2011年1月27日（木） 19時～21時

統合認証システム構築術

Unix認証OpenLDAP編

オープンソース・ソリューション・テクノロジ株式会社

代表取締役 チーフアーキテクト 小田切耕司

お問い合わせ info@osstech.co.jp

OSSTech

目　次

Part 1
講師紹介
オープンソース・ソリューション・テクノロジ
会社紹介

OSSTech

講師紹介

- 役　職 ： 代表取締役 チーフアーキテクト
- 氏　名 ： 小田切 耕司（おだぎり こうじ）
- 所属団体等
  - OpenSSO&OpenAMコンソーシアム 副会長
  - OSSコンソーシアム 副会長
  - 日本LDAPユーザ会設立発起人
  - 日本Sambaユーザ会初代代表幹事
  - 日本Webminユーザーズ・グループ副会長
  - オープンソースソフトウェア協会
- ブログ　Shall we Samba ？　　http://blog.odagiri.org/
- 執筆関係
  - ASCII.technologies 2011年2月号
    - 『キホンから学ぶLDAP』
    - http://tech.ascii.jp/elem/000/000/569/569412/
  - 技術評論社 Software Design 2010年9月号
    - 第1特集 クラウド対策もこれでOK！
      統合認証システム構築術
      OpenAM／SAML／OpenLDAP／Active Directory
    - http://gihyo.jp/magazine/SD/archive/2010/201009
  - @IT やってはいけないSambaサーバ構築：2008年版
  - 2006年5月 技術評論社 LDAP Super Expert
    - 巻頭企画
    - [新規／移行]LDAPディレクトリサービス導入計画
    - http://www.gihyo.co.jp/magazines/ldap-se

オープンソース・ソリューション・テクノロジ株式会社

- 2006年9月に設立
- **OSに依存しないOSSのソリューションを中心に提供**
  - Linuxだけでなく、SolarisやFreeBSDへも対応！
- **Samba、OpenLDAP, OpenAMなどによる統合認証やシングルサインオン、ID統合ソリューションを提供**
  - 製品パッケージ提供
  - 製品サポート提供
  - 技術コンサルティング提供

http://www.osstech.co.jp

会社概要

会 社 名	オープンソース・ソリューション・テクノロジ株式会社	所属 団体等	OpenSSO&OpenAMコンソーシアム理事 副会長 OSSコンソーシアム理事 副会長 LPI-Japanビジネスパートナー デルISVアリーナ パートナー NEC CLUSTERPRO WORKSパートナー レッドハット レディ・ビジネス・パートナー Solaris Community for Business (SCB)
英語表記	Open Source Solution Technology Corporation		
社名略称	OSSTech(オーエスエステック)または OSSテクノロジ		
業務内容	・OSS(オープンソース)を中心とするソフトウエアの 企画、開発、販売およびサポート ・システムの導入に関するコンサルティング ・ソフトウエアに関する教育、研修	取引先 および パートナー様	・株式会社野村総合研究所 ・デル株式会社 ・株式会社バッファロー ・日本電気株式会社 ・株式会社 大塚商会 ・キヤノンITソリューションズ株式会社 ・伊藤忠テクノソリューションズ株式会社 ・新日鉄ソリューションズ株式会社 ・株式会社PFU ・株式会社 日立ソリューションズ ・三菱電機インフォメーションシステムズ株式会社 ・ソフトバンク・テクノロジー株式会社 ・ニフティ株式会社 ・三井情報株式会社 ・ダイワボウ情報システム株式会社 ・NTTデータ先端技術株式会社
役 員	代表取締役　　小田切 耕司 技術取締役　　武田 保真		
オフィス	〒141-0022 東京都品川区東五反田1-12-10 三井住友海上五反田ビル6F Tel & FAX : 03-6670-5764		
Web	http://www.osstech.co.jp/		
設立	2006年9月		
資本金	1500万円		

OSSTechの製品群(すべてOSSで提供)

原則Linux/Solaris/AIX共にRPMで提供

① Samba for Linux/Solaris/AIX
- ADの代替、高性能NASの代替

② OpenLDAP for Linux/Solaris/AIX
- 認証統合、ディレクトリサービス、シングルサインオンのインフラ

③ OpenAM for Linux/Windows/Solaris
- Tomcat,OpenLDAP対応で高機能なシングルサインオン機能を提供

④ Unicorn ID Manager for Linux/Solaris
- Google Apps,ActiveDirectory,LDAP, Yahoo!メール Academic Editionに対応した統合ID管理

OSSTechの製品群(すべてOSSで提供)

原則Linux/Solaris/AIX共にRPMで提供

⑤ Chimera Search for Linux
- アクセス権の無いファイルは表示されない全文検索システム

⑥ LDAP Account Manager for Linux/Solaris
- 管理機能の弱いOSSのLDAP/SambaにWebベースのGUIを提供

⑦ SSLBridge for Linux
- リモートからのWindowsファイルサーバアクセス機能を提供

⑧ Mailman for Linux/Solaris
- Google Appsのメーリングリスト機能を補完

⑨ Netatalk for Linux/Solaris
- UTF-8に対応したMac OS対応のAFPファイルサーバー

エンジニア募集中です！

特にOpenAM（Java）のエンジニア募集中

http://www.osstech.co.jp/company/recruit

recruit@osstech.co.jp

- OpenAM（OpenSSO）を使ったシングルサインオンもしくはSamba、OpenLDAPを使った統合認証に関する開発エンジニア、コンサルタント、アーキテクト
- シングルサインオン、統合認証、Linux / UNIX / OSS経験
- Java,Cの知識があり、前向きに自分でスキル向上を目指せる方
- 紹介会社などを通さず直接弊社へ募集エントリされた方には、入社後現金20万円を差し上げます

Part 2
統合認証とシングルサインオン

OSSTech

統合認証とシングルサインオンの必要性

- クラウド(外部のWebサービスの業務利用)が普及したことで、統合認証/シングルサインオン/ID管理の必要性がが急上昇
- 社内Webアプリ(オンプレミス)の利便性・セキュリティ向上のための需要も同時に増加中
  - 社内にある多数ありWebアプリ(オンプレミス)へのアクセスをシングルサインオンで管理し、利便性を向上させたい
  - 社内のWebアプリと外部のWebサービス(Google Apps、Salesforceなど)をシングルサインオン連携したい(クラウドサービス利用者)
  - クラウド基盤の構成コンポーネントとして、OpenAMを利用したい(クラウドサービス提供者)

統合認証とシングルサインオンとID管理

- 統合認証とシングルサインオンとID管理は同時に使うことで最大の効果を発揮する
  - ユーザーID/パスワードはシングルサインオンシステムで一元管理可能でも、各アプリケーション・サービス毎に必要なユーザー情報は、基本的には個々に管理される
  - ID管理ツールなどを利用した一元管理をしなければ、ID管理は破綻する
- クラウドサービスにおいても、ID管理は必要
  - クラウドサービスもユーザー情報を保存することから、ID管理の対象となる
  - ID管理用のAPI(プログラムインタフェース)を備えているものが多い(Google Apps、Yahoo! など)

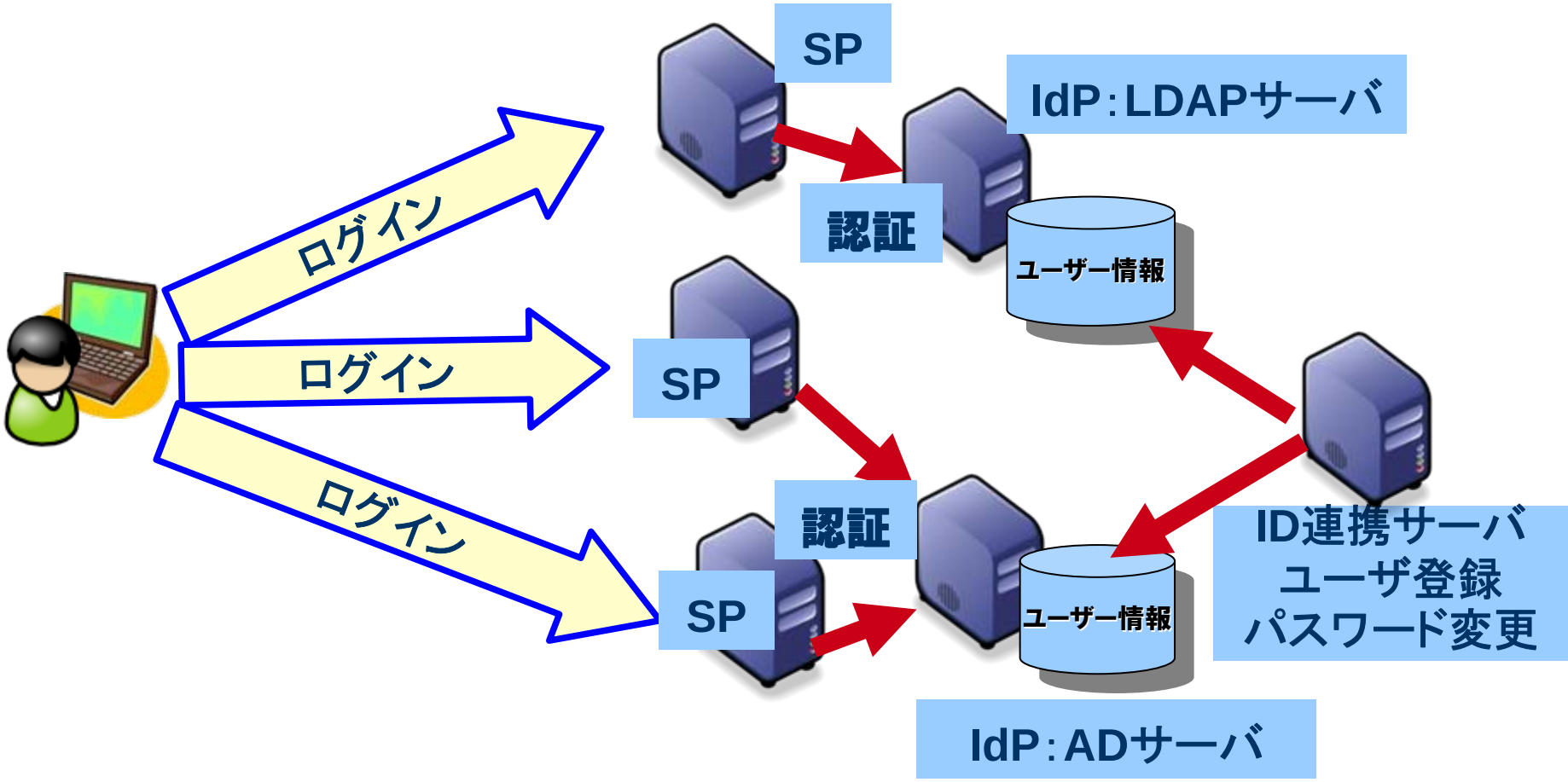
統合認証とは？

- WindowsやLinux/UNIXの認証をできる限りユーザの負担が少ない方法で提供する
- 色々なコンピュータへ同じアカウント名とパスワードでログインできる
- コンピュータのログインだけでなく、その上で動く様々なサービスへも同じパスワードでログインできる
  ※例）
  - メールサーバ（POP,IMAP,SMTPサービス）の認証
  - PPPやVPNなどのリモート接続のためのRadius認証
  - ApacheやIISなどのWebサーバのBasic認証やForm認証

ID連携による統合認証とは？

- 複数のシステムに同じアカウントとパスワードを設定
  - ID管理データベースは別々になっている
- システム毎にやるのは大変
  - 「統合ID管理機能」を持つソフトウェアを導入するのが一般的
- 統合ID管理ソフトは、複数システムのアカウントとパスワードを集中管理
  - 1ヶ所でIDを登録すると複数のシステムへ自動的にIDを一括登録する機能
  - ユーザが1ヶ所でパスワードを変更すると関連するすべてのシステムのパスワードを変更する機能
- 既存システムにできる限り手を加えずに実現できる方式として大変実用的
- パッケージソフト利用やSaaS利用において「ID統合による統合認証」に対応していない場合にも有用な方法

