D3.10 – Final Release of Debugging Tools

WP3: Development Environment

Project Acronym	CRESTA
Project Title	Collaborative Research Into Exascale Systemware, Tools and Applications
Project Number	287703
Instrument	Collaborative project
Thematic Priority	ICT-2011.9.13 Exa-scale computing, software and simulation

Due date:	M38
Submission date:	30/11/2014
Project start date:	01/10/2011
Project duration:	39 months
Deliverable lead organization	ALLINEA
Version:	1.1
Status	Final
Author(s):	David Lecomber (ALLINEA), Tobias Hilbrich (TUD)
Reviewer(s)	Alan Gray (UEDIN), Alistair Hart (Cray UK)

Dissemination level	
PU	*PU*

Version History

Version	Date	Comments, Changes, Status	Authors, contributors, reviewers
0.1	05/08/2014	First draft version	Tobias Hilbrich (TUD)
0.2	14/11/2014	Final version for reviewers	David Lecomber (ALLINEA)
1.0	28/11/2014	Final version for submission	Catherine Inglis (UEDIN)
1.1	5/12/14	Updated with official URL for the MUST release made for CRESTA	Catherine Inglis (UEDIN)

Table of Contents

Index of Figures

1 Executive Summary

This deliverable reports on the availability of and access to the final versions of the debugging tools developed in WP3 of the CRESTA project. This includes two tools:

- Allinea DDT *and*
- MUST

The first tool is a highly scalable debugger for parallel applications. It supports a wide range of programming languages and parallel programming paradigms. MUST provides automatic correctness checks that detect a wide range of MPI usage errors.

These tools aid efficient development workflows for highly scalable applications. Allinea DDT can be applied to investigate the root cause for faulty application behaviour. Without such a tool, developers can spend large quantities of time to find the root cause of an error. If an error only occurs at extreme scale (say above 100,000 processes), developers may even give up.

MUST on the other hand both serves to analyze the cause of an application crash due to an incorrect result—if the issue is MPI related—and to regularly check applications for hidden errors. Without this tool, many common mistakes have a high chance of not being spotted, especially type-matching and communication buffer errors.

The released versions of the tools include the extensions that the CRESTA proposal describes. This includes functionality extensions, such as novel correctness checks for MUST; usability extensions, such as the Allinea DDT-MUST integration; and scalability improvements, such as the scalable versions of MUST correctness checks. Further, we monitored the results of WP2 to understand available fault tolerance mechanisms. However, at the current point in time, it is not clear whether existing checkpoint-restart mechanisms for fault tolerance will suffice, or whether exascale software will require novel techniques. Both Allinea DDT and MUST support checkpoint-restart schemes, while they can often still operate/complete if a fault occurs.

2 Introduction

We report on the availability of both Allinea DDT and MUST in Section 3. Additionally we highlight the availability of an integration of the two tools that can combine their benefits.

2.1 Purpose

The purpose of this deliverable is to:

- Provide access information for the debugging tool Allinea DDT;
- Provide access information for the debugging tool MUST;
- Provide access information for the MUST-DDT integration.

2.2 Glossary of Acronyms

DDT   Distributed Debugging Tool
MUST   Marmot Umpire Scalable Tool

3 Debugging Tools

3.1 Allinea DDT

The release provided for CRESTA is hosted at

http://content.allinea.com/downloads/allinea-50-alpha-141110.tar

A full release including the changes for CRESTA will be integrated into a public release and hosted on http://www.allinea.com/downloads from January 2015.

Demonstration/testing licences are required to activate the software – which can also be obtained from www.allinea.com/trials.

3.2 MUST

The CRESTA release of MUST is based on version 1.3 of the Software. It incorporates smaller changes and bug fixes, visual/usage improvements, as well as the final version of the Allinea DDT-MUST integration. The release carries version 1.4rc1; a subsequent 1.4 release will then complete multithreading support features that are being developed outside of CRESTA. The release is available online at:

https://doc.itc.rwth-aachen.de/display/CCP/Project+MUST

The Software consists of three packages:

- PnMPI
- GTI
- MUST

The first package provides base services for MPI interception and a collaboration of fine-grain tool modules. PnMPI is jointly developed by the Lawrence Livermore National Laboratory, the RWTH Aachen, and TUD. Based on PnMPI, GTI adds tool infrastructure services and is developed by RWTH Aachen and TUD. Finally, MUST provides the actual MPI correctness checks and is developed by RWTH Aachen and TUD also. All three packages are available as open source and use a three-clause BSD license. The manual of MUST (doc/manual/manual.pdf) provides detailed steps for installation as well as a list of dependencies.

