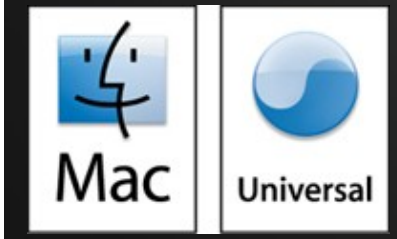
iContinuousIntegration

Oleksandr Dodatko

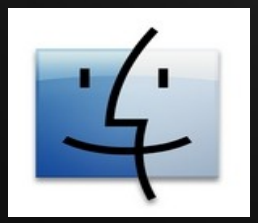
What's Covered

Managing shared projects with xCode

Building a project without xCode GUI



Creating “universal binary” libraries



Deploying project and libraries for QA

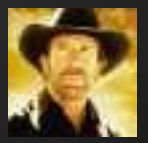
<epam>

More Fun for Developers

Unit testing with GHUnit

Using Hudson build server

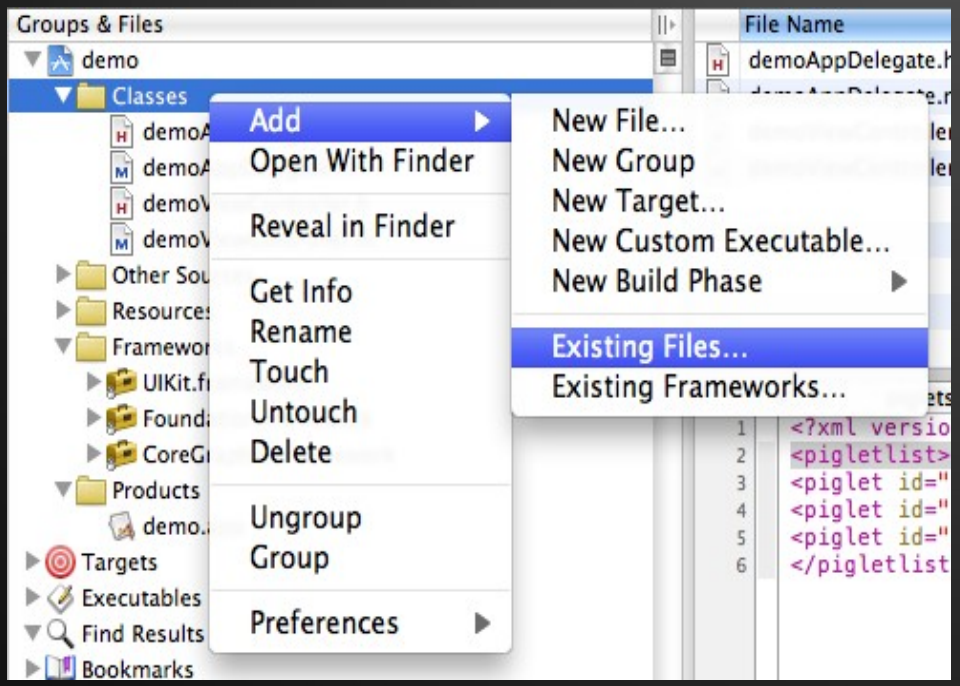
(it has Chuck Norris plug-in)

