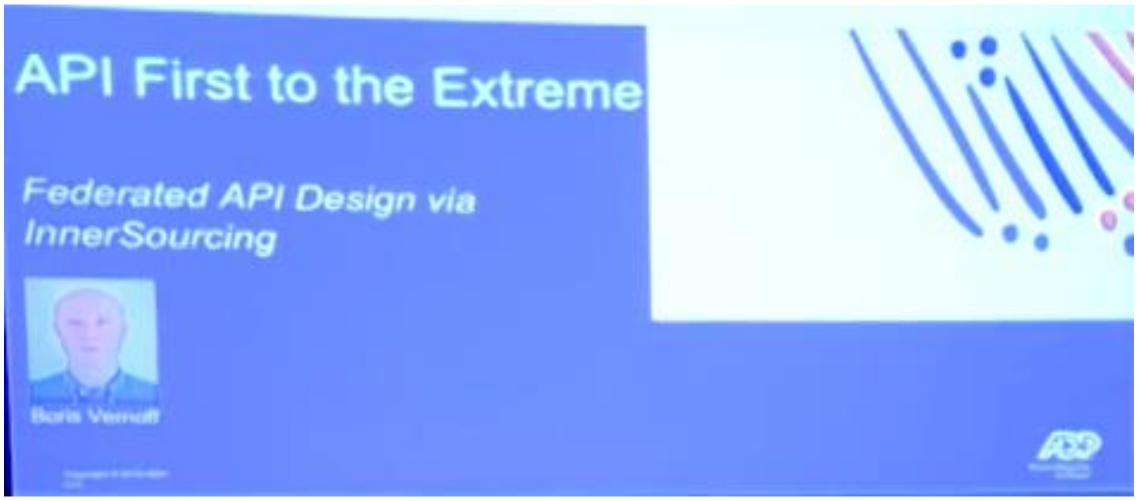
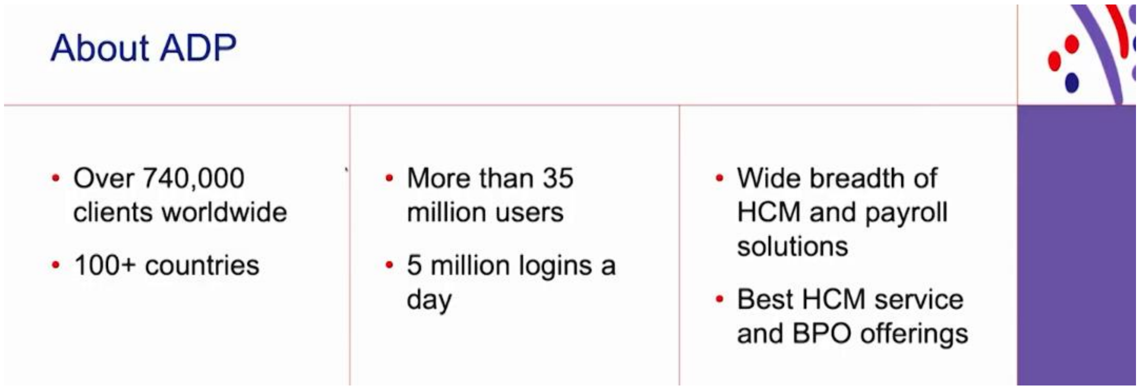


API First to the Extreme

Federated API Design via InnerSourcing

Boris Vernoff

An overview of the ADP API journey where the API First approach has become a fundamental principal of the API development within ADP. The session is focused on how to design the API specifications using an Open Source model to allow multiple product teams to design their APIs in a Federated mode to ensure the fastest time-to-market as well as a consistent look-and-feel across the Enterprise using the Open API Standard 3.0 (f.k.a. Swagger) and JSON schema. The session will also cover the API Governance aspects within an extremely diversified ADP business ecosystems via templates and API schema building ("Lego") blocks as well as the infrastructure to support an automated publication of the API specifications and the integration scenarios of the API Registry with the company-wide ESB, API authorization store and the Marketplace infrastructure.

