

■ [gotomeet.me/consultingwerk](https://gotomeet.me/consultingwerk)

# ■ API first - Building ABL Clients based on Swagger

**Mike Fechner, Consultingwerk**

The full stack modernization framework  
**SmartComponent Library**  
by Consultingwerk



## Mike Fechner

- Director, Lead Modernization Architect and Product Manager, Architect of the SmartComponent Library and WinKit
- Specialized on object-oriented design, software architecture, desktop user interfaces and web technologies
- 34 years of Progress experience (V5 ... OE12)
- Active member of the OpenEdge community
- Frequent speaker at OpenEdge related conferences around the world



## Consultingwerk Software Services Ltd.

- Independent IT consulting organization
- Focusing on **OpenEdge** and **related technology**
- Located in Cologne, Germany, subsidiaries in UK, USA and Romania
- Customers in Europe, North America, Australia and South Africa
- Vendor of developer tools and consulting services
- Specialized in GUI for .NET, Angular, OO, Software Architecture, Application Integration
- Experts in OpenEdge Application Modernization





# Sample Swagger use-cases

- Consumption of 3<sup>rd</sup> party API's
  - “What is the URL of your Swagger file?”
- Integration projects
  - applications may provide Swagger file that needs to be implemented by other application
  - e.g. standard WMS and bespoke ERP system
  - document API requirements for bespoke ERP
  - test ERP API by ERP developer and WMS provider

## Sample Swagger use-cases

- Document REST or RESTful API's
- Bringing the WSDL of SOAP to REST (less strict)
- Simplify tests of API's by the developers, simpler to use than Postman, no need to build a client application
- Can be imported into Postman and similar tools
- Generation of API clients and server stubs

# REST/RESTful in the SmartComponent Library

- Standard protocol for application integration and UI flexibility
- SmartComponent Library provides the simplest and most flexible method of implementing REST or RESTful services with OpenEdge
- Typical use-cases
  - Implement new functionality as RESTful services
  - Provide existing (legacy) functionality as RESTful service
- Open API / Swagger documentation / test suite out of the box, generated automatically
- No need to deploy services, code declares the API
- Sophisticated authentication and authorization features

## REST/RESTful – new Features

- Full support for JSON schema / Open API 3.0 – supporting implementation of every interface
- API-first design – implement service based on Open API specification; typical requirement in integration projects
- Generation of ABL clients for existing REST services
- Full support for ABL legacy code remaining like SHARED/GLOBAL SHARED variables when using OOABL (e.g. database trigger or executed legacy procedures)



# Agenda

- **What is Swagger / OpenAPI**
- Understanding the Swagger File
- Foundation for building ABL Clients
- Introducing an ABL based templating engine
- Sample ABL Client
- Future



# What is OpenAPI

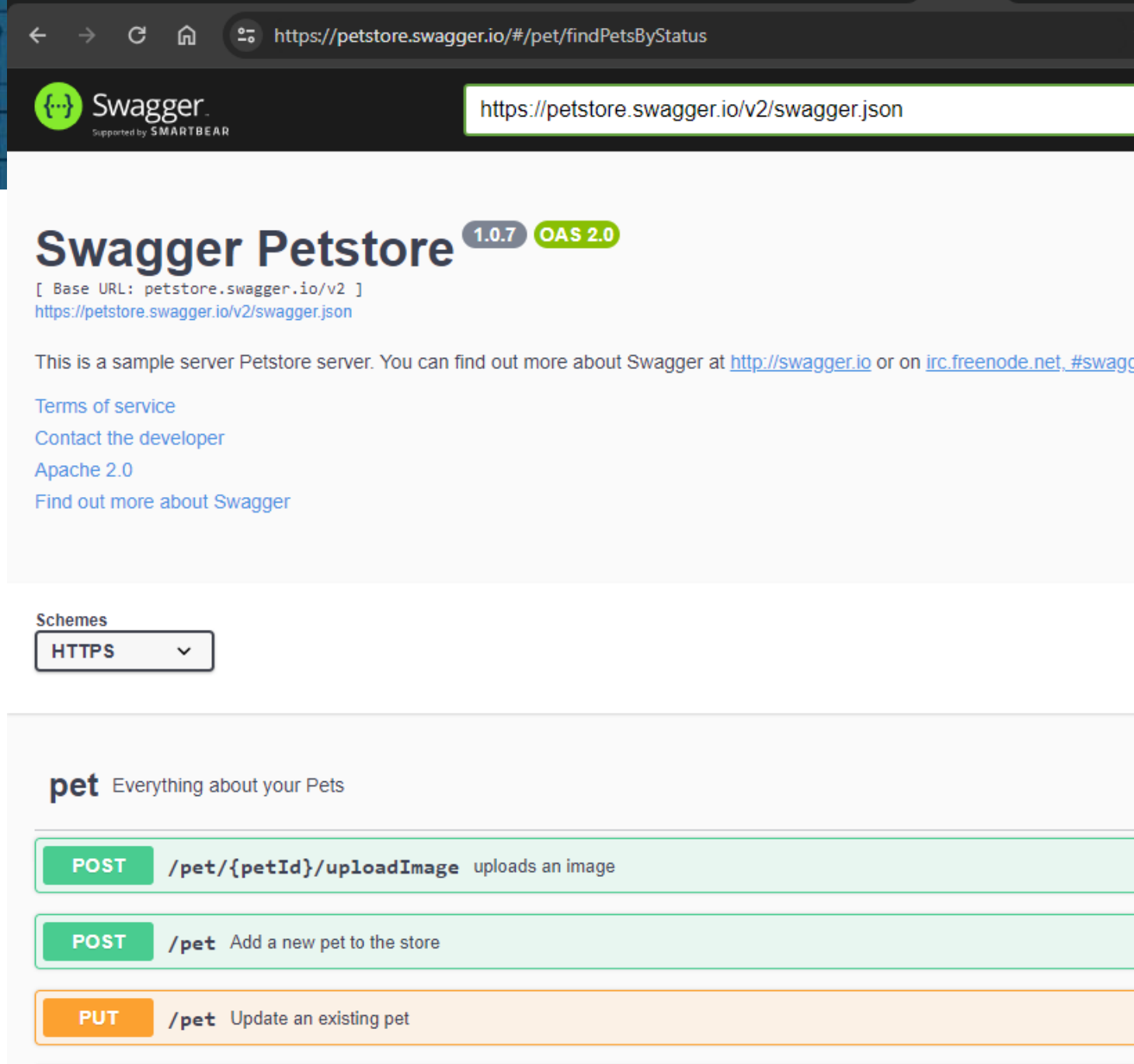
- Formerly known as Swagger Specification
- Standard for describing synchronous HTTP API's
- Related standard for async API's: AsyncAPI
- Swagger ca. 2011
- OpenAPI Initiative 2016, members include Atlassian, Google, IBM, PayPal, SAP, ...
- <https://www.openapis.org/>

