Model	Advantages	Disadvantages
Repository	• Efficient for sharing large amounts of data • Subsystems can store and retrieve data without concern for other subsystems • Data control tasks (backup, access control, etc) are centralized • Adding new subsystems is easy	• Data model is common, and must be used by all components • Data model is hard to change • Data control needs to be fine-grained enough to serve the needs of every subsystem • Distributing the repository over multiple machines is complex
Client-server	• Can use multiple machines effectively • Adding new clients or servers is easy • Upgrading servers is easy	• Changing installed clients is difficult • No shared data model between servers • Converting data formats is inefficient
Layered	• Allows incremental development • Easy to change layer internals • Interface changes only affect adjacent layers • Killer app: multi-platform systems	• Difficult to architect • Jumping layers is prohibited, so lots of pass-through functions • Can be low-performance