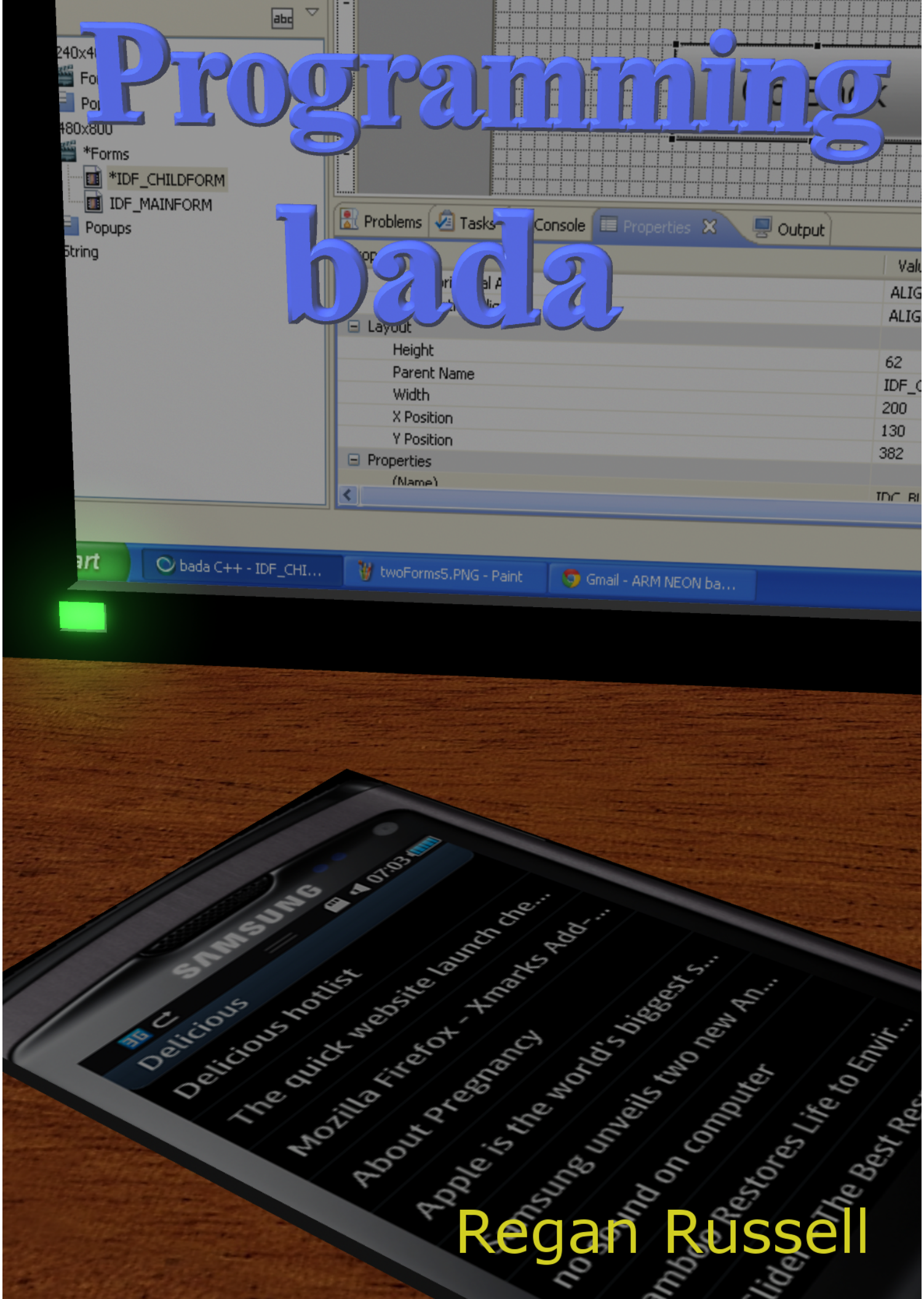
Programming bada



Regan Russell

Contents

A complete version of this book is available from pymblesoftware.com/book

This book is a guide to how to develop real world applications on the Samsung Wave and Samsung bada based mobile phones. Here you will find the examples you need to adapt your C++ or mobile device skills to this new platform from Samsung.