3.3 Allinea DDT-MUST Integration

The Allinea-MUST integration is described in D3.7 in Section 6.3. It can be activated with Allinea DDT 5.0 and the provided 1.4rc1 release of MUST. Once installed it uses a two-phase launch mode, which is subject to future improvement. In the first phase the user starts Allinea DDT (command line "ddt") and selects the manual launch mode (rather than the regular launch mode, Figure 1: Selection of the manual launch mode in Allinea DDT.). In the configuration dialog for the manual launch mode, the user then enables the MUST plugin for Allinea DDT (Figure 2: Manual launch mode configuration in Allinea DDT and selection of the MUST plugin.). Once the user starts the manual launch, Allinea DDT is ready to connect to the target application. The application is executed with MUST either with all analyses running or in the reproduce mode (see Deliverable D3.7). A simple invocation for the former is:

```
mustrun –must::ddt –np <X> <EXECUTABLE>
```

Immediately after the application starts up it connects to Allinea DDT (Figure 3: DDT is connected to an application that runs with MUST.). The user then controls the application in the graphical user interface of the debugger. Once MUST detects a correctness issue it will notify Allinea DDT, which can then display the message to the user (Figure 4: A breakpoint that is triggered by a MUST correctness message is triggered and displayed within Allinea DDT). Deeper integration provided by Allinea

DDT enables it to show MUST messages in its logbook view for rich error analysis (Figure 5: The logbook in Allinea DDT summarizes correctness messages from MUST and provides their detailed reports to the user.).
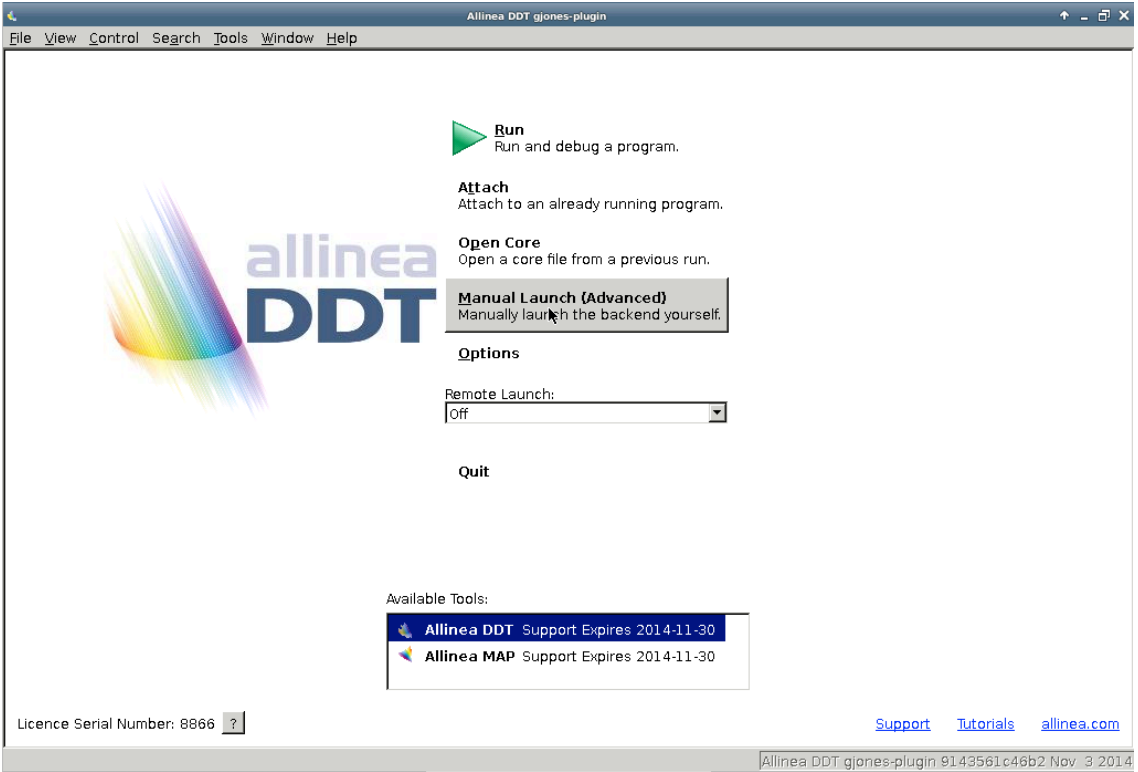

