| **Step** | **Element** | **Description** |
| --- | --- | --- |
| **Entry Point** | **Client Meeting Prep → “Launch Note” Button** | Banker is reviewing the prep brief for an upcoming client call (e.g., ServiceNow M&A). They click **“Launch Note”**, which opens the contextual note-taking view in the split screen. |
| **1** | **Auto-Load Pre-Context** | Prep brief details (company profile, last note, deal status, prior valuation signals) are loaded in a **read-only left pane**. The **right pane** is a fresh note canvas. |
| **2** | **Live Note Capture** | User begins taking notes:• Typed manually• Drag-and-drop transcript• Pull from meeting platform (Zoom, Teams) |
| **3** | **Context-Aware Enhancements** | As banker types, smart prompts emerge:• “Tag valuation anchor?”• “Track sentiment shift?”• “Flag buyer constraint?”Banker controls AI suggestions. |
| **4** | **Smart Summary + Timeline Anchor** | After call ends, the app auto-generates:• Key Points• Sentiment Tracker• Timeline updates• Context tags (e.g., “Valuation moved from 2.2x → 1.9x”) |
| **5** | **Threaded Integration** | Note links back to:• Client’s deal timeline• CRM record• Prior meetings |
| **6** | **Next Action Prompt** | “Prep next meeting?” → Brings forward relevant threads and suggests new objectives. |

### Step 1: AutoLoad Pre-context – Detailed Scenarios

#### Scenario A: User launches the note *before* the call starts

* **Use Case**: The banker opens the note canvas 5–10 minutes before the meeting begins.
* **Behavior**:  
  + The system loads the **prep context** and applies a **template** based on the meeting type (e.g., “Origination”).
  + Note canvas is in **edit mode**, with no live transcription.
  + A passive indicator appears: *“Meeting starts in 7 minutes. Transcription will activate when call begins.”*
  + User can pre-fill: agenda items, talking points, expectations.

✅ This mode supports proactive bankers who plan ahead and want to sketch talking points before the discussion starts.

#### Scenario B: User launches the note *after* the call has started

* **Use Case**: The call is already in progress; user clicks "Launch Note."
* **Behavior**:  
  + System detects the active Zoom/Teams/Webex meeting.
  + Loads meeting prep context + current call metadata.
  + **Live transcription begins immediately**, timestamped to call start.
  + The note canvas shows a visual indicator: *“Capturing from 2:07 PM — synced with transcript.”*

✅ This is for users who delay opening notes but still want full capture and context alignment.

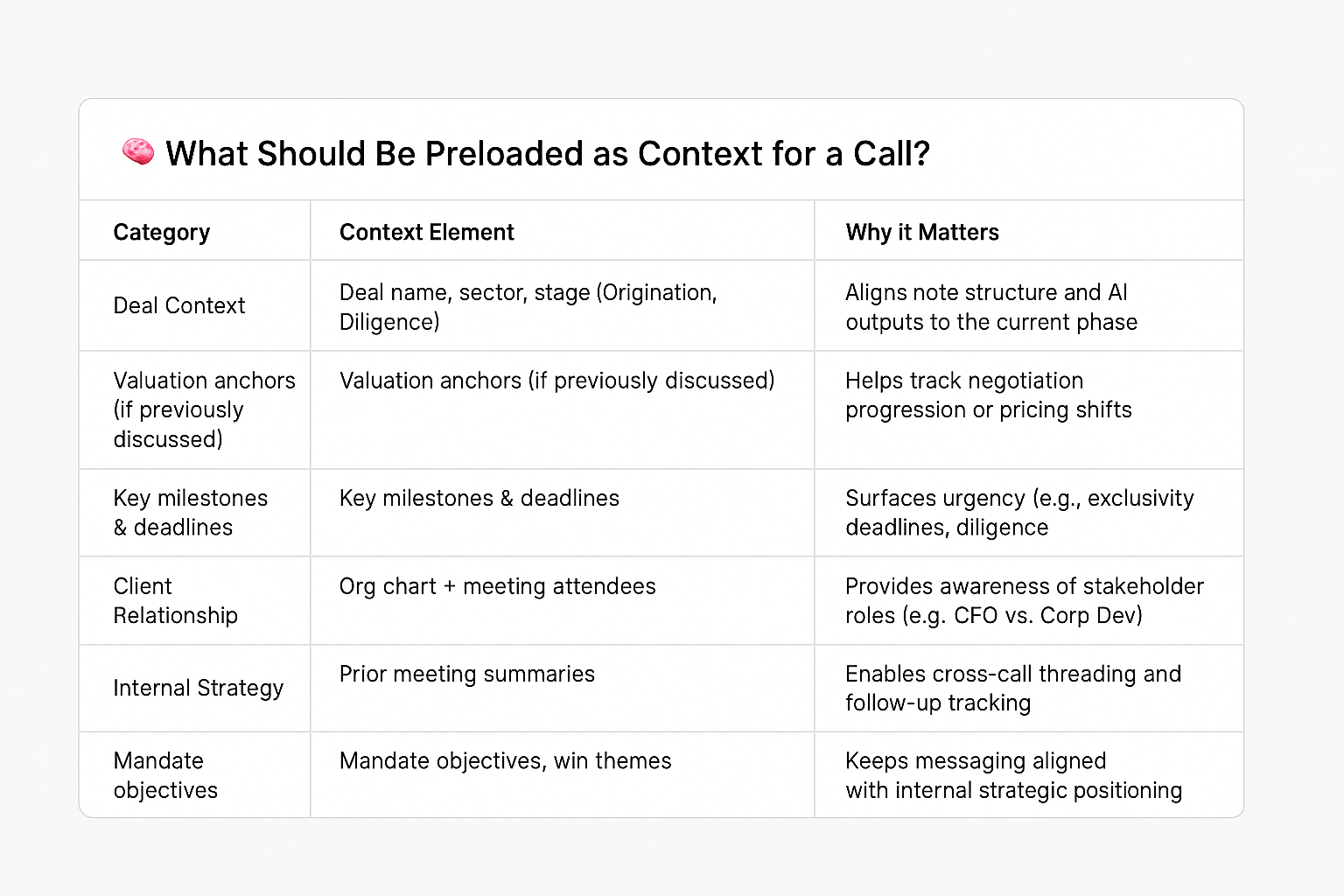
#### Scenario C: Call begins, but user hasn’t launched notes yet

