

FAST-1.0 User Installation and Verification Guide

Developed Under NQA-1-2017

March 2020

Kenneth J Geelhood
David V Colameco
Travis J Zipperer
Walter G Luscher
Michelle Bales
James R Corson

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor Battelle Memorial Institute, nor any of their employees, makes **any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights**. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or Battelle Memorial Institute. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from the
Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062;
ph: (865) 576-8401
fax: (865) 576-5728
email: reports@adonis.osti.gov

Available to the public from the National Technical Information Service
5301 Shawnee Rd., Alexandria, VA 22312
ph: (800) 553-NTIS (6847)
email: orders@ntis.gov <<https://www.ntis.gov/about>>
Online ordering: <http://www.ntis.gov>

FAST-1.0 User Installation and Verification Guide

Developed Under NQA-1-2017

March 2020

Kenneth J Geelhood
David V Colameco
Travis J Zipperer
Walter G Luscher
Michelle Bales
James R Corson

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99354

Project Summary and Document Characteristics

Project Name	FRAPCON and FRAPTRAN Fuel Performance Code Development and Assessment
--------------	---

Approvals

Role	Name	Signature	Date
Project Manager	Tara O'Neil		
Lead Software Developer	Ken Geelhood		
Code Custodian	David Colameco		

Revision History

Revision	Date	Comments
0	March 2020	Original

Introduction

The purpose of this document is to provide the user information about the installation of FAST-1.0 on their computers or servers. General information about the code and supported operating systems is described in Section 1.0. Self-service oriented FAST-1.0 software licensing steps are described in Section 2.0. An installation verification test suite is provided with FAST-1.0 and described in Section 3.0. A convenience script for converting FRAPCON to FAST inputs is discussed in Section 4.0.

FAST-1.0 was developed and released under a software quality assurance program based upon NQA-1-2017. FAST-1.0 is the latest baseline code and the result of a merger of FRAPCON-4.0 and FRAPTRAN-2.0 with other software and methodology developments. As such the installation verification test suite contains both steady state and transient Anticipated Operation Occurrences (AOOs). Capability to model accident conditions, such as Reactivity Initiated Accidents (RIAs) and Loss Of Coolant Accidents (LOCAs), are targeted for a later release of FAST. FRAPTRAN-2.0 will continue to be used for accident conditions until the release of FAST with accident condition modeling capabilities.

Definitions

These definitions were adopted by the software project.

Assessment	A review, evaluation, inspection, test, check, surveillance, or audit to determine and document whether items, processes, systems, or services meet specified requirements and perform effectively. (NQA-1-2017)
Acceptance Testing	The process of exercising or evaluating a system or system component by manual or automated means to ensure that it satisfies the specific requirements and to identify differences between expected and actual results in the operating environment. (NQA-1-2017)
Configuration Item	A collection of hardware or software elements treated as unit for the purpose of configuration control. (NQA-1-2017)
Configuration Management (software)	The process of identifying and defining the configuration items in a system (i.e. software and hardware), controlling the release and change of those items throughout the system's life cycle, and recording and reporting the status of configuration items and change requests. (NQA-1-2017)
Baseline	A specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for use and further development, and that can be changed only by using an approved control process. (NQA-1-2017)
Error	A condition deviating from an established baseline, including deviations from the current approved computer program and its baseline requirements. (NQA-1-2017)
Confluence	Confluence is an easy to use web-based tool that is utilized for electronically documenting software in a wiki format. Documents can also be controlled. It offers the ability to document, collaborate, and share.
Graded Approach	The process of ensuring that the level of analysis, documentation, and actions used to comply with a requirement is commensurate with: <ol style="list-style-type: none"> the relative importance to safety, safeguards, and security the magnitude of any hazard involved the life-cycle stage of a facility or item the programmatic mission of a facility the particular characteristics of a facility or item the relative importance of radiological and nonradiological hazards any other relevant factors (NQA-1-2017)
HDI	A web search engine that houses PNNL's Lab-level requirements and procedures and considerations for conducting work. The content is delivered via graphical workflows (step-by-step flowcharts with steps for each activity), through narrative work controls (listing of requirements and considerations for managing specific risks and hazards), or in forms or exhibits (linked documents that include greater detail).
Independent	(Independent Reviews or Independent Testing) Person sufficiently independent with respect to the material/product they are reviewing/testing; they did not perform the work they are reviewing or testing. Staff also possess enough subject matter expertise to adequately review/test/evaluate.
Operating Environment	A collection of software, firmware, and hardware elements that provide for the execution of computer programs. (NQA-1-2017)

Software Design Verification	The process of determining if the product of the software design activity fulfills the software design requirements. (NQA-1-2017)
Software Engineering	<p>(a) The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.</p> <p>(b) The study of approaches in (a). (NQA-1-2017)</p>
Test Case	A set of test inputs, execution conditions, and expected results developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement. (NQA-1-2017)
Test Plan (Procedure)	A document that describes the approach to be followed for testing a system or component. Typical contents identify items to be tested, tasks to be performed, and responsibilities for the testing activities. (NQA-1-2017)
Verification	Mathematical proof of the correctness of algorithms, by confirming that code subroutines and functions produce the expected numerical output.
Validation	The process of evaluating software to determine whether it satisfies specified requirements, by comparing code predictions to experimental data.
Unit test	Process or code developed to test the numeric accuracy and functionality of new or modified subroutines and functions.
Unit test suite	Set of unit tests created while developing and maintaining FAST.
Verification test suite	Set of input files that exercise all the code options, used to verify that code changes do not negatively impact code performance, and that results are as expected.
Validation test suite	Set of input files used to validate the codes' predictions against experimental measurements, to quantify the accuracy, bias, and uncertainty of code predictions.

