embarcadero

64 Bit Delphi
What does it all mean?

David Intersimone “David I”
VP of Developer Relations and Chief Evangelist
davidi@embarcadero.com
http://blogs.embarcadero.com/davidi
Twitter: davidi99
Skype: davidi99

Agenda

- RAD Studio Product Family Overview
- 64bit computing – What's the same? Different?
  - Data types
- The CPU - Instructions, Addresses
- What this means for Delphi
- What can I do today?
- Q&A

Product Editions

Delphi XE

- Starter
- Professional
- Enterprise
- Architect

C++Builder XE

- Starter
- Professional
- Enterprise
- Architect

Delphi Prism XE

- Professional
- Enterprise

RadPHP XE

RAD Studio XE

- Professional
- Enterprise
- Architect

Included all IDE's

ER Studio Developer Edition
Included on Architect edition

InterBase Developer Edition FREE
for development

Delphi XE and C++Builder XE Starter Edition

- Low cost solution for hobbyists, students, and independent developers to be able to get started building and distributing apps
- VCL Form Designer to visually build native Windows applications
- Full-featured debugger with color syntax highlighting
- Hundreds of included components
  - Touch and Gesture support
  - Ribbon Controls
  - IBX for database development
  - Allow third party components

Recent investments in RAD Studio

- Acquisitions

- Strategic acquisitions

- Investing in new and future technologies for software development

INVESTING IN YOUR FUTURE!!!

embarcadero

RAD Studio Future

Cross-Platform

Community 3.0

VCL+

DataSnap Server

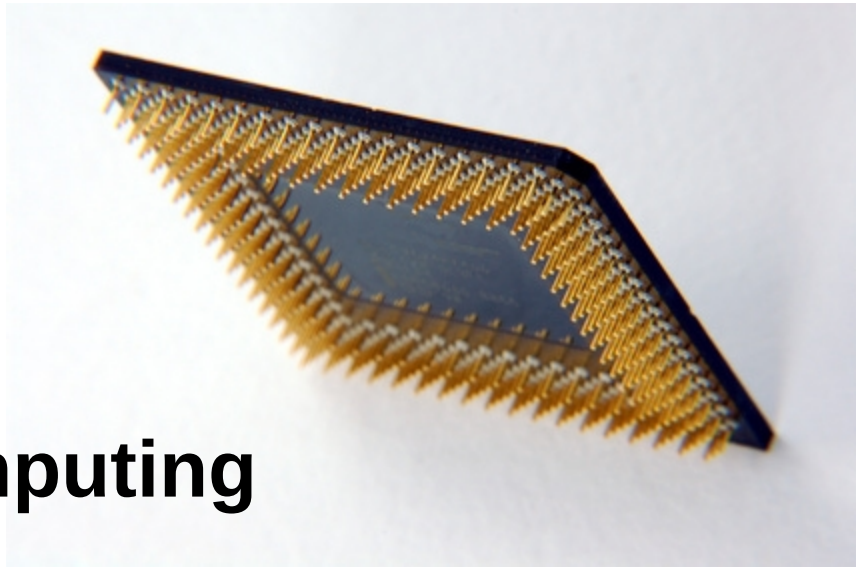
Modular Compiler Architecture

Delphi Language Everywhere

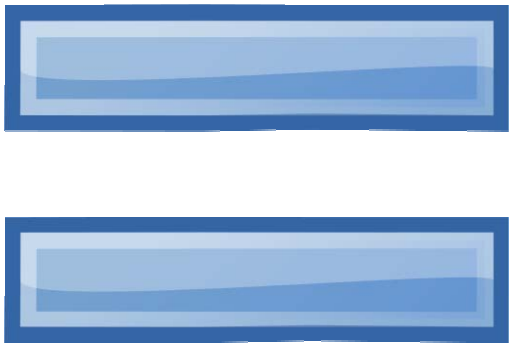
Mobile Connectivity

embarcadero

Bit Computing



What's the same?

