

mt



HI 007

007

Offline data manipulations for everyone with Plug-Ins and SQLite

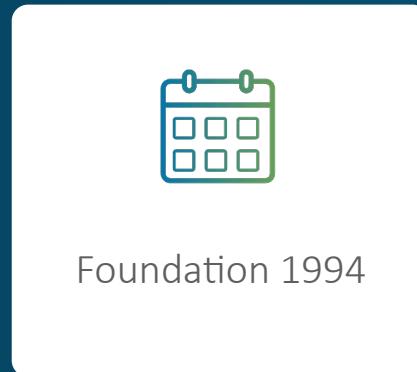
31.03.2023

Philipp Hartenfeller, Senior Consultant

APEX World 2023, Zeist

Facts and figures.

Your partner for digital transformation.
Individual IT solutions from one single source.



Foundation 1994



Headquarter

Ratingen

Branches

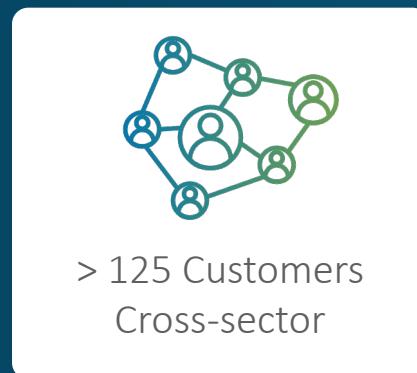
Frankfurt am Main,
Köln, München, Hamburg



> 360 Employees



approx. 48 Mio. €
Turnover in 2021



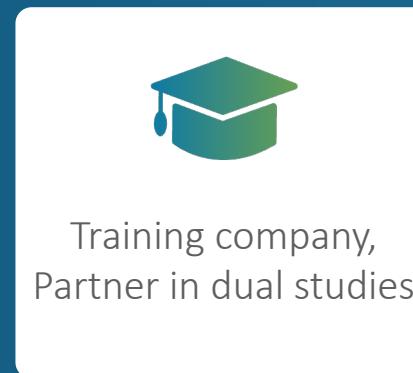
> 125 Customers
Cross-sector



Manufacturer-neutral



Certified partner
leading technology
manufacturer



Training company,
Partner in dual studies

\$ whoami



Philipp Hartenfeller

- Düsseldorf, Germany
- Master IT-Management
- Since 2016 @ MT AG
- Senior Consultant – Oracle APEX
- Mostly doing WebDev, DBs and APEX Testing (<https://lct.software>)

Blog: <https://hartenfeller.dev/blog/>



[@phartenfeller](https://twitter.com/phartenfeller)



[@phartenfeller@mastodon.social](https://phartenfeller@mastodon.social)



Oracle ACE
Associate

What this talk is about

- Offline APEX is no new topic, there have been a few methods already
- I am interested in **HOW** it should be usable in APEX (and the technical details)

What this talk is about

How should it be accessible ?

- Low Code Interface -> No JavaScript Code wrangling
 - **APEX Plug-Ins**
- Easy to configure
 - Enter your Query and let's go!

What this talk is about

How should it be accessible ? How should it work ?

- Low Code Interface -> No JavaScript Code wrangling
 - **APEX Plug-Ins**
- Easy to configure
 - Enter your Query and let's go!
- Fast
 - Handle lots of data
 - Concept to merge changes back to DB
 - Somewhat future proof

Proof of Concept: Not production ready!

Create App + PWA functionalities + Import Plug-Ins

App Builder +

apex.phartenfeller.de/ords/f?p=4000:1500:13850835133261::::

APEX App Builder SQL Workshop Team Development Gallery Search ? admin plugins

About
The App Builder enables you to design, develop and deploy applications which are beautiful and responsive.
[Learn More ...](#)

Recent
Offline First AG Gri - 120
AG Grid Test - 114
Flows for APEX - 115
World Map Plugin - 105
Slide-Over / Drawer - 106
Flows for APEX - 984339
Holiday Approval (de - 984341)
Rich File Preview PI - 322
AG Grid Test 114
Flows for APEX 115
Flows for APEX 984339
Holiday Approval (demo app to show how to integrate 984341)
Offline First AG Grid 120
World Map Plugin 105
Slide-Over / Drawer 106
Flows for APEX - 984339
Holiday Approval (de - 984341)
Rich File Preview PI - 322

Actions Go Reset Create >

1 - 8

AG Grid Test 114	Flows for APEX 115	Flows for APEX 984339	Holiday Approval (demo app to show how to integrate 984341)	Offline First AG Grid 120
Rich File Preview Plugin 322	Slide-Over / Drawer 106	World Map Plugin 105		

admin plugins en

Copyright © 1999, 2022, Oracle and/or its affiliates.

Oracle APEX 22.1.0

Setup Sync Plug-In

App Builder Offline APEX Demo

apex.phartenfeller.de/ords/f?p=4000:1:13850835133261::RP:FB_FLOW_ID,F4000_P1_FLOW:123,123

APEX App Builder SQL Workshop Team Development Gallery Search admin plugins

Application 123

Application 123 - Offline APEX Demo Edit Application Definition

About

Use the Application home page to run, edit, import, export, copy, or delete an application. Select a page to edit the page properties, or click **Create Page** to add a page to the application.

Learn More ...

