



SX-Aurora TSUBASA

輸出する際の注意事項

本製品(ソフトウェアを含む)は、外国為替および外国 貿易法で規定される規制貨物(または役務)に該当するこ とがあります。

その場合、日本国外へ輸出する場合には日本国政府の輸出許可が必要です。

なお、輸出許可申請手続きにあたり資料等が必要な場合 には、お買い上げの販売店またはお近くの当社営業拠点に ご相談ください。

はしがき

本書は NEC Parallel Debugger の利用方法ついて説明したものです。

本書の読み進め方

本書は、次の章から構成されています。表の右側の対象読者を参考にして読み進めてください。

章	タイトル	内容	対象読者
1	概要	NEC Parallel Debuggerの特徴とデバッ グ利用環境の構成等の説明	一般利用者
2	プロジェクトの作成	Eclipse PTPにおける同期プロジェクトの 作成方法の説明	一般利用者
3	プロジェクトのビルド	Eclipse PTPにおけるプロジェクトのビル ド方法の説明	一般利用者
4	Debug Configurationの 作成・設定	Eclipse PTPにおけるデバッグ実行の開始 方法の説明	一般利用者
5	デバッグ操作	Eclipse PTPにおけるデバッグ操作の説明	一般利用者
6	注意・制限事項	利用に関する注意・制限事項の説明	一般利用者

関連説明書

- C/C++コンパイラユーザーズガイド (G2AF01)
- Fortranコンパイラユーザーズガイド (G2AF02)
- NEC MPI ユーザーズガイド (G2AM01)

備考

- (1) 本書は、NEC Parallel Debugger Release 1.0.0 以降に対応しています。
- (2) Eclipseは、Eclipse Foundationの登録商標です。
- (3) その他、記載されている会社名、製品名は、各社の登録商標または商標です。

用語定義・略語

用語・略語	説明
ベクトルエンジン (VE、Vector Engine)	SX-Aurora TSUBASAの中核であり、ベクトル演算を行う部分で す。PCI Expressカードであり、x86サーバーに搭載して使用しま す。
ベクトルホスト、ホスト (VH、Vector Host)	ベクトルエンジンを保持するサーバー、つまり、ホストコンピュー タを指します。
ノード	共有メモリを有する単位を指します。ベクトルエンジン1枚につき 1ノードを構成します。
NQSV	NEC Network Queuing System V (NQSV) は、高性能なクラス タシステムの計算リソースを有効に活用するためのジョブ管理シ ステムです。
NEC MPI	NECが製造するMPI実装です。MPIは、Message Passing Interfaceの略語です。主にノード間で並列コンピューティングを 行うための標準化規格です。同一ノード内であっても、プロセスご とにメモリを共有しない場合、プロセス間の通信にMPIを使用する ことが可能です。OpenMPとの併用も可能です。
Eclipse PTP	Eclipse Parallel Tools Platform は、OSSの並列アプリケーション 向け統合開発環境です。
ビュー	Eclipseのウィンドウ内に表示されるサブウィンドウの1つ
パースペクティブ	Eclipseにおいて、メニューバーやツールバー、ビュー等を特定の 目的に応じてひとまとめにしたもの
ターゲットプロセス	Parallel Debug ビューまたはDebug ビューで選択したデバッグ 操作対象のプロセス

第1章	概要1
1.1	NEC Parallel Debugger とは1
1.2	利用環境の構成1
1.3	デバッグ実行の手順2
第2章	プロジェクトの作成3
2.1	Eclipse の起動3
2.2	リモートホスト上の make 環境をインポートするケース4
2.3	ローカルホスト上の make 環境をインポートするケース6
第3章	プロジェクトのビルド9
3.1	Build Configurations の設定9
3.2	Build Project の実行11
第4章	Debug Configuration の作成・設定12
4.1	Debug Configuration の作成12
4.2	Resources の設定14
4.3	Application の設定16
4.4	Debugger の設定17
4.5	Environmentの設定17
4.6	その他の設定18
4.7	デバッグ実行の開始18
第5章	デバッグ操作20
5.1	利用可能ビュー一覧20
5.2	複数プロセスの一括実行21
5.3	特定プロセスの実行22
5.4	スタックトレースの参照23
5.5	変数の参照
5.6	ブレイクポイントの設定
5.7	デバッグの終了25
第6章	注意・制限事項27
6.1	注意事項
6.2	その他27
付録 A	改版履歴

目 次

第1章 概要

1.1 NEC Parallel Debuggerとは

NEC Parallel Debugger は、Eclipse PTP (Parallel Tools Platform)に対する MPI アプリケーショ ンのデバッグ機能向けプラグインです。Eclipse PTP は、オープンソースソフトウェアの並列プログ ラム向け統合開発環境です。本プラグインを適用した Eclipse PTP を使用すると、OpenMP 又は 自 動並列化による共有メモリ型並列処理を行う Fortran 及び C/C++プログラムに加えて, MPI を使用 した分散メモリ型並列プログラムを効率よくデバッグすることができます。

