

# Integrating Linux and UNIX Systems with Novell® Identity Manager

**Jeremy Grieshop**

IDM Developer

[jgrieshop@novell.com](mailto:jgrieshop@novell.com)

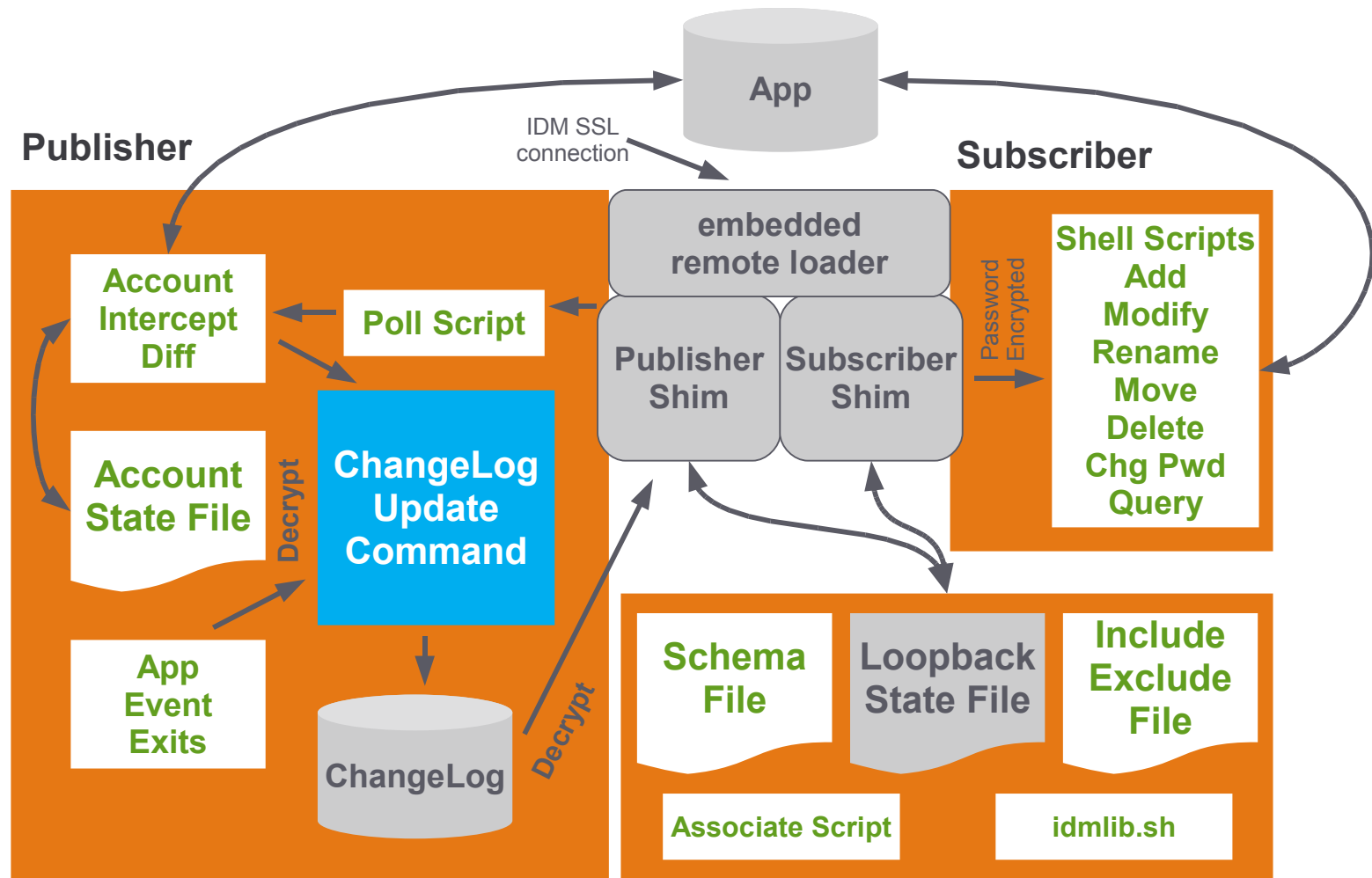
**Novell®**

# Agenda

- Options For Integration
  - Bidirectional Solution
    - > Architecture Overview
    - > Installation and Configuration Options
    - > Customization
  - Fan-Out Solution
    - > Architecture Overview
    - > Customization
    - > Passwords and Authentication
    - > Full Account Redirection with Name Service Switch
  - Linux and UNIX User Settings
    - > POSIX Management and LUM Settings
  - New Enhancements

# Bidirectional Solution

# Bidirectional Driver Architecture



# Supported Linux & UNIX Systems

- Configurations
  - Files (/etc/passwd, /etc/shadow, /etc/group)
  - NIS
  - NIS+
- Supported Operating Systems
  - Linux (RedHat & SUSE®) 32-bit or 64-bit Intel
  - Solaris 8, 9, 10 and x86
  - HP-UX 11 and 11i
  - AIX 5.1, 5.2, 5.3

