

DYNES

Dynamic Network System

DYNES: Building a Distributed Networking Instrument

Ben Meekhof

ATLAS Great Lakes Tier2

University of Michigan

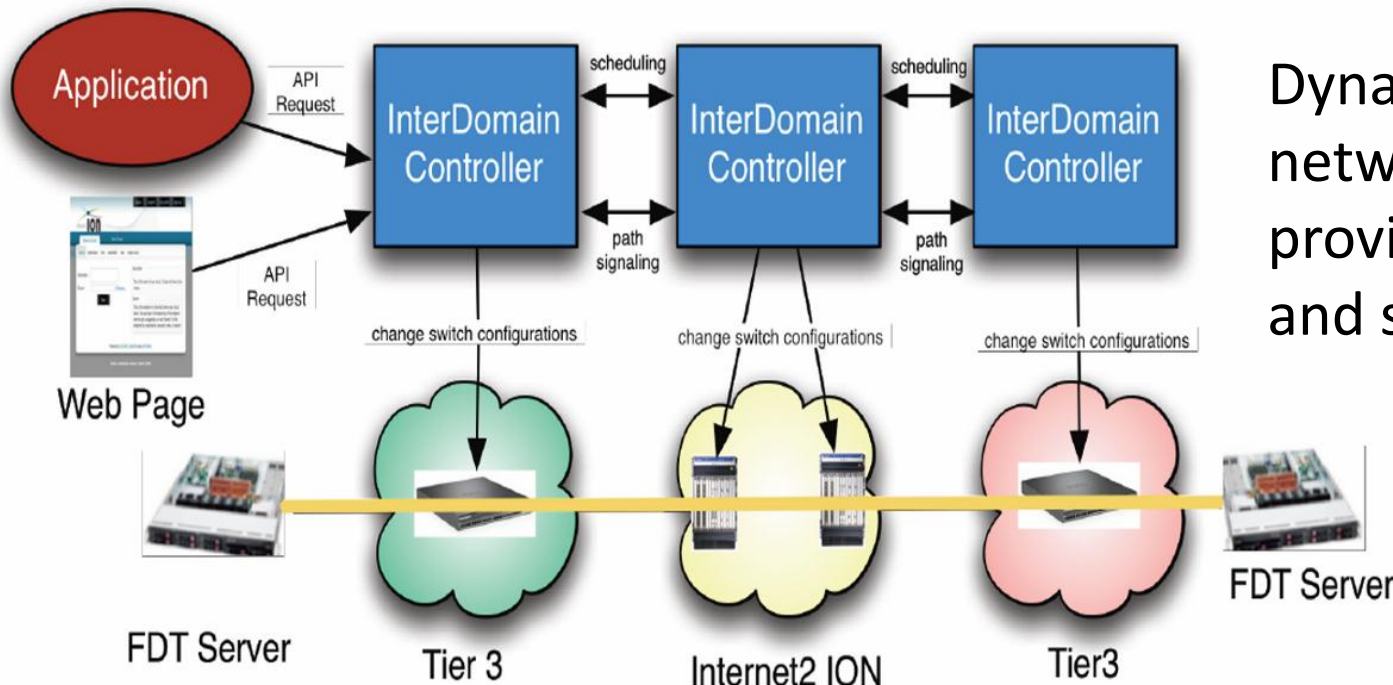
HEPiX 2012, Oct 15-19 2012



What is DYNES?

(other than a questionable acronym)

A nationwide cyber-instrument spanning about 40 US universities and 11 Internet2 connectors which interoperates with ESnet, GEANT, APAN, US LHCNet, and many others. Synergetic projects include OliMPS and ANSE.



