中規模私立大学における総合認証基盤の構築と大学間共有eラーニングシステムへの発展

京都産業大学
情 報 セ ン タ ー　　尾崎　孝治
コンピュータ理工学部　秋山　豊和

京都産業大学

- ◆開学1965年
- ◆１拠点８学部21学科
- ◆学生数　約13,000名
- ◆情報センター（事務組織）が基盤構築担当
- ◆オープンソースの積極的な利用
  - ➢情報処理教室のWindows/Linuxデュアルブート
    1999年603台，現在約2,000台
  - ➢moodleの全学利用 2005年～
  - ➢postfix, apache, openLDAP, postgreSQL, etc.

財団法人大学コンソーシアム京都

- ◆京都には大学がたくさん
- ◆地域・産業界を含めて協力体制の強化
- ◆単位互換制度
  - ➢京都では1994年から単位互換事業が開始
  - ➢他と大きく違う点は，比較的大学間が近いので直接キャンパスまで行って受講できること
  - ➢2008年度には46大学・短期大学から10テーマ506科目が提供。年間1万人が受講。
  - ➢単位互換制度申請システムも以前から稼動

大学間共有eラーニングシステム

◆本学が代表校の戦略的大学連携支援事業

－eラーニングシステムの共有共用化に伴う共用教育の大学間連携と効率化の促進－

遠隔講義やビデオオンデマンドなどのキーワードも含むが，今回の事例に関連するのは…

◆「共用できるeラーニングサーバの設置」

- 単位互換申請システムとLMSの2システムの連携
  - ①単位互換申請システム（スクラッチ開発）
  - ②LMS(moodle)
- 単位互換申請システムにログインすると，受講している科目が表示され，科目を選択するとmoodleに自動ログイン
- moodleのログイン画面や他の科目画面は見せないようカスタマイズ
- 大学コンソーシアム京都に設置

単位互換申請システムは大学と独立した認証

◆大学から学外システムに個人情報を出せない
- 学生本人に個人情報を提供（入力）させることで解決

◆学生はいわゆる会員登録をして利用する
- 学生証番号やメールアドレスを自分で入力
- パスワードも自分で設定

◆システムから印刷する申請書を大学窓口へ提出する流れとし，窓口で本人確認

◆一応利用できる程度に完成した

様々な要求・課題

- ◆大学のパスワードと連携したい
  - ➢とはいえ，学外システムにパスワードは渡せない
- ◆申請者が入力した学生証番号に信頼性が必要
  - ➢繁忙期の窓口で本人確認をする余裕がない
  - ➢学生証番号が信頼できるなら紙を提出させる手順をなくせる
- ◆申請者を装った，第三者による虚偽申請の可能性をなくしたい
  - ➢騙られた学生は初期化されるまで自分の申請ができない

学認(Shibboleth)なら…

- 大学とパスワード連携可能
- 学生証番号は確実に本人のものを大学システムから渡せる
- 会員登録不要なので，第三者の偽登録の心配がない
- IdP（接続する側）のセキュリティメリット
  - SPにはユーザのパスワードが渡らない
  - IdPの持つ情報が無条件にSPへ渡される訳ではない。IdPはSP毎に何の情報を渡すか定義できる
  - 学内システムのSSO化にも有効。ベンダー作成のシステムもShibboleth対応を条件とすれば，パスワードを渡す必要もなく，必要とされる情報も各SPごとに最小限に絞って提供できる
- SP（接続される側）のセキュリティメリット
  - SPになると無制限に接続される訳ではない。設定次第で，好きに制御できる。例えば特定のIdPからしか接続を許可しないよう定義できる
  - SPにユーザパスワードを持つ必要がない

更に学認なら…

GakuNin

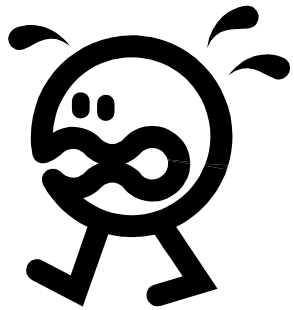
◆複数の大学で共有できる（参加校に推奨できる）手法である

- NIIが主導する学術認証基盤（学内に説明しやすい）
- 使わないユーザの個人情報は一切渡らない仕組み
- パスワードを学外に渡さず学内パスワードと連携できる

◆参加できる大学から順次始めることができる

- 従来の認証方式と併用できる
- 新規に参加する大学が学認IdPさえ用意すればOK
- SP（単位互換申請システム）側で特別な作業は不要
- NII側作業によりメタデータ（信頼するリスト）が更新され，これが自動的にSPに追加される（NIIにお任せ）

