SITE TAGGING BEST PRACTICES

VERSION 1.0

Published February 4, 2013

This document has been developed by the IAB Data Council

The Site Tagging Best Practices (SiteTaggingBP_v1.0) best practice was created by a taskforce of volunteers from 11 companies.

The Site Tagging Best Practices Taskforce was led by:

- Todd Chu, BrightTag
- Maggie Neuwald, TagMan

The following companies contributed to this document:

AMC Networks	Evidon	Mirror Image Internet
BlueKai	Google & YouTube	TagMan
BrightTag	Krux Digital	Yahoo!
Catchpoint	Microsoft, Atlas Advertiser	

The IAB lead on this initiative was Jessica Anderson

Contact jessica.anderson@iab.net to comment on this document.

ABOUT THE IAB'S DATA COUNCIL

The IAB Data Council is dedicated to demystifying data usage and control in the interactive advertising marketplace. The IAB Data Council objective is to enable revenue growth through the establishment of quality, transparency, accountability, and consumer protection in data usage. A full list of Council member companies can be found at: http://www.iab.net/data_council

This document is on the IAB website at: http://www.iab.net/sitetagging

Table of Contents

Executive Summary

Since the inception of digital advertising, advances in technology have made it possible for website owners and their partners to collect anonymous data that enables a cleaner experience for visitors and establishes a foundation for selling ad space to advertisers.

Websites are assembled fresh each time they are displayed in a browser. Content, advertisements and other customizations are provided by various partners and are stitched together to form the website viewed by the user. The website 'calls' to web servers for these individual bits of content using JavaScript or HTML code. These lines of JavaScript or HTML code are called tags. In the interactive advertising ecosystem, tags are essential because for making calls to various ad servers, as well as for transferring information between parties to help tailor an experience for the user.

While tags can add value to a site, increased tagging may also create technical and operational challenges for the site. Common challenges include: operational strain, unintentional transfer of data, user abandonment, negative impact to customer experience (including performance issues), and increased privacy concerns. These challenges exist in part due to the proliferation of tags in the current digital advertising landscape.

The IAB *Site Tagging Best Practice* is an educational and procedural reference addressing common challenges for site tag and data management faced by advertisers and publishers. This document provides insight into the rise of site tag usage to meet evolving marketing needs, identifies areas for potential value loss to site owners, and offers prescribed best practices to mitigate these risks.

Intended Audience

The intended audience is any site owner, including both advertisers and publishers that employ third-party services requiring a piece of code on the site. This document would be beneficial to all stakeholders in the organization that manage and use vendors' services and tags. This can include marketing, analytics, IT, legal, as well as outside stakeholders, such as agencies and consultancies. Providers of technology that require tags may also benefit from this document, including product, engineering, and sales.

