

“sysadmin-game”

La aventura Sysadmin



David Vargas Ruiz

Email : davidvargas.tenerife@gmail.com

Twitter : @dvarrui

Github : <https://github.com/dvarrui>

Presentación

David Vargas Ruiz

- IES Puerto de la Cruz
- Depto. de Informática
- ASIR/...daw/...asi/dai

=> NO pedagogía

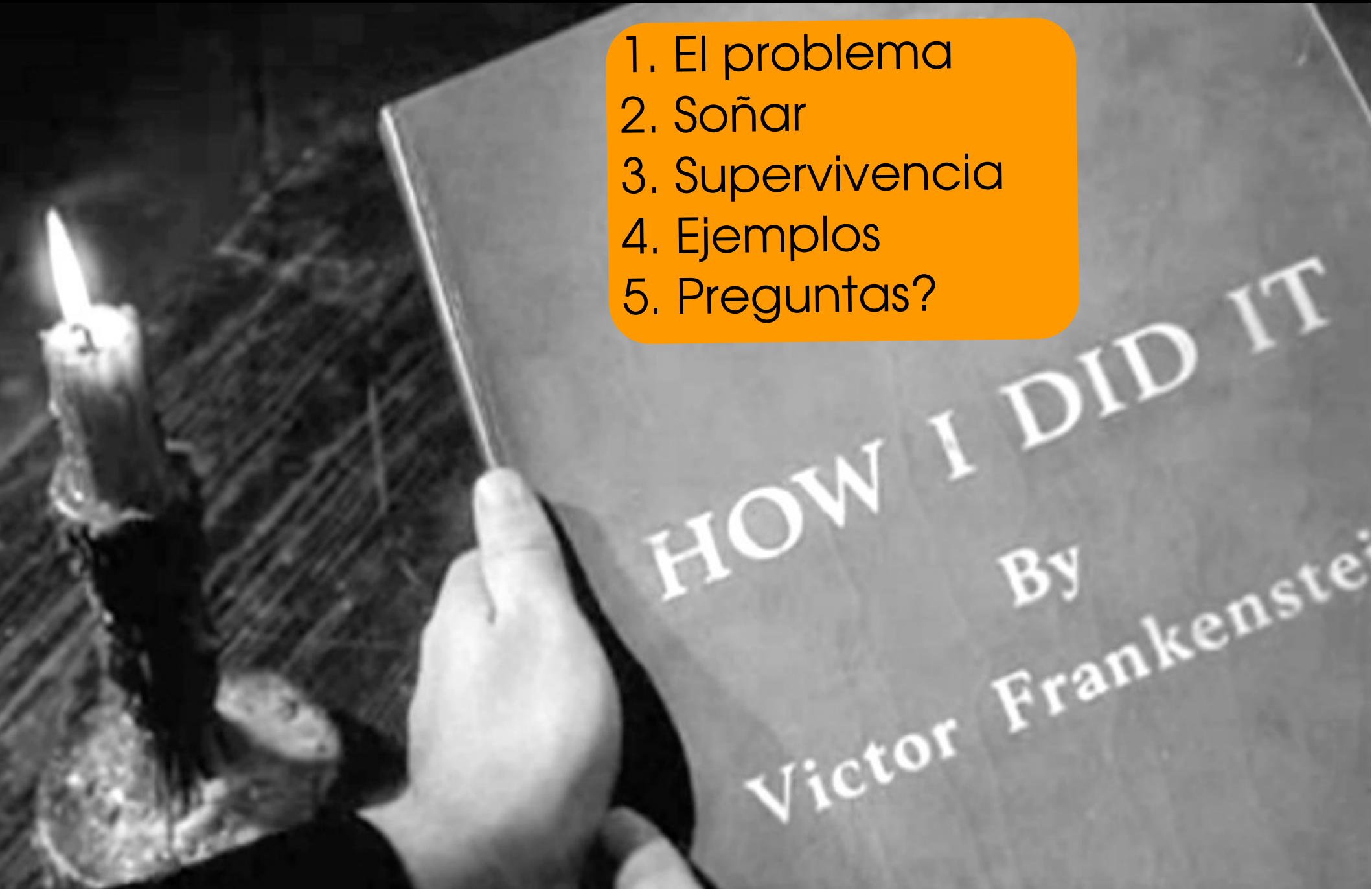
=> SI ingeniería

- * Ruby, Sevops
- * GNU/Linux, Software libre
- * Videojuegos
- * Hard rock, starwars



