

The Operator's Manual

Ark of Noahledge · offline knowledge archive · Revision 1.5 · 2026-09-27

IF YOU ARE HOLDING A PRINTED REVISION 1.2, TWO THINGS IN IT ARE WRONG, AND ONE OF THEM WILL COST YOU HALF THE SEARCH.

Chapter 8 has no step that installs the packages. Copying the archive does not install them, and revision 1.2's chapter 8 goes straight from *install Python* to *rebuild the two files*. Follow it and you get a working archive with **no hybrid search**, and every check in that chapter still passes. It is step 3 here. Nothing else in 1.2 will tell you.

The library file moved. Chapter 3's fallback command and chapter 8 both named it as `<ARCHIVE>/07-corpora-supplemental/library.xml`. **It is now** `<ARCHIVE>/library.xml`, at the top of the archive, covering every shelf rather than one. Corrected in the text 2026-09-12 but never printed, so a printed 1.2 sends you to a file that is not there. Chapter 6 symptom A carries the second symptom of the same fault: the server starting normally while every document link fails.

What changed since Revision 1.4 (2026-09-26). The assistant changed how it answers, so chapters 1, 3, 4 and 9 changed with it. **No step and no label changed.**

- **Chapter 1: it now answers from its own knowledge and checks the answer against the archive.** Until 1.4 it answered *using only the passages*. The answer still shows what the archive supports first, with a [1] on every statement, and the model's own full answer after it in a marked block. - **Chapter 3: a page listing everything the node serves**, at `http://localhost:8090/home/`, and a *Services* menu on the search page. - **Chapter 4: answers are longer.** About a minute for the answer, a few more for the second model's check. What the two parts of an answer are, and the one line in the second part that tells you to stop and read the passages. - **Chapter 9, item 1:** it finds what sources say *and adds what its models learned*, and marks which is which. - **Chapter 5 did not change.** Read the source passage before acting on any number. Longer answers make it matter more, not less.

What changed since Revision 1.3 (2026-09-17). Three figures, all because the index grew from about 7 million passages to 39 million. No step changed.

- **Chapter 3's** `status example` now reads `dense: pq over 39,073,563 vectors`. The number is how many passages the meaning search covers. What tells you hybrid is working is the words `dense: pq over`, not the number. - **Chapter 4 no longer says every hybrid search after the first is fast.** After the first, each one takes about ten seconds. That is normal, not a fault. - `index-provenance.py --rebuild` **takes about ten minutes, not one** (chapter 6 symptom **D**, chapter 8 step 4). With a printed 1.3 in hand you would think it had hung. It has not.

What changed since Revision 1.2 (2026-09-08).

- **Chapter 8 gained step 3**, which installs the packages, and step 6 now asks for `--mode hybrid` so that the final check is able to fail. Without it the chapter ended by declaring a half-recovered node complete. - **Chapter 3 no longer says hybrid is decided at startup.** It is not. The half that matches meaning loads on first use and can fail after everything looks fine, which is what happened on 2026-09-16 for three hours. There is a table of what `ark.py status` actually reports. - **Chapter 4's label table was wrong and is now complete.** It listed two of the five labels, described `uncovered` as something it is not, and had no entry for `model only - no citation`, the only one on the list that is a fault. The first column now quotes the screen word for word. - **Chapter 4 says** `hybrid` **is already selected** when the page opens.

What changed since Revision 1.1 (2026-09-07). Chapter 3 is rewritten around a single command that starts everything, and **starting the two models is now your job rather than the builder's** — Revision 1.1 told you it was not, which made two steps of the annual drill impossible to pass from these pages. Chapter 6 symptom **E** changed with it, and chapter 8 now says what a rebuilt machine needs before each model will run.

If you are holding a Revision 1.0 printing (2026-09-06), it is also missing what 1.1 added: *Finding a place by name* in chapter 3a, and the reason a place may be missing is the map's labelling and not the zoom; a second quarterly pass in chapter 7, because the one in 1.0 reads only the cold drive and the working drive needs its own; and the line below for the `python` that can read maps.

Before anything: fill this in

This manual cannot know where your archive lives. Write it here, once, in pen.

