

Agentforce Meeting Concierge — Setup Guide

Standing up the shipped wealth experience, plus how to set up if you run the FSC managed package

What you'll have when you're done

Meeting Concierge is a shipped **Agentforce for Financial Services** (formerly Financial Services Cloud / FSC) capability that carries a wealth advisor through the full client-meeting lifecycle:

- **Pre-Meeting** — an AI Client Summary, a prep-task checklist, and a Preparation Brief, generated from the client's CRM data.
- **In-Session** — a discussion guide, talk-to-text or typed notes, and document summarization.
- **Post-Meeting** — an auto-generated follow-up summary, extracted action items, a draft follow-up email, and one-click CRM updates.

After this guide you will have:

- The feature **turned on**.
- The two shipped playbooks — **Annual Review** and **Prospecting** — cloned and **active**.
- The trigger flow that **auto-creates a meeting playbook** activated.
- Your advisors **permissioned**.
- Critically, the permission that **prevents silent Flex-Credit consumption** in place.

Creating a calendar **Event** for a client will then automatically generate a **Meeting Playbook**.

Time: budget roughly **1–2 hours** for a first working instance — about **30–45 minutes** of guided setup clickwork, and **15–20 minutes** to run one meeting end to end and confirm it works. The value here is **de-risking, not raw speed**: a handful of steps are easy to miss and cost silent hours, and this guide calls each of them out.

This is shipped product, configured — not custom code. Meeting Concierge ships purpose-built for wealth. You are enabling and configuring it correctly, not building something new.

Before you begin — prerequisites

These are hard gates. Confirm all of them before you start.

1. **Agentforce for Financial Services** (fka FSC) is enabled in your org.
 2. **Licensing:** Agentforce for Financial Services (**PUPM**) plus an **Einstein for Financial Services** license. Agentforce 1 for FSC bundles these. Pre-2025 orgs may need the Einstein-for-Financial-Services add-on and permission-set license added through licensing/provisioning.
 3. **Person Accounts are enabled** — wealth clients are modeled as Person Accounts, and the shipped trigger flow routes client Accounts through a Person-Account check.
 4. **Data Cloud is NOT required** for Meeting Concierge.
 5. A **lock icon** next to Meeting Concierge in Salesforce Go means **licensing is missing or wrong** — the fix is almost always a missing permission-set license, not a missing feature. Resolve that first.
-

Which path applies to you — read this first

Meeting Concierge ships configured for the **standard-objects** data model (Banking-on-Core): its AI grounding flows read the standard **FinancialAccount**, **FinancialAccountBalance**, and related objects.

If your financial-account data instead lives in the **FSC managed package** (objects prefixed **FinServ__**, e.g. **FinServ__FinancialAccount__c**), the stock playbook shows an **"Enable required objects"** warning and the Client Summary cannot ground on your clients' accounts.

Decide which you are before configuring:

Your data model	Follow
Standard objects (Banking-on-Core) — client accounts in FinancialAccount	Part A
FSC managed package — client accounts in FinServ__FinancialAccount__c	Part A Steps 1, 2, and 5 for common enablement, then Part B in place of Steps 3–4

 **If you are on the managed package, do NOT "fix" the warning by enabling the standard Financial Account data model.** That provisions *empty* standard

objects while your real data stays in **FinServ__**, so the AI grounds on nothing. Part B installs the correct adaptation instead.

Part A — Baseline setup (standard-objects data model)

Step 1 — Turn on Meeting Concierge

1. Setup → **Salesforce Go** → search **Meeting Concierge** → open it.
2. Click **Get Started** → **Turn On Einstein Generative AI**.

Expected result: the "Turn on Meeting Concierge" card shows **On**. Confirm that Setup → *Einstein Setup* and Setup → *Agentforce Agents* are both toggled on.

Step 2 — Give your advisors access

1. On the Salesforce Go page, under **Complete the Required Steps** → **Manage User Access**, click **Manage**.
2. Assign **Einstein for Financial Services** to your demo/advisor user(s).

Expected result: "N users assigned to Einstein for Financial Services."

Step 3 — Clone the two shipped playbooks

The setup page shows **Annual Review** and **Prospecting** shipped templates (tagged *FSC WEALTH*). These are **templates**, not playbooks — clone each one to create an active playbook.

1. On the Salesforce Go page → **Create Meeting Playbooks** → **Go to Setup** (lands on the Agentforce Meeting Concierge Setup page).
2. Click **New Playbook** → select **Annual Review** → **Next**.
3. **Define:** keep the shipped name and developer name — **Annual Review** / **Annual_Review**. Add a description. **Save & Next**.
4. **Configure:** accept the pre-filled defaults on every card (Client Summary; Meeting Prep Tasks; Documents; Meeting Prep Brief; Attendees). Click **Save & Next** through Pre-Meeting, In-Meeting Session, and Post-Meeting Follow-Up.
5. **Assign & Activate:** search **Einstein**, check **Einstein for Financial Services**, leave **Active playbook = Enabled**, and **Save**.
6. **Repeat Steps 2–5 for Prospecting** (name **Prospecting** / **Prospecting**).

