SCOPE

ON-PREM DOCUMENTATION

ScopeAR Swarm API Reference (swarm.scopear.com/v1)

Version	3.96.0
Channel	on-prem
Mode	production
Revision	c62b658
Generated	2026-08-27

Confidential · For authorized Scope AR customers and partners

Contents

How to read this reference

API Reference

- Packages
- swarm.scopear.com/v1
  - Resource Types
  - AI Config
  - AI Spec
  - Cms Config
  - Common
  - Comms Config
  - Comms Spec
  - Config MQTT
  - Coturn Config
  - Gwl Spec
  - IOT Config
  - IOT Spec
  - Minio Config
  - MongoDB Config
  - Mysql Config
  - Nginx Config
  - Postgres Config
  - Redis Config
  - Release Channel
  - Resource
  - Resource Limit
  - Resources
  - ScopeAR Spec
  - ScopeAR (Swarm)
  - Secrets Spec
  - Sidekiq Config
  - Superset Config
  - Superset Spec
  - Traefik Spec

- Traefik TLS Provider
- Traefik TLS Spec
- Web Spec
- WLC2 Api Config
- WLC2 Client Config
- WLC2 Model Converter Config
- WLC2 Project Packager Config
- WLC2 Spec
- WLC2 Webapp Config
- Zeus Config

Samples

- `scopear_swarm_env_sample.yaml`

How to read this reference

This document is the generated field reference for the ScopeAR custom resource (`apiVersion: swarm.scopear.com/v1`, `kind: ScopeAR`). It is the canonical list of configuration keys, allowed values, and defaults for environment manifests.

The Go type for this resource is `ScopeARSwarm`; environment manifests use YAML `kind: ScopeAR` with `apiVersion: swarm.scopear.com/v1`.

- **Field** — YAML key under the parent object.
- **Type** — Scalar, object, or array; linked names are defined later in this document.
- **Req** — `Yes` must be set; `No` is optional (often has a default).
- **Default** — Value applied when the field is omitted (when known).
- **Description** — Behaviour, constraints, and operational notes.

Complete example manifests are in the **Samples** appendix and also ship as raw YAML files next to these PDFs (`samples/`).

API Reference

Packages

- swarm.scopear.com/v1

swarm.scopear.com/v1

Package v1 contains API Schema definitions for the v1 API group

Resource Types

- ScopeAR (Swarm)

AI Config

AIConfig configures deployment settings for AI service tasks.

Appears in - AI Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.

AI Spec

AISpec configures the ScopeAR AI service and its persistent data.

Appears in - Scope AR Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`enabled`	boolean	No	`false`	Enables deployment of the AI service. Configure model-provider secrets, MongoDB, and object storage before enabling it.
`ai`	AI Config	Yes		Configures replicas and resources for AI service tasks.
`config`	AI Config	No		Supplies AI application settings reused from k8s.scopear.com. Swarm defaults initialize Qdrant, Helicone, MongoDB, and MinIO integration values.
`data_volume`	string	No	`/opt/scopear/data/ai`	Sets the host directory used for persistent AI service data. Make it available on eligible swarm nodes and include it in backup planning.

Cms Config

CmsConfig configures the CMS service task for Docker swarm.

Appears in - Web Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`data_volume`	string	No	`/opt/scopear/data/cms`	SMTP v1.CmsConfigSMTP `json:"smtp,omitempty"` Mysql v1.CmsConfigMysql `json:"mysql,omitempty"` Sets the host directory used for persistent CMS data. The directory must exist or be creatable on nodes eligible to run CMS.

Common

Common defines deployment settings shared by swarm services.

Appears in - AI Config - Cms Config - Coturn Config - `DtoolConfig` - Gwl Spec - IOT Spec - Minio Config - Mongo Db Config - Mysql Config - Nginx Config - Postgres Config - Redis Config - Sidekiq Config - Superset Spec - Wlc 2 Api Config - Wlc 2 Client Config - Wlc 2 Model Converter Config - Wlc 2 Project Packager Config - Wlc 2 Spec - Wlc 2 Webapp Config - Zeus Config

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.

