



Matt Peterson

Burning Man

*Scaling an extreme
temporary event
network*

Matt Peterson

matt@burningman.com

whois



AS33612



AS12276 | 63055



Square

AS15211

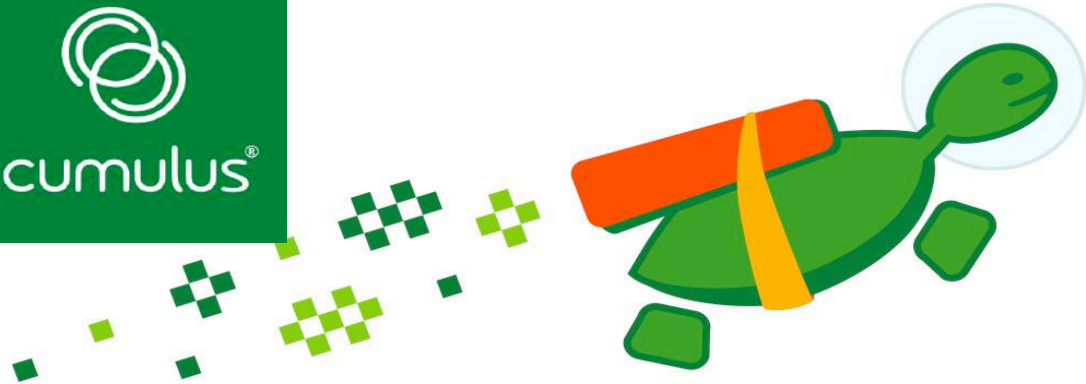


AS36472



AS19230 (56 - 59)

Day Job



Black Rock City, NV



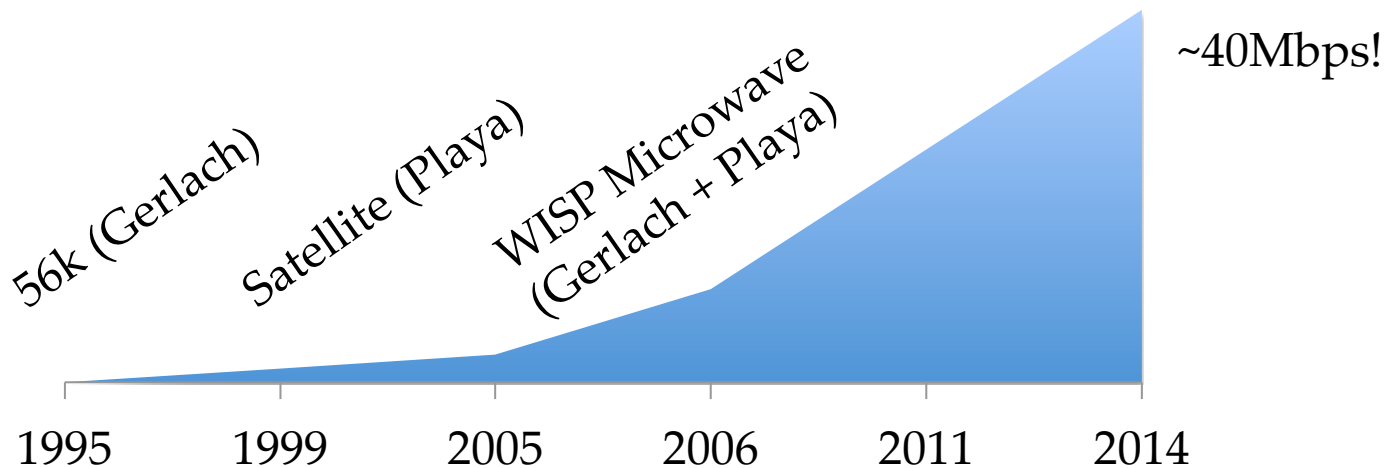
2014 E

matt@burningman.com 2014-10-02 18:10



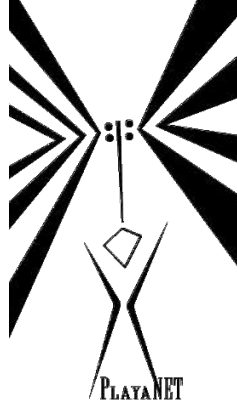
History

- *Dusty bits for a long time*
 - 1995: dial-up modem at Bruno's, not on playa
 - 90s/2000s: a little of everything (T1, satellites)
 - Today: 40Mbps via local WISP



Early Applications

- *BMorg*
 - Weather alerts
 - Newspaper upload



- *Participants*
 - “PlayaNET”: directory of camps, events, & attendees, digital mapping, IRC terminals
 - Video streaming
 - “RC” style control for art projects
 - Early “social media” blogs, ~/ .plan’s, etc

