

Поддержка legacy Perl кода

Поддержка legacy Perl кода



Непонятный код

```
1 package My::Lib::Common;
2
3 use strict;
4 no warnings;
5
6 use My::Lib::Flags "FLAG_2";
7
8 sub calc_rep_20210101 {
9     my $flag2 = FLAG_2();
10    return _load_special_report(date => "20210101") if $_[1] =~ /$flag2/;
11 }
12 sub now {
13     return time();
14 }
15
16 1;
17 |
```

Непонятный код

- Картинка отражающая настроение



Плохо документирован

```
1 package My::Lib::MainActor;
2
3 use strict;
4
5 =head1 NAME
6
7 MainActor
8
9 =head1 SYNOPSIS
10
11 my $o = My::Lib::MainActor(%args) or die "got error";
12
13 =head1 AUTHOR
14
15 gottacode@jimere.neverused.domain
16
17 =cut
18
19
20 1;
21
22 __END__
23
24 $ podlint ./My/Lib/MainActor.pm && echo "pod ok" || echo "pod error"
25 pod ok
```

Плохо спроектирован

- Картинки из интернета



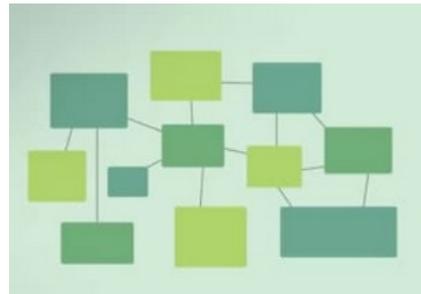
Плохо спроектирован

- Картинки из интернета



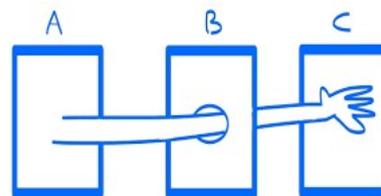
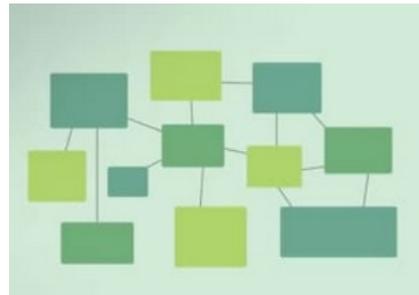
Плохо спроектирован

- Картинки из интернета



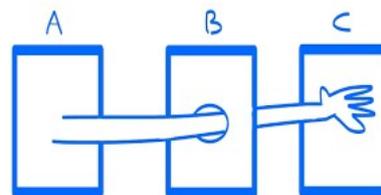
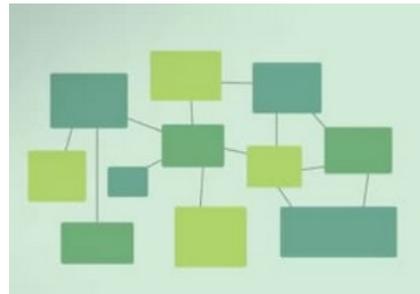
Плохо спроектирован

- Картинки из интернета



Плохо спроектирован

- Картинки из интернета



Непрозрачный деплоймент

- `sh /root/project_my_deploy.sh`
- Brainfuck based deployment bot.

