

CDN INTERCONNECTION: BUSINESS GOALS AND SERVICE REQUIREMENTS

www.adolforosas.com

adolfomaria.rosagomez@telefonica.com

Madrid, April 2014.

INTRODUCTION

What does it mean 'connecting two (or more) CDNs'?

There could be many possible 'CDN-to-CDN connections'. Let us have from the beginning a wide definition of 'interconnection of CDNs':

<<Agreement between two businesses (CDNs) by which some responsibilities and activities of one party (CDN) are delegated to another party or parties and some compensation is agreed and received for this exchange>>

Technical means may be needed to implement the exchange of responsibilities/activities and to handle compensation.

Why connect two (or more) CDNs?

Two distinct CDNs are two separate businesses, two separate networks. The reason that stands up is 'connect to increase both businesses'. If we cannot find business advantages for both parties in the connection the effort does not make sense.

What business advantages are enabled by interconnection?

A CDN gets money from bulk-delivered traffic and/or per transaction. A CDN receives requests over a 'URL portfolio' and gives either bulk traffic or transactions in response. A CDN is designed to gather requests from anywhere in the world BUT to deliver traffic only to some footprint (part of the world). The only way to increase your business results is to increase income or cut costs (obvious). Increasing income can be done either by increasing delivery/transactions or by raising traffic prices or by raising transaction value (have VAS). Cutting cost is completely another tale that can be achieved through many technically intricate ways.

HIGH LEVEL BUSINESS GOALS FOR INTERCONNECTION

1) Increase delivery

More requests coming in are an opportunity for more business (more transactions and/or more traffic). The only action available to CDN admins that want to increase the number of incoming requests is to increase the URL portfolio: 'CDN A' will see more requests if 'CDN B' delegates a set of URLs to CDN A (even if it is done temporarily).

(NOTE: End user demand is unpredictable. Total sum of requests over current URL portfolio may increase without actually increasing the size of the portfolio<number of URLs>, just imagine that each URL in our current portfolio receives some more requests, but that increase happens as the result to an action of the end users not the CDN).

But, why would CDN B delegate part of its precious portfolio to CDN A?

Resources are limited. More processing/delivery resources allow for more business. A CDN can never know nor be in control of how many requests come in, so it is possible that some requests coming into CDN B cannot be attended. In that case CDN B could 'overflow' some requests to CDN A thus retaining an otherwise lost profit.

1.1) Temporary delegation of portfolio. (impulsive overflow)

Maybe CDN B just do not have enough machines (limited processing and limited delivery) and 'success took them by surprise', in that case this is a pure-temporary-overflow and may be handed to another CDN A that operates in the same footprint. Over time CDN B will adjust capacity and will stop overflowing to CDN A as it is usually more profitable to use your own capacity and retain your whole portfolio of URLs and Clients. The handover in and out must be fast. It is important to be able to trigger overflow based upon some real world variables that are inspected in real time. Trigger action is real time, but all the agreements needed for this to happen are negotiated in advance and the limits and conditions are set in advance.

1.2) Long term delegation of portfolio. (delegation or 'traffic termination')

Maybe CDN B is not deployed in some footprint so it is suboptimal (not profitable) for CDNB to deliver traffic/transactions to that footprint. In this case CDN B needs a long-term-delegation of some URLs delivery to CDN A for a specific footprint. This is a case of 'keeping growth without increasing cost'.

2) Adjust prices

2.1) Mutual Overflow in the same footprint (Hidden balancing)

Competing CDNs that operate in the same footprint usually force a trend of diminishing prices, trying to capture clients. Two-way Interconnection at low transit prices may have the effect of presenting to the client a stable price in the area (footprint). Any CDN in the area may benefit from a new client of another CDN in case a Mutual Overflow is agreed at a reasonable price. This sounds much more reasonable than trying to compete in quality in the same footprint, as the biggest contributor to quality is the performance of the underlying carrier and that is something that all CDNs in the area can equally get. The mutual overflow means that under some conditions evaluated in real time a URL normally served by CDN A will be served by CDN B in the same footprint and all the way round. This mutual overflow can be thought of as 'Hidden Balancing' between CDNs, as it is a mechanism transparent to the CDN clients.

2.2) Balanced CDNs in the same footprint (Explicit balancing)

Two or more CDNs may go to market together through a CDN balancer. In fact many balancers are now in place built by CDN clients or by third parties. The business that comes out of the balancer works on the fact that in a big area (multi country) and through a long time (up to a year) the performance of any CDN is going to vary unexpectedly due to these factors:

- unknown expected behavior of demand over own URL portfolio
- unknown evolution of own client portfolio
- unknown instant performance of underlying transport networks (carriers)

A balancer will take a request and will route it to the best suited CDN in real time.

Opposed to 'Mutual Overflow'== 'Hidden Balancing' this can be called 'Explicit Balancing' as the mechanism is now visible to CDN clients. The reasoning for 'best CDN' will be complex, based on the real time status of involved CDNs, but also based on 'fairness for business' of all involved CDNs in case the balancer is controlled by all of these CDNs. *(In case the balancer is property of a third party fairness for all involved CDNs is not guaranteed.)*

Many CDN clients feel better when they know that their portfolio has more than one CDN ready for delivery. In that case it is the best option for the CDNs to work on their mutual balancing. In case a third party balances them the results will not be so good, and some money will go to the third party for joining the CDNs. It is better to identify CDNs that may complement our CDN and work together in a balancer that could be sold together and directly to clients.

