

# Features–Based Deployment

**Fox Valley Drupal Meetup Group**



**23 January 2013**

# About Me

Will Long



Kerasai <http://drupal.org/user/1175666>



@Kerasai



<http://www.kerasai.com> (soon, really)

- College degree, ~20 yrs programming, ~10 yrs IT industry, 2.5 yrs full-time Drupal developer
- I've been using this Git & Features thing for a while...

# Objectives

- Basic understanding of the Features module and its functionality
- Technical ability to create and manage Feature modules
- Organize your Feature modules for ease of use
- Capture and deploy changes

# Agenda

- Lecture
  - Tools needed to use with Features
  - Details of the Features module
  - Organizing Feature modules
- Demonstration
  - Create a Feature module
  - Capture and deploy changes

# Tools of the Trade

- Drupal
  - <http://drupal.org/project/drupal>
- Features
  - <http://drupal.org/project/features>
- Drush
  - <http://drupal.org/project/drush>
- Strongarm
  - <http://drupal.org/project/strongarm>
- Diff
  - <http://drupal.org/project/diff>
- Features Utils
  - <http://drupal.org/sandbox/bmoresafety20/1651098>

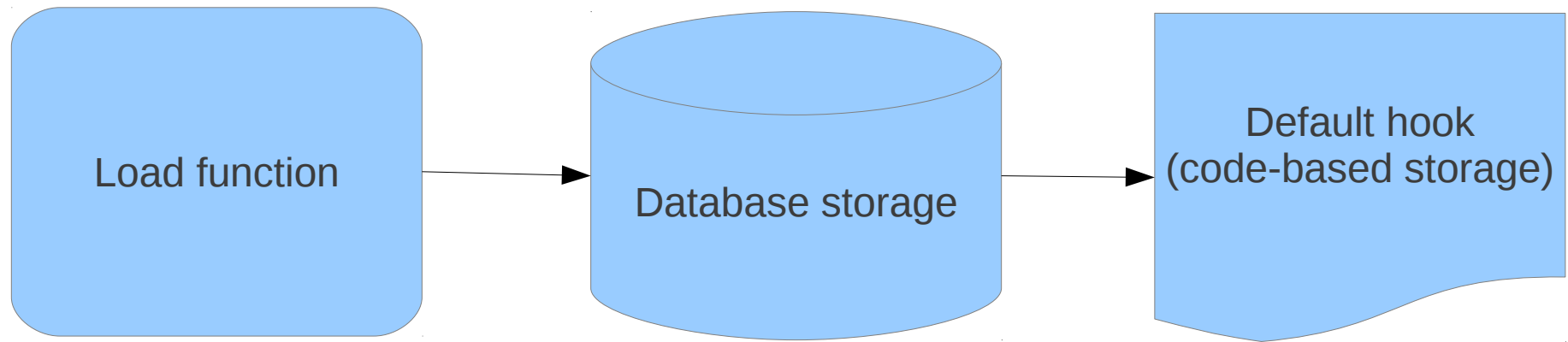
# What Can the Features Module Do?

- Creates Features modules
- Monitors the state of components
- Set/reset the state of components