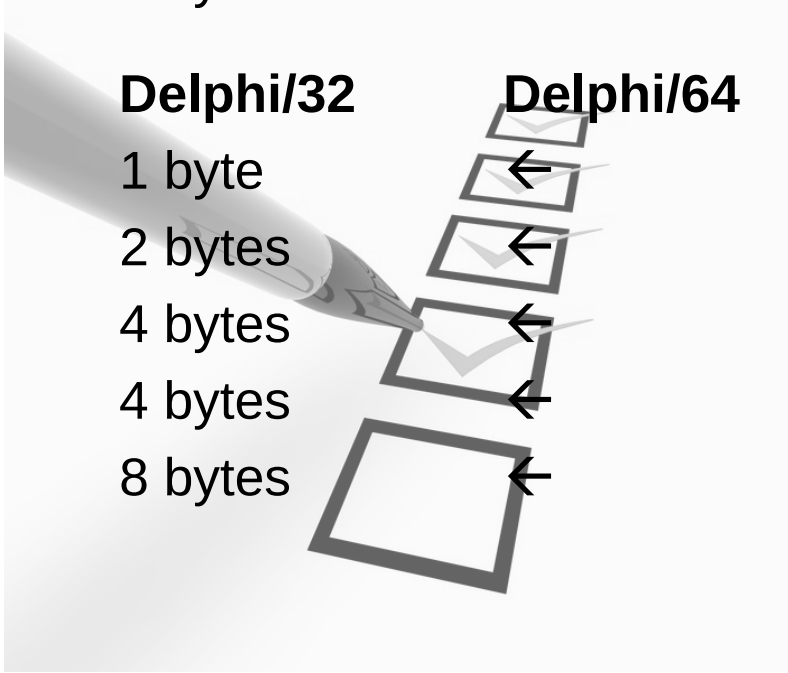


- Integer, Longint, Cardinal – still 32bits
- Int64, UInt64 – still 64bits
- UnicodeString, AnsiString, WideString
- Exceptions
- Runtime Library (RTL)
- SysUtils, Classes, etc…

Delphi 32 and 64-bit Type Sizes

Signed types	Delphi/32	Delphi/64
ShortInt	1 byte	←
SmallInt	2 bytes	←
LongInt	4 bytes	←
Integer	4 bytes	←
Int64	8 bytes	←

Unsigned types	Delphi/32	Delphi/64
Byte	1 byte	←
Word	2 bytes	←
LongWord	4 bytes	←
Cardinal	4 bytes	←
UInt64	8 bytes	←



What's different?

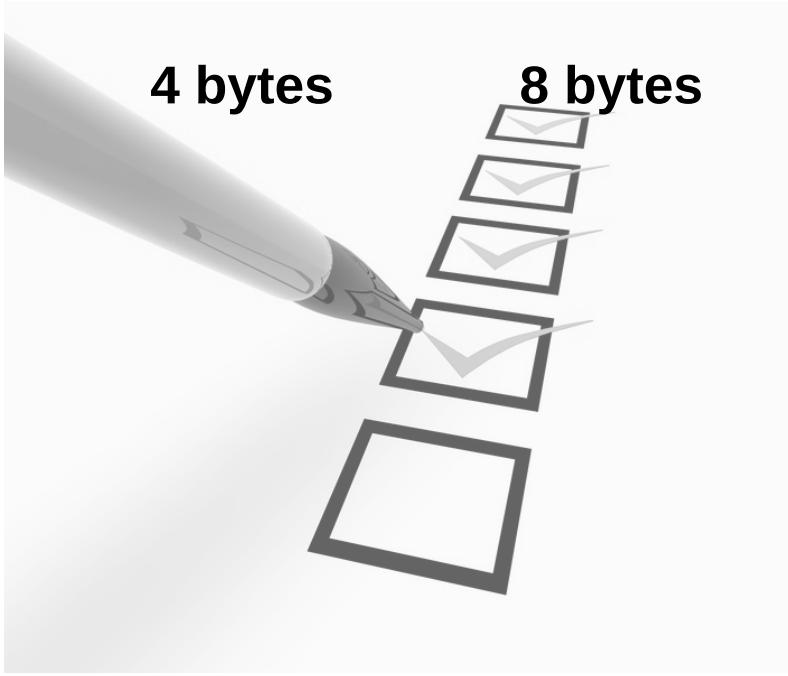
- NativeInt, NativeUint – 64bits
- Pointer (all pointers) – 64bits
- Dynamic Arrays – 64bit indexing
- Floating point math - Double

