

# OPENNTF WEBINAR

April 17, 2025

**Domino for Newbies: Development**

Jesse Gallagher, Graham Acres



# AGENDA

- Welcome
- Presentation
  - Jesse Gallagher, Graham Acres
- Q and A



# OPENNTF SPONSORS

- HCL made a contribution to help our organization
  - Funds these webinars!
  - Events, Community interaction on Discord
  - Running the organization
- Prominic donates all IT related services
  - Cloud Hosting for OpenNTF
  - Infrastructure management for HCL Domino Servers



# COMMUNITY – WE ARE ALL OPENNTF

- Participate – we all learn from each other
- 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

- “April”
  - Open Mic with the HCL Developers: Verse
    - May 1
- May
  - OpenNTF Webinar: Domino for Newbies: Administration
    - May 15
  - Engage User Group, The Hague, Netherlands
    - May 19 – 21
  - Open Mic with the HCL Developers
    - May 29

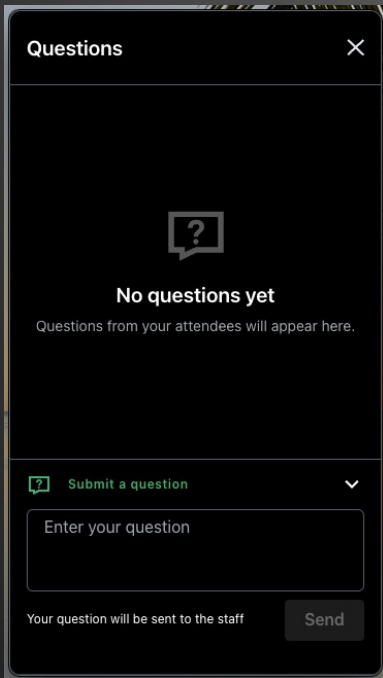


JOIN US ON DISCORD

<https://openntf.org/discord>



# 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 topics that our speakers are discussing
- Unrelated Question => post at:
  - <https://openntf.org/discord>



**Share  
your  
Dev  
Tips!**

# PRESENTATION

**Domino for Newbies: Development**





# YOUR PRESENTERS

## Jesse Gallagher

- Domino dev since around V7
- HCL Lifetime Ambassador and former IBM Champion
- OpenNTF Board and IP Manager 2014-2025
- Philadelphia, PA, USA

## Graham Acres

- Notes / Domino app architect since 1992
- HCL Ambassador, former IBM Champion
- Canadian User Group (C3UG)
- OpenNTF Board 2018-2025
- Vancouver, Canada



# TOPICS

- Who is this presentation for?
- What is Notes and Domino?
- Domino Designer
- Building a basic app
- Agents
- Debugging and Tracing
- Web apps in Domino
- Resources



# THIS PRESENTATION IS FOR...

- People who are new to the Domino world
  - You have a new job as a Domino developer
  - You inherit the dusty old server in the corner ... and something just broke
- Some level of development experience
  - Excel Macros
  - Full Stack devs
  - Java devs



# A BRIEF HISTORY OF TIME

Version	Release Date	Birthday	In Support
1	December 1989	35	No
5	March 1999	25	No
6	September 2002	22	No
8.5	December 2008	16	No
9	March 2013	12	No
10	October 2018	6	No
12	June 2021	3	Yes
14	December 2023	1	Yes
14.5	Later this year (2025)		



# WHAT IS NOTES AND DOMINO?

- Notes is the client app, but you also use a browser or mobile device
- Domino is the server
- NoSQL database
- Think like it's 1996
- Security: ID, Universal ID, Replica ID, User ID, Certifier ID
  - User certificate expiration
  - Server certificate expiration – see next month's webinar
- Replication



# THE SERVER: DOMINO

- All in one box\*
- The same server software handles data storage, client access, HTTP, and more
- This is quite unusual!
  - But it allows for neat tricks, like mail-in databases
- There's a strong community culture of "everything in Domino", since most things can fit

\* Mostly



# NOSQL DATABASE

- Historically, the DB structure was tricky to explain compared to SQL
- Now it's easier: it's a NoSQL document database
  - CouchDB, MongoDB, and others are like this
- It has those peculiar characteristics:
  - Flexible "schema": easy!
  - Attachments: easy!
  - Complex data: kind of easy!
  - Performant reporting: very hard!
  - Enforced data integrity: don't try!
- Domino replication is exceptional
  - It is straightforward to have a regularly-synced replica of a database on many servers or even locally
  - For example: servers on submarines replicating when surfacing



