



ALGOSHRED TECHNOLOGIES

WORK THAT MOVES ITSELF · PEOPLE TAKE THE ODD CASES

Stop carrying the work. Let each step start the next.

We help you see how work really moves through your business, snip the wasted loops, and link the steps so each one starts the next, with little helpers on the boring buttons and people on the odd cases.

Worth knowing: Start with the simplest thing that works. A linked run still needs an owner who watches it and fixes what it can't.

Stop carrying the work. Let it roll.

Start by following one request: email contact@algoshred.com with one request your team handles and where it seems to loop, and we'll suggest where to start.

Automation & process engineering as a service

Vignesh T.V., CEO & CTO, Algoshred Technologies Pvt. Ltd.

contact@algoshred.com



The problem, and the idea.

Someone has to carry each request from desk to desk, or inbox to inbox.

Nobody can say where a request is without asking around.

The same details get typed into two or three systems.

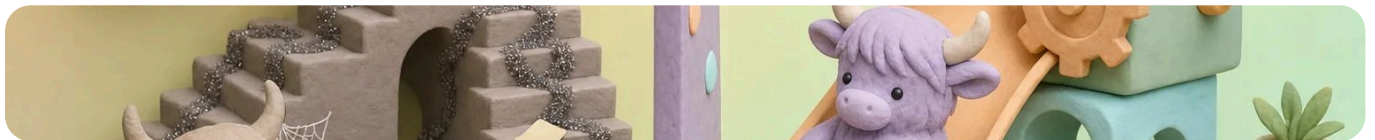
Work goes back and forth for fixes, and nobody has counted the loops.

Software that clicks through screens for someone can break when a screen changes.

Odd requests sit in someone's inbox because no step knows what to do with them.

Process automation means seeing how work really moves, cutting the wasted trips, and then linking the steps so each one starts the next, while people take the odd cases and keep watch.

The story, decoded. In our story, Hana's toy workshop is your business, and the toy orders are your customers' requests. Zippy running up and down the stairs is your team carrying work from desk to desk. The glitter trail is the record your apps keep. The marble run is the linked-up process, and the signs at the forks are your business rules. The wind-up crabs are software helpers that press buttons on screens. Duchess the octopus is an AI helper that sorts odd orders onto the routes Hana built (the story calls them paths). The jack-in-the-box is the call for a person.



01 See the real route first

Follow one real request from start to finish. The trail shows back-and-forth loops the plan on the wall leaves out.

Experts call this: process mining (reading the trail your apps keep), process discovery (finding the real route)

Worth knowing Trails show only what your software records; we also ask the people who do the work. Put a mess on autopilot, and it repeats the mess, so we fix the route first.

02 Each step starts the next

Once the loops are snipped, the steps are linked, people included, so work passes itself along and you can see where each request is. If a step sticks, it calls a person.

Experts call this: workflow automation, process orchestration (one system passing work between steps and people)

Worth knowing Someone on your team still owns the run and takes what sticks.

03 Rules on signs, little helpers on the buttons

House rules sit on a sign anyone can read and change, without rebuilding the run. Little software helpers press the same buttons on screens, the same way.

Experts call this: business rules, decision tables, robotic process automation (RPA)

Worth knowing A new sign gets a quick check before the rush, and a helper needs looking after when a screen moves.

04 Odd ones go to a sorter, or to a person

For orders no sign covers, an AI helper can pick a route, but only from routes you built, and it notes why. If no route fits, a person takes it.

Experts call this: exception handling, agentic automation (AI helpers that choose a next step), human in the loop (a person decides when the helper can't)

Worth knowing The sorter can be wrong, which is why it keeps to your routes and leaves a note.



Follow one order: the route on paper, or the glitter trail.

