1415 N CHERRY AVE
CHICAGO, IL 60642
(312) 281-6900
DMDII.ORG
DMDII@UILABS.ORG

DMDII
a UI LABS Collaboration
DIGITIZING AMERICAN
MANUFACTURING

DMDII FINAL PROJECT REPORT

Welding Cloud Computational Applications for Digital Manufacturing

Principle Investigator/Email Address	Yu-Ping Yang/yyang@ewi.org
Project Team Lead	EWI
Project Designation	15-16-03
UI LABS Contract Number	0320170001
Project Participants	Parallel Works, The Ohio Supercomputer Center, Applied Optimization, Composite Solutions, Gill Industries, Inc., Tesla, Westinghouse Electric Co., Scientific Forming Technology
DMDII Funding Value	$198,472.00
Project Team Cost Share	$206,194.00
Award Date	January 26, 2017
Completion Date	January 15, 2018

DISTRIBUTION STATEMENT A. Approved for public release.

This project was completed under the Cooperative Agreement W31P4Q-14-2-0001, between Army Contracting Command - Redstone and UI LABS on behalf of the Digital Manufacturing and Design Innovation Institute. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of the Army.

TABLE OF CONTENTS

I. EXECUTIVE SUMMARY

Welding is an important manufacturing process used in a broad range of industries and market sectors, including automotive, aerospace, heavy manufacturing, medical, and defense. It is a critical core competency that enables these industries and market sectors to support design innovation, with significant positive impact to employment, national security, the environment, and the general US economy. Designing, fabricating, and testing a welded structure to meet the application requirements generally is a time consuming and costly process. Welding processes often result in the softening of the weld heat-affected zone (HAZ), complicating prediction of the weld performance. Welding processes also induce residual stress and distortion in the welded structures, further complicating the implementation of new materials and weld designs.

Through weld modeling technology development, welding process parameters, fixtures, and tooling can be optimized to reduce the HAZ softening and minimize weld residual stress and distortion, improving performance and reducing design, fabrication, and testing costs. However, weld modeling technology tools are currently accessible only to engineers and designers with a background in finite element analysis (FEA) who work with large manufacturers, research institutes, and universities with access to high-performance computing (HPC) resources. Small and medium enterprises (SMEs) in the US do not typically have the human and computational resources needed to adopt and utilize weld modeling technology.

To address these issues, EWI teamed with OSC and Parallel Works to develop new online welding software modeling tools for digital manufacturing using the cloud for high performance computations. The tools used three-dimensional (3D) models and are applicable to the three most common types of welding processes: arc welding, laser welding, and hybrid laser arc welding. Standard weld joints, butt joint, T-joint, and lap joint, were included in the software tool. Open source finite element codes were used as a solver and as pre- and post-processing tools.

A new web application front end was developed that allows interactive rotation and zoom controls by a user. A python script was developed for automatic mesh generation of the three welded joints using an open-source program, SALOME, based on user input parameters. A material property database containing data for commonly used steels, aluminum alloys, and titanium alloys as well as corresponding filler wires was established. Weld modules for arc welding, laser beam welding, and hybrid laser arc welding were developed which can work with an open-source solver, CalculiX, to simulate welding processes to predict temperature, stress, and distortion. An open-source software, ParaView, has been identified for post processing of CalculiX results. A simulation flow and script has been developed for automatic analyses of welding processes. The development has reached the point where the developed pieces are ready to integrate into one software and install the package in digital manufacturing commons (DMC), OSC, and Parallel Works.

DMDII has chosen to terminate this project, since DMDII has ceased investment in the DMC in response to feedback provided by their Executive Committee. This has left some planned project activities incomplete. The following tasks remain outstanding as of the notice of termination:

- DMC software integration
- Software testing and validations
- Technology transfer, workforce development, and education

II. PROJECT REVIEW

Welding is a vital and fundamental manufacturing process used in many industries including automotive, aerospace, heavy manufacturing, medical, and defense, among others. Designing, fabricating, and testing a weld structure to meet product application requirements is a time-consuming and costly process. An example of this can be seen in the design of structural welds. Structural welds are designed to meet performance requirements for specific applications (*e.g.*, load, impact, collision, and fatigue). Implementing new connection designs, materials, and manufacturing methods often requires expensive and time-consuming prototyping and testing. This has become a serious problem as companies are driven to reduce product development timelines and costs, while improving weld structure performance.

As new materials are increasingly used in welded structures, designing and fabricating a welded structure to meet weld joint performance requirements becomes even more challenging. This is because welding processes damage the heat treatment conditions of these materials, resulting in softening of the weld HAZ. Current welded structure design methods ignore the effects of welding on part performance, including residual stresses and microstructural change in the weld region. Existing finite element models of part designs cannot predict the weld connection behavior accurately because most models do not have the capability to predict weld residual stress and strain. This leads to poor design of welded structures, and it results in unnecessary costs, timeline delays, material waste, safety concerns, and, in the worst case, structure failure during service.

Cloud welding computations are a new technology to optimize material, design, and manufacturing processes in a virtual environment using advanced computing, modeling, simulations, and data analysis. The implementation of the tool will result in numerous benefits to the US economy, including accelerated product development, reduction in time to market, cost reduction, broader adoption, and improved product quality. Because of the rapid development of modeling and simulation technology and the low cost of computational hardware, cloud computations have been growing in recognition and acceptance worldwide. To improve US industrial competitiveness, an innovative new software tool and cloud-based delivery service to perform welding computations has been advanced in this project.

This project has been developing new web-accessible welding software tools for digital manufacturing and integrating them into a cloud-based welding simulation service as shown in Figure 1. The application used 3D models to simulate gas metal arc welding (GMAW), laser beam welding (LBW), and hybrid laser arc welding (HLAW) of three classes of metals (i.e. steel, aluminum alloys, and titanium alloys). The model geometries of the welded structures initially included plates and pipes. Simulated weld joints included butt, T-, and lap joints. The applications predict temperature, residual stress, and distortion for a given set of user inputs. After development, industrial partners were to exercise and test the tools and compare the model prediction with experimental measurement data as a part of the effort to verify and validate the applications.

