

# Horizon DaaS Platform 6.1 REST API

This document provides an overview of the Horizon DaaS Platform REST API.

August 2014

**vmware**

## Revision History

Date	Version	Description
09/04/2014	1.0	Initial release

© 2014 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at <http://www.vmware.com/go/patents>.

VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

**VMware, Inc.**  
3401 Hillview Ave.  
Palo Alto, CA 94304  
[www.vmware.com](http://www.vmware.com)

# Contents

1 About This Document	6
1.1 Intended Audience	6
1.2 Organization of this Document	6
1.3 Questions	6
2 Understanding the API	7
2.1 Resources	7
2.2 Scope	7
2.3 Properties and Links	8
2.4 Link Attributes	8
2.5 HTTP Requests	8
3 Getting Started	10
3.1 Retrieve the Available Horizon DaaS Versions	10
3.2 Login and Save the HTTP Response Header Authorization	11
3.3 Get the DtSecurityManager	11
3.4 Retrieve All Users for a Domain	12
3.5 Get the DtInfrastructureManager	13
3.6 Retrieve the Mapped Pools or Pattern for a Specific User	13
3.7 Example Error	15
4 Resources	16
4.1 DtVersion	16
4.2 DtSystemManager	17
4.3 DtInfrastructureManager	17
4.3.1 DtDataCenter	20
4.3.2 DtDesktopModel	21
4.3.2.1 DtLicenseEntitlement	22
4.3.3 DtDesktopModelQuota	23
4.3.4 DtHypervisorManager	23
4.3.5 DtComputePool	24
4.3.5.1 DtComputePoolProperty	25
4.3.6 DtNetwork	26
4.3.7 DtRemoteProtocolQuota	26
4.3.8 DtSessionQuota	27
4.3.9 DtStorageConfig	27
4.3.10 DtStorageSystem	28
4.3.11 DtTemplateQuota	28
4.3.12 DtTenantDesktopManager	29
4.4 DtPoolManager	29
4.4.1 DtDesktopPool	30
4.4.1.1 DtPoolPolicy	32
4.4.2 DtDynamicDesktopPattern	33
4.4.3 DtGoldPattern	34
4.4.4 DtPoolTask	35
4.4.5 DtRemoteApplication	35
4.4.6 DtStaticDesktopPattern	36
4.4.7 DtVirtualMachine	37
4.5 DtInstallManager	38
4.5.1 DtAppliance	38
4.6 DtReportingManager	39
4.6.1 DtBillingReportFilter	40
4.6.2 DtConcurrentUsersFilter	40

4.6.3 DtConcurrentUsersReport	41
4.6.4 DtQuotaBillingReport	41
4.6.5 DtReportFilter	42
4.6.6 DtSuperTenantBillingReport	42
4.6.7 DtTenantReport	42
4.6.8 DtTenantReportKey	43
4.6.9 DtUserEventReport	43
4.6.10 DtUserEventReportFilter	44
4.7 DtSecurityManager	44
4.7.1 DtDomain	45
4.7.1.1 DtDomainAccount	46
4.7.1.2 DtGroup	46
4.7.1.3 DtRole	47
4.7.1.4 DtUser	47
4.7.2 DtOrganization	48
4.7.2.1 DtContact	49
<b>5 Enumerated Data Types</b>	<b>50</b>
5.1 DtComputePoolType	50
5.2 DtDisplayProtocol	50
5.3 DtDomainAccountType	51
5.4 DtEndpointPlatformType	51
5.5 DtNetworkType	51
5.6 DtPatternType	51
5.7 DtPoolSizeType	52
5.8 DtQuotaBillingType	52
5.9 DtTaskStatus	52
5.10 DtUserEvent	52
5.11 DtUserEventError	53
5.12 DtVMLifeState	53
5.13 DtVMPowerState	54

This page intentionally left blank

# 1 About This Document

---

This document provides an overview of the Horizon DaaS Platform REST API.

## 1.1 Intended Audience

The audience for this document is those who want to programmatically use Horizon DaaS Platform APIs. This document assumes that you are familiar with:

- REST
- Software as a service

## 1.2 Organization of this Document

[About This Document](#) describes the content and intent of subsequent sections and introduces some terminology used in the remainder of the document.

[Understanding the API](#) introduces the architecture of the Horizon DaaS Platform API and the concepts you need to understand to use the API, such as the links through which you traverse the object model to retrieve resources.

[Getting Started](#) presents examples that explain the basics of using the API. The examples show you how to login, save the HTTP Response Header authorization for use in subsequent HTTP requests, and perform several actions by traversing the object model to obtain the URIs for those actions.

[Resources](#) lists the resources, their links, and their attributes.

[Enumerated Data Types](#) lists the enumerated types that are the legal values for certain resource attributes.

## 1.3 Questions

If you need assistance or have questions about any of the information in this document, contact VMware Customer Support.

## 2 Understanding the API

The Horizon DaaS API provides access to Horizon DaaS platform features and functionality (resources) via REST-based web services. Exchanges between the caller and the platform take place using HTTP and HTTPS requests and the platform returns XML to the caller in the HTTP response.

### 2.1 Resources

Figure 1 illustrates the hierarchy of resources available in the Horizon DaaS REST API.

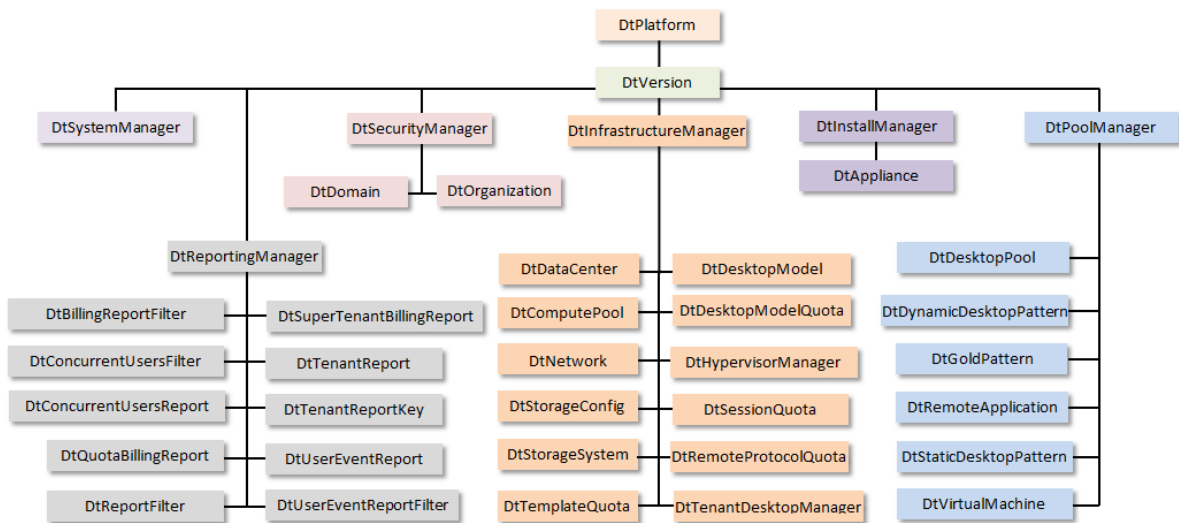


Figure 1: Resource Tree

### 2.2 Scope

The Horizon DaaS Platform REST APIs have a common object model across its management appliances – Service Provider or Tenant. However, there are some resources that are available only to Service Providers and others that are only available to the tenants. Amongst the resources that are common to both service provider and tenants, there is a scope associated to individual links in the resource instances that restricts the availability of some of the links to service providers or tenants. So the same resource will contain different links based on the context of its retrieval.

## 2.3 Properties and Links

Resources contain properties and links:

- Properties store values. For example, the DtUser resource has a loginName property that indicates the name used by a user to login.
- A link specifies a URI of a related resource or action. For example, the DtUser resource has a desktopPatterns link that retrieves the desktop patterns assigned to a user.

## 2.4 Link Attributes

A link has the following attributes:

- **name** – The unique name of the link. The name typically describes the purpose or target of the attribute.
- **href** – The URI relative to the server, webapp and version, for example /infrastructure/pool/1000
- **method** – The http method used with the href in the REST API invocation: GET, POST, PUT, or DELETE.
- **rel** – Describes the relationship between the link and the resource that contains the link. The rel attribute is for informational purposes and is not required when coding to the API. The table below defines the link relationships.

**Table 2–1 Link Relationships**

Relationship	Description
ACTION	Indicates that the URI performs an action with the resource
ASSOCIATION	Indicates that the URI retrieves an associated resource
MAP	Indicates that the URI retrieves a MAP between resources (one-one, one-many, many-many)
SELF	Indicates that the URI is the URI of the container resource. A REST response always contains a link to itself. A GET on this URI returns the same URI
TOP	Indicates that the URI is a top-level root resource

## 2.5 HTTP Requests

The HTTP request you use to retrieve a Horizon DaaS resource is constructed as follows:

`https://<host>/dt-rest/<version>/<URI>`

- *host* is the IP address of the Enterprise Center or Service Center.
- *version* is the version of the REST API (See Section 3.1 on page 5 for information on retrieving available versions).
- *URI* is the path to the resource, constructed by traversing the object model and extracting path information from the href attribute of a link.

For example:

`/infrastructure/manager/mapping/users/default?sep={sep}&users={users}&`

Note the following features of URI syntax:

- The first component of the path is always the functional area of the Horizon DaaS Platform: system, infrastructure, or security.
- The /manager component of the path is used only for top-level resources (infrastructure, system, and security) and links served by that resource.
- Curly braces are used to represent request parameter values which you must supply in the actual HTTP request. The href attribute uses the following syntax to represent request parameters:

?p1={p1}&p2={p2}

So for the preceding example, the actual values for the two request parameters (sep and users) might look as follows (where sep specifies the character you are using to delimit the user ids in the request):

?sep=,&users=1234,5678&

- URL encoding is used to replace characters outside the ASCII set with a percent sign character (%) followed by two hexadecimal digits. For example, the question mark character (?) is replaced with %3F and the ampersand character (&) is replaced with %26.

