

Software Deployment

Despliegue/Implementación de Aplicaciones

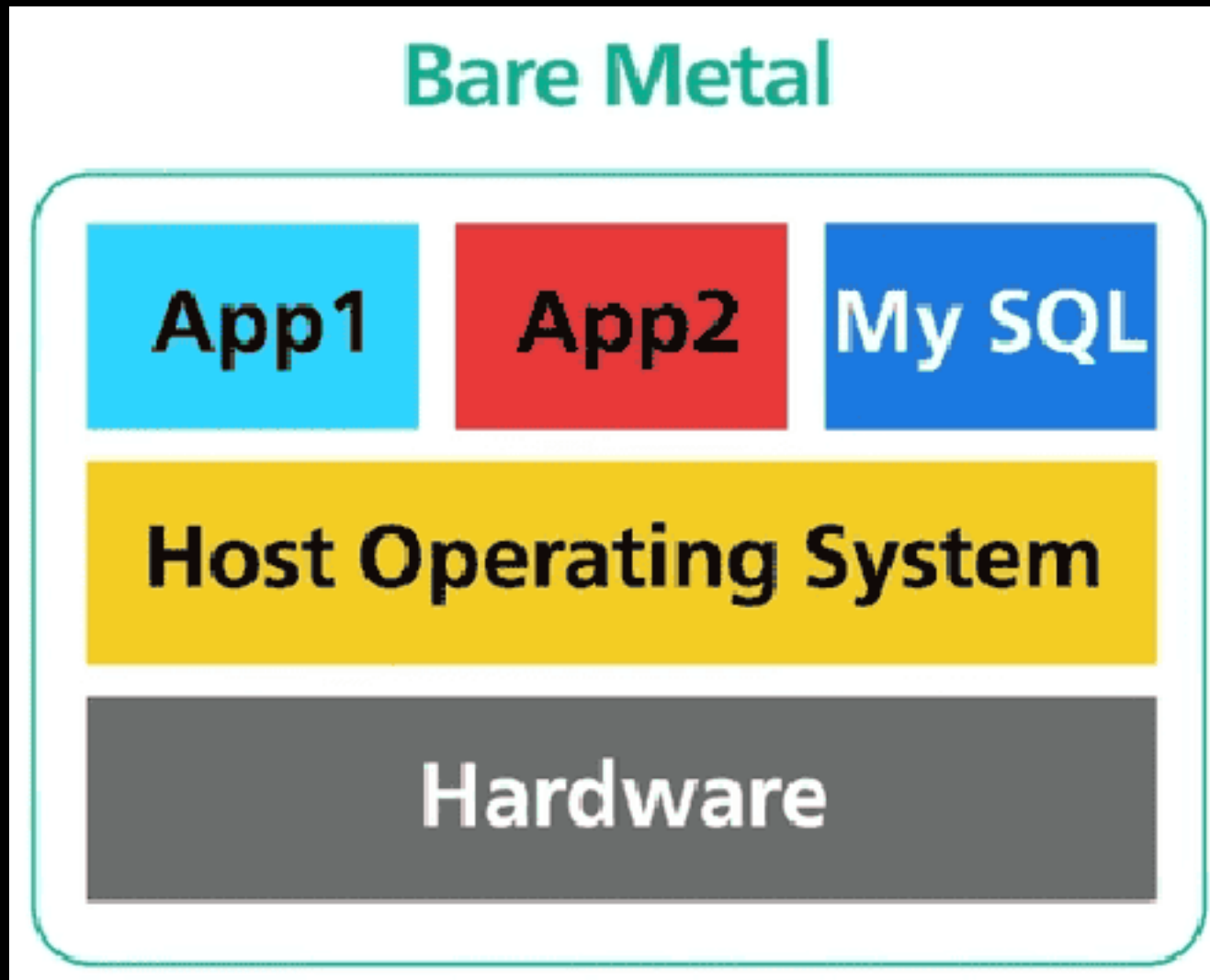


WiFi: ISFDyTN166_VOTAR

Pass: votar231011

www.vot.ar

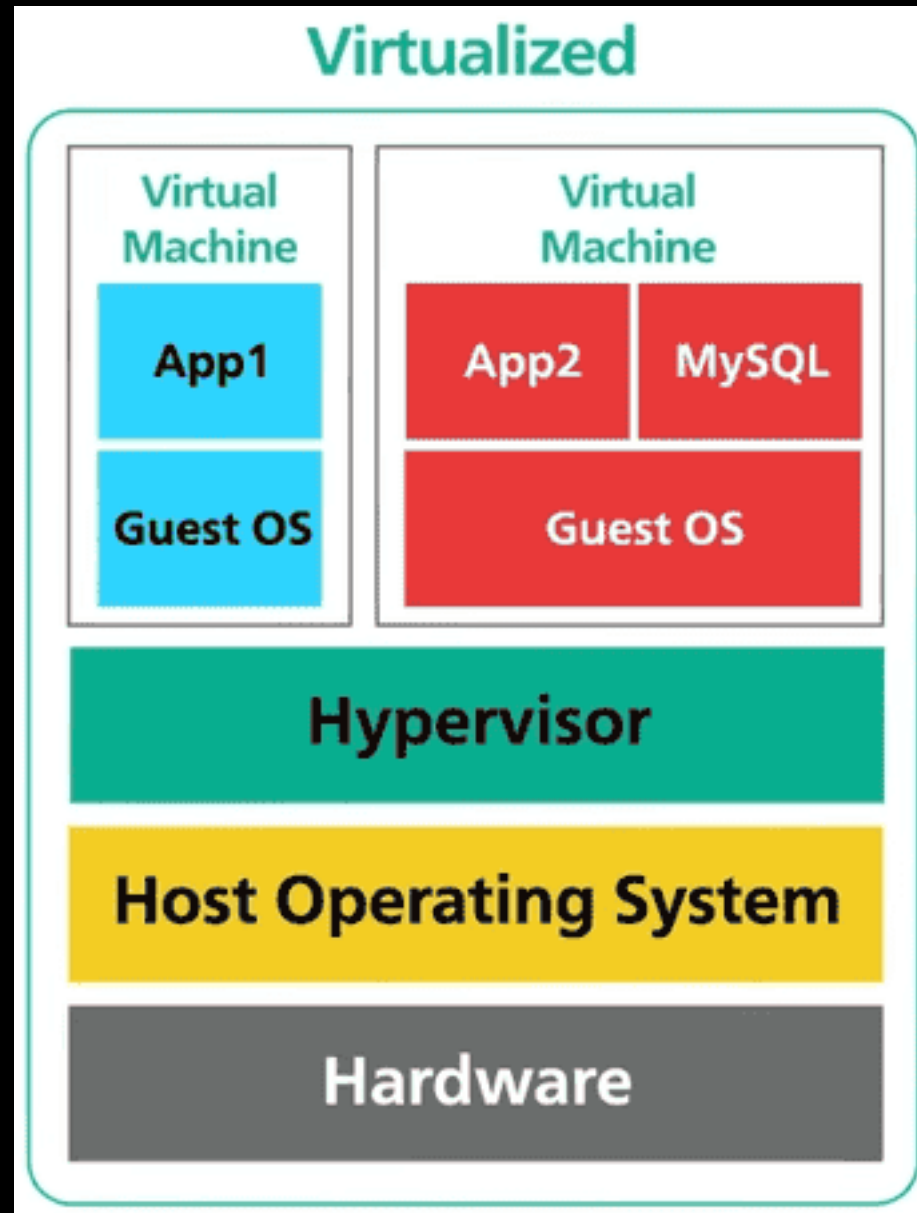
Enfoque Tradicional



Enfoque Tradicional

- ✗ Sin aislamiento de recursos.
- ✗ El alto consumo de recursos de una aplicación puede quebrar ("crash") todo el sistema.
- ✗ Difícil de escalar.
- ✗ Altos periodos de inactividad.
- ✗ Caro.

Virtualización



Virtualización

- ✓ Mejor utilización de recursos.
- ✓ Las aplicaciones están aisladas.
- ✗ Las imágenes de los Sistemas Operativos son "pesadas" y el proceso de inicio ("boot-up") es lento.
- ✗ Las aplicaciones no son portables.
- ✗ Difícil de escalar.
- ✗ Puede tornarse caro.

