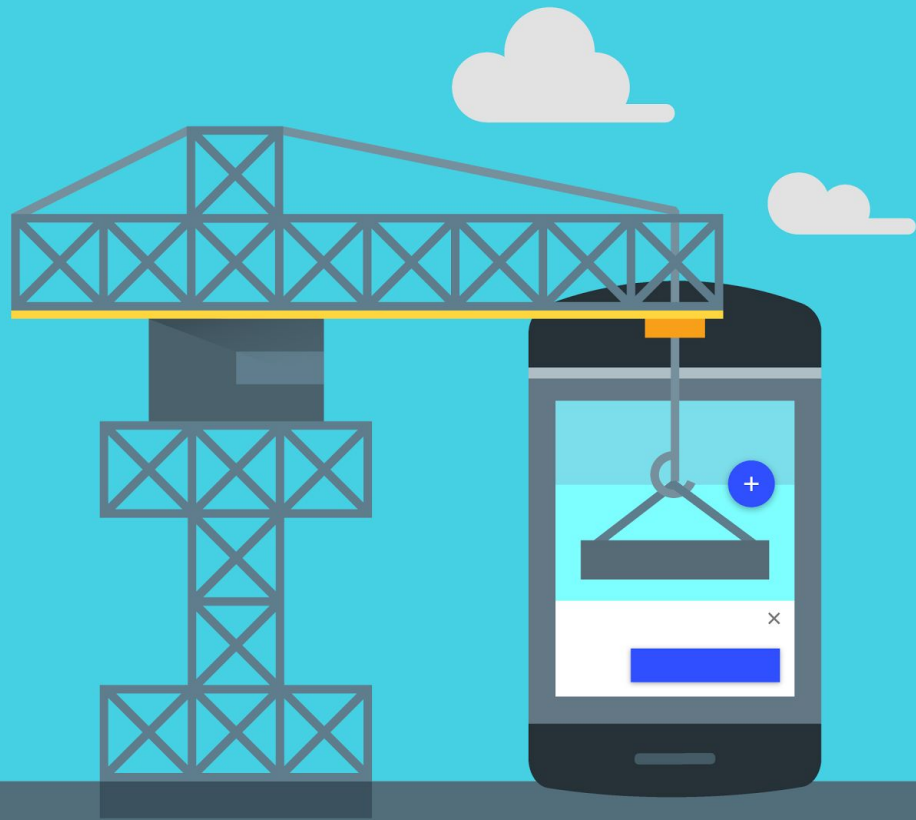


Live data in the Service Worker



IndexedDB and Cache

Why IndexedDB and Cache?

- Async
- Accessible
- Supported
- Promise-based wrappers
- Debuggable
- Storage

Storage limitations and warnings

Browser	Limitation	Notes
Chrome and Opera	No limit	Storage is per origin not per API
Firefox	No limit.	Prompts after 50 MB
Mobile Safari	50MB.	
Desktop Safari	No limit.	Prompts after 5MB
Internet Explorer (10+)	250MB.	Prompts after 10MB

How to organize data

Which data goes where?

Cache

- Static
- URL addressable
- Example: App Shell, images, stylesheets

IndexedDB

- Dynamic
- Example: JSON data

Charizard



FIRE

FLYING

#6

HP	78
Attack	84
Defense	78
Speed	100
Sp Atk	109
Sp Def	85

Flame Pokémon

It is said that Charizards fire burns hotter if it has experienced harsh battles.

Profile

Height:	1.7 m	Weight:	90.5 kg
Catch Rate:	0%	Gender Ratio:	87.5% ♂ 12.5% ♀
Egg Groups:	Monster, Dragon	Hatch Steps:	5100
Abilities:	Blaze, Solar-power	EVs:	3 Sp Att

Application

- Manifest
- Service Workers
- Clear storage

Storage

- Local Storage
- Session Storage
- IndexedDB
 - _pouch_descriptions - https://www.pokedex.org
 - _pouch_evolutions - https://www.pokedex.org
 - _pouch_monster-moves - https://www.pokedex.org
 - _pouch_monsters - https://www.pokedex.org
 - _pouch_monsters-supplemental - https://www.pokedex.org
 - _pouch_moves - https://www.pokedex.org
 - _pouch_types - https://www.pokedex.org
 - localforage - https://www.pokedex.org
- Web SQL
- Cookies

Cache

- Cache Storage
 - pokedex-static-5.0.0 - https://www.pokedex.org**
 - Application Cache