Comms Config

CommsConfig configures dependencies used by the Zeus service.

Appears in - Zeus Config

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`redis`	Redis Config	No		Configures the Redis instance used by communications workloads. Its default persistence path is shared with the web Redis default; avoid conflicting deployments.

Comms Spec

CommsSpec configures the Zeus and Coturn communications services.

Appears in - Scope AR Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`zeus`	Zeus Config	Yes		Configures the Zeus signaling and communications service.
`coturn`	Coturn Config	Yes		Configures the Coturn TURN relay service. Expose the configured control and relay port ranges through host firewalls.

Config MQTT

ConfigMQTT configures the MQTT listener and its certificates.

Appears in - IOT Config

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`port`	integer	No	`30883`	Sets the externally reachable MQTT listener port. Publish this port through the swarm and host firewall for device connectivity.
`cert_volume`	string	No	`/opt/scopear/data/mosquitto/certs`	Sets the host directory containing MQTT broker certificates. The directory is required by validation and should contain material matching the IoT CA.

Coturn Config

CoturnConfig configures the TURN relay service for real-time communications.

Appears in - Comms Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`enabled`	boolean	No	`false`	Enables deployment of the Coturn service. Enable it when clients require relay traversal and publish the entire configured UDP range.
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`turn_port`	integer	No	`3478`	Sets the TURN control/listener port. Ensure this port is reachable using the protocol expected by CMS clients.
`turn_min_port`	integer	No	`49152`	Sets the first UDP relay port allocated by Coturn. Open the inclusive range through turn_max_port on every eligible swarm node.
`turn_max_port`	integer	No	`50000`	Sets the final UDP relay port allocated by Coturn. A narrow range can exhaust under concurrent call load.
`turn_username`	string	No	`scopear`	Sets the static username accepted by Coturn. Coordinate changes with CMS/client TURN configuration.
`turn_password`	string	No		Sets the static Coturn password in ordinary configuration. Prefer secrets.coturn_turn_password for production so credentials are not committed with the spec.

Gwl Spec

GwlSpec configures the Guided Work Link service and client application links.

Appears in - Scope AR Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`host`	string	No		Sets the externally reachable GWL hostname. When empty, swarm derives gwl.; ensure DNS and TLS cover the result.
`ios_app_store_id`	string	No	`1099950714`	Sets the iOS App Store identifier used in generated client links. Change it only when directing users to a different managed application listing.
`android_package_name`	string	No	`com.scopear.RemoteAR`	Sets the Android package name used in generated client links. It must match the application installed or published for users.
`microsoft_store_id`	string	No	`9nblggh4v339`	Sets the Microsoft Store identifier used in generated client links. Change it only when deploying a different Windows application listing.
`deferred_deep_linking_firebase_domain`	string	No	`scopear.page.link`	Sets the Firebase domain used for deferred deep links. The domain must be configured in the Firebase project serving the selected mobile applications.

IOT Config

IOTConfig configures MQTT and certificate storage for the IoT service.

Appears in - IOT Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`mqtt`	Config MQTT	No		Configures the MQTT listener used by IoT devices.
`cert_volume`	string	No	`/opt/scopear/data/iot/certs`	Sets the host directory containing IoT service certificates. The directory is required by validation and must exist on eligible swarm nodes.

IOT Spec

IOTSpec configures the ScopeAR IoT service and MQTT integration.

Appears in - Scope AR Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`enabled`	boolean	No	`false`	Enables deployment of the IoT service. Configure CA material and certificate volumes before enabling device connectivity.
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`config`	IOT Config	No		Configures MQTT connectivity and certificate storage. Defaults allocate this object, but callers should avoid an explicit null value.

Minio Config

MinioConfig configures the bundled MinIO object-storage service.

