

# Funcionalidades y comprobación de la orden MPSTAT de forma experimental



# Especificación de objetivos

- Breve introducción de la orden mpstat
- Campos de la orden mpstat y su fiabilidad
- Hardware
- Carga de trabajo
- Diseño del experimento
- Análisis e interpretación del experimento

# ¿Qué es mpstat y para qué sirve?

Mpstat muestra actividades de los procesadores

Proporciona un promedio de las actividades de todos los núcleos en conjunto

Mpstat dispone de varias opciones, de las cuales:

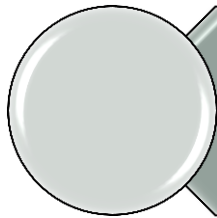
- -V. Muestra la versión que tenemos instalada
- -P. Opción para elegir sobre qué procesador queremos que se muestren las estadísticas. Si ponemos ALL, se nos muestran las estadísticas de todos los procesadores.
- -A. Es equivalente a mpstat -u, -l ALL ó -P ALL

# ¿Qué campos tiene mpstat?

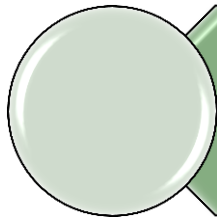
Dispone de 8 campos, de los cuales nos centraremos en 3:

- %usr: Porcentaje de uso en el nivel de usuario
- %iowait: Porcentaje del tiempo esperando operaciones de E/S
- %idle: Tiempo en la que la CPU ha estado sin trabajo y sin estar esperando E/S

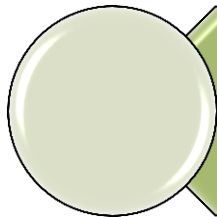
# ¿Es fiable dicha orden?



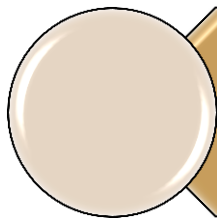
Si y no, como todo muestreo tiene sus errores



En este caso la CPU se recoge mediante la interrupción timer que mira en ese momento qué está haciendo la CPU



Ejecutamos la orden htop para saber los procesos que se están ejecutando y ver si hay ruido o no cuando empezamos a medir con mpstat



Por tanto, no significa que los datos no sean fiables, sino que, como en todo muestreo, hay un porcentaje de error a tener en cuenta