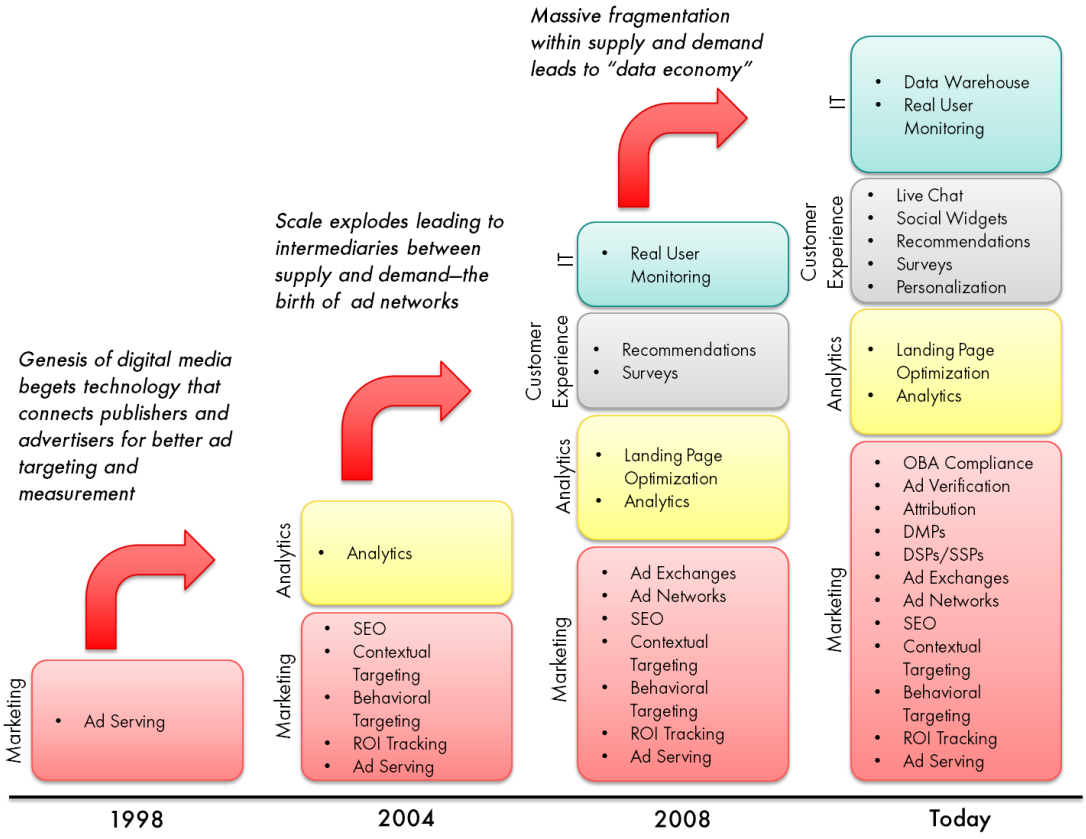
1 General Overview

In recent years, there has been significant growth in technologies allowing website owners to collect data used for optimizing their sites. An industry that began as a simple relationship between an advertiser and publisher to serve an ad on a publisher's page has evolved into a complex ecosystem of service providers that collect and manage data, sell inventory specific to select demographics, and offer advanced analytics and optimization tools. These services are enabled through the use of tags, also known as a pixel or beacon.

A tag is a lightweight fragment of code implemented on a website that, when called by the browser can facilitate real-time transfer of data between the originating site and another party, or may interact with the site layout and content. These transfers make it possible to create a targeted website or provide opportunities to optimize creative messaging for a more personalized user experience. As interactive advertising evolves, so does the proliferation in tags available on websites. The increase in tags has created an increase in operational strain, negative impact to user experience through latency, and increased privacy concerns with unintentional data transfers.

1.1 Background and History of Site Tagging

From its humble beginning, site tagging has become increasingly vital; it is used to target ads, customize content, and to provide insightful measurement and optimization tools. Site tagging has helped transform digital advertising from a system of basic ad serving to an extremely sophisticated and complex marketing system. The following figure depicts the evolution in site tagging as innovation in technology services provides insightful measurement and optimization tools, ultimately driving high value to site owners.



Basic Ad Serving: In the mid-1990s, ad serving technologies made it possible to deliver ad banners to web pages. This was the first use by websites of "tags" – the placement of a fragment of code sending information from a publisher website to a server (an ad server in this case) for marketing purposes. With ad serving, the recipient server returned HTML that rendered the ad banner in the site.

Ad Targeting: in the late 1990's, the use of tags rapidly expanded in marketing. Enhanced targeting techniques such as retargeting and the use of audience segments were established and quickly demonstrated the value of data driven advertising. This use of data ushered in a new era of personalized advertising and deeper marketing insights with improved measurement and analytics capabilities. More technology vendors emerged provisioning tags as the primary means of data collection from publisher and advertiser sites.

Data Collection and Analytics: by the mid-2000s, the flourishing digital advertising economy led to created opportunities for more participants in the digital advertising market. Marketing use cases expanded to include multiple channels and third-party stakeholders such as networks and exchanges that used tags to serve targeted ads or to collect data in order to make more informed decisions. Measurement and analytics also expanded - more data, collected through tags, allowed for deeper testing, modeling, and personalizing. Developments in technology enabled "container tags" (tags that you place on your site to trigger not one, but many other tags) to become commercially available to address operational challenges in implementing tag markup.

Real-Time Bidding and the Data Economy: today, rapid developments in technology have created a new interactive advertising environment where ad units are bought and sold (traded) automatically through advertising exchange systems. Ad inventory is traded between buyers and sellers by price matching or real-time bidding. Databases of aggregated user characteristics are the fuel in this automated marketplace, while tags are the conduits for collecting and moving data around the ecosystem.

