



## SUN VIRTUAL DESKTOP ACCESS KIT FOR VMWARE SUN DESKTOP VIRTUALIZATION SOLUTION

Dirk Grobler, Sun Ray™ Engineering  
Warren Ponder, VMware Desktop Technical Marketing

Sun BluePrints™ OnLine — March 2007

Part No 820-1325-10  
Revision 1.0, 3/8/07  
Edition: March 2007

© 2006, 2007 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, CA 95054 USA

All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California.

Sun, Sun Microsystems, Solaris, OpenSolaris, the Sun logo, Sun Ray, and Sun Ray Connector for Windows are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries.

VMware is a registered trademark and VMware Virtual SMP, VMotion, VMware Virtual SMP, VMware High Availability (HA), VMware Infrastructure 3 (VI3), and VMware Distributed Resource Scheduler, are trademarks of VMware.

UNIX is a registered trademark in the United States and other countries, exclusively licensed through X/Open Company, Ltd.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a). DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS HELD TO BE LEGALLY INVALID.



Please  
Recycle



Adobe PostScript

## Table of Contents

Introduction . . . . .	2
Scope . . . . .	3
<b>Virtual Desktop Access Kit Architecture Overview . . . . .</b>	<b>5</b>
Virtual Desktop Access Kit Features . . . . .	5
Virtual Desktop Access Kit Components . . . . .	5
<b>Virtual Desktop Access Kit Concepts . . . . .</b>	<b>7</b>
One Desktop from Anywhere . . . . .	7
Static Versus Dynamic . . . . .	7
Resource Optimization . . . . .	9
<b>Virtual Desktop Access Kit Components . . . . .</b>	<b>10</b>
VDA Client . . . . .	10
VDA Service . . . . .	11
VDA Tools . . . . .	11
VDA CAM Script for Sun Ray Server Software . . . . .	12
VDA Session Script for Sun Secure Global Desktop . . . . .	14
<b>Planning and Preparing Virtual Center . . . . .</b>	<b>15</b>
Virtual Desktop Golden Image . . . . .	16
Dynamic or Static Virtual Desktops . . . . .	17
Virtual Desktop Pools . . . . .	17
Virtual Desktop Factory . . . . .	17
VDA Service Configuration . . . . .	19
<b>Conclusion . . . . .</b>	<b>19</b>
<b>About the Authors . . . . .</b>	<b>20</b>
<b>Acknowledgments . . . . .</b>	<b>20</b>
<b>Important Links . . . . .</b>	<b>21</b>
<b>Glossary . . . . .</b>	<b>22</b>
<b>Appendix A: VDA Service Configuration Example (vda.properties) . . . . .</b>	<b>23</b>

## Introduction

As virtualization technology, which improves flexibility and mobility both for users and for IT organizations, has made its way into desktop architectures, there has been an increasing need for corresponding software tools and management capabilities.

A properly designed desktop virtualization solution should allow users to access their desktop environments from an assortment of devices, such as PCs, thin clients, and compatible web browsers. The Sun Desktop Virtualization solution consolidates desktop operating system instances into the data center, using remote display protocols on a wide array of devices and network access methods to present them to users. Secure hosting and management of virtual desktop instances within the data center make it easier to offer remote user access that is both local and secure. Secure remote access to virtual desktops can thus be delivered without compromising network security or increasing the management overhead of the desktop environment.

Sun's Desktop Virtualization solution enables customers to leverage the value of thin clients while minimizing the initial investment and high cost that is sometimes associated with migrating from a distributed, fat client-based solution. Customers often want to move to a thin client-based solution but struggle with the effort and cost of transition. Desktop Virtualization can help these customers to leverage the value of Sun's Virtual Display Client technology by moving desktop instances off physical personal computers and consolidating virtual versions of them onto a server-based solution using virtualization technology.

A solution based on Virtual Desktop Infrastructure (VDI) puts IT organizations in a position to develop and implement new, long-term desktop strategies, such as migrating to an Open Solaris™ or Linux or other open operating system, delivering more services through a web service-based architecture, migrating to a server-based computing architecture, or a hybrid strategy that leverages components of several architectures. In some cases, massive, all-inclusive desktop migrations and change in existing processes can introduce significant risk. Where reduced risk is necessary, a phased adoption approach that begins with desktop virtualization represents a comfortably smaller step, providing administration and cost advantages, in addition to minimizing risk, because the desktop operating system and application infrastructure remains the unchanged. In later phases, the customer is well positioned to implement a new desktop strategy for specific business needs while preserving the investment in thin client technology and Virtual Desktop Infrastructure.

With a Sun-based virtualization solution, all services are managed from within the data center. There is no configuration, operating system, or data to manage on the client device. This offers a more secure and manageable solution than traditional PCs or thin clients that use an embedded operating system. Lower total cost of ownership can be

achieved through more efficient management, increased productivity, and reduced power consumption.

The Sun Virtual Desktop Access Kit for VMware is an add-on component that integrates with Sun Desktop Infrastructure products and VMware Infrastructure 3 (VI3). In this context VI3 is the infrastructure to manage and execute virtual desktops. This is also known as Virtual Desktop Infrastructure (VDI). The Virtual Desktop Access Kit (VDA Kit) is the integration software between the access and virtualization tiers of the Sun Desktop Virtualization Solution. The integration between Sun's Desktop Infrastructure products and VMware's Virtual Infrastructure enables delivery of a dynamic and robust end-to-end desktop virtualization solution.

This document outlines the feature set and value of the Sun Virtual Desktop Access Kit for VMware, discussing how it integrates with the access and virtualization tiers of the Sun Desktop Virtualization Solution. A reference architecture is explained through a number of examples.

## Scope

This document describes the capabilities of the Sun Virtual Desktop Access Kit for VMware and how it can be integrated with VMware's Virtual Infrastructure 3 to deliver a dynamic and robust desktop virtualization solution. In order to gain a full understanding of how the Virtual Desktop Access Kit works and the value it brings to the Sun Desktop Virtualization Solution, the reader should have an understanding of the following:

- Sun Ray™ Software 4 or higher
- Sun Secure Global Desktop 4.2 or higher
- VMware ESX 3.01 or higher
- VMware Virtual Center 2.x
- VMware HA
- VMware DRS

This paper is not intended to replace the installation and administration guides for any of these technologies. The focus is on applied concepts and best practices for the Virtual Desktop Access Kit as a way to maximize the value you can gain from this VDI solution. The Sun Desktop Virtualization Solution white paper is also recommended as a starting point and to better understand the Sun Desktop Virtualization Solution. It is available at:

[http://www.sun.com/software/sdis/  
wp\\_desktop\\_virtualization\\_blueprint.pdf](http://www.sun.com/software/sdis/wp_desktop_virtualization_blueprint.pdf)

The initial version of the Virtual Desktop Access Kit concentrates on delivering Windows

XP Professional or Windows 2000 Professional virtual desktops to the user. Other kinds of virtual desktops may be targeted in later versions of the Virtual Desktop Access Kit but are currently out of scope.

## Virtual Desktop Access Kit Architecture Overview

The Virtual Desktop Access Kit is the software that integrates the Access Tier and Desktop Virtualization Tier components. Built by leveraging VMware's Virtual Infrastructure SDK, its function is to manage the life-cycle of the virtual desktop instances.

