Guía técnica

Requerimientos técnicos de servidores y terminales

Sistemas Ferrum · Ferrum Nova · Versiones Linux y Windows

Versión 1.0

Septiembre 2026

Contenido

Antes de empezar el módulo

Este manual dice qué equipo necesitas para que Ferrum Nova trabaje bien: el servidor, las terminales, la red local y el enlace de internet. Cubre las dos versiones del sistema, la **versión Linux** (las terminales entran por la red a un servidor Ubuntu) y la **versión Windows** (cada computadora corre su propio programa y se conecta a la base de datos).

Está escrito para dos lectores a la vez. Si eres el dueño o el encargado y no eres técnico, lee el cuerpo de cada sección: las tablas de **mínimo** y **recomendado** y los cuadros *En palabras simples* te dicen qué comprar y qué pedirle a tu proveedor de internet. Si eres el técnico que va a instalar, las secciones marcadas con 🔧 llevan los nombres de los paquetes, los puertos, las variables del servidor y los comandos para comprobar cada punto.

Mínimo es lo que hace falta para que el sistema funcione sin errores con la carga indicada. **Recomendado** es lo que conviene comprar hoy para que el equipo dure cinco años sin volverse el cuello de botella. Cuando el presupuesto obligue a elegir, invierte primero en el **disco del servidor** (que sea de estado sólido) y en el **no-break**: son las dos cosas que más fallas y más lentitud evitan en la práctica.

Resumen en una página

Para una empresa típica, éstas son las respuestas rápidas. El detalle y las variantes vienen en cada sección.

Pregunta	Respuesta corta
¿Qué versión me conviene?	Linux, salvo que tengas una razón concreta para Windows (ver → `infraestructura-conceptos`). La versión Linux permite reusar computadoras viejas como terminales y centraliza todo en un solo equipo.
¿Qué servidor compro para 1 a 5 terminales?	4 núcleos, 16 GB de RAM, dos discos SSD de 480 GB en espejo (RAID 1), Ubuntu Server 24.04, no-break. Con 2 núcleos, 8 GB y SSD de 240 GB funciona.
¿Y para 6 a 15 terminales?	6 núcleos, 32 GB de RAM, dos discos NVMe de 1 TB en espejo, no-break. Mínimo 4 núcleos, 16 GB y un SSD de 480 GB.
¿Y para 16 a 40, o con varias sucursales?	8 núcleos, 64 GB de RAM, dos NVMe de 2 TB en espejo, un segundo servidor de contingencia y doble enlace de internet.
¿Qué necesita una terminal?	En la versión Linux, cualquier computadora con teclado completo y un programa de acceso remoto (PuTTY 0.71 o más nuevo); sirven equipos de 10 años. En la versión Windows, una PC con Windows 10 u 11 de 64 bits y 8 GB de RAM.
¿Qué internet necesito?	Para un solo local, 50 Mbps de bajada y 10 de subida por fibra (mínimo 10/3). Si tienes sucursales que replican, 100/20 con IP pública fija o dominio dinámico, y un

Pregunta	Respuesta corta
	segundo enlace de respaldo.
¿Puedo vender si se cae internet?	Sí. La venta, el inventario y la caja trabajan en la red local. Lo que se detiene es el timbrado de facturas, y Ferrum lo reintenta cuando el enlace regresa.
¿Qué pasa si se apaga el servidor?	Sin no-break, la base de datos puede quedar dañada. Con un servidor de contingencia, las terminales se pasan solas a él en un minuto (→ `red-internet-requerimientos`).

Cómo está organizado. Primero, cómo está armado el sistema y cuál versión te toca. Después el servidor Linux, el servidor y las PCs de la versión Windows, las terminales y periféricos, la red e internet. Al final, una hoja para calcular tu caso y entregársela a Sistemas Ferrum o a tu proveedor de equipo. En el anexo, al final, está la tabla resumen de hardware en una sola página horizontal, lista para imprimir.

1. INFRAESTRUCTURA-CONCEPTOS — Entender cómo está armado Ferrum y elegir la versión

Catálogo: *Infraestructura* · *Requerimientos técnicos* · *Tiempo estimado:* *10 minutos de lectura* · *Permiso requerido:* `Ninguno (todos los usuarios)`

Qué vas a lograr

Saber qué piezas forman una instalación de Ferrum Nova (servidor, terminales, sucursales, internet), qué hace cada una y cuál de las dos versiones del sistema te conviene.

Cuándo usar esto

Antes de comprar equipo, contratar internet o abrir una sucursal. Todas las cifras de las secciones siguientes dependen de dos números que aquí vas a definir: cuántas terminales trabajan **al mismo tiempo** y cuántas sucursales tienen servidor propio.

También cuando un proveedor de equipo te pregunte "¿qué necesita el sistema?": puedes entregarle esta sección junto con la hoja de → `infraestructura-dimensionar`.

📋 Antes de empezar

Necesitas…	Dónde se obtiene
Cuántos puestos de trabajo usarán Ferrum a la vez	Cuenta cajas, escritorios y celulares que estarán abiertos en la hora pico
Cuántos locales tiene la empresa y si cada uno debe seguir vendiendo sin internet	Decisión del dueño o gerente
Si ya tienes computadoras y de qué tipo	Inventario de equipo actual

Las piezas del sistema

Ferrum Nova siempre tiene tres piezas: un **servidor** que guarda la base de datos, las **terminales** desde las que trabaja la gente y la **red** que las une. Lo que cambia entre las dos versiones es dónde corre el programa.

Pieza	Versión Linux	Versión Windows
Servidor	Un equipo con Ubuntu Server. Guarda la base de datos **y** corre el programa de todas las terminales.	Un equipo con la base de datos MariaDB (en Windows o, mejor, en Linux). Sólo guarda datos.
Terminal	Cualquier aparato con teclado y un programa de acceso remoto (PuTTY, Termux…). Sólo muestra la pantalla del servidor.	Una PC con Windows 10 u 11 que tiene instalado `ferrumop.exe` y se conecta a la base de datos.

Pieza	Versión Linux	Versión Windows
Dónde se instala y actualiza	Una sola vez, en el servidor. Las terminales no se tocan.	En cada PC. Cada una se actualiza sola al arrancar, pero cada una es un punto de falla.
Impresoras	Conectadas al servidor o a la red; las administra el servidor (CUPS).	Instaladas en cada PC de Windows con su nombre.
Reportes en PDF	Los arma el servidor con WeasyPrint.	Los arma cada PC con Microsoft Edge.
Trabajo desde fuera del local	Natural: la terminal puede estar en otra ciudad o ser un celular.	Sólo con una red privada virtual (VPN) o túnel; no se recomienda.

En palabras simples

En la versión Linux el servidor es como la cocina de un restaurante: ahí se prepara todo, y las terminales son las mesas donde sólo se sirve. Por eso las terminales pueden ser computadoras viejas o baratas, y por eso el servidor tiene que ser bueno. En la versión Windows cada mesa tiene su propia estufa: las PCs deben ser mejores y hay más cosas que pueden fallar, pero no dependes de que el programa corra en el servidor.

Cuál versión te conviene

Tu situación	Versión que conviene	Por qué
Empresa nueva, o vas a renovar equipo	Linux	Un solo equipo bueno y terminales económicas. Instalación, respaldos y soporte remoto centralizados.
Ya tienes PCs con Windows en buen estado y una sola sucursal pequeña	Cualquiera de las dos	Las mismas PCs sirven como terminales Linux (con PuTTY) o como PCs de la versión Windows.
Quieres vendedores con celular, o terminales en otro local sin servidor	Linux	Una terminal Linux es sólo una conexión segura; funciona desde cualquier lugar con internet.
Varias sucursales que deben seguir vendiendo sin internet	Linux en cada sucursal	Cada sucursal lleva su servidor y replica con la matriz por túnel seguro; la matriz recibe hasta 300 sucursales.
Dependes de programas de Windows en la misma PC de la caja (por ejemplo, un lector o pinpad que sólo funciona en	Windows	El programa corre en la misma PC que el periférico. En la versión Linux una báscula se conecta por el programa `serbas`, pero otros aparatos

Tu situación	Versión que conviene	Por qué
Windows)		pueden no tener puente.

Las dos versiones pueden convivir: una empresa puede tener un servidor Linux con terminales Linux y, además, una PC con la versión Windows conectada a la misma base de datos.

Pasos

1.1 Cuenta las **terminales simultáneas**: los puestos que estarán abiertos en la hora de más trabajo, no los usuarios registrados. Un usuario que entra desde dos lugares cuenta dos veces.

1.2 Cuenta los **locales** y marca cuáles deben seguir vendiendo aunque se caiga internet. Cada uno de ésos necesita servidor propio; los demás pueden trabajar como terminales remotas del servidor central.

1.3 Anota qué **periféricos** usa cada puesto: impresora de tickets, impresora de facturas, báscula, lector de código de barras, pinpad.

1.4 Anota si alguien trabajará **desde fuera** (vendedores en ruta, el dueño desde casa, un contador externo).

1.5 Con la tabla de arriba, elige la versión. Si dudas, elige Linux.

1.6 Pasa a la sección del servidor que te toque (→ `servidor-linux-requerimientos` o → `servidor-windows-requerimientos`) con esos números en la mano.

Cómo verificar que funcionó

Puedes llenar las primeras cuatro líneas de la hoja de → `infraestructura-dimensionar` (terminales simultáneas, locales con servidor, locales sin servidor, usuarios remotos) sin consultar a nadie.

🔧 Para el administrador

El programa es un ejecutable de Harbour en modo texto. En Linux corre en el servidor bajo `tmux`: la terminal entra por SSH, el perfil kiosko lanza `ferrumop`, éste llama a `iniciaferrumop`, que revisa si hay versión nueva en `ferrum.mx/actual/ferrumop/`, la descarga y arranca `./ferrumlopubn` dentro de una sesión `tmux` con el nombre del usuario. Si el usuario ya tiene una sesión abierta en otra terminal, pregunta si la desconecta.

En Windows el ejecutable vive en `C:\ferrum\ferrumop` y lo lanza `iniciaferrum.bat`, que compara `versionop` contra `versionactual`, baja `ferlinop<versión>.zip` con el `curl` que trae Windows 10 y lo descomprime con `Expand-Archive` de PowerShell. Un archivo llamado `sucursal` en esa carpeta desactiva la búsqueda de actualizaciones (igual que en Linux).

La base de datos es **MariaDB** con motor InnoDB. Cada empresa es una base `<prefijo>DATOS<n>` y la configuración común vive en `<prefijo>FERRUM`. Ferrum conecta con `SET NAMES latin1` y guarda los textos como bytes CP850 dentro de columnas utf8; no cambies el juego de caracteres del servidor ni de la conexión.

Cada terminal lee `sql.ini` con tres servidores: `[LOCAL]` (donde se trabaja), `[RESPALDO]` (a donde se pasa si el local no responde) y `[PRINCIPAL]` (la matriz, para traspasos y consultas

centrales). Con un solo servidor, las tres secciones apuntan al mismo equipo y el cambio automático queda inactivo.