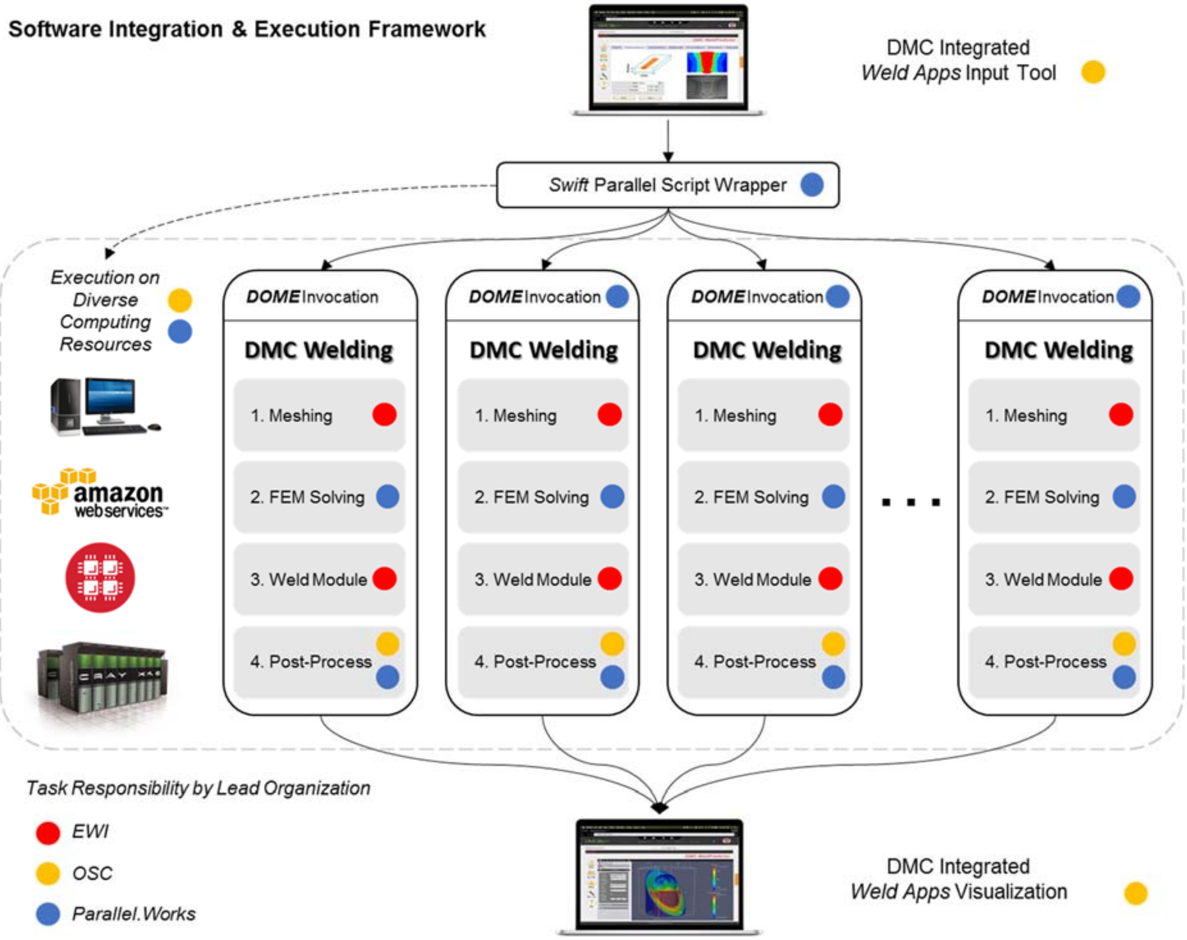


Fig. 1 Software Integration and Execution Framework

Based on EWI's current experience with the existing WeldPredictor product, EWI believes that the proposed cloud-based welding simulation service would empower US manufacturers to design and fabricate welded structures meeting specific product performance requirements significantly faster and at a far lower cost. This will in turn deliver significant benefits to these US manufacturers both large and small, including continued technology innovation in materials used for welded structures, improved welded structure design, improved manufacturing output, reduced design and product development cycle times, increased competitiveness, reduced waste, and added employment. In addition, the technology will enable engineers to imagine and implement welding technology where other, less capable joining technologies have traditionally been used, adding even more value to the application.

Figure 2 shows the process flow design for the cloud-welding simulation applications. It includes a front end and a back end. The front end consists of web pages that enable a user to define geometry, select materials, set welding parameters, provide weld fixture conditions, monitor analysis, and receive a simulation report. The back end consists of an automatic mesh generator, a weld module, a finite element solver, and a post-process software tool. The front end has been developed by OSC and the back end computational service has been developed by EWI and Parallel Works. The focus of this phase of the project was on two simple 3D structures: a plate and a pipe. Once successful, the software would be extended to model a complex welded structure in a follow-on project.

