Visual Basic

Programming the Future
An Unlikely Success Story

- Basic: The Epitome of a “Toy” Language
- GUI Programming: A Programmer’s Nightmare
- Custom Controls: Do-Dads and Useless Trinkets
- Database Programming: Complicated, Time-Consuming, Expensive
The “Basic” Solution

- The Dialog Box was Elevated from Second Class Status
- The Visual Programming Interface Allowed Controls to be *Drawn* rather than *Coded*
- A “Program” Now Consists of Several Drawings, Called *Forms*, and Some Basic Code
Standard Controls

- Check1
- Check2
- Check3D1
- Check3D2

Scroll Bar Parts:
- Up Button
- Above Slider
- Slider
- Below Slider
- Down Button

Print Destination:
- Default Printer
- Window (Preview)
- File

Relationship:
- Daughter
- Child
- Sgl Parent
- Mother
- Father
- Grand Parent
- Grand Mother
- Grand Father

Date Posted: 9/25/95
Control Interface

- Properties: Variables Known to the Control Which Affect Its Appearance
  - Text
  - Border

- Events: Asynchronous Function Calls in Response to Changes in The Interface
  - GotFocus
  - Changed

- A Few Standard Functions
Visual Basic Custom Controls

- Contained in A Separate Function Library
- Written in C (Usually)
- Responds to Reading & Writing Properties
- Issues Events
- May Process Standard Function Calls
Database Functionality

- A New *Data-Control* was Added
- A Data-Control Represents a Single Relational Table (Queries Included)
- Certain Other Controls Can be *Bound* to Database Fields Through the Data Control
- Changing The Contents of a Data-Aware Control Changes the Database
The Crucial Point

The Data-Control Supports Almost All Existing Database Systems, *without additional software*
The Grid Control

Ideal for Data-Base Support, BUT

It IS NOT Data-Aware
The Avalanche Begins

◆ 3rd-Party Developers Create Various Data-Aware Grids
◆ Some Developers Realize that a Grid Looks Like a Spreadsheet -- Why Shouldn’t it Act Like One?
◆ The Text Control Acts Like a Minature Text Editor -- Why Shouldn’t it Act Like a Real Text Editor?
Current Custom Controls

◆ Controls Reproduce the Functionality of Virtually All Existing PC Applications
◆ Distribution is, For the Most Part, Royalty-Free
◆ Visual Basic has become THE Preferred Implementation for DB Applications
◆ Proliferation of Controls has Extended VB Programming to Many Other Areas
Examples of Controls

- Full-Featured Word Processors
- Excel Compatible Spread Sheets
- Full-Featured Modem Communications
- Voice-Mail Systems
- Image Editors
- Object-Oriented Graphics Drawing
- Web Browsers
- FAX Interfaces
- Hundreds of Others
A Brief History of Programming

◆ Block-Structured Languages were Supposed to Revolutionize Programming
  – They Didn’t

◆ Top Down Structured Programming was Supposed To Revolutionize Programming
  – It Didn’t (Well, Structured Programming Helped!)

◆ Object-Oriented Languages were Supposed to Revolutionize Programming
  – They Didn’t
Current Events

- Visual-Basic wasn’t Supposed to Revolutionize Anything
  - It Did
- VB Enables one to Program “Far Above the Statement Level”
  - Nothing Else Really Worked
- VB Allows Code Reuse
  - Nothing Else Really Does
- VB Reduces Programming Time From Days to Minutes
  - Anyone Else Who Claims This is Lying
Win FHDL

- Human Interface was Developed Entirely With Custom Controls
- The Parser, Simulator, Macro Processor, PLA Processor, and ROM Processor are Locally Developed Custom Controls
- Several Commercial Custom Controls are Used in the Interface
You CAN Try This At Home!

