



Installing and Running the TechCon2006 Demos

Installing BBj and TechCon2006 Demos

1. Download and extract BBj

- a. Download BBj from <https://www.basis.com/products/bbj/download.html>
- b. Read the section entitled “Download Instructions” for specific download instructions.
- c. Select the release type “Current Release Product Version #.#,” then select the required platform.
- d. Select all components including the IDE; documentation is optional.
- e. Install BBj to the default directory
 - 1) For a Windows installation: C:\Program Files\basis
 - 2) For a *nix installation: /usr/local/basis

2. Download and extract the TechCon2006 archive

- a. Download the demo from <http://www.basis.com/solutions/training.html>
- b. Extract the archive into the root directory, which creates a \techcon06 directory
 - 1) For a Windows, run: techcon06.exe
 - 2) For a *nix, extract: techcon06.tar.gz

3. Install the SPROCs demo (Windows only)

- a. Browse to \techcon06\SProcs\ThirdPartyApp\Package directory inside File Explorer.
- b. Double click on the setup.exe file to launch BBjSProc setup.
- c. Click [OK].
- d. Click on the square button with a picture of a computer and accept the default directory for the installation.
- e. Click [Continue] to create a BBjSproc program group. The install then sets up the Microsoft Data Access Components.
- f. Click [Finish].

4. Install Visual PRO/5 6.0 (Windows only)

- a. Download Visual PRO/5 from <https://www.basis.com/products/vpro5/download.html>.
- b. Install Visual PRO/5 into the C:\basis directory according the [Visual PRO/5 6.01 Installation Guide for Windows](#).
- c. Edit the line beginning with the word PREFIX to read:
PREFIX C:/BASIS/tools/guibuild/ C:/BASIS/std/ C:/BASIS/GML/ C:/BASIS/VPRO5/
Be sure the guibuild directory appears before the other directories in the PREFIX.

5. Setup the ChileCo database

- a. Launch Enterprise Manager by invoking **Start->Programs->Basis->BBj->Enterprise Manager**.
- b. Double click on 127.0.0.1:2002.
- c. Enter `admin` as the user name and `admin123` as the password in the server login dialog box.
- d. Expand the “Databases” node and right click the “Databases” folder.
- e. Select **Attach to Existing Dictionary** and enter `ChileCo` for the database name. It is important to enter the same case as shown for the demos to successfully access the database.
- f. Click the browse [...] button beside the Data Files textbox and navigate to the `\techcon06\ChileCo` directory.
- g. Highlight the `data` directory and click [Select].
- h. Click beside the Dictionary Files textbox and go up one directory to highlight the `bbdict` directory.
- i. Click [Select], then [OK]. The ChileCo database now appears on the list in the right pane with the other databases.
- j. Run the `CreateDatabase.src` utility by launching a command prompt, typing in command below, and clicking [Enter]:
`C:\program files\basis\bbj - WDC:/techcon06/AuditTrail CreateDatabase.src`
- k. Choose an option in the BBj Sysconsole dialog box
 - 1) Option 1 populates the database.
 - 2) Option 0 exits the utility.
- l. Keep Enterprise Manager open and the “Databases” node expanded.

6. Setup the ESQL database

- a. Right click the Databases folder and select **Create New Database**.
- b. Enter `ESQL` as the Database **Name** field. It is important to enter the same case as shown for the demos to successfully access the database.
- c. Click on the browse button beside the **Data Files** field and navigate to the `\techcon06\ESQL` directory.
- d. Highlight the `data` with a single click and choose [Select].
- e. Click on the browse button beside the Dictionary Files textbox and go up one directory to highlight the `bbdict` directory.
- f. Click [Select], then [OK]. The ESQL database is complete.
- g. Exit Enterprise Manager.

Running the TechCon06 Demos

Introduction

BASIS engineers designed the TechCon06 demos to showcase new features in BBj and demonstrate possible applications of these features. They also arranged the demos into several categories and provided three methods to launch each demo.

Method 1. Launch a specific demo by selecting **File** from the top menu bar. Choose a category and then choose a specific demo. For example, to launch the “Payroll – Custom Object” demo, select **File->Language Features 1->Payroll – Custom Object**.

Method 2. Select a demo from the top toolbar. To launch the “Payroll – Custom Object” demo, select the third button from the left and then click on the **Payroll – Custom Object** item.

Method 3. Select a demo from the application tree. Click the leftmost tool button on the top toolbar. Expand the “Language Features 1” node and double click the “Payroll – Custom Object” node. The instructions for the specific demos use the application tree method.

