middleware for immersive VR system

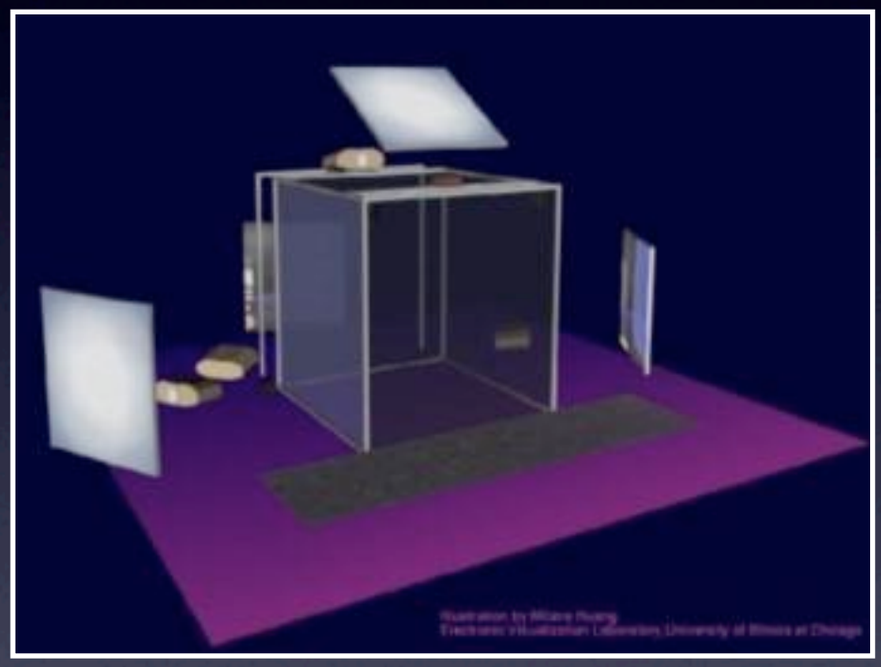
Shi Yin

once upon a time..

- First CAVE was built in EVL in1992

- CAVElib is the software library used by CAVE

- For years, CAVE and CAVElib were dominant in VR community



CAVE Automatic Virtual Environment

courtesy of evl

until one day..

- CAVElib was commercialized in 1996
- Organizations started to build their own in-house software libraries
- VR Juggler, FreeVR, CalVR

CAVE in work
courtesy of wikipedia

one.VR Juggler

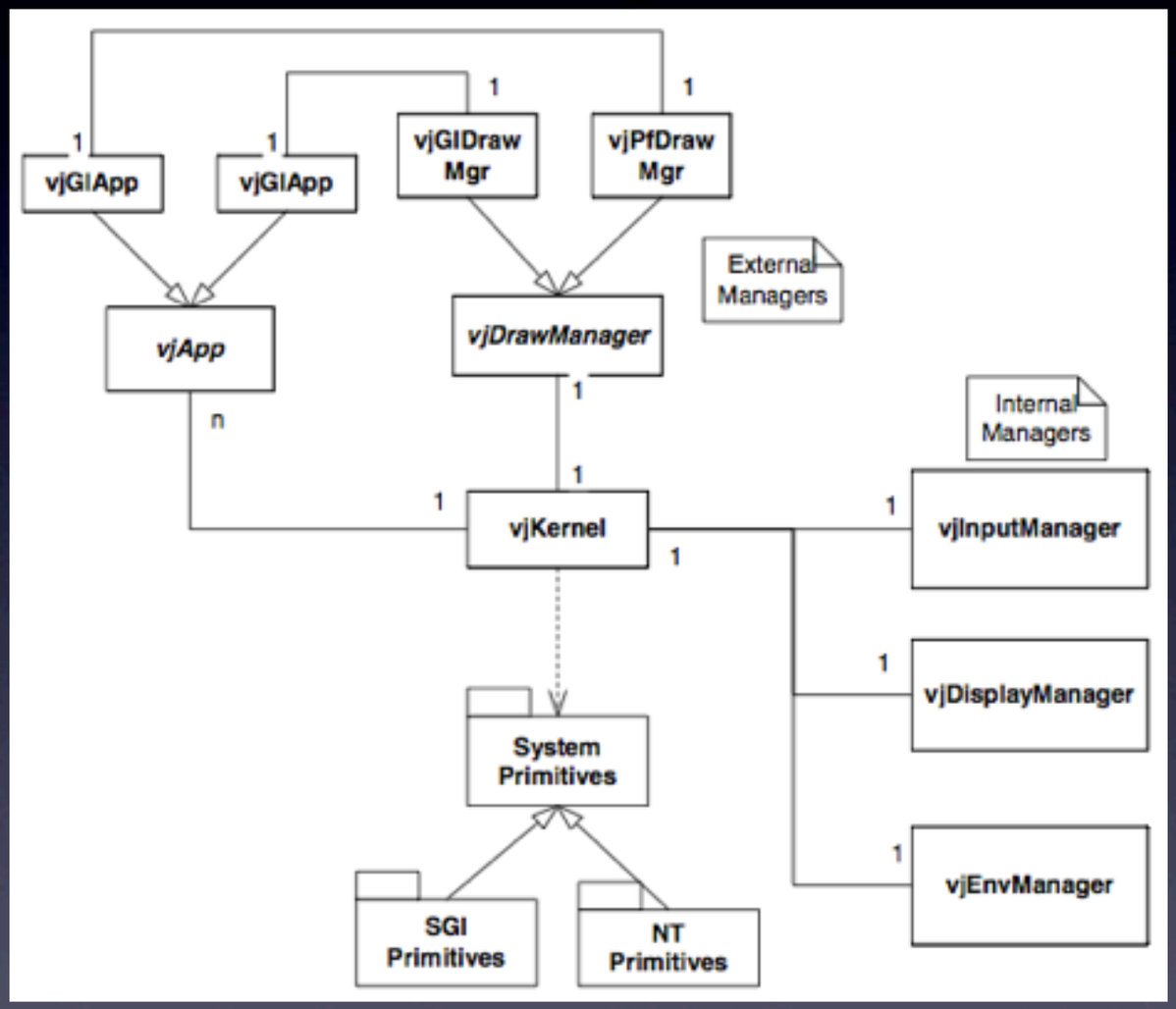
- virtual platform
- architecture
- evaluation