# Hardware

Procesador: Intel core i7-4720HQ CPU @ 2.60 GHz

Cores activos en la máquina virtual: 1 core

Memoria RAM: 4GB

Software de virtualización: VMWare workstation 12 pro

Sistema operativo: Ubuntu 15.04

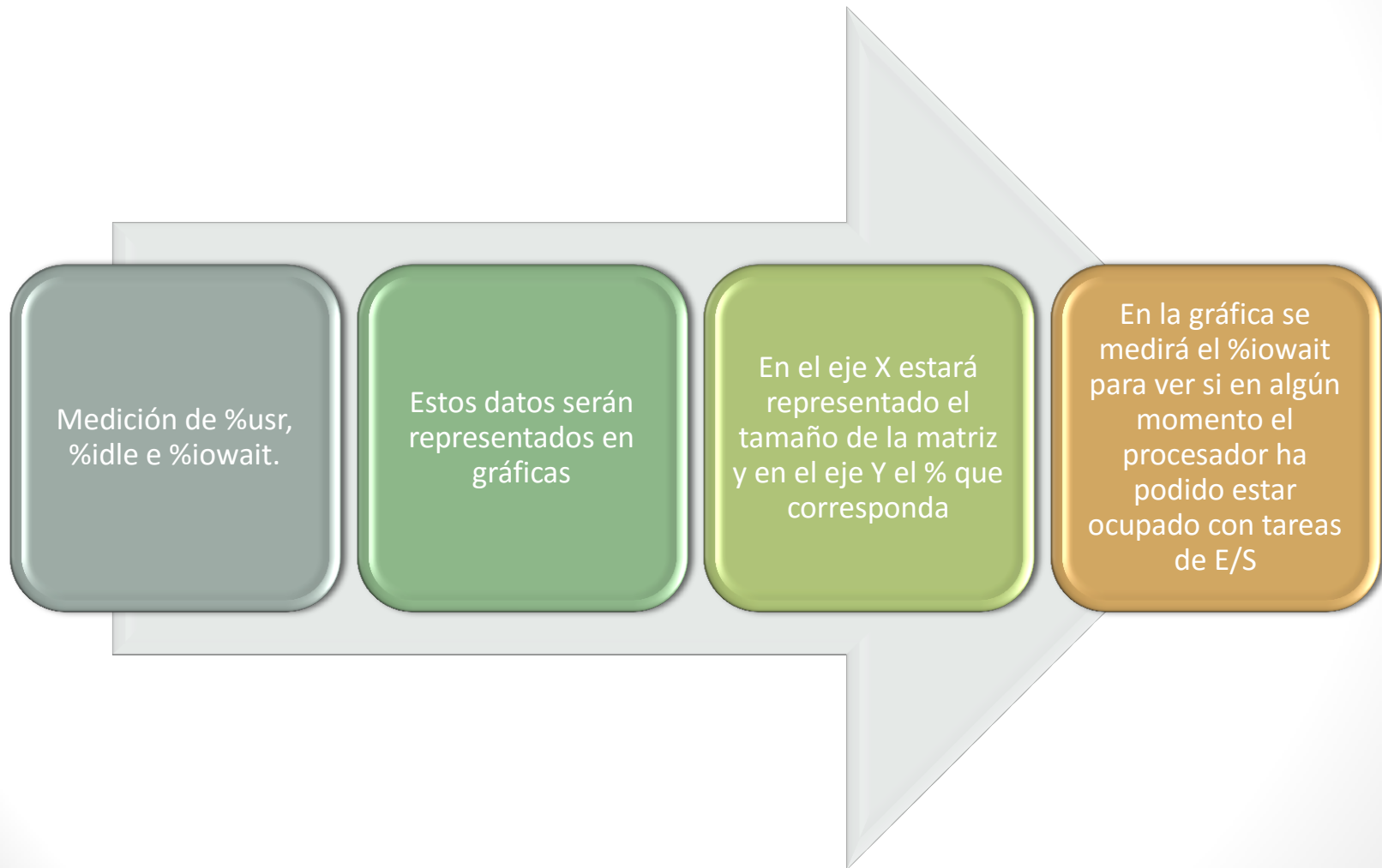
Disco duro: 15GB VMWare Virtual

Resolución de pantalla: 1920x1080

# Carga de trabajo

Para aumentar la carga de trabajo del sistema utilizamos un programa de multiplicación de matrices cuadradas en C++, empezando con matrices de orden 500x500 hasta 1500x1500, incrementando el tamaño de la matriz en 200 para cada ejecución

