# How important is 'Context'?

Software agents that are able to analyze problem situations, dynamically changing conditions, and events, must have some understanding of the meaning of the information that they are reasoning about. How can we create a computer-based environment that conveys to a software agent sufficient meaning for that agent to undertake reasoning tasks?
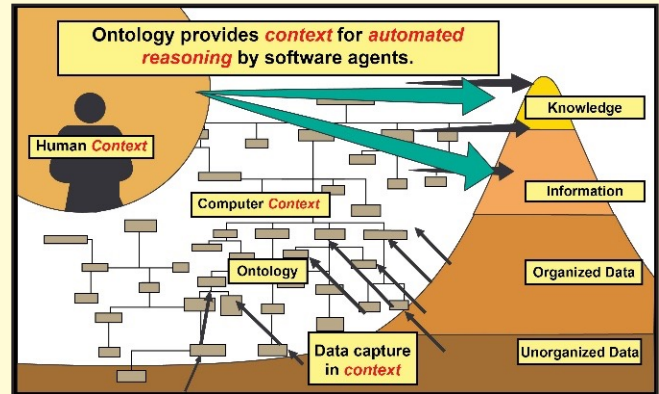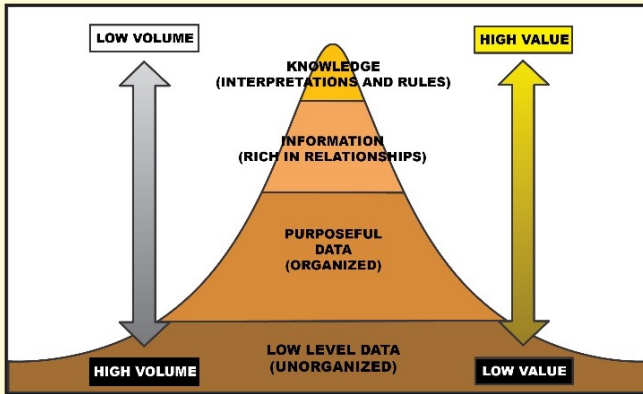
To answer this question, it is necessary to first draw a distinction between data and information. Data are simply numbers and words, while information adds to data another very important component, relationships. These relationships are critical to any reasoning process because they provide context. Without this context even a human being would have great difficulty making sense out of a bunch of data. What makes it so easy for us human beings to reason about a wide range of data is the context that we have accumulated in our cognitive system over time through an experience-based learning process. We automatically convert data to information as long as we can find in our memory the context within which the words and numbers (i.e., data) that our eyes see, convey meaning. In other words, subject to the existence of relevant experience our cognitive system automatically adds the relationships (i.e., context) that are necessary for us to reason about the data. Since this process is automatic, it is perhaps not unreasonable for us to forget that computers do not have this capability because they do not have an equivalent cognitive system. The same would apply if we were to ask a literate six-year-old child to interpret the meaning of a typical printed, single-page agenda of a business meeting. Although the child may be able to readily read the agenda it is unable to make much sense of its contents because it has no prior experience of such meetings. In other words, the child lacks the context that is necessary for reasoning about the agenda.

For a computer to be able to support automatic reasoning capabilities we have to create a software environment that incorporates context. This can be achieved fairly easily by constructing an information model as a virtual representation of the real-world context within which software agents are expected to apply their reasoning capabilities. Such an internal information model is referred to as an ontology. It describes the real-world context in terms of objects with characteristics and relationships. For example, in a military command-and-control context such objects would include different kinds of weapons, a wide range of infrastructure objects, weather forecasts, friendly and enemy units, and even conceptual objects such as the notions of threat, planning, mobility, and readiness. Generally speaking, the more relationships among objects that are included in the ontology the more context is provided by the ontology, and the more powerful (i.e., intelligent) the reasoning capabilities of the software agents are likely to be. In summary, the virtual information model provides the necessary context for the symbolic reasoning activities of the agents.