# DATA STRUCTURE

- Everything\* is a “note”
- Data documents are notes, but so are the forms, views, and other elements you work with
  - “Data” and “design” notes are the two types to care about
- All data documents are in the same pool, differentiated only by form name
- Every note has two IDs:
  - A “note ID” that is only unique to the specific file on the server
  - A “universal ID” (UNID) that is the same across replicas (use this one)
- There’s only one type of relation: response documents
  - A data document can contain the UNID of its “parent”, but there is no automatic syncing or enforcement between the two
  - Great for discussion databases – take care in other uses
  -

\* All the stuff you need to care about, anyway





# APPLICATION STRUCTURE

- Application structure (called “Design”) is stored in the NSF as well
- This is very unusual!
- It can be very convenient, but creates hurdles:
  - Source control is difficult and still culturally uncommon
  - It’s very straightforward to work direct in production, so people do so
  - It’s difficult to have “staging” and “production” versions
  - DBs based on templates often diverge subtly



# SECURITY AND IDS

- Much of Domino security is based on public-key encryption
- A “Notes ID” is a file that contains cryptographic keys that identify you as a user
  - IDs are issued by a “certifier ID”, which creates a validity chain
  - It’s kind of like SSH keypairs and the “trusted root” concept from TLS certificates
  - It’s been in there since the beginning
- Servers also have IDs
- IDs can be used to sign (to verify integrity) and encrypt (to create secrecy) data



# THINK LIKE IT'S 1996

- Notes dates to the late 80s with its opening heyday in the mid 90s
- It's very clever and it's evolved, but a lot of that era shines through:
  - 16/32k size limits abound
  - Form development is like Word with VB
  - Its data storage doesn't always quite map to JSON
  - Its core querying and indexing have severe limitations
  - Its development history has left layers of new, different, or abandoned ideas
  - There's a lot of "you can't do X for historical reasons" explanations



# APPLICATION DESIGN ELEMENTS

Element	Purpose
Form	Define UI for documents
Field (and Shared Field)	Store data (Readers, Authors, Rich Text)
Document	Actual user data (records)
View	Display / filter / sort documents
Folder	Manually group documents
Agent	Automate logic
Script	Add interactivity and logic
Page	Static content
Frameset	UI layout
Subform	Reusable form pieces
Outline	Navigation UI (menus)
Resources	Shared assets (images, files, code)



# ACCESS CONTROL LIST & ROLES

Level	
No Access	No Access (*Read or Write public documents)
Depositor	Create documents, otherwise no access
Reader	Read documents
Author	Create documents, Edit your own docs (*)
Editor	Create documents, Edit all documents (*)
Designer	Modify design elements, plus all above
Manager	Modify ACL, Encrypt, Change replication settings, delete database, plus all above

- Roles enhance what a user may be able to see in a database



# TEMPLATES

- Store design in a separate file and apply that design to numerous other databases.
- Great way to do version control as a designer
- Inheritance
  - Pro tip: design changes in an NSF that uses a template will be lost!



# THERE'S NO FIXING UGLY

- Developers are very rarely skilled in UI Design
- Great presentation by Scott Good
- Domino Restyle
  - [https://help.hcl-software.com/notes/14.0.0/client/restyle\\_applications.html](https://help.hcl-software.com/notes/14.0.0/client/restyle_applications.html)



# DOMINO DESIGNER

- Where to find things for development
- Forms, Views, Agents
- Code
- Design Synopsis





