



文書識別番号: DSP2017

日付: 2014/04/24

バージョン: 2.0.0

オープン仮想化フォーマットのホワイト・ペーパー

文書の種類: DMTF 情報

文書の位置付け: DMTF 情報

文書の言語: ja-JP

著作権情報

Copyright © 2007, 2016 Distributed Management Task Force, Inc. (DMTF). All rights reserved.

DMTF は、企業やシステムの管理および相互運用性を推進することに力を注いでいる、業界のメンバーから成る非営利団体である。メンバー、およびメンバー以外でも、出典を正しく表示することを条件に、DMTF の仕様と文書を複製することができる。DMTF の仕様は時折改定されることがあるため、特定のバージョンおよび公開日に、常に注意を払う必要がある。

本標準または標準案の特定のエレメントを実装することは、仮特許権を含む第三者の特許権(本書では「特許権」と呼ぶ)の対象となることもある。DMTF は本標準のユーザーに対し、上記権利の存在について何ら表明するものではなく、上記第三者の特許権、特許権者または主張者の、いずれかまたはすべてを認識、公開、または特定する責任を負わない。また、上記権利、特許権者、主張者の不完全または不正確な特定、公開に対しても責任を負わない。DMTF は、いかなる相手に対して、いかなる方法または環境、またいかなる法論理においても、上記の第三者特許権を認識、公開、または特定しないことに対し何ら責任を負わず、上記第三者の標準に関する信頼性、またはその製品、プロトコル、試験方法論に組み込まれた標準に関しても何ら責任を負わない。DMTF は、上記標準の実装が知見できるか否かにかかわらず、上記標準を実装するいかなる相手に対しても、また、いかなる特許権者または主張者に対しても、何ら責任を負わない。また、DMTF は、公開後に標準が撤回または修正されることにより生じるコストや損失に対し何ら責任を負わず、また、標準を実装するいかなる相手からも、上記実装に対して特許権者が起こす、いずれかまたはすべての侵害の主張から何ら損害を受けず、免責されるものとする。

第三者が保有する特許権であって、DMTF 標準の実装に関連するかまたは影響を与える可能性があるの特許権者が考え、すでに DMTF に通知済みである特許権に関する情報については、サイト <http://www.dmtf.org/about/policies/disclosures.php> を参照のこと。

概要

このオープン仮想化フォーマット(OVF)のホワイト・ペーパーでは、オープン仮想化フォーマット(OVF)標準を構成する仕様である DSP0243、DSP8023、および DSP8027 の適用について説明する。OVF パッケージおよび具体的なユースケースへのその適用について理解することを目的とするすべての人が本書の対象読者である。ここでは、仮想化と CIM モデルの一般的な概念に読者がある程度精通していることを前提にしている。

1

目次

2	1	はじめに	7
3	1.1	概説	7
4	1.2	設計上の留意事項	8
5	2	OVF の主要な概念	10
6	2.1	仮想アプライアンス	10
7	2.2	ライフサイクル	11
8	3	XML Backgrounder	13
9	3.1	XML スキーマ	13
10	3.2	OVF で使用する一般的な XML の概念	14
11	3.2.1	要素	14
12	3.2.2	属性	14
13	3.2.3	代替グループ	14
14	4	OVF パッケージ	15
15	4.1	OVF デスクリプタ	15
16	4.2	OVF スキーマで定義するグローバル属性	16
17	4.3	OVF の拡張性	16
18	4.4	OVF 最上位の要素	17
19	4.4.1	VirtualSystem 要素	20
20	4.4.2	VirtualSystemCollection 要素	20
21	4.4.3	References 要素	20
22	4.4.4	DiskSection 要素	20
23	4.4.5	NetworkSection 要素	21
24	4.4.6	DeploymentOptionsSection 要素	24
25	4.4.7	SharedDiskSection 要素	27
26	4.4.8	PlacementGroupSection 要素	29
27	4.5	仮想システムおよび仮想システムのコレクションで使用する OVF の Section 要素	29
28	4.5.1	AnnotationSection 要素	29
29	4.5.2	ProductSection 要素	30
30	4.5.3	EulaSection 要素	31
31	4.5.4	VirtualHardwareSection 要素	31
32	4.5.5	PlacementSection 要素	33
33	4.5.6	EncryptionSection 要素	37
34	4.6	仮想システム・コレクションで使用する OVF Section 要素	40
35	4.6.1	ResourceAllocationSection 要素	40
36	4.6.2	StartupSection 要素	41
37	4.6.3	ScaleOutSection 要素	43
38	4.7	仮想システムで使用する OVF Section 要素	45
39	4.7.1	OperatingSystemSection 要素	45
40	4.7.2	InstallSection 要素	46
41	4.7.3	EnvironmentFilesSection 要素	46
42	4.7.4	BootDeviceSection 要素	47
43	5	OVF パッケージのオーサリング	48
44	5.1	作成	48
45	5.2	国際化	49
46	5.3	拡張性	51
47	5.3.1	代替グループ	51
48	5.3.2	要素	53
49	5.3.3	属性	54
50	5.4	適合	54
51	5.5	仮想ハードウェアの記述	54
52	5.6	デスクリプタの例	56

53	6	OVF パッケージのデプロイ	57
54	6.1	デプロイ	57
55	6.2	OVF 環境デスクリプタ	57
56	6.3	デプロイ時のリソース構成オプション	58
57	6.4	Property 要素を使用したデプロイ時の製品のカスタマイズ	60
58	7	可搬性	62
59		ANNEX A (参考情報) 単一仮想システムの例	64
60		ANNEX B (参考情報) 多層 Pet Store の例	67
61	B.1	アーキテクチャとパッケージ化	67
62	B.2	プロパティ	68
63	B.3	ディスクのレイアウト	69
64	B.4	Pet Store OVF デスクリプタ	70
65	B.5	OVF 環境の全体	77
66		ANNEX C (参考情報) 単一仮想システムの LAMP スタックの例	79
67	C.1	デプロイ時のカスタマイズ	79
68	C.2	単純な LAMP OVF デスクリプタ	82
69		ANNEX D (参考情報) 複数仮想システムの LAMP スタックの例	86
70	D.1	2 層 LAMP の OVF デスクリプタ	86
71		ANNEX E (参考情報) 拡張性の例	93
72	E.1	カスタム・スキーマ	93
73	E.2	カスタム拡張を使用したデスクリプタ	94
74		ANNEX F (参考情報) 変更履歴	96
75			
76		図の一覧	
77		図 1 - OVF パッケージのライフサイクル	12
78		図 2 - OVF の作成機能	12
79		図 3 - OVF のデプロイ機能	13
80		図 4 - OVF パッケージの構造	15
81		図 5 - ネットワーク接続	22
82		図 6 - LAN-SAN ネットワーク接続	22
83		図 7 - アフィニティ配置	34
84		図 8 - 可用性配置	35
85		図 9 - アフィニティおよび可用性配置	37
86		図 B-1 - Pet Store OVF パッケージ	67
87		図 B-2 - Pet Store 仮想ディスクのレイアウト	69
88			
89		表の一覧	
90		表 B-1 - Web 層の構成	68
91		表 B-2 - データベース層の構成	68
92		表 C-1 - LAMP 構成	79
93			
94			

95

謝辞

96 オープン仮想化フォーマットのホワイト・ペーパー(DSP2017)は DMTF の OVF ワークグループにより編纂され
97 た。

98 この DMTF 情報仕様は、以下の諸氏をはじめとする個人やチームによる共同作業の賜物である。

99	Lawrence Lamers	VMware Inc.(委員長)
100	Marvin Waschke	DMTF Fellow(共同編集者)
101	Peter Wörndle	Ericsson AB(共同編集者)
102	Eric Wells	Hitachi, Ltd.(共同編集者)
103		
104	Hemal Shah	Broadcom Corporation
105	Shishir Pardikar	Citrix Systems Inc.
106	Richard Landau	DMTF Fellow
107	Robert Freund	Hitachi, Ltd.
108	Jeff Wheeler	Huawei
109	Monica Martin	Microsoft Corporation
110	Cheng Wei	Microsoft Corporation
111	Srinivas Maturi	Oracle
112	Steffen Grarup	VMware Inc.
113	Rene Schmidt	VMware Inc.
114	Ghazanfar Ali	ZTE Corporation
115		

116 オープン仮想化フォーマットのホワイト・ペーパー

117 1 はじめに

118 1.1 概説

119 オープン仮想化フォーマット(OVF)仕様は、仮想システムに基づくソフトウェア・ソリューションの標準パッケージ
120 化フォーマットを業界に提供し、ソフトウェア・ベンダーやクラウド・コンピューティング・サービス提供者に不可欠な
121 ビジネス・ニーズに応えるものである。

122 OVF パッケージを使用することにより、独立系ソフトウェア・ベンダー(ISV)はソフトウェア・ソリューションを公開で
123 き、データ・センターのオペレーターはデータ・センター間でソフトウェア・ソリューションを移動でき、顧客はソフト
124 ウェア・ソリューションをアーカイブできる。このほか、ソフトウェア・ソリューションの標準化パッケージを用意するこ
125 とで目的を達成できるあらゆるユースケースで、このパッケージを使用できる。

126 OVF 作業の主な基盤となるユースケースは以下のとおりである。

127 1) 複数のハイパーバイザーで使用可能なソフトウェア・ソリューションをパッケージできるようにする ISV
128 の機能

129 2) 仮想システムまたはそのコレクションをパッケージして、データ・センター間で移動できるようにする機能

130 その他のユースケースやそこから派生したユースケース(サブセット)もこれに該当する。

131 OVF バージョン 1 は、業界で広く採用され、現在は国際標準となっている。

132 OVF バージョン 2 では、パッケージ化機能が強化され、業界がクラウド・コンピューティングの時代に入るに伴って
133 出現してきた広範囲なユースケースに適応できるようになっている。

134 OVF 2 では以下の機能が追加される。

- 135 ● ネットワーク・ポートのサポート
- 136 ● デプロイ時の拡張
- 137 ● 基本的な配置ポリシーのサポート
- 138 ● OVF パッケージの暗号化
- 139 ● 実行時のディスク共有
- 140 ● 高度なデバイス・ブート順序
- 141 ● ゲスト・ソフトウェアへの高度なデータ転送
- 142 ● 向上した国際化(I18N)のサポート
- 143 ● HASH サポートの改善
- 144 ● 更新された CIM スキーマ

145 OVF では必要に応じて共通情報モデル(CIM)を採用しており、管理ソフトウェアでオープン標準を使用して、リ
146 ソースのプロパティを明確に把握し、容易にマップできるようにしている。特定のデバイス・タイプの
147 CIM_ResourceAllocationSettingData クラスとそのサブクラスは、仮想システムの動作に必要なリソー
148 スの指定に使用する。

149 OVF 2 は、ネットワーク・ポート・プロファイル(DSP8049)を使用するネットワーク構成および同様の IEEE Edge

150 Virtual Bridging の発見/構成プロトコルをサポートする。CIM_EthernetPortAllocationSettingData クラスは不
151 可欠なプロパティを提供する。

152 本書は、OVF の動機、目標、設計、および想定される用途の詳細説明を目的としている。本書と併せて、同じメ
153 ジャー・リビジョンの OVF 仕様に目を通して理解しておく必要がある。

154 1.2 設計上の留意事項

155 仮想インフラストラクチャの急速な採用に伴い、仮想化プラットフォーム上および仮想プラットフォーム間での仮想
156 システムの配布に使用できる標準的で可搬性のあるメタデータ・フォーマットの必要性に注目が集まっている。ソ
157 フトウェア・アプリケーションのテスト、開発、実稼働のプロセスを経る中で、ソフトウェア・アプリケーションとその稼
158 働が認定されているオペレーティング・システムをまとめ、事前に構成およびパッケージ化された外部依存性のな
159 いユニットとして容易に転送できるフォーマットにパッケージ化できれば、非常に魅力的である。実行を必要とする
160 仮想システムの構成とともにパッケージ化した、事前デプロイ済みでいつでも実行可能なアプリケーションは仮想
161 アプライアンスと呼ばれる。この概念を広範な規模で実用的なものにするには、このような仮想アプライアンスお
162 よびそれらアプライアンスをあらゆる仮想化プラットフォームに自動的かつ安全にインストール、構成、および実行
163 するために必要なメタデータをパッケージ化するための、特定のベンダーに依存しない標準を業界で採用するこ
164 とが重要である。

165 ユーザーから見れば、OVF は仮想アプライアンスのパッケージ化フォーマットである。インストールした OVF パッ
166 ケージは、特定のサービスを提供する自己完結型で自己矛盾のないソフトウェア・アプリケーションをユーザーの
167 インフラストラクチャに追加する。たとえば、OVF パッケージには、LAMP スタック(Linux + Apache + MySQL +
168 PHP)のような完全に機能する、テスト済みの Web サーバー、データベース、OS の組み合わせを含むことができる
169 ほか、ウィルス・チェッカーやそのアップデート・ソフトウェア、スパイウェア検出機能などを含むこともできる。

170 多くの仮想アプライアンスが持つ仮想システムは 1 つのみであるが、最新のエンタープライズ・アプリケーションは、
171 それぞれが 1 つ以上の仮想システムを持つ階層を複数備えたサービス指向アーキテクチャ(SOA)としてモデル
172 化される。したがって、多層サービスを提供するには、仮想システムが 1 つのみのモデルでは不十分である。また、
173 複雑なアプリケーションでは、ネットワークや顧客固有のプロパティをインストール時にカスタマイズする必要があ
174 る。さらに、仮想アプライアンスは、特定のハイパーバイザーに適したディスク・イメージおよび構成データとともに、
175 実行時フォーマットでパッケージ化される。実行時フォーマットは、実行用に最適化したものであり、配布を目的と
176 したものではない。効率的なソフトウェアの配布のためには、可搬性、プラットフォームからの独立性、検証、署名、
177 バージョニング、ライセンス条件など多数の側面が重要になる。

178 OVF 仕様は、1 つ以上の仮想システムで構成した仮想アプライアンスのパッケージ化と配布のために、特定のハ
179 イパーバイザーに依存せず、効率的で拡張可能なオープン・フォーマットを記述している。この仕様は、個々の仮
180 想システムだけではなく、機能ユニットとしての仮想アプライアンスについても、自動的で安全な管理を容易にす
181 ることを目標にしている。

182 このような面での成果を目指し、仮想化プラットフォーム・ベンダーだけではなく、ISV、仮想アプライアンス・ベン
183 ダー、オペレーティング・システム・ベンダーも OVF を開発し、承認している。さまざまなベンダーの仮想化プラッ
184 フォーム間で仮想システムの可搬性や相互乗り入れを実現する共通基準を共同開発する活動を通じ、OVF 仕様
185 は顧客からの信頼向上を促進している。

186 OVF の目標は、ただちに有用性を発揮すること、直近のビジネス・ニーズに対応すること、および仮想アプライ
187 アンスのパッケージ化を目指して下位互換性を備えながらも機能豊富な共通フォーマットの迅速な採用を促進する
188 ことにある。

189 OVF 仕様は、既存の IT 管理標準とフレームワークを補完し、開示性と拡張性を通じて、最善の組み合わせによ
190 る切磋琢磨を促進するものである。本書に記載されている明示的な著作権情報は、この仕様の自由な配布と実
191 用を許可する一方で、フォーマットに対する恣意的、独自の、断片的な拡張を回避することを目的にしている。

192

193

194 2 OVF の主要な概念

195 2.1 仮想アプライアンス

196 仮想アプライアンスは構成済みのソフトウェア・スタックであり、1 つ以上の仮想システムで構成する。各仮想シ
197 テムは単独でインストール可能な実行時エンティティであって、オペレーティング・システム、アプリケーション、ア
198 プリケーション固有のデータで構成するほか、仮想システムで必要とする仮想ハードウェアを記述するメタデー
199 ータも含んでいる。多くのインフラストラクチャ・アプリケーションのほか、ネットワーク経由でアクセス可能なエンドユー
200 ザー・アプリケーションも、仮想アプライアンスとして提供できる。このようなアプリケーションとして、DNS サー
201 バーやバグ追跡データベースのほか、Web 層、アプリケーション層、データベース層で構成する完全な CRM ソ
202 リューションも考えられる。複雑なソフトウェア・システムやサービスを構成済みのソフトウェア・スタックとして提供
203 することにより、堅牢性が飛躍的に高くなり、インストールを簡素化できる。

204 仮想アプライアンスを使用することで、特定アプリケーションのソフトウェア・スタックを最適化し、エンド・ユーザ
205 ーにターンキー・サービスを提供できる。このことから、ソフトウェアの配布パラダイムが変化している。ソリューシ
206 ョン提供者にとって、仮想アプライアンスの構築は、ハードウェア・アプライアンスの構築よりも単純でコスト効率に
207 優れている。アプリケーションが使用するオペレーティング・システムとともにアプリケーションが事前にパッケージ
208 化されていることから、互換性テストと認証が軽減される。これにより、ソフトウェアを開発する ISV は、実行環境
209 にソフトウェアを事前インストールできる。エンド・ユーザー側では、仮想アプライアンスを使用することで、標準化
210 された効率的な自動プロセスの採用を通じて、ソフトウェア管理のライフサイクルを大幅に簡略化できる。この自
211 動プロセスは、これまで使用されていた OS やアプリケーションに固有の管理タスクに替わるものである。

212 サードパーティの ISV が、仮想アプライアンスを開発し、提供する必要はない。特定のサービスで使用する仮想シ
213 ステム・テンプレートの作成、テスト、認証を IT 組織が担当し、同一のデプロイを全社的に実現できるようにパッ
214 ケージ化している企業でも、この概念は有用であり、またよく使用される。

215 一般に、ソフトウェア・サービスは、複数の仮想システムで実行して SOA モデルによってネットワーク経由で通信
216 する多層アプリケーションとして実装される。多くの場合、サービスは他の複数のサービスで構成され、それらの
217 サービス自体が多層アプリケーションであることや、さらに別のサービスで構成されることもある。実際、SOA モ
218 デルは仮想アプライアンス・ベースのインフラストラクチャに無理なく適合する。その理由は、仮想アプライアンスで
219 は、ネットワークを経由した XML ベースの管理およびサービス・インターフェースを使用することが普通で、これに
220 より、さまざまなアプライアンスを組み合わせて、1 つの完成されたアプリケーションを提供できることにある。

221 たとえば、プレゼンテーション・ロジックを実装する Web 層、ビジネス・ロジックを実装するアプリケーション・サー
222 ーバー層、バックエンド・データベース層の 3 層で構成する典型的な Web アプリケーションがあるとする。わかりやす
223 い実装では、この構成は各層に 1 つずつ、合計 3 つの仮想システムに分けられる。この方法によるアプリケーション
224 の規模は、1 つの物理ホストの一部分から始まり、3 つの物理ホストまで考えられる。また、各層をそれ自体の
225 サービスとして扱うという考え方もあり得る。したがって、各層の規模は、クラスター化したソリューションを提供す
226 る複数の仮想システム・サービスとしたものまで考えることができる。多数の Web サーバー、少数のアプリケー
227 ション・サーバー、および 1 つか 2 つのデータベース・サーバーで構成する別の Web アプリケーションを例に挙げる。
228 仮想システムとして実装した各層は、必要な数の物理マシンに合わせ、弾力的に拡大・縮小することができる。ま
229 た、各層は、サービスが要求する複数の仮想システム・インスタンスをサポートできる。

230 1 つの OVF には、1 つ以上の仮想システムを置くことができる。アプリケーションに最適な構成の見極めは、開発
231 者の判断に委ねられている。OVF を実行するには、その OVF をインストールする必要がある。OVF から仮想シ

232 テムを実行する仮想プラットフォームもあり得るが、そのような構成は必ずしも必要ではない。この構成を選択し
233 た場合、OVF自体をアプライアンスのゴールデンイメージとみなすことはできなくなる。その理由は、仮想システム
234 の実行時状態がOVFに行き渡ることにある。さらに、プラットフォームがOVFの整合性チェックに使用する電子署
235 名が無効になる。

236 トランスポート機能としてのOVFは、VMwareのVMDK仮想ディスク・フォーマット、MicrosoftのVHD仮想ハード・
237 ディスク・フォーマット、オープン・ソースのQCOWフォーマットのいずれとも異なる。これらは実行時仮想システ
238 ム・イメージ・フォーマットであり、1つの仮想ディスクの範囲内で動作するフォーマットである。これらはトランス
239 ポート・フォーマットとして広く使用されているが、可搬性の問題を解決するために設計されたものではない。1つ
240 の仮想システムで複数のディスクを有している場合や複数の仮想システムの場合には効果がない。そのような状
241 況に対応するには、インストールの際に仮想システムのカスタマイズが必要である。複数の仮想化プラットフォーム
242 上での実行を目的とした仮想システムでは、これらのフォーマットは役に立たない(特定の仮想ハード・ディス
243 ク・フォーマットのサポートをうたっている仮想化プラットフォームであっても同様である)。

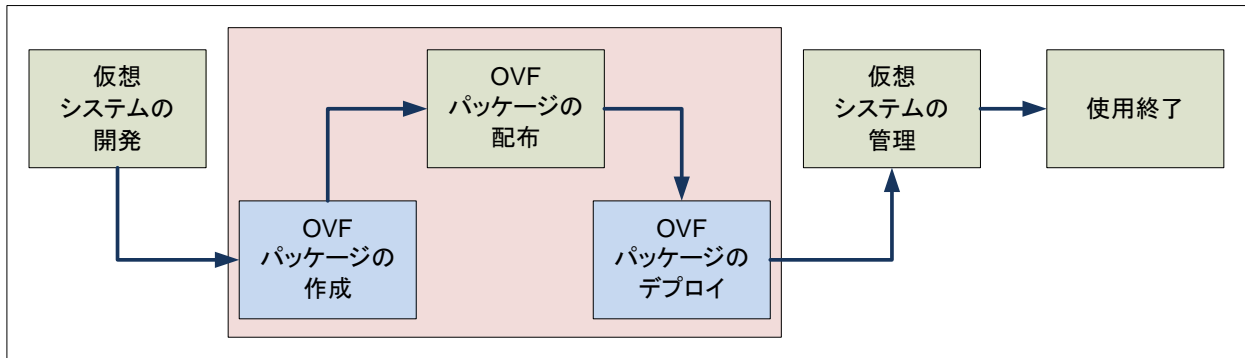
244 OVFでは、パッケージ化した仮想アプライアンスの認証と整合性の概念まで効力が及ぶ。この概念により、プラッ
245 トフォームはアプライアンスの出所を判断でき、エンド・ユーザーは適切な信頼性に基づく判断を下すことができる。
246 OVF仕様は、アプライアンスが自身の構成と変更を担当するように作られている。これは、アプライアンスのファ
247 イル・システムを読み取る機能を仮想化プラットフォームに必要としないようにすることを特に意図したものである。
248 このようにプラットフォームとアプライアンスを分離することで、どのようなオペレーティング・システムを使用しても
249 OVFパッケージを実装でき、またそのOVFフォーマットをサポートしているすべての仮想化プラットフォームに
250 OVFパッケージインストールできる。アプライアンスには、それがインストールされているプラットフォームを検知し、
251 反応するための特別な機能が用意される。この機能により、プラットフォームは、業界全体にわたるアプライア
252 ンスの互換性を維持しながら、この仕様を独自の方法で拡張できる。

253 OVFフォーマットには、複雑な多層サービスとそれに関連する配布、インストール、構成、および実行のために設
254 計された特別な機能がある。このような機能として、以下のものがある。

- 255 • 多層アプリケーションの構成と仮想システムの合成を直接サポートし、合成されたサービスを提供す
256 る。
- 257 • 仮想システムとアプリケーション・レベルの構成の両方の仕様を許可する。
- 258 • OVFの内容を検証するための堅牢な機能を備え、無人インストールを全面的にサポートすることによ
259 り、デプロイ時のユーザーの負荷を軽減し、これによってユーザー・エクスペリエンスの向上を図る。
- 260 • 署名および信頼済みサードパーティの使用を通じて、商業的に認められた手順を使用して、OVFの内
261 容の整合性を確認する。これは、アプライアンスの作成者が署名した時点以降、アプライアンスが改変
262 されていないことを利用者に保証する役割を果たす。この保証は、仮想アプライアンス市場の成功と、
263 アプライアンスの自主的な作成やオンライン・ダウンロードの実行可能性にとって重要なものととらえら
264 れる。
- 265 • アプライアンスに関連するライセンス条件の提示と合意のための基本的な手段を提供することで、ア
266 プライアンス・ベンダーとユーザーの商業的利益が尊重されるようにする。

267 2.2 ライフサイクル

268 図1に仮想システムのライフサイクルを示す。



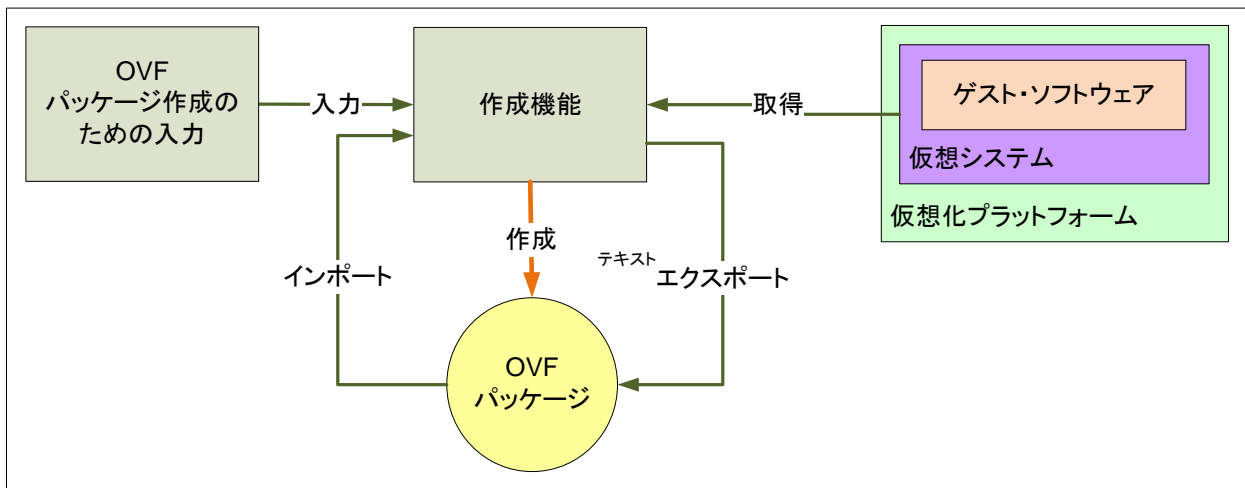
269

270

図 1 - OVF パッケージのライフサイクル

271 OVF パッケージは OVF 作成者が開発または入手したコンポーネントで構築される。これらのコンポーネントは、
 272 仮想アプライアンスを構成するひと揃いのファイルにパッケージ化され、このパッケージは、1 つ以上の仮想マシン
 273 と仮想マシン・コレクションおよび関連する構成とデプロイ・メタデータで構成される。たとえば、クラスター化した
 274 データベース・コンポーネントにはサードパーティの ISV から入手できるものがある。インストールしたサービスは
 275 管理対象となり、最後は使用終了となる。配布、管理、および使用中止は OVF で扱う範囲にはなく、OVF で使用
 276 する仮想化製品およびインストールした仮想アプライアンスに固有のものである。管理には、アプライアンスに対
 277 して現在実施している保守、構成、およびアップグレードの各作業がある。このような作業は、インストールした
 278 サービスと環境によって決まり、OVF パッケージには左右されない。具体的には、OVF 仕様の重点は作成段階と
 279 デプロイ段階にある。

280 図 2 に OVF の作成機能を示す。



281

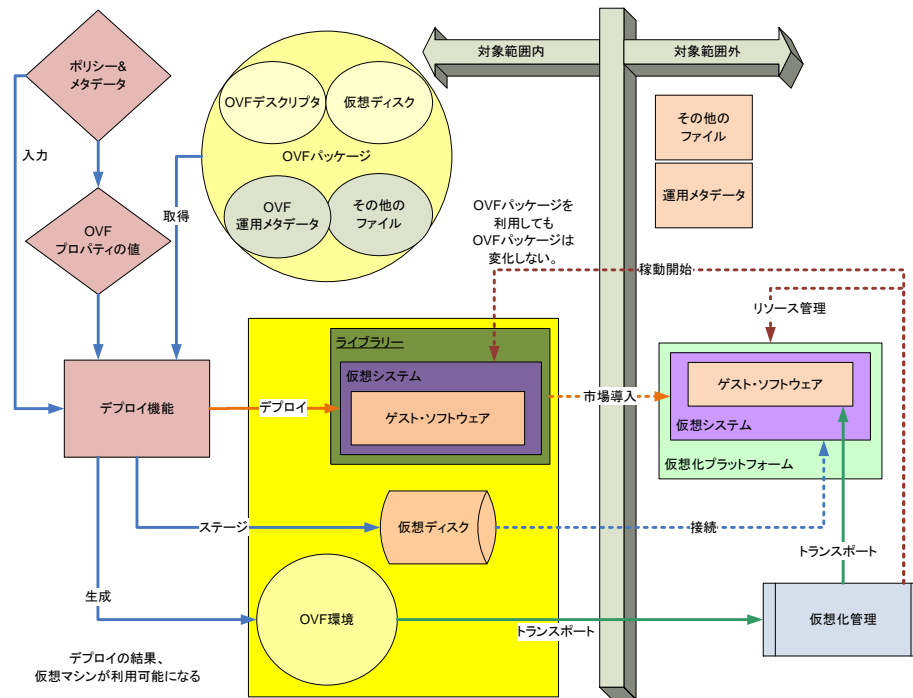
282

図 2 - OVF の作成機能

283 OVF パッケージを作成する方法には 2 種類がある。わかりやすい方法としては、テキスト・エディターまたは XML
 284 オーサリング・ツールを使用して OVF デスクリプタを作成し、必要なディスク・イメージとその他のファイルをまとめ
 285 たうえで、OVF パッケージを含んだ tar ファイルまたはファイル・システムを作成する。

286 もう 1 つは、仮想化プラットフォームから OVF パッケージをエクスポートする方法である。エクスポート後、OVF デ
 287 スクリプタを編集し、情報を追加する。仮想化プラットフォーム間での可搬性の向上や構成オプションの提供など、
 288 さまざまな理由からこの方法が選択されている。

289 図 3 に OVF のデプロイ機能を示す。この図からは、OVF 作業部会が扱っている作業範囲も知ることができる。



290

291

図 3 - OVF のデプロイ機能

292 OVF 運用メタデータは、仮想システムまたは仮想システムのコレクションが適切に動作するうえで必要と考えられ
 293 る情報である。OVF 運用メタデータは、仮想システムの運用を開始すると利用できるようになる、運用メタデー
 294 ータのサブセットである。

295 この図に示すように、OVF デプロイ機能は、OVF 環境を仮想化プラットフォームにトランスポートする。OVF 仕様
 296 は、トランスポートの厳密な性質に対しては柔軟である。この場合のトランスポートは、仮想マシンの仮想リーダ
 297 ーに CD-ROM などのメディアを置き、システムが起動するたびにゲスト・オペレーティング・システムがこのメディア
 298 を読み取ることと考えることができる。OVF 環境のメタデータは、オペレーティング・システム起動後の構成で使
 299 し、仮想アプライアンスが適切に動作するうえでゲスト・ソフトウェアや仮想化プラットフォームに求められる要件
 300 が満たされるようにする。