Agenda

1. El problema
2. Soñar
3. Supervivencia
4. Ejemplos
5. Preguntas?



El problema

El problema

=> Tareas: (1) Entregas, (2) Corregir, y (3) Feedback

=> Estado : (a) Aburrimiento, (b) Consumo de tiempo



Enseñanza congestionada



Potencial Motivador del Trabajo (Fórmula de Hackman & Oldham)

Una tarea es más motivadora cuanto mayor repertorio de habilidades involucre (**variabilidad**). La rutina es la peor enemiga de la motivación.

La motivación también aumenta cuando conocemos en qué consiste la tarea, para qué sirve, por qué es importante y hay que hacerla (**Identidad con la tarea**)


El **significado de la tarea** es el conocimiento de la importancia que tiene para otras personas y para el centro: cómo afectará a los demás, qué ganarán con la realización de esa tarea y cuáles serán las consecuencias de que o se haga.

$$\text{PMT} = \frac{\text{VH} + \text{IT} + \text{ST}}{3} \times \text{Autonomía} \times \text{Feedback}$$

Una tarea motiva más cuando la persona tiene **autonomía** para realizarla,; que pueda tomar decisiones sobre cómo hacerla, qué método emplear...

El **feedback** es imprescindible para que la persona sepa si lo está haciendo bien y, en su caso, qué debe mejorar. Da seguridad y favorece la motivación.

- * Corregir vs preparar clases.
- * Horario de trabajo vs personal.



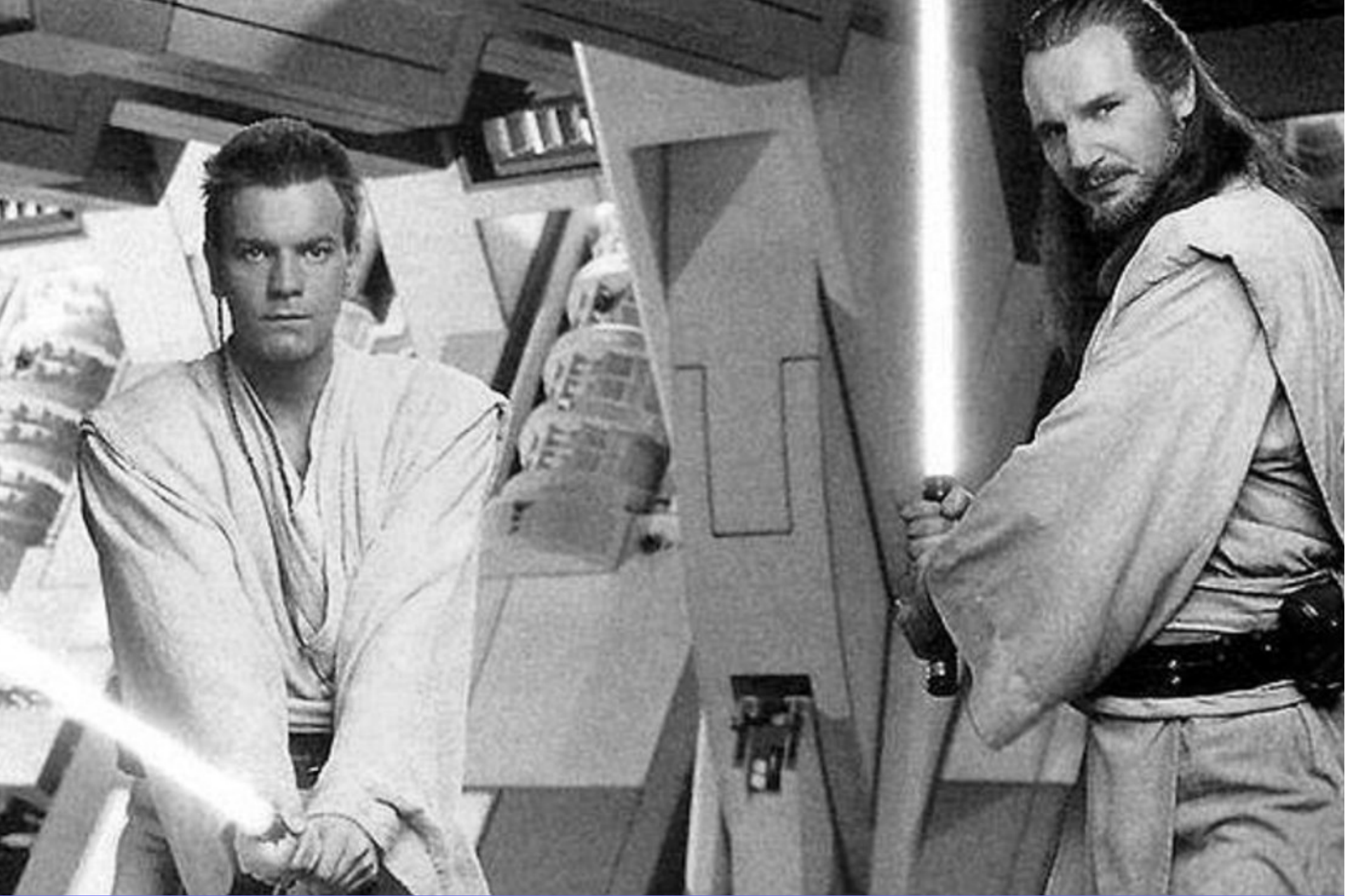
Módulos	4,0	
Alumnos X clase	18,0	
Horas semanales	7,5	
Minutos semanales	450,0	
Horas sem. X módulo	1,9	
	6,3	Minutos



