# Best Practices for Technical Writers Documenting XR Products

Prepared by The Docs Layer

## Executive Summary

Extended Reality (XR) encompasses virtual reality (VR), augmented reality (AR), and mixed reality (MR), immersive technologies that fundamentally change how users interact with information and interfaces. Technical documentation for XR products requires a paradigm shift from traditional software documentation approaches.

This report outlines evidence-informed best practices for documenting XR experiences across all three modalities, with emphasis on spatial interaction, user safety, and platform-specific considerations for leading frameworks such as Unity and Unreal Engine.

# 1. Understanding the XR Documentation Landscape

## 1.1 VR (Virtual Reality)

Virtual reality creates fully immersive, computer-generated environments where users are largely isolated from the physical world. Users interact with 3D content through VR headsets (e.g., Meta Quest series, PlayStation VR, HTC Vive) and controllers.

**Documentation Challenges:**

- Users cannot easily reference external documentation while wearing headsets
- Instructions must be intuitive and discoverable within the experience
- Error recovery requires clear, accessible in-experience feedback mechanisms

## 1.2 AR (Augmented Reality)

Augmented reality overlays digital content onto real-world environments, visible through mobile devices or smart glasses. Users interact with both physical and digital worlds simultaneously.

**Documentation Advantages:**

- Documentation can reference real-world contexts and surfaces
- Users can view instructional content on personal devices alongside AR experiences
- In-app tutorials can directly annotate interactive elements

## 1.3 MR (Mixed Reality)

Mixed reality blends VR and AR, allowing virtual objects to interact with real-world environments. Devices such as Microsoft HoloLens and emerging Android XR platforms support spatial awareness, occlusion, and persistent object placement.

**Documentation Complexity:**

- Must account for variable real-world environments (lighting, surfaces, spatial constraints)
- Requires explanation of spatial anchoring and persistence concepts
- Users need guidance on managing mixed environments safely and effectively

# 2. Core Principles for XR Documentation

## 2.1 From Explanation to Demonstration

Traditional documentation explains concepts with text and static images. XR documentation must emphasize ***demonstration over explanation***.

**Best Practice:** Embed instructional guidance directly into the user experience through:

- **In-app tooltips and annotations**: Spatially positioned hints highlighting interactive elements
- **Guided walkthroughs**: Step-by-step animated demonstrations within the experience
- **Spatial annotations**: AR overlay labels identifying objects, functions, and controls
- **Progressive disclosure**: Gradual introduction of features as users gain competence

**Example:** Rather than explaining controller mechanics in text, many XR frameworks demonstrate interaction through visual feedback such as reticle highlighting, hand proximity indicators, and haptic responses (see Unity's XR Interaction Toolkit).

## 2.2 Account for Hardware and Input Modalities

Different XR platforms support different input methods, which directly affects how users interact with both content and documentation.

**Input Types to Document:**

- **Hand-held controllers**: Buttons, triggers, thumbsticks
- **Hand tracking**: Gesture-based interactions such as pinch, grab, and point
- **Eye tracking**: Gaze-based focus and selection
- **Voice commands**: Speech-based interaction and help access
- **Ray casting and reticles**: Pointing systems indicating object presence and distance

**Documentation Requirement:** Favor input-agnostic language where possible (for example, "select input" rather than a specific button), and clearly distinguish platform-specific interactions when necessary. Always document fallback options for users unable to use certain input methods.

## 2.3 Design for User Safety and Comfort

XR documentation must prioritize user well-being, addressing both physical and cognitive safety.

**Safety Considerations:**

- **Real-world collision prevention**: Document both physical playspace requirements and comfort guidelines. Physical boundaries prevent collisions. XR UI recommendations (panel distances, gaze angles) prevent motion sickness.
- **Cybersickness mitigation**: Document locomotion options, field-of-view management, and rest intervals.
- **Cognitive load reduction**: Avoid overwhelming users with excessive UI elements or conflicting information.

- **Accessibility requirements**: Accommodate users with motion sensitivity, color blindness, hearing impairments, and mobility limitations.

