

Connection

Novell Connection Magazine // AUG_2010

ARTICLES

PUBLISHED FOR
NOVELL CONNECTION
MAGAZINE.

NOTE: All content © 2010 by Novell, Inc. All rights reserved. For more information about Novell Connection Magazine or to obtain approval for bulk reprints, contact editor@novell.com

Features

// **Avoid Disaster in Disaster Recovery**

Crash Landing: Choosing between affordable and effective with your disaster recovery solution? PlateSpin Forge and PlateSpin Protect span the gap for Windows and Linux.

// **Population Automation**

Pop Goes the World: Populating a CMDB and maintaining its accuracy is one of the biggest challenges in Business Service Management. Now there's an efficient, automated solution.

// **Always in Sync**

Automatic for the People: Take an in-depth look at how the architectural elements of Novell Data Synchronizer combine to deliver real-time synchronization that your users need.

Departments

// **Synovus Financial**

// **The Chase is On**

// **Making Programs—and Companies—Talk to Each Other**

Avoid Disaster in Disaster Recovery

PlateSpin Forge and PlateSpin Protect Span the DR Gap for Both Windows and Linux, Including SUSE Linux Enterprise from Novell

by Jason Dea

When you hear “disaster recovery,” if you picture expensive like-for-like infrastructures replicating your most essential servers, perhaps it’s time to rethink disaster recovery (DR). If you envision cumbersome backup servers running all night followed by tape cartridges traveling to a storage vault, again, it’s time to reconsider your DR strategy.

If you are a Linux administrator and see yourself orchestrating your DR solution with scores of commands, it’s time you find a more efficient way to manage DR in today’s environment of shrinking resources and growing IT needs.

In each of these scenarios a gap exists between the cost and the effectiveness of disaster recovery. You get either cumbersome and affordable or expensive and effective--the choices administrators have been facing for years.

Novell designed PlateSpin Forge and PlateSpin Protect from the ground up to span this gap, and Windows administrators have been using them for some time. Novell has recently released the latest updates to these products to now protect environments running Linux and mixed Linux-and-Windows environments.

If you see yourself orchestrating your DR solution with scores of commands, it’s time you find a more efficient way to manage DR.

> Spanning the Gap

A high-availability, or DR-by-duplication, strategy focuses on protecting workloads using local clusters, duplicate host sites or other like-for-like physical infrastructures. These strategies provide very low often near zero recovery time objective (RTO) and recovery point objective (RPO). These are the time it takes to move from a failure back to full production and the amount of data you lose in the process of recovery, respectively. However, even though RTO and RPO can be very low, these solutions can be very costly and extremely complex to configure and manage.

Traditional backup, on the other hand, eliminates the need for duplicate systems by storing backup data on tape or imaging devices, but acquiring new hardware, configuring it to the state of the original and restoring the data from the backup archive can take a long time, reducing effectiveness and increasing administrator aggravation.

The PlateSpin Forge and PlateSpin Protect products from Novell span the gap between cost and effectiveness with an entirely different way to think about DR.

The concept is quite simple, and the solutions are really equally easy to administer. PlateSpin Forge and PlateSpin Protect create a virtual machine (VM) copy of protected physical or virtual workloads and update the virtual machines at intervals you specify from 60 minutes to weekly

or longer. When a server fails, you receive a notification and switch the users to this backup VM, almost immediately. The VM resumes the application load using its most recent update. And once the failed server is operational, you can either restore the VM from scratch to the server or do a reverse replication of just the changes since the failure occurred. The reverse replication would be appropriate when the original server can be repaired. If the original server configuration cannot be obtained, PlateSpin Forge and PlateSpin Protect have the flexibility to restore back to any x86 server from any vendor.

The PlateSpin Forge and PlateSpin Protect products from Novell span the gap between cost and effectiveness with an entirely different way to think about DR.

> PlateSpin Forge versus PlateSpin Protect

Before we go further, you should understand that PlateSpin Forge and PlateSpin Protect are like two versions of the same product. They provide the same capabilities, use the same administration console and are managed the same way.

PlateSpin Forge is an all-in-one appliance that includes the storage, replication software, remote management interface and hypervisor. You just plug the appliance into your network, configure it, and it begins to protect up to 25 workloads. Novell designed PlateSpin Forge for medium-size enterprises and branch and field offices of larger enterprises.

PlateSpin Protect offers more flexibility in larger, more complex networks. It includes just the software the replication software and management interface. You install PlateSpin Protect on your own virtual infrastructure connected to your own storage system. You can purchase as many workload licenses as you want, giving you maximum flexibility.

With both products, you manage the entire DR system, regardless of the number of licenses and DR appliances, through a single graphical interface. You can also integrate these products into your existing DR infrastructure with an SDK that lets you configure a custom solution.

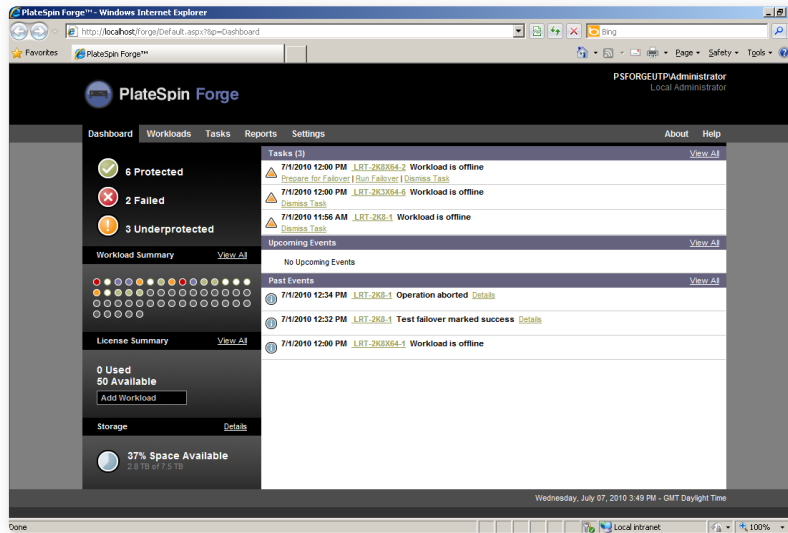
> How You Manage Disaster Recovery

Now that you have a basic idea of what PlateSpin Forge and PlateSpin Protect are and how they might fit into your DR plans, you're probably most interested in what you have to do as an administrator. Let's take a look at how you monitor workloads, configure workloads, execute a failover, perform a fallback, and conduct testing. We'll also look briefly at some of the reporting capabilities.

> Monitoring Workloads

The PlateSpin Forge and PlateSpin Protect Web-based console begins with the Dashboard. (See [Figure 1.](#)) Other than the name in the top left corner, the management consoles are nearly identical for both products.

The Dashboard gives you an overall view of the workloads that are being protected: how many workloads are protected, whether any have failed and how many workloads are not protected.



Main Dashboard

Figure 1: Management of the PlateSpin Forge and PlateSpin Protect DR protection begins in the Web-based console with the Dashboard.

