

高级编译

常规构建过程使用Google的在线JavaScript编译器将Blockly减少到六个文件，总计约720kb（压缩后的160kb）。另外，也可以在“高级编译”模式下使用Google的离线JavaScript编译器，它具有许多优点：

- 由于文件结构改变，总块大小减少到300kb（压缩了100kb）
- 由于本地编译器的执行，构建时间更快且没有网络流量
- 无限编译（在线编译器受速率限制）

设置

出于本教程的目的，首先在Blockly根目录中创建一个新目录。

下载Closure编译器

下载closure-compiler-vxxxxxxx.jar，将其重命名为closure-compiler.jar，并将其放置在您的目录中。

通过在命令行上运行以下代码，验证Java Runtime Environment是否可以运行编译器：

```
java -jar closure-compiler.jar --version
```

安装Closure库

将Closure库的副本下载到Blockly根目录的同级目录中。

Boiler Plate

首先，创建一个HTML文件，该文件定义一个最小的Blockly工具箱和一个将其注入的div。为此，请在目录中创建一个文件index.html，其中包含以下代码：

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Blockly: Advanced Compilation</title>
  <script src="main_compressed.js"></script>
</head>
<body>
  <h1>Blockly: Advanced Compilation</h1>
  <div id="blocklyDiv" style="height: 480px; width: 600px;"></div>
  <xml id="toolbox" style="display: none">
    <block type="controls_if"></block>
    <block type="logic_compare"></block>
    <block type="controls_repeat_ext"></block>
    <block type="math_number"></block>
    <block type="math_arithmetic"></block>
    <block type="text"></block>
    <block type="text_print"></block>
```

```
</xml>
</body>
</html>
```

其次，创建一个JavaScript文件，该文件会加载Blockly和任何必要的消息文件或块定义，然后将Blockly注入提供的div中。为此，请在目录main.js中创建一个包含以下代码的文件：

```
goog.provide('Main');
// Messages (in some language)
goog.require('Blockly.Msg.en');
// Core
goog.require('Blockly');
// Blocks
goog.require('Blockly.Constants.Logic');
goog.require('Blockly.Constants.Loops');
goog.require('Blockly.Constants.Math');
goog.require('Blockly.Constants.Text');

Main.init= function() {
  Blockly.inject('blocklyDiv', {
    'toolbox': document.getElementById('toolbox')
  });
};
window.addEventListener('load', Main.init);
```

编译

通过从命令行运行Closure编译器，一起编译main.js，Blockly和Closure库：

```
java -jar closure-compiler.jar --js='main.js' \
  --js='../blocks/**/*.js' \
  --js='../core/**/*.js' \
  --js='../generators/**/*.js' \
  --js='../msg/js/**/*.js' \
  --js='../closure-library/closure/goog/**/*.js' \
  --js='../closure-library/third_party/closure/goog/**/*.js' \
  --generate_exports \
  --externs ../externs/svg-externs.js \
  --compilation_level ADVANCED_OPTIMIZATIONS \
  --dependency_mode=PRUNE --entry_point=Main \
  --js_output_file main_compressed.js
```

将浏览器指向index.html以验证一切正常。

请注意，Closure Compiler将许可证保留在已编译的输出中。可以从此输出文件中随意剥离Google的Apache许可证，以进一步减小文件大小。

Closure编译器具有很多功能和选项，请查看其文档。