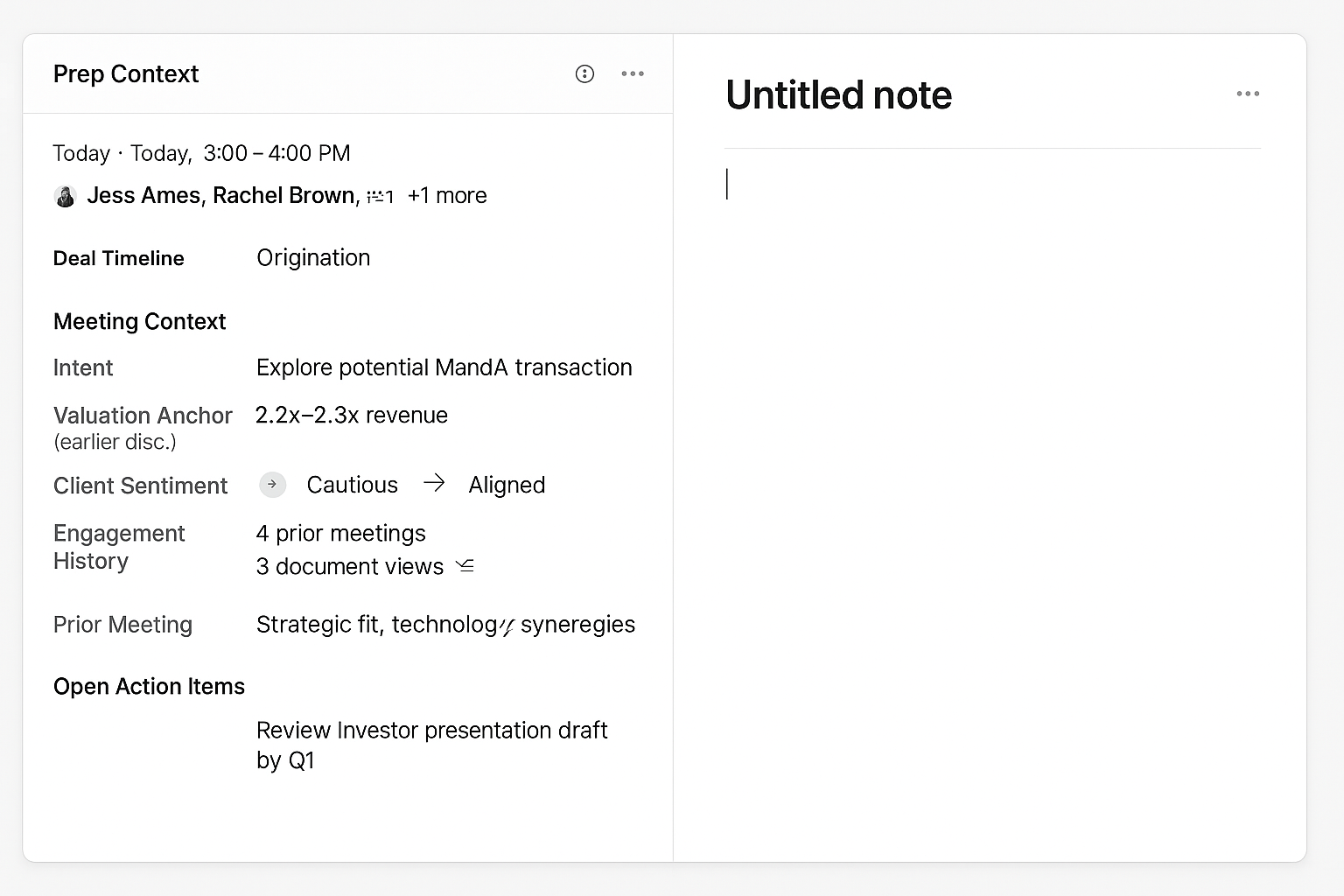
* **Use Case**: User forgets to launch note manually.
* **Behavior Options** (depending on product decision):  
  + **(a) Passive Transcription Mode**: System auto-captures the transcript silently in the background.  
    - Later prompts: *“Missed the note? Click here to reconstruct from today’s call.”*
    - Gives option to retro-tag conversation flow into the appropriate structure.
  + **(b) No auto-capture**: Warns user post-call: *“No note was launched. Would you like to generate one from meeting transcript?”*