Simple icons in the Workloads Summary show you the status of each workload. A green light means the workload is running normally. An orange light means there is an error in the configuration; perhaps you forgot to configure some part of the protection. A red light means the workload has failed.

The License Summary shows how many licenses are being used.

The Storage pie chart shows the status of the available storage. PlateSpin Forge includes storage in the appliance, but you can also attach external, SAN or NAS storage.

On the right side of the screen, you can see a log of the past, current and upcoming events.

Simple icons in the Workloads Summary show you the status of each workload.

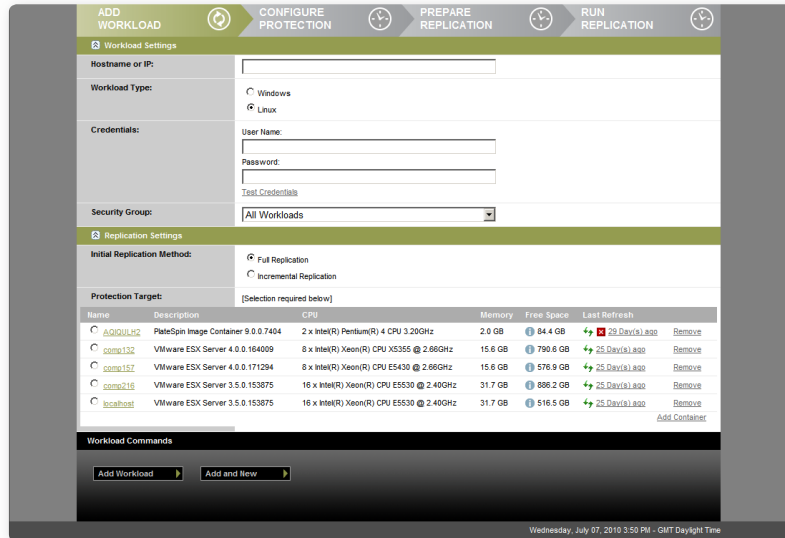
> Configuring Workloads

To configure a workload, you just need to tell PlateSpin Forge or PlateSpin Protect where the target workload is on the network by providing a host name or IP address. (See Figure 2.) You also provide the administrator credentials so the product can interact with the workload as needed. Then you identify the schedule for incremental replication based on how often the data changes and how much network bandwidth you want to use for replication.

One nice feature Novell added to these products is Smart Replication, which slices large batch replications into small portions that can be processed quickly without overloading the network.

A number of other configuration settings include whether you replicate by file or data block, how often you want to replicate, how often you want the devices to ping the workloads to ensure they are still functioning, whether to encrypt transfer data, the volumes that are protected and so on.

These are all menu driven and easy to follow. You can also set these configurations individually for each workload, or you can configure sets of workloads using templates.



Configuring protection for a workload

Figure 2: Inputting the workload hostname or IP address and its credentials gives PlateSpin Forge or PlateSpin Protect the information it needs to find and protect the workload.

> Managing DR by Workload

Novell designed these DR products to take the guesswork out of managing them. For example, in the Workloads tab, where you will spend the bulk of your day-to-day management, you can see that the console activates only the commands that are applicable to each workload. (See [Figure 3](#).) If you check on a workload that has failed, the Prepare for Failover and Run Failover commands become active.

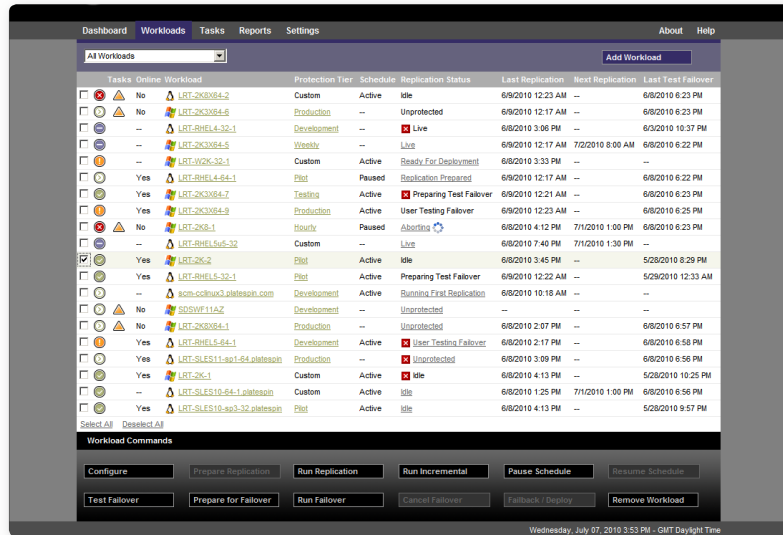
This window also provides the other information you need about each workload, such as whether it's online, its protection tier, its replication status and so on.

> When a Workload Fails

If a workload or a server fails, you have two options: conduct an immediate failover, wherein the VM will boot up and begin serving the applicable users from the last snapshot, or start with the Prepare for Failover command. If there is a chance the failover is a false alarm, say someone simply tripped over a network cable, you should start with Prepare for Failover. With this command, the DR appliance stages the failover by configuring the virtual machine to take over for the failed workload but leaves it in a paused state. You can then choose whether to complete the failover by selecting the Failover or Cancel Failover commands.

> Restoring the Production Server

This is where PlateSpin Forge and PlateSpin Protect really shine. Traditionally, once you've repaired or replaced a failed server, reinstalling the OS, applications and data can be a nightmare. However, with these products, all these steps are a one-click process. They can build the new workload from scratch, or in the case of a repaired machine, you can choose to restore just the changes that have occurred since the workload failed. This includes restoring plug-and-play drivers that have been updated or that are different because you're using new equipment. PlateSpin Forge or PlateSpin Protect automatically install any necessary up-to-date drivers for Windows and Linux.



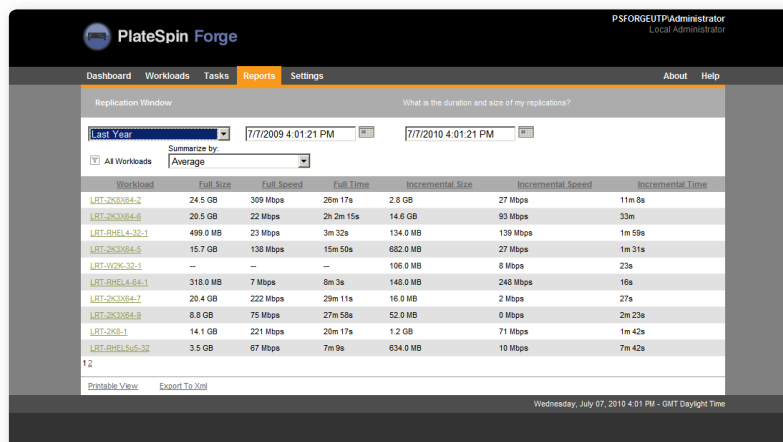
Workloads tab

Figure 3: Workload commands are available dynamically based on the workload you highlight in the workload protection list.

Novell designed these DR products to take the guesswork out of managing them.