1. Launching the Workbench Demo

Launch the TechCon demos by opening a command prompt and typing the following:

```
"\Program Files\basis\bin\bbj.exe" -WDC:\techcon06\Workbench\ -tT2 Workbench.src
```

Note: There is no space between WD and the path to the `techcon` install directory. The drive letter is required.

2. Payroll

These demos implement a payroll program three different ways. The first demo is a character-based demo. BASIS implemented the second demo using the AppBuilder tool in the IDE and BBJ custom objects in the third payroll program.

Payroll – Character Version

- a. Expand the “Language Features 1” node and double click the “Payroll – Character Version”. A BBJ Sysconsole window displays with a character-based payroll application running inside of it.
- b. Type `L` to list employees. Two employees are listed, “John Smith” and “Lisa Jones.”
- c. Type `R` to record a sale for a commissioned employee. Type `John Smith` and click [Enter].
- d. Record a large sale for John (i.e., “5400.68”) and click [Enter]. The number of unpaid sales for John will be reported. Click [Enter] to return to menu.
- e. Type `P` to pay an employee. Type `Lisa Jones` and click [Enter].
- f. Type `Q` to exit demo.

Payroll – AppBuilder

- a. Expand the “Language Features 1” node and double click the “Payroll – AppBuilder” node.
- b. Click on the [New Sale] button under the Commissioned Employee group box, type the amount `5400` and click [Update]. John’s unpaid sales increased by \$5400.
- c. Select “Lisa Jones” from the drop down list in the Pay Employee group box. Click [Pay]. Notice in Check group box, information regarding Lisa Jones’ payment appears.

Payroll – Custom Object

- a. Expand the “Language Features 1” node.
- b. Double click the “Payroll – Custom Object” node. This demo works the same way as the AppBuilder version of the demo.

3. CALL Timing Test

This demo is a timing test for the CALL verb. The number of generated variables and the number of iterations of the called program are configurable.

- a. Expand the “Language Features 1” node.
- b. Double click the “CALL Timing Test” node. Adjust the numbers, as desired, beside the fields “Generate a called program with how many variables?” and “Call this program how many times?”
- c. Click [Run] to start the test. When the test is complete, a line near the bottom of the window displays “Average Call Time =” with the results. Results will differ from one machine to the next.

4. BBjProgramInfo Demo

This demo demonstrates use of the BBjProgramInfo object to programmatically access information about BBj and BBx program files.

- a. Expand the “Project Development Tools” node and double click on the “BBjProgramInfo Demo” node.
- b. The top pane of the demo is a tree of folders under the `techcon06` directory.
- c. Expand the “AuditTrail” node and select the `AuditTrailTrigger.src` node. The contents of this program file will be listed in the bottom pane.
- d. Expand the “BBjProgramInfo” node and select “BBjProgramInfo.bbj.” Though this is a tokenized program, the source still shows up in the bottom pane.

5. View Source Notepad *(Windows only)*

This demo is a companion to the BBjProgramInfo demo, showing that tokenized files that are viewable with the BBjProgramInfo object are not viewable through an ordinary text editor.

- a. Under the “Project Development Tools” node, double click on the “View Source Notepad” node.
- b. Notice that the program “BBjProgramInfo.bbj” previously opened in the “BBjProgramInfo Demo” displays its tokenized binary code, but not the source.

6. BBjFileInfo Viewer

This demo demonstrates the use of the BBjFileInfo object. Depending on the type of file viewed, this demo shows the keys specified in a file, how the keys are configured, number of records in the file, size of the records, etc. This information can also be configured on VKEYED files through this demo.

- a. Expand the “BASIS DBMS 1” node and double click the “BBjFileInfo Viewer” node.
- b. Click the button in the upper-right corner of the window that has an opening folder icon.
- c. Select the `CUSTOMER.dat` file and click the [Open] button.
- d. View key number 3 for this file by selecting “KNUM 03” in the list box under the “Key Information” label. The information in the bottom two group boxes changes to display the information for this particular key. Attempting to modify keys on file types other than VKEYED files will result in the following error message “This capability is only available on VKEYED files.”

7. ByPass Extracted Record

This demo is an example of BBJ code that that report who is currently using a file when a file is busy. Unlock this file, if needed.