Here is the complete HTTP request:

```
https://Host/dt-  
rest/v100/infrastructure/manager/mapping/users/default?sep=,&users=1234,5678&
```

## 3 Getting Started

---

This chapter gives you the basics you need to get started using the API, showing you how to:

- Retrieve the available Horizon DaaS versions via the published URL `/system/platform`
- Login and save the HTTP Response Header authorization.
- Get the Security Manager.
- Retrieve a domain and all users for that domain.
- Get the Infrastructure Manager.
- Retrieve the mapped pools or pattern for a specific user.

Before using the samples in this chapter to begin coding to the API, you will need the Tenant or Service Provider IP address and version 6.0.0 or greater of the Horizon DaaS Platform installed.

### 3.1 Retrieve the Available Horizon DaaS Versions

To retrieve the available Horizon DaaS versions, submit the following HTTP request, where Host is the IP address of the Enterprise Center or Service Center.

```
https://Host/dt-rest/system/platform
```

The response includes a single `DtPlatform` XML element which contains one `DtVersion` element for each available Horizon DaaS version. The latest version is the `DtVersion` element for which the attribute `latest="true"`. For example:

```
<DtVersion id="v100" latest="true">
  <DtLink href="/system" method="GET"
    name="DtSystemManager" rel="top"></DtLink>
  <DtLink href="/infrastructure/manager" method="GET"
    name="DtInfrastructureManager" rel="top"></DtLink>
  <DtLink href="/security/manager" method="GET"
    name="DtSecurityManager" rel="top"></DtLink>
  <loginURI>/v100/system/login</loginURI>
</DtVersion>
```

The `DtVersion` element also defines the three top-level links you use to traverse the Horizon DaaS REST API: `DtSystemManager`, `DtInfrastructureManager`, and `DtSecurityManager`.

## 3.2 Login and Save the HTTP Response Header Authorization

The API provides secure access via domain authentication. The loginURI property of the DtVersion element specifies URI you need to append to the base URL to submit an HTTP POST to login:

```
<loginURI>/v100/system/login</loginURI>
```

For example, if the login URI is /v100/system/login, then you could submit the following HTTP POST.

```
https://Host/dt-rest/v100/system/login
```

where the POST parameters are the user, pw, and domain.

The HTTP response header you receive when you login contains key-value pairs, one of which is named Authorization. The table below lists some of values you receive in the HTTP response header.

**Table 3–1 Sample Hardware Requirements for Horizon DaaS Management Host**

Key	Value
Server	Apache-Coyote/1.1
Authorization	Basic REVWXGVudGFkbWluOkRlc2t0b25lMQ==
Transfer-Encoding	chunked
Content-Encoding	gzip
Vary	Accept-Encoding
Date	Wed, 07 Dec 2011 14:53:05 GMT

You need to save the value for Authorization when you receive the HTTP response to your login request. Then, in every subsequent HTTP request you submit after you login, include this Authorization key-value pair in the HTTP request header.

## 3.3 Get the DtSecurityManager

To retrieve the DtSecurityManager, use the path information specified by the DtSecurityManager link of the DtVersion element:

```
<DtLink href="/security/manager" method="GET" name="DtSecurityManager" rel="top">
```

The href attribute indicates the URI to append to the base URI. Using this information, you could construct the following HTTP request:

```
https://Host/dt-rest/v100/security/manager
```

The response contains the following XML:

```
<List type="DtLink">
  <DtLink href="/security/manager" method="GET" name="DtSecurityManager"
rel="self"></DtLink>
  <DtLink href="/security/manager/domains" method="GET"
name="Domains" rel="association"> </DtLink>
  <DtLink href="/security/manager/organizations" method="GET"
name="Organizations" rel="association"> </DtLink>
</List>
```

This XML specifies three links you can traverse:

- The first link is a link to the DtSecurityManager itself.
- The second link indicates the URI to append to the base URL to retrieve all the domains.

- The third link indicates the URI to append to the base URL to retrieve all the organizations.

### 3.4 Retrieve All Users for a Domain

To retrieve all domains, use the path information specified by the Domains link of the DtSecurityManager element:

```
<DtLink href="/security/manager/domains" method="GET" name="Domains"
rel="association"></DtLink>
```

The href attribute indicates the URI to append to the base URI. Using this information, you could construct the following HTTP request:

```
https://Host/dt-rest/v100/security/manager/domains
```

The response contains one DtDomain element for each available domain, and each domain element contains four links:

- The first link is a link to the DtDomain itself.
- The second link indicates the URI to append to the save a domain.
- The third link indicates the URI to append to the base URL to retrieve all the groups in the domain.
- The fourth link indicates the URI to append to the base URL to retrieve all the users in the domain.

For example:

```
<DtLink href="/security/domain/DEV" method="GET" name="DtDomain"
rel="self"></DtLink>
<DtLink href="/security/domain/DEV/save" method="POST" name="Save"
rel="action"></DtLink>
<DtLink href="/security/domain/DEV/groups" method="GET" name="Groups"
rel="association"></DtLink>
<DtLink href="/security/domain/DEV/users" method="GET" name="Users"
rel="association"></DtLink>
```

To retrieve all users for the DEV domain, use the path information specified by the Users link of the DtDomain element:

```
<DtLink href="/security/domain/DEV/users" method="GET" name="Users"
rel="association"></DtLink>
```

The href attribute indicates the URI to append to the base URI. Using this information, you could construct the following HTTP request:

```
https://Host/dt-rest/v100/security/domain/DEV/users
```

The response contains one XML DtUser element for each user. Each DtUser element specifies the links you can traverse to obtain properties for the User. The following XML fragment shows two of the returned links:

```
<DtLink href="/security/domain/DEV/user/563d81c8d85a6f46a6d7599ad7b6d1c5/pools"
method="GET" name="DesktopPools" rel="association"></DtLink>

<DtLink href="/security/domain/DEV/user/563d81c8d85a6f46a6d7599ad7b6d1c5/patterns"
method="GET" name="DesktopPatterns" rel="association"></DtLink>
```

The user id in this example is 563d81c8d85a6f46a6d7599ad7b6d1c5. To obtain information for this user, save the href attribute for use in subsequent calls (you can also save the self DtLink).

### 3.5 Get the DtInfrastructureManager

To retrieve the DtInfrastructureManager, use the path information specified by the DtSecurityManager link of the DtVersion element:

```
<DtLink href="/infrastructure/manager" method="GET"
  name="DtInfrastructureManager" rel="top"></DtLink>
```

The href attribute indicates the URI to append to the base URI. Using this information, you could construct the following HTTP request:

```
https://Host/dt-rest/v100/infrastructure/manager
```

The response specifies the links you can traverse to obtain properties such as patterns, pools, desktop models, and quotas. The following XML fragment shows two of the returned links:

```
<DtLink href="/infrastructure/manager/dcs" method="GET"
  name="DataCenters" rel="association"></DtLink>
<DtLink href="/infrastructure/manager/models" method="GET"
  name="DesktopModels" rel="association"></DtLink>
```

### 3.6 Retrieve the Mapped Pools or Pattern for a Specific User

To retrieve the mapped pools or pattern for a list of users, use the path information specified by the DefaultMappingOfUsers link of the DtInfrastructureManager:

```
<DtLink href="/infrastructure/manager/mapping/users/default?sep={sep}&users={users}&"
  method="GET" name="DefaultMappingOfUsers" rel="map"></DtLink>
```

The href attribute indicates the URI to append to the base URI. Using this information, you could construct the following HTTP request:

```
https://Host/dt-rest/v100/infrastructure/manager/mapping/users/default?sep=,&users=563d81c8d85a6f46a6d7599ad7b6d1c5,bcce77ea74cdd148b9c2c08d84dfff2a&
```

