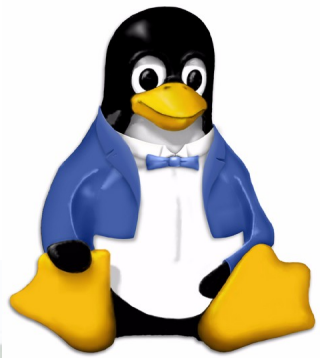


# *PowerPC Xen*

Round Peg, Square Hole?

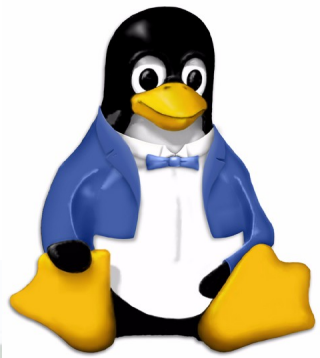
Hollis Blanchard  
Jimi Xenidis

The IBM logo, consisting of the letters "IBM" in a bold, sans-serif font, is positioned in the bottom left corner of the slide. The logo is white and set against a dark blue background.



## *Backing:*

- LTC (Hollis) commitment to productizing on Xen
- Research commitment to using Xen for Virtualization Research
- Power.org, commitment to PPC based open platform, Xen a part of its stack



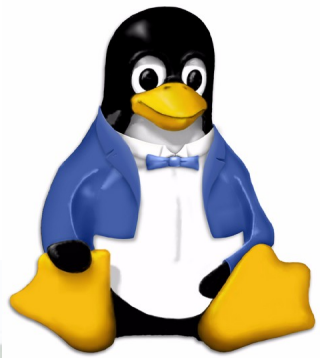
# Platforms:

- 970 series processors
  - Exploiting HV Mode in modern PPC processors
  - Particularly HV mode capable, not G5 :(
    - Some research on 970 processors that do not have HV mode
      - Should pave the way for 32bit and possibly embedded.
    - no POWER5 work, sorry
- JS2x blades
  - Comes with HV layer already but we have new Firmware to allow Xen to run



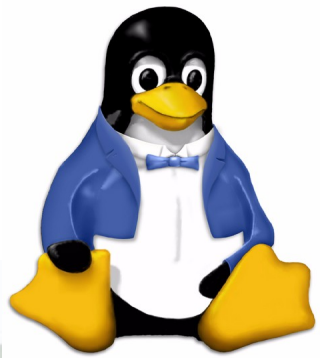
# *Maintenance*

- Linux changes
  - Isolated to arch/powerpc/platforms/xen
  - <200 lines of patch to common arch/powerpc code
- Linux tree based off of Chris Wright's linux\*tip.hg (thanks!)



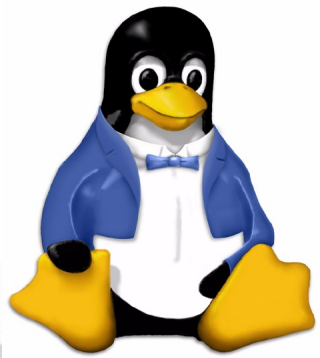
# Porting Issues

- 32 xm/xend apps
  - User level pointers
  - Xencomm is there but is it good enough?
  - `compat_ioctls` have saved the day in some respects.
- Large Pages
  - RMA
  - Can't ignore the performance gain
- Console/Serial access
  - Requirement for single Linux Image
- Multi-boot loader
  - Grub not there yet



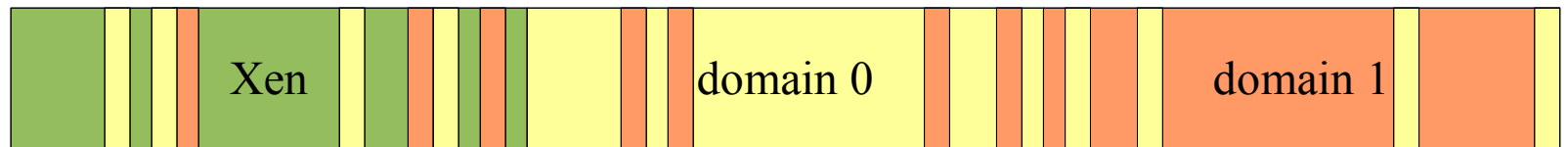
# *Status*

- Code now merged in -unstable
- Single Image for all Doms,
- All direct IO access is IOMMU based
- xm/xend support is 95% of the way there.
- SMP support with UP Domains
- Exploiting HA
- Grant Tables
- STILL, No VIO Yet, “netfront copy mode”?!
- start\_info is unused

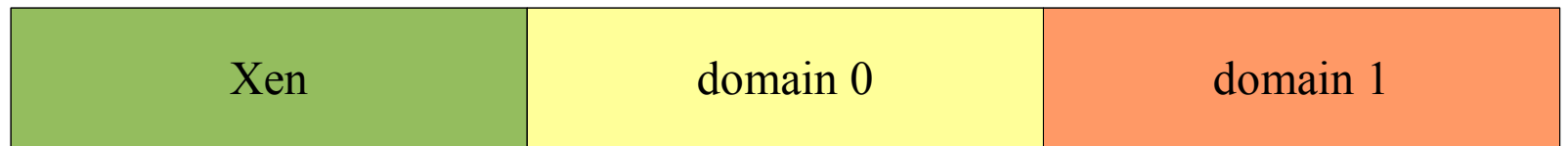


# Real Mode Areas

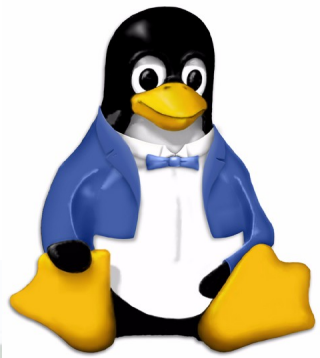
- PowerPC requires 64MiB of contiguous, naturally aligned memory for every domain.



current domain allocations: discontinuous



PowerPC real-mode areas: contiguous, naturally-aligned



# *Other Topics (portability talk)*

- Domain Firmware/Loaders
  - Is VIO too complex?
- Compat\_ioctls
- Too many pointers in hcalls when a value would do
- Guest\_handle should be “real” structures?
- Endian-ness, code for it always
- Time
- Config/Make issues
- WARN() or some recoverable version of BUG()
  - It is not unreasonable to “fix” a problem from the debugger.