### Virtual Desktop Access Kit Features

- Seamless connection to virtual desktops through Sun Ray Server Software (SRSS) and Sun Secure Global Desktop (SSGD)
- Session redirection between SRSS and SSGD
- Static and dynamic assignment of virtual desktops to users
- Lifecycle management of virtual desktops available from multiple pools
- On-demand creation of virtual desktops from golden images
- Support of Windows system preparation (*sysprep*) and joining new virtual desktops to Windows domains
- Automatic recycling of virtual desktops to a pre-used state after use
- Increased hardware utilization when using VMware Dynamic Resource Scheduler
- Automatic desktop restart when using VMware HA
- Support for Power Management of virtual desktops

### Virtual Desktop Access Kit Components

- Reference Controlled Access Mode (CAM) script for SRSS
- Reference session script for SSGD
- VDA Client for access tier servers such as SRSS or SSGD
- VDA Service for Virtual Center
- VDA Tools, a helper to be installed onto the Guest OS

The VDA Access Kit can be deployed with or without VMware's Virtual Center as a central point of control. The initial version of the VDA Kit for VMware requires Virtual Center; however, later versions might support a deployment without requiring Virtual Center, although such a solution will only support a limited feature set. See Figure 1 for an overview of the Virtual Desktop Access Kit architecture.

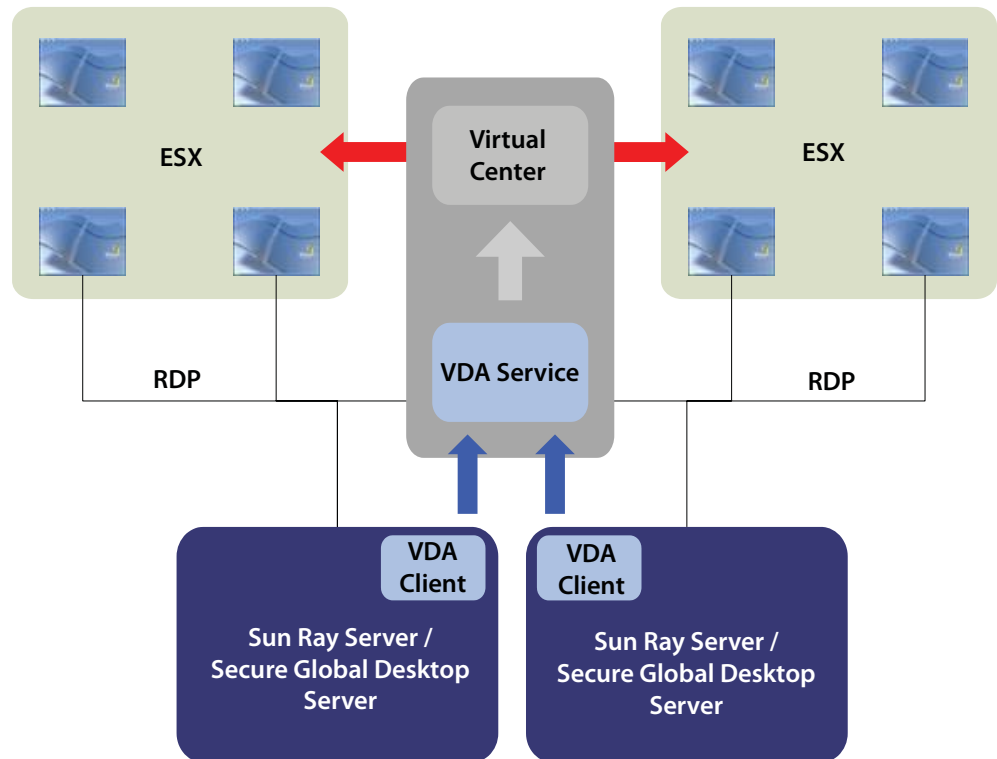


Figure 1. Architecture of the Sun Virtual Desktop Access Kit for VMware

The Virtual Desktop Access Kit is implemented using two core components, the VDA Client and the VDA Service. The VDA Client is installed on the Sun Ray Server or Secure Global Desktop Server that resides within the access tier. The VDA Client requests a running virtual desktop instance from the VDA Service, using a unique identifier, such as a user ID. Once the VDA Service identifies the correct virtual desktop instance, it returns an IP address identifying the requested virtual desktop. This information is then used by the access tier to establish an RDP connection to the given IP address. Communication between the VDA Client and VDA Service is handled using simple TCP/IP communication.

The VDA Service is a Java application installed on the Virtual Center server. It uses the Virtual Infrastructure SDK to interact with the Virtual Center. The VDA Service provides the control logic used to locate virtual desktops based on unique identifiers. In addition, for dynamic deployments where pooling is used, it handles creation of new virtual desktops on demand as well as starting and stopping virtual desktops when required. When contacted by a VDA Client, the VDA Service decides which virtual desktop to select for the given identifier. Once identified, the virtual desktop is started, if required. When the virtual desktop is fully up and running, which means RDP communication can be initiated, the IP address of the virtual desktop is handed over to the requesting VDA Client.



## Virtual Desktop Access Kit Concepts

The Virtual Desktop Access Kit for VMware does not introduce new management interfaces for virtual desktops, it simply leverages existing functionality delivered through the VMware Virtual Infrastructure SDK to deliver an end-to-end desktop virtualization solution.

This section focuses on the control logic, which is implemented mainly through the VDA Service. Understanding the ideas and concepts behind this logic will help to assess the available functionality and additional value of this VDI solution.

### One Desktop from Anywhere

The idea behind the Virtual Desktop Access Kit is to deliver a single desktop instance to the user for the duration of the current network connection. The user should be able to access a virtual desktop through a Sun Ray Desktop Unit (DTU) by providing a smart card identifier (token) or user ID. Accessing the virtual desktop through a browser, using Sun Secure Global Desktop from outside the company's firewall, should provide exactly the same virtual desktop previously accessed from the Sun Ray DTU. Although access methods may differ, the virtual desktop should always be the same.

### Static Versus Dynamic

Within an enterprise, it is common to have different types of user profiles. These user profiles are often categorized as power workers, knowledge workers, task workers, and mobile workers. Power workers, such as developers, typically require a large set of applications and more administrative rights than other users, and they are usually responsible for maintaining their own desktops. These users are candidates for their own virtual desktop instance, much like having their own PCs. This relationship between user and virtual desktop is often referred to as a static relationship. Such virtual desktops are typically named after a unique user ID and placed into a static virtual desktop pool along with other static virtual desktops. A virtual desktop pool is represented in Virtual Center by a folder containing virtual desktops. Placing a virtual desktop into a static pool and naming it after a certain user type guarantees that the user is always directed to this particular virtual desktop instance, regardless of the access method.

Other types of users, such as transactional workers or knowledge workers, have fewer requirements for customizing their desktop environment. For these users, a dynamic virtual desktop might be more appropriate. The dynamic virtual desktop is created from a standard golden image, or template, customized using the Windows system preparation tool (`sysprep`) and preserved in a dynamic virtual desktop pool.

Virtual desktops assigned from a dynamic virtual desktop pool are only temporarily

available to each user. If the user does not use the currently assigned virtual desktop instance or is idle for a pre-defined amount of time, the virtual desktop instance is recycled. When a virtual desktop instance from a pool is recycled, it is returned to its prior state before being checked out. Virtual desktops are recycled back to their prior state with VMware snapshot technology. See Figure 2 for a sample screenshot that illustrates the implementation of virtual desktop pools.

