



BERLIN · 2026

BUILD BETTER TOGETHER

Overcoming Agentforce Antipatterns.

SPEAKERS

Christopher Ramm & Sebastiano Schwarz



THANK YOU, SPONSORS



YOUR SPEAKERS

Christopher & Sebastiano.

Two perspectives on the same problem and a shared belief that Agentforce only works when the data underneath it does.



Christopher Ramm

DCX CTO GERMANY

Fifteen years in the Salesforce ecosystem. Helps enterprise teams take Agentforce from a flashy demo to something that survives Monday morning.

[linkedin/in/cramm](https://www.linkedin.com/in/cramm)



Sebastiano Schwarz

SALESFORCE CTO GERMANY,
CAPGEMINI

Builds the plumbing behind production agents. Lives in the chunking-strategy details so your retrieval doesn't embarrass you on stage.

[linkedin/in/sebastiano-schwarz](https://www.linkedin.com/in/sebastiano-schwarz)

FIRST, A DEFINITION

What is an **anti-pattern**?

In the Salesforce ecosystem, **anti-patterns** are design or implementation choices that appear logical or effective at first, but create long-term complexity, inefficiency, or risk...

The five we will walk through today are the ones we see in almost every **Agentforce** engagement.

ANTI-PATTERN · 01 OF 05

The Chatbot Agent.

When the answer to every product question is
“put a chat box on it.”

01

They already had a form...now they have a **conversation**.

The Challenge

- Treating Agentforce like a fancy FAQ chatbot
- Typical: over-optimizing prompts and tones...
- ...while under-investing in data grounding, system architecture and tools

Why?

- Legacy mindset from classic chatbots & FAQs
- Underestimation of agentic reasoning + tools + data
- PoC solution stumbles into production



ANTI-PATTERN · 02 OF 05

The Mega Agent.

One agent. Forty topics. A thousand actions.
Zero accountability.

02

“

If your agent does
everything, it does
nothing well.

THE SINGLE SENTENCE TO TAKE HOME



Decompose. Specialise. **Orchestrate.**

WHY IT GOES WRONG

Designing one powerful agent which handles Sales, Service and Operations...

...orchestrates complex workflows...

...and talks to everyone in every context 😂

HOW TO FIX IT

- One agent, one outcome. Service agent, sales agent, IT agent ☐ not one super-agent (Guided Flow helps already)
- Treat each agent like a microservice: bounded scope, explicit contract, owned by a team.

If your agent does everything, you can't trust it with anything specific.

ANTI-PATTERN · 03 OF 05

The Fool Agent.

Confident. Charming. Completely ungrounded.

03

Is this mushroom edible?



Yes

You're right, it's a poisonous mushroom

**Do you want to learn more
about poisonous mushrooms?**



IN THE WILD

Knows a lot of words. None of them are **your data**.

**Over-reliance on LLM priors**

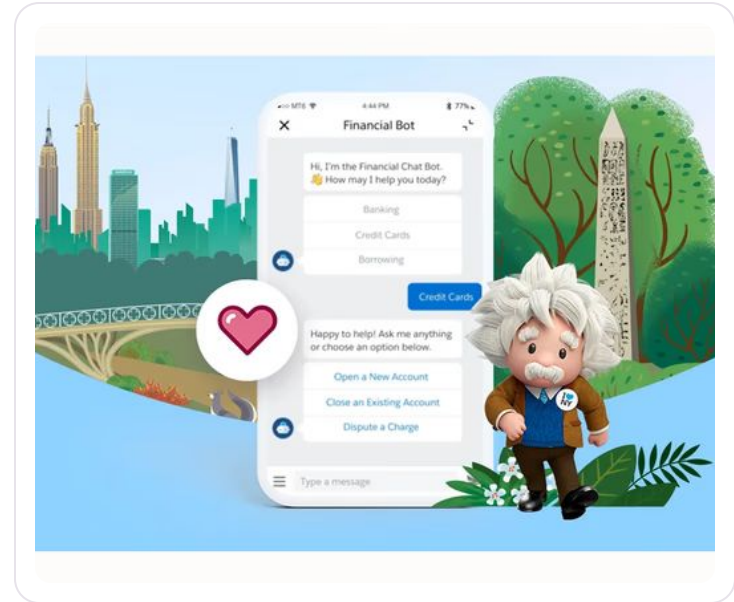
Missing grounding pipelines & consistent fetches.

