Novell e-Login

A single identity portal experience for Novell Customers

Novell Information Services and Technology

A Service Of Novell's Information Services & Technology Group

About this document

As IT guys ourselves, we see documentation, from every vendor, that's been sanitized and generalized. While their technical specs are accurate, they don't do the best job of showing you just how you're really supposed to use the stuff.

So, rather than the vague, generalized, and wholly fictional examples found in most documentation, we're going to tell you exactly how we use our own products to run our own company. We are not, after all, a small, tidy computing environment suitable for documentation. We are a big, sprawling, untidy computing environment made up of over 500 production servers and 20,000 workstations in 130 locations throughout the world. In other words, our problems are probably an awful lot like yours.

But please keep in mind that this document may be more than a little rough. It wasn't conceived by a committee, written by a committee, or approved by a committee, so it hasn't been edited, re-edited, tidied up, sanitized, and whitewashed. Don't think of this as an official Novell whitepaper. It's more like a beigepaper.

Table of Contents

Novell e-Login **1**
About this document 1
Introduction **3**
Problem – Too many static web sites 3
Solution – eLogin and the Customer Portal 3
Security and Access Control – iChain 2.0 3
Storage – eDirectory 8.6.2 3
User Management – Self Registration Application Code 4
Presentation – Novell Portal Services 1.5 (NPS) 4
e-Login Customer Portal Architecture **5**
Layer 4 Switches **6**
iChain 2.0 - Acceleration, Authentication, and Authorization **6**
Web Content Acceleration 6
Authentication 6
Authorization 7
Storage – eDirectory 8.6.2 ..... **7**
Scalability and Redundancy 7
Customization 8
User Management **8**
Self Registration 8
Legacy Applications 9
Self Registration Process Flow 10
Presentation – Novell Portal Services (NPS) **10**
Presentation – Novell Portal Services (NPS) **11**
Summary **11**

Introduction

Problem – Too many static web sites

The inefficiencies of looking for information in different applications, web sites, or databases, all with different access points and security, takes a heavy toll on productivity. Businesses are looking for web-based tools that bring web and application content to users so they can accomplish more in less time. People use personal portals because those portals know an individual's preferences, which simplifies web surfing, saves time, and makes life easier.

As Novell's external websites evolved and increased in complexity, it became necessary to restrict access to specialized content. This typically occurred for one of two reasons: either because Novell wanted to restrict access to generic premium content, or wanted to provide content customized for the individual. Business units began solving the problem for their particular customer populations, and today Novell has various databases of individuals and organizations, each containing anywhere from a few hundred to several hundred thousand records to meet the specific needs of each business unit.

Most Novell customers deal with more than one internal business unit. By creating separate islands for entitlement to each group's specialized content, Novell perpetuated a fragmented experience for its customers. In short, the problem is:

- Customers have many different sets of credentials when doing business with Novell on the web.
- Depending on Novell's relationship with an individual or an organization, several different unlinked digital identities could be required to authenticate when doing business online.

This problem is not unique to Novell. We bet you have similar problems in which your various web initiatives evolved separately. As a leading provider of Net Services software, Novell should provide a leadership role in the *complete* solution to this problem as the company aligns with strategic initiatives related to the eDirectory.

Solution – eLogin and the Customer Portal

Today at Novell, there are more than one million separate digital identities in over a dozen different relational databases. These are accessed by hundreds of applications to support the online authorization and authentication needs of Novell.com. It would be next to impossible to migrate those databases to a common Web Identity and Profile in a single cutover. Novell needed a way to begin working toward a common Web Identity and Profile that could be migrated over time, but that would still provide value to the end user and to the corporation. In short, the solution is to:

- Map the core components of each of the separate entitlement systems into a common Web Identity and Profile to be stored in Novell's eDirectory.
- Deploy an environment that allows each of the disparate systems that comprise Novell.com the ability to consume this new common Web Identity and Profile, which provides both a single identity and single sign-on.
- Provide a framework for web application developers to migrate from their existing identity systems to the common Web Identity and Profile.

The following are used to deliver a complete solution to this problem.

Security and Access Control – iChain 2.0

- Establish a web architecture that simplifies, accelerates, and secures user credentials.
- Design an architecture that enabled load balancing and fault tolerance.

Storage – eDirectory 8.6.2

- Establish a common eDirectory schema for the storage and retrieval of customer profile information for use on the Web.
- Enable the capturing of information needed to facilitate access to legacy systems.

