

Common Region *On Line*

Customer Services Enterprise Customer Support Knowledge Center 2

OpenVMS

1991年 8月

Vol.21

DECdns、DFS



コンパックコンピュータ株式会社
カスタマーサービス統括本部

COMPAQ

Vol.21
「DECdns,DFS」

目次

はじめに

第 1 部 DNS

第 1 章 DNS 概要	1-2
1.1 DNS とは	1-3
1.2 DECdns の特徴	1-4
1.3 サーバ/クライアント・モデル	1-5
1.4 DNS の構成要素	1-6
1.5 名前の検索	1-9
1.6 名前の更新	1-11
1.7 DECdns まとめ	1-12
第 2 章 DNS 管理	1-14
2.1 DECdns のインストレーション	1-15
2.2 DECdns の起動と停止	1-20
2.3 DECdns のプロセス	1-22
2.4 DECdns マネージメント	1-23
第 3 章 DNS プランニング	1-34
3.1 名前の付け方	1-35
3.2 ディレクトリの設定	1-36
3.3 サーバの設定	1-37
第 4 章 DFS	1-38
4.1 DFS とは	1-39
4.2 DFS の設定方法	1-41
4.3 DFS と DECdns	1-44
4.4 アクセスポイントの参照	1-46
4.5 アクセス制御	1-49
4.6 その他	1-51

第 2 部 CS セミナー Q & A

2.1 Phase V	2-2
2.2 OSI	2-10
2.3 ULTRIX & TCP/IP	2-12
2.4 SNA	2-14
2.5 DECmcc	2-15

第 3 部 TSC コールサマリー

3.1 MAIL が送れない。(EXDISKQUOTA エラー)	3-2
3.2 MAIL が送れない。(REMRSRC エラー)	3-3
3.3 SET HOST できない。(LINKABORT エラー)	3-4
3.4 SET HOST の制限	3-5
3.5 DECnet ノード名	3-7
3.6 NCP 実行時の特権	3-8
3.7 トラブルシューティング ~ UNREACHABLE エラー	3-10

はじめに

この夏 DEC は、新しいネットワーク・アーキテクチャ DECnet/OSI Phase V を発表しました。この Phase V のコンセプトの中に分散ネームサービスを提供する DECdns があります。DECdns は Phase IV にもありましたが、Phase V からは必須となります。

今回の TSC Common Region では、

第 1 部では、DECdns を説明いたします。

第 2 部では、先月行なわれました CSセミナーでお客様からいただいたご質問について説明させていただきます。

第 3 部では、Phase V とは関係ありませんが DECnet 関係の TSC コールサマリーを Q & A 方式で掲載いたします。

本誌が皆様のお役に立てれば幸いです。

各ソフトウェア・プロダクトのご購入のお問い合わせにつきましては、最寄りの営業所へご連絡ください。

第 1 部 DNS

第 1 部では、DNS についてご紹介いたします。

また、合わせて、DNS を使用するアプリケーションとして DFS を取り上げてみました。

ここでご紹介する DNS は、DECnet Phase IV での DECdns Version 1.1 ですが、以下の内容について説明いたします。

1. DNS 概要
2. DNS 管理
3. DNS プランニング
4. DFS 概要

なお、記述されている内容、コマンドとその結果は、VMS Version 5.4-1, DECdns Version 1.1 また VAX DFS Version 1.2 のシステム上のものです。

参考マニュアル

VAX Distributed Name Service Installation Guide
Management Guide
VAX Distributed File Service Installation Guide
Management Guide

第 1 章 DNS 概要

この章では、DNS 概要についてご説明いたします。

サーバ/クライアントの関係から、DNS のコンポーネントであるネームスペースやクリアリングハウスの概念、名前の検索と更新などについてご説明いたします。

1.1 DNSとは

質問：DECdnsとはどういうものですか？

DECdns (DEC Distributed Name Service) は、DECnet System Service (以下、DSS と略します。) 製品群の 1 つです。

DSS 製品群は、ネットワーク上の複数の VAX を 1 つのシステムとして利用できるようにしたものです。これによりユーザは、ネットワーク上に広がる各種資源を、あたかもローカルに接続されているかのように共有/管理することが可能になります。

その中で DECdns は、"名前" というものをネットワークワイドに提供する役割を果たすソフトウェアです。ネットワーク内で、ファイル名やディスク名、ノード名、キュー名、メールボックス名など各種資源をユニークな名前でアクセスできるようにしています。

DECdns を使っているソフトウェア製品がいくつかありますので、ご紹介します。

VAX Distributed File Service (VAX DFS)

ネットワーク内でのディスクの共有を可能にするソフトウェアです。
アクセスポイント名を DECdns に登録し、利用します。

Remote System Manager (RSM)

リモートノードからのバックアップ、ソフトウェアのインストール等のシステム管理を可能にするソフトウェアです。
サーバ/クライアント・モデルを使用し、クライアント・モデルの管理に DECdns を利用します。

VAXNotes

電子掲示板です。
コンファレンス名を DECdns で管理します。

DEC Distributed Time Service (DECdts)

ネットワーク内の各ノードのシステム時間の同期をとるためのソフトウェアです。
DECnet-VAX V5.4 Extensions よりサポートされます。
グローバルサーバの管理に DECdns を用いています。

DECmcc

ノードの管理に使用します。

1.2 DECdnsの特徴

質問：DECdns の特徴を教えてください。

いくつかありますが、まず例を上げて説明します。

DECdns を用いていない場合、例えばあるノードのディレクトリは次のように指定します。

```
VAXA::PROJECT$DISK:  
(ノード名::デバイス名:)
```

この場合、次のような不便な点があります。

- ユーザがそのディレクトリにアクセスしたい時に、どのノードにあるかを知っていなければいけない。また、他のノードに移動した場合には、それも認識し、変更しなくてはならない。
- 名前はノード上で定義されており、ネットワーク全体で定義されたものではない。
- ノードごとに名前が異なる可能性もある。
- 名前に対するアクセス制御がノード単位であるため、セキュリティ管理が困難。

その点 DECdns を利用した場合には、以下のように便利なことがいくつかあります。

- ユーザは名前さえ知っていれば、どこのノードに実際アクセスしているかを意識せず使える。また、名前で示されている実際の資源が移動したとしてもそれによってユーザは影響を受けずにすむ。
- ネットワーク内のどのノードでも、1つの資源に対しては同じ名前を利用できる。
- ネットワークワイドで名前はユニークである。
- 名前に対するアクセス制御をネットワークワイドで管理できる。

また、DECdns はサーバ/クライアントモデルを採用しています (詳細は後述します)。このことにより、次のような利点もあります。

- サーバを複数おくことにより、負荷が分散される。
- 複製 (レプリカ) を設定することにより、高い可用性が得られる。
- 分散させることにより、パフォーマンスが向上する。
- 一元管理のため、名前の管理が容易。

1.3 サーバ/クライアント・モデル

質問：DECdns のサーバ/クライアント・モデルの動きを教えてください。

DECdns は、大きく分けて次の 3 つから構成されます。

◆DNS サーバ

各資源の名前をデータベースとして保存し、DNS クライアントの利用者（ユーザ、アプリケーション）に対しリクエストに応じて名前を提供します。

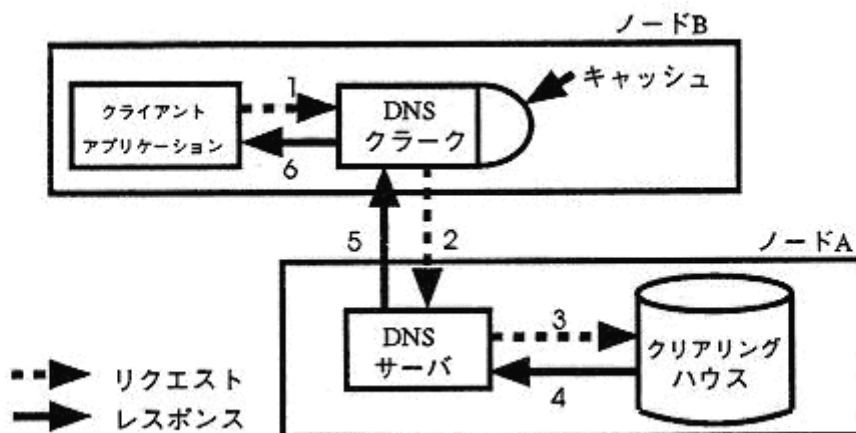
◆DNS クライアント

名前を実際に使っている利用者を指します。ユーザやアプリケーションです。

◆DNS クラーク

DNS クライアントとDNS サーバとの仲介を行っています。

簡単な動きを下図に示します。



1. ノードB のクライアント・アプリケーションが DNS クラークに対して検索リクエストを発行します。
2. DNS クラークは、リクエストのあった名前の情報がキャッシュにあるかどうかを調べ、あればクライアント・アプリケーションにレスポンスを返します。キャッシュに該当の名前に関する情報がない場合、ノードA の DNS サーバにリクエストを発行します。
3. リクエストを受け取った DNS サーバはディスク上にあるファイル（クリアリングハウス）の中を調べ、
4. その結果を受け取ります。
5. DNS サーバは、ノードB の DNS クラークに結果を返します。
6. DNS クラークは、クライアント・アプリケーションにそれを伝えます。

1.4 DNS の構成要素

質問：DECdns で用いられる言葉について教えてください。

DECdns の構成要素として以下のようなものがあげられます。

- ネームスペース (Namespace)
- オブジェクト(Object)
- ディレクトリ(Directory)
- クリアリングハウス (Clearinghouse)
- レプリカ (Replica)

順に説明いたします。

1. ネームスペース

種々の名前を管理するための仮想的な集合体、空間を指します。ネームスペースは、名前を共有したいノード群に 1 つだけ定義します。

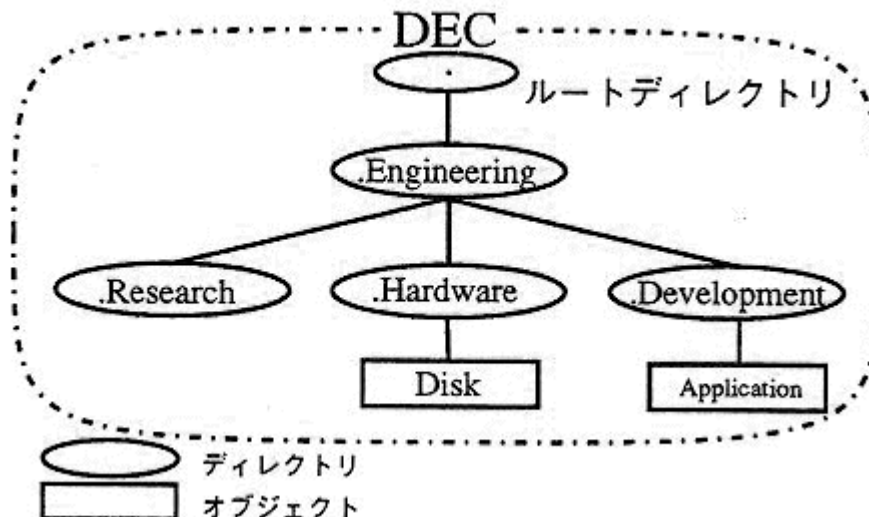
2. オブジェクト

ネームスペースで管理されている種々の名前をオブジェクトと言います。

3. ディレクトリ

ネームスペースは階層構造を持たせることができます。ネームスペースのディレクトリは、オブジェクトを階層的に管理します。最上位にあるディレクトリをルート・ディレクトリと言います。"." (ピリオド) で示します。あるディレクトリのサブ・ディレクトリをチャイルド・ディレクトリと言います。

例



Disk というオブジェクトを指定する際には、以下のようになります。

```
DEC: .Engineering.Hardware.Disk
```

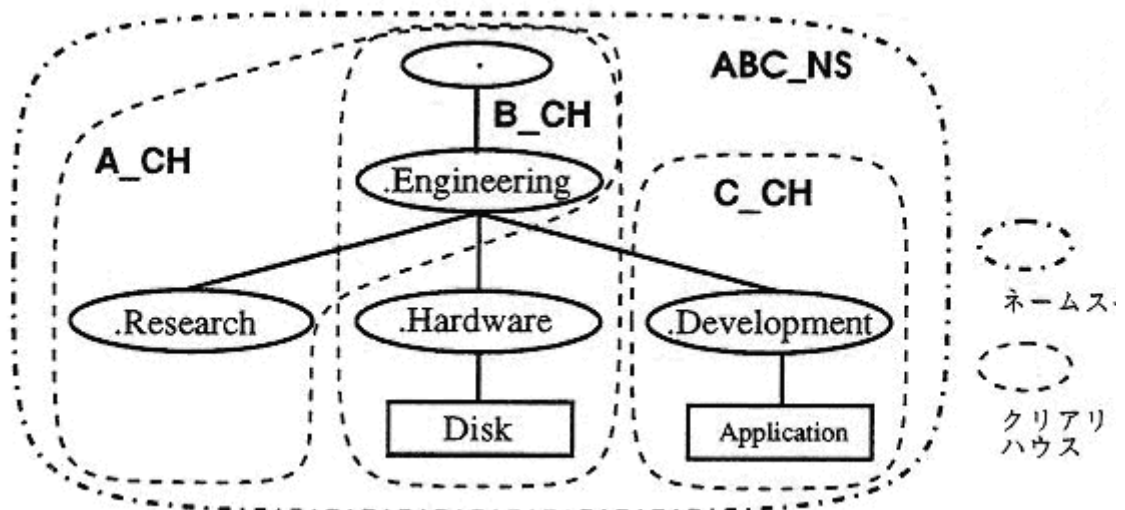
4. クリアリングハウス

DNS サーバを複数設置し、情報を分散することができますが、その際 DNS サーバが持っているディレクトリの集まりをクリアリングハウスと言います。ネームスペース全部または一部の情報を持っている情報センターのようなものです。

DNS サーバを複数持つ利点として、次のことがあげられます。

- サーバ・ノードの負荷を分散できる。
- 利用状況や環境に応じて設置することで、レスポンスを向上できる。
- サーバ・ノードの障害をバックアップできる。

複数のクリアリングハウスが同じ情報を持っていても構いません(下図参照)。クリアリングハウスは、各 DNS サーバ上にファイルとして存在します。



5. レプリカ

ネームスペースを管理する際に、ディレクトリ単位でコピー(複製)ができます。最初に存在しているものも含み、それらをレプリカと呼びます。このディレクトリ・レプリカを使って、クリアリングハウスに名前を分配しています。すなわち、クリアリングハウスはディレクトリ・レプリカの集まりとも言えます。

レプリカには、次の2種類があります。

・マスター (Master)

ディレクトリの最初のもので、このマスター・レプリカに対し登録、変更、消去などが行なわれます。

マスター・レプリカは、必ず1つだけ存在します。

•リードオンリー (Read-only)

マスター・レプリカに合わせて更新されていくもので、このレプリカに対しては直接登録など行うことはできません。名前を捜す際にのみ使われます。

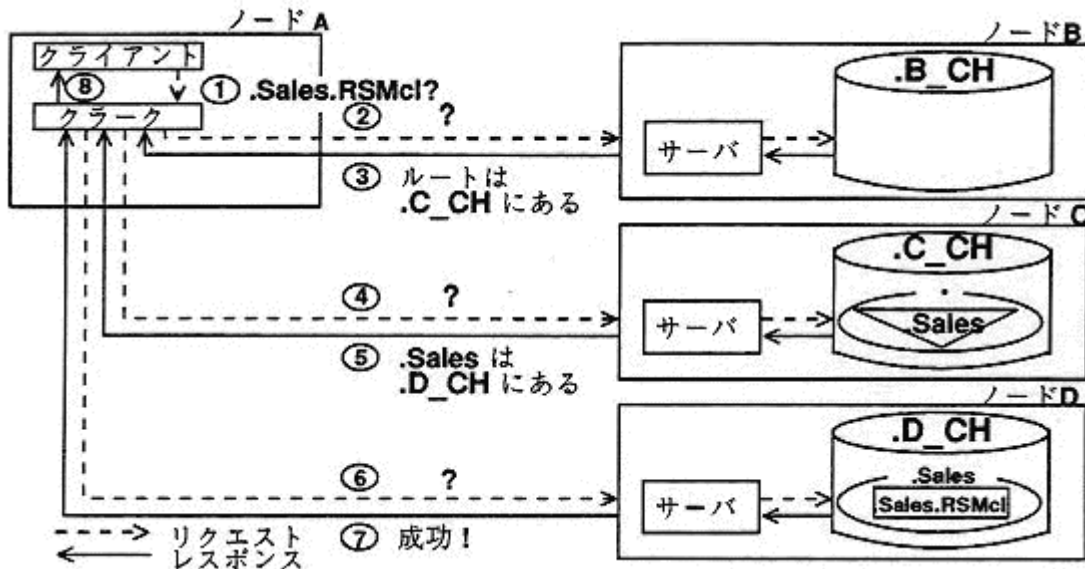
マスター・レプリカが変更されると、自動的にリードオンリー・レプリカも変更されるようになっています。詳細は、後程説明いたします。

1.5 名前の検索

質問:DECdns では、どのようにして名前を探すのですか?

