Lumen® CDN
Configuration Management v3 (CMv3)

What is Configuration Management v3?

Configuration Management v3 is Lumen CDN's latest API and UI to be released in Q1 2022. This new era for the Lumen CDN brings a brand new UX built on a modern stack, simplified RESTful API endpoints, and expanded configuration capabilities.

Why move to Configuration Management v3?

CMV3 is a step-change improvement in your CDN service.

- **Speed and autonomy**: CMv3 offers you 100% autonomy to manage the full caching configuration lifecycle. Features are designed to span the vast majority of simple and complex use cases, with more use cases being added iteratively every quarter.

- **Modern UX**: An entirely redesigned user interface will replace the Lumen Media Portal in mid-2022. New documentation will be available across the portfolio.

- **Improved API experience:** Brand new API with Restful endpoints designed for ease of use. Our API is easily discoverable via documentation, openAPI specification and UI. The configuration schema has been streamlined for easier discovery and parsing.

- **Go live easily:** New staged activation feature allowing you to safely deploy new versions of your Configurations, by testing them on a small percentage of your traffic (canary deployment principle).

- **Optimal performance**: Get full access to software upgrades in our network for enhanced performance

What is the migration process?

The migration process has been designed to be transparent for you and your production traffic.
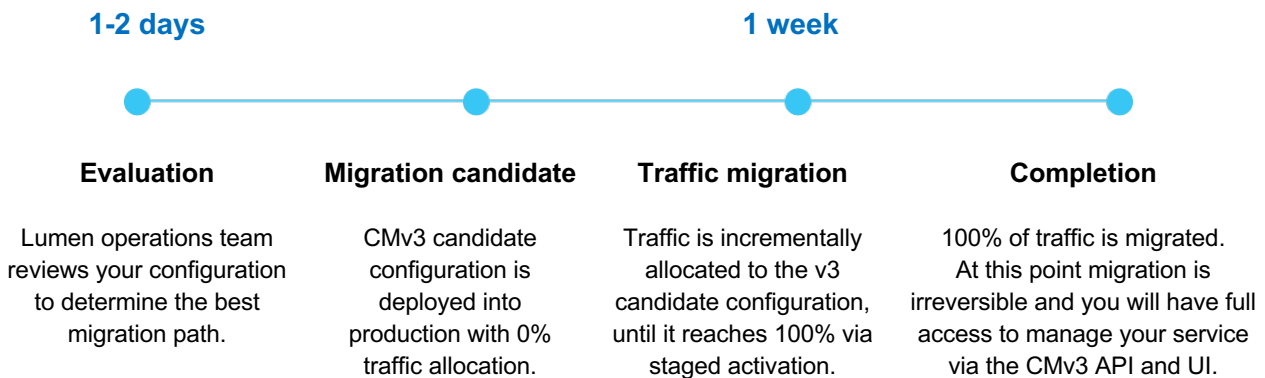
Dedicated Lumen CDN operations staff will work with you to determine a timeline for the migration and support you throughout the process.

Once the migration starts, our operations team will first re-create your configuration in a v3 format and test in thoroughly in our labs.

Operations teams will then gradually deploy the candidate CMv3 configuration using our new staged activation capability (described below).

Note, however, that to provide a frictionless experience, changes to your configuration must be paused until migration is complete.

LUMEN®
TECHNOLOGIES

Migration timeline and workflow

1-2 days

1 week

Evaluation

Lumen operations team reviews your configuration to determine the best migration path.

Migration candidate

CMv3 candidate configuration is deployed into production with 0% traffic allocation.

Traffic migration

Traffic is incrementally allocated to the v3 candidate configuration, until it reaches 100% via staged activation.

Completion

100% of traffic is migrated. At this point migration is irreversible and you will have full access to manage your service via the CMv3 API and UI.

Post-migration

CMv3 will grow based on your and other customers' business needs. We expect to release new configuration features and capabilities on a regular basis throughout the year.

What's new?

Staged activation

With Staged Activation, Lumen CDN makes it easier to deploy new versions of your configurations in a safe and controlled manner, inspired by the Canary Release development processes. Staged activation allows for two production environment "slots:"

- A **"current"** main slot stores your current configuration settings.
- An additional "**candidate**" slot can be used to introduce a new configuration into production and gradually move traffic to it.

You can start with 0% traffic, and then gradually increase the candidate's share to 5%, 10%, 50%, and up to 100% when you are confident all works as expected with no regressions. Monitoring and data reporting tools in the edge delivery portal help facilitate this process. Once you have validated your candidate configuration, you can promote to the "current" environment slot and use it as your main configuration.

LUMEN®
TECHNOLOGIES
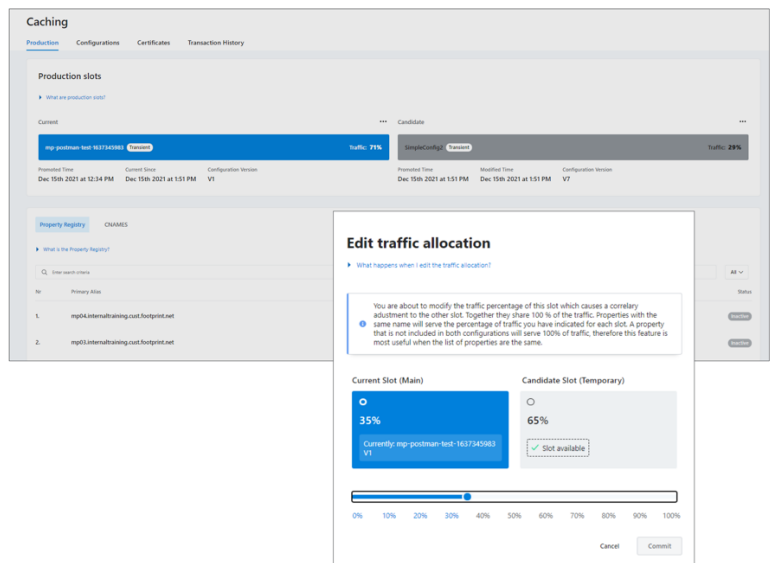
Configuration management

Our mechanisms for managing configurations have been designed with your customization needs in mind.