# What is Swagger

- Today: Collection of tools supporting work with OpenAPI specs
- Swagger and OpenAPI often used as synonyms
- Swagger Editor (Online)
- Swagger Codegen
- Swagger UI (Online documentation and test)
- Swagger Hub
- Swagger Inspector

# Demo Swagger

- <https://petstore.swagger.io/>



The screenshot shows the Swagger Petstore API documentation page. The browser address bar displays `https://petstore.swagger.io/#/pet/findPetsByStatus`. The Swagger logo is visible, along with the version `1.0.7` and the OpenAPI specification version `OAS 2.0`. The base URL is `petstore.swagger.io/v2`, and the Swagger JSON file is located at `https://petstore.swagger.io/v2/swagger.json`. The page includes links for [Terms of service](#), [Contact the developer](#), [Apache 2.0](#), and [Find out more about Swagger](#). A dropdown menu for Schemes is set to `HTTPS`. The API is titled `pet` with the description "Everything about your Pets". Three endpoints are listed: `POST /pet/{petId}/uploadImage` (uploads an image), `POST /pet` (Add a new pet to the store), and `PUT /pet` (Update an existing pet).

Swagger  
Supported by SMARTBEAR

`https://petstore.swagger.io/v2/swagger.json`

## Swagger Petstore

1.0.7 OAS 2.0

[ Base URL: `petstore.swagger.io/v2` ]  
<https://petstore.swagger.io/v2/swagger.json>

This is a sample server Petstore server. You can find out more about Swagger at <http://swagger.io> or on [irc.freenode.net, #swagger](irc://freenode.net/#swagger)

[Terms of service](#)  
[Contact the developer](#)  
[Apache 2.0](#)  
[Find out more about Swagger](#)

Schemes

HTTPS

**pet** Everything about your Pets

**POST** `/pet/{petId}/uploadImage` uploads an image

**POST** `/pet` Add a new pet to the store

**PUT** `/pet` Update an existing pet



GET

/pet/findByStatus Finds Pets by status

🔒 ⬆

Multiple status values can be provided with comma separated strings

Parameters

Cancel

Name	Description
<b>status</b> * required array[string] (query)	Status values that need to be considered for filter <div><div>available</div><div>pending</div><div>sold</div></div>

Execute

Clear

Responses

Response content type application/json

Curl

```
curl -X 'GET' \
  'https://petstore.swagger.io/v2/pet/findByStatus?status=sold' \
  -H 'accept: application/json'
```

Request URL

```
https://petstore.swagger.io/v2/pet/findByStatus?status=sold
```

Server response

Code	Details
200	<div>Response body</div> <div><pre>[   {     "id": 234735,     "category": {       "id": 713633,       "name": "y0dpqh"     },     "name": "WZj0nj",     "photoUrls": [       "lGp0IG"     ],     "tags": [       {         "id": 502437,         "name": "ctevUN"       }     ],     "status": "sold"   }, ]</pre></div>

## Demo Swagger

- OpenEdge OEmanager (PASOE)
- List endpoints for managing PASOE instance
- <http://localhost:8820/oemanager/>
- oemanager.war needs to be deployed
- Swagger enabled by default in OpenEdge 12
- Needs to be enabled in 11.7

The screenshot shows the Swagger UI interface for the PASOE Management APIs. At the top, the URL `http://localhost:8820/oemanager/doc/api-docs?url=/oemanager/doc/openapi.json#/` is visible. The Swagger logo and version `2.23.4` are present. The main title is **PASOE Management APIs** with a version tag `OAS3`. Below the title, there is a description: "Monitor and manage a PAS for OpenEdge instance with RESTful API calls. Expand each component to list the available ABL components. For each reference, use the **Try it out** button to test. These APIs use JSON input/output payloads, so request body content must be valid JSON." There is also a link to "Read more about PASOE Administration" and another link to "OpenEdge Communities".

Below the description, there is a "Servers" section with a dropdown menu showing `/oemanager - PASOE Administration`.

The main content area lists several API endpoints:

- AgentManager Administration** Manages the Agent Manager's behavior and metrics for an ABL instance.
- GET** `/applications/{AppName}/agents/{AgentID}/{Component}` Get a single MS-Agent's statistics
- GET** `/applications/{AppName}/agents/{AgentID}/sessions` Get an MS-Agent's Session Metrics
- DELETE** `/applications/{AppName}/agents/{AgentID}/sessions` Refresh ABL Sessions

# Agenda

- What is Swagger / OpenAPI
- **Understanding the Swagger File**
- Foundation for building ABL Clients
- Introducing an ABL based templating engine
- Sample ABL Client
- Future



# Understanding the Swagger File

- <https://swagger.io/docs/specification/basic-structure/>
- JSON or YAML (Yet another markup language)
- OpenAPI documentation in YAML “but JSON works equally well”
- We prefer JSON due to ABL language support and coolness factor
- YAML and JSON can be converted back and forth



## Sections of the Swagger File

- Metadata (Version, Title, Description)
- Tags (grouping of endpoints, links to documentation)
- Servers (for test and real)
- Paths (the definition of endpoints, relative URL, method, parameters, request body, responses)
- Component Schema (reusable types for request and responses)
- Authentication

# Describing Data Models (schema)

- <https://swagger.io/docs/specification/data-models/>
- OpenAPI data types are based on an extended subset of JSON schema
  - Objects
  - Arrays of objects
  - Property types and formats
  - Enums
  - Property descriptions, sample values
  - Validation (required, min/maxLength, min/maxValue, ...)

