



# DECnet to TCP/IP Migration Considerations

Scott Wattum  
16-Jan-2003

*Disclaimer – this is not meant to be a definitive discussion on migrating from DECnet to TCP/IP; more a starting point to look at what types of services might be offered to customers that desire to perform such a migration. Because of time constraints, this document has not been peer-reviewed; therefore any ridiculous statements should simply be ignored.*

## 1 Overview

This document will look at migration considerations, possible services and software, which may be offered to customers. We will begin by looking at the similarities and differences (though not in any great detail) between DECnet and TCP/IP as an aid in understanding some of the migration issues those customers may encounter. We will then take a look at what types of customers we have, followed by an examination of the migration services that we may be able to offer these specific customer types.

In some cases the migration services will pre-suppose the existence of some vaporware which is currently being discussed as possible future projects with hp, where appropriate, the document will discuss the services that this vaporware is expected to provide.

## 2 Comparison of DECnet to TCP/IP

Here we'll briefly compare and contrast DECnet and TCP/IP.

### 2.1 DECnet

DECnet exists in effectively 2 flavors, which is discussed briefly in the following sections, but for our purposes, one of the most important distinctions that DECnet has is that it is a **packet oriented** protocol – that is, application data is exchanged between systems in packets of a particular size.

#### 2.1.1 Phase IV

The first version of DECnet was released in 1975, making it now a relatively old protocol in computer terms. Highlights of DECnet Phase IV include:

- 16 bit addressing (large at the time, but by today's standards a very limited addressing space).
- Management via NCP (a fairly simple terminal oriented interface)
- Close coupling with the VMS operating system, especially with file services provided by RMS (the file system).
- Provided access for customer written applications via RMS services, language file services or the QIO interface.
- Over time, an incredibly stable and reliable networking platform.

#### 2.1.2 Phase V

A major enhancement of the DECnet Phase IV implementation – in fact, in many cases a complete re-write. Highlights of Phase V include:

- An expanded addressing space allowing a virtually unlimited number of systems.
- Management via NCL (a more carefully designed architecture but unfortunately a much more complex terminal oriented interface).
- Backward compatibility with PhaseIV
- Support for new ISO protocol standards in addition to the existing NSP protocol used by Phase IV.
- And as ISO fell into disfavor, support for using TCP/IP as the networking backbone protocol using RFC1859.

## **2.2 TCP/IP**

TCP/IP effectively exists in 2 flavors as well, though the main difference between the 2 flavors is in the area of routing or address space. IPV4 allows a 32 bit addressing space, while the new IPV6 extends the addressing space to 128 bits. Again, for our purposes, the most important distinction with TCP/IP is that it is a **stream or byte oriented** protocol. Data is exchanged between systems, as simply a stream of bytes, there is no concept of a packet unless the application using TCP/IP imposes this concept.

For VMS, TCP/IP was something of a latecomer to the party, so it's not quite as well coupled with the operating system, though in recent years this coupling has improved to the point that for all practical purposes, most customers probably do not see any major distinction in the services offered between DECnet and TCP/IP.

And TCP/IP has certainly won the "protocol war," being today the most widely deployed networking protocol.

## **2.3 The Comparison and Contrast**

In general, DECnet and TCP/IP offer the same basic services to applications and the operating system:

- Both offer a QIO interface that applications may use to communicate across the network.
- Both offer high-level utilities for file transfer, email and remote terminal access (among others).

In contrast:

- TCP/IP provides a stream or byte oriented protocol, while DECnet provides a packet oriented protocol.
- DECnet is more closely coupled to the file system offering additional file system related services transparently to the application via either file system calls or I/O operations provided by a particular language. TCP/IP is not quite as closely coupled in general, but is fairly transparent when using the C programming language. TCP/IP also provides an additional programming interface that is not available with DECnet.

## **3 The Customer Base**

Here we'll examine the current mix of VMS and OpenVMS customers along with reasons why a particular customer may be utilizing a particular networking protocol.

There are 2 aspects that need to be considered:

- The Network Backbone – this backbone may be based on Phase IV, Phase V, TCP/IP or a mix.
- Applications – as with the network backbone, customer applications may be a mix of Phase IV, Phase V, TCP/IP or a mix.

It should be noted that you shouldn't look at either the backbone or the application in isolation from each other. Helping the customer understand that the application migration must be planned in conjunction with the network migration is critical.

While it is tempting to combine the Phase IV and Phase V applications into a single group and simply label them "DECnet" we can't do this for various reasons that we'll examine.