Tasks

Delete this Application >

Copy this Application >

Add Feature Page >

Manage Backups >

Browse by Facets >

Recently Edited Pages

0. Global Page

9999. Login Page

1. Home

Run Application Supporting Objects Shared Components Utilities Export / Import

Actions Create Page >

0 - Global Page 1 - Home 9999 - Login Page 1 - 3

admin plugins en

Copyright © 1999, 2022, Oracle and/or its affiliates.

Oracle APEX 22.1.0

Setup Plug-Ins that use offline storage

The screenshot shows the Oracle APEX Page Designer interface. The top navigation bar includes links for Page Designer, Offline APEX Demo, App Builder, SQL Workshop, Team Development, Gallery, a search bar, and user icons. The left sidebar shows the application structure: Application 123 \ Page Designer. The main content area displays the layout of a page titled "Home". The layout consists of several regions: BANNER, AFTER LOGO, BEFORE NAVIGATION BAR, AFTER NAVIGATION BAR, TOP NAVIGATION, BREADCRUMB BAR, REGION BODY, REGION CONTENT, and SUB REGIONS. The BREADCRUMB BAR region contains the text "</> Offline APEX Demo". On the right side, there is a sidebar titled "Action" containing configuration options:

- Filter (dropdown menu showing "1")
- PK Column (dropdown menu showing "ID")
- Last Changed Column (dropdown menu showing "LAST_CHANGED")
- Rows to fetch per request (text input showing "500")
- Sync timeout minutes (text input showing "60")
- Execution Options** (checkbox checked)
- Sequence (text input showing "10")

The URL in the browser address bar is `apex.phartenfeller.de/ords/f?p=4000:4500:13850835133261::FB_FLOW_ID,FB_FLOW_PAGE_ID:123,1#render?ree_container`.

Go offline, change something and sync to DB

The screenshot shows a web browser window with the title bar "Offline APEX Demo". The main content area has a blue header "Offline APEX Demo" and a user icon "admin". Below the header, there are two main sections: "People List" and "Form".

People List: This section contains a "Selected Id" input field and a "Search" input field. Below these are a list of names: Ward Moncure, Marvella Moncure, Peggie Loera, Noelle Gervin, Lola Loera, Adam Binette Hello, Daphine Dardar, and Rosemarie Moncure. "Ward Moncure" is highlighted with a blue vertical bar.

Form: This section contains five input fields labeled "Id", "Email", "Salary", "Company", and "Skillset".

At the bottom of the page is a navigation bar with icons for Home, App 123, Page 1, Session, Debug, Quick Edit, Customize, and Help.

What I didn't show

You actually need packages

- For Plug-Ins
- Code to merge your data back to the source
(more on tha later)

Data Synchronization



Why this topic is important

- Allowing users to edit data without a connection to the DB has effects of decentralization
- You still want a single source of truth
- This results in new challenges for data consistency

Setup Sync

The screenshot shows the 'Identification' tab of the 'Sync Offline Data [Plug-In]' setup interface. It includes sections for 'Source Query', 'Unique Storage ID', 'Storage Version', 'PK Column', 'Last Changed Column', 'Rows to fetch per request', and 'Sync timeout minutes'. The 'Source Query' section contains a complex SQL query for selecting various columns from the 'OFFL_PEOPLE' table, including calculated fields for salary and last changed timestamp.

Action	Sync Offline Data [Plug-In]
Source Query	<pre>select ID , EMAIL , SALARY , COMPANY , SKILLSET , LAST_NAME , FIRST_NAME , SHIRT_SIZE , BIRTHDAY , round((LAST_CHANGED - date '1970-01-01') * 60 * 60 * 24) as LAST_CHANGED from OFFL_PEOPLE</pre>
Unique Storage ID	people
Storage Version	1
PK Column	ID
Last Changed Column	LAST_CHANGED
Rows to fetch per request	500
Sync timeout minutes	60

- Dynamic Action Plug-In on page-load:
Handles all the sync
- Changes to query -> Increase version
- Last Changed column needed for conflict management (more on that later)

Sync Timeline - 1

1. Cache current HTML page
2. Synchronize Client Changes with DB
 - Send client updates to central DB
 - Delete corresponding rows on the client
 - It is now in the control of the DB what will happen with these rows

