Fast Primer Search with DUP

Christian Grothoff
christian@grothoff.org

Technische Universität München

Overview

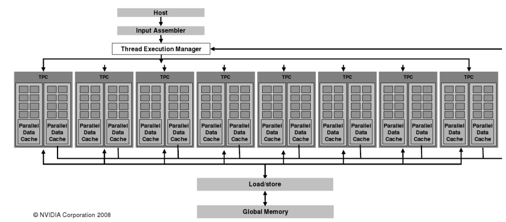
- **Research Area**
- Easy Distributed Stream Processing with DUP
- Case Study: Fast Primer Search[1]
  - Biological Questions
  - Primer Search Parallelization with DUP
  - Mapping Primers to Species with the BGRT
- Other Research Results

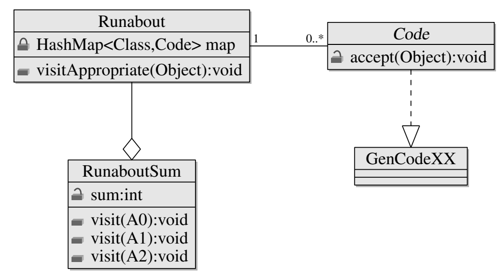
[1] "An algorithm for the comprehensive search of oligonucleotide signatures based on phylogenetic trees", joint work with K. Bader, W. Ludwig and H. Meier

fsnsg

Problem Domains

- Systems
- Programming Languages
- Software Engineering
- Secure Networking
- Privacy





fsnsg

The Problem: Developing Parallel Stream Applications

- Most developers (only) know how to write sequential code
- Parallel programing is error-prone (data races, deadlocks)
- High-performance parallel programming is really hard
- With GPUs for $4,000, we could have 2,600 cores...

$\Rightarrow$ Developers more expensive than hardware

X10 vs. the DUP System[2]

X10

10x faster, 10x as productive in 10 years for BlueGene

DUP

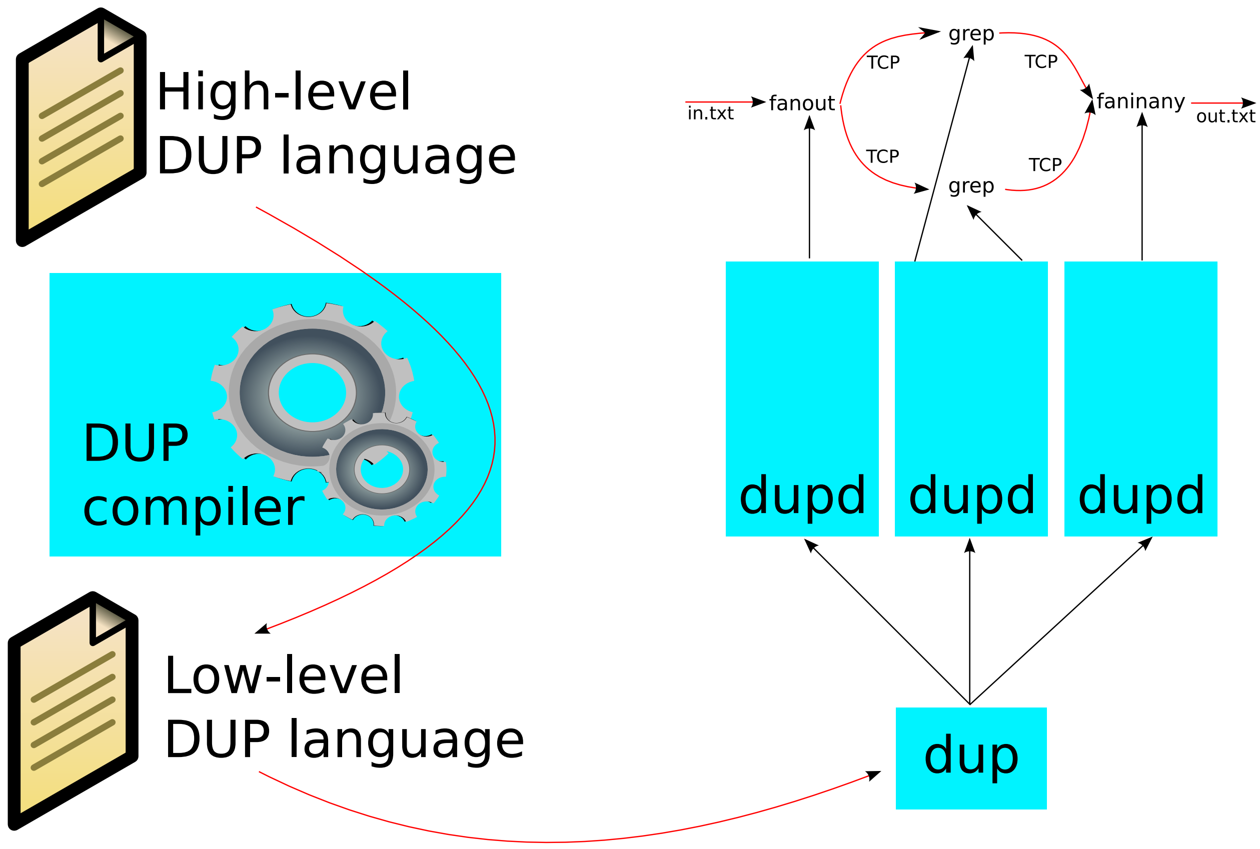
$\frac{1}{2}$ the speed, 10x as productive in 10 months for POSIX