Архаика

- Старый perl
- Старая ОС

Архаика

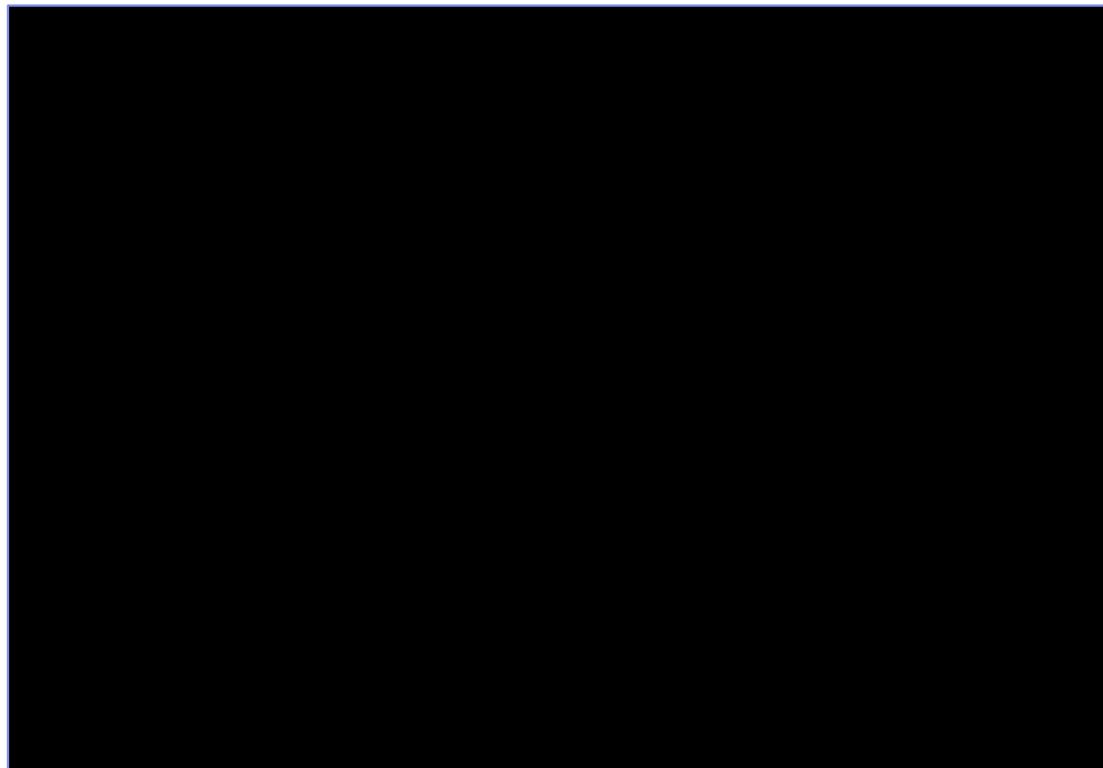
- Старый perl
- Старая ОС
- => Сложность добавления/обновления модулей



Бинари без исходников

...

Бинари без исходников



Недостаточное покрытие тестами

- Проверочные скрипты вместо тестов
- Прогонов тестов нет
- Отчётов нет
- Юнитов нет
- Уверенности нет
- Много тестового кода, но толку нет

Недостаточное покрытие тестами



Признаки

- Код непонятный
- Плохо документирован
- Плохо спроектирован
- Непрозрачный деплоймент.
- Старый Perl, OS
- Потерянные исходники кастомных библиотек
- Нет достаточного покрытия тестами
- Нет dev контура
- Другие недостатки, от которых подташнивает

Thanks God it's perl.

