Lessons from history relevant to the future of DNS – principles and examples

Paul V. Mockapetris

PVM@A21.com, PVM@ThreatSTOP.com, PVM@fsi.io,

pvm@coinweb.net pmockapetris@gmail.com

Darwin on Evolution:

“In the long history of humankind (and animal kind, too) those who learned to collaborate and improvise most effectively have prevailed.”

Two Theories:

- **Applies to the Internet as well**
- **Evolution is about prevailing, which may be cruel to sacred cows**

Evolutionary Survival Imperatives

- **Be Adaptable**
- **Be powered by an expanding resource**
- **Create a new desirable resource**

Examples

Moore's Law on density + adjustments to process, voltage, etc. creates more, faster transistors

Intel creates faster processors

Microsoft, Apple et al discover we need more text, justified text, colored displays, music, AI…

(DNS servers run faster, more names)

Internet protocols prevail over OSI stack because of:

Genius of the fathers of the Internet?

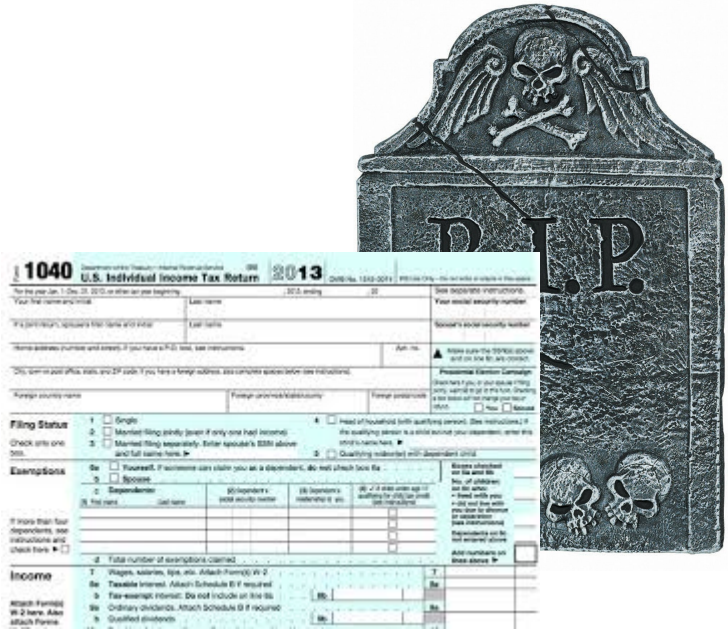
First to harness the explosive growth in bandwidth and processing power

All Distributed Systems have 3 Parts:

Hardware

Software

Configuration (e.g. DNS)



Nobody thinks complexity is good for you

But unlike tobacco, software complexity will always be with us.

Further, a DNS that was too complex for the critics in 1983 wasn't good enough in 1988.

Luckily we learn from experience and develop better tools, so the real issue is how much complexity you can handle.

Where did the Complexity come from?

Because we always build systems that balance:

- **the competition**
- **the complexity we can handle**

Security Economics:
Cost of Defense < Value of Target < Cost of Attack

- Cost of Defense < Value of Target - spending $100 to protect $5 will bankrupt you. This is also how you explain to your customers why you haven't put a bank vault door on a chicken coop.

- Value of Target < Cost of Attack - if your data is worth $1000 and it costs the other guy $999 to get it, then the other guy makes a buck on every attack.

- Cost of Defense < Cost of Attack - the arms race clause. If your spending $1000 and your attacker is spending $500, pretty soon you can't afford to play the game any more.

Maintenance Economics:
Effort of putting data in DNS<Value I get for putting the data in

- The “Obviously required” Internet directory that was endorsed by the Internet research community for decades died in various incarnations.

- Facebook discovered people would type all day to get dates, listen to themselves expound, etc, etc

- For the DNS, MX gets maintained so you get mail, WKS died since the maintainer saw no benefit

Shifting sands (100 queries/sec max?)

