

"The fun and challenging part of our job as designers in VR is to understand where the real world trumps the virtual, and where the virtual one can intersect it in awesome ways." – *Goodbye Metaphors, Hello Worlds*

THE UX OF VR

a VR Mini-guide, history, and exploration for dfs3

Benjamin D. Gibson

**2015------2017**

# Jul 2015: VR-ui FUNDAMENTALS in *Eve Valkyrie* [[src]](http://atomhawk.com/news/post/dan-talks-ui-vr-develop-2015)

* Instead of text, favor icons and animated graphics to guide gaze, draw a line going out of view
* Instead of cluttering user’s view, have UI only appear when necessary or gazed at intentionally
* Test with the winning entry for Best Jargon Award, “VRgins” (used in 3 talks here + a [hashtag](http://twitter.com/hashtag/vrgins))

# oct 2015: VR-UX seminal metrics (Mike alger, google vr) [[SRC]](https://youtu.be/id86HeV-Vb8)

## Spatial metrics (click images to be linked to fuller-res)

### pitch – VR ZONES (note: up > down)



### yaw – VR ZONES



### DEPTH – VR layers (think visual frame of reference)



### flat projection of all angles together (“your VR UX Real-estate”)



*Misc. aside: we assume clipping things = bad, but what if we use it to our advantage as a VR-UI conceit?*

# **Oct 2015 (GDC): Interaction Design in VR, Lessons from *the Lab*** [**[src]**](https://www.youtube.com/watch?v=_vQo0ApkAtI)

* **Advised *against* rotating while teleporting*,*** to preserve user’s visual frame of reference.
	+ Helps to hint to the user where they’re pointing, even when outside their field of view.
* If representations are less abstract/stylized, users expect concrete, real-world behaviors.
	+ Contrast the hands from *The Climb* against the cartoon gloves in *Job Simulator 2050*.
	+ **However**, aping the real world too literally can invite its flaws.
* **Keep Your Promises:** If an affordance hints “pick me up,” it better behave as such!
* **Real Interactions:** the fully pulled door lever example leading to fatigue on repetition.
	+ Beware too literally mimicking reality, as this can invite its limitations unnecessarily.
	+ Always question whether a gesture only exists for its own sake, or has consequences.
		- E.g. swishing a potion cauldron gesture to better mix items but possibly spill.
* **“Haptic Pulses” Feedback Tricks**
	+ Apply when entering and exiting interactive objects to convey a Hovered state.
	+ Rising haptic pulse intensity emitted every few units in drawing bow **lent it “texture”!**
	+ “Phone ring” approach to signal when users should look at their wands, cf. *Tilt Brush.*
* **The “Invisible Gorilla” Problem:** the more we engage users, the more they’ll end up missing!
	+ (Might actually be worth exploiting in certain scenarios!)

# Nov 2015: “The swayze effect” $\rightarrow $ Oculus story studio on *lost, henry* [[Src]](https://www.oculus.com/story-studio/blog/the-swayze-effect/)

* A separate talk calls this effect “The Ghost’s Dream,” also based on the namesake movie here.
	+ The problem: being in the world but not of it—characters behave, but do not react.
* In *Henry*, even without an identity established for “you” in the narrative, the look-at proved effective for many users (while at the same time contradicting the being lonely premise).
	+ Some argued no identity being needed was just because users were happy to take part

# **Jan 2016: msdn channel9 on vr environments and interactions** [**[SRC]**](https://channel9.msdn.com/blogs/misslivirose/Designing-for-VR-Environments-and-Interactions)

* On the one hand, VR users *expect* 360-content, else “why is this in VR” criticism can crop up
* On the other hand, Alger’s angles above and others corroborate that quickly gets exhausting
* No body > Wrong body (uncanny valley)
	+ “Heads and hands” armless examples: Oculus Rooms, BigScreen Beta, Dragon Front

# **mar 2016: deep dive on button UI in VR** [**[SRC]**](http://realityshift.io/blog/buttons-in-virtual-reality-a-ui-ux-design-approach)

