

CUNY Academic Commons - Feature #12598

Investigate and remediate database tables without a primary key

2020-03-31 04:31 PM - Boone Gorges

Status:	Resolved	Start date:	2020-03-31
Priority name:	Normal	Due date:	
Assignee:	Boone Gorges	% Done:	0%
Category name:	WordPress Plugins	Estimated time:	0.00 hour
Target version:	1.17.0		
Description			
<p>Lihua has recently asked our team to investigate why a number of our WordPress tables have no primary key. The context of the request is that he'd like to explore a cluster configuration for database servers, giving us greater redundancy and mitigating against downtime issues. (See eg #12436.)</p> <p>I'm opening this ticket to explore what needs to be done to ensure that all Commons tables have primary keys.</p> <p>The first step will be to analyze a list of tables without primary keys. While there are hundreds of tables without primary keys, the vast majority are generated by a small handful of plugins. See attached cac-database-tables-without-primary-keys.txt. By cataloging the distinct cases, we can then begin analyzing each individual case. The rough steps for each case will be as follows:</p> <ol style="list-style-type: none">1. Determine how the table is created. In general, this will mean identifying the responsible plugin (I believe it's always a plugin, never WP itself), and the finding the specific place in the plugin where the table schema is defined.2. Assess whether adding a primary key would in any way break functionality. As a rule, the answer should be no: we should always be able to add an `AUTO_INCREMENT` column, which would be ignored by the plugin code, but would satisfy the PXC requirements.3. Assuming we can add the primary key, write and execute a script that will `ALTER TABLE` for each relevant table in question in our database.4. Determine whether any application-level changes are necessary to ensure that tables created in the future will have the necessary key. In some cases, the responsible plugins have been deprecated - see eg firestats - so the point is moot. In other cases, the plugin is abandoned, so we can probably simply modify the table schema as defined in the plugin code. In other cases, we may need a plugin-level `ALTER TABLE` statement. <p>I've included Lihua's original request below, for reference:</p> <p>I am exploring feasible paths to migrate production databases to a PXC cluster.</p> <p>One major hurdle is that PXC is best run with strict mode set to ENFORCING or MASTER:</p> <p>https://www.percona.com/doc/percona-xtradb-cluster/LATEST/features/pxc-strict-mode.html</p> <p>And the ENFORCING or MASTER node will try to validate a whole bunch of items:</p> <p>https://www.percona.com/doc/percona-xtradb-cluster/LATEST/features/pxc-strict-mode.html#validations</p> <p>My question for you:</p> <p>Why so many wordpress database tables are created (or designed) without a primary key? Do you think there is a feasible path to set or create surrogate keys for wordpress tables in order to migrate to a mysql cluster?</p> <p>https://stackoverflow.com/questions/2515596/can-a-database-table-be-without-a-primary-key</p>			
Related issues:			
Related to CUNY Academic Commons - Feature #12800: Automated scan for databas...		Resolved	2020-05-13

History

#1 - 2020-03-31 04:44 PM - Boone Gorges

I've adjusted the raw list of tables to remove prefixes, and then to eliminate duplicates. Here are the culprits:

[_3wp_broadcast_broadcastdata](https://redmine.gc.cuny.edu/issues/12598#note-2)
[bp_activity_sitewide_test](https://redmine.gc.cuny.edu/issues/12598#note-3)
[cas_count](https://redmine.gc.cuny.edu/issues/12598#note-4)