# Features Components

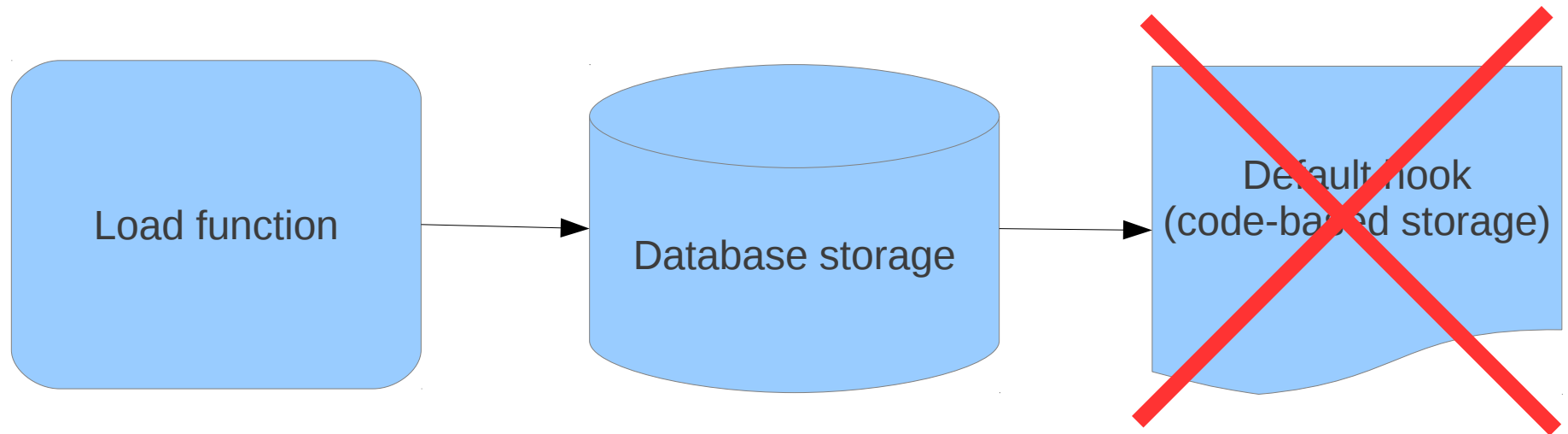
- A component is something that can be exported to code
  - Views
  - Contexts
  - Variables (variable\_set/variable\_get)
  - Image presets
  - etc
- Modules that provide something worth feature-izing should also provide a way to export it
- Additional modules exist to provide feature-izing to modules that are not Features integrated

# Exportables



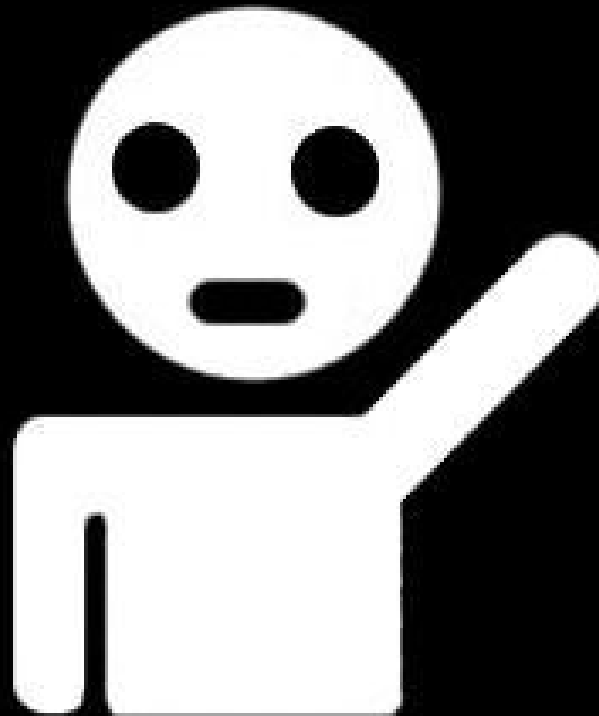


# Faux-Exportables



# On to the Good Stuff

This class is boring...



can we learn about dragons?

# "Sanctioned" Features Usage

## Blog Feature Module

Node Type: Blog

Field: Body

Field: Thumbnail

Field: Tags

Dependencies:  
Image  
Pathauto  
Taxonomy  
Views

View: blog\_display

Taxonomy (vocabulary): Tags

Image preset: thumbnail, medium

Variable: pathauto\_node\_blog\_pattern

# Now on to the Real World



# Recommended Features Usage

- Develop Feature modules to hold configurations for each project (not shared)
- Use a set of base Feature modules as a starting point\*
- Organize Feature modules so site builders and developers can easily identify which Feature modules do/should contain which components
- Organize Feature modules and/or implement process to avoid collisions on simultaneous changes

