

Page Domain

Actions and events related to the inspected page belong to the page domain.

Methods

[Page.addScriptToEvaluateOnNewDocument](#)
[Page.bringToFront](#)
[Page.captureScreenshot](#)
[Page.createIsolatedWorld](#)
[Page.disable](#)
[Page.enable](#)
[Page.getAppManifest](#)
[Page.getFrameTree](#)
[Page.getLayoutMetrics](#)
[Page.getNavigationHistory](#)
[Page.handleJavaScriptDialog](#)
[Page.navigate](#)
[Page.navigateToHistoryEntry](#)
[Page.printToPDF](#)
[Page.reload](#)
[Page.removeScriptToEvaluateOnNewDocument](#)
[Page.resetNavigationHistory](#)
[Page.setDocumentContent](#)
[Page.stopLoading](#)
[Page.clearGeolocationOverride](#) Deprecated
[Page.setGeolocationOverride](#) Deprecated
[Page.addCompilationCache](#) Experimental
[Page.captureSnapshot](#) Experimental
[Page.clearCompilationCache](#) Experimental
[Page.close](#) Experimental
[Page.crash](#) Experimental
[Page.generateTestReport](#) Experimental
[Page.getAdScriptId](#) Experimental
[Page.getAppId](#) Experimental
[Page.getInstallabilityErrors](#) Experimental
[Page.getOriginTrials](#) Experimental
[Page.getPermissionsPolicyState](#) Experimental
[Page.getResourceContent](#) Experimental
[Page.getResourceTree](#) Experimental
[Page.produceCompilationCache](#) Experimental
[Page.screencastFrameAck](#) Experimental
[Page.searchInResource](#) Experimental
[Page.setAdBlockingEnabled](#) Experimental
[Page.setBypassCSP](#) Experimental
[Page.setFontFamilies](#) Experimental
[Page.setFontSizes](#) Experimental
[Page.setInterceptFileChooserDialog](#) Experimental
[Page.setLifecycleEventsEnabled](#) Experimental
[Page.setPrerenderingAllowed](#) Experimental
[Page.setRPHRegistrationMode](#) Experimental
[Page.setSPCTransactionMode](#) Experimental
[Page.setWebLifecycleState](#) Experimental

[Page.startScreencast](#) Experimental
[Page.stopScreencast](#) Experimental
[Page.waitForDebugger](#) Experimental
[Page.addScriptToEvaluateOnLoad](#) ExperimentalDeprecated
[Page.clearDeviceMetricsOverride](#) ExperimentalDeprecated
[Page.clearDeviceOrientationOverride](#) ExperimentalDeprecated
[Page.deleteCookie](#) ExperimentalDeprecated
[Page.getCookies](#) ExperimentalDeprecated
[Page.getManifestIcons](#) ExperimentalDeprecated
[Page.removeScriptToEvaluateOnLoad](#) ExperimentalDeprecated
[Page.setDeviceMetricsOverride](#) ExperimentalDeprecated
[Page.setDeviceOrientationOverride](#) ExperimentalDeprecated
[Page.setDownloadBehavior](#) ExperimentalDeprecated
[Page.setTouchEmulationEnabled](#) ExperimentalDeprecated

Events

[Page.domContentEventFired](#)
[Page.fileChooserOpened](#)
[Page.frameAttached](#)
[Page.frameDetached](#)
[Page.frameNavigated](#)
[Page.interstitialHidden](#)
[Page.interstitialShown](#)
[Page.javascriptDialogClosed](#)
[Page.javascriptDialogOpening](#)
[Page.lifecycleEvent](#)
[Page.loadEventFired](#)
[Page.windowOpen](#)
[Page.frameClearedScheduledNavigation](#) Deprecated
[Page.frameScheduledNavigation](#) Deprecated
[Page.backForwardCacheNotUsed](#) Experimental
[Page.compilationCacheProduced](#) Experimental
[Page.documentOpened](#) Experimental
[Page.frameRequestedNavigation](#) Experimental
[Page.frameResized](#) Experimental
[Page.frameStartedLoading](#) Experimental
[Page.frameStoppedLoading](#) Experimental
[Page.navigatedWithinDocument](#) Experimental
[Page.screencastFrame](#) Experimental
[Page.screencastVisibilityChanged](#) Experimental
[Page.downloadProgress](#) ExperimentalDeprecated
[Page.downloadWillBegin](#) ExperimentalDeprecated

Types

[Page.AppManifestError](#)
[Page.DialogType](#)
[Page.Frame](#)
[Page.FrameId](#)
[Page.FrameTree](#)
[Page.LayoutViewport](#)
[Page.NavigationEntry](#)

[Page.ScriptIdentifier](#)
[Page.TransitionType](#)
[Page.Viewport](#)
[Page.VisualViewport](#)
[Page.AdFrameExplanation](#) Experimental
[Page.AdFrameStatus](#) Experimental
[Page.AdFrameType](#) Experimental
[Page.AdScriptId](#) Experimental
[Page.AppManifestParsedProperties](#) Experimental
[Page.AutoResponseMode](#) Experimental
[Page.BackForwardCacheNotRestoredExplanation](#) Experimental
[Page.BackForwardCacheNotRestoredExplanationTree](#) Experimental
[Page.BackForwardCacheNotRestoredReason](#) Experimental
[Page.BackForwardCacheNotRestoredReasonType](#) Experimental
[Page.ClientNavigationDisposition](#) Experimental
[Page.ClientNavigationReason](#) Experimental
[Page.CompilationCacheParams](#) Experimental
[Page.CrossOriginIsolatedContextType](#) Experimental
[Page.FontFamilies](#) Experimental
[Page.FontSizes](#) Experimental
[Page.FrameResource](#) Experimental
[Page.FrameResourceTree](#) Experimental
[Page.GatedAPIFeatures](#) Experimental
[Page.InstallabilityError](#) Experimental
[Page.InstallabilityErrorArgument](#) Experimental
[Page.NavigationType](#) Experimental
[Page.OriginTrial](#) Experimental
[Page.OriginTrialStatus](#) Experimental
[Page.OriginTrialToken](#) Experimental
[Page.OriginTrialTokenStatus](#) Experimental
[Page.OriginTrialTokenWithStatus](#) Experimental
[Page.OriginTrialUsageRestriction](#) Experimental
[Page.PermissionsPolicyBlockLocator](#) Experimental
[Page.PermissionsPolicyBlockReason](#) Experimental
[Page.PermissionsPolicyFeature](#) Experimental
[Page.PermissionsPolicyFeatureState](#) Experimental
[Page.ReferrerPolicy](#) Experimental
[Page.ScreencastFrameMetadata](#) Experimental
[Page.ScriptFontFamilies](#) Experimental
[Page.SecureContextType](#) Experimental

Methods

Page.addScriptToEvaluateOnNewDocument #

Evaluates given script in every frame upon creation (before loading frame's scripts).

parameters

source

string

worldName

string

If specified, creates an isolated world with the given name and evaluates given script in it. This world name will be used as the `ExecutionContextDescription::name` when the corresponding event is emitted.

