**STK Integration**

**Intro paragraph**

**Python Overview of some sort? (or maybe just a link to how to install python?)**

**Task 1: Develop an understanding of the STK API**

The STK API consists of two major sub-systems: Connect and the STK Object Model.

**Connect**

**[Connect](https://help.agi.com/stkdevkit/Subsystems/connect/Content/theVeryTop.htm)** is a library of string commands for STK, originally designed to operate over a TCP/IP socket.

Advantages of Connect include:

* Connect commands are easy to read, understand, and build.
* Connect commands make it very easy to set multiple parameters on one or multiple objects with a single command.
* The TCP/IP interface enables STK to receive Connect commands from other applications on the same machine, or other machines.
* Connect commands can be sent from applications in any programming language; all that is required is a socket connection to STK.

Disadvantages of Connect include:

* When generating Reports and Graphs via Connect, one is limited to the pre-defined Report and Graph styles installed on the target machine.
* Any report data returned from a Connect command will come in the form of a string, which must be parsed.
* The Connect library is weighted heavily towards setting object properties. It is not always possible to "get" current object properties via Connect.
* Connect does not support auto completion or compile-time debugging. The only way to determine whether a Connect command is valid and free of typos is to send the command to STK and receive an "ACK" (acknowledged) or "NACK" (not acknowledged) response.
* While Connect offers a "Verbose" mode for debugging, specific error messages are not always available.

**Object Model**

The STK **[Object Model](https://help.agi.com/stkdevkit/Content/automationTree/objModel.htm)** is an object-oriented interface to STK, built on Microsoft COM technology.

Advantages of the STK Object Model include:

* The Object Model supports auto completion (i.e., IntelliSense, Javadoc) and compile-time debugging.
* The Object Model is not dependent on report styles. Rather, data values are pulled directly from the STK Data Providers.
* Data values returned from STK are returned as objects of an appropriate form, eliminating unnecessary string parsing. For instance, a series of data over time is returned as an array of values.
* The Object Model generally offers parity between property setters and getters.

Disadvantages of the STK Object Model include:

* Becoming familiar with the Object Model involves a much steeper learning curve than Connect, particularly for non-programmers.
* The Object Model often requires many more lines of code than Connect.
* The Object Model, being COM technology, is limited to communication between applications on the same machine.

**Choosing the Right Technology**

The choice of Connect or Object Model should be driven by the requirements of the application and the needs of the developer. It is important to note, however, that this need not be an exclusive choice. Connect commands can be used in a COM application, side-by-side with Object Model code.

**Task 2: Establish a connection with STK and create a new scenario**

The following code will be created for this first Task:



**Component Object Model** (**COM**) is a [binary-interface](https://en.wikipedia.org/wiki/Application_Binary_Interface) standard for [software components](https://en.wikipedia.org/wiki/Component-based_software_engineering)  used to enable [inter-process communication](https://en.wikipedia.org/wiki/Inter-process_communication) [object](https://en.wikipedia.org/wiki/Object_(computer_science)) creation in a large range of [programming languages](https://en.wikipedia.org/wiki/Programming_languages).

**comtypes** is a lightweight Python COM package. **comtypes** allows to define, call, and implement custom and dispatch-based COM interfaces in pure Python. It works on Windows, 64-bit Windows, and Windows CE.

Win32com vs comtypes discussion here.

**Task 3: Automate inserting objects and defining properties**

**Task 4: Export large datasets through scripting**

**Task 5: Import large datasets in to STK and parse data**

**Get certified!**