

U-Cloud ネットワーク技術の利点

Strong Points of the Network Technology of U-Cloud System

林 和 行

要 約 U-Cloud[®] は 2008 年より業界に先駆けクラウド事業を展開し 5 年を迎えた。この間にクラウド基盤はインターネット向けクラウド基盤 MiF2009, イントラ接続専用クラウド基盤 MiF2010 と順次発展し、現在は両機能を有する基盤 MiF2.0 によるサービス提供が主となっている。これまでのサービス提供の中で顧客からの要望やシステム運用に関してのノウハウ、システム障害を通して様々なクラウドサービス継続に重要な基盤のノウハウを蓄積した。これらのノウハウは、サービス事業者にとって機能性・可用性・保守性・拡張性をネットワーク基盤として実現するために重要である。

Abstract U-Cloud[®] has provided Cloud services for five years as a pioneer in this industry since 2008. At the beginning, Cloud Platform MiF2009 was used for Internet users. Then Cloud Platform MiF2010, which can connect to Intranet of enterprises, was used exclusively for intranet users. At present a Cloud Platform MiF2.0, equipped with both internet and intranet connecting function is mainly used to provide the cloud services. From these experiences we have got various know-how's and reflected these in U-Cloud. Especially, the improvements of functionality, availability, maintainability and scalability in these network systems are very important in realizing the network platforms.

1. はじめに

2000 年当初からの IT インフラに対するユーザの指向が、IT インフラを「所有する」から「利用する」へ一層の加速を見せている。これには「安定した IT インフラの利用」「新規機能のスピーディーな享受」「固定の資産の削減による企業の損益 (P/L) 評価向上」「システム運用要員確保の安定化」「システム運用コストの削減」等が理由としてあげられる。これを基盤側から見ると、「必要な機能の提供」「システムの安定供給」「システムコストの削減」が要求されていると言い換えられる。基盤構築においてはこれらの目的を実現しつつサービスを提供することが要件となる。U-Cloud[®] 基盤はこれらの要求事項および設計の基本要件を満たした基盤を実現すべく、モジュール型の基盤として進化してきている。

一方、高度情報化社会におけるスピード化は顧客が IT システムに期待する要件の変化も加速する傾向にある。このためクラウドビジネスで顧客要件を適切に実現するためには IT システムのライフサイクルの変化について理解を深める必要がある。

本稿では、2 章でライフサイクル面から、3 章で多重度からサービス型システム用の基盤に対する要求を考察し、4 章でサービス型システムのネットワーク部分についてどのような構成を取ると良いか、U-Cloud としてどのようなネットワーク構成をとっているかを述べる。

2. システム基盤におけるライフサイクル

1章で述べた要求事項を実現するために社会的にもサービス型のシステム提供が発展を遂げてきたが、サービス提供を実現するシステム基盤がこれまでの所有型システム基盤とは「ライフサイクル」に違いがあることを認識して基盤設計をすることが重要になる。所有型基盤とサービス型基盤の差異を明確に把握・理解して基本設計をすることがクラウドビジネスにとって重要であり、サービス提供に大きな影響を与える。

2.1 所有型基盤のライフサイクル

所有型の基盤では通常、要件定義・設計・構築・利用・更改が一連の流れで実装・展開されていく。つまり基盤を所有するには機能・スペック・物理的な条件がはじめに合意され、その後構築がなされ、顧客に引き渡される。顧客は一旦基盤を所有すると投資コストが原価償却されるまで大きな基盤の変更はできず、決まった機能・スペック・サイジングの範囲内で基盤を利用し続けるのが通常である。そして、この大きなサイクルは5年以上に及ぶことが多い(図1)。利用者にとって、ライフサイクルは新機能や性能アップの契機として重要な位置づけを持っているので、利用者が新機能を楽しむには数年のサイクルが必要である。