Appears in - Web Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`data_volume`	string	No	`/opt/scopear/data/minio`	Sets the host directory used for persistent MinIO data. Back up this directory and ensure it follows the MinIO task during node maintenance.
`console_enabled`	boolean	No	`true`	Enables the MinIO administrative console. Restrict access to trusted operators when enabled.

MongoDB Config

MongoDbConfig configures the MongoDB service used by WLC2.

Appears in - Wlc 2 Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`enabled`	boolean	No	`true`	Enables the bundled MongoDB service. Disable it only when WLC2 points to an externally managed MongoDB instance.
`replicas`	integer	Yes	`1`	Sets the number of MongoDB swarm tasks. Use one unless replica-set initialization and persistent placement are configured.
`data_volume`	string	No	`/opt/scopear/data/mongodb`	Sets the host directory used for persistent MongoDB data. The directory is required by swarm validation and must follow the stateful task.
`database`	string	No	`wlc2`	Sets the MongoDB database name used by WLC2. Keep this aligned with the reused C2 application connection settings.

Mysql Config

MysqlConfig configures the bundled MySQL database service.

Appears in - Web Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`enabled`	boolean	No	`false`	Enables the bundled MySQL service. Disable it only when CMS is configured to use an external compatible database.
`data_volume`	string	No	`/opt/scopear/data/mysql`	Sets the host directory used for persistent MySQL data. The path is required by swarm validation even when supplied through defaults.

Nginx Config

NginxConfig configures the Nginx service used by the web stack.

Appears in - Web Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`data_volume`	string	No		Sets the host directory mounted for Nginx persistent or generated data. Ensure the directory is available on each node eligible to run Nginx.

Postgres Config

PostgresConfig configures the bundled PostgreSQL swarm service.

Appears in - Superset Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`data_volume`	string	No	`/opt/scopear/data/postgres`	Sets the host directory used for persistent PostgreSQL data. Back up the directory and ensure it follows the stateful task during node maintenance.

Redis Config

RedisConfig configures a Redis service used by ScopeAR components.

Appears in - Comms Config - Web Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`data_volume`	string	No	`/opt/scopear/data/redis`	Sets the host directory used for persistent Redis data. Multiple Redis consumers may share this default path, so avoid scheduling conflicting instances on one node.

Release Channel

Underlying type: string

ReleaseChannel identifies a supported ScopeAR image release stream.

Appears in - Scope AR Spec

VALUE	DESCRIPTION
`stable`	
`rc`	
`unstable`	
`develop`	
`on-prem`	

Resource

Resource defines CPU and memory reservations used by swarm placement.

Appears in - Resources

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`cpus`	string	No		Reserves CPU capacity using Docker's decimal CPU format. Ensure eligible nodes have enough unreserved capacity for every replica.
`memory`	string	No		Reserves task memory using a Docker-compatible byte value. Excessive reservations can leave services pending even when physical memory appears available.

Resource Limit

ResourceLimit defines CPU and memory ceilings for a swarm service task.

Appears in - Resources

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`cpus`	string	No		Caps CPU capacity using Docker's decimal CPU format. Values are passed to swarm deployment as CPU units rather than literal nanoseconds.
`memory`	string	No		Caps task memory using a Docker-compatible byte value. Include an explicit unit to avoid an unexpectedly small limit.

Resources

Resources groups swarm resource limits and scheduling reservations.

Appears in - AI Config - Cms Config - Common - Coturn Config - `DtoolConfig` - Gwl Spec - IOT Spec - Minio Config - Mongo Db Config - Mysql Config - Nginx Config - Postgres Config - Redis Config - Sidekiq Config - Superset Spec - Wlc 2 Api Config - Wlc 2 Client Config - Wlc 2 Model Converter Config - Wlc 2 Project Packager Config - Wlc 2 Spec - Wlc 2 Webapp Config - Zeus Config

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`limits`	Resource Limit	No		Caps CPU and memory available to each service task. Omitted values leave the corresponding Docker swarm limit unset.
`reservations`	Resource	No		Reserves CPU and memory for placement of each service task. A swarm node must satisfy these values before the scheduler places the task.