◀ ▶

#	Request	Response
0	https://www.pokedex.org/	
1	https://www.pokedex.org/css/sprites-webp-1.css	
2	https://www.pokedex.org/css/sprites-webp-10.css	
3	https://www.pokedex.org/css/sprites-webp-2.css	
4	https://www.pokedex.org/css/sprites-webp-3.css	
5	https://www.pokedex.org/css/sprites-webp-4.css	
6	https://www.pokedex.org/css/sprites-webp-5.css	
7	https://www.pokedex.org/css/sprites-webp-6.css	
8	https://www.pokedex.org/css/sprites-webp-7.css	
9	https://www.pokedex.org/css/sprites-webp-8.css	
10	https://www.pokedex.org/css/sprites-webp-9.css	
11	https://www.pokedex.org/favicon.ico	
12	https://www.pokedex.org/img/icon-144.png	
13	https://www.pokedex.org/img/icon-196.png	
14	https://www.pokedex.org/img/icon-48.png	
15	https://www.pokedex.org/img/icon-96.png	
16	https://www.pokedex.org/index.html?launcher=true	
17	https://www.pokedex.org/js/main.js	
18	https://www.pokedex.org/js/worker.js	
19	https://www.pokedex.org/manifest.json	

How to use them together

IndexedDB

Constants and data

```
const DBNAME = 'users';  
const DBVERSION = 1;  
var records = [  
  {name: 'carlos',  
    url: 'http://placekitten.com/g/200/200'}},  
  {name: 'nick',  
    url: 'http://placekitten.com/g/200/200'}},  
  {name: 'david',  
    url: 'http://placekitten.com/g/200/200'}  
];
```

Creating a database

```
function installDB() {  
  var dbPromise = idb.open(DBNAME, DBVERSION, upgradeDB => {  
    switch (upgradeDB.oldVersion) {  
      case 0:  
        upgradeDB.createObjectStore(DBNAME, {keyPath: 'name'});  
      }  
    })  
    .then(db => console.log('DB opened!', db))  
    .catch(...);  
}
```

Populating the database

```
function addUsers() {  
  var dbPromise = idb.open(DBNAME, DBVERSION);  
  dbPromise.then(db => {  
    var tx = db.transaction(['users'], 'readwrite');  
    var store = tx.objectStore('users');  
    records.map(record => {  
      store.put({name: record.name, url: record.url});  
    });  
    return tx.complete;  
  }).catch(...);  
}
```

Using the database

```
function showUsers() {  
    var dbPromise = idb.open(DBNAME, DBVERSION);  
  
    dbPromise.then(db => {  
        return db.transaction('users')  
            .objectStore('users').getAll()  
            .then(allUsers => console.log(allUsers));  
    });  
}
```

Cache

Constants and data

```
const CACHENAME = 'static-cache';  
const CACHEVERSION = 1;  
const EXPECTEDCACHENAMES = [CACHENAME + '-v' +  
    CACHEVERSION];  
  
var staticAssets = [  
    'index.html',  
    'styles/main.css'  
];
```


Caching the app shell

```
function cacheStaticAssets(event) {  
  event.waitUntil(  
    caches.open(CACHENAME + '-v' + CACHEVERSION)  
      .then(cache => {  
        return cache.addAll(staticAssets);  
      }).catch(...)  
  );  
}
```

Updating caches

```
function updateCaches(event) {  
  event.waitUntil(  
    caches.keys().then(cacheNames => {  
      return Promise.all(  
        cacheNames.map(cacheNames => {  
          if (EXPECTEDCACHENAMES.indexOf(CACHENAME) === -1) {  
            return caches.delete(CACHENAME);  
          }  
        })  
      )  
    })  
  )  
}
```

Fetching from the cache

```
function respondCacheFirst(event, altUrl) {  
  event.respondWith(caches.match(altUrl || event.request)  
    .then(response => {  
      if (response) {  
        return response;  
      }  
      return fetch(altUrl || event.request);  
    }  
  ))  
  
  ...  
}
```

IndexedDB & Cache

Import scripts

```
// In service worker
```

```
importScripts('scripts/idb.js');  
importScripts('scripts/idb_functions.js');  
importScripts('scripts/cache_functions.js');
```

Install event

```
self.addEventListener('install', event => {  
  installDB();  
  addUsers();  
  cacheStaticAssets(event);  
});
```

Fetch event

```
self.addEventListener('fetch', event => {  
  if (event.request.url.includes('users')) {  
    respondCacheFirst(event, 'index.html');  
    showUsers();  
  } else {  
    respondCacheFirst(event, false);  
  }  
});
```

Activate event

```
self.addEventListener('activate', event => {  
  updateCaches(event);  
});
```


Resources

- [Offline Storage for PWA's](#) (Addy Osmani)
- [Detailed support testing](#)
- [Support for IndexedDB](#)
- Jake Archibald's [IndexedDB Promised](#)
- [Pokedex.org](#)
- [importScripts](#)