The In-Between

- 2003 – 2008
 - Matt didn't go to Burning Man!
- *Meanwhile, the network...*
 - Kept growing
 - More customers + more bandwidth
 - Became “mission critical”

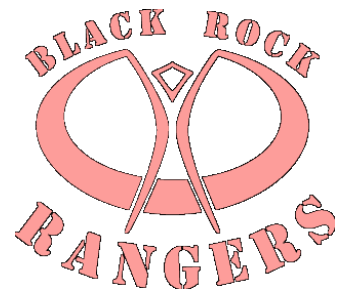


Today's Applications

- *Emergency Services Dept. / hospital*
 - Remote X-ray screening, life flight, PSAP, two-way radio system
- *DPW, Heavy Machinery, Power*
 - Vendor logistics (spare parts, more food, etc.)

Today's Applications

- *Rangers, Box Office*
 - shift scheduling
- *BMIR (public safety FM station), Webcast*
 - streaming
- *DMV, Commissary, Gate/Perimeter*
 - (many many) databases, ticketing







The I.T.S. Team

- *City wide technical services*
 - Network, apps, user support
 - Diverse background
 - RFC authors .. My first 1st crimp
- *Every team member does Layer 1*
 - Rare to sit behind a PC
 - Trenching, crimping, equipment installs, etc.
- *Partner with HEaT & Power departments*
 - Tower deployment, heavy equipment install, power circuits, trenching



Where We Started

- *The network worked, but it wasn't easy*
 - Large L2 bridged architecture, minimal L3 segmentation, multiple NAT layers
 - Two distinct “business units”
 - Departments & Participants
 - Manual configuration, “tribal knowledge”
 - Limited monitoring visibility
 - Numerous single points of failure
 - Staff burnout, turn over

Where We Went

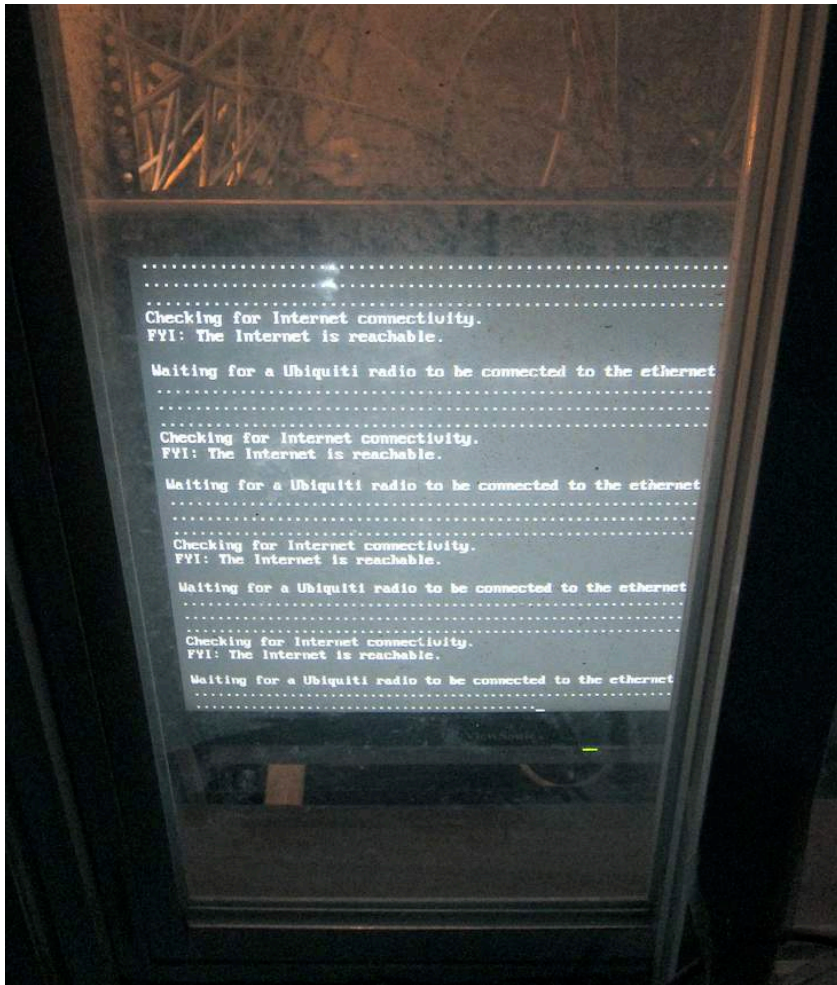
- *Needed to operate as a unified team*
 - Consistent support experience, improved RF spectral efficiency, coordinated IP allocations
- *Standardized COTS equipment*
 - “CCIE off the street” factor, escalation path
- *Standardized service offerings*
 - Org department handoff’s always wired gigE; as aggregated “islands” or single demarc
 - Participant camps supply very prescriptive equipment, “self-install” provisioning