	YOUR ANSWER
Where the archive is on this machine	
The name written on the cold drive	
Who built this	
Who to call	
Date of the last drill	
The python that can read maps (ch. 3a)	
The folder llama-server was extracted into (ch. 3)	

Everywhere below, <ARCHIVE> means the first line. It is the folder that contains 10-index , 07-corpora-supplemental and 13-ark-node .

Commands. Type them exactly. On Ubuntu use the Terminal. On Windows use **Git Bash**, not the Command Prompt, because the commands are written for it.

Three commands in this manual are not python , and they are the ones where the difference bites: chapter 3 starts kiwix-serve , chapter 3a starts pmtiles serve , and chapter 7 begins with bash . This page used to claim everything you need is python , which is what tells a Windows reader they can ignore the Git Bash line above. Chapter 7 is exactly where that would have failed, and chapter 7 is not skippable.

0. If you are reading this

Something has happened, or someone is testing whether these pages work. Either way this page is the right place.

You cannot break the archive by reading it. Nothing in chapters 1 to 5 changes a single file. If you are unsure, do those and stop.

What can wait. Everything in chapters 7 and 8. They are maintenance and repair, not emergencies. If the machine turns on and search works, nothing is urgent.

What cannot wait. If a drive is making a noise it did not make before, unplug it. A failing drive gets worse while it runs.

1. What this is, in one page

A computer with no internet connection holding a copy of a large amount of reference material: medicine, repair, agriculture, engineering, maps, textbooks. About two terabytes.

On top of that sits a search that finds passages, and, when its models are running, an assistant that answers questions from its own knowledge, **checks the answer against those passages**, and shows you which parts the archive supports and where each one came from.

Three things follow from that, and they are the whole reason this manual exists.

1. **It can only check what the archive contains.** When the archive does not cover a question, the part of the answer that cites passages says `NOT IN ARCHIVE`, and anything the model adds from what it learned is marked as not from the archive. That message is the system working, not failing.
 2. **The answer is a summary. The passage is the source.** Any number you are about to act on gets read in the passage, not in the answer.
 3. **It works in stages.** Search works with nothing but the archive. The assistant needs more, and chapter 3 shows you how to start it. If the assistant is unavailable, search still works, and the screen will say so.
-
-

2. The drives

There are at least two copies of everything: the one in the machine, and one or more **cold drives** that stay unplugged.

A cold drive that is left plugged in is not a backup. It shares whatever happens to the machine, which is the thing it exists to survive. Plug it in for the job, then unplug it.

To connect one: plug it into a USB port. Wait until the machine shows it. Note the letter or name it appears under; you will need it.

To disconnect one: always eject it properly first. On Windows, the tray icon for safely removing hardware. On Ubuntu, the eject symbol beside the drive. Then unplug. **Do not skip this.** Pulling a drive mid-write is the most common way an archive gets damaged by the person trying to protect it.

3. Turning it on and searching

One command starts everything.

```
python <ARCHIVE>/bin/ark.py up
```

It starts each part that is not already running and prints a table saying what came up, what did not, and why. **Running it twice is safe** — it asks each part whether it is already answering before it starts anything.

Two more are worth knowing:

```
python <ARCHIVE>/bin/ark.py status
python <ARCHIVE>/bin/ark.py down
```

`status` says what is running right now and starts nothing. `down` stops what the launcher started.

What it starts, and what you lose without each part:

PART	WITHOUT IT
archive	the source documents behind every citation. Links stop opening
node	the search page itself. Nothing works
tiles	the maps, chapter 3a. Everything else is unaffected
primary	the answer pane. Search and passages still work
crosscheck	the second opinion on answers carrying a figure

The first three are search. The last two are the assistant, and they are the two that a given machine may not be able to run.

Now open a web browser on this machine and go to:

```
http://localhost:8090/
```

You should see a search box. **That is the system running.** If the browser shows nothing, chapter 6, symptom **B**.

