Authentification multi plate-forme dans un système d'information universitaire

JRES 2005 marseille

Consortium

COCKTAIL

www.cocktail.org

- Espace de mutualisation et d'échange applicatif
- Une quarantaine d’établissements
- Domaines :
  - Scolarité
  - Gestion financière GFC
  - Gestion des ressources humaines GRH
  - Paie
  - …

Besoins de sécurité

- Manipulation de données sensibles
- Utilisateurs nomades
- Diversité des architectures
- Perte de contrôle du poste de travail
- Authentification forte

Les utilisateurs

Où est cette application ?

\+ Ai-je la dernière version ?

\+ S’authentifier à chaque fois

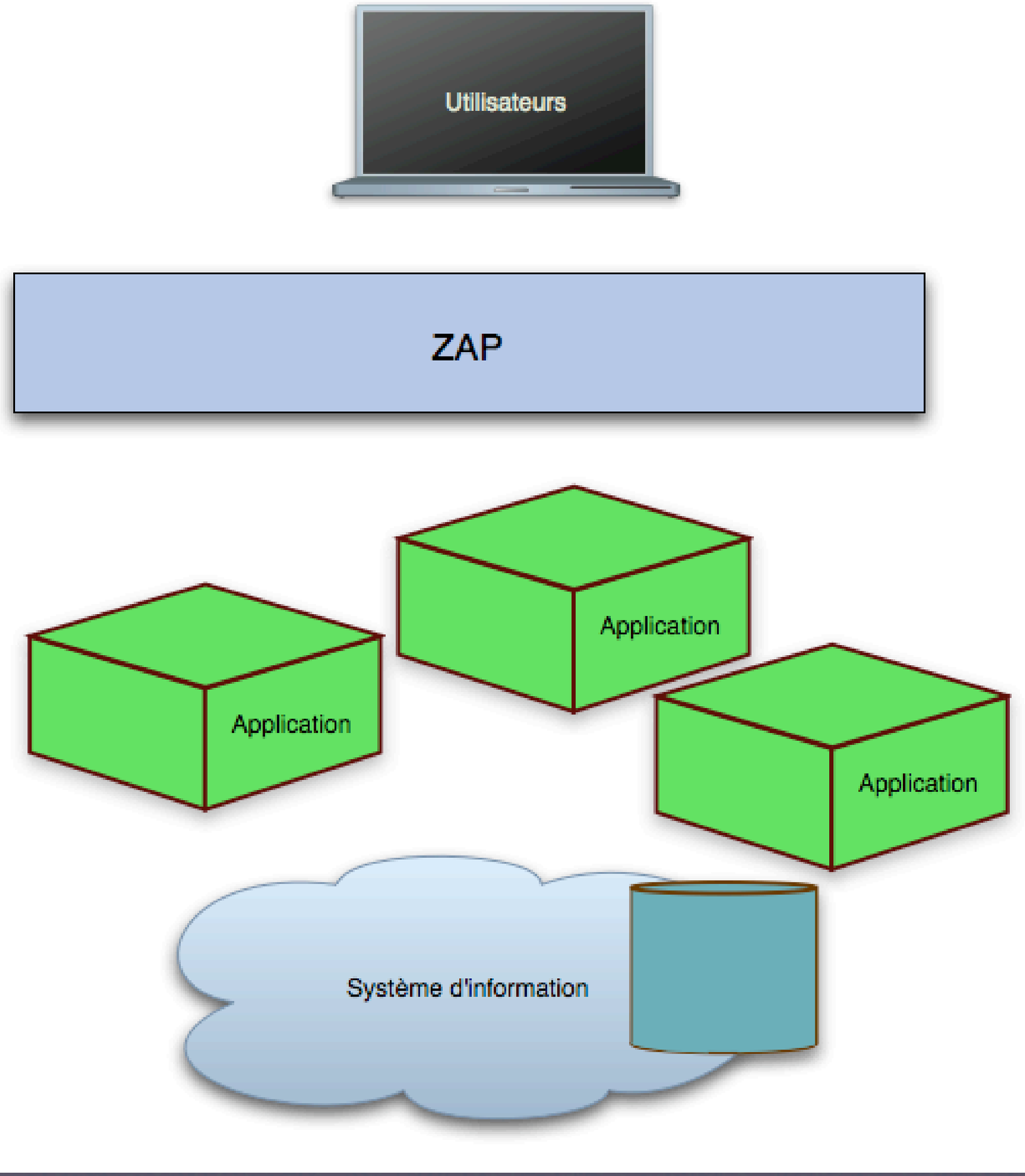
Un objectif

- Sécurité d'accès au système d'information
- Virtualiser l'accès aux applications (localisation, version, présentation, ...)
- Un seul lanceur