Where We Went

- *Route, always*
 - No L2 segments past a single device
 - OSPF everywhere, core backbone & “islands”
 - Segment where possible, even over WiFi
- *Automation*
 - Initially covering all routers & switches
 - Target goal to cover any device with a config or supplemental service (DNS, monitoring)

Where We Went

- *Model participant camps as self-serve*
 - Only support specific MIMO 5Ghz models
 - Units factory ship as STA, looking “Ubnt” AP...
 - Setup a “Ubnt” on a 360° omni at the NOC (in addition to the 4x 90° sectors)
 - Detect association, upgrade firmware, push parking config
 - Captive portal to capture email, physical address; associate to MAC address
 - Push pre-allocated config after portal

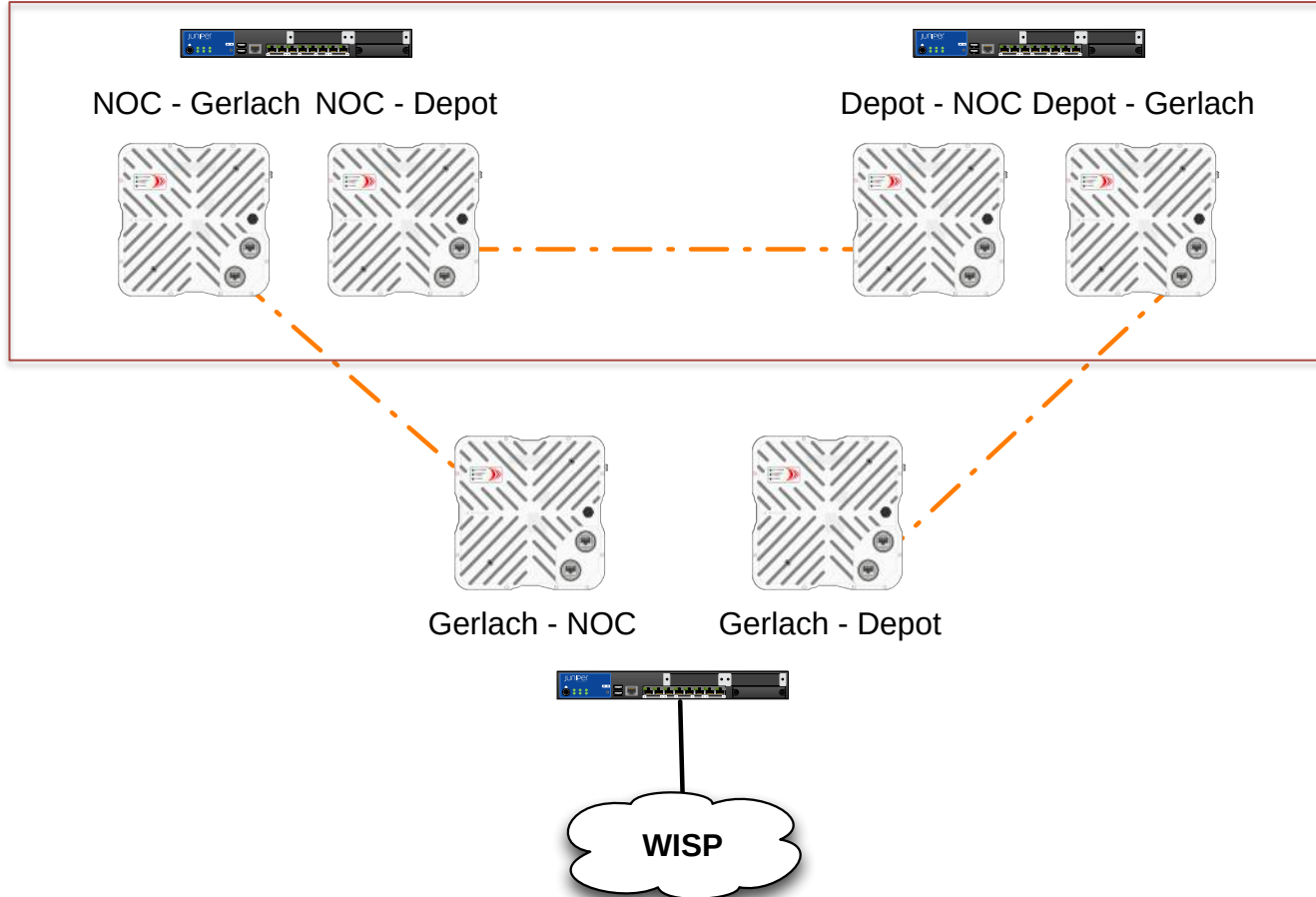


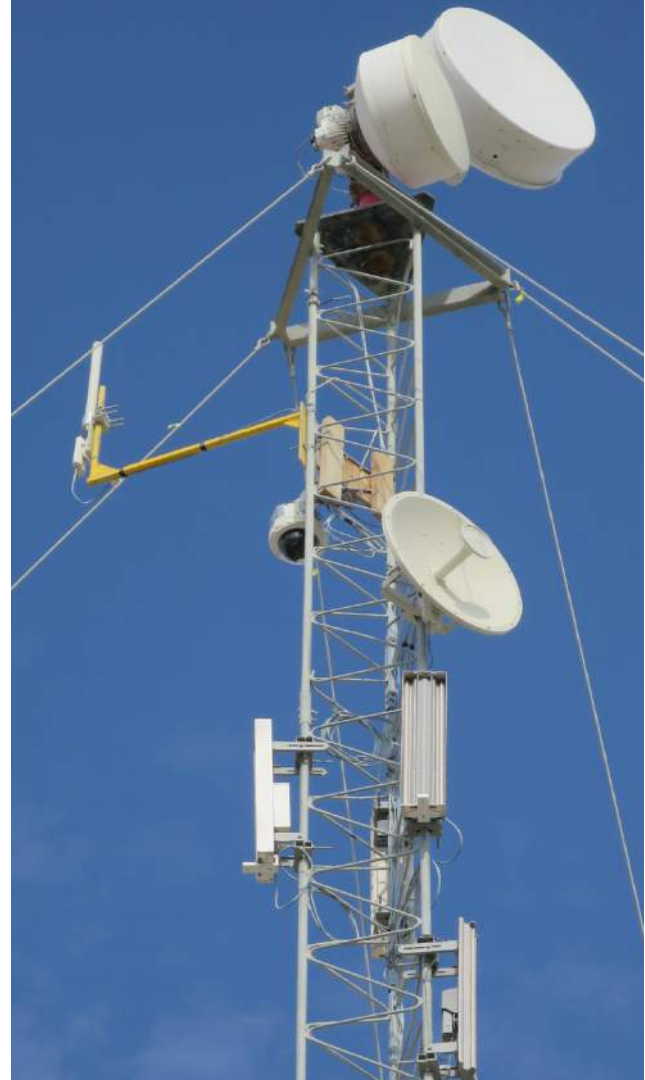
Equipment

- *Backbone (400Mbps; ~10 mile links)*
 - DragonWave 11Ghz PtP “town to playa”
 - Gerlach ➔ DPW Depot (pre & post staging area)
 - Gerlach ➔ NOC (center camp)
 - DragonWave 18Ghz PtP “playa to playa”
 - DPW Depot ➔ NOC
- *Distribution & access layers (~100Mbps, <1mi)*
 - Ubiquiti airMAX 5Ghz 802.11 TDMA'ish
- *Routing & switching*
 - Juniper EX2200-C & SRX220