Expected result: each playbook appears at the top of the list with a green **Active** badge.

⚠ **Keep the stock developer names.** The trigger flow in Step 4 finds playbooks by developer name (`Annual_Review` / `Prospecting`). If you rename them, you must also edit the flow's default values. For a clean baseline, keep them stock.

Note: if you mis-name a playbook and restart, an abandoned attempt still leaves an *inactive* definition behind. Your admin can safely delete a stray definition (it will show `IsActive = false`).

Step 4 — Clone and activate the trigger flow (mandatory)

🔴 **Nothing auto-creates a Meeting Playbook until you activate a clone of this flow.**

1. Setup → **Flows** → open **Meeting Concierge: Create Meeting from Event**. (*This is the shipped flow's exact label. Some documentation abbreviates it "Create Meeting Event" — the org label is authoritative.*)
2. Click **Save As** → **A New Flow**:
 - **Label:** `FINS Meeting Concierge: Create Meeting from Event`
 - **Flow API Name:**
`FINS_Meeting_Concierge_Create_Meeting_from_Event` (*a prefix that marks this as your customization*)
3. Because your playbooks kept their stock developer names, do **not** change the `AnnualReviewTemplate` / `ProspectingTemplate` default values — they already match.
4. **Save** → **Activate**.

Expected result: the flow shows as **Active**.

Step 5 — Prevent Flex-Credit burn (the #1 most-missed step)

This is a **setup/education gap, not a product bug** — and it is the single most-skipped step. Meeting Concierge and other **user-initiated** Agentforce actions are **unmetered** under Agentforce / A4FS (PUPM) licensing — but that coverage only engages when the advisor has **both** pieces below. Miss the second and Meeting Concierge **silently consumes Flex Credits** even though the org looks correctly licensed.

Do all three:

1. **Assign the "Unmetered User Based AI" permission-set license** to each advisor. Setup → **Users** → the user → **Permission Set License Assignments** → **Edit Assignments** → check **Unmetered User Based AI** → **Save**. (*If this PSL isn't available on your org, add it via licensing/provisioning first, then assign it.*)
2. **Create a Permission Set that grants the "Unmetered User Based AI" system permission.** Setup → **Permission Sets** → **New** → label it (e.g.

FINS_Meeting_Concierge_Unmetered_AI) → **Save** → **System Permissions** → **Edit** → check **Unmetered User Based AI** → **Save**.

3. **Assign that permission set** to each advisor. Setup → **Users** → the user → **Permission Set Assignments** → **Edit Assignments** → add the perm set → **Save**.

Verify: run one Meeting Concierge generation (e.g. a Client Summary) and confirm Flex-Credit consumption stays **flat** in Setup → **Digital Wallet**. That flat line is your proof the fix took.

Note: this covers *user-initiated* Meeting Concierge only. Autonomous or background agents (no human in the loop) still consume Flex Credits by design. Reference: help.salesforce.com — "*Configure User Permissions for Unmetered Licenses*."

Step 6 — Add Document Type picklist values (recommended)

The **Document Type** dropdown in the "Edit Document" modal ships **empty** — you add the values. This classifier tells the summarizer *what kind* of document each upload is (tax vs. statement vs. plan), so it extracts the right facts.

- The field is **Document Type** on the **Meeting Playbook Stage Content Document** object.
- On a fresh org this object isn't provisioned until the **first document is uploaded** to a playbook. Upload one file first, then configure the field.
- Setup → **Object Manager** → **Meeting Playbook Stage Content Document** → **Fields & Relationships** → **Document Type** → **Values** → **New**. Suggested values: *Account Statement, Tax Document, Financial Plan, Bank Statement, Correspondence, Other*.

Step 7 — (Optional) advanced enablement

- **Einstein Activity Capture** (calendar/email sync): Salesforce Go page → *Turn On Einstein Activity Capture* → connect Gmail/Outlook.
- **Slack channel component** on the playbook: optional; skip unless you're demoing it.
- **In-Person Meeting Assistance** (Sales Cloud mobile + Einstein Conversation Insights): requires an additional flow edit to pull the video-call transcript into the post-meeting summary. Advanced; defer unless needed.

Baseline verification

An admin can confirm:

- **Two active playbooks:** **Annual_Review** and **Prospecting**, both Active.
- **Four action-plan templates** (Annual Review + Prospect, each Pre/Post) published.
- The **FINS_ trigger flow** present and active.