- Unifier l'authentification auprès de toutes les briques donnant accès au SI
- Un accès rapide aux applications favorites

Une solution

Zap

zone d’applications “simple comme une télécommande”


Utilisateurs
ZAP
Application
Application
Application
Application
Système d'information

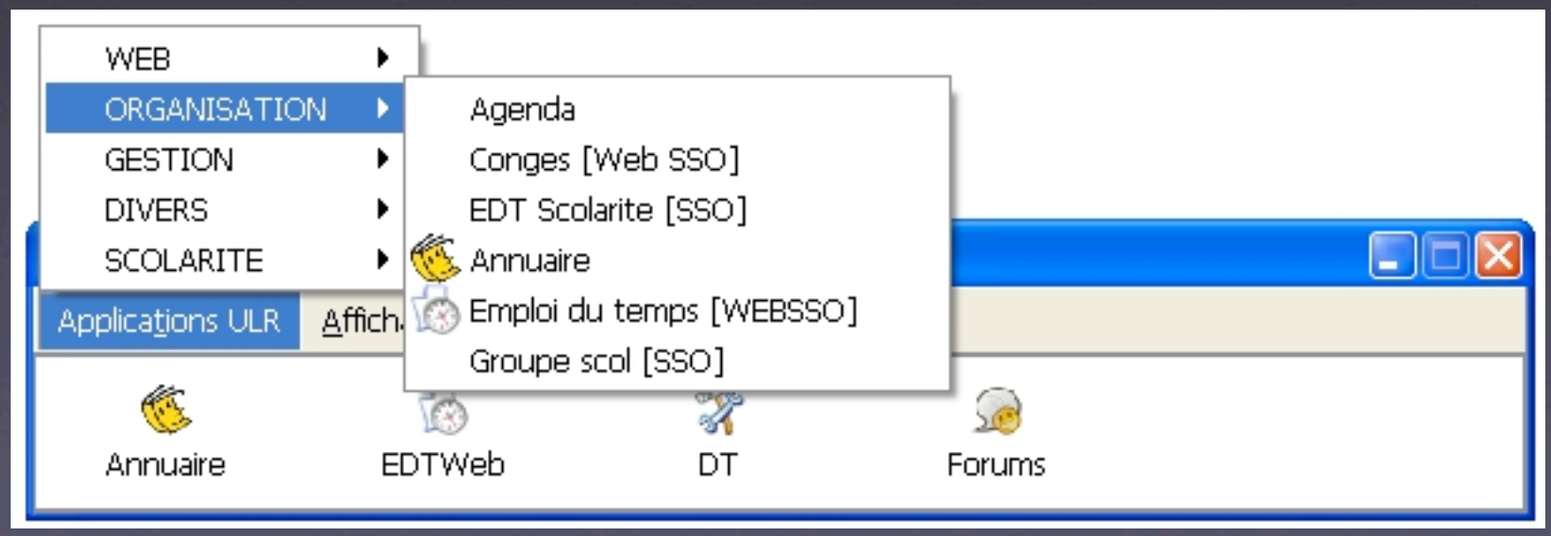
Un lanceur d'applications

- Multi application
- Multi plate-forme

- Authentification unique Single Sign On
- Recensement dynamique en temps réel des applications disponibles

Une interface graphique simple

- Un menu contenant toutes les applications
- Une barre d'outils personnalisable
- Développée en Java Swing



Une description XML

- Contient toutes les applications disponibles
- Elément graphique = description XML
- <application> : description graphique + description applicative
- <theme> : classer les applications

