

# HEADER BIDDING MYTHS: CLAIMS VS REALITY

Examining the Hype Surrounding Header Bidding



# INTRO

Header bidding processes are supposed to be asynchronous, but are they really? In the real world, where a User's device, browser, and network are often overwhelmed, not everything is as it seems.

Header bidding has brought us a long way, but modern implementations have become bloated and rife with hazards that can severely limit ad revenue.

The good news is, you can address these inefficiencies and release untapped revenue.



# HERE'S WHAT YOU NEED TO KNOW

Everyone knows pages that load fast rank better and provide a better user experience. Developers spend a great deal of time improving the efficiency of publisher content, but many overlook the significant impact of addressing inefficiencies in the ad stack.

Things can get technical and overwhelming pretty quickly, and it's easy to assume that established technologies have all of the essential bases covered.

But what if the most common approaches are falling short on their claims? What areas should Publishers focus on first?



# THE PERFECT NUMBER OF BIDDERS

You've tackled that ambiguous task of adding "just the right number" of bidders after sifting through vague industry recommendations such as "6 to 10," or "no more than 7," or "less than 15"—all with the hope of not adding "too much" latency.

Many variables are involved, but most advice rests on the assumption that latency is only a problem when it becomes noticeable or passes some well-defined threshold.

In reality, every bidder that gets added to a client-side setup taxes the resources of a visitor's device, slows page content from loading, and increases the potential of other bidder opportunities being "blocked."

**TIP**

**Always work with as few bidders as possible while still monetizing all inventory.**

Following the 80/20 rule, identify the five bidders (aside from Google) responsible for most of your ad revenue, then limit your stack to those five. Monitor performance closely, then rotate out the lowest-performing bidder once a month and try a new bidder.

# BIDDER PARTICIPATION

In the world of programmatic auctions, buyers sometimes choose not to return a bid. There are valid reasons for this, such as not having appropriate demand available, but this scenario should not be confused with buyers not being given the opportunity to participate.

The concept of an even playing field is widely touted, but in client-side header bidding, whether you have five bidders or 15 bidders, there is a high chance not all will experience an equal opportunity to participate.

In real-world scenarios, many of the client-side bidders who Publishers assume are participating never see the opportunity, at least not every time.

Adding bidders that rarely participate is like adding cylinders that don't fire to a car's engine! Five operating cylinders will always outperform ten if half of the ten are not contributing.

Load all bidders at the same time and as early as possible.

## Use a tool like the "Headerbid Expert" browser extension:

Look for bidders that consistently take too long or fail to return a result. If you've done what you can to request a bid from them as early as possible, decide whether the value they bring justifies their place among your five bidders. If not, swap them out and test another.

Bidders who consistently make the ad server wait will negatively impact page performance and overall revenue.



# LATENCY

Latency is the silent attacker of revenue, user experience, and site ranking. The term is used broadly in our industry and generally understood to mean added time, but is rarely defined with any real context. Latency, or "delays" exist across many aspects of page loading and ad serving. A certain amount is unavoidable, but far more latency than most publishers realize exists in areas that can be addressed. Latency is the enemy, the villain in your story.

## TIPS FOR LOADING PUBLISHER CONTENT

- ✔ Combine similar CSS files so the browser doesn't need to send separate requests for every file.
- ✔ Limit the number of files like images, videos, and other media where possible.
- ✔ Reduce the number of external scripts where possible by assessing which assets are not necessary on the page or by loading low priority assets later.
- ✔ Use browser caching so returning users don't have to load the entire page from the server on every visit.

- ✔ Use modern image formats such as WebP.
- ✔ Consider using a Content Delivery Network for a faster transfer of data.
- ✔ Utilize Minification or GZIP compression to reduce the size of JavaScript, CSS, and HTML files.
- ✔ Implement asynchronous page loading, so server requests are not blocking each other.
- ✔ Implement "lazy loading," where possible.
- ✔ Ensure the use of HTTP/2 Protocols.





# TIPS FOR LOADING ADS

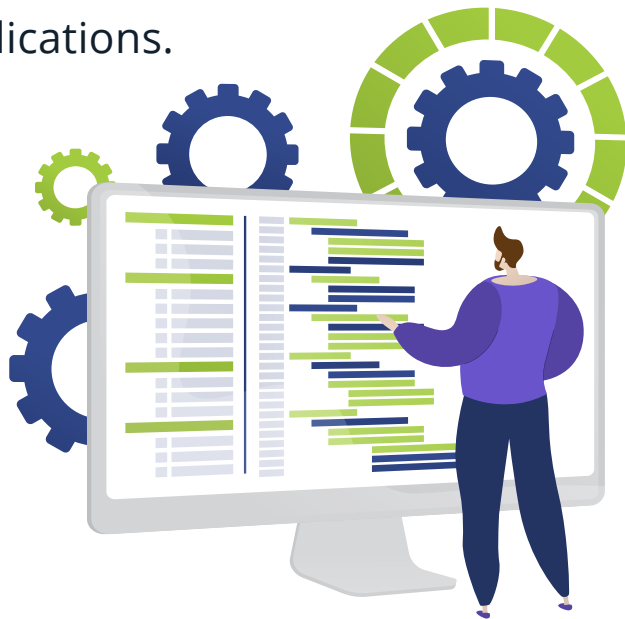
To improve viewability, implement lazy loading with a render distance of one viewport or less for desktop and two viewports or less for mobile.

## If utilizing client-side header bidding:

- ✔ Follow the Bidder selection tips above while being vigilant about identifying and removing underperforming partners.
- ✔ Work toward bringing the number of demand partners as close to five as possible while closely watching for decreases in bid pressure.
- ✔ If eCPM lowers notably, you likely need to address other areas before removing further demand. However, it is important to note that a slight decrease in eCPM during these changes should not be a show stopper. A slight decrease in eCPM short term, while seeing improvements to page speed, UX, and viewability, will often lead to higher bids long-term. Focusing too much on eCPM without considering other important elements leads to a decrease in long-term revenue.

## Move to a server-side (S2S) header bidding configuration:

- ✔ S2S allows page content to load independent of the ad auction. More than half of the network requests and processing tasks on a typical site are ad-related, affecting the loading of page content considerably.
- ✔ When the required complex and resource-intensive processes for auctions and ad serving move from the User's browser to highly optimized servers, the visitor's device can focus on loading publisher content. Offloading this burden onto servers typically results in publisher content loading 50% - 100% faster.
- ✔ Letting servers manage the heavy lifting aspects of ad serving also allows for auctions to start sooner and process much faster.
- ✔ Far more demand partners can be included in every auction while decreasing the errors and lost opportunities inherent to client-side applications.



## EQUAL PLAYING FIELD?

Tests show the last bidders called during an auction are less likely to be successful in submitting a bid before the timeout. While the bidders called first typically return the highest bids. This disparity grows as more demand partners get added, which undermines the core purpose of header bidding.

Popular wrappers attempt to address this issue by randomizing the order in which bidders are called, but randomizing which demand partners miss the opportunity is far from an efficient solution.

Only server-side solutions can offer a genuinely equal playing field to multiple bidders.

### EXAMPLE

The Datablocks platform can efficiently manage up to 75 external bidders, plus thousands of direct sales bidders in every auction without missing any opportunity.