1.2 利用環境の構成

利用者は、ssh 等でログインしたフロントエンドマシン上に NEC Parallel Debugger (図の EclipsePTP with NPDB) を起動し、ソースコードの編集・コンパイル、ssh で起動した SX-Aurora TSUBASA システム上で動作するアプリケーションのデバッグを行います。また、ssh や NQSV を利 用した会話リクエスト (qlogin) 等でログインした SX-Aurora TSUBASA システム上に NEC Parallel Debugger を起動して利用することもできます。NEC Parallel Debugger から MPI アプリケーショ ンのデバッグ実行を開始すると、SX-Aurora TSUBASA システム上に SDM (Scalable Debug Manager) が起動され、その配下に gdb およびデバッグ対象である MPI アプリケーションが起動さ れます。



System



1.3 デバッグ実行の手順

アプリケーションのデバッグ実行は、以下の手順で行います。ステップ 1-3 で行う設定は、当該ア プリケーションの初回のデバッグ実行でのみ必要な設定であり、2回目以降のデバッグ実行では、基 本的に設定は不要です。以降の章で、VE 上で動作する NEC MPI を使用したアプリケーションのデバ ッグ実行の手順について、各設定・操作の詳細を説明します。



第2章 プロジェクトの作成

プロジェクトとは、Eclipse においてソースコードの編集やアプリケーションの作成、デバッグ実行 を行う環境の単位です。Eclipse 上でアプリケーションを開発する場合、プロジェクトを作成する必 要があります。以下に、プロジェクトの作成および make 環境のインポートの手順を2つのケースに 分けて説明します。

- リモートホスト上のmake環境をインポートするケース
- ローカルホスト上のmake環境をインポートするケース

2.1 Eclipse の起動

以下のコマンドを実行し、Eclipse を起動します。(Eclipse のインストールディレクトリが/INST-DIR の場合)

% /INST-DIR/eclipse/eclipse

Eclipse Launcher のウィンドウが開くので、Workspace ディレクトリ(デバッグ対象であるアプ リケーションの make 環境を格納するディレクトリ)を入力します。Eclipse のウィンドウが開き、 Welcome ビューが表示されます。



表示された Welcome ビューを閉じます (タブのXをクリック)。Welcome ビューを閉じると、 C/C++パースペクティブが表示されます。



2.2 リモートホスト上のmake環境をインポートするケース

リモートホスト上に存在するアプリケーションの make 環境をインポートする場合、プロジェクトの種類としては、同期プロジェクトを推奨します。同期プロジェクトによって、Eclipse から操作した ソースコードの修正といったプロジェクト内の変更内容がインポート元に自動で反映されます。以下 に、同期プロジェクトの作成および make 環境のインポートの手順を説明します。

 Project Explorerビューの空白スペース上で右クリックし、New > Synchronized C/C++ ProjectまたはSynchronized Fortran Projectを選択します。



(2) New Synchronized Projectウィンドウが開くので、Project Nameを入力し、Connection name のNewボタンを押下します。

New Synchronized Project Please select a project type to continue	I.
Project name: ve_mpi_bm Local directory- ✓ Use default location Local directory: /usr/uhome/npdb/workspace/ve_mpi_bm Remote directory- Connection name: Please select a connection Remote directory:	Browse New Browse
Modify file filtering Project Type GNU Autotools GEXecutable Secutable (XL UPC) Shared Library Shared Library Static Library Static Library Makefile project Makefile project Show project types and toolchains only if they are supported on the platform	Remote Toolchain (select 1 or more) Local Toolchain (optional - select 0 or more)
0	< Back Next > Cancel Finish

(3) New Connectionウィンドウが開くので、Connection name, Host, User, Passphrase等を 入力し、Finishボタンを押下します。

New Connection Specify properties of a new connection						
Connection na	me: remotehost					
Host:	remotehost					
User:	npdb					
Public key based authentication Keys are set at <u>Network Connections, SSH2</u>						
Passphrase: ••••••						
O Password based authentication						
Password:						
Advanced						
?	Cancel Finish					

(4) New Synchronized Projectウィンドウ上でRemote directoryを入力し、Project Typeにおいて、Makefile project > Empty Projectを選択し、Finishボタンを押下します。

Project name: ve_mpi_bm	
ocal directory ✓ Use default location .ocal directory: /usr/uhome/npdb/workspace/ve_mpi_bm Remote directory Connection name: remotehost Remote directory: /usr/uhome/npdb/remotehost/ve_mpi_bm	Browse New Browse
Project Type	Remote Toolchain (select 1 or more) Other Toolchain Cygwin GCC GCC Fortran Local Toolchain (optional - select 0 or more) Other Toolchain Cygwin GCC GCC Fortran

以上で同期プロジェクトの作成と make 環境のインポートは完了です。

なお、インポート元の変更はインポート先である同期プロジェクトへ自動で反映されません。イン ポート元の変更を取り込む場合は、Project Explorer で同期プロジェクトを選択し、次いでメニュー から → ボタンを押下します。