- a. Expand the “BASIS DBMS 1” node, and double click the “ByPass Extracted Record” node.
- b. Launch a second instance of the demo. When the second demo attempts to load the first record, a message box appears titled “Customer 000001 is locked” containing the message “Search for the user who is locking the record?” Click [Cancel]. The record is locked by the first instance of the demo.
- c. Position both instances of the demo so both windows are visible.
- d. In the window that is colored red, click the inner right arrow of the navigator control at the bottom of the window to go forward one record.
- e. In the first instance of the demo, click the inner right arrow of the navigator control at the bottom of the window to go forward one record. The first instance of the demo turns red and displays the locked error message. Click [OK].
- f. Enter `admin` for the user and click [OK]. In the second dialog, type `admin123` for the password and click [OK].
- g. After a dialog appears indicating that Customer 000002 is locked, click [OK] to force the other file closed.
- h. Once a message box displays that Customer 000002 is now unlocked, click [OK].
- i. In the second instance of the demo, click the inner right arrow in the navigator control at the bottom of the window to go forward one record. Another message box pops up.
- j. Click [Retry]. The second instance of the demo once again has an open connection to the file.

8. VKEYED File Comparison

This demo compares the size of a VKEYED file to other BBJ file formats where each file has a similar key structure and holds similar data.

- a. Expand the “BASIS DBMS 1” node and double click the “VKEYED File Comparison” node.
- b. Confirm all file types are checked and click the [Make Files] button to generate the files and display their sizes as the test runs.
- c. Change any of the values displayed at the top of the demo and running the file test again.

9. ESQL File Demo

ESQL is our new database table format. This demo compares the times required to complete various SQL operations on two BBJ table formats: MKEYED and ESQL

- a. Expand the “BASIS DBMS 1” node and double click the “ESQL File Demo” node.
- b. Click on [Run Tests] in the lower right-hand corner of the window. The tests run with progress bars for each test, displaying average times for the operation when the test is done.

10. ESQL Constraints Demo

One feature of the new ESQL file format is the ability to implement data constraints. This file format provides a way to make database tables aware of business constraints such as “invoice numbers always begin with ‘I’”, “no free shipping on orders under \$40.00,” etc. If the constraints set up for the database are broken, BBJ throws an error upon the insertion or an update that violates the constraint.

- a. Expand the “BASIS DBMS 1” node and double click the “ESQL Constraints Demo” node.
- b. In the “Select a Test” list box, select the very first item “00-Drop the table if it exists”.
- c. Press the [Execute SQL] button. This may display the message “ERROR: 77 Base table does not exist, 'CONSTRAINTS.’” in the “Error Messages and SELECT * Results” upper pane. This message or a “No errors” message is okay.
- d. In the “Select a Test” list box, select the second item, “01 – Create the table with constraints and checks”.
- e. Examine the SQL statement in the “SQL Statement to Run” pane. Notice that the second table field “Name” must contain the letter “e” and more than 5 characters.
- f. Press the [Execute SQL] button and notice the message “No Errors” in the “Error Messages and Select * Results” pane.
- g. In the “Select a Test” pane, choose “03-Auto Populate Identity Field”. A SQL statement displays in the “SQL Statement to Run” pane which inserts one record.
- h. Press the [Execute SQL] button. A “No errors” message should appear in the “Error Messages and Select * Results” pane, and the inserted record should appear in the bottom pane. The bottom pane also shows the results of “SELECT * FROM CONSTRAINTS” after conducting particular tests and after pressing the [Select *] button in the bottom right corner of the window. Note: the SQL statement in the “SQL Statement to Run” pane is modifiable. Pressing the [Execute SQL] button runs the modified version.
- i. From the “Select a Test” list box, choose “07-NULL Constraint: Error when name field is set to NULL,” then press [Execute SQL]. The “Error 77: Problem with NAME: Column is not nullable” message should display.
- j. Select “09- Column Check: Name field must be at least 5 characters long” from the “Select a Test” list box.
- k. Look at the SQL statement in the “SQL Statement to Run” pane and notice the problem that “Fred” is only 4 characters long. Change the name to “Freddy.”
- l. Press the [Execute SQL] button and see the “No Errors” message appear in the middle pane.
- m. Experiment with other constraint tests.

11. Forms Demo

This demo shows two methods for displaying and printing forms in BBJ. The first method uses the ‘FMTTEXT’ mnemonic. The second method uses BBJ’s BBJForm object.

- a. Expand the “Language Features 2” node and double click the “Forms Demo” node.
- b. Click the [Demo FMTTEXT] button.
- c. Click “Demo BBJForm.”

12. Visual PRO/5 6.x Themed & Visual PRO/5 6.x Not Themed *(Windows only)*

Both of these demos run the BASIS Configurator tool in Visual PRO/5. The first demo is themed and shows rounded edges in the tabs, “glowing” buttons, and other effects common to Windows XP. The unthemed version has the look of earlier editions of Windows.

- a. Expand the “Language Features 2” node and double click the “Visual PRO/5 Ver 6 Themed” node.
- b. Expand the “Language Features 2” node and double click the “Visual PRO/5 Ver 6 Not Themed” node.

