

Extending WebGUI

- Patrick Donelan,
SDH Consulting Group



SDH
CONSULTING
GROUP

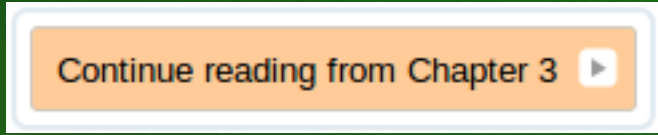
This Talk Is About..

- Adding tentacles to the Gooney
 - To help you get your job done
 - On your own terms
 - And fun ways to play with octopodes
- With examples

Traditional Approach

- The WebGUI API rocks
- Build on top of an API that was designed to be extended
 - Macro
 - Wobject
 - Workflow
 - Account Plugin
 - WebGUI::Form
 - WebGUI::Crud
 - etc..

Mining The WebGUI API

- Example:
 - Say you are building a book reading website and you want returning users to see:


The image shows a rectangular button with a light orange background and a white border. Inside the button, the text 'Continue reading from Chapter 3' is written in a dark font, followed by a small white play button icon on the right.
 - e.g. a Macro that figure out what **PageLayout** under a certain **book/** root the user last viewed

Passive Analytics



- WebGUI module for tracking and analysing user activity
- Available since WebGUI v7.7.0
- Keeps user asset requests in the **passiveLog** table
- Macro can return most recent sub-page by querying the passiveLog table

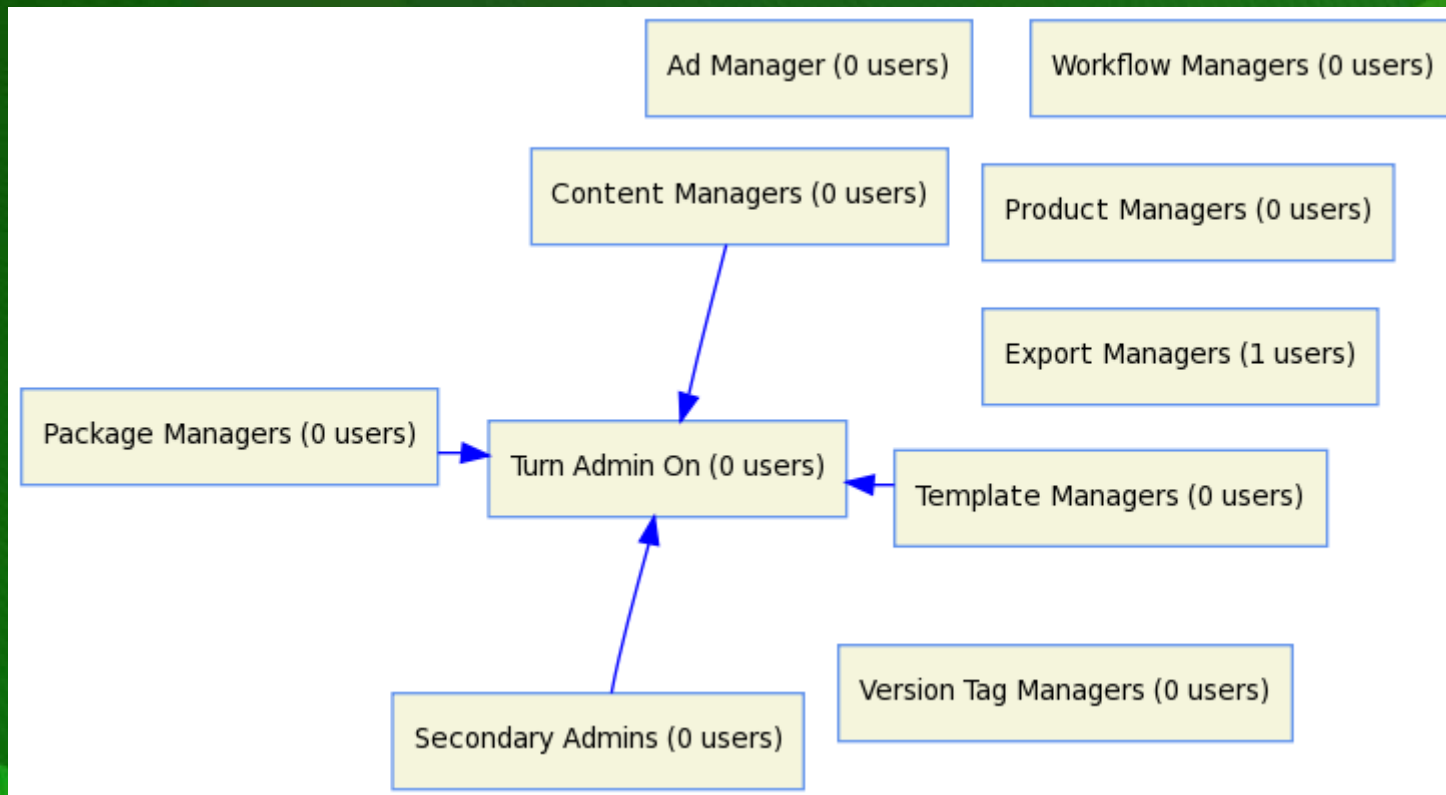
The CPAN is Bigger

The CPAN is Bigger

- Group Visualisation

http://github.com/pdonelan/webgui/tree/group_viz

- Uses **GraphViz** to graph Group groupings



Custom Modules

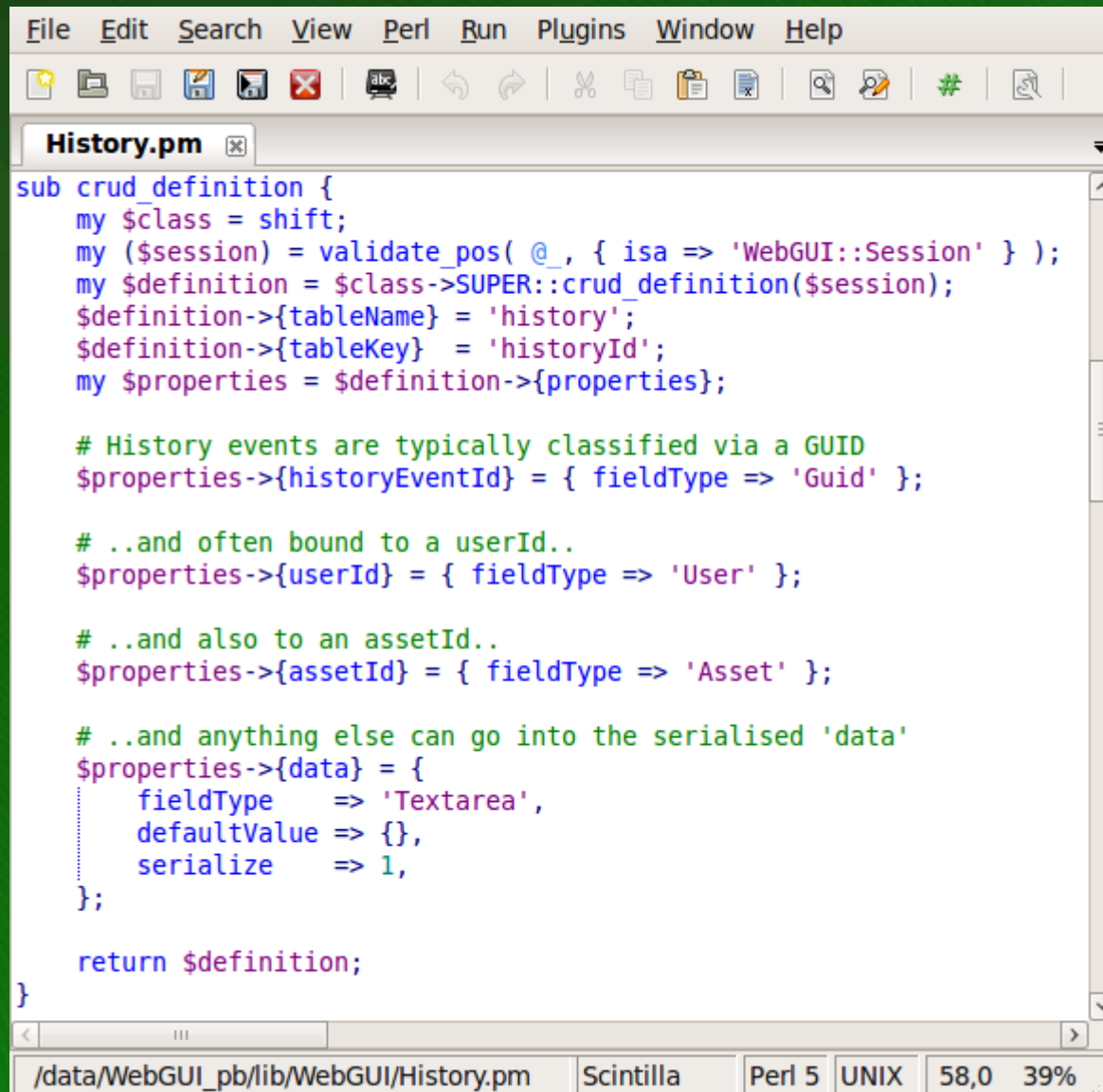
- We're developers
- We like building new stuff
- So let's build a custom module

