All instructions videos are in a server, which will be downloaded from the first run of each App. (Android or iOS) Also in each run, the App checks if there are updates for the videos; if so, the videos are downloaded again.

There are two phases. In the first phase, the App gets all information on the videos; `MovieInfo` class represents the information, which in turn is got from `get_movie_info.php` in a server. If videos are new or updated, the second phase downloads all necessary videos. All is done in `DownloadMovie` class.[1] Here is a typical usage.

```
dispatch_async(dispatch_get_main_queue(), ^{
NSURL *getMovieInfoUrl = [NSURL
URLWithString:@"http://ownphones.com/unity/get_movie_info.php"];
DownloadMovie *downloadMovie = [[DownloadMovie alloc] init];
if (downloadMovie) {
    BOOL b = [downloadMovie
        updateMovies:folderHierarchy.playMovieDirURL
        from:getMovieInfoUrl
        onProgressUpdate:^(int progress) {
            ALog(@"DownloadMovieProgressUpdateBlock:onProgressUpdate:");
            ALog(@"progress = %d %%", progress);
        }
        onEnd:^(NSArray *movies, NSArray *statusCodes) {
            ALog(@"DownloadMovieEndBlock:onEnd:");
            int n = (int) [movies count];
            for (int i = 0; i < n; ++i) {
                MovieInfo *mi = movies[i];
                NSNumber *ni = statusCodes[i];
                ALog(@"|%@|: downloaded? %d", mi.name, [ni intValue]);
            }
            [OwnUtil lsDir:[folderHierarchy.playMovieDirURL path]];
        }
    ];
    if (!b) {
        ALog(@"ERR: updateMovies:|%@| from:|%@|: failed",
folderHierarchy.playMovieDirURL, getMovieInfoUrl);
    }
} else {
    ALog(@"FATAL: [[DownloadMovie alloc] init] failed");
}
});
```

DowloadMovie class uses, internally, NSNRLConnection class, which should be initiated in a main thread, which is why all is wrapped with `dispatch_async(dispatch_get_main_queue(), ^{..})`.

---

[1] This class and internal `NetGet` class are connected to two PHP scripts in a server: `get_movie_info.php`, `net_get.php` with common `dev_filter.phpi`.

There should be a hierarchy in folder, which contains videos. This hierarchy is managed by `FolderHierarchy` class. Here is a typical usage.

```
FolderHierarchy *folderHierarchy = [
    [FolderHierarchy alloc]
    initWithDirType:NSLibraryDirectory withBaseName:@"OwnPhones"
];
ALog(@"INFO: rootDirURL: |%@|", folderHierarchy.rootDirURL);
ALog(@"INFO: baseDirURL: |%@|", folderHierarchy.baseDirURL);
ALog(@"INFO: playMovieDirURL: |%@|", folderHierarchy.playMovieDirURL);
ALog(@"INFO: backupMovieDirURL: |%@|", folderHierarchy.backupMovieDirURL);
```