

# YottaDB Management GUI

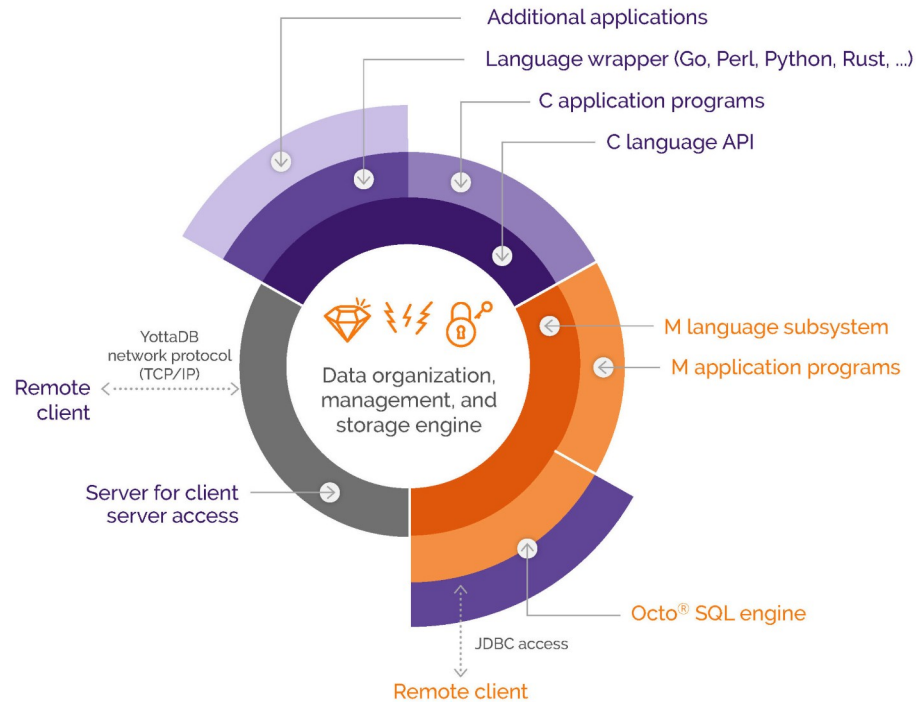
A new way to work with YottaDB

- A mature, high performance, hierarchical key-value, language-agnostic, NoSQL database whose code base scales up to mission-critical applications like large real-time core-banking and electronic health records, and also scales down to run on platforms like the Raspberry Pi Zero, as well as everything in-between.
- *Rock Solid. Lightning Fast. Secure. Pick any three.*