WebGUI::History

<http://github.com/pdonelan/webgui/tree/history>

- A generic History framework for recording significant user events, e.g.
 - "Joined group X"
 - "Departed group X"
 - "Completed Survey X"
 - "Paid \$49 via PayPal"
 - "Interacted with Thingy Y"

Extends WebGUI::Crud



```
File Edit Search View Perl Run Plugins Window Help
History.pm
sub crud_definition {
    my $class = shift;
    my ($session) = validate_pos( @_, { isa => 'WebGUI::Session' } );
    my $definition = $class->SUPER::crud_definition($session);
    $definition->{tableName} = 'history';
    $definition->{tableKey} = 'historyId';
    my $properties = $definition->{properties};

    # History events are typically classified via a GUID
    $properties->{historyEventId} = { fieldType => 'Guid' };

    # ..and often bound to a userId..
    $properties->{userId} = { fieldType => 'User' };

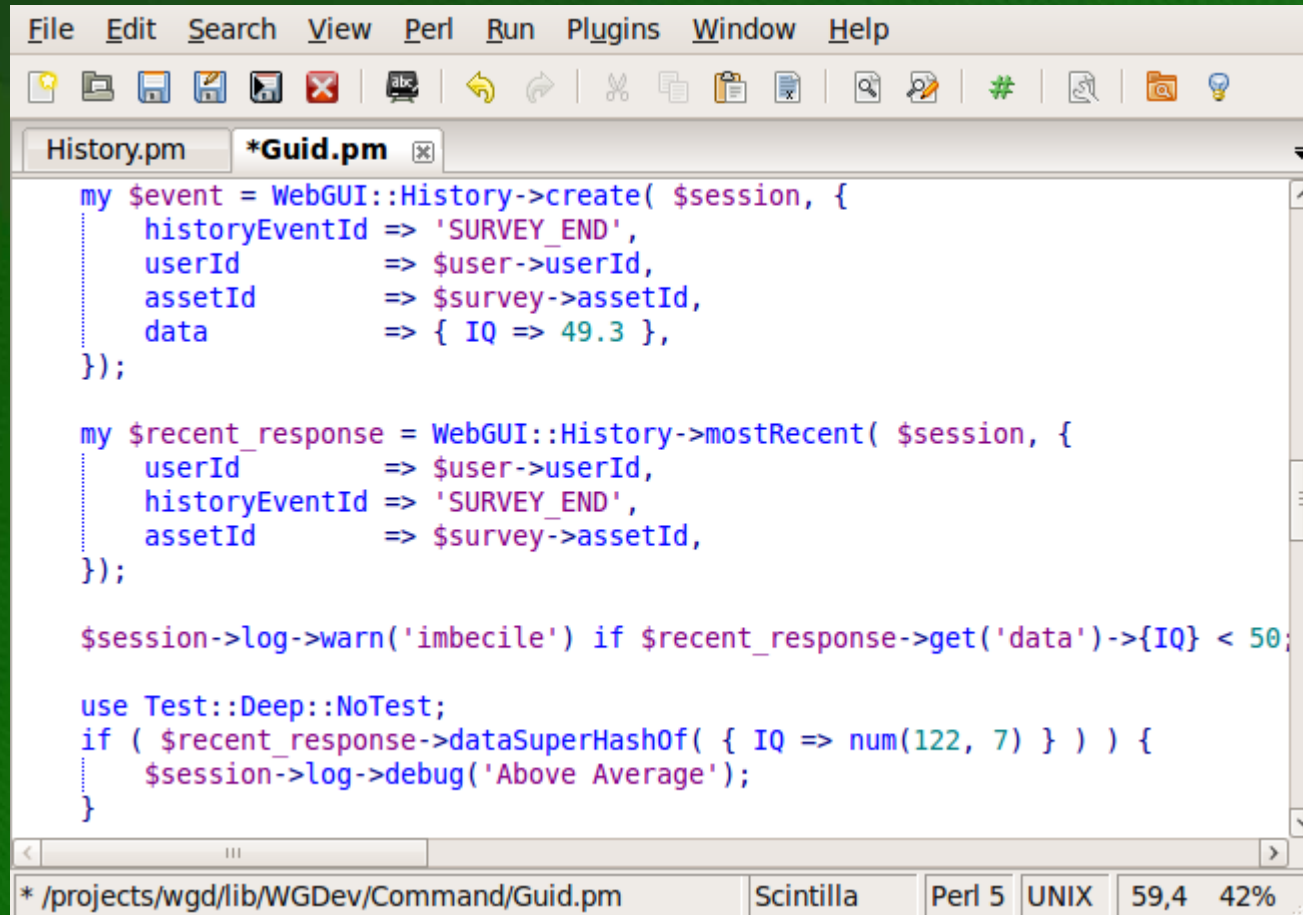
    # ..and also to an assetId..
    $properties->{assetId} = { fieldType => 'Asset' };

    # ..and anything else can go into the serialised 'data'
    $properties->{data} = {
        fieldType => 'Textarea',
        defaultValue => {},
        serialize => 1,
    };

    return $definition;
}
```

/data/WebGUI_pb/lib/WebGUI/History.pm Scintilla Perl 5 UNIX 58,0 39%

Using WebGUI::History



The image shows a screenshot of a Perl IDE window titled "Guid.pm". The code defines a history event and checks for a warning based on IQ values.

```
File Edit Search View Perl Run Plugins Window Help
History.pm *Guid.pm
my $event = WebGUI::History->create( $session, {
    historyEventId => 'SURVEY_END',
    userId         => $user->userId,
    assetId        => $survey->assetId,
    data           => { IQ => 49.3 },
});

my $recent_response = WebGUI::History->mostRecent( $session, {
    userId         => $user->userId,
    historyEventId => 'SURVEY_END',
    assetId        => $survey->assetId,
});

$session->log->warn('imbecile') if $recent_response->get('data')->{IQ} < 50;

use Test::Deep::NoTest;
if ( $recent_response->dataSuperHashOf( { IQ => num(122, 7) } ) ) {
    $session->log->debug('Above Average');
}
```

*/projects/wgd/lib/WGDev/Command/Guid.pm Scintilla Perl 5 UNIX 59,4 42%

RecordSurveyEndInHistory

```
File Edit Search View Perl Run Plugins Window Help
History.pm *RecordSurveyEndInHistory.pm
sub execute {
  my ( $self, $survey, $instance ) = @_;
  my $responseId = $survey->responseId;
  my $userId = $self->session->db->quickScalar(
    'select userId from Survey_response where Survey_responseId = ?', [ $responseId ]
  );

  WebGUI::History->create(
    $self->session,
    {
      historyEventId => 'SURVEY_END',
      userId         => $userId,
      assetId        => $survey->getId,
      data           => { responseId => $responseId }
    }
  );

  return $self->COMPLETE;
}
* /projects/WebFlow/lib/WebGUI/Workflow/Activity/RecordSurveyE Scintilla Perl 5 UNIX 2,4 5%
```

Time Limit:	<input type="text" value="0"/>
Do After Time Limit:	<input type="text" value="Exit URL"/>
Survey End Workflow	<input type="text" value="Record Survey End In History"/> <input type="button" value="Manage"/>
Allow back button	<input checked="" type="radio"/> Yes <input type="radio"/> No