Dynamic network circuit provisioning and scheduling

Uses For DYNES

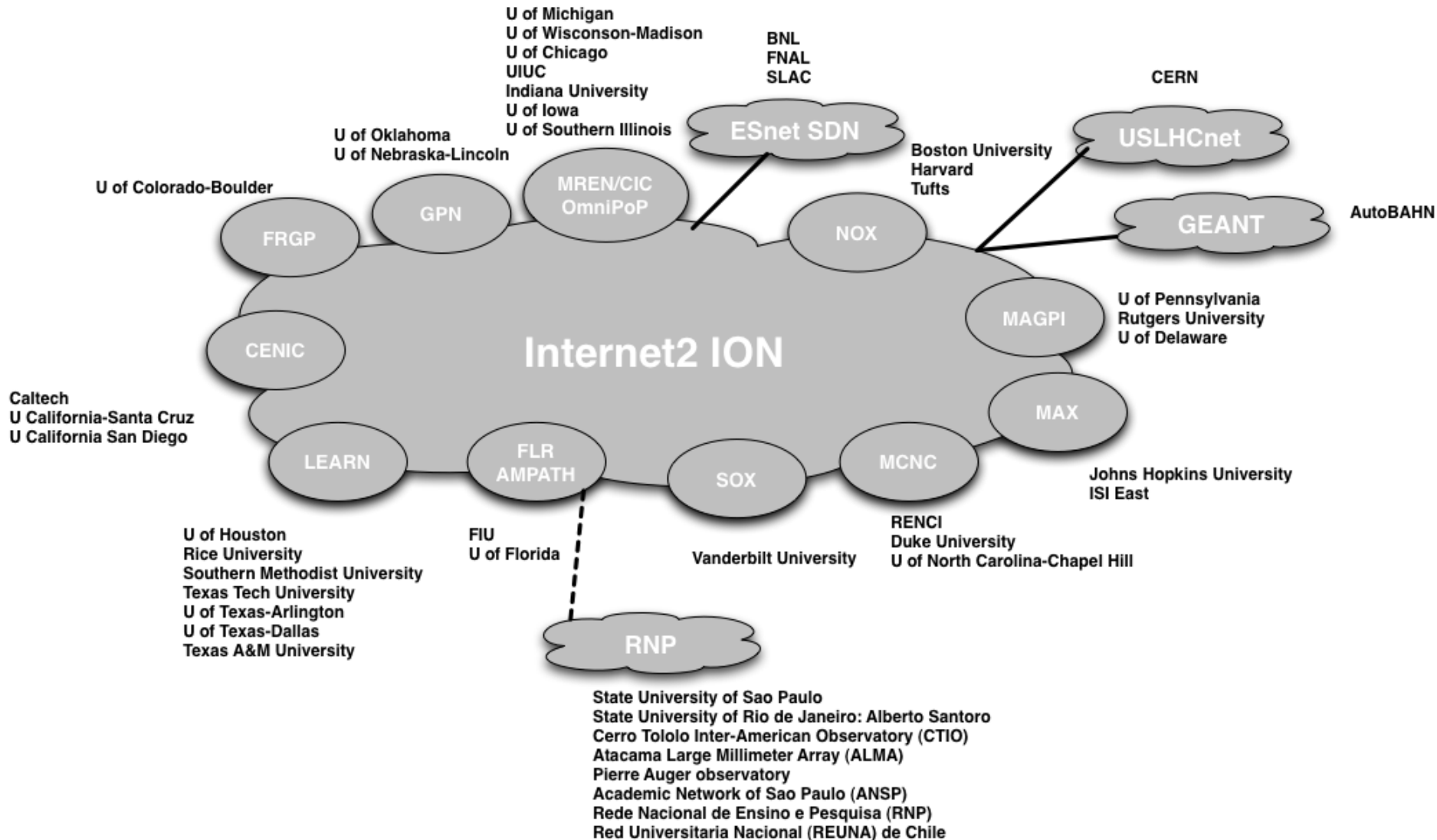
For regional networks and campuses to support large, long-distance scientific data flows

- In the LHC
- In other leading programs in data intensive science (such as LIGO, Virtual Observatory, and other large scale sky surveys)
- In the broader scientific community.

Broadening existing Grid computing systems by promoting the network to a reliable, high performance, actively managed component.

