AGNTCON + MCPCON JAPAN 2026

Legacy Meets LLM: Giving Frontier Models a Telephone (PSTN) over MCP

Testing our voice AI by
giving another AI a telephone

Shinya Saito / Gen-AX

2026-09-10

X-Ghost

Gen-AX

Agenda

Gen-AX

1. About me and X-Ghost ("Cross Ghost")
2. Testing a voice AI: the problem
3. Giving an AI a telephone: the MCP server
4. Putting a phone call behind MCP
5. What we use this MCP server for
6. Where operating the AI tripped us up

About me

Gen-AX

Shinya Saito

Backend engineer, Gen-AX Corp.

> Building and operating X-Ghost, our voice AI agent

> Built the telephone MCP server featured in this talk

Career

2016 Piala Inc.

2020 CyberAgent, Inc.

2025 Gen-AX Corp.

Gen-AX Corp.

Gen-AX

A wholly owned SoftBank subsidiary AI experts from across the SoftBank Group

We help companies drive AX (AI Transformation)

Generative AI SaaS

Building and running SaaS products powered by generative AI
Including X-Ghost ("Cross Ghost")

Consulting

AI strategy and organizational design

What is X-Ghost? ("Cross Ghost")

Gen-AX

A voice AI agent for contact centers

An AI operator that puts hospitality into every call

- AI **answers** the phone and handles customers in real time with **natural speech**
- A monitoring AI scores risk and enforces guardrails to **keep calls safe**
- Generally available since **November 2025**

Case study: Sumitomo Mitsui Card Company, Limited

X-Ghost deployed in the contact center

Announced December 2025, the first generative-AI project under the alliance

Background

- About 500,000 inquiries a month
- Needed more support capacity

What we did

- Deployed **X-Ghost** for inquiries about impersonation scams
- Risk scoring and guardrails **keep calls safe**


プレスリリース

三井住友カードのコンタクトセンターにGen-AXのAIオペレーターを導入

～24時間365日応対可能なコンタクトセンターの構築に向けて～

2025年12月3日

三井住友カード株式会社

Gen-AX株式会社

Have a good Cashless.

SMBC 三井住友カード

Gen-AX

三井住友カード株式会社（本社：東京都江東区、代表取締役 社長執行役員 CEO：大西　幸彦、以下：三井住友カード）と、Gen-AX株式会社（ジェナックス、本社：東京都港区、代表取締役社長 CEO：砂金 信一郎、以下：Gen-AX）は、Gen-AXが開発した自律思考型AIの音声応対ソリューション「X-Ghost（クロスゴースト）」を三井住友カードのコンタクトセンターに導入し、音声生成AIを活用した自律思考型AIサービス（以下、AIオペレーター）の提供を、2025年12月3日に開始いたします。

これは2025年5月に三井住友カードとソフトバンク株式会社が締結したデジタル分野における包括的業務提携の一環であり、生成AIを活用したビジネス創出の第1弾です。

<お客さま>　<AIオペレータ>

電話　照会　回答

X-Ghost

Gen-AXの自律思考型AI音声応答ソリューション

SECTION 2

Testing a voice AI: the problem

Testing a phone agent requires a conversation

Gen-AX

Web service

```
POST /orders
```

↓

```
{"status":"completed"}
```

↓

✓ expected result

can be tested automatically

Phone agent

☎️ place a call

↓

a conversation starts

↓

did it respond as expected?

👤 a person has to call and verify the response

So a person had to place the call and judge the response

Doing it by hand does not scale

Gen-AX

Call, talk, and check that the response is what we expected. **All of it by hand**

Does not scale

To test 46 concurrent calls, you need **46 people**

Takes time

A person is tied up for **two or three minutes** per call. Longer if complex

Costs money

Every run burns people's time. You can't run it often

Could an AI do the calling, talking, and checking?

SECTION 3

Giving an AI a telephone: the MCP server

An AI cannot place a phone call on its own

How the telephone MCP server works

Gen-AX

AI

Claude / Codex

an instruction →

MCP server

drives the call

over the PSTN →

X-Ghost

voice AI agent

The MCP tools we expose

place a call **speak** **read the reply** **hang up** **get a call** **list my calls**

Four to converse and two to observe. Six tools are enough to **hold a phone conversation**

Example instruction

"Call 03-1234-5678 and ask about my recent card activity."