* Large enough to find, not too close to one another, preferably with a brief caption
* Fuse buttons (what I call gaze-timer buttons) should never move (ex. rising eyes in *Samsara*)
* Fuse button countdowns should have a fill animation
	+ Along the circumference for a rounded button
	+ From one side to another for rectangular buttons (bottom-up or left-to-right)
* Hover states should do something on enter and exit
	+ Change the cursor, making it more pronounced
	+ Colorize the UI element, if it’s grayscale otherwise (not on top of fill + caption + cursor)
* Some developers like to place a “hide/show UI” toggle fuse button under the player’s feet
	+ Though this can be overlooked, so on other headsets it might be better to use a button

# Mar 2016 (GDC): Welcome back to geocities, VR (*bazaar, Spore*) [[SRC]](https://youtu.be/VtCuLd3wpEw?t=39m46s)

 - Have an NPC look at the player then somewhere else to get player to follow their gaze

 - Identifies the core of *Minority Report*’s problem: a lack of language to make its usage straightforward

 - The "nodding" input problem – too prone to a wide range of interpretations on what a nod means.

 - The magic carpet solution for locomotion

 - Thinking about what the avatar looks like in a reflective surface, due to the 1:1 expectations in VR.

 - **Don't design for what VR can do**, but solve the problems of users and VR in general (sickness, etc).

# May 2016: Two VR User TESTing tips [[Src]](https://www.slideshare.net/uxbri/steve-bromley-running-user-tests-for-vr-games)

* Don’t expect testers to be able to hear you when you talk after they’re in the headset
* Don’t expect useful opinion data, either ask nuanced questions or do full-on quantitative tests

# jun 2016: Goodbye metaphors, hello worlds [[SRC]](https://yvr.design/goodbye-metaphors-hello-world-interface-design-in-virtual-reality-d2462e52a029)

* "The fun and challenging part of our job as designers in VR is to understand where the real world trumps the virtual, and where the virtual one can intersect it in awesome ways."
	+ i.e. Just making it all be literally how we'd do something in real life is dumb, but current metaphors for 2D also break down.
* Try an interaction off-computer first, for physical realsies.

# **jun 2016: Vive-Specific TIPS** [**[SRC]**](http://www.doolwind.com/blog/virtual-reality-development-tips/)

* **Certain Vive buttons are easier to use than others, from best to worst:**
	+ Triggers
	+ Track-pad as single button
	+ 3D world interaction (e.g. press in a 3D button with wand)
	+ Menu hamburger button
	+ Track-pad directions as separate buttons
	+ Grip buttons – refrain from use unless absolutely needed
* **For room scale experiences, aim for 2mx1.5m or smaller to capture** [**81% of users**](http://steamcommunity.com/app/358720/discussions/0/350532536103514259/)
* Avoid anything that causes wand to in close towards the HMD
* But on the fade to black tip with teleporting, I think it's better to only do this if it's actually changing areas (like the bubbles in *The Lab*, but see prior bullet), else user loses visual frame of reference.
	+ Now on the other hand, an RE7-style vignette circle on teleport **might** **compromise** the two, when I think about it (if you must go this route).

# **JUN 2016: mid-early vr’s Design patterns** [**[src]**](https://www.youtube.com/watch?v=wZQB1705inM)

##  Selection Design Patterns

1. Cursor/Reticle with Hovered State
* Option 1: have [the cursor](https://cdn-images-1.medium.com/max/800/1%2Afst6G5cqowQtN6uuBSED-g.gif%20and%20https%3A/cdn-images-1.medium.com/max/800/1%2AtfUhTwslfHdh0-aEBbO3gg.gif) react.
* Option 2: have [the interactive object hovered over](https://cdn-images-1.medium.com/max/800/1%2Azhr81Fv0ugSlD4BNnCfLuw.gif) react.
* Option 3: have the cursor replaced [with relevant 3D objects](https://cdn-images-1.medium.com/max/800/1%2Ajk6WNMVi0jHCoP2HLYoUSw.gif), perhaps after a grab.
1. Fuse Button (esp. for zero/few-button HMDs, what I call a gaze-timer)
* Comes with the Unity VR sample framework
* Can cause problems if another trigger appears where the gaze-timer was upon completion
1. Ground-level Menu (i.e. around user’s feet)
* Be cautious of fatigue (see Mike Alger’s max pitch angle).
* Does make good on the curiosity zone aspect though.
1. Controller-based Menu (see *Tilt Brush*)
2. Dashboards (see Oculus Home)
* Important to have a non-blank room behind the dashboard for visual frame of reference.