- The DYNES team will partner with the LHC and astrophysics communities, OSG, and Worldwide LHC Computing Grid (WLCG) to deliver these capabilities to the LHC experiment as well as others such as LIGO, VO and eVLBI programs

DYNES Deployments



DYNES Hardware Components



Fast Data Transfer Server

- Dell R510
- ~11TB storage for data caching



Inter-Domain Controller (IDC) Server

- Dell R310
- Interfaces with switch OS to make configuration changes as needed to map new circuits



Dell Force10 S4810



Dell 8024F

Requirements/Challenges

DYNES sites are expected to be autonomous after initial deployment.

That means no “formal” funding for centralized services (but...we still have some services).

Nonetheless, we need to have a way to deploy and if necessary modify system configurations to get all sites functional and mostly “hands-off” in the long run.

We also need to have a way to determine if sites are functional and notify them if not, especially in initial stages.

DYNES Software Components

IDC and FDT systems run Scientific Linux 5 or 6 (initially 5, now deploying on 6)

Circuit provisioning is done with OSCARS (On-Demand Secure Circuits and Advance Reservation System)

Data transfer with well-known Fast Data Transfer software.

Work underway to integrate OpenFlow capable switches - new firmware will support it on S4810

Monitor component status with Nagios

Now in process of deploying perfSONAR nodes to each site

Track network switch configuration updates with Rancid

Approaches

A centralized configuration manager (cfengine, ncm, puppet) was rejected

- Too complex and too centralized.
- Who maintains it? Does everyone understand how to use it?

Building a base config into RPMS made sense

- Anyone can build or install, updates can be deployed from a yum repository
- RPM post/pre scripts allow some scripting
- Can specify other package requirements in RPM spec

How to access systems for administration?

- Systems run a cron job which regularly fetches ssh public keys from UM webserver
- HTTP/SSL with verified certificate used to assure source identity

Approaches - Kickstart

To quickly get FDT/IDC systems built we generated site and system specific kickstart files that could be referenced by sites via http in the event that they needed to rebuild a system.

- IDC/FDT systems reference certain specific repositories or packages in the kickstart so they come up ready to go, appropriate kernel (FDT uses UltraLight kernel), appropriate base packages.

These files were created in a batch process (shell/perl) to be downloaded at install time over http.

- Batch scripts referenced collection of site config files
- Just a fun note: used perl Geo::IP module to set timezones in kickstarts

Example Kickstart

```
install
url --url http://mirror.anl.gov/pub/centos/6/os/x86_64/
repo --name=Updates --mirrorlist=http://dynes.grid.umich.edu/dynes/ks/centos6-mirrorlist-updates
repo --name=Install --mirrorlist=http://dynes.grid.umich.edu/dynes/ks/centos6-mirrorlist

# DYNES repos
repo --name=DYNES --baseurl=http://dynes.grid.umich.edu/dynes/repo/el6
repo --name=Internet2 --baseurl=http://software.internet2.edu/branches/aaron-testing/rpms/x86_64/main
repo --name=EPEL --mirrorlist=http://mirrors.fedoraproject.org/mirrorlist?repo=epel-6&arch=x86_64

# Kernel repo here for FDT only
repo --name=DYNES-kernel --baseurl=http://dynes.grid.umich.edu/dynes/kernel-repo/el6

logging --host=141.211.43.110 --level=debug
skipx

lang en_US.UTF-8
keyboard us

network --device eth3 --hostname fdt-umich.dcn.umnet.umich.edu --ip 192.12.80.86 --netmask 255.255.255.252 --gateway 192.12.80.85 --nameserver
141.211.125.17 --onboot yes --bootproto static --noipv6
network --device eth1 --onboot yes --bootproto static --ip 10.10.3.240 --netmask 255.255.252.0 --noipv6

rootpw --iscrypted $1$qeLsd;fsdkljfklSdsdfnotourpasswordreally
firewall --enabled --port=22:tcp
authconfig --enableshadow --enablemd5
selinux --disabled
firstboot --disable
timezone America/New_York
ignoredisk --drives=sda
bootloader --location=mbr --driveorder=sdb --append="rhgb quiet selinux=0 panic=60 printk.time=1"

# partitions
clearpart --all --drives=sdb
part /boot --fstype=ext4 --size=500 --ondisk=sdb
part pv.dynes --size=1 --grow --ondisk=sdb
volgroup vg_dynes --pesize=4096 pv.dynes
logvol / --fstype=ext4 --name=lv_root --vgname=vg_dynes --size=1024 --grow
logvol swap --fstype=swap --name=lv_swap --vgname=vg_dynes --size=4096
```