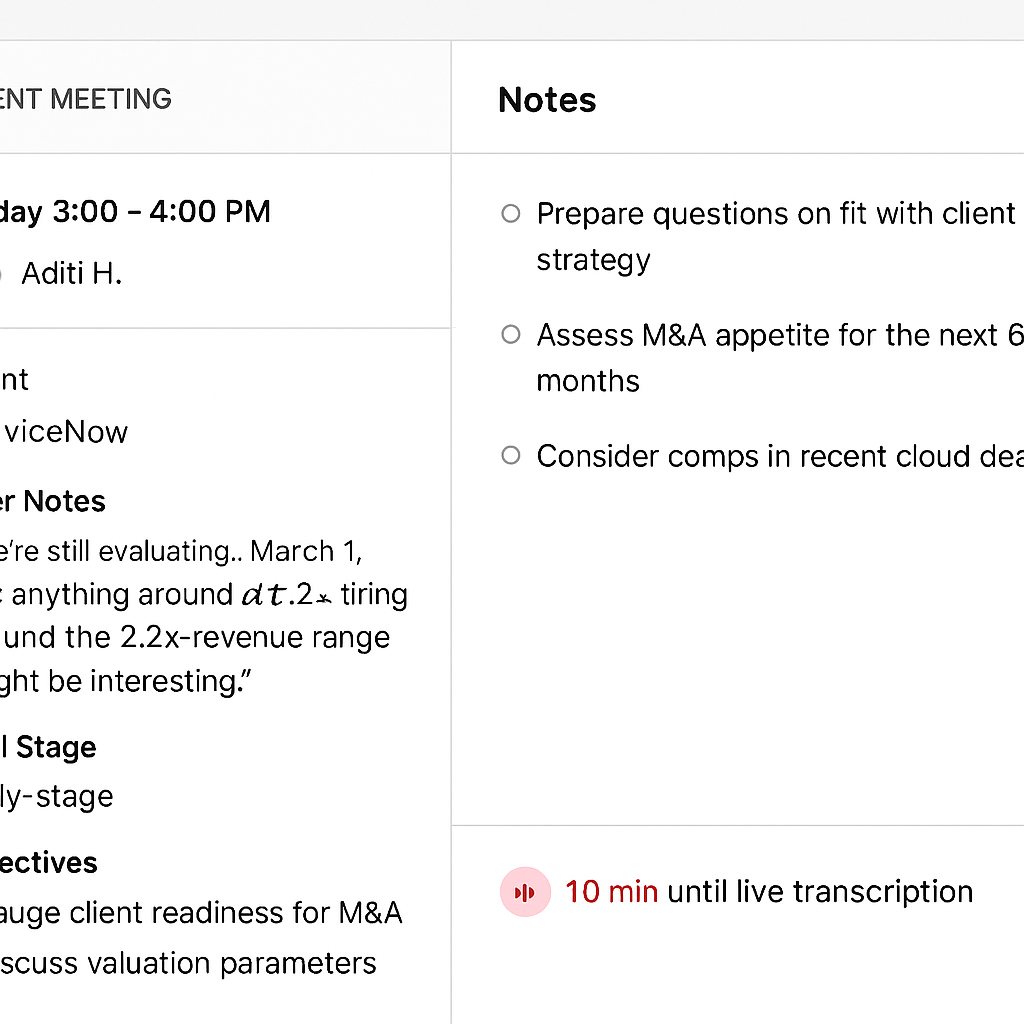
🔄 Trade-off between compliance/privacy defaults vs. usability — we recommend (a) for trusted enterprise clients with secure integrations.

### 🔄 Optional Enhancements

* “🧠 **Pre-fill Mode**”: Lets users jot thoughts pre-call. Becomes read-only once call starts and switches to live mode.
* “📡 **Live Sync Mode**”: Follows transcript in real-time and nudges user with tag suggestions mid-call.
* “🕒 **Missed Call Catch-up**”: If no note was launched, it still reconstructs from transcript afterward.







