Nathanaël COTTIN

Systèmes distribués et sécurité informatique

Principes fondamentaux

Nathanaël Cottin
www.ncottin.net

version 1.0.1 – 2010

Nathanaël COTTIN

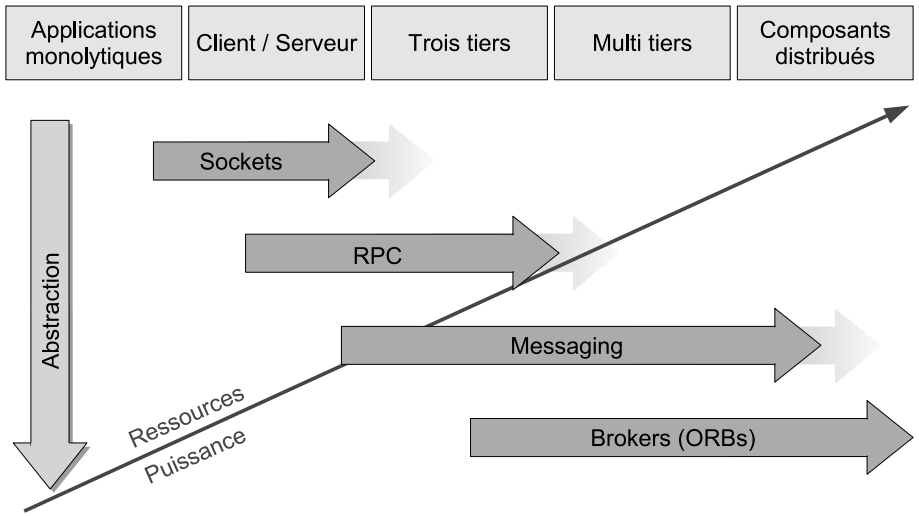
Partie 1

Les systèmes distribués

- Concepts de référence
- Présentation des architectures CORBA et SOA

Nathanaël COTTIN

Historique



Nathanaël COTTIN

Les systèmes distribués (ou répartis)

« Un système réparti s'exécute sur un ensemble de machines dépourvues de mémoire partagée mais que pourtant l'utilisateur voit comme une seule et unique machine » [TAN 94]

Un système distribué est un paradigme définissant un ensemble de composants œuvrant de concert et concourant à proposer des services à leurs clients

Les composants interagissent indépendamment de leur localisation par le biais d'un « logiciel du milieu » (intergiciel) ou middleware

Objectifs des systèmes distribués

- **Efficacité :**
  - Appels
  - Traitements
  - Retours
  - Erreurs
- **Flexibilité :**
  - Modularité
  - Transparence
    - Interopérabilité
  - Évolution dynamique
- **Consistence :**
  - Cohérence
  - Respect du comportement
- **Robustesse :**
  - Disponibilité
  - Sécurité

Ces objectifs permettent de définir des critères de qualité de service

Principaux avantages

- **Mise à l'échelle (globalisation des réseaux)**
- **Fonctionnement en mode dégradé**
- **Montée en charge**
- **Modularité**
- **Évolutive**
- **Transparence**
- **Indépendance interfaces / traitements**

Difficultés rencontrées

- **Efficacité :**
  - Taille des données transmises
  - Gestion des requêtes en attente
  - Degré de corrélation des composants
  - Latence
- **Flexibilité :**
  - Migration / réplication
  - Intégration de nouveaux composants
  - Partage de ressources
  - Composition
- **Consistence :**
  - Gestion transactionnelle
  - Interopérabilité
  - Globalisation des réseaux
  - Plus forte probabilité de panne
- **Robustesse :**
  - Disponibilité
  - Persistance (services et informations)
  - Gestion distribuée des autorisations et droits d'accès