- The Locally Developed Custom Controls Included in Win FHDL are *Developer Versions*
- You Can Use These Controls to Develop Your Own FHDL User Interface
- Source Code is Available on Request
- Although Win FHDL Source Code is Available, It is Useless without the Add-Ons
Visual Basic

Programming the Future
An Unlikely Success Story

• Basic: The Epitome of a “Toy” Language
• GUI Programming: A Programmer’s Nightmare
• Custom Controls: Do-Dads and Useless Trinkets
• Database Programming: Complicated, Time-Consuming, Expensive
The “Basic” Solution

• The Dialog Box was Elevated from Second Class Status
• The Visual Programming Interface Allowed Controls to be *Drawn* rather than *Coded*
• A “Program” Now Consists of Several Drawings, Called *Forms*, and Some Basic Code
Standard Controls

- Check1
- Check2
- Check3D1
- Check3D2

Scroll Bar Parts:
- Up Button
- Above Slider
- Slider
- Below Slider
- Down Button

Print Destination:
- Default Printer
- Window (Preview)
- File

Relationship:
- Daughter
- Child
- Sgl Parent
- Mother
- Father
- Grand Parent
- Grand Mother
- Grand Father

Date Posted: 9/25/95
Control Interface

• Properties: Variables Known to the Control Which Affect Its Appearance
  – Text
  – Border

• Events: Asynchronous Function Calls in Response to Changes in The Interface
  – GotFocus
  – Changed

• A Few Standard Functions
Visual Basic Custom Controls

• Contained in A Separate Function Library
• Written in C (Usually)
• Responds to Reading & Writing Properties
• Issues Events
• May Process Standard Function Calls
Database Functionality

• A New *Data-Control* was Added
• A Data-Control Represents a Single Relational Table (Queries Included)
• Certain Other Controls Can be *Bound* to Database Fields Through the Data Control
• Changing The Contents of a Data-Aware Control Changes the Database
The Crucial Point

The Data-Control Supports Almost All Existing Database Systems, without additional software
The Grid Control

Ideal for Data-Base Support, BUT

It IS NOT Data-Aware
The Avalanche Begins

• 3rd-Party Developers Create Various Data-Aware Grids

• Some Developers Realize that a Grid Looks Like a Spreadsheet -- Why Shouldn’t it Act Like One?

• The Text Control Acts Like a Minature Text Editor -- Why Shouldn’t it Act Like a Real Text Editor?
Current Custom Controls

• Controls Reproduce the Functionality of Virtually All Existing PC Applications
• Distribution is, For the Most Part, Royalty-Free
• Visual Basic has become THE Preferred Implementation for DB Applications
• Proliferation of Controls has Extended VB Programming to Many Other Areas
Examples of Controls

- Full-Featured Word Processors
- Excel Compatible Spread Sheets
- Full-Featured Modem Communications
- Voice-Mail Systems
- Image Editors
- Object-Oriented Graphics Drawing
- Web Browsers
- FAX Interfaces
- Hundreds of Others
A Brief History of Programming

- Block-Structured Languages were Supposed to Revolutionize Programming
  - They Didn’t

- Top Down Structured Programming was Supposed To Revolutionize Programming
  - It Didn’t (Well, Structured Programming Helped!)

- Object-Oriented Languages were Supposed to Revolutionize Programming
  - They Didn’t
Current Events

• Visual-Basic wasn’t Supposed to Revolutionize Anything
  – It Did
• VB Enables one to Program “Far Above the Statement Level”
  – Nothing Else Really Worked
• VB Allows Code Reuse
  – Nothing Else Really Does
• VB Reduces Programming Time From Days to Minutes
  – Anyone Else Who Claims This is Lying
Win FHDL

- Human Interface was Developed Entirely With Custom Controls
- The Parser, Simulator, Macro Processor, PLA Processor, and ROM Processor are Locally Developed Custom Controls
- Several Commercial Custom Controls are Used in the Interface
You CAN Try This At Home!

- The Locally Developed Custom Controls Included in Win FHDL are *Developer Versions*
- You Can Use These Controls to Develop Your Own FHDL User Interface
- Source Code is Available on Request
- Although Win FHDL Source Code is Available, It is Useless without the Add-Ons