¿Tiene que ser así toda la vida?...



Soñar



Mundo de fantasía

*Un pesimista
ve una calamidad
en toda oportunidad,
un optimista
ve una oportunidad
en toda calamidad.*



ProgramaMe: Concurso de Programación Ciclos Formativos

Estás aquí: / [Regionales 2016](#) / Inicio



Concursos regionales

VI Edición – Curso 2015-2016

Inicio

[Madrid](#)

[Terrassa](#)

[Zaragoza](#)

[On line](#)

Un año más, ¡ProgramaMe está en marcha! Como en ediciones anteriores, se realizarán cuatro regionales, tres de ellos presenciales y uno on-line que tendrá dos convocatorias.

Si aún no sabes qué es ProgramaMe, puedes enterarte [aquí](#). También puedes utilizar la opción *Información General* del menú superior para averiguar más cosas sobre el concurso.

Si ya sabes que es y quieres participar, elige en el mapa la región a la que pertenece tu instituto para obtener más información.



(c) ProgramaMe, 2011-2016

Concurso “*programame*”

Concurso de programación → Unidades de Prueba

Unit Tests

“Programas que chequean programas”

```

1 #!/usr/bin/ruby
2 require "minitest/autorun"
3
4 class TestDemo < Minitest::Test
5   def setup
6     @name = "obiwan"
7   end
8
9   def test_ok
10    assert_equal "OBIWAN", @name.upcase
11    assert_equal 6, @name.size
12  end
13
14 end

```

Programa:

- * Unit Test
- * minitest
- * ruby

Salida:

=> OK

```

david@camaleon:~/proy/tools/sysadmin-game> ./test/test_demo.rb
Run options: --seed 54996

# Running:

.

Finished in 0.000969s, 1032.0248 runs/s, 2064.0495 assertions/s.

1 runs, 2 assertions, 0 failures, 0 errors, 0 skips
david@camaleon:~/proy/tools/sysadmin-game>

```

Unit Test => OK

```
1 #!/usr/bin/ruby
2 require "minitest/autorun"
3
4 class TestDemo < Minitest::Test
5   def setup
6     @name = "obiwan"
7   end
8
9   def test_ok
10     assert_equal "OBIWAN", @name.upcase
11     assert_equal 6, @name.size
12   end
13
14   def test_fail
15     assert_equal "obi-wan", @name
16   end
17
18 end
```

Programa:
* Unit Test
* minitest
* ruby

Salida:
=> 1 failure!

