



Presentation webinar for **RB** RAD Basic

Kickstarter campaign: <http://bit.ly/radbasic>



HELLO!

I am Carles Royan

Lead developer of RAD Basic

Software Engineer for about 15 years

Worked in compilers in university

Did some work with ANTLR and hibernate (HQL parser).

You can find me at @radbasic



Outline

- ▶ **Why?**
- ▶ **RAD Basic components**
- ▶ **Architecture**
- ▶ **Main goal: compatibility**
- ▶ **Current status**
- ▶ **Live demo!**
- ▶ **AMA (Ask Me Anything)**



► Why? How it started?

- ▶ In 1990s, I learned to code in GW-BASIC, QBasic and Visual Basic
- ▶ In 2002, angry for VB6 abandoned by Microsoft
- ▶ In 2003 created my first compiler (transpiler from pseudocode to java).
- ▶ In 2008-2016 VB6 compatible compiler came up as a good project, but lack of confidence for do it.

Why? Time line

- ▶ Beginning 2018: made some research and found none project/product compatible with VB6.
- ▶ Mid 2018: start to do some code and testing if it could be made.
- ▶ 2018-2021: developing RAD Basic in free time.
- ▶ 2021: For achieving 100% compatibility, it is needed change from free time work to full time work. So this is the reason of this Kickstarter.

Why matters?

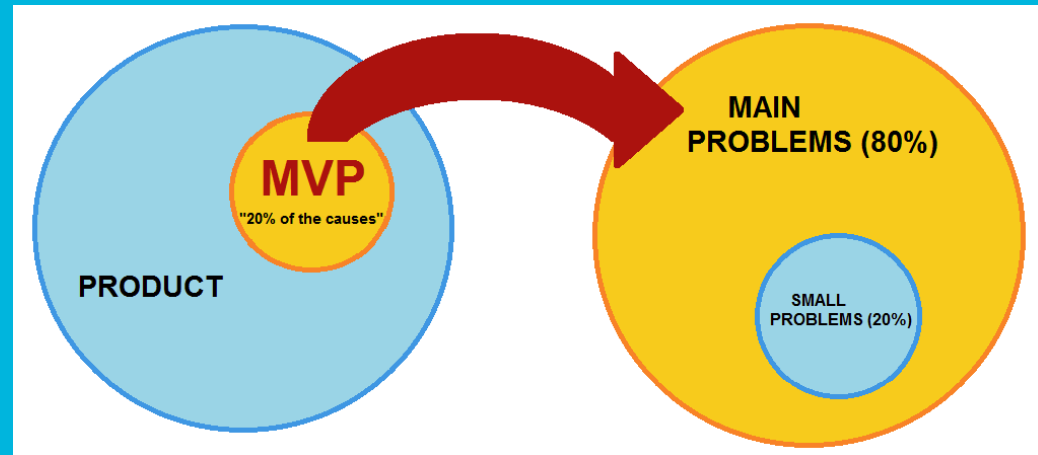
- ▶ In 2021 there is ton of maintained VB6 applications.
- ▶ We need a modernized IDE, Visual Basic 6 IDE is old and it don't have modern features as refactor operations.
- ▶ Visual Basic 6 Runtime (MSVBM60.DLL, OCXs, etc.) have minimum support from Microsoft. No new features.
- ▶ 64-bit: RAD Basic allows create and use EXE and OCX of both 32/64 bit.

About 100% Compatibility



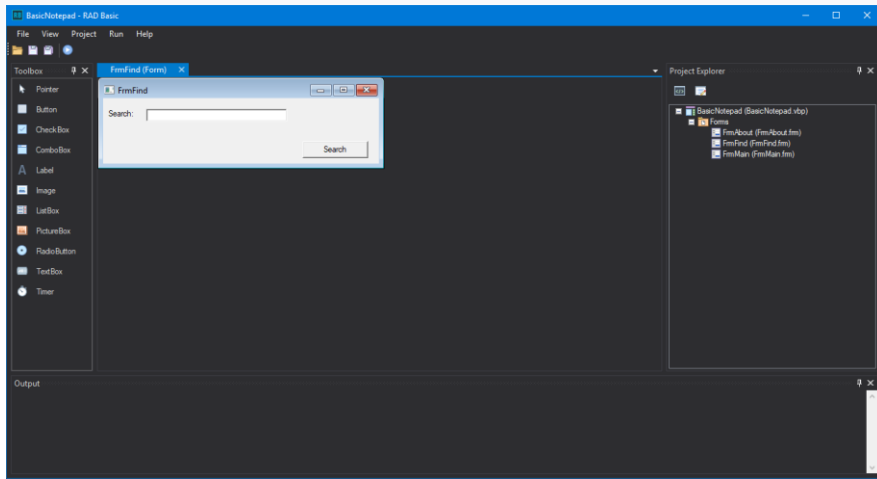
The Pareto Principle (80/20 rule):

20% of efforts bring 80% of results, and the other 80% of efforts bring only 20% of results.