3 XML Backgrounder

3.1 XML スキーマ

303 OVF 標準は、XML および XML スキーマ定義言語 (XSD または XSDL) を使用する。XML は、XML 要素とその
 304 属性を表現する文法を定義したマークアップ言語であるが、ドキュメントでの要素や属性の構造、要素や属性の
 305 値で使用できるデータ型などをほとんど定義していない。ドキュメント型定義 (DTD) は XML が誕生したときから
 306 XML の一部であり、XML に構造とデータ・タイプを追加することを目的としていたが、さまざまな状況で不十分で
 307 あることが指摘されていた。これに対応するため、W3C は XSD を開発した。XSD 自体が XML で記述されており、
 308 XML 文書の構造とデータ・タイプの定義の面で充実した内容の言語である。XSD と DTD はまとめて使用でき
 309 るが、最近では XSD が広く使用されるようになっている。XSD ファイルには、慣例上、拡張子 ".xsd" が使用されて
 310 る。

311 XSD ファイルは、XML 文書で使用できる要素の構造とデータ型および文書のデータ構造を記述したメタデータで
312 ある。また、XSD は継承などオブジェクト指向の構造もサポートしている。このような構造により、綿密で複雑な
313 XSD の作成を簡素化できるので、記述が容易になり、ファイルを小型化できて、エラーの発生も低減できる。XML
314 文法に基づく XML 文書の確認では文書の文法のみが検証可能で、文書の構造やデータの正確さは検証できな
315 い。XSD に基づく XML 文書の確認では、構造とデータを検証できる。他のプロセスで XML 文書を使用する場合、
316 XSD に対して有効な文書は、エラーなく処理される可能性が高くなる。また、XSD から XML 文書プロセッサを
317 生成できるツールもある。このようなツールを使用すると、開発時間の短縮が実現する。

318 OVF では、XSD を使用して OVF デスクリプタの構造とデータ型を指定する。OVF デスクリプタは、仮想システム
319 のインスタンス化に使用する恣意的で複雑なパターンを記述した XML 文書であり、OVF パッケージに含められる。
320 OVF デスクリプタのファイル拡張子は ".ovf" である。OVF 2.0.0 の正式な XSD は
321 <http://schemas.dmtf.org/ovf/envelope/2/dsp8023.xsd> にある。OVF のユーザーは、XSD 検証ツールを使用し、
322 OVF の XSD スキーマに基づいてそれぞれの OVF デスクリプタを確認する必要がある。問題がないことが検証さ
323 れたデスクリプタは、OVF 標準への適合に対する必要条件ではあるが、十分条件ではない。この標準仕様の制
324 約の中には dsp8023.xsd で扱っていないものもあるので、全面的な適合とするには、それらを手作業で確認する
325 必要がある。

326 3.2 OVF で使用する一般的な XML の概念

327 3.2.1 要素

328 XML 要素は OVF デスクリプタのデータ・コンテナである。空ではない要素は開始タグで始まり、これに要素の内
329 容が続き、最後は終了タグで終わる。開始タグは、<elementname>や<elementname attributes>のように要素
330 名、または要素名と要素の属性を<>で囲んだものである。終了タグは、</elementname>のように、/>と要素名を
331 <>で囲んだものである。要素には、他の要素、テキスト、属性、またはこれらを組み合わせたものを記述できる。

332 XML 要素の名前は、以下の命名規則に従う。

- 333 • 名前には、文字、数字、その他の記号を使用できる。
- 334 • 名前の先頭に数字と句読文字は使用できない。
- 335 • 名前の先頭に「xml」(XML や Xml も同様)を付けることはできない。
- 336 • 名前にはスペースを使用できない。
- 337 • 予約語はなく、どのような名前も使用できる。

338 3.2.2 属性

339 XML 要素は属性を持つことができる。属性には、その要素に関する補足情報を記述する。属性は、データには含
340 まれていない情報を提供することが普通である。属性値は引用符または二重引用符で囲む必要がある。

341 3.2.3 代替グループ

342 代替グループは XSD のオブジェクト指向機能の 1 つである。スキーマによって文書の中に生成された要素を置き
343 換える要素を、代替グループで指定できる。置き換え可能な要素はヘッド要素と呼ばれ、スキーマのグローバル
344 スコープで定義する。代替グループの要素は、ヘッド要素と同じタイプであるか、ヘッド要素のタイプから派生した
345 タイプである。

346 要するに、代替グループを使用すると、汎用的な要素に代わる要素のコレクションを構築できる。たとえば、3 種
347 類の商品を販売する会社向けの発注システムを構築する場合、この 3 種類の商品に共通するデータを記述した

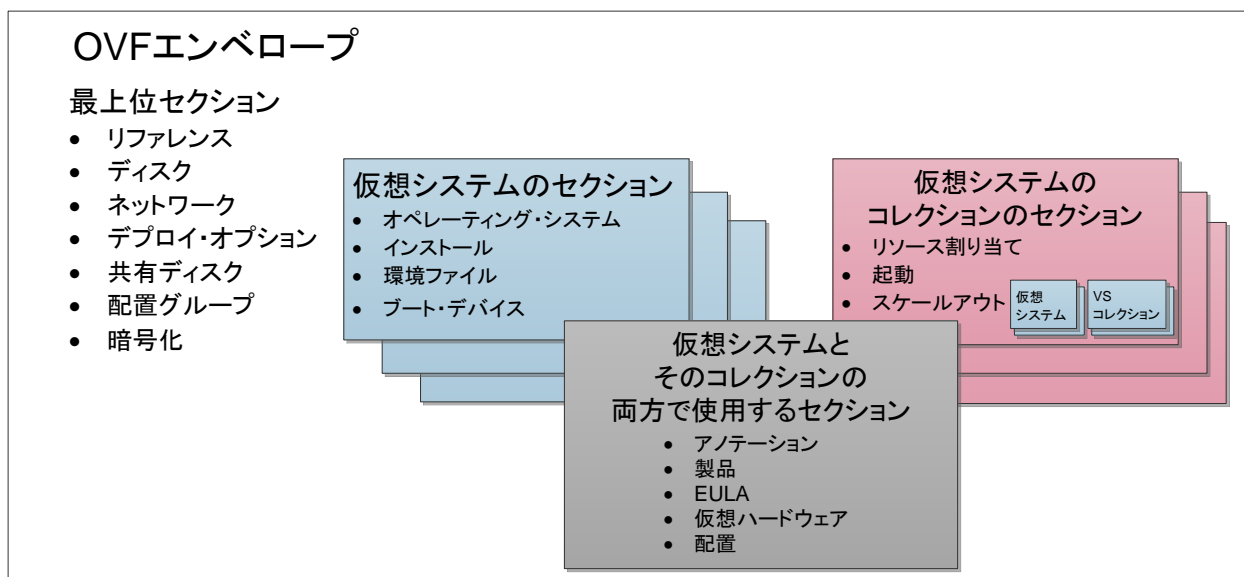
348 汎用商品要素を定義する。続いて、汎用商品から派生するより具体的な商品タイプを記述した、汎用商品に代わ
 349 るグループを定義する。このスキーマでは、汎用商品のシーケンスとして注文を定義できる。このスキーマに準拠
 350 する XML ファイルでは、注文は汎用商品のシーケンスではなく、代替グループにある商品のシーケンスとなる。
 351 代替グループのヘッド要素は、抽象的な情報として定義することが普通なので、汎用商品は XML ファイルには現
 352 れない。

353 4 OVF パッケージ

354 OVF パッケージは、1 つの仮想システムまたは複数の仮想システムのコレクションにデプロイするソフトウェア・ソ
 355 リューションを配布する方法を提供する。OVF パッケージは、OVF デスクリプタと関連する仮想ディスクで構成す
 356 る。この OVF パッケージは、URL で参照するファイルの集合または拡張子が'.ova'の圧縮ファイルとして存在す
 357 る。

358 4.1 OVF デスクリプタ

359 OVF デスクリプタは XML ファイルの一種である。OVF デスクリプタのルート要素は Envelope である。Envelope
 360 要素の子要素のうち最も重要な 2 つは VirtualSystem 要素と VirtualSystemCollection 要素である。
 361 Envelope 要素には、パッケージの VirtualSystem 要素と VirtualSystemCollection 要素ですべて
 362 に適用されるセクションがある。Envelope 要素には、VirtualSystem 要素と
 363 VirtualSystemCollection 要素の両方を記述できる。VirtualSystemCollection 要素は再帰構造
 364 であり、Envelope 要素のように、VirtualSystem 要素と VirtualSystemCollection 要素の両方を記
 365 述できる。OVF スキーマは、さまざまなセクションを定義する。このようなセクションの中には、Envelope 要素に
 366 のみ記述するものがある。また、VirtualSystem 要素にのみ記述するものもあり、
 367 VirtualSystemCollection element にのみ記述するものもある。一方で、VirtualSystem と
 368 VirtualSystemCollection elements の両方に記述するものもある。この構造をまとめたものが図 4 であ
 369 る。



370

371

図 4 - OVF パッケージの構造

372 4.2 OVF スキーマで定義するグローバル属性

373 OVF スキーマで定義する以下の OVF 属性はグローバル属性である。OVF 要素固有の OVF 属性も定義する
374 (4.4 および 4.3 を参照)。

- 375 • `required` 属性 - 要素が存在しない場合または認識されない場合、デプロイは失敗することを示す。
376 この属性は XSD Boolean であり、値には 'true'、'false'、'0'、'1' を指定できる。指定しなかった場合の値
377 は 'true' または '1' である。この OVF 属性を XSD `use` 属性の XSD 値 `required` と混同しないようにす
378 る必要がある。この 2 つの属性は、用語が似ているが重要度が異なる。
- 379 • `transport` 属性 - ゲスト・ソフトウェアに情報を伝達するために使用できるトランスポート・タイプをス
380 ペースで区切って並べたリストである。4.7.3 および 6.2 を参照。
- 381 • `configuration` 属性 - `DeploymentOptionSection` 要素の `Configuration` 要素で定義し
382 た構成を特定する。4.4.6 を参照。`configuration` 属性は、以下の場所で使用する。
 - 383 • `Item`、`EthernetPortItem`、`StorageItem` の各要素で
384 `VirtualHardwareSection` 要素に使用
 - 385 • `Item`、`EthernetPortItem`、`StorageItem` の各要素で
386 `ResourceAllocationSection` 要素に使用
 - 387 • `InstanceCount` 要素で `ScaleOutSection` 要素に使用
 - 388 • `Property` 要素で `ProductSection` 要素に使用
- 389 • `bound` 属性 - リソース割り当て設定データの最小値、標準値、最大値を示すために使用する範囲マー
390 カー・エントリー。使用できる値は 'min'、'normal'、および 'max' である。4.6.1 を参照。`bound` 属性は、リ
391 ソース割り当ての制限を設定するために、以下の 2 か所で使用する。
 - 392 • `Item`、`EthernetPortItem`、`StorageItem` の各要素で
393 `VirtualHardwareSection` 要素に使用
 - 394 • `Item`、`EthernetPortItem`、`StorageItem` の各要素で
395 `ResourceAllocationSections` 要素に使用

396 4.3 OVF の拡張性

397 OVF スキーマは、XSD 要素 '`any`' および '`anyAttribute`' を使用して、OVF デスクリプタと OVF 環境を拡張し、
398 カスタム・メタデータを提供する。この機能により、OVF パッケージは業界のさまざまなユースケースに対応でき
399 る。

400 以下の定義は、XML スキーマ・リファレンスの Web サイトからの引用である。

401 http://www.w3schools.com/schema/schema_elements_ref.asp を参照。

- 402 • `any` - この定義により、スキーマで指定されていない要素を使用して XML 文書を拡張できる。
- 403 • `anyAttribute` - この定義により、スキーマで指定されていない属性を使用して XML 文書を拡張で
404 きる。
- 405 • `##any` - あらゆる名前空間にある要素を許可する(これはデフォルトの定義である)。
- 406 • `##other` - 親要素の名前空間以外であれば、どのような名前空間にある要素でも使用できる。

407 `Envelope` 要素レベルの拡張とは、`OVF:Section` 代替グループの新しいメンバーを定義することである。

408 `Content` 要素レベルの拡張とは、`OVF:Section` 代替グループの新しいメンバーを定義することである。3.2.3 を

409 参照。これらの新しい Section 要素は、OVF スキーマによりセクションの存在が許されている場合に使用できる。
 410 新しい Section 要素の Info 要素は、新しい Section 要素がデプロイ・プラットフォームで認識されずにスキップさ
 411 れたときに、ユーザーにわかりやすく警告するために使用できる。

412 OVF スキーマで定義したタイプは、末尾に要素を付加して、拡張できる。拡張ポイントは、
 413 namespace="##other"として xs:any を使用して宣言する。

414 OVF スキーマでは、追加属性が認められている。拡張ポイントの宣言には xs:anyAttribute を使用する。

415 ovf:required 属性は、要素の情報が必須であるか省略可能であるかを表す。ovf:required 属性のデ
 416 フォルトは TRUE である。必須である要素拡張がデプロイ・プラットフォームで検知されても、それがそのプラット
 417 フォームで認識できない場合、デプロイは失敗する。

418 正しく動作するために、カスタム属性の情報が必要となることはない。

419 EXAMPLE 1:

```
420 <!-- Optional custom section example -->
421 <otherns:IncidentTrackingSection ovf:required="false">
422   <Info>Specifies information useful for incident tracking purposes</Info>
423   <BuildSystem>Acme Corporation Official Build System</BuildSystem>
424   <BuildNumber>102876</BuildNumber>
425   <BuildDate>10-10-2008</BuildDate>
426 </otherns:IncidentTrackingSection>
```

427 EXAMPLE 2:

```
428 <!-- Open content example (extension of existing type) -->
429 <AnnotationSection>
430   <Info>Specifies an annotation for this virtual machine</Info>
431   <Annotation>This is an example of how a future element (Author) can still be
432     parsed by older clients</Annotation>
433   <!-- AnnotationSection extended with Author element -->
434   <otherns:Author ovf:required="false">John Smith</otherns:Author>
435 </AnnotationSection>
```

436 EXAMPLE 3:

```
437 <!-- Optional custom attribute example -->
438 <Network ovf:name="VM network" otherns:desiredCapacity="1 Gbit/s">
439   <Description>The main network for VMs</Description>
440 </Network>
```

441 4.4 OVF 最上位の要素

442 OVF スキーマで定義するルート要素は Envelope 要素である。Envelope 要素の直接の子である OVF 要素を、
 443 OVF デスクリプタに記述する順序で以下に示す。

- 444 • References 要素
- 445 • Section 要素 - Section 要素の代替グループ
- 446 • Content 要素 - Content 要素の代替グループ
- 447 • Strings 要素

448 Envelope 要素の直接の子である Section 要素の代替グループである要素:

- 449 • DiskSection 要素
- 450 • NetworkSection 要素
- 451 • DeploymentOptionSection 要素
- 452 • SharedDiskSection 要素
- 453 • PlacementGroupSection 要素
- 454 • EncryptionSection 要素

455 Content 要素の代替グループである要素:

- 456 • VirtualSystem 要素
- 457 • VirtualSystemCollection 要素(ネスト可能)

458 Content 要素で使用する Section 要素の代替グループである要素を、OVF デスクリプタに記述する順序で以下
459 に示す。

- 460 • AnnotationSection 要素
- 461 • ProductSection 要素
- 462 • OperatingSystemSection 要素
- 463 • EulaSection 要素
- 464 • VirtualHardwareSection 要素
- 465 • ResourceAllocationSection 要素
- 466 • InstallSection 要素
- 467 • StartupSection 要素
- 468 • EnvironmentFilesSection 要素
- 469 • BootDeviceSection 要素
- 470 • ScaleOutSection 要素
- 471 • PlacementSection 要素

472 OVF スキーマで定義されている追加要素を以下に示す。なお、これらの要素は `ovf:` 以外の名前空間でも使用
473 できる。

- 474 • Annotation
- 475 • AppUrl
- 476 • bootc: CIM_BootConfigSetting
- 477 • Category
- 478 • Configuration
- 479 • Content
- 480 • Description
- 481 • Disk
- 482 • EthernetPortItem
- 483 • File
- 484 • FullVersion
- 485 • Icon
- 486 • Info
- 487 • InstanceCount
- 488 • Item
- 489 • Label
- 490 • License

- 491 • Msg
- 492 • Name
- 493 • Network
- 494 • NetworkPortProfile
- 495 • NetworkPortProfileURI
- 496 • Product
- 497 • ProductUrl
- 498 • Property
- 499 • SharedDisk
- 500 • StorageItem
- 501 • System
- 502 • Value
- 503 • Vendor
- 504 • VendorUrl
- 505 • Version
- 506 • xenc:EncryptedKey
- 507 • xenc11:DerivedKey

508 図 4 に示した OVF デスクリプタの基本構造の例を以下に示す。

509

```
510 ovf:Envelope
511 <xs:element name="References" type="ovf:References_Type">
512
513 <xs:element ref="ovf:Section" minOccurs="0" maxOccurs="unbounded">
514 <DiskSection>
515   <Info> Describes all virtual disks used with the package </Info>
516 <NetworkSection>
517   <Info>List of logical networks used in the package</Info>
518 <DeploymentOptionSection>
519   <Info>List of deployment options available in the package</Info>
520
521 <xs:element ref="ovf:Content">
522   <VirtualSystemCollection ovf:id="Acme VSC">
523     <Info>The packaging of the first virtual appliance</Info>
524     -----
525     <VirtualSystem ovf:id="Acme VS 1">
526       <Info>The packaging of the virtual machine 1</Info>
527     <VirtualSystem ovf:id="Acme VS 2">
528       <Info>The packaging of the virtual machine 2</Info>
529     -----
530     <VirtualSystemCollection ovf:id="Widget VSC">
531       <Info>The packaging of the second virtual appliance</Info>
532     <VirtualSystem ovf:id="Acme VS 3">
533       <Info>The packaging of the virtual machine 3</Info>
534     <VirtualSystem ovf:id="Acme VS 3">
```

```

535 <Info>The packaging of the virtual machine 4</Info>
536 -----
537 <xs:element name="Strings" type="ovf:Strings_Type" minOccurs="0" maxOccurs="unbounded">
538 <Info> Root element of I18N string bundle</Info>

```

539 4.4.1 VirtualSystem 要素

540 VirtualSystem 要素は、Content 要素の代替要素である。この要素には、1つの仮想システムを定義する多
541 数の Section 要素が含まれる。このような Section 要素は、仮想システムに適用可能な仮想ハードウェア、リ
542 ソース割り当て、製品情報を記述する。

543 4.4.2 VirtualSystemCollection 要素

544 VirtualSystemCollection 要素は、Content 要素の代替要素である。この要素には、1つ以上の
545 VirtualSystem 要素と、仮想システムのコレクションを定義する多数の Section 要素が含まれる。このような
546 Section 要素は、仮想システムのコレクションに適用可能な仮想ハードウェア、リソース割り当て、製品情報を記
547 述する。

548 4.4.3 References 要素

549 References 要素には、あらゆる外部ファイルへの参照が含まれる。

550 4.4.4 DiskSection 要素

551 DiskSection 要素は、OVF パッケージで仮想システムが使用する仮想ディスクを定義する。

552 仮想ディスク・フォーマット仕様が制約なく公開され、利用可能であれば、どのような仮想ディスク・フォーマットで
553 も使用できる。このフォーマットは、現在のハイパーバイザーで使用するあらゆる種類の仮想ハード・ディスク・
554 フォーマットをサポートしており、将来登場するフォーマットにも拡張可能である。

555 この仮想ディスク・フォーマットとしては、インストールされているゲスト・ソフトウェアには関係なく、簡素で基本的
556 なディスク・ブロック・フォーマットが使用できる。たとえば、VMware VMDK フォーマットは、さまざまなフラット、ス
557 パース、圧縮の形式で64 KBブロックに格納された512バイトのディスク・セクターを扱う。仮想化プラットフォーム
558 は、デプロイの際に、最適と考えられる基本的なディスク・ブロック・フォーマットで仮想ディスクを作成する。実行
559 時の仮想ディスク・フォーマットは配布フォーマットと同じものとすることもできるが、異なるフォーマットとす
560 ることが普通である。これは、圧縮したままの仮想ディスク・フォーマットを使い切ることが非効率であることが考えら
561 るからである。インストールされているゲスト・ソフトウェアは、NTFS、EXT3、ZFS など独自のファイル・システム・
562 フォーマットを使用している。しかし、OVF 仮想ディスクがこれらのファイル・システム・フォーマットを認識してい
563 なくても問題はない。

564 以下の例で仮想ディスクについて説明する。

```

565 <DiskSection>
566 <Info>Describes the set of virtual disks</Info>
567 <Disk ovf:diskId="vmdisk1" ovf:fileRef="file1" ovf:capacity="8589934592"
568     ovf:populatedSize="3549324972"
569     ovf:format=
570         "http://www.vmware.com/interfaces/specifications/vmdk.html#sparse">
571 </Disk>

```

```

572 <Disk ovf:diskId="vmdisk2" ovf:capacity="536870912"
573 </Disk>
574 <Disk ovf:diskId="vmdisk3" ovf:capacity="{disk.size}"
575     ovf:capacityAllocationUnits="byte * 2^30"
576 </Disk>
577 </DiskSection>

```

578 4.4.5 NetworkSection 要素

579 NetworkSection 要素は、OVF パッケージのデプロイに使用するネットワークおよびネットワーク接続を記述す
 580 る。ネットワークは、Red、Green、Blue ネットワークのように簡素化した語で定義してもよい。イーサネット・ポート
 581 の特性は CIM_EthernetPortAllocationSettingData クラスで定義する。この特性は、ネットワーク・
 582 ポート・プロファイル (DSP8049) を使用して詳細に定義することもできる。このプロファイルは OVF パッケージに
 583 含むことができる。

584 ここでは、イーサネット・ポートの接続先としてフラット・レイヤー2 ネットワークが使用できることが基本的な前提に
 585 なっている。このネットワークのサービスや特性に関する前提はない。イーサネット・ポートの速度は、
 586 CIM_EthernetPortAllocationSettingData を使用して設定できる。

587 4.4.5.1 OVF ネットワーク

588 以下に、NetworkSection で使用する OVF デスクリプタのスニペット例を示す。このコードでは、仮想システムで定
 589 義しているイーサネット・ポートを単に "Red" ネットワークに接続することを示している。epasd:Connection プロ
 590 パティは、イーサネット・ポートの接続先ネットワークを指定する。ここでは、すべてのイーサネット・ポートをレイ
 591 ヤー2 ネットワークに接続するという簡単な前提としている。たとえば、"Red" と "Green" のように複数のネットワ
 592 クが定義されている場合、"Red" ネットワークと "Green" ネットワークは互いに接続しないものとする。データ・セン
 593 ターの LAN や WAN のような外部ネットワークへの接続は、デプロイのプロビジョニングの一環とみなされるので、
 594 OVF デスクリプタでは指定しない。この場合、ネットワークの特性とサービスは、OVF パッケージの利用側とホス
 595 ティング・サービスとの間での取り決めによって決まる。

```

596 <!-- Describes all networks used in the package -->
597 <NetworkSection>
598     <Info>Logical networks used in the package</Info>
599     <Network ovf:name="Red Network">
600         <Description>The network that the virtual systems are attached to.
601         on</Description>
602     </Network>
603 </NetworkSection>
604

```

605 以下の例は、VirtualSystem 要素の OVF デスクリプタ VirtualHardwareSection に記述したアイテムの
 606 スニペットである。ここでは、CIM_EthernetPortAllocationSettingData クラスの Connection プロパ
 607 ティを使用して、仮想システムをどのように "Red" ネットワークに接続するかを示している。

```

608 <EthernetPortItem>
609     <epasd:AddressOnParent>7</epasd:AddressOnParent>
610     <epasd:AutomaticAllocation>true</epasd:AutomaticAllocation>
611     <epasd:Connection>Red Network</epasd:Connection>
612     <epasd:Description>Virtual Ethernet adapter</epasd:Description>

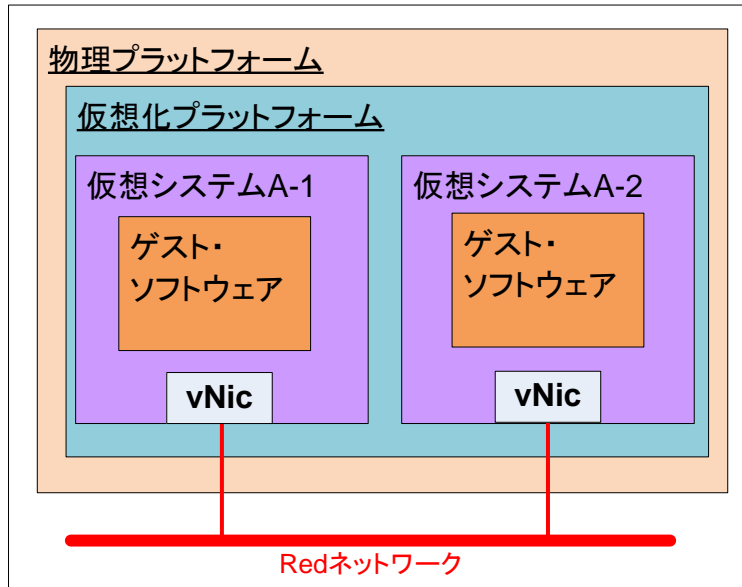
```

```

613 <epasd:ElementName>Virtual NIC 1</epasd:ElementName>
614 <epasd:InstanceID>8</epasd:InstanceID>
615 <epasd:ResourceType>10</epasd:ResourceType>
616 </EthernetPortItem>
    
```

617 ここでは複数のネットワークを定義できる。

618 図 5 は簡素化したネットワークを示している。このイーサネット・ポートは"Red"ネットワークに接続している。

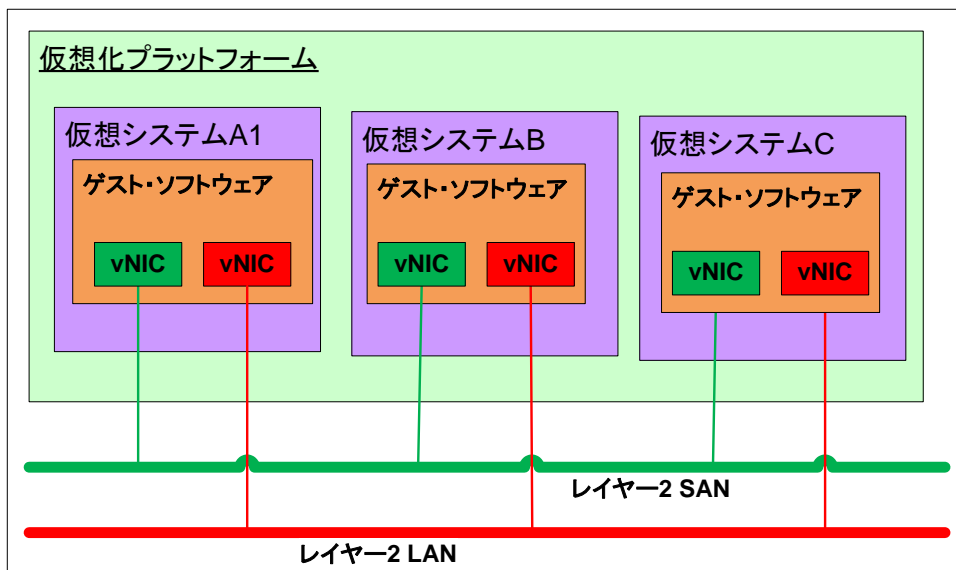


619

620

図 5 - ネットワーク接続

621 図 6 は、"Green"ネットワークはストレージに、"Red"ネットワークはローカル・エリア・ネットワークにそれぞれ接続
 622 するデュアル・ネットワーク構成である。



623

624

図 6 - LAN-SAN ネットワーク接続

625 4.4.5.2 ネットワーク・ポート・プロファイル

626 ネットワーク・ポート・プロファイル(DSP8049)は、OVF パッケージで定義した仮想システムで通信に使用する
627 ネットワークのネットワーク・ポートのプロパティ構成を定義する。詳細については、「仮想ネットワークの管理に関
628 するホワイトペーパー」(DSP2025)を参照。

629 EthernetPortAllocationSettingData クラスで指定可能なすべてのプロパティを以下に示す。各プロパ
630 ティに指定できる値は DSP0243、DSP1041、DSP1050、および CIM スキーマで定義する。ほとんどの OVF デ
631 スクリプタで使用するプロパティはこれらのプロパティの一部にすぎない。

```
632 <EthernetPortItem>
633   <epasd:Address>
634   <epasd:AddressOnParent>
635   <epasd:AllocationUnits>
636   <epasd:AllowedPriorities>
637   <epasd:AllowedToReceiveMACAddresses>
638   <epasd:AllowedToReceiveVLANs>
639   <epasd:AllowedToTransmitMACAddresses>
640   <epasd:AllowedToTransmitVLANs>
641   <epasd:AutomaticAllocation>
642   <epasd:AutomaticDeallocation>
643   <epasd:Caption>
644   <epasd:ChangeableType>
645   <epasd:ConfigurationName>
646   <epasd:Connection>
647   <epasd:ConsumerVisibility>
648   <epasd:DefaultPortVID>
649   <epasd:DefaultPriority>
650   <epasd:Description>
651   <epasd:DesiredVLANEndpointMode>
652   <epasd:ElementName>
653   <epasd:GroupID>
654   <epasd:HostResource>
655   <epasd:InstanceID>
656   <epasd:Limit>
657   <epasd:ManagerID>
658   <epasd:MappingBehavior>
659   <epasd:NetworkPortProfileID>
660   <epasd:NetworkPortProfileIDType>
661   <epasd:OtherEndpointMode>
662   <epasd:OtherNetworkPortProfileIDTypeInfo>
663   <epasd:OtherResourceType>
664   <epasd:Parent>
665   <epasd:PoolID>
666   <epasd:PortCorrelationID>
667   <epasd:PortVID>
```

```
668 <epasd:Promiscuous>
669 <epasd:ReceiveBandwidthLimit>
670 <epasd:ReceiveBandwidthReservation>
671 <epasd:Reservation>
672 <epasd:ResourceSubType>
673 <epasd:ResourceType>
674 <epasd:SourceMACFilteringEnabled>
675 <epasd:VSITypeID>
676 <epasd:VSITypeIDVersion>
677 <epasd:VirtualQuantity>
678 <epasd:VirtualQuantityUnits>
679 <epasd:Weight>
680 </EthernetPortItem>
```

681 4.4.6 DeploymentOptionsSection 要素

682 DeploymentOptionsSection 要素は、Envelope 要素の直接の子要素である。Configuration 要素は、
683 DeploymentOptionsSection 要素の直接の子要素である。DeploymentOptionsSection 要素は、1つ
684 以上の Configuration 要素を含む。

685 Configuration 要素を使用して、OVF パッケージのデプロイに使用するリソース構成を指定する。この構成は
686 デプロイ時に選択する。ユーザー入力を要求するには、userConfigurable 属性を使用する。

687 DeploymentOptionsSection 要素は、OVF パッケージで使用できる構成の ID、ラベル、および説明を記述
688 する。

689 デフォルト構成を示すには、default 属性を true に設定する。別途入力がない場合は、このデフォルト構成が
690 使用される。デフォルトが示されていない場合、先頭の構成がデフォルトとみなされる。

691 各構成には、その構成を特定する一意の ID 属性がある。この ID 属性の値は、VirtualHardwareSection
692 要素、ProductSection 要素、ScaleOutSection 要素など、その他の Section 要素の configuration
693 属性で指定する。

694 ある要素を複数の構成で使用する場合、この要素の configuration 属性には、それらの構成のすべての ID
695 をスペースで区切って記述する。