Delphi 32 and 64-bit Type Sizes

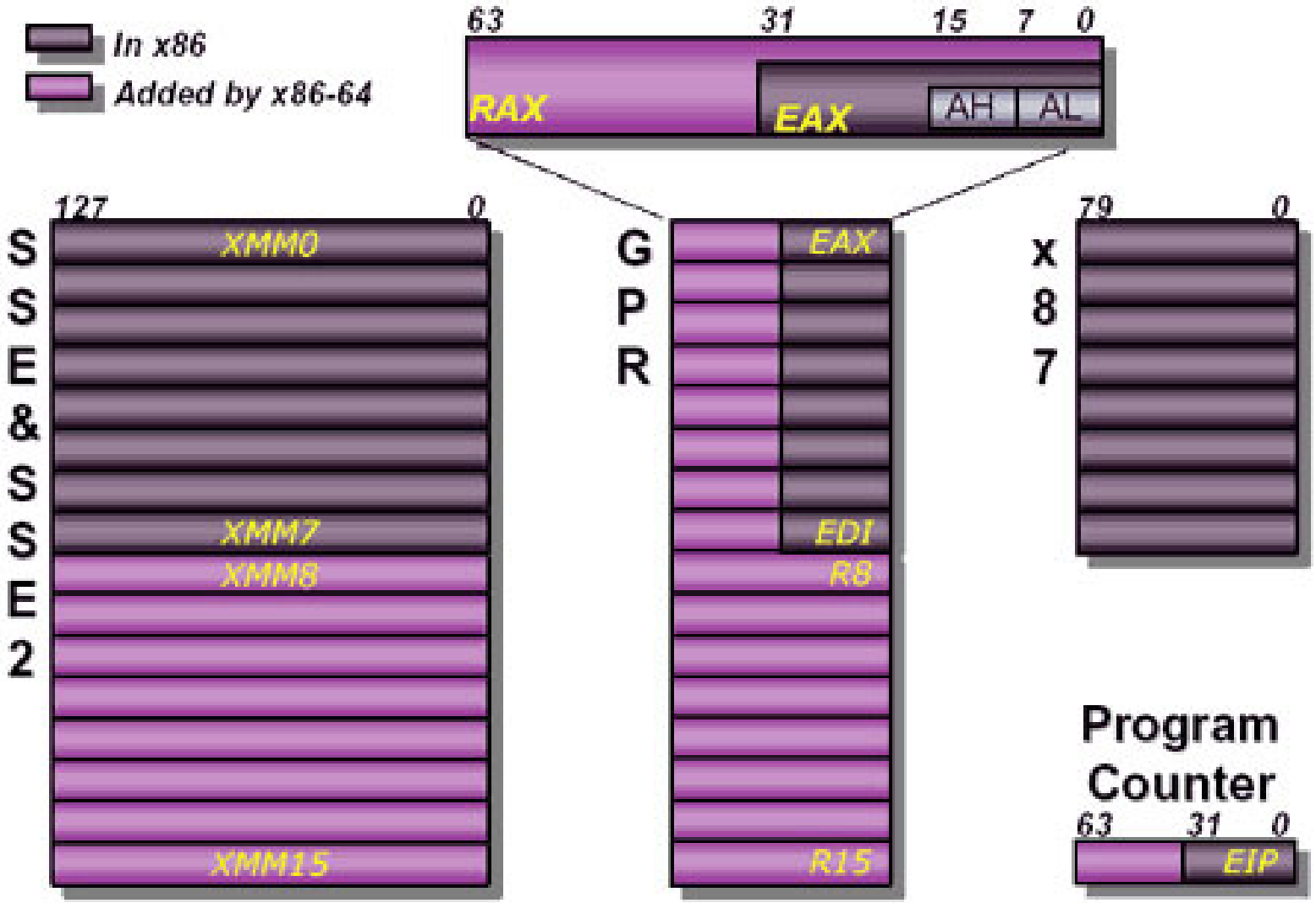
Signed types	**Delphi/32**	**Delphi/64**
NativeInt	**4 bytes**	**8 bytes**
Unsigned types	**Delphi/32**	**Delphi/64**
NativeUInt	**4 bytes**	**8 bytes**

