

Optional validation

Title (Goal)	Optional content and structural validation
Primary Actor	Information architect, developer
Scope	Component
Level	Summary
Author	Stefano Cossu
Story (A paragraph or two describing what happens)	Make validation optional for some or all properties

This use case exemplifies a repository in which the API-X server should issue a warning, or otherwise modify a resource in a way that does not prevent its being persisted in Fedora, if some validation rules are not met.

See [Enforce validation across repository](#) and [AIC Use Case: Content and Structural Validation](#) for more detailed examples of validation rule enforcement.

Example 1: all rules optional

1. User uploads image file and metadata to create a new resource of type `cma:Image`
2. Validation rules for `cma:Image` indicate that a resource **SHOULD** have at least one `myns:creator` property
3. If user-provided metadata do not include a `myns:creator` property, the validation engine forwards the request to a special process to handle the request. This process can perform any of the following actions:
 - a. Simply forward the request to Fedora or a pre-ingest pipeline as if validation passed
 - b. Forward the request, and return the subsequent request headers and body with additional headers or body issuing a warning that a certain validation rule did not pass
 - c. Modify the resource, e.g. by adding a `myns:Orphantype` to the resource so as to indicate that the creator property did not pass optional validation.
4. If user-provided metadata include a `myns:creator` property, the request is forwarded to Fedora or to a pre-ingest pipeline and the server returns the return headers and body of this subsequent request

Example 2: Some rules optional

Assuming the same validation rules and ingest scenario as Example 1:

1. In addition to a `myns:creator` recommended property, the image **MUST** have a `myns:uid` property
2. If the request does not include a valid `myns:uid` property, the request fails and the server returns a 4XX response explaining the reason for the failure
3. If the request includes a valid `myns:uid` property, steps 3. and 4. in example ones are applied

Personal consideration: this scenario may actually be merged with the [Enforce validation across repository](#) one. Instead of distinguishing between mandatory and optional rules, the configuration may indicate *what* to do if a validation rule does not pass (possibly making a 4XX response the default). This allows for a flexible combination of mandatory and recommended rules.