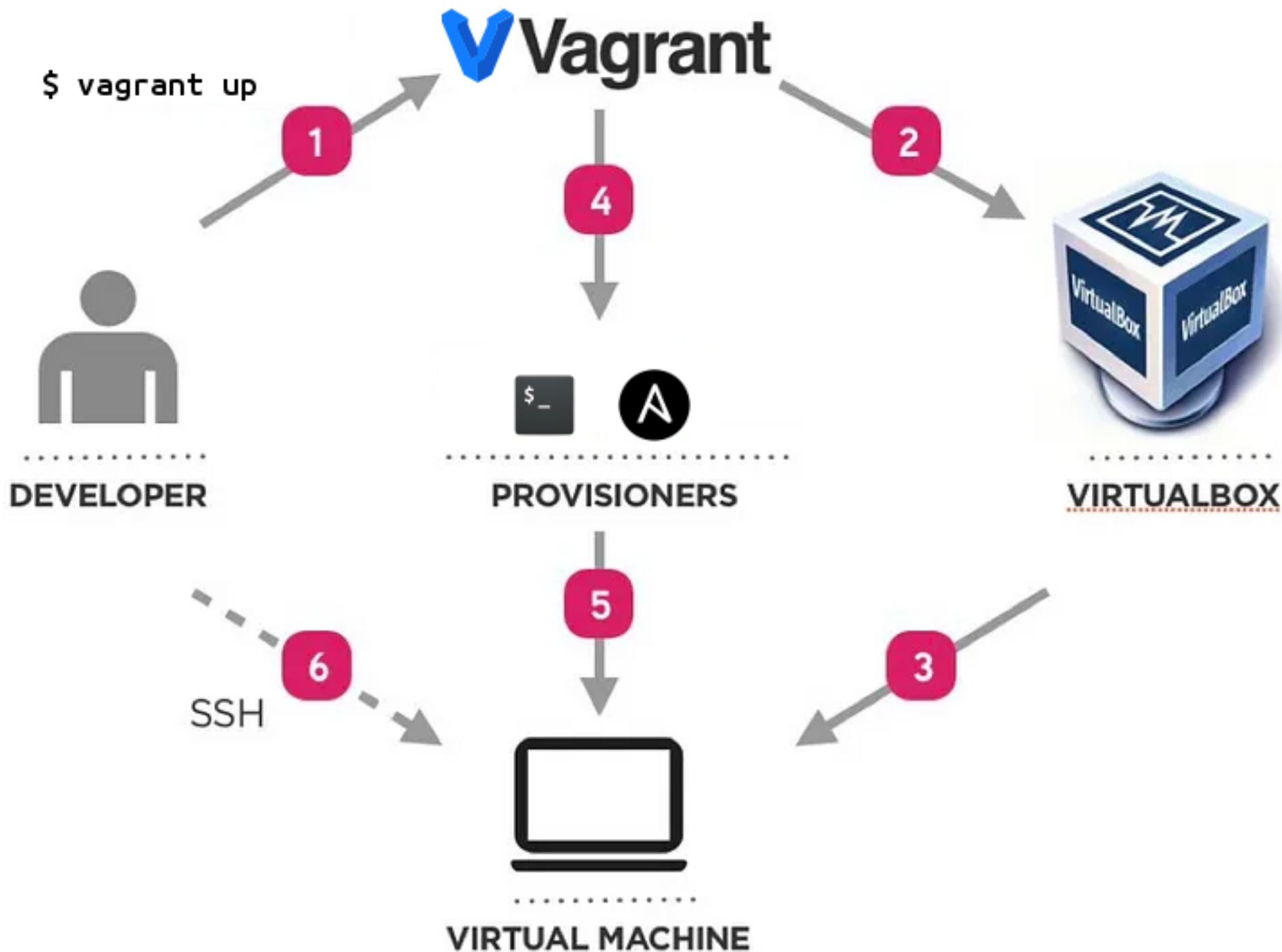
Virtualización

Herramientas para un entorno local de desarrollo

<https://www.virtualbox.org/>

<https://www.vagrantup.com/>

<https://www.ansible.com/>

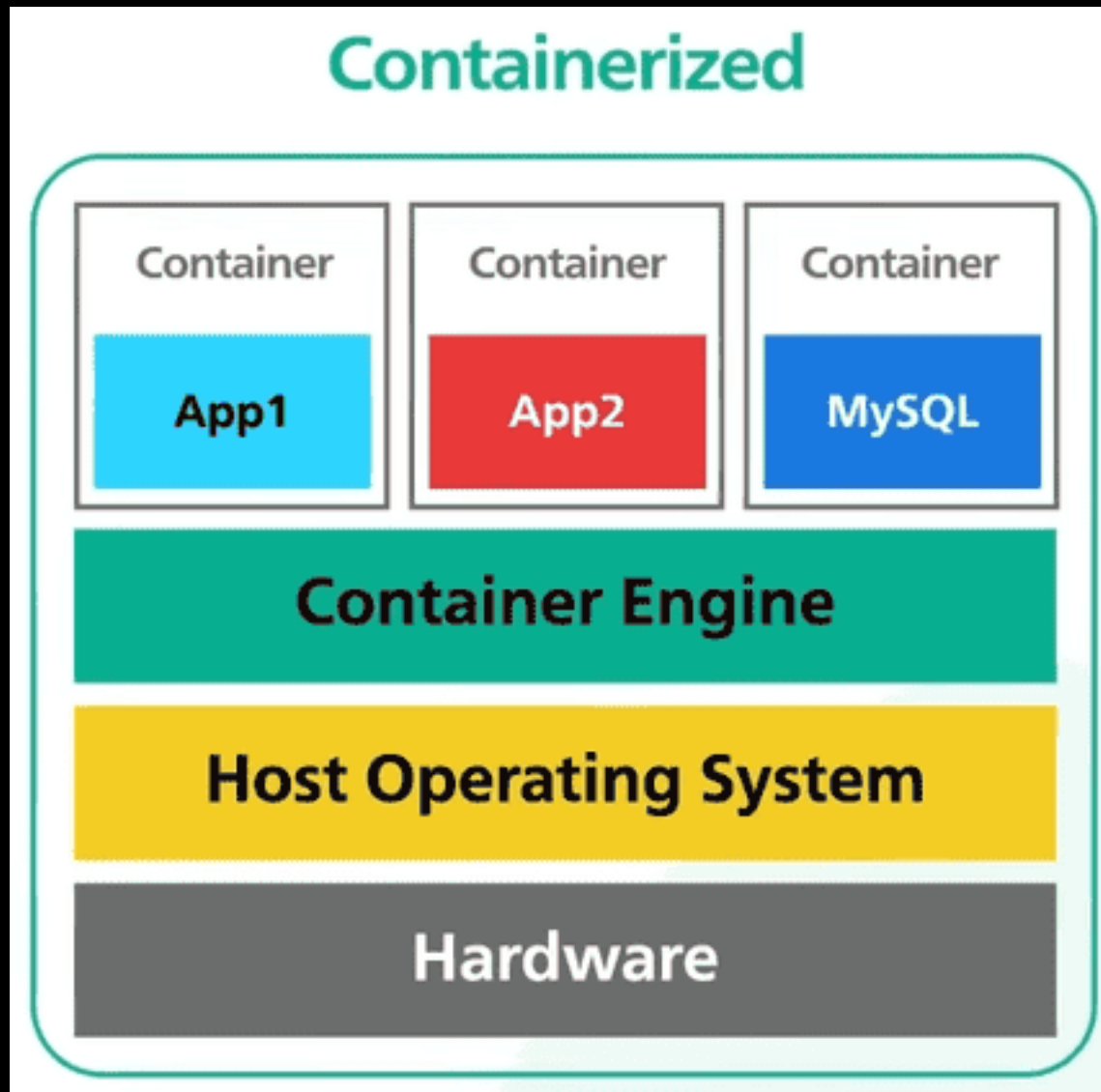


Containerization

(contenedorización)

Virtualización a nivel de sistema operativo (o aplicación) que facilita el desarrollo y estandariza el "empaquetado", distribución y ejecución de las aplicaciones. ([Wikipedia](#))

Containerization (contenedorización)



Containerization

(contenedorización)