> Testing Your DR Environment

If your DR system failed, now that would be a disaster. Testing is an important part of every DR strategy, and in the past this often required building an expensive and cumbersome parallel network just for testing. These DR products from Novell provide a Test Failover command that takes a snapshot of the VM and boots the snapshot to a virtual environment that you have defined as the test network. You don't boot the backup VM itself, and you don't affect the production network, so you can run this test anytime that is convenient for you.



Reports tab

Figure 4: PlateSpin Forge and PlateSpin Protect maintain a history of activities and events and provide reports online, as PDFs or as exportable data for spreadsheets and other applications.

> Tracking DR Performance

PlateSpin Forge and PlateSpin Protect keep a history of all DR activities and events and measures performance such as how long it takes to run a job. They provide this data in several reports that you can view online, save as PDFs or export to applications such as spreadsheets. (See [Figure 4.](#))

> An Easier, Affordable DR Solution

When one of your workloads fails, you no longer need to go through a cumbersome several-step process from retrieving a flat-file archive, sourcing a new server or repairing the existing server, redeploying the operating system and other software, and rebuilding the data files just to get your services back up and running. And you don't need to employ an expensive duplicate system. With PlateSpin Forge or PlateSpin Protect, you have a backup of the workload ready to move into place almost immediately, and it restores back to your production servers just as smoothly, with very little disruption to the users on your network.

Additional Resources // Novell Connection Magazine

- www.novell.com/products/protect
- www.novell.com/products/forge
- At the bottom of the Workloads tab in the PlateSpin Forge and PlateSpin Protect console are the Workload Commands. These buttons become active depending on the status of the workload you check. [Click here to see the full list online.](#)

Population Automation

By Bill Tobey

> Successful BSM Depends on an Accurate CMDB

If you're implementing Business Service Management (BSM) for any of the usual reasons—to increase the reliability and availability of your critical business services, to improve your service delivery performance, or to reduce the costs of service delivery—be aware that your success in achieving any of these objectives will be highly dependent on the quality and timeliness of the data in your Configuration Management Database (CMDB).

Today's CMDB is operational by definition. When an outage occurs, you can't do fast root-cause and impact analysis unless your CMDB can access and supply near real-time information about all services and configuration items (CIs) under management, the relationships between them, and their current service states. The CMDB is a special-case database that must provide four critical functions creating the Configuration Management System (CMS):

- Federation – Direct linkage to multiple data sources
- Reconciliation – Automated data coalescence and matching across sources to eliminate duplication
- Visualization – Automatic detection and graphic representation of the hierarchical relationships between physical and logical CIs
- Synchronization – The ability to immediately capture changes in the physical and logical infrastructure, preserving and providing a single version of the truth across all integrated systems

One of the key challenges in creating and maintaining a CMDB is the problem of populating it from the various system, asset and network management solutions where information about the IT environment originates. Manual processes may offer a high level of initial accuracy, but in large environments, their labor intensity virtually guarantees decreasing accuracy over time.

> SCM: Population Propagation, the Automated Way

But if you're implementing a Novell CMDB, you've got an integrated population automation solution sitting right under your right mouse button, ready to launch. The Service Configuration Manager functions as an intermediary between the CMDB and the adapters that integrate it with management applications and configuration information repositories, providing the Configuration Management System (CMS) automation. The Service Configuration Manager feature dynamically generates new element hierarchies from multiple sources. It automates CMDB synchronization to optimize data quality and reliably capture change in the IT infrastructure in near real time.

The Configuration Management System provides:

- Integration into tools that detects dependencies, characterizes normal baselines, and identifies both scheduled and unscheduled changes
- Integrated mapping of element relationships across an enterprise IT environment
- Integration of asset, configuration and change data from multiple sources into a global CMDB
- Automatic creation and dynamic maintenance of Business Service Views that eliminate manual modeling, mapping and maintenance.

CMS not only automates many of the most time- and labor-intensive aspects of configuration data management, it lets you create and store customized business rules, in the form of JavaScripts, to govern key processes. Let's look at an example.

> **From Many Sources, One CMDB**

The fundamental use case in CMDB implementation is populating the configuration data store from a large number of diverse management systems and data repositories across the environment, and potentially from other siloed CMDBs. Let's assume that you're starting such a project using the Novell Configuration Management System. You have the Novell engine up and running on a server, along with adapters for each of the management systems you'll need to integrate with. You also have the Service Configuration Manager administrative client running on a desktop or notebook. You want to begin federating basic information about configuration items—services and all the infrastructure elements that participate in their delivery—in the CMDB.

- Begin in the Java console by drilling down into the CMS service model. Select the CI container inside of a community—servers for instance—and right click. From the list of options that is presented, choose Service Configuration Manager (SCM). Then choose to create an SCM job.
- Service Configuration Manager will now display a list of the available source system adapters. Since we're seeking information on servers, select all the adapters for target systems that are likely to contain identity, role, configuration or service state information on server systems.
- Now it's time to decide what information we'll select from each of the available sources, and how we'll load that information into the CMDB. Service Configuration Manager provides a checkbox to populate the CI Attributes from the data sources, but in most cases, unless you have an exact naming match between the source attribute names and the destination CI attribute names, you will need to set up a script to do the actual processing. Regardless, the java script approach provides additional control over how CI Attributes are populated such as pre-processing to do validity checks.

To manage the selection, linking and reconciliation of CI attributes, SCM gives us the option of using a custom JavaScript, like the following:

```
// This script is called for different types of matches. It checks to see what elements exist
// in our source systems, and that each element has been accurately created in the CMDB.
// It begins with three conditions for execution.
if( this.source && element )
{
  // Define attributes (properties) that are on the source system
  if( !state.sourceAttrs )
    state.sourceAttrs = [ 'filename', 'name', 'ip_address' ]

  // Define the attributes (in matching order) that you want the source to populate
  if( ! state.destinationAttrs )
    state.destinationAttrs = [ 'Filename', 'DB Name', 'TCP/IP Address' ]

  //Using state variables above reduces memory utilization on the server.
  //Since this script is run over and over again for each matched element,
  //it reduces run time by not redefining these variables every time.
  //Use a for loop to traverse through the identified source attributes
  for( var i = 0; i < state.sourceAttrs.length; ++i )
  {
    // attempt to pull source attribute
    var value = source[ state.sourceAttrs[i] ]

    // if it exists
    if( value )
      element[ state.destinationAttrs[i] ] = value
    // then on the line above, we then set the element
    // attribute to the value of the source attribute
    // in turn, this is then persisted to the configStore (database)
  }
}
```

Additional information on scripting for SCM control can be found in the FormulaScript Guide that is included in the CMDB installation documentation.

> A Script for Accurate, Efficient, Cost-Effective Service Management

The built-in automation capabilities of the Novell Configuration Management System combined with Service Configuration Manager's support for customized JavaScript control take many of the complexity, labor intensity and data quality challenges out of creating and maintaining a configuration management database. They're just one instance of the extensive automation capabilities that help Novell BSM solutions deliver faster time-to-value, lower implementation and maintenance costs, greater accuracy, improved business value and operational performance from all services under management. To learn more visit www.novell.com/bsm.