Development of the Domain Name System*

Paul V. Mockapetris
USC Information Sciences Institute, Marina del Rey, California

Kevin J. Dunlap
Digital Equipment Corp., DECwest Engineering, Washington

Abstract

The Domain Name System (DNS) provides name service for the DARPA Internet. It is one of the largest name services in operation today, serves a highly diverse community of hosts, users, and networks, and uses a unique combination of hierarchies, caching, and datagram access.

This paper examines the ideas behind the initial design of the DNS in 1983, discusses the evolution of these ideas into the current implementations and usages, notes conspicuous surprises, successes and shortcomings, and attempts to predict its future evolution.

1. Introduction

The genesis of the DNS was the observation, circa 1982, that the HOSTS.TXT system for publishing the mapping between host names and addresses was encountering or headed for problems. HOSTS.TXT is the name of a simple text file, which is centrally maintained on a host at the SRI Network Information Center (SRI-NIC) and distributed to all hosts in the Internet via direct and indirect file transfers.

* This research was supported by the Defense Advanced Research Projects Agency under contract MDA903-87-C-0719. Views and conclusions contained in this report are the authors' and should not be interpreted as representing the official opinion or policy of DARPA, the U.S. government, or any person or agency connected with them.

The problems were that the file, and hence the costs of its distribution, were becoming too large, and that the centralized control of updating did not fit the trend toward more distributed management of the Internet.

Simple growth was one cause of these problems; another was the evolution of the community using HOSTS.TXT from the NCP-based original ARPANET to the IP/TCP-based Internet. The research ARPANET's role had changed from being a single network connecting large timesharing systems to being one of the several long-haul backbone networks linking local networks which were in turn populated with workstations. The number of hosts changed from the number of timesharing systems (roughly organizations) to the number of workstations (roughly users). This increase was directly reflected in the size of HOSTS.TXT, the rate of change in HOSTS.TXT, and the number of transfers of the file, leading to a much larger than linear increase in total resource use for distributing the file. Since organizations were being forced into management of local network addresses, gateways, etc., by the technology anyway, it was quite logical to want to partition the database and allow local control of local name and address spaces. A distributed naming system seemed in order.

Existing distributed naming systems included the DARPA Internet's IEN116 [IEN 116] and the XEROX Grapevine [Birrell 82] and Clearinghouse systems [Oppen 83]. The IEN116 services seemed excessively limited and host specific, and IEN116 did not provide much benefit to justify the costs of renovation. The XEROX system was then, and may still be, the most sophisticated name service in existence, but it was not clear that its heavy use of replication, light use of caching, and fixed number of hierarchy levels were appropriate for the heterogene-

At the SRI root server 2/12/87

Up for 3 days

114 queries/minute

48% A query type

74% normal responses

```
Name server statistics

Name server requests:433e79
Traps: 232a30 (0.5) Faults: 101385 (0.2) Pager time: 04:29.0 (0.6)
Runtime: 2:06:49.1 (17.6) Elapsed time: 62:09:09.7 (524.2)

114.3 queries/min

Query distribution by QUERY type

QTYPE    Count %All

A        207467  47.9.................................................
NS       10345   2.4...
MF         726   0.2
CNAME    13910   3.2...
SOA         56   0.0
MG           7   0.0
MR           7   0.0
WKS          8   0.0
PTR      38056   7.8.........
HINFO       25   0.0
MINFO        9   0.0
MX       39526   9.1.........
  252       27   0.0
NAILA        3   0.0
*        120307  27.7.............................
  255        5   0.0

TOTAL    432715

Query distribution by QUERY class

QCLASS    Count  %All

IN       312839  72.5.....................................................................
    3      2553   0.6.
  155         1   0.0
*        116214  26.8............................
  256         2   0.0

TOTAL    432715

Query distribution by response code

RCODE    Count %All

normal   318542  73.5.....................................................................
```

Can we make DNS a more powerful database?