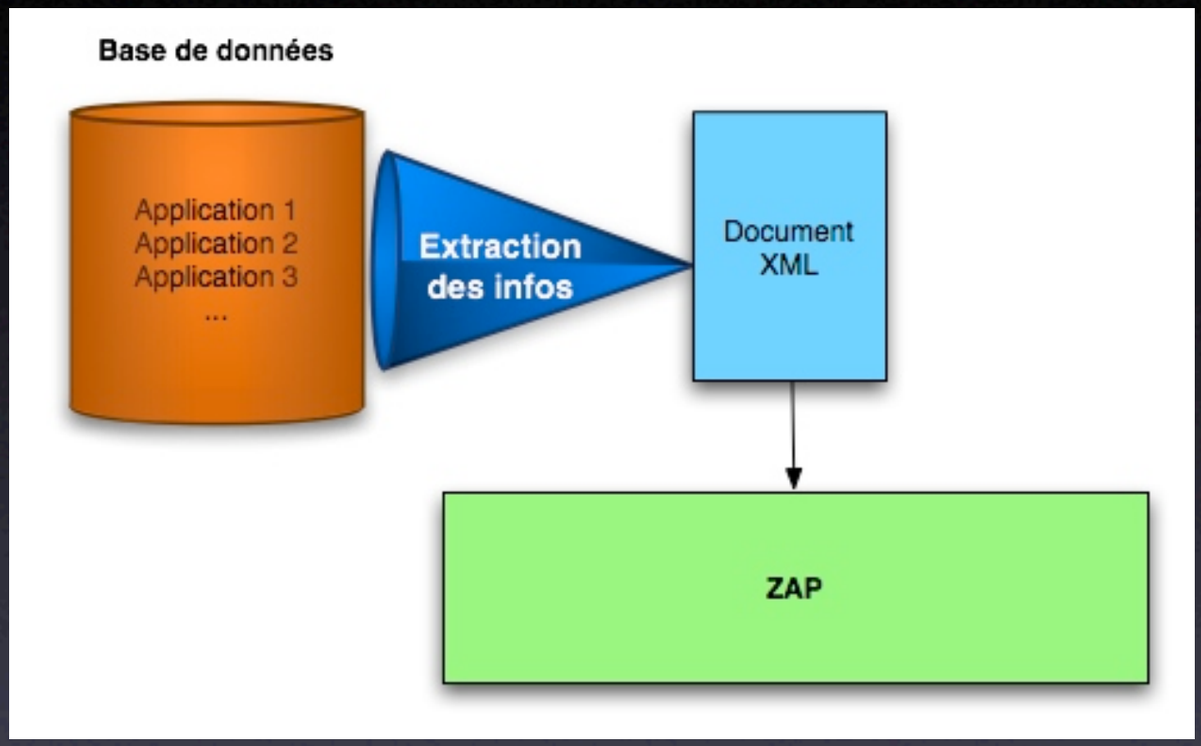
```
<theme
    name="ORGANISATION"
    comment="Applications d'organisation">

    <application
            name          =   "Annuaire"
            iconUrl       =   "http://www.univ-lr.fr/icones/annuaire32.png"
            comment       =   "Annuaire de l'etablissement"
                  location        =   "V:\Public\Apps\Annuaire.app\Annuaire.exe"
                  authentication=   "login"
                  type            =   "ExeWindows"
        />

    ...
  </theme>
...
```