Acronyms and Abbreviations

AOO	Anticipated Operational Occurrences
FGR	Fission Gas Release
hash	Hash Function used to map data of an arbitrary size to fixed-size values.
LOCA	Loss of Coolant Accident
LWR	Light Water Reactor
NRC	United States Nuclear Regulatory Commission
NQA-1	Nuclear Quality Assurance – 1
MD5	Widely used 128-bit message-digest hash value for uniquely identifying files.
PNNL	Pacific Northwest National Laboratory
POC	Point of Contact
QA	Quality Assurance
QAP	Quality Assurance Plan
RIA	Reactivity Initiated Accident
SHA-1	Widely used 160-bit message-digest hash value for uniquely identifying files.

Contents

Project Summary and Document Characteristics	ii
Introduction	iii
Definitions	iv
Acronyms and Abbreviations.....	vi
Contents	vii
1.0 General Software Information.....	9
1.1 Acquiring the Software.....	10
2.0 Self-Service Licensing	12
3.0 Installation Verification Suite.....	15
3.1 Expected Differences.....	16
3.2 FAST-1.0 Code Change Requests.....	18
4.0 Convenience Scripts.....	19
4.1 Execution of FRAPCON_to_FAST.py	19
4.2 FAST-1.0 AIG	19
4.3 FRAPlot	19
5.0 References.....	20
Appendix A – Windows Server 2019 Standard Differences	A.1
Appendix B – FAST Change Request Form.....	B.2

Figures

Figure 1: Window's CertUtil Hashing Example	11
Figure 2: Linux md5sum and sha1sum Example.....	11
Figure 3: FAST Website Licensing Link Location	12
Figure 4: FAST Licensing Webpage.....	12
Figure 5: Windows License Request Generation.....	13
Figure 6: Linux License Request Generation	13
Figure 7: FAST License Request Website File Selection	13
Figure 8: FAST License Request Website Submission	14
Figure 9: FAST License Request Completion	14

Tables

Table 1 – FAST-1.0 Executable Identification9

Table 2 – FAST-1.0 Tested Operating Systems9

Table 3 – FAST-1.0 File Identification 10

Table 4 – FAST-1.0 File Identification 10

Table 5 – FAST Verification Suite Tests..... 15

1.0 General Software Information

FAST-1.0 has been compiled on Windows 10 and Red Hat Enterprise Linux 7.7 operating systems. The compiled executables in Table 1 were then tested on Windows and Linux operating systems listed in Table 2 using the test suites available in Section 3.0. MacOS users are invited to use the Linux executable.

Table 1 – FAST-1.0 Executable Identification

Executable	OS	Hash
FAST.exe	Windows 10	MD5 5b7ac46985d22069133326e3fe994126 SHA-1 f412f6490721c1259b0d5dd76969c9ac5db34feb
FAST	Linux	MD5 f450ffac9f581d39303d52f4318ffab0 SHA1 e46e1cac223c1f8d10f0a1a60cd8f257f0b8afd2

Due to the large variety of operating systems available to our users' group, not all possibilities can be tested. The list in Table 2 is an attempt to provide wide coverage of currently supported systems: Windows and Linux. If your installation on a current operating system not listed below results in larger than expected differences or difficulties, please contact PNNL and we will attempt to identify and correct/mitigate the issue. The list in Table 2 below is comprehensive enough that unforeseen difficulties with installation should be rare.

Table 2 – FAST-1.0 Tested Operating Systems

Operating System	Version	Comments
Windows 10 Enterprise	1803	OS Build 17134.1304
Windows 10 Enterprise	1909	OS Build 18363.418
Windows Server 2016 Standard	1607	OS Build 14393.693
Windows Server 2019 Standard	1809	OS Build 17763.737
Debian Buster	10.0	Linux Kernel 4.19.0-6
Fedora	31.0	Linux Kernel 5.3.7-301
openSUSE Leap	15.1	Linux Kernel 4.12.14-lp152
Red Hat Enterprise Linux	8.1	Linux Kernel 4.18.0-147
Red Hat Enterprise Linux	7.7	Linux Kernel 3.10.0-1062
Ubuntu LTS	18.04.1	Linux Kernel 4.15.0-72

1.1 Acquiring the Software

The FAST website is <https://fast.labworks.org> and it contains limited information for visitors not logged in but more documentation and access to the software for logged in visitors. Visitors without log in credentials may apply using the “Join Us” tab on the homepage.

The Codes tab on the homepage lists the three codes available for download, FAST-1.0, FRAPCON-4.0 and FRAPTRAN-2.0. This document will focus on FAST-1.0. Follow the link to FAST-1.0 and download the zip or tgz file that is applicable to your target machine’s file decompression software. Typically, Windows users would choose the zip file while Linux users can decompress either zip or tgz files with preinstalled software. MacOS users are invited to use the Linux executable within the compressed files.

The following hash values are associated with the zip and tgz files along with the text file listing of the contents. The zip and tgz files contain identical contents.

Table 3 – FAST-1.0 File Identification

File	OS	Hash
FAST.zip	Windows/Linux	MD5 bf31043e0551fa9b7b5d637871dd6519 SHA-1 ac94f12352b78ce4bb62e586eb98fa92b3ab299d
FAST.tgz	Windows/Linux	MD5 8d2fd36281c318e9dae31ad14d1c9d7b SHA1 4b05b1d845b926051cd7bb79f5b3514f6bf8bdfa
md5sum_listing	Windows/Linux	MD5 70a9076c591690e4777190c2e0c70252 SHA1 7308ba768debab5ef98a381f99f6d499696ebefe
sha1sum_listing	Windows/Linux	MD5 05777e22a053cd8b4ca9f150592b027e SHA1 1d0fdd2a827b3ad803595e12a1c40cc873a89af1

The Linux command `tar -xzf FAST.tgz` will decompress the folder. The FAST.zip and FAST.tgz files have the same contents described in Table 4 below:

Table 4 – FAST-1.0 File Identification

Folder	Description
Convenience_Scripts	Python 3 conversion script, AIG, FRAPlot (See Section 4.0)
Installation_Verification	Installation Verification Tests (See Section 3.0)
Linux_Executable	Linux Executable
Linux_FAST_Licensing	Linux Licensing software (See Section 2.0)
Windows_Executable	Windows Executable
Windows_FAST_Licensing	Windows Licensing software (See Section 2.0)

These next steps are primarily for users needing to establish traceability from the identification in Table 1 and 3 above to the final installation location on their machines. These steps may also be used to verify that the executable file was not corrupted during the download and/or transfer process to the final location.