# Sample Swagger File

- Review OEManager OpenAPI spec

## Online editor and validation

- „Current“ Swagger Editor <https://editor.swagger.io/>
- „Next“ SwaggerEditor <https://editor-next.swagger.io/>



Swagger Editor

Swagger Editor

+

https://editor.swagger.io

Swagger Editor  
Supported by SMARTBEAR

FileEditInsertGenerate ServerGenerate ClientAbout

1openapi: 3.0.3  
2info:  
3  title: Swagger Petstore  
4  description: |-  
5    This is a sample API that uses Swagger 2.0 to expose a pet store interface. You can find out more about Swagger at <https://swagger.io>. In the third iteration of the pet store, we've switched to the design first approach! You can now help improve the API by making changes to the definition itself or to the code. That way, with time, we can improve the API in general, and expose some of the new features in OAS 3.0.  
6    Swagger at <https://swagger.io>. In the third iteration of the pet store, we've switched to the design first approach! You can now help improve the API by making changes to the definition itself or to the code. That way, with time, we can improve the API in general, and expose some of the new features in OAS 3.0.  
7    You can now help improve the API by making changes to the definition itself or to the code. That way, with time, we can improve the API in general, and expose some of the new features in OAS 3.0.  
8    That way, with time, we can improve the API in general, and expose some of the new features in OAS 3.0.  
9  
10  If you're looking for the Swagger 2.0/OAS 2.0 version of Petstore, then click [here](#) (https://editor.swagger.io/v2/swagger.yaml). Alternatively, you can load via the 'Edit > Load Petstore OAS 2.0' menu option!  
11  
12  Some useful links:  
13  - [The Pet Store repository](https://github.com/swagger-api/swagger-petstore)  
14  - [The source API definition for the Pet Store](https://github.com/swagger-api/swagger-petstore/blob/master/src/main/resources/openapi.yaml)  
15  termsOfService: http://swagger.io/terms/  
16  contact:  
17    email: apiteam@swagger.io  
18  license:  
19    name: Apache 2.0  
20    url: http://www.apache.org/licenses/LICENSE-2.0.html  
21  version: 1.0.11  
22  externalDocs:  
23    description: Find out more about Swagger  
24    url: http://swagger.io  
25  servers:  
26    - url: https://petstore3.swagger.io/api/v3  
27  tags:  
28    - name: pet  
29      description: Everything about your Pets  
30      externalDocs:  
31        description: Find out more  
32        url: http://swagger.io  
33    - name: store  
34      description: Access to Petstore orders  
35      externalDocs:  
36        description: Find out more about our store  
37        url: http://swagger.io  
38    - name: user  
39      description: Operations about user  
40  paths:  
41    /pet:  
42      put:  
43        tags:  
44          - pet  
45        summary: Update an existing pet  
46        description: Update an existing pet by Id  
47        operationId: updatePet  
48        requestBody:  
49          description: Update an existent pet in the store  
50          content:  
51            application/json:  
52              schema:  
53                \$ref: '#/components/schemas/Pet'  
54            application/xml:  
55              schema:  
56                \$ref: '#/components/schemas/Pet'  
57            application/x-www-form-urlencoded:  
58              schema:  
59                \$ref: '#/components/schemas/Pet'

Import URL  
Import file  
Save as YAML  
Convert and save as JSON  
Clear editor

Swagger Petstore - OpenAPI 3.01.0.11OAS 3.0

This is a sample Pet Store Server based on the OpenAPI 3.0 specification. You can find out more about Swagger at <https://swagger.io>. In the third iteration of the pet store, we've switched to the design first approach! You can now help improve the API whether it's by making changes to the definition itself or to the code. That way, with time, we can improve the API in general, and expose some of the new features in OAS3.

If you're looking for the Swagger 2.0/OAS 2.0 version of Petstore, then click [here](#). Alternatively, you can load via the [Edit > Load Petstore OAS 2.0](#) menu option!

Some useful links:

- [The Pet Store repository](#)
- [The source API definition for the Pet Store](#)

[Terms of service](#)

[Contact the developer](#)

[Apache 2.0](#)

[Find out more about Swagger](#)

Servers

https://petstore3.swagger.io/api/v3

Authorize

petEverything about your PetsFind out more

PUT/petUpdate an existing pet

POST/petAdd a new pet to the store

GET/pet/findByStatusFinds Pets by status

GET/pet/findByTagsFinds Pets by tags

GET/pet/{petId}Find pet by ID

POST/pet/{petId}Updates a pet in the store with form data

DELETE/pet/{petId}Deletes a pet

© 2024 Consulting

The screenshot displays the Swagger Editor interface. On the left, a file explorer shows a project structure with folders like 'src' and 'test', and files like 'index.html' and 'swagger.json'. The main editor area shows a JSON snippet representing an API response. A context menu is open over the JSON, with the 'Import File' option selected. The menu also includes 'Load Example >', 'Save (as JSON)', 'Convert and Save as YAML', 'Download Resolved JSON', and 'Download Resolved YAML'. The JSON snippet is as follows:

```
{
  "responses": {
    "500": {
      "description": "Failed to retrieve APSV transport properties"
    },
    "200": {
      "description": "Successfully retrieved APSV transport properties",
      "content": {
        "application/json": {
          "schema": {
            "properties": {
              "operation": {
                "type": "string"
              },
              "outcome": {
                "type": "string"
              },
              "result": {
                "$ref": "#/components/schemas/ApsvTransportProperties"
              },
              "errmsg": {
                "type": "string"
              },
              "versionStr": {
                "type": "string"
              },
              "versionNo": {
                "type": "string"
              }
            }
          }
        }
      }
    }
  }
}
```

## API Design Systems

**/applications/{AppName}/agents/properties** Update AgentManager Properties

# VS Code Plugins

- There are plenty in the marketplace
- Examples
  - <https://marketplace.visualstudio.com/items?itemName=42Crunch.vscode-openapi>
  - <https://marketplace.visualstudio.com/items?itemName=Arjun.swagger-viewer>

The image displays a dual-pane interface for OpenAPI documentation. The left pane, titled 'openapi-with-auth.json', shows a JSON schema for a GET endpoint. The schema includes a 'paths' object with a route for '/applications/{AppName}/webapps/{WebAppName}/transport/apsv/properties'. It defines two required path parameters: 'AppName' (Application Name) and 'WebAppName' (OEABL WebApp Name). The 'responses' section defines a 200 response with a description 'Successfully retrieved APSV transport properties' and a JSON example. The right pane, titled 'Swagger Preview', provides a visual representation of the API endpoint. It shows the GET method, the endpoint URL, and the parameters. The response section shows a 200 status code with the description 'Successfully retrieved APSV transport properties' and a JSON example. The JSON example is a complex object with various properties like 'operation', 'outcome', 'result', 'adapterEnabled', 'allowRuntimeUpdates', 'collectMetrics', 'enableRequestChunking', 'oeepingEnabled', 'oeepingProcedure', 'serviceFaultLevel', 'statusEnabled', 'useHTTPSessions', 'errmsg', 'versionStr', and 'versionNo'. The background of the entire image features a blue grid pattern with the text 'Consultingwerk' in the top right corner.