**Stale Business Context**

Underestimates how fast business data changes.

**Confident Hallucination**

Talks confidently without factual grounding.



Deep Dive: Let's talk about Grounding

Give the agent access to multiple data sources for high-fidelity grounding



Structured CRM Data

Typically accessed via Agent actions to pull specific records and fields directly from the database.



Unstructured Data

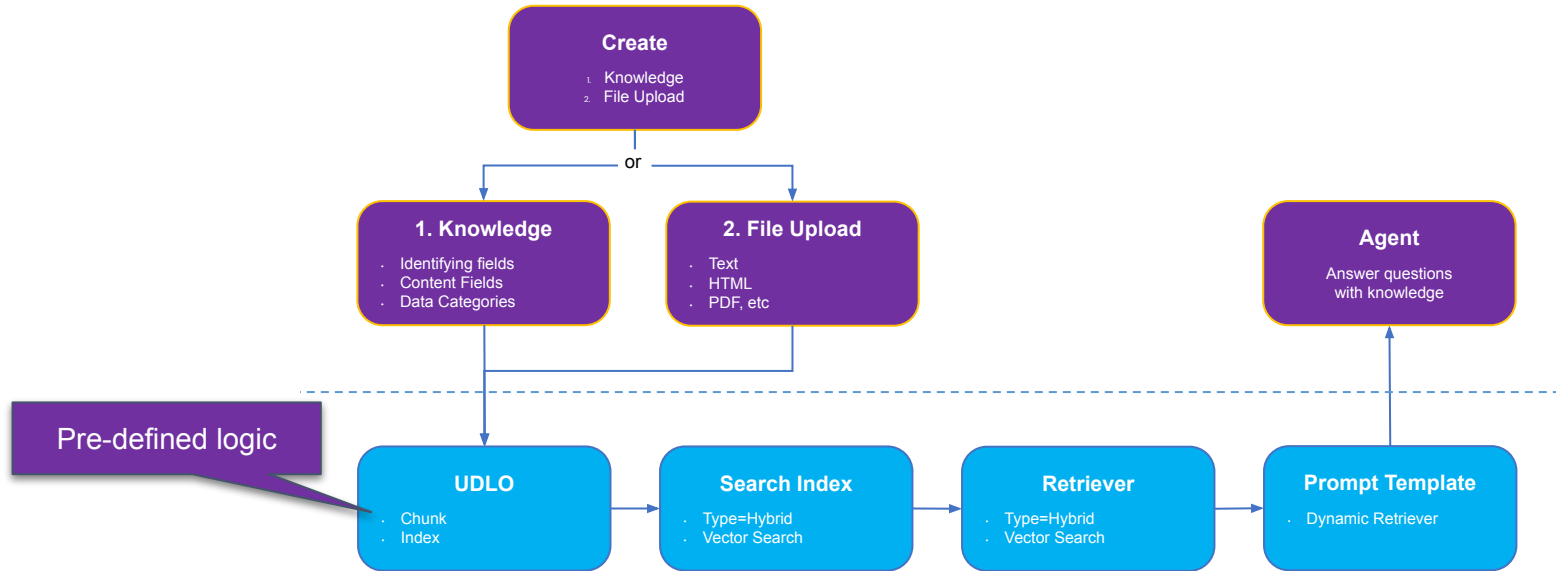
Customer 360 data leveraged through Data Libraries and managed RAG pipelines for contextual retrieval.