## Locomotion Design Patterns

1. Teleportation: either arc-based (*The Lab*) or even projectile-driven (*Budget Cuts*).
2. Waypoints: either fading-and-jumping or linearly interpolating between them (most tour apps).
3. Snow-globes: preview of destination inside a globe or helmet or portal.
* Sometimes the snow-globe is fixed and you move into it: portals in Cosmic Trip.
* Sometimes the snow-globe is grabbed and you move it onto yourself: bubbles in *The Lab* (triggers level load), the helmet *Fantastic Contraption* (real-time).
1. Free-roam: back at this point in time he notes the walk-in-place system (fatigue, feels dumb).

## Text Display

1. Head-fixed overlays (see “Newspaper” below).
2. AR-in-VR hover tips: labels floating a fixed offset as a child of the parent object.

# **JUL 2016: object weight in vr** [**[SRC]**](http://www.roadtovr.com/b-reel-simulating-object-weight-mass-virtual-reality-motion-controllers/)

* Tests the technique of using forces / velocity adjustments gravitationally pulled towards controller, but **only** on pickup -- direct-parents upon reaching a wand.
* **However**, the test concludes that when having to pick up lots of things, direct parenting immediately without the loose-link stuff wins *by far* in player feedback.
* That said, the loose pickup (when it includes a threshold to snap the link at, conveyed by a scaling haptic vibration going 0-100% with tension) *can* *work* *if used sparingly* on infrequent special objects.
* **Sound played a key factor in perceived weight**: landing with a thud SFX vs. a grass-tuft SFX.

# **AUG 2016: iron man vr – motion sickness gravity trick** [**[src]**](https://yvr.design/dont-teleport-fly-b175a5c8fd73)

* **KEY**: eliminating anything causing the camera to pitch/roll.
	+ **NOT** by controlling the virtual camera, but the virtual space/physics around it.
* **1st** moving Iron Man hands to only apply thrust to what I'd assume is the bottom-center of the avatar (while still making fire/sound emit spatialized to the hands to fake that it came from there).
* **2nd** seemingly by preventing the capsule collider from pitching upon collision with other objects, e.g. flying into slanted walls: [seen here](https://gfycat.com/BlaringTemptingKawala) (notice how the right side never pitches, even though the gray boot that I think represents the thrust does).

# sep 2016: Vr skybox designs and unity implementations [[src]](https://medium.com/aol-alpha/how-to-design-vr-skyboxes-d460e9eb5a75)

 - Sky’s important for sense of scale, and cannot be ignored in first-person VR experiences.

# **sep 2016: vr human interface guidelines** [[src]](http://vrhig.com)

 - Contains a solid list of current trends whose names are kind of useless to list here, and see the PDF!

# oct 2016: google sdk tips on spatial audio [[src]](https://developers.google.com/vr/concepts/spatial-audio)

 - No time to read it all, but I caught snippets that building on DFS1 ideas

 - e.g. animate 3D sound emitters to speed up users picking it out, and the types of sounds to use.

# Oct 2016: on VR & ACCESSIBILITY, by Accessibility guru ian hamilton [[Src]](http://www.gamasutra.com/blogs/IanHamilton/20161031/284491/VR__accessibility.php)

* It's sim sickness (body stationary, visuals move), being opposite of motion sickness (as in cars).
* **It's not possible for everyone to get used to it**, though some will, *and many who could won't care to go back for another try after getting sick once.*
* No blur effects in the sprint-dash, which notably Raw Data indeed doesn't do -- instead just UE4's trailing beam-particle VFX to fake it!
* Text readability: either offer resizing controls or **at least do not hard-code a fixed offset** that will prevent leaning forwards to read it.