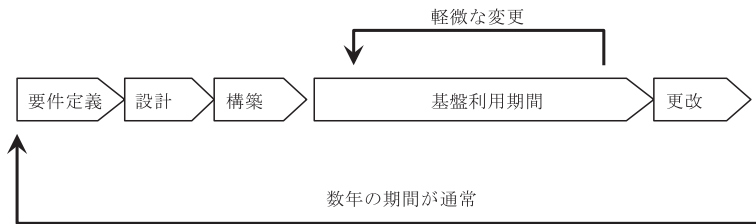


図1 所有型基盤のライフサイクル

2.2 サービス型基盤のライフサイクル

これに対してサービス型の基盤では、基盤がサービス提供されている状態で利用者毎の要件定義、システム構築が行われる。基盤全体のライフサイクルと利用者のシステム利用期間が直接結びついておらず(図2)、利用者にとって、基盤のライフサイクルは事前考慮が必須な事項ではなくなっている。

サービス提供側から考えると、既にリリースされた基盤であることを言い訳に時代の先端の要求変化に答えられずにいれば、利用者を獲得できない事態に陥る可能性が高い。このためサービス型の基盤構築においては、ライフサイクルの基本概念について大きな変革が要求されている。

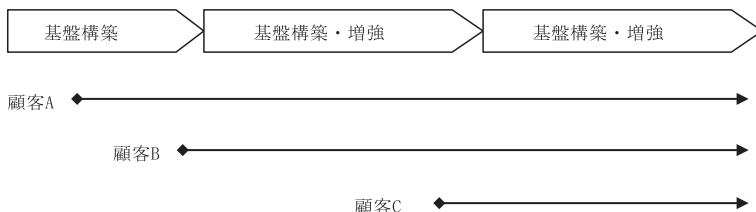


図2 サービス型の基盤のライフサイクル

2.3 ライフサイクルの違いがもたらす技術要件

当然、サービス事業ではそのサービスの適切なメニュー化によって、基盤の機能・性能が極端に変更されてもサービスが正常に維持されることが必須である。ただし、サービス型基盤提供が普及した今日では、機能・性能が基盤の長期的なライフサイクルと結びついて固定化してしまうと、新規顧客の獲得機会消失や既存顧客の他社への流出を招く。このことは所有型基盤では5年から7年とされている基盤のライフサイクルが、携帯電話等のサービス基盤では一般的に3年程度に短縮されてきている事実にも顕著に現れていると考えられる。

顧客の要求に如何に柔軟に応じていくかはビジネス面の課題であるが、この課題に柔軟に応えられる基盤を提供することが技術者に課せられる大きな要求事項となっている。つまり、これまでのように「はじめに要求ありき」のような設計思想ではなく、「欲しい時が必要なとき」を可能にする基盤を構築することが、設計思想の根本として要求され始めているのである。

3. テナント多重度

サービス型のシステムでは常に多くの利用者（マルチテナント）がサービスの利用を開始し、享受し、基盤を選択する機会を持っている。このため、単一ユーザ（シングルテナント）による所有型システムでは考えられなかった要件要求が出現する。サービス型と所有型のライフサイクルの違いはシステム内のテナント収容数の差、共有部分のシステムのテナント多重度の反映ともいえる。システム上のテナント多重度を的確に評価することが基盤設計上重要な要素となる。

3.1 多重化の利点

多重化はひとつの物理基盤を共有するため全基盤投資負担を総テナントで按分する。共通の構造を持つことにより運用も同一のスキームで対応できるため、運用コストの削減につながる。また構築は一括して行うため、個々のユーザへのリードタイムが短縮できるメリットがある。

3.2 多重化の注意点

反面、多重度を上げると必要機器のスペックが上がるため初期投資が大きくなる。また、VLAN数やIPアドレス数などにはリソースの規格上の上限があるため、実際には技術的に無限に多重度を上げられるものではない。さらに、多重化したシステムでの障害や計画停止の際の運用の対応を考えると、停止がビジネスにあたえるインパクトは概ね以下の式 [A] に相当すると考えられる。

$$T_t \geq N \times T_i \cdots \cdots [A]$$

T_i ：システムの停止時間

N ：そのシステムのユーザ多重度

T_t ：総合ビジネス停止時間

ビジネスインパクトの大きさを評価する関数 $I(T)$ を定義する。障害におけるビジネスイ

ンパクトは以下の式 [B] のように多重度が大きくなるに従って増大する。

$$I (Tt) \gg I (N \cdot Ti) \cdot \dots \cdot [B]$$

したがって、多重度が上がれば採算性やサービスが向上すると一概には決め付けられないことも事実である。

4. 顧客要求からのネットワーク基盤の必須要件

前章で述べたマルチテナント型のシステムの特徴を踏まえながらマルチテナント型のネットワーク基盤への要求を整理する。サービス提供においてネットワーク基盤に要求される具体的な項目には以下のものがあげられる。

