

## **COMPLEX TUNNEL RESOLUTION FOR TODAY'S CLASSIFICATION PROBLEMS**

Imagine what will happen when every teenager in the industrialized world gets a cell phone. And imagine what will happen when those cell phones can handle text messages as well as voice. The result: instant messages go wireless.

Add to that the mobile real estate agents, traveling business executives, away-from-home college students—all using wireless PDAs for email and wireless PCs for net surfing. With all this activity, the market for wireless base stations should make every networking executive drool. Welcome to the world of wireless data.

As networking professionals, the question we must ask ourselves is: What will these wireless base stations, and the networks they connect to, look like? How will the plethora of standards, protocols, and proprietary systems all interconnect seamlessly and economically? How will they handle exponential increases in packet traffic coupled with ever-increasing complexity? And how can we accomplish this on our existing infrastructure?

The predominant school of thought envisions that these wireless devices will be IPv6 enabled, thus providing more than enough address space for everyone. Though IPv6 may well solve the issue of addressing, it is highly unlikely that ubiquitous IPv6 networks will soon be built end to end.

The likely solution will be to tunnel IPv6 traffic over existing IPv4 networks and emerging MPLS networks. Clearly, a significant problem that we must solve to use existing networks will be the ability to quickly and efficiently resolve deep complex tunnels. However, tunnel resolution places severe processing burdens on edge routers in the form of multi-pass packet header classification and forwarding.

When it comes to handling Internet traffic, two separate classification problems have emerged: Layer 2-4 multi-pass header classification, and upper layer “deep packet classification.” Deep packet classification commonly referred to as Layer 7 is an increasingly important technology that has received much recent attention, but it is not the technology to solve the problems associated with complex tunnel resolution. In reality, most deep packet switching is done on payload data content, which is deeper into the packet than the Layer 7 header. The looming demands and increasing complexity on the more established Layer 2-4 header classification are the primary concern in the market for the next several years, and cannot be ignored. This paper explains why this is the case.

### ***IPv6 and Associated Market Requirements***

Existing networks, as well as the emerging MPLS networks, use traditional and well-understood Ethernet, PPP (Layer 2) and IPv4 (Layer 3) addressing protocols. These protocols serve existing networks reasonably well today; however, IPv4 layer 3 addressing space is beginning to run out with the proliferation of wireless products and a multitude of other networking appliances.

Tunneling IPv6 over IPv4 solves that problem. Combining IPv6/IPv4 over existing networks is a reasonable solution since IPv4 will remain ubiquitous. The infrastructure supporting the IPv4 protocol suite is well established, and implementing an all-native IPv6 infrastructure is far too expensive and disruptive to even contemplate. Therefore, working around the IPv4 address limitations must be accomplished on the existing IPv4 and emerging MPLS networks.

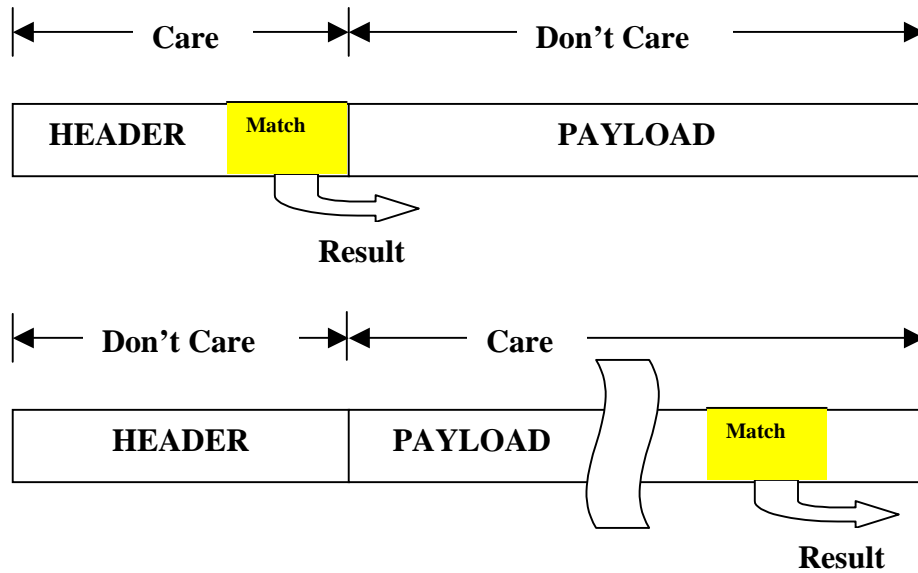
Recently much to do has been made over deep packet classification. These deep packet classification techniques are purported to allow service providers to provide “content aware” premium services with associated tiered billing. While this is an important new classification technology the associated market is in its infancy. On the other hand the markets for products that require layer 2-4 header classification today is huge and growing. According to Goldman Sachs Research, Layer 2/3 switching (including WAN switching, 10 GbE, and IP switching) will grow from \$13.3 billion in 2002 to \$18.6 billion in 2005.