eas_image ibid
 ep_ad_pop_daily <https://redmine.gc.cuny.edu/issues/12598#note-5>
 ep_ad_pop_total ibid
 ep_project_users <https://redmine.gc.cuny.edu/issues/12598#note-6>
 els_cus_chests <https://redmine.gc.cuny.edu/issues/12598#note-24>
 els_sus_signups
 els_sus_tasks
 ebd_link <https://redmine.gc.cuny.edu/issues/12598#note-7>
 ebd_posted_data ibid
 email_user <https://redmine.gc.cuny.edu/issues/12598#note-8>
 etcomment_rating <https://redmine.gc.cuny.edu/issues/12598#note-9>
 etright_answer ibid
 firestats_archive_countries - <https://redmine.gc.cuny.edu/issues/12598#note-10>
 firestats_archive_pages ibid
 firestats_archive_referrers ibid
 firestats_archive_sites ibid
 firestats_archive_useragents ibid
 firestats_options ibid
 firestats_rss_subscribers ibid
 firestats_url_metadata ibid
 firestats_user_sites ibid
 flickr_post <https://redmine.gc.cuny.edu/issues/12598#note-11>
 news_announcement <https://redmine.gc.cuny.edu/issues/12598#note-12>
 nf3_action_meta <https://redmine.gc.cuny.edu/issues/12598#note-25>
 nf3_actions ibid
 nf3_chunks ibid
 nf3_field_meta ibid
 nf3_fields ibid
 nf3_form_meta ibid
 nf3_forms ibid
 nf3_object_meta ibid
 nf3_objects ibid
 nf3_relationships ibid
 podpress_stats <https://redmine.gc.cuny.edu/issues/12598#note-13>
 revslider_css <https://redmine.gc.cuny.edu/issues/12598#note-14>
 revslider_layer_animations ibid
 revslider_navigations ibid
 revslider_navigations_bkp ibid
 revslider_sliders ibid
 revslider_slides ibid
 revslider_static_slides ibid
 revslider_static_slides_bkp ibid
 silas_flickr_cache <https://redmine.gc.cuny.edu/issues/12598#note-15>
 smush_dir_images <https://redmine.gc.cuny.edu/issues/12598#note-26>
 statpress <https://redmine.gc.cuny.edu/issues/12598#note-16>
 weu_subscribers <https://redmine.gc.cuny.edu/issues/12598#note-8>
 weu_unsubscriber ibid
 weu_user_notification ibid
 wp_cloudplugins_log
 woo_custom_nav_menus <https://redmine.gc.cuny.edu/issues/12598#note-17>
 woo_custom_nav_records ibid
 woo_tables_meta <https://redmine.gc.cuny.edu/issues/12598#note-18>
 wpmlfieldslists <https://redmine.gc.cuny.edu/issues/12598#note-19>
 yeastseo_meta <https://redmine.gc.cuny.edu/issues/12598#note-20>
 zotpress <https://redmine.gc.cuny.edu/issues/12598#note-20>
 zotpress_cache
 zotpress_images
 zotpress_oauth
 zotpress_zoteroCollections

I'll use this as a master list, striking items as I've reviewed them. I'll probably do those belonging to a single plugin as a group.

#2 - 2020-03-31 04:59 PM - Boone Gorges

_3wp_broadcast_broadcastdata

This table is generated by threewp-broadcast:

https://github.com/cuny-academic-commons/cac/blob/0d48729d1e275e1d6d684cfa567f95fd6fb8cab5/wp-content/plugins/threewp-broadcast/src/ThreeWP_Broadcast.php#L176

I don't understand how the plugin is supposed to work, and my hypothesis is that it doesn't in fact work at all. It creates a single database table for the entire network - see the base_prefix bit here

https://github.com/cuny-academic-commons/cac/blob/0d48729d1e275e1d6d684cfa567f95fd6fb8cab5/wp-content/plugins/threewp-broadcast/src/traits/broadcast_data.php#L32. And the table on the production site has 0 entries in it.

I've looked through the database methods in the plugin and I don't see any way in which adding an auto-increment column would cause a problem with reads or writes. Here's what I'll be running, on the single database table that we've got: ALTER TABLE wp__3wp_broadcast_broadcastdata ADD COLUMN id INT AUTO_INCREMENT PRIMARY KEY;

Because the plugin doesn't create new tables when activated on individual sites, no further action should be needed.

#3 - 2020-03-31 05:05 PM - Boone Gorges

bp_activity_sitewide_test

This table doesn't have any entries after 2009, and isn't referenced anywhere in the codebase. Judging by the name, it's a remnant of some tested but unimplemented feature many years ago. I made a backup of the table and then dropped it from the production database.

#4 - 2020-04-01 12:15 PM - Boone Gorges

- File cas-count.php added

cas_count, cas_image

These tables are created by the Peter's Custom Anti-Spam plugin

https://github.com/cuny-academic-commons/cac/blob/5dc1aab7f6e7114132e6dc9ba9a05a561ae7e699/wp-content/plugins/peters-custom-anti-spam-image/custom_anti_spam.php#L946

The tables already have an index on the `id` columns, it's just not marked as primary.

