Building Apps for Vision Pro

An introduction to the concepts behind spatial computing

Presented by: Rodney Aiglstorfer
LinkedIn: https://www.linkedin.com/in/raiglstorfer

About Me

- **Global Nomad**: Lived in 6 of the 7 continents, 4 before college.
- **Education**: BS Computer Science from Drexel University; with a minor in Japanese.
- **Early Career**: Worked in Kyoto, Japan, as a CIA intelligence asset, working for early stage Japanese internet startups. Appeared on cover of Wired Japan as the creator of “Bob & Angie’s Kitchen”.
- **Early Entrepreneur**: Started first company while still in college - employed fellow students to build CD-ROM content for the adult entertainment industry.
- **Most Known For**: Creating the Starbucks app, Orbitz.com and pioneering mobile banking and check deposit.
- **Startup Success**: Created and sold 2 startups; the 1st pioneered mobile banking and the 2nd made aviation maintenance software built on blockchain.
- **FIRE’d**: Achieved financial independence by age 30 (via traditional investing), and retired early at 38 (after first startup exit)

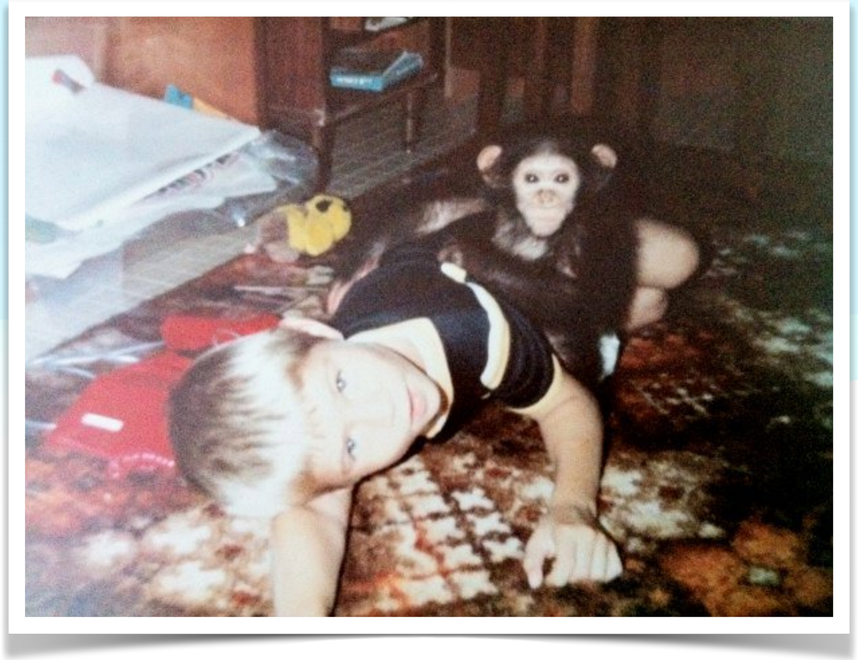
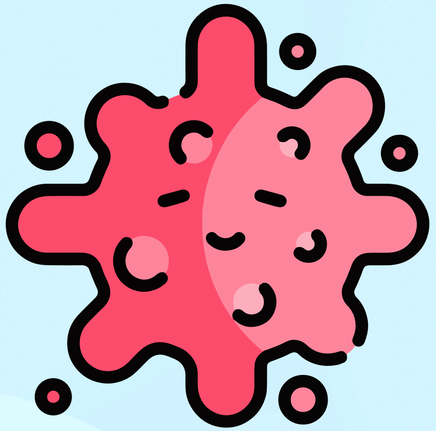


Photo: Rodney while living in the Congo with his pet Chimp “Sofie”

What I’m doing now …

- **Cancer**: DX’d Stage IV colon cancer in 2022, came to NYC for treatment … now in early stages of remission
- **ChemoBuddy**: Started working on app when DX’d - to help me track and analyze my chemo regimens so I could better understand what was causing my side effects and share that info with my Oncologist
- Initially created for personal use - but has since generated a lot of “grassroots” interest from the larger cancer community
- **HacktiveHealth**: I’m starting a new startup, to bring ChemoBuddy to market and to create innovative digital health solutions in the area of **AI predictive analytics** and **remote patient monitoring**
- I’m now in closed Beta with plans to launch an open Beta later this year.