[2] Available at `http://dupsystem.org/`

Agenda

- Research Area
- **Easy Distributed Stream Processing with DUP**
- Case Study: Fast Primer Search
  - Biological Questions
  - Primer Search Parallelization with DUP
  - Mapping Primers to Species with the BGRT
- Other Research Results

The DUP System



fsnsg

DUP Applications

- **Fast primer search**
- High-throughput, customizable video-conferencing
- Parallel and distributed discrete event simulation framework

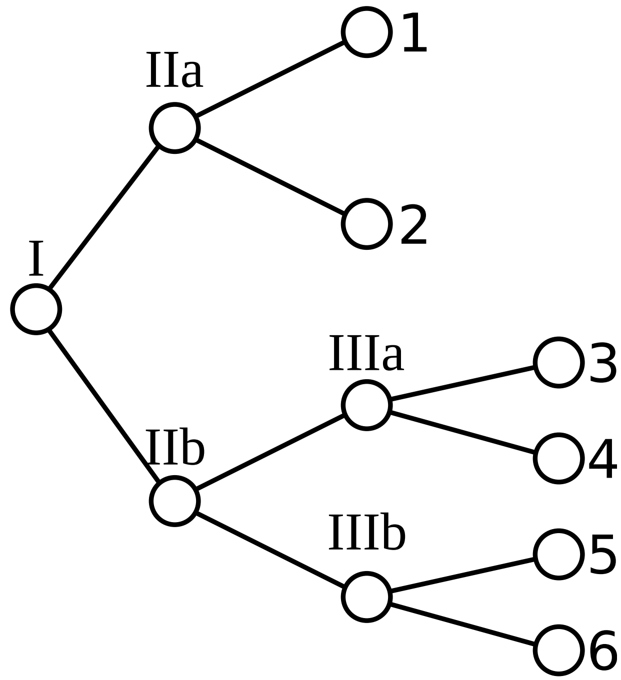
Agenda

- Research Area
- Easy Distributed Stream Processing with DUP
- **Case Study: Fast Primer Search**
  - Biological Questions
  - Primer Search Parallelization with DUP
  - Mapping Primers to Species with the BGRT
- Other Research Results

Biological Questions

- Which are the most specific OSSs for a species?
- Which are the most specific OSSs for a subtree in the phylogenetic tree?

