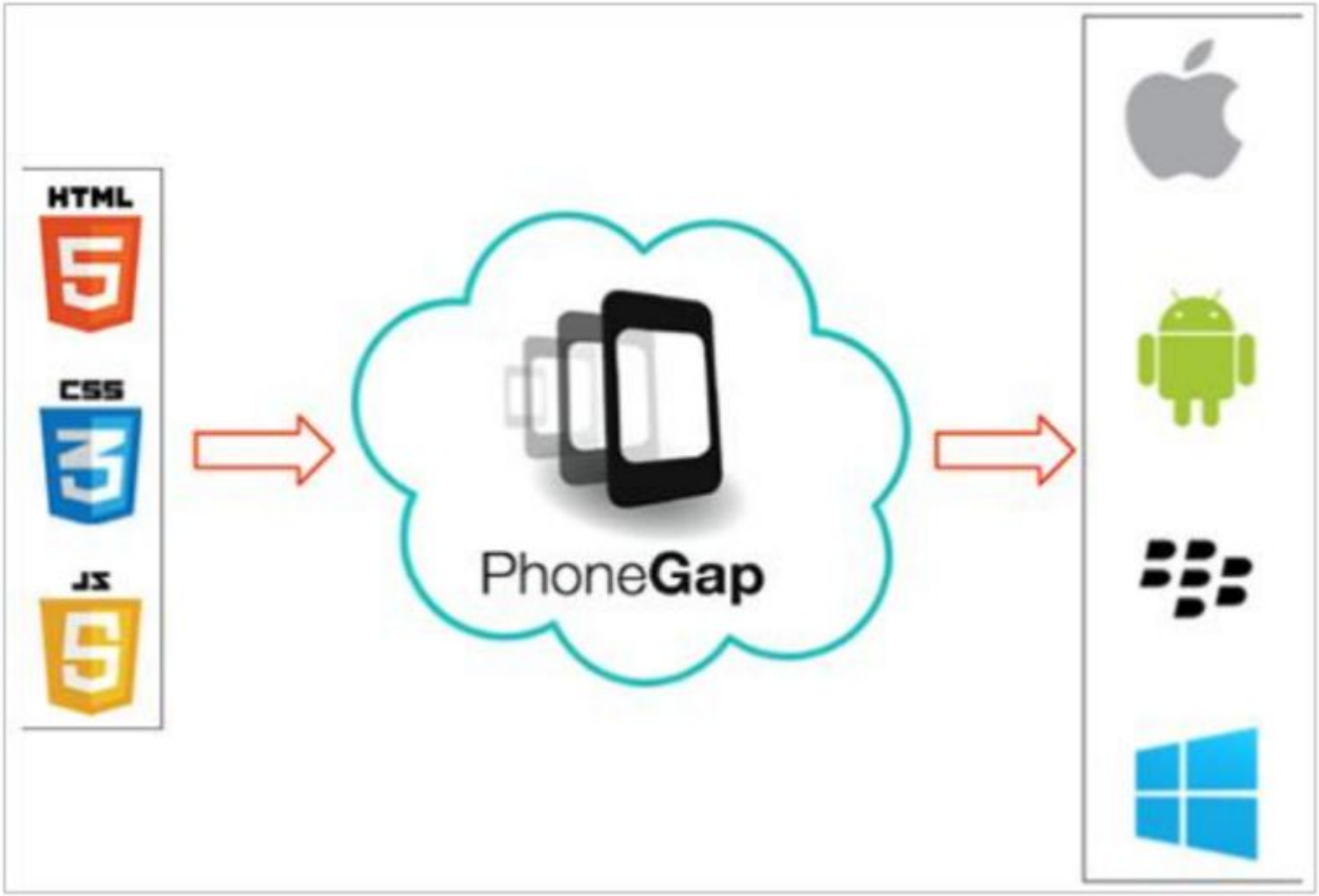
20CSI605 - Mobile Application Development

PhoneGap

PhoneGap

- **PhoneGap** is a framework that makes the developers develop their apps using standard web APIs for all major mobile operating systems. It is open-source and free. i.e cross platform mobile application development
- Developers only need to know web development using HTML, CSS and JavaScript.
- PhoneGap takes care of rest of the work, such as look and feel of the app and portability among various mobile operating systems.

