



THE DEVELOPER'S
CONFERENCE

DevOps para Sistemas Embarcados

Eduardo Witter

Diego Garcia

DATACOM

Sobre nós

DATAKOM



linkedin.com/in/**eduardo-witter**
witter@datacom.com.br
ewitter@gmail.com

- Engenharia Elétrica (UFRGS), MBA em Gestão e Liderança (Unisinós)
- Mestrando INF/UFRGS
- Na Datacom desde 2007. Nestes 11 anos:
 - Desenvolvedor C/C++, Python
 - Scrum Master; Product Owner
 - Gerente Técnico
 - DevOps



linkedin.com/in/**diego-cardoso-garcia**
dgc.cardoso@gmail.com
@diego_cgarcia

- Engenharia de Computação (IPA), Especialização em Engenharia de Software (UFRGS)
- Na Datacom desde 2009. Nestes 9 anos:
 - Desenvolvedor Ruby, Python
 - Scrum Master
 - DevOps

Bem-vindos ao nosso mundo

- Switches, roteadores, ...
- Maior P&D de Telecom da América Latina
- Hardware e Software feitos integralmente no Brasil
- Principais clientes: operadoras, provedores, governo e redes corporativas

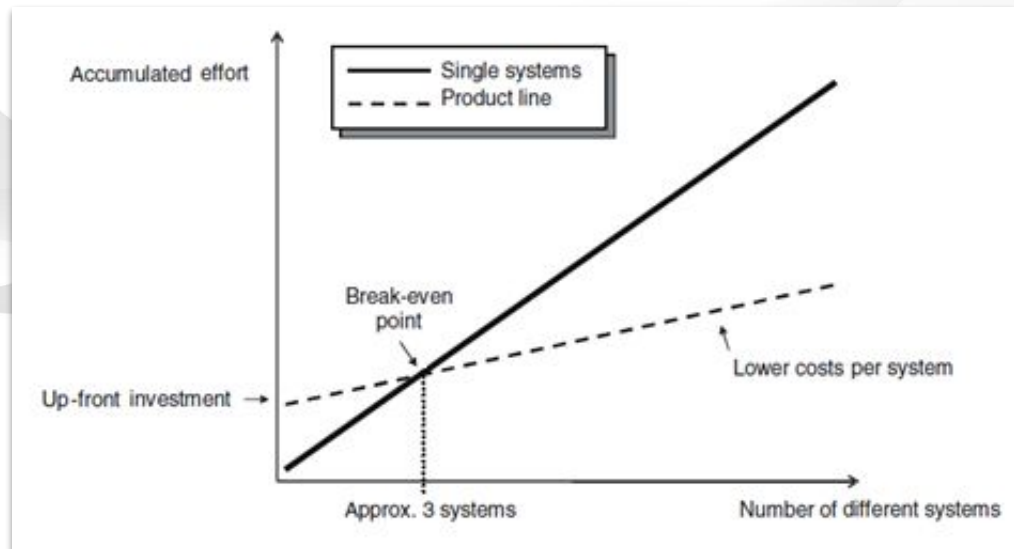


Como é o software?

- Linha de Produtos de Software, muito modular. Diversos benefícios, mas tem um custo
- O mesmo OS em todos os equipamentos
- Objetivos: plataforma para novos produtos por décadas.

Crescer não pode significar o colapso

- **~100 pessoas no projeto**
- **Desenvolvimento em 4 cidades**



Separação em repositórios

Repositórios:

- DmOS: 304
- Testes: 41
- Testbed: 117
- DevOps: 43

Tudo está sob controle de versão. Documentação, Testes, Ambientes de teste

Cada repositório tem alguns metadados:

- **Tipo:** testbed, doc, test, dmos, devops, etc
- **Maintainer:** responsável pelo módulo
- **Manager:** consolidação administrativa

Desenvolvimento

Ok, vamos fazer um commit?

All My **Projects** **People** **Plugins** **Documentation**

[Changes](#) [Watched Changes](#)

Search term

Change **58280** - **Merged**

[Bug89785] Trunk entries returning when traversing by port

Change-Id: If55e405b119bdc24a03334801f314b52f44a677f
Signed-off-by: lazzarotto <lazzarotto@datacom.ind.br>

Included in ▼ Patch Sets (4/4) ▼ Download ▼

Owner Fabio Lazzarotto

Reviewers Auto QA
 George Redivo
 Pedro Bianchi [Add...](#)
 Filipe Utzig
 Henrique Jung witter

Project [dmos-hal-switch-l2addr-l1-bcm](#)

Branch [develop](#)

Topic

Updated 2 days ago

[Cherry Pick](#) [Revert](#)

Author lazzarotto <lazzarotto@datacom.ind.br> Aug 13, 2018 1:22 PM
 Committer lazzarotto <lazzarotto@datacom.ind.br> Aug 13, 2018 1:22 PM
 Commit 49ec4bc3eb065137909c68d370d67abd8a81f1a [\(gitweb\)](#)
 Parent(s) 7c672b50b6f69b7380fbb96952a4b5bf19aa6585 [\(gitweb\)](#)
 Change-Id If55e405b119bdc24a03334801f314b52f44a677f [\(gitweb\)](#)











Code-Review +2 George Redivo
 +1 Pedro Bianchi
 Verified +1 Auto QA

Files

[Open All](#) Diff against: [Base](#) ▼

File Path	Comments	Size
<input type="checkbox"/> Commit Message		
<input type="checkbox"/> include/internal/hal_bcm_l2addr.h	4	
<input type="checkbox"/> src/hal_bcm_l2addr.cc	24	
	+20, -8	

