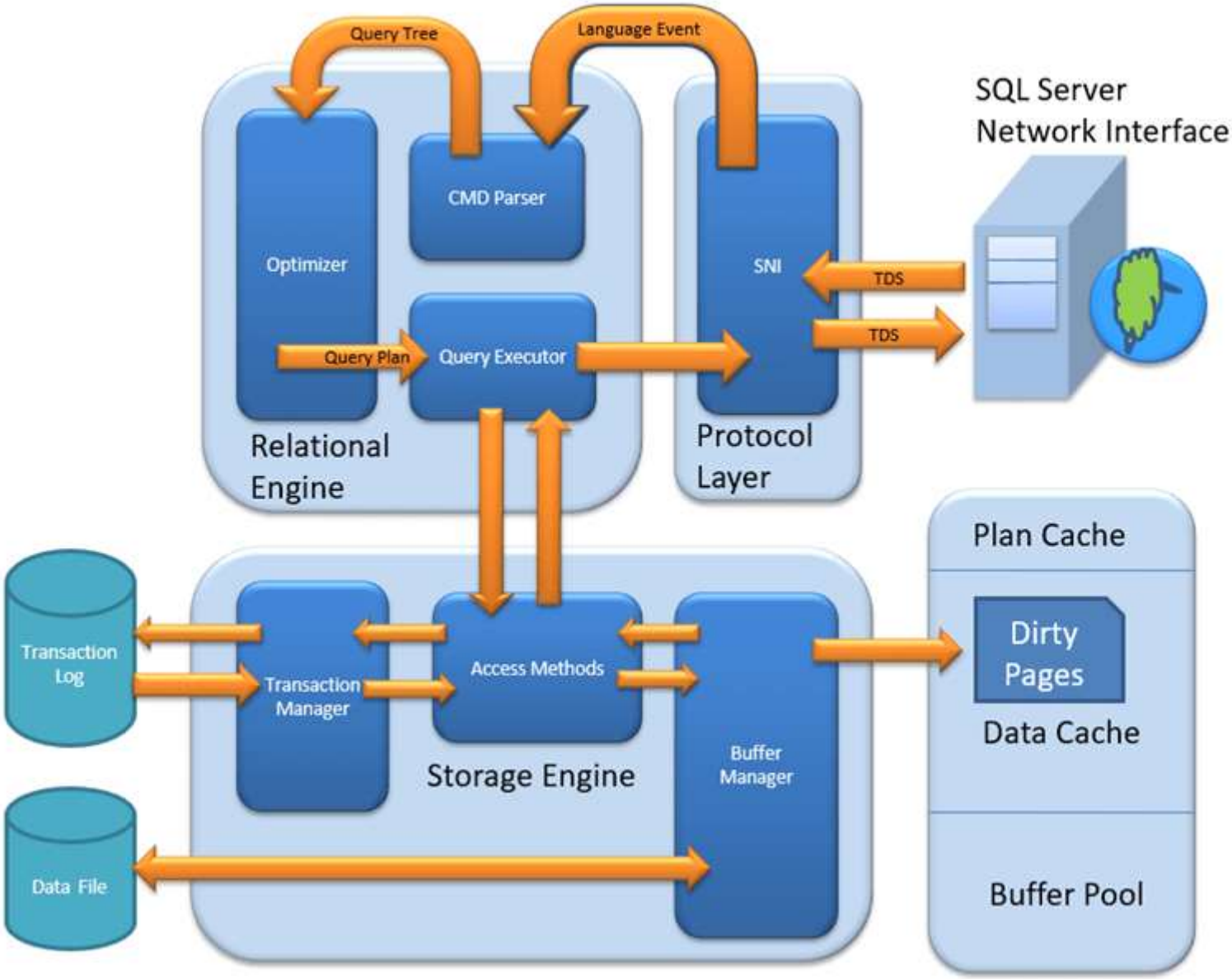
PROGRAM 4.0

Server is a client-server architecture. MS SQL Server process starts with the client application sending a request. The SQL Server accepts, processes and replies to the request with processed data. There are three major components in SQL Server Architecture:

1. Protocol Layer
2. Relational Engine
3. Storage Engine



SQL Server Architecture Diagram

Three primary components make up SQL Server architecture: Protocol Layer, Relational Engine, and Storage Engine

Protocol Layer

This layer supports three kinds of Client Server Architecture, plus a stream.