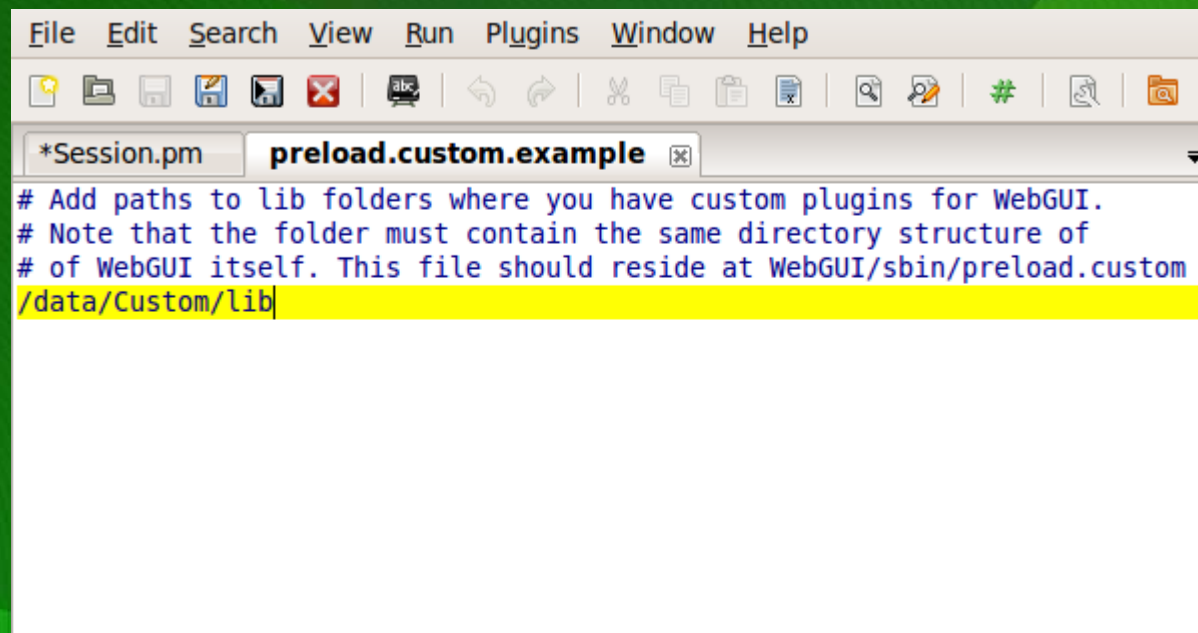
Generic History Reporting

- Account Plugins and Macros can further extend your custom module

Date Paid	Program	Notes	History	message	assign	delete
User History for user pat@patspam.com						
Date	Event Name	Related Asset	More Info			
2009-08-25 00:23:11	Joined comic book appreciation society	Comic Books Online				
2009-08-24 23:39:00	Completed Survey	Superhero or Villain?	View Responses View Feedback	fast forward	message	
2009-08-24 23:14:06	Completed Survey	What Would Batman Do?	View Responses View Feedback			
<input type="button" value="OK"/>						
am.com	gad (sh)	2009-08-23	2009-11-16	2		

Simple Deployment

- WebGUI::History only contains new files
- So deployment is easy:
 - Drop files into `/data/WebGUI/lib`
 - Or add custom dir to `preload.custom`



```
File Edit Search View Run Plugins Window Help
*Session.pm preload.custom.example
# Add paths to lib folders where you have custom plugins for WebGUI.
# Note that the folder must contain the same directory structure of
# of WebGUI itself. This file should reside at WebGUI/sbin/preload.custom
/data/Custom/lib
```

Modifying Core Files

- What if we want to modify core files?
 - Submit an RFE!
 - Core inclusion has many benefits
 - But...

Dealing with Rejection

- Sometimes your RFEs get turned down
 - Example: *RFE #9128 – Thingy pretext and subtext fields maxlength too small*

Thingy pretext and subtext fields maxlength too small (#9128)

Issue

Can we get the maxlength of pre-text and sub-text increased from 255 to something in the range of 1023? 255 characters is somewhat limiting when you're trying to put descriptive text into these fields.

Change required is essentially:

- add "-maxlength=>'1024'," to pretext and subtext field definition in getEditFieldForm()
- change size of pretext and subtext fields in Thingy_fields table in db upgrade script

Cheers,

Patrick

Comments

JT

n/a

16-December-2008 3:27 am

Only if you change them to HTMLAreas. HTMLAreas should be mediumtext. If they're just supposed to be simple instructional text, which is what you described to me when I allowed you to add them in the first place, then they should remain the length they are.

Details

Ticket Status	Pending
Rating	n/a
Submitted By	patspam
Date Submitted	2008-11-21
Assigned To	unassigned
Date Assigned	2009-09-03
Assigned By	
What to improve?	WebGUI Beta
URL	rfe/request-for-

Karma

Difficulty	1
Karma So Far	0
Karma Rank	0.00
Transfer Karma	<input type="text"/>

Keywords

Dealing with scheduling

- Sometimes the **feature freeze** schedule doesn't suit you
 - Example: *RFE #10741 – Account Modules over SSL/HTTPS*

```
File Edit View Terminal Help
commit 3448f0c43c0c12523d81fdc648753bf1e270c782
Author: Patrick Donelan <pat@patspam.com>
Date: Mon Aug 10 14:25:46 2009 +1000

RFE #10741 - Account Modules over SSL/HTTPS

It is useful to be able to turn on SSL/HTTPS for Account Modules
in the same way that you can toggle "Encrypt Page" on a per-Asset
basis.

For example, the User account module that has the change password
screen is one page that you probably want to serve over SSL.

This commit uses a new per-account module "ssl" option in the site
config. When enabled, it causes the account module to force
delivery of the page over HTTPS (basically the same way Asset does
it).
```

I want it yesterday

RFE #10659 – Template Usage

http://github.com/pdonelan/webgui/tree/template_usage

Show Instance Usage For

Replace Everywhere With

Instance Usage

The following Assets are currently using the [Style 03](#) template (which belongs to the [style](#) namespace):

- [Home \(Style Template\)](#)
- [Getting Started \(Style Template\)](#)
- [What should you do next? \(Style Template\)](#)
- [The Latest News \(Style Template\)](#)
- [Tell A Friend \(Style Template\)](#)
- [Site Map \(Style Template\)](#)
- [Documentation \(Style Template\)](#)
- [DataTable \(Style Template\)](#)

Going it alone

- In reality core inclusion is **NOT** always desirable
 - Increased design/implementation cost
 - Niche features
 - Someone else's IP
 - Secret code
 - Silly/fun mods
 - Or you might just not feel like sharing today (that's ok too)

Silly/fun Mod Example #1

```
package WebGUI::User;
```

```
...
```

```
sub new {
```

```
...
```

```
    $self->session->log->debug("Hello, World!");
```

```
}
```

<nothing>

log.conf:

```
log4perl.logger = ERROR, mainlog
```

```
log4perl.logger = DEBUG, mainlog
```

```
2009/09/03 14:33:09 - DEBUG - dev.localhost.localdomain.conf -  
WebGUI::User::new[1064] - Hello, world!
```

WebGUI Logging is not fun

- Turning on DEBUG produces a lot of noise
- WebGUI uses `sitename` as the Log4Perl category
- Options:
 - Use WARN for debugging
 - Grep the logfile
 - Use `Padre::Plugin::WebGUI`



WebGUI.pm Devel.pm Wx.p

```

Wx::Event::EVT_MENU( $main,
# Return our plugin with its
return ( $self->plugin_name
}
sub online_resources {
my %RESOURCES = (
'Bug Tracker' => sub {
    Padre::Wx::launch_br
},
'Community Live Support' => sub {
    Padre::Wx::launch_irc( 'irc.freenode.org' => 'we
},
'GitHub - WebGUI' => sub {
    Padre::Wx::launch_browser('http://github.com/pla
},
'GitHub - WGDev' => sub {
    Padre::Wx::launch_browser('http://github.com/haa
},
'Planet WebGUI' => sub {
    Padre::Wx::launch_browser('http://patspam.com/planetwebgui');
},
'RFE Tracker' => sub {
    Padre::Wx::launch_browser('http://webgui.org/rfe');
},
'Stats' => sub {
    Padre::Wx::launch_browser('http://webgui.org/webgui-stats');
}