**Documentation Practice:** Include explicit guidance on comfortable interaction distances, recommended session durations, and common warning signs of discomfort.

### 2.4 Embrace Spatial UI Principles

XR interfaces exist in three-dimensional space with depth, scale, and positioning. Documentation must reflect these spatial characteristics:

**Spatial UI Documentation Guidelines:**

- **Depth and distance**: Specify recommended positioning ranges using platform design heuristics
- **Field-of-view management**: Explain how interfaces adapt to user position and movement
- **Scale variability**: Describe how content scales based on distance and device capabilities
- **Anchoring and persistence**: Clarify how virtual objects maintain position relative to the real world or to the user

## 3. Platform-Specific Documentation Standards

### 3.1 Unity XR Development Framework

Unity provides cross-platform XR development through integrated tools and frameworks. Documentation for Unity XR projects should address the following systems:

**XR Interaction Toolkit Documentation**

The XR Interaction Toolkit is Unity's standard framework for VR and AR interaction. Documentation should clarify:

- **Component-based architecture**: Interactors, Interactables, and Controllers
- **Input abstraction**: How platform-specific bindings map to abstract actions
- **Feedback systems**: Haptic, visual, and audio feedback mechanisms
- **Interaction lifecycle**: Hover, select, activate, and deactivate states

**Example Documentation Pattern:**

> "To interact with an object, aim the controller or hand ray at the target. When the object highlights, it indicates a hover state. Activate the configured select input to manipulate the object. Release to deselect."

**AR Foundation Documentation**

For AR applications, Unity's AR Foundation abstracts platform differences. Documentation should cover:

- Plane detection and content placement
- Touch and gesture-based interaction patterns
- Light estimation effects on visual realism
- Occlusion behavior with real-world surfaces

**General AR Example:** Early consumer AR applications such as Pokémon GO highlighted the importance of adequate lighting and visually textured surfaces to improve plane detection and object placement reliability, a principle applicable across AR frameworks.

**Android XR and OpenXR**

Recent Android XR support in Unity introduces additional documentation needs:

- Spatial panels and their relationship to user position
- Orbiting UI elements that follow gaze or orientation
- Passthrough modes such as Full Space and Home Space (e.g., ARCore Extensions 1.45+), as defined in recent Android XR releases
- Environment setup, including spatial mapping and tracking

**Documentation Requirement:** Emphasize that Android XR behaviors evolve rapidly, and direct readers to current platform documentation for SDK and version-specific details rather than treating mode semantics as fixed.

## 3.2 Unreal Engine XR Development

Unreal Engine offers robust native XR support. Documentation considerations include:

- Motion controller input mapping
- HMD tracking and coordinate systems
- Stereo rendering behavior
- Performance optimization targets and techniques
- Platform-specific plugins and configuration differences

For parity with Unity documentation, Unreal-focused guidance should also reference:

- The OpenXR plugin as the primary abstraction layer for cross-device support

- Niagara systems for XR-compatible visual effects, with attention to performance constraints

Documentation should explain how XR features integrate into existing Unreal workflows rather than treating them as isolated systems.

### 3.3 Microsoft HoloLens and Mixed Reality

HoloLens documentation presents unique considerations due to passthrough-based MR.

**Key Topics:**

- Spatial anchoring and hologram placement
- Tag-along versus display-locked content
- Gaze, gesture, and voice input
- Spatial mapping and mesh visualization
- Project setup, build configuration, and deployment via Visual Studio

Documentation should note device-specific constraints such as shader compatibility and recommended render pipelines.

### 3.4 Meta Quest and Oculus Ecosystem

Meta provides extensive developer documentation and tooling.

**Key Areas to Document:**

- SDK and plugin setup
- Simulator-based iteration workflows, with clear limitations compared to real hardware
- Presence Platform and social features
- Controller layouts and haptic feedback patterns

Documentation should explicitly state that simulator testing supports early development and UI validation, but does not replace testing on physical devices for comfort, tracking fidelity, or performance validation. Testing on physical devices is a user safety requirement.

