
Language resources for 1.0.31.0.3 information entered in setupAdded ndocremoved ndocSD-293:Start Page and external tools.updated splashscreen/version number and NDoc to 1.3autoindent for set statementFixed SD-230:Wheel mouse use in CC dropdown.turned on project scout icon engine.ported some speed improvments from corsavy tree.Fixed SD-208:Non compliant primitive Datatypes corrupt the FormsDesigner.Fixed SD-246:Exception on negative value for Minimum property in NumericUpDown.Fixed SD-292:Scrolling behavior when selecting text with the mouse. (last fix was SD-247).Fixed SD-292:Scrolling behavior when selecting text with the mouse.Fixed SD-290:Resetting an property to an invalid value causes an exception.Fixed SD-274:Output path not remembered correctly for different configurations.fixed bug.(280)CodeCompletion: checking file extension is not case sensitive anymore.Fixed SD-275:Cursor position not remembered.Fixed SD-275:Cursor position not remembered. (Still not remembered, but textarea gets focus.)Fixed SD-288:Adding a MainMenu leads to Handles clause error.Fixed SD-282:User controls do not show up in Custom Components tab.Fixed SD-280:Exception in sidebar when component is uninstalled.Fixed SD-285:Ctrl+U / Ctrl+L combinations - restrict to selection.Fixed SD-289:SD 1.0.2 build 1726a doesn't show custom controls.Fixed exceptions in class browser and folding when there are types without region information.Language bindings can now add new parsers: Removed hardcoded file extension detection in DefaultParserService.fixed SD-287:Referenced assemblies are locked.Fixed SD-286:Missing project dependencies cause an exception. (Corsavy 140)Fixed SD-279:Clipboard Ring Scrolling Exception.Changed main menuFixed breakpoints pad.Changelog and language resources for 1.0.2a QFEwarning in line 220: CS0162Language resources update / setup 1.0.2a QFEFixed forms designer bugFixed 'flickering' of newly created viewcontents.Removed #ZipLib (#ZipLib has now an own repository).Checked in Corsavy patch rev. 60:Added forum bugfix from forum user 'mward'.#Zip tar magic no longer checkedSet Position of the ProgressBar to the end of the panelCheck whether to load referenced assemblies. This change shouldn't affect #D at all. The property is set to true. The reason of change is ETM.Checked in Corsavy r22 patch:Added forms designer bugfix from forum user AaKaa.Checked in Corsavy r19 patch:Added namespace lookup bugfix from #D Forum user Lelli.Fixed linker invoking bugFixed bug in project converter: converted project has same project type as source project Fixed bugs in AbstractASTVisitor: InvocationExpression and IndexerExpression didn't visit their child parametersFixed stupid DefaultParserService 'optimization'. Removed NRefactory (it's now part of CorsavyApplied bugfix from the #Dev forum (user UncleSam).Updated language resource files and assembly info files, plus change logSetup script version number bumped to 1.0.2removed some debug code.Fixed SD-272:No form designer tab when combine name contains a space.Fixed little bug that prevented .IL assemblies to be added to the code completion data.fixed some bugs in the assembly scout 'structural view' and made it faster.changed general text editor options (set fontsize dropdown anchor).fixed caret sizing bug.Monospaced fonts are now drawn bold inside the font chooser combobox.Changed tab order in text editor panels.Fixed SD-250:Syntax highlighting background color not used for whitespace.Fixed SD-254:Can't parse very large integer literals (c#). Put the fixes from Nikola in NRefactory.Fixed SD-260:Crash when converting object creation code. (Not 100% conversion - keep in mind that the parser is 'dumb' without semantics check. But at least it will be correct for some cases and it doesn't crash anymore).fixed font size bug (negative values).Tried to reimplement the correct behavior when presseing the up/down key: the desiredColumn in the cursor now represents a drawingPos. Please test all about it you can imagin. I hope it works with folding too.Changed text editor option dialog (general) a bit.Fixed General Text Editor Panel gui a bit.Added MouseWheel zoom option. Did some translation. Changed the way of font selection (no font dialog anymore). Still having a problem: How to determine if a FontFamily is monospaced ???Added ClownFishException (see About dialog)Fixed 'grid size bug'Cursor position is now corrected when indenting multiple lines, this fixes problems with inserting code templates. Fixed xml documentation in TypedCollection template.moved SharpDevelop/help to SharpDevelopHelpFixed folding 'drawing' bug (the right border of fold markers sometimes were over-drawn). New Features : Ctrl+Mousewheel now zooms text. New text editor option: Bracket marker can now swichtched to mark after/before bracket.Added XML smart indentation.Worked on Add-In scout. (Now you can SEE the add-in tree)started a real addInTreeView. Icons and descriptions of the nodes are still neededmodified the AddInScout to follow more our coding style-Made the addin scout a bit nicer.Fixed SD-268:Assembly scout crashes when pressing 'back'.Set versio number to 1.0.2Fixed SD-269:Folding crash.ProjectImporter now adds the namespace to the resource file names.ProjectImporter now imports non-buildable files.filereader: ability to specify the encodingCompiler result can now be null (easier for small backend bindings).Fixed FireSelectedItemChanged (http://www.icsharpcode.net/OpenSource/SD/Forum/topic.asp?TOPIC_ID=6851)App.config is now automatically copied to the output folder.1.FileCommand - SaveAll uses ICustomizedCommands 2.DefaultWorkbench, Workspace-Layout - Closing problems and bugs 3.DefaultFileService - OpenFile, GetOpenFile, JumpToPosition do not ignore SubViewContentsSD-267: update DirectX referencesFixed SD-253.corrected path in build.batFixed build error.fixed bugs in DefaultWorkbench.Combine is now closed on #D exit.Fixed SD-262: Exception in error drawerbase.dispose is now called at the beginning of SdiWorkspaceWindow.DisposeAdded disposed check to activeworkbenchwindow property in sdiworkspacelayout.fixed in resource editor: <http://icsharpcode.net> bug that error drawer didn't draw '&' character, added System.Drawing to UserControl template.patch from the forum: loading of MC++ dlls works now. So DirectX classes have finally code completitionfixed SD-248: Warnings are always treated as errorsbuild.bat now no longer alters the assembly infos. Added new build.bat to trunk\ which builds the language resources, updates the assembly infos and runs build#Zip add new test assemblyfixed SD-254: Can't parse large integer literals (bug still applicable to NRefactory)#Zip project - more tidying#Zip delete old tests II#Zip delete old tests#Zip tests removed from main combine#Zip fix for Compact framework + several small changesGZipOutputStream now has SetLevel capabilityre-committed rev. 1636, fixed the showstopper bugreverted NewFileDialog to rev 1635 due to a showstopper bug introduces with revision

1636Setup readme updatedpre-install task moved to vbs tooCleaned buildfiles a bit.Post Install Tasks bat->vbs call in installer scriptbat->vbs conversion for post install taskschange logSetup for 1.0.1 (rev# change only)Language resources for 1.0.1Fixed exception on #D close.fixed bug in some conditions. (added null check)SD-249 fixed: string., int. etc now shows the static membersFixed NullReferenceException in OpenWindowStateCondition on closing SharpDevelop.Creating new file in a subfolder now uses the folder name in the namespace, so a new class in the folder "Controls" of the project "ControlLib" will have the namespace ControlLib.ControlsRemoved unnecessary source file.Added assemblyinfo.cs files to regxptk and resource editor.Fixed bug in naming of new classes where the filename contained invalid chars."Fun language" resource deletedLanguage resources update; Serbian added, build file and xml file sorted alphabeticallychanged about dialog a bit.Added "using System.Drawing" to the C# WinForms project templates. Fixed small indentation bug. AutoAdd files now ignores hidden files.Changed splash to 1.0.1. Added arabian and servian flag.Added NProf 0.9alpharemoved nprofAdded assemblyinfo.cs to some backend bindings.debugbuild no longer changes the AssemblyInfo.cs files.#ZipLib has now own versioning sheme in build file.changed the assembly info revision set routine, now 3rd party libraries are skipped.Added script to buildfile: Set version information to ALL AssemblyInfo.cs files.Search Type now searches type with fully qualified name instead with 'class name' only.updated vb.doc integration (0.4); there is now a menu item 'run vb.doc' in the project menu for vb.net projects, this makes more sense to me than the former wayrankspecifier fixFix bug in start page: the cmbx-files were not closed after the name has been read.Fixed NRefactory C# & VB.Net array 'rank' bug.Improved VS.NET project exporter: - many more properties are exported - references to most-used system dlls are created - project references and assembly references are exported Fixed bugs in VS.NET project importer: - "threat warning as errors" was negated on import - .csproj and .vbproj files were not shown in import dialogrank list bugnothing changed.Changed ReDim statement in NRefactory. Now the redim clause is an invocation expression.redim statement bugfixFixed SD-244:Code completion is not working inside ELSE clauseFixed SD-218:Undo form designer operations in code.Fixed SD-241:Code generation doesn't work for own classes.Fixed SD-209:Folding inconsistency.Translated some strings in the code completion database wizard (choose location msgs/finished panel and replace/mark all messages in the search&replace.Disabled multiselect in alt+ins listview.Fixed code autoinsert bug where single 'option' wasn't shown.compilation errorFixed SD-243:Code completion overload insight window scroll arrows not workinguse linemanagerhorizontal scrollbar calculation algorithm ignored invisible (folded) linesSD-229Fixed SD-186: Permission problem in VS.NET importerFixed SD-239:Auto code generation exception.Fixed SD-242:Switching File Mode to VS-C# from C#, quotes will 'vanish'ReferenceService ignored all References that were not part of the host.Container.Components. That's why all properties with DesignerSerialisationVisibility.Content different from collections were not serialized Checked in Destructor.csdeleted loop-controlvariableexpression (again).Fixed SD-236:Code generation list has duplicate entries.Fixed SD-232:Toolbox Scrollbars-Fixed SD-224:Destructors can't be folded.comment trackingChanged VB.NET grammar. Removed LoopControlVariableExpression.enums in interfaces are handled by NonModuleDeclaration; loopcontrol variable doesn't can't be an arbitrary expressionenums in interfaces are handlely by nonmodule declarationFixed SD-231:Save / Save All commands context sensitive.Fixed SD-238:WinForms template namespace issue.disabled the code completion for numbersWorked on NRefactory.worked on NRefactory.SD-216 fixed: code completion for arrays like a[0].Perl token generation script now generates nunit lexer tests. Added lexer unittests for VB.NET and C#.Enable{Cut,Copy,Paste,Delete} uses now OleStatus callbroken buildwith the code generation list being sorted some items disappear. Looks like a windows forms bug in combination with Begin- and EndUpdate()updated the parser token generation script.SD-215 fixed by adding System.Object, System.Enum, System.Delegate and System.ValueType to the BaseClass Method in the DefaultParserService. Now the members of System.Object should be in every codecompletion list.SD-210 fixed: no code completion for type keywords like int or stringSwitched to 'old' #Coco in NRefactory (configurable in the gen.bat). The new Coco/R produces a parser that fails some unit tests. The old version is more reliable.NRefactory unit tests now working.Updated SharpCoco to Coco from Jun 28, 2004.worked on NRefactoryTestscode generation list is now sortedReDim statement now recognizes 'Preserve' modifier.Checked in NRefactory (former known as CsVbRefactory). NRefactory will be the #D parser layer for the .NET 2.0 version.adaptations for new sharprefactoryvbspecial tracking in #rvb is now similar to the tracking in #r; grammar didn'T accept attributes on return types of functionsmissing file to previous commitSD-234: Opening single quote does not suppress code completion; code reformattingSD-226: x = New Double(3)() {} didn't parseBuildHelpIndex call instead of minisetup, #ziplib also added to GAC (as well as removed on uninstall)Fixed "else if" indentation bug introduced in rev. 1552.SD 221: MiniSetup Changesmoved minisetupSD-223: Adding Projects with wrong file formatSD-220: About Box, Change logsrc\3rdParty deleted, build file updated. Decision has been reached that third party addins in the base distribution create more support / maintenance headache that can be gained. Third parties will be supported by a Manage Addins feature in the future.Setup now has post-install and pre-uninstall tasks managed as a batch file. This fixes the NUnit GAC registrations. GacUtil2 source code checked in.Fixed smart indentation and inserting of closing brace in auto indent mode.fixed SD-213: Wrong CC window location with horizontal rulerChange log updated, translations updatedNow Ctrl-Z needs to be pressed only once to undo a line feed.Fixed bug that the output window didn't show the correct number of errors/warnings.#Zip updated documentation#Zip failure to set flags in ZipFile correctedWorked on CsVbRefactory.Worked on CsVbRefactory.Readme.rtf is now recognized.Change Log for 1.0Fixed little typo..resources for updated .mdb (final sweep of missing strings)Fixed a bug in the DeclarationViewWindow (elements without description like namespaces didn't make the window invisible)Added new translation strings for various errors (the language resource need to be built!!!)Added error handling to CreateDirectory call