Everything else the node serves is listed on one page, with whether each part is running:

```
http://localhost:8090/home/
```

The search page, the library of every book on the drive, the map, the two models' own chat pages, and the node's health. The *Services* menu at the top right of the search page opens the same list. **The two chat pages talk to a model directly: nothing is looked up and nothing is cited**, and the page says so. They open only on this machine.

If a part says **down** and names something missing

The launcher tells you what to do on the same line, usually a file to unpack from **09-software**. Do what it says, then run **up** again.

If it refuses to start a model, saying the machine cannot hold it, that is not a fault. It is the launcher declining to fail slowly rather than letting the load run for a minute and then collapse. Everything else still starts.

If the launcher will not run at all

It is a convenience and it is not the system. Every command it would run can be printed instead, and then typed by hand:

```
python <ARCHIVE>/bin/ark.py commands
```

That starts nothing. If even that will not run, these two are the floor. Each goes in its own terminal window, and both windows stay open.

Window 1, the archive server, which serves the documents behind each citation:

```
<ARCHIVE>/09-software/kiwix-tools/bin/kiwix-serve --port 8080 --library <ARCHIVE>/li
```



Window 2, the search page itself:

```
python <ARCHIVE>/13-ark-node/ark-api/serve.py
```

The second ends with a line like `serving on http://0.0.0.0:8090/`. Closing either window stops that part. If `kiwix-serve` says the library file does not exist, go to chapter 6, symptom **A**. That pair gives you search, passages and every citation, with no maps and no assistant.

On Windows each part gets its own window. Closing one stops that part, and five windows closed at once looks exactly like five things failing at once. If `status` says everything is down and you did not run `down`, the windows were closed. Run `up` again.

Whether hybrid search is available is NOT decided at startup, and the launcher cannot tell you. The half that matches meaning loads the first time something asks for it, so it can still fail after everything looks fine. On 2026-09-16 this node ran for three hours with the launcher reporting a healthy setup and hybrid search quietly unavailable.

`up` tells you which Python it chose. That is all it can know at that moment. **`status` is what reports the answer**, on the `node` line:

WHAT STATUS SAYS ON THE NODE LINE	WHAT IT MEANS
dense: pq over 39,073,563 vectors	hybrid is working, whatever the number
dense not loaded yet - it is lazy, it loads on the first query	normal, not a fault. Nothing has asked for it yet. Run one search and look again
DENSE FAILED TO LOAD - keyword only	hybrid is not available. The reason is printed under the table
no keeper venv at C:\ark-env	the packages were never installed. Chapter 8 step 3 installs them

Both a working hybrid and keyword-only are working configurations; chapter 4 explains the difference.

3a. The maps (optional)

The archive holds **843 GB of maps** — a whole-planet street map and a whole-planet terrain model. They are about 38% of everything on the drive and they need one more part running.

`ark.py up` starts it as `tiles`, along with everything else, so if you followed chapter 3 it is already running and you can skip to the browser below. `status` tells you whether it is up.

By hand, if you are not using the launcher, it is a third window:

```
<ARCHIVE>/09-software/pmtiles-cli/pmtiles serve <ARCHIVE>/08-maps --port 8081 --cors="*"
```

On Windows the program is `pmtiles.exe` in that same folder. On Ubuntu, unpack `go-pmtiles_1.31.2_Linux_x86_64.tar.gz` from that folder first.

Correct output is one line naming the port:

```
Serving . on port 8081 and interface 0.0.0.0 with Access-Control-Allow-Origin: *
```

`--cors="*" is not decoration.` Without it the map draws nothing and the browser reports a security error rather than a missing file. If the map is blank, check that flag first.

Then open:

```
http://localhost:8090/map/
```

A world map, centred on Caracas. Two checkboxes: **hillshade** turns on the terrain shading, **3D terrain** tilts the view. Drag to pan, scroll to zoom.

What to expect. It draws roads, water, borders, place names and land cover down to street level, and elevation everywhere. It has no satellite photography, no traffic, no search box and no routing. It cannot tell you how to get somewhere; it can show you what is there.

On a phone or tablet on the same WiFi, use the node's address instead of `localhost` — the same address chapter 3 gave you, with `/map/` on the end. Nothing needs editing for this to work.

