A Lifecycle Approach

Phil Shafer
IETF96, Berlin
Problem

• Fuzzy terms
  – Applied, Intended, active, even configuration

• Rorschach test for readers
  – We agree to things where we have different understandings of terminology
Lifecycle Approach

• Let's talk about phases and processing
  – How data moves and is manipulated

• Avoid overloaded names
  – Use "Alpha", "Bravo", or Pokemon names; don't care
  – When the lifecycle is fully defined, names will appear (hopefully)
Goals

• Define a concrete lifecycle
  – An information model for on-box data

• Be explicit about phases and phase changes
  – What's added, removed, or changed between phases

• Call out distinct behaviors
  – Current and future, useful and real-world

• Lastly: Create meaningful names for phases
Example Phase Changes

– Remove data for absent hardware (FRUs)
  • "ephemeral interfaces", chip features
– Add system-defined hardware (mgmt ports)

– Add device-based defaults
  • Model-specific, lists

– Remove nodes marked "inactive"
– Add expansions of configuration groups / templates
– Add data from on-box scripting
Example Phase Changes (2)

– Add data from external controller
  • Persistant (survives reboot)

– Add data from external controller
  • Does _NOT_ persist

– Hand-built data (human hands)
  • Has higher priority
  • Persistant
  • (First phase change?)
Finalized Instructions
aka "x-ray"

True Device Behavior
aka "yankee"

Full Device Data
aka "zulu"

Replace nodes with current values

Add config=false nodes