Always in Sync

by Ken Baker

> A Deep Dive into Novell Data Synchronizer

As you read in the June issue of Novell Connection Magazine (See [Higher Levels of Collaboration](#)), Novell Data Synchronizer keeps applications and mobile devices constantly in sync so users can always have access to the data they need. That article talked about the product's mobility pack, its many-to-many synchronize engine and how to configure connectors. This article takes a deeper technical dive into the way connectors facilitate synchronization, as well as a few other supporting architectural elements.

> Synchronization Channels and Filters

The majority of the intelligence in Novell Data Synchronizer resides in its connectors. Written to application-specific or standard APIs, the connectors enable Data Synchronizer to understand how and what needs to be done with data to be synchronized. For example, the SharePoint connector uses the SharePoint Web Services API to access data and push it through the system. The Salesforce.com connector uses the Salesforce.com Web services API. Similarly, the GroupWise connector utilizes the SOAP interface to synchronize e-mail, tasks, calendar data and contact information. The use of standard or native APIs allows each connector to be built specifically for its target application.

Each connector in Novell Data Synchronizer is comprised of two main components: channels and filters. Channels conduct the flow of data from the synchronization engine through the connectors to the connected systems. Filters act upon the data—translating and manipulating the data in a prescribed manner.

At a very high level, channels and filters work together in the following fashion:

1. As data moves through the Novell Data Synchronizer channels between two or more applications, the Novell Data Synchronizer connectors act as code converters and data filters.
2. The data flows through the Novell Data Synchronizer system in the form of XML files that are either in an application-specific format or an application-neutral format.
3. As data moves from the application toward the synchronization engine, it will initially be in an application-specific XML file format.
4. As the application-specific XML file travels through the channel, the connector filter will translate the XML file into an application-neutral format.
5. After it's translated into an application-neutral format, the XML file is stored in the synchronization engine database.
6. The application-neutral XML files stored in the synchronization engine database can be consumed by different application-specific connectors (i.e., GroupWise, SugarCRM, Sharepoint, Salesforce.com), which in turn can translate them into the appropriate application-specific format as they transfer them to their associated application.

If you're using Novell identity and security products, you'll likely recognize that this design is very similar to that of Novell Identity Manager. While Novell Identity Manager synchronizes identities from different sources and targets, Novell Data Synchronizer synchronizes data in a similar fashion. These similarities are perhaps most notable in the use of a two-channel system with a source channel for outgoing events and a SINK channel for incoming events. In Novell Data Synchronizer, these channels exist in both the synchronization engine and the individual connectors.

The source channel in the synchronization engine is responsible for taking item or event data from its cache and passing it to the connector. After the connector translates the data into the appropriate application-specific format, the connector's SINK channel will pass that information to the application. For data coming from an application to the engine, the connector's source channel has responsibility for taking data from the application, while the engine's SINK channel will place the application-neutral formatted data into its cache.

Written to application-specific or standard APIs, the connectors enable Data Synchronizer to understand how and what needs to be done with data to be synchronized.

> How to Synchronize

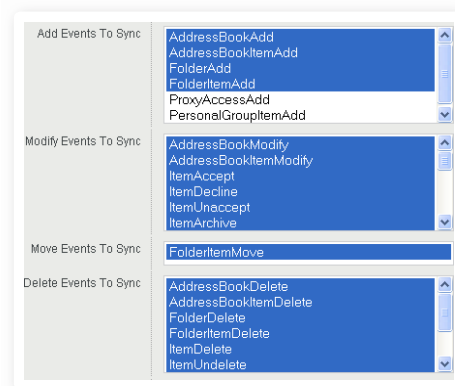
In addition to having translation filters, connectors have synchronization filters. Depending on application needs and purpose, each application will have different types of data that it will allow to be synchronized. Some of the common data types (referred to as data items) include e-mail messages, appointments, tasks, notes, address books, contacts and folders. Connectors will not necessarily support all the same types of data items. In other words, data item synchronization will vary based on the functionality and needs of the application.

In addition to determining what data types are synchronized, filters have responsibility for managing the synchronization of changes that occur to data items within a connected application, system or device. These changes (known as events) might be represented by actions such as an item addition, modification, move or deletion.

As data items and events pass through a connector, various filters applied to the data determine what and how the data is synchronized. Depending on the connector design, you can often use filters to determine which data items and events you want to synchronize.

Not only can you use filters to screen for certain content or fields, but you can use them to transform one data type into another data type, such as turning task data into e-mail data. Filter behavior can also be modified based on certain triggers. For example, you can have a trigger that causes a filter to treat a data item differently based on where that data was originally stored on the source application.

You configure a connector's filters through the Novell Data Synchronizer Web Admin console. For example, for the GroupWise connector you can choose from a list of add, modify, move and delete events that it will synchronize. (See [Figure 1.](#))



Events to synchronize
Figure 1: You configure Data Synchronizer filters to determine what data you want synchronized

> More on Filters

One of the standard filters in Novell Data Synchronizer provides a circular sync check. As data passes from the connector to the synchronization engine, this filter makes sure that it's not synchronizing the same data over and over again in a circular fashion. This circular check filter prevents the creation of duplicate information.

The default filters that come with a connector are typically sufficient to enable the translation of data from an application-specific format into an application-neutral format and vice versa, as well as apply other basic filter operations. However, since these filters are made up of XSLT files, if you're familiar with XSLT you can manually configure them if necessary. (See [Figure 2](#).)

While you can define additional filter functions using the connector's custom namespace, the following are standard functions provided in the Novell Data Synchronizer namespace:

```
def logMessage(self, context, message, level='warning'): log a message

def getSettingsForTarget(self, context, targetDn): returns a user's settings

def getConnectorSettings(self, context): returns a connector's settings

def getEngineSettings(self, context): returns the sync engine's settings

def getConfigEngineSettings(self, context): returns config engine's settings

def getMatchingUserDN(self, context, sourceName): returns the dn of a user with a
unique cn provided

def dnExists(self, context, dn): checks if the dn is a valid dn in the tree

def isTargetEnabled(self, context, dn): checks if the user is in the user table
for the connector

def base64Encode(self, context, message):

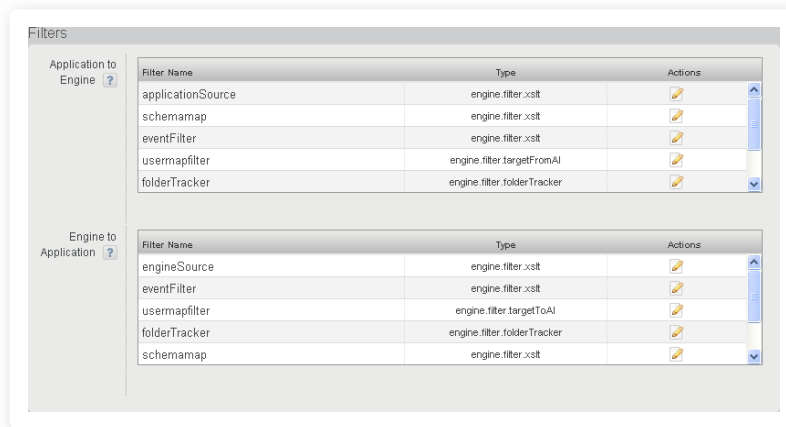
def base64Decode(self, context, encodedMessage, returnAsNodeset=False):

def stripHTML(self, context, text, unescapeFirst=False): removes html from the
text

def itemIDToObjectID(self, itemID):

def objectIDToItemID(self, objectID):
```

In addition to determining what data types are synchronized, filters have responsibility for managing the synchronization of changes that occur to data items within a connected application, system or device.



