

The cover features a white background with abstract, flowing blue and light blue wave-like patterns at the top and bottom. The title is centered in a black, italicized serif font.

Tealeaf CX Mobile Administration Manual

Contents

IBM Tealeaf CX Mobile Administration Manual.....	1
CX Mobile configuration.....	1
CX Mobile configuration components.....	1
Configuration prerequisites.....	3
CX Mobile initial configuration.....	3
Configure Tealeaf for mobile visitors.....	5
WURFL configuration.....	6
Tealeaf reference fields for CX Mobile.....	7
Overview of user agents for CX Mobile.....	8
Manage User Agents in Tealeaf Mobile.....	8
Configure User Agent Detection.....	13
User Agent tools.....	18
User agent configuration files.....	20
Maintain the CX system.....	20
Maintenance for CX Mobile.....	24
Tealeaf Reference Session Agent.....	25
Tutorial.....	25
Add the session agent.....	25
Path reference values.....	26
More reference parsing options.....	28
Configuration settings.....	37
Tealeaf reference session agent in multi-canister environments.....	42
IBM Tealeaf documentation and help.....	43
 Index.....	 46

IBM Tealeaf CX Mobile Administration Manual

This manual describes the administrative Tealeaf CX features that can be used for mobile sessions. These features and tasks include managing user and session agents, and configuring CX components.

CX Mobile configuration

The IBM® Tealeaf® CX Mobile module components are configured with the Tealeaf components that are required for IBM Tealeaf CX Mobile. The required Tealeaf components include IBM Tealeaf cxImpact. You must have a license for the Tealeaf mobile components.

The IBM Tealeaf CX Mobile module is a separately licensed module of the IBM Tealeaf CX platform. for more information, please contact your IBM Tealeaf representative.

The IBM Tealeaf CX Mobile module uses some resources and components that are shared with the base IBM Tealeaf cxImpact product. In this some sections of the manual, the documentation overlaps with content from the "cxImpact Administration Manual". For purposes of providing all background information that is required to configure IBM Tealeaf CX Mobile, these materials are included in this document.

CX Mobile configuration components

You configure several Tealeaf components for IBM Tealeaf CX Mobile. The IBM Tealeaf CX Mobile module is a separately licensed module of the IBM Tealeaf CX platform, for more information, please contact your IBM Tealeaf representative..

For more information about enabling Tealeaf and the data that is acquired from mobile devices, see "Overview of CX Mobile" in the *IBM Tealeaf CX Mobile User Manual*.

The IBM Tealeaf CX Mobile module uses some resources and components that are shared with the base IBM Tealeaf cxImpact product. In some sections of the manual, the documentation overlaps with content from the "cxImpact Administration Manual" in the *IBM Tealeaf cxImpact Administration Manual*. For purposes of providing all background information that is required to configure IBM Tealeaf CX Mobile, these materials are included in this document.

Tealeaf CX Mobile license

When you deploy the IBM Tealeaf CX Mobile license, your installation has access to the following functionality and components:

1. Capture of client-side events from your mobile native applications for all Tealeaf mobile client frameworks.
2. Capture and recording mobile device data with the session after the WURFL public standard is converted for use in Tealeaf.
3. Access to mobile-specific features during session replay in Browser-Based Replay (BBR) and IBM Tealeaf CX RealTime Viewer (RTV). After you deploy the IBM Tealeaf CX Mobile license, each RTV user in your environment must update the local cached version of the license key through the application. BBR automatically checks for the license key during replay.
4. Capture of mobile device events, such as gestures and touches, that are submitted to Tealeaf when you deploy IBM Tealeaf CX UI Capture for AJAX. mobile device events, such as gestures and touches, are captured from the device and submitted to Tealeaf.

When you enable the licensed capabilities, you can download the mobile dashboard template from Tealeaf to report on your mobile data.

CX Mobile components

This table lists and describes the CX Mobile components that you install and configure:

Component	Description
SessionAgentTLTRef.dll	<p>This pipeline agent adds the [ExtendedUserAgent] section to the HTTP(S) request buffer.</p> <p>Install Locations:</p> <ul style="list-style-type: none">• Windows pipeline• Installed on HBR machines if deployed.• See "TMS Pipeline Editor" in the <i>IBM Tealeaf cxImpact Administration Manual</i>.
UserAgentRevealer.exe	<p>Use this tool to manually add user agent entries.</p> <ul style="list-style-type: none">• Install Location: root Tealeaf installation folder.
Wurfl2csv.exe	<p>This file is a stand-alone executable file that converts public domain WURFL.xml for use by Tealeaf Reference session agent to populate the Extended User Agent fields in the HTTP(S) request buffer.</p> <ul style="list-style-type: none">• Install Location: Tealeaf directory on Windows Capture Pipeline
Browscap.csv	<p>This file contains known user agents in .csv formats, as known and reported by a publicly accessible standard.</p> <ul style="list-style-type: none">• Install Location: <Tealeaf>\System
UserCap.csv	<p>This file can be used to supplement Browscap.csv and WURFL with more definitions.</p> <ul style="list-style-type: none">• Install Location: <Tealeaf>\System
Mobile native application frameworks	<p>As part of the installation of IBM Tealeaf CX Mobile, the following mobile native application frameworks are installed on the Portal Server:</p> <ul style="list-style-type: none">• Tealeaf Android Logging Framework• Tealeaf iOS Logging Framework <p>These logging frameworks are installed as ZIP files in the following location on the Portal Server:</p> <div><pre><Tealeaf_install_directory>\NativeMobile SDK</pre></div> <p>When implemented, these frameworks capture client-side user interface events and other device information from mobile native applications on the supported platforms.</p> <p>Use of the Tealeaf client frameworks requires more installation and configuration and deployment with your mobile native applications.</p> <p>For more information:</p> <ul style="list-style-type: none">• See "Tealeaf Android Logging Framework Reference Guide" in the <i>IBM Tealeaf Android Logging Framework Reference Guide</i>• See "Tealeaf iOS Logging Framework Reference Guide" in the <i>IBM Tealeaf iOS Logging Framework Reference Guide</i>

Component	Description
	The UI Capture framework can be deployed to capture more user interface events through the mobile browsing experience.
Tealeaf RealTea Viewer	<p>IBM Tealeaf CX RealTea Viewer can be used to replay mobile sessions that use the same screen dimensions as the mobile device.</p> <ul style="list-style-type: none"> • Install Location: Tealeaf user desktop • See "RealTea Viewer (RTV) User Manual" in the <i>IBM Tealeaf RealTea Viewer User Manual</i>. • You might also replay mobile sessions through Browser-Based Replay, a Portal-based replay feature that requires no additional installation. See "CX Browser Based Replay" in the <i>IBM Tealeaf cxImpact User Manual</i>.
Tealeaf CX Mobile Dashboard (optional)	The IBM Tealeaf CX Mobile Dashboard can be downloaded and imported into your IBM Tealeaf CX solution. It contains a number of events that are configured for detecting characteristics of mobile user agents.

Configuration prerequisites

CX Mobile uses features from IBM Tealeaf cxImpact. When you you configure CX Mobile, you must first complete the initial configuration steps for IBM Tealeaf cxImpact.

Install

Before you begin configuring, install all IBM Tealeaf software on Windows or Linux servers. For more information, see the *IBM Tealeaf CX Installation Manual*.

UI Capture for RTV Replay

For RTV users, the IBM Tealeaf CX Mobile license must be deployed to the application from one of the hosting Tealeaf servers to enable proper replay of Mobile Web sessions. See "Search and Replay for Mobile Web" in the *IBM Tealeaf CX Mobile User Manual*.

UI Capture for Replay

To replay Mobile Web sessions in Release 8.1 or later, you must install or upgrade to IBM Tealeaf CX UI Capture for AJAX to build 2011.03.15.1 or later.

- See "UI Capture FAQ" in the *IBM Tealeaf UI Capture for Ajax FAQ*.
- See "UI Capture for Ajax Guide" in the *IBM Tealeaf UI Capture for Ajax Guide*.
- Replay of sessions that are captured from native applications has a different set of requirements. See "Search and Replay for Mobile App" in the *IBM Tealeaf CX Mobile User Manual*.

CX Mobile initial configuration

The IBM Tealeaf CX Mobile module extends the user agent detection, analysis, and reporting capabilities of IBM Tealeaf cxImpact to track user agents for mobile devices. Through IBM Tealeaf CX Mobile, you can identify and track the experiences of your customers connecting to your web application by mobile devices.

This section describes the initial configuration of one component of the IBM Tealeaf CX system in a simplified deployment model. Depending on your Tealeaf solution's deployment, more configuration might be required. If you have any questions about configuration, contact <http://support.tealeaf.com>.

Configuring Tealeaf CX Mobile

IBM Tealeaf CX Mobile for Mobile Web relies on the WURFL standard for detecting mobile user agents. This .csv file must be downloaded, converted into a usable form for Tealeaf, and then made available to the IBM Tealeaf CX platform. This standard must be updated regularly. See "Configuring Tealeaf for Mobile Visitors" in the *IBM Tealeaf CX Mobile Administration Manual*.

Acquire Mobile Dashboards

You can download a selection of dashboard templates for import into your Tealeaf environment, including the Mobile Traffic Dashboard that is specifically configured to track mobile visitors with a set of events that are included in the import.

Configure events for CX Mobile

The [ExtendedUserAgent] section is added to the request by enabling extended user agent parsing.

- For more information about enabling this feature, see "Tealeaf Reference Session Agent" in the *IBM Tealeaf CX Configuration Manual*.

For more information about the events for mobile visitors that become available through extended user agent parsing, see "Events for Mobile Visitors" in the *IBM Tealeaf CX Mobile User Manual*.

Initial configuration for CX Mobile for Mobile App

To capture data from mobile native applications, you must deploy one or more of the following Logging Frameworks with your application. When deployed and configured, the following Logging Frameworks capture user interface events and application properties from native applications that are developed for the listed mobile operating systems.

Note: Use of the Tealeaf Logging Frameworks for mobile native applications requires the Tealeaf CX Mobile license for Mobile App. For more information, contact your Tealeaf representative.

Note: Licensees must use code that is provided by Tealeaf in their apps. For more information on downloading Tealeaf software, see [Tealeaf CX Downloads](#).

Logging Framework Description

Android Logging Framework

Logging Framework for Android-based applications. See "Tealeaf Android Logging Framework Reference Guide" in the *IBM Tealeaf Android Logging Framework Reference Guide*.

iOS Logging Framework

Logging Framework for iOS-based applications. See "Tealeaf iOS Logging Framework Reference Guide" in the *IBM Tealeaf iOS Logging Framework Reference Guide*.

Testing your configuration

For IBM Tealeaf CX Mobile, you must verify that the Mobile Dashboard was properly imported and that mobile events are appearing in reports. Testing the CX Mobile is part of end-to-end testing of your Tealeaf components.

About this task

You might need to wait a few minutes after you enable user agent detection and import the mobile dashboard before mobile events are being triggered by session data. If you experience difficulties while you are searching for events or viewing the Mobile Dashboard, wait a few minutes and try again.

When all Tealeaf components are configured, complete an end-to-end test. See "Testing Your Tealeaf Solution" in the *IBM Tealeaf CX Configuration Manual*.

Procedure

1. Search for events. Through the Portal or RTV, run a search for any of the mobile events that are imported with the dashboard.
2. Review the Mobile Dashboard. If you installed the Mobile Traffic dashboard:
 - a) In the **Dashboard** menu, select the **Mobile dashboard**.
 - b) Review the content in the dashboard to verify that it is being populated with data.
3. Review other Tealeaf reports. Check the other mobile visitor reports in Tealeaf to verify that they are being populated with data.

Configure Tealeaf for mobile visitors

IBM Tealeaf CX Mobile is designed to integrate with the IBM Tealeaf cxImpact platform. IBM Tealeaf CX Mobile uses the user agent detection capabilities of the IBM Tealeaf cxImpact platform to detect user agent-related information in the capture stream and then performs a lookup of the user agent data against the WURFL public standard.

