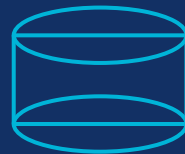


Get your Adabas data into CSV, JSON or XML. Without touching production.

nsadafile extracts records from Adabas files in read-only, non-blocking mode with a single command. Millions of records, no locks, no core downtime.

SEE THE FEATURES →

No locks · No core downtime · CSV · JSON · XML · TAB · RECORD



Adabas file

paged, non-blocking reads



CSV · JSON · XML

one command, your format



THE PROBLEM

Getting data out of Adabas should not hurt

Extracting information from the core usually means either custom scripts or taking the system down while the extract runs.

1

The core gets blocked

Extracts stress the files: locks, queues and waits that end up as incidents in production.

2

Every query is a new build

One custom script per need, nothing reusable, no documentation.

3

Formats everywhere

CSV here, JSON there, XML in the report: every consumer ends up writing its own converter.

nsadafile turns extraction into a safe, repeatable command-line operation.

WHAT IT IS

Nacarsoft's Adabas to file extractor

nsadafile connects through the native Adabas client in read-only mode and extracts exactly the records you ask for, in the format you want, at the pace the system allows.

It never writes to the core, never takes locks and needs no development: one binary, one command, one output file. It fits scripts, cron jobs and integration pipelines.

0

writes to the core

6

output formats

1

command



nsadafile

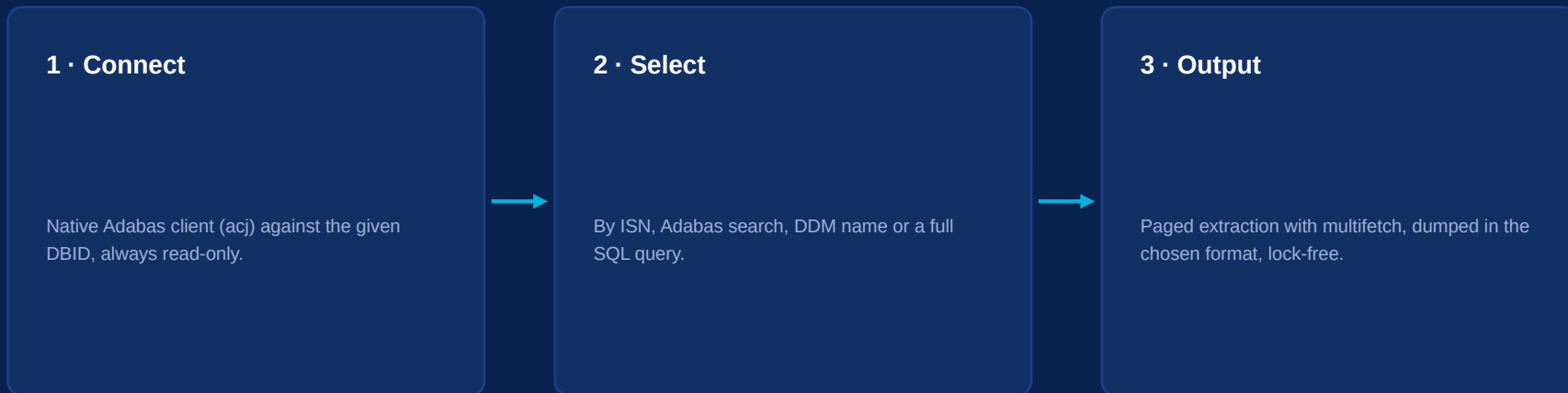
Adabas → file, read-only mode

- No locks: paged and non-blocking reads
- SQL, Adabas search, ISN or DDM name
- CSV, JSON, XML, TAB or RECORD output
- Field auditing with -checkfields

HOW IT WORKS

From the core to your file, in one command

Native connection, impact-free reads and output straight into the format your systems consume.