696 デプロイ機能では、ユーザーによる入力またはその他のメタデータのいずれかから構成を選択する必要がある。
697 構成を選択すると、この構成に設定されている configuration 属性を持つ要素がデプロイに使用される。
698 configuration 属性を持たない要素もデプロイされるが、構成の属性が異なる要素はデプロイされない。した
699 がって、デプロイ中に構成を 1 つ選択しただけで、OVF パッケージのさまざまなセクションにあるさまざまなアイテ
700 ムの構成に影響が及ぶ可能性がある。

701 以下の OVF デスクリプタの DeploymentOptionsSection 要素のスニペットは、Configuration 要素の
702 使用例を示している。メモリ・リソース割り当てのデフォルト構成を青色、"big"構成を灰色で示す。利用者による入
703 力が"big"構成を選択しない限り、このデプロイ機能ではデフォルト構成が選択される。

704

```
705 <DeploymentOptionsSection>
```



```

706     <Configuration ovf:id="big">
707         <Label>Big</Label>
708         <Description>Apply reservations for Memory</Description>
709     </Configuration>
710     ...
711 </DeploymentOptionsSection>
712
713 <VirtualHardwareSection>
714     <Info>...</Info>
715
716     <Item>
717         <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
718         <rasd:ElementName>512 MB memory size-no reservation</rasd:ElementName>
719         <rasd:InstanceID>0</rasd:InstanceID>
720         <rasd:ResourceType>4</rasd:ResourceType>
721         <rasd:VirtualQuantity>512</rasd:VirtualQuantity>
722     </Item>
723
724     <Item ovf:configuration="big">
725         <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
726         <rasd:ElementName>512 MB memory size &256 MB reservation</rasd:ElementName>
727         <rasd:InstanceID>0</rasd:InstanceID>
728         <rasd:Reservation>256</rasd:Reservation>
729     </Item>
730 </VirtualHardwareSection>
731

```

732 その結果、リソース割り当ての CIM インスタンスは以下のようになる。

```

733 <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
734 <rasd:ElementName>512 MB memory size & 256 MB reservation </rasd:ElementName>
735 <rasd:InstanceID>0</rasd:InstanceID>
736 <rasd:Reservation>256</rasd:Reservation>
737 <rasd:ResourceType>4</rasd:ResourceType>
738 <rasd:VirtualQuantity>512</rasd:VirtualQuantity>

```

739 灰色で示している部分は"big"構成のもの、青で示している部分はデフォルト構成のものである。

740 以下の例は、3 個の Configuration 要素を持つ DeploymentOptionsSection 要素である。デフォルトと
741 して指定されている構成は"normal"なので、構成を選択していない場合、デプロイでは configuration 属性
742 が"normal"の要素が使用される。

```

743 <DeploymentOptionSection>
744     <Configuration ovf:id="minimal">
745         <Label>Minimal</Label>
746         <Description>Smallest practical implementation</Description>
747     </Configuration>

```

```
748 <Configuration ovf:id="normal" ovf:default="true">
749   <Label>Normal</Label>
750   <Description>A typical implementation</Description>
751 </Configuration>
752 <Configuration ovf:id="large">
753   <Label>Large</Label>
754   <Description>A scaled up implementation</Description>
755 </Configuration>
756 </DeploymentOptionSection>
```

757 以下は、configuration 属性を使用して、前述の 3 つの構成へのリソース割り当てを定義した例である。
758 VirtualHardwareSection 要素は CIM_ResourceAllocationDescriptor プロパティを使用して、
759 必要なメモリ・リソース構成を指定する。VirtualHardwareSection の Item 要素はそれぞれ、前述の
760 DeploymentOptionsSection 要素で定義された構成を参照する。

```
761 <VirtualHardwareSection>
762   <Info>...</Info>
763   <Item ovf:configuration="normal">
764     <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
765     <rasd:ElementName>512 MB memory size and 256 MB reservation</rasd:ElementName>
766     <rasd:InstanceID>0</rasd:InstanceID>
767     <rasd:Reservation>256</rasd:Reservation>
768     <rasd:ResourceType>4</rasd:ResourceType>
769     <rasd:VirtualQuantity>512</rasd:VirtualQuantity>
770   </Item>
771   <Item ovf:configuration="minimal">
772     <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
773     <rasd:ElementName>256 MB memory size and 128 MB reservation</rasd:ElementName>
774     <rasd:InstanceID>1</rasd:InstanceID>
775     <rasd:Reservation>128</rasd:Reservation>
776     <rasd:ResourceType>4</rasd:ResourceType>
777     <rasd:VirtualQuantity>256</rasd:VirtualQuantity>
778   </Item>
779   <Item ovf:configuration="large">
780     <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
781     <rasd:ElementName>1024 MB memory size and 512 MB reservation</rasd:ElementName>
782     <rasd:InstanceID>0</rasd:InstanceID>
783     <rasd:Reservation>512</rasd:Reservation>
784     <rasd:ResourceType>4</rasd:ResourceType>
785     <rasd:VirtualQuantity>1024</rasd:VirtualQuantity>
786   </Item>
787 </VirtualHardwareSection>
```

788 この例では、デプロイ中に選択した構成によってメモリ・サイズを制御する。構成を選択していない場合、メモリ・
789 サイズはデフォルトの"normal"構成になる。

790 以下に、ProductSection で configuration 属性を使用した例を示す。

```

791 <ProductSection>
792   <Property ovf:key="app_log" ovf:type="string" ovf:value="low"
793     ovf:configuration="normal">
794     <Label>Loglevel</Label>
795     <Description>Loglevel for the service</Description>
796     <Value ovf:value="none" ovf:configuration="minimal">
797     <Value ovf:value="high" ovf:configuration="large">
798   </Property>
799 </ProductSection>

```

800 この例では、選択した構成に応じて"app_log"の値が異なる。仮想ハードウェアと同様、構成を選択しない場合、
801 "app_log"プロパティの値はデフォルトの"normal"構成値である"low"になる。

802 4.4.7 SharedDiskSection 要素

803 SharedDiskSection 要素は、クラスター化したデータベースのニーズを満たすために、複数の仮想システム
804 が仮想ディスクを参照できるようにする。使用するファイル共有システム・テクノロジーは、プラットフォームに固有
805 のものである。

806 SharedDiskSection 要素は Envelop レベルでのみ有効である。

807 共有ディスクはそれぞれ、一意の OVF パッケージ ID を持つ。SharedDiskSection 要素は、読み書きアクセ
808 スとするか (FALSE)、読み取り専用とするか (TRUE) を示す Boolean ovf:readOnly 属性を追加する。

809 以下の例は、SharedDiskSection 要素の基本を表している。

```

810 <ovf:SharedDiskSection>
811   <Info>Describes the set of virtual disks shared between VMs</Info>
812   <ovf:SharedDisk ovf:diskId="datadisk" ovf:fileRef="data"
813     ovf:capacity="8589934592" ovf:populatedSize="3549324972"
814     ovf:format="http://www.vmware.com/interfaces/specifications/vmdk.html#sparse"/>
815   <ovf:SharedDisk ovf:diskId="transientdisk" ovf:capacity="536870912"/>
816 </ovf:SharedDiskSection>

```

817 以下に、共有ディスクの使用例を示す。オペレーティング・システム (system)、クラスター・ソフトウェア
818 (crs_home)、データベース (db_home) のインストールに使用するディスクは、外部の File を参照して知ること
819 ができる。この例の共有仮想ディスクを外部の File 参照から知ることができない。複数の仮想システムで適切
820 に共有できるように、デプロイ・エンジンによって共有ディスクが作成される。

```

821 <ovf:References>
822   <ovf:File ovf:id="system" ovf:href="system.img" ovf:compression="gzip"/>
823   <ovf:File ovf:id="crs_home" ovf:href="crs_home.img" ovf:compression="gzip"/>
824   <ovf:File ovf:id="db_home" ovf:href="db_home.img" ovf:compression="gzip"/>
825 </ovf:References>
826 <ovf:DiskSection>
827   <ovf:Info>Virtual Disks</ovf:Info>
828   <ovf:Disk ovf:diskId="system" ovf:fileRef="system" ovf:capacity="5368709120"
829   ovf:format="Raw disk image"/>
830   <ovf:Disk ovf:diskId="crs_home" ovf:fileRef="crs_home" ovf:capacity="2147483648"

```

```
831 ovf:format="Raw disk image"/>
832   <ovf:Disk ovf:diskId="db_home" ovf:fileRef="db_home" ovf:capacity="4294967296"
833 ovf:format="Raw disk image"/>
834 </ovf:DiskSection>
835 <ovf:SharedDiskSection>
836   <ovf:Info>Virtual Disks shared at runtime</ovf:Info>
837   <ovf:SharedDisk ovf:diskId="crs_asm" ovf:capacity="4294967296" ovf:format="Raw disk
838 image"/>
839   <ovf:SharedDisk ovf:diskId="db_asm" ovf:capacity="12884901888" ovf:format="Raw disk
840 image"/>
841 </ovf:SharedDiskSection>
842 .....
843 <ovf:VirtualSystemCollection ovf:id="rac_db_asm">
844   <ovf:Info>Sample Oracle RAC using ASM</ovf:Info>
845   .....
846   <ovf:ScaleOutSection ovf:id="rac_db">
847     <ovf:Info>RAC DB</ovf:Info>
848     <ovf:Description>Number of instances</ovf:Description>
849     <ovf:InstanceCount ovf:default="2" ovf:minimum="2"
850 ovf:maximum="4"</ovf:InstanceCount>
851   </ovf:ScaleOutSection>
852   .....
853   <ovf:VirtualSystem ovf:id="rac_db">
854     <ovf:Info>RAC DB Instance</ovf:Info>
855     .....
856     <ovf:VirtualHardwareSection>
857       <ovf:Info>System requirements: 8192 MB, 2 CPUs, 5 disks, 2 nics
858 </ovf:Info>
859       .....
860       <ovf:Item>
861         <rasd:Description>Disk 1</rasd:Description>
862         <rasd:ElementName>Disk 1</rasd:ElementName>
863         <rasd:HostResource>ovf:/disk/system</rasd:HostResource>
864         <rasd:ResourceType>17</rasd:ResourceType>
865       </ovf:Item>
866       <ovf:Item>
867         <rasd:Description>Disk 2</rasd:Description>
868         <rasd:ElementName>Disk 2</rasd:ElementName>
869         <rasd:HostResource>ovf:/disk/crs_home</rasd:HostResource>
870         <rasd:ResourceType>17</rasd:ResourceType>
871       </ovf:Item>
872       <ovf:Item>
873         <rasd:Description>Disk 3</rasd:Description>
874         <rasd:ElementName>Disk 3</rasd:ElementName>
875         <rasd:HostResource>ovf:/disk/db_home</rasd:HostResource>
876         <rasd:ResourceType>17</rasd:ResourceType>
```

```

877     </ovf:Item>
878     <ovf:Item>
879         <rasd:Description>Disk 4</rasd:Description>
880         <rasd:ElementName>Disk 4</rasd:ElementName>
881         <rasd:HostResource>ovf:/disk/crs_asm</rasd:HostResource>
882         <rasd:ResourceType>17</rasd:ResourceType>
883     </ovf:Item>
884     <ovf:Item>
885         <rasd:Description>Disk 5</rasd:Description>
886         <rasd:ElementName>Disk 5</rasd:ElementName>
887         <rasd:HostResource>ovf:/disk/db_asm</rasd:HostResource>
888         <rasd:ResourceType>17</rasd:ResourceType>
889     </ovf:Item>
890     .....
891 </ovf:VirtualHardwareSection>
892 </ovf:VirtualSystem>
893 </ovf:VirtualSystemCollection>

```

894 4.4.8 PlacementGroupSection 要素

895 PlacementGroupSection 要素は、配置グループとそれに関連する配置ポリシーの ID を定義する。この配置
 896 グループは、PlacementSection 要素によって VirtualSystemCollection または VirtualSystem と
 897 関連付けられる。

898 以下に、PlacementGroupSection 要素の例を示す。

```

899 <ovf:PlacementGroupSection ovf:id="PG2" ovf:policy="availability">
900     <Info>Placement policy for group of virtual systems that need availability</Info>
901     <ovf:Description>Placement policy for a database tier</ovf:Description>
902 </ovf:PlacementGroupSection>
903     ...
904 <ovf:PlacementGroupSection ovf:id="PG1" ovf:policy="affinity">
905     <Info>Placement policy for group of virtual systems that need affinity</Info>
906     <ovf:Description>Placement policy for a web tier</ovf:Description>
907 </ovf:PlacementGroupSection>

```

908 PlacementGroupSection 要素は、Envelope 要素の直接の子要素である。4.5.5 を参照。

909 4.5 仮想システムおよび仮想システムのコレクションで使用する OVF の Section 要素

910 以下の OVF デスクリプタ Section 要素は VirtualSystem または VirtualSystemCollection 要素で使
 911 用できる。

912 4.5.1 AnnotationSection 要素

913 AnnotationSection 要素はユーザー定義要素であり、VirtualSystem 要素および
 914 VirtualSystemCollection 要素で使用できる。AnnotationSection 要素には Annotation 要素が 1
 915 つ存在する。Annotation 要素はローカライズ可能である。たとえば、パッケージをデプロイするときに

916 Annotation 要素を利用者に表示できる。なお、OVF で指定されている Annotation 要素は XML スキーマ
917 の Annotation 要素ではない。

```
918 <AnnotationSection>  
919     <Info>An annotation on this service. It can be ignored</Info>  
920     <Annotation>Contact customer support if you have any problems</Annotation>  
921 </AnnotationSection >
```

922 4.5.2 ProductSection 要素

923 ProductSection 要素は、アプライアンスの名前やベンダー、カスタマイズに使用できるプロパティ・セットなど
924 の製品情報を提供する。このようなプロパティは、アプライアンスのインストール時に構成される。そのためには
925 ユーザー入力を必要とすることが普通である。この詳細については、後述する。

```
926 <ProductSection ovf:class="com.mycrm.myService" ovf:instance="1">  
927     <Info>Describes product information for the service</Info>  
928     <Product>MyCRM Enterprise</Product>  
929     <Vendor>MyCRM Corporation</Vendor>  
930     <Version>4.5</Version>  
931     <FullVersion>4.5-b4523</FullVersion>  
932     <ProductUrl>http://www.mycrm.com/enterprise</ProductUrl>  
933     <VendorUrl>http://www.mycrm.com</VendorUrl>  
934     <Icon ovf:height="32" ovf:width="32" ovf:mimeType="image/png" ovf:fileRef="icon">  
935     <Category>Email properties</Category>  
936     <Property ovf:key="adminEmail" ovf:type="string" ovf:userConfigurable="true">  
937         <Label>Admin email</Label>  
938         <Description>Email address of administrator</Description>  
939     </Property>  
940     <Category>Admin properties</Category>  
941     <Property ovf:key="appLog" ovf:type="string" ovf:value="low"  
942     ovf:userConfigurable="true">  
943         <Description>Loglevel for the service</Description>  
944     </Property>  
945     <Property ovf:key="appisSecondary" ovf:value="false" ovf:type="boolean">  
946         <Description>Cluster setup for application server</Description>  
947     </Property>  
948     <Property ovf:key="appIp" ovf:type="string" ovf:value="{appserver-vm}">  
949         <Description>IP address of the application server VM</Description>  
950     </Property>  
951 </ProductSection>
```

952 ovf:key 属性にはピリオド('.')とコロン(':')を使用できず、ovf:class 属性および ovf:instance 属性には
953 コロン(':')を使用できない。

954 インストールされている製品のインスタンスが 1 つのみである場合、ovf:instance 属性は使用しない。

955 以下は、ProductSection 要素で OVF の Properties を使用する例を示している。

```
956 <ProductSection>
957   <Property ovf:key="adminEmail" ovf:type="string" ovf:userConfigurable="true"
958     ovf:configuration="standard">
959     <Label>Admin email</Label>
960     <Description>Email address of service administrator</Description>
961   </Property>
962   <Property ovf:key="appLog" ovf:type="string" ovf:value="low"
963     ovf:userConfigurable="true">
964     <Label>Loglevel</Label>
965     <Description>Loglevel for the service</Description>
966     <Value ovf:value="none" ovf:configuration="minimal">
967   </Property>
968 </ProductSection>
```

969 上の例では、adminEmail プロパティは標準構成でのみユーザー構成可能であり、appLog プロパティのデフォルト値は最小構成で"low"から"none"に変更される。

971 4.5.3 EulaSection 要素

972 EulaSection には、その親要素のライセンス契約を人間が読める形式で記述する。この親要素とは、
973 VirtualSystem 要素または VirtualSystemCollection 要素であることが普通である。これらの親要素
974 はそれぞれ、複数の EulaSection 要素を持つことができる。各 EulaSection 要素にある License 要素の
975 内容は、OVF パッケージのデプロイ時にその内容への同意を求めるためにユーザーに表示される。無人デプロイ
976 をサポートする場合は、暗黙的に Eula に同意するためのプロビジョニングを用意する。

977 Eula はローカライズを目的として外部化できるほか、外部ライセンス文書を参照先として指定することもできる。
978 国際化の詳細については、5.2 を参照。

979 以下に、EulaSection 要素の例を示す。

```
980 <EulaSection>
981   <Info>Licensing agreement</Info>
982   <License>
983   Lorem ipsum dolor sit amet, ligula suspendisse nulla pretium, rhoncus tempor placerat
984   fermentum, enim integer ad vestibulum volutpat. Nisl rhoncus turpis est, vel elit, congue
985   wisi enim nunc ultricies sit, magna tincidunt. Maecenas aliquam maecenas ligula nostra,
986   accumsan taciti. Sociis mauris in integer, a dolor netus non dui aliquet, sagittis felis
987   sodales, dolor sociis mauris, vel eu libero cras. Interdum at. Eget habitasse elementum
988   est, ipsum purus pede porttitor class, ut adipiscing, aliquet sed auctor, imperdiet arcu
989   per diam dapibus libero dui. Enim eros in vel, volutpat nec pellentesque leo, scelerisque.
990   </License>
991 </EulaSection>
```

992 4.5.4 VirtualHardwareSection 要素

993 VirtualHardwareSection 要素は、CIM リソース割り当て設定データ・モデルを使用する仮想ハードウェア
994 を記述する。このモデルは、要求されているリソースのタイプと数量を CIM プロパティの指定によって記述する
995 CIM_ResourceAllocationSettingData クラスを基にしている。CIM スキーマについては
996 <http://www.dmtf.org/standards/cim> に説明がある。

997 以下に、仮想プロセッサ・デバイスで必要最低限のリソース割り当て設定データを示す。これはわかりにくいので、`rasd:Description` と `rasd:ElementName` を追加すると効果的である。

```
999 <Item>
1000   <rasd:InstanceID>0</epasd:InstanceID>
1001   <rasd:ResourceType>3</epasd:ResourceType>
1002 </Item>
```

1003 この `VirtualHardwareSection` 要素は、仮想システムで使用するハードウェア抽象化レイヤーの仮想デバイス
1004 を記述する。`CIM_ResourceAllocationSettingData` にはデバイスのリストを置く。デバイスによっては、イーサネット・ポートやストレージのように、プロパティの拡張セットを使用してサブクラス化されるものもある。

1006 この例では、ごく一般的なハードウェア・セット(ゲスト・メモリー500 MB、CPU 1 個、NIC 1 個、仮想ディスク 1 個)
1007 を指定している。ここでは、上位のセクションにあるネットワーク ID とディスク ID を参照している。ハードウェア・セ
1008 クションが不完全である場合や存在しない場合は、デプロイが失敗する原因となる可能性がある。

1009 以下は `VirtualHardwareSection` 要素の例である。

```
1010 <VirtualHardwareSection>
1011   <Info>Memory = 4 GB, CPU = 1 GHz, Disk = 100 GB, 1 Ethernet nic</Info>
1012   <Item>
1013     <rasd:AllocationUnits>Hertz*10^9</rasd:AllocationUnits>
1014     <rasd:Description>Virtual CPU</rasd:Description>
1015     <rasd:ElementName>1 GHz virtual CPU</rasd:ElementName>
1016     <rasd:InstanceID>1</rasd:InstanceID>
1017     <rasd:Reservation>1</rasd:Reservation>
1018     <rasd:ResourceType>3</rasd:ResourceType>
1019     <rasd:VirtualQuantity>1</rasd:VirtualQuantity>
1020     <rasd:VirtualQuantityUnit>Count</ rasd:VirtualQuantityUnit>
1021   </Item>
1022   <Item>
1023     <rasd:AllocationUnits>byte*2^30</rasd:AllocationUnits>
1024     <rasd:Description>Memory</rasd:Description>
1025     <rasd:ElementName>1 GByte of memory</rasd:ElementName>
1026     <rasd:InstanceID>2</rasd:InstanceID>
1027     <rasd:Limit>4</rasd:Limit>
1028     <rasd:Reservation>4</rasd:Reservation>
1029     <rasd:ResourceType>4</rasd:ResourceType>
1030   </Item>
1031   <EthernetPortItem>
1032     <epasd:AllocationUnits>bit / second *2^30 </rasd:AllocationUnits>
1033     <epasd:Connection>VM Network</epasd:Connection>
1034     <epasd:Description>Virtual NIC</epasd:Description>
1035     <epasd:ElementName>Ethernet Port</epasd:ElementName>
1036     <epasd:NetworkPortProfileID>1</epasd:NetworkPortProfileID>
1037     <epasd:NetworkPortProfileIDType>4</epasd:NetworkPortProfileIDType>
1038     <epasd:ResourceType>10</epasd:ResourceType>
```



```

1039 <epasd:VirtualQuantity>1</epasd:VirtualQuantity>
1040 <epasd:VirtualQuantityUnits>Count</epasd:VirtualQuantityUnits>
1041 </EthernetPortItem>
1042 <StorageItem>
1043 <sasd:AllocationUnits>byte*2^30</sasd:AllocationUnits>
1044 <sasd:Description>Virtual Disk</sasd:Description>
1045 <sasd:ElementName>100 GByte Virtual Disk</sasd:ElementName>
1046
1047 <sasd:Reservation>100</sasd:Reservation>
1048 <sasd:ResourceType>31</sasd:ResourceType>
1049 <sasd:VirtualQuantity>1</sasd:VirtualQuantity>
1050 <sasd:VirtualQuantityUnit>Count</sasd:VirtualQuantityUnit>
1051 </StorageItem>
1052 </VirtualHardwareSection>

```

1053 ResourceSubType CIM プロパティの例を以下に示す。

```

1054 <rasd:ResourceSubType>buslogic lsilogic</rasd:ResourceSubType>

```

1055 以下は、デフォルト構成と'big'構成を持つ VirtualHardwareSection 要素の例である。構成オプションの使
 1056 い方については、4.4.6 を参照。

```

1057 <VirtualHardwareSection>
1058 <Info>...</Info>
1059 <Item>
1060 <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
1061 <rasd:ElementName>512 MB memory size and 256 MB reservation</rasd:ElementName>
1062 <rasd:InstanceID>0</rasd:InstanceID>
1063 <rasd:Reservation>256</rasd:Reservation>
1064 <rasd:ResourceType>4</rasd:ResourceType>
1065 <rasd:VirtualQuantity>512</rasd:VirtualQuantity>
1066 </Item>
1067 ...
1068 <Item ovf:configuration="big">
1069 <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
1070 <rasd:ElementName>1024 MB memory size and 512 MB reservation</rasd:ElementName>
1071 <rasd:InstanceID>0</rasd:InstanceID>
1072 <rasd:Reservation>512</rasd:Reservation>
1073 <rasd:ResourceType>4</rasd:ResourceType>
1074 <rasd:VirtualQuantity>1024</rasd:VirtualQuantity>
1075 </Item>
1076 </VirtualHardwareSection>

```

1077 4.5.5 PlacementSection 要素

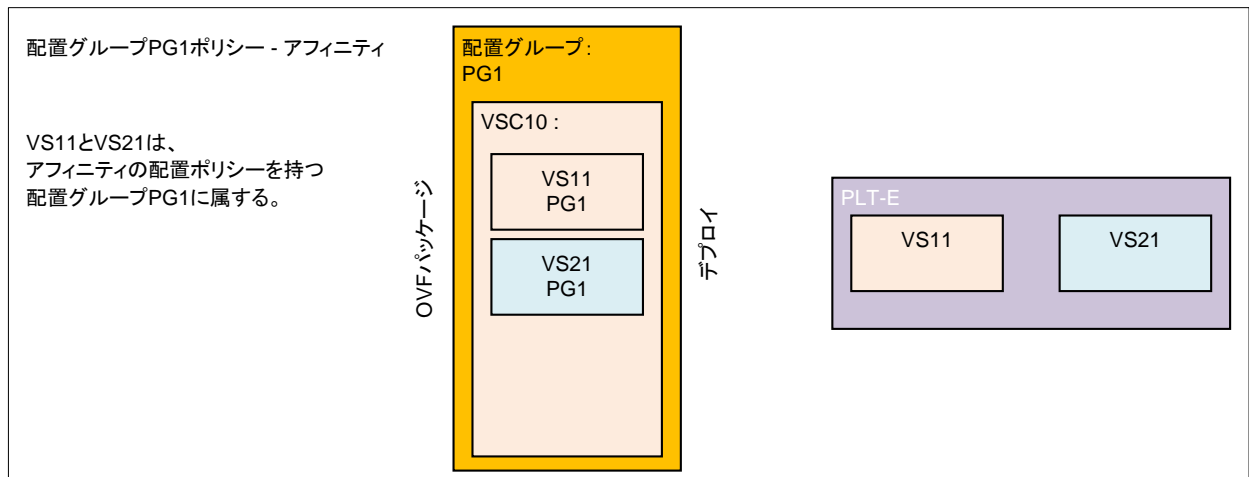
1078 PlacementSection 要素は、仮想システムまたは仮想システムのコレクションが属する配置グループを指定す
 1079 る。この配置グループで指定した配置ポリシー(4.4.8を参照)は、デプロイ機能により適用される。以下の OVF デ

1080 スクリプタのスニペットは、2つの仮想システムそれぞれの配置セクションを示している。

```

1081 <VirtualSystemCollection ovf:id="VSC10">
1082   <VirtualSystem ovf:id="VS11">
1083     <Info>Web server</Info>
1084     ...
1085     <ovf:PlacementSection ovf:group="PG1">
1086       <Info>Placement policy group reference</Info>
1087     </ovf:PlacementSection>
1088     ...
1089   </VirtualSystem>
1090   <VirtualSystem ovf:id="VS21">
1091     <Info>Web server</Info>
1092     ...
1093     <ovf:PlacementSection ovf:group="PG1">
1094       <Info>Placement policy group reference</Info>
1095     </ovf:PlacementSection>
1096     ...
1097   </VirtualSystem>
1098 </VirtualSystemCollection>
    
```

1099 この例では、仮想システムはインスタンス化されたときに、"PG1"配置グループで指定された配置ポリシーに従っ
 1100 て配置される。4.4.8に示すとおり、配置グループ'PG1'は、アフィニティ配置ポリシーを持つ。図7にアフィニティ配
 1101 置を示す。



1102

1103

図7 - アフィニティ配置

1104 以下の OVF デスクリプタのスニペットは、2つの仮想システムそれぞれの配置セクションを示している。

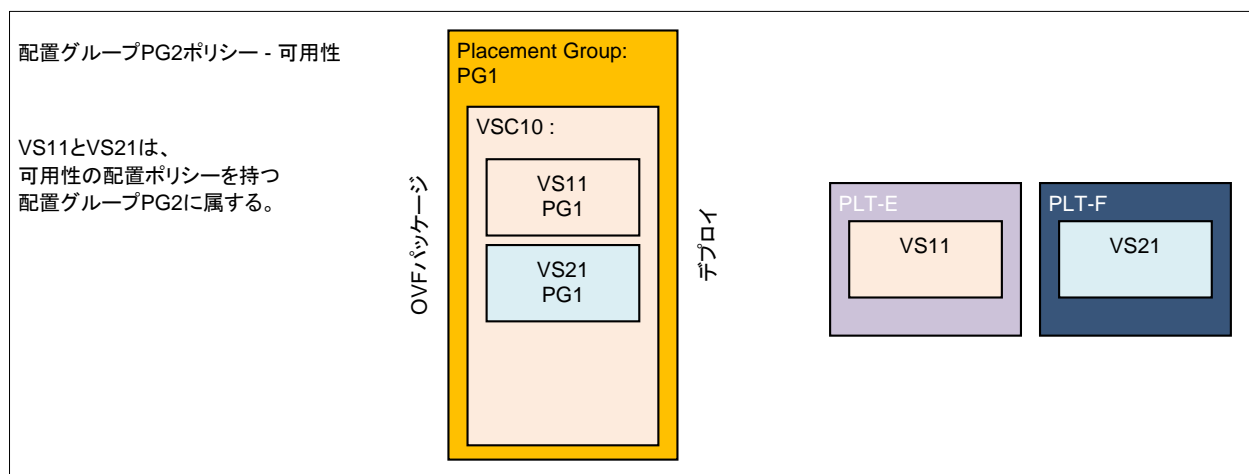
```

1105 <VirtualSystemCollection ovf:id="VSC10">
1106   <VirtualSystem ovf:id="VS11">
1107     <Info>Web server</Info>
1108     ...
1109     <ovf:PlacementSection ovf:group="PG2">
    
```

```

1110     <Info>Placement policy group reference</Info>
1111   </ovf:PlacementSection>
1112   ...
1113 </VirtualSystem>
1114 <VirtualSystem ovf:id="VS21">
1115   <Info>Web server</Info>
1116   ...
1117   <ovf:PlacementSection ovf:group="PG2">
1118     <Info>Placement policy group reference</Info>
1119   </ovf:PlacementSection>
1120   ...
1121 </VirtualSystem>
1122 </VirtualSystemCollection>
    
```

1123 この例では、仮想システムはインスタンス化されたときに、「PG2」配置グループで指定された配置ポリシーに従っ
 1124 て配置される。4.4.8 に示すとおり、配置グループ「PG2」は、可用性配置ポリシーを持つ。図 8 に可用性配置を示
 1125 す。