Configuration tasks

To capture the rapidly evolving mobile device customer base, IBM Tealeaf CX Mobile requires more configuration and periodic updates to reflect the changing landscape of mobile devices. The configuration tasks that you do include:

1. Update any of these files that are in your installation:

File	Description	Source
browscap.csv	<p>This file is a public standard that is used for detecting fixed and bot user agents. It is required for extended user agent detection.</p> <p>The file is included as part of your distribution. However, due to the frequency of introduction of new user agents, it should be updated now and regularly from the source.</p> <p>Tealeaf supports only the csv version of this file.</p>	http://browsers.garykeith.com
WURFL	<p>This file is a public standard that is used for detecting mobile user agents. It is required for IBM Tealeaf CX Mobile. See “WURFL configuration” on page 6.</p> <p>Due to the frequency of introduction of new mobile user agents, update the WURFL regularly from the source. See “Maintenance for CX Mobile” on page 24.</p> <p>After this file is downloaded, it must be converted with the <code>Wurfl2csv.exe</code> utility into a format that is usable by Tealeaf.</p>	http://wurfl.sourceforge.net/
UserCap.csv	<p>This file is used to overwrite the <code>Browscap.csv</code> file with changes to user agent strings.</p> <p>New entries can be inserted into <code>UserCap.csv</code> with the <code>UserAgentRevealer.exe</code> utility.</p>	<p>Tealeaf</p> <p>See “Configuring User Agent Detection” in the <i>IBM Tealeaf cxImpact Administration Manual</i>.</p>

File	Description	Source
UserSupplement.csv	<p>This file is used to supplement the Browscap.csv file with more user agent strings that you detect in the capture stream.</p> <p>New entries can be inserted into UserCap.csv with the UserAgentRevealer.exe utility.</p>	<p>Tealeaf</p> <p>See "Configuring User Agent Detection" in the <i>IBM Tealeaf cxImpact Administration Manual</i>.</p>

2. Install and enable the Tealeaf Reference session agent in your Canister pipeline or pipelines. This session agent detects user agent data and then perform lookups into the configured files to determine matches.
3. Within Tealeaf Reference session agent, enable extended user agent detection.
4. Restart the Transport Service on each Canister.
5. Use a mobile device to browse your web application. If you can locate the session, you can verify the presence of mobile user agent information.
6. Use the IBM Tealeaf CX Mobile dashboard to view Tealeaf pre-configured events. The pre-configured events detect mobile visitors and their related data through the IBM Tealeaf CX Mobile dashboard. For more information about downloading and installing this dashboard in your Tealeaf system, see the *IBM Tealeaf CX Mobile User Manual*.
7. Replay mobile visitor sessions with RTV.
8. Verify that events are being reported.

Maintenance

Periodically, update the BrowsCap.csv and WURFL.xml files.

- WURFL: See ["Maintenance for CX Mobile"](#) on page 24.
- BrowsCap.csv: See "Maintaining the CX System" in the *IBM Tealeaf CX Installation Manual*.

Configuring capture of user interface events from mobile browsers

If you licensed and deployed UI Capture, you can configure the deployed JavaScripts to capture user interface events for mobile browsers and to forward them to Tealeaf for processing.

- See "UI Capture FAQ" in the *IBM Tealeaf UI Capture for Ajax FAQ*.

See "UI Capture for Ajax Reference" in the *IBM Tealeaf UI Capture for Ajax Guide*.

WURFL configuration

IBM Tealeaf CX Mobile relies on the WURFL standard for mobile device identification. Available in XML and CSV format, the WURFL standards file contains a maintained list of all known mobile devices. When this file is downloaded, converted, and imported into Tealeaf, IBM Tealeaf CX Mobile can identify mobile devices by performing lookups against this provided data set.

Periodically download and update your version of WURFL from the official source. For more information about downloading, see "Maintaining the CX System" in the *IBM Tealeaf CX Installation Manual*.

To update your Tealeaf solution with a downloaded version of WURFL, you can use an external utility that is provided with IBM Tealeaf CX Mobile.

After you integrate a WURFL version into your Tealeaf solution, you can segment reports and dashboards that are based on the mobile device name, browser, and key functional characteristics.

User agent tutorial

For information about how user agent information is used within Tealeaf, including steps to configure event objects to track user agent information, see "E2E Scenario - Tracking User Agent Information in Tealeaf" in the *IBM Tealeaf cxImpact User Manual*.

Custom user agent detection

You can augment the WURFL standard with user-created information about user agents. Through the import of a customized version of the browser detection config file, you can update your Tealeaf solution to meet your enterprise data capture and reporting needs. See "Configuring User Agent Detection" in the *IBM Tealeaf cxImpact Administration Manual*.

Example user agent strings for Safari for iPhone

The Safari user agent string for iPhone follows a non-standard pattern that might require special event management. These examples demonstrate valid instances of the user agent string before and after a software update.

String before update	String after update
Mozilla/5.0 (iPhone; U; CPU iPhone OS 2_0 like Mac OS X; en-us) AppleWebKit/525.18.1 (KHTML, like Gecko)	Mozilla/5.0 (iPhone; U; CPU iPhone OS 2_0 like Mac OS X; en-us) AppleWebKit/525.18.2 (KHTML, like Gecko)
Mozilla/5.0 (iPhone; U; CPU like Mac OS X; en-us) AppleWebKit/420.1 (KHTML, like Gecko) Version/3.0 Mobile/3A110a Safari/419.3	Mozilla/5.0 (iPhone; U; CPU like Mac OS X; en-us) AppleWebKit/420.1 (KHTML, like Gecko) Version/3.0 Mobile/3A145a Safari/419.3
Mozilla/5.0 (iPhone; U; iPhone OS 2_0 like Mac OS X; en-us) AppleWebKit/525.18.1 (KHTML, like Gecko) Version/3.1.1 Mobile/5A308 Safari/525.20	Mozilla/5.0 (iPhone; U; iPhone OS 2_0 like Mac OS X; en-us) AppleWebKit/525.18.1 (KHTML, like Gecko) Version/3.1.1 Mobile/5A468 Safari/525.20

Replay for CX Mobile

To replay visitor sessions for mobile devices as the visitor experienced them, more configuration of IBM Tealeaf CX RealTea Viewer and Browser-Based Replay might be required. See "Search and Replay for Mobile Web" in the *IBM Tealeaf CX Mobile User Manual*.

Tealeaf reference fields for CX Mobile

When WURFL is integrated, Tealeaf adds the WURFL data into a self-pruning cache of reference information that Tealeaf can be configured to reference.

As data is passed through the Windows pipeline, the Tealeaf Reference session agent captures user agent strings and performs a lookup against the available public standards, including WURFL. When matches are found in WURFL, data is populated in the request and might later be evaluated by the Event Engine. This information is available for searching.

Without more information, Tealeaf provides the event objects to store captured mobile device data for search and reporting purposes.

Reference dimensions

Using dimensions, you can configure remappings of captured data values as identified through WURFL to values consistent with your enterprise practices. For example, you can create mappings to identify versions and subversions of browsers so that you can track browser type at any level of granularity.

Mobile web session Traffic type

When the Tealeaf Reference session agent matches the provided user agent string against an entry in the WURFL standard, the mobile traffic type request name-value pair is defined:

```
TLT_TRAFFIC_TYPE=MOBILE
```

The request variable TLT_TRAFFIC_TYPE is used to populate the **Traffic Type** dimension that is provided by Tealeaf. If sessions are initiated from a mobile web browser, this value is set to MOBILE.

You can use the **Traffic Type** dimension to segment your reporting for sessions that are initiated from mobile web browsers. In the Tealeaf Report Builder, you add the **Traffic Type** dimension and then filter the dimension to display only the MOBILE value.

Mobile native application Traffic Type

For sessions initiated from mobile native applications, the Tealeaf client frameworks submit user agent information, which is interpreted by the Tealeaf Reference session agent.

If the user agent information identifies the session as a mobile native application session, the traffic type value is populated in the request:

```
TLT_TRAFFIC_TYPE=MOBILE_APP
```

This value is recorded for the **Traffic Type** dimension.

Overview of user agents for CX Mobile

User agents are used with the IBM Tealeaf CX Mobile module. User agents are part of the Tealeaf cxImpact component. The information presented here is from the IBM TealeafcxImpact Administrator's Manual.

Manage User Agents in Tealeaf Mobile

Through Tealeaf, you can monitor the different types of user agents that contact your web application. Through a provided set of data objects, Tealeaf can identify the type of traffic that is requesting resources from your web application and then monitor counts and other information that pertains to the type of user agent. These objects can be used as the source data for developing a useful set of user agent-related reports, enabling you to closely monitor the composition of traffic to your site.

Tealeaf User Agents

The term user agent identifies a broad class of entities that can communicate with web servers across the Internet. This broad class includes desktop browsers, bots, and mobile devices, among others.

With each request to a web server, most devices include a string that uniquely identifies the device. This string can then be compared a public standard to ascertain other characteristics of the device that submitted the string.

How Tealeaf tracks user agent information

Tealeaf uses a four step process to handle user agents

Process

The process that Tealeaf uses to process user agents is:

1. Client sends user agent string. When a request is submitted to a web server, the client (browser or native application) submits a string that contains the publicly identifiable user agent string in the following variable, called a header, in the request. This example is a user agent string:

```
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0;
.NET CLR 1.1.4322; .NET CLR 2.0.50727; .NET CLR 3.0.04506.30; .NET CLR
3.0.04506.648; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729; .NET4.0C)
```

Depending on the type of client, the user agent string might be delivered in a different manner.

2. Perform lookup of the user agent string against public standards. When the user agent string is detected, Tealeaf performs a lookup against the applicable public standards:
 - Browscap for fixed desktop browsers
 - WURFL for mobile web browsers
 - Self-reporting from the Tealeaf client frameworks for mobile native applications
 - This step is performed by the Tealeaf Reference session agent. This step is included in the default pipeline. You do not need to do any additional configuration.
3. Populate request variables with data. If a match between the submitted user agent string is found in one of the applicable standards, additional information that is contained in the public standard entry for the user agent is populated in custom variables that are stored in the request by Tealeaf.
4. Use request data. Various Tealeaf components reference user agent information for search, replay, and reporting purposes, after you configure data objects.

User agent strings by client type

Depending on the type of client device that is used by the visitor to experience your web application, its user agent information might be forwarded in a different manner for verification by the listed public standard.

This table lists and describes the standards that are used for different user agent sources:

Table 1. User agent strings by client type		
Client Type	Source of user agent string	Standard used
fixed desktop browser	User agent string is submitted by default by the browser	browscap

Table 1. User agent strings by client type (continued)

Client Type	Source of user agent string	Standard used
Android mobile native application	<p>Tealeaf client framework submits user agent string</p> <p>Capture of Android-based mobile native applications requires the IBM Tealeaf CX license, the IBM Tealeaf CX Mobile license, and deployment of the Android Logging Framework. See "Tealeaf Android Logging Framework Reference Guide" in the <i>IBM Tealeaf Android Logging Framework Reference Guide</i>.</p> <p>Since mobile native applications submit a user agent string only if specified by the developer, the Android Logging Framework submits a string as well as a set of properties in the HTTP_X_TEALEAF_PROPERTY header.</p> <ul style="list-style-type: none"> For more information on the submitted properties, see "Tealeaf Reference Session Agent" in the <i>IBM Tealeaf CX Configuration Manual</i>. 	WURFL
iOS mobile native application	<p>Tealeaf client framework submits user agent string.</p> <p>Capture of IOS-based mobile native applications requires the IBM Tealeaf CX license, the IBM Tealeaf CX Mobile license, and deployment of the IOS Logging Framework. See "Tealeaf iOS Logging Framework Reference Guide" in the <i>IBM Tealeaf iOS Logging Framework Reference Guide</i>.</p> <p>Since mobile native applications submit a user agent string only if specified by the developer, the iOS Logging Framework submits a string as well as a set of properties in the HTTP_X_TEALEAF_PROPERTY header.</p> <ul style="list-style-type: none"> For more information on the submitted properties, see "Tealeaf Reference Session Agent" in the <i>IBM Tealeaf CX Configuration Manual</i>. 	WURFL
IBM Tealeaf CX UI Capture for AJAX fixed desktop browser	<p>User agent string is submitted by default by browser</p> <p>Note: Capture of AJAX-based web applications requires the IBM Tealeaf CX license and deployment of the IBM Tealeaf CX UI Capture for AJAX client framework. See "UI Capture for AJAX Guide" in the <i>IBM Tealeaf UI Capture for AJAX Guide</i>.</p>	browscap

Table 1. User agent strings by client type (continued)		
Client Type	Source of user agent string	Standard used
IBM Tealeaf CX UI Capture for AJAX mobile web browser	<p>User agent string is submitted by default by the mobile web browser.</p> <p>Capture of AJAX-based mobile web applications requires the IBM Tealeaf CX license, the IBM Tealeaf CX Mobile license, and deployment of the IBM Tealeaf CX UI Capture for AJAX client framework. See "UI Capture for AJAX Guide" in the <i>IBM Tealeaf UI Capture for AJAX Guide</i>.</p> <p>Since the user agent information is submitted by the browser, mobile web applications captured by UI Capture do not include the additional properties that are submitted in the HTTP_X_TEALEAF_PROPERTY header for mobile native applications.</p>	WURFL

How Tealeaf stores user agent information

When a match is found for the user agent string that is submitted by the client, the following information is inserted into the [ExtendedUserAgent] section of the request.

Common values

There are several values that are populated for a matched user agent.