### **3.1 The Network Backbone**

Network backbones except in the smallest of companies are usually a complex beast and regardless of what a customer may think, are frequently multi-protocol (with probably the 2 most common protocols being TCP/IP and NETBIOS). In situations where only a local area network (LAN) is involved, multi-protocol environments are usually easy to maintain depending on the protocols involved.

However, in environments where the network is no longer local, and contains an extended LAN or wide-area network (WAN), multiple protocols become much more problematic. Routers or bridges connecting the individual smaller LANs become more expensive and harder to manage as you add additional protocols to the mix. In these situations customers may wish to restrict the protocols shared across a router or bridge, hence the usual desire to settle on a single standard networking protocol.

For example, the complexity associated with managing a DECnet Phase IV and a TCP/IP network – some systems may need DECnet addresses but not TCP/IP addresses, some systems may need both and others may need only TCP/IP addresses. Managing 2 disjoint addressing spaces can easily become twice as difficult as managing one.

#### **3.1.1 Migration of the Backbone**

Migration of a network backbone will typically fall into one of 2 categories:

- Migration from a DECnet IV/V backbone to TCP/IP
- Migration from a mixed protocol backbone (such as DECnet IV/V and TCP/IP) to only TCP/IP.

##### **3.1.1.1 Moving from DECnet only to TCP/IP**

It seems unlikely to me that in this day and age, we'll have many customers with this particular type of network configuration. Consequently, I won't spend a lot of time on it.

Probably the single easiest way to migrate from a DECnet only backbone to a TCP/IP only backbone would be to perform the migration in 2 steps:

- Migrate to a mixed DECnet & TCP/IP backbone. This would likely be the least disruptive, allowing applications to run in parallel during the application migration process.
- Then migrate to TCP/IP only

Please note though that this type of a migration strategy carries with it its own risks and potential problems. For example, the complexity of maintaining 2 address spaces needs

to be carefully weighed against the reduction in risk associated with cutting over an application concurrent with the network.

There are a number of considerations when migrating in this situation where we can provide services to the customer:

- Training in TCP/IP networking – this is probably first and foremost, as there's a good chance the customer may have little if any experience with TCP/IP networking. They may have expertise with using TCP/IP in LAN situations, but may not have experience using TCP/IP in a WAN environment. Understanding the customer's training requirements and then helping the customer obtain the required training or delivering the required training.
- Understanding what their current network looks like and helping evaluate what equipment needs to be replaced during the migration process. For example, various routers may need upgrading or replacement with newer routes which can perform multi-protocol routing. Identify equipment which may not need to be replaced.

### ***3.1.1.2 Moving From Mixed Protocols to TCP/IP Only***

This is probably the most common type of networking “backbone” environment that customers have. The migration will be trivial from the networking perspective; here we focus on migrating the applications, and once all of the applications have been migrated, we simply start shutting down DECnet.

## **3.2 Applications**

This is the area where most of the work will probably be involved. Customer applications will typically fall into the following categories:

- Applications without source or that for any reason cannot be recompiled/relinked.
- Applications with source but that the customer does not have the expertise to modify, yet would be willing to pay for the modifications needed.
- Applications where the customer is perfectly happy with the DECnet over IP solution provided by DECnet-Plus and where the customer is concerned about future DECnet-Plus support, but is not willing to invest in modifying the application to be TCP/IP only.

Within the application categories, we'll further break down into 2 sub-categories:

- Applications that utilize the QIO interface to DECnet.
- Applications that use the file system interface (via RMS services, language I/O or other provided utilities such as DCL or COPY).

### **3.2.1 Degrees of Application Migration**

While it is possible for a network backbone to be TCP/IP only, existing DECnet applications will potentially have “degrees of migration.” Applications could:

- Migrate completely to TCP/IP by converting to use the TCP/IP QIO interface or some other programming interface supplied by TCP/IP (such as C sockets).
- Migrate from DECnet Phase IV to DECnet-Plus and utilize the DECnet over IP capabilities of DECnet-Plus. DECnet over IP in some fashion.

### 3.2.1.1 *The Complete Conversion*

Whether or not an application can be completely converted to TCP/IP depends largely on the application, and on how much effort or money the customer is willing to invest.

It should be noted that for many customers, making any type of change (even trivial changes) to an existing application might involve considerable expense and effort.