- PDD

PDD — Print Driven Debug

- Вывод аргументов конструктора без документации

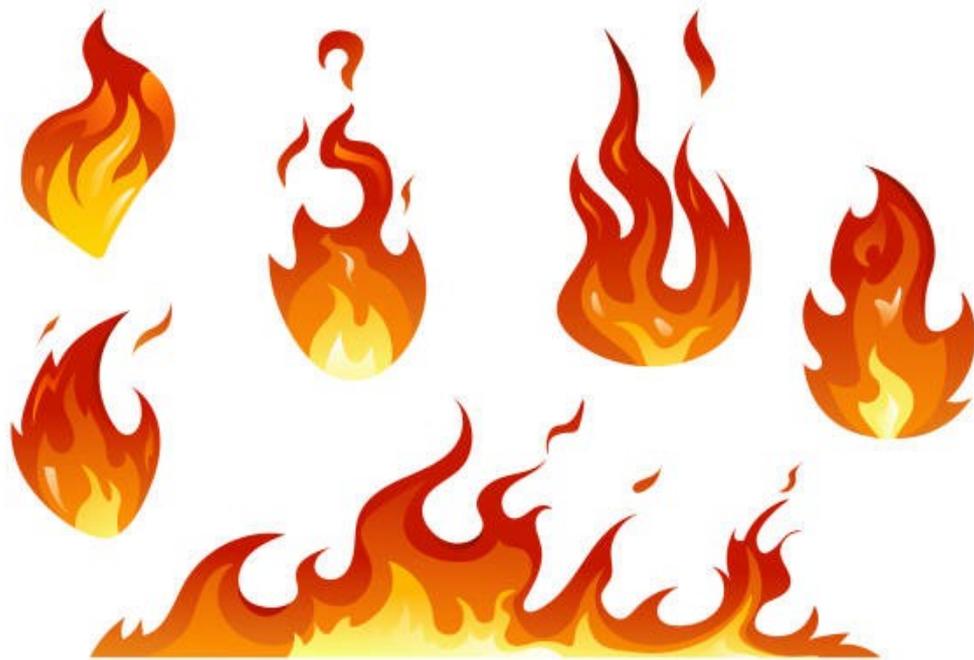
```
13 =head1 AUTHOR
14
15 gottacode@jimere.neverused.domain
16
17 =cut
18
19 sub new {
20     my $class = shift;
21     my %args = @_;
22
23     use Data::Dump "pp";
24     warn "MainActor new:".pp(\%args);
25     warn "MainActor new class: $class";
26
27     return bless %args, $class;
28 }
29
30 1;
31
32
```

PDD — Print Driven Debug

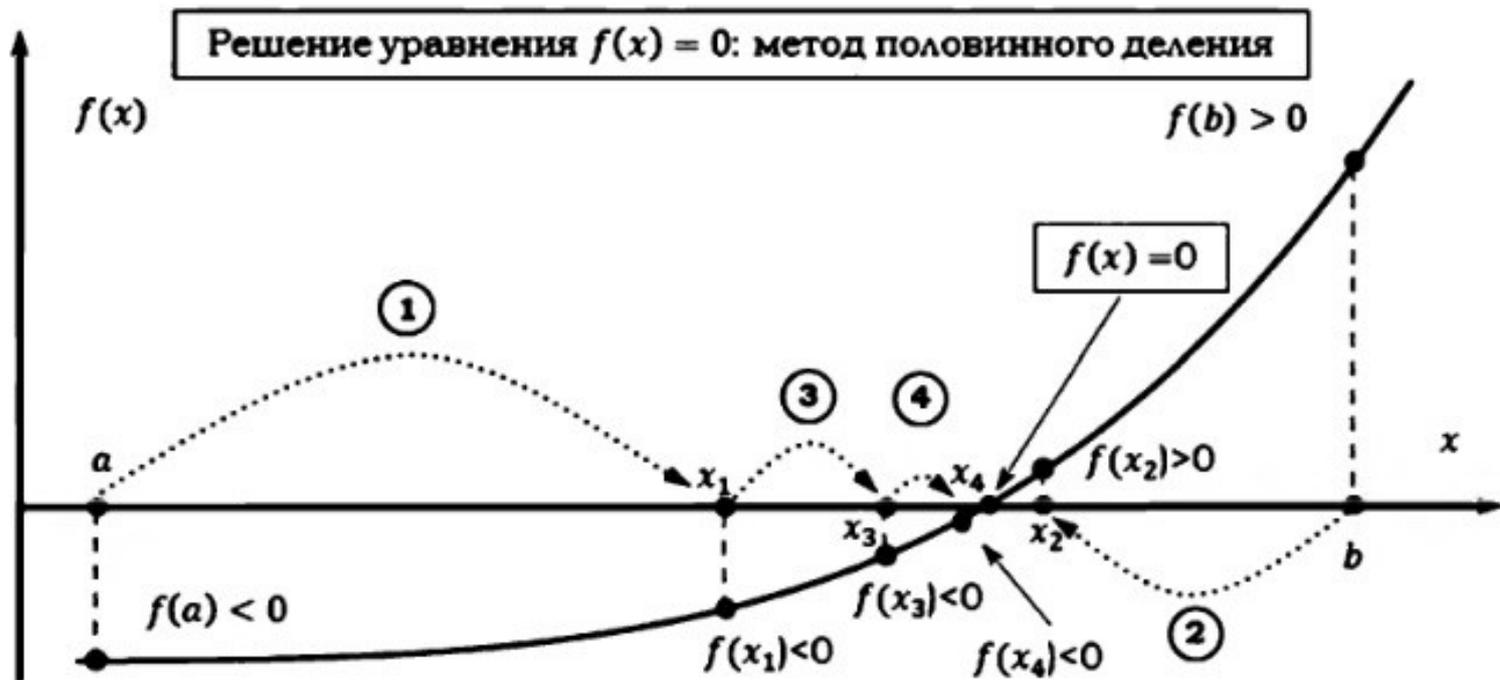
- Вывод нужных полей с метками.

```
19  sub new {
20    my $class = shift;
21    my %args = @_;
22
23    use Data::Dump "pp";
24    warn "MainActor new class: $class";
25    warn "MainActor new fields: ".join(", ", keys %args);
26
27    return bless %args, $class;
28 }
```