Approaches - Switches

Dell/Force 10 switches, like many switches, can be pointed to an initial configuration file available over TFTP when PXE booted out of the box

- Specify switch MAC and initial configuration file in DHCP server config, then PXE boot switches
- Batch scripts created site specific switch config files from site config files and placed into appropriate location on our tftp host

Configuration files are packaged and installed on IDC hosts

- Batch scripts package switch config files into dynes-base-idc RPM
- RPM at install sets up simple DHCP and TFTP servers (not enabled by default) which can be used to repeat the initial configuration process if a switch is ever replaced

Example DHCP Config

```
# For s4810 BMP (bare metal provisioning)
# option configfile code 209 = text;
# option tftp-server-address code 150 = ip-address;
# option tftp-server-address 10.1.1.10;
# option bootfile-name code 67 = text;

subnet 10.1.1.0 netmask 255.255.255.0 {
    range 10.1.1.200 10.1.1.209;
    option subnet-mask 255.255.255.0;
    default-lease-time 1200;
    max-lease-time 1200;
    # option routers 10.1.1.10;
    option domain-name "local";
    option broadcast-address 10.1.1.255;
    next-server 10.1.1.10;
group "local" {
    # rice S4810
    #host rice.local {
    #    hardware ethernet 00:01:e8:8b:09:a6;
    #    option configfile "/dynes/switch-configs/dynes-switch-config-rice.cfg";
    #    option bootfile-name "/dynes/images/FTOS-SE-8.3.10.1.bin";
    #}
    host iowa.local {
        hardware ethernet 5C:26:0A:F4:F7:6F;
        option bootfile-name "/dynes/switch-configs/dynes-switch-config-iowa.cfg";
    }
    host harvard.local {
        hardware ethernet 5C:26:0A:F4:F7:5F;
        option bootfile-name "/dynes/switch-configs/dynes-switch-config-harvard.cfg";
    }
}
```

DYNES RPMS

dynes-base

- configures core services like logging to DYNES loghost, snmp communities, ntp, perfSonar services (owamp), ssh. Also includes many configuration scripts.

dynes-config-
sitename

- puts in place site specific config files (same file used to build switch and server configs, now used locally for DYNES software config)

dynes-base-idc

- specific to IDC. Includes switch configuration and docs

dynes-base-fdt

- specific to FDT. Requires special kernel repo. Packages script to setup storage post-install.

dynes-nagios

- requires Nagios RPMS (EPEL repo) and installs public key used by nagios server to run checks.

dynes-repo-
kernel

- Ultralight kernels for FDT

Yum repository

Configuration updates will be automatically grabbed by yum update, but sites always have the option to disable the DYNES repos and update as they wish.

Example: After the initial installation run we wanted to incorporate Nagios. We packaged our Nagios setup into an RPM and made dynes-base require that RPM. Next yum update, all systems were accessible by Nagios.

Fairly low maintenance to maintain

Disadvantage that we have to be careful not to break yum updates with bad dependency specifications.

Configuration Scripts

install_dynes.sh

- Runs other config scripts
- Run manually after kickstarting system/installing RPMS

install_dell.sh

- Installs Dell Yum repository (source for firmware updates, OM software)
- Sets up Dell OpenManage software for CLI interface to hardware (BIOS, Storage Controller, etc)
- Updates firmware, configures settings for AC power recovery, CPU VT

dell_alerts.pl

- Configures OM software to email alerts to DYNES admin list

idrac6_setup.sh

- Configures Dell Remote Access Controller network and user info (references dynes-config-site file installed in /etc/dynes by RPMS)