La topología multisucursal es una estrella bidireccional: cada sucursal tiene su MariaDB, que replica hacia la matriz por un canal con nombre, y la matriz replica hacia cada sucursal (`log_slave_updates=1` en la matriz, `OFF` en los nodos). Los nodos alcanzan a la matriz por un túnel SSH inverso, así que la sucursal no necesita IP pública ni puertos abiertos.

Referencias en el código: `iniciaferrumop`, `iniciaferrum.bat`, `bases.prg` (`CONECTASQL`), `configur.prg` (captura del `sql.ini`), `analisis/monitor_replica_latido.md` (topología).

⚠️ Si algo sale mal

Síntoma	Causa probable	Qué hacer
Contaste usuarios y te salió un servidor enorme	Se dimensiona por terminales abiertas a la vez, no por personas dadas de alta	Vuelve a contar sólo los puestos que trabajan en la hora pico.
Una sucursal se queda sin sistema cada vez que falla su internet	Trabaja como terminal remota del servidor central y no tiene servidor propio	Si esa sucursal debe vender siempre, ponle servidor propio con réplica (→ `red-internet-requerimientos`).
Compraste PCs potentes para terminales Linux y sobran	En la versión Linux el trabajo lo hace el servidor	No es un problema, pero la próxima compra puede ir a un mejor servidor o a un no-break.

💡 **Tip**	Si tienes computadoras que ya no corren un Windows moderno, no las deseches: como terminal de la versión Linux sólo necesitan abrir una conexión con PuTTY, y eso lo hace una PC de hace diez años sin problema.

Ver también

- `servidor-linux-requerimientos` — Elegir el servidor para la versión Linux
- `servidor-windows-requerimientos` — Elegir el servidor y las computadoras para la versión Windows
- `infraestructura-dimensionar` — Calcular qué servidor te toca

2. SERVIDOR-LINUX-REQUERIMIENTOS — Elegir el servidor para la versión Linux

Catálogo: *Infraestructura* · *Requerimientos técnicos* · *Tiempo estimado:* *15 minutos (lectura)* · *10 minutos (verificación en el servidor)* · *Permiso requerido:* `Ninguno (la verificación requiere acceso al servidor)`

Qué vas a lograr

Saber qué procesador, memoria, disco y sistema operativo debe tener el servidor según el número de terminales, y comprobar que un equipo que ya tienes cumple.

Cuándo usar esto

Al comprar o rentar el servidor de una instalación nueva, al abrir una sucursal con servidor propio, o cuando el sistema empezó a sentirse lento y quieres saber si el equipo se quedó corto.

Si vas a usar la versión Windows, la base de datos también puede vivir en un servidor Linux de esta sección; es lo recomendado (→ `servidor-windows-requerimientos`).

📋 Antes de empezar

Necesitas…	Dónde se obtiene
Número de terminales simultáneas y de sucursales que replicarán a este servidor	→ `infraestructura-conceptos`
Un lugar ventilado, con llave y con contacto regulado para el servidor	Instalación eléctrica del local
Para verificar un equipo existente: acceso al servidor como usuario con `sudo`	El administrador o Sistemas Ferrum

Lo que debe tener el servidor

Elige el perfil por las terminales que estarán abiertas a la vez en **este** servidor. Si es la matriz, suma las sucursales que replican hacia él como si fueran terminales (cada sucursal que replica equivale a unas dos terminales de carga).

Perfil	Terminales a la vez	Procesador	Memoria RAM	Disco
Básico	1 a 5	Mínimo: 2 núcleos. Recomendado: 4 núcleos.	Mínimo: 8 GB. Recomendado: 16 GB.	Mínimo: SSD de 240 GB. Recomendado: dos SSD de 480 GB en espejo (RAID 1).

Perfil	Terminales a la vez	Procesador	Memoria RAM	Disco
Mediano	6 a 15	Mínimo: 4 núcleos. Recomendado: 6 núcleos.	Mínimo: 16 GB. Recomendado: 32 GB.	Mínimo: SSD de 480 GB. Recomendado: dos NVMe de 1 TB en espejo.
Grande	16 a 40	Mínimo: 6 núcleos. Recomendado: 8 núcleos.	Mínimo: 32 GB. Recomendado: 64 GB.	Mínimo: dos NVMe de 1 TB en espejo. Recomendado: dos NVMe de 2 TB en espejo.
Corporativo	Más de 40, o matriz con 4 o más sucursales	Mínimo: 8 núcleos. Recomendado: 16 núcleos.	Mínimo: 64 GB. Recomendado: 128 GB.	Recomendado: dos NVMe de 2 TB en espejo para datos y un tercer disco para respaldos.

Para cualquier perfil, además:

Elemento	Mínimo	Recomendado
Sistema operativo	Ubuntu Server 20.04 LTS	Ubuntu Server 24.04 LTS (la 20.04 ya no recibe actualizaciones de seguridad)
Base de datos	MariaDB 10.3 (la que trae Ubuntu 20.04)	MariaDB 10.11 (la que trae Ubuntu 24.04)
Tarjeta de red	1 Gbps con cable	1 Gbps con cable; nunca por wifi
No-break (UPS)	Que aguante 10 minutos con el servidor y el switch	Que aguante 20 minutos y avise al servidor para apagarse solo
Procesador	De escritorio, 64 bits, de menos de 8 años	De servidor o de escritorio reciente; importa más la velocidad por núcleo que el número de núcleos
Respaldo fuera del servidor	Copiar la carpeta `respaldo` a un disco externo cada semana	Copia automática diaria a otro equipo o a la nube

En palabras simples

El servidor hace tres cosas a la vez: guarda la base de datos, corre el programa de cada terminal y arma los reportes. La base de datos vive en el disco y la memoria; por eso **el disco**

de estado sólido no es opcional: un servidor con disco mecánico se siente lento a media mañana aunque tenga el mejor procesador (nos pasó con un servidor de 16 núcleos: el procesador estaba al 6 % y el disco al 100 %). La memoria sirve para que la base de datos trabaje sin ir al disco: Ferrum le reserva la mitad de la RAM. El procesador rara vez es el problema.

El **no-break** es la pieza que más se olvida y la que más daño evita. Ferrum configura la base de datos para escribir rápido, a cambio de que un apagón repentino pueda dañarla. Con no-break el servidor se apaga en orden y no pasa nada.

El **espejo** de dos discos protege contra la falla de un disco, no contra un borrado ni contra un incendio. El respaldo diario que Ferrum hace solo se queda en el mismo servidor: alguien debe copiarlo fuera.

Pasos

2.1 Entra al servidor con un usuario que tenga `sudo` (por SSH o en la consola).

2.2 Comprueba la versión de Ubuntu: `lsb_release -ds`. Debe decir 24.04 o 20.04.

2.3 Cuenta los núcleos: `nproc`. Compara con la tabla del perfil.

2.4 Mide la memoria: `free -h`. La columna `total` del renglón `Mem` es la RAM instalada.

2.5 Comprueba que el disco es de estado sólido: `lsblk -d -o NAME,ROTA,SIZE,MODEL`. En la columna `ROTA`, `0` es SSD y `1` es disco mecánico.

2.6 Mide el espacio libre donde viven los datos y el sistema: `df -h /home /`. Ferrum avisa cuando quedan menos de 5 GB libres; para operar con calma deja al menos 100 GB.

2.7 Comprueba la base de datos: `mariadb --version` (o `mysql --version`) y `systemctl is-active mariadb`, que debe contestar `active`.

2.8 Confirma que el no-break está conectado al servidor **y** al switch de la red, y anota cuántos minutos aguanta con la carga real (desconecta el contacto un momento y mide).

2.9 Si vas a comprar, entrega a tu proveedor el perfil de la tabla y la hoja de → `infraestructura-dimensionar`; pide que la cotización diga marca y modelo del disco.

Cómo verificar que funcionó

Cada comando responde con un valor igual o mayor al del perfil elegido, `ROTA` es `0`, `mariadb` está `active` y el no-break aguanta al menos 10 minutos con todo conectado.

🔧 Para el administrador

Instalación. El instalador oficial es `instalador/instalaopubuntu` (v2.5), pensado para Ubuntu 20.04 y 24.04; con otra versión avisa y continúa bajo tu responsabilidad. Tiene modo interactivo (whiptail) y desatendido (`--unattended` con variables de entorno). Instala: `mariadb-server mariadb-client html2ps a2ps barcode dialog qrencode samba smbclient gcc make build-essential cups cups-bsd printer-driver-all zip unzip wget curl tmux joe lynx mysqltuner openssh-server ddclient imagemagick weasyprint`. Después de instalar, y cada vez que algo falte, corre `sudo revisaderechos` (v8): repara permisos, paquetes y la pareja WeasyPrint/pydyf.

Usuarios y rutas. Crea el usuario `ferrum` (home `/home/ferrum`, grupos `sys,lp,ferrum_users`), `impresion` (administra CUPS) y los grupos `ferrum_users`, `local_admins` y `administraferrum`. El sistema vive en `/home/ferrum/ferrumop` con las subcarpetas `salida` y `respaldo`; los binarios `ferrumlop` y `ferrumop` llevan SUID. Los datos de

MariaDB se mueven a `/home/mysql` (con override de systemd `ProtectHome=false` y perfil AppArmor). Cada terminal es un usuario Linux creado con `ferrumnu <usuario>`, que genera su llave ed25519 y el perfil kiosko.

MariaDB. El instalador escribe `/etc/mysql/mariadb.conf.d/90-ferrum.cnf` con `innodb_buffer_pool_size` = 50 % de la RAM (mínimo 256 MB, tope 16 GB; súbelo a mano en servidores de 64 GB o más), una instancia de pool por GB (máx. 8), `innodb_flush_log_at_trx_commit=2`, `innodb_flush_method=O_DIRECT`, `innodb_file_per_table=1`, `innodb_log_file_size=256M`, `innodb_io_capacity` 200 (mecánico) o 1000 (SSD, detectado por `/sys/block/sda/queue/rotational`), `max_connections=200`, `table_open_cache=4000`, `tmp_table_size=64M`, `slow_query_log=1` con `long_query_time=2` y `lower_case_table_names=1` (Ferrum se niega a trabajar sin esto). Descarga `mysqltuner.pl` a la carpeta del sistema para reajustar tras unas semanas de operación.

Versión de MariaDB. No hay un bloqueo por versión: Ferrum lee `VERSION()` al conectar y degrada funciones por debajo de 10.1 (`IF NOT EXISTS` en rutinas e índices, `max_statement_time`) y de 10.0 (canales de réplica con nombre). La versión más vieja que se declara compatible en el código es 10.0.35; en producción hay 10.3 y 10.11. Usa la que trae la distribución.