1.1 Who am I?

I have been a professional software developer in C since 1988. I was a bada product specialist with Samsung in Samsungs Sydney Olympic Park offices at the time of the official launch of bada. I offer consoulting services and maintain a web presense at: www.pymblesoftware.com

1.2 Intended audience

It is assumed that you are a capable developer with at least a couple of years experience, possibly with some mobile app experience and comfortable with general programming constructs such as Object-Orientated Programming, pointers, Web Services, and asynchronous messaging. You should know a little about XML as the application manifest and application meta data relies on it.

C++ knowledge is strongly recommend before you tackle this book but programmers from a Objective-C, C, or Java background should be able to adapt themselves to C++ and the APIs covered in this book.

1.3 Why I wrote this book

The initial book on bada was not much more exciting than a cut and paste from the online help. As a Samsung employee around the time of the launch, I was tasked with the mission of teaching iPhone and Android developers the ways of bada and to write apps when where the need arose.

I had high hopes for the initial bada book before it was released and was expecting a book more tutorial and practial in nature. The initial book was also very rah-rah and not very objective of where bada has issues. At times I was forced to work around badas limitations and the only resources suffering from a dose of marketing spin didn't seem to help.

This book is the book I wanted to teach bada from. I was delighted to be approached by a publisher, although I had never heard of them, commisioned me to write the book that I wanted. After 180 pages of text, graphics and computer programs, and 6 rewrites on some chapters, the publisher dropped out of the project for their own reasons. This book is almost complete rewrite from start to finish, since then. I hope you find this book useful.

Regan Russell,
Sydney, Australia
March 17, 2012

A complete version of this book is available from pymblesoftware.com/book

7.3 Using Panels

Imagine that you run out of space on a form, for whatever reason you cant create another form, so you start recreating all your controls on a scroll panel Lets see what happens. Create a new app. Go to the UI Builder and drop a Scroll panel on to the main form. Double click on the scroll panel to open it. Drop a button control onto the scroll panel. Go to the main form source file and go to the OnInitializing() method. Add the following code to the OnInitialiszing method:

```
using namespace Osp::UI::Controls;

Button *myButton = static_cast<Button *> GetControl( L"
    IDC_BUTTON_OK" );
myButton->SetActionEventListener( *this );
```

Run the app. Does it crash at start up?
Change the GetControl( ) line to read
GetControl(L"IDC_BUTTON_OK" , true);
Run the app again, does it crash this time?

Ah ha! Gotcha! For some reason in their wisdom, the designers of bada decided that the 2nd parameter of the API function GetControl should be a default parameter and that default effectively returns NULL if your control is on the child of the form, not the form itself. And you would never move a control off a form and on to a panel would you? Will you remember to always set this second parameter in case you move controls from a form onto a forms panel? Or at the very least check the result of GetControl before trying to dereference it.

7.3.1 Tab UI controls

Tabs are useful for creating panes that can be switched between easily. Most apps that have a wizard like structure can be condensed down to a few tabs.

We are going to create a tab control 1. Create a new app.
2. Open the UI builder with the main form by double clicking the main form. Go to the properties section of the form and select "Show Tab" option drop down and click on the text tab drop-down option.
3. Open the source file behind the main form.
4. In the OnInitializing() method of the MainForm add the following code:

```
Tab *pTab;
pTab = GetTab();

if( pTab )
{
  pTab->AddItem( L"First", 1 );
  pTab->AddItem( L"Second", 2 );
```