ScopeAR Spec

ScopeARSpec defines the desired state of a ScopeAR swarm deployment.

Appears in - ScopeAR (Swarm)

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`hostnames`	string array	Yes		Lists ingress hostnames, with the first hostname treated as primary. Derived service hosts and certificates use the primary hostname, so order is significant. [*validation:* `minItems=1`; *example:* `["scopear.example.com"]`]
`extra_hosts`	string array	No		Adds colon-separated hostname-to-IP mappings to generated service definitions. Use this for names that swarm nodes or containers cannot resolve through DNS. [*example:* `["storage.example.com:10.0.0.24"]`]
`subdomainSuffix`	string	No		Appends a suffix to generated ingress service subdomains. Ensure the resulting names are covered by DNS and the configured TLS certificate.
`channel`	Release Channel	No	`stable`	Selects the release channel used when resolving ScopeAR images. On-prem installations commonly use stable or on-prem; unsupported values fail validation.
`on-prem`	boolean	No	`true`	Enables defaults intended for self-hosted and air-gapped deployments. In on-prem mode, public registry prefixes are stripped unless explicitly overridden.
`replicas`	integer	No	`1`	Sets the fallback replica count for services without a component override. Component-level replicas take precedence; account for swarm node capacity before increasing it.
`imageRegistry`	string	No		Prefixes generated image references with a custom container registry. Use a registry reachable by every swarm node and coordinate this with stripPublicRegistry.

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`stripPublicRegistry`	boolean	No	`true when on-prem, otherwise false`	Removes the public registry host from generated image references. This defaults true for on-prem mode so references match images imported with docker load.
`secrets_path`	string	No	`/opt/scopear/secrets`	Sets the host directory containing secret files mounted into services. Protect this directory on every eligible swarm node and keep its contents out of source control.
`images`	Image array	No		Overrides image metadata inherited from the k8s.scopear.com API. Each entry controls image resolution for the corresponding service in generated swarm manifests.
`logLevel`	string	No	`info`	Sets the global application log level. Individual reused service configurations may expose more specific logging controls.
`tlskey`	string	No		Sets the legacy host filesystem path to the TLS private key PEM. Prefer traefik.tls.static_host_key_path; Traefik uses this only when the newer field is empty.
`tlspem`	string	No		Sets the legacy host filesystem path to the TLS certificate PEM. Prefer traefik.tls.static_host_cert_path; Traefik uses this only when the newer field is empty.
`ui_username`	string	No	`scopear-onprem`	Sets the username for the swarm configuration UI. Change the default for exposed installations and manage the password through ui_passwordFile.
`ui_passwordFile`	string	No	`.scopear-ui-password`	Sets the path to a plaintext file containing the swarm configuration UI password. Restrict host permissions on this file; the path is resolved where scopearctl runs.
`traefik`	Traefik Spec	No		Configures Traefik ingress and TLS for the swarm deployment. Choose either ACME issuance or static host-mounted certificates and ensure ports are reachable.
`secrets`	Secrets Spec	No		Supplies credentials and key material consumed across swarm services. Values are rendered into deployment secrets; do not commit production values to source control.
`comms`	Comms Spec	No		Configures the Zeus and Coturn communications services. Enable and expose these services only when calls or tunneled communication are required.
`web`	Web Spec	No		Configures CMS, Sidekiq, MySQL, Nginx, Redis, and MinIO web services. This section also reuses the detailed CMS settings from k8s.scopear.com.
`gwl`	Gwl Spec	No		Configures the Guided Work Link service and client-store metadata. Its host is derived from the primary hostname when no explicit host is supplied.