# Schema

- Schema defined by RFC 2307
  - Auxiliary Classes and Attributes:
    - > posixAccount:
      - » cn:x:uidNumber:gidNumber:gecos:homeDirectory:loginShell
    - > posixGroup:
      - » cn:!:gidNumber:memberUid
    - > ShadowAccount:
      - » cn:authPassword:shadowLastChange:shadowMin:shadowMax:shadowWarning:s  
hadownInactive:shadowExpire:shadowFlag

# Subscriber

- The Subscriber consists of:
  - Subscriber Identity Manager Policy Rules
    - > Create Rules, Transforms, and Data Mappings
  - Connected Driver Shim
    - > Remote Daemon handles Communication and calls Scripts
  - Scriptable Framework
    - > A set of shell scripts designed for Files, NIS or NIS+
    - > Shell scripts call administrative commands such as useradd, usermod, userdel, passwd, etc.
    - > These scripts may be extended to perform additional administrative tasks such as home directory setup, profile configurations, rsync commands, etc.

# Publisher

- The Publisher consists of:
  - ChangeLog
    - > Implemented as a set of UNIX files
    - > ChangeLog tool for submitting events
  - Pluggable Authentication Module (PAM)
    - > Passively intercepts local password changes
  - Poll Script
    - > Run on configured intervals for detecting changes to local system
    - > Account Snapshots are compared to live data to reconcile changes
  - Identity Manager Publisher Policy
    - > Create Rules, Transforms, and Data Mappings

# Installation

- Metadirectory Installation
  - The LinuxUnix.xml can be imported using iManager or Designer.
  - The schema extensions for posixAccount, posixGroup and shadowAccount may be installed by the Identity Manager installation or using from the nxdrv.sch file.
- Platform Installation
  - A native self-extracting installer is executed.
  - The SSL and Loader/Driver passwords are configured.
  - PAM module may be optionally installed.
  - The driver shim is started from a Unix startup script.

# Configuration

- Basic/Advanced
  - > “Basic” option configures for a simple, default setup.
  - > “Advanced” option allows for advanced options such as Role-Based Entitlements, specific Posix management mode, and specific dataflow.
- Dataflow
  - > Controls the flow of data: Subscribe, Publish or Both
- Posix Dataflow
  - > Controls the management of Posix data: Vault, Application or Both
- Entitlements
  - > Configures policies to operate under RBE or Approval Workflow mode
- Database Type
  - > Files (/etc/passwd), NIS or NIS+

# Configuration (cont'd)

- Base Container
  - > Specifies a container for scoping events for Users and Groups
- Remote Host and Port
  - > Specifies the Linux or UNIX system Host and Port information
- Generate gecocos
  - > A “gecos” or comment field is created from the User's name and surname
- Lower-Case CN
  - > The CN from the Identity Vault is lower-cased on the Linux/UNIX system
- Use SSL
  - > Choose to use SSL or plaintext for communication

# Customization

- Shell Scripts

- The Linux & UNIX Driver Shim calls a series of Bourne shell scripts on the local system, each of which may be modified to suit your provisioning needs:
  - > subscriber.sh is called on every subscriber event
  - > poll.sh is called on the polling interval
  - > globals.sh is “sourced” into the other scripts, providing a central location for configuring global variables and settings.
  - > idmlib.sh is a library of shell functions used to access and update information from the driver shim.
  - > The files add.sh, modify.sh, delete.sh, rename.sh and query.sh all react to the corresponding event from the Identity Vault

# Customization (cont'd)

- Customizing Subscriber Shell Scripts

```
COMMAND="adduser $loginName"
```

```
EXEC $COMMAND
```

```
if [ $rc -eq 0 ]
```

```
then
```

```
    # ... custom commands
```

```
fi
```

# Customization (cont'd)

- Customized Publishing

- The Linux & UNIX Driver provides a changelog that can be accessed using the changelog utility, nxclh

- > Add Event Example:

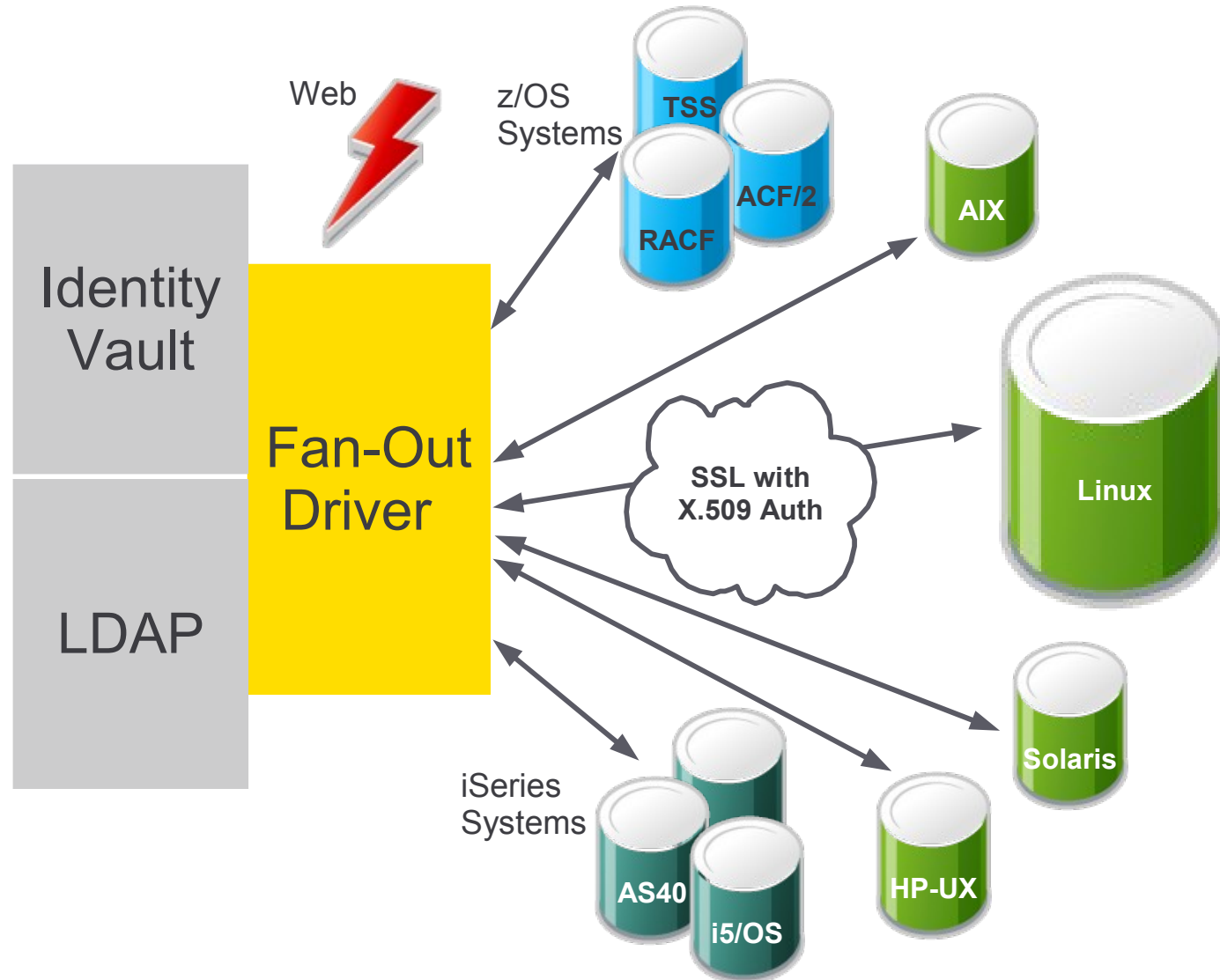
```
nxclh -t add -c User -a bobUser <<EOF
ADD_cn=bob
ADD_surname=Smith
EOF
```

- This tool may be invoked by command-line or from a script
- The events are encrypted and queued in `/usr/local/nxdrv/changelog/`

The background of the slide is a solid blue color with a pattern of thin, light blue lines radiating from the right side towards the left, creating a sense of motion or expansion.