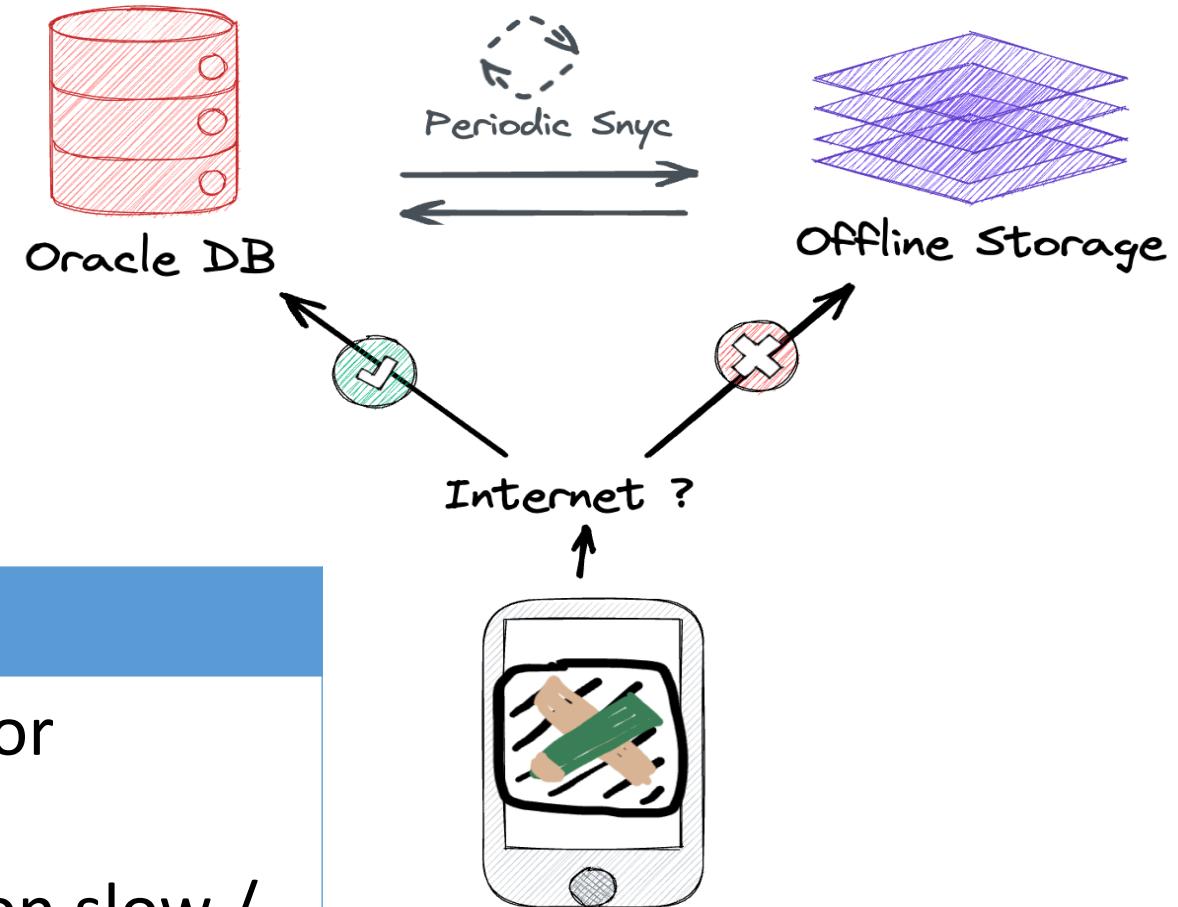
Sync Timeline - 2

3. Synchronize DB Changes with Client

- DB sends PK + Timestamp for each row:
 - PK known and Timestamp matches: 
 - PK unknown or Timestamp mismatch:  request full row -> insert / update
- Delete rows not present in central DB anymore:
 - Store every PK received from server in a temp table
 - After receiving all rows, delete local rows where PK not in temp table

Data Access Strategies - Online First

- First try to get data from the DB
- Only when no connection use offline storage



Pros

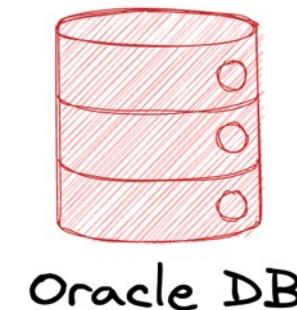
- “Normal APEX Mode”
- Less conflicts
- Live data

Cons

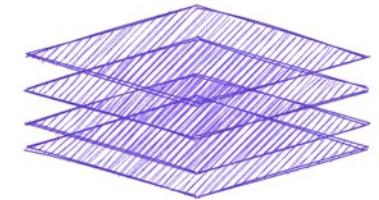
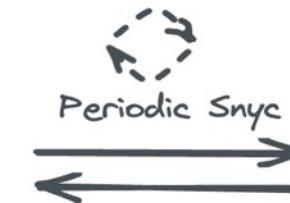
- Much logic for consumers
- What to do on slow / unreliable connections

Data Access Strategies - Offline First

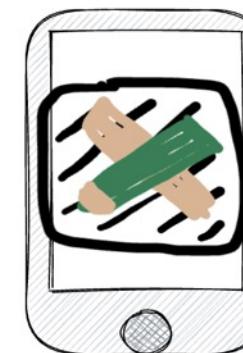
- First try to get data from the DB
- Only when no connection use offline storage