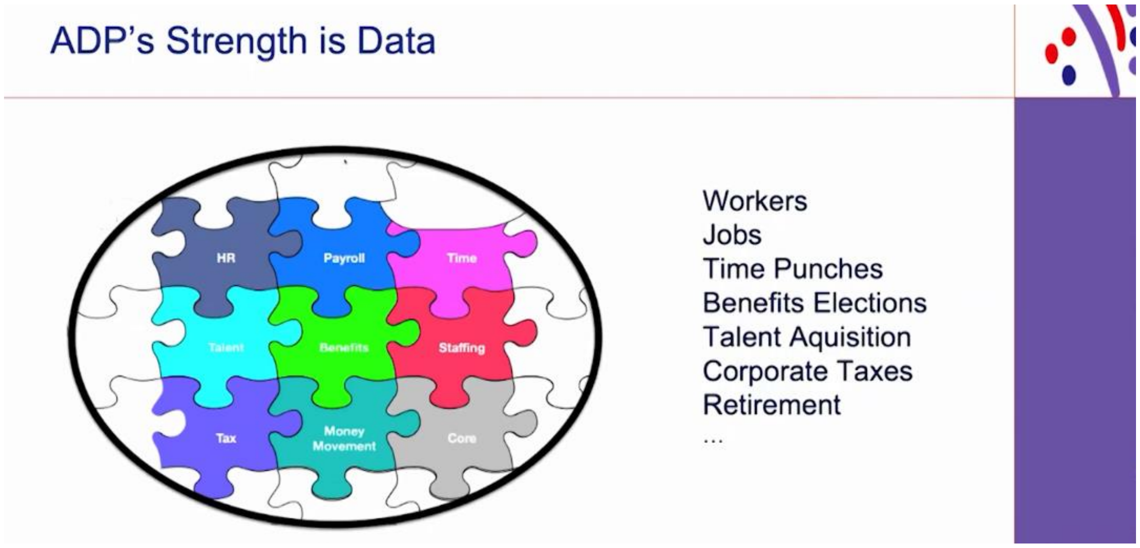


About ADP

- Over 740,000 clients worldwide
- 100+ countries

- More than 35 million users
- 5 million logins a day

- Wide breadth of HCM and payroll solutions
- Best HCM service and BPO offerings



ADP's Strength is Data

Workers
Jobs
Time Punches
Benefits Elections
Talent Aquisition
Corporate Taxes
Retirement
…

ADP as Technology Company

- Grew through acquisitions
- Variety of products - different platforms and technologies
- Difficult to integrate multiple products

API Everything

API Design Principals

- API First - specification comes before implementation
- Mapped to business use cases rather than UI
- Designed for backwards compatibility and reusability
- Self-descriptive and testable

API Specification is a Contract

Various Products – Various Formats

XML (XSD)

JSON Schema (Draft 2 through 6)

Swagger 2.0

OAS (Swagger) 3.0

JSON Samples

Word Documents

Code to Swagger

Code to JSON

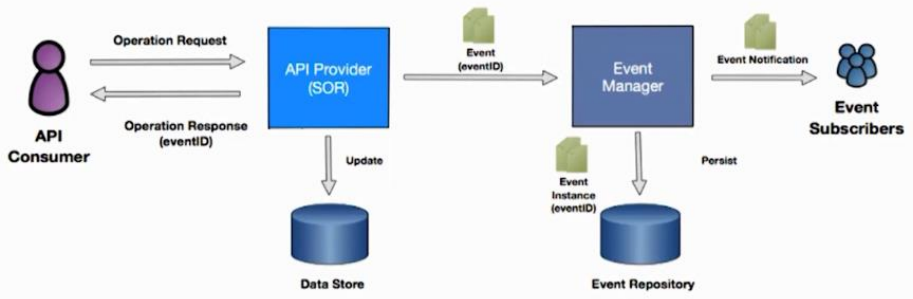
And more…

Various Formats – Disconnect

API Governance

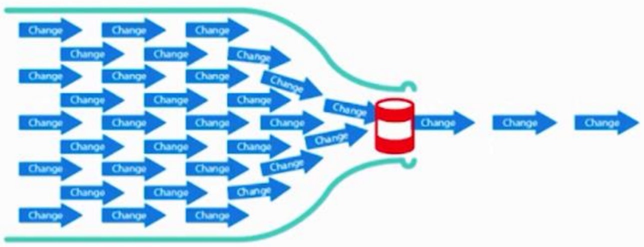
API Governance

- Canonical APIs: One capability – One API
- Single documentation platform: JSON Schema / OAS 3.0
- Event sourcing – audit, timeline, etc.