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`wlc2`	Wlc 2 Spec	No		Configures WorkLink Create 2 services, MongoDB, and shared C2 settings. This section reuses C2 application configuration from k8s.scopear.com.
`iot`	IOT Spec	No		Configures the IoT service, MQTT listener, and certificate mounts. Certificate directories must exist on swarm nodes that may run these services.
`ai`	AI Spec	No		Configures the AI service and its reused k8s.scopear.com application settings. Enable it only after configuring storage, database access, and required API secrets.
`superset`	Superset Spec	No		Configures Superset analytics and its PostgreSQL dependency. Enabling analytics requires matching database and OAuth secrets.

ScopeAR (Swarm)

ScopeARSwarm is the Go root type for the ScopeAR custom resource. The serialized Kubernetes YAML kind is ScopeAR, not ScopeARSwarm.

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`apiVersion`	`string`	Yes		Value must be `swarm.scopear.com/v1`.
`kind`	`string`	Yes		YAML `kind` is `ScopeAR` (Go type is `ScopeARSwarm`).
`metadata`	ObjectMeta	Yes		Refer to Kubernetes API documentation for fields of `metadata`.
`spec`	Scope AR Spec	No		Describes the desired ScopeAR swarm deployment. Defaults are applied before generated manifests are validated and rendered.

Secrets Spec

SecretsSpec defines credentials and key material used across swarm services.

Appears in - Scope AR Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`fireworks_api_key`	string	No		AI Service Secrets Supplies the Fireworks AI API key to the AI service. Set it only when Fireworks-backed models are configured and rotate it through your secret workflow.
`helicone_api_key`	string	No		Supplies the Helicone API key used for AI request observability. It is required only when the configured Helicone endpoint authenticates requests.

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`openai_api_key`	string	No		Supplies the OpenAI API key to the AI service. Set it when OpenAI-backed queries are enabled and avoid storing production values in source control.
`storage_access_key`	string	No		Supplies the object-storage access key used by AI and related services. The value must match the selected S3-compatible provider, commonly the bundled MinIO instance.
`storage_secret_key`	string	No		Supplies the object-storage secret key paired with storage_access_key. Rotate both credentials together and ensure all dependent services receive the updated values.
`minio_root_user`	string	No		MinIO Secrets Sets the root username for the bundled MinIO service. Use the same value in clients that administer or bootstrap on-prem object storage.
`minio_root_password`	string	No		Sets the root password for the bundled MinIO service. Use a strong unique value and do not expose the root credential to routine application clients.
`zeus_encryption_key`	string	No		Zeus/Comms Secrets Supplies the private encryption key used by the Zeus communications service. Keep it paired with zeus_encryption_cert and preserve it when migrating an existing deployment.
`zeus_encryption_cert`	string	No		Supplies the certificate paired with the Zeus encryption key. The certificate and private key must form a valid pair for communications startup.
`cms_password`	string	No		CMS Secrets Supplies the CMS database password. It must match the password configured for the CMS database user.
`cms_secret_key`	string	No		Supplies the CMS application secret key used for cryptographic signing. Preserve this value across upgrades to avoid invalidating signed application state.
`cms_secret_token`	string	No		Supplies the CMS service-to-service authentication token. Configure clients with the same token and rotate them together.
`cms_smtp_password`	string	No		Supplies the password for the SMTP account configured in web.config. Leave it empty when SMTP authentication is not required.
`cms_system_encryption_key`	string	No		Supplies the CMS system encryption key for encrypted persisted values. Changing it can make existing encrypted data unreadable, so retain it during upgrades and restores.
`cms_jwt_secret`	string	No		Supplies the CMS JWT signing secret. Rotating it invalidates existing tokens and requires coordinated client reauthentication.

