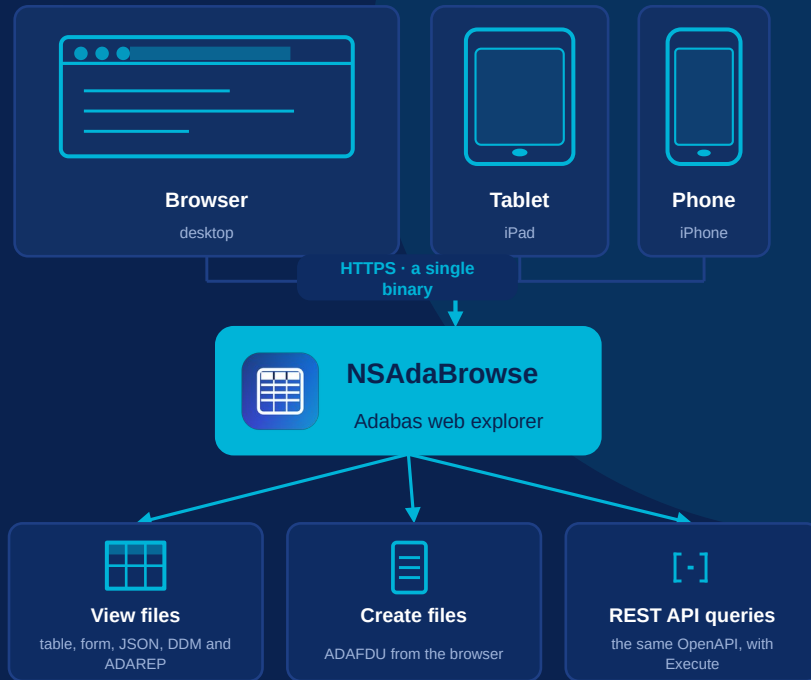


Your Adabas files, in the browser. No client.

NSAdaBrowse is the web explorer for Adabas files: see the records, create new files and run REST API queries, all from the browser and with the company's own access control.

[SEE FEATURES →](#)

[View files](#) · [Create files](#) · [REST API queries](#)



Client connections to view, create and query through the API

browse · DDM · ADAREP · ADAFDU · OpenAPI

Looking at Adabas needs a desktop client

Every query is an input file, an ADABAS session and an export: nobody looks at the data when they need it, and the access stays outside the company's control.

01 Local installation

Every operator needs the ADABAS client installed, with its DDM and its credentials.

02 No access control

The file key travels with the person. There is no user, no role, no per-file permission.

03 No traceability

What was queried, with which criteria and by whom is not recorded anywhere.

04 No API

The data lives in a silo: the web, the ERP and the reports share nothing.

- Per-seat installs

- Keys per person

- Siloed data

The Adabas explorer, as a website

NSAdaBrowse is an Adabas file explorer in a single binary: the administrator declares the databases and their permissions, and users browse the files, see the records and work with them from any browser.

- ✓ Databases by name, with a connection pool and reconnection
- ✓ Files by DDM, with the field definition and its indexes
- ✓ Six views of the same file: table, form, JSON, DDM, ADAREP and API
- ✓ Creation of physical files with ADAFDU, without leaving the browser