Once the executable and associated files are installed on your target machine, the SHA-1 or MD5 should be checked and compared to Table 1 and Table 3. Other, more secure, hashing algorithms are available such as SHA-2 however it is assumed that MD5 and/or SHA-1 will provide the user confidence the executable was not altered. Users that require hashes such as SHA-2 are encouraged to contact PNNL through the website.

Windows offers a pre-installed utility CertUtil and a downloadable utility FCIV available on the support.microsoft.com website. The pre-installed utility will be demonstrated here. Open a command prompt and navigate to the location of your executable or provide the path of the executable. Then execute the command: `CertUtil -hashfile SHA1 FAST.exe` (or MD5 in place of SHA1) as shown in Figure 1:

```
G:\VirtualBox_Shared_G\FAST-1.0_Release\FAST-1.0\Windows_Executable>CertUtil -hashfile FAST.exe MD5
MD5 hash of FAST.exe:
5b7ac46985d22069133326e3fe994126
CertUtil: -hashfile command completed successfully.

G:\VirtualBox_Shared_G\FAST-1.0_Release\FAST-1.0\Windows_Executable>CertUtil -hashfile FAST.exe SHA1
SHA1 hash of FAST.exe:
f412f6490721c1259b0d5dd76969c9ac5db34feb
CertUtil: -hashfile command completed successfully.

G:\VirtualBox_Shared_G\FAST-1.0_Release\FAST-1.0\Windows_Executable>
```

Figure 1: Window's CertUtil Hashing Example

Linux offers md5sum and sha1sum command line utilities as demonstrated in Figure 2 below:

```
cola105@linux-q3y0:/media/sf_VirtualBox_Shared_G/FAST-1.0_Release/FAST-1.0/Linux_Executable> md5sum FAST
f450ffac9f581d39303d52f4318ffab0 FAST
cola105@linux-q3y0:/media/sf_VirtualBox_Shared_G/FAST-1.0_Release/FAST-1.0/Linux_Executable> sha1sum FAST
e46e1cac223c1f8d10f0a1a60cd8f257f0b8afd2 FAST
cola105@linux-q3y0:/media/sf_VirtualBox_Shared_G/FAST-1.0_Release/FAST-1.0/Linux_Executable> █
```

Figure 2: Linux md5sum and sha1sum Example

A matching md5 or sha1 hash provides a very high likelihood that the executables were not corrupted during the download and transfer to the final location on the target machine. If the hashes do not match, try downloading from the website again. Changes to the files change the hash values.

The location of the FAST executable on your target machine depends largely on preferences and your organizations rules or guidelines. The Installation Verification that is described in Section 3.0 can be performed on your target machine by updating the associated python script.

2.0 Self-Service Licensing

Installations of FRAPCON 4.0 and FRAPTRAN 2.0 do not require these licensing steps.

A successfully copied FAST-1.0 executable in the target location of the target computer must now be licensed for it to operate. Licensing has been set up as self-service through the FAST website <https://fast.labworks.org> and the page https://fast.labworks.org/fast_license. Figure 3 below shows the website after clicking the “Codes” tab highlighted in a red box.

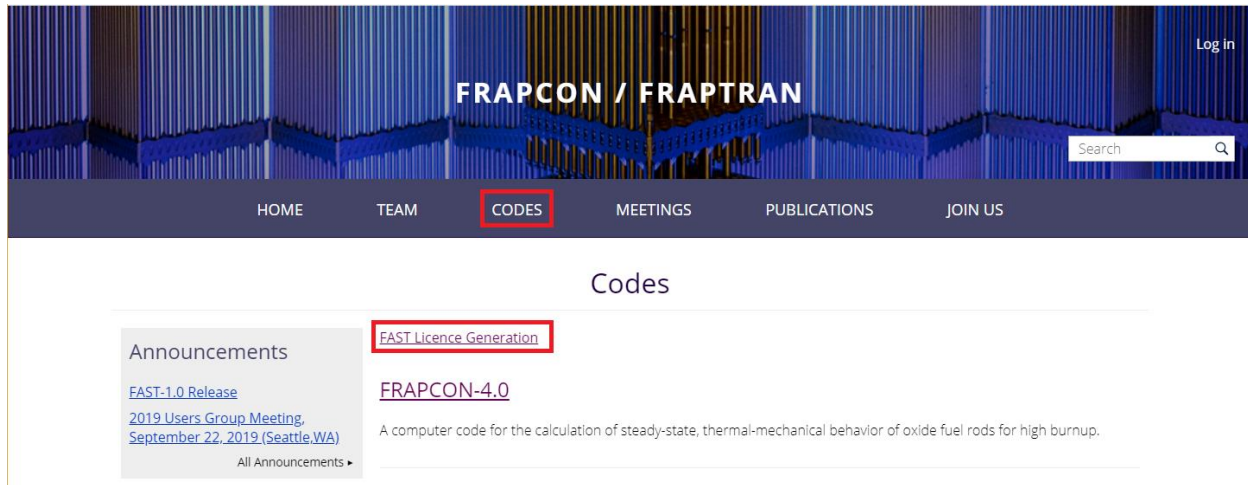


Figure 3: FAST Website Licensing Link Location

The link “FAST License Generation” in Figure 3 above will take you to the screen in Figure 4 below.