リスク



- ◆学認プロジェクトが終了する
  - 利用が多ければ継続されるはず
- ◆学認DSが(トラブルにより)停止する
  - 学内システムに影響が無いように設定可能
- ◆Shibbolethの開発が終了する
  - そこまで責任持てませんが…しばらくは大丈夫でしょう
- ◆信用できないIdPの参加
  - 公開SPが考えるべきリスク。当面は参加大学の数が少ないのでホワイトリスト方式でも運用可能
- ◆信用できないSPへのアクセスによる情報漏えい
  - IdPがデフォルトで属性を提供せず，SPごとに渡す属性をホワイトリストで設定していれば大丈夫

個人情報の取り扱いについて

- ◆本人が了承していれば問題ない
- ◆相手（大学コンソーシアム京都のシステム）は単位互換を申請するシステムであり，かつ信頼できるため，学生証番号，大学メールアドレスを渡すことは説明できる
- ◆アクセスする際に説明画面を表示すればよい
  （uApproveという仕組みもあります。参考まで）
- ◆説明画面を見ない人（＝利用しない人）の情報は一切渡らないので，そのユーザ層は考慮不要

既存システムとの融合

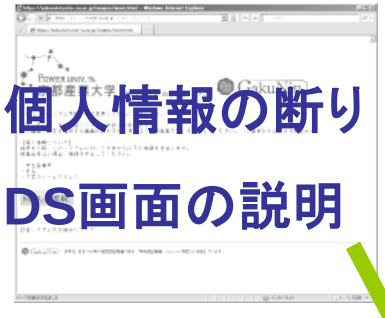
- ◆既存の学内システムは，お知らせ掲示板を見て欲しいので，まずはポータルサイトに誘導
- ◆全ての学内システムはポータルサイトからリンクが張られており，いくつかのシステムは擬似SSOも設定
- ◆この流れを阻害せず，融合

目指す総合認証基盤


ポータル
（学内SP）


学内SP


京都産業大学
IdP


個人情報の断り
DS画面の説明


moodle
学内SP


GakuNin
Discovery Service
DS


学外SP

具体的な設定

◆学内SP利用時にDS画面は不要
- 学内SPはデフォルト認証を学内IdPに向ける

◆DSがトラブルを起こしても学内は動く
- 学内IdPに学内SPのメタデータを固定でセット
- 学内SPに学内IdPのメタデータを固定でセット
- （加えて学認メタデータを自動更新設定）

◆学外SPはデフォルト認証がNIIのDSに向いている
- この場合は，IdP認証を済ませていても，アクセスするときにDS画面が出る（DS画面でIdPを選択した後はIdPのログインは終わっているので，認証後処理へ進む）

構築コスト

- ハードウェアは通常のサーバ
- ソフトウェアはオープンソースなので構築を内部の人間ができればタダ（人件費のみ）
  - 「学認」の方からの強力な支援もあります
- 業者に委託すると…x百万？
- （必要に応じて）LDAPの属性追加
- 学内の承認を取り付ける障壁
- 利用者にシステム切替を説明するコスト

運用コスト

◆IdPの設定変更
- 利用したいSPが増えた時に，そのSPに渡す属性の設定
- 署名証明書の更新

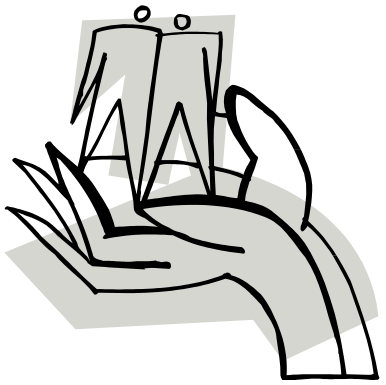
◆LDAPの設定変更
- 利用したいSPから要求される属性がLDAPに登録されていなければ追加