Then

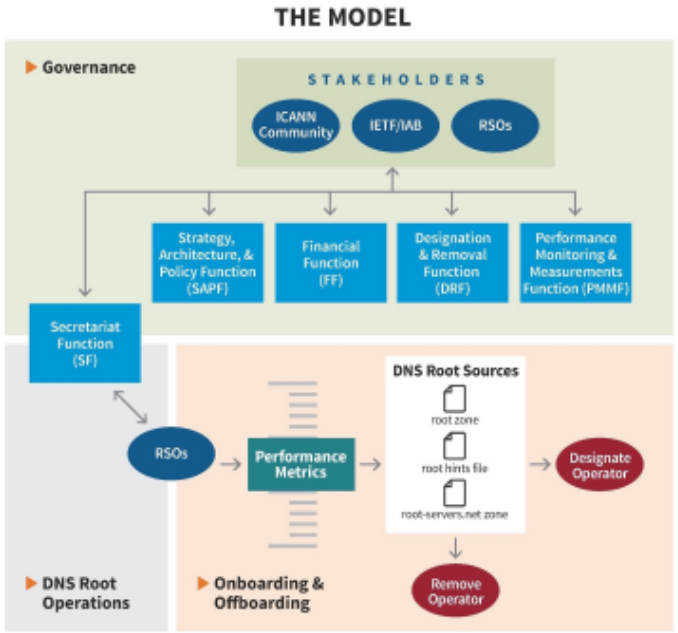
- **Separate Concepts and Implementation**
- **Concepts**
  - Tree structure
    - Delegation of ownership/control
    - Navigate top down guaranteed
    - Opportunistic caching
  - Sets of primitive data records
  - Two types of data transfer
    - Queries for RR sets
    - Transfers for zones

Now?

- **Separate Concepts, Implementation, and Theories/Ideas**
- **Name space as lattice**
- **Bidirectional navigation**
- **Bidirectional, Mutirooted security? (What did we learn from X.509 certificates?)**
- **Use signed zones for security**

The root servers

A Proposed Governance Model for the DNS Root Server System



We appreciate the efforts of the RSOs

- But its time to create a new redundant infrastructure to deliver signed copies of the root zone (including glue)
- One signature for the whole zone!
- Have all resolving servers check their authoritative zones first
- Do the same for all zones of the organization
- No hard target for DDOS anymore

Works in a lot of cases, not all, just like cloud

END-TO-END ARGUMENTS IN SYSTEM DESIGN

J.H. Saltzer, D.P. Reed and D.D. Clark*

M.I.T. Laboratory for Computer Science

This paper presents a design principle that helps guide placement of functions among the modules of a distributed computer system. The principle, called the end-to-end argument, suggests that functions placed at low levels of a system may be redundant or of little value when compared with the cost of providing them at that low level. Examples discussed in the paper include bit error recovery, security using encryption, duplicate message suppression, recovery from system crashes, and delivery acknowledgement. Low level mechanisms to support these functions are justified only as performance enhancements.

Introduction

Choosing the proper boundaries between functions is perhaps the primary activity of the computer system designer. Design principles that provide guidance in this choice of function placement are among the most important tools of a system designer. This paper discusses one class of function placement argument that has been used for many years with neither explicit recognition nor much conviction. However, the emergence of the data communication network as a computer system component has sharpened this line of function placement argument by making more apparent the situations in which and reasons why it applies. This paper articulates the argument explicitly, so as to examine its nature and to see how general it really is. The argument appeals to application requirements, and provides a rationale for moving function upward in a layered system, closer to the application that uses the function. We begin by considering the communication network version of the argument.

In a system that includes communications, one usually draws a modular boundary around the communication subsystem and defines a firm interface between it and the rest of the system. When doing so, it becomes apparent that there is a list of functions each of which might be implemented in any of several ways: by the communication subsystem, by its client, as a joint

* Authors' addresses: J.H. Saltzer and D.D. Clark, M.I.T. Laboratory for Computer Science, 545 Technology Square, Cambridge, Massachusetts 02139.: D.P. Reed, Software Arts, Inc., 27 Mica Lane, Wellesley, Massachusetts 02181.