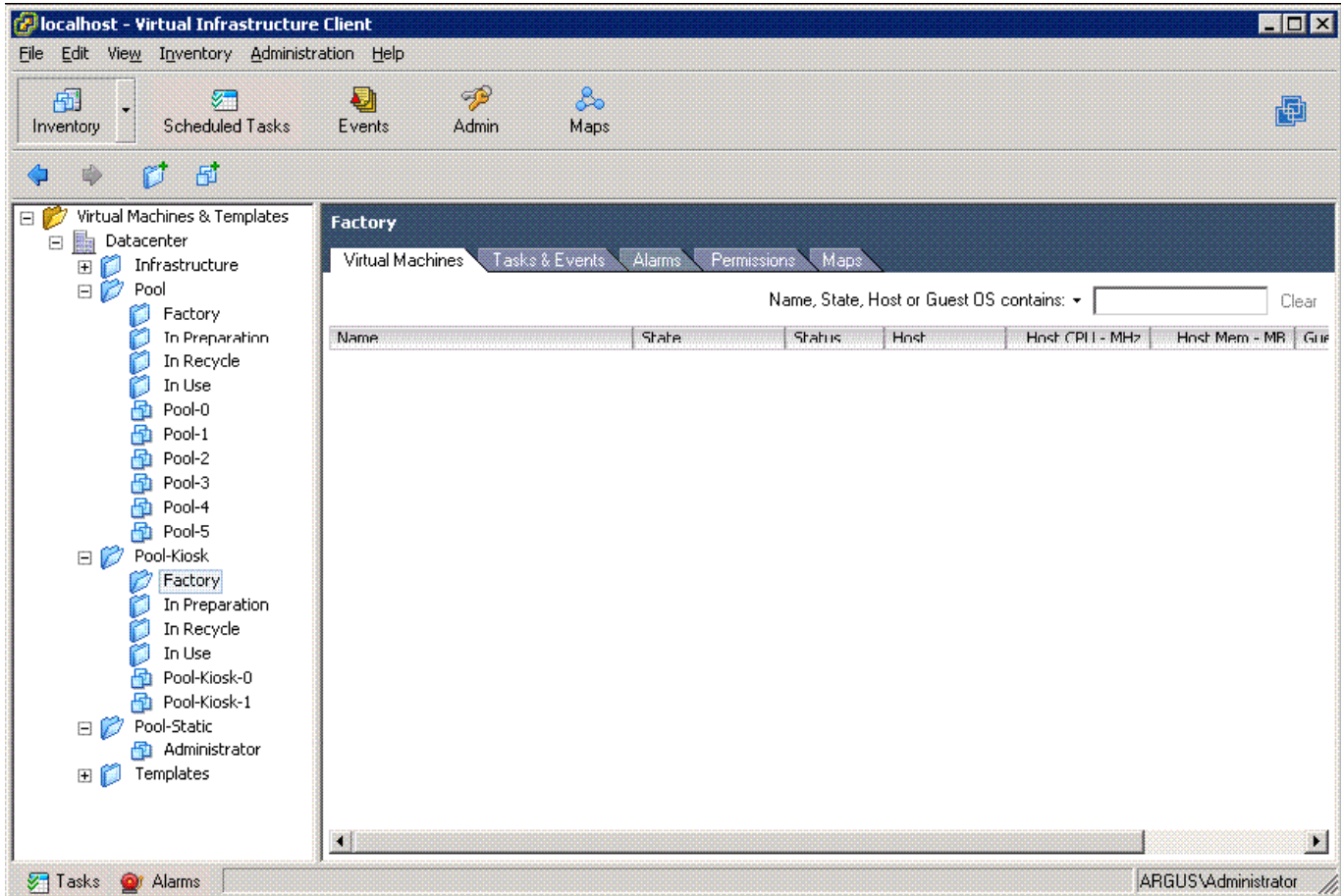


Figure 2. Virtual Desktop Pools in VMware Virtual Center

On the left side of the screenshot in Figure 2 is the hierarchy of folders available in a particular virtual data center. The folder called Pool-Static is the static virtual desktop pool. There is only one static virtual desktop pool per data center. Currently, it contains a single virtual desktop called Administrator. This is the static virtual desktop of the user who logs in as Administrator. Whenever a user logs in, the VDA Service first checks the static virtual desktop pool. In the case of the Administrator user, it finds a virtual desktop matching the given identifier. The virtual desktop is started, if necessary, and its IP address returned to the VDA Client.

All other users are assigned a virtual desktop from the dynamic virtual desktop pool

called Pool or Pool-Kiosk. A dynamic virtual desktop pool can be easily identified by its sub-folders, which are automatically created by the VDA Service. The folder called In Preparation is used for starting up a newly requested virtual desktop. Virtual desktops located in the In Recycle folder are scheduled to be converted back to a point in time, in preparation for the next requesting user or device. The In Use folder contains virtual desktops that are currently assigned to users and or devices. The Factory folder is a specific folder used during the automatic creation of new virtual desktops. Once these virtual desktops are cloned and assigned, they automatically move to the In Use folder.

IT organizations should try to minimize the number of golden images because of the effort required to manage and maintain a large collection of golden images in a consistent state. Access rights, desktop settings, and different sets of applications can all be configured with tools such as Microsoft Group Policy or application provisioning tools such as Sun Secure Global Desktop. In some cases, however, there may still be a need to deliver virtual desktops that vary in size, in terms of memory, disk space, etc.

For example, a knowledge worker may require a virtual desktop that is allocated more memory than one assigned to a transactional worker. In such cases, it might be necessary to operate with different dynamic virtual desktop pools. Each dynamic pool would be characterized by its number of elements, expiration policies for the virtual desktops, and the golden images used to create new virtual desktops.

The VDA Service is always configured so that at least one default dynamic pool is defined. The default dynamic pool serves every user or device that is not assigned to a static pool or specific dynamic pool. In Figure 2 above, there are two dynamic virtual desktop pools, Pool and Pool-Kiosk. Pool is the default dynamic pool. Pool-Kiosk is assigned to a specific use case, such as a contract worker or visitor who would access a virtual desktop temporarily from an Internet Kiosk. In the example illustrated, a user accessing a virtual desktop via a smart card is directed to the default dynamic pool. A user not using a smart card—assuming non smart card access has been enabled—is assigned a virtual desktop from the pool named Pool-Kiosk. The latter example is the real kiosk use case, where a different configuration of a golden image and virtual machine configuration, such as less memory, can be used. The association of a pool to a user or device is handled outside of the VDA Service, except in the case of the static pool. The VDA Kit includes a CAM script for Sun Ray Server Software that handles both smart card and non-smart card usage.

## Resource Optimization

The Virtual Desktop Access Kit supports the VMware Dynamic Resource Scheduling (DRS) feature, which is part of Virtual infrastructure 3. When the VDA Service needs to start a virtual desktop, it contacts Virtual Center to find out which ESX server is best suited to host the virtual desktop, based on the resource policies that have been

defined. Virtual Center notifies the VDA Service and the virtual desktop is started on the recommended ESX server.

In addition to DRS, the Virtual Desktop Access Kit also resolves issues associated with unused or idle virtual desktops. Virtual desktops that are idle or not being used consume memory and CPU resources that should be available for utilization of active virtual desktops. The Virtual Desktop Access Kit handles idle desktops by implementing a power management scheme for Windows XP. Desktop administrators are asked to turn on the power management features of Windows XP for the virtual desktops. This is easily handled by adding the power management scheme provided by the Virtual Desktop Access Kit and enabling power management in the golden master images. Virtual desktops should behave like laptops that go into a standby mode or suspend after a defined period of time in order to save power. The same principle applies to the virtual desktops, with the focus on saving CPU cycles and memory consumption. In the VDI scenario, the power scheme can be configured to define how long an idle virtual desktop is allowed to run idle or unused before it should be suspended.