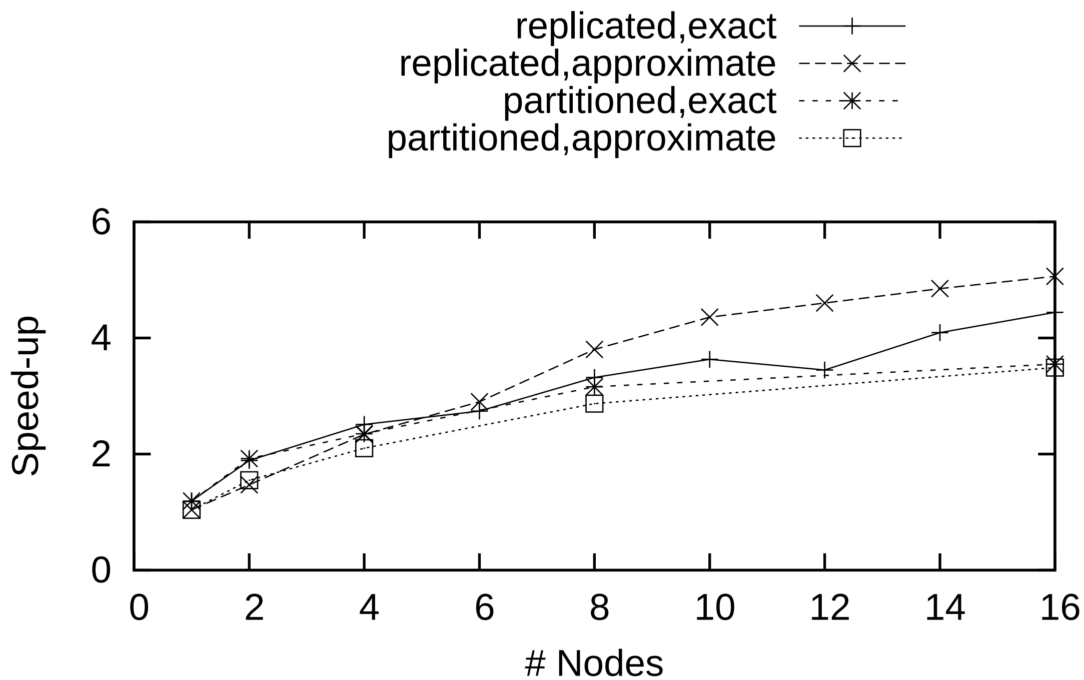
Input: Phylogenetic Tree



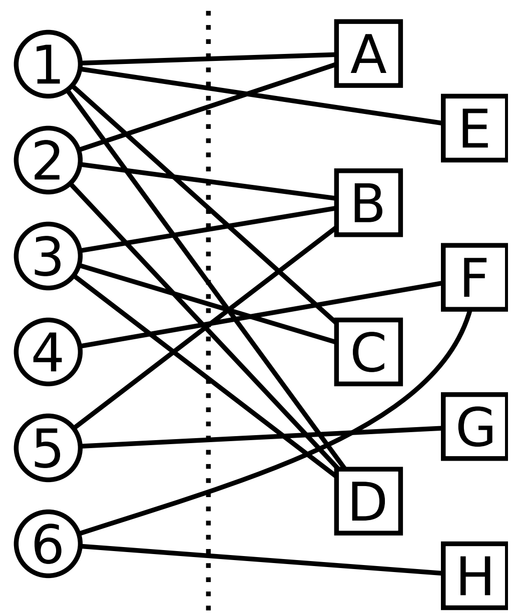
fsnsg

Parallel Mapping of Primers to Species

```
s @opt1:88[0<in.txt,1|p1:0,3|p2:0] $ fanout;
p1@opt1:88[1|pe:0]    $ arb_probe_dup;
p2@opt2:88[1|pe:3]    $ arb_probe_dup;
pe@opt2:88[1>out.txt] $ gather;
```



Intermediate Result: Species and OSS



fsnsg

Agenda

- Research Area
- Easy Distributed Stream Processing with DUP
- Case Study: Fast Primer Search
  - Biological Questions
  - Primer Search Parallelization with DUP
  - **Mapping Primers to Species with the BGRT**
- Other Research Results

Biological Questions

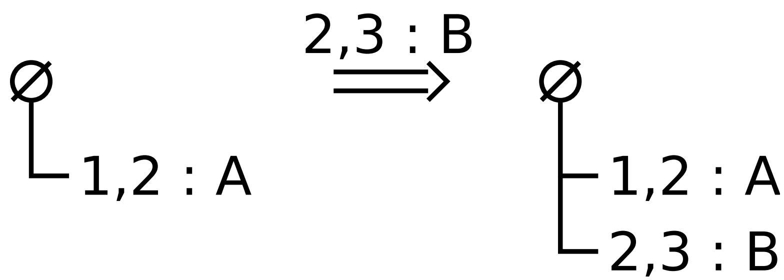
- Which are the most specific OSSs for a species?
- Which are the most specific OSSs for a subtree in the phylogenetic tree?
- Sequence information may contain errors; allow up to k out-group hits!
- Perfect OSS may not exist even with out-group hits; maximize number of in-group hits

BGRT Creation (1/5)

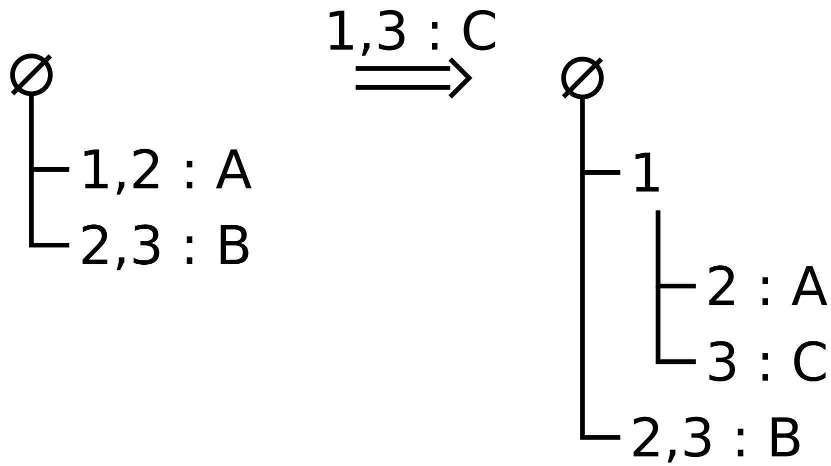
$\emptyset \xRightarrow{1,2\,:\,A} \emptyset$

