# Tapestry Network Complexity Analyzer

## A Loom Application

Sandhya Narayan

[work in progress]

## Introduction

Tapestry is a new system which extracts the “network complexity” (NCI) from from an enterprise by examining the DNS requests made on that network. The data can be collected either hourly, weekly, monthly, annual and any other period based on configuration or selection.

Figure 1 shows the Tapestry UI with a graph of the “Network Complexity Indicator” over time, and a dashboard showing some performance numbers.

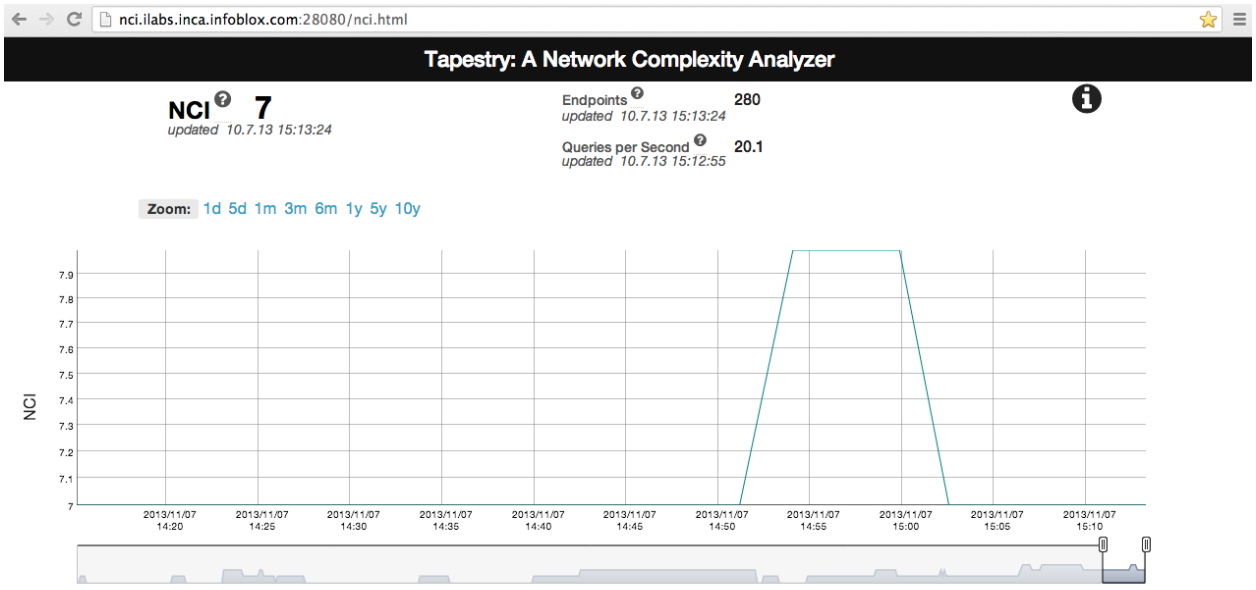



Figure 1

Tapestry is a SDN software application communicating with Network White Boxes (NWBs) which run LINC OpenFlow switch software and are in-line to one or more DNS servers and act as Tapestry Collectors. The collectors tap DNS responses being sent to the DNS clients, which are then sent to the Tapestry system (Figure 2) for analysis. Tapestry NCI provides instantaneous NCI values and a set of other indicators computed over selectable duration.