When a virtual desktop switches into standby mode, the power management scheme notifies the ESX host, which then suspends the virtual desktop, freeing up the CPU and memory currently being used by the virtual desktop for use by other virtual desktops hosted on the server. Static virtual desktops stay in the suspended mode until the assigned user requests the virtual desktop again. However, dynamic virtual desktops are made available for other users either immediately after suspension or after a defined grace period. If the grace period passes, dynamic virtual desktops revert to the state before they were assigned to the current user and are then made available again for others users. In the context of the Virtual Desktop Access Kit, this functionality is called recycling.

## Virtual Desktop Access Kit Components

The Virtual Desktop Access Kit components that work with the Sun Desktop Infrastructure products and VMware Virtual Infrastructure each play an important role in enabling the communication between all elements of the architecture.

### VDA Client

The VDA Client is an application installed on the Sun Ray server or Secure Global Desktop server. The VDA Client application handles communication with the VDA Service running on the Virtual Center server. The VDA Client requests access to virtual desktops for a given unique identifier, such as a user ID, a token ID from a smart card, or a pseudo ID from a CAM session. Optionally, it may provide information about the specific dynamic virtual desktop pool used for provisioning virtual desktops. The VDA

Client returns the IP address of the identified virtual desktop to the Sun Ray Server Software or Secure Global Desktop, depending on where the connection is initiated.

## VDA Service

The VDA Service is a Java application installed on the Virtual Center server as a Windows Service. It uses the VMware Virtual Infrastructure SDK to communicate with Virtual Center. The communication takes place locally on the Virtual Center server. The VDA Service encompasses the control logic of the VDI solution. The purpose is to select and prepare virtual desktops to be assigned to Sun Ray DTUs. This process is initiated through requests from the VDA Clients. The VDA Client passes a unique identifier such as a pseudo ID, smart card token ID, or user ID, to the VDA Service. A recommended dynamic pool name can also be passed with the identifier. The VDA Service handles the request by first looking into the static pool for a virtual desktop whose name matches the identifier. If this is unsuccessful, it looks into the recommended pool, if this information was part of the request. If all previous lookups fail, it uses the default dynamic pool to assign a virtual desktop.

Within a dynamic pool, the VDA Service first tries to match a running virtual desktop with the given unique identifier. If this is successful, the IP address of the running virtual desktop is returned to the caller. If there is no running virtual desktop matching the identifier, the next available virtual desktop is selected. Once selected, the virtual desktop is started and named after the given identifier to match subsequent requests. Once the virtual desktop is up and running, its IP address is returned to the VDA Client. In this context, it should also be noted that virtual desktops are automatically resumed as part of the lookup process. If the virtual desktop was previously suspended due to power management operations, it will be resumed.

Additionally, the VDA Service handles the recycling of unused or idle virtual desktops still assigned to users. Virtual desktops that are identified for recycling are reverted to the state prior to being checked out and are made available to other users or devices. The VDA Service can also trigger the creation of new virtual desktops within a dynamic pool, as when the number of available virtual desktops falls below a pre-defined minimum. In such a case, a clone operation of the defined golden image is triggered, creating a new virtual desktop instance. Once cloning is completed, a system preparation of the virtual desktop is initiated, which includes joining the Windows domain. After being joined to the Windows domain, the virtual desktop is made available to be assigned to users or devices.

## VDA Tools

VDA Tools is a service that is installed on the Windows virtual desktop instance to provide robust handling of power management. If a power management scheme is applied that puts the virtual desktop into standby mode after a certain amount of idle

time, the VDA Tools service makes sure the existing RDP clients disconnect safely and instantaneously. It also notifies the VDA Service running on the Virtual Center about the changed state of the virtual desktop instance.

## VDA CAM Script for Sun Ray Server Software

Sun Ray Server Software uses the VDA CAM script for the virtual desktop session handling. The VDA CAM script can handle both scenarios that do or do not use smart cards. When smart cards are used, the Sun Ray DTU displays a Windows virtual desktop instance looked up based on the smart card token or associated user ID. The virtual desktop instance can either be statically assigned or served from a dynamic pool. When smart cards are not used, the Sun Ray DTU displays a Windows virtual desktop based on the pseudo ID of the Sun Ray session. In this case, there is no up-front personalization, which means no user ID can be preset for the Windows login screen. The whole workflow for the session initialization is illustrated in Figure 3.

