

Webサービスの分散運用管理標準 – WSDM

加藤英晴

((株)日立製作所)

2006年8月、OASIS標準WSDM1.1がリリースされました。

本稿では、WSDMを構成するMUWS、MOWSの基本的な概念について解説します。

システム運用管理の課題とWSDM

今日の企業情報システムは、サーバ、ストレージ、ネットワーク機器などさまざまな種類の複数の分散するITリソースから構成され、これらのITリソースを提供するベンダもさまざまです。このような異種分散ITリソース環境の運用管理を行うためには、必然的に多種多様な技術、製品を用いることになります。また、運用管理と一言でいっても、性能監視、障害監視、資産管理、ファイル配布、セキュリティ管理などさまざまであり、それぞれの運用管理タスクごとに異なる種類の製品、技術が用いられます。このように、多種多様な技術、製品が混在する状況は、企業における運用管理コストを増大させ、企業経営に非常に大きな影響を与えます。このような課題を解決するために、さまざまな種類のITリソース、技術、製品を統合的に扱うことができる標準運用管理技術に対する期待が高まるなか、システム統合技術としてのWebサービス技術を運用管理に適用しようという考えから生まれたのがWSDMです。

2003年、Webサービス技術を用いたITリソース管理標準仕様の策定を目的として、Web、XML技術の標準化団体であるOASIS (Organization for the Advancement of Structured Information Standards) において、WSDM TC (Web Services Distributed Management Technical Committee) が設立されました。WSDMは、前回までに説明したWSRF、WS-Notificationを基盤仕様として利用する仕様です。

WSDM TCの目的には、Webサービス技術を用いたITリソース管理仕様の標準化に加え、Webサービス自体を管理するための標準仕様の策定も含まれています。WSDMはMUWS (Management using Web Services) とMOWS (Management of Web Services) の2つから構成される仕様群です。MUWSは、ITリソースの管理インタフェースをWebサービスとして表す方法を規定します。MOWSは、Webサービスを対象とした管理の

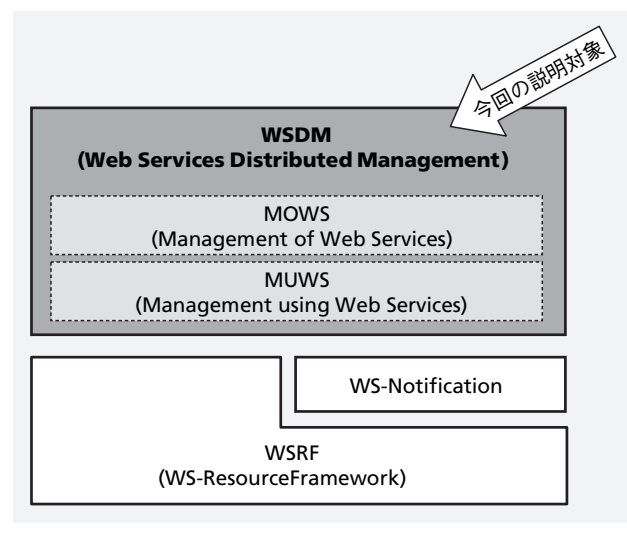


図-1 WSDMの位置付け

ための仕様であり、MUWSを用いたWebサービス管理モデルを規定します。つまり、MOWSはMUWSの拡張 (MUWSの1つのアプリケーション) という位置付けになります。図-1にWSDMの位置付けを示します。

WSDMの特徴

WSDMに準拠する管理インタフェースを公開するITリソースあるいは管理アプリケーションは、Webサービスインタフェースを公開する一般的なビジネスアプリケーションと同様、Webサービス技術を基盤として成り立っています (図-2)。Webサービス技術に基づくWSDMは、プラットフォーム非依存、サービス指向、疎結合といった特徴を持ちます。つまり、管理対象リソースへのアクセスは、管理対象リソースの実装とは分離することができ、クライアント (管理アプリケーション) は、管理対象ITリソースの実装を意識する必要はなく、管理対象ITリソースの実装に依存しない一貫したアクセス方法によって、管理対象ITリソースにアクセスすることができます。