Experimental
`includeCommandLineAPI`
 boolean

Specifies whether command line API should be available to the script, defaults to false.

Experimental
`runImmediately`
 boolean

If true, runs the script immediately on existing execution contexts or worlds. Default: false.

Experimental

Return Object

identifier
[ScriptIdentifier](#)

Identifier of the added script.

Page.bringToFront

Brings page to front (activates tab).

Page.captureScreenshot

Capture page screenshot.

parameters

format
 string

Image compression format (defaults to png).

Allowed Values: jpeg, png, webp

quality
 integer

Compression quality from range [0..100] (jpeg only).

clip
[Viewport](#)

Capture the screenshot of a given region only.

fromSurface
 boolean

Capture the screenshot from the surface, rather than the view. Defaults to true.

Experimental
captureBeyondViewport
boolean

Capture the screenshot beyond the viewport. Defaults to false.

Experimental
optimizeForSpeed
boolean

Optimize image encoding for speed, not for resulting size (defaults to false)

Experimental

Return Object

data
string

Base64-encoded image data. (Encoded as a base64 string when passed over JSON)

Page.createIsolatedWorld <#>

Creates an isolated world for the given frame.

parameters

frameId
[FrameId](#)

Id of the frame in which the isolated world should be created.

worldName
string

An optional name which is reported in the Execution Context.

grantUniversalAccess
boolean

Whether or not universal access should be granted to the isolated world. This is a powerful option, use with caution.

Return Object

executionContextId
[Runtime.ExecutionContextId](#)

Execution context of the isolated world.

Page.disable <#>

Disables page domain notifications.

Page.enable #

Enables page domain notifications.

Page.getAppManifest #

Return Object

url
string
Manifest location.

errors
array[[AppManifestError](#)]

data
string
Manifest content.

parsed
[AppManifestParsedProperties](#)
Parsed manifest properties
Experimental

Page.getFrameTree #

Returns present frame tree structure.

Return Object

frameTree
[FrameTree](#)
Present frame tree structure.

Page.getLayoutMetrics #

Returns metrics relating to the layouting of the page, such as viewport bounds/scale.

Return Object

layoutViewport
[LayoutViewport](#)
Deprecated metrics relating to the layout viewport. Is in device pixels. Use `cssLayoutViewport` instead.

Deprecated
visualViewport
[VisualViewport](#)

Deprecated metrics relating to the visual viewport. Is in device pixels. Use `cssVisualViewport` instead.

Deprecated
`contentSize`
[DOM.Rect](#)

Deprecated size of scrollable area. Is in DP. Use `cssContentSize` instead.

Deprecated
`cssLayoutViewport`
[LayoutViewport](#)

Metrics relating to the layout viewport in CSS pixels.

`cssVisualViewport`
[VisualViewport](#)

Metrics relating to the visual viewport in CSS pixels.

`cssContentSize`
[DOM.Rect](#)

Size of scrollable area in CSS pixels.

Page.getNavigationHistory #

Returns navigation history for the current page.

Return Object

`currentIndex`
 integer

Index of the current navigation history entry.

`entries`
 array[[NavigationEntry](#)]

Array of navigation history entries.

Page.handleJavaScriptDialog #

Accepts or dismisses a JavaScript initiated dialog (alert, confirm, prompt, or onbeforeunload).

parameters

`accept`
 boolean

Whether to accept or dismiss the dialog.

`promptText`
 string

The text to enter into the dialog prompt before accepting. Used only if this is a prompt dialog.

Page.navigate #

Navigates current page to the given URL.

parameters

url

string

URL to navigate the page to.

referrer

string

Referrer URL.

transitionType

[TransitionType](#)

Intended transition type.

frameId

[FrameId](#)

Frame id to navigate, if not specified navigates the top frame.

referrerPolicy

[ReferrerPolicy](#)

Referrer-policy used for the navigation.

Experimental

Return Object

frameId

[FrameId](#)

Frame id that has navigated (or failed to navigate)

loaderId

[Network.LoaderId](#)

Loader identifier. This is omitted in case of same-document navigation, as the previously committed loaderId would not change.

errorText

string

User friendly error message, present if and only if navigation has failed.

Page.navigateToHistoryEntry #

Navigates current page to the given history entry.

parameters

entryId

integer

Unique id of the entry to navigate to.

Page.printToPDF <#>

Print page as PDF.

parameters

landscape

boolean

Paper orientation. Defaults to false.

displayHeaderFooter

boolean

Display header and footer. Defaults to false.

printBackground

boolean

Print background graphics. Defaults to false.

scale

number

Scale of the webpage rendering. Defaults to 1.

paperWidth

number

Paper width in inches. Defaults to 8.5 inches.

paperHeight

number

Paper height in inches. Defaults to 11 inches.

marginTop

number

Top margin in inches. Defaults to 1cm (~0.4 inches).

marginBottom

number

Bottom margin in inches. Defaults to 1cm (~0.4 inches).

`marginLeft`
number

Left margin in inches. Defaults to 1cm (~0.4 inches).

`marginRight`
number

Right margin in inches. Defaults to 1cm (~0.4 inches).

`pageRanges`
string

Paper ranges to print, one based, e.g., '1-5, 8, 11-13'. Pages are printed in the document order, not in the order specified, and no more than once. Defaults to empty string, which implies the entire document is printed. The page numbers are quietly capped to actual page count of the document, and ranges beyond the end of the document are ignored. If this results in no pages to print, an error is reported. It is an error to specify a range with start greater than end.