└ 1,2 : A

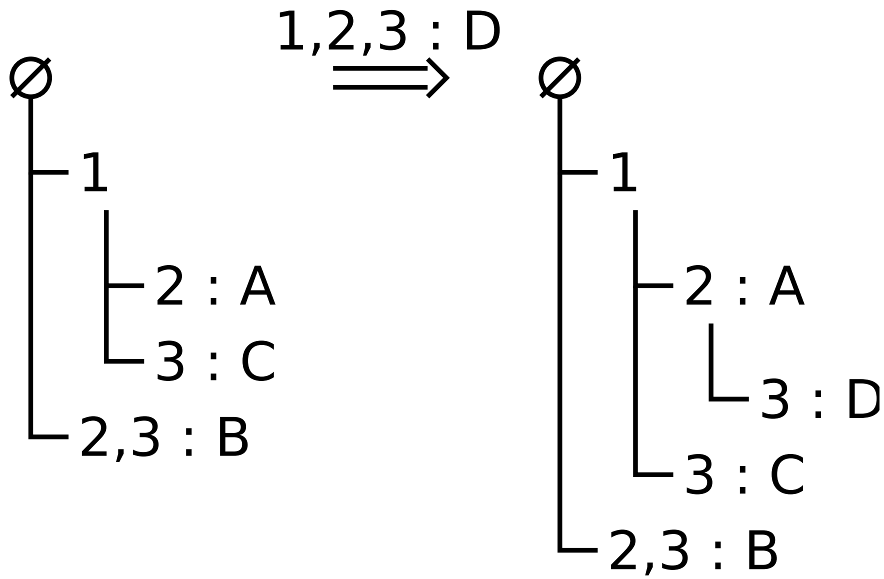
BGRT Creation (2/5)



BGRT Creation (3/5)



BGRT Creation (4/5)

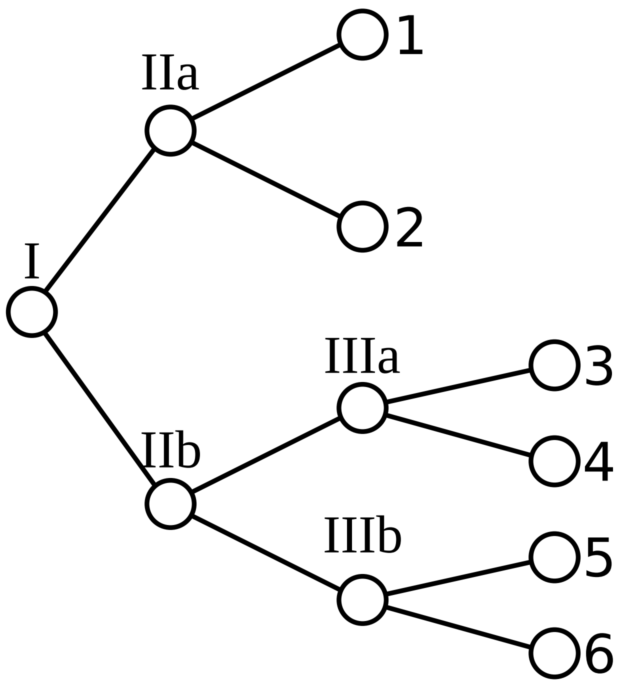


BGRT Creation (5/5)

