# Automated Build Generator - Bug #2021 Projects that contain pom.xml and CMakeLists.txt get treated as maven project only

09/24/2014 10:38 AM - S. Meyer zu Borgsen

Priority::::::::::::::::::::::::::::::::::::	Chattan	Deschued	Out of a last	00/04/0014	
Assignee:       J. Moringen       % Done:       100%         Category:       Project Analysis       Estimated time:       0.00 hour         Target version:       0.3       Description       Secondary Se	Status:	Resolved	Start date:	09/24/2014	
Category:       Project Analysis       Estimated time:       0.00 hour         Target version:       0.3       0.00 hour         Description       Image: Contains a point for building the java part and a CMakeLists for the opp part.         This results in incorrect dependency resolution.       Even if the template is set to cmake/cmake-cpp, the project gets only analyzed for the maxen part.         See bit-opp.project       Related to Automated Build Generator - Bug # 2022: Incorrect pkg-config requil       Rejected       09:24:2014         Associated revision       Revision c870a5e4 - 1008/2014 05:26 PM - J. Moringen       Multiple natures/nature specification in scrianalysis*/lisp         Musteple natures/nature specification in scrianalysis*/lisp       isses revision c970a5e4 - 1008/2014 05:26 PM - J. Moringen         Multiple natures/nature specification in scrianalysis*/lisp       escrianalysis lisp (guess-project-natures): renamed project-kind ~ guess project-natures): renamed project-kind ~ guess project-natures i: return a list of all applicable natures       escrianalysis/gitilisp (analyze uri eq] :sm): likewise         • scrianalysis/gitilisp (analyze uri eq] :sm): likewise       escrianalysis/gitilisp (analyze uri eq] :sm): likewise         • scrianalysis/gitilisp (analyze uri eq] :sm): likewise       escrianalysis/gitilisp (analyze uri eq] :sm): likewise         • scrianalysis/gitilisp (analyze uri eq] :sm): likewise       escrianalysis/gitilisp (analyze uri eq] :sm): likewise         • scrianalysis/gitilisp (analyze uri eq]	-				
Target Version:       0.3         Description         The BTL project (https://projects.cit.ec.uni-bielefeld.de/git/bit.git)         contains a pom for building the java part and a CMakeLists for the copp part.         This results in incorrect dependency resolution.         Even if the template is set to cmake/cmake-opp, the project gets only analyzed for the maven part.         See bit-opp.project         Related issues:         Related to Automated Build Generator - Bug # 2022: Incorrect pkg-config requi       Rejected       09/24/2014         Associated revisions         Revision c970a5e4 - 10/08/2014 05:26 PM - J. Moringen         Multiple natures/nature specification in src/analysis/".lisp         fixes #2021         • src/analysis/analysis/lisp (guess-project-natures): renamed project-kind -> guess-project-natures): renamed (analyze pathname cons): new method: analyze according to all natures (analyze pathname cons): new method: analyze according to all natures (analyze pathname cons): new method: analyze according to all natures (analyze pathname cons): new method: analyze according to all natures (analyze pathname cons): new method: analyze according to all natures (analyze pathname cons): new method: analyze according to all natures (analyze pathname cons): new method: analyze according to all natures (analyze pathname cons): new method: analyze according to all natures (analyze pathname cons): new method: analyze according to all natures (analyze pathname cons): new method: analyze according to all natures (analyze pathname cons): new method: analyze according to all natures (analyze pathname conalyze): accor	-	-			
Description         The BTL project (https://projects.cit.ec.uni-bielefeld.de/git/bit.git)         contains a pom for building the java part and a CMakeLists for the cpp part.         This results in incorrect dependency resolution.         Even if the template is set to cmake/cmake-opp, the project gets only analyzed for the maven part.         See bil-opp.project         Related issues:         Related issues:         Related issues:         Revision c9706564 - 1008/2014 05:26 PM - J. Moringen         Multiple natures/nature specification in src/analysis/* lisp         fixes #2021         • src/analysis/analysis.lisp (guess-project-natures): renamed         project-kind - guess-project-natures): renamed         gradyze pathname eql :auto): handle multiple natures         (analyze pathname eql :auto): handle multiple natures         (analyze pathname eql :auto): handle multiple natures         • src/analysis/jiteling (analyze ui eql :qit): pass unknown keyword         arguments to `analyze' iso call         • src/analysis/jiteling (analyze ui eql :qit): jites jites iso iso call         • src/analysis/jiteling (analyze ui eql :qit): jites jites iso in src/analysis/analys			Estimated time:	0.00 hour	