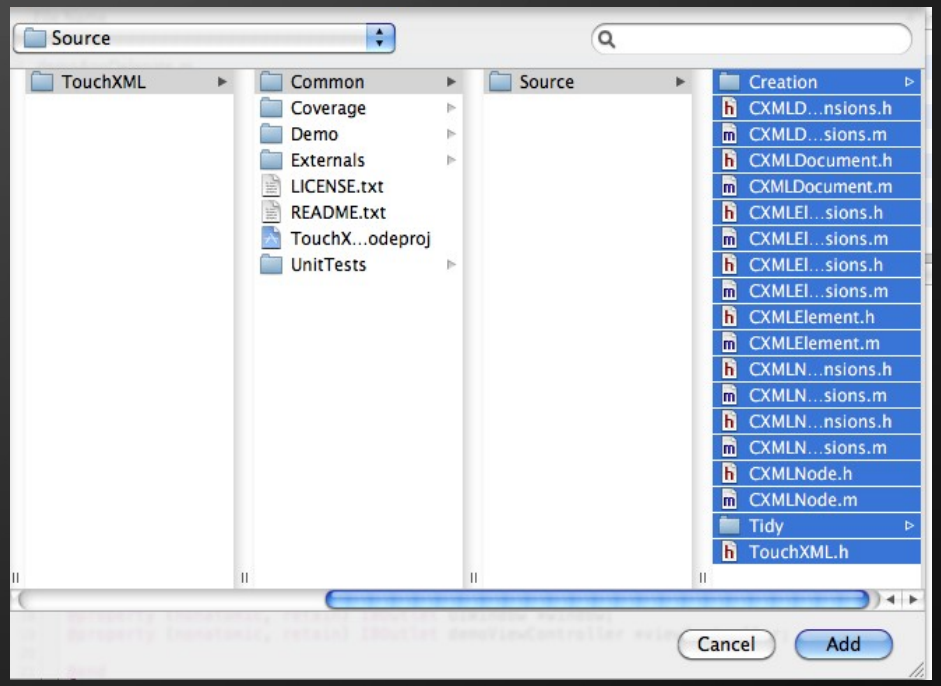


Running applications on simulator without xCode

<epam>

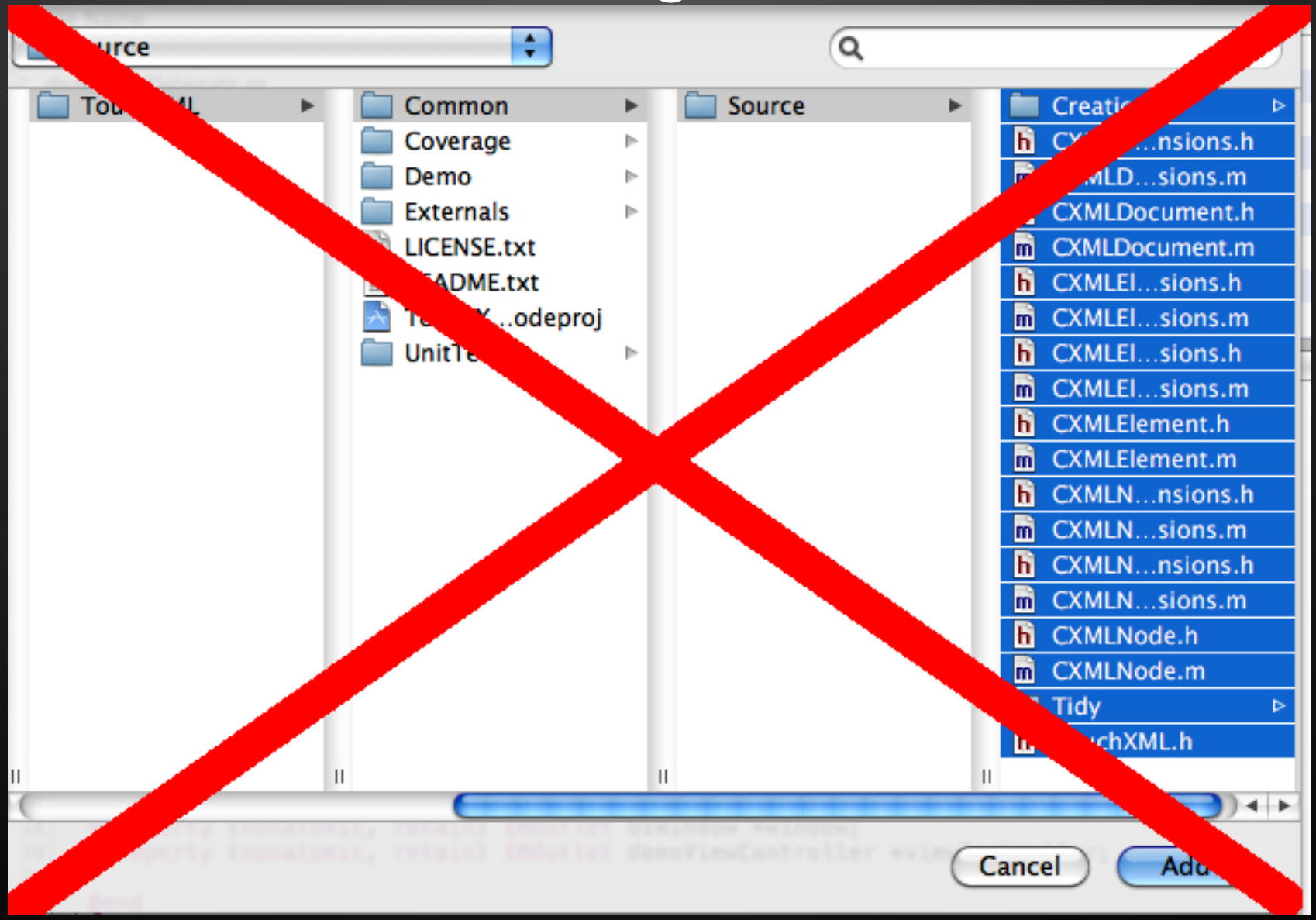