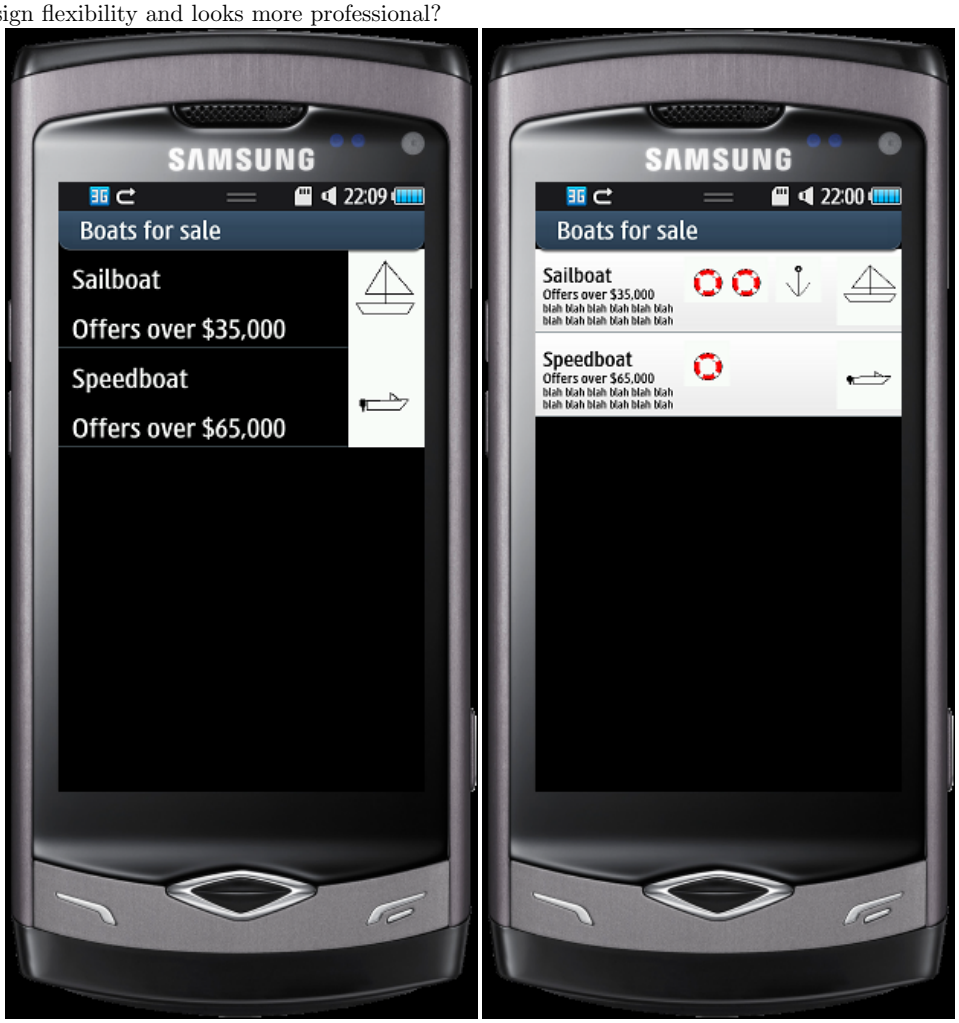
A complete version of this book is available from pymblesoftware.com/book

```
            blah blah blah blah blah blah blah blah blah
            blah blah" )) );

        pMainList->AddItem( *pPRItem2, 2);
    }

}
```

The two versions of the app should like this, which one gives you greater design flexibility and looks more professional?



We have created some images and decoded image files into bitmaps. We

A complete version of this book is available from pymblesoftware.com/book

created a custom list format and added elements to it. Some elements are text of different sizes and other elements are images.

7.8 Custom list items

It may take a bit to understand but here is what to think I need to create a custom list so I will create a format that lays out images and text in different sizes, styles and locations that looks good. Then I will need to iterate of the dataset from my data model and set images and text elements in a newly constructed Custom list item. Finally I will remember to add this item to the custom list. And that is all there is to it.

7.9 Image file formats

When I first came across bada I expected image file format stuff to be pretty simple. I expected something like Bitmap = LoadImageFromFile() .But no, it was not to be, image file formats have to be decoded into a bitmap using different bitmap pixel formats for each type of file. The decode method is a member of an image object.