Replicación y contingencia. Todo servidor que participe en réplica o failover necesita en `my.cnf`: `server_id` único en toda la empresa, `log_bin=mysql-bin`, `binlog_format=STATEMENT` (obligatorio: con ROW o MIXED el latido `SRVPING` se pisa y las sucursales que pasan días sin línea pierden cambios), `log_slave_updates` OFF en nodos y 1 en la matriz, `event_scheduler=ON` (sin él el latido nunca corre) y un `auto_increment_offset` propio por servidor (Ferrum alerta si dos servidores lo repiten). Con STATEMENT, MariaDB escribe un aviso `Unsafe statement` por cada escritura; siléncialo con `log_warnings=1`, no cambiando el formato.

Disco e I/O. Una base de una empresa con varios años de operación pesa unos 30 GB; el respaldo diario comprimido de esa base ronda los 5 GB y se conservan siete (uno por día de la semana, valores del sistema `RES1`..`RES7`). El parámetro `ALEDISCO` (default 5 GB) dispara la alerta de disco tanto en el `datadir` como en la carpeta del sistema. En el incidente de un servidor con 21 terminales y 43 usuarios, la CPU (16 núcleos) iba en carga 0.91 y el cuello fueron dos discos mecánicos de 7,200 rpm en RAID 1 compartidos entre `/` y `/home`: `/var/log` saturado saturaba también a MariaDB. Instala `sysstat` para conservar el historial de carga y evita `LogLevel debug` en CUPS (250 MB de log al día con 30 impresoras).

Memoria por terminal. Cada sesión (sshd + tmux + `ferrumlopubn`) ocupa entre 50 y 100 MB; un render de WeasyPrint sube temporalmente 200 a 300 MB. Con el pool en la mitad de la RAM, reserva 150 MB por terminal simultánea más 2 GB para el sistema y el cache de archivos.

Otros servicios que quedan configurados. SSH con `ClientAliveInterval 30`, `ClientAliveCountMax 5`, `UseDNS no` y `PermitRootLogin no`; CUPS con política `abort-job`; Samba con el bloque `ferrumsmb`; `ddclient` para el dominio dinámico; un cron `0 3 * * * /home/ferrum/respaldatodo.sh`; y el archivo `/etc/sudoers.d/zz-administraferrum` que da al grupo `administraferrum` control de CUPS, MariaDB, Samba, SSH, `apt` y apagado sin contraseña. El firewall y fail2ban se instalan con `util/activafirewall.sh` y `util/instalafail2ban.sh` (→ `red-internet-requerimientos`).

Monitor de eventos. En la matriz corre una instancia de Ferrum en modo monitor (valor del sistema 669 `USO DE TERMINAL` = `M`, con usuario automático y empresa predeterminada). Vive en una sesión `tmux` que vigila `monitorferrum.sh` desde el crontab cada 4 minutos; si no late

en 60 s la barra de estado de las terminales dice `SIN MONITOR`. Cuenta como una terminal más al dimensionar. El archivo `nomonitor` en la carpeta del sistema lo detiene para mantenimiento.

openSUSE. Instalaciones antiguas con openSUSE Leap siguen funcionando (`revisaderechos` tiene rama `zypper` y arranca con `screen` en vez de `tmux`), pero ahí no hay WeasyPrint: los PDF salen por `html2ps`. No instales servidores nuevos con openSUSE.

⚠️ Si algo sale mal

Síntoma	Causa probable	Qué hacer
El sistema se siente lento a media mañana, sobre todo al imprimir o al abrir reportes	Disco mecánico saturado; el procesador está ocioso	Cambia a SSD o NVMe. Mientras tanto, revisa que `/var/log` no esté creciendo (CUPS en `debug`, avisos de MariaDB) y pon `log_warnings=1`.
Después de un apagón la base de datos no arranca o faltan movimientos	El servidor no tiene no-break y MariaDB escribe con `flush_log_at_trx_commit=2`	Instala no-break. Recupera del último respaldo bueno (→ Sistemas Ferrum); no repares una base que arrancó rota.
Al entrar dice que el servidor no está configurado para usar minúsculas	Falta `lower_case_table_names=1` en `my.cnf`	Agrégalo y reinicia MariaDB **antes** de crear bases; cambiarlo después obliga a reconstruir.
Los reportes salen sin formato, o el PDF falla con un error de `pydyf`	WeasyPrint no está instalado o quedó con una versión de `pydyf` incompatible	Corre `sudo revisaderechos`; a mano: `sudo apt install weasyprint` y `sudo pip3 install pydyf==0.9.0`. `weasyprint --version` no sirve de prueba: genera un reporte real.
Ferrum avisa poco espacio en disco	Quedan menos de 5 GB (`ALEDISCO`) en el disco de datos o del sistema	Borra respaldos viejos que ya copiaste fuera y archivos de `salida`; si es recurrente, disco más grande.
Dos servidores repiten `auto_increment_offset` (alerta PELIGRO al entrar)	El servidor de contingencia se clonó del principal sin cambiar `my.cnf`	Cambia el `auto_increment_offset` y el `server_id` de uno de los dos y reinicia MariaDB.

💡 **Tip**	Al comprar, pide dos discos iguales aunque el perfil mínimo no lo exija: el espejo es lo más barato que puedes hacer hoy para no parar la operación el día que un disco muera. Y guarda la cotización: el modelo exacto del disco es lo primero que se pregunta al diagnosticar lentitud.

🚫 **No hagas esto**	**No** instales el servidor sobre un disco mecánico "mientras tanto": la base de datos queda ahí y moverla después implica parar la operación. Tampoco pongas el servidor a trabajar por wifi ni lo uses como computadora de escritorio de alguien.

Ver también

- `infraestructura-conceptos` — Entender cómo está armado Ferrum y elegir la versión
- `red-internet-requerimientos` — Preparar la red local y el enlace de internet
- `infraestructura-dimensionar` — Calcular qué servidor te toca
- `infraestructura-tabla-hardware` — Tabla resumen de hardware (anexo)

3. SERVIDOR-WINDOWS-REQUERIMIENTOS — Elegir el servidor y las computadoras para la versión Windows

Catálogo: *Infraestructura* · *Requerimientos técnicos* · *Tiempo estimado:* *15 minutos (lectura)* · *5 minutos por PC (verificación)* · *Permiso requerido:* *`Ninguno (la verificación requiere ser administrador de la PC)`*

Qué vas a lograr

Saber qué necesita el servidor de base de datos y cada PC con `ferrumop.exe` en la versión Windows, y comprobar que una PC cumple antes de instalarle el sistema.

Cuándo usar esto

Cuando la empresa ya trabaja con la versión Windows, o cuando decidiste usarla porque una caja depende de un aparato que sólo funciona en Windows (→ `infraestructura-conceptos`).

También cuando vas a agregar una PC de Windows a una instalación Linux: las dos versiones conviven contra la misma base de datos.

📋 Antes de empezar

Necesitas…	Dónde se obtiene
Número de PCs que correrán el sistema a la vez	→ `infraestructura-conceptos`
Decidir dónde vivirá la base de datos: en un servidor Linux (recomendado) o en una máquina Windows	Con Sistemas Ferrum
Para verificar una PC: entrar como administrador de Windows	El responsable del equipo

Lo que debe tener cada equipo

En la versión Windows hay dos tipos de equipo: **el servidor de base de datos**, que sólo guarda la información, y **las PCs**, que corren el programa. Como el trabajo se reparte, el servidor puede ser más chico que en la versión Linux, pero cada PC debe ser más capaz que una terminal Linux.

El servidor de base de datos

Elemento	Mínimo	Recomendado
Dónde instalarlo	Una PC con Windows 10 u 11 Pro de 64 bits dedicada, que nadie use de escritorio	Un servidor Linux de → `servidor-linux-requerimientos`; sirve para las dos versiones y trae instalador, respaldos y soporte remoto

Elemento	Mínimo	Recomendado
Si se queda en Windows	MariaDB 10.6 para Windows	MariaDB 10.11 (versión de soporte largo) en Windows Server 2019 o 2022 cuando pasen de 10 PCs
Procesador y RAM (1 a 5 PCs)	2 núcleos, 8 GB	4 núcleos, 16 GB
Procesador y RAM (6 a 15 PCs)	4 núcleos, 16 GB	6 núcleos, 32 GB
Más de 15 PCs	No se recomienda la base de datos en Windows	Servidor Linux del perfil Grande
Disco	SSD de 240 GB	Dos NVMe de 1 TB en espejo
No-break	Obligatorio	Con apagado automático

Cada PC con el sistema

Elemento	Mínimo	Recomendado
Windows	Windows 10 de 64 bits, versión 1803 o posterior (trae `curl` y Microsoft Edge)	Windows 11 Pro de 64 bits, con actualizaciones al día
Procesador	2 núcleos	4 núcleos
Memoria RAM	4 GB	8 GB
Disco	SSD con 10 GB libres	SSD de 240 GB o más
Programas	Microsoft Edge (viene con Windows) y `SumatraPDF.exe` junto a `ferrumop.exe`	Los mismos; sin otros programas pesados abiertos en la caja
Red	Cable a 100 Mbps al mismo switch que el servidor	Cable a 1 Gbps; nunca wifi para una caja
Pantalla y teclado	Monitor de 17 pulgadas o más; teclado completo con teclas `F1` a `F12`	Monitor de 19 pulgadas o más
Impresoras	Instaladas en Windows con su nombre	Impresora de tickets en USB o red, con controlador del fabricante

En palabras simples

Cada PC de Windows hace el trabajo que en la versión Linux hace el servidor: por eso pide más memoria y disco de estado sólido. Los reportes en PDF los arma Microsoft Edge en la propia PC, y para mandarlos a la impresora sin preguntar se usa un programa gratuito, SumatraPDF, que se copia en la carpeta del sistema. Si la PC es lenta, los reportes tardan en esa PC, no en las demás.

Windows 7 y Windows 8 **no sirven**: no traen `curl` (con el que Ferrum se actualiza) ni el Edge moderno (con el que arma los PDF). Windows 10 de 32 bits tampoco.

Pasos

3.1 En la PC, presiona `Win+R`, escribe `winver` y `Enter`. Debe decir Windows 10 (versión 1803 o mayor) o Windows 11.

3.2 Abre **Configuración** → **Sistema** → **Acerca de**. Revisa `RAM instalada` y que el `Tipo de sistema` sea de 64 bits.

3.3 Abre el símbolo del sistema (`Win+R`, `cmd`) y escribe `curl --version`. Debe responder con una versión; si dice que no se reconoce el comando, Windows está desactualizado.

3.4 Abre Microsoft Edge una vez para confirmar que existe y cierra la ventana.

3.5 Comprueba que existe la carpeta `C:\ferrum\ferrumop` y que dentro está `SumatraPDF.exe` junto a `ferrumop.exe` (si el sistema ya está instalado).

3.6 En **Configuración** → **Bluetooth y dispositivos** → **Impresoras**, anota el **nombre** exacto de la impresora que usará esta PC: es lo que se captura en Ferrum como puerto del dispositivo.

