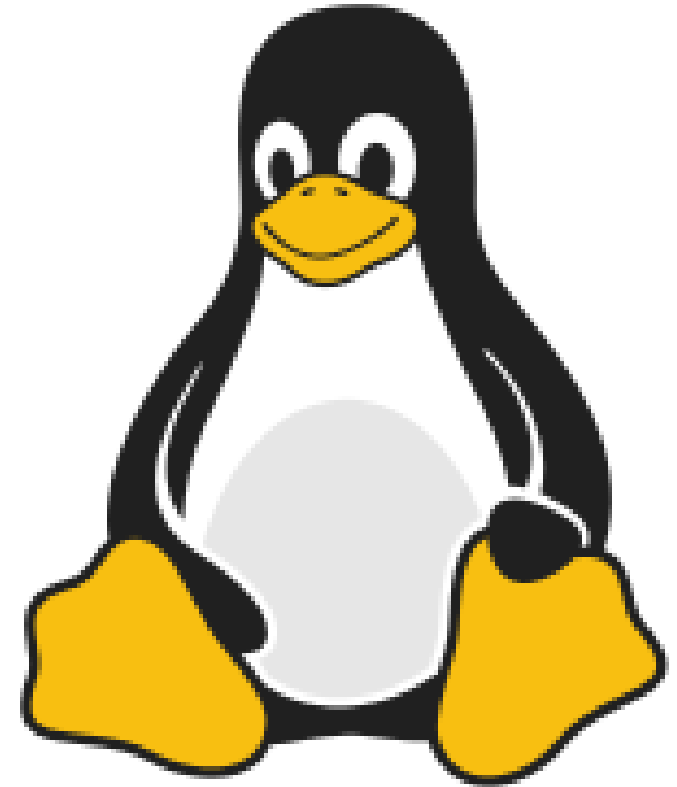




WSLG

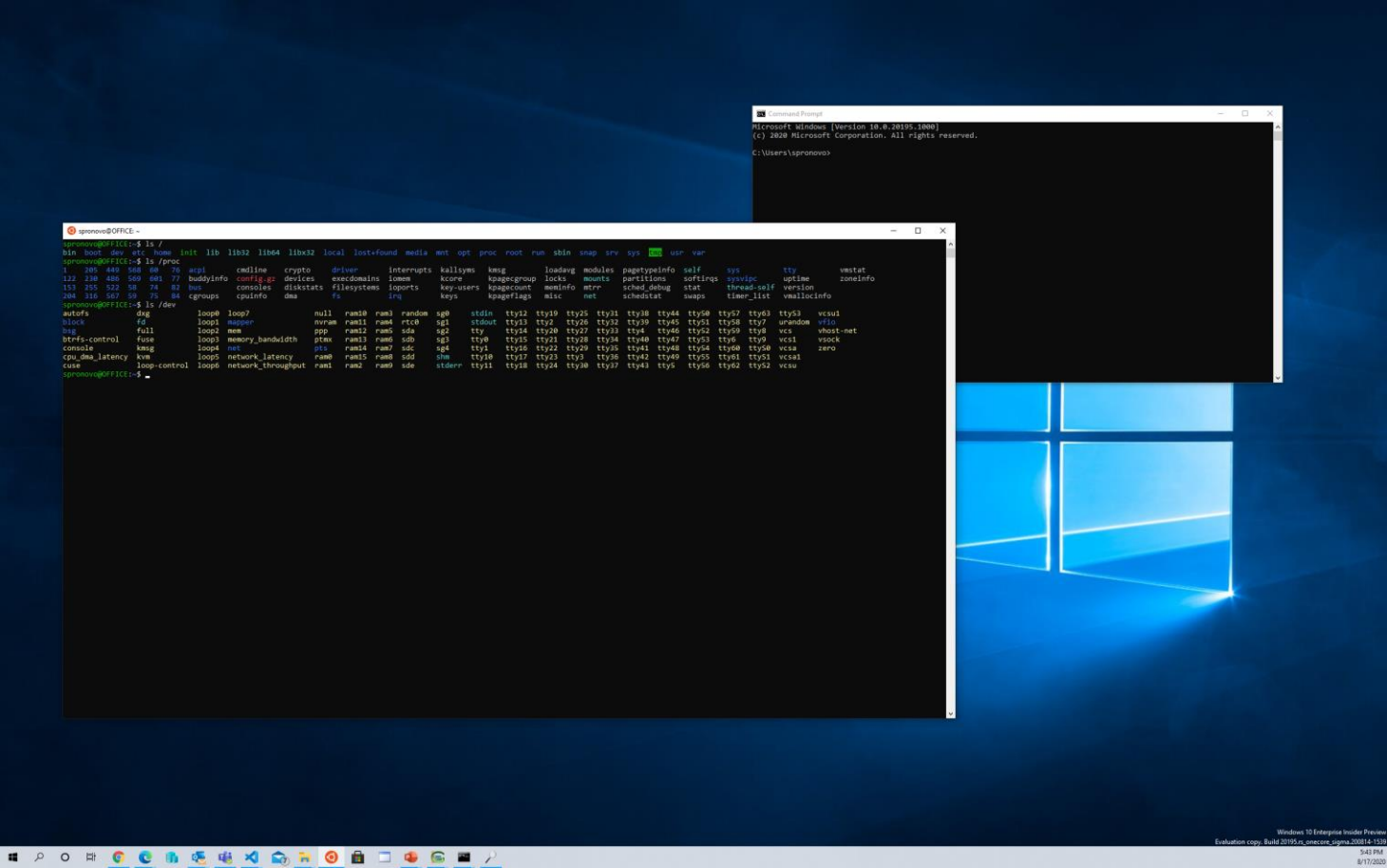
X11 and Wayland  
Applications in WSL



# What is WSL?

- **Windows Subsystem for Linux**
  - Infrastructure to run Linux applications inside of Windows
- For more context on WSL see our other XDC talk
  - *WSL – Graphics Architecture*

