

Refined YANG Datastores

draft-wilton-netmod-refined-datastores-01

IETF 96 – Berlin, NETMOD WG

Rob Wilton – Cisco

rwilton@cisco.com

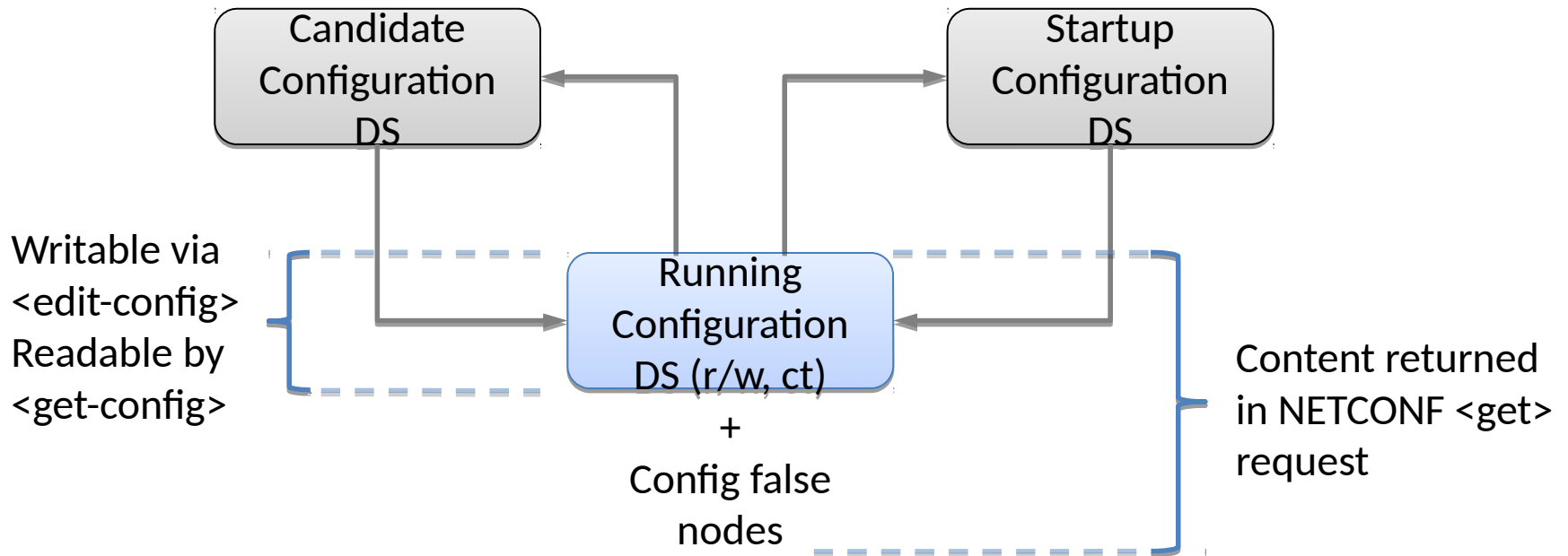
Problem Description

- One of two possible datastore solution drafts
- Primarily aims to solve two problems:
 - Opstate split between intended vs applied configuration
 - Remove the need for a config/state split between “feature” and “feature-state” sub-trees