Delphi 32 and 64-bit Pointer Types

Pointer types	Delphi/32	Delphi/64
Pointer		
String		
Class instance		
Class reference		
Interface		
AnsiString		
WideString		
UnicodeString		
Procedure pointer		
Dynamic array		
PAnsiChar		
PWideChar		
PChar		



The CPU – Programming model



embarcadero

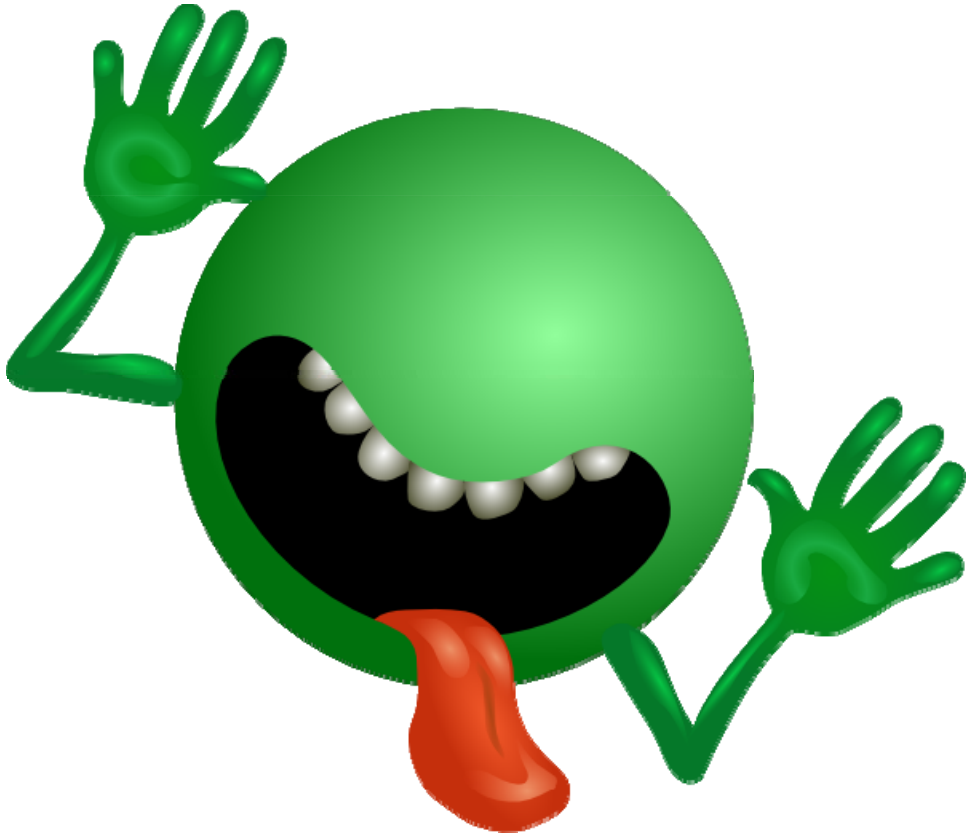
Instructions and Addresses

- 64 bit address space
- Limited by physical hardware
- Same core Intel instruction set
- New REX prefix for 64bit instructions
- RIP-relative addressing
- Jumps – relative +/- 2GB
- 16 byte stack alignments

- Online resources
  - http://msdn.microsoft.com/en-us/magazine/cc300794.aspx
  - http://msdn.microsoft.com/en-us/library/7kcdt6fy.aspx

What does this mean for Delphi?