Finding a place by name

The map has no search box. *Where is Maracaibo* is a different command, and it needs no server at all, not even the tile server:

```
python <ARCHIVE>/bin/index-gazetteer.py --find maracaibo

1 place named 'maracaibo':
  Maracaibo          locality  pop_rank=12    10.6498,  -71.6418  (z3)
```

Those two numbers are latitude and longitude, and they are exactly what the climate lookup has always wanted:

```
python <ARCHIVE>/bin/koppen-lookup.py 10.6498 -71.6418 Maracaibo
```

If that says `rasterio is not installed`, it is the right command and the wrong `python`.

Reading map rasters needs a separate environment, and its path is the last line of the table at the top of this manual. On the machine this was built on it is `C:\ark-env\Scripts\python.exe`, so the command becomes:

```
C:/ark-env/Scripts/python.exe <ARCHIVE>/bin/koppen-lookup.py 10.6498 -71.6418 Maracaibo
```

The node prints the working form for you: `/api/spatial` checks which interpreter can import rasterio before it writes the command out.

Many places share a name, so the answer is a list. `--find franklin` returns thirteen, ordered by population rank, largest first. The archive cannot know which one you meant. Take the top line without reading the coordinates and you can be eleven hundred kilometres from where you thought.

Ask in any language it holds. 46,861 places under 271,848 names in 42 languages: `--find tokyo` and `--find 東京都` reach the same point, and so do `--find cairo` and `--find القاهرة`.

If a place is not there, there are two reasons and the second is the common one. It was built from zoom levels 0 to 8, which is towns rather than villages. And it was built from the map's `places` layer, which labels **towns, regions and countries only** - a lake, a mountain, a national park or a set of ruins is drawn on the map and carries no searchable name at any zoom. `Lake Titicaca` will never be found by name; the lake is on the map. Find it by eye and point at it: the panel now shows the latitude and longitude under the cursor, and the exact climate command to paste. On a phone, tap instead; tap again to release.

If the map is blank or has no labels

WHAT YOU SEE	WHAT IT MEANS
grey screen, message about tiles	window 3 is not running, or <code>--cors="*"</code> was left off
map draws but no place names	the page and its fonts are being served from different places. Use <code>http://localhost:8090/map/</code> , not the file on disk
map draws, hills look wrong	terrain is drawn from elevation data, not photographs. Shading is exaggerated on purpose
nothing at all, page blank	the node itself is not running. Go back to chapter 3, window 2

The map cannot damage anything. It only reads. If it misbehaves, close the window and the rest of the system is unaffected.

4. Asking it a question

Type a question and press Search. Results appear on the left.

The mode selector, beside the search box:

SETTING	WHAT IT DOES
keyword	matches the words you typed. Always available
hybrid	also matches meaning, so it finds passages that use different words. Needs more of the system running. The first hybrid search takes up to a minute , and each one after it takes about ten seconds

hybrid is already chosen when the page opens. You do not have to select it. It became the default because a keyword-only search for *what do I do for someone in shock* returned a physics passage about shock waves in the third slot, and hybrid does not make that mistake.

If you choose **hybrid** and the screen says it ran as **keyword**, that is not a fault. It is telling you the truth about what it did, and chapter 3 says how to find out why.

On the right is the answer, when the models are running. If they are not, it says so plainly and the passages on the left are still real. To start them, see chapter 3; if the launcher declines to, chapter 6 symptom E says what that means.

An answer has two parts, always in this order.

1. **What the archive supports**, with a [1] on every statement. Every number in this part is copied from a passage.
2. **The model's own answer**, in a block that begins **UNSOURCED - NOT FROM THE ARCHIVE**: a full answer from what the model learned, which nothing on this drive has checked. **Its first line says whether it agrees with the archive, adds to it, or contradicts it. If it says it contradicts the archive, stop and read the passages** (chapter 5).

An answer takes about a minute. The second model's check appears below it a few minutes later.

What the labels mean. There are five, and the screen's own words are in the first column so that the page and this table can be compared without translating between them.