◆ハードウェアの保守
- 通常のサーバと同じ

◆各ソフトウェアのセキュリティアップデート
- 通常のサーバと同じ

大学間共有eラーニングシステムへの発展

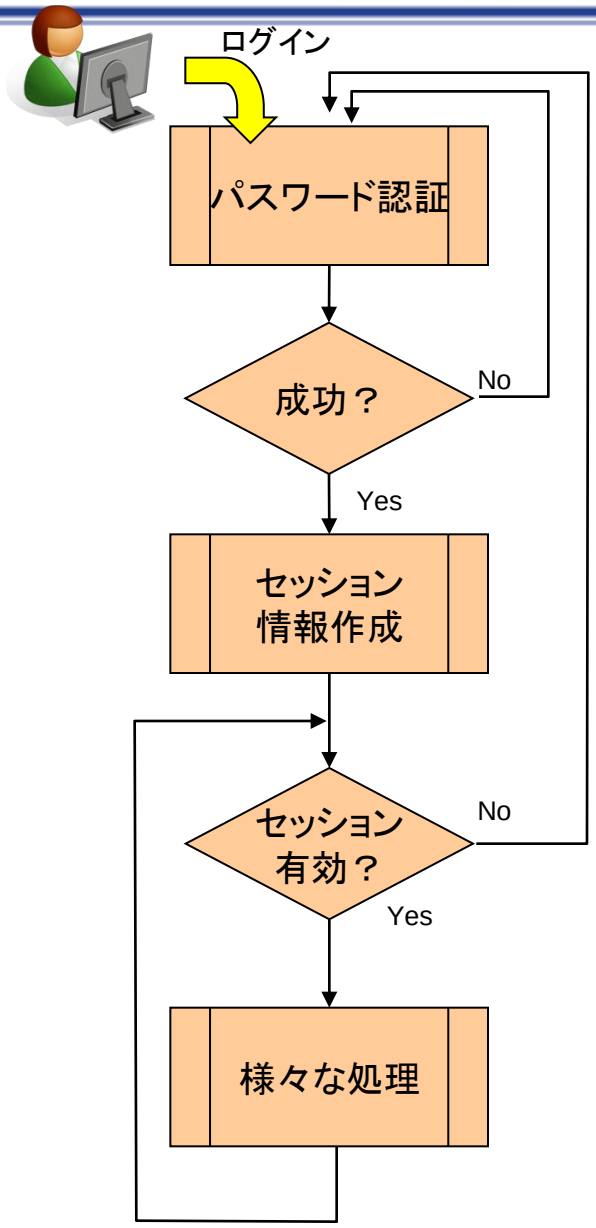
- ◆基盤は整った
  - ➢ 大抵の連携事業の要求に応えられるはず
  - ➢ 学認は大学間共有のシステムを作成する基盤として非常に優秀
- ◆後は参加者が増えれば自然と向かうべき方向にすすむだろう
  - ➢ 色々な学術SPの登場に期待
  - ➢ クラウド事業にも向いている
  - ➢ 実際，Gmailとも連携が可能