The image displays a dual-pane interface for OpenAPI documentation. The left pane, titled 'openapi-with-auth.json', shows a JSON schema for a GET endpoint. The schema includes a 'paths' object with a route for '/applications/{AppName}/webapps/{WebAppName}/transport/apsv/properties'. It defines two required path parameters: 'AppName' (Application Name) and 'WebAppName' (OEABL WebApp Name). The 'responses' section defines a 200 response with a description 'Successfully retrieved APSV transport properties' and a JSON example. The right pane, titled 'Swagger Preview', provides a visual representation of the API endpoint. It shows the GET method, the endpoint URL, and the parameters. The response section shows a 200 status code with the description 'Successfully retrieved APSV transport properties' and a JSON example. The JSON example is a complex object with various properties like 'operation', 'outcome', 'result', 'adapterEnabled', 'allowRuntimeUpdates', 'collectMetrics', 'enableRequestChunking', 'oeepingEnabled', 'oeepingProcedure', 'serviceFaultLevel', 'statusEnabled', 'useHTTPSessions', 'errmsg', 'versionStr', and 'versionNo'. The background of the entire image features a blue grid pattern with the text 'Consultingwerk' in the top right corner.



# Agenda

- What is Swagger / OpenAPI
- Understanding the Swagger File
- **Foundation for building ABL Clients**
- Introducing an ABL based templating engine
- Sample ABL Client
- Future



<https://editor.swagger.io>

```
1 openapi: 3.0.3
2 info:
3   title: Swagger Petstore - OpenAPI 3.0
4   description: |-
5     This is a sample Pet Store Server based on the OpenAPI 3.0
6     Swagger at [https://swagger.io](https://swagger.io). In the
7     switched to the design first approach!
8     You can now help us improve the API whether it's by making
9     That way, with time, we can improve the API in general, and
10    _If you're looking for the Swagger 2.0/OAS 2.0 version of
11    .swagger.io/?url=https://petstore.swagger.io/v2/swagger
12    `Edit > Load Petstore OAS 2.0` menu option!_
13    Some useful links:
14    - [The Pet Store repository](https://github.com/swagger-ap
15    - [The source API definition for the Pet Store](https://gi
16    /src/main/resources/openapi.yaml)
17 termsOfService: http://swagger.io/terms/
18 contact:
19   email: apiteam@swagger.io
20 license:
21   name: Apache 2.0
22   url: http://www.apache.org/licenses/LICENSE-2.0.html
23 version: 1.0.11
24 externalDocs:
25   description: Find out more about Swagger
26   url: http://swagger.io
27 servers:
28   - url: https://petstore3.swagger.io/api/v3
```

csharp

csharp-dotnet2

dart

dynamic-html

go

html

html2

java

javascript

jaxrs-cxf-client

kotlin-client

openapi

openapi-yaml

php

python

r

ruby

scala

swift3

swift4

swift5

typescript-angular

typescript-axios

typescript-fetch

## Foundation for building ABL Clients

- No ABL Client (or Server) generator on swagger.io
- Foundation for writing clients is in the “box”
- OpenEdge HTTP Client – available since OpenEdge 11.5 (11.7 preferred)
- Progress.Json.ObjectModel.JsonObject & Co.
- Temp-Table & ProDataset READ-JSON / WRITE-JSON
  - Nice to have – not always practical as many JSON structures don't match the flat array that's used to expose a Temp-Table

# The ABL HTTP Client

- A class library that provides support for HTTP(S)
  - Designed for API use
  - HttpClient, URI, HttpHeaders, Cookie, HttpRequest, HttpResponse all in OpenEdge.Net packages
  - Supports much of HTTP 1.1 spec
  - Shipped in \$DLC/[src|tty|gui]/netlib/OpenEdge.Net.pl ... make sure to add to PROPATH
- Simple, extensible programming interface
- Platform-portable (built on ABL sockets)
- Limitations
  - No streaming
  - Synchronous
- API doc at <https://documentation.progress.com/output/oehttpclient/index.html> (11.6.0+)



# Fetching data aka GET requests

```

define variable oClient    as IHttpClient          no-undo.
define variable oRequest  as IHttpRequest         no-undo.
define variable oResponse as IHttpResponse        no-undo.
define variable oLib      as IHttpClientLibrary  no-undo.
define variable cSSLProtocols as character extent 2 no-undo
    initial ['TLSv1.2', 'TLSv1.1'].

oLib = ClientLibraryBuilder:Build()
    /* With incorrect protocols, request times out */
    :SetSSLProtocols(cSSLProtocols)
    :Library.
oClient = ClientBuilder:Build()
    :UsingLibrary(oLib)
    :Client.
oRequest = RequestBuilder:Get("https://bbc.com"):Request.
oResponse = oClient:Execute(oRequest).
if oResponse:StatusCode eq 200 then
    message "Completed" view-as alert-box information.
else
    message "Error" oResponse:StatusReason view-as alert-box.
    
```

1. Create an HTTP client
  - Optional client library allows SSL configuration
2. Create a request object
  - Must have a method and URL
  - Headers, request body, cookies optional
3. Run the request
4. Process the response
  - Status code
  - Entity / response body



# Updating data aka PUT requests

```
define temp-table eCustomer like Sports2000.Customer.
```

```
create eCustomer.
```

```
/* assign field values */
```

```
oJsonBody = new JsonObject().
```

```
oJsonBody:Read(buffer eCustomer:handle).
```

```
oReq = RequestBuilder:Put("http://example.com/web/Entities/Customers", oJsonBody)  
      :ContentType("application/vnd.company+json")  
      :AddHeader("Authorization", substitute("Bearer &1", cJwt))  
      :AcceptJson()  
      :Request.
```

```
/* Reuse a HTTP Client */
```

```
oResp = oHttpClient:Execute(oReq).
```

- RequestBuilder can specify cookies, HTTP basic authentication, various headers, authentication callbacks etc

