

# Make extension functionality usable from Lua



**WIKIMEDIA**  
DEUTSCHLAND

# Scribunto

- Extension to MediaWiki that provides Lua support
- Communities can put own Lua code on *Module: pages*
- “Default” Lua libraries are in Scribunto
  - But extensions can add their own



# Installing Scribunto

Very easy, only one or two steps required besides loading the extension itself.

See [mw:Extension:Scribunto#Manual\\_installation](#)

# PageImages extension

- “The PageImages extension collects information about images used on a page.”
- Images retrievable via an API, but not via Lua (or a parser function)



# Adding Lua functionality to Pagelimages



**WIKIMEDIA**  
DEUTSCHLAND

# Create a PHP class for Scribunto

- `class Scribunto_LuaPageImagesLibrary extends Scribunto_LuaLibraryBase`
- Registers the Lua-library code and PHP callbacks

# Scribunto\_LuaPageImagesLibrary

```
public function register() {  
    $lib = [  
        'getPageImage' => [ $this, 'getPageImage' ],  
    ];  
  
    return $this->getEngine()->registerInterface(  
        __DIR__ . '/mw.pageImages.lua', $lib, []  
    );  
}
```

# Scribunto\_LuaPageImagesLibrary

```
public function getPageImage( $titleText ) {  
    $title = Title::newFromText( $titleText );  
    if ( !$title ) {  
        return [ null ];  
    }  
  
    $file = PageImages::getPageImage( $title );  
  
    return [ $file ? $file->getName() : null ];  
}
```



# Register the Scribunto Libraries

```
public static function onScribuntoExternalLibraries( $engine, array &$extraLibraries ) {  
    if ( $engine === 'lua' ) {  
        $extraLibraries['mw.pageImages'] = Scribunto_LuaPageImagesLibrary::class;  
    }  
}
```

# mw.pagelimages.lua

- Provides “setupInterface” method to Scribunto
- Has access to PHP callbacks (via local *php* variable)



# mw.pageImages.lua

```
local pageImages = {}

function pageImages.setupInterface()
  → local php = mw_interface
  → mw_interface = nil

  → -- ...

  → mw = mw or {}
  → mw.pageImages = pageImages
  → package.loaded['mw.pageImages'] = pageImages

  → pageImages.setupInterface = nil
end

return pageImages
```

# mw.pageImages.lua

```
function pageImages.setupInterface()
```

```
→ --- ...
```

```
→ function pageImages.getPageImage( titleText )
```

```
→ → return php.getPageImage( titleText )
```

```
→ end
```

```
→ --- ...
```

```
end
```



# It works

Debug console

- \* The module exports are available as the variable "p", including unsaved modifications.
- \* Precede a line with "=" to evaluate it as an expression or use print(). Use mw.logObject() for tables.
- \* Use mw.log() and mw.logObject() in module code to send messages to this console.

```
mw.log( mw.pagelimages.getPageImage( 'PageImageTest' ) )
```

```
Anunci_Wikimedia_Hackathon_Barcelona_3.png
```

```
mw.log( mw.pagelimages.getPageImage( 'Blah' ) )
```

```
nil
```



WIKIMEDIA  
DEUTSCHLAND

# Testing



**WIKIMEDIA**  
DEUTSCHLAND

# PHP part

- `class` Scribunto\_LuaPageImagesLibraryTest  
`extends` Scribunto\_LuaEngineTestBase
- Registers Lua tests with Scribunto
- (optional) Extension specific bootstrapping
- (optional) PHPUnit tests for the PHP callbacks



# Scribunto\_LuaPageImagesLibraryTest

```
class Scribunto_LuaPageImagesLibraryTest extends Scribunto_LuaEngineTestBase {  
  
    protected static $moduleName = 'LuaPageImagesLibraryTests';  
  
    # ...  
  
    protected function getTestModules() {  
        return parent::getTestModules() + [  
            'LuaPageImagesLibraryTests' => __DIR__ . '/LuaPageImagesLibraryTests.lua',  
        ];  
    }  
}
```



# LuaPageImagesLibraryTests.lua

```
local testframework = require 'Module:TestFramework'
```

```
local tests = {  
  → { name = 'mw.pageImages.getPageImage', func=mw.pageImages.getPageImage,  
  →   args = { 'User:Scribunto_LuaPageImagesLibraryTest' },  
  →   expect={ 'Example.jpg' }  
  → },  
  → { name = 'mw.pageImages.getPageImage no-such-page', func=mw.pageImages.getPageImage,  
  →   args = { 'No-Such-Page' },  
  →   expect={ nil }  
  → },  
}  
  
return testframework.getTestProvider( tests )
```

# Further reading

- [mw:Manual:Hooks/ScribuntoExternalLibraries](https://mw:Manual:Hooks/ScribuntoExternalLibraries)
- [mw:Extension:Scribunto/Lua reference manual#Writing Scribunto libraries](https://mw:Extension:Scribunto/Lua%20reference%20manual#Writing%20Scribunto%20libraries)
- <https://gerrit.wikimedia.org/r/434009>
  - Gerrit change containing the (not quite polished) code presented (in PS2)
  - *Might change substantially from there*
- [commons:File:Make extension functionality usable from Lua.pdf](https://commons.wikimedia.org/wiki/File:Make_extension_functionality_usable_from_Lua.pdf)