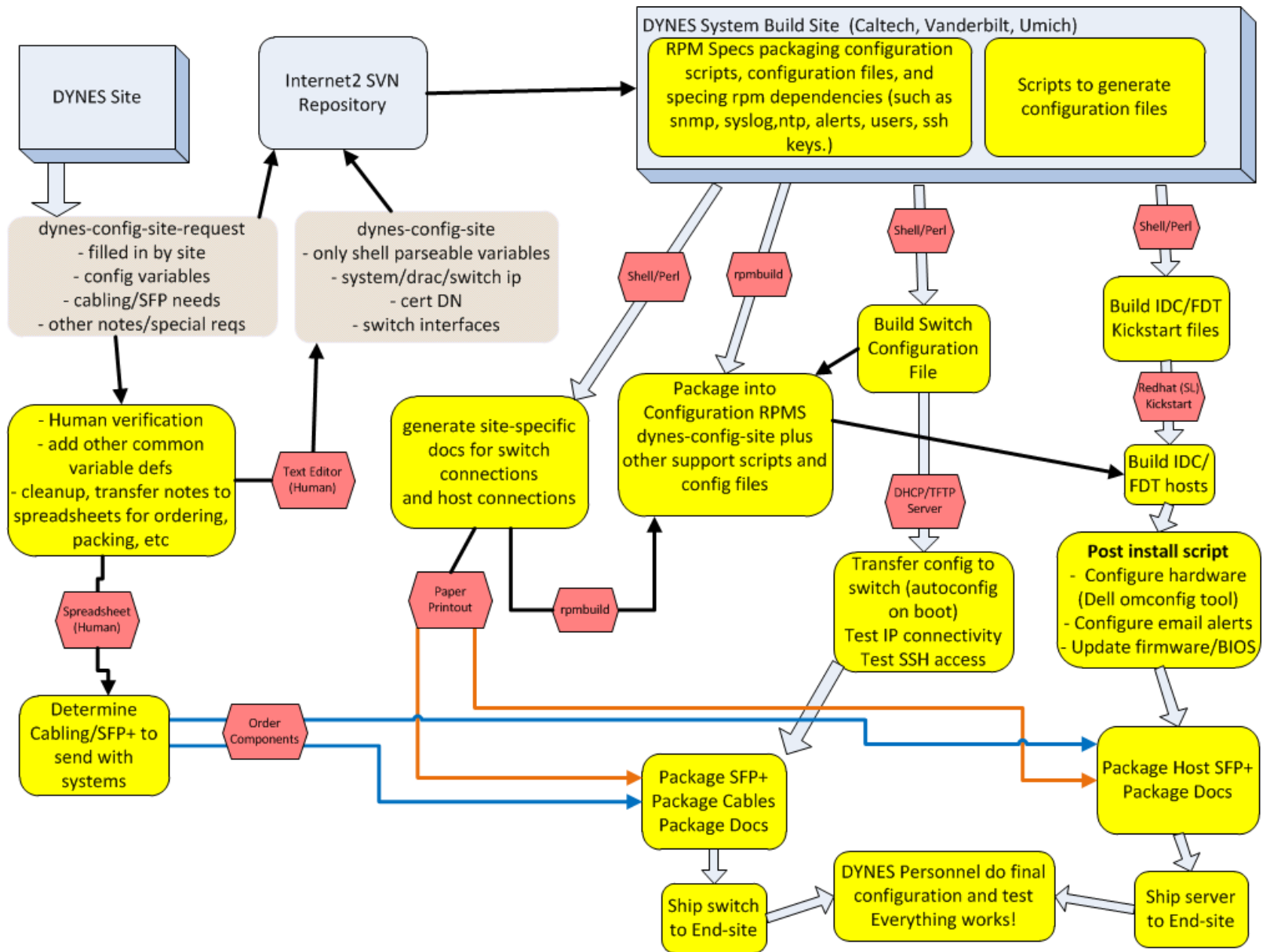
setup_storage.sh

- Configures RAID-0 volume for data caching (runs on FDT only)

configure_net.sh

- Configures bridged network interface, needed by KVM. DYNES IDC controller distributed as VM.

