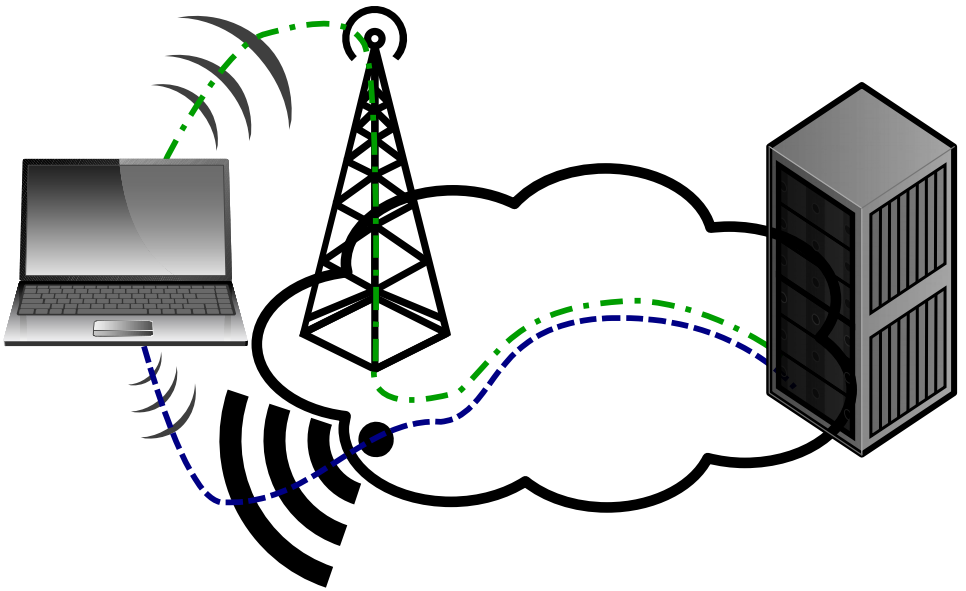
Path Awareness and Selection in the Socket Intents prototype

Theresa Enghardt
TU Berlin
theresa@inet.tu-berlin.de

Scenario: Multiple paths



Multiple paths via different access networks

- Laptop can use WiFi or cellular
- WiFi usually default, but not always better[1]

[1] Deng et al.: “WiFi, LTE, or Both? Measuring Multi-Homed Wireless Internet Performance” (2014)

→ Pick the better one? Use both?

Socket API

Connection 1

Connection 2

Connection 3

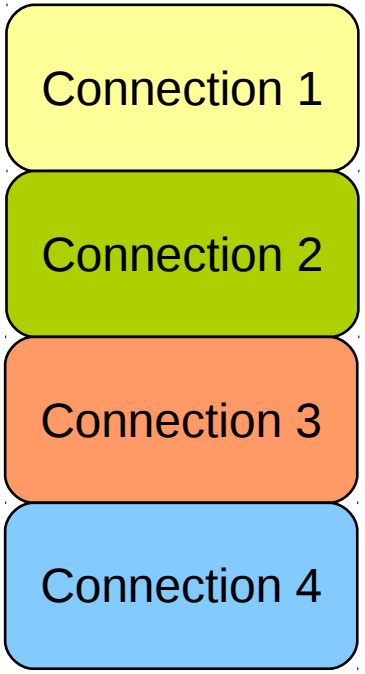
Connection 4

Vanilla BSD sockets:

- Connections „look the same“
- No information about paths

→ use **default path** based on system policy

Socket Intents



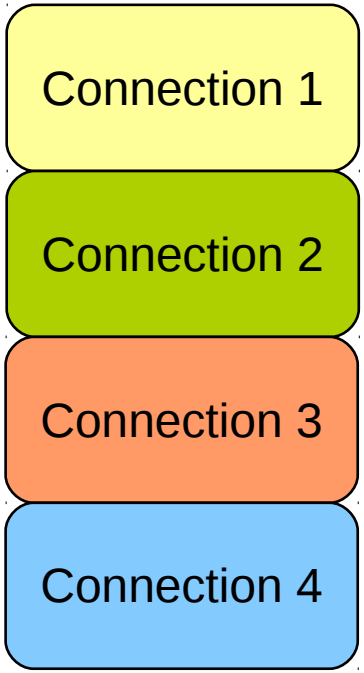
Intents:

What the application
wants,
knows,
prefers,
or *assumes* about its traffic
(connection or message)

