Direction des systèmes d'information

Journée fédération 2013

Réflexions et retours d'expérience sur l'intégration des web services et du fournisseur d'identité du CNRS

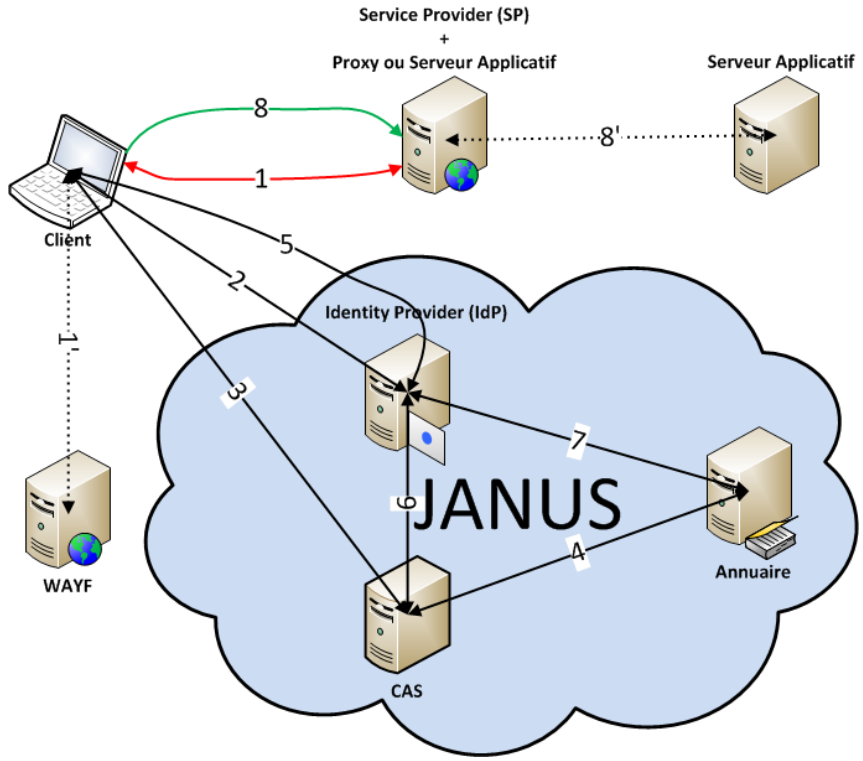
Plan

- Contexte
  - ✓ Janus et Web Services
- Cas d'usage & typologies
- Réflexions, solutions envisagées
  - ✓ POC
- Pistes
- Conclusion

CONTEXTE

Contexte

- **Direction des Systèmes d'Information du CNRS**
  - ✓ Fonctions support
- **Mise en place de JANUS en 2009**
  - ✓ Utilisation Certificat, (login/mot de passe)
  - ✓ Shibboleth reverse-proxy uniquement
- **Approvisionnement métier de l'annuaire**
  - ✓ 120 000 identités
  - ✓ 500 écritures/jour



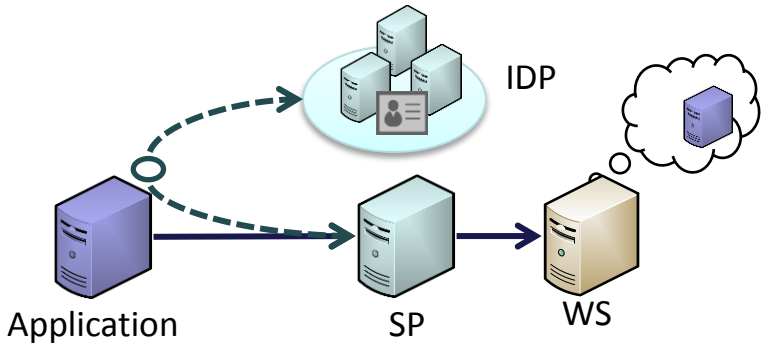
Les Web Services à la DSI

- **Utilisation des WebServices (WS)**
  - ✓ Communication interapplicative
  - ✓ Principe d'architecture d'application
    - ➢ HTML5, Javascript, mobile
- **Vers une Architecture Orientée Service ?**
  - ✓ ~ 10 WS (REST & SOAP)
  - ✓ Accès à des référentiels
  - ✓ Synchronisation d'état
- **Stratégie de sécurisation ?**
  - ✓ Shibbolisation d'application => WS
  - ✓ WS existant ?
  - ✓ Stratégie et coût
    - ➢ Globale, cas par cas, …