∅

- 1 : E
  - 2 : A
    - 3 : D
  - 3 : C
- 2,3 : B
  - 5 : G
- 4 : F
  - 6 : H

Iterate-And-Bound



∅

- 1 : E

I	IIa	IIb	IIIa	IIIb
1:0	1:0	0:1	0:1	0:1

  - 2 : A

I	IIa	IIb	IIIa	IIIb
2:0	2:0	0:2	0:2	0:2

    - 3 : D

I	IIa	IIb	IIIa	IIIb
3:0	2:1	1:2	1:2	0:3

  - 3 : C

I	IIa	IIb	IIIa	IIIb
2:0	1:1	1:1	1:1	0:2

- 2,3 : B

I	IIa	IIb	IIIa	IIIb
2:0	1:1	1:1	1:1	0:2

  - 5 : G

I	IIa	IIb	IIIa	IIIb
3:0	1:2	2:1	1:2	1:2

- 4 : F

I	IIa	IIb	IIIa	IIIb
1:0	0:1	1:0	1:0	0:1

  - 6 : H

I	IIa	IIb	IIIa	IIIb
2:0	0:2	2:0	1:1	1:1

Final Result

		Phase										
		I	IIa	1	2	IIb	IIIa	3	4	IIIb	5	6
Mismatches	0	3:D,G	2:A	1:E	%	2:H	1:F	%	1:F	%	%	%
	1	%	2:D^{+}	1:A,C^{+}	1:A,B	2:G^{+}	1:B,C,H^{+}	1:B,C	1:H^{+}	1:H	%	1:H
	2	%	1:G*	1:D^{+}	1:D,G^{+}	1:D*	1:D,G^{+}	1:D^{+}	%	1:G^{+}	1:G	%
	3	%	%	%	%	%	%	%	%	0:D*	%	%

“$+$” Should probably not be computed (mismatches $>$, matches $=$)

“$*$” Even more useless (mismatches $>$, matches $<$)

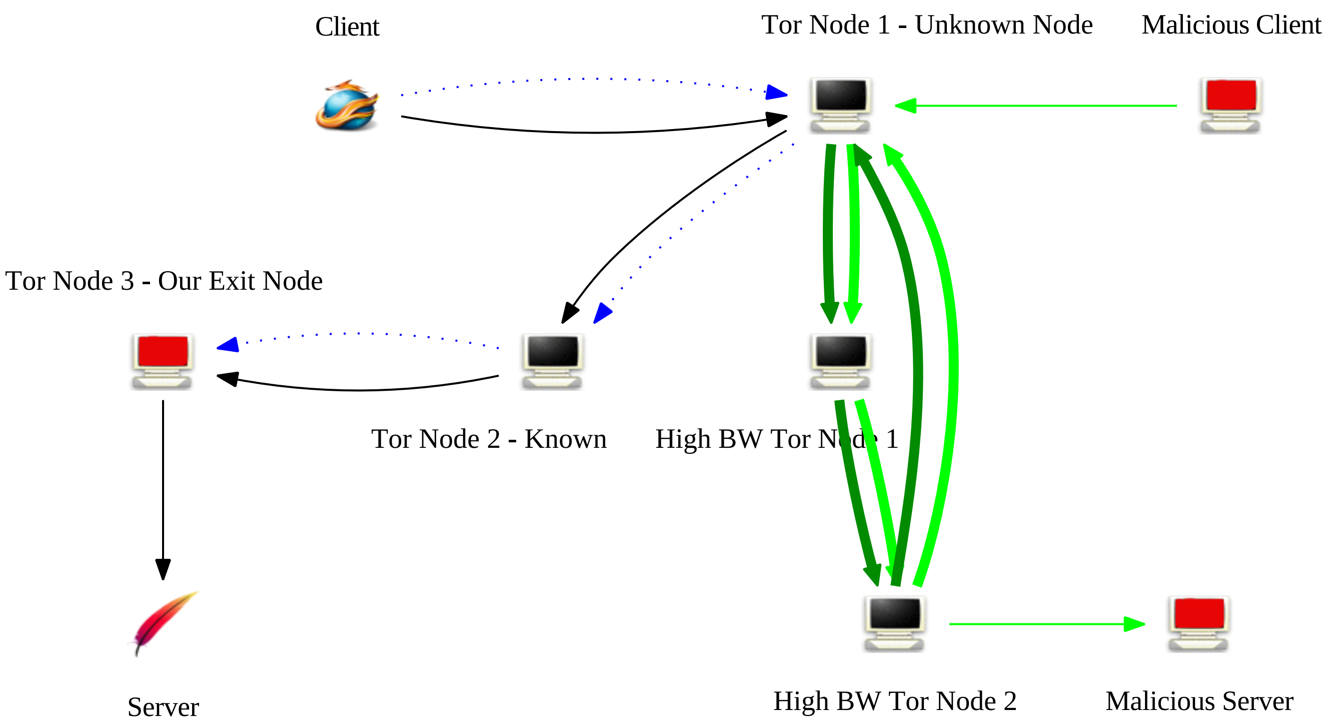
Agenda

- Research Area
- Easy Distributed Stream Processing with DUP
- Case Study: Fast Primer Search
  - Biological Questions
  - Primer Search Parallelization with DUP
  - Mapping Primers to Species with the BGRT
- **Other Research Results**