WHAT THE SCREEN SAYS	WHAT IT MEANS	WHAT TO DO
grounded in the archive	every statement traces to a passage shown	read the passage before acting on any number. Chapter 5
grounded in the archive + unsourced note	the answer is grounded and the model also answered from its own knowledge , in the marked block at the end. Since revision 1.5 that block is a full answer, not a note	the cited part is accountable. The block is not. Treat it as something you were told, not something you read
Not in the passages found - model recall only	nothing in this answer came from the archive. The model is saying what it remembers, and saying so	check it somewhere else before acting. It is not wrong by definition, but nothing on this drive supports it
archive does not cover this	the archive does not contain the answer, and the model said so instead of inventing one	a good outcome, not a failure. Try different words; if it repeats, the gap is real. Chapter 9
model only - no citation	an answer with no citation and no admission	the only one on this list that is a fault. Do not act on it, and tell whoever maintains the node

Revision 1.2 had this table wrong in a way worth naming, because a printed copy of it is still out there. It listed `uncovered` as *part of the answer is not supported by the passages*, which is not what that label means, and it gave the correct meaning to a separate row called `NOT IN ARCHIVE`. `NOT IN ARCHIVE` is the phrase the model writes; `archive does not cover this` is what the screen then shows. One label, not two. It also carried none of the three states below the first two, so an operator holding revision 1.2 has no entry for the only label on the list that is a fault.

Two more things on screen, which are not grounding labels:

adjacent	a passage the system fetched because another passage referred to a table it did not contain. It says why underneath
a [1] in the answer	click it. It opens the passage that statement came from

5. The one rule

Read the source passage before acting on any number.

Doses, pressures, voltages, temperatures, tolerances, timings. The system marks these when it notices them and tells you to verify. That warning is not a formality and it is not there because the system is unreliable in general.

It is there because **the failure this system produces is not an error message. It is a confident, well-written, correctly-cited answer with something wrong inside it.** That has happened in testing, more than once, and each time the answer looked exactly like the correct ones.

Click the [1]. Read the passage. Confirm the number is written there, with the same units, in the same context. Then act.

And write it down or keep it on screen. Do not act on a number you heard spoken and did not read.

6. When something is wrong

Find your symptom. Do only what it says.

A. kiwix-serve says the library file does not exist, or it starts and every document link gives an error page. The library is machine-specific and is rebuilt, not restored. **Both symptoms have the same cause and the same fix**, and the second one matters more, because the server will say it loaded successfully. Run:

```
python <ARCHIVE>/bin/kiwix-library.py
```

It writes `<ARCHIVE>/library.xml` and prints two lines, both beginning `all`. Then try chapter 3 again.

B. The browser shows nothing at `localhost:8090`. Look at window 2. If it ended with an error, write the last three lines down and go to **G**. If it is still running, try `http://127.0.0.1:8090/` instead.

C. Search returns nothing at all, for everything. The index is missing or unreadable. Go to chapter 8.

D. A `[1]` opens the wrong passage, or nothing. Stop using the answers. This means the index and its map disagree. Run:

```
python <ARCHIVE>/bin/index-provenance.py --rebuild
```

Wait for `0 artifact(s) still pending`. About ten minutes; it has not hung. Then restart window 2.

E. The screen says the model is not running. Search and passages are unaffected either way. To start it:

```
python <ARCHIVE>/bin/ark.py status
```

If `primary` says `down`, run `python <ARCHIVE>/bin/ark.py up`. Loading takes about a minute; `status` again will show it.

If the launcher refuses, read the line it prints. There are two kinds and they mean different things.

1. **It names a missing file**, usually something to unpack from `09-software`. Do that, then `up` again.
2. **It says the machine cannot hold the model**, with two numbers: what the model needs and what the machine has free. **This is not a fault and there is nothing to repair.** This machine runs search, passages and citations, and not the assistant. Chapter 8 says what a machine needs for each model.

The second model, `crosscheck`, runs in ordinary system memory rather than on a graphics card, so a machine that cannot run `primary` may still run that one.

F. A drive is clicking, grinding, or has gone silent when it was not. Unplug it. Do not run anything against it. Note which drive it was.