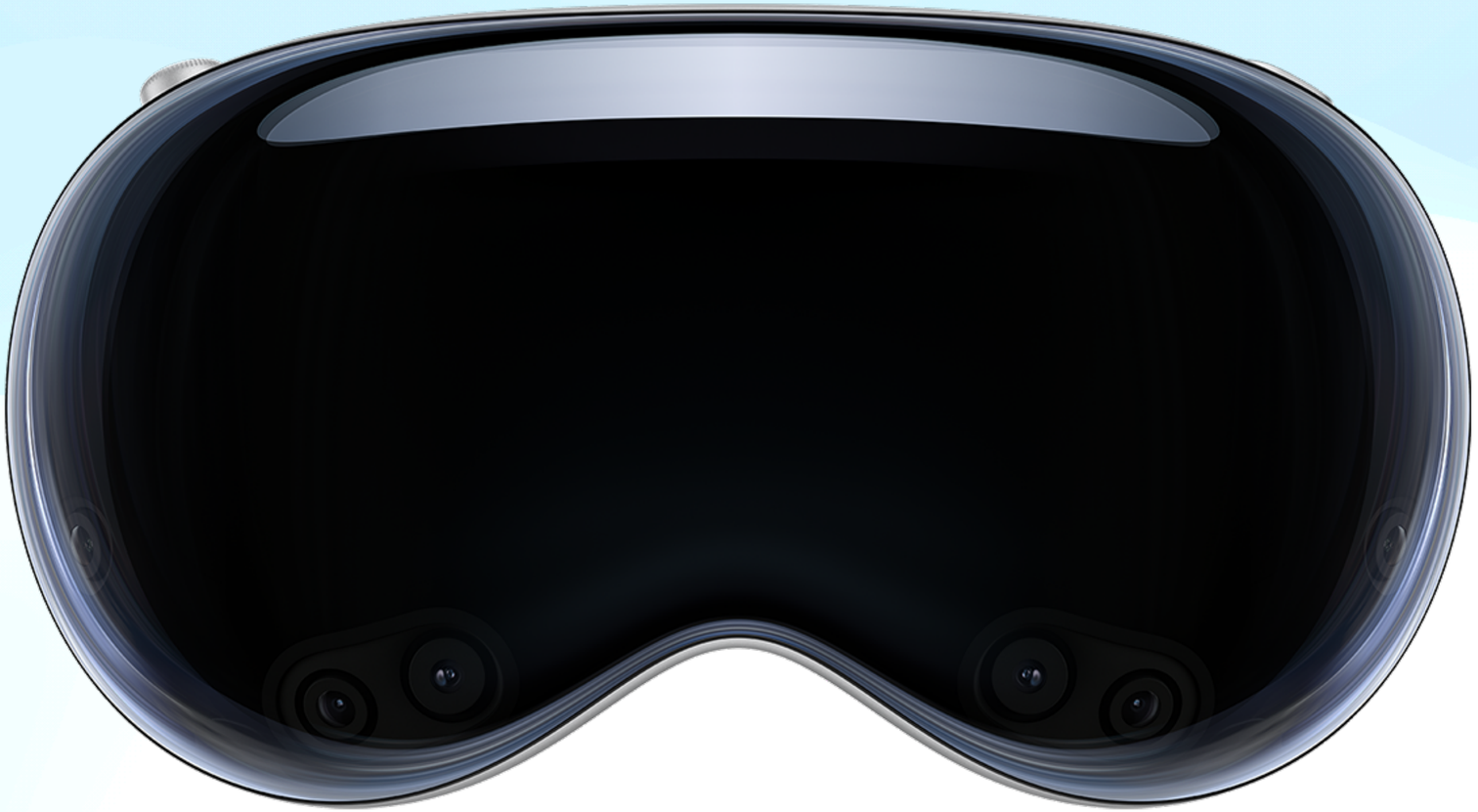


ChemoBuddy

https://chemobuddy.app

What is “Spatial Computing”