from NewProject dialog..resources files for changes .mdbchanged start page iframe generation.litte bugfix in 'close all but this' command. (case: dirty file, and 'cancel' in the do you want to save dialog)Little fix for the last correction.Setup now registers both .framework and .core assemblies of NUnitcorrected component naming bug.Fixed comboBox forms designer issue when you doubleclick on a comboBox that the caret line wasn't correct.Set focus back to main view after closing help (to fix a forum bug - I wasn't able to reproduce it.)fixed bug in xml formatting strategyAdding a menu item now sets the dirty flag.C# parser of CsVbRefactory is now working (and it's more than twice as fast as #Refactory)#Zip now reflects recent changes-Modified SharpCoco it now accepts template modifiers.added parser ATG files to CsVbRefactory.Updated DockPanelSuite to 0.9.2.9Execute after build script should now work in all cases. (Removed unnecesary code duplication, added some translation strings).Optimized CsVbRefactory C# lexer (I think it's now fast enough). Applied some optimizations to the VB.NET lexer too - but this can be made much better.ESC key now focuses the current view. (Now #D should be really be usable with keyboard only). Fixed little graphic 'glitch' that positioned the caret of the textarea to the upper left corner sometimes after opening the text editor.The creation of more than 1 file with a new file wizard should now work when the wizard is started as 'add new file to project' command.VB.NET lexer now working again (CsVbRefactory).Optimized the lexer in CsVbRefactory - it's much more faster (more than 2 times).Fixed 'select disposed object bug in propertygrid' in forms designer.#Zip added missing exceptions-fixed build file. (authors.xml validation)#Zip Exceptions cleaned up. Bug in Tar with all names being lowercase fixedworked on StartPage/Authorsremoved authors from about dialog (it's on the start page). Removed authors.xml.initial 1.0 rev of readme-Further updates, readme is structurally complete (not content-wise)removed how you can help (it's now inside the wiki).linked how you can help section to our wiki.linked #D authors to our wiki.Initial updates in doc folderRemoved items in favor of links to Wiki topics that are definitely maintained.worked on CsVbRefactory.added C#/VB.NET lexer to CsVbRefactory.fixed little bug in configure sidebar dialog.added the primitive types to the new completionfixed some translation strings.updated author conversionSetup updated for 1.0added System.Data.OracleClient to help definition file (the assembly is included in .net framework 1.1)Added some comments about the possible children in VBfixed little bug in HighlightColor.cs.- Automatic addition of closing quote in VB.NET on pressing enter - On pressing enter in a complete string in both C# and VB.NET the string constant is broken up on two lines.fixed little bug in caretline marker routine.fixed little bug in the text area view.C# syntax highlighting now recognizes verbatim double double quotes correct.Fixed "File already in use"-bug in HighlightingEditorchanged C# highlighting back.text editor now supports background colors for syntax highlighting colors.Improved VBFormattingStrategy, bug fixes in the VBStatement regular expressionsremoved some debug writelines.Fixed 'last panel closing can't be aborted' bug.Fixed bug that prevent the pads from closing. Now it's possible to close pads and show them again.Fixed bug in Highlighting editor.Added a new test (if the classes are INode) and finished to look wich children are possible in C#.Fixed little bug in text area (double click on fold marker unfolded it but didn't update the vertical scrollbar size).Added .csprj / .vbproj (VS.NET project file) converter.Optimized some string operations in the C#/VB.net backend (string.concat instead of + or stringbuilder).removed the same code duplication in the vb.net backend.Removed some code duplication in the C# compiler backend.changed /nowarn option (only is given to the response file when there is a specified warning) - the 2.0 beta compiler complains about an empty /nowarn option.Project converter: + don't convert resource files, copy them + don't change extension of resource files + resource files can live outside of the project base directory Fixed possible exception in the com reference panel.More VB.NET <-> C# conversion bug fixes. (abstract properties and using-statement)changed splashscreen.Updated splashscreen to 1.0Fixed SD-169:About Screen: text disappearsFixed SD-196:Visibility of undocked panelsAdded work-around for SD-89: Special characters issue in conversionFixed some bugs in C# -> VB.NET conversionHorizontal ruler is now working again.Line marker is now working again.Fixed SD-198:Scrolling over the end of the document whith folded regionsadded close all & close all but this command in the open file tab strip. (like in firefox)Added conversion done message in the project exporter.fixed the new completion bug SD-101 for VB too.the ctrl+space for VB did not work because the parser didn't delegates it to the resolver. It should work now.Cursor position is now adjusted if indentation was changed.Removed tab from end of line in templates. Added new indentation system for better auto-indentation of multiline statements and quicker indentation of whole files.Fixed SD-146:CC for 'new' popping up in commentsFixed SD-134:Generated tabs remain tabs (with 'Convert tabs to spaces' option set).removed some debug messages.fixed SD-101: Enhancemnet possible: Only show Types that are implementations or Subtypes of the Type from the declaration.Fixed SD-188:AxWebbrowser issuefixed SD-48:Code generators ignore 'Start code block on the same line' option.fixed bug in forms designer that occurred in the converted nunit 2.2 project.disabled export menu item if no combine is loadedFixed SD-205:Exception for Alt+Ins.#D now compiles again (.NET v2 resource conflict).Fixed SD-197:Exception for project system.Fixed SD-203:Task List - TODOs show up for old file name after renaming.Fixed bug in project browser. Added files were added with a 'wrong' (too few items) context menu.Fixed SD-204:Non-compileable files auto-added to compilationadded suggested text editor changes from Nathan Allan.Added basic debugger infrastructure.Speed up text editor drawing. Now the brushes & pens are stored in a global helper class (brushregistry) and are not re-created during draw. This speeds up the drawing especially in 'portrait' mode.little change in build.batFixed little vb.net issue.translated 'Convert' project context menu; Show message after successfull project conversionAdded changes from David Srbecky. Added ViewOpened/-Closed events to the workbench.Fixed SD-165: SearchVariable in Resolver now returns a returnType with the FullyQualified-Name.Fixed SD-201:Nested classes in form: No root serializer found and SD-199:Switching to design view fails with nested classesFixed SD-163:VB.NET date literals not working.fixed SD-138: no code completion for constupdated to nunit 2.2Fixed

SD-189:F1/Dynamic Help throws exception at end of file.VB.NET->C# conversion fixes: - Lexer now recognizes date literals - Unicode charset modifier was fixed - named arguments should be using := only - 'Handles MyBase.Error' was causing error getter/setter autocompletion improvementsFixed SD-193:'End' doesn't parseSD-191: power operator crashed conversionSD-190: multi monitor and tip painterFixed SD-192:VB.NET event conversion (Implements syntax version)Fixed Spanish (Mexico) bug re: wrong codeVB.NET <-> C# conversion updateadded new author conversion to svn2changelog toolupdated changelog.RC3 setup fileFixed SD-180:Clicking on a todo item doesn't take you to the sourcecodeSolid text markers should now work (text editor).RC3 splash screenadded lithuania flag.Translations updated; flag for Lithuania missing, thus might crash language selection panel in Options dialogadded C# <-> VB.NET conversion fixes from Nikola Kavaldjiev.ILAsmBinding: added possibility to choose compiler and runtime framework.Fixed SD-173.Fixed indenting of ?: operator. Now warnings are not threatened as errors in new projects.#ZipLib Small corrections to docs/comments and project settingsfixed licensed controls bug (forms designer)/little additionfixed licensed controls bug (forms designer)#Zip ZipFile backtrack on excluding spanned archives till more investigation has been done#Zip ZipInputStream remove invalid test in GetNextEntryReverted copy to clipboard workaround. Changed GC.Collect() to GC.Collect(0) in CodeCompletionDataProvider.Fixed build error.text editor now inserts the >correct< line terminator (taken from Environment.NewLine). Too much people complained about the plain '\n' because too many windows editors don't recognize it correctly.License context is set to DesignTimeLicenseContext for components created by the forms designer.Small fixes for indenting of switch. Added workaround for copying text to notepad. Moved SearchBracketForward and SearchBracketBackward to the FormattingStrategy as it is language dependent. Added special SearchBracket for VB.NET. fixed SD-177:Sorting per category in property grid.Fixed SD-170:Textbox property PasswordChar not roundtripped (1355).Fixed SD-181:VB.NET form designer 'forgets' custom colors.Fixed SD-185:NullRef Exception when selection controls with 'shared' event handlerRewritten switch-case indentation: multiple case-lines for one block are indented correctly now. Rewritten SearchBracketForward and SearchBracketBackward: Escape sequences and verbatim strings are now supported, // in strings is not recognized as comment anymore.WordCountDialog - fixed NullReferenceException CSharpFormattingStrategy - SmartIndentation now strips comments before indenting. InString/InComment functions now support character literals, escape sequences and verbatim strings. Optimizations in ReflectionReturnType.cs (NProf said much time was wasted there) ProjectImporterExporter: Checking if copying the file is really needed -> now you can import/export a project without creating a new directory. Now using Path.Combine in project importer, so references can now be correctly imported (a backslash was missing before) --> but the XSLT functions don't have the permission!!!! RecentOpen.cs: comparing the paths is now case-independent. ILAsmCompilerManager: new compile target: NetModule File extension is now changed to dll/netmodule if the CompilationTarget is Dll/netmodule. You don't need to set your PATH, #Develop finds ilasm.exe automatically. MiniSetup now only starts HelpConverter if some source help files were changed. #ZipLib build script only rewrites AssemblyInfo if it was changed.added vb.net compiler patch from Daniel Grunwald: fixes line calculation problems added missing filepatch from the forum: fixes wrong line calculation in gutter margin (thanks to ynik)added option strict and option explicit compiler parameters for vb.netadded compiler version selection panel for vb.net#Zip changes to allow compilation with compact framework#Zip Change TarEntry.cs so it compiles with 1.0 as well as 1.1added Region to list of vb.net keywordsfixed possible bug in AbstractASTVisitor.csfixed compilation error.checked in changes from alex.#Zip ZipEntry some constructors made internal#Zip Gzip using clause in sample comment correctedTranslations updated, Lithuanian added#Zip: Deflater TotalOut int->long to stop bug with very large archives where compressed size is larger than size. Files were invalid rather than throwing an exception#Zip Bumped version number#Zip.ZipFile stream checking loophole closed, MBCS flaw worked around#Zip: Updates to samples + changes.txt#Zip: ZipFile can get a stream by index. Add further parameter checks. Small tidy upsChecked in changes from alex.Checked in changes from Alex.Language resources refreshed, Dutch is now included#Zip: BZip2 wrong text used when throwing exception fixedImages from "edited" designer were lost, when user switched to source code and backUpdated language resource filesBug fix: TypeDescriptionFilterService.cs host wasn't checked against "null" DefaultSerializationObject.cs -> Mike has "included" bug on 8.6.2004. The lines are away nowNew Class Wizard can generate interface and implementation in separate files Fixed SD-170:Textbox property PasswordChar not roundtripped (1355)Fixed SD-176:Text editor weirdnessRefactored Class Wizard. I hate switch statements because #D cannot autocomplete them ;-) Checked in some uncritical changes from Alex.fixed cc window location problem with multiple monitorsLanguage resources refresh: Romanian and French are now active (more to hopefully be added soon)Fixed VB.NET resource naming bug.optimized parser thread a bit.Component Tray context menu localization fixed.reverted #Refactory to 1300 - 1st retryreverted #Refactory to 1300added missing file headers; updated project file#Zip some summary comments modified to aid readabilityadded new toolbar item (configuration selection combobox in toolbar)#Zip Deflater.GetLevel not just for test versions nowExplicitly set length of entry when compressing#Zip decryption bugs fixed. More comment updates. Minor additions and bug fixesSD-91: Adding configuration selection to toolbarchange log for RC2made changes (flag) for Spanish (Mexico)language resource files for RC2added mexican flag.Fixed in SD-166:CC not working for e. (1332)Fixed SD-167:Find in Files: Repeating search yields errorFixed SD-168:#regions (or collapsed code) gets mangled when deletingupdated splashscreen.Fixed SD-148:File Scout exceptionfixed bug SD-152:Font settings not persisted properlyFixed SD-151:Exception syntax highlighting editor (1280)CreateCodeCompletionDataProvider (we need to generate data in other way as Default does)reverted optimization line buffer in DefaultDocument -> produced errors elsewhere.Applied non breaking changes from alex.Replaced the RichTextBox in the StandardHeader and CodeTemplate panel with a TextBox