Ломается непонятно где



Метод половинного деления



Метод половинного деления

```
sub calc_next_step {
  my $self = shift;

  my %args = @_;
  my $actor = $args{actor};
  my $window = $args{window};
  my $day = $self->now();
  my $cart = $actor->cart();
  my $app = $self->application();
  if ($app->window() eq $window->id) {
    return;
  } elsif ($cart->is_empty) {
    $cart = $self->fill_cart(actor => $actor);
    return $self->retry_step(%args);
  } else {
    # ....
    # 3000 строк хорошо форматированного кода со 100 выходами
    # ...|
    return $self->step_or_stop(%args) if $day != $self->today();
  }
  return;
}
```

Метод половинного деления

```
.....
} else {
  # ....
  print STDERR "befor some call 2";
  # увидели это сообщение, прошли ещё
  # .. x\2 строка
  warn "after some suspicious call";
  # не увидели - поднялись выше
  return $self->step_or_stop(%args) if $day != $self->today();
}
```

Thanks God It's Perl

Получилось — документируй!

Видишь warn — предупреди коллег!

- SIGWARN

```
1 package My::Lib::NewLogger;
2
3 use strict;
4
5 use POSIX "strftime";
6 use Time::HiRes "gettimeofday";
7 use constant MAX Depths => 5;
8 sub import { turn_sigwarn()};
9
10 sub turn_sigwarn {
11     $SIG{__WARN__} = \&logg;
12 }
13 sub logg {
14     my $message = shift;
15     chomp($message);
16     my $pid = $$;
17     my ($sec, $msec) = gettimeofday();
18     my $ts = strftime("%F %T", localtime $sec).sprintf("%.06d", $msec);
19     my $trace = get_trace();
20     print STDERR "[$ts] [pid:$pid] [DEBUG] $message ---- $trace\n";
21     return 1;
22 }
```

Видишь warn — предупреди коллег!

- trace

```
24 sub get_trace {
25     my $i      = 0;
26     my $message = '';
27
28     while ( $i++ < MAX_DEPTHS() ) {
29         my @data = caller( $i );
30         last unless scalar @data;
31
32         my ( $mod, $line ) = @data[0, 2];
33         $message .= $message ? "/" : " at ";
34         $message .= sprintf ("%s(%s)", map{$_ || ""} ( $mod, $line ));
35     }
36
37     return $message;
38 }
39 1;
40
```

Видишь warn — предупреди коллег!

