

Parsoid

For the historic version of Parsoid written in JavaScript (Node.js) until 2020, see [Parsoid/JS](#).

Parsoid is a PHP library that is used for converting back and forth between [wikitext](#) and HTML. It has been under development since 2012, originally written in JavaScript and built to support the [VisualEditor](#). It has been bundled with MediaWiki since version 1.35. In MediaWiki 1.47, Parsoid is planned to become MediaWiki's default parser, replacing the current [native parser](#), in a project known as [parser unification](#).

Technical details

Parsoid is an application which can translate back and forth between MediaWiki's [wikitext](#) syntax and an equivalent HTML/RDFa document model with enhanced support for automated processing and rich editing.

It has been under development by a team at the Wikimedia Foundation since 2012. It is currently used extensively by [VisualEditor](#), [Content translation](#) and [other applications](#).

Parsoid is intended to provide flawless back-and-forth conversion, i.e. to avoid information loss and also prevent "dirty diffs".

On Wikimedia wikis, for several applications, Parsoid is currently proxied behind [REStBase](#), which stores the HTML translated by Parsoid. It is expected that REStBase [will eventually be replaced](#) with a cache more tightly integrated with MediaWiki.

For more on the overall project, see [this blog post](https://blog.wikimedia.org/2013/03/04/parsoid-how-wikipedia-catches-up-with-the-web/) (<https://blog.wikimedia.org/2013/03/04/parsoid-how-wikipedia-catches-up-with-the-web/>) from March 2013. To read about the HTML model being used, see [MediaWiki DOM spec](#).

Parsoid was originally structured as a web service and written in JavaScript, making use of [Node.js](#). A [tech talk](#) from February 2019 (<https://www.youtube.com/watch?v=lQGfuLP9MqA>) ([slides](#)) and [blog post](#) describes the process of porting it to PHP. The Parsoid extension API is currently under active development; a [tech talk](#) from August 2020 describes this work.

GitHub Repository: <https://github.com/wikimedia/parsoid>

Wikimedia Foundation projects

Parsoid

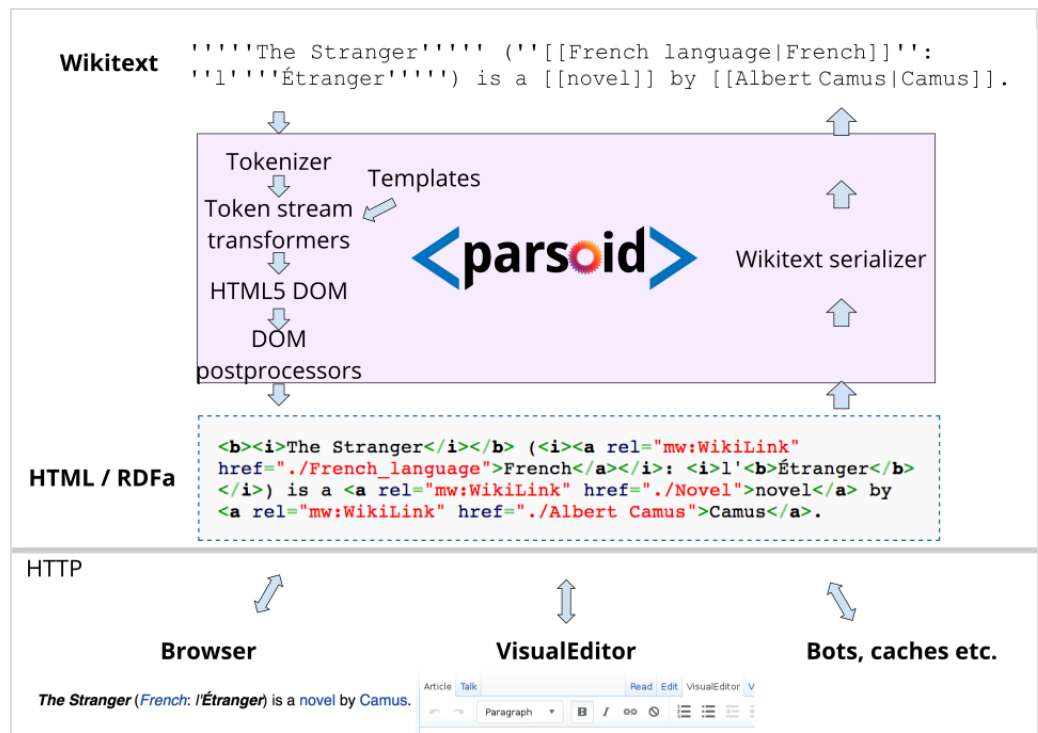


A bidirectional runtime wikitext parser. Converts back and forth between wikitext and HTML/XML DOM with RDFa.