that accepts Tabs AND Returns. DefaultDesignerHost.cs: bug fix FormDisplayBindingBase.cs: bug fix SharpDevelopTextAreaControl: someone has built in CodeCompletion only for Csharp and VB.NET. Now, there are properties that allow to start CodeCompletion from other languages as well DefaultParserService: mike has deleted ParseableContent.ParseableText, this is "NECESSARY" for me Tab now works in the edit standard header panel* added gacutil2.exe to src\tools * changed tools.build accordingly * /i and /u respectively execute during install and uninstall for nunit.framework.dll ResourceService.cs: localizable user strings, user icons* added file associations (cmbx\prjx) on install * changed compression from bzip2 to lzma * bumped setup rev to RC2Fixed little bug found by Alex.DockPane: this bug fix should be done by Weifen as well AbstractProject: IsFileInProject haven't worked for "non" file views while Saving fixed bug in forms designer resource file generation (they're now only added when they're stored). So no 'empty' resource file compilation failed messages.#Zip streams comments updated and problem spot with encryption bug notedFixed SD-161:Custom Controls various bugs / showstoppers#Zip BZip now throw NotSupportedException in many cases#Zip Small updates to BZIP#Zip Compression comments updated#Zip Zip tests updated#Zip ZipEntry version to extract is now read-only and dynamically evaluated when writing to zip files#Zip tar classes updated, ce compatibility, summary comments, failure on extract to write protected file...#Zip sz demo can now restore dates when extracting#Zip sample sz updated#Zip tar example updatedhopefully reverted all alex parser optimizations which broke code completion ... at least code completion is working again in my test cases.#Zip demo createzipfile target changed to WinExeFixed SD-162:Minor UI inconsistency in NUnit padApplied Gutter.cs optimizations from Alex.Rebuild to reflect the changes in VB.NET.ATGFixed bug in cc recognition.Fixed SD-147:CC shows up in non-c# filesincluded vb.net -> C# conversion bugfixes from Nikola Kavaldjiev <nikola@gotlink.net>Fixed SD-156:Exception on Find using RegExAdd function to support cleaning of names to Zip file convention External file and attributes are now handled IsDirectory now handles the attributes for archives made on DOS ZipFile now has an indexer property ZipFile has new functionality for searching for entries by name ZipFile now reads external attributes from central header entries Cleaned up lots of summary comments all over the place. Fixed bug in ZipInputStream where unix attributes altered date and time when reading causing problems. ZipInputStream now checks version required to extract. ZipOutputStream version made by is not fixed at 2.0#Zip exception becomes an ApplicationExceptionfixed SD-153: Exception when adding file that already exists and is read-onlyCorrected description of the new csharp compiler options (compiler parameter was in the name of the property instead of the description).fixed SD-159: Visited unknown statement: one-line if/then statementRemoved from #Zip as this file is now generatedIncorporate SVN revision number into #Zip version infoFixed stack overflow exception when using circular inheritance (but this bugfix doesn't make circular inheritance working :))Fixed resource name bug.Small change in C# compiling: Execute script now gets the command line parameters specified in the project options.Add a NewCompletion method in the resolver. But it is not ready yet. Searched a cc bug: in the resolver is the IClass Type not found. Perhaps the cc for references don't work correct.Wrapped 2 additional C# csc command line parameters : /nostdlib and /noconfig (required for Compact Framework programming).Applied some of Alex uncritical optimization changes.nothing changedafter/before build event labels were interchanged in german translationFixed SD-145:CC not working for form designer generated codevb.net grammar: correct operator precedence (thanks to Nikola Kavaldjiev)before/after build labels were interchangednew pt-br language resource file was missingLanguage Resources for RC1Change log for RC1RC1 Setup scriptApplied suggested bugfixes from Nikola Kavaldjiev <nikola@gotlink.net>Fixed nowarn option for C# backend.Fixed C# CodeDOM generator.changed splash screen.checked in some changes from alex.Corrected VB.NET Parser >constant< field parsing.Fixed VB.NET lexer bug with end location (did cause possible folding error).Fixed bug in GetNamespaceContents method in ParserService.Fixed SD-132:Code completion on numbersFixed SD-113:CC fails on reserved wordsFixed SD-143:Class scout doesn't show parametersFixed SD-142:Exception when inserting code templateCorrected naming error.SDMenuCommand: UpdateStatus optimizations from Alex.corrected bug in the textword. Tabs are now working (was a little typing error...)caught possible property exception.Worked a bit on the TextWord class but I can't measure any performance/memory gain :(Applied some uncritical changes from Alex.Get VB samples compiling againChecked in Notebook changes: Worked on C# pretty printer.Applied some changes from Alex. Most was text += -> stringBuilder stuff in the assembly scout.QuickClassBrowserPanel is now drawing more efficiently.Couldn't find the 'host re-use bug'. Host is now always re-created again but a 'clean' dispose is called before re creation. This should help the GC and save some memory when working with the forms designer.Fixed some forms designer bugs caused by 're-use'. Forms designer commands seem not to work with the re-used forms designer ... all works fine when re-creating the host on each initialize.Found SdiWorkspaceLayout type conversion bug.Forms designer changes: Designer Host is re-used when initialize is called (FormsDesignerBase) instead of creating each time a new forms designer -> faster, more memory efficient etc., Services are now disposed when host gets disposed.removed some Console.WriteLineFixed SD-139:Exception when cutting / pasting tray componentsFixed SD-140:The select statement's case bodies are missingDispose of all Designers will be calledRe-Added 'save' Forms designer disposing.'Fixed SD-124:Multiple projects DLL's files lockFixed SD-137:Code Completion not workingFixed SD-99:Control Naming in Groupbox w Cut/Paste OperationFixed error in VB.NET -> C# conversion (bitwise and/or)Bump library version noTighten up on unsupported stream methods. Update commentsapplied some changes from alex.Fixed SD-135:indentation after else ifWhen trying to start a library without execution command a message box is displayed instead of an 'ugly' error boxC# Binding project options: Output path is now stored.Fixed SD-95:Refresh problem in forms designer / custom controlsNew version of the dock manager control.Fixed SD-87:Project options lack validation.Fixed SD-102:Adding new tools to Tools menu requires restartFixed SD-20:No Codecompletion for imported ActiveX

DLL fixed bug in GeneralTextEditorPanel. Update changes.txt Corrections to Release output was exe somehow. ZipFile finding of central header optimised. More tests added. More summary comments. Fixed a bug in custom components toolbar. (Added custom resolve event handler for loading referenced dlls too and caught exceptions per dll not for the whole operation -> if the loading of a single assembly fails the others are tried to load.) Add info command to help diagnose codepage problems. More changes from Alex. QuickClassBrowserPanel is now sorted like in VS.NET. Fixed SD-131. Worked on C# pretty printer. Basic project conversion should now work... Fixed endless loop bug (when closing #D). Fixed VB.NET enumeration codedom visitor bug. Fixed some 'optimizations'. Rule: Never try to Dispose a SystemBrush!!! swapped align center commands in the 'Format' menu (seems that the commands were swapped internally in the forms designer implementation. added more changes from alex. Started project conversion (C#<-->VB.NET) commands. hopefully finally fixed the bug that the 'move/copy?' dialog popped up even if the file resided at the right location. Fixed SD-121: Switching Project keeps TODO list. More changes from Alex. Added using statements for some more GUI components (brushes). 1. DockPane.cs -> Invalidate instead of Refresh when got or lost focus 2. DesignerResourceService and Save 3. Dispose of DesignerHostInserted changes from Alex. Worked on Code Completion List View mouse events. removed debuggin code. fixed VB.NET grammar bug with obscure if statement syntax (bug in spec). Merged in changes from Alex. fixed SD-94 for VB.NET projects. checked in new changes from Alex. Fixed some bugs in the changes from Alex (mostly in the old ones). added Alexandre to the authors conversion list and created new changelog. cpp backend binding: store output location as relative path. checked in missing files. fixed SD-94: 'Before / After build commands' fixed some bugs. bug fix of the call that fixed alex with -I. Improve sample code. More GC friendly changes, fix of issue with first change compilation. fixed compilation error. added saving/saved event for display bindings. Secondary display bindings now get informed after save. More GC friendly corrections. Added simple archiver sample in cs. Minor optimization for GC, speedup of parsing, code completion fixes for multi-project multi-namespace combines. #D can now load projects/combines from MonoDevelop (path / <--> \ concurrency issue). fixed SD-106: Tools / Options / Text Editor / General not picking up fixed SD-93: 'use output window' option is not persisted for external tools. Add summary comments; remove unused code; BZip error fns now throw exceptions. Removed unused methods, error functions now throw exceptions, add summary comments. Changes: 1. ISecondaryDisplayBinding interface change, allows open more secondary views. Files: ISubDisplayBinding.cs AbstractFormDesignerSecondaryDisplayBinding.cs CSharpDesignerDisplayBinding.cs VBNetDesignerDisplayBinding.cs DisplayBindingService.cs 2. Allow to open designer as "main" window and not only as secondary. Files: CSharpDesignerDisplayBindingWrapper.cs 3. Ability to append own EventBindingService: Files: FormDesignerDisplayBindingBase.cs 4. Property that allows not to load referenced assemblies. Files: CustomComponentsSideTab.cs 5. Using of Property FileName in definition, where the ResourceFile lies: Files: DesignerResourceService.cs 6. ToolboxProvider.cs -> bug fix (bad assembly compare) 7. CodeCompletion also for the others :) CodeCompletionData.cs SharpDevelopTextAreaControl.cs 8. Toolbar has not only one "ToolbarSection" CommandBar.cs, CommandBarManager.cs, ToolbarService.cs 9. Ability to open Updated docking manager to 0.9.2.4. Fixed some bugs in C++ backend binding. SD-86 fixed. Nice message box appears when compiler cannot be found. nsgmls added, parse.bat updated. Removed resource copying; instead, created hard links in conversion directories to \css and \images (Note: this works only 2000+). Use setup.bat to set up hard links. Initial import of the #develop help file that was originally written by Gavin McKay. I have updated the included tools (/DocBook folder), streamlined the directory structure, checked the build files (should be changed to NAnt build files). Note that this help file dates back to 8/16/2002 - quite outdated! forgot to check in some files... the project importer now asks for the project location, if it can't find the project. (for example ASP.NET projects have this 'issue'). fixed bug in default parser service. divide integer binary operation (VB) is now converted to C# as 'normal' division. Fixed possible little bug in parser service.. ICSHarpCode folder is being removed on uninstall for the current user. New setup file; checks OS version. new dock manager library (0.9.2.2) fixed possible dead lock problem in the parser service. added Method parameters to ctrl space list. C++ class wizard. SD-98: imported System.ComponentModel namespace in user control templates. Fixed: VB.NET --> C# convert bug (return statement expression == null). Fixed: The MessageBox Wizard doesn't escape double-quotes in the Code Preview or the inserted code. More summary comments. Add more summary comments. fixed code completion window bug ('new' now works). New version of the DockPanel Suite (fixes some bugs). Attention: New MDILAYOUTCONFIG format!!! Disabled the visibility menu in the view menu. corrected issue with AbstractUsing. The debug information for the referenced assemblies is now copied into output path (when referenced are locally copied). Tried to lower memory consumption of the #D Structure DOM Model. Referenced assemblies still take too much memory :/ Translated missing project converter messages and errors. Documents now have their full path name as tooltip. code completion window now has a scrollbar. Changed back to 'non locking' load of referenced assemblies in the custom component side tab. Selected text is now pre selected in the find dialog again. Fixed some C# CodeDOM modifier creation bugs. Forms Designer improvement: #D now shows when an component can't be created (e.g. when a component throws an exception in the constructor). Fixed some bugs in AssemblyAnalyser. added dimension property to TypeReference class. This is used for declarations like 'Dim myArray(3) As String'. Fixed delete files bug. fixed SD-81: Exception Converting C# to VB.NET (Form) fixed SD-80: Auto-naming of controls in Forms Designer (general) fixed possible file service exception. fixed 'recursive' assembly references. Fixed tab page 'show tab order' exception. Changed revision number substitutor script. Fixed nunit pad bug. fixed little issue in revision number substitutor script. updated languages (taken out some, added slovenian). samples are now working ... bug fix for ctr space. ctr space lists the static members now too. Fixed little ctrl+space pre selection bug. Fixed SD-76: Single file compilation: compile buttons in toolbar not enabled. #D now tries to copy the non-GAC 'referenced' dlls from