Running:

F.

Finished in 0.001293s, 1546.4495 runs/s, 2319.6743 assertions/s.

1) Failure:
TestDemo#test_fail [./test/test_demo.rb:15]:
Expected: "obi-wan"
Actual: "obiwan"

2 runs, 3 assertions, 1 failures, 0 errors, 0 skips
david@camaleon:~/proy/tools/sysadmin-game>

Unit Test
Fail!



Nuevo punto de vista...

“Me hace falta algo como...”

System Unit Tests?
Host Unit Test?

SYSTEM/HOST UNIT TEST

“Proceso ideal”

- (1) Acceder a las máquinas de nuestros alumnos/concursantes.
- (2) Medir objetivos (targets)
- (3) Hacer un informe.
- (4) Entregar copia al alumno/concursante.

Buscando...



HERRAMIENTAS :

- * Programar
- * Multiplataforma
- * Sysadmin
- * Software libre
- * Redes

Do it yourself!

*Caminante, son tus huellas
el camino y nada más;
Caminante, no hay camino,
se hace camino al andar.
Al andar se hace el camino,
y al volver la vista atrás
se ve la senda que nunca
se ha de volver a pisar.
Caminante no hay camino
sino estelas en la mar.*

(Antonio Machado)

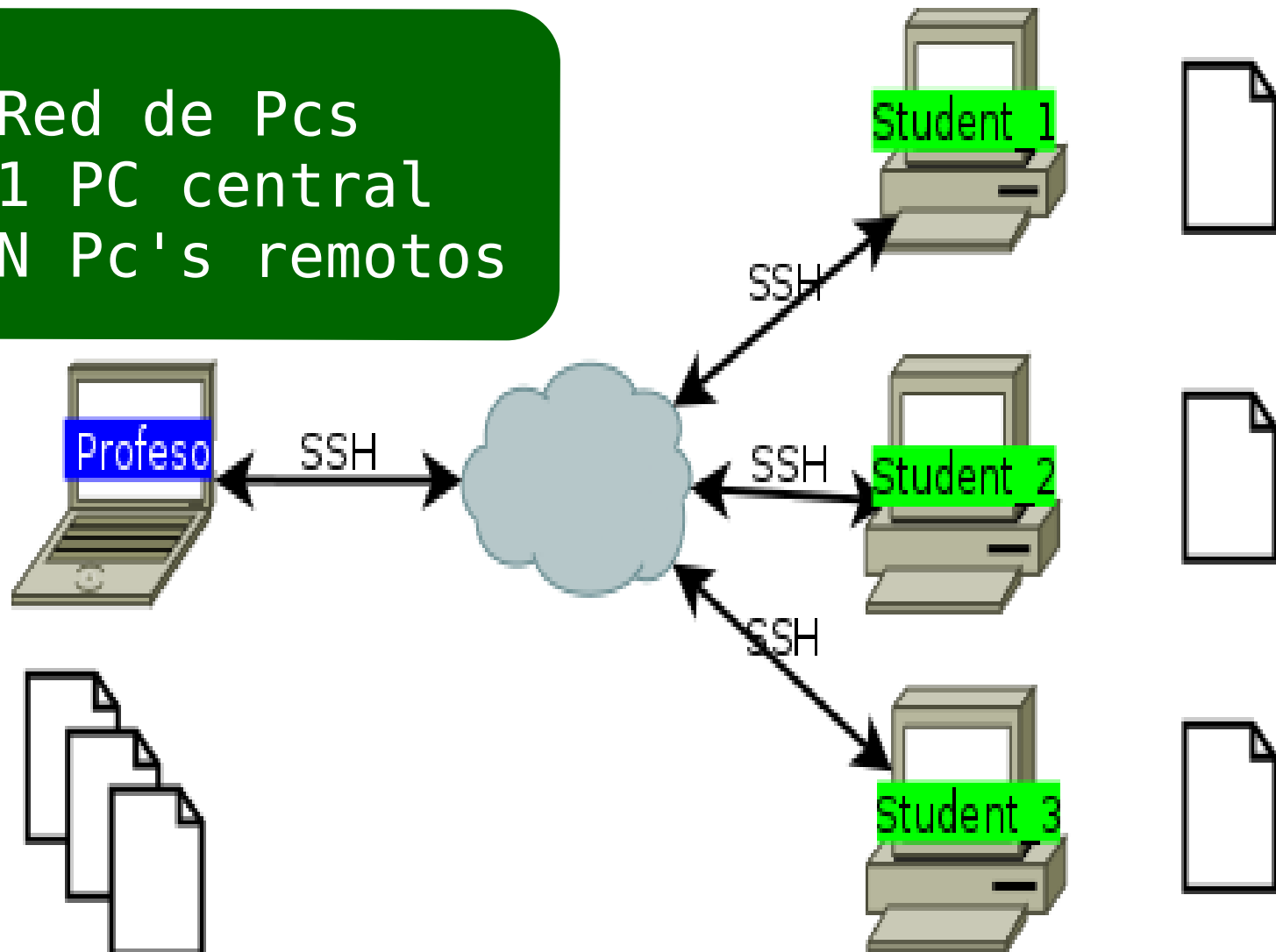
Guía de supervivencia “devops”