Don’t panic!

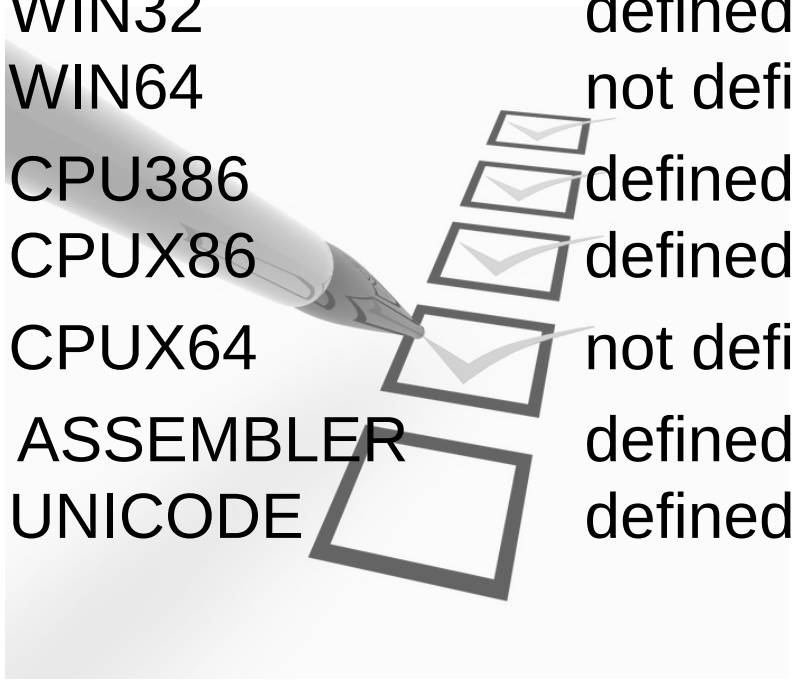


Delphi 64-bit on Windows

- Same Windows API
  - CreateWindowEx, PeekMessage, etc..
- Same Delphi RTL
  - SysUtils, Classes, Generics.Collections, etc…
- Same VCL
  - Forms, Graphics, Controls, Menus, etc..

Pre-Defined Conditionals

Category	Identifier	dcc32	dcc64
Compiler	DCC	defined	defined
	VER230	defined	defined
Platform	MSWINDOWS	defined	defined
	WIN32	defined	not defined
	WIN64	not defined	defined
CPU	CPU386	defined	not defined
	CPUX86	defined	not defined
	CPUX64	not defined	defined
Availability	ASSEMBLER	defined	defined
	UNICODE	defined	defined



Delphi 64-bit on Windows – some gotcha's

- SizeOf(Pointer) <> SizeOf(Integer)
  - Integer<->Pointer casts will break in 64bit
  - SizeOf(THandle) = SizeOf(Pointer)
  - All Handles = SizeOf(Pointer) (HWND, HDC, etc..).
- All code in process must be 64bit
  - Must have 64bit versions of external non-Delphi libraries (DLLs)
- One, and only one, calling convention
  - **`register,pascal,cdecl,stdcall`** ignored.
- **`safecall`** is still "special"
- Old "pointer math" code may break
  - `Works in 32 and 64bit: MyPtr := PByte(P) + 10;`

Delphi 64-bit on Windows – some gotcha's

- Inline Assembly
  - Cannot mix **`asm`** blocks with Pascal code
  - Only procedural level **asm** blocks supported
  - Stack **must be 16-byte** aligned at each call instruction
  - Define locals for temp storage
  - Do not modify the RSP stack pointer
  - New unified calling convention. First 4 parameters in registers, RCX, RDX, R8, R9 (or XMM0-XMM3)
- Exception unwinding
  - No change for pure Delphi code. Exceptions function identically.
  - Inline Assembly can cause exception unwinding to fail if not properly written.