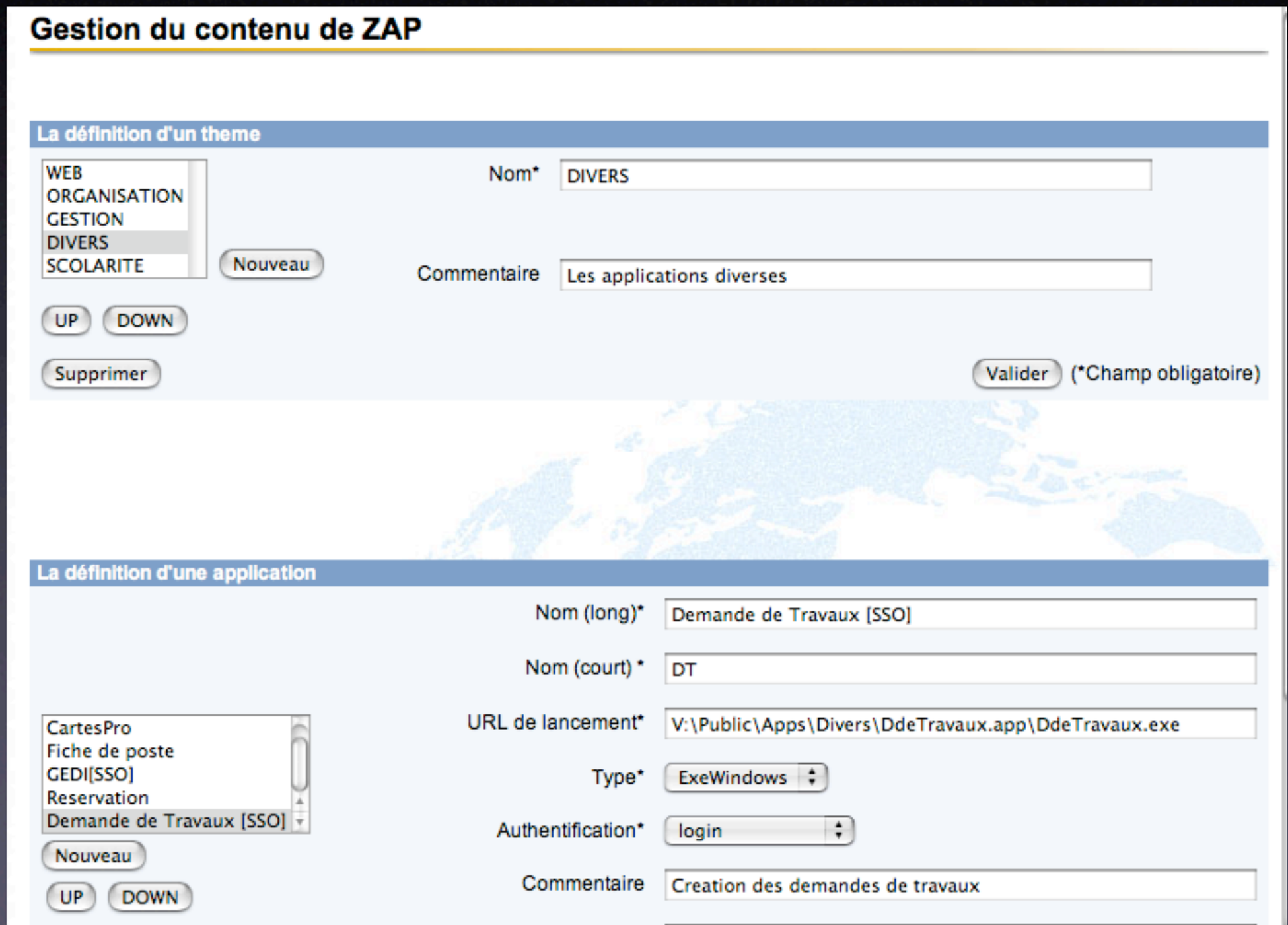
- Téléchargement XML au démarrage
- Filtre par OS
- Contenu du menu stocké dans une BDD
- Contenu de la barre d'outils personnalisable stocké dans le "home" de l'utilisateur

Une source dynamique



- Gestion des applications offertes : Servaut (Serveur d'authentification)
- Servaut : Authentification des applications
- Possibilité de source statique

Une administration centralisée



Un lanceur

- Code ouvert permettant de créer son propre lanceur
- Lanceurs déjà écrit :
  - Exécutables natifs : ligne de commande
  - Applications web : via un navigateur web
  - Applications Java Web Start

