ChatGPT

Interactive Shared 3D MindMapping on Quest 3: Landscape and Technical Possibilities

Interactive Shared 3D MindMapping refers to a co-located mixed reality experience where two or more users share an augmented space and can spawn relevant objects, media, or references during conversation. This could mean that as people discuss ideas, the system conjures 3D models, videos, articles, or even AI-generated content around them, creating a spatial "mind map" of the discussion. Building such an experience on the **Meta Quest 3** (a standalone MR headset) requires understanding the current technologies available and the design paradigms for collaborative AR. Below, we explore the landscape from several angles: available SDKs for shared AR on Quest 3, examples of multi-user MR apps and prototypes, tools for live object generation and media retrieval, conversational AI integration in AR, design patterns for 3D mindmapping, and the hardware constraints/affordances of Quest 3.

Platforms and SDKs for Co-Located AR on Quest 3

Developers targeting Quest 3 have a growing ecosystem of SDKs and APIs to enable **co-located** (same-room) shared AR experiences:

- **Meta Presence Platform (Shared Spatial Anchors):** Meta's own SDK provides core features for mixed reality on Quest. Notably, *Spatial Anchors* let apps pin virtual content to real-world locations, and the newer *Shared Spatial Anchors (SSA)* allow multiple headsets to **share a world-locked frame of reference** [1]. In practice, Shared Spatial Anchors enable local multiplayer AR – users in the same physical space see virtual objects anchored in identical real-world positions. This is the foundation for co-located AR on Quest. For example, two users can collaborate around a virtual model on a table, each seeing it in the same spot. Meta first introduced SSA support after 2022 and has continued improving the workflow. Recent updates (as of late 2024) added **Colocation Discovery** and **Group Sharing** APIs to streamline multi-user setup [2] [3]. Colocation Discovery uses Bluetooth to automatically detect nearby Quest devices (within ~30 feet) running the same app, so users can form a session without manual pairing [4]. The Group Sharing API lets developers share spatial anchors to a group ID (instead of individual user IDs) – meaning all users in a session get the anchors in one step, rather than the host sending anchors to each peer [3]. These features greatly reduce the overhead in syncing a shared coordinate system for co-located AR. Meta provides these capabilities through Unity, Unreal, and OpenXR integrations, along with sample code for developers [5]. In summary, Meta's Presence Platform on Quest 3 offers out-of-the-box solutions for **multi-user AR**: from establishing a common spatial map to syncing content across devices with minimal user friction.

- **OpenXR and Unity MRTK:** Quest 3 supports OpenXR, the open standard for XR development. Meta's extensions to OpenXR include support for passthrough AR and spatial anchors. Unity developers can leverage **AR Foundation** (Unity's cross-platform AR framework) with a Meta OpenXR plugin to develop AR apps on Quest. Unity's AR Foundation (as of 2023) added experimental support for Quest 2/3, enabling features like *stereoscopic passthrough, plane detection, raycasting,* and *anchors* via

Meta’s OpenXR extensions [6] [7]. This means Unity AR apps can run on Quest with similar APIs used for ARKit/ARCore, simplifying multi-platform development. On top of that, the **Mixed Reality Toolkit (MRTK)** by Microsoft is an open-source SDK that provides components for spatial interactions and UI in XR. MRTK was originally for HoloLens but has cross-platform support (including Meta Quest via Unity/OpenXR). Developers can use MRTK’s building blocks (hand menus, manipulation gestures, etc.) on Quest 3 to rapidly prototype MR interfaces, while relying on Meta’s platform underneath for device-specific features. Although MRTK itself doesn’t handle co-location natively, it can be combined with Quest’s Shared Spatial Anchors for multi-user scenarios. In summary, Quest 3 developers have multiple **SDK options**: use Meta’s Presence Platform SDK for direct access to Quest features (with Unity or Unreal engine), or use Unity’s AR Foundation/MRTK for a higher-level, cross-device approach – now compatible with Quest’s passthrough and anchors [8] [7]. Meta’s official stack (Presence Platform) is optimized for Quest hardware (with features like scene understanding and hands interaction), whereas frameworks like MRTK or AR Foundation can ease portability or UI development. Together, these tools make it technically feasible to build a shared AR app where all users see the same content anchored in the real world.

- **Other Multi-User AR Services:** Beyond Meta’s ecosystem, there have been cloud-backed AR cloud services (e.g. **Niantic Lightship** or **Azure Spatial Anchors**) which enable shared AR across devices by hosting spatial maps. In practice, for Quest 3 specifically, Meta’s own Shared Spatial Anchors are the most direct route (and do not rely on external cloud except Meta’s own). However, conceptually one could integrate third-party services if needed – e.g. using **Azure Spatial Anchors** SDK on Quest via OpenXR to share a spatial map between a Quest and a HoloLens or mobile device. Another avenue is using general networking engines (Photon, Mirror, etc.) in combination with a common calibration – for instance, having users scan a visual marker or QR code to align coordinate frames and then syncing object transforms over a network. But with Meta now providing built-in anchor sharing, the heavy lifting of alignment is handled by the platform. In summary, Quest 3’s support for **OpenXR** means developers aren’t locked in – they can mix and match these approaches – but the **Meta Presence Platform** (with Shared Anchors, Colocation APIs, voice SDK, etc.) is the most *technically robust and supported* path for co-located AR on Quest 3.