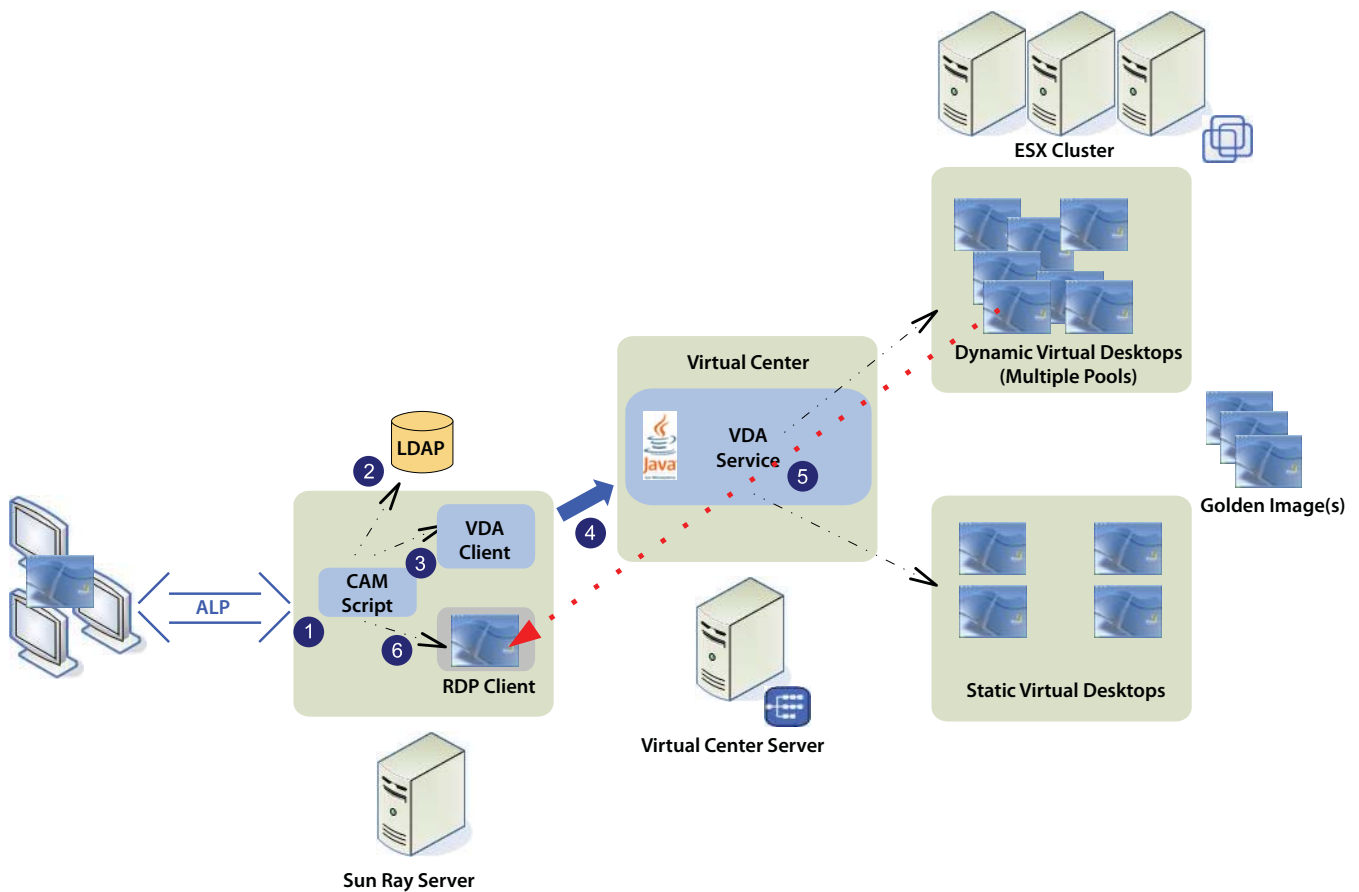


Figure 3. Virtual Desktop Access for Sun Ray

1. A new user inserts a smart card into the Sun Ray DTU, starting a new Sun Ray session. The VDA CAM script is invoked.

2. The VDA CAM script tries to resolve the smart card token ID into a user ID by contacting the Sun Ray local data store. (This could also be done through a lookup to any external database, such as an LDAP directory).
3. The VDA Client is invoked to start the lookup process for a virtual desktop, using either the user ID or the token ID.
4. The VDA Client passes the unique ID information to the VDA Service.
5. The VDA Service then looks for a matching virtual desktop, first in the static pool, then in a dynamic pool. Once the virtual desktop is selected, running, and ready to accept RDP connections, the IP address is returned.
6. The VDA CAM script invokes the Sun Ray Windows Connector with the returned IP address, and the assigned session is displayed on the Sun Ray DTU.

## VDA Session Script for Sun Secure Global Desktop

The VDA Session script for SSGD plays a nearly identical role to that played by the VDA CAM script. The VDA Session script can be configured into an application definition for SSGD. Once this is done, privileged users can launch a virtual desktop from a Webtop session. The exact configuration is described as part of the instructions for installing and configuring the VDA Session script for SSGD included with the Virtual Desktop Access Kit. The session instantiation process is similar to the one for Sun Ray, as illustrated in Figure 4.

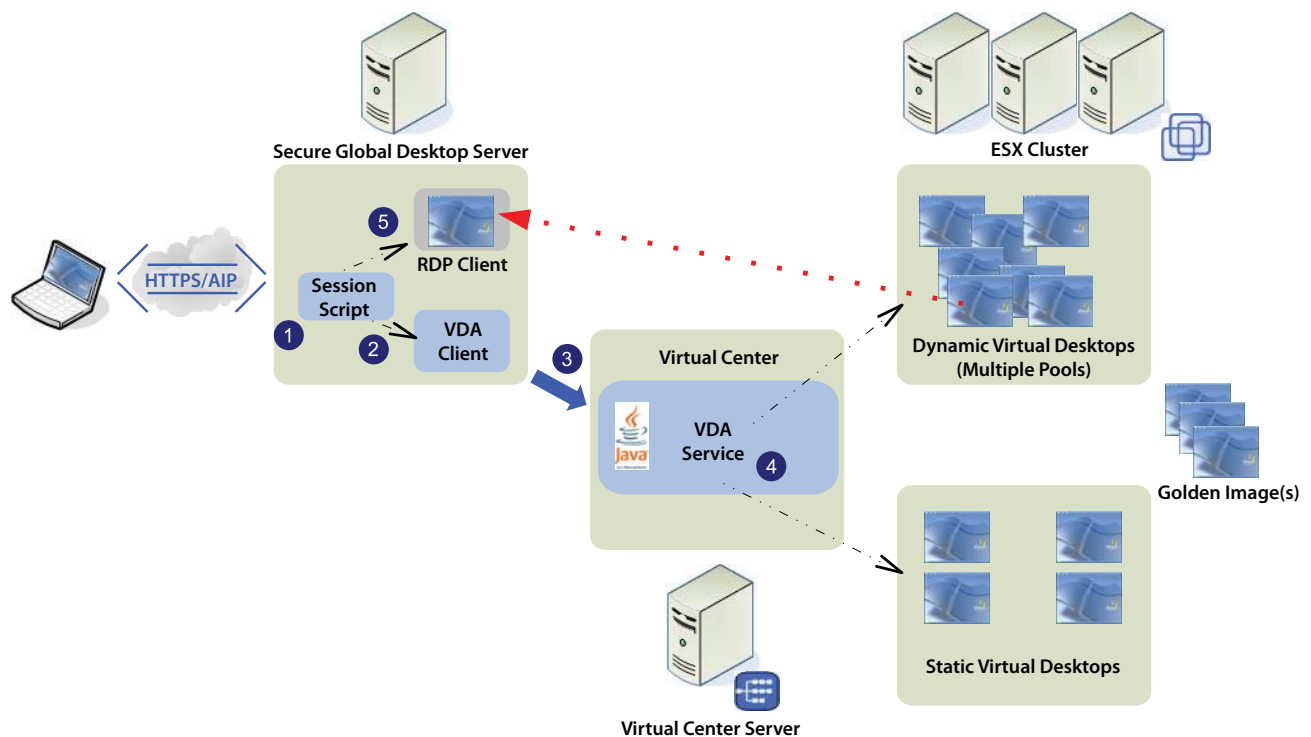


Figure 4. Virtual Desktop Access for Sun Secure Global Desktop

1. A user logs into the SSGD Webtop through a browser on any client device from outside the firewall. After successful authentication, the user is presented with a list of application launchers, one of which is configured for a virtual desktop. When the user invokes the specific launcher, the VDA Session script is triggered.
2. The VDA Client is invoked with a given user ID.

From this point forward, the process is identical to the Sun Ray scenario above. The final step differs in that the SGD RDP client is invoked with the IP address of the assigned virtual desktop, and the virtual desktop session is displayed in the browser of the invoking client device.

In the case where both SRSS and SSGD are provisioned to the user base, users can access a single virtual desktop instance from a Sun Ray DTU or a browser using another



client device, independent of whether virtual desktops are provisioned statically or dynamically. The only prerequisite is that the same identifiers be used when users login to their virtual desktop through SRSS or SSGD. The lookup method for both systems is identical. When the Virtual Desktop Access Kit is used, the resulting Windows session will always be the same one. This offers users the benefit of starting a Windows session from a Sun Ray DTU within the enterprise firewall, and then, when traveling or working from home, picking up the exact same session in a browser without having to re-login or to close any active sessions (provided, of course, that they use a static desktop or that their dynamic desktop was not recycled). The entire session redirection process is transparent to the user—a clear advantage. Figure 5 depicts this user experience.

