The Communications Intelligence Playbook

The KPIs to track, the work to automate, and an assessment to find the gap costing you most.

ANALYTICS AUTOMATION AI

VARIPHY.COM/PLAYBOOK

Contents

ABOUT VARIPHY

Two decades of turning communications data into answers.

Variphy is headquartered in California's Silicon Valley and has delivered unified communications analytics and tooling for more than 20 years.

Our mission is to build the most powerful and complete Communications Intelligence platform available, with a personalized level of service for every enterprise and partner we work with.

2,000+	10M+	35+	20+
CUSTOMERS	PHONES AND USERS	COUNTRIES	YEARS IN UC ANALYTICS

SUPPORTED PLATFORMS

An intelligent communications environment can answer questions, act on them, and be reached by asking.

Most organizations have moved communications to the cloud and added platforms along the way: Webex or Teams for calling, Webex or Cisco for contact center, Zoom at acquired sites, and on-premises Cisco still carrying the rest. Each platform works. What is missing is the layer that makes the whole thing legible to the people who run it.

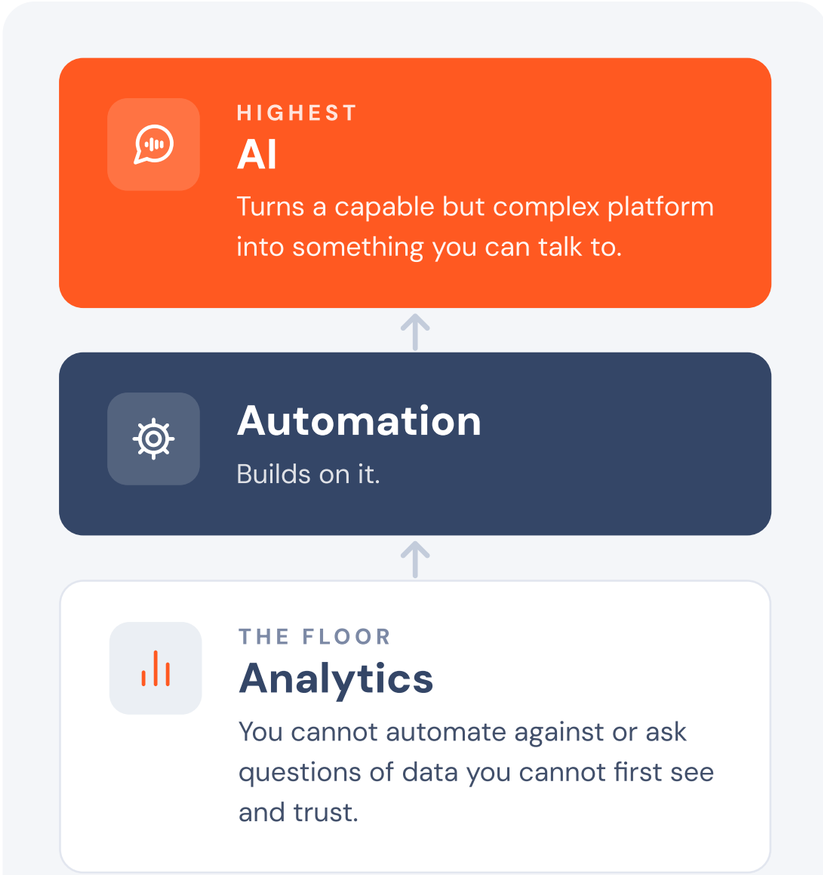
That layer is **Communications Intelligence**, and an environment that has it behaves differently in three ways.

First, it answers its own questions, so "how is call quality trending" is something you pull, not something an engineer builds from exports overnight.

Second, it handles routine work from a defined process instead of an admin clicking through a portal field by field.

Third, it is reachable, by the engineer or the new admin who needs an answer in plain language, and by the caller outside who needs help before a person is free.

Few environments are strong on all three. Use this ebook to guide your own evaluation, chapter by chapter, so you know where you stand and which gap costs you the most.



THE COMMUNICATIONS INTELLIGENCE STACK

Does your environment tell you what's happening, or do you have to piece it together yourself?

Start here, because nothing else works without it. You cannot automate against data you do not trust or ask a question of a system that cannot answer one.

The test is simple to run on your own environment. Pick a question your leadership actually asks, and see how you get to the answer.