```

- Plugin Manager
- Plugin List (CPAN)
- Plugin Tools >
- CSS >
- HTML >
- JavaScript >
- Padre Developer Tools >
- PerlCritc >
- PerlTidy >
- WebGUI >

- Reload WebGUI Plugin Shift+Ctrl+R
- wgd >
- WRE Services >
- Show Asset Tree
- Logview
- Online Resources >
- About

WebGUI Log Output

```

DEBUG: Using /tmp/35dID0_xo4 as control file
DEBUG: Watching file: /data/wre/var/logs/modproxy.error.log
DEBUG: Watching file: /data/wre/var/logs/modperl.error.log
DEBUG: Watching file: /data/wre/var/logs/modrewrite.log
DEBUG: Watching file: /data/wre/var/logs/mylog.log
DEBUG: Watching file: /data/wre/var/logs/webgui.log
/data/wre/var/logs/modperl.error.log (Fri Jul 3 17:56:43 2009) [Fri Jul 03 17:56:43 2009] [notice] caught SIGTERM, shutting down
/data/wre/var/logs/modperl.error.log (Fri Jul 3 17:56:43 2009) ap_get_server_banner() not available until httpd/2.2.4 falling back to ap_get_se
/data/wre/var/logs/modperl.error.log (Fri Jul 3 17:56:43 2009) ap_get_server_description() not available until httpd/2.2.4 falling back to ap_g
/data/wre/var/logs/modperl.error.log (Fri Jul 3 17:56:45 2009) ap_get_server_banner() not available until httpd/2.2.4 falling back to ap_get_se
/data/wre/var/logs/modperl.error.log (Fri Jul 3 17:56:45 2009) ap_get_server_description() not available until httpd/2.2.4 falling back to ap_g
/data/wre/var/logs/webgui.log (Fri Jul 3 17:56:46 2009) 2009/07/03 17:56:46 - WARN - dev.localhost.localdomain.conf - WebGUI::User::get[582] -
/data/wre/var/logs/modperl.error.log (Fri Jul 3 17:56:46 2009) [Fri Jul 03 17:56:46 2009] [notice] Apache/2.0.63 (Unix) WebGUI/7.7.11 mod_ssl/

```

Padre is fun

<http://github.com/pdonelan/Padre-Plugin-WebGUI>

- But `Padre::Plugin::WebGUI` is still in alpha
- (patches / pull requests welcome)

In the meantime..

- Wouldn't it be nice if we could tell Log4Perl what module we are interested in?

Log4Perl::Filter::CallerMatch

<http://github.com/pdonelan/log4perl-filter-callermatch>

```
log4perl.logger = ALL, mainlog, mylog

log4perl.appender.mainlog = Log::Log4perl::Appender::File
log4perl.appender.mainlog.Filter = LevelRange
log4perl.appender.mainlog.filename = /data/wre/var/logs/webgui.log
log4perl.appender.mainlog.layout = PatternLayout
log4perl.appender.mainlog.layout.ConversionPattern = %d - %p - %c - %M[%L] - %m%n

log4perl.appender.mylog = Log::Log4perl::Appender::File
log4perl.appender.mylog.Filter = CallerMatch
log4perl.appender.mylog.filename = /data/wre/var/logs/mylog.log
log4perl.appender.mylog.layout = PatternLayout
log4perl.appender.mylog.layout.ConversionPattern = %d - %p - %c - %M[%L] - %m%n

log4perl.filter.LevelRange = Log::Log4perl::Filter::LevelRange
log4perl.filter.LevelRange.LevelMin = WARN
log4perl.filter.LevelRange.LevelMax = FATAL

log4perl.filter.CallerMatch = Log::Log4perl::Filter::CallerMatch
log4perl.filter.CallerMatch.PackageToMatch = Survey
```

Logging as debugging

```
my %var = ( a => 1, b => 2 );  
$session->log->warn( Data::Dumper::Dumper(%var) );  
$VAR1 = 'a';
```

```
$session->d( %var );  
$ARG1 = 'a';  
$ARG2 = 1;  
$ARG3 = 'b';  
$ARG4 = 2;
```

```
$session->d( \%var );  
$ARG1 = {  
    'a' => 1,  
    'b' => 2  
};
```

\$session->d()

```
use Data::Dumper;
sub d {
    my $self = shift;
    local $Data::Dumper::Sortkeys = 1;
    local $Data::Dumper::Varname = 'ARG';
    $self->log->warn(
        scalar Data::Dumper::Dumper(@_)
    );
}
```

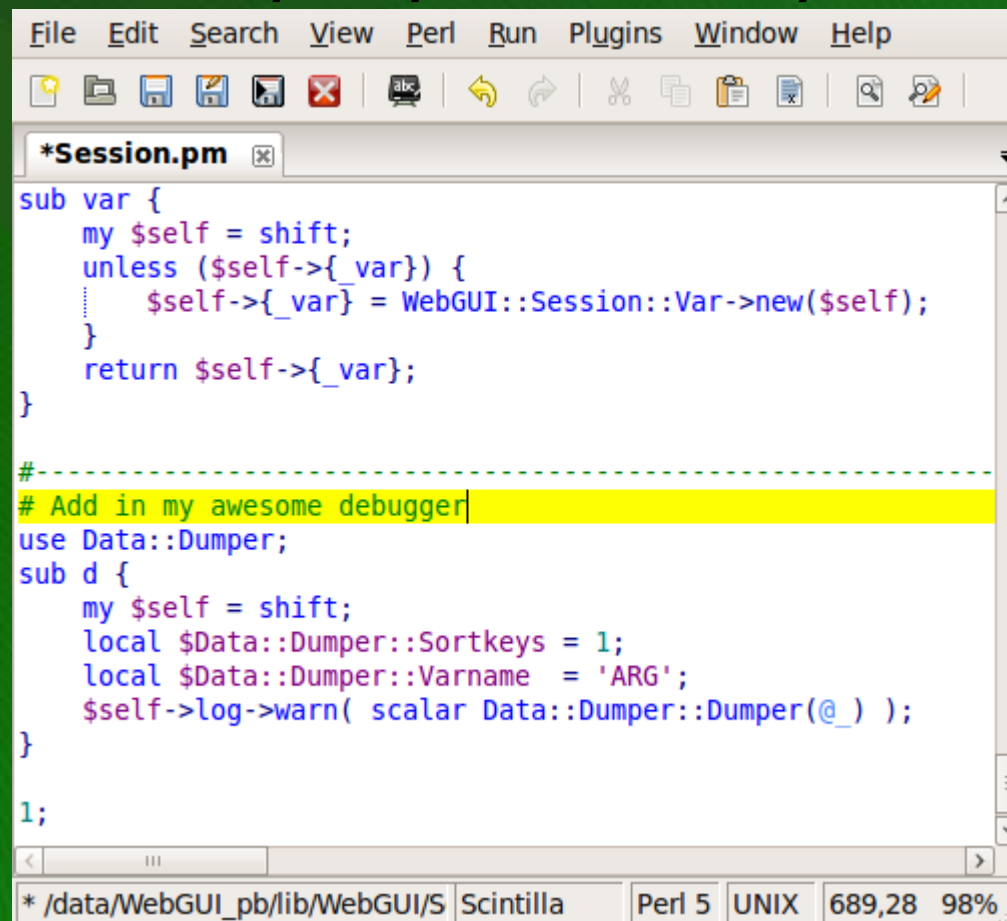
- `ack 'session->d\b'`

Modifying The Core

- Now that we've decided to modify the core, how should we proceed?

