NARLabs 國家實驗研究院

國家高速網路與計算中心

大資料傳輸與網路效能調適方法

楊哲男

2018/05/31

Today’s Talk

- Goal
  - 了解檔案傳輸時可能發生的問題瓶頸點
  - 了解傳輸大檔案時如何優化設定參數
    - 使用者如何調整
- Outline
  - 檔案傳輸之問題
  - 各種Level之調整方法
  - 網路效能監控
  - DTN介紹
  - 大資料傳輸工具選擇
  - 傳輸demo

Data transfer time

- You are getting speeds less than (for大檔案傳輸):
  - 校園內傳輸
    - 800 Mbps(當網路頻寬為 1 Gbps時)
      - **100 GB** = ~800,000 Mb. Takes ~ 17 minutes
    - 3-4 Gbps (當網路頻寬為 10 Gbps時)
      - **100 GB** = ~800 Gb. Takes ~ 5 minutes
  - 跨校傳輸
    - 200 Mbps –5 Gbps
      - **100 GB** = ~800,000 Mb. Takes ~ 1 hour- 4 minutes

Typical Scenario

- 使用者欲下載檔案，如iso檔案
- 使用一般常用scp或是透過網頁抓取
- 使用者所期待的
  - 對外網路是1Gbps
  - 4GB(4000MB) * 8 = 32000Mb
  - 預期傳輸時間:32000Mb / 1000Mbps = 32秒
- 可能出現的結果
  - 傳輸速度顯示每秒只有幾MB或是幾KB的速度
  - 網路擁塞及其他因素?
  - 使用者主機效能、通訊協定的選擇、應用程式的效能?
- 該如何解決??

使用者解決步驟

- Application Level
  - 是否有其他使用者反應一樣問題？所使用的傳輸程式版本是否為最新？傳輸的工具是否適合?
- Protocol Level
  - TCP/IP通訊協定是否有做過調整優化？
- Host Level
  - 硬體(網卡、CPU)、軟體(驅動程式、作業系統)是否運作正常
- LAN Networks
  - 詢問當地的網管目前之網路狀態
- Backbone Networks
  - 詢問遠端之骨幹網管目前之網路狀態

實際可能發生情形

- Application Level
  - 這步驟會被省略，程式開發者或使用者會直接抱怨網路有問題
- Protocol Level
  - 直覺認為這部份應該是自動化調整才對
- Host Level
  - Ping得通之後，就停止檢查及診斷主機狀態
- LAN Networks
  - 網管認為內部網路是良好的並認為是來源端的網路問題
- Backbone Networks
  - 網管只監控網路流量，認為骨幹還很空，應該是來源端或目的端的網路問題

解決時可能發生的問題

- 缺乏清楚的處理程序
  - 正確處理問題的程序其所需的知識很重要
  - 所需的知識不僅對使用者很重要，對於程式開發者及網路管理者亦相當的重要
- 缺少耐心
  - 使用者對管理者抱怨，而管理者不想聽到使用者的抱怨
- 無效的資訊來源
  - 缺乏有效的效能資訊，例如目前的提供對內、對外的網路之效能為何?
- 溝通問題
  - 有問題時該找誰詢問及幫忙(往往會不知找誰)

Data transfer: Overview

The key players

- Endpoints
- Network
- Transfer tool
- Transfer settings

- That`s a lot of work

Background Information

- Bandwidth*Delay Products (BDP)
  - 兩點間可用最小可用頻寬與延遲往返時間之乘積
  - 在一時間點上連結資料的容量，點對點間路徑最大可被傳送的容量
  - 例如:

BW = 10 Gbps
RTT = 10 msec
BDP = (10000000000 bps) * (0.01 sec) / (8 bits/byte) = 12.5MBytes

BW = 1 Gbps
RTT = 100 msec
BDP = (1000000000 bps) * (0.1 sec) / (8 bits/byte) = 12.5MBytes

TCP Review

- TCP 使用“congestion window, CWND”決定同時間有多少封包數可以同時被傳送.
- 越大的CWND數,越高的效能(throughput)
  - Throughput = Window size / Round-trip Time
- TCP“slow start”與“congestion avoidance”等演算法決定CWND之大小
- TCP buffer size 與CWND相關, 最佳的buffer size 為BDP值

主機端之調教

- 修改client端及server端之
  - OS buffer size(根據BDP)
  - MTU size(使用jumbo frame, 需點對點間設備全部支援)
  - 改變congestion control algorithm
  - WSCALE、SACK、NIC queue(使用10G網卡時)相關參數
  - 修改Interrupt Binding (使用多張10G網卡或40G網卡時)
- OS Auto tuning 預設最大 buffer size
  - Linux 2.6以後: 4MB
  - Windows 7後: 16MB
  - Mac OSX 10.5-10.6: 4MB
  - Mac OSX 10.7以後: 8MB
  - FreeBSD 7: 256KB
  - FreeBSD 8以後: 2MB

OS window size

- Linux
  - Linux 2.6後雖然支援Auto tuning，但預設之最大TCP buffer sizes還是太小
  - 修改/proc/sys/net/core/ ，或是利用sysctl指令，或是直接寫在/etc/sysctl.conf
    - 修改TCP 最大buffer size值 (例如修改至32MB)

      net.core.rmem_max = 33554432
      net.core.wmem_max = 33554432
    - 修改autotuning TCP buffer size值

      net.ipv4.tcp_rmem = 4096 87380 33554432 (min, default, max number)
      net.ipv4.tcp_wmem = 4096 65536 33554432

      net.core.netdev_max_backlog=25000 # for 10G NIC
    - 修改congestion control演算法

      net.ipv4.tcp_congestion_control = htcp

      其他設定(預設, 可檢查是否正確)

      net.ipv4.tcp_timestamps=1

      net.ipv4. tcp_window_scalings=1

      net.ipv4.tcp_tcp_sacks=1

OS window size

- Windows
  - Windows 7 / Windows Vista
    - 支援 TCP Autotuning，最大 receive window size 16 MB
  - Windows 8 / Windows 10:
    - 最大 receive window size 支援到1GB
    - 可利用Set-NetTCPSetting指令修改InitialCongestionWindowMss及Compound TCP(an advanced TCP congestion control algorithm)
    - 設定調整AutoTuningLevelLocal
- Mac OSX (/etc/sysctl.conf)
  - net.inet.tcp.win_scale_factor=8 (default為3或5)
  - 修改TCP 最大buffer size值
    - kern.ipc.maxsockbuf=16777216
  - For OSX Yosemite/Mavericks/Sierra
    - net.inet.tcp.win_scale_factor=8
    - net.inet.tcp.autorcvbufmax=33554432
    - net.inet.tcp.autosndbufmax=33554432

Network Monitoring

- delay與packet loss影響網路效能
  - delay越長，若不做任何調整，效能將不如預期
  - 當loss發生，封包必須重傳，並且降速，會造成效能之惡性循環
  - 因為loss發生所需回復的時間，亦將造成RTT時間變長，因此效能又會降低
- 隨時監控網路delay、對外網路可用頻寬等資訊
- 長時間觀察網路品質及趨勢

為何需要網路效能量測?