→ What to optimize for

See our draft-tiesel-taps-socketintents

Socket Intents

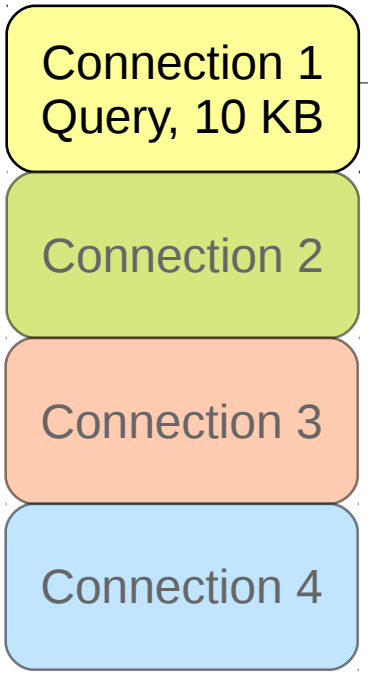


Intents:

- Traffic Category
- Size to be received
- Bitrate to send
- Timeliness
- Cost preferences

See our draft-tiesel-taps-socketintents

Socket Intents

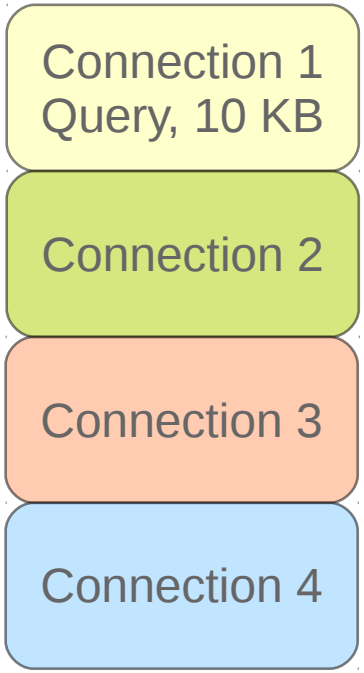


Connection 1:

- Traffic Category: **Query**
- Size to be Received: **10 KB**

See our draft-tiesel-taps-socketintents

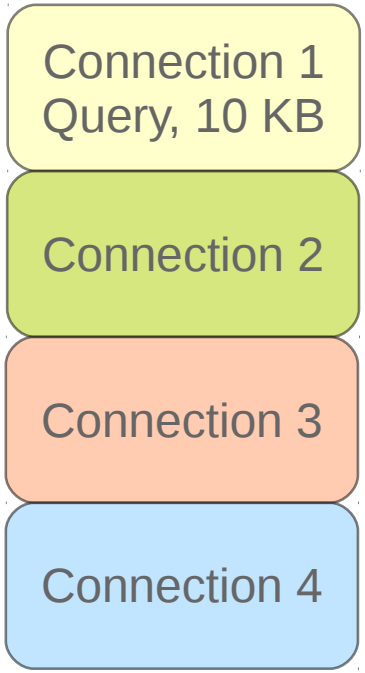
Socket Intents



Path property estimates:

- Median Round Trip Time (RTT)
- Maximum bitrate
- WiFi utilization
- ...