ID連携による統合認証



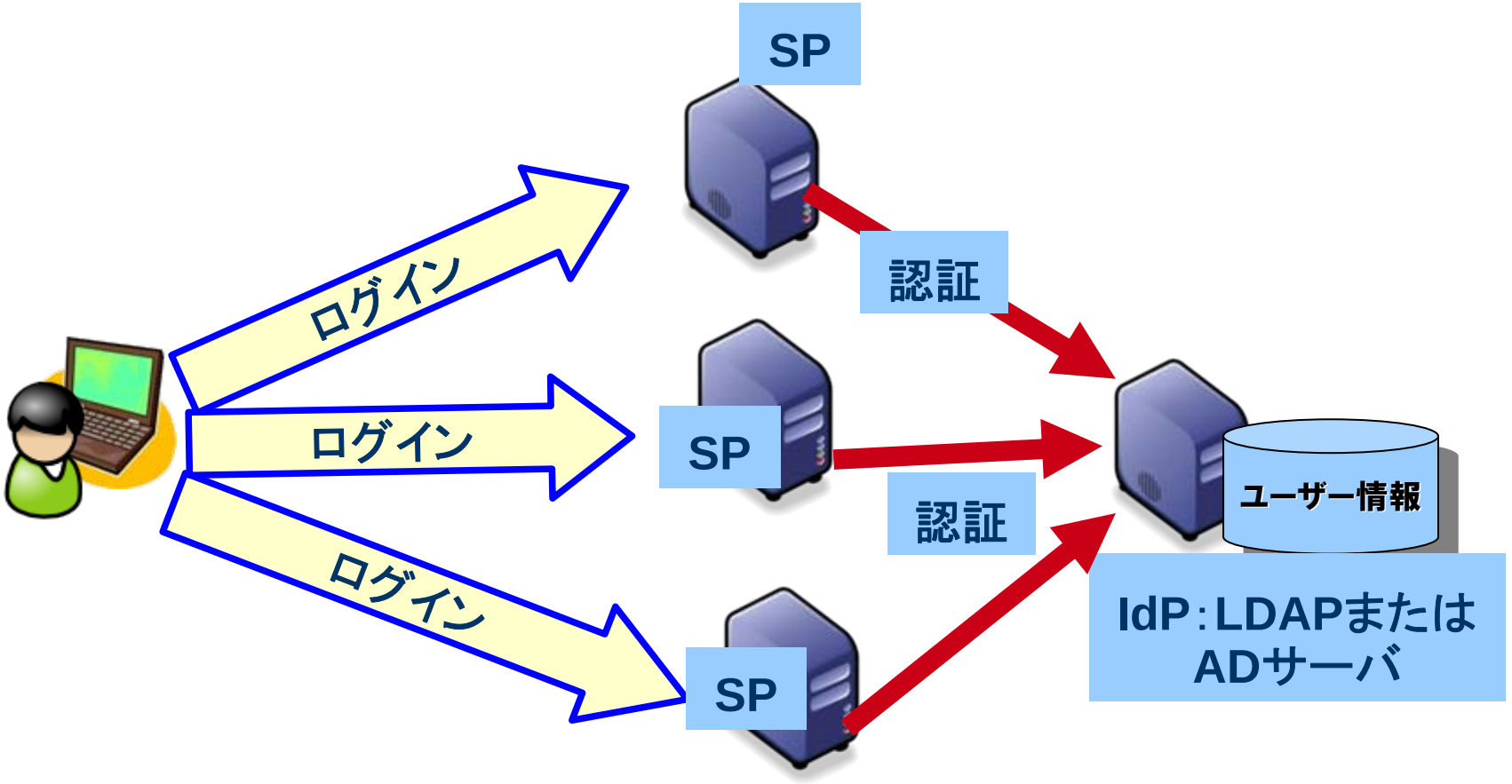
- ユーザ登録をADとLDAPに一括して行う
- パスワード変更も同時に行う

ID統合による統合認証とは？

- 望ましい形はIDをひとつに集約し、これですべての認証を統合してしまうこと
- ID連携による統合認証は、導入費用が既存ソフトを改修する費用よりも安くないとメリットはない
- IDをひとつに統合する方法
  ① UNIX/Linux上のLDAPによる統合認証
  ② Windows ActiveDirectoryによる統合認証
  ③ UNIX/Linux上のActiveDirectoryによる統合認証
  ④ Windows上のLDAPによる統合認証

※RDBによるID統合は不可能ではないが容易ではない

ID統合による統合認証



UNIX/Linux上のLDAPによる統合認証

- サーバやクライアントにUNIX/Linux(Mac OS XもUNIX系)を利用している場合に推奨
- メールサーバやWebサーバ、アプリケーションサーバがUNIX/Linux上で動作している場合も推奨
- UNIX/Linuxが標準でPAM (Pluggable Authentication Module:IDとパスワードで認証するモジュール、ICカード認証や生体認証など他の認証方式もPAMがあれば実現可能)とNSS(Name Service Switch:ユーザやプロセスにUNIXのuid,gidを提供する)をサポートしている
- JavaやRuby, Perl, PHP, Pythonなどのプログラム言語がLDAPのクラスやモジュールを提供している
- PAMはUNIX/LinuxのOSログインから、それの上で動く様々なサーバソフトの認証も制御することのできるモジュールで大変汎用的にできているため、PAMに対応しているvsftpやsshd, postfix, dovecotなどは簡単にLDAP認証に切り替えることが可能
- ApacheやProFTPdやFreeRADIUSなどPAM経由だけでなく、直接LDAPのAPIを利用することでLDAP認証を実現しているものもある
- すべてのOSやプログラムが認証データベースもしくは認証プロトコルとしてLDAPのそれを利用することで同じIDとパスワードでOSやアプリにログインできることになる

Windows ActiveDirectoryによる統合認証

- Windowsでは標準機能で実現可能
  - クライアントをADドメインに参加
- UNIX/Linuxの場合でもPAMを使うことで利用可能

① LDAPのPAMを使う方法

② KerberosのPAMを使う方法

③ SambaのWinbind機能のPAMを使う方法

LDAPのPAMを使う方法

- 認証後プロセスが動作するためのuid,gidが必要になるため、これをNSSから利用できるようにするにはAD側にSUA(Subsystem for UNIX-based Applications)などを使ってUNIX用の拡張スキーマを入れる必要がある
  - ADを変更する必要があるので敬遠されがち

KerberosのPAMを使う方法

- Kerberosサーバで認証後にチケットもらってサーバにアクセスする方法
- セキュリティが強固になるが、uid,gidが必要
- Kerberosサーバは提供してくれないため、AD側にUNIX用の拡張スキーマを入れ、NSSだけNISを使う
- Kerberosのチケットを使うことができるので、一度ログインすれば同じPAMを使うサービスに再度パスワードを入力せずに利用できるSSOも実現可能

SambaのWinbind機能のPAMを使う方法

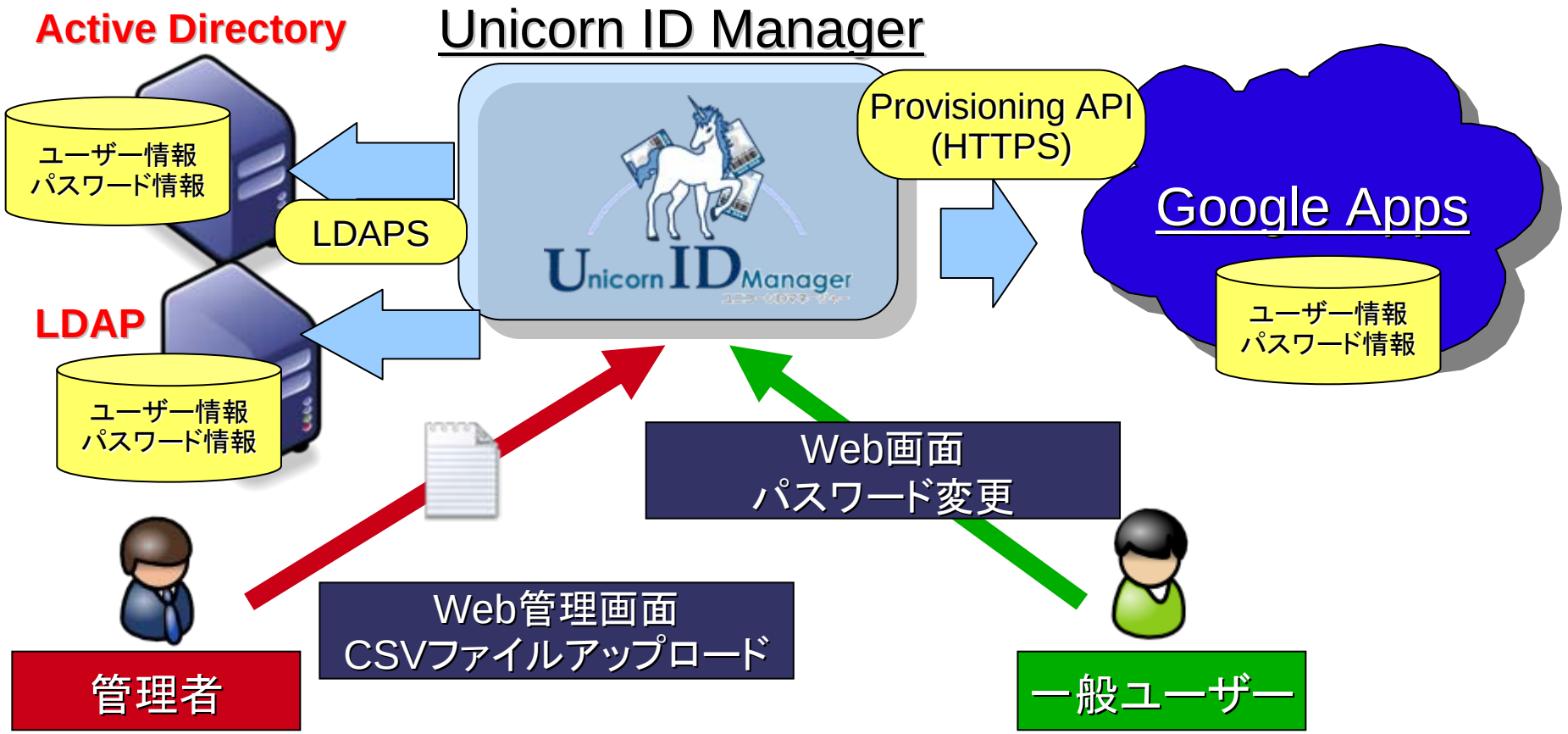
- Kerberosのチケット方式で認証し、udi,gidをWindowsのSID(セキュリティ識別子)から自動生成することが可能
- SUAのインストールは不要
(SUAを入れてADのUNIX拡張スキーマでuid,gidを提供することも可能)
- PAMもNSSもSambaが提供するwinbindモジュールを使うことで実現可能
- 認証でKerberosのチケットを使うことができるので、一度ログインすれば同じPAMを使うサービスに再度パスワードを入力せずに利用できるSSOも実現可能

それぞれの長所と短所

- LDAPの制限
  SambaをつかえばWindowsクライアント統合認証も可能だが
  Samba3系ではADのグループポリシーをサポートしていない
- ADの制限
  - ADで統合認証するにはWindowsのCALが必要
    ユーザ数が増えるとコストがかさむ
  - Windowsサーバの信頼性がUNIXサーバより劣る場合が多く、
    UNIXサーバの認証をADに任せるとWindowsサーバの障害が
    UNIXサーバの障害へつながってしまう
- ADとLDAPの両方を導入し、Windowsクライアントには**AD**を使った統合認証を提供し、UNIX/Linux/Mac OSクライアントには**LDAP**を使った統合認証を提供し、ADとLDAPの間をID連携による統合認証を行うという方式を取るユーザが多い