- Create an eDirectory tree design to facilitate the scalability and performance of Novell's web-based applications globally.
- Design replication for load balancing and fault tolerance.

User Management – Self Registration Application Code

- Create a self-registration process that will allow customers to create a WebIDUser and associate with legacy entitlement systems that have been made WebIDUser aware.
- Ensure that applications that are made WebIDUser aware will support both the legacy ID and the new common WebIDUser until all of their users are migrated
- Provide single sign-on among the separate systems in Novell.com that are made WebIDUser aware.
- Target key applications that will be migrated in the first phase to the new common Web Identity and Profile (WebIDUser). The WebIDUser was created to be a thin object that holds application-related information, as opposed to the standard NDS user object which was created as a LAN/WAN user that is part of the company.

Presentation – Novell Portal Services 1.5 (NPS)

- NPS automatically manages content presentation and overall portal administration based on each user's business context, creating a personalized, secure environment in which users can do business faster and smarter.
- NPS aggregates and integrates content, applications, multiple portal environments, and Net services into a single business experience accessible from anywhere.
- NPS leverages existing systems as part of a customized solution built around Novell's business needs.

The following diagram shows how using an L4 switch, Novell eDirectory, Novell iChain, Self-Registration, and Novell Portal Services components are configured to support Novell's e-Login and Customer Portal solution.

e-Login Customer Portal Architecture

Web User

xyz.novell.com

FIREWALL

Layer 4 Switches

Customers access service names bound on switch

Provides Load Balancing and Failover for iChain boxes

FIREWALL

iChain 2.0 Session Broker

Authenticated customer sessions are coordinated between iChain boxes through Session Broker

iChain 2.0 Proxy

Provides Authentication, Access Control, and content Acceleration

Multiple iChain boxes for scalability and redundancy

Content

User Attributes

Origin Web Servers

Mixed environment web servers (NetWare, Unix, NT)

Provide source web content

Novell Portal Services

Provide customized content to customer

Multiple replicated NPS servers

Self Registration

Application to allow customers to create their own profile in the Directory

User Authentication, Access Control, Attributes

Database Read/Write

Custom Application Attributes

Customization Attributes

User Creation

Business Databases

DirXML

eDirectory Customer Directory

Stores customer and company profiles and attributes. Stores business rules.

Consists of multiple replicated eDirectory systems

Layer 4 Switches

Traditional Layer 2 switching can forward packets based on MAC address, but Layer 4 switches can look deeper into the IP protocol and forward packets destined for the same MAC address to different ports or IP addresses based on a load-balancing algorithm. Novell eLogin uses an L4 switch architecture to provide transparent load balancing and failover between several redundant iChain servers.

In this scenario, the user's browser sends a request to a single virtual service IP address on the switch, which in turn gets directed to the least busy iChain server. Once a user connects to a specific iChain server, all subsequent requests are directed to this same server. In some instances, a new connection by the same user may get serviced by a different iChain server. IChain handles this by using an iChain Session Broker, which allows authentication session information to be shared between iChain servers.

In this way, the scalability of the iChain architecture is limited only by the number of ports available on the Layer 4 switch. Additional iChain systems can be added as traffic loads increase. This allows for an extremely scalable architecture that is fully load balanced and redundant.

iChain 2.0 - Acceleration, Authentication, and Authorization

Novell iChain 2.0 utilizes a proxy architecture to securely provide web acceleration, user authentication, and access control. As a proxy, it front ends all requests to and from the actual web servers. The proxy architecture offers several advantages in that authentication and access rules can be centrally managed and applied. iChain protects the company's web servers by rendering them essentially invisible: All inbound requests are directed to the iChain servers and all outbound information appears to originate from the same servers. It also integrate very easily with an existing web architecture because it can interact with any platform of web or application server and does not require modifications to the web server layer.

Web Content Acceleration
The Novell iChain Internet Caching System server operates in reverse proxy mode (also called *accelerator mode*). A reverse proxy accepts requests for web site data from external users and delivers most of that data from its own cache. The proxy contacts the web server hosting that data only when the data in its own cache is out of date or when additional processing, such as a database lookup, is required. This greatly conserves network bandwidth and provides quick access to content using iChain's optimized caching file system.