A Primer for Developers

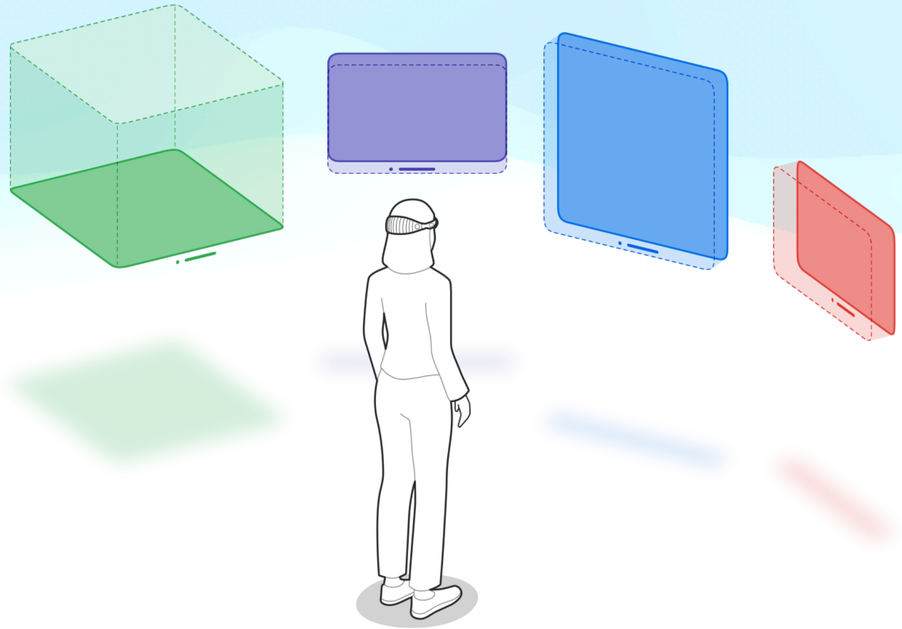


https://developer.apple.com/visionos/

What is “Spatial Computing”

A Melding of AR and VR concepts

- **Also Known As**: VisionOS and xrOS
- **Limitless 3D Space**: Apple Vision Pro provides a huge persistent area for exploring and creating in 3D.
- **Two Ways to Use**: Choose to stay aware of the real world while using apps (AR), or get fully lost in a virtual world (VR).
- **Easy Switching**: Move smoothly from a normal screen view to a full 3D experience and back.
- **Shared Experiences**: Two or more people can “share” the same experience from different POVs



Building Blocks of Spatial Computing

Windows

Volumes

Spaces

Building blocks of spatial computing

Level 1 - “Windows”

- **Most Familiar**: Easiest way to start developing for VisionOS.
- **Multiple Windows**: Less like iOS and iPadOS, and more like MacOS, apps can have multiple windows.
- **SwiftUI Framework**: Utilizes familiar SwiftUI for constructing windows with traditional views and controls.
- **Enhanced Depth with 3D Content**: Option to add 3D elements to enrich the user experience.
- **Compatibility**: Existing iPad and iOS apps can be run as Windows without any additional work.