File Edit Source Refactor Navigate Search Project Run Window Help	
┆ ◻º ▾ ;;; (\$) ≫ ▾ �� ▾ ;;; (* ▼ ; (* ▼ 6º ▼ ; * ▼ 0 ▼ 9; ▼ ; * * 0 ▼ 9; ▼ ; * * 0 ▼ 9; ▼ ; * * 0 ₽ 0; (* ; * * * 0 ▼ 9; ▼ ; * * * 0 ₽ 0; (* ; * * * * * * * * * * * * * * * * *) i 🖏 🗸
🕴 🖗 🔻 🏠 🗢 Synchronize the selected project	
Project Explorer X	
▶ 😂 ve_mpi_bm	

2.3 ローカルホスト上のmake環境をインポートするケース

ローカルホスト上に存在するアプリケーションの make 環境をインポートする場合、以下の手順で プロジェクトを作成します。以下の作業によって、インポート元と同じディレクトリにプロジェクト が作成されます。

(1) Project Explorer ビューの空白スペース上で右クリックし、Importを選択します。



(2) Importウィンドウが開くので、C++ > Existing Code as Makefile Projectを選択します。

Select Creates a new Makefile project in a directory containing existing code	Ľ
Select an import wizard:	
type filter text	Ð
🕨 🧀 General	
▼	
C/C++ Executable	
Existing code as Autotools project	
📴 Existing Code as Makefile Project	
🕨 🧀 Git	
🕨 🧀 Install	
🕨 🗁 Oomph	
RPM	
🕨 🗁 Run/Debug	
Erarget System Configurations	
(?) < Back Next > Cancel	Finish

(3) ウィンドウが開くので、Project Name, Existing Code Location(インポート元)、Toolchain としてNoneを選択し、Finishボタンを押下します。

Project Name		
ve_mpi_bm		
Existing Code L	cation	
/usr/uhome/np	db/workspace/localhost/ve_mpi_bm Browse	•
anguages		
🗹 C 🗹 C++		
Toolchain for Inc	exer Settings	_
<none></none>		
GCC Fortran		
GNU Autotools	Toolchain	
IBM XL Fortran	Tool Chain	
Intel(R) Fortrar	Toolchain on Intel(R) 64	
Linux Berkeley	UPC	
Linux GCC		
XL C/C++ Tool	Chain	
XL UPC Tool Ch	ain	
Show only a	vailable toolchains that support this platform	
	a Bash Nauka Canaal Cinish	

(4) Fortranプログラムの場合、C/C++プロジェクトからFortranプロジェクトへ変換します。Project ExplorerからConvert to Fortran Project を選択します。

Project Explorer 🕱			
E 🕏 ▶ 🗳 ve_mpi_bm	<u>N</u> ew Go <u>I</u> nto	•	
	Open in <u>N</u> ew Window Show in Local Terminal	Þ	
	Copy Copy Paste Delete Remove from Context Source Move Rename	Ctrl+C Ctrl+V	
	<u>≧ I</u> mport ☑ Exp <u>o</u> rt		
	Build Project Clean Project and Refresh Close Project Close Unrelated Projects	F5	
	Build Targets Index Build Configurations	> >	Properties ۂ Call C
	Profiling Tools <u>V</u> alidate B. Convert to Fortran Project	•	
🤟 ve_mpi_bm	Run As	•	

以上で、プロジェクトの作成と make 環境のインポートは完了です。

第3章 プロジェクトのビルド

プロジェクトをビルドすることで、アプリケーションを作成します。手順を以下に説明します。

3.1 Build Configurations の設定

ソースコードの編集やアプリケーションの作成を行うために、コンパイラやヘッダファイルのパス を設定します。手順を以下に説明します。

(1) Project Explorerビューでプロジェクトを右クリックし、Propertiesを選択します。

File Edit Source Refactor Navigate Se	earch Project	t Run Win	ndow Help						
New Sh	ift+Alt+N ♪	🗴 🝷 🕐	🔂 🔻 🛍	° 🕶 🖻 🕶	6 🕶	* • () - (-	1
Open File								ſ	QI
Open Projects from File System							_	-	10
Close	Ctrl+W						-		Н
Close All	Shift+Ctrl+W								
🔚 Save	Ctrl+S								1
📓 Save As									
🐚 Save All	Shift+Ctrl+S								
Revert									
Move									
🖻 Rename	F2								
8 Refresh	F5								
Convert Line Delimiters To	•								
👜 Print	Ctrl+P								
📐 Import									
🖄 Export									
Properties	Alt+Enter								
<u>1</u> vempi.c [ve_mpi_bm]									
Switch Workspace									
Bostart	,								儿
Fxit		asks 📮 Cor	nsole 🔲 Pr	operties #	ll Call Gra	ph			
Ent						_	_	_	_

(2) C/C++ Build > Environment > Add で PATH環境変数を設定します。Fortranプロジェクトの場合も同様に C/C++ Buildで設定します。

