

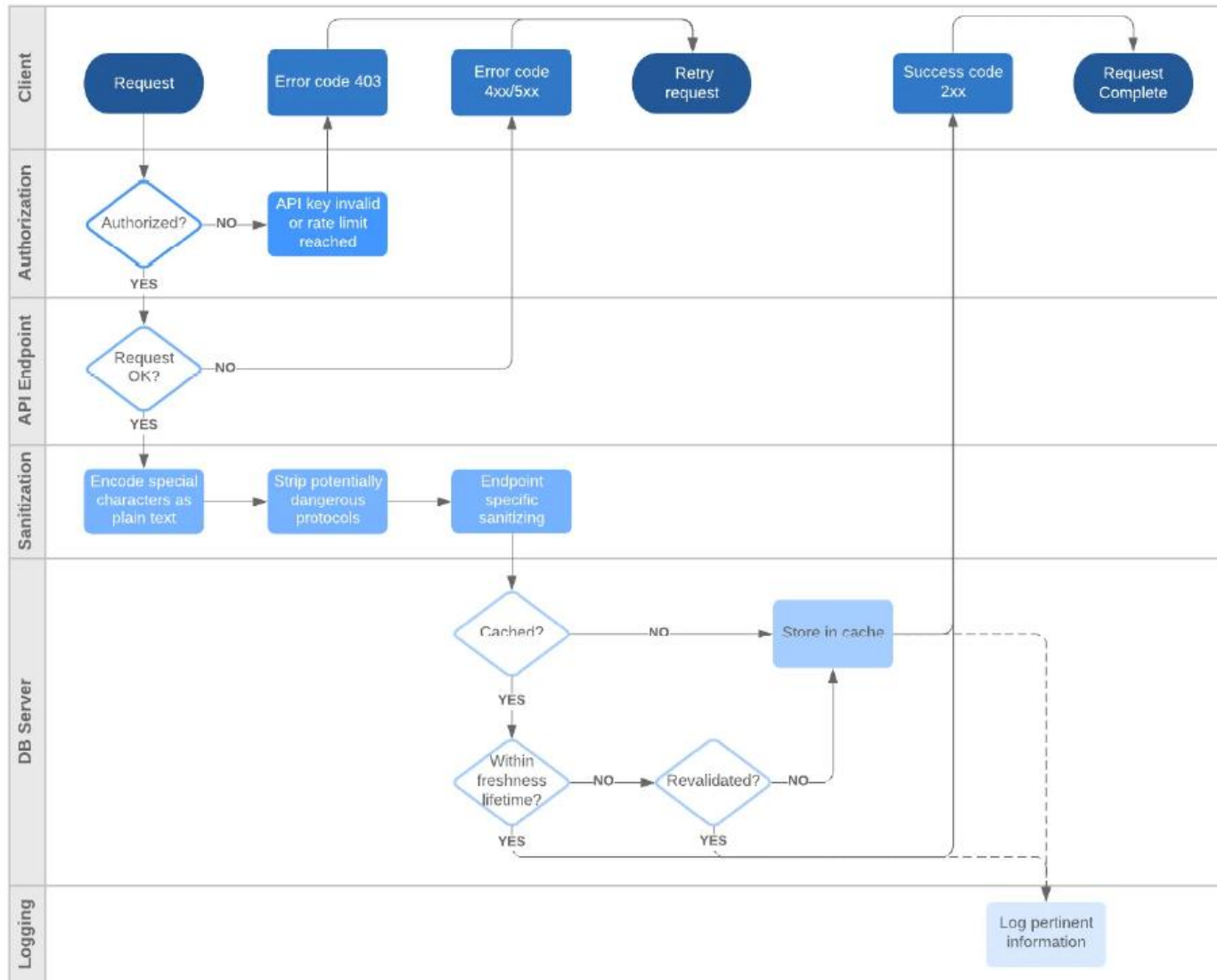
# Application Programming Interfaces (APIs)

What are APIs

APIs and NIEM

Opportunities to Standardize

An API is a messenger that sends a request to a provider you want data from and then delivers the response back to you.



