YOUNG CONAWAY STARGATT & TAYLOR, LLP

THE BRANDYWINE BUILDING
1000 WEST STREET, 17TH FLOOR
WILMINGTON, DELAWARE 19801

P.O. BOX 391
WILMINGTON, DELAWARE 19899-0391

JOSY W. INGERSOLL (NO. 1088)
DIRECT DIAL: 302-571-6672
DIRECT FAX: 302-576-3301
jingersoll@ycst.com

(302) 571-6600
(302) 571-1253 FAX
(800) 253-2234 (DE ONLY)
www.youngconaway.com

August 22, 2006

REDACTED – PUBLIC VERSION

BY HAND DELIVERY **CONFIDENTIAL – FILED UNDER SEAL**

The Honorable Sue L. Robinson
United States District Court
844 North King Street
Wilmington, DE 19801

Re: *Alcatel USA Resources et al. v. Foundry Networks, Inc.*,
Civil Action No. 05-418 (SLR)

Dear Chief Judge Robinson:

This letter is provided at the Court's direction by Alcatel USA Resources, Inc. ("Alcatel") to show Alcatel's good faith basis for seeking hardware discovery on certain of Foundry Networks' products which Alcatel contends infringe. At the August 8 hearing, which followed an August 1 teleconference, the Court requested a sample presentation showing that good faith basis for ten products.[1] Our understanding is that if Alcatel makes the minimum threshold showing to justify discovery, then Foundry Networks will be required to supplement its discovery as to all so-far withheld products.

This presentation addresses the '192 and '840 accused-products (hardware patents). We have selected the BigIron-RX series, the NetIron MLX/XMR Series and the EdgeIron series. We make our showing below on a series-basis. These series in total encompass at least the ten model numbers.

This presentation (Section 3, below) also points out that Foundry Counsel's statements to the Court that the BigIron-RX series has "different" hardware from Terathon were simply not true. We believe that this calls into question Foundry's credibility as to all of its alleged reasons for obstructing threshold discovery on the products Alcatel believes infringe its patents.

Our presentation is organized as follows:

1. We will explain our infringement claim charts which cited a key Foundry document -- the JetCore Architecture Brief -- to set the baseline for what functionality even Foundry concedes makes a model discoverable.

[1] Transcripts of the August 1 teleconference and August 8 hearing are attached hereto as Exhibits A and B, respectively.

2. We will show evidence demonstrating the functionality which causes the infringement to exist identically in the product lines withheld by Foundry.

There is more. To further support the accuracy of our good faith conclusions, we provide additional important information: that the withheld BigIron products are indeed Terathon, and hence discoverable by Foundry's own admissions. Foundry's representation to the Court that they are not Terathon (Ex. B, August 9 Hearing Transcript at 16-17, 28) is not true.

1. THE DETAILED INFRINGEMENT CLAIM CHARTS

A. Detailed discussion of '192 infringement

To show infringement of the '192 patent in its claim chart, Alcatel cited extensively to Foundry's JetCore Architecture Brief. The '192 claim chart is attached hereto as Exhibit C.

The first element of claim 1 is for "a buffer memory storing data structured as fixed or variable length blocks of data, each of said data blocks comprising one or more data elements." To show the presence of this element, Alcatel pointed to this language in the Foundry document: "The shared memory buffer pool provides storage for the received packet(s) till it is time to the send the packet out ...." At FN A 00163059. Hence, the Foundry share memory buffer pool feature corresponds to the patent's claim element as follows:

Buffer memory ('192) → Shared memory buffer pool (Foundry product)

The second element of claim 1 is for "a data block and outlet selection queue management logic comprising." To show the presence of this element, Alcatel pointed to this language in the Foundry document: "The buffer manager is responsible for managing the packet storage space inside the shared memory." Id. "Under the control of the buffer manager, the shared memory switch fabric, also, retrieves packets from the shared memory and sends them to their destination ports, as instructed by the buffer manager." At FN A 00163059. Hence, the Foundry feature corresponds to the claim element as follows:

Data block and outlet selection queue management logic ('192) → Buffer manager (Foundry)