Unicorn ID ManagerによるAD,LDAP連携

- Webブラウザから CSVを投入するだけで統合ID管理



UNIX/Linux上のADによる統合認証

- ADとLDAP両方の欠点を補う方法
- UNIX / Linux上でADを動作させ統合認証する方法
- ADとほぼ同機能を持ったSamba4が現在開発中
- LDAPによる統合認証が実現できる
  （今年2011年こそリリースして欲しい）
- UNIX/Linuxクライアントは LDAPクライアントになる
- WindowsクライアントはSamba4を通してLDAPの中に格納されたIDでAD(Kerberos)認証が可能

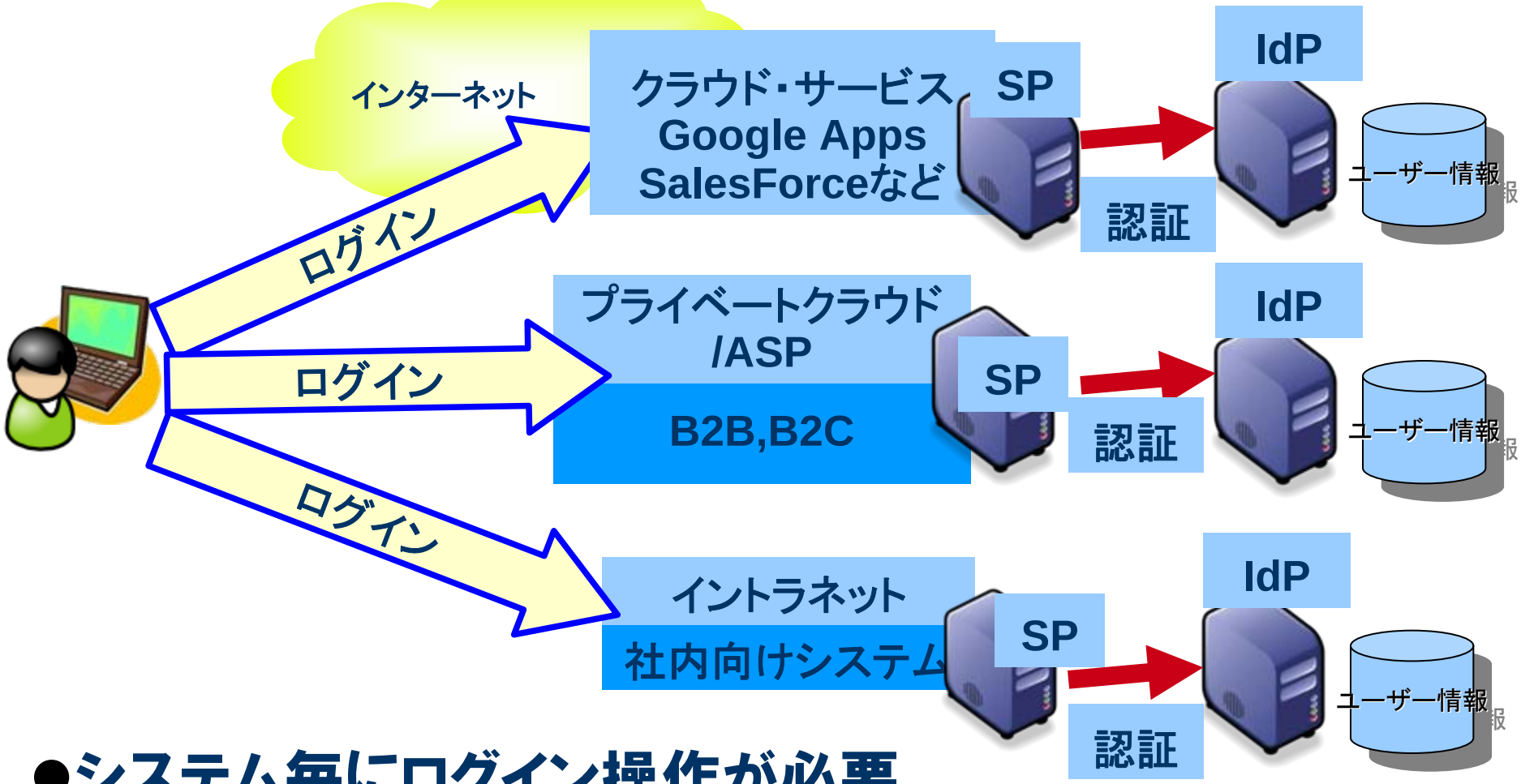
Windows上のLDAPによる統合認証

- ADはLDAPの機能を包含しており、Windowsの上でADではないLDAPを動かす意義はあまり無い
- しかし、OpenLDAPなどのUNIX系LDAPの代わりにADを使うのは品質、性能、柔軟性の綿で不安要素が多い
- OpenLDAP for WindowsというOSSの製品もある
- Javaの上で動くOpenDS/OpenDJやOracle Directory Serverなど商用のLDAP製品も多数ある
- Windowsサーバの上でOpenLDAPを利用する場合もCALを購入する必要がある
- 商用製品の場合は該当製品のユーザライセンスに加え、WindowsのCALの２重の費用がかかる
- Windows XP,Vista,7クライアントの上にサーバソフトをインストールしてサーバ用途に利用することは費用がかかる以前にライセンス違反となる
（Windowsクライアント製品の場合、サーバプロセスへの同時接続数に制限がかけられている）

シングルサインオンとは

- １回のパスワード入力で複数のシステムやサービスを同時利用
- 「ID統合を使った統合認証」ではIDとパスワードの管理を1カ所でできるためユーザの追加も楽、社員が退社した場合に1カ所IDを削除すれば、すべてのシステムが利用不可となる
- 近年クラウドサービス（SaaS, PaaS, IaaS, HaaSなど）の普及により、（社外にある）サービス毎にID／パスワードを登録しなければならないケースが増えており、「ID連携による統合認証」を使わざるを得ないケースが増えている
- ところがこのID連携が費用の問題や技術的な問題で完全に実現されていない場合、例えば社員が退社した時に社内システムのIDを削除しても、SaaS側のIDが残っているとクラウド側のシステムは社外から使えてしまう、といった問題が起きてしまう

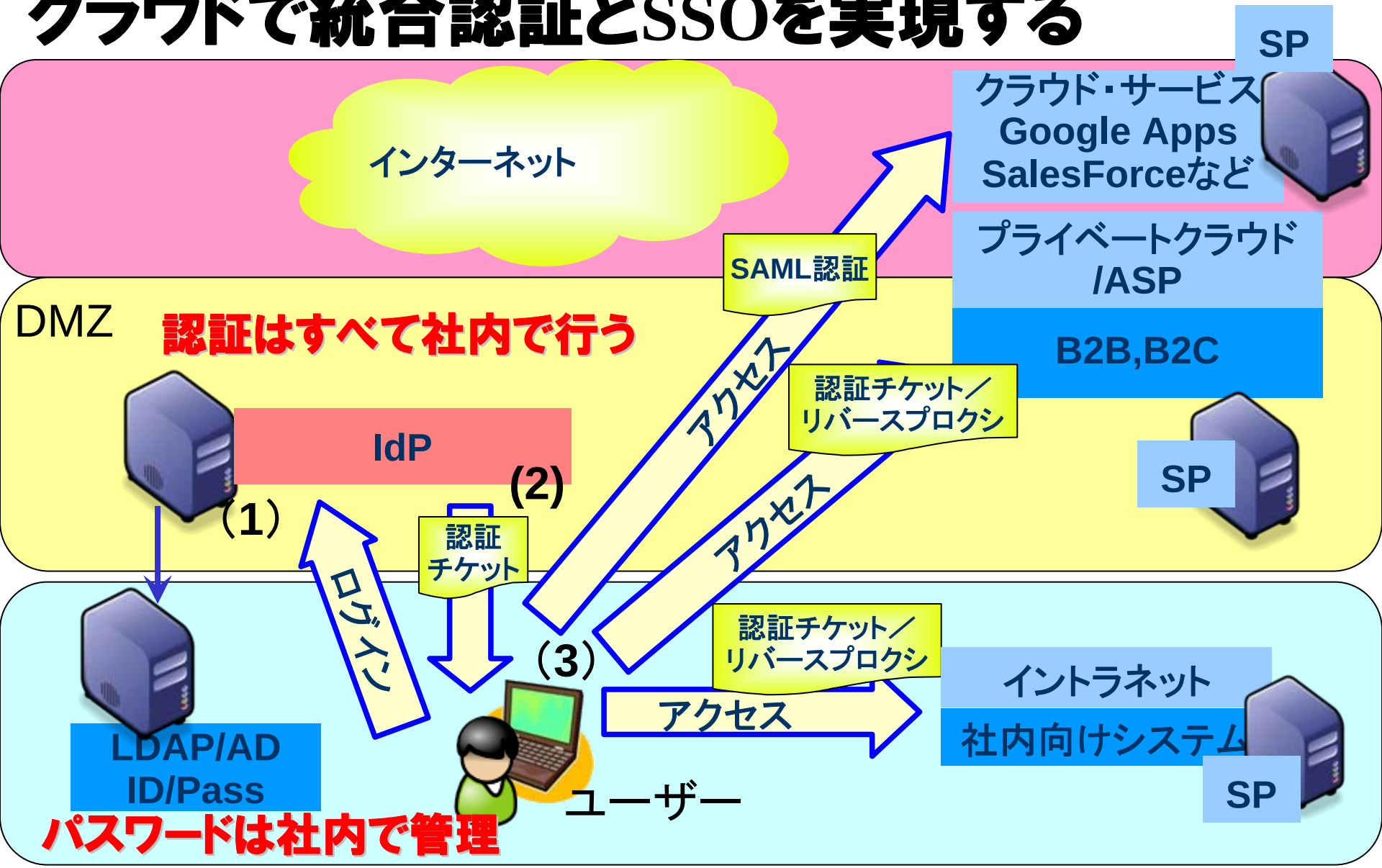
クラウドで統合認証ができていないと．．．



- システム毎にログイン操作が必要
- クラウドにID/パスワードとパスワードを置く必要がある

（パスワードを社外に置くと不正ログインされる危険性が高い）

クラウドで統合認証とSSOを実現する



Part 3
LDAP概念と設計入門

OSSTech

LDAPとは?