ネームスペースが複数のクリアリングハウスに分割されている場合には、複数のサーバを参照して目的のオブジェクトを検索しなくてはなりません。

動きを簡単に図で示します。



この図は、ノードAのクライアントが .Sales.RSMcl という名前を検索する際の動きを表しています。

1. ノードA上で、クライアントソフトウェアが .Sales.RSMcl という名前のアドレスを知りたいというリクエストをあげます。
2. クラークはまず DNS サーバであるノードBにリクエストを出します。
3. クリアリングハウス .B_ch には .Sales.RSMcl も .Sales もありませんが、それらを含む可能性のあるルート(.)を持つクリアリングハウス .C_ch がノードCにあるのがわかりました。ノードBのサーバは、ノードCのアドレスとクリアリングハウスの名前 .C_ch とをクラークに知らせます。
4. クラークは、DNS サーバであるノードCにリクエストを出します。
5. クリアリングハウス .C_ch は該当の名前に関する情報を含んではいませんが .Sales ディレクトリのポインタを持っており、それにより .Sales ディレクトリが .D_ch にあることがわかります。サーバは、ノードDのアドレスとクリアリングハウスの名前 .D_ch をクラークに返します。
6. クラークは、DNS サーバであるノードDにリクエストを出します。
7. ノードDのクリアリングハウス .D_ch の中の .Sales ディレクトリは、捜していた名前 .Sales.RSMcl を含んでいるので、サーバは .Sales.RSMcl のアドレスをクラークに伝えます。
8. クラークは、その情報をクライアントに渡します。

この検索方法をツリー・ウォーク(Treewalk) といいます。

このような長い検索はそれほど頻繁には起こりません。クレークは、キャッシュにネットワーク上のサーバの情報やサーバが持っている情報を保持することができるからです。最初だけ、この図に書かれている方法で名前を検索します。

1.6 名前の更新

質問：DECdns ネームスペース内の更新がどのように行なわれるのか教えてください。

ネームスペースの中にはたくさん名前が存在します。

この名前は、あるアクセス権をもっているユーザであれば変更することが可能です。ディレクトリの内容が追加、削除、変更などで変化した場合、この変化をディレクトリ中の全てのレプリカに反映し、内容が同一なものであることを保証しなければなりません。そのため DECdns は次の 3 つの機能を持っています。

1. アップデート・プロパゲーション (Update Propagation)

アップデート・プロパゲーションとは、変更されたディレクトリの内容を即座に全てのリードオンリー・ディレクトリ(レプリカ)に反映させる動きを言います。

以下の "スカルク・オペレーション" の項目で述べるコンバージェンスの値を HIGH にした場合に実行されます。

変更点を一括して周期的に更新するスカルク・オペレーションと異なり 1 つの変更だけを直ちに伝えられるところが特徴です。もし、レプリカ (のノート) が通信不能の場合には変更は通知されませんが、その時は後述のスカルク・オペレーションにて変更されます。

2. スカルク・オペレーション (Skulk Operation)

DECdns は一定間隔毎に、全てのリードオンリー・ディレクトリ(レプリカ)を自動的に更新します。それぞれのマスタ・ディレクトリは、コンバージェンス(一括更新)の間隔が指定されているアトリビュート(属性) DNS\$Convergence を持っており、そのタイミングにより変更点を収集し、全てのレプリカへ送ります。この DECdns の更新機能をスカルク・オペレーションと言います。

コンバージェンス値を HIGH にした場合は、DECdns はマスタ・ディレクトリに変化があった時すぐに更新します。もし、更新に失敗した場合は成功するまで 12 時間毎に更新を試みます。もし、コンバージェンス値を LOW にした場合は、次にスケジュールされた時間 (24 時間後) まで更新を待ちます。デフォルトは HIGH です。

このコンバージェンスの値は、DNS\$CONTROL での SET DIRECTORY CONVERGENCE コマンドにて変更することが可能です。

またスカルク・オペレーションには、期限切れとなった情報 (ソフトリンク) を削除したり、ネームスペースを保守する機能も持っています。

3. コマンドによる更新

DECdns では DNS\$CONTROL での UPDATE DIRECTORY コマンドで、ディレクトリを強制的に更新する機能も持っています。

1.7 DECdns まとめ

DECdns とは ...

DECdns (VAX Distributed Name Service) はロケーションに依存しない、ネットワークワイドなオブジェクト(ファイル、ディスク、ノード名、キュー等)の名前を利用者に提供します。

特徴

- ネットワーク内で (個々にユニークな) 共通した名前を用いることができます。また、新しい名前は容易に追加できます。
- 名前の情報を複数のノードで持つことにより、あるシステムがダウンしても他のシステムから名前を参照できます。
- キャッシングによりネットワーク・トラフィックを減少させます。
- 数ノードの小さなネットワークも、百以上もある大きなネットワークもサービスできます。また LAN、WAN どちらの環境でも使用できます。
- 名前に対するアクセスを個人またはグループ単位で制御できます。
- 名前を格納するノードを集中又は分散させることが可能です。

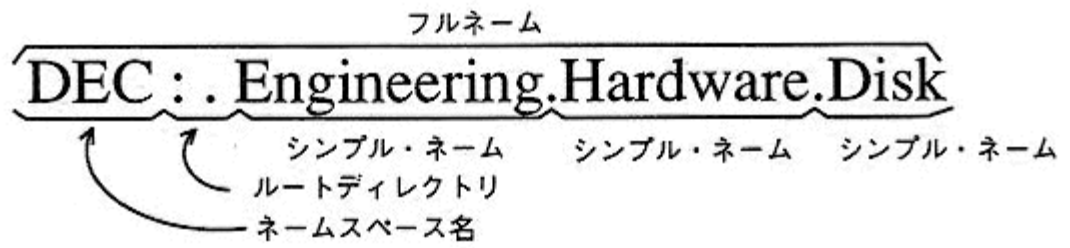
DECdns を使用するアプリケーション

RSM (Remote System Manager)	サーバとクライアントのノード名を格納
VAX DFS (Distributed File Service)	目的となるディスクまたはディレクトリのアクセス・ポイントを格納
VAX Notes	Version 2.0 よりノードのコンファレンス名を格納
* DECnet/OSI (Phase V) より	
DECdts (Digital Distributed Time Service)	グローバルサーバ・ノード名を格納
DECmcc	ノード名を格納

コンセプト

サーバ	---	名前の格納と管理
クライアント	---	アプリケーションで使用する情報を依頼
クレーク	---	クライアントとサーバとのインターフェース
オブジェクト	---	ネットワーク内の資源
ディレクトリ	---	オブジェクトを論理的にグループ化し、格納する場所
ネームスペース	---	オブジェクトとディレクトリを含む階層構造の空間
クリアリングハウス	---	各 DECdns サーバが持っているディレクトリの集まり
レプリカ	---	ディレクトリを物理的にコピーしたもの

名前の例



第 2 章 DNS 管理

この章では、DNS 管理についてご説明いたします。

DECdns のインストール、起動と停止、セキュリティ、DECdns の管理コマンドである DNS\$CONTROL の使用例についてご紹介いたします。

2.1 DECdnsのインストール

質問：DECdns (サーバ) のインストール方法を教えてください。

1. インストールをする前に

- DECdns のインストールは DECdns サーバ・ノードでのみ行います。DECdns クライアントの機能は VAX DFS や RSM の中に含まれていますので、クライアント・ノードでは DECdns のインストールは必要ありません。
- VAX/VMS Version 4.4 から 4.7 までは Version 1.0 を、VAX/VMS Version 4.6 から 5.4 までは Version 1.1 を使用します。
- VAX/VMS Version 5.0 以上で DECdns サーバ・ソフトウェアをインストールする際には PAK (Product Authorization Key) が必要です。
- ネットワーク内に初めて DECdns サーバ・ソフトウェアをインストールする時と 2 台目 (既にネームスペースが存在する場合) の DECdns サーバ・ソフトウェアをインストールする時では、方法が異なります。

1 台目のサーバのインストール時には、これから作成するネームスペースとクリアリングハウスの名前を指定します。

2 台目のサーバのインストール時には、ネームスペースを作成した 1 台目の DECdns サーバ・ノード名と、自分のノードが管理するクリアリングハウスの名前を指定します。

* クリアリングハウスは各サーバが個々に管理、所有するディレクトリの集まりです。

* ネームスペースはネットワーク内にただ 1 つだけ作成してください。ネームスペースを分けることは情報 (名前) の分離となり DEC はこれをお勧めいたしません。

- 既にネットワーク内に DECdns サーバ・ノードが存在する場合には、サーバ・ノードとこれからインストールしようとする 2 台目以降のノードとのシステム時間を合わせてください。もし、4 分以上の差があると BADCLOCK エラーでインストールは失敗します。この際、時間を修正する必要がありますが、既存の DECdns サーバ・ノード上で、SHOW TIME コマンドにて時間を確認した後、インストールしようとするノードで、SET TIME コマンドを実行してください。
- インストールを実行する前に以下のことを検討してください。
 1. DECdns サーバ・ノード(DECdns をインストールするノード) の決定

CPU パワー、メモリが十分あるもので、ダウンしないシステムを選択します。

2. ネームスペース名の決定

デフォルトでは nodename_NS となります。
ネットワーク全体で使用するのに分かりやすいものにします。

3. クリアリングハウス名の決定

デフォルトでは nodename_CH となります。
オブジェクトの識別、クリアリングハウスの管理が行いやすいものにします。

4. ディレクトリの構造とレプリカの計画

ネットワーク規模によって階層構造化するかどうか計画してください。
また、ディレクトリ作成時に、他のどのクリアリングハウスにレプリカを作成するかを検討してください。

* くわしくは、第 3 章 "DNS プランニング" の項をご覧ください。

2. DECdns サーバ・ソフトウェアのインストール (DECdns Version 1.1 の場合)

以下のコマンドで DECdns サーバ・ソフトウェアをインストールします。

```
@SYS$UPDATE:VMSINSTAL DNS011 ddcu (キットがマウントされたデバイス名)
```

以下に、1 台目のサーバとなるノード上でのインストール例を掲載します。ここでは、ノード名は DECDNS とします。(途中表示されるメッセージは、一部省略してあります。)

```
$ @SYS$UPDATE:VMSINSTAL DNS011 DUA1:
```

```
=====
VAX Distributed Name Service Installation.
=====
```

```
%DNS-I-LICCHECK, Checking for DNS license on this system
Product:      DNS
Producer:     DEC
Version:      1.1
Release Date: 01-MAY-1988
```

* Does this product have an authorization key registered and loaded? YES

* Do you want to run the IVP after the installation [YES]?

```
=====
VAX Distributed Name Service Site Specific Questions
=====
```

- * Is this a new namespace [NO]? YES
(1 台目のサーバの時は YES を、2 台目以降は NO を指定してください。)
- * Namespace Name [DECDNS_NS]:
(ネームスペース名を入力します。)
- * Timezone (+hh:mm or -hh:mm): +9:00
(Greenwich Mean Time [GMT] との時間差を入力します。日本は +9:00 です。)
- * Clearinghouse Name [DECDNS_CH]:
(クリアリングハウス名を入力します。)
- * Enter UIC for DNS server account (include brackets) [[371,371]]:
- * Enter password for the DNS server account:
- * Verify password for the DNS server account:
(DECdns サーバが使用するアカウント DNS\$SERVER の UIC とパスワードを指定します。)
- * Device to be used for storing DNS\$SERVER files [SYS\$SYSROOT]:
(クリアリングハウスに関する情報を格納するディスク名を指定します。)

This installation has added the following files:

```
[SYSEXEC]DNS$ADVER.EXE      - DNS advertisement process.
[SYSEXEC]DNS$BACK.EXE      - DNS background process.
[SYSEXEC]DNS$CONTROL.BIN   - DNS management command parsing table.
[SYSEXEC]DNS$CONTROL.EXE   - DNS control program.
[SYSEXEC]DNS$DISPLAY_TREE.EXE - DNS namespace display utility.
[SYSEXEC]DNS$ML.EXE        - DNS management listener.
[SYSEXEC]DNS$SOLICIT.EXE   - DNS client solicitation process.
[SYSEXEC]DNS$TRANS.EXE     - DNS transaction process.
[SYSHLP]DNS$011.RELEASE_NOTES - DNS release notes.
[SYSHLP]DNS$HELP.HLB       - DNS management help library.
[SYSLIB]DNS$CLIENT.EXE     - DNS client shareable library.
[SYSLIB]DNS$LIB.EXE        - DNS shareable library.
[SYSMGR]DNS$CHANGE_DEF_FILE.COM - DNS change default server command file
[SYSMGR]DNS$CLIENT_STARTUP.COM - DNS client startup command file.
[SYSMGR]DNS$CLIENT_STOP.COM - DNS client shutdown command file.
[SYSMGR]DNS$STARTUP.COM    - DNS startup command file.
[SYSMGR]DNS$STOP.COM       - DNS shutdown command file.
[SYSMMSG]DNS$MSG.BIN       - DNS management text file.
[SYSMMSG]DNS$MSG.EXE       - DNS error message file.
[SYSTEST]DNS$IVP.COM       - DNS installation verification procedure.
```

```

=====
Starting VAX Distributed Name Service
=====

```

Creating clearinghouse, please wait ...

Clearinghouse filename _____ SYS\$SYSROOT:[DNS\$SERVER]decdns_ch.DNS

(後は、インフォメーションと IVP (Installation Verification Procedure) の実行でインストールは終了です。)

3. インストール後

DECdns サーバ・ソフトウェアのインストールが正常に終了した後、以下の事を実行してください。

- システム・スタートアップ・ファイルである SYS\$STARTUP:SYSTARTUP_V5.COM の修正。

キューマネージャの起動 (\$ START/QUEUE/MANAGER) と DECnet の起動 (@SYS\$MANAGER:STARTNET.COM) の後に、DECdns サーバを起動するコマンド @SYS\$MANAGER:DNS\$STARTUP.COM を追加してください。

- (必要なら) DECdns を管理するためのコマンド・インターフェース DNS\$CONTROL のシンボル定義。

起動は、

```

$ RUN SYS$SYSTEM:DNS$CONTROL
DNS> dns-command

```

でも可能ですが、フォーリン・コマンド定義をしておく便利です。

```

$ DNSCP ::= $DNS$CONTROL
$ DNSCP dns-command

```

- 新しいネームスペースへのアクセス権の変更

最初のネームスペースを作成する際、インストール中に以下のコマンドが実行されます。

```

ADD ACCESS *::* DIRECTORY .
                                /RIGHT=(READ,WRITE,DELETE,TEST,CONTROL)
ADD ACCESS *::* DIRECTORY . /FLAGS=(DEFAULT)
                                /RIGHT=(READ,WRITE,DELETE,TEST,CONTROL)

```

このコマンドにより、インストール後はルート・ディレクトリ内の全てのオブジェクトに対し、全ユーザがアクセス権を持っています。ネームスペースの内容を守るためにも、管理者は必要に応じてアクセス権の削除などを行ってください。

- 1 台目もしくは新規に作成したクリアリングハウスへのアクセス権の変更

サーバをインストールし、そのノード上で新たにクリアリングハウスを作成した場合、インストール中に以下のコマンドが実行されます。

```
ADD ACCESS *::* CLEARINGHOUSE clearinghouse-name  
/RIGHT=(READ,WRITE,DELETE,TEST,CONTROL)
```

管理者は必要に応じてユーザのアクセス権を削除するなど、クリアリングハウスへの制御を行ってください。

2.2 DECdns の起動と停止

質問：DECdns の起動と停止の方法を教えてください。

サーバ・ノード上での起動/停止方法

[起動]

```
@SYS$MANAGER:DNS$STARTUP.COM
```

このスタートアップ・プロシージャ内で DECdns のクライアントもスタートさせています。

システム立ち上げ時に自動的に DECdns サーバを起動させるためには、SYS\$STARTUP:SYSTARTUP_V5.COM を、キューマネージャと DECnet をスタートさせた後に上記コマンドが実行されるように変更してください。

[停止] (以下の順で行ってください。)

1. DECdns サーバの停止

```
@SYS$MANAGER:DNS$STOP.COM
```

2. DECdns クライアントの停止

```
@SYS$MANAGER:DNS$CLIENT_STOP.COM
```

または、

```
@SYS$STARTUP:DNS$CLERK_STOP.COM
```

クライアント・ノード上での起動/停止方法

[起動]

```
@SYS$MANAGER:DNS$CLIENT_STARTUP.COM
```

または、

```
@SYS$STARTUP:DNS$CLERK_STARTUP.COM
```

ただし、これらは DECdns を用いるアプリケーション (RSM, VAX DFS 等) のスタートアップ・プロシージャ内で自動的に実行されていますので、通常特に実行させる必要はありません。