Many would assume that since the market for products that perform layer 2- 4 header classification is large and established, that solutions are well understood and implemented. While to some extent this may be true, many factors are moving forward simultaneously that dramatically complicate and challenge the existing solutions. The advent of IPv6 deployments is a prime example. These factors make continued enhancement to layer 2-4 classification an extremely critical area of new technology focus.

### ***Header Classification Vs. Deep-packet Classification***

Header classification that examines specific fields (such as an IPv4 or IPv6 addresses) at known locations is conceptually simpler than deep content inspection of data patterns that are located at random places deep within the packet payload. Header classification involves looking at known, fixed fields in a header and classifying them using table searches. Since only the header is analyzed, it is not necessary to wait until the entire packet has been received before beginning the process.

Deep-packet classification, on the other hand, uses string searches to look for data patterns (cookies, URL's, CGI's) deep within the packet payload. These string searches often involve the classification of complex regular expressions, requiring the use of general-purpose CPUs or special-purpose ASICs. Moreover, deep content inspection is inherently non-deterministic resulting in dramatic swings in time to resolution. This non-deterministic nature is generally incompatible with today's highly channelized transport networks where increasing bandwidth demands coupled with end-to-end latency and quality of service, require time-critical deterministic analysis. Consequently, forwarding decisions are made much sooner when only header fields are classified as less of the packet must be buffered before the process begins and subsequently completes.



**Figure 1: Header vs. Data Payload searches**

Bay Microsystems contends that for network processing silicon solutions, Layer 2-4 classification should be integrated directly into the fast path NPU function, while deep packet classification should be passed on to coprocessor silicon. There are several very important reasons for this. First, neither the market nor the technology supports the need to integrate deep packet functionality into the majority of applications. Second, the Layer 2-4 performance and buffering overhead would be severely impacted because the packet must be stored until the content match is found. Finally, updates in L2-4 policies are generally very fast ( $\sim\mu\text{s}$ ), and can make use of TCAM technologies, while deep packet policy updates are invariably made in software through the control plane, and can take seconds to perform.

The issues associated with Layer 2-4 header classification are therefore of an entirely different nature than deep packet, and from both a marketing and technical point of view should be decoupled.