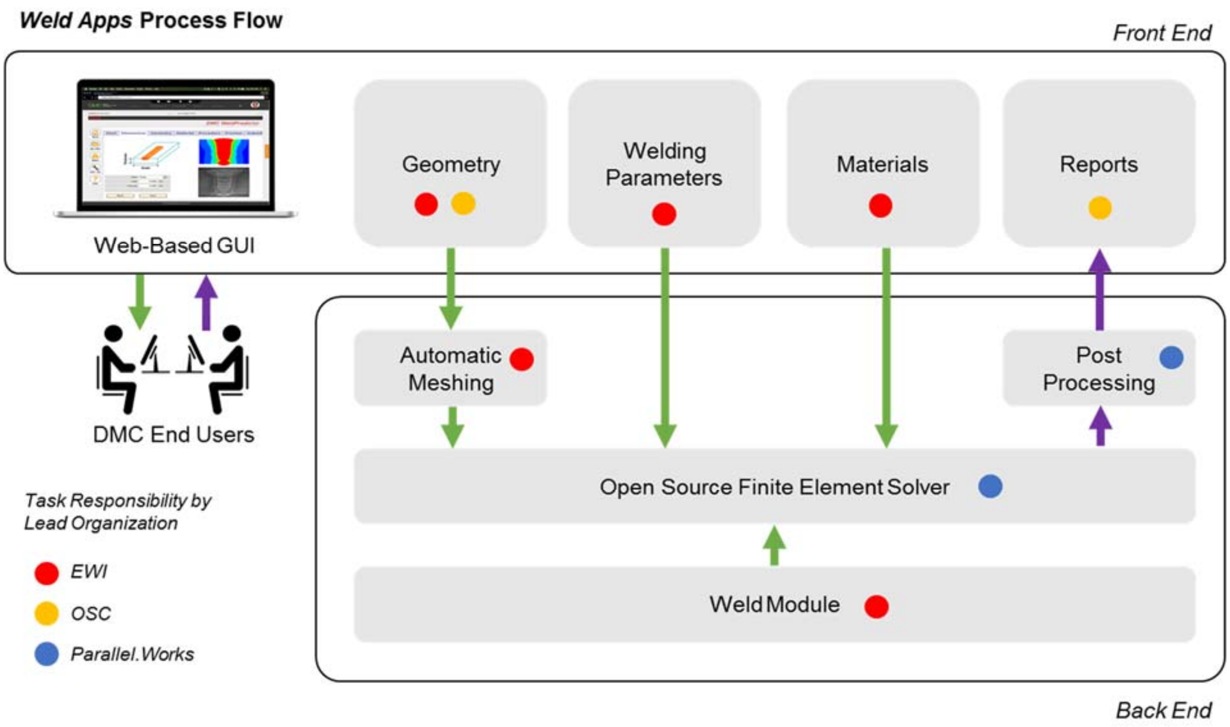
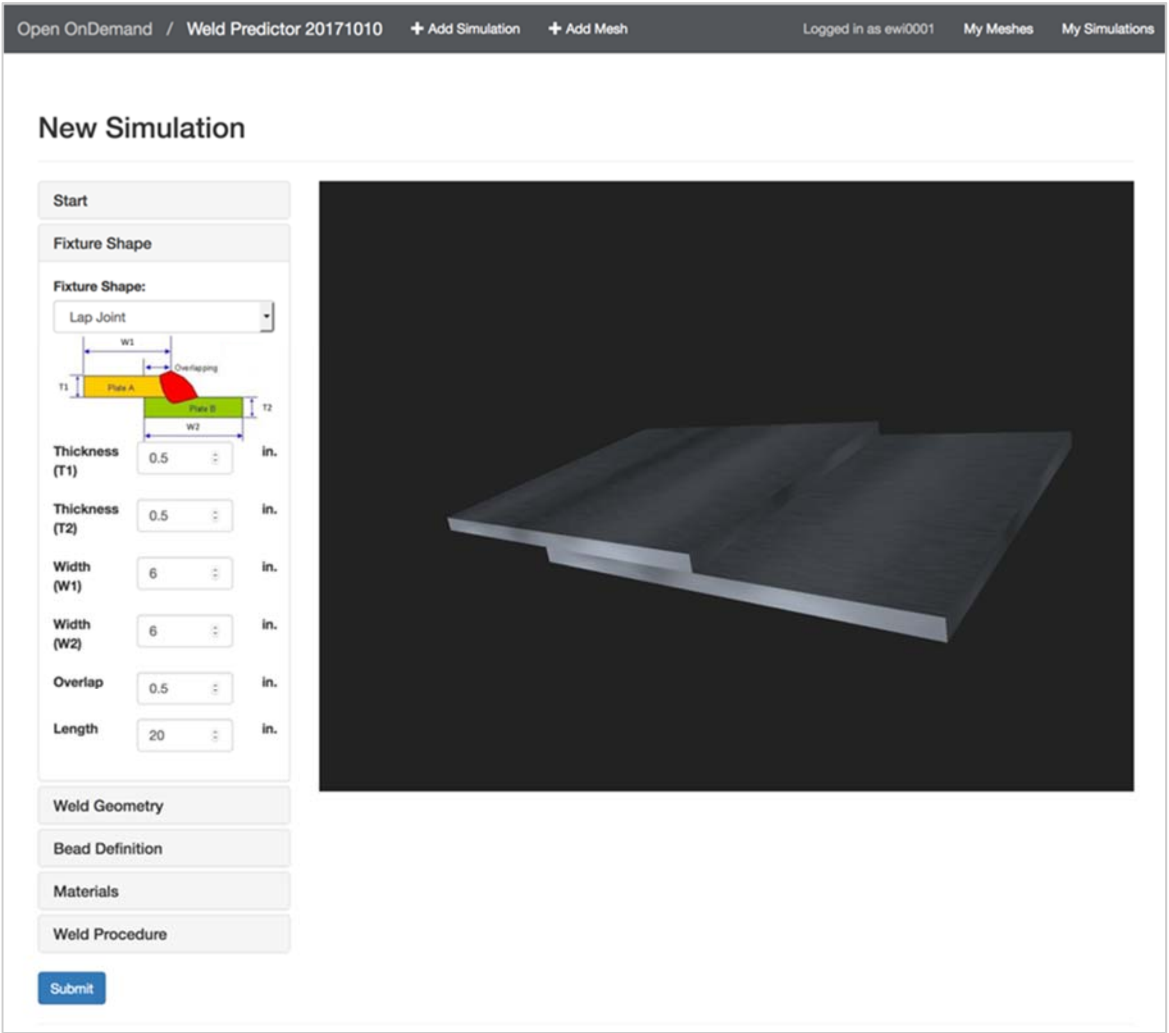


Fig. 2 DMC Weld Apps Process Flow Diagram

Task 1 –Weld Simulation App Development (OSC lead with EWI support)
A web user interface (UI) was designed and built for the open source cloud-based application for welding simulation, as shown in Figure 3. The UI consists of display forms and controls allowing users to create, customize, execute, and inspect weld simulations. The UI encapsulated the workflow of defining a weld simulation, including specifying (1) welding process, (2) weld geometry (i.e., dimensions of welded plates or pipes), (3) weld pass location and geometry via a 3D interface with interactive rotation and zoom control, (4) material selection for both base and filler materials, (5) material property customization, (6) weld parameter input including preheating and inter-pass temperatures, (7) weld fixture definition and inspection via a 3D interface with interactive rotation and zoom control. The weld fixture definition and inspection portion of the UI was not completed at the end for the project.



New Simulation

Fixture Shape: Lap Joint

Thickness (T1) 0.5 in.

Thickness (T2) 0.5 in.

Width (W1) 6 in.

Width (W2) 6 in.

Overlap 0.5 in.

Length 20 in.

Fig. 3 Web Interface of Welding Cloud-Based Computational Tool

Task 2 –Material Property Data Development (EWI lead)
Temperature dependent thermal-physical and mechanical material data for six base materials including steels, aluminum alloys, and titanium alloys and corresponding filler materials have been built into the application.

Task 3 –Automatic Mesh Generation (EWI lead)
A Python script was developed to create finite-element mesh automatically with Salome, based on the geometry information collected in the UI developed in Task 1. SALOME is an open-source software that provides a generic platform for pre- and post-processing for numerical simulation. SALOME is a cross-platform solution**.** Figure 4 shows the interface of Salome. Figure 5 shows an example of created meshes for a multi-pass V-groove weld. The user-defined weld fixture conditions in the web interface in Task 1 was converted to numerical thermal and mechanical boundary conditions. User input files were created for running the analysis.