7.9.1 A story that happened to me and stressed me out at the time

Be very careful about other peoples data. As the FBI guy on TV who chases space aliens said, "Trust no-one". I say "Trust their data even less". I was writing something that extracted data from someone elses web service and presented it in a custom list. That data contained URLs for images. So based on the URL the code I wrote would find a GIF, PNG or JPG and send some BITMAP_PIXEL_FORMAT parameters to DecodeURL( ) and then using a custom list control show the images along side the data pulled out of the web service. After more days debugging and tearing my hair out than I would like to admit, I found that almost by chance just because file is a .jpg file means that it could be a png or gif or even jpg file. And the bada GetBitmapN(..) type APIs require that you know the type of format in advance, and if you cannot trust the file extension, then you have to look for other means. The web service team let me in on seemingly unrelated fields and said something along the lines of frustrating hackers or stopping people reusing images on their site. Well it definitely frustrated me at the time.

7.9.2 Pop quiz Image file formats

1. What is the deal with bitmap pixel formats and their relation to decoded images?

2. What the 3 stages of creating a custom list?

A complete version of this book is available from pymblesoftware.com/book

9.1 Network programming

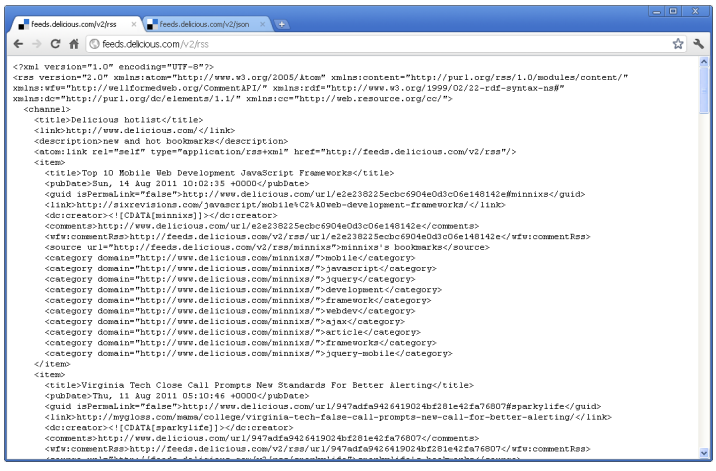
There is no doubt that we live in a connected world and demand for connectivity extends to when we are on the go. In this chapter we shall discuss: Web services o REST XML JSON example code Socket programming So lets get on with it...

9.1.1 Prior network programming knowledge

A lot of the network programming on bada is much like network programming in any environment. One thing I strongly suggest is to arm yourself with knowledge of TCP/IP and tools like Ethereal/Wireshark (www.wireshark.org) and/or Fiddler (www.fiddler2.com).