`headerTemplate`
string

HTML template for the print header. Should be valid HTML markup with following classes used to inject printing values into them:

- `date`: formatted print date
- `title`: document title
- `url`: document location
- `pageNumber`: current page number
- `totalPages`: total pages in the document

For example, `` would generate span containing the title.

`footerTemplate`
string

HTML template for the print footer. Should use the same format as the `headerTemplate`.

`preferCSSPageSize`
boolean

Whether or not to prefer page size as defined by css. Defaults to false, in which case the content will be scaled to fit the paper size.

`transferMode`
string

return as stream

Allowed Values: `ReturnAsBase64`, `ReturnAsStream`
Experimental

`generateTaggedPDF`
boolean

Whether or not to generate tagged (accessible) PDF. Defaults to embedder choice.

Experimental

Return Object

data

string

Base64-encoded pdf data. Empty if `|returnAsStream|` is specified. (Encoded as a base64 string when passed over JSON)

stream

[IO.StreamHandle](#)

A handle of the stream that holds resulting PDF data.

Experimental

Page.reload <#>

Reloads given page optionally ignoring the cache.

parameters

ignoreCache

boolean

If true, browser cache is ignored (as if the user pressed Shift+refresh).

scriptToEvaluateOnLoad

string

If set, the script will be injected into all frames of the inspected page after reload. Argument will be ignored if reloading dataURL origin.

Page.removeScriptToEvaluateOnNewDocument <#>

Removes given script from the list.

parameters

identifier

[ScriptIdentifier](#)

Page.resetNavigationHistory <#>

Resets navigation history for the current page.

Page.setDocumentContent <#>

Sets given markup as the document's HTML.

parameters

frameId

[FrameId](#)

Frame id to set HTML for.

html

string

HTML content to set.

Page.stopLoading <#>

Force the page stop all navigations and pending resource fetches.

Page.clearGeolocationOverride **Deprecated** <#>

Clears the overridden Geolocation Position and Error.

Page.setGeolocationOverride **Deprecated** <#>

Overrides the Geolocation Position or Error. Omitting any of the parameters emulates position unavailable.

parameters

latitude

number

Mock latitude

longitude

number

Mock longitude

accuracy

number

Mock accuracy

Page.addCompilationCache **Experimental** <#>

Seeds compilation cache for given url. Compilation cache does not survive cross-process navigation.

parameters

url

string

data

string

Base64-encoded data (Encoded as a base64 string when passed over JSON)

Page.captureSnapshot **Experimental** <#>

Returns a snapshot of the page as a string. For MHTML format, the serialization includes iframes, shadow DOM, external resources, and element-inline styles.

parameters

format

string

Format (defaults to mhtml).

Allowed Values: `mhtml`

Return Object

data

string

Serialized page data.

Page.clearCompilationCache [Experimental #](#)

Clears seeded compilation cache.

Page.close [Experimental #](#)

Tries to close page, running its beforeunload hooks, if any.

Page.crash [Experimental #](#)

Crashes renderer on the IO thread, generates minidumps.

Page.generateTestReport [Experimental #](#)

Generates a report for testing.

parameters

message

string

Message to be displayed in the report.

group

string

Specifies the endpoint group to deliver the report to.

Page.getAdScriptId [Experimental #](#)

parameters

frameId

[FrameId](#)

Return Object

adScriptId
[AdScriptId](#)

Identifies the bottom-most script which caused the frame to be labelled as an ad. Only sent if frame is labelled as an ad and id is available.

Page.getAppId Experimental <#>

Returns the unique (PWA) app id. Only returns values if the feature flag 'WebAppEnableManifestId' is enabled

Return Object

appId
string
App id, either from manifest's id attribute or computed from start_url

recommendedId
string
Recommendation for manifest's id attribute to match current id computed from start_url

Page.getInstallabilityErrors Experimental <#>

Return Object

installabilityErrors
array[[InstallabilityError](#)]

Page.getOriginTrials Experimental <#>

Get Origin Trials on given frame.

parameters

frameId
[FrameId](#)

Return Object

originTrials
array[[OriginTrial](#)]

Page.getPermissionsPolicyState Experimental <#>

Get Permissions Policy state on given frame.

parameters

frameId

[FrameId](#)

Return Object

states

array[[PermissionsPolicyFeatureState](#)]

Page.getResourceContent Experimental #

Returns content of the given resource.

parameters

frameId

[FrameId](#)

Frame id to get resource for.

url

string

URL of the resource to get content for.

Return Object

content

string

Resource content.

base64Encoded

boolean

True, if content was served as base64.

Page.getResourceTree Experimental #

Returns present frame / resource tree structure.

Return Object

frameTree

[FrameResourceTree](#)

Present frame / resource tree structure.

Page.produceCompilationCache Experimental #

Requests backend to produce compilation cache for the specified scripts. `scripts` are appended to the list of scripts for which the cache would be produced. The list may be reset during page navigation. When script with a matching URL is encountered, the cache is optionally produced upon backend discretion, based on internal heuristics. See also: `Page.compilationCacheProduced`.

parameters

scripts

array[[CompilationCacheParams](#)]

Page.screencastFrameAck Experimental <#>

Acknowledges that a screencast frame has been received by the frontend.

parameters

sessionId

integer

Frame number.

Page.searchInResource Experimental <#>

Searches for given string in resource content.

parameters

frameId

[FrameId](#)

Frame id for resource to search in.

url

string

URL of the resource to search in.

query

string

String to search for.

caseSensitive

boolean

If true, search is case sensitive.

isRegex

boolean

If true, treats string parameter as regex.

Return Object

result

array[[Debugger.SearchMatch](#)]

List of search matches.

Page.setAdBlockingEnabled Experimental #

Enable Chrome's experimental ad filter on all sites.

parameters

enabled

boolean

Whether to block ads.

Page.setBypassCSP Experimental #

Enable page Content Security Policy by-passing.

parameters

enabled

boolean

Whether to bypass page CSP.

Page.setFontFamilies Experimental #

Set generic font families.

parameters

fontFamilies

[FontFamilies](#)

Specifies font families to set. If a font family is not specified, it won't be changed.

forScripts

array[[ScriptFontFamilies](#)]

Specifies font families to set for individual scripts.

Page.setFontSizes Experimental #

Set default font sizes.

parameters

fontSizes

[FontSizes](#)

Specifies font sizes to set. If a font size is not specified, it won't be changed.