1126

図 8 - 可用性配置

1127

1128 以下の OVF デスクリプタのスニペットは、2つの仮想システムそれぞれの配置セクションを示している。

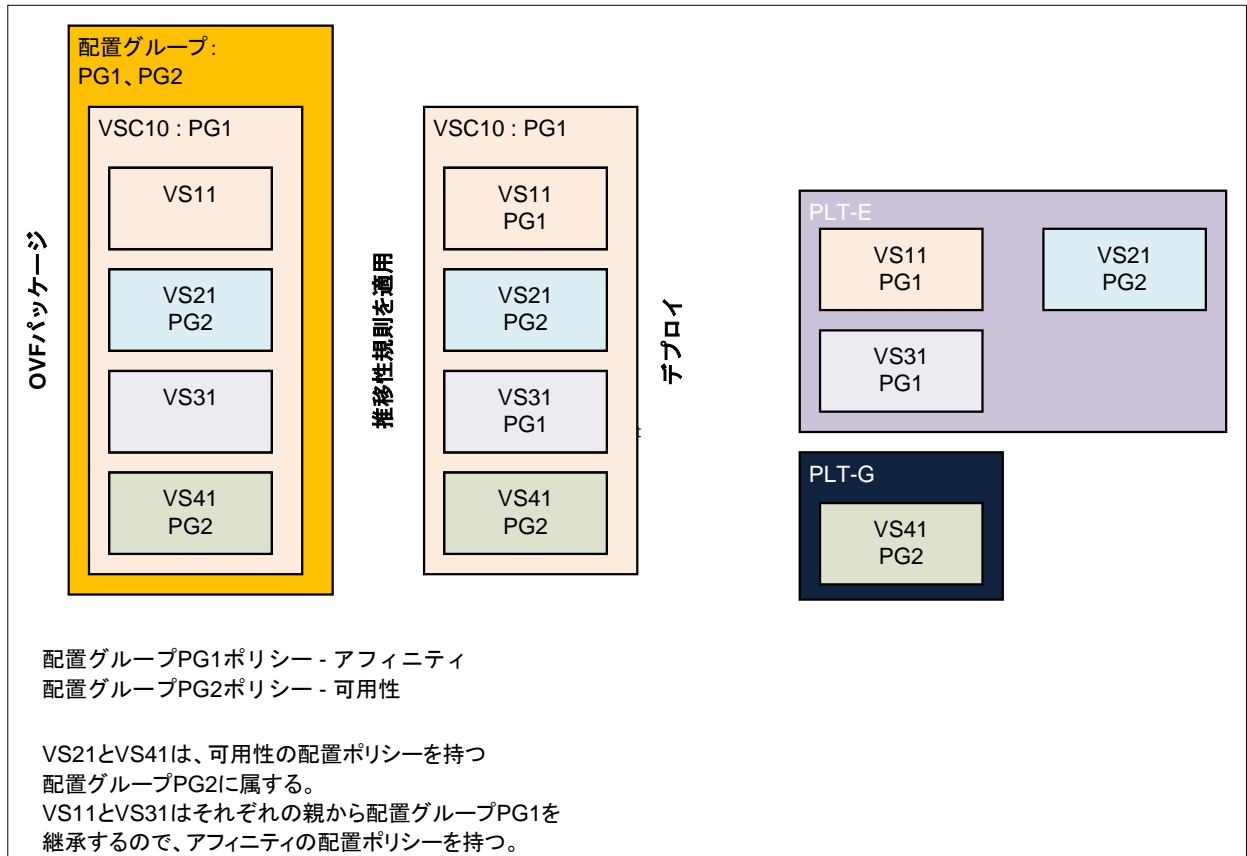
```

1129 <VirtualSystemCollection ovf:id="VSC10">
1130   <ovf:PlacementSection ovf:group="PG1">
1131     <Info>Placement policy group reference</Info>
1132   </ovf:PlacementSection>
1133
1134   <VirtualSystem ovf:id="VS11">
1135     <Info>Web server</Info>
1136     ...
1137   </VirtualSystem>
1138   <VirtualSystem ovf:id="VS21">
1139     <Info>Web server</Info>
1140     ...
    
```

```
1141 <ovf:PlacementSection ovf:group="PG2">
1142   <Info>Placement policy group reference</Info>
1143 </ovf:PlacementSection>
1144 ...
1145 </VirtualSystem>
1146 <VirtualSystem ovf:id="VS31">
1147   <Info>Web server</Info>
1148   ...
1149 </VirtualSystem>
1150 <VirtualSystem ovf:id="VS41">
1151   <Info>Web server</Info>
1152   ...
1153 <ovf:PlacementSection ovf:group="PG2">
1154   <Info>Placement policy group reference</Info>
1155 </ovf:PlacementSection>
1156 ...
1157 </VirtualSystem>
1158 </VirtualSystemCollection>
```

1159 この例では、仮想システムはインスタンス化されたときに、"PG1"配置グループおよび"PG2"配置グループで指定
1160 された配置ポリシーに従って配置される。4.4.8 に示すとおり、配置グループ'PG1'は、アフィニティ配置ポリシーを
1161 持つ。4.4.8 に示すとおり、配置グループ'PG2'は、可用性配置ポリシーを持つ。図 9 に、この複雑な例での仮想シ
1162 ステムの配置を示す。これは推移律を応用した例でもある。

1163



1164

1165

図 9 - アフィニティおよび可用性配置

1166 **4.5.6 EncryptionSection 要素**

1167 目的の利用者のみが OVF アプライアンスを使用できるようにして、その範囲でアプライアンスの交換を可能にする
 1168 暗号化スキーマの利用が望ましいことにはいくつかの理由がある。この仕様で提案されている暗号化スキーマ
 1169 では、既存の暗号化標準を利用してこの機能を取り入れている。

1170 EncryptionSection 要素は、暗号化アルゴリズムに関連するマークアップの配置先および暗号化されている
 1171 OVF コンテンツを参照するリファレンス・リストの配置先とする単一の場所を提供する。

1172 文書では、単一のキーによる単一の暗号化方法を使用することが普通である。一方で、この仕様は、OVF デスク
 1173 リプタのさまざまな部分を、さまざまな方法で導かれてさまざまな方法でエンド・ユーザーに伝達される多彩な
 1174 キーで暗号化する柔軟性も用意している。

1175 暗号化した OVF デスクリプタの部分によっては、それを復号化しないと、OVF デスクリプタと OVF スキーマが照
 1176 合されない可能性がある点に注意することが重要である。

1177 この暗号化では、XML 暗号化標準 1.1 を使用して、リファレンス・セクションにあるファイルまたは OVF 文書にある
 1178 XML マークアップの任意の部分を暗号化する。

1179 暗号化の観点から見ると、この標準で定義されている重要な特徴は以下のとおりである。

- 1180 a) 暗号化に使用するキーを導くために使用するアルゴリズム
- 1181 b) そのキーを使用してコンテンツを暗号化するブロック暗号化アルゴリズム

1182 c) OVF の XML 文書に埋め込まれているキーをトランスポートする方法

1183 OVF パッケージの作成者は、文書で使用する暗号化方法ごとに、これらの必要な点をすべて定義する必要がある。
1184 たとえば、使用するキーを文書に埋め込むことができるほか、別の方法で目的のエンド・ユーザーにキーを伝
1185 達することもできる。

1186 もう一つの特徴は、OVF エンベロープのマークアップ・セクションに対する参照のリスト、または指定の方法を使用
1187 して暗号化したリファレンス・セクションに置いたファイルである。OVF デスクリプタ内の任意のセクションを暗号化
1188 できるようにするため、use は ReferenceList 要素内の XML ID 属性から構成される。

1189 以下の例は、Encryption セクションの概念構造を表している。

```
1190 <!-- Start of encryption section --!>
1191 <!-- Start of Markup for encryption method 1 --!>
1192 <!-- Markup defining key derivation aspects per XML encryption 1.1 --!>
1193 <!-- Markup defining the usage of the key for encryption per XML encryption 1.1 --!>
1194 <!-- Optionally, the markup for key transportation per XML encryption 1.1 --!>
1195 <!-- Start of markup for pointers to the list of XML fragments encrypted using method
1196 1--!>
1197 <!-- Pointer 1 --!>
1198 .
1199 .
1200 <!-- Pointer N --!>
1201 <!-- End of markup for pointers to the list of XML fragments encrypted using method
1202 1 --!>
1203 <!-- End of Markup for method 1 of encryption --!>
1204
1205 <!-- Start of the markup for encryption method N --!>
1206 <!-- Markup defining key derivation aspects per XML encryption 1.1 --!>
1207 <!-- Markup defining the usage of the key for encryption per XML encryption 1.1
1208 --!>
1209 <!-- Optionally, the markup for key transportation per XML encryption 1.1 --!>
1210 <!-- Start of markup for pointers to the list of XML fragments encrypted using
1211 method 1--!>
1212 <!-- Pointer 1 --!>
1213 .
1214 .
1215 <!-- Pointer N --!>
1216 <!-- End of Markup for encryption method N --!>
1217 <!-- End of encryption section --!>
```

1218 以下は、OVF 暗号化セクションの例である。ここでは、OVF 文書で利用している暗号化方法と、暗号化したアイテム
1219 を参照するリファレンス・リストを記述している。

```
1220 <ovf:EncryptionSection>
1221 <!-- This section contains two different methods of encryption and the corresponding
1222 backpointers to the data that is encrypted. ->
1223 <!-- Method#1: Pass phrase based key derivation ->
```

```
1224 <!-- The following derived key block defines PBKDF2 and the corresponding backpointers
1225 to the encrypted data elements. -->
1226   <!-- Use a salt value "ovfpassword" and iteration count of 4096. --->
1227   <xenc11:DerivedKey>
1228     <xenc11:KeyDerivationMethod
1229 Algorithm="http://www.rsasecurity.com/rsalabs/pkcs/schemas/pkcs-5#pbkdf2"/>
1230 <pkcs-5:PBKDF2-params>
1231   <Salt>
1232     <Specified>ovfpassword</Specified>
1233   </Salt>
1234   <IterationCount>4096</IterationCount>
1235   <KeyLength>16</KeyLength>
1236   <PRF Algorithm="http://www.w3.org/2001/04/xmldsig-more#hmac-sha256"/>
1237 </pkcs-5:PBKDF2-params>
1238 ...
1239 <!-- The ReferenceList element below contains references to the file Ref-109.vhd via the
1240 URI syntax that is specified by XML Encryption.
1241 --->
1242 <xenc:ReferenceList>
1243   <xenc:DataReference URI="#first.vhd" />
1244 <xenc:DataReference URI=... />
1245 <xenc:DataReference URI=... />
1246 </xenc:ReferenceList>
1247 </xenc11:DerivedKey>
1248   <!-- Method#2: The following example illustrates use of a symmetric key transported
1249 by using the public key within a certificate. ->
1250 <xenc:EncryptedKey>
1251   <xenc:EncryptionMethod
1252 Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
1253   <ds:KeyInfo xmlns:ds='http://www.w3.org/2000/09/xmldsig#'
1254     <ds:X509Data>
1255       <ds:X509Certificate> ... </ds:X509Certificate>
1256     </ds:X509Data>
1257   </ds:KeyInfo>
1258   <xenc:CipherData>
1259     <xenc:CipherValue> ... </xenc:CipherValue>
1260   </xenc:CipherData>
1261 <!-- The ReferenceList element below contains reference #second-xml-fragment" to the XML
1262 fragment that has been encrypted by using the above method. --->
1263   <xenc:ReferenceList>
1264     <xenc:DataReference URI='#second-xml-fragment' />
1265     <xenc:DataReference URI='...' />
1266     <xenc:DataReference URI='...' />
1267   </xenc:ReferenceList>
1268 </xenc:EncryptedKey>
1269 </ovf:EncryptionSection>
```

1270 Below is an example of the encrypted file that is referenced in the EncryptionSection above
1271 by using URI='Ref-109.vhd' syntax.

1272 EXAMPLE:

1273 <ovf:References>

1274 <ovf:File

1275 ovf:id="Xen:9cb10691-4012-4aeb-970c-3d47a906bfff/0b13bdba-3761-8622-22fc-2e252ed9ce14"

1276 ovf:href="Ref-109.vhd">

1277 <!-- The encrypted file referenced by the package is enclosed by an EncryptedData with a
1278 CipherReference to the actual encrypted file. The EncryptionSection in this example has
1279 a backpointer to it under the PBKDF2 algorithm via Id="first.vhd". This tells the decrypter
1280 how to decrypt the file. -->

1281 <xenc:EncryptedData Id="first.vhd" Type='http://www.w3.org/2001/04/xmlenc#Element' >

1282 <xenc:EncryptionMethod

1283 Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc" />

1284 <xenc:CipherData>

1285 <xenc:CipherReference URI='Ref-109.vhd' />

1286 </xenc:CipherData>

1287 </xenc:EncryptedData>

1288 </ovf:File>

1289 </ovf:References>

1290 Below is an example of the encrypted OVF markup that is referenced in the EncryptionSection
1291 above by using URI='#second-xml-fragment' syntax.

1292 EXAMPLE:

1293 <!-- The EncryptedData element below encompasses encrypted xml from the original document.
1294 It is provided with the Id "second-xml-fragment", which allows it to be referenced from
1295 the EncryptionSection. -->

1296 <xenc:EncryptedData Type=http://www.w3.org/2001/04/xmlenc#Element

1297 Id="second-xml-fragment">

1298 <!-- Each EncryptedData specifies its own encryption method. -->

1299 <xenc:EncryptionMethod Algorithm=http://www.w3.org/2001/04/xmlenc#aes128-cbc/>

1300 <xenc:CipherData>

1301 <!-- Encrypted content --->

1302 <xenc:CipherValue>DEADBEEF</xenc:CipherValue>

1303 </xenc:CipherData>

1304 </xenc:EncryptedData>

1305

1306 4.6 仮想システム・コレクションで使用する OVF Section 要素

1307 4.6.1 ResourceAllocationSection 要素

1308 ResourceAllocationSection 要素は、仮想システムのコレクションに適用するリソース制約を設定する。
1309 対照的に、VirtualHardwareSection 要素は特定の仮想システムに適用される。

1310 ResourceAllocationSection 要素は bound 属性を使用して、仮想システムのコレクションにあるすべての
1311 仮想システムに適用するリソース割り当ての最小値、最大値、またはその両方を設定する。

1312 以下に、ResourceAllocationSection 要素の例を示す。このプロセッサの割り当ては、bound 属性の

1313 使用方法を説明している。

```
1314 <ResourceAllocationSection>
1315   <Info>Defines reservations for CPU and memory for the collection of VMs</Info>
1316   <Item>
1317     <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
1318     <rasd:ElementName>300 MB reservation</rasd:ElementName>
1319     <rasd:InstanceID>0</rasd:InstanceID>
1320     <rasd:Reservation>300</rasd:Reservation>
1321     <rasd:ResourceType>4</rasd:ResourceType>
1322   </Item>
1323   <Item ovf:bound="min">
1324     <rasd:AllocationUnits>MHz</rasd:AllocationUnits>
1325     <rasd:ElementName>500 MHz reservation</rasd:ElementName>
1326     <rasd:InstanceID>1</rasd:InstanceID>
1327     <rasd:Reservation>500</rasd:Reservation>
1328     <rasd:ResourceType>3</rasd:ResourceType>
1329   </Item>
1330   <Item ovf:bound="max">
1331     <rasd:AllocationUnits>MHz</rasd:AllocationUnits>
1332     <rasd:ElementName>1500 MHz reservation</rasd:ElementName>
1333     <rasd:InstanceID>1</rasd:InstanceID>
1334     <rasd:Reservation>1500</rasd:Reservation>
1335     <rasd:ResourceType>3</rasd:ResourceType>
1336   </Item>
1337 </ResourceAllocationSection>
```

1338 4.6.2 StartupSection 要素

1339 StartupSection 要素は、仮想システムのコレクションに対して稼働のオンとオフを制御するもので、
1340 InstallSection 要素の後で実行される。StartupSection 要素は Item 要素のリストである。Item 要素
1341 には、稼働のオンとオフの順序と時期を制御する属性がある。StartupSection 要素の Item 要素は、
1342 StartupSection 要素をスコープとする。これらの要素を VirtualHardwareSection 要素や
1343 ResourceAllocationSection 要素の Item 要素と混同してはならない。

1344 StartupSection 要素内の Item 要素は、VirtualSystem 要素または VirtualSystemCollection
1345 要素を参照する。StartupSection 要素は、仮想システムのコレクションに属する仮想システムとその仮想シ
1346 ステムのコレクションの子である仮想システムのコレクションの両方の稼働オン/オフを制御できる。このフォーマッ
1347 トでは、再帰的な起動構成が可能である。図 10 を参照。

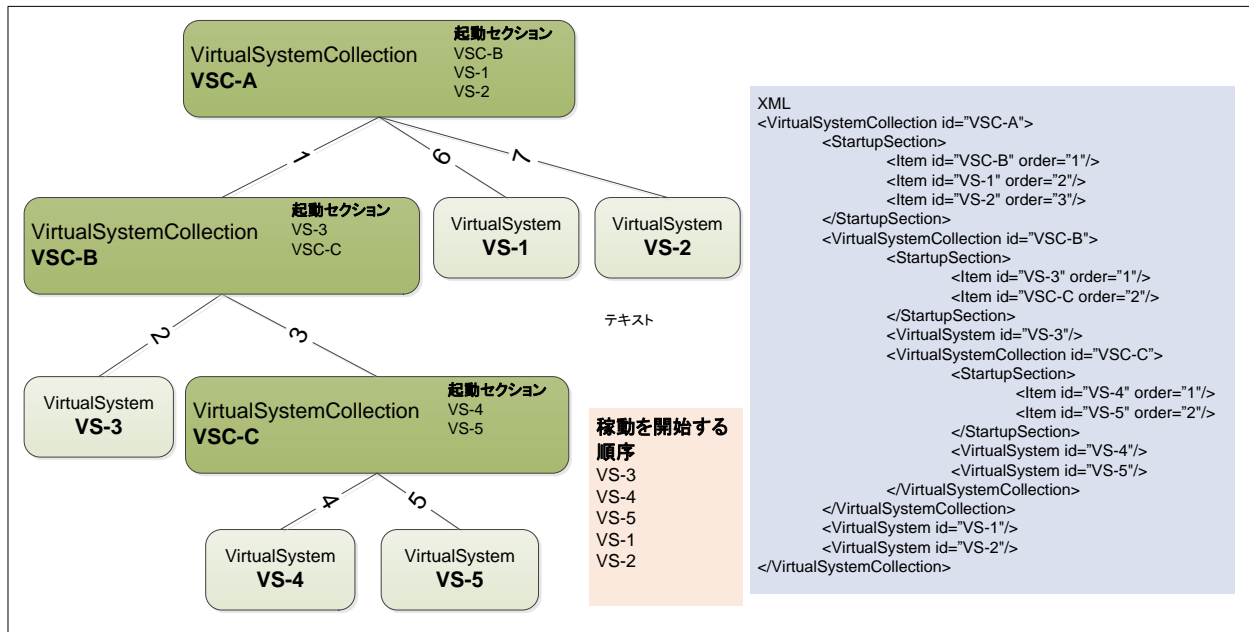


図 10 - StartupSection トラバース

1348

1349

1350 起動の順序は、VirtualSystemCollection 要素にある StartupSection 要素の中で、Item 要素の
1351 order 属性に指定された値で決まる。order 属性はゼロ(0)または正の整数である。

1352 Item 要素の order 属性の値が'0'(zero)の場合、仮想システムまたは仮想システムのコレクションはいつでも
1353 稼働を開始できる。仮想化プラットフォームは、order の値が大きいアイテムの稼働開始を待つ必要はない。

1354 Item 要素の order 属性の値が 0 以外の場合、仮想システムはその数値の昇順で稼働を開始する。order 属
1355 性の値が同じ仮想システムは同時に稼働を開始する。

1356 仮想システムの稼働停止は、この数値の降順で進めることを推奨する。Item 要素の order 属性値が
1357 '0'(zero)の仮想システムはいつでも停止できる。

1358 なお、shutdownorder 属性を指定していなければ、降順ではない実装固有の順序で仮想システムの稼働を停
1359 止できる。shutdownorder 属性には、シャットダウン順序を指定できる。

1360 Item 要素のオプション属性の中には、起動と停止をより詳細に制御できるものがある。startDelay 属性と
1361 stopDelay 属性は、順次実行で次のステップを実行するまでの待機時間を秒単位で指定する。startDelay、
1362 stopDelay ともデフォルトはゼロ(0)である。

1363 startAction 属性および stopAction 属性は、起動と停止で実行するアクションを指定する。
1364 startAction に有効な値は powerOn と none で、デフォルト値は'powerOn'である。stopAction 属性に有
1365 効な値は'powerOn'、'guestShutdown'、および'none'で、デフォルト値は'powerOff'である。stopAction
1366 属性を'guestShutdown'に設定している場合のアクションは、デプロイ・プラットフォームによって異なる。

1367 waitingForGuest 属性は、ゲスト・ソフトウェアが準備完了を報告してくるまでデプロイ・プラットフォームが待
1368 機するかどうかを指定する Boolean である。デフォルト値は'FALSE'である。その通信メカニズムはプラットフォー
1369 ムに依存する。

1370 以下に、StartupSection 要素の例を示す。

```
1371 <StartupSection>
1372   <Item ovf:id="vm1" ovf:order="0" ovf:startDelay="30" ovf:stopDelay="0"
1373     ovf:startAction="powerOn" ovf:waitingForGuest="true" ovf:stopAction="powerOff"/>
1374   <Item ovf:id="teamA" ovf:order="0"/>
1375   <Item ovf:id="vm2" ovf:order="1" ovf:startDelay="0" ovf:stopDelay="20"
1376     ovf:startAction="powerOn" ovf:stopAction="guestShutdown"/>
1377 </StartupSection>
```

1378 4.6.3 ScaleOutSection 要素

1379 ScaleOutSection 要素は、インスタンス化した仮想システムの数 VirtualSystemCollection 要素の
1380 中で動的に構成できるようにする。VirtualSystemCollection 要素に ScaleOut 要素を指定していない場
1381 合、仮想システムの数と仮想システムのコレクションの数は固定値となる。ScaleOutSection 要素は、作成す
1382 るレプリカの数の最大値と最小値を指定する。デプロイ時のデプロイ・プラットフォームでは、InstanceCount
1383 の最小値から最大値までの範囲でこの数が選択される。この値は利用者側に問い合わせることができるほか、
1384 デプロイ・プラットフォーム側で他のメタデータに基づいて決めることもできる。仮想システムと仮想システムのコレ
1385 クションは両方とも複製できるが、ScaleOutSection 要素は VirtualSystemCollection 要素にのみ記
1386 述される。

1387 以下に、ScaleOutSection 要素の例を示す。

```
1388 <VirtualSystemCollection ovf:id="web-tier">
1389   ...
1390   <ovf:ScaleOutSection ovf:id="web-server">
1391     <Info>Web tier</Info>
1392     <ovf:Description>Number of web server instances in web tier</ovf:Description>
1393     <ovf:InstanceCount ovf:default="4" ovf:minimum="2" ovf:maximum="8"/>
1394   </ovf:ScaleOutSection>
1395   ...
1396   <VirtualSystem ovf:id="web-server">
1397     <Info>Prototype web server</Info>
1398     ...
1399   </VirtualSystem>
1400 </VirtualSystemCollection>
```

1401 前述の例では、Web サーバー仮想マシンのインスタンスが2~8個(デフォルトは4個)存在するWeb層をデプロ
1402 イ・プラットフォームが作成している。デプロイ・プラットフォームは(たとえば、ユーザーに入力を促して)適切な選
1403 択を行う。たとえば、3個のレプリカが作成されているとすると、1番目のレプリカでゲスト・ソフトウェアが使用でき
1404 るOVF環境のコンテンツ構造は以下ようになる。

```
1405 <Environment ... ovfenv:id="web-server-1">
1406   ...
1407   <Entity ovfenv:id="web-server-2">
1408     ...
1409   </Entity>
1410   <Entity ovfenv:id="web-server-3">
1411     ...
1412   </Entity>
1413 </Environment>
```

1414 ここで、レプリカのOVF idは、プロトタイプ仮想システムのidに連番を付記したものである。デプロイ後はレプリ
1415 カ仮想システムすべてに連番のサフィックスが割り当てられるので、プロトタイプの基本 id のみを持つ仮想シス
1416 テムは存在しない。StartupSection 要素を指定している場合、各レプリカの起動番号はすべて同じになる。
1417 複数のレプリカにわたる起動番号を指定することはできない。

1418 例:

```
1419 <VirtualSystemCollection ovf:id="web-tier">
1420   ...
1421   <DeploymentOptionSection>
1422     <Info>Deployment size options</Info>
1423     <Configuration ovf:id="minimal">
1424       <Label>Minimal</Label>
1425       <Description>Minimal deployment scenario</Description>
1426     </Configuration>
1427     <Configuration ovf:id="common" ovf:default="true">
1428       <Label>Typical</Label>
1429       <Description>Common deployment scenario</Description>
1430     </Configuration>
1431     ...
1432   </DeploymentOptionSection>
1433   ...
1434   <ovf:ScaleOutSection ovf:id="web-server">
1435     <Info>Web tier</Info>
1436     <ovf:Description>Number of web server instances in web tier</ovf:Description>
1437     <ovf:InstanceCount ovf:default="4"/>
1438     <ovf:InstanceCount ovf:default="1" ovf:configuration="minimal"/>
1439   </ovf:ScaleOutSection>
1440   ...
1441 </VirtualSystemCollection>
```

1442 前述の例では、DeploymentOptionSection 要素を使用して、ScaleOutSection 要素の中で
1443 InstanceCount 要素の値を制御している。ScaleOutSection 要素での値は、OVF の property 要素を
1444 使用して制御することもできる。レプリカ 1 つにつき 1 度ずつ、OVF プロパティの指定を求められる。複数のレプリ
1445 カの間で OVF プロパティを共有する場合は、そのプロパティを VirtualSystemCollection 要素の中に配置
1446 する。

1447 4.7 仮想システムで使用する OVF Section 要素

1448 4.7.1 OperatingSystemSection 要素

1449 OperatingSystemSection 要素は、仮想システムで使用するゲスト・オペレーティング・システムを指定する。
1450 オペレーティング・システムは CIM_OperatingSystem.OSType プロパティで選択する。OVF の version と
1451 id は、プロパティの Value と ValueMap に対応する。

1452 id は必須の属性であり、ValueMap の整数値を参照する。version は省略可能な属性であり、対応する
1453 ValueMap の Value を参照する。version 属性はシンボリック文字列であることから国際化できない。

1454 Info 要素 (Section から導出) と Description 要素はいずれも、ローカライズを目的として外部化できる。5.2
1455 を参照。

1456 この例は、Microsoft Windows Server 2008 を指定するセクションである。

```
1457 <OperatingSystemSection ovf:id="76" ovf:version="Microsoft Windows Server 2008">
1458   <Info>Specifies the operating system installed</Info>
1459   <Description>Microsoft Windows Server 2008</Description>
1460 </OperatingSystemSection>
```

1461 4.7.2 InstallSection 要素

1462 InstallSection 要素は省略可能であり、VirtualSystem 要素でのみ使用する。この要素が存在する場合は StartupSection 要素の前に処理される。

1464 OVF パッケージの作成者は、InstallSection 要素を使用して、インストールを完了するには稼働を停止する前に仮想システムをリブートする必要があることを指定できる。ブートを実行すると、ゲスト・ソフトウェアはインストールを完了するために OVF 環境からスクリプトなどのソフトウェアを実行することが普通である。InstallSection 要素がないということは、インストールを完了するためのブートは必要ないということを意味する。たとえば、仮想システムにゲスト・ソフトウェアがない、またはシステム・イメージにゲスト・ソフトウェアがインストールされている場合、InstallSection 要素は必要ない。

1470 仮想システムのコレクションを構成する個々の仮想システムで InstallSection 要素を定義することもできる。この場合は複数のリブートを同時に実行してもかまわない。

1472 initialBootStopDelay 属性の値は、仮想システムの稼働停止を仮想化プラットフォームが待機する時間（単位は秒）である。この待機時間が経過しても仮想システムが停止しない場合、インストールは失敗したとみなされる。initialBootStopDelay のデフォルト値はゼロ(0)で、この待機時間に制限がないことを表し、仮想化プラットフォームは、仮想システムが自身の稼働を停止するまで待機する。仮想システムのゲスト・ソフトウェアは、仮想システムの稼働が停止するまで何回かブートすることがある。

1477 次の例では、ゲスト・ソフトウェアが仮想システムの稼働を停止するまで仮想化プラットフォームは5分間(300秒)待機する。5分以内に仮想マシンが停止しなかった場合、インストールは失敗したとみなされる。5分間の待機時間中、仮想システムが何回かリブートすることがある。

```
1480 <InstallSection ovf:initialBootStopDelay="300">
1481   <Info>Specifies that the virtual machine needs to be booted after having
1482   created the guest software in order to install and/or configure the software
1483   </Info>
1484 </InstallSection>
```

1485 4.7.3 EnvironmentFilesSection 要素

1486 EnvironmentFilesSection 要素を使用すると、カスタマイズの追加を許可しているゲスト・ソフトウェアに追加の環境ファイルを送ることができる。このようなファイルは、OVF 環境ファイルと同じトランスポート・メディアを使用して送られる。

1489 OVF 環境ファイルは、デプロイ機能により生成される。ただし、追加の環境ファイルは一切生成されないため、OVF パッケージの作成者が用意する必要がある。追加の環境ファイルは、EnvironmentFilesSection 要素の File 要素で指定する。この要素は各ファイルの ovf:fileRef 属性と ovf:path 属性を持つ。

1492 ovf:fileRef 属性は References 要素の File 要素を参照する。ovf:id 属性値と ovf:fileRef 属性値を照合することによって File 要素を特定する。

1494 ovf:path 属性は、ファイルが配置されているトランスポート・メディアでの相対位置を示す。

```

1495 <Envelope>
1496   <References>
1497     ...
1498   <File ovf:id="config" ovf:href="config.xml" ovf:size="4332"/>
1499   <File ovf:id="resources" ovf:href="http://mywebsite/resources/resources.zip"/>
1500 </References>
1501   ...
1502 <VirtualSystem ovf:id="...">
1503   ...
1504 <ovf:EnvironmentFilesSection ovf:required="false" ovf:transport="iso">
1505   <Info>Config files to be included in OVF environment</Info>
1506   <ovf:File ovf:fileRef="config" ovf:path="setup/cfg.xml"/>
1507   <ovf:File ovf:fileRef="resources" ovf:path="setup/resources.zip"/>
1508 </ovf:EnvironmentFilesSection>
1509   ...
1510 </VirtualSystem>
1511   ...
1512 </Envelope>

```

1513 前述の例では、OVF パッケージのファイル config.xml は OVF 環境の ISO イメージにコピーされ、ゲスト・ソフト
 1514 トウェアはロケーション /ovffiles/setup/cfg.xml でこのファイルにアクセスできる。一方、ファイル
 1515 resources.zip には、デプロイ時にロケーション /ovffiles/setup/resources.zip でアクセスできる。

1516 4.7.4 BootDeviceSection 要素

1517 以前のバージョンの OVF では、デフォルトのブート・デバイス以外では仮想システムをブートできなかった。これ
 1518 は、OVF のデプロイで見られるさまざまなシナリオで制限となることがわかった。

1519 a) これまで、NIC から PXE ブートするように仮想システムをセットアップする必要があるかどうかを指定す
 1520 る方法はなかった。また、セカンダリ・デバイスや USB デバイスからブートするように仮想システムを
 1521 セットアップする必要があるかどうかを指定する方法もなかった。したがって、このような代替ブート・
 1522 ソースと対応する設定を指定できるようにする必要があった。

1523 b) 実装経験を通じて、複数の代替ブート構成を指定できるようにする必要があることもわかった。たとえ
 1524 ば、OVF の「準備」フェーズでは、修正ディスクを使用して、仮想システムにパッチを適用することが必
 1525 要になる可能性がある。

1526 共通情報モデル (CIM) では、デスクトップやサーバーの BIOS 向けに業界で普及しているブート順序のユース
 1527 ケースを処理するアーティファクトを定義している。CIM_BootSourceSetting クラスは、ブート・ソースとして
 1528 使用する NIC やディスクのような個々のブート・ソース・デバイスを定義する。個々のデバイスは、
 1529 CIM_BootSourceSetting クラスで指定された一意の ID で特定する。

1530 ブート構成は、集約クラス CIM_BootConfigSetting の下でブート・デバイスのシーケンスとして定義する。し
 1531 たがって、1 つ以上の CIM_BootSourceSetting プロパティのシーケンスは、CIM_BootConfigSetting
 1532 クラスに集約される。

1533 OVF エンベロープでは、このようなブート構成を複数個まとめて `BootDeviceSection` 要素に集約できる。この
1534 ような各 `BootDeviceSection` 要素は `VirtualHardwareSection` 要素の中に記述できる。

1535 デプロイ機能は、それが選択したブート構成で定義されているとおりに、仮想システムのブート・ソース・シーケ
1536 スのセットアップを試みる。複数のブート構成が存在する場合、ブート構成の選択が問題になる。デプロイ機能は、
1537 デプロイの状態およびブート構成構造の `Caption` 要素に基づいてブート構成を選択する。