Socket Intents

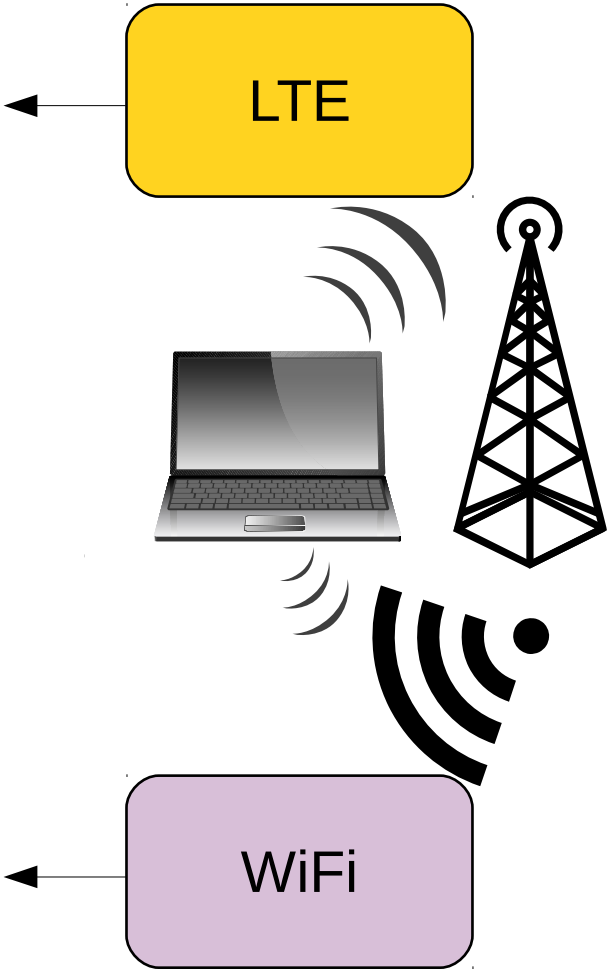


Path 1: LTE

- RTT = 100 ms
- Bandwidth = 20 Mbit/s

Path 2: WiFi

- RTT = 10 ms
- Bandwidth = 2 Mbit/s



Socket Intents

Connection 2

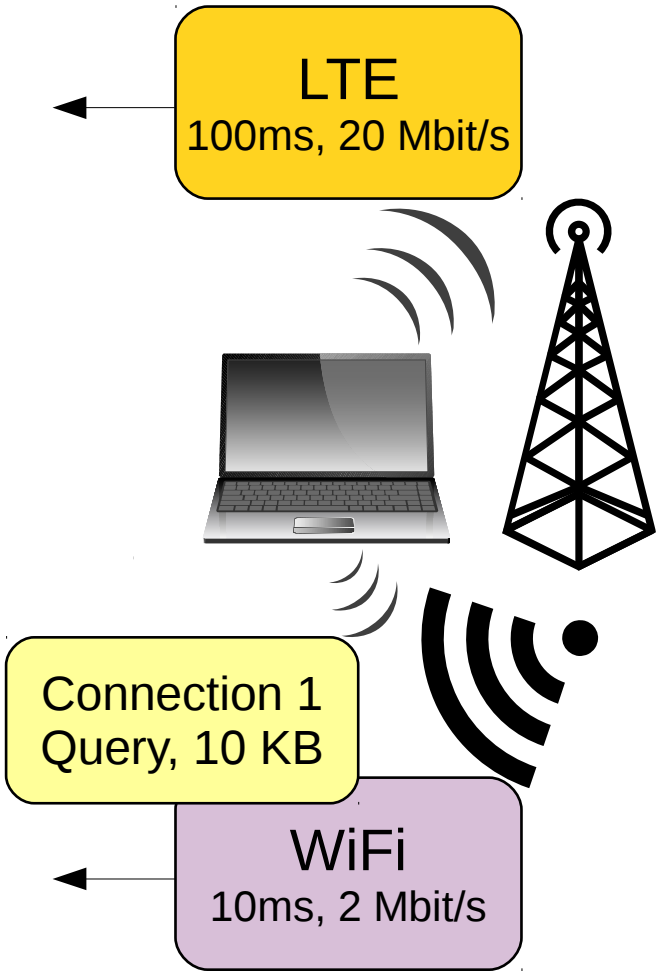
Connection 3

Connection 4

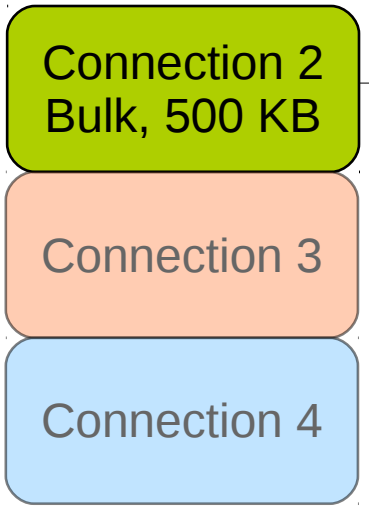
Path Selection Policy:

- “Use path with shorter completion time”
- LTE: ≈ 200 ms
- WiFi: ≈ 20 ms

→ **Use WiFi**

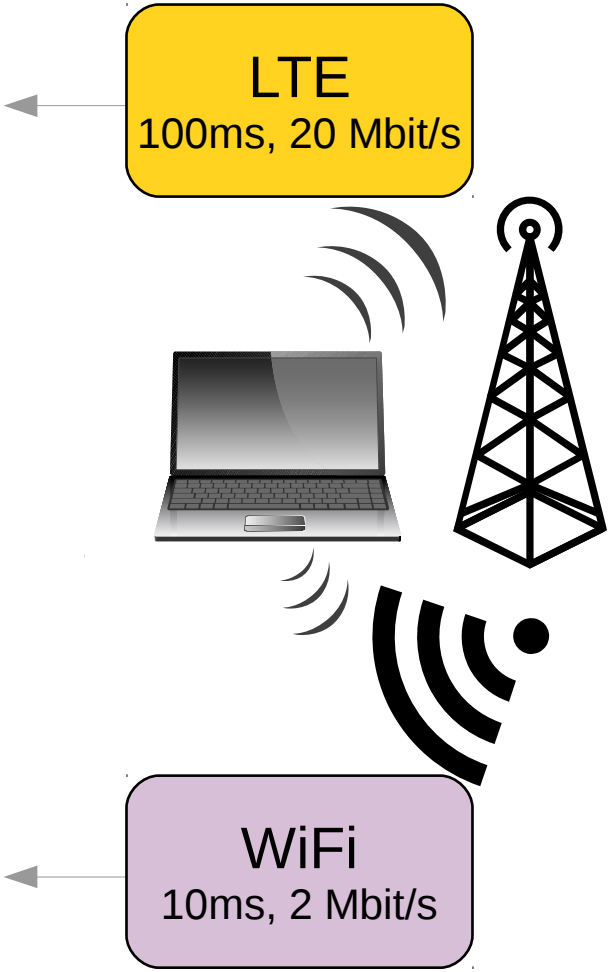


Socket Intents



Connection 2:

- Traffic Category: **Bulk**
- Size to be Received: **500 KB**



Socket Intents

Path Selection Policy:

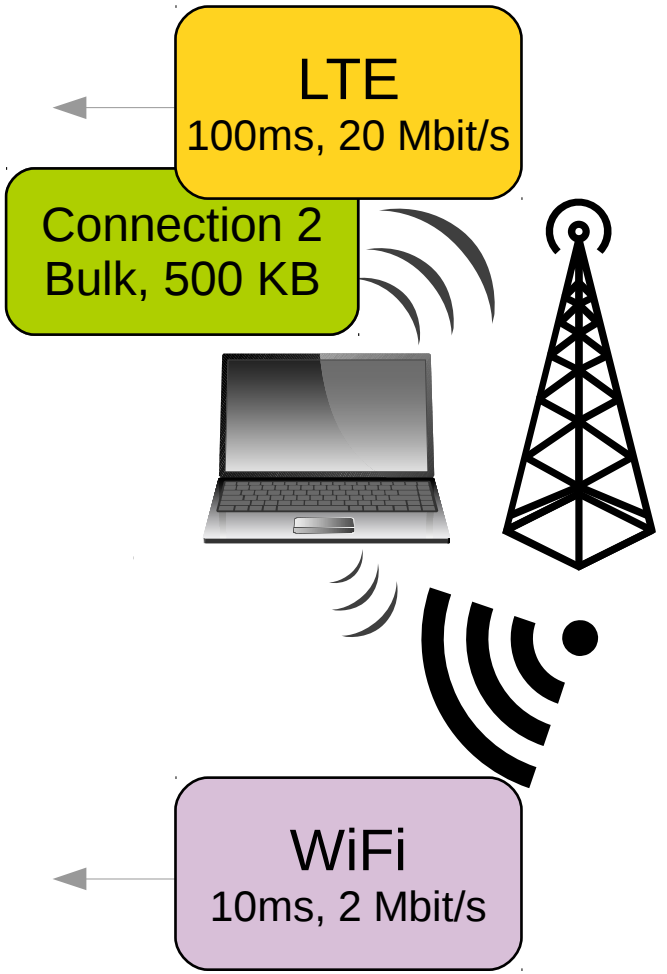
- “Use path with shorter completion time”
- LTE: ≈ 225 ms
- WiFi: ≈ 450 ms