“Commonly Used” Project Organization





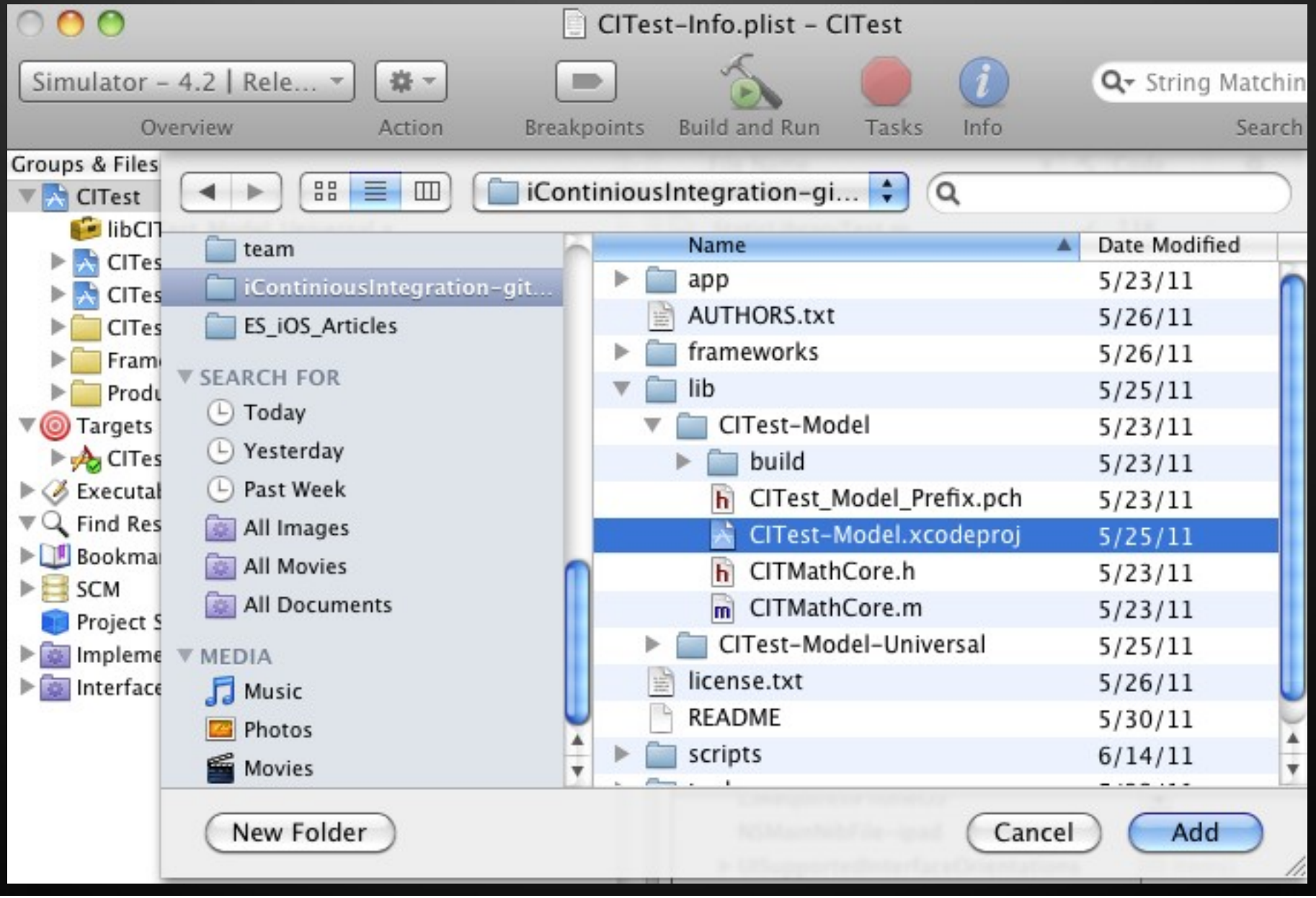
<epam>

Wrong !!!

