comscore

Media Metrix Reporting

Distributed Content Impression Tag

document version: 2.9.1; released on May 15, 2024

for further information, please contact:
Comscore
Tag Support
+1 866 276 6972

Any transmission of data to Comscore is subject to the terms of the Data Sharing Agreement which can be accessed on *Comscore Direct (https://direct.comscore.com/Signup/Signup.aspx)*. Your transmission of data to Comscore constitutes acceptance of the Data Sharing Agreement without further action.

1 Introduction

In our continuous efforts to provide the market with highest quality audience measurement data, Comscore Media Metrix uses the *Unified Digital Measurement* methodology.

1.1 Unified Methodology

Unified Digital Measurement is a best-of-breed approach that puts the consumer — the human, not the machine — at the center of Comscore measurement and relies on Panel data as well as Census or Server data. Data from the Comscore panel provides a 360° view of the consumer including demographics, cross-visitation, etc. while Census data from tags provides overall, site-specific usage activity. Through the advent of measuring consumption of content and ads, Comscore has been measuring more entities and campaigns as consumer's media consumption grows across channels, devices, and environments. Grounded in rigorous methodology and by tying this data to panel observations, Comscore uses these assets together in what Comscore calls Unified Digital Measurement (UDM).

We're ensuring our new iteration — UDM 2.0 — will stand the test of time. The transition to a cookie-less world in any way that works for publishers or brands, whether that's via website or video streaming tags, SDKs for mobile and OTT applications, or via server-to-server data sharing. Comscore works with you on “where your data resides” to ensure accurate and robust reporting. A separate document about UDM 2.0 is available from your Comscore account team or implementation support team.

1.2 UDM 2.0 tag specification

UDM 2.0 combines first-party data from digital publishers and TV networks in a privacy-preserving manner to ensure audiences are represented with the same granularity and precision as you have come to expect. Comscore Media Metrix Distributed Content reporting provides independent and consistent measurement of the increasingly popular distributed-content market (i.e. applications, syndicated content, or other distributed content assets). It offers a reliable solution for publishers seeking to monetize their audiences as well as an alternate means for advertisers aiming to reach elusive demographic segments online.

Measurement of distributed content entities require a tagging-based measurement approach that serves as one of the raw ingredients in facilitating insight into the total number of unique visitors and the number of content impressions associated with the distributed content. Distributed Content entities will also be eligible for inclusion in the vertical categories available within the Comscore Client Focus Dictionary, including the Ad Focus list.

A key element for UDM 2.0 is that *third-party* (*3P*) identifiers of the consumer — for example a third-party cookie — may not be available. Reach and demographic reports in Comscore's audience reporting services like Media Metrix and Video Metrix will use information from panels, enhanced with *first-party* (*1P*) publisher-specific data of the consumer provided by the publisher. 1P Data can consist of:

- 1P identifier data (e.g., a first-party cookie set by the Comscore Publisher Tag, an obfuscated value of a first-party publisher cookie value or a login identifier)
- Optional demographics data — age group and gender — provided these are available

Please refer to *Appendix B: How to add 1P data to distributed content tags on page 10* for details on providing 1P data in tags.

1.3 What are the options for data collection?

There are two options for a publisher to provide data:

1. Use the same tags you are using today, include your 1P identifier, and insert (hashed or obfuscated) additional consumer information
2. Server-to-server transfer of web/app/streaming analytics data, including 1P identifier, and (hashed or obfuscated) additional consumer information.

This document describes the specifications required for entities that help distribute / syndicate content to be measured for Comscore Media Metrix reporting.