YottaDB is a registered trademark of YottaDB LLC

# Architecture

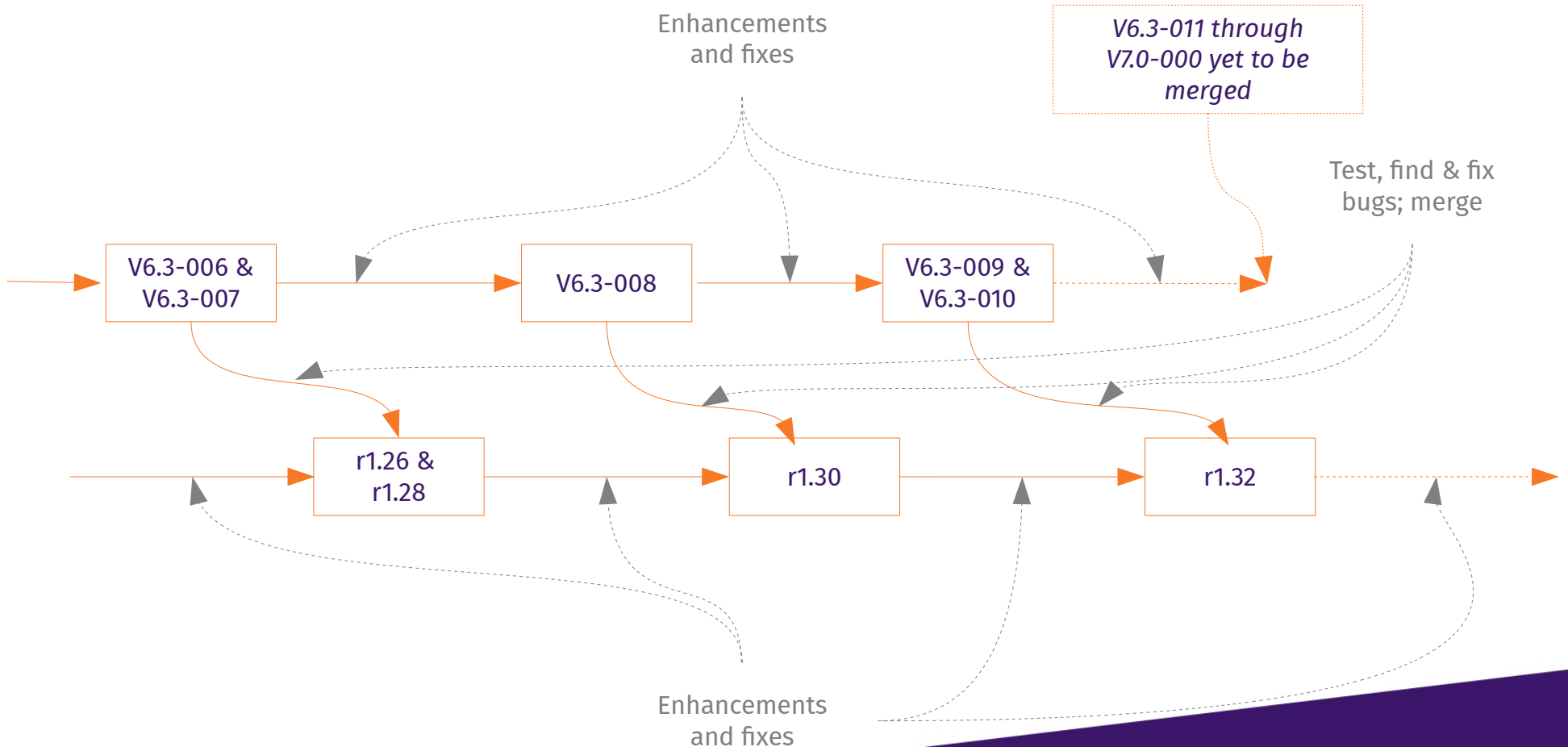
## YOTTADB DATA-CENTRIC ARCHITECTURE



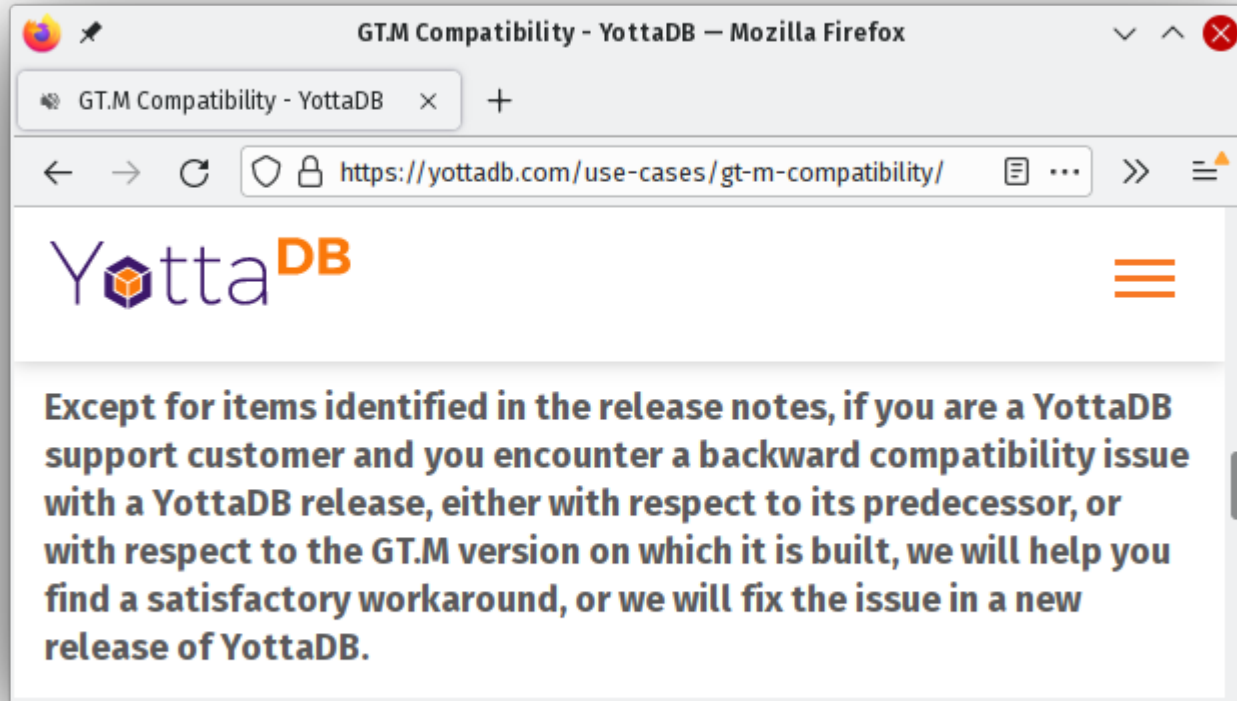
# Why a GUI?