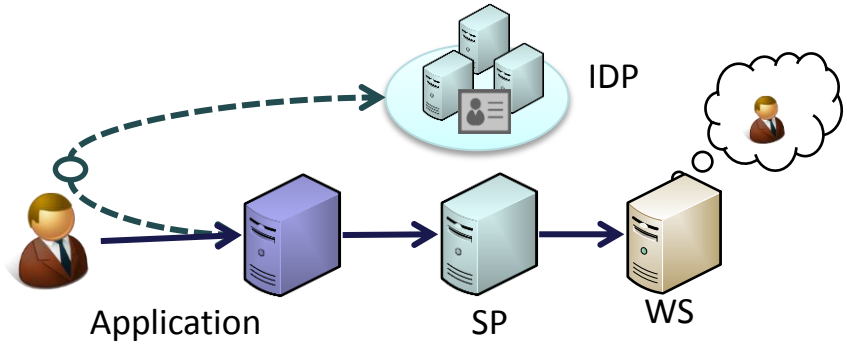
CAS D'USAGE

Cas d'usage

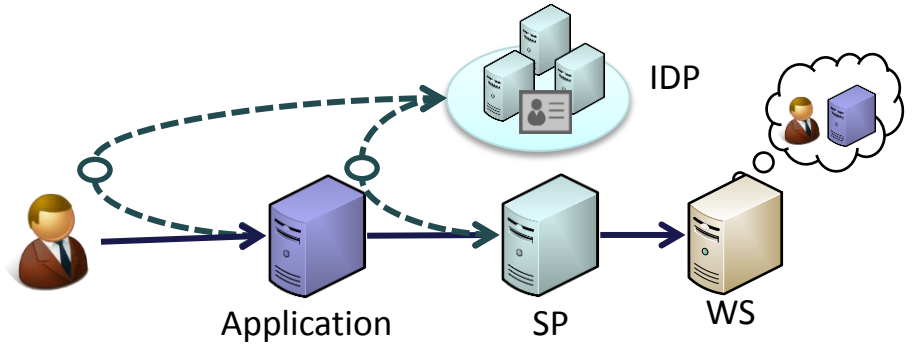
- **Inter-applicatif**
  - ✓ Par application
  - ✓ Référentiels de données



- **« Pour le compte de »**
  - ✓ Par utilisateur
  - ✓ GED, …



- **Mixte**
  - ✓ Par Application ET utilisateur
  - ✓ Opérations sensibles



Un WS est une Ressource comme une autre

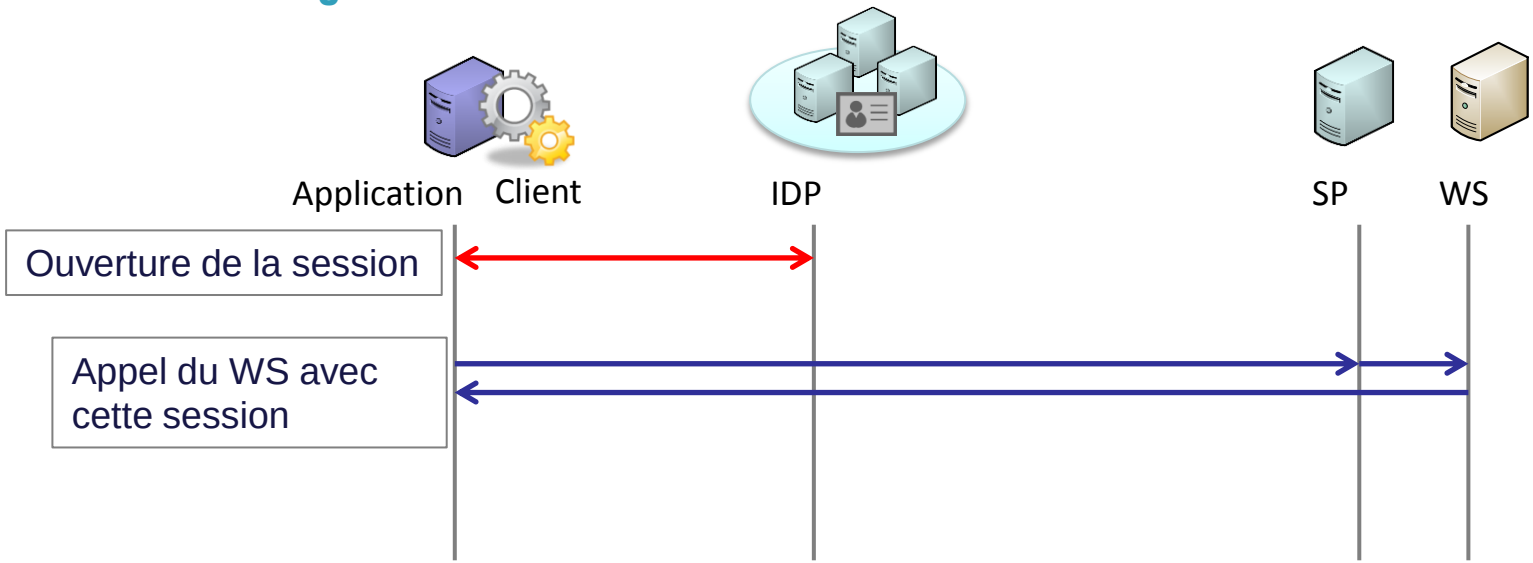
- **Mécaniques HTTP connues**
- **Mais…**
  - ✓ **HTTP 302**
  - ✓ **WAYF ?**
  - ✓ **Formulaire**
- **Dialogue complexe !**