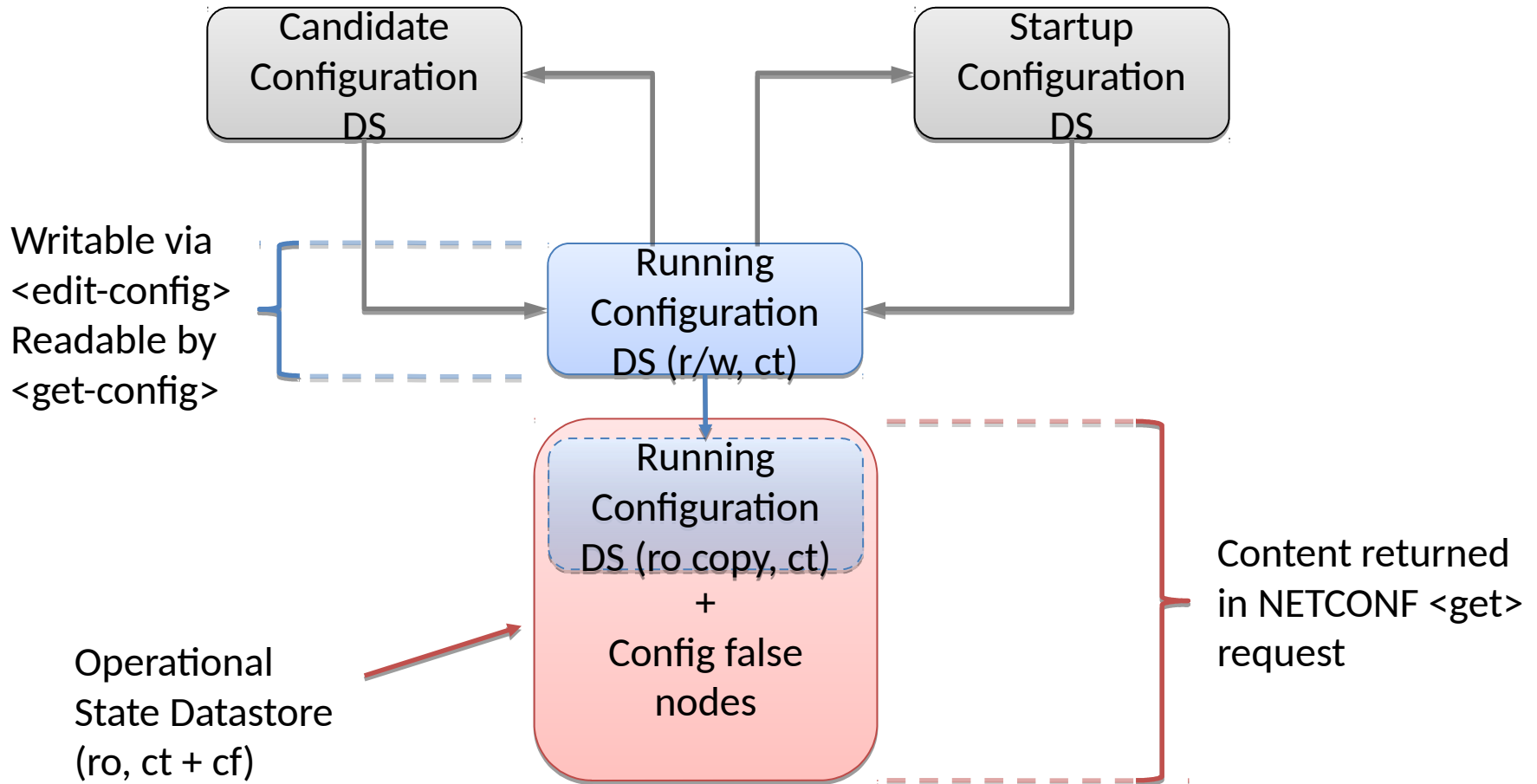
Proposed Solution

- 1) Formally defines an “Operational State Datastore”, that contains:
 - Applied configuration
 - System controlled configuration
 - All config false nodes (inc ephemeral opstate & statistics)
- 2) Split the Running Configuration Datastore into:
 - Abstract “Intended Configuration Ds” and “Applied Configuration Ds”
- 3) Split the abstract Intended Configuration Datastore into:
 - Persistent configuration datastore (recovered after reboot)
 - Ephemeral configuration datastore (for I2RS, lost on reboot)
- 4) Introduces abstract datastores