An additional proxy feature of iChain utilized in Novell eloign is that it can serve up content from a non-secure web server as secure SSL content. Since this "SSL-izing" is done on the iChain proxy, it is unnecessary to purchase individual web server certificates or separate secure web servers in order to provide secure content. Both secure and non-secure content can be served from a single non-secure web server.

Authentication
Novell iChain uses Secure Sockets Layer (SSL) to pass encrypted credentials from the browser to the iChain service. The user provides a username and password, which is then validated by iChain against the eDirectory LDAP directory. All authentication is done against the eDirectory using the secure LDAP protocol. Because all web authentication occurs through the iChain layer, iChain can determine the users authenticated state and provide a unified, single-sign-on user experience across different web sites.

Specifically, the iChain Authentication process works as follows:

- A user makes a request for web site data, and iChain checks its rule set to determine if the URL requested is defined as a public or a secured URL.
- If the URL requires authentication, the iChain server checks its authentication table to determine if the user is already authenticated. It may also consult an iChain Session Broker, which maintains authenticated session information provided by other iChain servers.
- Assuming the user is not already authenticated, iChain presents the user with a secure login screen served over SSL.
- The user enters authentication credentials, and iChain validates them against the directory.
- If successful, iChain sends a session cookie to the user's browser, which the browser stores in memory. This session cookie does not store any information about the user, but a key which is used to maintain authenticated session state with the iChain server.

Authorization
Once a user is authenticated, iChain checks the rules defined in eDirectory to determine whether or not the person is authorized to access the particular URL being referencing. IChain can enforce access control based on a users group or container associations in the directory. With iChain 2.1, it can also enforce access control based on any combination of attribute values that exist for the user. IChain caches these rules in its cache for improved performance.

Access can be controlled down to a very granular level, based on the URL pattern. This allows control to be applied down to specific pages, or even specific parameters passed to a page; however, this doesn't mean every page requiring access control must be listed. Wildcard options can be used to grant access to broader areas of the web site without having to specify the URL to each page within an area. For example, a rule that includes an asterisk (*) in the URL, as in www.novell.com/*, grants users associated with this rule access to all pages under www.novell.com. Access Rule objects also have an exception list, which allows additional security by disallowing access to users listed as exceptions. For instance, an entire web site can be defined as a public site with the exception of a few pages that require authentication.

In many instances, a web server may have an existing application that has its own proprietary authentication mechanism. It may require its own access control that is based on an existing database that contains users and entitlement information. In this case, iChain can retrieve the required authentication information for the user from the eDirectory and pass it on to the application. The user can then be given access to the application seamlessly. This way, existing “legacy” applications can be integrated very quickly and with minimal changes to the back end application and database. Novell eLoign integrates several of these types of existing applications to provide single-sign-on across many different web sites.

The ability for iChain to retrieve user attributes from the directory and pass them to an application is also very valuable in streamlining the user experience. For example, forms can be pre-populated, or even eliminated altogether as the pertinent information can be read directly from the directory.

Storage – eDirectory 8.6.2

In order to implement a world-class solution, Novell considered key functional areas where the directory is superior to other forms of data storage. The following is a discussion around those considerations.

Scalability and Redundancy
The directory must scale well to handle rapid and constant growth. Not only will the directory need to scale to millions of users, but it must be able to handle thousands and potentially tens of thousands of hits per minute. The directory must be redundant and be able to load balance in order to provide near-zero wait time response.

The system must have redundant fail-over capability so that there is no single point of failure. Each instance of the directory around the world must have similar redundancy, as the system must be available at all times. There cannot be a single point of failure once the system is running in a production environment.
LDAP is the access protocol that will communicate to eDirectory. It can also be used for management purposes. It enables the simplest form of directory management for disparate data stores. It is industry standard and Novell is a strong supporter of the directory protocol.

Customization
Novell created a custom NDS object called webidUser that contains the base user attributes such as name, address and phone number. Additional user attributes are attached to the webidUser object as AUX classes. These additional AUX classes hold credential data for the legacy applications, entitlements and rights to trusted systems and other additional down steam application values. A webidUser may or may not have AUX classes associated with their user. Using AUX classes allows for flexibility in managing the data stored in the directory.

