PIPS

Tutorial to use PIPS with Eclipse $_{Rapport \ technique \ E-334}$

Nelson Lossing

January 10, 2014

Contents

1	Introduction	3
2	Prerequisites	3
3	Add the repositories of PIPS and their dependencies in Eclipse	3
4	Make the projects for PIPS	5
5	Configure your projects	8
6	How can we compile PIPS with Eclipse?	12
7	How can we debug PIPS with Eclipse?	13
8	Why can I only launch the debug for tpips only once?	15
9	Why debug step go randomly top and down?	15
10	How can I ask Eclipse to not recompil before launch debug?	15
11	How can I use SVN in Eclipse?	16

1 Introduction

The goal of this tutorial is to explain how we can use Eclipse as IDE to work on PIPS. You can also read the PIPS developer_guide ¹ and the official web site of PIPS http://pips4u.org for more information.

This document will explain how we can configure the project linked with the SVN sources. But also how to compile and to launch PIPS with Eclipse and so use the debugger provided by Eclipse.

This tutorial and the screen shoots are done with Eclipse JUNO version.

2 Prerequisites

Your Eclipse has to support C/C++ project.

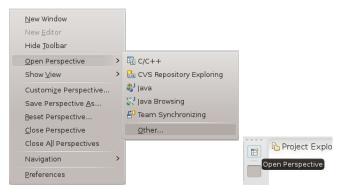
You need to install *subversive* in your Eclipse. You can do this in "*Help/In-stall New Software...*" and search svn to find subversive.

When you relaunch Eclipse after the first installation of *subversive*, it will probably ask you to install a connector. You can install the last version of SVNKit.

3 Add the repositories of PIPS and their dependencies in Eclipse

The first thing to do is to add the SVN repositories of PIPS and their dependencies.

- 1. Switch to the SVN Repository Exploring's perspective.
 - (a) You can find this perspective in the perspective windows in menu "Windows/Open Perspective/Other..." (right picture) or button Open Perspective (left picture).



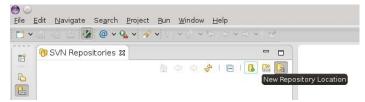
(b) Choose SVN Repository Exploring. It will also add the shortcut in the Perspective view.

¹pdf version: http://www.cri.ensmp.fr/pips/developer_guide.htdoc/developer_ guide.pdf

HTML version: http://www.cri.ensmp.fr/pips/developer_guide.htdoc

⊜ ⊙	Open Perspective	$\odot \odot \otimes$				
Ec/C++						
🖶 CVS Re	pository Exploring					
🕸 Debug						
品 Git Repo	ository Exploring					
🐉 Java						
💱 Java Bro	owsing					
🎦 Java Typ	e Hierarchy					
🖌 LaTeX						
🖫 Remote System Explorer						
🔓 Resource (default)						
🖶 SVN Repository Exploring						
<mark>≓</mark> 0 Team S	ynchronizing					
	ок	Cancel				

- 2. Add The repository location of PIPS (https://svn.cri.ensmp.fr/svn/pips/).
 - (a) Open the New Repository Location wizard.



(b) Put the URL of PIPS https://svn.cri.ensmp.fr/svn/pips/ and your authentication login of PIPS.

0	New Repository Loca	tion	\odot \otimes \otimes
Define the SVN	ory Location Information I repository location information ngs for proxy and svn+ssh, http		SVN
URL https://	anced SSE Settings SSE Set scm.cri.ensmp.fr/svn/pips/ Label repository URL as the label	tings	Browse
Use a <u>c</u> u	stom label:		
Password:	Authentication ossing		~
	hentication (could trigger secur our security data, please see <u>"s</u>	5 5 1	
	als For: <a>Repository Location> pository Location on finish	Re	set Changes
?		Einish	Cancel

(c) click Finish.