The response contains one XML Entry element for each user. Each Entry element specifies the default pool or pattern for the user. For example:

```
<Map>
  <Entry>
    <key xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
      xmlns:xs=http://www.w3.org/2001/XMLSchema
      xsi:type="xs:string">563d81c8d85a6f46a6d7599ad7b6d1c5</key>
    <DtDesktopPool id="1078">
      <DtLink href="/infrastructure/pool/1078" method="GET"
        name="DtDesktopPool" rel="self"></DtLink>
      <DtLink href="/infrastructure/pool/1078/delete" method="DELETE"
        name="Delete" rel="action"></DtLink>
      <DtLink href="/infrastructure/pool/1078/update" method="PUT"
        name="Update" rel="action"></DtLink>
      <DtLink href="/infrastructure/pool/1078/patterns" method="GET"
        name="Patterns" rel="association"></DtLink>
      <DtLink href="/infrastructure/pool/1078/vms" method="GET"
        name="VirtualMachines" rel="association"></DtLink>
      <DtLink
        href="/infrastructure/pool/1078/assign/domain/%7BdId%7D/group/%7BgId%7D"
        method="PUT" name="AssignGroup" rel="action"></DtLink>
      <actualSize>0</actualSize>
      <dateCreated>2011-09-02T17:23:57.473Z</dateCreated>
      <DtDesktopModel id="28428506-fce0-43c3-a4be-13043c79b3f5">
        <DtLink href="/infrastructure/model/28428506-fce0-43c3-a4be-13043c79b3f5"
          method="GET" name="DtDesktopModel" rel="self"></DtLink>
```

```

    <defaultProtocol>RDP</defaultProtocol>
    <desktopMemoryInMBs>2048</desktopMemoryInMBs>
    <diskSpaceInGBs>0</diskSpaceInGBs>
    <enabled>true</enabled>
    <name>DebStaticDesktopModel</name>
    <numberOfCpus>2</numberOfCpus>
    <supportedProtocols>ICA RDP VNC</supportedProtocols>
    <type>static</type>
  </DtDesktopModel>
  <domainName>DEV</domainName>
  <DtGoldPattern id="G.1001.62">
    <DtLink href="/infrastructure/pattern/gold/G.1001.62"
      method="GET" name="DtGoldPattern" rel="self"></DtLink>
    <DtLink href="/infrastructure/pattern/gold/G.1001.62/update"
      method="PUT" name="Update" rel="action"></DtLink>
    <DtLink href="/infrastructure/pattern/gold/G.1001.62/convert/desktop"
      method="POST" name="ConvertToDesktop" rel="action"></DtLink>
    <DtLink href="/infrastructure/pattern/gold/G.1001.62/pools"
      method="GET" name="DesktopPools" rel="association"></DtLink>
    <DtLink href="/infrastructure/pattern/gold/G.1001.62/rename?name=%7N%7D&"
      method="PUT" name="Rename" rel="action"></DtLink>
    <DtLink href="/infrastructure/pattern/gold/G.1001.62/vm"
      method="GET" name="VirtualMachine" rel="association"></DtLink>
    <DtLink href="/infrastructure/pattern/gold/G.1001.62/dcs"
      method="GET" name="DataCenters" rel="association"></DtLink>
    <deleted>false</deleted>
    <name>Deb_Gold_Pattern</name>
    <companyName>Desktone</companyName>
    <enabled>true</enabled>
    <key></key>
    <modified>false</modified>
    <notes></notes>
    <osType>Windows7_64</osType>
    <timeZoneId>EST</timeZoneId>
    <username>Administrator</username>
    <vmUuid>4222f9e2-0552-ade8-623b-6f77200599ab</vmUuid>
  </DtGoldPattern>
  <highlyAvailable>false</highlyAvailable>
  <lastUpdated>2011-11-14T20:25:39.937Z</lastUpdated>
  <name>DebUserPool</name>
  <DtOrganization id="1001">
    <DtLink href="/security/organization/1001" method="GET"
      name="DtOrganization" rel="self"></DtLink>
    <DtLink href="/security/organization/1001/quotas" method="GET"
      name="DesktopModelQuotas" rel="association"></DtLink>
    <dateCreated>2011-09-01T20:20:50.216Z</dateCreated>
    <dateUpdated>2011-09-01T20:20:50.216Z</dateUpdated>
    <disabled>false</disabled>
    <name>Tenant1001</name>
  </DtOrganization>
  <patternType>S</patternType>
  <poolModeMessage></poolModeMessage>
  <poolOnline>true</poolOnline>
  <DtPoolPolicy id="2140429207">
    <allocatorSessionTimeout>3600000</allocatorSessionTimeout>
    <allowFullDesktop>false</allowFullDesktop>
    <allowedToJoinDomain>true</allowedToJoinDomain>
    <assignedGroupNames></assignedGroupNames>
    <clipboardRedirect>true</clipboardRedirect>
    <comPortRedirect>false</comPortRedirect>
    <desktopDeallocAction></desktopDeallocAction>
    <drivesRedirect>true</drivesRedirect>
    <maxPoweredOnVMs>50</maxPoweredOnVMs>
  </DtPoolPolicy>

```

```

        <minPoweredOnVMs>10</minPoweredOnVMs>
        <nonPersistent>false</nonPersistent>
        <organizationalUnit></organizationalUnit>
        <printerRedirect>true</printerRedirect>
        <smartCardRedirect>false</smartCardRedirect>
    </DtPoolPolicy>
    <poolSizeType>Fixed</poolSizeType>
    <requestedSize>3</requestedSize>
    <supportedPoolProtocols>RDP</supportedPoolProtocols>
    <vmRootName>DebUserPool</vmRootName>
</DtDesktopPool>
</Entry>
</Map>

```

### 3.7 Example Error

The table below shows the key-value pairs returned in the HTTP response to an incorrectly formulated HTTP request to retrieve non-existent pool information.

**Table 3–2 Sample Error Code in HTTP Response**

Key	Value
x-dt-error-num	1
x-dt-error-code	DT_RESOURCE_NOT_FOUND
X-dt-error-msg	Cannot find desktop pool with id nonexistent

When you receive an HTTP error, the XML returned contains a DtErrorInfo element. For example:

```

<DtErrorInfo>
  <errorCode>DT_RESOURCE_NOT_FOUND</errorCode>
  <errorMessage>Cannot find desktop model with id nonexistent</errorMessage>
  <errorTime>2011-09-29T14:23:07.031-04:00</errorTime>
</DtErrorInfo>

```

## 4 Resources

---

The DtPlatform resource is the entry point for the API, accessed by the URI `/system/platform`. This resource has no links or properties, but instead returns a list of DtVersion resources. Once you login, the first HTTP request you make should access the URI `/system/platform`.

### 4.1 DtVersion

**Scope:** Service Provider and Tenant

#### Links

Name	Description	Method	Relationship	Scope
DtSystemManager	Represents the Horizon DaaS Platform and provides various miscellaneous functionality and information about the current installation	GET	top	BOTH
DtInfrastructureManager	A top level entry point to traverse the Horizon DaaS resources	GET	top	BOTH
DtSecurityManager	A top level entry point to traverse the Horizon DaaS security object model	GET	top	BOTH
DtReportingManager	A top level entry point to retrieve reports	GET	top	BOTH
DtInstallManager	A top level entry point to traverse resources related to the installation of Horizon DaaS appliances	GET	top	SP
DtPoolManager	A top level entry point to traverse desktop or session pool related resources	GET	top	TENANT

#### Properties

Name	Description	Data Type
domainRegistrationURI	URI used by clients to register a domain to the local appliance. This URI can only be used to register the first domain for the organization. Additional domains can be registered using DtSecurityManager's RegisterDomain link	String
loginURI	The URI used by clients to login to the Horizon DaaS Platform with this version	String

## 4.2 DtSystemManager

Represents the Horizon DaaS Platform and provides various miscellaneous functionality and information about the current installation.

Scope: Service Provider and Tenant

### Links

Name	Description	Method	Relationship	Scope
Login	Performs basic authentication and includes the "Authorization" http header in the response that must be included in all http requests to secure resources and all REST API invocations	POST	action	BOTH
Logout	Terminates any session associated with a previous login	DELETE	action	BOTH
PlatformInformation	Retrieves the Horizon DaaS Platform information including the supported versions by the platform	GET	top	BOTH
RegisterDomain	Registers the first domain to the local appliance's organization. The registration would enable authentication checks against the directory service specified in the domain. This call does not require the user to be authenticated as no domain exists at the time of invocation. Domain registration using this method is disallowed if a domain already exists for the organization. In that case, use the authenticated invocation DtSecurityManager's RegisterDomain.	POST	action	BOTH
DtSystemManager	Link to this DtSystemManager	GET	self	BOTH

## 4.3 DtInfrastructureManager

The top level entry point to traverse the Horizon DaaS resources.

Scope: Service Provider and Tenant

### Links

Name	Description	Method	Relationship	Scope
ApplianceTemplate	Assigns a virtual machine as an appliance template Return value - no return value (void)	POST	action	SP
AssignHostManagerToElement	Assign the specified hypervisor manager to a tenant desktop manager (Service Provider only)	POST	action	SP

AssignNetworkToDesktopManager	Assign the specified network to the desktop manager	PUT	action	SP
AssignStorageConfigToElement	Create a storage mount point(datastore) based on the given storage config on all hosts used by a given tenant desktop manager (Service Provider only)	POST	action	SP
CreateDesktopModel	Creates a new desktop model that you will be able to modify later	POST	action	SP
CreateOrUpdateDesktopManagerDMQuota	Sets or updates a collection of quotas for a specified desktop model to the requested amount	POST	action	SP
CreateOrUpdateTemplateQuota	Creates a new template quota or updates an existing template quota (Service Provider only)	POST	action	SP
CreateOrUpdateRemoteProtocolQuota	Creates a new remote protocol quota or updates an existing remote protocol quota (Service Provider only)	POST	action	SP
CreateOrUpdateDesktopModelQuota	Creates new or updates a collection of quotas for desktop models for a tenant organization and datacenter to the requested amounts (Service Provider only)	POST	action	SP
CreateOrUpdateSessionQuota	Creates a new session quota or updates an existing session quota for a given tenant organization and datacenter (Service Provider only)	POST	action	SP
CreateStorageConfig	Creates a new storage mount point on a discovered storage system (Service Provider only)	POST	action	SP
DataCenters	Retrieves a collection of all data centers in the platform	GET	association	BOTH
DesktopModels	Retrieves a collection of all available desktop models	GET	association	BOTH

DesktopModelQuotas	Retrieves a collection of all desktop model quotas	GET	association	BOTH
DiscoverHypervisorManager	Discover a particular hypervisor manager (Service Provider only)	POST	action	SP
DiscoverHypervisorManagerByAddress	Discover a hypervisor manager for the given resource manager by its DNS address Return Value: DtHypervisorManager	POST	action	SP
DiscoverStorageSystem	Discover a storage system (Service Provider only)	POST	action	SP
HypervisorManagers	Retrieves all the hypervisor managers currently discovered in this environment. (Service Provider only)	GET	association	SP
RemoveStorageConfig	Removes a storage mount point configuration in the environment (Service Provider only)	POST	action	SP
RemoveStorageSystem	Removes a storage system from the environment (Service Provider only)	POST	action	SP
RetrieveTemplateQuotaForTenant	Retrieves the template quotas for all datacenters in the tenant organization (Service Provider only)	GET	action	SP
RetrieveAllRemoteProtocolQuotas	Retrieves the remote protocol quotas for all datacenters in the tenant organization (Service Provider only)	GET	action	SP
RetrieveAllStorageConfigs	Retrieves all storage mount point configurations (Service Provider only)	GET	action	SP
RetrieveAllStorageSystems	Retrieves all discovered storage systems (Service Provider only)	GET	action	SP
RetrieveSessionQuotaForTenant	Retrieves the session quota for all datacenters in the tenant organization (Service Provider only)	GET	action	SP

RetrieveStorageConfigsForStorageSystem	Retrieve all storage configs belonging to the specified storage system (Service Provider only)	GET	action	SP
TenantDesktopManagers	Retrieves the collection of tenant desktop managers in the current organization	GET	association	BOTH
TenantNetworks	Retrieves all networks assigned to the given tenant organization. Return Value - Collection of DtNetwork resources	GET	association	SP
DtInfrastructureManager	Link to this DtInfrastructureManager	GET	self	BOTH
UnassignHostManagerFromElement	Remove the assignment of the specified hypervisor manager from the given tenant desktop manager (Service Provider only)	POST	action	SP
UnassignNetworkFromDesktopManager	Unassign the specified network to the desktop manager	PUT	action	SP
UnassignStorageConfigFromElement	Remove the specified datastore config from each host of the given tenant desktop manager (Service Provider only)	POST	action	SP

#### Properties

There are no properties.