# WSL Today – Terminal Experience



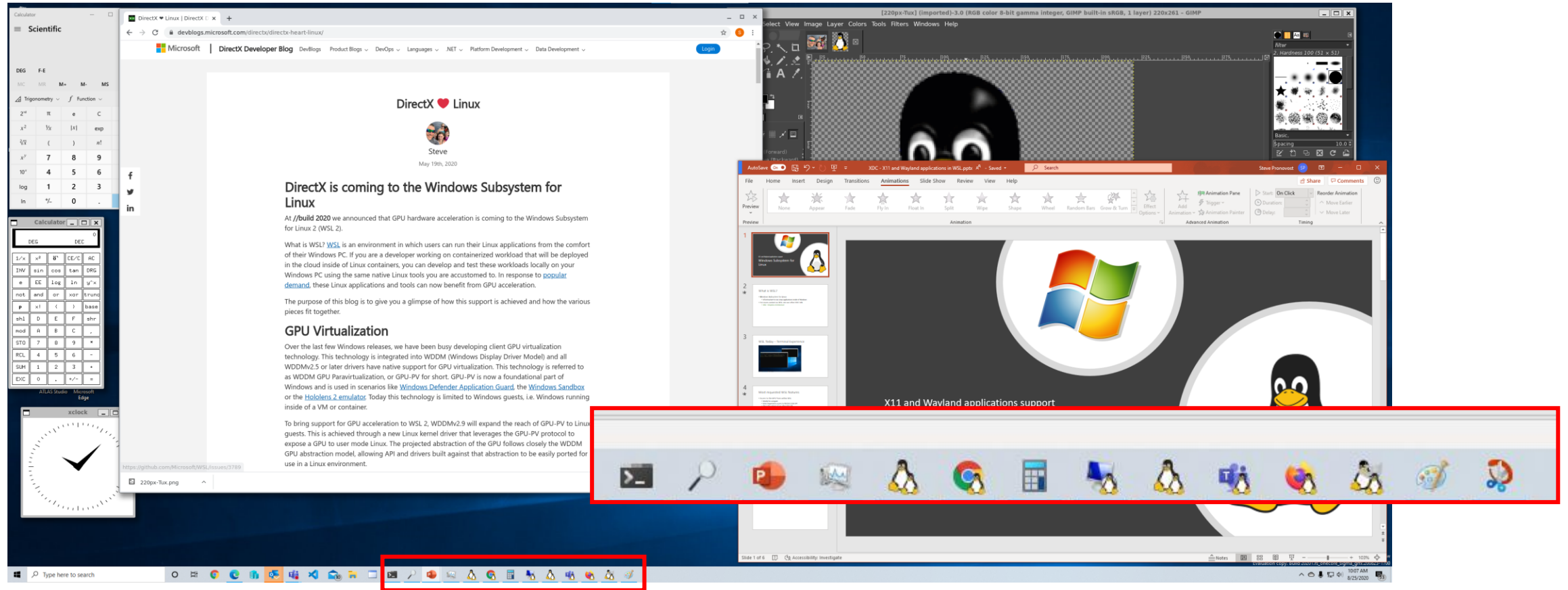
# Most requested WSL features

- Access to the GPU from within WSL
  - Mostly for compute
  - Most requested is access to NVIDIA CUDA API
  - This is the subject of our other XDC talk
    - *WSL – Graphics Architecture*
- Ability to run GUI applications
  - Going beyond a terminal only experience and the ability to run X11 and Wayland applications
  - *Subject of this talk*

# But wait...

- You can run X11 applications in WSL 2 today!
- You can run an X Server in WSL 2
  - Run an X Client on Windows and connect to WSL 2
  - Lots of instructions online and great videos on YouTube on how to set this up
- It works... but experience is not great
  - Fundamentally a desktop in a window experience
  - Performance is not great
  - No Wayland app support
  - Doesn't work out of the box

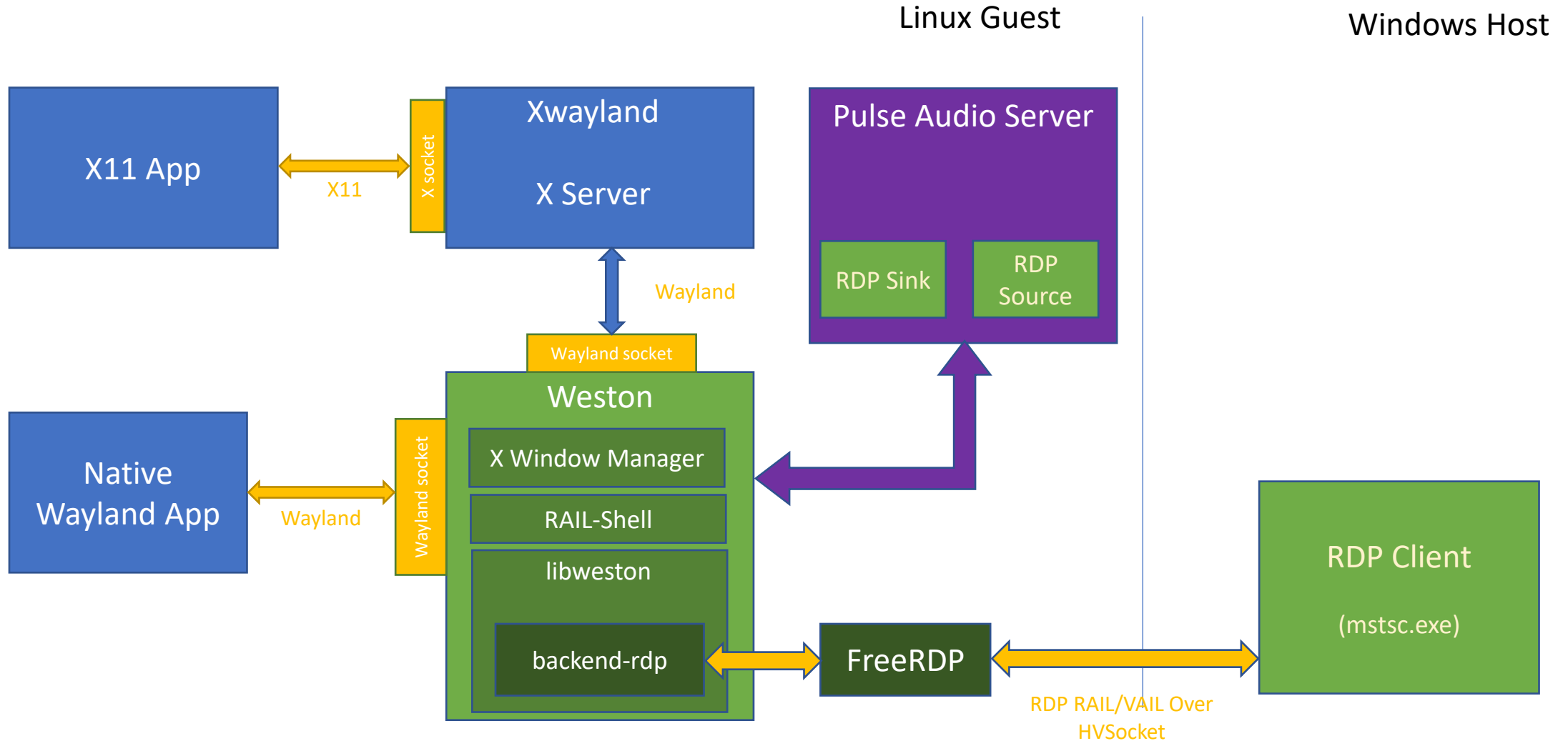
# Linux GUI App Integration (WSLG)



# What's our approach?

- Building on Weston
  - Reference/sample compositor for Wayland
- Extending Weston existing RDP backend and teach it about remoting application
  - RAIL – Remote application integrated locally
  - VAIL – Virtualized application integrated locally
- Leveraging Pulse Audio for audio in & out
  - Bridging over to RDP
- Building with intent of contributing back

