The attributes matches, exclude\_matches, include\_globs, and exclude\_globs in the **content\_scripts** section of a Chrome extension's manifest.json file allow for highly granular control over where content scripts are injected. Let's dive deeper into each of these attributes, including their functionality, use cases, and examples.

**1. matches**

* **Type**: Array of strings (each string is a match pattern).
* **Description**: Specifies the URLs (using match patterns) where the content script should be injected.
* **Use Case**: This is the primary attribute to control where the script is injected. You can specify general patterns like https://example.com/\* or more specific ones like https://example.com/page/\*.
* **Pattern Syntax**: The matches field uses match patterns which follow a simplified syntax that includes wildcards like \* (wildcard for any number of characters) and ? (wildcard for a single character).

**Examples:**

"matches": [

"https://example.com/\*",

"http://\*.example.org/\*",

"https://example.com/page/\*"

]

* "https://example.com/\*": Matches all pages under example.com.
* "http://\*.example.org/\*": Matches all pages under any subdomain of example.org.
* "https://example.com/page/\*": Matches any page under /page/ path of example.com.

**2. exclude\_matches**

* **Type**: Array of strings (each string is a match pattern).
* **Description**: Specifies URLs that should be **excluded** from being matched, even if they match a pattern defined in the matches field.
* **Use Case**: When you want to exclude specific pages or URLs that might otherwise match the matches patterns. This is helpful for excluding certain paths or domains.
* **Pattern Syntax**: The exclude\_matches patterns use the same match pattern syntax as matches, meaning you can use wildcards like \* and ?.

**Examples:**

"exclude\_matches": [

"https://example.com/settings/\*",

"https://example.com/login"

]

* "https://example.com/settings/\*": Excludes any URL under the /settings/ path on example.com.
* "https://example.com/login": Excludes the exact URL https://example.com/login.

**3. include\_globs**

* **Type**: Array of strings (each string is a glob pattern).
* **Description**: Further refines the match by only including URLs that also match the glob pattern, even if they match the patterns defined in matches. This can be used to narrow down the scope of URLs.
* **Use Case**: This is particularly useful when you want the content script to run only on URLs that match specific file types, query parameters, or other patterns after the initial matches filter is applied.
* **Glob Syntax**: Glob patterns are simpler than regular expressions and typically use wildcards like \* (any number of characters), ? (single character), and square brackets [] to specify character sets.

**Examples:**

"include\_globs": [

"\*.html",

"\*page=1\*"

]

* "\*.html": Only includes URLs that end in .html.
* "\*page=1\*": Includes URLs that have page=1 as a query parameter.

**4. exclude\_globs**

* **Type**: Array of strings (each string is a glob pattern).
* **Description**: Excludes URLs that match a given glob pattern, even if they match the matches patterns. This is the reverse of include\_globs, providing additional exclusion criteria.
* **Use Case**: This is useful when you want to exclude certain URLs that might match the general matches pattern, but you don't want the content script to run on those specific cases. You might use this to avoid script injection on certain file types (e.g., .json) or URLs containing certain query parameters.
* **Glob Syntax**: Similar to include\_globs, it uses simple glob patterns to match specific parts of a URL.

**Examples:**

"exclude\_globs": [

"\*.json",

"\*archive\*"

]

* "\*.json": Excludes any URL that ends with .json.
* "\*archive\*": Excludes any URL that contains the word archive.

**Combining matches, exclude\_matches, include\_globs, and exclude\_globs**

You can combine these attributes to refine exactly where and when the content script is injected. Here's an example of how they can work together:

{

"content\_scripts": [

{

"matches": ["https://example.com/\*"],

"exclude\_matches": ["https://example.com/settings/\*", "https://example.com/login"],

"include\_globs": ["\*.html", "\*page=1\*"],

"exclude\_globs": ["\*archive\*", "\*.json"],

"js": ["content-script.js"]

}

]

}

**Explanation:**

* **matches**: The content script will be injected into all pages under https://example.com/.
* **exclude\_matches**: Pages such as https://example.com/settings/\* and https://example.com/login are excluded from script injection.
* **include\_globs**: After the matches patterns are applied, the content script will only run on .html pages or those with a query parameter page=1.
* **exclude\_globs**: The script will **not** run on URLs containing archive or ending with .json, even if they match the matches pattern.

**Matching Logic Flow:**

1. The URL will first be checked against the patterns in matches.
2. If the URL matches a pattern in matches, it will then be checked against the patterns in exclude\_matches. If it matches an exclusion pattern, the content script will not run.
3. If the URL passes through the previous filters, it will then be checked against include\_globs and exclude\_globs. The content script will only run if it matches the inclusion patterns and does not match any exclusion patterns.

**Advanced Matching Examples:**

**Example 1: Match Specific Subdomains and Paths**

{

"matches": ["https://\*.example.com/\*"],

"exclude\_matches": ["https://\*.example.com/login/\*"],

"include\_globs": ["\*.html"],

"exclude\_globs": ["\*archive\*"],

"js": ["content-script.js"]

}

* **Matches**: Any subdomain of example.com.
* **Excludes**: Any subdomain under /login/.
* **Includes**: Only .html pages.
* **Excludes**: Pages with archive in the URL.

**Example 2: Match URLs Based on Query Parameters**

{

"matches": ["https://example.com/\*"],

"exclude\_globs": ["\*page=2\*"],

"include\_globs": ["\*page=1\*"],

"js": ["content-script.js"]

}

* **Matches**: All pages under https://example.com/.
* **Excludes**: URLs containing page=2.
* **Includes**: URLs containing page=1.

**Conclusion:**

* **matches**: Defines the URLs where the content script should be injected.
* **exclude\_matches**: Excludes specific URLs that would otherwise be included.
* **include\_globs**: Adds additional inclusion criteria for more fine-grained control.
* **exclude\_globs**: Provides additional exclusion criteria for even more control over where content scripts run.

By combining these attributes, you can achieve a highly tailored content script injection process, targeting only the URLs you want while excluding others based on specific criteria like file types, query parameters, or parts of the URL.