These values are populated for a matched user agent:

```
[ExtendedUserAgent]
TLT_BROWSER=Firefox
TLT_BROWSER_VERSION=Firefox3.6
TLT_BROWSER_PLATFORM=Win XP
TLT_TRAFFIC_TYPE=BROWSER
TLT_BROWSER_JAVASCRIPT=true
TLT_BROWSER_COOKIES=true
```

TLT_TRAFFIC_TYPE request variable

As part of its processing, the Tealeaf Reference session agent inserts the TLT_TRAFFIC_TYPE request variable.

This example shows a TLT_TRAFFIC_TYPE request variable:

```
TLT_TRAFFIC_TYPE=BOT
```

These values populate the Traffic Type dimension that is provided by Tealeaf.

- See "TEM Dimensions Tab" in the *IBM Tealeaf Event Manager Manual*.

Special Values

For the following values for TLT_TRAFFIC_TYPE, extra request variables might be inserted.

Value	Description
-------	-------------

MOBILE

TLT_TRAFFIC_TYPE=MOBILE indicates a session that is gathered from a mobile web browser through IBM Tealeaf CX UI Capture for AJAX.

- See [“Mobile Web sessions”](#) on page 12 below.
- For more information on UI Capture, see "UI Capture FAQ" in the *IBM Tealeaf UI Capture for AJAX FAQ*.

MOBILE_APP

TLT_TRAFFIC_TYPE=MOBILE_APP indicates a session that is gathered from a mobile native application that is monitored by one of the Tealeaf client frameworks.

- See [“User Agents and Mobile native app sessions”](#) on page 12.
- See "Tealeaf Android Logging Framework Reference Guide" in the *IBM Tealeaf Android Logging Framework Reference Guide*.
- See "Tealeaf iOS Logging Framework Reference Guide" in the *IBM Tealeaf iOS Logging Framework Reference Guide*.

Mobile Web sessions

If TLT_TRAFFIC_TYPE=MOBILE, then these fields can be inserted and populated for the Mobile Web user agent:

```
TLT_SCREEN_HEIGHT  
TLT_SCREEN_WIDTH  
TLT_COLOR_DEPTH  
TLT_PICTURE_SUPPORT  
TLT_VIDEO_SUPPORT  
TLT_STREAMING_SUPPORT
```

User Agents and Mobile native app sessions

User agent information for a mobile native application is submitted at the discretion of the application developer. To enable normalized user agent information, each Tealeaf client framework submits a specific header, which enables detection of user agent information.

When this header and the associated properties are detected in a hit, these values are populated in the [ExtendedUserAgent] based on gathered and submitted data from the client framework.

```
[ExtendedUserAgent]  
TLT_BROWSER=StraussAndPlesser Native  
TLT_BROWSER_VERSION=8.5  
TLT_BROWSER_PLATFORM=Android  
TLT_BRAND=Asus  
TLT_MODEL=Asus Eee Pad Transformer TF101  
TLT_SCREEN_HEIGHT=800  
TLT_SCREEN_WIDTH=1280  
TLT_COLOR_DEPTH=65536
```

Enable extended user agent detection

Before you extend user agent detection, you must configure the Tealeaf Reference session agent to enable extended user agent parsing. When enabled, captured user agent information is posted into the request in the [ExtendedUserAgent] section by the Tealeaf Reference session agent.

- See "Tealeaf Reference Session Agent" in the *IBM Tealeaf CX Configuration Manual*.

Mobile User Agents and robots

Tealeaf provides mechanisms for monitoring traffic from mobile devices and from robots.

The number of mobile web visitors continues to expand at a rapid rate. Depending on your web application, these visitors might have special requirements and customer issues. The ability to track their activities independent of desktop and other types of visitors is essential in providing good service to this market.

The IBM Tealeaf CX Mobile module enables insightful tracking and useful monitoring of mobile visitors and their devices. See "Tealeaf CX Mobile Administration Manual" in the *IBM Tealeaf CX Mobile Administration Manual*.

Short for "robot," a bot is a program that is designed to crawl pages of a website. A bot can be used for indexing a website for search engines, accumulating price information, and other organizational purposes. Tealeaf enables the monitoring of bot activities on your website. This type of activity can be tracked easily through RTV and Portal searches.

- See "bot Detection" in the *IBM Tealeaf cxImpact Administration Manual*.

Public Standards

Tealeaf uses two publicly maintained standards to monitor user agents. These standards are individual files that contain all known user agents and mobile devices.

- BrowsCap.csv is the standard for fixed user agents.
- WURFL.csv is the standard for mobile devices.

You can also customize your own set of user agents and apply it to your Tealeaf installation. Through UserCap.csv, you can provide configuration information for fixed and mobile user agents. These settings can be used to add new user agents to the standards listed. For more information on how Tealeaf uses these configuration files, see "Configuring User Agent Detection" in the *IBM Tealeaf cxImpact Administration Manual*.

These files should be updated regularly. See "Maintaining the CX System" in the *IBM Tealeaf CX Installation Manual*.

User Agent Tools

To facilitate easy management of configuration files, Tealeaf provides a set of external tools for converting the files to a format that Tealeaf can use and for reviewing the uploaded files.

See "User Agent Tools" in the *IBM Tealeaf cxImpact Administration Manual*.

User Agent Events

You can monitor and report on the user agents that visit your site. Tealeaf provides a set of default data objects that you can enable and use to develop the events that are required to meet your reporting needs.

- See "Configuring User Agent Events" in the *IBM Tealeaf cxImpact Administration Manual*.

Tutorial

Tealeaf has a tutorial that provides more information on how user agent information is used within Tealeaf, including steps to configure event objects to track user agent information.

See "E2E Scenario - Tracking User Agent Information in Tealeaf" in the *IBM Tealeaf cxImpact User Manual*.

Import User Agent Report Templates

Tealeaf publishes report templates to help monitoring user agent activity. These templates can be downloaded through Tealeaf Online Help and imported and customized to meet the requirements of your Tealeaf solution.

Configure User Agent Detection

This section describes how to acquire and maintain the configuration files necessary for parsing of fixed and mobile user agents and how to prepare those files for use in the IBM Tealeaf CX platform.

User agent detection must be enabled through the Tealeaf Reference session agent.

Types of files

There are several files needed by Tealeaf to collect and display various user agent information for reporting.

This table lists and describes the files used to collect and display various user agent information:

File	Description
WURFL.csv	This file contains the library of user agent information for mobile devices. This file is maintained by a third-party and should be updated in your Tealeaf solution on a regular basis. See “WURFL.csv” on page 14.
browscap.csv	A publicly available standard for user agents on the web. This file is maintained by a third-party and should be updated in your Tealeaf solution on a regular basis. See “browscap.csv” on page 14.
UserCap.csv	<p>This file is created and maintained by Tealeaf in a format similar to the above browscap.csv. You can use this file to overwrite the values for specific user agents in the browscap.csv standard.</p> <ul style="list-style-type: none"> • This file can be used to insert user agent values that are not present in the public standard. • See “UserCap.csv” on page 15.
UserSupplement.csv	This file is created and maintained by Tealeaf in a format similar to the above browscap.csv. You can use this file to add user agent values that do not appear in either public standard. See “UserSupplement.csv” on page 15.

WURFL.csv

The WURFL.xml public standard contains all the user agent information for known mobile devices.

Management of mobile user agents requires the IBM Tealeaf CX Mobile module, a separately licensable component of the IBM Tealeaf CX platform. for more information, please contact your IBM Tealeaf representative.

- For more information on enabling, see "Overview of CX Mobile" in the *IBM Tealeaf CX Mobile User Manual*.
- For more information on data that is acquired from mobile devices, see "Overview of CX Mobile" in the *IBM Tealeaf CX Mobile User Manual*.

Periodically update your local version of WURFL from the public standard.

Do not change to this standard-based file, which is overwritten whenever an update is made from the public standard. Apply new entries or update entries in the UserSupplement.csv and UserCap.csv files.

WURFL.XML must be downloaded and updated in your Tealeaf system regularly. See "Maintaining the CX System" in the *IBM Tealeaf CX Installation Manual*.

When it is downloaded, WURFL.xml must be converted from its source .xml file to the .csv format,. This format is required for Tealeaf use with the WURFL2CSV.exe utility.

- Before it is used, the Tealeaf software verifies that this file is in the proper format.
- For more information on converting WURFL.xml, see "User Agent Tools" in the *IBM Tealeaf cxImpact Administration Manual*.

browscap.csv

The browscap.csv file contains a list of all known user agents and system information that is related to them. This file is maintained by Gary Keith and can be collected from his website.

The browscap.csv file is available in various formats but only the .csv file format is supported by Tealeaf. The browscap.csv file is verified by Tealeaf to be in the original format.

Do not change this standard-based file, which is overwritten whenever an update is made from the public standard. Apply new entries or update entries in the UserSupplement.csv and UserCap.csv files.

The browscap.csv file must be downloaded and updated in your Tealeaf system regularly. See "Maintaining the CX System" in the *IBM Tealeaf CX Installation Manual*.

UserCap.csv

In some cases, you might need to overwrite the user agent strings and their corresponding data that is available in the public standards. The UserCap.csv file is used to overwrite user agent strings. For example, you might need to correct errors that your enterprise detects in the public standards with more accurate information.

Tealeaf provides the UserCap.csv file, which can be used to override values in the public standards. By gathering the user agent string value from the source file, you can then create a new entry in the UserCap.csv file through the UserAgentRevealer.exe utility. During evaluation of user agent strings, the Tealeaf Reference session agent checks UserCap.csv to see whether it contains replacement values for any detected user agents. If so, the replacement values are used instead.

Configurations in UserCap.csv replace corresponding entries in browscap.csv or WURFL.csv. If you want to add values that do not appear in these files, use UserSupplement.csv. For more information on replacing values by configuration file, see ["UserSupplement.csv" on page 15](#).

Be careful in creating replacement user agent strings. UserCap.csv takes precedence over the public standards at run time. It is possible for the public standards to become updated without having those updates reflected in the user agent data that is consumed by Tealeaf. Review the contents of UserCap.csv whenever updates are made to the public standards in use by your Tealeaf implementation.

An example version of UserCap.csv is included during installation. You can configure entries in this file with the UserAgentRevealer.exe utility. See ["Review Configuration Files" on page 16](#).

UserSupplement.csv

If you need to add custom user agent strings for fixed or mobile user agents, you can add the configurations to the UserSupplement.csv file. The UserSupplement.csv file is used for adding new definitions that are not covered in browscap.csv and WURFL.csv definitions.

Using the UserSupplement.csv file allows you to include anomalous or unreported data for purposes of reporting. For example, you can use this file to include a single value for reporting purposes without updating all configuration files.

Configurations in UserSupplement.csv do not override any corresponding entries in browscap.csv or WURFL.csv, which are public standards. For more information on replacing those values, see ["UserCap.csv" on page 15](#).

An example version of UserSupplement.csv is included during installation. You can configure entries in this file with the UserAgentRevealer.exe utility. See ["Review Configuration Files" on page 16](#).

Creating UserSupplement.csv

By default, a version of UserSupplement.csv is not provided with your installation. If you need to create a version of UserSupplement.csv, complete the following steps.

Procedure

1. Log in to one of the Processing Servers (Canisters) as an administrator.
2. Navigate to this directory:

```
<Tealeaf_install_directory>\System
```

3. Create a copy of usercap.csv.
4. Rename the copied file: UserSupplement.csv.
5. Edit the file in Microsoft Excel.
6. Remove all data lines in the file so that only the header remains.

7. Use UserAgentRevealer.exe to add your entries to the file. See [“Reviewing configuration information”](#) on page 17.
8. Save the file and store it in the same directory.
9. Copy this file to all Canisters that process hits.
10. Restart the Transport Service on each Canister.

Review Configuration Files

Tealeaf includes the User Agent Revealer utility (UserAgentRevealer.exe) for reviewing the contents of the user agent configuration files, which can grow large. You can enter a user agent string into the utility, which compares the provided values against all available configuration files then displays how the Tealeaf installation reports the string.

- This tool is used primarily for debugging purposes. You must run the tool from the machine where it is stored.

Note: This tool requires access to browscap.csv, which must be installed in the system directory inside the Tealeaf install directory.