# **nov, may 2016: two deep dives on throwing in vr**

* <http://blog.mattnewport.com/hand-tracked-controls-in-vr-throwing/> (May 2016)
* <http://www.gamasutra.com/blogs/CharlieDeck/20161118/285808/Why_Throwing_in_VR_Sucksand_How_to_Make_it_Better.php> (Nov 2016)

# **Nov 2016 (Vrdc): Lessons Learned from VR Prototyping at Google** [**[SRC]**](https://www.youtube.com/watch?v=G295PAPzZX8)

 **2. Spatial audio is more than just dynamic volume (it’s also a slight delay between ears, front/back, and beyond that letting the room’s physical properties use echo/reverb).** But it just amounts to using the provided spatial audio plugins.

 6. Not every experience will be better in VR.

 7. Don't be constrained by reality.

 8. Design ground-up for-VR.

 11. Be creative with teleportation preview (Budget Cuts) and other motion types (FC helmet).

 **12. Narrow the FOV while moving. (Some players seem to hate it, e.g. in *Eagle Flight.*)**

 13. Teleport to a psychologically safe space

 14. Players love experiences that allow their creativity.

 15. Let them throw it! **(Rather than letting go and it freezes midair.)**

 16. Think about gestures.

 17. Let sloppy actions have structured results--be smart detecting player intent.

 18. GIVE YOUR PLAYERS **SUPERPOWERS.**

 19. Use snapping to constraint motion. (cf. UE4's editor update!)

 20. Keep interactions in comfortable zones.

 21. Controls don't have to be mapped 1:1, e.g. go-go-gadget-arms extending as you reach out.

 22. Use haptics with audiovisuals to complete immersion (see above pulse tricks).

 23. We love HUGE spaces.

 24. Divide world into smaller spaces fitting into a typical physical space. (cf. *Cosmic Trip*!)

 25. Place objects in a way that keeps players in their play space.

 26. Texture **EVERYTHING**, *even if it's just a subtle noise texture.*

 27. **Show the chaperone bounds, even when they are not near it, e.g. on the floor.**

 28. Everyone **LOVES** particles--PUT THEM **EVERYWHERE**.

 29. Accuracy of scale matters, another reason to prototype ASAP.

 30. Use scale to convey power OR vulnerability.

 31. Scale as a tool, e.g. chess piece put on dollhouse to zip inside it for viewing.

 32. When you change scale, scale physics and audio with it

 33. Players expect things to happen when they touch each other! **(Always done first.)**

 34. Don't reward bad/anti-social behavior, e.g. fading gray and vanishing from world.

 35. Reward pro-social interactions, e.g. spark showers on high-fives, but not face punches.

 36. Enforce social boundaries. **(Always done second: clipping avatars inside each other.)**

 37. Voice chat multiplies immersion.

 38. Put shared objects at the center of the space.

 39. Think about asymmetric experiences, a la Keep Talking and Nobody Explodes.

 40. Avoid full-body avatars in first-person view, in Google's experience. Can have them for others to see you by, but **make Owner No See**, as they orient worse and worse the further down you look.

 41. Unblinking eyes on avatar are bad! "Eye-brain plugin" blinks randomly and eyes follow gaze’s turns.

 42. When you pick up an item, the item can **become your controller!**

 43. Beware uncanny valley.

 44. Avoid UI requiring you to turn your head (see Mike Alger metrics).

 45. Not too close, not too far (see Mike Alger metrics).

 46. Keep the text BIG.

 47. A well-designed "elbow model" heuristic can make interactions more natural. Like the "neck model," assuming our HMD pivot is actually back along the local forward vector slightly from the queried HMD-position transform value.

 49. Check out what others are doing. //You do not have to go down all dead ends yourself!

 50. You are a pioneer, think big!

# dec 2016: FB.design’s VR-UI Unity Template Project, Hand renders [[src]](http://facebook.design/vr)

* [Unity Template](http://facebook.design/vr-template): Drop-in StreamingAssets of Unity template to test UI flows, even on GearVR
* [Hand Pose Renders](http://facebook.design/virtual-hands): for Oculus Touch storyboarding/mockups

# **FEB 2017: The Frontiers of VR UX (vr market & ui approach)** [**[src]**](https://www.youtube.com/watch?v=-f2OgAyjDVU&list=PLA27jL5mOsuXdOup1ejEJ3h1L1cWTgGJr)