Other Current Work

- Successful attacks on various "secure" P2P networks (**Tor**, Freenet, Tahoe LAFS, ...)
- Randomized Resilient Routing in Restricted Route Topologies
- Autonomous NAT traversal
- Automatic Restart Management: RAS for GNU/Linux
- Parallelizing Protein Analysis (with Rost Lab)

An Attack on Tor (USENIX Security 2009)



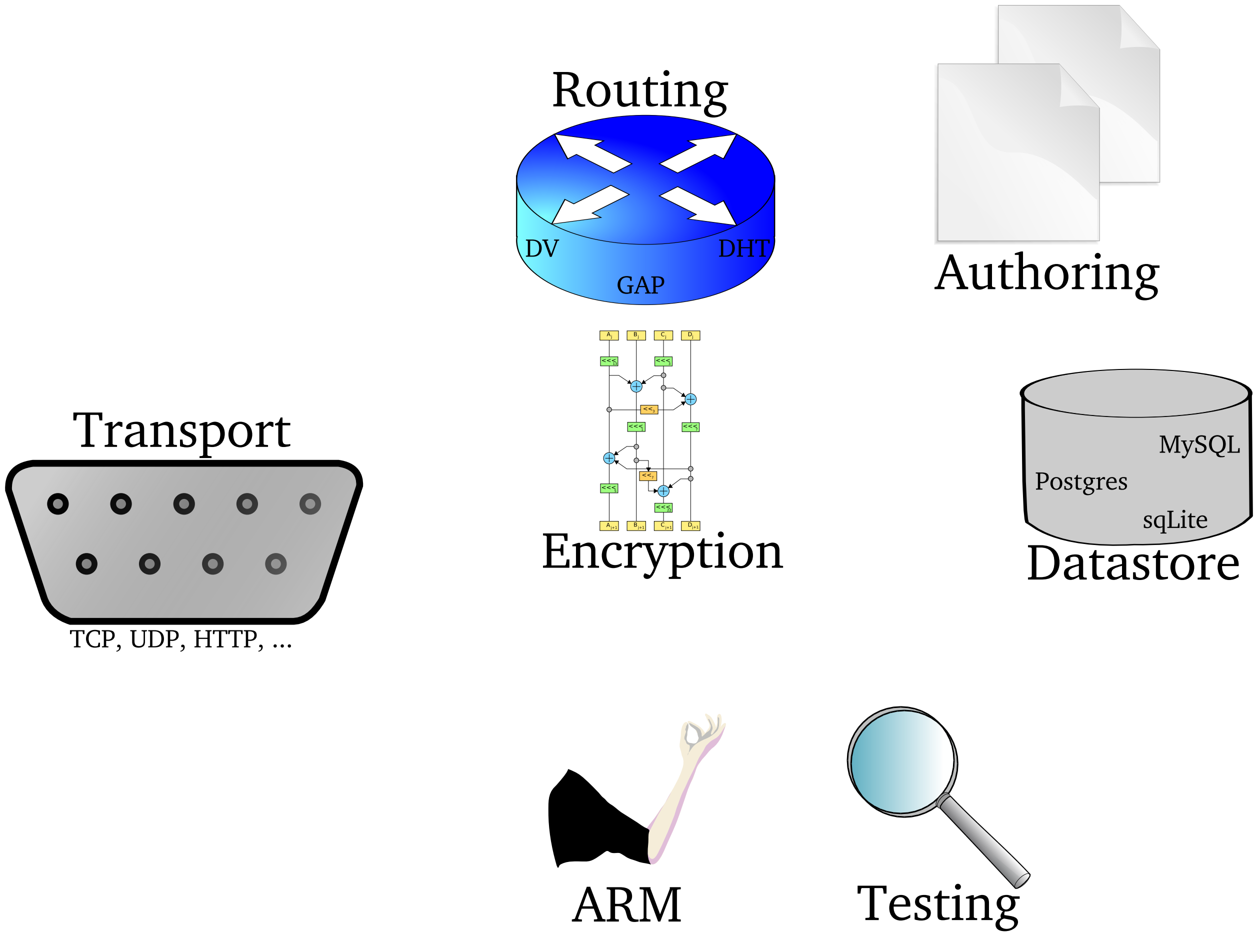
Future Work

- Resource allocation for DUP
- Parallelize more applications with DUP
- Aspect-oriented coordination language for DUP
- Migrating to IPv6 using secure P2P VPN over GNUnet
- Memory fragmentation analysis (Mallice)

