Agenda

1. Data Alternatives
2. The File System
3. Working with Relational Data

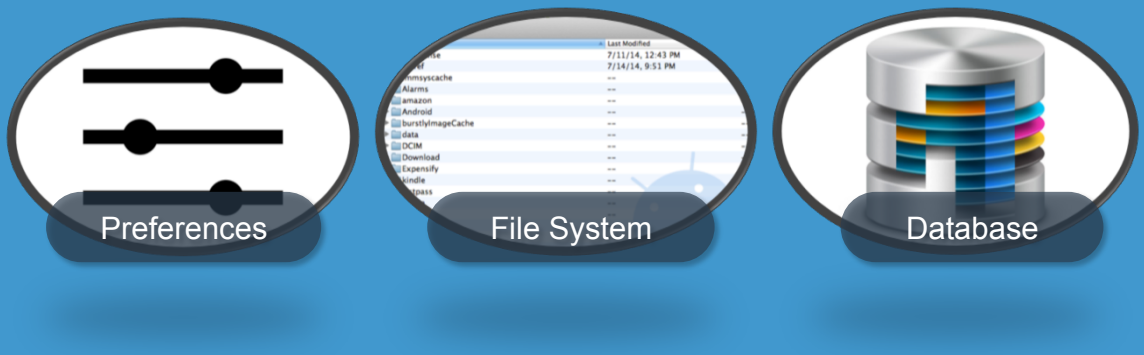
Xamarin University

Data Alternatives

Xamarin University

Data Storage Options

When storing local information, your app has several options to choose from

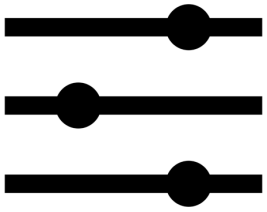


Xamarin University

Preferences

iOS, Android and WP8 support storage of app-specific settings as simple key-value pairs

Useful to store app configuration, user preferences, and other customization tweaks



github.com/jamesmontemagno/Xam.PCL.Plugins

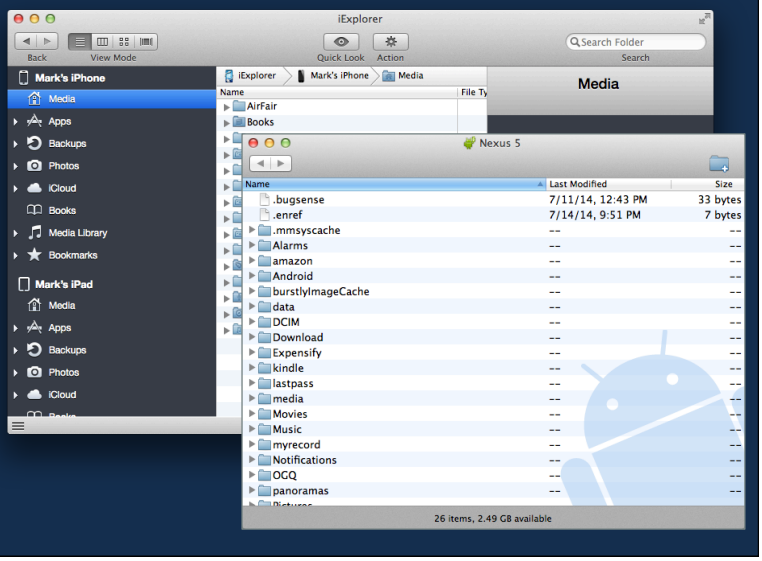
File System

Xamarin University

The file system

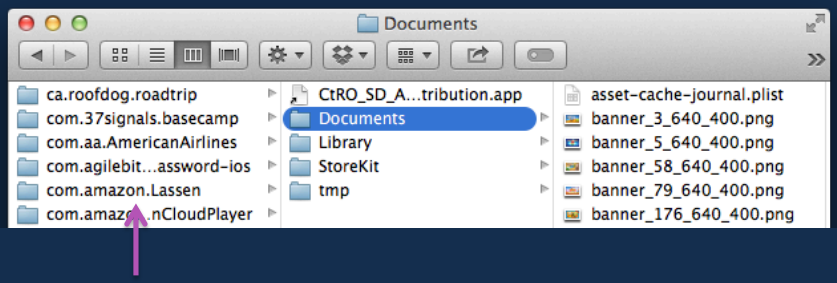
Your phone has a persistent file system to store settings, applications, data and other necessary pieces of information

The file system and it's contents vary based on the operating system



The app sandbox

Your application is given a dedicated folder, called the app folder or sandbox, on the file system which contains app-specific content



Each iOS application has a folder, which contains sub-folders, which contains your data and assets

Xamarin University

Working with Files and Folders

Can work directly with files and folders using classes in `System.IO` namespace

```
public IEnumerable<Task> LoadTasks(string filename)
{
    StreamReader reader = File.OpenText(filename);
    ...
}
```

Xamarin University

File Locations

Location you place writable files ("Documents") is platform-specific

These locations are common, but other options are available
(e.g. Android has a **database** folder, iOS has a **cache** folder, etc.)

File Locations

API to retrieve *default documents folder* is also platform-specific

```
string docFolder = Environment.GetFolderPath(
                Environment.SpecialFolder.MyDocuments);
```

```
string docFolder = ApplicationData.Current
                            .LocalFolder.Path
```

Xamarin University

Xamarin University

The code that determines the location of the file and the code to read and write the file will always be in your platform-specific projects in Xamarin.Forms

Demo

Xamarin University

Agenda

1. ~~Data Alternatives~~
2. ~~The File System~~
3. Working with Relational Data

Xamarin University

SQLite

Xamarin University

What is SQLite?

