

Lies, Damned Lies...

The unpredictable world of online marketing & web analytics

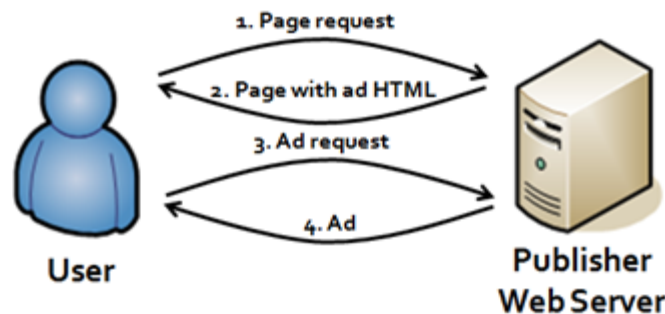
June 20, 2008

Online Advertising Business 101, Part II - How does adserving actually work?

Welcome to the second part of my [Online Advertising 101](#) series. One of the things that I've discovered people fail to appreciate about online advertising is how much goes on behind the scenes in order to bring you an innocuous-looking banner ad. Whilst this is a relatively technical topic to cover in a series which is supposed to deliver insights about how the industry works as a whole, I think it's instructive to consider it because the technical jiggery-pokery that goes on gives rise to many of the issues that define the industry (for example, security and behavioral targeting). So here goes.

Back in the olden days...

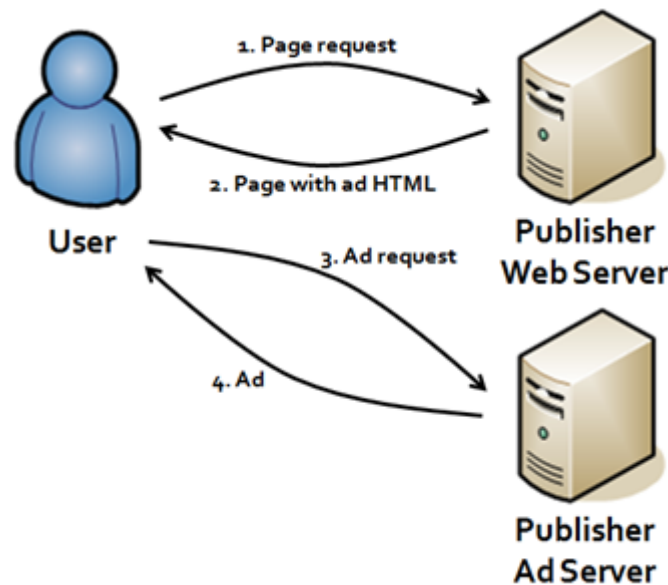
Once upon a time, when the online advertising industry was in its infancy (all of about 11 years ago), if you wanted to place an ad on a website, you had to call the publisher personally and broker your own deal, and then send the publisher the ad you wanted them to place on their site. The publisher would, essentially, include the ad as part of their site design ([example](#)), and, once it was live, users requesting the page(s) where the ad was referenced would see the ad, like any other graphic item on the page:



I've shown this basic scenario as four interactions, because the user first requests the page the ad is on, and then the HTML in the page tells the user's browser to fetch the ad, which is also on the publisher's web server.

Enter the publisher ad server

It didn't take long for publishers to tire of this arrangement - it meant that whenever the publisher wanted to change an ad, they had to make a change to their live website, which was time consuming and risky. So publishers started to put their ads onto a separate server - the publisher ad server - which would act as a repository for the ads.



Over time, publisher ad servers have evolved into sophisticated beasts, providing a wide range of functionality that helps the publisher manage and predict their ad inventory, handle billing to advertisers, and target ads.

But the most fundamental value that an ad server brings for a publisher is the ability to rotate multiple ads through a single ad unit (that is, a slot on a page for an ad). So if a publisher has 100,000 impressions on their home page a day, and they have a banner ad at the top of that page, they can sell 25,000 impressions to each of four advertisers, load the ads into the ad server, and let the ad server do the work of ensuring that each advertiser's ad is shown the right number of times.

Examples of publisher ad servers are [DoubleClick's DART for Publishers \(DFP\)](#), [Atlas AdManager](#), [24/7 RealMedia's OpenAdstream](#), [Yahoo's AMP](#) and [Google AdManager](#). Plus, there are a lot of small players who offer simpler systems which really just handle ad rotation, for small publishers.

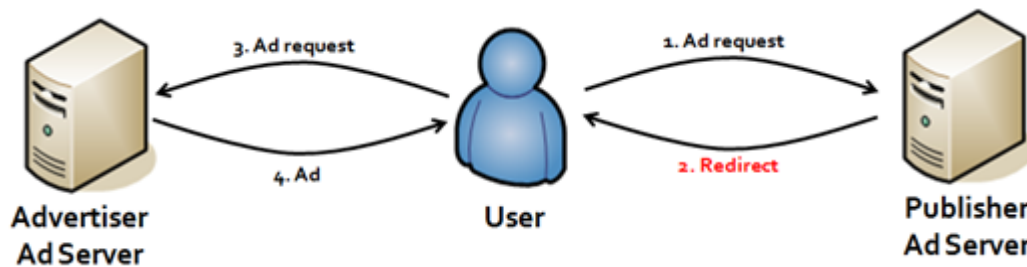
In days of yore, all four ads in the above example would be loaded onto the publisher ad server (in fact, this is still the case for premium advertising on very popular sites like MSN.com, partly to enable advertising that is very highly customized for the target site). But this arrangement quickly proved inefficient for advertisers, who would have to send their ads to each publisher they wanted to advertise with; and would then have to re-send those ads when they changed, etc.