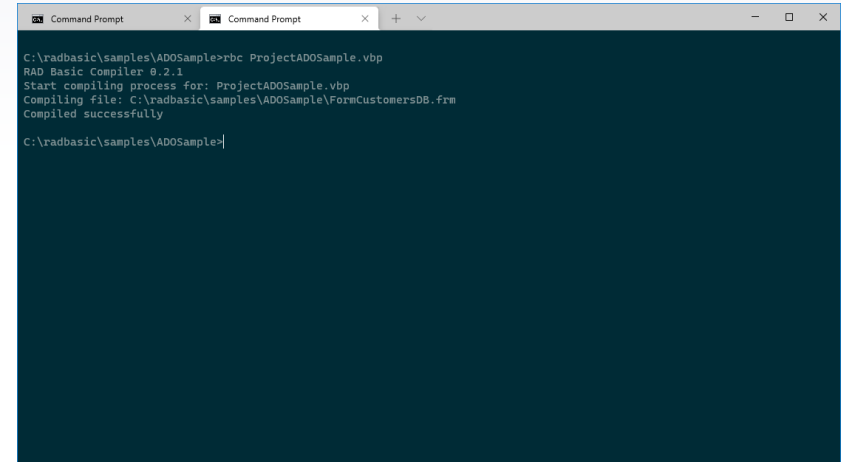
RAD Basic components

IDE



New and modern IDE with syntax highlighting, code completion, refactoring, ...

Compiler



New compiler with 32/64 bit support, documented for use by users (no obscure flags as c2.exe of VB6)

RAD Basic components

Runtime libraries

Tooling and support utilities

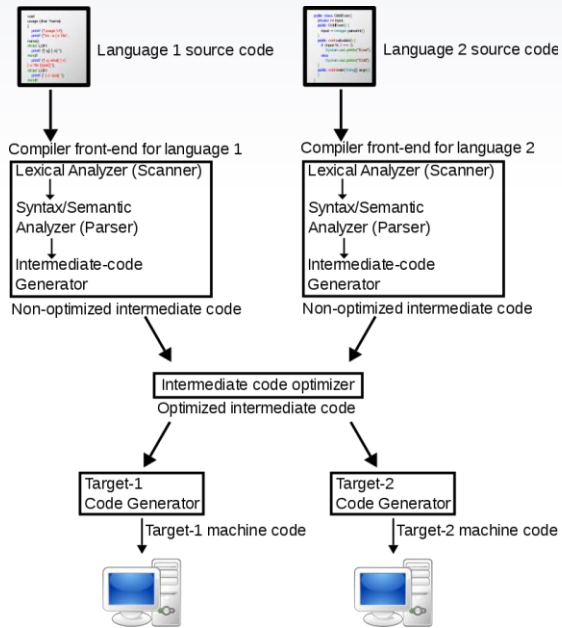
File/Folder Name	Size	Date	Description
aa			
com_enum_x64.exe	12 kB	5/2/2021 11:39	116 kB
CommandLine.dll	7 kB	5/2/2021 11:39	101 kB
Newtonsoft.Json.dll	10 kB	5/2/2021 11:39	195 kB
CommandLine	5 kB	5/2/2021 11:39	195 kB
Object Explorer.exe			Object Explorer Carles Royan
radbasic.pdb			Program Debug Database 23,5 kB
rbcomlib.dll			0.2.0.0 rbcomlib
rbforms.dll			0.2.0.0 rbforms
rbide.pdb			Program Debug Database 403 kB
WeifenLuo.WinFormsUI.Docking.dll			3.0.6.0
llvm			
com_enum_x86.exe			
CommandLine.xml			Embarcadero RAD Studio Config ... 161 bytes
Object Explorer.exe.config			Embarcadero RAD Studio Config ... 161 bytes
rbcx.exe			rbcx 0.2.1.0
rbcomlib.pdb			Program Debug Database 43,5 kB
rbforms.pdb			Program Debug Database 75,5 kB
rbmake.exe			rbmake Carles Royan
WeifenLuo.WinFormsUI.Docking.T			hemeVS2015.dll 3.0.6.0
Antlr4.Runtime.dll			4.6.5.0 Antlr4.Runtime
comigen.exe			comigen 0.2.0.0
Newtonsoft.Json.dll			12.0.3.23909 Json.NET
Object Explorer.pdb			Program Debug Database 53,5 kB
rbcx.exe.config			Embarcadero RAD Studio Config ... 189 bytes
rbcommon.dll			0.2.1.0 rbcommon
rbide.exe			RAD Basic IDE Carles Royan
rbmake.pdb			Program Debug Database 59,5 kB
Antlr4.Runtime.xml			Fitxer XML 519 kB
comigen.pdb			Program Debug Database 19,5 kB
Newtonsoft.Json.xml			Fitxer XML 691 kB
radbasic.dll			0.2.1.0 RAD Basic
rbcx.pdb			Program Debug Database 117 kB
rbcommon.pdb			Program Debug Database 733 kB
rbide.exe.config			Embarcadero RAD Studio Config ... 189 bytes
sojo_container.exe			retre 1.0.0.1