The BTL project (https://projects.clite.cu.nii.bielefeld.de/git/bill.git/ contains a pom for building the java part and a CMakeLists for the copp part. This results in incorrect dependency resolution. Even if the template is set to cmake/cmake-cpp, the project gets only analyzed for the maven part. See btl-cpp.project <b>Related issues:</b> Related issues: Related to Automated Build Generator - Bug # 2022: Incorrect pkg-config requi Rejected 09/24/2014 <b>Associated revisions</b> Revision c3708564 - 1008/2014 05:26 PM - J. Moringen Multiple natures/nature specification in src/analysis/".lisp fitxes #2021 • src/analysis/analysis.lisp (guess-project-natures): renamed project kind - guess-project-natures): renamed project kind - guess-project-natures): renamed groject kind - guess-project-natures): renamed project kind - guess-project-natures: return a list of all applicable antures • src/analysis/git.lisp (analyze uri eql :archive): likewise • src/analysis/git.lisp (analyze uri eql :archive): likewise • src/analysis/git.lisp (analyze in egl :archive): likewise • src/analysis/git.lisp (analyze in egl :archive): likewise Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp rels #2021 • src/analysis/git.lisp (analyze pathame eql :auto): when merging		0.3			
contains a pom for building the java part and a CMakeLists for the opp part. This results in incorrect dependency resolution. Even if the template is set to cmake/cmake-cpp, the project gets only analyzed for the maven part. See bil-cpp.project  Related Issues: Related Issues: Related 1b Automated Build Generator - Bug # 2022: Incorrect pkg-config requi Rejecte 09/24/2014  Associated revisions Revision c970a5e4 - 10/08/2014 05:26 PM - J. Moringen  Multiple natures/nature specification in src/analysis/*.lisp fixes #2021  - src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -> guess-project-natures): renamed (analyze pathname eqi :auto): handle multiple natures (analyze pathname eqi :auto): handle multiple natures (analyze pathname eqi :auto): handle multiple natures (analyze pathname eqi :auto): likewise - src/analysis/lilep (analyze uri eqi :git): pass unknown keyword arguments to 'analyze': auto call - seroianalysis/archive.lisp (analyze project): pass list of natures to 'analyze' Revision 4a874e6b - 1009/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp rofs #2021 - src/analysis/analyze lisp (analyze pathname eqi :auto): when merging	Description				
contains a pom for building the java part and a CMakeLists for the opp part. This results in incorrect dependency resolution. Even if the template is set to cmake/cmake-cpp, the project gets only analyzed for the maven part. See bil-cpp.project  Related Issues: Related Issues: Related 1b Automated Build Generator - Bug # 2022: Incorrect pkg-config requi Rejecte 09/24/2014  Associated revisions Revision c970a5e4 - 10/08/2014 05:26 PM - J. Moringen  Multiple natures/nature specification in src/analysis/*.lisp fixes #2021  - src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -> guess-project-natures): renamed (analyze pathname eqi :auto): handle multiple natures (analyze pathname eqi :auto): handle multiple natures (analyze pathname eqi :auto): handle multiple natures (analyze pathname eqi :auto): likewise - src/analysis/lilep (analyze uri eqi :git): pass unknown keyword arguments to 'analyze': auto call - seroianalysis/archive.lisp (analyze project): pass list of natures to 'analyze' Revision 4a874e6b - 1009/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp rofs #2021 - src/analysis/analyze lisp (analyze pathname eqi :auto): when merging					
