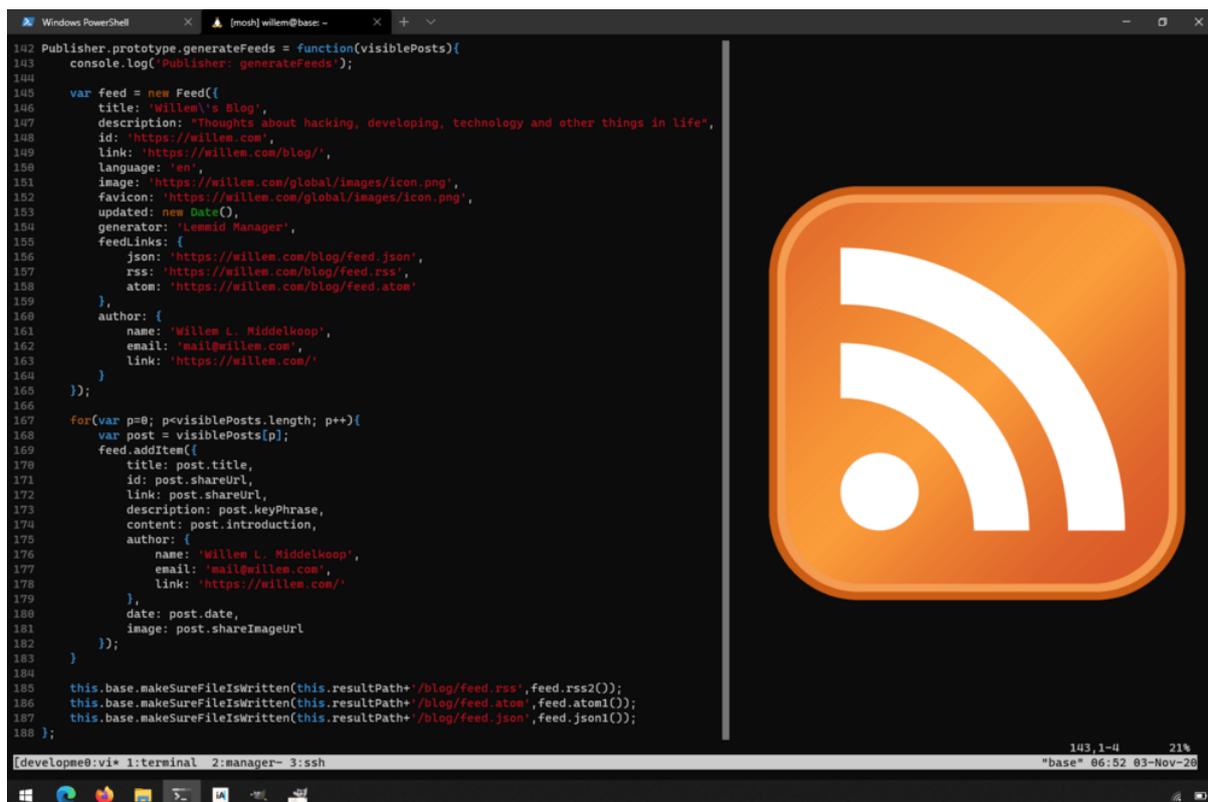


Steigern Sie den Traffic Ihres Blogs mit einem RSS-Feed

Einrichten von RSS/Atom mit NodeJS

Willem L. Middelkoop

Oct. 31, 2020



```
142 Publisher.prototype.generateFeeds = function(visiblePosts){
143   console.log('Publisher: generateFeeds');
144
145   var feed = new Feed({
146     title: 'Willem\'s Blog',
147     description: 'Thoughts about hacking, developing, technology and other things in life',
148     id: 'https://willem.com',
149     link: 'https://willem.com/blog/',
150     language: 'en',
151     image: 'https://willem.com/global/images/icon.png',
152     favicon: 'https://willem.com/global/images/icon.png',
153     updated: new Date(),
154     generator: 'Lemmid Manager',
155     feedlinks: {
156       json: 'https://willem.com/blog/feed.json',
157       rss: 'https://willem.com/blog/feed.rss',
158       atom: 'https://willem.com/blog/feed.atom'
159     },
160     author: {
161       name: 'Willem L. Middelkoop',
162       email: 'mail@willem.com',
163       link: 'https://willem.com/'
164     }
165   });
166
167   for(var p=0; p<visiblePosts.length; p++){
168     var post = visiblePosts[p];
169     feed.addItem({
170       title: post.title,
171       id: post.shareUrl,
172       link: post.shareUrl,
173       description: post.keyPhrase,
174       content: post.introduction,
175       author: {
176         name: 'Willem L. Middelkoop',
177         email: 'mail@willem.com',
178         link: 'https://willem.com/'
179       },
180       date: post.date,
181       image: post.shareImageUrl
182     });
183   }
184
185   this.base.makeSureFileIsWritten(this.resultPath+ '/blog/feed.rss', feed.rss2());
186   this.base.makeSureFileIsWritten(this.resultPath+ '/blog/feed.atom', feed.atom1());
187   this.base.makeSureFileIsWritten(this.resultPath+ '/blog/feed.json', feed.json1());
188 };
```



Das Schreiben für meinen Blog hat mir viel Spaß gemacht, ich erhalte Nachrichten aus aller Welt über die Dinge, die ich mag. Der Traffic wächst, jeden Monat finden mehr Leser zu meinem Blog. Diese Woche erhielt ich eine Anfrage, RSS-Feeds auf meinem Blog zu unterstützen. Ich habe mich gefragt, sind RSS-Feeds heute noch relevant?

RSS-Feeds