- ディレクトリサービスを利用するための規約の1つ(RFCで定義)
  - ディレクトリサービスとは、キーを基に関連情報を取り出す仕組み
  - ユーザ管理、電話帳、リソース管理などに利用
  - 高機能だが運用負荷や開発コストが高かったITU- T 勧告のX.500 ディレクトリ・サービスを「90 %の機能を10 %のコストで実現する」ために設計
- 商用LDAP製品も多数存在
  - Sun Java Directory Server, Red Hat Directory Server, Novell eDirectoryなど
  - MS Active DirectoryもLDAP準拠(認証はKerberos)
- オープンソースソフト
  - OpenLDAP
    - Linux ディストリビューションに同梱されるオープンソースのLDAP
  - Red Hat / Fedora Directory Server
    - かつてのNetscape Directory ServerをOSSにしたもの(RHは有償、Fedoraは無償)
  - Apache Directory Server
    - Apacheプロジェクトが進めるJavaで書かれたDS

LDAPのプロトコルスタック（X.500との違い）

X.500でのDAPとLDAPのプロトコルスタックの違い

LDAPにおいてX.500から削除されたもの

ROSE (Remote Operation Service Element)	遠隔操作サービス要素、処理の依頼と結果の通知という通信メカニズムを実現するプロトコル要素
RTSE (Reliable Transfer Service Element)	高信頼転送サービス要素、通信経路障害などによって情報の欠落や重複が起きないようにするプロトコル要素
ACSE (Association Control Service Element)	アソシエーション制御サービス要素、コネクションの確立、正常開放、異常解放を行なうサービス要素

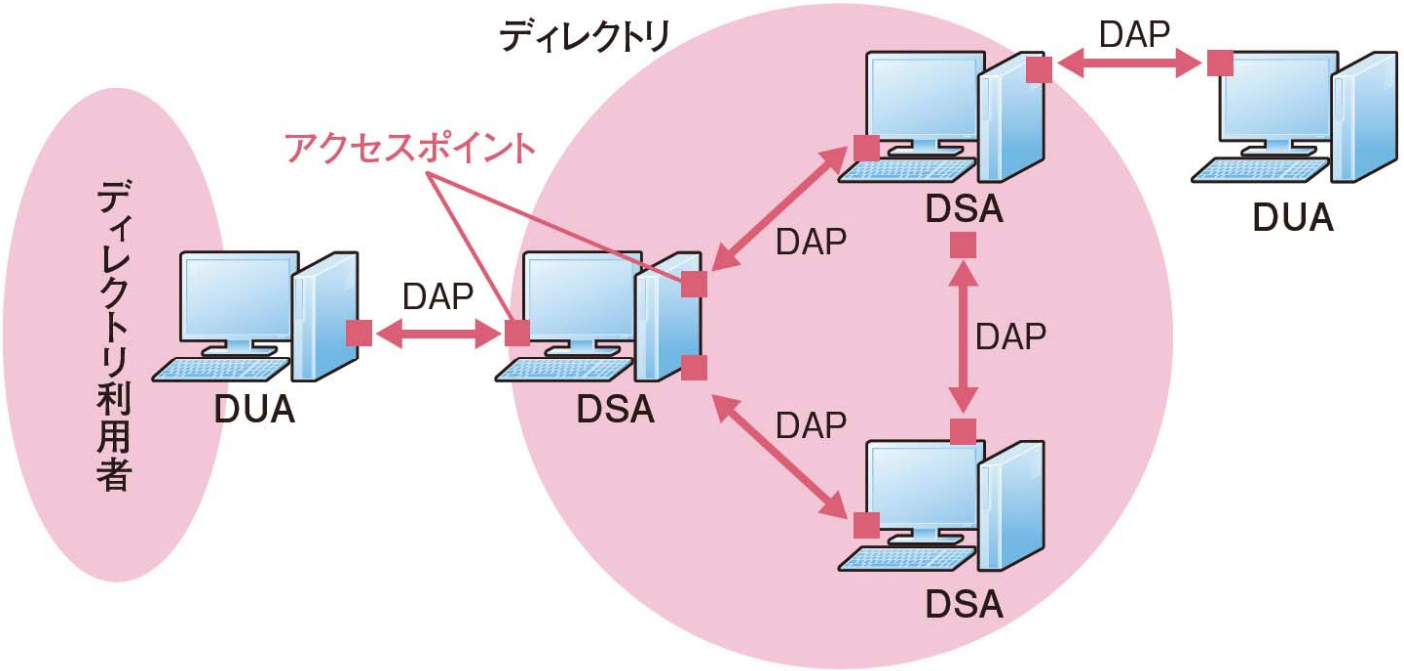
TCP/IP上で動作するためにDAPから不要になった機能

DSP (Directory System Protocol)	DSA間で分散協調動作(連鎖や紹介)を行なうためのプロトコル
DOP (Directory Operational binding management Protocol)	ディレクトリ運用結合管理プロトコル。DSA間の運用結合の規定内容や状態の交換に用いられるプロトコル
DISP (Directory Information Shadowing Protocol)	DSA間で複製情報を交換するためのプロトコル

DAP以外のX.500の機能

ディレクトリの機能モデル

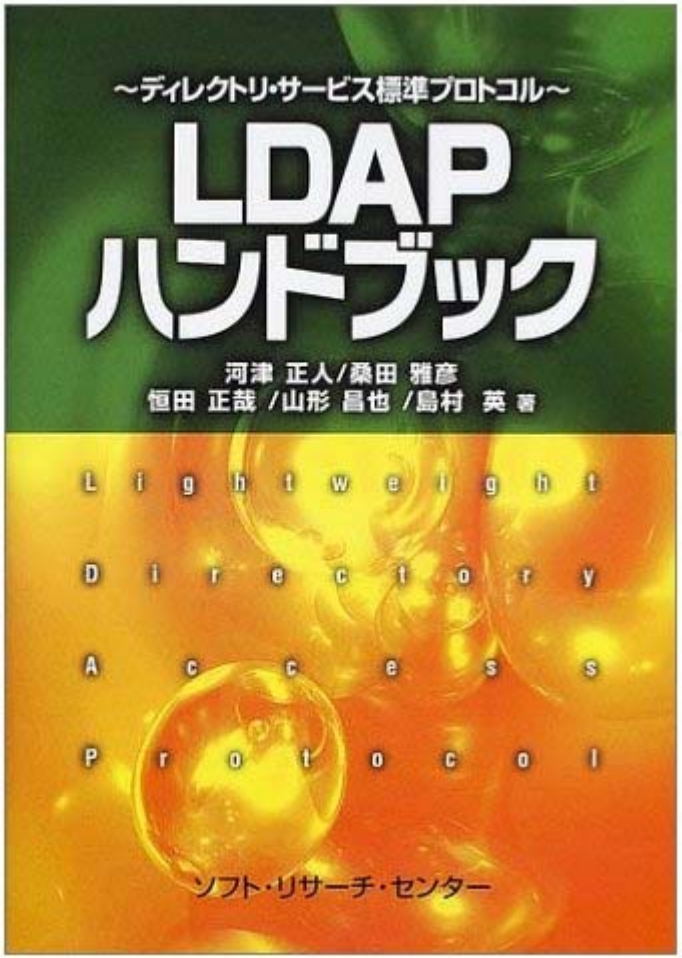
- DSA (Directory Service Agent) :ディレクトリ情報を管理する個々のシステム。ディレクトリはDSAの集合体として構成される。
- DUA (Directory User Agent) :ディレクトリの利用者に代わってディレクトリへアクセスする機能(プログラムやコマンド、ライブラリ)
- LDAPでは単純にLDAPサーバーとLDAPクライアントと呼ぶことが多い



ディレクトリの機能モデル

LDAP概念を勉強のための参考書

- LDAPハンドブック
  ディレクトリ・サービス標準プロトコル
  - 出版社: ソフトリサーチセンター (2002/03)
  - 発売日: 2002/03



LDAPとRDBMSの違い

- LDAPはネットワークプロトコル、SQLは言語

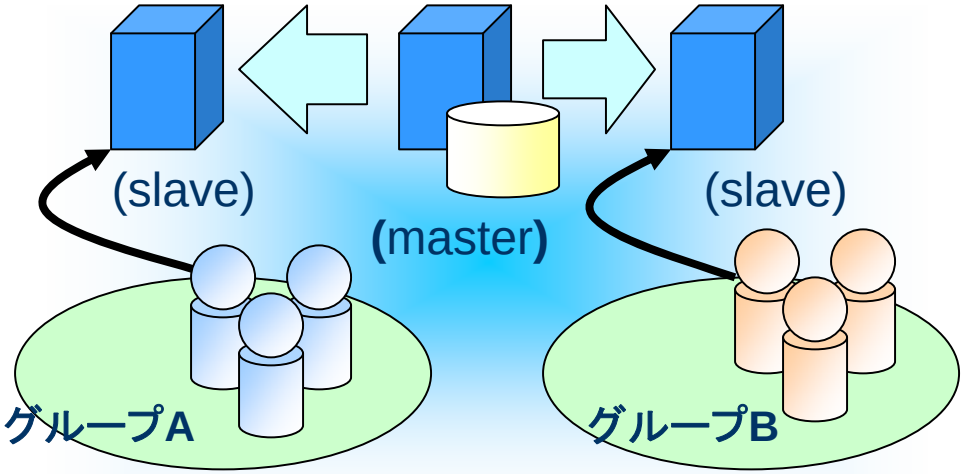
	LDAP	RDBMS
用途	検索性能重視、頻繁な更新には向かない	検索だけでなく頻繁な更新も重視
構造	木構造（行や列といった概念はない）	表構造（行や列が存在）
スキーマ	既存の登録済みスキーマ(ObjectClass)を利用するのが一般的	ユーザが業務に合わせて個別に設計し、利用する
更新	トランザクションの概念はない (トランザクション機能を持った製品もある) 大量更新には向かないので1時間に数件といった更新頻度のものに利用する	トランザクションの概念あり 1秒間に何十、何百もの更新に耐えられる設計となっている
分散	ツリーの枝単位で分散配置が可能	キーの範囲で分散配置が可能
操作	LDAP（ネットワークプロトコル）で操作 プロトコルは単純	SQL（プログラム言語）で操作 複雑な操作が可能
検索手法	木の枝葉をたどるイメージ	表の行を走査するイメージ

LDAP概念に関する勘違い

- RDBMSは永続的なユーザ情報を蓄えるために使う、LDAPは管理情報を集約するために使う（社員DBはRDMS、全社認証システムはLDAP）
- LDAPは検索重視となっているが、RDBより必ずしも早いわけではない
- LDAPはスケールアウト型負荷分散がやりやすいから
- 更新がすぐに反映されるとは限らない
  - ユーザ追加やパスワード変更がすぐにされないことがある(だからWindowsはパスワードをキャッシュする）
- マルチマスターの利用は要注意
  - トランザクションやロックの概念が弱い
  - uid,gidの自動割り振りをLDAPでやると危険

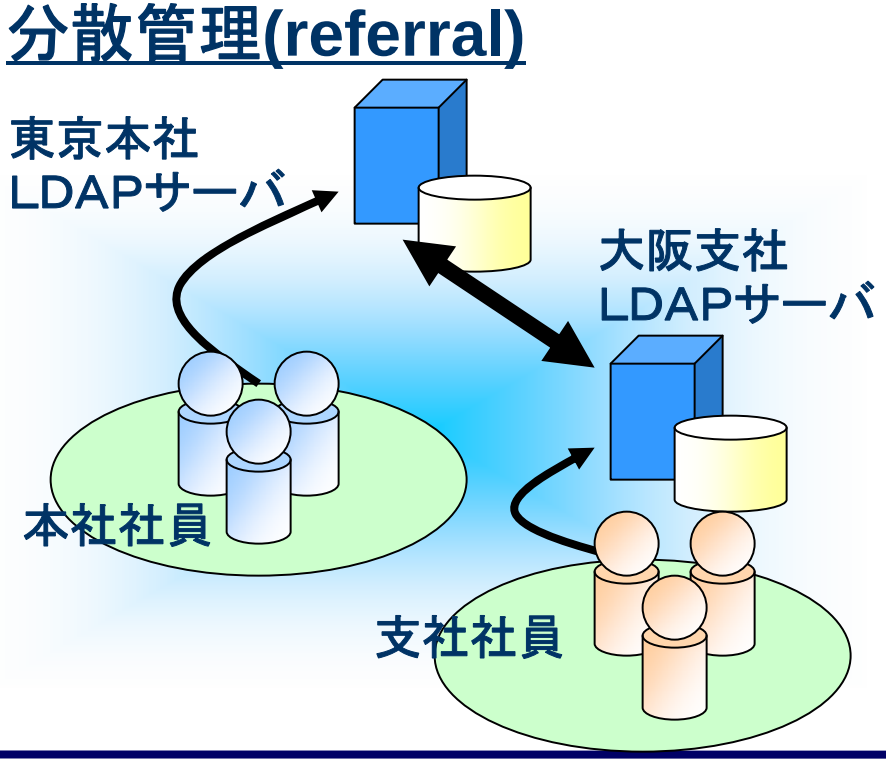
負荷分散方法1:レプリケーション

- 同じ内容のサーバを複数用意する
  - サーバを増やすだけでスケールアウトする
  - 負荷分散装置やldap.confで負荷を分散
  - 1つのサーバが持つデータ量は同じなので規模が大きくなると更新性能が低下
  - Syncreplではサブツリーだけを複製することも可能