Reimplemented VB6 runtime libraries (compatible at source code level). So, it could be maintained, fixed bugs and add new features.

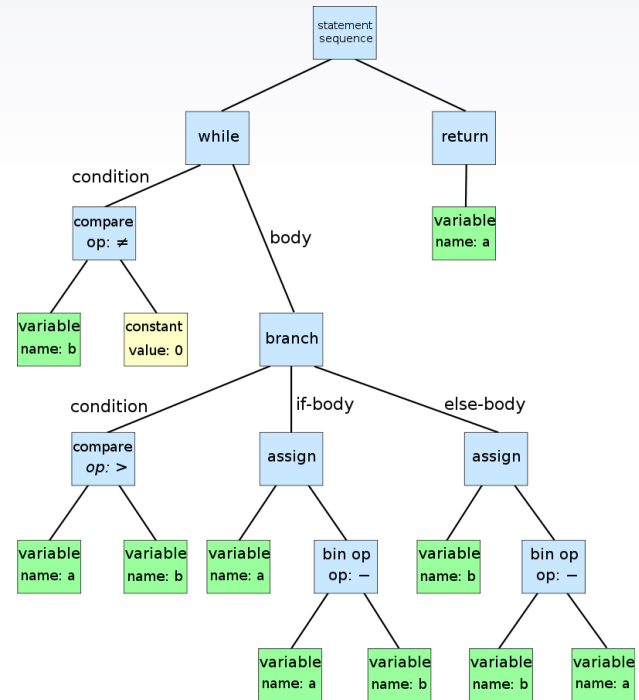
CLANG/LLVM, RBMake for C compiling, comigen for generating C code for wrapper COM classes, Object Explorer (COM and RAD Basic symbols), SOJO lib, ...

Architecture: build a Compiler

Modern compiler architecture



Build and walk an AST (Abstract Syntax Tree)



Architecture: RBC Compiler



VB6
grammar
definition

Automated lexical and syntax
analyzer

RBC implements frontend using ANTLR classes from grammar definition.

Delegates backend (optimization IL and generation of native code) to CLANG/LLVM.