Prototype and App Examples Enabling Shared MR Conversations

Several cutting-edge applications and research prototypes demonstrate aspects of real-time shared MR collaboration – from gaming to productivity – where multiple users can spawn or interact with objects in a shared AR space. These examples illustrate what is currently possible, providing inspiration for an interactive 3D mindmapping tool:

- **Meta “Magic Room” (Horizon Workrooms AR):** *Magic Room* is an experimental mixed reality experience by Meta that blends co-located and remote participants in the same physical room. Previewed at Meta Connect, Magic Room uses Shared Spatial Anchors so that people physically together (wearing Quest headsets) and remote users (as avatars) can all see the same virtual content integrated into the real room [9]. For instance, a group in a conference room could see a shared virtual whiteboard on the wall and a 3D model on the table that everyone – local or remote – can reference. Magic Room is essentially an R&D prototype extending **Horizon Workrooms** (Meta’s VR collaboration app) into true co-located AR. It shows how AR can support *“real-world” meetings enhanced with virtual media*, aligning exactly with the idea of shared mindmapping (people jointly manipulating idea-objects around them). While not commercially released (as of mid-decade), Magic

Room indicates Meta's official interest in **co-present AR collaboration** and leverages the Quest's colocation tech (SSA). It demonstrates that *mixed groups* (some co-located, some remote) can collaborate with persistent shared references – a scenario that an interactive mind map could also support [9].

- **Collaborative AR Games (Demeo, Spatial Ops):** Multiplayer AR games on Quest provide concrete proof of fluid co-located experiences. A notable example is **Demeo**, a VR tabletop RPG that introduced a "Mixed Reality 2.0" update with **colocation support** on Quest. Using Shared Spatial Anchors, Demeo allows two or more players in the same room to play around a single virtual game board that stays fixed to a physical surface (e.g. a real table) [10]. Each player sees game pieces in the same location, and their avatars in-game match their real-world positions, just like people sitting around a real board game [10] [11]. This essentially recreates the feeling of a physical board game night using AR. The success of Demeo's co-location (made possible by Quest headsets syncing their coordinate space) shows that virtual content can be **shared seamlessly**: when one player moves or spawns an object, it appears at the correct spot for others. Another example is **Spatial Ops** by Resolution Games – a first-person shooter in passthrough AR. Spatial Ops lets up to 8 players map a room or outdoor area and then engage in laser-tag-like battles with virtual cover and weapons in that shared space [12] [13]. It was first unveiled as an experiment and later released on the Quest platform. Players physically run around the same space while seeing the same virtual obstacles and objectives. The game uses co-location techniques (SLAM mapping and anchor sharing) to ensure each headset's virtual world is aligned [14]. These games prove the **technical feasibility** of real-time, co-located AR: dynamic objects remain in sync and latency is low enough for interactive gameplay. For an interactive mindmapping scenario, the implication is that two users could likewise see and manipulate idea-nodes or media in a shared coordinate frame without jarring mismatches – the underlying tech is already being battle-tested in games.

- **Collaborative AR Design Tools (Arkio, Figmin XR):** Beyond games, there are practical apps focusing on co-creation in MR. **Arkio** is a collaborative architectural design tool available on Quest (and other devices). It supports multi-user sessions where users (remote or co-located) can build and review 3D models together. Notably, Arkio can detect when multiple Quest users are in the **same physical room** and automatically align their avatars and content [15]. With Quest 3's passthrough, Arkio even allows placing a building model in your real room for scale reference [16]. This is very much a "shared 3D mindmapping" for architects – they can brainstorm design ideas by sketching shapes in AR, and everyone sees the changes live from their perspective. Another example is **Figmin XR**, a creativity app that enables users to spawn and arrange 3D models, drawings, and web content in their space. Figmin XR supports multi-user collaboration as well, letting people decorate or brainstorm in AR together (Figmin was cited by Meta as an early adopter of the Presence Platform) [17]. While primarily single-user, **Wooorld** (a maps visualization app) also hints at shared exploration of a 3D map in MR. These apps underscore design principles such as *anchoring virtual content to real surfaces* (floors, tables, walls) and providing multi-user **synchronous editing**. In a mindmapping context, one could imagine users placing labeled nodes or images onto the walls or around them, viewable and editable by all present – much like Arkio's collaborative models or Figmin's shared AR scenes.

- **Academic Research Prototypes:** In the research community, prototypes have explored shared AR conversations and brainstorming. For example, studies on **co-located VR/AR alignment** provide methods to sync coordinate frames using visual SLAM merges or shared markers [18] [19]. One recent paper, *"A Quest for Co-Located Mixed Reality: Aligning and Assessing..."*, surveyed approaches for aligning disparate device coordinate spaces, underlining how crucial a common spatial map is for co-located MR [20]. Another relevant prototype is a **collaborative AR brainstorming tool using virtual Post-it notes**, where users in the same space can create and move AR sticky notes during a design session (e.g. *Augmented Post-it* concept, CHI 2023). These experimental systems often report improved engagement when participants can physically gesture to shared virtual notes or objects as if they were real, highlighting that **natural collaboration cues** (like pointing or walking around an idea) translate well into AR. Although many such prototypes are not productized, they demonstrate specific features (like freehand drawing in AR, speech-controlled note creation, etc.) that could feed into an interactive mindmapping app.