History Expand All

 Fabio Lazzarotto	Uploaded patch set 1.	Aug 10 3:14 PM
 Auto QA	Patch Set 1: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_clang+dmos-hal-switch-l2addr-ll-bcm/59/ (2/10)	Aug 10 3:14 PM
 Auto QA	Patch Set 1: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_check-dependents+dmos-hal-switch-l2addr-ll-bcm/48/ (1/10)	Aug 10 3:14 PM
 Auto QA	Patch Set 1: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_doxycheck+dmos-hal-switch-l2addr-ll-bcm/59/ (4/10)	Aug 10 3:14 PM
 Auto QA	Patch Set 1: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_coverage+dmos-hal-switch-l2addr-ll-bcm/59/ (5/10)	Aug 10 3:14 PM
 Auto QA	Patch Set 1: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_doc+dmos-hal-switch-l2addr-ll-bcm/59/ (8/10)	Aug 10 3:14 PM
 Auto QA	Patch Set 1: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_gsan+dmos-hal-switch-l2addr-ll-bcm/25/ (7/10)	Aug 10 3:14 PM
 Auto QA	Patch Set 1: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_cpp-style+dmos-hal-switch-l2addr-ll-bcm/59/ (9/10)	Aug 10 3:14 PM
 Auto QA	Patch Set 1: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_dmcheck+dmos-hal-switch-l2addr-ll-bcm/59/ (10/10)	Aug 10 3:14 PM
 Auto QA		Aug 10 3:20 PM ←

Patch Set 1: Code-Review-1 Verified+1

Build Unstable

http://jenkins.ped.datacom.ind.br/job/+commit_clang+dmos-hal-switch-l2addr-ll-bcm/59/ : SUCCESS

http://jenkins.ped.datacom.ind.br/job/+commit_valgrind+dmos-hal-switch-l2addr-ll-bcm/59/ : SUCCESS

http://jenkins.ped.datacom.ind.br/job/+commit_doxycheck+dmos-hal-switch-l2addr-ll-bcm/59/ : SUCCESS

http://jenkins.ped.datacom.ind.br/job/+commit_coverage+dmos-hal-switch-l2addr-ll-bcm/59/ : SUCCESS

http://jenkins.ped.datacom.ind.br/job/+commit_py3compat+dmos-hal-switch-l2addr-ll-bcm/60/ : SUCCESS









http://jenkins.ped.datacom.ind.br/job/+commit_gsan+dmos-hal-switch-l2addr-ll-bcm/25/ : SUCCESS


http://jenkins.ped.datacom.ind.br/job/+commit_doc+dmos-hal-switch-l2addr-ll-bcm/59/ : SUCCESS


http://jenkins.ped.datacom.ind.br/job/+commit_cpp-style+dmos-hal-switch-l2addr-ll-bcm/59/ : UNSTABLE


http://jenkins.ped.datacom.ind.br/job/+commit_dmcheck+dmos-hal-switch-l2addr-ll-bcm/59/ : SUCCESS

http://jenkins.ped.datacom.ind.br/job/+commit_check-dependents+dmos-hal-switch-l2addr-ll-bcm/48/ : SUCCESS

	Filipe Utzig	Patch Set 1: Code-Review-1 (1 comment)
	witter	<p>Patch Set 1: Code-Review-1</p> <p>(2 comments)</p> <p>src/hal_bcm_I2addr.cc</p> <p>Line 439: é seguro sair sem preencher 'port' ? No if acima o tgid é preenchido.</p> <p>Line 454: nao era bom inicializar o tgid aqui?</p> <p>enfim, a função agora tem 2 parâmetros de saída. como o valor de retorno nao informa qual valor preenchido é o que de</p>
	Fabio Lazzarotto	<p>Patch Set 1:</p> <p>(2 comments)</p> <p>src/hal_bcm_I2addr.cc</p> <p>Line 439: A princípio é, pois port foi zerado antes da chamada dessa função com <code>bcm_I2_addr_t_init()</code> e não existe <code>gport=0</code>. Acredito que a MPU tenha um prefixo de local port, daí seria <code>0x800000</code>, ou algo assim. Dessa forma, mesmo que tenham aparecer em um traverse de lag com <code>port=0</code>. Vou testar para confirmar.</p> <p>Line 454: Não me preocupei muito com o que está fora do <code>if (SOC_IS_HELIX4(module))</code> Nos demais chips o tgid já está zerado pelo <code>bcm_I2_addr_t_init()</code> e nunca deu problema. Inclusive, nesses chips fazemos o traverse pelo <code>gport</code> mesmo para os trunks, e não pelo tgid. Vou colocar para não ficar dúvida.</p>
	Fabio Lazzarotto	Uploaded patch set 2.
	Auto QA	Patch Set 2: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_check-dependents+dmos-hal-switch-I2addr-II-
	Auto QA	Patch Set 2: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_dmcheck+dmos-hal-switch-I2addr-II-bcm/60/ (1
	Auto QA	Patch Set 2: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_valgrind+dmos-hal-switch-I2addr-II-bcm/60/ (1
	Auto QA	Patch Set 2: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_doxycheck+dmos-hal-switch-I2addr-II-bcm/60/

 Auto QA Patch Set 4: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_cpp-style+dmos-hal-switch-l2addr-ll-bcm/62/ (0)

 Auto QA Patch Set 4: Build Started http://jenkins.ped.datacom.ind.br/job/+commit_dmcheck+dmos-hal-switch-l2addr-ll-bcm/62/ (0)

 **Auto QA**

Patch Set 4: Verified+1

Build Successful

http://jenkins.ped.datacom.ind.br/job/+commit_clang+dmos-hal-switch-l2addr-ll-bcm/62/ : SUCCESS

http://jenkins.ped.datacom.ind.br/job/+commit_valgrind+dmos-hal-switch-l2addr-ll-bcm/62/ : SUCCESS

http://jenkins.ped.datacom.ind.br/job/+commit_doxyccheck+dmos-hal-switch-l2addr-ll-bcm/62/ : SUCCESS

http://jenkins.ped.datacom.ind.br/job/+commit_coverage+dmos-hal-switch-l2addr-ll-bcm/62/ : SUCCESS

http://jenkins.ped.datacom.ind.br/job/+commit_py3compat+dmos-hal-switch-l2addr-ll-bcm/63/ : SUCCESS


http://jenkins.ped.datacom.ind.br/job/+commit_gsan+dmos-hal-switch-l2addr-ll-bcm/28/ : SUCCESS

http://jenkins.ped.datacom.ind.br/job/+commit_doc+dmos-hal-switch-l2addr-ll-bcm/62/ : SUCCESS


http://jenkins.ped.datacom.ind.br/job/+commit_cpp-style+dmos-hal-switch-l2addr-ll-bcm/62/ : SUCCESS

http://jenkins.ped.datacom.ind.br/job/+commit_dmcheck+dmos-hal-switch-l2addr-ll-bcm/62/ : SUCCESS


http://jenkins.ped.datacom.ind.br/job/+commit_check-dependents+dmos-hal-switch-l2addr-ll-bcm/51/ : SUCCESS