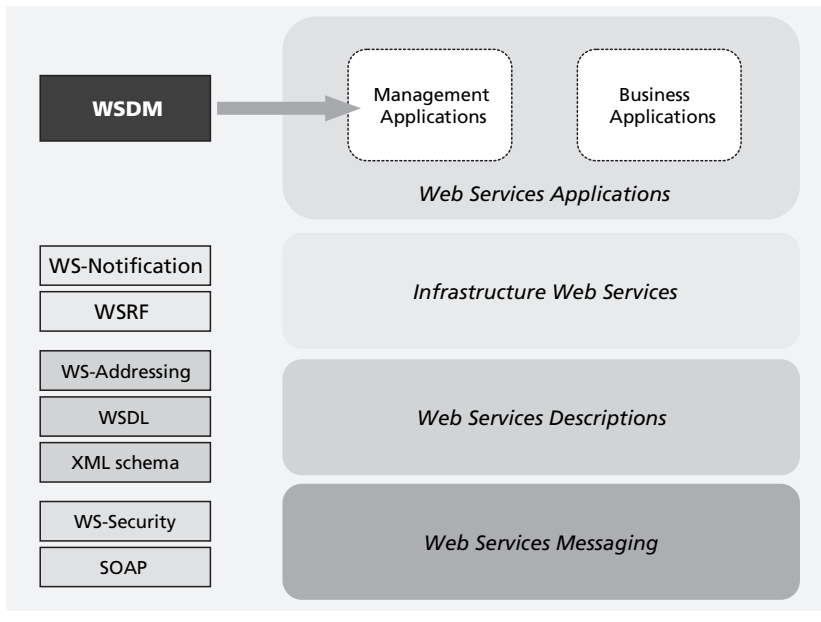


図-2 WSDMの基盤となるWebサービス技術仕様

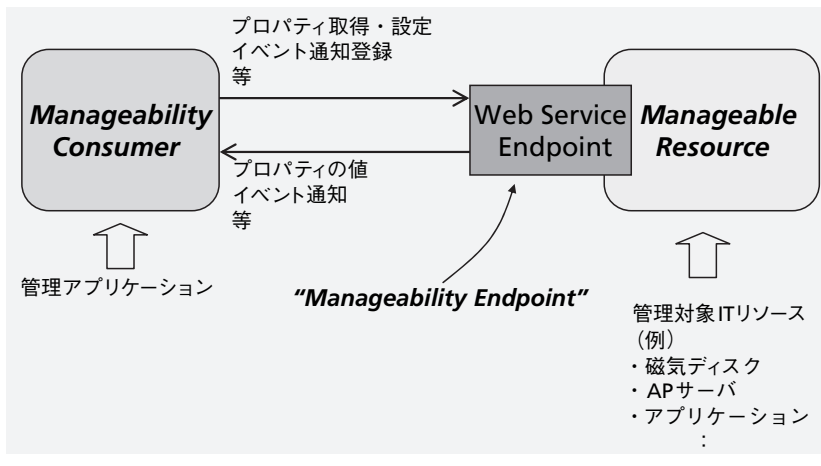


図-3 WSDM (MUWS) の概念

また、WSDMはリソース指向という特徴を持ちます。ITリソースの管理インタフェースをWebサービスインタフェースで提供することにより、管理対象ITリソースがSOAに加わることができます。ITリソースの管理サービスがビジネスプロセスの一部に参加できることにより、高度な管理機能が実現することがより容易になります。

WSDMでは、Webサービスインタフェースで管理可能なITリソースをManageable Resourceという概念として扱います。Manageable Resourceの管理インタフェースを提供するWebサービスエンドポイントをManageability Endpointと呼びます。図-3にWSDM (MUWS) の概念を示します。前述のように、WSDMはWSRFを基盤技術としており、Manageable ResourceはWSRF (Web Services Resource Framework) が規定

するWS-Resourceの概念として表されます。つまり、Manageability EndpointとManageable Resourceとの関係がWSRFで定義されるWS-Resourceということになります。管理対象ITリソース (Manageable Resource) が持つプロパティは、WSRFが規定するリソースプロパティ (Resource Property) で表されます。WSRFでは、リソースプロパティにアクセスするGetResourceProperty, SetResourcePropertiesなどのWSDLオペレーションを定義しています。WSDMでは、WSRFで規定されるこれらのオペレーションを用いてManageable Resourceのプロパティへのアクセスを実現します。Manageability Endpointは、Manageable Resource自身が直接提供する場合もあれば、Manageable Resourceが提供する管理インタフェースに接続する管理エージェントがManageability Endpointを公開する場合があります (図-4)。前述のように、WSDMは実装非依存という特徴を持っており、どちらもWSDM的には等価です。

さらに、WSDM (MUWS) は、モデル非依存という特徴を持ちます。WSDM (MUWS) は管理対象リソースが保持する管理情報へのアクセス方法を規定する仕様であり、ITリソースの種別ごとにどのようなプロパティ、オペレーション、イベント等をサポートすべきかといった管理情報モデルを規定しません。したがって、WSDM (MUWS) では、CIM (Common Information Model), SNMP/MIB, あるいは独自の管理情報モデル等、既存の管理情報モデルとともに利用します。

Manageability Capability

WSDMでは、Manageable Resourceが提供する管理機能を表現するために、Manageability Capabilityという概念を用います。具体的には、Manageability Capabilityは、Manageable Resourceが提供するプロパティ、オペレーション、イベントなどのセットで表現されます。Manageable Resourceは、自身がサポートする管理機能に相当するManageability Capability

