MongoDB is a document-oriented database management system. A document-oriented database stores data in “documents” rather than tables like in a traditional relational database. MongoDB specifically stores its data in collections, which are groupings of documents. A document in MongoDB is a record of data. MongoDB stores its documents in a format called BSON, which is short for Binary JSON. JSON is a format for exchanging data similar to XML. BSON is primarily used by MongoDB. MongoDB uses BSON documents for when it is storing data in storage, querying documents, and modifying documents.

MongoDB is considered a NoSQL database. NoSQL (which stands for “Not Only SQL”) databases differentiate from traditional relational databases in that they generally do not use SQL or store data in tables. Data can be stored in a number of different formats other than relational table structures which include documents, graphs, and key-value stores. MongoDB, along with some other NoSQL databases such as CoachDB, store data in documents. Instead of SQL, MongoDB uses its own query language. Basic operations are done using CRUD, which stands for create, read, update, and delete. These CRUD operations can be used to create, delete, and manipulate documents in the database. MongoDB also supports JavaScript shell scripting as well as drivers for compatibility with other programming languages.

There are several reasons why MongoDB would be chosen to be used over a traditional relational database. One reason is because MongoDB has sharding. Sharding is a method of splitting up data across multiple machines. Specifically in MongoDB, collections of data are partitioned according to whatever needs are specified and then distributed among shards. Each shard holds some of the partitioned data. The

purpose of sharding is to spread stored data across a distributed system. Sharding can be advantageous for storing large amounts of data across large distributed systems as well as allow for a system to be scaled to hold larger and larger quantities of data. An example of partitioning data through sharding would be to first organize names in alphabetical order, then names ranging from A to D could be recorded in different shards from names ranging from E to H.

Another potential reason for choosing to use MongoDB is to avoid following the necessary constraints in a traditional relational database management system. It is not necessary to perform joins on the data in a document based system, and time can be saved because it is not necessary to spend time figuring out how to fit data into relational and table form. Documents in MongoDB also eliminate the need to create empty attributes or columns that may need to be created in some cases for relational databases. MongoDB also does not follow the ACID properties of transactions, which could potentially speed up transactions because they do not need to follow all of the ACID constraints.

MongoDB can be used as a database system for content management systems, commercial websites, archiving, and a variety of other tasks. An example of commercial use of MangoDB includes using it for the content management system for MTV Networks' websites. One reason it is being used is for the flexibility that the document structure provides in storing data. Another commercial use of the database management system is to archive users' posts on the Craigslist website. One reason behind this choice is to use MangoDB's sharding feature to more easily scale the increasing number of machines used to store large amounts of data accumulated by the

website.

To go about using MangoDB, the open source system first needs to be downloaded and then installed to wherever it will be used. If any other programming languages will used with the database, drivers to support those languages should then be installed. To learn how to use the system, the MongoDB manual can be referenced to learn the CRUD operators as well as how to use the other features the system provides. The system can then be configured accordingly to the task, such as entering the data or setting the shard key that partitions data into desired shards.