2 Requirements

In order for Comscore to track distributed content assets, clients are required to:

1. Contact their Comscore client services representative to obtain a discrete Publisher ID — also known as the *Client ID* or `c2` *parameter value* — which is a number with at least 7 digits.
2. Each client is required to tag individual objects as the asset is being impressed upon the page. The tag call will include parameters that include the Publisher ID referenced above and a Content ID (passed in the `c3` parameter), which the client will be responsible for creating and assigning. The Content ID must be an integer and must be included in the `c3` parameter of the tag call as the object or content asset is instantiated based on a user request.
   a. Only one discrete tag per object impression event.
   b. Reporting will default to a single line item for each reportable entity. Comscore can provide the capability for more granular reporting levels through a hierarchy. Clients wishing to have a reportable hierarchy for their different content assets must discretely tag their various content assets with unique Content IDs in the `c3` parameter.
   c. Clients must keep track of their Content IDs and submit any additions/deletions to the inventory to Comscore on a monthly basis. The mechanism for this inventory submission will be through a monthly Excel file delivered to Comscore that contains the following:
      1. Content ID: Self assigned unique integer (of up to 19 digits) for each discrete content asset that needs to be measured. If no reportable hierarchy is being defined, set `c3=1` for all content assets.

2. Content Friendly Name (required): An object name that will be used for reporting purposes if a reportable hierarchy is desired. The maximum depth of the hierarchy will be 3 tiers under each reportable client.
   For example:

   Client X

   a. Client X Automotive Content

      1. Client X Sports Cars
      2. Client X Sedans

   b. Client X Financial Content

      1. Client X Stocks
      2. Client X Banking

3. Category Information (optional): An optional categorization of any level of the hierarchy into a traditional Comscore client-focus dictionary (CFD) category. This categorization will be in addition to the default classification within the Distributed Content list and will be subject to the CFD rules for inclusion in any given vertical category.

3 Implementing the Distributed Content Tag

The Comscore Distributed Content Tag uses asynchronous loading of its JavaScript file so that loading and processing of the web page is unaffected by the tag. A `<img>` HTML element is present for cases where JavaScript cannot be executed. The tag uses the parameters listed below, which are required to ensure appropriate data categorization.

Comscore Distributed Content Tag parameters

Parameter	Required or Optional	Description	Explanation	Example value
`c1`	**Required**	Tag Type	Pre-populated with fixed value `7`	`7`
`c2`	**Required**	Comscore Publisher ID	A Comscore-provided number with at least 7 digits that is unique for each publisher(1). This value is also used in the URL of the *beacon.js* file.	`1234567`
`c3`	**Required**	Content Identifier	Client defined unique integer (of up to 19 digits) to denote a reportable content hierarchy. Set `c3=1` for all content if no hierarchy is being defined.	`7394751` *(translates through dictionary to a "Game X" channel within the Distributed Content Hierarchy)*

Comscore combines 1P data from digital publishers and TV networks in a privacy-preserving manner to ensure your audiences are represented with the same granularity and precision as you have come to expect. The tag tries to set a third-party cookie regardless of whether or not JavaScript can be executed. The JavaScript code of the tag has opt-in functionality which will let it additionally try to set a publisher-specific first-party cookie with the name `_scor_uid`.

---

(1) The Publisher ID is also known as the *Client ID* or *c2 value*.

While the first-party cookie functionality is disabled by default, **it is strongly advised to enable the first-party cookie functionality** by using the `enableFirstPartyCookie: true` setting, as is shown in the code example below. Optionally, you can add additional information about the consumer to enhance your results. Please refer to *Appendix B: How to add 1P data to distributed content tags on page 10* for details on providing 1P data.

To implement the tag, copy its HTML code into your web page content and provide the appropriate parameter values. **This code uses two example values which you should replace:**

1. **The Comscore Publisher ID `1234567` is used in multiple locations, which should all be replaced with the value you are assigned.**
2. **The example Content Identifier value `7394751` should be replaced with an appropriate value.**

```
<!-- Begin Comscore Tag -->
<script>
    var _comscore = _comscore || [];
    _comscore.push({
        c1: "7", c2: "1234567", c3: "7394751",
        options: {
            enableFirstPartyCookie: true
        }
    });

    (function() {
        var s = document.createElement("script"), el = document.getElementsByTagName("script")[0]; s.async = true;
        s.src = "https://sb.scorecardresearch.com/cs/1234567/beacon.js";
        el.parentNode.insertBefore(s, el);
    })();
</script>
<noscript>
    <img src="https://sb.scorecardresearch.com/p?c1=7&c2=1234567&c3=7394751&cv=4.4.0&cj=1">
</noscript>
<!-- End Comscore Tag -->
```

This code example also includes the `options` property with the `enableFirstPartyCookie` setting set to to `true` to enable the first-party cookie functionality of the Publisher Tag.