3) Balance costs vs. income: increase occupation

Planning cost in a CDN is nothing straightforward. Agreements with Carriers cost money (CDNs have to pay for traffic and sometimes pay for dedicated links /ports or housing). Cost of traffic is directly linked to income, but cost of ports, links and housing are fixed costs not related to the amount of activity or the success of the service. Machinery for delivery costs money (edge), but maintenance of machinery: installation and operation may be the most costly part of a CDN.

In case a CDN is not able to maintain a high occupation, the fixed costs will make the business not worth the effort, thus it is a good idea to offer capacity to other CDNs either as overflow, or through a balancer or as traffic termination. The real-time measurement of available capacity in our CDN may be an input to the balancer/overflow/delegation.

SERVICE REQUIREMENTS THAT EMANATE FROM BUSINESS GOALS

(Conventions on language: **MUST (COMPULSORY)**> **DESIRABLE (GOOD)**> **COULD (OPTIONAL)**)

High-level CDN-Interconn Service goals:

- .Temporary Delegation of some part of portfolio (Impulsive Overflow)
- .Long Term Delegation of some part of portfolio (delegation or Traffic Termination)
- .Mutual Overflow (Hidden Balancing)
- .Explicit balancing

Requirements to implement Long Term Delegation one-way (receiving) in CDN 'X':

1. 'X' must have the ability to receive a delegation of some URL set from another CDN, intended ONLY for a given footprint.
2. Metadata must be given to the receiving CDN ('X') identifying the account in 'X' that is the owner of the URL set prior to handling any transaction on behalf of this account. (Delegation Metadata).
3. Metadata must be given to the receiving CDN ('X') containing any configuration data needed to perform the transactions on behalf of the donating CDN. Examples of transactions are: geo-targeting, IP blacklisting, geo-blocking, security token verification, etc.... (These metadata may apply for the whole account, and in that case they are part of

Delegation Metadata or they may apply to a URL or to a URL set, in that case we call them 'URL-service Metadata'.)

4. Creation of the client account in 'X' (to handle delegated transactions) could be done 'on the fly' on receiving the URLs + metadata (client auto-provisioning) or could be done in advance by an admin process in 'X' (client pre-provisioning).
5. Long Term Delegation must be 'actionable' immediately (for instance at the request from 'X') and also it must be possible to 'schedule' activation/termination planned ahead by both CDNs administrators.
6. Long Term Delegation must start to be effective 'fast', could have a defined termination date and must stop 'fast' either immediately (by admin action) or at the scheduled date. Here, in the context of 'X', 'fast' means as fast as it is convenient (SLA) for the donating CDN. (Usually this 'fast' will be in the range of minutes.)
7. Delegation must carry with it a feedback channel so the donating CDN (the one that 'gives URLs') regularly receives details of the delivery/transactions performed by the receiving CDN (the one that 'takes URLs'). This feedback as the very minimum must contain the full technical records generated at the edge of receiving CDN (this is commonplace in CDN business).
8. It is desirable that the receiving CDN ('X') builds 'Analytics' specific to delegated traffic, thus offering info about the extra business that comes to it through compensation. In absence of specific arrangements the Analytics and Mediation (Billing) Services in 'X' will create graphs and reports of the delegated traffic as for any other account so delegated traffic is not distinguishable 'per se'. For this reason it is desirable to mark delegation accounts so we can analyze separately traffic due to delegations.

Requirements to implement unrestricted (two-way) Long term Delegation:

1. Feedback data could transparently update any Analytics & Mediation (billing) service that the donating CDN may have. Records of deliveries/transactions that have been delegated must appear mixed and added with all the regular records of the donating CDN.
2. Records of deliveries/transactions that have been delegated could be separated from all the regular records of the donating CDN (in a view different from the mixed view), as an additional action that tries to give more information to the business. This information serves to plan capacity increases.
3. A CDN admin could have the ability to select a subset of the own URL portfolio and delegate it to another CDN ONLY for a given footprint. (Implementation of delegation 'the other way round': not receiving but exporting URLs.

Requirements to implement Temporary Delegation:

1. Temporary delegation (impulsive overflow) must be transparent to clients.
2. Temporary delegation (impulsive overflow) must be transparent to the exploitation systems of the donating CDN. Consumption records must be accounted as 'normal' deliveries/transactions.
3. Temporary delegation (impulsive overflow) must be 'triggered' by rules based on real time inspection of several variables.
4. Variables that trigger temporary delegation (impulsive overflow) must be defined by the CDN business management.

Requirements to implement Mutual overflow:

1. A CDN admin must have the ability to select a subset of the own URL portfolio and add it to a mutual balancing set with another CDN ONLY for a given footprint. All URLs in the mutual balancing set must be served to end users by both CDNs. Technical devices and rules used to balance must be set up by both CDNs.

Requirements to implement Explicit Balancing:

1. A CDN must have a tool for clients: an explicit balancer. The balancer must act as the highest level request router for all involved CDNs, affecting subsets of each CDN portfolio, and applying to a specific footprint for each URL in the sum of portfolios.
2. The explicit balancer must have public (known to anyone including clients) balancing rules, public input variables and public policies.
3. The explicit balancer must account for every transaction/delivery and offer Analytics that allow analyzing behavior of the balancer and fine tune balancing rules and giving feedback for pricing models of the balanced CDNs product.