Solutions envisagées

RÉFLEXIONS

Inter Applicatif

- **Machine to Machine**
  - ✓ Simuler une navigation (WAYF, Redirect, mire de login)
    - ➢ Renseigner le formulaire
  - ✓ Récupérer la session Shibboleth
  - ✓ Appeler le WS *via* cette session
- **Faible complexité**
  - ✓ Configuration

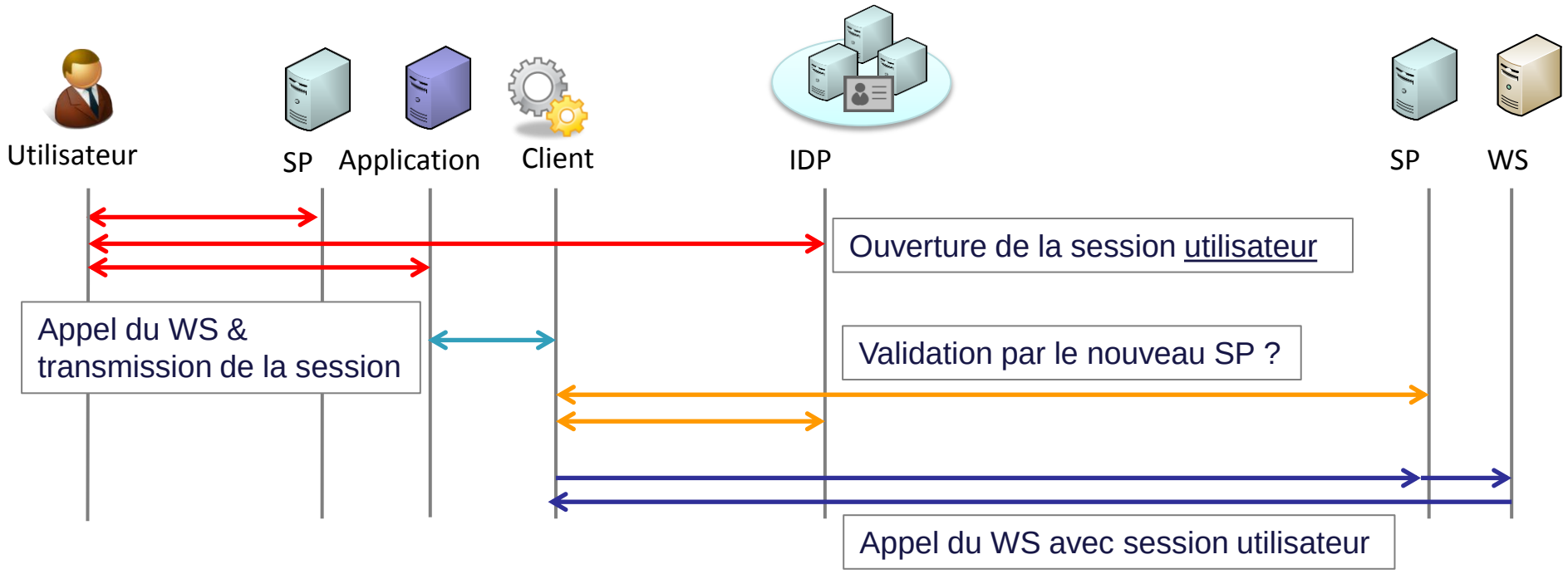


Points d'attention

- Ne pas appeler la ressource
- Gestion du Javascript suivant les implémentations (du WAYF, du SSO)
- Expiration de la session
- Objet session ?