### 4.3.1 DtDataCenter

A logical grouping of infrastructure elements, containing information about the server, network, and storage components that are logically grouped to aid administration and provisioning of services.

**Scope:** Service Provider and Tenant

#### Links

Name	Description	Method	Relationship	Scope
DtDataCenter	Link to this DtDataCenter	GET	self	BOTH

**Properties**

Name	Description	Data Type
assignedOrganizationIds	The organizations assigned to this data center	Long
backBoneNetworkId	The backbone network ID that is used for internal communications and management of the data center appliances	String
backBoneNetworkType	The network type of the backbone network (vlan id or network name)	
creationDate	The date of creation of the data center	String
description	Any description associated with this data center or its purpose	String
dnsServer	The IP address of the DNS server of this data center	String
friendlyName	The human readable friendly name	String
gateway	The gateway IP address of this datacenter	String
ipAddressBlock	The IP address block served by this data center	String
lastUpdate	The last update time on this data center	String
mcastAddress	The multicast IP address for this data center	String
mcastClusterName	The multicast cluster name for this data center	Integer
mcastPort	The multicast port of this data center	String
Name	The name of the data center	String
ntpServers	The list of NTP (Network Time Protocol) servers assigned to this data center	String
subnetMask	The subnet mask of this data center	String
frontNetworkId	The front-end network ID of this data center	String
frontNetworkType	The network type of the front-end network (vlan id or network name)	
vmgrId	The virtualization manager for this data center	String
vmgrUIDs	The list of virtualization manager UUIDs of this datacenter	String

**4.3.2 DtDesktopModel**

Provides information about a virtual desktop's hardware and software configuration. A desktop model is assigned to a desktop pool and is used by the pool when it creates virtual desktops.

**Scope:** Service Provider and Tenant

**Links**

Name	Description	Method	Relationship	Scope
DtDesktopModel	Link to this DtDesktopModel	GET	self	BOTH
Save	Persist the properties of this desktop model			

**Properties**

<b>Name</b>	<b>Description</b>	<b>Data Type</b>
currentDesktopCount	The current count of desktops using this desktop model	Long
dateCreated	The date the organization was created	String
defaultProtocol	The default remote access protocol used by desktops using this desktop model, for example RDP	Integer
desktopMemoryInMBs	The RAM size for the desktops using this desktop model	Integer
desktopQuota	The desktop quota of this desktop model	Long
diskSpaceInGBs	The disk space for desktops using this desktop model	Long
Enabled	Specifies if this desktop model is enabled or disabled	Boolean
lastUpdated	The date the organization was last updated	String
licenseEntitlements	The license entitlement of this desktop model	Collection of LicenseEntitlement
name	The name of the desktop model, for example Normal	String
numberOfCpus	The number of CPUs in this desktop model	Integer
refId	The reference ID for the given desktop model	String
supportedProtocols	The list of protocols supported by this desktop model: ICA, RDP, RGX, VNC, NX	Collection of DisplayProtocol
type	The type of the desktop model, for example selectable	String

**4.3.2.1 DtLicenseEntitlement****Properties**

<b>Name</b>	<b>Description</b>	<b>Data Type</b>
dateCreated	The date and time the license entitlement was created	String
desktopModelId	The unique id of the desktop model to which this license entitlement is applied	String
entitlementFileName	The file name of the license entitlement	String
fileLocation	The entitlement file location	String
lastUpdated	The date the desktop model quota was last updated	String
organizationId	The id of the organization to which the entitlement is applied	String
Version	The version of the license entitlement	Float
Deleted	Specifies if this license entitlement has been deleted	Boolean
tenantUpdate	Specifies if this is a tenant update	Boolean

### 4.3.3 DtDesktopModelQuota

Contains information about the number of virtual infrastructure components (like virtual machines) allowed for a tenant.

**Scope:** Service Provider and Tenant

#### Links

There are no links.

#### Properties

Name	Description	Data Type
dataCenterId	The data center id of this desktop model quota	String
dateCreated	The date the desktop model quota was created	String
desktopModelId	The unique id of the desktop model to which this quota is attached	String
lastUpdated	The date the desktop model quota was last updated	String
organizationId	The organization id of this desktop model quota	String
vmCount	The virtual machine count of this desktop model quota	Long

### 4.3.4 DtHypervisorManager

**Scope:** Service Provider

#### Links

Name	Description	Method	Relationship	Scope
AssignToElement	Assign this hypervisor manager to a tenant desktop manager	POST	action	SP
AssignToTenantDesktopManager	Assigns a tenant desktop manager to this hypervisor manager so that compute pools from the hypervisor manager can be assigned to the specified desktop manager. Return Value: DtTenantDesktopManager	POST	action	SP
Delete	A link to delete this hypervisor manager	POST	action	SP
DtHypervisorManager	A link to this DtHypervisorManager	GET	self	SP
Elements	Retrieves all tenant desktop managers using this hypervisor manager	GET	association	SP
ComputePools	Retrieves all compute pools managed by this hypervisor manager	GET	action	SP
UnassignFromElement	Removes assignment of this hypervisor manager to a given tenant desktop manager	POST	action	SP

**Properties**

Name	Description	Data Type
Address	The DNS name or IP address of this hypervisor manager	String
State	The current assignment state of this hypervisor manager (Discovered or Assigned)	String
Status	Current status of the hypervisor (for example, Online)	String
Type	The virtualization technology (for example, vCenter)	String
url	The URL for API invocations	String
vendorId	The unique identifier of the vendor	String

**4.3.5 DtComputePool**

Scope: Service Provider

**Links**

Name	Description	Method	Relationship	Scope
DtComputePool	A link to this DtComputePool	GET	self	SP
HypervisorManager	Retrieves the hypervisor manager that manages this compute pool	GET	association	SP
TenantDesktopManager	Retrieves the tenant desktop manager (element) assigned to use this compute pool, NULL for SP	GET	association	SP
Update	Persists updates to assign or unassign tenant desktop manager or organization to a compute pool. Return Value: no return value (void)	PUT	action	SP
UpdateOverutilizationRatios	Updates the memory and CPU overallocation ratios using the parameters memoryOverallocationRate and cpuRatioVirtualToPhysical in the link's URL	POST	action	SP

**Properties**

Name	Description	Data Type
computeRefId	Unique id that is used by the IaaS layer to uniquely reference the logical/physical instance this compute pool uses. It could be the UUID of an ESX hypervisor host, the unique cluster name or the org VDC name in vCloud	String
cpu	Total number of CPUs offered by the underlying resource of this compute pool	Integer
cpuOverallocation	The ratio of the virtual CPUs (that will be served by this hypervisor in its VMs) to its actual number of physical CPUs	Double

cpuPartition	The number of CPUs offered by this compute pool. If the underlying resource of the compute pool is shared amongst multiple organizations then this would be a subset of the total CPUs of the underlying resource	Integer
dcId	The datacenter in which this compute pool will serve appliances or desktops	String
displayName	The public display name for this compute pool	String
memoryOverallocation	The ratio of the virtual RAM (that will be served by this hypervisor in its VMs) to its actual physical memory	Double
memoryPartitionInMBs	The amount of memory in MBs offered by this compute pool. If the underlying resource of the compute pool is shared amongst multiple organizations then this would be a subset of the total CPUs of the underlying resource	Long
organizationId	Unique identifier of the organization of this compute pool	Long
partitioned	Specifies whether this hypervisor host is partitioned or not	Boolean
properties	A collection of DtComputePoolProperties that represent the variable properties of this compute pool based on the type	DtComputePoolProperty
tenantDesktopManagerId	The unique id of the tenant desktop manager that is using this compute pool, NONE if unassigned	String
type	The type of the hypervisor host, for example esx	String

#### 4.3.5.1 DtComputePoolProperty

**Scope:** Service Provider

##### Links

There are no links.

##### Properties

Name	Description	Data Type
computePoolId	Unique id that is used by the IaaS layer to uniquely reference the logical/physical instance this compute pool uses. It could be the UUID of an ESX hypervisor host, the unique cluster name or the org VDC name in vCloud.	String
name	Name of the compute pool property e.g. status	String
Role	The role that this host is in (active, standby, reserved)	String
Status	The high level status as reported from the AP	String
swDescription	The long description of the software running on this host	String
swVersion	The version of the software running on this host	String
totalCpuCount	The total number of physical CPUs on this host	Integer

value	Property value for this name	String
-------	------------------------------	--------

### 4.3.6 DtNetwork

#### Links

Name	Description	Method	Relationship	Scope
DtNetwork	Specifies if this is a link-local or a backbone network	GET	self	SP

#### Properties

Name	Description	Data Type
datacenterId	Datacenter where this network operates	String
dateCreated	Date network was created	Date
dateUpdated	Date network was last updated	Date
defaultVlan	Specifies if this network is the default network of a tenant organization	Boolean
dnsServer	Address of the DNS server in this network	String
gateway	Address of the gateway for this network	String
networkId	Unique ID of this network (vlan for a network type; vlan and name for a SDN)	String
networkLabel	User-defined network label	String
networkType	DtNetworkType of this network	DtNetworkType
orgId	Organization of this network	Long
privateBackbonePort	Private backbone port if applicable	Long
subnetMask	Subnet mask of this network	String

### 4.3.7 DtRemoteProtocolQuota

Remote protocol quota assigned to tenant clusters (datacenter + organization) on the usage of remote protocols.

**Scope:** Tenant

#### Links

There are no links in this object.