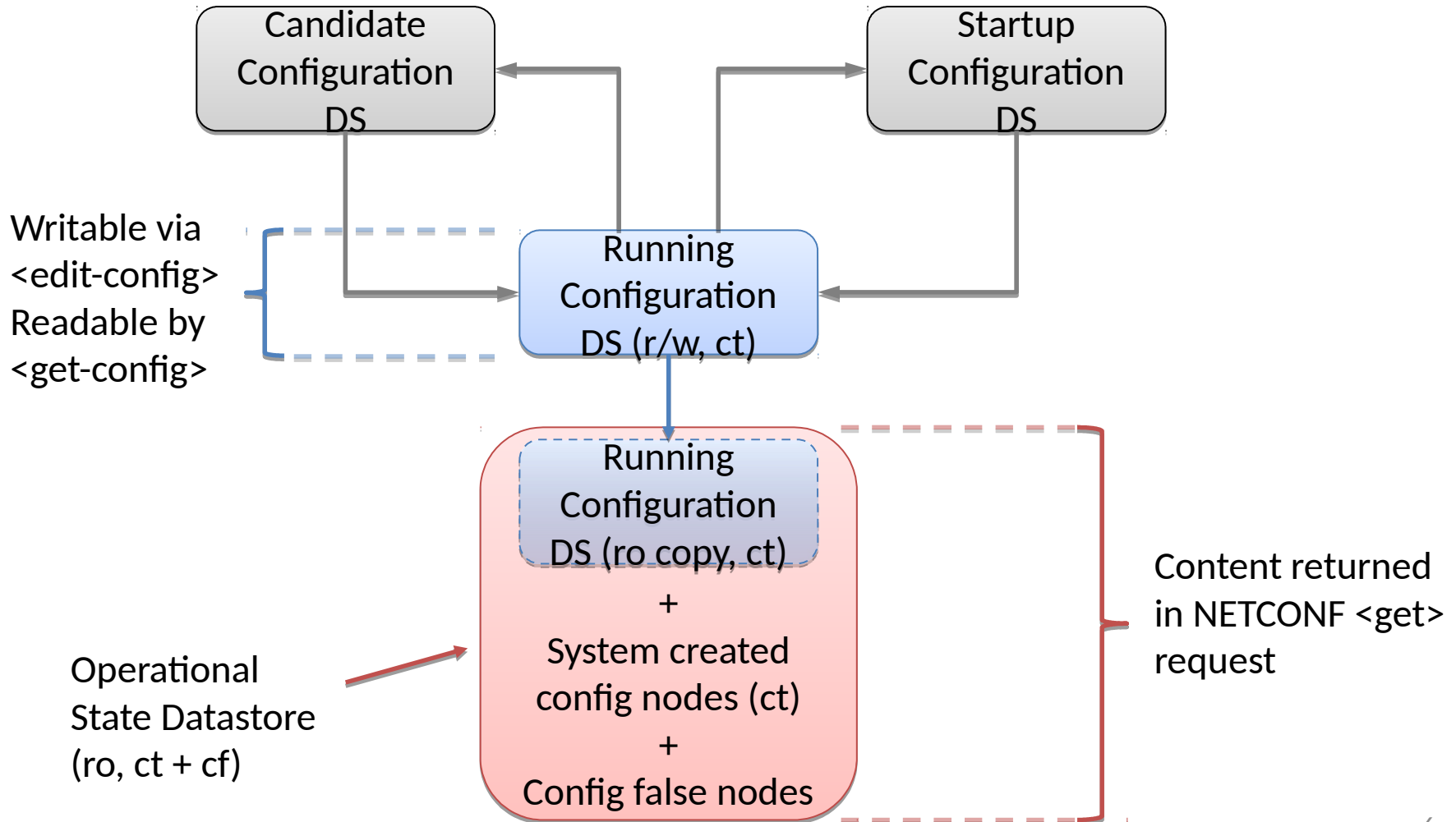
Existing NETCONF Datastores (as per RFC 6241)