PhoneGap



PhoneGap

- Using PhoneGap, one can create apps for all major mobile operating systems like Apple iOS, Android, BlackBerry, Windows etc.
- This does not require the developer to have an expertise over any of the above mentioned platforms, neither the developer is required to know programming to code the app from scratch.
- PhoneGap allows its users to upload the data contents on website and it automatically converts it to various App files.

PhoneGap

List of Operating Systems, Software, and Programming Languages Required by Different Mobile Platforms

Mobile OS	*Operating System*	*Software/IDEs*	*Programming Language*
iOS	Mac only	Xcode	Objective C
Android	Windows/Mac/Linux	Eclipse/Java/Android Developer Tools (ADT)	Java
BlackBerry	Windows	Eclipse/Java Development Environment (JDE)	Java
Symbian	Windows/Mac/Linux	Carbide	C++
WebOS	Windows/Mac/Linux	Eclipse/WebOS plug-in	HTML/ JavaScript/C++
Windows Phone 8	Windows Visual Studio 2010	Silverlight or Windows Presentation Foundation (WPF)	C#

PhoneGap

Why developers should use PhoneGap?

- It is very easy to start with multiple platforms based on single code Rapid testing and the development of apps. It is a very easy irrespective of the platform. It maintains uniformity, and here is no need to hire expert developers.
- Apps that are developed by using PhoneGap behave like native apps.
- It is open-source software.
- It is powered by a powerful source like Apache Cordova, which is now a part of Adobe.
- It is easy to work with the Plugin.
- The PhoneGap is cost-effective. If you have a low budget and want to get more benefits, then PhoneGap is the right tool for you.

PhoneGap

Hardware specific features supported by the PhoneGap API include:

- Geolocation
- Vibration
- Accelerometer
- Sound

PhoneGap

Requirements:

- In order to create applications with PhoneGap, you will need to first install the standard SDK for the mobile platforms you want to target for your app.

So, if you are developing for Android, you will need:

- ✔ Android NDK
- ✔ Android SDK

There are also some additional PhoenGap specific requirements for Android development, including:

- ✔ Eclipse IDE, ADT plugin for Eclipse, Apache Ant, Ruby, Git Bash

PhoneGap

Benefits of PhoneGap

- It reduces development cost
- Easy for web developers
- Each app can behave like native apps
- Provide Robust Backend Support
- Easy to work on Plugin