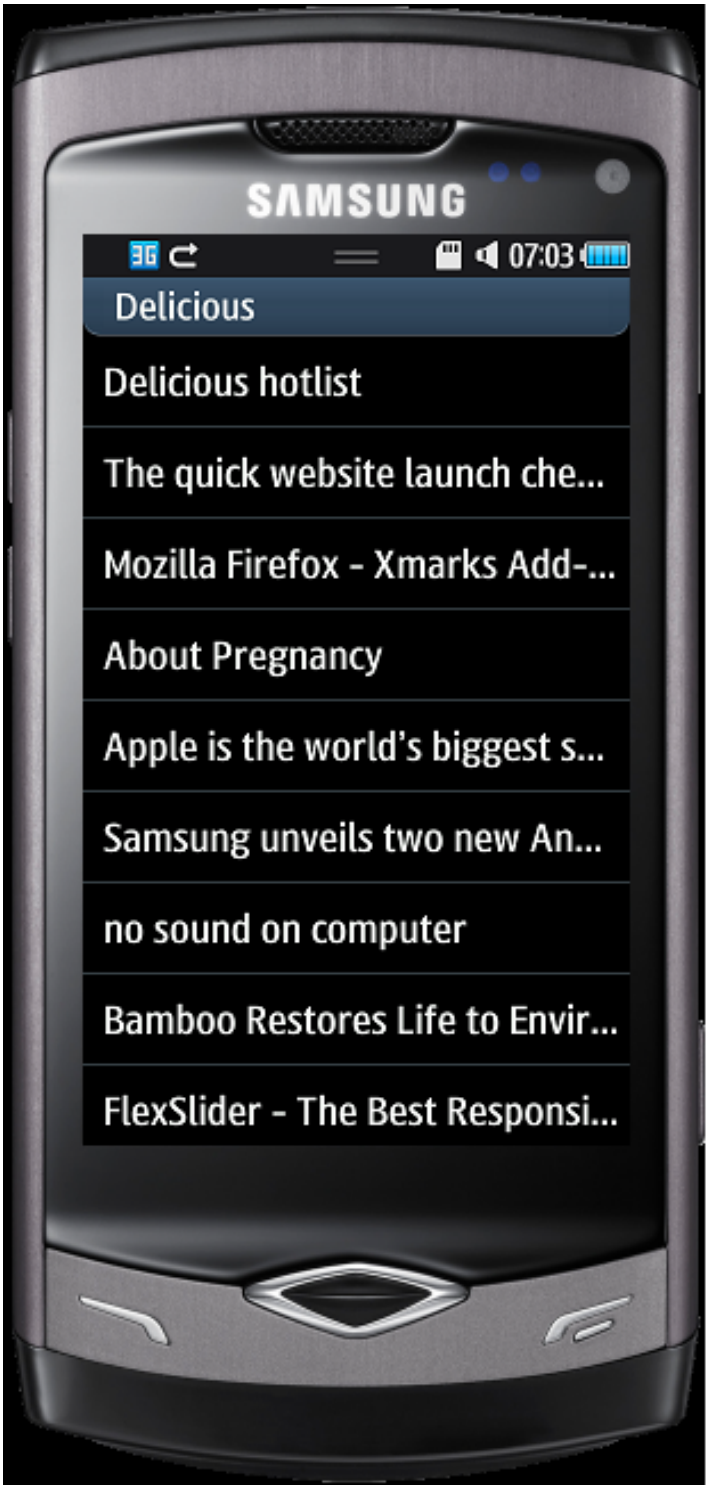
9.2 Accessing www.delicious.com web services

A lot of my initial work on bada came from web services. A lot of people are publishing RESTful web services and there are entire industries developing access to them on different devices and platforms. Accessing www.delicious.com web services Imagine that you are implementing a web service client for someone or something, maybe local public transport time tables for your city or region. Maybe a major company or sporting team want apps that customers and fans will use. Open a web browser and type the following address into the tabs: 1. http://feeds.delicious.com/v2/rss 2. http://feeds.delicious.com/v2/json You should get data back as seen in the following two screen shots:



and

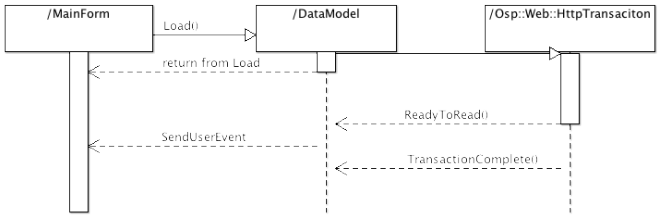
A complete version of this book is available from pymblesoftware.com/book


SAMSUNG
3G
07:03
Delicious
Delicious hotlist
The quick website launch che...
Mozilla Firefox - Xmarks Add-...
About Pregnancy
Apple is the world's biggest s...
Samsung unveils two new An...
no sound on computer
Bamboo Restores Life to Envir...
FlexSlider - The Best Responsi...

A complete version of this book is available from pymblesoftware.com/book

```
0043.980,DEBUG,P29,T03,A94,MainForm::
    OnUserEventReceivedN (107) > (How to Upload a File
    in C#.NET) (H)
0044.098,DEBUG,P29,T03,A94,DataModel::
    OnTransactionCompleted (130) >
    OnTransactionCompleted
0072.780,INFO,P29,T-01,A94,OspMain (39) > Application
    finished.
```