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`gql_secret_token`	string	No		Supplies the token used to authenticate GraphQL service requests. Keep producers and consumers synchronized when rotating this value.
`iot_ca_cert`	string	No		IoT Secrets Supplies the certificate-authority certificate for IoT device certificates. Mount it with the matching private key and preserve the CA while issued devices remain active.
`iot_ca_key`	string	No		Supplies the private key for the IoT certificate authority. Restrict access carefully; compromise permits issuance of trusted device certificates.
`mysql_root_password`	string	No		Database Secrets Sets the administrative root password for the bundled MySQL service. Use it only for database administration and keep it distinct from the application password.
`mysql_password`	string	No		Sets the MySQL password used by ScopeAR application services. It must match the application database account initialized by the deployment.
`postgres_password`	string	No		Sets the PostgreSQL password used by Superset and related services. It must match the account referenced by the Superset database configuration.
`superset_secret_key`	string	No		Superset Secrets Supplies the Superset application secret used for sessions and signing. Preserve it across restarts and upgrades to avoid invalidating active sessions.
`superset_oauth_client_secret`	string	No		Supplies the OAuth client secret used by Superset authentication. It must match the client registered with the configured identity provider.
`coturn_turn_password`	string	No		Turn Secrets Supplies the TURN shared password used by Coturn clients. Prefer this secret field over embedding credentials in ordinary communications configuration.
`gwl_airbridge_token`	string	No		GWL Secrets Supplies the Airbridge integration token used by Guided Work Link. Leave it empty when the Airbridge integration is not enabled.

Sidekiq Config

SidekiqConfig configures the Sidekiq background worker service.

Appears in - Web Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.

Superset Config

SupersetConfig configures Superset's Redis databases and CMS database identity.

Appears in - Superset Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`redis_celery_db`	integer	No	`3`	Selects the Redis logical database used by Celery queues. Keep it distinct from other Redis consumers sharing the same instance.
`redis_results_db`	integer	No	`2`	Selects the Redis logical database used for Celery results. Do not reuse redis_celery_db or an index allocated to another ScopeAR service.
`cms_db_user`	string	No	`superset`	Sets the database username Superset uses to read CMS data. The matching account and password must exist in the CMS database.

Superset Spec

SupersetSpec configures the Superset analytics service and PostgreSQL dependency.

Appears in - Scope AR Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`enabled`	boolean	No	`false`	Enables deployment of the Superset analytics service. Configure database and application secrets before enabling it.
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`config`	Superset Config	No		Supplies Superset application and Redis database settings. Defaults allocate this object; avoid an explicit null value when enabling Superset.

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`postgres`	Postgres Config	No		Configures the bundled PostgreSQL service used by Superset. Disable or omit deployment externally only when Superset is pointed at a managed database.

Traefik Spec

TraefikSpec configures the Traefik reverse proxy for swarm ingress.

Appears in - Scope AR Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`data_volume`	string	No	`/opt/scopear/data/traefik`	Sets the host directory bind-mounted at /etc/traefik. The directory persists ACME state and must be available on the node running Traefik.
`tls`	Traefik TLS Spec	No		Selects and configures certificate management for ingress. Static certificates use host paths, while Let's Encrypt requires public HTTP-01 reachability.

Traefik TLS Provider

Underlying type: string

TraefikTLSProvider identifies how Traefik obtains TLS certificates.

Appears in - Traefik TLS Spec

VALUE	DESCRIPTION
`letsencrypt`	
`static`	

Traefik TLS Spec

TraefikTLSSpec configures Let's Encrypt or host-mounted PEM certificates for Traefik.

Appears in - Traefik Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`provider`	Traefik TLS Provider	No	`static`	Selects static host-mounted certificates or Let's Encrypt ACME issuance. Static mode falls back to legacy top-level tlskey and tlspem paths when its paths are empty.
`acme_email`	string	No		Overrides the contact email used for ACME registration. When omitted, the deployment derives admin@.

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`acme_ca_server`	string	No		Sets a non-default ACME CA directory URL. Use the Let's Encrypt staging endpoint while testing to avoid production rate limits.
`acme_storage`	string	No	`/etc/traefik/acme.json`	Sets the path inside the Traefik container for persisted ACME account data. Keep this path within the /etc/traefik bind mount and protect the resulting file.
`static_host_cert_path`	string	No		Sets the host path to the static certificate PEM mounted read-only. Static TLS requires both certificate and key paths; otherwise ingress falls back to ACME.
`static_host_key_path`	string	No		Sets the host path to the static private key PEM mounted read-only. Restrict file permissions and ensure the path exists on the Traefik node.