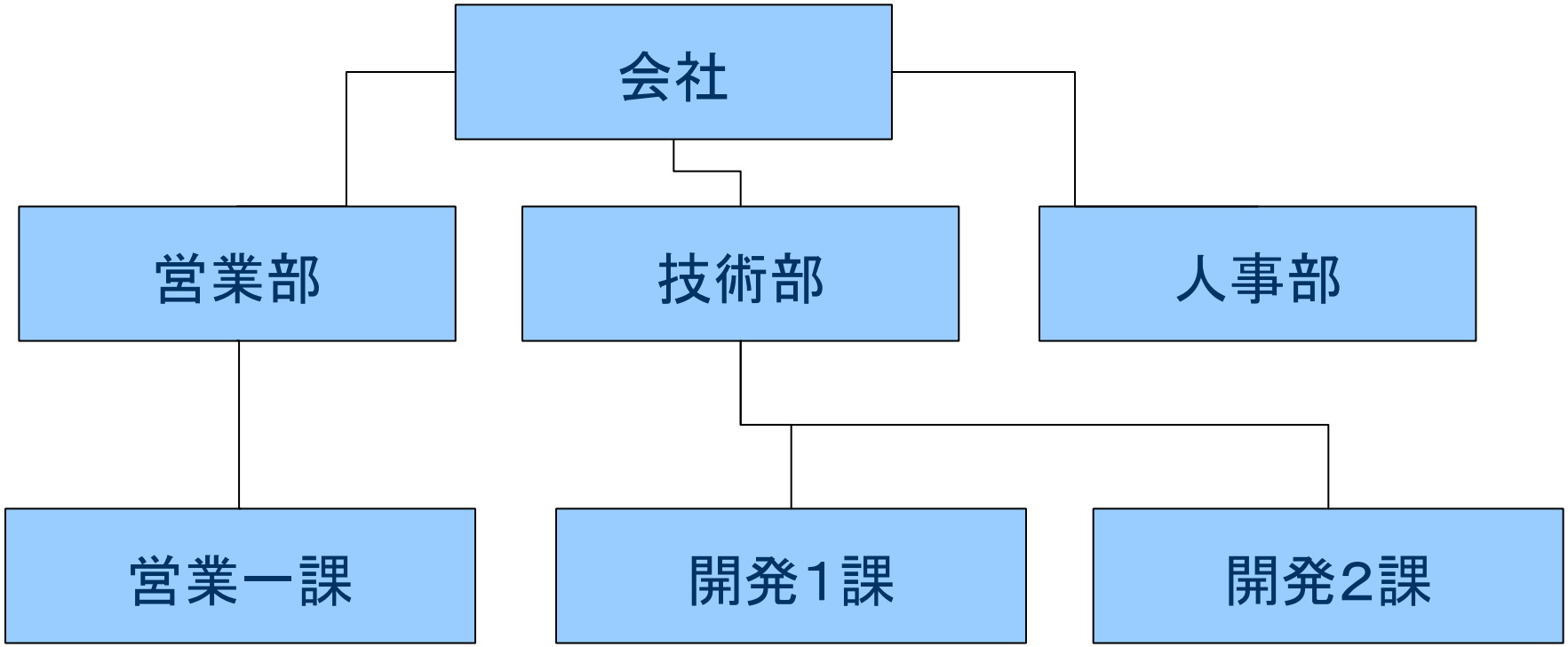
負荷分散方法2:リファラル

- サブツリー単位でサーバを分散する
  - ldap.confでbaseツリーを変える(負荷分散というよりも管理分散)
  - 1サーバがもつデータ量が減るので更新性能も上がる
  - referralが返ったら別なサーバを見に行くのはプログラム側の責任



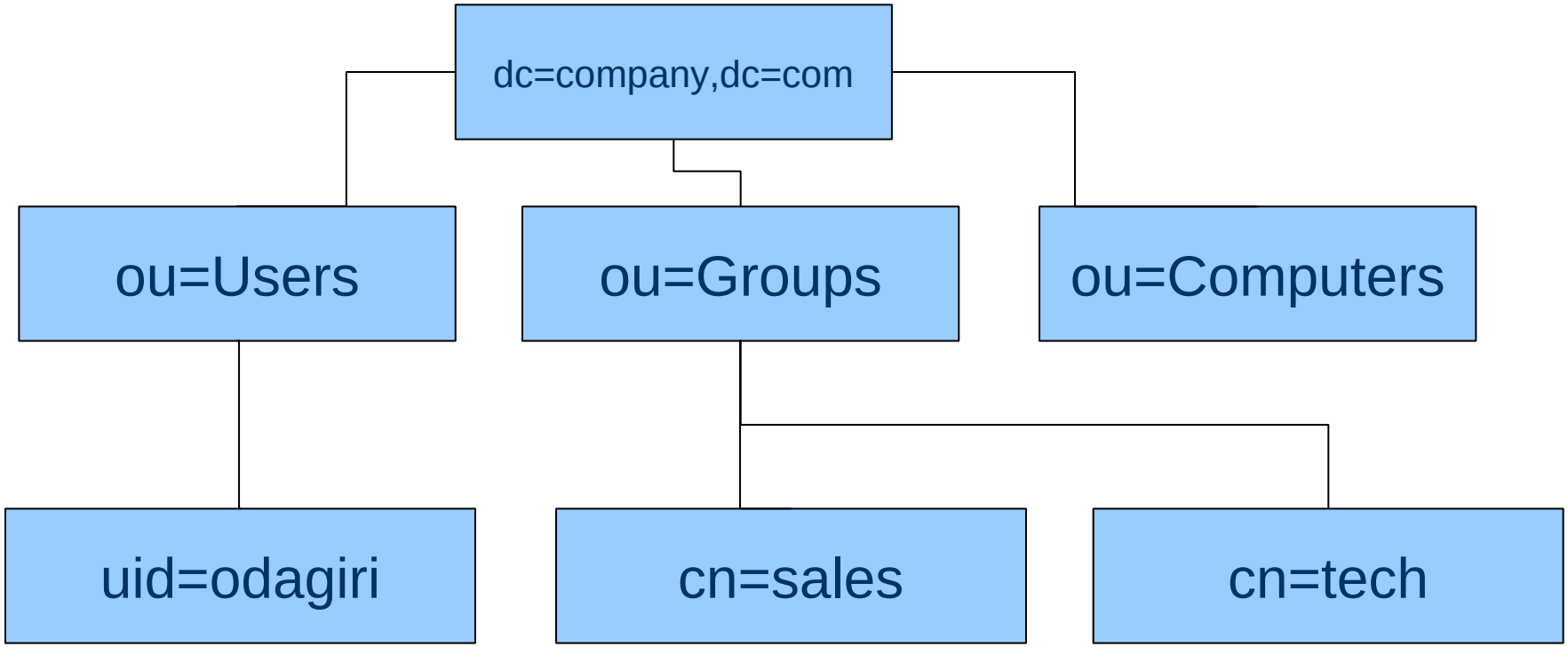
DIT (Directory information Tree) の概念

- 概念として組織構造をあげる書籍が多いが. . .