3. Do the same for linear (https://svn.cri.ensmp.fr/svn/linear/) and Newgen (https://svn.cri.ensmp.fr/svn/newgen/).

You will normally have something like that at the end:

	😚 SVN Repositories 🕱 🗧 🗖
	🖞 🔶 🔶 📔 🛯 👪 👫
6	https://scm.cri.ensmp.fr/svn/linear/
	https://scm.cri.ensmp.fr/svn/newgen/
£0	https://scm.cri.ensmp.fr/svn/pips/
Ec	
*	

4 Make the projects for PIPS

You can now make a project that will be managed by Eclipse for PIPS under SVN.

- 1. Go back to the Ressource Perspective or to the C/C++ Perspective (see Sec 3 step 1a to change the perspective).
- 2. Add a new SVN project for PIPS
 - (a) Create a new project (File/New/Project).
 - (b) Select a SVN project.

0	New Project	\odot \otimes \otimes			
Select a wiza	rd				
Create a new p repository.	project by checking out an existing project from SVN				
<u>W</u> izards:					
type filter text		•3			
👌 🗁 General					
> 🗁 C/C++					
> 🗁 CVS					
> 🗁 Java					
🖌 🗁 SVN					
Project from SVN					
> 🗁 Texlipse					
> 🗁 XL UPC					
> 🗁 Other					
L					
?	< Back Next > Einish	Cancel			
\sim					

(c) Select the PIPS URL.

0	Checkout from SVN	\odot \odot \otimes				
Checkout Project from SVN repository						
Select an existing repository location or create a new location.						
	to checkout a project from the SVN re	pository.				
<u>Create a new repos</u>	itory location					
• <u>U</u> se existing reposi	ory location:					
Label	∧ ∶URL					
https://scm.cri.ensmp	.fr/svn/pips/ https://scm.cri.ensmp.	fr/svn/pips/				
<						
(?)	< <u>B</u> ack <u>N</u> ext > <u>F</u>	inish Cancel				

(d) Select the trunk or your branch copy of the trunk. You can use "Browser..." to help you select it. You can also choose an older revision if you want.

WARNING: if you don't select the trunk or your specific branch, it will make a new project with all the depositories of PIPS (including trunk, all the branches and the tags).

0	Checkout from SVN	\odot \odot \otimes			
Select Resource Select a resource which v	will be checked out as project.	SVN			
UR: https://scm.cri.ensi	mp.fr/svn/pips/trunk	Browse			
• Head Revision	Revision				
Date: 1/ 7/2014	2:09:13 PM				
Revision:	Browse				
			• •	Select Resource	$\odot \odot \otimes$
			Select base resou Select a resource create patch oper	that will be used as a base variant in	SVN
			 ✓ m pips 22030 ↓ ↑ trunk 22030 		
			> 📥 branches 22		
			> 🌾 tags 20850		
			> 🗁 bundles 208	51	
			ROOT 22030		
(?)	< Back Next > Finish	Cancel	?	ОК	Cancel

- (e) Click "Finish".
- (f) A new window appears. Choose *Check out as a project with the name specified.*

		0.0.0
0	Check Out As	$\odot \odot \otimes$
Check Out As		SVN
You can checkou checkout	it the selected repository resource by different ways. Select the method of	
Choose how to c	heck out folder 'pips'	
O Check out as	s a project configured using the New Project <u>W</u> izard	
 Find projects 	; in the children of the selected resource	
O Check out a	s a folder into <u>e</u> xisting project	
 Check out as 	s a project with the name specified:	
pips		
Depth: Recursiv	ely	~
	Revision	
 Head Revis 	ion	
O Date:	1/ 7/2014 💭 2:09:58 PM 💭	
Revision:	Browse	
?	< Back Next > Finish	Cancel

- (g) Click "Finish".
- 3. Convert this new project into a C project
 - (a) Right click on this new SVN project and choose "New/Convert to C/C++ Project (Add C/C++ Nature)"