Deploying The Instrument



Monitoring the Instrument

Though the ideal is to have no central point of service it was decided that we need some way to know how things are going

- Nagios is well known
- Can script nagios checks for more detailed functional status

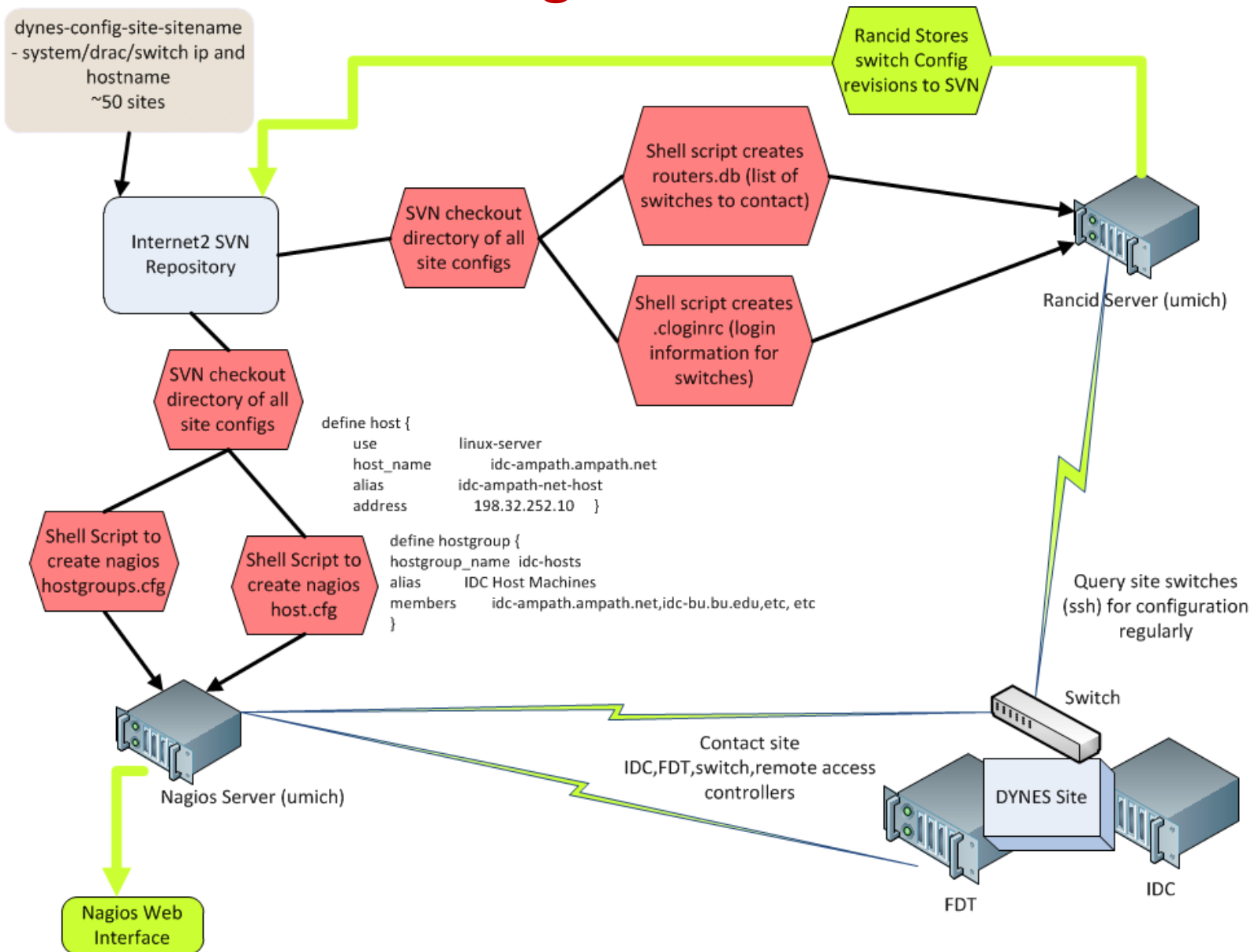
We needed a way to track switch configurations for sites in case of breakage or to restore from emergency