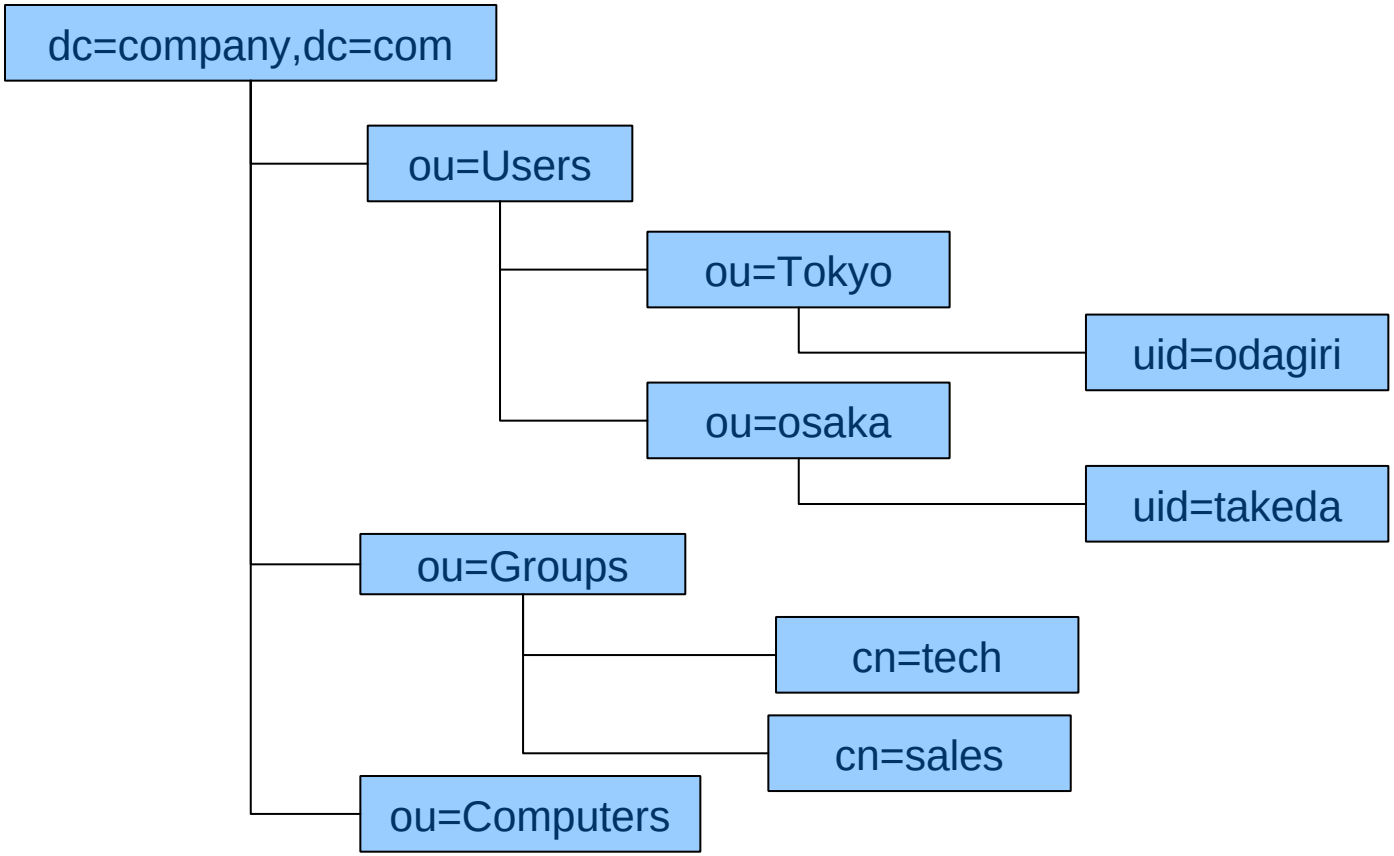
DIT (Directory information Tree) の概念

- 実構造としては管理単位で分ける



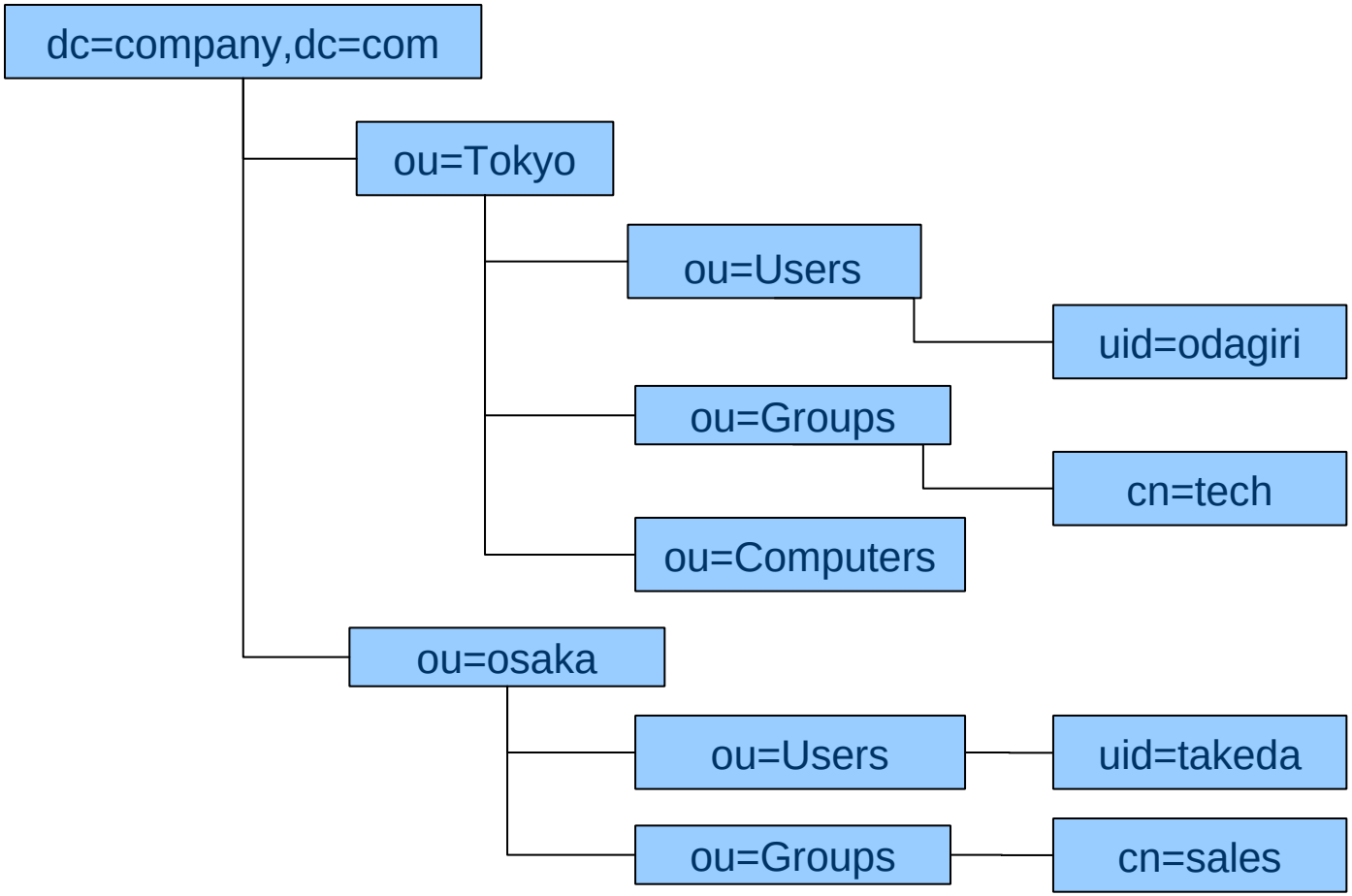
DIT (Directory information Tree) の設計 (1)

- 組織構造にマッピングしないこと、管理対象で分ける



DIT (Directory information Tree) の設計 (2)

- 組織構造にマッピングしないこと、管理対象で分ける



LDAPで何ができるか？

- Linuxユーザの統合管理
  （Mail,FTP,Telnet,Proxy,sshなど）
- Samba/Windowsユーザの統合管理
- Webサーバ（Apache）のアクセス制御
- 電話帳、メールアドレス帳
- PKI（公開キー）の保管場所として

- LDAPのスキーマはむやみに拡張しない
  本当に必要か精査する

OpenLDAPが標準で提供するスキーマ(1)

- 標準提供のスキーマを見ればLDAP何ができるかわかる
- core.schema
  - LDAPの核となるスキーマ、以下のRFCで定義されたスキーマが定義されている。
    - RFC 2252/2256 (LDAPv3)
    - RFC 1274 (uid/dc)
    - RFC 2079 (URI)
    - RFC 2247 (dc/dcObject)
    - RFC 2587 (PKI)
    - RFC 2589 (Dynamic Directory Services)
    - RFC 2377 (uidObject)
  - これだけでは何もできないが、CNやOUなど他のスキーマを使うための基本部分が定義されている。
- cosine.schema
  - X.500やX.400で規定されたアトリビュートなど以下のようなものが定義されている。
    - RFC1274で定義されるhost,manager, documentIdentifierなど
    - DNSレコードであるAレコード、MXレコード、NXレコード、SOAレコード、CNAMEレコード
  - これらからDNSレコードの格納先としてLDAPサービスが利用できることがわかる。

OpenLDAPが標準で提供するスキーマ(2)

- inetorgperson.schema
  - インターネット、特にメールアドレス帳のためのスキーマで、以下のようなものが定義される。
    - メールアドレス、社員番号、オフィスと自宅住所、会社と自宅の電話番号、写真、
- misc.schema
  - mailLocalAddressやnisMailAliasなどメールサーバが使うスキーマが定義される。
- nis.schema
  - posixAccountやposixGroupなどLinux/UNIXのユーザ認証統合に必須なスキーマが定義される。
  - NISをLDAPに置き換えるのに必要なスキーマも定義されている。
- samba.schema
  - このスキーマはOpenLDAPではなく、Sambaパッケージによって提供されるが、Sambaを使ってWindows/Linux/UNIXのユーザ認証統合に必須なスキーマが定義される。
  - WindowsドメインをSambaに置き換えるのに必要なスキーマも定義されている。
- java.schema
  - javaClassName, javaCodebaseなどJava Object（RFC 2713）を扱うためのスキーマが定義される。
- corba.schema
  - corbaIor、corbaRepositoryIdなどCorba Object（RFC 2714）を扱うためのスキーマが定義される。

アドレス帳の設定例

dn: uid=**ユーザ名**,ou=Users,dc=**ドメイン名**,dc=co,dc=jp
objectClass: posixAccount
objectClass: inetOrgPerson
cn: **ユーザ名**
sn: **名字**
givenname: **名前**
mail: **メールアドレス**
o: **会社名**
ou: **所属**
title: **役職**
employeeNumber: **社員番号**
telephoneNumber: **電話番号**
facsimileTelephoneNumber: FAX**番号**
mobile: **携帯電話**
st: **都道府県**
l: **市区**
street: **番地**
postalAddress: **番地**
postOfficeBox: **ビル名**
postalCode: **郵便番号**
homePostalAddress: **自宅住所**
homePhone: **自宅電話**

```
dn: uid=odagiri, ou=Users, dc=osstech,dc=co,dc=jp
objectClass: posixAccount
objectClass: inetOrgPerson
cn: odagiri
sn: 小田切
givenname: 耕司
mail: odagiri@osstech.co.jp
o: オープンソース・ソリューション・テクノロジ株式会社
ou: 技術部
title: チーフアーキテクト
employeeNumber: 1
telephoneNumber: 03-1234-5678
facsimileTelephoneNumber: 03-8765-4321
mobile: 090-5432-1234
st: 東京都
l: 品川区東五反田
street: 1-21-10
postalAddress: 1-21-10
postOfficeBox: 三井住友五反田ビル
postalCode: 107-0052
homePostalAddress: 神奈川県藤沢市藤沢123-45
homePhone: 0466-23-4567
```

Part 4
LDAP構築／設定入門

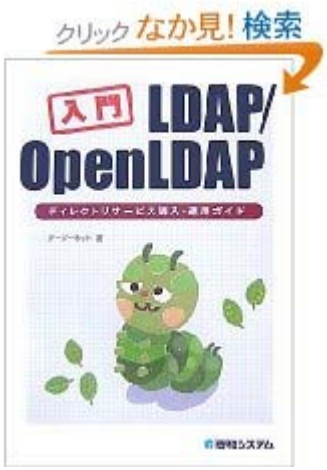
OSSTech

ソースからコンパイルしてインストール

- LPICの試験ではコンパイルの仕方も出る。勉強方法としては、configure；make によるインストールをやっておくこと。
- configureのオプションも確認しておくこと
- OpenLDAPをコンパイルするのに必要なライブラリ
  - BDB(今はLDBM、GDBMはほとんど使われないが、SQLを始めどんなバックエンドDBが使えるか知っておくこと)
  - OpenSSL(TLSライブラリとして使われる)
    - 通信の暗号化
  - Cyrus SASL
    - 安全な認証方式
  - Kerberos (MITかHeimdal)
    - 安全な認証
    - Kerberos認証のためのスキーマもLDAPに格納

OpenLDAP勉強のための参考書

- OpenLDAP入門
  - オープンソースではじめるディレクトリサービス
  - 出版社: 技術評論社
  - 発売日: 2003/07
- 入門LDAP/OpenLDAP
  - ディレクトリサービス導入・運用ガイド
  - 出版社: 秀和システム
  - 発売日: 2007/10