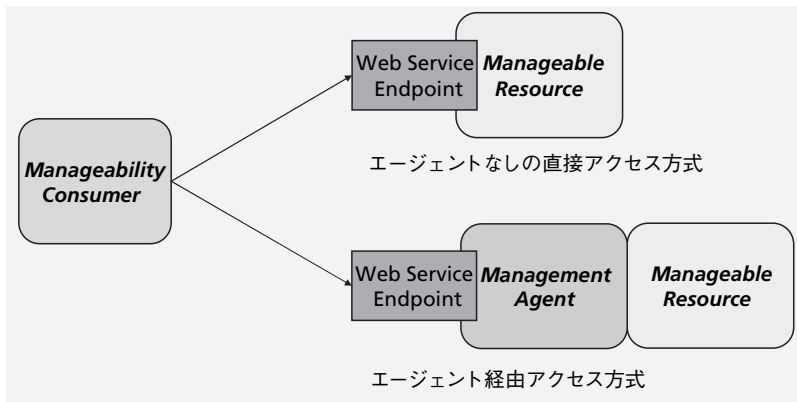


図-4 WSDM の Manageable Resource 実装非依存性

Manageability Capability	概要
Identity	リソースを一意に識別する機能 (必須の Manageability Capability)。
Manageability Characteristics	リソースが提供する Manageability Capability を提供する機能。
Correlatable Properties	リソースが同一であることを判断するために用いるプロパティを示す機能。
Description	リソースに関する記述、バージョンなどを示す機能。
State	リソースの状態、状態遷移を表現する機能。リソースの種別ごとに状態モデルは異なり、MUWS State Capability では具体的な状態、状態遷移を規定せず、それらを表現するためのデータタイプを規定。
Operational Status	リソースの状態を標準的な状態値 (Available, Unavailable, Partially Available, Unknown) で表現する機能。
Metrics	リソースの性能や動作などに関するメトリックを提供する機能。メトリックはリソース種別ごとに異なり、MUWS Metrics Capability では、具体的なメトリックは現在時刻を表す CurrentTime のみを規定。また、メトリックを表現するためのデータタイプ (属性) を規定。
Configuration	リソースの動作を変更する設定情報を提供する機能。リソース種別ごとに設定情報は異なり、MUWS Configuration Capability では具体的なプロパティは規定しない。WSRF の SetResourceProperties オペレーションにより設定情報を変更。
Relationships	リソース間の関連を表現する機能。
Relationship Resource	Relationship を Manageable Resource として提供する機能。
Advertisement	Manageable Resource や Manageability Endpoint の生成・削除を通知する機能。

表-1 MUWS Manageability Capability 一覧

を、ManageabilityCapability リソースプロパティとして公開することができます。MUWS では、リソースの種別に依存しない共通的な基本 Manageability Capability を規定しています。表-1 に MUWS が規定する Manageability Capability の一覧とその概要を示します。

MUWS が規定する Manageability Capability は、Identity Capability を除いてすべてオプションであり、MUWS 準拠の Manageable Resource が最低限サポートしなければならない Manageability Capability は Identity Capability のみです。

Manageable Resource の設計者は、リソース種別に特化した独自の Manageability Capability を、MUWS の基本 Manageability Capability を拡張して作ることができ、また、MUWS 標準 Capability、独自 Capability を自由に組み合わせてサポートすることができます。たとえば、携帯電話や PDA 等の小型デバイスではサポートする管理機能を制限し、アプリケーションサーバ等の処理能力

の大きい IT リソースでは、高機能な管理インタフェースを提供するといった MUWS の使い方が可能です (図-5)。

図-5 において、プリンタを Manageable Resource とした場合では、Identity Capability、Manageability Characteristics Capability、Operational Status Capability の3つの WSDMU 標準 Manageability Capability と、Printer Metrics Capability というリソース独自 Manageability Capability の4つの Manageability Capability を組み合わせてプリンタの管理機能を提供しています。具体的なスキーマ例を図-6 に、WSDL 例を図-7 に示します。

図-6 の点線で囲んだ部分がリソースプロパティドキュメント (Resource Properties Document) です。前述のように、WSDM は WSRF を基盤技術として利用しており、Manageable Resource が公開するプロパティは、WSRF が規定するリソースプロパティドキュメントによって表します。ここで

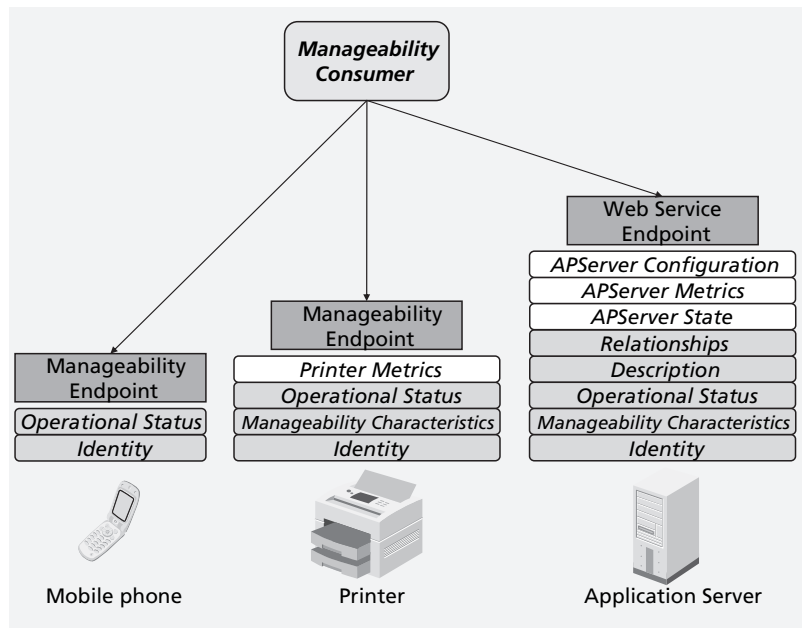


図-5 Manageability Capabilityの組合せ

は、WSDM 標準リソースプロパティとして、Identity Capabilityで規定される ResourceId リソースプロパティ、Manageability Characteristics Capabilityで規定される ManageabilityCapability リソースプロパティ、Operational Status Capabilityで規定される OperationalStatus リソースプロパティ、Metrics Capabilityで規定される CurrentTime リソースプロパティと、リソース独自 Capabilityである Printer Metrics Capabilityが規定する3つの独自リソースプロパティを組み合わせて利用しています。Printer Metrics Capabilityが規定するリソースプロパティは、プリンタを管理するためのメトリックとして利用するためのプロパティです。ここでは、QueuedJobs リソースプロパティは、キューに蓄積されているジョブの数を示すメトリックを、ReceivedRequests リソースプロパティは受信したリクエスト数を示すメトリックを、ServiceTime リソースプロパティは稼働時間を示すメトリックを表します。