In summary, the state-of-the-art apps and prototypes have **validated key pieces**: Multi-user AR is workable on Quest hardware, and people can jointly manipulate virtual elements (game pieces, design models, notes) in real time. The concept of spawning relevant objects or media during conversation is a logical extension – for instance, if discussing a new product idea, a team could speak a prompt and have a 3D model or reference image appear in their shared space. The examples above show that each piece (co-location, object spawning, real-time sync) has been demonstrated in isolation, if not together.

Live Semantic Object Generation & Media Retrieval Tools

A core aspect of "interactive 3D mindmapping" is the ability to **pull in new content on the fly** – whether that means generating a 3D model from a spoken description, fetching a YouTube video related to the discussion, or displaying an excerpt from an article or research paper. Achieving this involves a combination of *semantic understanding* (knowing what the conversation is about) and content creation/retrieval services. Here we survey the tools and techniques that can power such functionality:

- **Text/Speech to 3D Object Generation:** Recent advances in generative AI have made it possible to create 3D models from text prompts, though often with trade-offs between speed and quality. For instance, OpenAI's **Shap-E** and Google's **DreamFusion** are models that can produce a 3D asset given a caption (Shap-E outputs a low-res mesh quickly, DreamFusion produces high-detail neural radiance fields with more compute). A 2025 study evaluated state-of-the-art pipelines for speech-driven 3D generation in AR [21] [22]. They found that a two-step approach – first using a text-to-image generator (e.g. **FLUX** model) and then converting that to 3D with a model like **Trellis** – yielded the best visual quality (users rated the results 4.55/5 on average for quality and alignment) [23]. However, this approach took longer (several tens of seconds). By contrast, a direct text-to-3D model like Shap-E was **faster (~20 seconds)** but produced lower fidelity output [24]. Users in the study preferred waiting longer for higher-quality objects, indicating that **perceptual quality trumps latency** for AR content generation [25]. This is encouraging for mindmapping: participants might tolerate a short pause as an AI generates a useful 3D illustration of an idea, as long as it looks coherent. Beyond research prototypes, there are emerging commercial tools (Meshy.ai, 3D Studio AI, etc.) that claim to generate simple 3D models or avatars from text in seconds. Many use diffusion models under the hood or retrieval + morphing. While still in early stages, such tools could be integrated via an API: e.g. send the user's description ("spawn an office chair here") to a cloud service that returns a textured 3D model of a chair to place into the scene. Another pragmatic approach is **semantic retrieval of existing 3D assets**: large libraries like Sketchfab or Poly haven't integrated directly into Quest apps yet, but a mindmapping assistant could use keywords to find a relevant Creative Commons 3D model online and import it. This avoids generation time by leveraging pre-made assets for common concepts (e.g. a globe, a DNA helix, etc.). In summary, **live object generation** is becoming feasible – maybe not photorealistic models instantly, but prototypes show that within half a minute useful AR objects can be synthesized from a description [24]. For faster response during conversation, a hybrid approach might be best: retrieve a known model if available (for immediate visualization) while concurrently refining or generating a custom model in the background.

- **Media Search and Retrieval (YouTube, Articles, Papers):** In a shared AR conversation, users might want to pull up supporting material – say, *"Here's a relevant video clip"* or *"This article explains the concept"*. Technically, this can be achieved by integrating web APIs for search. For example, a system could use YouTube's Data API to search for a keyword (extracted from the conversation) and then display the top result as a floating video screen in AR. Quest 3's browser or a webview component could be invoked to play the video. Similarly, for articles or research papers, services like the Semantic Scholar API or arXiv API can retrieve paper titles, summaries, or PDF links given a query. The **challenge** is filtering and choosing the right content automatically. One approach is to use a **large language model (LLM)** as a controller: feed it the conversation context or keywords and let it generate a search query, then use that to fetch results. For instance, if the users say *"Remember that 2018 study on climate networks?"*, the system's agent could hit an academic search API and then spawn a small panel in AR showing the paper's title, authors, and a summary snippet for everyone to see. This effectively creates a "research paper node" in the mind map. Tools exist to facilitate this: for example, OpenAI's functions or retrieval plugins can perform web searches and return content. Within the Quest environment, developers could use the **Quest 2D app APIs** to show web content alongside VR (though integration may be limited for custom apps). Alternatively, pre-curating a knowledge base might be useful: an enterprise version could have a local repository of PDFs or media that an AI can index and retrieve in real time, showing relevant info nuggets in AR when triggered by conversation.

- **Real-Time Speech Recognition & Semantic Analysis:** To drive both of the above (object generation or media fetch), the system needs to *understand the conversation*. Fortunately, Quest 3 has on-board microphones and Meta provides a **Voice SDK** for speech recognition. Meta's Voice SDK uses **Wit.ai** NLP to convert speech to text and parse intents/entities [26] [27]. Developers can define custom intents (like "spawn [object]" or "search [topic]") or use built-in ones. In the context of a free-flowing conversation, the Voice SDK could continuously transcribe key phrases (with user consent) and detect when the users mention a query-worthy topic. For example, if two users say "Let's pull up the sales chart from last year," a voice intent could be recognized to fetch that data visualization. The Voice SDK promises real-time transcription with low latency and even supports wake-word activation or continuous dictation modes [28] [29]. It is cross-platform (works on Quest and other devices) and can differentiate some built-in commands. Using such a service, the mindmapping app can be **speech-driven** – the most natural interface during a conversation. One user's spoken idea can instantly trigger a search or creation event without them fiddling with menus. Additionally, semantic analysis (either via Wit.ai's NLU or a more powerful LLM in the backend) can determine context: e.g., linking a spawned object to the conversation topic. If the topic changes, the system might organize the AR content differently (grouping related items together spatially). In summary, *speech-to-text and NLP tools* are essential glue for interactive AR content. On Quest 3, developers can use Meta's SDK to handle voice input, which then can feed into custom logic or AI services that decide what object or media to bring into the shared space [30] [31].

