

Introduction

L'internet dans nos jours est devenu une nécessité pour la plupart des gens. On trouve ainsi des projets qui visent à partager l'internet même avec les pays tiers monde gratuitement.

Et grâce aux bénéfices de l'internet, un grand nombre des entreprises ont changé leurs utilisations des applications du bureau vers les applications web.

Une application web est une application accessible aux utilisateurs à partir d'un navigateur web (ou bien un agent spécifique), en utilisant des requêtes http avec des liens spécifiques qui seront mappés vers des serveurs qui à leur tour traduisent les demande des utilisateurs et les renvoient des pages html qui peuvent être afficher par le navigateur.

L'application web peut être composé de plusieurs couches (selon l'architecture utilisée pour la création de cette application). Un exemple typique est une application composées des trois couches : présentation, métier (business) et data, comme c'est montré dans la Figure 2: Architecture trois tiers.

La couche présentation contient l'interface utilisateur, la couche métier contient la partie logique de l'application, tandis que la couche data s'occupe de la gestion des données et la communication avec les bases des données utilisées.¹

Les utilisateurs de l'internet sont habitués à la notion qu'une application web nécessite avoir une connexion à internet pour qu'elle peu être utilisée.

Cependant, un grand nombre des applications web sont utilisés par les utilisateurs d'une façon régulière, et l'utilisateur de ces applications demande généralement les mêmes ressources chaque fois (images, voix, vidéo, et même des données stockées dans des bases existant chez le serveur).

Et ici plusieurs questions peuvent être demandées :

1. Est-ce-t' il vrai qu'une application web nécessite toujours de l'internet ?
2. Pourquoi ne pas sauvegarder les ressources qui doivent être utilisés fréquemment par le client?

¹ <https://msdn.microsoft.com/en-us/library/ee658099.aspx>

HTML5 Application Cache

Avec html5, on peut aujourd'hui créer une version déconnectée de l'application web en utilisant le cache de l'application « application cache ».

Avantages de l'utilisation du cache web

« Application cache » est le processus de la mise en cache des documents web tel que les médias, les pages web (html, scripts, librairies, ..) ce qui peut être utile pour :

Réduire la consommation de la bande passante, puisque les ressources de la page web seront téléchargé uniquement la première fois qu'on ouvre la page, ensuite le navigateur n'a besoin que de télécharger les changements et les mise à jour des ressources.

Réduire la charge du serveur web, puisque les clients vont travailler sans connexion, donc sans l'intervention du serveur pour chaque événement, donc ces clients ne nécessitent plus de ressources tel que le processeur et la mémoire centrale.

Naviguer sans connections, puisque le client a déjà télécharger toutes les ressources nécessaire pour que l'application (ou bien une partie de cette application) puisse travail, donc il pourrait utilisé cette application même s'il n'a pas une connexion à internet.

Diminuer le temps nécessaire pour la consultation des pages web, puisque le chargement des ressources déjà téléchargées nécessite un temps négligeable en le comparant au temps du téléchargement de ces ressources.

Support navigateur

Le tableau suivant montre les versions depuis les quelles chaque navigateur commence de supporter le cache web.






API					
Application Cache	4.0	10.0	3.5	4.0	11.5

Figure 1: support navigateur²

Pour Google chrome: la version 4.0 est disponible depuis l'année 2010

Pour IE : la version 10.0 est disponible depuis l'année 2012

Pour Mozilla : la version 3.5 est disponible depuis l'année 2009

Pour Safari: la version 4 est disponible depuis l'année 2009

Pour Opéra : la version 11.5 est disponible depuis l'année 2011.

² http://www.w3schools.com/html/html5_app_cache.asp

Donc on peut dire que cette propriété est maintenant supportée par les navigateurs les plus utilisés dans nos jours.

Utilisation de la cache d'application

Le cache d'application repose principalement sur l'utilisation d'un fichier « manifest » et à l'ajouter aux pages HTML du site web.

Pour activer le cache d'application dans une page web, il suffit d'ajouter un attribut « manifest » avec le chemin vers le fichier manifest à la balise <html>, comme c'est montré dans l'exemple suivant :

```
<!DOCTYPE HTML>
<html manifest="demo.appcache">
...
</html>
```

Chaque page contenant l'attribut manifest serait mise en cache quand l'utilisateur la visite.

Note : une recommandation par le standard est d'utiliser l'extension « .appcache » pour le fichier manifest.

Création du fichier manifest

Le fichier manifest est un fichier texte, dans le quel on déclare les ressources qui doivent être mises en cache et celles qui doivent être téléchargées à chaque fois du serveur.

Ce fichier comprend 3 sections dont la première est requise :

Cache Manifest : dans cette section on spécifie les fichiers qui doivent être mis en cache.

Network : dans cette section on spécifie les ressources qui doivent être prises à chaque fois du serveur.

Fallback : dans cette section on choisit une page html qui doit être utilisée par le client en cas où on ne peut pas établir une connexion à internet.

Note : on peut utiliser « # » pour mettre une ligne en commentaire

Exemple du fichier manifest

CACHE MANIFEST

#2012-02-21 v1.0.0

#ici on place les ressources qui doivent être mises en cache

/theme.css

/logo.gif

/main.js

NETWORK

#ici on place les ressources et les pages qui nécessitent une connexion à internet

login.asp

#on peut toujours mettre une étoile « * » pour dire « toutes les autres ressources doivent être mises en cache »

*

FALLBACK

/html/ /offline.html

Note : il faut faire attention de ce qu'on va mettre en cache, car lorsqu'un fichier est mis en cache, le navigateur continue à utiliser la version mise en cache du fichier, même si ce fichier change chez le serveur.

