WHITE PAPER

What AI actually needs from your data

Most AI pilots in associations and nonprofits do not stall on the AI. They stall on a member who exists three times, a status field nobody uses the same way, and a renewal history that was overwritten every year. This paper explains what AI does with your records, what it needs from them, and what to fix first.

01 Why pilots stall

02 One record, in practice

03 Fields AI can read

04 Definitions and history

05 Retrieval, privacy and consent

06 What to fix, in what order

07 The AI data-readiness grid

For
Anyone planning AI, or the data work behind it

Time to read
About 25 minutes

Use it
Before you fund an AI pilot

A NOTE BEFORE YOU START

The demo always works. The demo was built on clean data.

Every AI demonstration you have seen this year ran on a tidy set of records: one row per person, every field filled, every status spelled the same way. Then the pilot starts on your own data, and the answers get vague, the predictions get odd, and the project quietly loses its sponsor.

When that happens, the usual conclusion is that the tool was oversold. Sometimes it was. More often the tool did exactly what it was built to do. It read what was there, faithfully, and what was there did not agree with itself.

I want to be specific about what that means, because "clean your data first" has become advice that everyone repeats and nobody can act on. Clean for what? To what standard? Which part first, when the whole database has years of history in it?

This paper is my answer. It describes what AI does with an association's or nonprofit's records, in plain English: how it reads, how it predicts, how it finds the right record to answer from. Each of those places a specific demand on your data. Once you can see the demand, you can see the fix, and you can see that the fix is smaller than "clean everything".

The good news is that none of it is new. One record per person, fields used the same way, definitions the board agrees with, history you can trust. It is the work that makes your reporting honest, and AI is simply the most demanding reader you will ever give it to.

Sunil Dua
Executive Director and Principal Consultant, Kloudserv

The short version

- AI reads, predicts and retrieves. Each one fails on a different data problem.
- "One record" is a set of matching rules you write down, not a feeling.
- Inconsistent picklists and free text defeat counting and prediction alike.
- A model learns your definitions. If they changed, it learned two things.
- Permissions and consent belong in the data, not in the instructions to the AI.
- Fix what your first use needs, at the point of entry, then clean backwards.

Who it is for

Anyone deciding whether to fund an AI pilot, and the people who will be asked to make the data ready for it.

Worth remembering. AI does not fix inconsistent data. It reads it quickly, at scale, and writes confident sentences about it.

WHY PILOTS STALL

01 Three things AI does with your records, and how each one breaks

Strip away the product names and almost every practical use of AI in a membership or fundraising organization does one of three things with your data: it writes about records, predicts from history, or finds the right records and answers from them.

IT READS AND WRITES

Summaries and drafts

A renewal check-in drafted from a member's history. A board report commentary written from this quarter's figures. A summary of a donor's last five years before a visit.

Breaks when the record is split, so the summary describes a third of the person, or when the fields disagree, so the draft repeats the contradiction in good prose.

IT LEARNS AND PREDICTS

Scores and lists

Which members are likely to lapse. Which donors are likely to give again. Which event registrants are likely not to show.

Breaks when the history is thin, overwritten or recorded under shifting definitions. A model can only learn the pattern your past records actually contain.

IT FINDS AND ANSWERS

Assistants

A member asks the portal how many CE credits she still needs. A staff member asks which Ohio members have not renewed.

Breaks when it cannot find the one right record, finds an out-of-date one, or finds one the person asking should not see.

The pattern behind a stalled pilot

The first week looks promising, because the team tests it on members they know well, whose records are usually the good ones. Then it meets the rest of the database. A long-standing member is flagged as new, because her current record was created when she changed employers. A donor's summary leaves out his largest gift, because it sits on a household record the AI was not pointed at. The portal assistant tells someone their membership has lapsed when it has not, because the status field and the paid-through date disagree, and it read the wrong one.

Each mistake is small and explainable. Together they teach staff to check every line, at which point it saves nobody time, and the pilot ends without anyone formally ending it. Yet each mistake points at a specific, fixable property of the data.

A TEST YOU CAN RUN THIS WEEK

Pick ten members or donors at random. For each, ask a staff member to tell you, from the system alone, their status, when they last renewed or gave, and what they attended last year. Note every time they check a second place or ask a colleague. That is your AI's experience, minus the colleague.