By combining these tools, a powerful workflow emerges: **listen → understand → retrieve/generate → display.** For instance, as users talk about a historical event, the system's pipeline could (1) transcribe their mention of, say, "Apollo 11 mission," (2) recognize this as a cue, (3) automatically retrieve a relevant image or 3D model of the Saturn V rocket, and (4) spawn that model into the AR space for both users to examine. While a fully autonomous "mindmapping assistant" is complex, many building blocks exist. The limitations to note are computational – high-quality generative models might need cloud computing (Quest 3 can't run a big text-to-3D model locally in real time), and internet access – fetching live media requires network connectivity and raises content safety considerations. However, given Quest 3's Wi-Fi 6E capability and the trend of edge computing, one can envision such services running with minimal delay. The result would be a richly augmented conversation: essentially an AI that **injects visual and textual knowledge** into the shared AR scene as new nodes whenever helpful.

Conversational AI Integration in AR Environments

Integrating conversational AI (especially large language models) into AR can make the experience far more interactive and context-aware. In a shared 3D mindmapping scenario, an LLM could act as a facilitator or creative assistant – listening to the dialogue and proactively helping users by suggesting ideas or retrieving relevant info, all manifested in AR. There are emerging approaches and prototypes that illustrate how this could work:

- **AR Dialogue Assistants:** Researchers have started combining AR with LLMs to assist face-to-face conversations. One example is *ChatAR*, a system that uses a head-mounted AR display and an LLM (ChatGPT/GPT-4) to support users in conversation [32] [33]. ChatAR continuously speech-to-text transcribes the dialogue between two people. It then feeds keywords or the gist into an LLM to get contextual information or suggested responses, which are displayed in AR to one of the users. Importantly, ChatAR explores UI techniques to present this info subtly – for instance, floating text at a position that doesn't require obvious eye movement, so the other person isn't aware of any "assistant" intervention [34] [35]. In experiments, they found that this approach helped users contribute more meaningfully (reducing knowledge gaps) without breaking the flow of conversation [36]. While ChatAR focuses on text hints (like providing facts about a topic being discussed), the same concept could be extended to **visual aids**. An LLM could just as easily return: "Show diagram of XYZ" or "Play clip from X" as part of its output, which the AR system can then execute (with appropriate APIs to fetch that media). The key role of the LLM here is understanding context and **proactively injecting relevant content**. In a mindmapping session, the AI might detect a complex idea and spawn an explanatory 3D model or present a list of pros/cons from a knowledge base, essentially participating in the brainstorming.

- **LLM Agents for AR Content Creation:** Beyond retrieving known information, LLMs can interface with generative tools to create new content on the fly. For example, consider an AI agent that listens to the conversation and when it hears an idea like "we could have a mascot that's a mix between a cat and a unicorn," it uses an image generator or 3D generator to *create* that mascot and then spawns it into the AR space. While this is bleeding-edge, it's plausible by connecting LLMs with plugins or function calls. In fact, frameworks like Microsoft's **Guidance** or OpenAI's function calling allow an LLM to decide to call an image generation API when needed. If the AR app exposes a function like `spawn_object(description)`, a sufficiently tuned LLM could decide to call it with an appropriate description derived from the conversation. This effectively gives the AR system a kind of autonomous "imagination." Early demos of this idea include experimental projects where GPT-4 was used to generate Unity scene descriptions or VR environments from prompts (though not in real-time conversation). Another example is *SocialMind*, an LLM-based AR social assistant prototype that monitors social interactions and offers in-situ help; it hints at how an AI could analyze not just words but also context (e.g., recognizing objects or people around) and then provide AR guidance [37]. In a collaborative mindmapping scenario, an agent might perform tasks like summarizing the discussion so far (and displaying a summary node), keeping track of agreed action items in a corner of the AR space, or even role-playing a "Devil's Advocate" by spawning a counterargument node to stimulate debate. These are speculative use cases, but they align with the trend of **LLM as an autonomous partner** in creative workflows.

- **Multimodal Context Awareness:** A big advantage of AR devices is access to multiple sensors – video, audio, spatial mapping, etc. Future conversational AI in AR can leverage these to be context-aware. For example, Meta's research has looked at *contextual LLM assistants* that know what the user is doing or looking at. An arXiv paper (2023) presented a context-aware AR assistant where the LLM got input like the user's task description, hand movements, and surrounding objects, enabling it to give relevant real-time instructions [38]. In Quest 3's case, an AI could tap into the **Scene Understanding** data – knowing the geometry of the room and recognizing certain objects or surfaces. If users are discussing "let's put this diagram on the wall," the system (with scene API) knows where the wall is and can project the content there. Conversationally, the AI could remember previous ideas that were spawned and reference them (*"Earlier you mentioned X; here's a link between X and Y"*). Essentially, the LLM can maintain the **semantic map** of the mindmapping session: linking nodes that are related, suggesting missing connections, or flagging inconsistencies – all in natural language dialogue with the users. This turns the mind map from a static collection into something more like a collaborative knowledge graph with an AI mediator.

