

Visual Lint Build Server

Advanced incremental command line code analysis

Setting up a build server to run code analysis on your code is no trivial undertaking.

Not only can it be messy and slow, but the analysis itself usually runs as a single threaded task, leaving other CPU cores on the build server idle no matter how long it takes to run.

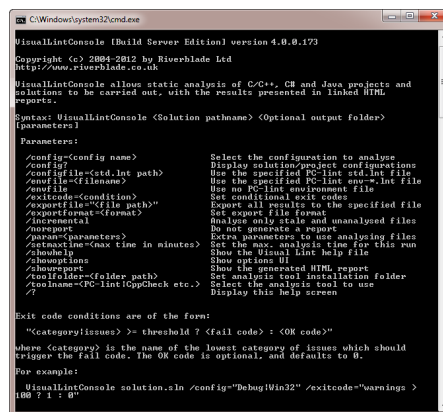
Needless to say, this can be a tedious, error prone and frustrating process which can frustrate your efforts to maintain or improve the quality of your software.

Introducing Visual Lint Build Server

Visual Lint Build Server changes that by bringing the incremental, parallelised analysis capabilities of Visual Lint to build and continuous integration server environments.

Visual Lint Build Server works in **exactly** the same way as the interactive versions of Visual Lint, but via a command line rather than a graphical user interface.

For ease of use, the command line interface is based on that of LintProject Professional.



```
VisualLintConsole (Build Server Edition) version 4.0.0.173
Copyright (C) 2004-2012 by Riverblade Ltd
http://www.riverblade.co.uk

VisualLintConsole allows static analysis of C/C++, C# and Java projects and
solutions to be carried out, with the results presented in linked HTML
reports.

Syntax: VisualLintConsole <Solution pathname> [<Optional output folder>]
[Parameters]

Parameters:
/Config<config name>          Select the configuration to analyse
/ConfigFile<std_int_path>    Display solution/project configurations
/ConfigFile<file_name>       Use the specified PC-lint std_int file
/EnvFile                      Use the specified PC-lint env-int file
/NoPCLint                    Use no PC-lint environment file
/ExitCode<condition>        Set conditional exit codes
/ExportFile<file path>       Export all results to the specified file
/ExportFormat<format>        Set export file format
/Incremental                 Analyse only stale and unanalysed files
/NoReport                    Do not generate a report
/Param<parameters>          Extra parameters to use analysing files
/SetMaxTime<max time in minutes> Set the max. analysis time for this run
/ShowHelp                    Show the Visual Lint help file
/ShowReport                  Show options UI
/ToolFolder<folder path>    Set analysis tool installation folder
/ToolName<PC-lint|CppCheck etc.> Select the analysis tool to use
/?                            Display this help screen

Exit code conditions are of the form:
"Category:issues" >= threshold ? <fail code> : <OK code>"
where Category is the name of the lowest category of issues which should
trigger the fail code. The OK code is optional, and defaults to 0.

For example:
VisualLintConsole solution -ln /config="DebugWin32" /exitcode="warnings >
100 ? 1 : 0"
```

Industry standard code analysis tools

Visual Lint supports a variety of analysis tools including **PC-lint**, **PC-lint Plus**, **CppCheck**, **Vera++** and **cppint.py** for C/C++, **FxCop** for C# and **FindBugs** for Java.¹

Adding support for additional languages or analysis tools is usually straightforward, so please contact us if you are interested in a particular language, project type or analysis tool.

Incremental and time limited code analysis

Visual Lint Build Server supports **both full and incremental analysis**—the latter ideally suited for use within continuous integration environments such as CruiseControl.NET, Jenkins and TeamCity in which rapid builds are essential.

If analysis time is critical you can even set a time limit on each run.

Accelerated code analysis

Visual Lint Build Server automatically **takes advantage of all CPU cores** on the host, delivering an immediate reduction in analysis time by comparison with conventional build server integration solutions.

If you need to reduce the analysis time further, you can even **distribute the analysis among idle machines on your network**, potentially reducing analysis time by a factor of 20.²



Key Benefits

- Parallelised multicore analysis of C, C++ C# and Java codebases using industry standard code analysis tools
- Analyse projects for Microsoft® Visual Studio®, Eclipse, CodeGear C++ and others, including embedded environments such as QNX Momentics and Texas Instruments Code Composer
- Analysis configurations automatically take project settings into account
- Optimised for continuous integration server environments
- Choose from full or incremental code analysis of complete projects, workspaces and solutions
- Set an optional time limit on incremental analysis
- Analyse your code even faster using idle machines on your network²
- Produce customisable reports giving a summary of the analysis results
- Export analysis data in a variety of formats for subsequent analysis, review and action
- Included as an optional component in the Visual Lint installer
- Licenced per server; optional priority support subscriptions available if required.

¹ Commercial analysis tools supplied separately; freeware and open source analysis tools (e.g. CppCheck) may be freely downloaded

² Requires IncrediBuild XGE interfaces extension package. Please visit www.xoreax.com for details.