Group:	Content Transform Team
Start:	2011-10
Team members:	Arlo Breault , C.Scott Ananian , Isabelle Hurbain-Palatin , Mateus Santos , Otto Šleger , Subramanya Sastry , Yiannis Giannelos
Backlog:	#parsoid
Lead:	C.Scott Ananian
Management:	Sérgio Lopes
Updates:	Parsoid/Releases
Progress:	Parsoid/Deployments

Usage

- [Parsoid/Releases](#) – List of releases made for Parsoid
- [Parsoid/API](#) – for the web API
- [MediaWiki DOM spec](#) – to make sense of the HTML that you get from the API, designed to be useful as a future storage format



A diagram of the Parsoid HTML5 + RDFa wiki runtime

- [Parsoid/LanguageConverter](#) – notes on Parsoid's implementation of [LanguageConverter](#)
- [Parsoid/Known differences with Core Parser output](#)

Installation

i This extension comes with MediaWiki 1.35 and later, so you do not need to download it. The remaining configuration instructions must still be followed.

Parsoid has been included in MediaWiki since version 1.35. No configuration is necessary to enable it.

Parsoid exports an internal REST API which was historically used by RESTBase and not accessible outside the WMF internal cluster. This is no longer required for Visual Editor or core read views, and the internal API is being deprecated and is planned for removal in MW 1.43.

Parsoid is nominally a composer library used by mediawiki core. If you still require the internal API for some reason, you can explicitly load Parsoid "as an extension" by adding the following to `LocalSettings.php`:

```
wfLoadExtension( 'Parsoid', "$IP/vendor/wikimedia/parsoid/extension.json" );
```

Any remaining third-party users of the internal Parsoid API are strongly encouraged to migrate to the core REST HTML page endpoint which provides equivalent functionality.

Development

Development happens in the [Parsoid Git repository](#). Code review happens in [Gerrit](#). See [Gerrit/Getting started](#) to set up an account for yourself.

If you use the MediaWiki-Vagrant development environment using a virtual machine, you can simply add the role `visualeditor` to it and it will set up a working Parsoid along with [Extension:VisualEditor](#).

The instructions below are for MediaWiki 1.35 or later. Check [Parsoid/JS](#) if you are running the old version of Parsoid written in JavaScript, and used for MW 1.34 and earlier.

Linking a developer checkout of Parsoid

In a standard MediaWiki installation, Parsoid is included from MediaWiki as a composer library, `wikimedia/parsoid`.

For development purposes you usually want to use a git checkout of Parsoid, and not the version bundled in MediaWiki core as a composer library. The following lines added to [LocalSettings.php](#) allow use of a git checkout of Parsoid (optionally), load the Parsoid REST API with `wfLoadExtension` (rather than using the version bundled in VisualEditor) and manually do the Parsoid configuration which is usually done by VisualEditor:

```
$parsoidInstallDir = 'vendor/wikimedia/parsoid'; # bundled copy
#$parsoidInstallDir = '/my/path/to/git/checkout/of/Parsoid';

// For developers: ensure Parsoid is executed from $parsoidInstallDir,
// (not the version included in mediawiki-core by default)
// Must occur *before* wfLoadExtension()
if ( $parsoidInstallDir !== 'vendor/wikimedia/parsoid' ) {
    function wfInterceptParsoidLoading( $className ) {
        // Only intercept Parsoid namespace classes
        if ( preg_match( '/(MW|Wikimedia\\\\)Parsoid\\\\\\\\/', $className ) ) {
            $fileName = Autoloader::find( $className );
            if ( $fileName !== null ) {
                require $fileName;
            }
        }
    }
    spl_autoload_register( 'wfInterceptParsoidLoading', true, true );
    // Autoloader::registerNamespaces was added in MW 1.39
    Autoloader::registerNamespaces( [
        // Keep this in sync with the "autoload" clause in
        // $parsoidInstallDir/composer.json
        'Wikimedia\\Parsoid\\' => "$parsoidInstallDir/src/",
    ] );

    wfLoadExtension( 'Parsoid', "$parsoidInstallDir/extension.json" );
    unset( $parsoidInstallDir );
}
```