a dll reference in the bin/ output path.added local variables as fields for ctr spaceexecute before/after scripts now take 'arguments'.When switching the config (debug/release) project is marked dirty for re-compilation. Exute after script is only executed when the build succeeds.file types are now registered using Path.GetFullPath.Auto register filetype is now set to standard value 'true'.Fixed SD-77: wrong description in Standard HeadersUpdated "Greetings fly out to" section for DockPanel Suite in Weifen Luo's entryChange section "Greetings fly out to" to use two columns like all other sectionsDockPanel Suite entered as exception instead of Magic libraryJeffrey Huntsman removed from authors.xmlCorrected AUTHORS.xmlremoved magic entry from list of included projectsremoved obsolete todo list for the form designertaken out changelog generation in the build file (createchangelog.bat should be enough).DragOver no longer grabs focus.References are now copied 'deeply' into the output path (the reference from a project you're referencing).author conversionSVNChangelogToXml doesn't crash when Subversion is not installedFixed SD-75:Browser window caption not shown in Window menu (1110)New version of the DockManager (attention: Altered MdiLayoutConfig3.xml file format!)Changelog is now auto generated during build.added some error handling in the revision substitutor script :) (not everyone has subversion installed...)Udated NSIS installer script for Beta 1revision substitutor script is now only run in the 'build' task (before it was 'global' -> it ran in the clean task too :)RevisionNumberSubstitor is now a script in Main.build and uses subversion to get the current revision number.property pad no longer creates unnecessary transactions.Fixed little 'culture' bug in dockmanager.Changed name of DockManager to DockPanel Suite in the included projects and weblinks. This is the new 'official' name.New version of the DockManager lib.Fixed some compile warnings in #ZipLib.fixed possible error in HelpConverter.Fixed SD-71:Syntax highlighting definitions not picked upFixed SD-35:Ctrl + Space: preselect matching list itemFixed SD-70:Mini Class-BrowserFixed SD-71:Modifying custom syntax highlighting definition leads to exceptionFix another end of stream (-1 vs 0) bug (ZipInputStream.ReadFully)Update more commentsfixed possible help pad bug.added jfreilly to changelog stylesheetfile filter for C++ files is now translatedWhen compressing stored can be converted to deflate automatically. Checks for exceeding max size addedAdd Xml comments in places, spaces -> tabs, and formatting fixupsAdd more test (some failing). ZipEntry set comment bug fixed. ZipEntry ExtraDataBug fixed. Some Xml comments added, and indenting fixed upFixed textarea 'scrollbar' bug.Fixed SD-62:StackOverflowException when selecting classtest.added additional name conversions to the changelog xsl filefixed 'resourcedesigner' bug (affected mainly VB.NET).fixed SD-60, SD-49: new, svn log message based changelogTranslated some file messages - altered the splash screen.Fixed SD-24:RootFolder property is lost when switching between source and design viewstarted on mouse routines for the code completion list view but wasn't able to get the OnMouseDown/Enter/Move etc. events -> could need some help ... and the vertical scrollbar doesn't seem to function too. But I see no issue which can cause this -> help needed there too.save title 'flicker' fixed.fixed possible null reference in Help-Browser.csupdated splash screen & copyrightsremoved GC.Collect() from parser update thread.Select/Deselect events are now in BaseViewContentoptimized form loading a bit (the old dock manager had a better behaviour).Set svn:keywords propertyUnset svn:keywords propertychanged #D version to 1.0.0.1-2changed #D version to 1.0.0.1Fixed SD-55:C++ binding: Umlauts not appearing in output & task listFixed SD-53:Exception when opening VB.NET fileFixed SD-57:Unknown software exception after CCDB wizard restart.dockmanger bug workaround.SVN revision number now is a part of #D assembly versionnew version of the dockmanager lib -> context menus are still missing.Forget a savecombine after adding resources in a project. (forms designer)added slovenia flag.changed referenced assembly loading in the custom components category. (Referenced assemblies are now loaded once and are 'locked').bug fix: Search in project didn't scroll to the right position in a new filefixed SD-52:Cyclic dependencyfixed SD-50:C# Compiler default optionsWhen adding a ZipEntry check that the limit isnt exceededAdd test for handling large number of entriesZip encryption fixes. Some tar fixes. Tests updated. Some code reformatted as well.splitted Content-Name property of the IViewContent in TitleName and FileName property.Start page: the authors section, the changelog and the help wanted section have now the same look as the project section. This is also better readablefixed bug in resource editor context menufixed bug in resource editor: http://icsharpcode.net/OpenSource/SD/Forum/topic.asp?TOPIC_ID=5063implemented abstract class code generator for VB.NET.forms designer is MORE generous when taking components (I think the next step would be to allow any object for beeing a component :))Fixed bug in ResourceService.Fixed SD-46:Reloaded: Class scout not updated after class renameFixed SD-45:Exception when converting c# file to vb.net.Code completion for DirectX and other GAC C++ DLLs is now working.MessageBox wizard can now create VB.NET code.Fixed SD-42:Convert Buffer language x to language yFixed SD-41:Convert Buffer VB.NET to C#Fixed SD-44:Code completion does not workFixed SD-38:Quick XML doc box has wrong positionfixed bug in lexer: 2.ToString() now is corretly parsedFixed SD-40:Incorrect XML documentation generationFixed problem with ignored test. Altered test to demonstrate encryption failure. Now compare compressed and decompressed lengths explicitlyTurn off code completion option now works again.ResourceService -> RegisterAssembly -> was implemented only for default languagehopefully fixed redraw bug when opening a file the first time the fold markers may redraw wrong.Fixed some minor GUI glitches (mostly FlatStyle.System not set.)Made the created in path label in new project nicer.translated the #D file templates. Made the new file & new project dialog look a bit nicer.SharpDevelop.cmbx opens again.Fixed add 'unnecessary' reference bug for VB.NET forms designerforgot to checkin new parser.csstatements in select sections are added to the AST; added a property EmbeddedStatement to the SelectSection classcase clause can have a trailing colonFixed bug in project compile files option panel.default return type (System.Object) for properties and functionsarray creation parameters don't end in nirvanaFixed bug in sidebar when 'components' don't have an icon assigned to.Controls which are in 'referenced' assemblies that aren't in the GAC (or project references) are now added too to the 'custom controls' side tab.fixed bug with display bindings that

don't contain file names (like web browser) Fixed several VB.NET bugs (abstract visitor & forms designer) Fixed new project 'encoding' bug. 1. PropertyService -> public static void SetConfigDirectory allows to set "individual" config directory for all apps that use #D core 2. CustomComponentsSideTab -> the tab will be reloaded when new project is opened (new assemblies take place) fixed bug in ZipLib sample. NUnit pad now recognizes combine close. Fixed bug in new file templates (csc no longer need to be in the path) Fixed SD-19: Class scout not updated after class rename Fixed SD-14: Syntax highlighting editor: IOException: 'File is already in use' Fixed SD-37: Clipboarding doesn't work Fixed SD-13: Wrong cursor position with non-monospaced fonts Worked around the component tray bug. Thanx to Ivo Kovacka for finding the bug cause. Working on C++ backend optimized the class browser refresh a little. Fixed bug when closing a web page. Fixed formatting strategy bug. internal members are now listed in code completion fixed project importer bug. worked on class browser. Fixed bug in VB.NET CodeDOM visitor - friend & protected modifiers are now working. array modifiers don't get lost (ArrayInitializationModifier production) RHS of declarations can't have rank modifiers Option Strict/Explicit options don't have a mandatory option value applied fixes for #ZipLib and Forms Designer SerializationService. made the #ZipLib source a bit nicer. changed the readme for #ziplib a bit. changed the buildfiles so that they're now working with the new #ziplib directory structure. restructured #ziplib directory layout. worked on #ziplib fix: 'File has changed externally, want to reload' message doesn't appear when the file has been deleted added undo/redo buttons in the toolbar (feature was requested). Fixed VB.NET bugs. Translated export project dialog and some strings in new file dialog. fix possible 'refresh' text editor bug. worked on the IL syntax highlighting. wrapped the 'ambience not found' case in a nicer error message. put try ... catch blocks around text editor drawing operations to handle the 'generic GDI+ exceptions'. fixed bug in ExtractBaseClass. tried to speed up assembly scout a bit nothing changed Implemented '///' documentation autogeneration feature. Fixed bug in #Refactory where constant field declarations had wrong start location. Classes are now 'designable' when they have Form or UserControl in their inheritance tree (they no longer need to inherit directly from Form or UserControl anymore). CodeDOM <> XML DOM removed some if(Node != null) from the AbstractASTVisitor which is obsolete now because of the null objects ToString() code generator now works with fields instead of properties. Updated HowYouCanHelp.xml removed dll. Removed magic library. Set flatstyle = System in the assembly scout panels. changed file menu 'E&xpport Project' to 'Export Pro&ject' because 'x' is used for e&xit. Removed unnecessary references to the Magic Library in the buildscripts. Mike: #Develop now is completely independent of Magic, so you can remove it from the repository. removed some unnecessary debug messages - Changed Assembly scout to use WeifenLuo interface library. There are some quirks, for example I don't know how to order the dockmanager to show the tabs at the bottom of the control as before, and the icons are not shown on the tabs. - Fixed what Mike had broken when he optimized the memory usage... one really shouldn't access null references ;) added assemblyinfo for ILAsm binding. worked a bit on the C++ backend. little bugfix in remove reference. Added IL Asm backend and simple hello world project. Still something todo: syntax highlighting, wrap all compiler options. Maybe more templates (file and project), and invent some 'icon' for the templates. And I took out the debugger for 1.0. added form bzip2 bugfix for #ziplib Fixed build process glitches in the C++ backend binding. Reverted some changes in the debugger (I broke something ...). Manual breakpoints now are working... used the new null objects made a base class for property get/set regions and event add/remove regions and added a null object for both C++ formatting strategy now works. Pre compiled header support for C++ backend should now work. (at least it does for the debugger.) Debugger should now compile without ATL and STL. restructured the directory layout of the debugger a little -> removed all VS.NET solution files. Compiling & linking the debugger now works inside #D. What is still needed is pre compiled header support and then we use ONLY #D for debugger development. Debugger now compiles with #D (still a linker bug...) Fixed registry helpkey bug reported by Gerd Klevesaat. Worked on C++ backend. CPP backend now only recompiles the changed files and not the entire project. CPP backend: compiler & linker are now run explicit. Html build logs are now generated. Empty project icon is now in the 'new' style. Introduced new (language neutral) naming scheme for file and project templates. This makes it easier to add new languages and the maintainance of the translation. Added user control library template for C# and VB.NET. Please test for spelling errors and all file/project templates. They need to be consistent and should work at least... added wxs extension to xml highlighting definition (wix filetype) Added wix project. Single file VB.NET compilation now works again. Added wix and a basic wix backend binding. added NullTypeReference and use the new null types everywhere warnings are now underlined orange. enabled 'error' drawer. (Does anybody remember the red 'wavelines' we had in former versions? fixed little issue in code completion placement. applied debugger patches from Cristian fixed bug in Collection-Suffix rule (Dictionaries). Fixed highlighting editor. new C++ files get now the correct highlighting. Added C++ backend TODO list Added debugger project files Fixed bug in VB.NET resolver. solved code completion focus problem (thanx to markus). fixed little code completion bug: own enum members have now the correct icon. Tried to find out why the component tray behaviour is strange (but only the first created tray). Didn't find anything -> could need help with this issue. Fixed some minor things (context menu stuff, added property action, translated 2 items). well formed authors.xml corrected bug in authors.xml fixed some show tab order bugs: Tab order is now shown for 'sub' controls (controls in controls for example inside group boxes) and should be handled correctly. Tab order change now sets the 'dirty' flag. Tab order controls now listen to tab order changes instead of being informed by the design panel. This is better because now it recognizes all changes (undo/redo or property grid). forgot to rename a file fixed code completion bug: Enum members were not shown, because they are neither static nor public Implemented some NullObjects (Expression and Statement) to avoid test for null in the Visitors. Added Show Tab Order feature from Gerd Klevesaat. Open the forms designer and select Show Tab Order in the context or main menu to see it. Use LeftClick-