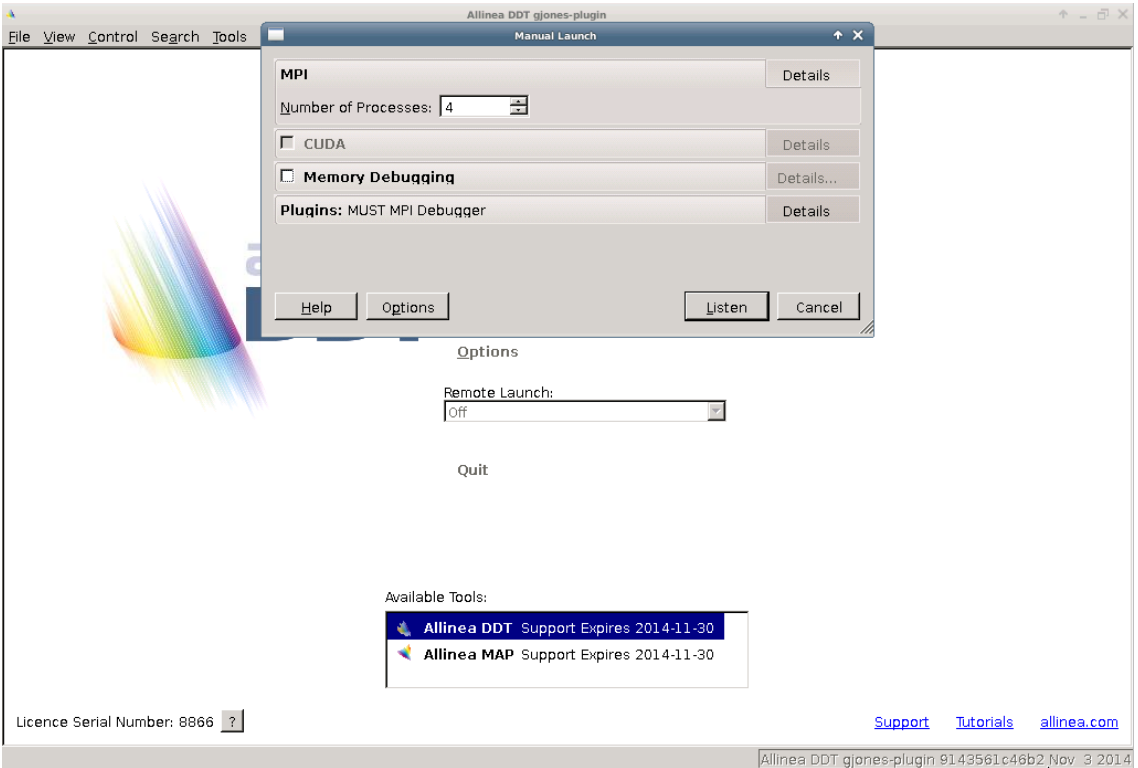**Figure 1: Selection of the manual launch mode in Allinea DDT.**