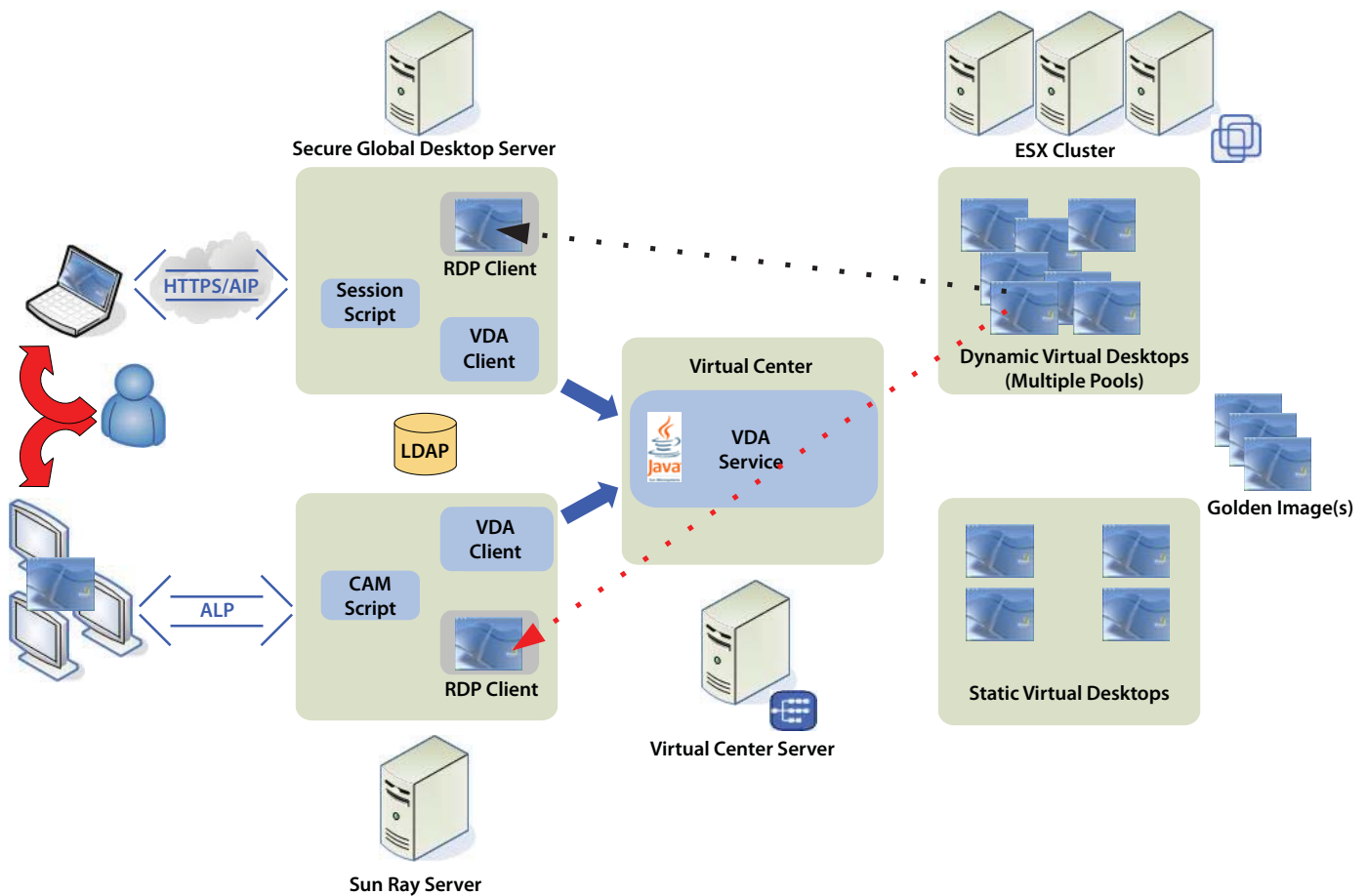


Figure 5. Seamless Access from Anywhere Through SRSS and SSGD

## Planning and Preparing Virtual Center

Before configuring Virtual Center, an administrator should understand the back-end Virtual Center infrastructure and the role each component plays in the workflow of virtual desktop life-cycle management. Some key points include the preparation of the

golden image, the folders required in Virtual Center, and the creation process and naming conventions. Of course, there is more to know about how to set up the Virtual Desktop Infrastructure; however, specific VMware Infrastructure 3 administration expertise and best practices are beyond the scope of this paper.

## Virtual Desktop Golden Image

The virtual desktop golden image is a virtual machine template used to seed the virtual desktop inventory. Depending on needs, there might be only a single golden image, but there could easily be more if user requirements are more complex. The configuration of the image does not vary much compared to traditional desktop images. It requires the installation of the Windows OS, all of the local applications, patches, and standard IT configurations. There are a few specific tasks that also need to be taken into account for the Sun Desktop Virtualization Solution:

- **VDA Tools**  
VDA Tools need to be installed on the golden image for robust handling of power events such as a virtual desktop suspend.
- **Power Options**  
Power Options should be configured to reduce the overall system load. Power Options can be found in the Windows XP Control Panel. Here you have to set the standby time to a suitable value. This is a machine setting which can only be set by the administrator of the virtual desktop. Controlling this setting for each individual box can be quite tedious and error-prone, so using Group Policy would be ideal, but no such GPO settings for Power Options are offered by default. Some third-party solutions, such as [http://www.terravivum.com/projects/energystar/ez\\_gpo.html](http://www.terravivum.com/projects/energystar/ez_gpo.html), are available free of charge. Windows Vista will include a Group Policy for Power Options.
- **Virtual Machine Configuration**  
It is important to make sure the virtual desktop suspends when the Guest OS initiates a standby. This will fully suspend the virtual machine and not keep the virtual desktop awake. These settings can be found under Options/Power Management.

Once the golden image (template) is created, it needs constant maintenance, especially if the update and patch process is to be maintained through frequent replacements of the virtual desktop instances instead of reliance on the Windows System update components typically used in traditional desktop deployments. There is an excellent document published by VMware, that explains the best practices for templates in detail:

[http://www.vmware.com/pdf/vc\\_2\\_templates\\_usage\\_best\\_practices\\_wp.pdf](http://www.vmware.com/pdf/vc_2_templates_usage_best_practices_wp.pdf).

## Dynamic or Static Virtual Desktops

Virtual desktops can either be static or dynamic. Static virtual desktops are assigned specifically to a user. When assigned a static virtual desktop, a user is always connected to that desktop and exclusively uses that virtual desktop until migrated to a new desktop.

Dynamic virtual desktops are created dynamically and assigned upon request to users or virtual display clients connecting to the network in order to deliver dedicated virtual desktops for temporary, short-term use. After a user has finished using a dynamically assigned virtual desktop, the desktop reverts to a clean state and returned to the pool, ready to be assigned to the next user or device. All the user data, home directories, profiles, etc. for dynamic virtual desktops should be stored on the network and be available regardless of which virtual desktop is assigned to the user. In other words, the whole desktop personalization should be served from the network.

## Virtual Desktop Pools

Pools are logical collections of dynamic or static virtual desktops organized in folders. Static virtual desktops are grouped into a single static pool, while dynamic virtual desktops can be grouped into multiple pools. The number of pools depends on the complexity of the desktop environment. Some workgroups may have different virtual desktop requirements for memory, disk, application load, etc. Pools need to be populated with virtual desktops that are ready to be assigned to users or devices. Virtual desktops can be created manually by an administrator or automated using a factory (see Figure 6). Dynamic pools also have a configured threshold, which defines the minimum number of virtual desktops that should be available at all times.

## Virtual Desktop Factory

The Virtual Desktop Access Kit makes it possible, through the concept of factories, to automate the creation of new virtual desktop instances within each pool. As existing virtual desktops are checked out for use by clients, the factory automatically creates new virtual desktop instances. Factories can share a common golden image or have their own specific golden image. When a new virtual desktop instance needs to be created for a pool, the factory clones a new virtual desktop instance from the assigned golden image. Once completed, the virtual desktop instance is powered on, run through Windows system preparation (if configured), and sent to join the Active Directory domain (again if configured). The VDA Service then takes a snapshot to serve as a baseline for future virtual desktops and, finally, suspends virtual desktop instance. Once completed, the virtual desktop is ready to be assigned from the pool to the next requesting user or device.