- ✓ Imágenes "livianas" e inicio rápido.
- ✓ Portables.
- ✓ Económicos.
- ✓ Escalables.
- ✗ No están completamente aislados (seguridad).
- ✗ La administración es crítica (se puede salir de control).
- ✗ Escasez de RRHH con conocimiento.

Container (contenedor)

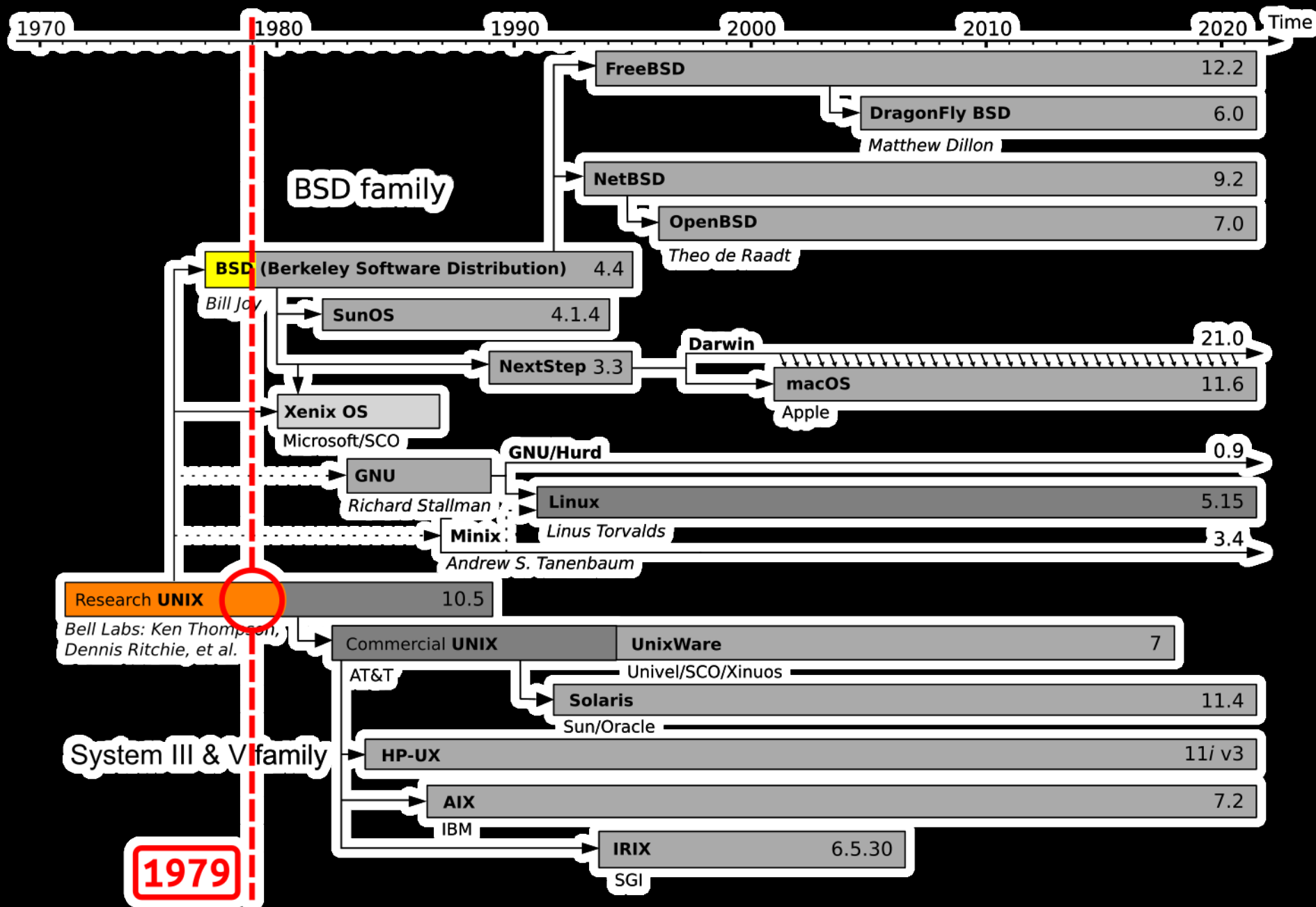
“Recipiente metálico grande y recuperable, de tipos y dimensiones normalizados internacionalmente, que sirve para transportar mercancías a grandes distancias. Los contenedores están provisto de ganchos o argollas para facilitar su carga y descarga mediante grúas.”