Filter management

Figure 2: If desired, you can manually configure the filters' XSLT files.

> Other Novell Data Synchronizer Components

In addition to the synchronization engine, connectors, channels and filters, the following three components play a critical role in the operation of the Novell Data Synchronizer system:

- **Web Administration Service** provides a Web-based interface called Synchronizer Web Admin to facilitate administration and management of the Data Synchronizer system. It allows you to add and remove connectors, as well as add or remove users for specific connectors and configure their individual synchronization settings. Users can also use the Synchronizer Web Admin to configure and control synchronization settings for each of their connected applications and devices.
- **Synchronizer Configuration Engine** provides communication between the Synchronizer Web Admin and the synchronization engine. The configuration engine passes configuration information from the synchronization engine to the Synchronizer Web Admin for viewing. It also passes configuration changes back to the synchronization engine for implementation.
- **Connector Manager** provides communication between the synchronization engine and connectors.

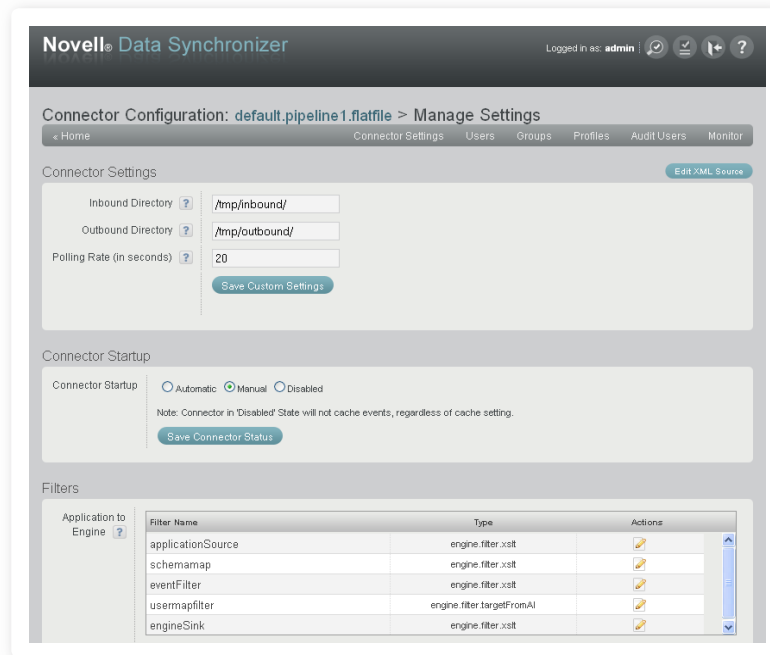
Novell Data Synchronizer also includes LDAP integration. While not required, Novell Data Synchronizer can leverage an LDAP directory to further facilitate user and group management.

> Developing Connectors

In addition to the connectors that Novell provides, Novell gives you a number of tools to facilitate the development of additional connectors. The first of these tools is the software development kit for Novell Data Synchronizer, which guides you in how to take advantage of the product's open API.

Novell also provides a generic SOAP connector, which accesses Novell Data Synchronizer using standard SOAP calls. Primarily intended to be used as an example connector, the SOAP connector provides a guide or reference point for someone who wants to develop a connector. You can use the connector as a model to see how the synchronization engine transforms data coming out of the GroupWise system.

As an additional connector development tool, Novell also provides a flat file connector. The flat file connector can take item and event data cached in the synchronization engine, translate it



Flatfile connector configuration

Figure 3: The flat file connector can take item and event data cached in the synchronization engine, and translate it into an XML format, and then push it to a specified directory.

into an XML format administrators can read, and push it to a specified directory or folder on the network. Once placed in the specified folder, it can be consumed by other applications or manually accessed by users. (See Figure 3.)

The flat file connector also has the ability to periodically poll a specified folder for newly stored data. When it identifies new data, it can grab that data and pass it back to the synchronization engine for consumption by connected applications or devices.

While it operates on a fairly simple and straightforward concept, its inherent flexibility and open nature make the flat file connector quite powerful in helping you extend the capabilities of Novell Data Synchronizer in a variety of ways. In essence, it gives you a semi-technical programming interface that can help you customize and extend the capabilities of Novell Data Synchronizer.

Leveraging the connector, you could create a custom application that when a certain event happens, it generates information in an XML file format and then stores it in a specified folder. Once in the folder, the flat file connector can grab it, transform it and store it in the synchronization engine where it can be used by other connected applications.

The synchronization engine, connectors, filters, channels, open development environment and other architectural components of Novell Data Synchronizer all combine to give you the real-time synchronization that your users need among their diverse applications and mobile devices.

Another possible use would be to use the flat file connector to retrieve and forward reports generated by a certain application. The application could generate the report as an XML file and store it in a specified folder. As the flat file connector polls that folder, it would discover the new report, transform it into an application-neutral format, and store it in the synchronization engine database where it can be consumed by other connected applications or sent to a specific user.

The flat file connector could also be used to pre-populate a new user's GroupWise, SharePoint, or SugarCRM system with standard information. To accomplish this, you might manually create a file in the proper XML format with the desired information and then store it in a specified directory. The flat file connector can then grab, transform and store that data in the synchronization engine cache. When a new account is created for a user, its application-specific connector can then automatically pull that data from the synchronization engine and pre-populate the user's account accordingly.

> Syncing It All Together

The synchronization engine, connectors, filters, channels, open development environment and other architectural components of Novell Data Synchronizer all combine to give you the real-time synchronization that your users need among their diverse applications and mobile devices. They work together to help you improve your organization's collaboration efforts through seamless and real-time synchronization of e-mail, calendar, contact, event and other collaboration data residing in your business and collaboration systems. Whether in the office, at home or on the road, Novell Data Synchronizer can help ensure that your users always have access to the right information in the right place at the right time. To learn more visit www.novell.com/datasync.

Synovus Financial

A multi-chartered financial services company with 30 different banks, Synovus Financial was looking to automate user provisioning and streamline access management. By using Novell Identity Manager and Novell Secure Login, Synovus has tightened security while vastly reducing IT costs.

> Overview

Synovus Financial (Synovus) is a more than \$33-billion dollar financial services holding company. Synovus provides commercial and retail banking, as well as investment services, to customers through 30 banks, 330 offices and 467 ATMs in Georgia, Alabama, South Carolina, Florida and Tennessee.

> Challenge

As a highly regulated institution, Synovus sought to tighten access and security controls to satisfy bank regulators and third-party auditors, while driving down escalating support costs. "User provisioning and deprovisioning used to take several days," said Steven Jones, director of Operational Risk for Synovus Financial. "We needed to be able to modify or revoke user access rights immediately in order to mitigate risk."