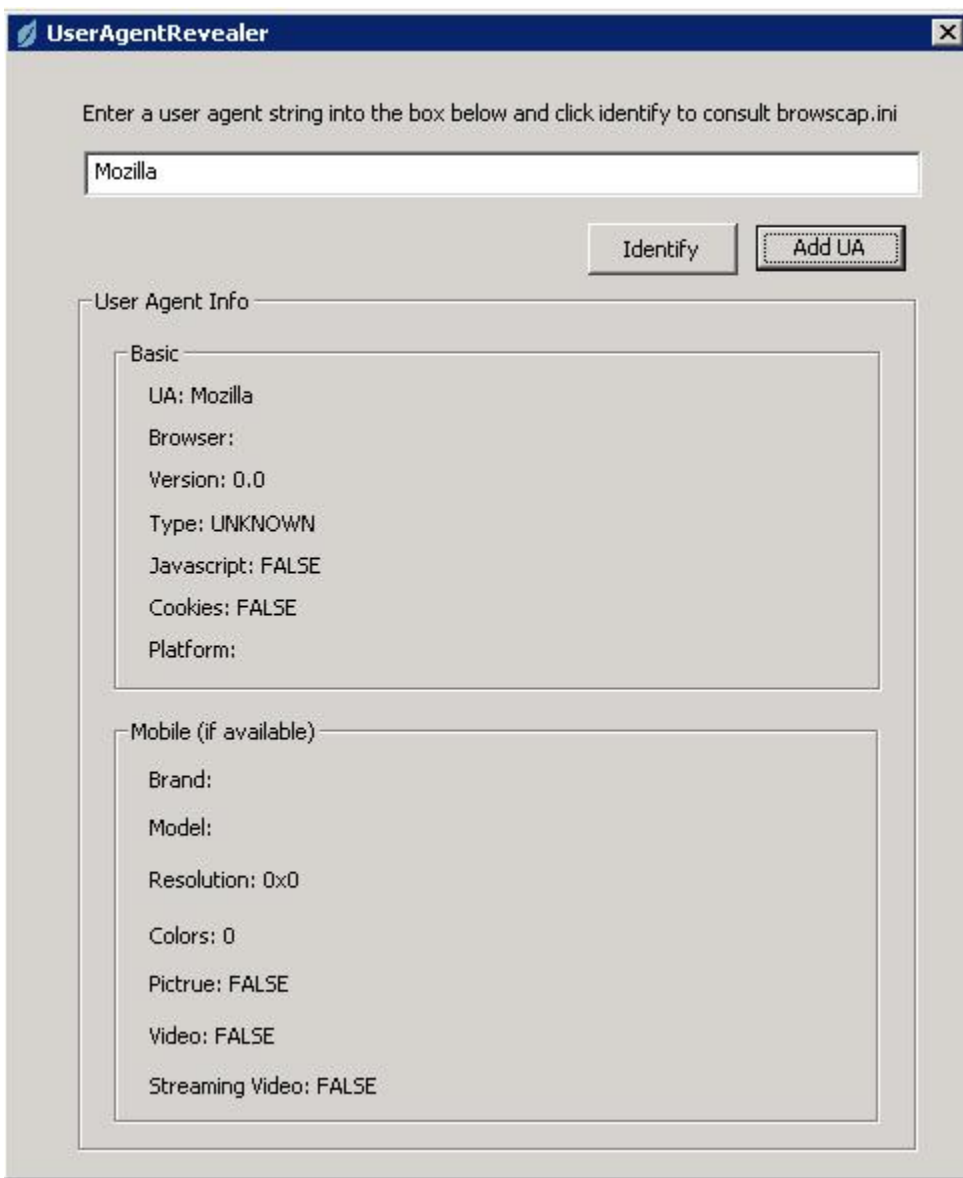


Figure 1. User Agent Revealer

Reviewing configuration information

Review the configuration information identify entries that should be added to the `UserSupplement.csv` file or corrected in the `UserCap.csv` file.

Procedure

1. In the Tealeaf install directory, double-click `UserAgentRevealer.exe`.
2. Paste a user agent string into the textbox. Click **Identify**.
3. Any available information for the user agent is displayed.
 - If no match is found, you may create a new entry in `UserSupplement.csv`.
 - If a match is found with incorrect data in it, you may create an entry in `UserCap.csv` to replace it.
 - See [“Editing or creating user agent entries” on page 17](#).

Editing or creating user agent entries

You can edit found entries or add new ones through `UserAgentRevealer.exe`.

About this task

If the user agent entry cannot be identified, the Type value in the `usersupplement.csv` file is set to UNKNOWN. In these instances, you can add the user agent string to the `usersupplement.csv` file with values that you define.

If the user agent entry in the `usercap.csv` file contains errors or values that you would like to change, you can create an entry with the overriding data in `usercap.csv`. These values are used instead of the values that are listed in the applicable public standard (`browscap.csv` or `WURFL.csv`).

All information that is entered into the fields is case-sensitive. Double quotation marks cannot be used and are stripped from the input.

Procedure

1. To edit an identified user agent entry or to create a new one, click **Add UA**.
2. Select the appropriate Destination:
3. Fill out all relevant fields.
4. To commit the changes, click **Commit**.
5. The selected Destination file is updated and saved.
6. When Tealeaf next refreshes its cache from the Destination file, the user agent information is applied to all subsequent hits.

Default User Agent fields

These fields are available by default for user agent tracking.

Field

Description

Destination

User-defined .csv where the new or edited entry is saved.

- `usersupplement.csv` - for new user agent entries
- `usercap.csv` - for edited user agent entries

Group Name

User agent group name to which the new agent belongs.

UA String

String that you entered in the Identify textbox.

Browser

Browser type.

Version

Version number, including major and minor version indicators. For example, the entries for IE 6.0 in these fields would be 6 and 0

Platform

Operating system.

Brand

Brand name for the user agent.

Model

Model name or number.

Screen

Screen size in Height and Width in pixels.

Color Depth

The number of bits of color depth. For example, if the user agent has 32-bit color depth, enter 32.

Pic Support

Select if the user agent supports display of picture formats.

Vid Support

Select if the user agent supports display of video formats.

Streaming Support

Select if the user agent supports display of streaming formats.

Cookie Support

Select if the user agent supports cookies.

Javascript Support

Select if the user agent supports JavaScript.

Crawler

Select if the user agent is a bot.

Mobile

Select if the user agent is a mobile device.

Stripper

Select if the user agent is a stripper (a bot that scrapes websites for content).

Syndication Reader

Select if the user agent is a reader, such as an RSS feed.

Enable extended user agent detection

After the configuration files are downloaded and prepared for use, you can configure the Tealeaf Reference session agent to use them for user agent detection.

By default, extended user agent parsing is enabled.

See "Tealeaf Reference Session Agent" in the *IBM Tealeaf CX Configuration Manual*.

User Agent tools

These external utilities can be used to monitor and update user agent configuration data that is required by Tealeaf.

Converting the WURFL file

Use the WURFL2CSV .exe utility to convert the downloaded source .xml for the WURFL file to the .csv format, which is required by Tealeaf.

About this task

For more information on where to acquire WURFL, see "Maintaining the CX System" in the *IBM Tealeaf CX Installation Manual*.

Converting the WURFL file with WURFL2CSV . exe requires the IBM Tealeaf CX Mobile module, a separately licensable component of the IBM Tealeaf CX platform. for more information, please contact your IBM Tealeaf representative.

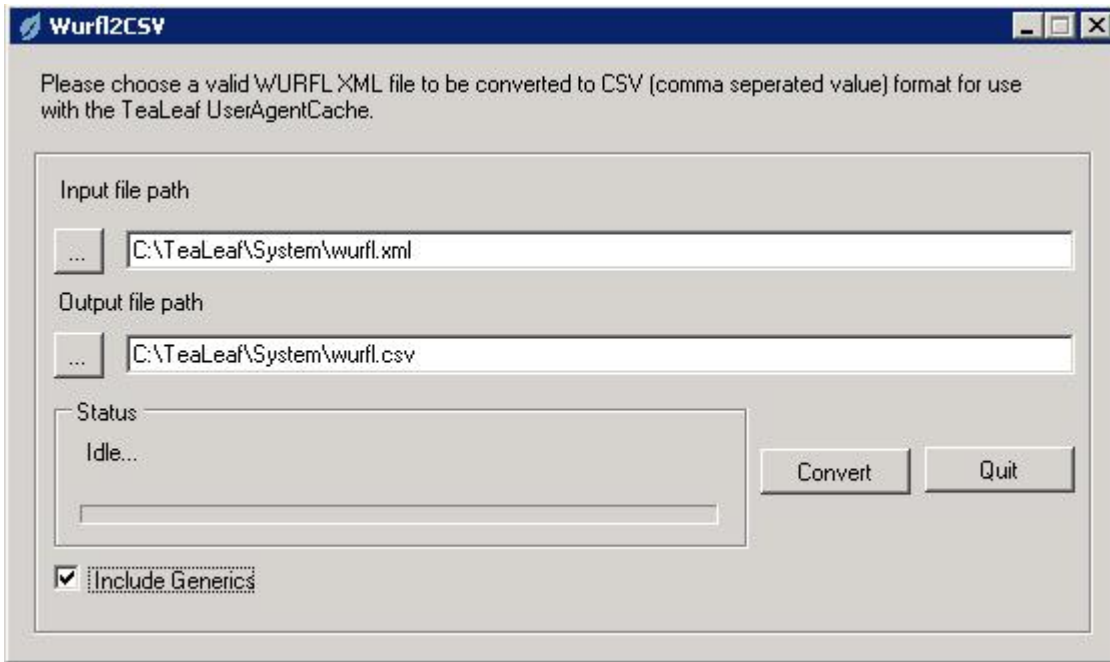


Figure 2. WURFL2CSV.exe

Procedure

1. Double-click WURFL2CSV . exe.
2. Select the downloaded WURFL . xml as the input file.
3. Provide an output file name or use the default WURFL . csv file name. Store the file in the location suggested by the application.
4. If you want to add generic versions of mobile user agents to the . csv version.
 - Generics are used as a fallback entry for partial matches. For example, if iPhone 10 is detected in the capture stream and no exact match exists in WURFL . csv, a generic user agent string (iPhone) can be used instead.
 - By default, use of generics is enabled.
5. To generate the . csv file, click **Convert**.
6. The file is generated and stored in the specified location.

Results

For more information on WURFL, see "Configuring User Agent Detection" in the *IBM Tealeaf cxImpact Administration Manual*. For more information on where to acquire WURFL, see "Maintaining the CX System" in the *IBM Tealeaf CX Installation Manual*.

Create User Agent Event Value Lists

In Release 8.5 and earlier, Tealeaf provided the UAValueListCreator . exe utility for the creation of value lists. This utility was replaced by gathering detected values from the logs for the dimensions where they are stored.

To begin gathering user agent information, you must add or configure the corresponding dimensions to gather and store values. These functions are performed through the Event Manager.

- You can use this method for capturing user agent information from the Browscap . csv standard for fixed user agents and from the WURFL standard for mobile user agents.

- For more information on the configuration files, see "Configuring User Agent Detection" in the *IBM Tealeaf cxImpact Administration Manual*.

For step-by-step instructions on how to create data objects to populate dimension value lists, see "E2E Scenario - Tracking User Agent Information in Tealeaf" in the *IBM Tealeaf cxImpact User Manual*.

Review configuration files

UserAgentRevealer.exe can be used to review the .csv configuration files for user agents.

See "Configuring User Agent Detection" in the *IBM Tealeaf cxImpact Administration Manual*.

User agent configuration files

You download, convert, and deploy the user agent detection files that are required for user agent detection, including mobile user agents. These files must be updated on a periodic basis.

The user agent files include:

File

Description

BrowsCap.csv

Converted public standard for detecting fixed user agents.

UserCap.csv

Local file that can be used to overwrite the public standard.

WURFL.csv

Converted public standard for detecting mobile user agents.

UserSupplement.csv

Local file that can be used to supplement the public standards with data on user agents you detect in the capture stream.

Maintain the CX system

There are several maintenance tasks that you should complete regularly to keep your Tealeaf solution updated and properly functioning. The required tasks are not limited to this list and this list does not replace any IT tasks that are required to maintain the databases, storage, servers, and infrastructure where your Tealeaf solution resides.

Cycling services

Tealeaf recommends cycling the services on all Processing Servers (Canisters) and Report Servers once per day during off-peak hours.

Cycling services on a nightly basis alleviates the following issues:

- On the Processing Server, Canister services use much memory. At the end of the day, memory that is used for the Short-Term Canister can be highly fragmented and therefore less efficient. Recycling the services automatically defragments the Canister memory and ensures consistent performance in the Short-Term Canister.
- On the Processing Server, cycling services process also runs scripts through TLTMaint.exe to verify the integrity of the Long-Term Canister. See ["Canister maintenance \(TLTMaint\)" on page 21](#).
- Cycling services flushes out any residual information about existing sessions and prepares the system for the next day.

Configure cycle services

By default, the Tealeaf Scheduling Service is configured to run a cycle services job on each Tealeaf server at 12:30am, local time.

- All Tealeaf servers can be cycled at the same time. The Processing Servers and Reporting Servers are the most important ones to cycle daily.

If you deployed a Health-Based Routing server in your system, configure cycle services on the Processing Servers so that the HBR always has a Processing Server available to which to send hits. Otherwise, data might be lost. For more information about HBR, see "Health-Based Routing (HBR) Session Agent" in the *IBM Tealeaf CX Configuration Manual*.

For more information about enabling and scheduling cycle services, see "Configuring the Scheduling Service" in the *IBM Tealeaf CX Configuration Manual*.

