亚马逊云科技

中国峰会

2026年6月23日-24日 上海 · 世博中心

亚马逊云科技
中国峰会

构建 Agentic 可信查询链路

付德义

基础架构与云平台赋能负责人（Data & AI）

斯堪尼亚制造（中国）有限公司

王维超

解决方案架构师

亚马逊云科技

Agenda
议程

① 可信的起点：在亚马逊云科技上建好的数据底座

② 结构化的可信查询：从数据湖仓到 Agent 数据查询的工程实践

③ 非结构化的可信查询：通用 RAG 与 GraphRAG 的工程实践

④ 可信查询链路：把结构化与非结构化收进同一范式

Scania — 从瑞典 1891 到 2025 的如皋

Scania，1891 年创立于瑞典，TRATON 集团旗下全球重型商用车头部品牌。
2025 年如皋工业枢纽落成，中国战略 2.0 启动 —— 今天的分享，发生在这个底色之上。

Scania （全球）

1891 · Sweden

Södertälje 总部 · TRATON 集团旗下

100+ 国家

业务覆盖 · 全球商用车头部品牌

Heavy / Bus / Engines

重型卡车 · 客车 · 工业 / 海事发动机

Scania 中国 （中国战略 2.0）

Rugao 2025

如皋工业枢纽落成 · 制造+研发+供应链+售后

Super Trucks

首批本地化生产 · 下线交付

Sustainable

面向中国市场的可持续运输方案

*前述特定亚马逊云科技生成式人工智能相关的服务目前在亚马逊云科技海外区域可用。亚马逊云科技中国区域相关云服务由西云数据和光环新网运营，具体信息以中国区域官网为准。

