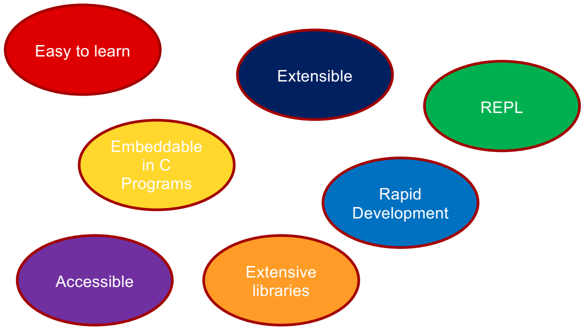
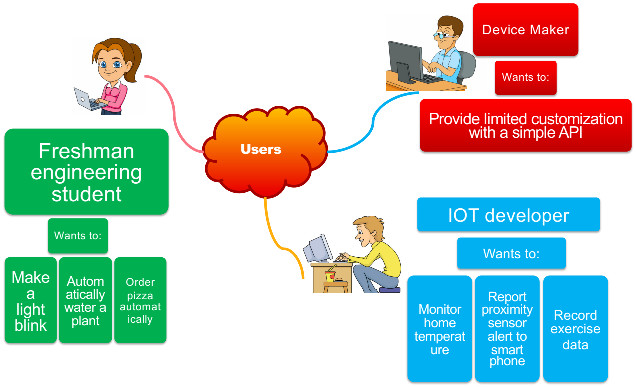
Supporting Scripting Languages on TI SimpleLink™ MCUs

Why Scripting?



1. Not everyone works/thinks in C
2. Out-of-box experience is slow/tedious/esoteric
3. Abundance of scripting languages

Target Markets


Users
Freshman engineering student
Wants to:
Make a light blink
Automatically water a plant
Order pizza automatically
Device Maker
Wants to:
Provide limited customization with a simple API
IOT developer
Wants to:
Monitor home temperature
Report proximity sensor alert to smart phone
Record exercise data

Scripting on MCUs requires a language interpreter running on the hardware.

Instead of rewriting existing C libraries, we “glue” the libraries into the interpreter.

> *Interpreters provide an API for this exact purpose.*

The scripting user needs a way to call our C libraries.

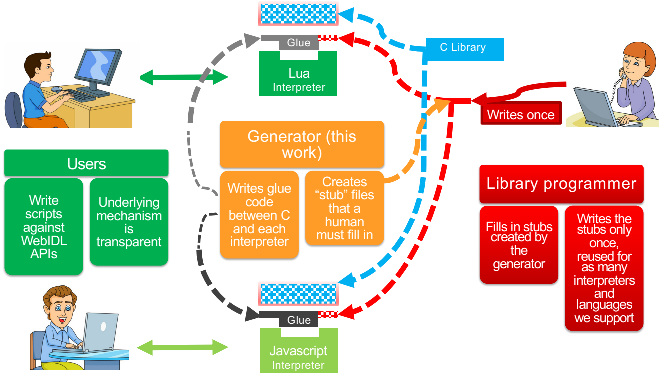
> *We use an Interface Definition Language (here, WebIDL) to provide an abstraction.*

The ”glue” code can be long and repetitive.

> *We generate most of it automatically*

Overarching design goal:

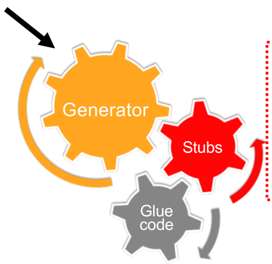
Minimize the amount of new code necessary to support new scripting languages.


Glue
Lua Interpreter
C Library
Writes once
Generator (this work)
Users
Write scripts against WebIDL APIs
Underlying mechanism is transparent
Writes glue code between C and each interpreter
Creates “stub” files that a human must fill in
Library programmer
Fills in stubs created by the generator
Writes the stubs only once, reused for as many interpreters and languages we support
Glue
Javascript Interpreter

The “generator”

```
Interface SimpleCalc {
    long add     (long x, long y);
    long subtract(long x, long y);
}
```

Consumes WebIDL



```
#include SimpleCalc.h

long add (long x, long y)
{
    //USER CODE GOES HERE
} /* add */

long subtract (long x, long y)
{
    // USER CODE GOES HERE
} /* subtract */
```

Creates 2 sets of C files:

1. The “stubs” that encapsulate the abstraction of the WebIDL and hide details of interaction with the interpreter
2. The ”glue code” to connect the stubs to the interpreter

```
#include SimpleCalc.h

Create_interface_Simplecalc {

    /* GENERATED AUTOMATICALLY

} /* create_interface_Simplecalc */
```

Authors: Mark Grosen, Timothy J. Harvey, Todd Waterman