This results in incorrect dependency resolution. Even if the template is set to cmake/cmake-cpp, the project gets only analyzed for the maven part. See btl-cpp.project Related issues: Related issues: Related issues: Related to Automated Build Generator - Bug # 2022: Incorrect pkg-config requi Rejected 09/24/2014 Associated revisions Revision 0970a564 - 10/09/2014 05:26 PM - J. Moringen Multiple natures/nature specification in src/analysis*.lisp fixes #2021 - src/analysis/analysis.lisp (guess-project-natures): renamed project-kinds guess-project-natures; return a list of all applicable natures (analyze pathname cons): www.method: analyze according to all natures - src/analysis/git.lisp (analyze uri eql :git); pass unknown keyword arguments to `analyze' :auto call - src/analysis/git.lisp (analyze uri eql :git); pass unknown keyword arguments to `analyze' :auto call - src/analysis/arahuyse' :uto call - src/analysis/arahuyse' :uto call - src/analysis/arahuyse' :project): pass list of natures to `analyze' Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 - src/analysis/arahuyse is in src/analysis/analysis.lisp					
Even if the template is set to cmake/cmake-cpp, the project gets only analyzed for the maven part. See bil-cpp.project  Related issues: Related to Automated Build Generator - Bug # 2022: Incorrect pkg-config requi Rejected 09:24:2014  Associated revisions Revision c970a5e4 - 10:08/2014 05:26 PM - J. Moringen  Multiple natures/nature specification in src/analysis/*.lisp fixes #2021  - src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -> guess-project-natures; return a list of all applicable natures (analyze pathname econs): new method: analyze according to all natures sercianalysis/git.lisp (analyze uri eq/:spit): pass unknown keyword arguments (analyzei/auto) cal serc/analysis/git.lisp (analyze uri eq/:spit): pass list of natures to `analyze' auto cal serc/analysis/git.lisp (analyze uri eq/:archive): inewise serc/analysis/git.lisp (analyze uri eq/:archive): inpass list of natures to `analyze'  Revision 4a674e6b - 10:09/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021  - src/analysis/analysis.lisp (analyze pathname eq!:auto): when merging	contains a pom for b	uilding the java part and a CMakeLi	sts for the cpp part.		
Even if the template is set to cmake/cmake-cpp, the project gets only analyzed for the maven part. See bil-cpp.project  Related issues: Related to Automated Build Generator - Bug # 2022: Incorrect pkg-config requi Rejected 09:24:2014  Associated revisions Revision c970a5e4 - 10:08/2014 05:26 PM - J. Moringen  Multiple natures/nature specification in src/analysis/*.lisp fixes #2021  - src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -> guess-project-natures; return a list of all applicable natures (analyze pathname econs): new method: analyze according to all natures sercianalysis/git.lisp (analyze uri eq/:spit): pass unknown keyword arguments (analyzei/auto) cal serc/analysis/git.lisp (analyze uri eq/:spit): pass list of natures to `analyze' auto cal serc/analysis/git.lisp (analyze uri eq/:archive): inewise serc/analysis/git.lisp (analyze uri eq/:archive): inpass list of natures to `analyze'  Revision 4a674e6b - 10:09/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021  - src/analysis/analysis.lisp (analyze pathname eq!:auto): when merging					
See btl-cpp.project         Related issues:         Related to Automated Build Generator - Bug # 2022: Incorrect pkg-config requi       Rejected       09/24/2014         Associated revisions         Revision c970s5e4 - 10/08/2014 05:26 PM - J. Moringen         Multiple natures/nature specification in src/analysis/*.lisp         fixes #2021         • src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -> guess-project-natures; return a list of all applicable natures         (analyze pathname cons): new method: analyze according to all natures         • src/analysis/git.lisp (analyze uri eql :gii): pass unknown keyword arguments to 'analyze' auto call         • src/analysis/git.lisp (analyze uri eql :archive): Ikewise         • src/analysis/git.lisp (analyze uri eql: archive): Ikewise         • src/analysis/archive.lisp (analyze uri eql: archive): Ikewise         • src/analysis/archive.lisp (analyze tri eql: archive): Ikewise         • src/analysis/arch					
Related issues:       Rejected       09/24/2014         Associated revisions	Even if the template	is set to cmake/cmake-cpp, the proj	ect gets only analyzed for the m	aven part.	
Related issues:       Rejected       09/24/2014         Associated revisions	<b>6</b>				
Related to Automated Build Generator - Bug # 2022; Incorrect pkg-config requi       Rejected       09/24/2014         Associated revisions	See bil-cpp.project				
Related to Automated Build Generator - Bug # 2022; Incorrect pkg-config requi       Rejected       09/24/2014         Associated revisions	Deleter				