Page.setInterceptFileChooserDialog Experimental #

Intercept file chooser requests and transfer control to protocol clients. When file chooser interception is enabled, native file chooser dialog is not shown. Instead, a protocol event `Page.fileChooserOpened` is emitted.

parameters

enabled
boolean

Page.setLifecycleEventsEnabled Experimental #

Controls whether page will emit lifecycle events.

parameters

enabled
boolean

If true, starts emitting lifecycle events.

Page.setPrerenderingAllowed Experimental #

Enable/disable prerendering manually.

This command is a short-term solution for <https://crbug.com/1440085>. See https://docs.google.com/document/d/12HVmFxYj5Jc-eJr5OmWsa2bqTJsbgGLKI6ZIyx0_wpA for more details.

TODO(<https://crbug.com/1440085>): Remove this once Puppeteer supports tab targets.

parameters

isAllowed
boolean

Page.setRPHRegistrationMode Experimental #

Extensions for Custom Handlers API: <https://html.spec.whatwg.org/multipage/system-state.html#rph-automation>

parameters

mode
[AutoResponseMode](#)

Page.setSPCTransactionMode Experimental #

Sets the Secure Payment Confirmation transaction mode. <https://w3c.github.io/secure-payment-confirmation/#sctn-automation-set-spc-transaction-mode>

parameters

mode

[AutoResponseMode](#)**Page.setWebLifecycleState Experimental #**

Tries to update the web lifecycle state of the page. It will transition the page to the given state according to:

<https://github.com/WICG/web-lifecycle/>

parameters

state

string

Target lifecycle state

Allowed Values: frozen, active

Page.startScreencast Experimental #

Starts sending each frame using the `screencastFrame` event.

parameters

format

string

Image compression format.

Allowed Values: jpeg, png

quality

integer

Compression quality from range [0..100].

maxWidth

integer

Maximum screenshot width.

maxHeight

integer

Maximum screenshot height.

everyNthFrame

integer

Send every n-th frame.

Page.stopScreencast Experimental #

Stops sending each frame in the `screenshotFrame`.

Page.waitForDebugger Experimental <#>

Pauses page execution. Can be resumed using generic `Runtime.runIfWaitingForDebugger`.

Page.addScriptToEvaluateOnLoad ExperimentalDeprecated <#>

Deprecated, please use `addScriptToEvaluateOnNewDocument` instead.

parameters

`scriptSource`
string

Return Object

`identifier`
[ScriptIdentifier](#)

Identifier of the added script.

Page.clearDeviceMetricsOverride ExperimentalDeprecated <#>

Clears the overridden device metrics.

Page.clearDeviceOrientationOverride ExperimentalDeprecated <#>

Clears the overridden Device Orientation.

Page.deleteCookie ExperimentalDeprecated <#>

Deletes browser cookie with given name, domain and path.

parameters

`cookieName`
string
Name of the cookie to remove.

`url`
string
URL to match cookie domain and path.

Page.getCookies ExperimentalDeprecated <#>

Returns all browser cookies for the page and all of its subframes. Depending on the backend support, will return detailed cookie information in the `cookies` field.

Return Object

cookies

array[[Network.Cookie](#)]

Array of cookie objects.

Page.getManifestIcons ExperimentalDeprecated #

Deprecated because it's not guaranteed that the returned icon is in fact the one used for PWA installation.

Return Object

primaryIcon

string

Page.removeScriptToEvaluateOnLoad ExperimentalDeprecated #

Deprecated, please use removeScriptToEvaluateOnNewDocument instead.

parameters

identifier

[ScriptIdentifier](#)**Page.setDeviceMetricsOverride ExperimentalDeprecated #**

Overrides the values of device screen dimensions (window.screen.width, window.screen.height, window.innerWidth, window.innerHeight, and "device-width"/"device-height"-related CSS media query results).

parameters

width

integer

Overriding width value in pixels (minimum 0, maximum 10000000). 0 disables the override.

height

integer

Overriding height value in pixels (minimum 0, maximum 10000000). 0 disables the override.

deviceScaleFactor

number

Overriding device scale factor value. 0 disables the override.

mobile

boolean

Whether to emulate mobile device. This includes viewport meta tag, overlay scrollbars, text autosizing and more.

scale
number

Scale to apply to resulting view image.

screenWidth
integer

Overriding screen width value in pixels (minimum 0, maximum 10000000).

screenHeight
integer

Overriding screen height value in pixels (minimum 0, maximum 10000000).

positionX
integer

Overriding view X position on screen in pixels (minimum 0, maximum 10000000).

positionY
integer

Overriding view Y position on screen in pixels (minimum 0, maximum 10000000).

dontSetVisibleSize
boolean

Do not set visible view size, rely upon explicit setVisibleSize call.

screenOrientation
[Emulation.ScreenOrientation](#)

Screen orientation override.

viewport
[Viewport](#)

The viewport dimensions and scale. If not set, the override is cleared.

Page.setDeviceOrientationOverride ExperimentalDeprecated <#>

Overrides the Device Orientation.

parameters

alpha
number

Mock alpha

beta
number

Mock beta

gamma
number

Mock gamma

Page.setDownloadBehavior ExperimentalDeprecated <#>

Set the behavior when downloading a file.

parameters

behavior
string

Whether to allow all or deny all download requests, or use default Chrome behavior if available (otherwise deny).

Allowed Values: deny, allow, default

downloadPath
string

The default path to save downloaded files to. This is required if behavior is set to 'allow'

Page.setTouchEmulationEnabled ExperimentalDeprecated <#>

Toggles mouse event-based touch event emulation.

parameters

enabled
boolean

Whether the touch event emulation should be enabled.

configuration
string

Touch/gesture events configuration. Default: current platform.

Allowed Values: mobile, desktop

Events

Page.domContentEventFired <#>

parameters

timestamp
[Network.MonotonicTime](#)

Page.fileChooserOpened <#>

Emitted only when `page.interceptFileChooser` is enabled.

parameters

frameId

[FrameId](#)

Id of the frame containing input node.

Experimental

mode

string

Input mode.

Allowed Values: `selectSingle`, `selectMultiple`

backendNodeId

[DOM.BackendNodeId](#)

Input node id. Only present for file choosers opened via an `<input type="file">` element.