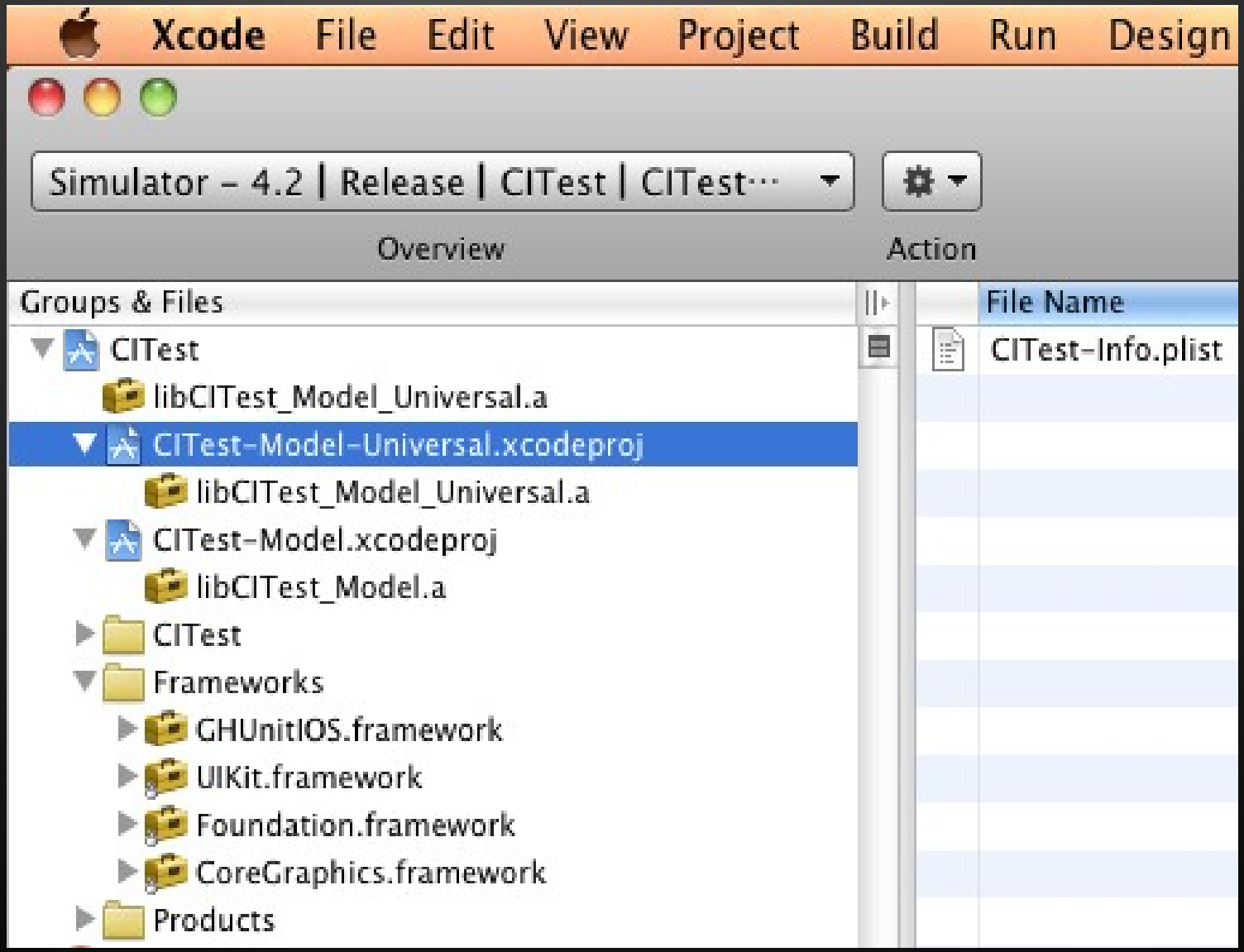

Common
Coverage
Demo
Externals
LICENSE.txt
Source
CXMLDocument.h
CXMLDocument.m
CXMLElement.h
CXMLElement.m
CXMLNode.h
CXMLNode.m
Tidy
Cancel