付録

- 以降のスライドは技術的な情報です
- 時間がないので説明しませんが，参考になれば幸いです

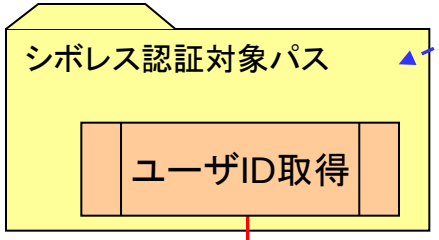
既存システムのSP化



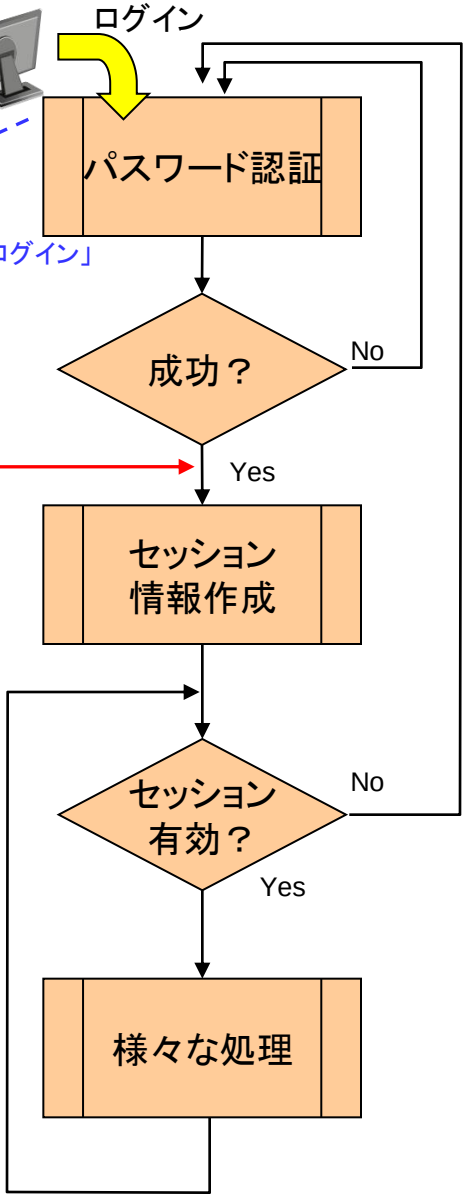
◆ 通常のログイン処理

- パスワードが正しいことを確認する
- セッション情報を作成する
- セッション情報が有効であることを確認しながら動作する
- パスワード認証は最初だけ

既存システムのSP化



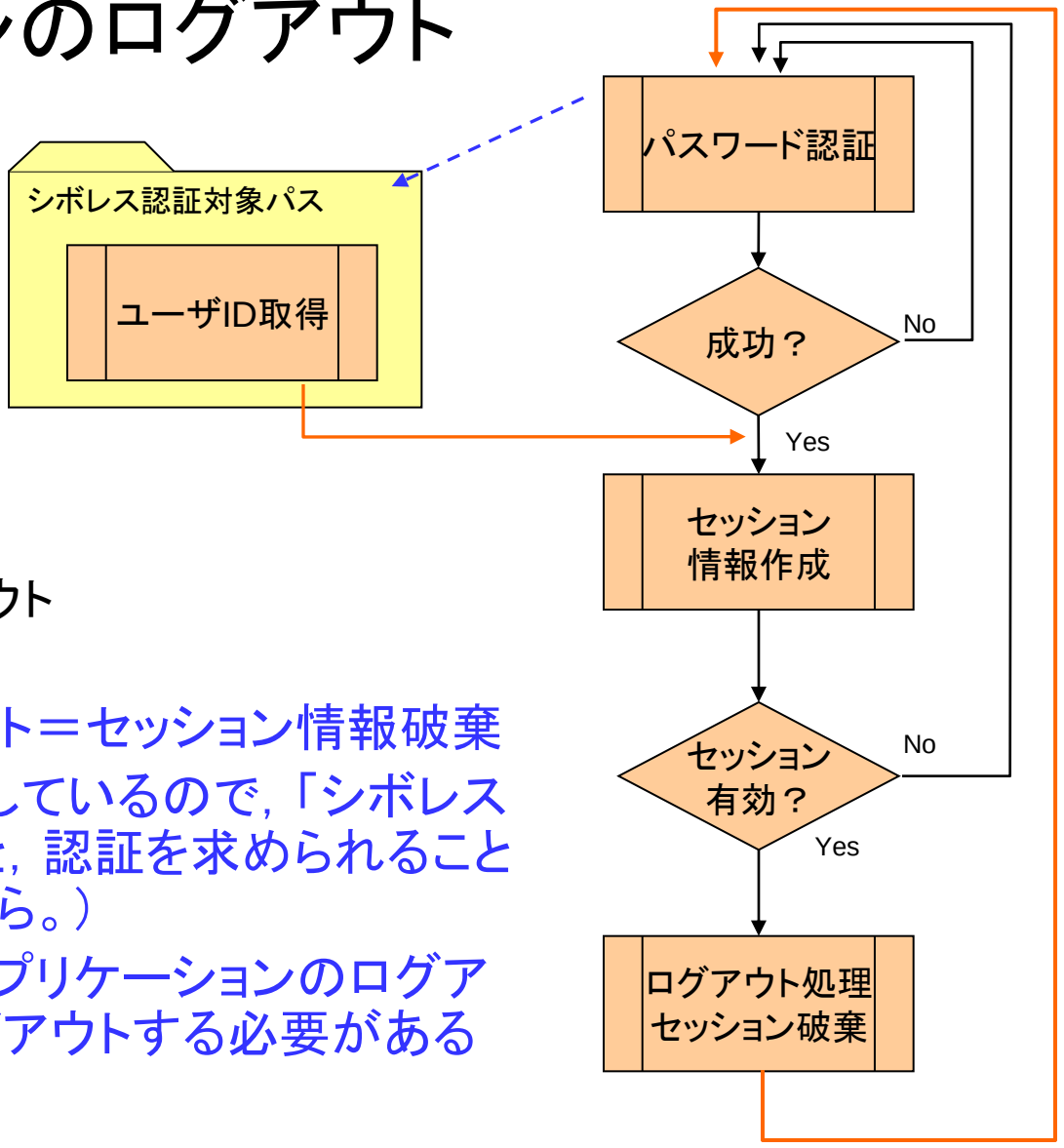

```
<Location /auth_dir>
   AuthType shibboleth
   ShibRequireSession On
   require valid-user
</Location>
```



