

# Calling Node.js Modules via Proxy Programs

## About Proxy Programs

A Proxy Program object automatically transfers control to a Profound.js module that is named the same as the Proxy Program and is located in the same *modules* subdirectory as the library in which the Proxy Program is located. All parameters passed to the Proxy Program are automatically passed along to the Profound.js module. Parameters can be passed by reference or by value. Proxy Programs typically replace the object of a converted RPG program.

While a Proxy Program holds no application logic within itself, it offers the following benefits:

- Allows Node.js scripts to be called as if they were ILE programs
- Provides the option to use standard IBM i security and object level authority on Node.js code
- Allows you to assign an activation group to the called program
- Maintains program dependencies after an RPG program is converted to Node.js
- Allows for an agile, iterative, and incremental modernization process
- Allows Node.js modules to be called from RPG or other IBM i languages using a standard CALL operation

## Creating Proxy Programs

You can create a Proxy Program object by using the `PJSCRIPTPRXY` command located within your Profound.js installation library. The command takes the following parameters:

- **Library / Program** - Specifies the name of program object to create
- **Activation group** - Specifies the activation group the program is associated with when it is called; default value is `*NEW`
- **Creation command options** - Specifies a parameter string, enclosed in apostrophes, that is added to `CRTPGM` command when the program is created

Once the object is created, you can work with it like any other program object. For example, you can use the `WRKOBJ` (Work with Objects) command to locate it. Or, you can use the `EDTOBJAUT` (Edit Object Authority) command to modify the program's authority settings.

## Calling Proxy Programs

You can call a Proxy Program from [Genie](#) by using the standard `CALL` command or by selecting the option that called the original RPG program that you've now replaced with the Proxy Program. This will invoke the respective Node.js module. The Node.js script may either present an interactive user interface or just perform back-end business logic, in which case control is immediately passed back to Genie after the Node.js script finishes executing.

Before calling the Proxy Program, your library list should include libraries `PROFOUNDUI` and `PROFOUNDJS` or their equivalents if you installed Profound UI and Profound.js into non-default libraries.

## Video Tutorial

For more information on how to set up Modules, please see [Creating Callable Profound.js Modules](#).