- Do you remember how to change a key size on GDE?
  - Is it -key, or -rec? (I honestly have no idea!)
  - Does it go on the region or segment?
- Too difficult to use the vi editor to edit a routine using “zedit”?
- How do you know what processes are running on your system?
- Can’t figure out how to get a list of tables in Octo?
- A GUI is for you!

# Compatibility with Upstream



# Upward Compatibility Guarantee







# YottaDB




**System Management** ▾



**System Explorer** ▾



**Utilities** ▾



**Documentation** ▾



# YottaDB



System Management ▾



System Explorer ▾



Utilities ▾



Documentation ▾






# YottaDB




ادارة النظام



مستكشف النظام



خدمات



توثيق



### Running Processes

PID	Process Name	Device	Routine	CPU Time
<a href="#">33</a>	mumps			00:00:00
<a href="#">279</a>	yottadb	Background-S8089	LOOP+2*%YDBWEB	00:00:00
<a href="#">287</a>	mumps			00:00:00
<a href="#">291</a>	mumps			00:00:00
<a href="#">304</a>	mumps			00:00:00
<a href="#">306</a>	mumps			00:00:00





Home / System Administration / Global Directory Editor (GDE)

# Global Directory Editor (GDE)

## Names

Name	Region	Segment	File
<input type="checkbox"/> Local Locks	DEFAULT	DEFAULT	/home/vehu/g/vehu.dat
<input type="checkbox"/> %ydbocto*	OCTO	OCTO	/home/vehu/g/octo.dat
<input type="checkbox"/> *	DEFAULT	DEFAULT	/home/vehu/g/vehu.dat
<input type="checkbox"/> BMXTMP	TEMP	TEMP	/home/vehu/g/temp.dat
<input type="checkbox"/> CacheTemp*	TEMP	TEMP	/home/vehu/g/temp.dat
<input type="checkbox"/> DISV	TEMP	TEMP	/home/vehu/g/temp.dat
<input type="checkbox"/> DOSV	TEMP	TEMP	/home/vehu/g/temp.dat
<input type="checkbox"/> HLTMP	TEMP	TEMP	/home/vehu/g/temp.dat
<input type="checkbox"/> KMPTMP	TEMP	TEMP	/home/vehu/g/temp.dat
<input type="checkbox"/> SPOOL	TEMP	TEMP	/home/vehu/g/temp.dat
<input type="checkbox"/> TEMP	TEMP	TEMP	/home/vehu/g/temp.dat
<input type="checkbox"/> TMP	TEMP	TEMP	/home/vehu/g/temp.dat



[Home](#) / [System Explorer](#) / Globals

## Globals

Search  
PS\* 

16 Global(s)

[^PS](#)[^PSB](#)[^PSCST](#)[^PSD](#)[^PSDRUG](#)[^PSF](#)[^PSI](#)[^PSMDE](#)

^%Z ×

^PS ×

Globals Subscript Search

Number of nodes to show  
100 ▾

&gt; ^PS(50.073)

&gt; ^PS(50.0731)

v ^PS(50.0732)

^PS(50.0732,0)

DUE QUESTION^50.0732^23^22

&gt; ^PS(50.0732,1)

&gt; ^PS(50.0732,3)

&gt; ^PS(50.0732,4)

&gt; ^PS(50.0732,5)

&gt; ^PS(50.0732,6)