Services communs normalisés

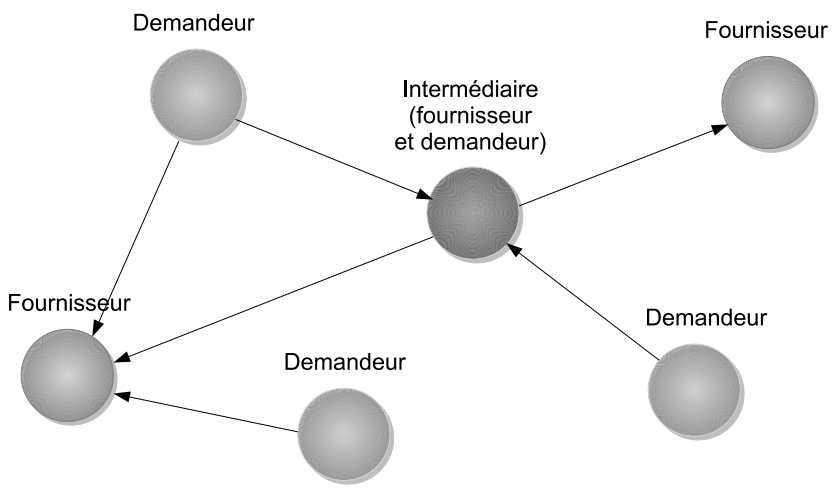
Un service est un composant serveur ou intermédiaire normalisé fourni avec l'environnement de déploiement

Les services participent au bon fonctionnement du système

Exemples :

- Naming Service
- Trading Service
- Life cycle Service
- Transaction Service
- Time Service
- Event Service
- Security Service

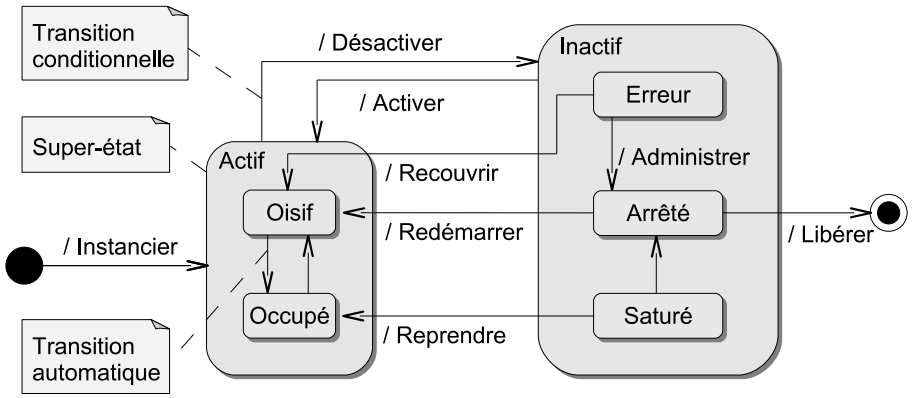
Interactions entre composants distribués



Corrélation entre composants distribués

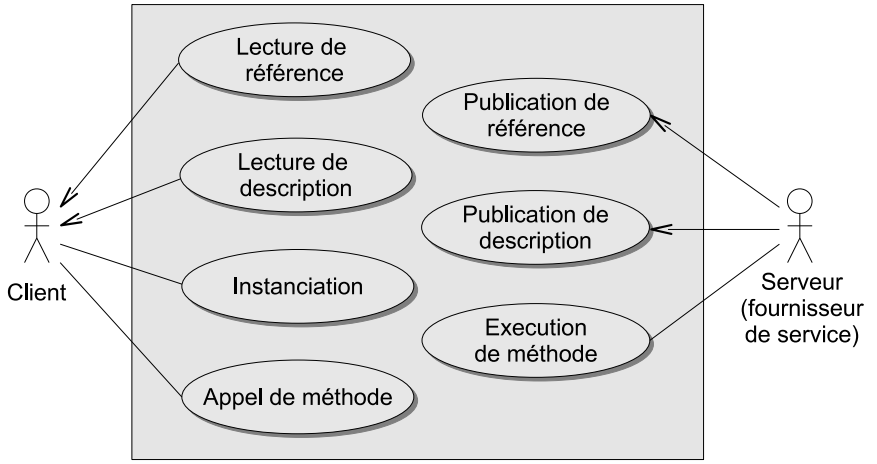
- **Système distribué = graphe orienté où les nœuds sont des composants :**
  - Composants univoques : orientation serveur
  - Composants intermédiaires (mixtes) : à la fois serveurs et clients
  - Composants clients
  - Possiblité d'appliquer les algorithmes sur les graphes
- **Agrégation et composition**
- **Couplage global du système**
- **Définition du degré d'interaction entre composants**
- **Indicateurs pour décisions de migration (supervision)**