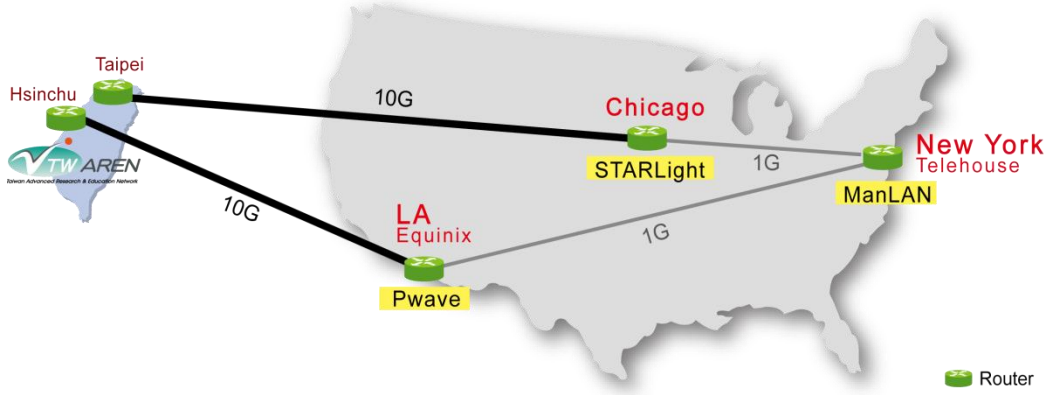
- 網路應用者對於網路品質是貪婪的
- 網路設計者較考慮保護性(protection)及可用性(availability)甚於網路效能
- 頻寬再大，但效能品質不佳，無法發揮效益
- 了解管理單位至連線單位之連線品質
- “**Big**” Science正在全球流行，唯有良好的品質，才可傳輸海量資料

為何需要網路效能量測?

- 由於TWAREN網路管理之設備眾多，當設備故障、接頭老化、管溝施工等眾多物理性因素，甚至是人為設定失誤、網路攻擊等因素影響品質。
- 許多的網路管理工具大都只監控線路是否斷線或是設備是否故障等硬故障(hard failures) 。
- 使用 **Server-Server** 的主動式模式進行，將量測伺服器佈署在網路上各節點並週期性地進行監測，它可發現比較難發現的軟故障(soft failures) 。
- 當骨幹發生了軟故障時，它並不會出現無法傳輸的問題，但卻會出現傳輸效能降低的問題
- **長期監控效能**，了解網路品質是否有異常(即時今天網路沒問題，也無法保證明日網路是好的)
- 以TWAREN骨幹為例...

TWAREN 網路架構

承諾 · 熱情 · 創新



TWAREN國內骨幹網路有:

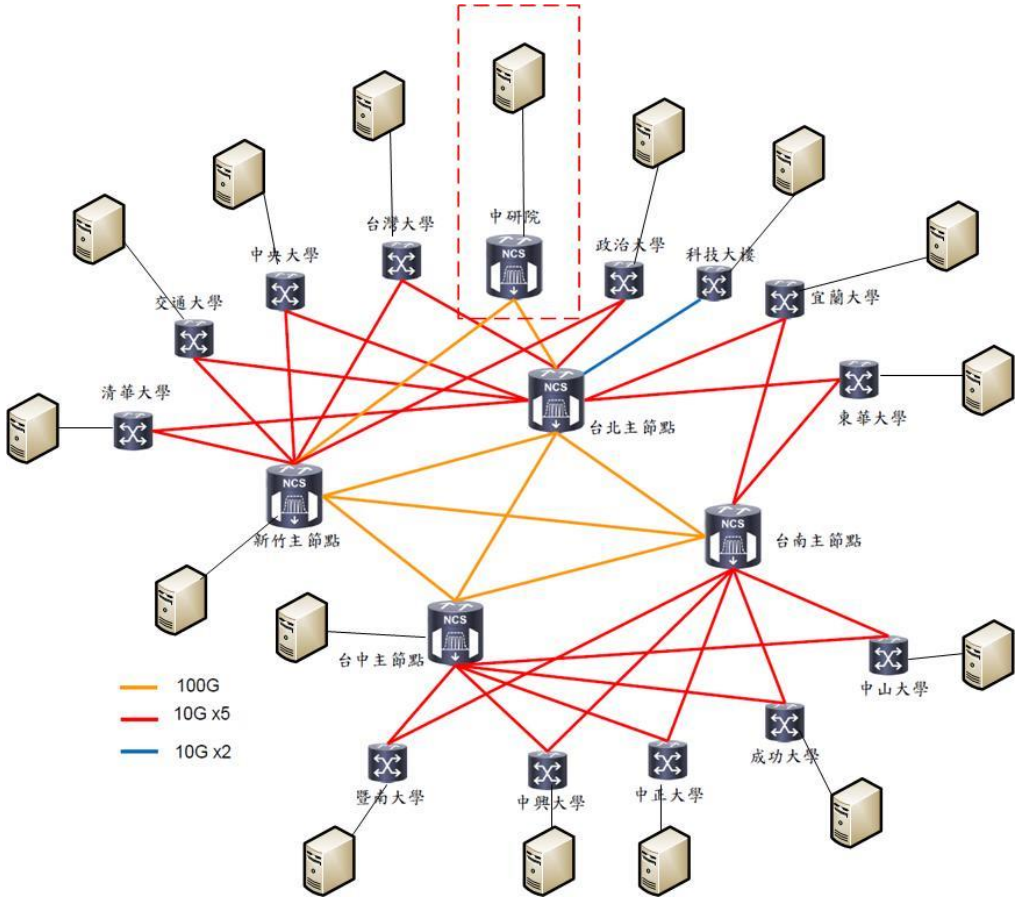
- 五個核心主節點(Core Node)、
- 十二個區域網路中心(GigaPOP)

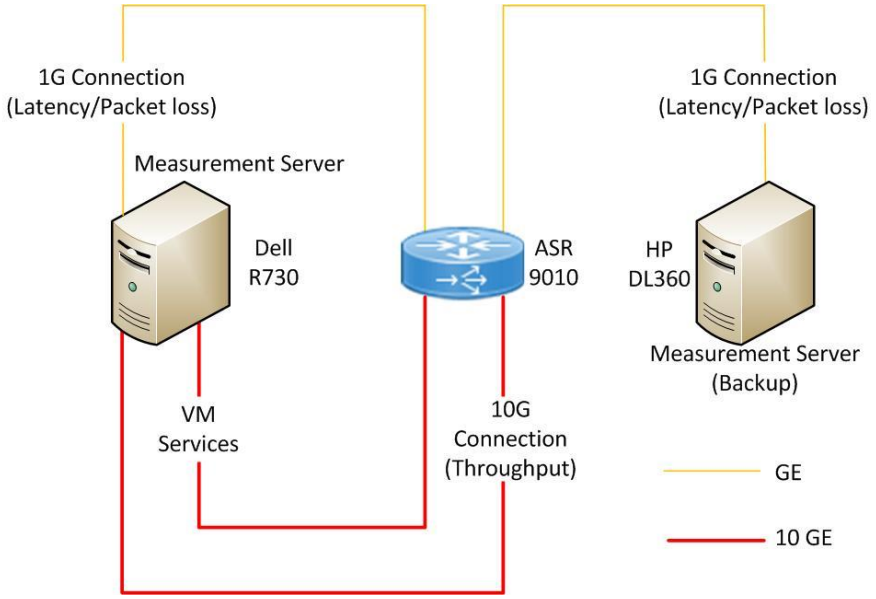
TWAREN國際骨幹網路有:

- 三個網路交換點(Exchange Node)
  - ➢ 洛杉磯、芝加哥、紐約

TWAREN perfSONAR量測架構

承諾・熱情・創新





- 每個節點建置兩台伺服器互為備援
- 兩台伺服器皆執行IPv4及IPv6網路封包遺失、RTT與可用率之計算
- 收集到之資料會回傳至TWAREN 維運中心資料庫並產生報表

每日報表範例

TWAREN-POP Packet Loss Rate (%) -- 2017-05-24 00:00:00 ~ 2017-05-24 23:59:59 --																	
Soc/Des	TP	HC	TC	TN	NTU	SINICA	NIU	NDHU	NSYSU	NCKU	CCU	NCNU	NCHU	NTHU	NCTU	NCU	TRTC	NCCU
TP	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HC	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TC	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TN	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0
NTU	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0
SINICA	0.003	0.002	0.001	0.001	0.001	-	0.002	0.002	0.002	0.002	0.001	0.001	0.001	0.001	0.002	0.001	0.002	0.002
NIU	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0
NDHU	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0
NSYSU	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0
NCKU	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0
CCU	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0
NCNU	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0
NCHU	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0
NTHU	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0
NCTU	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0
NCU	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0
TRTC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0
NCCU	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-

狀況描述 (如何發現)	國網中心於5/24 19:12~19:14、19:18~19:20監控到光纜出現異常狀況
原由探討 (判斷原因)	
影響範圍 (地區與用戶)	A7東華大學-台北主節點

選擇perfSONAR之原因

- 利用各地的路由器(Router)兩兩相互執行ping指令並收集RTT(Round Trip Time)及計算可用率，此監控模式我們可以稱為是Router based的網路效能量測方式。
- 然而因Router的設計會在設備忙碌時專注於封包Forwarding，優先忽略回應ICMP。因此Router間互ping易受Router負載影響，在網路仍然順暢，客戶端體驗正常時，即有可能在量測結果中出現封包遺失，誤導網路品質的判斷結果。
- Server間的量測封包對於Router來說屬於使用者封包，優先權與一般訊務相同。因此量測封包是否遺失或被主動丟棄，可以等量反應正常訊務流經相同路段時的狀況。
- Server-based perfSONAR除了一般的ping之外，還支援單向(One Way) ping，能夠反映網路單向的品質，更有利於網路單邊障礙，例如接頭鬆脫、GBIC/SFP模組故障的位置確認，有必要時亦可進行定時頻寬量測。

