

10 December 2014



# Disruption and the DNS

Paul Mockapetris

[pvm@a21.com](mailto:pvm@a21.com)

[pvm@nominum.com](mailto:pvm@nominum.com)

Paul Mockapetris <[Paul-Vincent.Mockapetris@npa.lip6.fr](mailto:Paul-Vincent.Mockapetris@npa.lip6.fr)>

---

**Thesis: We need more!**

---

# Today's Agenda

- Introduction to DNS. How Does it Grow?
- How did first disrupt?
- Later disruption and invention
- Future Directions

# All Distributed Systems have 3 Parts:

Hardware



Software



Configuration

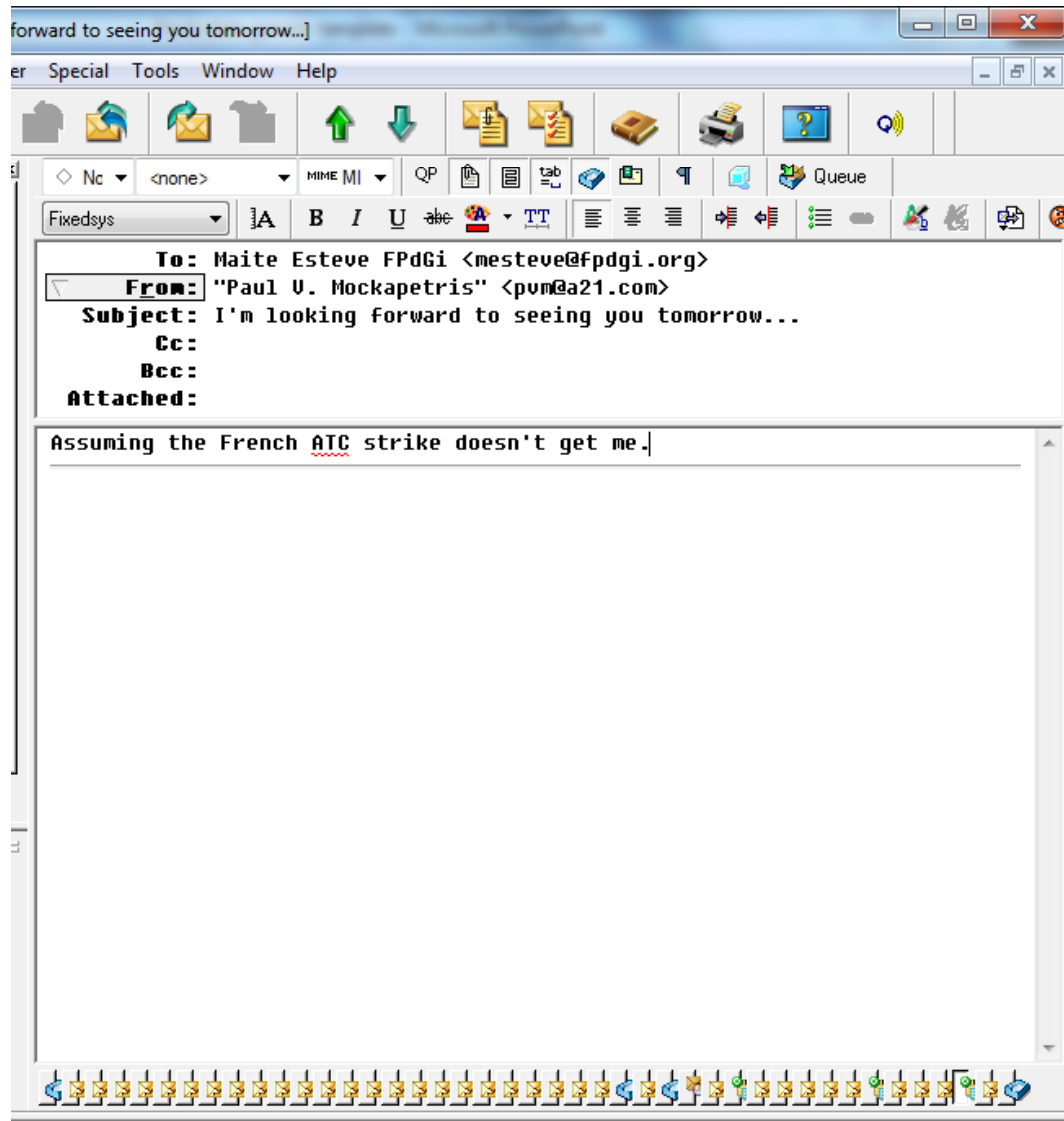


## Why is it always so messy?



- Because we always build systems that challenge:
  - the competition
  - the complexity we can handle