Approach #1

- Edit
/data/WebGUI/lib/WebGUI/Session.pm
in-place



```
File Edit Search View Perl Run Plugins Window Help
*Session.pm
sub var {
    my $self = shift;
    unless ($self->{_var}) {
        $self->{_var} = WebGUI::Session::Var->new($self);
    }
    return $self->{_var};
}

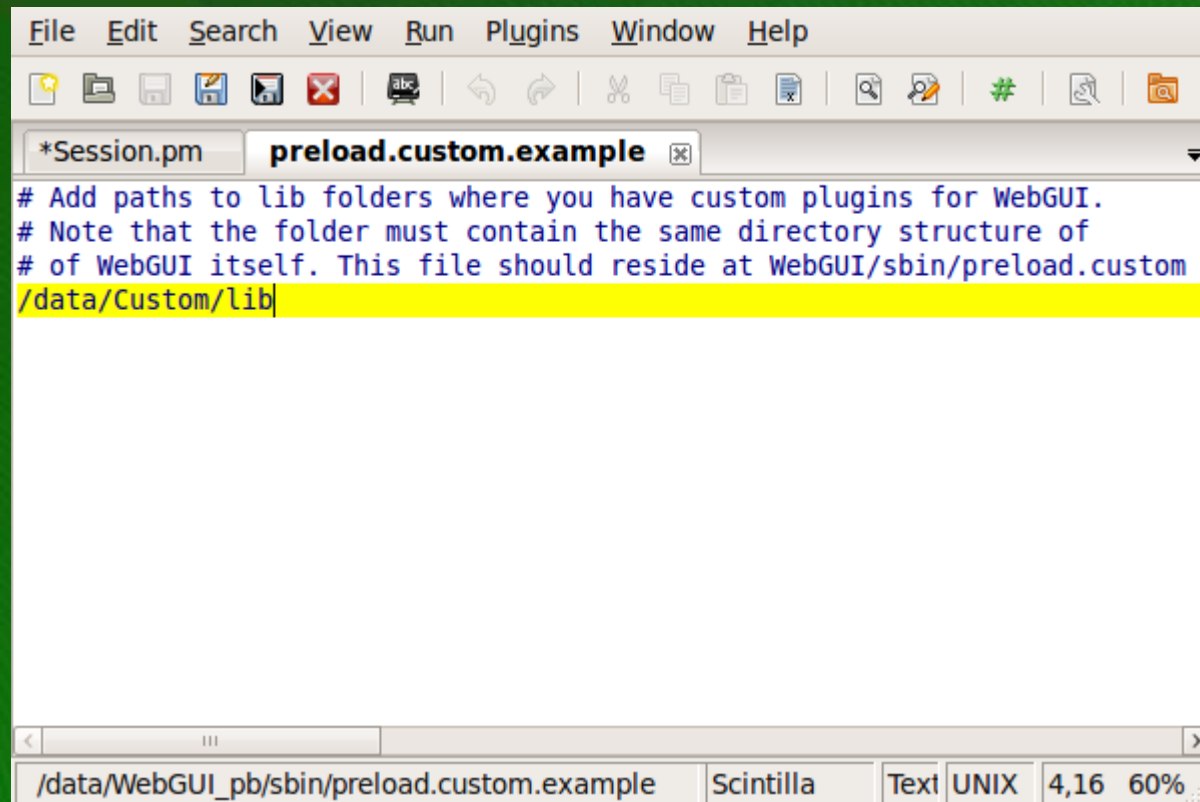
#-----
# Add in my awesome debugger
use Data::Dumper;
sub d {
    my $self = shift;
    local $Data::Dumper::Sortkeys = 1;
    local $Data::Dumper::Varname = 'ARG';
    $self->log->warn( scalar Data::Dumper::Dumper(@_) );
}

1;
```

* /data/WebGUI_pb/lib/WebGUI/S Scintilla Perl 5 UNIX 689,28 98%

Approach #2

- Copy Session.pm to custom directory
- Add custom dir to **preload.custom**



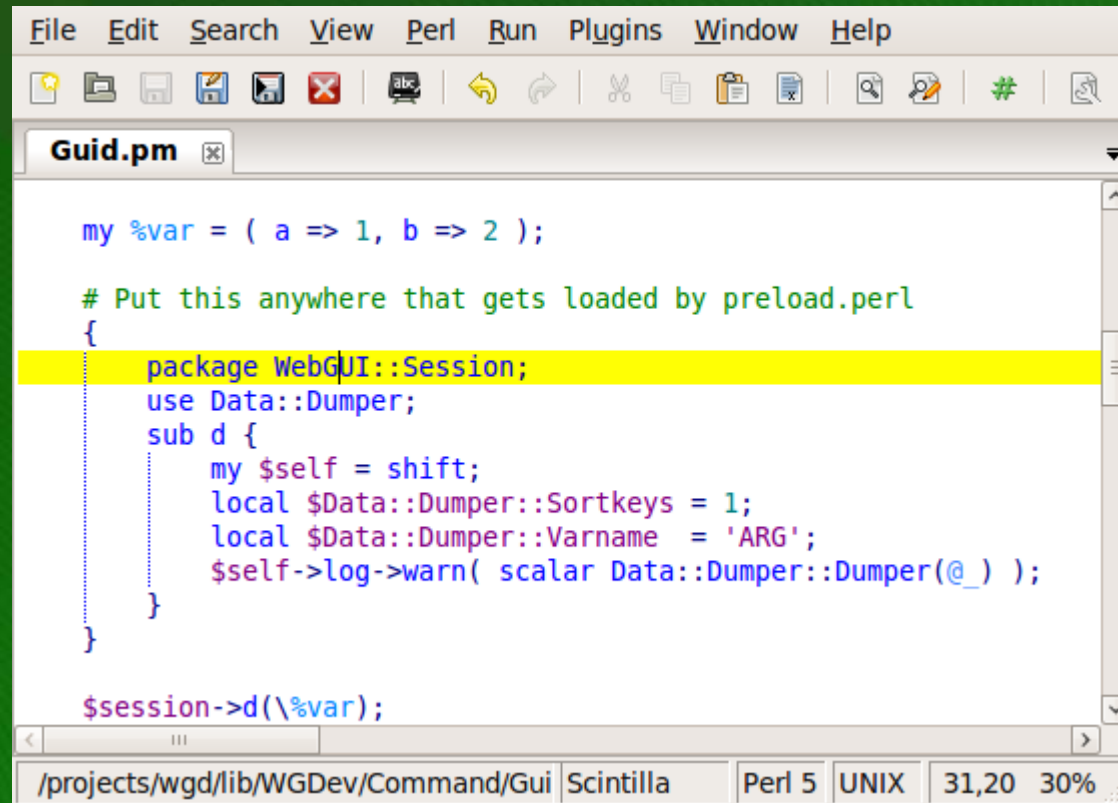
The screenshot shows a code editor window with the following content:

```
File Edit Search View Run Plugins Window Help
# Add paths to lib folders where you have custom plugins for WebGUI.
# Note that the folder must contain the same directory structure of
# of WebGUI itself. This file should reside at WebGUI/sbin/preload.custom
/data/Custom/lib
```

The status bar at the bottom of the window displays: /data/WebGUI_pb/sbin/preload.custom.example Scintilla Text UNIX 4,16 60%

Approach #3

- Use Perl's **package** keyword



The screenshot shows a Perl IDE window titled "Guid.pm". The code in the editor is as follows:

```
File Edit Search View Perl Run Plugins Window Help
my %var = ( a => 1, b => 2 );

# Put this anywhere that gets loaded by preload.perl
{
    package WebGUI::Session;
    use Data::Dumper;
    sub d {
        my $self = shift;
        local $Data::Dumper::Sortkeys = 1;
        local $Data::Dumper::Varname = 'ARG';
        $self->log->warn( scalar Data::Dumper::Dumper(@_) );
    }
}

$session->d(\%var);
```

The line `package WebGUI::Session;` is highlighted in yellow. The status bar at the bottom shows the file path `/projects/wgd/lib/WGDev/Command/Gui`, the editor `Scintilla`, and system information `Perl 5 UNIX 31,20 30%`.