These lines are not necessary for most users of VisualEditor, who can use VisualEditor's auto-configuration and the bundled Parsoid code included in MediaWiki, but they will be required for most developers.

If you're serving MediaWiki with Nginx, you'll need to also add something like this in your server block (Assuming your MediaWiki setup has its files residing in `/w/`):

```
location /w/rest.php/ {
    try_files $uri $uri/ /w/rest.php?$query_string;
}
```

If you are running Mediawiki using Docker, and linking your local Parsoid repository to Mediawiki, you need to map additional volume to the docker container in `docker-compose.override.yml` file in media wiki project. To do so, the simplest way is to create `docker-compose.override.yml` in mediawiki project and put code bellow inside (with path modification). If you already have `docker-compose.override.yml` file, modify it accordingly.

```
services:
  mediawiki:
    volumes:
      - ./:/var/www/html/w:cached
      - /my/path/to/git/checkout/of/Parsoid:/my/path/to/git/checkout/of/Parsoid
```

To test proper configuration, visit `{\$wgScriptPath}/rest.php/{\$domain}/v3/page/html/Main%20Page` where [\\$domain](#) is the hostname in your [\\$wgCanonicalServer](#). (Note that production WMF servers do not expose the Parsoid REST API to the external network.)

Running the tests

To run all parser tests and mocha tests:

```
$ composer test
```

The parser tests have quite a few options now which can be listed using `php bin/parserTests.php --help`.

If you have the environment variable `MW_INSTALL_DIR` pointing to a configured MediaWiki installation, you can run some additional tests with:

```
$ composer phan-integrated
```

Converting simple wikitext

You can convert simple wikitext snippets from the command line using the `parse.php` script in the `bin/` directory:

```
$ echo '[[Foo]]' | php bin/parse.php
```

The parse script has a lot of options. `php bin/parse.php --help` gives you information about this.

Debugging Parsoid (for developers)

See [Parsoid/Debugging](#) for debugging tips.

Continuous Integration

Parsoid is always available as a library since it is a composer dependency of MediaWiki core. But two pieces are not enabled:

- Parsoid ServiceWiring
- Parsoid's external REST api

The test runner `Quibble` would enable it if it detects `mediawiki/services/parsoid.git` has been cloned as part of the build. In which case it:

- points the autoloader for `Wikimedia\Paroid` to the cloned code (effectively replacing the version installed by composer)
- Load the extension `wfLoadExtension( 'Paroid', '/path/to/cloned/repo' );`

The ServiceWiring should be enabled in MediaWiki starting with 1.38.

The REST API would theoretically never get merged in MediaWiki: a) it has never been exposed to the public in production, it is an internal API used by RESTBase which is going away; b) it never has been security audited and c) it is redundant with the enterprise MediaWiki API. The solution will be for VisualEditor to invoke Parsoid directly via the VisualEditor Action API which would save a round trip through the REST API.

Loading the extension is thus a hack which enables using interfaces subject to change and which we don't really want people to use yet.

For most purposes, parsoid should thus not be added as a CI dependency, the only exception as of October 2021 is the Disambiguator MediaWiki extension.

Loading `parsoid` as an extension let us run MediaWiki integration test jobs against `mediawiki/services/parsoid.git` (such as `Quibble`, `apitestng`) and ensure Parsoid and MediaWiki work together.

An extension may be able to write tests with Parsoid even when the repository has not been cloned. Since it is a composer dependency of MediaWiki core the `MediaWiki\Paroid` namespace is available, but the service wiring part is not (it is `extension/src` in the Parsoid repository and exposed as the `\MWParoid` namespace). The `ParoidTestFileSuite.php` code would only run the parser tests if Parsoid has been loaded (which should be the default with MediaWiki 1.38).