1538 次の例では、ブート・ソースを特定のデバイス(ネットワーク)として `Pre-Install` 構成で指定し、デバイス・タイプ
1539 (ハード・ディスク)を `Post-Install` 構成で指定している。

```
1540 EXAMPLE:  
1541 <Envelope>  
1542 ...  
1543 <VirtualSystem ovf:id="...">  
1544 ...  
1545 <ovf:BootDeviceSection>  
1546 <Info>Boot device order specification</Info>  
1547 <bootc:CIM_BootConfigSetting>  
1548 <bootc:Caption>Pre-Install</bootc:Caption>  
1549 <bootc:Description>Boot Sequence for fixup of disk</bootc:Description>  
1550 <boots:CIM_BootSourceSetting>  
1551 <boots:Caption>Fix-up DVD on the network</boots:Caption>  
1552 <boots:InstanceID>3</boots:InstanceID> <!-- Network device-->  
1553 </boots:CIM_BootSourceSetting>  
1554 <boots:CIM_BootSourceSetting>  
1555 <boots:Caption>Boot virtual disk</boots:Caption>  
1556 <boots:StructuredBootString>CIM:Hard-Disk</boots:StructuredBootString>  
1557 </boots:CIM_BootSourceSetting>  
1558 </bootc:CIM_BootConfigSetting>  
1559 </ovf:BootDeviceSection>  
1560 ...  
1561 </VirtualSystem>  
1562 </Envelope>
```

1563 5 OVF パッケージのオーサリング

1564 5.1 作成

1565 OVF パッケージの作成では以下の作業を実施する。

- 1566 i) VM のセットを仮想ディスクのセット上にパッケージ化する。
- 1567 ii) これらの仮想ディスクを適切にエンコードする。
- 1568 iii) 仮想ハードウェア、ライセンシングなどのカスタマイズ・メタデータの仕様を OVF デスクリプタに追加す
1569 る。
- 1570 iv) 必要に応じて、パッケージに電子署名を添付する。

1571 仮想化プラットフォームが OVF を利用し、そのコンテンツから一連の仮想マシンを作成すると、OVF パッケージの
1572 デプロイ・プロセスが発生する。

1573 OVF の作成は簡単で、仮想化プラットフォームから OVF パッケージに既存の仮想マシンをエクスポートし、その
1574 仮想マシンに正しいインストールと実行に必要なメタデータを追加するだけである。これにより、仮想マシンは特
1575 定のハイパーバイザーでの現在の実行時状態から OVF パッケージに変換される。このプロセスでは、仮想マシ
1576 ンのディスクを圧縮して、配布しやすくすることができる。

1577 業務用の仮想アプライアンスでは、標準のビルド環境を使用して、OVF パッケージを作成することもできる。たと
1578 えば、OVF デスクリプタはソース制御システムを使用して管理できる。また、OVF パッケージは、再現性のあるス
1579 クリプト環境 (make ファイルなど) やさまざまなベンダーが提供しているアプライアンス・ビルド・ツールキットを使
1580 用してビルドできる。

1581 作成した OVF パッケージには、アプライアンス固有のインストール後の構成メタデータを添付できる。このメタ
1582 データとして、アプライアンスのインターフェース言語を必要に応じてローカライズするためのメタデータ、EULA の
1583 確認、サインオフ、適用に関するメタデータ、リソース構成のメタデータなどがある。また、I/O、タイムキーピング、
1584 メモリ管理、監視、順序を指定したシャットダウンなどを強化するためにゲストに追加する専用のドライバーやエー
1585 ジェントなどの各種ツールもある。

1586 OVF デスクリプタの作成プロセスとは、本質的には、仮想アプライアンスで使用する構成要素を組み立てることで
1587 ある。前述のとおり、仮想アプライアンスは、それを構成する仮想システムの記述、アプライアンスとゲスト・ソフト
1588 ウェアに関連するメタデータ、およびひと揃いの参照ファイルで定義する。OVF デスクリプタは、必須情報すべて
1589 を集約し、参照する中心的要素である。OVF デスクリプタの主な構成要素はセクションである。仮想アプライア
1590 ンスの記述に使用できるセクションについては、第 4 節で紹介している。

1591 5.2 国際化

1592 OVF 仕様は、オプションの `ovf:msgid` 属性を使用して、ローカライズ可能なメッセージをサポートしている。ロー
1593 カライズしたメッセージを使用して、デプロイの際に目的の言語でユーザー・メッセージを表示できる。

```
1594 <Envelope ...>  
1595   ...  
1596   <Info ovf:msgid="info.os">Operating System</Info>  
1597   ...  
1598   <Strings xml:lang="da-DA">  
1599     <Msg ovf:msgid="info.os">Operativsystem</Msg>  
1600     ...  
1601   </Strings>  
1602   <Strings xml:lang="de-DE">  
1603     <Msg ovf:msgid="info.os">Betriebssystem</Msg>  
1604     ...  
1605   </Strings>  
1606 </Envelope>
```

1607 前述の例では、あるセクションの中で `Info` 要素を定義している。このセクションの情報は、仮想システムのオペ
1608 レーティング・システムに関連するものである。属性 `ovf:msgid="info.os"` は、`Info` 要素の開始タグと終了
1609 タグの間の文字列がローカライズしたメッセージに置き換え可能であることを示している。ローカライズしたメッ

1610 セージは、`info.os`の中でそのメッセージのメッセージIDで参照される。適切なローカライズ済みメッセージが
1611 `Strings`セクションにある場合、デフォルトのメッセージ"Operating System"は、現在の地域に対応する
1612 `Strings`セクションから取得したローカライズ済みのメッセージに置き換えられる。

1613 前述の例では、ローカライズした文字列は、OVFデスク립タの中に格納されている。また、外部文字列バンドル
1614 を使用して、ローカライズした文字列を外部のOVFデスク립タに格納することもできる。この例を以下に示す。

```
1615 <Envelope ...>  
1616   <References>  
1617     ...  
1618     <File ovf:id="da-DA-resources" ovf:href="danish.msg"/>  
1619     <File ovf:id="de-DE-resources" ovf:href="german.msg"/>  
1620     ...  
1621   </References>  
1622   ...  
1623   <Info ovf:msgid="info.os">Operating System</Info>  
1624   ...  
1625   <Strings xml:lang="da-DA" ovf:fileRef="da-da-resources"/>  
1626   <Strings xml:lang="de-DE" ovf:fileRef="de-de-resources"/>  
1627 </Envelope>
```

1628 "Operating System"のローカライズ済みメッセージは、`danish.msg` ファイルと `german.msg` ファイルで定義す
1629 る。外部メッセージ・ファイル `german.msg` のフォーマットを以下の例に示す。

```
1630 <Strings  
1631   xmlns:ovf="http://schemas.dmtf.org/ovf/envelope/1"  
1632   xmlns="http://schemas.dmtf.org/ovf/envelope/1"  
1633   xml:lang="de-DE">  
1634   ...  
1635   <Msg ovf:msgid="info.os">Betriebssystem</Msg>  
1636   ...  
1637 </Strings>
```

1638 一番上の `Strings` セクションでは、`xml:lang` 属性を使用して、特定の外部メッセージ・ファイルのロケールを
1639 定義している。この外部メッセージ・ファイルには、OVF デスク립タで使用するローカライズ済みメッセージの
1640 `Msg` 要素が記述されている。

1641 ローカライズしたリソースを使用するには、現在位置に基づいて外部ファイルを参照するという方法もある。この
1642 方法は、たとえば、現在の場所に基づいてライセンス条項を表示するために使用できる。この場合は、場所ごと
1643 のテキスト・ファイルにライセンスのテキストを記述する。以下の例は、外部プレーン・テキスト・ファイルを参照して、
1644 ローカライズしたライセンスを表示する方法を示している。

```
1645 <Envelope xml:lang="en-US">  
1646   <References>  
1647     <File ovf:id="license-en-US" ovf:href="license-en-US.txt"/>  
1648     <File ovf:id="license-de-DE" ovf:href="license-de-DE.txt"/>  
1649   </References>  
1650   ...
```

```

1651 <VirtualSystem ovf:id="...">
1652   <EulaSection>
1653     <Info>Licensing agreement</Info>
1654     <License ovf:msgid="license">Unused</License>
1655   </EulaSection>
1656   ...
1657 </VirtualSystem>
1658   ...
1659 <Strings xml:lang="en-US">
1660   <Msg ovf:msgid="license" ovf:fileRef="license-en-US">Invalid license</Msg>
1661 </Strings>
1662 <Strings xml:lang="de-DE">
1663   <Msg ovf:msgid="license" ovf:fileRef="license-de-DE">Ihre Lizenz ist nicht
1664   gültig</Msg>
1665 </Strings>
1666 </Envelope>

```

1667 License 要素には `ovf:msgid` 属性がある。Strings セクションでは、`ovf:fileRef` 属性を使用して、さま
 1668 ざまな場所の `ovf:msgid` をファイル参照にリンクしている。`ovf:fileRef` 属性には、OVF デスクリプタの
 1669 References セクションに対応するエントリーがある。References セクションのエントリーは、最終的にライセ
 1670 ンス・テキストを記述した外部テキスト・ファイルに解決される。

1671 5.3 拡張性

1672 OVF仕様では、いくつかの方法でカスタム・メタデータをOVFデスクリプタに追加できる。

- 1673 • 新しい Section 要素を Section 代替グループの一部として定義しておく、OVF スキーマでセクショ
 1674 ンの設定が認められていれば、その新しい Section 要素を使用できる。
- 1675 • OVF スキーマはオープン・コンテンツ・モデルを使用する。このモデルでは、既存のタイプはすべて、末
 1676 尾に要素を追加して拡張できる。拡張ポイントを宣言するには、`namespace="##other"`を指定した
 1677 `xs:any` 宣言を OVF スキーマで使用する。
- 1678 • OVF スキーマでは、既存のタイプに属性を追加できる。

1679 OVF仕様の設計目標は、上位互換性と下位互換性を確保することである。上位互換性とは、より新しい仕様(ま
 1680 たはカスタム拡張)の機能を使用しているOVFデスクリプタを、過去の仕様に沿って記述されたOVF利用側また
 1681 は新しい仕様による特定の拡張を認識できないOVF利用側でも理解できるという意味である。OVF利用側では、
 1682 予測可能で信頼性が高く、ユーザーに有益な方法で、拡張を含むOVFパッケージを受け入れるか、拒否するか
 1683 を判断する必要がある。

1684 5.3.1 代替グループ

1685 OVF は、セクションの追加が可能であると同時に、既存のセクションを新しいコンテンツで拡張できるオープン・コ
 1686 ンテンツ・モデルをサポートしている。拡張では、正しい動作のためには要素内の情報が必須なのか省略可能な
 1687 のかを Boolean 属性 `ovf:required` で指定する。

1688 追加セクションを OVF デスクリプタに挿入するには、`ovf:Section` 代替グループに新しいメンバーを定義する。
 1689 つまり、新しいセクションが、Section 要素のベース・スキーマを拡張するのである。新しいセクションを使用して、

1690 OVF 仕様で定義されている既存のセクションとは無関係なメタデータを定義できる。新しい Section にある
1691 <Info>要素を使用すると、デプロイ機能で理解できないセクションがあった場合に、そのセクションに関する情
1692 報を利用側に表示できる。

1693 次の例は、新しい Section <ns:BuildInformationSection>を追加する方法を示している。Section
1694 は名前空間 ns を使用する。この名前空間は親要素で参照される。たとえば親要素が<Envelope>要素の場合
1695 は、次のようになる。<Envelope xmlns="http://schemas.dmtf.org/ovf/envelope/2"
1696 xmlns:ns="http://acme.org/ovf/extension/ns">のように参照される。ovf:Section の
1697 substitutionGroup での要求どおり、新しいセクションには<Info>要素が含まれる。BuildNumber、
1698 BuildDate、BuildSystem の各要素が新しい要素である。これらの要素は、参照先の名前空間スキーマで定
1699 義する。ovf:required 属性は"false"に設定する。これは、このセクションを実装できない場合でも、デプロイ機
1700 能は警告を表示するだけで、失敗はしないことを示す。

1701 新しいセクションを追加する例:

```
1702 <ns:BuildInformationSection ovf:required="false">  
1703   <Info>Specifies information about how a virtual machine was created</Info>  
1704   <BuildNumber> ... </BuildNumber >  
1705   <BuildDate> ... </BuildDate >  
1706   <BuildSystem> ... </BuildSystem>  
1707   ...  
1708 </ns:BuildInformationSection>
```

1709 前述の例で追加したセクションの XSD スキーマは以下のようになる。

```
1710 <?xml version="1.0" encoding="UTF-8"?>  
1711 <xs:schema xmlns:ns=http://acme.org/ovf/extension/ns  
1712   xmlns:ovf=http://schemas.dmtf.org/ovf/envelope/2  
1713   xmlns:xs=http://www.w3.org/2001/XMLSchema  
1714   targetNamespace=http://acme.org/ovf/extension/ns  
1715   elementFormDefault="qualified"  
1716   attributeFormDefault="qualified">  
1717 <xs:import namespace=http://schemas.dmtf.org/ovf/envelope/2  
1718   schemaLocation="dsp8023.xsd"/>  
1719   <xs:element name="BuildInformationSection" type="ns:BuildInformationSection_Type"  
1720   substitutionGroup="ovf:Section">  
1721     <xs:annotation>  
1722       <xs:documentation>Element substitutable for Section since  
1723       BuildInformationSection_Type is a derivation of Section_Type  
1724       </xs:documentation>  
1725     </xs:annotation>  
1726   </xs:element>  
1727   <xs:complexType name="BuildInformationSection_Type">  
1728     <xs:annotation>  
1729       <xs:documentation>An ACME specific section.</xs:documentation>  
1730     </xs:annotation>  
1731     <xs:complexContent>
```

```
1732 <xs:extension base="ovf:Section_Type">
1733   <xs:sequence>
1734     <xs:element name="BuildNumber" maxOccurs="unbounded">
1735       <xs:complexType>
1736         <xs:anyAttribute namespace="##any" processContents="lax"/>
1737       </xs:complexType>
1738     </xs:element>
1739     <xs:element name="BuildDate" maxOccurs="unbounded">
1740       <xs:complexType>
1741         <xs:anyAttribute namespace="##any" processContents="lax"/>
1742       </xs:complexType>
1743     </xs:element>
1744     <xs:element name="BuildSystem" maxOccurs="unbounded">
1745       <xs:complexType>
1746         <xs:anyAttribute namespace="##any" processContents="lax"/>
1747       </xs:complexType>
1748     </xs:element>
1749   </xs:sequence>
1750   <xs:anyAttribute namespace="##any" processContents="lax"/>
1751 </xs:extension>
1752 </xs:complexContent>
1753 </xs:complexType>
1754 </xs:schema>
```

1755 このスキーマは、`ovf:Section` セクションの `BuildInformationSection` 代替グループを定義する。
1756 `BuildInformationSection` 代替グループのタイプは `BuildInformationSection_Type` である。
1757 `BuildInformationSection_Type` タイプは、`ovf:Section_Type` を基本タイプとして定義し、
1758 `BuildNumber`、`BuildDate`、`BuildSystem` の各要素を使用して `ovf:Section_Type` を拡張する。

1759 5.3.2 要素

1760 既存のセクションにある新しい要素を、そのセクションの末尾に追加できる。`Envelope`、`VirtualSystem`、
1761 `VirtualSystemCollection`、`Content`、および `Strings` の各セクションは、セクション末尾への要素の追
1762 加をサポートしていない。使用している名前空間を親要素で参照する必要がある。また、この名前空間はOVFの
1763 名前空間と異なるものでなければならない。追加の要素を使用して、OVFデスクリプタの特定セクションで指定し
1764 た情報を拡張できる。

1765 以下に、既存セクションの拡張例を示す。

```
1766 <AnnotationSection>
1767     <Info>Specifies an annotation for this virtual machine</Info>
1768     <Annotation>This is an example of how a future element (Author) can still be parsed
1769 by older clients</Annotation>
1770     <!-- AnnotationSection extended with Author element -->
1771     <ns:Author ovf:required="false">John Smith</ns:Author>
1772 </AnnotationSection>
```

1773 この例では、Annotation セクションに要素が追加されている。この要素は、このデスクリプタの Author に関する情報で Annotation セクションを拡張する。この新しい要素は、ns 名前空間に属する。

1775 5.3.3 属性

1776 追加情報を使用して OVF デスクリプタを拡張するための 3 番目の方法では、既存の要素にカスタム属性を追加する。このような属性を使用して、既存の要素で得られた情報を拡張できる。

```
1778 <!-- Optional custom attribute example -->
1779     <Network ovf:name="VM network" ns:desiredCapacity="1 Gbit/s">
1780         <Description>The main network for VMs</Description>
1781     </Network>
```

1782 前述の例は、Network 要素の desiredCapacity 属性を追加する例を示している。この新しい属性は ns 名前空間で定義する。

1784 OVF ドキュメント拡張の詳しい例については、ANNEX E を参照。

1785 5.4 適合

1786 OVF 仕様は、OVF デスクリプタの OVF 仕様適合レベルを 3 段階で定義している。最高の適合レベルを 1 として、各適合レベルは以下ようになる。

- 1788 • OVF 仕様で定義しているメタデータのみを使用し、カスタム拡張が存在しない OVF デスクリプタ。
1789 適合レベル: 1。
- 1790 • カスタム拡張を設定したメタデータを使用しているが、それらの拡張がすべてオプションである OVF デ
1791 スクリプタ。
1792 適合レベル: 2。
- 1793 • カスタム拡張を設定したメタデータを使用し、それら拡張のうち少なくとも 1 つが必須である OVF デスク
1794 リプタ。
1795 適合レベル: 3。

1796 適合レベル 3 を使用すると可搬性が制限される。つまり、このカスタム拡張をサポートしていない仮想プラットフォームには、この OVF パッケージをデプロイできない可能性がある。

1798 5.5 仮想ハードウェアの記述

1799 ここでは、ごく一般的なハードウェアについて説明する。特に、ここでは仮想ディスクとネットワーク・アダプターが必要であると規定しているが、具体的にどのハードウェアであるべきかは示していない。たとえば、SCSI または IDE ディスク、E1000 または Vlance ネットワーク・カードが適切に使用できる。すなわち、この仕様が汎用なもの

1802 であれば、アプライアンスがデバイスの存在を発見し、関連するドライバーをロードするとみなすことができる。こ
1803 の場合、アプライアンスの作成者は、このアプライアンスをさまざまなドライバーとともに開発し、関連する仮想
1804 ハードウェア上でアプライアンスをテストして、動作を確認していると考えられる。

1805 アプライアンスで使用しているゲスト・ソフトウェアが必要とするものと同じハードウェア・デバイスまたは同じカテ
1806 ゴリーのデバイスを用意していない仮想化プラットフォームでは、OVF パッケージをデプロイすると、重要ではある
1807 ものの把握しにくいインストール・エラーが発生する可能性がある。この場合、インストールやブートに失敗するう
1808 え、ユーザー側では問題をデバッグできない可能性がある。また、カスタマー・サポートへの問い合わせが増加し、
1809 全般的に顧客満足度が低下することも考えられる。より制約が強く、きめ細かな仮想ハードウェア仕様は(必要な
1810 デバイスがリストされているため)誤った実行の可能性を低減できるが、このような仕様では、このアプライアンス
1811 をインストールまたはブートできる可能性のあるシステムが制限される。

1812 仮想アプライアンス・モデルを目指す傾向が ISV にあるのは、インストールの簡索性、堅牢性、予測可能性が主
1813 な理由であると考えられる。したがって、アプライアンス開発者は、目的のアプライアンスがきわめて限定的な
1814 ハードウェアを必要としない限り、ハードウェア仕様の汎用性が低いものよりも、より高いアプライアンスを作成す
1815 べきである。まず、アプライアンスの可搬性は、仮想マシンで使用するゲスト・ソフトウェアおよびそのゲスト・ソフト
1816 ウェアがサポートする仮想ハードウェアの範囲に基づくものとする。

1817 アプライアンス・ベンダーが目的とするターゲット仮想化プラットフォームすべての仮想ハードウェアのデバイス・ド
1818 ライバーを備えた仮想マシンを、アプライアンス・ベンダーが作成することが理想的である。しかし、現在、仮想ア
1819 プライアンス・ベンダーや作成者に依存しないドライバーを配布する仮想化プラットフォーム・ベンダーは多くない。
1820 一方、OVF モデルでは、仮想ハードウェアとアプライアンス間のインターフェース管理をさらに簡素化することを目
1821 的として、明示的なインストール・モードをサポートしている。このモードでは、個々の仮想マシンをインストール直
1822 後に一度ブートし、特定の仮想化プラットフォームのローカライズやカスタマイズを可能にする。このモードでは、
1823 仮想マシンは仮想化プラットフォームを検出し、適切な一連のデバイス・ドライバーをインストールできる。このドラ
1824 イバーとして、インストールしたゲストが初めてリポートすると(最初のブートでゲストに接続したフロッピー・ドライ
1825 ブや CD ドライブなどを使用して)利用できるようになる、プラットフォーム固有のドライバーなどがある。また、再イ
1826 ンストールと名前指定などによるカスタマイズのみを必要とする sysprep 済み Windows VM では、リポートするこ
1827 とで、利用者の操作を必要とせずに、イメージの名前付けおよび調整を実現できる。

1828 以下の例は、同一のデスク립タで指定したさまざまな仮想化プラットフォームで使用する複数の仮想ハードウェ
1829 ア・プロファイルを示している。

```
1830 <VirtualHardwareSection>  
1831   <Info>500Mb, 1 CPU, 1 disk, 1 nic virtual machine, Platform A</Info>  
1832   <System>  
1833     ...  
1834   </System>  
1835   <Item>  
1836     ...  
1837   </Item>  
1838   ...  
1839 </VirtualHardwareSection>  
1840 <VirtualHardwareSection>  
1841   <Info>500Mb, 1 CPU, 1 disk, 1 nic virtual machine, Platform B</Info>  
1842   <System>
```

```
1843     ...
1844     </System>
1845     <Item>
1846     ...
1847     </Item>
1848     ...
1849 </VirtualHardwareSection>
```

1850 このタイプのプロファイルでは、さまざまな仮想化プラットフォームや機能をサポートするようにベンダーがハード
1851 ウェアの記述を調整できる。ローカル・プラットフォームで使用できる機能セットから最新または最も能力の高いも
1852 のが OVF デプロイ機能で選択されるという前提の下で、具体的な仮想化プラットフォームでは、それがサポート
1853 可能な特定の仮想ハードウェア・セクションから目的のハードウェアを選択できる。

1854 以下の例は、具体的なタイプの仮想ハードウェアの定義方法を示している。`rasd:ResourceSubType` に複数
1855 のオプションを指定する場合は、オプション間をスペース文字 1 つで区切る。これにより、デプロイ機能はインスタ
1856 ンス化する仮想ハードウェア・タイプを選択する。

1857 OVF パッケージで複数のリソース・サブタイプを指定しておくこと、管理対象環境の
1858 `CIM_ResourceAllocationSettingData` クラスに割り振るうえで適切な値がそこからデプロイ機能によって選択さ
1859 れる。適切な値が複数個存在する場合は、それらの値のいずれかがデプロイ機能で選択される。デプロイ機能で
1860 適切な値を見出せなかった場合、デプロイは失敗する。

```
1861 <Item>
1862     <rasd:ElementName>SCSI Controller 0</rasd:ElementName>
1863     <rasd:InstanceID>1000</rasd:InstanceID>
1864     <rasd:ResourceSubType>LsiLogic BusLogic</rasd:ResourceSubType>
1865     <rasd:ResourceType>6</rasd:ResourceType>
1866 </Item>
1867 <Item>
1868     <rasd:ElementName>Harddisk 1</rasd:ElementName>
1869     <rasd:HostResource>ovf:/disk/vmdisk1</rasd:HostResource>
1870     <rasd:InstanceID>22001</rasd:InstanceID>
1871     <rasd:Parent>1000</rasd:Parent>
1872     <rasd:ResourceType>17</rasd:ResourceType>
1873 </Item>
```

1874 5.6 デスクリプタの例

1875 以下の例は、OVF デスクリプタの完成例である。これらの例は XML 検証に合格している。

1876 ANNEX A は、単一の仮想システムに使用する OVF デスクリプタの例である。

1877 ANNEX B は、複数の仮想システムに使用する OVF デスクリプタの例である。

1878 ANNEX C は、複数のアプリケーションを備えた単一の仮想システム(LAMP スタック)に使用する OVF デスクリ
1879 プタの例である。

1880 ANNEX D は、複数のアプリケーションを備えた複数の仮想システム(2つの仮想システムを持つ LAMP スタック)
1881 に使用する OVF デスクリプタの例である。

1882 6 OVF パッケージのデプロイ

1883 6.1 デプロイ

1884 デプロイにより、OVF パッケージ中の仮想マシンが対象の仮想化プラットフォームが理解できる実行時フォーマット
1885 に変換され、適切なリソースを割り当てられ正しい仮想ハードウェアでサポートされる。デプロイでは、プラット
1886 フォームによる OVF の整合性検証で、OVF パッケージがその配布段階で改変されていないことおよびローカル
1887 仮想ハードウェアと互換性があることを確認する。また、対象の仮想化プラットフォーム上にある特定の環境にリ
1888 ソースを割り当て、またこの環境に合わせて仮想マシンを構成する。このプロセスの作業には、仮想マシンの接
1889 続先ネットワーク(物理または仮想)の割り当てと構成、仮想ハードディスク、一時的なデータセットの割り当て、ク
1890 ラスター化したストレージやネットワークストレージなどへの接続を含むストレージ資源の VM への割り当て、CPU
1891 やメモリ・リソースの構成、アプリケーション・レベルのプロパティのカスタマイズなどがある。OVF は、プロセッ
1892 サー・アーキテクチャやハードウェア・プラットフォームとの間でのゲスト・ソフトウェアの変換はサポートしていない。
1893 デプロイでは、OVF デスクリプタから得られた要件と、OVF パッケージ内に指定されたコンテンツを持つ仮想ディ
1894 スクの集合と互換性を持つ1つ以上の仮想マシンがインスタンス化される。

1895 OVF パッケージのデプロイ環境は、デプロイ先の仮想化プラットフォームによって異なる。コマンドライン・ベース
1896 のこともあれば、スクリプト化されていることも、グラフィカルなデプロイ・ウィザードの可能性もある。OVF の一般
1897 的なデプロイでは、以下の情報を表示するか、または入力を求める。

- 1898 • OVF パッケージに関する情報(ProductSection に記述されている情報)を表示し、ライセンス契約
1899 に同意するかどうかをユーザーに尋ねるか、無人インストールを処理する。
- 1900 • 仮想ハードウェアに OVF 仕様との互換性があることを検証する。
- 1901 • 仮想マシンのストレージの場所および OVF パッケージにある論理ネットワークの接続先物理ネットワー
1902 クをユーザーに尋ねる。
- 1903 • ProductSection に構成されているプロパティの値の入力をユーザーに求める。

1904 この構成が完了すると、仮想マシンが正常に起動し、(DHCP などの標準的な手順を使用して)ローカル・ネット
1905 ワークで有効な ID を取得できることが普通である。特定の IP ネットワーク構成など、そのデプロイ環境固有の値
1906 の入力を求めるためにプロパティを使用する。アプライアンスを初めてブートした後、Web インターフェースなどア
1907 プライアンス自体に用意されている管理インターフェースを使用して、アプライアンス内部のソフトウェアを詳しく構
1908 成する。

1909 6.2 OVF 環境デスクリプタ

1910 OVF 環境デスクリプタは、仮想ディスクにインストールされたソフトウェアに関するメタデータを記述した XML 文書
1911 である。OVF 仕様は、仮想ハードウェア、ディスク、ネットワーク、リソース要件、カスタマイズ用パラメーターなど、
1912 ソフトウェアのデプロイに使用する共通セクションを定義する。デスクリプタは、後の段階で詳しい情報を追加でき
1913 るように拡張可能として設計されている。

1914 仮想アプライアンスは、デプロイ先の環境で適切に機能するようにカスタマイズが必要になることが多い。OVF 環
1915 境は、デプロイ構成をゲスト・ソフトウェアに伝達するための標準的で拡張可能な方法を仮想化プラットフォーム
1916 向けに用意している。

1917 OVF 環境は、デプロイ時にゲスト・ソフトウェア向けに使用するカスタマイズ情報を記述した XML 文書である。こ
1918 の XML 文書で提供できる情報の例として次のものがある。

- 1919 • ホスト名、IP アドレス、サブネット、ゲートウェイなど、オペレーティング・システム・レベルの構成
- 1920 • Active Directory サーバーやデータベースなどの外部サービスの DNS 名をはじめとするアプリケーション・レベルの構成
- 1921

1922 デプロイの際に構成するプロパティは、ProductSection メタデータを使用して OVF デスクリプタで指定し、デ
1923 プロイ中にはウィザード形式のインターフェースを使用して、ユーザー入力を受け付ける。

1924 たとえば、OVF 環境では、ゲスト・ソフトウェアが多層サービス間のネットワーク設定を自動化できる。また、Web
1925 サーバーはデータベース・サーバーの IP アドレスを使用して自身を自動的に構成できるので、ユーザーによる手
1926 動操作を必要としない。

1927 仮想マシンのゲスト・ソフトウェアと基盤となる仮想化プラットフォームとの通信に使用できるベンダー間共通で使
1928 える仮想化された標準的なデバイスは存在しないので、標準 OVF 環境の定義にはいくらかの課題がある。OVF
1929 仕様は、以下のように、OVF 環境定義を 2 つの部分に分割する手法を採用している。

- 1930 • 標準的に用いる protocol: どの情報をどのようなフォーマットで利用可能とするかを指定する。
- 1931 • transport: その情報の入手方法を指定する。

1932 本仕様では、動的に生成される ISO イメージ上でゲスト・ソフトウェアが利用できる OVF 環境(XML 文書)を実現
1933 する ISO トランスポートのサポートをすべての実装に要求する。

1934 6.3 デプロイ時のリソース構成オプション

1935 OVF パッケージには、仮想アプライアンス向けのリソース構成オプションを追加できる。これにより、パッケージの
1936 利用者は、使用目的に基づいて個々のリソースを決定しなくても、容易に初期設定を入手できる。
1937 Description 要素および Label 要素は、人間が読める形式で以下のようなリソース構成を列挙する。

- 1938 • ソフトウェア評価設定
- 1939 • 10~100 人のユーザーで構成するワークグループの設定
- 1940 • 100~1,000 人のユーザーで構成するワークグループの設定
- 1941 • 大規模エンタープライズのワークグループ設定

1942 デプロイ機能では構成の選択を要求する。上記の構成のリストは、適切な初期リソース構成での使用を想定した
1943 ものである。

1944 構成の列挙の例:

```
1945 <DeploymentOptionSection>
1946    <Configuration ovf:id="eval">
1947      <Label>Software Evaluation</Label>
1948      <Description>Software evaluation setup</Description>
1949    </Configuration>
1950    <Configuration ovf:id="small" ovf:default="yes">
1951      <Label>Small</Label>
1952      <Description>10-100 person workgroup setup</Description>
1953    </Configuration>
1954    <Configuration ovf:id="medium">
```

```

1955     <Label>Medium</Label>
1956     <Description>100-1000 person workgroup setup</Description>
1957 </Configuration>
1958 <Configuration ovf:id="large">
1959     <Label>Large</Label>
1960     <Description>Large enterprise workgroup setup</Description>
1961 </Configuration>
1962 </DeploymentOptionSection>