Some applications cannot rely on eDirectory for all profile information and will maintain their own data store in addition to what is stored in eDirectory. If the data in one data store, or the other, is modified by the user and is accessible by each data store, it will need to be synchronized. DirXML drivers were written in order to synchronize user data stores with legacy applications. This allows event driven synchronization to go one way or bi-directionally. When a user performs self-management on profile information (e.g. home address, email address, etc.), the information propagates to the entitlement systems that require synchronization. Novell used event triggers to capture modifications in the application's data store, and the DirXML connectors are the best tools to use for this level of synchronization.

User Management

Self Registration
Novell determined that it was not efficient to create the user object from the legacy application data and try to combine the duplicates, and therefore determined to allow the user to create their identity, map themselves to their application credentials, and prove authenticity through the use of self-registration. Self-registration is the simplest and quickest way to populate the eDirectory with any sort of accuracy. The legacy entitlement systems require different credentials that must be captured by self-registration. These credentials will be tested prior to initial inclusion in the eDirectory.

The customer has visual contact to the results of e-Login in three key areas:

1. **Self-registration** – Customers are be allowed to self-register in order to participate in single point authentication of the eDirectory. Self-registration consists of a web form designed by Novell, which captures required profile information and authentication credentials for legacy entitlement systems to which the user has been granted access. Self-registration validates authentication credentials prior to storing them in the eDirectory as part of the user's webidUser. It also allows adding authentication credentials to the eDirectory in situations where the user has been granted new access to an application that he/she did not previously have when the webidUser was originally created. Self-registration allows users to change their identity information and application credentials, and allows for forgotten username, forgotten password, and change password functionality with out having to contact Novell.
2. **Login** - Customers will be prompted to login to the eDirectory whenever a participating entitlement system is accessed on a per-session basis, or they can login upon accessing the website and be taken to the Welcome Back page to view the links to the systems to which they are entitled. The login form will prompt for eDirectory authentication credentials which the user selected during self-registration.

3. **Updating profile information** - Customers will be allowed to modify select profile attributes of his/her WebIDUser such as password, address, and so forth.

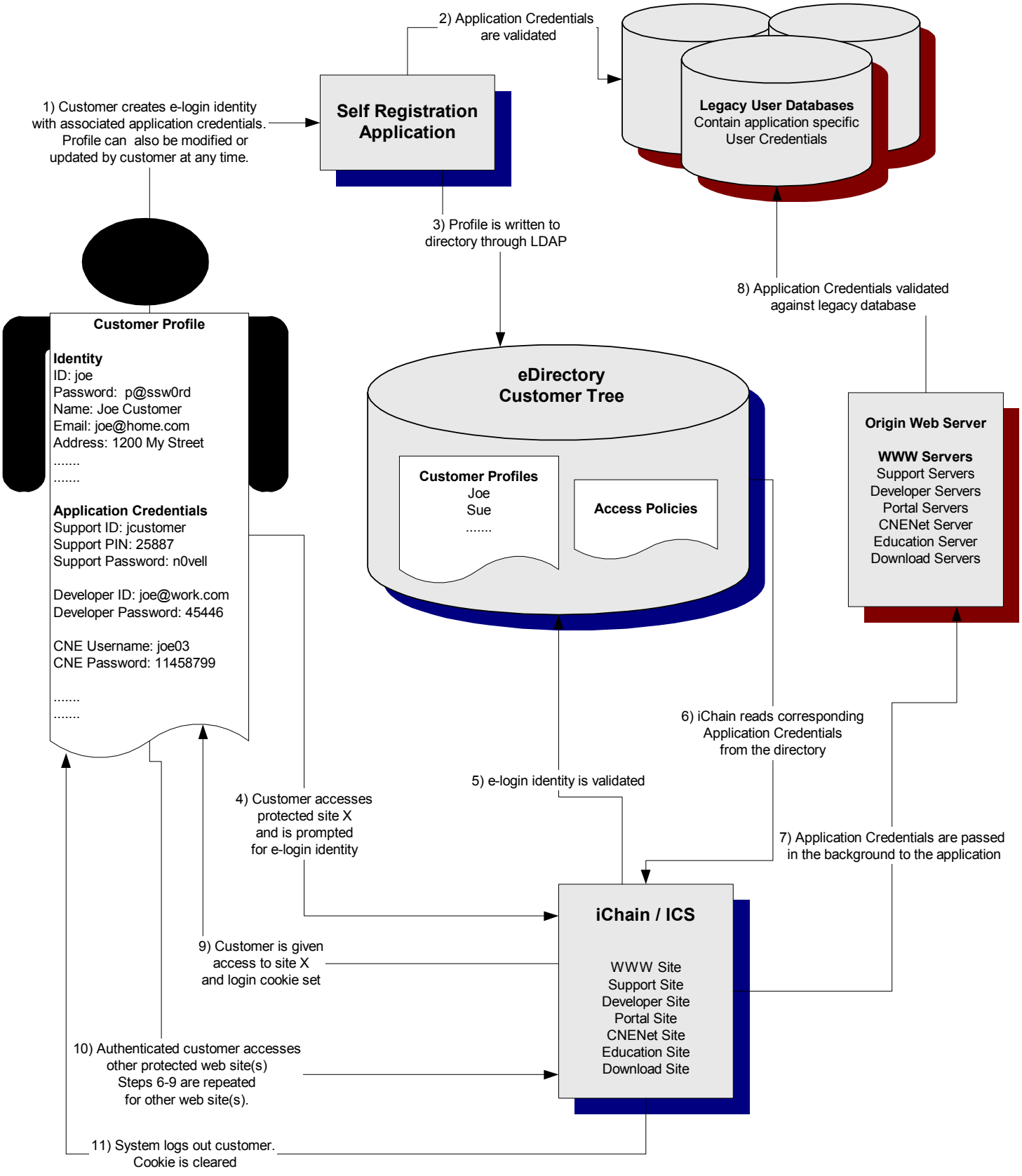
Customers will no longer experience the disparate login systems once authenticated to the eDirectory.