Oracle DB



Offline Storage



Pros

- Reliable Experience
- Less logic for consumers

Cons

- Only updates after re-sync
- More conflicts

I chose this approach
because of this

💡 Build different experiences for different use cases

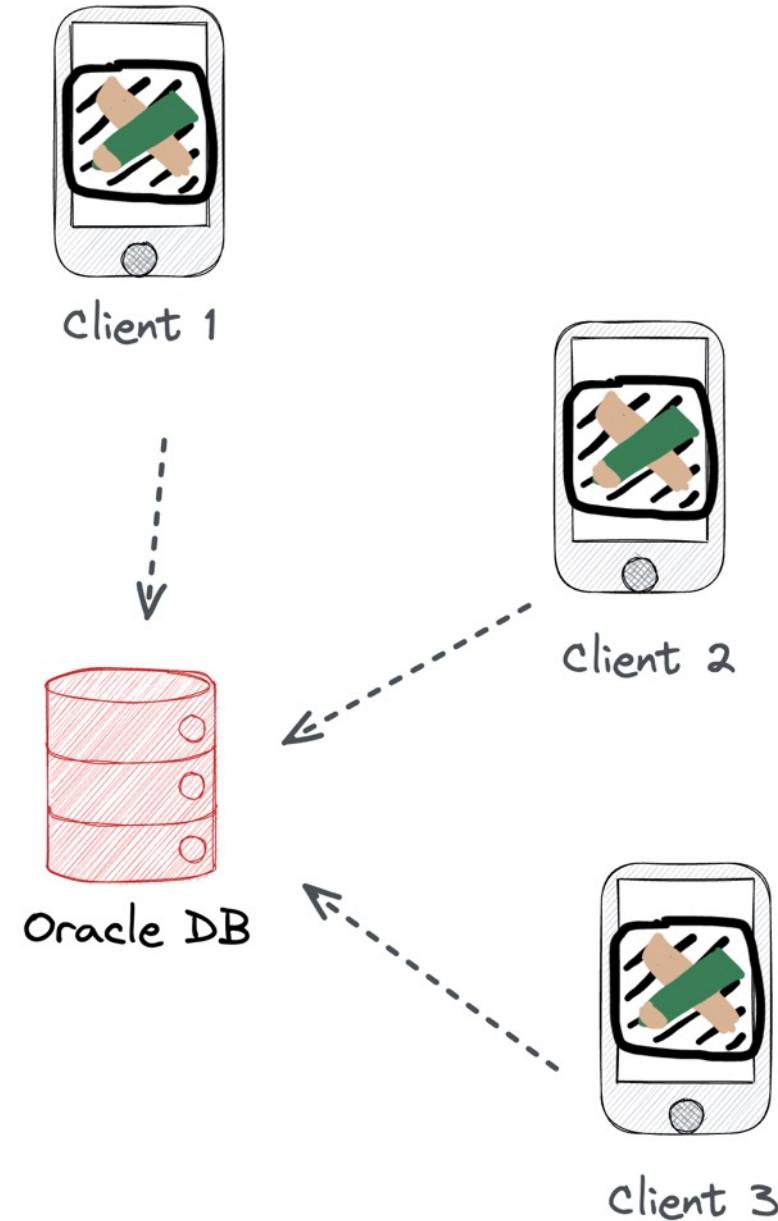
- Offline-first has downsides
- Make sure only people who need offline access use it
- Field technician gets offline App / Page
- Office worker gets online App / Page



Photo by [MIOPS Trigger](#) from [Unsplash](#)

Change Synchronization

- Local changes need to be written back to the central DB
- Ensure all local updates are properly saved in the central DB
- Different scenarios



Change Synchronization - Potential outcomes

a: Single Client Row Update (✓)

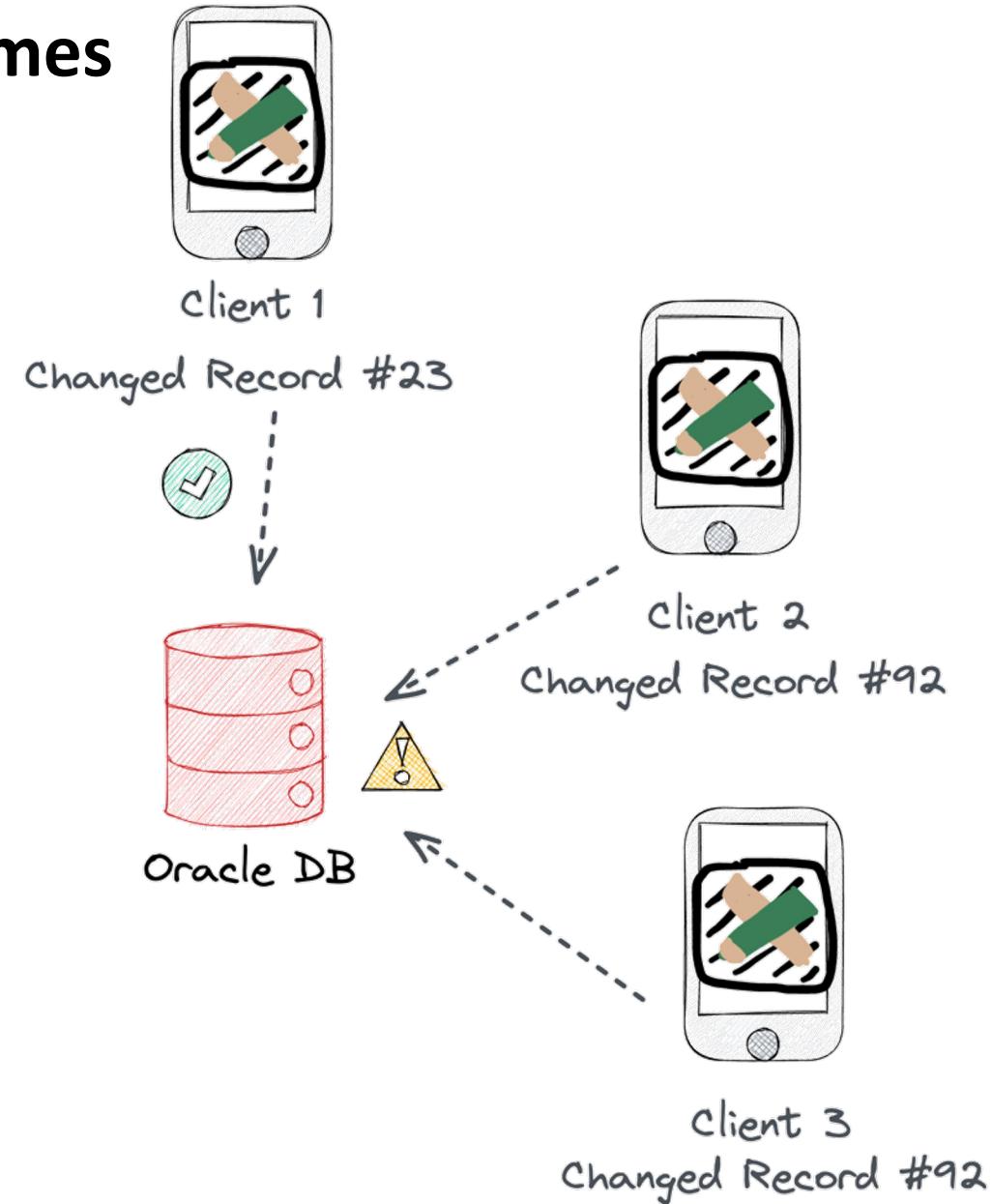
- Only one client modifies the row, straightforward synchronization