In Figure 6, the dynamic pool called Pool-Kiosk has one virtual desktop instance running

that is assigned to a pseudo kiosk session on a Sun Ray DTU (the highlighted entry in the In Use folder in the screenshot). The assignment of the virtual desktop instance to the requesting Sun Ray DTU triggers the creation of a new instance to fulfill subsequent requests. As long as the creation process remains ongoing, the new virtual desktop instance resides in the Factory sub-folder. Once the process is completed, the virtual desktop instance, in this example called Pool-Kiosk-2, is moved up to the parent and is then available for new requests.

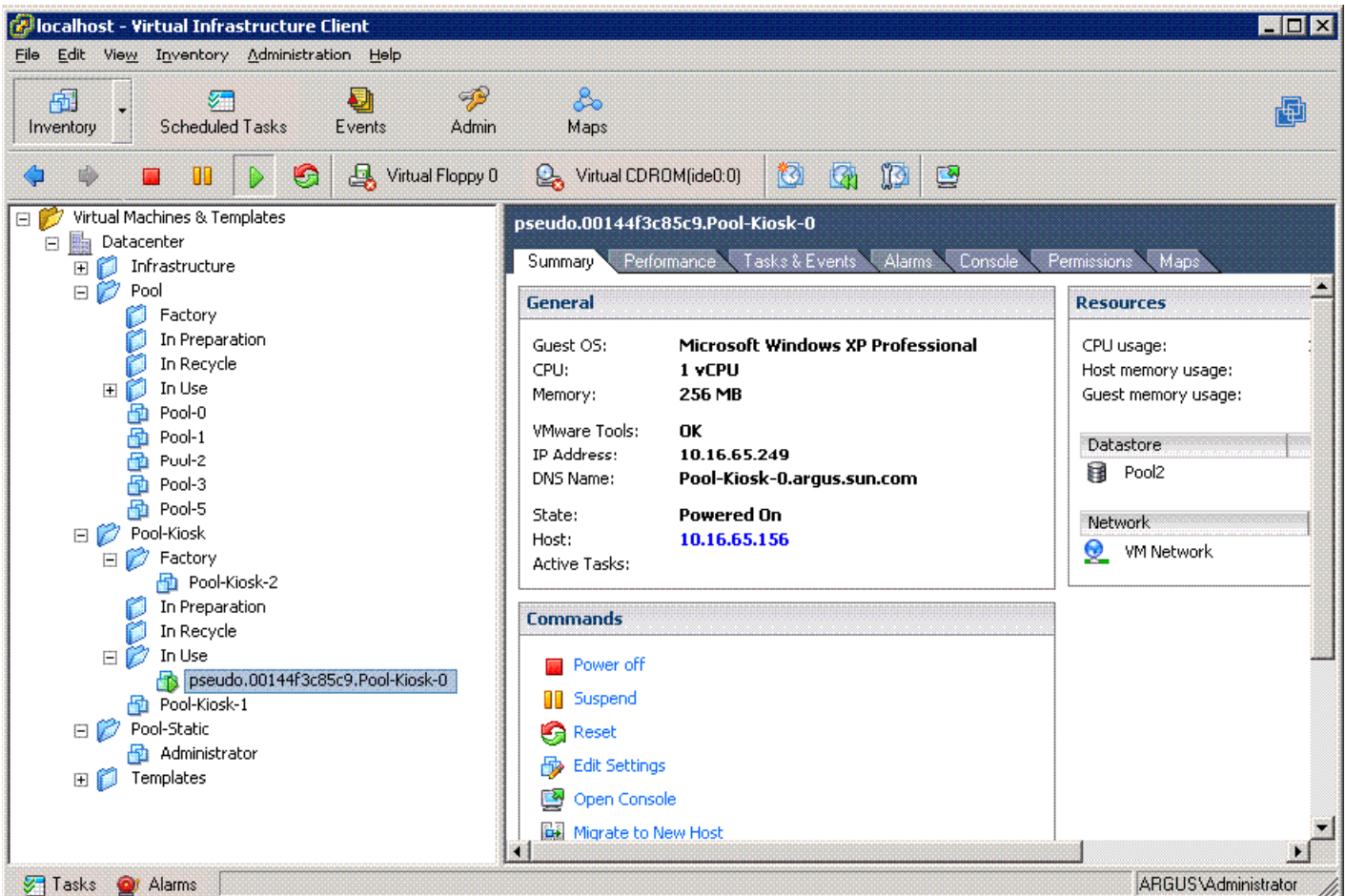


Figure 6. Factory Creating a New Virtual Desktop

The factory applies a specific naming scheme. New elements are named after the name of the pool plus a unique numeric identifier. The applied algorithm tries to avoid holes, which means that if certain virtual desktops are deleted, their names are reused. If Windows system preparation is used, the Windows computer name is identical to the virtual desktop name, which helps to identify the virtual desktop within the Windows Domain and Virtual Center.

## VDA Service Configuration

Communication between the VDA Service and Virtual Center is handled by the VMware Infrastructure SDK Web Service. The URL for accessing the VMware Infrastructure is `protocol://localhost/sdk`. The protocol can either be `http` or `https`, depending on security requirements. Both access methods need to be configured before the VDA Service is started. The configuration steps are described in the VMware Infrastructure SDK Programming Guide available at:

<http://www.vmware.com/pdf/ProgrammingGuide201.pdf>.

Once the organization of a virtual desktop population — how many pools are needed and what their purpose will be — has been decided, the structure can be implemented in Virtual Center with the following steps:

- Create folders for the pools.
- Create and configure the golden images.
- Define and store the required sysprep configurations.

Once this is done, the structure needs to be reflected in the configuration of the VDA Service. All configuration settings for the VDA Service are stored in a single file called `vda.properties`. See “Appendix A: VDA Service Configuration Example (`vda.properties`)” for a sample configuration file. The settings match those in the example in Figure 6.

Once the configuration file has been changed, the configuration itself should be verified with the command:

```
..\sun-vda\bin\vda-service test.
```

After a successful test, the VDA Service must be restarted with the Windows service panel.

## Conclusion

Sun Ray Server Software and Sun Secure Global Desktop provide secure, low-cost remote access to Solaris, Linux, and Windows desktops and applications. The inclusion of a Microsoft certified RDP client (Sun Ray Connector for Windows OS) in Sun Ray Server Software substantially increases the potential for additional Windows Terminal Services deployments.

The Sun Virtual Desktop Access Kit for VMware brings to this remote desktop solution a comprehensive desktop brokering technology that manages the virtual desktop life-cycle, handling both static and dynamic assignment of virtual desktops to users and devices. It also promotes resource utilization within the virtual infrastructure.

Virtual Desktop Access Kit's integration of Sun's desktop access tier with VMware's Virtual Infrastructure 3 components provides an effective, elegant way to implement the type of Desktop from Anywhere increasingly needed by flexible, mobile workforces.

Sun is the only vendor thus far to provide an end-to-end desktop virtualization solution based on VMware VI3 software.

## About the Authors

Dirk Grobler joined Sun through the StarOffice acquisition in 1999. Since then, he has led various projects in Sun's Desktop organization, with a primary focus on the management and administration of desktops and embedded applications. Since 2005, he has been responsible for the Sun Ray Server Software and Sun Secure Global Desktop administration interfaces as well as for the Sun Desktop Manager. Most recently, he initiated the Virtual Desktop Access Kit project

Warren Ponder is a Senior Technical Marketing Engineer for the Enterprise Desktop and Solutions Product Team at VMware. Prior to joining VMware, he was a Global Desktop Architect in Sun's Global Desktop Practice. In this role, he worked with customers to design alternative approaches to traditional desktop architectures. During more than five years with Sun, Warren focused primarily on thin client computing, Windows interoperability, Linux-based desktop solutions, and, most recently, how to use virtualization technology to solve desktop computing challenges.