Windows API gotcha's

- SetWindowLong / GetWindowLog should be replaced by SetWindowLongPtr / GetWindowLongPtr for GWLP_HINSTANCE, GWLP_WNDPROC, etc… as they return pointers and handles.
  - Pointers passed to SetWindowLongPtr should be type-casted to LONG_PTR and not to Integer / Longint.
- SetWindowLong mapped to SetWindowLongPtr in Windows.pas.
  - Calls to our declaration of SetWindowLong are safe, as long as they are cast correctly.

Windows API gotcha's (con't)

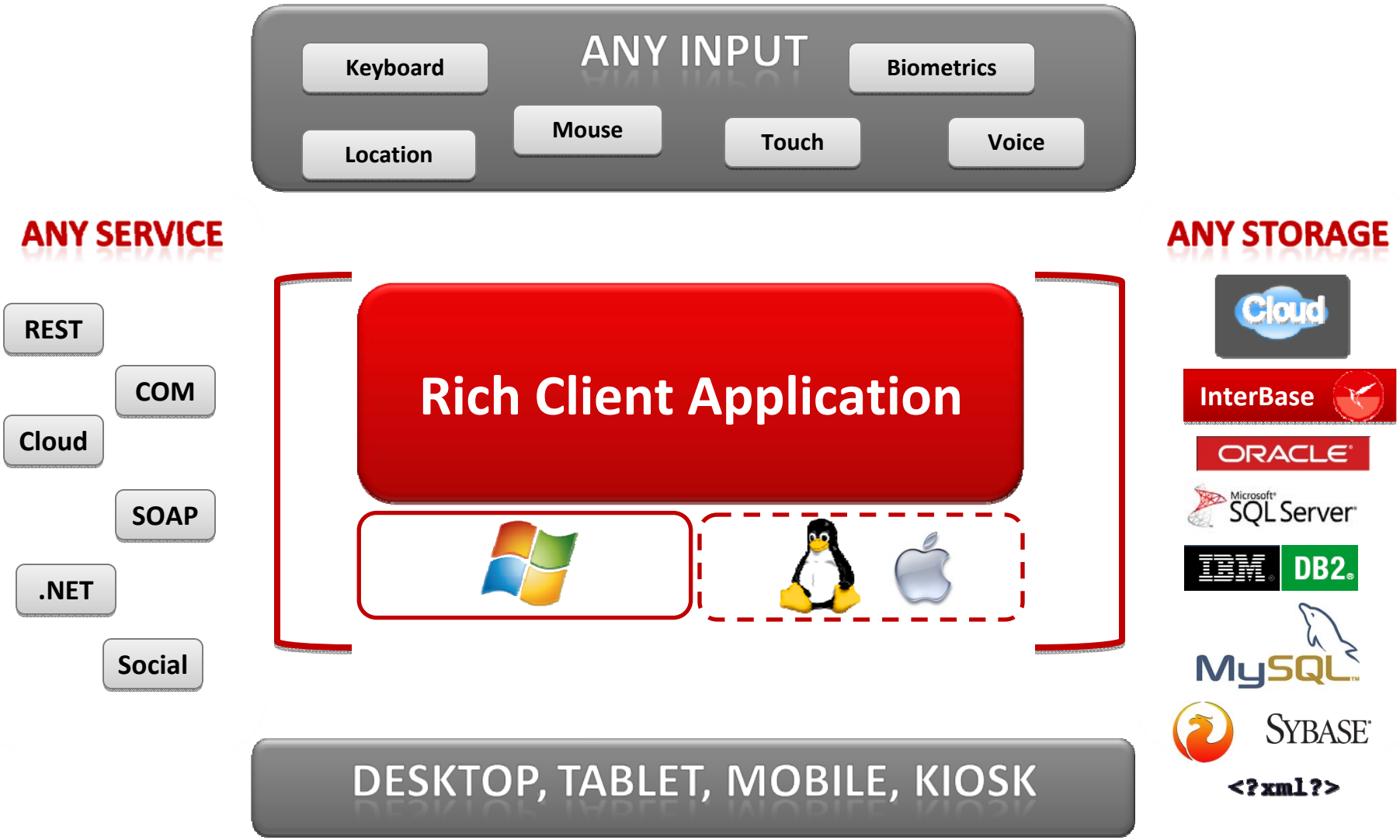
- Use explicit casts to WPARAM and LPARAM where appropriate.
  - Example: Passing pointers through SendMessage
    ```
    SendMessage(hWnd, WM_SETTEXT, 0,
    LPARAM(@MyCharArray));
    ```
- `Use LRESULT to cast message results`
  - `Example: Message.Result := LRESULT(Self);`
- `Message cracker records (TWMxxx) changed`
  - `Alignment changes and field-size changes`