For CI, Parsoid is tested against the tip of MediaWiki, whereas MediaWiki is tested with the Composer dependency. In case of a breaking change, the Parsoid change get merged first (which breaks its CI but not MediaWiki one) and MediaWiki get adjusted when Parsoid is updated. It is thus a one way change.

Release build

For MediaWiki release builds, we have an integration of Parsoid ServiceWiring into VisualEditor in order to have VisualEditor work without further configuration (beside a `wfLoadExtension( 'VisualEditor' )`). The release build also enables the REST API and hook everything us so that parsoid works out of the box.

This is done by copying a bit of Parsoid code into VisualEditor which is not in the master branch of VisualEditor since that would be obsolete as soon as Parsoid is updated. Instead the code is maintained in two places.

Technical documents

- [Parsoid/Internals](#) – documentation about Parsoid internals with links to other details.
- [PHP Porting notes](#) and [help-wanted tasks](#)
- [Parsoid deployment agenda on Wikimedia cluster](#) (code normally deployed every Monday and Wednesday between 1pm - 1:30pm PST)
- [Parsoid/Round-trip testing](#) – The round-trip testing setup we are using to test the wikitext -> HTML DOM -> wikitext round-trip on actual Wikipedia content.
- [Parsoid/Visual Diffs Testing](#) – Info about visual diff testing for comparing Parsoid's html rendering with php parser's html rendering + a testreduce setup for doing mass visual diff tests.
- [Parsoid/limitations](#) – Limitations in Parsoid, mainly contrived templating (ab)uses that don't matter in practice. Could be extended to be similar to [the preprocessor upgrade notes](#) (**Might need updating**)
- [Parsoid/Bibliography](#) – Bibliography of related literature

Links for Parsoid developers

- See [Parsoid/Debugging](#) for debugging tips.
- [Upgrading or adding packages to Parsoid](#)
- See these instructions (<https://github.com/wikimedia/parsoid/blob/39ea4e49783b1f88326d6ebddd14e08bb39cb649/tools/sync-parserTests.js#L5-L34>) for syncing Parsoid's copy of parser tests to/from core
- [Parsoid has a limited library interface for invoking it programatically.](#)
- [Tech Talk about Retargeting extensions to work with Parsoid](#) (<https://www.youtube.com/watch?v=IS1xPkERWCM>)
- [So you want your extension to work with Parsoid](#)
- [Parsoid HTML Specification Versioning](#)
- [So you are going to change Parsoid output](#)

Links for Parsoid deployers (to the Wikimedia cluster)

- [Parsoid/Deployments](#)
- [RT testing commits](#) (<https://parsoid-rt-tests.wikimedia.org/commits>) (useful to check regressions and fixes)
- [Deployment instructions for Parsoid](#)
- [Kibana dashboard](#) (<https://logstash.wikimedia.org/app/kibana#/dashboard/parsoid>)
- [Grafana dashboard for wt2html metrics](#) (<https://grafana.wikimedia.org/d/TCWoeymIk/parsoid-timing-wt2html-entry>)
- [Grafana dashboard for html2wt metrics](#) (<https://grafana.wikimedia.org/d/jmaNtVMlZ/parsoid-timing-html2wt-entry>)
- [Prometheus breakdown for the Parsoid cluster on eqiad](#) (<https://grafana.wikimedia.org/d/000000607/cluster-overview?orgId=1&from=now-24h&to=now&var-datasource=eqiad%20prometheus%2Fops&var-cluster=parsoid&var-instance=All&var-site=eqiad>)

- Prometheus breakdown for the Parsoid cluster on codfw (<https://grafana.wikimedia.org/d/000000607/cluster-overview?orgId=1&from=now-24h&to=now&var-datasource=codfw%20prometheus%2Fops&var-cluster=parsoid&var-instance=All&var-site=codfw>)
- [Jenkins Job Builder docs](#) for updating jenkins jobs

History

The original application was written in JavaScript (using Node.js) and started running on the Wikimedia cluster in December 2012. In 2019, Parsoid was ported to PHP, and the PHP version replaced the JS version on the Wikimedia cluster in December 2019. Parsoid is being integrated into core MediaWiki, with the goal of eventually replacing MediaWiki's current native parser. In early 2024, Parsoid began to be used on some production wikis of the Wikimedia Foundation as the default parser for read views. You can see them in the production config list (<https://noc.wikimedia.org/conf/highlight.php?file=dblists/parsoidrendered.dblist>).