Cycle de vie d'un composant distribué



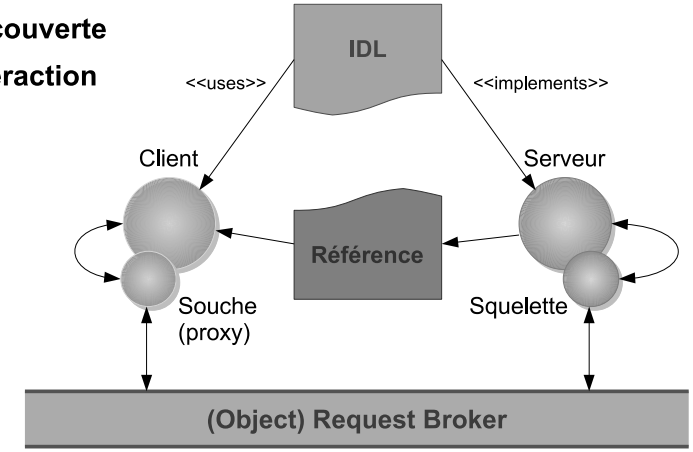
Aperçu du fonctionnement général

Chaque composant univoque ou intermédiaire dispose d'une référence (url ou autre) permettant de le localiser sur le réseau



Communication entre composants distribués

1. **Publication**
2. **Découverte**
3. **Interaction**



Principales architectures disponibles

- **CORBA :**
  - Proposée et maintenue par l'OMG :
    - En 1989
    - Consortium de plus de 800 membres (dont entreprises et universités)
    - Rôle de publication et promotion de CORBA
  - Indépendant des langages et plateformes cibles
  - Socle de conception des EJB et services web
- **SOA :**
  - Standards reprennent les concepts de CORBA 3
  - Couche intergiciel simplifiée grâce à l'utilisation de HTTP et XML
  - Évolutivité des formats d'échanges grâce à XML

→ **Architectures non propriétaires soutenues par des consortiums**

Solutions informatiques disponibles sur le marché

- **CORBA (GIOP) :**
  - Mises en œuvre payantes :
    - OOC Orbix et ORBacus
    - Inprise Visibroker
  - Solutions libres :
    - ORBacus (contexte universitaire)
    - JavaORB
    - JacORB
    - JavaIDL
    - MICO
    - TAO
- **Services web (XML – HTTP) :**
  - .NET
  - Java
  - Python
  - PHP
  - Ruby
  - ...

Mise en œuvre à l'aide de serveurs d'applications

Conclusion sur les systèmes distribués

- **Évolution du modèle trois-tiers**
- **Standards et normes (aujourd'hui) éprouvés**
- **Implémentations de qualité**

Les systèmes distribués ouvrent la voie aux applications pervasives et ubiquitaires de demain :

- **Disponibilité de l'information**
- **Agrégation de l'information**

Partie 2

La sécurité des systèmes d'information

- Chiffrement et signature électronique
- Modèle hiérarchique X.509
- Chaîne de confiance

La sécurité des systèmes d'information

La sécurité d'un système d'information est l'ensemble des procédures et moyens (physiques et logiques) permettant de :

- Protéger les accès aux machines
- Limiter les accès logiques aux services proposés
- Lutter contre les personnes malveillantes
- Prévenir et détecter les failles et dysfonctionnements du système

La sécurité est omniprésente dans la mesure où les applications et services d'aujourd'hui sont accessibles par le biais d'Internet

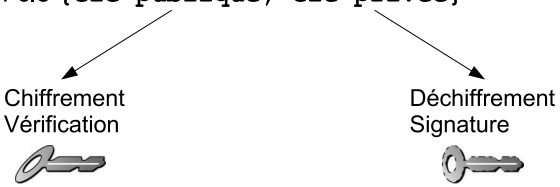
La cryptographie

Objectifs :