Mise à jour du cache

Il existe 3 méthodes pour changer ce qui est mis en cache par le navigateur :

1. L'utilisateur efface la cache du navigateur
 - Par exemple en Google Chrome (pour les utilisateurs de Windows) : si l'utilisateur tape sur son clavier « ctrl » + « alt » + « del », puis il clique sur le bouton « clear browsing data » après avoir choisi « cached images and file » de la liste des options.
2. Le fichier manifest change et **non pas les ressources mises en cache**
 - Autrement dit si on a mis en cache une image nommée logo.gif, puis on a remplacé notre logo par un autre, et on n'a pas changé le fichier manifest, le client de cette application ne va pas avoir le nouveau logo, mais il va utiliser celle qui est cachée dans son navigateur.
3. Le cache application est mis à jour par le code.
 - **Note :** les commentaires utilisés dans le fichier manifest peuvent avoir un autre objectif que de créer un code bien organisé et compréhensible. **Le changement de la date et de la version dans une ligne de commentaire**

est une autre méthode pour demander au navigateur de remettre à jour les fichiers mis en cache.

Figures

Web Application

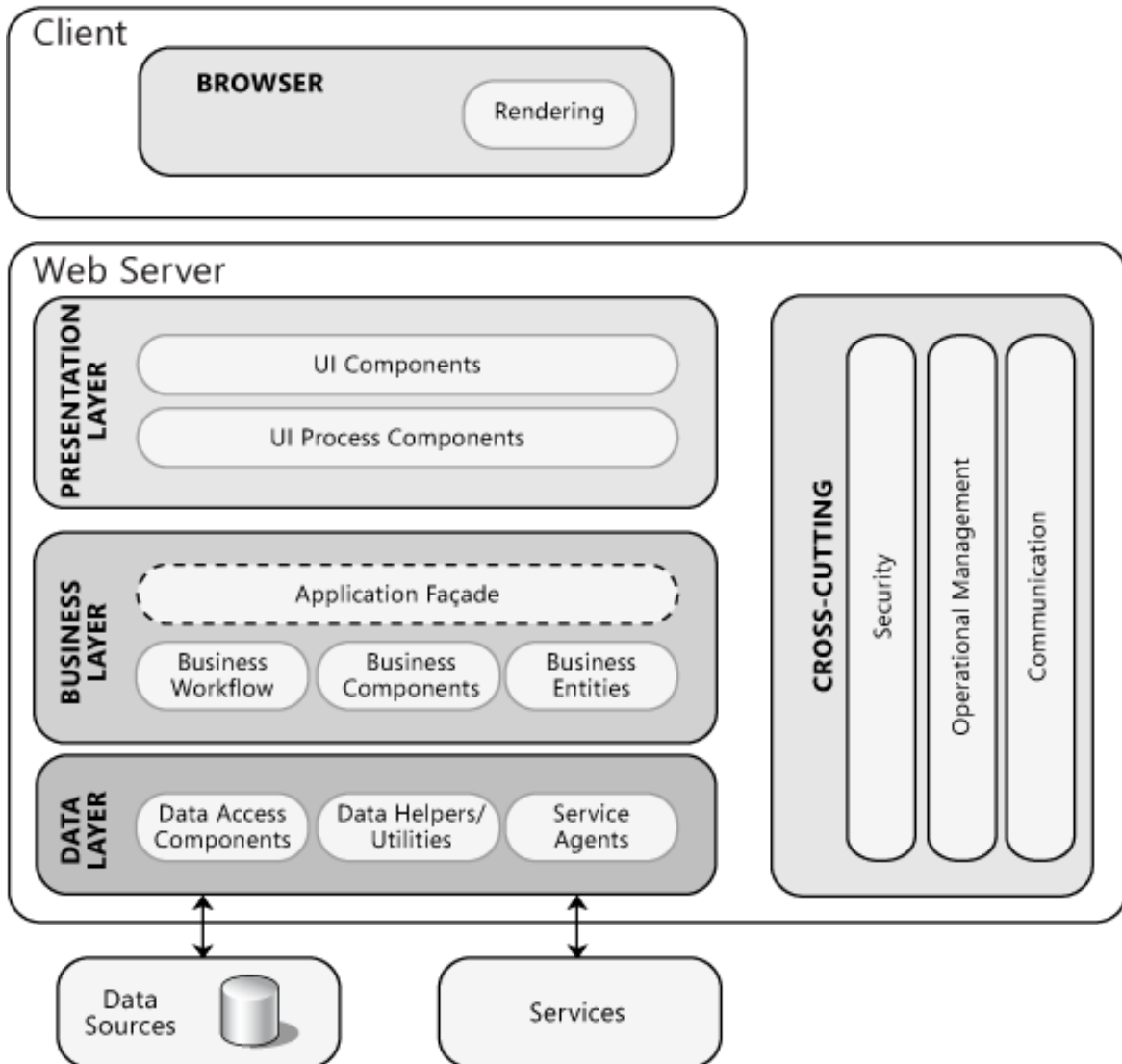


Figure 2: Architecture trois tiers³

³ <https://msdn.microsoft.com/en-us/library/ee658099.aspx>