One current limitation is that running a large model directly on Quest 3 is not practical; the headset would rely on cloud services or perhaps a paired device for the AI computations. This introduces latency and connectivity requirements. However, Meta has been exploring on-device assistants (their *Meta AI* assistant, though geared toward text and image generation, could eventually extend to AR). Projects like **Llama 2** (open-source LLM) could be fine-tuned for AR instructions and run on a nearby PC or edge server with results streamed to the Quest. Privacy and UX must also be considered – users may not want an AI listening continuously, so explicit activation (like a wake word or a gesture to "call the assistant") could be used.

In summary, **conversational AI integration** can transform a shared AR experience from user-driven commands to a more fluid, AI-supported dialogue. Early research (like ChatAR) demonstrates improved conversations by providing real-time info overlays [34]. Extrapolating from this, an LLM in the loop can dynamically grow the 3D mind map, suggest new branches, retrieve clarifications, and ensure both participants have a richer context. The design goal should be that the AI remains a helpful participant *without overshadowing the human-to-human interaction*. When done right, it's like having an expert facilitator in the room who can conjure any reference or object you need – truly a "magical" augmentation of a discussion.

Interaction Design Paradigms for Shared 3D Mindmapping

Designing an interactive 3D mindmapping experience in MR requires rethinking traditional mind map and brainstorming tools for a spatial, multi-user context. A number of paradigms and patterns can guide this design:

- **Spatial Node-Link Diagrams:** At its heart, a mind map is a network of nodes (ideas) connected by links (relationships or hierarchy). In 3D space, these can be represented as floating labeled objects (nodes) connected by lines or arrows. Apps like **Noda VR** (a mind-mapping tool for VR/Quest) have embraced this by giving users an *infinite 3D canvas* to create node-link diagrams around themselves [39]. Each idea is a sphere or card that can be grabbed and placed anywhere; users can draw connections or use a menu to link nodes. Noda even supports collaborative mode, meaning multiple people can contribute to the same 3D diagram in real time [39]. The advantage of 3D mind maps is that you're not limited by screen space – you can branch ideas in any direction, cluster related concepts in different areas of the room, and utilize depth (important or core ideas could be larger or placed centrally, with sub-ideas radiating outwards). **Walking around** becomes part of the interaction: users can physically move closer to a cluster of interest or step back to see the big picture. This embodies the concept of *"immersive knowledge management"*. Design patterns here include allowing users to tag nodes with colors or icons (for visual categorization), having the ability to "pin" nodes to real-world locations (e.g., attach a note to a specific wall or on top of a physical object that's relevant), and maintaining legible text at various distances (possibly by billboarding text toward each user or scaling it with distance).

- **Use of Physical Surfaces and Props:** Mixed reality mindmapping can take advantage of the physical environment as part of the interface. For example, a large blank wall in the room could serve as a canvas where the AR system arranges a 2D overview of the mind map for reference (like a projector painting the mind map on the wall for everyone to see). Meanwhile, the table could hold detailed 3D models or media content that have been spawned, treating it like a shared workbench. This division leverages natural human behaviors: walls for posting notes (as in real-world sticky note brainstorming) and tables for examining objects or documents. The **Meta Scene Understanding** API on Quest can identify walls, tables, ceiling, etc., enabling the app to intelligently place content in ergonomic locations [40] [41]. For instance, when a user asks for a web article, the system might open it as a large virtual screen on the wall (since reading is easier when it's fixed in front of you), whereas a 3D object spawns at the center of the table where everyone can walk around it. This approach was hinted at by earlier AR experiences – for example, design workshops with AR often project text on vertical surfaces and models on horizontal ones to avoid clutter. Additionally, physical props or markers can be used: imagine a neutral object like a cube that, when placed on the table, acts as a "topic marker" – the AR system can bind a sub-map to that cube's location. Moving the cube could reposition that entire sub-section of the mind map. While experimental, such tangible interfaces could add a tactile dimension to brainstorming, grounding abstract ideas in physical action.

- **Natural Gestures and Multi-Modal Input:** In a collaborative AR space, users should be able to use **hands, voice, and even eye gaze** fluidly to interact with content. Quest 3 supports hand tracking (with gestures like pinch for select, grab, etc.) and even **Direct Touch** where you can poke virtual objects as if they were in front of you. This means a user could grab a virtual sticky note with their hand and stick it somewhere, or draw a connecting line between two nodes with a gesture. Hand tracking in MR has improved (as seen in Demeo's hand-tracking update which felt seamless for grabbing and moving game pieces [42]), implying that grabbing and moving idea-nodes is quite feasible and intuitive. Voice input, as discussed, allows rapid creation of content (e.g., saying “new node: Market Trends” could instantly create a node with that label). Combining inputs leads to powerful workflows: you might say “show video about topic X” then use your hand to resize the video panel or move it to a better spot. For multi-user considerations, if two people reach for the same object, the system needs rules (maybe objects can be “owned” when grabbed, turning a different color to indicate who’s moving it). Gaze can be leveraged for subtle UI hints – e.g., if both users are looking at a particular node, the system could highlight it or prompt “do you want to expand this idea?” Since maintaining conversational flow is key, the interactions should be as unobtrusive as possible – **no complex menus or controllers** if it can be helped. The design principle is that the AR content behaves like real objects or notes that the group can manipulate directly, with voice and gaze as additional pointers to intention.

