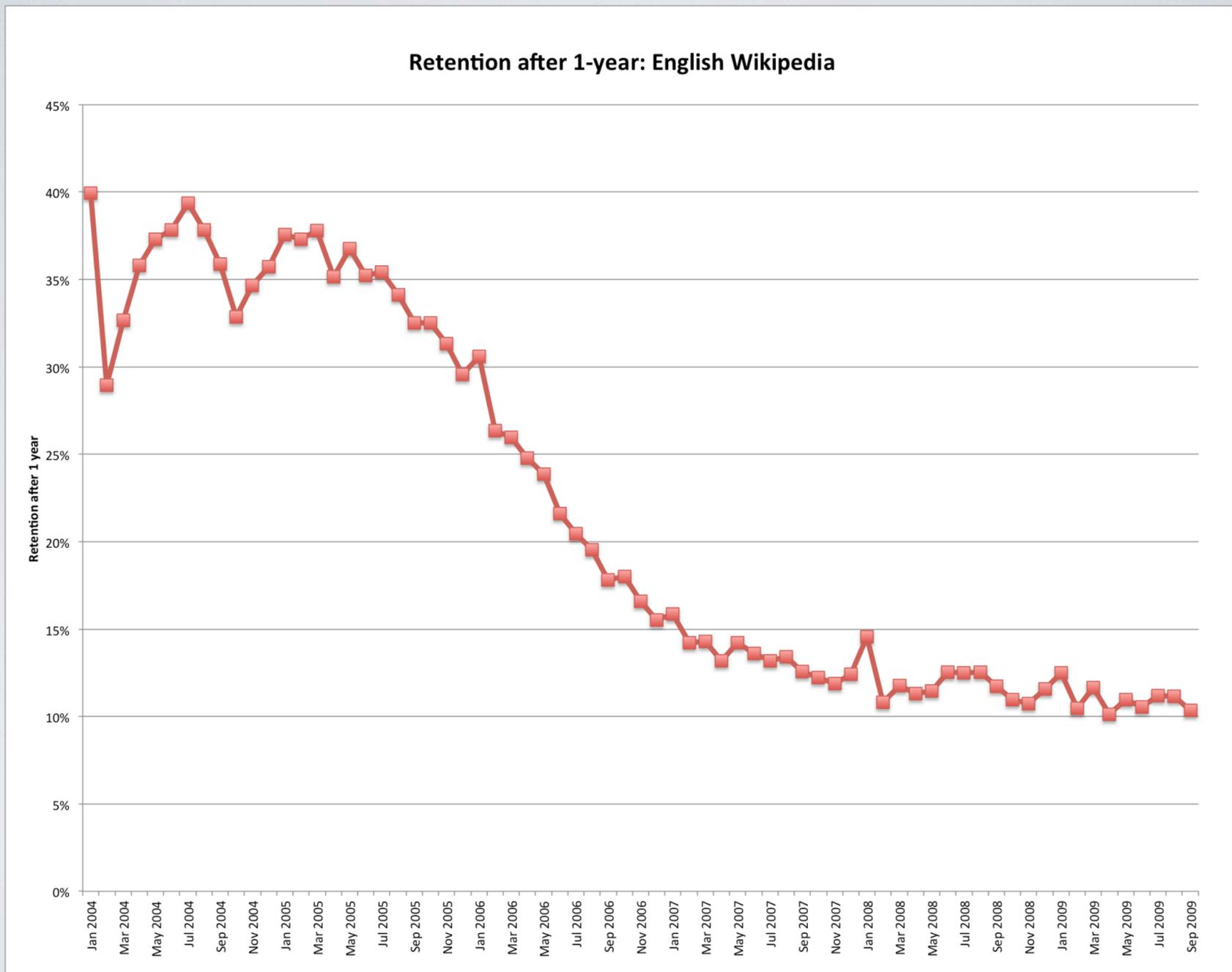


IN-LINE EDITING: A NEW APPROACH TO EDITING WIKIS

Jan Paul Posma



Wikipedia has a problem. New volunteers stop contributing earlier than they used to.



This chart shows the percentage of new volunteers that are still active after a year. This percentage started to drop in 2005/2006. There are several reasons for this drop, and one of them is the interface to edit pages.

W Bezig met bewerken van Gro x

nl.wikipedia.org/w/index.php?title=Groningen_(stad)&action=edit

JanPaul123 Overleg Voorkeuren Volglijst Bijdragen Afmelden

Artikel Overleg Lezen Bewerken Geschiedenis Zoeken

Bezig met bewerken van Groningen (stad)

V C Geavanceerde functies Speciale tekens Hulp

```
{{Infobox plaats in Nederland
| soort      = Stad
| naam       = Groningen
| bestandsnaam vlag = Flag Groningen city.svg | vlagartikel= vlag van Groningen (stad)|vlag van Groningen
| bestandsnaam wapen = Groningen stad wapen.svg | wapenartikel=wapen van Groningen (stad)|wapen van Groningen
| provincie  = {{NL-vlag|Groningen (provincie)}}
| gemeente  = {{NL-vlag|Groningen (gemeente)}}
| opp       =
| inwoners  = {{Inwonertal NL gemeente|0014}} ([[Groningen (gemeente)|gemeente]])
| datum inwoners = {{Inwonertal NL datum}}
| dichtheid =
| verkeersader = {{A-wegNL|7}} {{A-wegNL|28}} {{N-wegNL|46}}
| netnummer = 050
| web       = [http://portal.groningen.nl/ www.groningen.nl]
| lat_deg   = 53
| lat_min   = 13
| lat_sec   = 0
| lon_deg   = 6
| lon_min   = 34
| lon_sec   = 0
| foto      = [[Bestand:Martini Toren.JPG|250px|De Martinitoren]]
| ondertitel = De 97 meter hoge [[Martinitoren]] ("d'Olle Grieze") aan de Grote Markt
}}
{{Gesproken Wikipedia klein|NL-Groningen (stad)-article.ogg|23544539}}
'''Groningen''' {{Audio|000_Groningen.ogg|<small>uitspraak</small>}} ([[Gronings]]: "Grunnen" of "Stad", [[Westerlauwers Fries|Fries]]: "Grins") is de [[hoofdstad]] van de [[Nederland]]se [[Groningen (provincie)|provincie Groningen]] en de grootste kern in de [[Groningen (gemeente)|gelijknamige gemeente]]. De plaats wordt in grote delen van Noord-Nederland ook kortweg aangeduid met "Stad". De gemeente Groningen had per {{Inwonertal NL datum}} {{Inwonertal NL gemeente|0014}} inwoners (bron: [[Centraal Bureau voor de Statistiek|CBS]]). Van Groningen zijn geen geschreven [[stadsrechten]] bekend.
```

Door uw bewerking op te slaan, gaat u akkoord met het onherroepelijk vrijgeven van uw bijdrage onder de licentie [Creative Commons Naamsvermelding/Gelijk delen](#) en de [GFDL](#). U gaat ermee akkoord door hergebruikers minimaal genoemd te worden via een hyperlink of URL naar de pagina waaraan u bijdraagt. Zie de [gebruiksvoorwaarden](#) voor meer informatie.

With this interface the source code of a page can be edited, in a language called 'wikitext'. This editing has become a lot harder over time, because the pages became more and more complex.