# WSLG Overview

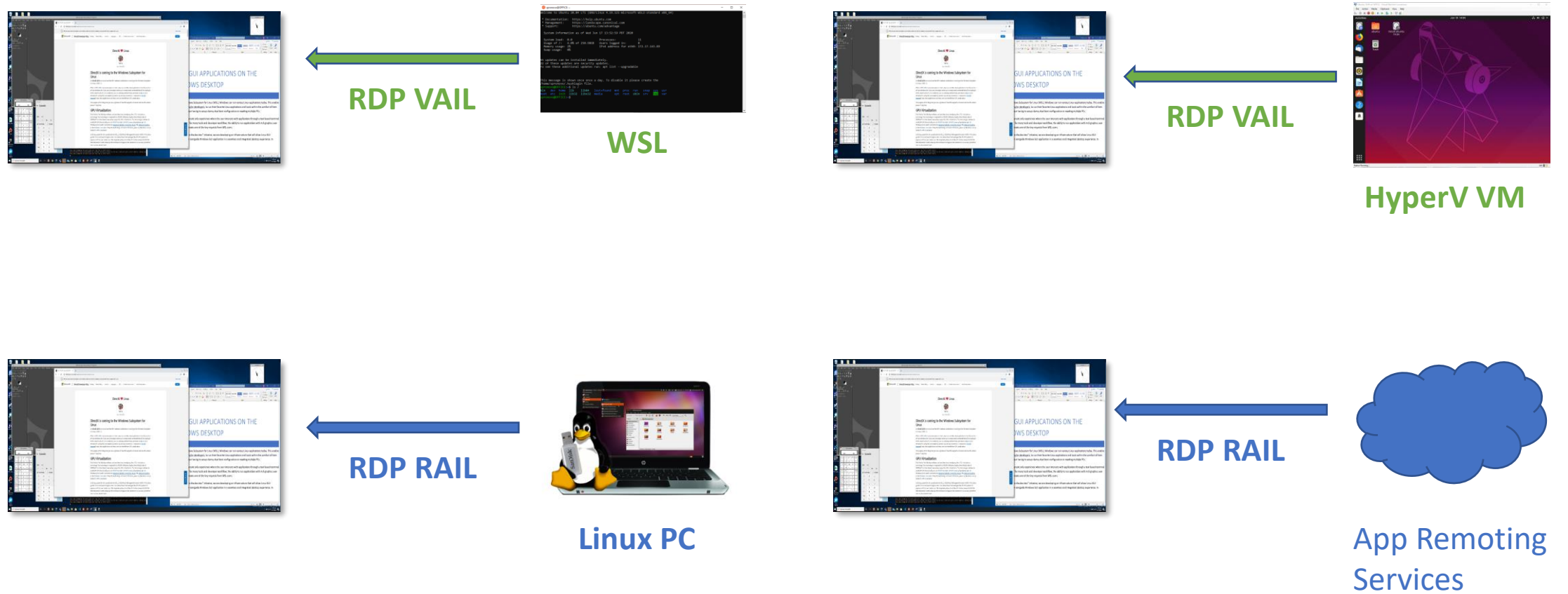




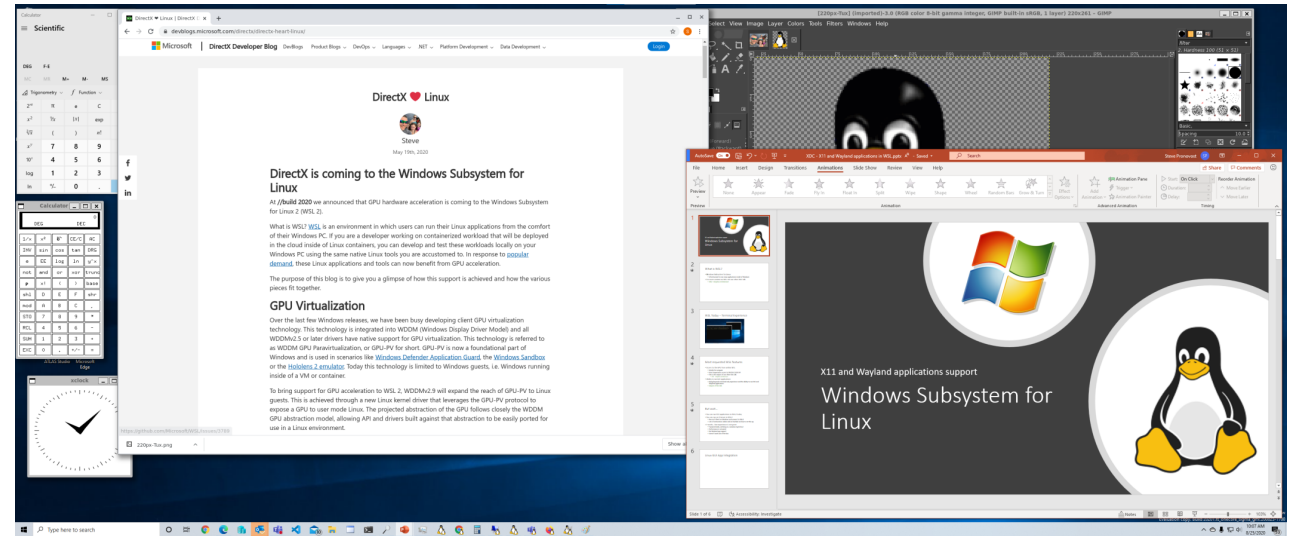
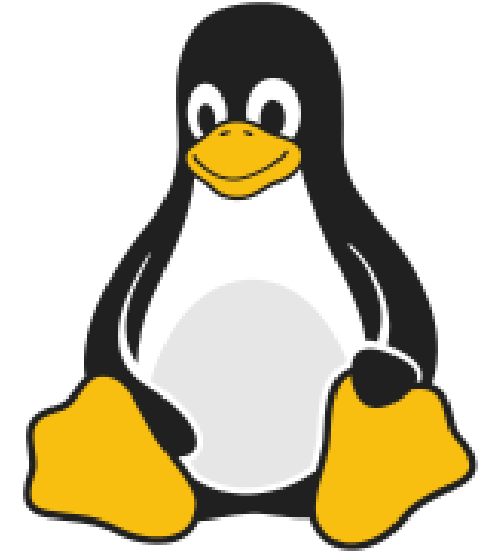
# Why based this on RDP RAIL/VAIL?

- Mature technology and host side client
  - Been remoting applications through RDP for years
  - Have mature business built on this technology (Azure WVD)
  - Host side client is same irrespective of guest type
  - Built-in network transparency and performance scaling
    - Over the network -> RAIL -> Pixels copied over the transport
    - Over VM boundary -> VAIL -> Shared guest <-> host memory.
  - Enable remoting of both X11 and Wayland applications
  - Documented & Open Source implementation available
    - Continue to use FreeRDP with Weston