What can I do today?

- Find all Integer<->Pointer casts, including Integer<->instance casts.
  - Check for Pointer size assumptions
- Ensure external dependencies are also 64bit
  - Image/bitmap libraries
  - Hardware interfaces libraries
  - ActiveX controls
- Consider rewriting Assembler in pure-Pascal
  - Better future portability (think ARM CPUs…)
  - Rely more on algorithmic performance rather than raw assembly performance.

Delphi 64-bit Demonstration

RAD Studio Client Vision

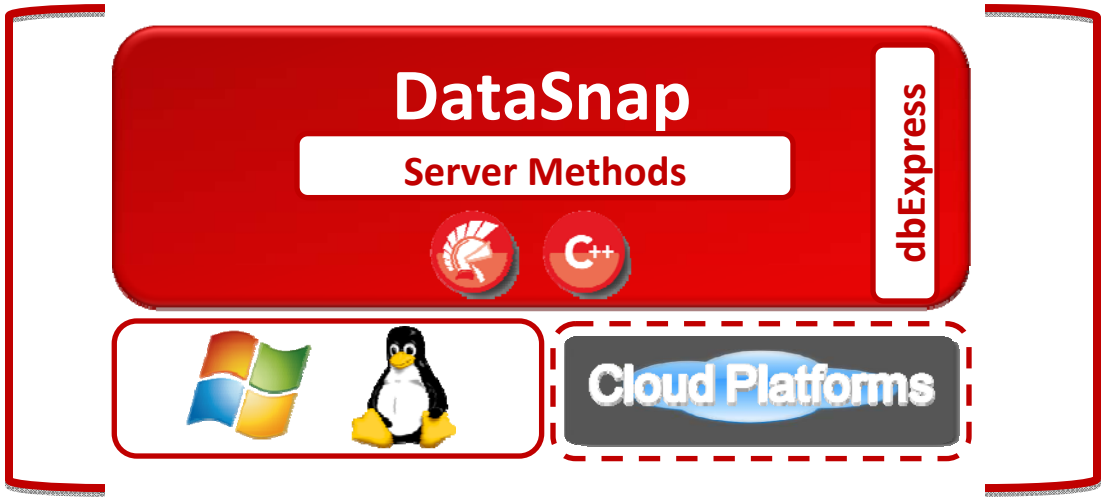


embarcadero

RAD Studio Server Vision

ANY SERVICE

- REST
- COM
- Cloud
- SOAP
- .NET
- Social



ANY STORAGE

Cloud
InterBase
ORACLE
Microsoft SQL Server
IBM DB2
MySQL
SYBASE
<?xml?>

embarcadero

embarcadero

Q & A

Thank You ☺

David Intersimone “David I”
VP of Developer Relations and Chief Evangelist
davidi@embarcadero.com
http://blogs.embarcadero.com/davidi
Twitter: davidi99
Skype: davidi99