## JSON in ABL

- WRITE-JSON introduced in 10.2B for temp-tables, datasets
  - Updated to add "JsonObject" and "JsonArray" as output destinations
- SERIALIZE-ROW() and WRITE-JSON() updates
  - Optionally exclude envelope (i.e. temp-table name)
- General purpose Progress.Json.\* classes introduced in 11.0
- Populated via
  - NEW JsonObject() / JsonArray()
  - ObjectModelParser:Parse() and ParseFile()
  - JsonConstruct :Read() and :Write() methods

# JsonObject

```
using Progress.Json.ObjectModel.* from propath.
define variable oJsonObject as JsonObject no-undo.
define variable oChildObject as JsonObject no-undo.
```

```
oJsonObject = new JsonObject().
oJsonObject:Add("itemDescription", "A parcel for you").
oJsonObject:Add("itemWeight", 13.4).
oJsonObject:Add("itemHeight", 42).
oJsonObject:Add("itemWidth", 22).
oJsonObject:Add("itemDepth", 17).
```

```
oChildObject = new JsonObject().
oJsonObject:Add("units", oChildObject).
```

```
oChildObject:Add("height", "cm").
oChildObject:Add("width", "cm").
oChildObject:Add("depth", "cm").
oChildObject:Add("weight", "kg").
```

```
oJsonObject:WriteFile("object.json", true).
```

*object.json:*

```
{
  "itemDescription": "A parcel for you",
  "itemWeight": 13.4,
  "itemHeight": 42,
  "itemWidth": 22,
  "itemDepth": 17,
  "units": {
    "height": "cm",
    "width": "cm",
    "depth": "cm",
    "weight": "kg"
  }
}
```

# JsonArray

```
using Progress.Json.ObjectModel.* from propath.

define variable oJSONArray as JsonArray no-undo.

oJSONArray = new JsonArray().
/* oJsonObject from previous example */
oJSONArray:Add(oJsonObject).
oJSONArray:Add(now).

message oJSONArray:Length.      /* 2 */

oJSONArray:WriteFile("array.json", true).

oJSONArray:Set(1, "replace the item object").
```

```
array.json:

[
  {
    "itemDescription": "A parcel for you",
    "itemWeight": 13.4,
    "itemHeight": 42,
    "itemWidth": 22,
    "itemDepth": 17,
    "units": {
      "height": "cm",
      "width": "cm",
      "depth": "cm",
      "weight": "kg"
    }
  },
  // array items have different datatypes
  "2024-06-11T14:01:20.619-04:00"
]
```

# Progress.Json.ObjectModel.ObjectModelParser

- Read JSON using `Parse()` and `ParseFile()` methods
  - Return `JsonConstruct`, which can be cast down to `JsonObject` or `JsonArray`

```
define variable oParser as ObjectModelParser no-undo.
define variable oConstruct as JsonConstruct no-undo.
define variable oObject as JsonObject no-undo.
define variable oArray as JsonArray no-undo.
```

```
oParser = new ObjectModelParser().
```

```
oConstruct = cast(oParser:Parse("~/~"p1~/":42}"),
                  JsonObject).
```

```
/* Probably safer */
```

```
oConstruct = oParser:Parse("~/~"p1~/":42}").
```

```
if type-of(oConstruct, JsonObject) then
  oObject = cast(oConstruct, JsonObject).
else
  oArray = cast(oConstruct, JsonArray).
```



# Consultingwerk.JsonSerializable

- Base class (INHERITS) for ABL classes supporting flexible JSON serialization and deserialization
- Will be used in samples in this presentation
- <https://github.com/consultingwerk/ListsAndEnumSamples/blob/master/Consultingwerk/JsonSerializable.cls>

# Agenda

- What is Swagger / OpenAPI
- Understanding the Swagger File
- Foundation for building ABL Clients
- **Introducing an ABL based templating engine**
- Sample ABL Client
- Future



# Components that should be generated

- Response and Request parameter classes (types, schema)
- Enums
- Interfaces for anyOf kind references
- Clients

# Sample ApsvTransportProperties

- Part of OEManager OpenAPI spec
- 9 string/Character properties
- “1”/”0” for logical values ☹

```
{
  "ApsvTransportProperties": {
    "type": "object",
    "properties": {
      "adapterEnabled": {
        "type": "string"
      },
      "allowRuntimeUpdates": {
        "type": "string"
      },
      "collectMetrics": {
        "type": "string"
      },
      "enableRequestChunking": {
        "type": "string"
      },
      "oepingEnabled": {
        "type": "string"
      },
      "oepingProcedure": {
        "type": "string"
      },
      "serviceFaultLevel": {
        "type": "string"
      },
      "statusEnabled": {
        "type": "string"
      },
      "useHTTPSessions": {
        "type": "string"
      }
    }
  }
}
```

# ABL Class

- Straightforward ABL class
- 9 Character Properties
- INHERITS JsonSerializer

```
CLASS Demo.Swagger.oemanager.Client.ApsvTransportPropertiesd  
... INHERITS JsonSerializerd  
... SERIALIZABLE:d  
  
... DEFINE PUBLIC PROPERTY adapterEnabled AS CHARACTER NO-UNDOd  
... GET.d  
... SET.d  
  
... DEFINE PUBLIC PROPERTY allowRuntimeUpdates AS CHARACTER NO-UNDOd  
... GET.d  
... SET.d  
  
... DEFINE PUBLIC PROPERTY collectMetrics AS CHARACTER NO-UNDOd  
... GET.d  
... SET.d  
  
... DEFINE PUBLIC PROPERTY enableRequestChunking AS CHARACTER NO-UNDOd  
... GET.d  
... SET.d  
  
... DEFINE PUBLIC PROPERTY oeepingEnabled AS CHARACTER NO-UNDOd  
... GET.d  
... SET.d  
  
... DEFINE PUBLIC PROPERTY oeepingProcedure AS CHARACTER NO-UNDOd  
... GET.d  
... SET.d  
  
... DEFINE PUBLIC PROPERTY serviceFaultLevel AS CHARACTER NO-UNDOd  
... GET.d  
... SET.d
```



# ApsvTransportResponse

- Missing in official spec from OpenEdge
- 5 string/Character properties
- 1 object reference of type ApsvTransportProperties

```
"200": {↓
  "description": "Successfully retrieved APSV transport properties",↓
  "content": {↓
    "application/json": {↓
      "schema": {↓
        "properties": {↓
          "operation": {↓
            "type": "string"↓
          },↓
          "outcome": {↓
            "type": "string"↓
          },↓
          "result": {↓
            "$ref": "#/components/schemas/ApsvTransportProperties"↓
          },↓
          "errmsg": {↓
            "type": "string"↓
          },↓
          "versionStr": {↓
            "type": "string"↓
          },↓
          "versionNo": {↓
            "type": "string"↓
          }
        }↓
      }↓
    }↓
  }↓
}
```