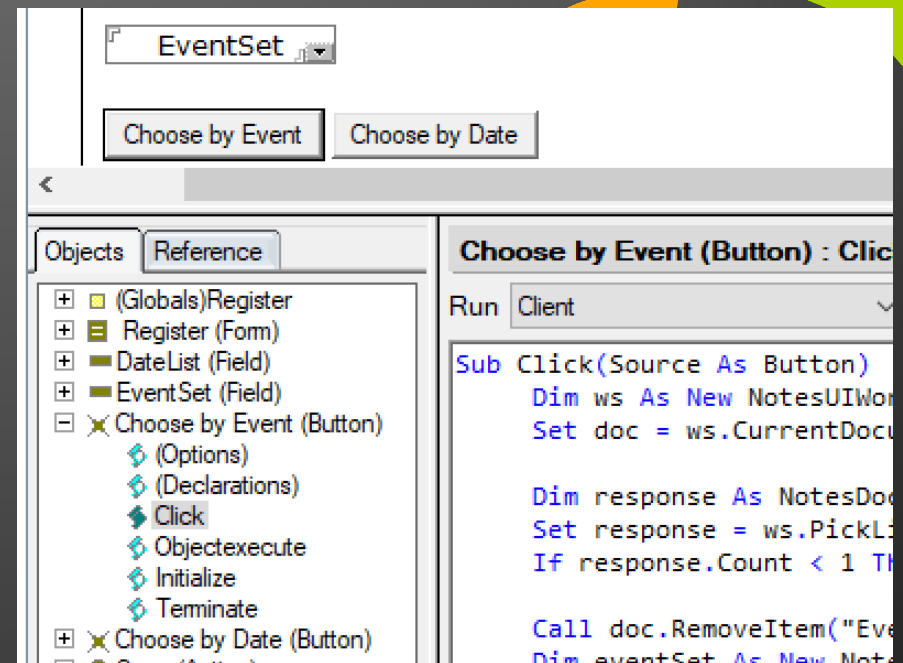
# BUILDING A BASIC APP

- Pro tip: Hide When Formulas ... adjust True/False thinking
- Let's have a look



# DOMINO DESIGNER

- Code hides in a lot of places:
  - Form and view scripts
  - Form and view action buttons
  - Buttons on a page
  - Database script
  - Outline and Navigator actions
  - Shared fields and columns
  - Subforms
  - Hide when formulas
  - Most of all: agents



# AGENTS

- Agents are (very) roughly similar to stored procedures in SQL
- The main conceptual use is for scheduled or triggered actions
  - Daily maintenance
  - When new mail arrives
  - When documents are modified
- The signer of the agent determines much of what it can do
  - Can specify another user to run as
  - Can specify whether higher-privilege operations are allowed



# AGENTS – THE “SWISS ARMY KNIFE”

- Agents have long served as a “catchall” place for logic that doesn’t fit elsewhere
- They can be used for:
  - Scheduled batch processing
  - On-command complex code (e.g. a button on a form)
  - Permission elevation (RunOnServer)
  - Complex logic in classic web apps (WebQueryOpen and WebQuerySave)
  - Using Java in the client or server
  - Janky REST APIs
  - Writing out an entire web app (rarely)



# DEBUGGING AND TRACING

- LotusScript / Java
  - NotesLog class
    - Agentlog.nsf
  - On Error statements in LotusScript
    - Print statements
    - Log.nsf
  - Try / Catch in Java
  - Debuggers for tracing
  - Object variable not set
- Where errors are happening
  - Agents, scripts, buttons, Actions, queryOpen, querySave, queryClose, postSave, postOpen, Database script
  - Database A agent acting on data in Database B
  - Security: ACL restrictions, Agent signer rights



# DEBUGGING AND TRACING

- Logging
  - Print "Error " & Cstr(err) & ": " & Error\$ & " – on line " & Cstr(eri)
  - Use the NotesLog class
  - OpenNTF Open Log project + others
- Why errors are happening
  - Logic, syntax, bad data
  - Security
  - Upgrades
- What you cannot debug
  - @Function language scripts



# WEB APPS IN DOMINO

- Default rendering or Passthru HTML
  - You really shouldn't
- XPages
  - Powerful if outdated
  - OpenNTF Jakarta EE XPages project
- Nomad
  - Nomad web Notes client in a browser
- DRAPI for data access & other REST APIs
  - <https://opensource.hcltechsw.com/Domino-rest-api/topicguides/introducingrestapi.html>
  - Domino Access Services, custom Java, XPages
  - Serdar Basegmez
    - <https://www.youtube.com/watch?v=xWOcgCJMKI&t=34s>



# “CLASSIC” DOMINO WEB

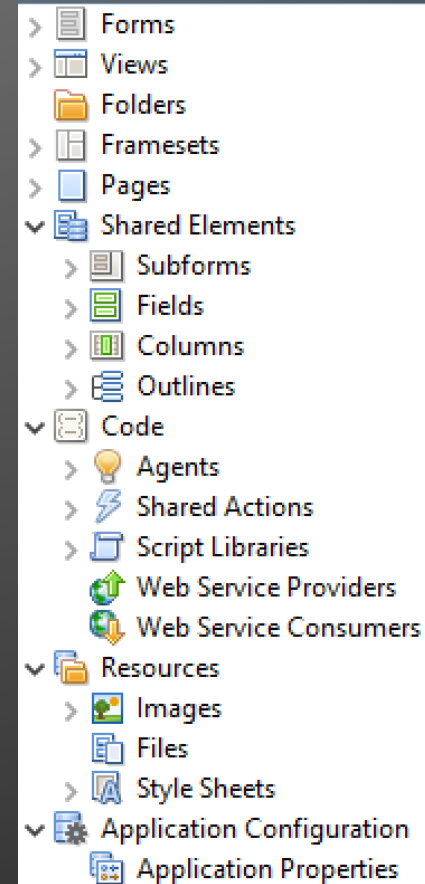
- Adapted Notes client elements to the web... kind of
  - Uses the same types of Forms, Subforms, Views, etc. as the client
- Any moderately-complex app is going to be full of “pass-through HTML” parts
- Very, very difficult to maintain at scale
- Basically: only use this if you inherited the project

```
<input type="text" value="HTTP_X_GEO_COUNTRY T" /> <input type="text" value="HTTP_X_GEO_COUNTRYCO" />
<input type="text" value="HTTP_X_GEO_CONTINENT T" /> <input type="text" value="HTTP_X_GEO_LATITUDE" />
<input type="text" value="ExistingRegistrations T" /> <input type="text" value="DateList T" /> <input type="text" value="DisplayListAs T" /> <input type="text" value="Fail" />
<div id="header"><div class="blockImport">
<Computed Subform>
</div></div>
<div id='main-body'>
<div id="linksbar"><div class="blockImport"> <input type="text" value="Linksbar T" /> </div>
</div></div>
<div id="content"><div class="blockImport">
<Computed Subform>
[</span>]</div></div>
</div> <!-- #main-body -->
```