Canister maintenance (TLTMaint)

TLTMaint checks the stability of the Canister and its files. When a corruption is detected in any of the Canister data files, an attempt is made to correct the condition by rebuilding the corrupted table. If stability of the Canister cannot be maintained, TLTMaint does not allow the Canister services to start.

This information is provided for reference purposes only. TLTMaint rarely requires extra configuration after installation. Unless explicitly told to do so by Tealeaf personnel, you should not need to use or configure TLTMaint. For more information, contact Tealeaf <http://support.tealeaf.com>.

TLTMaint command-line options

TLTMaint is normally started by the Tealeaf Canister Server program but can also be run manually at the command line.

The command-line options are:

- -h Help
- -v Verbose output
- -CheckOnly Performs check on tables without fixing anything.
- -Noserver Recovers files without datastore (Canister/Archive) services running.

TLTMaint publishes its processing to the Windows application log. It also stores log files in the Tealeaf \Logs directory.

TLTMaint processing

On services start, TLTMaint performs several configuration checks.

The configuration checks include:

- Locates the installation path of Tealeaf by checking the Registry.
- Checks for the `ctsrvr.cfg` file.
- Loads its own CFG file (`TLTMaint.cfg`).
- Creates Canister spacer file. TLTMaint calculates spacer file size by reading the Max Ctree Bytes size from the Registry. Every time that it starts, it can modify the spacer file size that is based on these settings.
- Checks Canister size and files to index.
- Checks database version.
- Begins checking tables that are defined in the `TLTMaint.cfg`. If you want to skip the check of these tables, you can comment these tables out in `TLTMaint.cfg`. On bad Canister shutdowns, data files might be corrupted. An attempt is made to salvage the data file.
- Runs CanTrim. Checks the CanTrim setting to see whether it is the right time and day to start.
 - CanTrim is enabled through TMS. See "Configuring the CX Canister" in the *IBM Tealeaf CX Configuration Manual*.

Backup and restore

Tealeaf provides backup and restore facilities that can be used to maintain backups of Tealeaf session data.

See "TLBackup and TLRestore" in the *IBM Tealeaf cxImpact Administration Manual*.

Creating a shutdown script in Windows

You can create a shutdown script that runs a Tealeaf program that is called CanSvcs.exe before the computer shuts down. This program keeps the operating system running until all Tealeaf services are stopped, allowing in-process data to be saved and preventing the Canister from becoming corrupted.

About this task

When you shut down the IBM Tealeaf CX server, it is important to ensure all Tealeaf services were stopped before the server shuts down. If services are not stopped in time, the Tealeaf Canister can be corrupted and lose all stored Tealeaf data. Creating a shutdown script prevents this problem.

Procedure

1. Open a Group Policy as a stand-alone Microsoft Management Console (MMC) snap-in.
2. Assign a computer shutdown script.

Opening group policy

Open a group policy as a stand-alone MCC snap-in.

Procedure

1. Open the Microsoft Management Console (MMC) by clicking **Start > Run** and typing mmc in the **Open** field.
2. In the **Console** menu, select **Add/Remove Snap-in**.
3. In the Add/Remove Snap-in dialog, click **Add** and select **Group Policy** from the **Add Standalone Snap-in** dialog. Click **Add**.
4. In the Select Group Policy Object dialog, click **Browse** to find the Group Policy that you want. If you want to save the Group Policy console and be able to choose which Group Policy object opens in it from the command line, select the **Allow focus of the Group Policy Snap-in to be changed when launching from the command line** checkbox. Click **Finish** when completed.
5. Click **Close** in the **Add Standalone Snap-in** dialog and **OK** in the Add/Remove Snap-in dialog.

Assigning a computer

Assign a shutdown script to shut down the system.

Procedure

1. In the Console tree of Management Console, navigate to **Local Computer Policy > Computer Configuration > Windows Settings > Scripts (Startup/Shutdown)**.
2. In the details pane, double-click **Shutdown**.
3. In the Shutdown Properties dialog, click **Add**.
4. In the Add a Script dialog, enter the following information:
 - a) Click **Browse** beside the **Script Name** field and navigate to the following path:

```
\Program Files\TeaLeaf\CanSvcs.exe
```

- b) In the **Script Parameters** field, enter -stop.
 - c) Click **OK** twice.
5. Close the **MMC** window and save the console. By default, the file is saved here:

```
C:\Documents and Settings\<current user>\Start Menu\Programs\
Administrative Tools\ Console1.msc
```


WaitToKillServiceTimeout registry entry

The WaitToKillServiceTimeout registry entry determines how long the system waits for Tealeaf services to stop after the system notifies the service that the system is shutting down. This value is automatically set to 600,000 milliseconds (10 minutes) by the IBM Tealeaf CX installation program.

Set the registry entry with the **HKEY_LOCAL_MACHINE > SYSTEM > CurrentControlSet > Control > WaitToKillServiceTimeout** optional.

Updating user agent files

Tealeaf uses two publicly available standards for detecting user agents. Update the files to detect user agents.

Procedure

1. Update the BrowsCap.csv file to add or modify entries to detect fixed user agents.
2. Update the WURFL.csv to add or modifies entries to detect mobile user agents.
3. Update the UserCap.csv to supplement the contents of the BrowsCap.csv file with user agents that are not listed in the standard.

Update locations and servers

The user agent files must be stored on any Canister server or Health-Based Routing server where the Tealeaf Reference session agent is installed.

The user agent files must be stored in this location:

```
<Tealeaf_Install_Directory>\system
```

- To enable detection of fixed and mobile user agents, the Tealeaf Reference session agent must be installed in each Windows pipeline that processes session data. See "Tealeaf Reference Session Agent" in the *IBM Tealeaf CX Configuration Manual*.
- Health-Based Routing enables load balancing between multiple Canisters in a Tealeaf environment. See "Health-Based Routing (HBR) Session Agent" in the *IBM Tealeaf CX Configuration Manual*.

Updating BrowsCap.csv

When you first install Tealeaf, acquire the most recent version of BrowsCap.csv.

About this task

The BrowsCap.csv file contains definitions for various web-capable devices and is widely used in commercial applications.

Procedure

1. Go to <https://browscap.org/> and download the latest version of BrowsCap.csv to the IBM Tealeaf CX server.

BrowsCap.csv is available in various formats. Tealeaf supports only the .csv format of this file. On start, Tealeaf verifies that this file was not modified and rejects any modified versions. See "Managing User Agents" in the *IBM Tealeaf cxImpact Administration Manual*.

2. On the IBM Tealeaf CX server machine, save the file into the following directory:
<Tealeaf_Install_Directory>\system

As new browsers and web devices are released or updated, your version of BrowsCap.csv might require updating. Periodically check the website for updates.

3. Restart the Transport Service through the Tealeaf Management System.
See "TMS WorldView Tab" in the *IBM Tealeaf cxImpact Administration Manual*.

Update WURFL . csv

Periodically, the WURFL . csv file that is used by your Tealeaf solution must be updated with the most recent public standard. Then, the data must be prepared for use with the Tealeaf platform.

See "Maintenance for CX Mobile" in the *IBM Tealeaf CX Mobile Administration Manual*.

Maintenance for CX Mobile

Tealeaf uses several published standards. To keep the system current, update and review the standards files regularly.

The IBM Tealeaf CX Mobile module uses a publicly available standard for detecting mobile devices. The Wurf1 . xml (XML) file contains definitions for various mobile web-capable devices and is widely used in commercial applications.

Complete the maintenance tasks regularly to ensure that your Tealeaf solution is up to date with the most recent mobile user agent information. Complete the tasks steps in this order:

1. Updating the WURFL.
2. Generating User Agent values lists for mobile visitors.
3. Updating Mobile events with new User Agent templates.
4. Reviewing UserSupplement.csv.

Updating WURFL

Keep the WURFL updated to have the most recent mobile device definitions.

Procedure

1. When you first install Tealeaf, download the most recent version of WURFL . xml (XML):
<http://wurfl.sourceforge.net/>
2. Convert the file into . csv format for use with the Tealeaf system.
 - See "User Agent Tools" in the *IBM Tealeaf cxImpact Administration Manual*.
3. Save the file on the IBM Tealeaf CX server machine, into the following directory:
<Tealeaf_Install_Directory>\system
 - You can save this file to the server directory using the Tealeaf Management System. See "TMS Advanced Tab" in the *IBM Tealeaf cxImpact Administration Manual*.
4. Restart the Transport Service.
 - You can configure a task in TMS to restart specific servers and servers. See "TMS WorldView Tab" in the *IBM Tealeaf cxImpact Administration Manual*.

Generating user agent value lists for mobile visitors

Whenever you update WURFL, regenerate the user agent value lists that are required to track mobile visitors. After you downloaded or updated your WURFL . csv file, generate new value lists for mobile events.

If you imported the Mobile Dashboard, you do not need to generate these user value lists, as they are included as part of the import.

- See "User Agent Tools" in the *IBM Tealeaf cxImpact Administration Manual*.

Updating mobile events with new user agent value lists

When the value lists for user agent reporting are generated, import them for use by the IBM Tealeaf CX system.

Reviewing UserSupplement.csv

If you added user agents through the UserSupplement.csv, you might want to periodically review this file to remove references to agents that are now reflected in the public standard.

If a user agent appears in both the UserSupplement.csv file and one of the public standards, the definition in the public standard takes precedence. Removing the duplicated user agent definition from UserSupplement.csv is an optional housekeeping chore but is not required.

Tealeaf Reference Session Agent

The Tealeaf Reference session agent is deployed in the Windows pipeline to detect specific dimensional data and capture it for use by Tealeaf.

The Reference session agent can be used for these activities:

- User-agent detection - The Tealeaf Reference session agent can detect the type of user agent, browser, and operating system in use by your visitors. For more information, see "Extended User Agent Parsing" in the *IBM Tealeaf CX Configuration Manual*.
- Referrer parsing and prepending - The Tealeaf Reference session agent can be configured to normalize referrer values into name-value pairs for easier analysis and can prepend these values with a configured string to eliminate potential conflicts with other URL values. See ["Referrer Parsing" on page 28](#).
- Client-side capture references - The Tealeaf UI Capture capability can be deployed to capture client-side events and other data, including reference values. These values are detected by the Tealeaf Reference session agent. See "Client-Side Capture References" in the *IBM Tealeaf CX Mobile Administration Manual*.

For some deployments of the Processing Server, this session agent is included in the default pipeline and is required. See "CX Pipeline Configuration" in the *IBM Tealeaf CX Configuration Manual*.

Tutorial

For more information about how user agent information is used within Tealeaf, including steps to configure event objects to track user agent information, see "E2E Scenario - Tracking User Agent Information".

"E2E Scenario - Tracking User Agent Information in Tealeaf" is in the *IBM Tealeaf cxImpact User Manual*.

Add the session agent

Session agents can be added through the Pipeline Editor in TMS. This session agent is added to the default pipeline on each Canister. The session agent must be present in any pipeline that processes session hits.

See "Adding a Session Agent" in the *IBM Tealeaf CX Configuration Manual*.

For more information about the Pipeline Editor and TMS, see "Tealeaf Management System" in the *IBM Tealeaf cxImpact Administration Manual*.

The configuration options are changed with the TealeafCaptureSocket.cfg file that is stored on the server. These settings are also available through the Pipeline Editor, which is the recommended method for configuring session agents.

Client-Side Capture References

Tealeaf Reference session agent adds TLT_CUI_URL and TLT_CUI_APPLICATION_NAME values to the request buffer to track the URLs and application names from which events captured from the client user interface are generated.

These values are generated when the IBM TealeafCX UI Capture for AJAX is licensed and deployed in your environment to capture user interface events occurring in the visitor's browser (for more information, please contact your IBM Tealeaf representative.) For more information on IBM TealeafCX UI Capture for AJAX, see "UI Capture FAQ" in the *IBM Tealeaf UI Capture for AJAX FAQ*.

These values are generated like normal reference values, but they might not match the values in the normal members. The normal members represent the parent page from which the AJAX client-side events were started. Since the child event might post to any page, its value might not match the parent page.

- TLT_CUI_URL is rendered as the normalized value for the client-side URL.
- If the session agent is configured for overloading the application field, TLT_CUI_APPLICATION_NAME might be used in a different way. See [“Overloading and UI Capture”](#) on page 37.

Tealeaf Reference session agent also adds TLT_CUI_URL_ID and TLT_CUI_APPLICATION_NAME_ID values to the request buffer, which allows coupling of data from AJAX requests and the parent pages from which they are started. They follow the same scheme as the regular TLT IDs. All of their information comes from the same reference files, and they can be manipulated and augmented in the same way.

Path reference values

The hardest dimension to make meaningful is the Path dimension, which contains URL information. Dynamic content generation technologies can serve differing page content in response to requests for a path. Thus, URLs might differ only in their query string, POST data, or other HTTP request content.