 **Pedro Bianchi**

Patch Set 4: Code-Review+1

 **George Redivo**

Patch Set 4: Code-Review+1

 **George Redivo**

Patch Set 4: Code-Review+2

Gerrit Code Review

Change has been successfully merged by George Redivo

DATACOM Getting started DmOS Docs DevOps Testbed Tests Other Tools

Search

All Unknown fischer obsolete furlan-dmos stein-platt stein stein-dmos eder.costa fabricio.brito-dmos trombetta shared furlan ricardo.severo fabricio.brito castanheira victor

Module	Links	pyang	doxycheck	codesonar	py3compmpat	doc	cpp-styl e	spars e	valgrind	clang	checkp atchpl	dmchec k	gsan	avoid-fu ncs	covera ge	testbe d-sch ema	valgrind-tra ce-children	checkd mcmdref
dmos-hal-leds-ll-pl-interface		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	OK	94.87%		OK	OK
dmos-swupdate-app		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	OK	97.89%		OK	OK
dmos-libosprocessinfo		OK	1 Failure	OK	OK	OK	3 Failures		OK	OK		OK	OK	1 Failure	97.96%		OK	OK
dmos-hal-tcv		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	OK	95.99%		OK	OK
dmos-logdictionary		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	OK	100.0%		OK	OK
dmos-hal-cardinit		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	OK	99.18%		OK	OK
dmos-element-to-syslog		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	OK	96.44%		OK	OK
dmos-hal-leds		OK	3 Failures	OK	OK	OK	OK		OK	OK		OK	OK	OK	96.69%		OK	OK
dmos-fwupgrade		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	4 Failures	90.11%		OK	OK
dmos-notificationd		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	3 Failures	95.86%		1 Error	OK
dmos-core-dump-mgr		OK	1 Failure	OK	OK	OK	OK		OK	OK		OK	OK	OK	96.5%		OK	OK
dmos-proxy-ha		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	1 Failure	95.06%		OK	OK
dmos-hal-mpu-redundancy		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	OK	100.0%		OK	OK
dmos-hal-inventory		OK	19 Failures	OK	OK	OK	OK		OK	OK		OK	OK	OK	96.06%		OK	OK
dmos-libdmidware		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	3 Failures	97.92%		OK	OK
dmos-libpolledmem		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	OK	93.16%		OK	OK
dmos-libctx		OK	OK	OK	OK	OK	OK		OK	OK		OK	OK	OK	94.92%		OK	OK

O que aconteceu por baixo dos panos?

- Cada critério de qualidade é um job do Jenkins
- Cada job:
 - Sobe uma máquina virtual ou container, efêmeros
 - Configura o ambiente, idêntico ao do desenvolvedor e roda o teste
 - Envia o voto para o Gerrit
- Critérios de qualidade são project wide, adoção não é voluntária.

Que tipo de testes são rodados?

- Testes de unidade
- Testes de módulo
- Coverage
- Valgrind, Callgrind
- Sanitizers
- Documentação: do módulo, do código do módulo e para usuário
- Linters/checkers: yang, json, lua, python, shell/bash
- Coding standards
- Complexidade ciclomática
- Funções inseguras / obsoletas

Quebra de dependentes!

- APIs
- Comportamentos

Integração Contínua e Testes

integration queue

1: build



buildroot



farm de builders:
*10-40 minutos por build
*23 builds em paralelo
*sem virtualização
*muito I/O



2: provision



ANSIBLE



VAGRANT



16 execuções paralelas


3: sanitize




firmware images



4: tests



robot framework



tempo:
1 hora (hard limit)
testes: cobertura x tempo
testes confiáveis

ambientes heterogêneos:
*hardware físico
*simuladores (virtual)
*outros fabricantes
*produtos legados Datacom

5: finalize



git



integrado:



2 horas (+-20 minutos), 7x24



Versionando os ambientes

```

#-----
# Config Agent01
#-----
# The T2 Test Runner
config.vm.define :agent01, primary: true do |agent01|
  agent01.vm.box = "dmos-t2infra-packer/basebox"
  agent01.vm.box_version = "0.0.3"
  agent01.vm.hostname = 'runner'
  agent01.vm.synced_folder './', '/vagrant', type: 'nfs'
  agent01.vm.synced_folder './support', '/support/', type: 'nfs'
  agent01.vm.synced_folder '/var/lib/tftpboot', '/var/lib/tftpboot', type: 'nfs'

  agent01.vm.network :forwarded_port, host: 42022, guest: 22

# With a public interface
# VM's eth1 is mapped to MGMT and host's MGMT interface
# VM's eth0 is used for internal communication between VM and host
agent01.vm.network :public_network,
  :ip => '172.22.125.31',
  :netmask => '255.255.254.0',
  :dev => 'MGMT',
  :type => 'bridge',
  :mode => 'bridge',
  :mac => mgmt_address

# With a public interface
# VM's eth#{i} is mapped to veth#{i} and host's eth#{i}
# so VM's eth2-5 are mapped to host's eth2-5
(2..5).each do |i|
  agent01.vm.network :public_network,
    :ip => "10#{i}.10#{i}.10#{i}.1",
    :netmask => '255.255.255.224',
    :auto_config => false,
    :dev => "veth#{i}",
    :type => 'bridge',
    :mode => 'bridge',
    :mac => runner_mac_address[i-2]
end

# RAM and CPUs
agent01.vm.provider :libvirt do |domain|
  domain.memory = 1024
  domain.cpus = 1
end