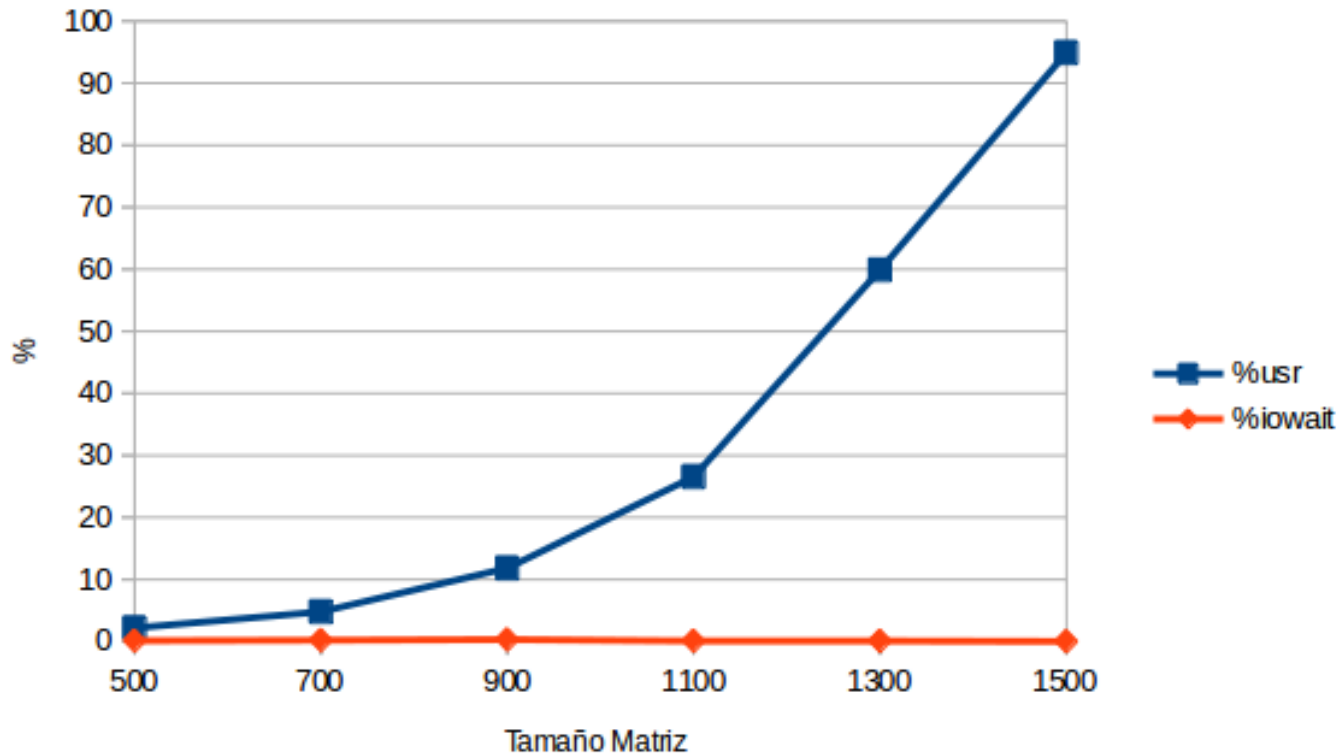
# Diseño del experimento





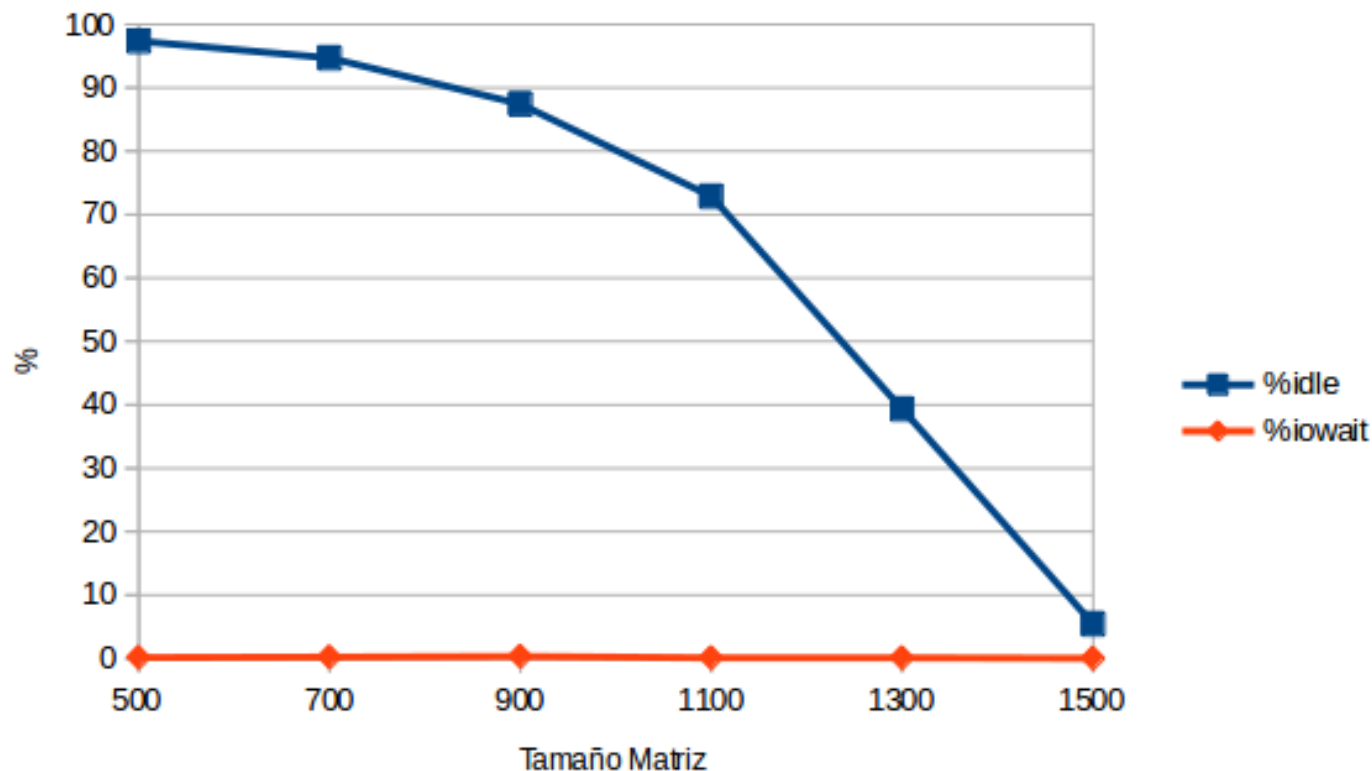
# Análisis e interpretación

- %usr

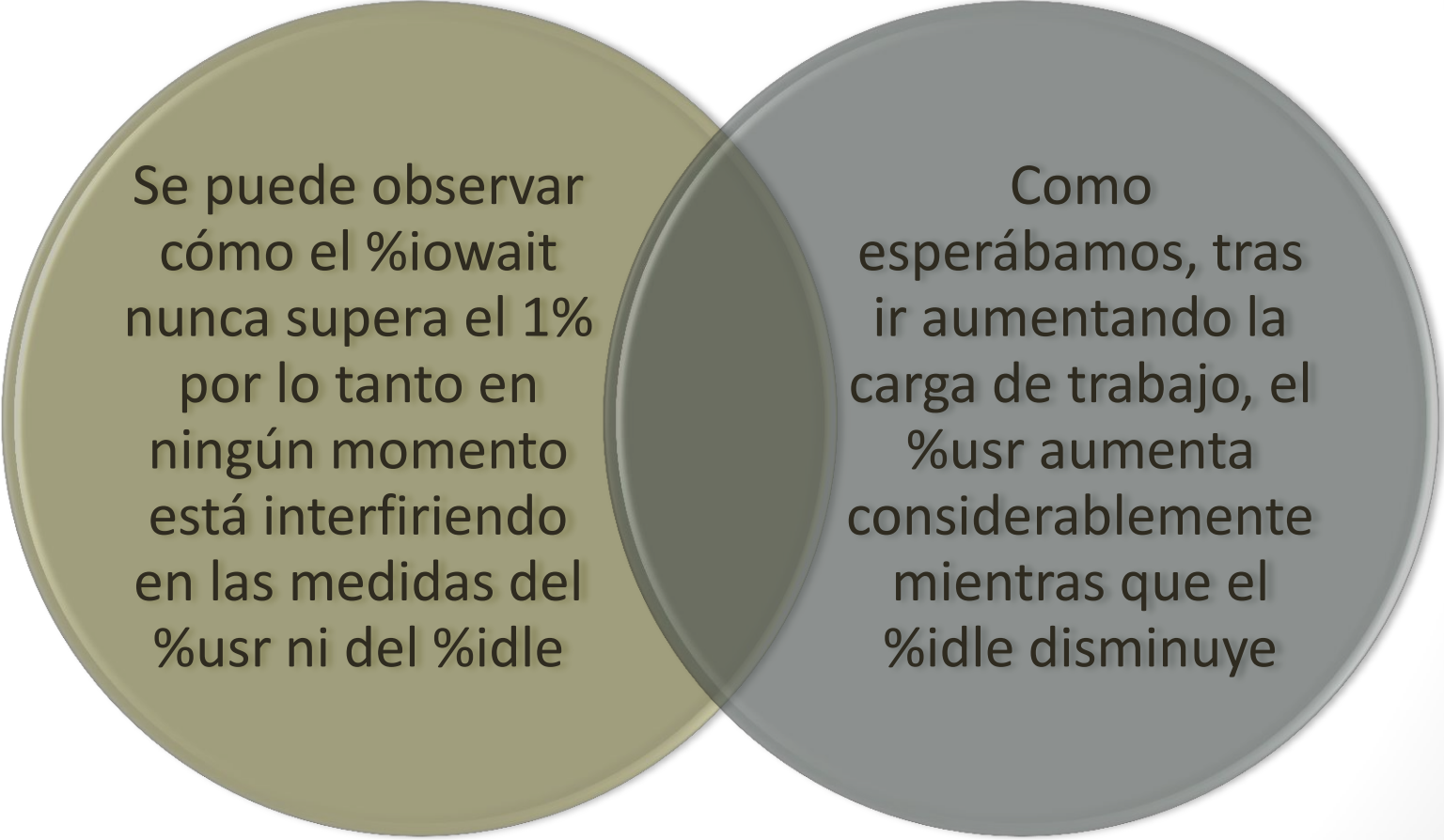


# Análisis e interpretación

- %idle



# Análisis e interpretación



Se puede observar cómo el %iowait nunca supera el 1% por lo tanto en ningún momento está interfiriendo en las medidas del %usr ni del %idle

Como esperábamos, tras ir aumentando la carga de trabajo, el %usr aumenta considerablemente mientras que el %idle disminuye

# Referencias

[http://linuxcommand.org/man\\_pages/mpstat1.html](http://linuxcommand.org/man_pages/mpstat1.html)

<http://systemadmin.es/2009/11/como-leer-el-uso-de-cpu-en-el-top-o-el-mpstat>

<http://linux.die.net/man/1/htop>

Prácticas de Sistemas Operativos

Prácticas de Arquitectura de Computadores