Graph Database

The type of database is based on graph theory. A graph database uses a graph structure, with nodes, edges, and properties to store data. Each node in the graph contains a list of relationship records that represent the relationship it has with other nodes around it.

This can be especially useful when doing the JOIN operation, which can be very resource heavy in relational databases. This means that the JOIN operation can be done much more efficiently with a graph database. The reason a graph database can be so efficient in retrieving data from a complex database is because it stores relationships between records directly. This can be seen when the search that is performed is more than one level deep. This also means that the main advantage of a graph database can be seen when the searches get more complex.

A graph database does require a different language than SQL to do the searching. This is mainly because SQL just is not a good language to represent a graph database. One graph database, named Neo4j, uses a query language called Cypher. Here is a sample of the Cypher language:

```
MATCH (node:Label) RETURN node.property
MATCH (node1:Label1)-->(node2:Label2)
WHERE node1.propertyA = {value}
RETURN node2.propertyA, node2.propertyB
```

Cypher is case sensitive, they surround nodes with parentheses, and a relationship is made by using → between two nodes. Cypher also uses the dot operator to refer to properties of nodes.