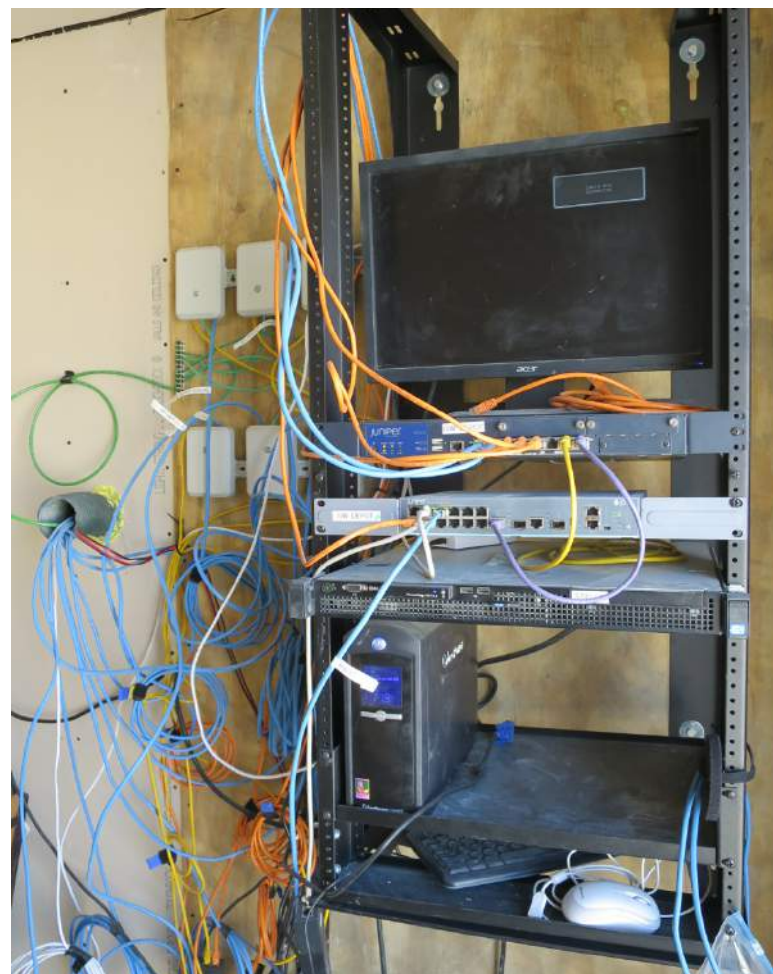
Backbone

Playa

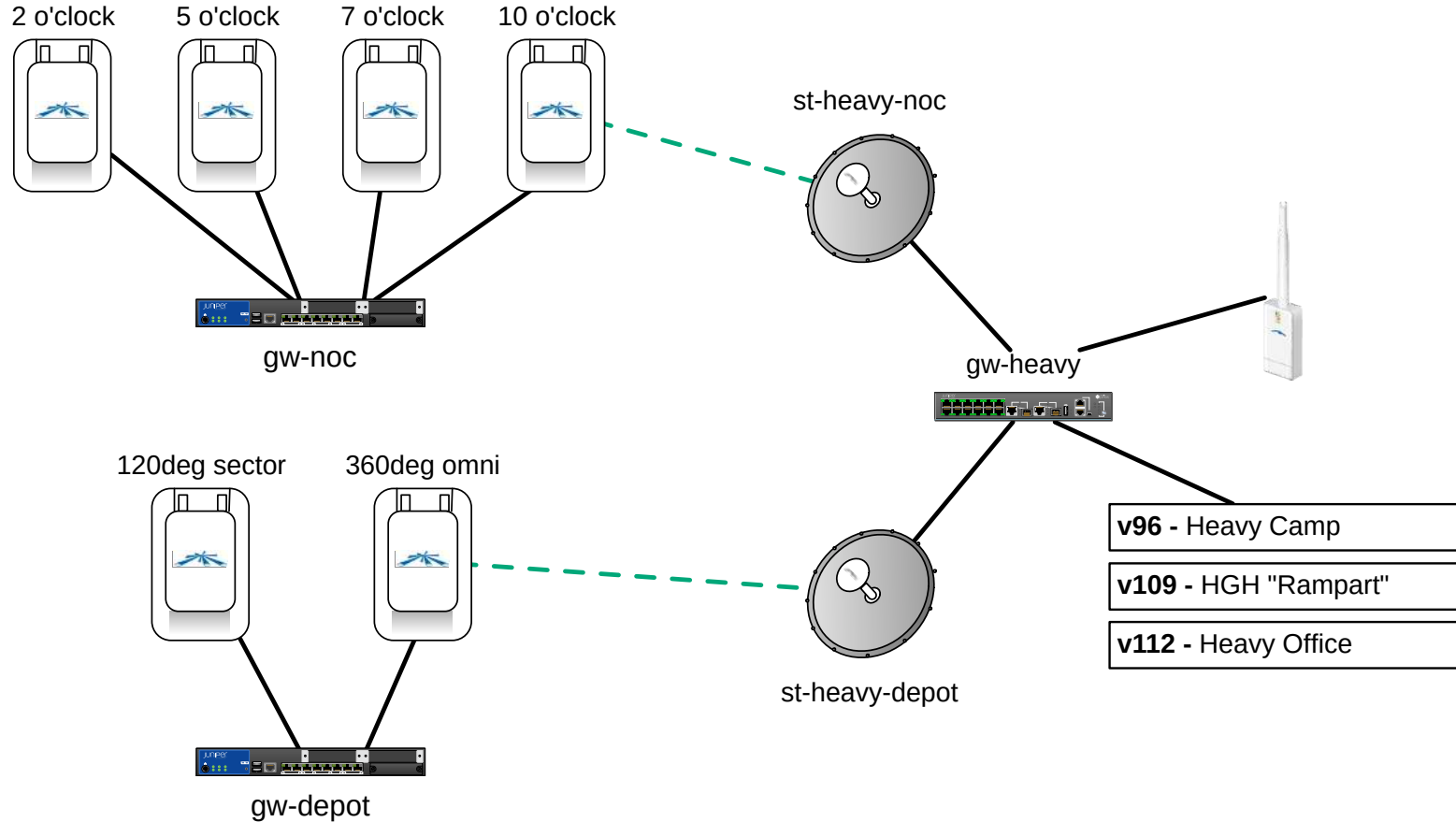


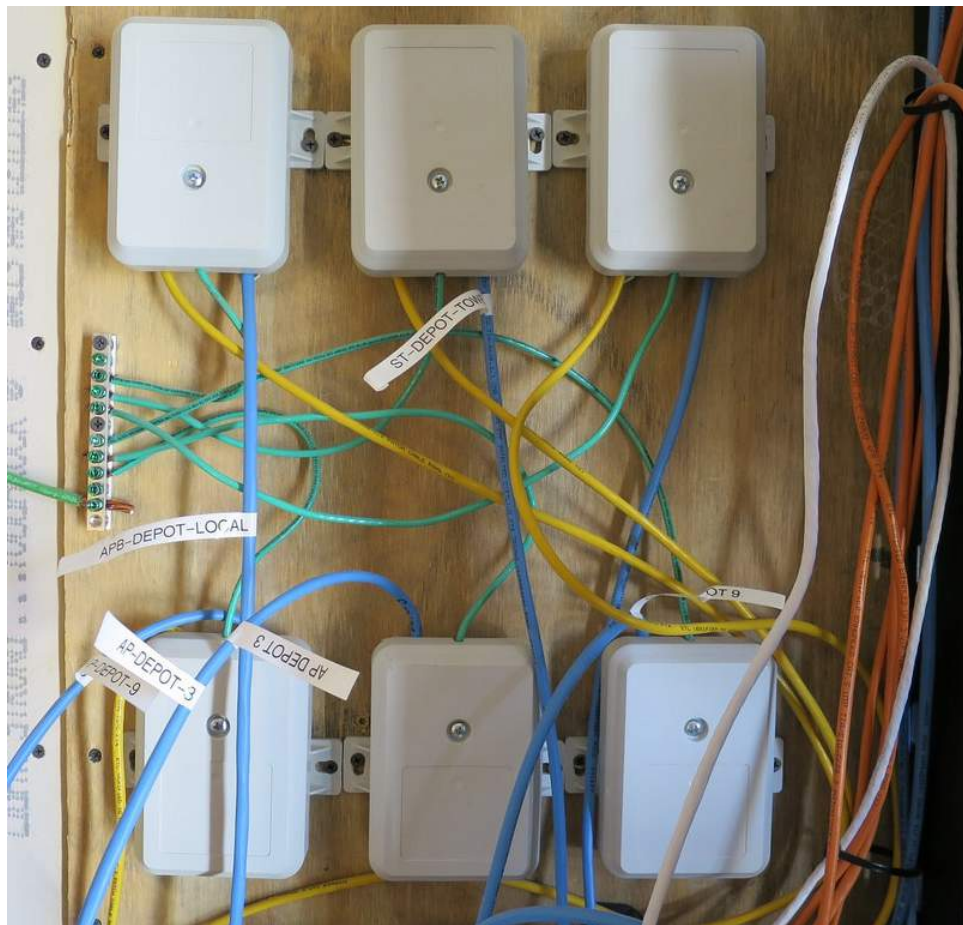


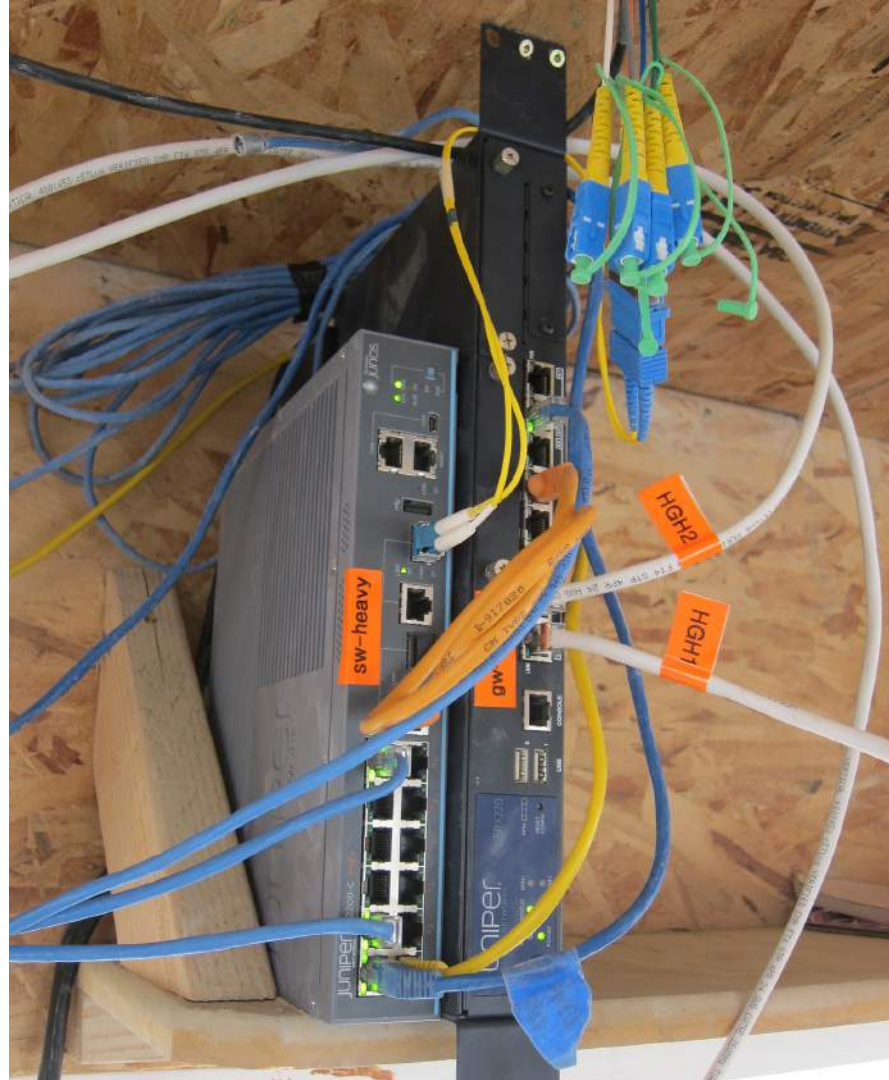




Access







Automation!

- *Held bakeoff (mid 2011 evaluation)*
 - Homegrown YAML → config templates
 - Prototyped NCG (see NANOG49 Tutorial “Automating Network Configuration”)
- *NCG won*
 - Open source, vendor agnostic
 - Initial steep curve, very easy to embrace
 - Principal developer already a team member

“devices” input file

site	device_name	make	model
tower	gw-tower	juniper	srx220
noc	sw-noc2	juniper	ex2200
heavy	st-esd-depot	juniper	ex2200

“ipaddrs” input file

static_ipnm	device	interface
100.96.209.2/24	st-fusionvalley	br0.209
162.212.145.2/28	stella-server	dnsonly

“vlans” input file

id	name	cat	gateway	type	src	static_ipnm
108	dmv	dept	gw-quad4	noradio	dhcpd	-