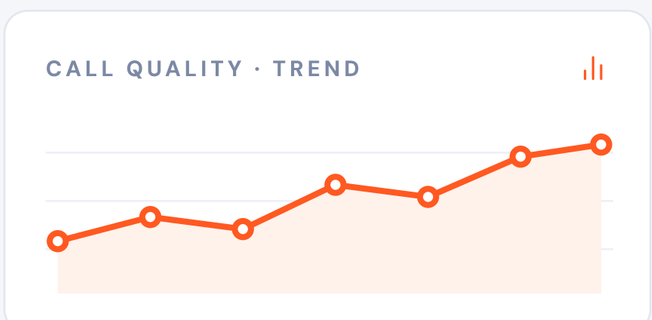
If answering means logging into a native portal, exporting raw records, and building the answer in a spreadsheet, your environment cannot yet answer for itself. You are the analytics reporting layer, and that does not scale.

Most environments sit here, and the native tools are usually why. Control Hub and Analyzer, CAR and CUIC, the Teams admin console: each was built to configure its own platform, not to explain it. The complaints are consistent — no historical depth, data you search by hand, a manager asking for per-extension history the tool will not produce.

An environment that can answer looks different in two ways. Every platform's data lands in one view, so "how is calling overall" becomes answerable. And the answers reach people on their own, through scheduled reports, self-serve dashboards, and a search that returns a dropped call in seconds. The work comes off the one engineer who knows where everything lives.

RUN THE TEST ON YOUR OWN ENVIRONMENT

- ☐ How is call quality trending across all our platforms?
- ☐ Which departments are not using the licenses we bought them?
- ☐ Who changed the call flow before Monday's outage?



What an environment that can answer reports on

These are the KPIs your team should be tracking. The real question is whether you can — across Cisco, Webex Calling, Webex Contact Center, Microsoft Teams, and Zoom Phone, and without rebuilding the data by hand each time.

1

Dropped calls

The rate and pattern of calls ending for technical rather than human reasons. A steady low rate is background noise; a spike on one site, trunk, or time window is a fault worth tracing before the complaints start. During a migration, this is the clearest early signal that the new platform is misconfigured, so watch it on both systems at once.

2

Disconnect cause codes

The reason code behind each ended call. These turn "a call dropped" into "here's why," separating a carrier or trunk problem from your own configuration. A new code climbing the list right after a cutover is usually the first place a migration problem shows.

3

International and toll fraud exposure

Unexpected international or premium-rate calling, whether from an attacker who got into your system or legitimate traffic you were not tracking. Once someone gets in, the charges add up fast, and the same visibility that catches fraud catches international cost you did not know you carried. Routing and permissions are most likely misconfigured right after a move, which is when exposure is highest.

4

Robocall and spam volume

High-volume automated inbound calling that eats call paths and gives real callers busy signals. The signal is volume concentrated on a few calling parties with very short or zero-duration calls. Identifying the top offenders is what lets you block them before they affect capacity.

5

Hunt group and queue performance

Calls presented versus calls answered for the groups that route callers to people. This tells you whether callers reach someone in reasonable time or land in voicemail, watched by answer rate, abandonment, and the volume curve across the day.

The KPI set, continued

6

Trunk and capacity utilization

How much of your calling capacity is actually in use against what you pay for. Over-provisioning wastes money; under-provisioning blocks calls and triggers overage charges. Capacity needs shift as users move platforms, so watching both sides through a migration keeps you from paying for capacity the traffic has already left.

7

Change management and as-built reporting

A searchable record of who changed what, and when. It turns the classic "worked Friday, broke Monday" from an investigation into a lookup. A migration is one long sequence of configuration changes, and a record of them is what lets you roll back cleanly when one breaks something.

8

Dial plan, DID, and DN management

The state of your numbers: assigned, reserved, available, active, and where. In a spreadsheet this drifts the moment someone forgets to update it. Porting and reassigning numbers across platforms is where migrations strand users without working lines, which makes this one of the most migration-critical metrics in the set.

9

Abandoned calls

Callers who give up before reaching anyone. A rising rate is a direct readout of frustration, usually from understaffing or misrouting, and when it peaks tells you when to add coverage.

10

Contact center performance

The operational metrics for queues and agents: queue time, handle time, abandonment, and agent-level activity. Consistently high transfers or long handle times for one agent can flag a training gap before it becomes a customer complaint.

THE RECURRING ASK: ALERTING

The recurring ask from teams running real environments is to be told when a metric crosses a line, whether that is phones dropping offline, a queue past its abandonment threshold, or concurrent paths near the limit, rather than having to go look.