- 1) リソースの拡張性
- 2) 機能拡張性
- 3) 運用負荷の最適化
- 4) 耐障害性

これらを利用者のサービス享受に影響少なく実現できていることがサービスビジネスの成功に大きく寄与する。これらをさらに本章で詳細に述べる。

4.1 リソースの拡張性

サービス型基盤上には日々顧客が増え続け、顧客の基盤リソース利用量が増加することが予想される。リソース拡張作業がサービスの継続に影響しないことが重要である。単純拡張性だけを追求するならば大規模システムを当初から構築すればよいが、これでは初期投資が大きくなり、ビジネス採算に大きな影を落とす。

この課題がビジネス要件ではなく技術要件であることも明白である。サーバの増設を例にとって考えてみる。サーバ増設に紐付いてネットワーク機器のポートやネットワーク機器自体を増設する必要がある。通常、増設時にはネットワーク上の経路の切り替えが行われる。この時、ネットワーク通信は基盤冗長構成の一つに寄せられる。切り替わりによってネットワークトポロジーが変更するタイミングで通信の障害がおきることがある。このネットワーク上の障害時間は選択された基盤冗長化プロトコルによって数十 ms ～数分と様々な値になる。切り替わり時間の参考値を表 1 に示す。

表 1 冗長化プロトコルの切り替わり参考値

	通称	切り替わり時間	動作仕様
1	Ring冗長プロトコル	数十～数百ミリ秒	ASIC処理
2	DCBP*1	数十～数百ミリ秒	ソフトウェア処理
3	HSRP、VRRP	数秒～十数秒	ソフトウェア処理
4	STP	数十秒～数分	ソフトウェア処理
5	RSTP	数秒～数十秒	ソフトウェア処理
6	Dynamic ルーティング	数秒～数分	ソフトウェア処理

どのような冗長化プロトコルを選択するかを考えると、3.2節で述べた式 [A] の考え方が重要になってくる。マルチテナントサービス基盤の事業継続において、数分間のネットワーク

断はテナント多重と相関してシングルテナントの数百倍分に等しいかそれ以上のビジネス損失を生む。したがってネットワーク基盤の冗長化プロトコル・切り替わり方式は慎重に選定する必要がある。

4.2 機能拡張性

サービス事業はサービスカタログに基づいて提供されているため、個別の顧客要求によって機能が追加されることは基本的にはない。ただし複数の顧客から同様な要求がある場合、市場全体の潜在要求である可能性が大きい。サービス継続のためには、市場要求に対してそれを的確に分析しサービスメニューに取り込み、サービスに反映していくことが必須条件である。高度情報化市場では基盤自体のライフサイクルに比べ市場のメニューへの要求の変化のサイクルが短くなる傾向にあることは2章に述べた。そこで新規機能追加の度に基盤の大幅な更改が必要にならないことが重要である。

以下に拡張しやすい基盤について考えてみる。サービスはメニュー単位で提供されるため、基盤の構成にも次のことが要求される。

- 1) 基盤の構成単位とメニューの単位とが一定の相関性をもっていること
- 2) メニュー単位での基盤増強が容易なこと
- 3) メニューの追加・変更にとまなう基盤の追加・変更が容易なこと

以上のような要求を実現するためにプログラミングの世界でも、高級言語を中心に機能ごとに構成をモジュールとしてまとめることが一般的である。構築物の機能ごとに分界点をはっきりしており、互いのモジュールが汎用性のあるインターフェース^{*2}をもって結合される。このようなモジュール構造をとることによって、大規模な構築ネットワーク基盤がサービス提供システムとして利用しやすく機能拡張も容易になる。

4.3 運用負荷の最適化

マサービス・マルチテナント型サービスを成功させるには、運用を考慮した基盤設計が不可欠である。日々のサービス経営において客先との調整や基盤運用の人件費が最も経営採算を圧迫する傾向があるからである。そのため設計者は基盤の設計に際し、運用上の負荷軽減と保守作業の負荷軽減という二つの観点に留意することが重要である。本節で解説する。