Web Spec

WebSpec configures the web-facing ScopeAR services and their dependencies.

Appears in - Scope AR Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`cms`	Cms Config	Yes		Configures the CMS application service. CMS persistence and replicas are managed specifically for the swarm deployment.
`sidekiq`	Sidekiq Config	Yes		Configures the Sidekiq background worker service. Size replicas and resources to match CMS job volume.
`config`	Cms Config	Yes		Supplies CMS application settings reused from k8s.scopear.com. Swarm defaults are applied to nested logging, Coturn, Zeus, SMTP, and storage fields.
`mysql`	Mysql Config	Yes		Configures the bundled MySQL database service. Disable it only when application database connectivity is provided another way.
`nginx`	Nginx Config	Yes		Configures the Nginx web proxy service. Resource and persistence settings apply to generated swarm tasks.
`redis`	Redis Config	Yes		Configures the Redis service shared by web workloads. Its persistent data directory is also reused by default communications settings.
`minio`	Minio Config	Yes		Configures the bundled MinIO object-storage service. Storage credentials are supplied through the top-level secrets section.

WLC2 Api Config

Wlc2ApiConfig configures the WLC2 API swarm service.

Appears in - Wlc 2 Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.

WLC2 Client Config

Wlc2ClientConfig configures the WLC2 client swarm service.

Appears in - Wlc 2 Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.

WLC2 Model Converter Config

Wlc2ModelConverterConfig configures WLC2 model-conversion workers.

Appears in - Wlc 2 Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.

WLC2 Project Packager Config

Wlc2ProjectPackagerConfig configures WLC2 project-packaging workers.

Appears in - Wlc 2 Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.

WLC2 Spec

Wlc2Spec configures WorkLink Create 2 services and their dependencies.

Appears in - Scope AR Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`api`	Wlc 2 Api Config	Yes		Configures the WLC2 API service.
`client`	Wlc 2 Client Config	Yes		Configures the WLC2 client service.
`webapp`	Wlc 2 Webapp Config	Yes		Configures the WLC2 browser application service.
`model_converter`	Wlc 2 Model Converter Config	Yes		Configures the model-conversion worker service. Allocate sufficient CPU and memory for the expected CAD model workload.
`project_packager`	Wlc 2 Project Packager Config	Yes		Configures the project-packaging worker service. Scale it independently when package generation becomes a queue bottleneck.
`config`	`C2Config`	Yes		Supplies C2 application settings reused from k8s.scopear.com. Swarm defaults initialize storage and CAD-pipeline values on this object.
`mongodb`	Mongo Db Config	Yes		Configures the MongoDB service used by WLC2.

WLC2 Webapp Config

Wlc2WebappConfig configures the WLC2 web application swarm service.

Appears in - Wlc 2 Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.

Zeus Config

ZeusConfig configures the Zeus communications service.

Appears in - Comms Spec

FIELD	TYPE	REQ	DEFAULT	DESCRIPTION
`enabled`	boolean	No	`true`	Enables the Zeus communications service. Zeus is enabled by swarm defaults; disable it only when communications are provided externally.
`replicas`	integer	No	`1`	Sets the desired number of swarm service replicas. A component value overrides the top-level replica fallback; zero disables scheduled tasks for that service.
`resources`	Resources	No		Resources describes the compute resource requirements. Swarm uses these values to place containers; reservations affect scheduling while limits cap usage.
`config`	Comms Config	No		Supplies Zeus dependency configuration. The current swarm configuration controls its Redis service.

Samples

The following sample manifests are also included as standalone YAML files under `samples/` in the documentation package.