①

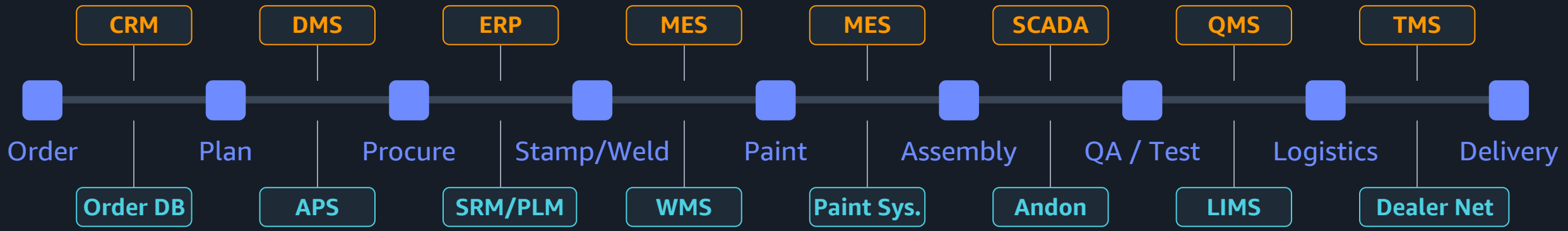
可信的起点

在亚马逊云科技上建好的数据底座

生产现场的数据 — 从订单到交付穿越 30+ 系统

① 可信的起点

一辆 Scania 卡车从订单到下线，要穿越多套业务系统。
数据湖建立之前，这些数据物理隔离 —— 跨系统分析靠人工取数。



多系统 · MES · SCADA · QA · ERP · SCM · CRM · Dealer Net …　——　每一段都是数据，曾经也都是数据孤岛。

可信数据基座——亚马逊云科技中国区端到端架构

① 可信的起点

六层云原生架构：摄取 / 存储 / 处理 / 治理 / 服务 / 可观测。

Governance · Lake Formation · Glue Data Catalog · IAM

On-Prem	Ingestion	Storage (S3 + Iceberg)	Processing (Spark on EKS)	Serving
Source DBs ME S · ERP · SCADA · QMS · CRM · SCM … 30+ 系统 **External Files** 供应商 / 合作伙伴文件 (SMB)	**Database Migration Service (DMS)** CDC + Full Load · Multi-AZ · Secrets Manager **DBImport** Spark on EKS + MariaDB cfg · Airflow DAG **File Drop** NiFi → 跨账户 IAM → Lambda 服务	**Landing Zone** S3 · 短期落地 · 生命周期清理 **History Zone** S3 + Iceberg · INSERT-only · 全量血缘 **Raw Zone** S3 + Iceberg · MERGE / Upsert	**Engine** Spark on EKS · Airflow on EKS **CDC Streaming / Batch** 近实时 + 小时级 · Synch + Merge **Full Load (DB / File)** 天 / 周 / 月 · MERGE / Replace	**Snowflake (Consumer)** External DB / Iceberg · 只读 · 零拷贝 **Event-driven Refresh** SNS → SQS → Lambda · ALTER ICEBERG TABLE REFRESH **BI · Agent** 下游统一从 Snowflake 消费数据底座

*前述特定亚马逊云科技生成式人工智能相关的服务目前在亚马逊云科技海外区域可用。亚马逊云科技中国区域相关云服务由西云数据和光环新网运营，具体信息以中国区域官网为准。

可信数据基座 — 把「可信」作为底座的工程承诺

① 可信的起点

在数据湖之上做 Agent 之前，先回答一个更早的问题：这个数据底座本身值不值得被业务、被 Agent 信任。

数据可追溯

Iceberg 三 Zone：Landing 落地 · History 全变更（INSERT-only）· Raw 业务可查询。任意指标都能时间旅行回放，异常能被复现。

凭证零泄漏

GitLab Secret Manager Repo + IAC Repo 双仓职责分离；秘密永不入 Git，仅写入 Secrets Manager。

元数据统一 + 细粒度治理

Glue Data Catalog 是所有 Zone 的唯一真相源；Lake Formation 提供列 / 行级 FGAC 与跨账户授权。

全链路可观测 + 多档 SLA

Spark / Airflow / Glue 任务全链路指标进入自研 Observability API；近实时到天级多档 SLA 由四种处理模式覆盖。

—— 这四件事都是「不做就没法上 Agent」的工程前提。可信查询，从可信数据基座开始。

②

为 Agent 提供结构化的可信查询

用语义层和 VQR 为模型划定边界

为 Agent 做好准备 — 我们在 AI 问数场景里看到了什么

② 结构化的可信查询

底座建好之后，下一步是为 Agent 打好可靠的查询底座。
系统化地验证了「自然语言 → SQL」 —— 我们看到了三件事。

选错表 / 用错口径

POC 里的真实错误：业务问「BOMs 缺件」——

模型选了 BOM_V 表 + EPO_STATUS = 'R';

正确口径是 COMP_BOM_RESULT_V + COMPARE_*=Mismatch。

这不是模型不行，是模型不知道业务口径。

业务黑话猜不对

POC 真实错误：业务问「PP4.1 的数据」——

模型完全不知道 PP4.1 = '2023041';

缺件」在 用户 的口径里是 BOM 对比的 Mismatch

不是模型不行，是模型缺一份业务字典。

答得对 ≠ 答得快

ReAct + 多轮反思的 Agent 单次 8–20 秒，

Token 成本随调用量飙升 ——

但更重要的是：业务对 Agent 的第一印象只有一次。

准确度，是 Agent 能进生产的前提。

纯 NL2SQL 错在哪，对的样子是什么

② 结构化的可信查询

工程师真实问句：「2026 年一月份第三批 BOMs 中，缺件多少，前 10 的缺件有哪些。」
左边是模型自由生成 —— 选错表、错口径；右边是语义层治理后 —— 模型只填槽位。

裸 NL2SQL（错）

```
SELECT PART_NUMBER,
       COUNT(*) AS BOM_COUNT
FROM SYS_PNL_P_MONA_CHA_BOM_V
WHERE PART_PERIOD = '202601'
  AND EPO_STATUS  = 'R'
GROUP BY PART_NUMBER
ORDER BY BOM_COUNT DESC
LIMIT 10;
```

