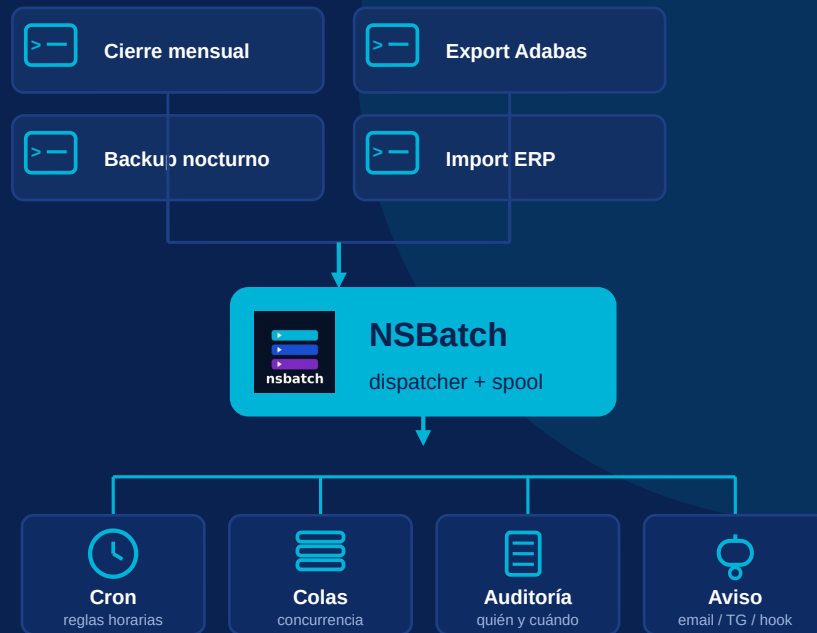


Sus procesos batch, en cola, con control y trazabilidad

nsbatch encola, planifica, ejecuta y audita cada proceso del servidor: control tipo cron, colas con concurrencia, spool en vivo y aviso por email, Telegram o webhook.

[VER FUNCIONALIDADES →](#)

Control cron · Colas · Auditoría · Notificación



...y cada ejecución queda registrada

Queues · jobs · executions · spool · audit · notify

Los batch se lanzan a ciegas

Scripts sueltos en cron: sin control de concurrencia, sin spool central y sin saber qué está corriendo ahora mismo.

01 Sin orden

Un cierre contable y un backup compiten por la CPU. El primero que llega, manda.

02 Sin visibilidad

Nadie sabe qué corre, desde cuándo, con qué PID ni cuánto ha costado.

03 Sin traza

Los logs se pierden en /tmp. Al día siguiente no hay forma de reconstruir la ejecución.

04 Sin control

Cualquiera lanza y cualquiera borra. Sin roles, sin permisos y sin registro.

• Horas de retraso

• Errores sin ver

• Cero trazabilidad

QUÉ ES

Un gestor de colas de procesos, al estilo del mainframe

nsbatch es el JES de Nacarsoft: cada trabajo entra en una cola, el dispatcher lo ejecuta con un PID aislado y su spool queda en disco. Lo lanza la web, el CLI, la API o una regla cron; y siempre queda quién lo hizo y cuándo.

- ✓ Colas con prioridad, concurrencia y ventanas horarias
- ✓ Reglas tipo cron, con importación del crontab que ya tenía
- ✓ Un PID aislado por ejecución: si el proceso muere, el gestor sigue
- ✓ Spool en disco con tail en vivo, búsqueda y descarga