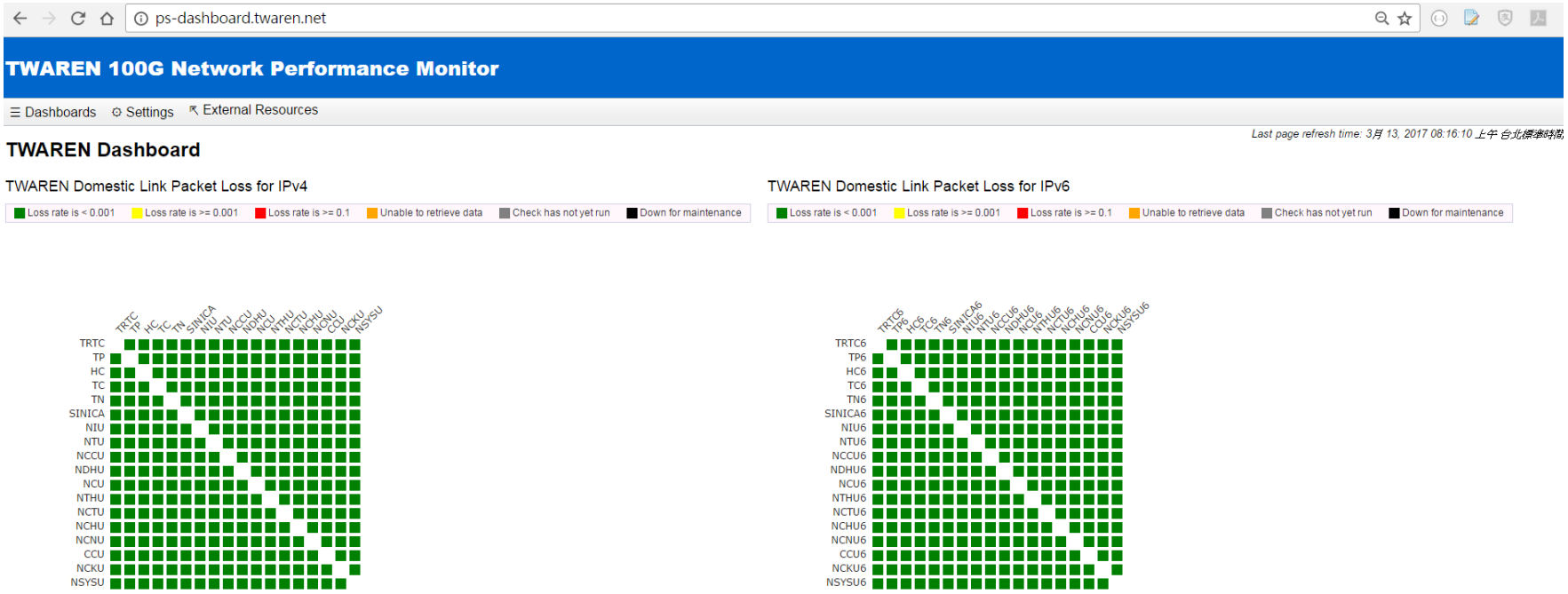
perfSONAR安裝與執行方式

- 主機安裝完成perfSONAR Toolkit。
- 透過中央統一管理的方式，讓每台主機讀取遠端的同一個json設定檔案。
- 每台主機執行OWAMP及PingER等daemons，執行fully-mesh之監控。
- 每分鐘定期量測one-way之延遲時間及封包遺失率，每5分鐘定期量測RTT之延遲時間。
- 透過TWAREN Dashboard作為統一管理之入口網站。


```
{
  "members": {
    "members": [
        "trtc-ow6.twaren.net",
        "tp-ow6.twaren.net",
        "hc-ow6.twaren.net",
        "tc-ow6.twaren.net",
        "tn-ow6.twaren.net",
        "sinica-ow6.twaren.net",
        "niu-ow6.twaren.net",
        "ntu-ow6.twaren.net",
        "ndhu-ow6.twaren.net",
        "ncu-ow6.twaren.net",
        "nthu-ow6.twaren.net",
        "nctu-ow6.twaren.net",
        "nchu-ow6.twaren.net",
        "ncnu-ow6.twaren.net",
        "ccu-ow6.twaren.net",
        "ncku-ow6.twaren.net",
        "nsysu-ow6.twaren.net",
        "nccu-ow6.twaren.net"
    ],
    "type": "mesh"
  },
  "parameters": {
      "force_bidirectional": "1",
      "bucket_width": "0.001",
      "ipv6_only": "1",
      "packet_padding": "0",
      "sample_count": "600",
      "packet_interval": "0.1",
      "type": "perfsonarbuoy/owamp"
  },
  "description": "Loss Test Between TWAREN Domestic Latency New Hosts for IPv6"
},
```

TWAREN perfSONAR Dashboard

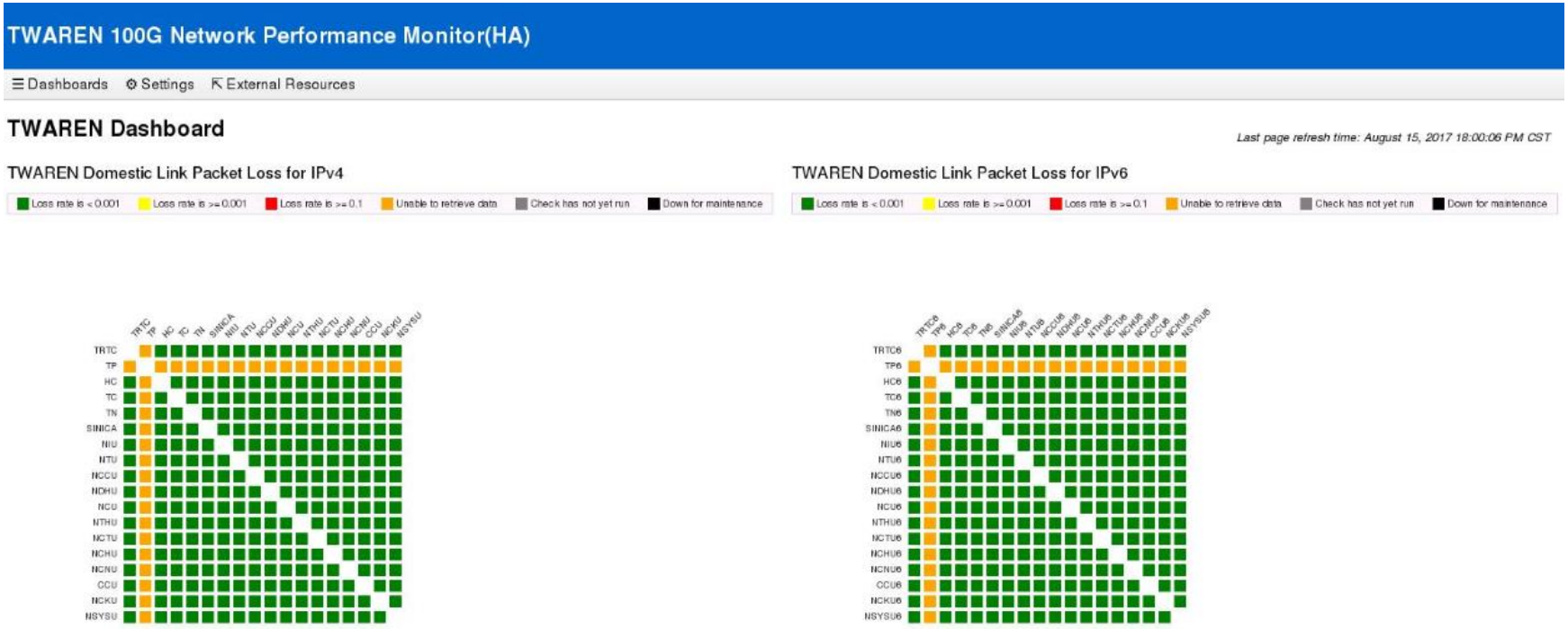
承諾・熱情・創新



- 透過國內TWAREN骨幹18台主機兩兩互相監測，共會收集到306筆資料
- 利用OWAMP量測單邊之延遲時間即封包遺失率來量測點對點的效能
- 透過矩陣式的統一管理方式來收集各節點之one-way延遲時間及封包遺失率等資料
- 設定臨界值之方式來發出告警通知管理中心調查異常原因，利用顏色來表現封包遺失率之影響情形，其中綠色代表封包遺失率低於百分之0.001，黃色代表封包遺失率介於百分之0.001至百分之0.1之間，紅色代表封包遺失率大於百分之0.1