```

1963 以下の OVF デスクリプタのスニペットは、リソース要求での構成オプションの使用例である。この場合、利用者が
 1964 'eval'を選択すると、Item ovf:configuration="eval"に記述されたリソース割り当てデータが使用され
 1965 る。

1966 リソース要求の例:

```

1967 <ResourceAllocationSection>
1968     <Info>Defines reservations for CPU and memory</Info>
1969     <Item>
1970         ... default configuration ...
1971     </Item>
1972     <Item ovf:configuration="eval">
1973         ... replaces the default configuration if the "eval" configuration if selected ...
1974     </Item>
1975 </ResourceAllocationSection>

```

1976 以下の OVF デスクリプタのスニペットは、VirtualHardwareSection での構成オプションの使用例である。
 1977 この場合、利用者が'large'を選択すると、Item ovf:configuration="large"に記述されたリソース割り当
 1978 てデータが使用される。

1979 VirtualHardwareSection の例:

```

1980 <VirtualHardwareSection>
1981     <Info>...</Info>
1982     <Item>
1983         <rasd:AllocationUnits>hertz * 10^6</rasd:AllocationUnits>
1984         <rasd:ElementName>1 CPU and 500 MHz reservation</rasd:ElementName>
1985         <rasd:InstanceID>1</rasd:InstanceID>
1986         <rasd:Reservation>500</rasd:Reservation>
1987         <rasd:Limit>1100</rasd:Reservation>
1988         <rasd:ResourceType>3</rasd:ResourceType>
1989         <rasd:VirtualQuantity>1</rasd:VirtualQuantity>
1990     </Item>
1991     ...
1992     <Item ovf:configuration="large">
1993         <rasd:AllocationUnits>hertz * 10^6</rasd:AllocationUnits>
1994         <rasd:ElementName>1 CPU and 800 MHz reservation</rasd:ElementName>
1995         <rasd:InstanceID>1</rasd:InstanceID>

```

```
1996 <rasd:Reservation>800</rasd:Reservation>
1997 <rasd:ResourceType>3</rasd:ResourceType>
1998 <rasd:VirtualQuantity>1</rasd:VirtualQuantity>
1999 </Item>
2000 </VirtualHardwareSection>
```

2001 **6.4 Property 要素を使用したデプロイ時の製品のカスタマイズ**

2002 OVF デスクリプタには、OVF 環境によって提供されるカスタマイズに関する情報を含むゲストソフトウェアの記述
2003 を含むことができる。この情報は、OVF デスクリプタの ProductSection で Property 要素を使用して指定す
2004 る。

2005 各 Property 要素で指定できる属性には、ovf:key、ovf:type、ovf:qualifiers、ovf:value、
2006 ovf:userConfigurable の 5 種類がある。

2007 ovf:key 属性は、Property 要素固有の ID である。

2008 ovf:type 属性は、Property 要素に記述された値のタイプを示す。

2009 ovf:qualifiers 属性は、ovf:type 属性に関する追加情報を表す。これにより、CIM 値のマッピングを使用でき
2010 るようになる。

2011 ovf:value 属性は、Property 要素に値を提供するために使用する。

2012 ovf:userConfigurable 属性は、指定の値がデフォルト値であるかどうか、またデプロイ時に変更可能であ
2013 るかどうかを決定する。

2014 以下に、Property 要素の使用例を示す。

```

2015 <ProductSection>
2016   <Info>Describes product information for the service</Info>
2017   <Product>MyService Web Portal</Product>
2018   <Vendor>Some Random Organization</Vendor>
2019   <Version>4.5</Version>
2020   <FullVersion>4.5-b4523</FullVersion>
2021   <ProductUrl>http://www.vmware.com/go/ovf</ProductUrl>
2022   <VendorUrl>http://www.vmware.com/</VendorUrl>
2023   <Property ovf:key="adminEmail" ovf:type="string"
2024     ovf:userConfigurable="true">
2025     <Description>Email address of administrator</Description>
2026   </Property>
2027   <Property ovf:key="appIp" ovf:type="string"
2028     ovf:userConfigurable="true">
2029     <Description>IP address of the application</Description>
2030   </Property>
2031   <Property ovf:key="Gateway" ovf:type="string" ovf:value="192.168.0.1"
2032     ovf:userConfigurable="false" >
2033     <Description>Gateway address to be used</Description>
2034   </Property>
2035   <Property ovf:key=" SoftwareResourceType" ovf:type="uint16"
2036     ovf:qualifiers="uint16,uint16,uint16,uint16,uint16,uint16,uint16,uint16,uint16,
2037     uint16,"
2038     ovf:value="Unknown", "Buffer", "Queue", "Protocol Endpoint", "Remote Interface",
2039     "Pool", "Cache", "File", "Database",
2040     ovf:userConfigurable="false" >
2041     <Description>Value Map example based on SoftwareResourceType property in
2042     CIM_SoftwareResource class</Description>
2043   </Property>
2044 </ProductSection>

```

2045 CIM_SoftwareResource クラスには、以下のCIMスキーマのスニペットに示されているプロパティ
 2046 SoftwareResourceTypeがある。

```

2047   [Description (
2048     "The type of the software resource. Although the behavior "
2049     "of the different software resource types is modeled "
2050     "similarly, different names for resources transferring "
2051     "data over time or/and space have been established. "
2052     "SoftwareResourceType conveys their original, most common "
2053     "name. ¥n"
2054     ValueMap { "0", "2", "3", "4", "5", "6", "7", "8", "9",
2055     "10..32767", "32768..65535" },
2056     Values { "Unknown", "Buffer", "Queue", "Protocol Endpoint",
2057     "Remote Interface", "Pool", "Cache", "File", "Database",

```

```
2058     "DMTF Reserved", "Vendor Reserved" ]]
```

```
2059     uint16 SoftwareResourceType;
```

2060 ovf:type は、ValueMap のインデックスに対応するので、uint16 である。

2061 ANNEX D に、複雑な多層アプリケーションの詳細なカスタマイズ例を示す。

2062 7 可搬性

2063 OVF は、仮想アプライアンスとそれに関連する仮想マシンの可搬性強化を実現するテクノロジーである。OVF
2064 パッケージは、仮想化プラットフォームで簡潔に解釈可能な仮想マシンを作成するためのレシピを収めている。
2065 パッケージ化したメタデータにより、確実にわかりやすい方法で仮想アプライアンスをインストールできる。特に、
2066 管理インフラストラクチャでメタデータを使用すると、OVF に記述された特定の VM をインストールするか拒否する
2067 かを確実に判断できる。また、インストール先特有の実行コンテキストで使用できるように、適切な変換やローカ
2068 ライズを案内することもできる。

2069 パッケージ化した仮想マシンの可搬性を左右する要因には、OVF フォーマット仕様で取り上げていないものが数
2070 多く存在する。OVF フォーマット仕様に全面的に準拠している実装でも、これは同様である。つまり、OVF パッ
2071 ケージにパッケージ化した仮想マシンであっても、必ずしもすべてのハイパーバイザーにわたって可搬性が得ら
2072 れるわけではなく、インストール可能であるわけでもない。以下に、可搬性を制限する可能性のある要因の例を
2073 挙げる。

- 2074 • OVF 内の VM がインストールを試みるハイパーバイザーでは理解できないフォーマットの仮想ディスク
2075 を含むことが考えられる。主要な仮想ハード・ディスク・フォーマットによる VM のインポートおよびエク
2076 スポートが大半のハイパーバイザーで可能であると想定することは妥当ではあるが、OVF がサポートし
2077 ても、特定のハイパーバイザーがサポートしていない新しいフォーマットが登場する可能性があ
2078 る。
- 2079 • ハイパーバイザーが提示した仮想ハードウェアを、インストールされたゲスト・ソフトウェアがサポートし
2080 ていない可能性がある。たとえば、Xen ハイパーバイザーはデフォルトでは仮想化したフロッピー・ディ
2081 スク・デバイスをゲストに提供しない。フロッピー・ディスク・コントローラーとの対話を必要とするゲスト
2082 VM が思い浮かぶが、これでは VM を正しく実行できない。
- 2083 • インストールされたゲスト・ソフトウェアがサポートしていない CPU アーキテクチャである。たとえば、ゲ
2084 スト・ソフトウェアには、特定のプロセッサ・モデル固有の CPU 操作を実行する可能性、特別な浮動
2085 小数点サポートを必要とする可能性、特定ベンダーの CPU の命令コードを使用している可能性などが
2086 ある。
- 2087 • OVF デスクリプタで要求している機能を仮想化プラットフォームが理解できない可能性がある。たと
2088 えば、作成したサービスがサポートされていない場合がある。OVF 標準は仮想化製品とは関係なく進化
2089 していることから、対象とする OVF 仕様より以前から存在する仮想化プラットフォームでは、いずれか
2090 の時点で OVF をサポートできなくなる可能性がある。

2091 OVF の可搬性は、以下のクラスに分類される。

- 2092 • **可搬性クラス 1**。複数の仮想ハードウェア・ファミリーで稼働する。たとえば、アプライアンスは Xen、
2093 Sun、Microsoft、VMware の各ハイパーバイザーで使用できる。レベル 3 の互換性を確保する目的で、
2094 複数のハイパーバイザーに対応するデバイスをサポートするゲスト・ソフトウェアが開発されている。ゲ
2095 スト・ソフトウェアのクリーン・インストールとブートは OVF のレベル 3 可搬性の例である。このインストー

- 2096 ルとブートの間、ゲスト・ソフトウェアはハードウェア・デバイスを発見し、仮想化プラットフォームとの対
2097 話に必要な専用ドライバーをすべてインストールする。また、Microsoft Windows®オペレーティング・
2098 システムの"sysprep"レベルの可搬性も1つの例である。このようなゲスト・ソフトウェア・インスタンスは、
2099 仮想ハードウェアなどの複数のハードウェア・プラットフォームに再インストールし、名前を変更して、
2100 パーソナライズしなおすことができる。
- 2101 • **可搬性クラス 2。**特定の仮想ハードウェア・ファミリーで稼働する。これは通常、インストール済みのゲスト
2102 ト・ソフトウェアでサポートされているドライバーが不足しているためである。
 - 2103 • **可搬性クラス 3。**特定の仮想化製品、CPU アーキテクチャ、仮想ハードウェアでのみ稼働する。これは
2104 通常、一時停止中の仮想マシン、または稼働状態にある仮想マシンのスナップショット(CPU および実
2105 デバイスまたはエミュレートしたデバイスの最新の実行時状態など)が OVF に存在するためである。こ
2106 のような状態により、OVF は特別な仮想化とハードウェアのプラットフォームに関連付けられる。
- 2107 組織内での使用には、クラス 2 またはクラス 3 の互換性で十分である。これはハードウェアや仮想化プラット
2108 フォームの購入判断が明確に下され、OVF の基盤機能セットの一貫性が保証された管理環境に OVF パッケージ
2109 が配布されるからである。仮想マシンを単純にエクスポートすると、クラス 3 または 2 の可搬性を持つ OVF が作
2110 成される(特定の仮想ハードウェア・セットに関連付けられた可搬性)。しかし、この比喩を拡大すれば、Windows
2111 の"sysprep"などのユーティリティを使用して、クラス 1 の可搬性のエクスポートをサポートすることも簡単にできる。
- 2112 ISV が独自に開発し、配布している業務用アプライアンスについては、クラス 1 の可搬性が望ましい。実際、クラス
2113 1 の可搬性により、評価と実稼働の両面できわめて広範囲の顧客がアプライアンスをすぐに利用できるようになる。
2114 ツールキットを使用して、認証された「既知の優良な」クラス 1 のアプライアンス・パッケージを作成し、複数の仮想
2115 プラットフォームに広く配布し、インストールできる。また、企業の特定の開発グループのように限定した仮想ハー
2116 ドウェアでアプライアンスを利用する場合は、クラス 2 の可搬性パッケージを作成することも考えられる。
- 2117 OVF の仮想ハードウェアの記述は、クラス 1 から 3 までの可搬性をサポートするように設計されている。クラス 1
2118 の可搬性では、ごく一般的なハードウェア要件のみを記述できるが、複数の代替仮想ハードウェアの記述を規定
2119 することもできる。仮想ハードウェア仕様をどの程度まで柔軟にするか、または制限するかという管理は、アプライ
2120 アンス・プロバイダーに一任される。厳密な仕様を使用すると、既知の優良な仮想ハードウェアでのみアプライ
2121 アンスを実行するように制限を適用できるが、可搬性はある程度制限される。制限を緩和した仕様では、可能な限
2122 り広い範囲にわたる仮想ハードウェアでアプライアンスを使用できる。これにより、顧客は、望み得る最高のユー
2123 ザー体験を得ることができる。これは仮想アプライアンス概念の成功に欠かせない主要な条件の 1 つである。

2124
2125
2126
2127

ANNEX A (参考情報)

単一仮想システムの例

2128 デスクリプタの大半は定型的である。デスクリプタは、ファイル・セットとデスクリプタ自体の記述で始まる。この場
2129 合、存在するファイルは 1 つのみである(vmdisk1.vmdk)。続いて、アプライアンスが使用する仮想ディスクの
2130 セットとネットワーク・のセットを記述する。ファイル、ディスク、ネットワーク・リソースにはそれぞれ固有の ID を割
2131 り当てる。これらはすべて、互いに別々の名前空間にあるが、異なる名前を使用することが最良である。

2132 サンプルの OVF にあるコンテンツは、単一の仮想マシンである。このコンテンツには 5 つのセクションがある。

2133 典型的な単一仮想マシン・アプライアンスの OVF デスクリプタ全体を以下に示す。

```
2134 <?xml version="1.0" encoding="UTF-8"?>
2135 <Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2136     xmlns="http://schemas.dmtf.org/ovf/1/envelope"
2137     xmlns:ovf="http://schemas.dmtf.org/ovf/1/envelope"
2138     xmlns:vssd="http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_VirtualSystemSetting
2139     Data"
2140     xmlns:rasd="http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_ResourceAllocationSe
2141     ttingData">
2142     <!-- References to all external files -->
2143     <References>
2144         <File ovf:id="file1" ovf:href="vmdisk1.vmdk" ovf:size="180114671"/>
2145     </References>
2146     <!-- Describes meta-information for all virtual disks in the package -->
2147     <DiskSection>
2148         <Info>Describes the set of virtual disks</Info>
2149         <Disk ovf:diskId="vmdisk1" ovf:fileRef="file1" ovf:capacity="4294967296"
2150         ovf:format="http://www.vmware.com/interfaces/specifications/vmdk.html#sparse"/>
2151     </DiskSection>
2152     <!-- Describes all networks used in the package -->
2153     <NetworkSection>
2154         <Info>List of logical networks used in the package</Info>
2155         <Network ovf:name="VM Network">
2156             <Description>The network that the services are available on</Description>
2157         </Network>
2158     </NetworkSection>
2159     <VirtualSystem ovf:id="vm">
2160         <Info>Describes a virtual machine</Info>
2161         <Name>Virtual Appliance One</Name>
```



```
2166     <ProductSection>
2167         <Info>Describes product information for the appliance</Info>
2168         <Product>The Great Appliance</Product>
2169         <Vendor>Some Great Corporation</Vendor>
2170         <Version>13.00</Version>
2171         <FullVersion>13.00-b5</FullVersion>
2172
2173 <ProductUrl>http://www.somegreatcorporation.com/greatappliance</ProductUrl>
2174     <VendorUrl>http://www.somegreatcorporation.com/</VendorUrl>
2175     <Property ovf:key="adminEmail" ovf:type="string">
2176         <Description>Email address of administrator</Description>
2177     </Property>
2178     <Property ovf:key="appIp" ovf:type="string" ovf:defaultValue="192.168.0.10">
2179         <Description>The IP address of this appliance</Description>
2180     </Property>
2181 </ProductSection>
2182 <AnnotationSection ovf:required="false">
2183     <Info>A random annotation on this service. It can be ignored</Info>
2184     <Annotation>Contact customer support if you have any problems</Annotation>
2185 </AnnotationSection>
2186 <EulaSection>
2187     <Info>License information for the appliance</Info>
2188     <License>Insert your favorite license here</License>
2189 </EulaSection>
2190 <VirtualHardwareSection>
2191     <Info>256MB, 1 CPU, 1 disk, 1 nic</Info>
2192     <Item>
2193         <rasd:Description>Number of virtual CPUs</rasd:Description>
2194         <rasd:ElementName>1 virtual CPU</rasd:ElementName>
2195         <rasd:InstanceID>1</rasd:InstanceID>
2196         <rasd:ResourceType>3</rasd:ResourceType>
2197         <rasd:VirtualQuantity>1</rasd:VirtualQuantity>
2198     </Item>
2199     <Item>
2200         <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
2201         <rasd:Description>Memory Size</rasd:Description>
2202         <rasd:ElementName>256 MB of memory</rasd:ElementName>
2203         <rasd:InstanceID>2</rasd:InstanceID>
2204         <rasd:ResourceType>4</rasd:ResourceType>
2205         <rasd:VirtualQuantity>256</rasd:VirtualQuantity>
2206     </Item>
2207     <Item>
2208         <rasd:AutomaticAllocation>true</rasd:AutomaticAllocation>
2209         <rasd:Connection>VM Network</rasd:Connection>
2210         <rasd:ElementName>Ethernet adapter on "VM Network"</rasd:ElementName>
```

```
2211         <rasd:InstanceID>4000</rasd:InstanceID>
2212         <rasd:ResourceType>10</rasd:ResourceType>
2213     </Item>
2214     <Item>
2215         <rasd:ElementName>Harddisk 1</rasd:ElementName>
2216         <rasd:HostResource>ovf:/disk/vmdisk1</rasd:HostResource>
2217         <rasd:InstanceID>22001</rasd:InstanceID>
2218         <rasd:ResourceType>17</rasd:ResourceType>
2219     </Item>
2220 </VirtualHardwareSection>
2221 <OperatingSystemSection ovf:id="58" ovf:required="false">
2222     <Info>Guest Operating System</Info>
2223     <Description>Windows 2000 Advanced Server</Description>
2224 </OperatingSystemSection>
2225 </VirtualSystem>
2226 </Envelope>
2227
```

2228
2229
2230
2231

ANNEX B (参考情報)

多層 Pet Store の例

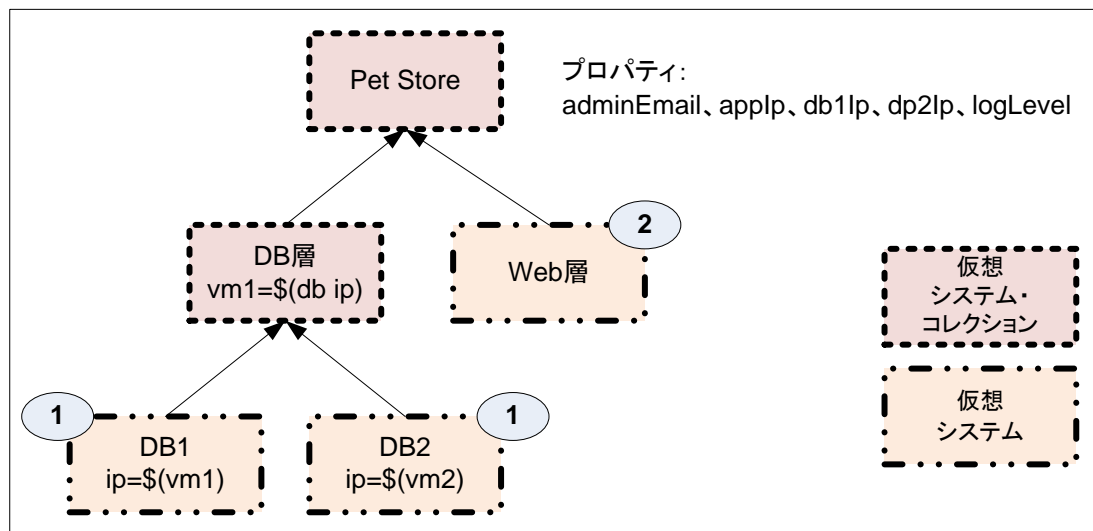
2232 この Pet Store OVF デスクリプタは、以下のような高度な OVF 概念を示している。

- 2233 • 複数 VM のパッケージ - VirtualMachineCollection エンティティ・サブタイプの使用
- 2234 • 複合サービス組織 - ネストした VirtualMachineCollection エンティティ・サブタイプの使用
- 2235 • ユーザー定義したデプロイ構成の伝搬
- 2236 • OVF 環境を使用してデプロイ時にサービスに対して適用するカスタマイズ
- 2237 • ダウンロードを最小限に抑えるための仮想ディスク・チェーンの使用
- 2238 • 個々の仮想マシンにインストールされているソフトウェアに関する情報を提供するための
2239 ProductSection 要素のネスト化

2240 このサービス例は Pet Store と呼ばれ、フロントエンド Web サーバーとデータベースで構成している。データベー
2241 ス・サーバー自体が、フォールトトレランスを目的として 2 つの VM で構成した複雑な多層サーバーである。

2242 B.1 アーキテクチャとパッケージ化

2243 Pet Store OVF パッケージは 3 つの仮想システム (WebTier、DB1、DB2) と 2 つの仮想システムのコレクション
2244 (Pet Store と DBTier) で構成される。図 B-1 に、この OVF パッケージの構造および仮想マシンのプロパティと起
2245 動順序を示す。



2246

2247

図 B-1 - Pet Store OVF パッケージ

2248 この OVF デスクリプタの全体は本書巻末に掲載されている。OVF のプロパティとディスク・レイアウトの使用法
2249 について、以下で詳しく説明する。

2250 **B.2 プロパティ**

2251 Pet Store サービスには、ユーザー構成可能なプロパティが 5 種類ある。これらはサービスにとって主要な制御パ
 2252 ラメーターで、デプロイ環境でサービスが正常に起動するように構成する必要がある。これらのプロパティは、
 2253 OVF 環境文書の形で、ゲスト・ソフトウェアに渡される。ゲスト・ソフトウェアは、起動時に OVF 環境を読み込み、
 2254 プロパティの値を抽出して、ソフトウェア構成に適用するように作成されている。したがって、OVF デスクリプタは、
 2255 ゲスト・ソフトウェアが扱うプロパティを反映している。

2256 このサービスには、Web 層とデータベース層に 1 つずつ、合計 2 種類のソフトウェア構成がある。各ソフトウェア
 2257 構成でサポートしているプロパティを以下の表に示す。

2258 表 B-1 は、Web ゲスト・ソフトウェアのプロパティをまとめたものである。

2259 **表 B-1 - Web 層の構成**

プロパティ	説明
<i>applp</i>	Web サーバーの IP アドレス
<i>dblp</i>	接続先データベース・サーバーの IP アドレス
<i>adminEmail</i>	サポートの電子メールアドレス
<i>logLevel</i>	ロギングのレベル

2260

2261 子 `VirtualSystem` または `VirtualSystemCollection` は、直接の親 `VirtualSystemCollection` コン
 2262 テナで定義されているすべてのプロパティを使用できる。したがって、WebVM の例にあるとおり、OVF デスクリプ
 2263 タでは、VM ごとに `ProductSection` を明示的に記述する必要はない。

2264 表 B-2 は、データベース・ゲスト・ソフトウェアのプロパティをまとめたものである。

2265 **表 B-2 - データベース層の構成**

プロパティ	説明
<i>ip</i>	仮想マシンの IP アドレス
<i>primaryAtBoot</i>	ブート時にインスタンスがプライマリまたはセカンダリのどちらとして動作するかを指定
<i>ip2</i>	ホットスペアまたはプライマリとして動作する、対のデータベース VM の IP アドレス
<i>log</i>	ロギングのレベル(ここでは log としている)

2266 クラスタ化したデータベースは、仮想システムのコレクションとして編成し、`vm1`、`vm2`、`log` の各プロパティを使
 2267 用して構成する。この組織では、データベースの実装を OVF パッケージの他のソフトウェアから分離し、仮想アプ
 2268 ライアンス(ゲスト・ソフトウェア + 仮想マシン構成)を簡単に構成できるようにして、再利用を促進している。

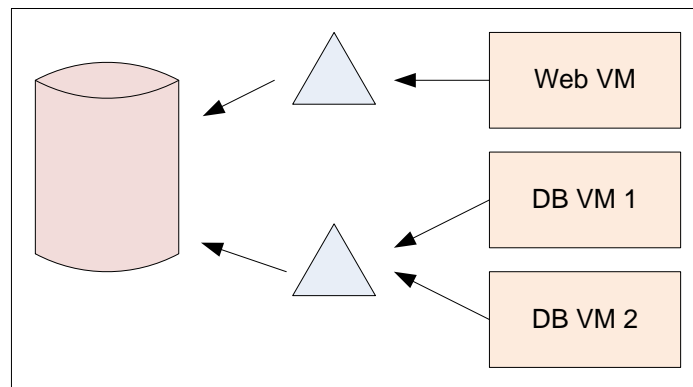
2269 データベース・ソフトウェアは市販のソフトウェア・パッケージであり、ベンダーはすべてのプロパティで使用する固
 2270 有のデータベース名として"`com.mydb.db`"を選択している。OVF デスクリプタでは、`ProductSection` で
 2271 `ovf:class` 属性の値としてこの文字列を使用している。

2272 OVF デスクリプタのエンティティ階層の中で、値を上位レベルから下位ノードに伝播するために $\{<name>\}$ プロパティ構文を使用している。このメカニズムでは、事前に命名規則の調整やゲスト・ソフトウェアの変更を必要とせず
 2273 に、さまざまなコンポーネントをリンクできる。子エンティティが使用できるのは、直接の親
 2274 `VirtualSystemCollection` コンテナで定義されているプロパティのみである。したがって、Petstore 仮想シ
 2275 ステムで定義したプロパティは、DB1 仮想システムでは使用できない。これにより、
 2276 `VirtualSystemCollection` のインターフェースはカプセル化され、親の `VirtualSystemCollection` で
 2277 適切に記述されることが保証され、ソフトウェアが構成可能になり、再利用しやすくなる。
 2278

2279 OVF デスクリプタは、ユーザー割り当てができない固定プロパティを使用して、2 台のデータベース仮想マシンが
 2280 互いに異なる役割でブートすることを保証している。これらの仮想マシンが、元は同一のソフトウェア・イメージか
 2281 らブートしたものであっても同様である。プロパティ `com.mydb.db.primaryAtBoot` には、ユーザー設定できない固
 2282 定値が指定されているが、その値は 2 つのイメージで互いに異なっている。ソフトウェアはブート時にこの値を調
 2283 べ、それに従って動作をカスタマイズする。

2284 B.3 ディスクのレイアウト

2285 Pet Store OVF パッケージは、ディスクの共有およびディスク階層の差分エンコードの機能を使用して、サイズを
 2286 最小化し、パッケージのダウンロード時間を最小限に抑えるようにしている。この例では、イメージが 2 種類のみ
 2287 (Database と Web) なので、これらが同じ基本 OS 上で構築されていると仮定すれば、この構成を図 B-2 に示す
 2288 OVF デスクリプタとしてエンコードできる。



2289

2290

図 B-2 - Pet Store 仮想ディスクのレイアウト

2291 したがって、このパッケージは 3 種類の仮想マシンを収めているが、合計ダウンロード・サイズはきわめて少ない。
 2292 実際、完全な VM が 1 つあれば、比較的小さな 2 つの差分だけをダウンロードすればすむ。

2293 デプロイ・システムでの仮想ディスクの物理的レイアウトは、OVF パッケージでのディスク構造には関係しない。
 2294 OVF パッケージは、仮想ディスクのサイズとコンテンツ(ディスクに配置する必要があるデータ)を記述する。また、
 2295 各仮想マシンが独立したディスクを取得するかどうかも指定する。したがって、仮想化プラットフォームでは、前述
 2296 のパッケージを 3 つの独立したフラット・ディスクを持つ 3 つの VM としてインストールできる。また、初期ブートに
 2297 記述されたコンテンツを収めたディスクを各仮想マシンが認識しており、ある仮想マシンで記述した変更が他の仮
 2298 想マシンに影響しないのであれば、前述の編成や類似する別の内容を複製する方法も考えられる。