VMS Version 5.2 より DECdns クライアント関係のファイルは VAX/VMS に標準に含まれています。もし、VAX DFS 等のアプリケーション・ソフトウェアがインストールされていない場合に、DECdns クライアントを起動させるためには以下のように行います。

1. SYS\$MANAGER:DNS\$CHANGE_DEF_FILE.COM を実行し、既存の DECdns サーバ・ノード名とネームスペース名を入力します。
2. SYS\$MANAGER:DNS\$CLIENT_STARTUP.COM を実行します。

[停止]

@SYS\$MANAGER:DNS\$CLIENT_STOP.COM

または、

@SYS\$STARTUP:DNS\$CLERK_STOP.COM

2.3 DECdns のプロセス

質問：DECdns で動く各プロセスの役割を教えてください。

DECdns サーバ・ソフトウェアを起動すると、以下の 3 つのプロセスが作成されます。

DNS\$TA (Transaction Agent process)

クライアントとコミュニケーションを行うプロセスです。
クライアントからのリクエストを受け付けたり、クレークに対して目的の名前を持つサーバへのルートを示したりします。また、ディレクトリの変更等ネームスペース内で構造の変更があった場合、サーバ間で同期を取った更新を DNS\$BACK に依頼します。

DNS\$BACK (Background process)

ディレクトリのコピーをアップデートするプロセスです。
名前の更新を行うアップデート・プロパゲイトとスカルク・オペレーションを実行します。
詳細は章 1.6 "名前の更新" の項をご参照ください。

DNS\$ADVER (Advertisement process)

クレークが全サーバに対して、自分の存在確認と名前の検索要求を出すソリシテーション (Solicitation) の機能と、サーバが全クレークに対して、自分がサーバであるという情報 (ネームスペース名、クリアリングハウスの情報、イーサネットアドレス、タイムスタンプ) を出すアドバタイズメント (Advertisement) の機能を実行します。

DNS\$ADVER はサーバ、クライアント共に存在するプロセスです。

2.4 DECdns マネージメント

質問 : DECdns を管理するコマンドを具体的に教えてください。

この章では、DECdns のサーバ管理者に知っておいていただきたい事を、具体的に DNS\$CONTROL コマンドを用いてご紹介いたします。

以下の内容について述べます。

1. DECdns 管理コマンド
2. ネームスペース内の情報表示
3. アクセス制御の管理
4. 階層化ネームスペースの設計
5. ネームスペース内での分散
6. ネームスペースの再構築

2.4.1 DECdns 管理コマンド

ネームスペースを管理するためのコマンド・インターフェースとして、VAX Distributed Name Service Control Program (DNS\$CONTROL) が提供されています。

DNS\$CONTROL の起動

```
$ RUN SYS$SYSTEM:DNS$CONTROL
DNS> dns-command
```

または、以下のようにフォーリン・コマンド定義をしておけば便利です。

```
$ DNSCP ::= $DNS$CONTROL
$ DNSCP
DNS> dns-command
あるいは、
$ DNSCP dns-command
```

DNS\$CONTROL の終了

DNS> のプロンプトで EXIT を入力するか、Ctrl/Z を押してください。

DNS\$CONTROL の詳細なコマンドにつきましては、マニュアル VAX Distributed Name Service Management Guide の第 4 章をご覧ください。

2.4.2 ネームスペース内の情報表示

ネームスペース自身またはネームスペース内に含まれるエントムの情報は、DNS\$CONTROL の SHOW コマンドで参照することができます。

ここでは、以下の情報を参照する方法について述べます。

- ネームスペース内のクリアリングハウス
- ディレクトリに含まれるチャイルド・ディレクトリ、オブジェクト、リンク(ソフトリンク)
- グループのメンバー
- エントリの詳細情報
- アクセス制御情報

A. ネームスペース内のクリアリングハウス

ネームスペース内のクリアリングハウスの情報を得るには、DNS\$CONTROL での SHOW KNOWN CLEARINGHOUSE コマンドか SHOW ACTIVE CLEARINGHOUSE コマンドを入力します。クリアリングハウス名が分かれば、SHOW CLEARINGHOUSE clearinghouse-name にてさらに詳細な情報が得られます。

```
DNS> SHOW KNOWN CLEARINGHOUSE
```

```
Clearinghouse name _____ DECDNS_CH
Clearinghouse state _____ On
Clearinghouse filename _____ SYS$SYSROOT:[DNS$SERVER]DECDNS_CH.DNS
RMS buffer count _____ 3
Directories in this clearinghouse _ 74
UID _____ aa-00-04-00-47-fc-40-1c-bf-b1-ad-0e-94-00
```

```
DNS> SHOW CLEARINGHOUSE DECDNS_CH
```

```
Clearinghouse name _____ DECDNS_CH
Clearinghouse state _____ On
Clearinghouse filename _____ SYS$SYSROOT:[DNS$SERVER]DECDNS_CH.DNS
RMS buffer count _____ 3
Directories in this clearinghouse _ 74
UID _____ aa-00-04-00-47-fc-40-1c-bf-b1-ad-0e-94-00
Next scheduled update _____ 22-JUL-1991 21:36:36.56
Time since last zeroed _____ 10 06:56:58.11
Security violation _____ 0
Tree broken _____ 0
Root lost _____ 0
Crucial replica _____ 0
Parent pointer _____ 0
Data corruption _____ 0
Possible cycle _____ 0
Skulk failed _____ 0
Propagate failed _____ 0
Lookups satisfied _____ 122280
Modifies satisfied _____ 155
```

Requests assisted _____ 0

B. ディレクトリに含まれるチャイルド・ディレクトリ オブジェクト リンク

指定されたディレクトリ内のチャイルド・ディレクトリ オブジェクト リンク(ソフトリンク)の名前を得るには、DNS\$CONTROL での SHOW DIRECTORY コマンドを入力します。

ディレクトリに含まれるチャイルド・ディレクトリの参照

```
DNS> SHOW DIRECTORY . KNOWN CHILDREN
Child _____ ENGINEERING
Child _____ MARKETING
Child _____ SALES
```

このコマンドによって、ルート・ディレクトリ以下に ENGINEERING, MARKETING, SALES の 3 つのチャイルド・ディレクトリがあることがわかります。

```
DNS> SHOW DIRECTORY .ENGINEERING KNOWN CHILDREN
Child _____ RSM
Child _____ DFS
```

このコマンドによって、ルート・ディレクトリ下のディレクトリ ENGINEERING にはさらに RSM, DFS というチャイルド・ディレクトリがあることがわかります。

ディレクトリに含まれるオブジェクトの参照

```
DNS> SHOW DIRECTORY .ENGINEERING.DFS KNOWN OBJECTS
Object _____ DISK_KIT1
Object _____ DISK_KIT2
Object _____ DISK_SOURCE
Object _____ DISK_USER
```

このコマンドによって、ENGINEERING.DFS というディレクトリには、DFS で用いるアクセス・ポイントを 4 つオブジェクトとして持っていることがわかります。

ディレクトリに含まれるリンク(ソフトリンク)の参照

```
DNS> SHOW DIRECTORY ENGINEERING.DFS KNOWN LINKS
No information
```

通常このコマンドによってリンク(ソフトリンク)名を表示しますが、現在リンクが存在しないため、上記メッセージが表示されました。

ソフトリンクとは、VMS の論理名のような役割をします。1 つのオブジェクトに複数の名前をつけることができます。たとえば、今まで用いていた .ABC という名前を .XYZ に変更する時、運用上直ちに変更できない場合は、移行期間中このソフトリンクの機能により .ABC と指定されても実際は .XYZ を指すように設定することができます。

リンク(ソフトリンク) 名が分かれば、DNS\$CONTROL での SHOW LINK コマンドでポイント先などの情報が得られます。

C. グループのメンバ

グループの参照は、DNS\$CONTROL での SHOW DIRECTORY dir-name KNOWN GROUP コマンドで参照できます。SHOW GROUP コマンドでは、属性、そのグループのメンバ名等が表示されます。

```
DNS> SHOW DIRECTORY . KNOWN GROUP
Object _____ ENGINEERING
Object _____ MARKETING
Object _____ SALES
```

```
DNS> SHOW GROUP ENGINEERING
Attribute (Set) _____ DNS$ACS
Attribute (Single) ____ DNS$Class
Attribute (Single) ____ DNS$ClassVersion
Attribute (Set) _____ DNS$Members
Attribute (Single) ____ DNS$UID
Attribute (Single) ____ DNS$UTS
```

```
DNS> SHOW GROUP ENGINEERING KNOWN MEMBER
Member ____ DECDNS::SYSTEM (DECDNS_NS:.DNS$IV.DECDNS.SYSTEM)
Member ____ DECPHV::SYSTEM (DECDNS_NS:.DNS$IV.DECPHV.SYSTEM)
Member ____ *::DNSMGR (DECDNS_NS:.DNS$IV.*.DNSMGR)
```

D. エンドの詳細情報

ディレクトリ、クリアリングハウス、リンク(ソフトリンク)、オブジェクト等につけられている属性(アドリビュート)のいくつかを参照するためには、DNS\$CONTROL では SHOW CHARACTERISTICS コマンドを入力します。

以下のコマンドは、クリアリングハウス内にあるディレクトリを表示します。

```
DNS> SHOW CHARACTERISTICS CLEARINGHOUSE DECDNS_CH
Directory _____ DECDNS_NS:.ENGINEERING
Directory _____ DECDNS_NS:.ENGINEERING.RSM
Directory _____ DECDNS_NS:.ENGINEERING.DFS
Directory _____ DECDNS_NS:.MARKETING
Directory _____ DECDNS_NS:.MARKETING.RSM
Directory _____ DECDNS_NS:.MARKETING.DFS
Directory _____ DECDNS_NS:.SALES
Directory _____ DECDNS_NS:.
```

以下のコマンドは、チャイルド・ディレクトリENGINEERINGの詳細情報を表示します。

```
DNS> SHOW CHARACTERISTICS DIRECTORY .ENGINEERING
Clearinghouses _____ ALLOWED
Convergence _____ HIGH
Last successful update __ 22-JUL-1991 00:37:06.56
```

Directory copies:

```
Address _____ DECDNS (1.1)
Replica Type _____ Master
Clearinghouse _____ DECDNS_NS:.DECDNS_CH
```

以下のコマンドはオブジェクトの存在するアドレス、クラス、バージョンを表示します。

```
DNS> SHOW CHARACTERISTICS OBJECT DECDNS_CH

Address _____ DECDNS (1.1)
Class _____ DNS$Clearinghouse
Version _____ V1.0
```

DNS\$CONTROLでのSHOW CHARACTERISTICSコマンドでは、属性の全てが表示されるわけではありません。このコマンドで表示されない属性についても、参照するコマンドが用意されています。

以下のコマンドはディレクトリのコンバージェンス属性を表示します。

```
DNS> SHOW DIRECTORY ENGINEERING ATTRIBUTE DNS$CONVERGENCE
Convergence _____ HIGH
Timestamp _ 10-DEC-1990 05:08:43.86 aa-00-04-00-47-fc
```

属性につきましては、マニュアル VAX Distributed Name Service Management Guide の Appendix B をご覧ください。

E. アクセス制御情報

クリアリングハウス、ディレクトリ、オブジェクト等に対するユーザのアクセス制御情報を参照するには、DNS\$CONTROLのSHOW ACCESSコマンドを入力します。アクセス制御の管理については次の項目をご参照ください。

```
DNS> SHOW ACCESS CLEARINGHOUSE DECDNS_CH

Entry _____ DECDNS::DNS$SERVER _ (DECDNS_NS:.DNS$IV.DECDNS.DNS$SERVER)
Rights _____ READ WRITE DELETE TEST CONTROL

Entry _____ *::* _ (DECDNS_NS:.DNS$IV.*.*)
Rights _____ READ WRITE DELETE TEST CONTROL
```

```
Entry _____ DECDNS::system _ (DECDNS_NS:.DNS$IV.DECDNS.system)
Rights _____ READ WRITE DELETE TEST CONTROL
```

```
DNS> SHOW ACCESS DIRECTORY ENGINEERING
```

```
Entry _____ DECDNS::DNS$SERVER _ (DECDNS_NS:.DNS$IV.DECDNS.DNS$SERVER)
Flags _____ NOPROPAGATE
Rights _____ READ WRITE DELETE TEST CONTROL
```

```
Entry _____ DECDNS_NS:.DECDNS_CH
Flags _____ NOPROPAGATE
Rights _____ READ WRITE DELETE TEST CONTROL
```

```
Entry _____ DECDNS::SYSTEM _ (DECDNS_NS:.DNS$IV.DECDNS.SYSTEM)
Flags _____ DEFAULT PROPAGATE
Rights _____ READ WRITE DELETE TEST CONTROL
```

```
Entry _____ DECDNS::SYSTEM _ (DECDNS_NS:.DNS$IV.DECDNS.SYSTEM)
Flags _____ NOPROPAGATE
Rights _____ READ WRITE DELETE TEST CONTROL
```

```
Entry _____ *::* _ (DECDNS_NS:.DNS$IV.*.*)
Flags _____ NOPROPAGATE
Rights _____ READ
```

```
Entry _____ *::* _ (DECDNS_NS:.DNS$IV.*.*)
Flags _____ DEFAULT PROPAGATE
Rights _____ READ
```

```
Entry _____ DECDNS_NS:.ENGINEERING
Flags _____ NOPROPAGATE GROUP
Rights _____ READ WRITE DELETE TEST CONTROL
```

```
Entry _____ DECDNS_NS:.ENGINEERING
Flags _____ DEFAULT PROPAGATE GROUP
Rights _____ READ WRITE DELETE TEST CONTROL
```

2.4.3 アクセス制御の管理

ネームサーバの管理者は、ネームスペース内の情報 (エントリ) にアクセス権を与えたり、あるいは取り除くことで、ユーザに情報を提供したり、セキュリティの点から情報を提供できなくしたりすることができます。

DECdns をインストールした後は、全てのユーザにネームスペース内の全てのエントリに対して READ,WRITE,DELETE,TEST,CONTROL のアクセス権が与えられています。必要に応じ

て変更してください。

アクセス権を正しく設定しないとネームスペースに対して、正しく参照できなかつたり 不正な変更が行なわれたりします。

ネームスペース管理者は十分考慮してアクセス制御を行ってください。

A. アクセス権の参照

ネームスペース内のエントリに対するアクセス権を参照するには、DNS\$CONTROL の SHOW ACCESS コマンドを入力してください。

```
DNS> SHOW ACCESS OBJECT .ENGINEERING.DFS.DISK_USER
```

```
Entry _____ *::* _ (DECDNS_NS:.DNS$IV.*.*)
Rights _____ READ
```

```
Entry _____ DECDNS::SYSTEM _ (DECDNS_NS:.DNS$IV.DECDNS.SYSTEM)
Rights _____ READ WRITE DELETE TEST CONTROL
```

上記例では、全ノードの全てのユーザは ENGINEERING.DFS.DISK_USER というオブジェクトを参照でき、ノード DECDNS の SYSTEM アカウントのユーザは参照、変更、削除、属性の参照、制御が許されていることを表しています。

アクセスの指定は VMS のノード名::ユーザ名という組み合わせになっています。または DECdns で取り決めたグループ名となっています。

アクセスの種類は以下の 5 つがあります。

READ	オブジェクトの参照
WRITE	アクセス制御を除く属性の変更とオブジェクトの作成
DELETE	オブジェクトの削除
TEST	オブジェクトに設定されている属性の参照
CONTROL	オブジェクトに関する全属性の制御

また、各オブジェクトにつけられるフラグには以下の 3 つがあります。

PROPAGATE	チャイルド・ディレクトリにそのアクセス権を継承
DEFAULT	ディレクトリに作られたオブジェクトにデフォルトで付くアクセス権
GROUP	ユーザがグループで定義されていることを示す

B. グループの管理

グループとは、オブジェクトに対してアクセス権を設定する際に、ある複数のユーザを 1 つのまとまりとして管理するためのものです。グループを作成して、グループ単位でアクセスの制限を行うことができます。

グループには、各ユーザをメンバー登録します。この方法はオブジェクトに対し、ユーザを個々に登録するより管理は楽になります。

1) グループの作成

グループの作成は DNS\$CONTROL での CREATE GROUP コマンドで行います。グループもディレクトリ内の 1 つのオブジェクトです。

```
DNS> CREATE GROUP ENGINEERING.PERSONNEL
  UID_____aa-00-04-00-8e-09-19-98-26-45-72-36-d2-f1
```

このコマンドで ENGINEERING.PERSONNEL というグループを新しく作成しました。ディレクトリ内の全てのグループを参照するためには、SHOW DIRECTORY dir-name KNOWN GROUP コマンドを実行してください。