4.3.1 運用上の負荷軽減

サービス型基盤には顧客要求を受容できる柔軟性を持ちながら顧客ごとに個別に運用が変わらないような統一性が必要である。各テナントの構成を極力相似形に収めることにより、運用手順の簡素化・汎用化を図る。このためにはユーザテナントに必要な共通構成を洗い出し、構成の公約数と言えるシステムの組み合わせを顧客に提供する必要がある。U-Cloudにおいては図3のような構成をネットワークメニューとして各要素を顧客が取捨選択できるようにしている。

提供形態が決められた範囲を逸脱すると運用負荷が増えサービス提供のコスト高騰につながる。したがって、メニュー型ビジネスの標準構成を定義することは重要である。運用の観点からユーザシステムは標準構成に収める必要がある。

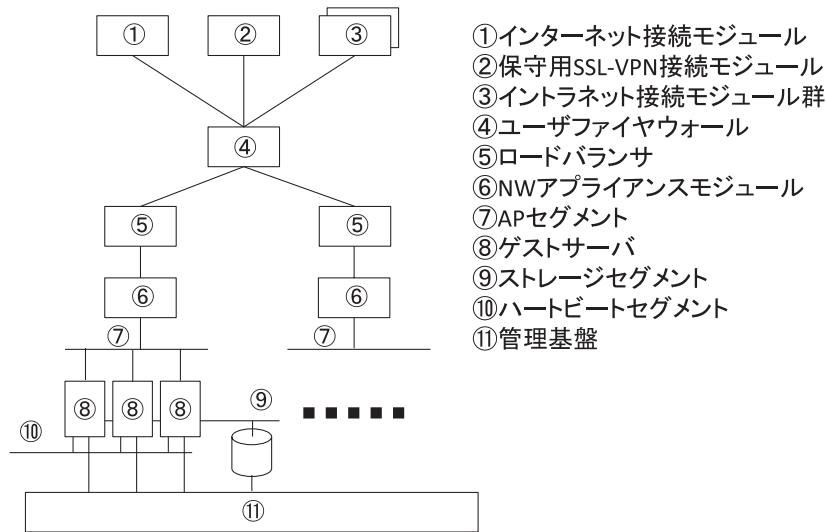


図3 U-Cloudの標準的構成

U-Cloudでは④のファイアウォールを基点として、顧客が各モジュールを必要に応じて選択しシステムを構成できる。モジュールの数も一定の制限内で増減可能である。⑦のAPセグメントに顧客サイトから任意のセグメントを切り出してIPを割り振ることも可能である（適切なセグメント分割とルーティングの整合性がとられていることが前提）。このようにかなりの自由度を選択できる構成となっているが、それでもなおこの範囲を超えた構成を要求^{*3}される場合がある。将来的にこの要求にメニューで応えていく検討は必要であるが、個別の要求に都度対応すると受け入れ工数や運用負荷が上がるため、受け入れにはサービス全体の採算性の検討が必要である。

4.3.2 保守作業の負荷軽減

サービス型基盤を提供する上で更に重要なのは、保守作業の負荷を軽減しつつ、確実にサービスを実施することである。基盤の基本設計にこのための思想を盛り込むことが非常に重要である。シングルテナント、またはオンプレミスのシステム基盤では基盤の管理者が利用者に対して比較的強い権限をもって保守の時間調整や保守作業を実施できる。ユーザの利用時間や利用形態が比較的類似しているため、システムの繁忙期と閑散期が設定しやすく、メンテナンス等の時間の調整が容易なのである。これに対してマルチテナントのサービス型基盤では様々なユーザが様々なビジネスサイクルをもってシステムを稼働させている。あるユーザにとってシステムの閑散期は他のユーザにとっての繁忙期である可能性もあり、保守作業の時間帯（メンテナンスウィンドウ）を設定することがシングルテナントに比べてはるかに難しい。また、3.2節の式[A]でみたように、保守停止によるビジネス上の顧客に与えるインパクトはシングルテナントに比べて飛躍的に増大する。このようにマルチテナントビジネスの特色を考えると、保守性を如何に上げ利用者からみた停止時間を如何に軽減するかという課題が基盤の設計思想に重要な要素であることがわかる。