Building blocks of spatial computing

Level 2 - “Volumes”

- **3D Volume:** A finite 3D volume for added depth.
- **Versatile Viewing:** Create experiences viewable from any angle in Shared Space or an app's Full Space.
- **Minimal Environment Interaction**: Some detection of lighting sources but no ability to interact with your physical environment
- **Compatibility**: 3rd party 3D platforms like Unity and Spline have support for Volumes

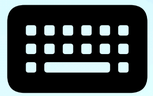
Building blocks of spatial computing

Level 3 - “Spaces”

- **Multi Modal:** Can contain Windows, Volumes
- **Unbounded 3D Content**: Can fully react to the external environment (tables, chairs, walls, floor)
- **Shared Spaces**: The default modality where content exists side by side with other apps
- **Full Spaces:** a dedicated space where only that app’s content will appear (only option for ARKit)

Spatial Computing Input Options

Eye Tracking



Keyboard

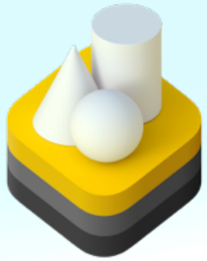
Mouse

Gestures

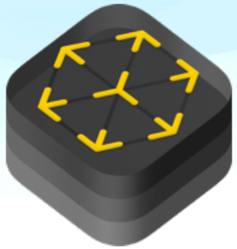
Controller

Spatial Computing Frameworks

SwiftUI

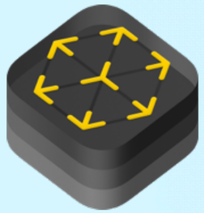


RealityKit



ARKit

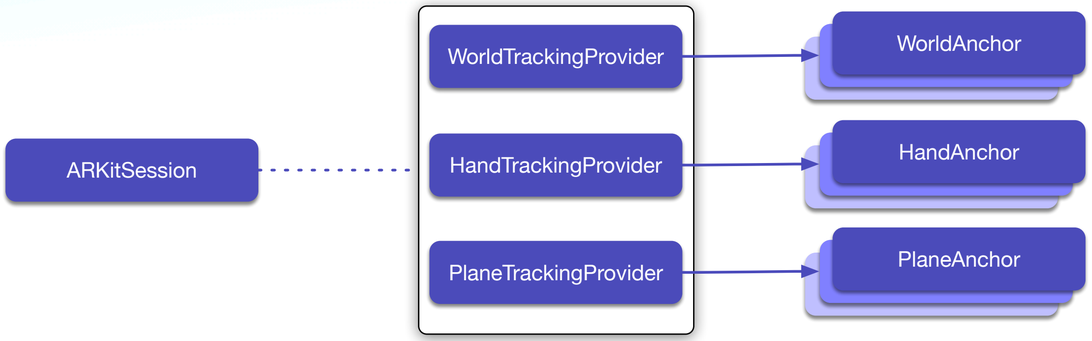
Metal

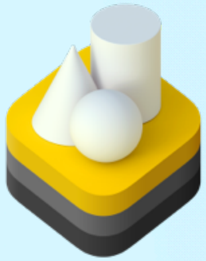


ARKit

Augmented Reality Framework

- **Languages**: in Swift and C
- **Privacy-First**: Must enter a **Full Space** No data to the shared space. Permissions needed.
- **World Tracking**: Persisted world anchors *relative* to app origin (and geographic locations)
- **Scene Understanding**: Plain detection (.wall, .floor, .ceiling, .table, .seat, .window, .door). Reference image detection. Scene geometry & mesh anchors
- **Hand Tracking**: Skeleton, joints and Chirality all have achors





RealityKit (and Metal)

3D Rendering and Animation

- Works well within ARKit but not exclusive to spatial computing
- For realistic rendering, animating, and simulating 3D models
- Used mostly to create 3D games and animated effects in Apps
- Supports open standards for shaders, models, particle emitters, and screen description

Reality Composer Pro

Metal

Universal Scene Description

MaterialX

Core Tools

Dev Tools	 Xcode	 Unity
Content Tools	 Reality Composer Pro	 Spline
		 Blender

Where to go from here?

Resources for Developers

- Apple Developer - https://developer.apple.com/visionos/
- Philly CocoaHeads - https://www.meetup.com/phillycocoaheads/
- VisionsList - https://www.visionslist.com/tutorials/

Vision Pro Demo

Questions?

Presented by: Rodney Aiglstorfer
LinkedIn: https://www.linkedin.com/in/raiglstorfer