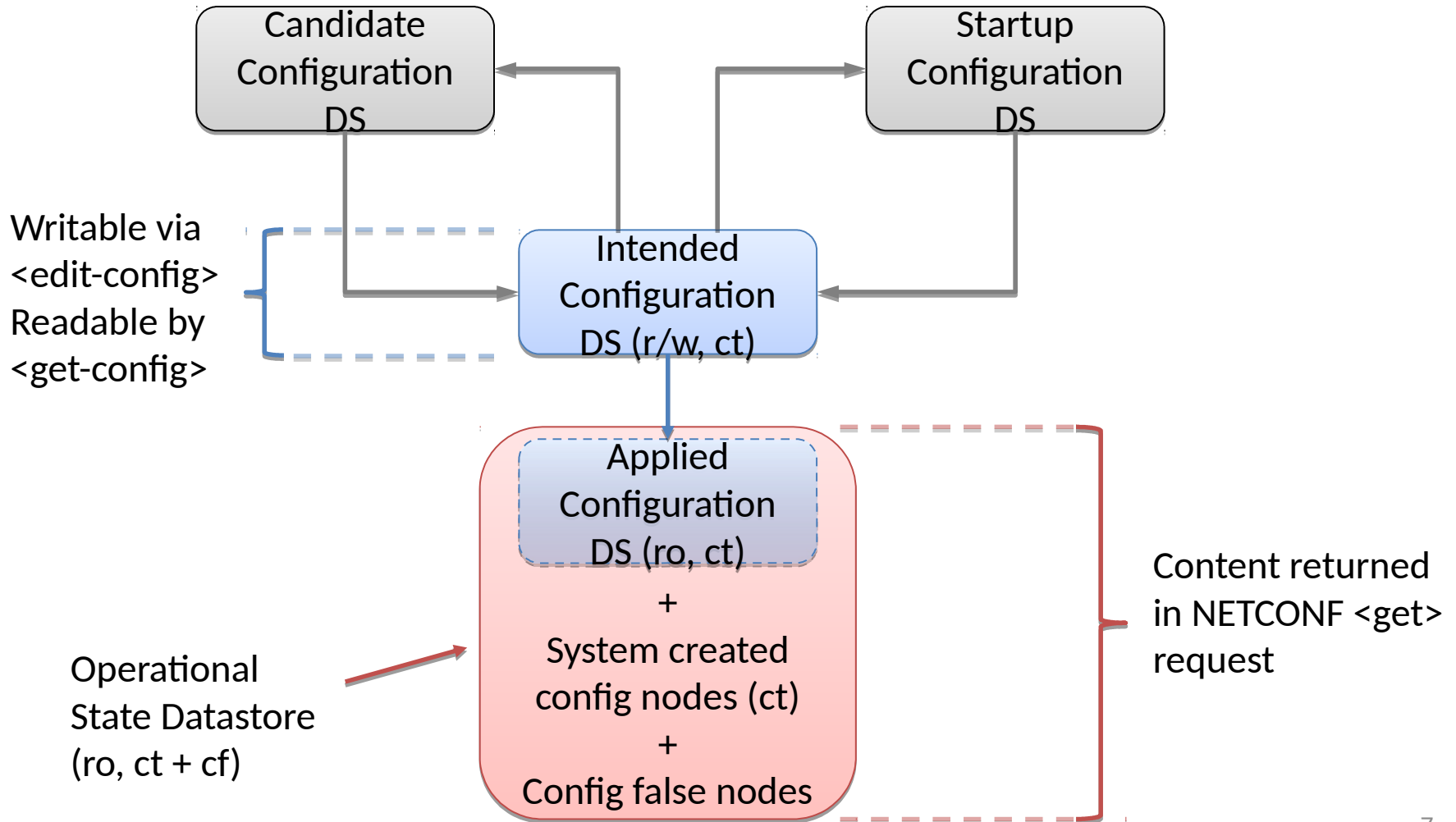
Define “Operational State Datastore”