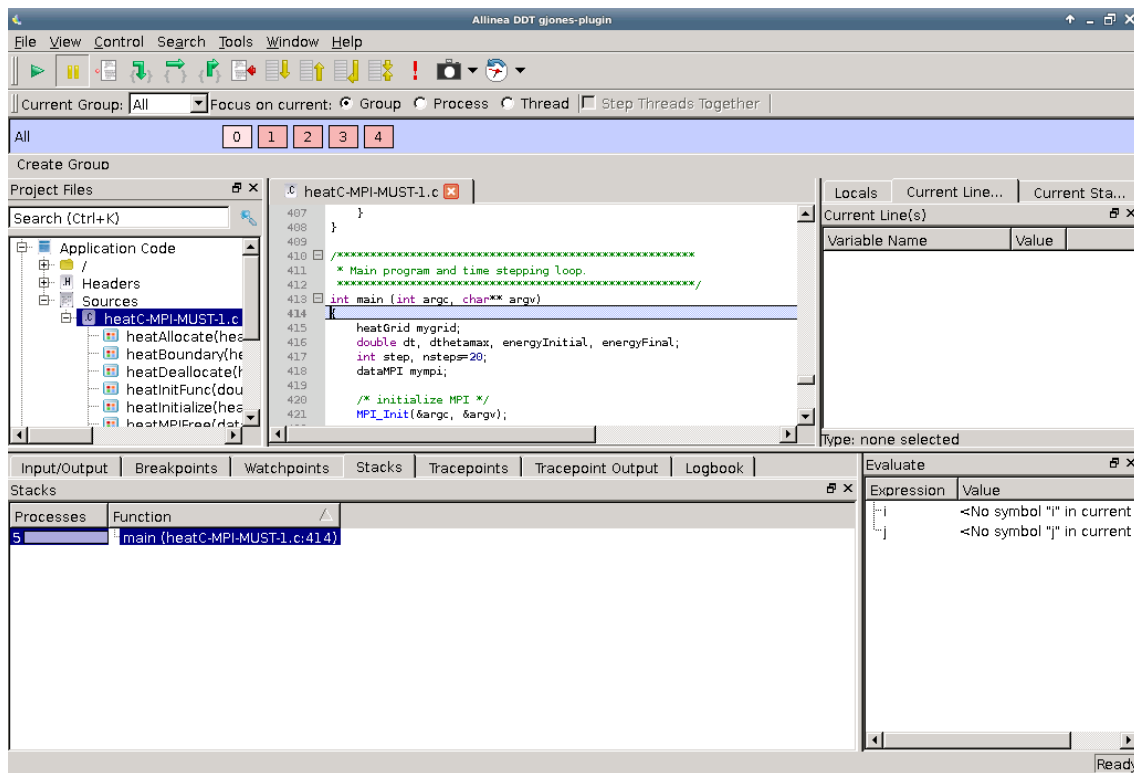**Figure 2: Manual launch mode configuration in Allinea DDT and selection of the MUST plugin.**