#### Properties

Name	Description	Data Type
dcId	Unique id of the datacenter of this remote protocol quota	String
displayProtocol	Display protocol of this remote protocol quota	DtDisplayProtocol
numVms	The quota amount that applies to this display protocol for the given datacenter and organization	Long

orgId	Unique identifier of the organization of this remote protocol quota	Long
protocolId	Unique identifier of the protocol of this remote protocol quota	Long
Unlimited	Specifies if the quota is unlimited	Boolean

### 4.3.8 DtSessionQuota

Session quota assigned to a tenant cluster (organization + datacenter)

**Scope:** Service Provider and Tenant

#### Links

There are no links in this object.

#### Properties

Name	Description	Data Type
DCId	Unique id of the datacenter of this session quota	String
dateCreated	Creation date this session quota	Date
dateUpdated	Last update of this session quota	Date
orgId	Unique id of the organization of this session quota	Long
Quota	The maximum number of sessions allowed for this tenant cluster	Long

### 4.3.9 DtStorageConfig

A storage mount point configuration upon a discovered storage system.

**Scope:** Service Provider

#### Links

Name	Description	Method	Relationship	Scope
DtStorageConfig	A link to this storage configuration	GET	self	SP

#### Properties

Name	Description	Data Type
localMountPoint	Path to local mount point of the storage configuration on the host	String
name	Name of this mount point on the host	String
ownerId	The unique id of the owner of this storage configuration (e.g. organization id)	Long
remoteMountPoint	Path to remote mount point of the storage configuration on the storage system	String
storageSystemId	The unique id of the storage system where this storage configuration resides	String

tenantDesktopManagerIds	Unique ids of all tenant desktop managers using this storage configuration to create end-user desktops	Collection of String
-------------------------	--	----------------------

### 4.3.10 DtStorageSystem

A storage system discovered and assigned to resource managers. Storage mount points on this storage system persist virtual machines (management appliances or desktops).

**Scope:** Service Provider

#### Links

Name	Description	Method	Relationship	Scope
DtStorageSystem	A link to this storage system instance	GET	self	SP
StorageConfigs	Retrieves all storage mount points configured for this storage system instance for the given resource manager	GET	association	SP

#### Properties

Name	Description	Data Type
address	IP or DNS address of the storage system	String
type	Type of the storage system (Local, NFS, Desktone, Netapp, Nexenta, Isilon, WSS)	String
vendorId	Vendor ID that uniquely identifies the vendor of this storage system	String

### 4.3.11 DtTemplateQuota

Template quota assigned to a tenant cluster (organization + datacenter).

**Scope:** Service Provider and Tenant

#### Links

There are no links in this object.

#### Properties

Name	Description	Data Type
DCId	Unique id of the datacenter of this template quota	String
dateCreated	Creation date of this template quota	Date
dateUpdated	Last update of this template quota	Date
numTemplates	The maximum number of templates allowed for this tenant cluster	Long
orgId	Unique id of the organization of this templates quota	Long

### 4.3.12 DtTenantDesktopManager

An element in the infrastructure that manages tenant desktops.

**Scope:** Service Provider and Tenant

#### Links

There are no links.

#### Properties

Name	Description	Data Type
dataCenterId	The datacenter that contains the tenant desktop manager.	String
dateCreated	The date of creation of the tenant desktop manager.	String
lastMonitorTime	The last time this tenant desktop manager was contacted by the monitoring host.	String
lastUpdated	The last update time on this tenant desktop manager.	String
monitoringHost	The unique id of the host currently monitoring this tenant desktop manager.	String
name	The friendly name of this tenant desktop manager	String
organizationId	The organization id of this tenant desktop manager	Long
state	The current operational state of this tenant desktop manager	String

## 4.4 DtPoolManager

A top level entry point to manage desktop and session pools of tenants and associated resources and actions.

**Scope:** Tenant

#### Links

Name	Description	Method	Relationship	Scope
ConvertToGoldPattern	Converts a static desktop pattern to a gold pattern.	POST	action	Tenant
CreateDesktopPool	Creates a new pool of desktops or sessions (Tenant only)	POST	action	Tenant
DefaultMappingOfUsers	Retrieves the default mappings of a list of users.	GET	map	Tenant
DesktopPoolOfUsers	Retrieves the desktop pools assigned to this user.	GET	map	Tenant
DesktopPools	Retrieves a collection of all available desktop pools.	GET	association	Tenant
Patterns	Retrieves all patterns that belong to the current organization of the specified type. Specify one of the following types of patterns:	GET	association	Tenant

	<ul style="list-style-type: none"> <li>G - A pattern based on a template of the hypervisor.</li> <li>D - A pattern that describes a dynamic virtual desktop machine.</li> <li>S - A pattern that describes a static virtual desktop machine</li> <li>U User - A pattern that describes a non-VM desktop machine.</li> <li>A - A pattern that describes a desktop or application served by an app server that describes a non-VM desktop machine.</li> </ul>			
PatternsOfUsers	Retrieves the pattern mappings of a list of users.	GET	map	Tenant
PurgeRecyclePool	Purges the recycle pool. Static patterns are moved to the recycle pool when a user does not want the virtual desktop any more. The purge recycle pool action goes ahead and deletes virtual desktops in the recycle pool from the hypervisor.	POST	action	Tenant
RefreshDynamicPool	Starts a Dynamic Pool Refresh task for the given gold pattern and dynamic pool	POST	action	Tenant
ReserveDesktopPattern	Reserves a static pattern so it can then become eligible for gold pattern conversion. (Tenant only)	POST	action	Tenant
DtPoolManager	Link to this DtPoolManager.	GET	self	Tenant
VirtualMachine	Retrieves a virtual machine by virtual machine id and pattern id.	GET	association	Tenant
VirtualMachines	Retrieves a collection of all available virtual machines	GET	association	Tenant

#### Properties

There are no properties.

### 4.4.1 DtDesktopPool

A pool of virtual desktops or sessions. Every pool has a gold pattern that serves as the base image to create desktops in the pool. A pool can be static or dynamic - a static pool has a static pattern for each virtual desktop that typically resides on a persistent disk so that user activity is preserved over multiple sessions. A dynamic pool has a single dynamic pattern that is applied to every virtual desktop in the pool typically in a non-persistent disk so every user session resets the virtual machine to its original image.

**Scope:** Tenant

#### Links

Name	Description	Method	Relationship	Scope
DtDesktopPool	A link to this DtDesktopPool.	GET	self	Tenant
AssignGroup	Assigns the specified user group to a desktop/session pool	PUT	action	Tenant

AssignUserToPool	Assigns a specific user to a virtual desktop/session pool	PUT	action	Tenant
CreateRemoteApplication	Creates a new remote application and will assign the remote application to the pool	POST	action	Tenant
Delete	Deletes a pool and any relationships the pool had with other resources	DELETE	action	Tenant
Patterns	Retrieves the patterns in this desktop pools	GET	association	Tenant
RemoteApplications	Retrieves all remote applications configured for this pool	GET	association	Tenant
Update	Updates this desktop pool and any new relationships this pool has with other resources	PUT	action	Tenant
VirtualMachines	Retrieves the Virtual Machines in this pool	GET	association	Tenant

### Properties

Name	Description	Data Type
actualSize	The actual size of the pool.	Long
customerId	Customer id for the pool, should only be used in conjunction with super tenant feature.	String
dateCreated	The date the pool was created, for example 2011-09-02T17:23:57.473Z	String
DesktopModel	The DesktopModel of this pool that specifies various VM parameters for the virtual desktops in the pool.	DtDesktopModel
domainName	The security domain name for the desktops in this pool, for example DEV.	String
GoldPattern	The gold pattern used by this desktop pool.	DtGoldPattern
highlyAvailable	Specifies if the pool has high availability. True or false.	Boolean
lastUpdated	The date the pool was last updated, for example 2011-10-17T17:57:21.173Z	String
minimumAvailable	The minimum number of desktops that need to be started and available for login.	Long
Name	The desktop pool name.	String
Organization	The organization of this desktop pool.	DtOrganization
patternType	The pattern type of this pool, this determines if the pool is a static pool or a dynamic pool. S=Static, D=Dynamic.	DtPatternType
poolModeMessage	The Pool Mode Message describes about the Pool why the pool is Online or Offline.	String
poolOnline	Specifies if the pool is online or offline.	Boolean
PoolPolicy	The policy settings and configurations used to allocate, initialize and manage this pool.	DtPoolPolicy
poolSizeType	The pool size type, for example Fixed.	DtPoolSizeType

requestedSize	The requested size of the pool.	Long
sessionBased	Specifies if the pool is a desktop or a session pool. True or false.	Boolean
supportedPoolProtocols	The list of remote access protocols that can be used to access virtual desktops from this pool, for example RDP.	List of DtDisplayProtocol
usedByDataCenterIds	A list of data center IDs where this pool can have desktop instances.	String
VirtualMachines	The Virtual Machines in this desktop pool.	Collection of DtVirtualMachine
vmRootName	The base name for all virtual machines provisioned from this pool.	String

#### 4.4.1.1 DtPoolPolicy

Configuration and initial setup options for desktop pools. Each desktop pool has one pool policy and is used for provisioning and pool management operations.

**Scope:** Tenant

##### Properties

Name	Description	Data Type
AllocatorSessionTimeOut	The session timeout for allocators in the pool using this policy.	Long
AllowedToJoinDomain	Specifies the desktop pool is allowed to join a security domain.	Boolean
AllowFullDesktop	Connecting to the full desktop is allowed from clients (in addition to remote applications) for virtual desktops in the pool using this policy.	Boolean
AssignedGroupNames	The domain group names assigned to the pool with this policy.	Collection of String
AssignedNetworks	Tenant networks (network ids) assigned to this pool. All desktops in the pool will be provisioned to these networks.	Collection of String
ComPortRedirect	Specifies remote access settings have enabled mapping client's local com ports to the virtual desktop.	Boolean
DesktopDeallocAction	The action to be taken after a desktop is de-allocated from the pool (reboot, no action, etc.)	String
DrivesRedirect	Specifies remote access settings have enabled mapping client's local drives to the virtual desktop.	Boolean
LicenseKey	The license key for the desktop pool that uses this policy.	String
MaxPoweredOnVMs	The maximum number of virtual machines that can be PoweredOn state in the pool that has this policy.	Integer
MinPoweredOnVMs	The minimum number of virtual machines that must be PoweredOn in the pool that has this policy.	Integer

NonPersistent	Specifies the virtual desktops in the pool using this policy use non-persistent disks.	Boolean
OrganizationalUnit	The AD organization unit of the pool using this policy.	String
PrinterRedirect	Specifies remote access settings have enabled mapping client's printers to the virtual desktop.	Boolean
RegisteredTo	Used to populate the registered owner and registered organization properties of a windows desktop in that pool.	String
SmartCardRedirect	Specifies remote access settings have enabled mapping client's smart cards to the virtual desktop.	Boolean
Timezone	The time zone of the pool that uses this policy.	String

#### 4.4.2 DtDynamicDesktopPattern

A virtual desktop that does not have specific user assignments. Any authorized user can use the virtual desktops from a pool of desktops.