B	Environment			¢	• = = •
 Resource Builders C/C++ Build 	Configuration: De	fault [Active]	•	Manage Cor	figurations
Build Variables	Environment variat	oles to set			Add
Logging Settings Tool Chain Editor XL C/C++ Compiler C/C++ General Fortran Build Linux Tools Path Paths and Symbols Project References Run/Debug Settings	Variable CWD PWD	Value /usr/uhome/npdb/work /usr/uhome/npdb/work	Origin BUILD SYSTEM BUILD SYSTEM		Select Edit Delete Undefine
 Task Repository Task Tags Validation WikiText 	 Append variable Replace native e 	es to native environment environment with specified	one Restore	e Defaults	Apply

Makeする際に利用するコマンドのパスが通るように設定して下さい。

Name:	РАТН	
Value:	/opt/nec/ve/bin:/usr/bin:/bin	Variables
🗹 Add	to all configurations	
	Cancel	ОК

 (3) C/C++ General > Paths and Symbols > Add でインクルードファイルのディレクトリを 設定します。Fortranプロジェクトの場合も同様に C/C++ Generalで設定します。

B	Paths and Symbols	(⇒ ▼ ⇒) ▼ ▼
 Resource Builders C/C++ Build 	Configuration: Default [Active]	Configurations
 C/C++ General Code Analysis 	< 🕒 Includes # Symbols 🛋 Libraries 👼 Libra	ary Paths 💙
Documentation File Types	Languages Include directories	Add
Formatter	Fortran	Edit
Language Mappin	GNU C GNU C++ GNU C++	Delete
Paths and Symbo Preprocessor Incl Profiling Categori	UPC	Export
XL C/C++ Langua	UDersenanse Indude Daths Marris	Move Up
Linux Tools Path	Show built-in values	Move Down
Paths and Symbols Project References	🖓 Import Settings	
Run/Debug Settings	Restore Defaults	Apply
?	Cancel	Apply and Close

ソースコード上で利用するインクルードファイルのディレクトリを指定します。以下は設定 の例です。適切なインクルードファイルのディレクトリを指定して下さい。

Directory:		
/opt/nec/ve/ncc/1.0.3/include/		
Add to all configurations		Variables
Add to all languages		Workspace
		File system
	ОК	Cancel

3.2 Build Project の実行

プロジェクトをビルドすることで、アプリケーションを作成します。Project Explorer ビューに表示されるプロジェクト名を右クリックし、Build Project を選択します。

第4章 Debug Configuration の作成・設定

Debug Configuration を作成し、デバッグ実行の設定を行い、アプリケーションのデバッグを開始 します。デバッグの手順を以下に説明します。

4.1 Debug Configuration の作成

Debug Configuration を作成します。手順を以下に説明します。

(1) Window > Perspective > Open Perspective > Other を選択します。



(2) Parallel Debug パースペクティブを選択します。

Open Perspective (on a1sb4_003)
EC/C++ (default)
☆ Debug
Reference and the second secon
🔚 Git
🗱 Parallel Debug
() Planning
Resource
System Monitoring
🛱 SystemTap IDE
E ⁰ Team Synchronizing
础 Tracing
X XML
Use F2 to display the description for a selected perspective.
Cancel Open

(3) Debug(虫マーク) > Debug Configurations を選択します。

File Edit Navigate Search Project Run Window Help		
1 → 1 0 + 1 0 + 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0	O ▼ 9 ▼ : B A ▼ : B → A ▼ : C → →	Quick Access
🎇 Parallel Debug 🏼 👘 👘 🖷 🚴 🥱 😥 (%) () 🔍 🔍	Breakpoints & Expressions (*)* Varia	ibles 👼 Signals 🏥 Arrays 🔀 🛛 🖶 🗖
Please select a job	Debug As	
	De <u>b</u> ug Configurations	
	Organize Fa <u>v</u> orites	
🔯 Debug 🏼 🐘 🕄 🖷 🕅 🗏 🥱 .	🤜 i 💀 💌 🗖	
1/)(<u>_</u>))

(4) Debug Configurations ウィンドウが開くので、Parallel Application > New を選択します。



(5) 本Debug Configuraion につける名前をNameテキストボックスに入力します。

Create, manage, and run configurations

😣 No target system configuration has been selected

	7
	Name: test_ve_mpi_bm
type filter text	😫 Resources 📱 Application 🕬 Arguments 🕸 Debugger 🖾 Environment Synchronize 🤤 Source 🔭
E C/C++ Application E C/C++ Attach to Application E C/C++ Postmortem Debugger E C/C++ Remote Application	Target System Configuration: Please select a target system configuration Connection Type Local
Launch Group Launch Group (Deprecated) Equal: Parallel Application	
Filter matched 8 of 8 items	Revert Apply
?	Close Debug