4

colas por defecto

9

permisos RBAC

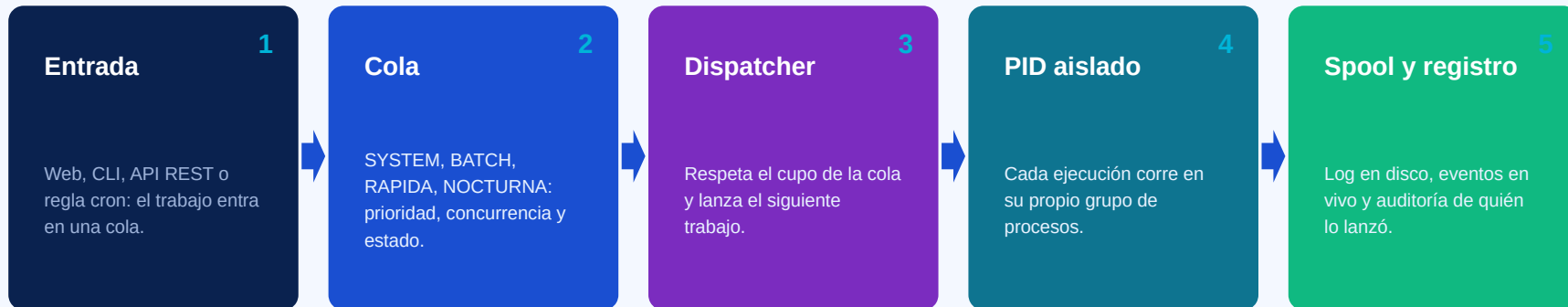
6

idiomas y temas

1

solo binario

Cómo viaja un proceso, paso a paso



⚠ Si un trabajo se cuelga, el timeout mata su árbol de procesos: el dispatcher nunca se bloquea.

Sin cron externo

Sin scripts sueltos

Todo en un binario

Cuatro maneras de lanzar el mismo trabajo

1

A mano, web o CLI

`nsbatch submit CIERRE --params "--modo full"`: encola y devuelve el ID de la ejecución.

2

Por API

`POST /api/jobs/{id}/submit`: tu ERP o Natural encolan el trabajo sin tocar la consola.

3

Por regla cron

Hora y días por trabajo, con la próxima ejecución calculada y opción de recuperar la ventana perdida.

4

Importando el crontab

Pega el crontab que ya tenías: `nsbatch` lo analiza y deja el trabajo planificado.

Lo que hoy vive en `/etc/crontab` pasa al panel: con estado, con pool y con aviso.

La planificación, pero con estado y con aviso

- Regla por trabajo: hora, días de la semana y días del mes
- Próximas ejecuciones calculadas y mostradas en la vista global
- Recuperación de la ventana perdida (catch-up) al arrancar
- Historial de la regla: qué se ejecutó y con qué resultado
- Importación del crontab existente y vista global de reglas



0 2 * * 6 → BACKUP-NOCHE

sábados y domingos a las 02:00, cola NOCTURNA

ID	TRABAJO	HORARIO	COLA	USUARIO	ON	RUES	ULTIMA	PRÓXIMA	ACCIONES
#3	BACKUP-NOCHE /tmp/openorder/ada/si/backup.sh --dest /backup	02:00 dom 1	NOCTURNA	cli	Activa	▶ 0 m0	—	2026-10-24 02:00	🔍 📅 ✕
#4	REORD-CUSTOMERS /tmp/openorder/ada/si/cierre.sh --file 100	03:15 vie 3	SYSTEM	cli	Activa	▶ 0 m0	—	2026-10-21 03:15	🔍 📅 ✕
#2	EXPORT-LEDGER /tmp/openorder/ada/si/export.sh --format csv	06:45 dom 1,2,3,4,5	RAPIDA	cli	Activa	▶ 0 m0	—	2026-10-30 06:45	🔍 📅 ✕
#5	IMPORT-ORDENES cp -f /tmp/openorder/...	07:00 dom 1,2,3,4,5	BATCH	cli	Activa	▶ 0 m0	—	2026-10-30 07:00	🔍 📅 ✕
#1	CIERRE-MENSUAL month Totals	23:30 vie 3	BATCH	cli	Activa	▶ 0 m0	—	2026-10-19 23:30	🔍 📅 ✕

Planificadas: todas las reglas, su próxima ejecución, sus ejecuciones y su historial.