/Shift+LeftClick for altering the tab order. Added error message in the Assembly Analyser when the dll can't be loaded. Added TagComment.cs Added textarea context sensitive help for .NET classes (go with the caret to 'Form' or 'System.Console' and press F1) Added the ClsCompliantRule to the AssemblyAnalyser. fixed bug in solution converter. UI Permission error ... -> I could need someone who looks at this. I've no understanding why this permission isn't granted and I wasn't able to enable it (current solution: the project importer don't give feedback). tags for untitled files are now longer inserted. Tags are now working for VB.NET too. Fixed some forms designer issues. translated the 'fill out first' message in the new file dialog. translated missing resource editor messages fixed Typedhashtable template for vb.net fixed Typedhashtable template for c# Code completion <Tab>-key now works correctly (doesn't insert a 'tab' anymore) Fixed code completion 'performance problem' ... code completion is >MUCH< faster now (think of 20 ms vs. 650ms) resource editor list: avoid flickering when deleting items Task list 'tokens' now work again (TODO/HACK tags etc.). Added option panel to alter these tokens Added the C# ATG file and started to look at it. Added some Comments what children are possible A couple of C++ templates Fixed signed/unsigned comparison Working on C++ backend Can't desialize error isn't longer a popup (now it is displayed when trying to switch to design view). resource editor: toolbar/context menu items are enabled/disabled when selection changes; context menu uses base commands for copy&paste operations forgot to check in new addin file for resource editor NUnit pad bugfix: now each assembly gets it's own appdomain. moved resource editor context menu to the .addin file (-> commands) fixed various resource editor bugs enhanced the nunit pad a little (yellow state now shown.) Forgot to check in NUnitFramework. started to look at the C# ATG file Removed NUnit except the nunit framework which is now under libraries and get build during the build process. Add NUnit pad which is now the test runner for NUnit tests inside SharpDevelop. finished the AbstractASTVisitor finished the statements in AbstractASTVisitor in CsVbRefactory code completion list box is now sorted alphabetical (again) fixed code completion window placement bug and little (very little) code completion bug. finished goto source feature of the assembly analyser. Implemented a SearchMember method in the ParserService and completed the GetPosition method started the goto 'error' functionality in the assembly analyser. delegates are now put into the class tree (C#). Added 3 new rules for the assembly analyser tab now completes the word in code completion again. Assembly analyser now uses the localized menu item. new code completion window is now faster. fixed code completion bug. forgot to check in file. improved code templates a bit : now it is possible to add the selected text to the template with \${Selection}. I changed the default C# templates now you can surround code with blocks (for example try ... catch blocks) by selecting the statements, press ctrl+j select try ... catch and give it a go. bug fixed the new code completion window ... it should be 'usable' right now. added a Filename to the ICompilationUnitBase added a Position class that contains a CompilationUnit, a line and a column wich I want to use as mathod parameter for the methods wich were moved from the Resolver started a GetPosition method wich gives a Position of a member or class wich is specified as namespaceName.Class.InnerClass.MemberName need to move the SearchMember method from the resolver to the ParserService, for that a base class for all members is needed worked on code completion window (still not done ... therefore be careful with the new code completion window). moved some methods from the resolver to the parserService and let both resolvers (C# and VB) use them worked on insight completion window. fixed code completion bug: Now inner classes get resolved. For example : Controls is from a inner collection and 'myCtrl.Controls.' is now resolved Worked on AssemblyAnalyser. Fixed Gui error enhanced the details view (now it uses the help stylesheet) #D forms designer now accepts more components (without bitmap assigned to). You now have to turn off components explicit with the DesignTimeVisibleAttribute ... I think that is the correct behaviour. fixed little AssemblyAnalyser UI issues. renamed more classes & files renamed filenames. renamed rules Tweaked the GUI from the AssemblyAnalyser a bit. Added new rules for the AssemblyAnalyser. worked on translation of the assembly analyser. All descriptions and resolutions are now translated. AssemblyAnalyser is now localized. added more naming rules to the assembly analyser... worked on assembly analyser. moved the unit tests to the files where the class to test is defined. Added 2 new rules. last change in assembly analyser for today ... Assembly analyser is now split into 2 dlls one strong named that contains the engine and one addin. Unfortunately we can't strong name the 'base' dll now therefore we need to split it up when a strong name is required. It is required in the Assembly analyser because it scans the assemblies in another appdomain which is unloaded before each build (therefore the assemblies do not lock). Furthermore the assemblies are reloaded after each build -> the analyser status get updated. Re-Structured assembly analyser directory layout worked on Assembly Analysers small bugfix + coding rules (issue identified by Assembly Analyser (c)) Working on C++ backend added some rules to the AssemblyAnalyser. Changed component tab assembly loading (now without shadow copy). Started FxCop clone plugin (AssemblyAnalyser). fixed code completion bug. fixed code completion bug: base class was not found in some cases Fixed code completion/insight window placing bug. fixed AltGR bug in command bar library. Please try to find errors with hot key combos Alt, Strg, AltStrg and especially AltGR behaviour. All non-english users should try if their AltGR+Key works as expected. added boolean view for resource values; please play with the resource editor and tell me what you think. overworked resource editor; still missing: filtering and views for integer and boolean values Working on linker option-scompiler message view got its own subdirectory. refactored CompilerMessageView -> now we can handle more categories than build/debug. Code generators now recognize internal classes. Fixed SD-36: Folding doesn'T work for nested types added texteditor bugfix from Frederic de Gombert Fixed bug in C# formatting strategy that could scramble a line when using block comments. fixed some projectfiles (xommens on revision 836 from xo) #D now supports the compilation to different installed runtimes. folding now works when switching between design/source code mode (folding positions and states were lost). worked a bit on the vb.net code generation. fields have now the correct return types in the vb.net files. fixed tool bug (properties weren't enabled). VB.NET end

constructs bug fixed for vb.net code generation.added default keyword to c# highlightning schemeworked on AbstractASTVisitor: TypeLevel#D now compiles again (renamed bringtofront method in iPadContent to bringpadtofront) -> thats why it is much better to inherit from abstractpadcontentclassscout & task view pad now inherit from AbstractPadContent.improved the string literal conversion from C# --> VB.NETRestructured the 'View' menuworked on AbstractASTVisitor: GlobalScopefixed 'end' bug in select statementadded new exe configuration to #rvb projectfixed nullreference exception in project options dialogfile name arguments for the c# compiler are enclosed by quotes -> compiling with mono works againMade a test for the ToString() method and updated AbstractASTVisitor and IASTVisitorFixed little bug in localized property grid. Now it is possible to swap boolean values with double click.added NUnit test file templates for C# and VB.NETmethod insight window is now positioned correctly with folding lines above the first visible line.fixed another bug in debugger service.fixed bug in debugger service.tried another method for getting the GDI+ exceptionsadded last missing VB Files (statements, expressions, global scope, type level) and merged the enumsenhanced the key listening of the method insight window (old version might produce a 'memory leak' or that up/down keys might not work in the textarea.added new condition: isprocessrunning. Added icon for stop command and put the stop command into the toolbar.worked on debugger (GetFunction in THSThread) ... still can't receive the information about the sourcefile position of a step/breakdebugger stepping now works (still a long, long road to go...)Made the up/down marker in method insight window look nicer.now you can scroll again to the overloads in the method insight window.added 'new' completionworked on code completion database -> please regenerate...fixed project template 'header' bugworked on debugger stepperworked on debugger breakpoints and THSModuleArray values are now printed from local variables (I still don't know how to get the name ... I think that it has something todo with the ICorDebugFunction stuff -> you've to go deep into the IL for getting just the NAME of the variables)moved some files and added C# TypeLevel initialized all ArrayList with capacity 1 to avoid waste of memory debugger variables now recognize >null< valuesworked on debugger 'local' variables (they work pretty good :) missing are 'null' values, arrays, and objects but all base types work)worked on the debuggerset project browser context menu to null when combine is being closed - fixes SD-17fixed bug in text area memento loadingremoved separator item from build menu, moved other separator to the cinditional section (--> no separator at the end of a menu)fixed 'stackalloc' bug in our C# grammar.worked a bit on the debugger.worked on the debugger: STOP process now should work, the break & continue command should work too but you currently aren't able to set breakpoints other than Debugger.Break(); calls - still to do: get all infos about processes/threads/moduleswoked on CsVbRefactory: General/GlobalScope files added and GetCollectionString used in ToString methodsfinished new 'classproxy'reverted change in Resolverfixed code completion bug. (had to do with the classproxy changes)checked in new CPPCompilerParams file (I garbled that :/)enabled ctrl+spaceparser service changed...Proper killing of debugged process, allowing dumb compilation of cyclic dependent projectsWorking on C++ compiler optionsimplemmented text marker for the text editor. Text can now be highlighted, underlined, wavelined.still forgot one case :) (folding marker update)improved the fold marker listening to remove/replace/insert events (I hope I now got all cases)improved the 'position' of vb.net constructs in the parserworked on VB.NET expression parser.added VB.NET expression parser.hopefully fixed textarea drawing errors THIS time (maybe it is a red herring ... anyways a #D long term joke).fixed little bug in abstractdecoration (parser layer) ... this happens when optimizing code :)Optimized the classproxy class -> no empty collections anymore this get you 3MB less memory usage on startup (43MB on my system). btw. idea came from MonoDevelop (Ben Maurer) ... but our solution is cleaner :)fixed vruler bugfixed a bug in createlines method (may have produced 'stange' lines.Added some folding mouse actionstext editor state is now preserved (caret position, bookmarks, folding states, choosen highlighting)added Typelevel files to CsVbRefactoryadded mouseover tooltip for folding.it is no longer needed giving all environment colors in each highlighting definition file (the Environment section isn't longer required to be defined).added the data/modes directory to the highlighting search directories - hey nobody missed the C64 C# Syntax Highlighting ?added on mouse over effects for the fold margin ... and made it much easier to add new environment colors to the highlighting filehome/end keys are now working with foldings. (we're underway ... folding is 'pretty usable' now :)improved the folding behaviour when using insert/remove/replace functionsfoldings are now drawn as 'selected' when they're inside selection range.word left/right now works with foldingsfixed bug with bookmarks + selection (selection wasn't cleared when goto next/prev markfixed texteditor bugcaret left/right/up/down is now working with folding. !! Desired column still needs some work !!triet to update caretactions but found bug: Linelength was inconsistently used: sometimen with delimiter and sometimes without need to look over all uses of the line length in Texteditor and after that good testing is needed :-)removed some console.writelinesexpression finder now recognizes 'using'fixed endless loop bug in expressionfinderfixed vscrollbar resize on folding change.corrected code completion window placement bug (when folded lines above first visible line)The GetLogicalPosition from view coordinates function is now working with folding (this is used in the mouse events but may be useful for edit actions too). Fixed C# abstract visitor bug.worked on CsVbRefactory, all strings and ArrayLists have to be not null: added Assertions added cs only expressions and statementsbug fixed, builds again :->added expressions/statements to the common ASTupdated the VbCsRefactory projectdeleted some duplicated filessplitted up code actions in separate files. Added Equals/GetHashCode code generator. Code generation is now on Alt+Insertaaaaaaaaaathe expression finder now recognizes the 'inside string/comment' better (using 'bullet-proof' algorithms from our C# lexer) - I hope it works now but only time will tell...removed some unnecessary binary filesnothing changedfixed openfile command bugworked on project browser 'add file to directory' command and fixed some other project browser bugs.found one 'speed' problem which has something todo with the treeview -> going through the treenodes in another thread (using invoke) may lock the UI thread ... currently I disabled