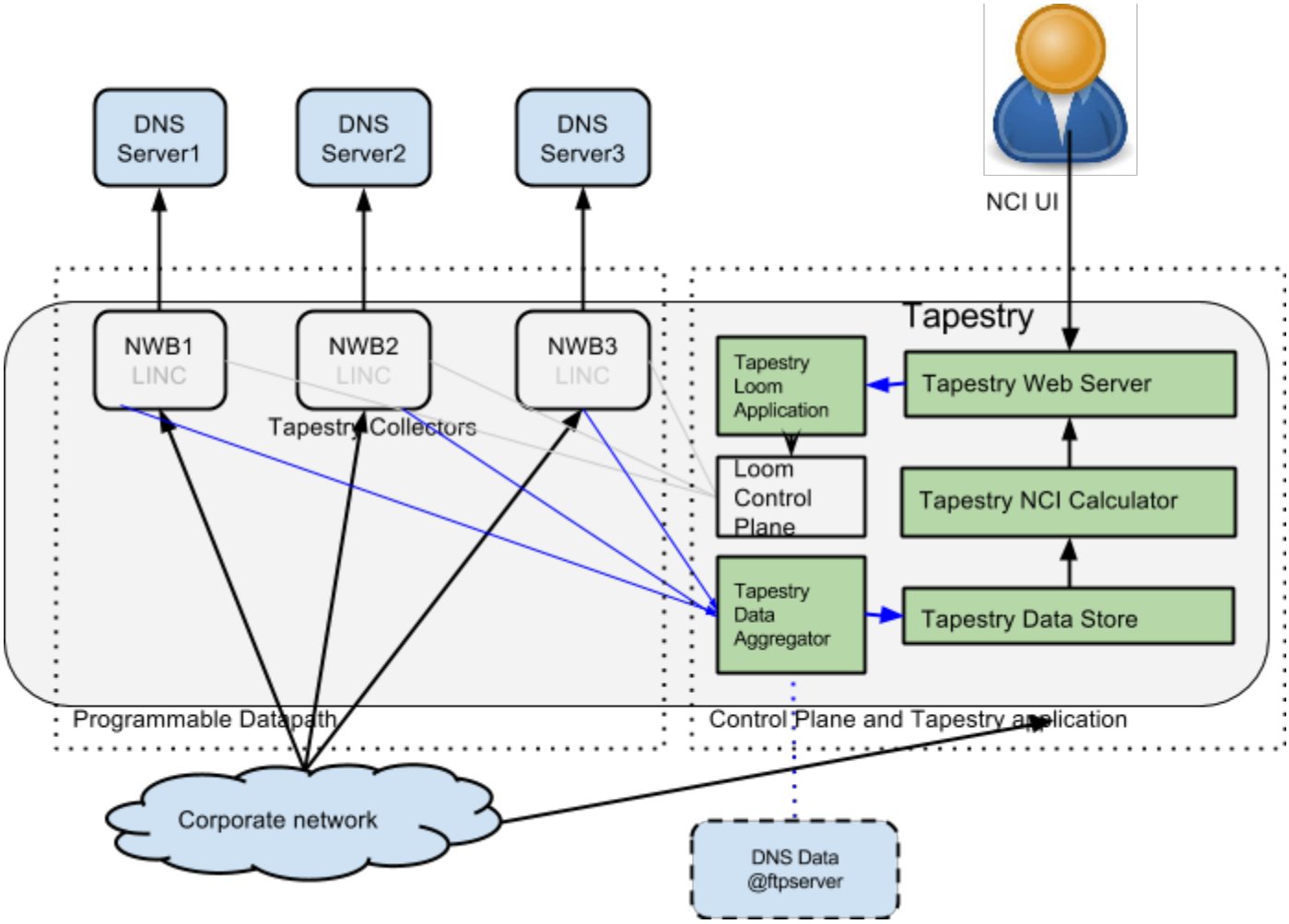



Figure 2

Tapestry can also use the DNS log data made available by InfoBlox DDI appliances. In this case, the Tapestry Data Aggregator accesses the DNS log data via FTP. The data is filtered and parsed to find the IP address of the client requesting DNS resolution and the IP address of the destination resolved by the DNS server.

## Tapestry Architecture

Figure 2 shows the components of the Tapestry system listed below and their interconnections.

- Tapestry Network Complexity UI
- Tapestry NCI Calculator
- Tapestry Web Server
- Tapestry Data Store
- Tapestry Data Aggregator
- Tapestry Collectors
- Tapestry Loom Application
- Loom Control Plane

## Tapestry Network Complexity UI

NetWork Complexity UI allows the user to select the data for analysis by specifying the following: fields of DNS data, range and type of DNS entries.

## Tapestry NCI Calculator

The details of the NCI algorithm are discussed in http://www.flowforwarding.org/nci-article. The input to the calculator consists of pairs: {DNS client IP address, resolved IP address}.

## Tapestry Web Server

A Web server that facilitates the users to view Network Complexity Index in real time or for any chosen period.

## Tapestry Data Aggregator

Tapestry Data Aggregator receives the DNS packets sent by all the switches, pre-processes them and stores them in Tapestry Data Store. In cases where DDI logs are available, it collects and pre-processes the logs before storing them.

## Tapestry Data Store

This stores the data collected and aggregated by the Tapestry Data Aggregator.

## Tapestry Collectors

Network White Boxes running LINC switch software are placed in-line on the path to DNS servers and serve as Tapestry Collectors. An example NWB with four network interfaces and running LINC is shown in Figure 2. Port 1 is connected to the Loom controller. Port 2 is connected to the legacy network that was previously connected to the DNS server and Port 3 is connected to the DNS server.