PhoneGap

Drawbacks of PhoneGap

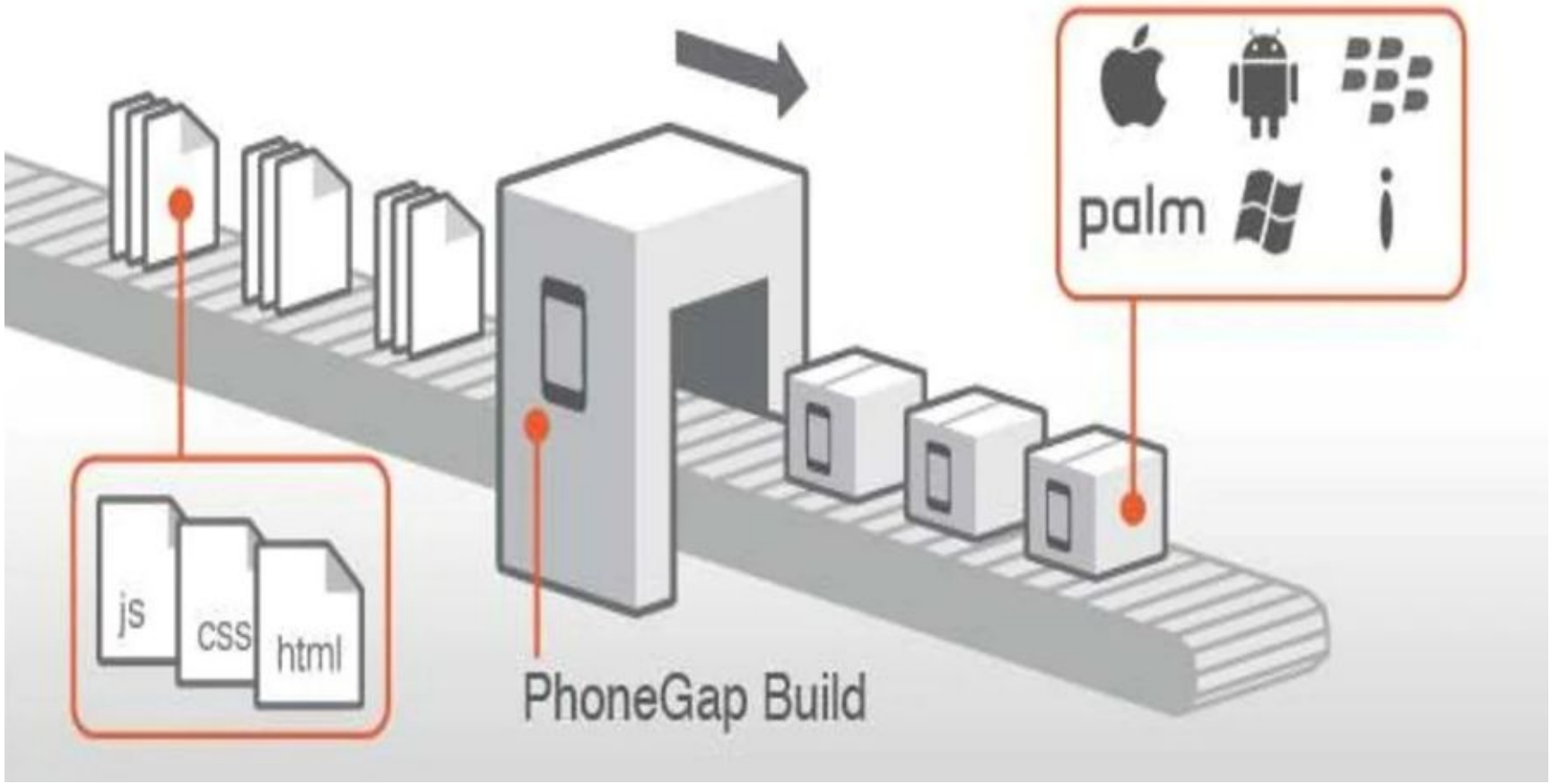
- PhoneGap doesn't support the plugins with Hooks
- PhoneGap doesn't support hardware intensive app
- PhoneGap require Mac to develop ios applications
- Plugins can get outdated

Applications:

- Wikipedia, DHS Program, Solution, Tripcase

https://www.formget.com/install-phonegap/

How phonegap works?



PhoneGap

How it works?

- Install the native SDK of your target mobile platform
- Download the PhoneGap zip file
- Setup a new project in the target platform with PhoneGap parameters
- Write Javascript and HTML mobile application
- Build and deploy your target mobile platform's SDK

Offline and Online Mode

M .Sivakumar, AP/CSE

Offline and Online Mode

- Offline websites are copied to local hard drive and accessed whenever the user needs to without any internet connection.
- Likewise, this offline web app will let you create a web application that is downloaded to its entirety to the mobile devices of a user who can access that offline.
- An application for this type of app may include app having collection of stories, short tutorials or any other offline content of users' interest, which he/she can read offline even when internet is not available.