4.4 耐障害性

すべてのシステムにおいて耐障害性が高いに越したことはない。メンテナンスの停止と同様に障害による停止は大きなビジネスインパクトになる。また当然のことながら障害は予期しないタイミングで発生するためその被害はさらに大きくなる。そのため障害に対して次のようなスタンスで基盤を設計する必要がある。

- 1) 障害原因についての検討・検証が十分行われ、障害の影響を受けにくいプロダクトを選定すること
- 2) 基盤の構成をできるだけシンプルにし重複障害が起きないように設計すること
- 3) 障害の予兆分析・早期発見がしやすい基盤とすること
- 4) 万一の障害発生時に障害の影響の波及を局所的に押さえ込めるようにすること

5. 基盤に対する MiF2.0 の解

4章までに述べてきたように、マルチテナントのサービス型基盤で設計思想に取り入れなくてはならない概念は、これまでのオンプレミス・シングルテナントのシステム基盤とはかなり違ったものである。4章までの課題を解決するため、現在の U-Cloud 基盤 MiF2.0 は、以下に挙げる解をもってネットワークを構築している。主なものについて本章で解説する。

- 1) 基盤のモジュール構成の実現
- 2) 基盤とサービスメニューの区分との相関関係の明確化
- 3) ベンダー、OS 非依存性または低依存性の実現

5.1 モジュール構成

現行の U-Cloud サービスのシステム基盤である MiF2.0 では、ネットワーク基盤の構成をモジュール化することに重点を置いている。モジュール化とは基盤の構成を機能ごとにモジュールとして分割し、それぞれのモジュールを比較的シンプルで堅牢に動作する接続部位（インターフェース）で接続することにより、各モジュール間の機能的な相互の依存関係を極小にした「疎結合」形態とすることをいう。また、基盤の各モジュールをサービスメニューの単位と相関関係をもたせ「モジュールの単位」イコール「サービスメニューの単位」に近いシステムを作る。このようなモジュール化基盤の実現により4章までに見てきた課題を解決する重要な手段としている。

モジュール化は各部位の技術的な依存性を一定のインターフェース^{*2}で区切るためベンダーや OS による各モジュールの相互依存性を低くすることにより基盤の保守性向上に寄与する。このような基盤のモジュール化についてももう少し詳しく述べる。

5.1.1 密結合の基盤

モジュール化の利点を述べる上で、「疎結合」とは逆に「密結合」の基盤を考える。「密結合」の基盤とは、各機器の結合度や機器同士の OS のバージョン依存性が強く一部のネットワーク構成変更が他のネットワーク部位に伝搬しやすい構成のことをさす。

例えば、これまでのネットワーク設計のように一般的なダイナミックルーティングを基盤上の主要部分部位で広範囲に動作させるとする。この場合、一箇所での障害や保守時の切り離しの影響が基盤全体にルーティング情報とともに伝搬し影響が拡大する 경우가多々ある。シング

ルテナントで考えるものとは比較にならない大きなインパクトがマルチテナントのシステム基盤上のサービスにおいてあることはこれまで見てきた通りである。

古くからのネットワーク設計においてもこのことは認識され、L3 冗長プロトコルである OSPF においてはエリア分けを行ったり、L2 冗長プロトコルであるスパンニングツリーではグループ化がなされたりしてきているが、もう一步踏み出して基盤の構造自体を明確にブロック化・モジュール化することがサービス型基盤においては重要であるという「疎結合」の思想のもとに MiF2.0 の設計がなされている。

