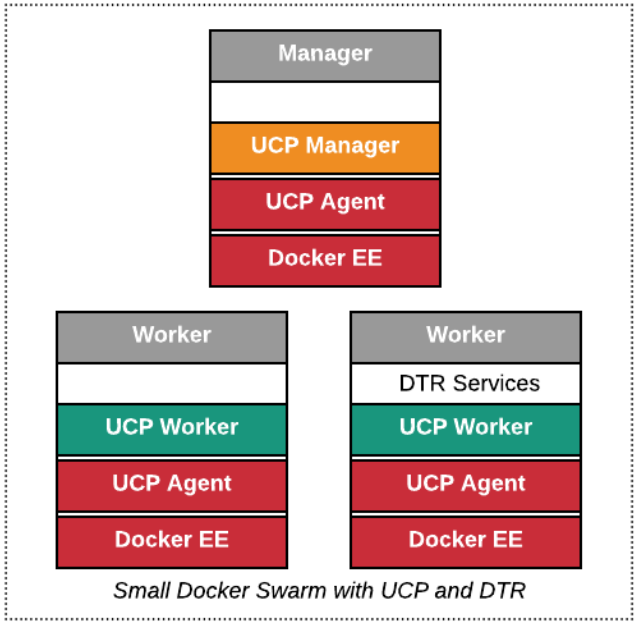
Docker Certified Associate

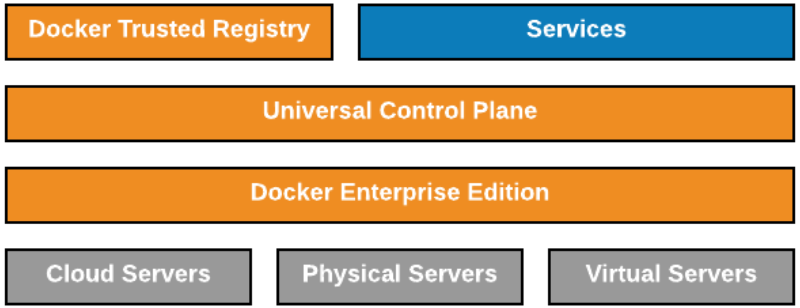
Communication – Docker Engine, UCP and DTR

Communication Traffic Between Docker Engine, UCP and DTR



Small Docker Swarm with UCP and DTR

Given this small Docker Swarm example, there are a number of components that are communicating across multiple nodes and using multiple services. The overall 'stack' view of these components is below.



UCP Components (Manager)

UCP Component	Description
ucp-agent	Monitors the node and ensures the right UCP services are running.
ucp-reconcile	When the agent detects the node is not running correct components, it has the container convert to the desired state.
ucp-auth-api	Centralized service for identity and authentication used by UCP and DTR.
ucp-auth-store	Stores authentication configurations and data for users, organizations, and teams.
ucp-auth-worker	Performs scheduled LDAP sync (when configured) and cleans authentication and authorization data.
ucp-client-root-ca	Certificate authority to sign client bundles.
ucp-cluster-root-ca	Certificate authority used for TLS communication between UCP components.
ucp-controller	UCP Web Server.
ucp-dsinfo	Docker system information collection script to assist with troubleshooting.
ucp-kv	Used to store the UCP configurations (internal use only).
ucp-metrics	Used to collect and process metrics for a node (i.e. disk available).
ucp-proxy	A TLS proxy that allows the local Docker Engine secure access to the UCP components.
ucp-swarm-manager	Used to provide backwards-compatibility with Docker Swarm.

UCP Components (Workers)

UCP Component	Description
ucp-agent	Monitors the node and ensures the right UCP services are running.
ucp-dsinfo	Docker system information collection script that assists with troubleshooting.
ucp-reconcile	When the agent detects the node is not running correct components, it has the container convert to the desired state.
ucp-proxy	A TLS proxy that allows the local Docker Engine secure access to the UCP components.

Communication between the Docker Engine, UCP, and DTR can happen:

- Over TCP/UDP – depends on the port and whether a response is required or if the message is a notification.
- IPC – services on the same node can use IPC to communicate amongst each other.
- API – will take place over TCP (of course), but uses the API directly to query and update the components across the entire cluster.