\* <http://drupal.org/sandbox/bmoresafety20/1896242>

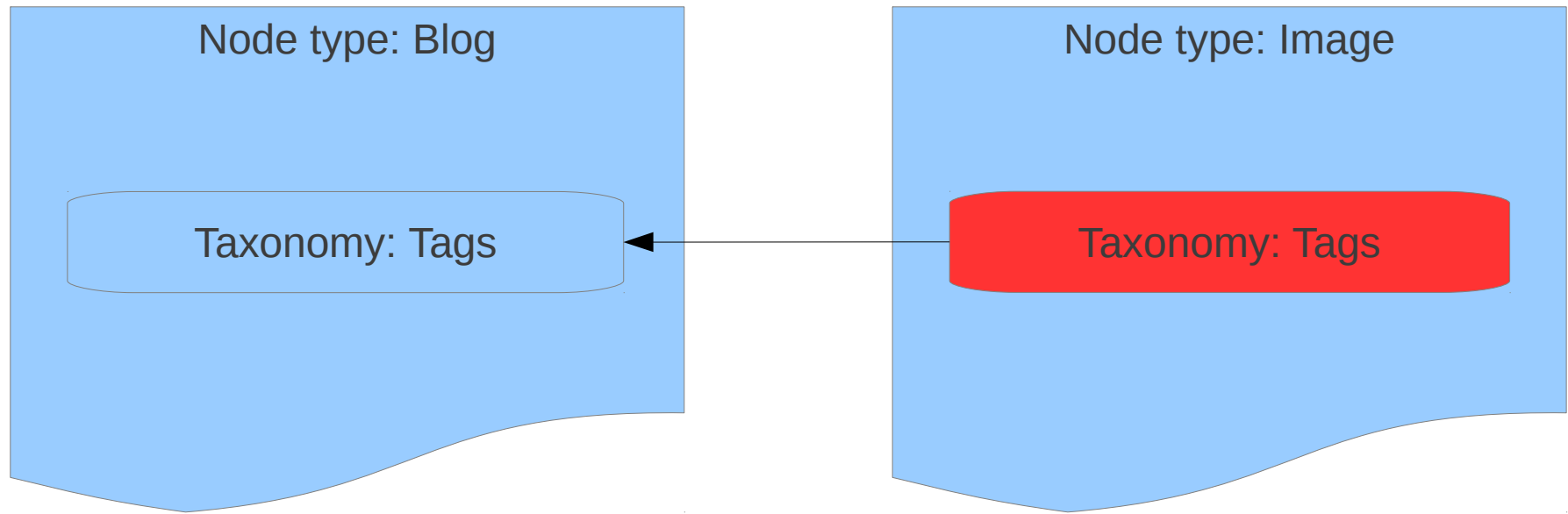
# Organizing Your Feature Modules

- Configuration
  - General site configuration
  - User configuration
  - Theme configuration
  - Media configuration
  - Other configuration by module or related modules
    - Auto Node Title
    - Pathauto
    - XML Site Map
- Content types
  - 1 for each node type
- Natural Exportables
  - Context
  - Taxonomies
  - Views