**Scope:** Tenant

##### Links

Name	Description	Method	Relationship	Scope
DataCenters	Retrieves the data centers of this pattern.	GET	association	Tenant
Rename	Assigns a name to the pattern as would be known in the hypervisor (e.g. the VM name or the template name).	PUT	action	Tenant
DtDynamicDesktopPattern	A link to this DtDynamicDesktopPattern	GET	self	Tenant
VirtualMachine	Retrieves the virtual machine associated with this pattern.	GET	association	Tenant

##### Properties

Name	Description	Data Type
dataCenterName	The name of the data center where this pattern is available.	String
dateCreated	The date the pattern was created.	String
deleted	Specifies if this pattern is deleted.	Boolean
name	The name of the pattern as known by the hypervisor (e.g. the VM name or the template name).	String
dependantPoolIds	A collection of pool ids where the pattern is being currently used.	Collection of String
desktopPoolId	The id of the desktop pool of this desktop pattern.	String
lastUpdated	The last update time and date.	String
numberOfInstances	The number of instances of this desktop pattern at this location.	Integer

previousDesktopPoolId	The pool id of this pattern prior to migration to another pool.	String
templateId	The templateId of the pattern.	String

### 4.4.3 DtGoldPattern

A reserved virtual machine that can be cloned to create virtual desktops. A DtGoldPattern is created by reserving a virtual desktop. DtGoldPatterns are assigned to pools that create virtual desktops from them depending upon the desktop pool properties and desktop model

**Scope:** Tenant

#### Links

Name	Description	Method	Relationship	Scope
DataCenters	Retrieves the data centers of this pattern.	GET	association	Tenant
DtGoldPattern	A link to this DtGoldPattern.	GET	self	Tenant
Rename	Assigns a name to the pattern as would be known in the hypervisor (e.g. the VM name or the template name).	PUT	action	Tenant
VirtualMachine	Retrieves the virtual machine associated with this pattern.	GET	association	Tenant

#### Properties

Name	Description	Data Type
dataCenterName	The name of the data center where this pattern is available.	String
dateCreated	The date the pattern was created.	String
deleted	Specifies if this pattern is deleted.	Boolean
dependantPoolIds	A collection of pool ids where the pattern is being currently used.	Collection of String
desktopPools	The desktop pools using this gold pattern.	Collection of DtDesktopPool
hostManagerId	The host manager's ID.	String
inventoryPath	The inventory path of the gold pattern.	String
name	The name of the pattern as known by the hypervisor (e.g. the VM name or the template name).	String
companyName	The company name.	String
enabled	Specifies if this pattern can be used for pool creation.	Boolean
key	The windows key for the virtual machine referenced by the pattern.	String
lastUpdated	The last update time and date.	String
md5sum	The MD5 checksum of this gold pattern.	String
modified	Specifies if the pattern has been modified.	Boolean

notes	The notes associated with this gold pattern.	String
osType	The operating system type of this gold pattern.	String
parentId	The ID of the parent pattern from which this gold pattern is derived.	String
templateId	The templateId of the pattern.	String
timeZoneId	The time zone id of this gold pattern.	String
username	The local admin username.	String
vmUuid	The UUID of the VM referenced by this pattern.	String

#### 4.4.4 DtPoolTask

An asynchronous task performed on a pool. Contains information about the task, the targets, the progress of the task, and the result after the task completes.

**Scope:** Tenant

##### Links

Name	Description	Method	Relationship	Scope
Refresh	Refreshes the task's status with latest information about the progress of the task.	POST	action	Tenant
DtPoolTask	A link to this DtPoolTask.	GET	self	Tenant

##### Properties

Name	Description	Data Type
DesktopPoolId	The pool that the task acts upon.	String
PercentageComplete	The percentage of the task that is complete.	Int
Status	The current status of this pool task.	DtTaskStatus
StatusDescription	The description of the current state, including reasons for failures	String
Type	The type of the pool task	String

#### 4.4.5 DtRemoteApplication

**Scope:** Tenant

Remote application configured and served in a desktop or session pool

##### Links

Name	Description	Method	Relationship	Scope
Delete	Deletes this remote application	DELETE	action	Tenant
DtRemoteApplication	A link to this remote application	GET	self	Tenant
Update	Updates this remote application with the current properties	PUT	action	Tenant

**Properties**

Name	Description	Data Type
commandLine	Command to launch this remote application using CLI	String
iconLocation	Location of the icon that would represent this remote application. Limitation: may not be longer than 2048 characters	String
Name	Name of this remote application	String
poolId	Unique identifier of the desktop or session pool that serves this remote application	Long
primaryFileLocation	Location of the primary executable file	String

**4.4.6 DtStaticDesktopPattern**

A virtual desktop with static user assignments. The virtual desktop is available only to users explicitly assigned to it.

**Scope:** Tenant

**Links**

Name	Description	Method	Relationship	Scope
AssignUserToDesktopPattern	Assign a specific user to a static virtual desktop	PUT	action	Tenant
AssignedUsers	Retrieves all users in the given domain that have been assigned the use of this static desktop pattern.	GET	action	Tenant
DataCenters	Retrieves the data centers of this pattern.	GET	association	Tenant
Rename	Assigns a name to the pattern as would be known in the hypervisor (e.g. the VM name or the template name).	PUT	action	Tenant
DtStaticDesktopPattern	A link to this DtStaticDesktopPattern.	GET	Self	Tenant
VirtualMachine	Retrieves the virtual machine associated with this pattern.	GET	association	Tenant

**Properties**

Name	Description	Data Type
dataCenterName	The name of the data center where this pattern is available.	String
dateCreated	The date the pattern was created.	String
deleted	Specifies if this pattern is deleted.	Boolean
name	The name of the pattern as known by the hypervisor (e.g. the VM name or the template name).	String

dependantPoolIds	A collection of pool ids where the pattern is being currently used.	Collection of String
desktopPoolId	The id of the desktop pool of this desktop pattern.	String
hostmanagerId	The host manager's ID.	String
modified	Specifies if the pattern has been modified.	Boolean
lastUpdated	The last update time and date.	String
osType	The operating system type of this pattern.	String
previousDesktopPoolId	The pool id of this pattern prior to migration to another pool.	String
templateId	The templateId of the pattern.	String
vmUuid	The UUID of the VM referenced by this pattern.	String

#### 4.4.7 DtVirtualMachine

An active virtual machine used by the end-user as a virtual desktop or virtual server. It provides information about a virtual machine as it is used on a hypervisor.

**Scope:** Tenant

##### Links

Name	Description	Method	Relationship	Scope
PerformOperation	Executes the specified power operation upon this virtual machine.	POST	action	Tenant
DtVirtualMachine	A link to this DtVirtualMachine.	GET	self	Tenant

##### Properties

Name	Description	Data Type
description	The description of the VM.	String
inventoryPath	The inventory path of the VM.	String
ipAddress	The VM IP address.	String
name	The VM name.	String
numCPUs	The number of CPUs.	Int
patternId	The pattern ID.	String
refId	The VM reference id.	String
templateId	The ID of the template.	String
type	The VM type.	String
vmId	The VM ID.	String
vmLifeState	The VM life state.	DtVMLifeState
vmPath	The VM path.	String
vmPowerState	The VM power state.	DtVMPowerState

## 4.5 DtInstallManager

A top level entry point to manage all management appliances and associated resources and actions.

**Scope:** Service Provider

### Links

Name	Description	Method	Relationship	Scope
AllAppliances	Retrieves a collection of all appliances managed by the service provider organization	GET	association	SP
DtInstallManager	A link to this DtInstallManager.	GET	self	SP

### Properties

There are no properties.

### 4.5.1 DtAppliance

A management appliance.

**Scope:** Service Provider

### Links

Name	Description	Method	Relationship	Scope
DtAppliance	A link to this appliance	GET	self	SP
Restore	Restores the management appliance by deleting it and recreating and reinstalling it. (CAUTION - all snapshots of the appliance are lost if restore is performed)	POST	action	SP
ResourceManagerId	Retrieves the unique id of the resource manager that this appliance uses to interface with computing resources.	GET	action	SP

### Properties

Name	Description	Data Type
capabilities	Bit-wise value representing all capabilities of this management appliance( possible values are SP_FABRIC = 0, TRANSIT_SERVER=1, VMGR=2, APPLIANCE_TEMPLATE=3, TENANT_FABRIC=4, EMGR=5, PRIMARY_NODE=6, PRIMARY_NODE_ACROSS_DCS=7)	Long
datacenterId	Unique Id of the datacenter where this management appliance resides	String
displayName	Management appliance's human readable display name	String
computePoolId	Unique Id of the compute pool that hosts this management appliance	String
lastMonitorTime	Time when the management appliance was last monitored	Date

lifeState	Life state of this management appliance (Online, Offline, Unknown)	String
monitoringHost	Unique id of the monitoring host of this management appliance	String
name	Management appliance's name	String
numMissedHeartbeats	Number of missed heartbeats from this management appliance	Integer
organizationId	Unique Id of the organization that this management appliance serves	String
state	State of this appliance (Reserved, Created, Installed, Disabled, Deleted)	String
version	Version of this management appliance	String
virtualMachineId	Unique id of the virtual machine in the hypervisor host that operates this management appliance	String

## 4.6 DtReportingManager

A top level entry point to manage reporting features in the Horizon DaaS Platform.