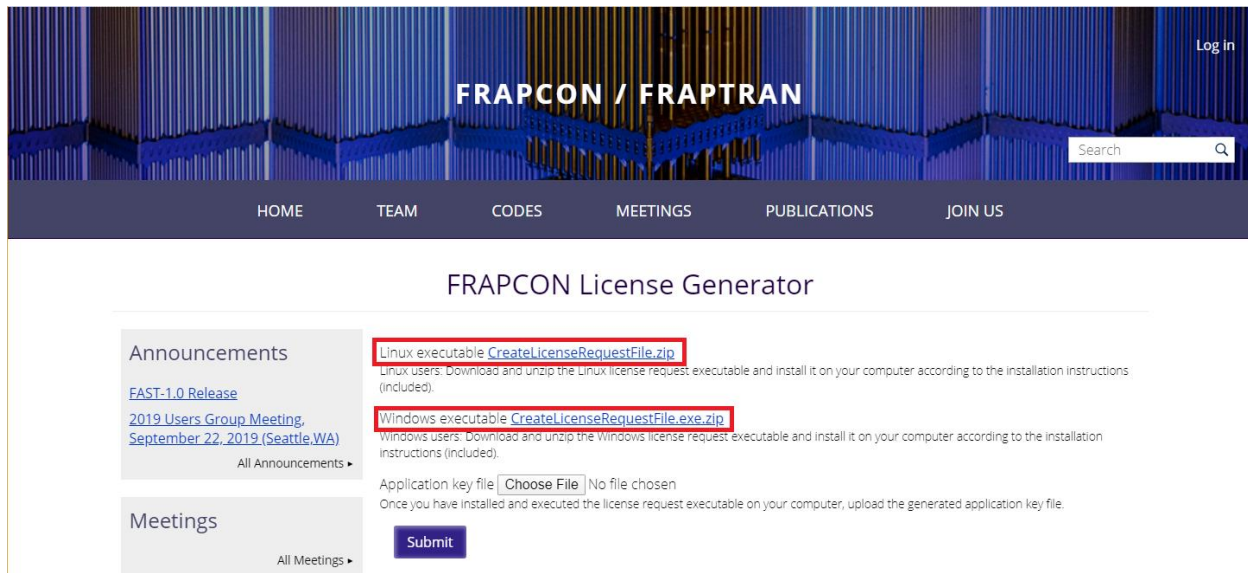


Figure 4: FAST Licensing Webpage

First select the target machine’s OS: Linux or Windows highlighted in red boxes in Figure 4. MacOS users are invited to use the Linux licensing software. There is a limit of 5 license files per website user. If you need more licenses, please contact PNNL through the website e-mail.

With the `CreateLicenseRequestFile(.exe)` on the target machine, execute the program as shown in Figures 5 and 6 depending upon your target machine's operating system.

```

Command Prompt
G:\FAST>CreateLicenseRequestFile.exe
Creating license request file "license.request"...
CreateLicenseRequestFile.Exe v.1.3.0.0

SystemUniqueID:

Created license request file.

Please visit https://fast.labworks.org/fast_license for self-service licensing activation.
Note: Instructions are also available at the above website location.

G:\FAST>

```

Figure 5: Windows License Request Generation

```

Test]$ ./CreateLicenseRequestFile
Creating license request file "license.request"...
CreateLicenseRequestFile.Exe v.1.3.0.0

SystemUniqueID:

Created license request file.

Please visit https://fast.labworks.org/fast_license for self-service licensing activation.
Note: Instructions are also available at the above website location.

[Test]$

```

Figure 6: Linux License Request Generation

A file named "license.request" is generated. Select "Choose File."

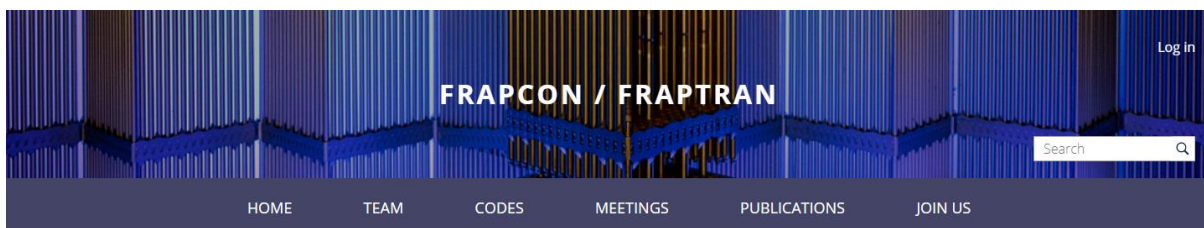


Figure 7: FAST License Request Website File Selection

Clicking “Choose File” highlighted in Figure 7 opens a File Explorer in Windows and a File Manager in Linux. Once a license.request file is selected, it’s name will appear to the right of the “Choose File” Icon as seen in Figure 8 below. Do not change the name of the license.request file before uploading it. Click the “Submit” button which is also highlighted in red to create a FASTProduct.key