The plugin has not been updated in 6+ years, so I've added the necessary declaration to the schema in the plugin itself.

<https://github.com/cuny-academic-commons/cac/commit/5dc1aab7f6e7114132e6dc9ba9a05a561ae7e699>. I've pulled this change as a hotfix to the production site so that future tables are not created without the primary key.

For existing sites, I queried the db (SHOW TABLES LIKE '%_cas_%') for an updated list of existing tables, and then I used that to create a script that altered the existing tables as necessary. I've attached the script for reference.

#5 - 2020-04-01 02:11 PM - Boone Gorges

cp_ad_pop_daily, cp_ad_pop_total

These are created by the Classipress theme.

<https://github.com/cuny-academic-commons/cac/blob/caae201cf1f208db91131ee5607d92a5f4059f79/wp-content/themes/classipress/includes/admin/install-script.php#L61> It is currently being maintained by our team (no public updates in many years). It appears to be active on only two sites, and is not network enabled.

There are existing indexes, but they're not flagged as primary. I've made the necessary change to the theme in

<https://github.com/cuny-academic-commons/cac/commit/caae201cf1f208db91131ee5607d92a5f4059f79>. This change is deployed as a hotfix.

I've also applied the necessary table alterations to the existing tables. There were only four of them - two of each.

#6 - 2020-04-01 02:19 PM - Boone Gorges

cp_project_users

Created by the CollabPress plugin

<https://github.com/cuny-academic-commons/cac/blob/bdaa07a420791daa287dbc8ff54f7830ebe6bb4e/wp-content/plugins/collabpress/includes/update.php#L8>. This plugin is no longer maintained, and we've applied a few compatibility fixes of our own, so we essentially run a fork. I've changed the existing UNIQUE key declaration in the schema so that it's PRIMARY

<https://github.com/cuny-academic-commons/cac/commit/bdaa07a420791daa287dbc8ff54f7830ebe6bb4e>. This has been deployed as a hotfix.

I've made the corresponding change to existing tables on the production site.

#7 - 2020-04-01 03:30 PM - Boone Gorges

ebd_link, ebd_posted_data

Created by the email-before-download plugin

<https://github.com/cuny-academic-commons/cac/blob/043e9df0d8b88ae906519ece7f3d06305a6744d3/wp-content/plugins/email-before-download/includes/class-email-before-download-activator.php#L30>. The tables have indexes, but they're not flagged as PRIMARY.

This plugin can no longer be activated on the Commons. Starting about two years ago (from what I can gather) it began requiring Contact Form 7 and Download Monitor <https://wordpress.org/plugins/download-monitor/>. We don't have the latter plugin, which means that the plugin has been un-activatable for some time now. As such, I've hidden it from the Plugins panel.

<https://github.com/cuny-academic-commons/cac/commit/f89348e67be7963460430264d175e36a49c5a122>. As such, I don't think we need to make any mods to the plugin itself - no one will activate it, so no new tables will be created. I'll evaluate at some point in the future whether and how we can fully remove the plugin from the codebase. See [#10380](#).

The tables only exist on a single site in the network. I've added the necessary primary key on the two tables belonging to that site.

#8 - 2020-04-01 03:44 PM - Boone Gorges

email_user, weu_subscribers, weu_unsubscriber, weu_user_notification

These tables are created by the email-users plugin.

<https://github.com/cuny-academic-commons/cac/blob/043e9df0d8b88ae906519ece7f3d06305a6744d3/wp-content/plugins/wp-email-users/wp-email-functions.php#L139>

We deprecated this plugin in [#9289](#). As such, I don't think we need to make any mods to the plugin, as it won't be permitted to create any new tables.

The tables already have ID indexes, just not PRIMARY ones. I've made the necessary modifications on the existing tables. (They only existed on four sites in the network.)

#9 - 2020-04-02 04:59 PM - Boone Gorges

- File *askit-table-keys.php* added

etcomment_rating, etright_answer

These tables are created by the AskIt plugin. These are the first tables I've looked at that are a bit more complex to address.

First, the theme creates the tables without any key at all, and without an auto_increment field that would be guaranteed unique. So I'm forced to add a new column to each, which will serve as the primary key. I took a look at the places in the theme where the tables are queried to see if the additional column would cause issues. I found that the SELECT statements either use COUNT(*) or specific field names, so that there's no chance of having the new ID fields pollute any post-query activity (such as a foreach loop on the returned fields). As such, I believe the new fields are safe to add.

