Serialization:

The serialize interface consists of one method, serialize. The serialize method returns a char*. It was decided that the Sys class would inherit from serialize, and that an Object would return “{Object}” when serialized. Classes were decided to be serialized by returning a char* with curly brackets around the object name, and params specified with seperators (“|”) before and after the parameter and specified as parameter_name=_____. For Example, the return of a string “hi” being serialized would be {String|cstr_=hi|}. Arrays are serialized such that every object is serialized inside curly brackets. For example a Array of strings containing “bob”, “Builder” would be serialized as {Array|arr_={String|cst_=bob}{String|cst_=Builder}. Size_ts are stored as strings of at least two characters. Structs are designated with fields separated by brackets instead of “|”.

Deserialization:

The deserialize interface consists of one method, deserialize. The deserialize method takes in a char* representing the string being deserialized, and returns an Object*. Every that you want to deserialize has its own deserialize class, ie DeserializeString, DeserializeArray, ect. Deserializing a class inside a class causes the deserialize class to use an instance of another deserialize class.