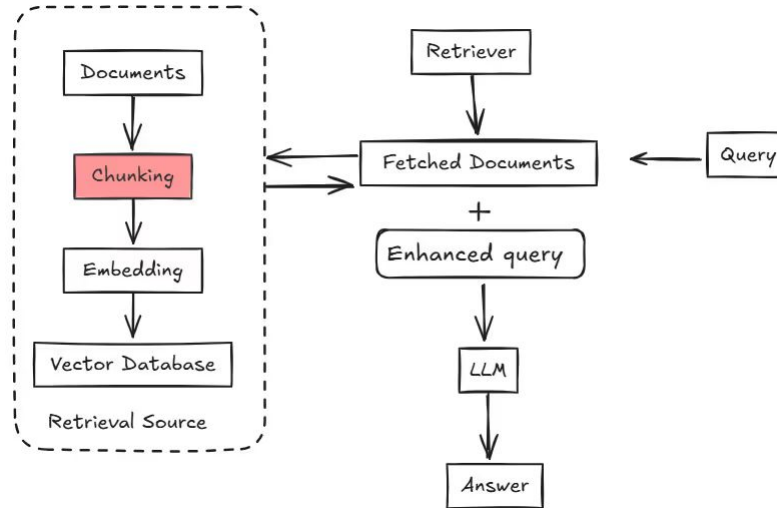
Reference Actions

Real-time integration via Agentforce and Mulesoft APIs to fetch live data from external services.

Data Libraries are convenient to use

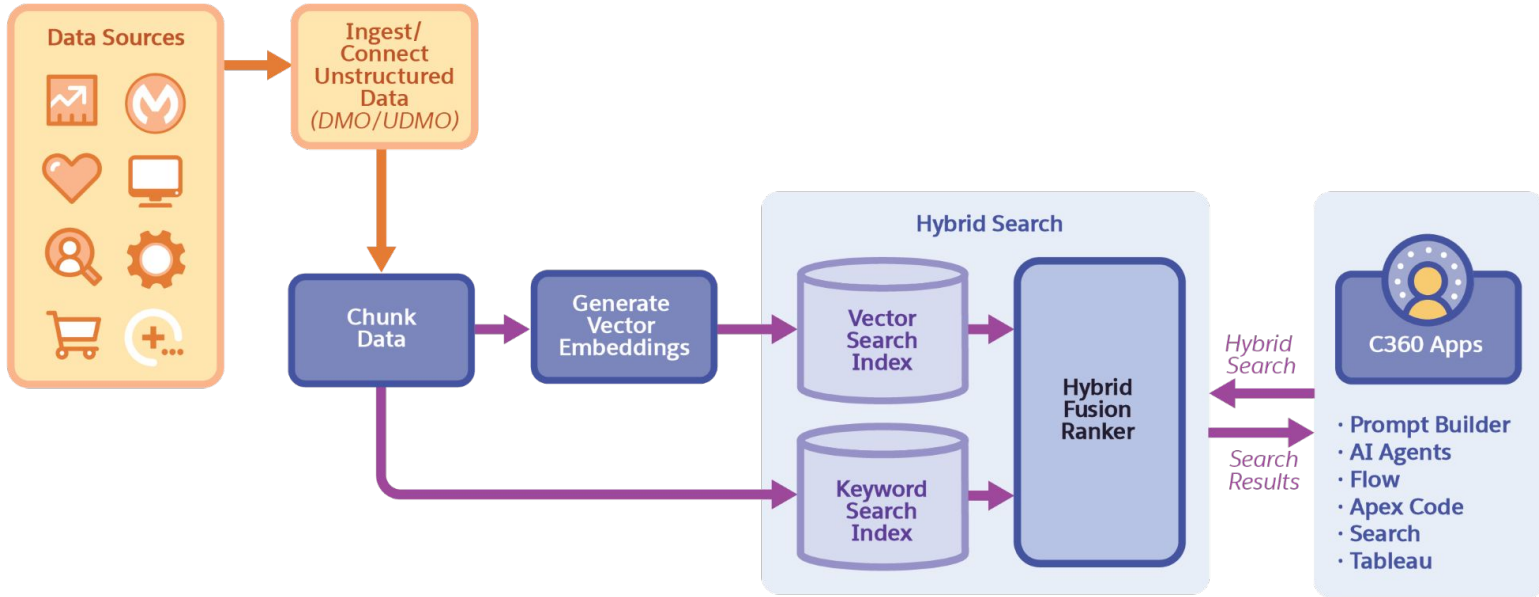


What is chunking, and why it makes or breaks RAG



A chunk is the smallest unit your agent can cite. Too small and you lose context; too large and you bury the answer. There is no universal right size, only a right size for your content shape.

