

TEN7AUDIT REPORT EXAMPLE

<https://somesite.org/>

Prepared for
Example Organization

Submitted
Jan 1, 2018



612-868-7884 · hello@ten7.com · ten7.com · @TEN7

TABLE OF CONTENTS

VERSIONS OF THIS DOCUMENT	7
OVERVIEW	7
Audit methodology	7
CRITICAL FINDINGS	7
Multiple Drupal 6 modules are installed and active	7
Solution	8
Risk	8
Custom code creating PHP Notices	8
Solution	8
Risk	8
ImageAPI module enabled	8
Solution	8
Risk	9
PERFORMANCE FINDINGS	9
Site served with PHP 5.6	9
Solution	9
Risk	9
Drupal search used as front-line feature	9
Solution	9
Risk	9
Lack of HTTP Caching	10
Solution	10
Risk	10
Lack of Database Query Caching	10

Solution	10
Risk	10
Core Block UI used for Block Placement	10
Solution	10
Risk	10
THEME AND UX FINDINGS	11
Design is not mobile friendly	11
Solution	11
Risk	11
Site organization is noisy	11
Solution	11
Risk	11
Overlay module in use	11
Solution	11
Risk	12
CONTENT FINDINGS	12
Content types well structured	12
Unused Content Types	12
Solution	12
Risk	12
INFRASTRUCTURE FINDINGS	12
Site is not backed up	12
Solution	12
Risk	12
Shared hosting in use	13
Solution	13
Risk	13

Drush Unusable	13
Solution	13
Risks	13
Database is very large	14
Solution	14
Risk	14
Deleted field tables present in database	14
Solution	14
Risk	14
CI Not Possible	14
Solution	14
Risks	14
SITE CODE FINDINGS	15
Core and contrib modules have been modded	15
Solution	15
Risk	15
Site Not Git Backed	15
Solution	15
Risk	15
Site Files Should be Moved out of Web Root	16
Solution	16
Risk	16
Large number of unmanaged files in files directory	16
Solution	17
Risk	17
Contrib Modules Should be Moved to sites/all/modules/contrib	17
Solution	17

Risk	17
Custom Modules Should be Moved to sites/all/modules/custom	17
Solution	17
Risk	17
Features module present, but underutilized	18
Solution	18
Risk	18
Security Findings	18
PHP filter module in use	18
Solution	18
Risk	18
Secure Pages unnecessary	18
Solution	18
Risk	19
Analytics Findings	19
Tier 1 Recommendations	19
Remove/Update Drupal 6 modules	19
Remove bad-practice and unnecessary modules	19
Update custom code to correct PHP notices	19
Revert modded core and contrib code	20
Implement a backup mechanism	20
Tier 2 Recommendations	20
Consider a change in hosting	20
Implement a Git-based workflow	20
Create a new responsive web theme	21
Tier 3 Recommendations	21
Move CiviCRM to a separate site	21

VERSIONS OF THIS DOCUMENT

Date	Author	Description
01.12.2018	Tess Flynn	V1.0 Initial draft of this document.
01.17.2018	Madeleine Lowry	Review and feedback
01.24.2018	Les Lim	Review audit report

OVERVIEW

Audit methodology

The site was evaluated locally using a combination of Site Audit, Hacked module and manual analysis. Typically this would be done in situ on the live site, but because Drush was not operational, the analysis was performed on a local copy.

CRITICAL FINDINGS

Multiple Drupal 6 modules are installed and active

At least 23 Drupal 6 modules were located within the site code. These modules are incompatible with Drupal 7, and can cause performance and stability issues. The following modules are incompatible:

- Content
- FileField
- FileField Meta
- ImageField
- Option Widgets
- TPC Staff Profiles
- Tpc_calendar
- Twitter Feed
- ImageCache
- ImageCache UI
- Autoload

- Backup and Migrate
- DBTNG
- Menu Trails
- TPC General
- Conditional Actions
- Free Order
- jQuery UI
- Vertical Tabs
- Bonus: Paged Feed
- Bonus: Panels
- Bonus: Views Export
- Views Max Limit

Solution

Update the modules to Drupal 7 versions where possible, deprecate or remove functionality elsewhere.

Risk

Until updated, some functionality may not work correctly or as intended. The incompatible modules may also be affecting site performance.

Custom code creating PHP Notices

Several custom modules, create frequent PHP notice messages per request. This affects the site performance.

Solution

Update the custom code to eliminate the error messages.

Risk

None.

ImageAPI module enabled

The ImageAPI module has been enabled on the site. This module is no longer necessary for most Drupal 7 sites as functionality has been moved into Drupal core.

Solution

Remove the module, rely on core's built-in gd image toolkit, or the imagemagick module.

Risk

Minimal, unless a hard dependency has been created in custom code.

