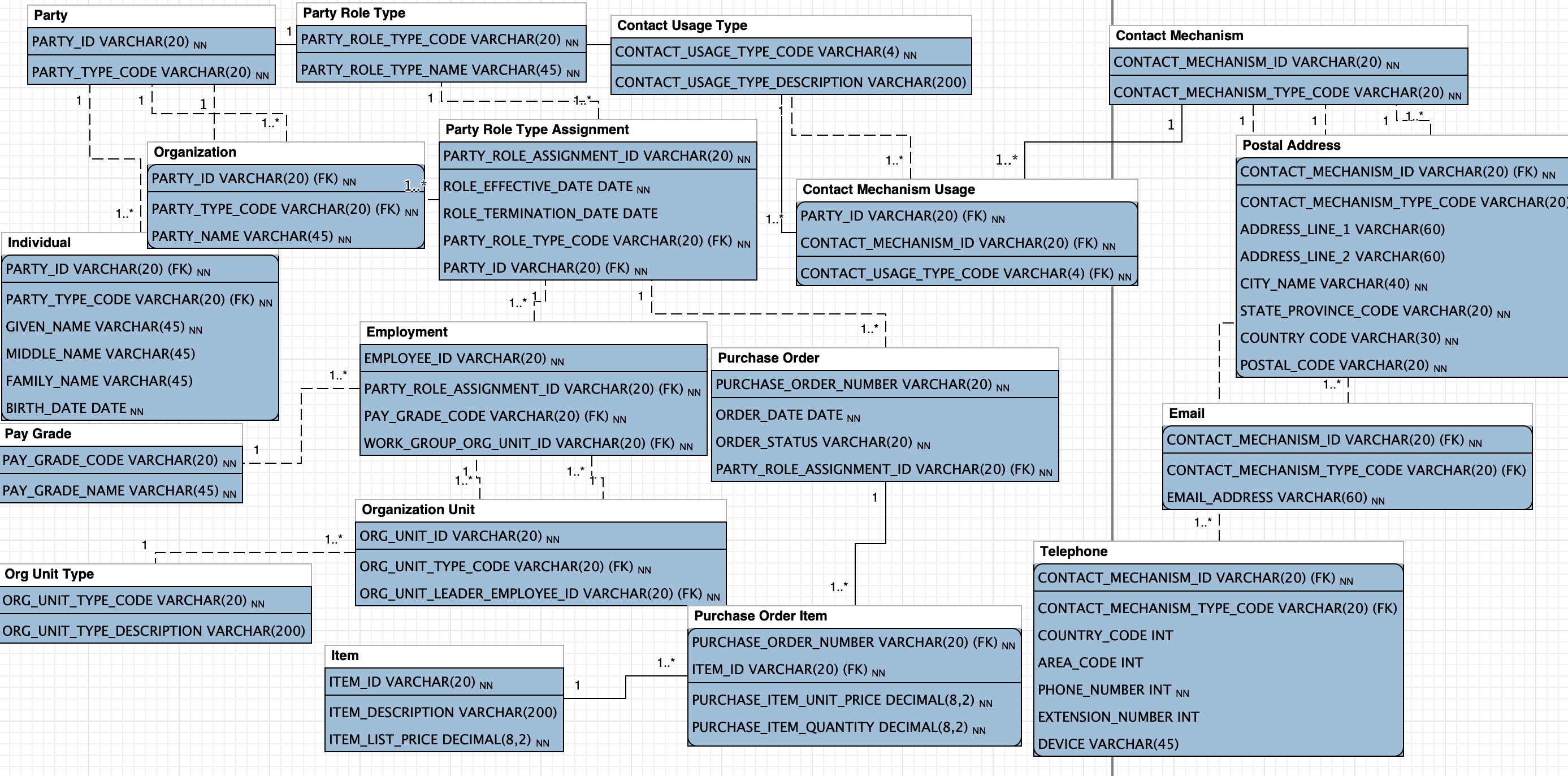
Module 5 Assignment



CONTACT\_MECHANISM\_TYPE\_CODE is value from {‘Postal’, ‘Email’, ‘Telephone’}

PARTY\_TYPE\_CODE is value from {‘Individual’, ‘Organizational’}

List of index(except primary indexes):

|  |  |
| --- | --- |
| Table | Index |
| Party Role Type Assignment | PARTY\_ID |
| Employment | PARTY\_ROLE\_ASSIGNMENT\_ID |
| Purchase Order | PARTY\_ROLE\_ASSIGNMENT\_ID |
| Individual | GIVEN\_NAME + MIDDLE\_NAME + LAST\_NAME |
| Organization | PARTY\_NAME |

Explanation: The first three indexes are for referential integrity. The last two indexes are not unique, but they are often included in a WHERE clause. So adding them in index can speed up the query.

Diagram

Description automatically generated

Key-value database is suitable for this model. It is relatively simple compared to other database models. Key-value database provides scalability, speed, and flexibility. Assuming we are storing a huge amount of data in this model, key-value database can scale out by implementing partition, replication, and auto-recovery. Although in key-value database, queries are generally limited to searching values using the corresponding keys. There are a few database management products allow searching on values. So if users would like to query project for a specific id or name, they can do so.

Columnar database is also suitable for this model. Compared to the row-oriented databases, the columnar database only needs to read the selected columns, which significantly speeds up the data processing. In this model, several columns can be grouped together to enhance the query efficiency. For example, Worker’s first name and last name are usually queried together, so we can group them.