Printer Metrics Capabilityは、WSDM (MUWS) 標準の Metrics Capabilityを拡張して定義された Capabilityです。Manageability Capabilityを拡張する場合、ベースとなる Manageability Capabilityが規定するリソースプロパティ、オペレーション、イベントをサポートする必要があります。Printer Metrics Capabilityでは、MUWS Metrics Capabilityで規定される CurrentTime リソースプロパティも提供します。

図-7のWSDLはプリンタの管理機能を提供する Manageability EndpointのWSDLであり、Manageable Resourceに対するアクセスインタフェースを定義し

ています。Manageable Resourceのリソースプロパティドキュメントは、図-7中の点線で記した部分で指定されており、インポートした図-6のスキーマ中のリソースプロパティドキュメントを参照しています。この例では、Manageable Resourceに対するアクセスオペレーションとしては、WSRFが標準的に提供する GetResourceProperty オペレーションのみを利用しています。

今回用いた例では、リソースプロパティの組合せのみの簡単な例でしたが、前述のように、Manageability Capabilityとは特定の管理機能を実現するためのリソースプロパティ、オペレーション、イベントのセットであり、“Manageability Capabilityを組み合わせる”際には、複数

の Manageability Capabilityで規定されているリソースプロパティ、オペレーション、イベントを組み合わせてスキーマ、WSDLを定義することになります。

WSDM 管理イベント

WSDMでは、Manageable Resourceの状態変化を管理イベントとして通知することができます。イベント通知の登録、通知の方法は、WS-Notificationが規定する通知メカニズムを基盤としています。Manageable Resourceがサポートするイベントは、WS-Notification仕様群のWS-Topicが規定するトピックという概念で表され、イベント通知を希望する Manageability Consumerは、Manageable Resourceに対してトピックを指定してイベント通知登録(サブスクライブ)を行うことにより、そのトピックが表すイベントが発生した際に、イベント通知を受信することができます。WSDMでは、ManageabilityCapabilityごとに、イベントが定義されています。たとえば、OperationalStatus Capabilityに対してはOperationalStatusCapabilityトピックが用意されており、OperationalStatus Capabilityに関するイベント(OperationalStatus リソースプロパティの値変化など)を表します。

また、WSDMでは、管理イベント通知メッセージのフォーマット(WSDM Event Format)を規定しています。WSDM Event Formatには、イベント識別情報、イベント発生元リソース識別情報、イベント報告者識別情報、イベントの内容を示すテキストメッセージ、イベン

```
<xs:schema
  targetNamespace="http://example.org/services/MyPrinter.xsd"
  xmlns:tns="http://example.org/services/MyPrinter.xsd"
  xmlns:muws-p1-xs="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-part1.xsd"
  xmlns:muws-p2-xs="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-part2.xsd"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:import
    namespace="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-part1.xsd"
    schemaLocation="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-part1.xsd"/>
  <xs:import
    namespace="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-part2.xsd"
    schemaLocation="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-part2.xsd"/>
  <xs:import
    namespace="http://example.org/services/localDefinitions.xsd"
    schemaLocation="http://example.org/services/localDefinitions.xsd"/>
```

```
<xs:element name="QueuedJobs" type="tns:NonNegativeMetric"/>
<xs:element name="ReceivedRequests" type="tns:NonNegativeMetric"/>
<xs:element name="ServiceTime" type="tns:DurationMetric"/> リソース独自メトリック
```

```
<xs:complexType name="NonNegativeMetric">
  <xs:simpleContent>
    <xs:extension base="xs:nonNegativeInteger">
      <xs:attributeGroup ref="muws-p2-xs:MetricAttributes"/>
      <xs:anyAttribute namespace="##other" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="DurationMetric">
  <xs:simpleContent>
    <xs:extension base="xs:duration">
      <xs:attributeGroup ref="muws-p2-xs:MetricAttributes"/>
      <xs:anyAttribute namespace="##other" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

リソース独自メトリック用データタイプ定義

```
<xs:element name="MyPrinterProperties">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="muws-p1-xs:ResourceId"/>
      <xs:element ref="muws-p1-xs:ManageabilityCapability"
        minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="muws-p2-xs:OperationalStatus"/>
      <xs:element ref="muws-p2-xs:CurrentTime"/>
      <xs:element ref="tns:ReceivedRequests"/>
      <xs:element ref="tns:QueuedJobs"/>
      <xs:element ref="tns:ServiceTime"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