ONE RECORD, IN PRACTICE

02 "One record per person" is a set of rules you write down

Everyone agrees they want one record for every member and donor. Fewer organizations have decided what makes two records the same person. That decision is called identity resolution, and it is the first thing AI depends on.

Take Maria Lopez at the Lakeview Association. Over eight years she registered for events with her work email, joined as a member through her employer, Harbor Clinic, and made a personal gift with her home email. The database now holds "Maria Lopez", "M. Lopez" and "Maria Lopez-Reyes", and her employer exists as "Harbor Clinic" and "Harbour Clinic Inc."

Ask an AI whether Maria is at risk of lapsing and it will look at one of those records. Whichever one it picks, it sees a fraction of her relationship with you. The honest answer to the question lives across all three.

Fixing it is not a one-off merge. It is a set of matching rules, agreed and written down, that decide what happens every time a new record arrives.

SIGNAL	HOW MUCH TO TRUST IT
Same personal email	**Strong.** Usually the same person. Merge, with a check.
Same member number	**Strong,** if the number is never reused.
Name plus organization plus postcode	**Medium.** Suggest a match; a person confirms.
Same work email	**Medium.** Shared inboxes like info@ belong to many people.
Same name only	**Weak.** Never merge on this alone.

An example set of rules. Yours will differ; the point is that they are written down.

Three decisions that make a record "one"

What counts as a match. Strong signals can merge automatically; medium ones go to a person; weak ones never merge. This is the difference between AI that suggests "Maria Lopez and M. Lopez, likely the same person" for someone to approve, and a script that merges two different people called Maria Lopez.

Which value survives. When two records merge and their addresses differ, which wins? The most recent? The one the member entered herself? The one from the finance system? These are called survivorship rules, and without them every merge is a coin toss that quietly loses information.

Who the record belongs to. Associations with organizational members face a question individual donors do not: when Maria leaves Harbor Clinic, does her membership leave with her or stay with the clinic? The answer is a policy decision, not a technical one, and the data model has to reflect it. If it does not, AI will read a job change as a lapse. Nonprofits meet the same question with households: is a gift from the Lopez household Maria's gift, her partner's, or both, for the purposes of a thank-you, a receipt and a renewal ask?

The test: can someone on your team explain, without opening the system, what happens when a person who already exists registers for an event with a new email address? If the answer is "it depends who enters it", the record is not yet one.

FIELDS AI CAN READ

03 Why a messy picklist defeats both the summary and the prediction

People tolerate inconsistency because they carry the context. A membership coordinator knows that "Grace", "Grace period" and "GP" mean the same thing, and that "Expired" was what the old system called a lapse. AI has only what is written. To it, these are four different states.

What it does to a summary

Ask for a count of members in their grace period and the answer will be right for whichever spelling the question happened to match, and silently wrong for the others. A commentary on why renewals fell will describe the field as it finds it, including the category that exists only because someone typed it differently in 2021.

What it does to a prediction

A model that learns who lapses treats each value as a separate signal. Split one meaning across four values and each one has a quarter of the examples, too few to learn from. Worse, if one spelling was used by one team or one era, the model learns the team or the era, not the member.

Free text is a special case

A free "reason for leaving" field will hold "cost", "$$", "budget cuts", "see email from Jan" and nothing at all. Nobody can count that, and no model can learn from it.

Here AI is genuinely useful on the other side of the problem. Proposing a category for each of a few thousand free-text entries is something language models do well, with a person checking a sample. Used that way, AI helps you build the clean field. It just cannot stand in for one.

Blank is not "no"

An empty "attended conference" field might mean the member did not attend, or that nobody recorded it. A model cannot tell, and concludes nobody came before you started tracking.

WHAT YOU WILL FIND	WHAT AI DOES WITH IT	THE FIX
Same meaning, several values	Counts and learns each value separately; summaries are partly right	One restricted picklist; map old values once
Free text where a choice belongs	Cannot count or predict from it	Add a picklist going forward; use AI to sort the backlog, with review
Blanks with two meanings	Reads every blank as "no"	Record "not tracked" explicitly, or note the date tracking began
Two fields that disagree	Picks one, often the wrong one, with confidence	Decide which field is the truth; retire or derive the other

The quiet lesson: a restricted picklist on the form does more than any clean-up project, because it stops the problem returning next month.

DEFINITIONS, HISTORY AND OWNERS

04 A model learns your definitions, including the ones you changed

Agreed definitions

What is a member? Someone with a paid invoice, someone inside a grace period, anyone with portal access? What is an active donor: a gift in the last twelve months, the last eighteen, or this fiscal year? Most organizations have two or three answers in use at once.

