NETDIRECTOR
INTEGRATE ONCE

PRICED PER PAGE · NO SUBSCRIPTION

DOCUMENT DATA EXTRACTION

AI DocQuery

Your recorded documents already hold the data your systems need — it is just locked inside a PDF. AI DocQuery reads them and hands back structured values you can load into your own database, or query directly in AI CaseQuery at the document level.

91 fields ready on day one · **Switch any off, add your own** · **Runs inside your network** · **Every value traced to its page**

WHAT BECOMES ASKABLE ONCE DOCUMENTS ARE DATA

Questions your documents can already answer

The answers are in the file room today. They are simply not in a form anybody can query — so nobody asks, and the work gets done by hand, one file at a time.

> *"Which files have a maturity date that does not match what our system says?"*

> *"Show me every loan where the recorded assignment chain has a gap."*

> *"Which mortgages carry a prepayment premium, and how much?"*

> *"Find the latest recorded instrument on this file and tell me what changed."*

> *"Which properties are secured by more than one parcel?"*

Pages Bought Up Front

You buy a block of pages and we send you a key. No seat licences, no minimum, no subscription. Buy more at any time — the pages you have left are never lost.

Your Fields, Your Call

91 fields are active from the start. Switch off any you do not want, and add fields of your own with a short description.

Every Value Verified

Nothing is returned unless it can be traced back to the words on the page. A value that cannot be is flagged for review, never presented as fact.

Installed On Your Server

It runs where you already run things, reachable from your own network. Your developers call it as documents arrive, or send a backlog overnight.

HOW IT WORKS

Send a document, get back data

There is nothing to model first and nothing to tell us about your documents. Send a PDF — a single instrument or a packaged file holding several — and the values come back structured.

1 **Send it — no sorting required**

You do not tell us what the document is. A packaged PDF holding a mortgage, a rider and an assignment comes back as three documents, each with its own values and page range.

2 **Read, including the difficult scans**

Purpose-built document OCR, not a general chatbot reading pictures. Every page carries a confidence score, so poor scans, stamps and handwriting are flagged rather than guessed at.

3 **Extracted, then checked against the page**

Each value must be quoted from the document before it is accepted. If it cannot be found in the text, it goes to a review list instead of into your data.

4 **Delivered in a form that joins**

Values arrive twice — as the document wrote them, and cleaned. `$ 75,803.00` and `$75,803.00` become one figure, so three documents about one loan agree.

What comes back

91 fields across ten groups, all active from the start. Anything the document does not state is simply left out — never guessed.

- **Recording** · number, date, county, fees
- **Parties** · borrowers, lender, MERS, trustee
- **Loan terms** · principal, rate, maturity, FHA
- **Property** · address, parcel, occupancy
- **Assignment** · assignor, assignee, chain
- **Modification** · new balance, new maturity
- **Court** · case, judgment, sale, deficiency
- **Bankruptcy** · chapter, stay, claim, discharge
- **Title** · vesting, priority, exceptions
- **Corporate** · entity, successor, filing

Several borrowers? Several parcels?

Each one comes back as its own row, in the order the document lists them — not squeezed into a single field for somebody to unpick later.

Two ways to hold it

Field level only — recommended

We keep the values and nothing else. No document text is retained anywhere: just the field, the page it came from, and whether it verified. It is the shorter conversation with a servicer — **"we hold these values, not your documents."** The trade is that a field you did not ask for later means running the PDF again.

Field level plus full text

The same values, plus every page as searchable text — so you can ask which documents mention an environmental indemnity, and add new fields later without re-reading a thing. The trade is that you are now holding a searchable copy of the document.

One or the other

Chosen per client, never both. We will tell you plainly which one you are on.

WHERE IT RUNS

On your server, inside your network

AI DocQuery installs on a machine of yours and answers on your own network. There is no vendor cloud, no account to open with us, and no document of yours ever passes through us. It is three files and needs nothing installed but Node.js.

Your own accounts, your own bill

You bring your own OCR and AI provider keys, so the reading and the extraction are billed to you, at your rates, under your agreements with them. We are not in the middle of that relationship and never see your documents or your keys.

The administration page stays on the machine

The endpoint is reachable from your network so your software can call it. The page where pages are topped up is not — it answers only on the server itself, so opening the service to your developers never opens its console to your network.

Your file numbers come back untouched

Send your own file number and document id with each document and both come back on every row, exactly as you sent them. Nothing to reconcile afterwards and no matching on file names, which stops being reliable the moment two documents share one.

Send the same document twice, safely

Re-sending a document replaces what was there rather than duplicating it — so a corrected extraction can simply be run again over the top of a bad one.

Into your own systems

Your developers call the endpoint as documents arrive and get structured data straight back — ready to write into your tables, on your schedule, under your control.

Or as a script your DBA reads

No integration work at all: it writes a **.sql** file for SQL Server, Oracle, PostgreSQL or MySQL. It only ever writes to its own two tables — never to yours — so a person reviews it and decides.

Or straight into AI CaseQuery

Load it beside your case data and ask questions across both — document values and case records answered in a single question.

Nothing is presented as fact unless the document says so

Every value is checked against the page it was read from before it is accepted. Anything that cannot be traced back is flagged for a person, not quietly added to your data — which is what makes the output safe to load into a system of record.

- → Values verified against the document text
- → Per-page confidence scores identify poor scans
- → Nothing inferred, estimated or calculated
- → Missing is reported as missing, never invented
- → Every page read is counted and reconcilable