Resource Properties Document

図-6
プリンタリソースの
スキーマ

ト発生日時、イベント分類、シーケンス番号、イベント間の関連情報、メッセージカタログ情報等と、イベントごとに特化した通知メッセージ内容を含みます。図-8にWSDM イベントフォーマット例を示します。WSDMにおける管理イベントメッセージは、WS-Notificationが規定するNotifyメッセージ中のMessage要素以下に、WSDM イベントフォーマットのメッセージが配置されたかたちで通知されます。

Webサービスの管理 (MOWS)

今日の企業情報システムにおいては、異種システム間連携、柔軟なシステム構築などの観点で、Webサービスの担う役割は大きくなっています。また、SOAに基

づき、ビジネスプロセスを構成する各サービスや、各種グリッドサービスなど、Webサービスを管理する技術の標準化が重要になってきています。すでに述べましたように、MOWSは、MUWSを基盤として、管理対象をWebサービスとしたMUWSの拡張仕様です。

MOWSでは、管理対象Webサービスは、Webサービスエンドポイントの粒度を基本単位として扱います。たとえば、特定のエンドポイントの利用可能状態、メッセージ受信数等を監視するためにエンドポイントの単位で管理します。一方、あるサービスが複数のエンドポイントで提供されており、エンドポイントの単位ではなく論理的なサービスの単位でWebサービスを管理したい場合も考えられます。この場合、MUWSが規定するRelationshipを用いて、複数のエンドポイント単

```

<wsdl:definitions
targetNamespace="http://example.org/services/MyPrinter.wsdl"
xmlns:printer="http://example.org/services/MyPrinter.xsd" ...>

<wsdl:import namespace="http://docs.oasis-open.org/wsrf/2004/06
/wsrf-WS-ResourceProperties-1.2-draft-01.wsdl"
location= ... />
<wsdl:import
namespace="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-part2.wsdl"
location= ... />
...
<wsdl:types>
<xs:schema>
<xs:import namespace="http://example.org/services/MyPrinter.xsd"
schemaLocation= ... />
<xs:import namespace="http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceProperties-1.2-draft-01.xsd"
schemaLocation= ... />
<xs:import namespace="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-part1.xsd"
schemaLocation= ... />
<xs:import namespace="http://docs.oasis-open.org/wsdm/2004/12/muws/wsdm-muws-part2.xsd"
schemaLocation= ... />
...
</xs:schema>
</wsdl:types>
...
<wsdl:portType name="MyPrinterPortType" [wsrf-rp:ResourceProperties="printer:MyPrinterProperties"]>
Resource Properties Document 参照
<wsdl:operation name="GetResourceProperty">
<wsdl:input name="GetResourcePropertyRequest" ... />
<wsdl:output name="GetResourcePropertyResponse" ... />
<wsdl:fault name="ResourceUnknownFault" ... />
<wsdl:fault name="InvalidResourcePropertyQNameFault" ... />
</wsdl:operation>
</wsdl:portType>
ポートタイプ定義

<wsdl:binding name="MyPrinterSoapOverHttpBinding" type="printerw:MyPrinterPortType">
<soapw:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
<wsdl:operation name="GetResourceProperty">
...
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="MyPrinterService">
<wsdl:port name="MyPrinterSoapPort" binding="printerw:MyPrinterSoapOverHttpBinding">
<soapw:address location="http://example.org/services/MyPrinterEndpoint"/>
</wsdl:port>
</wsdl:service>

</wsdl:definitions>

```

図-7 プリンタリソースの WSDL

位の Manageable Resource (Manageable Web Service Endpoint) を関連付けることによって、論理的なサービスの単位での管理を実現することができます。

MOWS では、Manageable Resource が Web サービスであり、Manageable Resource が提供する管理インタフェースも Web サービスであるため、これら 2 つの Web サービスエンドポイントを同一の Web サービスエンドポイントとすることも可能です (図-9)。

MOWS では、MUWS が規定する基本 Manageability Capability を拡張した Manageability Capability を定義しています。表-2 に MOWS が規定する Manageability Capability の一覧とその概要を示します。MUWS Manageability Capability を拡張した MOWS Manageability Capability の例として、State Capability と Metrics Capability について紹介します。