# Fan-Out Solution

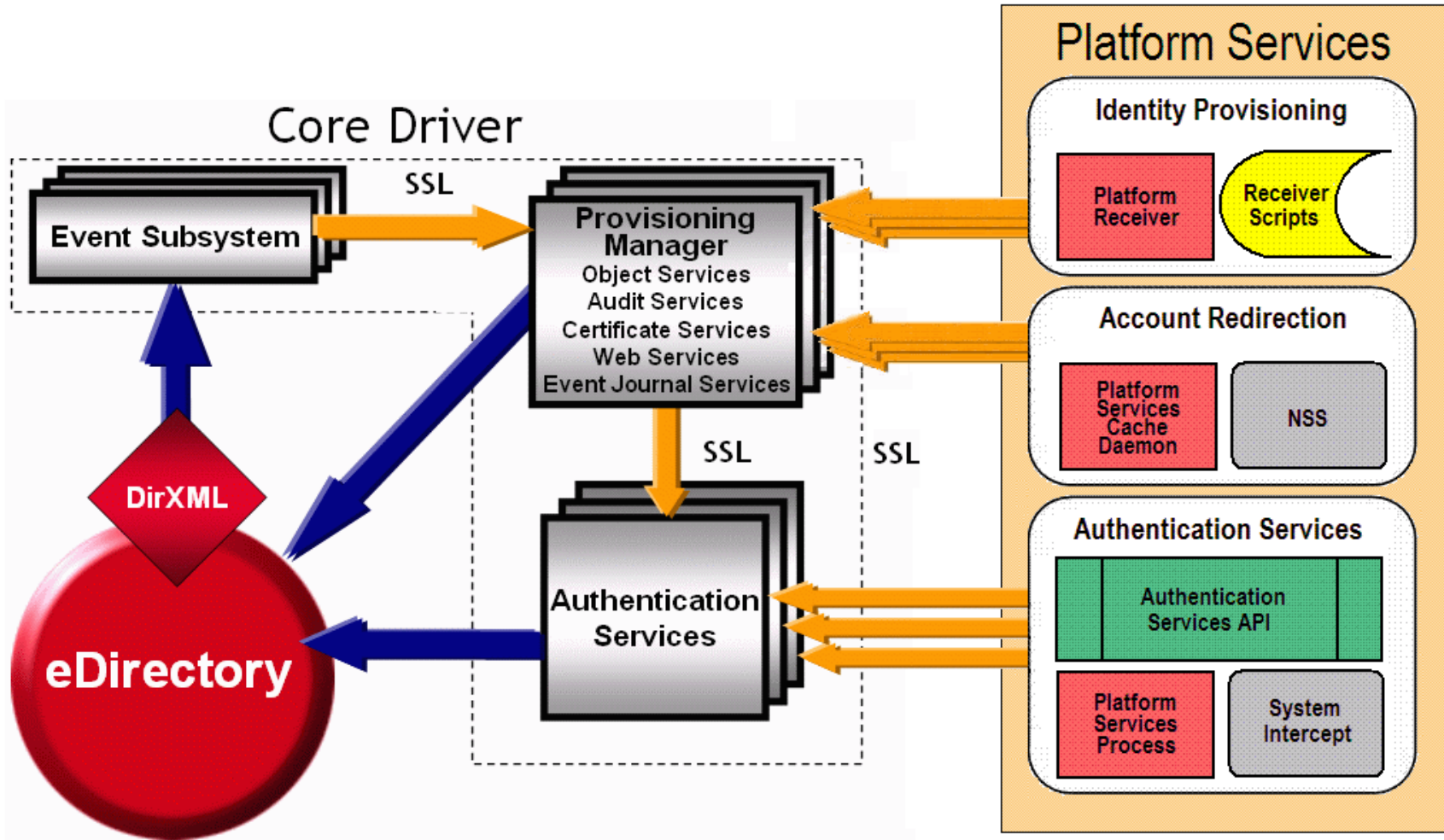
# Fan-Out Driver Overview



# Supported Systems

- Supported Configurations:
  - Files (/etc/passwd, /etc/shadow, /etc/group)
  - Additional scripts are provided for RBAC
- Supported Operating Systems:
  - Linux (RedHat & SUSE®) 32-bit and 64-bit
  - Solaris 8, 9, 10 & x86
  - HP-UX 11 and 11i
  - AIX 5.1, 5.2, 5.3
  - FreeBSD 5

# Fan-Out Architecture



# Fan-Out Subscriber

- Event “Snapshot” Model
  - Events are not sent to the platforms. Instead, a current snapshot of the object is sent out when changes occur
  - Timestamps determine which objects to process
- Platform Receivers
  - Receivers on the platform may be configured to retrieve these snapshots in three modes: Persistent, Polling, Scheduled
- Shell Scripts
  - Bourne shell scripts are used to react to events and determine the appropriate administrative actions