What it takes to hold a conversation

Gen-AX

① Listen → ② Think → ③ Speak

Let us look at each step in turn

① Listen

Gen-AX

① **Listen** › ② Think › ③ Speak

The callee's speech is transcribed in real time

X-Ghost (the callee):

partial "Thank you for call..."

partial "Thank you for calling. How can I..."

final **"Thank you for calling. How can I help you today?"**

...silence... → treated as end of turn

Act only once the final result arrives

We once detected the end of a turn too early and cut the callee off mid-sentence

② Think

Gen-AX

① Listen › **② Think** › ③ Speak

Input (transcript)

"Thank you for calling. How can I help you today?"

read →

Reasoning model

generate →

Output (reply text)

"I'd like to check the recent activity on my card."

Real-time speech-to-speech models

Low latency and natural, but **hard to keep on script**

✓ Our pick

Reasoning models

A little slower, but **better at following instructions and staying on script**

③ Speak

Gen-AX

① Listen › ② Think › **③ Speak**

What the calling AI says next

"I'd like to check the recent activity on my card."

Text-to-Speech →

Audio

onto the phone line →

X-Ghost hears it on the line

Real audio over the PSTN. **A true end-to-end test**

SECTION 4

Putting a phone call behind MCP

The server is stateless. The call is not.

Gen-AX

🤖 the LLM calls `create_call()`

MCP server

STATELESS

Requests can arrive at any replica

`call_id` →

☎️ Call

STATEFUL

dial › ring › talking › ended

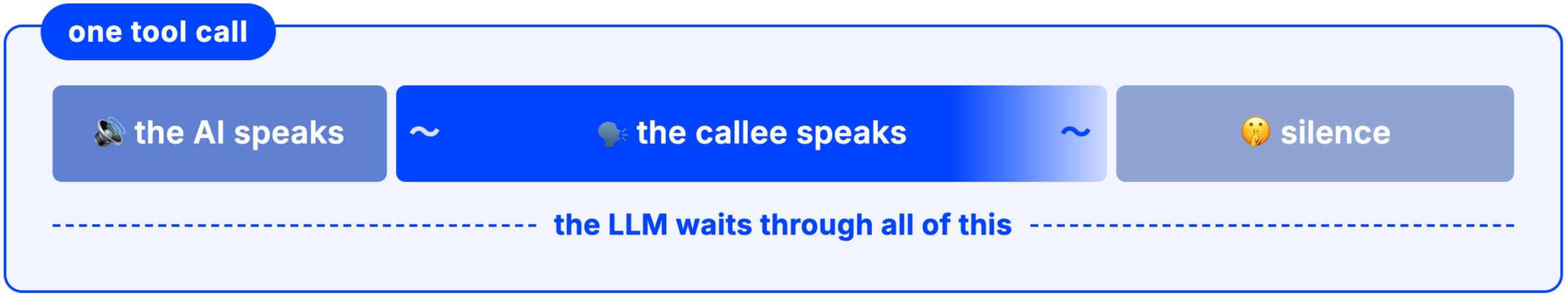
The call keeps moving even when MCP is idle

The MCP server holds no state. The call does

Making MCP stateless does not make the call stateless

The phone won't wait. The tool has to.