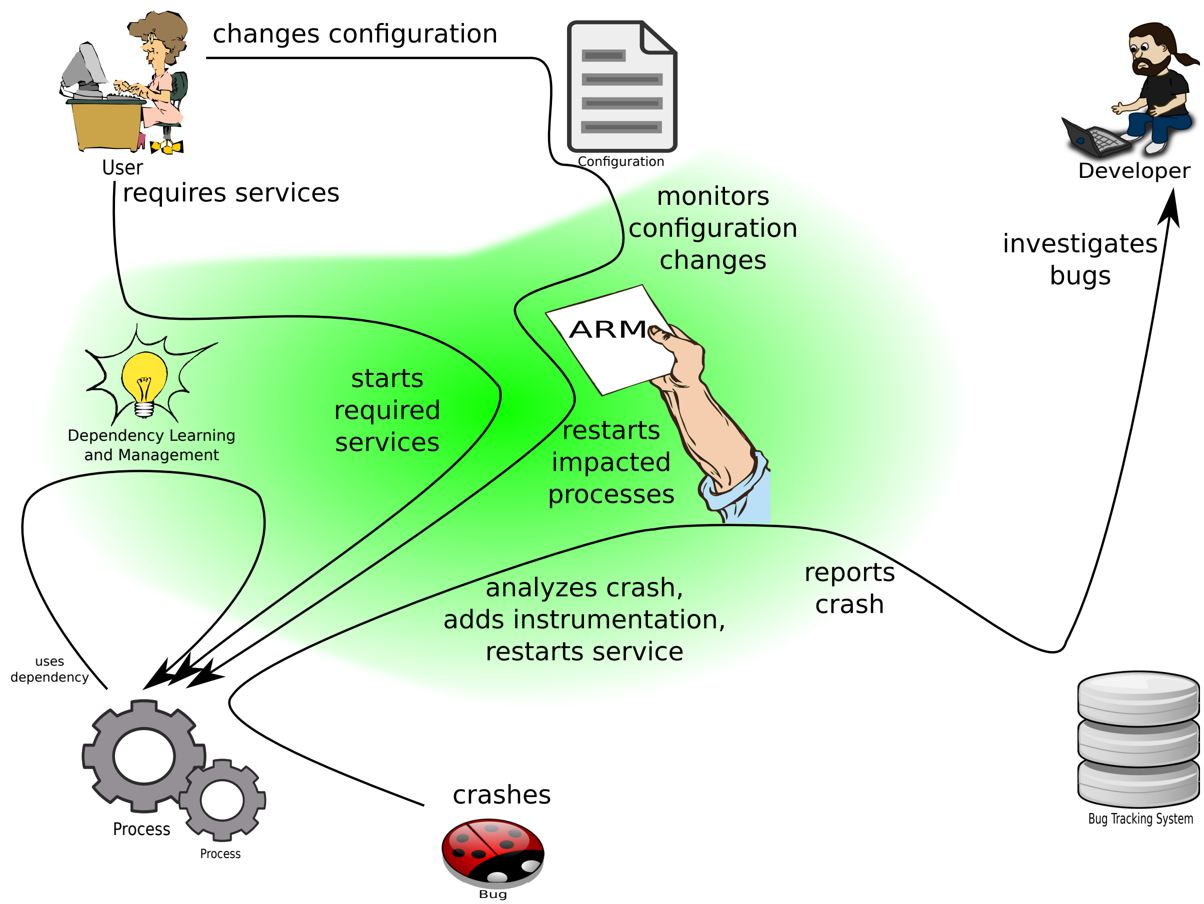
Questions

?