COLAS

Colas con concurrencia, prioridad y color

SYSTEM

max 2

prio 90

RAPIDA

max 6

prio 70

BATCH

max 4

prio 50

NOCTURNA

max 3

prio 30

OPEN

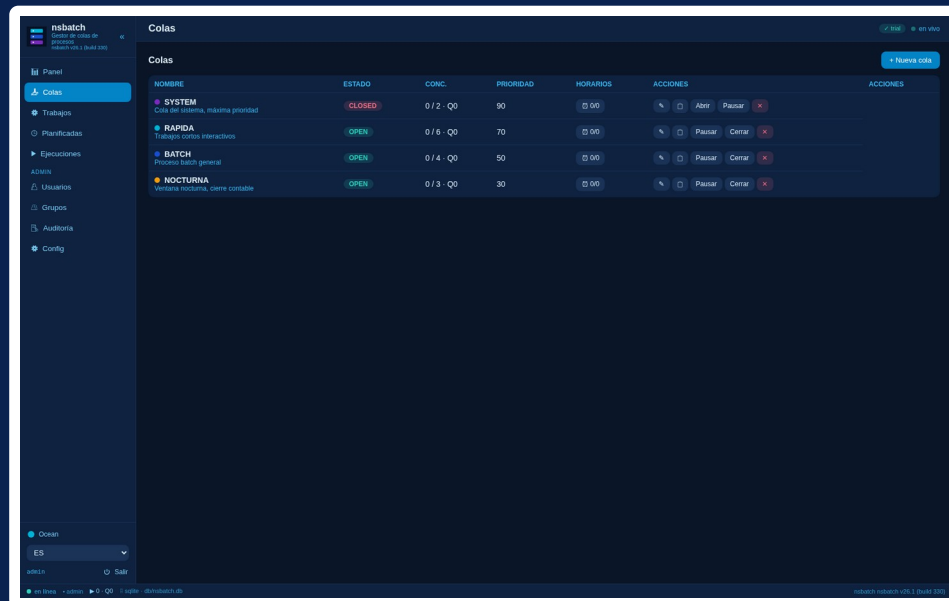
PAUSED

CLOSED

OPEN ejecuta · PAUSED retiene · CLOSED rechaza

- Cuatro colas de fábrica: SYSTEM, BATCH, RAPIDA y NOCTURNA
- Cupo de ejecuciones simultáneas y prioridad por cola
- OPEN, PAUSED y CLOSED: se pausa o cierra sin perder el trabajo
- Ventanas horarias por cola y mover un trabajo encolado a otra

Nacarsoft Soluciones Informáticas S.L. · nacarsoft.com · [nsbatch](https://nscarsoft.com/nsbatch)



Colas: cupo ocupado, prioridad, color y encolados de cada una.

El catálogo: un trabajo es un comando y sus reglas

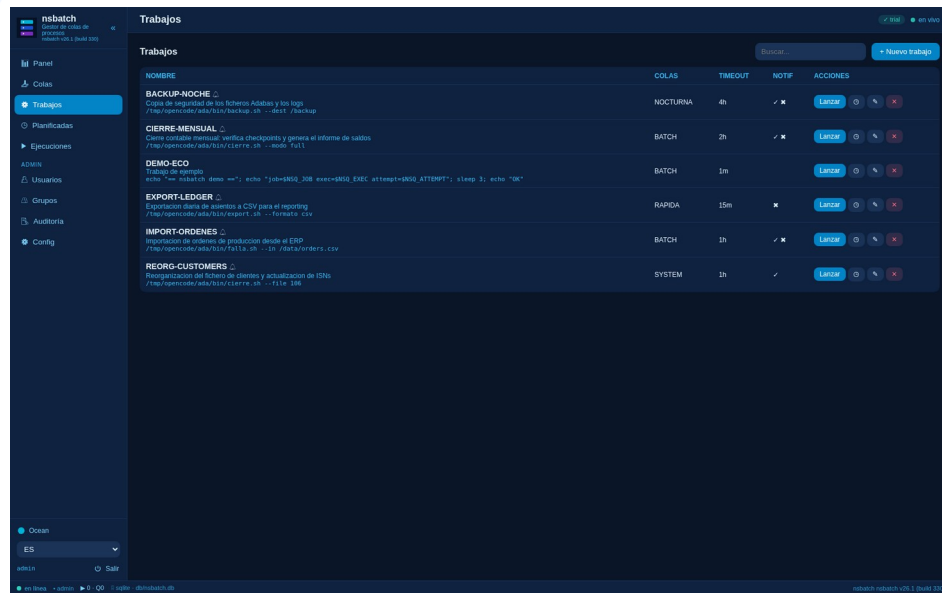
- Comando y argumentos: el trabajo es una línea de shell real
- Cola de destino y parámetros de cada ejecución
- Timeout por trabajo, con kill del árbol de procesos
- Avisos de inicio, fin y error, con destinatarios
- Reintentos y re-ejecución con los mismos parámetros