This research was supported in part by the Advanced Research Projects Agency of the U.S. Department of Defense and monitored by the Office of Naval Research under contract number N00014-75-C-0661.

Revised version of a paper from the Second International Conference on Distributed Computing Systems, Paris, France, April 8-10, 1981, pp. 509-512.: Copyright 1981 by The Institute of Electrical and Electronics Engineers, Inc. Reprinted with permission.

Published in ACM Transactions in Computer Systems 2, 4, November, 1984, pages 277-288.

Reprinted in Craig Partridge, editor Innovations in internetworking. Artech House, Norwood, MA, 1988, pages 195-206. ISBN 0-89006-337-0. Also scheduled to be reprinted in Amit Bhargava, editor. Integrated broadband networks. Artech House, Boston, 1991. ISBN 0-89006-483-0.

Scribe/FinalWord source: http://web.mit.edu/Saltzer/www/publications/

In Today's DNS, can I outsource AND keep control

Increasing outsourcing:

- **Authoritative service**
- **Authoritative tailoring (GSLB, etc)**

- **Recursive service (X.X.X.X)**
- **Recursive filtering (DNS firewall)**

Questions:

Is the increasing centralization a good idea?

Is it good to let X.X.X.X look at my data, or set my filtering policy?

Where do I get the best filtering threat intelligence?

OH NO – Internet Censorship

OH NO! Internet Censorship!!!!!

Harkens back to SOPA and PIPA debates where DNS policy enforces copyright

My bottom line:

It's effective (like antispam, which everyone seems to accept)

It's OK so long as the user controls policies

We'll argue when policy is set by government, ISP, parents …

While the user controls policies, the user's ISP may be doing the work

Key issue: diversified structure of industry, i.e. user choice, including DNS provider

Internet history

“The rapid growth of the network made it impossible to maintain a centrally organized hostname registry and in 1983 the Domain Name System was introduced on the ARPANET and published by the Internet Engineering Task Force as RFC 882 and RFC 883.” - Wikipedia

“The first IETF meeting was attended by 21 U.S.-government-funded researchers on 16 January 1986.” - Wikipedia

“January 1, 1983, was an official 'flag day' for the ARPANET, which became what we know as the Internet.” – Internet Society

Raw materials for the DNS design

Candidates:

IEN116

Xerox Naming System

NSF name server

X.500 –The anointed choice

PVM Background:

IBM Cambridge Scientific Center

- Virtual machine technology ~1966

MIT Architecture Machine (now Media Lab)

- MAGIC distributed computing system ~1969
  - Multiple minicomputers acting as one system
  - ~~two level hierarchies are enough~~

Charles Stark Draper Labs

- Highly reliable systems for space ~1971

UC Irvine Distributed Computer System

- Networking by name ~1973

Good Artists Copy; Great Artists Steal

We should think about stealing from several emerging technologies:

Blockchain

Database

…

Some predictions

Intermediate term predictions

The research community has dozens of projects, such as:

- **Named Data Networking**
- **Information Centric Networking**
- **Mobility First, etc, etc**

Common Theme

- **Named, digitally signed objects accessed by name not address**

Historians might claim that looks like X.509

- **Challenge is reducing the cost of doing this for every piece of data**
- **It's a problem of tailoring, simplifying and cost reducing security**
- **e.g. BII Yeti work, et al**

Long term prediction

As an undergraduate at MIT, I learned a saying:

“Data is just the stupidest form of program”

In the ultimate, the DNS should hold programs as well as data.

Questions?

Perhaps of interest:

https://www.icann.org/resources/pages/video-mockapetris-2013-08-15-en

Regarding the death of the internet by new TLDs, name collisions, and why you shouldn't trust all the experts

https://www.icann.org/en/system/files/files/iti-report-15may14-en.pdf

ICANN 2014 report of a study by experts (including me and several others here in Montreal) on the future of DNS