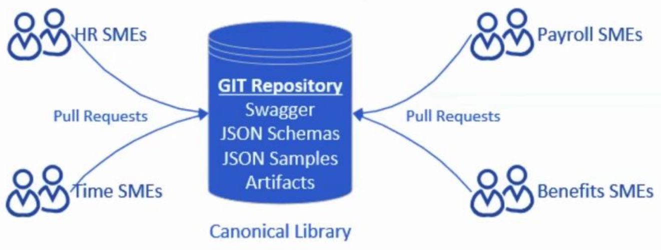


Centralized API Design Team – Doesn't Scale

- Collect requirements
- Evaluate design by various product teams
- Define JSON Schema / URIs
- Publish / maintain specifications



Scale the API Portfolio via InnerSourcing



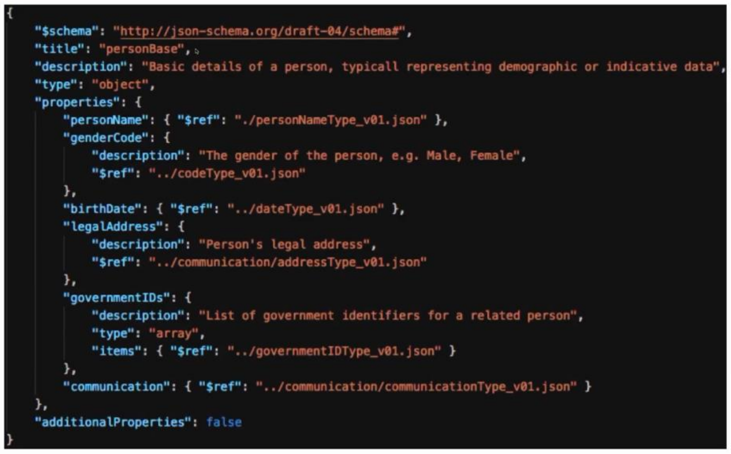
Federated Governance (or Guidance?) Team

- Maintain logical domain model
- Develop re-usable JSON schema component library
- Define specification templates
- Publish guidelines, procedures and acceptance criteria
- Educate and guide API product teams
- Manage GIT repositories / pull requests
- Publish API specifications

Reusable Component Library – “Lego” Blocks

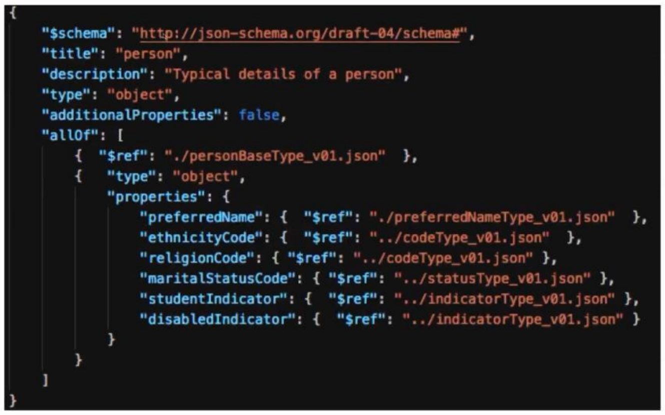
- Developed over last 5 years
- 500+ components - primitive and coarse / business
- “Living and Breathing” via InnerSourcing
- Indexed and searchable

Referencing Components



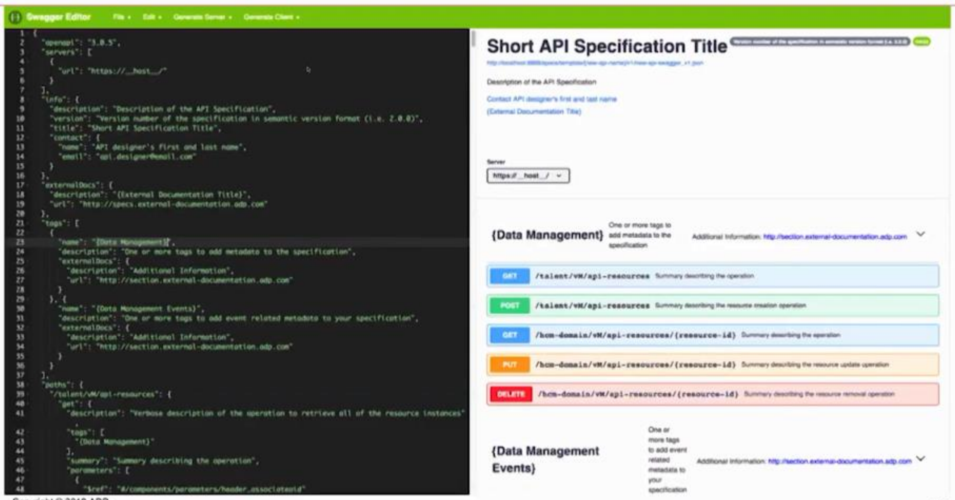
```
{
    "$schema": "http://json-schema.org/draft-04/schema#",
    "title": "personBase",
    "description": "Basic details of a person, typicall representing demographic or indicative data",
    "type": "object",
    "properties": {
        "personName": { "$ref": "./personNameType_v01.json" },
        "genderCode": {
            "description": "The gender of the person, e.g. Male, Female",
            "$ref": "../codeType_v01.json"
        },
        "birthDate": { "$ref": "../dateType_v01.json" },
        "legalAddress": {
            "description": "Person's legal address",
            "$ref": "../communication/addressType_v01.json"
        },
        "governmentIDs": {
            "description": "List of government identifiers for a related person",
            "type": "array",
            "items": { "$ref": "../governmentIDType_v01.json" }
        },
        "communication": { "$ref": "../communication/communicationType_v01.json" }
    },
    "additionalProperties": false
}
```