Ideas, ideas, ideas, ...

=> Nagios3

=> Puppet/Chef/Ansible

- * Red de Pcs
- * 1 PC central
- * N Pc's remotos



Definiciones

DEFINICIONES DE CONCEPTOS

- (1) target : elemento medible
- (2) task : conjunto de “targets”
- (3) Script : Fichero de texto con “tasks” y “targets”.
- (4) case : Configuración para un alumno
- (5) YAML : Fichero que contiene todos los casos a revisar.

DSL (Language para definir “tasks” y “targets”)

sysadmin-game

GitHub, Inc. (US) <https://github.com/dvarrui/sysadmin-game> ← URL del repositorio

Personal Open source Business Explore Pricing Blog Support This repository Search Sign in Sign up

[dvarrui/sysadmin-game](#) ← Usuario/repositorio

Watch 2 Star 3 Fork 1

Code Issues 0 Pull requests 0 Wiki Pulse Graphs

Check a targets list into a group of hosts (students classwork), and evaluate the final results of every one.

0 releases 1 contributor

Find file HTTPS <https://github.com/dvarrui/sysadmin-game> Download ZIP

Latest commit 6f8b925 3 hours ago

words 8 days ago

3 hours ago

3 hours ago

Descargar el proyecto

INSTALACIÓN

- (1) Instalar ruby y rake
- (2) Descargar
"git clone URL"
- (3) "cd sysadmin-game"
- (4) "rake install_gems"

example-01

```
1
2 task "Create user obiwan" do
3
4   target "Checking user <obiwan> using commands"
5   goto :localhost, :exec => "id obiwan| wc -l"
6   expect result.equal(1)
7
8 end
9
10 start do
11   show
12   export
13 end
```

- * target
Definir objetivo
- * goto
Ejecutar comando
- * expect
Evaluar resultado

```
1 --
2 :global:
3 :cases:
4 - :tt_members: student1
```

Fichero de configuración

Adelante vídeo 1...



example-02

```
1
2 task "Create user with your name" do
3
4   target "Checking user <" + get(:username) + "> using count! method"
5   goto :localhost, :exec => "id #{get(:username)}"
6   expect result.count!.eq 1
7
8 end
9
10 start do
11   show
12   export
13 end
14
```

* get

Lee fichero de configuración
Devuelve el valor del
parámetro para cada caso.

```
1 --
2 :global:
3 :cases:
4 - :tt_members: Student-name-1
5   :username: root
6 - :tt_members: Student-name-2
7   :username: darth-maul
```

Configuración con 2 casos.