C# classes for lexical
and syntax analyzer

RAD Basic Compiler
source code

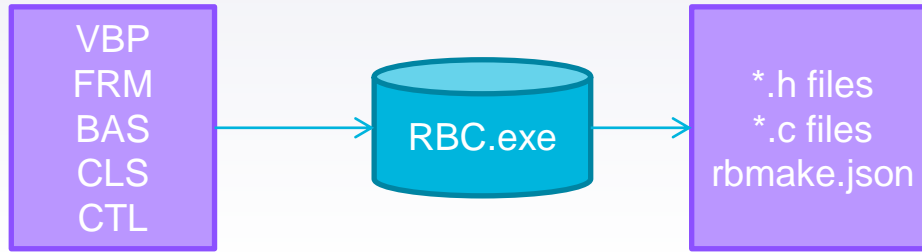
Semantic analyzer and code
generator



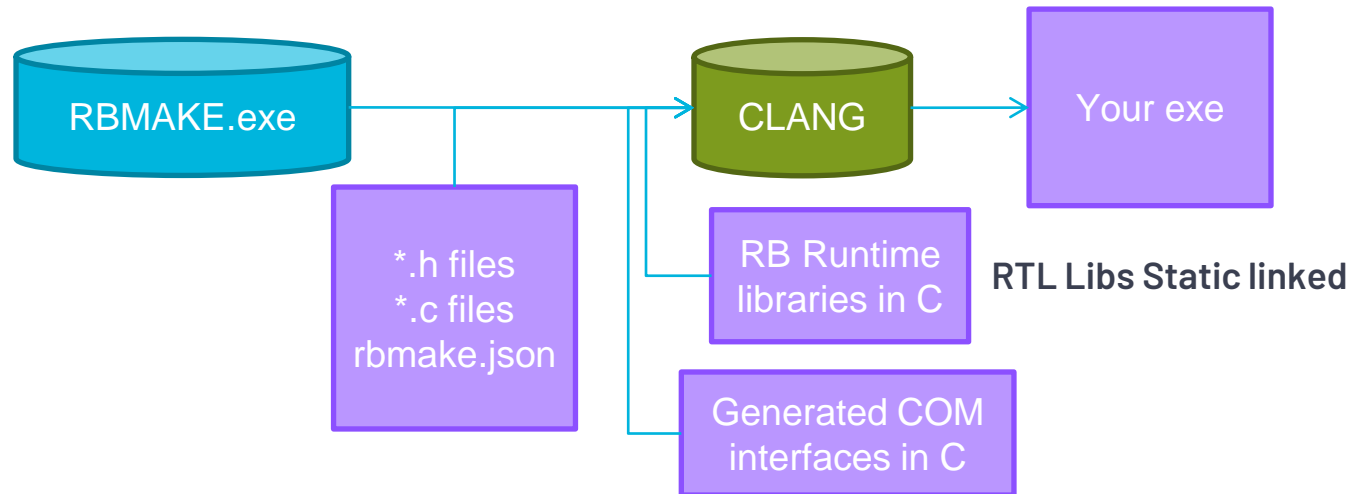
RBC generates C code from VB6. Runtime Library is written in C too. The output exe is generated from c code by CLANG (LLVM C compiler)

Architecture: Compiler process

1



2



► Architecture: Languages

- ▶ RBC Compiler in C#/.NET (one of the ANTLR lang output)
- ▶ For maintainability, IDE and toolchain in C#/.NET too
- ▶ Runtime Library in C (portability, low level)
- ▶ Generated executables are native: No .NET dependencies.

► Architecture: Why not in VB6?

- ▶ It could be great to have RAD Basic as self-hosting compiler.
- ▶ But:
 - ▶ Visual Basic is great for graphical apps. But lack support for command line.
 - ▶ There are no tools as ANTLR for automatize Lexical and Syntactical analyzer. It could be written by hand, but it slows down development-
 - ▶ IDE and language are pretty old (hey! We are here for...) so it will slow down development.
- ▶ Maybe rewritten in future... 🤔