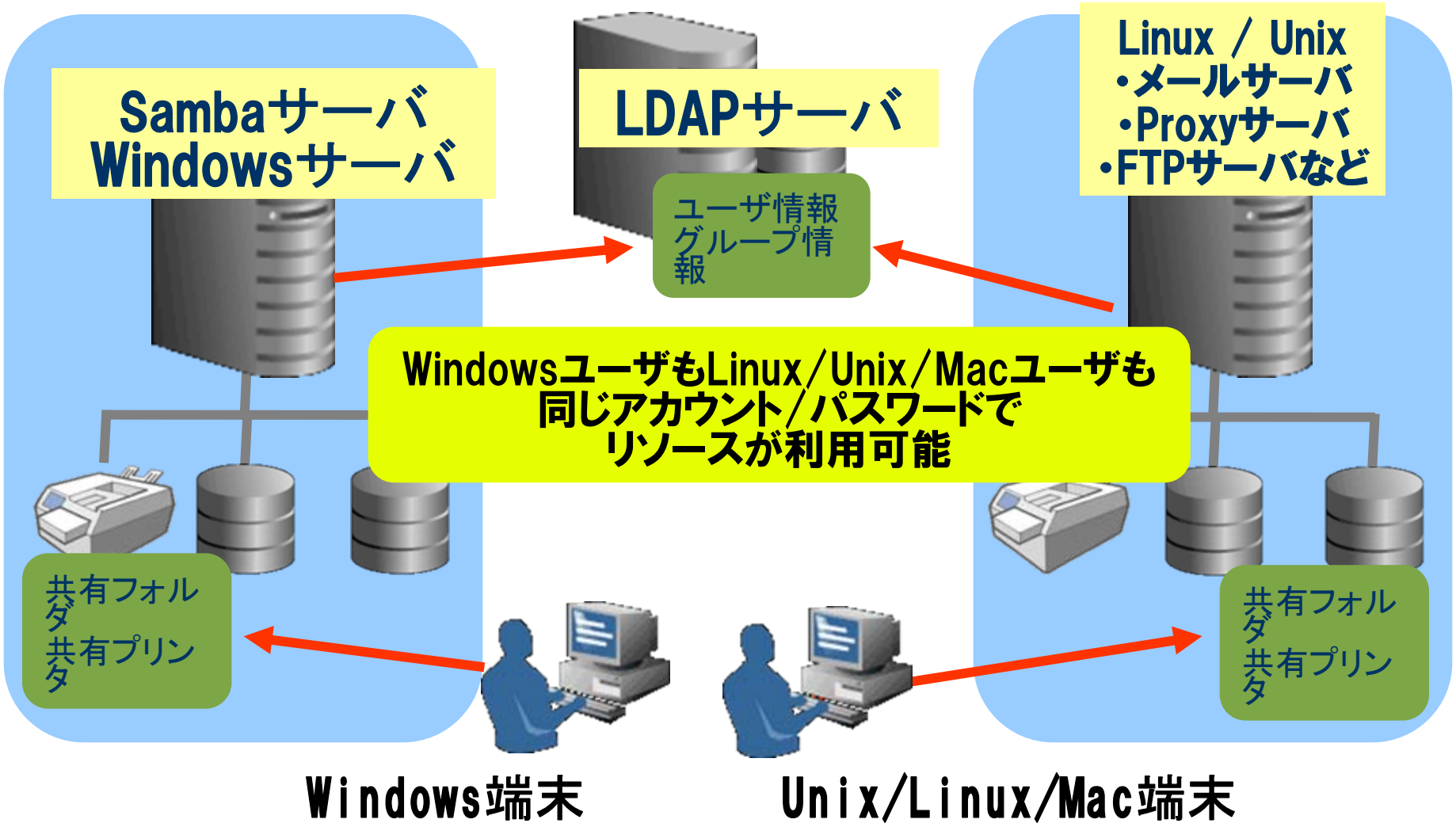


標準インストール：実システムでの注意

- OpenLDAPはどんどん新しくなるので、書籍の情報では古いことがある。
  - www.openldap.org のドキュメントを読むしかない
- 実際の業務システムでは、configure ; make でインストールしないこと。
- 業務システムではRPMやDEB、PKGなどOS標準のパッケージ管理システムを使うこと
- コンパイルするのに必要なライブラリは、OS標準のものを使うのが一般的だがBDBだけはOpenLDAP専用のものを使った方が良い。
  - Red Hat のRPMはBDBだけはOS標準を使わないようにSPECファイルが書かれているので、これを参考にすると良い。

✓ 上記理由からRed HatではOpenLDAPのBDBリカバリに db_recover は使わない ！ slapd_db_recover を使う

LDAPによる認証統合

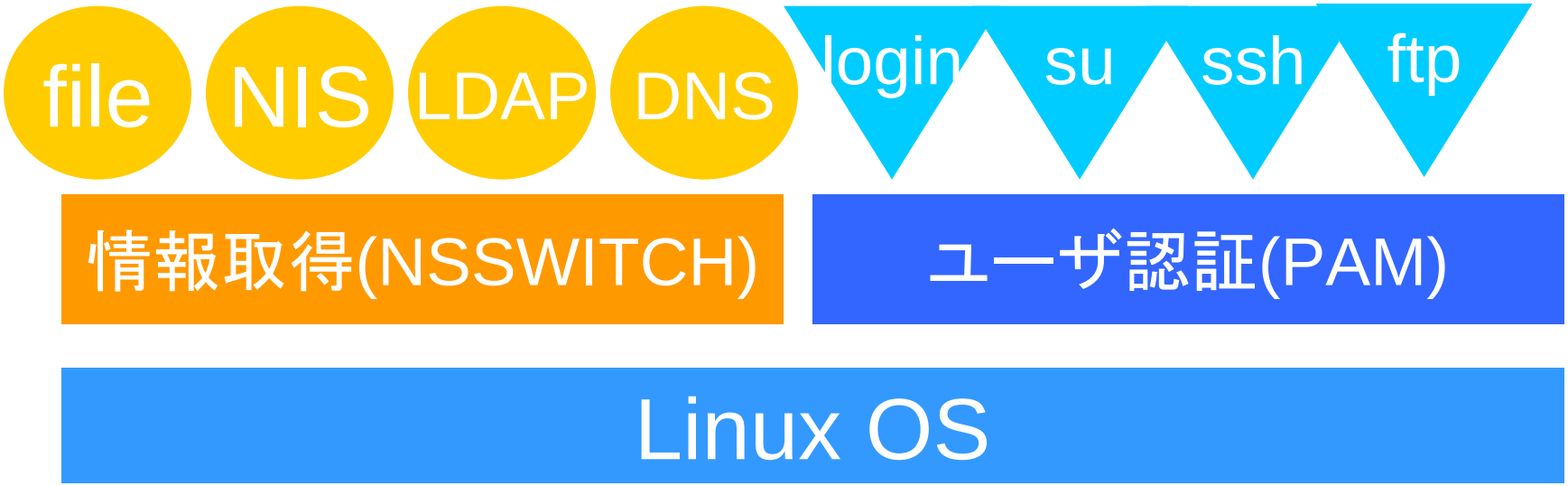


LDAPの設定

- LDAPサーバとしての設定
  - slapd.confの設定
- LDAPクライアントとしての設定
  - NSS設定
  - PAM設定
  - ldap.conf設定

•LDAPクライアントとしての設定

- NSS(ネーム・サービス・スイッチ)機能
  - システムのユーザ名、グループ名、ホスト名の解決方法を設定
  - /etc/nsswitch.confで、各種情報の取得先を指定可能
- PAM認証機構
  - アプリケーション毎の認証方法を設定
  - /etc/pam.d/の中でアプリケーションごとの認証ルールを指定可能



ネームサービススイッチ機能

- LDAPを認証で使用するには/etc/nsswitch.confを以下のように変更

```
passwd:   files   ldap
group:    files   ldap
shadow:   files   ldap
hosts:    files   dns   wins
```

- /lib/libnss_ldap.so.2が呼ばれる。
- /lib/libnss_wins.so.2 を使うとWINS(Windows Internet Name Service)を使って名前解決可能

プラグマブル認証機能

- /etc/pam.d/system-authに以下を設定

```
[root@fs02 /etc]# cat /etc/pam.d/system-auth
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth        required      /lib/security/pam_env.so
auth        sufficient    /lib/security/pam_unix.so likeauth nullok
auth        sufficient    /lib/security/pam_ldap.so use_first_pass
auth        required      /lib/security/pam_deny.so

account     required      /lib/security/pam_unix.so
account     [default=ok user_unknown=ignore service_err=ignore system_err=ignore] /lib/security/pam_ldap.so

password    required      /lib/security/pam_cracklib.so retry=3
password    sufficient    /lib/security/pam_unix.so nullok use_authtok md5 shadow
password    sufficient    /lib/security/pam_ldap.so use_authtok
password    required      /lib/security/pam_deny.so

session     required      /lib/security/pam_limits.so
session     required      /lib/security/pam_unix.so
session     optional      /lib/security/pam_ldap.so
session     required      /lib/security/pam_mkhomedir.so skel=/etc/skel umask=0022
```

- /etc/pam.d/sshdなどに以下を設定

```
#%PAM-1.0
auth        required     /lib/security/pam_stack.so    service=system-auth
account     required     /lib/security/pam_stack.so    service=system-auth
password    required     /lib/security/pam_stack.so    service=system-auth
session     required     /lib/security/pam_stack.so    service=system-auth
```

NSSとPAMをGUIで簡単設定（RHEL,CentOSの場合）

authconfigで設定

OpenLDAPサーバの設定

設定ファイル

サーバ：/etc/openldap/slapd.conf

クライアント：

NSS,PAM用：/etc/ldap.conf

ldapaddなどの管理コマンド用:/etc/openldap/ldap.conf

OpenLDAP 管理者ガイド

http://www.ldap.jp/doc

http://www5f.biglobe.ne.jp/~inachi/openldap/

Red Hat Enterprise Linux 4 リファレンスガイド

http://www.redhat.com/docs/manuals/enterprise/RHEL-4-Manual/ja/pdf/rhel-rg-ja.pdf

/etc/openldap/slapd.confﾊﾟﾗﾒｰﾀ(1)

- suffix **ベース・サフィックスを指定する**
  通常はﾄﾞﾒｲﾝ名をﾍﾞｰｽに指定
  例）suffix　dc=osstech,dc=co,dc=jp
  suffix ″ou=sales,ou=yokohama,o=company,c=jp″

CN=commonName
L=localityName
ST=stateOrProvinceName
O=organizationName
OU=organizationalUnitName
C=countryName
STREET=streetAddress
DC=domainComponent
UID=userid

/etc/openldap/slapd.confﾊﾟﾗﾒｰﾀ(2)

- rootdn
  LDAPサーバの管理者のDN（Distinguished Name：識別名）を指定する。
  なお管理者DNを含むユーザDNには、英大文字、英子文字の区別はない。
  管理者DNの例）
  - rootdn "cn=Manager,dc=osstech,dc=co,dc=jp"
- rootpw
  LDAPサーバの管理者パスワードを設定する。
  - そのままのパスワードを指定するか暗号化したものを設定する
  - 例）secret1234というパスワードをSSHAハッシュする
    # slappasswd -s secret1234 -h {SSHA}
  - rootdnをLDAPに登録されているユーザを指定し、LDAPの中にパスワードが格納されていれば、rootpwを指定する必要はない。

/etc/openldap/slapd.confパラメータ(3)

- include
  - 与えたファイルから追加の設定情報を読み込む。
  - 通常はスキーマ定義ファイルを読み込むために使用する
    例） include /etc/openldap/schema/samba.schema
- database
  - LDAPのデータを格納するのに使用するバックエンド・データベースを指定。
- directory
  - databaseファイルを格納するディレクトリを指定
  - 例）directory /var/lib/ldap
- index
  - 作成する索引の属性とタイプを指定する。
    - 例1） uid,gidに関してequal（等値）検索用の索引を作成
      index uidNumber,gidNumber eq
    - 例2） mail（メールアドレス）、surname（名字）に関して、equal検索用とsubinitial（前方一致）の索引を作成
      index mail,surname eq,subinitial

/etc/openldap/slapd.confパラメータ(4)

- Slapd.confの例：サフィックスを"dc=osstech,dc=co,dc=jp"、管理者DNを"cn=Manager,dc=osstech,dc=co,dc=jp"、管理者パスワードをsecret1234

```
include /etc/openldap/schema/core.schema
include /etc/openldap/schema/cosine.schema
include /etc/openldap/schema/inetorgperson.schema
include /etc/openldap/schema/nis.schema

database  bdb
directory /var/lib/ldap
suffix "dc=osstech,dc=co,dc=jp“
rootdn "cn=Manager,dc=osstech,dc=co,dc=jp“
rootpw secret1234
index  objectClass,uidNumber,gidNumber,memberUid  eq
index  cn,mail,surname,givenname                  eq,subinitial
index  uid                                        pres,eq
index  rid                                        eq
```

- 設定が終了したら、OpenLDAPデーモンを起動させる。
  # service ldap restart　　※Red Hat系
- システム起動時に自動的に動くように以下を設定
  # chkconfig ldap on　　※Red Hat系

LPIC301 LDAP 例題解説

http://www.lpi.or.jp/skillcheck/301/index.php

各自自宅で挑戦してみてください

Part 5

やってはいけないOpenLDAPサーバ構築

OSSTech

Webの情報を鵜呑みにしないこと！

- LDAP(OpenLDAPやRedHatDS,ApacheDS)に関する情報はとても少ない。特に日本語は少ない
- 本当に正しい(推奨)設定に関する情報が少ない
- OpenLDAPの品質は近年急速に良くなった
- ディストリビューションに含まれるOpenLDAPのバージョンに注意が必要
- 心配なら有償サポートやLDAPユーザ会メーリングリストなどに聞きましょう

やってはいけないOpenLDAPサーバ構築

- バージョンの古いOpenLDAPは使うな！
- replog (slurpd) は使うな！
- 複数LDAPを同時更新してはいけない！
- TLSを使おう（SSLじゃあないんだよ）