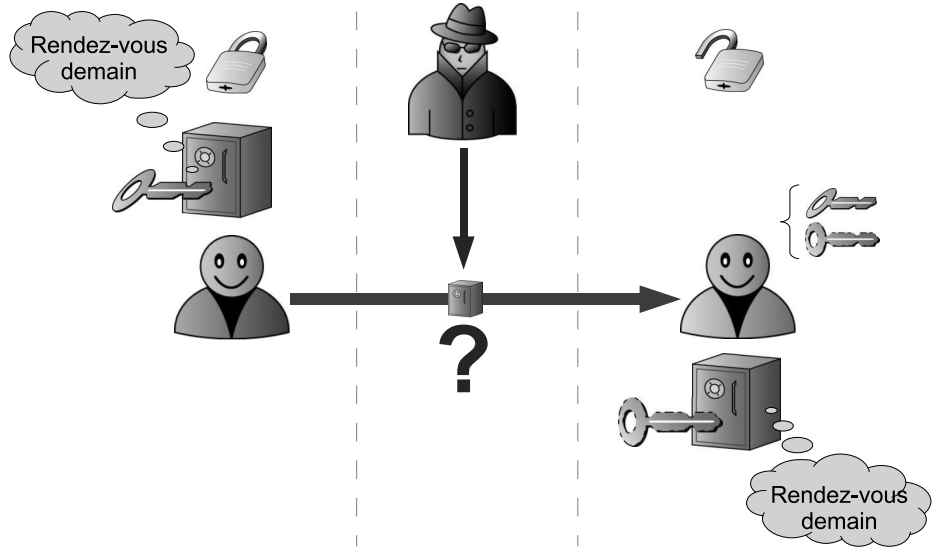
- Confidentialité : limiter la prise de connaissance des contenus
- Authentification : si employée au sein d'un protocole informatique
- Contrôle d'intégrité

Paradigmes :

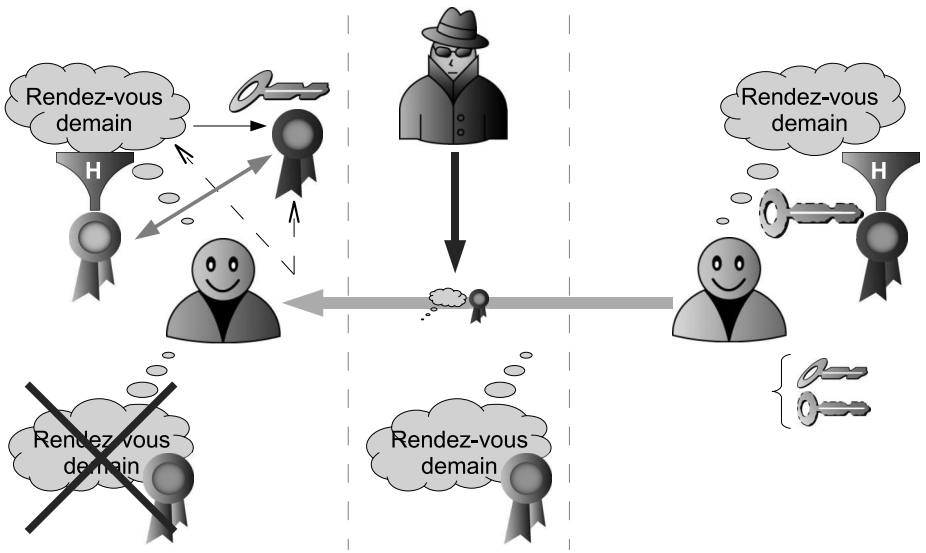
- Clé secrète : clé unique pour chiffrement et déchiffrement
- Clé publique : bi-clé `{clé publique, clé privée}`