The company also was under pressure to enforce strong password management. "Each user had at least six unique passwords, and inevitably many of them forgot passwords or wrote them down," said Jones. "Forgotten passwords and locked accounts actually represented 70-80 percent of our helpdesk calls. We needed a solution that would alleviate this burden while bringing tighter security across the enterprise."

In addition, Synovus wanted to find a way to improve user convenience while strengthening security. "We wanted to bring ease of operability to our employees to create a more efficient front-line experience," said Jones. "We have a dynamic workforce. When a teller moves from one branch to another, for example, we needed a way to tie their credentials to their role so they could quickly get up and running."

"Novell identity and access management solutions enable us to demonstrate appropriate access control and risk management practices for Sarbanes-Oxley and other regulations like Gramm-Leach-Bliley, while cutting the associated IT costs by 80 percent."

Steven Jones, Director of Operational Risk, Synovus Financial

> Solution

Synovus is now leveraging Novell SecureLogin to provide enterprise single sign-on capabilities. “Novell scripting capabilities gave us the ability to handle scenarios that other solutions couldn’t provide,” said Jones.

The company is also using Novell Identity Manager to automate user provisioning and deprovisioning. “Novell Identity Manager was the only solution that suited our needs,” said Jones. “We liked that we could leverage Novell eDirectory services for the provisioning of third-party banking applications.”

Synovus now uses Novell Identity Manager to synchronize and manage more than 6,700 identities across its HR system, FIS deposit and loan origination system, homegrown banking applications, multiple directories, Novell GroupWise, as well as its corporate intranet and training software. Additionally, Synovus used SOAP services to integrate some of its legacy banking systems.

Synovus has benefited from the solution’s attribute-level data management capabilities. “We can now use a wide variety of attributes to determine user roles,” said Jones. “By leveraging Novell workflow modules, we’ve really been able to automate the entire user provisioning and deprovisioning processes, which now occur in near real time. In addition, if a user changes roles or moves from one location to another, all of their access rights are updated automatically. Novell Identity Manager helps us ensure the integrity of our workflows and gain more efficient processes.”

Novell SecureLogin provides Synovus with enterprise single sign-on capabilities to ensure its internal and external password security policies can be enforced with far less effort. Users now only need to remember one password, instead of six or more. “We’ve been very impressed with Novell SecureLogin,” said Jones. “It met all of our requirements, offering tight integration with directory services, the ability to leverage directory service attributes and support for credential provisioning.”

Synovus worked with Novell Consulting throughout the implementation. “We have an exceptional working relationship with the Novell consulting team and have always found it to be a positive experience,” said Jones.

“Novell Secure Login has made it easier for users to move from application to application. They can focus on the customer they’re working with, rather than having to fumble through applications.”

Steven Jones, Director of Operational Risk, Synovus Financial

> Results

Working with Novell has helped Synovus tighten security and access controls while lowering its total cost of ownership. “Novell Identity Manager gives us a framework for effectively managing user identities. We can now maintain security controls while realizing the productivity gains that come from an integrated solution.”

“Novell identity and access management solutions enable us to demonstrate appropriate access control and risk management practices for Sarbanes-Oxley and other regulations like

Gramm-Leach-Bliley, while cutting the associated IT costs by 80 percent," said Jones. "We've further reduced risk by eliminating excess accounts that could otherwise have been misused."

Using SecureLogin, Synovus has further increased its security controls, while significantly reducing password-related helpdesk calls and enabling the rapid deployment of new Web-based applications. "From a risk management perspective, Novell SecureLogin offers tremendous advantages," said Jones. "We've eliminated risky password behaviors while ensuring the right people have access to the right information."

Since the rollout of a single sign-on solution, the company has seen tremendous financial and productivity gains. "With Novell SecureLogin, we've reduced our password-related helpdesk calls by 40 percent," said Jones. "The solution has also made it easier for users to move from application to application. They can now focus on the customer they're working with, rather than having to fumble through applications."

The Chase Is On

by Todd Swensen

> Introducing a Bold New Voice in Cloud Computing

There's no way around it. Because it's sponsored and produced by Novell, CloudChasers technically qualifies as a "corporate podcast," which is both an oxymoron and a red flag in the eyes of many podcasting enthusiasts. But if you actually take a moment to listen to CloudChasers, it doesn't take long to realize that this is not another vehicle for marketing fluff and corporate spin. Maybe it's the fact that the show's host, Matthew T. Grant, loves to surprise his unsuspecting guests with off-kilter questions like, "Is Google about to drink Microsoft's milkshake?" Or maybe it's a willingness to seek out guests who tend to say things like, "Every time you think [in terms of] compliance or security, God kills a kitten" into a live microphone. Regardless, it's clear that CloudChasers works overtime to renounce, reject and move beyond the familiar dry, safe, warmed-over webinar approach to corporate podcasts. So if it doesn't fit the typical mold, exactly what is CloudChasers all about? And more important, what does it add to the cloud computing discussion?

“We wear seatbelts because we are afraid of \$30 fines, not because we are afraid of dying. That’s also the case with regulations.”

Dr. Anton Chuvakin, on the May 20, 2010 episode of CloudChasers

> Chasing Clouds for Fun and Profit (or just fun)

The answers to these questions start with the brains behind CloudChasers, Novell director of new and social media Frank Days and the show's host, Matthew T. Grant. "From the moment we started talking about doing a Novell sponsored cloud computing podcast, we knew we would be facing some fairly unflattering corporate podcast stereotypes," said Days. "To counteract those pre-conceptions, we knew we had to go out of our way to find guests who represented a lot of different cloud computing perspectives, turn them loose to speak their minds and find ways to have some fun in the process. It took us a few weeks to nail down a format that worked, but now I think we've really hit our stride."

The show's guests have certainly helped. So far, CloudChasers has featured prominent, no-nonsense cloud bloggers and podcasters like [Geva Perry](#) and [David Linthicum](#), cloud experts from the likes of IBM and VMware and a range of experienced cloud computing practitioners. "Since we launched the podcast in April, we've made it a point to feature guests with a lot of different experience and perspectives," said Grant. "Then, it's my job to get past their filters and get them to say things that are unexpected and maybe even a bit controversial. That's the great thing about the podcasting format. It's a totally spontaneous, unscripted conversation where my guests have no idea what I'm going to ask them about. That certainly makes things a lot more fun and interesting."

Anyone familiar with cloud computing knows there is certainly no shortage of hot (and even contentious) topics to work with. "In a lot of ways, cloud computing is still in its infancy," said

Grant. “That means there’s an almost overwhelming number of interesting things to discuss. We’ve been doing a show every week since late April, and I feel like we haven’t even scratched the surface.”

Over Cloud Chaser’s first three months, the show has mainly featured cloud computing pundits, bloggers and experts. “To start out, we wanted to get most of the big cloud issues and challenges out on the table, so we brought on a lot of fairly prominent experts,” said Days. “Now that we’ve set the stage, we plan to start featuring more practitioners—people who are in the trenches doing interesting and innovative things with private and public clouds in their own companies. Of course, we’re always willing to sit down with anyone who has interesting, relevant things to say about the cloud—as long as they’re willing to speak their mind and tackle the tough questions.”