In the above model, the advertiser also **lacks the ability to track the performance of their ads across multiple publishers**, and is **completely dependent on the publisher** for data on how many times their ads were served (which still remains the primary way of calculating the price of a campaign); and they have **no means of changing the delivery rules** for a campaign (for example, by increasing the frequency of one ad creative, whilst decreasing another), except by calling the publishers and asking them individually.

Enter the advertiser (third-party) ad server

Advertiser ad servers are often referred to as third-party ad servers because the advertiser's ad server is a 'third party' to the ad transaction (the publisher ad server being the 'first party'). I'm going to refer to this server as the advertiser ad server in this post to save you the trouble of trying to remember who is the first and third party in this world.

When the advertiser ad server enters the picture, the ad request is not fulfilled directly by the publisher's ad server. Instead, the publisher ad server responds to the ad request with a redirect, telling the browser to fetch the content it's asked for somewhere else - in this case, the advertiser's ad server.



Note in the above diagram that I've removed the original call to the publisher's website; take that as a given. So the publisher ad server receives the ad request, and passes it on to the advertiser ad server.

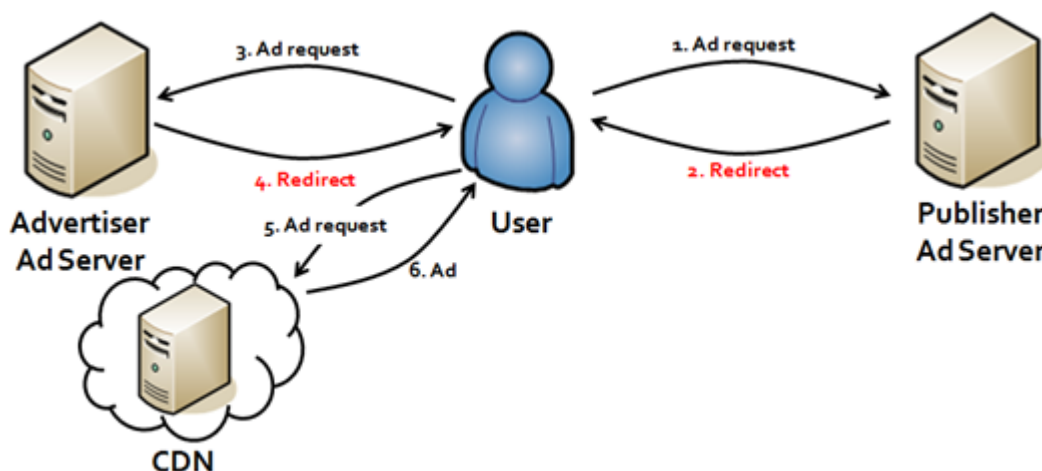
Advertiser ad servers have also become very sophisticated over the years, but in the context of ad delivery, they perform just a couple of major functions:

- They manage which actual ad for a given campaign is delivered when an ad is asked for, implementing rules like **frequency capping, ad rotation, and targeting**
- They help with **managing the actual ad creatives** (the GIFs or Flash files) which constitute the ads themselves
- They **track ad delivery across all the sites** that are **involved in a particular campaign**, measuring the number of times ads were served, the number of times they were clicked, and (through instrumentation of the advertiser's site) the number of conversions generated by those clicks.

The benefit to using their own ad server to an advertiser is that they only have to upload their creative to one place, and they can plan and execute a campaign across multiple publishers easily (at least in theory - in practice, it's not that simple, but we'll come back to that point later).

The third-party advertiser ad server market is dominated by a couple of players: Microsoft/Atlas (with Media Console) and Google/DoubleClick (with DART for Advertisers, or DFA). But there is a host of smaller providers, particularly specializing in Rich Media (interactive Flash and Video) ad serving, which also play this role in the ad serving process. And there are a number of ad servers which are really front-ends for ad networks. But I'll cover both of these cases in a future post.

Ironically, one of the things that ad servers tend not to do these days is actually serve the ad - this grunt-work task is usually farmed out to a Content Delivery Network (CDN), such as Akamai, which maintains thousands of servers across the Internet to serve ads (and other content) on behalf of other people, as quickly as possible. So when a CDN is involved, the advertiser ad server simply returns yet another redirect, telling the browser where the content really (really, really) is.



Sheesh! Why so complicated?