## 4. Documentation Formats for XR

### 4.1 In-Experience Documentation (Primary)

XR documentation should primarily live within the experience itself.

**Implementation Methods:**

- Spatial tooltips

- Guided onboarding sequences
- Help panels and menus
- Visual gesture demonstrations
- Optional voice-activated help for accessibility and hands-free use

**Quality Standards:**

- Keep text concise
- Maintain high contrast and legibility
- Avoid obstructing interaction zones
- Provide audio alternatives for visual-only cues

### 4.2 External Documentation (Supplementary)

Supplement in-experience guidance with:

- Mobile-friendly web documentation
- Video tutorials
- Printable quick-start guides
- Community forums and FAQs
- Developer-facing API references

### 4.3 Safety and Warning Documentation

Consolidate safety guidance into a clearly labeled section:

- Physical space requirements
- Health and age warnings
- Environmental hazard awareness
- Cybersickness prevention guidance
- Comfort and accessibility settings

## 5. Best Practices by Documentation Type

Best practices for XR products vary depending on the type of documentation.

### 5.1 Spatial Interaction Documentation

Documentation should clarify inherently three-dimensional concepts:

- Distance and depth cues
- Reach and interaction zones
- Occlusion and collision behavior

- Gesture recognition feedback
- Volumetric UI distribution

### 5.2 Accessibility in XR Documentation

Documentation must explicitly address accessibility:

- Alternative input methods
- Non-color-dependent signaling
- Visual equivalents for audio cues
- Motion sensitivity accommodations
- Seated and standing mode support
- Text legibility and audio descriptions

### 5.3 Performance and Optimization Guidance

Performance expectations should be clearly segmented by modality and device:

- VR frame rate targets, commonly ranging from 72–120 fps depending on hardware and configuration (see this video for more details).
- MR and AR performance constraints, which may prioritize stability and power efficiency over high refresh rates
- Graphics quality tiers; device-dependent and important performance
- Network requirements for multiplayer experiences
- Device generation differences

## 6. Standards and Guidelines

Where relevant, documentation may reference:

- Emerging ISO and IEC guidance related to immersive systems
- Vision-based spatial registration and tracking standards
- UL and regional safety standards for XR hardware
- Platform-published design and accessibility guidelines

All standards references should clearly indicate whether guidance is normative, advisory, or emerging. Standards for emerging tech are subject to change and evolution, and should be regularly reviewed to ensure documentation stays consistent with any changes.

## 7. Common Documentation Gaps and Solutions

**Gap:** Spatial interaction concepts are underexplained.
**Solution:** Add dedicated spatial interaction sections with diagrams and in-experience examples.

**Gap:** Platform differences are blurred.
**Solution:** Explicitly segment documentation by platform and capability.

**Gap:** Setup guides assume prior expertise.
**Solution:** Include beginner-friendly walkthroughs with visuals and troubleshooting.

**Gap:** Safety guidance is fragmented.
**Solution:** Centralize safety documentation with clear warnings and best practices.

# 8. Emerging Trends in XR Documentation (2025–2026)

The following trends (as per this [2025 mid-year report)]() reflect early signals and industry direction rather than established standards:

- Adaptive, context-aware documentation that adjusts based on user behavior and experience level
- Multimodal learning combining text, audio, video, and spatial visuals
- Increasing experimentation with voice-assisted help systems, particularly in VR
- Greater regulatory and organizational emphasis on explicit safety and comfort disclosures

# 9. Recommendations for Technical Writers

Tech writers can develop skills, documentation processes, and collaborative practices that directly impact their contributions to XR product documentation quality:

## 9.1 Skill Development

- Spatial design literacy
- Hands-on platform experience
- Knowledge of accessibility best practices
- Safety and compliance awareness
- XR-focused user research methods

## 9.2 Documentation Process

1. Experience the product in XR
2. Observe real users of the product
3. Plan in-app experience and external content together
4. Iterate in-device when possible
5. Validate accessibility
6. Test on multiple types of XR hardware

## 9.3 Collaboration

- Work closely with UX and engineering
- Involve safety and compliance stakeholders early
- Establish reusable documentation templates
- Align documentation updates with product releases

# 10. Strategic Implementation Framework

Success documenting XR products hinges on four interdependent pillars that technical writers must master:

**In-Experience Supremacy**: Primary guidance must reside within the XR environment itself through spatial tooltips, contextual help panels, voice-activated assistance, and haptic feedback patterns, reducing cognitive friction compared to external references users cannot easily access mid-experience.