The GNUnet Framework

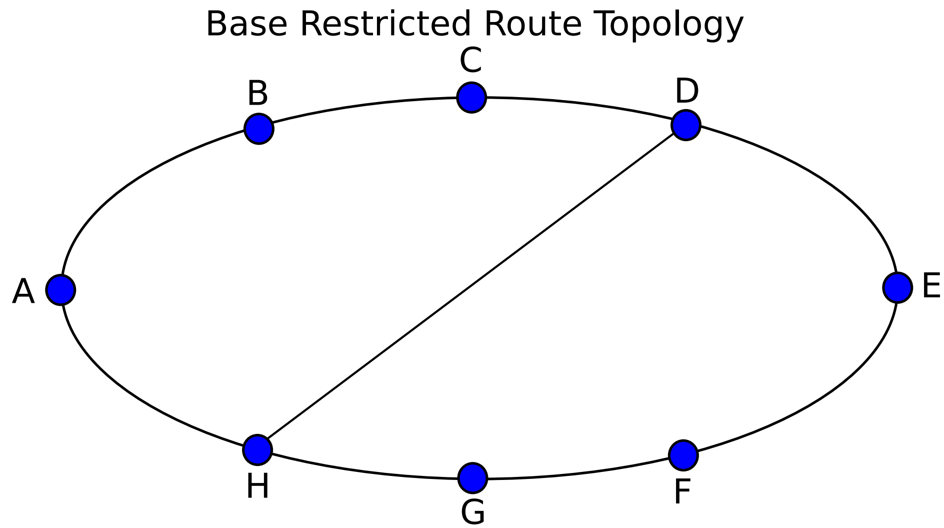


GNUnet's Automatic Restart Manager

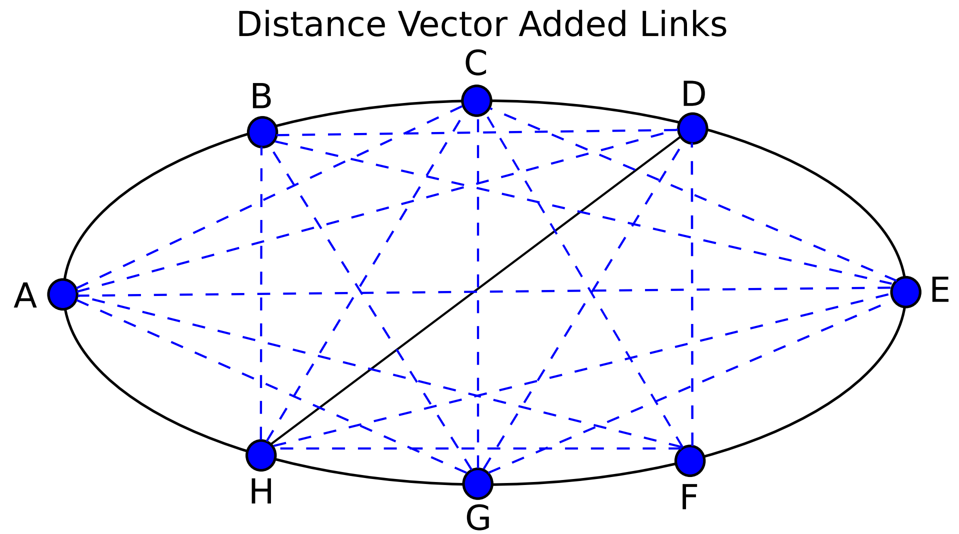


fsnsg

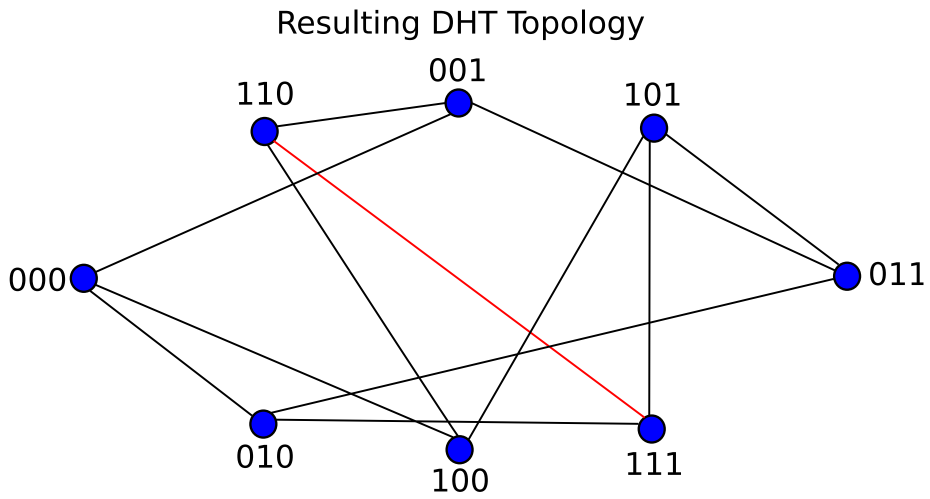
Secure Routing (1/3)



Secure Routing (2/3)



Secure Routing (3/3)



Teaching

- Programming Languages
- Mainframe Administration
- Computer Networking
- Peer-to-Peer Networking

Group Infrastructure

Software:

- Drupal / Mantis
- Doxygen / Subversion
- Buildbot / lcov
- clang (LLVM)
- Coverity Prevent

Hardware:

- Server (i7)
- Sheeva Plug (ARM)
- iMac (PowerPC)
- GNU/Linux VMs (AMD)
- PMP-capable NAT router

Furthermore, we have access to a wide range of networking equipment (via Lehrstuhl) and HPC facilities at the LRZ.