# “CLASSIC” DOMINO WEB

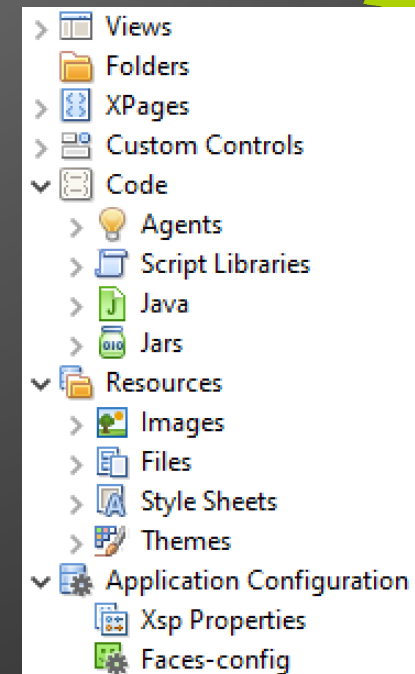
- Most of the design elements in Designer are used for web apps
  - Some more than others based on developer and app age
- Elements can be marked as hidden from Notes or the web
  - Web-heavy apps often have entirely-distinct forms for the web





# XPAGES

- XPages make use of some previously-existing and XPages-specific design elements
  - The presence of XPages and Custom Controls are a dead giveaway
  - Some Agents and Script Libraries may be used just for XPages or also for older apps



# NOAMD

- “Nomad” as a term refers to both iOS/Android apps and Nomad Web
- In all cases, it’s a version of the traditional Notes client built for the target device
- There are special considerations for apps, but largely these are “run my existing old app as-is”
- Though Nomad Web is in a browser, it doesn’t use HTML, CSS, etc. – it runs in its own Web Assembly environment



# DOMINO REST API

- Known as DRAPI, this is a relatively-recent entrant
- It's a way to layer schemas and rules on your Domino data, and then access it via a REST API
- It's unrelated to other dev types, other than also using documents and views
- This is a route to go if your app is written outside of Domino or you want to integrate with existing Domino data



# RESOURCES

- More information
  - <https://collaborationtoday.info/>
  - [https://ds\\_infolib.hcltechsw.com/ldd/dominowiki.nsf/](https://ds_infolib.hcltechsw.com/ldd/dominowiki.nsf/)
  - <https://hclsoftwareu.hcltechsw.com/>
- For XPages: NotesIn9
- What can OpenNTF offer?
  - Projects library
  - Code snippets
  - Education on YouTube
    - [openntf.org/webinars](https://openntf.org/webinars)
- HCL Support News on Telegram
  - <https://t.me/dominonotes>

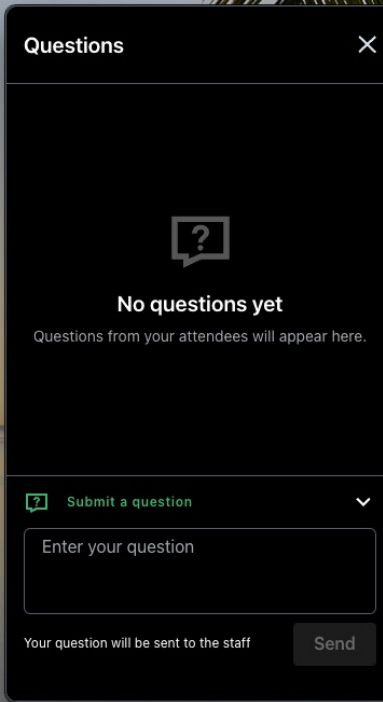


# QUESTIONS?

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>



The screenshot shows a mobile interface for a 'Questions' pane. At the top, it says 'Questions' with a close button (X). Below that, there is a question mark icon in a speech bubble. The text reads 'No questions yet' and 'Questions from your attendees will appear here.' At the bottom, there is a 'Submit a question' button with a dropdown arrow. Below the button is a text input field with the placeholder 'Enter your question'. At the very bottom, it says 'Your question will be sent to the staff' and a 'Send' button.