Second, the theme is network enabled on the Commons. This means that we need to worry about the creation of future tables. Moreover, the theme is from Elegant Themes, and they appear to make at least occasional updates, meaning it's unwise to modify the theme directly. The theme uses a "pluggable" function to create tables, so I thought about overriding it, but in this case I worry that there may be upstream changes which would not be reflected in my overrides. Instead, I added a 'query' filter callback, which tries to be as targeted as possible, which adds the new column and flags it as a PRIMARY KEY. <https://github.com/cuny-academic-commons/cac/commit/3bb8bf3194a9ab8d767ed8683b92a3036d1510f>

I used the attached script to fix all existing tables on the installation. There were several hundred of them.

#10 - 2020-04-03 04:05 PM - Boone Gorges

firestats_archive_countries, firestats_archive_pages, firestats_archive_referrers, firestats_archive_sites, firestats_archive_useragents, firestats_options, firestats_rss_subscribers, firestats_url_metadata, firestats_user_sites

These tables all belong to Firestats. This plugin is no longer in the codebase, and has not been since

<https://github.com/cuny-academic-commons/cac/commit/ef582d8c8421b9b104a78d60380085108bf9e1f0> in 2013. See [#2614](#). As such, these tables - very large in some cases - are sitting in the database, just taking up space. Not only does this mean that we don't need to make any codebase mods to ensure that future tables have primary keys, but it means that we can get rid of the tables altogether. I have done so. I first took a backup of every table matching '%firestats%', which is stored in my home directory on ldv2. I then dropped all those tables.

#11 - 2020-04-03 04:40 PM - Boone Gorges

flickr_post

These tables were created by the photo-dropper plugin. Looking through the logs, it appears that the plugin created tables prior to 2012, but no longer does so. So no mods to the plugin or codebase are necessary.

I've modified the existing tables so that the auto-increment uid column is the designated primary key.

#12 - 2020-04-03 05:12 PM - Boone Gorges

news_announcement

Created by news-announcement-scroll plugin. This plugin is still actively used, so we need to ensure that future tables have the primary key. Because the plugin uses dbDelta(), we're able to use a targeted filter to ensure this:

<https://github.com/cuny-academic-commons/cac/commit/086c7444439d7f6fec356e01b3b3dc0f8f2f14b9>

I've made the necessary changes to existing tables.

#13 - 2020-04-06 04:05 PM - Boone Gorges

podpress_stats

It looks like this is probably created by the plugin podpress. But this plugin directory is malformed - it looks like the proper plugin files must have been deleted at some point. See

<https://github.com/cuny-academic-commons/cac/tree/086c7444439d7f6fec356e01b3b3dc0f8f2f14b9/wp-content/plugins/podpress>. The directory has not been touched since 2010, which leads me to believe that this plugin was activated (and intact) on the production site before we moved to version control, but was messed up before the GitHub migration. This is partly confirmed by the database, which shows that only four sites - 1, 26, 185, and 186, all low numbers - have the corresponding tables. In any case, I don't think that any further steps are necessary to protect against future tables being created incorrectly.

The existing tables have a unique index, but it's just not marked as Primary. I've added the primary key index to each.

#14 - 2020-04-07 12:23 PM - Boone Gorges

revslider_css, revslider_layer_animations, revslider_navigations, revslider_navigations_bkp, revslider_sliders, revslider_slides, revslider_static_slides, revslider_static_slides_bkp

Created by revslider (Slider Revolution). All the tables are created with a UNIQUE KEY (id), but none is marked as PRIMARY. I've created a filter for dbdelta_queries that will catch these in the future.

<https://github.com/cuny-academic-commons/cac/commit/e4d24919b4220697dbd6d7648e9f36b2ffa2509>

I used a script to fix the issue for existing tables; see attached. There's a small handful of tables that already had a PRIMARY KEY, and I have a clause in the script to skip them. (Looks like revslider once defined some of these columns as PRIMARY, but then switched away. Not sure of the reasoning because it's a private plugin, but I can't think of a reason why our change would break anything, since PRIMARY implies UNIQUE.)