THE LOOPS ON THE TRAIL	WHAT HAPPENS	WHY IT PROBABLY HAPPENS	ONCE IT IS SNIPPED
Back up: the customer changed the order	The order goes up to the loft, down to paint, then back up because the customer changed it.	Changes arrive by word of mouth, so someone runs them upstairs.	Changes now join the order where it is, so nobody runs them upstairs.
Stuck: a slip left on the stairs	The slip falls on stair nine. Downstairs, the paint bench has nothing to start on, and the customer waits.	The only record of the order rides on a mouse's cap.	The slip rides with the order, so the next step knows to start.
Up again: which colour?	Halfway through, someone climbs back up to ask which colour.	The answer lives in one person's head.	The colour is on the slip from the start.

Steps the trail can't see: A phone call to the customer leaves no glitter, so we ask the people who make the calls.

Roll a marble down this route

NO LOOPS SNIPPED

The marble follows every loop: the same mess, now on autopilot.

SOME SNIPPED

Some loops are still there, and the marble follows those too.

ALL THREE SNIPPED

The marble rolls the short route. Now the steps are worth linking.

Carry it, or let it roll? One order, carried or linked, when something sticks.

The order comes in → **Hana draws it - a person's step** → The crabs press the paint buttons → The sign picks a box → Down the chute

WHAT STICKS	ZIPPY CARRIES IT	EACH STEP STARTS THE NEXT
Nothing sticks	Zippy runs the order up, down and back again. It gets there, after a lot of stairs.	Each step starts the next, and the board shows where the order is.
The paint runs out	Zippy waits at the paint bench, and the next orders pile up on the stairs. Nobody upstairs knows.	The paint step tries again. Still stuck, so the jack-in-the-box pops up, a person tops up the paint, and the board shows which step is waiting.
Someone moves a button	Zippy has to run upstairs to ask where the paint button went, and the order waits.	The crab helpers press where the button used to be and get lost. The paint step doesn't finish, so the jack-in-the-box pops up, and the person keeping watch shows them the new spot.
A slip falls on the stairs	The slip lies on stair nine. The paint bench has nothing to start on, and the customer taps on the window.	There are no stairs to drop it on: the slip rides the run with the order, and the board shows where it is.

A sign, a sorter, or a person? Duchess, the octopus in our story, stands in for an AI helper: a sorting step that picks only from routes you set, and notes why.

ORDER	WHO ROUTES IT	WHERE IT GOES	WHY
A glass teapot	The sign	Bubble wrap	The sign says: fragile goes in bubble wrap.
A wooden car	The sign	Small box	The sign says: wooden goes in a small box.
A barking kite	Duchess, an AI sorting helper	The soft-toy path	Her note says: it barks like the toy dogs, so it goes their way. She can be wrong, so the note shows why.
A life-size unicorn	A person: Hana	Hana's bench	No row and no path fits a life-size unicorn, so the jack-in-the-box pops up and Hana builds it herself.

A sign change: someone adds Metal → tin box and drops Fragile → bubble wrap by mistake. A glass teapot: The fragile row is gone and no row covers glass now, so it slips through to Duchess. She picks the wooden-toy path, and her note says: no glass row, nearest match was wooden. The note shows why, but the teapot goes unwrapped. That's why a sign change gets a quick test. Before the rush, test the new sign with one marble. The test marble was a glass teapot, and it slipped past the sign. So Hana puts the fragile row back before the rush. The run didn't change, only the sign.

Worth knowing

Trails show only what your software records; we also ask the people who do the work.

Worth knowing

A linked run still needs an owner who watches it and fixes what it can't.

Worth knowing

Click-through helpers get lost when a screen changes, so someone has to look after them.

Worth knowing

Rule changes still need a second pair of eyes and a quick test.

Worth knowing

A sorting step can be wrong, so it picks only from routes you set, notes why, and sends anything that fits no route to a person.





What we help with.

Start with one request, or link the pieces into one route your team keeps watch on.



Follow one request from start to finish

We follow a real request through your apps and talk to the people who handle it, using the trail your software keeps where there is one, and watching desk-level clicks only with consent.

A map of the real route, its loops and waits, and which to fix first.