2299 **B.4 Pet Store OVF デスクリプタ**

```
2300 <?xml version="1.0" encoding="UTF-8"?>
2301 <Envelope
2302     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2303     xmlns="http://schemas.dmtf.org/ovf/envelope/1"
2304     xmlns:ovf="http://schemas.dmtf.org/ovf/envelope/1"
2305
2306     xmlns:vssd="http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_VirtualSystemSetting
2307     Data"
2308
2309     xmlns:rasd="http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_ResourceAllocationSe
2310     ttingData">
2311     <!-- References to all external files -->
2312     <References>
2313         <File ovf:id="base" ovf:href="base.vmdk" ovf:size="180114671"/>
2314         <File ovf:id="webdelta" ovf:href="webapp-delta.vmdk" ovf:size="123413"/>
2315         <File ovf:id="dbdelta" ovf:href="dbapp-delta.vmdk" ovf:size="343243"/>
2316     </References>
2317     <!-- Describes meta-information about all virtual disks in the package.
2318     This example is encoded as a delta-disk hierarchy.
2319     -->
2320     <DiskSection>
2321         <Info>Describes the set of virtual disks</Info>
2322         <Disk ovf:diskId="base" ovf:fileRef="base" ovf:capacity="4294967296"
2323             ovf:populatedSize="1924967692"
2324
2325             ovf:format="http://www.vmware.com/specifications/vmdk.html#streamOptimized"/>
2326         <Disk ovf:diskId="web" ovf:fileRef="webappdelta" ovf:parentRef="base"
2327             ovf:capacity="4294967296"
2328
2329             ovf:format="http://www.vmware.com/specifications/vmdk.html#streamOptimized"/>
2330         <Disk ovf:diskId="db" ovf:fileRef="dbdelta" ovf:parentRef="base"
2331             ovf:capacity="4294967296"
2332
2333             ovf:format="http://www.vmware.com/specifications/vmdk.html#streamOptimized"/>
2334     </DiskSection>
2335     <!-- Describes all networks used in the package -->
2336     <NetworkSection>
2337         <Info>List of logical networks used in the package</Info>
2338         <Network ovf:name="VM Network">
2339             <Description ovf:msgid="network.description">The network that the services
2340             are available on</Description>
2341         </Network>
2342     </NetworkSection>
2343     <!-- Deployment options for the packages -->
2344     <DeploymentOptionSection>
2345         <Info>List of deployment options available in the package</Info>
2346         <Configuration ovf:id="minimal">
2347             <Label ovf:msgid="minimal.label">Minimal</Label>
2348             <Description ovf:msgid="minimal.description">Deploy service with minimal
2349             resource use</Description>
2350         </Configuration>
2351         <Configuration ovf:id="standard" ovf:default="true">
2352             <Label ovf:msgid="standard.label">Standard</Label>
```

```
2353     <Description ovf:msgid="standard.description">Deploy service with standard
2354         resource use</Description>
2355     </Configuration>
2356 </DeploymentOptionSection>
2357 <!-- PetStore Virtual System Collection -->
2358 <VirtualSystemCollection ovf:id="PetStore">
2359     <Info>The packaging of the PetStoreService multitier application</Info>
2360     <Name>PetStore Service</Name>
2361     <!-- Overall information about the product -->
2362     <ProductSection>
2363         <Info>Describes product information for the service</Info>
2364         <Product>PetStore Web Portal</Product>
2365         <Vendor>Some Random Organization</Vendor>
2366         <Version>4.5</Version>
2367         <FullVersion>4.5-b4523</FullVersion>
2368         <ProductUrl>http://www.vmware.com/go/ovf</ProductUrl>
2369         <VendorUrl>http://www.vmware.com/</VendorUrl>
2370         <Category ovf:msgid="category.email">Email properties</Category>
2371         <Property ovf:key="adminEmail" ovf:type="string"
2372 ovf:userConfigurable="true">
2373             <Label ovf:msgid="property.email.label">Admin email</Label>
2374             <Description ovf:msgid="property.email.description">Email address of
2375                 service administrator</Description>
2376         </Property>
2377         <Category ovf:msgid="category.network">Network properties</Category>
2378         <Property ovf:key="appIp" ovf:type="string"
2379 ovf:userConfigurable="true">
2380             <Label ovf:msgid="property.appIp.label">IP</Label>
2381             <Description ovf:msgid="property.appIp.description">IP address of the
2382                 service</Description>
2383         </Property>
2384         <Property ovf:key="dbIp" ovf:type="string" ovf:userConfigurable="true">
2385             <Label ovf:msgid="property.dpip.label">IP for DB</Label>
2386             <Description ovf:msgid="property.dpip.description">Primary IP address of
2387                 the database</Description>
2388         </Property>
2389         <Property ovf:key="db2Ip" ovf:type="string"
2390 ovf:userConfigurable="true">
2391             <Label ovf:msgid="property.dpip2.label">IP for DB2</Label>
2392             <Description ovf:msgid="property.dpip2.description">A secondary IP
2393                 address for the database</Description>
2394         </Property>
2395         <Category ovf:msgid="category.logging">Logging properties</Category>
2396         <Property ovf:key="logLevel" ovf:type="string" ovf:value="normal"
2397 ovf:userConfigurable="true">
2398             <Label ovf:msgid="property.loglevel.label">Loglevel</Label>
2399             <Description ovf:msgid="property.loglevel.description">Logging level for
2400                 the service</Description>
2401             <Value ovf:value="low" ovf:configuration="minimal"/>
2402         </Property>
2403     </ProductSection>
2404     <AnnotationSection ovf:required="false">
2405         <Info>A annotation on this service</Info>
2406         <Annotation ovf:msgid="annotation.annotation">Contact customer support for
2407             any urgent issues</Annotation>
```

```
2408 </AnnotationSection>
2409 <ResourceAllocationSection ovf:required="false">
2410   <Info>Defines minimum reservations for CPU and memory</Info>
2411   <Item>
2412     <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
2413     <rasd:ElementName>512 MB reservation</rasd:ElementName>
2414     <rasd:InstanceID>0</rasd:InstanceID>
2415     <rasd:Reservation>512</rasd:Reservation>
2416     <rasd:ResourceType>4</rasd:ResourceType>
2417   </Item>
2418   <Item ovf:configuration="minimal">
2419     <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
2420     <rasd:ElementName>384 MB reservation</rasd:ElementName>
2421     <rasd:InstanceID>0</rasd:InstanceID>
2422     <rasd:Reservation>384</rasd:Reservation>
2423     <rasd:ResourceType>4</rasd:ResourceType>
2424   </Item>
2425   <Item>
2426     <rasd:AllocationUnits>MHz</rasd:AllocationUnits>
2427     <rasd:ElementName>1000 MHz reservation</rasd:ElementName>
2428     <rasd:InstanceID>1</rasd:InstanceID>
2429     <rasd:Reservation>500</rasd:Reservation>
2430     <rasd:ResourceType>3</rasd:ResourceType>
2431   </Item>
2432   <Item ovf:bound="min">
2433     <rasd:AllocationUnits>MHz</rasd:AllocationUnits>
2434     <rasd:ElementName>500 MHz reservation</rasd:ElementName>
2435     <rasd:InstanceID>1</rasd:InstanceID>
2436     <rasd:Reservation>500</rasd:Reservation>
2437     <rasd:ResourceType>3</rasd:ResourceType>
2438   </Item>
2439   <Item ovf:bound="max">
2440     <rasd:AllocationUnits>MHz</rasd:AllocationUnits>
2441     <rasd:ElementName>1500 MHz reservation</rasd:ElementName>
2442     <rasd:InstanceID>1</rasd:InstanceID>
2443     <rasd:Reservation>1500</rasd:Reservation>
2444     <rasd:ResourceType>3</rasd:ResourceType>
2445   </Item>
2446 </ResourceAllocationSection>
2447 <StartupSection>
2448   <Info>Specifies how the composite service is powered-on and off</Info>
2449   <Item ovf:id="DBTier" ovf:order="1" ovf:startDelay="120"
2450     ovf:startAction="powerOn" ovf:waitingForGuest="true"
2451     ovf:stopDelay="120"
2452     ovf:stopAction="guestShutdown"/>
2453   <Item ovf:id="WebTier" ovf:order="2" ovf:startDelay="120"
2454     ovf:startAction="powerOn" ovf:waitingForGuest="true"
2455     ovf:stopDelay="120"
2456     ovf:stopAction="guestShutdown"/>
2457 </StartupSection>
2458 <VirtualSystem ovf:id="WebTier">
2459   <Info>The virtual machine containing the WebServer application</Info>
2460   <ProductSection>
2461     <Info>Describes the product information</Info>
2462     <Product>Apache Webserver</Product>
```



```

2463     <Vendor>Apache Software Foundation</Vendor>
2464     <Version>6.5</Version>
2465     <FullVersion>6.5-b2432</FullVersion>
2466 </ProductSection>
2467 <OperatingSystemSection ovf:id="97">
2468     <Info>Guest Operating System</Info>
2469     <Description>Linux 2.4.x</Description>
2470 </OperatingSystemSection>
2471 <VirtualHardwareSection>
2472     <Info>256 MB, 1 CPU, 1 disk, 1 nic virtual machine</Info>
2473     <System>
2474         <vssd:ElementName>Virtual Hardware Family</vssd:ElementName>
2475         <vssd:InstanceID>0</vssd:InstanceID>
2476         <vssd:VirtualSystemType>vmx-04</vssd:VirtualSystemType>
2477     </System>
2478     <Item>
2479         <rasd:Description>Number of virtual CPUs</rasd:Description>
2480         <rasd:ElementName>1 virtual CPU</rasd:ElementName>
2481         <rasd:InstanceID>1</rasd:InstanceID>
2482         <rasd:ResourceType>3</rasd:ResourceType>
2483         <rasd:VirtualQuantity>1</rasd:VirtualQuantity>
2484     </Item>
2485     <Item>
2486         <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
2487         <rasd:Description>Memory Size</rasd:Description>
2488         <rasd:ElementName>256 MB of memory</rasd:ElementName>
2489         <rasd:InstanceID>2</rasd:InstanceID>
2490         <rasd:ResourceType>4</rasd:ResourceType>
2491         <rasd:VirtualQuantity>256</rasd:VirtualQuantity>
2492     </Item>
2493     <Item>
2494         <rasd:AutomaticAllocation>true</rasd:AutomaticAllocation>
2495         <rasd:Connection>VM Network</rasd:Connection>
2496         <rasd:ElementName>Ethernet adapter on "VM
2497 Network"</rasd:ElementName>
2498         <rasd:InstanceID>3</rasd:InstanceID>
2499         <rasd:ResourceSubType>PCNet32</rasd:ResourceSubType>
2500         <rasd:ResourceType>10</rasd:ResourceType>
2501     </Item>
2502     <Item>
2503         <rasd:AddressOnParent>1</rasd:AddressOnParent>
2504         <rasd:ElementName>SCSI Controller 0 - LSI Logic</rasd:ElementName>
2505         <rasd:InstanceID>1000</rasd:InstanceID>
2506         <rasd:ResourceSubType>LsiLogic</rasd:ResourceSubType>
2507         <rasd:ResourceType>6</rasd:ResourceType>
2508     </Item>
2509     <Item>
2510         <rasd:AddressOnParent>0</rasd:AddressOnParent>
2511         <rasd:ElementName>Harddisk 1</rasd:ElementName>
2512         <rasd:HostResource>ovf:/disk/web</rasd:HostResource>
2513         <rasd:InstanceID>22001</rasd:InstanceID>
2514         <rasd:Parent>1000</rasd:Parent>
2515         <rasd:ResourceType>17</rasd:ResourceType>
2516     </Item>
2517 </VirtualHardwareSection>

```

```

2518 </VirtualSystem>
2519 <!-- Database Tier -->
2520 <VirtualSystemCollection ovf:id="DBTier">
2521   <Info>Describes a clustered database instance</Info>
2522   <ProductSection ovf:class="com.mydb.db">
2523     <Info>Product Information</Info>
2524     <Product>Somebody Clustered SQL Server</Product>
2525     <Vendor>TBD</Vendor>
2526     <Version>2.5</Version>
2527     <FullVersion>2.5-b1234</FullVersion>
2528     <Property ovf:key="vm1" ovf:value="{dbIp}" ovf:type="string"/>
2529     <Property ovf:key="vm2" ovf:value="{db2Ip}" ovf:type="string"/>
2530     <Property ovf:key="log" ovf:value="{logLevel}" ovf:type="string"/>
2531   </ProductSection>
2532   <StartupSection>
2533     <Info>Specifies how the composite service is powered-on and off</Info>
2534     <Item ovf:id="DB1" ovf:order="1" ovf:startDelay="120"
2535       ovf:startAction="powerOn" ovf:waitingForGuest="true"
2536       ovf:stopDelay="120" ovf:stopAction="guestShutdown"/>
2537     <Item ovf:id="DB2" ovf:order="2" ovf:startDelay="120"
2538       ovf:startAction="powerOn" ovf:waitingForGuest="true"
2539       ovf:stopDelay="120" ovf:stopAction="guestShutdown"/>
2540   </StartupSection>
2541   <!-- DB VM 1 -->
2542   <VirtualSystem ovf:id="DB1">
2543     <Info>Describes a virtual machine with the database image installed</Info>
2544     <Name>Database Instance I</Name>
2545     <ProductSection ovf:class="com.mydb.db">
2546       <Info>Specifies the OVF properties available in the OVF
2547 environment</Info>
2548       <Property ovf:key="ip" ovf:value="{vm1}" ovf:type="string"/>
2549       <Property ovf:key="ip2" ovf:value="{vm2}" ovf:type="string"/>
2550       <Property ovf:key="primaryAtBoot" ovf:value="yes"
2551 ovf:type="string"/>
2552     </ProductSection>
2553     <VirtualHardwareSection>
2554       <Info>256 MB, 1 CPU, 1 disk, 1 nic virtual machine</Info>
2555       <System>
2556         <vssd:ElementName>Virtual Hardware Family</vssd:ElementName>
2557         <vssd:InstanceID>0</vssd:InstanceID>
2558         <vssd:VirtualSystemType>vmx-04</vssd:VirtualSystemType>
2559       </System>
2560       <Item>
2561         <rasd:Description>Number of virtual CPUs</rasd:Description>
2562         <rasd:ElementName>1 virtual CPU</rasd:ElementName>
2563         <rasd:InstanceID>1</rasd:InstanceID>
2564         <rasd:ResourceType>3</rasd:ResourceType>
2565         <rasd:VirtualQuantity>1</rasd:VirtualQuantity>
2566       </Item>
2567       <Item>
2568         <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
2569         <rasd:Description>Memory Size</rasd:Description>
2570         <rasd:ElementName>256 MB of memory</rasd:ElementName>
2571         <rasd:InstanceID>2</rasd:InstanceID>
2572         <rasd:ResourceType>4</rasd:ResourceType>

```

```

2573         <rasd:VirtualQuantity>256</rasd:VirtualQuantity>
2574     </Item>
2575     <Item>
2576         <rasd:AutomaticAllocation>>true</rasd:AutomaticAllocation>
2577         <rasd:Connection>VM Network</rasd:Connection>
2578         <rasd:ElementName>Ethernet adapter on "VM
2579 Network"</rasd:ElementName>
2580         <rasd:InstanceID>3</rasd:InstanceID>
2581         <rasd:ResourceSubType>PCNet32</rasd:ResourceSubType>
2582         <rasd:ResourceType>10</rasd:ResourceType>
2583     </Item>
2584     <Item>
2585         <rasd:AddressOnParent>1</rasd:AddressOnParent>
2586         <rasd:ElementName>SCSI Controller 0 - LSI
2587 Logic</rasd:ElementName>
2588         <rasd:InstanceID>1000</rasd:InstanceID>
2589         <rasd:ResourceSubType>LsiLogic</rasd:ResourceSubType>
2590         <rasd:ResourceType>6</rasd:ResourceType>
2591     </Item>
2592     <Item>
2593         <rasd:AddressOnParent>0</rasd:AddressOnParent>
2594         <rasd:ElementName>Harddisk 1</rasd:ElementName>
2595         <rasd:HostResource>ovf:/disk/db</rasd:HostResource>
2596         <rasd:InstanceID>22001</rasd:InstanceID>
2597         <rasd:Parent>1000</rasd:Parent>
2598         <rasd:ResourceType>17</rasd:ResourceType>
2599     </Item>
2600 </VirtualHardwareSection>
2601 <OperatingSystemSection ovf:id="97">
2602     <Info>Guest Operating System</Info>
2603     <Description>Linux 2.4.x</Description>
2604 </OperatingSystemSection>
2605 </VirtualSystem>
2606 <!-- DB VM 2 -->
2607 <VirtualSystem ovf:id="DB2">
2608     <Info>Describes a virtual machine with the database image installed</Info>
2609     <Name>Database Instance II</Name>
2610     <ProductSection ovf:class="com.mydb.db">
2611         <Info>Specifies the OVF properties available in the OVF
2612 environment</Info>
2613         <Property ovf:key="ip" ovf:value="\${vm2}" ovf:type="string"/>
2614         <Property ovf:key="ip2" ovf:value="\${vm1} " ovf:type="string"/>
2615         <Property ovf:key="primaryAtBoot" ovf:value="no" ovf:type="string"/>
2616     </ProductSection>
2617     <VirtualHardwareSection>
2618         <Info>256 MB, 1 CPU, 1 disk, 1 nic virtual machine</Info>
2619         <System>
2620             <vssd:ElementName>Virtual Hardware Family</vssd:ElementName>
2621             <vssd:InstanceID>0</vssd:InstanceID>
2622             <vssd:VirtualSystemType>vmx-04</vssd:VirtualSystemType>
2623         </System>
2624         <Item>
2625             <rasd:Description>Number of virtual CPUs</rasd:Description>
2626             <rasd:ElementName>1 virtual CPU</rasd:ElementName>
2627             <rasd:InstanceID>1</rasd:InstanceID>

```

```

2628         <rasd:ResourceType>3</rasd:ResourceType>
2629         <rasd:VirtualQuantity>1</rasd:VirtualQuantity>
2630     </Item>
2631     <Item>
2632         <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
2633         <rasd:Description>Memory Size</rasd:Description>
2634         <rasd:ElementName>256 MB of memory</rasd:ElementName>
2635         <rasd:InstanceID>2</rasd:InstanceID>
2636         <rasd:ResourceType>4</rasd:ResourceType>
2637         <rasd:VirtualQuantity>256</rasd:VirtualQuantity>
2638     </Item>
2639     <Item>
2640         <rasd:AutomaticAllocation>true</rasd:AutomaticAllocation>
2641         <rasd:Connection>VM Network</rasd:Connection>
2642         <rasd:ElementName>Ethernet adapter on "VM
2643 Network"</rasd:ElementName>
2644         <rasd:InstanceID>3</rasd:InstanceID>
2645         <rasd:ResourceSubType>PCNet32</rasd:ResourceSubType>
2646         <rasd:ResourceType>10</rasd:ResourceType>
2647     </Item>
2648     <Item>
2649         <rasd:AddressOnParent>1</rasd:AddressOnParent>
2650         <rasd:ElementName>SCSI Controller 0 - LSI
2651 Logic</rasd:ElementName>
2652         <rasd:InstanceID>1000</rasd:InstanceID>
2653         <rasd:ResourceSubType>LsiLogic</rasd:ResourceSubType>
2654         <rasd:ResourceType>6</rasd:ResourceType>
2655     </Item>
2656     <Item>
2657         <rasd:AddressOnParent>0</rasd:AddressOnParent>
2658         <rasd:ElementName>Harddisk 1</rasd:ElementName>
2659         <rasd:HostResource>ovf:/disk/db</rasd:HostResource>
2660         <rasd:InstanceID>22001</rasd:InstanceID>
2661         <rasd:Parent>1000</rasd:Parent>
2662         <rasd:ResourceType>17</rasd:ResourceType>
2663     </Item>
2664 </VirtualHardwareSection>
2665 <OperatingSystemSection ovf:id="97">
2666     <Info>Guest Operating System</Info>
2667     <Description>Linux 2.4.x</Description>
2668 </OperatingSystemSection>
2669 </VirtualSystem>
2670 </VirtualSystemCollection>
2671 </VirtualSystemCollection>
2672 <!-- External I18N bundles -->
2673 <Strings xml:lang="de-DE" ovf:fileRef="de-DE-bundle.xml"/>
2674 <!-- Embedded I18N bundles -->
2675 <Strings xml:lang="da-DA">
2676     <Msg ovf:msgid="network.description">Netværket servicen skal være tilgængelig
2677 på</Msg>
2678     <Msg ovf:msgid="annotation.annotation">Kontakt kundeservice i tilfælde af
2679     kritiske problemer</Msg>
2680     <Msg ovf:msgid="property.email.description">Email adresse for
2681 administrator</Msg>
2682     <Msg ovf:msgid="property.appIp.description">IP adresse for service</Msg>

```

```

2683     <Msg ovf:msgid="property.dpIp">Primær IP adresse for database</Msg>
2684     <Msg ovf:msgid="property.dpIp2.description">Sekundær IP adresse for
2685 database</Msg>
2686     <Msg ovf:msgid="property.loglevel.description">Logningsniveau for service</Msg>
2687     <Msg ovf:msgid="minimal.label">Minimal</Msg>
2688     <Msg ovf:msgid="minimal.description">Installer service med minimal brug af
2689         resourcer</Msg>
2690     <Msg ovf:msgid="standard.label">Normal</Msg>
2691     <Msg ovf:msgid="standard.description">Installer service med normal brug af
2692         resourcer</Msg>
2693     </Strings>
2694 </Envelope>

```

2695 B.5 OVF 環境の全体

2696 以下に、WebTier 仮想マシンおよび DB1 仮想マシンから見た OVF 環境の例を示す (DB2 仮想マシンは DB1 と
2697 事実上同じなので省略する)。

2698 WebTier 仮想マシンの OVF 環境:

```

2699 <?xml version="1.0" encoding="UTF-8"?>
2700 <Environment
2701     xmlns="http://schemas.dmtf.org/ovf/environment/1"
2702     xmlns:ovfenv="http://schemas.dmtf.org/ovf/environment/1"
2703     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2704     ovfenv:id="WebTier">
2705
2706     <!-- Information about hypervisor platform -->
2707     <PlatformSection>
2708         <Kind>ESX Server</Kind>
2709         <Version>3.0.1</Version>
2710         <Vendor>VMware, Inc.</Vendor>
2711         <Locale>en_US</Locale>
2712     </PlatformSection>
2713
2714     <!-- Properties defined for this virtual machine -->
2715     <PropertySection>
2716         <Property ovfenv:key="adminEmail" ovfenv:value="ovf-admin@vmware.com"/>
2717         <Property ovfenv:key="appIp" ovfenv:value="10.20.132.101"/>
2718         <Property ovfenv:key="dbIp" ovfenv:value="10.20.132.102"/>
2719         <Property ovfenv:key="db2Ip" ovfenv:value="10.20.132.103"/>
2720         <Property ovfenv:key="logLevel" ovfenv:value="warning"/>
2721     </PropertySection>
2722
2723     <Entity ovfenv:id="DBTier">
2724         <PropertySection>
2725             <Property ovfenv:key="adminEmail" ovfenv:value="ovf-admin@vmware.com"/>
2726             <Property ovfenv:key="appIp" ovfenv:value="10.20.132.101"/>
2727             <Property ovfenv:key="dbIp" ovfenv:value="10.20.132.102"/>
2728             <Property ovfenv:key="db2Ip" ovfenv:value="10.20.132.103"/>
2729             <Property ovfenv:key="logLevel" ovfenv:value="warning"/>
2730             <Property ovfenv:key="com.mydb.db.vm1" ovfenv:value="10.20.132.102"/>
2731             <Property ovfenv:key="com.mydb.db.vm2" ovfenv:value="10.20.132.103"/>
2732             <Property ovfenv:key="com.mydb.db.log" ovfenv:value="warning"/>

```

```
2733     </PropertySection>
2734   </Entity>
2735 </Environment>
```

2736 DB1 仮想マシンの OVF 環境:

```
2737 <?xml version="1.0" encoding="UTF-8"?>
2738 <Environment
2739   xmlns="http://schemas.dmtf.org/ovf/environment/1"
2740   xmlns:ovfenv="http://schemas.dmtf.org/ovf/environment/1"
2741   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2742   ovfenv:id="DB1">
2743
2744   <!-- Information about hypervisor platform -->
2745   <PlatformSection>
2746     <Kind>ESX Server</Kind>
2747     <Version>3.0.1</Version>
2748     <Vendor>VMware, Inc.</Vendor>
2749     <Locale>en_US</Locale>
2750   </PlatformSection>
2751
2752   <!-- Properties defined for this virtual machine -->
2753   <PropertySection>
2754     <Property ovfenv:key="com.mydb.db.vm1" ovfenv:value="10.20.132.102"/>
2755     <Property ovfenv:key="com.mydb.db.vm2" ovfenv:value="10.20.132.103"/>
2756     <Property ovfenv:key="com.mydb.db.log" ovfenv:value="warning"/>
2757     <Property ovfenv:key="com.mydb.db.ip" ovfenv:value="10.20.132.102"/>
2758     <Property ovfenv:key="com.mydb.db.ip2" ovfenv:value="10.20.132.103"/>
2759     <Property ovfenv:key="com.mydb.db.primaryAtBoot" ovfenv:value="yes"/>
2760   </PropertySection>
2761
2762   <Entity ovfenv:id="DB2">
2763     <PropertySection>
2764       <Property ovfenv:key="com.mydb.db.vm1" ovfenv:value="10.20.132.102"/>
2765       <Property ovfenv:key="com.mydb.db.vm2" ovfenv:value="10.20.132.103"/>
2766       <Property ovfenv:key="com.mydb.db.log" ovfenv:value="warning"/>
2767       <Property ovfenv:key="com.mydb.db.ip" ovfenv:value="10.20.132.103"/>
2768       <Property ovfenv:key="com.mydb.db.ip2" ovfenv:value="10.20.132.102"/>
2769       <Property ovfenv:key="com.mydb.db.primaryAtBoot" ovfenv:value="no"/>
2770     </PropertySection>
2771   </Entity>
2772 </Environment>
2773
```

2774
2775
2776
2777

ANNEX C (参考情報)

単一仮想システムの LAMP スタックの例

2778 ここでは、LAMP 仮想アプライアンスの OVF デスクリプタがどのようなものであるかを2つの具体的な例で説明す
2779 る。単一 VM の LAMP 仮想アプライアンスと複数 VM の LAMP 仮想アプライアンスの両方の例を示す。LAMP は、
2780 Linux オペレーティング・システム、Apache Web サーバー、MySQL データベース、および PHP Web 開発ソフト
2781 ウェア・パッケージを使用して構築したサービスの略称である。

2782 この例では、`ProductSection` を使用して、オペレーティング・システムおよびアプリケーション・レベルのデブ
2783 ロイ・パラメーターの両方を指定する方法を示す。たとえば、このようなパラメーターを使用して、特定の環境にデ
2784 プロイするサービスのパフォーマンスを最適化できる。このデスクリプタは完成してはいるが、必要最小限のもの
2785 である。たとえば、EULA セクションは記述していない。

2786 C.1 デプロイ時のカスタマイズ

2787 OVF パッケージをデプロイする段階の作業として、カスタマイズ用パラメーターの指定がある。カスタマイズ用パ
2788 ラメーターは OVF デスクリプタで指定し、OVF 環境を使用してゲスト・ソフトウェアに提供する。このデプロイ時の
2789 カスタマイズは、仮想マシン・レベルのパラメーターを補足するものであり、仮想スイッチの接続や物理ストレージ
2790 の場所などを扱う。

2791 LAMP ベースの仮想アプライアンスでは、サービスの IP アドレスとポート番号、ゲートウェイやサブネットなどの
2792 ネットワーク情報、所定のデプロイでのパフォーマンスを最適化する各種パラメーターなどをデプロイ時のカスタ
2793 イズで扱う。デプロイ機能に公開するプロパティは、ベンダーやサービスごとに異なる。このデスクリプタの例では、
2794 表 C-1 に記述する以下のプロパティを4種類の LAMP コンポーネントで使用する。

2795

表 C-1 - LAMP 構成

製品	プロパティ	説明
Linux	<code>hostname</code>	アプリケーションのネットワーク ID (IP アドレスを含む)
	<code>ip</code>	
	<code>subnet</code>	
	<code>gateway</code>	
	<code>dns</code>	
Linux	<code>netCoreRmemMax</code>	IP スタックの転送速度を最適化するパラメーター
	<code>netCoreWmemMax</code>	

製品	プロパティ	説明
Apache	httpPort	Web サーバーのポート番号
	httpsPort	
	startThreads	Web サーバーのパフォーマンスを最適化するパラメーター
	minSpareThreads	
	maxSpareThreads	
	maxClients	
MySQL	queryCacheSize	データベース・サーバーのパフォーマンスを最適化するパラメーター
	maxConnections	
	waitTimeout	
PHP	sessionTimeout	PHP エンジンの動作をカスタマイズするパラメーター(セッションのタイムアウト方法、セッション数など)
	concurrentSessions	
	memoryLimit	

2796 イタリアック体のパラメーターはユーザーが指定しなければならない必須の構成である。それ以外は、妥当なデフォルト値が設定されているので、ユーザーが値を指定する必要はない。

2798 各ソフトウェア製品のカスタマイズ用パラメーターは、独立した製品セクションにカプセル化されている。たとえば、
2799 Apache Web サーバーについては、以下のセクションにまとめられている。

```

2800 <ProductSection ovf:class="org.apache.httpd">
2801   <Info>Product customization for the installed Apache Web Server</Info>
2802   <Product>Apache Distribution Y</Product>
2803   <Version>2.6.6</Version>
2804   <Property ovf:key="httpPort" ovf:type="uint16" ovf:value="80"
2805     ovf:userConfigurable="true">
2806     <Description>Port number for HTTP requests</Description>
2807   </Property>
2808   <Property ovf:key="httpsPort" ovf:type="uint16" ovf:value="443"
2809     ovf:userConfigurable="true">
2810     <Description>Port number for HTTPS requests</Description>
2811   </Property>
2812   <Property ovf:key="startThreads" ovf:type="uint16" ovf:value="50"
2813     ovf:userConfigurable="true">
2814     <Description>Number of threads created on startup. </Description>
2815   </Property>
2816   <Property ovf:key="minSpareThreads" ovf:type="uint16" ovf:value="15"
2817     ovf:userConfigurable="true">
2818     <Description> Minimum number of idle threads to handle request
2819 spikes.</Description>
2820   </Property>
2821   <Property ovf:key="maxSpareThreads" ovf:type="uint16" ovf:value="30"
2822     ovf:userConfigurable="true">
2823     <Description>Maximum number of idle threads </Description>
    
```



```
2824     </Property>
2825     <Property ovf:key="maxClients" ovf:type="uint16" ovf:value="256"
2826         ovf:userConfigurable="true">
2827         <Description>Limit the number of simultaneous requests that are served.
2828     </Description>
2829     </Property>
2830 </ProductSection>
```

2831 ovf:class="org.apache.httpd" 属性は、これらのプロパティのプレフィックスを表す。したがって、
2832 Apache データベースは、OVF 環境で以下のプロパティを見つけることになる。

```
2833 <Environment
2834     ...
2835     <!-- Properties defined for this virtual machine -->
2836     <PropertySection>
2837         <Property ovfenv:name="org.apache.httpd.httpPort" ovfenv:value="80"/>
2838         <Property ovfenv:name="org.apache.httpd.httpsPort" ovfenv:value="443"/>
2839         <Property ovfenv:name="org.apache.httpd.startThreads" ovfenv:value="50"/>
2840         <Property ovfenv:name="org.apache.httpd.minSpareThreads" ovfenv:value="15"/>
2841         <Property ovfenv:name="org.apache.httpd.maxSpareThreads" ovfenv:value="30"/>
2842         <Property ovfenv:name="org.apache.httpd.maxClients" ovfenv:value="256"/>
2843         ...
2844     </PropertySection>
2845     ...
2846 </Environment>
```

2847 **C.2 単純な LAMP OVF デスクリプタ**

2848 LAMP スタックを持つ単一 VM の仮想アプライアンスで使用する OVF デスクリプタ全体を以下に示す。

```
2849 <?xml version="1.0" encoding="UTF-8"?>
2850 <Envelope
2851     xmlns="http://schemas.dmtf.org/ovf/envelope/1"
2852     xmlns:ovf="http://schemas.dmtf.org/ovf/envelope/1"
2853     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2854     xmlns:vssd="http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_VirtualSystemSetting
2855     Data"
2856     xmlns:rasd="http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_ResourceAllocationSe
2857     ttingData"
2858     <!-- References to all external files -->
2859     <References>
2860         <File ovf:id="lamp" ovf:href="lamp.vmdk" ovf:size="180114671"/>
2861     </References>
2862     <!-- Describes meta-information about all virtual disks in the package. -->
2863     <DiskSection>
2864         <Info>List of the virtual disks used in the package</Info>
2865         <Disk ovf:diskId="lamp" ovf:fileRef="lamp" ovf:capacity="4294967296"
2866             ovf:populatedSize="1924967692"
2867             ovf:format="http://www.vmware.com/specifications/vmdk.html#streamOptimized"/>
2868     </DiskSection>
2869     <!-- Describes all networks used in the package -->
2870     <NetworkSection>
2871         <Info>Logical networks used in the package</Info>
2872         <Network ovf:name="VM Network">
2873             <Description>The network that the LAMP Service is available
2874             on</Description>
2875         </Network>
2876     </NetworkSection>
2877     <VirtualSystem ovf:id="MyLampService">
2878         <Info>Single-VM Virtual appliance with LAMP stack</Info>
2879         <Name>LAMP Virtual Appliance</Name>
2880         <!-- Overall information about the product -->
2881         <ProductSection>
2882             <Info>Product information for the service</Info>
2883             <Product>Lamp Service</Product>
2884             <Version>1.0</Version>
2885             <FullVersion>1.0.0</FullVersion>
2886         </ProductSection>
2887         <!-- Linux component configuration parameters -->
2888         <ProductSection ovf:class="org.linuxdistx">
2889             <Info>Product customization for the installed Linux system</Info>
2890             <Product>Linux Distribution X</Product>
2891             <Version>2.6.3</Version>
2892             <Property ovf:key="hostname" ovf:type="string">
2893                 <Description>Specifies the hostname for the appliance</Description>
2894             </Property>
2895             <Property ovf:key="ip" ovf:type="string">
```

```

2899         <Description>Specifies the IP address for the appliance</Description>
2900     </Property>
2901     <Property ovf:key="subnet" ovf:type="string">
2902         <Description> Specifies the subnet to use on the deployed network
2903         </Description>
2904     </Property>
2905     <Property ovf:key="gateway" ovf:type="string">
2906         <Description> Specifies the gateway on the deployed network
2907         </Description>
2908     </Property>
2909     <Property ovf:key="dns" ovf:type="string">
2910         <Description> A comma separated list of DNS servers on the deployed
2911         network </Description>
2912     </Property>
2913     <Property ovf:key="netCoreRmemMaxMB" ovf:type="uint16" ovf:value="16"
2914         ovf:userConfigurable="true">
2915         <Description> Specify TCP read max buffer size in mega bytes. Default is
2916         16. </Description>
2917     </Property>
2918     <Property ovf:key="netCoreWmemMaxMB" ovf:type="uint16" ovf:value="16"
2919         ovf:userConfigurable="true">
2920         <Description> Specify TCP write max buffer size in mega bytes. Default is
2921         16. </Description>
2922     </Property>
2923 </ProductSection>
2924 <!-- Apache component configuration parameters -->
2925 <ProductSection ovf:class="org.apache.httpd">
2926     <Info>Product customization for the installed Apache Web Server</Info>
2927     <Product>Apache Distribution Y</Product>
2928     <Version>2.6.6</Version>
2929     <Property ovf:key="httpPort" ovf:type="uint16" ovf:value="80"
2930         ovf:userConfigurable="true">
2931         <Description>Port number for HTTP requests</Description>
2932     </Property>
2933     <Property ovf:key="httpsPort" ovf:type="uint16" ovf:value="443"
2934         ovf:userConfigurable="true">
2935         <Description>Port number for HTTPS requests</Description>
2936     </Property>
2937     <Property ovf:key="startThreads" ovf:type="uint16" ovf:value="50"
2938         ovf:userConfigurable="true">
2939         <Description>Number of threads created on startup. </Description>
2940     </Property>
2941     <Property ovf:key="minSpareThreads" ovf:type="uint16" ovf:value="15"
2942         ovf:userConfigurable="true">
2943         <Description> Minimum number of idle threads to handle request spikes.
2944         </Description>
2945     </Property>
2946     <Property ovf:key="maxSpareThreads" ovf:type="uint16" ovf:value="30"
2947         ovf:userConfigurable="true">
2948         <Description>Maximum number of idle threads </Description>
2949     </Property>
2950     <Property ovf:key="maxClients" ovf:type="uint16" ovf:value="256"
2951         ovf:userConfigurable="true">
2952         <Description>Limit the number of simultaneous requests that are
2953         served. </Description>