#15 - 2020-04-08 11:12 AM - Boone Gorges

- File *tantan-flickr-table-keys.php* added

silas_flickr_cache

Created by the tantan-flickr plugin:

<https://github.com/cuny-academic-commons/cac/blob/e4d24919b4220697dbd6d7648e9f36b2ffa2509/wp-content/plugins/tantan-flickr/flickr/class-admin.php#L96>

This plugin is blacklisted for activation on the Commons, so no changes are necessary to account for future tables.

<https://redmine.gc.cuny.edu/issues/10564#note-19>

Existing tables were created without a guaranteed-unique field, so a new column is necessary. I reviewed all SELECTs against this table, and all of them specify fields, so adding new data should not cause issues. See attached script.

#16 - 2020-04-08 11:37 AM - Boone Gorges

- File *statpress-primary-key.php* added

statpress

Generated by the statpress-visitors plugin.

<https://github.com/cuny-academic-commons/cac/blob/1.16.x/wp-content/plugins/statpress-visitors/statpress.php#L139>

The plugin is disabled for activation across the Commons, so no mods are necessary for future tables.

Existing tables already have a unique id column, so I just needed to make it a primary key. See attached script. It took a long time to run because of the size of some of the statpress tables.

#17 - 2020-04-08 11:52 AM - Boone Gorges

- File *woo-custom-nav-primary-key.php* added

woo_custom_nav_menus, woo_custom_nav_records

Created by a number of themes from Woo. Here's an example:

<https://github.com/cuny-academic-commons/cac/blob/1.16.x/wp-content/themes/bigeasy/functions/admin-custom-nav.php#L56>

The themes in question are all network-disabled, so the issue of future tables shouldn't arise.

Existing tables all have unique `id` columns, so I used a script to add a primary key to them. See attached.

#18 - 2020-04-08 12:04 PM - Boone Gorges

woo_tables_meta

Appears to have been created by a WooTheme, but the code was removed as part of

<https://github.com/cuny-academic-commons/cac/commit/0b2c66d92ea1d813b92eb4cf717e7edb730b2300> (I didn't look any more closely because it doesn't matter very much). The table is no longer created by any Woo theme in the codebase.

As for existing tables, there's only one on the entire network, so I manually added the necessary index (it already had an auto-increment field).

#19 - 2020-04-08 12:46 PM - Boone Gorges

- File *newsletters-lite-primary-key.php* added

wpmfieldslists

Belongs to newsletters-lite. The table schema defined in the current plugin has a PRIMARY KEY on the rel_id field, but a number of older tables on the Commons do not have this field at all. This suggests that the plugin has an update routine that was not run properly on some sites.

The plugin is not available for activation, and even if it were, newly created tables would have the proper schema, so no action is required there.

For existing tables, I used a script to update the schema to match what the plugin now states. See attached.

#20 - 2020-04-08 03:02 PM - Boone Gorges

yoast_seo_meta

Created by wordpress-seo

<https://github.com/cuny-academic-commons/cac/blob/e4d24919b4220697dbd6d7648e9f36b2fffa2509/wp-content/plugins/wordpress-seo/admin/class-meta-storage.php#L63> The table has a UNIQUE key but not a PRIMARY.

The plugin is widely used and frequently updated, so we must make changes but cannot fork. And it does not use dbDelta() to create the table. So I've used the post-install hook in wordpress-seo to run an ALTER TABLE command, which looks like this:
<https://github.com/cuny-academic-commons/cac/commit/cce5280c17bc89636e75acc163d13d78d9677ab5>

I've run the same query on all existing yoast_seo_meta tables.

#21 - 2020-04-08 03:23 PM - Boone Gorges

zotpress, zotpress_cache, zotpress_images, zotpress_oauth, zotpress_zoteroCollections

These belong to zotpress. See

https://github.com/cuny-academic-commons/cac/blob/2141b4d4fe02b91dd1ce02804e39e349027f6647/wp-content/plugins/zotpress/lib/admin/admin_install.php#L5

A number of these tables are legacy and no longer created by zotpress: zotpress_zoteroCollections, zotpress_images. zotpress_cache used to have a schema without a primary key, but in the current plugin it's created with a primary key. So the remaining tables are zotpress and zotpress_oauth. In each case, there's an existing UNIQUE index on the `id` column. In a dbdelta_queries filter callback, I'm turning this into a PRIMARY KEY.
<https://github.com/cuny-academic-commons/cac/commit/2141b4d4fe02b91dd1ce02804e39e349027f6647>

I looked at the schema of all existing zotpress table types without primary keys, and they all share the same structure of having an `id` field with a unique index. So I was able to write a script that swapped them all for a primary key.