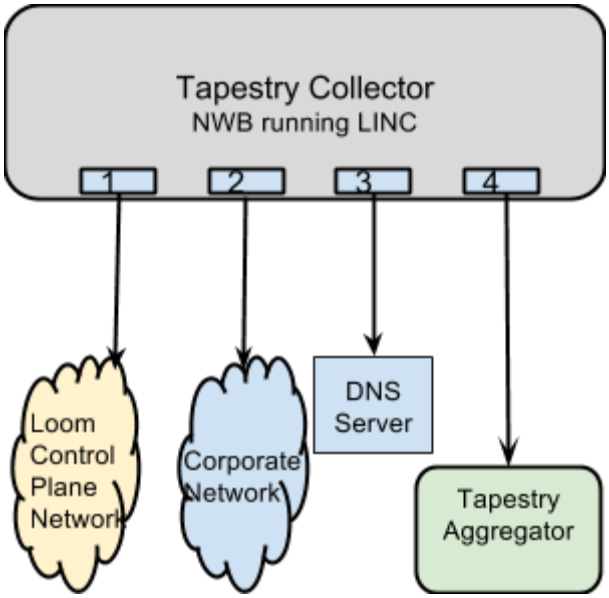
There are several ways to collect the DNS traffic:

1. Directly connect a cable between tap-port where all traffic is copied to the Aggregator.
2. Send “matched” packets as Packet-in messages to the OpenFlow controller, which sends it to the Aggregator.
3. Use a flow-entry to change the destination IP address and port of the “tapped-packets” to that of the Aggregator IP address and port and output it to the legacy network.
4. Same as (2) but make use of “In-Band” Controller.

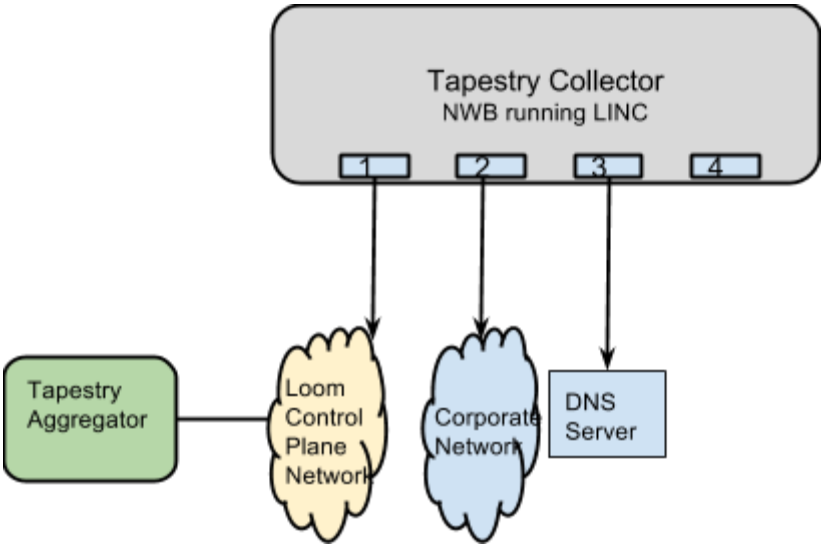
These options are discussed below:

### Direct Connection

In the figure below, Port 4 of the switch is the tap port and is connected directly to the Aggregator. This configuration is the simplest to implement, but has deployment issues requiring new wiring to connect tap port to the Aggregator.




### Use of PacketIn



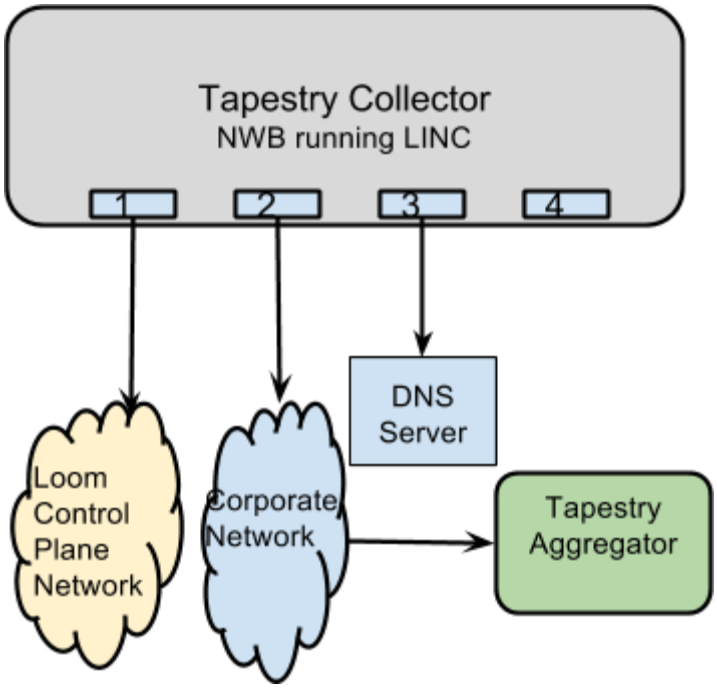

In this configuration the Aggregator is set up as a controller for the LINC switches. This method uses the “Packet-in on action” feature of OpenFlow. A switch sends Packet_in messages to the controller when there is a table-miss flow entry or the action-output port of a flow-entry is set as “controller”.