# Built for remoting transparency



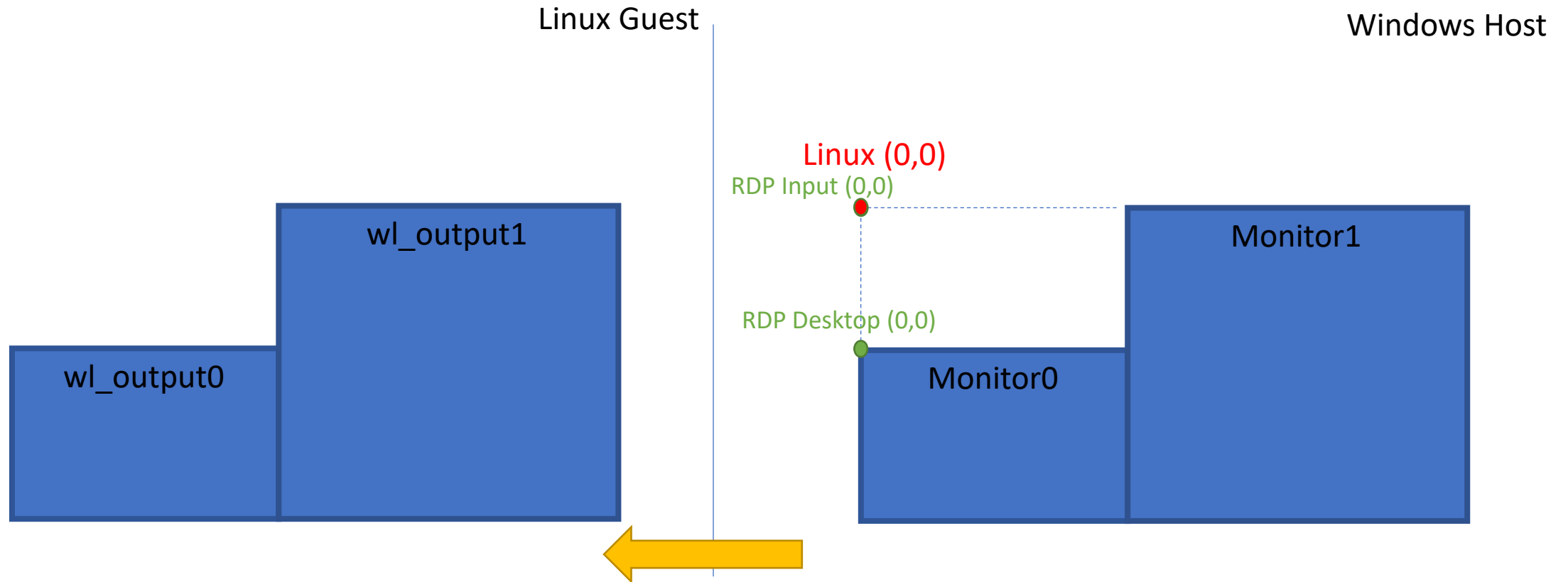
# RAIL & VAIL Overview



# Mirrored Desktop

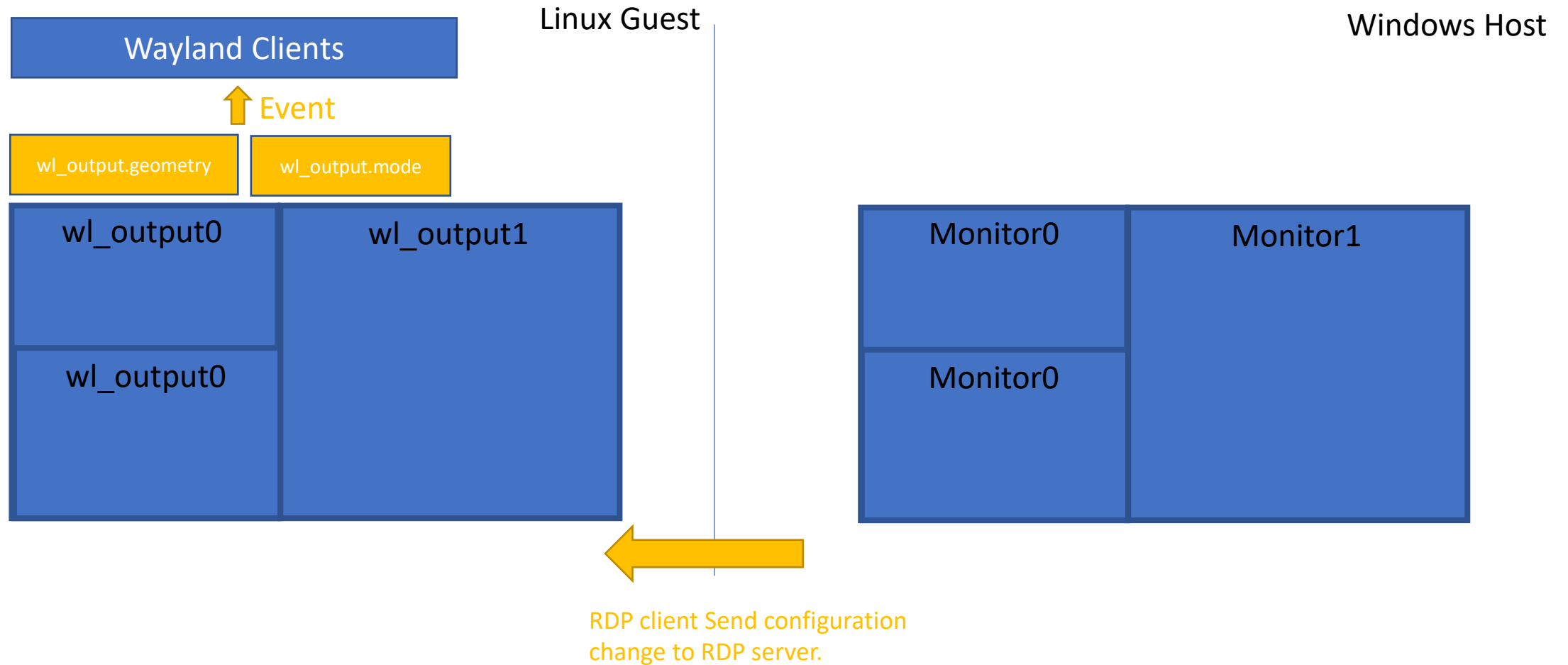
- X11 applications have knowledge and can have direct control over their placement on the X desktop
- In RAIL mode, the guest and the host have the same desktop configuration:
  - Same number of monitors, at same location, resolution, orientation, etc...
- Initial desktop configuration sent by RDP Client upon connecting
  - New configuration information sent by RDP Client when desktop configuration changes
    - Monitor added/remove/repositioned/rotated/resolution change/...

# RAIL – Mirrored Desktop - Connection



On RDP Connection, RDP guest sends initial desktop configuration to the RDP server

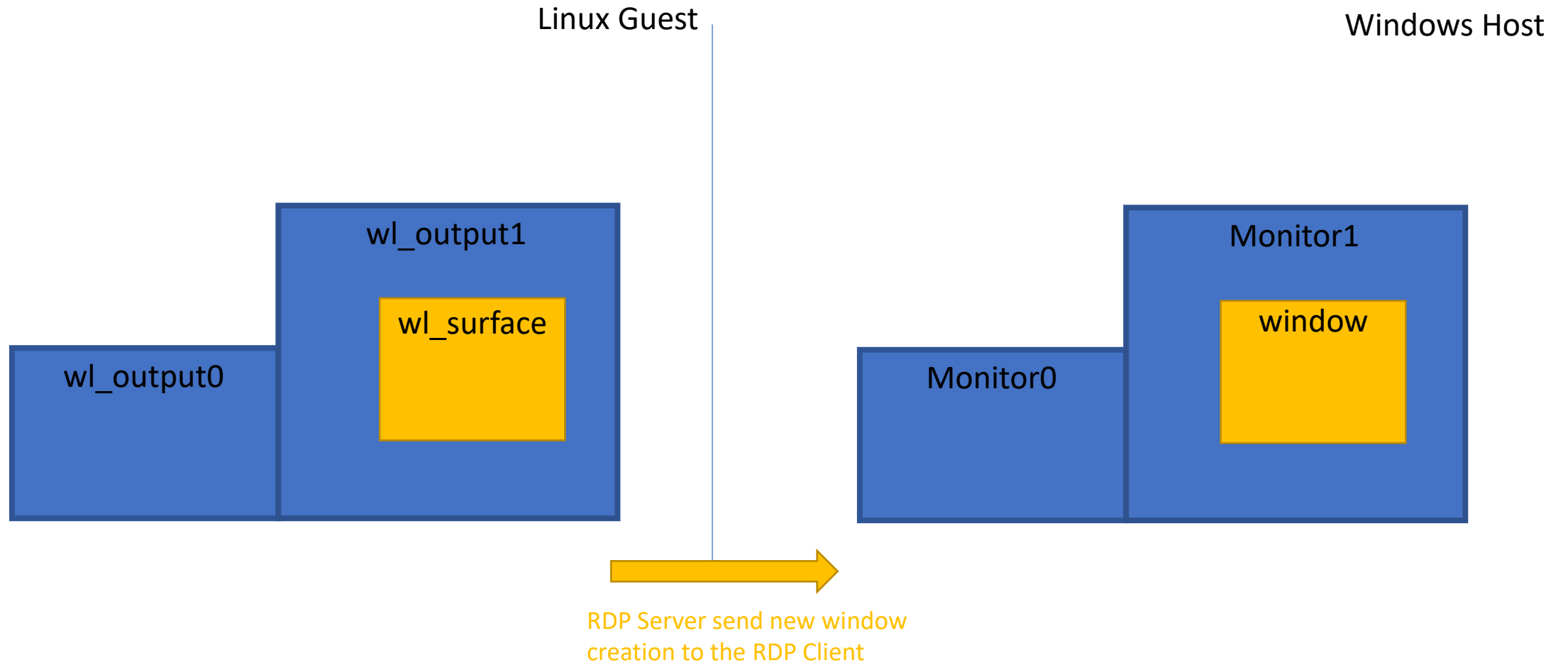
# RAIL – Mirrored Desktop – Desktop Change