6

views per file

0

installations per desk

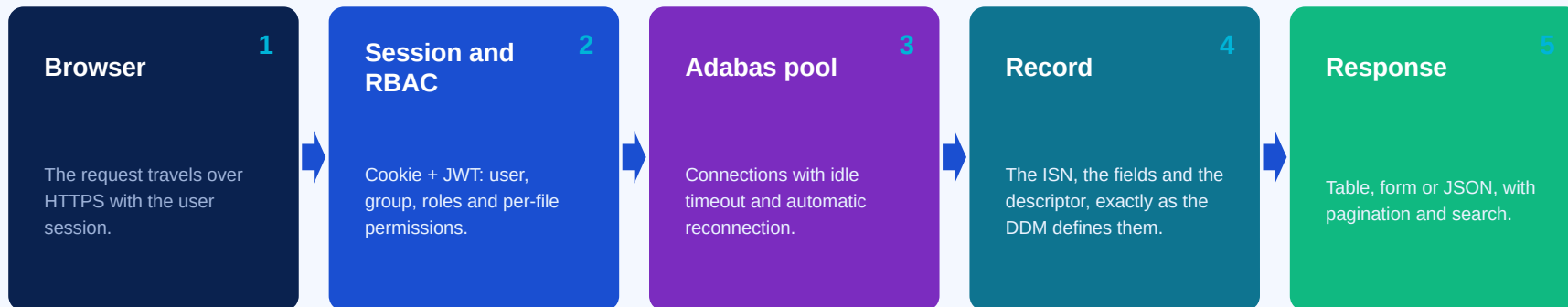
1

binary, no
dependencies

6

languages

How a record reaches the screen



⚠ The local DDM leads: the field definition is cached on disk and does not depend on a live connection.

No local client

No DDM on the desk

The same rules for everyone

VER

Viewing files, in six views

Search by field or by ISN, with operators, sorting by descriptor and pagination.

Table

The file as a grid, with the descriptor and the ISN of every record.

Form

One record per form, with the DDM fields and their group elements.

JSON

The same information as JSON, ready to copy or integrate.

DDM

The full definition: level, type, length, description and indexes.

ADAREP

The real output of the server's ADAREP command.

API

The equivalent REST call, with its response and its cURL.

ISN	1. AA A1111 REG-NUM	1. AB B1111 CHASSIS-NUM	1. AC C1111 PERSONNEL-ID	1. CD D1111 CAR-DETAILS	1. AE E1111 MAKE	1. AF F1111 MODEL	1. AG G1111 COLOR	1. AH H1111 YEAR	1. AI I1111 CLASS	1. AJ J1111 LEASE-PUR	1. AK K1111 DATE-ACQ	1. AL L1111 CAR-MAINTENANCE
1/1	1	3449094	23321	50000200	RENAULT	R5	ROUGE	1980	E	P	19801101	
2/2	2	40114175	1100	50000200	RENAULT	R5	BLANCHE	1980	E	P	19801101	
3/3	3	99917175	1100	50000300	PEUGEOT	305	GRIS	1980	E	P	19801101	
4/4	4	650775	99001	50000400	PEUGEOT	305	BLANCHE	1985	E	P	19851101	
5/5	5	736375	94875	50000500	FIAT	PANDA	ROUGE	1984	E	P	19851201	
6/6	6	124775	23875	50000600	RENAULT	R4	BLEUE	1982	E	P	19821201	
7/7	7	3336794	90875	50000700	RENAULT	R18	GRIS	1982	E	P	19821201	
8/8	8	4851094	83875	50000800	PEUGEOT	205	BLANCHE	1982	E	P	19821201	
9/9	9	5374475	55875	50000900	FIAT	UNO	BLANCHE	1982	E	P	19821201	
10/10	10	788375	33207	50001000	FIAT	UNO	CREME	1980	E	P	19801011	
11/11	11	430075	53256	50001100	BMW	30	GRIS	1993	E	P	19931011	
12/12	12	348676	95256	50001200	RENAULT	R25	GRIS	1986	E	P	19861011	
13/13	13	358676	44256	50001300	CITROEN	CX	BLEUE	1984	E	P	19841011	
14/14	14	8084676	44256	50001400	CITROEN	BX 19	BLANCHE	1984	E	P	19841011	
15/15	15	550775	20256	50001500	RENAULT	R5	BLANCHE	1982	E	P	19821011	
16/16	16	610775	54256	50001600	RENAULT	R5	BLEUE	1982	E	P	19821011	
17/17	17	570475	40046	50001700	FIAT	PANDA	ROUGE	1980	E	P	19801011	
18/18	18	881794	4496	50001800	FORD	SERRA	CORCABANA	1982	E	P	19820331	
19/19	19	440775	5009	50001900	PEUGEOT	305	BLANCHE	1982	E	P	19821111	