The advantage of this method is that a Packet-in message encapsulates the “matched” packet and so the entire packet can be captured at the Aggregator.  It also has the obvious advantage of

not needing any extra cabling to connect the tap port to the Aggregator.

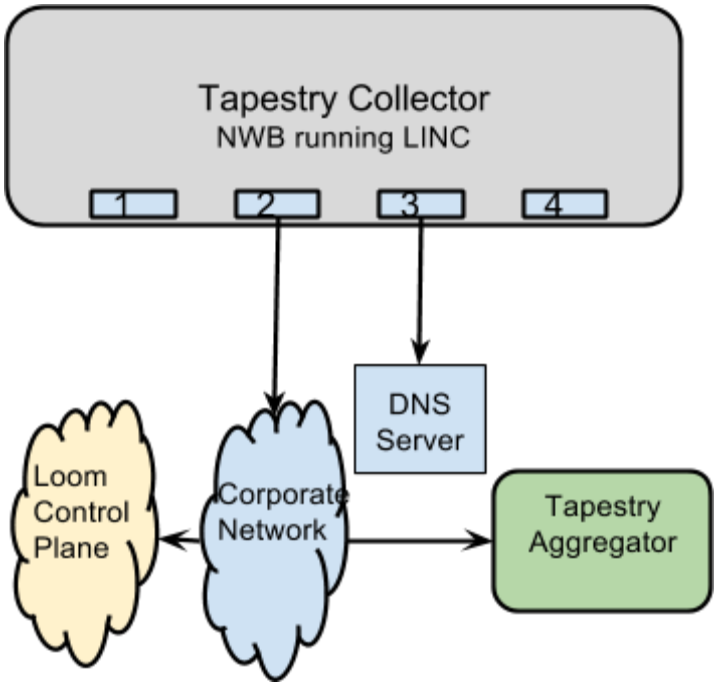
### Change Destination IP of the “tapped” packet

This method sends the “tapped” packet to the Aggregator via the Legacy network by rewriting the destination IP address in the header and outputting the packet to the port connected to the Legacy Network. It has the obvious advantage of not needing any extra cabling to connect the tap port to the Aggregator as in the direct connection method. But it loses the actual destination IP address and port. It also uses udp because original packet is a udp packet, and hence may not be reliable tap.




### Packet-In message with In-Band Controller

This method uses a packet-in message to send the “tapped” packet to a controller which is the Aggregator. It has the advantage of the packet-in method where all headers are preserved. It also has the advantage of using the legacy network to gather the tapped traffic from the collector and not needing extra cabling to collect the tapped traffic. It also has the advantage of using TCP to send the “tapped” packet, because OpenFlow is being used here, and OpenFlow uses TCP, and thus not losing by using the original udp traffic as is.




### Tapestry Loom Application

Tapestry Loom application prepares hand-coded flow entries for all the LINC OpenFlow switches and uses the Loom Control Plane to send them to the NWB OpenFlow switches.

Flow entries for direct connection:

- Higher priority: Forward UDP traffic from port 3 to port 2 and port 4. (Tap DNS responses).
- Lower priority: Forward traffic from port 2 to port 3
- Lower priority: Forward traffic from port 3 to port 2

Flow entries for Packet-In:

- Higher priority: Forward UDP traffic from port 3 to port 2 and controller. (Tap DNS responses).
- Lower priority: Forward traffic from port 2 to port 3
- Lower priority: Forward traffic from port 3 to port 2

## Tapestry Requirements:

### Datapath

- Programmable Ethernet Datapath supporting OpenFlow 1.3
- Depends on the environment of deployment.

Some example hardware requirements:

- Low cost low power NWB
    - Intel Atom N450 1.66GHz based desktop platform with 4GbE: http://www.portwell.com/products/detail.asp?CUSTCHAR1=CAD-0205-06-08
- Medium cost medium power NWB
- Soft switches on any x86 server

## Loom Control Plane

- Depends on the environment of deployment.
- x86 running Linux or Windows

## Tapestry Application

- Depends on the environment of deployment.
- x86  running Linux or Windows

## Glossary

Network White Box (NWB): An X86 based system with at least 3  network interfaces, preferably with more cores than interfaces.

LINC: A software OpenFlow switch supporting OpenFlow 1.2 and 1.3.1 specifications written in Erlang.

Loom: A distributed controller platform for SDN supporting OpenFlow 1.0, 1.2 and 1.3 specifications.