Docker (estibador)

“Persona que tiene por oficio cargar y descargar las mercancías de las embarcaciones y se ocupa de la adecuada distribución de los pesos.”

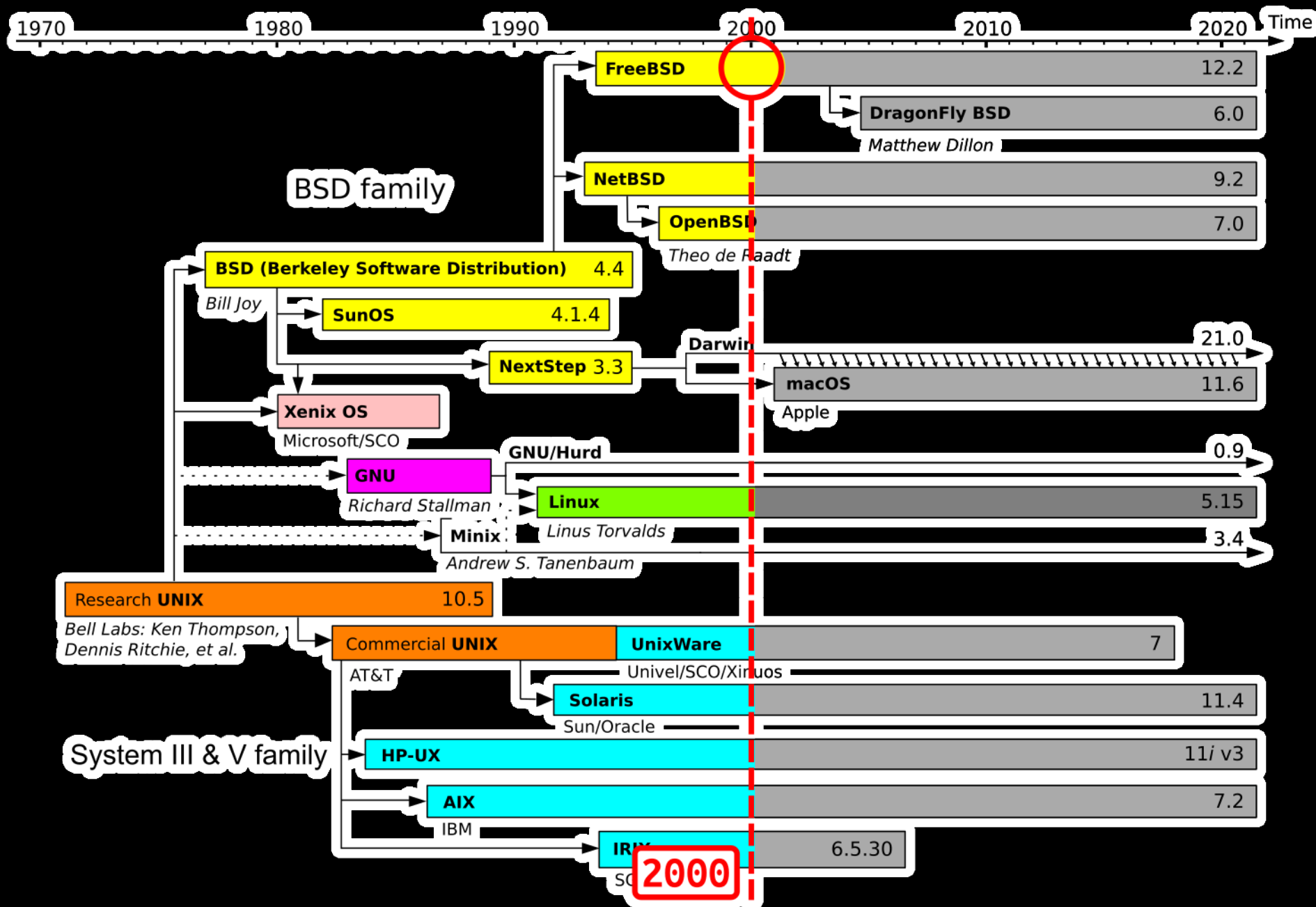




1979 - chroot (Unix version 7)

Sirve para ejecutar programas en un sistema de archivos restringido. ([Wikipedia](#))

```
function build_chroot () {
    local CHROOT_DIR="${1}"
    local APPS="${@:2}"
    for application in $APPS; do
        APP_ORIG=$(which $application)
        APP_DEST="${CHROOT_DIR}${(dirname $APP_ORIG)}"
        mkdir -p "${APP_DEST}"
        cp "${APP_ORIG}" "${APP_DEST}/"
        LIBS=$(ldd $APP_ORIG | grep -o -e '/[a-zA-Z0-9\._-]*')
        for library in $LIBS; do
            LIB_DEST="${CHROOT_DIR}${(dirname $library)}"
            mkdir -p "${LIB_DEST}"
            cp "${library}" "${LIB_DEST}/"
        done
    done
}
build_chroot prison bash ls
sudo chroot prison $(which bash)
```