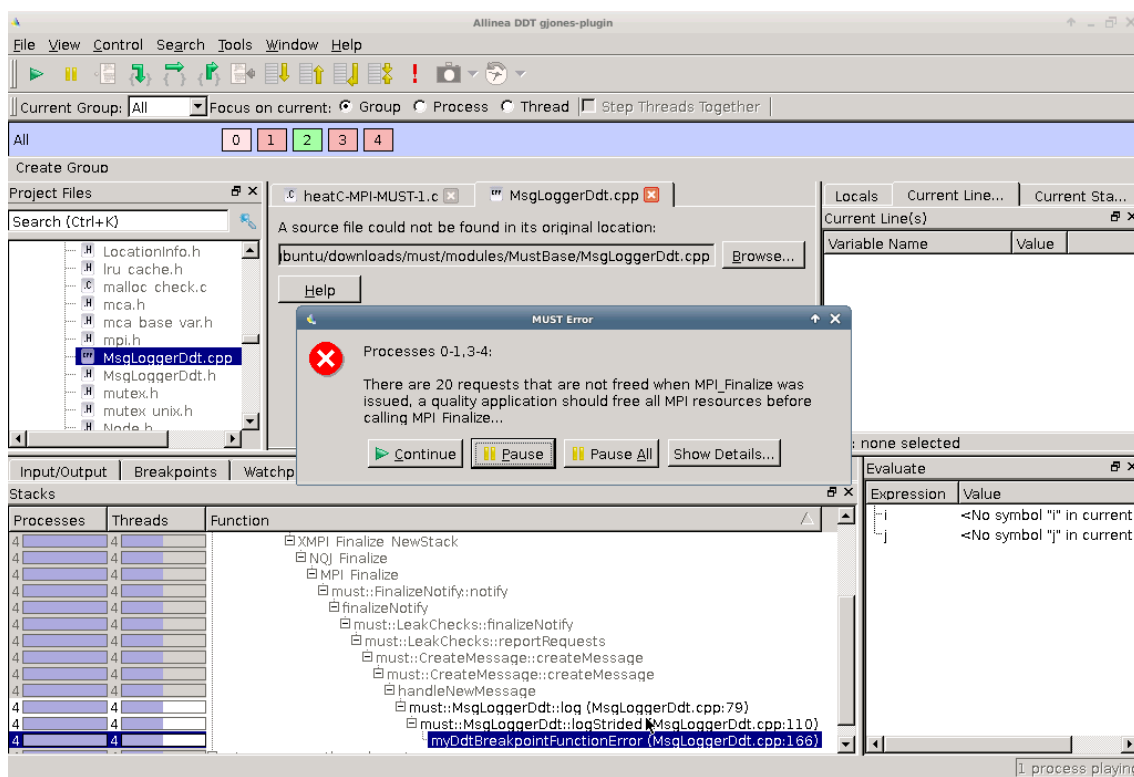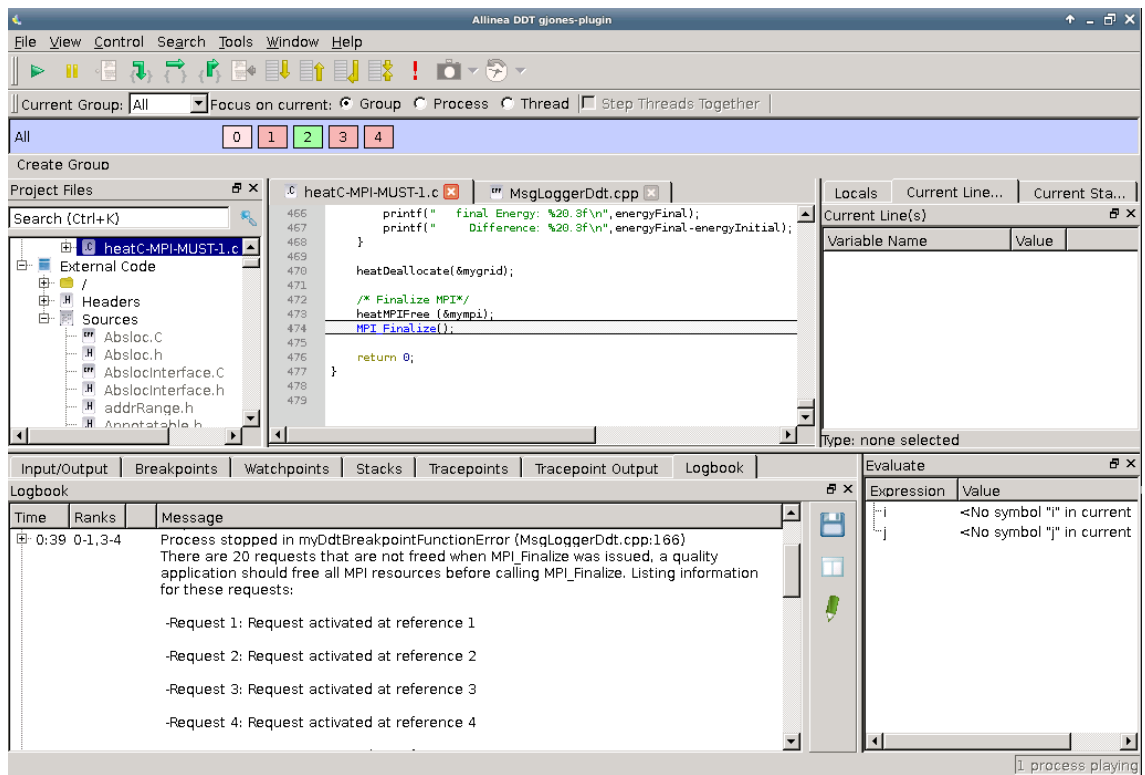**Figure 3: DDT is connected to an application that runs with MUST.**

Figure 4: A breakpoint that is triggered by a MUST correctness message is triggered and displayed within Allinea DDT

Figure 5: The logbook in Allinea DDT summarizes correctness messages from MUST and provides their detailed reports to the user.

4 Conclusions

We provide releases of Allinea DDT and MUST to application developers. The CRESTA releases include novel features to further support their needs at increased scale and with complex compute architectures. While deliverable D3.7 describes the individual extensions of the two tools, this deliverable summarizes access to the tool and showcases the new Allinea DDT-MUST integration that we developed within CRESTA.