Referencing Components (cont'd)



```
{
    "$schema": "http://json-schema.org/draft-04/schema#",
    "title": "person",
    "description": "Typical details of a person",
    "type": "object",
    "additionalProperties": false,
    "allOf": [
        {  "$ref": "./personBaseType_v01.json"  },
        {   "type": "object",
            "properties": {
                "preferredName": {  "$ref": "./preferredNameType_v01.json"  },
                "ethnicityCode": {  "$ref": "../codeType_v01.json"  },
                "religionCode": { "$ref": "../codeType_v01.json" },
                "maritalStatusCode": { "$ref": "../statusType_v01.json" },
                "studentIndicator": {  "$ref": "../indicatorType_v01.json" },
                "disabledIndicator": {  "$ref": "../indicatorType_v01.json" }
            }
        }
    ]
}
```

Specification Template



```
{
  "openapi": "3.0.5",
  "servers": [
    {
      "url": "https://__host__/"
    }
  ],
  "info": {
    "description": "Description of the API Specification",
    "version": "Version number of the specification in semantic version format (i.e. 2.0.0)",
    "title": "Short API Specification Title",
    "contact": {
      "name": "API designer's first and last name",
      "email": "api.designer@email.com"
    }
  },
  "externalDocs": {
    "description": "{External Documentation Title}",
    "url": "http://specs.external-documentation.adp.com"
  },
  "tags": [
    {
      "name": "{Data Management}",
      "description": "One or more tags to add metadata to the specification",
      "externalDocs": {
        "description": "Additional Information",
        "url": "http://section.external-documentation.adp.com"
      }
    }, {
      "name": "{Data Management Events}",
      "description": "One or more tags to add event related metadata to your specification",
      "externalDocs": {
        "description": "Additional Information",
        "url": "http://section.external-documentation.adp.com"
      }
    }
  ],
  "paths": {
    "/talent/vM/api-resources": {
      "get": {
        "description": "Verbose description of the operation to retrieve all of the resource instances"
        ,
        "tags": [
          "{Data Management}"
        ],
        "summary": "Summary describing the operation",
        "parameters": [
          {
            "$ref": "#/components/parameters/header.associateoid"
```

Short API Specification Title

Description of the API Specification

Contact API designer's first and last name

{External Documentation Title}

Server

https://__host__/

{Data Management} One or more tags to add metadata to the specification — Additional Information: http://section.external-documentation.adp.com

GET /talent/vM/api-resources Summary describing the operation

POST /talent/vM/api-resources Summary describing the resource creation operation

GET /hcm-domain/vM/api-resources/{resource-id} Summary describing the operation

PUT /hcm-domain/vM/api-resources/{resource-id} Summary describing the resource update operation

DELETE /hcm-domain/vM/api-resources/{resource-id} Summary describing the resource removal operation

{Data Management Events} One or more tags to add event related metadata to your specification — Additional Information: http://section.external-documentation.adp.com

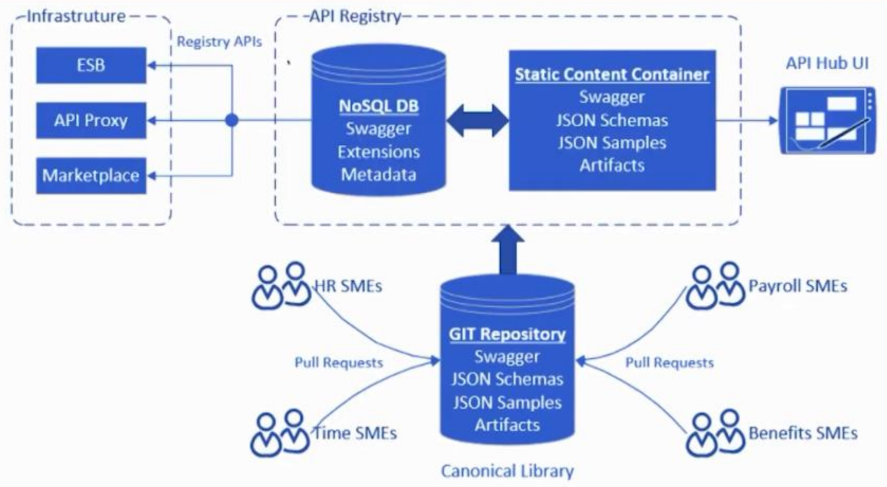
API Specifications

- Treated as code
- Sent through build and packaging process
- Deployed to multiple environments
- Used by the infrastructure components