Mit RSS (oder "Really Simple Syndication") können Benutzer und Anwendungen auf Aktualisierungen von Websites in einem standardisierten und computerlesbaren Format zugreifen. Dies ermöglicht es ihnen, Aktualisierungen von vielen verschiedenen Websites in einem einzigen News-Aggregator zu kombinieren, im Grunde wie eine personalisierte "Zeitung".

Bevor soziale Medien sehr populär wurden, wurden RSS-Feeds häufig verwendet, um Links und Inhalte zwischen verschiedenen Nachrichten- und Blogging-Quellen auszutauschen. Das Abonnieren des Feeds einer Website ist ähnlich wie das "Folgen einer Sache" in sozialen Medien, Sie erhalten automatisch Updates.

Das Tolle an RSS ist, dass es dezentral ist, es gibt keine einzelne Behörde (oder großes Unternehmen), die den Inhalt oder die Plattform kontrolliert.

Rückgang der RSS-Nutzung

Da soziale Medien die RSS-Feeds an Popularität übertroffen haben, ist die Unterstützung für RSS zurückgegangen. Plattformen wie Twitter und Facebook ermöglichen es einzelnen, einfachen Nutzern, Inhalte zu teilen. Dies ist viel einfacher, als einen Blog mit einem entsprechenden RSS-Feed einzurichten. Der Großteil der Social-Media-Inhalte wird dann moderiert und durch Algorithmen gefiltert, gepaart mit Werbung, um zu einer geldgenerierenden Massenmedienplattform zu werden.

Wiederauferstehung des E-Mail-Newsletters

Als Google Reader seine Pforten schloss, [some bloggers lost 6.500 RSS subscribers](#) über Nacht! Wenn Sie darüber nachdenken, den Kontakt zu Ihrem Publikum aufrechtzuerhalten, macht E-Mail Sinn. Für die meisten von uns ist E-Mail mit unseren täglichen Routinen verwoben. Wir überprüfen sie regelmäßig und haben von unseren mobilen Geräten aus Zugriff auf unsere E-Mails. Richtig gemacht, ist das Versenden von E-Mail-Newslettern eine praktikable Möglichkeit, mit Ihrem Publikum in Kontakt zu bleiben. Es ist keine Überraschung, dass [people describe this to be a news media renaissance](#).

Warum immer noch RSS-Feeds verwenden?

Im Zeitalter von Social Media und Newsletter-Wiederauferstehungen mag man sich fragen, warum man immer noch RSS-Feeds unterstützen sollte? Ich denke, der Hauptgrund ist, dass es sehr wenig Aufwand erfordert, es richtig zu machen, und es (immer noch) einen Mehrwert bietet.