You might still be wondering why there is all this complexity just to serve a single ad. Well, the answer comes when you

think about the above pictures when you have a hundred advertisers, and a hundred publishers. The publisher and advertiser ad servers enable the one-to-many relationship that publishers and advertisers need to maintain. Or, put another way:

- The publisher's ad server helps the publisher manage their inventory across many advertisers
- The advertiser's ad server helps the advertiser manage their media buying across many publishers

I'm afraid to say, however, that the picture I've painted above represents a fairly simplistic view of the mechanics of ad serving. The picture becomes more complicated when you include ad networks, rich media, tracking & targeting, or ad exchanges in the mix. But you've probably read enough for today, so I'll pick up these topics in subsequent installments.

Feel free to add your comments/questions in the comments box.


[Online Advertising Business 101 - Index of all posts](#)

June 20, 2008 in [Ad industry 101](#), [Adserving](#) | [Permalink](#)

 [del.icio.us](#)

 [Digg](#)

 [Reddit](#)

 [StumbleUpon](#)

 [Facebook](#)

 [Live Favorites](#)

TrackBack

TrackBack URL for this entry:

<http://www.typepad.com/services/trackback/6a00d8341bf6d253ef00e5536379658833>

Listed below are links to weblogs that reference [Online Advertising Business 101, Part II - How does adserving actually work?:](#)

Comments



kj said...

This series of posts is quite helpful. I think the most helpful are those that you are yet to publish. Please keep them coming (and in as much gory detail as you can manage)! I eagerly await the next post.

[Reply June 20, 2008 at 01:02 PM](#)



[SEM Pune](#) said...

Quite a content...although i didn't get to go through the first part i am happy i could go through this one.

[Reply June 26, 2008 at 12:58 AM](#)



[Stephen Tompkins](#) said...

Hi -

Great post, very insightful. I just have one quick questions. Why is the person in the last two diagrams in between the #2 redirect and the #3 ad request? Doesn't this usually go on behind the scenes independent of the user's actions or do cookies come into play and that is where the user fits in?

[Reply July 02, 2008 at 12:13 PM](#)



[Brad](#) said...

Thanks for this overview. I'm a bit new to the online publishing business, so I have a question about ad-serving

technology that I hope you can help me with. My company sells display advertising on our own sites. Deals are typically for a specified number of impressions at a certain CPM. CTR does not determine how much revenue we make in the short-term, but it has a big effect on whether advertisers feel like they got their money's worth, and whether they keep buying ads from us. We've dabbled with manually targeting ads against specific pages that have related content and seen CTR skyrocket – as much as 10x higher than average. However, manually targeting like this is not scalable. My question is this: Can any ad servers *learn* which characteristics (page URL, page content, visitor data, time of day, etc.) positively correlate with the CTR of an ad, and then attempt to serve all the ads in such a way as to maximize the overall CTR? All the ads would have to receive their contracted number of impressions, but they would be served in such a way that average CTR is maximized. Thanks for any help you can offer.

Reply July 18, 2008 at 02:58 PM



ian_d_thomas said...

Stephen:

Well, the thing is, any 302 redirect technically bounces back to the browser - the server returns the 302 response (along with a cookie, if desired), and the browser goes "Ok, I'll check there". A lot of "how ad calls work" diagrams kind of show a chain passing from the browser through the intermediaries that are serving the redirects, but I actually think that's a little misleading, since the ad (when it is finally served) doesn't pass through all the intermediaries; it goes straight from the final server back to the browser.

Brad:

You ask a very perceptive question. This kind of 'yield optimization' is top of every publisher's mind, as you would expect. I don't know of any ad servers which have the functionality that you describe. It's actually a very difficult math problem to solve, since on the one hand CTR depends on so many things, and on the other hand the publisher is looking to meet certain guaranteed impression targets. The only place where this problem is in the process of being solved is in non-guaranteed inventory, in the form of multi-network optimization. Here, companies like [The Rubicon Project](#) and [PubMatic](#) are looking to optimize publisher remnant inventory by delivering it to the network which generates the best return for the publisher, on an impression-by-impression basis. I think we'll have to wait a little longer for a guaranteed-inventory solution.

Reply July 28, 2008 at 11:19 AM



Fady said...

Hi,

Great informative articles. Will be looking forward to your next one. I was wondering if you can recommend any books that explain to a beginner (preferably as well as you do) the concepts of online advertising in general, or mobile advertising in particular. Any references would be greatly appreciated.

Thanks, and keep up the great work.

Reply August 12, 2008 at 08:46 AM



Sidney Song said...

Just want to say that your posts on the industry are great! What I thought before was totally wrong. I just thought all the ads should be hosted in publishers! Now, fairly clear to me.

Thanks a lot.

Reply September 04, 2008 at 11:48 PM



ahuang said...

Good article!

Our website actually working with two ad networks for the same inventory. Depending on which partner has higher estimated yield, we selectively make ad call to one partner. But recently we found there is a problem of ad coverage (ad return/ad call), so we are wondering if we can make multiple ad calls, say, making ad call to one partner, if returned nothing, then make another ad call. (The potential problem is latency, makes the ad loading slow)

Can I know if it's feasible? do you know any companies that can make this happen?

Thanks!

Reply September 24, 2008 at 02:55 PM

Comments on this post are closed.

POWERED BY  TypePad