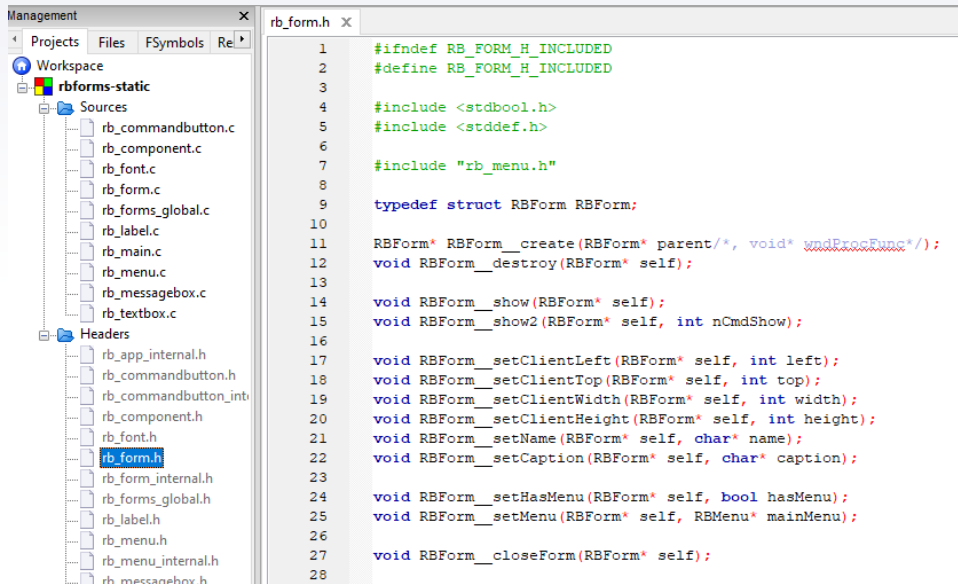
► Architecture: COM/OCX support

- ▶ RAD Basic supports COM (OLE/ActiveX) and COM is a first citizen. Visual Basic relies heavily in COM/OCX.
- ▶ RAD Basic toolchain (comigen.exe) do the plumbing for creating COM interface with the help of RBCOMLIB.
- ▶ It generates header files and utility C file for being called from application generated code. All is done with no user interaction, as in VB6.
- ▶ So using OCX/COM/OLE is as easy and transparent as it is in Visual Basic.

► Architecture: Why not COM based?

- ▶ RAD Basic is designed with cross platform in mind. COM technology is very hard to be made in a portable way.
- ▶ Easier for user to not have external dependencies. Prefer to static linked and avoid dll hell.
- ▶ If it is needed, it could have a COM wrapper interface for plain static lib runtime.
- ▶ You could use COM and RAD Basic static runtime library in the same project with no problem!

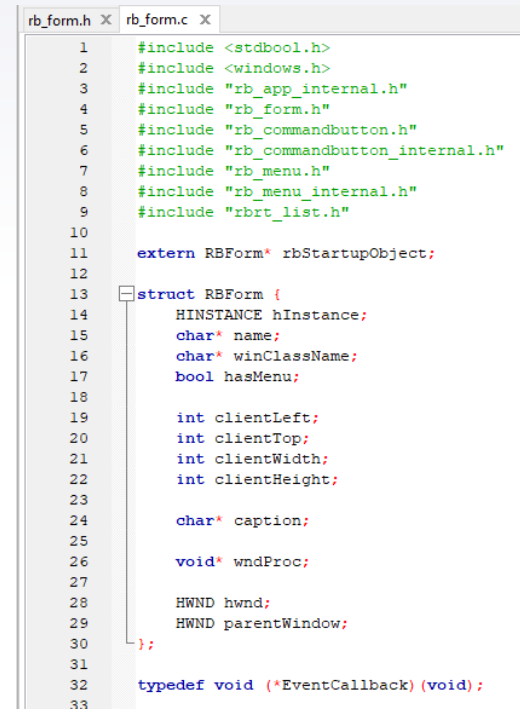
Architecture: cross platform design



The screenshot shows a development environment with a project tree on the left and the `rb_form.h` header file open in the main editor. The project tree shows a workspace named `rbforms-static` with subfolders for `Sources` and `Headers`. The `Headers` folder contains various header files, with `rb_form.h` highlighted. The `rb_form.h` file contains the following code:

```
1 #ifndef RB_FORM_H_INCLUDED
2 #define RB_FORM_H_INCLUDED
3
4 #include <stdbool.h>
5 #include <stddef.h>
6
7 #include "rb_menu.h"
8
9 typedef struct RBForm RBForm;
10
11 RBForm* RBForm_create(RBForm* parent/*, void* wndProcFunc*/);
12 void RBForm_destroy(RBForm* self);
13
14 void RBForm_show(RBForm* self);
15 void RBForm_show2(RBForm* self, int nCmdShow);
16
17 void RBForm_setClientLeft(RBForm* self, int left);
18 void RBForm_setClientTop(RBForm* self, int top);
19 void RBForm_setClientWidth(RBForm* self, int width);
20 void RBForm_setClientHeight(RBForm* self, int height);
21 void RBForm_setName(RBForm* self, char* name);
22 void RBForm_setCaption(RBForm* self, char* caption);
23
24 void RBForm_setHasMenu(RBForm* self, bool hasMenu);
25 void RBForm_setMenu(RBForm* self, RBMenu* mainMenu);
26
27 void RBForm_closeForm(RBForm* self);
28
```