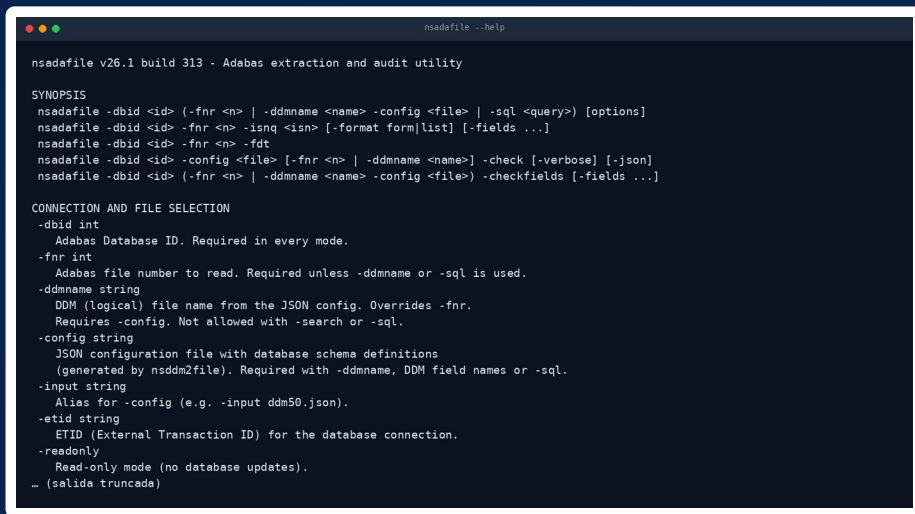


The core keeps serving traffic while you extract: it is a read, not a maintenance window.

THE TOOL

The whole product is in --help

One utility, no dependencies, no prior configuration. The manual is the program's own help.



```
nsadafile v26.1 build 313 - Adabas extraction and audit utility

SYNOPSIS
nsadafile -dbid <id> [-fnr <n> | -ddmname <name> -config <file> | -sql <query>) [options]
nsadafile -dbid <id> -fnr <n> -isnq <isn> [-format form[list] [-fields ...]
nsadafile -dbid <id> -fnr <n> -fdt
nsadafile -dbid <id> -config <file> [-fnr <n> | -ddmname <name>] -check [-verbose] [-json]
nsadafile -dbid <id> (-fnr <n> | -ddmname <name> -config <file>) -checkfields [-fields ...]

CONNECTION AND FILE SELECTION
-dbld int
  Adabas Database ID. Required in every mode.
-fnr int
  Adabas file number to read. Required unless -ddmname or -sql is used.
-ddmname string
  DDM (logical) file name from the JSON config. Overrides -fnr.
  Requires -config. Not allowed with -search or -sql.
-config string
  JSON configuration file with database schema definitions
  (generated by nsddm2file). Required with -ddmname, DDM field names or -sql.
-input string
  Alias for -config (e.g. -input ddm50.json).
-etid string
  ETID (External Transaction ID) for the database connection.
-readonly
  Read-only mode (no database updates).
_ (salida truncada)
```

Real output of nsadafile --help

- No client installation
- No DSN or profiles
- No libraries to maintain

Full SQL, not just "export everything"

Besides ISN, Adabas search and DDM name, nsadafile understands SQL: SELECT, JOIN, WHERE, GROUP BY, ORDER BY, LIMIT and DISTINCT.

```
nsadafile --help - opción -sql

-sql string
  SQL SELECT statement: SELECT, FROM, WHERE, JOIN (INNER, LEFT, RIGHT,
  FULL), ORDER BY, GROUP BY, LIMIT, OFFSET, DISTINCT.
  Mutually exclusive with -ddmname, -fmr and -search.
-search string
  Adabas search syntax, by short name (AA) or DDM name.
  Numeric: "AA=12345678" String: "BB='SMITH'" Range: "CC=[10:20]"
  Logic: "AA=1 and BB='X'", "BB='X' or BB='Y'". Not allowed with -ddmname or -sql.
-sortby string
  Descriptor/superdescriptor for logical read order (short or DDM name).
```

- JOIN across files
- Filters and aggregations
- LIMIT for sampling

Real excerpt of the nsadafile help (-sql option)

FEATURES

What it does, point by point

Built for the daily work of a data team or a DBA.