Associated revisions         Revision c970a5e4 - 10/08/2014 05:26 PM - J. Moringen         Multiple natures/nature specification in src/analysis/*.iisp         fixes #2021         - src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -> guess-project-natures; return a list of all applicable natures (analyze pathname eql :auto): handle multiple natures (analyze pathname cons): new method: analyze according to all natures         - src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to `analyze' auto call         - src/analysis/git.lisp (analyze uri eql :snr): likewise         - src/analysis/git.lisp (analyze uri eql :antibic): likewise         - src/analysis/git.lisp (analyze uri eql :antibic): likewise         - src/analysis/archive.lisp (analyze uri eql :antibic): likewise         - src/analysis/git.lisp (analyze uri eql :antibic): likewise         - src/analysis/git.lisp (analyze uri eql :antibic): likewise         - src/analysis/git.lisp (analyze uri eql :archive): likewise         - src/analysis/git.lisp (analyze uri eql :archive): likewise         - src/analysis/git.lisp (analyze uri eql: archive): likewise         - src/analysis/git.lisp (analyze intervertion in atures to `analyze'         Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen         Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp         refs #2021         - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging			alan anafin yang d	Delected 00/04/0011	
Revision c970a5e4 - 10/08/2014 05:26 PM - J. Moringen         Multiple natures/nature specification in src/analysis/*.lisp         fixes #2021         - src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -> guess-project-natures; return a list of all applicable natures (analyze pathname eql :auto): handle multiple natures (analyze pathname cons): new method: analyze according to all natures         - src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to `analyze' :auto call         - src/analysis/git.lisp (analyze uri eql :srn): likewise         - src/analysis/git.lisp (analyze uri eql :archive): likewise         - src/analysis/git.lisp (analyze uri eql :archive): likewise         - src/analysis/archive.lisp (analyze uri eql :archive): pass list of natures to `analyze'         Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen         Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp         refs #2021         - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging	Related to Automated E	Build Generator - Bug # 2022: Incorrect	okg-config requi	Rejected 09/24/2014	
Revision c970a5e4 - 10/08/2014 05:26 PM - J. Moringen         Multiple natures/nature specification in src/analysis/*.lisp         fixes #2021         - src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -> guess-project-natures; return a list of all applicable natures (analyze pathname eql :auto): handle multiple natures (analyze pathname cons): new method: analyze according to all natures         - src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to `analyze' :auto call         - src/analysis/git.lisp (analyze uri eql :srn): likewise         - src/analysis/git.lisp (analyze uri eql :archive): likewise         - src/analysis/git.lisp (analyze uri eql :archive): likewise         - src/analysis/archive.lisp (analyze uri eql :archive): pass list of natures to `analyze'         Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen         Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp         refs #2021         - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging					
Multiple natures/nature specification in src/analysis/*.lisp fixes #2021					
fixes #2021 - src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -> guess-project-natures; return a list of all applicable natures (analyze pathname eql :auto): handle multiple natures (analyze pathname cons): new method: analyze according to all natures - src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to 'analyze': auto call - src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to 'analyze': auto call - src/analysis/git.lisp (analyze uri eql :archive): likewise - src/analysis/archive.lisp (analyze uri eql :archive): likewise - src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze' Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging	Revision c9/0abe4 - 1	0/08/2014 05:26 PM - J. Moringen			