Determine path values

The value of the Path reference dimension can be set by the Tealeaf Reference session agent or by any pipeline agent before it. The session agent provides the following three methods for setting this variable: Simple, Mapping, and Programming.

The *path* is defined as all characters in the URL from the first / after the host identifier through the character that precedes the first ?, if it is present. The following elements are not part of the path:

- Protocol identifier: http://
- Host: (for example, www.tealeaf.com)
- Port (for example, :80)

Simple

In the Simple case, the file name extension and HTTP status code checks done by Tealeaf Reference session agent are sufficient to limit or determine the virtual path values. This case implies the following conditions:

- The URL does not contain distinct session or hit identifiers. It does not carry any state information.
- Paths end with file name extensions
- The website software is homogeneous across all servers. Default page names (for example, index.html or default.asp) for URLs that end with / are identical for all web servers.

Mapping

The mapping technique can be used for the following cases:

- The path for multiple pages can be the same (for example, /page.cgi or /ISAPI.dll), but one or two query strings or other request variables can distinguish pages from one another.
 - For example, a code can be embedded in the query string to distinguish pages for the web application's own purposes, as in the catid value 700 in /page.asp?catid=700 signifies the **Product View** page).
 - The session agent's "default page" algorithm for virtual directories (for example, URL=/ or URL=/somedir/) use the mapping configuration file to set the TLT_URL value.
 - The mapping file has an initial entry that is created during the installation process. For example:

```
# TLT_URL URL ReqVar1 ReqVar2
/default.asp /
```

This example configuration assigns TLT_URL=/default.asp in the [appdata] section when / is the value of the URL variable in the Tealeaf request.

- Virtual directory start patterns determine validity of paths. All paths that conform to a specified start pattern are included, such as all paths that begin with /server/ or /support/customers/).

This method is more open-ended than the first mapping method and allows more junk path values to be accepted.

Programming

Programming, in the form of RTA rules or a custom pipeline agent, using TCL or Managed Code session agents, is required for cases that cannot be handled by the Simple or Mapping techniques. For example, if the URL value contains any type of application state or tracking information that should be stripped out to produce a good value for TLT_URL, the TLT_URL value must be set through a custom pipeline agent upstream of Tealeaf Reference session agent.

As performed by DoubleClick or Coremetrics for example, page tagging cannot be handled natively by the session agent as a source for the value of TLT_URL. You must create a custom pipeline agent and apply it before the session agent.

Order of precedence of Path processing methods

The URL processing precedence is as follows:

- URLReferenceRules
- URLReferenceVirtualDir
- NormalizeURLExt and NormalizeURLStatusCode

If URLReferenceRulesMode is set to STOP, then method 3 is not used to validate the URL, and Tealeaf Reference session agent performs strict interpretation of the URL.

- If the mode is CONT, then a combination of rules and status code and file name extension tests can be used.

An example of combining methods 1 and 3 would be for a site with multiple possible virtual directory names, most likely because the web servers are a heterogeneous mixture of IIS and Java™ Platform, Enterprise Edition. Use step 1 to determine the default file name for virtual directories. For all other types of URLs, use the normal extension/status code rules.

Example:

```
#TLT_URL URL ReqVar1 ReqVar2
/default.asp / IISSESSIONID
/page.jhtml / JSESSIONID
```

High-volume dimensions

Path values are used to populate the URL dimension that is provided by Tealeaf. Depending on how this dimension is configured, your database can grow without limit.

Data management of dimensions is especially important for dimensions that capture a high volume of values, such as URL. Implementation of specific instructions for managing URL and other high-volume dimensions can help to prevent runaway database growth.

- See "Managing URL and Other High-Volume Dimensions" in the *IBM Tealeaf Event Manager Manual*.

More reference parsing options

The Tealeaf Reference session agent can parse the HTTP_REFERER value in the REQ buffer into a more accessible set of name-value pairs. The Tealeaf Reference session agent manages the detection and population of user agent information that is extracted from the request buffer.

Referrer Parsing

The Tealeaf Reference session agent can parse the HTTP_REFERER value in the REQ buffer into a more accessible set of name-value pairs.

Consider the following example name-value pair:

```
HTTP_REFERER=[http://www.yourdomain.com/parent/first_child/second_child/
page.html?id=1234&req=25&page_id=2]
```

The [referrer] section in the REQ buffer looks like this:

```
[referrer]
REFERRER_DOMAIN=yourdomain.com
REFERRER_FILEPATH=/parent/first_child/second_child/page.html
req=25
page_id=2
id=1234
```

This configuration allows Tealeaf to generate events on and collect data for values within certain referring URLs of your choosing.

- First Hit: The referrer parsed from the first hit of a session is used as input for the session attribute that tracks referrer values. This value is the referrer for the session.
- Each Hit: Tealeaf Reference session agent also detects the referrer value for each hit in the session.

To enable referrer parsing in Tealeaf Reference session agent, set ReferrerParsing=True in the session agent configuration through TMS.

If ReferrerParsing is not enabled, the pattern Referrer Domain provided by Tealeaf does not capture any data.

Referrer prepending

To eliminate potential conflicts between referrers and URL fields, you can prepend values in the referrer section.

For example, add this configuration property to the Tealeaf Reference Session Agent configuration:

```
ReferrerPrepend=MY_PREPEND_
```

To prepend referrers, Referrer Parsing must be enabled (ReferrerParsing=True).

When **ReferrerParsing** is enabled, the output is displayed in the [Referrer] section of the request. For example:

```
[Referrer]
MY_PREPEND_REFERRER_DOMAIN=somewhere.somplace.com
MY_PREPEND_REFERRER_FILEPATH=/testytime/testytime2/file.dll
```

Extended User Agent Parsing

The Tealeaf Reference session agent manages the detection and population of user agent information that is extracted from the request buffer. When a user submits a request to the web application,

information about the visitor's browser, operating system, and type of application (mobile, desktop, BOT) is included in the request.

The information in the request is compared to a set of publicly maintained standards and overrides that you define to determine if the submitted user agent information matches known user agents.

- Initially, the Tealeaf Reference session agent examines a self-pruning cache that it maintains in memory for faster access to the mostly commonly detected values. See [“Tune the self-pruning cache” on page 32](#).
- If no match is found, the session agent examines other file-based resources for matches. See [“Order of evaluation for extended user agent parsing” on page 30](#).

Note: By default, user agent parsing is enabled for new installs and upgrades. Legacy Mode must be enabled separately. See [“Legacy Mode for user agent parsing” on page 36](#).

Example output

This example shows the output for a name-value pair in a request.

Suppose this user agent name-value pair is detected in the request:

```
HTTP_USER_AGENT=Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.2.13)
                Gecko/20101203 Firefox/3.6.13 ( .NET CLR 3.5.30729)
```

When the user agent is detected, the session agent performs lookups to return the values that are associated with this user agent from the public standards. These values are inserted as name-value pairs in the request:

```
[ExtendedUserAgent]
TLT_BROWSER=Firefox
TLT_BROWSER_VERSION=Firefox3.6
TLT_BROWSER_PLATFORM=Win XP
TLT_TRAFFIC_TYPE=BROWSER
TLT_BROWSER_JAVASCRIPT=true
TLT_BROWSER_COOKIES=true
```

If TLT_TRAFFIC_TYPE=MOBILE, then these fields can be inserted and populated for the mobile user agent:

```
TLT_SCREEN_HEIGHT
TLT_SCREEN_WIDTH
TLT_COLOR_DEPTH
TLT_PICTURE_SUPPORT
TLT_VIDEO_SUPPORT
TLT_STREAMING_SUPPORT
```

Additionally, if user agent logging is enabled, these entries can appear in the [UALog] section that is inserted into the request:

```
[UALog]
Retrieve From Cache took 9.029e-005 seconds
Checking TLT_USER_CAP
Selected class User Agent Parsing::Firefox User Agent Matcher
User Agent Parsing::Wurfl Data Manager::Apply Exact Match
m_user Agent Map.find took 0.000342962 seconds
User Agent Parsing::Wurfl Data Manager::Apply Exact Match
    took 0.000636579 seconds
User Agent Parsing::Browscap Data Manager::Apply Exact Match
m_exact Search Map.find took 0.000381108 seconds
Regex match: Mozilla/5\.0\s+\(Windows;\s+.*;\s+Windows\s+NT\
    s+5\.1;\s+.*;\s+rv:1\.9\.2.*\)
    \s+Gecko/.*/\s+Firefox/3\.6.*
m_regex Search Map lookup took 0.00878997 seconds
User Agent Parsing::Browscap Data Manager::Apply Exact Match
    took 0.00971492 seconds
User Agent Parsing::Filter::Get Data Node took 0.0113107 seconds
```

Order of evaluation for extended user agent parsing

When the Tealeaf Reference session agent detects a value in the HTTP_USER_AGENT request variable, the value is evaluated through a several step process.

The value evaluation steps are:

<i>Table 2. Order of evaluation for extended user agent parsing</i>		
Order of Evaluation	File name	Description
1	special non-valid values	If the HTTP_USER_AGENT contains blank or white space values, special values are inserted into the appropriate attributes. See “Blank user agent values” on page 31.
2	in-memory cache	Check for the value is made to the in-memory cache maintained by the session agent. See “Tune the self-pruning cache” on page 32.
3	UserCap.csv	User-defined values that override any values in the public standards (WURFL.csv and BrowsCap.csv) are specified in this file, which is checked before the public standards. <ul style="list-style-type: none"> • UserCap.csv may be used to insert values that are not present in the public standards. • These matches are identified using the UAMatchers DLLs. See “UAMatchers” on page 34.
4	WURFL.csv	Public standard for mobile devices. This file must be downloaded and converted from its native XML for use by the session agent. <ul style="list-style-type: none"> • During conversion, you may optionally include the use of generic values in your converted WURFL file. See “Use of generics” on page 32. • These matches are identified using the UAMatchers DLLs. See “UAMatchers” on page 34. • See "User Agent Tools" in the <i>IBM Tealeaf cxImpact Administration Manual</i>.

Table 2. Order of evaluation for extended user agent parsing (continued)		
Order of Evaluation	File name	Description
5	mobile native application	For mobile native applications monitored by Tealeaf, user agent information is automatically submitted by the Tealeaf client framework that is monitoring the application. This information is used to populate the standard extended user agent parsing properties. See “Extended user agent parsing for mobile native applications” on page 35.
6	BrowsCap.csv	Public standard for fixed and bot user agents. This file must be converted from its native format for use by the session agent. <ul style="list-style-type: none"> Generics are automatically included in the converted version of BrowsCap.csv. See “Use of generics” on page 32. See "User Agent Tools" in the <i>IBM Tealeaf cxImpact Administration Manual</i>.
7	UserSupplement.csv	If no match is found in any of the above files, the session agent checks the UserSupplement.csv, which can be used to add user agents that are not currently listed in the public standard and are appearing in your capture stream.

Blank user agent values

If the value of HTTP_USER_AGENT is a value that cannot be detected, then these values are inserted into the relevant attributes:

Table 3. BrowserType hit attribute:		
Detected Value	Attribute	Inserted Value
""	BrowserType	UserAgentBlank
whitespace	BrowserType	UserAgentBlank
HTTP_USER_AGENT not found in request	BrowserType	UserAgentNotFound

Table 4. TrafficType hit attribute:		
Detected Value	Attribute	Inserted Value
""	TrafficType	UserAgentHeaderIsBlank
whitespace	TrafficType	UserAgentBlank
HTTP_USER_AGENT not found in request	TrafficType	NoUserAgentHeaderFound

To search for TrafficType values:

- **Active sessions:** Search the Text in Request for TLT_TRAFFIC_TYPE=<Inserted Value>.

- **Completed sessions:**

- Create an event to record the value of the Traffic Type hit attribute. See "TEM Events Tab" in the *IBM Tealeaf Event Manager Manual*.
- Create a privacy rule to move the TLT_TRAFFIC_TYPE request variable and value to the [appdata] section. See "Privacy Session Agent" in the *IBM Tealeaf CX Configuration Manual*.

For more information about search, see "Searching Session Data" in the *IBM Tealeaf cxImpact User Manual*.

Tune the self-pruning cache

Depending on the traffic volume and the variety of traffic to your website, you might need to tweak the settings for the self-pruning cache. Finding an appropriate setting for these two parameters is paramount for optimum performance of your pipeline when you are using extended user agent parsing.

Pruning interval

You can define the interval at which the self-pruning cache is updated to reflect the most recently detected user agents. The PruningInterval setting is the number of seconds between updates.

- Adjust the setting after extended user agent parsing is in operation.

By changing the PruningInterval value of the cache, you are affecting how likely it is that a user agent is removed from the cache due to inactivity. When this value is large, a user agent can be in the cache for a long time before it is removed. When it is small, the user agent must be seen frequently, or it is more likely to be removed.

Maximum cache size

The MaxCacheSize setting is the maximum number of entries in the cache.

- Adjust the setting after extended user agent parsing is in operation.