W Bezig met bewerken van Gro x

nl.wikipedia.org/w/index.php?title=Groningen_(stad)&action=edit

JanPaul123 Overleg Voorkeuren Volglijst Bijdragen Afmelden

Artikel Overleg Lezen Bewerken Geschiedenis Zoeken

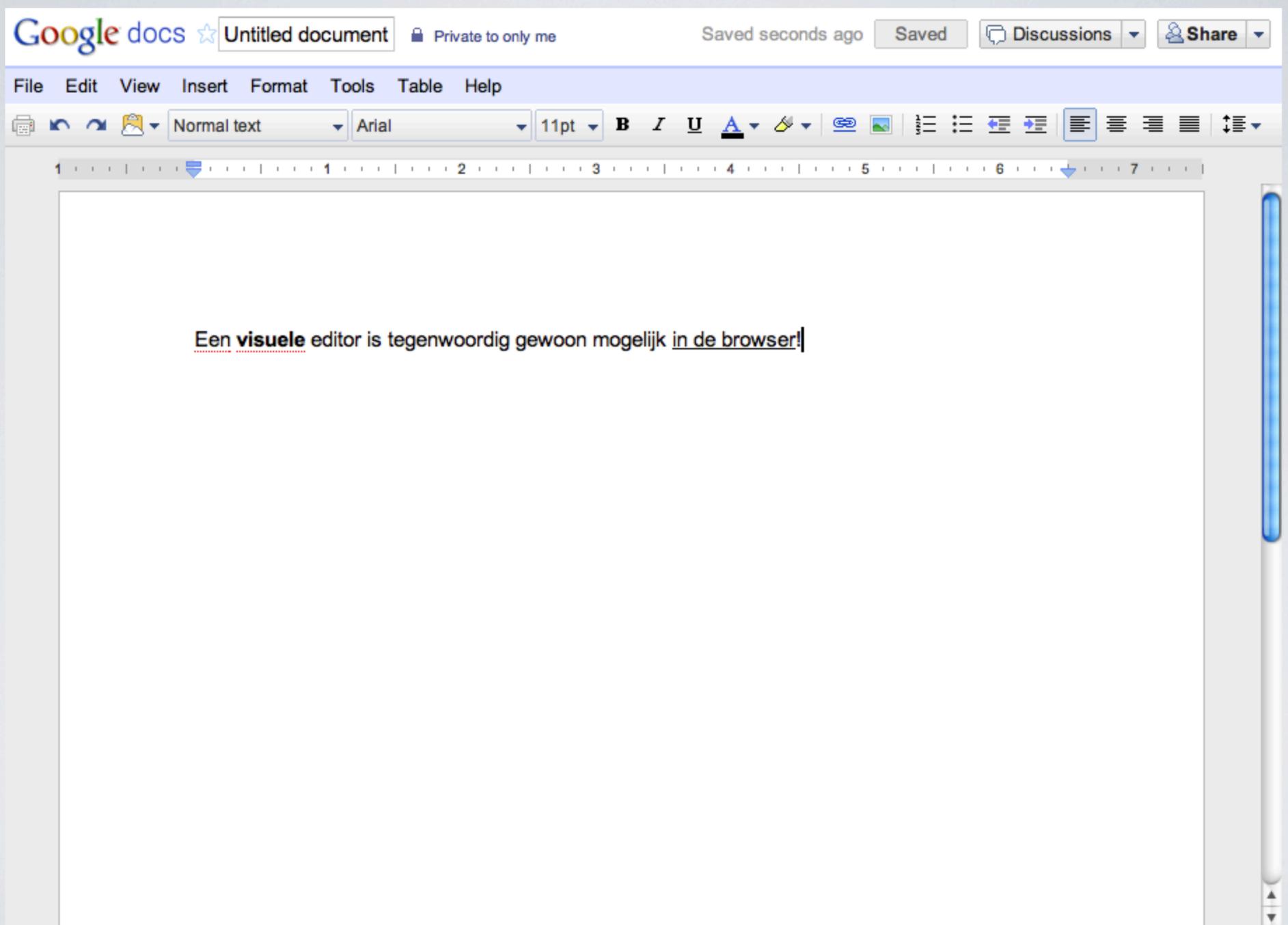
Bezig met bewerken van Groningen (stad)

V C Geavanceerde functies Speciale tekens Hulp

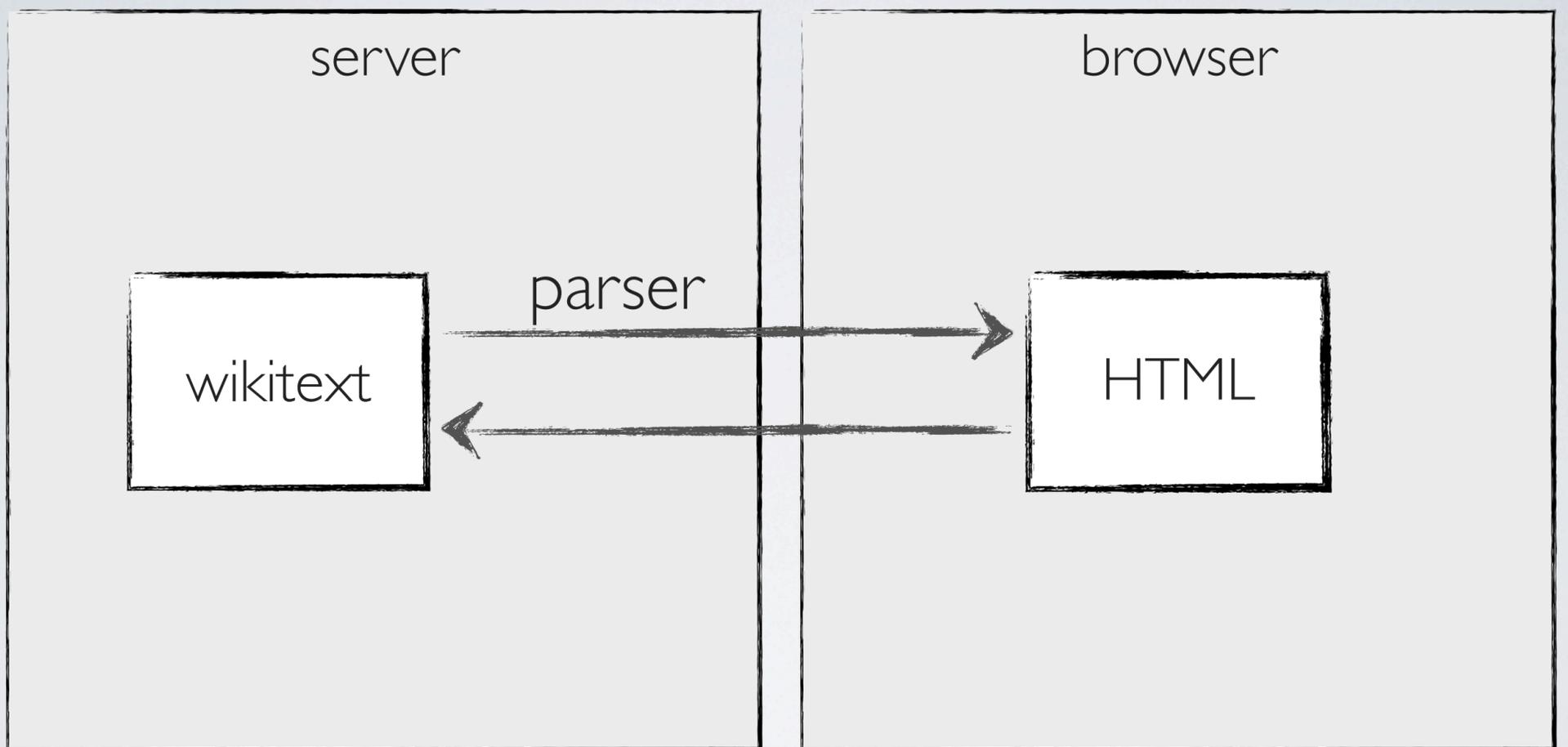
```
{{Infobox plaats in Nederland
| soort          = Stad
| naam           = Groningen
| bestandsnaam vlag = Flag Groningen city.svg | vlagartikel= vlag van Groningen (stad)|vlag van Groningen
| bestandsnaam wapen = Groningen stad wapen.svg | wapenartikel=wapen van Groningen (stad)|wapen van Groningen
| provincie      = {{NL-vlag|Groningen (provincie)}}
| gemeente      = {{NL-vlag|Groningen (gemeente)}}
| opp            =
| inwoners      = {{Inwonertal NL gemeente|0014}} ([[Groningen (gemeente)|gemeente]])
| datum inwoners = {{Inwonertal NL datum}}
| dichtheid     =
| verkeersader  = {{A-wegNL|7}} {{A-wegNL|28}} {{N-wegNL|46}}
| netnummer    = 050
| web           = [http://portal.groningen.nl/ www.groningen.nl]
| lat_deg      = 53
| lat_min      = 13
| lat_sec      = 0
| lon_deg      = 6
| lon_min      = 34
| lon_sec      = 0
| foto         = [[Bestand:Martini Toren.JPG|250px|De Martinitoren]]
| ondertitel    = De 97 meter hoge [[Martinitoren]] ("d'Olle Grieze") aan de Grote Markt
}}
{{Gesproken Wikipedia klein|NL-Groningen (stad)-article.ogg|23544539}}
'''Groningen''' Audio000 Groningen.ogg <small>uitspraak</small> ([[Gronings]]: "Grunnen" of "Stad", [[Westerlauwers Fries|Fries]]: "Grins") is de [[hoofdstad]] van de [[Nederland]]se [[Groningen (provincie)|provincie Groningen]] en de grootste kern in de [[Groningen (gemeente)|gelijknamige gemeente]]. De plaats wordt in grote delen van Noord-Nederland ook kortweg aangeduid met "Stad". De gemeente Groningen had per {{Inwonertal NL datum}} {{Inwonertal NL gemeente|0014}} inwoners (bron: [[Centraal Bureau voor de Statistiek|CBS]]). Van Groningen zijn geen geschreven [[stadsrechten]] bekend.
```

Door uw bewerking op te slaan, gaat u akkoord met het onherroepelijk vrijgeven van uw bijdrage onder de licentie [Creative Commons Naamsvermelding/Gelijk delen](#) en de [GFDL](#). U gaat ermee akkoord door hergebruikers minimaal genoemd te worden via een hyperlink of URL naar de pagina waaraan u bijdraagt. Zie de [gebruiksvoorwaarden](#) voor meer informatie.