1.2 Marketing Use Cases

The following table describes some of the ways in which companies can use tags to collect marketing data.

Issue	Description
Ad Serving	Delivers markup that returns targeted advertising in various formats to the page.
Optimization and monetization	Broad classification of tags that collect user interaction attributes with the page – typically for the purposes of segmentation, retargeting and data monetization.
Analytics	Ingests data for the purposes of measurement, common applications include attribution and enterprise analytics.
Customer Experience and Content	Delivers markup that returns requested content to the page, typically for site personalization or functions such as recommendation engines, surveys, chat or discussion, and social media experiences.
Who Benefits from use cases?	For more information on who benefits from the data collected from tags, refer to the Data Usage and Control Primer, page 6

1.3 Challenges and Risks

Although tags provide many marketing and revenue benefits for site owners, they also introduce risks and considerations that require action. This document addresses four central issues generated by tag use—data leakage, operational strain, customer experience impact, and privacy—and provides methods for managing, minimizing, and potentially eliminating their challenge or risk. Addressing these issues will allow site owners to reap the benefits that are provided by services that rely on tags, while maintaining strong practices to manage undesired effects.

Issue	Description
Data Leakage	The unintentional transfer of data from one party to another. Data leakage can lead to distinct business risks to publisher or marketer sites: To publisher sites: degrades the value of the audience as other parties may be able to sell campaigns against that publisher's audience. The end result is overall revenue leakage or market share erosion. To marketer sites: puts the site at a disadvantage in the market if its competitors are able to leverage its valuable data.
Operational Strain	The process of implementing, conducting Quality Assurance (QA), and maintaining tags can place unnecessary burden and strain on engineering resources, and increase the time it takes to go live with a vendor.
Customer Experience Impact	Tag code can interfere with page load, introduce latency, set cookies, cause XSS errors, security or browser warnings, and block or manipulate the contents of the page resulting in a poor user experience on the site.
Privacy	Due to a tag's ability to capture user data from the page and potentially reuse or pass it to other parties, consumer privacy is a concern when implementing third-party code on the site.

2 Site Tagging Best Practice Details

Tags provide substantial value to site owners, but may create negative impacts if tags are mismanaged or there is a lack of awareness as to their function and ability to transfer data. Planning, implementation and maintenance of tags is imperative to establish healthy workflow practices and procedures and to ensure security in page performance, data control, and privacy compliance.

2.1 Workflow: Planning, Implementation and Maintenance

2.1.1 Planning

In order to effectively manage tags, organizations should determine how they intend to manage data collection, storage and usage with business partners. Contracts and procedures should be constructed with data collection, storage and usage in mind. If these points are not acknowledged, the site owner risks data leakage and page latency, and subsequent loss in value or revenue. The following section will provide recommendations in recognizing and asserting your organizations data management and usage policies. For guidance around regulatory and self-regulatory compliance refer to the Privacy section (2.4)

2.1.1.1 Data Management Strategy

Publishers and advertisers should have an explicit data management strategy and adhere to best practices that ensure compliance, enforce strong privacy policies, actively monitor data leakage across web pages, and enable auditing the data practices of particular partners. Site owners should also develop a policy on what type of data is acceptable to collect, and to be more explicit about the value of their data assets. Leading web publishers and advertisers are beginning to reassert control of their data assets while also working more closely with partners to bring full transparency and protection to the creation, collection, buying and selling of consumer data on the web.

To effectively control data assets, it is important to have an established process to ensure proper data collection and control with each partner and vendor transaction. The following list of steps is an example of a process that can help reduce the operational cost of managing data.

Steps to secure data collection and control

1. Focus organizational attention on the potential opportunity cost that result from data leakage.
2. Maintain a single corporate data strategy with strong cross-team coordination.
3. Verify and assert privacy policy.
4. Define and codify data usage right (first-party, third-party, trusted partners).
5. Define and codify acceptable practices Terms and Conditions (Ts & Cs) with enforcement.
6. Ensure strong alignment between your privacy policy and acceptable practices for partners.
7. Assess existing relationships to see how well they are aligned with your data strategy and privacy policy.

Operational procedures are unique to each organization, and data management operations are no exception. While this document does not provide strategic operational details for data management, section 2.1.1.2 and 2.1.1.3 provide some general suggestions to consider when establishing an operational procedure for data management.