3.7 Confirma que la PC llega al servidor: `ping <ip-del-servidor>` debe responder en menos de 5 ms dentro del local.

3.8 En el servidor de base de datos (si es Windows), confirma que MariaDB está instalado como servicio y arranca solo, y que `lower_case_table_names=1` está en su `my.ini`.

Cómo verificar que funcionó

La PC arranca `iniciaferrum.bat`, se actualiza sola si hay versión nueva, entra al sistema, genera un reporte en PDF y lo manda a la impresora sin que aparezca ningún cuadro de diálogo.

🔧 Para el administrador

Sin instalador automático. A diferencia de Linux, no hay script de instalación para Windows: se copia la carpeta `C:\ferrum\ferrumop` con `ferrumop.exe`, sus bibliotecas y `iniciaferrum.bat`, se captura `sql.ini` y se crea un acceso directo al `.bat`. El `.bat` baja `versionop` de `ferrum.mx` (4 s de tope), compara con `versionactual` y, si hay nueva, descarga `ferlinop<versión>.zip` (120 s de tope) y lo expande con PowerShell. El propio programa también sabe actualizarse (`VerificaActualizacionWin`: baja `ferrum.mx/actual/ferwin<versión>.zip` y aplica `ferrumop_new.exe`) y, si falta `unzip.exe`, lo baja de `ferrum.mx/util/`.

curl. Ferrum usa el `curl.exe` que Windows 10 incluye desde la versión 1803 (`C:\Windows\System32\curl.exe` o `Sysnative`), con respaldo a un `curl` en la carpeta del sistema. Con él se actualiza, timbra, manda correo y consulta al SAT.

PDF e impresión. El motor HTML→PDF es Microsoft Edge en modo headless: `RUTAEDGE()` busca `msedge.exe` en `Program Files`, `Program Files (x86)` y `LOCALAPPDATA`. Si no lo encuentra, avisa `MICROSOFT EDGE NO SE ENCONTRO EN ESTE EQUIPO` y usa el motor clásico. Edge se lanza con un `--user-data-dir` propio por terminal: con el perfil compartido, un Edge abierto de fondo captura la orden y no genera el PDF. La impresión silenciosa la hace `SumatraPDF.exe` portable (3.6.1 de 64 bits), buscado primero junto a `ferrumop.exe` y después en `LOCALAPPDATA` y `Program Files`; sin él, Ferrum avisa `COLOQUE SumatraPDF.exe EN LA CARPETA DEL SISTEMA` y no imprime. El campo `DISPUERTO` del dispositivo debe ser el **nombre** de la impresora en Windows (se valida contra la lista del sistema y cae a la predeterminada si no coincide); un puerto crudo `LPT1` o una ruta `\\pc\imp` no funciona con Sumatra. WeasyPrint no se usa en Windows.

Base de datos en Windows. MariaDB para Windows se instala como servicio con su `my.ini`. Aplica los mismos valores que el instalador Linux (→ `servidor-linux-requerimientos`): `lower_case_table_names=1` **antes** de crear bases, `innodb_buffer_pool_size` a la mitad de la RAM, `innodb_flush_log_at_trx_commit=2`, `max_connections=200`, y si va a replicar, `server_id`, `log_bin`, `binlog_format=STATEMENT` y `event_scheduler=ON`. El respaldo diario lo hace cada terminal con las plantillas `RES1`..`RES7` de Windows: `mysqldump ... > respaldo\res%EMP%NDIA%TER.sql` sin comprimir, así que reserva disco para siete copias del tamaño de la base y necesitas `mysqldump.exe` en el `PATH` de la PC que respalda.

Conexión. Cada PC abre su propia conexión al puerto 3306 del servidor, así que ese puerto debe estar permitido en el firewall del servidor sólo para la red local; el usuario SQL debe aceptar conexiones desde las IP de la LAN (no sólo `localhost`). Los tiempos del `sql.ini` (`TIMELECTURA`/`TIMEESCRITURA`) van en 30 a 60 s en red local. Ferrum no distingue sesiones remotas en Windows: `CC["SUDO"]` queda siempre en falso y las utilerías de administración del servidor (desconectar usuarios, reiniciar servicios, reparar réplicas) sólo existen en la versión Linux.

Parque mixto. Una PC de Windows puede convivir con un servidor y terminales Linux. El valor global `WEASYOK` no se escribe desde Windows precisamente para no apagar WeasyPrint en las terminales Linux; en Windows el motor moderno se decide por `WEASYOK<>'D'` y la presencia de Edge.

Básculas. Una PC de Windows lee la báscula directo del puerto serial (`SERUBI=LOCAL`, `BASCULA=S`, `SERBAS=<COM>`) sin instalar nada más. Si la báscula está en otra máquina, corre ahí `serbas.exe` y abre el puerto 9999 en su firewall (→ `terminales-requerimientos`).

⚠️ Si algo sale mal

Síntoma	Causa probable	Qué hacer
El reporte se abre en pantalla pero no se imprime, o pide elegir impresora	Falta `SumatraPDF.exe` junto a `ferrumop.exe`, o el puerto del dispositivo no es el nombre de la impresora de Windows	Copia `SumatraPDF.exe` a `C:\ferrum\ferrumop`. En Ferrum captura como puerto el nombre exacto de la impresora.
El PDF no se genera cuando Edge está abierto	Se está usando el perfil compartido de Edge (instalación vieja del sistema)	Actualiza Ferrum: las versiones actuales usan un perfil propio (`--user-data-dir`) por terminal.

Síntoma	Causa probable	Qué hacer
Dice `MICROSOFT EDGE NO SE ENCONTRO` y los reportes salen feos	Windows sin Edge Chromium (Windows 7/8 o Windows 10 muy viejo)	Actualiza Windows o instala Microsoft Edge desde microsoft.com.
`iniciaferrum.bat` no se actualiza y muestra `curl no se reconoce`	Windows 10 anterior a la versión 1803	Actualiza Windows. Como paliativo, copia un `curl.exe` a la carpeta del sistema.
Una PC entra y las demás dicen que no alcanzan el servidor	El firewall del servidor sólo permite 3306 desde `localhost`, o el usuario SQL está restringido a `localhost`	Permite 3306 desde la red local y crea el usuario SQL con `'usuario'@'192.168.%'`.
La caja se desconecta a ratos y hay que volver a entrar	La PC está por wifi o el cable de red tiene falso	Cablea la caja al switch; revisa el cable y el puerto.

💡 **Tip**	Aunque las PCs sean de Windows, pon la base de datos en un servidor Linux: tendrás el instalador, los respaldos automáticos, el failover y el soporte remoto de la versión Linux, y las PCs ni se enteran.

🚫 **No hagas esto**	**No** instales MariaDB en la misma PC que usa la cajera como caja principal "para ahorrar un equipo": el día que esa PC se apague, se reinicie por una actualización de Windows o se llene de archivos, se detiene toda la empresa.

Ver también

- `infraestructura-conceptos` — Entender cómo está armado Ferrum y elegir la versión
- `servidor-linux-requerimientos` — Elegir el servidor para la versión Linux
- `terminales-requerimientos` — Elegir las terminales, impresoras y básculas
- `infraestructura-tabla-hardware` — Tabla resumen de hardware (anexo)

4. TERMINALES-REQUERIMIENTOS — Elegir las terminales, impresoras y básculas

Catálogo: *Infraestructura* · *Requerimientos técnicos* · *Tiempo estimado:* *10 minutos (lectura)* · *5 minutos por terminal (verificación)* · *Permiso requerido:* `Ninguno (todos los usuarios)`

Qué vas a lograr

Saber qué aparato sirve como terminal en cada versión, qué teclado y pantalla necesita, y cómo se conectan impresoras, básculas y lectores.

Cuándo usar esto

Al equipar un puesto nuevo, al decidir si una computadora vieja todavía sirve, o al comprar impresoras y básculas para que funcionen con el sistema desde el primer día.

Las PCs de la versión Windows tienen su propia tabla en → `servidor-windows-requerimientos`; aquí se tratan las terminales de la versión Linux y los periféricos de las dos.

📋 Antes de empezar

Necesitas…	Dónde se obtiene
Saber qué versión usa la empresa	→ `infraestructura-conceptos`
La dirección del servidor en la red local y el usuario de cada terminal	Los crea el administrador en el servidor (`ferrumnu`)
Para impresoras de red: un lugar en el switch y una IP fija	El responsable de la red

Lo que sirve como terminal en la versión Linux

Una terminal Linux no corre el programa: sólo abre una conexión segura (SSH) al servidor y muestra su pantalla. Por eso casi cualquier aparato con teclado sirve.

Aparato	Qué necesita	Comentario
PC con Windows 10 u 11	PuTTY 0.71 o más nuevo, con la casilla **Allow terminal to use 24-bit colours** marcada	El caso más común. Sirven PCs de hace diez años; 2 GB de RAM bastan.
PC con Linux (Ubuntu, Mint…)	La terminal del sistema (`gnome-terminal`, Terminator) y el comando `ssh`	Colores completos sin configurar nada.
Mac	La aplicación Terminal	Igual que Linux.
Celular o tablet	Termux con el menú de Ferrum	Ideal para vendedores en ruta y

Aparato	Qué necesita	Comentario
Android	(`instalatermux`) y un teclado Bluetooth para trabajo largo	para el dueño. Consume muy poco datos.
Cliente ligero, Raspberry Pi, computadora reciclada	Cualquier sistema que corra un cliente SSH	La opción más barata para cajas: sin disco, sin ventilador, sin Windows que mantener.
Windows 7 u 8	PuTTY 0.71 o más nuevo	Funciona, pero Microsoft ya no le da seguridad; úsalo sólo en una red sin acceso a internet.

Para cualquiera de ellos:

Elemento	Mínimo	Recomendado
Pantalla	80 columnas por 25 renglones legibles: monitor de 15 pulgadas	Monitor de 19 pulgadas o más; Ferrum aprovecha ventanas más grandes
Teclado	Completo, con `F1` a `F12`, `Alt`, `Ctrl` y teclado numérico	Teclado de escritorio; en laptops revisa que las teclas `F` no requieran `Fn`
Red	Cable a 100 Mbps al mismo switch del servidor	Cable a 1 Gbps. Wifi sólo para celulares y tablets
Conexión	1 Mbps y menos de 150 ms de latencia hacia el servidor	Red local (menos de 5 ms) o internet con menos de 80 ms

Impresoras

Tipo	Versión Linux	Versión Windows
Tickets (térmica 58 u 80 mm)	USB al servidor, o de red (puerto 9100 o LPD 515). La administra CUPS en el servidor y se asigna a cada terminal o sucursal.	USB o red, instalada en la PC con su nombre.
Facturas y reportes (láser o inyección)	De red con IP fija, dada de alta en CUPS. Los PDF los arma WeasyPrint.	Instalada en cada PC. Los PDF los arma Edge y los imprime SumatraPDF.
Compartida desde	Posible por Samba o LPD, pero	Igual: la PC que comparte se

Tipo	Versión Linux	Versión Windows
una PC de Windows	esa PC debe estar encendida; evítalo para cajas.	vuelve un punto de falla.
Etiquetas y códigos de barras	El servidor genera el código (`barcode`, `qrencode`); cualquier impresora de red o USB.	Cualquier impresora instalada en Windows.