User Stringification

```
$session->log->debug( "Processed $u" );  
Processed WebGUI::User=HASH(0xa7a0710)  
Processed patspam  
Processed WebGUI::User(boO-MGbV459vpvMLyIJIew)
```

```
$session->log->warn("Processed @{"  
  $u->username ? $u->username : 'WebGUI::User(' . $u->userId . '  
}");
```

```
{  
  package WebGUI::User;  
  use overload q("") => sub {  
    my $self = shift;  
    my ( $userId, $username ) = ( $self->userId, $self->username );  
    return $username if $username;  
    return "WebGUI::User($userId)";  
  };  
}
```

WebGUI::Group

```
{
  package WebGUI::Group;

  # Override WebGUI::Group stringification
  use overload q("") => sub {
    my $self = shift;
    my ( $groupId, $name ) = ( $self->getId, $self->name );
    return $name if $name;
    return "WebGUI::Group($groupId)";
  };

  # Same as L<WebGUI::Group::getUsers>, except that disabled users are filtered out
  sub get_enabled_users {
    my $self = shift;
    my @users = $self->session->db->buildArray(<<END_SQL, [$self->getId]);
select g.userId
from groupings g, users u
where g.userId = u.userId
  and groupId = ?
  and u.status = "Active"
END_SQL
    return \@users;
  }
}
```

More WebGUI::User Fun

```
sub remove_friends {
    my $self = shift;
    my $friends = WebGUI::Friends->new( $self->session, $self );
    $friends->delete( $self->friends->getUsers );
}

sub reset_notification_settings {
    my $self = shift;
    $self->profileField( 'cellPhone', '' );
    $self->profileField( 'receiveInboxSmsNotifications', '' );
    $self->profileField( 'receiveInboxEmailNotifications', 1 );
}

# Hooked into L<WebGUI::Asset::Account::Inbox>
sub can_send_messages {
    my $self = shift;
    return 1 if $self->isInGroup(NETC::GROUP_RESEARCHERS);
    return 1 if $self->isInGroup(NETC::GROUP_MANAGERS);
    return 1 if $self->isInGroup(NETC::GROUP_THERAPISTS);
    return 1 if $self->isInGroup(NETC::GROUP_CONSUMERS) && $self->get_therapist;
    return;
}
```

can_send_messages

```
File Edit View Terminal Help
Author: Patrick Donelan <pat@patspam.com>
Date: Sun Aug 23 14:11:43 2009 +1000

Override who can send messages in Inbox.pm

Calls our $user->can send messages rather than $user->hasFriends

diff --git a/lib/WebGUI/Account/Inbox.pm b/lib/WebGUI/Account/Inbox.pm
index b49af19..3bbf644 100644
--- a/lib/WebGUI/Account/Inbox.pm
+++ b/lib/WebGUI/Account/Inbox.pm
@@ -1511,7 +1511,7 @@ sub www_view {
    $var->{'has_messages' } = $msgCount > 0;
    $var->{'message_total' } = $msgCount;
    $var->{'new_message_url' } = $self->getUrl("module=inbox;do=sendMessage");
-   $var->{'canSendMessages' } = $user->hasFriends;
+   $var->{'canSendMessages' } = $user->can send messages;

    tie my %rpps, "Tie::IxHash";
    %rpps = (25 => "25", 50 => "50", 100=>"100");
@@ -1720,7 +1720,7 @@ sub www_viewMessage {
    && !$sentByVisitor
    && $status ne "pending"
    && $status ne "completed" ) {
-   $var->{'canReply' } = "true";
+   $var->{'canReply' } = $user->can send messages;
    $var->{'reply_url' } = $self->getUrl("module=inbox;do=sendMessage;messageId=" . $messageId);
}
:
```

Topic Branches

- Monkey patching is fun
- But sooner or later you need cages for your monkeys
- Git is your friend
- Now that WebGUI uses Git, creating a new topic branch is as easy as:

```
File Edit View Terminal Help
patspam@pattop:~/data/WebGUI (master)$ git checkout -b crypt master
Switched to a new branch "crypt"
patspam@pattop:~/data/WebGUI (crypt)$
```

WebGUI::Crypt

<http://github.com/pdonelan/webgui/tree/crypt>

- Adds a pluggable database encryption layer to WebGUI
- Modifies core files and introduces new perl modules, tests, dependencies, etc..
- Lives in 2 topic branches on github



WebGUI::Crypt

- You can diff files against other branches

```
File Edit View Terminal Help
patspam@pattop:/data/WebGUI (crypt)$ git diff master -- lib/WebGUI/Inbox/Message.pm
diff --git a/lib/WebGUI/Inbox/Message.pm b/lib/WebGUI/Inbox/Message.pm
index 642aed2..62f659e 100644
--- a/lib/WebGUI/Inbox/Message.pm
+++ b/lib/WebGUI/Inbox/Message.pm
@@ -141,7 +141,7 @@ sub create {
    $self->{_properties}{messageId} = "new";
    $self->{_properties}{status}     = $properties->{status} || "pending";
    $self->{_properties}{subject}    = $properties->{subject} || $!18n->get(523, 'WebGUI');
-   $self->{_properties}{message}    = $properties->{message};
+   $self->{_properties}{message}    = $session->crypt->encrypt_hex($properties->{message}, {table=>'inbox', fi
    $self->{_properties}{dateStamp}  = time();
    $self->{_properties}{userId}     = $properties->{userId};
    $self->{_properties}{groupId}    = $properties->{groupId};
@@ -163,6 +163,7 @@ sub create {
    }

    $self->{_messageId} = $self->{_properties}{messageId} = $session->db->setRow("inbox", "messageId", $self->{
+   $self->{_properties}{message} = $properties->{message}; # undo encryption for memory/cache
    $self->{_userId} = $self->{_properties}{userId};
    $self->{_inbox} = $self->{_properties};

@@ -431,6 +432,8 @@ sub new {
    my $self = {};

    my %properties = (%{$inbox}, %{$statusValues});
+   $properties{message} = $session->crypt->decrypt_hex($properties{message});
+
    bless {_properties=>\%properties, _inbox=>$inbox, _session=>$session, _messageId=>$messageId, _userId=>$us
}
```

WebGUI::Crypt

- Reviewing changes is easy

```
File Edit View Terminal Help
patspam@pattop:~/data/WebGUI (crypt)$ git diff --stat v7.7.18-tagged-by-patspam
crypt_benchmark.pl | 158 ++++++
lib/WebGUI/Asset/Wobject/Survey.pm | 69 +++-
lib/WebGUI/Content/Crypt.pm | 56 +++
lib/WebGUI/Crypt.pm | 492 ++++++
lib/WebGUI/Crypt/Admin.pm | 280 ++++++
lib/WebGUI/Crypt/Provider/None.pm | 93 ++++
lib/WebGUI/Crypt/Provider/Simple.pm | 142 ++++++
lib/WebGUI/Crypt/Provider/_NewProvider.skeleton | 92 ++++
lib/WebGUI/Exception.pm | 10 +
lib/WebGUI/Form/CryptProvider.pm | 197 ++++++
lib/WebGUI/Inbox/Message.pm | 5 +-
lib/WebGUI/Operation/Settings.pm | 13 +-
lib/WebGUI/Session.pm | 33 ++
.../Workflow/Activity/CryptUpdateFieldProviders.pm | 162 ++++++
lib/WebGUI/i18n/English/Asset_Survey.pm | 8 +
lib/WebGUI/i18n/English/Crypt.pm | 70 +++
lib/WebGUI/i18n/English/WebGUI.pm | 9 +
.../Workflow_Activity_CryptUpdateFieldProviders.pm | 13 +
sbin/installCrypt.pl | 278 ++++++
sbin/testEnvironment.pl | 1 +
t/Asset/Wobject/Survey.t | 38 +-
t/Crypt/Crypt.t | 188 ++++++
t/Crypt/None.t | 62 +++
t/Crypt/OpenSSL.t | 54 +++
t/Crypt/Simple.t | 94 ++++
t/Form/CryptProvider.t | 47 ++
t/Inbox.t | 52 +-
t/Workflow.t | 2 +-
t/Workflow/Activity/CryptUpdateFieldProviders.t | 56 +++
t/lib/WebGUI/CryptTest.pm | 127 ++++++
www/extras/adminConsole/crypt.png | Bin 0 -> 2578 bytes
www/extras/adminConsole/small/crypt.png | Bin 0 -> 687 bytes
32 files changed, 2892 insertions(+), 9 deletions(-)
```

When should I branch?

- Pretty much always
- Including afterwards when you realise you should have done so earlier
- Rewriting history with git is fun

Branch on!