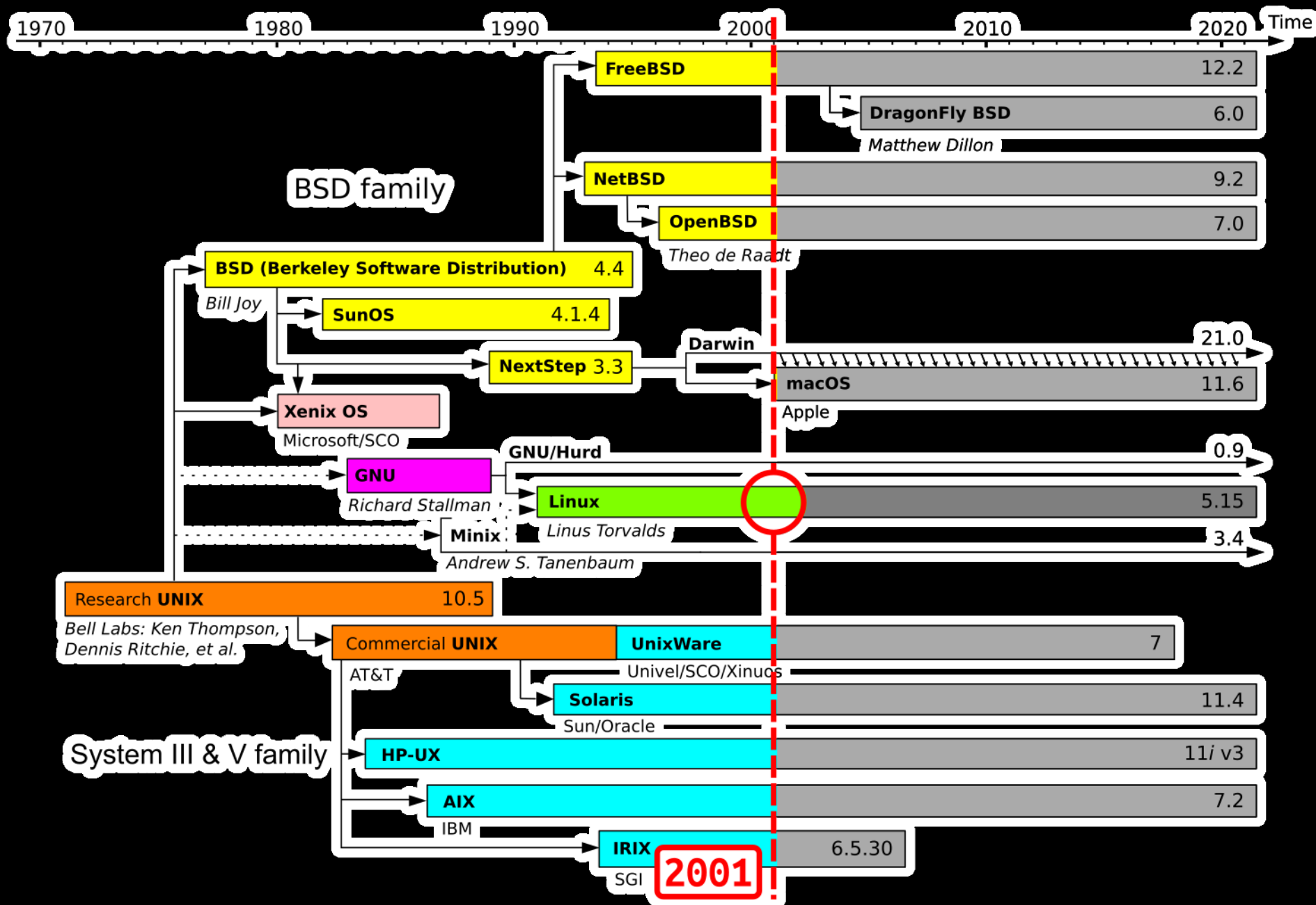


2000 - jails (FreeBSD)



Virtualización a nivel de Sistema Operativo que permite particionar el sistema en multiple mini-sistemas llamados jails.

La necesidad de jails provino de un pequeño proveedor de hosting de entornos compartidos, quien necesitaba establecer una separación limpia y clara entre sus servicios y el de sus clientes. ([Wikipedia](#))