- **Visual Encoding and Organization:** A mind map can grow complex, so spatial organization and visual cues are important. Designers can use the third dimension to layer information. For example, one could stack related content along the vertical axis (like a timeline or layered categories), or use distance from the center to indicate priority (central ideas vs peripheral extras). Colors and shapes can denote different types of content: perhaps all video nodes have a filmstrip icon and blue tint, all 3D model nodes are represented as floating 3D thumbnails, and textual notes are yellow like sticky notes. Lines connecting ideas could be drawn in 3D space as subtle glowing threads – possibly even animated or curved for clarity. One interesting paradigm is **temporal mindmapping**, where the conversation’s progression is mapped along one axis – e.g., earlier discussed items appear on the left side and later ones on the right, creating a chronological layout. Users could then literally walk from the start to the end of the discussion in AR space to recap. Design patterns from information visualization (like clustering, fisheye views, etc.) might be employed: users could perform a “zoom out” gesture to collapse details and see only high-level nodes, or conversely zoom in on a cluster to isolate and delve deeper. Since this is shared among users, visual consistency and legibility for all parties is crucial. Techniques like **shared focus** (highlighting the object one user is talking about for everyone) can guide attention. Also, providing personal view options may help – each user might summon a private miniature overview of the mind map that they can glance at (floating near their headset) without disturbing the main shared space.

- **Inspiration from Existing Mind Mapping Tools:** Traditional mindmapping software and newer AR brainstorming apps offer insights. As an example, **Mind Map AR (ScapeHop)** for smartphones placed 3D mind maps into your environment via your phone’s camera [43] . Users of that app reported that being able to *“walk through and around”* their idea structures was highly engaging [44] . The creator noted that the 3D format taps into our natural spatial memory – the hippocampus – making it easier to remember and relate ideas [45] . They also highlighted voice input as a key enabler to “grow your mind map on pace with the flow of your thoughts” [46] , which is precisely what we’d want in a conversational setting. These insights reaffirm that **movement and spatial immersion amplify creativity** and recall. Similarly, the VR app Noda emphasizes that once you use 3D space, you don’t want to go back to 2D for brainstorming – the freedom and intuitiveness are unparalleled [47] [48] . Therefore, design should maximize the benefits of spatial interaction (not confining users to 2D panels or forcing VR keyboard use unnecessarily). Another guideline from these tools is to keep the interface simple and intuitive: Mind Map AR’s creator mentions that just tapping or gazing at a node to select and speak to edit was sufficient for most actions [49] . In a multi-user scenario, perhaps a simple rule like *raise your hand to "claim" a node for editing* could avoid confusion.

Overall, the interaction design for shared 3D mindmapping should strive for a **"walk-up and use" experience**: two users put on Quest 3 headsets, see each other and their surroundings in passthrough, and then simply start talking and gesturing as if using an invisible whiteboard that magically manifests their ideas. The technology (voice recognition, object generation, etc.) should operate in the background, presenting results in an easy-to-consume way in the environment. By adhering to familiar metaphors (sticky notes, whiteboards, round-table discussion) while enhancing them with AR superpowers, the experience can feel natural and *collaboratively empowering* rather than gimmicky.

Quest 3 Hardware Affordances and Constraints

Building such an advanced MR application on the Quest 3 means working within the device's hardware capabilities and limitations. Fortunately, the Quest 3 brings notable improvements over its predecessors, but developers must still plan around performance and UX constraints:

- **High-Fidelity Color Passthrough with Depth:** The Quest 3 is designed for mixed reality with dual 4 MP RGB cameras providing a **stereoscopic color passthrough** view of the real world [50]. This means users see their surroundings in relatively sharp color (18 PPD resolution) instead of the grainy B&W of Quest 2. The stereo passthrough combined with the *new depth sensor (IR projector)* gives much better depth perception and occlusion capabilities [50] [51]. Virtual objects can be more convincingly locked to real surfaces and will appear at correct scale and depth. For our use case, text and images spawned in AR are easier to read and 3D models integrate more naturally (e.g., a virtual object will correctly appear *behind* a real wall if placed so). The depth sensor allows the system to occlude virtual content behind real objects and to reconstruct a mesh of the environment for physics and collision. Quest 3's **field of view** (~110° H × 96° V) is slightly larger than Quest 2's [52], meaning users can see more of the AR content around them without turning their head – useful for spatial mind maps that might surround the user. However, it's still a limited FOV compared to human vision, so designers should keep important content within that central area.

- **Inside-Out Tracking and Shared Alignment:** Quest 3 uses inside-out SLAM tracking (with cameras) for 6DOF head and hand tracking. In co-located use, each headset builds its own map of the environment. To align them, the **Shared Spatial Anchors** approach effectively merges these maps via cloud or direct sharing – but there are some practical constraints. For example, when users initiate a colocation session, Quest headsets may require uploading portions of their environment point cloud to Meta's servers to find a common anchor (as indicated by the system prompt in Demeo asking to "share point cloud data with Meta" [53]). This implies an internet connection is needed (at least briefly) for the alignment step, and there might be slight delays or failures if the environment lacks distinctive features. Additionally, lighting conditions affect tracking – very dim or very bright direct light can reduce camera tracking quality. For optimal results, the physical space should be moderately lit and have enough visual features (textured walls, furniture) for the SLAM to latch onto. Quest 3's improved cameras help in lower light than Quest 2, but it's not night-vision. From a hardware perspective, two users in the same room should ideally stay within a certain range (Bluetooth discovery works ~30 feet, and shared anchors presumably assume users are in one contiguous space). If one wanders to the next room, the shared experience may break or require another anchor.