2) グループへのメンバー登録と移動

既存のグループにメンバーを登録するには、以下のコマンドを入力します。

```
DNS> ADD MEMBER DECRSM::RSMUSER GROUP ENGINEERING.PERSONNEL
```

あるグループを他のグループに追加する場合は、以下のコマンドを入力します。

```
DNS> ADD MEMBER ENGINEERING.PERSONNEL GROUP OFFICE.PERSONNEL/FLAG=GROUP
```

3) グループ内メンバーの表示

グループ内のメンバーを参照するには、以下のコマンドを入力します。

```
DNS> SHOW GROUP ENGINEERING.PERSONNEL KNOWN MEMBERS
Member ____ DECDNS::SYSTEM (DECDNS_NS:.DNS$IV.DECDNS.SYSTEM)
Member ____ DECRSM::SYSTEM (DECDNS_NS:.DNS$IV.DECRSM.SYSTEM)
Member ____ DECRSM::RSMUSER (DECDNS_NS:.DNS$IV.DECRSM.RSMUSER)
Member ____ *::DNSMGR (DECDNS_NS:.DNS$IV.*.DNSMGR)
```

C. アクセス権の付与

あるエンディにアクセス権を与えるには、DNS\$CONTROL の ADD ACCESS コマンドを用います。

以下のコマンドは、ノード DECRSM の全ユーザにディレクトリ ENGINEERING.RSM に対して READ,WRITE のアクセス権を与えるコマンドです。/FLAG=DEFAULT を付けているため、そのディレクトリに新たに作成されたオブジェクト、グループ、リンクにも READ,WRITE のアクセス権は付加されます。もし DEFAULT フラグを省略した場合は、ディレクトリのエンディには付加されず、そのディレクトリにだけアクセス権が付きます。

```
DNS> ADD ACCESS DECRSM::* DIRECTORY ENGINEERING.RSM/FLAG=(DEFAULT)
  /RIGHT=(READ,WRITE)
```

同じエンディに対するアクセス権の設定は上書きされます。したがって以下のコマンドを入力後、

結局付くアクセス権は DELETE 権だけです。

```
DNS> ADD ACCESS DECDFS::* DIRECTORY ENGINEERING.DFS/RIGHT=(READ,WRITE)
DNS> ADD ACCESS DECDFS::* DIRECTORY ENGINEERING.DFS/RIGHT=(DELETE)
```

D. アクセス権の削除

ユーザに与えられたアクセス権を削除するには、DNS\$CONTROL の REMOVE ACCESS コマンドを用います。

```
DNS> REMOVE ACCESS DECRSM::* DIRECTORY ENGINEERING.RSM/FLAG=(DEFAULT)
```

上記コマンドは、ノード DECRSM の全ユーザに対して与えられていた ENGINEERING.RSM ディレクトリへのアクセス権を削除するコマンドです。

DEFAULT フラグ付きで設定されたアクセス権を削除する時には、/DEFAULT を指定して REMOVE ACCESS コマンドを実行しなければなりません。

2.4.4 階層化ネームスペースの設計

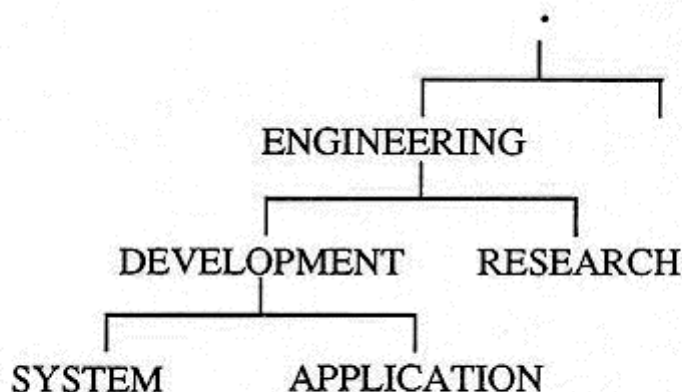
ネットワーク規模等により、オブジェクトの分散や名前管理の面において、ネームスペース内に複数レベルのディレクトリを作成する場合があります。

DNS\$CONTROL コマンドの CREATE DIRECTORY コマンドにて、ルート・ディレクトリ下にディレクトリを作成できます (ルート・ディレクトリは DECdns インストール時に作成されます)。作成されたディレクトリはマスタ・ディレクトリとなります。

ネットワーク内に 1 つだけ DECdns サーバがある時、全てのディレクトリは 1 つのクリアリングハウス内に構成されます。また、DECdns サーバが複数ある場合は、階層化されたディレクトリは複数のクリアリングハウスにまたがって配置されます。

1) シングル・サーバ環境下の階層型ネームスペースの作成

図 1 のように構成する時のコマンドを以下に示します。



- 図 1 -

```
DNS> CREATE DIRECTORY .ENGINEERING
DNS> CREATE DIRECTORY .ENGINEERING.RESEARCH
DNS> CREATE DIRECTORY .ENGINEERING.DEVELOPMENT
DNS> CREATE DIRECTORY .ENGINEERING.DEVELOPMENT.SYSTEM
DNS> CREATE DIRECTORY .ENGINEERING.DEVELOPMENT.APPLICATIONS
```

2) マルチ・サーバ環境下の階層型ネームスペースの作成

次項の図 2 のように構成する時のコメントを以下に示します。

```
DNS> CREATE DIRECTORY .ENGINEERING CLEARINGHOUSE MYNODE_CH
DNS> CREATE DIRECTORY .ENGINEERING.RESEARCH
DNS> CREATE DIRECTORY .ENGINEERING.RESEARCH.HARDWARE CLEARINGHOUSE
THEIRNODE_CH
DNS> CREATE DIRECTORY .ENGINEERING.DEVELOPMENT
DNS> CREATE DIRECTORY .ENGINEERING.DEVELOPMENT.SYSTEM
DNS> CREATE DIRECTORY .ENGINEERING.DEVELOPMENT.APPLICATIONS CLEARINGHOUSE
YOURNODE_CH
```

2.4.5 ネームスペース内での分散

ネットワーク内の複数のノードに DECdns サーバ・ソフトウェアをインストールすると、各サーバ・ノードはそれぞれクリアリングハウスを 1 つ持ちます。ディレクトリを複数のクリアリングハウスに分散することによって、オブジェクトの可用性やパフォーマンスの向上を図ることができます。

イーサネットの LAN には少なくとも 1 台の DECdns サーバを、また 100 ノード以上のネットワークでは、複数の DECdns サーバを置いてください。

また、ネームスペースを複数のクリアリングハウスに分けても、あるディレクトリを 1 つのサーバしか持たなければ、そのノードに障害があった場合、そのディレクトリ下の名前は参照できなくなってしまいます。それを防ぐためにレプリカを作成することができます。

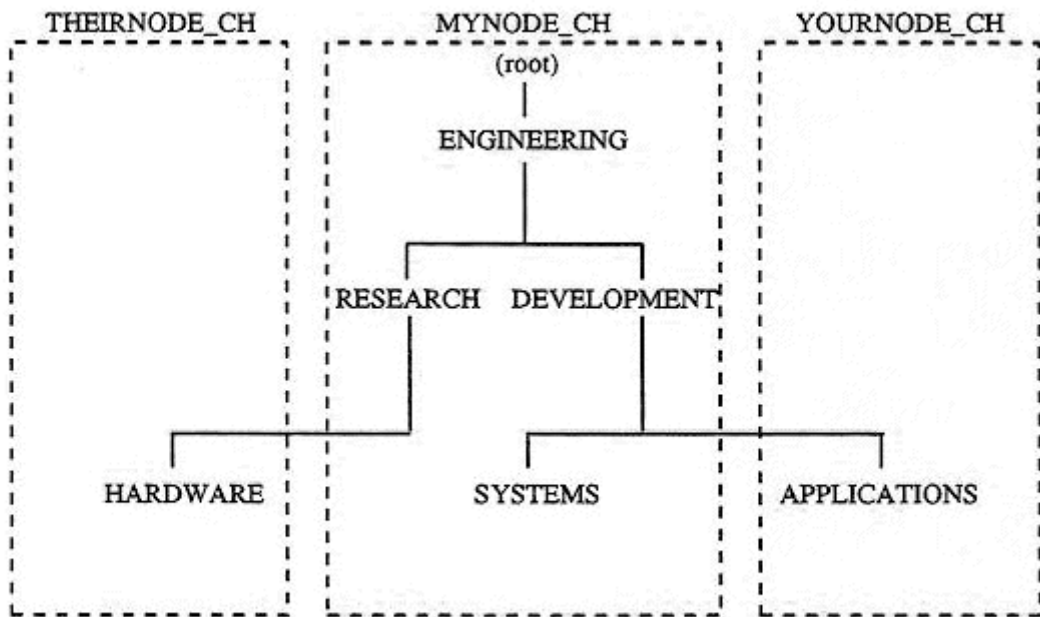
DECdns のディレクトリには、マスタとリードオンリーの 2 つのタイプのレプリカがあることは先に説明いたしました。

DNS\$CONTROL の CREATE DIRECTORY コマンドで作成したディレクトリは、マスタ・ディレクトリとなり、ディレクトリ内に対するすべての変更はこのマスタ・ディレクトリに対して行なわれます。

また、DNS\$CONTROL の COPY DIRECTORY コマンドによって、マスタ・ディレクトリのコピーであるリードオンリー・ディレクトリを他のクリアリングハウスに作成することができます。リードオンリー・ディレクトリは参照のみ許されるディレクトリですので、ユーザは変更 (作成、削除等) できません。

リードオンリー・ディレクトリは以下のコマンドで作成できます。

```
DNS> COPY DIRECTORY dir-name CLEARINGHOUSE ch-name
```



- 2 -

第 3 章 DNS プランニング

この章では、DNS を設計する上でのガイドラインをご説明いたします。

3.1 名前の付け方

質問：DECdns での名前の付け方はどのようにすればいいのでしょうか？

DECdns では

- ネームスペース
- クリアリングハウス
- ディレクトリ
- オブジェクト

の名前を決めなければなりません。それぞれの名前を付けるときの注意点をまとめてみました。

ネームスペース

- 長さは 255 文字まで。
- なるべく短く分かりやすい名前にします。ネットワーク全体で意味のある名前がいいでしょう。会社名でもかまいません。
- ネームスペースはなるべくネットワーク全体で 1 つにしてください。
- ネームスペースの名前は変更しにくいので、作成する前に慎重に選んでください。

クリアリングハウス

- クリアリングハウスだということがわかるように _CH を名前に入れてください。
- ネットワーク全体で 100 ノード以下であれば、全てのクリアリングハウスの名前をルート・ディレクトリに置いてください。
- 100 ノード以上の大きなネットワークでは、いくつかのクリアリングハウスを階層構造の低いところにおいた方が資源の分散になります。

ディレクトリ オブジェクト

- DECdns のフルネームは、(シンプルネームを区切るピリオドも含んで) 255 文字まで。
- 大文字、小文字のあらゆる組み合わせが可能です。しかし、検索する時は大文字、小文字を区別しません。(VMS の場合)
- 名前は意味があり ユーザに覚えやすいものにするべきでしょう
- 唯一性を失わない程度に短くて、実用的な名前をつけるとよいでしょう
- ディレクトリ名前は安定していて、会社の組織変更などの影響を受けないものにします。

3.2 ディレクトリの設定

質問：ディレクトリはどのように設計したらよいのでしょうか？

ディレクトリ構造のプランニング・ガイドラインを以下に示します。

階層構造にする必要があるかどうか

- 小規模なネットワーク(25 ノード未満)では必要ありません。すべての名前をルートに置きます。
- 大規模なネットワーク(25 ノード以上)では必要です。

階層構造を使用するメリット

- オブジェクトのグループが、どこで使用されるか、どのくらいの頻度で使用されるか、だれから使用されるかなどに基づいて分けることができます。
- 名前が重複する可能性を減らすことができます。
- ディレクトリ オブジェクトの単位でアクセス制御や管理を委任することができます。

階層のレベル

- 4 階層以上にすることはお勧めしません。
- 大規模ネットワークにおいては、下の階層は 1 台のサーバのみに置か、または複数のサーバに作成する場合も最小限にとどめるようにします。
- それ以上のレベルにすることにより多くのオーバーヘッドが発生します。

情報のアップデートにかかる処理時間が長くなります。
管理作業が多くなります。
名前が長くなり覚えられなくなります。

3.3 サーバの設定

質問：ネットワーク上のどのノードを DECdns サーバにすればいいのでしょうか？

DECdns サーバの選択をする時には次のことに注意してください。

- ダウンタイムが短く、すぐに再スタートできる安定したシステムで、十分なメモリやディスク容量のあるものが望まれます。
- DECdns サーバ間は、信頼性のあるネットワークにします。DECdns がレプリカのある他のサーバと通信するので重要です。
- アプリケーションのプランにもとづいて、クリアリングハウス、ディレクトリの構成を考えます。

サーバの数のガイドラインを以下に示します。
必要な台数はネットワークの構成に依存します。

[LAN]

最低でも LAN に 2 つのサーバを置くようにします。

[拡張 LAN]

ブリッジが DECdns のマルチキャストをフィルタリングしない場合には、それぞれに 1 台ずつあれば十分です。

ブリッジがマルチキャストをフィルタリングする場合には、双方に 2 台ずつおくことをお勧めします。これは、DECdns ではサーバが他のサーバやクラークに対してマルチキャストメッセージを出して自分の存在を知らせるためです。

[WAN]

安定していない WAN 環境経由の接続はさけるようにします。

WAN の接続が失敗しても DECdns のデータにアクセスできるようにサーバを配置するべきです。

度々参照や変更が予想されるサイトにサーバを置くことで、WAN のトラフィックを減らすようにします。

第 4 章 DFS

この章では、DECdns を使用するソフトウェア製品の 1 つである、VAX DFS (VAX Distributed File Service) をご紹介いたします。

DECdns をクライアント・アプリケーションからどのように使用するかを中心に、DFS の機能紹介、DFS とDECdns の関係などをご説明いたします。

4.1 DFS とは

質問：DECdns で資源が共有できると思いますが、ディスクの共有ができるということでしょうか？

DECdns は "ネットワークワイドな名前" の登録/照会を提供するソフトウェアです。言い換えれば、資源を利用するクライアント・ソフトウェアが資源の場所を知るために DECdns を使用します。

ディスクの共有といった機能の提供は、DECdns を使用するアプリケーションが行います。

ディスクの共有のために DEC は VAX Distributed File Service (以下 DFS といいます。) というソフトウェアを提供しています。DFS は DECdns をディスクやディレクトリの場所を登録するために使用します。

これから DFS の概要と使用例をご紹介します、DECdns とそのクライアント・アプリケーションの関係をご説明いたします。

質問：DFS とはどのようなものですか？

- DFS は DECnet で接続されたノード同士で、相手のノードに存在するディスク上のファイルを仮想のローカル・ファイルとして、透過的にアクセスするためのオプション・ソフトウェアです。
- LAVc でのディスク I/O とほぼ同等の速度を実現し、DECnet でのリモートファイル・アクセスより高速です。
- DFS は、ネットワーク・ノード間のサーバ/クライアントの関係が基本になっています。

DFS サーバ	--	ディスクをネットワーク上に提供
DFS クライアント	--	DFS サーバのディスクを利用

DFS サーバ 及び DFS クライアントには、DFS ソフトウェアのインストールが必要になります。インストール時に、サーバあるいはクライアントを選択する必要があります。

- DFS サーバでは、DFS コントロール・プログラム (DFS\$CONTROL) を使用し、ネットワーク上に提供するディレクトリを "アクセスポイント" として登録します。
- DFS クライアントでは、DFS\$CONTROL の MOUNT コマンドにより、リモート・ノードのディレクトリ(アクセスポイント) に接続し、仮想ディスクとして使用できるようになります。
- アクセスポイントの名前は DECdns に登録されます。
DECdns を使用することにより、たとえネットワーク上のシステム構成が変わり、そのために物理的なデバイス名 (名前、場所など) が変更されても、ネットワークワイドに共通な名前 (アクセスポイント) をただ 1 つ変更するだけで構いません。

- DFS は、そのアクセスポイントの格納/管理を DECdns に委ねるため、DFS を実行させるためには、ネットワーク内のあるノードで、あらかじめ DECdns サーバ・ソフトウェアが起動されていないけません。
- DFS が使用する DNS クラークは DFS のキット内に含まれ、合わせてインストールされます。
- DFS ソフトウェアは、VMS 上でのみ使用できます。

4.2 DFS の設定方法

質問：ディスクを共有するための具体的な方法を教えてください。

4.2.1 アクセスポイント

アクセスポイントはサーバがクライアントに提供するファイル資源を表します。一般的に特定デバイス上の特定ディレクトリを示します。アクセスポイントはディスクのマスタ・ファイル・ディレクトリ (MFD) を指定することも、またサブ・ディレクトリを指定することもできます。