RBC compiles using this platform agnostic header

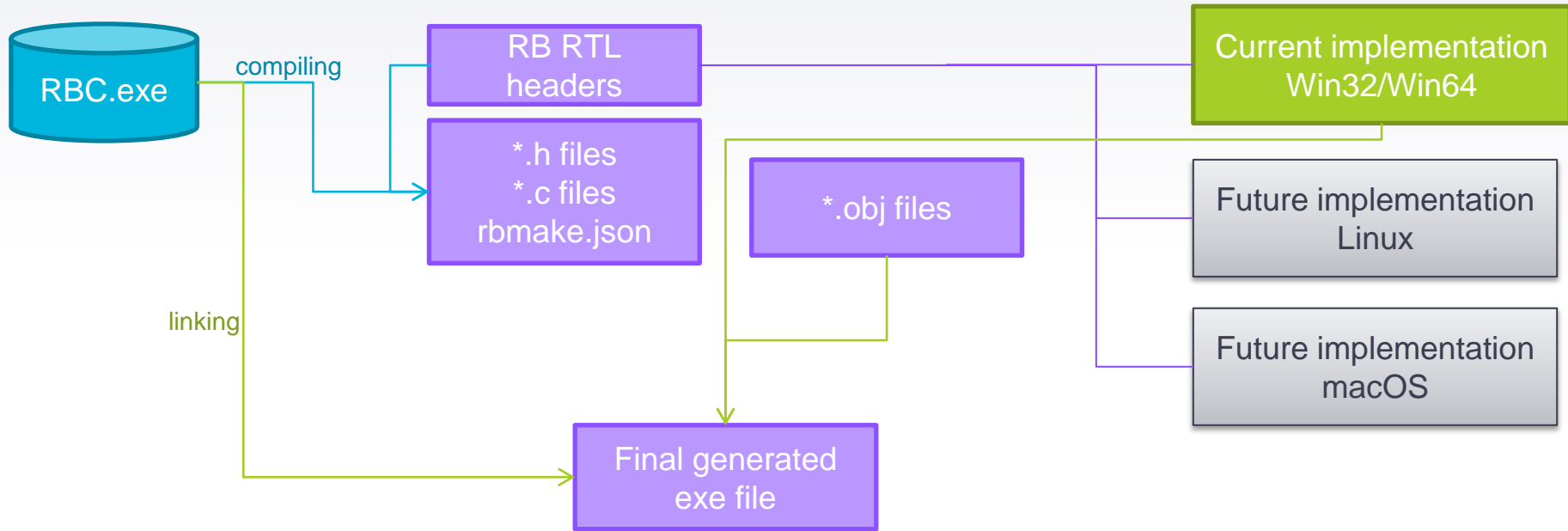


The screenshot shows a development environment with the `rb_form.c` source file open in the main editor. The file contains the following code:

```
1 #include <stdbool.h>
2 #include <windows.h>
3 #include "rb_app_internal.h"
4 #include "rb_form.h"
5 #include "rb_commandbutton.h"
6 #include "rb_commandbutton_internal.h"
7 #include "rb_menu.h"
8 #include "rb_menu_internal.h"
9 #include "rbrt_list.h"
10
11 extern RBForm* rbStartupObject;
12
13 struct RBForm {
14     HINSTANCE hInstance;
15     char* name;
16     char* winClassName;
17     bool hasMenu;
18
19     int clientLeft;
20     int clientTop;
21     int clientWidth;
22     int clientHeight;
23
24     char* caption;
25
26     void* wndProc;
27
28     HWND hwnd;
29     HWND parentWindow;
30 };
31
32 typedef void (*EventCallback)(void);
33
```