Menschen, aber vielleicht noch wichtiger, Computer können RSS-Feeds verwenden, um automatisch Updates von Ihrem Blog abzurufen. Beliebte [newsletter software can use RSS](#), um den Prozess der Newsletter-Erstellung zu automatisieren. Indem Sie RSS auf Ihrem Blog unterstützen, machen Sie es anderen leichter, Newsletter *für Sie* zu versenden.

Wussten Sie, dass andere Leute RSS-Feeds verwenden, um automatisch über *Ihre* Inhalte zu twittern, indem sie moderne Dienste wie IFTTT (if this then that) verwenden? Sie sollten sich [diesen article](#) ansehen. Kurz gesagt, RSS ist eine großartige Möglichkeit, die Automatisierung von Inhalten zu unterstützen. Selbst wenn *Sie* dies nicht beabsichtigen, sollten Sie in Erwägung ziehen, andere bei ihren Bemühungen zu unterstützen, indem Sie RSS unterstützen.

Dasselbe gilt für Software wie Narrowcasting (oder Digital Signage), bei der Schlagzeilen automatisch von Ihrer Website abgerufen werden können. [RSS feeds are widely supported by digital signage software](#), um Nachrichtenticker zu generieren.

Mit modernen RSS-Reader-Apps wie [Feedly](#) können normale Benutzer viele der Probleme der ursprünglichen RSS-Reader-Apps überwinden: überwältigende Informationsflut. Im Moment bedienen sie eine Nische, da sie weniger populär sind als soziale Medien, aber

diese Nische kann eine wichtige sein (Quantität vs. Qualität). Vielleicht gibt es eines Tages eine RSS-Feed-Renaissance, genau wie bei den E-Mail-Newslettern.



Moderne RSS-Reader-Apps bieten erweiterte Funktionen zum Filtern und Organisieren verschiedener Feeds - so werden Sie zu einem ZEN-Meister der Informationsflut gewissermaßen

Hinzufügen von RSS-Unterstützung zu Ihrem Blog

Wenn Sie mit gängiger Blogging-Software wie WordPress bloggen, ist die Wahrscheinlichkeit groß, dass Sie bereits [RSS support build in](#) haben. Versuchen Sie, `"/feed"` an die Adresse Ihres Blogs anzuhängen, es könnte automatisch erscheinen. Gut für Sie! Wenn Sie eine benutzerdefinierte Blogging-Software verwenden, wie ich es tue, müssen Sie möglicherweise RSS-Feeds selbst implementieren.

Implementierung der RSS-Unterstützung selbst

Technisch gesehen ist RSS nichts anderes als ein Dialekt von XML. Es ist eine Datei, die Ihre Inhalte in einer standardisierten, strukturierten Weise enthält. Die [RSS specification is open](#) und birgt sehr wenige Geheimnisse, da sie gut dokumentiert ist.

Dieser Blog wird mit einer benutzerdefinierten Software verwaltet, die ich für meine Kunden entwickelt habe. Meine Software generiert sogenannte "statische HTML-Seiten" aus einem dynamisch verwalteten Backend, das auf NodeJS läuft. Das bedeutet, dass ich einen Weg finden musste, die RSS-Feeds aus meinem statischen Site-Generator zusammen mit den HTML-Dateien zu generieren. Nun ist es an der Zeit, technisch zu werden! :-)

Install

> npm i feed

Weekly Downloads

64.226

Version	License
4.2.1	MIT

Unpacked Size	Total Files
104 kB	52

Issues	Pull Requests
15	4

Developer Ready: Quickly generate syndication feeds for your Website.

Strongly Typed: Developed using TypeScript / type-safe.

Tested: Tests & snapshot for each syndication format to avoid regressions.

Das beliebte NodeJS-Modul "feed" macht die Verbreitung von Inhalten einfach und intuitiv