About first-party cookie lists...

If you enable the first-party cookies and you maintain a list of first-party cookies — for example as part of your privacy statement — then **please make sure to add the `_scor_uid` cookie** to your list. Its purpose is Audience Measurement. Please be aware the Comscore Publisher Tag uses this cookie in accordance with policies surrounding user consent signals as described in *on page 0*.

In addition to the mentioned parameters, the following parameters are automatically collected by the JavaScript code of the tag:

Automatically collected parameters

Parameter	Description
`c7`	Full Page URL
`c8`	Page Title
`c9`	Referring URL
`cs_fpcu`	First-party `_scor_uid` cookie value. Only collected if the first-party cookie functionality is enabled and tag can read a previously set first-party cookie.

The Comscore Distributed Content Tag will continue to collect data for the same purposes as approved with your current implementation.

Appendix A: User Consent Tagging

Applicable privacy and data protection laws and regulations may require companies to capture and/or document a user's consent for measurement. For example, the European Union's General Data Protection Regulation (“GDPR”) and the Privacy and Electronic Communication Directive 2002/58/EC and the California Consumer Privacy Act (“CCPA”) have requirements regarding capturing user consent or providing consumers the ability to opt-out of the sale of personal information, where appropriate. Please note that the implications of applicable privacy and data protection laws and regulations may vary and are best evaluated by each individual business.

Using a Consent Management Platform

If you are using a Consent Management Platform (CMP) which implements iAB Transparency and Consent Framework (TCF) version 2.0, then the Comscore Distributed Content Tag integrates with the CMP to automatically collect user consent. No additional steps are necessary to enable this integration, other than to make sure the Comscore Distributed Content Tag is in the web page where it can access the CMP as per the TCF 2.0 technical specification.

Sometimes JavaScript code execution might not be possible. Publishers are expected to provide two extra parameters in the `<img>` HTML code of the tag, but only if they can retrieve and incorporate Transparency and Consent data. The two parameters are `gdpr` and `gdpr_consent`, both used according to the Consent Management Platform (CMP) specification.

> **For populating `gdpr_consent` it is important to note Comscore's TCF vendor id is 77.**

If JavaScript code execution is not possible and Transparency and Consent data cannot be retrieved, then please implement manual consent communication in the `<img>` HTML code of the tag as explained in the next section.

If you are using the Comscore Distributed Content Tag in an iframe and the tag is expected to use its TCF 2.0 CMP integration to automatically retrieve and collect user consent data, then you are expected to emulate `__tcfapi()` inside the iframe. An example script that emulates the in-frame `__tcfapi()` call, is provided by the *TCF 2.0 specification (https://github.com/InteractiveAdvertisingBureau/GDPR-Transparency-and-Consent-Framework/blob/master/TCFv2/IAB%20Tech%20Lab%20-%20CMP%20API%20v2.md#is-there-a-sample-iframe-script-call-to-the-cmp-api)* . This script can be copied verbatim from the TCF 2.0 specification and implemented in your iframe.

Manually Communicating Consent

If you are not using a Consent Management Platform then you need to manually communicate user preference (e.g., did a user opt in or out of measurement), where required, by adding parameter `cs_ucfr` to the collected data. The required values for this user consent parameter are:

Label `cs_ucfr` values for communicating user consent

Value	Interpretation	Usage
`0`	User has not given consent or has opted out	Use this value to indicate the user 1. has been asked for consent where the user did not give consent, or 2. enabled the option to opt out (e.g., opt out of the sale of personal information)
`1`	User has given consent	Use this value to indicate the user has been asked for consent where the user has given consent to collect data for measurement
	User has not taken an action	Use an empty string value (i.e., blank) to indicate the user has not taken an action

About including parameter `cs_ucfr` when not communicating user consent…

If you do not communicate consent for a user, then **do not populate** label `cs_ucfr`.

A publisher must add parameter `cs_ucfr` with an appropriate value to the web page impression tagging by adding the parameter and its value to the HTML and JavaScript code. With this change of the HTML and JavaScript code the publisher **should not change any other collected data values**.