- **Performance (SoC and Battery):** Quest 3 is powered by the Snapdragon **XR2 Gen 2** chipset, which offers significantly more processing power (CPU and GPU) than Quest 2 [54]. This enables more complex scenes and heavier algorithms – e.g., rendering multiple high-poly 3D models or running advanced computer vision for scene understanding – at acceptable framerates. It also has 8 GB of RAM (up from 6 GB in Quest 2), helpful for loading large assets or caching voice/AI models [55]. However, it's still a mobile chipset; developers must be mindful of the 72–120 Hz VR framerate target. A rich mindmapping app might need to draw many text labels and object models, manage dynamic networking, and possibly stream media – all of which can tax the system. Techniques like level-of-detail reduction, culling objects not currently viewed, and using simple shaders for UI will be important to keep performance smooth. **Thermal throttling** can occur if the CPU/GPU are stressed for long periods (the headset can get warm and then downclock). On the battery side, Quest 3 offers around **2 hours of intensive use** per charge (similar to earlier Quests) [56]. A collaborative session may well approach this length, so for longer workshops an external battery or intermittent charging might be needed. In terms of on-device AI, the XR2 Gen 2 has a Hexagon DSP that can accelerate some neural network workloads, but running a large LLM or image diffusion model locally would likely be too slow and drain battery quickly. Offloading AI tasks to a server or edge PC is the expected solution, trading a bit of latency for preserving device performance.

- **Networking and Sharing:** Quest 3 supports **Wi-Fi 6E** and Bluetooth 5.2, which are relevant to multi-user experiences [57] [58]. Wi-Fi 6E means in a local network (e.g., same room with a Wi-Fi 6 router) the headset can have very high bandwidth and low latency – beneficial if the app streams content or if using local PC for computation. For example, if a PC is used to run the LLM or 3D generation, Wi-Fi 6E can send the results (like a generated model) to the headset quickly. Bluetooth 5.2 is used by the **Colocation Discovery** feature to detect nearby devices and bootstrap the session [4]. It's also used for pairing controllers, etc. The range and reliability of Bluetooth is usually solid in-room, but developers should ensure the app handles cases where the initial discovery fails (fallback to manual session codes, etc.). Another consideration: if multiple headsets are sharing data peer-to-peer (like sending anchor IDs or even streaming point clouds to each other), a local network or **Wifi-Direct** connection could be utilized. Quest doesn't expose Wifi-Direct easily to developers, but using a local router all devices are connected to is standard. The **Group Sharing API** introduced by Meta essentially abstracts this by using a group ID on their backend to let all devices fetch shared anchors [59].

- **User Comfort and Safety:** When two or more users are moving around the same space with VR headsets on, safety is critical. Quest 3's passthrough AR gives users awareness of each other and obstacles, which is a big improvement over full VR where collisions can happen. Still, designers should consider keeping virtual content at a comfortable **distance** – e.g., avoid forcing users to focus on something 10 cm from their face or constantly refocus between very near and far objects. The Quest's optical design has a fixed focus around 1.3 m; viewing virtual objects much closer can cause eye strain. Thus, placing most content at least an arm's length away (0.5 m or more) is advisable for legibility and comfort. Another hardware affordance: Quest 3 has **stereo speakers** for spatial audio, which could be used to provide subtle audio cues when new content spawns (e.g., a chime to draw attention). If using voice input heavily, echo and cross-talk can be an issue (each headset might pick up both users speaking). Good noise-canceling on the mics and possibly coordination (maybe only one device actively transcribes at a time, or each device focuses on its wearer's voice) might be needed to ensure accurate recognition. Finally, **controller vs hands**: Quest 3 comes with improved Touch Plus controllers (no tracking rings, inside-out tracked). Some users might prefer using controllers as laser pointers or for precision selection, while others use hand gestures. A well-designed app can support both interchangeably. For example, one person could grab and move objects with their hands while another uses a controller raycast to drag connections between far-apart nodes. The system can unify these inputs so it feels like a coherent interaction paradigm (perhaps abstracting both as "pointers" in code).

In conclusion, the Meta Quest 3 provides a very capable platform for shared AR: its **sensor array (stereo RGB + depth)** and **shared anchor system** address the core MR alignment challenges, and its improved performance and connectivity open the door to integrating advanced AI and rich content. Developers just need to remain mindful of the *mobile nature* of the device – optimizing for performance, managing battery life, and using cloud resources for heavy lifting when needed. By leveraging the Quest 3's strengths (full-color passthrough, hand tracking, spatial anchors) and mitigating its constraints, one can indeed create a compelling **co-located AR mindmapping experience** that feels responsive and immersive for all participants.

Conclusion

The vision of an "Interactive Shared 3D MindMapping" tool on Quest 3 is well within reach given the current state of technology. **Meta's Presence Platform** supplies the essential glue for co-located AR (shared spatial mapping, user discovery, anchors), enabling multiple users to share a stable AR space [1] [3]. Real-world applications and prototypes – from co-op AR games like Demeo and Spatial Ops to collaborative design apps like Arkio – have demonstrated that people can naturally collaborate with virtual content in a shared physical room. The ability to spawn rich media in AR is supported by a convergence of AI and XR tech: generative models can create on-demand 3D assets or fetch information, while voice and conversational AI can make interacting with those systems as easy as talking to a helpful teammate. Designing such an experience will draw on UI paradigms of both the physical and digital realms: the familiarity of sticky notes, sketches, and hand gestures, combined with the power of infinite virtual space and intelligent assistance.