# Did you use the DNS?



- 1983

1 DNS lookup per message

- 2014

30 or so

# Did you use the DNS?



- 1995:  
1 lookup per webpage
- 2014  
100 lookups  
– 1 per piece of page

# DNS, Cui Bono?

- Conventional View

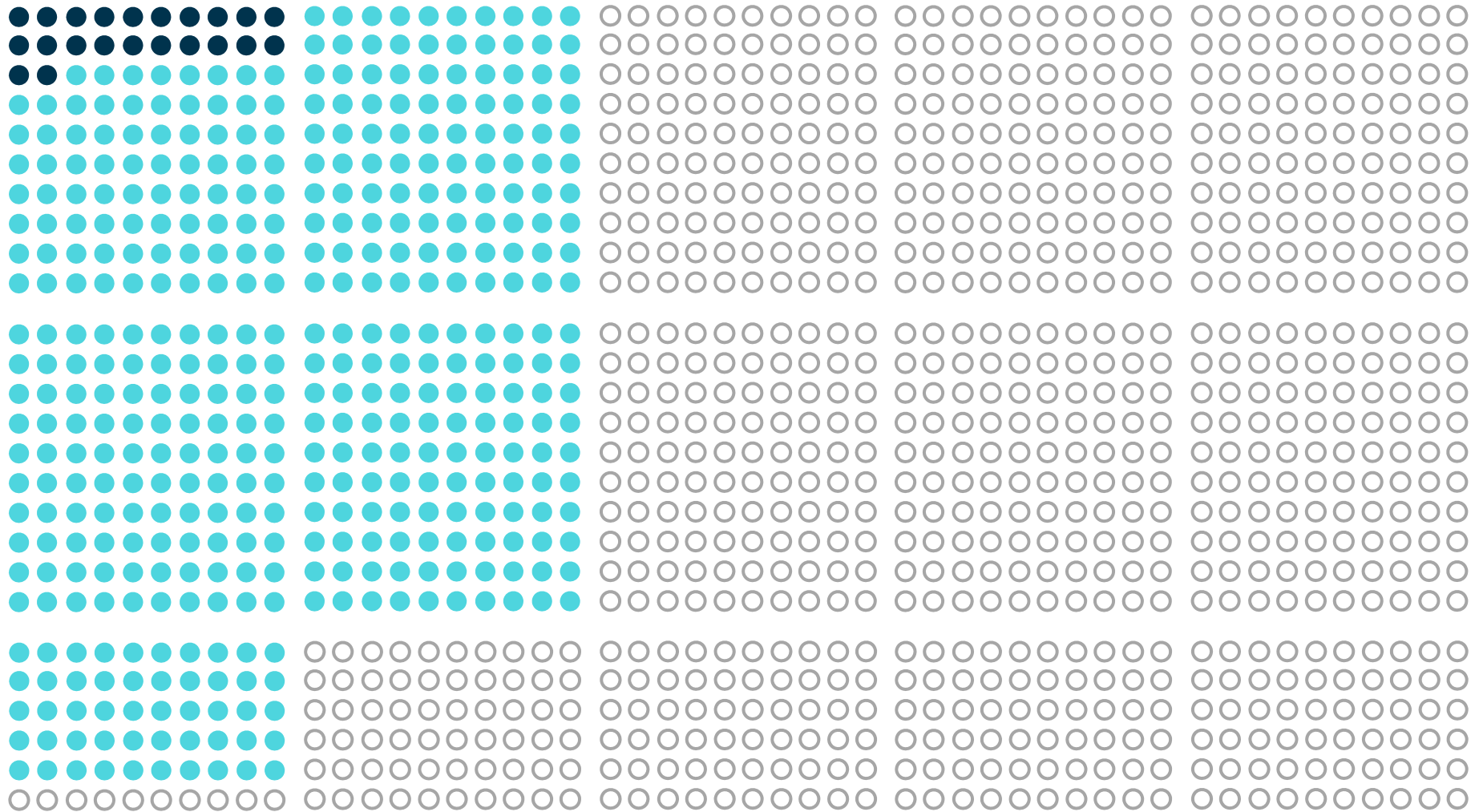
- Replaced host table, which would be too big today if in a single file

- Reality

- Let 100M+ organizations manage their own domains
- Created 10B domain names, 2B+ are public
- 110 pages of original standards lead to thousands of pages of other uses



# GTLD Progress - Halloween 2014



## Key

- Prior to October 2013
- Current New gTLDs
- Potential New gTLDs

**How did we first disrupt?**

---

## My Original Marching Orders from Jon Postel



- Find something better than hosts.txt
- Look at 5 or so proposals, find a compromise
- But very clear that we needed something that scaled differently...