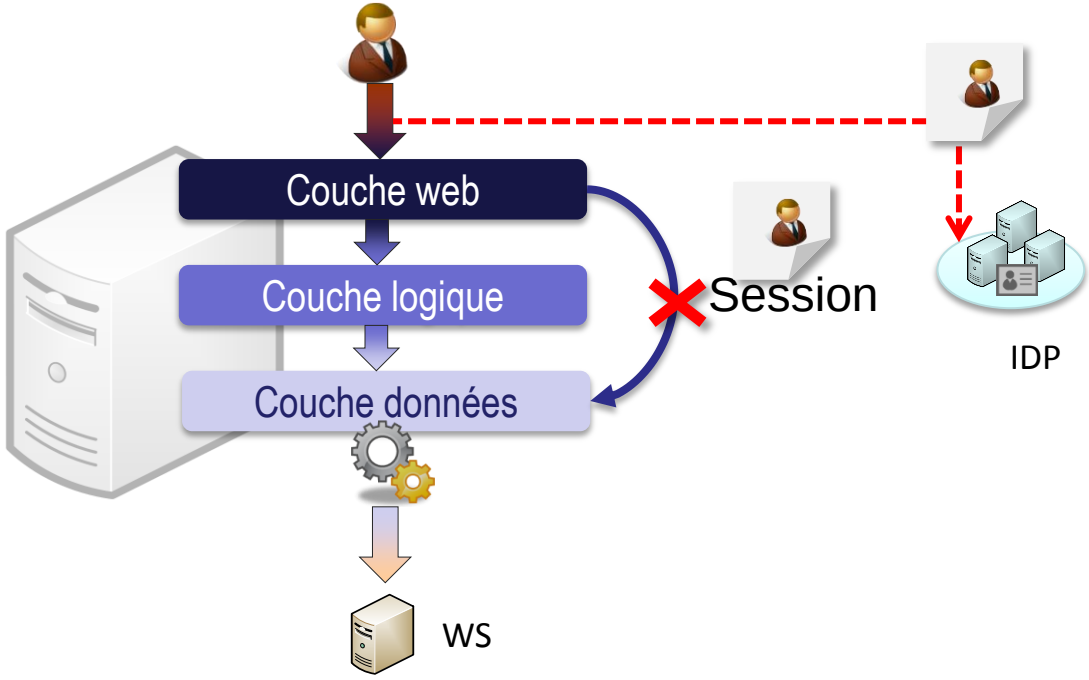
« Pour le compte de »

- **Deux phases**
  - ✓ Obtenir session côté utilisateur
  - ✓ Réutiliser informations de session côté client
- **Oui mais…**
  - ✓ Quelles informations?