4.2 Resources の設定

Resources タブにおいて、MPI アプリケーション実行におけるオプション等を設定します。手順を 以下に説明します。

(1) Target System Configuration で、Generic NECMPI Interactiveを選択します。

Create, manage, and run configura No target system configuration has b	tions een selected		Ť
C 🗎 🗶 🗆 🛱 🕶	Name: test_ve_mpi_bm		
type filter text	Resources P Application	n 🕪 Arguments 🕸 Debugger 🗷 Environment Synchronize 🦗 Source 🔲 Common	
 C/C++ Application C/C++ Attach to Application C/C++ Postmortem Debugger C/C++ Remote Application Launch Group Launch Group (Deprecated) Parallel Application New_configuration 	Target System Configuration: Connection Type Local Remote Please	Please select a target system configuration Please select a target system configuration ALCF PBS BGP Batch ALCF PBS BGQ Batch BlueWaters PBS Batch Generic Grid Engine Batch Generic MPICH2 Interactive Generic NECMPI Interactive Generic PBS BGP Batch Generic PBS BGP Batch Generic PBS BGP Batch Generic NECMPI Interactive Generic PBS BGP Batch Generic PBS BGP Batch Generic PBS BGP Batch Generic PBS BGP Batch Generic PBS BACH Generic PBS BACH Generic PBS BACH Generic SURM ALPS Batch Generic SUURM BGP Batch Generic SUURM BGP Batch Generic SUURM BACH Generic SUURM BACH Generic Torque ALPS Batch Generic Torque Batch	•
lter matched 8 of 8 items		Generic Torque Interactive IBM LoadLeveler IBM LoadLeveler (Blue Gene) IBM Platform ISF	
?		IBM Platform LSF Interactive	

- (2) Connection Type で、MPIアプリケーションの実行をLocalに起動するか、Remoteに起動するかを指定します。Remoteに起動する場合、起動先のホスト情報を入力します。
- (3) MPIアプリケーション実行コマンド(mpiexec)に渡す実行時情報をBasic Optionsで設定します。

Create, manage, and run configura (2) [Application]: Application program no	tions to specified
[] 🗎 🗶 🖻 🛟 ▼	Name: test_ve_mpi_bm
type filter text	😫 Resources 📲 Application 🏟 Arguments 🕸 Debugger 📠 Environment Synchronize 🤤 Source 🔲 Common
© C/C++ Application © C/C++ Attach to Application © C/C++ Postmortem Debugger © C/C++ Remote Application ■ Launch Group ▶ Launch Group ▶ Launch Group ■ Launch Group ■ Launch Group ■ Launch Group ■ Launch Group	Target System Configuration: Generic NECMPI Interactive Connection Type Local Remote Please select a connection Basic Options Advanced Options Number of processes on VE nodes: 64 - + VE Nodes VE Nodes VE Nodes VE Nodes VE Nodes VE Not inst: host1,host2 View Configuration Restore Defaults
Filter matched 8 of 8 items	Revert Apply
?	Close Debug

各設定項目を以下に説明します。

ビュー名	説明
Number of	VE ノード上に起動する MPI アプリケーションの総プロセス数。
processes on VE	(-np オプションに渡されます)
nodes	
VE node range	MPI アプリケーションを起動する VE の VE ノード番号。
	1 つの VE ノード番号を単独で指定するか、または 2 つの VE ノード
	番号を"-"でつないで VE ノードの範囲を指定します。指定がない場
	合、VE#0 上に MPI アプリケーションを起動します。
	(-ve オプションに渡されます)
	単独指定の例:1 (VE ノード番号#1 を使用する場合)
	範囲指定の例:0-7 (VE ノード番号#0-#7 を使用する場合)
Host file	MPI アプリケーションを起動するホストが記述されたファイル名を
	指定します。ファイルには、ホスト名を1行づつ記述します。Host
	file および Host list に指定がない場合、ローカルホスト上に MPI ア
	プリケーションを起動します。
	(-hostfile オプションに渡されます)

ビュー名	説明
Host list	MPI アプリケーションを起動するホストを指定します。1 つのホスト
	名を単独で指定するか、または2つ以上のホスト名を","でつないで
	指定します。Host file および Host list に指定がない場合、ローカル
	ホスト上に MPI アプリケーションを起動します。
	(-hosts オプションに渡されます)
	指定例:host1,host2,host3

 (4) 上記以外の実行時情報を指定したい場合、Advanced Optionsで設定します。まずExtra Argumentsのチェックボックスを選択し、次いでテキストボックスにMPIアプリケーション 実行コマンドに渡すオプションを指定します。

] 🗎 🗶 🗆 🗗 🕇 🗸	Name: test_ve_mpi_bm	
type filter text 🛛 🛛	😫 Resources 🛯 Application 🕬 Arguments 🕸 Debugger 🚾 Environment Synchronize	Source »1
C/C++ Application	Target System Configuration: Generic NECMPI Interactive	
C/C++ Attach to Application		
C/C++ Postmortem Debugge	Iocal Remote Please select a connection	▼ Nev
C/C++ Remote Application		
Launch Group	Basic Options Advanced Options	
Parallel Application		
New configuration	Launch Arguments	
	Arguments: -hosts host1,host2 -ve 0-3 -np 64 -ppn 2	
	Rextra Arguments: -ppp 2	
	Installation location	
	Use default path	
	Executables directory (bindir):	Browse
	View Configuration Restore Defaults	
	Revert	Apply
Itor matched 9 of 9 itoms	never	Apply