New	>	Project	
Go <u>I</u> nto		📑 File	
📄 Сору	Ctrl+C	🗳 Folder	
📋 Paste	Ctrl+∨	C Project	
💢 <u>D</u> elete	Delete	C++ Project	
Mo <u>v</u> e		Convert to a C/C++ Project (Adds C/C++ Nature)	
Rena <u>m</u> e	F2	T <u>O</u> ther	Ctrl+N
🚵 Import			
🛃 Exp <u>o</u> rt			
🐑 Refresh	F5		
Clo <u>s</u> e Project			
Close <u>U</u> nrelated Projects			
Show in Remote Systems view			
@ Build Documentation			
<u>B</u> un As	>		
<u>D</u> ebug As	>		
Profile As	>		
Team	>		
Comp <u>a</u> re With	>		
Rep <u>l</u> ace With	>		
Restore from Local History			
Checkstyle	>		
Properties	Alt+Enter		
Resource Configurations	>		

(b) Choose to convert into a C project.

Note: you can convert many projects in one step, if your SVN project is added as Candidate for conversion (empty in my screen shot)

		0.0.0		
🔵 🕙 Convert to a	C/C++ Project	$\odot \odot \otimes$		
Convert to a C/C++ project				
The wizard adds C/C++ Nature to the selected pro them. It also converts old-style C/C++ projects to				
Candidates for conversion:				
Convert t	o C or C++			
• C Project	○ C++ Project			
Projec	t options			
 Specify project type 				
Project type:	Toolchains:			
- 🗁 GNU Autotools	Other Toolchain			
- 🗁 Executable	Cross GCC			
– 🗁 Executable (XL UPC)	GNU Autotools Toolchain			
– 🗁 Shared Library	Linux Berkeley UPC			
– 🗁 Shared Library (XL UPC)	Linux GCC			
– 😕 Static Library	XL C/C++ Tool Chain			
– 😕 Static Library (XL UPC)	XL UPC Tool Chain			
🕞 Makefile project				
Show project types and toolchains only if they	are supported on the platform			
?	<u> </u>	Cancel		

(c) Click "Finish".

4. Do the same for linear and newgen and your personal version of PIPS (in your branch). (Sec 4 step 2.)

5 Configure your projects

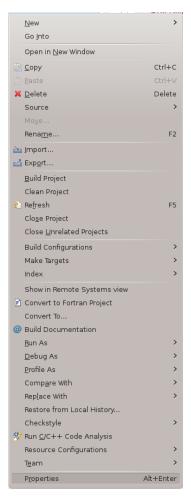
1. Add the *include* path for the different dependencies.

For this purpose you can see the developer-guide part Section 3.2.2 Missing includes. In summary, these configurations are done in the property project, menu "C/C++ General", submenu "Paths and Symbols", "Include" tab.

Without these includes, you won't be able to use the power of Eclipse.

List of includes for the different projects:

- (a) **newgen** needs: /newgen/include
- (b) **linear** needs: /linear/inlcude AND include from polylib
- (c) **PIPS** needs: /newgen/include, /linear/inlcude and /pips/include
- 2. Modify the *Path* Environmement to add pipsrc. It corresponds to the path added with pipsrc.sh.
 - (a) Open the properties of your pips project *Alt+Enter* or right-click on your pips project or *File/Properties*.

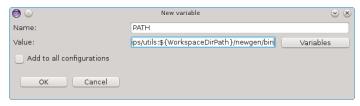


(b) Select C/C++ Build/Environment. Click Add.