2.1.1.2 Enforcing Control through Insertion Order

Enforcing control of data management begins by aligning internal stakeholders' expectations and responsibilities. (For example, if you're a publisher, make sure that your audience development team is aligned with your sales/business development team regarding data collection practices and concerns.) Be sure to understand data protections afforded by existing agreements and have a data collection monitoring strategy to audit partner activity and enforce Terms and Conditions (Ts & Cs). Finally, ensure that all deals are reviewed with partners before implementation.

Reviewing IOs and partner agreements

You may not always be able to use your own contract during an insertion order (IO) or contract process. If you are asked to sign a partner's agreement, review it for the following considerations to ensure that it meets your policies:

- Record primary commonalities and differences in Ts & Cs.
- Classify the clear data threats and the opportunities to exert more control over your data.
- Provide recommendations across primary agreement/partner/client categories.
- Define and implement a monitoring process to enforce Ts & Cs and track top offenders.
- Examine the contract's controls and limitations around data collection and re-use, check to see:
    - If the contract explicitly allows for collection and reuse; review these deals more closely to make sure terms are worth the transfer of data rights.
    - If the contract lacks controls, definitions, or remedy for data usage:
        - Investigate actual collections.
        - Add a data rider to IO or publish a vendor policy.
    - If some protections are in place; encourage a shift to the Standard Terms and Conditions for Interactive Advertising Version 3.0, and refer specifically to clause XII: Non-Disclosure, Data Usage and Ownership, Privacy and laws.
    - If based on Standard Terms and Conditions for Interactive Advertising Version 3.0 with clear definitions; consider changes to definition of 'performance data' to no longer include site header.

Terms and Conditions	Standard Terms and Conditions for Interactive Advertising Version 3.0 can be found under the Terms and Conditions page of the IAB website. See clause XII: Non-Disclosure, Data Usage and Ownership, Privacy and Laws.

Data contract rider

If you'd like to sign a contract with a partner that does not include strong data collection and usage controls, encourage a shift to the Standard Terms and Conditions for Interactive Advertising Version 3.0. In instances where partners do not want to use the Standard T's & C's V3.0, we recommend attaching a data contract rider to explicitly describe data collection and usage.

When partners don't want to use the Standard T's & C's, but have no issue with the XII: Non-Disclosure, Data Usage and Ownership, Privacy and Laws clause within the Standard T's & C's V3.0, this clause could be used as the template for the rider rather than drafting one on your own.

But, if using the XII clause as your rider is not applicable, then your contract rider should do the following:

- Define your data as inclusive of all information served by the publisher or advertiser to the page.
- Establish a clear connection to your privacy policy.

- Establish responsibility for collection when a party acts as an usher to other parties, also known as "daisy-chaining."
- Mandate disclosure of all related-party collectors (served client-side or server to server) and their role for a service or campaign.
- Establish clear, practical audit rights.
- Establish remedies that provide equitable relief for unauthorized data reuse.
- Reconfigure terms and conditions with agencies, advertisers, and other business partners to establish new norms and limits for acceptable data collection.

B2B Data Usage and Control	For additional best practices on business-to-business data usage and control, refer to the Data Usage and Control Primer, page 10

2.1.1.3 Performance, Service Level Agreement's (SLA) and Outages

In addition to the above contractual considerations, a vendor's tag on your site can also affect page performance. Site owners should discuss the following with vendors

Timeout thresholds and SLA's

Is the vendor able to kill a tag response after a certain time? Vendors who are unable to do so may block the page from loading and cause an outage unless their tag can be loaded asynchronously (see section 2.2.1 for more information about performance). Will the vendor provide a guaranteed uptime or response time and actions for non-compliance?

Points of Presence

Depending on how widely dispersed a site's traffic is and how heavy it is at peak volume, a site owner may need to consider whether a vendor has global data centers with available capacity to handle large transient traffic spikes without incurring unacceptable latency or availability issues. The vendor should have a worldwide resource footprint and provide SLAs for latency and availability regardless of the location of the requests.

Escalation processes

Sites owners should have a technical escalation plan and points of contact from the vendor in the case of outages or errors. The vendor should guarantee a level of support that includes issue escalation and resolution with which the site owner is comfortable.