**Platform Fidelity**: Cross-platform documentation must precisely delineate Unity XR Interaction Toolkit workflows from Unreal OpenXR implementations, HoloLens spatial anchoring from Quest passthrough behaviors, and ARCore plane detection from Vuforia image targets, eliminating dangerous ambiguity across ecosystems.

**Heuristic Transparency**: Design recommendations (comfortable viewing distances, 72-120fps targets, locomotion options) must be clearly positioned as platform-derived heuristics rather than universal mandates, directing users to authoritative sources for device-specific requirements.

**Universal Accessibility**: Every interaction pattern must offer input-agnostic alternatives (gaze/voice for motor-impaired users, seated modes for motion-sensitive individuals, high-contrast audio-supported UI for visual/hearing impairments), transforming compliance from checkbox to core design principle.

# 11. Measurable Impact Areas

The practices described in this report are associated with **observable improvements** across XR products when evaluated through qualitative research, usability testing, and operational metrics. Rather than asserting fixed numerical outcomes, impact should be assessed using context-appropriate measures aligned with product goals.

## Comprehension and Learnability

Spatial demonstrations and in-experience guidance support faster understanding of XR interactions by allowing users to learn through action rather than recall.

**Indicators to measure:**

- Time to complete first successful interaction
- Reduction in repeated onboarding prompts
- Fewer clarification requests during early use
- Usability test observations showing correct gesture execution without external reference

Well-designed spatial instruction shifts learning from memorization to embodied understanding, which is particularly important for gesture-driven and controller-based interfaces.

## Comfort, Safety, and Session Continuity

Clear documentation of spatial boundaries, locomotion options, and comfort settings helps users manage physical and cognitive load during XR sessions.

**Indicators to measure:**

- Early session abandonment rates
- Frequency of comfort-setting adjustments
- User-reported discomfort in post-session surveys
- Support requests related to motion sickness or disorientation

Documentation that normalizes comfort customization and explicitly addresses safety considerations can reduce friction during first-time use and encourage longer, more sustainable engagement with XR hardware and software products.

## Adoption and Ongoing Use

Accessible, intuitive onboarding and help systems lower barriers for first-time users and support broader adoption, particularly in enterprise and training contexts.

**Indicators to measure:**

- Completion rates for onboarding or tutorial flows
- Time to independent task execution
- Reduction in training support interventions
- User confidence ratings collected during pilot programs

In organizational XR deployments, documentation quality directly affects training efficiency and perceived value, influencing whether XR tools are adopted, expanded, or abandoned.

## Documentation as a Feedback Surface

XR documentation also serves as a diagnostic tool for product maturity.

**Indicators to measure:**

- Areas where users consistently request clarification

- Help content invoked most frequently during sessions
- Discrepancies between intended and observed interaction patterns
- Documentation updates driven by usability findings

Tracking these signals helps teams iteratively refine both the product and its documentation, reinforcing documentation as an active component of the XR system rather than a static artifact.

# Conclusion

XR documentation represents a transformative departure from traditional software documentation practices, demanding innovative approaches that account for the unique spatial, embodied, and safety-critical characteristics of immersive experiences. Unlike conventional 2D interfaces where users navigate static text, images, and hyperlinks, XR environments require documentation that lives within three-dimensional space, responds to user gaze and gestures, and provides immediate, contextual feedback without breaking immersion.