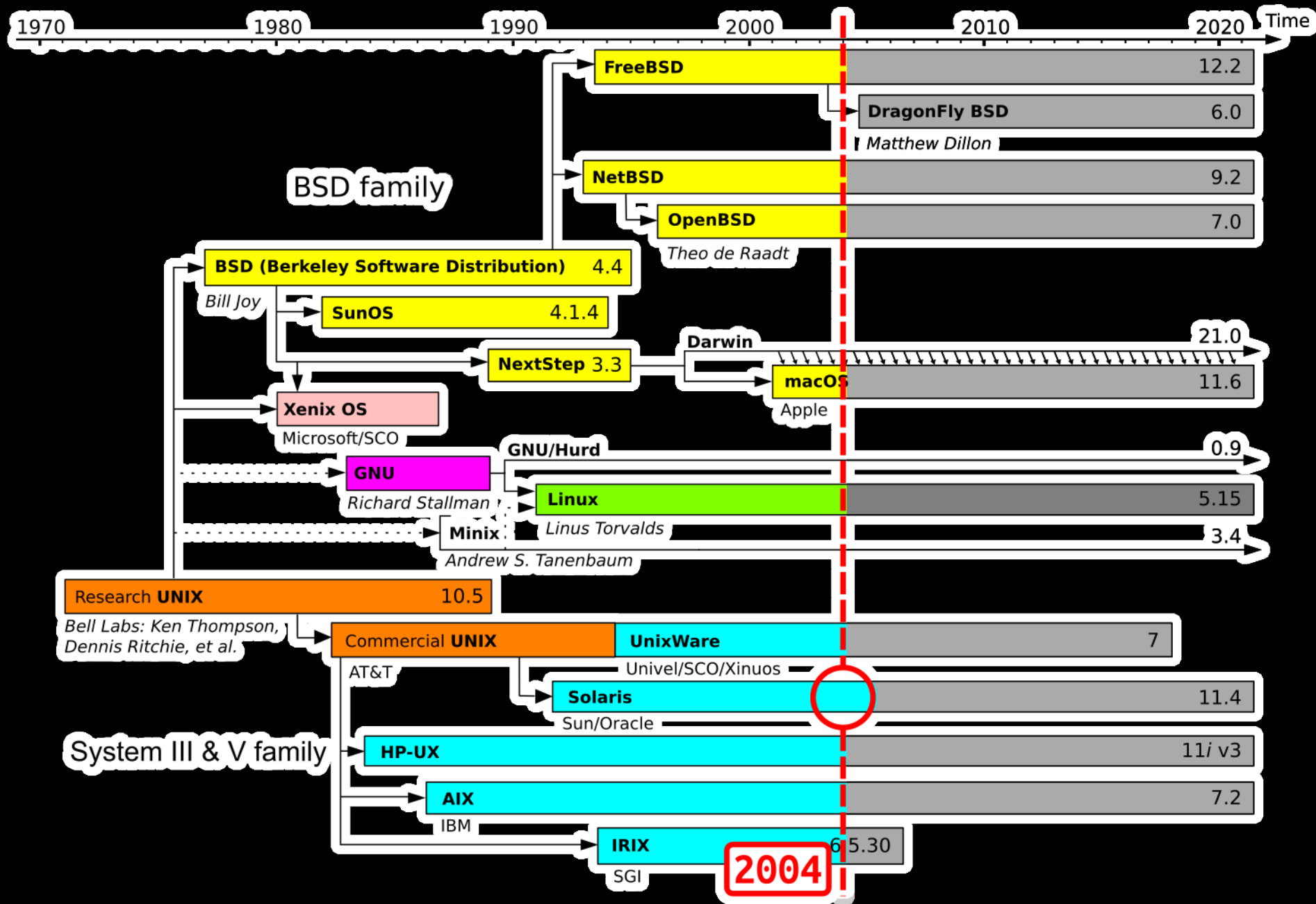


2001 – Linux VServer



Porta la aislación a nivel de kernel, pero requiere recompilar el mismo.

Conceptualmente Linux-VServer es similar a Solaris Containers (incluyendo la tecnología de aislamiento de Solaris Zones), OpenVZ o FreeBSD “jail”. ([Wikipedia](#))

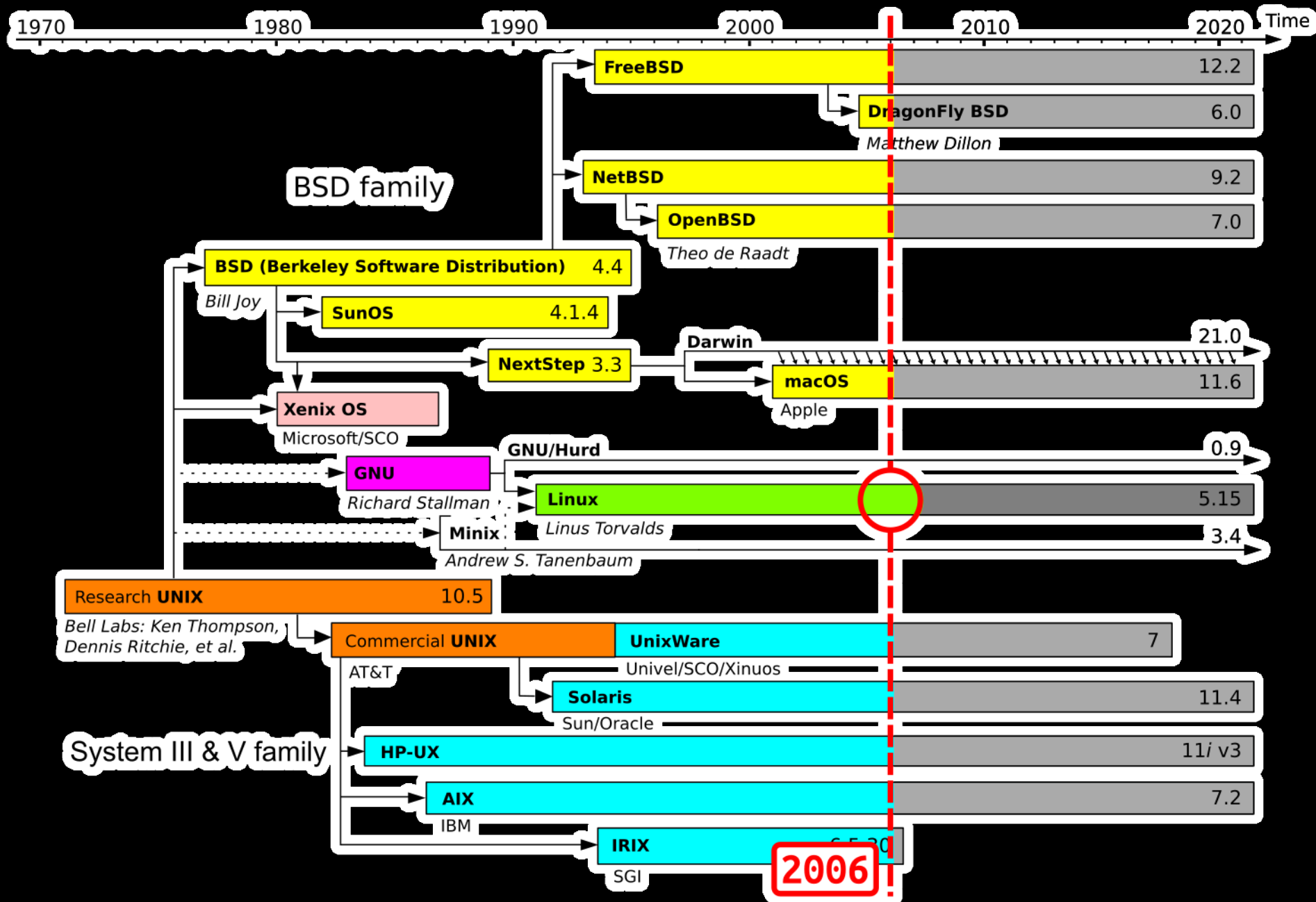


2004 - Solaris Zones



Introduce el concepto de "snapshots". ([Wikipedia](#))