「密結合」のネットワーク基盤の例として、一般的なネットワーク教科書に掲載されているデータセンターのネットワーク構成（図4）を考えてみる。

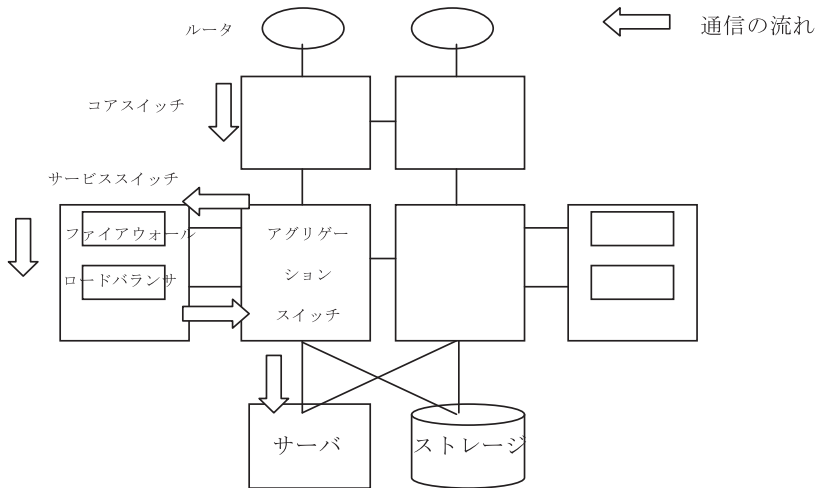


図4 教科書的ネットワーク基盤模式図

この構成はサービス層→アグリゲーション→コアと上位に向かうほど集約度が上がり配下の接続ノード数が増す逆ツリー構造をとる。上位の層のメンテナンスはその枝である下位の層に直接影響を与える。このため上位になるほどメンテナンスの条件が厳しく、サービス断が配下の階層全域に影響する。つまり多重度の大きい「密結合」の基盤では重要な基幹部になるほど、解決すべき課題が顕在化していてもサービス断の伴うメンテナンスはビジネスインパクトが大きく実行しにくいというジレンマに陥りやすい。

また図4のような二台冗長構成を基本にすると、正系・副系の対になる機器ではOS・ファームウェア等は同じバージョンを使用するのが通例であるから、特定のOSに内在するBugなどの影響を正副の機器で同時に受けてしまいサービス継続を阻害しやすい弱点がある。結論として密結合の基盤は、運用上の解決課題と運用負荷を増大させる方向にある。

5.1.2 疎結合の基盤

基盤の機器構成がどのようなものであれ、ユーザに提供されるサービス構成が4.3.1項の図3のようなトポロジーであれば、現状のU-Cloudサービスの提供が可能である。したがって、利用者のネットワークトポロジーを同じ形で提供しながら「密結合」基盤の課題を解決できる基盤が求められる。

ここで図3のネットワークのトポロジーを再度確認する。外部から通信が入ってきたときの通過ネットワークノードの流れを順に追うと

「ルータ」→「L2スイッチ」→「ファイアウォール」
 →「ロードバランサ」→「L2スイッチ」→「仮想サーバ」・・・[C]

という順にパケットが流れる。このトポロジーによる「疎結合」の基盤が実現できれば、これまで議論した問題について多くが改善される。

「疎結合」基盤を実現する構成の具体例は、メニューに対応した機能の塊「モジュール」が高速に切り替わる冗長機能を持つ接続モジュールにて相互に接続されているものである。これはちょうど電子ブロック工作で回路を組み上げるスキームと酷似しているため、電子ブロックの構造を思い浮かべてみると理解しやすい(図5)。

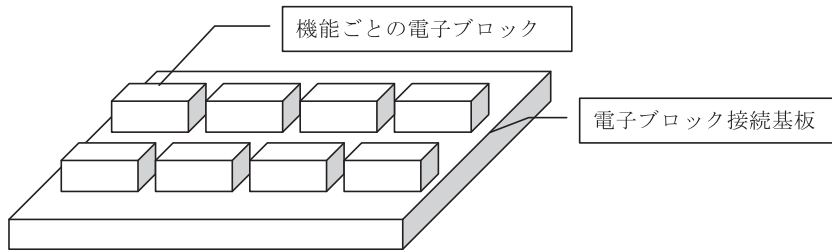


図5 電子ブロックのイメージ図

ここで電子ブロックの接続基板の機能は、安定した接続機能(冗長性)と各ブロックを受け入れる汎用性にのみ特化している。そして、接続基板にて相互に接続される各ブロックはそれぞれのブロックごとに単独で閉じた特定の機能を持っており、ブロックごとの改修やグレードアップがしやすい。これは「疎結合」基盤に要求される特性そのものである。