#22 - 2020-04-16 07:40 PM - Lihua Wang

It took me a few days to dump the commons production database and export it to a different server to run the query, since I did not want bring the production db server to its knees. Below is the result of the query:

```
MariaDB [commons_wp_prod]> select tab.table_schema as database_name,
tab.table_name
from information_schema.tables tab
left join information_schema.table_constraints tco
on tab.table_schema = tco.table_schema
and tab.table_name = tco.table_name
and tco.constraint_type = 'PRIMARY KEY'
where tco.constraint_type is null
and tab.table_schema not in('mysql', 'information_schema',
'performance_schema', 'sys')
and tab.table_type = 'BASE TABLE'
and tab.table_schema = 'commons_wp_prod'
order by tab.table_schema,
tab.table_name;
```

```
wp_10897_dls_sus_sheets
wp_10897_dls_sus_signups
wp_10897_dls_sus_tasks
wp_11264_nf3_actions
wp_11264_nf3_action_meta
wp_11264_nf3_chunks
wp_11264_nf3_fields
wp_11264_nf3_field_meta
wp_11264_nf3_forms
wp_11264_nf3_form_meta
wp_11264_nf3_objects
wp_11264_nf3_object_meta
wp_11264_nf3_relationships
wp_1145_nf3_actions
wp_1145_nf3_action_meta
wp_1145_nf3_chunks
wp_1145_nf3_fields
wp_1145_nf3_field_meta
```

wp_1145_nf3_forms
wp_1145_nf3_form_meta
wp_1145_nf3_objects
wp_1145_nf3_object_meta
wp_1145_nf3_relationships
wp_11828_nf3_actions
wp_11828_nf3_action_meta
wp_11828_nf3_chunks
wp_11828_nf3_fields
wp_11828_nf3_field_meta
wp_11828_nf3_forms
wp_11828_nf3_form_meta
wp_11828_nf3_objects
wp_11828_nf3_object_meta
wp_11828_nf3_relationships
wp_1544_smush_dir_images
wp_1855_wp_cloudplugins_log
wp_2045_nf3_actions
wp_2045_nf3_action_meta
wp_2045_nf3_chunks
wp_2045_nf3_fields
wp_2045_nf3_field_meta
wp_2045_nf3_forms
wp_2045_nf3_form_meta
wp_2045_nf3_objects
wp_2045_nf3_object_meta
wp_2045_nf3_relationships
wp_3253_nf3_actions
wp_3253_nf3_action_meta
wp_3253_nf3_chunks
wp_3253_nf3_fields
wp_3253_nf3_field_meta
wp_3253_nf3_forms
wp_3253_nf3_form_meta
wp_3253_nf3_objects
wp_3253_nf3_object_meta
wp_3253_nf3_relationships
wp_3284_nf3_actions
wp_3284_nf3_action_meta
wp_3284_nf3_chunks
wp_3284_nf3_fields
wp_3284_nf3_field_meta
wp_3284_nf3_forms
wp_3284_nf3_form_meta
wp_3284_nf3_objects
wp_3284_nf3_object_meta
wp_3284_nf3_relationships
wp_3350_smush_dir_images
wp_3485_nf3_actions
wp_3485_nf3_action_meta
wp_3485_nf3_chunks
wp_3485_nf3_fields
wp_3485_nf3_field_meta
wp_3485_nf3_forms
wp_3485_nf3_form_meta
wp_3485_nf3_objects
wp_3485_nf3_object_meta
wp_3485_nf3_relationships
wp_3546_nf3_actions
wp_3546_nf3_action_meta
wp_3546_nf3_fields
wp_3546_nf3_field_meta
wp_3546_nf3_forms
wp_3546_nf3_form_meta
wp_3546_nf3_objects
wp_3546_nf3_object_meta
wp_3546_nf3_relationships
wp_3555_nf3_actions
wp_3555_nf3_action_meta
wp_3555_nf3_chunks
wp_3555_nf3_fields
wp_3555_nf3_field_meta
wp_3555_nf3_forms
wp_3555_nf3_form_meta
wp_3555_nf3_objects