Experts call this: process discovery (finding the real route), process mining (reading the trail your apps keep), task mining (recording on-screen clicks, with consent)

Let each step start the next

Link the steps across your apps and people, so work passes itself along, waits when it should, tries a stuck step again and calls a person when it stays stuck.

Work stops being carried from desk to desk, and you can see where each request is.

Experts call this: workflow orchestration, BPMN (process diagrams a computer can run), durable execution (work that picks up where it stopped after a crash)

Little helpers for repeated clicks, with an owner

Software helpers for the same clicks on the same screens, mainly for older systems with no other way in, plus a check-up of helpers that keep breaking.

Fewer details typed twice, and helpers someone looks after.

Experts call this: robotic process automation (RPA), bot maintenance (keeping the helpers working)

AI helpers for odd cases, on your routes only

AI helpers that sort unusual requests onto routes you built, note why, keep a record, and pass anything that fits no route, or anything you choose, to a person.

AI used where it helps, with people deciding the rest.

Experts call this: agentic automation (AI helpers that choose a next step), human in the loop (a person decides when the helper can't), guardrails (limits on what the helper may do), MCP (a common plug between AI helpers and tools)

Snip the loops before automating

Cut needless steps, back-and-forth, and passing work between people for no reason, and agree who owns each step and what "done" means.

A simpler route that is worth linking up.

Experts call this: process redesign

Put the rules on a sign people can read

Take routing rules out of people's heads and buried code, and write them as tables the business can read, test and change.

One rule, applied the same way, that is easier to change.

Experts call this: business rules, decision tables, DMN (a standard format for decision tables)

Reading forms, invoices and emails into the right boxes

Pulling the right details out of documents into the right fields, with a person checking the ones the helper is unsure about.

Less typing by hand, with a person on the doubtful ones.

Experts call this: intelligent document processing

A view of where work waits, and someone keeping watch

One view of where each request is and where it waits, a few plain measures agreed with you, and a few people in your own team, trained by us, who look after the helpers and pick the next loop to snip.

You can see whether it is getting smoother, in your own numbers.

Experts call this: process monitoring, automation centre of excellence (a small in-house team that owns the helpers)

For your technical team: we can work with common tools such as Camunda, Temporal, Apache Airflow, n8n, Power Automate, UiPath, Automation Anywhere, Celonis and the workflow features inside your business software.

Capabilities, from following one request to a run your team watches.

From finding the real route, to snipping loops, to linking the steps, to rules people can read, little helpers on the boring buttons and an AI sorter that stays on your routes.

See the work

Where work really goes, and what to fix first.

- Following real requests end to end
- Reading the records your apps keep, to see the real route (process mining)
- Watching on-screen clicks at a desk, only with consent (task mining)
- Finding loops, waits and redone work
- Snipping loops before automating

Rules and helpers

Rules people can read, helpers on the boring buttons.

- Business rules as readable decision tables, tested before they go live
- Software helpers (RPA) for repeated clicks on older systems
- Reading forms, invoices and emails into fields

Link the steps

Work that passes itself along, people included.

- Linking the steps across apps and people (workflow orchestration)
- Links between apps, so details are typed once
- Work that picks up again after a failure, with retries and a call for a person
- Trays where odd work waits for a person, with the reason attached

Odd cases and keeping watch

AI only where it helps, and a person watching.

- AI helpers for odd cases, on your routes, with notes of why
- Letting AI helpers use only the tools you allow (MCP, a common plug between AI helpers and tools)
- A view of where each request is, with measures agreed with you
- Owners, standards and training for your own team

WHERE IT BECOMES REAL · ILLUSTRATIVE EXAMPLES

Retail and e-commerce

For example, a return that goes up and down three desks could become one route: returns that fit the rules move on, and odd ones reach a person with a note.

Finance teams

For example, invoices that match the order could pass straight through, and the ones that don't could wait in a tray for a person, with the reason attached.