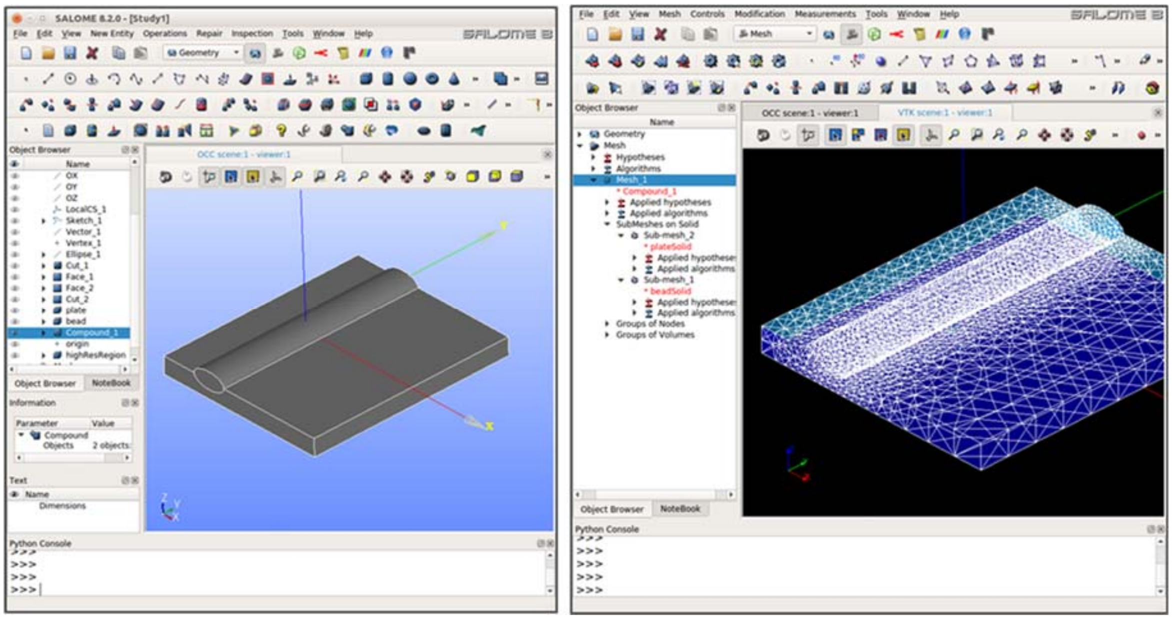


Fig. 4 SALOME Interface for Creating Geometry and Mesh for a Bead-On-Plate Weld

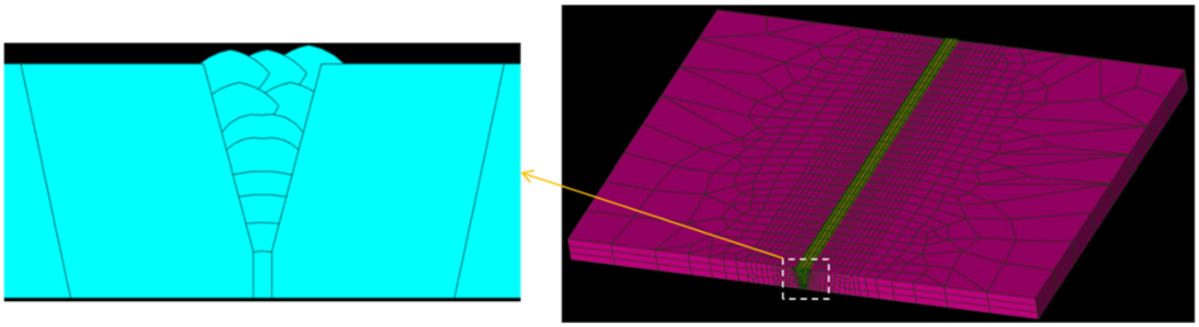


Fig. 5 Automatically Created Mesh with SALOME

Task 4 –Develop Weld Module (EWI lead)
A Fortran computer program was developed to implement the heat source models for GMAW, LBW, and HLAW to calculate heat flux at a given time and at a given location in the weld joints. The code was used as a user subroutine in the finite-element solver, CalculiX, for thermal-mechanical analysis to predict temperature, stress, and deformation.

Task 5 –Open Source Thermal-Mechanical Solver (Parallel Works lead with EWI support)
Open-source code, CalculiX, was identified as a thermal-mechanical solver for welding simulation from a list of open-source codes. CalculiX is a free finite-element software package including pre- and post-processor and a solver. The solver can do linear and non-linear calculations. Static, dynamic and thermal solutions are available. Because the solver makes use of the Abaqus input format it is possible to use commercial pre-processors as well. The program is designed to run on Unix platforms like Linux computers, but also on MS-Windows. The CalculiX package was developed by a team of enthusiasts working at MTU Aero Engines in Munich, Germany. The development in this project used version 2.12. The latest version, 2.13, was released on October 8, 2017.

CalculiX and the weld module developed in Task 4 was used to simulate welding processes in terms of temperature, stress, and distortion. Welding parameters are input to the weld module for calculating heat flux at each time increment. A parametric bead-on-plate model was used as the first test case, as shown in Figure 6, with discrete mesh elements for the bead and plate, each with unique material properties.

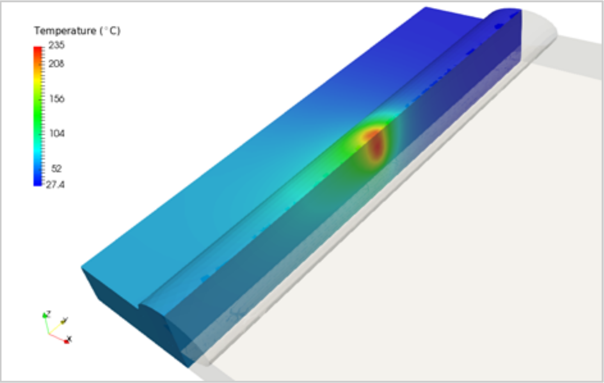


Fig. 6 Predicted Temperature on a Bead-on-Plate Weld

CalculiX and the weld module were tested with a multi-pass, V-groove weld model. Weld-pass depositions were modeled with an element-rebirth technique, as shown in Figure 7. In this technique, the element sets representing each pass are generated. During analysis, elements associated with the higher numbered passes are first removed and then reactivated at the beginning of the appropriate weld pass. Figure 8 demonstrated an extreme case in which the predicted temperatures is about 2.5 to 5 times the material melting temperature, and CalculiX can still obtain a converged solution.