# Avoiding Dependencies

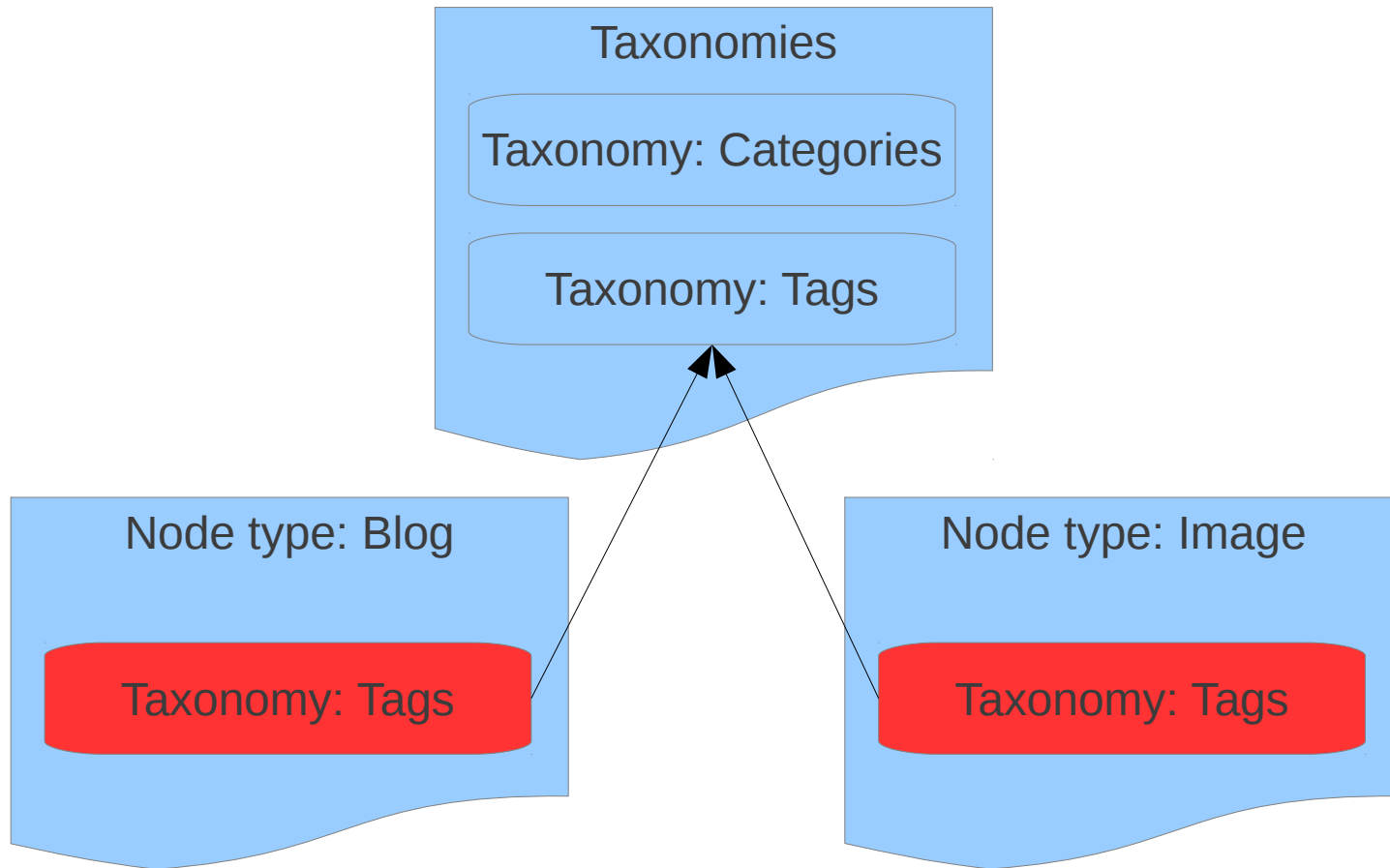
- Dependencies are a good thing
- Dependencies cannot be completely avoided
- Structure Features modules so dependencies are natural
- When you encounter a strange dependency, just rebuild it so it makes sense
- Fix it upfront or wait until it becomes an issue