三峽主節點停電

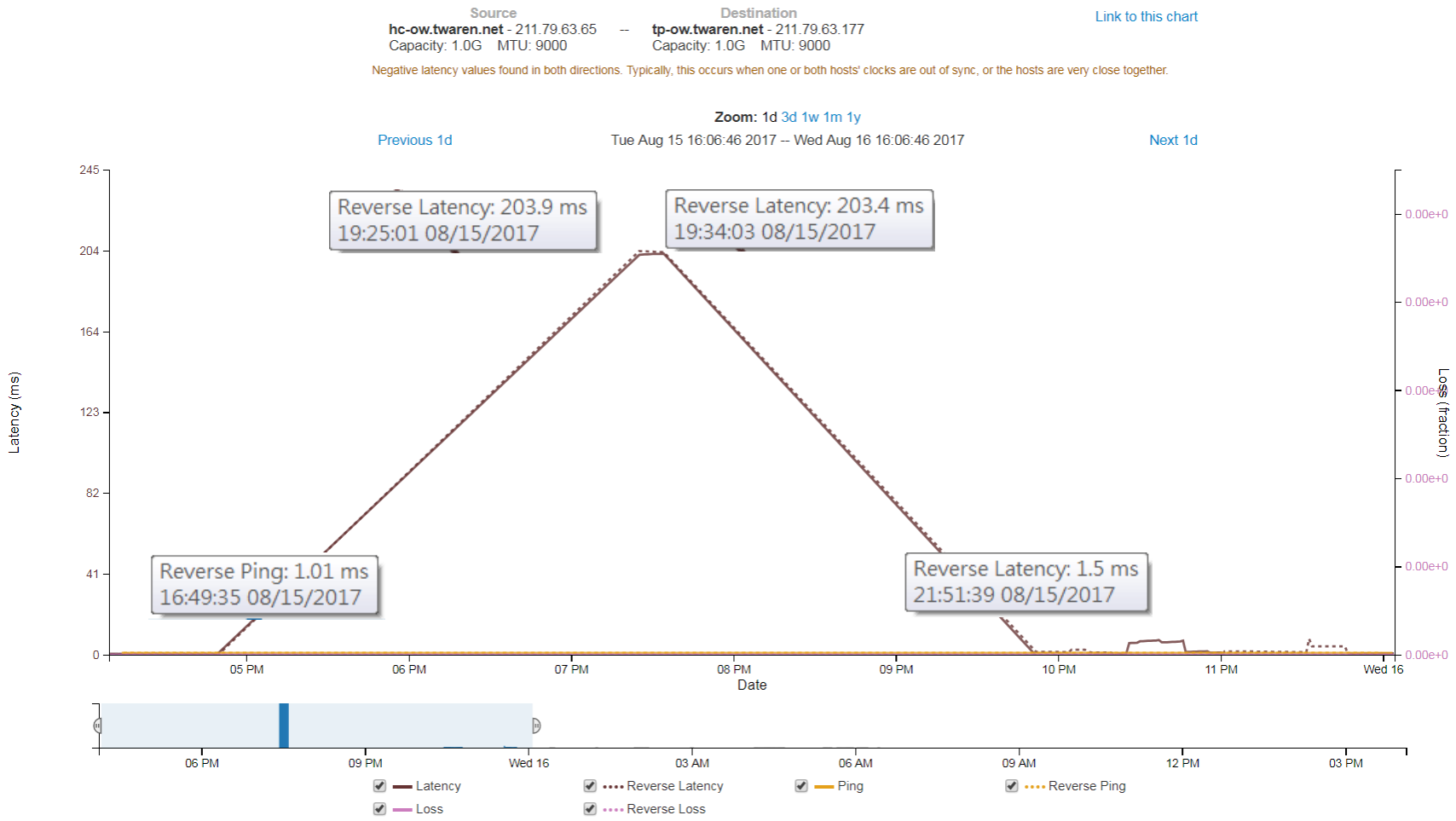
承諾．熱情．創新



- 2017年8月15日下午5點鐘開始，TWAREN 兩台perfSONAR Dashboard系統開始陸續出現無法擷取三峽網路效能量測主機之原始資料的告警。
- 三峽機房兩台網路效能量測監控伺服器同時關機。
- 主要網路設備，例如光網路設備，骨幹核心路由器以及交換器因為採用了直流電力系統，故暫時不受影響。