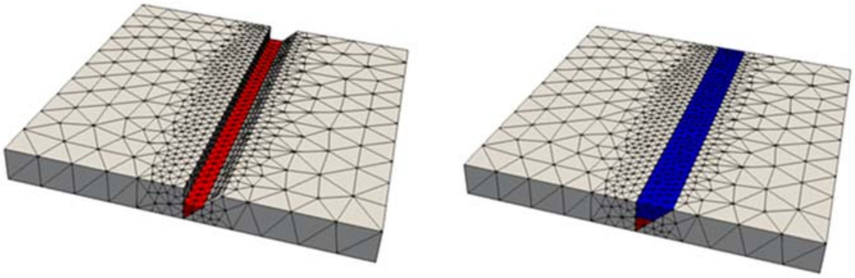


Fig. 7 Modeling the Addition of Filler Metal for a Multi-Pass Weld

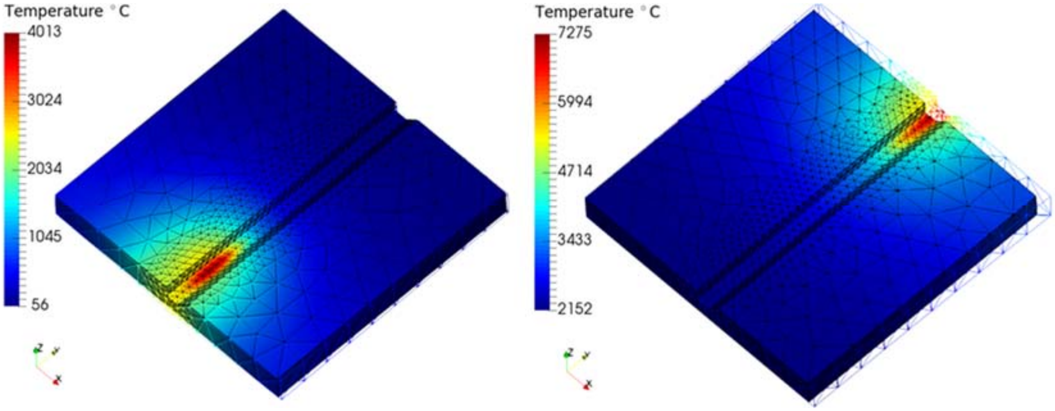


Fig. 8 Modeling the Over-Heating of Welding Filler Metal

Calculix was also tested in the thermomechanical simulation of a complex welding case, as shown in Figure 9, using multiple threads. Simulating and optimizing simulation control parameters for this challenging example were investigated to obtain satisfactory results, which are comparable to the commercial finite-element program Abaqus to ensure that CalculiX can successfully handle the most complex welding problems.

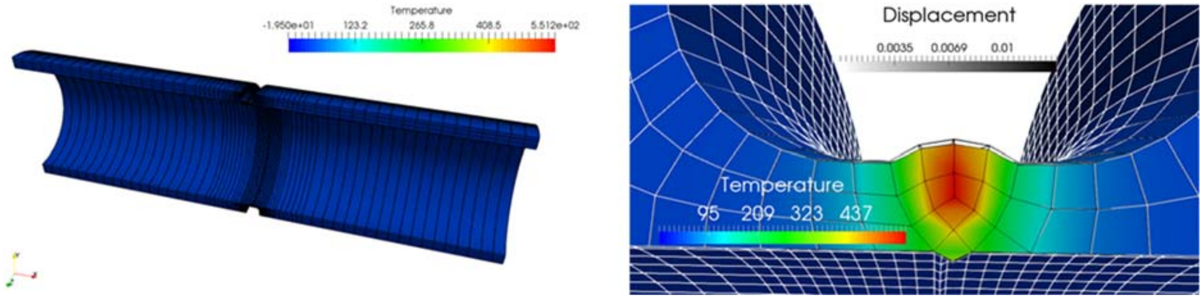


Fig. 9 Modeling the Over-Heating of Welding Filler Metal

Task 6 –Automatic Post-Processing (Parallel Works lead with EWI support)
ParaView has been identified for post-processing of CalculiX analysis results. ParaView is an open-source, multi-platform data analysis and visualization application. ParaView was developed to analyze extremely large datasets using distributed memory computing resources. It can be run on supercomputers to analyze datasets of petascale size as well as on laptops for smaller data. The data exploration can be done interactively in 3D or programmatically using ParaView's batch processing capabilities.

A Python library has been developed for automated post-processing of CalculiX results using ParaView. Analysis results have been successfully shown in ParaView, as shown in Figure 10. The results including distributions of temperature, stress, and distortion, can be presented using contour plots. The temperature history and cooling rate can be provided as line plots.

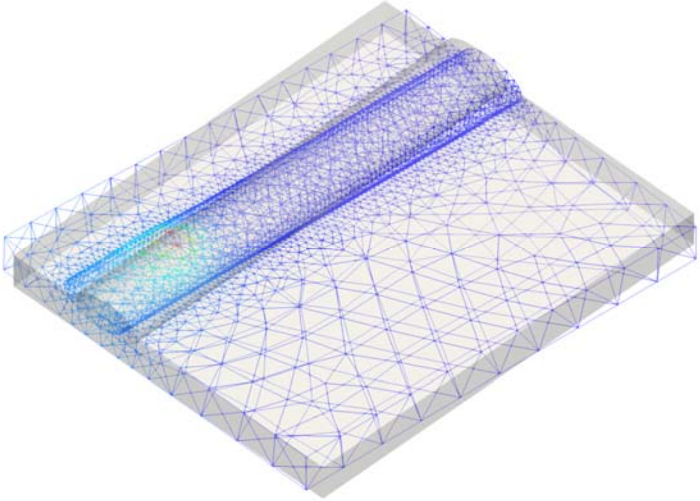


Fig. 10 Modeling the Over-Heating of Welding Filler Metal

Task 7 –Analysis Report Generation (OSC lead with EWI support)
Weld simulation output would be made accessible via the web using ParaView. The app would provide a PDF report summarizing inputs and outputs. The outputs include contour plots of temperature, stress, and distortion and line plots of temperature, stress, and distortion history.

This task had not been performed before the project's termination.