Experimental

Page.frameAttached <#>

Fired when frame has been attached to its parent.

parameters

frameId

[FrameId](#)

Id of the frame that has been attached.

parentFrameId

[FrameId](#)

Parent frame identifier.

stack

[Runtime.StackTrace](#)

JavaScript stack trace of when frame was attached, only set if frame initiated from script.

Page.frameDetached <#>

Fired when frame has been detached from its parent.

parameters

frameId

[FrameId](#)

Id of the frame that has been detached.

reason

string

Allowed Values: remove, swap

Experimental

Page.frameNavigated #

Fired once navigation of the frame has completed. Frame is now associated with the new loader.

parameters

frame

[Frame](#)

Frame object.

type

[NavigationType](#)

Experimental

Page.interstitialHidden #

Fired when interstitial page was hidden

Page.interstitialShown #

Fired when interstitial page was shown

Page.javascriptDialogClosed #

Fired when a JavaScript initiated dialog (alert, confirm, prompt, or onbeforeunload) has been closed.

parameters

result

boolean

Whether dialog was confirmed.

userInput

string

User input in case of prompt.

Page.javascriptDialogOpening #

Fired when a JavaScript initiated dialog (alert, confirm, prompt, or onbeforeunload) is about to open.

parameters

url

string

Frame url.

message
string

Message that will be displayed by the dialog.

type

[DialogType](#)

Dialog type.

hasBrowserHandler
boolean

True iff browser is capable showing or acting on the given dialog. When browser has no dialog handler for given target, calling alert while Page domain is engaged will stall the page execution. Execution can be resumed via calling Page.handleJavaScriptDialog.

defaultPrompt
string

Default dialog prompt.

Page.lifecycleEvent #

Fired for top level page lifecycle events such as navigation, load, paint, etc.

parameters

frameId
[FrameId](#)

Id of the frame.

loaderId
[Network.LoaderId](#)

Loader identifier. Empty string if the request is fetched from worker.

name
string

timestamp
[Network.MonotonicTime](#)

Page.loadEventFired #

parameters

timestamp
[Network.MonotonicTime](#)

Page.windowOpen #

Fired when a new window is going to be opened, via window.open(), link click, form submission, etc.

parameters

url

string

The URL for the new window.

windowName

string

Window name.

windowFeatures

array[string]

An array of enabled window features.

userGesture

boolean

Whether or not it was triggered by user gesture.

Page.frameClearedScheduledNavigation Deprecated <#>

Fired when frame no longer has a scheduled navigation.

parameters

frameId

[FrameId](#)

Id of the frame that has cleared its scheduled navigation.

Page.frameScheduledNavigation Deprecated <#>

Fired when frame schedules a potential navigation.

parameters

frameId

[FrameId](#)

Id of the frame that has scheduled a navigation.

delay

number

Delay (in seconds) until the navigation is scheduled to begin. The navigation is not guaranteed to start.

reason

[ClientNavigationReason](#)

The reason for the navigation.

url
string

The destination URL for the scheduled navigation.

Page.backForwardCacheNotUsed Experimental <#>

Fired for failed bfcache history navigations if BackForwardCache feature is enabled. Do not assume any ordering with the Page.frameNavigated event. This event is fired only for main-frame history navigation where the document changes (non-same-document navigations), when bfcache navigation fails.

parameters

loaderId
[Network.LoaderId](#)

The loader id for the associated navigation.

frameId
[FrameId](#)

The frame id of the associated frame.

notRestoredExplanations
array[[BackForwardCacheNotRestoredExplanation](#)]

Array of reasons why the page could not be cached. This must not be empty.

notRestoredExplanationsTree
[BackForwardCacheNotRestoredExplanationTree](#)

Tree structure of reasons why the page could not be cached for each frame.

Page.compilationCacheProduced Experimental <#>

Issued for every compilation cache generated. Is only available if Page.setGenerateCompilationCache is enabled.

parameters

url
string
data
string

Base64-encoded data (Encoded as a base64 string when passed over JSON)

Page.documentOpened Experimental <#>

Fired when opening document to write to.

parameters

frame

[Frame](#)

Frame object.

Page.frameRequestedNavigation [Experimental](#) <#>

Fired when a renderer-initiated navigation is requested. Navigation may still be cancelled after the event is issued.

parameters

frameId

[FrameId](#)

Id of the frame that is being navigated.

reason

[ClientNavigationReason](#)

The reason for the navigation.

url

string

The destination URL for the requested navigation.

disposition

[ClientNavigationDisposition](#)

The disposition for the navigation.

Page.frameResized [Experimental](#) <#>

Page.frameStartedLoading [Experimental](#) <#>

Fired when frame has started loading.

parameters

frameId

[FrameId](#)

Id of the frame that has started loading.

Page.frameStoppedLoading [Experimental](#) <#>

Fired when frame has stopped loading.

parameters

frameId

[FrameId](#)

Id of the frame that has stopped loading.

Page.navigatedWithinDocument [Experimental #](#)

Fired when same-document navigation happens, e.g. due to history API usage or anchor navigation.

parameters

frameId

[FrameId](#)

Id of the frame.

url

string

Frame's new url.

Page.screencastFrame [Experimental #](#)

Compressed image data requested by the `startScreencast`.

parameters

data

string

Base64-encoded compressed image. (Encoded as a base64 string when passed over JSON)

metadata

[ScreencastFrameMetadata](#)

Screencast frame metadata.

sessionId

integer

Frame number.

Page.screencastVisibilityChanged [Experimental #](#)

Fired when the page with currently enabled screencast was shown or hidden `.

parameters

visible

boolean

True if the page is visible.

Page.downloadProgress [ExperimentalDeprecated #](#)

Fired when download makes progress. Last call has `|done| == true`. Deprecated. Use `Browser.downloadProgress` instead.

parameters

guid

string

Global unique identifier of the download.

totalBytes

number

Total expected bytes to download.

receivedBytes

number

Total bytes received.

state

string

Download status.

Allowed Values: `inProgress`, `completed`, `canceled`

Page.downloadWillBegin ExperimentalDeprecated <#>

Fired when page is about to start a download. Deprecated. Use `Browser.downloadWillBegin` instead.

parameters

frameId

[FrameId](#)

Id of the frame that caused download to begin.

guid

string

Global unique identifier of the download.

url

string

URL of the resource being downloaded.

suggestedFilename

string

Suggested file name of the resource (the actual name of the file saved on disk may differ).

