

@W3C Web & Networks IG

LPP Network Trace Tools

Jonas Svennebring

March 9 – 2021



intel®

Legal Notices/disclaimer

- Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer.
- No computer system can be absolutely secure.
- Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors.
- Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit <http://www.intel.com/performance>.
- Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit <http://www.intel.com/performance>.
- Cost reduction scenarios described are intended as examples of how a given Intel- based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.
- Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.
- Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.
- Intel, the Intel logo and others are trademarks of Intel Corporation in the U.S. and/or other countries. *Other names and brands may be claimed as the property of others.
- © 2021 Intel Corporation.

Agenda

- Network Trace Tools Overview
 - Trace Emulator
 - Trace Capture
 - Trace Format & Library of network trace files
- LPP Predictions with Trace Emulator
- Next Step

- Backup & additional details

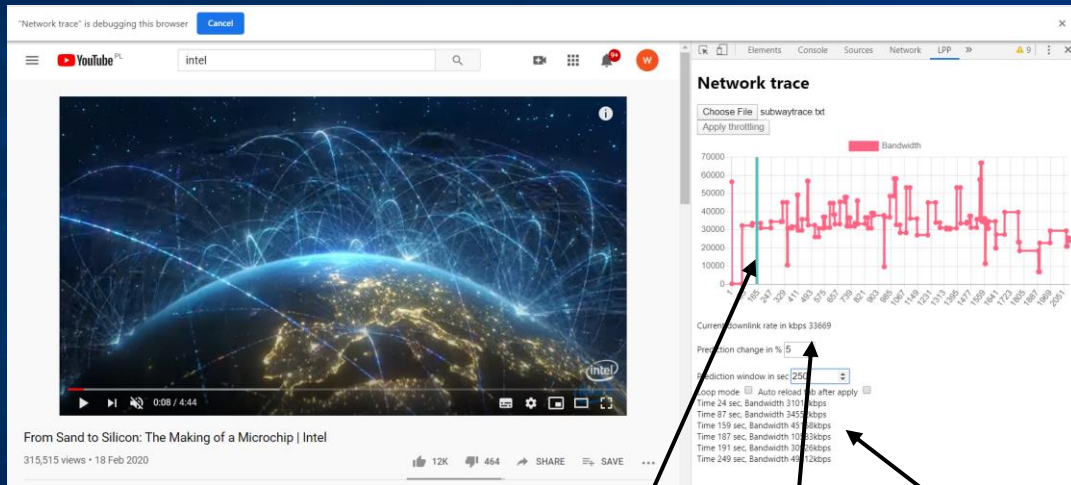
Background - full LPP TPAC-19 presentation:

<https://www.w3.org/2019/09/17-web-networks-lpp.pdf>

Trace tools Overview

A collection of browser based dev tools to emulate network behaviors

- Trace emulator
- Trace collector
- Library of trace files
- Trace format specification



GPL2 License – feel free to contribute
github.com/intel/lpp-network-trace

Current
time

Current
bandwidth

LPP
predictions

Emulation

Trace Emulation

Network Trace Emulator is a **Chromium** development tool extension enabling emulation of network conditions based on pre-recorded trace files.

- A. Enables a simple way to test any web application when put through different network conditions.
- B. The emulator allows for development and test of web applications that leverages LPP predictions for forward looking network conditions.

Trace Emulation

Mat din internethastighet | Fast.com

fast.com/sv/

Network Trace började felsöka webbläsaren [Avbryt](#)

English (UK)

FAST

Your internet speed is

12 Mbps

[Show more info](#)

?

f

t

Network Trace Emulator

Valj fil: car_trac4 ntrace

Apply throttling Pause Cancel

Bandwidth

Time in sec: 12608

Current downlink rate in kbps 12608

Generate prediction if difference to previous value is more than 5 %

Prediction window in sec 60

Loop mode Auto reload tab after apply

Time 0 sec	Bandwidth 12608kbps
Time 7.66 sec	Bandwidth 7641kbps
Time 16.96 sec	Bandwidth 11532kbps
Time 26.56 sec	Bandwidth 16551kbps
Time 38.78 sec	Bandwidth 50116kbps
Time 47.18 sec	Bandwidth 10540kbps
Time 56.86 sec	Bandwidth 12362kbps