```

Roles

```

---
- hosts: all
  serial: 1
  roles:
    - { role: '../agents/dmos-t2infra-roles/roles/network' }
  vars:
    gateway: 172.22.125.254

- hosts: runner
  serial: 1
  roles:
    - { role: '../agents/dmos-t2infra-roles/roles/test-runner' }
    - { role: '../agents/dmos-t2infra-roles/roles/traffic-pc' }
  vars:
    testbed: 'dmos-testbed-oam-l2-3'
  tests:
    dmos-tests-hal-oam-l2: HEAD
    dmos-tests-l2-switch: HEAD
    dmos-tests-oam-l2: HEAD
    dmos-tests-qos: HEAD
    dmos-testlib-shared-keywords: HEAD

  utilities:
    devops-dmnetconf: '3dec4964e37e28463a70defeb9a977af430ba78f'
    devops-uengine: '99946318ec65830ae2822250a578bf53122376f1'
    dmos-sanitize: '34f9dc4ebfbbeb122dc792dce49d9fda3be5b402'
    dmos-testlib-testbedtopology: HEAD
    dmos-t2infra-testselection: HEAD
    dmos-testlib-softlink: '1c37e26da5aec9b152d3cb39b7e060ca65d28ba2'
    dmos-testlib-pc: HEAD

  traffic_pc_utilities:
    l2tester: '28cddd96e0e6beea849900160ee744218a75bbc3'

```

integration queue

1: build



buildroot



farm de builders:
*10-40 minutos por build
*23 builds em paralelo
*sem virtualização
*muito I/O



2: provision



ANSIBLE



VAGRANT



16 execuções paralelas


3: sanitize





firmware images



4: tests



robot framework



ambientes heterogêneos:
*hardware físico
*simuladores (virtual)
*outros fabricantes
*produtos legados Datacom

tempo:
1 hora (hard limit)
testes: cobertura x tempo
testes confiáveis

5: finalize



git



integrado:

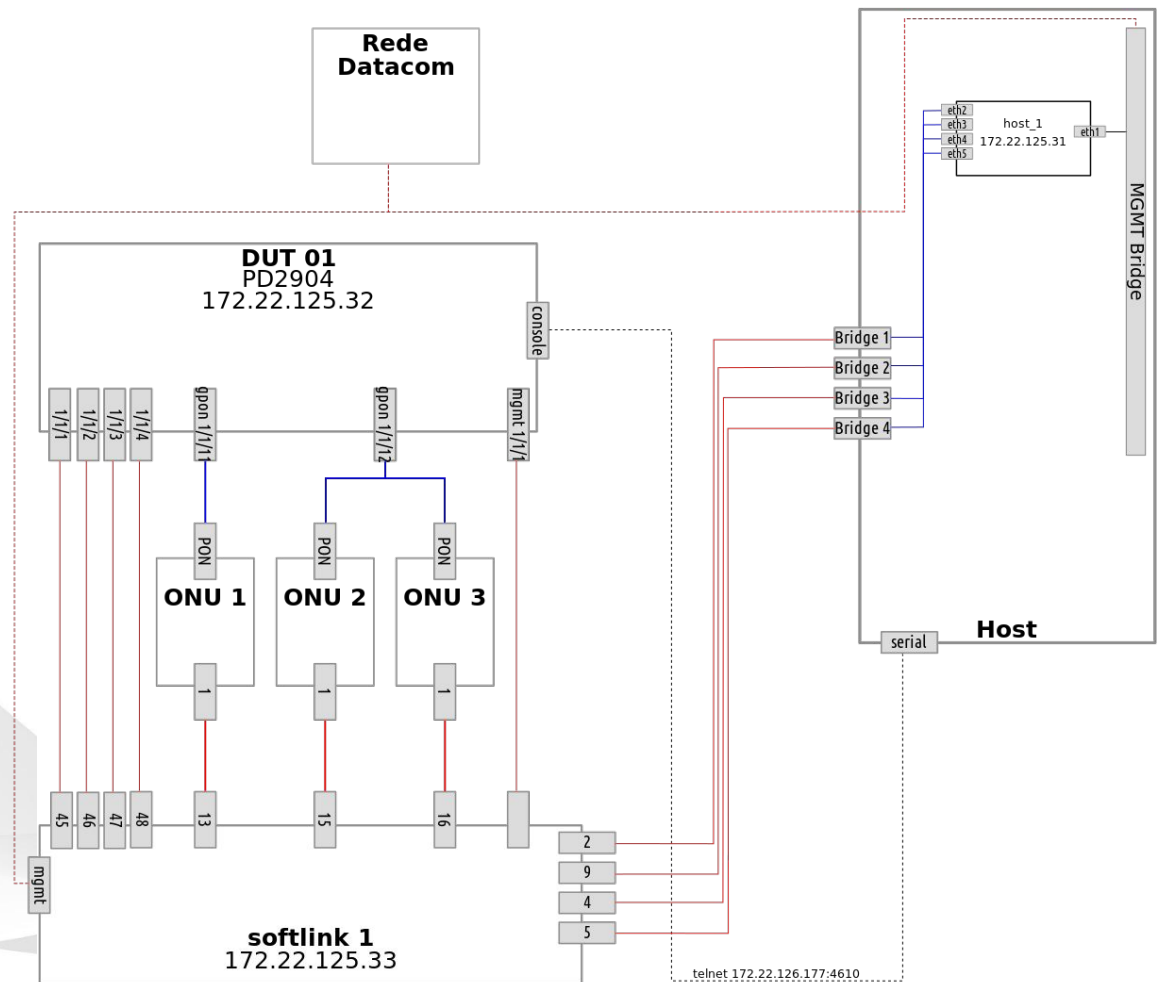


2 horas (+-20 minutos), 7x24



Como é um ambiente de testes na integração?

- **DUT** – Device Under Test
- **Host** – PC que dispara o sequenciamento de testes
- **Softlink** – equipamento de apoio, responsável por eventuais conexões entre os elementos



integration queue

1: build



buildroot



farm de builders:
*10-40 minutos por build
*23 builds em paralelo
*sem virtualização
*muito I/O



2: provision



ANSIBLE



VAGRANT



16 execuções paralelas


3: sanitize




firmware images



4: tests



robot framework



tempo:
1 hora (hard limit)
testes: cobertura x tempo
testes confiáveis

ambientes heterogêneos:
*hardware físico
*simuladores (virtual)
*outros fabricantes
*produtos legados Datacom

5: finalize



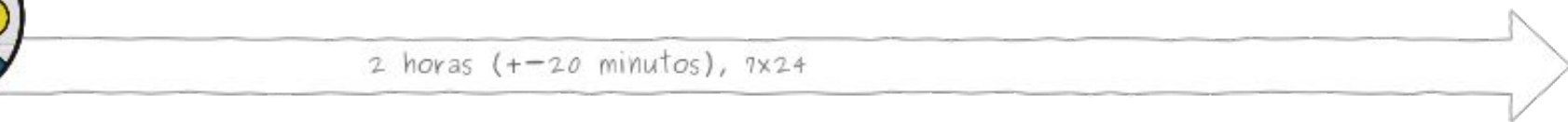
git



integrado:



2 horas (+-20 minutos), 7x24



Como se parece um teste?

```
*** Test Cases ***
```

Generate Remote MEP RDI Alarm

```
[Documentation] Create MEP 1 on switch_dmos1 to communicate with MEP 2 on switch_dmos2.  
... After that destroy MEP 1 on switch_dmos1 and create MEP 3 to communicate  
... with MEP 2 on switch_dmos1. The Remote RDI alarm must be detected on  
... switch_dmos1 for MEP 3.
```

```
Given I have "eth_port1" and "eth_port2" from "switch_dmos1" on VLANs 10,15,20  
And I have "eth_port2" and "eth_port1" from "switch_dmos2" on VLANs 10,15,20  
And data flows from "vm_port1" to "vm_port2" from "host1" on VLAN 20  
And MD "Domain" level 5 is created with MA "Association" with ccm-interval 1s, remote-meps 2, vlan  
And I have MEP 1 "down" attached to "eth_port2" with CCI enabled on vlan 20 from MA "Association"  
And MD "Domain" level 5 is created with MA "Association" with ccm-interval 1s, remote-meps 1,3, v  
And I have MEP 2 "down" attached to "eth_port2" with CCI enabled on vlan 20 from MA "Association"  
And the state of remote MEP 2 in MA "Association" in MD "Domain" on "switch_dmos1" is "Ok"  
And the state of remote MEP 1 in MA "Association" in MD "Domain" on "switch_dmos2" is "Ok"
```

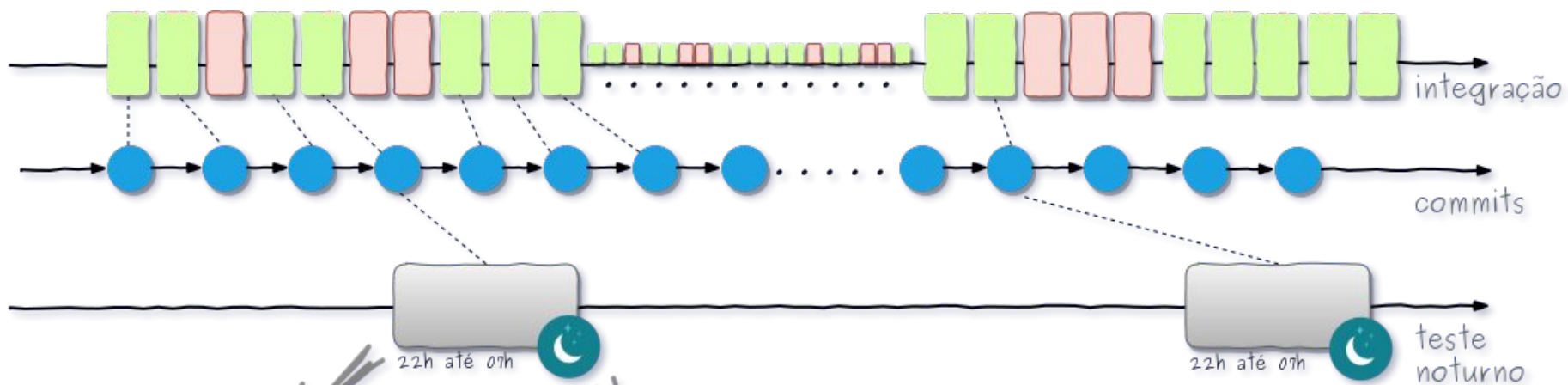
```
When I enter the configuration mode on "switch_dmos1"  
And I destroy MEP 1 from MA "Association" in MD "Domain" on "switch_dmos1"  
And I create a MEP with id 3, direction "down", attached to interface "eth_port2" and primary-vlan  
And I enable CCM generation in MEP 3 in MA "Association" in MD "Domain" on "switch_dmos1"  
And I set MEP 3 lowest fault priority defect to "remote-rdi" in MA "Association" in MD "Domain" on  
And I commit the configuration on "switch_dmos1"
```

```
Then the state of remote MEP 2 in MA "Association" in MD "Domain" on "switch_dmos1" is "Ok"  
And the state of remote MEP 3 in MA "Association" in MD "Domain" on "switch_dmos2" is "Ok"  
And the state of remote MEP 1 in MA "Association" in MD "Domain" on "switch_dmos2" is "Failed"  
And alarm "CFM_RMEP_RDI" is detected in MEP 3 from MA "Association" within 10 seconds on "switch_c  
And alarm "CFM_RMEP_CCM" is detected in MEP 2 from MA "Association" within 10 seconds on "switch_c
```

```
Generate Remote MEP with interface Error Alarm
```

Idioma
Gherkin:
*given...
*when...
*then...





Durante o dia, os desenvolvedores usam o ambiente



~8h úteis de teste (sanitize + outros procedimentos)