Data and third parties

For performance, QA, and data control purposes, sites owners should ask their partners what data they will collect and whether they will pass it to third-parties. This will help site owners identify additional network calls they may make, and assess performance impact. Sites owners should require transparency from third parties regarding shared tags (also known as daisy-chaining or piggybacking) and which endpoints, or additional tags might be included in the tag.

2.1.2 Implementation

When implementing tags, your main concerns should be that the tags will execute and collect data as expected in all browsers and that no unintended tags are included on the page. This is done through proper auditing and maintenance processes, and a good QA process during the initial implementation that incorporates flushing caches as part of the test sequence. Testing tags is done using proper auditing and maintenance procedures, and a good QA process during the initial implementation. After each change to a tag configuration, always QA

the site to ensure that the right tags are firing on the right pages and the right tags are transferring and collecting data properly. A number of free http debugging proxy tools, such as Firebug or HTTP Headers can help you test tag implementation.

2.1.2.1 New Tag Deployment Checklist:

The following provides some key deployment issues to check for:

Are there pre-existing issues?

- Take a snapshot of the page before and after a tag configuration.
- Document any loading errors that are pre-existing issues unrelated to tag implementation.

Are the intended tags firing?

- Take inventory of existing and intended tag implementations and carefully assess all use cases and which tags should be active.
- Using your browser QA tool of choice, confirm all tags are executing as they should. Make sure that there are no tags blocking the site at the top of the page and that requests and redirects are happening correctly.
- Check for any JavaScript run-time errors identified by the browser tool in the tag load process. If any new errors appear, speak to the vendor and investigate the error type and reason.

Is data being collected and passed correctly?

- Use tag QA software to review the parameters the tag is collecting
- Verifying with the vendor or in your account that the data has been properly collected.
- You should validate accuracy of data collection with each vendor.

Are tags affecting each other?

- After each tag configuration, confirm tags are not negatively affecting each other.
- Confirm that any tag dependencies are being met. For example, if an event-tracking tag in an analytics tool calls a library, the library tag must fire first.

Implementation Note	Successful QA requires a defined process and timeline with buy-in from all relevant stakeholders (relevant agency partners, IT, marketing, etc.).

2.1.3 Maintenance

As tag deployment is being planned, site owners should consider, monitoring and maintaining the tags on your site on an ongoing basis either through strict process or through specialized tag management technology to manage performance issues, privacy compliance, and control of data. Below are some considerations that should be included in a good maintenance plan:

Tag Management Technology

Container tags are the common technology for managing tags and effectively useful if maintained properly. Over time, multiple tags can be added to a container tag, which will eventually impact page performance. If deploying a tag management technology, site owners should take care to ensure that the technology is consistently and properly implemented on all relevant pages and that outdated tags are removed. The recommended practice for maintaining tag management technology is to audit assigned tags on a quarterly basis to ensure the desired tags are firing and that any necessary data is

always available. This practice will also reduce the occurrence of latency as unnecessary tags are removed.

Data Flighting
Sites owners should be aware of when data should be actively transferred. A good practice is to turn off data transfer when not needed.

Tag Placement
Site owners should have reasonable control and monitoring around placing tags on relevant categories or pages, allowing more fine-grained control and visibility into your tag configuration and conditional tag loading so that tags do not need to fire unless required so that tags do not need to fire unless required.

Review Tag Request
Site owners should put in place a system to continually monitor the number of third-party requests on your pages and make these metrics available for all stakeholders. If a large number of unexpected requests are occurring, the site owner should speak to the vendors to understand where they are coming from and why.

Ensure Version Control
Vendors will make changes to their tags and release new versions over time. Site owners should have a system in place to document and manage tag versioning and updates.

Security Certificates
Confirm vendors have current, valid security certificates where applicable.

Incorporating these maintenance best practices into your operations will ensure better hygiene and control of third-party assets on a site, which in turn can improve performance, compliance, and protection of data for site owners.

2.2 Performance

As the breadth of marketing use cases supported by tags expands, website IT teams that implement tags face increasing challenges. Any tag is a potential point of failure, and one failed tag can affect performance; producing poor customer experience, latency, interruption of content delivery, outages, and errors.