Home / System Explorer / OCTO Tables

# OCTO Tables

PATIENT ×

Search PATIENT

128 Table(s)

PATIENT

PATIENT\_ALIAS

PATIENT\_ALLERGIES

PATIENT\_ALLERGIES\_COM

PATIENT\_ALLERGIES\_COM

PATIENT\_ALL\_CHART\_MAI

PATIENT\_ALL\_DRUG\_CLAS

PATIENT\_ALL\_DRUG\_INGR

PATIENT\_ALL\_ID\_BAND\_M

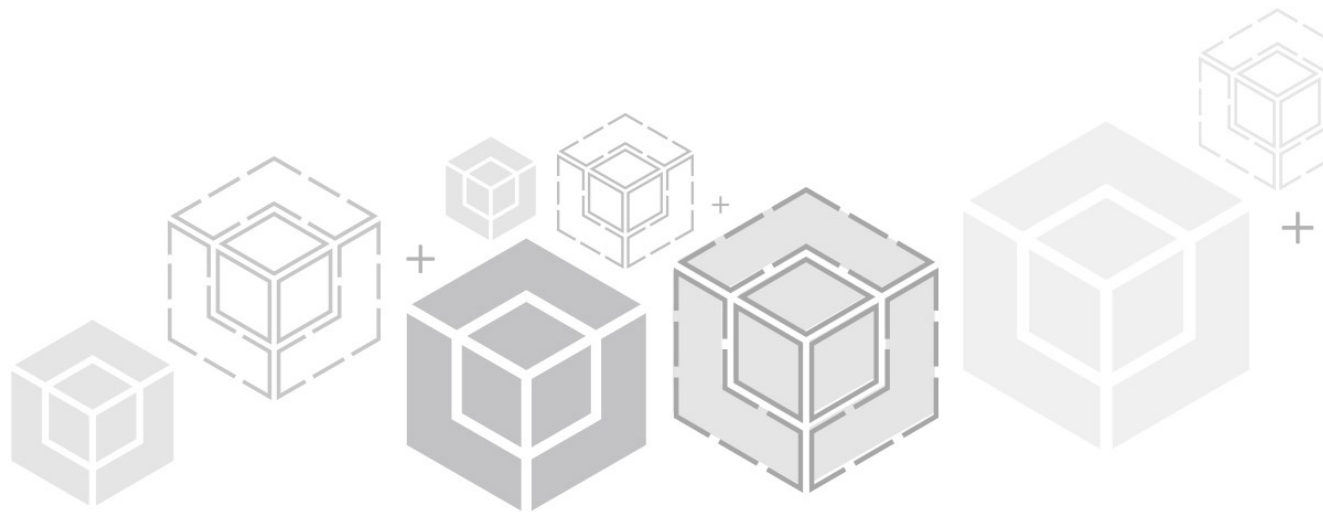
## SQL Statement:

```
1 SELECT * FROM PATIENT LIMIT 100;
```

## EXECUTE

	PATIENT_ID	NAME	SEX	SELF_IDENTIFIED_GENDER	DATE_OF_BIRTH	AGE	MARITAL_STATUS	RACE	OCCUPATI
1	1	ZZZRETFIVEFIFTYONE,PATIENT	M		2350407	86	2		
2	2	ZZZRETFIVEFORTYSEVEN,PATIENT	M		2350407	86	2	3	NURSE
3	3	EIGHT,PATIENT	M		2350407	86	2	3	LABORER
4	4	ZZZRETIREDZERO,PATIENT	M		2350407	86	6	3	TENNIS ELBOW :
5	5	ZZZRETIREFIFTYSEVEN,PATIENT	M		2350407	86	2	1	BUS DRIVER
6	6	ZZZRETIREFIFTYTHREE,PATIENT	M		2350407	86	6	3	TENNIS

# Demo



# Software Stack

- <https://gitlab.com/YottaDB/UI/YDBAdminOpsGUI/>
- Developed using npm packages
  - Vue.js with Quasar Framework
  - Built using npm into a servable bundle
  - Node.js is not used to serve the code; a small M Web Server is used instead
  - Talks to YottaDB using web services

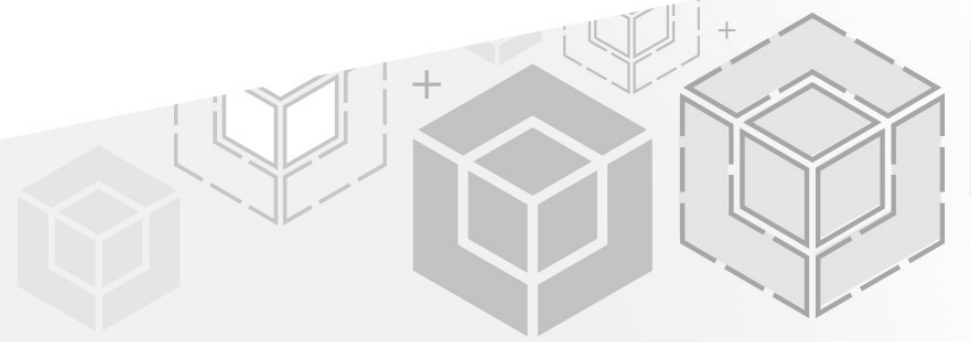


## Try for yourself (port 8089)

```
docker run -d -p 2222:22 -p 8001:8001 -p 9430:9430 -p  
1338:1338 -p 8089:8089 -name=vehu samindcon/vehu
```

# Install and Running yourself

- The software is in flux right now
- We are trying to have the most streamlined installation
- Current installation is at the repo
  - `npm install && npm run build && $ydb_dist/yottadb -run Start^%YDBWEB`



YottaDB

*Thank You!*

[yottadb.com](http://yottadb.com)