13. Control Repaint and Grid Repaint

These demos are timing tests that exhibit how turning off repainting can improve BBJ's performance. The "Control Repaint" demo tests the speed of populating a window with a large number of controls with and without repainting, and the "Grid Repaint" demo tests populating a large grid with data with and without repainting.

- a. Expand the "Language Features 2" node and double click the "Control Repaint" node.
- b. Press the "Create Controls" button to run the test. A message box appears to report the amount of time it took to create the controls.
- c. Check the checkbox to disable repainting and run the test one more time. A message box again appears to report the amount of time the test took to run.
- d. Under the "Language Features 2" node, double click the "Grid Repaint" node.
- e. Press the "Repaint Test" button. A message box reports the amount of time it took to populate the grid.
- f. Check the "Repaint Disabled" checkbox to disable repainting and run the test again. Again, a message box reports how long the test ran.

14. Create Customer Database

This demo is actually a utility program that recreates and populates the CUSTOMER table in the ChileCo database.

- a. Expand the "Basis DBMS 2" node and double click the "Create Customer Database" node.
- b. Choose option 1, to drop, recreate, and repopulate the CUSTOMER table.
- c. After 1 is done running, choose option 0 to exit the program.

15. Audit Trail Viewer

This demo demonstrates using stored procedures for auditing file access performed on the data files used by the "Customer File Manipulation" demo.

- a. Expand the "BASIS DBMS 2" node and double click on the "Audit Trail Viewer" node.
- b. Uncheck the "Limit to last 10 records" and click [Refresh]. The results adjust to display more than the last ten records.
- c. Select "Write" under "View by Operation." The results adjust so that only "writes to database files" display. Keep the Audit Trail Viewer demo open and proceed to the Customer File Manipulation demo.

16. Customer File Manipulation

Customer File Manipulation demo is the program that accesses the files audited by the "Audit Trail Viewer." This demo manipulates the files three different ways: through regular filesystem access, through the BASIS JDBC driver, and through BBJ SQL commands.

- a. Expand the BASIS DBMS 2" node and double click the "Customer File Manipulation" node.

The checkboxes on the left side of the window describe different operations that can be performed on the customer file. Whenever the checkboxes are clicked code is generated that will read, write, or modify customer information in the customer file.

- b. Select at least one checkbox and click [Execute Code]. A message box pops up confirming that the code has been executed.
- c. Switch back to the Audit Trail Viewer by selecting the Audit Trail Log View tab. Click the [Refresh] button to update the log and monitor the changes as a result of the Customer File Manipulation demo.

17. BBJ SPROCS - Third Party App *(Windows only)*

This demo requires an installation of BBJ in the `c:\program files\basis\` directory and the `config.bbx` file in the `c:\program files\basis\cfg\` directory.

- a. Expand the “BASIS DBMS 2” node and double click the “BBJ SPROCS (Third Party App)” node.

The first tab is a Data Aware Grid bound to a database with stored procedures. In this grid, clicking on the inner arrows of the navigator control in the top-left corner of the window moves the cursor through the records one record at a time. Clicking on the outer arrows puts the cursor at the beginning or end of the recordset.

The second tab demonstrates running a query embedded in code. Click on [Load Grid] to load the grid with the results of the query.

- b. In the textbox beside the SQL label, change the query to
`Select City from Customer where Last_Name like 'W%'`
- c. Click on [Load Grid] again. The third tab demonstrates a stored procedure that performs a search based upon the criteria entered in the bottom two textboxes.
- d. Click the [Load Grid] button.
- e. Change the “Search for Name” textbox to “De” and the “Maximum Results” textbox to 1.
- f. Click [Load Grid] again.
- g. Change the “Max Results” textbox to “1000” and click [Load Grid] again. There are only 2 matching records.

Tracing the TechCon06 Demos

The demos include the Application Monitor that displays in a bottom pane a code trace of the demos. The code trace shown in the pane corresponds to the demo with the active tab. The Application Monitor will trace any of the demos, but for this example, we will trace the “Payroll – Character Version” demo.

- a. Expand the “Language Features 1” node and double click the “Payroll – Character Version” node.
- b. Launch the “Application Monitor” by clicking the second button from the left in the top toolbar. The pane on the bottom displays the code that is currently running for the Payroll application.
- c. Adjust the refresh rate (how often the lines from the trace appear) by typing a whole number into the “Refresh every # seconds” textbox. A faster refresh rate displays more lines of code as they execute but will put a higher load on the processor. A slower refresh rate will cause the trace to miss lines.
- d. Click [Enter] after typing a number into the textbox. The “Max Lines” textbox can be set for how many lines it is possible to scroll back through in the pane. Adjusting the number higher, it is possible to review more lines, but it also takes a more memory.