OPENNTF WEBINARS

April 20, 2023 OpenNTF Webinar:
Getting to Know Domino REST API
AGENDA

• Welcome
• Presentation
  Paul Withers
• Q and A
THANKS TO THE OPENNTF SPONSORS

• HCL made a contribution to help our organization
  • Funds these webinars!
  • Contests like Hackathons
  • Running the organization

• Prominic donates all IT related services
  • Cloud Hosting for OpenNTF
  • Infrastructure management for HCL Domino and Atlassian Servers
  • System Administration for day-to-day operation
THIS IS OUR COMMUNITY

• Join us and get involved!
• We are all volunteers
• No effort is too small
• If your idea is bigger than you can do on your own, we can connect you to a team to work on it
• Test or help or modify an existing project
• Write guides or documentation
• Add reviews on projects / stars on Snippets
UPCOMING EVENTS

• Engage User Group Meeting
  The Future is NOW
  April 24-26
  Felix Meritis
  Amsterdam, Netherlands

• Collabsphere - MWLUG
  • In Person!
  • August 30 and 31
  • Chicago Botanical Garden
DEVELOPER REPAIR CAFE

• Bring your topic!
• Discuss ideas on how to tackle the issue
• Live at Engage on April 26 at 5:00 PM CET
NEXT WEBINAR

May 18 Webinar:
Java Collections, Maps and Streams
Paul Withers
ASKING QUESTIONS

• First Question – Will this be recorded?
  • Yes, view on YouTube!!!
  • https://www.youtube.com/user/OpenNTF

• Use the Questions Pane in GoToWebinar

• We will get to your questions at the end of the webinar

• The speakers will respond to your questions verbally
  • (not in the Questions pane)

• Please keep all questions related to the topics that our speakers are discussing!!!

• Unrelated Question => post at:
  • https://openntf.org/discord
Getting to Know Domino REST API

Paul Withers
Associate Director - Research
HCL Digital Solutions Software
Overview and Design Goals

Domino REST API and Standards

Documentation

AdminUI

Schemas, Scopes and Applications

Applications

Identity Providers and Security

Demos
Overview and Design Goals
The challenge

Domino is a world-class **NoSQL** database (The first commercial > 30 years ago). It tightly-couples UI and data with business logic in @Formula language, LotusScript & Java. Domino provides excellent security with per-database access control and its unique declarative access to read and write protection.

AND

Applications written in any system, tooling and platform need secure, standard compliant access to Domino data: Angular, Spring, ReactJS, NodeJS, Volt MX, Command line tools, Middleware & [insert-your-favorite]
Design Goals

**Access**
- Platform agnostic
- Any language
- Maintain security

**Data**
- Schema
- OData
- Translation from/to Native

**Documentation**
- API first (Swagger)
- User contribution
HCL Software

Domino REST API and Standards
## Domino and Open Standards

<table>
<thead>
<tr>
<th>Domino</th>
<th>Open Standard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Documents</td>
<td>JSON</td>
</tr>
<tr>
<td>NRPC</td>
<td>HTTPS</td>
</tr>
<tr>
<td>No Schema</td>
<td>JSON schema OData</td>
</tr>
<tr>
<td>NotesID</td>
<td>Access Token</td>
</tr>
</tbody>
</table>

* Many more
Domino vs. JSON & JSON schema

- Text
  - Names
  - Readers
  - Authors
- Number
- RichText
- DateTime

*Multivalue

- String
- Number
- Integer
- Boolean
- Object
- Array

- float
- double
- int32
- int64
- date
- date-time
- password
- byte
- binary

... and others
<table>
<thead>
<tr>
<th>Method</th>
<th>Endpoint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/attachments/{uid}</td>
<td>Creates a new attachment in an existing document</td>
</tr>
<tr>
<td>DELETE</td>
<td>/attachments/{uid}/(&lt;attachmentName&gt;)</td>
<td>Removes an attachment</td>
</tr>
<tr>
<td>GET</td>
<td>/attachments/{uid}/(&lt;attachmentName&gt;)</td>
<td>Retrieve a document's attachment</td>
</tr>
<tr>
<td>POST</td>
<td>/bulk/create</td>
<td>Create multiple documents in a single call</td>
</tr>
<tr>
<td>POST</td>
<td>/bulk/delete</td>
<td>Delete multiple documents in a single call</td>
</tr>
<tr>
<td>POST</td>
<td>/bulk/uuid</td>
<td>Retrieve multiple documents in a single call by uuid</td>
</tr>
<tr>
<td>PATCH</td>
<td>/bulk/update</td>
<td>Update multiple documents in a single call</td>
</tr>
<tr>
<td>POST</td>
<td>/document</td>
<td>Create a new document for a specified form.</td>
</tr>
<tr>
<td>DELETE</td>
<td>/document/{uid}</td>
<td>Delete the given document</td>
</tr>
<tr>
<td>GET</td>
<td>/document/{uid}</td>
<td>Retrieve a given document at its appropriate mode</td>
</tr>
<tr>
<td>PATCH</td>
<td>/document/{uid}</td>
<td>Perform an update on the document at the relevant mode</td>
</tr>
<tr>
<td>PUT</td>
<td>/documentmode/{uid}</td>
<td>Returns the modes available and fields that can be edited</td>
</tr>
<tr>
<td>GET</td>
<td>/lists</td>
<td>Provide information on available viewfolders</td>
</tr>
<tr>
<td>GET</td>
<td>/lists/{name}</td>
<td>Pulls in view data</td>
</tr>
<tr>
<td>GET</td>
<td>/lists/pivots/{name}</td>
<td>Processes view data as pivot</td>
</tr>
</tbody>
</table>
**apicur.io & OData**

