Les réseaux de distribution de contenu (CDN)

Christophe Deleuze
ActiVia Networks
Christophe.Deleuze@activia.net

Séminaire ENST Paris

21 mai 2002

Introduction aux réseaux de contenu

Plan

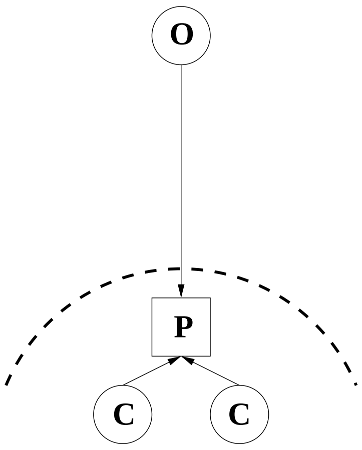
- introduction : les réseaux de contenu
- principe des CDN
- systèmes
  - ✓ *delivery*
  - ✓ distribution
  - ✓ redirection
  - ✓ comptabilité *(accounting)*
- autres approches

Les réseaux de contenu *(Content Networks)*

- **contenu numérique**
  - ✓ pages web (statiques, dynamiques) (HTTP)
  - ✓ *streaming* (RTSP)
  - ✓ à la demande
  - ✓ en direct
- **le problème de la distribution**
  - ✓ fournisseur du contenu = serveur origine
  - ✓ consommateur = client
  - ✓ point de rendez-vous

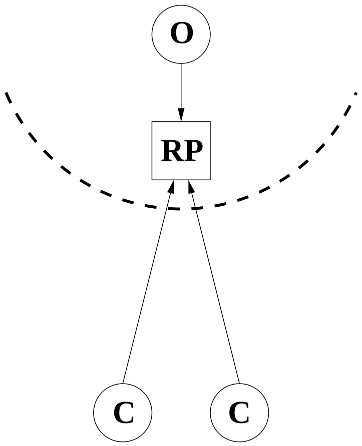
Foward Proxy Caching

- un *proxy* sert le contenu au client
- explicite/forcé/transparent
- géré par le FAI
  - ✓ + débit
  - ✓ + délai
  - ✓ + charge origine
  - ✓ – stats origine
  - ✓ – fraicheur



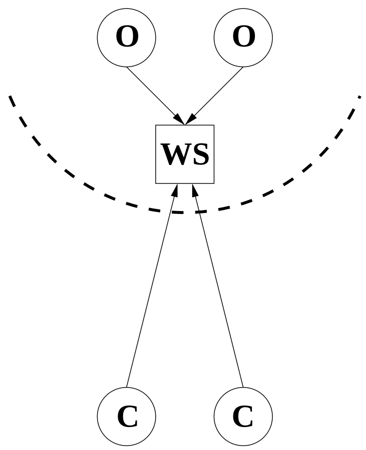
Accélérateur web *(reverse proxy)*

- origine caché derrière un proxy
- géré par le fournisseur de contenu
  - ✓ + charge origine
- variante : *web switch/server farm*



web switch/server farm

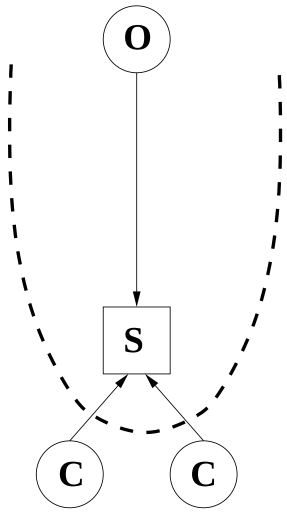
- *server farm* = 1 seule adresse IP
- *web switch* dispatche les requêtes sur les différents serveurs
  - ✓ charge
  - ✓ redondance



Content Delivery Network

surrogate
placé près des clients
géré par le fournisseur de contenu

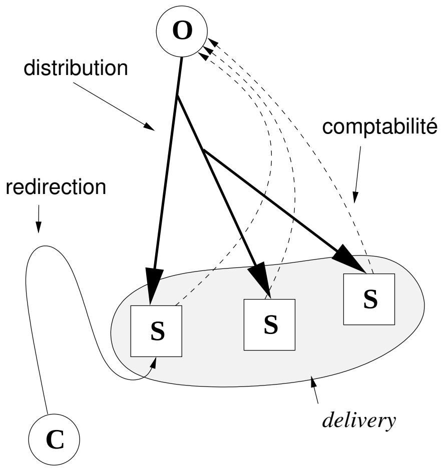
- + charge origine
- + débit réseau
- + délai client
- + stats origine
- + fraicheur



