|  |  |
| --- | --- |
|  | **2014** |
|  | The Company  Martien Huijsmans |

|  |
| --- |
| **[High performance]** |
| Todo |

Inhoud

[Articles 1](#_Toc397983350)

[Loopback 1](#_Toc397983351)

[TCP Options 1](#_Toc397983352)

[Technology Options 2](#_Toc397983353)

[Tools 2](#_Toc397983354)

[TCP Networking packages 2](#_Toc397983355)

# Measurements

With ipref measured loopback performance (about 400 MBytes/sec) on windows.  


# Aspects (in arbitrary order)

* Cache (misses)
* Data centric algorithms, i.e. keep data in cache
* Contiguous memory access allows CPU to pre-fetch, pre-execute, etc.
* memory alignment, i.e. at 16 bits
* SSE/SIMD
  + has own optimal memory alignment , i.e. 4-bytes

# Articles

Article on: Enabling High Performance Data Transfers  
<http://www.psc.edu/index.php/networking/641-tcp-tune>   
Includes

* Programming guides
* Tooling
* System optimization for different platforms, including Linux

Blog: <http://fastcpp.blogspot.nl/> on high performance computing with link to

* what every programmer should know about cpu, cache, memory, etc . <http://www.akkadia.org/drepper/cpumemory.pdf>
* examples of SSE/SIMD

Effects of cache: code examples, measurements and diagrams

* http://igoro.com/archive/gallery-of-processor-cache-effects/

# Loopback

<http://stackoverflow.com/questions/11276002/what-can-be-expected-in-terms-of-latency-with-tcp-ip-over-the-loopback-interface?lq=1>  
ping localhost gives impression on lowerbound

* On windows 7, TTL was <=0 ms
* On Fedora VM on win7, ttl was 0.o77 ms avg.

# TCP Options

TCP\_NODELAY  
TCP\_CROCK  
WSACLE: RFC1323, These enable optional TCP protocol features (window scale and time stamps) which are required to support large BDP paths.

# Technology Options

* Tcp/ip
* Unix sockets
* pipes
* Shared memory

This articles explains that Unix domain sockets perform better than sockets or pipes.  
<http://bhavin.directi.com/unix-domain-sockets-vs-tcp-sockets/>   
With Unix domain sockets sender would write directly into the buffer of the receivers.  
No references are provided.

# Tools

Nuttcp: <http://www.nuttcp.net/nuttcp/Welcome%20Page.html>

<https://iperf.fr/>   
Iperf was orginally developed by NLANR/DAST as a modern alternative for measuring TCP and UDP bandwidth performance.

# TCP Networking packages

Boost::aio

Poco: <http://pocoproject.org/>