Does routine admin work run itself, or does someone have to click through a portal?

Once your data is trustworthy, the next question is how much of the repetitive work still lands on a person. Onboarding a site, provisioning a hundred users, or repeating a change across locations means clicking through the same screens every time, with no memory of the last pass.

To find out whether you have an automation layer or just people absorbing the work, ask the five questions alongside.

The more of these that land on the second answer, the more of your administration is riding on people rather than process. Manual provisioning drifts, because no two admins do it identically, and that drift is what someone digs through later when something breaks.

An environment that can act turns a defined standard into a job that runs the same way every time. Provisioning becomes one action with the configuration built in, not a sequence of manual fields. Number assignments and changes leave a record as they happen, so the work is accountable after the fact rather than reconstructed.

PROCESS, OR PEOPLE ABSORBING THE WORK?

- ☐ When you add a site, does the effort scale with templates, or with hours at a console?
- ☐ Is provisioning one defined action, or a sequence of manual fields repeated for each user?
- ☐ Does adding the tenth site cost about what the first did, or proportionally more?
- ☐ Can two different admins provision the same way, or does each do it slightly differently?
- ☐ When a configuration breaks, can you see what changed, or do you have to hunt for it?

What this looks like in the workflow

The difference between manual and automated is clearest in the tasks you do most.

Onboarding a user

By hand, a new hire means working through a portal to assign a number, set features, apply policy, and attach a device, then repeating it for the next hire. From a template, it is one job with the standard built in, and the new hire has working calling on day one without anyone remembering which boxes to tick.

Changing a setting across locations

Updating one setting across forty sites by hand is forty chances to do it forty slightly different ways. Run as a single validated job, the change applies everywhere at once, and you can confirm it took instead of spot-checking sites and hoping.

Assigning and porting numbers

Manual DID work lives in a spreadsheet that stays accurate until someone forgets to update it. Managed as a task, every assignment is recorded as it happens, so the inventory matches reality and porting a block during a move does not leave users without working lines.

With the routine handled, what is left is whether the people who depend on the environment, and the callers outside it, can get to it by asking. **That is where AI comes in.**

Can the people who rely on it, and the callers outside it, get what they need just by asking?

Once an environment can answer questions and handle routine work, the last capability is reach: getting to all of it in plain language.

Facing your team

A good analytics platform holds a lot, and that depth has a cost. Someone has to know where the data lives and how to build the report to get an answer out of it. An assistant takes that step away. You ask for last week's call quality issues, or the sites with the lowest adoption, and the answer comes back without anyone building a query first. What matters here is who can now get an answer.

Suddenly the work that used to route through one person happens directly.

These are the same questions from the analytics section: quality trends, adoption gaps, what changed and when. Reaching them no longer depends on knowing the tool, only on knowing what you want to ask.

WHO CAN NOW GET AN ANSWER

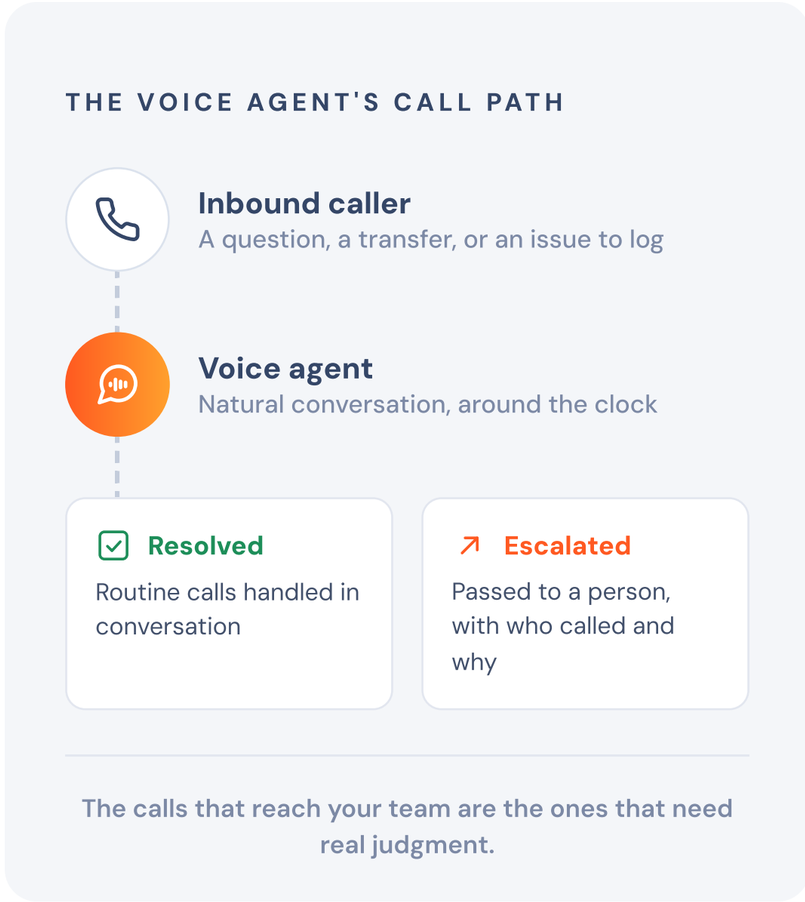
- The supervisor who wants this morning's queue numbers asks for them, instead of filing a request and waiting.
- The admin three weeks into the job gets a usable answer without first learning where every report lives.
- The MSP managing a dozen customers asks the same question across all of them, rather than rebuilding it in each.
- The engineer who used to be the report desk stops getting interrupted for numbers anyone could now pull.