✗ 选错表 —— 「缺件」不在 BOM_V，在 COMP_BOM_RESULT_V
✗ 把「缺件」当成 EPO_STATUS = 'R' —— 业务口径错
✗ 「一月第三批」≠ '202601' —— 真实编码是 '2026013'
✗ 错得理直气壮，业务一眼看不出来

语义层治理后（对）

```
SELECT PART_NUMBER,
       COUNT(*) AS BOM_COUNT
FROM SYS_PNL_P_MONA_CHA_COMP_BOM_RESULT_V
WHERE PART_PERIOD = '2026013'
  AND ( COMPARE_SEU_SAS  = 'Mismatch'
     OR COMPARE_MONA_SAS = 'Mismatch')
GROUP BY PART_NUMBER
ORDER BY BOM_COUNT DESC
LIMIT 10;
```

✓ 「缺件」= COMPARE_* = 'Mismatch' （业务口径登记）
✓ PART_PERIOD 格式 '2026013' （字段值口径治理）
✓ 表别名「BOMs / 缺件」→ COMP_BOM_RESULT_V （同义词）
✓ LLM 只填槽位，剩下都是确定性的编译

语义层 — 把业务口径编译成确定性 SQL 模板

② 结构化的可信查询 · 语义层

语义层是可信度的真正来源。把业务口径显式登记成参数化 SQL 模板。
LLM 只识别口径并填槽位 —— POC 通过率 90%以上 。

编译流程

自然语言问题

「华南本月在产订单」

槽位识别

zone='华南'
period='本月'
status=in_prod

口径登记表

in_production_count
(zone, period)
→ 参数化模板

最终 SQL

```
SELECT COUNT(*)
... WHERE
zone_name=:zone
  AND status IN
(:codes)
```

语义层做了什么

- 业务术语字典 —— 「missed」「缺件」「PP4.1」这种业务黑话被显式登记
- 表 / 列同义词 —— 把业务实体映射到具体表与字段（不靠概率）
- 字段值口径 —— PART_PERIOD 的「2026年1月第三批」格式编码 = '2026013'
- 模板与提示词规则 —— 转义、输出语言、多意图拆分等编译期规则
- 治理对象，不是一次建模 —— 每条口径有 owner、有版本、有复核流程

语义层是怎么炼成的 —真实优化矩阵

② 结构化的可信查询 · 语义层是怎么炼成的

POC 是一个工程闭环 —— 每个失败用例对应一次明确动作，立即跑回归。
下表是 20 个用例归纳出的 8 种优化策略 —— 语义层就是这样从 0 走到 95% 的。

错误类型	优化策略	真实例子
表识别错	修改语义模型 · 增加表同义词	「BOMs / 缺件」 → SYS_PNL_P_MONA_CHA_COMP_BOM_RESULT_V
列名识别错	修改语义模型 · 增加列同义词	GROUP_ID vs POPID（车标识列）
业务术语错	修改提示词 · 增加业务字典	「missed / 缺件」 = COMPARE_*='Mismatch'，不是 EPO_STATUS='R'
字段值格式错	修改提示词 · 解释字段格式口径	PART_PERIOD：'2026年一月第三批' → '2026013'，不是 '202601'
转义词错误	修改提示词 · 增加值约束	SQL 单引号转义（''xxx'' vs 'xxx'）
输出语言错	修改提示词 · 增加语言约束	Insight 文字结论的输出语言（中/英）
多意图理解错	修改提示词 · 增加问题拆分规则	「PP4.1 的数据」需要先拆出 PP4.1 = '2023041'
功能性缺失	修改代码 · 增加文件下载等功能	「以表格形式返回」 → 生成下载链接