G. Anything not listed here. This is a defect in this manual, not a mistake by you. Write down: what you were doing, the exact command, and the last three lines on screen. That note is the

most valuable thing you can produce; it is what gets a page added here.

7. Quarterly: checking the drives are still good

Every three months. Takes an hour or more. Not urgent, and not skippable, because the failure it catches is silent.

Files decay on disk without anything reporting it. The file keeps its name, its size and its date, and some of its contents are no longer what was written. The only way to find this is to read every byte and compare it against a fingerprint taken when the file was healthy.

Connect the cold drive (chapter 2), then:

```
bash <ARCHIVE>/bin/backup-cold.sh --verify
```

It will appear to hang. It has not.

This command reads every byte on the drive, more than two terabytes, and prints nothing at all while it does. An hour of no output is normal and expected. Do not stop it. Do not unplug anything. Go and do something else.

The line to wait for is:

```
cold copy verified clean
```

Anything else, including any line containing **FAILED**, means write down exactly what it said and stop. Then eject the drive properly (chapter 2) and unplug it.

On Windows this command needs Git Bash. If it is not installed, this is a builder task, not yours.

This checks the COLD copy. The working drive needs its own pass

The command above reads the cold drive. That is the copy you would restore from, so it is the one that matters most, and it is not the whole job.

Bit rot changes neither a file's size nor its date, so the mirror that refreshes the cold copy skips a rotten source file and copies nothing. The cold copy's older, good bytes then verify clean, and the

working drive is the one carrying the damage. The passage above says the failure is silent; this is how it stays silent even after a clean verify.

The pass that reads the working drive:

```
bash <ARCHIVE>/bin/verify.sh
```

Same shape, same patience, same rule about `FAILED`. Run both, in either order, once a quarter. If you only ever run one, you are checking the copy you do not use.

8. Rebuilding onto another machine

The long chapter. You need the cold drive and a working computer.

Read this before starting

What you get back depends on the machine, and the archive cannot give you everything on either.

If the machine already runs Windows: everything in this manual works, including `hybrid` search. Python and all its packages are on the drive — and **step 3 below is what installs them**. Copying the archive does not.

If the machine is empty: the drive can install Ubuntu on it (`09-software/os-ubuntu`), and you will then have **search, passages and citations, but not `hybrid`** — the packages that half needs are not on the drive for Linux. That is a known gap, recorded in the spec as §12 item 7.

The assistant is a third question, separate from both. The two models are not installed by copying the archive; `llama-server` is unpacked from `09-software/llamacpp-bin` into a folder of your choosing, and the launcher names the exact file and folder when it is missing. Then:

- **`primary`** wants a graphics card with about **15 GB free**. On this build it takes 14,801 MiB of a 16,050 MiB card, so a smaller card will be refused rather than half-loaded. - **`crosscheck`** runs **entirely in system memory** and wants about **21 GB of RAM** and no graphics card at all. A machine with plenty of memory and no usable GPU can run this one and not the other.

A keyword search with a citation for every answer over two terabytes of reference material is a working archive. It is the floor this system was designed to fall back to, not a failed recovery.

1. Copy the archive from the cold drive to the new machine. Use the operating system's own file copy. Copy the whole `civbackup` folder. It is large; expect hours. Verify by comparing the folder size on both sides before continuing.

2. Install Python, if the machine has none. Windows: run `09-software/python-3.12.9-amd64.exe` from the copy and tick **Add Python to PATH**. Ubuntu: it is already installed.

Check it worked:

```
python --version
```

3. Install the packages. WINDOWS ONLY. Copying the archive does not install them. Without them the node runs **keyword only**: search, passages and citations all work, and `hybrid` does not. Nothing warns you in words you would recognise as a warning, which is why this step is numbered rather than mentioned.

```
python -m venv C:\ark-env  
C:\ark-env\Scripts\python.exe -m pip install --no-index --find-links=<ARCHIVE>\09-software
```

No network is used or needed. It takes several minutes and installs 221 packages from the drive. The second command is long; type it as one line.

Check it worked:

```
C:\ark-env\Scripts\python.exe -c "import numpy, faiss, sentence_transformers; print('ok')"
```

It must print `ok`. If it prints anything else, the two commands above did not finish — run them again and read what they say.