Compra impresoras **de red** con IP fija siempre que puedas: no dependen de que una computadora esté encendida y se comparten entre terminales.

Básculas y lectores

Los **lectores de código de barras** USB se comportan como un teclado: se conectan a la terminal (o a la PC) y no requieren configuración. Las **básculas** se conectan por puerto serial (o USB-serial). En la versión Windows, la PC lee la báscula directamente. En la versión Linux, la terminal no tiene acceso a puertos, así que la báscula se conecta a cualquier máquina de la red que corra el programa `serbas`, que la expone por la red en el puerto 9999; puede ser una PC pequeña junto a la báscula o el propio servidor.

Pasos

4.1 En una PC de Windows, instala PuTTY desde `putty.org` (versión 0.71 o más nueva). Ábrelo y en **Window** → **Colours** marca **Allow terminal to use 24-bit colours**.

4.2 En **Session**, escribe la dirección del servidor, puerto `22`, y guarda la sesión con el nombre de la terminal.

4.3 Conéctate con el usuario que te dio el administrador. Ferrum arranca solo: no hay que escribir ningún comando.

4.4 Presiona `Ctrl+F1` para abrir la pantalla de estado y revisa que la paleta de colores se vea con degradados y no con colores planos o al azar.

[CAPTURA: pantalla de estado de Ferrum con la paleta de colores completa]

4.5 Prueba las teclas de función: `F2` abre la búsqueda, `Esc` regresa. Si `F1`..`F12` no responden, revisa el teclado (tecla `Fn` en laptops) o la configuración del programa.

4.6 En un celular Android, instala Termux desde F-Droid, ejecuta la línea `curl -sL https://ferrum.mx/ins/instalatermux | bash` y agrega el servidor en el menú que aparece al abrir Termux.

4.7 Conecta cada impresora a la red (o al servidor) y pide al administrador que la dé de alta; después imprime un reporte de prueba desde la terminal que la usará.

Cómo verificar que funcionó

La terminal entra directo a Ferrum sin escribir comandos, los colores se ven completos, las teclas `F` y `Alt` responden, y el reporte de prueba sale en la impresora asignada.

🔧 Para el administrador

Colores. El driver de terminal de Harbour decide el modo de color una sola vez al arrancar, por el valor de `TERM`. `NORMALIZATERMLINUX()` en `ferrum.prg` promueve `TERM` a `xterm-256color`

con truecolor en sesiones SSH, usa `screen-256color` dentro de `screen` (GNU screen no soporta truecolor) y respeta `linux` en la consola física (16 colores). El `/etc/tmux.conf` del instalador fija `default-terminal screen-256color`, `escape-time 10`, `status off` y la capacidad `Tc` para `xterm-256color`. PuTTY anterior a 0.71 recibe las secuencias `38;2;r;g;b` y pinta colores al azar; si no se puede actualizar, exporta `HARBOUR="GTTRM:EXCLR:2"` en el perfil de ese usuario. El valor del sistema `TIPTERM` (default `putty-256color`) no cambia el modo de color: sólo fija `TERM` de los subprocesos y activa el tratamiento de `Shift+F3..F10` de PuTTY.

Teclas. En terminal sólo son confiables `Ctrl+letra` y casi siempre `Alt+letra`; los modificadores sobre flechas, `Home`, `End` y `F` dependen del emulador (VTE manda `CSI 1;5C`, PuTTY manda `ESC O C`). `Shift+flecha` no llega. `Ctrl+B` y `Ctrl+Z` son el respaldo universal de primera y última columna en los browses.

Usuarios de terminal. `ferrumnu <usuario>` crea el usuario Linux (grupo `users`, shell `bash`), genera su par de llaves ed25519 (RSA 4096 si no hay soporte) e imprime la privada con instrucciones para PuTTYgen; copia el perfil kiosko `util/auto` que promueve `TERM` y lanza `ferrumop` seguido de `exit`. `util/ver <usuario>` entra a la sesión `tmux` de otra terminal para asistirla; con tmux 3.3 o más nuevo lo hace vía `sudo` (regla en `/etc/sudoers.d/zz-administraferrum`), así que hay que correr `revisaderechos` en cada servidor para que no pida contraseña.

Termux. `instalatermux` instala `openssh` y `sshpass`, agrega teclas extra en `~/.termux/termux.properties`, deja `menuferrum` en `$PREFIX/bin` y engancha el arranque en `$PREFIX/etc/termux-login.sh`. Las contraseñas viven en `~/.ferrum/conexiones.conf` (600) y se pasan por `SSH_ASKPASS_REQUIRE=force` (OpenSSH 8.4 o más nuevo), sin tocar disco al conectar. Soporta saltos encadenados para llegar a sucursales detrás de la matriz.

Impresoras en Linux. `util/nuevaimpresora.sh` hace `lpadmin -p <nombre> -E -v <url> -o printer-error-policy=retry-job`; `util/impresoras.sh` diagnostica CUPS y ofrece conexión Socket/JetDirect (9100), IPP y LPD (515, la recomendada para compartir desde Windows remoto). Ferrum imprime con `lpr -l -# %COP -P %PUE %ARC` y los reportes de texto plano con `a2ps`. Las impresoras cuyo RIP pinta barras negras con el PDF de WeasyPrint se marcan `DISIMAGEN='S'` y Ferrum rasteriza con `gs` antes de mandar. La asignación puede ser por terminal (`IMPGLOBAL='N'`), global (`'S'`) o por sucursal (`'U'`, tabla `IMPSUC` de cada empresa, sin caída a global).

Básculas. `serbas` es un ejecutable aparte (`serbas.prg`, hay binario Windows y ELF estático para Linux). Escucha en el puerto 9999 (`-p:NNNN` para cambiarlo), protocolo de texto `COMANDO,param,...` con una conexión por comando; `PING` contesta `PONG` para probar con telnet. No tiene autenticación: limita el 9999 a la LAN o VPN. En la terminal se captura `SERUBI=<ip>` o `<ip:puerto>` y `SERBAS` con el puerto COM de la máquina donde está la báscula; timeout del cliente 5 s. `serbas --sim` simula una báscula para capacitar. Manual completo en `bascula.serbas-instalar`.

⚠️ Si algo sale mal

Síntoma	Causa probable	Qué hacer
Los colores se ven planos, o cambian al azar en cada pantalla	PuTTY anterior a 0.71, o la casilla de 24 bits apagada, o `TERM` sin `256color`	Actualiza PuTTY y marca **Allow terminal to use 24-bit colours**. Como paliativo, `HARBOUR="GTTRM:EXCLR:2"`

Síntoma	Causa probable	Qué hacer
		para ese usuario.
`Shift+F3`, `Ctrl+flecha` o `Alt+letra` no hacen nada	El emulador no manda esos códigos, o la tecla la captura el sistema operativo	Usa las teclas alternas (`Ctrl+B`/`Ctrl+Z` para columnas) o cambia de emulador. En laptops revisa `Fn`.
La sesión se cierra sola después de unos minutos sin uso	Un router o firewall corta conexiones inactivas; el servidor manda latidos cada 30 s pero la red los descarta	En PuTTY, **Connection → Seconds between keepalives** en `20`. Revisa el firewall de la red.
Al entrar dice que ya hay una sesión abierta en otra terminal	El mismo usuario quedó abierto en otra máquina	Contesta que sí para desconectar la otra, o pide al administrador que la revise con `ver`.
La báscula funciona en Windows pero no desde una terminal Linux	La terminal Linux no lee puertos seriales	Instala `serbas` en la máquina donde está la báscula y captura su IP en `SERUBI`.
La impresora imprime desde una terminal y no desde otra	Está asignada por terminal o por sucursal a un dispositivo distinto	Revisa la asignación de dispositivos de esa terminal (→ manual de impresoras).

💡 **Tip**	Para una caja nueva, un cliente ligero o una mini PC sin disco duro con PuTTY o Linux cuesta una fracción de una PC completa, no hace ruido, no se llena de virus y, si se descompone, se cambia por otro en cinco minutos sin instalar nada.

🚫 **No hagas esto**	**No** pongas una caja por wifi "mientras llega el cable": cada microcorte se convierte en una sesión colgada y en una venta a medias. El wifi es para celulares y tablets.

Ver también

- `servidor-windows-requerimientos` — Las PCs de la versión Windows
- `red-internet-requerimientos` — Preparar la red local y el enlace de internet
- `bascula.serbas-instalar` — Instalar el servidor de báscula

5. RED-INTERNET-REQUERIMIENTOS — Preparar la red local y el enlace de internet

Catálogo: *Infraestructura* · *Requerimientos técnicos* · *Tiempo estimado:* *15 minutos (lectura)* · *10 minutos (verificación)* · *Permiso requerido:* `Ninguno (la verificación requiere acceso al servidor)`

Qué vas a lograr

Saber qué red local y qué internet contratar según el número de sucursales y de usuarios remotos, qué debe dejar pasar el firewall y cómo se protege el servidor.

Cuándo usar esto

Al contratar o cambiar el enlace de internet, al abrir una sucursal, al conectar vendedores desde fuera, y cada vez que el timbrado, el correo o las actualizaciones dejan de funcionar después de que alguien tocó el router o el firewall.

Ferrum vende, cobra y controla inventario en la red local sin internet. El enlace se usa para timbrar facturas, mandar correos y WhatsApp, actualizarse, replicar sucursales, atender terminales remotas y dar soporte.

📋 Antes de empezar

Necesitas…	Dónde se obtiene
Número de sucursales con servidor propio y de usuarios remotos	→ `infraestructura-conceptos`
Los datos del enlace actual: proveedor, velocidad contratada, si tiene IP pública fija	El contrato o el proveedor de internet
Para verificar: acceso al servidor	El administrador o Sistemas Ferrum

Lo que debe tener la red

Red local

Elemento	Mínimo	Recomendado
Switch	100 Mbps, con un puerto por terminal e impresora	1 Gbps administrable; el servidor y las impresoras en cable
Cableado	Categoría 5e	Categoría 6, con canaleta y etiquetas en cada extremo
Direcciones	IP fija para el servidor y para cada impresora de red	IP fijas fuera del rango que reparte el router; el servidor como servidor

Elemento	Mínimo	Recomendado
		DNS interno no es necesario
Wifi	Sólo para celulares y tablets, con contraseña	Red de invitados separada de la red de las cajas
No-break	Para el servidor	Para el servidor, el switch y el router (sin router no hay timbrado)

Enlace de internet

Escenario	Mínimo	Recomendado
Un solo local; terminales en la red local	10 Mbps de bajada, 3 de subida, estable	50/10 Mbps por fibra
Matriz que recibe réplicas de sucursales	20/10 Mbps con IP pública fija o dominio dinámico	100/20 Mbps por fibra, IP pública fija y un segundo enlace de otro proveedor (doble WAN)
Sucursal con servidor propio	10/5 Mbps; no necesita IP pública	50/10 Mbps y un enlace 4G/5G de respaldo
Local sin servidor (terminales remotas)	5/2 Mbps y latencia menor a 150 ms al servidor	20/5 Mbps y latencia menor a 80 ms; enlace de respaldo, porque sin internet no vende
Usuario remoto ocasional (celular, casa)	Datos de celular; una sesión consume entre 10 y 50 kbps	Cualquier enlace estable
Módulos web públicos (portal de clientes, facturación en línea)	20/10 Mbps e IP pública fija	100/20 Mbps y dominio propio con certificado

En palabras simples

Ferrum consume muy poco internet: una terminal remota gasta menos que una llamada de voz, y la réplica de una sucursal manda sólo los cambios. Lo que sí importa es que el enlace **no se caiga** y que responda rápido (latencia). Por eso la recomendación es fibra y, donde la operación no puede parar, un segundo enlace de otro proveedor: el servidor cambia solo de uno a otro.

