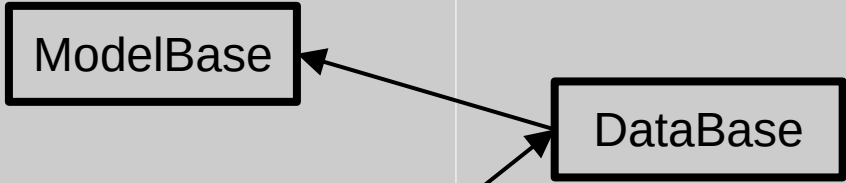
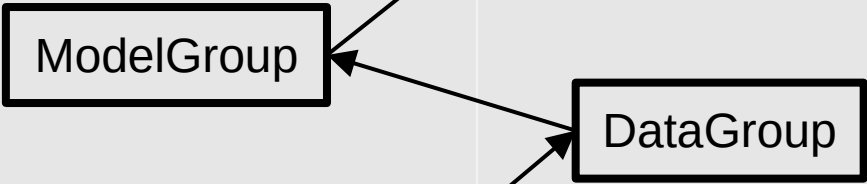
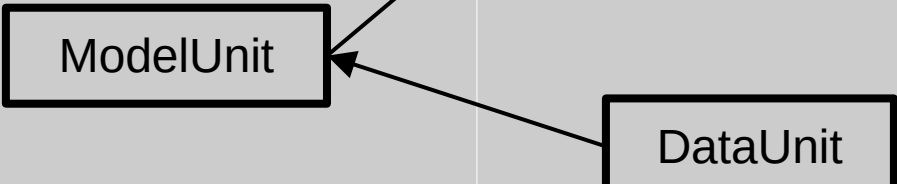
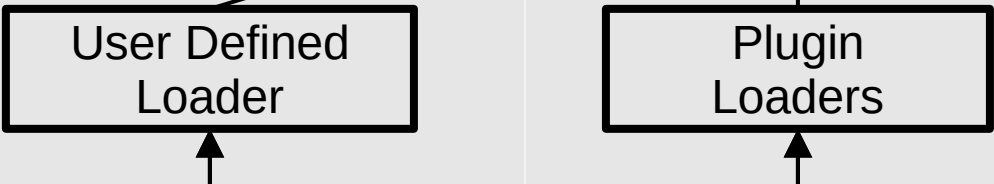
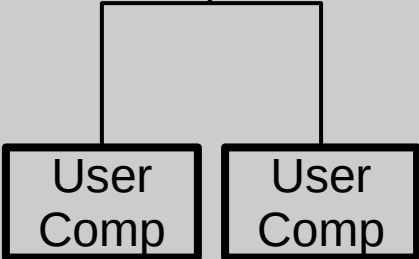
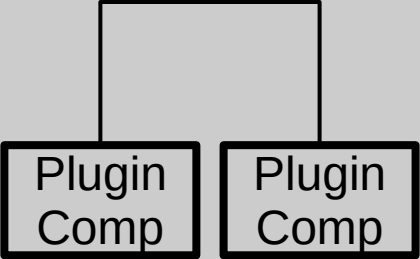


	User Defined	Dmanage Functionality
Data Base - This level inherits all methods from the data group level and adds functionality to read and process data from multiple data group locations and consolidate databases. - This is under development.	 <pre>graph LR; ModelBase <--> DataBase;</pre>	
Data Group - This level inherits all methods from the Data Unit level and adds functionality to process multiple data units and generate simple databases - This utilizes the isValid() method from the component assembly to search the folder/file structure and find valid data units.	 <pre>graph LR; ModelGroup <--> DataGroup;</pre>	
Data Unit - This class inherits from the component assembly class and has access to all the data metadata related to your simulation or experimental “run” - This is the level were you process and visualize data from one run. - The DataUnit class adds functionality to the user defined ModelUnit class for database creation.	 <pre>graph LR; ModelUnit <--> DataUnit;</pre>	
Component Assembly - This class assembles the component into one class - An isValid() method is required to determine if the data location contains valid data. This is used for the dataGroup level - This level is reserved for data loading, and little to no data processing should be done here. - Plugin loaders may exist for your specific simulation or experimental protocol.	 <pre>graph LR; UserDefinedLoader <--> PluginLoaders;</pre>	
Components - These component classes contain the methods to load the data itself and the metadata associated with it. - Data components can be thought of as the “files” containing the data available for loading for the datatype. Examples: - CSV files - Oscilloscope waveform - Electric fields - Input variables 1D, 2D, or 3D data	 <pre>graph BT; UC1[User Comp] --> UDL[User Defined Loader]; UC2[User Comp] --> UDL;</pre>	 <pre>graph BT; PC1[Plugin Comp] --> PL[Plugin Loaders]; PC2[Plugin Comp] --> PL;</pre>