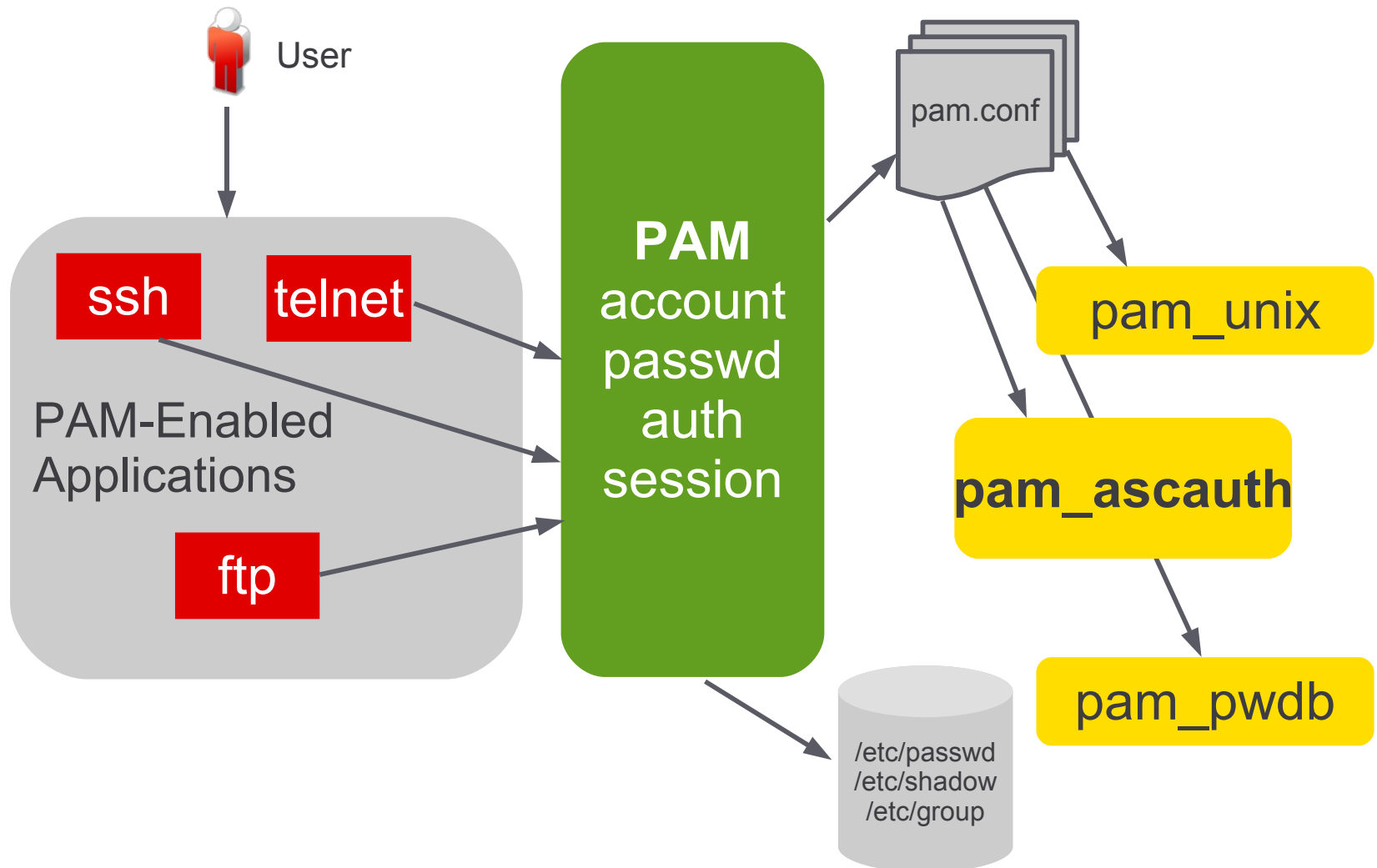
# Fan-Out Publisher

- Passwords
  - Password changes are the only events that are published to the Identity Vault
  - Pluggable Authentication Module (PAM) or Loadable Authentication Module (LAM, on AIX) are library modules used to send password changes and enforce password rules based on your Universal Password policies
  - These changes are performed real time, not queued events

# Authentication

- Two Modes:
  - Password Replication
    - > Passwords are synchronized to local system (/etc/shadow)
    - > All interactive logins interact with local system security
  - Authentication Redirection
    - > Passwords are stored in Identity Vault, not local system
    - > Interactive logins are configured against the Identity Vault, using PAM
    - > Multiple drivers may be used for failover and load balancing
    - > Platform Services Process provides local caching and SSL connection pool management

# Pluggable Authentication Module Architecture



# Configuring PAM: Pluggable Authentication Module

- The Fan-Out platform services provides a PAM module system library, `ascauth`, which implements PAM functions required to perform authentication checks and password changes.
  - `/etc/pam.d/`
    - > On Linux hosts, this directory contains a set of files, each specific to an application on the local system.
  - `/etc/pam.conf`
    - > On other UNIX systems, this file contains the configuration for PAM for all applications on the local system.

# PAM Example: Auth Redirection

```
# /etc/pam.d/sshd on sles9: Authentication Redirection
#
## If user can authenticate with pam_ascauth.so, auth stage is done, else
# standard auth modules are invoked.
auth sufficient pam_ascauth.so stats debug
auth required pam_unix2.so use_first_pass # set_secrcp
auth required pam_nologin.so
auth required pam_env.so
#
# managed users who are disabled in eDir are also disabled on platform
# when pam_ascauth.so is required in account stanza. pam_ascauth.so
# returns PAM_IGNORE for unmanaged users in account stanza.
account required pam_ascauth.so stats debug
account required pam_unix2.so
account required pam_nologin.so
#
# managed users whose passwords are expired in eDir are forced to
# change their password.
password sufficient pam_ascauth.so stats debug
password required pam_pwcheck.so
password required pam_unix2.so use_first_pass use_authtok
#
# no need for pam_ascauth.so in session stanza
```

# PAM Example: Local Authentication

```
#  
# /etc/pam.conf fragment, solaris 8: Local Authentication  
#  
# All users use standard modules for local password change, managed  
# users' passwords are reflected back to eDirectory.  
#  
other password required pam_dhkeys.so.1  
other password requisite pam_authtok_get.so.1  
other password requisite pam_authtok_check.so.1  
other password required pam_authtok_store.so.1  
other password required pam_ascauth.so.1 use_first_pass
```

# PAM Example: Account Redirection

```
#  
# /etc/pam.conf fragment, solaris 8: Account Redirection  
#  
# managed users are done when they change their eDir password,  
# unmanaged users are subjected to subsequent modules...  
other password sufficient pam_ascauth.so.1  
other password required pam_dhkeys.so.1  
other password requisite pam_authtok_get.so.1  
other password requisite pam_authtok_check.so.1  
other password required pam_authtok_store.so.1
```

# Configuring LAM: Loadable Authentication Module

- The Fan-Out platform services provides a LAM module system library, LAM, which implements LAM functions required to perform authentication checks and password changes for AIX systems.
  - /etc/security/methods.cfg
    - > On AIX hosts, this file contains the configuration for LAM on the local system.
  - Note: PAM is also available for AIX 5.2 and higher.