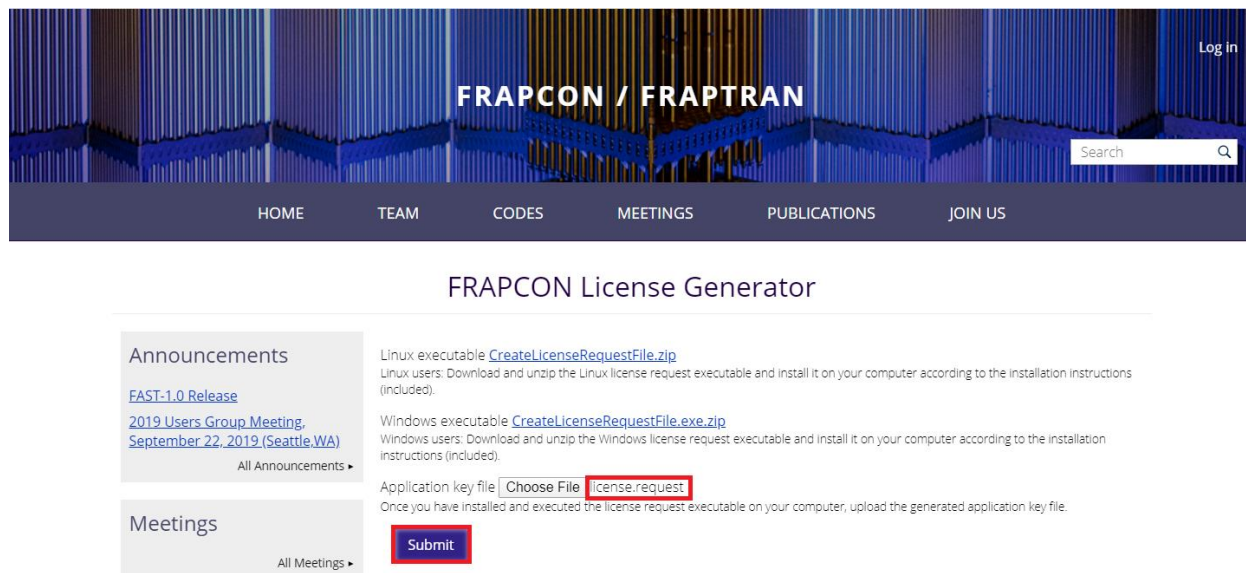


Figure 8: FAST License Request Website Submission

Clicking “Submit” generates the FASTProduct.key which shows up as a download link as highlighted in Figure 9 below. Download this file and place it in the same directory as the FAST executable on your target machine. Please do not change the name of this FASTProduct.key file. Your installation of FAST-1.0 is now complete and ready for verification testing in Section 3.0. It may be necessary to “right-click” the FASTProduct.key and choose “Save as”.

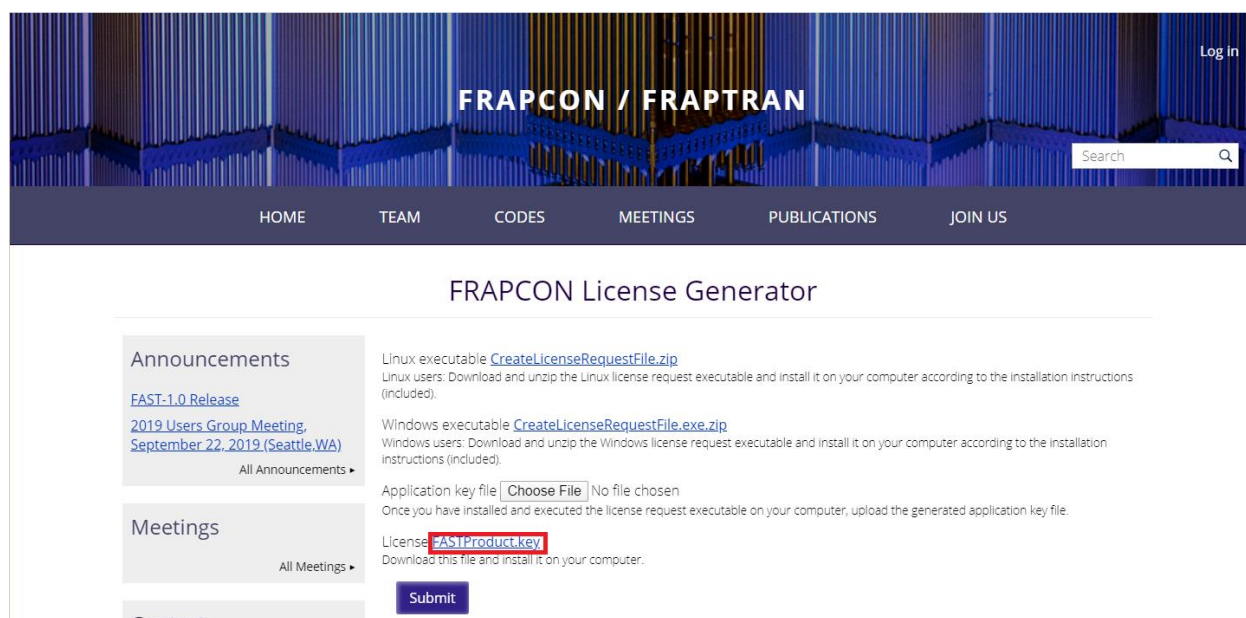


Figure 9: FAST License Request Completion

3.0 Installation Verification Suite

Now that your copy of FAST-1.0 has been downloaded and installed on the target machine in the target location (Section 1.0) and a FASTProduct.key (Section 2.0) has been obtained, the installation test suite can be executed. The FASTProduct.key must be in the location where you are running FAST-1.0 (where the outputs are generated). Typically, the user will run FAST-1.0 in a manner where the FAST-1.0 outputs are generated in the same location as the FAST-1.0 inputs by typing FAST-1.0 and the input file on the command line. The verification script keeps the inputs and outputs separate as a convenience to the user. The FASTProduct.key must be placed in the User_Verification_Outputs folder prior to executing the verification script.

This section will describe the test suite and the steps to execute the testing automation script. The script verification_diff.py is written in Python 3 and used with Python 3.6 to 3.8 (as of January 2020 Python 2 was frozen and will no longer be developed). The script will execute the set of supplied FAST-1.0 inputs and provided files with the extension “.diff” that contain the differences between the FAST-1.0 output files generated on your machine and the output files generated by the FAST development team.

Use of a Python version other than 3.6 to 3.8 is possible; it is up to the person performing the testing to ensure that the diff files reflect the differences in output files. This can be done by spot checking a diff file and its associated files.

Tests included in the Verification Suite are listed in Table 5 below.