Example Flow chart:

The software that needs to access information (i.e., X hotel room rates for certain dates) or functionality (i.e., a route from point A to point B on a map based on a user's location) from another software, calls its API while specifying the requirements of how data/functionality must be provided. The other software returns data/functionality requested by the former application.

Software products exchange data and functionalities via machine-readable interfaces – *APIs (application programming interfaces)*

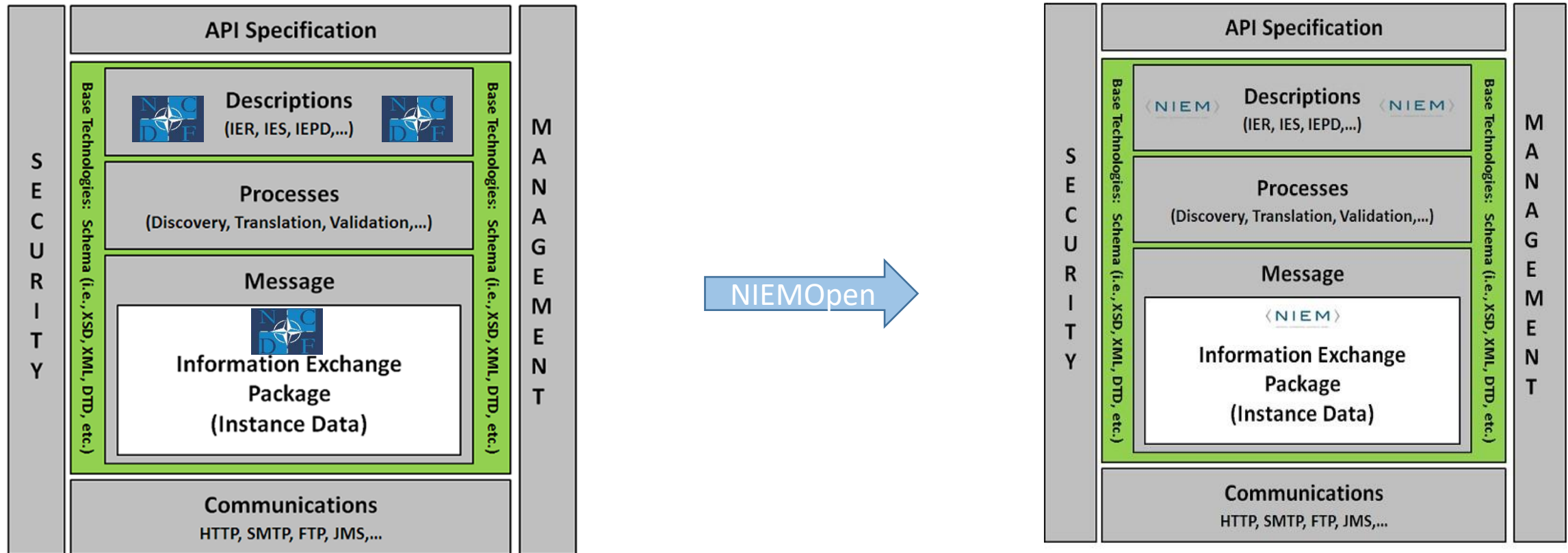


Figure 2: Generic API Architecture

# APIs

- Application programming interfaces consist of two components
  - Technical specification describing the data exchange options between solutions with the specification done in the form of a request for processing and data delivery protocols
  - Software interface written to the specification that represents it

# Standardizing APIs

- Management
  - Documentation
  - Registering (Data.gov)
- Security
  - Zero Trust
- Payload
  - NIEM Conformant
  - Minimum Essential Metadata

# Minimum Documentation

- a quick start guide
- authentication information
- explanations for every API call (request)
- examples of every request and return with a response description, error messages, etc.
- samples of code for popular programmatic languages like Python, [Java](#), JavaScript, or PHP;
- tutorials
- . . . .