➡ Setup complete — jump to **Prove it works** below.

Part B — If you run the FSC managed package (**FinServ__** objects)

Follow **Part A Steps 1, 2, and 5** exactly as written (turn on the feature, assign access, and apply the Flex-Credit fix). The **only** difference is Steps 3–4: instead of cloning the stock playbooks and activating the stock trigger flow, you install Salesforce's **published managed-package adaptation** — the same flows already remapped to the **FinServ__** objects, plus a managed-package playbook (**Annual_Review_FSC_Package**).

Why: the stock flows read standard objects. On a **FinServ__** org, the fix is Salesforce's published adaptation — **not** enabling the empty standard objects.

Published repo (public, Apache-2.0):

<https://github.com/salesforce/next-gen-wealth> (Redirects to [salesforce/financial-services-agentic-advisor](https://github.com/salesforce/financial-services-agentic-advisor) — same repo. Meeting Concierge needs the **PackageResources** folder **only**. Do NOT deploy the **DataKit** folder — that is a separate Data Cloud feature Meeting Concierge does not use.)

This is NOT installing a new managed package. It deploys metadata (flows, prompt templates, a playbook definition) that references your *existing* managed-package objects. The field mappings (**FinancialAccount** → **FinServ__FinancialAccount__c**, **FinancialGoalParty** → **FinServ__FinancialGoal__c**, **PersonLifeEvent** → **FinServ__LifeEvent__c**, ...) are already baked into the published flows — no hand-remapping needed.

Step MP-1 — Install the adaptation (choose one method)

Both methods deploy the **same** metadata. Deploy **PackageResources only** — never the DataKit.

Method B — Salesforce CLI (if your admin uses the CLI):

None

```
git clone https://github.com/salesforce/next-gen-wealth.git
cd next-gen-wealth/AgenticAdvisor
sf project deploy start --metadata-dir PackageResources -o <YOUR_ORG_ALIAS>
--ignore-warnings --wait 30
```

Expected result: deploy status **Succeeded**.

Method C — Workbench (browser-based, no CLI preferred):

The repo doesn't ship a ready zip — a script generates one. This one step needs `python3` (your Salesforce team can run it and hand you the zip if you'd rather not use a terminal):

1. `git clone https://github.com/salesforce/next-gen-wealth.git`
2. `cd next-gen-wealth`
3. `python3 tools/CreateClone.py` → produces `PackageResourcesSinglePackage.zip` in your system temp directory (the script prints the path). You need only that zip.
4. Go to **Workbench** (workbench.developerforce.com) → **Login with Salesforce** as a System Administrator (choose **Production** or **Sandbox** to match your org).
5. **migration** → **Deploy** → **Choose File** = `PackageResourcesSinglePackage.zip` → check **Single Package** → **Next** → **Deploy**.

Expected result: the status table shows **Succeeded**. Do **not** upload the `DataKit` zip.

Step MP-2 — Deploy and publish the Action Plan Templates

The action-plan templates aren't part of the deployed package.

1. Setup → **Action Plan Template Settings** → create/deploy and **Publish**:
 - **Annual Review Pre Meeting Template**
 - **Annual Review Post Meeting Template**

Expected result: both show published (Status **Final**).

Step MP-3 — Activate the managed-package playbook

1. Setup → **Agentforce Meeting Concierge** → edit `Annual_Review_FSC_Package` (installed in MP-1; it ships **inactive**).
2. Walk the wizard, assigning the action-plan templates (Pre → *Meeting Preparation Tasks*; Post → *Meeting Follow-Up Tasks*), then **Activate**.

Expected result: the playbook is **Active** and the "Enable required objects" warning is gone — it now reads your `FinServ__` objects.

Step MP-4 — Clone and activate the trigger flow, pointed at the FSC-Package playbook

 Nothing auto-creates a playbook without an active trigger flow, and it routes by **developer name**.

1. Setup → **Flows** → open **Meeting Concierge: Create Meeting from Event** → **Save As** → **A New Flow** (use a **FINS_** prefix, e.g. **FINS_Meeting_Concierge_Create_Meeting_from_Event**).
2. Because the managed-package playbook is **Annual_Review_FSC_Package** (**not** the stock **Annual_Review**), set the flow's **AnnualReviewTemplate** default value to **Annual_Review_FSC_Package**.
3. **Save** → **Activate**.

Expected result: creating an Event for a client auto-creates a Meeting Playbook using the FSC-Package playbook. *(A developer-name mismatch here = the flow runs but no playbook is created — the #1 silent failure. Double-check it.)*

Step MP-5 — Prevent Flex-Credit burn

Identical to **Part A Step 5**. Do it before your pilot widens, or Meeting Concierge silently burns Flex Credits.