Offline Mode

- Creating an offline experience is not something that most developers consider when designing an app.
- Since smartphones allow users to access the internet from any location or at any time so long as the app communicates to the server, the need for making the app work offline might seem insignificant.
- However, mobile networks aren't always reliable since the strength of coverage varies from one place to another.
- Therefore, mobile apps should be equipped to handle issues related to network failure and latencies. That's where offline architecture can make a difference.

Offline Mode

- Developers are shifting to the ‘offline-first’ approach whereby an app is built to function even without a network connection, thereby providing users with the opportunity to sync their data, get updates and more.
- Offline architecture also offers advantages such as reducing roaming costs when travelling, minimizing the data usage on maps, saving battery power as well as providing quick loading time.

Offline Mode

The three commonly used approaches used for developing offline app are:

- **Storing data offline:** Caching is the most straightforward approach for helping users access data offline. It involves storing data on a local server or browser from where the device can access it, providing a better user experience in a short time.
- **Edit offline & sync online:** Under this approach, users can edit data offline, marking changes in the object. Once the network connectivity is back, the changes get synced through a push and pull operation.
- **Users can edit other users' data offline:** Here, the data sync happens automatically without manual intervention. The system resolves conflicts on its own.

Offline Mode

Offline-First Best practices to communicate app with the backend server

Pushing data in

- In this strategy, the server sends a signal, usually a push notification, to the user to alert him that new data is available for download.
- It works best in situations where the server is the primary source of data, and the client application is a data consumer.

- **Synchronizing databases** – The server and the app try to track every change to ensure that the data on both sides are equal.
- **Prefetching** – Here, the application tries to predict what the user will use and sends and stores that data in the cache whenever the network connection is active.

Offline Mode

Recommended tools and frameworks for developing offline-first apps

High level Tools

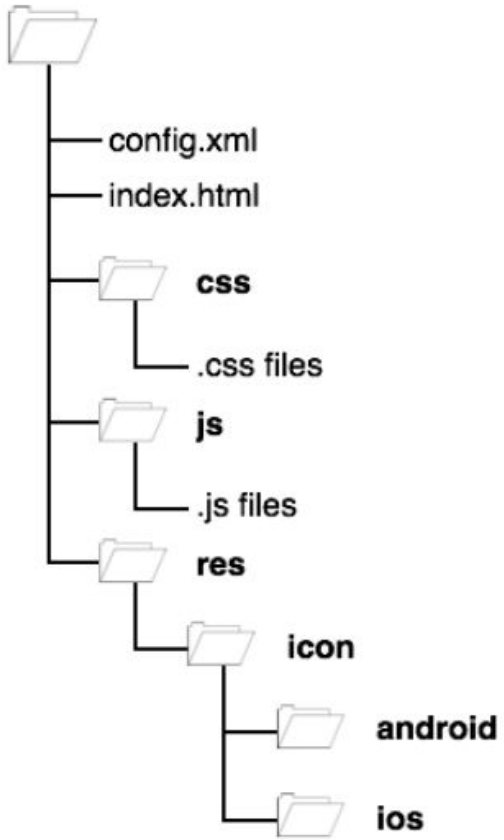
- ✔Progressive web apps
- ✔Polymer AppToolbox
- ✔Couchbase Mobile
- ✔Hoodie
- ✔Ionic
- ✔Mapbox Mobile
- ✔Realm Mobile Platform

Low level Tools

- ✔Service workers
- ✔localForage
- ✔PouchDB
- ✔Cloudant Sync
- ✔Cloudant Envoy

Online Mode

The following image shows folder structure for our app to be in online mode. In online mode, entire web content is loaded from internet website



Online Mode

- You may notice that **data** folder is missing in online mode app, because all the files reside on the actual server and are accessible via internet.
- The index.html file contains actual links as it contains at the web server and all its links are either absolute or used with **base href** tag.
- After you have decided the mode of your app and organized its files in the file structure mentioned above, you need to zip your file with any standard zip tool and save it.

THANK YOU