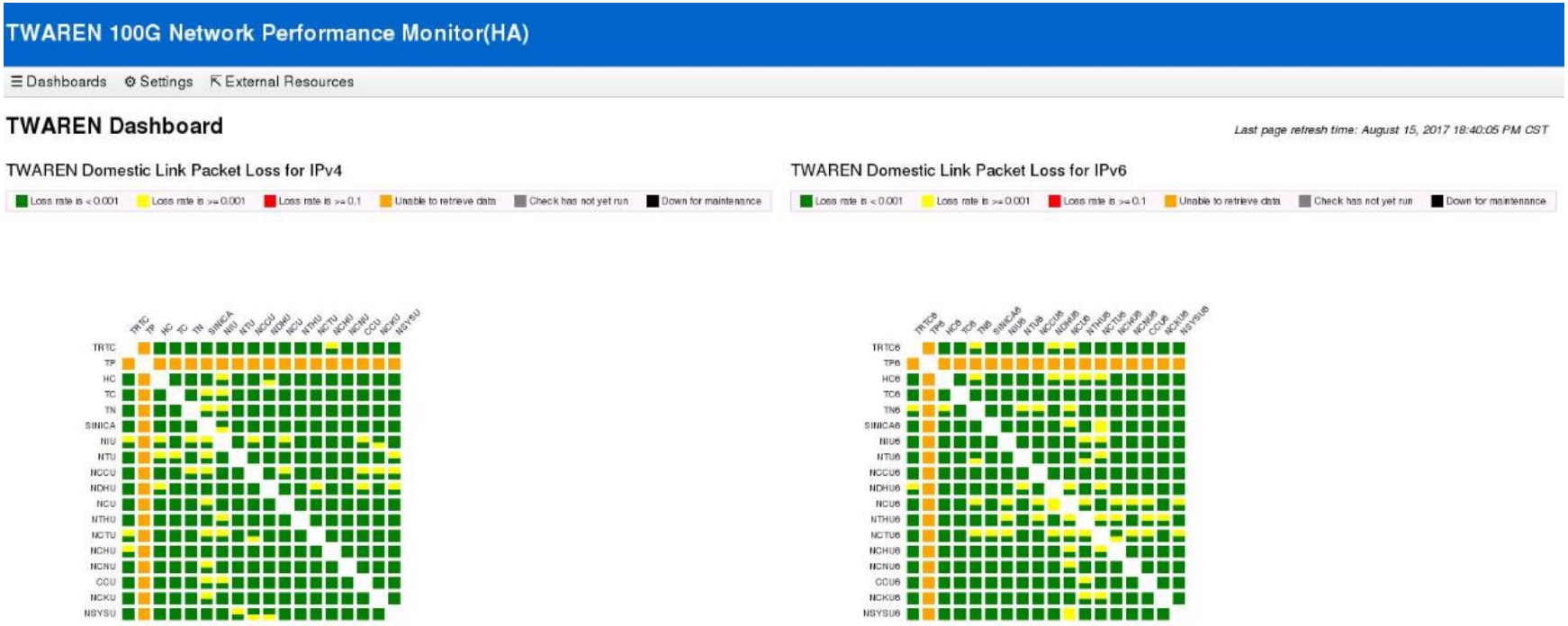
三峽主機監控畫面

承諾・熱情・創新



連接三峽之網路骨幹進行re-route

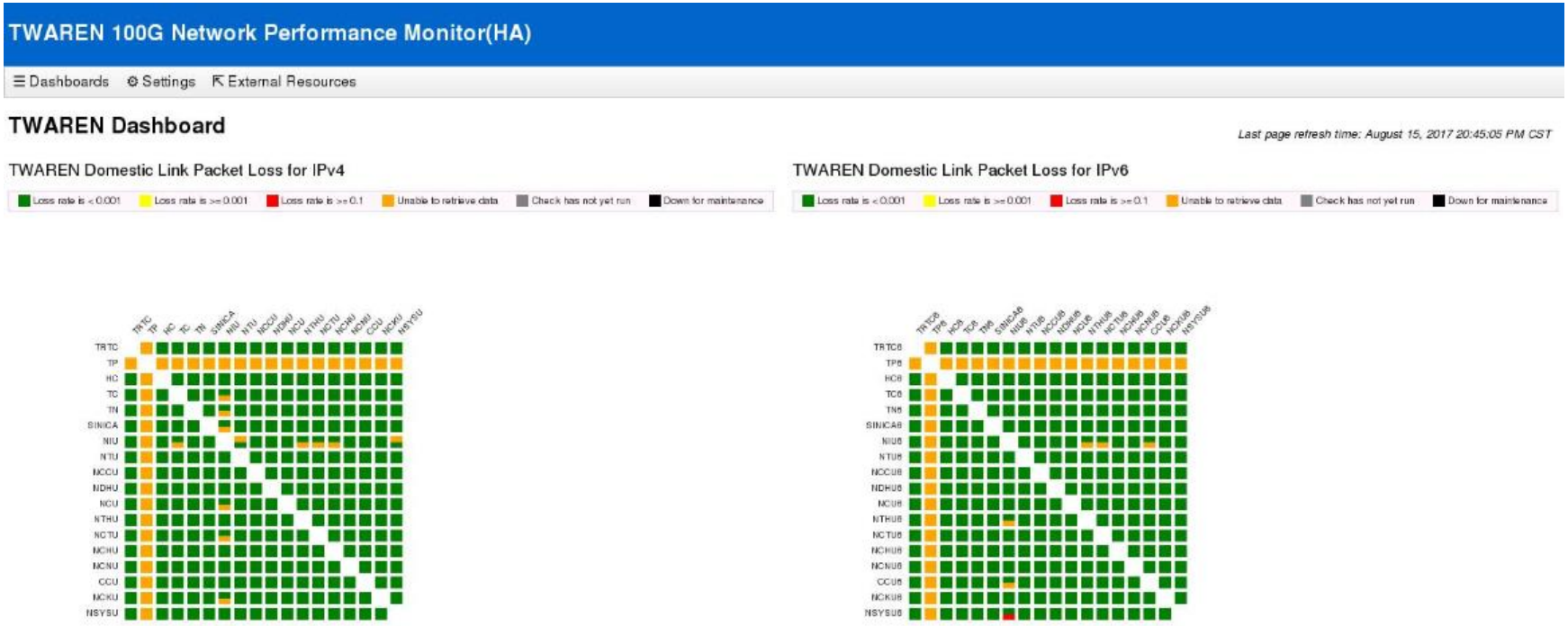
承諾・熱情・創新



- 下午5點54分左右，因三峽機房溫度超過設定之臨界值，為了避免設備損壞，TWAREN NOC開始針對網路設備進行預防性關機，多條原先與三峽主節點連接之國內骨幹網路重新選擇路由。 下午5點57分公告三峽網路設備預防性關機。
- 路由協定在切換路由的速度相當的快，一般小於50ms，因此使用者會感受不出網路切換之變異perfSONAR透過小而密集的封包偵測，可發覺網路細小之變化。

宜蘭大學網路效能監控伺服器停電

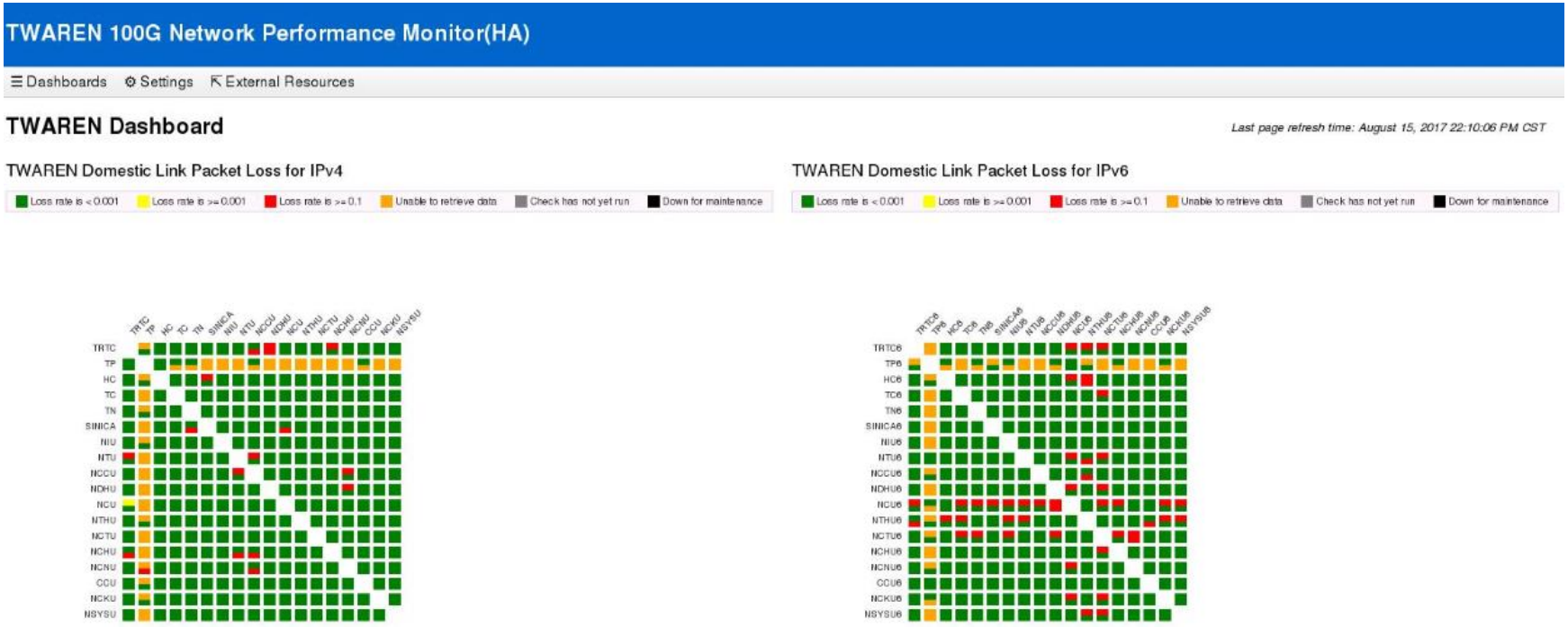
承諾・熱情・創新



- 晚間8點13分左右，Dashboard 系統發現宜蘭大學無法擷取資料，後來從宜蘭大學網路效能監控伺服器之日誌發現主要因素為停電關係，後來於晚間8點28分左右恢復電力。
- 宜蘭大學之TWAREN核心網路設備則不受影響。

