<https://www.oschina.net/news/61942/cpp-json-compare?utm_source=tuicool>

坊间有非常多的 C/C++ JSON 库，怎么选择是一个难题。

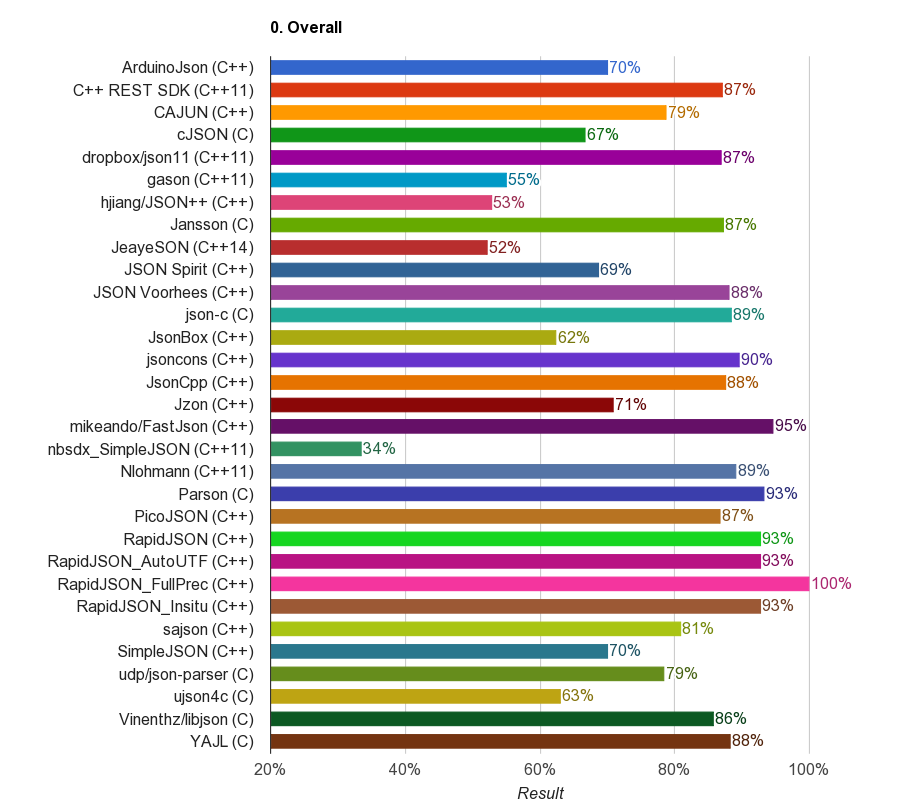
[nativejson-benchmark](https://github.com/miloyip/nativejson-benchmark)应该是史上第一个评测了28个C/C++开源JSON程序库的开源项目，它评测了每个库是否符合 JSON 标准 ([RFC7159](http://www.ietf.org/rfc/rfc7159.txt), [ECMA-404](http://www.ecma-international.org/publications/standards/Ecma-404.htm))，并且在性能上测试了各个操作（如解析、生成）的耗时、内存用量等数据，甚至可执行文件的大小也有统计。

