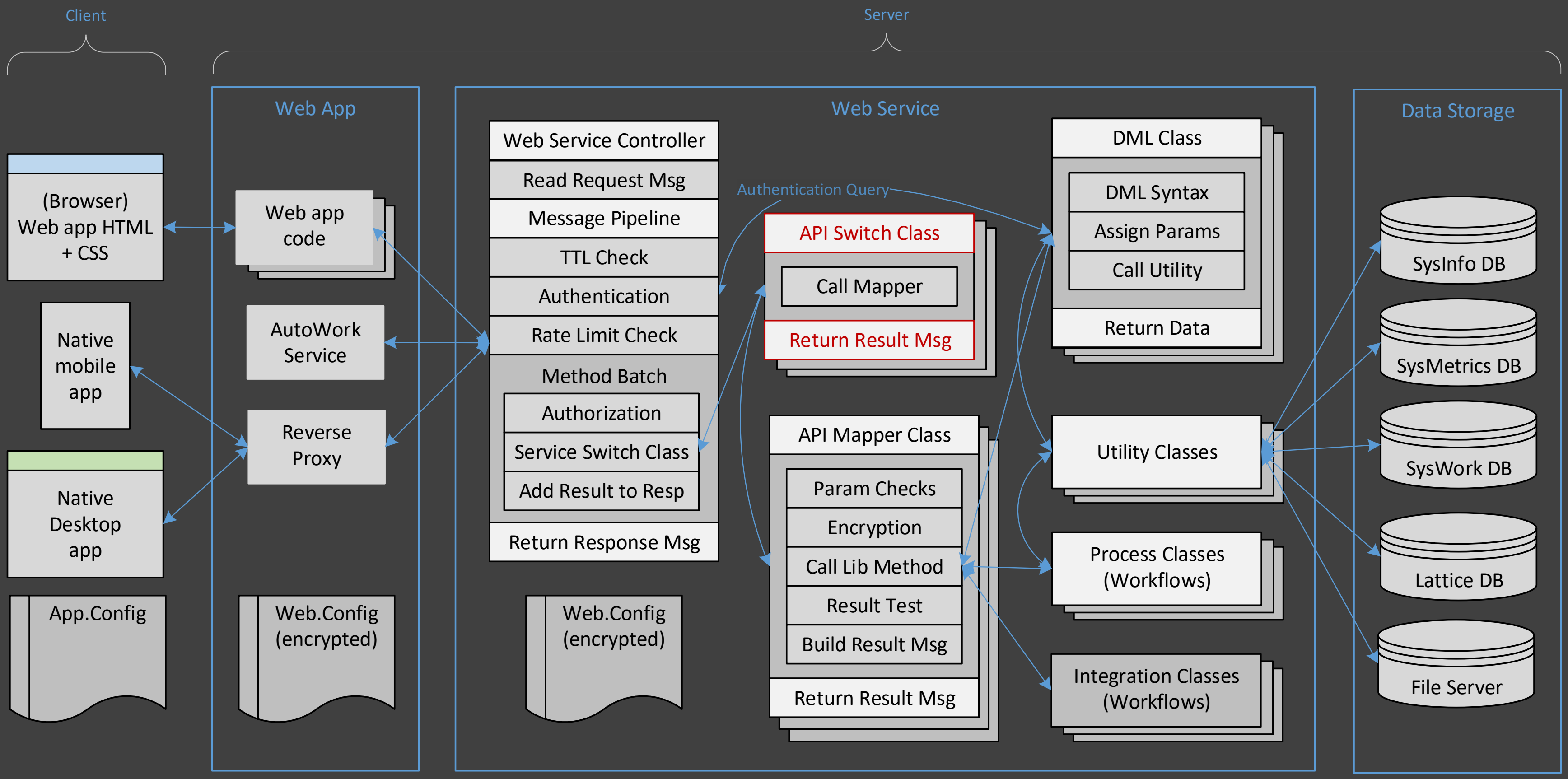
Distributed Grid Platform (DGP): API Switch Class



Every API Switch class contains a switch with case statements for each of the methods in the API. The case statements are selected based on the API name label passed as an input parameter in each API request message. The API name is a text string that follows an APIName.MethodName.VesionName pattern, which looks similar to the namespaces in the .NET framework itself. However, these API name values are simply labels that are completely decoupled from the internal classes and methods that do the actual work of the system.

This abstraction/decoupling makes it much easier to maintain consistent and intuitive naming conventions for all of the API names and method names in a system. It is also extremely important functionality for service evolution and the immutable append-only conventions that enable constant extensions to a system without breaking backward compatibility. It also allows the internal code libraries to each evolve independently with their own logical naming conventions, etc.

The case statements instantiate a specific mapper class and pass the API request message into the appropriate method of the mapper object. The overall collection of mapper methods for an API can be broken up into as many mapper classes as desired.

The collection of input parameters can vary among the different mapper methods. Some of the switch case statements can pass default input parameter values into the mapper object methods which act as flags for branching logic in the mapper method. For example, this is used to differentiate between update and delete actions for different API case statements calling the same mapper object Save method. The results from the mapper method are then returned to the web service controller when complete.