For a summary, that produces a figure that does not match the one the board saw last quarter. For a prediction it is more serious. A model that learns "who lapses" learns whatever "lapsed" meant in your records. If you moved from a 30-day to a 90-day grace period three years ago, your history holds two populations under one label, and the model learns a blend that describes neither.

History you can trust

Prediction needs a record of what happened, and when. Many systems keep only the current state. The membership status is updated in place each year, the paid-through date is overwritten at each renewal, and the history of who lapsed and came back is simply gone. You cannot learn renewal risk from a database that only remembers today.

Two more history problems make a pilot look better in testing than it will be in use.

Dates that describe the data entry, not the event. Records brought across from an older system often carry the date they were loaded, not the date the member joined. A model reading them concludes that half your membership joined on the same Tuesday.

Fields filled in after the outcome. A "reason for leaving" is only ever filled in for people who left. Include it in a lapse model and it looks brilliant in testing, because it is reading the answer. In real use the field is empty for everyone it is asked about. This is usually called leakage.

Known owners

Every one of these fixes needs someone with the authority to decide: which status values exist, what counts as a lapse, which record survives a merge. Without an owner, each question goes to whoever notices, and gets decided differently each time. That is how the inconsistency got there in the first place. Ownership does not need a data team. It needs a name against each area, usually the person who already runs it: membership owns status and renewals, development owns gifts and donor preferences, events owns attendance. The decision is theirs, written down, and AI suggestions in their area come to them.

Start keeping history now. Even if your past is overwritten, switching on field history for status, paid-through dates and gift dates today means that in one renewal cycle you will have history you can trust. It is the cheapest AI preparation there is.

RETRIEVAL, PRIVACY AND CONSENT

05 How an assistant answers from your records, and who decides what it can see

When Maria signs in to the Lakeview portal and asks how many CE credits she still needs, the assistant does not "know" the answer. It does three things in sequence.

It finds. The system looks up the records that might hold the answer: Maria's credit history, the credit requirement for her certification, the upcoming events that award credits. Only Maria's records, because she is the one signed in.

It hands over. Those records, and nothing else, go to the language model with an instruction: answer from this, and say so if the answer is not here.

It writes. The model turns what it was given into a sentence: "You have 12 of the 18 credits you need by December 31." It shows where the answer came from.

This pattern is usually called retrieval. It is what separates an assistant that answers from your records from a chatbot that answers from whatever it absorbed during training.

What retrieval asks of your data

The model can only be as right as what was found. If Maria has three records, the finder picks one. If the credit requirement lives in a PDF that was updated last year but the old version is still in the folder, it may find the old one. The answer will be fluent, sourced and wrong.

Permissions belong in the finding step

The most important design choice in any assistant is where access is controlled. Telling the model "do not reveal other members' information" is an instruction, and instructions can be argued with by a determined user. Making sure the finder can only ever return Maria's records is a control. If another member's record never reaches the model, the model cannot disclose it.

For a staff assistant the same rule applies: it should find only what that staff member could already see in the system. Your existing permissions do the work, which is one more reason to get them right before AI arrives.

Privacy and consent: four questions to settle per use

What may it see? Name the fields, not the database. A renewal-risk list needs membership, events and email activity. It does not need gifts or case notes, and notes fields are where sensitive detail about health, family and money tends to collect.

Are preferences in the data? Do-not-contact, do-not-solicit and communication preferences must be fields the AI reads, so a drafted check-in never goes to someone who asked to be left alone.

Is this use within what people agreed to? Data given to register for an event was given for that purpose. Read your privacy notice and ask whether a member would be surprised by the new use. If yes, say so to them before you start.

Where does the data go? Ask any vendor whether your records are used to train their models, where they are processed, and how long they are kept. Check the answers against the laws that apply to your members and donors, in the US and in Canada.

WHAT TO FIX, IN WHAT ORDER

06 Do not clean everything. Clean what your first use reads.

"Clean the data first" becomes a project that never finishes. The better sequence starts from one use, works out which records and fields it reads, and fixes those. This is the order we follow.

01 **Choose one use, tied to a number the leadership already watches.** For most associations that is renewals; for most nonprofits, repeat gifts. The use decides the scope of the clean-up.

