Metadata Management

Eland Anthony

Metadata is an important aspect in organizing and outlining a database. One is likely aware of the term meta, which describes the topic or subject at hand as a relationship to itself. For example, one may hear or read that this text or picture is meta. By this, meta describes the topic as being frequently used or popular. The described subject describes or relates to a common subject. Meta within the term metadata is a hint into its definition. That is, metadata describes the data of the data - similar to a subject describing a common subject as described above. This definition may give an idea as to what metadata is, but is not sufficient to understand how, why, and where it is implemented. Metadata is frequently used in database management systems. This should come as no surprise, as databases contain vast amounts of data. There must be some way to efficiently organize and describe the data as it relates to aspects within the data. In this paper, metadata uses, implementations, aspects and available tools will be discussed utilizing scholarly sources for each topic.

Now that we have a general idea of what metadata is and its prevalent use in databases, let us discuss why it is used. A metadata management system requires several tools to satisfy the management of the database. Metadata is used in conjunction with tools such as SQL: “The accompanying query language is SQL extended with reification and reflection capabilities” [1]. Reification refers to the process of defining abstract concepts concretely, and reflection refers to using such reifications and the data itself to gain insight within the data and metadata. SQL works with data and metadata to query the data and its relational attributes described for the data. Thus, metadata can be thought of as a meta-entity describing the object-entity meta-relationship, “which can be used to store semantic and design information about various object-entites” [1]. SQL is dependent on metadata, and if metadata is not defined it cannot be used within an SQL query. So, it can be concluded that metadata and SQL are useful tools toward managing databases. Metadata is also useful in databases for data security: it can include data about constraints [1], thereby limiting access to certain profiles or users.

We are now informed of some of the reasons metadata is useful within database management systems; now let us look at some examples of metadata management in such a system. Suppose we have defined data including contact information for individuals. Such information can be thought of as the inputs for web forms before ordering an item or signing up for an account. This information collected online is likely stored within a database of some sort.

In such an example, metadata may describe the relationship between the user and its object entities. Thus, each time an entry is added to the database the structure of the database is changed to appease such changes [1]. The SQL language, described above as the common language for metadata management and database management systems, also must define metadata for the language. This allows for the metadata in the database to be described. Thus, we may seek to find the shipping address of a certain user who has passively entered our data into the database via a web page. The metadata for such a relation must be described within the SQL structure [1]. We can then use this to define the relationship between the employee and the shipping address and store this information in our database. Such a definition provides the metadata of our database. It is important to note that the simplistic example described above is only one example of metadata representations. We will now take a look into other representations used in database systems.

The metadata application discussed earlier is an example of structural metadata. Structural metadata can be thought of as answering: "Which variable's value appears in which column? Which row represents which case? Are there hierarchical relationships?" [2]. Our structural metadata would contain the relationship between the user and each of their attributes such as shipping address. This is imperative in a database in order to define its structure. One such implementation is the relational model. The relational model utilizes entities, relationships, attributes and other metadata to organize and provide functionality for SQL queries. The second metadata representation is reference metadata. Such a method of representing metadata may also be described as descriptive or footnote metadata, as it can be thought of as relating to sampling, methodology, production notes, quality measurements, or other aspects [2]. Such descriptions of reference metadata are separate from structural metadata as descriptive metadata is not necessary in defining a relational database. Rather, it is extra information that can be used to describe the data. One such implementation of this representation is in conjunction with a structural database to further describe data, such as defining data integrity, or quality and completeness of said data [2]. Another representation of metadata is defined as administrative metadata. Think of the definition of administrative in the context of database management. An administrator is likely the creator or upkeeper of the database, or an organization executive who uses said data for the benefit of said organization. This metadata may have a security clearance such that only the roles described above have access. Such data may include "data created through the process of administering data, covering its collection, production, publication, and archiving" [2]. The most likely use and purpose of this metadata is to keep track of past changes, debug errors that had occurred after a change within the database, or track the actions of an unauthorized user making administrative changes within the database due to a virus of some sort. One example of this representation can be seen in Salesforce. As a Salesforce developer, the utilization of such metadata to track, validate, and deploy changes is imperative in keeping the production organization stable for users whether it be employees, clients, or consumers. The final metadata representation is that of behavioral metadata. The adjective described in this metadata representation is used to illustrate the behavior of the users of a database: "[It includes] information about the reaction and behavior of users when working with data, and that of respondents when the data is being collected" [2]. This representation of metadata describes the usage of said data by users, and may be computed inherently within the database depending on user interaction as opposed to being explicitly defined. Such data is not necessary within a database, but may be helpful to organizations looking to perfect their implementation. A use case for such a metadata representation may be within an organization or even applications such as Oracle's SQL Developer to analyze its users' main usage within the database. This metadata may help such organizations to improve functionality within their database or application. In addition to SQL Developer, there are several more tools for metadata management.

Metadata tools are very useful for the management of metadata in order to streamline and add functionality to the management process. One popular metadata management tool is the IBM InfoSphere Information Server. This server utilizes standard relational database technology to allow users the ability to "import metadata into the repository from multiple sources, export metadata by various methods, and transfer metadata between design, test, and production repositories" [3]. Another popular metadata management tool is Oracle Enterprise Metadata Management. As implied by its name, Oracle Enterprise Metadata Management is utilized by companies to search and browse metadata. This tool specifically intends to allow for data lineage, semantic definition, impact analysis, and semantic usage analysis for metadata assets within a given data set. Additionally, the tool claims to support algorithms to stitch together metadata assets from separate providers [3].

In conclusion, metadata management is an important consideration when crafting a database. The implementation of metadata management among other tools allows for querying data within a system to return data based on its metadata. Such a method has several use cases described above, such as in research and enterprise development. Metadata can be represented in four ways at minimum: structural metadata, reference metadata, administrative metadata, and behavioral metadata for a variety of purposes. Strategies of metadata management have improved throughout the years; however, there is always room for improvement. One such improvement may be the implementation of decentralized database systems. Such a system may use the benefits of decentralization, or the lack of an authoritative server or system, to protect against attacks to gain ultimate privilege into the database. Such a system could consist of innate definitions such as privileges to certain unique individuals which may not be changed without approval of several users of the system. Thus, the system is inherently not centralized and may provide for greater security.

References:

[1] S. Y. Cheung, J. J. Lu, and C. M. Wyss, “Metadata Management and Relational Databases,” in *Proceedings of the 43nd Annual Southeast Regional Conference, 2005, Kennesaw, Georgia, USA, March 18-20,* 2005, vol. 1, pp. 1–6.

[2] A. Gregory, P. Heus and J. Ryssevik, “Metadata” in *Building on progress: expanding the research infrastructure for the social, economic, and behavioral sciences*. Opladen: Budrich UniPress, 2010, ch. 3, pp. 489–508.

[3] T. King, “The 13 Best Metadata Management Tools for 2020,” *Best Data Management Software, Vendors and Data Science Platforms*, 10-Dec-2020. [Online]. Available: https://solutionsreview.com/data-management/the-best-metadata-management-tools/. [Accessed: 15-Feb-2021].