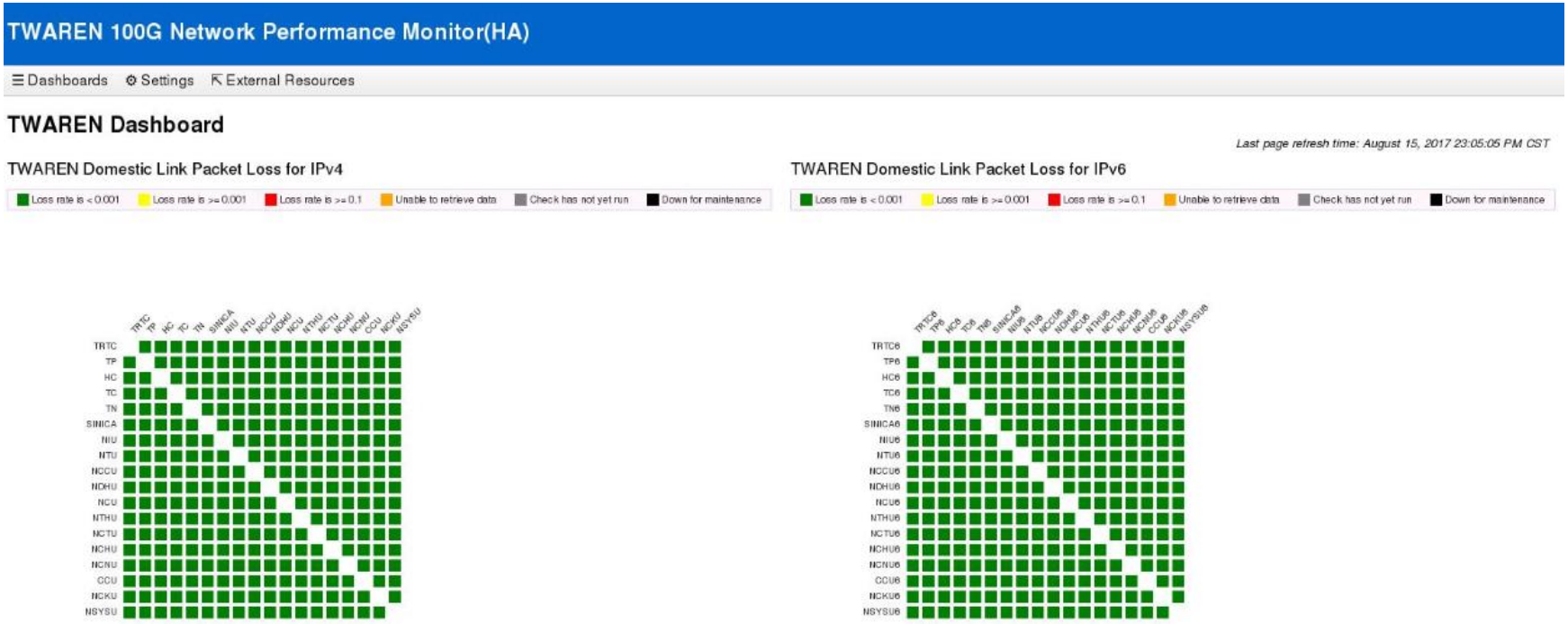
perfSONAR程式偵測到三峽網路接通

承諾・熱情・創新



- 晚間8點45分三峽光設備完成開機，晚間9點40分三峽路由設備完成開機。
- 晚間9點51分左右，perfSONAR程式偵測到網路接通，並開始產生監控數據。
- Dashboard也開始陸續收到三峽主機擷取之原始資料。
- 晚間10點15分，TWAREN NOC公告網路障礙恢復。

系統全部恢復正常



- 晚間11點左右，Dashboard全部恢復正常，顯示整體網路無任何網路封包遺失現象。

Science DMZ

其目的為如何讓科學資料能夠最佳化的在廣域網路上傳輸的設計方法。Science DMZ整合了三大重要關鍵因素，來達成快速傳遞資料之目的:

1. 適合於High-performance應用之專用網路，最好能與一般使用之網路區隔，避免防火牆，直接連接border router，減少其他網路設備發生錯誤時所帶來的影響，亦可減少除錯時之時間。
2. 擁有專屬的資料傳輸機器(群)，例如建置DTN(Data Transfer Node)並配合高效能的傳輸軟體及機器調教。
3. 良好的效能量測機制，可隨時量測點對點間之網路，以確保網路品質良好。

The Data Transfer Node (DTN)

- Dedicated, high-performance host for data transfer
- Typically PC-based Linux servers built with high-quality components and configured specifically for wide area data transfer
- Proper tools

DTN subsystems

- 儲存
  - Local storage(RAID, SSD)
  - Networked storage- Distributed file system(Infiniband, SAN)
- 網路
  - 10G以上之網路或專用網路
- 主機
  - Server bus(PCI-e)
  - File system(ext4, xfs, btrfs)
- Tuning
  - bios、CPU、IRQ、儲存、網路、檔案系統、應用軟體

NUMA Issues

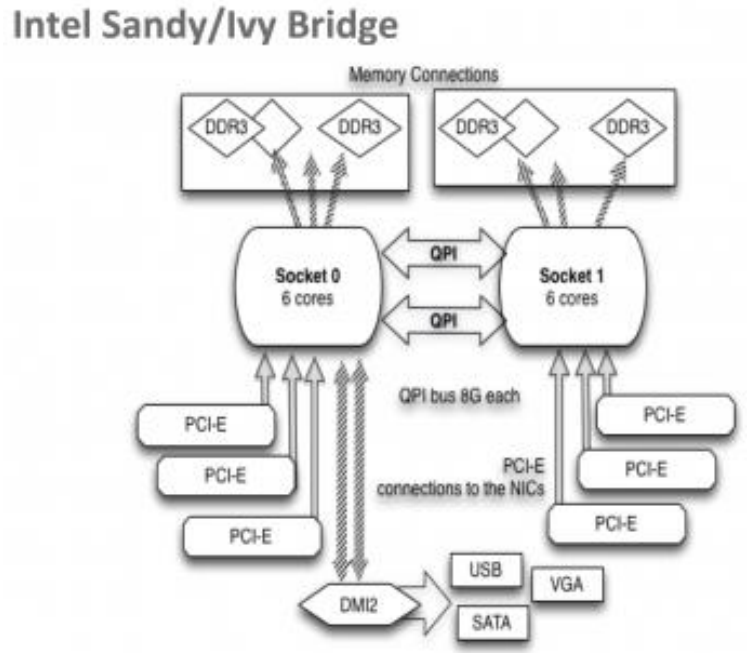
- Up to 2x performance difference if you use the wrong core.
- If you have a 2 CPU socket NUMA host, be sure to:
  - Turn off irqbalance
  - Figure out what socket your NIC is connected to:

    cat /sys/class/net/ethN/device/numa_node

  - Bind your program to the same CPU socket as the NIC:

    numactl -N 1 program_name
- Which cores belong to a NUMA socket?

  cat /sys/devices/system/node/node0/cpulist



DTN Storage

- 儲存種類
  - Local Storage(RAID, SSD): ex: NVME SSD
  - External Storage: Distributed file system(ex:Lustre檔案系統), lustre client mount
- 連接方式
  - DTN透過Ethernet、IB、omni-path連接後端DFS
  - NFS mount

Data Transfer Tools

- anonymous: anyone can access the data.
  ex: FTP HTTP(wget)
- simple password: most sites no longer allow this method since the password can be easily captured.
  ex: FTP HTTP(wget)
- password encrypted: control channel is encrypted, but data is unencrypted.
  ex: bbcp, bbftp, GridFTP, FDT
- everything encrypted: both control and data channels are encrypted.
  ex: scp, sftp, rsync over ssh, GridFTP, HTTPS-based web server