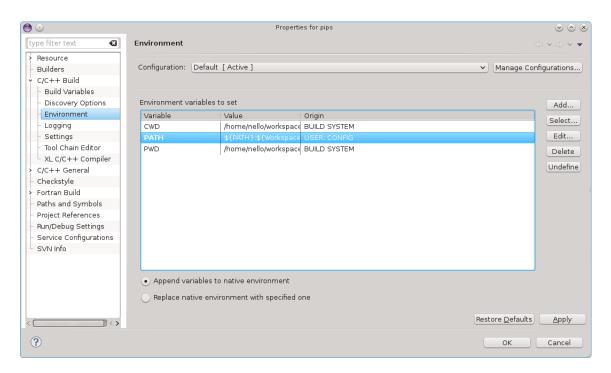
) 🖸		Properti	es for pips		\odot \odot
	•2	Environment				$(\neg \lor \neg \lor \lor \lor$
	Resource					
	- Builders	Configuration:	Default [Active]		✓ M	anage Configurations
•	 C/C++ Build 					
	– Build Variables					
	- Discovery Options	Environment v	ariables to set			Add
	Environment	Variable	: Value	Origin		Select
	 Logging 	CWD	/home/nello/workspace	BUILD SYSTEM		Select
	 Settings 	PWD	/home/nello/workspace	BUILD SYSTEM		Edit
	- Tool Chain Editor					Delete
	L C/C++ Compiler					
:	C/C++ General					Undefine
	- Checkstyle					
	 Fortran Build 					
	- Paths and Symbols					
	- Project References					
	- Run/Debug Settings					
	- Service Configurations					
	SVN Info					
		 Append va 	riables to native environment			
		Replace na	tive environment with specified (ne		
		0.11				
					Restore	Defaults Apply
<						
	?					OK Cancel

- (c) Add the variable PATH with the value *\${PATH}:*
 - \${WorkspaceDirPath}/pips/bin: \${WorkspaceDirPath}/pips/utils: \${WorkspaceDirPath}/newgen/bin

Replace pips by the name of your pips project if it needs.



(d) Apply your modification and click OK.

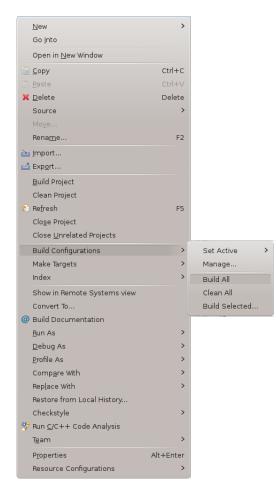


After this configuration has been done, you can use PIPS in Eclipse.

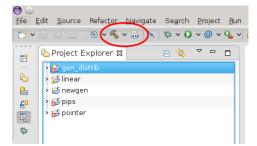
6 How can we compile PIPS with Eclipse?

Compilation corresponds to build your project in Eclipse.

1. Right click on the project you want to compile and choose build:



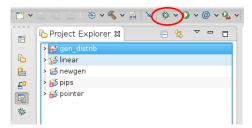
2. You can also select your project and use the shortcut in the C/C++ perspective:



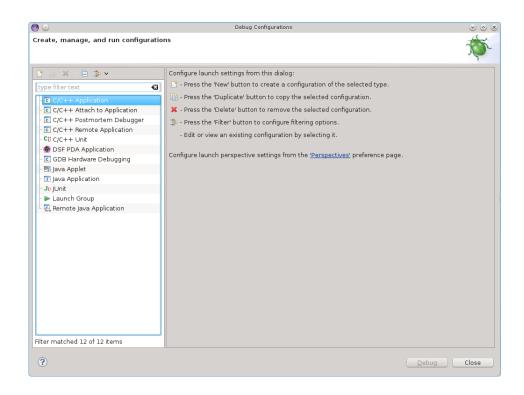
7 How can we debug PIPS with Eclipse?

For this purpose, you will need to set a debug configuration.

1. Right click on your PIPS project, and choose "Debug As/Debug Configuration". In C/C++ perspective you can use the arrow near the shortcut.