fixes #2021 - src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -> guess-project-natures; return a list of all applicable natures (analyze pathname eql :auto): handle multiple natures (analyze pathname cons): new method: analyze according to all natures - src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to 'analyze': auto call - src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to 'analyze': auto call - src/analysis/git.lisp (analyze uri eql :archive): likewise - src/analysis/archive.lisp (analyze uri eql :archive): likewise - src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze' Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging					
<ul> <li>src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -&gt; guess-project-natures; return a list of all applicable natures (analyze pathname eq! :auto): handle multiple natures (analyze pathname cons): new method: analyze according to all natures</li> <li>src/analysis/git.lisp (analyze uri eq! :git): pass unknown keyword arguments to `analyze' :auto call</li> <li>src/analysis/git.lisp (analyze uri eq! :svn): likewise</li> <li>src/analysis/git.lisp (analyze uri eq! :svn): likewise</li> <li>src/analysis/git.lisp (analyze uri eq! :svn): likewise</li> <li>src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze'</li> <li>Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen</li> <li>Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp</li> <li>refs #2021</li> <li>src/analysis/analysis.lisp (analyze pathname eq! :auto): when merging</li> </ul>	Multiple natures/nature	specification in src/analysis/*.lisp			
<ul> <li>src/analysis/analysis.lisp (guess-project-natures): renamed project-kind -&gt; guess-project-natures; return a list of all applicable natures (analyze pathname eq! :auto): handle multiple natures (analyze pathname cons): new method: analyze according to all natures</li> <li>src/analysis/git.lisp (analyze uri eq! :git): pass unknown keyword arguments to `analyze' :auto call</li> <li>src/analysis/git.lisp (analyze uri eq! :svn): likewise</li> <li>src/analysis/git.lisp (analyze uri eq! :svn): likewise</li> <li>src/analysis/git.lisp (analyze uri eq! :svn): likewise</li> <li>src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze'</li> <li>Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen</li> <li>Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp</li> <li>refs #2021</li> <li>src/analysis/analysis.lisp (analyze pathname eq! :auto): when merging</li> </ul>					
<ul> <li>project-kind -&gt; guess-project-natures; return a list of all applicable natures <ul> <li>(analyze pathname eql :auto): handle multiple natures</li> <li>(analyze pathname cons): new method: analyze according to all natures</li> </ul> </li> <li>src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to `analyze' :auto call</li> <li>src/analysis/git.lisp (analyze uri eql :svn): likewise</li> <li>src/analysis/archive.lisp (analyze uri eql :archive): likewise</li> <li>src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze'</li> </ul> <li>Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen</li> <li>Backport: Fixes for multi-nature analysis in src/analysis/lisp</li> <li>refs #2021 <ul> <li>src/analysis/analysis.lisp (analyze pathname eql :auto): when merging</li> </ul> </li>	fixes #2021				
<ul> <li>project-kind -&gt; guess-project-natures; return a list of all applicable natures <ul> <li>(analyze pathname eql :auto): handle multiple natures</li> <li>(analyze pathname cons): new method: analyze according to all natures</li> </ul> </li> <li>src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to `analyze' :auto call</li> <li>src/analysis/git.lisp (analyze uri eql :svn): likewise</li> <li>src/analysis/archive.lisp (analyze uri eql :archive): likewise</li> <li>src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze'</li> </ul> <li>Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen</li> <li>Backport: Fixes for multi-nature analysis in src/analysis/lisp</li> <li>refs #2021 <ul> <li>src/analysis/analysis.lisp (analyze pathname eql :auto): when merging</li> </ul> </li>	ara (analyzaia (analyza	in line (augus project poturos), renomes			
<ul> <li>natures <ul> <li>(analyze pathname eql :auto): handle multiple natures</li> <li>(analyze pathname cons): new method: analyze according to all natures</li> <li>src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to `analyze' :auto call</li> <li>src/analysis/git.lisp (analyze uri eql :svn): likewise</li> <li>src/analysis/archive.lisp (analyze uri eql :archive): likewise</li> <li>src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze'</li> </ul> </li> <li>Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen</li> <li>Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp</li> <li>refs #2021 <ul> <li>src/analysis/analysis.lisp (analyze pathname eql :auto): when merging</li> </ul> </li> </ul>					