109	medical	dept	gw-heavy	noradio	gw	162.212.146.1/29
221	geeklandia	part	gw-noc	sta	dhcpd	-

```
system {
    host-name <%= @target["name"] %>;
    auto-snapshot;
    domain-name nv.burningman.com;

.....
<% if @target["model"] == "ex2200" then -%>
chassis {
    alarm {
        management-ethernet {
            link-down ignore;

.....
name-server {
<%
@target["!global!dns!servers"].each { |k,n|
-%>
        <%= n["server_ip"] %>;
```



```

<%
empty_node_if_nil(@target["interfaces"]).each_by_subkey(
  "name") { |k,n|
    if n["type"] == "trunk" then
->%>
      interface <%= n["name"] %>.0 {
        cost 1000;

        . . . . .

if ((n["dhcpsrc"] == "gw") && (n["gw"] == @target["name"])) then
  pool = {}
  pool["pool_ipnm"] = n["dhcp_subnet"]
  pool["description"] = n["name"]
  pool["domain_name"] = "nv.burningman.com"
  pool["pool_range_low"] = ip_union(n["dhcp_subnet_ip"], "0.0.0.8")
  pool["pool_range_high"] =
IPAddr.new(n["dhcp_subnet"]).to_range.last.mask("255.255.255.247").to_s
  pool["router"] = n["dhcp_gw_ip"]
  dhcp_pools << pool

```

```
% make
ncg -v playa.neto
NETOMATA_LIB=/usr/local/lib/netomata/lib
Reading file(s) playa.neto
| Reading file(s) ./users/users.neto
| | Reading table(s) ./users/users
| | Reading file(s) ./users/sshkeys/brent.neto...
| Reading file(s) ./gw/gw-tower.neto
| | Reading table(s) ./gw/gw.interfaces.hdr ./gw/gw-tower.interfaces
| | Reading file(s) ./gw/gw.per-device.neto
| Reading file(s) ./gw/gw-depot.neto
| | Reading table(s) ./gw/gw.interfaces.hdr ./gw/gw-depot.interfaces
| | Reading file(s) ./gw/gw.per-device.neto
| Reading file(s) ./gw/gw-noc.neto
| | Reading table(s) ./gw/gw.interfaces.hdr ./gw/gw-noc.interfaces
```

.....

```
% find ~/BurningMan/2014/ncg/generated -type f | wc -l
```

What Else?