2.2.1 Loading Options

There are several options for loading tags, applying the right option depends on the tags' use cases and intended functionality. A site owner should be aware of loading options and select the appropriate loading method to help minimize potential risks to the site's pages:

Asynchronous loading
Loading a tag asynchronously means that the tag is loaded in parallel with page content so that any issues that occur in loading the tag won't interfere with page content loading. In most cases, tags should be rendered in the browser and loaded asynchronously so that it doesn't interfere with the user experience as page content is loaded. Tag loading should never block events such as `DomContentLoaded` and `Onload`. Asynchronous loading of tags may not be possible for all tags, however. For example, a tag may be delivering content that would create an awkward customer

experience if loaded asynchronously, or may be delivering code that other elements on the page require.

Synchronous tags
Synchronous loading is a method of loading tags sequentially, in line with the page content, so that page loading is delayed until assets defined by the tag have completely loaded. If a synchronous tag does not contain functionality that requires its placement at the top of the page, place it at the bottom when possible. (Some tools are also able to load normally synchronous tags such as these, but do it asynchronously without risk of overwriting the web page).

Server-side loading
Some vendors may be able to pass data between each other through a server-to-server connection rather than requiring a tag on the page. Server-side loading reduces the number of tags on the page, eliminating potential failure points, and also allows them to retrieve data from vendors with a presence on the page at a later time, even in the event of server outage.

2.2.1.1 Additional Engineering Checks

In addition to loading options, there are other factors that could impact page or site performance, such as the tag's mark-up or code functions. The following points provide best practices for engineers in the organization to identify and handle possible issues affecting page performance:

Real-time Monitoring
There are many ways tags can error: external resources could be unavailable, data passed in a pixel call could cause endpoints to error, and executing the tag code in the browser may throw any manner of exceptions. Consider having a real-time monitoring and logging system in place to identify issues that may occur after an initial QA check.

Proactive tag-monitoring should test tag latency asynchronously to browser-requests and suspend poorly performing tags. Constantly testing tags (or employing a system that has testing built-in) will provide better visibility into a tag's performance. Automated performance monitoring should deactivate tags that are performing poorly and reactivate upon improved performance. Site owners should consider using one or more independent monitoring services to check performance. Using an independent monitoring service can provide early warning alerts about performance or availability problems and can provide an independent measure to judge a vendor's performance. Independent monitoring is especially important for site owners who place tags on multiple sites and who might otherwise have no way to measure a vendor's performance.

Meaningful Logging
For meaningful logging, create an error logging system that allows you to easily identify when an issue is occurring. For instance, when code fails while executing in the browser (syntax errors, etc.), a simple try/catch block will often suffice for reviewing the event. If tags are loaded asynchronously such that the site is using standard DOM methods for adding elements (i.e. `document.createElement`), then `onload` and `onerror` event handlers can be leveraged to detect when an external resource cannot be loaded or fails.

Ensure safe client-side code
If your JavaScript is executing on the window document ensure that your variables, methods, objects and events do not conflict or override what is already in the webpage. Tags should not be overriding, hiding, or blocking any of the page content or any pre-existing behavior. Aside from explicit services

such as overlay ad display, any interference with page content or operation could lead to bad user experience.

Eliminate `Document.write`
`Document.write` is a JavaScript method that is frequently provisioned by vendors supplying tags. In general, implementing the tag with `document.write` is a bad practice because it executes immediately when encountered by a browser; while most other tags can be loaded at any time during the life of a page (such as after the DOM is ready, after the page is loaded, etc.), Therefore, the site and its vendors should avoid using `document.write` in any of their tags. Also, be aware that certain DOM/page events may not be available.

Remove redundant utility code from tag markup
Tags are often provisioned with inefficient code for functions like random number generation for cache busting, protocol detection, and `<noscript>` markup handling. This logic should not reside in the tag markup, but rather should be controlled by the site at run-time.

Browser and device compatibility
Proper testing is encouraged to ensure that vendor tags work properly across all browsers and devices. These tags should never break the user experience or cause JavaScript errors.

Consider adjusting the way the tag is called
Sometimes, a tag provisioned as an `<iframe>` or `<script>` can also be called as a lighter-weight `<img>`. Sites should consider the benefits and risks in changing the way the tag is delivered.

Run-Time Escalation Procedure
Site owners should have an escalation procedure that allows for real-time monitoring of unresponsive or problematic tags. This may include setting timeout thresholds for vendor tags, and implementing procedures around real-time de-activation and reactivation of tags.