```

```
2954     </Property>
2955 </ProductSection>
2956 <!-- MySQL component configuration parameters -->
2957 <ProductSection ovf:class="org.mysql.db">
2958     <Info>Product customization for the installed MySql Database Server</Info>
2959     <Product>MySQL Distribution Z</Product>
2960     <Version>5.0</Version>
2961     <Property ovf:key="queryCacheSizeMB" ovf:type="uint16" ovf:value="32"
2962         ovf:userConfigurable="true">
2963         <Description>Buffer to cache repeated queries for faster access (in
2964             MB)</Description>
2965     </Property>
2966     <Property ovf:key="maxConnections" ovf:type="uint16" ovf:value="500"
2967         ovf:userConfigurable="true">
2968         <Description>The number of concurrent connections that can be
2969             served</Description>
2970     </Property>
2971     <Property ovf:key="waitTimeout" ovf:type="uint16" ovf:value="100"
2972         ovf:userConfigurable="true">
2973         <Description>Number of seconds to wait before timing out a connection
2974             </Description>
2975     </Property>
2976 </ProductSection>
2977 <!-- PHP component configuration parameters -->
2978 <ProductSection ovf:class="net.php">
2979     <Info>Product customization for the installed PHP component</Info>
2980     <Product>PHP Distribution U</Product>
2981     <Version>5.0</Version>
2982     <Property ovf:key="sessionTimeout" ovf:type="uint16" ovf:value="5"
2983         ovf:userConfigurable="true">
2984         <Description> How many minutes a session has to be idle before it is
2985             timed out </Description>
2986     </Property>
2987     <Property ovf:key="concurrentSessions" ovf:type="uint16" ovf:value="500"
2988         ovf:userConfigurable="true">
2989         <Description> The number of concurrent sessions that can be served
2990             </Description>
2991     </Property>
2992     <Property ovf:key="memoryLimit" ovf:type="uint16" ovf:value="32"
2993         ovf:userConfigurable="true">
2994         <Description> How much memory in megabytes a script can consume before
2995             being killed </Description>
2996     </Property>
2997 </ProductSection>
2998 <OperatingSystemSection ovf:id="99">
2999     <Info>Guest Operating System</Info>
3000     <Description>Linux 2.6.x</Description>
3001 </OperatingSystemSection>
3002 <VirtualHardwareSection>
3003     <Info>Virtual Hardware Requirements: 256MB, 1 CPU, 1 disk, 1 NIC</Info>
3004     <System>
3005         <vssd:ElementName>Virtual Hardware Family</vssd:ElementName>
3006         <vssd:InstanceID>0</vssd:InstanceID>
3007         <vssd:VirtualSystemType>vmx-04</vssd:VirtualSystemType>
3008     </System>
```

```
3009     <Item>
3010         <rasd:Description>Number of virtual CPUs</rasd:Description>
3011         <rasd:ElementName>1 virtual CPU</rasd:ElementName>
3012         <rasd:InstanceID>1</rasd:InstanceID>
3013         <rasd:ResourceType>3</rasd:ResourceType>
3014         <rasd:VirtualQuantity>1</rasd:VirtualQuantity>
3015     </Item>
3016     <Item>
3017         <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
3018         <rasd:Description>Memory Size</rasd:Description>
3019         <rasd:ElementName>256 MB of memory</rasd:ElementName>
3020         <rasd:InstanceID>2</rasd:InstanceID>
3021         <rasd:ResourceType>4</rasd:ResourceType>
3022         <rasd:VirtualQuantity>256</rasd:VirtualQuantity>
3023     </Item>
3024     <Item>
3025         <rasd:AutomaticAllocation>true</rasd:AutomaticAllocation>
3026         <rasd:Connection>VM Network</rasd:Connection>
3027         <rasd:ElementName>Ethernet adapter on "VM Network"</rasd:ElementName>
3028         <rasd:InstanceID>3</rasd:InstanceID>
3029         <rasd:ResourceType>10</rasd:ResourceType>
3030     </Item>
3031     <Item>
3032         <rasd:ElementName>SCSI Controller 0 - LSI Logic</rasd:ElementName>
3033         <rasd:InstanceID>4</rasd:InstanceID>
3034         <rasd:ResourceSubType>LsiLogic</rasd:ResourceSubType>
3035         <rasd:ResourceType>6</rasd:ResourceType>
3036     </Item>
3037     <Item>
3038         <rasd:ElementName>Harddisk 1</rasd:ElementName>
3039         <rasd:HostResource>ovf:/disk/lamp</rasd:HostResource>
3040         <rasd:InstanceID>5</rasd:InstanceID>
3041         <rasd:Parent>4</rasd:Parent>
3042         <rasd:ResourceType>17</rasd:ResourceType>
3043     </Item>
3044 </VirtualHardwareSection>
3045 </VirtualSystem>
3046 </Envelope>
3047
```

3048
3049
3050
3051

ANNEX D (参考情報)

複数仮想システムの LAMP スタックの例

3052 これは、LAMP 仮想アプライアンスに使用する OVF デスクリプタが、複数 VM の LAMP 仮想アプライアンスでど
3053 のようなものになるかを示す例である。LAMP は、Linux オペレーティング・システム、Apache Web サーバー、
3054 MySQL データベース、および PHP Web 開発ソフトウェア・パッケージを使用して構築したサービスの略称であ
3055 る。

3056 D.1 2層 LAMP の OVF デスクリプタ

3057 2層 LAMP スタックでは、高い拡張性を実現する目的で、アプリケーション層(Linux、Apache、PHP)とデータ
3058 ベース層(Linux、MySQL サーバー)をそれぞれ別々の仮想マシンとして実行する。

3059 この OVF フォーマットでは、サービスがどのように実装されるかという点をユーザー側で把握している必要がない
3060 ようにしている。この結果、特に、ユーザーが単一 VM をインストールする場合と2層 LAMP アプライアンスをイン
3061 ストールする場合では、そのデプロイ経験が類似したものになる。ユーザーにとっての唯一の明らかな違いは、2
3062 つの IP アドレスと2つの DNS ホスト名を指定しなければならないという点である。

3063 単一 VM デスクリプタと比べ、2層 LAMP アプライアンスでは以下の点を変更されている。

- 3064 • ユーザー構成可能なパラメーターはすべて `VirtualSystemCollection` エンティティに置かれて
3065 いる。Apache、MySQL、PHP の `ProductSection` 要素は単一 VM の場合と変わらない。
- 3066 • 2つの仮想マシンの Linux ソフトウェアは、IP とホスト名を異なるものにする必要があるが、ほとんどの
3067 パラメーターを共有できる。ユーザーにプロンプトを表示するために、新しい `ProductSection` を
3068 `VirtualSystemCollection` に追加している。また、各 `VirtualSystem` エンティティに値を割り当て
3069 るために、`${property}` 式を使用している。
- 3070 • ダウンロード・サイズを単一 VM アプライアンスと同程度のサイズに維持する目的で、ディスク・チェーン
3071 を使用している。インストールした Linux は共有ベース・ディスクに格納されるので、Linux のダウンロード
3072 は実質的に1回だけですむ。

3073 以下に、この OVF デスクリプタの全体を示す。

```
3074 <?xml version="1.0" encoding="UTF-8"?>
3075 <Envelope
3076   xmlns="http://schemas.dmtf.org/ovf/envelope/1"
3077   xmlns:ovf="http://schemas.dmtf.org/ovf/envelope/1"
3078   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3079   xmlns:vssd="http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_VirtualSystemSetting
3080 Data"
3081   xmlns:rasd="http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_ResourceAllocationSe
3082 ttingData"
3083   <!-- References to all external files. -->
3084   <References>
3085     <File ovf:id="lamp-base" ovf:href="lampdb.vmdk" ovf:size="180114671"/>
3087
```

```
3088     <File ovf:id="lamp-db" ovf:href="lampdb.vmdk" ovf:size="1801146"/>
3089     <File ovf:id="lamp-app" ovf:href="lampapp.vmdk" ovf:size="34311371"/>
3090 </References>
3091 <!-- Describes meta-information about all virtual disks in the package.
3092      This example is encoded as a delta-disk hierarchy.
3093 -->
3094 <DiskSection>
3095     <Info>List of the virtual disks used in the package</Info>
3096     <Disk ovf:diskId="lamp-base" ovf:fileRef="lamp-base" ovf:capacity="4294967296"
3097         ovf:populatedSize="1924967692"
3098     ovf:format="http://www.vmware.com/specifications/vmdk.html#streamOptimized"/>
3099     <Disk ovf:diskId="lamp-db" ovf:fileRef="lamp-db" ovf:capacity="4294967296"
3100         ovf:populatedSize="19249672"
3101     ovf:format="http://www.vmware.com/specifications/vmdk.html#streamOptimized"
3102     ovf:parentRef="lamp-base"/>
3103     <Disk ovf:diskId="lamp-app" ovf:fileRef="lamp-app" ovf:capacity="4294967296"
3104         ovf:populatedSize="2349692"
3105     ovf:format="http://www.vmware.com/specifications/vmdk.html#streamOptimized"
3106     ovf:parentRef="lamp-base"/>
3107 </DiskSection>
3108 <!-- Describes all networks used in the package -->
3109 <NetworkSection>
3110     <Info>Logical networks used in the package</Info>
3111     <Network ovf:name="VM Network">
3112         <Description>The network that the LAMP Service is available
3113         on</Description>
3114     </Network>
3115 </NetworkSection>
3116 <VirtualSystemCollection ovf:id="LampService">
3117     <Info>Virtual appliance with a 2-tier distributed LAMP stack</Info>
3118     <Name>LAMP Service</Name>
3119     <!-- Overall information about the product -->
3120     <ProductSection ovf:class="org.mylamp">
3121         <Info>Product information for the service</Info>
3122         <Product>My Lamp Service</Product>
3123         <Version>1.0</Version>
3124         <FullVersion>1.0.0</FullVersion>
3125     </ProductSection>
3126     <ProductSection ovf:class="org.linuxdist">
3127         <Info>Product customization for Operating System Level</Info>
3128         <Product>Linux Distribution X</Product>
3129         <Version>2.6.3</Version>
3130         <Property ovf:key="dbHostname" ovf:type="string">
3131             <Description>Specifies the hostname for database virtual
3132             machine</Description>
3133         </Property>
3134         <Property ovf:key="appHostname" ovf:type="string">
3135             <Description>Specifies the hostname for application server virtual
3136             machine</Description>
3137         </Property>
3138         <Property ovf:key="dbIp" ovf:type="string">
3139             <Description>Specifies the IP address for the database virtual
```

```
3143         machine</Description>
3144     </Property>
3145     <Property ovf:key="appIp" ovf:type="string">
3146         <Description>Specifies the IP address for application server
3147         VM</Description>
3148     </Property>
3149     <Property ovf:key="subnet" ovf:type="string">
3150         <Description> Specifies the subnet to use on the deployed network
3151         </Description>
3152     </Property>
3153     <Property ovf:key="gateway" ovf:type="string">
3154         <Description> Specifies the gateway on the deployed network
3155         </Description>
3156     </Property>
3157     <Property ovf:key="dns" ovf:type="string">
3158         <Description> A comma separated list of DNS servers on the deployed
3159         network </Description>
3160     </Property>
3161     <Property ovf:key="netCoreRmemMaxMB" ovf:type="uint16" ovf:value="16"
3162         ovf:userConfigurable="true">
3163         <Description> Specify TCP read max buffer size in mega bytes. Default is
3164         16. </Description>
3165     </Property>
3166     <Property ovf:key="netCoreWmemMaxMB" ovf:type="uint16" ovf:value="16"
3167         ovf:userConfigurable="true">
3168         <Description> Specify TCP write max buffer size in mega bytes. Default is
3169         16. </Description>
3170     </Property>
3171 </ProductSection>
3172 <!-- Apache component configuration parameters -->
3173 <ProductSection ovf:class="org.apache.httpd">
3174     <Info>Product customization for the installed Apache Web Server</Info>
3175     <Product>Apache Distribution Y</Product>
3176     <Version>2.6.6</Version>
3177     <Property ovf:key="httpPort" ovf:type="uint16" ovf:value="80"
3178         ovf:userConfigurable="true">
3179         <Description>Port number for HTTP requests</Description>
3180     </Property>
3181     <Property ovf:key="httpsPort" ovf:type="uint16" ovf:value="443"
3182         ovf:userConfigurable="true">
3183         <Description>Port number for HTTPS requests</Description>
3184     </Property>
3185     <Property ovf:key="startThreads" ovf:type="uint16" ovf:value="50"
3186         ovf:userConfigurable="true">
3187         <Description>Number of threads created on startup. </Description>
3188     </Property>
3189     <Property ovf:key="minSpareThreads" ovf:type="uint16" ovf:value="15"
3190         ovf:userConfigurable="true">
3191         <Description>Minimum number of idle threads to handle request spikes.
3192         </Description>
3193     </Property>
3194     <Property ovf:key="maxSpareThreads" ovf:type="uint16" ovf:value="30"
3195         ovf:userConfigurable="true">
3196         <Description>Maximum number of idle threads </Description>
3197     </Property>
```



```

3198     <Property ovf:key="maxClients" ovf:type="uint16" ovf:value="256"
3199         ovf:userConfigurable="true">
3200         <Description>Limits the number of simultaneous requests that are
3201             served. </Description>
3202     </Property>
3203 </ProductSection>
3204 <!-- MySQL component configuration parameters -->
3205 <ProductSection ovf:class="org.mysql.db">
3206     <Info>Product customization for the installed MySql Database Server</Info>
3207     <Product>MySQL Distribution Z</Product>
3208     <Version>5.0</Version>
3209     <Property ovf:key="queryCacheSizeMB" ovf:type="uint16" ovf:value="32"
3210         ovf:userConfigurable="true">
3211         <Description>Buffer to cache repeated queries for faster access (in
3212             MB)</Description>
3213     </Property>
3214     <Property ovf:key="maxConnections" ovf:type="uint16" ovf:value="500"
3215         ovf:userConfigurable="true">
3216         <Description>The number of concurrent connections that can be
3217             served</Description>
3218     </Property>
3219     <Property ovf:key="waitTimeout" ovf:type="uint16" ovf:value="100"
3220         ovf:userConfigurable="true">
3221         <Description>Number of seconds to wait before timing out a connection
3222             </Description>
3223     </Property>
3224 </ProductSection>
3225 <!-- PHP component configuration parameters -->
3226 <ProductSection ovf:class="net.php">
3227     <Info>Product customization for the installed PHP component</Info>
3228     <Product>PHP Distribution U</Product>
3229     <Version>5.0</Version>
3230     <Property ovf:key="sessionTimeout" ovf:type="uint16" ovf:value="5"
3231         ovf:userConfigurable="true">
3232         <Description> How many minutes a session has to be idle before it is
3233             timed out </Description>
3234     </Property>
3235     <Property ovf:key="concurrentSessions" ovf:type="uint16" ovf:value="500"
3236         ovf:userConfigurable="true">
3237         <Description> The number of concurrent sessions that can be served
3238             </Description>
3239     </Property>
3240     <Property ovf:key="memoryLimit" ovf:type="uint16" ovf:value="32"
3241         ovf:userConfigurable="true">
3242         <Description> How much memory in megabytes a script can consume before
3243             being killed </Description>
3244     </Property>
3245 </ProductSection>
3246 <StartupSection>
3247     <Info>Startup order of the virtual machines</Info>
3248     <Item ovf:id="DbServer" ovf:order="1" ovf:startDelay="120"
3249         ovf:startAction="powerOn" ovf:waitingForGuest="true"
3250         ovf:stopDelay="120"
3251         ovf:stopAction="guestShutdown"/>
3252     <Item ovf:id="AppServer" ovf:order="2" ovf:startDelay="120"

```

```
3253         ovf:startAction="powerOn" ovf:waitingForGuest="true"
3254 ovf:stopDelay="120"
3255         ovf:stopAction="guestShutdown"/>
3256     </StartupSection>
3257     <VirtualSystem ovf:id="AppServer">
3258         <Info>The configuration of the AppServer virtual machine</Info>
3259         <Name>Application Server</Name>
3260         <!-- Linux component configuration parameters -->
3261         <ProductSection ovf:class="org.linuxdistx">
3262             <Info>Product customization for the installed Linux system</Info>
3263             <Product>Linux Distribution X</Product>
3264             <Version>2.6.3</Version>
3265             <Property ovf:key="hostname" ovf:type="string"
3266 ovf:value="{appHostName}"/>
3267             <Property ovf:key="ip" ovf:type="string" ovf:value="{appIp}"/>
3268             <Property ovf:key="subnet" ovf:type="string" ovf:value="{subnet}"/>
3269             <Property ovf:key="gateway" ovf:type="string" ovf:value="{gateway}"/>
3270             <Property ovf:key="dns" ovf:type="string" ovf:value="{dns}"/>
3271             <Property ovf:key="netCoreRmemMaxMB" ovf:type="string"
3272             ovf:value="{netCoreRmemMaxMB}"/>
3273             <Property ovf:key="netCoreWmemMaxMB" ovf:type="string"
3274             ovf:value="{netCoreWmemMaxMB}"/>
3275         </ProductSection>
3276         <OperatingSystemSection ovf:id="99">
3277             <Info>Guest Operating System</Info>
3278             <Description>Linux 2.6.x</Description>
3279         </OperatingSystemSection>
3280         <VirtualHardwareSection>
3281             <Info>Virtual Hardware Requirements: 256 MB, 1 CPU, 1 disk, 1 NIC</Info>
3282             <System>
3283                 <vssd:ElementName>Virtual Hardware Family</vssd:ElementName>
3284                 <vssd:InstanceID>0</vssd:InstanceID>
3285                 <vssd:VirtualSystemType>vmx-04</vssd:VirtualSystemType>
3286             </System>
3287             <Item>
3288                 <rasd:Description>Number of virtual CPUs</rasd:Description>
3289                 <rasd:ElementName>1 virtual CPU</rasd:ElementName>
3290                 <rasd:InstanceID>1</rasd:InstanceID>
3291                 <rasd:ResourceType>3</rasd:ResourceType>
3292                 <rasd:VirtualQuantity>1</rasd:VirtualQuantity>
3293             </Item>
3294             <Item>
3295                 <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
3296                 <rasd:Description>Memory Size</rasd:Description>
3297                 <rasd:ElementName>256 MB of memory</rasd:ElementName>
3298                 <rasd:InstanceID>2</rasd:InstanceID>
3299                 <rasd:ResourceType>4</rasd:ResourceType>
3300                 <rasd:VirtualQuantity>256</rasd:VirtualQuantity>
3301             </Item>
3302             <Item>
3303                 <rasd:AutomaticAllocation>true</rasd:AutomaticAllocation>
3304                 <rasd:Connection>VM Network</rasd:Connection>
3305                 <rasd:ElementName>Ethernet adapter on "VM
3306 Network"</rasd:ElementName>
3307                 <rasd:InstanceID>3</rasd:InstanceID>
```

```

3308         <rasd:ResourceSubType>PCNet32</rasd:ResourceSubType>
3309         <rasd:ResourceType>10</rasd:ResourceType>
3310     </Item>
3311     <Item>
3312         <rasd:ElementName>SCSI Controller 0 - LSI Logic</rasd:ElementName>
3313         <rasd:InstanceID>4</rasd:InstanceID>
3314         <rasd:ResourceSubType>LsiLogic</rasd:ResourceSubType>
3315         <rasd:ResourceType>6</rasd:ResourceType>
3316     </Item>
3317     <Item>
3318         <rasd:ElementName>Harddisk 1</rasd:ElementName>
3319         <rasd:HostResource>ovf:/disk/lamp-app</rasd:HostResource>
3320         <rasd:InstanceID>5</rasd:InstanceID>
3321         <rasd:Parent>4</rasd:Parent>
3322         <rasd:ResourceType>17</rasd:ResourceType>
3323     </Item>
3324 </VirtualHardwareSection>
3325 </VirtualSystem>
3326 <VirtualSystem ovf:id="DB Server">
3327     <Info>The configuration of the database virtual machine</Info>
3328     <Name>Database Server</Name>
3329     <!-- Linux component configuration parameters -->
3330     <ProductSection ovf:class="org.linuxdistx">
3331         <Info>Product customization for the installed Linux system</Info>
3332         <Product>Linux Distribution X</Product>
3333         <Version>2.6.3</Version>
3334         <Property ovf:key="hostname" ovf:type="string"
3335             ovf:value="{dbHostName}"/>
3336         <Property ovf:key="ip" ovf:type="string" ovf:value="{dbIp}"/>
3337         <Property ovf:key="subnet" ovf:type="string" ovf:value="{subnet}"/>
3338         <Property ovf:key="gateway" ovf:type="string" ovf:value="{gateway}"/>
3339         <Property ovf:key="dns" ovf:type="string" ovf:value="{dns}"/>
3340         <Property ovf:key="netCoreRmemMaxMB" ovf:type="string"
3341             ovf:value="{netCoreRmemMaxMB}"/>
3342         <Property ovf:key="netCoreWmemMaxMB" ovf:type="string"
3343             ovf:value="{netCoreWmemMaxMB}"/>
3344     </ProductSection>
3345     <OperatingSystemSection ovf:id="99">
3346         <Info>Guest Operating System</Info>
3347         <Description>Linux 2.6.x</Description>
3348     </OperatingSystemSection>
3349     <VirtualHardwareSection>
3350         <Info>Virtual Hardware Requirements: 256 MB, 1 CPU, 1 disk, 1 nic</Info>
3351         <System>
3352             <vssd:ElementName>Virtual Hardware Family</vssd:ElementName>
3353             <vssd:InstanceID>0</vssd:InstanceID>
3354             <vssd:VirtualSystemType>vmx-04</vssd:VirtualSystemType>
3355         </System>
3356         <Item>
3357             <rasd:Description>Number of virtual CPUs</rasd:Description>
3358             <rasd:ElementName>1 virtual CPU</rasd:ElementName>
3359             <rasd:InstanceID>1</rasd:InstanceID>
3360             <rasd:ResourceType>3</rasd:ResourceType>
3361             <rasd:VirtualQuantity>1</rasd:VirtualQuantity>
3362         </Item>

```

```
3363         <Item>
3364             <rasd:AllocationUnits>byte * 2^20</rasd:AllocationUnits>
3365             <rasd:Description>Memory Size</rasd:Description>
3366             <rasd:ElementName>256 MB of memory</rasd:ElementName>
3367             <rasd:InstanceID>2</rasd:InstanceID>
3368             <rasd:ResourceType>4</rasd:ResourceType>
3369             <rasd:VirtualQuantity>256</rasd:VirtualQuantity>
3370         </Item>
3371         <Item>
3372             <rasd:AutomaticAllocation>>true</rasd:AutomaticAllocation>
3373             <rasd:Connection>VM Network</rasd:Connection>
3374             <rasd:ElementName>Ethernet adapter on "VM
3375 Network"</rasd:ElementName>
3376             <rasd:InstanceID>3</rasd:InstanceID>
3377             <rasd:ResourceType>10</rasd:ResourceType>
3378         </Item>
3379         <Item>
3380             <rasd:ElementName>SCSI Controller 0 - LSI Logic</rasd:ElementName>
3381             <rasd:InstanceID>4</rasd:InstanceID>
3382             <rasd:ResourceSubType>LsiLogic</rasd:ResourceSubType>
3383             <rasd:ResourceType>6</rasd:ResourceType>
3384         </Item>
3385         <Item>
3386             <rasd:ElementName>Harddisk 1</rasd:ElementName>
3387             <rasd:HostResource>ovf:/disk/lamp-db</rasd:HostResource>
3388             <rasd:InstanceID>5</rasd:InstanceID>
3389             <rasd:Parent>4</rasd:Parent>
3390             <rasd:ResourceType>17</rasd:ResourceType>
3391         </Item>
3392     </VirtualHardwareSection>
3393 </VirtualSystem>
3394 </VirtualSystemCollection>
3395 </Envelope>
3396
```

3397
3398
3399

ANNEX E (参考情報) 拡張性の例

3400 OVF 仕様では、いくつかの方法でカスタム・メタデータを OVF デスクリプタに追加できる。

- 3401 • 新しい Section 要素を Section 代替グループの一部として定義しておく、OVF スキーマでセクションが出現できる場所ならばどこでも、その新しい Section 要素を使用できる。
- 3403 • OVF スキーマはオープン・コンテンツ・モデルを使用する。このモデルでは、既存のタイプはすべて、末尾に要素を追加して拡張できる。拡張ポイントを宣言するには、namespace="##other"を指定した xs:any 宣言を OVF スキーマで使用する。
- 3406 • OVF スキーマでは、既存のタイプに属性を追加できる。

3407 カスタム・メタデータは、OVF XML 名前空間を使用できない。カスタム要素では、正しい動作のためには要素内の情報が必須なのか、省略可能なのかを Boolean の ovf:required 属性で指定する。

3409 OVF スキーマのオープン・コンテンツ・モデルでは、既存のタイプの末尾でのみ拡張が可能である。XML スキーマ 1.0 では、一意のパーティクル属性の規則があることおよびスキーマのあらゆるところに xs:any 宣言を追加する必要があることから、より柔軟なオープン・コンテンツ・モデルを可能にすることは簡単ではない。XML スキーマ 1.1 標準の原案には、xs:openContent mode="interleave"宣言を使用した、さらに柔軟なオープン・コンテンツ・メカニズムが規定されている。

3414 E.1 カスタム・スキーマ

3415 以下に、2 つの拡張タイプを定義するカスタム XML スキーマを示す。最初の宣言は、OVF の Section 代替グループのカスタム・メンバーを定義し、2 番目の宣言は単純なカスタム・タイプを定義している。

```

3417 <?xml version="1.0" encoding="UTF-8"?>
3418 <xs:schema
3419   targetNamespace="http://schemas.customextension.org/1"
3420   xmlns:custom="http://schemas.customextension.org/1"
3421   xmlns="http://schemas.customextension.org/1"
3422   xmlns:ovf="http://schemas.dmtf.org/ovf/envelope/1"
3423   xmlns:xs="http://www.w3.org/2001/XMLSchema"
3424   attributeFormDefault="qualified"
3425   elementFormDefault="qualified">
3426
3427   <!-- Define a custom member of the ovf:Section substitution group -->
3428   <xs:element name="CustomSection" type="custom:CustomSection_Type"
3429 substitutionGroup="ovf:Section"/>
3430
3431   <xs:complexType name="CustomSection_Type">
3432     <xs:complexContent>
3433       <xs:extension base="ovf:Section_Type">
3434         <xs:sequence>
3435           <xs:element name="Data" type="xs:string"/>
3436         </xs:sequence>
3437         <xs:anyAttribute namespace="##any" processContents="lax"/>
3438       </xs:extension>
3439     </xs:complexContent>

```

```

3440     </xs:complexType>
3441
3442
3443     <!-- Define other simple custom type not part of ovf:Section substitution group -->
3444     <xs:complexType name="CustomOther_Type">
3445         <xs:sequence>
3446             <xs:element name="Data" type="xs:string"/>
3447         </xs:sequence>
3448         <xs:attribute ref="ovf:required"/>
3449         <xs:anyAttribute namespace="##any" processContents="lax"/>
3450     </xs:complexType>
3451
3452 </xs:schema >

```

3453 E.2 カスタム拡張を使用したデスクリプタ

3454 以下に、カスタム・スキーマを使用したOVFデスクリプタの全体を示す。このデスクリプタはOVFスキーマとカスタム・スキーマと照合して検証されているが、拡張の例を除けば、必要最小限のデスクリプタとなっている。

3456 このデスクリプタは、OVF のカスタム Section 要素、拡張ポイントでのカスタム要素、およびカスタム属性の 3
3457 つの拡張タイプをすべて記述している。

```

3458 <?xml version="1.0" encoding="UTF-8"?>
3459 <Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3460
3461 xmlns:vssd="http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_VirtualSystemSetting
3462 Data"
3463
3464 xmlns:rasd="http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_ResourceAllocationSe
3465 ttingData"
3466     xmlns:ovf="http://schemas.dmtf.org/ovf/envelope/1"
3467     xmlns="http://schemas.dmtf.org/ovf/envelope/1"
3468     xmlns:custom="http://schemas.customextension.org/1">
3469
3470     <!-- Dummy References element -->
3471     <References/>
3472
3473     <!-- EXAMPLE: Optional custom OVF section element with validation against custom schema
3474 -->
3475     <custom:CustomSection ovf:required="false">
3476         <Info>Description of custom extension</Info>
3477         <custom:Data>somevalue</custom:Data>
3478     </custom:CustomSection>
3479
3480     <!-- Describes all networks used in the package -->
3481     <NetworkSection>
3482         <Info>Logical networks used in the package</Info>
3483         <!-- EXAMPLE: Optional custom attribute -->
3484         <Network ovf:name="VM Network" custom:desiredCapacity="1 Gbit/s"/>
3485         <!-- EXAMPLE: Optional custom metadata inserted at extension point with validation
3486 against custom schema -->
3487         <custom:CustomOther xsi:type="custom:CustomOther_Type" ovf:required="false">
3488             <custom:Data>somevalue</custom:Data>
3489         </custom:CustomOther>
3490     </NetworkSection>

```

```
3491  
3492     <!-- Dummy Content element -->  
3493     <VirtualSystem ovf:id="Dummy">  
3494         <Info>Dummy VirtualSystem</Info>  
3495     </VirtualSystem>  
3496 </Envelope>
```

3497 OVF 環境 XML スキーマには、OVF エンベロープ XML スキーマと同じ拡張メカニズムが定義されているので、
3498 OVF 環境文書も同様に拡張可能である。

3499

3500

3501

ANNEX F
(参考情報)
変更履歴

3502

バージョン	日付	説明
1.0.0	2009/02/17	
1.0.1	2011/10/20	DMTF 正誤表公開用リリース
2.0.0	2014/04/24	情報として DMTF 公開用リリース

3503