Applications need to be evaluated on a case-by-case basis to determine whether they can or cannot be completely converted. In this process, the following considerations will need to be taken into account:

- Does the application rely on packets? Many DECnet applications expect a complete packet (or record) of information to be returned on a QIO/READVBLK request. TCP/IP makes no promises about complete packets or records being returned via QIO/READVBLK, so an application may need to be re-coded to handle the possibility that incomplete data, or more data than requested may be returned. A careful evaluation needs to be done to determine whether the application is a good candidate for complete conversion.
- Does the application use language I/O routines and is not written in C? Because DECnet is closely coupled with the file system on VMS, many applications may choose to take advantage of the ability to access DECnet via I/O functions supplied by the particular language. In the case of C, this conversion **might** be trivial depending on the I/O method used; for other languages though the conversion will require much more effort and in fact, it might be better to provide some additional software (see the vaporware section) to the customer that emulates the language I/O operations over TCP/IP.
- Does the application use RMS services or other utilities? We have the same considerations here that we have with language I/O; but the problem is that RMS provides many more features and thus makes providing emulation software an increasingly complex proposition. In addition, these applications probably rely heavily on complete packets or records of information being exchanged with each I/O operation. These applications may not be good candidates for complete conversion.
- Does the application have source code and can it be recompiled/relinked? Applications that lack source code or cannot be recompiled/relinked are not candidates for a complete conversion to TCP/IP. The only option the customer has in this case is to utilize DECnet over IP.

### *3.2.1.2 The DECnet/IP option*

One of the advantages with DECnet-Plus is that it provides relatively transparent access to TCP/IP as a networking backbone.

DECnet-Plus achieves this by utilizing RFC1859, which basically wraps a small amount of protocol information around the packets sent by the application that allows the remote receiver to re-assemble the stream of bytes supplied by TCP/IP into a packet that is subsequently delivered to the cooperating application. Basically we impose a packet mechanism on the stream mechanism supplied by TCP/IP.

Almost every extant DECnet application is a candidate for this type of migration. Applications that are not a candidate are those that utilize hard coded numeric addressing, or hard coded DECdns addressing (hopefully a minority).

The major drawback to DECnet-Plus for many customers is the perceived complexity of the product, especially in comparison with the relative simplicity of DECnet Phase IV. Even though DECnet-Plus has made great progress over the years in reducing the complexity, many customers simply refuse to migrate from Phase IV. For these customers, migration to DECnet over IP might be an option if a less complex product than DECnet-Plus were offered (discussed further in the vaporware section).

## **4 Vaporware**

Recently engineers within the DECnet group held some discussions regarding how we could further facilitate the migration from DECnet to TCP/IP. The following outlines some of the ideas.

### **4.1 DECnet-Lite**

The goal of DECnet-Lite is to provide DECnet-Plus in a form that favors configuration of DECnet in a TCP/IP only environment. Many of the components of DECnet-Plus are still provided, but significant changes to the configuration procedures are made such that DECnet-Plus will only work over TCP/IP.

Customers are not required to make application changes with this option, however, some configuration changes and management procedures may need to be modified.

### **4.2 DECnet Phase VI**

This is really a hybrid of DECnet Phase IV and DECnet-Plus – the primary goal here is to provide a complete NCP management interface to DECnet-Plus, thus customers are not required to use NCL for management. Configuration would be oriented toward TCP/IP as the networking backbone, but all existing transport and addressing options would still be accessible to the customer.

Customers are not required to make application changes with this option. No changes to configuration or management procedures are required.

### **4.3 DECnet/IP**

A new NETDRIVER (the interface that all applications and utilities use to access DECnet) is provided that supports access to RFC1859 (DECnet over IP). Two types of driver would be available:

- RFC1859 only mode – no access to other DECnet transports/addressing is provided. This option would be available for DECnet Phase IV or DECnet-Plus systems and installation would effectively remove most of the previous DECnet version.
- Migration mode – access to RFC1859 as well as other DECnet transports/addressing is provided. This would be targeted at DECnet Phase IV systems.

Customers are not required to make application changes with this option. In general, changes to configuration are not required. Management capabilities are severely limited.

### **4.4 Language I/O Emulation**

The goal of language I/O emulation would be to provide a set of run-time libraries that emulate the actual I/O operations over a TCP/IP connection. These would not necessarily be general purpose, but instead would provide functions on an application-by-application basis. In addition, each application would need to be evaluated to see if this was a viable option.

Coding changes to the application would be required, though changes should not be extensive.

## **5 Conclusions**

Each customer is unique and will have unique migration requirements. Some customers will not want to make intrusive changes to their applications, thus eliminating the “Complete Migration” option. Some customers may be willing to make application changes, but will compare the cost of those changes against the cost associated with migrating to an entirely different application that doesn’t have the same migration issues.

Because each application is unique, it’s difficult to create a step-by-step procedure for application modification that will apply to every application. This was one of the driving reasons behind the implementation of RFC1859 in DECnet-Plus.