- Shared Memory. The client and SQL server run on the same machine and can communicate by a shared memory protocol.
- TCP/IP. This protocol allows the client and SQL server to interact even though they are installed on separate machines and are remote to each other.
- Named Pipes. This protocol lets the client and SQL server via a Local Area Network (LAN).
- TDS. All three protocols use Tabular Data Stream packets. These packets allow data transfers from the client machine to the server machine.

Relational Engine

It's also called the Query Processor, and it contains the SQL Server components that determine precisely what a query must do and how to best accomplish it. The relational engine executes user queries by requesting data from the storage engine and processing the returned results. The engine has three major components:

- CMD Parser: The CMD Parser’s chief purpose is to check the query for Semantic and Syntactic errors, then generate a Query Tree. The Parser is the first Relational Engine component to receive Query data.
- Optimizer: The Optimizer works on built-in exhaustive and heuristic algorithms to ultimately minimize query run times and create an execution plan. Note that the Optimizer finds the cheapest plan, not the best one.
- Query Executor: The Executor creates the data fetching logic’s execution plan. When the Executor receives the data from the Storage Engine, the result is published to the Protocol layer. After the results are published, the resulting data goes to the end-user.

Storage Engine

Storage Engine stores the data in a storage system such as SAN or a disk and retrieves it when needed.

- File Types. The Storage Engine has three files: the Primary, Secondary, and Log files.
- Access Method. This component interfaces between the query executor and the buffer manager and transaction logs.
- Buffer Manager. This component manages the core functions for the following three modules:
    - Plan Cache. The buffer manager looks for the existing execution plan stored in the Plan Cache.

- Data Parsing. The buffer manager then provides access to the needed data.
- Dirty Pages. These pages hold the Transaction Manager's processing logic data.

- Transaction Manager. The Transaction Manager activates when there are Non-Select Transactions and manages these transactions using Log and Lock Managers.

4.2 Viewing Query Tree using ssqtv

Step 1: Download SSQTV – SQL Server Query Tree Viewer is a completely free educational tool that can be used to visualize query trees within SQL Server and to display internal optimizer information.

Current version is available on : ssqtv-0.9.zip (131KB) (August 2018)

https://www.tf3604.com/wp-content/uploads/ssptv/ssqtv-0.9.zip

Note that .NET Framework 4.5 must be installed on your machine to be able to use the program.

Step 2: Installation Extract the contents of the ZIP file into a directory of your choice, then run SqlServerQueryTreeViewer.exe. Step 3: Restore a CorpDB sample Database or create any DB Restore CorpDB from a backup file given in the link below (we will learn about Restoring DBs in later chapters). If not, create some sample database and run some sample queries.

https://www.tf3604.com/wp-content/uploads/optimizer/CorpDB.bak.zip

Step 4: Run the Query As a very simple example, connect to copy of the CorpDB sample database. Paste in and execute the following query: Trace Flags used for viewing Query Tree

dbcc traceon (3604);