Djurklint

Ojupdalvägen

Capture

Network Capture Tool

Description*
Trip from A to B

Client model
SM-G390

Client name
Samsung S10e

Additional information
Demo trace collection

Download bandwidth test interval (sec)*
10

Download bandwidth test max duration (sec)*
4

Download limit (Kbytes)*
4096

URL resource*
http://192.168.1.1/10MB.data

Trace GPS position
 Keep screen on while tracing

START PAUSE STOP Stopped

Test progress:

Measured download bandwidth: 0 kbps

Network: Unknown

Effective type: 4g

Longitude: 0
Latitude: 0
Accuracy: 0

Recorded data as JSON:

Trace Capture

- Place the Trace tools on a web server
 - it does not run from local files
 - Optionally use https/ssl connection if GPS tracking is intended to be used
 - Or run simply run github version from:
<https://intel.github.io/lpp-network-trace/>
- Load trace tools in a Chromium based browser:
 - Open source Chromium
 - Google Chrome
 - Microsoft Edge (not older Explorer)
 - ...

Capture Trace

- Press "Start" button to activate trace collection, old results will be removed

- Pause will hold the test

- Stop writes out the results in JSON format, you can then copy them, manually, download or with copy button and then e.g. save to file

- Note: network effective connection type is what the browser reports as roughly estimated connection, this can be ignored



START PAUSE STOP Stopped

Test progress:

Measured download bandwidth: 0 kbps

Network: Unknown

Effective type: 4g

Longitude: 17.9830883
Latitude: 17.9830883
Accuracy: 2090

Recorded data as JSON:

```
{  
  "version": 1,  
  "description": "Trip from A to B",  
  "clientModel": "SM-G390",  
  "clientName": "Samsung S10e",  
  "note": "Demo trace collection",  
  "dlBwTestDuration": 4,  
  "dlBwTestDownloadLimit": 4096,  
  "dlBwTestUrl": "http://192.168.1.1/10MB.data",  
  "entries": [  
    {  
      "entryNo": 1,  
      "dateTime": "2021-03-04T19:49:13.274+01:00",  
      "dlBw": 0,  
    }  
  ]  
}
```

Trace format

Trace Format

Name	Description
version	Format version
description	Trace description (e.g. "From Stockholm to Szczecin")
clientModel	Client model (e.g. "SM-G390")
clientName	Client name (e.g. "Samsung Galaxy S10")
note	additional description (e.g. means of travel, location, residentail/rural/urban, weather info)

Name	Description
entryNo	Trace entry number in ascending order
dateTime	UTC timestamp in format "YYYY-MM-DDThh:mm:ssTZD" according to ISO 8601
network	Network operator name
dBw	Downlink bandwidth in kbps
dLLatency	Downlink latency to first bytes in microseconds
uLBw	Uplink bandwidth in kbps
uLLatency	Uplink latency to first bytes in microseconds
latitude, longitude, accuracy	GPS latitude, longitude in decimal degrees (DD) in WGS-84 format, accuracy in meters
note	Additional description

- flexible and extensible format that contains series of bandwidth data and (optionally) location
- JavaScript Object Notation (JSON)
- *.ntrace* file extension
- Trace header
 - general information
- Trace entries
 - each entry contains measured bandwidth and/or GPS coordinates

github.com/intel/lpp-network-trace/blob/main/Documentation/lpp-network-trace-format.md