Six output formats

CSV, JSON, XML, tab-delimited (TAB) and record (RECORD), plus file output.



Optimised search

Adabas search syntax by short name or DDM name, with AND and OR logic.



Field auditing (-checkfields)

Finds the records whose values exceed the length defined in the DDM.



Record limit

Run sample extractions for testing without touching the whole file.



Field and ISN selection

Extract only the columns you need and optionally include the ISN of each record.



SQL queries

SELECT with JOIN, WHERE, GROUP BY, ORDER BY, LIMIT, OFFSET and DISTINCT.



Non-blocking mode

Paged reads with multifetch: the core keeps serving your users.



Execution trace

Elapsed time and system information reported at the end of the run.

KEY OPTIONS

The parameters used every day

They combine with each other; every other flag is in --help.

-db id <id>

Source Adabas database. Required in every mode.

-fn r <n> | -ddm name

File by number or by logical DDM name (requires -config).

-isnq <isn> | -search

A single ISN or an Adabas search with AND/OR logic.

-sql <query>

SELECT with JOIN, WHERE, GROUP BY, ORDER BY and LIMIT.

-fields <list>

Fields to extract (short or DDM name), comma separated.

-check fields

Auditing: checks that fields do not exceed the definition.

-form at <csv |json... >

Output format: csv, json, xml, tab (list) or record (form).

-lim it <n>

Maximum number of records to process (0 = no limit).

USE CASES

Where it is used every day

The same use cases we solve with customers in banking, healthcare and the public sector.

01

Feeding a data lake

Nightly extraction to CSV or Parquet for the data team, without touching the core.

02

Auditing and counts

SQL counts and validation for reconciliations and month-end closes.

03

Migrating a file

Full dump to JSON as a first step of a migration or a test.

04

Supporting users

A user asks for their data: extracted in a minute, fully traceable.

Install and extract in five minutes

Requirements

- Linux x86-64 (RHEL 8 or RHEL 9)
- Native Adabas client (acj) in the PATH
- Read access to the database
- Product licence or the 30-day trial
- No extra libraries, no configuration

Install and first run

```
# Install the suite
./install.sh --prefix /opt/ns

# Extract a file to CSV
nsadafile -dbid 40 -fnr 120 \
  -output clients.csv

# SQL query with JOIN
nsadafile -dbid 40 -sql "SELECT a AB \
  FROM 120 a JOIN 130 b ON a AB = b AB"
```

The licence is resolved from \$NS_LICENSE_FILE, next to the product data or in /opt/ns/licenses/. Without one: 30-day trial.

FREQUENTLY ASKED QUESTIONS

What customers always ask

Does it block the core?

No. Reads are paged and non-blocking; the file keeps accepting user operations while you extract.

Do I need the DDM?

Not to work by file number. The DDM (JSON) is only needed for logical field names or SQL queries.

What about the licence?

Each product has its own perpetual licence. Without one, the tool runs in trial mode for 30 days.

Can it run in cron?

Yes. It is a dependency-free binary: it drops into scripts, cron jobs and pipelines like any other command.

NEXT STEP

Get your data out without moving a single byte of the core.

Ask for a demo with your own DDM and your own file: you will see the real extraction running against your database.

REQUEST A DEMO →

nacarsoft.com · support@nacarssoft.com · +34 910 038 502



NacarSoft Soluciones Informáticas S.L. · nacarssoft.com · nsadafile

NSAdaFile at a glance

- Extraction to CSV, JSON, XML, TAB and RECORD
- Read-only: zero locks, zero impact
- Filters by ISN, Adabas search and SQL
- Field auditing with -checkfields