virtual platform

- A VP enables researchers to concentrate their efforts on the content of the application
- provides abstraction, scalability and flexibility
- allows run-time changes of configuration
- runs multiple applications simultaneously
- tools for evaluating and tuning performance of the application

architecture - microkernel

- kernel object
- internal managers
  - input manager
  - display manager
  - environment manager
- external managers
  - draw manager
- application interfaces
  - application objects



microkernel architecture
[Bierbaum et al. 2001]

evaluation

- developers would only need to focus on implementing application interface
- running application as object
  - run multiple at the same time
  - add or remove during run-time
- decoupling between different components provided portability, scalability and flexibility

two. FreeVR

- standard features
- unique features
- future work

FreeVR

- primary goals
  - keep the library as simple as possible, avoiding complex dependencies
  - permit application developers the freedom to choose VR contents (world simulation and rendering) of their own preference
  - keep existing applications alive for as long as possible, allowing VR practitioners to continue to benefit from experiences of history

standard features

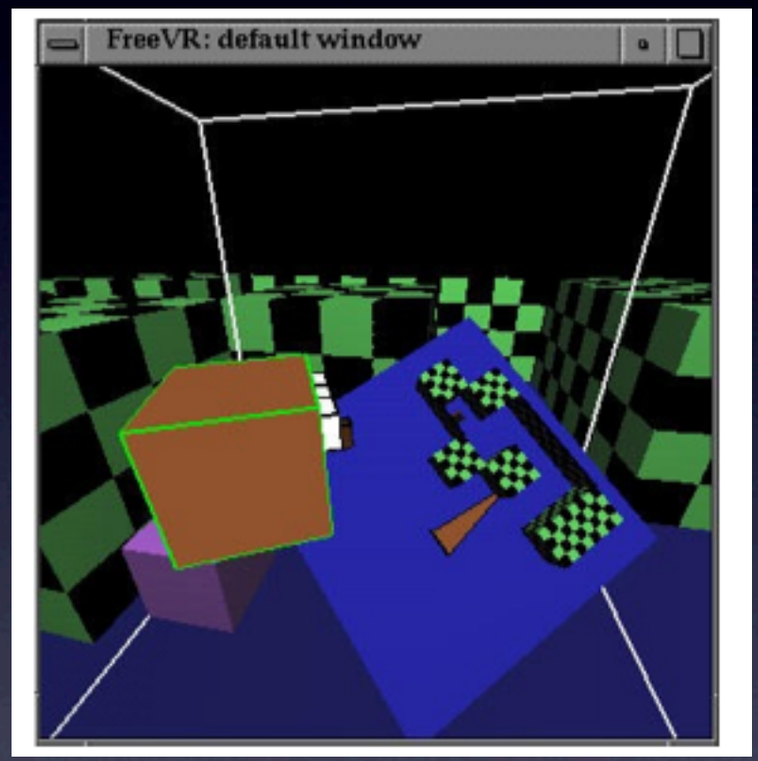
- multiprocessing
- flexible configuration
- support for most major input devices
- (later) clustering

unique features

- multi-user capabilities
  - a concept called *eyelist* to naturally handle multiple-user scenarios
  - not very clearly stated
- inputs in the rendering
  - a lesson learned during the long course of development
  - input should only be read in the world-simulation portion of the application

unique features (cont'd)