◆ SP化

- ログインに迂回路を作成する
- シボレス認証が必要なパスを作成する
- ユーザIDを環境変数で受け取り，セッション情報作成処理に合流するプログラムを置く
- ここに置いたプログラムにアクセスしようとすると認証ページに飛び，認証に成功するとプログラムが実行される
- 乱暴に言えば以上
- 既存のログインも併用可能

アプリケーションのログアウト



- 2種類のログアウト
  - アプリケーションのログアウト
  - シボレスのログアウト
- アプリケーションのログアウト＝セッション情報破棄
- シボレスのログインは継続しているので，「シボレスでログイン」リンクをたどると，認証を求められることなく利用できる（SSOですから。）
- 再認証を求めたければ，アプリケーションのログアウトと同時にシボレスもログアウトする必要がある

シボレスのログアウト

- シボレスのログアウトとは
  - IdPのログアウト＝IdPのCookieの削除
  - SPのログアウト＝SPのCookieの削除
- IdPだけをログアウトすると
  - SPアプリケーションはそのまま使い続けることができる
  - アプリのセッションだけで動いている時はもちろんだが，シボレス制限ファイルにもアクセス可能。SP側がキャッシュしているような動き。実際，SPのデーモンを再起動するとアクセスできなくなる
  - 新たにSPにアクセスする際にIdPで認証が求められる
- シングルログアウト（IdPをログアウトしたときに，すべてのSPをログアウトさせる）を行うには，かなり作りこみが必要（IdPからSPのCookieは消せないから。）
  - ただ，メールをログアウトしたら書きかけのmoodleレポートが消えた，とかいう状況はまずいだろう。シングルログアウトは不要では？
- シボレスをログアウトするには？
  - Cookieを削除すればいいので，技術的にはブラウザを全部終了させること
  - ただ，そういう指導をするのは教育上よくない。ログアウトは意識させたい

アプリケーションとシボレスの同時ログアウト

- SPの「ログアウト」メニューを選択した際，SPアプリ独自のセッション情報をクリアした後，SPの次のリンクに飛ばす
  - /Shibboleth.sso/Logout?return=https://idp.domain/logout.html
    （SPのCookieを削除してhttps://idp.domain/logout.htmlに飛ぶ）
  - 「/Shibboleth.sso」はシボレスのSPデーモンが処理する特別なURL
- https://idp.domain/logout.html には次の情報を含める
  - <meta http-equiv="Set-Cookie" content="_idp_session=; path=/idp; expires=Sat, 1-Jan-2000 00:00:00 GMT; secure">
  - 要するに，IdPの「_idp_session」というCookieを無効にする
- 必要に応じてlogout.htmlから更にユーザが戻るべきリンク先を示せばよいだろう
  - logout.htmlをCGIにしておくと，SP毎に特定の引数を持たせて呼び出すことで戻り先を動的に変更できて便利