wp_3555_nf3_object_meta
wp_3555_nf3_relationships
wp_3588_nf3_actions
wp_3588_nf3_action_meta
wp_3588_nf3_fields
wp_3588_nf3_field_meta
wp_3588_nf3_forms
wp_3588_nf3_form_meta
wp_3588_nf3_objects
wp_3588_nf3_object_meta
wp_3588_nf3_relationships
wp_3651_nf3_actions
wp_3651_nf3_action_meta
wp_3651_nf3_chunks
wp_3651_nf3_fields
wp_3651_nf3_field_meta
wp_3651_nf3_forms
wp_3651_nf3_form_meta
wp_3651_nf3_objects
wp_3651_nf3_object_meta
wp_3651_nf3_relationships
wp_3973_nf3_actions
wp_3973_nf3_action_meta
wp_3973_nf3_chunks
wp_3973_nf3_fields
wp_3973_nf3_field_meta
wp_3973_nf3_forms
wp_3973_nf3_form_meta
wp_3973_nf3_objects
wp_3973_nf3_object_meta
wp_3973_nf3_relationships
wp_4192_dls_sus_sheets
wp_4192_dls_sus_signups
wp_4192_dls_sus_tasks
wp_5009_dls_sus_sheets
wp_5009_dls_sus_signups
wp_5009_dls_sus_tasks
wp_5012_nf3_actions
wp_5012_nf3_action_meta
wp_5012_nf3_chunks
wp_5012_nf3_fields
wp_5012_nf3_field_meta
wp_5012_nf3_forms
wp_5012_nf3_form_meta
wp_5012_nf3_objects
wp_5012_nf3_object_meta
wp_5012_nf3_relationships
wp_6922_nf3_actions
wp_6922_nf3_action_meta
wp_6922_nf3_chunks
wp_6922_nf3_fields
wp_6922_nf3_field_meta
wp_6922_nf3_forms
wp_6922_nf3_form_meta
wp_6922_nf3_objects
wp_6922_nf3_object_meta
wp_6922_nf3_relationships
wp_82_nf3_actions
wp_82_nf3_action_meta
wp_82_nf3_chunks
wp_82_nf3_fields
wp_82_nf3_field_meta
wp_82_nf3_forms
wp_82_nf3_form_meta
wp_82_nf3_objects
wp_82_nf3_object_meta
wp_82_nf3_relationships
wp_8362_dls_sus_sheets
wp_8362_dls_sus_signups
wp_8362_dls_sus_tasks
wp_848_smush_dir_images
wp_9718_nf3_actions
wp_9718_nf3_action_meta
wp_9718_nf3_chunks
wp_9718_nf3_fields

wp_9718_nf3_field_meta
wp_9718_nf3_forms
wp_9718_nf3_form_meta
wp_9718_nf3_objects
wp_9718_nf3_object_meta
wp_9718_nf3_relationships

#23 - 2020-04-17 10:45 AM - Boone Gorges

Thanks, Lihua. I've updated my list and will work through the new items.

#24 - 2020-04-17 10:57 AM - Boone Gorges

- File *sign-up-sheets-primary-key.php* added

dls_sus_sheets, dls_sus_signups, dls_sus_tasks

These are created by the sign-up-sheets plugin.

<https://github.com/cuny-academic-commons/cac/blob/3385bf07c89238a22f8051cdea8362c552e26a1c/wp-content/plugins/sign-up-sheets/sign-up-sheets.php#L958> The tables already have a unique key, and they just need to be made primary.

In <https://github.com/cuny-academic-commons/cac/commit/3385bf07c89238a22f8051cdea8362c552e26a1c> I added a `dbdelta_query` to change the UNIQUE KEY to PRIMARY KEY on table creation.

I've run a script to create primary indexes on each existing production table.