Technical writers must shift from explanatory text to demonstrative experiences, replacing lengthy prose with spatial annotations, animated gesture guides, and progressive onboarding sequences that teach through doing rather than reading. This embodied approach acknowledges that XR users process information through movement, depth perception, and multi-sensory cues rather than linear reading flows. Safety considerations elevate documentation from convenience to necessity, as poor guidance risks physical harm through spatial collisions, cybersickness, or inaccessible controls.

***A call to action for technical writers:*** Embrace XR documentation as a spatial design discipline requiring hands-on device experience, cross-team collaboration with UX engineers, and continuous validation against real user behaviors. By internalizing these demonstrative, context-aware, user-centered principles, technical writers become pivotal architects of successful XR adoption, bridging the gap between complex immersive technologies and seamless human understanding across VR, AR, and MR ecosystems.

# XR Documentation Bibliography


- 5K and 120Hz, if You've Got the Hardware for It — HTC Vive Pro 2 Review (GamingTrend, 2025).
- Develop with Unity for Android XR (Android Developers, 2025).
- FIX YOUR FPS TODAY! Quest 3, Pimax, Bigscreen + MORE! (YouTube, 2025).
- Mixed Reality Documentation (Microsoft, 2026)
- Meta Developer Documentation (Meta, 2026)
- Overview of Niagara Effects for Unreal Engine (Epic Games Docs, 2025).
- Pokémon Go Gets an AR Boost on Android with ARCore (The Verge, 2018).
- Spatial UI | XR Headsets & Wired XR Glasses (Android Developers, 2025).
- Standard For Safety For Virtual Reality, Augmented Reality and Mixed Reality Technology Equipment (SCC CNN, 2023)
- Unity Development for HoloLens (Microsoft, 2024)
- Virtual Reality and Technical Communication (ClickHelp Technical Writing Blog, 2021).
- What's New in ARCore (Google Developers, 2025).
- XR (Unity Manual, 2026).
- XR & AI Trends 2025 Mid-Year Report (Yord Studio, 2025).
- XR Interaction Toolkit | 0.9.4-preview (Unity Docs, 2019).
- XR Interaction Toolkit | 3.0.10 (Unity Docs, 2025).
- XR Trends 2026: The Future of AR and VR for Business (Yord Studio, 2025).

# Appendix A: Scope and Assumptions

## Scope

This report focuses on **best practices for technical writers documenting XR products**, including virtual reality (VR), augmented reality (AR), and mixed reality (MR) experiences. It is intended to support writers working on:

- End-user documentation for XR applications and devices
- In-experience onboarding, tutorials, and help systems
- Developer-facing documentation for XR platforms, SDKs, and frameworks
- Safety, accessibility, and usage guidance related to immersive experiences

The guidance applies primarily to **software-driven XR products**, including applications built with common engines and frameworks, and does not attempt to provide exhaustive coverage of hardware engineering, regulatory compliance, or clinical or medical XR use cases.

## Out of Scope

This report does **not** aim to:

- Define or mandate regulatory or legal compliance requirements
- Serve as a substitute for platform-specific SDK documentation or safety manuals
- Provide detailed implementation instructions for XR development workflows
- Establish universal performance, comfort, or safety thresholds across all devices

Where platform guidelines, standards, or heuristics are referenced, they are included to inform documentation strategy, not to define binding technical or legal requirements.

## Assumptions

This report assumes that:

- Readers are technical writers, documentation strategists, or content designers with baseline familiarity with software documentation practices

- XR platforms, SDKs, and device capabilities are evolving rapidly, and documentation must be treated as a living system rather than a static artifact

- Platform-provided guidance may change over time; readers are expected to consult current official documentation for version-specific details

- Many recommendations reflect **design heuristics and industry practices**, not universally enforced standards

Where future-facing trends or emerging practices are discussed, they are presented as **directional observations**, not predictions or guarantees of adoption.

## Intended Use

The guidance in this report is designed to:

- Improve clarity, usability, and safety of XR documentation
- Help technical writers adapt documentation practices to spatial and immersive contexts
- Support collaboration between docs teams, design, engineering, and compliance/legal

Readers should adapt these practices to their specific product, audience, platform, and organizational constraints.