4.3 Application の設定

Application タブにおいて、プロジェクトおよびアプリケーションを指定します。

Create, manage, and run configu Create a configuration to debug a pa	rallel application	Ť
Image: Constraint of the second s	Name: test_ve_mpi_bm Image: test_ve_mpi_bm <td>»1</td>	»1
C/C++ Postmortem Debugger	ve_mpi_bm Application program:	<u>B</u> rowse:
	/usr/uhome/npdb/workspace/ve_mpi_bm/vempi.out Copy executable from local filesystem Path to local executable: ✓ Display output from all processes in a console view	Browse
Filter matched 8 of 8 items	Revert	Apply
?	Close	Debug

4.4 Debugger の設定

Debugger タブにおいて、デバッガ情報を設定します。

- (1) Debugger backend として、sxaurora-gdb-miを選択します。
- (2) Use built-in SDM if available for the target platform のチェックボックスを外します。

	Name:	test_ve_mpi_bm		
ype filter text 🛛 🛛	₽ Res	urces 🖹 Application Թ Arguments 🕸 Debugger 🖉 Environment Synchronize) 🦆 Source 🔲 <u>C</u> omm	on
© C/C++ Application © C/C++ Attach to Application © C/C++ Postmortem Debugger © C/C++ Remote Application ■ Launch Group ▶ Launch Group (Deprecated) ■ Parallel Application ■ New_configuration	Debug Debug Debug Path t Adv	ger: SDM p at main() on startup ger Options ger backend sxaurora-gdb-mi e built-in SDM if available for the target platform o SDM executable (if built-in is not used) /opt/nec/ve/npdb/bin/sdm anced Options		• Browse
			Revert	Apply

4.5 Environmentの設定

必要に応じて、Environment タブでアプリケーション実行で有効にする環境変数を設定します。

ここで設定された環境変数は、mpiexec コマンドの起動ホスト(Resources タブの Connection Type で設定したホスト)でのみ有効となります。本環境変数を MPI アプリケーション全体で有効に する場合には、NEC MPI でサポートされている NMPI_EXPORT 環境変数を併用して下さい。

] 🗎 🗙 🖻 🆆 🖌	Name: test_ve_mpi_bm		
type filter text 🛛 🛛	Resources (▲ Applicati	ion 🕪 Arguments 🕸 Debugger 📧 Environment 🛛 Syna	chronize 🖏 Source 🔲 <u>C</u> ommo
c C/C++ Application	Environment variables to s	et:	
C/C++ Attach to Application	Variable	Value	New
C/C++ Postmortem Debugger	ENV_VAR1	ABC	
C/C++ Remote Application	ENV_VAR2	XYZ	Select.
🖶 Launch Group	NMPI_EXPORT	ENV_VAR1 ENV_VAR2	Edit
Launch Group (Deprecated)			
Parallel Application			Remov
∃New_configuration			
	• Append environment t	to native environment	
	Replace native environ	ament with specified environment	
ter matched 8 of 8 items			Revert Apply

NEC Parallel Debugger でサポートする環境変数は以下の通りです。

環境変数	内容
NPDB_SDM_PORTRANGE	SDM間のTCPIP接続時に利用するポート番号の範囲を指定します。ポート番号を":"で区切って指定します。既定値は、 50000:50079です。

4.6 その他の設定

必要に応じて、Arguments タブ等の他のタブでの設定を行います。

4.7 デバッグ実行の開始

Apply ボタンを押下して上記設定を保存し、次いで Debug ボタンを押下してデバッグ実行を開始 します。

Create, manage, and run configura Create a configuration to debug a paral	ations IIIel application	Ť.
Image: Second system Type filter text Image: C/C++ Application Image: C/C++ Attach to Application Image: C/C++ Postmortem Debugger Image: C/C++ Remote Application Image: Launch Group Launch Group (Deprecated) Image: Parallel Application Image: New_configuration	Name: test_ve_mpi_bm Image: test_ve_mpi_bm Image: Application (M= Arguments (M= Debugger) Environment Synchronize (M= Common) Debugger: SDM Image: Stop at main() on startup Debugger Options Debugger backend Image: Use built-in SDM if available for the target platform Path to SDM executable (if built-in is not used) /opt/nec/ve/npdb/bin/sdm Image: Advanced Options	• Browse
Filter matched 8 of 8 items	Revert	pply
?	Close	Debug

MPI アプリケーションを起動するホスト上で適切にアプリケーションを実行できるように、ログインシェル等で NEC MPI のセットアップスクリプトを読み込むように設定されていることを事前にご確認下さい。

第5章 デバッグ操作

Eclipse PTP における主要なデバッグ操作について説明します。

なお、詳細なデバッグ操作については、Eclipse 公式サイトの Help に掲載されている「Parallel Development User Guide」をご参照下さい。

5.1 利用可能ビュー一覧

Parallel Debug パースペクティブにおいて、デフォルトで表示されるビューの内、VE 向けアプリ ケーションに対して利用できるビューは以下の通りです。各ビューのタブ部分にビュー名が表示され ます。Editor については、タブ部分に開いているファイルの名前が表示されます。