<ul> <li>(analyze pathname eql :auto): handle multiple natures</li> <li>(analyze pathname cons): new method: analyze according to all natures</li> <li>src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword</li> <li>arguments to `analyze' :auto call</li> <li>src/analysis/git.lisp (analyze uri eql :svn): likewise</li> <li>src/analysis/archive.lisp (analyze uri eql :archive): likewise</li> <li>src/commandline-interface/main.lisp (analyze-project): pass list of</li> <li>natures to `analyze'</li> </ul> Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 <ul> <li>src/analysis/analysis.lisp (analyze pathname eql :auto): when merging</li> </ul>			phicable		
<ul> <li>(analyze pathname cons): new method: analyze according to all natures</li> <li>src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to `analyze' :auto call</li> <li>src/analysis/git.lisp (analyze uri eql :svn): likewise</li> <li>src/analysis/archive.lisp (analyze uri eql :archive): likewise</li> <li>src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze'</li> </ul> Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 src/analysis/analysis.lisp (analyze pathname eql :auto): when merging		a agl (auto), bandle multiple potures			
<ul> <li>src/analysis/git.lisp (analyze uri eql :git): pass unknown keyword arguments to `analyze' :auto call</li> <li>src/analysis/git.lisp (analyze uri eql :svn): likewise</li> <li>src/analysis/archive.lisp (analyze uri eql :archive): likewise</li> <li>src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze'</li> <li>Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen</li> <li>Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp</li> <li>refs #2021</li> <li>src/analysis/analysis.lisp (analyze pathname eql :auto): when merging</li> </ul>					
arguments to `analyze' :auto call - src/analysis/git.lisp (analyze uri eql :svn): likewise - src/analysis/archive.lisp (analyze uri eql :archive): likewise - src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze'  Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging			-		
<ul> <li>src/analysis/git.lisp (analyze uri eql :svn): likewise</li> <li>src/analysis/archive.lisp (analyze uri eql :archive): likewise</li> <li>src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze'</li> </ul> Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 <ul> <li>src/analysis/analysis.lisp (analyze pathname eql :auto): when merging</li> </ul>			/word		
<ul> <li>src/analysis/archive.lisp (analyze uri eql :archive): likewise</li> <li>src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze'</li> <li>Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen</li> <li>Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp</li> <li>refs #2021</li> <li>src/analysis/analysis.lisp (analyze pathname eql :auto): when merging</li> </ul>	-				
<ul> <li>src/commandline-interface/main.lisp (analyze-project): pass list of natures to `analyze'</li> <li>Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen</li> <li>Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp</li> <li>refs #2021</li> <li>src/analysis/analysis.lisp (analyze pathname eql :auto): when merging</li> </ul>					
natures to `analyze' Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging	-	,			
Revision 4a874e6b - 10/09/2014 07:12 PM - J. Moringen Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging			s list of		
Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging	natures to `analyz	ze'			
Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging					
Backport: Fixes for multi-nature analysis in src/analysis/analysis.lisp refs #2021 - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging					
refs #2021 - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging	Revision 4a874e6b - 1	0/09/2014 07:12 PM - J. Moringen			
refs #2021 - src/analysis/analysis.lisp (analyze pathname eql :auto): when merging					
- src/analysis/analysis.lisp (analyze pathname eql :auto): when merging	Backport: Fixes for mult	ti-nature analysis in src/analysis/analysi	s.lisp		
- src/analysis/analysis.lisp (analyze pathname eql :auto): when merging					
	refs #2021				
	- src/analysis/analys	is.lisp (analyze pathname eql :auto): wh	en merging		
anaryolo roodilo, produco a pilot, not an allot, inco logging	analysis results, p	produce a plist, not an alist; fixed logging	l		