b: Sequential Client Row Updates (✓)

- Client 2 updates row after receiving Client 1's changes, preserving data integrity

c: Concurrent Client Row Updates (!)

- Two clients modify the same row version simultaneously, potential data conflict



How changes are stored

- Don't directly insert into source table to handle conflicts
- Generic changes table

Storage ID Storage Version	Data JSON Change Type	Timestamp Src Timestamp Chng	Audit Cols	Import Failed
people 2	{ ... } I, U or D	192....0003 192....9843	JAMESBOND 00716(Session)	0 Someday False :D
...				

How changes are processed

- Package gets called where you can implement your own logic
 - Resolve the conflicts
 - Call your existing Table- / Transaction-APIs
 - Future consideration: Bulk Processing with Transaction IDs
 - No trigger, instead directly call a process package
 - Add transaction IDs so you can bulk process all changes
- ← Up to you how to do it

Conflict Resolution #1 – Last Write Wins

Design Decision



Nothing to consider:
Just write changes

Photo by [Matthias Speicher](#) from [Unsplash](#)



Photo by [Ryan Moreno](#) from [Unsplash](#)

Conflict Resolution #2 – Merge Conflict Detection

Changes Table

ChngType	Data	RowSrcTimeStamp	RowChngTimeStamp	...
----------	------	-----------------	------------------	-----

U	{...}	192....0003	192....5843
---	-------	-------------	-------------



= src table row timestamp?  → update src table row timestamp now 192....5843

Conflict Resolution #2 – Merge Conflict Detection

Changes Table

ChngType	Data	RowSrcTimeStamp	RowChngTimeStamp	...
U	{...}	192....0003	192....5843	



= src table row timestamp?  → update src table row timestamp now 192....5843

Later...

U	{...}	192....0003	192....9932
			↓

src table row ts 192....5843 != 192....0003 

→ changes need to be manually merged

Conflict Resolution #2 – How to do a merge

- On conflict:
 - don't delete row from changes table and set failed = 1
- User must do merge manually
 - Show current row next to changed row (like Git 3-way-merge)

192...0003

Firstname: Peter

192...5843

Firstname: Peter

After Merge Row

Lastname: Parka



Lastname: Parker



Lastname: ???

Age: 23



Age: NULL



Age: ???

....

....

....

Technical Implementation



Where to store data

Browser Storage Options:

- LocalStorage
 - Key-Value object-store, slow, designed for small amounts
- WebSQL
 - Deprecated
- IndexedDB
 - For large data, transactions, Key-Value object-store

The thing with IndexedDB:

- Complex API
- No support for complex queries
- History of nasty bugs (mostly Safari)
- Not ACID compliant

[The pain and anguish of using IndexedDB: problems, bugs and oddities](#)

Not the DB we deserve, but the hero we need

The image shows a screenshot of a Twitter post. At the top left is the profile icon of the account, which is the Google Chrome logo. To the right of the icon is the account name "Chrome Developers" in bold black text, followed by a blue oval badge with a white checkmark and the word "Verified". Below the account name is the handle "@ChromiumDev". On the far right of the header is a three-dot ellipsis menu. The main content of the tweet is a block of text: "Yes, the plan is to eventually remove Web SQL, but 🥁 our intention is to empower developers to create their own solutions for structured storage, and we're therefore working with the **#SQLite** team to create a SQLite implementation over Wasm. This solution will replace Web SQL 💪!". Below the text is the timestamp "12:33 AM · Sep 1, 2022".

Yes, the plan is to eventually remove Web SQL, but 🥁 our intention is to empower developers to create their own solutions for structured storage, and we're therefore working with the **#SQLite** team to create a SQLite implementation over Wasm. This solution will replace Web SQL 💪!

12:33 AM · Sep 1, 2022

[Source](#)

Wait a minute... What is SQLite?