Table 5 – FAST Verification Suite Tests

File Name	Test Name	Comments
FGR_BWstudR1.in	FGR_BWstudR1	B&W Studsvik Rod 1 for predicted fission gas release.
Oxide_N05.in	Oxide_N05	N05 Case for predicted oxide thickness.
Strain_GE7.in	Strain_GE7	GE7 Case for permanent hoop strain following a power ramp.
Temperature_681-3.in	Temperature_681-3	IFA-681 Rod 3 for predicted fuel centerline temperature.
Temperature_ifa_432r3.in	Temperature_ifa_432r3	IFA-432 Rod 3 for predicted fuel centerline temperature.
Void_24i6.in	Void_24i6	24i6 case for predicted end of life rod void volume.

To execute the Verification Suite first download and transfer the compressed file to the target machine. The verification_diff.py script must be updated if the FAST-1.0 executable is installed in a location other than that of the folder structure described in Table 4.

3.1 Expected Differences

The differences listed in this section were generated from machines which are less than 5 years old. Differences in architecture may result in round off error. The user is invited to use their professional judgement or contact PNNL via the website contact information for interpretation of larger than expected differences.

Sample difference files are included in the Installation_Verification/Sample_Difference_Files/ folder. The testing conducted at PNNL using virtual machines and physical machines showed that the verification_diff.py script only shows file names. The script ignores expected differences in execution dates, usernames, and computer names. When Linux and Windows outputs are compared it was found that the peak axial node in a flat power profile changed but the fuel performance values remained the same.