Trace Format

```
{
  "version": 1,
  "description": "Telenor LTE, by car on large roads outside Stockholm",
  "clientModel": "SM-G970F",
  "clientName": "Samsung s10e",
  "note": "Around Kista/Stockholm on E18 and E4, BW server in Frankfurt/DE",
  "entries": [
    {
      "entryNo": 1,
      "dateTime": "2020-11-09T13:09:44.462+01:00",
      "dlBw": 53264,
      "gpsCoordinates": {
        "latitude": 59.4053376,
        "longitude": 17.9784013,
        "accuracy": 92.9000015258789
      },
      "network": "cellular (Effective connection type: 4g)"
    },
    {
      "entryNo": 2,
      "dateTime": "2020-11-09T13:09:54.104+01:00",
      "dlBw": 86964,
      "gpsCoordinates": {
        "latitude": 59.4054321,
        "longitude": 17.9764641,
        "accuracy": 19.111000061035156
      },
      "network": "cellular (Effective connection type: 4g)"
    },
    {
      "entryNo": 3,
      "dateTime": "2020-11-09T13:10:04.254+01:00",
      "dlBw": 40196,
      "gpsCoordinates": {
        "latitude": 59.4053716,
        "longitude": 17.9764878,
        "accuracy": 7.6620001792907715
      },
      "network": "cellular (Effective connection type: 4g)"
    },
    {
      "entryNo": 4,
      "dateTime": "2020-11-09T13:10:14.181+01:00",
      "dlBw": 92156,

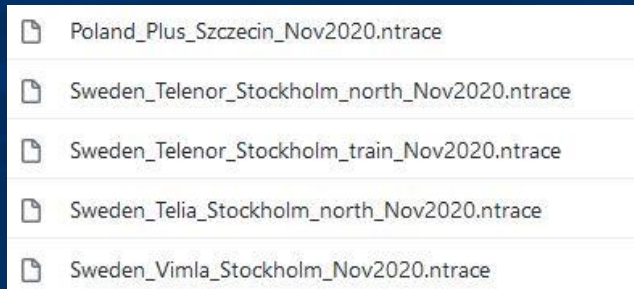
```

Library of trace files

Trace Library

- Pre-collected set of traces from different parts of the world

[Trace Library at github.com](https://github.com)



Poland_Plus_Szczecin_Nov2020.ntrace
Sweden_Telenor_Stockholm_north_Nov2020.ntrace
Sweden_Telenor_Stockholm_train_Nov2020.ntrace
Sweden_Telia_Stockholm_north_Nov2020.ntrace
Sweden_Vimla_Stockholm_Nov2020.ntrace

- Pandemic has hit our collection, but we anticipate to add a large set of relevant traces from around the world soon!
- Please feel free to collect and contribute your own traces