For websites with a core set of user agents, the cache performs well with a relatively small MaxCacheSize setting. Set this value to 2000. This value size effectively means that up to 2000 different user agents are stored for quick lookup.

- If you begin to see spooling due to the user agent cache, increasing the MaxCacheSize value decreases the likelihood that an incoming user agent string is not already cached. Increasing this number also use more memory, as more data must be stored.

Use of generics

The public standards that are used for extended user agent parsing can contain generic values for user agents.

For example, if a user agent string of iPhone 10 is detected in the capture stream and there is no exact match in the public standard, a generic user agent string (iPhone) can be inserted, if the public standard contains a generic entry.

- BrowsCap.csv - For fixed user agents, use of generics is automatically enabled and cannot be configured.
- WURFL.csv - For mobile user agents, use of generics is enabled by default when the public standard is converted for use in Tealeaf. See "User Agent Tools" in the *IBM Tealeaf cxImpact Administration Manual*.

Enabling generics is useful for capturing in some form new versions of user agents that are not yet in the public standards. When the public standards are updated, the entries for these user agents match on the public standard without configuration changes. See ["Order of evaluation for extended user agent parsing" on page 30](#).

Specifying paths to skip

Optionally, you can configure extended user agent parsing to treat specified URLs that are triggered by specific user agents to be marked as a different traffic type. Complete this task to specify the paths to skip.

About this task

For example, suppose that for the user agent Jakarta, a visit to any page other than / MyXMLService.aspx indicates that the user agent is a search engine while visits to / MyXMLService.aspx indicate that the user agent is not a robot. By defining this path to be skipped during normal user agent parsing, you can specify the appropriate traffic type value for the non-robot exception (TrafficType=XMLServiceConsumer).

Procedure

1. Log in to the Tealeaf Portal as an administrator.
2. From the **Portal** menu, select **Tealeaf > TMS**. The Tealeaf Management System is displayed.
See "Tealeaf Management System" in the *IBM Tealeaf cxImpact Administration Manual*.
3. Click the **WorldView** tab.
4. From the Servers drop-down, select the server hosting the capture pipeline where the Tealeaf Reference session agent is deployed.
5. Click the **Transport Service** node.
6. Select **Transport Service configuration**.
7. In the Config Actions tab, click **View/Edit (Raw)**.
8. The raw version of the TealeafCaptureSocket.cfg file is displayed.
You can choose to copy and paste this content into your favorite text editor for editing.
9. In the file, locate the [TLTRef] section.
10. Look for the following configuration settings in the [TLTRef] section. If they are not present, insert them.

```
PathsToSkip=^MyPath*$;^YourPath*$  
TrafficTypeForSkippedPaths=XMLService  
BrowserForSkippedPaths=XMLServiceConsumer
```

where

Setting

Description

PathsToSkip

Enter a list of paths, delimited by semi-colons (;). You can enter regular expressions. For example, you can enter a value of ^MyPath*\$ to match all paths that contain the substring MyPath. For more information about regular expressions, see [“Regular expressions in extended user agent parsing”](#) on page 34.

TrafficTypeForSkippedPaths

For paths that match a value that is specified in PathsToSkip, this setting identifies the value to insert into the Traffic Type object. See "Configuring User Agent Events" in the *IBM Tealeaf cxImpact Administration Manual*.

BrowserForSkippedPaths

For paths that match a value that is specified in PathsToSkip, this setting identifies the value to insert into the Browser Type object. See "Configuring User Agent Events" in the *IBM Tealeaf cxImpact Administration Manual*.

11. To save your changes, click **Save**.
12. To push the change to other Canisters in your environment, click **Add Tasks and Submit**.

Regular expressions in extended user agent parsing

You can use regular expressions in the values you define in user agent parsing configuration.

This table lists exceptions to the standard implementation of regular expressions:

Character	How Used
*	Wildcard
?	Wildcard
.	String literal. Appears frequently in user agent definitions.
(String literal. Appears frequently in user agent definitions.
)	String literal. Appears frequently in user agent definitions.
Space character	Treated as white space.

Because of these exceptions and some other differences, the implementation of regular expressions for this session agent and the IBM Tealeaf CX RealTea Viewer are not the same. See "Regular Expressions in the RealTea Viewer" in the *IBM Tealeaf RealTea Viewer User Manual*.

This table shows some examples and how they match values in user agent parsing.

RegEx pattern	Matches
^IP\$	IP SHIP SHIPS
S.P	SP SIP SIT UP
S?P	SIP SAP SOP
S*P	SIP SAP STOP

UAMatchers

Matching between detected values and the available set of user agent values is managed by plug-in. For each major type of user agent, Tealeaf provides a separate .dll that performs the matching analysis. UAMatchers plug-ins are provided by Tealeaf. Customers cannot create their own plug-ins now.

When user agent data is not available in the public standard definition files, a UA Matcher can augment the user agent information that is recorded in the session with additional information. The additional information includes browser version or browser platform, that are read directly from the user agent string that is provided in the original request.

- Currently, the Browscap standard does not distinguish between the WinXP 64-bit platform and the Windows Server 2003 64-bit platform, as user agent strings from both platforms self-identify as Windows NT 5.2. By default, Browscap identifies this user agent string as a WindowsXP 64-bit machine.

UAMatchers are stored in this directory:

```
<Tealeaf_install_directory>\System\UAMatchers
```

Do not move, rename, or otherwise modify these files.

Extended user agent parsing for mobile native applications

Individual mobile applications typically send a unique user agent string that is not known to any public repository, or they might not send a user agent at all. As a result, user agent detection for mobile native applications cannot rely on WURFL, Browscap, or other public standard.

To enable tracking of user agent-related information, the Tealeaf client frameworks for mobile native applications submit this header information:

Table 5. Mobile native applications header information	
Logging Framework	Example Submitted Header
Android Logging Framework	HTTP_X_TEALEAF=device (Android) Lib/0.0.10
iOS Logging Framework	HTTP_X_TEALEAF=device (iOS) Lib/8.5.4.1

The header is also used by the IBM Tealeaf CX UI Capture for AJAX solution. However, user agent detection for IBM Tealeaf UI Capture is sourced from the client used in the session. See "Client-Side Capture References" in the *IBM Tealeaf CX Mobile Administration Manual*.

When the Tealeaf Reference session agent detects the HTTP_X_TEALEAF header and either the (android) or (ios) value, it does these tasks:

- Tealeaf Reference session agent overwrites the request variable for traffic type:

```
TLT_TRAFFIC_TYPE=MOBILE_APP
```

- This value is eventually surfaced as the value MOBILE_APP in the Traffic Type dimension. See "TEM Dimensions Tab" in the *IBM Tealeaf Event Manager Manual*.

- Tealeaf Reference session agent overwrites the request variable for browser platform:

- Android:

```
TLT_BROWSER_PLATFORM=Android
```

- iOS:

```
TLT_BROWSER_PLATFORM=iOS
```

- Tealeaf Reference session agent looks for the HTTP_X_TEALEAF_PROPERTY header, which is submitted from the client frameworks with extended user agent information. For example:

```
HTTP_X_TEALEAF_PROPERTY=TLT_BROWSER=StraussAndPlesser Native;  
TLT_BROWSER_VERSION=8.5; TLT_BROWSER_PLATFORM=Android;  
TLT_BRAND=Asus; TLT_MODEL=Asus Eee Pad Transformer TF101;
```

```
TLT_SCREEN_HEIGHT=800; TLT_SCREEN_WIDTH=1280;  
TLT_COLOR_DEPTH=65536
```

- These values are used to populate the [ExtendedUserAgent] section. For example:

```
[ExtendedUserAgent]  
TLT_BROWSER=StraussAndPlessner Native  
TLT_BROWSER_VERSION=8.5  
TLT_BROWSER_PLATFORM=Android  
TLT_BRAND=Asus  
TLT_MODEL=Asus Eee Pad Transformer TF101  
TLT_SCREEN_HEIGHT=800  
TLT_SCREEN_WIDTH=1280  
TLT_COLOR_DEPTH=65536
```

These [ExtendedUserAgent] fields are not populated with data from the mobile native applications, as the fields are applicable to browsers only:

```
TLT_BROWSER_JAVASCRIPT  
TLT_BROWSER_COOKIES  
TLT_PICTURE_SUPPORT  
TLT_VIDEO_SUPPORT  
TLT_STREAMING_SUPPORT
```

Some of the variables can be inserted with false values if the request itself includes a user agent. After these fields were inserted into the [ExtendedUserAgent] section, the fields and properties of HTTP_X_TEALEAF_PROPERTY are applied.

Legacy Mode for user agent parsing

In Release 7.2.12.729x, Tealeaf introduced a new implementation of user agent parsing that provides much more accurate matching of user agents that are detected in the capture stream. If you want, you can enable the previous implementation of user agent parsing as a Legacy Mode.

Tealeaf strongly recommends that you use the most recent implementation of user agent parsing. Legacy Mode user agent parsing is provided for customers who upgrade from versions that supported the previous implementation of user agent parsing yet cannot switch to the new implementation now. In a future release, Legacy Mode user agent parsing is likely to be deprecated.

For more information on Legacy Mode, see "Tealeaf Reference Session Agent - Legacy Mode" in the *IBM Tealeaf CX Configuration Manual*.

Overloading application reference field

As of Release 8.0, the ability to capture the type of user agent is better handled by creating a dimension that is populated by a pattern that detects the value directly from the request. You disable this feature if it was enabled and design a dimension to capture the content. In a future release, this capability is likely to be deprecated.

About this task

See "TEM Dimensions Tab" in the *IBM Tealeaf Event Manager Manual*.

Optionally, you can configure the Tealeaf Reference session agent to replace the default contents that are inserted in the TLT_APPLICATION_NAME field of the [appdata] section of the request.

When the OverloadAppRef option is set to True, then the value of TLT_APPLICATION_NAME is overridden with the value for the traffic type that is extracted from the user agent cache. Possible values are as follows:

- BOT
- MOBILE
- BROWSER
- MOBILE_BOT
- UNKNOWN

When the Application reference field is overloaded, the only identifiers visible to the Canister are any associated with the traffic type values.

Procedure

1. Optional: If you are configuring with the `TealeafCaptureSocket.cfg` file, in the [TLTRef] of `TealeafCaptureSocket.cfg`, set `OverloadAppRef=true`.
2. If you are configuring through TMS:
 - a) Add the session agent to your pipeline through the Pipeline Editor. See "TMS Pipeline Editor" in the *IBM Tealeaf cxImpact Administration Manual*.
 - b) Select the **TLTRef session agent** icon in the Pipeline Editor and select **Edit**.
 - c) Select **Add Config Items**.
 - d) Select the `Overload Application Reference Field` check box.
 - e) Select **OK** twice.
 - f) Select **Save**.
 - g) Configure a task to update all servers immediately.

Overloading and UI Capture

If UI Capture is deployed when the **Application reference** field is overloaded, then the value of `TLT_APPLICATION_NAME` extracted from user agent information is also written to the `TLT_CUI_APPLICATION_NAME` value, which is inserted by the session agent when the `HTTP_X_TEALEAF_PAGE_URL` is detected in the request.

`HTTP_X_TEALEAF_PAGE_URL` is inserted by UI Capture. See "UI Capture FAQ" in the *IBM Tealeaf UI Capture for Ajax FAQ*.

By default, this value contains the URL path of the parent page. For more information about configuring this header, see "UI Capture for Ajax Reference" in the *IBM Tealeaf UI Capture for Ajax Guide*.

In the table below, you can review the mapping of destination request variables and how they are populated by data when overloading is disabled and enabled.

Table 6. Overloading and UI Capture		
Destination Request Variable	Standard Request Source	Overloaded Request Source
<code>TLT_CUI_APPLICATION_NAME</code>	<code>HTTP_X_TEALEAF_PAGE_URL</code> header	If overloaded: <code>TLT_TRAFFIC_TYPE</code>
<code>TLT_APPLICATION_NAME</code>	<code>[env]/URL</code>	If overloaded: <code>TLT_TRAFFIC_TYPE</code>

Both application variables in the request contain the traffic type information that is extracted for the detected user agent.

The value for `TLT_CUI_URL` is set by the session agent and is not affected by overloading the application field. See "Client-Side Capture References" in the *IBM Tealeaf CX Mobile Administration Manual*.

Configuration settings

Several configuration settings are available for the Tealeaf Reference session agent.

`Display Name` values are displayed in TMS, which is the recommended method for configuring session agents. See "Tealeaf Management System" in the *IBM Tealeaf cxImpact Administration Manual*.

`Name` values are displayed in `TealeafCaptureSocket.cfg`.