- Rancid has “saved” us at AGLT2 a couple times
- Can store configs to any SVN repository – use web interface to Internet2 repo to reference configs easily

Our installation includes Dell OpenManage software configured to send email alerts for system problems

- It’s easy to rack a system and never look at it, email alerts assure we can inform sites of problems
- CLI utils included in OM are useful

Monitoring the Instrument



Nagios Monitor

Current Network Status

Last Updated: Thu Oct 11 10:55:21 EDT 2012
 Updated every 90 seconds
 Nagios® Core™ 3.3.1 - www.nagios.org
 Logged in as nagiosadmin

[View Service Status Detail For All Host Groups](#)
[View Host Status Detail For All Host Groups](#)
[View Status Summary For All Host Groups](#)
[View Status Grid For All Host Groups](#)

Host Status Totals

Up	Down	Unreachable	Pending
146	71	0	0
All Problems		All Types	
71		217	

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
134	5	2	65	0
All Problems		All Types		
72		206		

Service Overview For All Host Groups

FDT Host Machines (fdt-hosts)

Host	Status	Services	Actions
dynes-fdt-utd.utdallas.edu	DOWN	1 CRITICAL	
dynes-fdt.physics.uiowa.edu	UP	1 OK	
dynes-fdt.unl.edu	UP	1 OK	
fdt-bu.bu.edu	DOWN	1 CRITICAL	
fdt-caltech.hep.caltech.edu	UP	1 OK	
fdt-dynes-ucb.colorado.edu	UP	1 OK	
fdt-dynes1sum.fas.harvard.edu	DOWN	1 CRITICAL	
fdt-flr.net.flnet.org	UP	1 OK	
fdt-indiana-2.bls.w.net.rt.iu.edu	DOWN	1 CRITICAL	
fdt-indiana.ctc.net.rt.iu.edu	UP	1 OK	
fdt-jhu.jhu.edu	UP	1 OK	
fdt-learn.tx-learn.net	DOWN	1 UNKNOWN	
fdt-max.dragon.maxgigapop.net	DOWN	1 CRITICAL	
fdt-mren.mren.org	DOWN	1 UNKNOWN	
fdt-ou.dynes.ou.edu	UP	1 OK	
fdt-renci.renci.org	UP	1 OK	
fdt-rice.dynes.rice.edu	UP	1 OK	
fdt-rutgers.rutgers.edu	UP	1 OK	
fdt-smu.smu.edu	UP	1 OK	
fdt-texastech.ttu.edu	DOWN	1 CRITICAL	
fdt-tufts.tufts.edu	UP	1 OK	
fdt-uchicago.uchicago.edu	DOWN	1 CRITICAL	
fdt-ucsc.ucsc.edu	UP	1 OK	

FDT Remote Access Controllers (fdt-racs)

Host	Status	Services	Actions
dynes-fdt-idrac.physics.uiowa.edu	DOWN	1 CRITICAL	
fdt-caltech-idrac.hep.caltech.edu	UP	1 OK	
fdt-dynes1sum-idrac.fas.harvard.edu	DOWN	1 CRITICAL	
fdt-flr-idrac.net.flnet.org	UP	1 OK	
fdt-idrac-dynes-ucb.colorado.edu	UP	1 OK	
fdt-idrac-uchicago.uchicago.edu	DOWN	1 CRITICAL	
fdt-idrac.ucsc.edu	DOWN	1 CRITICAL	
fdt-idrac6.physics.indiana.edu	DOWN	1 CRITICAL	
fdt-idrac6.ucsd.edu	DOWN	1 CRITICAL	
fdt-indiana-idrac.ctc.net.rt.iu.edu	UP	1 OK	
fdt-jhu-idrac6.jhu.edu	UP	1 OK	
fdt-mgmt.bu.edu	DOWN	1 CRITICAL	
fdt-ou-idrac.dynes.ou.edu	UP	1 OK	

IDC Host Machines (idc-hosts)

