# Site technologies and requirements

## Functioning LAMP stack

The Meetings Imagined website runs off of a traditional Linux, Apache, MySQL, PHP stack.

* Debian 7.2
  + The specific distribution of Linux does not impact the functionality of the site. As long as this version is somewhat recent and continues to receive security updates as needed.
* Apache 2
* MariaDB 5.5
  + MariaDB is a high performance, drop-in MySQL compatible alternative.
  + <https://mariadb.org/>
* PHP 5.4 or PHP 5.5

## Task scheduler

The application assumes that a task scheduler of some sort is available for the scheduling of maintenance and report jobs. Cron is suitable for this purpose.

## Memcached

Memcached is an object caching system used for high performance storing and retrieval of information. Basic usage is as a key/value store for information that is intensive to calculate and does not frequently change.

<http://memcached.org/>

A suitable alternative would be Redis.

## Zend Optimizer

Zend Optimizer compiles PHP scripts at run-time into byte code that is stored in memory for retrieval on page request. The compiled scripts speed up page execution time by alleviating the amount of work the PHP executable needs to do for each request.

<https://wiki.php.net/rfc/optimizerplus>

Alternatives include APC and XCache, though it is recommended to stick with Zend Optimizer when possible. Zend Optimizer receives current updates and will be bundled with newer versions of PHP, resulting in tighter integration, better optimization, and continued development.

## Drupal 7

Drupal 7 is a robust CMS and application framework with thousands of users, notable enterprise-level clients (http://whitehouse.gov, for example), and a large community.

<https://drupal.org/drupal-7.0>

## Solr 4.6

Solr is a powerful search system built upon Lucene that allows the indexing of and complex querying against arbitrary document types over a REST API.

<http://lucene.apache.org/solr/>

Solr has additional requirements on whatever sever(s) host it

* Java 1.5 or higher
* Tomcat 5.5 or higher

## SimpleSAML PHP

SimpleSAML PHP is the most popular and robust option available for integrating PHP applications into a SAML system.

<https://simplesamlphp.org/>

SimpleSAML PHP requires the following PHP modules be installed:

* date
* dom
* hash
* libxml
* openssl
* pcre
* SPL
* zlib
* mcrypt
* memcache

The other requirements are met through the general Drupal requirements and other information listed in this document.

# Community contributions

A core concept for Drupal applications is the use of community-contributed modules (plugins). Involving community work results in ongoing updates, security reviews, and other improvements by thousands of individuals. The Meetings Imagined website is no different, and leverages over 115 modules (including those that come with Drupal).

Most of these modules are used as a starting point for the custom features found on Meetings Imagined, while others provide “must-have” functionality that can be found on any site.

Some major components include:

**Solr integration:**  We were able to leverage the existing Search API (<https://drupal.org/project/search_api>) and Search API Solr (<https://drupal.org/project/search_api_solr>) modules to provide the basic integration with the Solr server. Additional work was then done using custom modules to refine the functionality to that which was required on the Meetings Imagined site.

**Memcached integration:** Instead of spending development time tying Drupal into memcache we were able to use the existing, community maintained Memcache module (<https://drupal.org/project/memcache>).

**SEO**: There are numerous community modules that assist in SEO optimization, ranging from automatic path aliasing (<https://drupal.org/project/pathauto>) to generating and including relevant metatags (<https://drupal.org/project/metatag>) without user intervention.

**Security:** As with SEO, we include a number of community maintained modules that harden Drupal against various web-based attacks. Some of these modules include SecKit (<https://drupal.org/project/seckit>) and UEP (<https://drupal.org/project/username_enumeration_prevention>).

## Custom modules

Any Marriott specific functionality is provided in the form of a custom, application-specific module. There are 27 such modules on the site at this time. Below is a list of each with a high level description of what they do.

1. mcm\_fontawesome
   1. Houses theme functionality for the icons throughout the site.
2. mrt\_articles
   1. Modifies the article functionality and related blocks.
3. mrt\_associate\_guide
   1. Provides the associate guide implementation, blocks, and permission restrictions.
4. mrt\_associates
   1. Largely responsible for the EID integration and related changes.
5. mrt\_breadcrumbs
   1. Controls the breadcrumb functionality present throughout the site.
6. mrt\_collaboration
   1. The core provider for the Set collaboration functionality, including invites, sharing, and report alterations.
7. mrt\_dashboard
   1. The associate and other dashboards.
8. mrt\_inspiration
   1. The Inspiration Gallery functionality including alterations to default Drupal and search functionality for tighter integration.
9. mrt\_linkicons
   1. Modifies certain theme implementations to include additional markup for styling purposes.
10. mrt\_meeting\_purposes
    1. Provides the Meeting Purposes pages and additional meeting purpose functionality on the site.
11. mrt\_pages
    1. Provides static content pages like About Us.
12. mrt\_partners
    1. Provides partner profile functionality and theme implementations.
13. mrt\_pointbreak
    1. Defines site-wide image breakpoints used for rendering images at different sizes based on device dimensions.
14. mrt\_properties
    1. Alterations to the property listing, property themes, featured property blocks, and more.
15. mrt\_property\_import
    1. The property importer that imports the data from EPIC.
16. mrt\_related
    1. Provides the related content blocks that appear in various sections of the website.
17. mrt\_search
    1. Adjustments to the way the search operates on the website including helper functions for use in other modules.
18. mrt\_sets
    1. Functionality for the Set feature available on the site, including adjustments to templates to provide the “add to set” feature.
19. mrt\_sharing
    1. Alters templates to include the share icons in various places on the site.
20. mrt\_slider
    1. Implements the two sliders visible on the homepage.
21. mrt\_tiles
    1. Implements the complex aspects of the tile functionality on the homepage (largely ensuring images are rendered with the correct size when available).
22. mrt\_tracking
    1. Tracks specific events for use in reports.
23. mrt\_user\_files
    1. Implements pages related to user file uploads (My Files, My Uploads) and associated functionality.
24. mrt\_user\_menu
    1. Implements the user dropdown menu.
25. mrt\_user\_registration
    1. Heavily customizes the user registration screen and process.
26. mrt\_users
    1. Heavily customizes the user edit screen and other aspects of working with users.
27. mrt\_xmlsitemap\_file
    1. Allows the xmlsitemap (<https://drupal.org/project/xmlsitemap>) community module to work with files.