Gen-AX

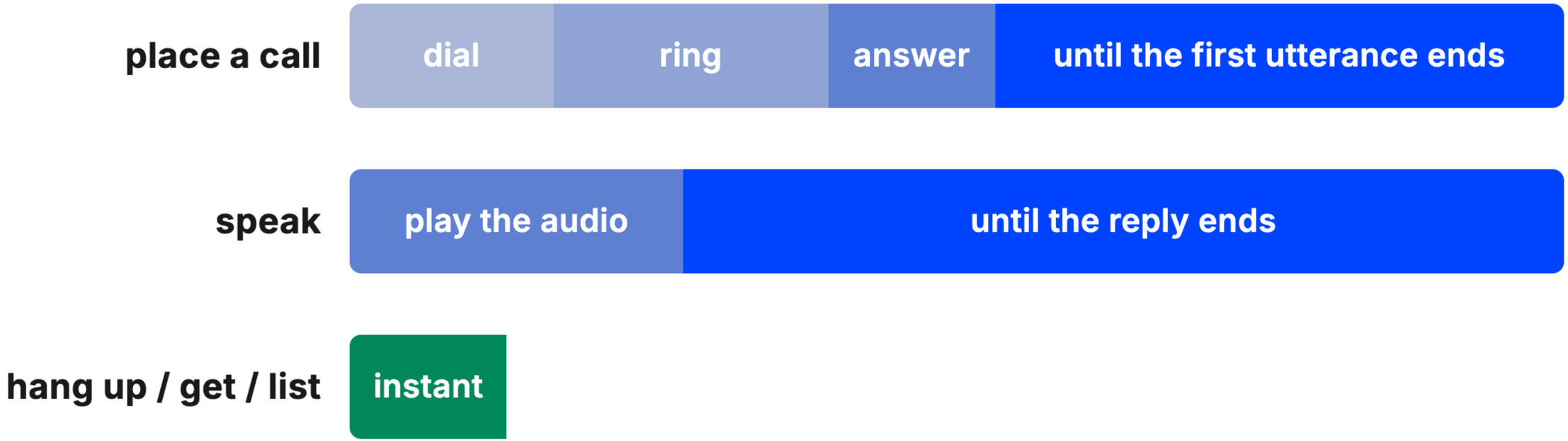


How do we get the reply? ✕ keep asking ✕ be notified ✓ wait right there

One tool call = one turn of the conversation

Nothing returns until the callee finishes speaking

Gen-AX

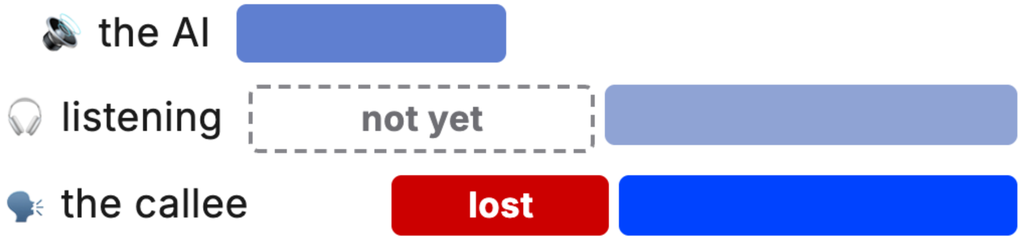


The server does the waiting. The model just calls and waits

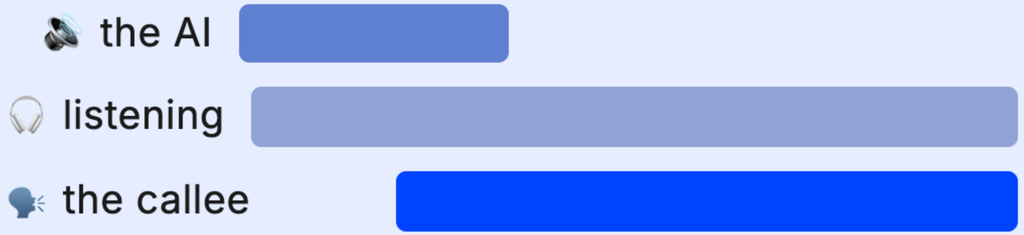
Start listening before you speak

Gen-AX

❌ **Too late: listen after speaking**



✓ **Correct: listen before speaking**



Anything said before listening starts is lost

What we use this MCP server for

Gen-AX

Deep eval

Does it behave as expected?

Talk through a scenario,
let an AI judge the response

Load

Can it handle the load?

Short calls,
many at the same time

Ping

Is it up?