“En informática, una copia instantánea de volumen o snapshot (del inglés foto instantánea) es una instantánea del estado de un sistema en un momento determinado. El término fue acuñado como una analogía a la de la fotografía. Puede referirse a una copia real del estado de un sistema o de una capacidad que ofrecen los sistemas de copia de seguridad.” ([Wikipedia](#))



2006 - cgroups (Google)

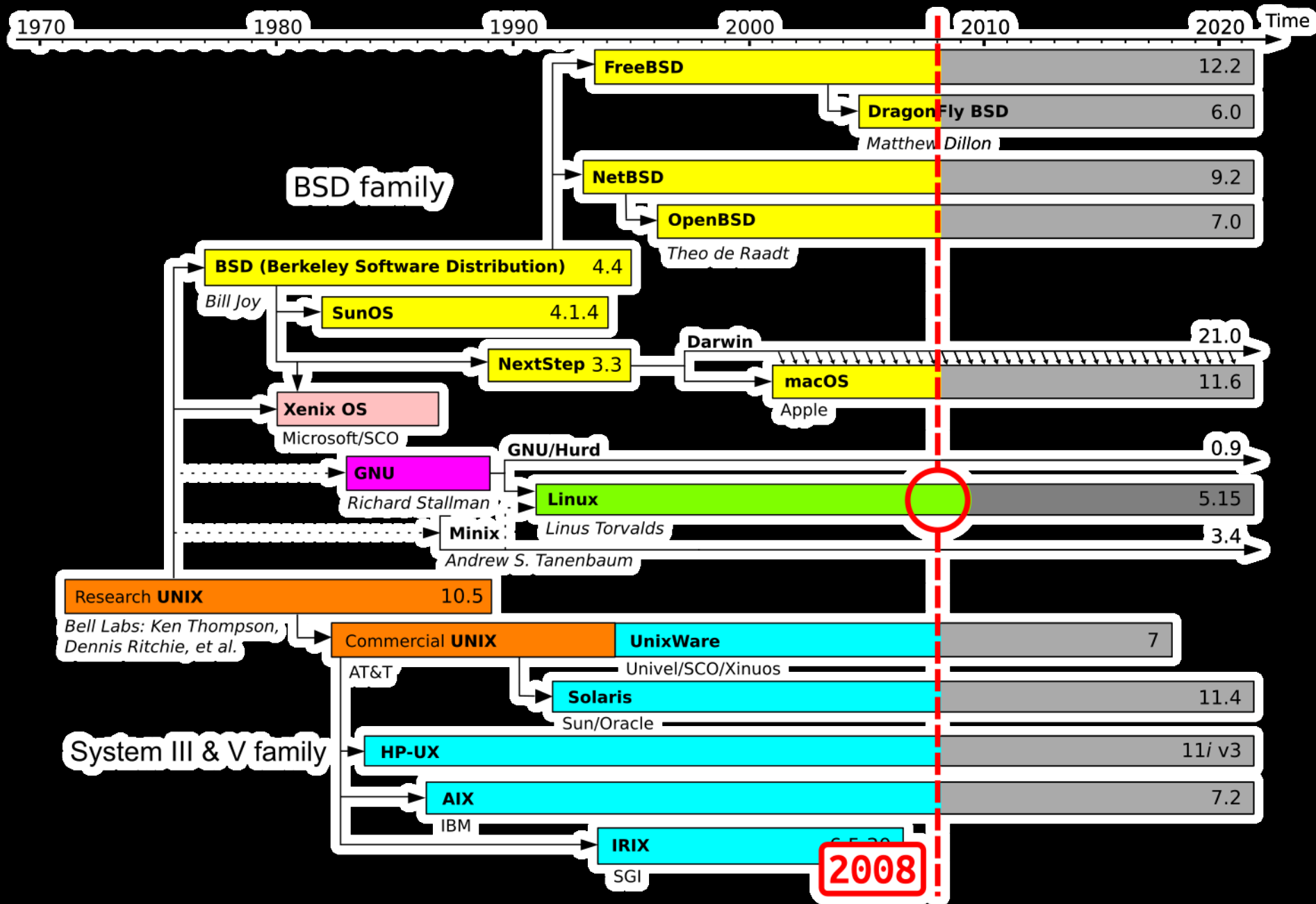


“cgroups” (abreviacion de control groups) es una característica del kernel Linux que limita, contabiliza y aísla el uso de los recursos (CPU, memoria, entrada y salida de discos, etc.).

El diseño de la segunda versión comenzó en el año 2013 (`'mount | grep cgroup'`). ([Wikipedia](#))

2006 - cgroups (Google)

```
# Install stress tool
sudo apt install stress-ng
stress-ng --cpu 4 --timeout 5s --metrics-brief
# Create control group
sudo mkdir /sys/fs/cgroup/edi_dos
# Check controllers
cat /sys/fs/cgroup/cgroup.controllers
cat /sys/fs/cgroup/cgroup.subtree_control
sudo bash -c 'echo '+cpu' > /sys/fs/cgroup/cgroup.subtree_control'
# Check and set max cpu
cat /sys/fs/cgroup/edi_dos/cpu.max
echo '10000 100000' | sudo tee /sys/fs/cgroup/edi_dos/cpu.max
# New bash instance
echo $$
bash
echo $$
stress-ng --cpu 4 --timeout 5s --metrics-brief
# Check group process and add current bash
cat /sys/fs/cgroup/edi_dos/cgroup.procs
echo $$ | sudo tee /sys/fs/cgroup/edi_dos/cgroup.procs
cat /sys/fs/cgroup/edi_dos/cgroup.procs
stress-ng --cpu 4 --timeout 5s --metrics-brief
# Exit bash, check process and remove group
cat /sys/fs/cgroup/edi_dos/cgroup.procs
sudo rmdir /sys/fs/cgroup/edi_dos
```

2008 - User Namespaces (Red Hat)



Red Hat

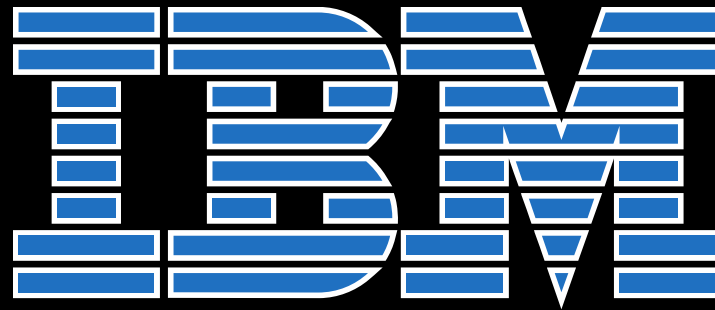
Los espacios de nombres son una característica del kernel Linux que particionan los recursos de tal forma que un conjunto de procesos vea un conjunto de recursos, mientras que otro conjunto de procesos vea un conjunto diferente de recursos. Los recursos pueden existir en multiples espacios. Algunos ejemplos de recursos son IDs de procesos, IDs de usuarios, nombres de archivos, acceso de red, etc.