2.3 Data Capture and Transfer

As discussed earlier, tags are usually placed on pages to collect data of some sort and pass it to a third-party system. Therefore, at the heart of tagging is the need for data capture. Site owners should be familiar with all the ways that data is transferred to third-parties including passing variables as parameters in tags, implementing tags that can read Document Object Model (DOM) attributes including AJAX and other interaction events to capture data, or via server-to-server connections after ID synchronization has been accomplished. Passing data can be a time-consuming, ongoing process that needs to happen with each new implementation.

2.3.1 Data Layer

When passing data to tags, a useful best practice is to create and leverage an explicit data layer, or to leverage emerging metadata standards that can be called by any tag. For example, a site owner may create a JavaScript variable or object that can be passed into any tool they wish. A data layer may include page attributes, visitor information, conversion information, or other parameters of interest. Using a data layer reduces the strain of passing variables with each implementation or redoing implementations when changes to the site occur.

2.3.2 Tag Wrapping

When a tag is placed directly onto a webpage, it has full access to the contents of that page and can execute JavaScript functions to collect a wide variety of data. Many tags do not need direct access to the page. One alternative is to wrap tags inside of an `<iframe>`, explicitly passing data through the iframe and into the tag.

When wrapping a tag in an iFrame, consider using an IAB SafeFrame. IAB SafeFrame enables rich interactions between ad content wrapped in an iFrame and the page on which it displays without allowing direct access to page data. With IAB SafeFrame, the publisher is offered more control over its data and both parties can execute rich interactions with transparency. IAB SafeFrame1.0 was released for public comment in November 2012, for further reference see http://www.iab.net/safeframe.

2.3.3 Server-Side

An alternative means for capture and transfer of data is via server-to-server connectivity where two parties sharing data with each other have a common identity for the user. In this scenario, related data is broadcasted between the parties without the use of a tag executing in the browser either in real-time or through batch delivery processes.

2.4 Privacy

Transfer of potentially sensitive information introduces privacy risks for all site owners. Best practices around site tagging should include awareness of the approved tags and how those tags will be seen under the relevant regulatory and self-regulatory procedures of the site owner. For recommendations in recognizing and asserting your organizations data management and usage policies, see the Workflow, Planning section (2.1.1)

For each tag, the most important privacy concerns are:

- What data will be collected
- How collected data will be used

2.4.1 Acceptable data collection and cookie usage

In most cases, data collected by vendors is limited to anonymous clickstream, behavioral data, and basic browser attributes. While personally identifiable information (PII), like address, phone number, or email address, is excluded from data collection, the site owner is responsible for immediately flagging any PII that shows up in results as a major concern for legal to review. Legal should also review any data collection that relates to actual or inferred information about medical conditions, sexual orientation, political affiliations, children, or children-related interest segments.

Site owners should ensure that vendors have full disclosure around the vendor's use of browser cookies, (either third-party or first-party) or other shared objects, and that this usage complies with the owner's own policies around acceptable cookie usage.

2.4.2 Transparency on data usage

How data will be used by each company that collects data on a tag should be clearly identified and documented. Vendors should disclose whether any service providers that they use (such as infrastructure / CDN vendors) in the delivery of the service keep, use, transfer (including across country borders), or monetize the

site's data in any way. These vendor-supplier relationships can have compliance implications, site owners should ensure that data usage complies with internal policies and any governing bodies with which the company complies.

2.4.3 Self-Regulatory Programs and Governing Laws

National and international trade associations offer regulatory guidelines that companies may comply with for good legal practice as well as a positive user experience. In addition, certain countries may require certain privacy practices in their laws. Publishers should ensure that any data collection, use and storage practices comply with any applicable laws or self-regulatory programs in which the publisher participates.

In the United States, the Digital Advertising Alliance (DAA) administers the Self-Regulatory Program for Online Behavioral Advertising which provides consumers with notice and choice as to how online data is collected and used. The U.S. IAB requires all of its members to participate in this program to the extent they are engaged in online behavioral advertising. Another example of a policy governing how tags may collect and use data is the eDirective in the European Union, which is written into law and is implemented differently in the 27 EU Member States.