- desktop simulation interface
  - developers would not have to sit in CAVE to test their applications



desktop simulation
[Sherman et al. 2013]

future work

- yes, future work
- make some new components more robust and elegant
- interface to low-cost off-the-shelf tracking devices and stereo TVs

three. CaIVR

- standard features
- unique features

CaIVR

- from Calit2: California Institute for Telecommunications and Information Technology
- one of the newest and most advanced software framework
- motivations
  - solve issues that did not handled well by first and second generations of virtual reality libraries (VR Juggler, FreeVR)
  - realize new functionality that is necessary for today's VR systems and is difficult to add into old libraries without changing inner code

standard features

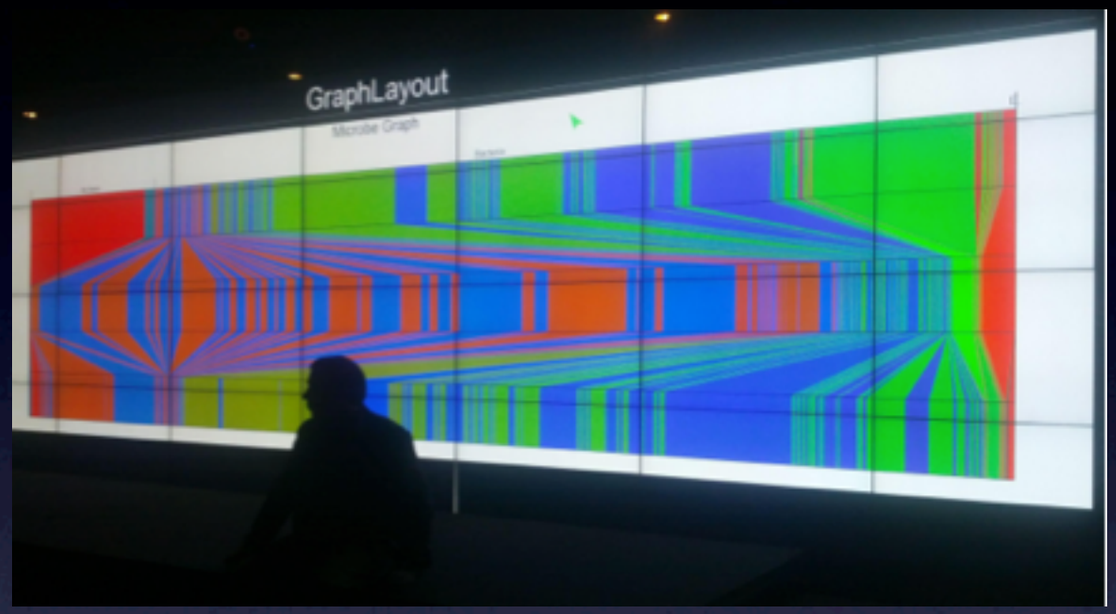
- standard features since day one
  - hardware driver interface to different input devices
  - OS interface to all display devices
  - multiprocessing support
  - configuration for different VR system
- standard features later on
  - cross-platform support
  - object-oriented class hierarchy similar to VR Juggler
  - support for distributed/clustered computing system
  - configuration information stored in a single file

unique features

- “SceneObject” class
  - This class handles navigation, interaction and context menu of a specific object in the scene.
- four rendering modes on top of OSG
  - auto-stereoscopic mode
  - anti-aliased mode
  - omnistereo mode
  - a mode in which head position can be specified differently for each screen

unique feature (cont'd)

- support for 2D monoscopic display
- add/move applications during run-time
  - detail not included
  - can't see why unique
- an integrated menu system
- track multiple users, each with multiple input devices
- others..



display wall at Calit2 & menu system
[Schulze et al. 2013]

ever after..

- principle stays the same
- features evolve
  - old standard feature keeps
  - old unique feature becomes standard
  - old unique feature becomes nothing
  - new unique feature appears
- library can’t live forever
  - no matter how much effort was put into making a library be popular longer, it can’t be popular (or even alive) forever
  - some new requirements be met only by totally new systems

references

- [1] Bierbaum, A., Just, C., Hartling, P., Meinert, K., Baker, A., & Cruz-Neira, C. (2001). **VR Juggler: a virtual platform for virtual reality application development**. *Proceedings of Virtual Reality 2001 Conference* (VR '01) (pp. 89–96).
- [2] Sherman, W. R., Coming, D., & Su, S. (2013). **FreeVR: honoring the past, looking to the future**. In M. Dolinsky & I. E. McDowall (Eds.), *IS&T/SPIE Electronic Imaging* (pp. 864906–864906–15). doi:10.1117/12.2008578
- [3] Schulze, J. P., Prudhomme, A., Weber, P., & DeFanti, T. A. (2013). **CalVR: an advanced open source virtual reality software framework**. In M. Dolinsky & I. E. McDowall (Eds.), *IS&T/SPIE Electronic Imaging* (pp. 864902–864902–8). doi:10.1117/12.2005241

Thank you!

questions?