REST Essentials

CSCI E-94

Fundamentals of Cloud Computing - Azure

Joseph Ficara

Agenda

- REST Essentials
  - Overview
  - URI Design
  - Extensibility

REST Overview

- Why do you care about REST?
  - Most new Web APIs are built using REST
  - "Big players" all have REST interfaces
    - Microsoft, Amazon Google, Facebook, Twitter…
  - Simple, scalable & architecturally flexible
- Who created REST?
  - Dr. Roy Thomas Fielding
    - Best known for proposing REST architectural style
      - See Dissertation

REST Overview

- What is REST?
  - Representational State Transfer
    - An architectural style
      - Designed for distributed systems
  - Practical & pragmatic
  - Embraces constraints necessary for
    - Highly stable distributed systems
    - Separation of concerns - Client & Server
    - Stateless design

REST Overview

- Embraces constraints necessary for …
  - Improved network efficiency through caching
  - A uniform interface
    - Decoupling implementation from services
  - Layered Architecture
    - Composed in a constrained hierarchy
      - Layers are isolated
  - Optionally - Dynamic algorithmic extension
    - Code on demand
    - Extending systems by downloading code
      - Simplifying client feature implementation

REST Overview

- What was considered in REST's design?
  - Network
    - Reliability
    - Latency
    - Bandwidth
    - Security
  - Adaptation / Change!
  - Egress Cost
  - Heterogeneous Systems
    - Tech stack, device, & platform

REST Overview
REST presents a different model

- Basic differences REST vs method calls:
  - Actions Are defined by HTTP Verbs
    - DELETE, GET, PATCH, POST, PUT
    - See: Request Methods
  - Resources
    - Presented in the URI, this is the “Uniform Interface”
    - https://myserver.com/api/v1/notes

REST Overview

Basic Differences REST vs Method Calls Continued …

- Result data
  - Provided as an HTTP response
- Result / Status codes
  - Standardized - HTTP Status Codes
    - 200 OK, 400 Bad Request, 404 Resource Not Found …
    - See: Status Codes
    - Provided as an HTTP response

REST Overview

- The essentials
  - Resources are identified in requests as URIs
    - Makes up the Uniform Interface
      - Defines the interface clients will access
    - Resources presented as segments in URI
      - http://acme.com/customer
        - Identifies the customer resource
        - Allows for CRUD & List operations
      - http://acme.com/customer/1/order
        - Identifies the sub-resource order
          - For the customer with an ID of 1

REST Overview

- How are resources manipulated ?
  - HTTP Requests via VERBS
    - GET
      - Is nullipotent does not modify the state of the resource
      - Returns representation of the addressed resource
    - PUT
      - Is idempotent, multiple identical requests same effect
      - Replaces the addressed resource
      - Create the addressed resource if it does not exist
      - Client defines all attributes
        - Including unique ID

REST Overview
How are resources manipulated ?...

- HTTP Requests via VERBS ...
  - POST
    - Creates a new resource entry
      - Server assigns unique id
    - Not necessarily idempotent
      - May have further side effects
  - DELETE
    - Is idempotent, multiple identical requests same effect
    - Delete the addressed resource

REST Overview

- How are resources manipulated ?...
  - HTTP Requests ...
    - PATCH
      - Used to apply partial updates to a resource
      - Standard JSON patch payload, add, remove attributes
        - JsonPatch in ASP.NET Core web API | Microsoft Learn
        - RFC 6902: JavaScript Object Notation (JSON) Patch (rfc-editor.org)
    - HEAD
      - Is nullipotent has no side effects
      - Returns meta-information
      - Number of items in a collection for example

REST Overview
How are parameters provided ?...

- In the URI as
  - Segments
    - http://example.com/customer/1/order
      - 1 identifies a customer Id
        - Order identifies the orders for customer 1
  - As query string parameters
    - http://example.com/customer?addedAfter=2014-04-23&sortAscBy=date
      - Returns a list of customers
        - Added after 2014-04-23
        - Sorted ascending by date

REST Overview
How are parameters provided ?...

- In the URI as
- As header values
  - Accept: application/json
    - Content negotiation request a json response
    - MIME Types: Media Types
  - Authorization: Basic ZXuhVsRpqjateIPuTHBAe4WhJK==
    - Credentials for HTTP authentication
- In the request body typically
  - As Json or XML payload
  - For PUT, POST & PATCH actions

REST Overview

- How is information returned ?
  - Resource attributes in response body
    - Typically, as JSON or XML
  - Status and Standards data in response header
    - HTTP:/1.1 201 Created
    - Location: http://example.com/customer/1
      - Indicates the "URI" of the new resource
      - Via a PUT or POST action
        - In the POST case the server is providing the ID
        - In the PUT case the client had provided the ID
          - ID is being returned in the response

REST Overview

- What about machine readable description
  - Something like WSDL for REST?
- OpenAPI Specification See:
  https://swagger.io/specification/
  https://spec.openapis.org/oas/latest.html
  - Essentially a json document
    - Describes:
      - Resources & Supported HTTP Verbs
      - Input schema
      - Output schema per HTTP Response type
  - Used for UI and Proxy generation

REST Overview

- Key things to remember:
  - REST is a stateless protocol
  - Utilize nouns in URI design
    - Don't use verbs!
  - The action is defined by the HTTP Verb
  - Always utilize HTTP Status codes
    - To describe the results of a request

REST Essentials
Key things to remember

- Utilize uniform and simple interfaces
- Communication is done by representation
  - A self describing data media-type such as
    - JSON
    - XML
    - Etc…
  - Consisting of both
    - Data
    - Metadata describing the data in the payload

REST Overview

Further Reading

- RESTful Web Services Cookbook
  Solutions for improving Scalability and Simplicity
  - Author: Subbu Allamaraju
  - ISBN: 978-0596801687
  - Publisher: Yahoo Press (March 11, 2010)