Les réseaux de distribution de contenu

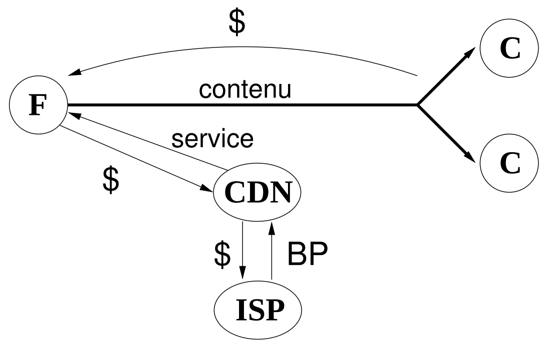
Qu'est-ce qu'un CDN ?

- rapprocher le contenu des clients
- quatre systèmes
  - ✓ *delivery*
    - ☞ serveurs *surrogates*
  - ✓ redirection
  - ✓ distribution
    - ☞ réplication
    - ☞ routage du *live*
  - ✓ comptabilité



Qu'est-ce qu'un CDN ?

- fournisseur de contenu
- consommateurs
- opérateur réseau
- opérateur CDN



Delivery system

- proxy caches
  - ✓ web (statique, dynamique)
  - ✓ *stream*
    - ☞ real networks
    - ☞ windows media
- machines spécialisées (hardware, OS)
  - ✓ Network Appliances (NetApp), CacheFlow, Inktomi
  - ✓ Cisco, Intel
- besoins spécifiques des CDN ?

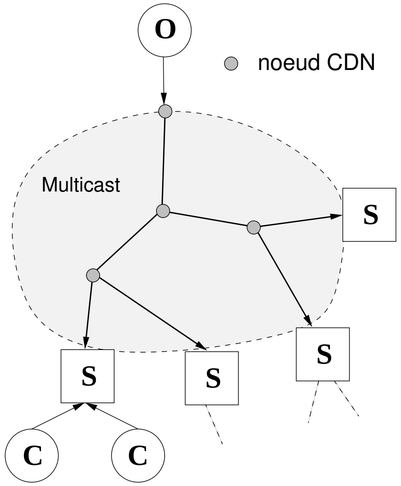
Comptabilité

- les logs sont répartis dans tous les *surrogates*
- besoin de générer une vision globale
  - ✓ pour le fournisseur de contenu
  - ✓ pour l'opérateur de CDN
- scalabilité ?
- structure hiérachique (multicast inverse)

Distribution : direct *(live)*

CDN ➠ réseau *overlay*

- multicast
  - ✓ soulage l'origine
  - ✓ soulage le réseau
- routage ad-hoc
  - ✓ ex min bandwidth
  - ✓ ...
- protection des flots
  - ✓ FEC
  - ✓ flots redondants



Distribution : réplication

- Réplication
  - ✓ copie
    - ☞ *push*
    - ☞ *pull*
  - ✓ synchronisation

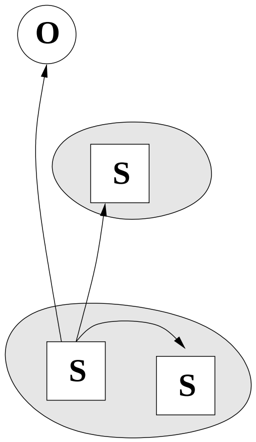
Push

- multicast
  - ✓ problème de la fiabilité
  - ✓ contrôle de flux
  - ✓ interface des *surrogates*

Réplication : copie

Pull

- où chercher le contenu ?
  - ✓ origine
  - ✓ voisins (ICP)
  - ✓ hiérarchie
- forme de routage
  - ✓ déterminer les routes possibles
  - ✓ choisir la meilleure (distance réseau + charge serveur)