falhas aqui abrem bugs de forma automática

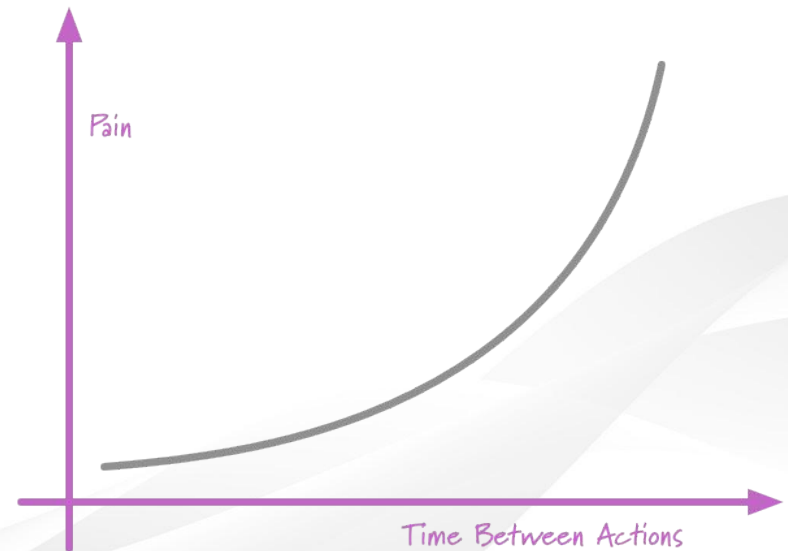


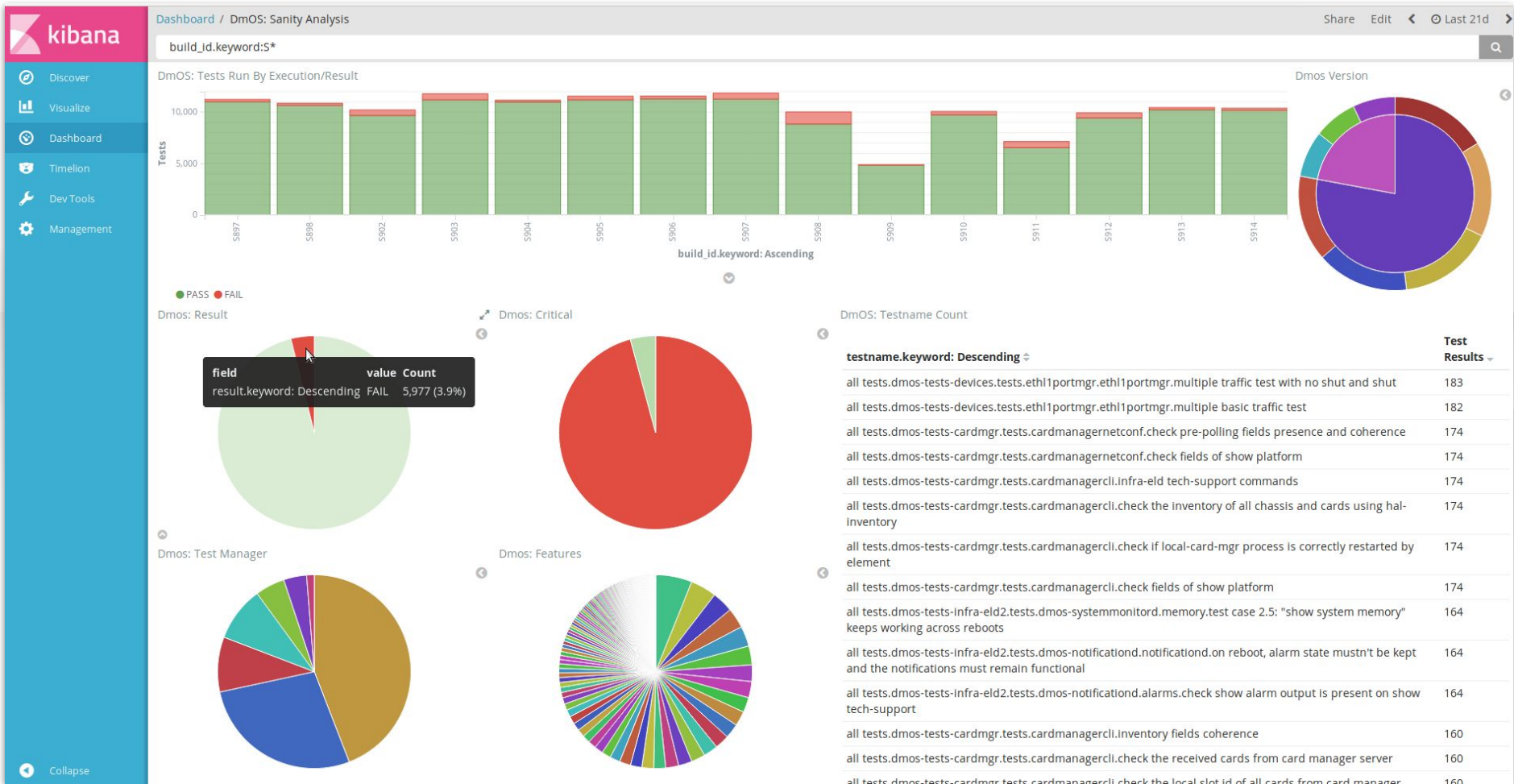
Porém os ambientes de integração são exclusivos! Usados só durante 1 a 2 turnos por semana para manutenção programada. Foco: estabilidade, redução de falsas falhas



If it hurts, do it more often: cuide de seus testes

- A gestão de testes é, e deve ser, uma rotina
- **Testes instáveis devem ser corrigidos**
- **Features instáveis devem reportar defeito**
- **Testes ruins ou mal gerenciados estrangulam projetos (criam gargalos)**





Monitore seus ambientes de teste

ZABBIX Monitoring Inventory Reports Configuration Administration
Dashboard Problems Overview Web Latest data Triggers Graphs Screens Maps Discovery IT services

Screens

All screens / Testbeds - integration

Filter ▾

Testbeds - integration

ZABBIX Monitoring Inventory Reports Configuration Administration
Dashboard Problems Overview Web Latest data Triggers Graphs Screens Maps Discovery