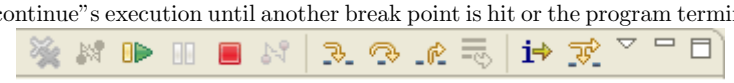
We verified that an ordinary web browser could access the feeds from the web service we wanted or needed to use. This is important place to start because the server side guys point the finger at the client side developers and client side developers point the finger at the server side guys. Verifying it in a browser establishes the basic service exists. We have used the bada wrapper around a web kit based Http classes to implement a simple bookmark reader. We broke the project up into a model view controller (MVC) structure. The app itself could be considered the controller in the pattern and form the view and the data model is obviously the model. We have built on the work of the previous chapters. By saving the pointer the form that initiated the load request we are able to send a user event to it as covered in the chapter on UI programming. Also from the chapter on UI programming we have the skills and tools to make a simple (Custom)List. To see the general pattern of http transactions, consider the following sequence diagram.



The thing to take away from the diagram is the asynchronous nature of the HttpTransaction. The flow of execution has returned from Load as soon as the HttpTransaction is set up. It comes back via the ReadyToRead() and SendUserEvent() when the load has completed. This is a pattern in dealing with web service request apps in bada. The app (controller) creates (Main)forms (views) and models (data models). The form calls load on the data model and passes a pointer to itself for later notification, the data model sets up a Http-Transaction. The transaction asynchronously goes of through the network stack out onto the web and returns execution flow back to the data model through calls to the interface methods of ReadyToRead(), TransactionCompleted() and error handler methods. When there is data is read, the model sends a user event. The main form calling the load method is not a correct implementation of the MVC design but it is handy just to pass a "this" pointer to the load method. The load could just as easily (more correctly) be done from the controller (app)

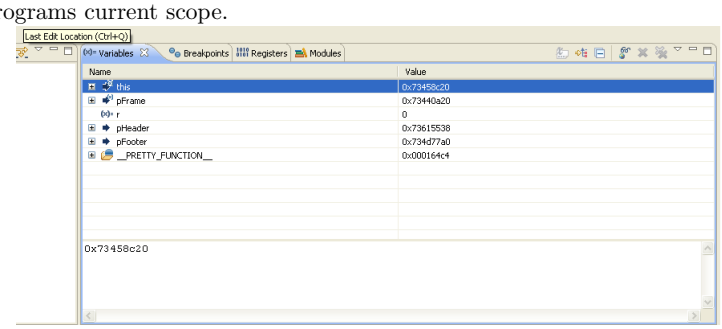
A complete version of this book is available from pymblesoftware.com/book

buttons. F5 (arrow falls between the two dots icon) is for "Step into", if the instruction where the debugger is stopped is a class method or ordinary function, the debugger will switch to the source file containing the code that is called and step into it. F6 (arrow over the dot icon) is for "Stepping over" which just moves to the next C++ statement F7 (arrow up and out icon) "Steps out" of the current method or function. It executes all the code up to the return or exit point of the function or method. F8 (yellow bar, green arrow head icon) "continue"s execution until another break point is hit or the program terminates.



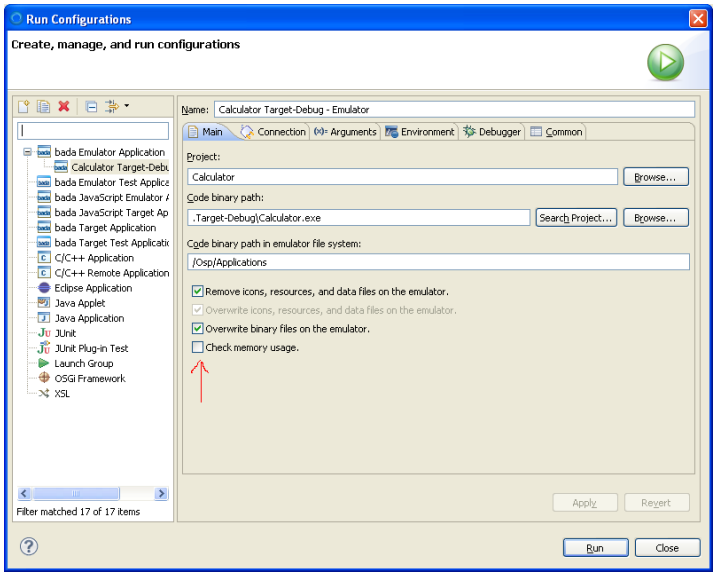
Watching and examining memory

In the default layout, the top right pane of the Debug perspective contains variables, breakpoints, registers, and modules tabs. The variable tab contains names and contents that are relevant to the current context of the program. Global, local and member variables will be shown here when they are in the programs current scope.