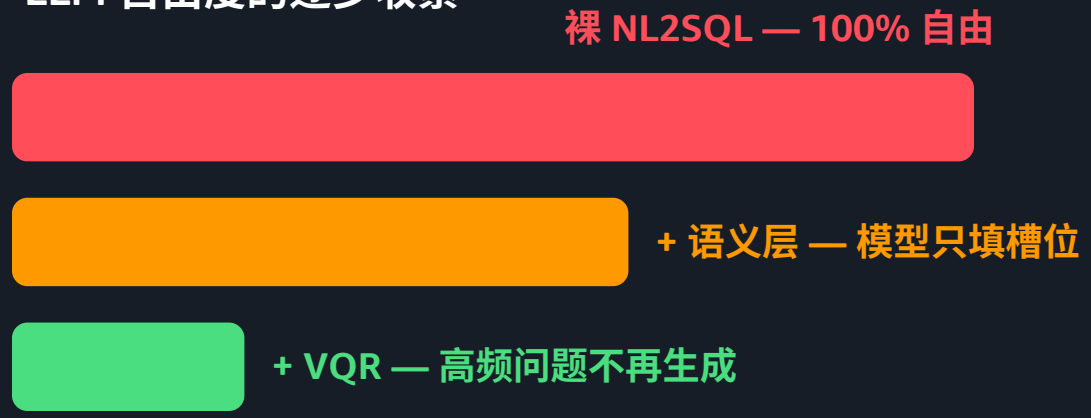
POC 结果：LLM+ 8 种优化策略治理过的语义层 → 20 用例 19 通过 = 95% 通过率（Agent_chat 92.3% · Insight 100% · 多轮对话 100%）

VQR — 当语义层成熟，自然延伸出的工程问题

② 结构化的可信查询 · VQR

高频问题是有限的、重复的 —— 每次都让模型重新生成 SQL，本质是浪费。
VQR 把已被业务验证的问答整条沉淀为资产：语义层管「对的 SQL」，VQR 管「确认过的问答」。

LLM 自由度的逐步收紧



↓ 从「自由生成」逐步降级到「填槽位」再到「直接命中」

推进的方向

- 数据模型：question + 同义改写 + 参数化 SQL + 命中口径 + owner + 验证记录
- 入库门槛：业务负责人复核 + 离线回归通过
- 运行时：语义检索（向量 + 关键词）召回 Top-K → 命中即直出
- 多轮对话已在 POC 验证（2/2 = 100%）：上下文中的口径与槽位一致性，是 VQR 的天然延伸
- 与语义层联动：模板更新 → 受影响的 VQR 资产自动失效

③

为 Agent 提供非结构化的可信查询

通用 RAG 与 GraphRAG 的工程实践

Scania 中国的非结构化数据 — 跨业务域覆盖

③ 非结构化的可信查询

Scania 中国 General IT 横向赋能各业务部门，沉淀了制造、研发、供应链、HR、法务等多类核心技术与业务文档。
让各部门员工用自然语言问到具体段落、表格或步骤，是一个跨业务域的真实生产需求。

文档类型 · 跨业务域覆盖

制造与质量	维修与服务手册 · 工艺 SOP · 质量规范
研发与产品	设计文档 · 标准 / 规范 · 测试报告
供应链与采购	采购规则 · 合同模板 · 供应商规格书
人力资源	员工手册 · 培训资料 · 流程指引
法务与合规	政策文件 · 内控规范 · 合规手册
IT 与信息安全	系统使用手册 · 信息安全规范