DFS サーバ・ノード上で DFS の管理者は利用者に対し、どのデバイスのどのディレクトリを提供するかを決めます。DFS\$CONTROL での ADD ACCESS_POINT コマンドにてアクセスポイントを設定できますが、このコマンドによりアクセスポイントは DFS サーバのデータベースとDECdns のネームスペースに格納されます。

```
$ RUN SYS$SYSTEM:DFS$CONTROL
DFS> ADD ACCESS_POINT access-point-name directory-name
```

一方、DFS クライアント上では、DFS\$CONTROL での MOUNT コマンドを発行することによりリモート・ノードのディスクをアクセス可能とします。

MOUNT コマンドによりDFS クライアント・デバイス (DFSCn) が仮想デバイスとして作成されます。

```
$ RUN SYS$SYSTEM:DFS$CONTROL
DFS> MOUNT access-point-name [logical-name] [/VOLUME_NAME=string]
                               [/DEVICE=DFSCn]
```

4.2.2 DFS 設定例

図 1 をご覧ください。

1. DFS サーバにおいて、アクセスポイントの登録を行います。

```
DFS> ADD ACCESS_POINT .TKO.DISK.KOALA   DUA0:[KOALA]
-----
```

アクセスポイント名 ディレクトリ名

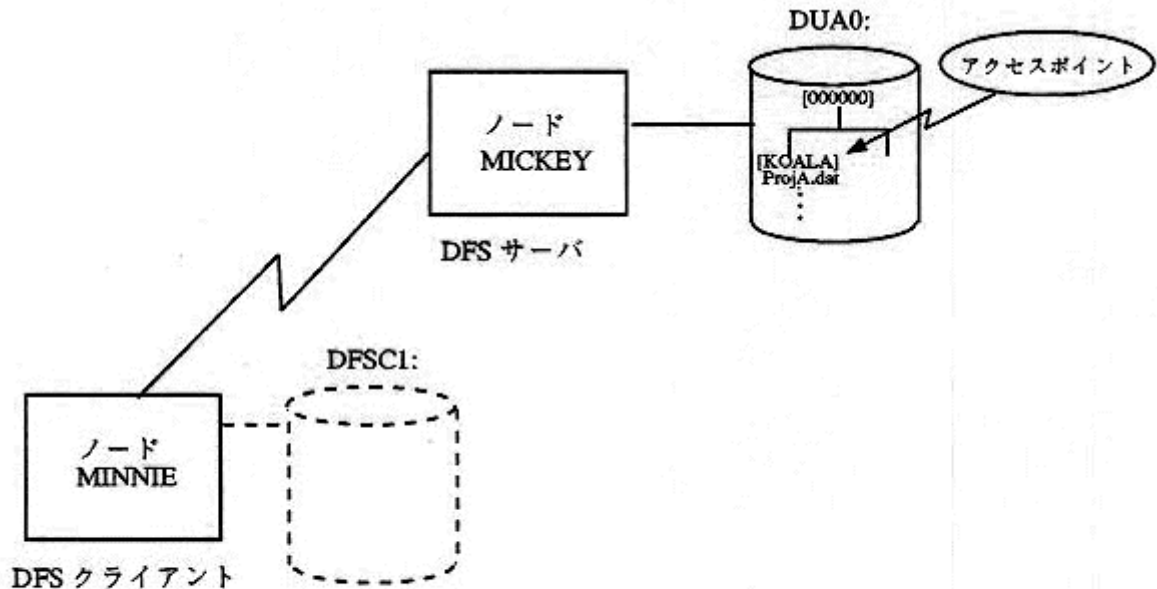
2. DFS クライアントにおいて、アクセスポイントへのマウントを行います。

```
DFS> MOUNT/SYSTEM .TKO.DISK.KOALA   KOALA   /DEVICE=DFSC1: /VOL=KOALA
-----
```

アクセスポイント 論理名 仮想デバイス名 ラベル

```
%MOUNT-I-MOUNTED, .TKO.DISK.KOALA mounted on _DFSC1: (MICKEY)
```

このメッセージは、DFS サーバ・ノードMICKEY のディスク DUA0: のディレクトリKOALA が、DFS クライアント・ノード上で 仮想デバイスDFSC1: としてマウントされたことを表しています。



-図 1-

4.2.3 DFS のディスクへのアクセス

ここでは、DFS でマウントされた仮想デバイスに対する、DCL コマンドでのオペレーションを説明いたします。

[デバイスの参照]

```
$ SHOW DEVICE DFSC
Device      Device      Error   Volume      Free      Trans      Mnt
Name        Status      Count   Label        Blocks    Count      Cnt
DFSC1:     Mounted     0       KOALA        *****  2          1
```

[ファイルへのアクセス]

デバイス名として以下の指定ができます。

- 仮想デバイス名 (この例では DFSC1:)
- マウントした時に設定した論理名 (この例では KOALA)

例)

```
$ DIR KOALA:[000000]
$ TYPE KOALA:[000000]A.DAT
```

```
$ CREATE/DIR KOALA:[TEMP]  
$ CREATE DFSC1:[TEMP]B.DAT
```

DFS デバイスのファイルは、VMS 標準の FILES-11 形式です。DFS で可能なオペレーションは以下のとおり 通常のファイル・アクセスとほとんど変わりません。

- ディレクトリの作成と操作
- リード・アクセスでのファイルの共有
- すべての RMS 機能とSY\$QIO インターフェース

ただし、次のような制限事項があります。

- 論理的、物理的な I/O はできません。
(ノウンイメージとしてインストールすることやページ・ファイル、スワップ・ファイルをおくことも含まれます。)
- 書き込みアクセスでの共有はできません。

4.3 DFS とDECdns

質問：DFS とDECdns の関係について教えてください。

前述しましたように、ディスクの共有のために DFS では、アクセスポイントをネットワーク上の資源として DECdns に登録し、使用します。

DFS と同じように、DECdns も資源を提供するサーバとそれを利用するクライアントの関係をとっています。DECdns の利用者として、DFS を使用する各ノードは DNS クライアントとも言えるでしょう。そしてファイルの場所についての情報を提供するのが DNS サーバです。

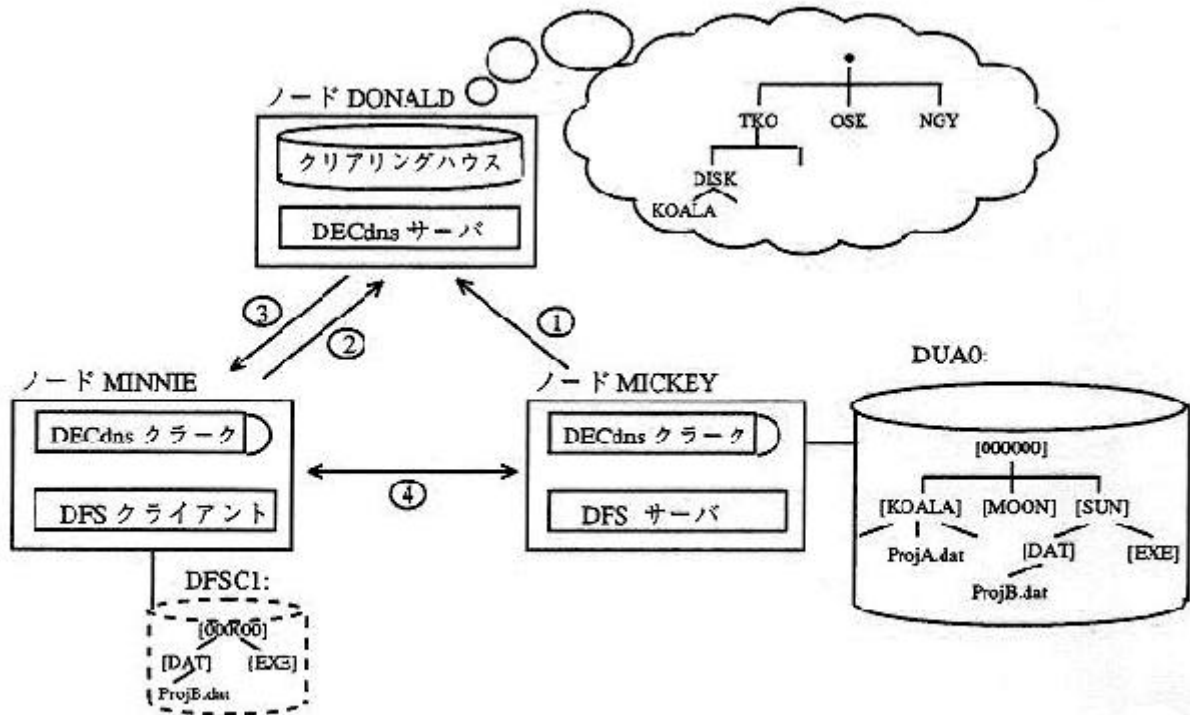
DFS とDNS サーバの動きを簡単に説明します。

1. DFS サーバがスタートすると、提供できるファイルを DNS サーバに知らせます。
2. DNS サーバはそのファイル名 (アクセスポイント) とDFS サーバの DECnet アドレスをデータベース内に保管します。
3. DFS クライアントからのアクセス要求に対し、DNS サーバは該当する DFS サーバの DECnet アドレスを教えます。
4. DFS クライアントは DFS サーバとリンクを張り、ファイル・アクセス等のコミュニケーションを行います。

以下に具体的な例を載せます。次項の図 2 をご参考ください。

[前提]

- この図では、ノードDONALD が DNS サーバとして、ネームスペースの一部を持っています。
- このネットワークは、東京、大阪、名古屋に拠点を置いていて、それぞれ異なる業務を行っているので TKO, OSK, NGY というロケーションごとのディレクトリを作り、それぞれの管理をすることにしました。
- ディスクのアクセスポイントは、各ロケーションごとに DISK というチャイルド・ディレクトリを作成し、その下におくことにしました。
- ノードMICKEY は、ディスク資源が他ノードよりはるかに多く、以前からネットワーク経由でファイルの作成、参照などを行っていました。これを DFS サーバに変更して、ディスクを提供します。
- そのディスクを利用するクライアントは、ここではノードMINNIE です。



-図 2-

[DFS の操作と内部的な動き]

(1) DFS サーバでアクセスポイントを登録します。

DFS サーバのデータベースおよび、DECdns ネームスペースの登録を行ないます。ネームスペースにそのアクセスポイントが既に登録されている時には、DFS サーバのデータベースへの登録のみ行ないます。

(2)(3) DFS クライアント・ノードがマウントのオペレーションを行ないます。

この時、DECdns クラークを通じて DECdns サーバにそのディスクの実際の場所を聞きます。

(4) 実際の DFS サーバが MICKEY だということがわかりましたので、マウント要求をします。

(5) マウントが成功した時には、DFSCn: という名前のデバイスが作成されます。

このディスクには、ローカルのディスクと同様なオペレーションでアクセスできます。

仮想ディスクにアクセスした時に、DFS\$COM_ACP というプロセス同士が DECnet のリンクを張ります。

4.4 アクセスポイントの参照

質問：アクセスポイントの参照方法を教えてください。

アクセスポイントを参照するには、以下の2つのコマンドがあります。

- DFS\$CONTROL の SHOW ACCESS_POINT コマンド
- DNS\$CONTROL の SHOW OBJECT コマンド

4.4.1 DFS\$CONTROL の SHOW ACCESS_POINT コマンド

DFS に関する管理は、DFS コントロール・プログラム (DFS\$CONTROL) を使用します。

設定されているアクセスポイントは、DFS\$CONTROL の SHOW ACCESS_POINT コマンドで参照できます。このコマンドは DFS サーバ/クライアントどちらのノード上でも実行できます。

ただしアクセスポイントを登録/削除する ADD / REMOVE ACCESS_POINT コマンドは DFS サーバ・ノード上でのみ実行可能です。

1. そのノード自体が提供しているアクセスポイントの参照

```
$ RUN SYS$SYSTEM:DFS$CONTROL
DFS> SHOW ACCESS_POINT .TKO.DISK.* /LOCAL
```

```
.TKO.DISK.SUN on MICKEY::DUA0:[SUN]
.TKO.DISK.KOALA on MICKEY::DUA0:[KOALA]
```

(説明)

ローカルノードMICKEYでは、DUA0:[SUN]とDUA0:[KOALA]を、それぞれ.TKO.DISK.SUN、.TKO.DISK.KOALAという名前(アクセスポイント)で提供しています。

2. すべてのDFSサーバにより提供されてるアクセスポイントの参照

```
$ RUN SYS$SYSTEM:DFS$CONTROL
DFS> SHOW ACCESS_POINT .TKO.DISK.* /FULL
```

```
DISNEY_NS:.TKO.DISK.SUN on MICKEY::DUA0:[SUN]
DISNEY_NS:.TKO.DISK.KOALA on MICKEY::DUA0:[KOALA]
DISNEY_NS:.TKO.DISK.DRAW on GOOFY:: Access point is not presently
                                     being served
```

(説明)

.TKO.DISK.DRAWというアクセスポイントは、DFSサーバがノードGOOFYでスタートしていないため使用できないことを示しています。

4.4.2 DNS\$CONTROL の SHOW OBJECT コマンド

DECdns に関する管理は、DNS コントロール・プログラム (DNS\$CONTROL) を使用します。

DFS で用いるアクセスポイントもDECdns のネームスペース内では 1 つのオブジェクトです。ネームスペースのオブジェクト情報は、DNS\$CONTROL の SHOW OBJECT コマンドにより参照できます。

以下にノードMIKCEY が提供しているアクセスポイントKOALA の例を載せます。

1. オブジェクトの参照

```
$ RUN SYS$SYSTEM:DNS$CONTROL
DNS> SHOW OBJECT .TKO.DISK.KOALA
```

```
Attribute (Set) _____ DNS$ACS
Attribute (Set) _____ DNS$Address
Attribute (Single) ____ DNS$Class
Attribute (Single) ____ DNS$ClassVersion
Attribute (Single) ____ DNS$UID
Attribute (Single) ____ DNS$UTS
```

(説明)

オブジェクト.TKO.DISK.KOALA の属性を表示しています。

2. オブジェクトのロケーションの参照

```
$ RUN SYS$SYSTEM:DNS$CONTROL
DNS> SHOW OBJECT .TKO.DISK.KOALA ATTRIBUTE DNS$ADDRESS
```

```
Protocol ID _____ DNA$ProtID$DFS_svc_agent,Identifier_1.1
Protocol Task _____ DNA$ProtID$TASK _ DFS$COM_ACP
Protocol Session Control_ DNA$ProtID$SessCtrlv2
Transport _____ DNA$ProtID$NSP
Routing _____ DNA$ProtID$Routingv2
Address _____ MICKEY (1.2)
Timestamp _ 1-FEB-1991 05:36:10.50 aa-00-04-00-01-04
```

(説明)

オブジェクト.TKO.DISK.KOALA のひとつの属性 DNS\$ADDRESS の情報を表示しています。ここに、このオブジェクトを持つノードが MICKEY(1.2) であることが記録されています。

このような属性の設定は、DFS が内部的に行なっています。

3. オブジェクトのクラスの参照

```
$ RUN SYS$SYSTEM:DNS$CONTROL  
DNS> SHOW OBJECT .TKO.DISK.KOALA ATTRIBUTE DNS$CLASS
```

```
Class _____ DFS$AccessPoint  
Timestamp _ 1-FEB-1991 05:36:10.12 aa-00-04-00-01-04
```

(説明)

オブジェクト.TKO.DISK.KOALA の 1 つの属性 DNS\$CLASS を表示しています。
DNS\$CLASS にはそのオブジェクトを使用するクライアントを識別する情報が入ります。こ
の例は、DFS のアクセスポイントであることを表しています。