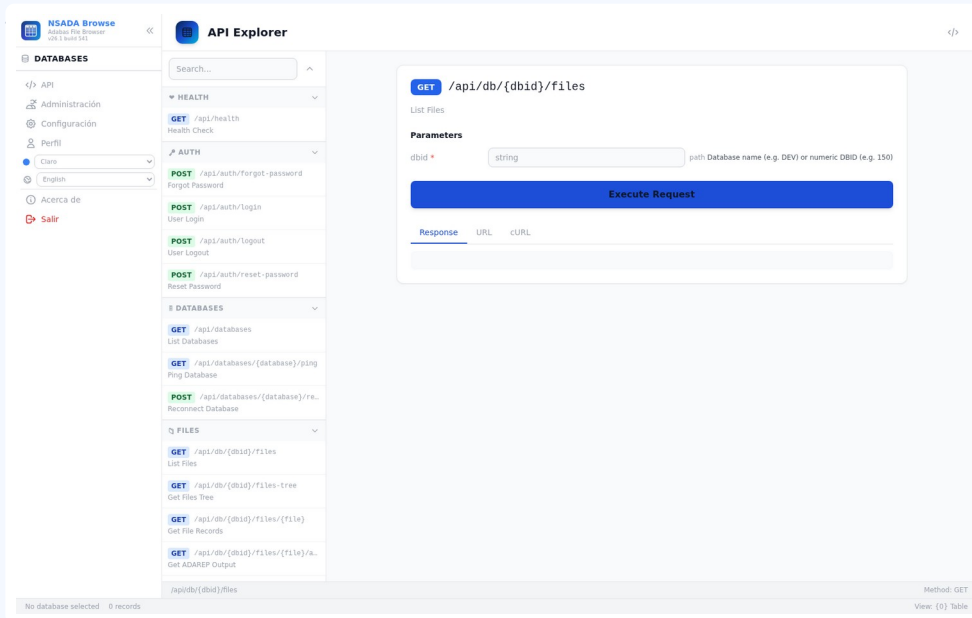
LEVEL	NAME	FIELD NAME	TYPE	LENGTH	DESCRIPTION	RELATED FIELDS
1	AA	REG-NUM	A	15	Registration number	-
1	AB	CHASSIS-NUM	B	4	Chassis number	-
1	AC	PERSONNEL-ID	A	8	Personnel id of holder	-
1	CD	CAR-DETAILS	-	-	Car details group	-
2	AD	MAKE	A	20	Make	-
2	AE	MODEL	A	20	Model	-
2	AF	COLOR	A	10	Colour	-
1	AG	YEAR	U	4	Year of manufacture	-
1	AH	CLASS	A	1	Class	-
1	AJ	LEASE-PUR	A	1	Leased or purchased	-
1	AK	DATE-ACQ	U	8	Date of acquisition	-
1	AL	CAR-MAINTENANCE	-	-	Car maintenance group	-
2	AM	CURR-CODE	A	3	Currency code	-
2	AN	MAINT-COST	P	4	Maintenance cost	-

REST API queries from the browser itself

The API explorer is a full REST console: the same endpoints the tool.

- OpenAPI 3.0 served and explorable from the application itself
- Endpoints grouped by domain: health, auth, databases, files, records and admin
- Parameters, JSON body and Execute, with the response, the URL and the cURL
- Authentication with the session token or with a scoped API key

The very same calls as the web, in a script: cURL ready to copy.



API Explorer: the endpoint, its parameters, Execute Request and the response.

Creating physical files from the browser

ADAFDU

FNR 250

Nombre MI - FICHERO

Descripción Clientes 2026

Nivel/Tipo/Long 1 · A · 20

Campos AA, AB, AC ...

1

FNR and name

The file number and its logical name are chosen in the header.



2

Default values

Level, type, length and the NU, DE, UQ, PE, GR and MU flags.



3

Field table

Each field with its level, name, type, length and description.



4

ADAFDU

NSAdaBrowse generates the physical file and its DDM on the server.