The third element of claim 1 is for "a plurality of queues for interlinking each of the data blocks to a respective destination of a plurality of destinations, such that when a data block is assigned to a destination it is sufficient to write its identity in the respective queue specific to this destination." To show the presence of this element, Alcatel pointed to this language in the Foundry document:

> "When the buffer manager is notified of the buffer number of a new packet in the shared memory, the buffer manager assigns the buffer number to the appropriate queue based on the packet's destination and priority." Id. "Foundry's architecture offers 4 queues per destination to allow up to 4 levels of traffic prioritization. The architecture honors those priority levels, and schedules service for the prioritized traffic flows through the use of:
>
> - 4 output queues per physical port within the same port group
> - 4 output queues per neighboring port group
> - 4 output queues per neighboring interface module" At FN A 00163062 and FN A 00163063

Alcatel further noted that in the Foundry device, a data block's identity is its buffer number. Hence, the Foundry features correspond to the claim element as follows:

> **Plurality of queues ('192) → Output queues (Foundry)**
> **Data block identity ('192) → Buffer number (Foundry)**

The fourth element of claim 1 describes a "data block selection means for assigning each of said data blocks to a respective individual outlet of a respective plurality of outlets associated with said respective destination, said data block selection means further comprising." To show the presence of this element, Alcatel pointed to this language in the Foundry document: "When a packet reaches the head of its queue, the buffer manager instructs the shared memory switch fabric as to where to retrieve the packet from the shared memory (using the buffer number), and which physical port to send it to." At FN A 00163059. Alcatel further noted that in the Foundry device, a data block's destination is its destination port. Hence, the Foundry features correspond to the claim element as follows:

> **Data block selection means ('192) → Buffer manager (Foundry)**
> **Destination ('192) → Physical port (Foundry)**

The fifth element of claim 1 is for (as a sub-element of the data block selection means) "first means for identifying a specific said queue associated with said respective destination with which said respective individual outlet is associated ". To show the presence of this element, Alcatel pointed to this language in the Foundry document:

> "The fabric … reports the following information to the buffer manager:
> Buffer number - where the packet is stored
> Forwarding Identifier (FID) …."
> "The buffer manager uses the above information to determine the location of the packet in the shared memory, … the destination(s) of the packet, and consequently, which queue(s) it … [is] placed on." At FN A 00163058

Alcatel further noted that in the Foundry device, the CAM & PRAM translate a destination port which implies a specific outlet and its associated queue:

> "The content addressable memory (CAM) is used by the packet classifier to provide reverse lookups based on the content of the packet. The packet classifier extracts a portion(s) of the packet headers - such as destination IP address, destination MAC address, source and destination IP addresses, etc. - and hands it over to the high speed CAM to perform a lookup. In case a match is found, the CAM returns an index of a location in the PRAM, where information on how to forward the packet is stored.
>
> The PRAM is a high speed memory that takes the index returned by the CAM lookup and returns to the packet classifier information on how to forward the packet: FID (destination port or ports), MAC address of next hop, VLAN ID that should be attached to the packet as it goes out a given interface, etc. The information retrieved by the PRAM is used to build the internal Foundry header that gets attached to the packet. This information is used by other modules of the hardware to determine how to forward the packet." At FN A 00163060

Hence, the Foundry feature corresponds to the claim element as follows:

First means ('192) → Buffer manager (using CAM, PRAM, FID, etc.) (Foundry)

The sixth element of claim 1 (also a sub-element of the data block selection means) is for "second means for obtaining from said specific queue an address of said each data block". To show the presence of this element, Alcatel pointed to this language in the Foundry document: "When a packet reaches the head of its queue, the buffer manager instructs the shared memory switch fabric as to where to retrieve the packet from the shared memory (using the buffer number),..." At FN A 00163059. Hence, the Foundry feature corresponds to the claim element as follows:

Second means ('192) → Buffer manager (Foundry)

Finally, the seventh and final element of claim 1 (also a sub-element of the data block selection means) is for "third means for transferring said each data block to said respective individual outlet." To show the presence of this element, Alcatel pointed to this language in the Foundry document: "Under the control of the buffer manager, the shared memory switch fabric, also, retrieves packets from the shared memory and sends them to their destination ports, as instructed by the buffer manager." At FN A 00163059. Alcatel noted that the individual outlet is the hardware associated with the chosen queue. Hence, the Foundry features corresponds to the claim element as follows:

Third means ('192) → Buffer manager (Foundry)
Individual outlet ('192) → Hardware for chosen queue (Foundry)

To summarize, this is how Alcatel mapped the '192 claim elements onto the JetCore architecture for purposes of its initial showing of discoverability (obviously this is not Alcatel's final proof of infringement, which will be provided at the appropriate time in the case):

Buffer memory ('192) → Shared memory buffer pool (Foundry)
DB/outlet selection queue management logic ('192) → Buffer manager (Foundry)
Plurality of queues ('192) → Output queues (Foundry)
Data block identity ('192) → Buffer number (Foundry)
Data block selection means ('192) → Buffer manager (Foundry)
Destination ('192) → Physical port (Foundry)
First means ('192) → Buffer manager (using CAM, PRAM, FID, etc.) (Foundry)
Second means ('192) → Buffer manager (Foundry)
Third means ('192) → Buffer manager (Foundry)
Individual outlet ('192) → Hardware for chosen queue (Foundry)

B. Detailed discussion of '840 infringement

To show infringement of the '840 patent, Alcatel also quoted extensively from the JetCore Architecture Brief, plus certain documents reflecting that Foundry devices support two specific features: "VLAN flooding" and "trunk groups." The '840 claim chart is attached hereto as Exhibit D.

The first element of claim 8 (the only asserted independent claim) is for "Routing logic means for a switching element (S11, ..., S3N) having at least one inlet and a plurality of outgoing routes each comprising at least one outlet, said routing logic means (MC1; MC2; RL) selecting for cells received at said at least one inlet at least one of said plurality of outgoing routes as a selected route based on an internal routing label associated with said cells and routing said cells to said at least one outlet of said selected route." To show the presence of this element, Alcatel pointed to this language in the Foundry documents:

> Foundry document Foundry Switch and Router Installation and Basic Configuration Guide (2004) ("Configuration Guide"), "Foundry devices support ... 802.1p/q VLAN Tagging." C-1. Per IEEE 802.1Q (1998), "In port-based VLAN classification within a Bridge, the VID associated with an untagged or priority-tagged frame (i.e. a frame with no tag header, or a frame with a tag header that contains the null VLAN ID), is determined, based on the Port of arrival of the frame into the Bridge ...." p.31. Per Foundry document Switch and Router Command Line Interface Reference (2004) ("CLI Reference"), "When you enable hardware flooding for a VLAN, unknown unicast packets are flooded by hardware to all ports in the VLAN." At FN A 00080263.

> The PRAM is a high speed memory that takes the index returned by the CAM lookup and returns to the packet classifier information on how to forward the packet: FID (destination port or ports), MAC address of next hop, VLAN ID that should be attached to the packet as it goes out a given interface, etc. The information retrieved by the PRAM is used to build the internal Foundry header that gets attached to the packet. This information is used by other modules of the hardware to determine how to forward the packet. From White Paper "JetCore Based Chassis Systems – An Architecture Brief ...." At FN A 00163060

Alcatel also noted that the internal routing label is the internal Foundry header as mentioned above. Hence, the Foundry feature corresponds to the claim element as follows:

Routing logic means ('840) → Switching logic for hardware flooding (Foundry)
Internal routing label ('840) → Internal Foundry header (Foundry)

The second element of claim 8 is for "wherein said routing logic means (MC1; MC2; RL) includes preselection means (MB; M; RM) which preselect under control of said internal routing label, at least one preselected set having at least one of said plurality of outgoing routes." To show the presence of this element, Alcatel pointed to this language in the Foundry documents: "A Layer 2 entry is made in the Foundry device's CAM that matches all packets for that VLAN. Any traffic for the VLAN that doesn't match a CAM entry for an individual destination matches this CAM entry." At FN A 00080263. Per Foundry document Foundry Security Guide (2004), "CAM entries contain Layer 2, Layer 3, or Layer 4 next-hop information .... Traffic that matches the match-all CAM entry is hardware-flooded to all ports in the VLAN ...." At FN A 00133095. Alcatel also noted that the internal Foundry header therefore allows pre-selection for such a frame of all outgoing ports in the inlet port's VLAN. Hence, the Foundry feature corresponds to the claim element as follows:

Preselection means ('840) → Logic placing all-packet entry into CAM (Foundry)

The third and final element of claim 8 is for "selection means (MX; MX'; SEL) to select said selected route from said at least one preselected set." To show the presence of this element, Alcatel pointed to this language in the Foundry documents: Per Configuration Guide, "A trunk group is a set of physical ports that are configured to act as a single physical interface. Each trunk group's port configuration is based on the configuration of the lead port, which is the lowest numbered port in the group. If you add a trunk group's lead port to a VLAN, all of the ports in the trunk group become members of that VLAN." At FN A 00135747. "For switch trunk groups, load sharing occurs as follows: .... To select a port, the device determines the port in the trunk group that has the fewest number of flows. Traffic for the destination address is sent over the selected port." At FN A 00135639. Alcatel also noted that when a trunk group is part of a flooded VLAN, the device selects the outgoing port from among its pre-selected member ports. Hence, the Foundry feature corresponds to the claim element as follows:

Selection means ('840) → Logic that sends over selected trunk group port (Foundry)

To summarize, this is how Alcatel mapped the '840 claim elements onto the JetCore architecture, in the case of VLAN flooding into a trunk group:

Routing logic means ('840) → Switching logic for hardware flooding (Foundry)
Internal routing label ('840) → Internal Foundry header (Foundry)
Preselection means ('840) → Logic placing all-packet entry into CAM (Foundry)
Selection means ('840) → Logic that sends over selected trunk group port (Foundry)

2. EVIDENCE THE WITHHELD PRODUCTS INFRINGE

Foundry attempts to hold Alcatel to an impossible standard -- "show us detailed reasons why we infringe before we will give you documents to let you show us your detailed reasons." The Court aptly identified this as a "chicken-and-the-egg" problem (Ex. B at 27), and told Foundry counsel that it has "never seen a claim chart that is that detailed at this stage of the game," even by the time people get to trial (Ex. B at 33).

As explained at the August 8 hearing, the numerous additional Foundry documents Alcatel cited in its supplemental interrogatory answer also show that the withheld products have the same functionality named in the detailed infringement claim charts. Here are specific point-citations showing this to be true:

BigIron-RX series as described and highlighted in Foundry document beginning at FN A_382797 (excerpts attached as Exhibit E)

Buffer: FN A_382818 (showing "same feature set" as Terathon products)
Queues: FN A_382818 (showing "same feature set" as Terathon products)
VLANs: FN A_382818 (showing "same feature set" as Terathon products)
Trunk Groups: FN A_382818 (showing "same feature set" as Terathon products)

NetIron MLX/XMR series as described and highlighted in Foundry document beginning at FN A_395966 (excerpts attached as Exhibit F)

Buffer Manager: FN A_396102
Queues: FN A_396299
VLAN floods: FN A_396169
Trunk Groups: FN A_395972

EdgeIron as described and highlighted in Foundry document beginning at FN A_355006 (excerpts attached as Exhibit G)

Buffers: FN A_355012
Queues: FN A_355013

VLANs: FN A_355032
Trunk Groups: FN A_355032

In addition, Foundry recently published on its web site the BigIron RX Architecture Brief, and the NetIron XMR Architecture Brief. Foundry did **not** produce these documents in discovery. As highlighted in respective Exhibits H and I, these documents show the **very same** infringing functionality. Alcatel has highlighted those documents to point to these features.

There is more. There is only one software family running all of Foundry's different products. Foundry calls it IronWare. The same software family is used to control the hardware of ALL Foundry architectures. When the same software offering the same features operates both disclosed and non-disclosed hardware architectures, we may reasonably infer that the disclosed and non-disclosed architectures are substantially the same.

Finally, another factor is worth noting, although not dispositive. The same product designations (e.g., BigIron, NetIron) apply to some models which Foundry alleges contain different architectures. The fact that the same product name is used for two allegedly separately classified devices provides a reasonable basis (at least for discovery purposes) to "look under the hood" to see if the hardware is in fact different.