CIERRE-MENSUAL

```
/opt/ada/bin/cierre.sh --modo full
```

BATCH · timeout 7200s · aviso al terminar y al fallar

Un mismo trabajo puede estar asignado a varias colas: la que elija el operador o la de su regla cron.



The screenshot shows the nsbatch web interface. On the left is a sidebar with navigation links: Planet, Colas, Trabajos (selected), Planificadas, Ejecuciones, Admin, Usuarios, Grupos, Auditoría, and Config. The main panel is titled 'Trabajos' and contains a table of jobs. The table has columns for 'NOMBRE', 'COLAS', 'TIMEOUT', 'NOTIF', and 'ACCIONES'. The jobs listed are: BACKUP-NOCHE (NOCTURNA, 4h), CIERRE-MENSUAL (BATCH, 2h), DEMO-ECO (BATCH, 1m), EXPORT-LEDGER (RAPIDA, 15m), IMPORT-ORDENES (BATCH, 1h), and REORG-CUSTOMERS (SYSTEM, 1h). Each row has a 'Lanzar' button and icons for edit, delete, and refresh.

NOMBRE	COLAS	TIMEOUT	NOTIF	ACCIONES
BACKUP-NOCHE	NOCTURNA	4h	✓	Lanzar
CIERRE-MENSUAL	BATCH	2h	✓	Lanzar
DEMO-ECO	BATCH	1m	✓	Lanzar
EXPORT-LEDGER	RAPIDA	15m	✓	Lanzar
IMPORT-ORDENES	BATCH	1h	✓	Lanzar
REORG-CUSTOMERS	SYSTEM	1h	✓	Lanzar

Trabajos: comando, cola, timeout, avisos y reintentos de cada uno.

EJECUCIONES

Cada ejecución deja su spool

QUEUED

En la cola,
esperando turno

HOLD

Retenido a
mano

RUNNING

Con PID propio

OK

Terminó bien

ERROR

Terminó con
error

- Estados: QUEUED, HOLD, RUNNING, OK, ERROR y CANCELLED
- Un PID por ejecución y número de intento
- Exit code, tiempo de pared, CPU y memoria pico
- Spool en disco con tail en vivo, búsqueda y descarga
- Cancelar, retener, mover o re-ejecutar sin salir del panel