# ABL Class

- Straight forward ABL class
- 5 Character Properties
- 1 Property of type previously shown ABL class
- INHERITS JsonSerializerable

```
class Demo.Swagger.oemanager.Client.ApsvTransportResponse
... inherits JsonSerializerable
... serializable:

... define public property operation as character no-undo
... get.
... set.

... define public property outcome as character no-undo
... get.
... set.

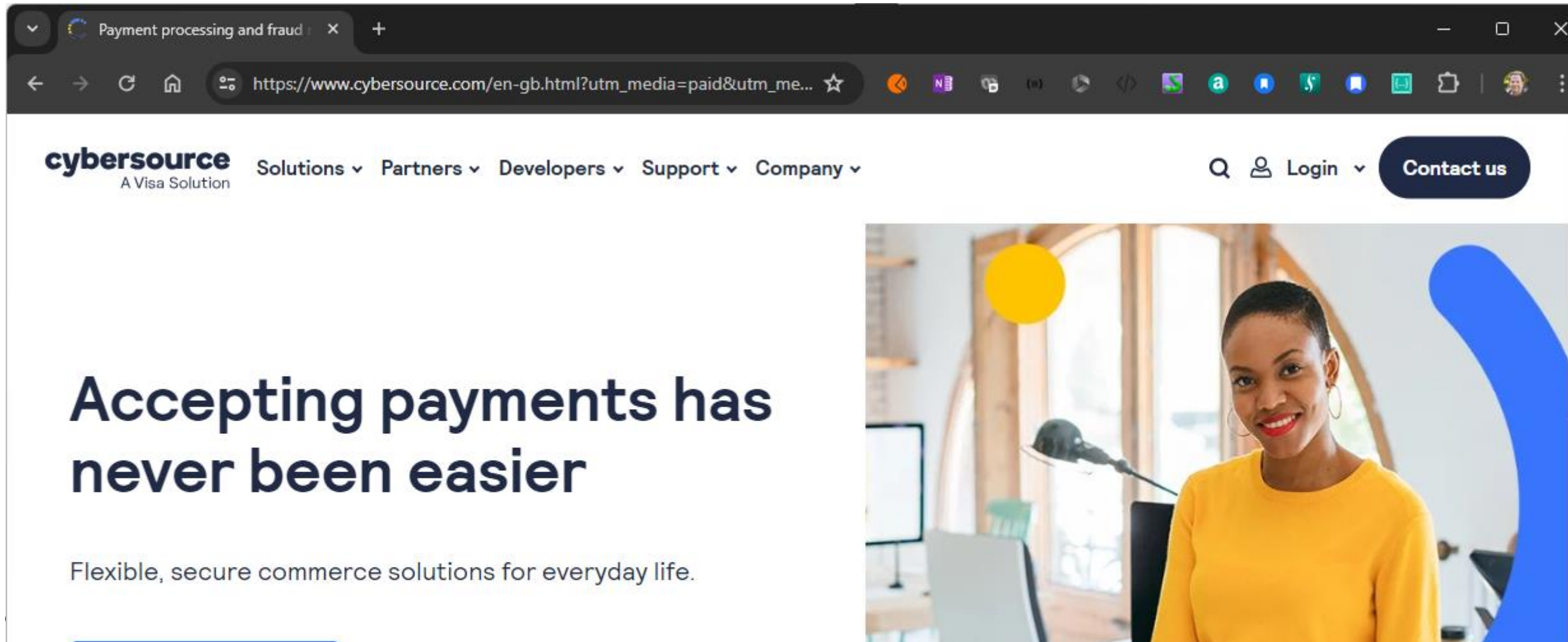
... define public property result as ApsvTransportProperties no-undo
... get.
... set.

... define public property errmsg as character no-undo
... get.
... set.

... define public property versionStr as character no-undo
... get.
... set.

... define public property versionNo as integer no-undo
... get.
... set.
```

# Cybersource – A Visa Solution



The image is a screenshot of a web browser displaying the Cybersource website. The browser's address bar shows the URL [https://www.cybersource.com/en-gb.html?utm\\_media=paid&utm\\_me...](https://www.cybersource.com/en-gb.html?utm_media=paid&utm_me...). The website's header features the Cybersource logo with the tagline "A Visa Solution" on the left. To the right of the logo is a navigation menu with links for "Solutions", "Partners", "Developers", "Support", and "Company", each followed by a dropdown arrow. Further right are icons for search, user profile, and "Login", followed by a dark blue button labeled "Contact us". The main content area on the left has a large heading "Accepting payments has never been easier" and a subtext "Flexible, secure commerce solutions for everyday life." On the right side of the page, there is a photograph of a smiling woman with short dark hair, wearing a bright yellow sweater, standing in a modern office environment with large windows and wooden beams. A large blue abstract shape is partially visible on the far right edge of the image.

Payment processing and fraud

[https://www.cybersource.com/en-gb.html?utm\\_media=paid&utm\\_me...](https://www.cybersource.com/en-gb.html?utm_media=paid&utm_me...)

**cybersource**  
A Visa Solution

Solutions ▾ Partners ▾ Developers ▾ Support ▾ Company ▾

Q Login ▾ **Contact us**

## Accepting payments has never been easier

Flexible, secure commerce solutions for everyday life.

# Cybersource API

- VISA Solution for accepting online payments
- Real world use case from a customer project (USA)
- Largest wholesale retailer for flowers in North-East USA
- API Spec provided as Swagger 2.0
  - 110.000 lines ...
  - Massive number of endpoints and types
  - Validation rules provided (mandatory fields, value ranges, Enums, ...)
- Conversion to OpenAPI 3.0 using online Swagger tooling failed
- Conversion using NodeJS api (from Github) succeeded



## ABL Classes ...

- 193 types generated for the createPayment Request/Response API alone

```
CLASS Dvflora.Payment.CyberSource.Api.Payments.createPaymentBody
... INHERITS ParameterObject
... IMPLEMENTS IValueObjectValidatorWithMessages
... SERIALIZABLE:

... DEFINE PUBLIC PROPERTY clientReferenceInformation AS createPaymentBodyClientReferenceInformation NO-UNDO
... GET.
... SET.

... DEFINE PUBLIC PROPERTY processingInformation AS createPaymentBodyProcessingInformation NO-UNDO
... GET.
... SET.

... DEFINE PUBLIC PROPERTY issuerInformation AS createPaymentBodyIssuerInformation NO-UNDO
... GET.
... SET.

... DEFINE PUBLIC PROPERTY paymentInformation AS createPaymentBodyPaymentInformation NO-UNDO
... GET.
... SET.
```



## Houston, we have a problem...

- How many interns does it take to code those 193 classes effectively?
  - Classes
  - Interfaces (anyOf, oneOf)
  - Enums
- $\text{time} = \text{total effort} / \# \text{ of interns} * \text{frustration level}$
- $193 * 10 \text{ minutes} = 32:10 \text{ hours} \dots$  close to a week and a mental crisis
- Testing? Code-Review? Updates when spec changes?