3. FOUNDRY SEEKS TO MISLEAD: BIGIRON-RX IS TERATHON-BASED

Without citing any evidence or providing any specifics, Foundry has insisted the withheld discovery relates to architectures that are new and different, and are not Terathon based products. Foundry has not been candid. The proof is contained in its own documents:

A Foundry marketing document describes the BigIron-RX series as including "the powerful Terathon ASIC-based architecture." (Ex. J). Other Foundry internal documents describe the BigIron-RX having the "same feature set" (with minor exceptions) as the "BigIron MG8" and "NetIron 40G". (Ex. E, Field Training Guide at FN_A 382797, esp. 382818). The same document says the MG8 and 40G are Terathon-based products. (Id. at 382827-28, wherein MG8 and 40G are the subject of the "Terathon Update" slideshow). Both the BigIron-RX and the MG8/40G use "Software release 2.2.0." (Id.). Thus Foundry documents reveal that the BigIron-RX is Terathon-based.

At the August 8 hearing, Foundry counsel told this Court that the BigIron-RX series is not Terathon-based:

> Mr. Poppe: There are ***three different architectures*** which, for each architecture, there's a group of products that share that particular architecture, which have been discussed in the past before your Honor. There's the IronCore architecture, which was the second phase of products that Foundry introduced back in the 90's, the JetCore architecture, which was the next generation, and the Terathon architecture.

> Alcatel has presented its arguments as if those three architectures apply to all of the accused Foundry products, but ***that is not the case***.
>
> There is an AccessIron series of products with its own architecture, an EdgeIron series of products with its own architecture. ***The BigIron RX has a different architecture***, and the FastIron Edge Switch, FastIron Edge Switch X series, et cetera, et cetera. And I've provided a list of these.
>
> And not only do they have a different architecture, but ***we specified that they specifically lack hardware features that have been listed in these interrogatory responses as being features that Alcatel is relying on for their infringement claims***.

(Ex. B, Transcript at 16-17, emphasis added). Counsel went on to say that "the BigIron RX has completely different chips, computer chips in it than the ones that are in JetCore, IronCore and Terathon." (Id. at 28).

As shown above, this is untrue. The BigIron-RX ***is in fact a Terathon product***.

Given these proofs, we suggest that the Court weigh carefully Foundry's statements that architectures are new and different.

CONCLUSION

While Foundry has repeatedly told the Court that the withheld products relate to a "different" architecture, Foundry never once tells the Court (or Alcatel) why or how it is different. Meanwhile, Alcatel's presentation above proves beyond any doubt that the infringing functionality crosses architectures, crosses product designations, and is a common thread within virtually all of Foundry's product lines.

Alcatel has shown far more than the minimum of good faith to support its infringement contentions. See, e.g., Andco Environmental Processes, Inc. v. Niagara Environmental Associates, Inc., 215 USPQ (BNA) 224, 226 n.3 (W.D.N.Y. 1982) ("Plaintiff is not required to prove its case on the pleadings. Certainly, the fact that it has found three instances of potential infringement entitles it to determine ***whether other such instances exist***.") (emphasis added); Rohm and Haas Co. v. Brotech Corp., 1990 U.S. Dist. LEXIS 20117 at *20 (D. Del. 1990) ("[A] defendant cannot simply claim noninfringement and limit discovery according to its view of the case."). See also, The Chamberlain Group, Inc. v. Interlogix, Inc., 2002 U.S. Dist. LEXIS 3046 at *4-5 (N.D. Ill. 2002) (permitting accused product scope discovery). Notwithstanding that Foundry has withheld discovery, Alcatel has shown a good faith basis to seek discovery, and gather additional evidence of infringement for trial. Alcatel respectfully requests the Court to

order Foundry to produce the withheld discovery -- both technical and financial -- so Alcatel may complete its preparations for trial on the full range of infringing devices.

Respectfully submitted,

Josy W. Ingersoll (No. 1088)

JWI:cg
cc: Clerk of the Court (by e-filing and hand delivery)
Philip A. Rovner, Esquire (by e-filing and hand delivery)
Matthew H. Poppe, Esquire (by e-mail)
Henry C. Bunsow, Esquire (by e-mail)
Robert P. Greenspoon, Esquire (by e-mail)