## I: The VR Market's hardware/input is fragmented to all hell and back.

1. Personas, user maps/flows, journey/treasure maps, etc. from non-VR UX in a new context. The major difference is that users will expect more realistic, natural input, and in split-up market that means that rapid prototyping’s mandated.
2. **VR Market as a whole**: illustrates interesting patterns of fragmentation.
* Mobile: Gear VR, Daydream (succeeding Cardboard)
* Tethered: PSVR, Rift, Vive
1. **VR user market data for 2016**: shows that mobile far outstrips tethered when it comes to hardware units sold, software’s more frequently purchased for tethered VR. If we can assume this reflects non-VR demographic traits, it makes sense that there are less frequently-playing gamers than there are casual mobile players—but the latter play less, and thus buy less software as a result. (Strategy Analytics)
* Hardware: 87% mobile, 9% console, 4% PC
* Software: 46% console, 31% PC, 23% mobile
1. **VR Industry Sectors**: “In which primary industries do you think VR use will significantly accelerate in the next 2 years?”
* The answers included: Entertainment (84.4%), Education (74%), Media (62.5%), Medical/Healthcare (60.4%), High-tech (56.3%), Travel (54.2%), Arts (52.1%). (Yeti.co 2016)
* *Hotly discussed*: Architecture (VR sims), Professional Training (cf. Lowes, or Ericsson at job fair), Marketing (VR experiences promoting brands), and again Entertainment.
1. **VR Developers**: Are we developing for the platforms that consumers are most interested in?
	1. VR Game Studios: “For which VR platform(s) is your company actively making games for?”
	🡪 Gear VR and PSVR both got about half as many responses as Rift/Vive. (GDC Europe 2016)
	2. VR Non-Game: “If you started a VR project now, which platform holds most potential?”
	🡪 Less than a quarter of the developers answered Gear VR + PSVR, in total. (Yeti.co 2016)

🡪 *Actually no,* developers appear more interested in high-end PC VR, mismatched with the pro-mobile/console consumer install base. **However, vitally**, this may drive consumer demand assuming said developers create killer apps for tethered PC VR as a result of their focus on it.

1. **VR Input Methods:** *gaze is the core interaction verb across every platform right now.*
* It’s worth noting that microphone for voice/speech recognition’s near-consistently supported.
* Beyond these two, input schemes still just as fragmented as the above platform market.
* A warning – if your user’s not a hardcore game, blind use of a gamepad’s out of the question.

CONCLUSION: settle on a specific platform instead of a universal approach, and prototype iteratively.

## II: so what do we do as interactive (i.e. not 360-video) UX designers?

* Tools: Unity & Unreal
* Inspiration: Reality (3D Disciplines: Automotive, Interior, and Furniture Design, or Architecture)
* Simulation Sickness: never move camera, make use of grounding elements in the environment.
* **Two VR-UI Types:**
1. "Newspaper" (Head-Fixed), e.g. Iron Man – moves with your gaze.
2. Diegetic – breaks immersion far less, no screen clutter’s generally more comfortable.
* **Diegetic UI Tip #1:** *Integrate with Spatial Scene Flows*
1. What info do we want to present to the user on entry into a scene?
2. What info do we want to present when they leave?
3. What about the points they could move to between 1 and 2?
* **Diegetic UI Tip #2:** Prototype to standardize interaction distance and font size conventions.
* **Kinesthetic Satisfaction**: why players try to throw everything in the scene around.
* **Square UI to Sphere UI**
	+ Top-priority information goes in the center, not the top-left.
	+ Lower-priority information pushed to more peripheral zones (reasons to look around!).
		- *"OK, I need to do that, but it's not as critical as what's in the middle."*
* The User’s Play Space – Real *and* Virtual
	+ Don’t make them crane their neck if they aren’t in a non-swivel chair.
	+ Player height: let them enter a value, or at least select sitting versus standing.
* The more real the scene feels, the easier it gets to break that immersion—i.e. uncanny valley.
	+ Recommends stylized for VR because it also benefits performance and tends to lessen the visual load on the user’s cognition taking it all in (esp. if also tasked to meet goals).
* **UX designers in web/mobile are used to feedback that audio’s actually annoying to users.**
	+ Here, it’s vital to use audio/dialogue to not have to jam as much into visual feedback.
	+ E.g. Vanishing Realms’ lever going thunk, spin, and slam behind you instead of a ding.
* User Testing: get the mirror mode output *and* footage of the user.
	+ *“Users of Asian descent tend to be statistically and genetically predisposed to simulation sickness, which can help for worst-case analysis.”* **(?)**
	+ Heat-mapping solutions really help, e.g. Unity Heatmaps Plugin for Unity Pro
	+ No equivalent solution nor plans for it in UE4, despite it having been present in UE3!

