

# High-level Requirements

## Requirements

1. Infrastructure
  - a. The API-X framework shall be deployable on a separate host from Fedora
    - HTTP requests routed to these instance(s) before being forwarded to their ultimate destination (Fedora, backend service, etc)
  - b. It shall be possible to deploy multiple instances of the API-X framework for the sake of scalability of high availability
  - c. The framework shall allow extensions to be hot deployable, configurable, and removable at runtime
  - d. Extension services are exposed as HTTP endpoints
  - e. API-X shall provide a specification that extensions will implement in order to support high level requirements such as service discovery, binding, etc
2. Patterns
  - a. API-X shall support a filtering pattern in extensions, where an HTTP request to an existing Fedora URI (or response) is routed through an extension and possibly modified in the process
    - The extension is free to do whatever it wants to the request, including route it to an alternate destination, or respond directly
  - b. API-X shall support exposing new URIs associated with individual repository resources, bound to individual extensions
  - c. API-X shall support exposing URIs associated with the global repository a whole, bound to individual extensions
  - d. API-X shall support a proxy pattern in extensions, where requests to a URI exposed by an extension are proxied to some other web service
    - This can also be seen as implicitly URL rewriting
3. Service Discovery & Binding
  - a. It shall be possible to enumerate the list of all extensions deployed and active within the framework
  - b. It shall be possible to enumerate the list of extensions that provide services on a given object in the repository
  - c. For any given object in the repository, it shall be possible to enumerate all URIs exposed by extensions
  - d. API-X shall provide a mechanism for HTTP clients to discover all URIs exposed by extensions on a given object
  - e. It shall be possible to bind extensions to all objects in the repository
  - f. It shall be possible to bind extensions to specific objects in the repository, based on their rdf:type

## Design considerations

These considerations are non-binding, and represent topics of relevance to the design process. Decisions related to any of these items are of interest to the stakeholders, and should be communicated where appropriate.

1. Would it be possible for extensions to be deployable "in close proximity to" an instance of a Fedora implementation (e.g. bundled together as part of an application), for the sake of optimization and/or efficiency? Can extensions be specified in a way that this is possible?
2. Would it be possible for The API-X framework to be deployable as an optional add-on module to current reference Fedora implementation,?
  - This would allow standalone Fedora installs to deploy API-X extensions within them
3. Will the API-X framework to be deployable as OSGi bundles?
  - a. The API-X framework could be available as a Karaf Feature, for ease of deployment into Apache Karaf
  - b. Will Individual extensions shall be deployable as OSGi bundles?
    - i. If so, will they be configurable through OSGi's own ConfigurationAdmin service?
    - ii. Will creating multiple instances of extensions via ConfigurationAdmin factory pattern shall be allowable?
      - i.e separately configured instances of extensions may be created via means defined by OSGi.
4. Will the specification for API-X include an abstraction layer used for communicating with Fedora?