It can be compared with having users edit a document in LaTeX, while they are used to working with Microsoft Word. Why is it not possible for users to edit Wikipedia pages in a visual manner, in a way similar to Word?



After all, it is possible nowadays to do visual editing in the browser, see for example Google Docs. The fact that we cannot do this on Wikipedia has everything to do with this language pages are edited and saved in: wikitext. Let's look first at how such a visual editor can be implemented in wikis.



To convert the wikitext source code of a page into a web page, a parser is used. This parser converts the text to HTML, which can be showed by the browser. If we want to edit this document, we also have to be able to convert the page back to wikitext.

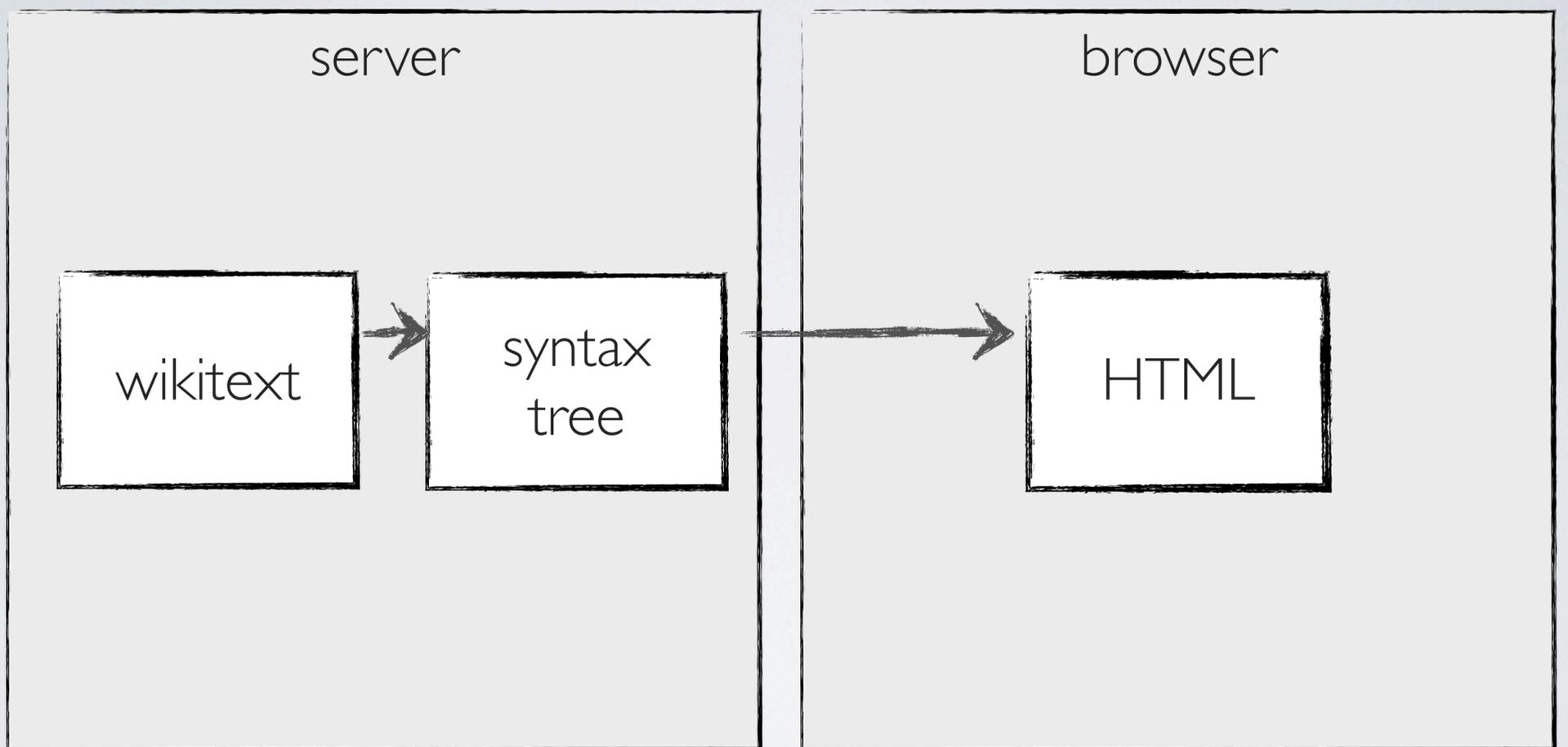
Saving the page in wikitext is vital, as many current volunteers for Wikipedia and other projects are used to editing wikitext, so editing in this way must be preserved in order to keep these existing volunteers happy.

Internally, the parser generates a syntax tree, for example by using a formal grammar. This is a mathematically well-defined way of expressing how valid wikitext looks like, and is used by computer scientists for decades, for example to define programming languages. The syntax tree is a direct representation of the wikitext based on this formal grammar. This syntax tree is then used to generate the HTML.

When editing the page, one option is to send the syntax tree to the client. Now, every time the user interacts with the page by changing it, the syntax tree is updated to reflect this change. Then, the browser can regenerate the HTML in order to show the page again in it changed state.

Because the syntax tree is a direct representation of wikitext, it can be converted back to raw wikitext when saving the page.

The problem is, that the parser used at Wikipedia, is not based on a formal grammar. There isn't even any way to convert edited HTML back to wikitext! Because of the lack of such a grammar, it is hard to reconstruct some kind of "reverse parser". It turns out that it is even very tough to reconstruct the grammar of the wikitext language.



To convert the wikitext source code of a page into a web page, a parser is used. This parser converts the text to HTML, which can be showed by the browser. If we want to edit this document, we also have to be able to convert the page back to wikitext.

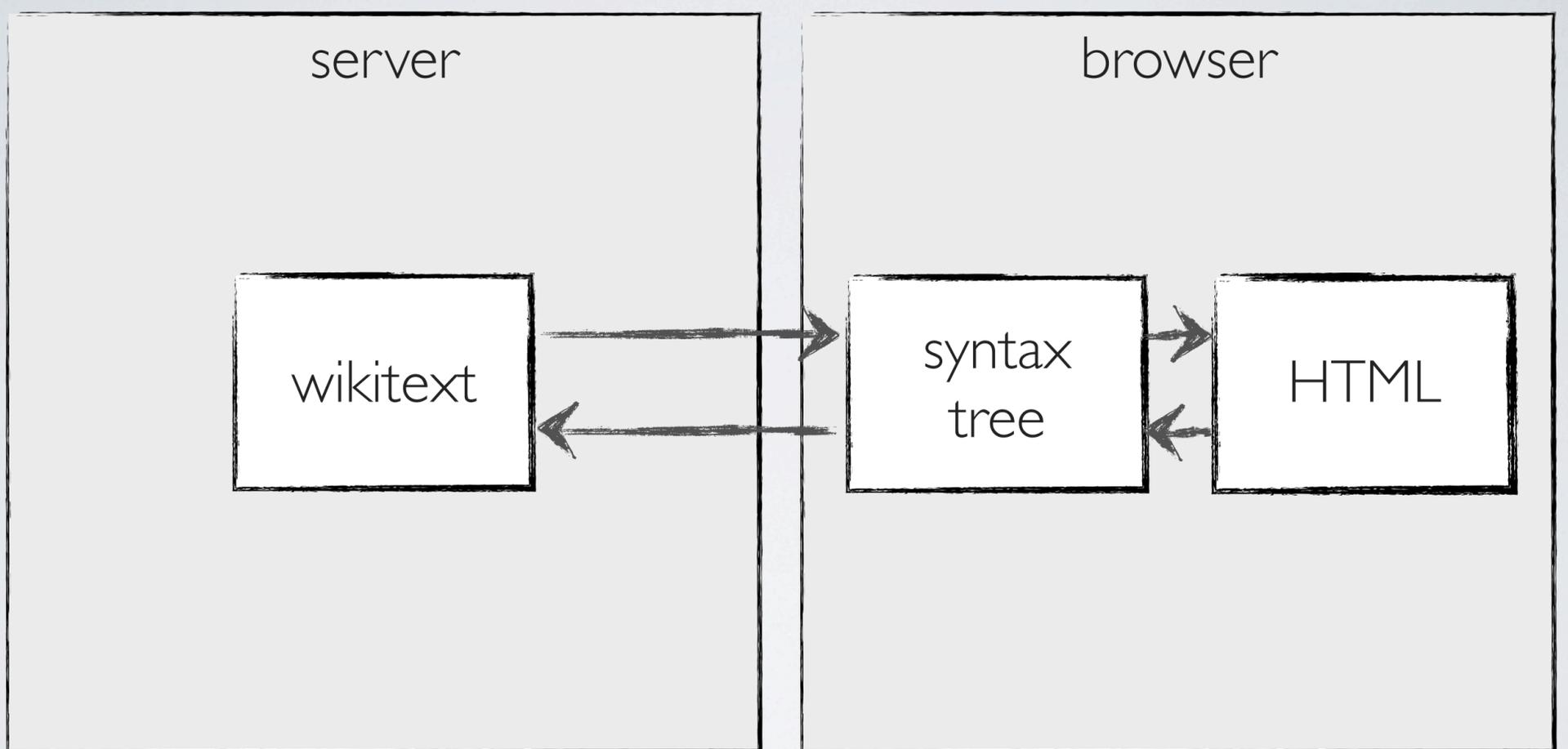
Saving the page in wikitext is vital, as many current volunteers for Wikipedia and other projects are used to editing wikitext, so editing in this way must be preserved in order to keep these existing volunteers happy.