Screens

All screens / Testbeds - fabricio.brito-testbeds

Testbeds - fabricio.brito-testbeds

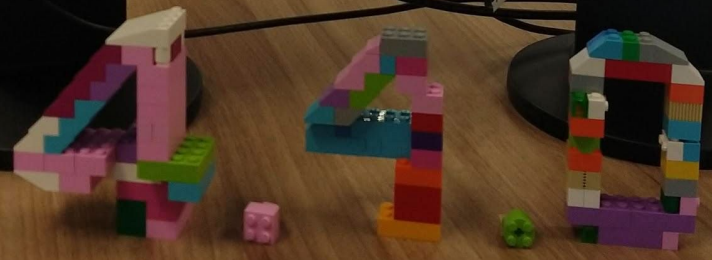
dmos-buildroot integration

Int - P0	Int - P1 cleanup	Int - P1 prermage and deploy	Int - P2 build as512	Int - P2 build pd2901
Int - P2 build pd2901.qemu	Int - P2 build pd2902	Int - P2 build pd2902.qemu	Int - P2 build pd2904	Int - P2 build pd2904.qemu
Int - P2 build pd3401	Int - P2 build pd3401.ft	Int - P2 build pd3402	Int - P2 build pd3402.ft	Int - P2 build pd3402
Int - P2 build pd3520	Int - P2 build pd3520.qemu.x86	Int - P2 build pd3544	Int - P2 build pd3544.qemu	Int - P2 build pd3602
Int - P2 build pd3602.qemu	Int - P2 build pd3604	Int - P2 build pd3604.qemu	Int - P2 build pd3604.qemu	Int - P2 build pd3602
Int - P2 build pd3901	Int - P2 build pd3901.qemu	Int - P2 build pd3804	Int - P2 build pd3804.qemu	Int - P2 build pd3804.six
Int - P2 provision dmos-testbed-4050-qemu-3	Int - P2 provision dmos-testbed-4170-5	Int - P2 provision platform.scope	Int - P2 provision dmos-testbed-4050-5	Int - P2 provision dmos-testbed-4250-1
Int - P2 provision dmos-testbed-4400-qemu-2	Int - P2 provision dmos-testbed-4404-1	Int - P2 provision dmos-testbed-4170-qemu-2	Int - P2 provision dmos-testbed-4610mu2	Int - P2 provision dmos-testbed-4400-6
Int - P2 provision dmos-testbed-cub-mgts-1	Int - P2 provision dmos-testbed-devices-2	Int - P2 provision dmos-testbed-4170-qemu-3	Int - P2 provision dmos-testbed-4170-qemu-2	Int - P3 sanitize dmos-testbed-4050-5
Int - P3 sanitize dmos-testbed-4050-4	Int - P3 sanitize dmos-testbed-4050-qemu-3	Int - P3 sanitize dmos-testbed-4170-5	Int - P3 sanitize dmos-testbed-4170-qemu-2	Int - P3 sanitize dmos-testbed-4250-1
Int - P3 sanitize dmos-testbed-4400-6	Int - P3 sanitize dmos-testbed-4400-qemu-2	Int - P3 sanitize dmos-testbed-4404-1	Int - P3 sanitize dmos-testbed-4610mu2	Int - P3 sanitize dmos-testbed-4400-6
Int - P3 sanitize dmos-testbed-cardmgr	Int - P3 sanitize dmos-testbed-cub-mgts-1	Int - P3 sanitize dmos-testbed-devices-2	Int - P3 sanitize dmos-testbed-4050-qemu-3	Int - P3 sanitize dmos-testbed-4170-qemu-2
Int - P4 pybot dmos-testbed-4050-5	Int - P4 pybot dmos-testbed-4050-6	Int - P4 pybot dmos-testbed-4050-qemu-3	Int - P4 pybot dmos-testbed-4170-5	Int - P4 pybot dmos-testbed-4170-qemu-2
Int - P4 pybot dmos-testbed-4170-1	Int - P4 pybot dmos-testbed-4400-6	Int - P4 pybot dmos-testbed-4400-qemu-2	Int - P4 pybot dmos-testbed-4404-1	Int - P4 pybot dmos-testbed-4610mu2
Int - P4 pybot dmos-testbed-4610mu2-eps2-3	Int - P4 pybot dmos-testbed-cardmgr	Int - P4 pybot dmos-testbed-cub-mgts-1	Int - P4 pybot dmos-testbed-devices-2	Int - P4 pybot dmos-testbed-4400-6
Int - P4 pybot dmos-testbed-qos-2	Int - P5 results	Int - P6 finalizer		

devops-stats

Integration: last 3 weeks (sprint duration)

Failures by phase, and Success



Principais desafios enfrentados

- Necessidade de compilação cruzada para dezenas de plataformas
- Espaço físico, custo de equipamentos e de energia elétrica para ambientes de teste
- Nós construímos o S.O., então é fácil quebrar o HEAD pra valer (exemplo: linux não boota)
- Nós testamos vários tipos de pipeline
 - Para o nosso caso, este deu mais certo que os outros
- Cada caso é um caso: pense, teste a hipótese, e adapte para a sua realidade

Obrigado! Dúvidas?



linkedin.com/in/**eduardo-witter**
witter@datacom.com.br
ewitter@gmail.com

linkedin.com/in/**diego-cardoso-garcia**
dcg.cardoso@gmail.com
[@diego_cgarcia](https://twitter.com/diego_cgarcia)