この電子ブロックに相似したシステム基盤上にてサービスができれば、「密結合」基盤上で課題としてあげられた事項を解決していくことが容易になる。このような発想のもとに[C]のネットワーク論理トポロジーを実現する「疎結合」基盤として、U-Cloud MiF2.0 ネットワーク基盤が構築されている。電子ブロック模型と疎結合基盤の比較表(表2)とMiF2.0 ネットワーク基盤模式図(図6)を下記に示す。

表2 電子ブロック模型と疎結合基盤の比較表

電子ブロック模型	疎結合基盤	基盤機能特徴
接続基板	相互接続モジュール	高速切り替わりのL2面
電子ブロック	各機能モジュール	外部接続やサービス系の機能を提供

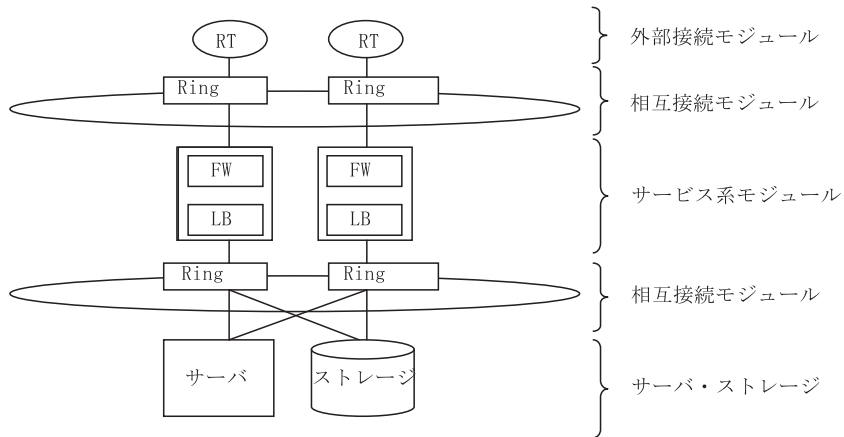


図6 MiF2.0 ネットワーク基盤模式図

電子ブロックの接続基板にあたる部分（図6では「相互接続モジュール」にあたる）はMiF2.0基盤ではASICベースで動作し、高速な切り替えを実現するRing構成が採用されている。Ring機能がASICベースの動作であることにより、OSバージョン依存によるBugの影響や保守性の低下も回避する目的がある。

図6の物理基盤構成上にネットワークノードの流れ[C]で表現されるユーザネットワークトポロジーを実現する。ユーザから見たネットワークの構成は「密結合」基盤であろうと「疎結合」基盤であろうと変わらない。一方、物理的な構成を見た時、各機能を実現するネットワークの機器の集まりであるモジュールは各々がRing面に接続され、Ring面を通じて相互に通信する。運用者の立場から見れば、メニューに即した機能を持ったモジュールの追加や取り外しは各Ring面に対して行う作業であって、基本的に他の機能のモジュールとは物理的な関係を持たない。

機能を持つモジュール群の要素が相互に直接接続されることなく、シンプルで堅牢に動く接続モジュールで接続されている様子が、「疎結合」のネットワーク基盤である所以である。

5.1.3 MiF2.0の効果

「疎結合」のネットワーク基盤構成を採用することによって、MiF2.0は4章で述べたリソースの拡張性、機能拡張性、運用負荷の最適化、耐障害性について「密結合」の基盤より大きく向上させることができている。また、MiF2.0は「疎結合」のモジュール基盤を採用することにより以下の点が改善されており、一般的な基盤と較べて多くのアドバンテージを獲得している。

- 1) インターネット接続及びNATのいらぬイントラネット接続の同時提供
- 2) モジュール構造採用による機能拡張性の装備
- 3) 疎結合の基盤による障害波及の軽減
- 4) 疎結合の基盤による保守作業の簡易化

6. さらなる取り組み

U-Cloudではモジュール化基盤の採用で機能拡張、基盤の安定化、保守性の向上を実現して

きたが、サービス型基盤の拡張・拡充・安定化に関して、本章に挙げるさらなる取り組みを予定している。

6.1 基盤安定化

モジュール化基盤上では、N+1 型の冗長化や 1+0.5 型^{*4}の基盤冗長化が実現可能である。ユーザーニーズに合わせて今後 U-Cloud のさらなる安定化を目指して基盤拡充を行う必要がある。