4.5 アクセス制御

質問：DFS でのファイルに対するセキュリティはどのようになっていますか？

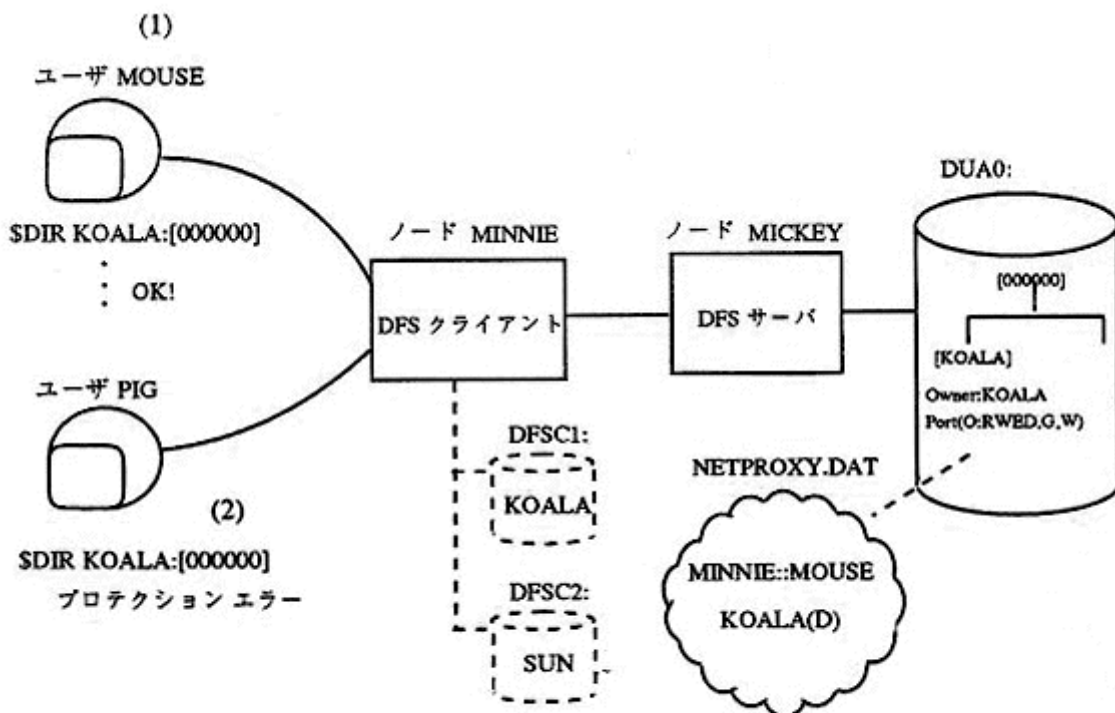
まず DECdns で設定されているアクセスポイントにアクセスできるかどうか、その次にサーバが提供するディスク、ファイルのプロテクションがアクセスを許しているかどうかによって制御されます。

1. DECdns ネームスペースに対するアクセス制御は、DECdns のセキュリティ機能を利用して行ないます。(章 2.5.3 "アクセス制御の管理" をご覧ください。)
2. 共有ディスク上のディレクトリ、ファイルに対するアクセスは、要求を出したクライアントのユーザ名で行なうのではなく、DECnet のプロキシ (Proxy) で定義されているサーバ・ノード上のユーザで実行されます。

また、DFS はプロキシが定義されていないユーザに対しては、DFS\$DEFAULT というユーザを作成しておけば、デフォルトのユーザとしてファイルをアクセスさせることができます。

ファイルの参照や作成もすべて、DFS サーバ・ノード上のアカウントで行なわれます。ファイル・アクセスの際はそのファイルのオーナーが誰なのかということを十分考慮してください。

ACL についてはクライアントからのアクセスを制御するために DFS\$SERVICE と NETWORK の 2 つの識別子を用いることができます。クライアントからは ACL を操作したり参照したりすることはできません。



-図 3-

[アクセス制御の実際例]

図 3 を見ながら次をお読みください。

DFS クライアントMINNIE からユーザが次のコマンドで DFS ディスクをアクセスした場合を例とします。

```
$ DIR KOALA:[000000]
```

- アクセスするKOALA というディスクは、実際にはノードMICKEY 上の DUA0:[KOALA] だとします。
- このディレクトリのオーナーは [KOALA] で UIC は [100,1] とします。
- プロテクションは (S:RWED,O:RWED,G:;W;) とします。
- AUTHORIZE ユーティリティで、DFS デフォルト・ユーザ DFS\$DEFAULT を UIC [356,1] で DECNET アカウントをコピーして作成してあります。

1. PROXY 定義のあるユーザ (MOUSE) からのアクセスは成功します。

要求を受けた DFS サーバ・ノードMICKEY では、まず PROXY データベース (SYS\$SYSTEM:NETPROXY.DAT) を参照します。

ここでは MINNIE::MOUSE からのアクセスを KOALA で代行するように定義がされています。

結果、DUA0:[KOALA] のオーナー [KOALA] としてアクセスし、参照できました。

2. PROXY 定義のないユーザ (PIG) からのアクセスは、NOPRIV というエラーになります。

要求を受けた DFS サーバ・ノードMICKEY では、まず PROXY データベース (SYS\$SYSTEM:NETPROXY.DAT) を参照します。

ここでは MINNIE::PIG からのアクセスを代行する設定がありません。

この場合、デフォルト・ユーザの DFS\$DEFAULT が代行します。

結果、DUA0:[KOALA] には World のカテゴリでアクセスしたため、プロテクションに違反しました。

もし、DUA0:[KOALA] のプロテクションが (S:RWED,O:RWED,G:RE,W:RE) であればアクセスできるということになります。

4.6 その他

4.6.1 パフォーマンス

十分なチューニングを行なった場合には、DAP (DECnet のファイル転送) と比べて 3 ~ 4 倍のスループットを得られるという結果がでています。

DFS ディスクへのパフォーマンスには次のようなことが影響します。

- DFS サーバの CPU パワー
- DFS サーバのコミュニケーション・デバイスの性能
- DFS サーバとDFS クライアントの通信経路
- 共有ディスクデバイスの性能

4.6.2 クラスタエリアスの指定

DFS サーバがクラスタシステムであれば、クラスタエリアス名で共有ディスクを提供することも可能です。アクセスポイントの登録時に /CLUSTER_ALIAS 修飾子を付けてください。

```
$ RUN SYS$SYSTEM:DFS$CONTROL
DFS> ADD ACCESS_POINT access-point-name directory-name /CLUSTER_ALIAS
```

第 2 部 CS セミナー Q & A

今年 7 月に全国 4 ヶ所で開催されました VMS Customer Services Seminar では、DEC の新しいネットワーク・アーキテクチャである DECnet/OSI Phase V についてご紹介させていただきました。

OSI を取り込んだ、この Phase V につきまして、皆様よりたくさんのご質問をいただきました。

この第 2 部では、一部ではありますが、以下の項目に分け、ご説明させていただきます。ご参考にしていただければ幸いです。

1. Phase V
2. OSI
3. ULTRIX & TCP/IP
4. SNA
5. DECMcc

尚、DSIN にもこの情報は登録されます。合わせてご利用ください。

2.1 Phase V

質問：DECnet-VAX V5.4 Extensions のインストールはどのように行いますか？

DECnet-VAX V5.4 Extensions は通常のレイヤード・ソフトウェアと同様に VMSINSTAL ユーティリティを用いてインストールを行います。

インストール時は、使用するオプション・ソフトウェアを選択するようになっていますが、それ以外の設定に関する事項に対しての入力は特にありませんので、簡単にインストールすることができます。

各コンポーネントに対する設定はそれぞれのコンフィグレーション・プロシージャを用いて行いません。

質問：Phase V に対応する VMS のバージョンはいくつになりますか？

最初の VAX/VMS 向け Phase V 製品は DECnet-VAX V5.4 Extensions です。この製品は VMS のバージョンが V5.4 以上である必要があります。

質問：Phase V にする事によりどのようなメリットがありますか？

DECnet Phase V は、将来必ずコンピュータネットワーク技術の世界標準となる OSI の仕様を先取りした形で採用しています。したがって、今までのように独自のネットワーク技術の製品があった上に、また別に異機種間で通信を行うために OSI 製品を購入するといった二重投資の必要がなくなります。将来、DEC 以外のベンダ各社からの OSI 製品が普及してくればその効果は更に上がります。

また Phase V ではアプリケーション・インターフェースも国際標準化されていますので、今後 Phase V を採用しているシステム上で開発されるネットワーク・アプリケーションの移植性が高まり、アプリケーションを機種毎に開発していた今までより投資効果が上がります。

その他、Phase V ではネットワークを効率よく機能させるための技術がたくさん採用されています。したがって非常に大規模なネットワークから小規模なネットワークまでバランスの取れたネットワークを自由に構築する事ができます。

質問：今後 Phase IV はサポートされなくなりますか？

今のところ、DECnet Phase IV のサポートを止める予定はありませんが、機能拡張は行なわれません。DECnet の Phase III から IV への変更の時と同じように、Phase IV から V への移行後もしばらく Phase IV は引き続いてサポートされます。しかし将来的には新しい製品は全て Phase V が標準になりますが、この場合でも Phase IV との互換性を持たせるようにしてあります。

質問：Phase IV では OSI を使用できませんか？

Phase IV のシステム上でも OSI 製品を使用することができます。

(注) オペレーティングシステムは VMS に限ります。

次に、OSI 製品を簡単にご紹介します。

- VAX OSI トランスポートサービス (VOTS)

ISO 8073 に準拠したトランスポートサービスを提供します。トランスポートクラス 0, 2, 4 をサポートし、LAN および WAN におけるエンドシステム・ルーティング機能を提供します。

- VAX OSI アプリケーション・カーネル (OSAK)

ISO が制定している OSI 参照モデルのうち、第 5～7 層で定義されたプロトコルを提供します。ISO 8327 で規定されているセッション層バージョン 1 に対応しています。

- メッセージルータ X.400 ゲートウェイ

CCITT で 1984年に承認された X.400 標準規格に基づいて、異なるメーカーの電子メールシステムと接続することができます。また X.400 をサポートしている PTT および公衆サービス提供者との間で、電子メールの交換を可能にします。

- VAX FTAM

ISO 8571 に基づいて、異機種間でのファイル転送、ファイルアクセス、ファイル管理を行うことができます。FTAM-1、FTAM-2、FTAM-3 の 3 つのドキュメント型をサポートしています。

質問：NSAP アドレスと IP アドレスの関係はどのようになりますか？

NSAP アドレスは DECnet Phase V の OSI プロトコルのためにアサインされたアドレスです。したがって TCP/IP の IP アドレスとは全く別のものでそれぞれが混在しても問題はありません。

質問：コネクションレスのトランスポートはサポートされますか？

現在のところ Phase V のアーキテクチャには ISO 8602 (コネクションレスのトランスポート) を取り入れていません。将来この機能が一般的になるようであれば DEC は積極的にこの機能を組み込んでいきます。

質問：VAX P.S.I. は Phase V に含まれますか？

VAX P.S.I. は Phase V になっても含まれません。別途ライセンスを必要とします。ただし VAX P.S.I. Access は Phase V に含まれます。VAX P.S.I. Access を用いて、網に直接接続されている VAX P.S.I. または、X.25 Gateway を経由して間接的に網にアクセスする事は可能です。

質問：Phase IV と Phase V でルーティング・アルゴリズムはどのように変更されるのですか？

Phase V ではリンクステイトと呼ばれるルーティング・アルゴリズムを採用しています。これは DEC が原案を作成し、現在 DIS 10589 になっています。まもなく IS 版になる予定です。

Phase IV のルーティング・アルゴリズムはディスタンス・ベクター又は "Bellman-Ford" と呼ばれ、ノードは隣接ノードから受信したネットワークのルーティングの情報（コストとホップ）を元に自分のノードからのコストとホップを計算します。それからその計算したルーティング情報を隣接ノードに送信します。

Phase V で使用されるリンクステイト・アルゴリズムでは、ノードは隣接ノードの ID とそのノードへのコストをルーティング情報としてネットワーク全体へブロードキャストします。各ノードではそのトポロジー情報を受信してトポロジーマップを作ります。

Phase IV と V のルーティング・アルゴリズムを比較すると以下のようになります。

	Phase IV	Phase V
アルゴリズム	ディスタンス・ベクタ	リンクステイト
利点	アルゴリズム・デザインが簡単	大きなサイズのネットワークの構成が可能 サーキットの状態の変化の伝搬が速い ルーティング情報のループがない
欠点	ルーティング情報がループすることがある	アルゴリズムのデザインが複雑

質問：動的ルーティングとはどのような仕組みで動作しているのですか？

ネットワーク内で、ある経路が使用できなくなった場合には自動的に別の経路（迂回路）を選択し元の経路が復旧するまでは、迂回路を使用してノード間のデータ転送を可能にします。この動作を動的ルーティングと呼びます。

DECnet では、データの中継機能を持つルーティングノードが各々の経路に関する情報（ルーティング情報）を交換し合ってネットワーク内の経路選択を行います。ある経路が使用不可能に

なると新たなルーティング情報を交換しそれを元に迂回路を決定しています。元の経路が復旧した時や、新たに経路ができた場合にもそれに応じたルーティング情報を交換して、常に最適な経路を選択できるように動的ルーティングは動作します。

質問：専用ルータとは何ですか？

いままでホストが行ってきたデータの中継転送 (ルーティング) 機能をすべて専用のハードウェアである WANrouter が行うこととなります。このことによりホストは負荷が軽減されます。

質問：Phase V ではホストベースのルーティング機能はどうなりますか？

Phase IV から Phase V への移行の最初のステップとして提供される DECnet-VAX V5.4 Extensions では、実際には Phase IV のノードと同様に動作します。このため DECnet-VAX V5.4 Extensions を使用している時には Phase IV としてのホストベースのルーティング機能だけは使用できます。

Phase V では、ホストにおけるルーティング情報の処理にかかる負担を無くすためにルーティング機能は専用ルータにて行なわれます。Phase V に移行後は、ホストベースのルーティング機能はなくなります。

将来的にはホストにルーティング機能を持たせる事も計画されていますが、詳細な日程や対応する OS のバージョン等は明らかではありません。

質問：WANrouter を購入するとTransLAN は不要となりますか？

TransLAN は拡張 LAN を構成するブリッジです。物理層とデータリンク層を担当し、LAN と LAN を接続します。たとえばターミナルサーバからあたかも同一 LAN にあるようにホストにアクセスする事が可能です。WANrouter はその上のネットワーク層までを担当し、データの中継転送 (ルーティング) 機能を行っています。つまりノード間でデータの中継転送を行っています。

WANrouter はマルチプロトコル (DECnet, OSI, TCP/IP) をサポートしますので、今までお使いのネットワークにさらに柔軟性は生まれます。したがって現在 TransLAN をこれらのプロトコル (DECnet, TCP/IP) だけでお使いの場合、WANrouter に置き換える事は可能です。ただし、これ以外のプロトコル (たとえばターミナルサーバからホストにアクセスするような LAT) をお使いの場合は、TransLAN のようなブリッジが必要です。

質問：リピータ、ブリッジ、ルータ、ゲートウェイの違いは何ですか？

リピータは物理層を接続する機器です。500m の長さのイーサネット・セグメントをリピータを使用して最大 2800m まで延長する事が可能です。リピータは一方のイーサネット・セグメントから受信したデータを他方のイーサネット・セグメントにそのまま送信します。

ブリッジは物理層とデータリンク層、ルーティング層を接続します。複数の LAN 同士を接続する時にルータが使用されます。ルータはルーティング層も担当するために特定のルーティング・プ

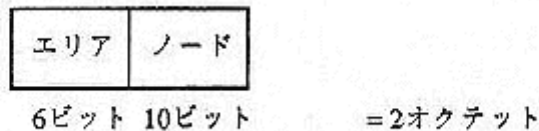
ロトコルを使用しているデータのみ送受信します。(例えば、DECnet や TCP/IP など)。これに対し、リピータ、ブリッジはルーティング・プロトコルは意識しないために DECnet や TCP/IP と違ったプロトコルとは無関係に動作します。

ゲートウェイは異なるプロトコルを変換する機器です。DECnet のデータと SNA のデータや TCP/IP のデータを相互変換します。例えば、DECnet と SNA のデータを変換する SNA Gateway は DECnet の方式で表記されているデータを SNA の方式に従ったデータ形式に変換します。このためゲートウェイは基本的に全ての層を使用します。

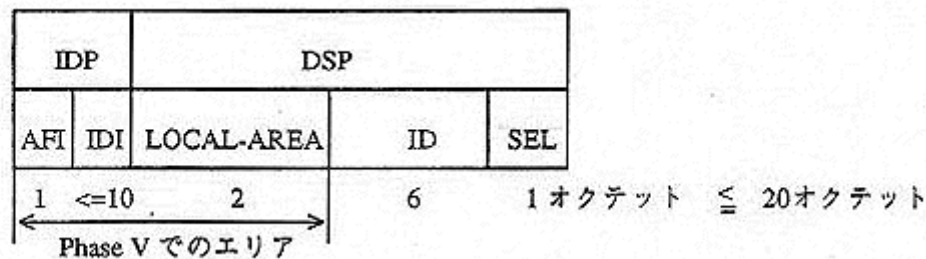
質問：Phase IV でのエリアの概念は Phase V でも使用できますか？

Phase V においてもエリアの概念は存在します。アドレス体系は以下のようになります。ルーティング情報はエリア内のレベル 1 とエリア間のレベル 2 の 2 種類が存在します。

Phase IV では



Phase V では



IDP とは Initial Domain Part の略

DSP とは Domain Specific Part の略

AFI とは Authority and Format Identifier の略

ID は通常イーサネットのハードウェア・アドレスから生成されます。

SEL とはトランスポートのプロトコルのセレクトラになります。

質問：互換アドレスがどうして Phase IV と互換があるのか教えてください。

互換アドレスとは Phase V の形式に従っており、なおかつ Phase IV のアドレスの制限にあったものです。Phase IV のアドレスの制限とは、エリアナンバーが 1 ~ 63、ノードナンバーが 1 ~ 1023 です。Phase V のノードと Phase IV のノードが通信する場合は、お互いが認識し合うためにこの互換アドレスを使用しなければなりません。