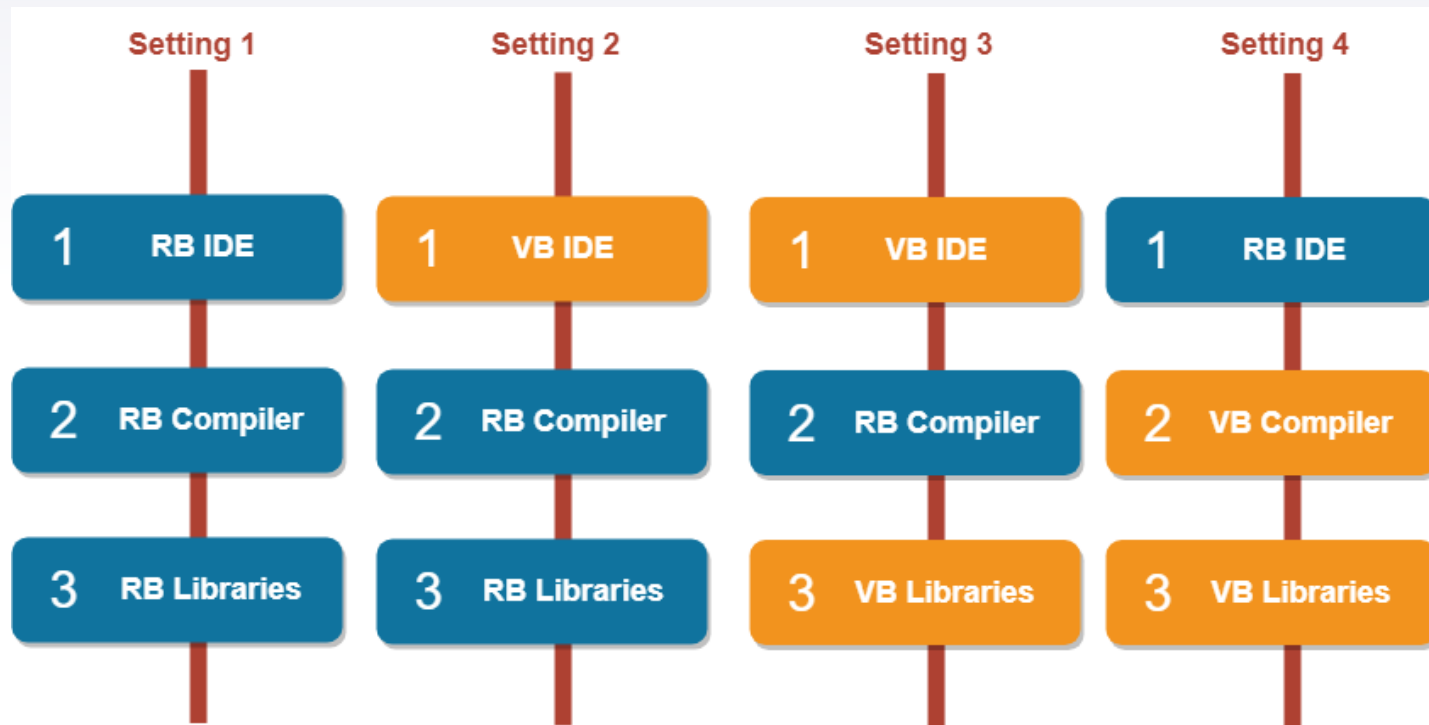
Win 32 implementation is wrapped in static lib

Architecture: cross platform design

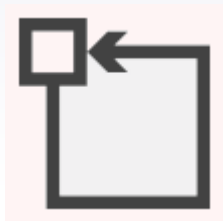
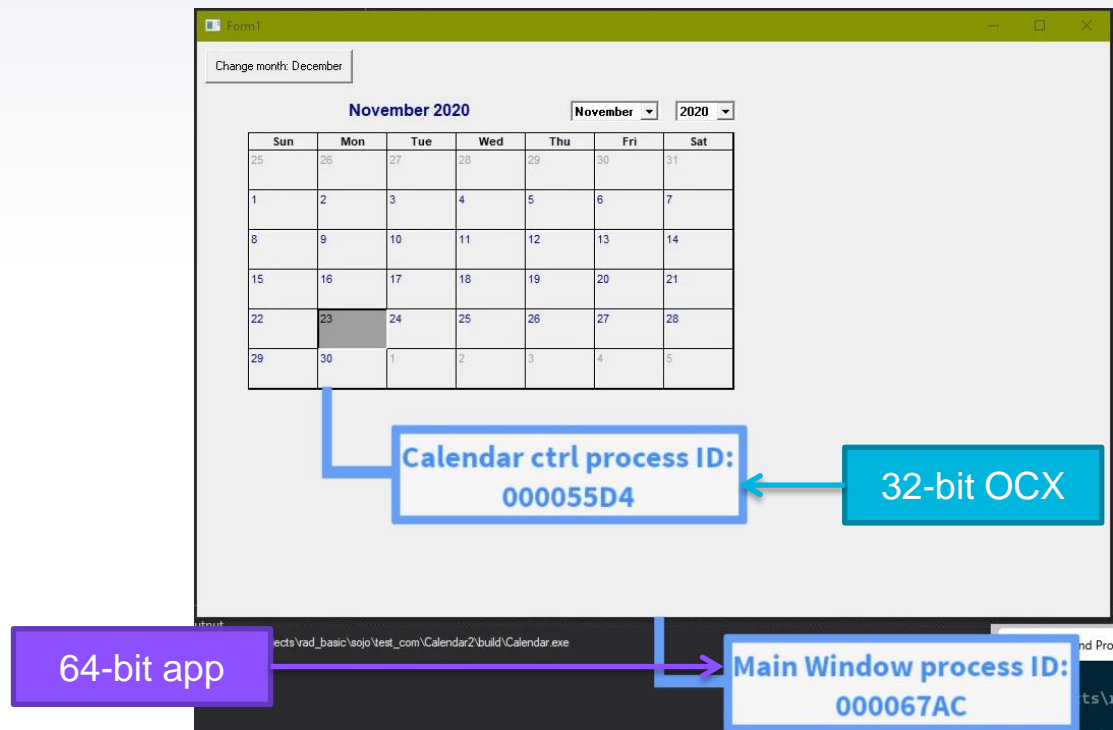