# Can we satisfy both design criteria?

- Elegant



- Modular & Expandable



# Root Server Progress

**JEEVES**  
**ISI Domain Software for TOPS-20**  
**Version 5**

Paul V. Mockapetris

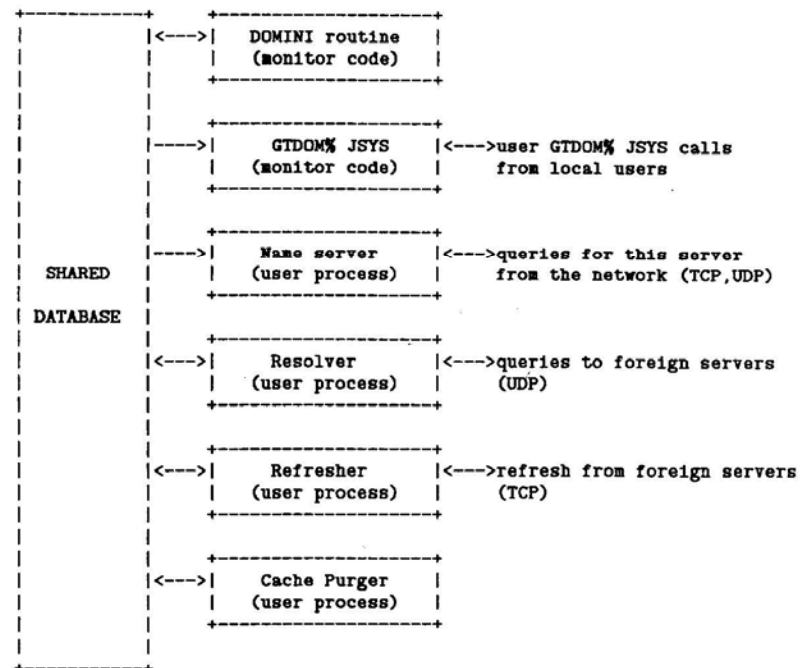
10 October 1986

This memo describes the domain software and its use. The software is currently in use under release 5.1 of TOPS-20, although some related versions are available for other monitors. This memo is a complete rewrite of previous versions. Questions or suggestions for improvement should be directed to [Mockapetris@ISLEDU](mailto:Mockapetris@ISLEDU).

- 1983 Initial specifications done
- 1984 redundant roots operational
- 1985 Symbolics.com
- 1987 Standards done
  
- Parallel progress in resolvers

# Early Implementation Ideas

- Shared Memory Architecture
- No reparse to restart
- Separate upgrades of server functions



- But, zero interest in implementation standardization

# But the fire was lit – DNS RFC family tree



1983

→ Present

# What happened?



**RFC 882/883**

1. Little “DNA” from the original proposals
2. UDP and Server Redundancy recipe is novel
3. RFC 882 & 883 (1983) lead to small changes and 1034 & 1035 (1987)

Thank you ARPA for supporting ISI and UCB and ...