**Scope:** Service Provider and Tenant

### Links

Name	Description	Method	Relationship	Scope
AvailableReportDates	Retrieves all available report dates for tenant organization (Tenant only)	GET	association	Tenant
DtReportingManager	A link to this DtReportingManager resource	GET	self	BOTH
CurrentReportDate	Retrieves the active cycle's report date in format yyyyMM	GET	action	BOTH
LastCompletedReportDate	Retrieves last month's report date in format yyyyMM	GET	action	BOTH
TenantReports	Retrieves all tenant reports filtered by the given report filter	POST	association	BOTH
CreateBillingReportFilter	Creates an empty DtBillingReportFilter to retrieve billing reports	GET	action	BOTH
CreateConcurrentUsersFilter	Creates a new filter to retrieve concurrent users report	GET	action	SP
CreateReportFilter	Creates a generic Report Filter	GET	action	BOTH
CreateReportKey	Creates a generic Report Key	GET	action	BOTH
CreateUserEventReportFilter	Creates a filter to help retrieve specific groups of user events	POST	action	BOTH
MaxConcurrentUsersReport	Retrieves the maximum concurrent users for a given DtConcurrentUsersFilter	POST	association	SP

QuotaBillingReports	Retrieves a list of the latest quota billing reports based on the given DtBillingReportFilter	POST	association	SP
SuperTenantBillingReports	Retrieves a list of super tenant billing reports based on the given DtBillingReportFilter	POST	association	BOTH
UserEventReports	Retrieves user event reports based on a given UserEventReportFilter	POST	action	BOTH

**Properties**

There are no properties.

**4.6.1 DtBillingReportFilter**

A filter object that specifies the criterion to retrieve billing reports.

**Scope:** Service Provider and Tenant

**Links**

There are no links in this object.

**Properties**

Name	Description	Data Type
customerIds	List of customer ids to restrict the results while retrieving billing reports. This must be used only in conjunction with super tenants. The customer ids are URL encoded. (Super Tenant only)	Collection of Strings
organizationIds	List of organization ids to restrict the results while retrieving billing reports. (Service Provider only)	Collection of Longs

**4.6.2 DtConcurrentUsersFilter**

A filter object that specifies the criterion to retrieve concurrent users.

**Scope:** Service Provider

**Links**

There are no links in this object.

**Properties**

Name	Description	Data Type
beginDate	The start date to filter concurrent users when retrieving reports within a time-window. The date should be in the format - "2013-06-14T12:00:00.000Z"	Date
endDate	The end date to filter concurrent users when retrieving reports within a time-window. The date should be in the format - "2013-06-14T12:00:00.000Z"	Date

### 4.6.3 DtConcurrentUsersReport

A report of the maximum number of concurrent users in a given time window.

**Scope:** Service Provider

#### Links

There are no links in this object.

#### Properties

Name	Description	Data Type
maxConcurrentUsersCount	The count of the maximum number of users concurrently connected	Long
tenantExceptions	A comma-separated string of tenant organization ids that were not reachable at the time of computing the max concurrent users count	String

### 4.6.4 DtQuotaBillingReport

Service provider report that provides billing information related to quotas (desktop model quota, session quota, protocol quota and template quota).

**Scope:** Service Provider

#### Links

There are no links in this object.

#### Properties

Name	Description	Data Type
datacenterId	datacenter ID pertaining to this report	String
disabled	Specifies if the organization is currently disabled	Boolean
errorReport	Specifies if this is an error report	Boolean
inUseCount	The count of the amount of quota in use by the organization	Long
organizationId	Organization ID of the super tenant	Long
protocolType	The protocol type of this report, available only if the quotaType is DtQuotaBillingType.PROTOCOL	DtDisplayProtocol
quota	Assigned quota to the organization for this quota type, -1 indicates unlimited quota	Long
quotaId	The ID of the quota associated with this billing record, if applicable	String
quotaType	The type of the quota for this record	DtQuotaBillingType
refId	Reference ID for this report	String
snapshotId	Retrieves the snapshot ID for this report. It is formatted as "YYYYMMddhhmm"	String

### 4.6.5 DtReportFilter

A filter object that specifies the criterion to retrieve tenant reports.

**Scope:** Service Provider and Tenant

#### Links

There are no links in this object.

#### Properties

Name	Description	Data Type
reportKeys	A collection of DtTenantReportKey instances that defines the filter	Collection of DtTenantReportKey

### 4.6.6 DtSuperTenantBillingReport

Collects billing data for super tenants by their customer ids.

**Scope:** Service Provider and Tenant

#### Links

There are no links in this object.

#### Properties

Name	Description	Data Type
customerId	Sub-tenant customer ID pertaining to this record	String
desktopCount	List of desktop model to the in-use count of those desktop models by this customer in a super tenant. Count for each desktop model is wrapped within DtDesktopCountWrapper instances.	Collection of DtDesktopCountWrapper
organizationId	Organization ID of the super tenant	Long
sessionCount	Count of the number of sessions allocated to this customer	Long

### 4.6.7 DtTenantReport

Tenant report for billing purposes. Each instance contains information about a single month's activities.

**Scope:** Service Provider and Tenant

#### Links

Name	Description	Method	Relationship	Scope
DtTenantReport	A link to this report	GET	self	BOTH

#### Properties

Name	Description	Data Type
desktopLoginCount	Number of logins per (VDI) desktop	Long

desktopRemoteAppLoginCount	Number of remote application logins serviced by (VDI) desktops	Long
reportKey	ReportKey containing the organization and the date of the report	DtTenantReportKey
sessionLoginCount	Number of logins to session based desktops	Long
sessionRemoteAppLoginCount	Number of remote application logins served by session based desktop	Long

#### 4.6.8 DtTenantReportKey

Specifies the organization id and report date used to filter tenant reports.

**Scope:** Service Provider and Tenant

##### Links

There are no links in this object.

##### Properties

Name	Description	Data Type
orgId	Unique id of an organization	Long
reportDate	The report cycle month for a report in the format yyyyMM	Integer

#### 4.6.9 DtUserEventReport

Contains data pertaining to a user event. See DtUserEvent for all supported event types.

**Scope:** Service Provider and Tenant

##### Links

Name	Description	Method	Relationship	Scope
DtUserEventReport	A link to this user event report	GET	self	BOTH

##### Properties

Name	Description	Data Type
datacenterName	Name of the datacenter where the event originated	String
displayProtocol	Display protocol associated with the event	DtDisplayProtocol
endpointIPAddress	IP address of the endpoint that originated this event	String
endpointPlatformType	Endpoint type	DtEndpointPlatformType
eventError	Error during the event, if any	DtUserEventError
eventErrorDetails	Details about the event error, if any	String
patternId	Pattern Id associated with this event	String
poolName	Name of the pool associated with the event	String
remoteAppName	Name of remote application associated with the event	String

reportTimestamp	Timestamp of the report	Date
userEvent	Event type	DtUserEvent
userGuid	GUID of the user associated with the event	String
virtualMachineId	Virtual machine associated with the event	String
vMSessionId	The virtual machine session ID of this event, or 'unavailable' if there's none	String

#### 4.6.10 DtUserEventReportFilter

A filter object that specifies the criterion to retrieve user event reports.

**Scope:** Service Provider and Tenant

##### Links

There are no links in this object.

##### Properties

Name	Description	Data Type
displayProtocol	The display protocol to filter in the event reports	DtDisplayProtocol
eventTypes	List of DtUserEvent types to filter	Collection of DtUserEvent
numberReportDays	The number of report days from the current day (inclusive) to fetch. One day is interpreted as 24 hours from the current date and time.	Integer
patternId	The pattern Id to filter	String
remoteAppName	Name of the remote application to filter	String
userGuid	The GUID of the user to filter	String
virtualMachineId	The virtual machine's UUID to filter	String

## 4.7 DtSecurityManager

A top level entry point to traverse the Horizon DaaS security object model.

**Scope:** Service Provider and Tenant

##### Links

Name	Description	Method	Relationship	Scope
Domains	Retrieves a collection of security domains.	GET	association	BOTH
Organizations	Retrieves collection of all available organizations.	GET	association	BOTH

RegisterDomain	Registers a new domain to the local appliance's organization. The registration would enable authentication checks against the directory service specified in the domain. The first domain must be registered using the DtSystemManager's RegisterDomain link	POST	action	BOTH
UserById	Retrieves A link to this DtSecurityManager a DtUser with a given GUID	GET	action	BOTH
DtSecurityManager		GET	self	BOTH

### Properties

There are no properties.

## 4.7.1 DtDomain

A security domain.

**Scope:** Service Provider and Tenant

### Links

Name	Description	Method	Relationship	Scope
DtDomain	A link to this Domain	GET	self	BOTH
Groups	Retrieves groups in this domain	GET	association	BOTH
Save	Persist the properties of this domain		action	BOTH
Users	Retrieves the user account names registered in this domain	GET	association	BOTH
UserByLoginName	Retrieves the user registered to this domain with the given login name	GET	action	BOTH

### Properties

Name	Description	Data Type
adminGroups	A collection of groups with administrative privileges.	Collection of DtGroup
contextRoot	The context root of the directory server associated with this domain.	String
dateCreated	The date the domain was created domain.	String
directoryAccessPort	The directory server access port for this domain.	String
directoryAccessProtocol	The directory access protocol of this domain.	String
directoryServerIps	A list of directory server IPs used by this domain	Collection of String
directoryServerName	The directory server name of this domain	String
dnsServers	The list of DNS servers for this domain	String
domainSuffix	The domain suffix	String

lastUpdated	The date the domain was last updated	String
maxQueryRange	The maximum number of attributes returned when doing an LDAP query in this domain	String
name	The name of the Domain	String
serviceAccounts	A collection of service accounts of the domain on the directory server	Collection of DtDomainAccount
sysPrepAccounts	A collection of accounts used to prepare images for virtual desktop creation	Collection of DtDomainAccount
userGroups	A collection of groups with user privileges	Collection of DtGroup

#### 4.7.1.1 DtDomainAccount

An account used by a domain, containing information, such as unique identifier and the password.