We have introduced **Match Rules**, a condition-based system to activate features and modify requests with high granularity. You can add Match Rules on parameters related to client request, origin request, and origin response today. Client response match rules will be available soon.

We have also added new capabilities to our property definitions, including support for failover origins; simpler HTTPS traffic management; re-usable geo, IP, and token list definitions; and more.

Features and capabilities

Property definition features: origin and aliases

Basic property details, default protocol settings, traffic type, as well as primary/secondary aliases have been carried over into CMv3. In addition, the following features have been added or improved:

Feature	Description
Multi-origin failover	Add fail-over origins to be used in case the first one is not accessible.
Origin authentication via AWS v4	Generate authentication headers for AWS S3-compatible origins.
Origin - ignore invalid certificate	Fill from an origin instead even if the origin's certificate is expired or otherwise invalid.
Origin webroot	Fill from a specific directory on the origin that is not part of the original or processed URI.
Origin fill protocol	HTTP or HTTPS
Alias overrides	Secondary aliases no longer need to conform to primary property default settings. Service type, protocol settings, and certificate referenced may be unique for one or a group of secondary aliases.

Conditional match rule capabilities

With match rules, you can now create rules based on conditions from nearly any request and responds metadata: path, headers, query string, etc. Complex logic can be created with the help of AND/OR operators, RegExp matches, and more functionalities in our in-house conditions building language.

Feature	Description
Declare a set of sequential match rules for a specific request block type (stage)	Match rules are directly mapped to the stages of a CDN-governed request: client request, origin request, and origin response; client response (coming soon).
Add/manage match expression	Set granular match criteria for a set of behaviors.
Path match	Add/manage match expression based on path.

Header match	Add/manage match expression based on headers.
Query string match	Add/manage match expression based on query strings.
Multi-expression support	Add/Manage match expression for a single set of match rules with "And" and "Or" operators
Multi-match-block support	Match blocks are run sequentially, as are the rules within them.
Reference reusable objects for a given match rule feature	Some behaviors (features) refer to re-usable objects, such as geo region lists, etc.
Reporting overrides	By default, all usage is reported under the primary alias - this feature, implemented through a match rule directive, allows customizations on how traffic is reported (to specified aliases or groups thereof).
Origin - follow redirects	When origin returns 3xx in response to the CDN request, send another request to the URL from the location header instead of serving 3xx to the client.

Match rule features

The following functions can be declared within a property match rule. Note that + indicates improved.

Type	Feature	Description
Cache control	**Cache control header override (ccho)**	Overrides the cache-control header provided by the origin or the default system behavior.
	Negative TTL	Overrides the default system setting for caching "negatively cacheable" responses (a defined set of HTTP response codes).
	Failed refresh TTL	Specifies for how long the system should serve a stale resource after a refresh of it failed and before trying again.
	Default cache control policy	Defines for how long the system should cache a resource if the origin did not provide an explicit policy.
	Disable If-None-Match header	When requesting a resource from upstream servers or origin CDN will use If-None-Match header.
	Stale content control	Serve stale content based on conditions you specify. Multiple conditions are assumed to be an "or" operator.
	Query String Handling (QSH)	Ignore all or specific query strings to cache your content. Override our CDN's default behavior of including all query string parameters to compute the caching key.

LUMEN®
TECHNOLOGIES

Access control	**Deny policies**	Generate a 403 response. Allows you to reject a request or redirect it to another URL.
	Geo restrictions	Supports allowing or rejecting a request based on the client's country code.
	IP restrictions	Supports allowing or rejecting a request based on the client's IP.
	Token authentication	Defines token generation characteristics for a protected resource that is matched against a request with an embedded token.
	Allow compress	Allows the CDN to automatically Gzip your content and cache a compressed version on the CDN edges. For content already Gzipped at the origin, allows the CDN to unzip your content on-the-fly and serve it requested.
Request manipu-lation	**Generated response**	Create an auto-generated response template that is used instead of the one provided by the cached resource or the origin.
	Request / response header manipulation	Override request or response headers.
	Serve 200 for 416	When a client range request is out of range for the resource served, return the full resource instead of an HTTP 416 error.
	URI rewrite	Change the URI for further processing.
Media-specific	**Large objects**	Optimizes the CDN requesting and caching large objects in chunks. This is controlled via Traffic Types.
Logging	**Custom log data**	Populate a custom field in the log with specific data.
	Reporting overrides	Report some requests separately within a single property.
Lua	**Lua scripts – request & responses**	Executes a specified Lua script while processing a request / response. Only Lumen support engineers can create Lua scripts; however, you decide the variable values and where to apply them.
Origin selection	**Origin fill policy (use alternate)**	Use an alternate origin definition that overrides the default one set up in the property.

LUMEN®
TECHNOLOGIES