02 **Write the definitions that use depends on.** What a lapse is, what an active donor is, what counts as attending. One paragraph each, agreed by the owner and, where the board sees the figure, by the board.

03 **Settle identity for the people it touches.** Matching rules, survivorship rules, and a merge of the duplicates among current members and recent lapses. Not the whole archive.

04 **Fix the fields at the point of entry.** Restrict the picklists, add the missing choice, make the required field required. Do this before cleaning the backlog, or the backlog refills while you clean it.

05 **Then clean backwards.** Map old values to new ones, sort the free text (AI can help here, with review), and mark what was never tracked rather than leaving it blank.

06 **Start keeping history.** Turn on field history for the fields the use reads.

07 **Decide access and consent.** Which fields it may see, who may see its output, which preferences it must honor, and what you will tell members and donors.

08 **Name the owner and the reviewer.** Often the same person, with time set aside each week.

An honest point about prediction. A smaller organization may not have enough lapses in its history for a model to learn anything reliable. A weekly list built from rules a person can read is often the better first step, and it starts the history a model will need later.

Where AI helps with the fixing. Suggesting likely duplicates, proposing categories for free text, finding empty fields that could be filled from a source you trust. Each suggestion goes to the owner to approve. Nothing merges, deletes or overwrites on its own.

THE PRACTICAL TOOL

The AI data-readiness grid

Score each row 0, 1 or 2 for the area your first AI use will read, not for the whole database. Do it with the person who owns that area. Be strict: when in doubt, take the lower score.

AREA	0 · NOT YET	1 · PARTLY	2 · READY	SCORE
One record	Duplicates are common and nobody is sure how many	Duplicates are merged now and then, by whoever notices	Matching and survivorship rules are written down and applied to new records	☐
Consistent fields	Key fields are free text or carry several spellings of one meaning	Picklists exist but old values linger or teams use them differently	Restricted picklists, one definition per field, old values mapped	☐
Agreed definitions	Member, lapse or active donor mean different things in different reports	Definitions exist but are not written down or not used everywhere	Written, agreed by the owner, used by every report the board sees	☐
History you can trust	Status and key dates are overwritten; past states are lost	Some history kept, with gaps or loaded dates you cannot rely on	Field history on for key fields; past changes can be reconstructed	☐
Known owners	Nobody can say who decides what a field means	Owners are understood informally	A named owner per area, with time to review AI suggestions	☐
Access and consent	Permissions are broad; preferences live in notes or people's heads	Permissions set but not reviewed; some preferences are fields	Permissions reviewed; preferences are fields; the use is within your privacy notice	☐
			Total, out of 12	☐

10 TO 12
Ready for a small, measured pilot. Keep a person reviewing every output.

6 TO 9
Fix the rows scored 0 for the first use, then pilot. Often weeks rather than months.

0 TO 5
The foundation comes first. AI here would repeat the mess, faster.

Which rows matter most for which use

USE	ONE RECORD	FIELDS	DEFINITIONS	HISTORY	OWNERS	ACCESS
Renewal or repeat-gift risk list	●	●	●	●	○	○
Member and donor assistant	●	●	○	○	○	●
Staff assistant	○	●	●	○	○	●
Board report drafts	○	○	●	●	●	○
Data quality suggestions	●	○	○	○	●	○

● must be ready before the use starts ○ matters, can be at "partly" for a first pilot

WHAT TO DO NEXT

Two steps for the next month

01 Name your first use, and score the grid for it

One hour with the person who owns that area. Pick the use closest to a number the board already watches, then score the six rows honestly for the records it would read. Run the ten-record test from page 3 alongside it. Between them you will know whether you are weeks or months from a pilot.

02 Fix the point of entry before anything else

Whatever scored lowest, start where new data comes in: the form, the picklist, the matching rule, the field history setting. It costs little, it stops the problem growing, and it is useful whether or not the AI pilot ever happens. Clean data makes your board reporting honest long before it makes AI work.

None of this is glamorous, and that is rather the point. The organizations that will get real value from AI over the next few years are not the ones that bought first. They are the ones whose records can be read by a stranger, and trusted. AI is that stranger.

IF YOU WOULD LIKE A SECOND PAIR OF EYES

The readiness scorecard on our website takes a few minutes and covers data alongside people, systems and process. If you want the whole picture, the Digital Readiness Blueprint looks at your records with your team over four to six weeks and tells you plainly what it would take to get to one record, and which AI use is worth starting with.