Internally, the parser generates a syntax tree, for example by using a formal grammar. This is a mathematically well-defined way of expressing how valid wikitext looks like, and is used by computer scientists for decades, for example to define programming languages. The syntax tree is a direct representation of the wikitext based on this formal grammar. This syntax tree is then used to generate the HTML.

When editing the page, one option is to send the syntax tree to the client. Now, every time the user interacts with the page by changing it, the syntax tree is updated to reflect this change. Then, the browser can regenerate the HTML in order to show the page again in it changed state.

Because the syntax tree is a direct representation of wikitext, it can be converted back to raw wikitext when saving the page.

The problem is, that the parser used at Wikipedia, is not based on a formal grammar. There isn't even any way to convert edited HTML back to wikitext! Because of the lack of such a grammar, it is hard to reconstruct some kind of "reverse parser". It turns out that it is even very tough to reconstruct the grammar of the wikitext language.



To convert the wikitext source code of a page into a web page, a parser is used. This parser converts the text to HTML, which can be showed by the browser. If we want to edit this document, we also have to be able to convert the page back to wikitext.

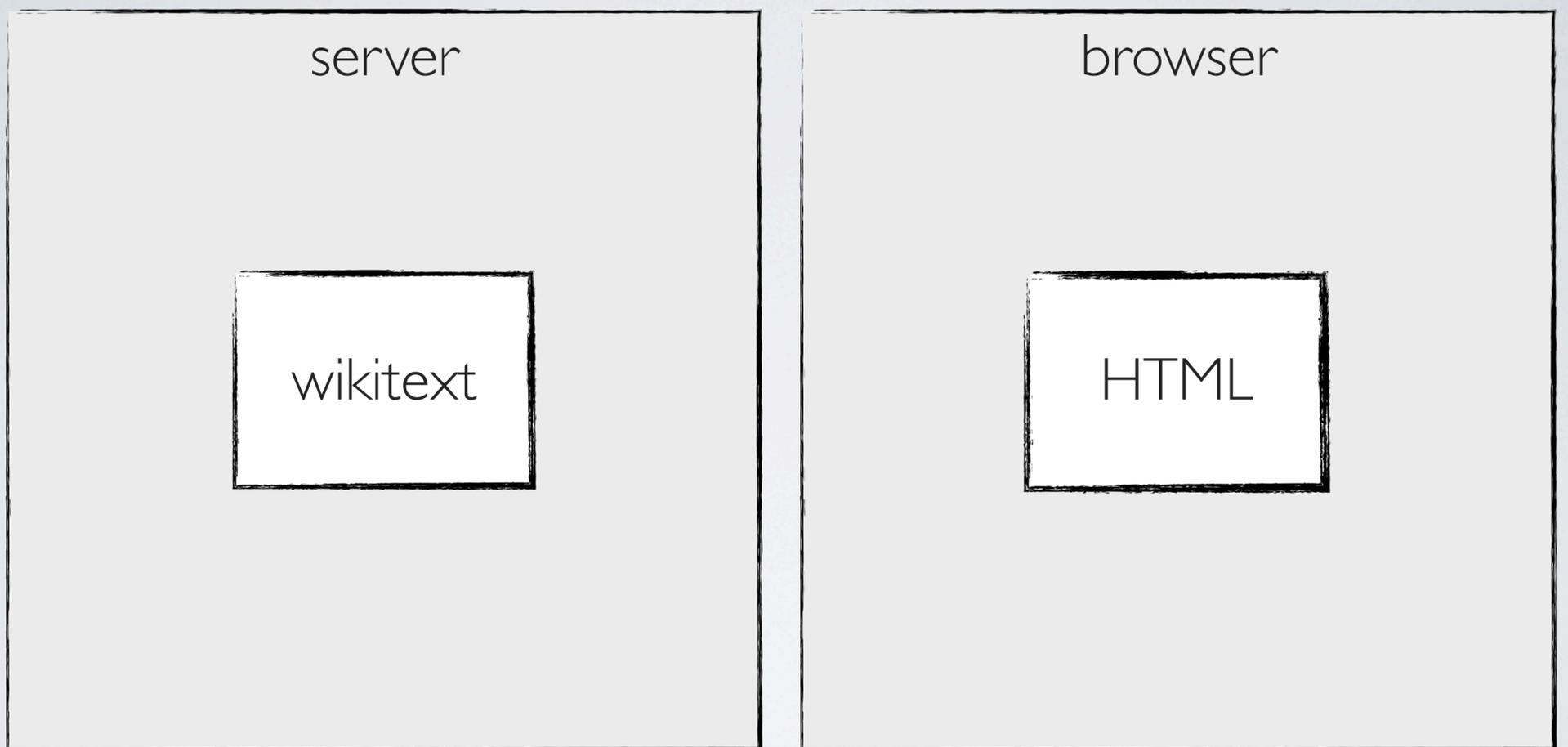
Saving the page in wikitext is vital, as many current volunteers for Wikipedia and other projects are used to editing wikitext, so editing in this way must be preserved in order to keep these existing volunteers happy.

Internally, the parser generates a syntax tree, for example by using a formal grammar. This is a mathematically well-defined way of expressing how valid wikitext looks like, and is used by computer scientists for decades, for example to define programming languages. The syntax tree is a direct representation of the wikitext based on this formal grammar. This syntax tree is then used to generate the HTML.

When editing the page, one option is to send the syntax tree to the client. Now, every time the user interacts with the page by changing it, the syntax tree is updated to reflect this change. Then, the browser can regenerate the HTML in order to show the page again in it changed state.

Because the syntax tree is a direct representation of wikitext, it can be converted back to raw wikitext when saving the page.

The problem is, that the parser used at Wikipedia, is not based on a formal grammar. There isn't even any way to convert edited HTML back to wikitext! Because of the lack of such a grammar, it is hard to reconstruct some kind of "reverse parser". It turns out that it is even very tough to reconstruct the grammar of the wikitext language.



To convert the wikitext source code of a page into a web page, a parser is used. This parser converts the text to HTML, which can be showed by the browser. If we want to edit this document, we also have to be able to convert the page back to wikitext.

Saving the page in wikitext is vital, as many current volunteers for Wikipedia and other projects are used to editing wikitext, so editing in this way must be preserved in order to keep these existing volunteers happy.

Internally, the parser generates a syntax tree, for example by using a formal grammar. This is a mathematically well-defined way of expressing how valid wikitext looks like, and is used by computer scientists for decades, for example to define programming languages. The syntax tree is a direct representation of the wikitext based on this formal grammar. This syntax tree is then used to generate the HTML.

When editing the page, one option is to send the syntax tree to the client. Now, every time the user interacts with the page by changing it, the syntax tree is updated to reflect this change. Then, the browser can regenerate the HTML in order to show the page again in it changed state.

Because the syntax tree is a direct representation of wikitext, it can be converted back to raw wikitext when saving the page.

The problem is, that the parser used at Wikipedia, is not based on a formal grammar. There isn't even any way to convert edited HTML back to wikitext! Because of the lack of such a grammar, it is hard to reconstruct some kind of "reverse parser". It turns out that it is even very tough to reconstruct the grammar of the wikitext language.