Insurance

For example, usual claims could follow the same checks, and unusual ones could reach a claims handler with what the sorting step noticed and why.

HR and onboarding

For example, one start date could set off accounts, equipment and training, and the manager can see what is done and what is waiting.

Manufacturing

For example, a quality hold could call the right person and pause the right batch, instead of starting an email chain.

Customer service

For example, repeated requests could get the same answer through the same steps, and anything unusual could go to a person with the history in front of them.

See the route first, then link the steps.



01 SEE

See the real route

We follow one real request through your apps, using the trail your software keeps, and mark the loops and waits we find.

02 ASK

Ask the people doing it

Phone calls and paper leave no trail, so we sit with the people who do the work and add the steps only they know about.

03 CUT

Cut wasted trips

We agree with you which loops, and which needless passing of work between people, to snip first, so the route we link doesn't carry the worst loops with it.

04 LINK

Link the steps

We link the steps for one team and one kind of request, with rules on readable signs, helpers on the boring buttons and a person on the odd cases. Experts call this a pilot (a first small try).

05 WATCH

Keep watch

We set up a plain view of where work waits and hand over the know-how, so your team can look after the helpers, test rule changes and pick the next loop to snip.

Ways to start

Follow one request. A walk-through of one real request, start to finish, and a map of the loops we find.

Snip one loop. Fix and link one painful spot where work passes between people, with one team.

Helper check-up. Look at software helpers that keep breaking, and decide with you whether to fix, replace or retire each one.

What you get

- A map of one real request, start to finish, with the loops marked
- A short list of loops to snip first, agreed with you
- One linked route for one team, with a person on the odd cases
- Routing rules written as a table your team can read and change
- A plain view of where each request is waiting
- Hand-over sessions for the people who will own and watch it



Questions, myths and the next step.

What is process automation, in plain words?

Seeing how work really moves, cutting the wasted trips, and linking the steps so each one starts the next. People take the odd cases and keep watch.

Is it the same as RPA (software robots)?

No. RPA is one kind of helper: software that clicks through screens the way a person would. Process automation is the whole route: finding it, fixing it, linking it, and picking the right helper for each step, including a person.

Will it replace our people?

That isn't the aim. It takes the boring, repeated button-pressing; people keep the odd cases, the judgement calls and the job of keeping watch.

Do we need to buy a big platform?

Not to start. A first fix can be removing a step rather than adding software, and we begin with the tools you already have. Start with the simplest thing that works.

Where do AI helpers fit, and how do you keep them in check?

Only on the odd cases, where fixed steps fall short. They choose only from routes you set, note why they chose, and send anything that fits no route to a person. They can still be wrong, which is why the note and the person are there.

What happens when a step gets stuck?

The run is built to notice: it can try the step again, and if it is still stuck, it calls a person and shows where the work is waiting. Helpers that click through screens also need someone looking after them.

How will we know it is working?

We agree a few plain measures with you at the start, such as how long a request waits and how often it goes back for fixes. The numbers are yours, so you can judge it for yourself.

How long does it take?

It depends on the work. We start by following one request, then agree the next step together.

MYTHS, ANSWERED

“Just automate what we do today.”

Linked up as it is, a messy route just repeats its mess. Follow the route first, snip the loops, then link the steps.

“Once a software helper is built, it looks after itself.”

It presses only what it was shown. Move a button, and it gets lost, so someone has to look after it.

“We already know how our work moves.”

The trail can show loops nobody planned. It shows only what your apps record, so we also ask the people who do the work.

“An AI helper can run the whole thing alone.”

Start with the simplest thing that works. Fixed steps suit work that follows the same rules. An AI helper takes only the odd cases, on routes you set, notes why, and passes anything that fits no route to a person.

How many stair trips does your work take?

Start by following one request: email contact@algoshred.com with one request your team handles and where it seems to loop, and we'll suggest where to start.

contact@algoshred.com

WATCH THE FILM

algoshred.com/process-automation