Phase V のルーティング方法を用いているエリアに存在する Phase V のノードは、Phase V のアドレスの表記方法を使用します。このノードが Phase IV のルーティング方法を用いている別のエリアのノードにデータを送信する時には、Phase V のエリアと Phase IV のエリアの境界で Phase V のアドレス表記方法から Phase IV の表記方法に変換されます (反対方向も同様に変換されます)。この様に互換アドレスを使用する事により Phase V のノードと Phase IV のノードの通信が可能になります。

質問：リーチャブル・アドレス・テーブルはどのように設定しますか？

方法は 3 つあります。

1. NET\$MIGRATE (SYS\$UPDATE:NET\$MIGRATE.EXE) という移行ツールを使用する。
このツールにてリーチャブル・アドレス・テーブルの初期化コマンドファイルを作成し、実行します。(SYS\$UPDATE:NET\$MIGRATE.EXE は、将来ツール名が変更されるかもしれません。マニュアル等をご参照ください。)
2. WANrouter のコンフィグレーションツールを使用する。
現在の DECrouter 2000 の ROUSETUP.COM と同等なコンフィグレーションツールによって設定できます。
3. NCL を使用する。
NCL によって WANrouter に対し直接設定できます。

質問：アドレスが重複しそうなネットワーク同士を接続する事は可能ですか？

Phase V のアドレス体系では、異なる IDP を使用する事により Phase IV では同一であったアドレスを結果として異なるアドレスとすることができます。したがってそれらのネットワーク同士を接続する事が可能となります。これは互換アドレスでも可能です。

さらに、完全に Phase V に移行されたネットワークでは拡張アドレス (Phase IV のアドレス制限を越えたアドレス) が使用できます。

この拡張アドレスでは、そのノード固有の値 (Ethernet ハードウェアアドレスなど) を使用しますので Phase IV では同一アドレスに定義されていたノード同士でも全く異なるアドレスを持ちます。これによりネットワーク内でのアドレスの重複は基本的になくなくなります。

質問：IDP の取得が必要となるのはどういう場合ですか？

OSI 環境で全てのシステムがユニークなアドレスを持ち、互いにコミュニケーションできるようにアドレスの形式が定義されています。このアドレス・フィールドの一部が IDP です。DECnet Phase V を用いて OSI ネットワークを構築し、他の DECnet 以外の OSI ネットワークと接続する場合は、固有の IDP が必要になります。

質問：IDP "49" を使用していて、他の OSI ドメインと接続するためにはどうすればよいのですか？

ネットワーク全体として接続性が必要な場合は、ユニークな IDP を取得します。一つのシステムが両方のネットワークのエンド・システムとして機能するだけなら必要ありません。

質問：DSP のアドレスフィールドとしてイーサネットを持たないシステムはどのように設定するのですか？

Phase V では、Phase V ルーティング・ドメイン内のアドレスとして、イーサネット・アドレスを用います。イーサネットのインターフェイスを持たないシステムは、Phase V ドメインでユニークなアドレスを割り振ることができます (手作業)。将来的には、DEC が提供するインターフェイスすべてがユニークなアドレスを持つことになる予定です。

質問：Phase V で使用できなくなる製品は何ですか？

以下に示します。(将来変更される可能性があります。)

- PDP-11 システム
- DECsystem-10、DECSYSTEM-20
- DMR11/DMV11 (DECnet-VAX の場合)
- KM シリーズのデバイス
- DECSAベースのルータ
- DEQNA (DECnet-VAX の場合)
- DECrouter200

質問：Phase V でサポートされない製品はどうなりますか？

Phase IV システムを残し、その中で使用します。互換アドレスを使用することで Phase V のシステムは存在でき、Phase IV と V のコミュニケーションも保証されます。

質問：DECdns/DECdts サーバは DECnet-VAX V5.4 Extensions に含まれますか？

DECdns サーバは、オプションのレイヤード・ソフトウェアです。VMS 版では DECdns サーバ V1.1 が、Ultrix 版では DECdns サーバ V2.0 が用意されています。

DECdts サーバは、DECnet-VAX V5.4 Extensions に含まれます。ただし、インストレーション作業が必要です。

質問：DECdns サーバで名前の変更が行なわれた場合、クライアントのキャッシュの中身は自動的に変更されますか？

DECdns サーバの変更は DECdns クラークに反映されます。
DECdns で名前の変更が行なわれるなどの操作により、キャッシュの中身とネームスペースに登録されている情報が不一致になるケースが考えられます。

これらの名前の不一致/不整合の解決は、DECdns を利用するクライアントアプリケーションで行うものです。そのためアプリケーション作成時には、この不一致を発見し、解決できるように設計する必要があります。そのようになっていない場合には、DECdns の再スタートなどを行い、キャッシュをリフレッシュしなければなりません。

質問：DECdns でも クラスタエリアスは使用できますか？

Phase V では、クラスタエリアスを引き続き使用できます。エリアス名の登録は、個々のノード名の登録と同様のツールが提供されます。

DECnet-VAX V5.4 Extensions は、Phase IV のクラスタエリアスになりますが、エリアス名は NET\$DNS_REGISTER コマンドプロシージャで他のノードと同様に登録します。
(SY\$MANAGER:NET\$DNS_REGISTER.COM は将来ファイル名が変更されるかもしれません。
マニュアル等をご参照ください。)

2.2 OSI

質問：OSI にはどのようなメリットがありますか？

国際標準のネットワーク・アーキテクチャである OSI を採用する場合のメリットについて、その内容を以下に示します。

1. システム構築のための自由度が拡大します。

従来のコンピュータや端末装置間の接続は、一般に同一メーカーのものに限定されていましたが、OSI を採用することにより同一の標準に準拠した製品であれば、基本的には接続が可能となります。したがって、機種やオペレーティングシステムの異なるシステム間での接続が容易になり、業務に合った自由なシステム構築ができます。

2. ネットワークの拡張に対応できます。

OSI のアドレッシング方式の採用により従来の制限枠を越えてネットワークの拡張が可能になります。また、OSIの標準プロトコルを使用することにより相互接続性が向上し、システムの拡張を容易に行なうことができます。

3. 企業間接続や国際的なネットワーク間接続を可能にします。

OSI に準拠したネットワーク同士を相互に接続することが容易にできます。

4. 分散処理ネットワークを構築することができます。

相互運用性の高い OSI は、分散アプリケーションの実現に適しています。

質問：DEC の OSI 製品の接続実績を教えてください。

DEC では国内外の相互運用試験や展示会へ積極的に参加し、各社が開発した OSI 製品との接続性を向上するように努めています。その結果、数多くの他社製品と接続実績を持ち、現在では国内外の企業間で OSI による電子メールシステムを利用しているところもあります。

質問：OSI、DECnet、TCP/IP の関係はどうなりますか？

今後 DECnet は OSI の仕様になります。しかし TCP/IP をサポートしなくなるわけではありません。DEC では TCP/IP のネットワークにアクセスするための製品を提供し続けます。TCP/IP は現在最も幅広く利用されており、事実上の業界標準ネットワークのプロトコルとして定着しつつありますが、実際には TCP/IP は厳密に標準化されたものではありません。そのためその機能を決定している仕様書 RFC(Request for Comments) も徐々に OSI 仕様を採用するようになってきています。簡単にいえば TCP/IP も OSI の機能を徐々に取り込んでいく事になります。DEC では DECnet 内部の仕様はコンピュータ同士の解放型ネットワークを前提に国際標準化された OSI

を全面的に採用し、TCP/IP へのサービスは DECnet PhaseV の共通トランスポートインターフェースを通じて DECnet PhaseV のクライアントソフトウェアの一部として提供する事になります。

2.3 ULTRIX & TCP/IP

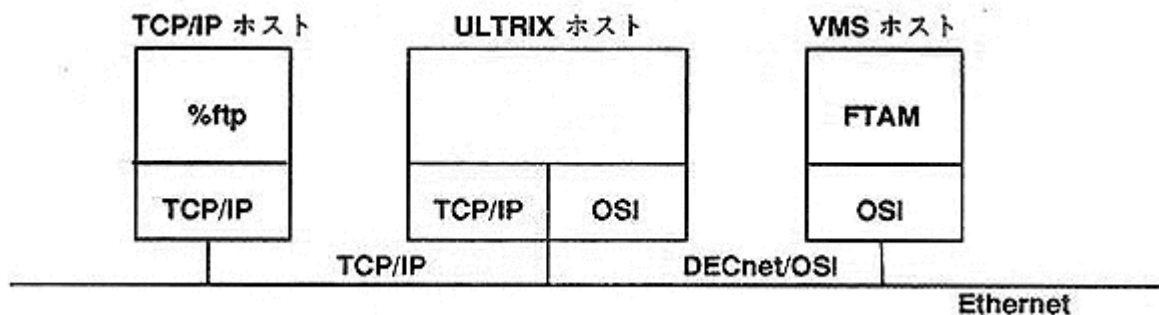
質問：ULTRIX とVMS を統合するためにはどのような方法がありますか？

今後、DECnet-ULTRIX では DECnet、OSI、TCP/IP の 3 つの通信方法をサポートしていきます。

一方 DECnet-VAX では DECnet と OSI の通信方法をサポートしていきます。また DECnet/VAX に VMS/ULTRIX Connection ソフトウェア(UCX)や ANET+ をインストールすることによりVMS ホスト上で TCP/IP を用いた通信を行なうことが可能です。したがって、次の 3 つの方法で接続できます。

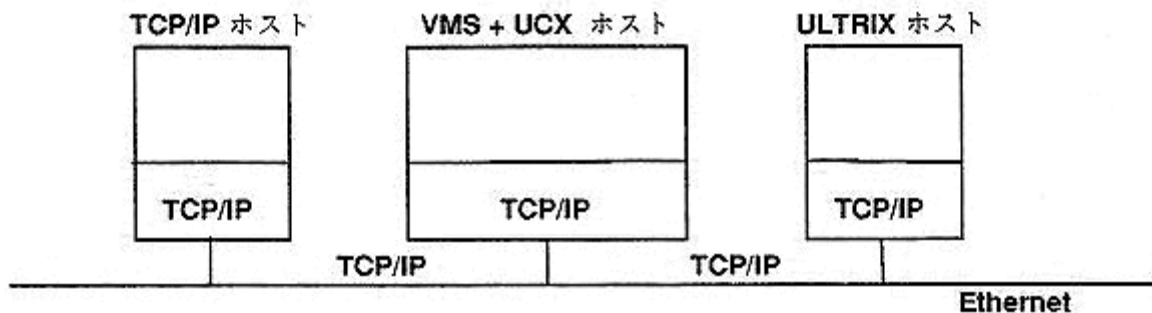
1. DECnet Phase IV による通信
2. DECnet/OSI による通信
3. TCP/IP による通信

上記 1. および 2. に関しては、下図のように DECnet/ULTRIX システムが Internet Network へのゲートウェイとし動作できる機能も提供します。



3. の TCP/IP による方法は、ANET+、WIN/TCP および UCX の 3 通りで実現されます。VMS ホストにこれらのソフトウェアをインストールする必要があります。これにより VMS ホストを Internet Network 上の 1 ノードとして機能させることが可能となります。

ANET+、WIN/TCP、UCX はいずれも種々の Internet プロトコルをサポートしていますが、ANET+ ではオプションであった NFS サーバなども UCX では標準装備となっています。



質問：DECnet ULTRIX に DECnet-VAX V5.4 Extensions のようなマイルドリリースは存在しますか？

存在しません。DECnet ULTRIX は最初から Phase V がリリースされます。

質問：ANET+ は引き続き使用できますか？

ANET+ は今後も VAX/VMS のバージョンアップに対応してサービスを継続します。

質問：WIN/TCP は引き続き使用できますか？

WIN/TCP は Wollongong 社の製品です。Wollongong 社でも引き続き VMS のバージョンアップに伴い WIN/TCP をサポートしていく方針だそうです。

質問：DECnet/OSI Phase V は ANET+ に影響を与えますか？

ANET+ は VAX/VMS 上に TCP/IP プロトコルを実現するソフトウェアです。DECnet Phase V とはデータリンク層の一部 (Ethernet) のみを共有するソフトウェアです。したがって ANET+ がインストールされているホストに DECnet Phase V をインストールしても影響はありません。但し DECnet Phase IV から Phase V にアップグレードした後、イーサネット・フィジカルアドレス AA-00-04-00-xx-xx から 08-00-2B-xx-xx-xx のオリジナルのハードウェアアドレスに変更になります。これらのアドレスを登録して使用している場合のみ注意が必要です。

質問：Internet-Gateway 機能を実現するには ULTRIX が必要になりますか？

Internet-Gateway 機能は DECnet-ULTRIX でのみ提供される機能です。この機能は、DECnet のみのホスト TCP/IP のみのホスト同士を間接的に接続する機能です。TCP/IP を標準プロトコルとしている ULTRIX のみが発現します。

質問：UCX にて Internet-Gateway 機能は実現できますか？

現バージョンの UCX は NFS (Network File System), TELNET (Network Virtual Terminal), FTP (File Transfer Protocol) 等を VAX/VMS 上で実現するソフトウェアです。したがって Internet-Gateway の機能は含まれていません。

2.4 SNA

質問：VMS/SNA は Phase V ではどうなりますか？

VMS/SNA は将来は Phase V エンドシステムに対応すべく予定されておりますが、現在の VMS/SNA は Phase V では動作しません。VMS/SNA は Phase IV ノード上でご使用いただく事になります。

質問：DECnet/SNA Gateway は Phase V ではどうなりますか？

DECnet/SNA Gateway は将来は Phase V システムとしての機能を備える予定があります。現在の DECnet/SNA Gateway は DECnet Phase IV のエンドノードとして動作しています。Phase IV のエンドノードは Phase V のエリア及び Phase IV のエリアのどちらでも動作する事が可能です。しかしこの場合ネットワークアドレスとして互換アドレスを使用し、ノード名は Phase IV 互換のためにシノニムを使用する必要があります。また、ロードホストとして Phase IV ノードが必要となります。

質問：DECnet/SNA の各アクセスルーチンは Phase V エンドシステム上で動作しますか？

DECnet/SNA の各アクセスルーチンは DECnet アプリケーションにあたります。現在のアクセスルーチンは Phase V でのノード名体系 (DECdns のフルネーム) は使用できませんので、シノニムを用いた 6 文字までのノード名を使用する事により Phase V エンドシステム上でも使用することができます。

2.5 DECMcc

質問：DECMcc は X.25 を管理できますか？

DECMcc では (DEC のプロダクトに関しては) X.25 の管理を行うことはできます。

質問：DECMcc は DECwindows 以外でも使用できますか？

VMS のコマンドラインでもご使用頂けます。

DECMcc では、Iconic Map PM, FCL(Form and Command Line) PM という2種類のユーザインターフェイス(プレゼンテーションモジュール:PM)を用意しています。DECwindows を使用する際には Iconic Map PM が使用されますが、VMS のコマンドラインで使用する際には FCL PM が使用されます。FCL PM では、コマンドの入力形式は DECnet Phase V の管理プログラム NCL (Network Control Language) と同じ書式になります。

質問：EMA とは何ですか？

Enterprise Management Architecture (EMA) は、これまであったネットワークマネージメントやマネージメントの定義が拡張された OSI マネージメント・スタンドアードを基にしたもので、単にネットワークのマネージメントや DEC のプロダクトに限定したものではありません。EMA の管理対象として考えられる環境は次のものです。

- マルチベンダ (コンピュータ、PBX、モデム、ブリッジ、etc..) 及びマルチテクノロジー (音声、データ、ビデオ、イメージ、etc..) によるネットワーク
- ネットワーク内のコンピュータシステム、オペレーティングシステム
- 各システム内のアプリケーション、データ

EMA は、管理システム (ディレクタ) と管理対象 (エンティティ) およびそれらの関係をモデル化したものです。このアーキテクチャの枠組みによって、企業内の様々な管理ディレクタ/エンティティを扱うことができます。各ディレクタは相互に通信でき、個々のディレクタは多くのエンティティを管理できるほか、複数のディレクタが同じエンティティを管理する事も可能です。

ディレクタのプレゼンテーション、機能、アクセスの各コンポーネントは、階層をなすか、またはモジュールに分けられています。独自機能の開発 (DEC 内部および他のベンダー) は、定義のしっかりしたオープン API (Application Programming Inter-face) により、正しく動作するソフトウェア・モジュールを実現できます。

AM (アクセス・モジュール) は、DECnet ノード、モデム、ターミナル・サーバ、アプリケーション等の "エンティティ" (管理の対象となる実体) を管理するためのリンクを提供します。企業ネットワーク内に存在するあらゆるものがエンティティであり、管理可能です。AM とエンティティの間は、適切な管理プロトコルでリンクされます。