# Final and misc. notes on UX: FORGET THE RULES

## Recurring themes

* Theme 1: Reality vs. Virtuality tensions (“Less is more” – visually, emotively, and physically)
	+ The Realism Expectation and Uncanny Valley vs. Stylized Aesthetics
	+ Inviting Real-world Limitations vs. Superpowers
	+ 360-content, *Minority Report*, and physics-objects’ kinesthetic joys vs. Fatigue
		- Perhaps more succinctly “fun UX” vs. “effective UX”
* Theme 2: Things everyone always does
	+ If hands: Grab/throw anything and everything as well as shove it against all else
	+ If social: Touch each other, stand inside each other
* Theme 3: The ease in/out transitions to/from engaging content
	+ Some kind of airlock-type screen before starting the main simulation
	+ Some level of ability to interrupt anywhere
	+ A final “airlock” when quitting back out

## Misc. Notes, Personal stances

* ***Minority Report***: Everyone really likes to take shots at it, back from Alger in 2015 up into 2017.
	+ It feels reminiscent of the snap reply to a “wouldn’t it be cool if” suggestion.
	+ As with that scenario, I want to explore the seed behind the request in DFS3.
	+ "Make players do something PHYSICS-related!" vs. UI/UX’s demand: clarity & laziness
* **Locomotion**: I see the “solution” mirroring my experiences in discrete math class—no solve-all.
	+ Instead, this is at its best if integral to the theme, and thankfully, games have those!
		- The world-to-you solution in *Lone Echo* for space stations
		- The magic carpet solution in *Bazaar* for mobile fits the Arabian Nights theme
		- The teleporting in *Budget Cuts* as spy tech, *Psychonauts VR* as mind-hopping
	+ But consider adding all possible options vs. unfair gameplay advantages (ex. of [debate](https://www.reddit.com/r/oculus/comments/5yf0ga/regarding_locomotion_and_why_options_arent_always/))
* **Presence**: [GDC 2012: Forget 'immersion' -- player attention is what matters](http://www.gamasutra.com/view/news/164914/GDC_2012_Forget_immersion__player_attention_is_what_matters_says_Lemarchand.php) (note, pre-VR!)
* Hover-cast Finger-blade: <https://github.com/aestheticinteractive/Hover-UI-Kit> (Leap Motion)
* Prototype Techniques
	+ Physical: saw one video of a developer holding a laser pointer to your head and videotaping while another developer holds out strips of text to represent buttons.
	+ VR-UI Paper Prototypes & Size Mockups In-browser Toolset: <http://prototool.co>
* Personal jargon: airlocks, gaze-timers, snow-globes.
	+ I’ve also liked newspaper UIs, VR-gins in the talks above.

# Future To-See

* <https://uploadvr.com/five-ways-to-reduce-motion-sickness-in-vr/>
	+ Limit movements in directions besides the player’s forward direction
	+ <https://uploadvr.com/introducing-limbo-a-vr-camera-movement-technique-by-the-developers-of-colosse/> ?
	+ No accelerations
	+ No camera-instigated yaw (as opposed to player-instigated)
	+ Safety cages
* [Cardboard Design Lab](https://play.google.com/store/apps/details?id=com.google.vr.cardboard.apps.designlab&hl=en)
* [Google I/O 2015 Livefeed - Designing for virtual reality](https://www.youtube.com/watch?v=Qwh1LBzz3AU)
* [Mitch's UE4 VR Lab](https://www.youtube.com/channel/UChvlNUgZKmEd-Gul_Tdv8Uw) **YouTube Series**
* [Introduction to VR UI in Unity](https://www.youtube.com/watch?v=NYoqUomgTGU) **Video Tutorial**
* [UE4 HTC Vive - How to interact with a menu using Motion controllers](https://www.youtube.com/watch?v=cyqgKZ7O-Sc&feature=share)