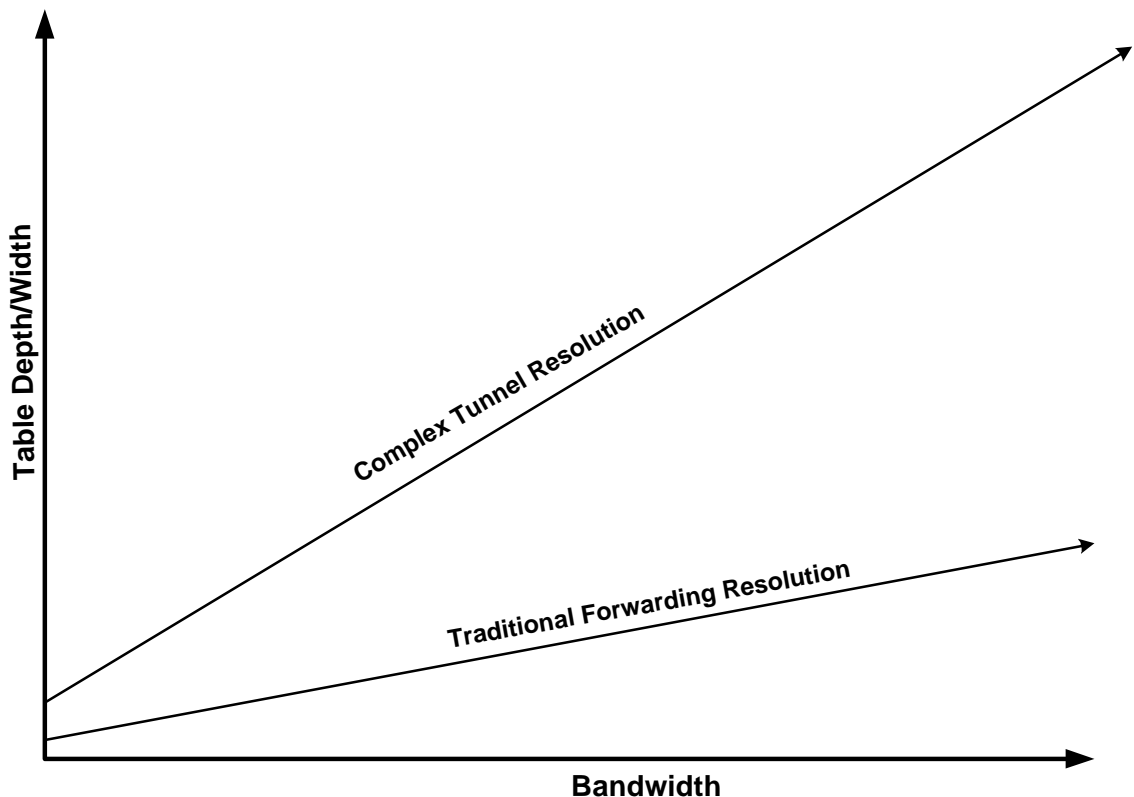
### ***Complexity Factors Affecting Header Classification***

What are the factors that impact the effectiveness of header classification techniques? And why must we continue to innovate? They are:

- **Growth of tables.** Tables expand in both depth due to ever increasing numbers of users (number of entries) and width due to more complex and numerous addressing (the number of bits that must be compared per search)
- **Available search time.** The time to perform the search shrinks with increasing network bandwidth

- **Stateful analysis.** Repetitive searches on the same packet require state to be passed from search to search or compared for validity after parallel searches in order to resolve the classification
- **Traffic engineering.** Requirements to add fine grained QoS for traffic engineering
- **Deterministic classification.** Determinism is an important factor in maintaining wire speed system efficiency because without it packet forwarding rates could not be guaranteed
- **Increasingly complex tunnels.** Increasing tunnel complexity requires greater and greater numbers of repetitive searches to be performed on the same packet

All of these factors taken together present a significant processing challenge. As tunneling techniques have been developed to provide internetworking between disparate infrastructures, they cause the advent of very deep, complex encapsulations that must be resolved. Coupled with the demands for higher and higher bandwidth, and greater and greater numbers of users, these conditions all necessitate silicon solutions that are at once deterministic, powerful, highly flexible, and cost-effective. Silicon vendors must have a strategy to provide the right resources to meet these demands today.



**Figure 2: The table size required for searches increases as a function of bandwidth, depth of classification, and complexity (multiple fields, deep encapsulation, multiple passes)**

The factors of growth of users, increases in bandwidth demands, and the need to guarantee wire-speed packet forwarding rates should be well appreciated. However, appreciation of complex tunnel resolution and the associated state issues may not be commonly understood. So we will take a closer look at these last two issues.

### ***Complex Tunnel Resolution***

To appreciate the daunting challenge of resolving complex tunnels, it is necessary to understand what they look like. The following example illustrates encapsulating IPv6 inside IPv4 as a good representation of tunnel complexity. LT2P could also have been used as a good example tunnel as it is quite common today.