the add/remove event listener for the classscout.C# expression finder no longer accepts 'System..' as a valid expression.fixed bug in C# ATG file: int.Parse(5); is now a correct stament.introduced 'expression finder' interface. The IParser implementations give it back. It is used for finding an expression before a give offset. This is used for code completion. I implemented the C# version as a finite state machine. This works much better than the 'old' version (no code completion inside strings/comments for example and it recognized object creations better)added #Develop file templatesadded another #ZipLib bugfix for the GZIPOutputStreambugfix (return -1) for the #ZipLib in 'some' inputstreamsRefactored localized property gridforgot to check in some files for the standard header panel. Added GotoSourceFile function in the debugger and overwritten it in the debugger service ... let the breakpoint testing beginmanifest files are only created for dllsmade the C++ syntax highlighting better.project templates are now using the standard headers too. Removed java project templatesadded panel xformcreated standard headers edit panelfixed code completion bugs: LookUpTableVisitor didn't visit the statements within the foreach statement and Abstract visitor caused an exceptionAdded C++ project templates. Worked on C++ backend (I could need a lot of help in this area)fixed code completion window bugadded option to toggle the quick class browser panelworked a bit on code completion gui.I think I've fixed the 'textarea update bug'localized all file templatesfile new template property grid is now localizableadded typed hashtable for VB.NET. Removed typed collection generator wizard. Instead we now have typed collection generator templates.worked a bit on memory consumption and speed. Code completion database wizard now only has ONE way of creating the database (the fast one :))fixed invalid cast exception in visitors and changed them, so that all visitors inherits from AbstractASTVisitor and write the visitor and the Node to the console if an INode is visited'new' files are now named correctly (no dupes)Added StandardNamespace project option. Added localizable true/false values for the localized propertygrid.worked on C++ compiler optionssscripts are now possible inside file templates ... see the C# TEST script for detailsInitial revision of C++.NET backendchecked in html control change from alex (surpresesses warning boxes)added vb.net singleton templateforgot to add NewFileWithNameDialog.xfrmfixed bug in project templateschecked in quick class combobox changes from Alextweaked parser thread a little. QuickComboBox now no longer listens on the parser thread (it's not required it listens to caret changes which is enough)fixed bug in Main.buildAdded Misc and VB.NET file templates. Please test them carefully (and enhance when possible) ... the same for C#.Added new file template (typed collection), file templates can now define 'wizards' ... look at the typed hastable for an example.added first 'new' templatesremove file templatesworked on file templates'solved' image list creation problem with creating manifest files for all compiled projectsworked on pretty printermethods/properties/indexer/events without body no longer can't be foldedworked on the pretty printer. Fixed numerous C# parser/lexer bugs.Pretty printer now preserves comments (unfortunately most people don't share my opinion about 'advanced garbage filtering'). The same method can be used for preserving comments in the conversion visitors maybe the output visitors can be used by the conversion visitors.worked on SharpRefactory/PrettyPrinter. Operator declarations are now put into the AST (we forgot them), fixed error in operator declaration grammar.added the gdi+ exception workaround'SharpRefactory has now autogenerated token/keyword classes tooparser token build script now generates a method which gives the string for a given token value (required for pretty printer)fixed SD-7:Code completion window pops up when typing . inside a string literalaltered buildfileremoved java script backend (wasn't used)corrected folding update bugcorrected the grabled tool menufolding is now unfolded when the caret is inside folding boundsquick class browser combobox now works with untitled filesfixed fold margin click positionfixed screen update bugs (happened with folding)new debugger dll ... hopefully this one works on other systems too :)Checked in new debugger dlladded THSProcess class in the debuggerchecked in new debugger and 'missingdebuggerfeatures.txtlicensed ICSharpCode.Debugger as LGPLrefactored the debugger a bit (mostly renaming/removing unneccessary code and signed it.checked in new debugger dlladded module load/unload events. Removed unneccessary 'container'Changed untitled file bug from forumfixed SD-30: NullReferenceException when opening C# filefixed SD-31:Can't edit form in design mode: 'Please correct all errors first'debug messages now are working in the debugger windowremoved core2002.suoadded release version of the debugger dllchanged debugger assembly infoadded compiled debugger dllrenamed directory#D now compiles clean with the new debugger. Changed namespace name of the new debugger.added new debugger sourcesworked on the 'fold' merge ... but still having threading issues.fixed bug in ToString() generator :)Added new ctrl+w code generator: ToString() generator generates default ToString methods in the form : [CLASSNAME: PROPERTY1=VALUE, ..., PROPERTYN=VALUE]. Code Generations NEED to be overworked (using CodeDOM when possible, make nicer UI add new ideas etc.). Anyone cares to take the job ?added #region areas to C# and VB.NET. Enhanced support of untitled files.worked on translationscorrected fold description drawing (was sometimes off one pixel)correct caret location errorTweaked font rendering a bit. altered C# highlighting: no italic fonts (because of font drawing bugs with italic fonts currently the text editor can't draw them correct)added missing fileadded dynamic help (F1) command. works currently only in the forms designer and propertygrid (for forms)fixed SD-5:Exception when trying to save moved filestart/stop without debugging should now workfolding comment rectangle was one pixel too short#regions are now inserted as folding region for C# and VB.NET. They're folded automatically ... but keep in mind that folding is still a dangerous thing to doadded new debugger coremessage box text is now scrolledfixed event name bug for C# visitorWorked on scrolling routines to recognize foldings.breakpoint drawing only updates once during togglemessage pad should now behave betterworked on debugger. debugger output is now captured. started with localization of the debugger pads/commands.debugger can now be stopped ... but it is still buggy. but at least it can stop/rerunadded breakpoints padMouseclick line with folding works now TODO: scrollbars, mouse column and actionsWorked on debugger.corrected build scripts againcorrected 'test' errorchanged the build scripts that they only pause when a build error occurred ... this helps to

recognize errors (idea from andrea)added some missing filesworked on the debugger. When folding disabled the calculation in the text area is now faster.fixed bug with the quick class browser panel that allowed two class browser panels on new files and it should now behave correctly on file rename. Added debugger service (still empty)Added debugger library to build scriptGot permission from Jon Shute to release his debugger as LGPL. I've included the LGPL'ed sources and the compiled dll. (nant doesn't support C++ yet)the selected text in the quick class browser combobox is now drawn with correct highlight coloradded project reference load with partial name bugfix from leedgitarcorrected folding drawing and caret positioning calculationadded try ... catch for potential exception in customcomponentssidetabcursorcolumn is almost right with foldingfilewatcher sets file dirty if reload question is answered with 'no'made GetVisibleColumn(), but it is still buggytweaked the format flags for the quick class browser combobox a bitquick class browser no longer shows the parsing of the whole combine. It listens now only to the parse change of the current file.positions of interface members are now parsed correctly (C#)it is no longer possible to compile/run untitled single files. Added new condition: ActiveViewContentUntitledui update is now started when switching secondary view contentsThe labels from the secondary view tab are now updated on language changefixed bug in command bar library and informed Lutz Roederworked on quick class browser combobox. This feature is now in 'testing' state. please test & report bugsworked on quick combobox. changed some ambiances.new rule: anyone who changes the ambiances must read the #D coding styleguide.fixed C# verbatim string lexer bugworked on quick class browseradded satguru srivastava to the authors listadded addinscout from satguru srivastavareverted inadvertent check in, sorrychecked in missing fileAdded 'quick class browser' panel in top of the text area cut, paste and delete sets dirty flagresource designer implements IEditable, copy paste operations work from the toolbarresources can be renamed in a more intuitive way: like renaming files in windows explorerconverted RenameEntry dialog to a xml formreformatting and renaming of a ResourceEditor class (--> ResourceList)Changed SD-27: 'Reload file' is always enabled, even if there is no file openendundo/redo restores now the caret positionCaught Clipboard.SetDataObject exception for the exception boxgutter margin now handles all mouse events itself and should work correctly with folded regionsfold markers can now be folded using the fold marginadded VB.NET direct3d project templatefixed SD-26: Exception when opening a item from the Help | Web menuprinting folding text now worksfixed SD-28: Exception when switching to design viewcorrected bugs in vb.net converter + grammar.worked on vb.net converter.worked on vb.net --> C# converter.now add/remove handler is converted from VB.NET to C#. Corrected bug in VBNet grammar: parameter name wasn't inserted into the AST.fixed C# formatting strategy bug. (auto insert curly bracket might have scrambled code sometimes a bit.worked on folding: LineNumbers are working, merge is working missing: actions and mouse handling, folding descriptionVB.NET projects can now be exported with the VS.NET exporter tooworked on vs.net exporter. primitive C# projects should be exportable.Changed assemblyinfo.cs from commandbar lib added new command bar sourcesremoved old command bar sources!!!TADAAA!!! printing now working again !!!TADAAA!!! ... this time it prints lines longer than fit onto a single page line, it's syntax colored and doesn't contain line numbers, tab & space markers rulers etc. Most people didn't want them and I'm currently too lazy to do a print config dialog.first Folding is working. Linenumbers are not working and the foldings are not mergedAdded upper/lower case conversion to ctrl+L and shift+ctrl+Lcontinued work on VB.NET --> C# converter ... the most stuff is done and I can now need bug reports. btw. don't report bugs like : XYZ isn't converted it produces '//TODO : Statment XYZ' outputcorrected bug in vb.net ATG file and VB.NET to C# converterworked on the vb.net --> C# converter.fixed vb.net parser bug: raise event statment wasn't inserted in the block.Worked on the VB.NET --> C# converter (but most stuff is still missing)Corrected AUTHORS.xmladded 'optimize' flag to the build scriptsAdded WinformsUI libfixed SD-16:NullReferenceException when opening VB.NET filefixed SD-21:Double click on event causes Exceptionchanged the dockmanager control xml format to be localization independent.99b setup file (only version # modified)added debug code for layout file loadingdefault layout is now embedded as resourcechanged the default layoutfixed menu command checkmark bugremoved magic info from authors.xmlFixed bug in the sdi workspace windowthe MdiLayoutConfig.xml to MdiLayoutConfig2.xmlupdated authors.xmladded dockmanager control from Weifen Luo. Removed magic source code.took out duplicate namemade project import dialog localizablere-activated the file system watcher-corrected AUTHORS.xmlupdated translators list in AUTHORS.xmlAdded VS.NET importer improvements from Scott Gilroy-Cut/Copy/Paste should now be enabled correctly in the forms designer (toolbar buttons work as expected and ctrl+x/cv now are working)Added UnRegisterProfileHook.batFixed empty menu 'del/backspace' bugForms designer change: added new base class for forms recognition. works for C# and VB.NET and reduces the parsing to a minimum -> faster form loading timescomponent combobox is now filled after the end of transaction instead after each 'event' -> combobox updates are greatly reduced -> faster loading and cut&paste for files with many componentsdisabled the wrong convert buffer command :)added install/uninstall nunit batch filesChanged splashscreen to 0.99btemporarily disabled CSharpConvert buffer (for 0.99b)Changed the code generation option panel for C# backend binding to a localizable property grid. Wrapped /nowarn: option. Moved output type/application icon to the 'output' panel (for VB.NET too) because this makes more sense.#D builds again...added missing .prj File for the SD combinenow using the observed load for combine loading and removed some ancient combine loading error dialogsmade the file filter search in the open command somewhat smarterWorked on the VS.NET Solution import filter (they should now work better and translate the solution layout 1:1)forgot to add visitor file for converteradded VB2C# converter skeletonfixed SD-8: swapped internal & public enum values in the vb.net modifiersfixed SD-9: Error in VB.NET grammar catch clausefixed bug in VB.NET lexerfixed a bug in C#/VB.NET forms designer that may remove fieldsfixed bug in project scoutupdated project file (error.cs --> errors.cs)renamed error.cs corresponding to the class name (errors.cs)renamed error.cs corresponding to the class