# Bad Dependency





# Good Dependency



# Demonstration



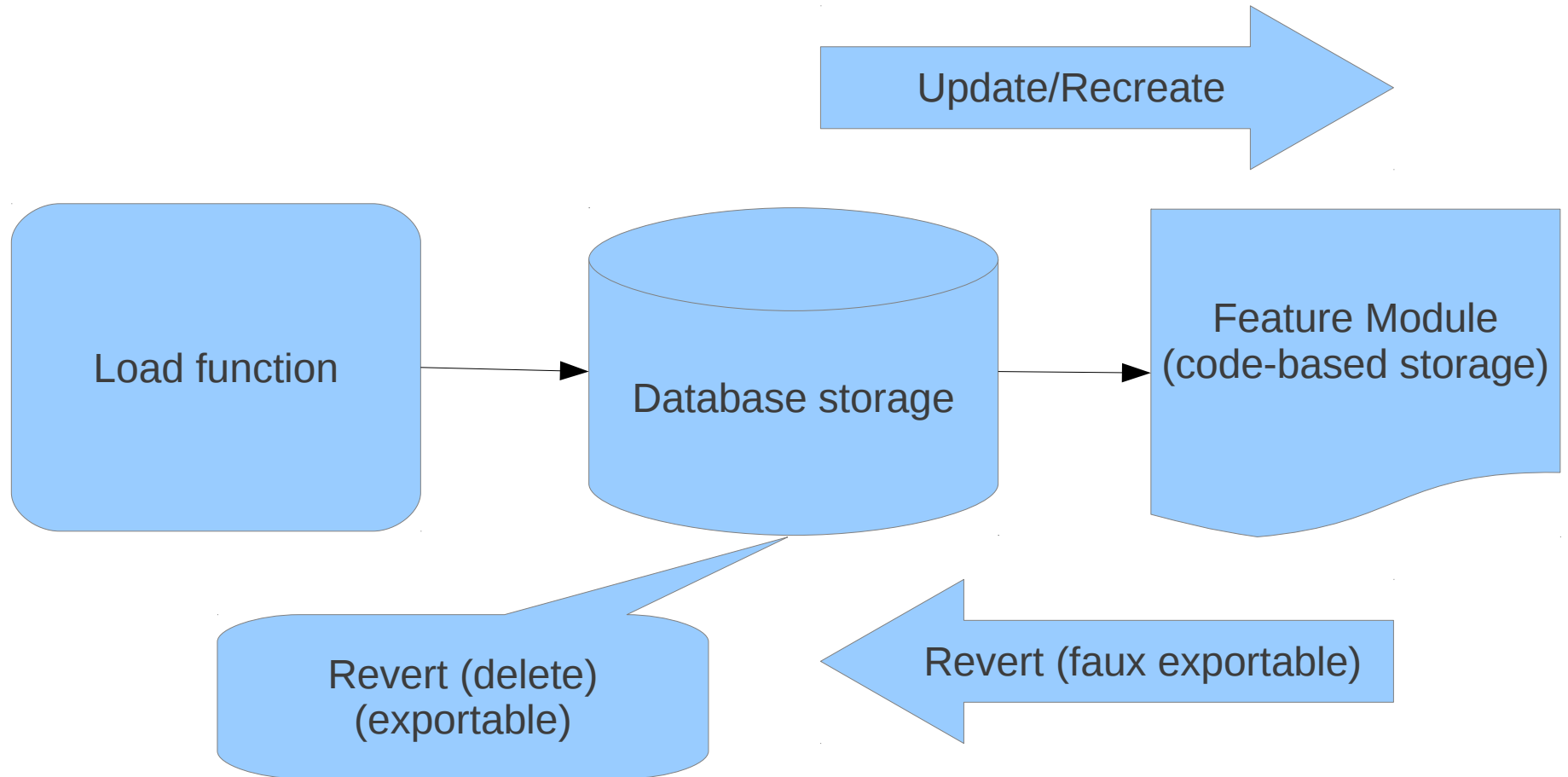
# Summary

- Configuration changes
  - Added a new content type
  - Added a View to display the new content
- Deployed new functionality to production site
  - Enabled new Feature module

# HELP! My Feature is Overridden

- Overridden state means the site's current configuration does not match what's defined in the Feature module's code
- Use the Diff module to analyze differences between the actual configuration and code
- Revert, Update, or Recreate