ビュー名	説明
Parallel Debug	複数プロセスに対する一括実行を操作します。
Debug	ターゲットプロセスに対する実行操作およびスタックトレース表 示を行います。
Editor	ソースコードを表示します。
Breakpoints	ブレイクポイントの一覧を表示します。
Variables	ターゲットプロセスの選択したスタックにおける引数・局所変数を 表示します。
Arrays	配列情報を表示します。
Expressions	コード入力によって、データを検査できます。
Signals	ターゲットプロセスのデバッガにおけるシグナル操作を一覧表示 します。
Outline	Editor ビューで表示しているファイルの構造を表示します。
Console	プログラムの標準出力および標準エラー出力を表示します。
Problems	プロジェクトのビルド時のエラーを表示します。
Error Log	プラグインにおいて発生した警告およびエラーを表示します。

5.2 複数プロセスの一括実行

複数プロセスに対して一括して実行操作を行う場合、Parallel debug ビューで実行を操作します。 手順としては、Parallel Debug ビューで実行対象とするプロセスセットを選択し、次いで実行コマン ドを選択します。既定値のプロセスセットは、実行中の全プロセス(ルートセット)です。したがっ て、プロセスセットを選択せずに実行コマンドのみ選択した場合、ルートセットに対して一括して実 行を操作します。

🗱 Parallel Debug 🛿	1	ij,	Ē	3	R	<u></u>	0 9	Q 0	•) <mark>×</mark> {-}	(宮)	•	
6964 - Root [4]													
🔬 6964	0 🔶 🔶 🔶												

各実行コマンドについて、以下に説明します。

実行コマンド	名称	説明
(i)	Resume Group	選択したプロセスセットの実行を再開します。
400	Suspend Group	選択したプロセスセットの実行を停止します。
<u>_</u> \$	Step Into Group	選択したプロセスセットの実行を再開し、ソー スコードの現在行を実行し、次行の先頭で停止 します。 現在行が手続き呼び出しの場合、当該手続きの 先頭で停止します。
	Step Over Group	選択したプロセスセットの実行を再開し、ソー スコードの現在行を実行し、次行の先頭で停止 します。 現在行が手続き呼び出しの場合、当該手続きを 実行し、当該手続き呼び出しの次行の先頭で停 止します。
@	Step Return Group	選択したプロセスセットの現在停止している手 続きを実行し、当該手続き呼び出しの次行の先 頭で停止します。

実行コマンド	名称	説明
	Terminate	選択したプロセスセットのデバッグセッション
	Group	を終了します。

5.3 特定プロセスの実行

特定プロセスに対して実行操作を行う場合、Debug ビューで実行を操作します。手順としては、まずターゲットプロセス・スタックを選択し、次いで実行コマンドを選択します。手順の詳細を以下に 説明します。

- (1) ターゲットにしたいプロセスをParallel Debugビューで選択(ダブルクリック)します。選択 したプロセスのスタックトレースがDebugビューに追加されます。
 - (※) 菱形マークが1プロセスを表します。カーソルを合わせるとランク番号が表示されます。
 - (※) Debugビューに追加されたプロセスには、黒枠マークがつきます。

🗱 Parallel Debug 🛿	🎼 🗓 💼 速 🐵 🖄 😝 O 🔍 🔍 Oʻ 🕶 Oʻ (-) 😂 🖛 🖻)
6964 - Root [4]		
🔬 6964	8 🔶 🌔 🔶	
		- 1

(2) スタックをDebugビューで選択します。



(3) 実行コマンドを選択します。

各実行コマンドについて、以下に説明します。

実行コマンド	名称	説明
	Resume	ターゲットプロセスの実行を再開します。

実行コマンド	名称	説明
00	Suspend	ターゲットプロセスの実行を停止します。
P.	Step Into	ターゲットプロセスの実行を再開し、ソースコー ドの現在行を実行し、次行の先頭で停止します。 現在行が手続き呼び出しの場合、当該手続きの先 頭で停止します。
¢	Step Over	ターゲットプロセスの実行を再開し、ソースコー ドの現在行を実行し、次行の先頭で停止します。 現在行が手続き呼び出しの場合、当該手続きを実 行し、当該手続き呼び出しの次行の先頭で停止し ます。
_62	Step Return	ターゲットプロセスの現在停止している手続き を実行し、当該手続き呼び出しの次行の先頭で停 止します。
	Terminate	ターゲットプロセスのデバッグセッションを終 了します。
**	Remove All Terminated Launches	終了したデバッグセッションの表示を Debug ビ ューから削除します。

5.4 スタックトレースの参照

ターゲットプロセスのスタックトレースは、Debug ビューで参照します。

🎄 Debug 😂		×				М	12	₽_	<u>.</u>	ŕ	3	i⇒	ক্ষ	$\overline{\nabla}$		
▼ @ Process Ø (Suspended)																
T 💁 🔻	✓ m Thread [1] (Suspended: Breakpoint hit.)															
	■ 2 exchange_data() bcast.c:26 4009e9															
	■ 1 main() bcast.c:39 40082f															
🐨 🥐 Pro	cess 1															
ד 🌬	[hread [1] (Running)															
📕 Run	time process 8525															