# We need a code generator!

## Sample code generator

- OpenEdge provides a code generator „out-of-the-box“
- Originally implemented as part of WebSpeed “Embedded SpeedScript” (E4GL)
- Similar to active templates, like
  - Eclipse JET Templates
  - ASP.NET
  - ...

```
<% DEFINE VARIABLE i ..... AS INTEGER ..... NO-UNDO. %>
<% DEFINE VARIABLE iCount ..... AS INTEGER ..... NO-UNDO. %>
<% DEFINE VARIABLE cProperties ..... AS CHARACTER EXTENT NO-UNDO. %>
<% DEFINE VARIABLE oProperties ..... AS JsonObject ..... NO-UNDO. %>
<% DEFINE VARIABLE oProperty ..... AS JsonObject ..... NO-UNDO. %>
<% DEFINE VARIABLE lcDescription AS LONGCHAR ..... NO-UNDO. %>
<% DEFINE VARIABLE lcDescription2 AS LONGCHAR EXTENT NO-UNDO. %>
<% FIX-CODEPAGE(lcDescription) = "UTF-8":U. %>
<% oProperties = SwaggerToAblHelper:GetProperties (poSchema) . . %>
<% cProperties = oProperties:GetNames() . . %>
<% iCount = EXTENT (cProperties) . . %>
<% EXTENT (lcDescription2) = iCount . . %>

/*-----
. . . File . . . : <%= poClassName:ClassName SKIP(0) %>
. . . Purpose . . . :
. . . Syntax . . . :
. . . Description :
. . . Author(s) . . :
. . . Created . . . : <%= STRING(NOW, "99.99.9999 HH:MM:SS") SKIP(0) %>
. . . Notes . . . :
. . . -----*/ <%= SKIP(1) %>

<%= '' SKIP %>
BLOCK-LEVEL ON ERROR UNDO, THROW.
<%= '' SKIP %>
{Consultingwerk/products.i}<%= SKIP(1) %>
<%= '' SKIP %>
USING Consultingwerk.* ..... FROM PROPATH .
USING Consultingwerk.Framework.Collections.* ..... FROM PROPATH .
USING <%= poClassName:PackageName %>.* <%= FILL (' ', MAX (1, 40 -- LENGTH (poClassName:PackageName, "CHARACTER"))) %> FROM PROPATH .
USING Progress.Lang.* ..... FROM PROPATH .
<%= '' SKIP %>
CLASS <%= poClassName:PackageName %> . <%= poClassName:ClassName SKIP (0) %>
. . . INHERITS ParameterObject
. . . IMPLEMENTS IValueObjectValidatorWithMessages
. . . SERIALIZABLE:
```

## Escape the source code

- Text in the Template will be pasted into generated file
- `<%= cValue %>` - Expression escape
- `<% DEFINE VARIABLE %>` - Statement escape
- Statement escapes support Loops

# Properties loop

```
<% DO i = 1 TO iCount: %>
<%   ASSIGN oProperty = oProperties:ToJsonObject(cProperties[i]) . . %>
<%   IF oProperty:Has ("description":u) THEN DO: %>
<%       ASSIGN lcDescription = oProperty:GetLongchar ("description":u) no-error. . %>
<%       IF ERROR-STATUS:ERROR THEN %>
<%       ASSIGN lcDescription = oProperty:ToJsonText ("description":u) . . %>

<%       ASSIGN lcDescription2[i] = substring(StringHelper:Indent(
replace(replace(replace (CodepageHelper:ConvertToCodePag
"\n":u, "~n":u),
"/*":u, "/ *":u),
"* /":u, "* /":u),
16),
17, -1, "character") . . %>

    /**
    * Purpose: <%= right-trim(STRING(lcDescription2[i])) SKIP(0) %>
    */
<%   END. %>

    DEFINE PUBLIC PROPERTY <%= cProperties[i] %> AS <%= AblDataType (cProperties[i], oProperties) %> NO-UNDO
    GET.
    SET.

<%= ' ' SKIP %>
<% END. %>
```



# Validation method

```

... METHOD PUBLIC CharacterDictionary IsValid (poValueObject AS Progress.Lang.Object):
<%= '' SKIP %>
...   DEFINE VARIABLE oMessages AS CharacterDictionary NO-UNDO.
...   DEFINE VARIABLE oObject AS <%= poClassName:ClassName %> NO-UNDO.
<%= '' SKIP %>
...   ASSIGN oObject = CAST (poValueObject, <%= poClassName:ClassName %>).
<%= '' SKIP %>
<% DO i = 1 TO iCount:
...   ASSIGN oProperty = oProperties:ToJsonObject(cProperties[i]).

...   IF oProperty:Has ("type") THEN DO:
...     CASE oProperty:GetCharacter ("type"):
...       WHEN "string" THEN DO:
...         IF oProperty:Has ("maxLength") THEN DO: %>
...         IF LENGTH (oObject:<%= cProperties[i] %>, "CHARACTER":U) > <%= oProperty:GetInteger ("maxLength") %> THEN
...           oMessages:Add ("<%= cProperties[i] %>":u, "Exceeds <%= oProperty:GetInteger ("maxLength") %> characters."&tran
<%   ..... END.
...   ..... END.
...   END CASE .
...   END.
... END.