WikiOnCD

FlexBisonParse

wik2dict

Waikiki

Magnus' magic wiki-to-XML converter

mwlib

Textile-J

Marker

Wiki2XML

wiki2pdf

WikiCloth

JAMWiki

WikiTaxi

Sweble

mw2html

WikiPDF

XWiki

Kiwi

Perl Wikipedia Toolkit

Mediawiki2HTML Machine

Wikiwyg

YaCy

Live Preview

Wikifilter

TomeRaider export

WikiModel

Tero-dump

Mylyn WikiText

Bliki engine

WikiPress Publisher

MessageParser

It turns out to be so tough, that more than 30 attempts at building a compatible parser have been made, but none of these are 100% compatible with the parser used by Wikipedia. This is a large problem, not only for reusing data on Wikipedia – which is why most of these parsers have been made – but also for building a reliable visual editor.

my solution: a compromise

My solution to this problem is a kind of a compromise. Users can still edit the page visually, but I remove the translation back to wikitext. This means that users do have to edit raw wikitext.

That looks like this. The page looks the same as when simply viewing it, only parts of the page are colored blue. By clicking such an element, a popup is shown containing the original wikitext of this element. The idea is that this is a lot less scary for new users, as there is only a minimal amount of wikitext syntax in, for example, a single sentence. Also, the user can directly manipulate the page, and instantly see how the changes in the wikitext reflect in the actual page.

Groningen (stad)

Groningen is de **hoofdstad** van de **Nederlandse provincie Groningen**, het **bisdom Groningen**, de **gemeente Groningen** en de studentenhoofdstad van het noorden. De plaats wordt in grote delen van Noord-Nederland ook kortweg aangeduid met *Stad*. De gemeente Groningen had per 1 januari 2010 187.623 inwoners. Hoewel de stad zelf een lager inwoneraantal heeft, behoort Groningen tot de tien **grootste steden in Nederland**. Groningen is een zeer fietsvriendelijke stad!

Groningen behoort tot de **oudste steden** van **Nederland**. Door zijn relatief geïsoleerde ligging, ten opzichte van de opeenvolgende feitelijk machtscentra (Utrecht, Brussel en Den Haag), was de stad historisch gezien vooral op zich zelf en de directe omgeving aangewezen. Als **Hanzestad** maakte Groningen deel uit van het Noord-Duitse handelsnetwerk. Later werd de stad vooral een regionaal marktcentrum. De stad ontwikkelde zich en was de facto een **stadstaat**.^{[1][2][3][4]} Sinds de **Republiek** hoorde Groningen nominaal bij Nederland, maar tot aan de Franse tijd bleef Groningen feitelijk een autonome stad, die heer was in het grootste deel van de **provincie**.

V C  [Gevorderde functies](#) [Speciale tekens](#) [Hulp](#)

Tegenwoordig is Groningen een stad met veel gevarieerde [[handel (economie)|handel]] en [[industrie]].

[Preview](#) [Cancel](#)

Ten noordoosten van de Grote Markt ligt het **Martinikerkhof** met het **provinciehuis** en het **Prinsenhof**. Achter dat pand ligt de fraaie **Prinsentuin** met hierin een van de rijkst vormgegeven **zonnewijzers** van het land.

Aan de westkant van de Grote Markt ligt de **Vismarkt**. Met name de zuidkant van dit plein bevat een aantal fraaie panden. Het plein wordt aan de westkant begrensd door de **Korenbeurs**, een verwijzing naar de rijkdom van de **provincie** in de **19e eeuw** die gebaseerd was op de graanteelt op de vruchtbare **zeeklei**.

Achter de Korenbeurs verheft zich de **Der Aa-kerk**, na de Martinikerk de tweede parochiekerk van de historische stad. Iets verder naar het westen, langs het water, of in het Gronings 'het diep', staat langs de **Hoge der Aa** een aantal fraaie pakhuizen en woonhuizen. Ook langs de **Noorderhaven** staan dergelijke panden.

Vanaf de Grote Markt naar het noorden loopt de **Oude Boteringestraat** met een van de oudste huizen van de stad, het **Calmershuis** en de oude **Rechtbank**. Over de **diepenring** loopt de straat door als Nieuwe Boteringestraat en loopt dan langs de **Nieuwe Kerk**, uit **1660** en de eerste kerk in Groningen die gebouwd is voor de Protestantse eredienst.

Een bijzonder aspect van Groningen zijn de vele **hofjes**, in Groningen gasthuizen genoemd. De bekendste zijn het **Pelstergasthuis** en het **Pepergasthuis**. Het **hoofdstation** uit 1896, direct buiten de zogenaamde **diepenring**, staat bekend als een van de fraaiste stations in **Nederland**.

Aan de zuid-west kant staat het markante **Gasuniegebouw**, hoofdkantoor van de **Gasunie**. Het is gebouwd in de stijl van de **organische architectuur**. Andere interessante moderne architectuur is te vinden in de wijk **Hooimse Meer**. Hier is het **Wall House #2** te vinden, het enige gerealiseerde ontwerp uit een serie die wel als het meesterstuk van de Amerikaanse architect **John Hejduk** wordt beschouwd.^[7]

Kunst en cultuur

Groningen is de grootste, Groningers zouden zeggen enige, stad in het noorden van **Nederland**. De stad vervult daardoor een centrumfunctie, die nog versterkt wordt door de aanwezigheid van de **universiteit** en het grote aantal jonge inwoners die dat meebrengt. Dat uit zich in een zeer gevarieerd cultureel aanbod.

Musea

Het **Groninger Museum** is zonder meer het belangrijkste museum van de stad. Oorspronkelijk was dit een vrij traditioneel provinciaal streekmuseum maar het heeft zich ontwikkeld tot één van de meest vernieuwende musea in Nederland, met name sinds de nieuwbouw, op een kunstmatig eiland in het **Verbindingskanaal**.

De stad heeft nog enkele andere musea. Zo is er het **Universiteitsmuseum**, waar tentoonstellingen over zowel de geschiedenis van de universiteit als de nieuwste ontwikkelingen in het onderzoek te zien zijn. De geschiedenis van Groningen, stad en provincie, op zeevaartgebied is het thema van het **Noordelijk Scheepvaartmuseum** in de Brugstraat. Het **Grafisch Museum** is gewijd aan de grafische kunst en druktechnieken.

Kunst in de openbare ruimte

Er zijn in Groningen veel beeldhouwwerken (circa 400) in de openbare ruimte te zien waaronder enkele van **Wladimir de Vries**. Ook het beeldenproject van de **Rijksuniversiteit Groningen** heeft meerdere kunstwerken opgeleverd. Verder bevindt zich bij iedere uitgaande weg van de gemeente een beeld of installatie van het zogenaamde "**project Stadsmarkeringen**" uit 1990.

Parken

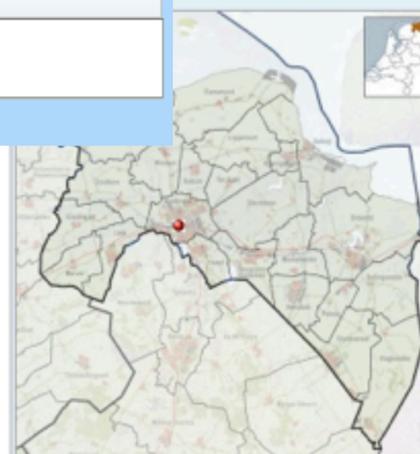
Het grootste park van de stad is van oudsher het **Stadspark**, aan de westzijde van de stad. De aanleg van de westelijke **Ringweg** heeft het park in tweeën gedeeld. In het park

Groningen

Stad in **Nederland** 



[\(Details\)](#)



Situering

Provincie 22px **Groningen**
Gemeente 22px **Groningen**
Coördinaten 53°13'N 6°34'E 

Algemeen

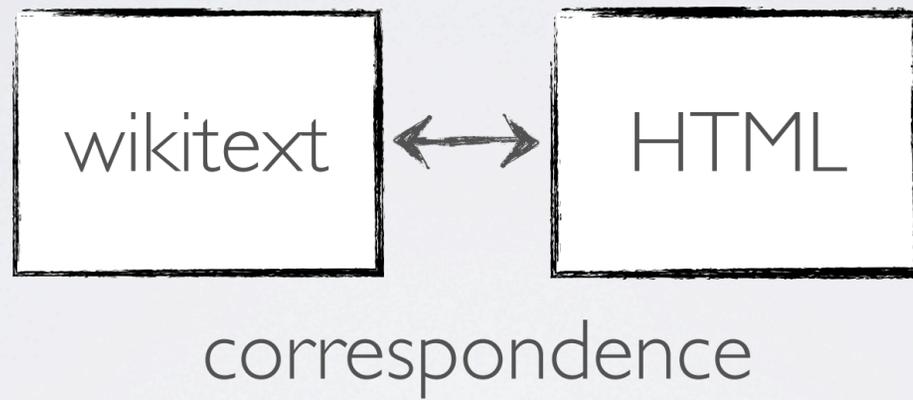
Inwoners (1 april 2011) 190.334 (**gemeente**)
Overig
Netnummer 050
Belangrijke verkeersaders   
Website www.groningen.nl 

Foto's



My solution to this problem is a kind of a compromise. Users can still edit the page visually, but I remove the translation back to wikitext. This means that users do have to edit raw wikitext.