For example, assuming the user has given consent, the aforementioned tag code would be changed into:

```
<!-- Begin Comscore Tag -->
<script>
    var _comscore = _comscore || [];
    _comscore.push({
        c1: "7", c2: "1234567", c3: "7394751", cs_ucfr: "1",
        options: {
            enableFirstPartyCookie: true
        }
    });

    (function() {
        var s = document.createElement("script"), el = document.getElementsByTagName("script")[0]; s.async = true;
        s.src = "https://sb.scorecardresearch.com/cs/1234567/beacon.js";
        el.parentNode.insertBefore(s, el);
    })();
</script>
<noscript>
    <img src="https://sb.scorecardresearch.com/p?c1=7&c2=1234567&c3=7394751&cs_ucfr=1&cv=4.4.0&cj=1">
</noscript>
<!-- End Comscore Tag -->
```

Bypass User Consent Requirement for First-party Cookie

When the first-party cookie functionality is enabled the Publisher Tag inspects available user consent signals to determine setting the cookie is allowed.

Based upon the self-determination of a website/publisher that no applicable jurisdiction requires it to obtain consent for cookies, Comscore offers the option for the implementer to indicate that the website/publisher is exempt from any user consent requirements. If this applies, the implementer can use the `bypassUserConsentRequirementFor1PCookie: true` setting:

```
<!-- Begin Comscore Tag -->
<script>
    var _comscore = _comscore || [];
    _comscore.push({
        c1: "7", c2: "1234567", c3: "7394751"
        options: {
            enableFirstPartyCookie: true
            bypassUserConsentRequirementFor1PCookie: true
        }
    });

    (function() {
        var s = document.createElement("script"), el = document.getElementsByTagName("script")[0]; s.async = true;
        s.src = "https://sb.scorecardresearch.com/cs/1234567/beacon.js";
        el.parentNode.insertBefore(s, el);
    })();
</script>
<noscript>
    <img src="https://sb.scorecardresearch.com/p?c1=7&c2=1234567&c3=7394751&cv=4.4.0&cj=1">
</noscript>
<!-- End Comscore Tag -->
```

Appendix B: How to add 1P data to distributed content tags

The following optional parameters can be added to provide 1P identifier data and additional demographics data about the consumer:

Tag parameters for 1P data

Parameter	Required or Optional	Description	Explanation	Example value
`cs_fpid`	Mandatory(2)	1P identifier	Contains the pseudonymized 1P identifier value, which could be either a user_id, login (preferred), OpenID or a first-party cookie.	`1570113661206_6059410249`
`cs_fpit`	Optional	Type of identifier in `cs_fpid`	Specifies the type of identifier ▪ `li`: logged-in ID ▪ `lo`: logged-out ID ▪ `o`: OpenID ▪ `c`: first-party cookie If `cs_fpit` is not included alongside `cs_fpid` then Comscore will assume the identifier is a first-party cookie.	`c`
`cs_fpdm`	Optional	1P demographics data	Contains an obfuscated value of the demographics data which belong to the 1P identifier. This obfuscation calculation is explained further below this table.	`39642313001`
`cs_fpdt`	Optional	Type of 1P demographics data in `cs_fpdm`	Indicates the origin of the demographics values. Accepted values: ▪ `01`: collected by publisher ▪ `02`: collected through / purchased from third party ▪ `03`: mixed sources or modelled ▪ `99`: unknown origin	`01`

Please ensure that all four parameters are provided together. If you cannot populate a suitable value for any of the parameters — e.g., when demographics data is not available — then please use value `*null` instead.

Demographics data should not be collected for children. If the consumer is a child (ages 0 - 17) then please use value `*null` for `cs_fpdm`.

The 1P identifier is expected to stick to the same consumer in the same browser where possible, or even across browsers if a user- or login identifier is used. 1P demographics data is calculated as `19991999999` added to to the concatenation of birth date, age group and gender, where birth date has the format `yyyyymmdd`, age group has the format `xx` and gender has the format `z`. In other words: `yyyyymmddxxz + 19991999999`.

Age groups and gender are shown in the following tables. **The age groups in the range 0 - 17 are not listed because demographics data should not be collected for children.**

(2) `cs_fpid` is required if you choose not to enable first-party cookie functionality. Please refer to *Implementing the Distributed Content Tag on page 4* for more details about the first-party cookie functionality.