#25 - 2020-04-17 11:47 AM - Boone Gorges

- File *ninja-forms-primary-key.php* added

nf3_action_meta, nf3_actions, nf3_chunks, nf3_field_meta, nf3_fields, nf3_form_meta, nf3_forms, nf3_object_meta, nf3_objects, nf3_relationships

These are created by ninja-forms. An example:

<https://github.com/cuny-academic-commons/cac/blob/36216631331ef0d504503b30589600d1ca9c8284/wp-content/plugins/ninja-forms/includes/DataBase/Migrations/ActionMeta.php#L27>

I've reviewed each table type and confirmed that (a) they all have UNIQUE keys on the ``id`` field, so it is safe to create PRIMARY keys on that field, and (b) they all use `dbDelta()` for schema modifications, so we can use a `dbdelta_queries` callback. I've implemented this in <https://github.com/cuny-academic-commons/cac/commit/36216631331ef0d504503b30589600d1ca9c8284>

I've also run a script to update all the existing tables as needed. See *ninja-forms-primary-key.php*.

#26 - 2020-04-17 12:19 PM - Boone Gorges

smush_dir_images

Created by the wp-smushit plugin

<https://github.com/cuny-academic-commons/cac/blob/3f0ca8903fe6c587cad2c7684773d1b1e6abc397/wp-content/plugins/wp-smushit/core/modules/class-dir.php#L340>. The table has a unique key that needs to be made primary.

As far as I can see, the plugin intends to create just a single table for the whole network, so it's not clear to me (a) that we need to account for the creation of future tables, and (b) why there are three tables on the network. Out of an abundance of caution, I've added a `dbdelta_queries` callback to modify the CREATE TABLE query in the plugin <https://github.com/cuny-academic-commons/cac/commit/3f0ca8903fe6c587cad2c7684773d1b1e6abc397>

I've modified the three existing production tables to add primary indexes.

#27 - 2020-04-17 12:40 PM - Boone Gorges

wp_cloudplugins_log

Created by the use-your-drive plugin.

<https://github.com/cuny-academic-commons/cac/blob/54b73761cc7fe9b648b10783f243f1baf783995b/wp-content/plugins/use-your-drive/includes/Events.php#L843>

This is a UNIQUE key that must be forced to PRIMARY. `dbdelta_queries` callback added in <https://github.com/cuny-academic-commons/cac/commit/54b73761cc7fe9b648b10783f243f1baf783995b>

I've made the change to existing tables.

#28 - 2020-05-13 09:43 PM - Boone Gorges

- Related to Feature #12800: Automated scan for database schema validity added

#29 - 2020-05-13 09:45 PM - Boone Gorges

I've opened [#12800](#) to track the automated scanning of the codebase to keep out future violations of the primary-key requirement. As I build that tool, I'm finding a few more instances that weren't caught by Lihua's original scan - perhaps because they've never in fact created tables on the Commons (though they could in theory do so).

The first example is bloom. We appear to maintain a fork, so I've added my fix directly to the plugin.

The first example of this is <https://github.com/cuny-academic-commons/cac/commit/c7af39e712a8cb73c5a9e4028382889adc72f06b>

#30 - 2020-05-13 10:16 PM - Boone Gorges

Another one in amazon-link: <https://github.com/cuny-academic-commons/cac/commit/7b70c48f8f56aaf2174189df9fb9cf7b922428b5>

#31 - 2020-05-14 12:48 PM - Boone Gorges

monarch: <https://github.com/cuny-academic-commons/cac/commit/3ea50a9eb3a60aab1db4d348d3d3e46c1adeefb2>

#32 - 2020-07-14 01:15 PM - Boone Gorges

- Status changed from New to Resolved

I think we are done here - the linter ensures that new instances are not introduced.

Files

cac-database-tables-without-primary-key.txt	51.3 KB	2020-03-31	Boone Gorges
cas-count.php	1.48 KB	2020-04-01	Boone Gorges
askit-table-keys.php	539 Bytes	2020-04-02	Boone Gorges
tantan-flickr-table-keys.php	416 Bytes	2020-04-08	Boone Gorges
statpress-primary-key.php	301 Bytes	2020-04-08	Boone Gorges
woo-custom-nav-primary-key.php	306 Bytes	2020-04-08	Boone Gorges
newsletters-lite-primary-key.php	431 Bytes	2020-04-08	Boone Gorges
sign-up-sheets-primary-key.php	350 Bytes	2020-04-17	Boone Gorges
ninja-forms-primary-key.php	349 Bytes	2020-04-17	Boone Gorges