- Most used DB, top 5 most deployed software ever (estimated one trillion active DBs)
- Full-featured SQL
- Serverless (not the cloud thing)
- Transactional
- ACID
- Public Domain
- Really fast
- Stores data in a single file
- Recommended Storage Format by the US Library of Congress

[More on SQLite](#)

How does it run in the browser



WEBASSEMBLY

Compile existing software to hardware-near browser understood layer

Origin Private File System (OPFS)

- Sites get a private file system
- SQLite directly writes to here

Chrome	Edge	Firefox	Opera	Safari	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Internet	WebView Android
✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓
86	86	111	72	15.2	109	111	74	15.2	No	109

[Source](#)

Implement into APEX

WebAssembly & JavaScript

[sqlite-wasm-3410200.zip](#)
(702.24 KiB)

A precompiled bundle of `sqlite3.wasm` and its JavaScript APIs, ready for use in web applications.
(SHA3-256: 1713c3a4c25b09f5d52ed25fb81948f6081e9eeefb5a6d449aead1cd848e962c)

[Link](#)

- OPFS requires these headers:

`Cross-Origin-Embedder-Policy: require-corp`
`Cross-Origin-Opener-Policy: same-origin`

- APEX bug: .wasm invalid mime-type in header

✖ Uncaught (in promise) TypeError: Failed to execute 'compile' on 'WebAssembly': Incorrect response MIME type. Expected 'application/wasm'.

Response Headers

```
cache-control: max-age=315360000, immutable
content-disposition: inline; filename="sqlite3.wasm"; filename*=UTF-8''sqlite3.wasm
content-language: en
content-type: application/wasm; charset=utf-8
date: Mon, 16 Jan 2023 12:27:26 GMT
```

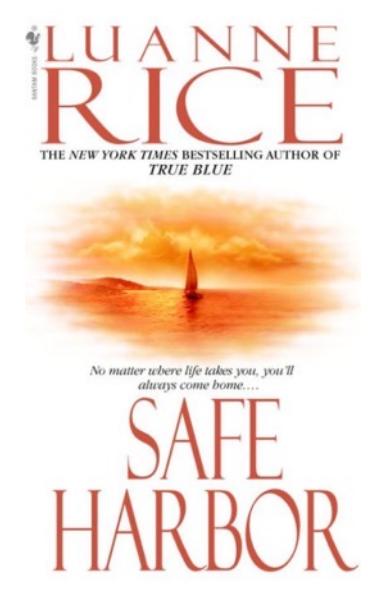
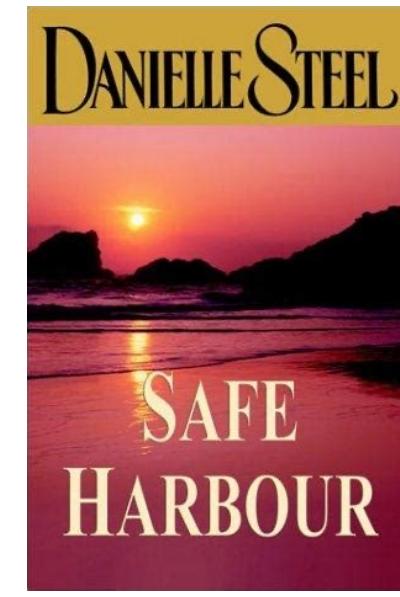
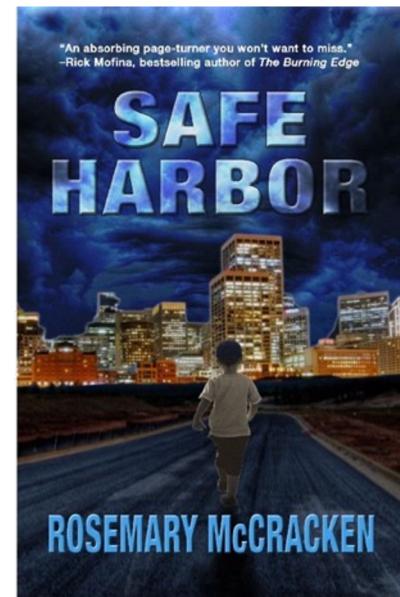
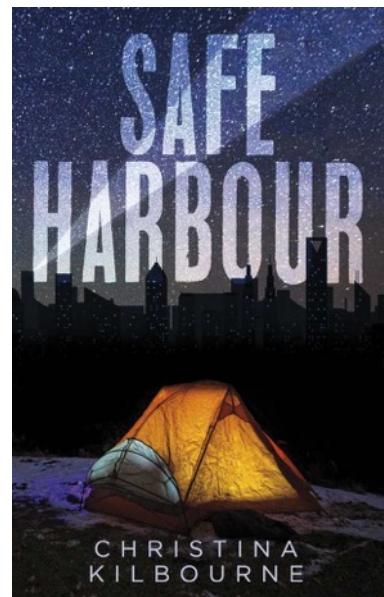
Will be fixed in 23.1

- Thanks to [Stefan !!!](#)



For now proxy magic:

- "traefik...traefik-plugin-rewrite-headers.rewrites[0].header=Content-Type"
- "traefik...traefik-plugin-rewrite-headers.rewrites[0].regex=application.*wasm.*"
- "traefik...traefik-plugin-rewrite-headers.rewrites[0].replacement=application/wasm"