PID 2535367 · exit 0 · wall 2s · mem 1.7 MB

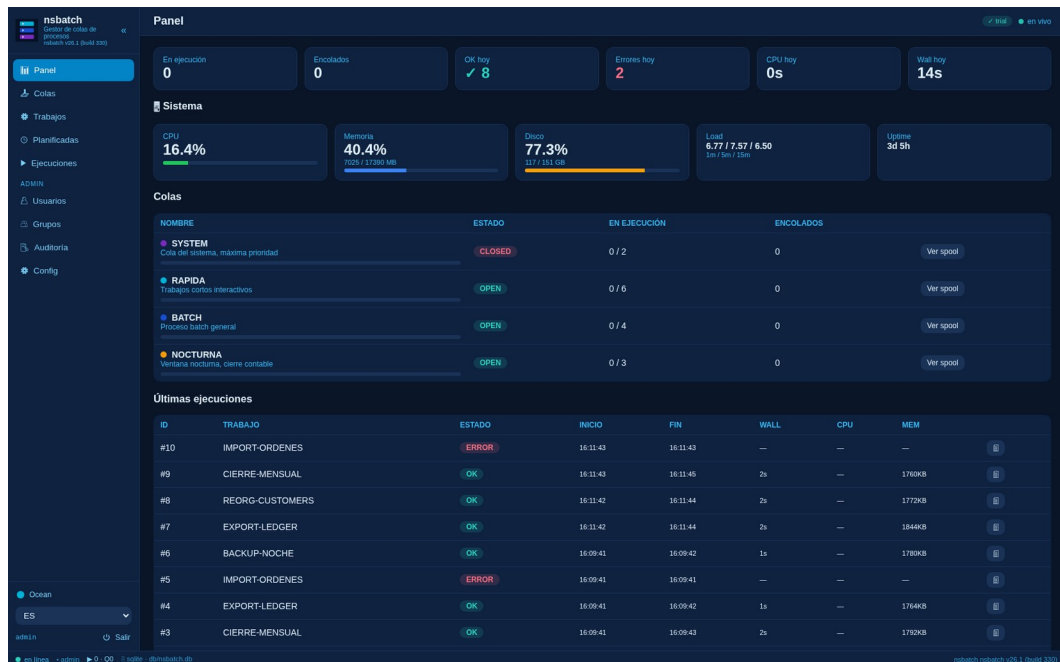
The screenshot shows the nsbatch web interface. On the left is a sidebar with navigation links: Panel, Cola, Trabajos, Ejecuciones (selected), Acciones, Usuarios, Grupos, Ayuda, and Config. The main area displays a table titled 'Ejecuciones' with columns: ID, TRABAJO, COLA, ESTADO, PID, CREADA, and ACCIONES. The table lists several jobs, including 'IMPORT-ORDENES', 'CIERRE-MENSUAL', 'REGORG-CUSTOMERS', and 'EXPORT-LED'. A modal window is open over the 'EXPORT-LED' job, showing details for 'spool #10' which is in an 'ERROR' state. The log content in the modal is as follows:

```
== nsbatch spool == exec=10 job=IMPORT-ORDENES attempt=1 started=2026-10-10T18:11:43+02:00
== comando == /bin/bash -c 'cat /dev/null > /dev/null' --sh /data/orders.csv
== importación de órdenes ==
Logando /data/orders.csv
ERROR: no such file or directory: /data/orders.csv
== wall=0.000 user=0.000 sys=0.000 que total=0.000 peak_rss=0MB exit=2 (ERROR)
== notify_email == produccion@empresa.com: ERROR: setup no configurado (host requerido)
== notify_email == produccion@empresa.com: ERROR: setup no configurado (host requerido)
```

At the bottom of the modal are buttons for 'Descargar log', 'Tail', and 'Cerrar'.

El spool: cabecera con el job, el intento, las fechas y el exit code, y el log completo.

Todo el estado en una pantalla



- KPIs vivos: en ejecución, encolados, OK y errores de hoy
- CPU y wall de hoy, con el consumo real de cada trabajo
- Semáforo por cola con el uso del cupo de concurrencia
- Últimas ejecuciones con su PID y su spool a un clic
- Estado del sistema: CPU, memoria, disco, carga y uptime

