Section 1: Week 2: Plan for Distributed Software Architecture

Nate Bachmeier

TIM-8120: Distributed Systems

September 22nd, 2019

North Central University

# Part I: Expand the Conceptual Diagram

Contoso operates a distributed supply chain of clothing manufacturing and retailer outlets. Within their systems, multiple autonomous agents exist and need to interact with the same shared resources. These requirements create the need to consider the operations and service management of the network. Without these capabilities, a single node failure could cascade and compromise the reliability of the topology. As the reliability and available of the environment degrades, it causes a loss of productivity and consumer confidence. It is therefore advantageous to minimize these scenarios.

## Sharing Resources Between Agents

Autonomous agents include humans, machinery, mobile devices, office equipment, and web jobs. As these agents perform work, they persist modifications to centralized data services. These update operations can and will occur in parallel, which leads to the need for protocols to handle these concurrent and potentially conflicting operations.

Consider the scenario where three employees are attempting to access a garment within the clothing catalog. (1) The first wants to get a scarf, the second update its price, and the third wants to discontinue (delete) the item. Each of these requests arrives at the application load balancer and becomes routed to healthy front end web service instances. (2) Within the front end instance, the operation passes through a collection of middleware layers; these are responsible for decoupling the business logic from the transport configuration. (3) Next the message moves across another service load balancer into a ring of stateless data store providers. (4) The stateless data store provider instance is responsible for interacting with the stateful data store provider. (5) In the case of the update or delete the caller might provide a revision number or similar entity tag so that the caller can be alerted to changes since they retrieved the remote object. If the revision number has not changed then the modification can be safely committed. (6) To ensure that the commit is highly durable, the stateful data store would need to wait for a sufficient number of nodes within the cluster to acknowledge the store operation. (7) Only after confirming the successful handoff can the system return to the caller the process was successful.

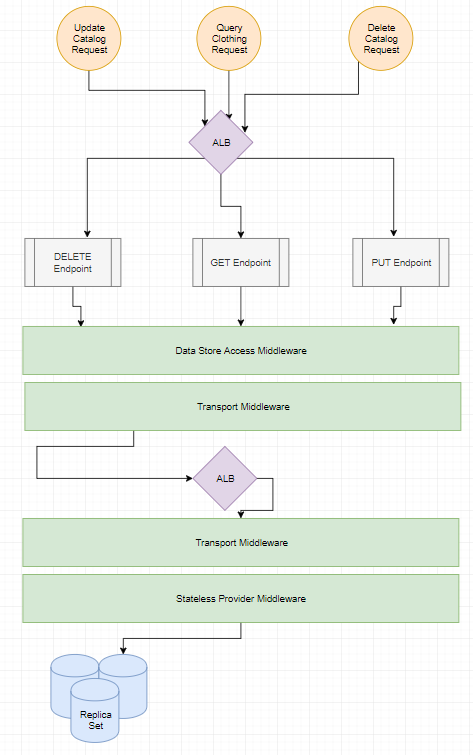


Figure : Single System View

## Reducing the Blast Radius

The Clothing Catalog reference architecture applies to many scenarios. First, brokering the network traffic across an Application Load Balancer (ALB) spreads the traffic evenly across a target group. These target groups are continuously joining and evicting service instances by monitoring their performance metrics. Second, middleware within the client is responsible for managing the retry policy as operations fail with transient issues — for example, a service host’s physical power state cycle, resulting in arbitrary failures. The transport middleware could then detect this state and reissue the request for the ALB to route to an alternative instance. Third, to ensure that the modification is not lost, the caller needs to get an acknowledgement from the callees they completed the work. If the completion cannot be acknowledged or is erroneous, then that failure needs to be propaiged to the client for compensation actions.

## Management and Operational Services

External management systems need to exist to provide operational capabilities for the broader topology. These capabililities include directory and domain services (e.g., authentication, name resolution, etc.); operational monitoring (e.g., system health); provisioning services (e.g., machine replacement); and desired state configuration (e.g., policy management). Each of these systems is self-contained with dedicated target groups and private data stores to ensure high availability.