Es ist immer eine gute Idee, sich bestehende Implementierungen offener Standards anzusehen, bevor man sich daran macht, "das Rad neu zu erfinden". Die Chancen stehen gut, dass jemand die ganze harte Arbeit bereits erledigt hat. Aber darüber hinaus kann es klug sein, ein gut getestetes Paket einer selbst erstellten Implementierung (mit unentdeckten Fehlern und Mängeln) vorzuziehen. Für NodeJS gibt es das "Feed" module, das die Generierung von RSS-, Atom- und JSON-Feeds vereinfacht.

```

102 Publisher.prototype.generateFeeds = function(visiblePosts){
103   console.log('Publisher: generateFeeds');
104
105   var feed = new Feed({
106     title: 'Willem's Blog',
107     description: 'Thoughts about hacking, developing, technology and other things in life',
108     id: 'https://willem.com',
109     link: 'https://willem.com/blog/',
110     language: 'en',
111     image: 'https://willem.com/global/images/icon.png',
112     favicon: 'https://willem.com/global/images/icon.png',
113     updated: new Date(),
114     generator: 'Lemmy Manager',
115     feedlinks: {
116       json: 'https://willem.com/blog/feed.json',
117       rss: 'https://willem.com/blog/feed.rss',
118       atom: 'https://willem.com/blog/feed.atom'
119     },
120     author: {
121       name: 'Willem I. Middelkoop',
122       email: 'mail@willem.com',
123       link: 'https://willem.com/'
124     }
125   });
126
127   for(var p=0; p<visiblePosts.length; p++){
128     var post = visiblePosts[p];
129     feed.addItem({
130       title: post.title,
131       id: post.shareUrl,
132       link: post.shareUrl,
133       description: post.keyPhrase,
134       content: post.introduction,
135       author: {
136         name: 'Willem I. Middelkoop',
137         email: 'mail@willem.com',
138         link: 'https://willem.com/'
139       },
140       date: post.date,
141       image: post.shareImageUrl
142     });
143   }
144
145   this.base.makeSureFilesWritten(this.resultPath+'/blog/feed.rss', feed.rss2());
146   this.base.makeSureFilesWritten(this.resultPath+'/blog/feed.atom', feed.atom1());
147   this.base.makeSureFilesWritten(this.resultPath+'/blog/feed.json', feed.json1());
148 };

```

Implementierung von RSS-, Atom- und JSON-Feeds in meinem Blog mit NodeJS und dem Feed-Modul

Mit dem Feed-Modul habe ich meinem statischen Site-Generator eine Funktion namens "generateFeeds()" hinzugefügt, die den Code implementiert, der notwendig ist, um RSS-, Atom- und JSON-Feeds zu generieren. Ihre Hauptfunktion besteht darin, die verschiedenen Felder, wie sie in der RSS-Feed-Struktur festgelegt sind, auf die meines Website-Objektmodells abzubilden. Es ist nicht schwer, es richtig zu machen, wenn man bereits ein anständiges Objektmodell mit Verweisen auf Felder wie eine Bild-URL, den Titel eines Beitrags und das Veröffentlichungsdatum hat.

```
location = /blog/feed/ {
    return 301 https://willem.com/blog/feed.rss;
}

location = /feed/ {
    return 301 https://willem.com/blog/feed.rss;
}
```

Konfiguration von NGINX zur Weiterleitung gängiger Feed-URLs zu den generierten RSS-Dateien

Ich dachte, es wäre eine gute Idee, meinen Webserver, NGINX, so einzurichten, dass er gängige Feed-URLs automatisch auf die generierten RSS-Dateien umleitet. Auf diese Weise ist meine benutzerdefinierte RSS-Implementierung für andere Menschen und Computer leichter zu entdecken, da sie das Verhalten gängiger Blogging-Software nachahmt.

Sie können das Ergebnis selbst überprüfen, indem Sie eine der folgenden URLs verwenden:

- <https://willem.com/feed/>
- <https://willem.com/blog/feed/>
- <https://willem.com/blog/feed.rss> (RSS)
- <https://willem.com/blog/feed.atom> (Atom)
- <https://willem.com/blog/feed.json> (JSON)

Fazit

Manchmal ist es sinnvoll, einen zweiten Blick auf die Technologie von gestern zu werfen. Auch wenn RSS-Feeds vielleicht nicht mehr so beliebt sind wie früher, können sie Ihrem Blog dennoch einen Mehrwert bieten.

Da die Implementierung eines RSS-Feeds einfach ist und die Pflege vollständig automatisiert werden kann, gibt es wirklich keinen Grund, warum Sie nicht in Erwägung ziehen sollten, ihn Ihrem Blog hinzuzufügen. Als Bonus sind Sie bereit für die nächste Renaissance!