Task 8 –DMC Software Integration (Parallel Works lead and OSC and EWI support)
This task was partially completed before the project's termination. Software integration into a product available with a single request has been started, but has not been completed. Parallel Works proposed to replace this task with software deployment and integration to result in a standalone, portable deployment of the 3D WeldPredictor application, which would be available to DMDII members and to companies in a broader range of industries. However, this plan was not accepted by DMDII.

Integration preparation
A Swift workflow was created and made runnable that couples user inputs (Task 1), material data (Task 2), the geometry creation, meshing (Task 4), CalculiX solving (Tasks 3 and 5), and ParaView post-processing (Tasks 6 and 7) together. The workflow can be used to run a single case, or run a parameter sweep through multiple cases, which was demonstrated in the quarterly meeting on October 17, 2017. Parallel Works created an Amazon Machine Image (AMI) with pulled dockerized images of required software modules for the welding workflow to enhance the performance of running and testing the welding workflow on Amazon Compute Cloud (EC2).

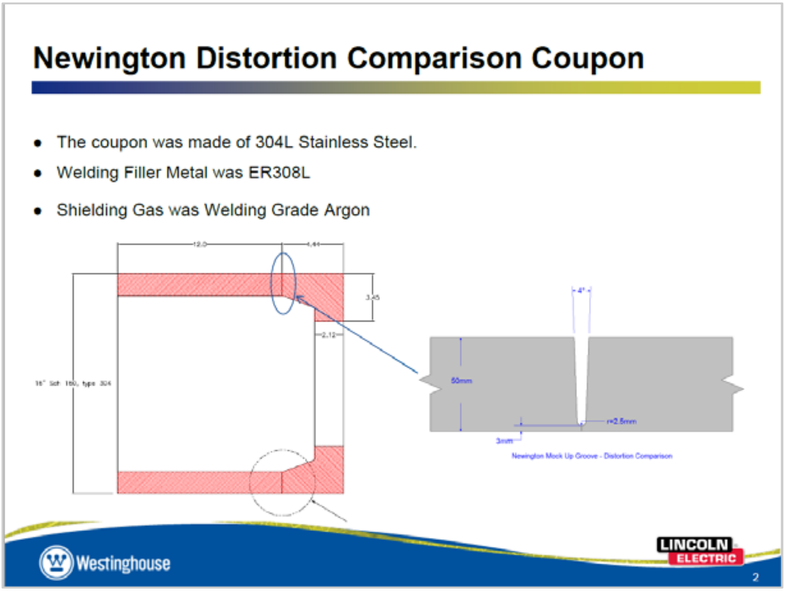
OSC has installed the SALOME pre-processing application for mesh-generation, as well as built and made available a version of the CalculiX FEA solver with thread-level (OpenMP) and node-level (MPI) multiprocessing support. The modifications for building the solver in this configuration were integrated into the codebase of the application itself.

Option to replace this task with software deployment and integration:
In place of the DMC/DOME software integration, Parallel Works considered modifying this task to create a standalone, portable deployment of the 3D WeldPredictor application. Parallel Works would have made a portable implementation of the WeldPredictor application to run on a desktop computer or on cloud HPC platforms. DMDII did not implement this option.

Implementation of this modification would have resulted in a standalone, open-source, runnable project, and a hosted SaaS offering running on Parallel Works. It would have included a fully dockerized backend Swift workflow to support the portability and ease of integration into open HPC platforms, and would have been coupled to a standalone version of the 3D interface developed by OSC in Task 1. Parallel Works would have provided documentation needed to run the WeldPredictor application standalone from a github repository. Parallel Works would have also investigated the integration of design exploration methods as well as the Swift workflow, and they would have prototyped the workflow enhancements on the Parallel Works platform.

Task 9 –Software Testing and Validations (Industrial partners lead with EWI support)
This task was partially completed before the project's termination. Westinghouse Electric Company conducted welding experiments to collect measurement data for calibrating and validating the finite-element models. Filler metals, welding parameters, and welding fixtures, as shown in Figure 11, were selected based on the standards and codes of the American Welding Society (AWS).

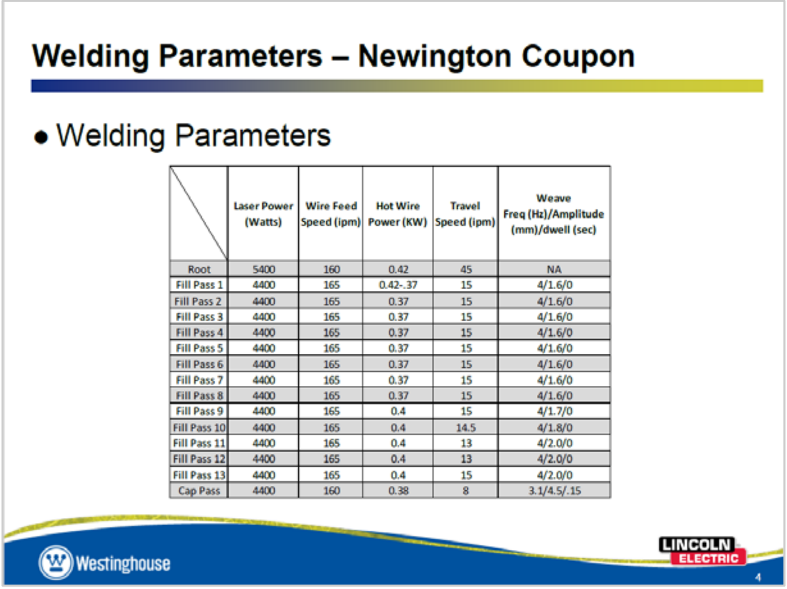
If the cloud-welding simulation application were to be completed, the functions of the application should be tested by industrial partners for predicting temperature, residual stress, and distortion and comparing the model predictions with the experimental data. Industrial partners could then provide feedback to the development teams to confirm the developed software meets their welding development requirements.