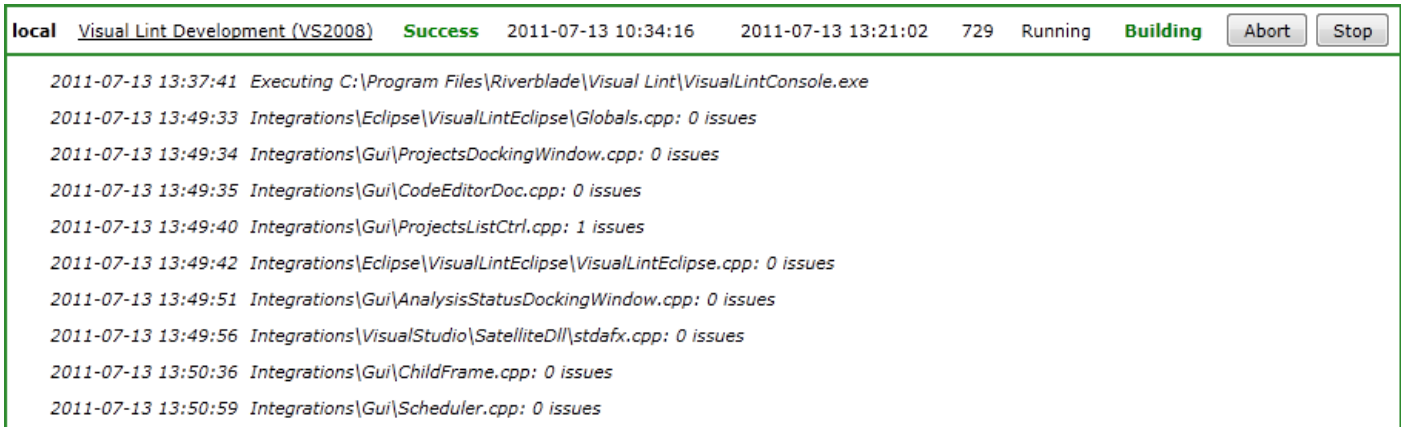


Visual Lint Build Server

Advanced incremental command line code analysis

Straightforward integration with build server environments

The command line interface of Visual Lint Build Server is based on that of LintProject Professional, and can easily be integrated into build and continuous integration server environments such as CruiseControl.NET, Jenkins or TeamCity.



```
local Visual Lint Development (VS2008) Success 2011-07-13 10:34:16 2011-07-13 13:21:02 729 Running Building Abort Stop
2011-07-13 13:37:41 Executing C:\Program Files\Riverblade\Visual Lint\VisualLintConsole.exe
2011-07-13 13:49:33 Integrations\Eclipse\VisualLintEclipse\Globals.cpp: 0 issues
2011-07-13 13:49:34 Integrations\Gui\ProjectsDockingWindow.cpp: 0 issues
2011-07-13 13:49:35 Integrations\Gui\CodeEditorDoc.cpp: 0 issues
2011-07-13 13:49:40 Integrations\Gui\ProjectsListCtrl.cpp: 1 issues
2011-07-13 13:49:42 Integrations\Eclipse\VisualLintEclipse\VisualLintEclipse.cpp: 0 issues
2011-07-13 13:49:51 Integrations\Gui\AnalysisStatusDockingWindow.cpp: 0 issues
2011-07-13 13:49:56 Integrations\VisualStudio\SatelliteDll\stdafx.cpp: 0 issues
2011-07-13 13:50:36 Integrations\Gui\ChildFrame.cpp: 0 issues
2011-07-13 13:50:59 Integrations\Gui\Scheduler.cpp: 0 issues
```

Direct support for a variety of project, solution and workspace file types

Visual Lint Build Server can read project, solution and workspace files for the development environments (e.g. Microsoft® Visual Studio® and Eclipse) supported directly by Visual Lint plug-ins.

However, it can **also** read project files for environments (for example CodeGear C++, Atmel Studio or VisualDSP++) for which a Visual Lint plug-in is not currently available. Please contact us if you are interested in support for a particular project type.

Flexible output data formats

As well as generating analysis reports, Visual Lint Build Server can also export data in a variety of text, CSV, XML and user defined formats. If you need assistance exporting data in a particular format please contact us.

Make the build go red if the quality of the code drops

Visual Lint Build Server allows **conditional exit codes** to be set based on the results of the analysis. So if you need your build server to fail the build if the number of warnings exceeds a predetermined threshold, you can easily do so.

Effective code analysis

Visual Lint gives you the tools you need to use code analysis *effectively* within your team.

By deploying **Visual Lint Build Server** in conjunction with the interactive versions of Visual Lint, you can ensure that everybody on your team is fully aware of where the trouble spots in your code are —and thus where you might need to focus your attention to ensure that the quality of your codebase meets your standards.

For full details of the capabilities of Visual Lint, please visit www.riverblade.co.uk/products/visual_lint.

About Riverblade:

Riverblade is a UK based independent software vendor founded in 2004 and specialising in code analysis tools and development environment integration.

Additional information about us and our products can be found at www.riverblade.co.uk.

Riverblade Ltd, Platinum House, 23 Hinton Road,
Bournemouth, Dorset BH1 2EF, UNITED KINGDOM.

Tel: +44 (0)20 3289 7997 Email: enquiries@riverblade.co.uk

Registered in England no. 5187132.

VAT Registration no. GB 847 0045 38.