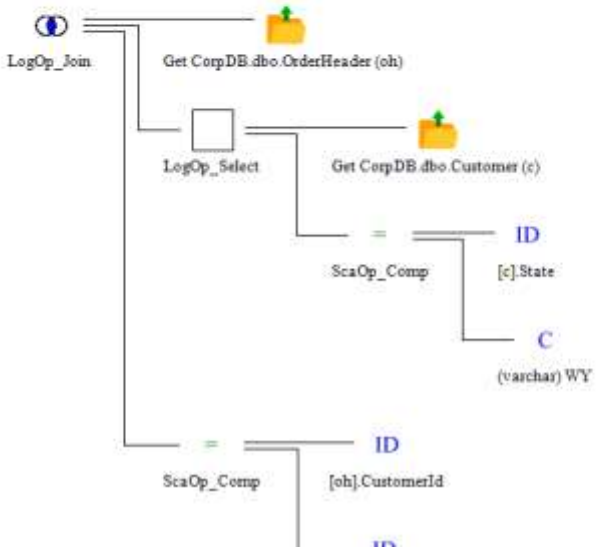
select c.CustomerID, c.FirstName, c.LastName, c.Address, c.City, c.State, oh.OrderId, oh.OrderDate from CorpDB.dbo.OrderHeader oh inner join CorpDB.dbo.Customer c on c.CustomerID = oh.CustomerId where c.State = 'WY' option (recompile, querytraceon 8605, querytraceon 8606);

Step 5: Query Tree Display SSQTV will display the various query trees generated by the optimizer. For instance, the --simplified tree is displayed as show

Trace Flag	Result
3604	Output extra information to "Messages" tab in SSMS
8605	Show initial parse tree (converted)
8606	Show transformed parse trees (input, simplified, join-collapsed, normalized)
8607	Show output tree

Step 5: Query Tree Display

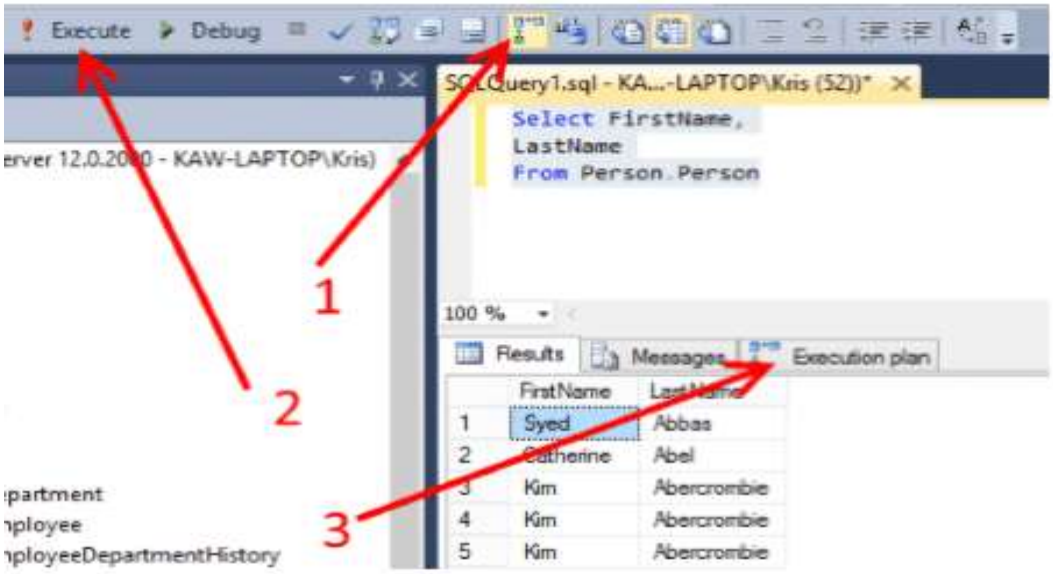
SSQTV will display the various query trees generated by the optimizer. For instance, the --simplified tree is displayed as shown below.



4.3 Viewing a Query plan

A query plan is a set of steps that the database management system executes in order to complete the query.

The reason we have query plans is that the SQL you write may declare your intentions, but it does not tell SQL the exact logic flow to use. The query optimizer determines that. The result of that is the query plan. In SQL Server a query plan is called an execution plan



One you have created the plan, run the query and then select the Execution plan tab **(3)** to view it.

There are also some Shortcut Keys to see the Query Plans

- You can display the **Estimated Execution** Plan in SQL Management Studio by pressing CTRL + L in the query window.
- You can include the **Actual Execution** Plan in the results set by pressing CTRL + M.

And this is our query plan.