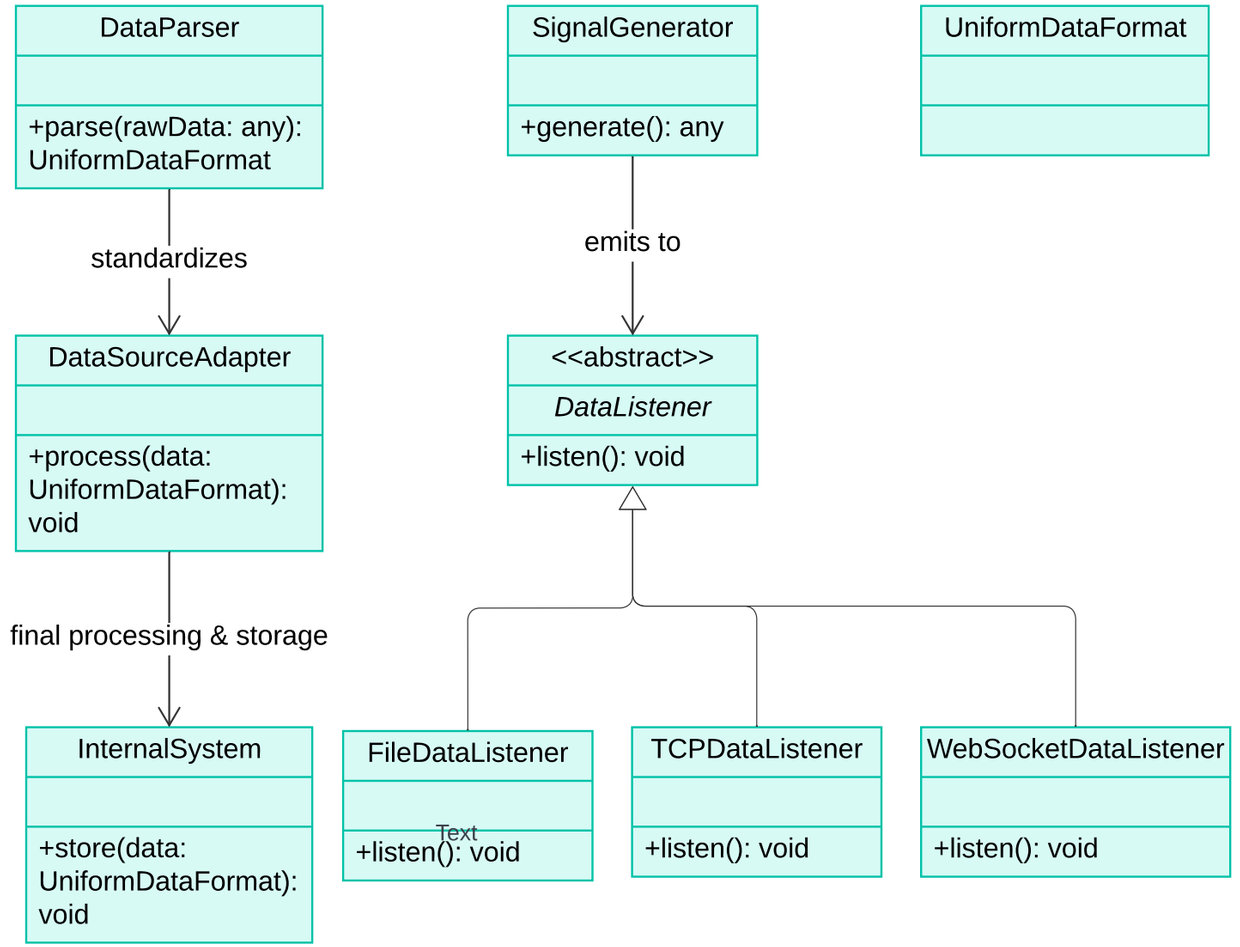

1. **DataParser**: Transforms raw data into a standardized format, indicated by the parse method which returns a UniformDataFormat. The arrow labeled "standardizes" pointing to DataSourceAdapter implies that it serves to prepare the data for further processing.
2. **UniformDataFormat**: Represents the standardized data structure that the system uses. Its presence suggests that data from various sources must be unified into a single format for consistent processing.
3. **DataSourceAdapter**: Acts upon the uniform data, likely preparing it for storage or further analysis, as suggested by the process method. The arrow "final processing & storage" to InternalSystem suggests that DataSourceAdapter is a middleware component, not the endpoint of data handling.
4. **SignalGenerator**: Emits signals for the system to react to. Its generate method indicates that it might be creating data events or alerts. The arrow labeled "emits to" pointing to the DataListener implies it's a source of events for listeners.
5. **DataListener**: An abstract class representing a generic listener interface with a listen method, indicating an event-driven design pattern where different implementations will react to signals.
6. **FileDataListener**, **TCPDataListener**, and **WebSocketDataListener**: These three concrete classes implement the DataListener interface, each with a listen method. They represent different endpoints or systems that act upon the received data signals.