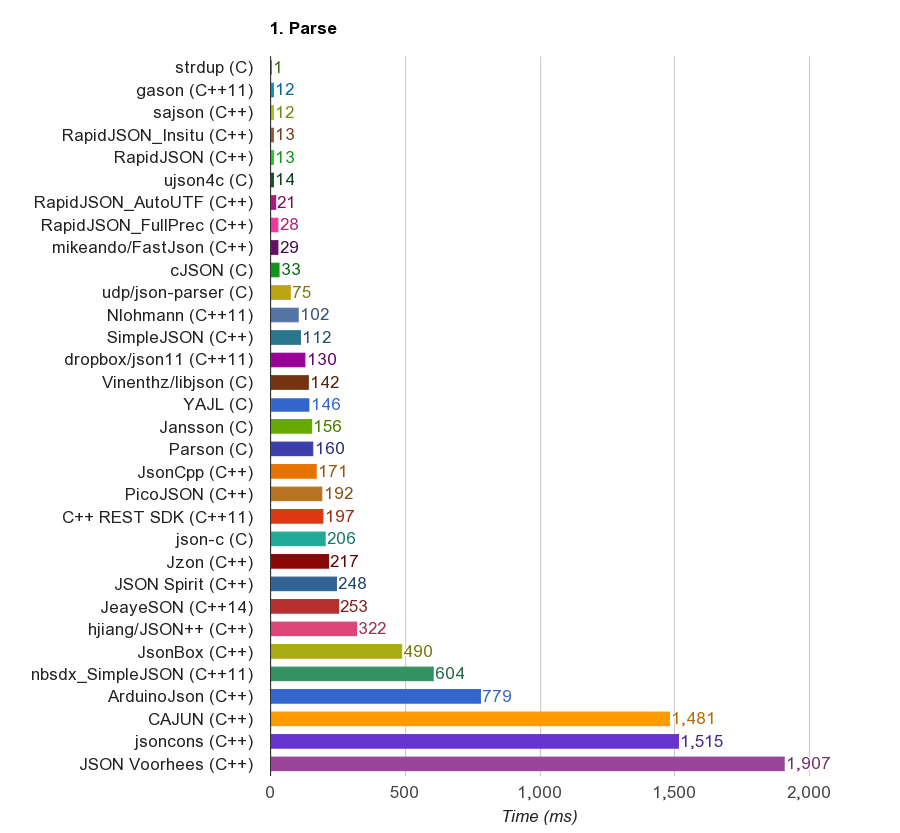
这个项目也希望能够通过互相学习借镜，提升现时或未来的 JSON 程序库品质。

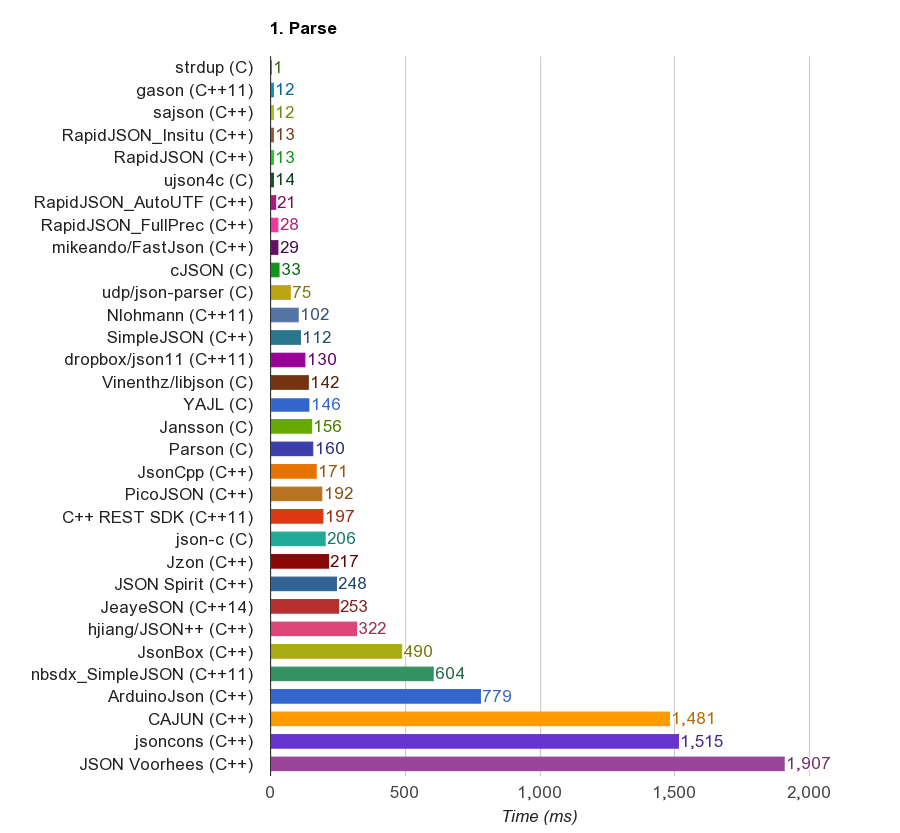
由于每个评测也不能完全覆盖所有数据及软硬件组合，建议使用者可自行在所需的平台和数据下进行测试。

以下是 iMac (Corei5[-](mailto:Corei5-3330S@2.70GHz)3330S@2.70GHz) 及clang 6.1\_1 64-bit下的部分结果，详情可访问[这里](https://rawgit.com/miloyip/nativejson-benchmark/master/sample/conformance.html)。