Newington Distortion Comparison Coupon

- The coupon was made of 304L Stainless Steel.
- Welding Filler Metal was ER308L
- Shielding Gas was Welding Grade Argon

2

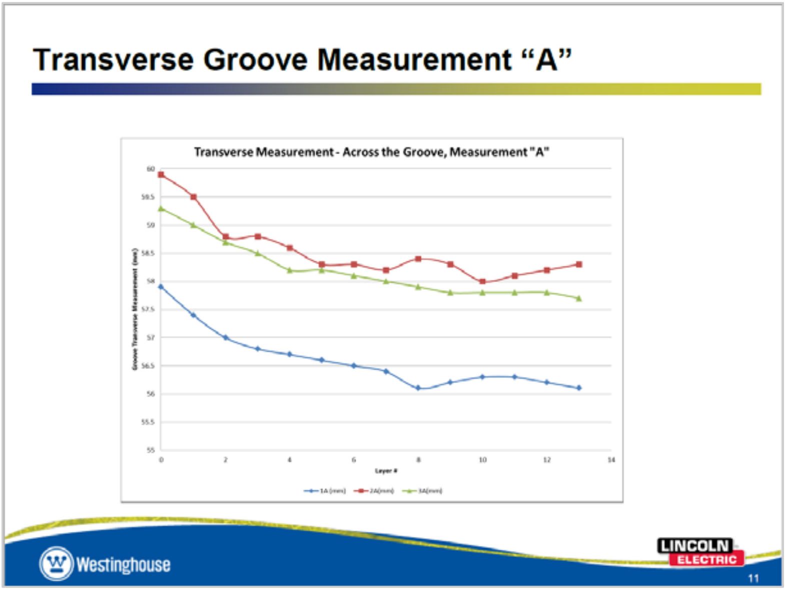


Welding Parameters – Newington Coupon

- Welding Parameters

	Laser Power (Watts)	Wire Feed Speed (ipm)	Hot Wire Power (KW)	Travel Speed (ipm)	Weave Freq (Hz)/Amplitude (mm)/dwell (sec)
Root	5400	160	0.42	45	NA
Fill Pass 1	4400	165	0.42-.37	15	4/1.6/0
Fill Pass 2	4400	165	0.37	15	4/1.6/0
Fill Pass 3	4400	165	0.37	15	4/1.6/0
Fill Pass 4	4400	165	0.37	15	4/1.6/0
Fill Pass 5	4400	165	0.37	15	4/1.6/0
Fill Pass 6	4400	165	0.37	15	4/1.6/0
Fill Pass 7	4400	165	0.37	15	4/1.6/0
Fill Pass 8	4400	165	0.37	15	4/1.6/0
Fill Pass 9	4400	165	0.4	15	4/1.7/0
Fill Pass 10	4400	165	0.4	14.5	4/1.8/0
Fill Pass 11	4400	165	0.4	13	4/2.0/0
Fill Pass 12	4400	165	0.4	13	4/2.0/0
Fill Pass 13	4400	165	0.4	15	4/2.0/0
Cap Pass	4400	160	0.38	8	3.1/4.5/.15

4



Transverse Groove Measurement “A”

Fig. 11 Experimental Data for Software Validation

Task 10–Reporting and Communication (EWI lead)
The project team held bi-weekly teleconferences during the project to discuss the progress and the issues of the project. Two face-to-face meetings were held at the DMDII UI Lab to update the project results. A quarterly meeting was conducted to provide an update of the project results to DMDII members on October 17, 2017.

Monthly Word reports and a quarterly PowerPoint report were prepared during the project to provide summaries of high-level technical status, project risks, and opportunities, schedule status and/or schedule modifications, project issues, budget expenditures, and cost share.

This final technical report provides a comprehensive, cumulative, and substantive summary of the progress and significant accomplishments achieved during the period of the project effort.

Task 11–Technology Transfer, Workforce Development, and Education (EWI lead)
If the project had been completed, two software training webinars were planned for the members of DMDII to transfer the developed welding simulation service. The cloud-based welding simulation application would have been demonstrated during the training webinars. EWI would have trained DMDII members on the use of the welding applications during the training.

III. KPIs and METRICS

Metric	Baseline	Goal	Results	Validation Method
Web interface	Available for 2D	Develop for 3D	Developed	Code testing
Material property data	Available for steels	Add for aluminum and titanium alloy	Found	Property data is available
Weld module	Available	Adapted for open source solver	Working	Code testing
Automatic weld mesh generation	Not available	Created	Working	Code testing
Open source solver for welding simulation	Not available	Created	Working	Code testing
Automatic post-processing of results	Not available	Created	Working	Code testing
Integrated software	Not available	Created	Not complete	Software testing
Software testing and validation	Not performed	Complete	Not done	Software testing and physical experiment
Technology transfer	Not performed	Transfer	Not done	Training class

IV. TECHNOLOGY OUTCOMES

This project developed a cloud-based welding simulation application with open-source codes, Salome (pre-processing), CalculiX (solver), and ParaView (post processing). It greatly advances the welding simulation technology, produces a low-cost welding simulation tool for industrial uses, and creates innovation for welding simulation. This application, if completed, could be run in three ways: local computers, high-performance computers hosted in OSC, and Amazon Compute Cloud hosted by Parallel Works.

As shown in Figure 2, the application includes a front end and a back end. The front end consists of web pages that enable a user to define geometry, select materials, set welding parameters, provide weld fixture conditions, monitor analysis, and receive a simulation report. The back end consists of an automatic mesh generator, a weld module, a finite-element solver, and a post-process software tool.

This application includes key features and attributes designed to allow the user to:

- Model the welding process in a cloud environment.
- Accept guidance through a user-friendly web interface to conduct a welding simulation.
- Use open source, finite-element codes for welding simulation.
- Accelerate welding simulation through parallel computation.
- Use a 3D model to simulate an arc welding, laser welding, and hybrid welding processes.
- Analyze common materials including steels, aluminum alloys, and titanium alloys.
- Predict temperature, stress, and distortion induced by the welding process.

The remained work not completed at the close of the project includes:

- Automatic output report generation.
- Integration of the cloud-based welding simulation application.
- Software development documentation/design document:
    - An end user tutorial and documentation would have been prepared to guide users to apply the developed applications in welding process development.
- Software training:
    - Training materials including presentations and user cases would have been developed and provided in a training webinar.
    - The software training would have been made available to the members of DMDII for workforce development, technology transition, and education.

III. ACCESSING THE TECHNOLOGY

The application was designed to be accessed using a personal computer, a smart phone, or a tablet from an office, manufacturing floor, or the field—anywhere with access to the internet. For example, a welding engineer would have been able to apply the welding cloud computational applications to develop welding procedures by taking the following steps:

1. Login to the application with the user's credential on a computer or a tablet;
2. Input geometry dimensions, welding sequence, and welding parameters;
3. Define numerical boundary conditions based on welding fixture conditions;
4. Submit the analysis—the analysis would be run in a remote high-performance computer;
5. Check analysis results: temperature, residual stress, and distortion to compare with objectives.

If the prediction cannot meet the design requirements, the user would have been able to repeat the above five steps by adjusting the input parameters. Once an optimal welding procedure was determined, the welding engineer could then conduct limited experiments to fine-tune the procedure.