Let the user stay in the **Client Meeting Prep + Note** interface **as the Zoom meeting begins**, without jumping back to Outlook or Zoom manually — unless they want to.

## ✅ Ideal Live Note Capture Flow (Minimal Disruption)

### 📍 10–5 mins before meeting:

* Banker opens **Client Meeting Prep** tool inside CRM or platform.
* They click **“Launch Note”**, review context, maybe pre-fill talking points.

### 🚨 As the Zoom meeting starts:

* **One of two things happens**:  
  + **Option A: Embedded Zoom** If supported by IT, Zoom launches **in a split panel** on the right or bottom (like a minimized call widget).  
    - Note panel stays open
    - Banker never switches tabs
    - Audio/video captured for transcript
  + **Option B: External Zoom, Internal Sync** If Zoom runs separately (most common), the system:  
    - Auto-detects when Zoom starts (via calendar or Zoom API)
    - Triggers **transcription to start**
    - Keeps banker in the note canvas
    - Shows subtle banner: *“Recording from Zoom in progress”*

### 🟢 Live note capture view:

* Banker types in real time (or not)
* System overlays inline suggestions:  
  + “Capture Valuation Anchor?”
  + “Sentiment Shift Detected”
* Transcript appears in background or as a side tab

## 🔌 Supporting Integrations Required

| **Integration** | **Role** |
| --- | --- |
| **Zoom Calendar SDK or Webhook** | Detect meeting start, pull transcript, link metadata |
| **Outlook Add-in or OAuth** | Auto-join detection, launch prep + note view from calendar |
| **Optional: Zoom App SDK** | Embed Zoom UI inline if bank permits (nice to have) |

## Interaction Detail: Live Note Capture (During Meeting)

| **Time / Moment** | **What the Banker Does** | **System Behavior** | **Live Meeting Status Icon (UI)** |
| --- | --- | --- | --- |
| **10 min before** | Opens *Client Meeting Prep* → clicks **Launch Note** | Loads prep context in left panel + note canvas on rightShows meeting countdown | ⏳ Grey dot“Meeting starts in 10 minutes” |
| **2 min before** | (Optional) Reviews talking points / pre-fills note | Adds subtle CTA to open Zoom | ⏳ Yellow pulse“Meeting starting soon”🔘 Open Zoom |
| **30 sec before** | Opens Zoom via Outlook invite | Detects Zoom via metadata/calendar sync | 🔄 Detecting…“Matching call to prep session…” |
| **Call time** | Focused in Zoom (note app may stay open or minimized) | Confirms Zoom matchPreps for transcription start | 🔴 Red pulse“Zoom detected — transcription starts soon” |
| **Transcription begins** | — | Background transcription begins silentlyPrep thread is updated with call metadata | 🔴 Red waveform“Recording in progress” |
| **During call** | Optionally toggles to notes; types manually or reviews context | Insight Timeline Strip begins collecting real-time AI nudges (valuation, sentiment, urgency, etc.) | 🔴 Red icon stays activeWaveform pulses gently |
| **AI nudges detected** | — | Nudges appear in Insight Timeline:• Add to note• Edit• Dismiss | No popups — nudges appear in side timeline only |
| **After call** | Returns to review AI suggestions + finalize notes | Auto-generates summary, tracks sentiment deltas, offers “Next Action” prompt | ✅ Green checkmark“Meeting complete — review summary” |

## Context-Aware Enhancements (AI Nudges During Call)

AI listens passively and logs signals without showing the transcript. Nudges appear as a **timeline of AI insights**, not pop-ups.

### ✅ Unified Nudge Experience: Insight Timeline Strip

* 📌 **Location**: Fixed side panel on right
* 🔁 **Behavior**: Accumulates AI insights in real-time, with clear icons, timestamps, and action buttons
* ✍️ **User can**:  
  + Add an insight into the note with one click
  + Edit suggested text
  + Dismiss irrelevant insights

### ✨ Example Timeline Entries (during call):

pgsql

CopyEdit

🧠 [10:03 AM] Valuation anchor detected → 1.9x (↓ from 2.3x)

➕ Add to Notes ✏️ Edit ✖️ Dismiss

👤 [10:04 AM] Salman: “Client now pushing for exclusivity – use this for negotiation framing”

🧠 [10:06 AM] Sentiment shifted: exploratory → committed

➕ Add to Notes

## ✅ Design Principles

| **Risk** | **Resolution** |
| --- | --- |
| Too much noise from transcript | Transcript hidden by default; timeline entries are summarized signals |
| Confusion between AI vs human input | Visual labels, icons, and interactivity affordances differentiate clearly |
| Lost signals if not acted on | Timeline persists through and after call — banker can review async |

## Generated image

## hreaded Integration: Design Overview

### 🧠 Concept

Instead of treating each meeting as a one-off, the system **threads** notes, actions, and insights across touchpoints (prep → live → post) under a single, evolving “deal thread” — tied to a client, opportunity, or mandate.