La **IP pública fija** (o el dominio dinámico) sólo la necesita el servidor al que otros se conectan: la matriz que recibe sucursales, o el servidor que atiende terminales remotas y módulos web. Una sucursal con servidor propio no la necesita: ella abre la conexión hacia la matriz.

Lo que el firewall debe dejar salir

Todo servidor y toda PC de la versión Windows necesitan estas salidas. Sin ellas, el sistema arranca pero falla en el momento de timbrar, actualizar o verificar la licencia.

Destino	Puerto	Para qué
`ferrum.mx`	80 y 443	Actualizaciones del sistema, catálogos del SAT, instaladores
`ferrumoficina.dynns.com` y `ferrum.no-ip.net`	**3306**	Verificación de la licencia y de la versión publicada. Es el puerto que más firewalls bloquean
`facturehoy.com` (todos sus subdominios: `wsprod3`, `cancelacion`, `cfdiglobal`, `pruebasclientes`)	443 (y 80 para servicios antiguos)	Timbrado, cancelación y verificación de CFDI
`consultaqr.facturaelectronica.sat.gob.mx`, `verificacfdi.facturaelectronica.sat.gob.mx`, `siat.sat.gob.mx`	443	Estado de los CFDI y validación de constancias
`mail.exclusivehosting.net`	2525	Correo de avisos y soporte de Ferrum, y WhatsApp (sale por correo)
El servidor de correo de la empresa (`smtp.office365.com`, `smtp.gmail.com`…)	587 (o el que use)	Correos con facturas y reportes a nombre del negocio
`api.anthropic.com`	443	Sólo si se usa la inteligencia artificial en reportes
`ipinfo.io`, `ifconfig.me`, `api.ipify.org` y el proveedor de dominio dinámico	80 y 443	Conocer y publicar la IP pública del servidor
Servidores de paquetes de Ubuntu (`archive.ubuntu.com`, `security.ubuntu.com`)	80 y 443	Actualizaciones de seguridad del servidor
La matriz (sólo desde sucursales con servidor)	22 (o el puerto SSH de la matriz)	Túnel inverso por el que replican y reciben soporte

Lo que el firewall debe dejar entrar

Hacia el servidor	Puerto	Desde dónde
Terminales Linux (SSH)	22	Red local; desde internet sólo si hay terminales remotas, protegido con fail2ban y llaves
PCs de la versión Windows y módulos web (MariaDB)	3306	Sólo red local. Nunca abierto a internet sin túnel o VPN
Túneles de sucursales (sólo la matriz)	22 y los rangos 22001 a 22500 y 33001 a 33500	Internet; los usa el servidor de cada sucursal
Módulos web: surtido, entregas, turnos, tableros	6464 y 6465	Red local; internet sólo si el módulo es público
Portal público y facturación en línea	80 y 443	Internet, con dominio y certificado
Impresoras compartidas, carpetas compartidas	631 (CUPS), 445 (Samba)	Sólo red local
Báscula compartida (`serbas`)	9999	Sólo red local
Documentación (BookStack)	65000	Red local o internet, según se decida

Cómo se conectan las sucursales

Cada sucursal con servidor propio abre un **túnel seguro** hacia la matriz y por él replica en las dos direcciones. Si el enlace se cae, la sucursal sigue vendiendo con sus datos y al regresar la conexión se pone al corriente sola; puede pasar días sin línea. La matriz sabe cada 10 segundos qué sucursales están en línea y al corriente, y con eso decide a dónde mandar las terminales si un servidor falla. Un local pequeño sin servidor trabaja como **terminales remotas** de la matriz: es más barato, pero sin internet no vende.

Pasos

5.1 Desde el servidor, mide la latencia a internet: `ping -c 5 ferrum.mx`. El promedio debe ser menor a 100 ms; si es mayor de 200 ms el enlace es lento aunque tenga muchos megas.

5.2 Prueba las salidas críticas una por una: `nc -zv ferrum.mx 443`, `nc -zv ferrumoficina.dynns.com 3306`, `nc -zv wsprod3.facturehoy.com 443` y `nc -zv mail.exclusivehosting.net 2525`. Cada una debe contestar `succeeded`. Si falta `nc`, instálalo con `sudo apt install netcat-openbsd`.

5.3 Si mandan correo con la cuenta de la empresa, prueba también su servidor: por ejemplo `nc -zv smtp.office365.com 587`.

5.4 Mide la velocidad real desde una PC del local en `fast.com` a la hora de más trabajo; compárala con lo contratado.

5.5 Anota la IP pública del servidor: `curl -s https://api.ipify.org`. Si cambia de un día a otro, el enlace es dinámico y la matriz necesita dominio dinámico o IP fija.

5.6 Si es matriz, confirma con el proveedor que la IP pública es fija (o contrata el dominio dinámico) y que el puerto 22 no está filtrado.

5.7 Si hay dos enlaces, pide al administrador correr `activafirewall.sh` con doble WAN y desconecta un enlace un minuto para comprobar que el servidor sigue saliendo por el otro.

5.8 Pide al administrador que instale fail2ban (`instalafail2ban.sh`) en todo servidor con el puerto 22 abierto a internet.

Cómo verificar que funcionó

Las cuatro pruebas de salida contestan `succeeded`, la latencia promedio a `ferrum.mx` está por debajo de 100 ms, y en el panel de alertas del sistema no aparece ninguna sucursal como fuera de línea.

🔧 Para el administrador

Licencia por 3306. `CONECTASQLFERRUM()` resuelve `ferrumoficina.dynns.com` y `ferrum.no-ip.net` por DNS y abre una conexión MySQL saliente al 3306 para verificar la licencia y la versión publicada. Un `/etc/resolv.conf` que apunte al DNS de un router caído tumba la verificación aunque haya internet; `activafirewall.sh` publica `1.1.1.1 8.8.8.8 9.9.9.9`. El timbrado va por `curl` con timeouts de 60 a 600 s; si el PAC exige lista blanca por IP, coordínalo con él antes de cambiar de enlace o poner doble WAN.

Sin respuesta del PAC. Si el timbrado no obtiene respuesta, el CFDI queda en estado `P` sin token, con el XML conservado; el reintento lo reenvía byte a byte y el PAC devuelve el mismo timbre dentro de las 24 h del primer envío. Pasadas 24 h, Ferrum pregunta antes de reenviar. Por eso una caída de internet corta no genera duplicados.

Túneles. `util/tunel-principal.sh` prepara la matriz (usuario `tunel64`, hasta 300 túneles): SSH inverso en 22001 a 22500 y reenvío de MariaDB en 33001 a 33500 (en firewalld 22001 a 22300 y 33001 a 33300). `util/tunel-remoto.sh` en la sucursal abre el túnel inverso con failover opcional y reenvío de MySQL y CUPS; `util/tunel-conecta.sh` gestiona las conexiones desde `/etc/tunel-servers.conf` (`PUERTO_SSH:NOMBRE:DESCRIPCION`). Harbour sólo alcanza el MariaDB de la matriz; por eso el alta de un nodo se hace con el paquete que genera `BUNDLEPREPARANODO` (dump con `--master-data=2 --routines --events --triggers`, `my.cnf` del nodo, `CHANGE MASTER` y el `EVENT` del latido).

Latido. `EV_LATIDO_PING` en la matriz escribe `SRVPING` cada 10 s y `EV_LATIDO` en cada nodo contesta `SRVPONG`; sin eco en 90 s el nodo aparece FUERA. Requiere `event_scheduler=ON` y `binlog_format=STATEMENT` en todos. El failover automático (`FAILAUTO`, default S) cambia todas las terminales al `[RESPALDO]` (o `[PRINCIPAL]`) tras 60 s sin respuesta, sólo si el candidato está en línea y al corriente; una revisora vuelve a probar el local cada 15 min y tras 2 revisiones buenas programa el regreso coordinado. Requisitos: mismo `sql.ini` en todas las terminales de un servidor y `auto_increment_offset` propio en el respaldo.

Timeouts del sql.ini. `TIMELECTURA` y `TIMEESCRITURA` (default 30 s) se aplican como `net_read_timeout`/`net_write_timeout`; la ayuda en pantalla sugiere 30 a 60 en red local y 15 a 30 en remoto, y `BITACORA=S` en conexiones inestables.

Firewall y fail2ban. `activafirewall.sh` v2.0 configura UFW (permite 22, 3306, 6464 y 65000; Samba y CUPS sólo desde la LAN) y, con dos interfaces, doble WAN con ruteo por política, CONNMARK, `rp_filter=2`, vigilancia con `ping -I` y histéresis (3 fallos para caer, 12 aciertos para volver); deja `ferrum-wan-estado` y `ferrum-wan-forzar`. Aún no se ha probado con dos proveedores reales. Ojo: en modo "puertos públicos" el 3306 queda abierto al mundo, mitigado con fail2ban. `instalafail2ban.sh` v2.0 banea con ipset en todos los puertos, castigo creciente de 1 a 30 días, jaula `recidive` de 90 días, jaulas SSH y MySQL, y lista blanca por dominio dinámico tomada de `SERVIDOR.SRVDOMINIO`; nunca banea `ferrumoficina.dynns.com` ni `ferrum.no-ip.net`.

DDNS. `ddclient` publica una sola IP pública; tras un failover de WAN hay que republicar. Las sucursales suelen no tener IP fija, por eso fail2ban y el latido trabajan con dominios.

Módulos web. `instalaweb.sh` v1.3 instala Apache 2.4 y PHP 7.4 o más nuevo (con `pdo_mysql`, `mbstring`, `curl`, `xml`), más `poppler-utils` y `zbar-tools`. Puertos: 6464 web compartida, 6465 tableros, 6469+clave facturación dedicada, 6479+clave web dedicada, 6489+clave tableros dedicados, 80/443 portal con Let's Encrypt, 65000 BookStack. El usuario SQL de la web (`'%'`) sólo tiene SELECT.