That looks like this. The page looks the same as when simply viewing it, only parts of the page are colored blue. By clicking such an element, a popup is shown containing the original wikitext of this element. The idea is that this is a lot less scary for new users, as there is only a minimal amount of wikitext syntax in, for example, a single sentence. Also, the user can directly manipulate the page, and instantly see how the changes in the wikitext reflect in the actual page.



Such an in-line editing interface as this one can be constructed without a formal grammar. The only thing that we need, is a correspondence between certain elements in the final HTML page, and the original wikitext. This means that we only need to know which pieces of wikitext belong to a sentence, a paragraph, a reference, or a template.

```
A sentence. Another.  
<ref>Reference</ref>  
Sentence with a  
[[link]].  
  
{{this is a template}}
```



```
<p>  
A sentence. Another.  
<a href="#1">[1]</a>  
Sentence with a <a  
href="Link">link</a>.  
</p>  
  
This is the template  
output.
```

Such an in-line editing interface as this one can be constructed without a formal grammar. The only thing that we need, is a correspondence between certain elements in the final HTML page, and the original wikitext. This means that we only need to know which pieces of wikitext belong to a sentence, a paragraph, a reference, or a template.

```
A sentence. Another.  
<ref>Reference</ref>  
Sentence with a  
[[link]].  
  
{{this is a template}}
```

(text positions are incorrect, too lazy to translate that from Dutch!)

For this, we use a trick. Before parsing the wikitext to HTML, we detect elements in the wikitext. For that we don't need a formal grammar, as we only need to know how these elements look like in general. We only need to know where, say, a reference, starts and ends, not what the exact syntax of such a reference is.

This detection gives us a list with start and end points in the wikitext, which we call a list of markings. We assign a unique identifier to each marking. Then we add some text of our own to the wikitext, at the exact positions of the markings. This added text contains the identifier of the marking, and is chosen in such a way that it survives the parsing to HTML. In this example XML tags are used, but in reality alphanumeric strings are often the best choice.

This creates the correspondence between the final output and the original wikitext, as we can look up markings in the output by finding them in the list. This way, we know the exact pieces of wikitext that belong to a certain piece of the output. For example, when editing the template in this example, we can look up 'template0' in the list of markings, and find that in the original wikitext it is located at position 67 to 90. We can now show this to the user, and replace the original wikitext once the user edited it. Then we parse the page again, and show the edited page to the user.

This means that every time the user makes a change, the page has to be reparsed at the server. By leveraging the existing parser, we avoid the problem of having to create a new parser ourselves!

```
A sentence. Another.  
<ref>Reference</ref>  
Sentence with a  
[[link]].
```

```
{{this is a template}}
```

```
sentence0      : 0-8  
sentence1     : 9-21  
reference0    : 22-43  
sentence2    : 44-65  
paragraph0   : 0-65  
template0    : 67-90
```

“markings”

(text positions are incorrect, too lazy to translate that from Dutch!)

For this, we use a trick. Before parsing the wikitext to HTML, we detect elements in the wikitext. For that we don't need a formal grammar, as we only need to know how these elements look like in general. We only need to know where, say, a reference, starts and ends, not what the exact syntax of such a reference is.

This detection gives us a list with start and end points in the wikitext, which we call a list of markings. We assign a unique identifier to each marking. Then we add some text of our own to the wikitext, at the exact positions of the markings. This added text contains the identifier of the marking, and is chosen in such a way that it survives the parsing to HTML. In this example XML tags are used, but in reality alphanumeric strings are often the best choice.

This creates the correspondence between the final output and the original wikitext, as we can look up markings in the output by finding them in the list. This way, we know the exact pieces of wikitext that belong to a certain piece of the output. For example, when editing the template in this example, we can look up 'template0' in the list of markings, and find that in the original wikitext it is located at position 67 to 90. We can now show this to the user, and replace the original wikitext once the user edited it. Then we parse the page again, and show the edited page to the user.

This means that every time the user makes a change, the page has to be reparsed at the server. By leveraging the existing parser, we avoid the problem of having to create a new parser ourselves!

```
<paragraph0><sentence0>A sentence.</sentence0>
<sentence1>Another.</sentence1>
<reference0><ref>Reference</ref></reference0>
<sentence2>Sentence with a [[link]].
</sentence2></paragraph0>
```

```
<template0>{{this is a template}}</template0>
```

```
sentence0      : 0-8
sentence1      : 9-21
reference0      : 22-43      "markings"
sentence2      : 44-65
paragraph0     : 0-65
template0      : 67-90
```

(text positions are incorrect, too lazy to translate that from Dutch!)

For this, we use a trick. Before parsing the wikitext to HTML, we detect elements in the wikitext. For that we don't need a formal grammar, as we only need to know how these elements look like in general. We only need to know where, say, a reference, starts and ends, not what the exact syntax of such a reference is.

This detection gives us a list with start and end points in the wikitext, which we call a list of markings. We assign a unique identifier to each marking. Then we add some text of our own to the wikitext, at the exact positions of the markings. This added text contains the identifier of the marking, and is chosen in such a way that it survives the parsing to HTML. In this example XML tags are used, but in reality alphanumeric strings are often the best choice.

This creates the correspondence between the final output and the original wikitext, as we can look up markings in the output by finding them in the list. This way, we know the exact pieces of wikitext that belong to a certain piece of the output. For example, when editing the template in this example, we can look up 'template0' in the list of markings, and find that in the original wikitext it is located at position 67 to 90. We can now show this to the user, and replace the original wikitext once the user edited it. Then we parse the page again, and show the edited page to the user.

This means that every time the user makes a change, the page has to be reparsed at the server. By leveraging the existing parser, we avoid the problem of having to create a new parser ourselves!

```
<p><paragraph0><sentence0>A sentence.  
</sentence0> <sentence1>Another.</sentence1>  
<reference0><a href="#1">[1]</a></reference0>  
<sentence2>Sentence with a <a  
href="Link">link</a>.</sentence2>  
</paragraph0></p>
```

```
<template0>This is the template output.  
</template0>
```

```
sentence0      : 0-8  
sentence1      : 9-21  
reference0     : 22-43    "markings"  
sentence2     : 44-65  
paragraph0    : 0-65  
template0     : 67-90
```

(text positions are incorrect, too lazy to translate that from Dutch!)

For this, we use a trick. Before parsing the wikitext to HTML, we detect elements in the wikitext. For that we don't need a formal grammar, as we only need to know how these elements look like in general. We only need to know where, say, a reference, starts and ends, not what the exact syntax of such a reference is.

This detection gives us a list with start and end points in the wikitext, which we call a list of markings. We assign a unique identifier to each marking. Then we add some text of our own to the wikitext, at the exact positions of the markings. This added text contains the identifier of the marking, and is chosen in such a way that it survives the parsing to HTML. In this example XML tags are used, but in reality alphanumeric strings are often the best choice.

This creates the correspondence between the final output and the original wikitext, as we can look up markings in the output by finding them in the list. This way, we know the exact pieces of wikitext that belong to a certain piece of the output. For example, when editing the template in this example, we can look up 'template0' in the list of markings, and find that in the original wikitext it is located at position 67 to 90. We can now show this to the user, and replace the original wikitext once the user edited it. Then we parse the page again, and show the edited page to the user.

This means that every time the user makes a change, the page has to be reparsed at the server. By leveraging the existing parser, we avoid the problem of having to create a new parser ourselves!

In my thesis we also look at optimizing this approach, for example by parsing only a part of the page when the user makes an edit, instead of having to reparse the entire page again. We also look at some problems with this algorithm, such as dependencies in the page and misnesting of markings. However, this is outside the scope of this presentation.

Problem:

new users who click on references,
still see complex wikitext

Solution:

teaching users wikitext by making
different elements editable

How can we present this interface to the user in a convenient way? One problem that arises, is that when a new user is editing a page, he or she can click a reference, which usually contains a lot of complex syntax. Because of that, I created a learning method, where the user starts with editing only sentences, and can then progress to lists, references, etc. By doing this, the user will learn more and more complex wikitext. Also, specific instructions can be given for each type of element. I call this interface the “functional editor” because of the functional differentiation in edit modes.

