The classical definition of metadata is “data about data”. But this simplistic definition belies the complexities of metadata. The simple taxonomy of metadata as seen in Fig md.1 shows some of the variations of metadata.

There are two basic types of metadata – technical metadata and business metadata. Technical metadata consists of those technical descriptions of data such as tables, attributes, indexes and so forth. These technical types of metadata are found in data dictionaries, directories, and repositories. The world of business metadata is made of non technical definitions, formulae, descriptions, and so forth. Business metadata relies on context in order to give meaning and shades of meaning to business metadata. In addition valuable reference tables are stored in master data management facilities.

The value of metadata is often not apparent. In order to illustrate the importance of metadata, consider the progression shown in Fig md.2.

At the left of the progression is the number “7”. The number “7” by itself means almost nothing. It is bare and stripped of context. Next in the progression we see “$7”. Now we know that the number “7” is in the context of seven dollars. Next in the progression is “$7 NYSE”. The number “7” has something to do with the New York Stock market rising $7. Then we see “$7 NYSE Aug 13, 2004”. Now we understand that the New York Stock Exchange has risen 7 dollars on Aug 13, 2004.

In this case the data is “7” and everything else is metadata. Without the metadata the number “7” really has no meaning. The number “7” takes on meaning when context – metadata – is added.
One of the major issues surrounding metadata is whether metadata should be centralized or decentralized (i.e., distributed).

Fig md.3 illustrates this basic choice.

The decentralized or distributed approach is depicted by Fig md.4.
In Fig md.4 it is seen that every architectural component of DW2.0 has its own metadata. In nearly every case these “local” metadata facilities are built by the vendor servicing the architectural entity. In other words, for the data marts there is a Business Objects universe, for the data warehouse there is an Oracle directory, for the ETL component there is Informatica’s super glue, and so forth. Each vendor supplies the metadata that is needed for the architectural component.

The bad news is that metadata exchange is neither easy to do nor semantically meaningful. DB2 cannot easily pass its metadata to Filetek. Filetek cannot pass its metadata to Cognos. Cognos cannot pass its metadata to Ascential, and so forth. And even if these different technologies could pass their data to another architectural component, it probably would not mean much and would not be terribly useful.

The advantage of the distributed metadata approach is that it comes naturally with the products that underlie the DW2.0 architecture. The disadvantage of the distributed approach is that there is no real universal sharing of metadata.

The centralized approach to metadata management is seen in Fig md.5.
Fig md.5 shows that each of the architectural components of DW2.0 contribute metadata to a centralized repository. The idea of a centralized repository is appealing. But the reality is that there are many pitfalls to a centralized approach to metadata.

When metadata is centralized it must be gathered. It must be kept up to date. It must be reconciled when two or more architectural entities have conflicting values, and so forth.

In fact there is no right answer. Metadata can be centralized or decentralized. Both approaches have their advantages and disadvantages.

The advantages and disadvantages of metadata centralization or distribution are shown in Fig md.6.
### Advantages
- Each architectural entity has its own metadata
- Autonomy of processing
- Automatic update as each change is made locally
- The technical support for local metadata is built into each product
- Local metadata can be changed on an as-needed basis with no interference from a centralized authority
- The budget for metadata is natural as local products are acquired and installed

### Disadvantages
- There is no central agreed definition or standard
- There is a need for exchange of metadata which is not met
- There are no standard definitions across the enterprise
- It is easy to fool one’s self into thinking that local metadata is all that is needed
- Once local metadata has been passed to another location, there is the issue of ownership of the metadata
- Once local metadata has been passed to another location, there is the issue of keeping the metadata up to date

### Centralized
- There is a need for centralized definition that is easily met by this approach
- The past history of metadata has always favored this approach

### Initial population is gruesome
- Keeping the centralized repository up to date is very difficult
- Population of data after the fact is difficult because all of the original developers have left
- Population of data after the fact is difficult because all budget has been used up
- There is no guarantee that the centralized definition is correct
- Access to the centralized repository can be onerous
- The resources required for ongoing maintenance do not fit into any convenient budget
- People who need to use and change local metadata do not want a central authority controlling their usage of metadata
- If used actively, the centralized approach can become a choke point

**Fig md.6**
There are advantages and disadvantages to different implementations of metadata.