En ejecución

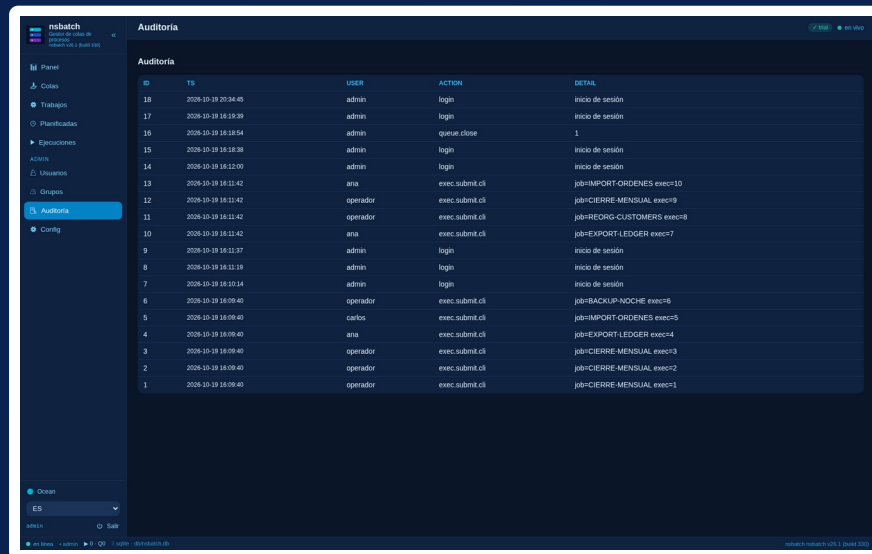
Encolados

OK hoy

Errores hoy

Quién lanzó qué, y queda registrado

- Cada acción del panel entra en el registro de auditoría
- Usuario, acción, detalle, hora y cola afectada
- Altas de trabajos y de colas, cambios de estado y de permisos
- Cancelaciones, re-ejecuciones y movimientos entre colas
- Búsqueda y filtrado por usuario, acción y fecha



The screenshot shows the 'Auditoria' (Audit) section of the nsbatch application. It features a sidebar with navigation options like 'Panel', 'Colas', 'Trabajos', 'Planificadas', 'Ejecuciones', 'Admin', 'Usuarios', 'Grupos', 'Auditoria' (selected), and 'Config'. The main area displays a table of audit logs with columns for ID, TS (Timestamp), USER, ACTION, and DETAIL. The table lists various system events such as logins, session starts, and job submissions.

ID	TS	USER	ACTION	DETAIL
18	2026-10-19 20:04:40	admin	login	Inicio de sesión
17	2026-10-19 18:19:39	admin	login	Inicio de sesión
16	2026-10-19 18:18:54	admin	queue.close	1
15	2026-10-19 18:18:38	admin	login	Inicio de sesión
14	2026-10-19 18:12:00	admin	login	Inicio de sesión
13	2026-10-19 18:11:42	ana	exec.submit.cl	job=IMPORT-ORDENES exec=10
12	2026-10-19 18:11:42	operador	exec.submit.cl	job=CIERRE-MENSUAL exec=9
11	2026-10-19 18:11:42	operador	exec.submit.cl	job=REORG-CUSTOMERS exec=8
10	2026-10-19 18:11:42	ana	exec.submit.cl	job=EXPORT-LEDGER exec=7
9	2026-10-19 18:11:37	admin	login	Inicio de sesión
8	2026-10-19 18:11:19	admin	login	Inicio de sesión
7	2026-10-19 18:10:14	admin	login	Inicio de sesión
6	2026-10-19 18:09:40	operador	exec.submit.cl	job=BACKUP-NOCHE exec=6
5	2026-10-19 18:09:40	carlos	exec.submit.cl	job=IMPORT-ORDENES exec=5
4	2026-10-19 18:09:40	ana	exec.submit.cl	job=EXPORT-LEDGER exec=4
3	2026-10-19 18:09:40	operador	exec.submit.cl	job=CIERRE-MENSUAL exec=3
2	2026-10-19 18:09:40	operador	exec.submit.cl	job=CIERRE-MENSUAL exec=2
1	2026-10-19 18:09:40	operador	exec.submit.cl	job=CIERRE-MENSUAL exec=1

Immutable, con usuario e IP, y exportable desde el panel.

Auditoría: qué ha pasado en el servidor y quién lo ha hecho.

NOTIFICACIÓN

El proceso avisa cuando termina

Cada trabajo decide si avisa al empezar, al terminar o solo cuando hay error. nsbatch manda el aviso por los canales que queráis y deja constancia en el spool.

Email (SMTP)

STARTTLS o TLS, con usuario y remitente configurables. Varios destinatarios por trabajo.

Telegram

Bot + chat ID: el aviso llega al móvil del equipo sin montar ningún servidor de correo.

Webhook

POST a la URL que defináis, con cabeceras propias: se integra con Slack, Teams o tu ITSM.

Termina la ejecución

OK, ERROR o CANCELLED



Se compone el aviso

job, intento, params, wall, CPU y exit



Sale por los canales

email · Telegram · webhook



Queda en el spool

ok o ERROR de cada canal

Destinatarios por trabajo, por grupo de usuarios o por el creador de la ejecución.