Principe de chiffrement



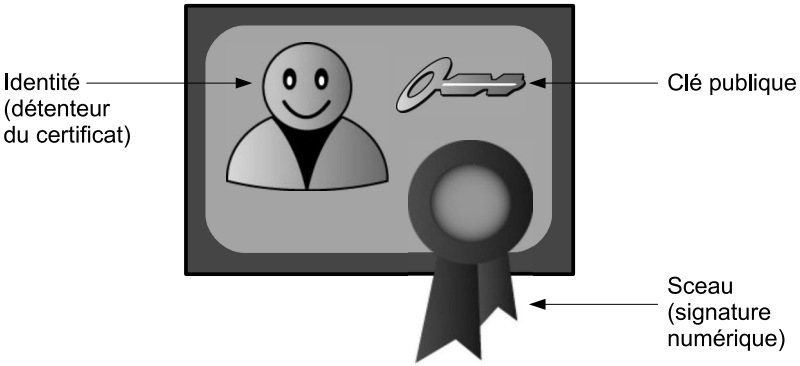
Principe de signature numérique



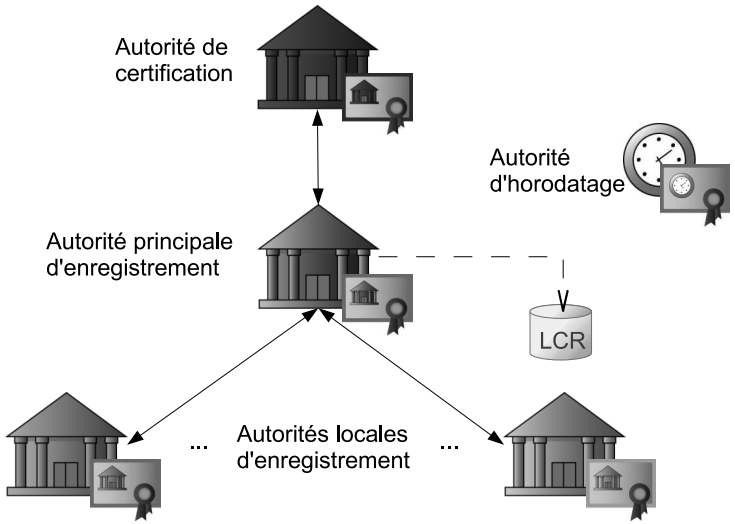
Certificats numériques au format X.509

Un certificat numérique apporte un lien entre une identité et une clé publique.

Le sceau numérique d'une tierce partie de confiance (PSCE) en garantit l'authenticité et l'intégrité



Modèle de confiance hiérarchique

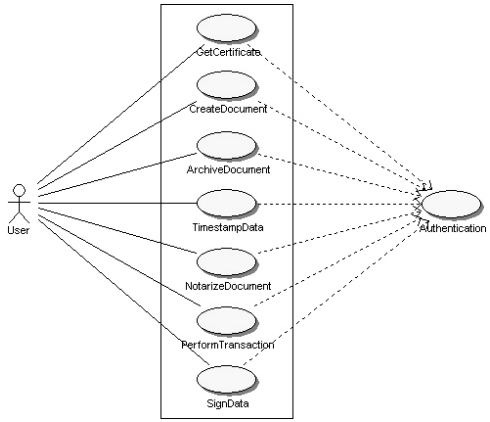


Nécessité d'une chaîne de confiance

- **Objectifs :**
  - Utilisation réelle de la cryptographie et de la signature numérique
  - Proposition de services avancés aux usagers (entreprises et particuliers) par des tiers de confiance (autorités)
- **Besoins :**
  - Cadre législatif : directive européenne, législations particulières, décrets
  - Professionnels des métiers liés à la sécurité :
    - PSCE
    - Archiveurs sécurisés
    - Horodateurs
    - ...
  - Procédures d'accréditation : évaluations architecturales et logicielles