Réplication : synchronisation

maintenir la cohérence des *surrogates* avec l'origine

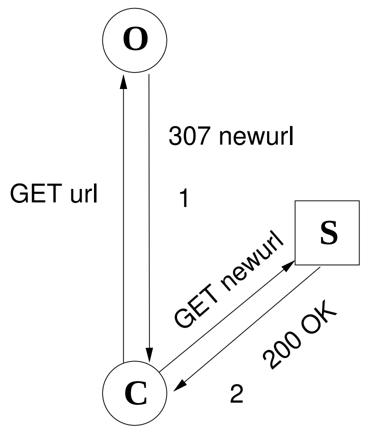
- *surrogate* = serveur authoritatif
  - ✓ invalidation de contenu
  - ✓ mises à jour
  - ✓ multicast
- *content signaling*
- tentatives actuelles (en sommeil)
  - ✓ WCIP *Web Cache Invalidation Protocol* (IETF/Cisco)
  - ✓ RUP *Resource Update Protocol* (IETF)

Redirection *(request routing)*

- trouver le "meilleur" surrogate (équilibrage de charge + distance au client)
- surrogate = f(consommateur, contenu, état réseau, états surrogates)
- routage de requête
  - ✓ routage (construction des tables)
  - ✓ relayage *(forwarding)*
- transparent pour le client...
  - ✓ différentes méthodes pour le relayage

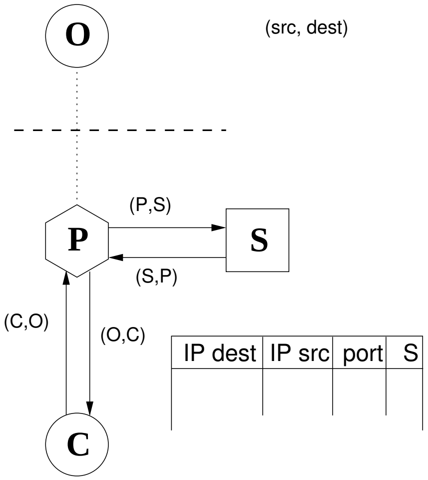
Redirection : *application redirect*

- ex HTTP : 307 temporary redirect
- + précis
  - ✓ adresse client
  - ✓ URL contenu
- – connexion vers l'origine
- – à implémenter pour chaque type de contenu
- – l'URL du document change !



Interception niveau transport

- switch L4 = proxy TCP transparent
- examen du premier paquet (`SYN`)
  - ✓ adresses IP (C,O)
  - ✓ port TCP
- triangulation possible



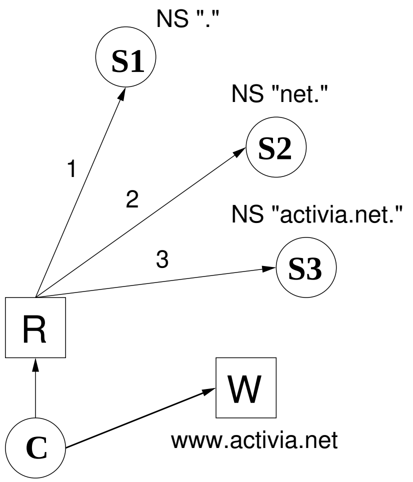
Interception niveau application

- URL
- en-têtes MIME

- proxy transparent *in-path*
  - ✓ utilisé dans les *Access Content Networks*
  - ✓ + rien à faire (f = cste = proxy)
  - ✓ – point de faiblesse pour **tout** le trafic
- proxy filtrant redirecteur
  - ✓ triangulation possible (mais délicat)

Rappels sur le DNS

- BD distribuée et redondante
- associations (nom, adr. IP)
- espace de noms hiérarchique
- serveur
- *resolver (cache server)*

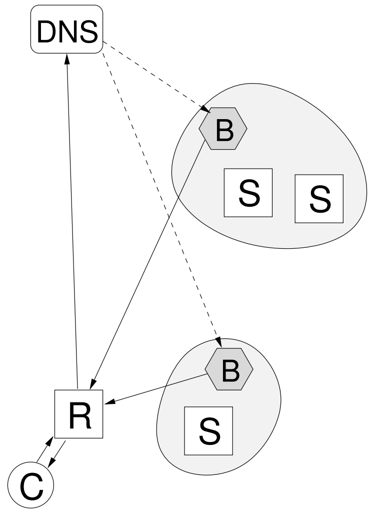


Redirection : DNS

- répondre selon le client
- + élégant
- + scalable, robuste (sécurisé ?)
- – précision
  - ✓ adresse du *resolver* du client
  - ✓ contenu : nom de domaine
    - ☞ `http://www.example.com/pic.gif`
    - ☞ possibilité de conventions (www.* = HTTP port 80)
    - ☞ réécriture des URL
      - ➠ `http://gif.example.com/pic.gif`
- – *DNS accelerator*