# LAM Example: Account Redirection

- Add a stanza for ASCAUTH (the AIX Account Redirection LAM) to `/usr/lib/security/methods.cfg`:

```
ASCAUTH:  
program = /usr/lib/security/ASCAUTH
```

- Edit `/etc/security/user` and change the default settings for SYSTEM and registry:

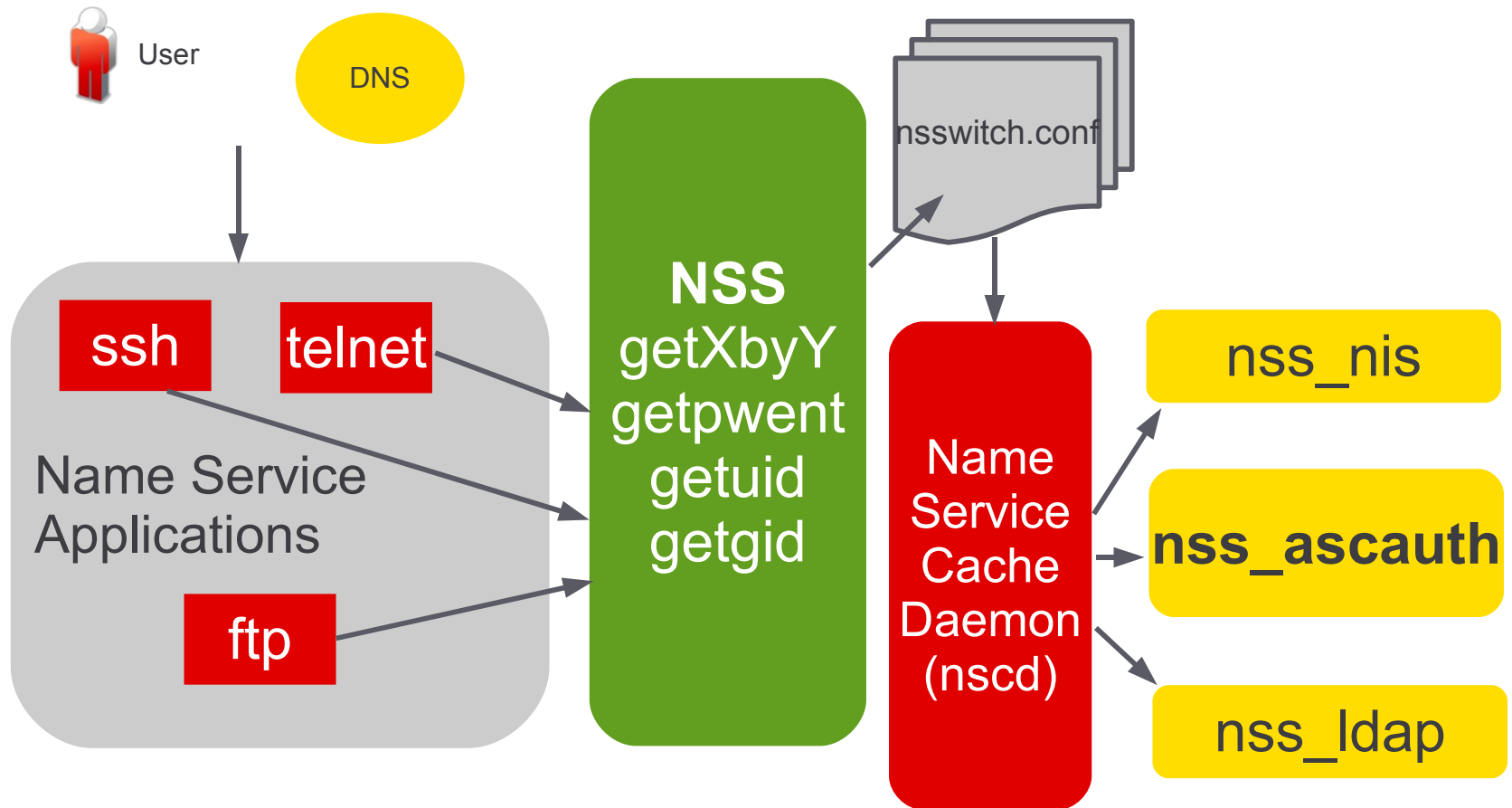
```
SYSTEM = ASCAUTH  
registry = ASCAUTH
```

- Any non-managed users need to have the SYSTEM and registry entries in their personal stanzas set to reflect the appropriate LAM, or “compat”

# Account Redirection

- No Local Accounts
  - Users and Groups are not synchronized to `/etc/passwd` and `/etc/group`
  - Accounts are virtually provisioned, real time, when the User logs onto the local Linux or UNIX system
  - The Fan-Out Platform Services Pieces:
    - > Name Service Switch (NSS)
      - » This library implements an interface provided by UNIX to retrieve information about Users and Groups stored in a custom location (Identity Vault)
      - » The file `/etc/nsswitch.conf` is used to configure Fan-Out NSS for Users or Groups
    - > Platform Services Cache Daemon
      - » A local UNIX daemon that caches events from the Fan-Out driver into local memory and provides the NSS with attribute data