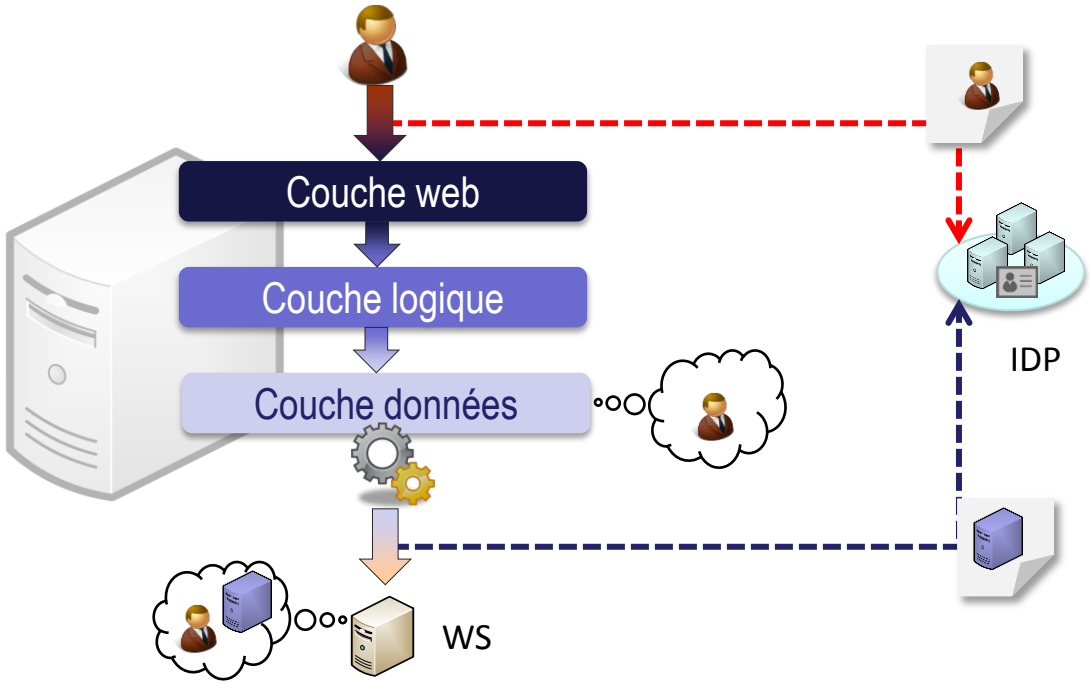
Problème de design

- **Architecture en couches**
  - ✓ Limitation du couplage
- **Transmission session utilisateur**
  - ✓ Introduction couplage
    - ➢ HTTP <-> métier
  - ✓ Coût élevé de l'adaptation



Identification du tiers : la solution ?

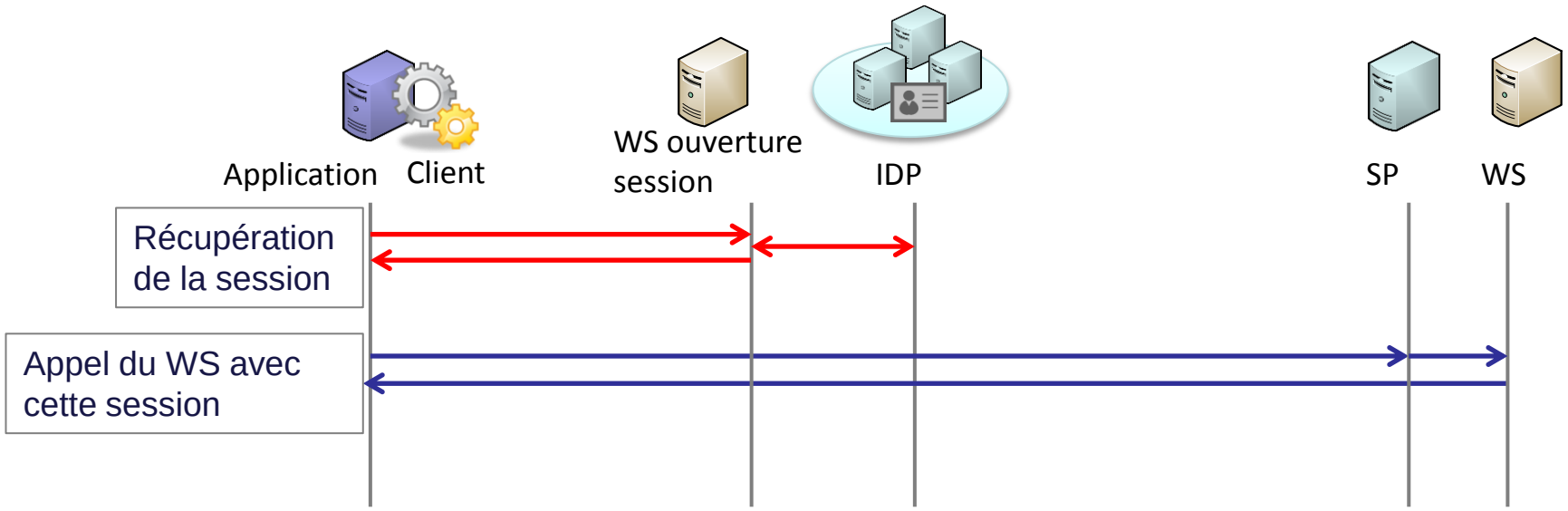
- **Déléguer l'identification à l'application**
  - ✓ Relation de confiance avec le WS
  - ✓ Transmission de l'identité de l'utilisateur
  - ✓ Respect de l'architecture en couche
- **Charge de travail côté WS**