```

## Generating code from the template

- JSON schema of current object passed as input parameter
- Recursive generation for nested types
- Resolving schema references (\$ref)
- Templates executed by customized version of e4glgen.p  
<https://github.com/consultingwerk/ADE-Sourcecode/blob/master/src/webutil/e4gl-gen.p>
- Template will be converted into ABL procedure using {&OUT}
- Standard include file pasted into template ABL source

```
block-level on error undo, throw.
```

```
using Consultingwerk.* ..... from propath.
using Consultingwerk.Studio.SwaggerToAbl.* from propath.
using Consultingwerk.Util.* ..... from propath.
using Consultingwerk.Util.LoggingStream.* from propath.
using Progress.Json.ObjectModel.* ..... from propath.
```

```
&global-define OUT put stream out unformatted
```

```
define input parameter poClassName ..... as ClassName ..... no-undo.
define input parameter poSchema ..... as JsonObject ..... no-undo.
define input parameter poApiSpec ..... as JsonObject ..... no-undo.
define input parameter poGeneratorParameter as SwaggerGeneratorParameter no-undo.
define input parameter poLoggingStream ..... as ILoggingStream ..... no-undo.
define output parameter pcOutputFile ..... as character ..... no-undo.
```

```
define stream out .
```

```
/* ***** Main Block ***** */
```

```
assign pcOutputFile = substitute ("%1~\&2.cls":U,
..... replace (poClassName:PackageName, ".":u, "/" :u),
..... poClassName:ClassName) .
```

```
if valid-object (poLoggingStream) then
..... poLoggingStream:WriteFormattedMessage("Writing to: &1 &2":U, poClassName:ClassName, pcOutputFile) .
```

```
output stream out to value (pcOutputFile) .
```

```
function AblDataType returns character
..... (pcPropertyName as character,
..... oProperties as JsonObject):
..... return SwaggerToAblHelper:ToAblDataType (poClassName:ClassName, pcPropertyName, oProperties, poApiSpec) .
end function.
```

# Agenda

- What is Swagger / OpenAPI
- Understanding the Swagger File
- Foundation for building ABL Clients
- Introducing an ABL based templating engine
- **Sample ABL Client**
- Future





## Review Sample ABL Client

- JSON based Get APSV Transport Properties
- JSON based Update APSV Transport Properties
- Strong Typed Get APSV Transport Properties
- Strong Typed Update APSV Transport Properties



# California sales tax REST service

- Website: <https://services.maps.cdtfa.ca.gov/>
- Swagger Document:  
<https://services.maps.cdtfa.ca.gov/swagger/LibraryOpenAPISpecification/swagger.json>
- REST service to provide sales tax rate based on address or geo coordinates

```
oClient = new OeManagerTransportsApsvClient-TYPED ().  
  
oResponse = oClient:getProperties_1 (new Credentials("oemanager":u, "tomcat":u, "tomcat":u),  
..... "smartpas_stream":u,  
..... "ROOT":u) .  
  
message oResponse:operation skip (2)  
..... "Adapter enabled?" oResponse:result:adapterEnabled view-as alert-box.  
  
assign oProperties ..... = oResponse:result  
..... oProperties:adapterEnabled = "0":u.  
  
oResponse = oClient:updateProperties_1 (new Credentials("oemanager":u, "tomcat":u, "tomcat":u),  
..... "smartpas_stream":u,  
..... "ROOT":u,  
..... oProperties) .  
  
message oResponse:operation skip (2)  
..... "Adapter enabled?" oResponse:result:adapterEnabled view-as alert-box.
```

# Fixing OEManager OpenAPI spec

- OpenEdge's OEManager's Swaggerfile deserves more TLC
  - Adding response schema
  - Defining basic authentication
  - Providing servers
  - Better proposed methods names: GetProperties\_1 ... GetProperties\_5
- Personally, not happy with "1" and "0" as string representation for logical values ...
  - Does not match an Enum as ABL Enum members need to start with a character value
  - So not really a cure for that issue

# Adding response schemas

- Provided by OpenEdge like this – no schema for HTTP responses
- Schema required for generation of strong typed API clients

```

"/applications/{AppName}/webapps/{WebAppName}/transports/apsv/properties" : {
  "get" : {
    Scan | Try it | Audit
    "tags" : [ "APSV Transport" ],
    "summary" : "Get APSV Transport properties by OEABL WebApp name",
    "description" : "Get APSV Transport properties by OEABL WebApp name",
    "operationId" : "getProperties_1",
    "parameters" : [ {
    }, {
    } ],
    "responses" : {
      "500" : {
        "description" : "Failed retrieving APSV transport properties"
      },
      "200" : {
        "description" : "Successfully retrieved APSV transport properties"
      }
    }
  }
}

```

```
"200": {↓
  "description": "Successfully retrieved APSV transport properties",↓
  "content": {↓
    "application/json": {↓
      "schema": {↓
        "properties": {↓
          "operation": {↓
            "type": "string"↓
          },↓
          "outcome": {↓
            "type": "string"↓
          },↓
          "result": {↓
            "$ref": "#/components/schemas/ApsvTransportProperties"↓
          },↓
          "errmsg": {↓
            "type": "string"↓
          },↓
          "versionStr": {↓
            "type": "string"↓
          },↓
          "versionNo": {↓
            "type": "string"↓
          }↓
        }↓
      }↓
    }↓
  }↓
}
```

Provided by OpenEdge as  
request body for set  
properties request



# Authentication

- By default, basic Authentication used („tomcat“, „tomcat“)
- May depend on PASOE configuration
- Required for client generation

```
    , ,  
    "securitySchemes": {  
      "BasicAuth": {  
        "type": "http",  
        "scheme": "basic"  
      }  
    },  
    "security": [  
      {  
        "BasicAuth": []  
      }  
    ]  
  }  
}
```

# Agenda

- What is Swagger / OpenAPI
- Understanding the Swagger File
- Foundation for building ABL Clients
- Introducing an ABL based templating engine
- Sample ABL Client
- **Future**



## Fixing OEManager OpenAPI file

- Discussion with Progress product management if we can release a fixed (improved) version of the OEManager OpenAPI spec on Github
  - Basic authentication scheme
  - Definition of response schema
  - Enumerations
  - Better names for reused methods (GetProperties\_5)
  - Further documentation
  - Community effort to keep it up to date with new OpenEdge releases ...
- Alternatively log issues in OpenEdge as issues with tech support

# Open sourcing parts of the implementation

- Plans to release on Github
  - the template engine (based on E4GL template engine)
  - the templates for the schema types
  - necessary helper code
  - sample for clients
- Planned to be available for workshops at PUG Challenge 2024

## Next events with Consultingwerk

- PUG Challenge Prague, September 18th-20th, 2024
- PUG Challenge Boston, September 29th-October 2nd, 2024
- Further webinars planned during the fall/winter