- Create a topic branch
`$ git checkout -b rfe_9128 <wg_release>`
- Apply your changes, create install script and commit
- Create a custom branch for your site
`$ git checkout -b mysite <wg_release>`
- Merge in topic branch
`$ git merge rfe_9128`
- Deploy on server and run install script

Rinse and repeat

- Your custom site branch can contain multiple topic branches
- Topic branches have a life of their own
- Topic branches make collaboration easier
- Good idea to make install scripts idempotent

Staying in touch

- Rebase against new WebGUI releases
- And then re-merge into your custom site branch
- Deploy and upgrade as normal

Perl code on the Web



Eval is ~~evil~~ fun

- Giving trusted users the ability to eval code on the server is immensely powerful
- Even if you only trust yourself
- **Safe.pm** can limit opcodes

Workflow::Activity::Eval

	<input type="button" value="save"/>
Title	<input type="text" value="Eval"/>
Description	<input type="text"/>
Code	<pre>\$self->session->log->warn('eval reporting for duty');</pre>
	<input type="button" value="save"/>

Survey Branching

- The Survey branching language is Perl

Jump expression:

```
# count the number of items that score >= 4 on either scale
my $four_and_up_count = 0;
for my $item ('a..'s') {
  # fear scales are A3a_f, A3b_f, A3c_f, ..
  # avoidance scales are A3a_a, A3b_a, A3c_a, ..
  my $f = "A3${item}_f";
  my $a = "A3${item}_a";
  $four_and_up_count++ if score($f) >= 4;
  $four_and_up_count++ if score($a) >= 4;
}

# Jump to S_A_post if no score of 4 or above on
# any one item on either fear or avoidance scales
jump { $four_and_up_count == 0 } S_A_post;
```

Yes No

Text on every page:
 Yes No

Terminal section:
 Yes No

Terminal section URL:

Logical Section
 Yes No

Survey Tagging

- The Survey tagging language is also Perl

Jump expression:

```
if ( score( section1 ) > 5 ) {  
  tag( CAN_HAZ_CHEEZBURGER, { name => 'lolcat' } )  
} else {  
  tag( SOMETHING_ELSE );  
}
```

Body Mass Index

$$\text{BMI} = \frac{\text{weight (kg)}}{\text{height}^2 \text{ (cm)}}$$

Please enter your weight in kilograms (kg) or pounds (lbs)

Kilograms (kg)

Pounds (lbs)

Please enter your height in centimetres (cm) or inches (in)

Centimetres (cm)


Inches (in)


Interactive BMI

Do you or would you refuse to increase your body weight by 16 kg to what is considered the minimum healthy weight for your height (66 kg)?

Yes

No

 Back

Continue 

Jump expression:

```
if ( value(AN1a) eq 'lbs' ) {
    $weight *= $LBS_TO_KG;
}
tag( AN_WEIGHT, $weight ) if $weight;
my $height = value(AN1b_verbatim);
if ( value(AN1b) eq 'in' ) {
    $height *= $IN_TO_CM;
}
$height /= 100; # convert to metres
tag( AN_HEIGHT, $height ) if $height;

# need a valid height and weight..
if ( $height > 0 && $weight > 0 ) {
    my $bmi = $weight / $height**2;
    tag( AN_BMI, $bmi );
    tag( 'FEEDBACK.WARNING_BMI' ) if $bmi < 16.5;

    if ( $bmi < $BMI_UNDERWEIGHT ) {
        tag(AN_BMI_UNDERWEIGHT_RANGE);
        my $normal_weight = round( $BMI_NORMAL * $height**2 );
        tag( AN_BMI_NORMAL_WEIGHT, $normal_weight );
        tag( AN_BMI_NORMAL_WEIGHT_INCREASE, round(
$normal_weight - $weight ) );
    }
} else {
    if ($weight > 0) {
        tag(AN1a_VALIDATION_MESSAGE, "");
    } else {
        tag(AN1a_VALIDATION_MESSAGE, "Please check you have
entered valid weight value");
    }
    if ($height > 0) {
        tag(AN1b_VALIDATION_MESSAGE, "");
    } else {
        tag(AN1b_VALIDATION_MESSAGE, "Please check you have
entered valid height value");
    }
}
```

Question text:

Font Name and Size

Arial 13

Font Style

B *I* U abc

Lists



Insert Item



Do you or would you refuse to increase your body weight by $[[AN_BMI_NORMAL_WEIGHT_INCREASE]]$ kg to what is considered the minimum healthy weight for your height ($[[AN_BMI_NORMAL_WEIGHT]]$ kg)?

Submit

Copy

Cancel

Delete

Survey Games

[Back to site.](#) | [Edit](#) | [Visualize](#) | [Test Suite](#) | [Take Survey](#)

Survey Objects...

- S1: Init
- S2: Hangman
- Q1: What is this word?**
[[DISPLAY_STATE]]
- You have [[LEFT]] chances left**
- You have already guessed:**
[[DISPLAY_GUESSES]]
- [[MSG]]**
- S3: FAIL
- S4: SUCCESS

[Add Section](#) [Add Question](#) [Add Answer](#)

Please enter question information

Question number:

1

Question variable name:

Q

Question type:

Tagged

Question score:

1

Required

Yes No

Randomize answers:

Yes No

Vertical display:

Yes No

Show text in button:

Yes No

Allow comment:

Yes No

Comment Cols:

10

Comment Rows:

5

Maximum number of

answers:

1

Jump to:

Jump expression:

```
# Retrieve tagged data
my $word      = tagged(WORD);
my $length    = tagged(LENGTH);
my %freq      = %{ tagged(FREQ) };
my @state     = @ { tagged(STATE) };
my $left      = tagged(LEFT);
my %guessed   = %{
tagged(GUESSED) };
my @tagged_answers = @ {
tagged(Q_TAGGED_ANSWERS) };

if ( $left <= 0 ) {
  jump { 1 } FAIL;
}
```

Question text:

Font Name and Size

Arial 13

Font Style

B *I* U abc

Lists

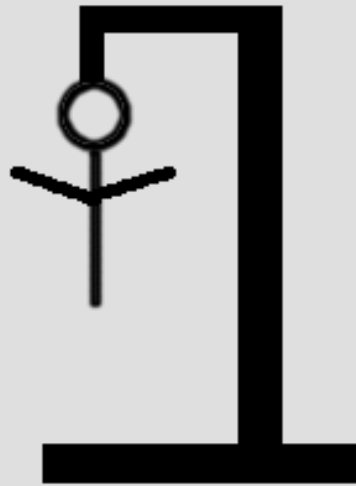
Insert Item

What is this word? **[[DISPLAY_STATE]]**

You have **[[LEFT]]** chances left

You have already guessed: **[[DISPLAY_GUESSES]]**

Hangman



Q: What is this word? **a p p _ e**

You have **4** chances left

You have already guessed: a, b, c, d, e, f, g, h, p

p matched!

i

j

k

l

m

n

o

q

r

s

t

u

v

w

x

y

z

Continue

Can Haz Tests?





EDIT TEST

Test Name

Test Spec

```
1  [{
2    name: "S_AN1 branching and AN_BMI_UNDERWEIGHT_RANGE",
3    test: {
4      "AN1a": "kg", // weight in kilograms
5      "AN1b": "cm", // height in cm
6      next: "S_AN2",
7      setup: { tag: { AN_WEIGHT: 33.75, AN_HEIGHT: 1.5 } },
8      tagged: {
9        AN_BMI: 15,
10       AN_BMI_UNDERWEIGHT_RANGE: 1,
11       AN_BMI_NORMAL_WEIGHT: 44,
12       AN_BMI_NORMAL_WEIGHT_INCREASE: 10, // rounded down from 10.25
13     },
14   },
15 }, {
16   name: "S_AN2 Yes/No branching",
17   test_mc: [
18     'AN2',
19     'AN3', // yes
20     'S_BN1', // no
21   ],
22 }, {
23   name: "S_AN5: AN4 + AN5 = 0 --> skip to next section",
24   test: {
25     "page": { "AN4": 'n' },
26     "AN5": 'n',
27     next: "S_CHEEZEBURGER",
28     setup: { tag: [ 'CAT' ] },
29   },
30 },
```

Show admin console

- Edit
- Edit Survey
- Take Survey
- Visualize
- Test Suite
- Run All Tests
- Run All Tests (TAP)
- Edit Test
- Run Test

- [Back to site.](#)
- [Turn Admin Off!](#)
- [Logout](#)