If you break point inside a method of a class. The main object you will be interested in most of the time is the "this" pointer. Take some code that initializes some member values in a constructor or OnInitializing() method. Watch as the assignments of newly allocated objects is assigned to pointers in the class as you step through the code. In the bottom pane there is a memory tab. Expressions can be put into this frame and the memory contents evaluated. This is helpful in looking at arrays and data structures where offsets into memory locations are calculated. Using the output pane The output pane is always visible in the bada perspective. In the Debug perspective you can also enable this view from the menu item Window, choose other and then in the bada folder, choose Output. Anything you put in the code with AppLogDebug(.) messages will be display in the output pane. Be aware of one thing, AppLogDebug, AppLog and friends are pre-processor macros. There is no type checking and no compile time warnings about anything you pass to AppLogDebug(.) . I forgot to put -¿GetPointer() at the end of a bada String pointer in an AppLogDebug() message at the point where code was crashing for some other reason. It wasnt funny at the time, and I am ashamed to admit that Ive done it more than once.

A complete version of this book is available from pymblesoftware.com/book



In the code at some random point, say in a constructor or OnInitializing() method add some code to leak some memory:

String *Fred = new String(); String *Bob = new String(); You should see in the problems pane at the bottom of the IDE window showing how many bytes where leaked by which object.

We looked at the memory leak tool and learnt how to track down memory leaks.

11.5.2 Problems with the memory leak tool

The memory leak tool doesnt track all memory allocation. There are times when memory is consumed indirect of your codes requests. This is not always tracked or was not always tracked in some SDK releases. The Samsung bada SDK team in Korea denied the existence of the issue of the memory leak tool not picking up leaks in the bada APIs. However, the memory leak detection has been improving over the many recent bada releases. You need to remove all leaks from you own code and then check memory usage and use other bada APIs or refractor your code to restrict the number of calls to any APIs you suspect are leaking.

11.5.3 Pop quiz using the memory leak tool

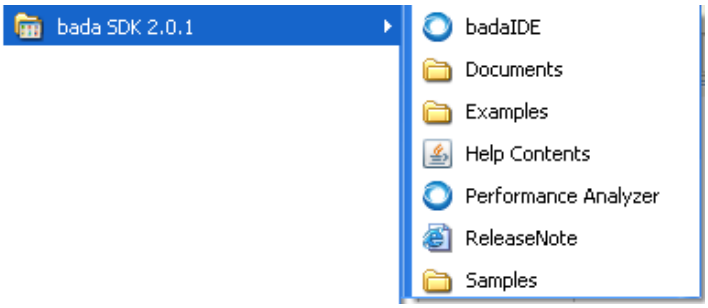
1. What would take the term "Heap exhaustion" to mean?

Answer:

A complete version of this book is available from pymblesoftware.com/book

small amount of time, or it turns out that you need more accuracy than a lookup table provides. Assembler is not a panacea for all performance ills and dont get lost counting cycles if it degrades the effectiveness of the app. Optimization is good nay, important, thing to do after the app is complete and you still have time in the project to squeeze a few extra frames per second (or similar) out of the app.

From the start menu, under the "bada SDK x.x.x" group select "Performance Analyser"



Run an app to launch the emulator and then shutdown the app on the emulator and stop debugging in the bada IDE, but leave the emulator running.

Click on the New button that looks like blank document with a yellow plus in the top right corner.

You will see a configuration screen like:

A complete version of this book is available from pymblesoftware.com/book

the project, ignore that it no longer builds, change the value back to "-c but do not build it.

In the Windows file explorer, navigate to the root of the project then to

```
".Target-Debug\src"
```

. Open the the .o file with a Text editor gVim/ Edit/Notepad/etc. If you have done the previous step of changing the miscellaneous option to S correctly the file should contain an assembly language representation of the corresponding C++ source file. Change the file extension of all the .o files to .s and with the "c in the "Miscellaneous flags make sure the project builds again.

