Parsing Gigabytes of JSON per Second

Geoff Langdale,

Daniel Lemire, Université du Québec (TÉLUQ)

Montreal 🇨🇦

Web: https://simdjson.org

twitter: @lemire

GitHub: https://github.com/simdjson/simdjson

How fast is your disk?

PCIe 4 disks: 5 GB/s reading speed (sequential)

benchmark: hothardware.com 📰

✏️ *Network speeds of 50 GB/s (400GbE) and better are coming near you.*

Unless you can eat data at gigabytes per second, you may be CPU bound when reading from disk!!!

😟😟😟😟

JSON

- Specified by Douglas Crockford
- RFC 7159 by Tim Bray in 2013
- Ubiquitous format to exchange data

```
{"Image": {"Width":  800,"Height": 600,
"Title":  "View from 15th Floor",
"Thumbnail": {
    "Url":    "http://www.example.com/81989943",
    "Height": 125,"Width":  100}
}}
```

RapidJSON

- High speed, standard compliant, C++
- 0.3 GB/s (Skylake 3.4GHz, GNU GCC8, file: twitter.json)

getline

```
size_t sumofalllinelengths{0};
  while(getline(is, line)) {
    sumofalllinelengths += line.size();
  }
```

1.4 GB/s (Skylake 3.4GHz, GNU GCC8, file: twitter.json)

simdjson

2.5 GB/s (Skylake 3.4GHz, GNU GCC8, file: twitter.json)

- Full JSON and UTF-8 validation, lossless parsing.
- Selects a CPU-tailored parser at runtime. No configuration needed.

Where to get simdjson?

- https://simdjson.org
- GitHub: https://github.com/simdjson/simdjson/
- Modern C++, single-header (easy integration)
- 64-bit ARM (e.g., iPhone), x64
- Apache 2.0 (no hidden patents)
- wrappers in Python, PHP, C#, Rust, JavaScript (node), Ruby
- ports to Rust, Go and C#
- Available from Debian, brew, conan, vcpkg
- Linux, macOS, Windows, FreeBSD