ONE Product,
ONE XCODE PROJECT

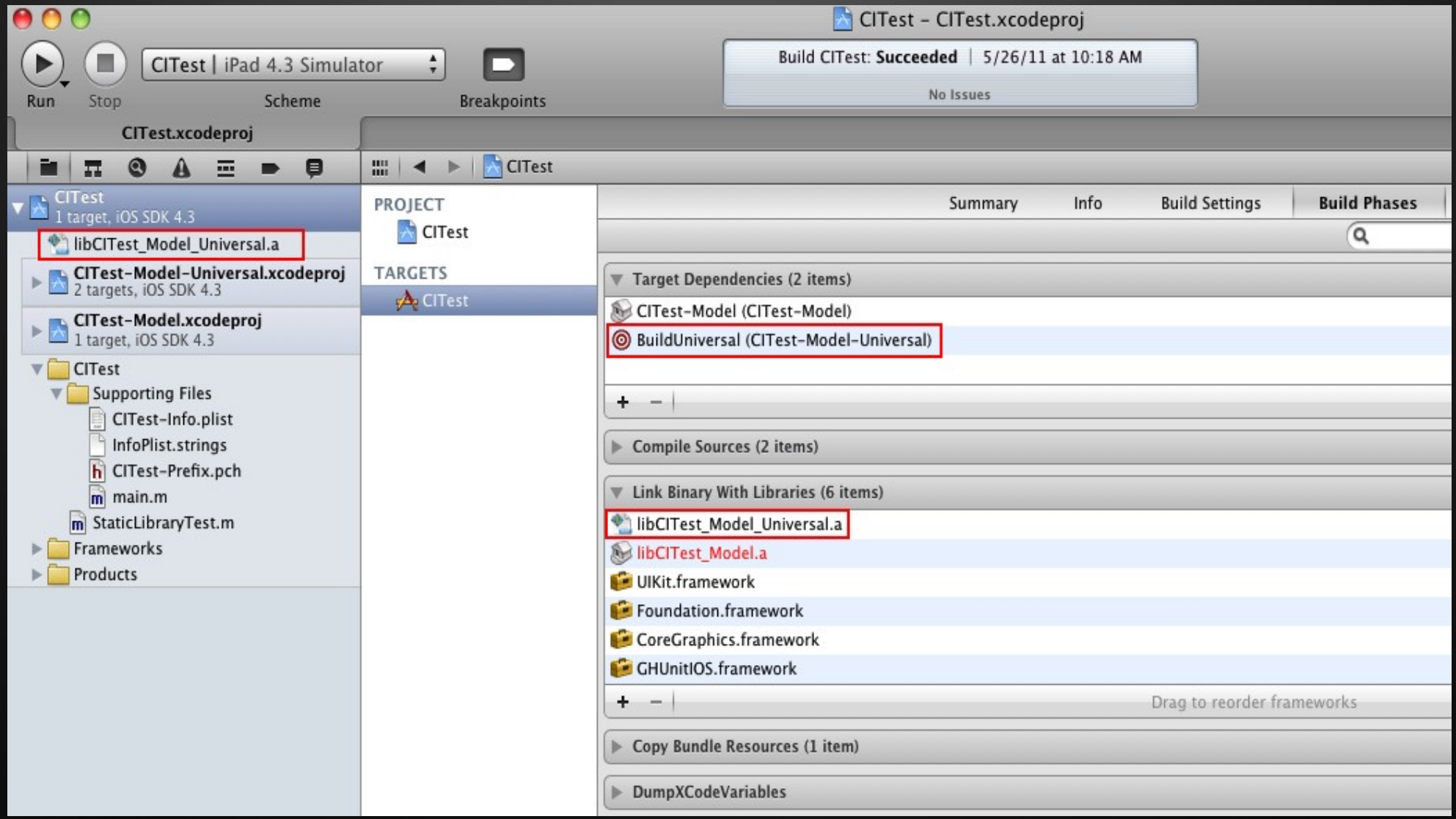
Library Project How-To



Library Project How-To

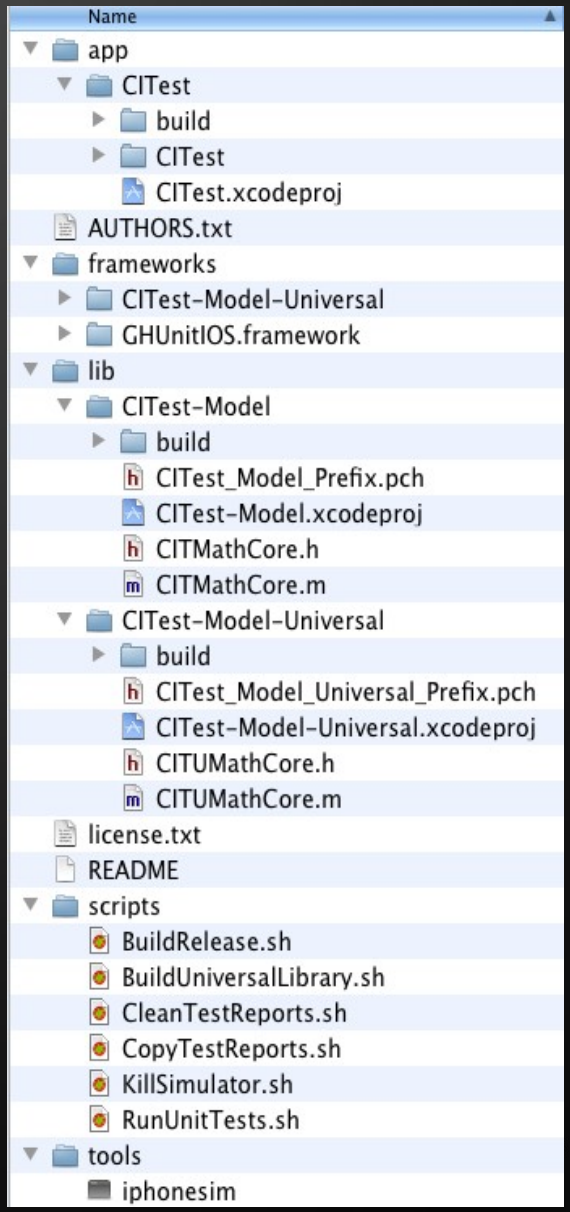