Fichero físico creado

con su DDM, sus campos y sus permisos

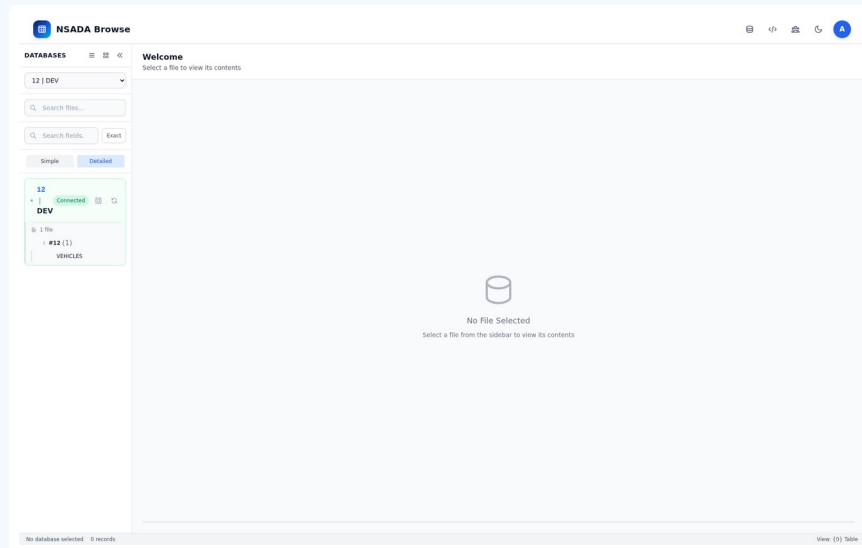
- ✓ Fichero 250 en la base
- ✓ DDM MI-FICHERO generado
- ✓ Visible para quien tenga permiso
- ✓ Listo para escribir y leer

A new file, its DDM and its permissions, without hand-writing a DDM or requesting an input file.

Databases declared in the configuration

- Each database with name, DBID, connection string and configuration file
- Several databases in the same instance, each with its own pool
- Live state: connected, stopped for inactivity or in error
- Reconnect, stop and ping from the panel itself

Each file's DDM is stored on disk: the list does not depend on the connection.



The database tree with its state and the files of each one.

PERMISOS

Permissions per file, not per person

- Users → groups → roles → permissions per file
- Read, write and delete per file, with a wildcard
- File listings and records are filtered by permissions
- API keys with scopes per integration, with revocation and audit

Access control

- RBAC by roles and groups
- Permission per file (read/write/delete)
- Listings filtered by permissions
- Access by token or API key

API keys

- Scope per integration
- Configurable expiry
- Revoke and reactivate
- Usage audit

Session

- HS256 JWT in a cookie
- Passwords with bcrypt
- Recovery by email with a code
- Lockout after failed attempts

config.yaml 0600 · bcrypt · JWT HS256 · per-operation audit · 30-day trial

What you can do with a file

Read

Records by pagination, with search by field, by ISN and by range.

Update

Update a record by ISN, with its audit trail.

Delete

Delete by ISN, one or many, with a record.

Empty

Empty a whole file, with the matching permission.

Create

A new physical file with ADAFDU, with its fields and its DDM.

Drop

Remove the physical file and its definition from the DDM.

Todo lo que se hace con un dato queda registrado: acción, usuario y resultado.

GETTING STARTED

From zero to queryable files, in three steps

1

Create the environment

`nsadabrowse init /opt/nsadabrowse`: it creates `config.yaml`, the SQLite user database, the logs and the admin user.

2

Declare the databases

In `config.yaml`: name, DBID, connection string and the DDM file of each database.

3

Register the users

Administration → Users and groups: each user only sees the files their role allows.

REQUIREMENTS

- Linux (RHEL 8 / RHEL 9)
- Native Adabas client (CGO)
- A single binary, no dependencies
- Reverse proxy with HTTPS in production
- Web panel on port 11010

```
nsadabrowse init /opt/nsadabrowse → nsadabrowse start /opt/nsadabrowse → http://localhost:11010
```

FREQUENTLY ASKED

The questions we hear most

Is the Adabas client needed?

Only on the server, and optionally: the DDM is stored on disk, so the file list and its definition are shown without a live connection.

Can the data be queried through an API?

Yes: the REST API with OpenAPI 3.0 is published, explorable from the panel itself and consumable with a token or scoped API keys.

Can new files be created?

Yes: from the browser, with the ADAFDU form: FNR, name, default values and the field table. The DDM is generated too.

Who sees which file?

The permission lives on the role, per file, with read, write and delete. Listings and records are filtered with those permissions.

Is what they do recorded?

Yes: every operation is logged with its user and its result, and the API keys have their own usage log.

Does the user need the DDM on their desk?

No: the DDM travels with the file definition and the browser shows it. The operator installs nothing.

Your company Adabas, open in the browser

Try NSAdaBrowse with your databases: we set it up with your files
and watch it run.

Guided demo

Try it in your environment

Let's talk about your case

nacarsoft.com · support@nacarsoft.com