Scope: Service Provider and Tenant

##### Properties

Name	Description	Data Type
type	The type of this domain account	DtDomainAccountType

#### 4.7.1.2 DtGroup

A security group with a role that can be assigned to users to give them privileges.

Scope: Service Provider and Tenant

##### Links

Name	Description	Method	Relationship	Scope
assignPool	Assigns this group to a desktop pool.	PUT	action	Tenant
DtGroup	A link to this DtGroup.	GET	self	BOTH
removePool	Removes a desktop pool assignment from this group.	PUT	action	Tenant

##### Properties

Name	Description	Data Type
domainName	The container security domain name of this group, for example DEV.	String
name	The name of the group, for example cn=enterprise admins,cn=users	String
role	A set of permissions or privileges that can be assigned to a group. There is a one-one association between a group and a role.	DtRole

#### 4.7.1.3 DtRole

A set of permissions or privileges that can be assigned to a group.

**Scope:** Service Provider and Tenant

##### Properties

Name	Description	Data Type
creationDate	The date of creation of the role.	String
lastUpdate	The date of the last update to the role.	String
name	The name of the role.	String
permissions	The permissions that a group of users possess when this role is assigned to the group.	String

#### 4.7.1.4 DtUser

A security domain user. A user or its group is typically assigned to a desktop pool that serves the virtual desktop for the user. However, a user may or may not have an active session. And a user may or may not have an associated virtual machine.

**Scope:** Service Provider and Tenant

##### Links

Name	Description	Method	Relationship	Scope
assignPool	Assigns this user to a desktop pool.	PUT	action	Tenant
assignStaticDesktop	Assigns a virtual desktop to this user.	PUT	action	Tenant
defaultMapping	Retrieves this user's default desktop pattern or pool.	GET	association	Tenant
defaultMapping	Assigns the default pattern or pool to be used for desktops associated with this user.	PUT	action	Tenant
displayProtocol	Retrieves the display protocol preference of this group for the given desktop pattern.	GET	action	Tenant
displayProtocol	Assigns the given protocol as a preferred display protocol for this user.	PUT	action	Tenant
desktopPatterns	Retrieves the desktop patterns assigned to this user.	GET	association	Tenant
desktopPools	Retrieves the desktop pools assigned to this user.	GET	association	Tenant
removePool	Removes a desktop pool assignment from this user.	PUT	action	Tenant
removeStaticDesktop	Removes a virtual desktop assigned to this user.	PUT	action	Tenant

updateStaticDesktopAssignment	Updates the static desktop mapping of a user from an existing assignment to a new static desktop.	PUT	action	Tenant
DtUser	A link to this DtUser.	GET	self	BOTH

**Properties**

Name	Description	Data Type
domainName	The security domain of the user, for example DEV	String
loginName	The name used by the user to login.	String
userDn	The LDAP distinguished name of the user.	String

## 4.7.2 DtOrganization

A completely separate business entity with its own disjoint set of resources.

**Scope:** Service Provider and Tenant

**Links**

Name	Description	Method	Relationship	Scope
DesktopModelQuotas	A list of desktop model quotas available to this organization	GET	association	BOTH
DtOrganization	A link to this DtOrganization	GET	self	BOTH

**Properties**

Name	Description	Data Type
adminContact	The administrative contact of this organization. Consists of email address, name, and phone number.	DtContact
brandedCSSURL	The URL to the CSS file used for this specific organization.	String
businessUrl	The business URL of this organization.	String
crmURL	The Customer Relationship Management URL of this organization.	String
customFields	All custom fields in a key-value pair specific to this organization.	Map of String
dateCreated	The date the organization was created, for example 2011-09-01T20:04:27.207Z	String
dateUpdated	The date the organization was last updated, for example 2011-09-01T20:04:27.207Z	String
disabled	Specifies if this organization is disabled	Boolean
diskQuota	The disk quota of this organization.	Integer
helpDesk	The help desk contact of this organization	DtContact
jmxPassword	The password used to use the dt-console JMX application for this organization	String

licenseFileURL	The URL to the license file used for this specific organization.	String
name	The name of this organization.	String
superTenant	Specifies if the organization is a super tenant	Boolean
support	The support contact of this organization	DtContact
supportURL	The support URL of this organization.	String
technicalContact	The technical contact of this organization.	DtContact
vmQuota	The virtual machine quota for this organization.	Integer

#### 4.7.2.1 DtContact

Contact details for various functions in an organization, such as admin and support.

**Scope:** Service Provider and Tenant

##### Properties

Name	Description	Data Type
cellPhone	The contact name	String
email	The contact email address	String
name	The contact name	String
phone	The contact phone	String

## 5 Enumerated Data Types

---

This chapter lists the enumerated types that are the legal values for certain resource attributes.

### 5.1 DtComputePoolType

Represents the list of compute pool types that can be discovered by the Horizon DaaS Platform

#### Properties

Name	Description
ESX	Direct ESX hypervisor host
VCENTER_CLUSTER	vCenter cluster
VCLLOUD_VDC	vCloud virtual datacenter

### 5.2 DtDisplayProtocol

Represents the list of display protocols that can be used to access virtual desktops from remote clients.

#### Properties

Name	Description
ICA	Citrix's Independent Computing Architecture
NX	The unique id of the desktop model to which this license entitlement is applied.
RDP	Remote desktop protocol
RGS	
VNC	Virtual network computing
PCOIP	Teradici's PC-over-IP
OTHER	
UNKNOWN	

## 5.3 DtDomainAccountType

Specifies the domain's account type. Value is one of the following:

- SERVICE
- SYS\_PREP

## 5.4 DtEndpointPlatformType

Represents the platform types of the endpoint (end-user) device used to access a virtual desktop or remote application. Value is one of the following:

- ANDROID
- IOS
- LINUX
- MAC
- OTHER\_MOBILE
- OTHER\_NON\_MOBILE
- THIN\_CLIENT
- UBUNTU
- UNKNOWN
- WINDOWS

## 5.5 DtNetworkType

Indicates the type of network. Value is one of the following:

- vlan - vlan network
- sdn - software defined network

## 5.6 DtPatternType

Specifies the types of patterns.

### Properties

Name	Description
A	(App) A pattern that describes a desktop or application served by an app server that describes a non-VM desktop machine.
D	(Dynamic) A pattern that describes a dynamic virtual desktop machine.
G	(Gold) A pattern based off of a template from the hypervisor.
S	(Static) A pattern that describes a static virtual desktop machine.
U	U (User) A pattern that describes a non-VM desktop machine.

## 5.7 DtPoolSizeType

Specifies the types of pool sizes. Value is one of the following:

- Elastic
- Fixed
- Test

## 5.8 DtQuotaBillingType

Specifies the types of quota that can be used for billing reports. Value is one of the following:

- DESKTOP\_MODEL
- SESSION
- TEMPLATE
- PROTOCOL

## 5.9 DtTaskStatus

Specifies the current status of this pool task. Value is one of the following:

- FAILED
- OTHER
- RUNNING
- SUCCESSFUL

## 5.10 DtUserEvent

Specifies the user event types that are reported by the platform. Values are described below.

Value	Description
ALLOCATE	VM or RDS session is allocated to a user successfully
ALLOCATE_FAILURE	VM or RDS session fails to be allocated to a user
DISCONNECT	User disconnects from a VM or RDS session
LOG_OFF	User logs off of a VM or RDS session
LOG_ON	User logs on to a VM or RDS session
RECONNECT	User reconnects to an existing VM or RDS session
SESSION_TIMEOUT	VM or RDS session times out due to lack of activity
UNKNOWN	

## 5.11 DtUserEventError

Specifies the type of user event errors that can occur when desktop allocation fails. Values are described below.

Value	Description
BROKER_PARAM_ERROR	Internal error when a parameter required for allocation is incorrect or missing
CONNECTION_TYPE_MISMATCH	Session to a full desktop or a remote application currently exists and attempt has been made to connect with the other type (full desktop or remote application)
FAILED_TO_LOCK_VM	VM for allocation could not be locked for exclusive access
DESKTOP_UNAVAILABLE	VM or RDS session cannot be allocated because either no VMs in a pool are available or an unrecognized error occurred
DESKTOP_NOT_RUNNING	VM for allocation was not powered on
GUEST_OS_NOT_RUNNING	Operating system of the VM is not running
VM_TOOLS_NOT_INSTALLED	VM tools have not been installed on the VM for allocation
VM_TOOLS_NOT_RUNNING	VM tools are not running on the VM for allocation
IP_ADDRESS_UNKNOWN	IP address of the VM to be allocated is not known or has not been reported by the hypervisor
IP_ADDRESS_UNREACHABLE	IP address of the VM to be allocated is not reachable
VM_STATE_ERROR	Allocation state of the VM is not in the required state (generally, "AVAILABLE")
AGENT_STATE_ERROR	DaaS Agent of the VM is not in the required state ("ACTIVE")
INCOMPATIBLE_PROTOCOL	Reconnect to an existing VM or RDS session and requested protocol is not compatible with the one previously used to establish the session. User must be logged off of the existing session before proceeding with the new session.
PCoIP_AGENT_ERROR	PCoIP agent is not active on the VM or there was an error with the PCoIP gateway on the VM
RAM_SERVICE_FAILURE	RAM service failure is when a desktop cannot be reached through the dtRAM appliance

## 5.12 DtVMLifeState

Specifies the various states of the virtual machine. Value is one of the following:

- CLONE\_FAILED
- CLONING
- DESTROYING
- JOINING\_DOMAIN
- NOT\_IN\_DOMAIN
- OFF
- READY

- STARTING
- UNKNOWN

## 5.13 DtVMPowerState

Specifies the states of power for a VM. Value is one of the following:

- POWER\_STATE\_POWERING\_OFF
- POWER\_STATE\_POWERING\_ON
- POWER\_STATE\_RESETTING
- POWER\_STATE\_SUSPENDING
- POWERED\_OFF
- POWERED\_ON
- SUSPENDED
- SYS\_PREP