我们的目标

- 各部门员工用自然语言提问
- 答复定位到具体段落 / 表格 / 步骤
- 答复带可引用证据，不是一段总结
- 跨文档版本一致 · 跨部门复用

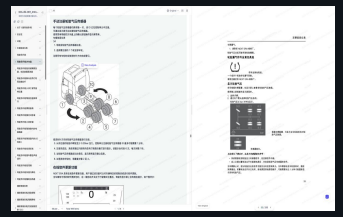
通用 RAG 的工程深度

③ 非结构化的可信查询，为 Agent 提供确定性问答

我们把「日常问答能进生产」做成了五层工程化的能力。

① 高保真文档解析

保留表格 / 公式 / 图注的层级结构 —— 维修手册的零部件位号图、整车 BOM 表能完整召回。

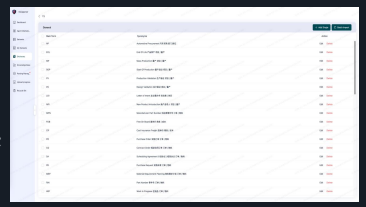


② 切片策略优化

多粒度分块（段落 / 章节 / 表格行 / 图注），元数据丰富 —— 终端用户不用懂分块，由平台默认治理。

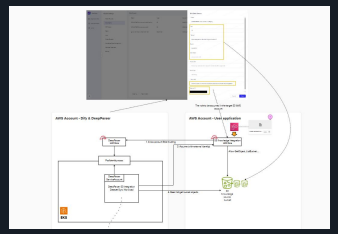
③ 业务术语字典

「AP / EOL / MP / SOP / NPI / FOB / MPN ……」业务黑话被显式登记 —— 模型不再靠概率猜术语。



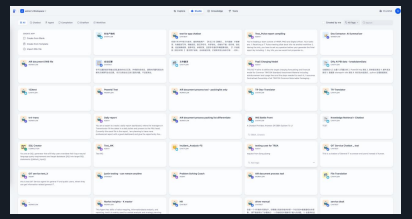
④ 多源知识统一接入

SharePoint（文档库 + 站点） + Amazon S3 + 第三方系统 → 统一在 Dify 知识库；Entra ID 认证、自动化同步。



⑤ 工程化运营

一套引擎 · 多个应用 —— Studio 内 30+ 个 Workflow / Agent / Chatbot 在跑（Daily Report / 翻译 / Service Chatbot / HR …）。



给各部门一个「统一的、可治理的文档智能入口」。

把传统系统聚合到 Agent 时代的统一入口

③ 非结构化的可信查询

工厂里的文档与系统是历史沉淀的：SharePoint、文件服务器、各部门自建系统、共享盘 ……
通用 RAG 把这些来源统一接到一个对话入口 —— 用户不用再「先想去哪查」，直接用自然语言问。
这是把传统 IT 资产，无侵入地升级到 Agent 时代的工程方式。

之前

- 用户要先记住「这个问题去哪个系统查」
- 跨系统切换、跨格式复制粘贴
- 同一问题，不同人问出不同答案
- 文档版本散落，无法追溯

之后

- 一个对话入口，跨业务域、跨系统、跨格式
- SharePoint / S3 / 第三方系统 统一接入
- 答复带可引用证据，可一键回到原文
- 各部门员工直接用自然语言问

工程价值

- 不替换传统系统，而是叠一层智能入口
- 传统 IT 资产无侵入纳入 Agent 时代
- 同一底座，未来支持更多 Agent 应用
- General IT 的赋能能力可被复用、可治理

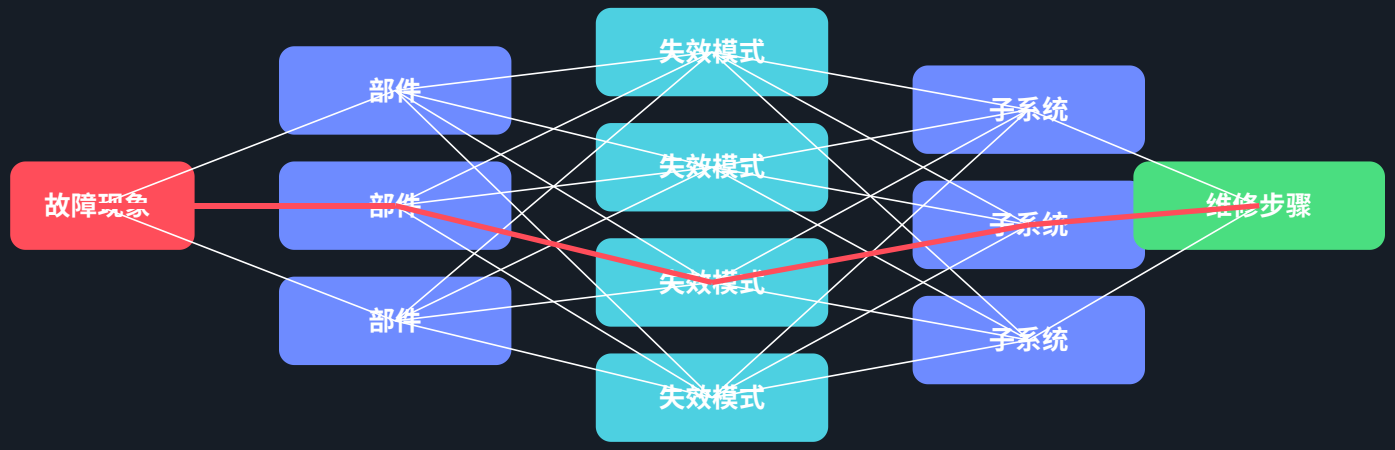
IT 在 Agent 时代的新角色：把传统系统聚合起来，提供可信的、跨域的查询入口。