Age groups identifiers

Identifier	Age group
00	00
06	18 - 20
07	21 - 24
08	25 - 34
09	35 - 44
10	45 - 54
11	55 - 64
12	65+

Gender identifiers

Identifier	Age group
0	Unknown
1	Male
2	Female
3	Unspecified / Other

Age group and birth date are complementary, but the preference is to collect birth date. When birth date is provided, age group can be omitted by using 0 values and vice versa. To illustrate:

Demographics values examples

Demographics of consumer	Demographics value calculation	Description
Female born on March 13, 1965	19650313002 + 19991999999 = 39642313001	The age group uses zeroes because the birth date is known
Male in the age between 35-45	00000000091 + 19991999999 = 19992000090	The birth date uses zeroes because only the age group was available

Further below is a tag code example, assuming the following 1P data values:

- Identifier: 1605266069802_50777152
- Identifier type is first-party cookie: c
- Demographics values: 39642313001
- Demographics collected by publisher: 01

```
<!-- Begin Comscore Tag -->
<script>
    var _comscore = _comscore || [];
    _comscore.push({
        c1: "7", c2: "1234567", c3: "7394751",
        cs_fpid: "1605266069802_50777152", cs_fpit: "c", cs_fpdm: "39642313001", cs_fpdt: "01",
        options: {
            enableFirstPartyCookie: true
        }
    });

    (function() {
        var s = document.createElement("script"), el = document.getElementsByTagName("script")[0]; s.async = true;
        s.src = "https://sb.scorecardresearch.com/cs/1234567/beacon.js";
        el.parentNode.insertBefore(s, el);
    })();
</script>
<noscript>
    <img src="https://sb.scorecardresearch.com/p?c1=7&c2=1234567&c3=7394751&cs_fpid=1605266069802_50777152&cs_fpit=c
&cs_fpdm=39642313001&cs_fpdt=01&cv=4.4.0&cj=1">
</noscript>
<!-- End Comscore Tag -->
```

Appendix C: Update an Existing Comscore Publisher Tag Implementation

To update an existing Comscore Distributed Content Tag Implementation typically only the URL of the *beacon.js* file needs to be updated. This URL is typically found in one of the JavaScript statements. For example:

```
<!-- Begin comScore Tag -->
<script>
    var _comscore = _comscore || [];
    _comscore.push({ c1: "7", c2: "1234567", c3: "7394751" });

    (function() {
        var s = document.createElement("script"), el = document.getElementsByTagName("script")[0]; s.async = true;
        s.src = (document.location.protocol == "https:" ? "https://sb" : "http://b") + ".scorecardresearch.com/beacon.js";
        el.parentNode.insertBefore(s, el);
    })();
</script>
<noscript>
    <img src="https://sb.scorecardresearch.com/p?c1=7&c2=1234567&c3=7394751&cv=2.0&cj=1">
</noscript>
<!-- End comScore Tag -->
```

Please replace the old URL with one using the Comscore Publisher ID and let the URL use secure transmission. Please also enable the first-party cookie functionality feature as advised. With these changes, the example code would be changed like this:

```
<!-- Begin Comscore Tag -->
<script>
    var _comscore = _comscore || [];
    _comscore.push({
        c1: "7", c2: "1234567", c3: "7394751",
        options: {
            enableFirstPartyCookie: true
        }
    });

    (function() {
        var s = document.createElement("script"), el = document.getElementsByTagName("script")[0]; s.async = true;
        s.src = "https://sb.scorecardresearch.com/cs/1234567/beacon.js";
        el.parentNode.insertBefore(s, el);
    })();
</script>
<noscript>
    <img src="https://sb.scorecardresearch.com/p?c1=7&c2=1234567&c3=7394751&cv=4.4.0&cj=1">
</noscript>
<!-- End Comscore Tag -->
```

In the above example, the Comscore Publisher ID is `1234567` and is used in the URL of the *beacon.js* file.

If your existing implementation has additional parameters being collected aside from the `c1`, `c2` and `c3` shown in the examples above then **please do not change those parameters as part of your update**.