DNS boomerang

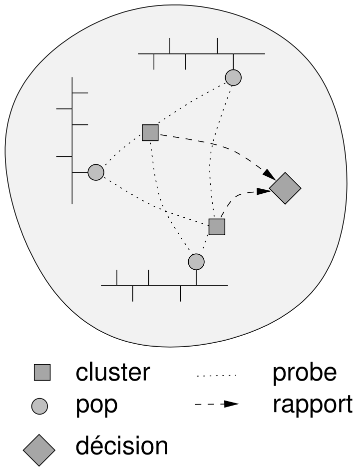
- Boomerang/Flash DNS/DNS flooding
- les clusters répondent
- proximité réseau
- si charge ➠ retarder la réponse
- routage très simple !



Redirection : routage

- requête *client* pour *document*
  - ✓ ensemble des Sgtes ayant le document
  - ✓ charge (CPU, nb. connexions...)
  - ✓ distance au client (hops, délai, BP ...)
- réponse rapide + infos à jour
  - ✓ méthode passive (IGP/BGP...)
  - ✓ méthode active *(network probing)*
    - ☞ à la demande
    - ☞ à l'avance
- Akamai ➠ *secret sauce*
- trouver un *surrogate* raisonnable

Redirection : routage : intra AS



- approche active/à l'avance
- décision centralisée
- clusters de *surrogates*
- pop (agrégats de clients)
- charge des services dans chaque cluster
- chaque cluster probe tous les pops
- association (pop, service) ➠ cluster
- associations pop ➠ { cidr } (ex. IGP)

Autres approches

Sites avec miroirs

- un site principal
- des sites miroirs
- synchronisation généralement périodique
- sélection du miroir + ou − manuelle
  - ✓ CPAN sauve le choix dans un cookie puis HTTP redirect

Réseaux peer-to-peer : Freenet

- réseau anti-censure
- documents identifiés par une "clé"
- requêtes passent de nœud en nœud
  - ✓ document caché sur tout le retour de la requête
  - ✓ gestion de cache LRU *(least recently used)*
    - ☞ réplication des documents populaires
    - ☞ déplacement des documents près de la demande
- uprizer.com

URN : Uniform Resource Name

URL espace de noms uniforme

suppose que ressource est disponible à **un** endroit **stable**

- réplication
- web éphémère *(transient)*

➠ espaces de noms alternatifs

➠ "hacks" de redirection CDN

URN id. persistent et indépendant de l'emplacement

résolveur d'URN ?

URI Uniform resource identifier = URL ou URN

Qu'est-ce qu'un CDN ?

Les CDN...

1. partagent dynamiquement l'infrastructure de réplication
2. fournissent un espace de noms adapté
3. utilisent les URL classiques

2 et 3 paraissent contradictoires

Un CDN est :

- une infrastructure de réplication
- un changement de sémantique de l'espace de noms URL

Perspectives

- URN
- *Application Distribution Networks*
  - ✓ IETF OPES *(Open Pluggable Edge Services)*
- CDN "ad-hoc"
- remise en cause du principe de bout en bout
  - ✓ *middleware* d'interception
  - ✓ *proxies* en tout genre

Références

- WREC *Web Replication and Caching* `www.wrec.org` (maintenant WEBI)
- WEBI *Web Intermediaries* (IETF)
- CDI *Content Distribution Internetworking* (IETF)
- OPES *Open Pluggable Edge Services* `www.ietf-opes.org` (IETF)
- `openp2p.com` O'Reilly network

[CMT01] I. Cooper, I. Melve et G. Tomlinson. *Internet Web Replication and Caching Taxonomy*. RFC 3040, IETF, janvier 2001.

[Hus99] Geoff Huston. Web caching. *Internet Protocol Journal*, vol. 2, n° 3, septembre 1999, pp. 2–20.

[STA01] Anees Shaikh, Renu Tewari et Mukesh Agrawal. On the effectiveness of DNS-based server selection. *In : Proceedings of Infocom2001*.

Fin

Document réalisé avec LaTeX.
Généré le 29 mai 2002.
Classe de document `foils`.
Dessins réalisés avec `xfig`.

Christophe Deleuze, `Christophe.Deleuze@activia.net`
`http://www.activia.net`

Ce document est disponible en postscript compressé avec gzip à
`http://rp.lip6.fr/~deleuze/activia.html`