

Integration Module - ABAQUS

Xuguang Wang



Civil Engineering
UNIVERSITY OF TORONTO

2017 UT-SIM Workshop

Overview

- **Introduction**
- **Communication Method**
- **System Setup**
 - ◇ Accessing the User Subroutine
 - ◇ Accessing the Data Exchange Library
- **Running Analysis with the Integration Module**
 - ◇ Preparing the input file
 - ◇ Defining the user element
- **Example**
 - ◇ Braced Portal Frame (Example Structure I in the manual)

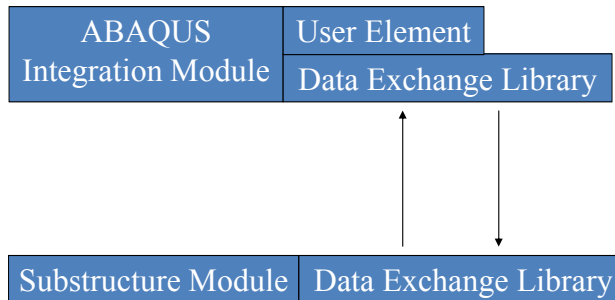


2017 UT-SIM Workshop

2

Communication Method

- ❑ **User element**
 - ◆ Defined in the integration module for exchanging data
- ❑ **Data exchange library**
 - ◆ A dynamic link library contains UTNP



System Setup

- ❑ **Accessing the user subroutine**
 - ◆ Required software:
 - ABAQUS
 - Fortran and C++ Compilers
 - ◆ Link compilers to enable the subroutine function
 - Commanded package: ABAQUS 6.13, Visual Studio 2012, Intel Composer XE 2013
 - Add the path of compilers to the system environment variables
 - Add command lines in the batch file of running ABAQUS
 - A step-by-step instruction is in the manual chapter 6.5

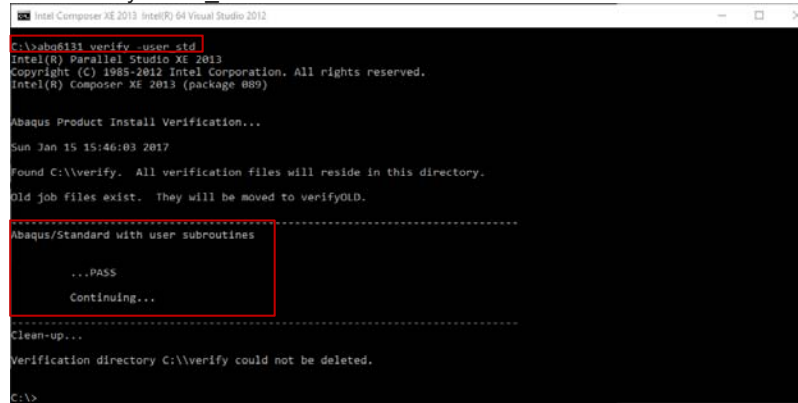
```
abq6131.bat x
1 @echo off
2 @call ifortvars.bat intel64 vs2012
3 "C:\SIMULIA\Abaqus\6.13-1\code\bin\abq6131.exe" *
4
```



System Setup

❑ Accessing the user subroutine

- ◆ Verifying the subroutine function
 - Verify the subroutine function is working properly using ABAQUS 'verify -user_std' command



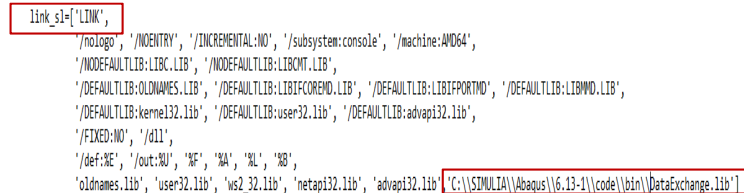
```
Intel Composer XE 2013 Intel(R) 64 Visual Studio 2012
C:\>abq6131 verify -user_std
Intel(R) Parallel Studio XE 2013
Copyright (C) 1985-2012 Intel Corporation. All rights reserved.
Intel(R) Composer XE 2013 (package 889)

Abaqus Product Install Verification...
Sun Jan 15 15:46:03 2017
Found C:\verify. All verification files will reside in this directory.
Old job files exist. They will be moved to verifyOLD.
-----
Abaqus/Standard with user subroutines
-----
...PASS
Continuing...
-----
Clean-up...
Verification directory C:\verify could not be deleted.
C:\>
```

System Setup

❑ Accessing the data exchange library

- ◆ Adding the data exchange library to ABAQUS library folder
 - Copy the library files into the 'bin' folder under the ABAQUS directory
- ◆ Customizing the ABAQUS environment file
 - Adding the path of added library into the ABAQUS environment file 'abaqus_v6.env'



```
link_s1={"LINK",
'/noLogo', '/NOENTRY', '/INCREMENTAL:NO', '/subsystem:console', '/machine:AMD64',
'/NODEFAULTLIB:LIBC.LIB', '/NODEFAULTLIB:LIBMT.LIB',
'/DEFAULTLIB:OLDNAMES.LIB', '/DEFAULTLIB:LIBFCOREMD.LIB', '/DEFAULTLIB:LIBFPORTMD', '/DEFAULTLIB:LIBBMD.LIB',
'/DEFAULTLIB:kernel32.lib', '/DEFAULTLIB:user32.lib', '/DEFAULTLIB:advapi32.lib',
'/FIXED:NO', '/dll',
'/def:SE', '/out:BU', 'BF', 'SA', 'SL', 'SB',
'oldnames.lib', 'user32.lib', 'ws2_32.lib', 'netapi32.lib', 'advapi32.lib' 'C:\SIMULIA\Abaqus\6.13-1\code\bin\DataExchange.lib'}
```