name (errors.cs) correct namespace for errors class added missing addin combineCode completion database wizard can now be skipped. Reversed default mouse direction. fixed forms designer (c# and vb.net) modifier bug files that are at the right location won't be moved/copied anymore fixed single file 2 times open bug Form designer: double click should misbehaviour fixed (now the caret will be at the correct line in the source code) Form designer components may now be something other than private- fixed duplicate component bug without disabling Add(component) fixed duplicate component bug regexptk uses now file utility service for validation of filenames Fixed bug in fileUtilityService.IsValidFileName made the last bug fix more elegant bug fixed the last bug fix fixed file new bug Checked in new resource strings updated language xml Updated to nant 0.84 updated to npro 0.8b alpha fixed C# code completion bug enhanced VB.NET forms designer field placement in source code updated how can help.xml removed #unit references Setup files updated for NSIS 2.0 Release syntax, .99 WithStatement start and end added, code completion in withStatements should now work fixed vb.net code completion bug Readme version for .99 version Updated Start- page copyright notice Samples are compiling again fixed menu designer bug Component drop down is now sorted Added forms designer changes from Ivo Kovacka Bug fixes in Designer Resource Service ... now no open project is needed for designing forms Updated TODO Bug fix : Extracting data from table and view that have space into their name Couldn't save an data extract view into an xml file Add : Execution of a data procedure. Bug fix : Extracting data from table and view that have space into their name Couldn't save an data extract view into an xml file Add : Execution of a data procedure. Bug fix : Extracting data from table and view that have space into their name Couldn't save an data extract view into an xml file Add : Execution of a data procedure. Bug fix : Extracting data from table and view that have space into their name Couldn't save an data extract view into an xml file Add : Execution of a data procedure. fixed some bugs in VB code completion empty expressions are given to the resolver (used for with statement in vb.net) added statements to CsVbRefactory fixed some Bugs in VB code completion VB.NET code completion now works again and is case insensitive fixed 'edit twice' bug in highlighting editor Checked in codedom generator change from niv lederer resources should now work Checked in Niv Lederers resource service changes (doesn't work yet) Fixed 'remove' forms designer menu command bug Checked in improved CodeDOM Visitor from Ray Gilbert Fixed possible text editor generic typography misbehaviour. StringFormat.GenericTypeGraphic doesn't return a new object or a struct instead it points always to the same ... and sometimes the text editor needs to alter the string format which could lead to a wrong string format for other components. Therefore the original string format needs to be cloned. Project for only one AST for VB and C# started Fixed Bug in VB code completion: the usings didn't work try to take care about casing in VB, is not completed because the ParserService must be changed WithStatements added in VBReolve bugs fix : 1) resources string aren't prefixed with "{\$res:" 2) Table, view and procedure with white spaces aren't interpreted correctly into SQL command. Added FirstVisibleRow property to the text area control Added a SharpDevelop Combine (over 200000 lines of code overall now) and updated some combine and project files CtrlSpace Method in VBResolver added and lookUpTable switched on bug fixed labels in #query removed obsolete doc directory and todo.txt little bug fix for 'end' statement solution Forgot to add end statement to the block :) Hopefully fixed 'end' statement in the vb.net grammar fixed text editor font update bug Re-Arranged the menu structure a bit. Updated translation resources Added VB.NET LookupTableVisitor. Put WithStatment into the abstract visitor structure (it was missing) Initial ImportMethod for CtrSpace in C#Resolver added Fixed designer 'update' bug Fixed vb.net designer bug vb.net forms designer bug fixed set form designer version to 0.99 Fixed shema bug :) VB.NET project templates fixed Worked on VB.NET code insight first VB Code completion Worked on VB.NET code completion Disabled addin.xsd because the shema blocks extensibility changed version to 0.99 worked on TypeVisitor added hungary translation multiple inherits and implements work in all cases multiple inherits and implements work in all cases vb.net form designer sets dirty flag new vb.net windows forms project template Fixed bug in VB.NET CodeDOM Visitor abstract subs, functions, properties without body worked on vb.net abstract visitor statement level Fixed various vb.net abstract visitor bugs Finished the expression level visit in the abstract VB.NET visitor label statements have an embedded statement property; LocalDeclarationStatements are added to AST 3rd party buildfile is well formed again- correct variable name, compiles again VB AbstractVisitor improved Removed Dr.C# Debugged VB.NET designer (event creation should now work) empty (=optional) arguments in invocations work short expressions in se with blocks work VB AbstractVisitor improved Worked on the VB.NET Forms Designer (Event Handlers). Fixed bug in VB.NET --> #D AST Visitor (method parameter type bug) Corrected a bug in VB Ambience+ label is not mandatory for resume statement + added LiteralDecimal to SimpleExpr + added MustOverride to MemberModifier Fixed minor bug in the CSharpParser exit select statement added worked on AbstractASTVisitor so that one day the whole tree is visited Corrected parser update thread (sorry Alex, please look over this). Corrected locations of some VB.NET constructs inside the AST. updated Lexer so that there is a EOL token before the EOF token in all cases Fixed a text editor bug where sometimes a just deleted character (the last one in the line) is displayed inside the text editor (backspace key). Bug fixed vb.net lexer and vb.net forms designer Core change: Assemblies are no longer loaded twice Corrected 2 import solution bugs Corrected resource creation bug in import solution routine VS.NET Importer should now work (at least for VB.NET/C# projects) Worked on VS.NET Solution Importer/Exporter. Enhanced the smart indenting inside documentation comment fixed variable initializer bug fixed type of bug moved 'dim' to the modifiers productions syntax error in codedom visitor Corrected type conversion list in the VB.NET CodeDOM visitor Resolver and TypeVisitor added to VB they don't work, but compile updated todo Added SharpQuery- Redim statement bug fix - variable declarations work with rank and size modifiers Put the VB.NET Parser into #Develop (the ClassScout should now work for VB.NET like C#), Corrected the method body region in the VB.NET Parser, worked on the VB Forms round trip (now it should at least be 'usable') conversion

of VBTree to SD Tree code completion bug fixed: variables declared in foreach statements are now found
Changed the VB.NET Form/Usercontrol templates.
First working vb forms designer. Beware round trip not 'really' supported (class is re-generated
-> you loose ALL manual, non gui changes)
type name doesn't eat argument lists
CodeDOMVisitor for VB completed (except bugs)
VBCodeDOMVisitor started and missing methods in IASTVisitor and AbstractASTVisitor added
fixed bug in ranklist argument list instead of paramete list in raise event statement
removed debug output
repeated EOL Bug fixed in VB.Lexer array modifier improvements
updated todoarrayinitializer expression
loop control variable like comparison operator
VB.NET parser bugfix
new project descriptor.cs from Ivo Kovacka
directory attribute in the project templates should now work
fixed bug in register assembly method in the resource service
String/Bitmap resources are now loaded from addin runtime dll files the resources should be called like in the entry assembly
StringResources.resources or BitmapResources.resources
comment problem fixed, one problem remaining (eols)
hopefully fixed the text editor redraw bug...fix for the java backend binding (execute should now work when the output path is set correctly)
removed NamedAssign Tokens: there can be spaces between ":" and "=" renamed field reference expression
Correct spelling error in tray context menu+ converting field reference expressions to invocation expressions if possible
Single file compilation/running works again
enums in code completion, expr. after comment line
parsing and code completion after comment lines
extra whitespaces in code templates and newline insert bugs
reparing on closing bracket for foreach both array syntaxes (how is that spelled correctly ?) supported: a() as integer a as integer() + fixed the problem with pseudo-keyword identifier + all kinds of param modifiers are now handled corrected and validated (added a new class ParamModifiers very similar to the existing Modifiers class) + added pseudo keywords (binary, text) to the IdentifierOrKeyword production + added missing binary operator expressions: concat and power (a & b; a ^ b) + removed obsolete revolver routine + parsed a bunch of testfiles, bugs are listed in the todo.txt
invocations with call modifiers get checked
constructor being added to compilation unit
invocations parse (simple forms work)
helpapp finds the prefix if VS.NET is installed without a language pack- local declarations need modifier (dim, const, static) - added missing operator tokens (XXAssign) - added assignment statement
added missing file to project
added current language resources
fixed code completion type bug
changed version number to 98
Set default form grid size to 8x8
added new sdsetup.nsi
Code completion wizard now restarts #D after finishing the creation
applied status bar problem fix
Taken out the 'vb.net forms designer' because in it's infant stages it may lead to unexpected crashes
NewFile/New-Project dialog now recognizes last selected category and large/small icon selection is preserved
Fixed encoding support
inserted serialization complete event after deserialization in the designer (Ivo Kovacka), disabled chinese and danish (too much missing resource strings)
#D builds again...labeled statements
goto and on error statement
Started VB.NET forms designer
fixed clear all bookmarks redraw bug
Started the print routines
Goto line number dialog works again
Enabled search & replaced
divided literal token into IntegerLiteral, DoubleLiteral, SingleLiteral, StringLiteral and CharLiteral, this is needed for better parsing of the goto statement (not implemented, yet)
code completion bug fixed
Fixed text editor panel bug
constructor decl
Fixed bug in wizard build file
Fixed bug in forms designer (had something todo with begin/endload I inserted lately ... need to recheck this)
localized the messagebox builder wizard
fixed two exceptions
New Class Wizard is now working again
Included magic library in the library directory to ensure that it is compiled with the same runtime version as #D
AbstractASTVisitor completed: the whole tree should be visited now
Hopefully fixed redraw error in the textarea
Applied flat style to message box builder wizard, include new files in project dialog, select culture panel, tip of the day view
Quick XML Doc works again
bugfix in CustomComponentSideBarEnd load event should now work
corrected little bug in vb grammar
added missing file- redim statement - IdentifierOrKeyword for + attribute arguments + field access + qualified identifiers - added missing keywords to keywordlist
Fixed bottom drawing bug in the text editor
Caret position in the status bar is updated
Cut/Copy/Paste toolbar works in conjunction with the texteditor, removed some unnecessary syntax modes from data/modes (all which are now build into the text editor)
fixed 'set tabsz to 0' bug
fixed bug when .ICSharpCode directory does not exist
inserted project scout refresh method
updated texteditor sampleTypeOf and MyClass expressions
sharppad sample updated (new TextEditor)
localized highlighting editor+ moved EndOfStmt from end of every statement to Block (otherwise, singleline ifs can't be handled, the spec is again wrong here) + single-line if statement + switch statements + stack for with statements re-enabled loading of user-defined highlighting modes from config dir, updated highlighting editor for use with resource modes
moved ambiances to backend bindings
fixed delegate handling bug in #assembly-layer
reformatting if statement (hopefully) finished work on assembly scout, added menu and beautified documentation
added assembly scout option panel, added graying of tree nodes, reworked search function- erase statement - stop statement - local var declarations - error statement worked on the project import/exporter
AbstractASTVisitor enhanced: GlobalScope, TypeLevel and Statements up to ReturnStatement
code actions and template insertion is working again (returns weren't inserted?!?!), import vs.net solution should now work
fixed drawing bug in the text editor, taken out folding debug comments
AbstractASTVisitor now visits AttributesLookupTableVisitor enhanced: ForeachStatement added
textarea is now focused when calling JumpTo, first stub version of project importer/exporter
fixed formatting behavior for else clause
Fixed mouse handler bug
insight window pops up again, fixed some small text editor issues
corrected scroll to caret behaviour of the textarea when caret is placed outside a visible column
added resource preview ability to assembly scout
added parser folding strategy
Code completion and xml comment tag completion are working again
fixed compilation bug
Fold update thread works again (but don't try to toggle the fold markers :))
added the ability to view the field values in enums and to display this in attributes
Templates and formatting strategies are now working again
Morelined Selection is now drawn beyond EOL even if EOL markers are turned off
Fixed menu editor del/backspace key bug
Added SharpDevelopTextEditorProperties
correct type: Date --> DateTime
included .cur file for the text-