Setting up dependencies



Defining the Project Structure

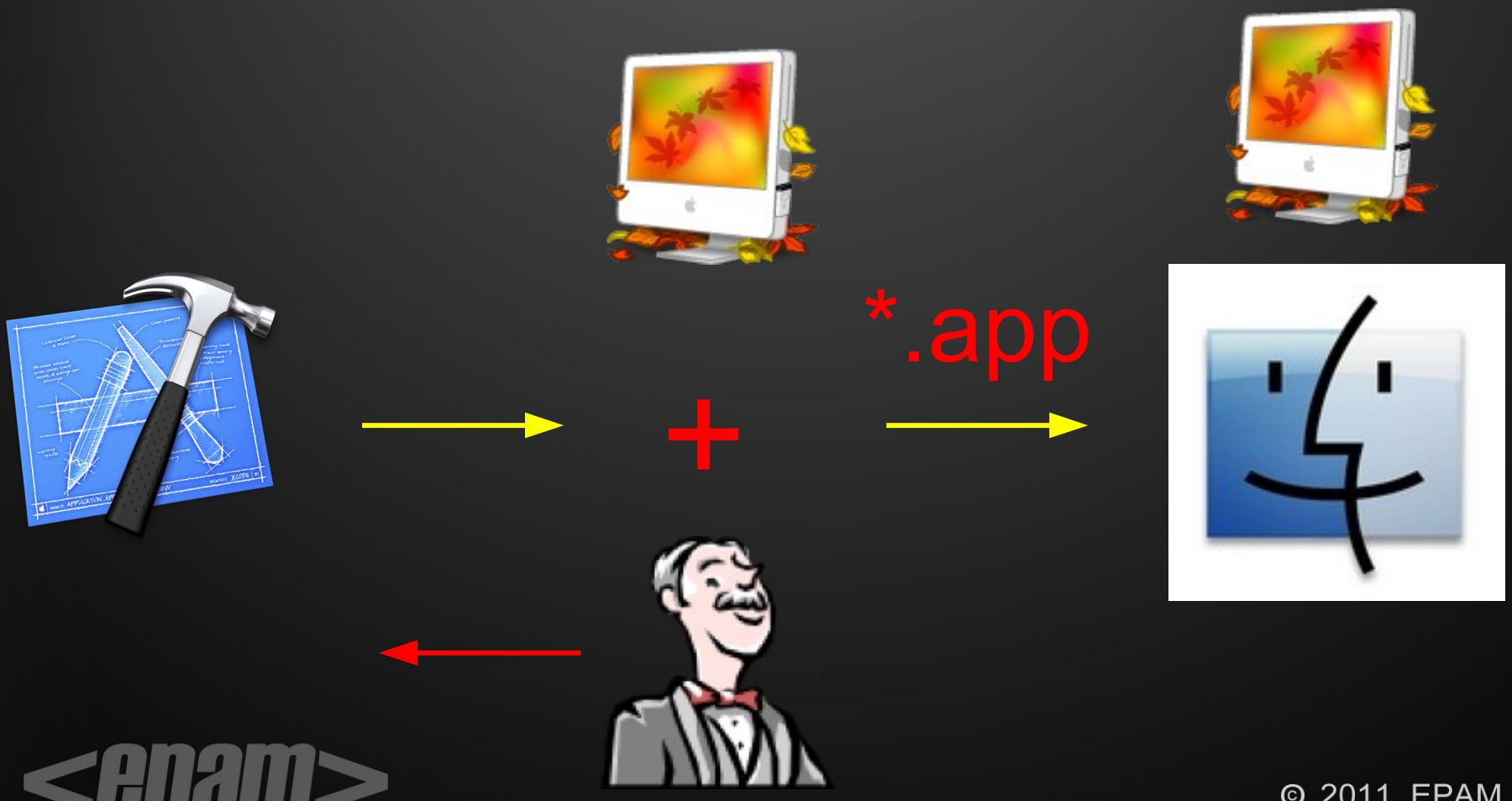
app
lib
frameworks
scripts
tools
test
certificates
deployment