PERFORMANCE FINDINGS

Site served with PHP 5.6

While PHP 5.6 is a supported version of PHP for Drupal 7, additional performance can be gained by moving to PHP 7. PHP 7 has additional performance enhancements and caching abilities compared to PHP 5.6. PHP 5.6 is no longer an actively supported version, and will no longer receive security updates starting on 1/1/2019.

Solution

Consider moving to PHP 7. This may only require updating the .htaccess file to point to the proper PHP CGI wrapper script.

Risk

Some custom modules and out of date contrib modules may be incompatible with PHP 7. Local testing is required to be sure there are no serious incompatibilities.

Drupal search used as front-line feature

While Drupal provides a built-in search feature, it relies heavily on the database to index and perform search options. It has a limited ability to customize search results. The performance of Drupal search is often unsatisfactory compared to alternatives.

Solution

Consider switching to a custom search solution powered by Apache Solr. Solr is a highly customizable and Free and Open Source search engine that integrates well with Drupal.

Risk

Solr implementations require the service to be installed on infrastructure accessible from the site. This may be available from Media Temple as a value-added service at additional cost.

Lack of HTTP Caching

Presently, there is no front-line HTTP caching of the site. Instead, it relies entirely on Drupal's internal abilities. It is highly recommended to use an external, reverse proxy HTTP caching system to improve load times for anonymous users.

Solution

Implement Varnish as a reverse proxy HTTP cache.

Risk

Varnish is a server-level technology and must be installed and accessible local to the web server. Media Temple may offer Varnish as a value-added service at additional cost.

Lack of Database Query Caching

Drupal is heavily reliant on the database server for storing content and generating pages. While many database servers such as MySQL offer some internal caching, an external cache can greatly improve performance.

Solution

Implement Memcache or Redis as a database object cache.

Risk

Memcache and Redis are both server-level technologies and need to be installed local to the web server.

Core Block UI used for Block Placement

Drupal's built-in UI has been used for Block Placement. While this works, it has a known performance problem as the number of blocks increase. With the core UI, blocks are loaded on each page regardless of if they appear on each page.

Solution

Transition to block placement using Context module. This removes the performance constraint.

Risk

Minimal. New context rules can be created all at once, or iteratively. When done properly, changes should be transparent to the end-user.

THEME AND UX FINDINGS

Design is not mobile friendly

The theme was designed using a desktop-first modality with a fixed width. This requires visitors using smartphones into a frustrating cycle of zoom-pan-tap.

Solution

Consider creating a new theme based on Responsive Design principles. This would allow the site to adapt to different screen sizes and modalities easily.

Risk

Creating a new theme is a time intensive process. Often, a new theme may also suggest changes to menu navigation, site organization, and content changes.

Site organization is noisy

The site presents too much information at once, resulting in a design that is busy and difficult to parse. New users may be confused by the site organization or have trouble finding anything.

Solution

Consider creating a new theme that reduces the number of blocks and sidebars. Many modern designs are organized along fluid-width stripes that present single informational or navigation items.

Risk

Like any change in site organization, the process can be disruptive to long-time users.

Overlay module in use

The Overlay module allows administration pages to be viewed in a Javascript modal. This functionality is overcomplicated and unnecessary. Unfortunately, it is enabled by default in Drupal 7.

Solution

Disable the Overlay module.

Risk

Minimal, as this over affects site administrators and developers.

CONTENT FINDINGS

Content types well structured

Most content types on the site are well structured and take advantage of Drupal's field system effectively.

Unused Content Types

The catalog_card and membership_history content types are presently unused. While this doesn't harm anything, it can lead to editor and administrator confusion.

Solution

Remove the unused types.

Risk

None.

INFRASTRUCTURE FINDINGS

Site is not backed up

We could not determine if a backup and recovery strategy was in place. While the Backup and Migrate module was installed, it is an incompatible Drupal 6 version, meaning local backups have not been taken. An external or hosting-provider backup solution may be in place that was not discovered during the course of the audit.

Solution

Implement a local and off-site backup strategy. This can rely on the Drupal 7 version of Backup and Migrate, TEN7's Tractorbeam, or another solution.

Risk

The lack of backups is a greater risk than implementing backups. If the hosting provider is unable to recover the site, data may be irrevocably lost.

Shared hosting in use

Pair networks is a shared hosting provider. While this is cost effective, it does constrain the available resources to run the site. Furthermore, Pair may not be able to provide additional infrastructure services such as Apache Solr or Memcache.

Solution

Consider moving to a dedicated or managed hosting provider. TEN7 can assist you with discussing available options.

Risk

Moderate. Migrating to new server infrastructure can be difficult as well as disruptive. A new production environment must be created in advance, and tested thoroughly. This can reduce downtime to hours.