# Name Service Switch Architecture



# Account Redirection (cont'd)

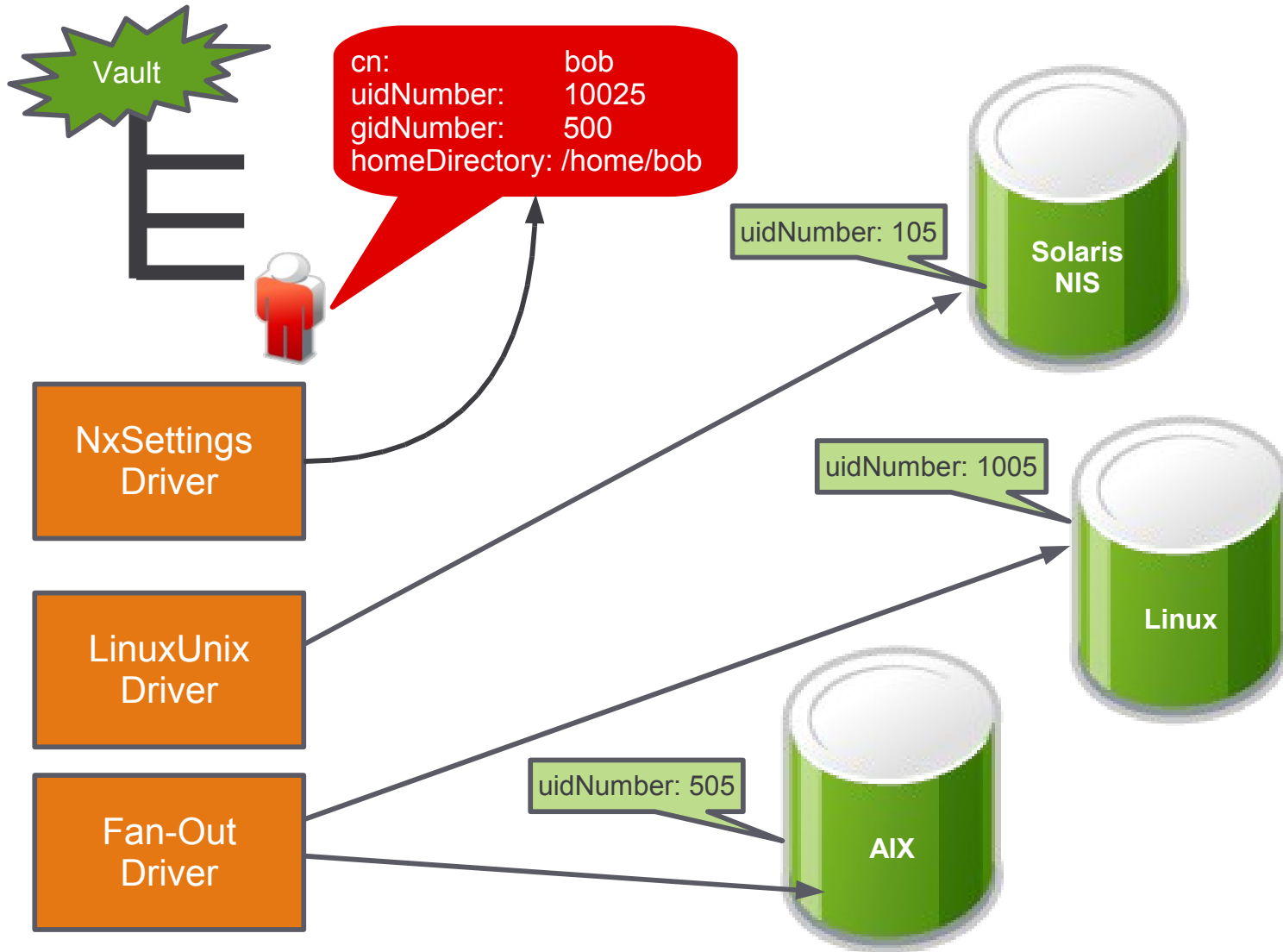
- How UNIX Attributes are Populated
  - UID's and GID's use the Platform Set's UID/GID set
  - Alternatively, UID's and GID's may be taken from the RFC 2307 Posix Account and Group auxiliary classes
  - Posix information may be populated manually or by the Linux & UNIX Settings Driver automatically
- Home Directories
  - Home directories must exist on the local system
  - Automounting home directories can be done with NFS automount or with a CIFS Samba module

# Configuring NSS

- `/etc/nsswitch.conf`
  - This file provides the configuration of NSS on the local Linux or UNIX system.
  - By default, systems are set up for “files”, “nis” or “ldap” for name service.
  - The Fan-Out driver provides an alternative, `ascauth`.
- The Name Service Cache Daemon (`nscd`)
  - This daemon keeps cached entries name service switch modules to provide quick retrieval.
  - The configuration is stored in `/etc/nscd.conf`
  - The Fan-Out NSS does require this service, because it provides its own caching mechanisms.