FM (機能モジュール) は、DEC や他のベンダーから提供される「プラグ・イン」アプリケーションであり OSI 定義の 5 つの管理領域 (構成、障害、機能、機密保護、課金) に対応しています。FM はすべての適切な EMA エンティティにわたって動作し、構成管理や性能分析、障害発生アラーム等の機能を提供します。

PM (プレゼンテーション・モジュール) は、共通のユーザ・インターフェイスを通じてすべての情報を統合します。これらのモジュールによって、DECwindows ワークステーション、グラフィック端末やテキスト端末、あるいは、表形式、グラフィック形式、レポート形式等の複数のプレゼンテーション・デバイスやスタイルを使用できます。

各モジュールのインターフェイスは公開されており DECmcc System Reference Manual に記載されています。各ベンダーやユーザは、独自に AM, FM, PM を追加することができます。

第 3 部 TSC コールサマリー

第 3 部では、皆様から TSC にお寄せいただいているお電話の中から、皆様のお役にたつと思われるトピックを選び、ご説明いたします。

これから Q & A 形式でご紹介する情報は、全て DSIN にも登録されています。

DSIN のデータは毎月アップデートされ、今後も新しい情報を提供していきます。

今後とも、DSIN をご利用いただきますようお願い申し上げます。

3.1 MAIL が送れない。(EXDISKQUOTA エラー)

質問：メールを送ろうとすると以下のエラー・メッセージが出力されるのですが。

```
%MAIL-E-SENDERR, error sending to user ユーザ名 at ノード名
%MAIL-E-OPENOUT, error opening SYS$SPECIFIC:[MAIL$SERVER]MAIL.TXT; as output
-RMS-E-CRE, ACP file create failed
-SYSTEM-F-EXDISKQUOTA, disk quota exceeded
```

MAIL ユーティリティは、ローカルノード及びリモートノードのユーザにメッセージを送/受信するためのものです。

MAIL ユーティリティを使用してリモートノードのユーザとメッセージをやりとりする場合、指定したリモートノード上の MAIL\$SERVER アカウントとリンクをします。

(MAIL\$SERVER が存在しない時は、Default Nonprivileged User が使用されます。)

リモートノードに対して MAIL を送信する際、リモートノード上の MAIL\$SERVER は MAIL.TXT というファイルをテンポラリに作成しデータを受信します。

このエラー・メッセージは、ファイルを作成しようとした時に DISK QUOTA が不足していたためにファイルが作れなかったことを表わしています。

このエラー・メッセージが出力された時には、相手ノードの MAIL\$SERVER (MAIL\$SERVER が存在しない時は、Default Nonprivileged User) アカウントの DISK QUOTA を確認してください。

3.2 MAIL が送れない。(REMRSRC エラー)

質問：メールを送ろうとすると以下のエラー・メッセージが出力されるのですが。

```
%MAIL-E-LOGLINK, error creating network link to node ノード名
-SYSTEM-F-REMRSRC, insufficient system resource at remote node
```

MAIL ユーティリティは、ローカルノード及びリモートノードのユーザにメッセージを送/受信するためのものです。

MAIL ユーティリティを使用してリモートノード上のユーザとメッセージをやりとりする場合、指定したリモートノード上では MAIL\$SERVER アカウントを用いてリンクをします。(MAIL\$SERVER が存在しない時は、Default Nonprivileged User が使用されます。)

このエラー・メッセージは、リモートノードとリンクをはろうとした時に相手ノードのリンク数が、既に相手ノードで設定されている MAXIMUM LINKS の値に達しているためにリンクがはれないということによって出力されています。

このエラー・メッセージが出力された時には、相手ノードの現在はられているリンク数と相手ノードの MAXIMUM LINKS の設定値を確認し、必要ならば相手ノードの MAXIMUM LINKS の値を変更してください。あるいは、しばらく時間をおいてから同じオペレーションを行ってください。

また、ローカルノードのリンク数がエグゼキュータノードで設定している MAXIMUM LINKS の値に達している場合には次のエラー・メッセージになります。この場合には、エグゼキュータノードの設定を確認してください。

```
%MAIL-E-LOGLINK, error creating network link to node ノード名
-SYSTEM-F-NOLINKS, maximum network logical links exceeded
```

なお、MAXIMUM LINKS は NCP ユーティリティの SHOW EXECUTOR CHARACTERISTICS コマンドで確認することができます。

3.3 SET HOST できない。(LINKABORT エラー)

質問：システムをブートしDECnet をスタートさせた後に、SET HOST コマンドでリモートノードにログインしようとしたところ、以下のエラー となりました。

```
$ SET HOST REMNOD
```

```
%REM-F-NETERR, DECnet channel error on remote terminal link
```

```
%SYSTEM-F-LINKABORT, network partner aborted logical link
```

LINKABORT エラーは、ロジカル・リンク確立の際に相手とうまく通信ができずに、リンクが切れてしまったことを示しています。

原因の 1 つとして DECnet アドレスが重複していることが考えられます。念のためネットワーク管理者に確認し、重複していたならば DECnet アドレスをユニークなものに変更してください。

エグゼキュータノードのアドレスの変更方法は以下の通りです。

例) ノードVAXABC のアドレス 1.1 を 2.2 に変更する。

```
$ RUN SYS$SYSTEM:NCP
NCP> DEFINE EXECUTOR ADDRESS 2.2
NCP> PURGE NODE VAXABC ALL
NCP> DEFINE NODE VAXABC ADDRESS 2.2
NCP> SET EXECUTOR STATE OFF
NCP> EXIT
$ @SYS$MANAGER:STARTNET
```

注意)

SYSGEN パラメータの SCSSYSTEMID が定義されている場合には、この値も変更し、リポートする必要があります。

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE CURRENT
SYSTEM> SHOW SCSSYSTEMID
SYSGEN> SET . 2050          (2.2 の場合 1024*2 + 2)
SYSGEN> WRITE CURRENT
SYSGEN> EXIT
$ @SYS$SYSTEM:SHUTDOWN
```

3.4 SET HOST の制限

質問：あるノードの SET HOST オペレーションを制限したい。方法はありますか？

SYSS\$SYSTEM:RTPAD.EXE はローカルノードからリモートノードへの SET HOST (出力) 要求を実行しています。また、REMACP プロセスはリモートノードからローカルノードへの SET HOST (入力) 要求を受け付けています。
以下に方法の一例をご紹介します。

[出力要求の制限]

SET HOST 実行時に起動されるイメージ SYSS\$SYSTEM:RTPAD.EXE にファイル・プロテクションをかけるという方法で制限がかけられます。下記の例のでは、World のカテゴリに EXECUTE アクセス権をはずして、オーナー [100,1] とは World に属するユーザは他ノードにリモートログイン出力要求を出せないようにしています。
(no privilege for attempted operation のエラーとなります。)

例)

```
$ SET FILE/OWNER_UID=[100,1]/PROTECT=(S:RWED,O:RWE,G:RWE,W)
                                SYSS$SYSTEM:RTPAD.EXE
```

[入力要求の制限]

(方法 1)

UAFの設定で各ユーザごとにリモートログインの制限をかけることができます。
UAF での例を下記に載せます。

例) \$ RUN AUTHORIZE
 UAF> MODIFY user1/NOREMOTE=(PRIMARY,SECONDARY)

この例では、SET HOST 後、user1のアカウントではログインすることはできません。
(You are not authorized to login from this source のエラーとなります。)

(方法 2)

リモートログイン後のターミナル名は RTAnn: になりますので、SYLOGIN.COM の中 RTAnn デバイスなら LOGOUT させるといったような操作も可能です。

(方法 3)

システム上で幾つまで RTAnn デバイスを作成できるかという指定は、SYSGEN パラメータの RJOBLIM の値で示されています。この値を 0 にすればそのノードにはどこからも SET HOST できなくなります。

(Connect to network object reject のエラー となります。)

例) \$MCR SYSGEN
SYSGEN> USE CURRENT
SYSGEN> SET RJOBLIM 0
SYSGEN> WRITE CURRENT
SYSGEN> WRITE ACTIVE
SYSGEN> EXIT

RJOBLIM を変更後は DECnet の再起動が必要です。

\$ MCR NCP SET EXECUTOR STATE OFF

\$ @SYS\$MANAGER:STARTNET

3.5 DECnet ノート名

質問 : DECnet のノート名は、どのような名前でもよいですか?

DECnet のノート名は、以下のことに注意して決めてください。

< アルファベットを少なくとも1文字以上含む6文字以内の英数字 >

この条件を満たしているなら、どのような名前を付けても結構です。

3.6 NCP 実行時の特権

質問：NCP にて設定 (データベース) を参照/変更するには、どの特権が必要ですか？

NETACP プロセスが制御するネットワーク・データベース (ボラタイル・データベース) を参照するには、NETMBX とTMPMBX の 2 つの特権が必要となります。
この 2 つの特権を持っていない場合、以下のようなエラーが起こります。

```
$ MCR NCP
NCP> SHOW EXECUTOR STATUS
Node Volatile Status as of 01-JAN-1991 12:00:00

%NCP-W-OPEFAL, Operation failure
%SYSTEM-F-NONETMBX, operation requires NETMBX privilege
NCP>
NCP> SET EXECUTOR NODE TOKYO1
%NCP-I-CUREXEC, The current executor node is the local node.
%NCP-F-CONNEX, unable to connect to listener
-SYSTEM-F-NOPRIV, no privilege for attempted operation
NCP> EXIT
```

ボラタイル・データベースを変更する際には、OPER 特権が必要となります。特権を持っていない場合には、以下のようなエラーメッセージが出力されます。

```
$ MCR NCP
NCP> SET EXECUTOR MAXIMUM ADDRESS 1023
%NCP-W-PRVVIO, Privilege violation
NCP> EXIT
```

パーマネント・データベース (ファイル) を表示、変更する際には、そのファイルに READ もしくは WRITE できるアクセス権を持つ必要があります。
アクセスできない場合には、以下のようなエラーが起こります。

```
$ MCR NCP
NCP> LIST EXECUTOR CHARACTERISTICS
Node Permanent Characteristics as of 01-JAN-1991 12:00:00
%NCP-W-FILOPE, File open error , Permanent database
%MNL-E-OPENIN, error opening SYS$SPECIFIC:[SYSEXE]NETNODE_LOCAL.DAT; as input
-RMS-E-PRV, insufficient privilege or file protection violation
```

以下にディレクトリSYS\$SYSTEMにあるDECnet関係の主なシステム・ファイルを示します。

```
NETCIRC.DAT;1          --- サーキット・データベース
NETLINE.DAT;1         --- ライン・データベース
NETLOGING.DAT;1       --- ロギング・データベース
```

NETNODE_LOCAL.DAT;1	---	ローカル・ノード・データベース
NETNODE_REMOTE.DAT;1	---	リモート・ノード・データベース
NETOBJECT.DAT;1	---	オブジェクト・データベース
NETPROXY.DAT;1	---	プロキシ・データベース

3.7 トラブルシューティング ~ UNREACHABLE エラー

質問：ネットワークオペレーションを行うと、次のエラーが出てしまうのですが、どこが悪いのでしょうか？

```
%SYSTEM-F-UNREACHABLE, remote node is not currently unreachable
```

次のような原因が考えられます。

- リモートノードが立ち上がっていない。
- ローカルノード上あるいはリモートノード上でラインやサーキットが正常に作動していない。
- ルーティング経路の中での問題。
- ローカルノード上の NCP データベースでリモートノードが正しく定義されていない。
- ローカルノード上で次のような問題が考えられる。
 - サーマット及びラインが定義されていない。
 - MAXIMUM HOPS や MAXIMUM COSTS, MAXIMUM VISITS の値が低すぎる。
- リモートノードがクラスターであれば、次のような問題が考えられる。
 - ルーティングノードの MAXIMUM BROADCAST NONROUTER の値が低すぎる。
 - クラスター内に少なくともひとつはルーティングノードが存在しなければいけないにもかかわらず、ルーティングノードが定義されていない。
 - ルーティングノード上の NETACP のエンキュー・リミット(ENQLM) が低すぎる。

具体的に調べる方法を次に示します。大きく以下の 3 つに分けて進めていきます。

- A : ローカルノード上で確認すること
- B : ルーティング経路上で確認すること
- C : リモートノード上で確認すること

A. まずローカルノードに問題があるかどうかを調べます。

1. リモートノードの DECnet アドレスの確認及び設定

a. 確認方法

```
$ RUN SYS$SYSTEM:NCP
NCP> SHOW NODE node-name
```

b. 設定方法

```
NCP> CLEAR NODE node-name ALL
NCP> PURGE NODE node-name ALL
NCP> DEFINE NODE node-address NAME node-name
NCP> SET NODE node-name ALL
```


2. ラインとサーキットの確認

a. 確認方法

```
NCP> SHOW KNOWN LINES    <-- 認識しているラインの確認
NCP> SHOW KNOWN CIRCUITS <-- 認識しているサーキットの確認
```

b. 上記のコマンドで status が on-starting になっていた場合、サーキットの問題になります。

c. status が on になっていなかった場合、以下のコマンドで on にしてください。

```
NCP> SET LINE line-id STATE ON
NCP> SET CIRCUIT circuit-id STATE ON
```

3. MAXIMUM HOPS, MAXIMUM COSTS, MAXIMUM VISITS の値の確認及び変更

a. 確認方法

```
NCP> SHOW EXECUTOR CHARACTERISTICS
```

特別大きいネットワークでなければ MAXIMUM HOPS は 15 が最適値です。
MAXIMUM VISITS は MAXIMUM HOPS の 2 倍です。

(注)

MAXIMUM HOPS をあげることは、ルーティング・トラフィックを増やすことにもなります。充分注意してください。

b. 変更方法

```
NCP> SET EXECUTOR MAXIMUM HOPS value
NCP> SET EXECUTOR MAXIMUM COSTS value
NCP> SET EXECUTOR MAXIMUM VISITS value
```

4. ローカルノードが LAVc の一部である場合、ENQLM の問題も考えられます。

NETACP の ENQLM の確認及び設定

a. 確認方法

```
$ SHOW LOGICAL NETACP$ENQUEUE_LIMIT
```

論理名定義されていない場合、SYS\$MANAGER:LOADNET.COM の中の ENQLM を調べてください。ENQLM の値は、以下のような値でなくてはなりません。

$$\text{ENQLM} = (\text{サテライトの ENQLM} + 10) * 2$$

b. 変更方法

```
$ DEFINE/SYSTEM NETACP$ENQUEUE_LIMIT value
```

```
$ RUN SYS$SYSTEM:NCP
```

```
NCP> SET EXECUTOR STATE OFF
```

```
NCP> EXIT
```

```
$ @SYS$MANAGER:STARTNET
```

注) DECnet が起動されているときは再起動が必要です。

B. 問題がルーティング経路上にあるかどうかを調べます。

ループバック・テストやモデムテスト、サーキットテスト等を使ってどこで切れているかを調べてください。

異常がなければ、次のステップに進んでください。

C. リートノード上に問題があるかどうかを調べます。

1. まずリートノードが立ち上がっていることを確認します。

2. リートノードで DECnet が起動されていることを確認します。

```
$ SHOW NETWORK
```

3. リートノード上のライン、サーキットの確認

a. 確認方法

```
NCP> SHOW KNOWN LINES
```

```
NCP> SHOW KNOWN CIRCUITS
```

b. サーキットが on-starting になっていた場合、サーキットの問題になります。

c. on になっていなかった場合、次のコマンドで on にしてください。

```
NCP> SET LINE line-id STATE ON
```

```
NCP> SET CIRCUIT circuit-id STATE ON
```

4. リートノードがクラスタの一部である場合、MAXIMUM BROADCAST NONROUTER パラメータに関する問題とも考えられます。

a. 確認方法

```
NCP> SHOW EXECUTOR CHARACTERISTICS
```

MAXIMUM BROADCAST NONROUTER の値は、そのノードがサーキット上にもつことができるエンドノードの数を示しています。少なくともイーサネット上のエンドノードの数にしてください。デフォルトは 64 です。

b. 変更方法

```
NCP> SET EXECUTOR MAXIMUM BROADCAST NONROUTER value
```

5. クラスタであった場合には、クラスタ環境の設定にも問題があると考えられます。

a. クラスタの各ノード上の確認

```
NCP> TELL NODE node-id SHOW EXECUTOR CHARACTERISTICS
```

TYPE というフィールドを確認してください。クラスタは ROUTING IV と定義されているノードが存在し、かつそのルータがクラスタ・エイリアスとして起動されていないわけではありません。

b. クラスタ内にルータがない場合、ひとつのノードをルータに設定します。

```
NCP> SET EXECUTOR STATE OFF
NCP> DEFINE EXECUTOR TYPE ROUTING IV
NCP> EXIT
$ @SYSS$MANAGER:STARTNET
```

ルータは定義されているが起動されていない場合には、再起動するか、あるいは別のノードをルーティングノードに設定します。

6. リポートノードが LAVc の一部である場合、ENQLM の問題も考えられます。確認及び変更方法は、A.4. をご覧ください。