How develop Plug-Ins

```
> window.hartenfeller_dev.plugins.sync_offline_data.storages.people_v1
< ▼ {config: {...}, getColInfo: f, getRowByPk: f, getRows: f, getCount: f, ...} ⓘ
  ► config: {pageSize: 500}
  ► getColInfo: () => _getColInfo(storageId, storageVersion, apex)
  ► getRowByPk: (pk) => _getRowByPk(storageId, storageVersion, pk, apex)
  ► getCount: ({ searchTerm }) => {...}
  ► getRows: ({ offset = 0, maxRows = 100, orderByCol, orderByDir, searchTerm, }) => {...}
  ► sync: () => {...}
  ► writeChanges: (rows) => {...}
```

Offline Data List

Search

→ `getRows({ searchTerm: "xyz" })`

Ward Moncure

Marvella Moncure

Peggie Loera

Noelle Gervin

Lola Loera

Adam Binette 1234

Dave Gervin

Daphine Dardar

Rosemarie Moncure

Dean Koschnitzki

→ `getRows({ offset: 10 })`

<< < Page 1 > >>

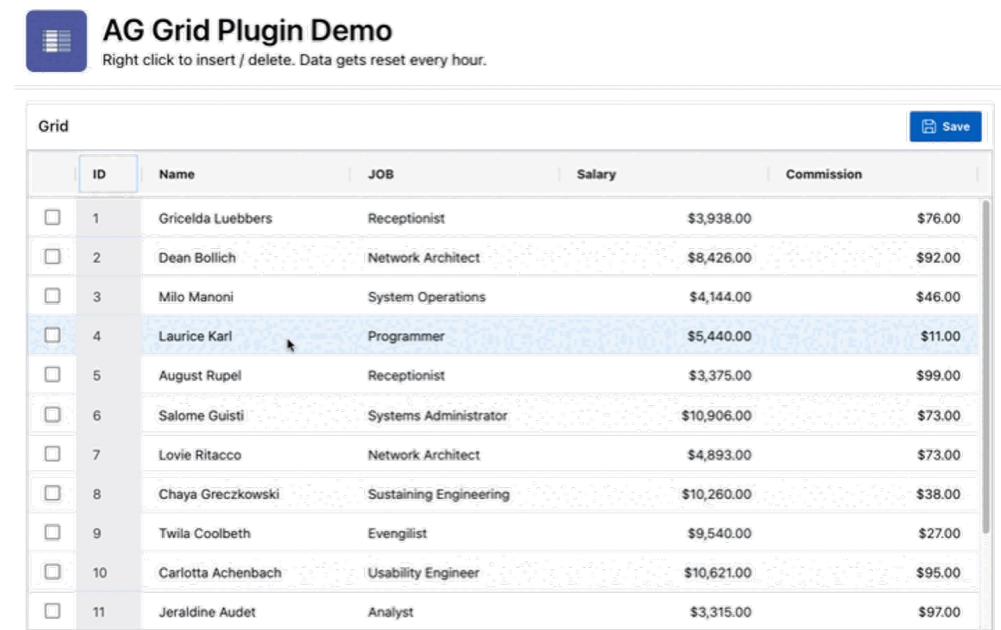
→ `getRowCount()`

Wrapping Up



What still needs to be done

- Data type handling
 - Dates, Timestamps ...
 - BLOBs
- LOVs (Select Lists, Popup LOV, Shuttle, Checkboxes...)
- Apply more table rules to client (FKs, check constraints, indexes, defaults...)
- Error message improvements
- Notify when sync completed with data changes
- Expand Plug-In Ecosystem



The screenshot shows a data grid titled "AG Grid Plugin Demo". The grid has a header row with columns labeled "ID", "Name", "JOB", "Salary", and "Commission". Below the header, there are 11 data rows, each containing a checkbox in the first column and information about an employee. Row 4, which contains the data for Laurice Karl, is highlighted with a blue selection bar and has a cursor pointing at it. The grid is set against a white background with a light gray border.

ID	Name	JOB	Salary	Commission
1	Gricelda Luebbers	Receptionist	\$3,938.00	\$76.00
2	Dean Bollich	Network Architect	\$8,426.00	\$92.00
3	Milo Manoni	System Operations	\$4,144.00	\$46.00
4	Laurice Karl	Programmer	\$5,440.00	\$11.00
5	August Rupel	Receptionist	\$3,375.00	\$99.00
6	Salome Guisti	Systems Administrator	\$10,906.00	\$73.00
7	Lovie Ritacco	Network Architect	\$4,893.00	\$73.00
8	Chaya Greczkowski	Sustaining Engineering	\$10,260.00	\$38.00
9	Twila Coolbeth	Evengilist	\$9,540.00	\$27.00
10	Carlotta Achenbach	Usability Engineer	\$10,621.00	\$95.00
11	Jeraldine Audet	Analyst	\$3,315.00	\$97.00

Why you may not want to use this

- You lose authority over your data
 - Data on the client can easily be duplicated / shared
- Merge conflicts / Inconsistent data state
- Divergent development approach for offline apps
- Increased complexity
- Scalability: Synchronization processes can demand significant data transfers
- SQLite as fast as the device using it

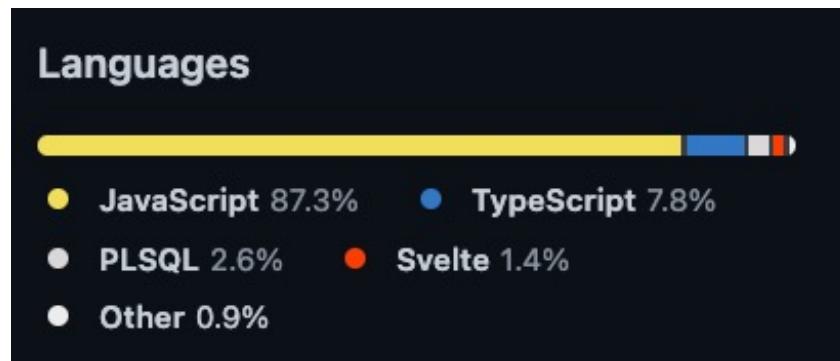


Photo by [JESHOOTS.COM](#) from [Unsplash](#)

Where can I get this ?