“Cloud computing is going to force new relationships in the IT organization. So get ready for it.”

Dustin Amrhein, technology evangelist for IBM, on the April 15, 2010 episode of CloudChasers

> Top 3 CloudChaser Episodes

Of course, the best way to get a feel for CloudChasers is to simply start listening. With 15 weekly episodes already in the can, there are plenty of great episodes and highlights to choose from. If you’re not sure where to start, here are three especially interesting and entertaining episodes to get you started:

1. **Cloud Security and Compliance: It’s all about the Logs**—Some might consider it ironic that one of the most entertaining and laugh-out-loud funny episodes of CloudChasers revolves around the topic of log management. But when you combine an opinionated, outspoken guest like Dr. Anton Chuvakin, inappropriate references to the untimely demise of kittens and a tortured Star Trek metaphor, you end up with an especially quotable cloud computing discussion.
2. **Battle of the Public Clouds: Who Is Winning?**—Everyone loves a big, no-holds-barred industry turf war, and the fight over cloud computing dominance among vendors like Google, Amazon and Microsoft is a doozie. In this lively episode, leading cloud computing blogger Geva Perry and Novell Canada president Russ Chivaliet break down the battle and give their opinions on where businesses should place their bets.
3. **Going Rogue: Embracing the Cloud without IT’s Permission**—What does it mean when end users start adopting cloud applications without approval from their IT department? And does this trend fundamentally change the relationship between IT organizations and the organizations they serve? CloudChasers tackles these tough questions with outspoken Infoworld cloud computing blogger David Linthicum.

“The adoption of cloud is happening at an extremely rapid pace. There’s no doubt in my mind, this is the future of IT. [Enterprises and small businesses] need to know [which] players to bet on, which technologies to bet on and which standards to bet on.”

Geva Perry, author of the Thinking Out Cloud blog, on the April 22, 2010 episode of CloudChasers

> Join the Chase

Like any new podcast, CloudChasers is working hard to build an audience. “The best podcasts are almost always part of larger social media experience, and CloudChasers is no exception,” said Grant. “We want to create a social, interactive community around CloudChasers where conversations with our listeners take place before, during and after every episode.” To make that happen, Frank Days is taking advantage of all the podcast distribution and social media tools at his disposal. Cloud computing enthusiasts can listen live or subscribe to CloudChasers on iTunes or download pre-recorded episodes through Novell’s own mp3 feed. Listeners can also comment on the show and interact with guests through a [CloudChasers LinkedIn group](#) and [blog](#).

Of course, all those tools are worthless if people don’t like the show. “In the podcasting world, it’s all about content,” said Days. “Every podcaster, including CloudChasers, starts at ground zero. If the show is interesting, relevant and entertaining, word will get around, people will get involved, and CloudChasers will build an audience and become a strong voice in the cloud computing community.” Obviously, Frank Days and Matt Grant are strong believers. “We’ve worked very hard to put something interesting and worthwhile together,” said Grant, “and we’re proud of the results. Of course, it only takes a few minute for listeners to form their own opinions. We just hope everyone who is interested in cloud computing will check out CloudChasers and decide for themselves.” So far, this approach appears to be working. CloudChasers’ audience has grown steadily since its debut last April, and the show is getting ready to boost its listenership even more by joining a major syndicated online radio network. So look for a new episode of CloudChasers coming to a PC or mp3 player near you. And find out just how interesting and entertaining cloud computing can be. Check out CloudChasers today at www.novell.com/promo/cloudchasers.html.

Learning More // Novell Connection Magazine

Get Involved with CloudChasers

- [Listen live](#) every Thursday at 4:00 pm ET
- [Subscribe](#) to CloudChasers through iTunes
- Download and comment on episodes through the [CloudChasers mp3 feed](#)
- Visit our [main show page](#)
- Follow CloudChasers and ask questions via [Twitter \(#cloudchasers\)](#)
- Join the [CloudChasers LinkedIn group](#)

The Chase is On // Novell Connection Magazine

Inside Look // Novell Connection Magazine

The Man Behind the Mic

What does host Matthew T. Grant bring to CloudChasers? The best way to find out is to listen in. You'll find that Matt is smart, articulate, easy to listen to and the proud owner of a dry, understated sense of humor. He also brings a surprisingly diverse background to the show, from a PhD from Cornell to a varied career in many different industries and an impressive writing resume'. But the bottom line is that Matt always knows how to keep the conversations on CloudChasers moving in interesting, unexpected and entertaining directions.

Making Programs—and Companies—Talk to Each Other

Developments in Virtualization Continue to Push Interoperability

Microsoft, Windows, proprietary software. Novell, Linux, opensource. The visions could hardly seem more different. So when Novell CEO Ron Hovsepian and Microsoft CEO Steve Ballmer announced a new collaborative relationship between their two companies in a [press conference](#) back on November 3, 2006, people took notice. The landmark agreement between the two software giants covered three areas:

- improvement of the interoperability of SUSE with Microsoft Windows
- promotion of both products together with a cross-license on the use of patents, and
- an agreement on marketing.

Three years later, Susan Heystee, at that time Vice President and General Manager for Global Strategic Alliances at Novell, and Ted MacLean, then General Manager for Strategic Partnerships and Licensing at Microsoft, spoke again publicly about the state of the alliance and the two companies' continued efforts towards better interoperability. (See the 3rd Year Anniversary Video at [moreinterop.com](#).)

“Interoperability is a two-way street. If you want your products to talk to each other, you need your software providers to also talk to each other.”

Ted MacLean ([O'Reilly Media's Gov 2.0 Summit interview](#))

As part of these efforts towards interoperability, the two companies have been working together for better identity management, systems management and document format compatibility solutions. Success in these three areas will make it easier to manage mixed Windows and SUSE Linux Enterprise environments. (See “[Interop Ability Benefits](#)” at the joint Interop Web site.)

Perhaps above all, however, the efforts have focused on virtualization. They noted that the vast majority of IT Centers are heterogeneous, and run multiple operating systems and products for different platforms. They also said that interoperability efforts are increasingly focused on virtualization. Indeed, as almost everyone in the IT industry knows, server virtualization is the best way to combat the growing trend towards server proliferation. And we all know that more often than not, that produces very low overall utilization. Additionally, as the push grows for internal clouds, there is a real need for truly shared resources across platforms.

“We see this agreement with Microsoft as a fundamental step forward for Linux, and we’re strongly committed to driving Linux more firmly into enterprise environments, from the desktop to the data center,” said Susan Heystee, vice president and general manager for Global Strategic Alliances at Novell. **“For customers who want to use both Linux and Windows, we’re going to make life significantly easier. We know customers, partners and the community are looking for concrete results from this agreement.”** ([Novell's Susan Heystee to Oversee Microsoft Pact in Strategic Partner Role](#))

> **The Nuts and Bolts of Cooperation**

In April 2006, months before the November 3 announcement, representatives from Microsoft and Novell, met on the MIT campus to coordinate their efforts, primarily in the area of virtualization. The companies needed to produce code that would allow Windows programs to run on the Novell-preferred Xen hypervisor and Linux solutions to run on Microsoft's Hyper-V hypervisor. To that end, not only techs but support personnel were also brought in to help focus on what users really needed when it comes to virtualization.