Managed-package troubleshooting

Symptom	Cause	Fix
"Enable required objects" warning persists	Using the stock Annual_Review playbook, not the FSC-Package one	Activate Annual_Review_FSC_Package (MP-3) and point the flow at it (MP-4)
Client Summary generates but shows no accounts	Test client has no FinServ__FinancialAccount__c records, or the deploy didn't land	Confirm MP-1 Succeeded and the client has account records
No playbook created on Event	Trigger flow inactive or wrong developer name	Confirm the flow is Active and AnnualReviewTemplate = Annual_Review_FSC_Package

Symptom	Cause	Fix
Deploy errors on a playbook component	Known snag deploying the package set	Re-run the deploy; if it persists, deploy the PackageResources sub-components incrementally

Prove it works — run one meeting end to end

Step 0 — Create the meeting (this triggers the playbook)

The playbook is born from a **calendar Event**, not created by hand.

1. **App Launcher** → **Calendar** (or the **+ New Event** global action).
2. **New Event:**
 - **Subject:** e.g. **Annual Review – [Client Name]**
 - **Related To:** the client **Account** (a Person Account). *This field routes the flow.*
Set a **Lead** here instead to route to the Prospecting playbook.
 - Add a time and **Save**.
3. The active trigger flow fires and creates a **Meeting Playbook** (Status: **In Prep**).

Open it: App Launcher → **Agentforce Meeting Concierge** → **Meeting Playbook** tab → open the record. The page ships pre-wired with three tabs: **Pre Meeting / In Session / Post Meeting**.

Managed-package: choose a client that has **FinServ__FinancialAccount__c** records, and confirm the Client Summary shows their **real accounts** — that's the proof the remapped flows are wired correctly.

Step 1 — Pre-Meeting

- **Client Summary** — auto-runs on first open. Produces a real brief from CRM: assets under management, goals, financial context, and open service cases. Click **Refresh** to regenerate.
- **Meeting Preparation Tasks** — four tasks instantiate from the Pre Meeting template with staggered due dates; completing them advances the progress bar.
- **Documents** — upload client docs; flag **Summarize / Include in Prep Brief / Client-Facing**.
- **Meeting Preparation Brief** — an AI agenda that pulls in a "Since Last Meeting" section from prior Interaction Summaries. (*Asking Agentforce to refine rewrites the entire brief.*)

Step 2 — In-Session

1. Click the **In Session** tab (Status → In Session).
2. **Generate Discussion Guide** → structured talking points.
3. **Capture notes** (at least one): **Draft Meeting Notes** (type), **Record Meeting Notes** (talk-to-text — *no audio or transcript is stored, only a summary*), or **Upload Files**.
4. Click **Generate Meeting Follow Up**.

Step 3 — Post-Meeting

1. Click the **Post Meeting** tab. The **Meeting Follow-Up Summary** generates with sections: **Summary · Decisions & Outcomes · Action Items (Owner/Action/Due) · Salesforce Updates · Sentiment**.
2. A banner says "Your draft is ready. Review and save it to keep your changes." → **Review / Edit** → **Save**. 🗝️ **Saving is the gate that unlocks the Agentforce Actions card.**
3. **Meeting Follow-Up Tasks** — a 3-item action plan instantiates as Tasks.
4. **Agentforce Actions** (click-to-run — nothing writes to CRM automatically): **Create or Update Goal, Create Tasks, Generate Follow-up Email** (drafts text to copy/paste — OOTB does not send), **Create Life Event, Update Household**. The advisor stays in control of every write.

Good to know — common questions

- **Where do the AI outputs live?** In the `MeetingPlaybookStageText` object — versioned, one row per card (Client Summary, Prep Brief, Follow-Up Summary). This is the system of record, not a custom object per card.
 - **Does Meeting Concierge replace our Interaction Summary process?** **No.** It *reads* prior Interaction Summaries for context and stores its own outputs in `MeetingPlaybookStageText`. It does **not** auto-write an Interaction Summary — writing back is a configurable Agentforce Action you add, not default behavior.
 - **Does anything write to CRM automatically?** **No.** Every CRM update is a click-to-run Agentforce Action, and they unlock only after you Review and Save the follow-up summary.
 - **Call recording?** Talk-to-text only. **No audio or transcript is stored** — only the summary — which supports compliance needs.
-

Where to go next

- Add your own meeting types by authoring custom prompt and action-plan templates, then a new playbook.
- Tune the shipped wealth Agentforce Actions (Financial Goals, Life Event, Household) to your practice's workflow.
- Layer optional features: Einstein Activity Capture, a custom send-email action, In-Person Meeting Assistance.

All official references cited here are on help.salesforce.com. Meeting Concierge is shipped Agentforce for Financial Services product — you are configuring product, not installing custom code.