GraphRAG — 非结构化数据的「确定性查询」

③ 非结构化的可信查询 · GraphRAG

故障诊断是多跳问题 —— 相似不等于相关，全库 Top-K 反而稀释信号。
把技术规范或用户手册抽成本体图谱，向量检索只在图谱框定的子集内进行 —— 召回路径，被图谱约束。

本体图谱（示意）



—— 红色路径示意一次 2–3 hop 的图谱召回路径

召回链路

1. 实体识别 —— 从问句中识别故障现象 / 部件
2. 图谱推理 —— 在 2–3 hop 内找出合理的故障路径
3. 路径绑定文档 —— 路径上的节点关联到维修手册段落
4. 向量检索 —— 只在路径关联的文档子集内做相似检索
5. LLM 推理 —— 仅在「图路径 + 命中段落」上回答

④

可信查询链路

把结构化与非结构化收进同一范式

可信查询链路 — 三步推进，一种工程哲学

④ 可信查询链路

可信查询链路 —— 通过工程资产，逐步收紧 LLM 的自由度。
结构化用语义层 + VQR；非结构化用图谱约束召回路径 —— 三件事，同一范式。

P1 · 语义层

锁住业务口径

把「在产订单」「关键件」等业务术语显式登记成参数化 SQL 模板。

LLM 失去了写 JOIN 与过滤逻辑的自由度，只识别口径并填槽位。

P2 · VQR

沉淀已验证问答

把已被业务确认的问答整条沉淀为资产；运行时同义匹配命中即直出 SQL。

LLM 在高频问题上失去「重新思考」的自由度——它不再生成。

P3 · Graph

约束召回路径

构建本体图谱，向量检索只在图谱框定的子集内进行。

LLM 失去「全库自由联想」的自由度，只在确定子集内推理。

共同范式 —— 先把召回 / 编译路径变成确定性的，再让 LLM 在确定空间里完成最后一步。

可复用的工程经验

④ 可信查询链路

三条可以复用的场景工程判断。

01 给 LLM 划边界

可信查询的本质，是把 LLM 从「生成」降级到「在确定空间里推理」。

语义层、VQR、Graph都不是新组件——它们是给 LLM 划边界的三种工程方式。

02 结构化 ≈ 非结构化 · 跨域同构

Text→SQL 和 RAG 的工程哲学是同一个：先把召回 / 编译路径变成确定性的，

再让模型在确定空间里完成最后一步。

已在 Mona 制造端 + RND 研发端两个业务域同时跑通——同一套范式，跨域复用。

03 可信链路从可信底座开始

Agent 进生产的前提，是数据本身就是可信的——可追溯、可治理、可观测、有 SLA。

没有可信的数据基座，再好的 Agent 工程也站不住脚。

LLM + 语义层治理，自然语言查询通过率 95% (19/20) —— 仍在共同推进可信查询链路的演进。

「让 Agent 答得对，一半靠模型，
一半靠工程把它收进确定空间。」

亚马逊云科技
中国峰会

Thank You · Q & A

感谢倾听 —— 期待与你们一起把 Agent 推进生产

亚马逊云科技
中国峰会

扫描上方二维码
填写调查问卷