Core to the alliance's interoperability efforts was the setting up of the Microsoft+Novell Joint Interoperability Lab (JIL). The JIL is a unique facility that develops original technologies while engaging with two different company cultures and that of the open source developer community. As a technical and collaboration bridge between the two companies, the JIL is a model for cooperation between open source communities and proprietary software development companies.

Some of the goals Novell and Microsoft are working towards at the Joint Interoperability Lab (JIL) are:

- more improvements of SUSE Linux Enterprise Server running on Hyper-V. While the stated performance is only slightly different today, the ultimate goal is parity with the operating system running on "bare metal" and no performance degradation when moving to SUSE Linux Enterprise Server on Hyper-V
- more Linux Integration Components developments, released to the Linux community
- System Center Management Packs will continue to evolve with support for advanced Linux support triggers and analysis tools, i.e., analyzing clusters, connectivity, security compliance, update deployments, etc.
- continued support of Novell for future enhancements to Moonlight, including support for some Silverlight v3 features

([See more at Novell and Microsoft: Building Bridges](#))

As noted in the [National Computing Centre's White Paper](#), the JIL focuses on new product development and testing for both partners. Its objective is to find and eliminate actual and potential cross-platform or cross-product interoperability problems before they get into products released to market.

> **Understanding Virtualization and Its Benefits**

Of course interoperability is best manifest in virtualization. Likewise, virtualization is increasingly the way to go in large, complex data centers that are striving to avoid server sprawl. Because most IT centers are heterogeneous, Xen on SUSE Linux Enterprise Server and Windows Hyper-V must be truly interoperational.

Simply put, virtualization is the execution of software in an environment that is separated from the underlying hardware. The process of virtualization replaces a direct interface between a user and the hardware with a software layer that communicates between the two. Virtualization provides a platform to present a logical view of physical computing resources to an operating system so that multiple operating systems may share a single computer, unaware that it does not have complete control of the underlying hardware. The common theme of all virtualization technologies is the hiding of underlying technical characteristics by creating a logical interface that is indistinguishable from its physical counterpart.

Because system virtualization creates many virtual systems within a single physical system, it provides the following:

- server consolidation
- dynamic provisioning
- virtual hosting
- reliability, availability and serviceability
- workload management

And so the benefits of virtualization are clear; system virtualization enables you to:

- consolidate systems, workloads and operating environments
- optimize resource use, and
- increase server flexibility and responsiveness.

> **Interoperability and Virtualization**

Both Xen and Windows Hyper-V use hypervisors, which are virtualization platforms that allows multiple OSes to run on a single host computer. The main job of a hypervisor is to provide isolated execution environments for the guest OSes and to control access to the underlying hardware.

There are two types of hypervisor:

- **Type I: (native)** - runs on the hardware without another layer between it and the hardware. It manages all the hardware resources for the various operating systems running on top of it.
- **Type II: (hosted)** - runs on top of a host operating system and is the communication layer between the host OS and the various guest operating systems that are running virtually on top of the Type II hypervisor. The host OS provides I/O sharing and a management interface to the hypervisor.

Whether you plan to deploy SUSE Linux Enterprise Server virtual machines as guests on Hyper-V hosts or Windows virtual machines as guests on Xen hosts, you need to understand what it takes to make those operating systems work with the best possible performance. Both virtualization technologies have interoperability components that you need to be aware of. These are:

- **Hyper-V Linux Integration Components (IC)** (See [Figure 1.](#)) This driver code was designed to enhance the performance and usability of Linux guest operating systems running on Windows servers so that Linux virtual machines can run in enlightened mode
- **SUSE Linux Enterprise Virtual Machine Driver Pack.** This driver pack contains paravirtual disk and network device drivers for several third-party operating systems. They enable hosting of the unmodified guests on top of SUSE Linux Enterprise Server 10 SP2 and Xen 3.2 or later.

> **The Unlikely but Revolutionary Collaboration**

Improving interoperability, especially in the area of virtualization, requires continued collaboration. The technical and support issues users continue to face with networking show the importance of the collaboration between Novell and Microsoft. Since the Novell/Microsoft relationship began almost four years ago, Novell has been writing drivers for Windows to run on top of Linux and Microsoft has been writing and maintaining drivers that allow SUSE Linux Enterprise Server to run

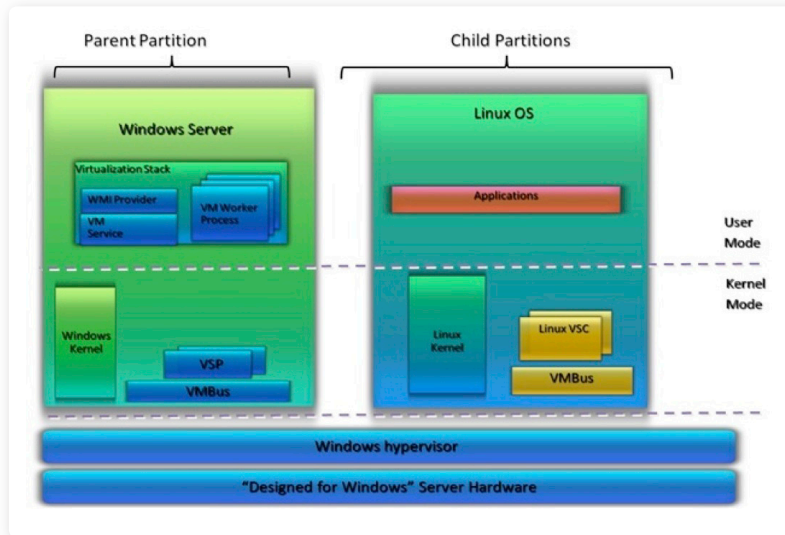


Figure 1: Hyper-V Linux Integration Components (IC) implementation with characteristics and functionality of the Linux IC modules, VMBus and VSCs.

on top of Windows. All along, work in the JIL has focused on fixing problems before software is released to provide truly seamless interoperability.

“The technical collaboration between Microsoft and Novell has enabled us to reduce overall costs while increasing cross-platform manageability. This means we’re able to deliver more consistent quality of service to our customers at the best price possible,” said Dr. Happy Sithole, Ph.D., director of South Africa’s Centre for High Performance Computing. **“In addition, migrating to the Linux support offered by Novell was a great decision for us. Not only do we get the interoperability advantages of seamless integration between our platforms, but we also have the confidence of knowing we are supported by two world-class vendors.”** ([“Microsoft and Novell Collaborate to Deliver Hybrid Options for High-Performance Computing”](#))

So what's in it for Microsoft and Novell to cooperate so closely in these areas? In the end, it is the question of “coop-etition.” While the two companies continue to compete in many areas, a pragmatic approach to the market requires that they work together to keep their customers satisfied. Because customers are using both platforms in heterogeneous data centers, working together in the area of virtualization but also in systems management and in office and document format compatibility, interoperability is good business, with joint customers growing by as much as 58 percent.