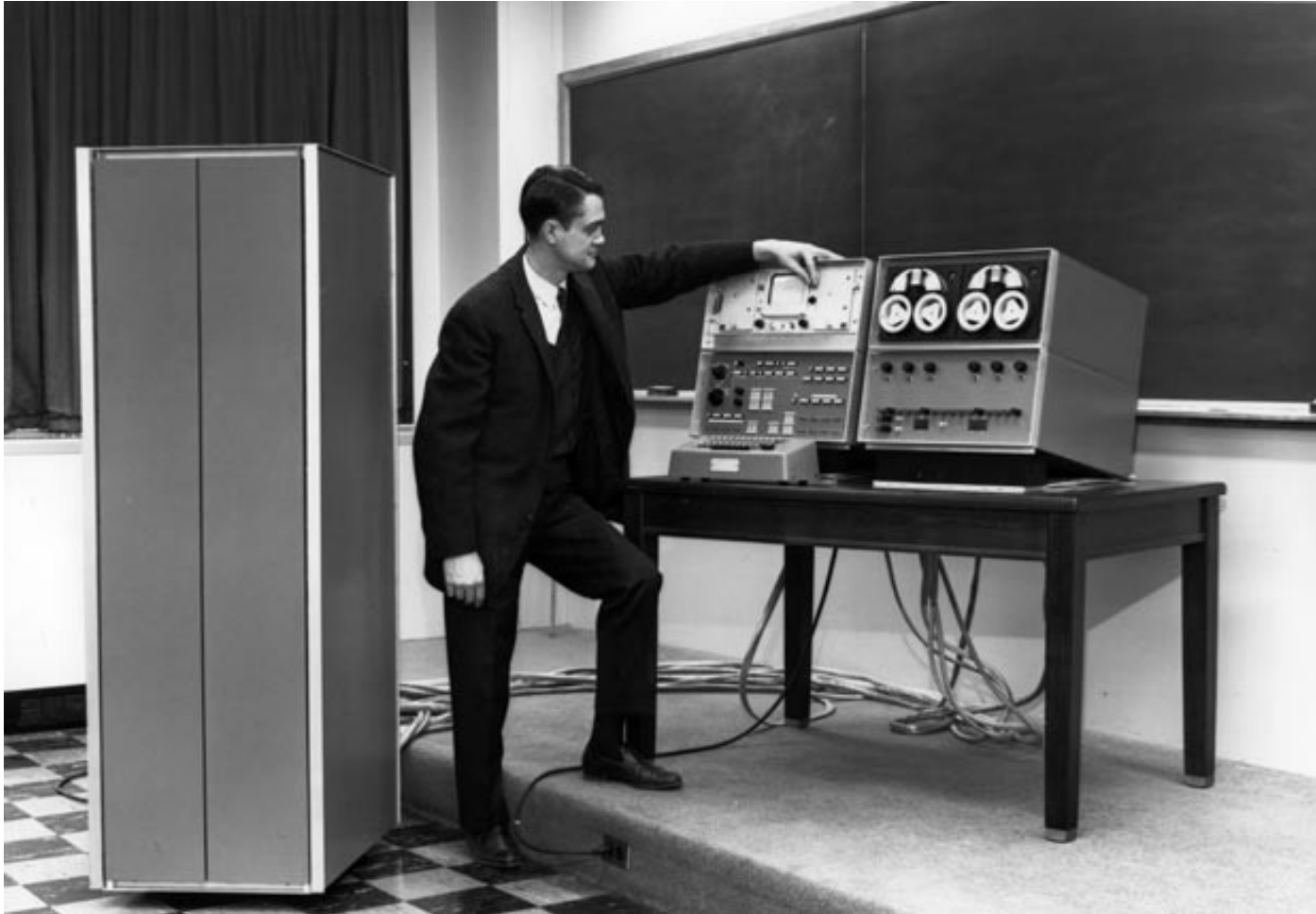
# Updating and Reverting Features



# Drush, You'll Thank Me Later

- See the status of your Feature modules
  - drush fl
- Revert a Feature module
  - drush fr <feature\_module\_name>
- Update a Feature module
  - drush fu <feature\_module\_name>
- Update all Feature modules
  - drush fu-all

# Demonstration



# Summary

- Dev site configuration changes
  - Added Image field to content type
  - Updated View to display Image
  - Set site homepage to the View's path
  - Updated Features module
- Deployed changes to production site
  - Reverted Features modules



# Tips

- Keep Feature modules at “Default” state
- Start small, Feature-ize changes when needed
- Organize Feature modules by Package (tabs)
- Organize Feature modules files in specific site directories, i.e. not sites/all/modules
- Manually adjust Feature module code
- Appoint a Features Czar for each project

# Questions?

Email: [will@kerasai.com](mailto:will@kerasai.com)

Twitter: @Kerasai