

# Big Idea (Week 10): File Systems: Storing and Organizing Information in a Standard, Self-Describing Manner

Persistent storage—a way of storing data that has long life and does not go away when the power is removed from the system—is an important component of computer and media systems. This allows us to store our programs and applications (*e.g.* the OS, the mp3 player, our apps), our media content (*e.g.* songs, videos, games), and our data (*e.g.* address book, calendar, saved game state) so it is readily accessible when we turn on our computer or computationally enabled appliance. Since we can make high capacity persistent storage media (*e.g.* disk drives, flash memory), it is convenient and economical to mix all these things (applications, content, data) on a single storage device.

This raises the important question of how we organize these things on the storage device so we can find them again, use them properly, and add/modify/delete elements. The raw hardware interface is too low of a level and error-prone, supporting reads and writes to addresses. Consequently, it is useful to define a structured interface and discipline for storing the raw data on the hardware. This software abstraction layer hides the hardware details (*e.g.* address where data is stored) from the applications. It also allows the storage device to be self-describing—so that appropriate computer programs can use some data on the device to discover all the content, applications, and data on the storage device. Standardizing this interface, we allow the storage device to be used and understood by different computers (*e.g.* file sharing with USB flash drivers).

This structured interface is a *file system* and it organizes the raw bits into files and directories. The files are collections of bits and the file system allows us to name each of the collections and associate with it useful information such as the role of the collected bits (*e.g.* is this an mp3-compressed file? an uncompressed audio stream at 44KHz? an executable program? a collection of text messages?). Directories give us a way of providing structured organization for collections of files and are, themselves, special files that tell us how to find other files.