

Calling DLL Functions from BASIC+

A **DLL** (Dynamic Link Library) is a library of functions that can be loaded and executed at runtime. These functions are typically written in C/C++. DLLs are usually identified by the ***.DLL** extension (for example, COMMDLG.DLL). However, a DLL may also have an ***.EXE** extension.

Functions within a DLL can be called from the OpenInsight command line, in the same manner as other BASIC+ stored procedures.

Calling a function in a DLL from BASIC+ is a three step process. These steps are as follows:

- 1. . Create the DLL function prototype record.
- 2. .Use Declare_FCNS to create BASIC+ callable functions from the prototype record.
- 3. . Declare the function within BASIC + code, and use it like any other BASIC+ function.

This chapter provides you with an overview of the process of creating the DLL function prototype record, creating a callable function, and calling the function from BASIC+.

Creating the Prototype Record
Creating BASIC+ Callable Functions
Calling a Function in a DLL from BASIC+

Related Topics

For more information about the functions that are mentioned in this chapter, including GetPointer, GetValue, LockVariable, and UnlockVariable, refer to Chapter 3: BASIC+ Command Reference.

Search this documentation

Recent space activity



[Corby Nagel](#)
[Calling DLL Functions from BASIC+](#) updated Dec 03, 2015 [view change](#)
[GetSystemDirectory\(\)](#) created Oct 09, 2015
[GetWindowsDirectory\(\)](#) created Oct 09, 2015
[BASIC + Strings](#) created Oct 09, 2015
[Examples](#) created Oct 09, 2015