<https://github.com/phartenfeller/apex-off-grid>

- Needs Documentation
- Needs Refactoring
- Contributions welcome 😊



Conclusion

- I am quite happy with the result
- Sync-Plug-In easy to use, hides complexity
- Confident that SQLite is a great choice for small to large data sizes
- Great browser support
- Not sure about header requirements



Photo by [Sebastian Herrmann](#) from [Unsplash](#)

Testing APEX Apps is now as easy as creating them.

- Tailored to APEX
- Save a lot of time on regression tests
- Use our intuitive LCT-App and don't write any test code
- Testing on multiple platforms simultaneously

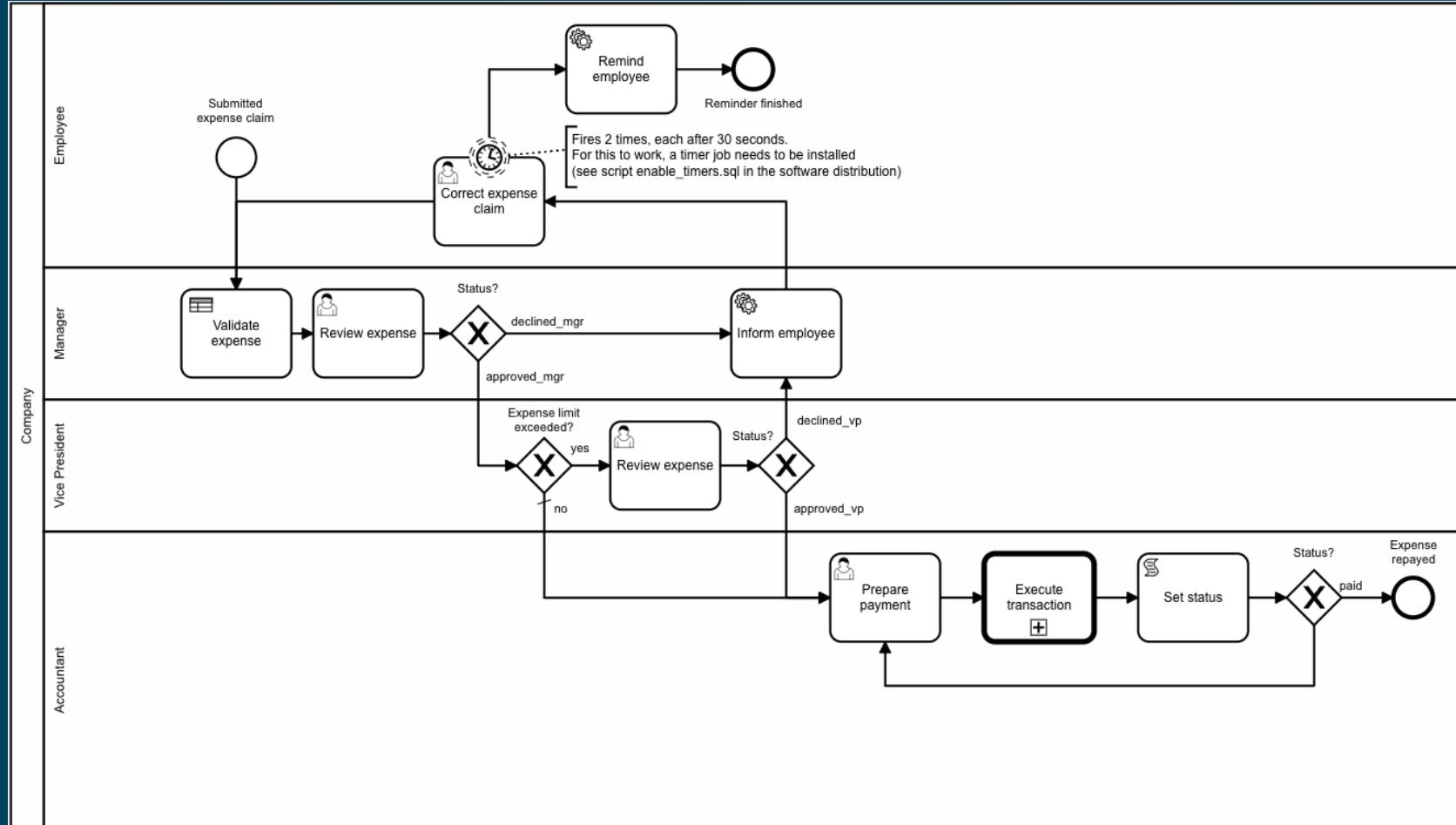


LCT



Flows for APEX

BPMN 2.0 Workflows for APEX



- Open Source
- Community Driven
- Support available



RATE ME!



**Offline data manipulations in APEX for
everyone with Plug-Ins and SQLite**

Philipp Hartenfeller