## 📐 UX Design Elements

### 1. Thread Timeline View (Side Panel or Modal)

**Purpose:** Show progression across all meetings tied to the same deal or prospect.  
 **Includes:**

* Meeting dates + types (Origination, Kickoff, Diligence, etc.)
* Key signals per meeting (valuation shifts, sentiment moves, redlines)
* Tags and nudges surfaced
* User-written + AI-inferred summaries

**Example UI (Right-Side Drawer)**

yaml

CopyEdit

📁 Deal Thread: Project Redwood

🔹 Mar 1 – Origination Call

• Valuation Anchor: 2.3x

• Sentiment: Exploratory

• No mandate issued

🔹 Apr 15 – Negotiation Kickoff

• Valuation Revised: 1.9x

• Sentiment: Committed

• Exclusivity Asked: 10d

🔹 [Next Meeting Scheduled: Apr 30]

### 2. Backlinked Context Cards Inside Note Editor

When viewing or editing notes for a specific meeting, prior signals are accessible inline as context cards.

**Example:**

📌 *Previous Valuation Anchor: 2.3x from Origination Call – Mar 1* *Add to Note* | *Dismiss*

This enables **temporal threading** without the user needing to jump views.

### 3. Tag Anchoring + Timeline Filters

Each smart tag (e.g. #ValuationAnchor, #ExclusivityAsk) links back to where it first appeared and how it evolved.

**UI Features:**

* Hover on tag → see all prior mentions
* Click → jump to full threaded history
* Filters: “Show all #ValuationAnchor shifts”

### 4. CRM Integration Hook

Each thread syncs back to CRM as a dynamic record, not static text:

* **One Deal Thread per Opportunity**
* Updates pushed after each call (meeting summary, updated valuation, new risk flags)
* Threaded view visible in CRM as well (e.g. Salesforce plugin)

## 🎯 Why This Matters for IB

| **Problem** | **Threaded Integration Solves** |
| --- | --- |
| Fragmented call history | Timeline shows evolution across meetings |
| Lost institutional memory | Tagged insights persist across handoffs |
| Repeated context lookup | Prior data is surfaced just-in-time |
| Inconsistent CRM notes | System pushes consistent, structured updates |

Would you like a **Figma-style visual mockup** of the Thread Timeline panel or inline context cards?

##### You said:

does this step happen after the AI reconsytruction?

###### ChatGPT said:

Great question — here's the sequencing with clarity:

### ✅ Yes, Threaded Integration activates after AI Reconstruction, but it also builds over time.

#### 📍 Sequence of Events:

1. **Before/Live Call:**
   * Meeting is tied to a known deal/opportunity (via Client Meeting Prep).
   * AI is logging nudges and user is taking notes.
   * A thread ID is implicitly created or retrieved for the deal.
2. **🧠 After Call – AI Reconstruction Runs:**
   * Transcription → signal detection → structured summary.
   * AI merges:  
     + Human notes
     + Accepted/rejected nudges
     + Transcription-backed signals
   * Populates post-call **Negotiation Template** or **Origination Template** based on context.
3. **🧵 Threaded Integration Kicks In:**
   * AI-enhanced notes from this call are:  
     + Appended to the existing thread timeline (e.g., “Apr 15 – Negotiation Kickoff”).
     + Backlinked to past meetings (e.g., shows valuation dropped from 2.3x to 1.9x).
     + Stored as a **context-aware segment** inside the deal’s evolving profile.
4. **CRM Update & Frontend UX:**
   * Banker sees a scrollable or collapsible thread across all past meetings.
   * CRM gets synced version of the updated deal thread.

### 🔄 Summary

* **AI Reconstruction = Generate a structured artifact of a single meeting**
* **Threaded Integration = Evolve a narrative across meetings**

Threading is *what makes the system feel like a memory*, not just a transcript tool.

Would you like a timeline of how these pieces evolve over a full 3-meeting arc (e.g., Origination → Kickoff → Diligence)?