Types

Page.AppManifestError <#>

Error while parsing app manifest.

Type: **object**

properties

message

string

Error message.

critical

integer

If critical, this is a non-recoverable parse error.

line

integer

Error line.

column

integer

Error column.

Page.DialogType <#>

Javascript dialog type.

Allowed Values: alert, confirm, prompt, beforeunload

Type: **string**

Page.Frame <#>

Information about the Frame on the page.

Type: **object**

properties

id

[FrameId](#)

Frame unique identifier.

parentId

[FrameId](#)

Parent frame identifier.

loaderId

[Network.LoaderId](#)

Identifier of the loader associated with this frame.

name

string

Frame's name as specified in the tag.

url

string

Frame document's URL without fragment.

urlFragment

string

Frame document's URL fragment including the '#'.

Experimental

domainAndRegistry

string

Frame document's registered domain, taking the public suffixes list into account. Extracted from the Frame's url. Example URLs: <http://www.google.com/file.html> -> "google.com" <http://a.b.co.uk/file.html> -> "b.co.uk"

Experimental

securityOrigin

string

Frame document's security origin.

contentType

string

Frame document's contentType as determined by the browser.

unreachableUrl

string

If the frame failed to load, this contains the URL that could not be loaded. Note that unlike url above, this URL may contain a fragment.

Experimental

adFrameStatus

[AdFrameStatus](#)

Indicates whether this frame was tagged as an ad and why.

Experimental

secureContextType

[SecureContextType](#)

Indicates whether the main document is a secure context and explains why that is the case.

Experimental
crossOriginIsolatedContextType
[CrossOriginIsolatedContextType](#)

Indicates whether this is a cross origin isolated context.

Experimental
gatedAPIFeatures
array[[GatedAPIFeatures](#)]

Indicated which gated APIs / features are available.

Experimental

Page.FrameId #

Unique frame identifier.

Type: **string**

Page.FrameTree #

Information about the Frame hierarchy.

Type: **object**

properties

frame
[Frame](#)

Frame information for this tree item.

childFrames
array[[FrameTree](#)]

Child frames.

Page.LayoutViewport #

Layout viewport position and dimensions.

Type: **object**

properties

pageX
integer

Horizontal offset relative to the document (CSS pixels).

pageY
integer

Vertical offset relative to the document (CSS pixels).
clientWidth
integer

Width (CSS pixels), excludes scrollbar if present.

clientHeight
integer

Height (CSS pixels), excludes scrollbar if present.

Page.NavigationEntry <#>

Navigation history entry.

Type: **object**

properties

id
integer
Unique id of the navigation history entry.

url
string
URL of the navigation history entry.

userTypedURL
string
URL that the user typed in the url bar.

title
string
Title of the navigation history entry.

transitionType
[TransitionType](#)
Transition type.

Page.ScriptIdentifier <#>

Unique script identifier.

Type: **string**

Page.TransitionType <#>

Transition type.

Allowed Values: link, typed, address_bar, auto_bookmark, auto_subframe, manual_subframe, generated, auto_toplevel, form_submit, reload, keyword, keyword_generated, other

Type: **string**

Page.Viewport <#>

Viewport for capturing screenshot.

Type: **object**

properties

x

number

X offset in device independent pixels (dip).

y

number

Y offset in device independent pixels (dip).

width

number

Rectangle width in device independent pixels (dip).

height

number

Rectangle height in device independent pixels (dip).

scale

number

Page scale factor.

Page.VisualViewport <#>

Visual viewport position, dimensions, and scale.

Type: **object**

properties

offsetX

number

Horizontal offset relative to the layout viewport (CSS pixels).

offsetY

number

Vertical offset relative to the layout viewport (CSS pixels).

pageX
number

Horizontal offset relative to the document (CSS pixels).

pageY
number

Vertical offset relative to the document (CSS pixels).

clientWidth
number

Width (CSS pixels), excludes scrollbar if present.

clientHeight
number

Height (CSS pixels), excludes scrollbar if present.

scale
number

Scale relative to the ideal viewport (size at width=device-width).

zoom
number

Page zoom factor (CSS to device independent pixels ratio).

Page.AdFrameExplanation Experimental <#>

Allowed Values: ParentIsAd, CreatedByAdScript, MatchedBlockingRule

Type: **string**

Page.AdFrameStatus Experimental <#>

Indicates whether a frame has been identified as an ad and why.

Type: **object**

properties

adFrameType
[AdFrameType](#)

explanations
array[[AdFrameExplanation](#)]

Page.AdFrameType Experimental <#>

Indicates whether a frame has been identified as an ad.

Allowed Values: none, child, root

Type: **string**

Page.AdScriptId Experimental #

Identifies the bottom-most script which caused the frame to be labelled as an ad.

Type: **object**

properties

scriptId

[Runtime.ScriptId](#)

Script Id of the bottom-most script which caused the frame to be labelled as an ad.

debuggerId

[Runtime.UniqueDebuggerId](#)

Id of adScriptId's debugger.

Page.AppManifestParsedProperties Experimental #

Parsed app manifest properties.

Type: **object**

properties

scope

string

Computed scope value

Page.AutoResponseMode Experimental #

Enum of possible auto-reponse for permisison / prompt dialogs.

Allowed Values: none, autoAccept, autoReject, autoOptOut

Type: **string**

Page.BackForwardCacheNotRestoredExplanation Experimental #

Type: **object**

properties

type

[BackForwardCacheNotRestoredReasonType](#)

Type of the reason

reason

[BackForwardCacheNotRestoredReason](#)

Not restored reason

context

string

Context associated with the reason. The meaning of this context is dependent on the reason:

- `EmbedderExtensionSentMessageToCachedFrame`: the extension ID.

Page.BackForwardCacheNotRestoredExplanationTree Experimental <#>

Type: **object**

properties

url

string

URL of each frame

explanations

array[[BackForwardCacheNotRestoredExplanation](#)]

Not restored reasons of each frame

children

array[[BackForwardCacheNotRestoredExplanationTree](#)]

Array of children frame

Page.BackForwardCacheNotRestoredReason Experimental <#>

List of not restored reasons for back-forward cache.