- Было

```
a.gnatyna@rosa2021 ~/own/perl/perl-conf.ru_24/code $ perl -I. -e 'use My::Lib::Runner; My::Lib::Runner->run();'  
MainActor new class: My::Lib::MainActor at My/Lib/MainActor.pm line 24.  
MainActor new fields: at My/Lib/MainActor.pm line 25.  
error
```

- Стало

```
a.gnatyna@rosa2021 ~/own/perl/perl-conf.ru_24/code $ perl -I. -MMy::Lib::NewLogger -e 'use My::Lib::Runner; My::Lib::Runner->run();'  
[2024-09-15 18:22:15.966064] [pid:135552] [DEBUG] MainActor new class: My::Lib::MainActor at My/Lib/MainActor.pm line 24. --- at My::Lib::MainActor(24)/My::Lib::Runner(9)/main(1)  
[2024-09-15 18:22:15.966159] [pid:135552] [DEBUG] MainActor new fields: at My/Lib/MainActor.pm line 25. --- at My::Lib::MainActor(25)/My::Lib::Runner(9)/main(1)  
[2024-09-15 18:22:15.966200] [pid:135552] [DEBUG] error --- at My::Lib::MainActor(27)/My::Lib::Runner(9)/main(1)
```

\$SIG{__DIE__}

```
19 sub new {
20   my $class = shift;
21   my %args = @_;
22
23   use Data::Dump "pp";
24   warn "MainActor new class: $class";
25   warn "MainActor new fields: ".join(", ", keys %args);
26   unless($args{iknowwhatido}){
27     my $ret = eval {
28       get_result();
29     } or do {
30       die "$@\n";
31     };
32     return;
33   }
34   return bless %args, $class;
35 }
36 sub get_result {
37   return;
38 }
```

\$SIG{__DIE__}

```
8 sub import { turn_sigwarn(); turn_sigdie()};
9
10 sub turn_sigdie {
11     $SIG{__DIE__} = sub {logg("called die: ".$_[0]); die $_[0]};
12 }
13
14 sub turn_sigwarn {
15     $SIG{__WARN__} = \&logg;
16 }
```

```
a.gnatyna@rosa2021 ~/own/perl/perl-conf.ru_24/code $ perl -I. -MMy::Lib::NewLogger -e 'use My::Lib::Runner; My::Lib::Runner->run(); print "done job"'
[2024-09-15 18:37:51.710362] [pid:139199] [DEBUG] MainActor new class: My::Lib::MainActor at My/Lib/MainActor.pm line 24. --- at My::Lib::MainActor(24)/My::Lib::Runner(9)/main(1)
[2024-09-15 18:37:51.710448] [pid:139199] [DEBUG] MainActor new fields: at My/Lib/MainActor.pm line 25. --- at My::Lib::MainActor(25)/My::Lib::Runner(9)/main(1)
[2024-09-15 18:37:51.710483] [pid:139199] [DEBUG] called die: --- at My::Lib::NewLogger(11)/My::Lib::MainActor(30)/My::Lib::Runner(9)/main(1)
```

Sentry

Unresolved Issues (450) ▾

Sort by: Last Seen ▾

🔍 is:unresolved



<input type="checkbox"/>	Resolve ▾	Ignore ▾	Merge	★	⋮	▶	GRAPH:	24h 14d	EVENTS	USERS	ASSIGNEE
<input type="checkbox"/>	✓								500	45	NM ▾
TypeError func(components/App) in components/App.js this.myCodelsPerfect is not a function 🔧 FRONTEND-REACT-V ⌚ an hour ago – 2 months old SI-20											
<input type="checkbox"/>									490	80	👤 ▾
Error cart.forEach(app) in /Users/vu/Documents/sdk_demos/demos/express/app.js No inventory for nails 🏠 EXPRESS-DEMO-4 ⌚ 3 hours ago – 2 months old											
<input type="checkbox"/>									140	49	NM ▾
Error apply(components/App) in components/App.js 500 - Internal Server Error 🔧 FRONTEND-REACT-R ⌚ 3 hours ago – 3 months old											
<input type="checkbox"/>									86	38	👤 ▾
TypeError null.<anonymous>(app) in /app/app.js obj.DoesNotExist is not a function 🏠 EXPRESS-DEMO-6 ⌚ 21 hours ago – a month old											

Ура! Оно работает!