- *Layer 1*
 - Remove any 2.4Ghz usage, available for participant WiFi, drone, etc. “last 100ft” usage
- *Layer 3*
 - Deprecate use of RFC1918, use RFC6598 instead (since we are the de facto service provider)
 - NAT in town during egress, never on playa
- *Layer 8*
 - Yearly refactoring of NCG
 - Too organized, no engineers sitting around during event = on-call pager duty rotation

The Future

- *Redundancy, multi-home*
 - Avoid SPOF (common mountain repeater site)
 - Current supplier is \$100/Mbps ...
 - Next available provider is 108 miles away
 - Join newly formed “TahoeIX” in Reno
 - *{Wink wink – Charter, CENIC, UNR}*
- *IPv6*
 - Tunnel performance was sub-optimal
 - WISP went dual-stack last month(!)

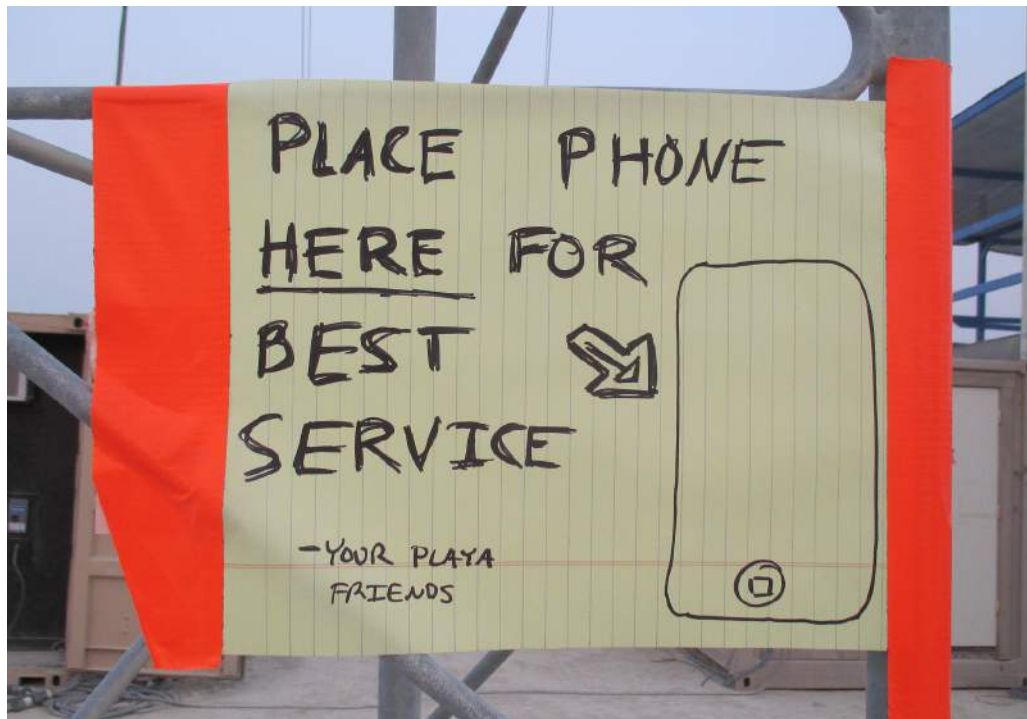
The Future

- *Great wall of congestion*
 - Too convenient for tube/flix, torrents, *Game of Thrones*, NSFW content
 - Capacity constrained
- *Revisit network automation*
 - Ansible...?



Summary Overview

- *{Automation} data modeling isn't easy*
 - Imagine all your inputs & outputs (device configs, DNS, monitoring, billing, etc.)
- *Single source of truth*
 - Git, a wiki, fancy IPAM: choose what fits your organization's workflow, stress level, & budget
- *Start at L8 and L1, meet in the middle*
 - People + physical layer = organic processes



Questions? Comments?

Rogue trenching
available in northern
Nevada..?

Have spare transit
capacity along I-80..?

Matt Peterson

matt@burningman.com