# RAIL – Proxying top level Visual

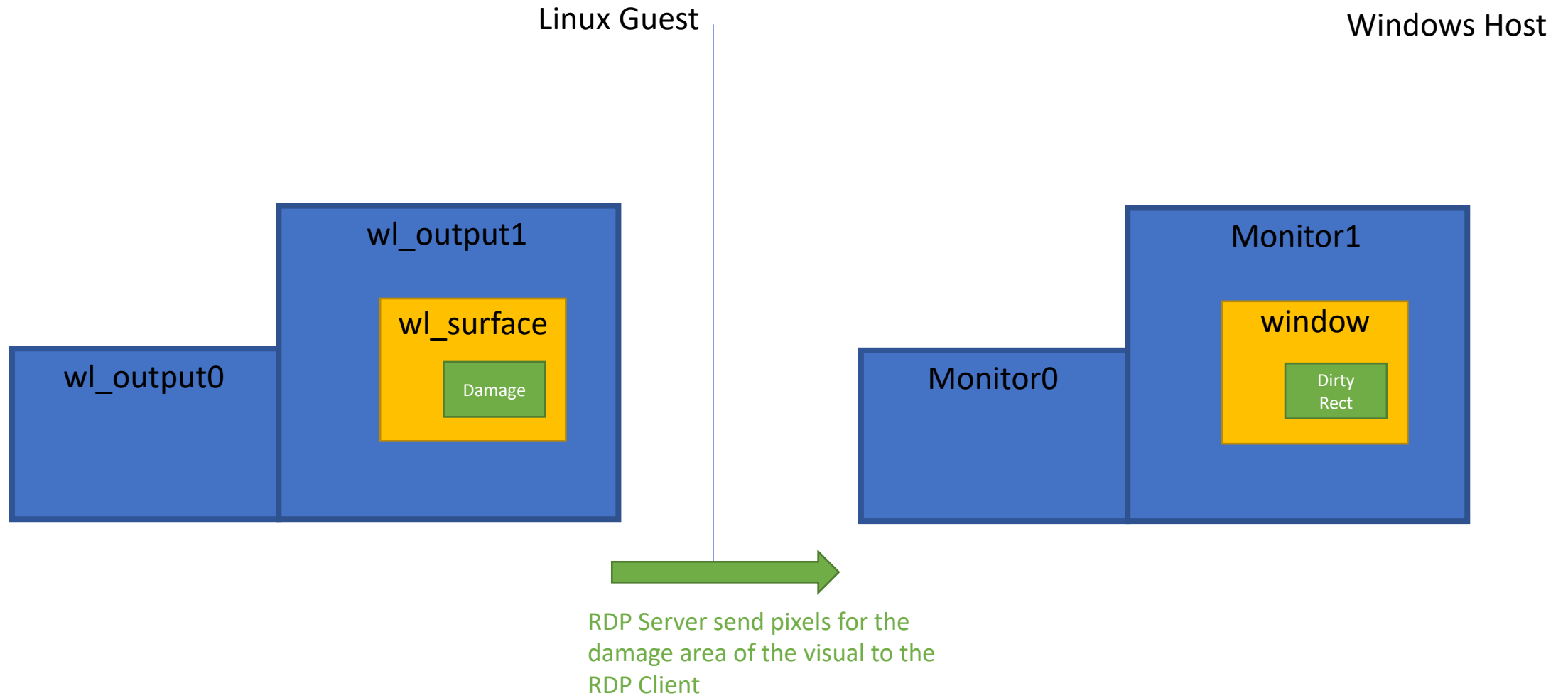
- RDP client on the host create a proxy visual for every top level visual created by the server in the guest
  - Proxy window is borderless. No non-client area. All pixels are owned by the RDP Server.
- Damage pixel in the guest/server are copied to the host

