**Recommendations for XYZ, Data Collection and Storage for a Water Meter Company,**

**Data Storage for an Appliance Retailer, and File Creation and Access**

David LaCivita

CTIM 281 Software Design and Development

Professor Chuck Bisson

November 3, 2020

**Recommendations for XYZ**

In the offices of XYZ the employees need to share applications and data. The recommended storage of data for ease of access and security is to maintain the data on network drive. As the employees work with the data it should be backed up periodically during the work day. At the end of the day the data should be backed up to a separate network drive as well as to a drive at a different location. The second networked drive provides protection against disk failure and the drive at a separate location protects against a catastrophic event like a fire or flood at the office building. The backup drives should be in a RAID configured for reliability.

The applications should be stored on the local computer for each employee that needs to use them. This keeps the application running at top performance with no network lag. The data can be read to the local machine’s memory for processing and output to the new file in memory and written to the networked drive. Any sensitive data should be stored on a non-networked drive and backed up the same as the regular data. Employees that need to work with the sensitive data can access it in a secure room and on the same machine the data is stored on.

**Data Collection and Storage for a Water Meter Company**

Mueller Systems is a manufacturer and distributor of connect water meters. They also can collect and process data for the municipality or utility. The meters are either connected to a network or transmit data to a mobile device. When the data arrives at the data center it is stored in a temporary database and processed for usage and billing. The database is backed up every fifteen minutes and full backups are made weekly.

Once the usage and billing is finished the data is transferred to two separate databases for additional processing. One database is processed for live reporting and the other for long term reporting and forecasting.

**Data Storage for a Major Appliance Retail Store**

Before my son was born and I became a stay home dad, I sold major appliances for Appliance1 in Hanover, MA. Part of my job was to maintain the inventory for stocked items as well as floor models. In addition to our local inventory, we were connected to the inventory of the buying group we belonged to. All of the inventories were read into our point of sale software and updated at different times. The floor model inventory was initially kept in a spreadsheet saved in a Google Drive and shared with the sales team. Eventually, the developer of our point of sale software updated it so a spreadsheet formatted correctly could be uploaded to it. Prior to the update we depended on each member of the sales team to update the spreadsheet when a floor model was sold. After the update the point of sale software could update sold items but new floor items had to be added manually. Also, after the update, we could keep track of store stock as it was sold and we bought scanning hardware to track incoming stock.

Our buying group served every independent dealer in New England and all of them used the same inventory and point of sale software. Our buying group’s inventory was maintained at their distribution center and incoming inventory was updated as manufacturers’ deliveries were made. Each night we used our point of sale system to place orders for all the sales we made that day. At the distribution center, outgoing inventory was removed from the inventory when it was scanned out and placed “in line” for our trucks to pick up each morning. The driver was given a report of everything ordered so he knew exactly what was to be picked up.

Finally, we maintained multiple licenses for our point of sale software and it was run on a central server and accessed by a remote desktop application. Not only did the program track sales and inventory, we used it for scheduling shifts, a time clock and making notes on customers’ records.

**File Creation and Access**

Computer files are collections of related data stored on a persistent memory device in a computer system. Files are generally either text files or binary files. A text file contains data that are somewhat human readable and can be read by a text editor. Binary files contain data that are stored as 0s and 1s and are not encoded as text. Either file is identified by a name that describes its contents followed by a dot then an extension that describes the type of data in the file. One example is the file name for this paper, *discussion-7.docx*. The extension, .docx, indicates it as a file that can be read by Microsoft’s Word, a word processing program. The name describes it as a discussion written for the seventh module of this class. Another example would be *MaryPortrait.jpg* which is an image file, likely a portrait of my wife. Each file also has an associated time indicating when it was created and a time when it was last modified.

Every file has a location on a storage device and a size which is generally described in bytes. A byte is a very small unit that contains a single character. Since modern files are large and contain a huge number of characters we usually describe files in kilobytes, megabytes, or gigabytes (thousands, millions, or billions of bytes respectively).

Data stored on a computer system are stored in a data hierarchy. At the top of this hierarchy are the files. Data files are a collection of related records which are made up of fields that describe a single aspect of the data. At the base of the hierarchy are the characters which can be collected to compose a field.

All files are organized and stored in directories on storage devices. The path to a file’s location contains a hierarchy of all the directories in which the file is stored. The path to this paper’s location on my Linux machine is /home/dave/Documents/massasoit/discussion-7.docx and on a Windows machine it might be d:\user\dave\documents\massasoit\discussion-7.docx. For an application to use a file that isn’t in the same directory the full path must be used to call the file.

To process any data in a file we must first know the operations used to access the file. Like a variable or constant, a file must be declared and assigned a data type and an identifier. Much like string or num, there are two data types used to describe a file: InputFile and OutputFile. Once declared, to access the data in an InputFile, the file must be opened. Opening a file associates its location in storage with the declared identifier in the application. Once the InputFile is opened the data can be read from its storage device into memory for processing. After the application has processed the data the new version can be written to the OutputFile. Once any file is no longer needed by the application it should be closed making it inaccessible to the application. Closing the file frees up computer resources and saves the data, allowing it to be accessed by other applications that may need it.