Drush Unusable

Drush is a command line application used to interact with Drupal sites. It allows special developer and administrator access to the site to inspect content, perform maintenance, and do other operations. Drush is critical for any Drupal site on Continuous Integration or an automated deploy system. Currently Drush is unusable for the following reasons:

- Drush is an out of date dev version (5.0). The current version is 9.
- The command line environment is PHP 7, which is incompatible with Drush 5.

Solution

Update Drush to at least 7.4.0, although Drush 9 may be compatible with both PHP 7 and Drupal 7.

Risks

Minimal.

Database is very large

When actively deployed, the database requires a minimum of 3GB of disk space. The largest table is watchdog, containing the Drupal log. This table should be emptied as PHP notices and other issues are resolved. Furthermore, the search_index table would no longer need to be present if Apache Solr were implemented.

Solution

Resolve PHP error notices and search infrastructure, then truncate the appropriate tables.

Risk

Minimal. The watchdog table can be truncated at any time, and the search_index table can be rebuilt on-demand.

Deleted field tables present in database

There are several database tables that correspond to deleted fields. These cause unnecessary database overhead as the data within these tables is not used.

Solution

Take a backup of the tables as they exist now, then drop the tables.

Risk

None if a database backup is taken.

CI Not Possible

At the moment, is not possible to automate the deployment (“Continuous Integration”) of the site. Therefore, each deployment requires manual administrator work to build the release and deploy it to infrastructure.

Solution

While it is technically possible to do this on existing infrastructure, it would require a fully functional version of Drush. This would allow the use of automation tools to deploy the site.

Risks

Minimal, unless a server migration is involved.

SITE CODE FINDINGS

Core and contrib modules have been modded

Several core and contrib source files have been modified compared to their official versions on Drupal.org. Many of these included additional debug code, but there were some modifications that appear to be intended to be performance tweaks. These pose difficulty with the ongoing maintenance of the site as an update may revert key changes.

The following modules have been modified:

- node
- search_files
- uc

Solution

Identify and examine changes carefully. If critical, extract them as a patch file and persist them to a known patches directory managed by git. Otherwise, revert the changes completely.

Risk

Moderate. Modifications need to be examined closely so they presence or absence does not affect the site operation.

Site Not Git Backed

At present, there's no evidence to suggest the site is managed by a version control system such as Git. Source control systems maintain a running history of all changes made to a site's code, who made them, and when. It also acts as a backup in the case of total server failure, as the source control repository is often retained by a separate system.

Solution

Choose or implement a source control system to house the site code. This could be an internally managed system such as Gitlab, or a publicly available service such as Gitlab.com, Bitbucket, or Github. Many web agencies such as TEN7 will maintain the site in a source control system as a matter of course.

Risk

Trivial. This does not affect the running or operation of the site, but rather its current and ongoing development.

Site Files Should be Moved out of Web Root

The default location for uploaded files in a Drupal site is in the **site/default/files** subdirectory. The consequence is added complexity to deployment and backup workflows as it is sizable and stored alongside site code. you currently has 55G of uploaded files.

Solution

Move the files directory out of the web root. A symlink could then be used in place of **site/default/files**, maintaining all current URLs. The web server must be configured to use symlinks appropriately.

Risk

Minimal. Many web servers are already configured to follow symlinks out of the box. Some downtime is associated with the file directory move, but this is a one-time operation.

Large number of unmanaged files in files directory

There are a large number of unmanaged files in the site files directory. This isn't a problem, per-se, but it should be understood as it can complicate site migrations.

The following lists the top five items by size in the files directory:

Name	Type	Size
filpbooks/	Directory	49GB
csa31.pdf	File	706MB
csa32.pdf	File	681MB
csa1.pdf	File	635MB
csa2.pdf	File	631MB

Solution

While this does not affect the working of Drupal, the large size of the Filpbooks directory is concerning. A possible solution would be to consider moving the Filpbooks directory to a dedicated subdomain.

Risk

The biggest risk will be broken links. An .htaccess redirect can be configured to automatically redirect anyone to the updated location.

Contrib Modules Should be Moved to sites/all/modules/contrib

In Drupal 7, additional modules are stored in the **sites/all/modules** directory. Best practice suggests that any modules that are downloaded from Drupal.org are stored in a single subdirectory, **sites/all/modules/contrib/**. This makes them easy to identify and plays well with any Composer-managed build process.

Solution

Move any contrib modules to **sites/all/modules/contrib/**. This may involve some database manipulation and cache regeneration in order for Drupal to pick up the new module locations.

Risk

Minimal. A small amount of downtime might be involved during the move while any database updates or cache clearing is performed.

Custom Modules Should be Moved to sites/all/modules/custom

While you can store custom modules in **sites/all/modules/**, it is best practice in Drupal 7 to move them to **sites/all/modules/custom/**. This makes them easily identifiable to developers.