Facing your callers

A large share of inbound calls are routine. A caller wants a question answered, a transfer to the right department, or an issue logged. A voice agent handles those in natural conversation, around the clock, and passes the rest to a person. What separates it from the phone menu you already have is that it understands what the caller is actually asking and acts on it, so the calls that reach your team are the ones that need real judgment.

Some questions to place what you actually need:

- ☐ **Do your callers ask things with consistent answers, or is every call different?** Consistent answers are where a voice agent earns its keep.
- ☐ **Is call volume high enough that routine calls crowd out the ones that need a person?** If not, your problem may be routing rather than AI.
- ☐ **Do you need the system to understand natural speech, or just to route a keypad choice?** If a menu still works, an IVR may be all you need.
- ☐ **When a call escalates, does the agent need to pass along who called and why, or is a cold transfer fine?** An informed handoff means a faster resolution.



What you need in place first

A voice agent is only as good as what you give it to work with. Before it can resolve a call, a few things have to exist for it to draw on:

Something to answer from

The agent resolves questions against a knowledge base, FAQ, or policy you provide. If that information is not written down anywhere, the agent has nothing to answer with.

A directory to route against

Sending callers to the right person or department means the agent needs your directory, whether that lives in Webex, Microsoft Teams, Active Directory, or a spreadsheet, and it needs to be current.

Defined escalation logic

The agent needs to know which calls it should hand to a person and where those go, so the rules for escalation are something you decide before it goes live, not after.

Somewhere to send tickets

When a call needs follow-up, the AI voice agent creates a ticket in a system like ConnectWise or ServiceNow. The value of that depends on the destination being one your team already works in.

WHY THIS MATTERS

None of this is heavy lifting, but skipping it is why voice agent rollouts stall. The technology answers calls on day one; what makes it useful on day one is having prepared what it answers with.

With reach in place, the three capabilities are working together: an environment that can answer, act, and be asked. The next step is seeing where yours stands on each.

Migration and multiple platforms

The three gaps are easiest to ignore in a steady state. They open at once the moment you start moving between platforms.

An environment running one stable platform can paper over a weak analytics, automation, or AI layer for a long time. The native tools are familiar, the routine work is a known quantity, and nobody is asking questions the old reports cannot answer. Migration ends that.

When you move from Cisco to Webex Calling, or stand up Teams Phone alongside what you already run, you are operating two environments at once for months. Calls live in both the old system and the new one, and no native tool reports across the seam.

The question leadership asks during a migration is exactly the one neither platform can answer alone: **Is the new system performing as well as the old one?** Without a layer that sees both, you are back to exporting from each and comparing by hand, at the precise moment you can least afford to be guessing.

The administrative load doubles in the same window. Every user moved is provisioned again on the new platform, and every difference in how it was done becomes drift you inherit on the other side. An environment that provisions from a template carries its standard across the move. One that provisions by hand carries its inconsistencies across instead.

An environment that can already answer, act, and be asked treats a migration as a managed transition. One that cannot treats it as months of manual reconciliation.

Rate your environment on each dimension

Your profile across the three capabilities shows what to build next.

For each row, choose the level that matches your environment today. The levels correspond to a point value. Then, add your score and find out where your environment stands.

●○○ **Level 1** = 1 point ●●○ **Level 2** = 2 points ●●● **Level 3** = 3 points

Analytics — can it answer?