ditor gutter marginreplaced convert tabs to spaces radio button with a checkboxOverworked some icons (included GNU head :) see the help menu)set flatstyle to some dialogs and panelstext editor options are now working againSharpened the run iconreenabled assembly scout display bindingupdated Main.prjxsome work on the array front, rank specifiers and initializers work now Changed the texteditor property systemreadadd change from rev 230readadd changes from rev 230textarea context menu works again and code actions too- loops (for, for each, while, do) - updated visitor interface/classes enhanced the custom component resolutionAdded split command to the window menualtered some icons, fixed text editor addin fileadded text editor iconupdated some iconsre-added Main sourcesFirst version of the refactored text editor control (Expect some bugs) part 3First version of the refactored text editor control (Expect some bugs) part 3First version of the refactored text editor control (Expect some bugs) part 2First version of the refactored text editor control (Expect some bugs)added attribute support to assembly scoutAdded alphablending to some icons, inserted run icon from Markus Palme (looks great :)added preloading for #assemblies, fixed c# and vb compiler issues (spaces in path) and disabled VB casing in strings and commentsFixed NullReferenceException on selecting file mode context menuenabled iLDasm, outline and xml view in assembly scout and improved speed of it+ addressof expression + cast expressions (GetType, DirectCast, CXxx) + delegate declarations + procudctions have mostly same name as in spec + remove/add-handler statement + nested types corrected error in buildfileadded new assembly scoutfinished (almost) work on #Assembly, preparing for integrating overworked assembly scout- raise event statement - with statement - property declarations - try/catch/finally statements - throw statement - return statement - synclock statement further #Assembly and #Assembly-layer changesrenamed some #assembly classesbug fixed in #assembly, #assembly layer, started localization of highlighting editorinterface declarations- clode cleanup (enums.cs) - property and accessor declarations corrected column order (lines and words were interchanged)interfaces, enums and correct inheritiance statements - Enhanced MessageService (ShowXxxFormatted takes FormatString and FormatItems and passes the FormatString through StringParserService then calls String.Format) - Localized some strings in search/replace in files dialog - Enhanced translation (new file/project descriptions) - Added "Clear recent files/projects" menu items - Task view is shown automagically after searching in files - added property grid sorting option to forms designer option panel - enhanced browser display binding appearanceAdded a todo list, supposed to be a place for notes that otherwise would be forgotten - new token: NamedAssign (":=") - LL(1) conflict in Argument production resolved fixed anchorfields and constants according to the spec enhanced file template behavior on invalid template filesimproved visual appearance of dialogsadded pretty casing of VB keywords, added VB text editor option panel, reworked highlighting and templateschanged highlighting, added vb compiler option file for standard assembly referencesdeclare (extern subs/functions) support event support subs and functions ready fixed icon bug in the file type registerer (now #D combines, projects etc. get the correct icons assigned)adjusted the external tool panelAdjusted the new file dialog a bitxml form changed - ClientSize is taken instead of the size this should now provide a common look&feel between XP and 2000 styleAdded new code template option panel and new code template functionality (now we have code template groups which can be apply to certain file extensions)Get expression enhancedaccept/cancel button and tooltips are now working correctly in xml forms (designer load &save works and xml forms lib can load them)Fixed a bug in the forms designer for the case the file can't be parsedreadadd added parser.csinterface methods have now the correct return type not only "void"Fixed infinite loop bug in lexer which may appear under special circumstancesWorked on the tools, now should the \${...} tags workfixed option panel bug (option dialog closed on error)Completed the new tools option panel however I need to add the \${xxx} replacements in the IDEFixed bug in the parserserviceConverted the WordCount dialog to xmlformimplements/handles for sub hotkey prefixes in the xp themed controls should now workChanged some more iconsNow all classbrowser icons have an alpha channel shadowConverted more dialogs and panels to xml formmoved the folder icons one pixel upCorrected alpha channel in the folder icons, set all image lists to 32bit depthConverted the new project dialog to xmlformTranslated the new file dialog to xml form, fixed bug in the forms designer (shortcut keys), added ghengis xp theming controlsupdated to nant 0.8.3sub is working, end statement and exit statement Set the color depth for the file and project imagelist to 32 bitAdded new project and file icons, note that there is a standard naming sheme for them and I added a full C++, JScript and Java icon set tooFolding is now running on a separate threadFixed bug in the project options dialog (the option panels didn't get the correct project configurationChanged the C# icon to greenChanged some iconsI've converted some more icons to the XP style ... if someone can draw better icons he (or she) is welcome to replace my bad ones :)Translated more icons to XP Styleadded include tag to completion, corrected paramref tag Added classbrowser XP style icons (no 'official' icons ... but mostly the same style :))- corrected compile errors - missing files added constants, ...null referece exception in typeVisitor fixed (resolver.CallingClass == null)Enhanced the code completion (The expression used for the completion should be now more correct)Changed the option panel 'header' border to 3d style like in georg's highlighting editor because it looks much better than the dark line borderUpdated translation for the new load/save panelTranslated almost all xml panels to the new structure (all project options most IDE option panels)Changed the return value StorePanelContents to bool which signals if the operation was successfull (if returned false the option dialog doesn't close), converted some more panels to the new structureAdded xml form loadsave option panel, added new option panel/xml form infrastructureCorrected bug #443 (build scripts: xml form labels interchanged)Added caret position patchFixed textarea undo/redo update bugRenameCategory bugfixbug fix in code completionmembers of static members are now foundenum members are now public and listed for code completionAdded highlighting editor add-in and fixed bugs in highlighting engine. The add-in is not localized yet, I will do this later. And, the tree view still looks a bit poor, but I have no idea what icons will be the right ones :)code compiltion for interfaces now OKVBRefactory Project file changed (files were

missing)attributes improvedFixed tools bugRemoved #unit from the tools menu added nprofnew Readme.sxw, error message for ndoc doc generation without compiled projectremoved US. EnglishActivated the class browser againFixed SDI layout manager bugAdd new project configuration bug fixedCompileerror fixedinternal members are now not listedbase. is now workingdatabase generation should restart on old database fileschanged run commandCatched the textarea code completion exceptions ...Console. is now workingFixed AbstractASTVisitor.csFixed bug in AbstractASTVisitor and disabled Dr.C# for 0.97 (got exception from Drcs.dll loading.Enhanced the project browse drag&drop behaviourChanged the CSharpVisitorAdjusted some controls on option panels (They didn't show up correctly on my resolutionSpeeded up the parser/code completion layerCompletion/insight window: appears at cursor x-posTranslation updateUser interface improvements; made project options dialog derived from normal options dialogmethods and indexer in basetypes now are foundFixed classbrowser icon bugexception fixes in resolverbugfixes code complitionTransformed the configure sidebar dialog to xml form and translated it.Fixed file scout behavior so that folder tree is visible at startupIncluded project template changes from Ivo KovackaTypeVisitor updated and code clean up in ResolverAdded SharpZipLib 5.1 (with forum crypt patches)Removed old SharpZipLib version (0.4)exception bug fixedImplemented the two options on the Projects and Combines IDE option panel (show Tasklist, show Output window) that were neither saved nor applied yet.#refactoryvb sourcen addedWorked on the 'multi view' structure, separated interfaces and added base interface for viewcontents. Moved the forms designer to the new structure and some commands. Expect some bugs in the 'command' arearesolver updated, Position in AST now correct (X and Y were the wrong way)Forgot to add some files ...First step for enhanced 'tab page' functionality. I subdivided the IViewContent in 2 intefaces: IViewContent and ISubViewContent. Display bindings can now specify the ISubDisplayBinding objects which attach a 'sub view' to a main view (IViewContent) this should solve the multi tab problems (forms designer). Changed the Assembly scout display binding to the usage of the abstractviewcontent instead of using the interface directly.Added load last project on startup optionFixed Bug tracker item 432: Several Forms designer bugsFixed bug tracker item 434: The CopyLocal property of References is not preserved between settings.Fixed some #Assembly problems; added debug information files to clean target of build scriptsThe projects follow now the Dispose patternadded modifier and attributes in CSharpVisitorFixed little bug in the lookup table visitorChanged tab index in the sort options dialogAdded sort lines option into the edit->format menuResolver updatedremoved debug outputAdded Tokens.csForms designer bugfixes, work on the parser layer, added forms designer option panelAdded Reflection compatibility classes to #Assembly, added IMember interface, updated SharpAssemblyLayerresolver updatedBugfixed folding related a gutter bug, localized folding menuWorked on csharpvisitor, folding and SharpRefactoryResolver updatedWorked on folding (property bodies, corrected class bodies)Resolver updatedWorked on SharpRefactory lexer positions (now Token.EndLocation works correctly in all cases) which led to a better class body foldingChanged the file templates a littleAdded some SharpDevelop specific file templates. If you have ANY idea what for templates are missing please inform me and I'll add them. Look over these templates too ... maybe I forgot something ?Added LookupTableVisitor (for getting a variable lookup table to a file), worked on #refactoryForget to check in the Folding-Command :(Worked a bit on foldingTypeVisitor updatedrestructured the icon codon (added a 'resource' attribute). Now it is possible to specify icons inside the resource file too. Moved the loading of the 'custom' icons into the standard resources (should give some speed improvement on startup).moved iconsAdded ExecuteCommand option in the project options (now for example 'dll' projects can execute something), fixed possible dead lock in the run command, added Dr.C# to the distribution, fixed a little bug in the resolver (doesn't work but shouldn't crash anymore)Fixed the well known key bug with in the forms designer (I missed one occasion), worked a bit on the translation, tried to port as many as possible MessageBox.Show calls to the Message-ServiceFixed bug in the MenuService (now submenu builders should be working correctlyLittle speed improvement for the forms designer back to code switch (no more redrawing). I've seen that there is MANY room for speed improvements: Don't use the #D parser service, use #Refactory directly (it is already needed to do this), the forms designer parses too often and many more-Worked on folding and corrected some parser positionsCondons can now choose if they handle conditions on their own using the HandleConditions property. The condition change caused many bugs all over other parts of sharpdevelop because the conditions needed to get handled. Now the handling is centralized again and all parts should now work as expected again. MenuItems and ToolStripItem codons however do handle the conditions on their own and any new codons may do so.updated path to nunit 2.1 path in the run commandIntegrated NUnit 2.1Removed nunit 2.0Introduced the MenuService, fixed dialog panel bug (conditions were not recognized correctly by the option dialogsFixed bug in the submenu update which caused a View menu bugreload command now doesn't work on the startpage and it is no longer possible to view more than one startpageThe syntax highlighting menuitem builder working againTool Bar 'update' changes'Implemented select next/prev window actionFixed bug in the CommandBar library (cursor keys left/right in the main menu didn't recognize 'invisible' menusBugfix in the forms designer (C# forms designer window didn't show up)changed the casing of the magic libraryMenu shortcuts should now workCleaned up compiler warningsChanged the menu subsystem: splitted the menus up into ICheckableMenuCommand and IMenuCommand (The ICheckable has IsCheckable, the other only IsEnabled) and ICheckableMenuCommand inherits from IMenuCommand.Updated the menu engine, should work better now ... but I still hunt bugs here and thereAdded SdMenu.csIntegrated the CommandBar library further (menus are replaced), shortcuts don't work + maybe a context menu is missing here and there + enabled/checked/removed item currently NOT workingRemoved the UtilityLibrary from SharpDevelop and included Lutz Roeders CommandBar library insteadWorked on the new resolver (Code Completion is still not working)Nothing special. Just svn commit hook tested. Fixed Tracker item 431: About Box / Version info improvement requestsall tools and external projects are now stored in a subdirectory

of bin/Removed CSharpParser.dll replaced with #Refactory, added the conversion layer to the CSharpBackend binding, should
broke Code Completion but forms designer and class browser are still workingLocalized the SidebarView categories, updated the
sidebar configuration files to newer standardsfixed small bug in CombineActiveConfigurationTypeConverterLocalized the file
type registerer addin :)Applied assembly deploy patch from Alexandre Semenovupdated lexer, localized property grid (localiza-
tion classes in Gui.Components.LocalizedPropertyGrid, bugfixed some property grid optionsThe lexer should now recognize all
comments, pre processor directives correctly and store them :)Included nunit (integrated it too), removed #unit, put nprof to the
toolsfixed BackendBinding Combine and renamed some DisplayBinding combinesSome more pretty printer chanesHopefully
finally this program will stop being trash. Here are the proper files again, in their proper place, I hopeThis causes SharpDevelop
to give an error instead of throwing an exception message when a file from a combine (or the combine itself) cannot be found,
when using either Recent Files or Open Combine.Fixed a bug wherein, if a project inside the current combine cannot be found...
ack, TortoiseSVN doesn't have this as multiline... idiotic... anyway, in that case it would throw an exception and not have all the
projects which had not loaded, BUT the exception would be trapped and the current combine set to the invalid one. So, if you
opened a combine with an invalid project, it would give the error. BUT if you tried to open it *again*, it would try to save the set-
tings, goof up the project file, and load and throw an exception. This is fixed. I noticed it because the BackendBindings combine
is screwed up for some reason.Removed some unnecessary filesRefactored SharpRefactory :) / worked on pretty printerAnother
little #Assembly changechanges to ambience service, parser interfaces and #assemblyWork on the C# pretty printerUpdated
some language strings and inserted more & shortcutsFixed the bug that removed components in the designer with an upper case
letterApplied menu designer delete key bugfix from the forumDeleted some assembliesFixed the C# grammar in #Refacotry
for parsing through to EOF (now all C# bugs should be recognized)The component combobox (property pad) now updates on
component renameChanged version to 0.97Included our modified coco (SharpCoco) to toolsInitial Import