MUWS State Capability は、Manageable Resource の状態を表現する機能を提供する Manageability Capability です。しかし、リソースの状態モデルは、リソース種別に依存するため、リソースの種別に非依存な管理機能を定義する MUWS では、具体的なリソースプロパティは規定しません。MUWS State Capability では、汎用的なリソースの状態、状態遷移を表現するためのデータタイプのみを規定しています (図-10)。MOWS Operational State Capability では、図-10 のデータタイプを用いて、Web サービスに特化したリソースプロパティを規定しています (図-11)。MUWS Metrics Capability は、Manageable Resource のメトリックを表現する機能を提供する Manageability Capability です。しかし、リソースのメトリックは、リソース種別に依存するため、リソース種別に非依存な管理機能を定義する

```

...
<soapenv:Body>
<wsn:Notify>
<wsn:NotificationMessage>
<wsn:Topic Dialect="http://docs.oasis-open.org/wsn/2004/06/TopicExpression/Simple">
muws-p2-events:OperationalStatusCapability
</wsn:Topic>
<wsn:ProducerReference>
<wsa:Address ... > ... </wsa:Address>
<wsa:ReferenceProperties>
...
</wsa:ReferenceProperties>
<wsa:PortType> ... </wsa:PortType>
<wsa:ServiceName PortName= ... />
</wsn:ProducerReference>
<wsn:Message>
<muws-p1-xs:ManagementEvent ReportTime="2006-09-11T15:53:18+09:00">
<muws-p1-xs:EventId>
urn:uuid:70721500-04b4-11da-b6a8-c7e9640ea2f1
</muws-p1-xs:EventId>
<muws-p1-xs:SourceComponent/>
<muws-p2-xs:Situation>
<muws-p2-xs:SituationCategory>
<muws-p2-xs:AvailabilitySituation/>
</muws-p2-xs:SituationCategory>
<muws-p2-xs:SituationTime>"2006-09-11T15:52:18+09:00"</muws-p2-xs:SituationTime>
<muws-p2-xs:Message>"off-line"</muws-p2-xs:Message>
<muws-p2-xs:Severity>4</muws-p2-xs:Severity>
</muws-p2-xs:Situation>
<wsrf-rp:ResourcePropertyValueChangeNotification>
<wsrf-rp:OldValue>
<muws-p2-xs:OperationalStatus>Available</muws-p2-xs:OperationalStatus>
</wsrf-rp:OldValue>
<wsrf-rp:NewValue>
<muws-p2-xs:OperationalStatus>Unavailable</muws-p2-xs:OperationalStatus>
</wsrf-rp:NewValue>
</wsrf-rp:ResourcePropertyValueChangeNotification>
</muws-p1-xs:ManagementEvent>
</wsn:Message>
</wsn:NotificationMessage>
</wsn:Notify>
</soapenv:Body>
...

```

図-8
WSDM イベントフォーマット例

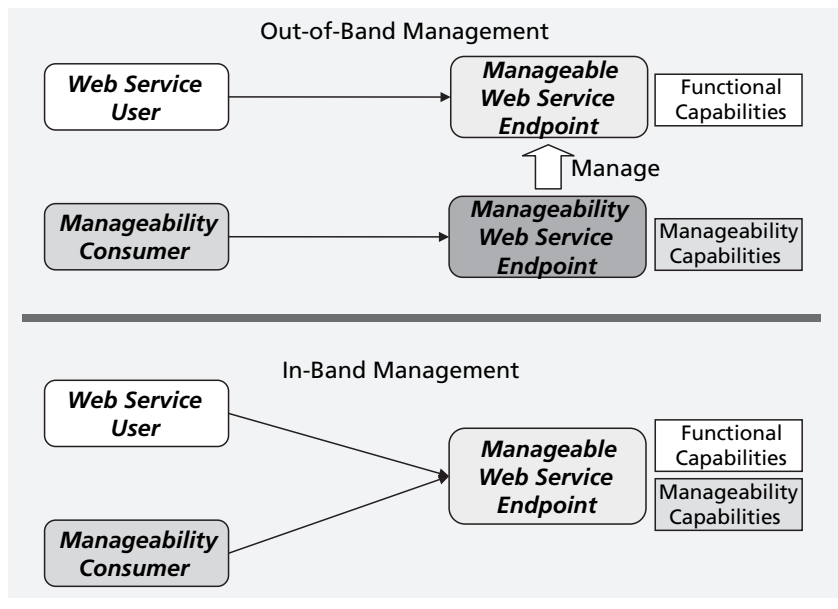


図-9 MOWSによるWebサービス管理

MUWSでは、具体的なメトリック(リソースプロパティ)は規定しません。MUWS Metrics Capabilityでは、リソース種別に依存しない現在時刻を表すCurrentTime リソースプロパティと、汎用的なデータタイプ(属性)のみ

を規定しています(図-12)。MOWS Metrics Capabilityでは、図-12のデータタイプ(MetricAttributes)を用いて、Webサービスに特化したメトリックを規定しています(図-13)。

Apache Muse (WSDMオープンソース)

WSDMのオープンソース実装としてApache Museがあります。Apache Museは、2004年にオープンソースソフトウェア開発団体Apacheに設立されたMuseプロジェクトにおいて、HPの開発者が中心となって開発が始まったオープンソースWSDM実装

です。Apache Museは、同時期にApache内に設立されたWSRFプロジェクト、Subscribeプロジェクトでそれぞれ開発されていたWSRF実装Apache WSRF、WS-Notification実装Apache Subscribeを基盤ソフトウェア

Manageability Capability	概要
Identification	管理対象 Web サービスエンドポイントに関する情報を示す機能。
Metrics (MUWS Metrics 拡張)	Web サービスの性能, 使用状況を監視するための機能。
Operation Metrics (MUWS Metrics 拡張)	Web サービスの, オペレーション単位での性能, 使用状況を監視するための機能。
Operational State	W3C Web Services Architecture Management Task Force が規定する Web サービスの状態を管理する機能。
Operational Status (MUWS Operational Status 拡張)	3C Web Services Architecture Management Task Force が規定する Web サービスの状態を MUWS Operational Status Capability が規定する基本的な状態にマッピングする機能。
Operation Operational Status (MUWS Operational Status 拡張)	オペレーションごとの状態を管理するための機能。
Request Processing State	管理対象 Web サービスが受信したリクエストの処理状態を管理するための機能。
Manageability References	自身の管理インターフェースを提供する Manageability Endpoint を示す機能 (管理対象 Web サービスエンドポイントの Manageability Capability)。

