

Towards a Platform for Homes

Colin Dixon
University of Washington

Homes Today

- Each home is it's own custom machine
- Each home administrator is writing custom programs for this custom machine
- We don't even have artisans yet!
- Ruled by Turning, not yet visited by Clark

Heterogeneity Kills

- “It takes a few hours just to understand how their house is set up”
- Best practices describe configurations which ease management by humans
- An abstraction layer or platform would ease management by “programs”

A Platform Approach

- x86, HAL, Windows, POSIX, HTML/CSS/Javascript
- Lets me run other people's solution to my problems rather than rolling my own
- Lets me provide my solution to others without too much additional work
- Can we do this for homes?

Possible Platforms

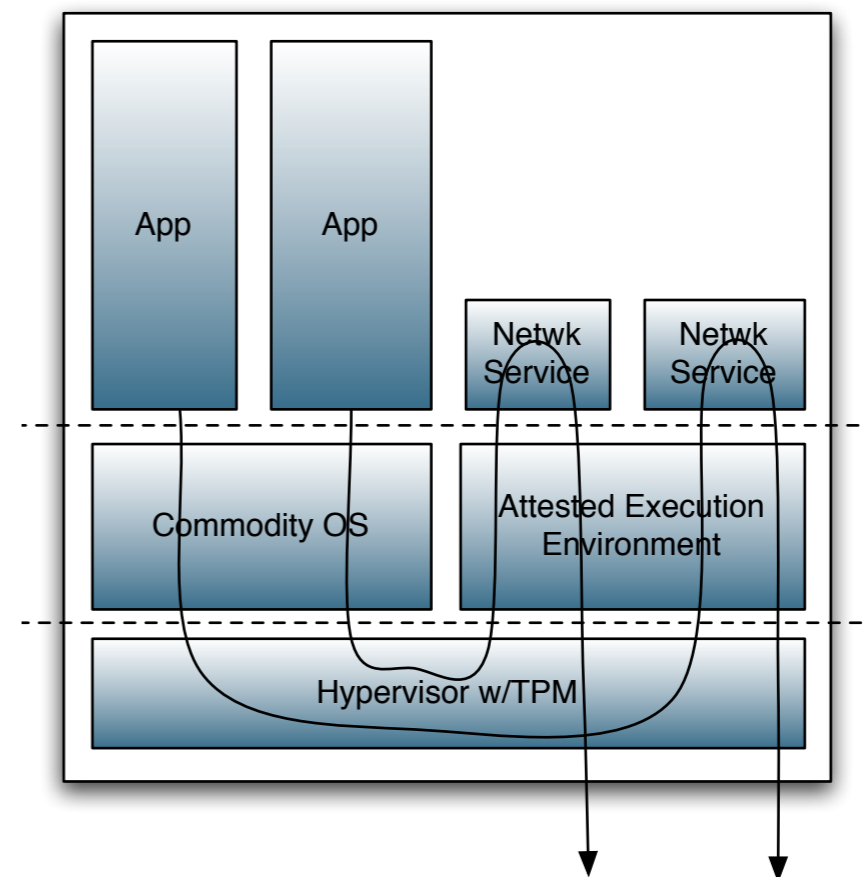
- Standardize everything (Boxes & Arrows)
- One smart, standardized box
(OneBox, box in the basement)
- Remove everything complicated
(thin client + cloud)
- Standardize what we can easily and put for
what we can't

Features

- Visibility
- Control
- Composability
- Discovery/Naming
- Diagnosis

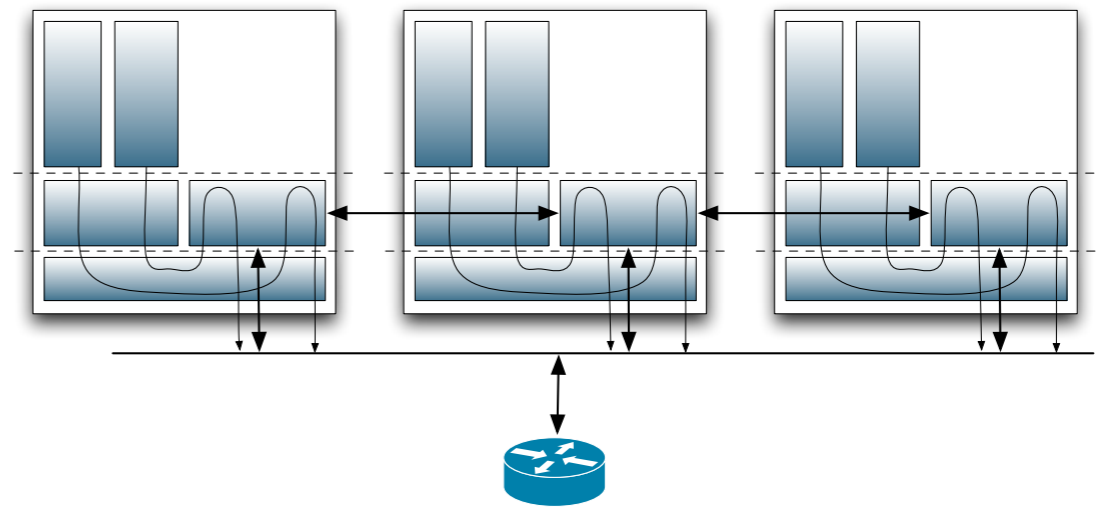
An End to the Middle

- Each device that can run a virtual machine
- This machine is visibility to and control of traffic to and from this device
- This is where code for the platform is executed
- Assume multiple concurrent applications



An End to the Middle

- These VMs coordinate to use, export, compose and manage services
- Manage other devices via “device drivers”
 - on the device (standards)
 - in the VM (translator)



Diagnosis

- Being on the host gives a superior level of visibility and control
- There is an explicit agent responsible for understanding how to achieve a goal
- Somewhat closer to end-to-end understanding and testing