PISTES ET CONCLUSIONS

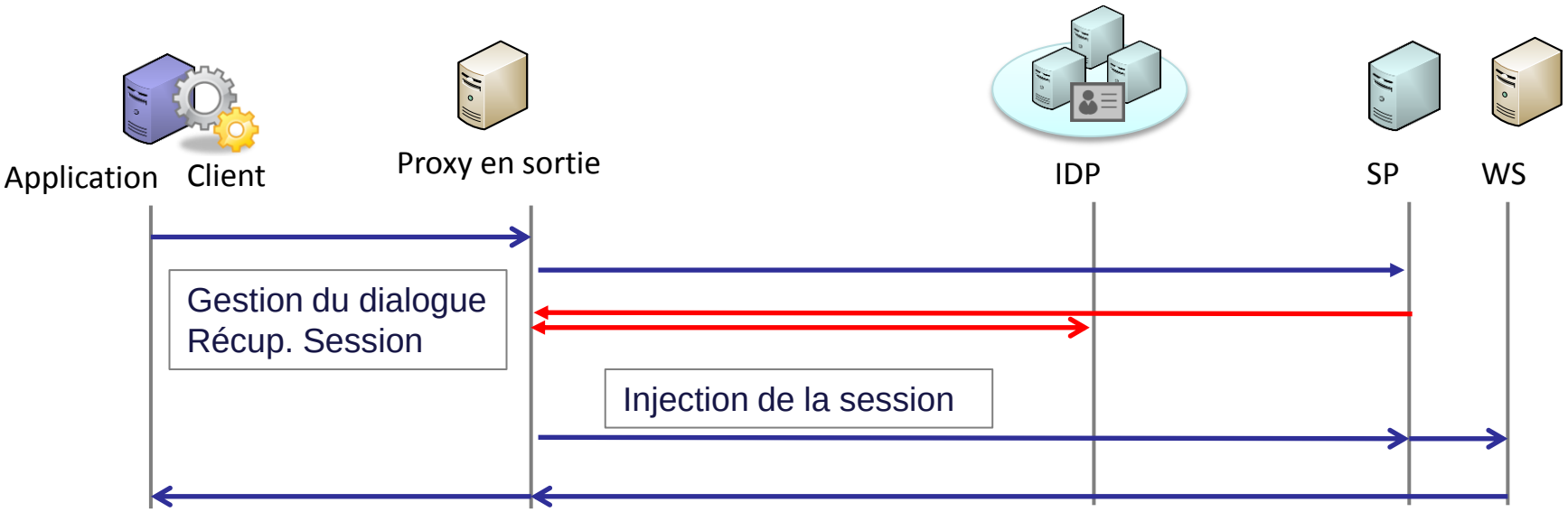
Inter Applicatif: Déléguer le dialogue

- **Machine to Machine**
  - ✓ Factoriser la récupération de la session
  - ✓ Présenter une façade unifiée et agnostique
- **Optique développeur**



Proxy intercepteur en sortie dédié

- **Déléguer la gestion du dialogue**
  - ✓ Intercepter, injecter
  - ✓ Limiter le coût
- **Optique administrateur**
  - ✓ Solution
    - ➢ Agnostique et locale (=/= globale)



Conclusion

- **Plusieurs solutions**
  - ✓ À tester et valider
  - ✓ A adapter
    - ➢ Contexte
    - ➢ Culture
- **Effort certain !**
  - ✓ Résultat prometteurs
  - ✓ Encore du travail avant d'être mature
- **Toutes les idées sont les bienvenues**

Merci

- Codes sources des PoC disponibles :
  - ✓ https://github.com/aresupie/journee-renater-2013-05-27

- Marc DEXET (marc.dexet@dsi.cnrs.fr)
- Stéphane DERACO (stephane.deraco@dsi.cnrs.fr)