Consumo. Una sesión SSH de Ferrum mueve entre 10 y 50 kbps; 20 terminales remotas caben en 1 Mbps. La réplica manda sólo sentencias (STATEMENT), del orden de decenas de MB al día por sucursal activa. Lo que sí pesa es el respaldo si lo copias a la nube: unos 5 GB por empresa por noche.

⚠️ Si algo sale mal

Síntoma	Causa probable	Qué hacer
Al arrancar dice que no pudo verificar la licencia, pero hay internet	El firewall bloquea la salida por 3306, o el DNS no resuelve `ferrumoficina.dynns.com`	Permite la salida 3306 a los dos dominios de Ferrum y pon DNS públicos en el servidor.
Las facturas se quedan en proceso y no se timbran	No hay salida 443 a `facturehoy.com`, o el enlace está caído	Prueba `nc -zv wsprod3.facturehoy.com 443`. Al volver el enlace, reintenta con `Enter` sobre la factura: no se duplica.
Los correos y WhatsApp no salen	Bloqueado el 2525 a `mail.exclusivehosting.net` o el 587 al correo de la empresa	Abre esos puertos de salida (→ manual de correo).
La sucursal aparece FUERA en la matriz aunque tenga internet	Cayó el túnel, `event_scheduler` está en OFF o el reloj del servidor está muy desfasado	Reinicia el túnel en la sucursal; revisa `SELECT @@event_scheduler` y la hora (`timedatectl`).
Las terminales remotas escriben con retraso o se traban al teclear	Latencia alta (más de 200 ms) o pérdida de paquetes, no falta de megas	Cambia a un enlace de menor latencia (fibra). Baja `TIMELECTURA` a 15 a 30 en

Síntoma	Causa probable	Qué hacer
		el `sql.ini` de esas terminales.
El servidor sigue sin salir a internet aunque el segundo enlace está bien	El doble WAN no está activado o el dominio dinámico sigue publicando la IP vieja	Corre `ferrum-wan-estado`; fuerza con `ferrum-wan-forzar` y republica el DDNS.
Alguien de la empresa quedó bloqueado al entrar por SSH desde su casa	fail2ban lo baneó por contraseñas equivocadas	El administrador lo desbanea (`fail2ban-client set sshd unbanip <ip>`) y le entrega su llave en vez de contraseña.

💡 **Tip**	Antes de firmar con el proveedor de internet, pide por escrito la latencia y si la IP es fija. "200 megas" por un enlace inestable sirve menos a Ferrum que 50 por fibra con 10 ms de latencia.

🚫 **No hagas esto**	**No** abras el puerto 3306 del servidor a internet para que una PC remota se conecte "rapidito": es la base de datos completa expuesta. Usa una terminal Linux por SSH o, para la versión Windows, una VPN o el túnel.

Ver también

- `servidor-linux-requerimientos` — Elegir el servidor para la versión Linux
- `terminales-requerimientos` — Elegir las terminales, impresoras y básculas
- `correo-office365` — Conectar Ferrum con Microsoft 365

6. INFRAESTRUCTURA-DIMENSIONAR — Calcular qué servidor te toca (hoja de dimensionamiento)

Catálogo: Infraestructura · Requerimientos técnicos · Tiempo estimado: 15 minutos · Permiso requerido: `Ninguno (todos los usuarios)`

Qué vas a lograr

Llenar una hoja con los datos de tu empresa y obtener el perfil de servidor, la cantidad de terminales y el tipo de enlace que necesitas, lista para entregarla a Sistemas Ferrum o a tu proveedor de equipo.

Cuándo usar esto

Antes de cotizar equipo o internet, y cada vez que la empresa crezca: una sucursal nueva, diez cajas más, vendedores con celular. También cuando el sistema se siente lento y quieres saber si el servidor sigue en el perfil correcto.

📋 Antes de empezar

Necesitas…	Dónde se obtiene
Terminales simultáneas, locales y usuarios remotos	→ `infraestructura-conceptos`
Cuántas facturas, notas y compras se hacen al día en total	Reportes de ventas y compras del último mes
Antigüedad de la base de datos (si ya usan Ferrum)	Sistemas Ferrum, o `df -h /home/mysql` en el servidor

La hoja

Copia esta tabla y llena la columna de la derecha. Los renglones 8 a 12 se llenan con las reglas que siguen.

#	Dato	Tu valor
1	Terminales simultáneas en la hora pico (cajas, escritorios, celulares)	
2	Locales con servidor propio (deben vender sin internet)	
3	Locales sin servidor (trabajan como terminales remotas)	
4	Usuarios remotos ocasionales (dueño, contador, vendedores en ruta)	

#	Dato	Tu valor
5	Documentos al día en toda la empresa (ventas + compras + traspasos)	
6	Años de historia que se conservarán en línea	
7	Versión: Linux o Windows	
8	Carga equivalente del servidor central (regla A)	
9	Perfil del servidor central (regla B)	
10	Perfil del servidor de cada sucursal (regla C)	
11	Espacio en disco estimado (regla D)	
12	Tipo de enlace de internet (regla E)	

Reglas

Regla	Cómo se calcula
A · Carga equivalente	Terminales del propio local + terminales de los locales sin servidor + usuarios remotos + 2 por cada sucursal con servidor + 1 por el monitor de eventos.
B · Perfil del central	Hasta 5: Básico. 6 a 15: Mediano. 16 a 40: Grande. Más de 40, o 4 o más sucursales con servidor: Corporativo (→ `servidor-linux-requerimientos`).
C · Perfil de sucursal	Por las terminales de esa sucursal, con la misma tabla. Un local con 3 cajas lleva un servidor Básico aunque la matriz sea Corporativo.
D · Disco	Base de datos: 5 GB de arranque + 3 GB por cada 100 documentos diarios por cada año de historia. Multiplica por 3 para caber los siete respaldos diarios y la carpeta de salida. Nunca menos de 240 GB.
E · Enlace	Sin sucursales ni remotos: básico (50/10). Con sucursales o remotos: matriz con IP fija o dominio dinámico (100/20) y segundo enlace. Cada local sin servidor necesita su propio enlace de respaldo.

Tres ejemplos

Empresa	Datos	Resultado
Ferretería de un local	4 cajas, 1 escritorio, el dueño desde casa; 150 documentos al día; 5 años	Carga 7: perfil **Mediano** (por poco; con 4 núcleos, 16 GB y dos SSD de 480 GB en espejo va bien). Disco: 5 + 3 × 1.5 × 5 = 28 GB de base, × 3 = 84 GB: dos SSD de 480 GB en espejo. Enlace básico 50/10.
Distribuidora con 3 sucursales	Matriz con 8 terminales; 2 sucursales con servidor (5 y 3 cajas); 1 local chico sin servidor (2 cajas); 6 vendedores con celular; 600 documentos al día; 7 años	Carga central 8 + 2 + 6 + 4 + 1 = 21: **Grande**. Sucursales: Básico cada una. Disco: 5 + 3 × 6 × 7 = 131 GB de base, × 3 = 393 GB: dos NVMe de 1 TB en espejo. Matriz con IP fija, 100/20 y doble WAN; el local sin servidor con enlace de respaldo.
Cadena de 8 sucursales	Matriz con 12 terminales; 8 sucursales con servidor de 3 a 6 cajas; 15 remotos; 2,000 documentos al día; 10 años	Carga central 12 + 15 + 16 + 1 = 44 y 8 sucursales: **Corporativo**, con servidor de contingencia. Sucursales: Básico o Mediano. Disco: 5 + 3 × 20 × 10 = 605 GB de base, × 3 = 1.8 TB: dos NVMe de 2 TB en espejo y disco aparte para respaldos.

Las cifras de disco son estimaciones para cotizar, no medidas: una empresa real con varios años de operación pesa unos 30 GB de base, y el respaldo diario comprimido ronda los 5 GB.

Pasos

6.1 Llena los renglones 1 a 7 de la hoja con lo que contaste en → `infraestructura-conceptos`.

6.2 Aplica la regla A y anota la carga equivalente en el renglón 8.

6.3 Con la regla B, anota el perfil del servidor central en el renglón 9. Si el resultado queda en el límite de un perfil (5, 15, 40), toma el siguiente: la empresa crece.

6.4 Con la regla C, anota el perfil de cada sucursal con servidor en el renglón 10.

6.5 Calcula el disco con la regla D y anótalo en el renglón 11.

6.6 Decide el enlace con la regla E y anótalo en el renglón 12.

6.7 Entrega la hoja a Sistemas Ferrum antes de comprar: con ella se confirma el perfil y se preparan los usuarios, el `sql.ini` y los túneles.

Cómo verificar que funcionó

La hoja tiene los doce renglones llenos y el perfil del renglón 9 coincide con el que te confirma Sistemas Ferrum.

🔧 Para el administrador

De dónde salen las reglas. La regla A pondera cada sucursal que replica como dos terminales porque la matriz aplica su binlog completo y contesta su latido; el monitor de eventos es una sesión más. Los perfiles de la regla B salen del autoajuste del instalador (pool = 50 % de la RAM, tope 16 GB) y de la reserva de 150 MB por sesión: 16 GB dan pool de 8 GB y hasta 40 sesiones con holgura; 32 GB cubren 30 GB de base entera en memoria. La regla D se calibró con una base real de 30 GB tras varios años con unos 100 a 150 documentos diarios; ajústala con `SELECT table_schema, ROUND(SUM(data_length+index_length)/1024/1024/1024,1) FROM information_schema.tables GROUP BY table_schema` en una empresa parecida.

Qué revisar después de instalar. A las dos semanas corre `mysqltuner.pl` (queda en la carpeta del sistema) y revisa `Innodb_buffer_pool_reads` contra `read_requests`: si el pool no cubre el working set, sube la RAM antes que el procesador. Con `sysstat` instalado, `sar -d` muestra si el disco pasa del 60 % de utilización en hora pico; ése es el aviso para migrar a NVMe. El panel SISTEMAS → UTILERÍAS → ALERTAS DEL SISTEMA (derecho XB) reúne respaldo atrasado, disco, RAID, replicación y vencimientos.

Cuándo agregar servidor de contingencia. Desde el perfil Mediano, si la empresa no puede parar más de una hora: una máquina de perfil igual o un escalón menor, con `sql.ini [RESPALDO]` en todas las terminales, `auto_increment_offset` propio y réplica bidireccional con la matriz. El failover automático la usa sólo si el latido la reporta en línea y al corriente.

⚠️ Si algo sale mal

Síntoma	Causa probable	Qué hacer
El perfil te sale más alto de lo que esperabas	Sumaste usuarios registrados, o contaste dos veces a los remotos	Repite la regla A sólo con puestos abiertos a la vez.
Ya tienes servidor y el sistema va lento aunque el perfil coincide	Casi siempre es disco mecánico o falta de memoria para el pool, no el perfil	Verifica `ROTA` y `free -h` (→ `servidor-linux-requerimientos`) antes de comprar otro equipo.
El disco calculado es enorme por los años de historia	La regla D es conservadora y multiplica por 3 para respaldos	Mueve los respaldos a un disco aparte o a otro equipo y deja en el NVMe sólo la base.