表-2 MOWS Manageability Capability 一覧

```

<xs:element name="State" type="muws2:StateType" />
<xs:element name="StateTransition" type="muws2:StateTransitionType" />

<xs:complexType name="StateType">
  <xs:complexContent>
    <xs:extension base="muws2:CategoryType" />
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="CategoryType">
  <xs:sequence>
    <xs:any minOccurs="0" namespace="##any" processContents="lax" />
  </xs:sequence>
</xs:complexType>

<xs:complexType name="StateTransitionType">
  <xs:sequence>
    <xs:element ref="muws2:EnteredState" />
    <xs:element ref="muws2:PreviousState" minOccurs="0" />
    <xs:any minOccurs="0" maxOccurs="unbounded" namespace="##other" processContents="lax" />
  </xs:sequence>
  <xs:attribute name="TransitionIdentifier" type="xs:anyURI" use="optional" />
  <xs:attribute name="Time" type="xs:dateTime" use="required" />
  <xs:anyAttribute namespace="##other" />
</xs:complexType>

<xs:element name="EnteredState" type="muws2:StateType" />
<xs:element name="PreviousState" type="muws2:StateType" />

```

図-10
MUWS State Capability
規定データタイプ

として利用する形態として開発が行われ、2005年8月に Muse1.0 としてリリースされています。

WSDM の普及のためには、すでに広く普及している既存管理技術と WSDM との連携機能が必要になってきます。筆者は Apache Muse プロジェクトの contributor (開発者) となり、Java で実装されたリソースを管理するための標準技術としてすでに広く普及している JMX と WSDM とを連携する機能を開発しました。

その後、Apache WSRF, Pubsub, Muse の 3 プロジェクトは統合されることとなり、また、IBM の開発者から提供された新たなソースコードを基に Muse1.0 を置き換えるかたちで開発が新たに進められ、現在、2006年9月に Muse 2.0 としてリリースされています。

Web サービス管理標準の最近の動向

現在、WSDM の基盤仕様である WSRF, WS-Notification は、OGF の OGSA WSRF Basic Profile として参照されており、また、WSDM, WSRF, WS-Notification は OASIS 標準仕様となっています。しかし、WSRF, WS-Notification, WSDM と競合する仕様群である WS-Transfer, WS-Enumeration, WS-Eventing, WS-Management が Microsoft, Intel 等から提案されており、これまで、業界での仕様の一本化は実現されていない状況が続いていました。

このような状況を打開すべく、2006年3月、両仕様群を統合して新しい Web サービス管理標準仕様を策定する計画が HP, IBM, Intel, Microsoft から発表されました。新しい仕様は、WS-Transfer, WS-Enumeration,


```

<xs:complexType name="OperationalStateType">
  <xs:complexContent>
    <xs:extension base="muws2:StateType"/>
  </xs:complexContent>
</xs:complexType>

<xs:element name="UpState">
  <xs:complexType>
    <xs:complexContent>
      <xs:restriction base="mows:OperationalStateType"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>

<xs:element name="IdleState">
  <xs:complexType>
    <xs:complexContent>
      <xs:restriction base="mows:OperationalStateType">
        <xs:sequence>
          <xs:element ref="mows:UpState"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
</xs:element>

<xs:element name="BusyState">
  <xs:complexType>
    <xs:complexContent>
      <xs:restriction base="mows:OperationalStateType">
        <xs:sequence>
          <xs:element ref="mows:UpState"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
</xs:element>

<xs:element name="DownState">
  <xs:complexType>
    <xs:complexContent>
      <xs:restriction base="mows:OperationalStateType"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>

```

```

<xs:element name="StoppedState">
  <xs:complexType>
    <xs:complexContent>
      <xs:restriction base="mows:OperationalStateType">
        <xs:sequence>
          <xs:element ref="mows:DownState"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
</xs:element>

<xs:element name="CrashedState">
  <xs:complexType>
    <xs:complexContent>
      <xs:restriction base="mows:OperationalStateType">
        <xs:sequence>
          <xs:element ref="mows:DownState"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
</xs:element>

<xs:element name="SaturatedState">
  <xs:complexType>
    <xs:complexContent>
      <xs:restriction base="mows:OperationalStateType">
        <xs:sequence>
          <xs:element ref="mows:DownState"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
</xs:element>

<xs:element name="CurrentOperationalState"
  type="mows:OperationalStateType"/>
<xs:element name="LastOperationalStateTransition"
  type="muws2:StateTransitionType"/>

```

図-11 MOWS State 拡張データタイプ

```

<xs:element name="CurrentTime" type="xs:dateTime" />

<xs:attributeGroup name="MetricAttributes">
  <xs:attribute name="ResetAt" type="xs:dateTime" />
  <xs:attribute name="LastUpdated" type="xs:dateTime" />
  <xs:attribute name="Duration" type="xs:duration" />
</xs:attributeGroup>

```

図-12 MUWS Metrics Capability 規定データタイプ

WS-Eventing を基盤仕様とし、WSRF、WS-Notification で規定される主要な概念は、WS-ResourceTransfer、WS-EventNotification として、新たな仕様スタックに追加されることとなりました。この発表では、新しい仕様の完成までには 18～24 カ月かかるといわれています。図-14 に両仕様スタックの統合イメージを示します。なお、新たにリリースされる計画が発表された仕様のうち、WS-ResourceTransfer、WS-MetadataExchange1.1 はすでにリリースされています。