<epam>

Deployment : Desktop vs. Mobile

Desktop Applications

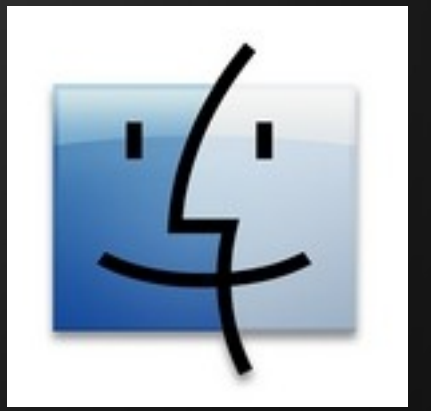


iOS Applications

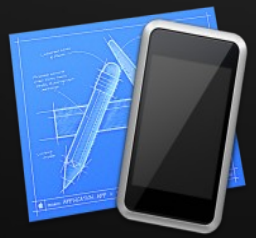
QA

?

*.ipa



?



Mobile QA

*.ipa

<epam>

Building Without xCode GUI

xcodebuild -project CITest.xcodeproj

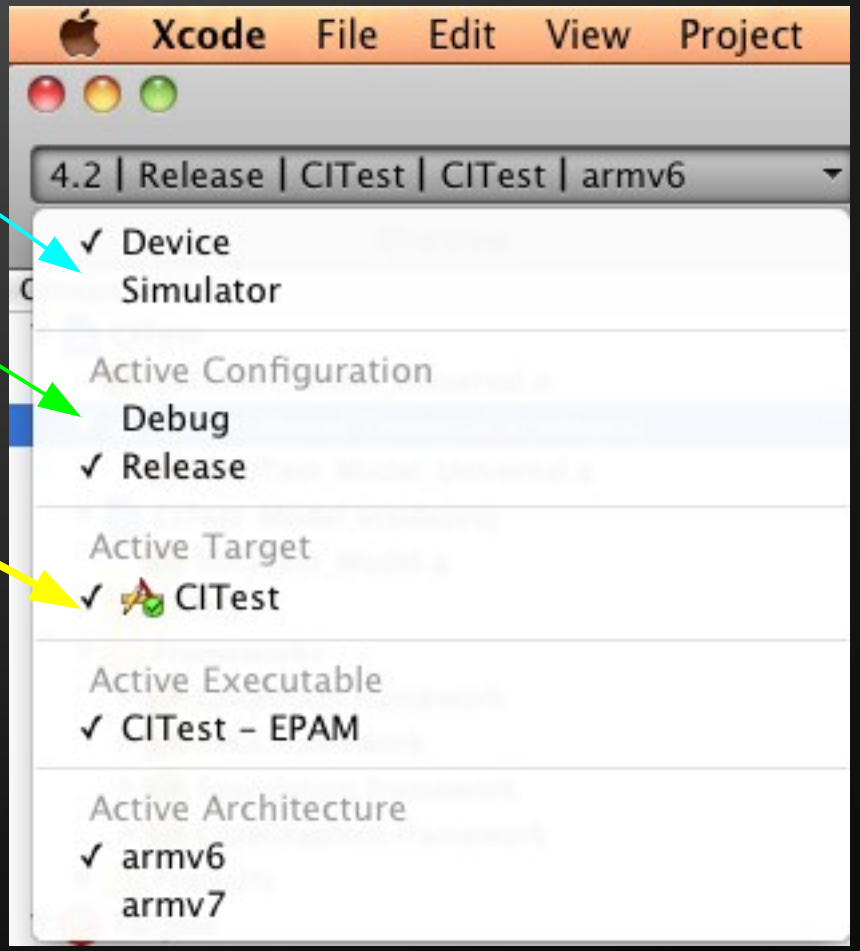
-sdk iphonesimulator4.3

-configuration Release

-target CITest

-parallelizeTargets

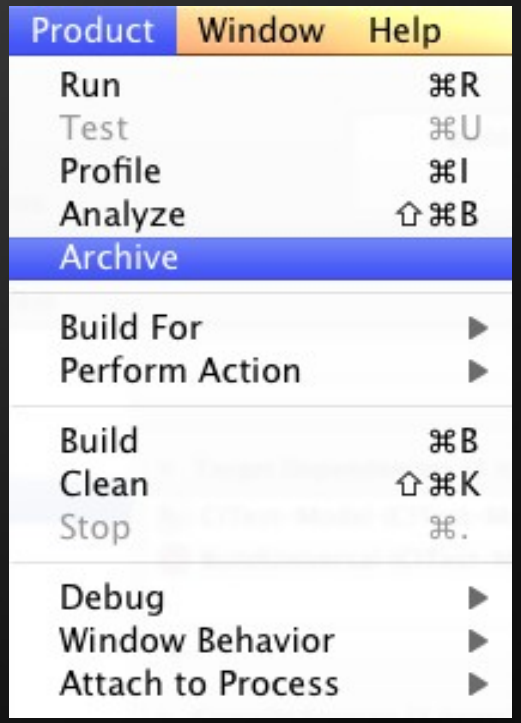
clean build



<epam>

Creating Installable *.ipa File

```
/usr/bin/xcrun -sdk iphoneos PackageApplication
-v "${BUILD_DIR}/Release-iphoneos/CITest.app"
-o "${DEPLOYMENT_DIR}/CITest.ipa"
--sign "${DEVELOPER_NAME}"
--embed "${PROVISONING_PROFILE}"
```



DEVELOPER_NAME="iPhone Developer: Oleksandr Dodatko (ABCDEFG123456)"

How About Unit Testing?

Picking a framework

Running a test

Collecting results

Test Frameworks Chart

	SenTest	Google	GHUnit
Xcode integration	+	+	---
UIKit Support	---	---	+
Bundles support	---	---	+
Xml reports	---	---	+ (lack of support for hudson CI)
Runs on device	+- (Runtime tests only)	+- (Runtime tests only)	+
Runs on simulator	+- (logic tests only)	+- (logic tests only)	+
Debugging (out of box)	---	---	+
UI snapshots comparing	---	+	---

GHUnit Configuration

Add GHUnit.framework

Replace Main.h with the one from GHUnit

Remove “MainNibFile” entry from the info.plits

```
setenv( "GHUNIT_AUTORUN" , "YES", 1 );
setenv( "WRITE_JUNIT_XML", "YES", 1 );

// Not supported in the official GHUNIT
setenv( "GHUNIT_AUTOEXIT" , "YES", 1 );
```

Running a Test

```
iphonesim launch