Parsoid (the PHP version) has been natively bundled with MediaWiki since version 1.35, released in September 2020. For non-Wikimedia installations, Parsoid/JS was supported until the end-of-life of MediaWiki 1.31 (LTS) in September 2021.

FAQ

- How do I see if a page was rendered with Parsoid?
 - The footer will say "Page was rendered with Parsoid".
- How do I set a page to temporarily render with Parsoid?
 - Install [Extension:ParserMigration](#)
 - Add `?useparsoid=1` to the URL
- How do I get an entire wiki to render with Parsoid?
 - Install [Extension:ParserMigration](#)
 - Set `$wgParserMigrationEnableParsoidArticlePages = true;`
 - Set `$wgParserMigrationEnableParsoidDiscussionTools = true;`
- Which Wikimedia wikis render with Parsoid?
 - The list is at [parsoidrendered.dblist](#). As of August 2025, it is approximately 250 of the 1000 wikis. Rollout will continue until all wikis are using Parsoid.

See also

- [API](#)
- [RESTBase](#) – a caching / storing API proxy for page HTML translated by Parsoid
- [Parser 2011/Parser plan](#) – Early (now relatively old) design ideas and issues from 2011
- [Special:PrefixIndex/Parsoid/](#) – Parsoid-related pages on this wiki
- [Extension:ParsoidBatchAPI](#) (archived)
- [parsoid-jsapi](#) (<https://github.com/wikimedia/parsoid-jsapi>) – a high-level interface for extraction and transformation of wikitext, similar to the [mwparserfromhell](#) (<https://github.com/earwig/mwparserfromhell>)

mhell) API.

- [Alternative parsers](#)
- [Parsoid/Parser Unification](#)

External links

- [Source code](#) (GitHub mirror (<https://github.com/wikimedia/parsoid/>))
- [JS Documentation](#) (old version of Parsoid)
- [PHP Documentation](#)
- [Parsoid on the Wikimedia Commons](#)

Contact

If you need help or have questions/feedback, you can contact us in **#mediawiki-parsoid** (<ircs://irc.libera.chat:6697/mediawiki-parsoid>) [connect](https://web.libera.chat/?channel=#mediawiki-parsoid) (<https://web.libera.chat/?channel=#mediawiki-parsoid>) or **the wikitext-l mailing list** (<https://lists.wikimedia.org/mailman/listinfo/wikitext-l>). If all that fails, you can also contact us by email at [content-transform-team](mailto:content-transform-team@wikimedia.org) at the [wikimedia.org](https://www.wikimedia.org) domain.

Parsoid is maintained by **[Content Transform Team](#)**.

Get help:



-  [#mediawiki-parsoid](ircs://irc.libera.chat:6697/mediawiki-parsoid) (<ircs://irc.libera.chat:6697/mediawiki-parsoid>) [connect](https://web.libera.chat/?channel=#mediawiki-parsoid) (<https://web.libera.chat/?channel=#mediawiki-parsoid>) ← [Live chat \(IRC\)](#)
-  [wikitext-l](https://lists.wikimedia.org/mailman/listinfo/wikitext-l) (<https://lists.wikimedia.org/mailman/listinfo/wikitext-l>) ← [Mailing list](#)
- [Talk:Parsoid](#) ← [Talk page](#)
-  [parsoid](https://phabricator.wikimedia.org/maniphest/task/edit/form/1/?tags=parsoid) (add task (<https://phabricator.wikimedia.org/maniphest/task/edit/form/1/?tags=parsoid>)) ← [Phabricator project page](#)

Retrieved from "<https://www.mediawiki.org/w/index.php?title=Parsoid&oldid=8502644>"