表示内容について、以下に説明します。

```
セッション名

Process MPIランク番号

Thread [スレッド番号] (実行状態(停止している場合は停止理由))

第N スタック 関数名 ファイル名:行番号 アドレス

第N-1スタック 関数名 ファイル名:行番号 アドレス

:
```

5.5 変数の参照

選択しているスタックの引数および局所変数は、Variables ビューで参照します。まず Parallel Debug ビューまたは Debug ビューでターゲットプロセスを選択し、次に Debug ビューでスタック を選択します。選択したスタックの引数および局所変数が Variables ビューに表示されます。

● Breakpoints 🕰 Expressions	(×)=Variables ⊠	चि∳ Signals	🏝 🎫 🖻 i
Name	Value		
 ◆ argc ◆ argv ◆ buf ◆ myrank ◆ nprocs 	0 9d36 [0 - 1 4	533ed · 1024]	

また、外部変数を含む各変数は、Editor ビューに表示されるソースコード上からも参照できます。 参照したい変数上にポインタを移動することで、変数情報がホバー表示されます。

33	MPI_Init(0,0);	
34	<pre>MPI_Comm_rank(MPI_COMM_WORLD,</pre>	&myrank);
35	<pre>MPI_Comm_size(MPI_COMM_WORLD,</pre>	&nprocs);
36		1: norocs - 4
37	calculate_data(buf, myrank);	1. hprocs - 4

5.6 ブレイクポイントの設定

ブレイクポイントを設定する場合、Editor ビューで操作を行います。Editor ビューで表示されるソ ースコードの行番号の左側をダブルクリックすることで、ブレイクポイントの設定・削除を行います。 既定値では、全プロセスに対して有効なブレイクポイントが設定されます。



設定したブレイクポイントは、Breakpoints ビューで一覧表示されます。



5.7 デバッグの終了

アプリケーションのデバッグを終了する場合、Debug ビューにおいて、 🦲 ボタンを押下してプロ セスを終了させ、その後 🦋 ボタンを押下してデバッグセッションを削除します。

File Edit Navigate	Search Project Run Window Help
-	〕 ▾ ┆ 🖳 ┆ 🕪 💷 🛤 🌫 🕫
🗱 Parallel Debug 🛿	🍺 🗓 🖷 🦻 🚸 69 () 🔍 🔍 () 🔻 () (() 🗮 🖛 🗖 🗖
26550 - Root [4]	
26550	0 \$ \$ \$
💠 Debug 🛙	🎇 🕨 II 🔳 🕅 🔍 🕾 🔍 😾 🗡 🌱 🗖
▼∃ <terminated>test</terminated>	ve_mpi_br_Bemove All Terminated Launches
📲 <terminated, ex<="" td=""><td>it value: 0-Runtime process 20550</td></terminated,>	it value: 0-Runtime process 20550

次に、Parallel Debug ビュー において ¹ ボタンを押下して、残りのプロセスを終了させます。 その後、Parallel Debug ビューでデバッグ中のジョブを右クリックし、デバッグセッションを削除し ます。



第6章 注意・制限事項

6.1 注意事項

- NEC Parallel Debuggerは、Eclipse PTPの最新版である Oxygen v3に対応しています。ご利用 に際しては、Eclipse PTP の Oxygen v3版をご利用下さい。
- デバッグ開始時にConsoleビューに以下のメッセージが表示されます。これは、MPIプロセス情報の取得のため、デバッグ開始時にmpiexecコマンドにgdbでアタッチする際に表示される警告メッセージです。MPIアプリケーションのデバッグに影響はありません。

Missing separate debuginfos, use: debuginfo-install glibc-2.17-157.el7.x86_64

• watch point, trace point の機能はご利用いただけません。

6.2 その他

 VE上で動作するアプリケーションについて、非MPIアプリケーションをデバッグする場合、 EclipseのCDTプラグインやFortranプラグイン等をご利用いただけます。その際はDebug Configuration 作成時に Debugger タブで行うデバッガの設定として、 VE 向け gdb (/opt/nec/ve/bin/gdb)を指定して下さい。

(ご参考)

各プラグインの詳細については、Eclipseの公式サイトのHelpより下記のユーザガイドをご 参照下さい。

- C/C++ Development User Guide
- Fortran Development User Guide

付録 A 改版履歴

版数	発行日	区分	追加・変更機能	対象箇所
初版	2018年5月			
2版	2018年12月	削除	変数参照の制限事項	6.2
3版	2019年5月	更新	図・説明の見直し	1.2

SX-Aurora TSUBASAシステムソフトウェア

NEC Parallel Debugger ユーザーズガイド

2019年5月 第3版

日本電気株式会社

東京都港区芝五丁目7番1号

TEL(03)3454-1111(大代表)

© NEC Corporation 2018-2019

日本電気株式会社の許可なく複製・改変などを行うことはできません。

本書の内容に関しては将来予告なしに変更することがあります。