ANNEXES

Script CURL avec de délégation d'ouverture de session

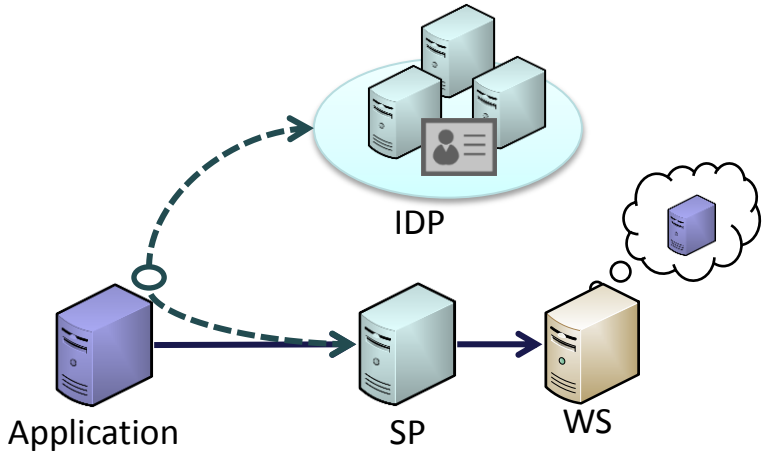
```
LOGIN=$1
RESOURCE=$2

# Recuperation du cookie
COOKIE=$(curl -k -s -u $LOGIN "http://localhost:13000/jaxrs-service/janus/hello/cookie?url=$RESOURCE")
DATA=$(echo $COOKIE | cut -d';' -f4-5 | tr ';' '=')

# Appel de la ressource avec le cookie
curl -k -b "$DATA" "$RESOURCE"
```

Autres pistes et ouvertures en vrac

- Apache CXF implémente un SP (http://cxf.apache.org/docs/saml-web-sso.html)
- Authentification de l'utilisateur sur une page web, lancement d'un client lourd Java via JNLP (Java WebStart)
- Passer « par derrière » (backoffice)
- Ouvrir des canaux dédié aux WS dans l'IdP

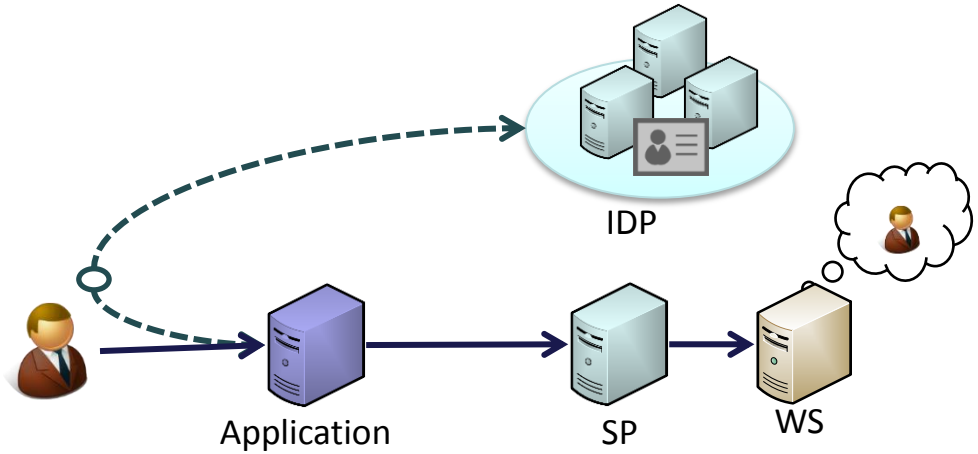
Cas d'usage: Inter Applicatif

- **Machine to Machine**
  - ✓ Application invoquant un service
  - ✓ Ressource « protégée » par un Service Provider
  - ✓ Application authentifiée et identifiée
  - ✓ Restitution par application
- **Typologie**
  - ✓ Référentiels communs



Cas d'usage : « Pour le compte de »

- **On Behalf**
  - ✓ Application invoquant un service « Pour le compte de »
  - ✓ Utilisateur authentifié et identifié
  - ✓ Restitution par utilisateur
- **Typologie**
  - ✓ Gestion Electronique de Document,
  - ✓ Données de gestion transverse …



Cas d'usage mixte

- **Accès utilisateur ET application**
  - ✓ Application invoquant un service « Pour le compte de »
  - ✓ Utilisateur et application auth/id/entifiée
  - ✓ Restitution par utilisateur et application
- **Typologie**
  - ✓ Accès canalisés (système RH)
  - ✓ Contraintes de sécurité élevée