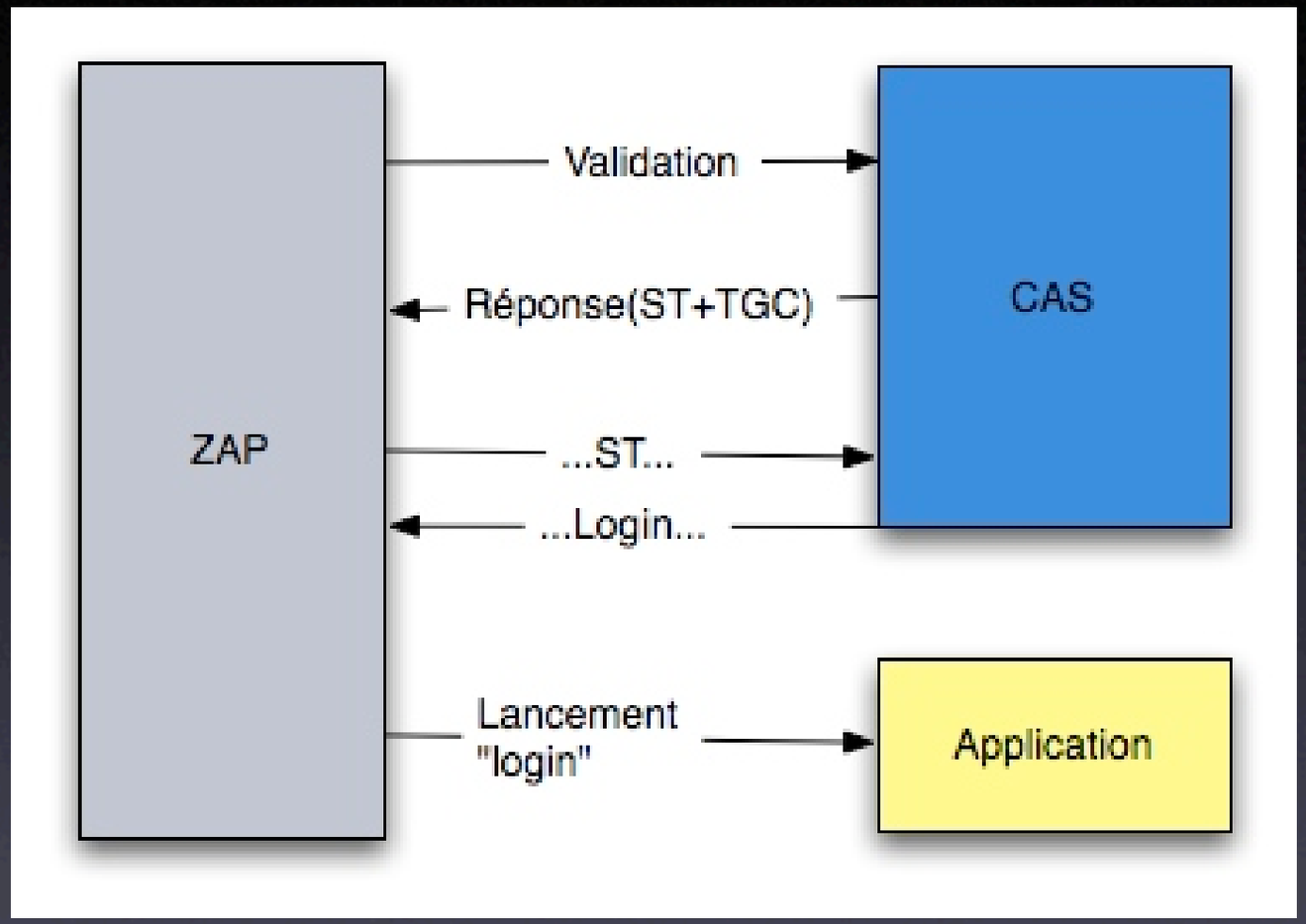
Single Sign On

- Une authentification pour *n* applications

Solution basée sur CAS

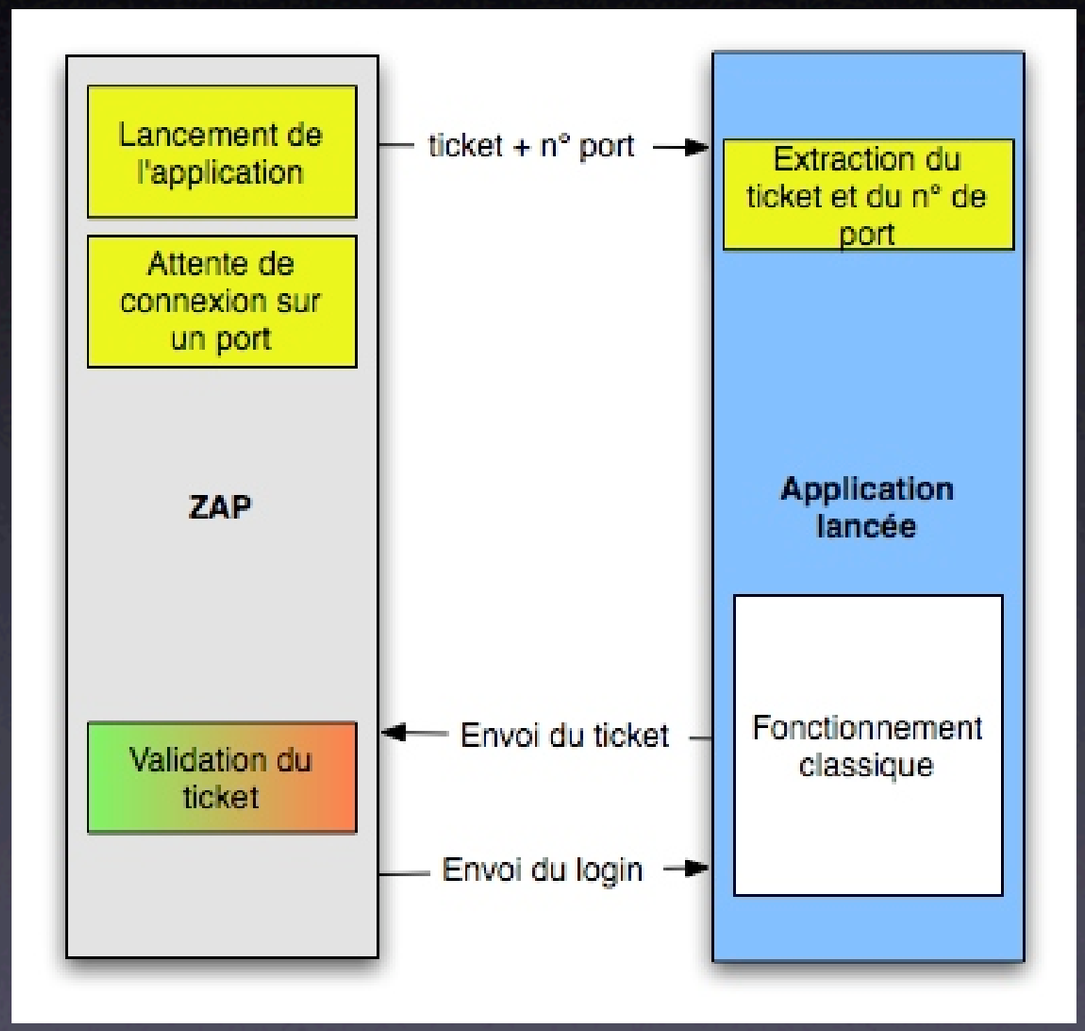
- Produit fiable et utilisé dans la plupart des établissements
- Authentification pour les applications web
- ZAP n'est qu'une extension de CAS