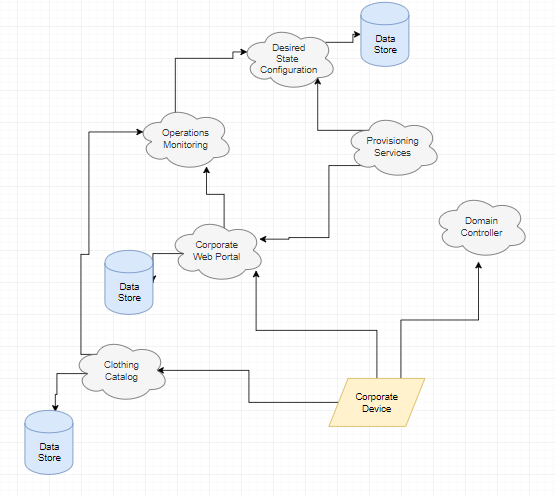


Figure : Management View

## Modifications to the Headquarters Diagrams

The headquarters system diagram was missing routers and switches and was updated. There were additional considerations to the global network topology of Contoso and those requirements.

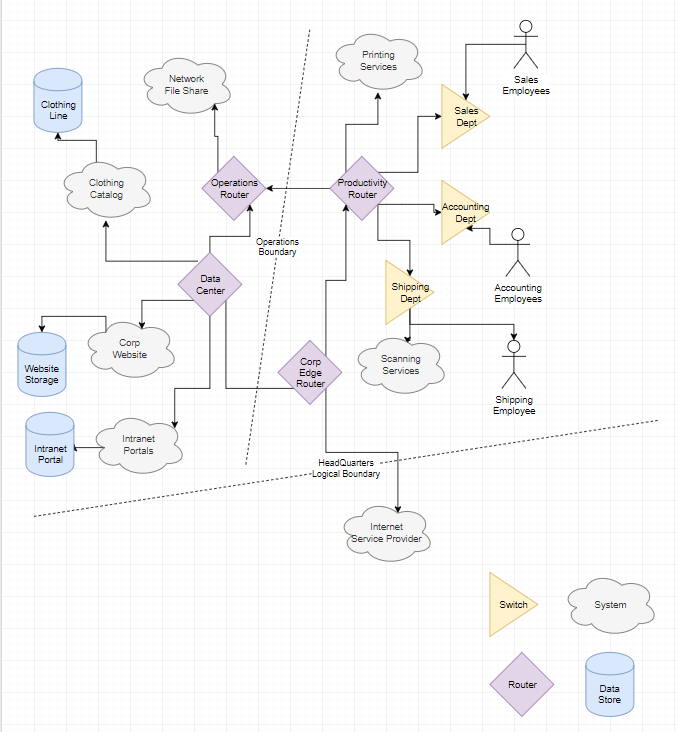


Figure Headquarters Topology

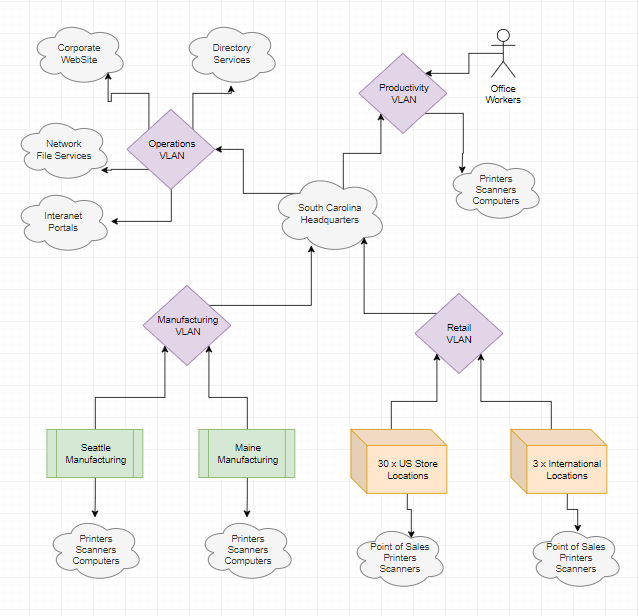


Figure Global Topology

# Part II: Cobra and Apache River