Cada caso con 2 Variables:

- * tt_members
- * username

```
david@camaleon:~/proy/tools/sysadmin-game> ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp2s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast state DOWN group default qlen 1000
    link/ether 50:b7:c3:06:e2:d1 brd ff:ff:ff:ff:ff:ff
3: wlp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether c8:f7:33:18:ba:be brd ff:ff:ff:ff:ff:ff
    inet 172.18.99.154/16 brd 172.18.255.255 scope global dynamic wlp1s0
        valid_lft 85449sec preferred_lft 85449sec
    inet6 fe80::caf7:33ff:fe18:babe/64 scope link
        valid_lft forever preferred_lft forever
david@camaleon:~/proy/tools/sysadmin-game> |
```

Comando => "ip a"


```
david@camaleon:~/proy/tools/sysadmin-game> ip a |grep 'inet '  
inet 127.0.0.1/8 scope host lo  
inet 172.18.99.154/16 brd 172.18.255.255 scope global dynamic wlp1s0  
david@camaleon:~/proy/tools/sysadmin-game> ip a |grep 'inet ' | wc -l  
2  
david@camaleon:~/proy/tools/sysadmin-game> |
```

```
goto :host, :exec => "ip a"  
expect result.grep!("inet ").count!.eq 2
```

```
expect result.find!("inet ").count!.eq 2  
expect result.grep!("inet ").size!.eq 2  
expect result.find!("inet ").size!.eq 2
```

example-03.rb

```
1
2 task "Configure hostname and DNS server" do
3
4   target "Hostname is <" + get(:host1_hostname) + ">"
5   goto :host1, :exec => "hostname -f"
6   expect result.equal(get(:host1_hostname))
7
8   target "DNS Server OK"
9   goto :host1, :exec => "host www.google.es"
10  expect result.find!('has address').count!.greater(0)
11
12 end
13
14 task "Create user with your name" do
15
16   target "Exist user <" + get(:username) + ">"
17   goto :host1, :exec => "id #{get(:username)}"
18   expect result.count!.eq 1
19
20 end
21
22 start do
23   show
24   export
25 end
```

- * Script con 2 tareas y 3 objetivos.
- * Ejecución remota (goto :host1, :exec => "...")
- * Datos en pantalla (show)
- * Informes con los resultados (export)

example-03.yaml

```
1 ---
2 :global:
3   :host1_username: root
4   :host1_password: profesor
5 :cases:
6 - :tt_members: darth-maul
7   :host1_ip: 192.168.1.200
8   :host1_hostname: sith.starwars
9   :username: dmaul
10 - :tt_members: r2d2
11   :host1_ip: 192.168.1.201
12   :host1_hostname: robot.starwars
13   :username: r2d2
14 - :tt_members: obiwan kenobi
15   :host1_ip: 192.168.1.202
16   :host1_hostname: jedi.starwars
17   :username: obiwan
```

FICHERO DE CONFIGURACIÓN

Variables/parámetros:

=> **globales**

- * :host1_username
- * :host1_password

=> **De cada caso**

- * :tt_members
- * :host1_ip
- * :host1_hostname
- * :username

DEFINIR UN HOST (luna)

- * :luna_ip
- * :luna_username
- * :luna_password

Adelante vídeo 3...



example-03/resume.txt

```
4 | tt_scriptname | ./docs/examples/example-03.rb |
5 | tt_configfile | ./docs/examples/example-03.yaml |
6 | host1_username | root |
7 | host1_password | profesor |
8 | tt_testname | example-03 |
9 | tt_sequence | false |
```

```
10 +-----+
```

11 TARGETS HISTORY

```
12 - Case_01 => 33 ? darth-maul
13 - Case_02 => 0 ? r2d2
14 - Case_03 => 100 obiwan kenobi
```

Resumen de los casos
y su puntuación final

15 FINAL VALUES

```
16 +-----+
17 | start_time | 2016-03-27 13:47:27 +0100 |
18 | finish_time | 2016-03-27 13:47:33 +0100 |
19 | duration | 5.883090953 |
```

```
20 +-----+
```


example-03/case-01.txt

```
5 | host1_hostname | sith.starwars |
6 | username      | dmaul      |
7 | tt_skip       | false      |
8 +-----+-----+
```