CAS -- ZAP



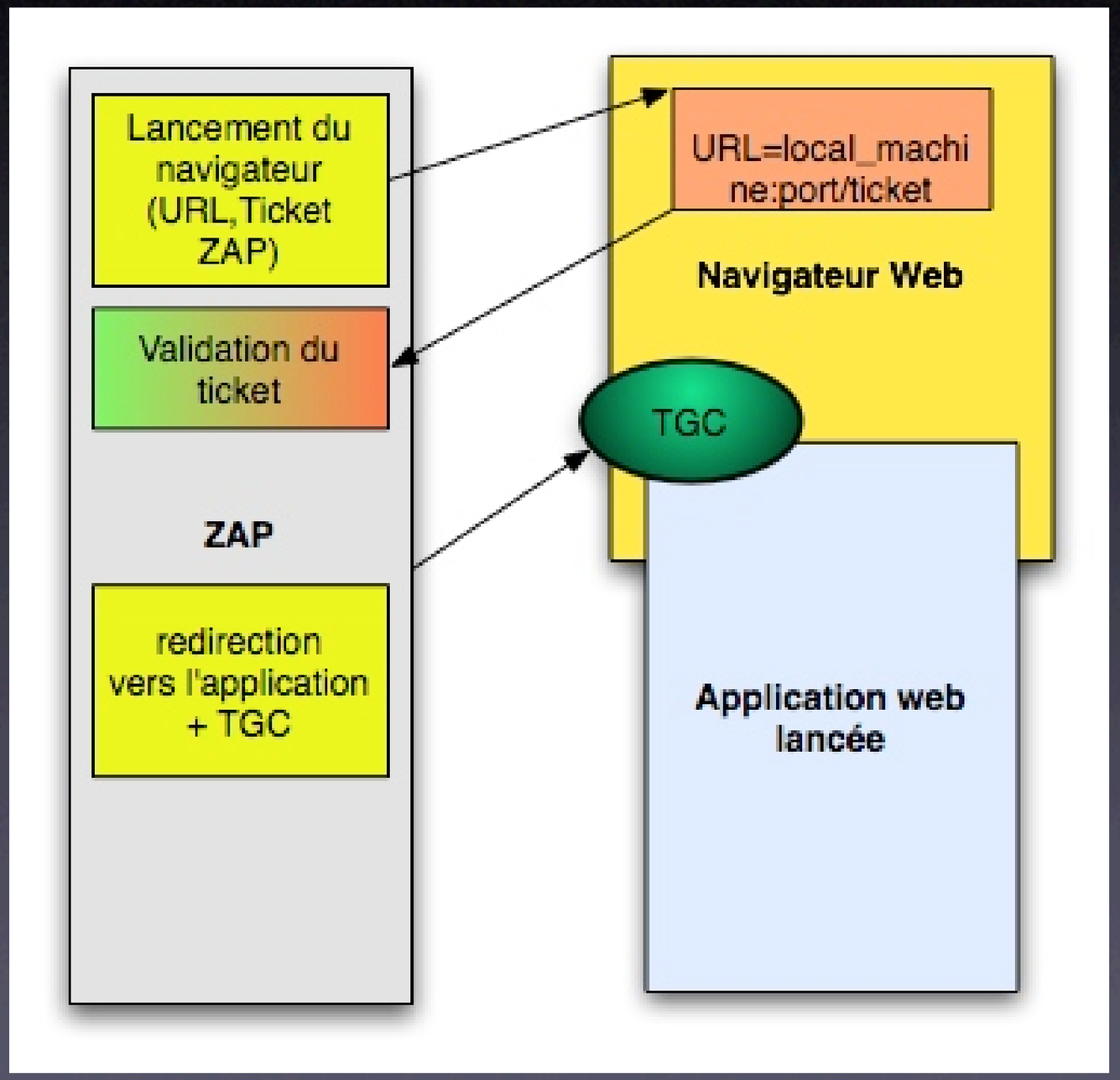
- Communication sécurisée : HTTPS
- Validation du certificat serveur
- Revalidation de l'authentification

Lancement d'applications



- Transmission de l'identifiant à l'application
- Principe du callback

ZAP
et les applications web



L’authentification

- 3 niveaux d’authentification
  - Nul
  - Login/passwd (CAS)
  - Certificat (ZAP)
- Distingué par application
- Login + certificat

Les certificats

- Utilisation de certificats X509 délivrés par le CRU
- Token USB PKCS#11

- Authentification forte (code pin + token)
- ZAP vérifie le certificat : validité, origine, non révocation, code de l'établissement.
- CRL téléchargées à chaque validation
- Possibilité de recherche dans un annuaire LDAP

Développement pour ZAP

- Les applications Java et exécutables natifs doivent être adaptées/zappifiées
  - Prise en compte des paramètres de callback
  - Connexion à zap
  - Gestion des erreurs
- Les applications web doivent être compatibles CAS

A venir...

- Meilleure utilisation des certificats sous Mac OS X (keychain.app, drivers Rainbow)
- Utilisation du “proxy CAS”
- Java 1.5, PKCS#11
- Interface graphique
- Nouveaux lanceurs

Conclusion

- ZAP est distribuée au sein du consortium cocktail
- Résolution des problèmes du déploiement hétérogène
- Simplification du travail de l'administrateur
- Facilité pour les utilisateurs

Remerciements :
Arunas Stockus
Jean-Marc Coris
Hugues Villesuzanne
Le Cri de La Rochelle

?!