Allowed Values: `NotPrimaryMainFrame`, `BackForwardCacheDisabled`, `RelatedActiveContentsExist`, `HTTPStatusNotOK`, `SchemeNotHTTPOrHTTPS`, `Loading`, `WasGrantedMediaAccess`, `DisableForRenderFrameHostCalled`, `DomainNotAllowed`, `HTTPMethodNotGET`, `SubframeIsNavigating`, `Timeout`, `CacheLimit`, `JavaScriptExecution`, `RendererProcessKilled`, `RendererProcessCrashed`, `SchedulerTrackedFeatureUsed`, `ConflictingBrowsingInstance`, `CacheFlushed`, `ServiceWorkerVersionActivation`, `SessionRestored`, `ServiceWorkerPostMessage`, `EnteredBackForwardCacheBeforeServiceWorkerHostAdded`, `RenderFrameHostReused_SameSite`, `RenderFrameHostReused_CrossSite`, `ServiceWorkerClaim`, `IgnoreEventAndEvict`, `HaveInnerContents`, `TimeoutPuttingInCache`, `BackForwardCacheDisabledByLowMemory`, `BackForwardCacheDisabledByCommandLine`, `NetworkRequestDatapipeDrainedAsBytesConsumer`, `NetworkRequestRedirected`, `NetworkRequestTimeout`, `NetworkExceedsBufferLimit`, `NavigationCancelledWhileRestoring`, `NotMostRecentNavigationEntry`, `BackForwardCacheDisabledForPrerender`, `UserAgentOverrideDiffers`, `ForegroundCacheLimit`, `BrowsingInstanceNotSwapped`, `BackForwardCacheDisabledForDelegate`, `UnloadHandlerExistsInMainFrame`, `UnloadHandlerExistsInSubFrame`, `ServiceWorkerUnregistration`, `CacheControlNoStore`, `CacheControlNoStoreCookieModified`, `CacheControlNoStoreHTTPOnlyCookieModified`, `NoResponseHead`, `Unknown`,

ActivationNavigationsDisallowedForBug1234857, ErrorDocument, FencedFramesEmbedder, CookieDisabled, HTTPAuthRequired, CookieFlushed, WebSocket, WebTransport, WebRTC, MainResourceHasCacheControlNoStore, MainResourceHasCacheControlNoCache, SubresourceHasCacheControlNoStore, SubresourceHasCacheControlNoCache, ContainsPlugins, DocumentLoaded, DedicatedWorkerOrWorklet, OutstandingNetworkRequestOthers, RequestedMIDIPermission, RequestedAudioCapturePermission, RequestedVideoCapturePermission, RequestedBackForwardCacheBlockedSensors, RequestedBackgroundWorkPermission, BroadcastChannel, WebXR, SharedWorker, WebLocks, WebHID, WebShare, RequestedStorageAccessGrant, WebNfc, OutstandingNetworkRequestFetch, OutstandingNetworkRequestXHR, AppBanner, Printing, WebDatabase, PictureInPicture, Portal, SpeechRecognizer, IdleManager, PaymentManager, SpeechSynthesis, KeyboardLock, WebOTPService, OutstandingNetworkRequestDirectSocket, InjectedJavascript, InjectedStyleSheet, KeepaliveRequest, IndexedDBEvent, Dummy, JsNetworkRequestReceivedCacheControlNoStoreResource, WebRTCSticky, WebTransportSticky, WebSocketSticky, ContentSecurityHandler, ContentWebAuthenticationAPI, ContentFileChooser, ContentSerial, ContentFileSystemAccess, ContentMediaDevicesDispatcherHost, ContentWebBluetooth, ContentWebUSB, ContentMediaSessionService, ContentScreenReader, EmbedderPopupBlockerTabHelper, EmbedderSafeBrowsingTriggeredPopupBlocker, EmbedderSafeBrowsingThreatDetails, EmbedderAppBannerManager, EmbedderDomDistillerViewerSource, EmbedderDomDistillerSelfDeletingRequestDelegate, EmbedderOomInterventionTabHelper, EmbedderOfflinePage, EmbedderChromePasswordManagerClientBindCredentialManager, EmbedderPermissionRequestManager, EmbedderModalDialog, EmbedderExtensions, EmbedderExtensionMessaging, EmbedderExtensionMessagingForOpenPort, EmbedderExtensionSendMessageToCachedFrame

Type: **string**

Page.BackForwardCacheNotRestoredReasonType [Experimental #](#)

Types of not restored reasons for back-forward cache.

Allowed Values: SupportPending, PageSupportNeeded, Circumstantial

Type: **string**

Page.ClientNavigationDisposition [Experimental #](#)

Allowed Values: currentTab, newTab, newWindow, download

Type: **string**

Page.ClientNavigationReason [Experimental #](#)

Allowed Values: formSubmissionGet, formSubmissionPost, httpHeaderRefresh, scriptInitiated, metaTagRefresh, pageBlockInterstitial, reload, anchorClick

Type: **string**

Page.CompilationCacheParams [Experimental #](#)

Per-script compilation cache parameters for `Page.produceCompilationCache`

Type: **object**

properties

url

string

The URL of the script to produce a compilation cache entry for.

eager

boolean

A hint to the backend whether eager compilation is recommended. (the actual compilation mode used is upon backend discretion).

Page.CrossOriginIsolatedContextType Experimental <#>

Indicates whether the frame is cross-origin isolated and why it is the case.

Allowed Values: Isolated, NotIsolated, NotIsolatedFeatureDisabled

Type: **string**

Page.FontFamilies Experimental <#>

Generic font families collection.

Type: **object**

properties

standard

string

The standard font-family.

fixed

string

The fixed font-family.

serif

string

The serif font-family.

sansSerif

string

The sansSerif font-family.

cursive

string

The cursive font-family.

fantasy

string

The fantasy font-family.

math

string

The math font-family.

Page.FontSizes Experimental <#>

Default font sizes.

Type: **object**

properties

standard

integer

Default standard font size.

fixed

integer

Default fixed font size.

Page.FrameResource Experimental <#>

Information about the Resource on the page.

Type: **object**

properties

url

string

Resource URL.

type

[Network.ResourceType](#)

Type of this resource.

mimeType

string

Resource mimeType as determined by the browser.

lastModified

[Network.TimeSinceEpoch](#)

last-modified timestamp as reported by server.

contentSize
number

Resource content size.

failed
boolean

True if the resource failed to load.

canceled
boolean

True if the resource was canceled during loading.