## Acknowledgments

This paper would not have been possible without the assistance of Cyrille Moureaux. Thanks for his in-depth review and contribution to the design and implementation of the Virtual Desktop Access Kit.

We also would like to thank Rolf-Per Thulin for his help in proving these concepts and testing the Virtual Desktop Access Kit in customer environments.

## Important Links

Sun Desktop Virtualization Blueprint:

[http://www.sun.com/software/sdis/wp\\_desktop\\_virtualization\\_blueprint.pdf](http://www.sun.com/software/sdis/wp_desktop_virtualization_blueprint.pdf)

Sun Ray Software:

<http://www.sun.com/software/sunray/index.jsp>

Sun Ray Virtual Display Clients:

<http://www.sun.com/software/index.jsp?cat=Desktop&tab=3&subcat=Sun%20Ray%20Clients>

Secure Global Desktop Software:

<http://www.sun.com/software/products/sgd/>

VMware Infrastructure:

<http://www.vmware.com/products/vi/>

VDI Server Sizing and Scaling:

[http://www.vmware.com/pdf/vdi\\_sizing\\_vi3.pdf](http://www.vmware.com/pdf/vdi_sizing_vi3.pdf)

Templates Usage and Best Practices in Virtual Center 2:

[http://www.vmware.com/pdf/vc\\_2\\_templates\\_usage\\_best\\_practices\\_wp.pdf](http://www.vmware.com/pdf/vc_2_templates_usage_best_practices_wp.pdf)

Virtual Infrastructure 3 on-line library:

<http://pubs.vmware.com/vi301/wwhelp/wwhimpl/js/html/wwhelp.htm>

Sun x64 Servers:

<http://www.sun.com/x64/index.jsp>

Sun StorageTek Storage:

<http://www.sun.com/storagetek/products.jsp>

Free Power Management GPO:

[http://www.energystar.gov/index.cfm?c=power\\_mgt.pr\\_pm\\_ez\\_gpo](http://www.energystar.gov/index.cfm?c=power_mgt.pr_pm_ez_gpo)

## Glossary

**CAM - Controlled Access Mode**

A mode of the Sun Ray Server Software that allows to run only specified applications. In the context of this solution this is the VDA Client and the Sun Ray Windows Connector that are executed to serve Windows sessions to users and devices.

**dynamic virtual desktop**

A virtual desktop that is temporarily assigned to users or devices.

**golden image**

A virtual machine template that is used to create new virtual desktops through the cloning mechanism offered by VMware VI3 enterprise products.

**static virtual desktop**

A virtual desktop that is permanently assigned to a user.

**VDI - Virtual Desktop Infrastructure**

A virtual infrastructure that serves desktop sessions. VMware brought up this term initially and uses it for desktop solutions around their Virtual Infrastructure products.

**Virtual Desktop Access Kit for VMware**

The set of tool that glues Sun's access tier together with the VMware VI3 enterprise software to deliver an end-to-end Desktop Virtualization Solution.

**virtual desktop**

A virtual machine containing a desktop instance that is executed and managed within the virtual desktop Infrastructure. In the context of the discussed solution this is an instance of Windows XP or Windows 2000.

**virtual desktop factory**

A sub-component of the VDA Service that automatically creates new instances of virtual desktops.

**virtual desktop pool**

A folder within Virtual Center that contains virtual desktops sharing the same characteristics.



## Appendix A: VDA Service Configuration Example (vda.properties)

```

# *****#
# Properties for the Sun VDA Service for VMware
# *****#
# *****#
# Connection: Information to connect to the VDA Service #
# *****#
# Connection.User: User to connect to the service. This should be a dedicated
# user, with limited privileges
Connection.User=Administrator
# Connection.Pwd: Pwd to connect to the service. In a real deployment it is
# recommended to avoid to make these settings world readable.
Connection.Pwd=XXXXXX
# Connection.Pwd: URL of the service
Connection.WebServiceUrl=http://localhost/sdk
# *****#
# Pools: Definition of the pools #
# *****#
# Pool.Count: Defines the number of pools. Defines the number of pools. For each
# pool setting there needs to be a pool number appended.
# <default>: 0
Pool.Count = 2
# Pool.Path: Location of pooled VMs. If not set or invalid, only assigned VMs
# will be considered. As ESX has only one folder for all VM's, the default is set
# to the ESX folder. Please not, this folder does not exist on VC!
# <default>: "/ha-datacenter/vm"
Pool.Path0 = "/Datacenter/vm/Pool"
# Pool.Template: Location of the template VM
# <default>: If not set, the factory is disabled for the pool
Pool.Template0 = "/Datacenter/vm/Templates/Golden Master"
# Pool.MinSize: Minimal number of available machines in the pool
# <default>: 5
Pool.MinSize0 = 3
# Pool.Customization: Name of the customization spec to use.
# <default>:

```

```

Pool.Customization0 = Argus-1
Pool.Path1 = "/Datacenter/vm/Pool-Kiosk"
# Pool.Template: Location of the template VM
# <default>: If not set, the factory is disabled for the pool
Pool.Template1 = "/Datacenter/vm/Templates/Golden Master"
# Pool.MinSize: Minimal number of available machines in the pool
# <default>: 5
Pool.MinSize1 = 2
# Pool.Customizationx: Name of the customization spec to use.
# <default>:
Pool.Customization1 = Argus-1
# *****#
# Folders:
# *****#
# Folders.Static: Location of user assigned VMs. If not set or invalid, all VMs
# will be taken from dynamic pools.
Folders.Static = "/Datacenter/vm/Pool-Static"
# Folders.Cluster: Location of a cluster. If not set or invalid, there must be at
# least a single host available. All advanced features will not work if
# clustering is disabled
Folders.Cluster = "/Datacenter/host/Desktops"
# *****#
# General: (optional)
# *****#
# General.LogLevel: Log level, output into the logs subdir
# <default>:INFO
General.LogLevel = FINEST
# General.Agent: Port for communication with the vda-client
# <default>:3809
General.Agent = 3809
# *****#
# Startup: Settings to test the sanity of a VM after start (optional)      #
# *****#
# Startup.RDPPort: Port that is used for RDP connection. Before a VM is handed
# over to the Access Tier is verified that the RDP port is open for communication.
# <default>: 3389
Startup.RDPPort=3389
# Startup.RDPTIMEOUT: Defines how many iterations should be made to

```

```
# validate if RDP communication is possible. If none of them is successful, the
# requester will fail
# <default>: 30
Startup.RDPTimeout=30
# Startup.Timeout: Defines the duration for how long it is tried to get a valid IP
# for a VM
# <default>: 60s
Startup.Timeout = 60
# *****#
# Monitor: Settings to control the behavior of the VM monitor (optional)      #
# *****#
# Monitor.RecyclePollInterval: Interval in which the monitor observes the pool
# and the factory.
# <default>: 5 minutes
Monitor.RecyclePollInterval = 5
# Monitor.IdleTimeout: Time that a suspended VM can rest in production.
# If the limit is exceeded, the VM will be reverted and moved back into the
# pool.
# WARNING: If you use VC, make sure that time between VC and ESX servers is in
# sync. Otherwise, the value gets arbitrary.
# <default>: 60 minutes
Monitor.IdleTimeout = 5
# Monitor.FactoryPollInterval: Length between iterations that check if enough
# virtual machines are available in defined pools. If there are not enough VM
# available, the creation of new VMs is initiated.
# <default>: Monitor.RecyclePollInterval
```