There is a script, and it is not for this. `bin/rebuild-venv.sh` builds the same environment with every check written in, but it **refuses to build into** `C:\ark-env` — any spelling of it — because its job is to prove a rebuild works *beside* a keeper that is already running, without destroying it. It is a verification tool. **The two commands above are the recovery procedure**, and they are written here rather than referenced so that a script that will not run is never the reason you are stuck.

On Ubuntu, skip this step. The packages on this drive are built for Windows. You will have search, passages and citations, and not `hybrid` — the §12 item 7 gap named at the top of this chapter.

4. Rebuild the two files that are specific to this machine. Neither is restored from backup, because both describe where things are, and things have moved.

```
python <ARCHIVE>/bin/index-provenance.py --rebuild
python <ARCHIVE>/bin/kiwix-library.py
```

The first ends with `0 artifact(s) still pending`, after about ten minutes. The second writes `<ARCHIVE>/library.xml` and prints two lines beginning `all`.

Do not skip the second one because the file is already there. A restored archive brings a `library.xml` with it, and the paths inside name the drive it was built on. The integrity check does not cover that file, so nothing will warn you: `kiwix-serve` will load it, report success, list the books, and every document link will fail.

5. Start it exactly as in chapter 3:

```
python <ARCHIVE>/bin/ark.py up
```

Read the table it prints. Any part that says `down` names what it needs on the same line. **And read the node line even though it says UP** — the table in chapter 3 says what it can tell you there. If it reports `no keeper venv`, step 3 did not take.

6. Confirm the chain is whole. On Windows:

```
C:\ark-env\Scripts\python.exe <ARCHIVE>/13-ark-node/ark-api/serve.py --selftest --mode hyl
```



On Ubuntu, where there is no hybrid half to test:

```
python <ARCHIVE>/13-ark-node/ark-api/serve.py --selftest --mode keyword
```

This asks real questions and **fails if any result cannot be turned into a citation**, which is the one property worth checking.

`--mode` is not optional here, and leaving it off is how this chapter used to end with a **false result**. The selftest defaults to `keyword`, which needs no packages at all — so on Windows it would pass on a machine where step 3 had never been run, and the chapter would then tell you the recovery was complete over a node with no hybrid search. Asking for `hybrid` is what makes the check able to fail.

Then, by hand: search for something, and click a citation.

The recovery is complete when both are true: the citation opens the source document, and `ark.py status` reports a working `dense:` line on the `node` line — or, on Ubuntu, says keyword only, which is the floor this chapter opened by describing and is a working archive.

9. What this system cannot do

Stated plainly, because a tool whose limits are unknown gets trusted where it should not be.

1. **It is not a doctor, an engineer, or a lawyer.** It finds what published sources say and adds what its models learned, and it marks which is which. Judging whether either applies to the situation in front of you is yours.
 2. **It does not know anything that happened after it was built.** No news, no updates, no corrections published since.
 3. **Its Spanish medical material is thinner and older than its English.** Roughly a quarter the size, and several editions behind on some titles. **Where a Spanish and an English source disagree on a clinical figure, the English one is more likely to be current.**
 4. **It can be confidently wrong.** See chapter 5. This is the failure mode.
 5. **It cannot fix itself.** Everything here either works or reports that it did not.
 6. **Not every machine runs every part of it.** The assistant needs hardware the search does not, and the archive cannot supply hardware. Chapter 8 says what each model needs; chapter 6 symptom E says what to do when a machine cannot meet it, which is nothing.
-
-

10. Who to ask, and what if nobody can be asked

Fill in the box at the front of this manual. If that person is unavailable:

1. **Nothing here is secret.** The archive is open material and these procedures are published.
Anyone comfortable with a computer can read this manual.
2. **The system does not decay if left alone.** It decays if run badly. When in doubt, do nothing and leave both copies unplugged.
3. **Chapter 8 is the whole system.** If a machine dies, the drive plus that chapter is a complete recovery. That is what the drive is for.

End of manual. Every step that could not be completed from these pages alone is a defect in this document. Write it down and it will be fixed here.