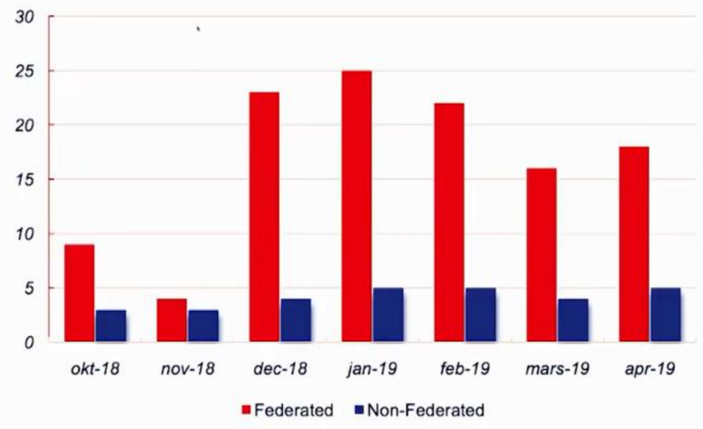
Code or Documentation?

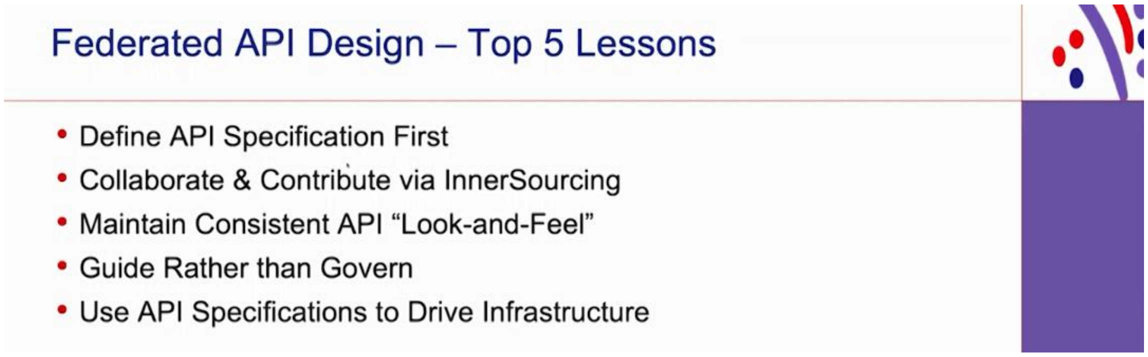
API Registry and Infrastructure

API Registry



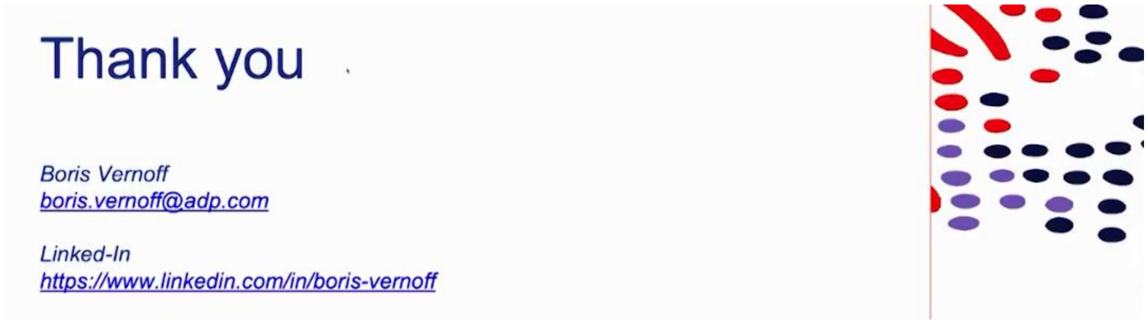
Federated vs. Non-Federated API Stats





Federated API Design – Top 5 Lessons

- Define API Specification First
- Collaborate & Contribute via InnerSourcing
- Maintain Consistent API “Look-and-Feel”
- Guide Rather than Govern
- Use API Specifications to Drive Infrastructure



Thank you

Boris Vernoff
boris.vernoff@adp.com

Linked-In
https://www.linkedin.com/in/boris-vernoff