6.2 サービスレイヤの分離

5.1 節で機能ごとにモジュール化を行い疎結合の基盤を作成することを重点に議論を進めた。今後は SDN 技術^{*5}などの導入によりネットワークの機能をネットワークのレイヤの観点からモジュール化する必要もある。具体的には、ファイアウォールのようなレイヤ 4 に近い層で動作する機能モジュールと、ロードバランサのようなレイヤ 7 に近い層で動作する機能モジュールをはっきりと分離・疎結合化することなどである。レイヤ 4 の機能モジュールとレイヤ 7 の機能モジュールはその処理内容から機能の CPU の利用状況や障害時の動作にはっきりと差がでるため、これらを同一の筐体でまとめてマルチテナントサービスすることはリスクが大きい。このためレイヤ 7 の機能は積極的に SDN 技術を取り込む一方でレイヤ 4 付近までの処理は高速安定稼働を重要視するというように役割と動作の概要をより明確に基盤のモジュール化に取り入れる必要がある。このような基本概念とモジュール化を基に、顧客が品質・機能・価格の各面でより自由度の高い組み合わせでメニュー選択が可能な基盤を構築していくことが今後の課題である。

6.3 運用基盤の充実

U-Cloud 事業は、設計・構築・運用全般をオールインワンで運営していく事業形態をとっている。このためクラウドビジネスに必要なニーズ、技術を集約して運用まで含めたシステム構築が可能である。今後利用者・運用者の視点を一層取り込んで基盤を発展させていく。

7. おわりに

MiF2.0 の構想から構築・運用にわたってご尽力頂いた社内・社外の各位に心から感謝申し上げます。特に、基盤構築初期に発想部分から共に取り組んできたネットマークスの富田啓太郎氏、実構築にご尽力頂いた設計担当大塚泰弘氏、前川さやか氏はじめ各位のお陰で基盤実現化にこぎつけ、また運用技術部諸氏によって支えられています。

また、ご多忙にかかわらず査読にご協力頂いた寺嶋浩信氏、郡司雄太郎氏に心より感謝いたします。

-
- * 1 DCBP: Data Center Bridging Protocol. TRILL などに代表されるデータセンター向けプロトコル。ネットワークングを利用してデータトラフィックやサーバ I/O を統合する。IT ベンダー各社で独自の方法が提案されている。IEEE802.1 部会で標準化対応。
 - * 2 ここでいうインターフェースは結合機能のインターフェースの意味で NW 機器のインターフェースに特定しない。
 - * 3 要求例として、U-Cloud 基盤内の AP セグメントのお客様基盤への L2 延伸、LB の個別設定負荷分散、図 3 ⑥の位置への他のネットワークアプライアンス挿入設置などがあげられる。
 - * 4 1+0.5 型: 冗長化されたネットワークに非常時に同様の機能を提供できるシングル経路を

付与するシステムを指す。一般的な冗長機能のように OS 依存の同期は行わず独立した別経路を設けることで非常時の退避路を確保する。四重化の低コスト形態。命名は著者の造語。

- * 5 SDN : Software Defined Networking の略。ソフトウェアによって仮想的なネットワークを作り上げる技術全般。

参考文献 [1] 立花 幸治, 浅井 保行, 「MiF における仮想化技術の利用」, ユニシス技報, 日本ユニシス, Vol.29 No.1 通巻 100 号, 2009 年 5 月
[2] 森 駿, 「日本ユニシスの次世代 iDC コンセプト “MiF”」, ユニシス技報, 日本ユニシス, Vol.29 No.1 通巻 100 号, 2009 年 5 月

執筆者紹介 林 和 行 (Kazuyuki Hayashi)

2001 年ユニアデックス株式会社入社。Cisco 製品サポート, 営業支援, NW 設計, プロダクト主管等を経て, 2009 年より MiF 基盤 (U-Cloud 基盤) 主管を担当。MiF2.0 のコンセプト設計を担当。通常業務として基盤主管部業務の他にクラウド基盤の提案等も行う。