Integration Structure Input File

❑ ABAQUS Standard Input File

- ◇ Generate from ABAQUS user interface
- ◇ Create using text editor (Template provided in the package)

❑ User Element

- ◇ Only for the purpose of transmitting data
 - Usually have the same coordinates as the user element in the substructure
- ◇ Define user element using ABAQUS key words in the input file using text editor
 - `*USER ELEMENT`
 - Define basic information of the user elements and calls the subroutine
 - `*UEL PROPERTY`
 - The subroutine file, *IntSub.for*, is provided in the package



Integration Structure Input File

❑ User Element

```
25 *****
26 *** User Element ***
27 *****
28 *USER ELEMENT, TYPE=U1, NODES=2, COORDINATES=4, PROPERTIES = 6, VARIABLES = 12
29 1,2, 6
30 *ELEMENT, TYPE=U1, ELSET=user
31 999,4,1
32 *UEL PROPERTY, ELSET=user
33 8090, 1, 3, 4, 2, 1,
34 22962.4673816532, 12628.9427262300, 0, -22962.4673816532, -12628.9427262300, 0,
35 12628.9427262300, 6945.69062337916, 0, -12628.9427262300, -6945.69062337916, 0,
36 0, 0, 0, 0, 0, 0,
37 -22962.4673816532, -12628.9427262300, 0, 22962.4673816532, 12628.9427262300, 0,
38 -12628.9427262300, -6945.69062337916, 0, 12628.9427262300, 6945.69062337916, 0,
39 0, 0, 0, 0, 0, 0,
```



Running the Analysis

- ❑ Having the input file and the subroutine file, *IntSub.for*, in the same folder
- ❑ Creating batch file to run the analysis or type **ABAQUS** command in the command line
- ❑ **Abaqus job= file name user = IntSub**
 - ◆ The ABAQUS command is NOT case sensitive.
 - ◆ Do not include file extension
 - ◆ The execution information can be shown in the command window while running analysis by adding '- interactive' command. However, log file will be created when this command is absence.
- ❑ **Run the integration structure when the substructure is ready**

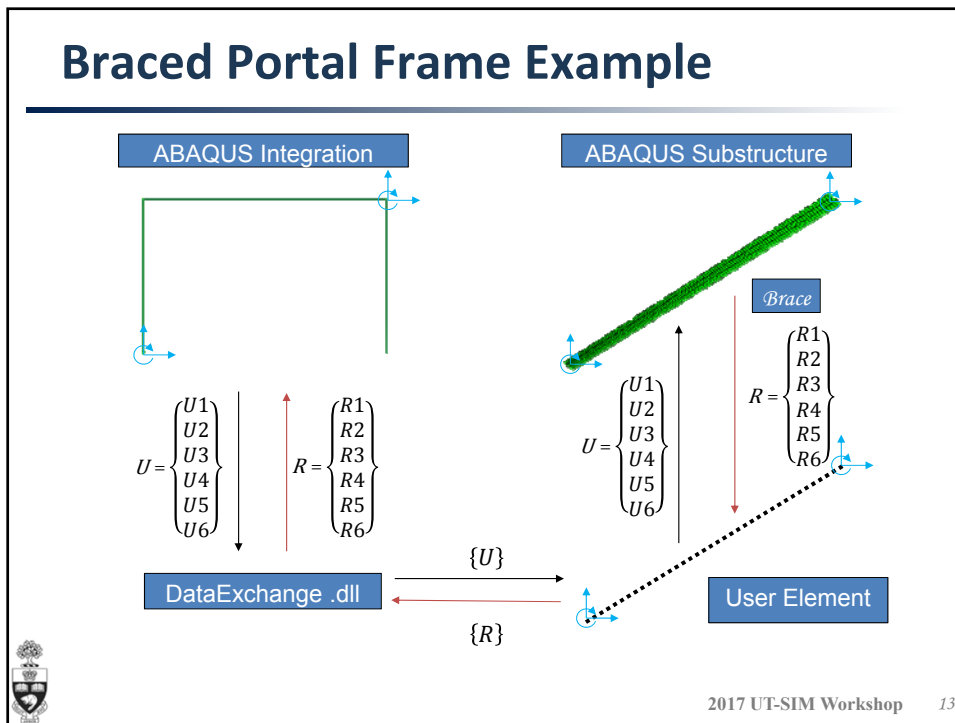


Braced Portal Frame Example

- ❑ Portal frame with BRB brace
- ❑ The brace is buckling restrained and has bi-linear material response
- ❑ Frame is modeled as the integration module using **ABAQUS** frame element
- ❑ Brace is modeled using **ABAQUS** solid element (C3D8R, 8 nodes linear brick element with reduced integration) as the substructure



Braced Portal Frame Example



Braced Portal Frame Example

- Push over analysis with 165mm lateral displacement is carried out on both the hybrid and Standalone ABAQUS models