- Но деплой не всегда успешный...

Ура! Оно работает!

- Но деплой не всегда успешный...
- Тут нам поможет `docker`, `docker-compose`
- `docker-compose -f docker-compose.yml -f docker-compose_snapshots.yml -f my.yml up`

Пример конфига для docker-compose

```
62 perltest:
63   image: ${IMAGE_TERMBASE:-registry.host.my/app1/bff/api_base/dev:latest}
64   hostname: term
65   working_dir: /home/www/
66   depends_on:
67     - api
68     - nginx
69     - mock
70     - redis
71     - db_mysql
72   networks:
73     - my
74   environment:
75     GITLAB_MR: ${CI_PROJECT_URL}/-/merge_requests/${CI_MERGE_REQUEST_IID}/diffs
76     GITLAB_JOB: ${CI_JOB_URL}
77     GITLAB_PIPELINE: ${CI_PIPELINE_URL}
78     TZ: Europe/Moscow
79     PERLSOPT: '-I/home/www/test/lib -I/home/www/lib'
80     FILTER_TESTS: ${FILTER_TESTS}
81   volumes:
82     - ../../docker/bin/multi-wait-for.sh:/usr/local/bin/multi-wait-for.sh:ro
83     - ../../:/home/www/
84     - ./run-deploy.sh:/run-deploy.sh:ro
85     - ./config:/home/www/config
86   entrypoint: ["/bin/bash", "-xeuc"]
87   command:
88     - |
89     multi-wait-for.sh \
90       db_mysql:3306 \
91       redis:6379 \
92       -t180
93     /add_hosts.sh
94     /run-deploy.sh
95     if [[ -z ${RUN_TESTS-''} ]]; then
96     | sleep 10000
97     else
98     prove \
99     -P HTML=outfile:report${REPORT_NAME}.html,force_inline_css:1,force_inline_js:1 \
100     --harness Test::Report::HarnessAllureNotBail \
101     -m --timer $$((cat ${TEST_SUITE-'docker/dev/list_of_tests'} | perl -ne '/^(.*)/ ? system($$1) : /\w/ and !/#/ and print $$_'
102     fi
```

Проверки/тесты/проверки?

- TAP
- Test::Harness prove

```
a.gnatyna@rosa2021 ~/own/perl/perl-conf.ru_24/code $ find tests
tests
tests/dummy.pl
tests/stable.list
tests/trace.pl
a.gnatyna@rosa2021 ~/own/perl/perl-conf.ru_24/code $ prove -I. -m --timer $(awk '!/#/{ print $0 }' tests/stable.list)
[20:44:05] tests/trace.pl .. ok      58 ms ( 0.00 usr 0.00 sys + 0.05 cusr 0.00 csys = 0.05 CPU)
[20:44:05] tests/dummy.pl .. ok    56 ms ( 0.00 usr 0.00 sys + 0.05 cusr 0.01 csys = 0.06 CPU)
[20:44:05]
All tests successful.
Files=2, Tests=3, 0 wallclock secs ( 0.02 usr 0.01 sys + 0.10 cusr 0.01 csys = 0.14 CPU)
Result: PASS
```