Solution

Move any custom modules to **sites/all/modules/custom/**. This may involve some database manipulation and cache regeneration in order for Drupal to pick up the new module locations.

Risk

Minimal. A small amount of downtime might be involved during the move while any database updates or cache clearing is performed.

Features module present, but underutilized

The Features module is used to persist Drupal configurations to code. While this functionality is on the site, the modules generated by Features appear to be out of date. Furthermore, many site-wide configurations are not managed by Features.

Solution

Create more feature modules to persist key site configurations to code. This can include content types, user permissions, and other key configurations.

Risk

None. Persisting configurations to code allows quicker detection and remediation of misconfigurations.

Security Findings

PHP filter module in use

The PHP filter module is in use on the site. This poses a security risk as it allows PHP code to be stored in content. This functionality is enabled by default in Drupal 7, but is often unused. It is a possible attack vector for malicious users.

Solution

Determine if any content relies on PHP filter. Remove and PHP code in content and reimplement as custom modules instead. Finally, disable the PHP module.

Risk

Some sites extensively rely on the PHP module, and disabling it outright without extensive testing can result in broken functionality.

Secure Pages unnecessary

The Secure Pages module is in use on the site. The intention of this module is to use HTTPS on key pages (such as store or login pages) while the majority of traffic relies on the

unencrypted site. For you, the configuration is apparently unnecessary, as all traffic is directed to HTTPS. The www.* subdomain is served unencrypted, but is redirected to the apex domain.

Solution

Implement the redirect with .htaccess, and remove the Secure Pages module.

Risk

Minimal. Some browsers do cache redirects for 24 hours, and can cause some disruption until the cache expires internally.

ANALYTICS FINDINGS

Google Analytics appears to be configured correctly and is functioning properly on the site. The account used to access analytics details was not provided to TEN7.

TIER 1 RECOMMENDATIONS

The top tier of recommendations is intended to resolve critical, and time-dependant issues on the site. Tier 1 recommendations must be enacted before other issues can practically be resolved.

Remove/Update Drupal 6 modules

The highest priority is to remove all Drupal 6 modules on the site. If the functionality provided by the modules is still desired, the Drupal 7 version should be installed instead.

Remove bad-practice and unnecessary modules

The following modules should be removed:

- PHP
- ImageAPI
- Secure Pages

Update custom code to correct PHP notices

Custom code should be fixed so that it does not generate frequent PHP notices. This removes a potential performance constraint.

Revert modded core and contrib code

Continuing to run modded core and contrib code can cause difficult to diagnose issues and complicate software updates unnecessarily.

Implement a backup mechanism

At present, the site is not backed up either locally or remotely. If the server hosting the site becomes unrecoverable, data may be lost. For this reason, it is best to implement a regular backup mechanism to persist site files and the database to a remote location. Ideally, a multi-tier backup solution would be used to give you multiple days, weeks, and months of backups on an external server.

TIER 2 RECOMMENDATIONS

The following are near-term recommendations help you to maximize your current size without fundamentally altering the technology behind it.

Consider a change in hosting

Pair Networks may not be an appropriate host for you any longer. The amount of server resources available to the site depends on the load of other sites on the same server. This can create additional performance constraints. Migrating the site to new, modern hosting would allow these constraints to be removed, as well as correct the following issues:

- Leverage additional infrastructure resources such as HTTP caching, query caching, and improved search functionality.
- Correct the PHP version mismatch between the web environment and the command line environment.

Implement a Git-based workflow

The site code should be maintained by a source control system such as Git. This will help standardize the management of the site using freely available tools. There are many options for third-party git hosting, many with free tiers:

- Github
- Gitlab
- Bitbucket

Many web agencies provide git hosting as part of a support agreement.

Create a new responsive web theme

The you is busy, dated, and does not work well on smartphones. An updated design can be created that can leverage a cleaner, more responsive appearance. Furthermore, a new design can bring gains in site organization as well as accessibility (a11y) compliance.

TIER 3 RECOMMENDATIONS

The third tier or recommendations represent longer term goals which lay a foundation for the long-term sustainability and success of your site.

Move CiviCRM to a separate site

CiviCRM can add complexity for some 3rd party agencies whose primary specialty is the development and maintenance of Drupal-powered websites. Provided that the integration of the CiviCRM data on the site is loose or non-existent, moving it to a dedicated “backoffice” site can reduce the cost and difficulty in locating a 3rd party agency to manage the site.

Create a new Drupal 8 site

While Drupal 7 is still supported, the most current version of Drupal is Drupal 8. Given the number of issues on the site, building a new Drupal 8 site may be more cost effective than correcting the issues on the existing site. Creating a new site would also provide the opportunity to implement new themes, functionality, and content organization without affecting the existing site.