# Revision 3b8f2655 - 10/09/2014 07:16 PM - J. Moringen

Fixes for multi-nature analysis in src/analysis/analysis.lisp

# refs #2021

- src/analysis/analysis.lisp (analyze pathname eql :auto): when merging analysis results, produce a plist, not an alist; fixed logging

### History

## #1 - 09/24/2014 08:18 PM - J. Moringen

- Related to Bug #2022: Incorrect pkg-config requirement extraction for cmake-cpp projects added

# #2 - 09/26/2014 06:09 PM - J. Moringen

- Status changed from New to Feedback
- Target version set to 0.3

One solution could be performing all applicable analyses and merging the results.

@Johannes: do you see any problems (e.g. for RST) with this approach?

For example, for BTL these results could be merged:

CL-USER> (jenkins.analysis:analyze #P"/tmp/btl/" :auto)

```
(
```

```
; Maven result
(:VERSIONS
((:MAIN ("de.unibi.citec.clf/btl" "2.0-SNAPSHOT")))
:PROVIDES
((:MAVEN "de.unibi.citec.clf/btl" (2 0 "SNAPSHOT"))
 (:MAVEN "de.unibi.citec.clf/btl-xml" (2 0 "SNAPSHOT"))
 (:MAVEN "de.unibi.citec.clf/btl-base" (2 0 "SNAPSHOT"))
 (:MAVEN "de.unibi.citec.clf/btl-rst" (2 0 "SNAPSHOT")))
:REQUIRES
((:MAVEN "log4j/log4j" (1 2 16))
 (:MAVEN "junit/junit/test" (4 10))
 (:MAVEN "java3d/vecmath" (1 3 1))
 (:MAVEN "org.apache.commons/commons-lang3" (3 1))
 (:MAVEN "rsb/rsb" (0 11 "SNAPSHOT"))
 (:MAVEN "rsb/rst-sandbox" (0 11 "SNAPSHOT"))
 (:MAVEN "rsb/rst" (0 11 "SNAPSHOT"))
 (:MAVEN "org.reflections/reflections" (0 9 8))
 (:MAVEN "de.unibi.citec.clf/btl-base" (2 0 "SNAPSHOT"))
 (:MAVEN "xom/xom" (1 2 5))
 (:MAVEN "org.easymock/easymockclassextension/test" (2 5 2))
 (:MAVEN "org.easymock/easymock/test" (2 5 2))
 (:MAVEN "org.apache.commons/commons-lang3" (3 1))
 (:MAVEN "java3d/vecmath" (1 3 1))
 (:MAVEN "junit/junit/test" (4 10))
 (:MAVEN "log4j/log4j" (1 2 16)))
:URL "https://projects.cit-ec.uni-bielefeld.de/projects/btl"
:LICENSE "LGPL-3"
```

:PROPERTIES (("project.build.sourceEncoding" . "UTF-8") ("license.licenseName" . "btl\_license") ("license.licenseResolver" . "\${project.baseUri}/src/license")))

; CMake result (:VERSIONS ((:MAIN NIL)) :PROVIDES ((:CMAKE "BTL" NIL) (:CMAKE "btl" NIL) (:CMAKE "BTL-BASE" NIL) (:CMAKE "btl-base" NIL) (:CMAKE "BTL-RST" NIL) (:CMAKE "btl-rst" NIL) (:CMAKE "BTL-XML" NIL) (:CMAKE "btl-xml" NIL) (:PKG-CONFIG "BielefeldTypeLibrary" ("@BTL\_VERSION@")) (:PKG-CONFIG "BielefeldTypeLibrary" ("@BTL\_VERSION@")) (:PKG-CONFIG "BielefeldTypeLibrary" ("@BTL\_VERSION@")) (:PKG-CONFIG "BielefeldTypeLibrary" ("@BTL\_VERSION@"))) :REQUIRES NIL :PROPERTIES ((:LICENSE . "LGPL-3")))

)

#### #3 - 09/26/2014 06:49 PM - J. Wienke

I think that for RST this doesn't occur since the pom file is generated by cmake and hidden in subfolders.

Merging might also be the wrong behavior since in case the project only builds the C++ parts, it should not depend on java artifacts. Maybe there should be a way to couple the template to the performed repository analysis so that for a cpp project only analysis of cmake is performed and vice versa.

### #4 - 09/26/2014 06:52 PM - J. Moringen

Merging might also be the wrong behavior since in case the project only builds the C++ parts, it should not depend on java artifacts. Maybe there should be a way to couple the template to the performed repository analysis so that for a cpp project only analysis of cmake is performed and vice versa.

Your suggestion implies having two or more recipes, one for each "nature" (in Eclipse speak), of such projects?

### #5 - 09/26/2014 06:57 PM - J. Wienke

### Jan Moringen wrote:

Merging might also be the wrong behavior since in case the project only builds the C++ parts, it should not depend on java artifacts. Maybe there should be a way to couple the template to the performed repository analysis so that for a cpp project only analysis of cmake is performed and vice versa.

That's probably necessary? Or does it work to have a project that uses the maven template as well as the cmake template in parallel? I would suspect a lot of unexpected results from that.

Actually looking at the BTL source code, it seems that the CMake script triggers maven manually. So in this case, merging the dependencies is probably ok, but I don't think one can generalize this across all different projects. So it would probably be good if there was a variable in the generator to select the analyzers to use. Each template can then set the one analyzer it is designed for. In case of BTL, the user then still has the option to additionally configure the maven analyzer. In case multiple analyzers exists as chosen by the user, merging dependencies is then ok, to my mind.

# #6 - 10/09/2014 01:10 PM - J. Moringen

- Status changed from Feedback to Resolved
- % Done changed from 0 to 100

Applied in changeset commit:c970a5e422ea540920a3b0e8a7a150772c366a43.