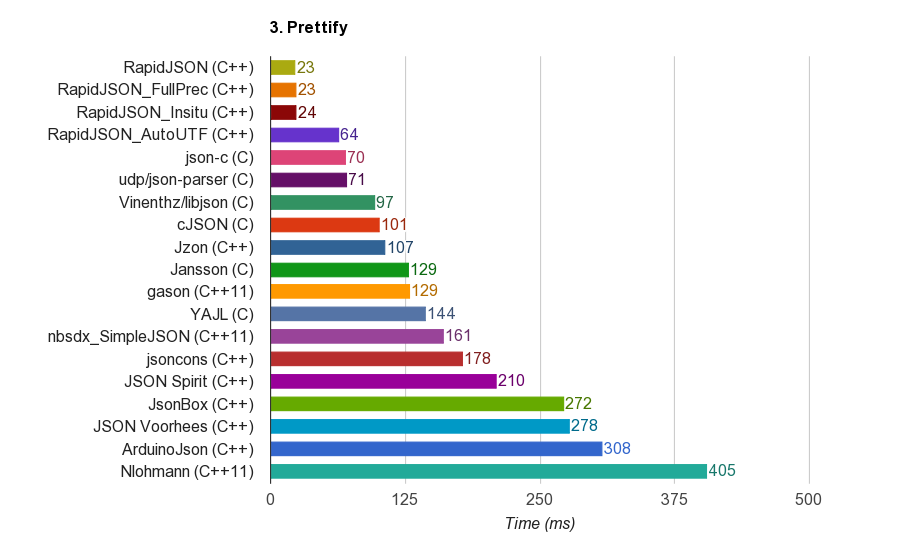
符合标准程度（越高越好）：

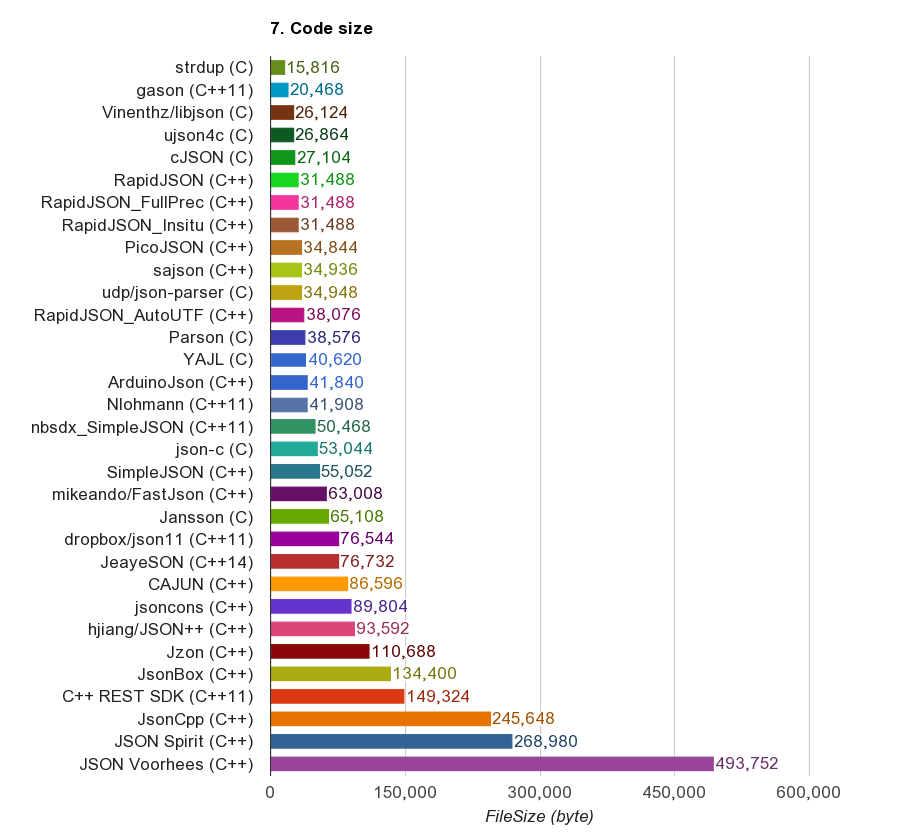


解析至DOM的时间（越低越好）：解析至DOM后的内存（越低越好）：

把DOM生成JSON的时间（越低越好）：

把DOM生成含换行及缩进的JSON的时间（越低越好）：



可执行文件（把JSON解析至DOM，然后统计JSON类型）的大小（越低越好）：

相关链接

* RapidJSON 的详细介绍：[点击查看](https://www.oschina.net/p/rapidjson)
* RapidJSON 的下载地址：[点击下载](https://www.oschina.net/home/login?goto_page=https%3A%2F%2Fwww.oschina.net%2Fnews%2F61942%2Fcpp-json-compare)