Host	Status	Services	Actions
dynes-idc-utd.utdallas.edu	UP	1 OK	
dynes-idc.net.uiowa.edu	UP	1 OK	
dynes-idc.unl.edu	UP	1 OK	
idc-ampath.ampath.net	UP	1 OK	
idc-bu.bu.edu	DOWN	1 CRITICAL	
idc-caltech.hep.caltech.edu	UP	1 OK	
idc-dynes-ucb.colorado.edu	UP	1 OK	
idc-dynes1sum.fas.harvard.edu	DOWN	1 CRITICAL	
idc-flr.net.flnet.org	DOWN	1 CRITICAL	
idc-frgp.dynes.frgp.net	DOWN	1 CRITICAL	
idc-indiana.ctc.net.rt.iu.edu	UP	1 OK	
idc-jhu.jhu.edu	UP	1 OK	
idc-learn.tx-learn.net	DOWN	1 CRITICAL	
idc-max.dragon.maxgigapop.net	UP	1 OK	
idc-mren.mren.org	UP	1 WARNING	
idc-nox300bent.nox.org	UP	1 OK	
idc-ou.dynes.ou.edu	UP	1 OK	
idc-renci.renci.org	UP	1 OK	
idc-rice.dynes.rice.edu	UP	1 OK	
idc-rutgers.rutgers.edu	DOWN	1 CRITICAL	
idc-smu.smu.edu	UP	1 OK	
idc-sox.sox.net	UP	1 OK	
idc-sys.dynes.upenn.edu	UP	1 OK	

Nagios Monitor

Status Summary For All Host Groups

Host Group	Host Status Summary	Service Status Summary
FDT Host Machines (fdt-hosts)	<p>24 UP</p> <p>10 DOWN : 10 Unhandled</p>	<p>24 OK</p> <p>2 UNKNOWN : 2 on Problem Hosts</p> <p>8 CRITICAL : 8 on Problem Hosts</p>
FDT Remote Access Controllers (fdt-racs)	<p>15 UP</p> <p>9 DOWN : 9 Unhandled</p>	<p>15 OK</p> <p>9 CRITICAL : 9 on Problem Hosts</p>
IDC Host Machines (idc-hosts)	<p>30 UP</p> <p>8 DOWN : 8 Unhandled</p>	<p>29 OK</p> <p>1 WARNING : 1 Unhandled</p> <p>8 CRITICAL : 8 on Problem Hosts</p>
IDC Remote Access Controllers (idc-racs)	<p>18 UP</p> <p>12 DOWN : 12 Unhandled</p>	<p>17 OK</p> <p>1 WARNING : 1 Unhandled</p> <p>12 CRITICAL : 12 on Problem Hosts</p>
IDCs running OSCARS v5 (idcs-v5)	<p>24 UP</p> <p>14 DOWN : 14 Unhandled</p>	<p>23 OK</p> <p>1 WARNING : 1 Unhandled</p> <p>14 CRITICAL : 14 on Problem Hosts</p>
Local Hosts (local-hosts)	<p>1 UP</p>	<p>7 OK</p> <p>1 WARNING : 1 Unhandled</p>
Switch J8697A (sw-J8697A)	<p>1 UP</p>	<p>1 OK</p>
Switch PC6248 (sw-PC6248)	<p>2 UP</p>	<p>2 OK</p>
Switch PC8024F (sw-PC8024F)	<p>18 UP</p> <p>14 DOWN : 14 Unhandled</p>	<p>17 OK</p> <p>1 WARNING : 1 Unhandled</p> <p>14 CRITICAL : 14 on Problem Hosts</p>
Switch S4810 (sw-S4810)	<p>1 UP</p> <p>2 DOWN : 2 Unhandled</p>	<p>No matching services</p>

Conclusion

- Our deployment procedure has worked pretty well. Sites are consistent and generally functional out of the box.
- We have a pretty good idea of status from Nagios and can tell at a glance which sites are not reachable.
- Biggest issue has been making sure we adequately document how site admins can access their own systems
 - ...and remember to put that document in the box!
- Second big issue in monitoring and config tracking is sites that (understandably) don't like to have switches on public net. Most are ok once we tell them the limited ACL we put on the switch.

More Information

<http://www.internet2.edu/dynes>