For each tag being implemented on the publisher site, verify that any legal and self-regulatory compliance is met concerning data collection, usage, and storage.

3 Site Tagging Terminology

AJAX Events: A method that sites use to send data to, and retrieve data from, a server asynchronously. In the context of site tagging, this is typically associated with the need for a site to send data to third-parties when specific events occur.

Asynchronous tag loading: A load performance method where the browser loads tags parallel to the rest of the page rather than sequentially, therefore page content is not slowed or blocked as a result of loading the tag.

Cache-busting: The process by which content or HTML is served in such a manner as to minimize or prevent browsers or proxies from serving content from their cache. Cache-busting forces the browser or proxy to fetch a fresh copy of content for each request. Among other reasons, cache-busting is used to provide a more accurate count of the number of browser requests from a computer or to avoid serving an ad that had been previously cached.

Client-side scripts: A piece of code that contains or calls a computer program that is to be executed by the user's browser (the client) rather than on the web server. This functionality is often employed to enable web pages to have different and changing content depending on user input, environmental conditions, or other variables.

Conditional tag loading: A set of rules that are executed at run-time to determine when and which tags to load on user-assigned pages under specific pre-arranged conditions.

Container tag: A tag, or snippet of code, which, when called in the browser, loads and executes additional tags as specified by the user. Once the container tag is placed into production, users can add, change, or remove tags using the container without needing to re-code the page.

Content Delivery Network (CDN): A large, distributed system of servers deployed in multiple data centers in the Internet with the sole purpose of serving content to end-users with high availability and high performance

Daisy-chaining: Sometimes referred to as "sequential piggybacking", this describes a situation where a parent tag executed in the browser makes a call to its system and then redirects the request to additional tags, which in turn redirects to additional tags. Daisy-chaining usually happens when a seller has run out of inventory and would like to supplement it through an additional source or when an advertiser redirects the ad request to a different demand source instead of showing the ad. With daisy-chaining, a lack of transparency may result regarding the inventory an advertiser is purchasing and the creative that will be displayed on a site. It also creates additional latency by increasing the number of requests and decisions at run-time for a given ad call.

Data: In the context of this document, data refers to any piece of information about a webpage or how the webpage is used on a site that the publisher or any of its partners, vendors, or customers may be able to collect from a tag.

Data leakage: When a partner collects, uses, or provides data to a set of third parties without the site owner's authorization and/or knowledge.

Data elements: In context of this document, a variable present on the page in which the value may be potentially passed to a third-party script for collection and usage. Data elements are often an attribute of the page, user, or environment.

Document Object Model (DOM): A cross-platform and language-independent convention for representing and interacting with objects in HTML, XHTML and XML documents.

Measured latency: Usually tracked by a monitoring system, the calculated time it takes the page to load based on some event such as the DOM complete or `window.onload` events.

Noscript: A browser experience where JavaScript is unavailable or disabled. In this context, noscript is typically associated with an alternative way to deliver a JavaScript-formatted vendor tag (vendor JavaScript will not function).

Payload: The resulting code that is downloaded when a tag is invoked. Site owners should assess the payload of a tag as the code may impact customer experience.

Perceived latency: The latency of the page from a user experience perspective, or how long it takes the content of the page to load for the user, regardless of what tags are loading in the background.

Personally Identifiable Information (PII): Any information that can be used to uniquely distinguish individual identity, such as name, phone number, bank information, etc.

Piggybacking: When one tag redirects to or calls an additional tag that is not placed directly on the site.

Query-string: A common method to assert data elements from a site to a vendor, typically in a `key=value` form in the tag markup.

Server-side: A method to deliver data from the site to a vendor server, bypassing the need for any tag to be rendered in the browser.

Single point of failure (SPOF): A part of a system that, if it fails, will stop the entire system from working.

SLA: An acronym for Service-level agreement, a services contract where the level of service is formally defined, often promising a maximum response time or system performance.

Synchronous loading: A method of loading tags sequentially, in line with the page content, so that page loading is delayed until assets defined by the tag have completely loaded.

Tag: A fragment of code, also known as a pixel or beacon, typically implemented within a web site to enable the function of collecting data. The tag is typically called by a browser to facilitate real-time transfer of data between an originating site and another party, or may interact with the site layout and content.

Timeout thresholds: The ability to terminate a tag request or response after a pre-configured time.