Para integrarlo con lo que ya tienes

CLI

```
nsbatch submit <job> --params "..."
```

```
nsbatch queues list | create | pause | open |  
close
```

```
nsbatch jobs list | create | update | cron
```

```
nsbatch execs [ESTADO] | cancel | rerun | move |  
log
```

```
nsbatch users | groups | roles | audit | dashboard
```

REST

```
POST /api/jobs/{id}/submit
```

```
GET /api/queues · /api/jobs · /api/executions
```

```
GET /api/cron · /api/cron/next · /api/cron/import
```

```
GET /api/executions/{id}/log
```

```
GET /api/events (SSE en vivo)
```

La consola es una de las vías: la API y el CLI hacen lo mismo sin abrir el navegador.

Quién ve qué, y queda registrado

Control de acceso

- RBAC con 9 permisos
- Usuarios, grupos y roles
- Permiso por operación
- Sin admin binario

Sesión y acceso

- JWT HS256 en cookie
- Contraseñas con bcrypt
- Recuperación por email
- Caducidad configurable

Trazabilidad

- Auditoría de cada acción
- Usuario e IP de cliente
- Log por cola y por trabajo
- Retención configurable

config.yaml 0600 · base de datos 0700 · bcrypt · JWT HS256 · trial 30 días · licencia por máquina

CARACTERÍSTICAS

Todo lo que un operador necesita

6 temas

Dark, Ocean, Light, Forest, Sunset y alto contraste.

6 idiomas

ES, EN, FR, DE, IT y PT en toda la interfaz.

RBAC

Usuarios → grupos → roles → permisos.
Nada de admin binario.

API + SSE

Integra tus scripts: submit por REST y eventos en vivo.

Timeouts

Timeout y kill del árbol de procesos por ejecución.

Reintentos

Re-ejecuta con los mismos parámetros en un clic.

Retención

Purga automática de spools antiguos e histórico de ejecuciones.

Licencia

Trial de 30 días y licencia firmada por máquina.

De cero a procesos en cola, en tres pasos

1

Crear el entorno

nsbatch init /opt/nsbatch: crea config.yaml, la base SQLite, el spool y el usuario admin.

2

Ajustar y arrancar

Rellena el secreto JWT, el puerto y los canales de aviso. Arranca con nsbatch start.

3

Dar de alta el trabajo

Crea la cola, el trabajo y su regla cron, o importa el crontab que ya tenías.

REQUISITOS

- Linux (RHEL 8 / 9)
- Un solo binario, sin dependencias
- SQLite para el estado y el spool
- Proxy inverso con HTTPS en producción
- Panel web en el puerto 11050

```
nsbatch init /opt/nsbatch → nsbatch start /opt/nsbatch → panel web :11050
```

Lo que más se pregunta

¿Tengo que quitar el cron del sistema?

No hace falta, pero puede: nsbatch importa el crontab existente y lo deja planificado con su estado, su spool y sus avisos.

¿Puedo tener varios entornos?

Sí: cada instancia tiene su directorio, su base, su spool y su puerto. Se registran con nombre para cambiar rápido.

¿Se puede integrar con mi ERP?

Sí: la API REST y el CLI exponen el mismo catálogo de trabajos. POST /api/jobs/{id}/submit encola la ejecución.

¿Y si un trabajo se cuelga?

El timeout del trabajo mata su árbol de procesos y la ejecución pasa a ERROR. El dispatcher sigue con la cola.

¿Quién puede lanzar un trabajo?

Cada permiso es una operación: encolar, cancelar, re-ejecutar, mover, ver el spool o gestionar usuarios. Se asigna por rol y grupo.

¿Qué se ve en la auditoría?

Cada acción del panel: quién encoló, quién canceló, quién creó una cola o cambió un permiso, con usuario, hora e IP.

Sus procesos, en cola y con nombre y apellidos

Pruebe nsbatch con sus propios trabajos: se lo montamos y lo vemos funcionando con sus datos.

Demo guiada

Pruébalo en su entorno

Hablemos de su caso

nacarsoft.com · soporte@nacarsoft.com