Below is the packet header associated with a CMTS network. In the example, an IPv6 packet was encapsulated inside and IPv4, which was further tunneled within a three deep MPLS label stack. Finally, a layer 2, 802.3 SNAP header with VLAN was added. At the time this packet arrives at the final wireless base station for transmission to the IPv6 enabled handheld device, it is necessary for the base station router to statefully examine each portion of the header independently starting with the layer 2, then the MPLS stack, then the IPv4, and finally arriving at the IPv6 in order to propagate the packet to the appropriate device. If an examination results in the decision to search again on another piece of header information (as it would in this example) it is critical to pass what is known as a context (state) on to the next instruction prior to the formation of the search. This context is used to form the resulting search key.

Two types of table searches can be utilized to resolve this tunnel. The first type, known as best match, attempts to find a match of the greatest number of equivalent bits within the key, even though not all of the bits within the key match exactly. The second type, known as exact match, requires that all bits within the key match exactly. Since two types of searches can occur on the same packet and state must be passed amongst all of them intelligent use of resources (processing engines, algorithms, and memory tables) can result in dramatic improvements in efficiency, effectiveness and implementation cost.

In the figure below the fields that must be examined are shown in white. The header depth classified in this example is 108 bytes (864 bits) and the number of bits that must be examined and resolved is 592. And they must all be resolved at wire speed.