💡 **Tip**	Guarda la hoja llena junto con la cotización del equipo. Cuando en dos años el sistema se sienta lento, compararás en un minuto lo que tienes con lo que la empresa necesita hoy.

Ver también

- `infraestructura-conceptos` — Entender cómo está armado Ferrum y elegir la versión
- `servidor-linux-requerimientos` — Elegir el servidor para la versión Linux

- `red-internet-requerimientos` — Preparar la red local y el enlace de internet

Glosario del módulo

Los términos que se usan en este módulo, en orden alfabético. Si buscas uno que no está aquí, revisa el glosario de los otros módulos del programa.

- **Ancho de banda**: La cantidad de datos por segundo que pasa por tu enlace de internet. Se mide en megabits por segundo (Mbps). Bajada es lo que recibes; subida es lo que envías.
- **Buffer pool**: La memoria que MariaDB reserva para tener la base de datos a la mano sin ir al disco. Ferrum le da la mitad de la RAM del servidor.
- **Contingencia**: El servidor de reserva al que se pasan las terminales cuando el servidor de trabajo deja de responder. En el archivo de conexión es la sección `RESPALDO`.
- **CUPS**: El sistema de impresión del servidor Linux. Ahí se dan de alta las impresoras que usan las terminales.
- **DDNS (dominio dinámico)**: Un nombre fijo, como `misucursal.dynns.com`, que apunta a la dirección de internet de tu servidor aunque el proveedor la cambie cada día. Lo mantiene actualizado el programa `ddclient`.
- **Disco SSD / NVMe**: Discos de estado sólido, sin partes móviles. Un SSD es de 10 a 50 veces más rápido que un disco mecánico para el trabajo de una base de datos; NVMe es la variante más rápida de SSD.
- **Doble WAN**: Dos enlaces de internet de proveedores distintos conectados al servidor; si uno falla, el servidor sale por el otro sin intervención.
- **Espejo (RAID 1)**: Dos discos que guardan exactamente lo mismo. Si uno se daña, el servidor sigue trabajando con el otro. No sustituye al respaldo.
- **fail2ban**: Programa del servidor que bloquea las direcciones desde las que alguien intenta adivinar contraseñas.
- **Firewall**: El filtro que decide qué conexiones entran y salen de un equipo o de una red. Ferrum necesita que ciertas salidas estén permitidas (→ `red-internet-requerimientos`).
- **IP pública fija**: La dirección con la que tu enlace de internet se ve desde afuera, contratada para que no cambie. La necesita la matriz que recibe sucursales; una alternativa más barata es el dominio dinámico.
- **Latencia**: El tiempo que tarda un dato en ir y volver por la red, en milisegundos (ms). Para una terminal remota importa más que el ancho de banda: con 200 ms cada tecla se siente lenta.
- **Latido**: La señal que cada 10 segundos intercambian la matriz y las sucursales para saber quién está en línea y al corriente.
- **Matriz**: El servidor central de la empresa. Recibe los cambios de todas las sucursales y les reenvía los suyos.
- **Microsoft Edge**: El navegador que viene con Windows 10 y 11. En la versión Windows, Ferrum lo usa por debajo para armar los reportes en PDF.
- **No-break (UPS)**: Batería que mantiene encendido el servidor cuando se va la luz, el tiempo suficiente para apagarlo en orden. Es obligatorio para el servidor.
- **Núcleo**: Cada unidad de cálculo dentro del procesador. Un procesador de 4 núcleos atiende cuatro trabajos a la vez.
- **PAC**: El proveedor autorizado por el SAT que timbra las facturas. Ferrum trabaja con Facture Hoy.
- **Puerto**: El número que identifica cada servicio dentro de una conexión de red: el 22 es acceso remoto seguro, el 3306 es la base de datos, el 443 es web cifrada.

- **PuTTY**: Programa gratuito para Windows que abre una conexión segura (SSH) a un servidor Linux y muestra su pantalla. Es la terminal más usada de Ferrum.
- **Réplica / replicación**: El mecanismo con el que el servidor de cada sucursal copia sus cambios a la matriz y recibe los de ella, de forma continua y automática.
- **serbas**: Programa pequeño que se instala junto a una báscula y la comparte por la red para que cualquier terminal la lea.
- **Servidor**: La computadora que guarda la base de datos y, en la versión Linux, también corre el programa de todas las terminales. Es el único equipo que no se puede apagar.
- **SumatraPDF**: Programa gratuito y ligero que Ferrum usa en Windows para mandar un PDF a la impresora sin preguntar nada.
- **Terminal**: Cada puesto de trabajo desde el que una persona usa Ferrum: una caja, un escritorio de compras, el celular del vendedor.
- **Termux**: Aplicación para Android que ofrece una terminal Linux; con el menú de Ferrum se conecta a los servidores con un toque.
- **Túnel SSH inverso**: La conexión cifrada que abre el servidor de una sucursal hacia la matriz para que ésta lo alcance sin que la sucursal tenga IP pública ni puertos abiertos.
- **Versión Linux**: La forma de trabajo en la que el programa corre en el servidor Ubuntu y las terminales sólo muestran la pantalla por una conexión de red segura (SSH).
- **Versión Windows**: La forma de trabajo en la que cada computadora tiene el programa `ferrumop.exe` instalado y se conecta directamente a la base de datos del servidor.
- **WeasyPrint**: El programa del servidor Linux que convierte los reportes a PDF con formato. Sin él los PDF salen con presentación pobre.

Anexo. INFRAESTRUCTURA-TABLA-HARDWARE — Tabla resumen de hardware

Mín. (mínimo): lo necesario para funcionar con esa carga. Rec. (recomendado): lo que conviene comprar hoy para cinco años. Los perfiles se eligen por las terminales abiertas a la vez en ese servidor (una sucursal que replica cuenta como dos). El detalle y los pasos de verificación están en las secciones 2 a 5.

Característica	Versión Linux · servidor (por terminales a la vez)								Versión Windows						Terminal Linux	
	Básico 1 a 5		Mediano 6 a 15		Grande 16 a 40		Corporativo +40 o 4+ sucursales		Servidor de BD 1 a 5 PCs		Servidor de BD 6 a 15 PCs		PC con el sistema (cada una)		Cualquier cantidad	
	Mín.	Rec.	Mín.	Rec.	Mín.	Rec.	Mín.	Rec.	Mín.	Rec.	Mín.	Rec.	Mín.	Rec.	Mín.	Rec.
Procesador	2 núcleos	4 núcleos	4 núcleos	6 núcleos	6 núcleos	8 núcleos	8 núcleos	16 núcleos	2 núcleos	4 núcleos	4 núcleos	6 núcleos	2 núcleos	4 núcleos	De los últimos 10 años	—
Memoria RAM	8 GB	16 GB	16 GB	32 GB	32 GB	64 GB	64 GB	128 GB	8 GB	16 GB	16 GB	32 GB	4 GB	8 GB	2 GB	4 GB
Disco	SSD 240 GB	2 SSD 480 GB en espejo	SSD 480 GB	2 NVMe 1 TB en espejo	2 NVMe 1 TB en espejo	2 NVMe 2 TB en espejo	2 NVMe 2 TB en espejo	2 NVMe 2 TB en espejo + disco de respaldo	SSD 240 GB	2 NVMe 1 TB en espejo	SSD 240 GB	2 NVMe 1 TB en espejo	SSD, 10 GB libres	SSD 240 GB	No requiere	—
Sistema operativo	Ubuntu Server 20.04 LTS	Ubuntu Server 24.04 LTS	Ubuntu Server 20.04 LTS	Ubuntu Server 24.04 LTS	Ubuntu Server 20.04 LTS	Ubuntu Server 24.04 LTS	Ubuntu Server 20.04 LTS	Ubuntu Server 24.04 LTS	Windows 10/11 Pro 64 bits	Servidor Linux (perfil Básico)	Windows 10/11 Pro 64 bits	Windows Server 2019/2022 o Linux	Windows 10 64 bits (1803+)	Windows 11 Pro 64 bits	Windows, Linux, Mac o Android	Igual
Base de datos	MariaDB 10.3	MariaDB 10.11	MariaDB 10.3	MariaDB 10.11	MariaDB 10.3	MariaDB 10.11	MariaDB 10.3	MariaDB 10.11	MariaDB 10.6 para Windows	MariaDB 10.11	MariaDB 10.6 para Windows	MariaDB 10.11	—	—	—	—
Red	1 Gbps, cable	1 Gbps, cable	1 Gbps, cable	1 Gbps, cable	1 Gbps, cable	1 Gbps, cable	1 Gbps, cable	1 Gbps, cable	100 Mbps, cable	1 Gbps, cable	100 Mbps, cable	1 Gbps, cable	100 Mbps, cable	1 Gbps, cable	100 Mbps, cable	1 Gbps, cable; wifi sólo celulares
No-break	10 min (servidor y switch)	20 min; se apaga solo	10 min (servidor y switch)	20 min; se apaga solo	10 min (servidor y switch)	20 min; se apaga solo	10 min (servidor y switch)	20 min; se apaga solo	10 min (servidor y switch)	20 min; se apaga solo	10 min (servidor y switch)	20 min; se apaga solo	Conviene	Sí	No requiere	—
Servidor de contingencia	No	Opcional	Opcional	Sí, si no puede parar 1 hora	Conviene	Sí	Sí	Sí	No	Opcional	Opcional	Conviene	—	—	—	—
Pantalla y teclado	No requiere	—	No requiere	—	No requiere	—	No requiere	—	No requiere	—	No requiere	—	17 pulg.; teclado con F1-F12	19 pulg. o más	15 pulg.; teclado con F1-F12	19 pulg. o más
Programas	Instalador de Ferrum	+ firewall y fail2ban	Instalador de Ferrum	+ firewall y fail2ban	Instalador de Ferrum	+ firewall y fail2ban	Instalador de Ferrum	+ firewall y fail2ban	MariaDB como servicio	Igual	MariaDB como servicio	Igual	Edge + SumatraPDF	Igual	PuTTY 0.71 o más nuevo	Igual
Enlace de internet	10/3	50/10, fibra	10/3	50/10, fibra	20/10, IP fija o DDNS	100/20, fibra, doble WAN	20/10, IP fija	100/20, fibra, doble WAN	10/3	50/10, fibra	10/3	50/10, fibra	El del local	—	Remota: 1 Mbps, máx. 150 ms	Máx. 80 ms

Espejo = dos discos iguales en RAID 1. Doble WAN = dos enlaces de proveedores distintos. Enlace = bajada/subida en Mbps. La PC con el sistema y la terminal Linux usan el internet del local; la cifra de la terminal aplica sólo cuando entra desde fuera.