The following rules were considered for user management within the Self-Registration code:
1. User must be able to modify password.
2. User must be able to submit changes to remaining WebIDUser fields.
3. User interface for password modification must be simple.
4. User must be able to get their user name via email and change their password if they know their username and challenge string answer.
5. User must have access to some form of help, electronic, help button, live, etc.

Legacy Applications
The following rules were considered for deciding which legacy applications become WebIDUser aware:
1. Developers on staff to make modifications to accept authentication credentials
2. Developers on staff to export profile information to desired format. This will allow for easier tree population.
3. Each group has direct access to the web interface.
4. Application can accept name value pair, or application can query DS for the authentication credentials it needs.
5. User creation, modification, and deletion functions fall within scope of eDirectory functionality.
6. Application owners have testing staff available to validate implementation.

Self Registration Process Flow

The following diagram illustrates the process flow for Novell's Self Registration solution.



Presentation – Novell Portal Services (NPS)

Three big differences exist between a static website and a portal.

The first difference is typical content that is displayed as fairly static Web pages, much like this white paper. Although this paper could be presented on an intranet, the content will not change unless the author comes back and changes it. Additionally, in order to find this white paper again, you would have to search for it on your intranet or bookmark it.

The second difference is islands of content. If you want the right information, you have to search at length for it, authenticate with yet another password and then compare it with other slices of information elsewhere on your site to do any type of analysis. There is no 'one' view into your information.

The third difference is customization. A portal provides a view into dynamic, customized content, which is presented on the Web as if it were a Web page. For example, your portal might have a view into sales information, which changes daily. The portal could present the sales view with other views into pricing and inventory databases. When you click on a tab or a navigation icon, you see another page with personal information about your support incidents along with education events that interest you. Another tab or navigation icon on the portal could take you to the Web applications that you use. By logging into the portal, you also log into these integrated applications. And of course, a portal can provide updated information or news feeds about points of interest you may have about specific Novell products.

The key components for building a business contextual portal include:

- Security based on business context
- One-step authentication
- Integration of information and business processes
- Custom solution services
- Personalization

The first step in setting up a portal solution and streamlining business processes for Novell was to tie the portal to information stored in the existing business directory. The next step was to further integrate other databases with the directory to create a very data rich and dynamic data store for portal customization. The third step was to create a template for each business unit within the company. With that framework in place, the business units could define the information and applications to be directed to users with the appropriate credentials.

Summary

The problem of too many websites within a company can be minimized using the technologies and processes outlined in this paper. Using Novell Net Services to provide the storage, security, and presentation layers of the architecture, an organization can provide a customized web experience to improve productivity and communication across multiple channels.

Enabling a business to customize content for end-users allows for better targeting of information and provisioning of eBusiness applications. Enabling end-users to personalize their experience on a website promotes satisfaction and facilitates transactions. Novell provides heterogeneous products that compliment existing architectures and enable organizations to increasingly do business on the Web.