おわりに

WSDM と WS-Management との統合計画が発表され

たことにより、現在の OASIS 標準 WSDM1.1 が、そのまま業界で広く普及する可能性は低くなったといえるでしょう。しかし、本稿で述べました WSDM の基本的な概念、特に Manageability Capability という基本管理機能を組み合わせて IT リソースの Web サービス管理インタフェースを設計するフレームワークは、相互接続性、システム構築の容易化という点で大変有用であり、新 Web サービス管理統一仕様で同様の概念が規定される可能性があると考えます。

また、今後、Web サービス管理標準が広く普及するためには、既存管理技術 (CIM, SNMP, JMX 等) との連携を容易にするオープンソースソフトウェア等も重要な役割を担うと考えます。

```

<xs:element name="NumberOfRequests" type="mows:IntegerCounter"/>
<xs:element name="NumberOfSuccessfulRequests" type="mows:IntegerCounter"/>
<xs:element name="NumberOfFailedRequests" type="mows:IntegerCounter"/>
<xs:element name="ServiceTime" type="mows:DurationMetric"/>
<xs:element name="MaxResponseTime" type="mows:DurationMetric"/>
<xs:element name="LastResponseTime" type="mows:DurationMetric"/>
<xs:element name="MaxRequestSize" type="mows:IntegerCounter"/>
<xs:element name="LastRequestSize" type="mows:IntegerCounter"/>
<xs:element name="MaxReponseSize" type="mows:IntegerCounter"/>
<xs:element name="LastReponseSize" type="mows:IntegerCounter"/>

<xs:complexType name="IntegerCounter">
<xs:simpleContent>
<xs:extension base="xs:nonNegativeInteger">
<xs:attributeGroup ref="muws2:MetricAttributes"/>
<xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:extension>
</xs:simpleContent>
</xs:complexType>

<xs:complexType name="DurationMetric">
<xs:simpleContent>
<xs:extension base="xs:duration">
<xs:attributeGroup ref="muws2:MetricAttributes"/>
<xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:extension>
</xs:simpleContent>
</xs:complexType>

```

図-13 MOWS Metrics 拡張データタイプ

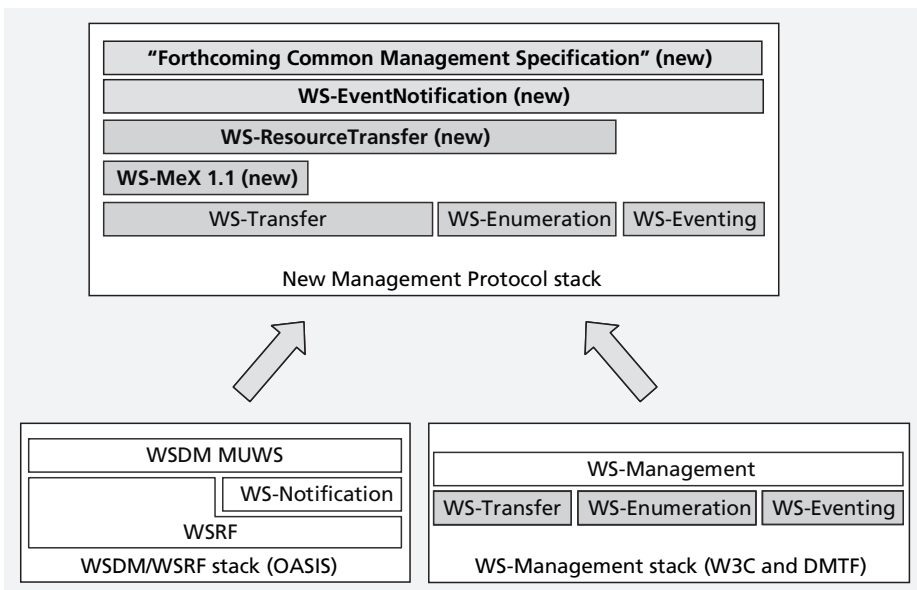


図-14 WSDM と WS-Management の統合

冒頭に述べた現在のシステム運用管理の課題を解決し、さらに、ビジネスグリッドコンピューティングが目指す自律的なシステム運用管理の実現のためには、Web サービス技術に基づく標準管理技術は必要不可欠であり、Web サービス管理標準仕様の早期実現、普及が待ち望まれます。

参考文献

- 1) OASIS WSDM TC Web ページ : http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsdm
- 2) OASIS WSRF TC Web ページ : http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsrf
- 3) OASIS WS-Notification TC Web ページ : http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsn
- 4) OGF Web ページ : <http://www.ogf.org/>

- 5) Apache Muse プロジェクト Web ページ : <http://ws.apache.org/muse/index.html>
- 6) Toward Converging Web Service Standards for Resources, Events, and Management : http://download.boulder.ibm.com/ibmdl/pub/software/dw/webservices/Harmonization_Roadmap.pdf
(平成 18 年 11 月 29 日受付)

加藤英晴 (正会員)
hideharu.kato.mq@hitachi.com

1997 年名古屋大学大学院理学研究科物質理学専攻修士課程修了、同年日本電信電話(株)入社、2002 年(株)日立製作所入社、現在、システム開発研究所においてシステム運用管理技術、Web サービス関連技術の研究開発に従事。