VI. INDUSTRY IMPACT AND POTENTIAL

The cloud-based welding application was designed to be used in any industry or market sector where welding or joining of materials is part of the manufacturing process, within both small and large companies. The users could be a welding engineer, a material engineer, an industrial engineer, and a manufacturing engineer.

The application was designed to be a valuable tool for exploring "what if" scenarios in a reliable, cost effective, and rapid manner. Application of this technology was aimed at greatly reducing the time and cost to design and fabricate weld structures that reliably meet application requirements, while also expanding the menu of available options for the welding or design engineer to use in their efforts. Possible next steps were to extend the application to simulate a complex welded structure.

VII. TECH TRANSITION PLAN AND COMMERCIALIZATION

If the application development were to be completed, it would be presented at DMDII, and conference and training webinars should be conducted to transfer the developed technology to DMDII members and broader industries. The tools and software developed in this project are open source, which can be made available and downloaded by DMDII members and broader industries for their applications.

The developed welding applications are available to be commercialized after the project closeout.

VIII. WORKFORCE DEVELOPMENT

The developed cloud-welding computational application could be used in classrooms of colleges and universities to teach students how to conduct welding simulation. It could also be used in companies to train engineers to understand the application of welding simulation.

IX. CONCLUSIONS AND RECOMMENDATIONS

The following conclusions could be drawn from this project:

- Cloud computational welding application is a potential technology to reduce the time and cost to design and fabricate weld structures.
- Open source codes are capable of welding simulation including pre-processing for mesh generation, solving thermomechanical problem, and post-processing of analysis results.

The following recommendations could be made from this study:

- Completion of development should be done to deliver this cloud computation welding application for industries involving welding process in fabrication.
- Further development should be done to create a standalone, portable deployment of the 3D WeldPredictor application that can be run on a desktop computer or on cloud HPC platforms.

X. LESSONS LEARNED

Efforts were made to overcome issues than originally planned. The following lessons were learned during development:

- The web interface had to be rewritten for dynamically showing a 3D model.
- A different algorithm had to be used in creating weld passes for creating a high-quality, finite-element mesh.
- Numerical convergence is a challenging issue for using open-source codes for welding simulation.

XI. DEFINITIONS

2D	two-dimensional
3D	three-dimensional
CalculiX	an open-source finite element software package
DMC	digital manufacturing commons
GMAW	gas metal arc welding
HAZ	weld heat affected zone
HLAW	hybrid laser arc welding
HPC	high-performance computing/computer
LBW	laser beam welding
ParaView	an open-source, multi-platform data analysis and visualization application
SALOME	an open-source software for pre- and post-processing of numerical simulations
UI	web user interface

XII. APPENDICES

How to Install Calculix 2.12 Multi-Thread on Ubuntu

Reference:
http://www.libremechanics.com/?q=node/9
http://www.dhondt.de/
http://www.netlib.org/linalg/spooles/spooles.2.2.html
http://www.caam.rice.edu/software/ARPACK/

1. Installing all the necessary tools. The system will say the latest version has been installed.

```
sudo apt-get install gcc
sudo apt-get install gfortran
sudo apt-get install make
sudo apt-get install libcnf-dev
sudo apt-get install liblapack-dev
sudo apt-get install liblapack3gf
sudo apt-get install f2c
sudo apt-get install libblas-dev
sudo apt-get install psutils
sudo apt-get install imagemagick
sudo apt-get install gnuplot
sudo apt-get install libextutils-f77 (may be not needed)
```

2. Download all files.

   a. The Calculix 2.12 source code

      ccx_2.12.src.tar.bz2
      http://www.dhondt.de/ccx_2.12.src.tar.bz2

   b. SPOOLES 2.2: SParse Object Oriented Linear Equations Solver

      spooles.2.2.tgz
      http://www.netlib.org/linalg/spooles/spooles.2.2.tgz

   c. ARPACK is a collection of Fortran77 subroutines designed to solve large scale eigenvalue problems.

      arpack96.tar.Z
      http://www.caam.rice.edu/software/ARPACK/SRC/arpack96.tar.Z
      patch.tar.Z
      http://www.caam.rice.edu/software/ARPACK/SRC/patch.tar.Z

3. Copy the downloaded files to /usr/local/.

4. Untar the downloaded files.

```
untar Calculix source code
$ sudo tar -xvf ccx_2.12.src.tar.bz2
untar spooles file
$ sudo mkdir SPOOLES.2.2
$ sudo mv spooles.2.2.tgz /usr/local/SPOOLES.2.2/.
$ cd SPOOLES.2.2
$ sudo tar -xvf spooles.2.2.tgz
untar ARPACK
$ sudo tar -xvf arpack96.tar.Z
$ sudo tar -xvf patch.tar.Z
```

5. Compile SPOOLES, we should make some changes before we use MAKE, the changes are:

In /usr/local/spooles.2.2/Tree/src/makeGlobalLib change: drawTree.c to draw.c
In /usr/local/spooles.2.2/Make.inc change: CC = /usr/lang-4.0/bin/cc to CC = /usr/bin/cc
On spooles.2.2 folder make:

```
$ sudo make lib
$ cd MT/src/
$ sudo make
```

6. Compile ARPACK, we should make some changes before we use MAKE, the changes are:

In /usr/local/ARPACK/ARmake.inc change:
home = $(HOME)/ARPACK to home = /usr/local/ARPACK
PLAT = SUN4 to PLAT = linux
FC = f77 to FC = gfortran
FFLAGS = -O -cg89 to FFLAGS = -O2
MAKE = /bin/make to MAKE = /usr/bin/make
In /usr/local/ARPACK/UTIL/second.f, comment out the line
EXTERNAL ETIME
by adding a * in the beginning of line
EXTERNAL ETIME
On ARPAK folder make: $ sudo make lib

7. Compiling CalculiX:

Go to the directory /usr/local/CalculiX/ccx_2.12/src
Rename Makefile to Makefile_original
Copy Makefile_MT to Makefile
Open the file, Makefile change libarpack_intel.a to libarpack_linux.a
Then compile by

```
$ sudo make
```

By this point CalculiX compiling will create an executable named "ccx_2.12_MT" which is the one we have been looking for, now just put it on your work folder /usr/local/bin, remember to add run permissions to this file.
(Note: change the names to “ccx_2.12” for easy use)

8. Usage:

```
$ export OMP_NUM_THREADS=#
$ ccx_2.12 -i jobname
```

*Change “#” for the number of threads you want to use, 1 by default.