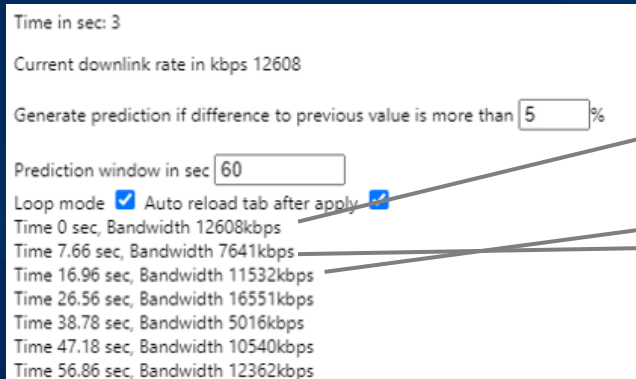
LPP Prediction Handling

LPP prediction using Java script API

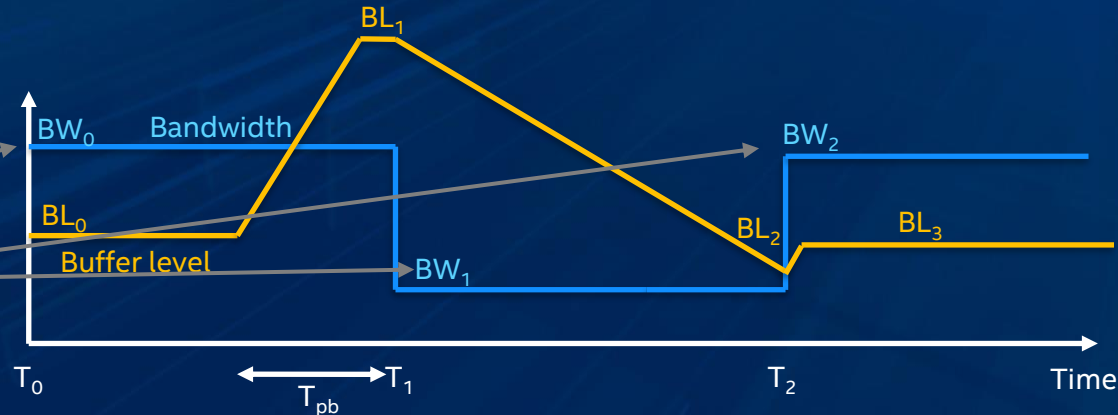
Network Trace Emulator also delivers LPP Java Script API that provides data about future predicted network conditions.

It can be used by a web application to test different network scenarios and adjust it based on predicted data.

What do I get?



How do I use it?



LPP prediction – create / subscribe

1. Create LppSession object using constructor

```
LppSession(LppPredictionOptions options)
```

where:

```
LppPredictionOptions = {  
    Boolean downlinkBandwidth; Boolean uplinkBandwidth;  
    Boolean latency;  
    Number maxPredictions;}
```

e.g.

```
lpp = new LppSession({ downlinkBandwidth: true, maxPredictions: 10 });
```

2. Subscribe to LPP Service using start method

```
lpp.start();
```

LPP prediction – get predictions

3. Get LPP predictions via handler

Handler for lpp event fired when data is received from LPP service connection.

```
lpp.onpredictions(data)
```

The data message will contain an array of predictions. There can be several predictions of different types at one specific point in time. The predictions are always in chronological order with the nearest prediction first in the array

```
data = {  
  predictions: [  
    {  
      time: <Number>,  
      type: <lppType>,  
      value: <Number>,  
      variation: <Number>,  
      probability: <Number>  
    }, {...}, ... ]];
```

time: Number of sec since midnight 1 January 1970.
type: LPP prediction type
value: Number or String depending on prediction type
variation: Standard deviation of the prediction
probability: Statistical probability of the prediction

LPP prediction - example

```
lpp = new LppSession({ downlinkBandwidth: true, maxPredictions: 10 });  
lpp.start();
```

```
lpp.onpredictions = (event) => {  
    alert(JSON.stringify(event.predictions))  
}
```

- **A simple sample code is available under /samples/test_chrome.html**

Next Step

LPP teams next step

These are our plans moving forward, but we welcome input and contributions for the wider community

- Extend emulator / trace collector with uplink and latency, other metrics desired?
- Extend library of trace files
- Publish our LPP MPEG-DASH ref. lib patches to showcase how the LPP predictions can be used to improve streaming, which can easily be tested and further developed in the emulator
 - Let us know if this is of interest for further collaboration

Backups & Additional details

Trace Emulation - installation

1. Open a Chromium based browser (e.g. Chrome, Edge, etc)
2. Enter `chrome://extensions` in address field.
3. On the top right corner, toggle the Developer Mode switch.
4. On the top left corner, click on Load Unpacked.
5. Locate and load the Emulator folder (that contains manifest.json file).
6. *[Optional] Go to `chrome://apps`.*
7. *[Optional] Remove Google apps related to website that emulator will be used on.*

The Network Trace Emulator will be available in the developer tools, accessed by pressing F12. Navigate the tabs in the Chrome developer window to find Network Trace.

Load a ntrace file and let it emulate the network for you

Remember that it can only throttle down the network, you can never get higher bandwidth than what you actually have

Trace Emulation - limitations

1. Emulation cannot be used for web application that makes use of WebSocket protocol due to chrome bug 423246.
2. To avoid issues in certain use cases e.g. using the extension on youtube, please remove the corresponding website application under chrome://apps – chrome bug 1126825.

intel®