9 TARGETS HISTORY

```
10 - TASK: Configure hostname and DNS server
```

```
11 01 (0.0/1.0)
12     Description : Hostname is <sith.starwars>
13     Command      : hostname -f
14     Expected     : sith.starwars (String)
15     Result       : curso1516.ies (String)
```

```
16 02 (1.0/1.0)
17     Description : DNS Server OK
18     Command      : host www.google.es| grep 'has address'| wc -l
19     Expected     : Greater than 0 (String)
20     Result       : 16 (String)
```

```
21 - TASK: Create user with your name
```

```
22 03 (0.0/1.0)
23     Description : Exist user <dmaul>
24     Command      : id dmaul |wc -l
25     Expected     : 1 (Fixnum)
26     Result       : id: dmaul: No existe ese usuario (String)
```

27 FINAL VALUES

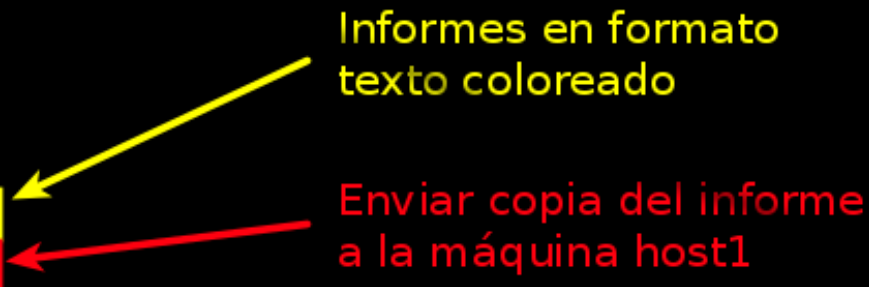
```
28 +-----+-----+
```

Objetivos
FALLADOS

Objetivo
conseguido

example-04.rb

```
1
2 # encoding: utf-8
3
4
5 task "Configure hostname and DNS server" do
6   target "Hostname is <"+get(:host1_hostname)+">"
7   goto :host1, :exec => "hostname -f"
8   expect result.equal(get(:host1_hostname))
9
10  target "DNS Server OK"
11  goto :host1, :exec => "host www.google.es| grep 'has address'| wc -l"
12  expect result.greater(0)
13 end
14
15 task "Create user with your name" do
16   target "Exist user <"+get(:username)+">"
17   goto :host1, :exec => "id #{get(:username)} |wc -l"
18   expect result.equal(1)
19 end
20
21 start do
22   show
23   export :format => :colored_text
24   send :copy_to => :host1
25 end
```



Informes en formato texto coloreado

Enviar copia del informe a la máquina host1

Más instrucciones...

- (1) result permite transformar el resultado.
result.find!("filter")
result.count!
result.eq value
result.neq value
result.gt value
result.lt value
...
- (2) unique establece valores únicos.
unique "MAC", result.value
...
- (3) log registra un texto en el informe.
log("Hola Mundo!")
log("Error 42", :error)

Próximas versiones

- * Mejorar la documentación (ES/EN)
- * "goto/exec" => Integración con Chef/Puppet
- * "report" => Formatos HTML, XML, CSV.
- * Multithreading de tareas.
- * Evaluación por tareas/objetivos
- * Unidades de prueba para el código.
- * etc.

A black and white background image featuring Star Wars characters. In the center, Darth Vader is shown from the chest up, wearing his iconic black armor and hood. To his left and right, Stormtroopers are visible, some holding blasters. The scene is set in a dark, industrial environment with some light sources in the background.

¡Muchas gracias!

David Vargas Ruiz

Email : davidvargas.tenerife@gmail.com

Twitter : @dvarrui

Github : <https://github.com/dvarrui>