Secure Copy (SCP)

- **Secure Copy (SCP)**
  - Widely used for file transfers
  - Uses SSH for authentication and data transfer (TCP port 22)
  - Unix-based systems
  - Windows: WinSCP
- 若需使用scp或是rsync等傳輸軟體，可更新OpenSSH版本(例如hpn-ssh)

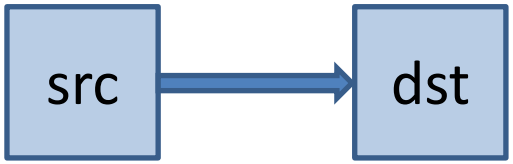
```
[sun1@tp-server1 ~]$ scp sun1@chi-server1:/home/100GB /home/sun1/worker/
100GB                                         0%  386MB   9.1MB/s 3:03:14 ETA
```

Data Transfer Tools

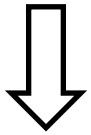
- Parallelism is key
  - It is much easier to achieve a given performance level with four
  - parallel connections than one connection
  - Several tools offer parallel transfers
- Latency interaction is critical
  - Wide area data transfers have much higher latency than LAN transfers
  - Many tools and protocols assume a LAN
  - Examples: SCP/SFTP

Single vs multi stream

- Single stream
  - scp
  - ftp
  - rsync



Better utilization of link



Faster transfer speed

- Multi stream(建議使用)
  - GridFTP
  - BBCP
  - FDT
  - Nuttcp
  - Aspera(商業軟體)

GridFTP

- GridFTP from ANL has features needed to fill the network pipe
  - Buffer Tuning
  - Parallel Streams
- Supports multiple authentication options
  - Anonymous
  - **ssh**
  - X509

```
[sun1@tp-server1 ~]$ globus-url-copy -vb  -p 10 -sync  \
sshftp://sun1@chi-server1/home/100GB   file:///dev/null
Source: sshftp://sun1@chi-server1/home/
Dest:   file:///dev/
  100GB  ->  null

  25728909312 bytes       766.78 MB/sec avg       995.20 MB/sec inst
```

BBCP & FDT

- BBCP
  - comparable performance to Globus
  - Mac OS X, Linux-based systems. SSH-based access control
  - Both endpoints need it installed, but easier to install and configure

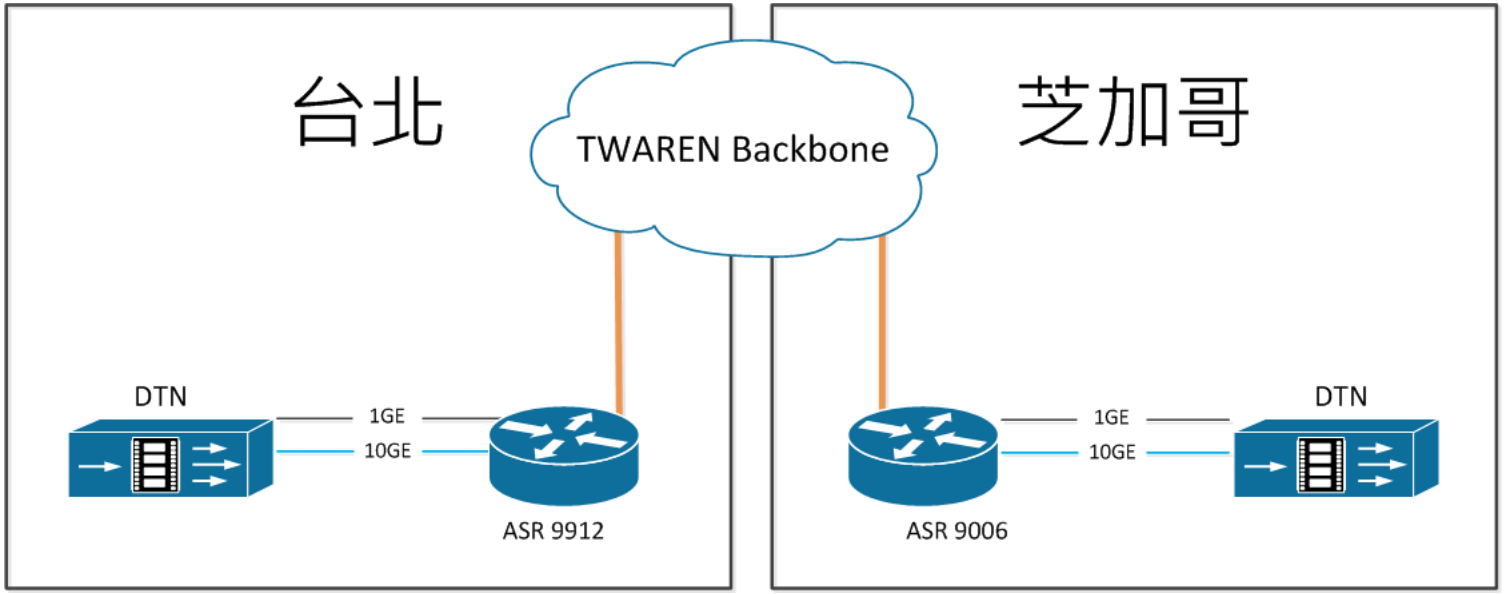
"$bbcp -V -s 16 /local/path/largefile.tar remotesystem:/remote/path/largefile.tar"

- Fast Data Transfer (FDT)
  - Java-based tool from Caltech & CERN
  - Can theoretically run in any Operating System, including Windows
  - Need server-side running in server mode

"$java -jar fdt.jar -c <remote_address> -d destinationDir ./local.data"

Demo

- 架構



- 從芝加哥節點傳輸100GB檔案至台北節點

  100GB to /dev/null (disk to ram)

  100GB to /worker/ (disk to disk)

Summary

- 為了優化TCP效能，我們可以：
  - 使用支援TCP buffer auto tuning的作業系統
  - 增加TCP auto tuning buffer size之最大值設定(例如16MB或是32MB以上)
  - 使用支援多重串流傳輸之傳輸軟體(例如GridFTP、BBCP、FDT)，或是同時傳輸多個單一串流之傳輸(同時多個scp或是多個wget)
  - 使用其他的congestion control algorithm(例如htcp、cubic)

Summary

- 為了得到較佳之傳輸效能，我們可以：
  - 使用多重(個)串流之方法
  - Large bulk transfers: 建置DTN並透過DTN傳輸
  - 確定可用頻寬是足夠的，或可建立傳輸專用頻寬
  - 確定TCP之相關參數已調整
  - 為了得到較好之傳輸效能，可規劃檔案傳輸DMZ架構
  - 考量磁碟I/O效能，可使用磁碟陣列或是GPFS、Lustre等平行分散檔案系統
  - 需注意網路品質，特別是RTT與packet loss
  - 長期監控效能，了解網路品質是否有異常(即時今天網路沒問題，也無法保證明日網路是好的)