Définition de la chaîne de confiance

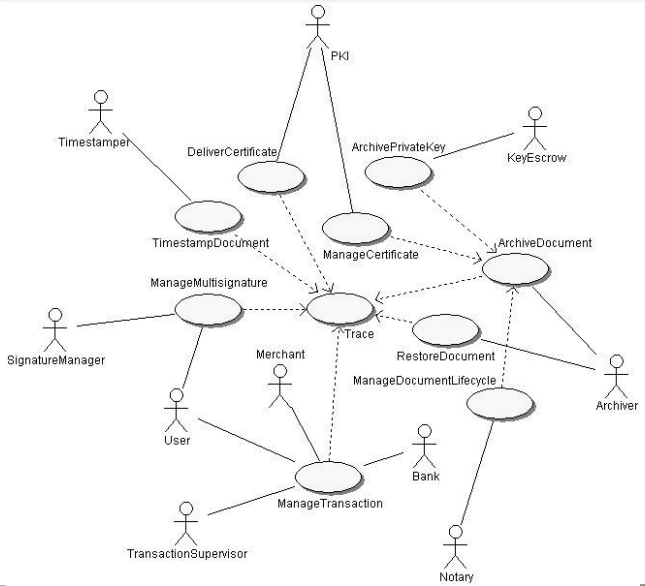
Une chaîne de confiance offre des services de haut niveau permettant un suivi depuis la génération des certificats par un PSCE jusqu'à la conservation sécurisée sur le long terme des documents ayant produit leurs effets juridiques (inactifs)



Acteurs de la chaîne de confiance

- **PSCE (CA – PKI) : met en œuvre une ICP**
- **Autorité de séquestre (key escrow)**
- **Autorité d'horodatage (timestamper), intégrée ou non à l'ICP**
- **Autorité d'archivage sécurisé (archiver)**
- **Notaire électronique (notary)**
- **Autorité de transactions électroniques sécurisées (transaction supervisor)**
- **Autorité de signature (signature manager)**

Services de la chaîne de confiance



Conclusion sur la sécurité

- **La sécurité est fondamentale**
- **Domaine d'expertise :**
  - Architectures des réseaux de communication
  - Développement de solutions logicielles (API)
  - Intérêt des logiciels libres (open source)
- **Complexité et sécurité sont antagonistes (B. Schneier)**
- **Son usage requiert un cadre législatif**

Conclusion générale

1. Systèmes distribués

2. Sécurité

Domaines complémentaires :

- ICP
- Chaîne de confiance

} Architectures distribuées

Nécessaires à la construction d'applications industrielles

Nombreuses recherches en cours (services web notamment)

Références utilisées

[GEI 01] J.-M. Geib, C. Gransart, P. Merle, *CORBA : des concepts à la pratique*, 2001

[DAN 00] J. Daniel, *Au cœur de CORBA avec Java*, Vuibert, 2000

[ORF 96] R. Orfali, D. Harkey, J. Edwards, *Objets répartis – Guide de survie*, Intern. Thomson Publication, 1996

[TAN 94] A. Tanenbaum, *Systèmes d'exploitation : systèmes centralisés et systèmes distribués*, Interéditions, Paris, 1994

www.omg.org www.oasis.org

www.w3.org www.ietf.org

Partie 3

Annexes

Global Inter-ORB Protocol

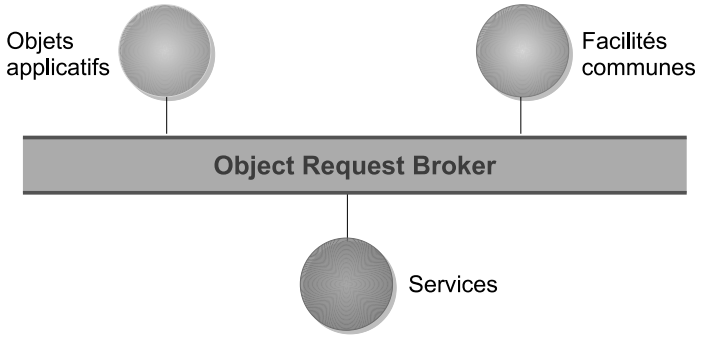
Spécification d'un protocole permettant l'interopérabilité des solutions mettant en œuvre CORBA 2

GIOP définit un format d'échanges commun en apportant un emballage normalisé permettant de :

- Transmettre les appels (invocations)
- Assurer des opérations communes aux mises en œuvre techniques :
  - Existence d'un serveur
  - Vérification de type (`is_a()`)
  - Récupération d'interface
  - ...
- Définir les opérations d'empaquetage (marshalling) et désempaquetage (unmarshalling)