→ **Use LTE**

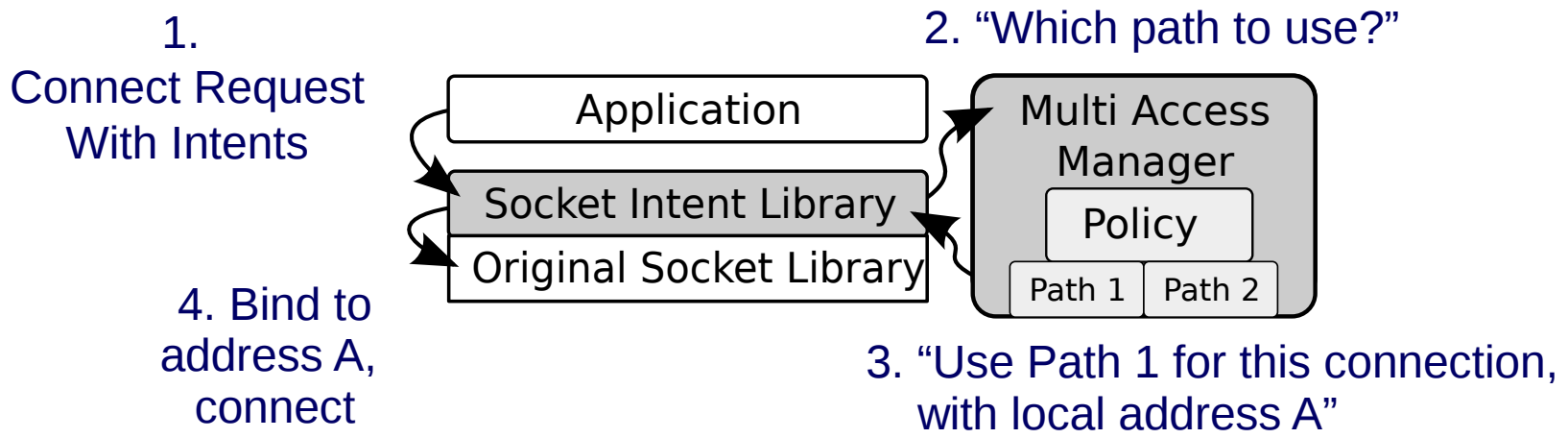
… or both (MPTCP)

Connection 3

Connection 4



Socket Intents Prototype



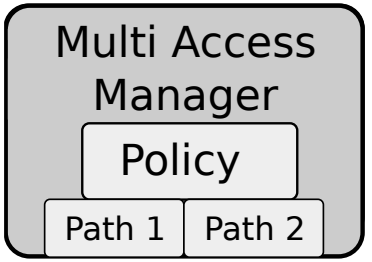
- Socket Intents Library: Augmented Socket API
- Multi Access Manager:
  - Standalone daemon with policy modules
  - Gathers current performance estimates
  - Chooses path and local address

See our draft-tiesel-taps-socketintents-bsdsockets
and code https://github.com/fg-inet/socket-intents

Paths

Multi Access Manager:

- Detects locally configured interfaces with their prefixes and addresses
- Gathers statistics on them passively, based on current and past traffic



	Path 1		Path 2	
Local prefix	a:a:a:a::a/64	1.2.3.4/24	b:b:b:b::b/64	5.6.7.8/24
Minimum RTT	95 ms	105 ms	8 ms	10.5 ms
Maximum Bitrate	18.9 Mbit/s		1.8 Mbit/s	
Utilization	N/A		2%	

Path Bitrate

- Bitrate per interface
  - Read interface counter every *n ms*[1]
  - $bitrate_{current} = \frac{counter_{current} - counter_{prev}}{n}$
  - bitrate_max: Maximum within the last *m minutes*[2]

    → estimate of bandwidth of the path
- Assumes the bottleneck on each path is within the first few hops

[1] 100 ms works for us
[2] 5 minutes works for us

Path RTT

- RTT per prefix
  - Linux kernel keeps list of current TCP connections
  - Each TCP connection has a current Smoothed RTT (SRTT)
  - Query SRTTs of all connections over prefix every *n ms*
  - Compute current mean, median or SRTTs
  - If no current values, retain values for up to *m minutes*
  - Compute minimum of the last *m minutes*
- We expect the first hop or first few hops to dominate latency

Radio properties on path

- For WiFi:
  - Current Received Signal Strength
  - Last observed modulation bitrates
  - Utilization: QBSS Information Element from Beacon frames
- Other wireless technologies possible, but hard

Current and Future Work

- Show page load time reduction for web browsing
- More path selection policies
- More path properties
  - RTT variation
  - Packet loss
  - Information from the network

Summary

- Socket Intents:
  - Application provides hints on Connection (or Message…)
  - We know what to optimize for
- Path properties:
  - Socket Intents prototype gathers them locally
  - Observed median SRTT, maximum bitrate, WiFi Utilization
- Path selection:
  - E.g. use Path with shortest expected completion time
  - Other optimization possible, e.g. for cost

See our draft-tiesel-taps-socketintents-bsdsockets
and code https://github.com/fg-inet/socket-intents