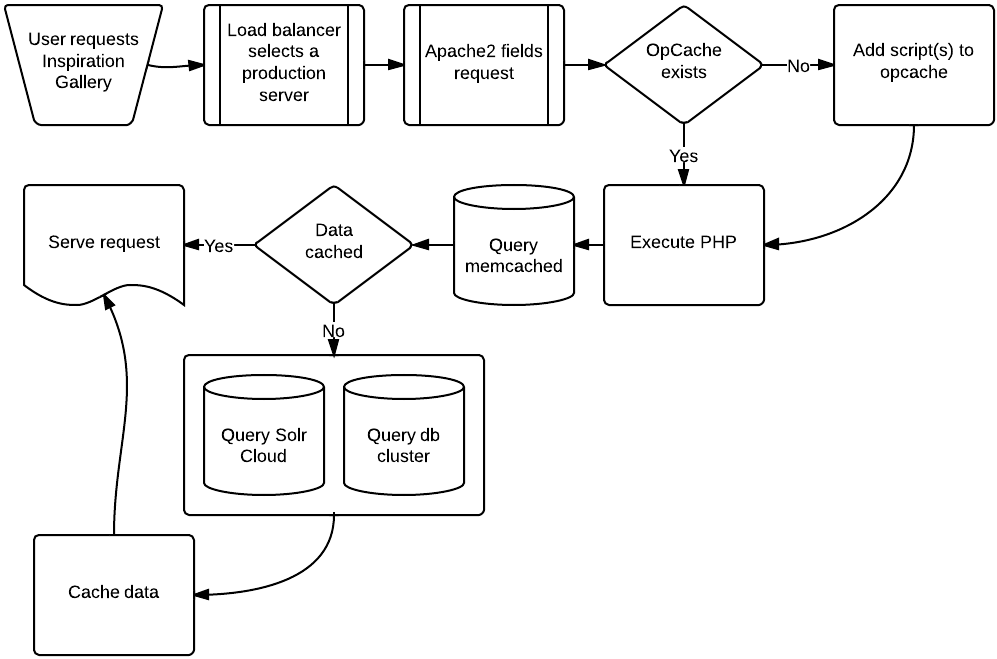
# Property data imports

Marriott provides updated property information through the use of a FTP account. This updated information is in XML format and adheres to the 2004B specification as available from <http://www.opentravel.org/Specifications/ReleaseNotes.aspx?spec=2004B> .

A scheduled job exists that checks for an updated file every week on Sunday. If there are changes available then the modified properties are updated within Drupal. If properties are removed from the XML then they are unpublished from Drupal. If they are added back to the feed then they will be republished in Drupal.

# Typical request workflow

For a user making a request to the Inspiration Gallery (and most other pages) a request will look like the following:



# Acquia Cloud (and Acquia Search)

Acquia Cloud is a scalable, hosted solution specifically designed and tuned for Drupal.

Specifics about Acquia Cloud can be found at <https://www.acquia.com/products-services/acquia-cloud/hosting-services> . More technical information is available through their documentation pages <https://docs.acquia.com/cloud/arch> . Security and compliance information, SLAs, and more can be found on these pages.

## System differences between McMURRY/TMG and Acquia

Acquia uses Percona while McMURRY/TMG uses MariaDB. Both of these are drop-in MySQL replacements tuned for performance and scalability. This change is transparent to the actual website.

Acquia provides a hosted Solr solution called Acquia Search. This solution is optimized for Drupal and scalable. Acquia Search uses Solr 3.5, but this is not a problem. Meetings Imagined is not using any features introduced with Solr 4 or above.

Acquia uses APC instead of Zend OptimizerPlus for its op code caching. This change is transparent to the actual website.

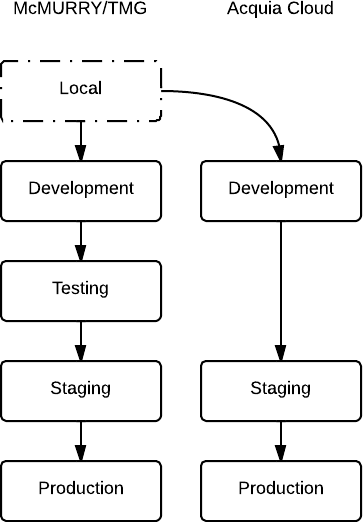
Acquia provides an additional level of caching using Varnish (<https://www.varnish-cache.org>), which minimizes the requests the server has to process.

# Additional planned changes

Integration with Amazon’s CloudFront CDN will be set up to handle serving images to users from localized servers when possible.

# Workflow adjustment

## Current workflow



## New workflow