SQLite is an open-source, lightweight database that has become the de-facto industry standard for mobile apps

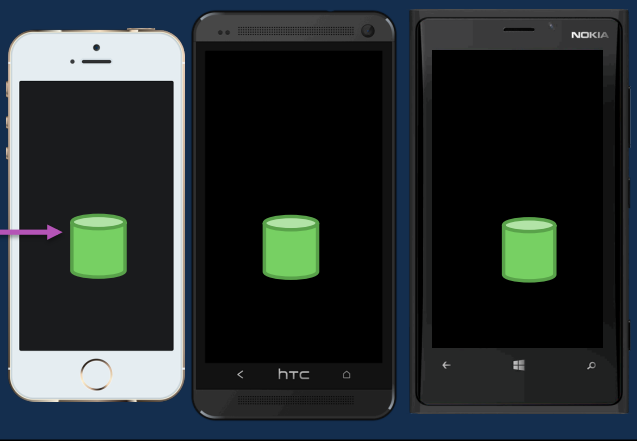
Xamarin University

Xamarin University

Local execution

SQLite provides a local database for a single mobile application

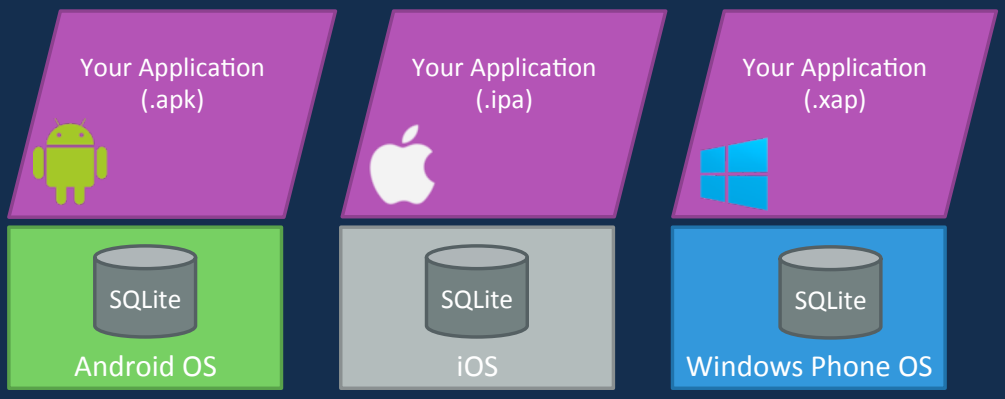
runs in-process (no server) and uses local file system for storage



Xamarin University

Packaging

- SQLite is built-in to Android and iOS, but *not* Windows Phone



Xamarin University

Install SQLite for Windows Phone

- Windows Phone requires you to install the database engine on your dev machine once

Accessing SQLite data

There are multiple C# APIs available from different vendors, most work with Xamarin so you can choose based on features or coding style

Microsoft .net ADO.NET

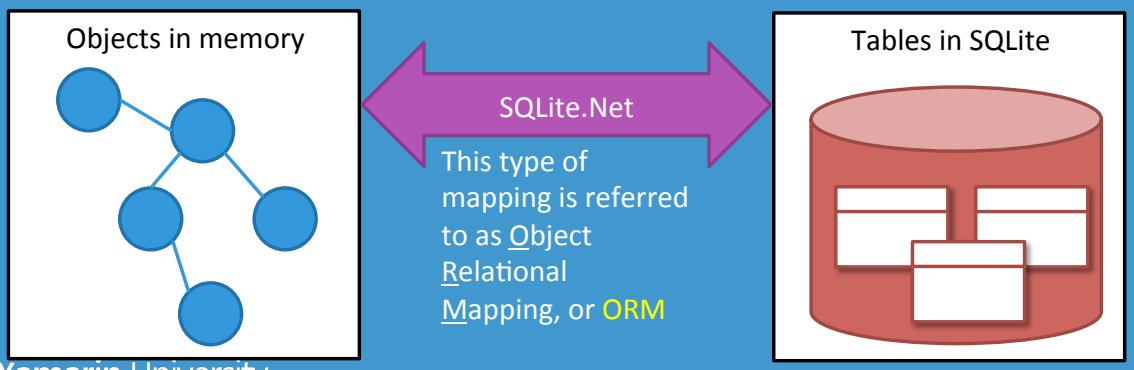
Microsoft Open Technologies
PCL for SQLite
sqlitepcl.codeplex.com

SQLite-net Object Relational Mapper

Xamarin University

SQLite.Net PCL

SQLite.Net provides a mechanism to map classes to tables



Xamarin University

Xamarin University

Mapping classes to tables

- Database schema is defined through attributes

```
[Table("checks")]
public class Check
{
   [PrimaryKey, AutoIncrement]
   public int Id { get; set; };

   public int CheckNumber { get; set; }
   ...
}
```

SQLite.Net uses the attributes to identify which table the class represents and how to map columns to properties

Xamarin University

Working with a database

Typical database steps involved in working with the database

① Create a SQLite connection

② Create the database tables (*optional*)

③ Execute queries and CRUD statements

Demo

Xamarin University

Agenda

1. Data Alternatives
2. The File System
3. Working with Relational Data

Xamarin University

Questions?

Xamarin University

Xamarin Evolve 2014

Working with Data in Mobile Apps

Mark Smith
mark.smith@xamarin.com

Xamarin University