##### You said:

**Uploaded image**

so this thread should show up when the launches notes. previoulsy the ui you showed doesnt have that. its attached

###### ChatGPT said:

Exactly — great catch. The **thread context should absolutely be present when the banker launches notes**, especially for returning users or recurring calls. In the UI you've shared, we’re only seeing the **"Prep Context" for the current meeting**, but what’s missing is:

### 🧵 The “Thread History” Module

A collapsible section just below (or adjacent to) “Prep Context” that shows the deal's **cumulative intelligence across past interactions.**

### ✅ Updated UX Suggestion

| **Section** | **Content** |
| --- | --- |
| **Prep Context** (existing) | Today’s meeting metadata: participants, timeline, intent, sentiment, valuation, open action items |
| **🔁 Thread History (NEW)** | Past meetings + structured summary entries pulled from: |

* AI Reconstructions
* Banker-entered notes
* Key timeline tags (e.g., valuation, sentiment, exclusivity)  
   Option to expand/collapse |

### 🖥️ UI Layout Suggestion (In the Left Panel)

yaml

CopyEdit

📅 Meeting: April 15 – Origination

💬 Prep Context

• Intent: Explore MandA transaction

• Valuation Anchor: 2.2x–2.3x

• Sentiment: Cautious → Aligned

• Open Action: Review investor deck by Q1

🧵 Thread History (3 past entries) ▼

---------------------------------------

🔹 Mar 1 – Initial Coverage Intro

• “Too early for a mandate”

• Sentiment: Cold

• Banker Note: Focused on GTM scaling

🔹 Mar 15 – Strategy Deep Dive

• CEO open to options if tech buyer

• Anchor: 2.3x floated

🔹 Mar 30 – Mandate Alignment Call

• Legal + CFO alignment

• Action: Confirm draft timeline

### 💡 Design Notes

* Thread History should **not be passive history** — it's an active reference object that evolves as notes and AI signals compound.
* Smart tags (e.g., #ValuationShift, #MandateSignal) should be hyperlinked to original context
* Users should be able to click **“Insert into Today’s Note”** to carry forward insight

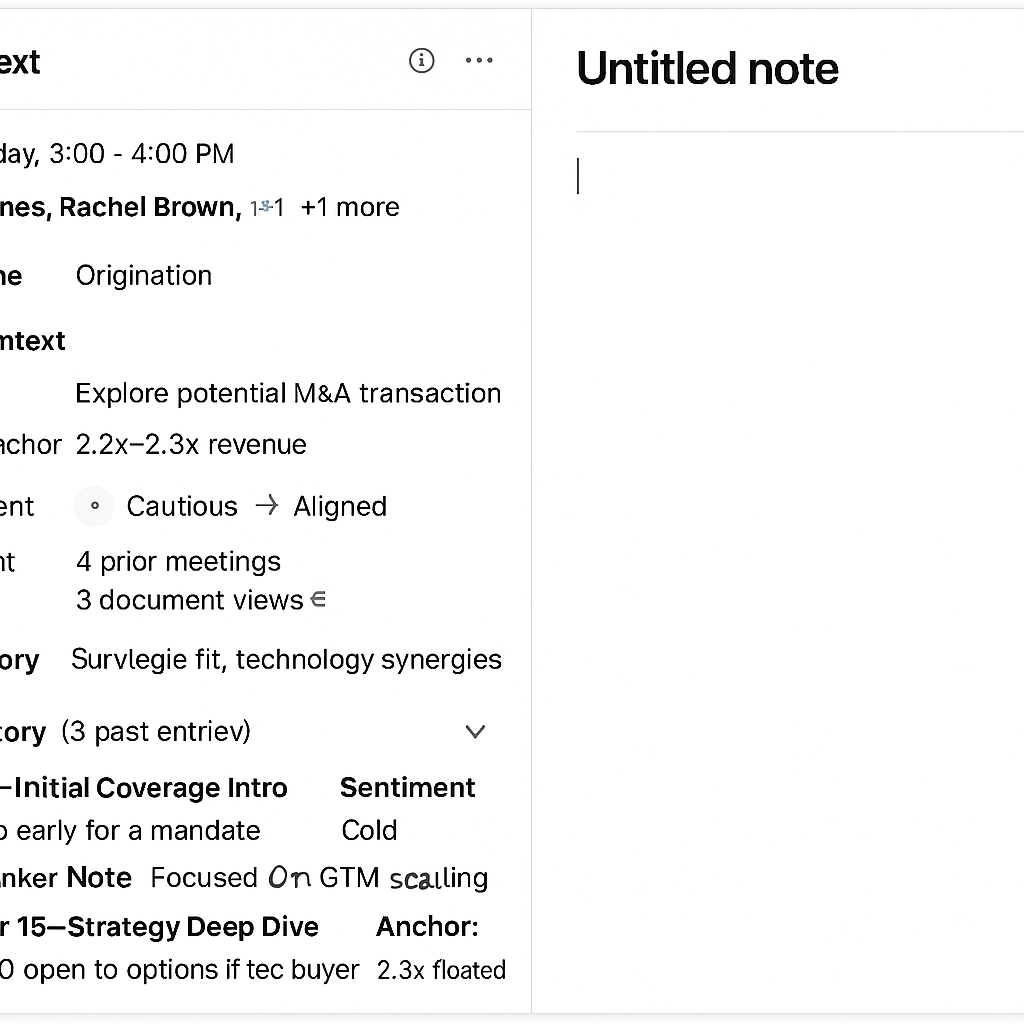
Would you like me to mock this up as a full left-panel Figma UI next to the current “Untitled Note” view?