 "$DEPLOYMENT_DIR/CITest.app"
 4.2
 ipad
```

NOTE : Use only FULL PATH to the app
as shown above

Collecting Test Results

TEMP_DIR=$(/usr/bin/getconf DARWIN_USER_TEMP_DIR)

All Test results are here :

$TEMP_DIR/test-results

Terminating the Simulator

killall -s -KILL -c "iphonesim"

killall -KILL -c "iphonesim"

killall -s -KILL -c "iPhone Simulator"

killall -KILL -c "iPhone Simulator"

Do it before you run a test app

<epam>

Creating Universal Binaries

1. Build a library version for the device.
2. Build a library version for the simulator.
3. Combine them to a single binary
4. Deploy universal library to the “frameworks” directory.

<epam>

Combining Binaries

```
lipo -create
"${LIB_BUILD_DIR}/Release-iphoneos/libCITest_Model_Universal.a"
"${LIB_BUILD_DIR}/Release-iphonesimulator/libCITest_Model_Universal.a"
-output "../frameworks/CITest-Model-Universal/Lib/libCITest_Model_Universal.a"
```

<epam>

Build Script Should

Build main products

Create *.ipa packages for main products

Build unit tests

Run unit tests with iphonesim

Package *.ipa and *.app entries to *.zip archive

Prepare unit test reports for deployment

<epam>

Build server should

- Checkout project sources
- Run build script
- Deploy product archives
- Publish test reports

Contacts

EPAM systems (Dnipropetrovsk) http://www.epam.com/

Github page : **https://github.com/EmbeddedSources**

https://github.com/EmbeddedSources/iOS-articles

Oleksandr Dodatko

mail/jabber : dodikk88.reg@gmail.com

Skype : alexander.dodatko.work@skype.com

Github page : **https://github.com/dodikk**