There are certainly challenges ahead – ensuring the timeliness and accuracy of AI-generated content, keeping the system's suggestions relevant and unobtrusive, and maintaining performance on a standalone headset. User acceptance factors (like privacy concerns of an always-listening AI, or the comfort of wearing a headset in group settings) will also influence adoption. Yet, the technical pieces are aligning at the right time: Quest 3's mixed reality capabilities, coupled with advances in large language models and generative 3D, create a fertile ground for innovation. By prioritizing robust technical solutions (as seen in academic research and Meta's own frameworks) and thoughtful design, developers can deliver a shared MR mindmapping experience that truly feels like a *"magical leap beyond ordinary reality"* [60] – where ideas literally take shape around you, knowledge is at your fingertips (or voice command), and collaboration is elevated to an immersive spatial canvas.

Ultimately, such a tool could transform how we brainstorm and communicate, making conversations more visual, interactive, and memorable. Two people with Quest 3 headsets could sit in a room and have a free-flowing discussion that automatically **materializes into a web of insights** – a living mind map they build together, see together, and remember together. The technology and research surveyed here suggest that this is not science fiction, but an imminent reality that just requires clever integration and experience design.

Sources:

- Meta Quest Developer Blog – *"Mixed Reality with Scene Understanding, Full-Color Passthrough, and Shared Spatial Anchors."* [1] [9]
- Meta Quest Developer Blog – *"Seamlessly Power Colocated Mixed Reality Experiences with Colocation Discovery and Group Sharing."* [2] [3]
- UploadVR – *"Demeo Now Supports Hand Tracking & Colocation on Quest."* [10] [11]
- Resolution Games – Spatial Ops description [12]
- Unity Learn XR Blog – AR Foundation for Meta Quest (Dilmer) [6] [7]
- Xiu et al. (2025), *"Say It, See It: Speech-Based 3D Content Generation for AR."* [23] [24]
- Fujimoto (2025), *"ChatAR: Conversation Support using LLM and AR."* [32] [35]
- MindMappingSoftwareBlog – *Interview on Mind Map AR (ScapeHop).* [45] [46]
- Meta Developers – Quest device specs and features [50] [57]
- Meta Developers – Voice SDK Overview [30] [31]

---

[1] [9] [17] [40] [41] Blog | Meta Horizon OS Developers
https://developers.meta.com/horizon/blog/presence-platform-mixed-reality-social-presence-connect-2022/

[2] [3] [4] [5] [59] [60] Blog | Meta Horizon OS Developers
https://developers.meta.com/horizon/blog/colocation-discovery-group-sharing-shared-spatial-anchors-mixed-reality/

[6] [7] [8] AR Foundation With Meta Quest Support Is Here! (Unity Setup & Demos) — LEARN XR BLOG
https://blog.learnxr.io/xr-development/ar-foundation-with-meta-openxr-package

[10] [11] [19] [42] [53] Demeo Now Supports Hand Tracking & Colocation On Quest Headsets
https://www.uploadvr.com/demeo-hand-tracking-colocation/

[12] Spatial Ops - Competitive Multiplayer Shooter in Mixed Reality
https://www.resolutiongames.com/spatialops

[13] [14] Behind the Scenes of Spatial Ops: Building a Seamless MR ...
https://www.resolutiongames.com/blog/behind-the-scenes-of-spatial-ops-building-a-seamless-mr-experience

[15] Streamlined workflows and enhanced collaboration - Arkio 1.7
https://www.arkio.is/blog/release1-7/

[16] American Structurepoint Accelerates Client Decisions with Arkio
https://www.arkio.is/blog/case-study-structurepoint/

[18] Co-Located VR with Hybrid SLAM-based HMD Tracking and ... - arXiv
https://arxiv.org/html/2509.06582v1

[20] A Quest for Co-Located Mixed Reality: Aligning and Assessing ...
https://dl.acm.org/doi/10.1145/3385956.3418968

[21] [22] [23] [24] [25] [2508.12498] Say It, See It: A Systematic Evaluation on Speech-Based 3D Content Generation Methods in Augmented Reality
https://arxiv.org/abs/2508.12498

[26] [27] [28] [29] [30] [31] Meta for Developers
https://developers.meta.com/horizon/documentation/unity/voice-sdk-overview/

[32] [33] [34] [35] [36] ChatAR: Conversation Support using Large Language Model and Augmented Reality
https://arxiv.org/html/2506.16008

[37] SocialMind: LLM-based Proactive AR Social Assistive System with ...
https://dl.acm.org/doi/10.1145/3712286

[38] Context-Aware Real-Time Assistance in Augmented Reality - arXiv
https://arxiv.org/html/2511.00730v3

[39] noda - mind mapping in virtual reality, solo or group
https://noda.io/

[43] [44] [45] [46] [47] [48] [49] Mind Map AR: The world's first augmented reality visual mapping tool
https://mindmappingsoftwareblog.com/mind-map-ar-augmented-reality/

[50] [52] [54] [55] [56] [57] [58] Meta for Developers
https://developers.meta.com/horizon/resources/compare-devices/

[51] Meta Quest 3: Next-Gen Virtual Reality Headset
https://www.meta.com/quest/quest-3/?srsltid=AfmBOopjeIsFWvJ-XAU9ajY3sPXGTco4_CUtIWGsOQqJD-sO7vfAsKsy