# RAIL – Proxying top level Visual - Create

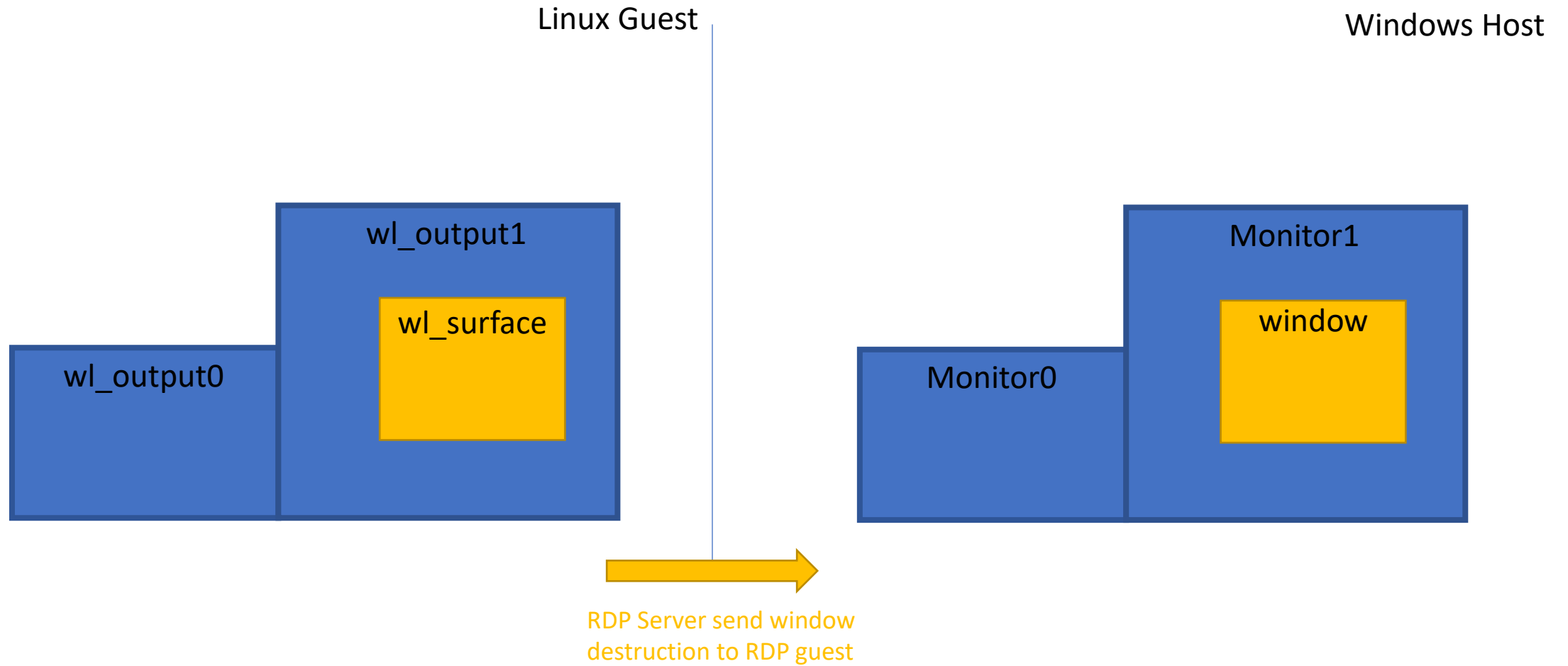




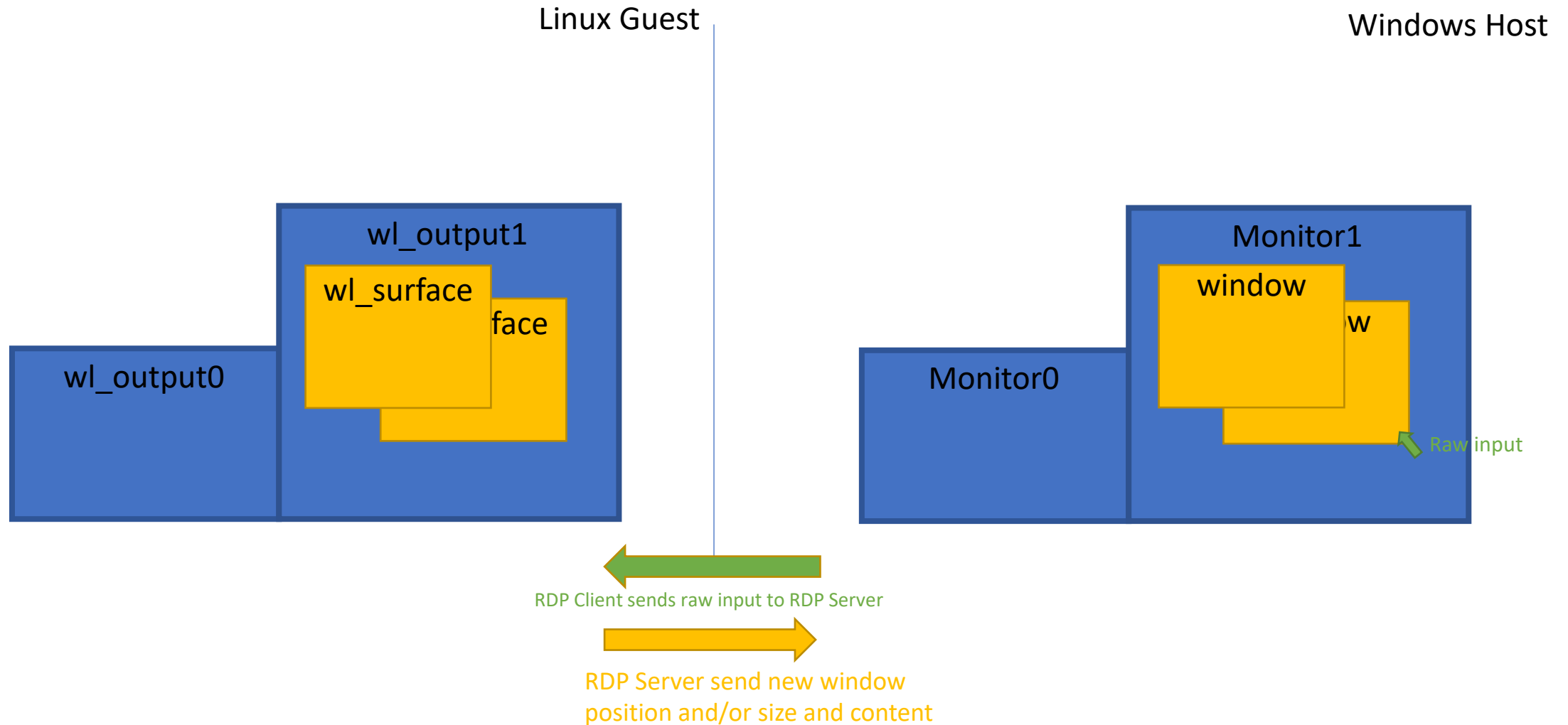
# RAIL – Proxying top level Visual - Update



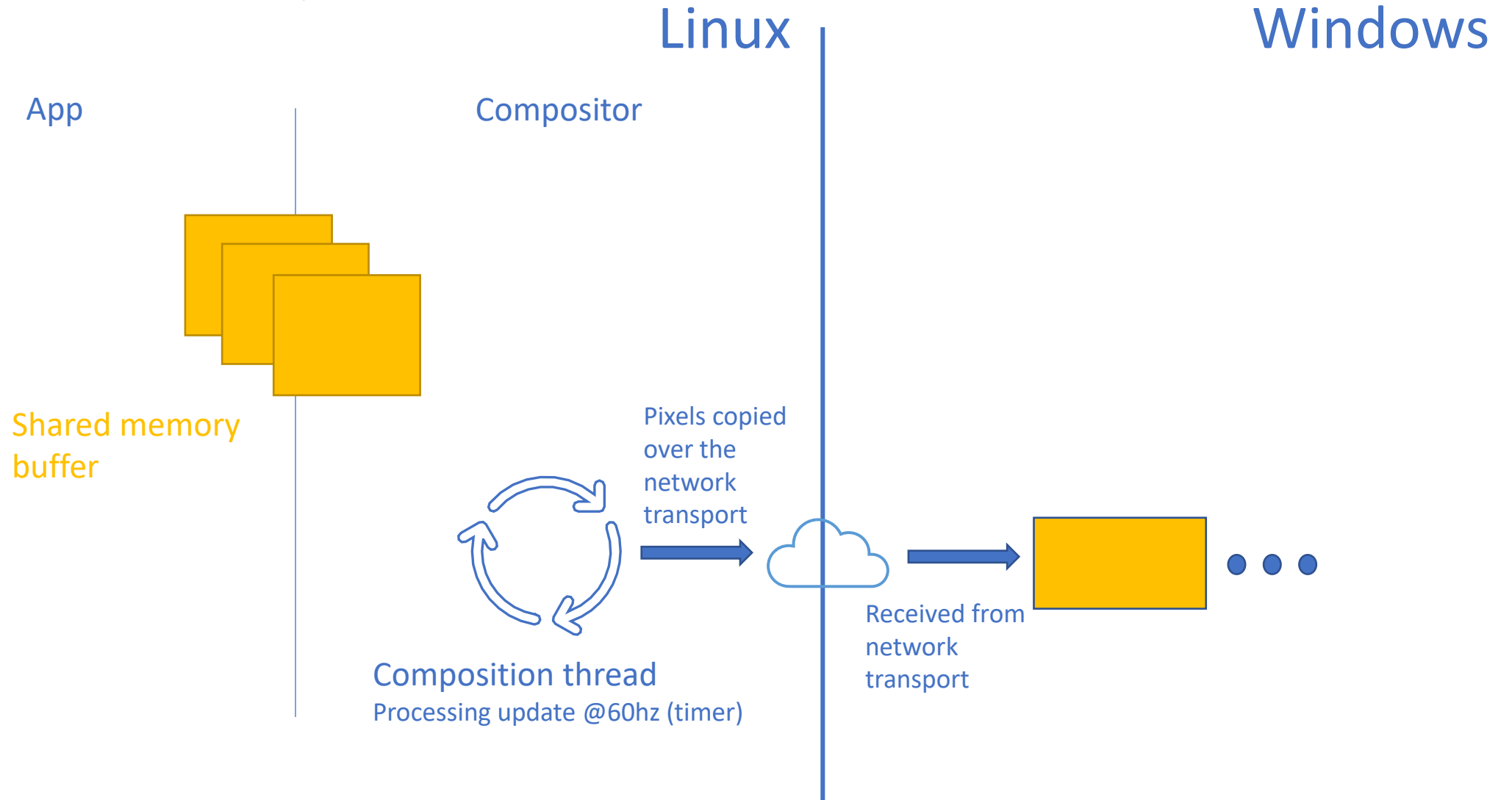
# RAIL – Proxying top level Visual - Destroy