# Managing Posix Information

# Managing Posix Information



# Assignment of Posix Information

- iManager
  - Use the iManager Plug-In to assign Posix information to new and existing users
  - This is a manual process for a Help Desk role
- LDIF
  - Import Posix information onto existing users
  - This process must be scheduled for new users
- Linux & UNIX User Settings Driver
  - Identity Manager Migration and Create Rules to auto-populate Posix information for all new and existing Users and Groups
  - Complete UID/GID management based on configurable ranges and Identity Manager Policy logic

# Bidirectional and Fan-Out Comparison

# Fan-Out vs Bidirectional

- Fan-Out Advantages

- Scalability

- > Hundreds of connected platforms may be configured for a single driver

- Authentication:

- > Accounts and Passwords may be checked in real-time
    - > Accounts and Passwords may be always stored centrally

- Mixed Environment:

- > A single IDM solution for a variety of connected platforms, including UNIX, Mainframe and Midrange systems

- Logging:

- > The Fan-Out driver provides out of the box audit and operational logging facilities

- Built-in Time-Based Events:

- > Pending Deletes and Expiration dates may be synchronized

# Fan-Out vs Bidirectional

- Bidirectional Advantages
  - Bidirectional Data Flow
    - > Provides complete bidirectional data flow for all information, including custom data fields
  - IDM Policy
    - > The IDM Policy Builder may be used to build rules and policies for mapping, transforming and synchronizing data
  - Simple Configuration
    - > There are far less pieces and technologies involved
    - > Security is easy to configure

# Linux & UNIX User Settings

# Driver Architecture

- LoopBack Design
  - Posix and LUM Information are added to new and existing Users in reaction to create and modify events from the Identity Vault
  - Migration or Real-Time Events may trigger the loopback process
- Policy Builder
  - The rules for information population are controlled by Identity Manager Policies
  - Java extensions are included to retrieve and update data from the NxSettings configuration data

# Default Features

- Posix Management
  - The RFC 2307 posixAccount and posixGroup auxiliary classes are automatically populated in the Identity Vault
    - > uidNumber, gidNumber, homeDirectory, gecos, loginShell
  - Range values may be specified for uidNumber and gidNumber
- LUM Management
  - When using Linux User Management (LUM) with OES, the Settings Driver may be configured to automatically enable users for LUM and Samba access based on Identity Manager rules

# Use Case Scenarios

# Working Together

- A Bidirectional Use Case:
  - > 1. A User is created in the Identity Vault
  - > 2. Identity Manager policies allow this user to obtain UNIX attributes
  - > 3. Identity Manager policies configure this particular user for a default loginShell and gecos field. The uidNumber is taken from a preconfigured pool
  - > 4. The bidirectional driver processes this user and sends all the appropriate information to the local platform shim
  - > 5. The driver creates an entry in /etc/passwd and /etc/shadow and creates a local home directory for this user
  - > 6. The user logs onto the system and acquires the UNIX attributes set by the Settings driver and synchronized by the bidirectional driver
  - > 7. The user changes his or her password with passwd and the change is queued up and replicated to the Identity Vault using PAM and the changelog facility

# Working Together

- A Fan-Out Use Case:

- > 1. A User is created in the Identity Vault
- > 2. Identity Manager policies allow this user to obtain UNIX attributes
- > 3. Identity Manager policies configure this particular user for a default loginShell and gecos field. The homeDirectory attribute is populated with a NFS or CIFS mountable location. The uidNumber is taken from a preconfigured pool
- > 4. The Fan-Out driver processes this user and sends all the appropriate information to the local platform cache
- > 5. The user logs onto a Fan-Out system, using the Name Service Switch (NSS) and acquires the UNIX attributes set by the Settings driver
- > 6. PAM or automountd mounts the user's homeDirectory
- > 7. The user changes his or her password with passwd and the change is replicated to the Identity Vault using PAM and Fan-Out

# New Enhancements

- Self-extracting installers with native OS packaging for Fan-out platforms
- Complete non-interactive and automated installations
- New platforms: Tru64 and HP-UX Itanium
- Performance Increases for large scale deployments
- Better support for dynamic groups
- Embedded Remote Loader for Fan-out driver
- Platform errors and responses logged centrally for easy viewing

**Novell®**

## **Unpublished Work of Novell, Inc. All Rights Reserved.**

This work is an unpublished work and contains confidential, proprietary, and trade secret information of Novell, Inc. Access to this work is restricted to Novell employees who have a need to know to perform tasks within the scope of their assignments. No part of this work may be practiced, performed, copied, distributed, revised, modified, translated, abridged, condensed, expanded, collected, or adapted without the prior written consent of Novell, Inc. Any use or exploitation of this work without authorization could subject the perpetrator to criminal and civil liability.

## **General Disclaimer**

This document is not to be construed as a promise by any participating company to develop, deliver, or market a product. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. Novell, Inc. makes no representations or warranties with respect to the contents of this document, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. The development, release, and timing of features or functionality described for Novell products remains at the sole discretion of Novell. Further, Novell, Inc. reserves the right to revise this document and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes. All Novell marks referenced in this presentation are trademarks or registered trademarks of Novell, Inc. in the United States and other countries. All third-party trademarks are the property of their respective owners.