**OData**

OData (Open Data Protocol) is an OASIS standard that defines a set of best practices for building and consuming RESTful APIs. OData helps you focus on your business logic while building RESTful APIs without having to worry about the various approaches to define request and response headers, status codes, HTTP methods, URL constructions, media types, payload formats, query options, etc. OData also provides guidance for tracking changes, defining functions/actions for reusable procedures, and sending asynchronous/batch requests.

OData RESTful APIs are easy to consume. The OData metadata, a machine-readable description of the data model of the APIs, enables the creation of powerful generic client proxies and tools.
HCLSoftware

Documentation
Welcome to HCL Domino REST API documentation

HCL Domino REST API provides a secure REST API with access to HCL Domino servers and databases while running on HCL Domino and HCL Notes on Windows, Linux, and Mac. Designed to re-establish Domino as a world class, modern, standards-compliant, cloud native and enterprise-level collaboration platform, it adds contemporary REST APIs to Notes and Domino, enabling a modern programming experience with the tools of your choice. For more information, see Introducing Domino REST API.
HCL Software

AdminUI
HCL Domino REST API Administrator

Database Management - REST API
CREATE/UPDATE SCHEMA

Database Management - Activation
CREATE/MANAGE SCOPES

Application Management - OAUTH
ADMIN
The REAL admin UIs

```

```

```server [https://localhost:8880]:
Username [John Doe]:
Password: accessing https://localhost:8880 as John Doe

```

```

```

```
Schemas, Scopes and Applications
What is What?

<table>
<thead>
<tr>
<th>Area</th>
<th>Who</th>
<th>What</th>
<th>Where</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schema</td>
<td>Developer</td>
<td>Configures what can be done</td>
<td>NSF Replicated</td>
</tr>
<tr>
<td>Scope</td>
<td>Administrator</td>
<td>Enables schema for this server</td>
<td>Server</td>
</tr>
<tr>
<td>Application</td>
<td>Developer / Administrator</td>
<td>Allows OAuth access to scope</td>
<td>Server</td>
</tr>
</tbody>
</table>
OAuth applications
SPA and others

```
stw@ ~ $ cat manifest.json
{
  "short_name": "Approval SPA",
  "name": "Where requests are treated with respect",
  "start_url": ".",
  "display": "standalone",
  "theme_color": "#000000",
  "background_color": "#ffffff",
  "icon": "approval.jpg"
}
```
Identity Provider and Security
Domino REST API contains an OAuth compliant IdP

- Use JWT for access tokens
- Use Domino as IdP
- Bring your own IdP

```json
{
    "response_types_supported": [
        "code",
        "token"
    ],
    "token_endpoint_auth_methods_supported": [
        "client_secret_post",
        "client_secret_basic",
        "none"
    ],
    "grant_types_supported": [
        "authorization_code",
        "refresh_token"
    ],
    "issuer": "https://acra.projectkeep.io",
    "authorization_endpoint": "https://acra.projectkeep.io/oauth/authorization",
    "token_endpoint": "https://acra.projectkeep.io/oauth/token",
    "scopes_supported": [
        "MAIL",
        "$DATA",
        "$DECRYPT",
        "Domino.user.all",
        "openid",
        "email",
        "keepconfig"
    ],
    "jwks_uri": "https://acra.projectkeep.io/.well-known/jwks.json",
    "userinfo_endpoint": "https://acra.projectkeep.io/api/v1/userinfo"
}
```
Permission dialogue

The **Project Castle** application is requesting permission to access your account doctor.notes@projectkeep.io

This will allow the **Project Castle** application to access the following HCL Domino applications and resources:

**SDATA**

REST

Any Domino database you have access to

[Buttons: Deny, Allow]
Specific Access

- **Domino REST API Admins (Scopes and OAuth)**
  - LocalKeepAdmins group
  - Editor Access to KeepConfig.nsf

- **Domino Developers (Schemas)**
  - Designer access to relevant NSF
  - Reader access to KeepConfig.nsf

- **Functional Accounts**
  - Management Console (Port 8889), Metrics (Port 8890), Health Check (Port 8886)
  - Stored in keepconfig.d/functionalUsers.json
  - secret = salted and encrypted password retrieved from Management Console
Actually….

Try it out - now
QUESTIONS?

Use the GoToWebinar Questions Pane.

Please keep all questions related to the topics that our speakers are discussing!!!

Unrelated Question => post at: https://openntf.org/discord