Edit mode:	<input checked="" type="radio"/> Text	<input type="radio"/> Lists	<input type="radio"/> References	<input type="radio"/> Media	<input type="radio"/> Templates	<input type="radio"/> Full editor	#0	Undo	Redo
-------------------	---------------------------------------	-----------------------------	----------------------------------	-----------------------------	---------------------------------	-----------------------------------	----	------	------

Edit sentences by clicking on them. You can use wiki syntax to format the text. Some examples:

Code	Output
Here's a link to the [[Main Page]].	Here is a link to the Main Page .
This is <i>'italic text'</i> and this is 'bold text' .	This is <i>italic text</i> and this is bold text .
[http://meta.wikimedia.org/wiki/Help:Editing More information]	More information 

“functional editor”

How can we present this interface to the user in a convenient way? One problem that arises, is that when a new user is editing a page, he or she can click a reference, which usually contains a lot of complex syntax. Because of that, I created a learning method, where the user starts with editing only sentences, and can then progress to lists, references, etc. By doing this, the user will learn more and more complex wikitext. Also, specific instructions can be given for each type of element. I call this interface the “functional editor” because of the functional differentiation in edit modes.

Problem:

users never learn editing *more* wikitext at once

Solution:

different teaching method, by progressing to larger blocks

One problem of the functional editor is that users can never edit larger pieces of wikitext. Adding more edit modes would be a mess, so I created a different learning method. With this, the user progresses from sentences to paragraphs, and then to sections or even the entire page. This way, the user learns to be more comfortable with larger amounts of wikitext. A problem, though, is that it is harder to give specific instructions. Because of the possibility to edit entire blocks of wikitext, I call this the “block editor”.

Edit mode:	<input type="radio"/> Text	<input type="radio"/> Paragraphs	<input type="radio"/> Sections	<input checked="" type="radio"/> Full editor	#0	Undo	Redo
-------------------	----------------------------	----------------------------------	--------------------------------	--	----	------	------

If you want full control over the page, you can use the full editor. With this editor you can edit the [Wiki markup](#) directly. The editor is also more complex to use, but to have full control of editing this wikis, may be worth it!

[Use the full editor »](#)

“block editor”

One problem of the functional editor is that users can never edit larger pieces of wikitext. Adding more edit modes would be a mess, so I created a different learning method. With this, the user progresses from sentences to paragraphs, and then to sections or even the entire page. This way, the user learns to be more comfortable with larger amounts of wikitext. A problem, though, is that it is harder to give specific instructions. Because of the possibility to edit entire blocks of wikitext, I call this the “block editor”.

Usability experiment with the functional editor,
the block editor, and the traditional editing interface

To evaluate whether or not in-line editing is better than one huge text box with wikitext, I did a usability experiment. 10 students of Information Sciences took part in this experiment, divided in 3 groups: one with the functional editor, one with the block editor, and one with the large text box as seen on Wikipedia.

Tasks with Wikipedia page “Groningen”

- Fixing a typo
- Adding a sentence
- Adding an external link
- Adding a reference
- Population count + 1

The students had 15 minutes to complete a few tasks on the Wikipedia page of the city of “Groningen”. First they had to fix a typo, then they had to add a sentence, and then they had to add an external link in the section at the bottom. As you can see, the tasks were of increasing difficulty. They had to add a reference, which is in the text, but shows at the bottom of the page, and finally they had to increase the population count of the city by one, both in the text and in an info box.

```
V C  ▸ Gevorderde functies ▸ Speciale tekens ▸ Hulp
| lat_sec      = 0
| lon_deg      = 6
| lon_min      = 34
| lon_sec      = 0
| foto         = [[Bestand:Martini Toren.JPG|250px|De Martinitoren]]
| ondertitel   = De 97 meter hoge [[Martinitoren]] (''d'Olle Grieze'') aan de Grote Markt
}}

'''Groningen''' is de [[hoofdstad]] van de [[Nederland]]se [[Groningen (provincie)|provincie
Groningen]] en de grootste kern in de [[Groningen (gemeente)|gelijknamige gemeente]]. De plaats
wordt in grote delen van Noord-Nederland ook kortweg aangeduid met ''Stad''. De gemeente Groningen
had per 1 januari 2010 187.622 inwoners. Hoewel de stad zelf een lager inwoneraantal heeft,
behoort Groningen tot de tien [[Lijst van grote Nederlandse steden|grootste steden in Nederland]].

Groningen behoort tot de [[Lijst van Nederlandse plaatsen met stadsrechten|oudste steden]] van
[[Nederland]]. Door zijn relatief geïsoleerde ligging, ten opzichte van de opeenvolgende feitelijk
machtscentra (Utrecht, Brussel en Den Haag), was de stad historisch gezien vooral op zich zelf en
de directe omgeving aangewezen. Als [[Hanze]]stad maakte Groningen deel uit van het Noord-Duitse
handelsnetwerk. Later werd de stad vooral een regionaal marktcentrum. De stad ontwikkelde zich en
was de facto een [[stadstaat]].<ref>http://www.rug.nl/kennisdebat/onderwerpen/regio/verleden/geschiedenisgroningen Interview Maarten Duijvendak, hoogleraar regionale geschiedenis aan de RUG
en hoofdredacteur ''Geschiedenis van Groningen''</ref><ref>Jan van de Broek, in het boek
''Groningen, een stad apart. Over het verleden van een eigenzinnige stad (1000-1600)'', Assen:
Koninklijke Van Gorcum B.V. 2007, p.71:</ref><ref>http://www.varenius.nl/index.php?id=7&tx\_windcatalog\_pi1&#91;uid&#93;=14 Meindert Schroor, Stadstaat Groningen, Regio-Projekt 1999</ref>
```

The experiment was a so-called “Think aloud, talk aloud” experiment, which means that the students had to explicitly say what they were doing, and what they were thinking about. This has been recorded, together with the actions on the screen, and is available (in Dutch) on Wikimedia Commons, an online media repository.

Here you can see a few screenshots. The first shows the original editor, the control group. Then we see the functional editor, with the “Lists” edit mode selected. And finally we see the block editor with the “Sections” edit mode selected.

Ijsbaan omvat het complex ook een [ijs hockeyvloer](#), [tennisvelden](#) en [fitnessruimtes](#). Het omvat ook nog een sportterrein met een 10-tal sportvelden. Zwemmen kan in Groningen in meerdere overdekte [baden](#). Het enige openluchtbad, [De Papiermolen](#), is sinds 2007 een [rijksmonument](#).

Zie ook

- [Gemeente Groningen](#)
- [Groningen van A tot Z](#)
- [Lijst van rijksmonumenten in de stad Groningen](#)
- [Lijst van stadspoorten van Groningen](#)
- [Straatnamen in de stad Groningen](#)
- [Stadsmarkeringen](#)

Externe links

- [Officiële website van de gemeente Groningen](#)
- [Historische foto's van Groningen](#)
- [Rijks Universiteit Groningen](#)

```

== Referenties ==
{{references}}

[[Categorie:Groningen (stad)|Groningen (stad)]]
[[Categorie:Plaats in Groningen]]
[[Categorie:Hoofdstad van een Nederlandse provincie]]

```

Bekijken **Annuleren**

Categorieën: [Groningen \(stad\)](#) | [Plaats in Groningen](#) | [Hoofdstad van een Nederlandse provincie](#)

Referenties

- ↑ Interview Maarten Duijvendak, hoogleraar regionale geschiedenis aan de RUG en hoofdredacteur *Geschiedenis van Groningen*
- ↑ Jan van de Broek, in het boek *Groningen, een stad apart. Over het verleden van een eigenzinnige stad (1000-1600)*, Assen: Koninklijke Van Gorcum B.V. 2007, p.71:
- ↑ Meindert Schroor, *Stadstaat Groningen*, Regio-Projekt 1999
- ↑ *De geschiedenis van Groningen - deel I*, hoofdstuk IV, R.W.M. Van Schaik
- ↑ CBS
- ↑ architectenweb.nl John Hejduk
- ↑ http://www.ringzuidgroningen.nl/

Categorieën: [Groningen \(stad\)](#) | [Plaats in Groningen](#) | [Hoofdstad van een Nederlandse provincie](#)

The experiment was a so-called “Think aloud, talk aloud” experiment, which means that the students had to explicitly say what they were doing, and what they were thinking about. This has been recorded, together with the actions on the screen, and is available (in Dutch) on Wikimedia Commons, an online media repository.

Here you can see a few screenshots. The first shows the original editor, the control group. Then we see the functional editor, with the “Lists” edit mode selected. And finally we see the block editor with the “Sections” edit mode selected.

Conclusions

- Edit modes don't work!
- References are hard!
- In-line editing seems to be better, but the difference is not really significant

After analysis of the videos, I came to a few conclusions. The first, most interesting conclusion, is that edit modes do not work at all. Many users didn't even notice them, and if they did they used them incorrectly.

The second conclusion is that references are quite hard to understand, even for these well-educated students. Many had a hard time figuring out how they worked, and ended up trying to add them at the bottom of the page, rather than inside the text.

Finally, it seemed that in-line editing was an improvement, but it wasn't quite a significant difference. This may be due to the fact that these students were used to editing source code, so in-line editing wouldn't necessarily be better for them. Therefore, it's hard to make a clear statement on this.