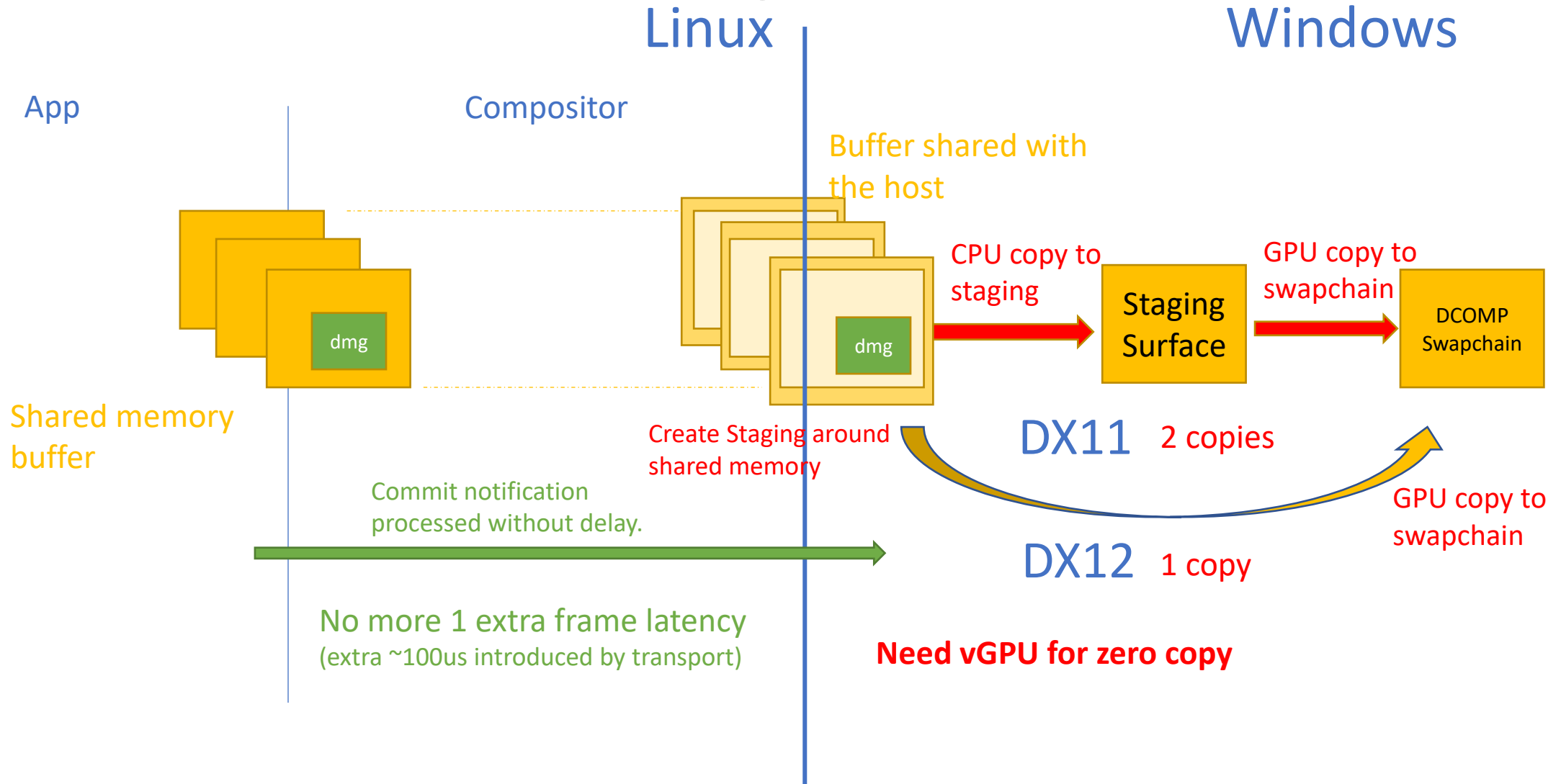
# RAIL – Proxying top level Visual – Move/Resize