Around the clock,
with a cheaper model

Same tools every time. Only the prompt we hand the AI changes

Use case ① Verifying a re-architecture

Gen-AX

We are actually rebuilding the X-Ghost dialogue engine, and this is how we verify it

🤖

The AI runs the same scenario against both

through the MCP server

→

☎️ **Current X-Ghost**

☎️ **Rebuilt X-Ghost**

→

🔍

Compare the behavior

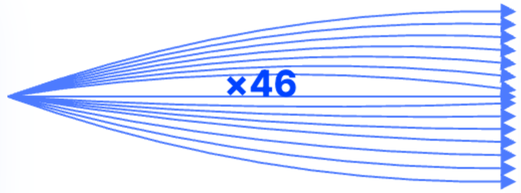
conversations and stored data

Check that the rebuilt system behaves the same, without anyone listening to every call

Use case ② Load testing

Gen-AX

MCP server
the AI dials out



X-Ghost
answers in parallel

Result **46 calls** at once. Parallel call testing without a crowd

Before

Round up a crowd and test by hand

→

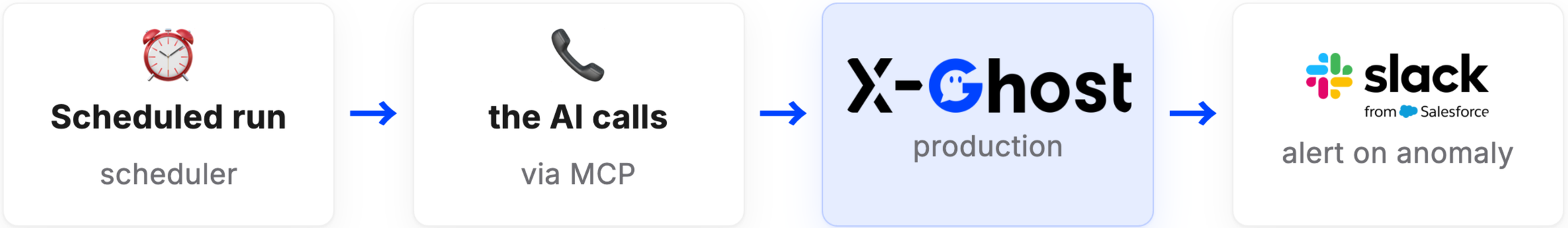
After

A skeleton crew plus AI

Use case ③ Synthetic monitoring

Gen-AX

On a schedule, an AI places a real call and checks the entire flow end to end



No humans on the phone

Nights and weekends too

End-to-end quality

The whole call, in production

Immediate detection

A Slack alert triggers incident response

Where operating the AI tripped us up

Gen-AX

Problem 01 **We hit the rate limit**

Running many agents in parallel drains the subscription quota in minutes

→ Moved from the subscription plan to **API-based usage**. No subscription-level usage cap

Problem 02 **The agent hesitated to dial**

From the outside, repeatedly calling the same number **looks like harassment**

→ Documented in `CLAUDE.md` that the number is **our own test line**

Problem 03 **Launching a fleet was slow**

Subagents start one at a time, so a large fan-out takes minutes to get going

→ Dropped subagents and launched top-level agents with a **GitHub Actions matrix**

What we fixed was not the code. It was **how we run the AI**

We Are Hiring!

Gen-AX

Gen-AX is full of engineers who get the most out of AI

we just have not told the world yet

- A **software company that builds its own generative-AI products**
- A team of **expert AI users**, not just AI builders
- A **small, senior team that makes decisions fast**
- Shipping **products like X-Ghost** at the frontier

Come build with us at Gen-AX

Takeaways

Gen-AX

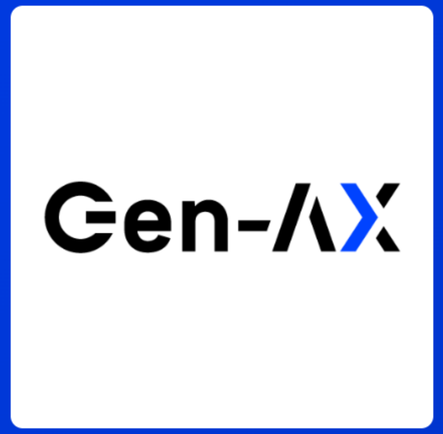
AI calls our voice AI to test it

> We built an **MCP server** that gives an AI a telephone. One tool call covers one turn

> **A tool that waits** is not specific to phones. The same idea applies to deployments, human approvals, and long-running jobs

> Testing that used to need a crowd now runs with **a skeleton crew plus AI**

AI products, tested and monitored by AI


Gen-AX