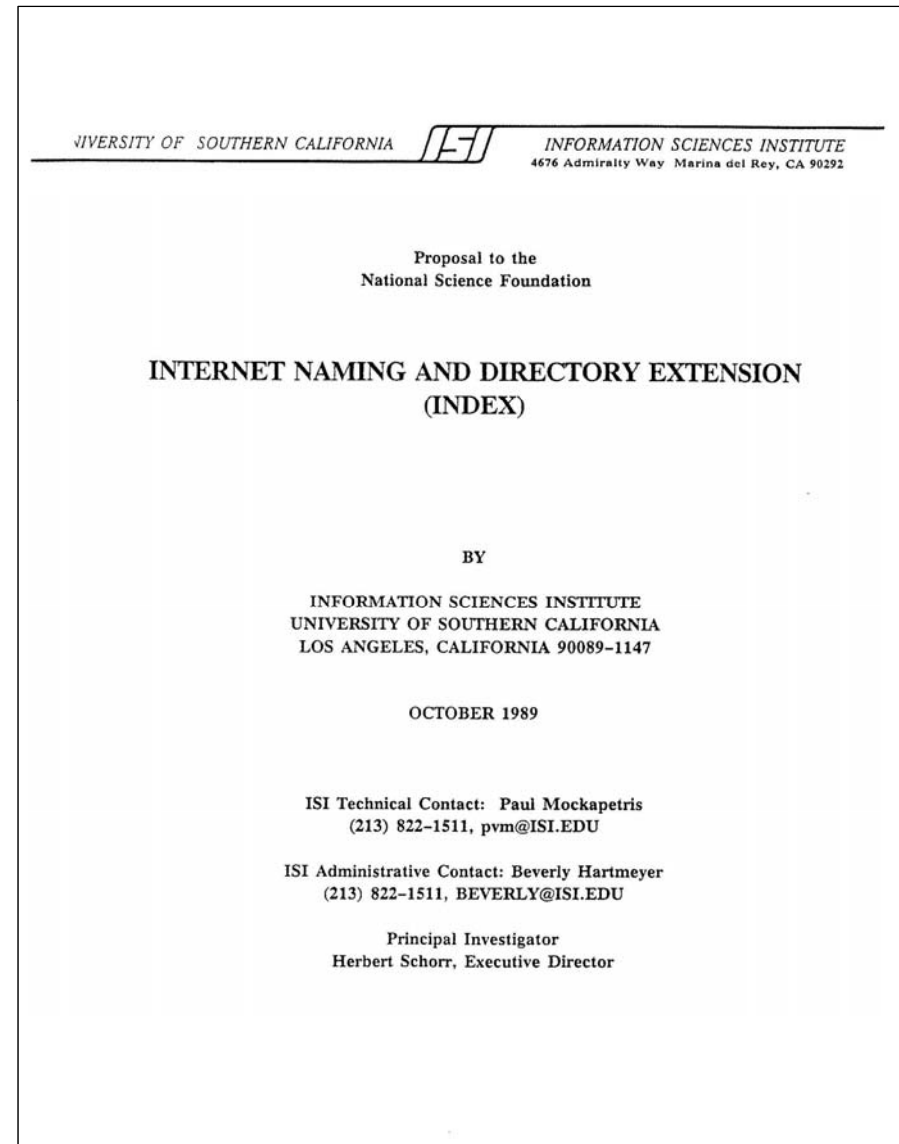


# Later Disruption and Invention

---

# It's 1989 - NSF, Want to improve DNS?

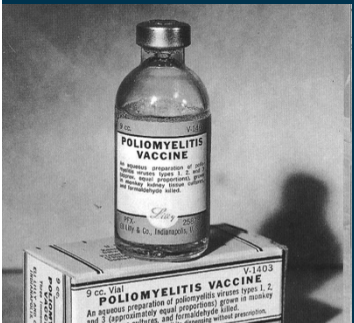
- Propose:
  - Fix bind
  - Address
    - Incremental update
    - Security
    - Crawl and build a DNS index of the Internet
    - Abuse (accidental DDOS)



# NSF feedback

- Reviewer 1: Excellent
- Reviewer 2: Very Good (critical, but not research)
- Reviewer 3: Very Good (please just fix bind)
  
- NSF Result: Can't decide
  
- So much for planned evolution...

# Other inventions and thoughts



# 1. DNS Basic Algorithms

- Initial algorithms were purposely minimal – We can afford more now!
  - Don't just go to the top and then down
- Is there a way to kill backward compatibility?
- Is there a way to get people to integrate authoritative and caching servers?

## Different Rules for Yesterday and Tomorrow

- Datagrams are fast
- Opportunistic Caching
- One key to rule them all
- Datagrams for DDOS
- Privacy of queries and responses
- Multiple trust anchors

## 2. Information Centric Networks

- In some ways a better DNS
- Can we:
  - Merge the best ICN ideas into DNS?
  - Kill off DNS, replace with ICN?
- But ICN has its own set of issues:
  - Replacing infrastructure means a IPv6-like timeline, so just layer and get over it
  - More research on name structures, less on hardware
  - Which ICN?

### 3. Algorithmic Contracts – a personal favorite

- Do away with central management entirely, a la Bitcoin, etc
- Zone management becomes:
  - An accepted set of rules
  - Non-repudiable logs per delegation
  - No jurisdictional locus
  - One or more zone generators
- Extend to other applications
  - Number Portability
  - Contact Sharing
  - ...



# Sample Problems

- Registration
  - Internet TLDs and their management
    - Also addresses, ASNs, ...
  - Portable Phone Numbers
  - “Do Not Call” registries
- Connection
  - Require security: car, airplane, smartphone busses
  - Require privacy: IOT tag call home, bluetooth, WiFi tracking
- Peering?
  - End to end QOS?
  - E2E virtual circuits

# A brief Introduction to the DNS root

- A database of TLD data which is growing to ~2K entries, some TLDs are countries (ccTLD) e.g. .ES, some generic (gTLD) e.g. .COM. Or .ORG
- New varieties created recently e.g. .BANK
- Each TLD configured by a few records (5-10)
- Example records
  - Nameserver and nameserver addresses
  - Digital signatures

# The DNS root (ccTLDs)

- Today:

1. TLD submits change to ICANN / Verisign on even/odd days
2. ICANN vets, Y/N
3. ICANN submits to USG
4. USG vets, Y/N
5. ICANN generates a candidate root zone twice a day, sends to Verisign
6. Verisign vets, Y/N
7. Verisign signs, sends to root operators
8. Root operators distribute

- One possible tomorrow:

1. TLD writes change to its own non-repudiable journal.
2. Other TLDs, ICANN can register requests for reconsideration
3. If TLD doesn't retract, independent zone builder collects from all TLD journals.
4. Sign it somehow (TBD)

Thank You!