This table lists and describes the configuration settings:

Table 7. Configuration Settings		
Display Name	Name	Description
Advanced User Agent Parsing	AdvancedUAParsing	When set to True, the Tealeaf Reference session agent uses the self-pruning cache to speed up lookups of user agent information. This cache contains the most recently seen user agents. The default value is True. See "Configuring User Agent Detection" in the <i>IBM Tealeaf cxImpact Administration Manual</i> .
Allow Empty Extensions	AllowEmptyExtension	When True, URLs that do not contain an extension are permitted to be normalized. The default value is True.
Days to Keep Stats	StatsKeepDays	The number of days to retain statistics on the Tealeaf Reference session agent in rolling log files. The default value is 30.
Log Reference Statistics	OutputReferenceStats	When set to True, statistics on the Tealeaf Reference session agent operations are generated each hour for reporting through the Statistics Logger session agent. The default value is True.
Maximum Cache Size	MaxCacheSize	The maximum number of entries in the cache. The default value is 2000. Note: If Advanced User Agent Parsing is enabled, this value must be specified.
Normalize App Name	NormalizeAppName	When set to True, the value of TLT_APPLICATION_NAME is set to the virtual directory defined as the value between the first two slashes of the URL. The default value is True.
Normalize Host	NormalizeHost	When set to True, extra data manipulation is performed to normalize host values before they are saved as reference IDs. The default value is True.
Normalize Server	NormalizeServer	When set to True, TLT_SERVER is set to the value of LOCAL_ADDR. The default value is True.

Table 7. Configuration Settings (continued)

Display Name	Name	Description
Normalize URL	NormalizeURL	<p>Only URLs with the extension and status codes that are listed under NormalizeURLExt are considered.</p> <ul style="list-style-type: none"> • If the URL ends in a slash, the default root is appended, and the URL is decoded. • If the URL contains one of the characters (/ , : , ; , ?), the URL is truncated at that character. • The default value is True.
Pruning Interval	PruningInterval	<p>The interval in minutes when the user agent cache is updated with the most recently seen user agents. The default value is 10.</p> <p>Note: If Advanced User Agent Parsing is enabled, this value must be specified.</p>
Reference Statistics Min Hits	OutputReferenceStatsMin	<p>Minimum number of hits during a statistic log interval to qualify the reference statistics for logging. The default value is 1.</p>
Reference Update Interval	UpdateInterval	<p>Interval in seconds at which the references files are updated or read. The default value is 60.</p>
Referrer Parsing	ReferrerParsing	<p>When set to True, referrer values are parsed and normalized. The default value is True. See “Referrer Parsing” on page 28.</p>
Statistics Reporting Interval	StatsReportingInterval	<p>The interval in minutes at which the session agent reports statistics. The following values are accepted: 5, 10, 15, 30, or 60.</p> <p>Note: This interval is relative to the start of the hour. For example, a value of 30 generates statistics at 12:00, 12:30, and so on.</p>
Status Codes	NormalizeURLStatusCode	<p>The list of qualifying URL status codes that can be normalized. The default value is 0;200;302;304;400;402;403;404;410;500;501;502;503;504;505.</p>

Table 7. Configuration Settings (continued)

Display Name	Name	Description
URL Default Document	NormalizeURLRootDefault	If the URL indicates that the resource is a directory, then this value is inserted as the default file name. The default value is DEFAULTPAGE.
URL Extensions	NormalizeURLExt	The semi-colon delimited list of URLs to be normalized. The default value is ACTION; ASMX; ASP; ASPX; CSS; D0; HTM; HTML; GIF; ICO; JPG; JS; JSP; JHTML; PDF; PHP; SWF.
URL Reference Rules	URLReferenceRules	Whether URL reference rules are enabled. The default value is true.
URL Reference Rules File	URLReferenceRulesFile	The path and file name of the URL reference rules. The default value is .\system\Reference_Path_Rules.txt. <ul style="list-style-type: none"> Relative paths are relative to the directory that contains the DLL in use.
URL Reference Rules Mode	URLReferenceRulesMode	The mode for URL reference rules. The default value is cont. <ul style="list-style-type: none"> stop - URL processing ends after the rule set or virtual directory fails to match. Processing always stops after a matched rule set or virtual directory. cont - URL processing continues even if a failure condition is met.
Use Full Virtual Dir	UseFullVirtualDir	If set to True, matches are made for the entire virtual directory string, even if it includes a slash (for example, myapp\subdir). If set to False, matches are determined by the URLReferenceVirtualDir setting in use. The default value is False. <ul style="list-style-type: none"> See “Reference virtual directories” on page 42.

Table 7. Configuration Settings (continued)		
Display Name	Name	Description
URL Virtual Dirs	URLReferenceVirtualDir	<p>A list of URL virtual directories, which are separated by commas and without spaces. The default value is <code>servlets</code>. Example:</p> <pre>servlets;cgi-bin</pre> <ul style="list-style-type: none"> Other options are available. See “Reference virtual directories” on page 42.
User Agent Files Directory	UAFilesDir	<p>This directory contains all required user agent parsing files:</p> <ul style="list-style-type: none"> <code>BrowsCap.csv</code> for desktop user agent detection <code>WURFL.csv</code> for mobile user agents <code>UserCap.csv</code> for any user-defined user agent detection strings <p>Note:</p> <ul style="list-style-type: none"> If Advanced User Agent Parsing is enabled, this value must be specified. This value is a full path to the directory. Do not use a relative path. You can specify this path as a UNC path. <ul style="list-style-type: none"> See "Configuring User Agent Detection" in the <i>IBM Tealeaf cxImpact Administration Manual</i>.

This table lists and describes more configuration items:

Table 8. More Configuration Items:		
Display Name	Name	Description
Log URL Statistics	OutputURLStats	This debugging tool can be used to output URL statistics. The default value is <code>False</code> .
Overload Application Reference Field	OverloadAppRef	When set to <code>True</code> , the value of <code>TLT_APPLICATION_NAME</code> in the request buffer is overridden by the value for the type of browser that is extracted from the user agent cache. See “Overloading application reference field” on page 36.

Reference virtual directories

You can specify all or specific parts of the URL to use as the virtual directory.

To use the entire URL, set `UseFullVirtualDir` to `True`.

To use part of the URL, set `UseFullVirtualDir` to `True`. You can determine the part of the URL to use as the virtual directory by setting one of the following parameters to `True`.

Suppose that this URL is encountered:

```
/dir/sub-dir/sub-sub-dir/page.html
```

- If `URLReferenceVirtualDir=true`, then the virtual directory is:

```
dir
```

- If `URLReferenceVirtualDir2=true`, then the virtual directory is:

```
sub-dir
```

- If `URLReferenceVirtualDir3=true`, then the virtual directory is:

```
sub-sub-dir
```

Any setting for `URLReferenceVirtualDir4` and above is ignored.

Tealeaf reference session agent in multi-canister environments

The session agent should be deployed in a consistent location in each Windows pipeline of each Canister that is processing hits. In a multi-canister environment, all canisters can communicate independently with the database to retrieve reference values.

Tealeaf session agents

- "Adding a Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Archive Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Attribute Indexing Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Canister Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Cookie Parser Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Data Drop Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Data Parser Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Decouple Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Extended Decoupler Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Extended Privacy Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Health-Based Routing (HBR) Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Inflate Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "JSON Mobile Parser Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Managed Code Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Null Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Privacy Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Real-Time Monitoring and Alert (RTA) Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Response Tags to Request Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "RTA Split Session Agent" in the *IBM Tealeaf CX Configuration Manual*

- "Sessioning Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Session Router Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Socket Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "Statistics Logger Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- Tealeaf Reference Session Agent
- "Tealeaf Sessioning Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "TimeGrades Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "TLI Session Agent" in the *IBM Tealeaf CX Configuration Manual*
- "URL Decode Session Agent" in the *IBM Tealeaf CX Configuration Manual*

IBM Tealeaf documentation and help

IBM Tealeaf provides documentation and help for users, developers, and administrators.

Viewing product documentation

All IBM Tealeaf product documentation is available at the following website:

[Tealeaf Customer Experience Support](#)

Use the information in the following table to view the product documentation for IBM Tealeaf:

Table 9. Getting help	
To view...	Do this...
Product documentation	On the IBM Tealeaf portal, go to ? > Product Documentation .
IBM Tealeaf Knowledge Center	On the IBM Tealeaf portal, go to ? > Product Documentation and select <i>IBM Tealeaf Customer Experience in the ExperienceOne Knowledge Center</i> .
Help for a page on the IBM Tealeaf Portal	On the IBM Tealeaf portal, go to ? > Help for This Page .
Help for IBM Tealeaf CX PCA	On the IBM Tealeaf CX PCA web interface, select Guide to access the <i>IBM Tealeaf CX PCA Manual</i> .

Available documents for IBM Tealeaf products

The following table is a list of available documents for all IBM Tealeaf products:

Table 10. Available documentation for IBM Tealeaf products

IBM Tealeaf products	Available documents
IBM Tealeaf CX	<ul style="list-style-type: none"> • <i>IBM Tealeaf Customer Experience Overview Guide</i> • <i>IBM Tealeaf CX Client Framework Data Integration Guide</i> • <i>IBM Tealeaf CX Configuration Manual</i> • <i>IBM Tealeaf CX Cookie Injector Manual</i> • <i>IBM Tealeaf CX Databases Guide</i> • <i>IBM Tealeaf CX Event Manager Manual</i> • <i>IBM Tealeaf CX Glossary</i> • <i>IBM Tealeaf CX Installation Manual</i> • <i>IBM Tealeaf CX PCA Manual</i> • <i>IBM Tealeaf CX PCA Release Notes</i>
IBM Tealeaf CX	<ul style="list-style-type: none"> • <i>IBM Tealeaf CX RealTime Viewer Client Side Capture Manual</i> • <i>IBM Tealeaf CX RealTime Viewer User Manual</i> • <i>IBM Tealeaf CX Release Notes</i> • <i>IBM Tealeaf CX Release Upgrade Manual</i> • <i>IBM Tealeaf CX Support Troubleshooting FAQ</i> • <i>IBM Tealeaf CX Troubleshooting Guide</i> • <i>IBM Tealeaf CX UI Capture j2 Guide</i> • <i>IBM Tealeaf CX UI Capture j2 Release Notes</i>
IBM Tealeaf cxImpact	<ul style="list-style-type: none"> • <i>IBM Tealeaf cxImpact Administration Manual</i> • <i>IBM Tealeaf cxImpact User Manual</i> • <i>IBM Tealeaf cxImpact Reporting Guide</i>
IBM Tealeaf cxConnect	<ul style="list-style-type: none"> • <i>IBM Tealeaf cxConnect for Data Analysis Administration Manual</i> • <i>IBM Tealeaf cxConnect for Voice of Customer Administration Manual</i> • <i>IBM Tealeaf cxConnect for Web Analytics Administration Manual</i>
IBM Tealeaf cxOverstat	<i>IBM Tealeaf cxOverstat User Manual</i>
IBM Tealeaf cxReveal	<ul style="list-style-type: none"> • <i>IBM Tealeaf cxReveal Administration Manual</i> • <i>IBM Tealeaf cxReveal API Guide</i> • <i>IBM Tealeaf cxReveal User Manual</i>
IBM Tealeaf cxVerify	<ul style="list-style-type: none"> • <i>IBM Tealeaf cxVerify Installation Guide</i> • <i>IBM Tealeaf cxVerify User's Guide</i>
IBM Tealeaf cxView	<i>IBM Tealeaf cxView User's Guide</i>

Table 10. Available documentation for IBM Tealeaf products (continued)

IBM Tealeaf products	Available documents
IBM Tealeaf CX Mobile	<ul style="list-style-type: none"> • <i>IBM Tealeaf CX Mobile Android Logging Framework Guide</i> • <i>IBM Tealeaf Android Logging Framework Release Notes</i> • <i>IBM Tealeaf CX Mobile Administration Manual</i> • <i>IBM Tealeaf CX Mobile User Manual</i> • <i>IBM Tealeaf CX Mobile iOS Logging Framework Guide</i> • <i>IBM Tealeaf iOS Logging Framework Release Notes</i>

Index

A

administration [5](#), [8](#), [13](#), [18](#), [20](#), [24](#)
agent [25](#)

B

bot [8](#), [13](#)
bots [8](#), [13](#)
browscap [8](#), [13](#), [20](#)

C

CanTrim [20](#)
configuration [1](#), [25](#)
crawler [13](#)
CX Mobile [1](#), [5](#), [8](#), [13](#), [20](#)
cxImpact [8](#), [13](#), [18](#)

D

data segmentation [20](#)
detection [8](#)

E

extended [25](#)

F

fixed [13](#)

M

maintenance [20](#), [24](#)
mobile [5](#), [13](#), [24](#)

O

overview [1](#), [8](#)

R

report template [8](#)

S

session agent [25](#)
stripper [13](#)

T

TLTRef [25](#)

U

user [25](#)
user agent [8](#), [13](#), [18](#), [20](#)
user agent revealer [13](#)
user supplement [13](#)
usercap [13](#), [20](#)

W

WURFL [5](#), [8](#), [13](#), [20](#), [24](#)