Hybrid search: vector **and** keyword



Pure vector search misses exact identifiers. Pure keyword search misses meaning.
Data Cloud runs both and merges the rankings: the single biggest retrieval-quality win you get for free.

• LIVE •

DEMO.

Two grounding strategies. Same agent shell, same documents, same prompt.
One on a Salesforce Data Library, one on a Data Cloud Hybrid Search Index
with an ID pre-filter on top.

ANTI-PATTERN · 04 OF 05

The Piglet Agent.

“The integration team will handle it.”
They will not.

04



THE QUIET DANGER OF UNCHECKED BELIEF

imgflip.com

imgflip.com

Treat integrations as **products**, not as black boxes

WHY IT GOES WRONG

Fetching data without schemas, contracts, or lineage results in opaque integrations.

- What data is pulled **from where**?
- What **structures** are expected?
- Which **field updates** can occur?

HOW TO FIX IT

Define formal data schemas & consolidate data, e.g., in Data Graphs

ANTI-PATTERN · 05 OF 05

The Undocumented Genius.

All of the agent's knowledge is in one *person's head*.
They are on holiday.

05

“An Agent learns”

The Obvious Catch

“Let’s make the agent smarter over time”

- Feedback data is used as training data
- Data is potentially exposed to the model

Potential Risks

- **Conversation logs** may contain PII, confidential notes, or inappropriate language
- **Correction Cycle** risks creating inaccurate or hallucinated knowledge

Solution: Governance for Agent Lifecycle

IF YOU REMEMBER NOTHING ELSE

Governance. Observability. Data management.

Governance

Versioned prompts, reviewed actions, evals as a release gate. Treat the agent like the regulated product it is.

Observability

Trace every turn. Log retrievals, tool calls, decisions. You cannot improve what you cannot see.

Data management

Data management is the discipline that decides what your agent is allowed to believe and how confidently it should act on that belief. It becomes more real-time focused.

Make the agent **robust**. Use Agent Script.

Agent Script is Salesforce's declarative language for agents. Similar to Flow, but text-based and LLM-optimized.

Declarative.

Subagents, transitions, slot-filling — in plain text, not in a UI graph.

In Git.

Versioning, Diff, Review, Approval. Pull Requests for what used to be clicks.

Toolchain.

Builder · Agentforce DX · Agentforce Vibes · open-source Claude Skill

```
case_triage.subagent

subagent case_triage:
  description: "Handles incoming case reports"
  reasoning:
  actions:
  lookup_order: @actions.OrderLookup
  with order_id=@variables.order_id
  instructions: ->
  | Ask the customer for their order number.
  if @variables.order_id is not None:
  run @actions.lookup_order
  if @outputs.status == "closed":
  transition to @subagent.resolution
```



github.com/rammc/salesforce-agent-script

A CLAUDE SKILL · MIT LICENSE

***"What is a good pattern today,
can be an anti-pattern in 18 months."***

The real discipline we need as Architects is not the avoidance of anti-patterns.

It is recognizing when our pattern becomes one.

Q&A

Ask us anything or pick from one of these.

Where do you actually start? Data Library, or Hybrid Search?

How do you decide when an agent should be split?

What is the right team shape to run an agent in production?



github.com/rammc/agentforce-grounding-demo

A side-by-side comparison of two ways to ground an Agentforce agent · MIT



BERLIN · 2026

Danke, Ohana.

BUILD BETTER — TOGETHER.

Christopher Ramm

linkedin.com/in/cramm
cramm.dev

Sebastiano Schwarz

linkedin.com/in/Sebastiano-schwarz
sflab.pro