2. Select C/C+++ Application and click on "New".



3. Click on "Search Project" to define what you want to launch.

• •	Debug Conf	figurations		$\odot \odot \otimes$
Create, manage, and run configurati Register Reg	ons			Ś
🖹 🗎 🗶 🗆 🋸 🗸	Name: pips Default			
type filter text	🎦 Main 🖉 Arguments 🖉 En	nvironment) 🏇 Debugger) 🦆 S	ource 🗖 🗖 <u>C</u> ommon	
C/C++ Application	C/C++ Application:	C/C++ Application:		
c pips Default				
- C/C++ Attach to Application			richten (Gesenh Berliet)	
- 🖸 C/C++ Postmortem Debugger		Va	riables Searc <u>h</u> Project	Browse
- C/C++ Remote Application	Project:			
- Cü C/C++ Unit	pips			Browse
- 🚭 DSF PDA Application		Build (if required) before la	unchina	
- 🖸 GDB Hardware Debugging				
- 🖭 Java Applet	Build configuration:	Use Active		~
- 🗊 Java Application		Select configuration	using 'C/C++ Application'	
– Ju JUnit		Select configuration	using even in Application	
- 🕨 Launch Group	🔵 Enable auto build	🔵 Disa	ble auto build	
모 Remote Java Application	 Use workspace settings 	Configur	e Workspace Settings	
Filter matched 13 of 13 items	Using GDB (DSF) Create Proce	ss Launcher - <u>Select</u>	Apply	Revert Close

4. Select tpips and the binary corresponding.

e 🖸	Program Selection	$\odot \odot \otimes$
Choose a progran	n to run:	
Binaries:		
🚺 fpips		
🜔 hpfc_interactiv	/e	
🚺 pips		
💽 tpips		
		E
Qualifier:		
🏇 ×86_64le - /pi	ips/bin/LINUX_x86_64_LL/tpips	
🏇 x86_64le - /pi	ips/src/Passes/tpips/LINUX_x86_6	4_LL/tpips
?	ОК	Cancel

5. Configure the arguments to send your tpips file to execute. Without them, you have to write yourself the different instructions.

● ⊙	Debug Configurations	$\odot \odot \otimes$
Create, manage, and run configuration	ons	Ś
🖹 🗎 🗶 I 🖻 🌞 🗸	Name: pips Default	
type filter text	📄 Main 🕪 Arguments 🛛 📠 Environment) 🏇 Debugger 💱 Source 🔲 Common	
C/C++ Application	Program arguments:	
 ↓ [c] pips Default ■ C/C++ Attach to Application ■ C/C++ Postmortem Debugger ■ C/C++ Remote Application ■ Cit C/C++ Unit ■ D5 PDA Application ■ GDB Hardware Debugging ■ Java Applet ■ Java Application Jor JUnit ■ Launch Group ■ Remote Java Application 	/home/nello/CRI2/30MYPIPS/validation/Semantics-New/abs.tpips	Ĵ
		~
		∨ariables
	Working directory:	
	\${workspace_loc:pips}	
	✓ Use de <u>f</u> ault	
		/ariabl <u>e</u> s
Filter matched 13 of 13 items	Using GDB (DSF) Create Process Launcher - <u>Select</u> Apply	Re⊻ert
?	Debug	Close

- 6. Click on "Apply", Click on "Debug"
- 7. Enjoy your debug with Eclipse.

8 Why can I only launch the debug for tpips only once?

You have to configure the PATH environmement variable, see step 2 in part 5 Configure your project.

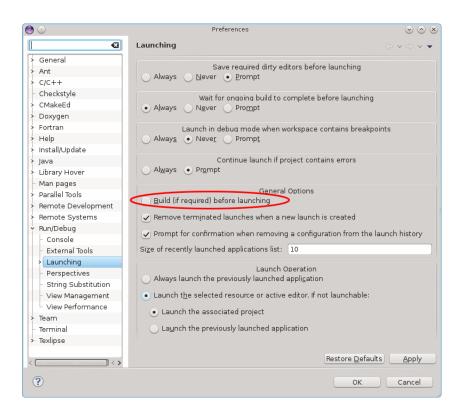
9 Why debug step go randomly top and down?

You have to compile without optimization options. The same thing will happens with gdb if the compilation is not configure.

In the file makes/gnu-stuff.mk, put -O0 instead of -O2.

10 How can I ask Eclipse to not recompil before launch debug?

In Windows/Preferences, select Run/Debug menu, Launching submenu. Uncheck Build (if required) before launching.



11 How can I use SVN in Eclipse?

You can open the *Team Synchronizing* perspective for this purpose. You can also right click on a folder or file and look the different possibilities of *Team*.



WARNING: Take care to not commit the Eclipse settings or some files coming from compilation especially in the trunk.