

WordPress Export Feature Plugin

Overview

The main goals of this project are:

1. A proposal for a new WXR 1.3:
 - a. WXR 1.3-proposed has several advantages over WXR 1.2 that make it easier to do XML Namespace-aware and streaming import of WXR instances, as well as smaller WXR instances.
 - b. Additionally, it makes a few minor changes to the WXR markup to bring it into compliance with the RSS spec and the RSS Advisory Board's "Best Practices" profile (<http://www.rssboard.org/rss-profile>).
 - c. The details of that proposal can be found at <https://github.com/pbiron/wxr/1.3-proposed>.
2. Use a real streaming XML serializer to generate the XML. This is the opening salvo in producing a complete XML API for WP, that could be used not only for export/import but also for RSS/Atoms feeds, etc.
3. Provide hooks that allow plugins to add extension markup to the generated WXR instance, e.g., rows from custom tables that are associated with various content included in the output.
4. Provide additional hooks that allow plugins that hook into the `export_filters` action that exists in the standard exporter to actually do something useful with the "export filters" they have added.
5. Fix what I consider to be bugs in the standard exporter that are unrelated to any of the above. For example, when exporting posts of a single post type the standard exporter exports the terms associated with the posts in the `/rss/channel/item/category` element, but does not export the "full" terms. Thus, hierarchical relationship between those terms are not represented (and cannot be produced by the importer), nor are any term metas and descriptions for those terms. The same applies to terms in the `post_format` taxonomy.

Current Status

A complete plugin that implements the above goals is available at <https://github.com/pbiron/wordpress-exporter-v2>. The implementation of that plugin is very close to the standard exporter. I fully realize that the current implementation of the standard

exporter has A LOT of problems, but I felt that for the initial release of this plugin that it would be easier to compare/contrast it with the standard exporter if its implementation was as close as possible to that of the standard exporter. I fully expect to implement ideas from the Export API trac ticket at <https://core.trac.wordpress.org/ticket/22435> (although at this point I don't expect to incorporate the use of Oxymel for serializing XML instances) and to ditch the "function soup" style in future releases.

Companion/Demonstration Plugins

Several plugins have been written that demonstrate the use of the various hooks added in the Exporter V2 plugin:

1. <https://github.com/pbiron/wordpress-importer-v2>
 - a. Since the markup of WXR 1.3-proposed is significantly different from WXR 1.2, exports generated with the Exporter V2 plugin cannot be imported with the standard importer.
 - b. This plugin is a fork of <https://github.com/humanmade/WordPress-Importer> that offers several enhancements:
 - i. It can import WXR 1.3-proposed instances in a fully XML Namespace-aware manner.
 - ii. It remains backwards compatible with WXR 1.0, 1.1 and 1.2 (by seamlessly transforming them into WXR 1.3-proposed instances); but can do so in a fully XML Namespaces-aware manner (see <https://core.trac.wordpress.org/ticket/27896>).
 - iii. It can run side-by-side with the standard importer (to help test that when importing WXR 1.0, 1.1 and 1.2 instances it produces the same results as the standard importer).
 - c. The development of this importer plugin will likely proceed semi-independently from its base plugin until WXR 1.3-proposed gains acceptance, but the intention is to merge it back into the base plugin when that happens.
2. <https://github.com/pbiron/wordpress-exporter-v2-extension>
 - a. This plugin doesn't actually do anything useful, but it does demonstrate all of the new hooks.
3. <https://github.com/pbiron/p2p-export-import>
 - a. This plugin actually does something useful: it exports info from P2P custom tables and re-imports that info when used with <https://github.com/pbiron/wordpress-importer-v2>. Note that currently it is only able to import "post to post" relationships (and not, e.g., "user to term", etc relationships that P2P is able to create).
 - b. The need for exporting/importing such custom tables can be seen when, for instance, trying to export/import the DevHub Code Reference which relies so heavily on the custom tables that P2P produces.

Contributors

Paul V. Biron (pbiron on wordpress.org and slack). Paul has almost 20 years experience as a recognized XML expert. He was one of the editors of the W3C's XML Schema spec, a past member of the W3C Advisory Committee, past co-chair of several Health Level Seven (<http://www.hl7.org>) XML-related working groups and editor of many of HL7's XML-related standards.

Ryan McCue (rmccue on wordpress.org and slack). Ryan is the lead on the WordPress Importer feature plugin and REST API.

What is Needed

1. Feedback on WXR 1.3-proposed (the markup changes as well as the XML schema that describes that markup). This is the most urgent need, as it impacts both this and the importer redux plugin.
2. Feedback on the goals, design, and implementation of an XML API for WP. Some initial ideas for an XML API can be found at <https://github.com/pbiron/wordpress-exporter-v2#to-dos>.
3. Feedback on the extra hooks available in this exporter or other "functional" changes to the exporter.
4. Creation of a slack channel (e.g., #feature-exporter), for on-going discussions of this project.

Timeline

I fully realize that this is not going to be picked up for WP 4.9. Thus I am targeting 1-2 releases after that. I wouldn't be surprised if it takes 12-18 months before it is fully merged into core.