Protocole IIOP dérivé de GIOP, utilisé pour les transmissions sur TCP/IP

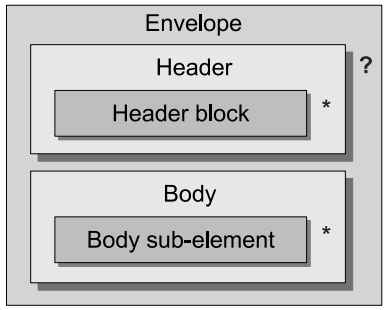
Object Management Architecture



Types de facilités communes :

- Horizontales : partagées entre différents domaines (gestion des impressions, etc.)
- Verticales : objets métiers

Simple Object Access Protocol : présentation du format



Simple Object Access Protocol : construction d'une requête

- **Requêtes HTTP POST**

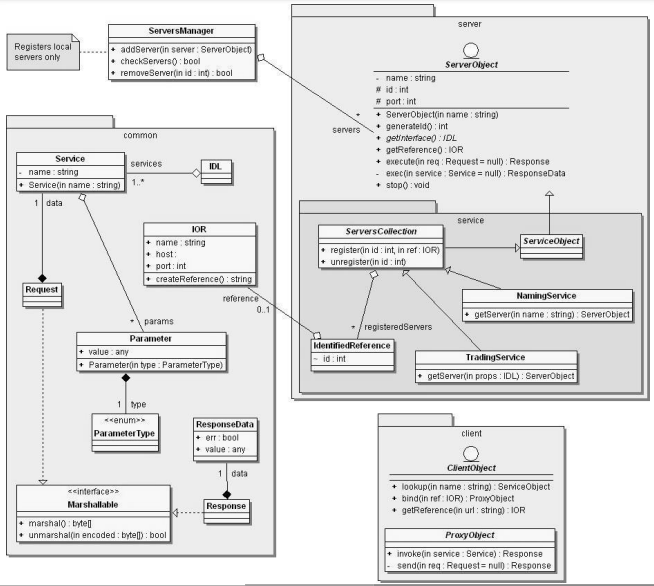
```
POST <url> HTTP/1.1
Host: www.myserver.com

SOAPMethodName: <uri:webServiceId#methodName>
Content-Type: text/xml
Content-Length: NNNN
```

- **Corps des requêtes (exemple SOAP-RPC)**

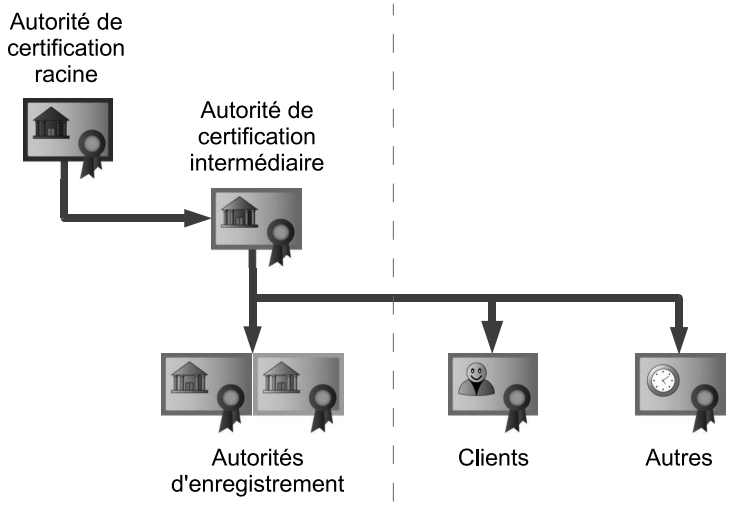
```
<?xml version="1.0"?>
<SOAP:Envelope xmlns:SOAP="<uri>">
  <SOAP:Body>
    <instance:methodName xmlns:instance="<uri>">
      <parameterName>Value</parameterName>
    </instance:methodName>
  </SOAP:Body>
</SOAP:Envelope>
```


Modélisation des composants des systèmes distribués CORBA (*)



(*) Il s'agit d'un exemple, non d'une référence

Chaîne de certification



Exemple d'architecture technique de mise en œuvre d'une IGC