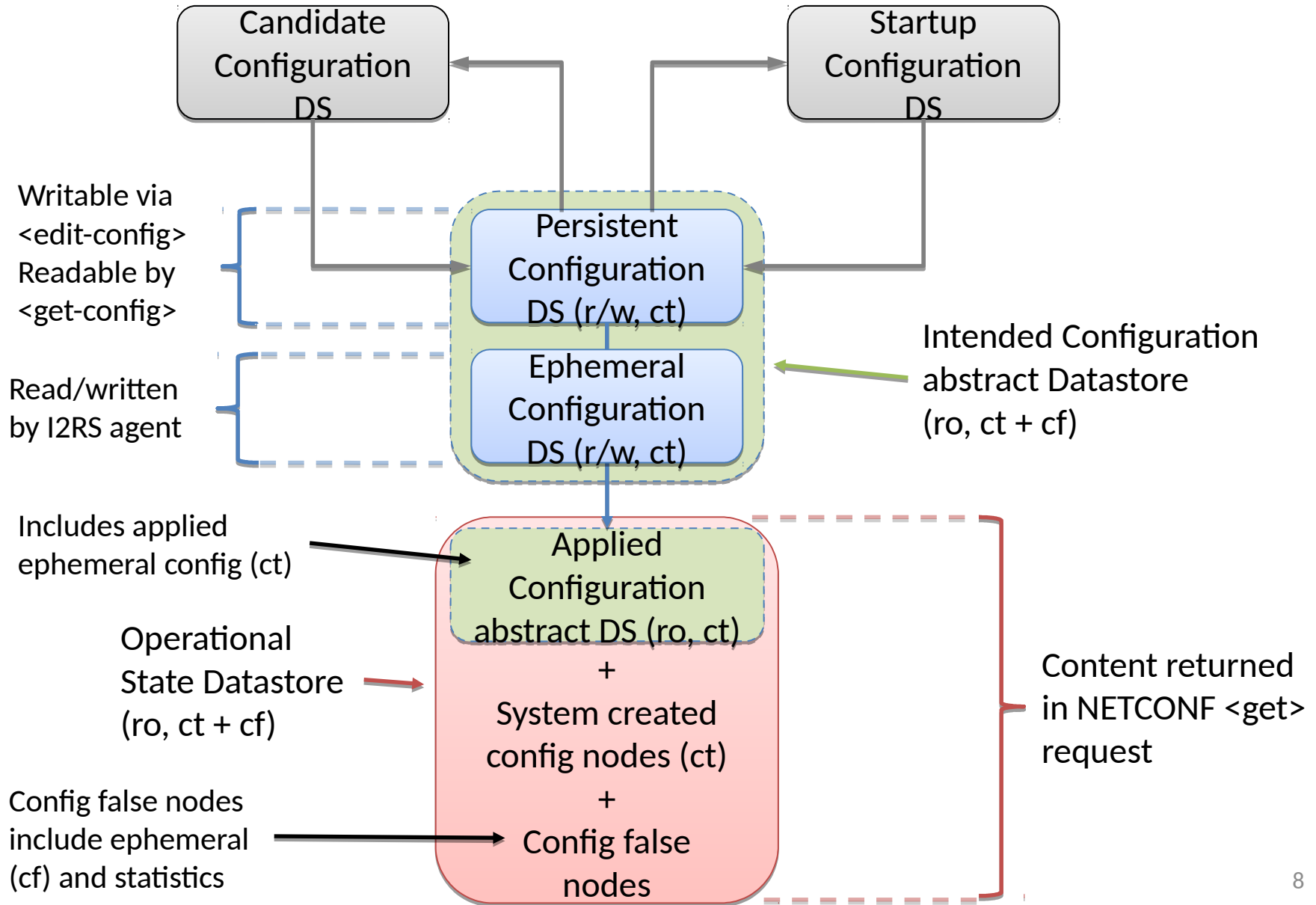
Add System Controlled Configuration



Split intended vs applied configuration



Add support for Ephemeral & Statistics



Summary

- A “partly backwards compatible” refinement of datastores
- Solves:
 - Opstate split between intended vs applied configuration
 - Removes the need for a config/state split between “feature” and “feature-state” sub-trees
- Also potentially solves:
 - Pre-configuration (present in intended but not applied)
 - System created configuration (e.g. interfaces)
 - Device deviations (when not possible to express in a YANG model)

Thanks for listening. Any questions?