Please see the Sample_Difference_Files folder for text files with differences. See Appendix A for a Windows difference file listing. The following is a listing of the Centos 8 differences:

```
*****
*****
|FGR_BWstudR1|

+++ /home/cola105/Installation_Verification_Centos8/Verification_Cases/Outputs/FGR_BWstudR1
--- /home/cola105/Installation_Verification_Centos8/User_Verification_Outputs/FGR_BWstudR1

#####
#####

*****
*****
|Oxide_N05|

+++ /home/cola105/Installation_Verification_Centos8/Verification_Cases/Outputs/Oxide_N05
--- /home/cola105/Installation_Verification_Centos8/User_Verification_Outputs/Oxide_N05

#####
#####

*****
*****
|Strain_GE7|

+++ /home/cola105/Installation_Verification_Centos8/Verification_Cases/Outputs/Strain_GE7
--- /home/cola105/Installation_Verification_Centos8/User_Verification_Outputs/Strain_GE7

#####
#####

*****
*****
|Temperature_681-3|

+++ /home/cola105/Installation_Verification_Centos8/Verification_Cases/Outputs/Temperature_681-3
--- /home/cola105/Installation_Verification_Centos8/User_Verification_Outputs/Temperature_681-3

+++ line:32602   3   53.0   6   0.63   2.02  464.0  471.9  479.7  2.43  601.1  716.4  809.8
0.0 -1741.2 -1077.3 0.1199 0.32467 659.4 324.4 0.0 0.01 11.3
--- line:32602   3   53.0   3   0.63   2.02  464.0  471.9  479.7  2.43  601.1  716.4  809.8
0.0 -1741.2 -1077.3 0.1199 0.32467 659.4 324.4 0.0 0.01 11.3
+++ line:32603   4   63.5   3   0.81   2.30  464.1  473.0  481.9  2.33  613.9  748.8  860.9
0.0 -1653.1 -1036.1 0.1203 0.32485 689.9 334.3 0.0 0.01 11.6
```

```

--- line:32603 4 63.5 6 0.81 2.30 464.1 473.0 481.9 2.33 613.9 748.8 860.9
0.0 -1653.1 -1036.1 0.1203 0.32485 689.9 334.3 0.0 0.01 11.6
+++ line:32609 10 123.5 6 2.26 3.76 464.3 478.8 493.3 1.87 663.4 916.6 1152.1
0.0 -1226.0 -836.2 0.1219 0.32577 876.0 393.7 0.0 0.03 14.4
--- line:32609 10 123.5 3 2.26 3.76 464.3 478.8 493.3 1.87 663.4 916.6 1152.1
0.0 -1226.0 -836.2 0.1219 0.32577 876.0 393.7 0.0 0.03 14.4
+++ line:32611 12 143.2 9 2.89 4.60 464.4 482.1 499.9 1.71 687.4 1007.4 1308.7
0.0 -993.5 -727.4 0.1298 0.32612 970.9 425.5 0.0 0.04 15.7
--- line:32611 12 143.2 1 2.89 4.60 464.4 482.1 499.9 1.71 687.4 1007.4 1308.7
0.0 -993.5 -727.4 0.1298 0.32612 970.9 425.5 0.0 0.04 15.7
+++ line:32613 14 157.2 9 3.36 5.47 464.6 485.6 506.7 1.56 708.7 1101.3 1475.5
0.0 -746.5 -611.7 0.1344 0.32643 1071.6 459.3 0.0 0.04 16.6
--- line:32613 14 157.2 3 3.36 5.47 464.6 485.6 506.7 1.56 708.7 1101.3 1475.5
0.0 -746.5 -611.7 0.1344 0.32643 1071.6 459.3 0.0 0.04 16.6
+++ line:32616 17 169.9 9 3.84 5.33 464.6 485.1 505.7 1.54 700.4 1082.9 1446.5
0.0 -771.5 -623.4 0.1350 0.32648 1083.2 455.9 0.0 0.05 17.5
--- line:32616 17 169.9 1 3.84 5.33 464.6 485.1 505.7 1.54 700.4 1082.9 1446.5
0.0 -771.5 -623.4 0.1350 0.32648 1083.2 455.9 0.0 0.05 17.5
+++ line:32617 18 176.9 9 4.12 5.57 464.7 486.2 507.6 1.49 704.2 1108.0 1493.1
0.0 -695.7 -588.0 0.1365 0.32659 1122.2 466.3 0.0 0.05 18.1
--- line:32617 18 176.9 1 4.12 5.57 464.7 486.2 507.6 1.49 704.2 1108.0 1493.1
0.0 -695.7 -588.0 0.1365 0.32659 1122.2 466.3 0.0 0.05 18.1
+++ line:32618 19 182.2 4 4.34 5.74 464.8 486.9 509.0 1.45 706.5 1125.7 1526.5
0.0 -640.7 -562.2 0.1375 0.32667 1151.8 473.8 0.0 0.05 18.5
--- line:32618 19 182.2 1 4.34 5.74 464.8 486.9 509.0 1.45 706.5 1125.7 1526.5
0.0 -640.7 -562.2 0.1375 0.32667 1151.8 473.8 0.0 0.05 18.5
+++ line:32621 22 199.7 1 5.12 6.58 465.0 490.3 515.6 1.28 714.2 1207.6 1684.4
0.0 -372.7 -436.8 0.1419 0.32703 1312.3 510.5 0.0 0.06 20.0
--- line:32621 22 199.7 6 5.12 6.58 465.0 490.3 515.6 1.28 714.2 1207.6 1684.4
0.0 -372.7 -436.8 0.1419 0.32703 1312.3 510.5 0.0 0.06 20.0
+++ line:32622 23 207.1 9 5.48 6.65 465.1 490.7 516.2 1.25 712.0 1210.9 1691.9
0.0 -344.0 -423.3 0.1425 0.32709 1345.2 514.2 0.0 0.07 20.7
--- line:32622 23 207.1 1 5.48 6.65 465.1 490.7 516.2 1.25 712.0 1210.9 1691.9
0.0 -344.0 -423.3 0.1425 0.32709 1345.2 514.2 0.0 0.07 20.7
+++ line:32623 24 217.7 9 5.99 6.61 465.2 490.6 516.0 1.24 710.0 1205.9 1681.4
0.0 -346.9 -424.7 0.1427 0.32711 1350.7 513.8 0.0 0.07 21.7
--- line:32623 24 217.7 1 5.99 6.61 465.2 490.6 516.0 1.24 710.0 1205.9 1681.4
0.0 -346.9 -424.7 0.1427 0.32711 1350.7 513.8 0.0 0.07 21.7
+++ line:32624 25 232.5 9 6.71 6.68 465.4 491.1 516.7 1.22 709.2 1211.3 1689.7
0.0 -314.8 -409.7 0.1435 0.32716 1375.8 518.0 0.0 0.08 23.1
--- line:32624 25 232.5 1 6.71 6.68 465.4 491.1 516.7 1.22 709.2 1211.3 1689.7
0.0 -314.8 -409.7 0.1435 0.32716 1375.8 518.0 0.0 0.08 23.1
+++ line:32625 26 235.6 9 6.87 7.07 465.5 492.6 519.7 1.15 712.4 1247.7 1759.5
0.0 -196.6 -354.3 0.1451 0.32729 1453.2 534.5 0.0 0.08 23.4
--- line:32625 26 235.6 1 6.87 7.07 465.5 492.6 519.7 1.15 712.4 1247.7 1759.5
0.0 -196.6 -354.3 0.1451 0.32729 1453.2 534.5 0.0 0.08 23.4
+++ line:32631 32 281.6 9 9.23 7.52 466.1 495.0 523.8 1.01 704.6 1309.7 1867.9
0.0 53.0 -237.4 0.1543 0.32758 1647.1 568.7 0.1 0.11 27.9
--- line:32631 32 281.6 1 9.23 7.52 466.1 495.0 523.8 1.01 704.6 1309.7 1867.9
0.0 53.0 -237.4 0.1543 0.32758 1647.1 568.7 0.1 0.11 27.9
+++ line:32632 33 295.6 9 10.00 7.59 466.3 495.4 524.5 0.99 703.5 1319.4 1887.2
0.0 108.3 -211.5 0.1574 0.32765 1679.7 576.2 0.1 0.12 29.4
--- line:32632 33 295.6 6 10.00 7.59 466.3 495.4 524.5 0.99 703.5 1319.4 1887.2
0.0 108.3 -211.5 0.1574 0.32765 1679.7 576.2 0.1 0.12 29.4
+++ line:32633 34 308.3 9 10.66 7.17 466.4 493.9 521.4 1.03 697.8 1278.5 1810.2
0.0 -7.6 -265.8 0.1595 0.32757 1609.7 560.3 0.1 0.13 30.7
--- line:32633 34 308.3 1 10.66 7.17 466.4 493.9 521.4 1.03 697.8 1278.5 1810.2
0.0 -7.6 -265.8 0.1595 0.32757 1609.7 560.3 0.1 0.13 30.7
+++ line:32634 35 314.4 9 11.00 7.52 466.5 495.4 524.2 0.99 700.9 1316.0 1881.6
0.0 126.6 -203.0 0.1657 0.32768 1686.1 578.6 0.1 0.13 31.3
--- line:32634 35 314.4 1 11.00 7.52 466.5 495.4 524.2 0.99 700.9 1316.0 1881.6
0.0 126.6 -203.0 0.1657 0.32768 1686.1 578.6 0.1 0.13 31.3
+++ line:32635 36 321.9 6 11.41 7.52 466.6 495.5 524.3 0.98 700.6 1318.1 1885.4
0.0 142.9 -195.3 0.1682 0.32770 1689.9 580.8 0.1 0.14 32.1
--- line:32635 36 321.9 1 11.41 7.52 466.6 495.5 524.3 0.98 700.6 1318.1 1885.4
0.0 142.9 -195.3 0.1682 0.32770 1689.9 580.8 0.1 0.14 32.1
+++ line:32636 37 331.5 6 11.94 7.55 466.8 495.7 524.7 0.97 699.6 1323.3 1895.8
0.0 175.1 -180.2 0.1707 0.32774 1711.0 585.2 0.1 0.14 33.2
--- line:32636 37 331.5 1 11.94 7.55 466.8 495.7 524.7 0.97 699.6 1323.3 1895.8
0.0 175.1 -180.2 0.1707 0.32774 1711.0 585.2 0.1 0.14 33.2

```

```
#####
#####
```

```
*****
*****
```

```
|Temperature_ifa-432r3|
```

```
+++ /home/cola105/Installation_Verification_Centos8/Verification_Cases/Outputs/Temperature_ifa-432r3
```

```
--- /home/cola105/Installation_Verification_Centos8/User_Verification_Outputs/Temperature_ifa-432r3
```

```
#####
#####
```

```
*****
*****
```

```
|Void_24i6|
```

```
+++ /home/cola105/Installation_Verification_Centos8/Verification_Cases/Outputs/Void_24i6
```

```
--- /home/cola105/Installation_Verification_Centos8/User_Verification_Outputs/Void_24i6
```

```
#####
#####
```

3.2 FAST-1.0 Code Change Requests

The adoption of NQA-1-2017 raises the bar on quality however issues with code may arise. Please use the form in Appendix B to request changes and e-mail it to PNNL via the contact information on the FAST website.