`scopear_swarm_env_sample.yaml`

```
# Generated by hack/generate-docs-samples — do not edit by hand.
# Kitchen-sink example with SetDefaults() applied. Delete or change values as needed.
# Docker Swarm on-premises ScopeAR environment (channel: on-prem).
apiVersion: swarm.scopear.com/v1
kind: ScopeAR
metadata:
  name: scopear-swarm
  namespace: scopear
spec:
  ai:
    ai:
      resources: {}
    config:
      airbrake: {}
      helicone_api_url: https://oai.hconeai.com/v1
      mongodb:
        database: ai
        host: mongodb
        port: 27017
      qdrant_url: http://qdrant:6333
      storage:
        minio_server_address: minio.swarm.localdomain
        minio_server_port: 443
        minio_server_use_ssl: true
        provider: minio
      true_openai_queries: false
    data_volume: /opt/scopear/data/ai
    enabled: false
  channel: on-prem
  comms:
    coturn:
      enabled: false
      resources: {}
      turn_max_port: 50000
      turn_min_port: 49152
      turn_password: Augmenttheworld2025!
      turn_port: 3478
      turn_username: scopear
    zeus:
      config:
        redis:
          data_volume: /opt/scopear/data/redis
          resources: {}
      enabled: true
      resources: {}
  gwl:
    android_package_name: com.scopear.RemoteAR
    deferred_deep_linking_firebase_domain: scopear.page.link
```

```
    ios_app_store_id: "1099950714"
    microsoft_store_id: 9nblggh4v339
    replicas: 1
    resources: {}
  hostnames:
    - scopear.example.com
  iot:
    config:
      cert_volume: /opt/scopear/data/iot/certs
      mqtt:
        cert_volume: /opt/scopear/data/mosquitto/certs
        port: 30883
    enabled: false
    resources: {}
  logLevel: info
  on-prem: true
  secrets: {}
  secrets_path: /opt/scopear/secrets
  stripPublicRegistry: true
  superset:
    config:
      cms_db_user: superset
      redis_celery_db: 3
      redis_results_db: 2
    enabled: false
    postgres:
      data_volume: /opt/scopear/data/postgres
      resources: {}
    resources: {}
  traefik:
    data_volume: /opt/scopear/data/traefik
    tls:
      acme_storage: /etc/traefik/acme.json
      provider: static
  web:
    cms:
      data_volume: /opt/scopear/data/cms
      replicas: 1
      resources: {}
    config:
      admin: {}
      airbrake: {}
      auth_method: db
      coturn:
        enable_multi_region: false
        enabled: false
        label: us-east-1
        tunnel_external_host: proxy-aws-us.scopear.com
```

```
        tunnel_external_port: 443
        turn_external_host: proxy-aws-us.scopear.com
        turn_external_port: 3478
        turn_protocol: udp
      hubspot: {}
      ldap: {}
      log_level: info
      metabase: {}
      mysql: {}
      open_id: {}
      redis: {}
      sentry: {}
      smtp:
        auth_type: plain
        email_format: all
        enable_starttls_auto: true
        openssl_verify_mode: none
      storage:
        file_storage: s3
      twilio: {}
      zeus:
        enabled: false
        external_host: master.scopear.com
        external_port: 24000
    minio:
      console_enabled: true
      data_volume: /opt/scopear/data/minio
      resources: {}
    mysql:
      data_volume: /opt/scopear/data/mysql
      resources: {}
    nginx:
      resources: {}
    redis:
      data_volume: /opt/scopear/data/redis
      resources: {}
    sidekiq:
      replicas: 1
      resources: {}
  wlc2:
    api:
      replicas: 1
      resources: {}
    client:
      replicas: 1
      resources: {}
    config:
      cad_pipeline:
```

```
      processing_mode: none
      query_time: 10000
    mongodb: {}
    storage:
      bucket_name: wlc2
  model_converter:
    replicas: 1
    resources: {}
  mongodb:
    data_volume: /opt/scopear/data/mongodb
    database: wlc2
    enabled: true
    replicas: null
    resources: {}
  project_packager:
    replicas: 1
    resources: {}
  resources: {}
  webapp:
    replicas: 1
    resources: {}
```