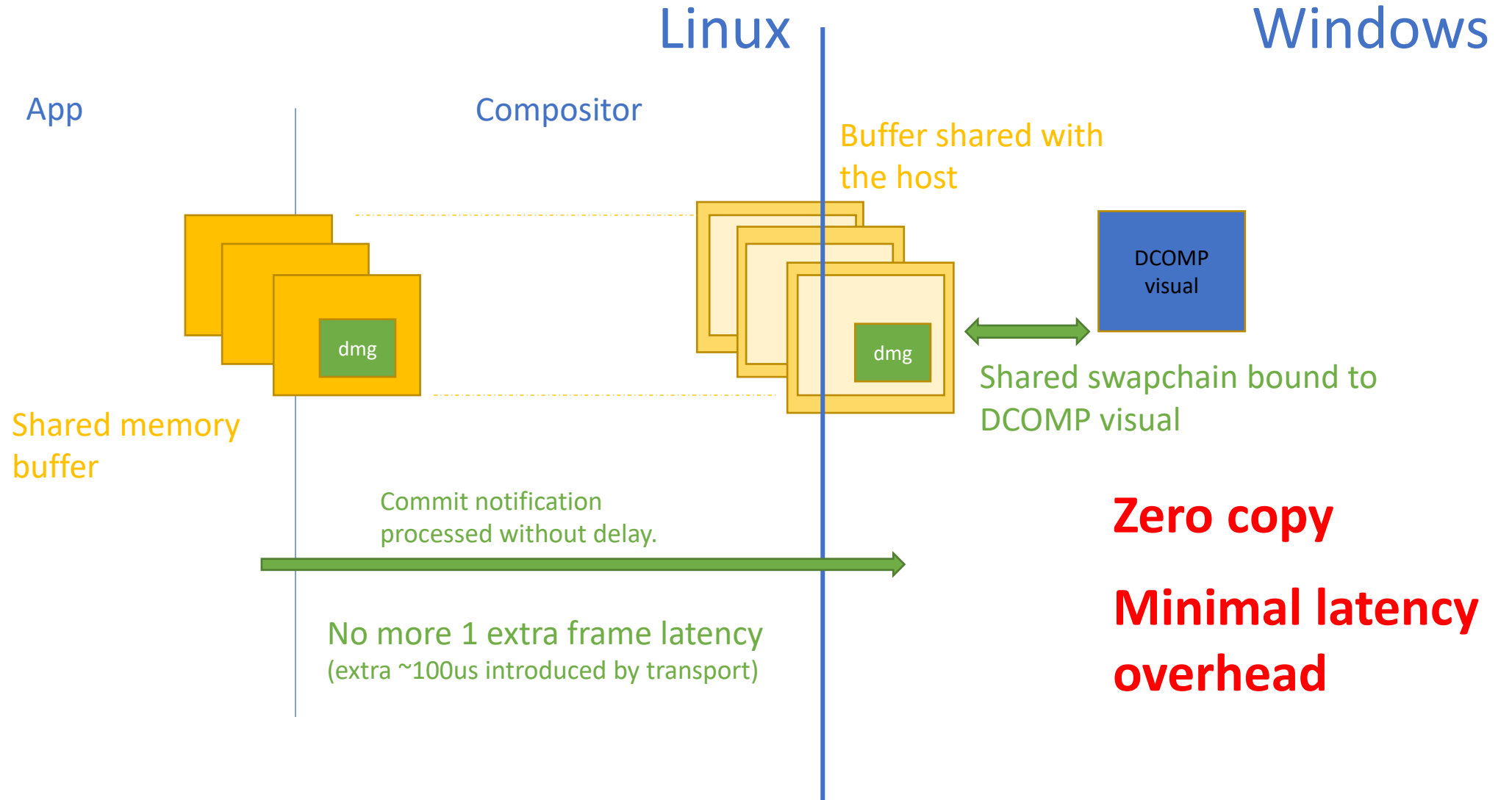
# RAIL & Composition Beat



# VAIL + Software Rendering

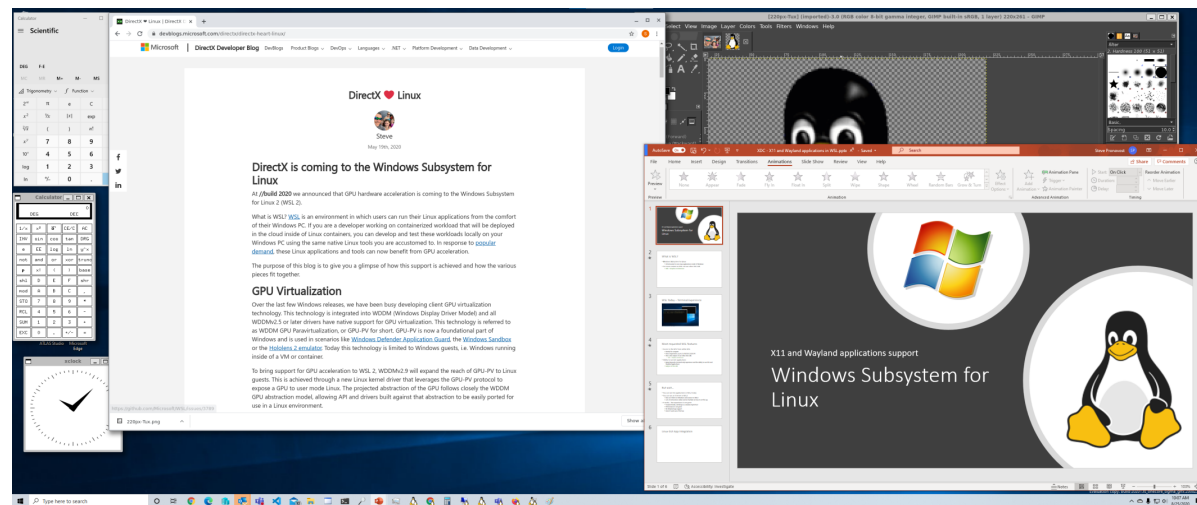
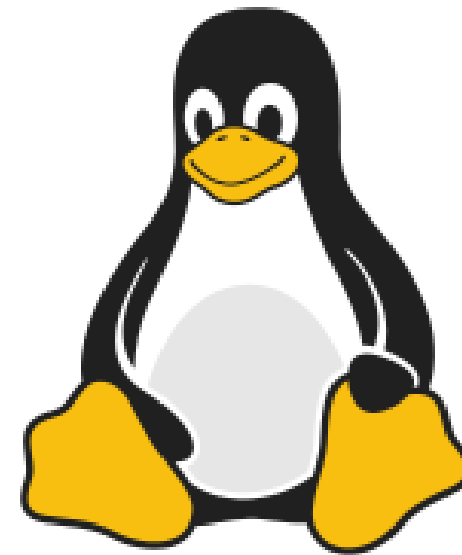
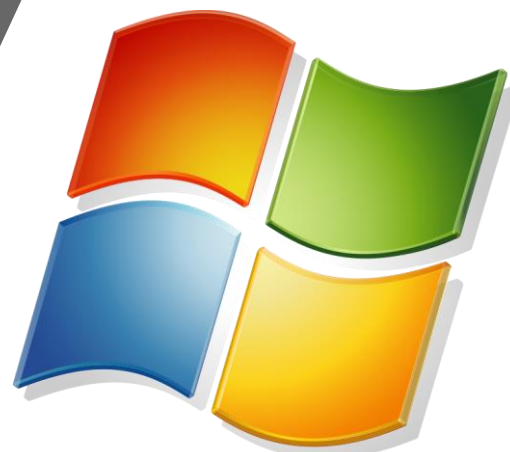


# VAIL + vGPU

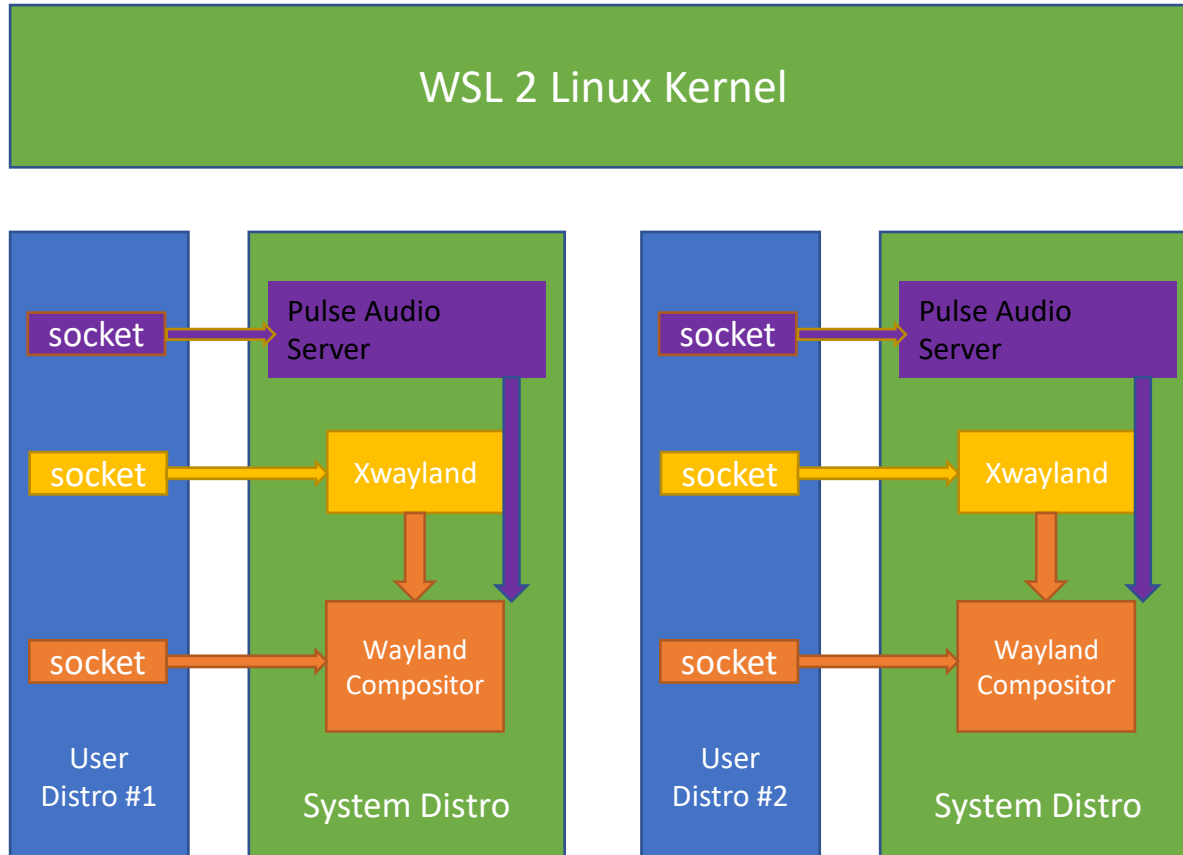


# Packaging and Servicing

wsl --update



# System Distro



- WSLG ships side by side in a system distro
  - Compositor has a clean and isolated user space and project socket back into user distro
    - Sort of like a container, isolated mount and PID space.
  - Each distro has a unique instance of the system distro
  - Each system distro has RDP connection back to the host

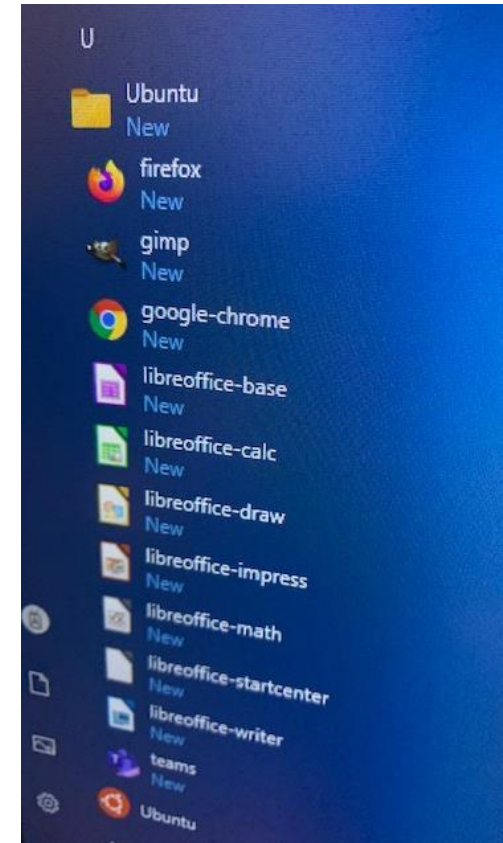


# Servicing

- Shipped separately from Windows
- Always kept up to date
  - Let Window Update handle this for you if you want
  - Or take control of update
    - *wsl --update*
- System distro is read-only, replaced on update
  - Same model & technology used to update the WSL 2 Kernel
  - Simple rollback mechanism in case something goes wrong
- Custom system distro
  - User can use wsl config options to replace system distro with one of their own
  - Everything needed to build a system distro open source
  - Updates only changes the default system distro, don't impact user custom

# Works out of the box

- Automatically setup on WSL install or update
  - User distro pre-configured during WSL custom init
  - Install GUI app & launch
- Linux app appears in the Windows Start Menu
  - Any applications with a .desktop file
- Don't want it or prefer to use an X-Server & Client?
  - Can disable WSLG in wsl config
  - Can keep today's behavior
- Preview coming soon to Windows Insider



# Demo