Groningen (stad)

Groningen is de hoofdstad van de Nederlandse provincie Groningen, het bisdom Groningen, de gemeente Groningen en de studentenhoofdstad van het noorden. De plaats wordt in grote delen van Noord-Nederland ook kortweg aangeduid met *Stad*. De gemeente Groningen had per 1 januari 2010 187.623 inwoners. Hoewel de stad zelf een lager inwoneraantal heeft, behoort Groningen tot de tien grootste steden in Nederland. Groningen is een zeer fietsvriendelijke stad!

Groningen behoort tot de oudste steden van Nederland. Door zijn relatief geïsoleerde ligging, ten opzichte van de opeenvolgende feitelijk machtscentra (Utrecht, Brussel en Den Haag), was de stad historisch gezien vooral op zich zelf en de directe omgeving aangewezen. Als Hanzestad maakte Groningen deel uit van het Noord-Duitse handelsnetwerk. Later werd de stad vooral een regionaal marktcentrum. De stad ontwikkelde zich en was de facto een stadstaat.^{[1][2][3][4]} Sinds de Republiek hoorde Groningen nominaal bij Nederland, maar tot aan de Franse tijd bleef Groningen feitelijk een autonome stad, die heer was in het grootste deel van de provincie.

Tegenwoordig is Groningen een stad met veel gevarieerde handel en industrie. Groningen is daarnaast vooral een studentenstad met bijna 50.000 studenten, van wie ruim 30.000 in de stad wonen.^[5] De stad werkt samen met de omliggende gemeenten in de Regiovisie Groningen-Assen^[6].

Bezienswaardigheden

Groningen heeft het zwaar te verduren gehad tijdens de Tweede Wereldoorlog. Met name de Grote Markt werd tijdens de bevrijding in april 1945 zwaar beschadigd. Het symbool van de stad, de Martinitoren (door de "stadjers" *d'Olle Grieze* genoemd) en de bijbehorende kerk bleven gespaard, evenals het Goudkantoor en het stadhuis.

Ten noordoosten van de Grote Markt ligt het Martiniekerkhof met het provinciehuis en het Prinsenhof. Achter dat pand ligt de fraaie Prinsentuin met hierin een van de rijkst vormgegeven zonnewijzers van het land.

Aan de westkant van de Grote Markt ligt de Vismarkt. Met name de zuidkant van dit plein bevat een aantal fraaie panden. Het plein wordt aan de westkant begrensd door de Korenbeurs, een verwijzing naar de rijkdom van de provincie in de 19e eeuw die gebaseerd was op de graanteelt op de vruchtbare zeelei.

Achter de Korenbeurs verheft zich de Der Aa-kerk, na de Martinikerk de tweede parochiekerk van de historische stad. Iets verder naar het westen, langs het water, of in het Gronings 'het diep', staat langs de Hoge der Aa een aantal fraaie pakhuizen en woonhuizen. Ook langs de Noorderhaven staan dergelijke panden.

Vanaf de Grote Markt naar het noorden loopt de Oude Boteringestraat met een van de oudste huizen van de stad, het Calmershuis en de oude Rechtbank. Over de diepenring loopt de straat door als Nieuwe Boteringestraat en loopt dan langs de Nieuwe Kerk, uit 1660 en de eerste kerk in Groningen die gebouwd is voor de Protestantse eredienst.

Een bijzonder aspect van Groningen zijn de vele hofjes, in Groningen gasthuizen genoemd. De bekendste zijn het Pelstergasthuis en het Pepergasthuis. Het hoofdstation uit 1896, direct buiten de zogenaamde diepenring, staat bekend als een van de fraaiste stations in Nederland.

Aan de zuid-west kant staat het markante Gasuniegebouw, hoofdkantoor van de Gasunie. Het is gebouwd in de stijl van de organische architectuur. Andere interessante moderne architectuur is te vinden in de wijk Hoomse Meer. Hier is het Wall House #2 te vinden, het enige gerealiseerde ontwerp uit een serie die wel als het meesterstuk van de Amerikaanse architect John Hejduk wordt beschouwd.^[7]

Kunst en cultuur

Groningen is de grootste, Groningers zouden zeggen enige, stad in het noorden van Nederland. De stad vervult daardoor een centrumfunctie, die nog versterkt wordt door de aanwezigheid van de universiteit en het grote aantal jonge inwoners die dat meebrengt. Dat uit zich in een zeer gevarieerd cultureel aanbod.

Musea

Het Groninger Museum is zonder meer het belangrijkste museum van de stad. Oorspronkelijk was dit een vrij traditioneel provinciaal streekmuseum maar het heeft zich ontwikkeld tot één van de meest vernieuwende musea in Nederland, met name sinds de nieuwbouw op een kunstmatig eiland in het Verbindingskanaal.

De stad heeft nog enkele andere musea. Zo is er het Universiteitsmuseum, waar tentoonstellingen over zowel de geschiedenis van de universiteit als de nieuwste ontwikkelingen in het onderzoek te zien zijn. De geschiedenis van Groningen, stad en provincie, op zeevaartgebied is het thema van het Noordelijk Scheepvaartmuseum in de Brugstraat. Het Grafisch Museum is gewijd aan de grafische kunst en druktechnieken.

Kunst in de openbare ruimte

Er zijn in Groningen veel beeldhouwwerken (circa 400) in de openbare ruimte te zien waaronder enkele van Wladimir de Vries. Ook het beeldenproject van de Rijksuniversiteit Groningen heeft meerdere kunstwerken opgeleverd. Verder bevindt zich bij iedere uitgaande weg van de gemeente een beeld of installatie van het zogenaamde "project Stadsmarkeringen" uit 1990.

Groningen

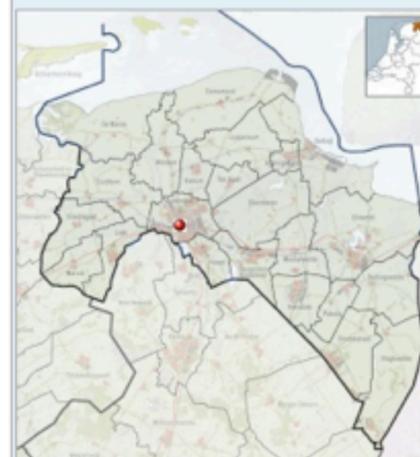
Stad in Nederland 



(Details)



(Details)



Situering

Provincie 22px Groningen

Gemeente 22px Groningen

Coördinaten 53°13′N 6°34′E 

Algemeen

Inwoners (1 april 2011) 190.334 (gemeente)

Overig

Netnummer 050

Belangrijke verkeersaders A7 A38 N46

Website www.groningen.nl 

Foto's



I found the first conclusion most interested, as apparently it didn't work to have users learn wikitext this way. Therefore, I decided to drop the idea of edit modes entirely. Instead, I chose to have users edit all different elements at once. With sentences, references, and templates, this is no problem, as they usually don't overlap. Paragraphs and sections are harder though, and finally I came to this solution: bars on the left side of the screen, that make editing of larger blocks of wikitext possible. Because this interface is somewhat of a hybrid between the functional editor and the block editor, I call this the "hybrid editor".

GRNET

Usability experiment with the hybrid editor

When working on the hybrid editor, there was already some interest in this project, most notably from the Wikimedia Foundation. But at a certain point, I was contacted by GRNET, a Greek research institute, who were interested in devoting some time to improve in-line editing. They did a second usability experiment, this time with the hybrid editor. The experiment was about the same as my first experiment, only they didn't use a control group, unfortunately. However, they did have some more inexperienced users, which was an improvement.

The videos have not been made public, unfortunately, but I can show you some fragments. Transcripts of the videos are public, though. While this video plays, I can tell you the conclusions.

First, the hybrid editor seemed to work a lot better than the other two interfaces with the edit modes. Still, many times it took users a while to figure out that they could edit entire paragraphs or sections using the bars at the left. When they figured this out, or when the researches gave a small hint, they usually used it correctly.

Many problems are still inherent to wikitext, such as the problems with references we saw earlier. There isn't much we can do about that, although it would be possible to give specific instructions when clicking certain elements.

Sometimes users tried to use the editor as a visual editor, for example by trying to select text. This is inherent to this compromise: a real visual editor is probably better, but right now this is a more feasible alternative for Wikipedia.

The video shows how many things go wrong, but the truth is that most tasks went alright. Therefore, I think in-line editing is a real improvement for editing pages on Wikipedia, and has some real potential to be extended to make the editing experience even better. It's one of the few alternatives for improving the usability of editing on the short term, and it's now up to the Wikimedia Foundation and the community to decide whether or not to implement it.

Thank you.