4.0 Convenience Scripts

A convenience script that converts FRAPCON input to FAST input is included. FAST-1.0 performs calculations for normal operations and AOOs. FRAPTRAN initialized with FRAPCON will be used for design basis accidents such as RIAs and LOCAs. The convenience script contains preliminary conversions for standalone FRAPTRAN inputs however the FAST-1.0 code is only for normal operations and AOOs.

4.1 Execution of FRAPCON_to_FAST.py

The Python3 script `FRAPCON_to_FAST_Inputs.py` will convert FRAPCON inputs to FAST inputs. The script retains the order of the variables from the FRAPCON input in the FAST input for easier comparisons. Changes to Fortran NAMELIST block beginnings and endings are included for compliance with the latest Fortran standards. The previous files used Intel formatting which can be slightly different from the general Fortran standard. The user of the script will find indices added to arrays and comments added for each variable. Please note that the FRAPCON input file can only have one instance of each NAMELIST block to properly work.

Execution of the script is best performed in a new directory with input files containing the ending with “.in”. The script will convert FRAPCON files by writing a new file with the ending `_FAST.in`. Executing the command `python3 FRAPCON_to_FAST_Inputs.py all` will convert all files in the working directory. Executing the command `python3 FRAPCON_to_FAST_Inputs.py 24i6.in` will only convert the file name given to `24i6_FAST.in`.

4.2 FAST-1.0 AIG

The Microsoft Excel-based tool for generating input, developed for use with the FAST codes is FAST-1.0 AIG. This auto input-generator is similar to the AIG for FRAPCON and FRAPTRAN but with differences for FAST.

4.3 FRAPlot

The Microsoft Excel-based tool for plotting developed for use with the FAST codes is FRAPlot. Instructions for its use are located in Appendix C, Section C.3, in FAST-1.0: A Computer Code for Thermal-Mechanical Nuclear Fuel Analysis under Steady-state and Transients, PNNL-29720.

5.0 References

The following documents were utilized to develop and/or are referenced in this document:

- 10 CFR, Title 10 Code of Federal Regulations, United State Government, 2018.
- ASME NQA-1-2017, Quality Assurance Requirements for Nuclear Facility Applications, January 18, 2018.
- DOE G 414.1-4, Safety Software Guide for use with 10 CFR 830 Subpart A, Quality Assurance Requirements, and DOE O 414.1C, Quality Assurance, November 2010.
- NRC NUREG/BR-0167 Software Quality Assurance Program and Guidelines, February 1993.

Appendix A – Windows Server 2019 Standard Differences

The following difference is a comparison of Windows Server 2019 Standard with files run on Windows 10. Only the file names are written to the .txt file. Rounding may occur with your setup depending upon hardware architecture.

```
*****
*****
|FGR_BWstudR1|

+++ Z:\Installation_Verification\Verification_Cases\Outputs\FGR_BWstudR1
--- Z:\Installation_Verification\User_Verification_Outputs\FGR_BWstudR1

#####
#####

*****
*****
|Oxide_N05|

+++ Z:\Installation_Verification\Verification_Cases\Outputs\Oxide_N05
--- Z:\Installation_Verification\User_Verification_Outputs\Oxide_N05

#####
#####

*****
*****
|Strain_GE7|

+++ Z:\Installation_Verification\Verification_Cases\Outputs\Strain_GE7
--- Z:\Installation_Verification\User_Verification_Outputs\Strain_GE7

#####
#####

*****
*****
|Temperature_681-3|

+++ Z:\Installation_Verification\Verification_Cases\Outputs\Temperature_681-3
--- Z:\Installation_Verification\User_Verification_Outputs\Temperature_681-3

#####
#####

*****
*****
|Temperature_ifa-432r3|

+++ Z:\Installation_Verification\Verification_Cases\Outputs\Temperature_ifa-432r3
--- Z:\Installation_Verification\User_Verification_Outputs\Temperature_ifa-432r3

#####
#####

*****
*****
|Void_24i6|

+++ Z:\Installation_Verification\Verification_Cases\Outputs\Void_24i6
--- Z:\Installation_Verification\User_Verification_Outputs\Void_24i6

#####
#####
```


Appendix B – FAST Change Request Form

The following form can e-mailed to PNNL via the FAST website contact information to request a change or report a software deficiency.

(select one)

No. _____

(Assigned by PM)

1. RELEASED VERSION ID. _____
2. REQUESTOR _____ PHONE ____/____
3. ERROR REPORTED: ☐ YES ☐ NO E-MAIL _____
4. PROJECTED START DATE: _____
5. ESTIMATED TIME FOR COMPLETION: _____
6. DESCRIPTION (attach continuation sheets if necessary)

7. REASON FOR MODIFICATION (attach continuation sheets if necessary)

8. CODE PERFORMANCE IMPACTS (If applicable)

To be completed by PM

ACTION: APPROVED DENIED

DATE RECEIVED _____

ASSIGNED TO _____

Approval for new modification _____ Date _____

Pacific Northwest National Laboratory

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99354
1-888-375-PNNL (7665)

www.pnnl.gov