Allure

Suites

order ▾ name ▾ duration ▾ status ▾ Status: 0 0 8 0 0 Marks:     

io.qameta.allure.IssuesRestTest	4
✓ #2 shouldCreateUserNote(String) Create issue via api	4ms
✓ #1 shouldCreateUserNote(String) Create issue via api	64ms
✓ #4 shouldDeleteUserNote(String) Close issue via api	3ms
✓ #3 shouldDeleteUserNote(String) Close issue via api	5ms
> io.qameta.allure.IssuesWebTest	2
> io.qameta.allure.PullRequestsWebTest	2

io.qameta.allure.IssuesRestTest.shouldCreateUserNote

Passed shouldCreateUserNote(String) Crea

Overview History Retries

Success rate 100% (7 of 7)

passed 9/5/2023 at 16:01:30

passed 6/19/2023 at 14:15:26

passed 7/5/2022 at 0:17:25

passed 7/5/2022 at 0:15:29

passed 7/5/2022 at 0:13:04

passed 7/5/2022 at 0:10:19

Тесты? Тесты! Тесты!

- Test::Class::Load

```
use Test::Class::Load qw(t/tests t/lib);  
Test::Class->runtests;
```

Test::Class::Load

```
1 package Test::My::Loader;
2
3 use strict;
4
5 use Test::More;
6 use Module::Runtime 'require_module';
7
8 use base qw( Test::Class::Load );
9
10 sub _load {
11     my ( $class, $file, $dir ) = @_;
12     $file =~ s/^\Q$dir\E//;
13     $file =~ s/\.pm$//;
14     $file =~ s#^/##;
15     my $package = join(":",split('/',,$file));
16     require_module($package);
17     diag("call _make_world for $package");
18     $package->_make_world();
19 }
20 sub is_test_class {
21     my ( $self, $file, $dir ) = @_;
22
23     return unless $self->SUPER::is_test_class( $file, @_ );
24     return 0 if $file !~ m{Test/My/T/};
25     return 1;
26 }
27
28 1;
29
```

```
1 package Test::My::T::MainActor;
2
3 use strict;
4
5 use Test::More;
6
7 use base qw( Test::Class );
8
9 sub _make_world {
10     diag("setup before run");
11     1;
12 }
13 sub t01_do_test: Test {
14     ok(1, "dummy test");
15 }
16
17 1;
18
```

Test::Class::Load

```
a.gnatyna@rosa2021 ~/own/perl/perl-conf.ru_24/code $ perl -I. -e 'use Test::My::Loader qw(.);Test::Class->runtests()'
# call _make_world for Test::My::T::Logger
# setup before run
# call _make_world for Test::My::T::MainActor
# setup before run
1..2
ok 1 - dummy test
ok 2 - dummy test2
```

Must Have

- VCS

Must Have

- VCS
- TT

Must Have

- VCS
- ТТ
- Связывание

Подружить VCS & TT

```
09:22 $ git hist|head -4
```

```
* 2134567981 2024-09-11 | MY-35217: Fix JSON bug [Chuck Norris]
```

```
* 2fde232333 2024-09-12 | MY-23198 "remove old code" [Jim]
```

```
* dac2322333 2024-09-11 | MMY-123: add no check option [Mick Mouse]
```

```
* cf32489822 2024-09-11 | MY-33028: script prototype [Alexander Gnatyna]
```

Must Have

- VCS
- ТТ
- Связывание
- Документирование

Must Have

- VCS
- ТТ
- Связывание
- Документирование
- Линтеры

Линтеры

- Test::Pod
- Perlcritic (Test::Perl::Critic::Git)

Must Have

- VCS
- ТТ
- Связывание
- Документирование
- Линтеры
- `perl -cw`

perl -cw

```
10:57 $ git diff --cached --name-only|egrep '\.(pm|pl|t)$'|xargs -I{} perl -I. -cw {}  
My/Lib/Common.pm syntax OK  
My/Lib/Flags.pm syntax OK  
My/Lib/Main.pm syntax OK  
My/Lib/MainActor.pm syntax OK  
My/Lib/NewLogger.pm syntax OK  
My/Lib/Runner.pm syntax OK  
✓ ~/own/perl/perl-conf.ru_24/code [master L|●7...3]
```

Legacy! Legacy?

- Опыт
- Стать legacy проще, чем кажется.

Legacy! Legacy?