バージョンの古いOpenLDAPは使うな！

- OpenLDAP 2.3以前はサポート終了
- OpenLDAP 2.3.40以前は複製が抜ける、BDBアクセスでデッドロックなどのバグあり

	OpenLDAP 2.0	OpenLDAP 2.1	OpenLDAP 2.2	OpenLDAP 2.3	OpenLDAP 2.4
初期リリース	2000年8月	2002年6月	2003年12月	2005年6月	2007年10月
最終リリース	2002年9月	2004年4月	2005年11月	2008年7月	2010年7月
最新版	2.0.27	2.1.30	2.2.30	2.3.43	2.4.23
サポートの有無	× 終了	× 終了	× 終了	× 終了	○ サポート中
採用Linux	RHEL3 (2.0.27) 2002/9		RHEL4 (2.2.13) 2004/6	RHEL5 (2.3.43) 2010/2	RHEL6 (2.4.19) 2009/10
推奨複製方式	replog	replog	replog	syncrepl	syncrepl Mirror mode

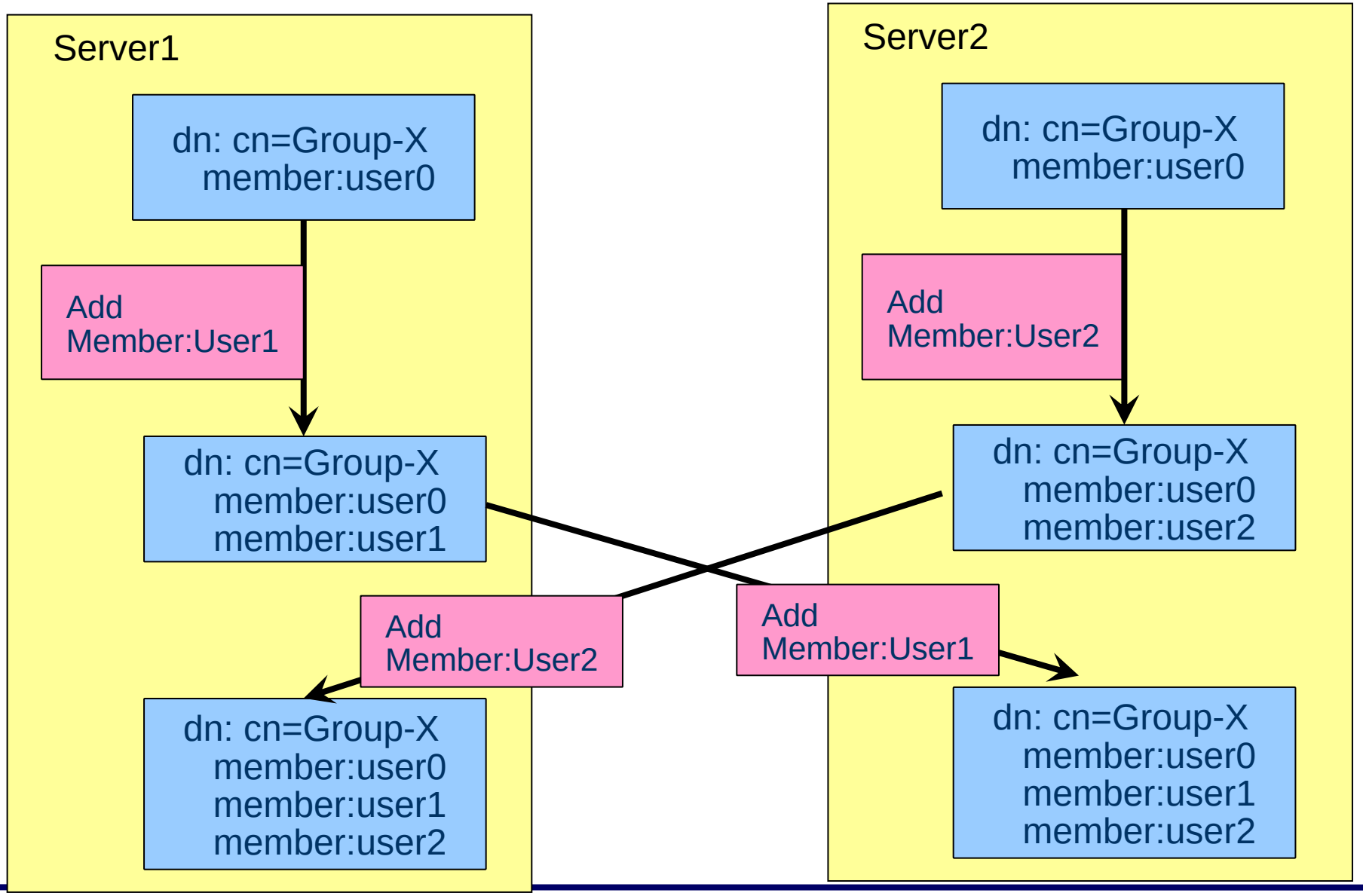
replog (slurpd) は使うな！

- replogは運用が大変
  - エラーリカバリは手操作
  - スレーブの追加時にマスターを止める必要あり
  - スレーブ故障後の修復でもマスターを止める必要あり
  - スレーブ台数が多いと性能劣化
- syncreplは運用が楽
  - エラーリカバリは自動
  - スレーブの追加時にマスターを止める必要なし
  - スレーブ故障後の修復でもマスターを止める必要なし
    データを空にして再起動すれば自動修復
  - syncreplはOpenLDAP 2.2.30 / 2.3.41以降が安全

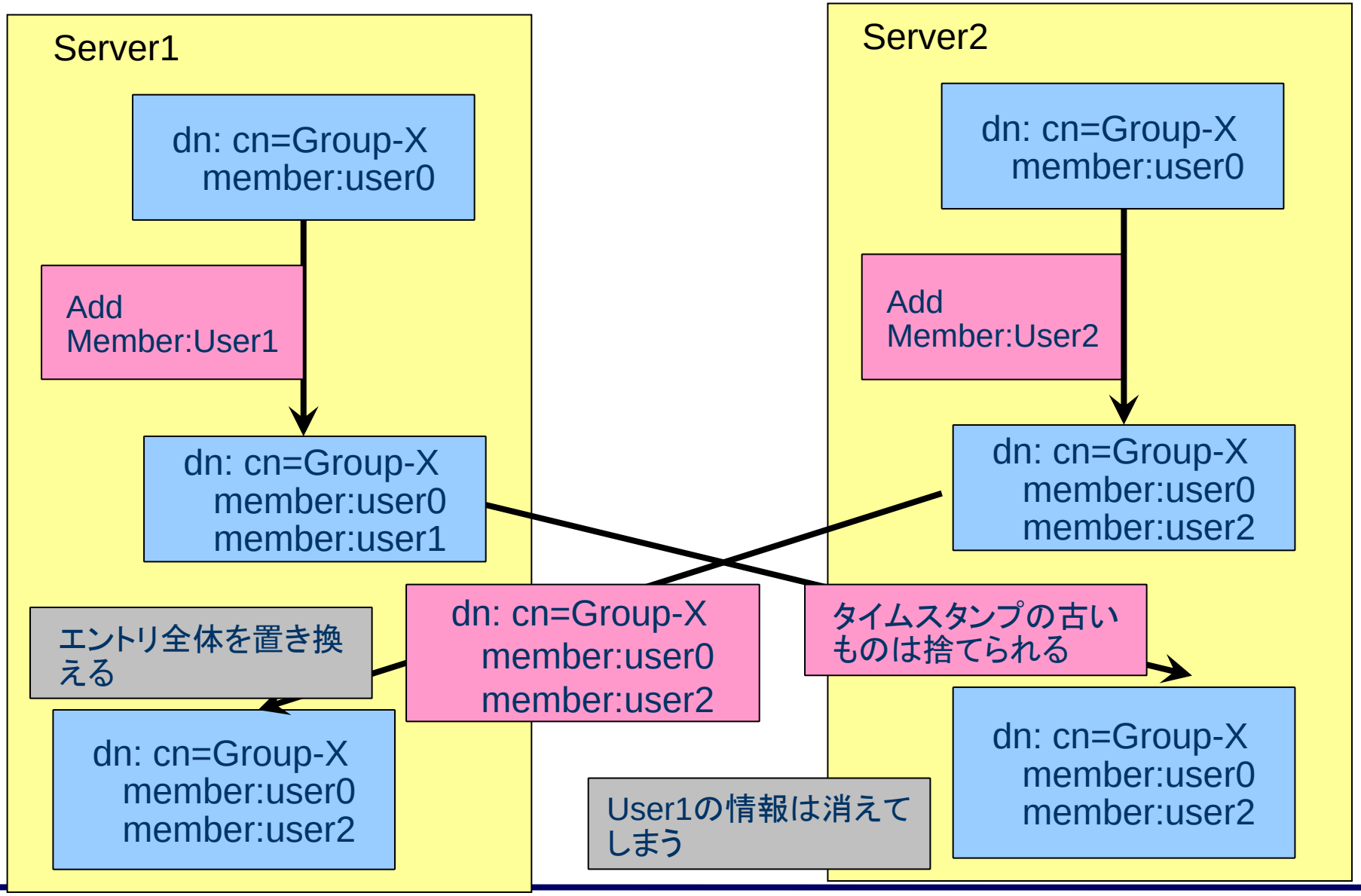
複数LDAPを同時更新してはいけない！

- OpenLDAP 2.4よりマルチマスター(ミラーモードに対応)
- マルチマスター構成は書き込み可能なLDAPサーバーを複数設置する機能
- 1台のLDAPサーバーが故障しても、ほかのサーバーに切り替えができればサービスに影響がない
- データの整合性はデータベースのようなロックする機能を使わずタイムスタンプを使って管理しているので、連続の書き込みが異なるLDAPサーバーに分散された場合は、データの不整合が発生する可能性がある。
- 基本的に書き込み操作を1台のLDAPに集中するデザインが必須である。
- 例えば、ユーザのuid,gid自動割り振りをLDAPのカウントを使ってやるのは危険である。

操作ログによるマルチマスターの動き



エントリ複製方式によるマルチマスターの動き



TLSを使おう(SSLじゃあないんだよ)

- Mac OS XをLDAPクライアント(LDAP認証)にするにはOpenLDAPでTLSかSASLの設定が必要
- 暗号なしのSimple認証はMac OS Xでは受け付けない
- セキュリティ強化のためにはTLSを使った方が良い
- OpenLDAPはSSLではなく、TLSをサポート
  - 正確にはSSLとTLSは違う
  - OpenLDAPはOpenSSLで実装されており、OpenSSLはSSLとTLSの両方をサポートしているのでOpenLDAPはSSLと思われているが正確にはTLSを使う

実は知らないと困るBDBコマンドとパラメータ

- 現在OpenLDAPの推奨バックエンドはBDBなので、BDBのチューニングやコマンドを知ることも重要
- slapd.conf
  - checkpoint　<更新量> <間隔>
  - cache size　<エントリ数>
- DB_CONFIG
  - cachesize
  - DB_LOG_AUTOREMOVE
  - lg_max
- db_recover（slapd_db_recover）コマンド
  予期しないアプリケーション、データベース、またはシステムの障害が発生した後、データベースを整合性のある状態に復元します。
- db_verify（slapd_db_verify）コマンド
  ファイルおよびファイル内に含まれるデータベースの構造を検証します。
- db_archive（slapd_db_archive）
  不要になったログファイルを表示したり、削除する

まとめ＞実システム構築での注意点

- DITは複雑にしない、組織にマッピングしない、管理者にあわせる
- 自分でconfigure,makeはしない
- OS標準のBDBは使わない
- 複製の仕組みの理解
- セキュリティに対する考慮
  - TLS通信やSASL GSSAPI機構による認証
  - rootdnと複製dnを分ける
  - プログラム毎に使用するdnを分ける
  - 細かなアクセス制御