Run Tests Online



TEST RESULTS

Test Results - Pass

1..32

- ok 1 - S_AN1 branching and AN_BMI_UNDERWEIGHT_RANGE
- ok 2 - S_AN1 branching and AN_BMI_NORMAL_RANGE (end module)
- ok 3 - S_AN1 branching and AN_BMI_NORMAL_RANGE (end module 2)
- ok 4 - S_AN2 Yes/No branching mc answer 1
- ok 5 - S_AN2 Yes/No branching mc answer 2
- ok 6 - S_AN3 Yes/No branching mc answer 1
- ok 7 - S_AN3 Yes/No branching mc answer 2
- ok 8 - S_AN4 Yes/No branching mc answer 1
- ok 9 - S_AN4 Yes/No branching mc answer 2
- ok 10 - S_AN5: AN4 + AN5 = 0 --> skip to BED
- ok 11 - S_AN5: otherwise (MALE)
- ok 12 - S_AN5: otherwise (FEMALE)
- ok 13 - S_AN6 Yes/No branching mc answer 1
- ok 14 - S_AN6 Yes/No branching mc answer 2
- ok 15 - S_AN6a Yes/No branching mc answer 1
- ok 16 - S_AN6a Yes/No branching mc answer 2
- ok 17 - AN_DEFAULT_SUBCLINICAL_AN6

[Show admin console.](#)

- [Edit](#)
- [Edit Survey](#)
- [Take Survey](#)
- [Visualize](#)
- [Test Suite](#)
- [Run All Tests](#)
- [Run All Tests \(TAP\)](#)
- [Edit Test](#)
- [Run Test](#)

- [Back to site.](#)
- [Turn Admin Off!](#)
- [Logout](#)

Run Test Suite Online

Test Results - Pass

Tests=832, Passed=832, Failed=0, 29 wallclock secs (26.53 usr + 0.09 sys = 26.62 CPU)

Test Name	Tests Run	Tests Passed	Tests Failed	Test Result	
W	5	5	0	Pass	Edit Run Test
PD	23	23	0	Pass	Edit Run Test
PPD	4	4	0	Pass	Edit Run Test
A	29	29	0	Pass	Edit Run Test
SAD	31	31	0	Pass	Edit Run Test
SP	22	22	0	Pass	Edit Run Test
GAD	34	34	0	Pass	Edit Run Test
OCD	51	51	0	Pass	Edit Run Test
PTSD	34	34	0	Pass	Edit Run Test
MDE	38	38	0	Pass	Edit Run Test
AN	32	32	0	Pass	Edit Run Test
BN	16	16	0	Pass	Edit Run Test
BED	17	17	0	Pass	Edit Run Test
SD_CAN	22	22	0	Pass	Edit Run Test
SD_st	26	26	0	Pass	Edit Run Test
SD_o	28	28	0	Pass	Edit Run Test
SD_ss	22	22	0	Pass	Edit Run Test

Automated Testing

```
# Surveys - validate and run test suite
if ( $asset->isa('WebGUI::Asset::Wobject::Survey') ) {

    # Validate Survey
    my $warnings = $asset->surveyJSON->validateSurvey;
    for my $warning (@$warnings) {
        push @{$items{survey}}, error( $asset, $warning );
    }
    # Don't bother running tests if Survey has validation warnings
    next if @$warnings;

    # Create test user
    my $user = WebGUI::User->new($session, 'new');
    $test_group->addUsers([$user->userId]);
    $session->user( { user => $user } );

    # Run tests
    $session->log->debug("Running tests for Survey: @{{[ $asset->getUrl ]}");
    my $out = eval { $self->run_tests($asset) };
    $out = $@ if $@;

    if ($out !~ m/NOTESTS|PASS/) {
        push @{$items{survey}}, error( $asset, $out );
    }

    # Clean up
    $user->delete;
}
}
```

Automated Checking

- Also useful for alerting about:
 - Incorrect permissions
 - Incorrect asset settings
 - Absolute URLs
 - Broken or missing Templates
- Using whatever heuristic is relevant for your site
- Can email you as soon as something breaks

Summary

- Extending WebGUI
 - Traditional Approach
 - The Core
 - Git
 - Perl on the Web
 - Tests on the Web

See Also

- Flux

A pluggable rule-based authorisation layer for WebGUI that gives administrators and content managers the ability to extend their site with dynamic behaviour

- Carousel

Makes it possible for end-users to extend Flash and Javascript objects from the WebGUI content management interface



Questions?



Choosing when to merge/rebase

- Stay on release commits for live branches

```
File Edit View Terminal Help
patspam@pattop:/data/WebGUI (crypt)$ git tag v7.7.19-tagged-by-patspam edae7a3b1c955abea006cb90fa73663c1d5187a9
patspam@pattop:/data/WebGUI (crypt)$ git rebase v7.7.19-tagged-by-patspam
First, rewinding head to replay your work on top of it...
Applying: Squashed commit of the following:
/data/WebGUI/.git/rebase-apply/patch:198: trailing whitespace.
    my $message_body =
/data/WebGUI/.git/rebase-apply/patch:295: trailing whitespace.

/data/WebGUI/.git/rebase-apply/patch:365: trailing whitespace.
=head2 handler ( session )
/data/WebGUI/.git/rebase-apply/patch:376: trailing whitespace.

/data/WebGUI/.git/rebase-apply/patch:379: trailing whitespace.

warning: squelched 83 whitespace errors
warning: 88 lines add whitespace errors.
patspam@pattop:/data/WebGUI (crypt)$
```

Keep it clean

- Using rebase keeps your history linear

The screenshot shows a version control interface with a commit history list at the top and a detailed view of a specific commit below.

Commit History:

Commit Message	Author	Date
template_usage: skip any assetClass that cannot be instantiated	Patrick Donelan <pat@patspam.com>	2009-08-17 03:24:35
Added Template usage options to Template Edit Page	Patrick Donelan <pat@patspam.com>	2009-08-04 03:25:03
v7.7.18-tagged-by-patspam	Patrick Donelan <pat@patspam.com>	2009-07-12 06:37:37
Preparing for 7.7.18 release.	Colin Kuskie <colink@perldreamer.com>	2009-08-21 00:30:14
Update number of tests to skip when the correct modules aren't pr	Colin Kuskie <colink@perldreamer.com>	2009-08-21 00:23:16
Update test to follow new style of adding words to search index.	Colin Kuskie <colink@perldreamer.com>	2009-08-21 00:13:15
requestAutoCommit forks a workflow to commit, use skipAutoCommit	Colin Kuskie <colink@perldreamer.com>	2009-08-20 22:06:24
Document that Readonly is a required module. Fixes bug #10801	Colin Kuskie <colink@perldreamer.com>	2009-08-20 20:30:10
Give the Carousel a minimum width, and expose it as a template vs	Colin Kuskie <colink@perldreamer.com>	2009-08-20 19:12:59
Update existing EMS'es with a default schedule template. Fix for	Colin Kuskie <colink@perldreamer.com>	2009-08-20 16:53:12
Log any errors found when calling Shop::Pay->getDrivers, instead	Colin Kuskie <colink@perldreamer.com>	2009-08-20 16:32:00
add a few additional macro tests	Graham Knop <graham@plainblack.com>	2009-08-20 15:24:39
fixed #10797: Unable to search for non-ASCII text	Graham Knop <graham@plainblack.com>	2009-08-20 15:24:31

SHA1 ID: 3af70971d84834dc4f19928dac0ba6c65d8cbc3b

Find: next prev commit containing: Exact All fields

Search: [Search box]

Diff: Old version New version Lines of context: 3

Commit Details:

- Author: Patrick Donelan <pat@patspam.com> 2009-08-17 03:24:35
- Committer: Patrick Donelan <pat@patspam.com> 2009-08-23 01:31:52
- Parent: [20b44671d1757cf060097de779bbe73df5d3622](#) (template_usage: skip
- Branch: [template_usage](#)
- Follows: [v7.7.18-tagged-by-patspam](#)
- Precedes:

Made t/Whitespace.t pass for template_usage pod

lib/WebGUI/Asset/Template.pm

```
index 99dc3aa..8891953 100644
@@ -1340,8 +1340,11 @@ sub www_styleWizard {
 }
```