Open a cmd.exe from Start/run and cd to

```
C:\bada\1.2.1\Tools\ToolChains\ARM\bin
```

. Do a directory listing. Note what you see.

If you have been paying attention from the beginning of the book, youll remember that the project manifest files are XML, now we see that the forms themselves are XML files.

13.3.1 XML fun

All of the resource files for your projects are stored as XML. You can edit any of your project files with an xml fiel extension in a text editor like notepad in the Windows Accessory group, or the bada IDE in text mode as we saw in the previous pages. I worked with a bada developer who would not open the bada SDK UI Builder for love or money not when he could do what he wanted in a text editor any old time. "See all the controls are lined up perfectly, see the start position values and all the widths are exactly the same number, so there" he might say and none of the graphic designers could argue.

13.3.2 GCC

The bada compiler is GNUs gcc. Full documentation on language extensions and other topics is available at http://www.gnu.org/software/gcc/ .. Notice that your directory listing included programs such as arm-samsung-nucleuseabi-g++.exe .. At the core of the iPhone, bada and Android phones is the Samsung Cortex licensed from ARM. Note core RTOS of Samsungs bada and Android offerings is Nucleus (http://www.mentor.com/embedded-software/nucleus/) .. bada sits on top of SHP (which as an API internal to Samsung has been around for something like a decade or more) SHP is kind of like a portability layer from TVs to washing machines, internal to Samsung and SHP sits directly on (in the case of the current bada phone offerings) Nucleus. From the documentation you should see the ARM specific options include:

```
-mapcs-frame -mno-apcs-frame
-mabi=name
-mapcs-stack-check -mno-apcs-stack-check
```

A complete version of this book is available from pymblesoftware.com/book

specify. You can set it in the property in the bada UI Builder or you can create it the section specifying a control in the XML file representing the form. I like to keep things grouped in a project in some consistent way. All my forms might be numbered in hundreds 100, 200, 300 and the controls in the forms might be units within each form: 101, 102, 201, and 301, belonging to forms 100, 100, 200 and 300 respectively. The Apple SDK is getting easier to use in terms of UI builder and like but more recent versions of Xcode have begun to resemble iTunes and that may not necessarily be an improvement in terms of usability for infrequent users of iTunes. The bada UI builder does not contain controls as slick and glossy as the iOS SDK and some developers from an iOS background spent a lot of time desperately trying to recreate the iPhone look and feel by sub-classing controls and overriding OnDraw() methods. It was amazing (and sometimes frustrating) to watch a project change from 70% application logic to get the main task done to 70% UI fiddling and 30% actual providing application logic for what the user wanted to achieve with the app.

14.2.2 FObject vs NSObject

All items that end up in bada specific container classes have to be derived from FObject and have comparator sort methods and remember that they must always be created on the heap in bada or bad things will happen in your code when containers clean up. Other than that FObject and NSObject can be considered the root of all evil, uh, I mean the root of all classes in each platform.

14.2.3 Two stage initialization

The two-stage initialization construction of bada objects is generally not as difficult to grasp for iOS developers. This is probably from the two-phase construction of Objective-C objects, in the form:

```
MyClass *myPtr = [ [ MyClass alloc] init ];
```

The bada equivalent would be:

```
MyClass *myPtr = new MyClass();
myPtr->Construct();
```

14.2.4 Application structure

Mobile apps go through an execution cycle: creation, suspension (phone call or user suspends it), resume from suspension and shutdown. Here is a comparison table of significant event methods in iPhone and bada:

iOS	**bada**
UIApplicationDelegate - applicationDidFinishLaunching:	myApp::OnAppInitializing()
UIApplicationDelegate - applicationWillResignActive:	myApp::OnBackground()
UIApplicationDelegate - applicationWillEnterForeground:	myApp::OnForeground()
UIApplicationDelegate - applicationWillTerminate:	myApp::OnAppTerminating()

A complete version of this book is available from pymblesoftware.com/book

Index