DATACOM

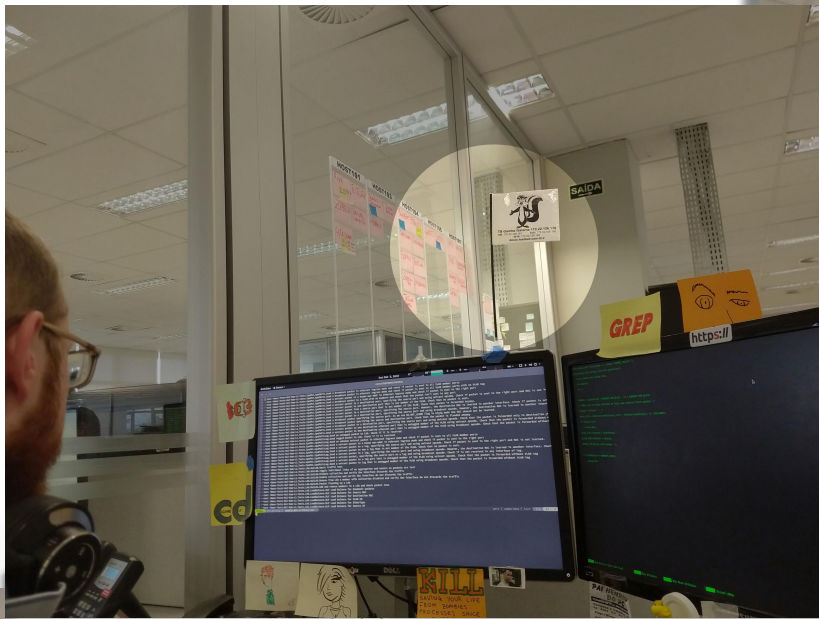
Rua América, 1000 | Eldorado do

Sul | RS | Brasil

+55 51 3933 3000

www.datacom.com.br

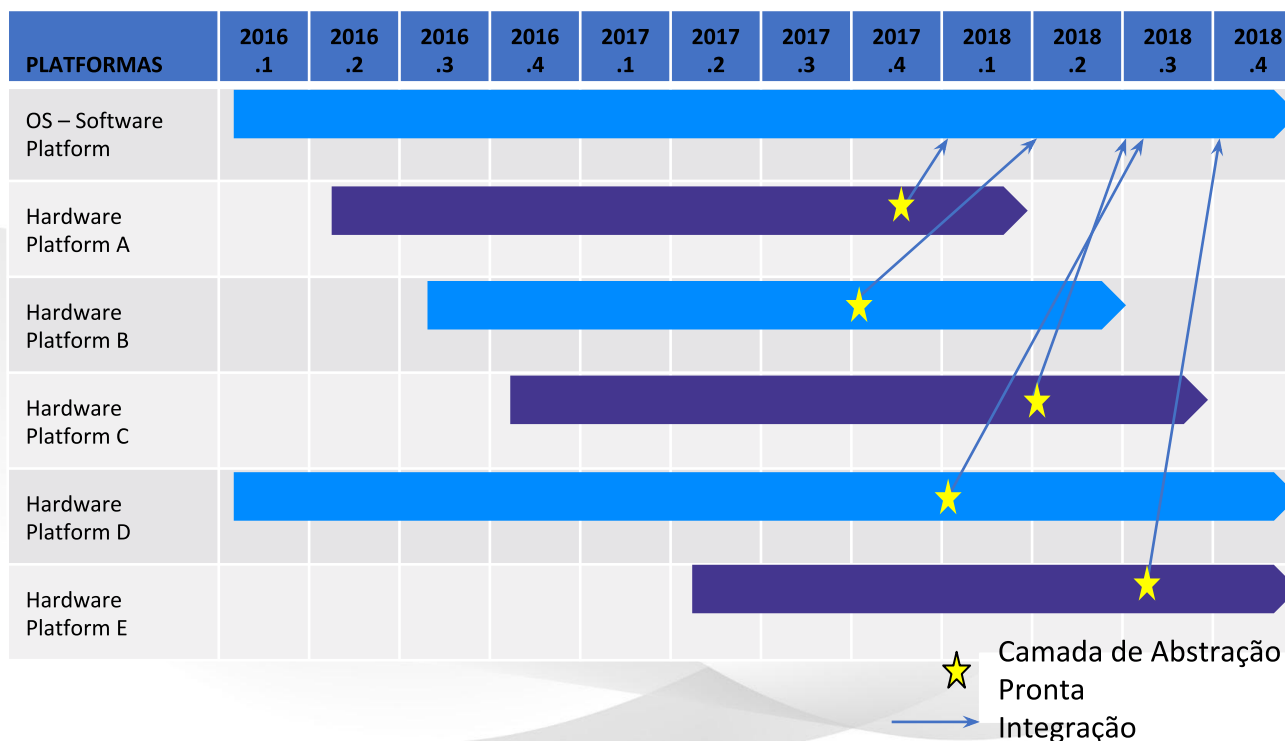
Backup slides



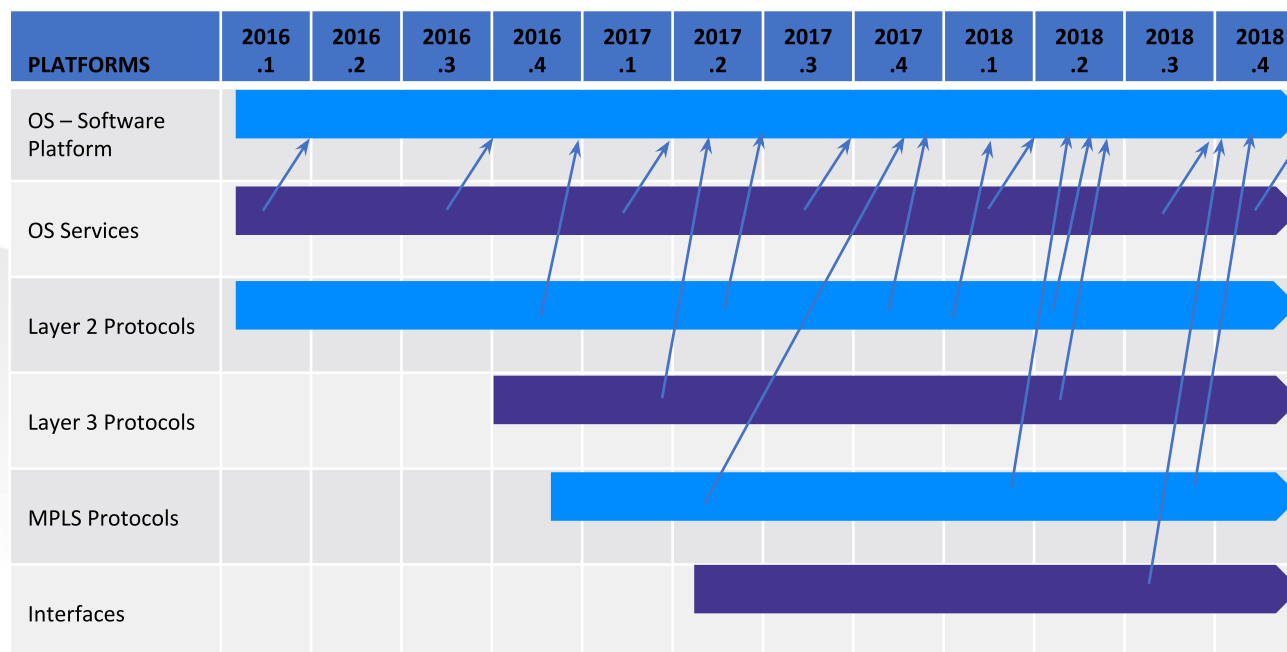
Então, porque desacoplar as coisas

- Plataformas de Software e Hardware são projetos de natureza diferentes
 - Plataformas de Software são projetos contínuos, sem fim – tipicamente Ágil
 - Plataformas de Hardware são projetos preditivos, bem definidos, repetitivos – tipicamente Waterfall
- Produto Datacom: uma Plataforma de Software vs Múltiplas Plataformas de Hardware
 - Matriz multidimensional de plataforma de software, plataformas de hardware e funcionalidades

Sincronização de Plataformas



Integração de Funcionalidades



→ Integração de Componentes