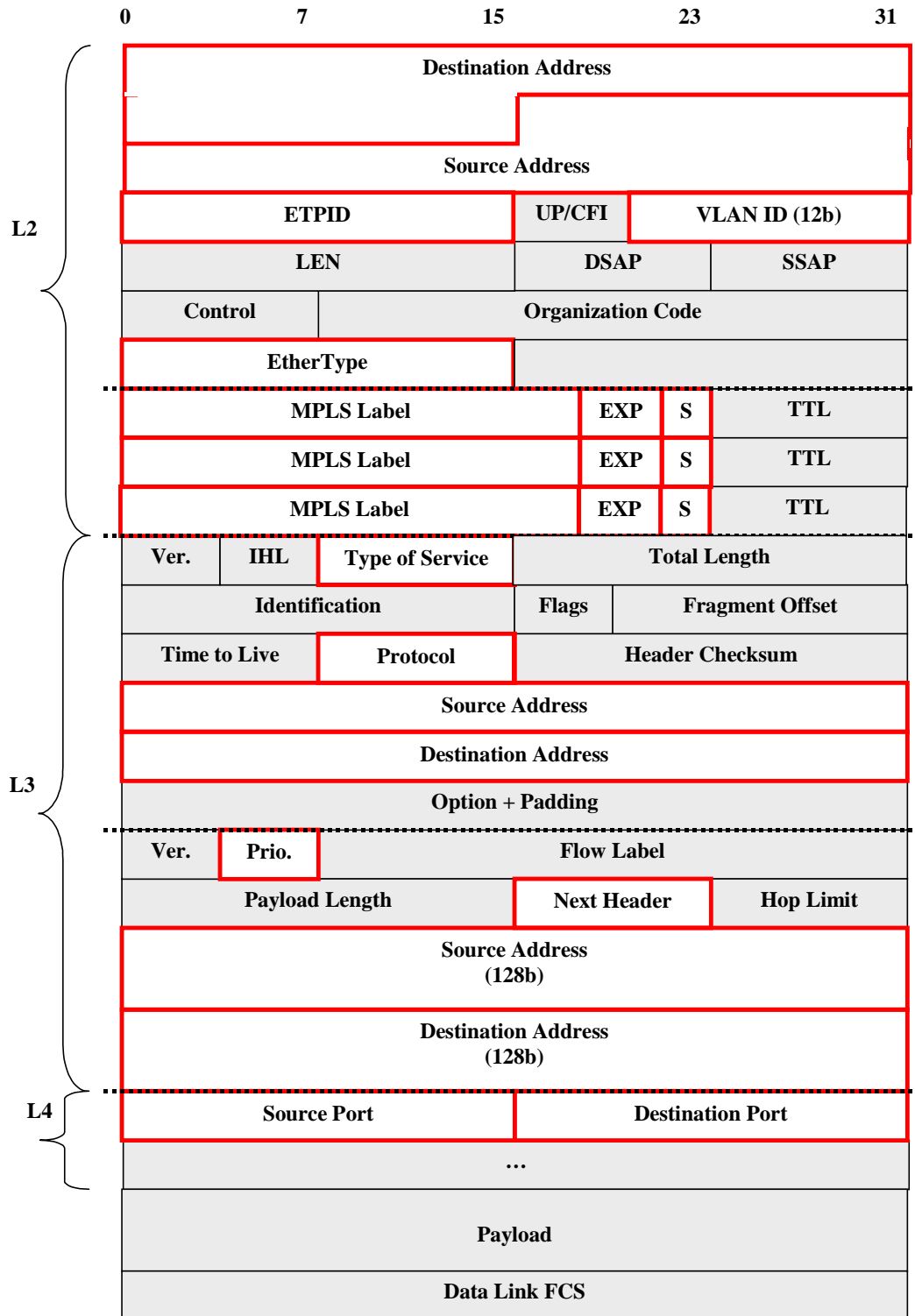
### ***The Solution***

Most solutions today use either an algorithmic tree-based approach running on a standard NPU coupled to SDRAM/SSRAM tables, or the solutions rely on Content Addressable Memories (CAMs). Algorithmic-based solutions (Hash, Trie, Chained Index, Chained Hash, etc.) have serious disadvantages at high search rates. They are primarily non-deterministic, and can require back tracking in the search tree, resulting in unpredictable response times. Because of this, they

are generally applicable to lower search rate applications (1Gbps streams or less) where table size efficiency can be traded-off for performance. For high-end applications, CAMs are generally the solution of choice, with three state CAMs known as ternary CAMs (TCAMs) predominating.

A third approach has arisen in the form of classification co-processors utilizing internal embedded table memory. The market has simply not accepted this approach because table sizes grow extremely quickly and must be tuned on a system-by-system application basis. System designers must have the freedom to appropriately 'right size' their tables, thus only a solution supporting external tables (or a combination of limited internal tables and external tables) is acceptable.

The market has clearly indicated that for high performance applications, solutions must utilize TCAMs. However, TCAMs alone will be insufficient to efficiently stem the rise in search rates while maintaining cost-effective implementations.



**Figure 3: 802.3 SNAP packet with VLAN/3 deep MPLS label carrying IPv6/IPv4 traffic (RFC2529)**

With the advent of OC192c packet streams requiring complex tunnel resolution, it will soon be commonplace to see applications requiring well in excess of 100 million packet searches per second. A single TCAM array today has a maximum search rate of 100 million searches per second and this does not factor in the inefficiencies associated with table updates. Though it is clear that TCAM tables represent the highest performance most deterministic solution available today for best-match searches, they impose cost and power penalties upon the system designer.

It is important to pay close attention to the balance between cost, power and performance of various table memory types and carefully choose the proper combination of external memory interfaces supported upon the classification processor.

We propose that the industry needs Layer 2-4 classification processors that are designed to work with search tables stored in external TCAM and SRAM to guarantee deterministic, complete examination with wire speed performance. TCAM stores tables where best-match results are needed, and SRAM stores tables where exact-match results are needed. The classification processor must provide the intelligence to exercise external memories statefully and appropriately by seamlessly toggling between best match and exact match tables. This will result in a maximization of performance, and efficiency while giving the system designer the ability to minimize power and to 'right size' their tables. Furthermore, the use of limited internal memory tables can be used for specific matches that are not likely to grow significantly in size. The use of external memory interfaces allows customers to populate their boards with the right amount and combination of memory required.

## **Conclusion**

We cannot ignore the growing demands and increasing complexity on the more established Layer 2-4 header classification in the market. To meet this demand and complexity, Bay Microsystems' classification approach is to integrate complete Layer 2-4 classification into the central network processing solution, and work with third party vendors to provide complimentary solutions for deep-packet classification as appropriate. Bay's vision of complex tunnel resolution implements a multi-pass protocol classification technique that addresses the objectives of:

- Minimizing cost
- Minimizing power consumption
- Maximizing performance

These objectives can be achieved by the use of the appropriate resource for the appropriate task as defined by the system designer. Too few resources would necessitate classification in many multiple stages, driving up costs. Too many resources would result in excessive power consumption, real estate, and silicon – again driving up costs. If the right resources are applied, a classification result can be achieved deterministically and economically.