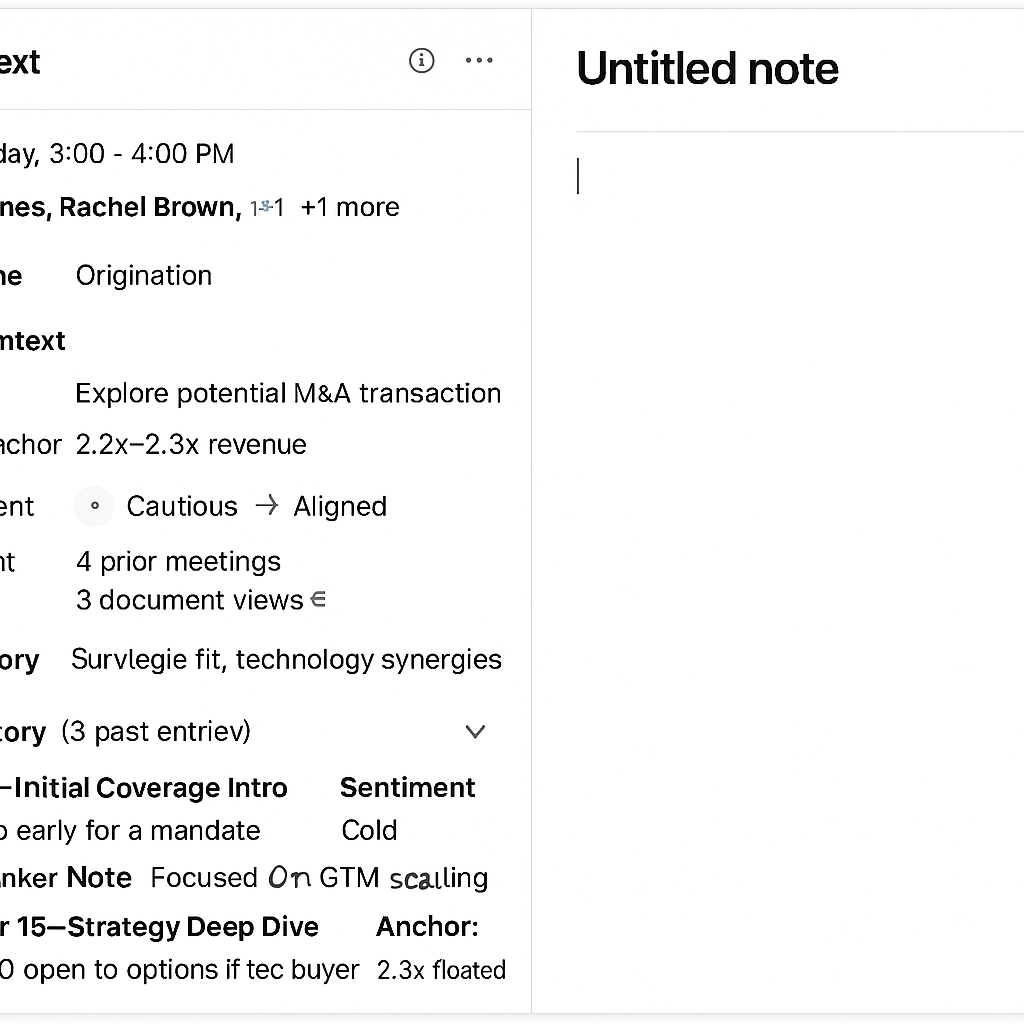
##### You said:

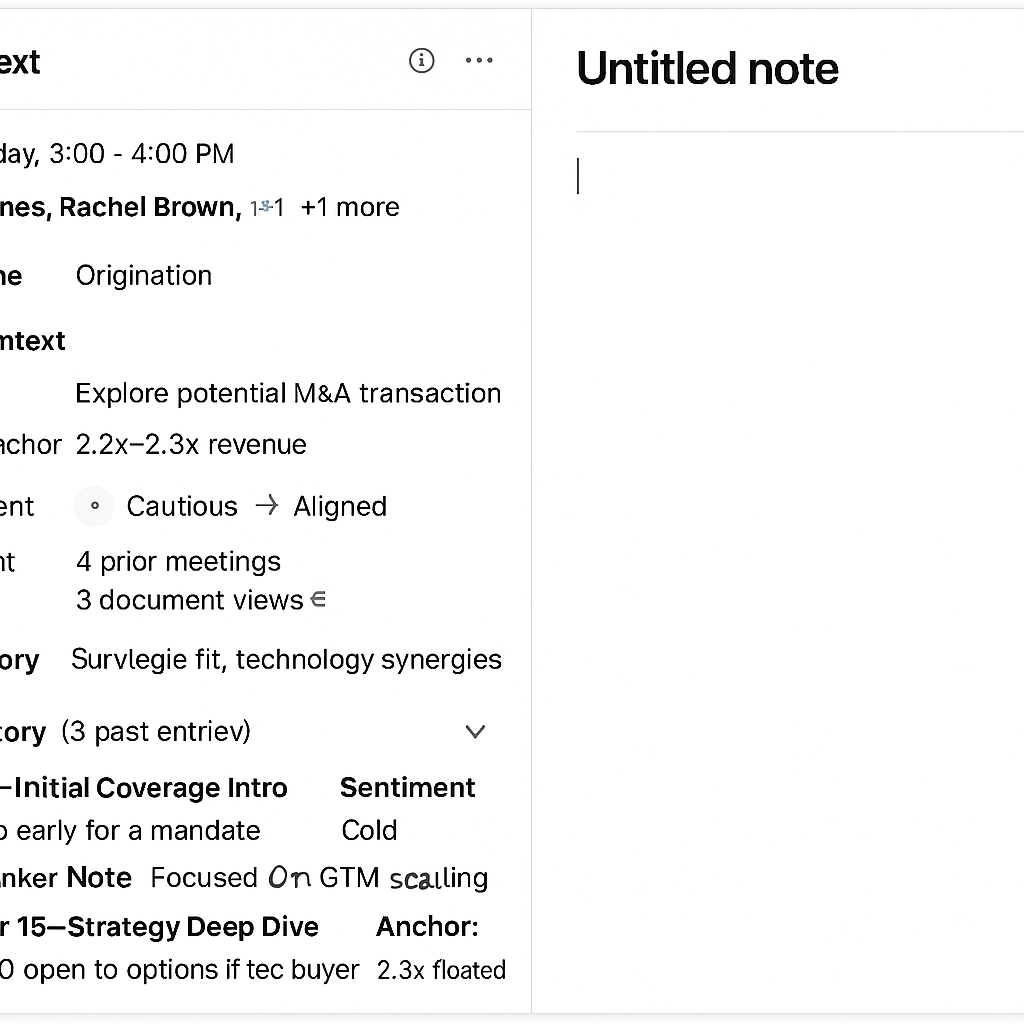
yes please

###### ChatGPT said:

Image created







##### You said:

now what will be the last experience ..Next action would look like

###### ChatGPT said:

Great — now that the banker has completed the call, reviewed the AI-enhanced notes, and seen the thread stitched together, the **final step** is to drive **momentum forward** with a focused **Next Action experience**.

## ✅ "Next Action" Experience – Post-Call Wrap-Up

### 🎯 Purpose:

Surface the *next logical step* in the deal lifecycle — based on meeting content, thread signals, and user behavior.

### 🧠 Inputs to Generate Next Action

| **Signal Source** | **Example Trigger** | **Implication** |
| --- | --- | --- |
| **Smart Summary Tags** | #ExclusivityAsk, #InternalAlignment | Suggest follow-up on exclusivity term sheet |
| **Open Action Items** | “Draft investor deck by Q1” remains unresolved | Re-surface task with updated deadline |
| **Thread Timeline** | Gap since last buyer-side interaction | Suggest warm reactivation |
| **User Notes** | "Client wants more clarity on valuation comps" | Suggest sending targeted materials |

## ✨ UX Design for “Next Action” Section

### 📍Placement:

Appears at the bottom of the note editor (or in a dedicated right-panel widget after the call ends).

### 🖥️ Example UI:

#### ✅ Suggested Next Actions

Based on today’s call and prior thread history:

* 📎 **Send revised exclusivity draft** to CFO by Friday
* 💡 **Prep valuation comps slide** on vertical SaaS carve-outs
* 📅 **Schedule internal sync** with MD before Monday EOD
* 🔄 **Log follow-up call** for May 1 – Negotiation Continuation

[➕ Add to CRM] [🗓 Create Calendar Event] [✏️ Edit]

## ✍️ Optional: Smart Compose for Follow-Up Email

If applicable, clicking a “Send Summary” or “Next Touch” could generate:

**Draft Email** *Hi Jess –  
 Thanks for the thoughtful discussion earlier today. As discussed, we’ll follow up with the exclusivity draft by Friday and prepare the valuation comps for internal alignment.* *Looking forward to next steps – happy to lock in time for May 1 to continue.*

### ✅ Why It Matters

| **Value** | **Description** |
| --- | --- |
| **Closes the loop** | Converts insights into motion — no drop-off post-meeting |
| **Saves time** | Banker doesn’t have to re-think next steps from scratch |
| **Improves handoffs** | Next user (MD, Analyst) sees structured guidance |
| **Boosts CRM hygiene** | Ensures next steps are logged and time-bound |

Would you like this rendered visually below the note canvas and thread timeline in the same Figma UI style?