Los espacios de nombres se originaron en el 2002 con el kernel 2.4.19 ("mount namespace"). A partir del 2006 comenzaron a agregarse nombre de espacios adicionales. Con en el kernel 3.8 se culminó el adecuado soporte a las funciones de contenedores, con la introducción del espacio de nombre de usuario. ([Wikipedia](#))

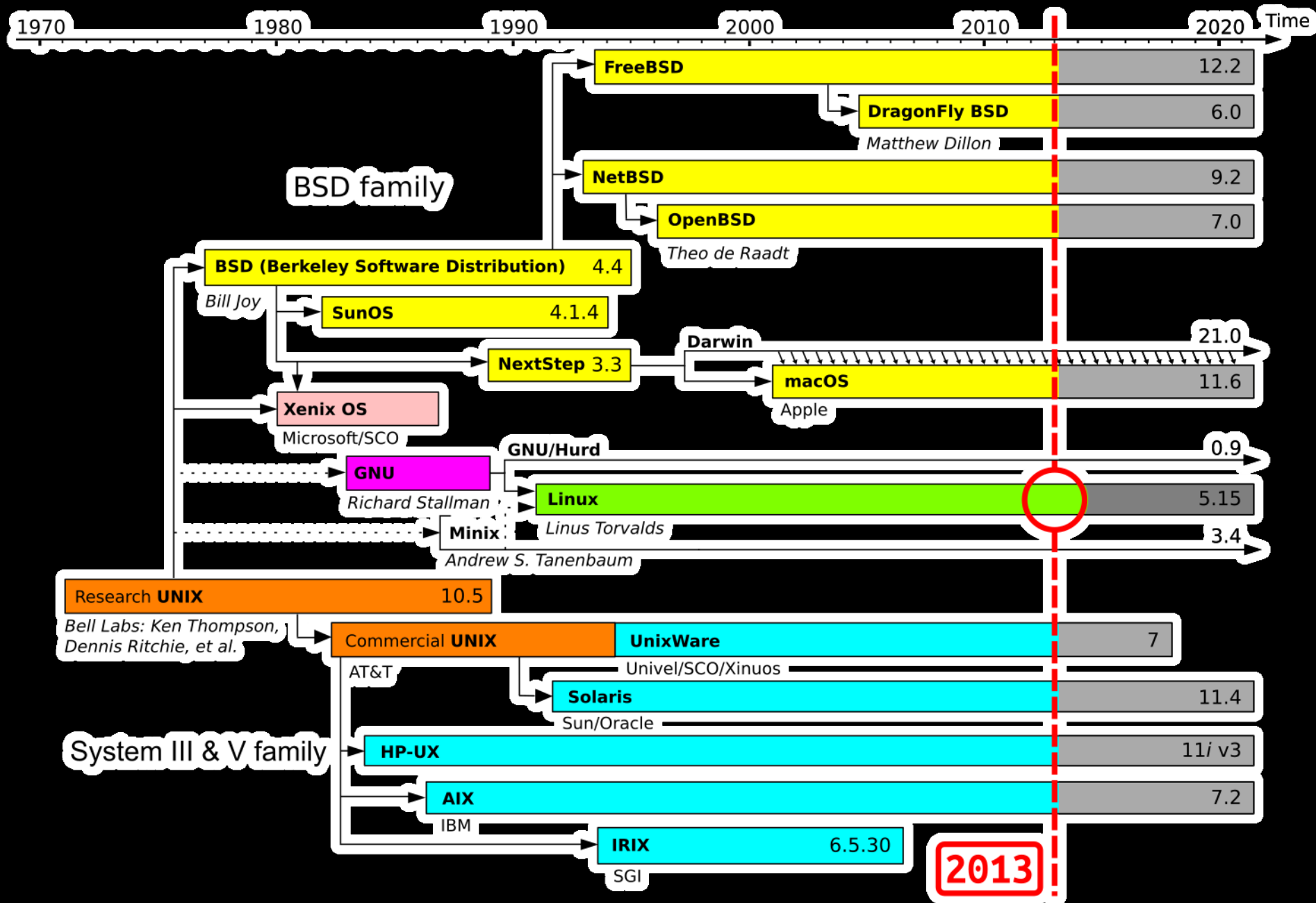
2008 - User Namespaces (Red Hat)

```
# Build chroot
build_chroot prison bash ps grep
# Mount "/proc" (process information) virtual filesystem
mkdir -p prison/proc
sudo mount --types proc /proc prison/proc/
# Enter the chroot
sudo chroot prison $(which bash)
# Find and kill external process
ps aux | grep htop
kill HTOP_PID
exit
# Setup a namespace for chroot
sudo unshare --pid --fork \
    --mount-proc=$PWD/prison/proc \
    chroot prison /bin/bash
ps aux
# Exit and umount virtual filesystem
exit
sudo umount prison/proc
```

2008 - LXC (IBM)



Linux Containers, también conocido por el acrónimo LXC, es una tecnología de virtualización a nivel de sistema operativo para Linux, que utiliza las funciones de containers del kernel. LXC permite que un servidor físico ejecute múltiples instancias de espacios de usuario aislados, conocidos como contenedores, jaulas, Servidores Privados Virtuales o Entornos Virtuales (EV). LXC no provee de una máquina virtual, más bien provee un entorno virtual que tiene su propio espacio de procesos y redes. LXC se basa en las funciones de cgroups y el aislamiento de espacios de nombres del kernel Linux. ([Wikipedia](#))



2013 - Docker



<https://docs.docker.com/get-docker/>

```
# Install (Debian derived systems)
sudo apt install docker docker-compose
sudo groupadd docker
sudo usermod -aG docker $USER
docker run hello-world
```

“All things are difficult before
they are easy”

- Thomas Fuller (1608-1661)