Page.FrameResourceTree Experimental #

Information about the Frame hierarchy along with their cached resources.

Type: **object**

properties

frame
[Frame](#)

Frame information for this tree item.

childFrames
array[[FrameResourceTree](#)]

Child frames.

resources
array[[FrameResource](#)]

Information about frame resources.

Page.GatedAPIFeatures Experimental #

Allowed Values: SharedArrayBuffers, SharedArrayBuffersTransferAllowed, PerformanceMeasureMemory, PerformanceProfile

Type: **string**

Page.InstallabilityError Experimental #

The installability error

Type: **object**

properties

errorId

string

The error id (e.g. 'manifest-missing-suitable-icon').

errorArguments

array[[InstallabilityErrorArgument](#)]

The list of error arguments (e.g. {name:'minimum-icon-size-in-pixels', value:'64'}).

Page.InstallabilityErrorArgument Experimental <#>

Type: **object**

properties

name

string

Argument name (e.g. name:'minimum-icon-size-in-pixels').

value

string

Argument value (e.g. value:'64').

Page.NavigationType Experimental <#>

The type of a frameNavigated event.

Allowed Values: Navigation, BackForwardCacheRestore

Type: **string**

Page.OriginTrial Experimental <#>

Type: **object**

properties

trialName

string

status

[OriginTrialStatus](#)

tokensWithStatus

array[[OriginTrialTokenWithStatus](#)]

Page.OriginTrialStatus Experimental <#>

Status for an Origin Trial.

Allowed Values: Enabled, ValidTokenNotProvided, OSNotSupported, TrialNotAllowed

Type: **string**

Page.OriginTrialToken Experimental #Type: **object****properties**

origin

string

matchSubDomains

boolean

trialName

string

expiryTime

[Network.TimeSinceEpoch](#)

isThirdParty

boolean

usageRestriction

[OriginTrialUsageRestriction](#)**Page.OriginTrialTokenStatus Experimental #**Origin Trial(<https://www.chromium.org/blink/origin-trials>) support. Status for an Origin Trial token.

Allowed Values: Success, NotSupported, Insecure, Expired, WrongOrigin, InvalidSignature, Malformed, WrongVersion, FeatureDisabled, TokenDisabled, FeatureDisabledForUser, UnknownTrial

Type: **string****Page.OriginTrialTokenWithStatus Experimental #**Type: **object****properties**

rawTokenText

string

parsedToken

[OriginTrialToken](#)

parsedToken is present only when the token is extractable and parsable.

status

[OriginTrialTokenStatus](#)**Page.OriginTrialUsageRestriction Experimental #**

Allowed Values: None, Subset

Type: **string****Page.PermissionsPolicyBlockLocator Experimental #**

Type: **object**

properties

frameId

[FrameId](#)

blockReason

[PermissionsPolicyBlockReason](#)

Page.PermissionsPolicyBlockReason Experimental <#>

Reason for a permissions policy feature to be disabled.

Allowed Values: Header, IframeAttribute, InFencedFrameTree, InIsolatedApp

Type: **string**

Page.PermissionsPolicyFeature Experimental <#>

All Permissions Policy features. This enum should match the one defined in `third_party/blink/renderer/core/permissions_policy/permissions_policy_features.json5`.

Allowed Values: accelerometer, ambient-light-sensor, attribution-reporting, autoplay, bluetooth, browsing-topics, camera, ch-dpr, ch-device-memory, ch-downlink, ch-ect, ch-prefers-color-scheme, ch-prefers-reduced-motion, ch-prefers-reduced-transparency, ch-rtt, ch-save-data, ch-ua, ch-ua-arch, ch-ua-bitness, ch-ua-platform, ch-ua-model, ch-ua-mobile, ch-ua-form-factor, ch-ua-full-version, ch-ua-full-version-list, ch-ua-platform-version, ch-ua-wow64, ch-viewport-height, ch-viewport-width, ch-width, clipboard-read, clipboard-write, compute-pressure, cross-origin-isolated, direct-sockets, display-capture, document-domain, encrypted-media, execution-while-out-of-viewport, execution-while-not-rendered, focus-without-user-activation, fullscreen, frobulate, gamepad, geolocation, gyroscope, hid, identity-credentials-get, idle-detection, interest-cohort, join-ad-interest-group, keyboard-map, local-fonts, magnetometer, microphone, midi, otp-credentials, payment, picture-in-picture, private-aggregation, private-state-token-issuance, private-state-token-redemption, publickey-credentials-get, run-ad-auction, screen-wake-lock, serial, shared-autofill, shared-storage, shared-storage-select-url, smart-card, storage-access, sync-xhr, unload, usb, vertical-scroll, web-share, window-management, window-placement, xr-spatial-tracking

Type: **string**

Page.PermissionsPolicyFeatureState Experimental <#>

Type: **object**

properties

feature

[PermissionsPolicyFeature](#)

allowed

boolean

locator

[PermissionsPolicyBlockLocator](#)

Page.ReferrerPolicy Experimental <#>

The referring-policy used for the navigation.

Allowed Values: noReferrer, noReferrerWhenDowngrade, origin, originWhenCrossOrigin, sameOrigin, strictOrigin, strictOriginWhenCrossOrigin, unsafeUrl

Type: **string**

Page.ScreencastFrameMetadata Experimental <#>

Screencast frame metadata.

Type: **object**

properties

offsetTop
number

Top offset in DIP.

pageScaleFactor
number

Page scale factor.

deviceWidth
number

Device screen width in DIP.

deviceHeight
number

Device screen height in DIP.

scrollOffsetX
number

Position of horizontal scroll in CSS pixels.

scrollOffsetY
number

Position of vertical scroll in CSS pixels.

timestamp
[Network.TimeSinceEpoch](#)

Frame swap timestamp.

Page.ScriptFontFamilies Experimental <#>

Font families collection for a script.

Type: **object**

properties

script

string

Name of the script which these font families are defined for.

fontFamilies

[FontFamilies](#)

Generic font families collection for the script.

Page.SecureContextType Experimental #

Indicates whether the frame is a secure context and why it is the case.

Allowed Values: Secure, SecureLocalhost, InsecureScheme, InsecureAncestor

Type: **string**