DIMENSION	LEVEL 1 ●○○	LEVEL 2 ●●○	LEVEL 3 ●●●
Multi-platform visibility	Each platform in its own portal	Some consolidated, gaps remain	Every platform in one view
Troubleshooting	Tracing a bad call takes hours	Possible but slow or partial	Traced end to end in minutes
Adoption measurement	Unknown until renewal	Measured occasionally, by hand	Continuous, by department, site, and feature
Change governance	"Who changed what" is an investigation	Logged but hard to search	Searchable change history

Analytics score: _ / 12

Automation — can it act?

DIMENSION	LEVEL 1 ●○○	LEVEL 2 ●●○	LEVEL 3 ●●●
Provisioning	Manual, field by field	Partial templates, manual steps remain	One defined job with config built in
Bulk change	One user or site at a time	Scripted in places, inconsistent	Run across many at once, validated
Standardization	Each admin does it their way	Standards exist, unevenly followed	Enforced by template every time
Auditability	Reconstructed after the fact	Recorded, not easily reviewed	Captured as it happens

Automation score: _ / 12

AI — can it be reached?

DIMENSION	LEVEL 1 ●○○	LEVEL 2 ●●○	LEVEL 3 ●●●
Internal reach	Every request routes through one expert	Some self-service, limited scope	Anyone gets an answer by asking
Caller-facing handling	Every call waits for a person	Menu or IVR routes but does not resolve	Routine calls resolved, escalations carry context
Fit to need	AI considered without a clear problem	In use but sits apart from the environment	Matched to real call types and volume
Integration	Stands alone, nothing connected	Connected to some systems	Draws on your directory, ticketing, and data

AI score: _ / 12

Score and band each capability

Analytics _/12	Automation _/12	AI _ /12
4–6 Low	**7–9** Developing	**10–12** Strong
Most dimensions are manual; the capability runs on people.	Some are handled, but enough manual work remains that the capability is not dependable yet.	Most are system-handled, and people manage the exceptions rather than the routine.

Reading your profile

Low analytics, anything else.

Start here regardless of the other scores. Automation and AI both draw on data the environment can see and trust, and neither holds up on a weak analytics base. A high AI score sitting on a low analytics score is the least stable profile of all: a system answering from data it cannot fully see will sound confident and sometimes be wrong.

Strong analytics, low automation.

The foundation is built and underused. The data exists to drive provisioning and feed an assistant, and until that happens the cost shows up as manual hours and slow answers you have already earned the ability to remove.

Strong analytics and automation, low AI.

The groundwork is solid and ready for a reach layer. This is the profile where adding an assistant or a voice agent pays off fastest, because both finally have trustworthy data and standardized processes underneath them.

Even across all three.

The capabilities compound rather than compete. The work now is widening coverage and scaling efficiently.

BUILD IN DEPENDENCY ORDER

Whatever the shape, build in dependency order. Analytics carries automation, and both carry AI. A weak layer low in the stack caps the value of everything above it.

07 · WHAT STRONG LOOKS LIKE ACROSS ALL THREE

One example of an environment that scores level 3 on every capability, and where to start if yours does not yet.

ON ANALYTICS

Variphy brings Cisco, Webex Calling, Webex Contact Center, Microsoft Teams, and Zoom Phone into one view. Call records, quality data, and configuration history are searchable in one place, reports schedule and deliver themselves, and a single call can be traced end to end in the time the assessment calls level 3.

ON AUTOMATION

Variphy turns provisioning into defined jobs rather than manual screens, for Webex Calling and for Microsoft Teams DID assignment, with a record of every change as it happens. The standard is enforced by template, so the tenth site costs about what the first did and the drift that manual work leaves behind does not accumulate.

ON AI

Variphy covers both directions the framework describes. Colleen AI Assistant is the plain-language way in: ask for the call quality issues or the adoption gaps, and the answer comes back without building a report. Colleen AI Voice Agent answers inbound calls in natural conversation, resolves routine requests from a knowledge base, routes and escalates with context, and creates tickets in ConnectWise or ServiceNow, around the clock and in front of any hunt group or call queue.

More than 2,000 organizations already rely on Variphy communications intelligence solutions.

Where to start

The fastest way to see where Variphy fits your environment is against your own data.

→ **Book a demo** and discover what it can do for your unique environment.

→ **Get a proof of concept** in your environment and your data, and judge it on your results.

VARIPHY.COM/PLAYBOOK