There is one header as API and there will be **n implementations** for each platform. So, it could be swapped in compile time.

Architecture: Three layer arch



Architecture: SOJO library



SOJO library: allows 32-bit OCX run inside 64-bit applications.

It could be integration problems, so it have to be used as a workaround

▶ Main goal: compatibility

- ▶ RAD Basic is focused on compatibility. So, there won't be new features or language enhancements.
- ▶ RAD Basic has no conversions or migrations.
- ▶ RAD Basic allows easy “transition” from VB. It works with native files, so you could go back and forth between environments freely.
- ▶ RAD Basic follow Visual Basic language reference. But not only this, it behaves some way as VB6 does, implementing same tricks and oddities.

▶ Main goal: compatibility

build	3/5/2021 2:14	Carpeta de fitxers	
tmp	3/5/2021 2:14	Carpeta de fitxers	
FormCustomersDB.frm	28/4/2021 2:32	Visual Basic Form ...	4 kB
ProjectADOSample.exe	20/4/2021 3:41	Aplicació	24 kB
ProjectADOSample.vbp	28/4/2021 2:31	Visual Basic Project	1 kB
ProjectADOSample.vbp.rbproperties	28/4/2021 2:32	Fitxer RBPROPERTI...	1 kB
ProjectADOSample.vbw	28/4/2021 4:08	Visual Basic Projec...	1 kB
sample_customers.accdb	28/4/2021 4:08	Microsoft Access ...	412 kB

Extended properties as platform (32-bit/64-bit) are stored in a separate file, for preserve compatibility with VB6 file format

▶ True current status (not as good)

- ▶ There is support for using OCX, although it has to be improved (it fails with coClasses with more than one interface, and sometimes needs to tweak the output code) Thanks **OleView.exe!!!**
- ▶ At this moment only compiles exe files. Working in OCX generator: OCX projects could be loaded in IDE but compiling fails.
- ▶ IDE with code completion deactivated. Working in LSP (Language Server protocol) implementation.
- ▶ There is a solid foundation, but there is a lot of work to do for reaching 100% compatibility.

▶ True current status (the good)

- ▶ It has support for basic flow structures (IF, WHILE), method calls, variable declaration and some basic math operation.
- ▶ Core features are developed and RBC (RAD Basic Compiler) could compile some small application test (calculator, ADO navigating through database, simple text editor,...).
- ▶ There is a solid foundation, well designed architecture and there are solved many challenges as: visual forms designer, (partial) COM support, future cross platform, ...

A stage spotlight shines down from the top center, illuminating the text "LIVE DEMO" in a bright, glowing purple and white light. The text is surrounded by several starburst and spark effects, creating a dynamic and attention-grabbing visual. The background is a dark, deep purple, and the spotlight's beam creates a soft glow on the floor below.

**LIVE
DEMO**

THANKS!

AMA time: Any questions?

You can find me at:

- ▶ Twitter: @radbasic
- ▶ <https://www.facebook.com/radbasic>
- ▶ info@radbasic.dev

