User Interface I

Jonathan R. Madsen

Department of Nuclear Engineering
Texas A&M University
College Station, TX, USA 77843

madsen_jr@tamu.edu

A͞TM | **NUCLEAR ENGINEERING**
TEXAS A&M UNIVERSITY

Outline

Geant4 UI Command

- A UI command consists of:

  - Command directory
  - Command
  - Parameter(s)

  /run/verbose 1
  /vis/viewer/flush
  /vis/scene/add/axes -1 -1 -1 0.5 m

- A parameter can be a type of string, boolean, integer, or doule
  - Space is delimiter
  - Use double quotes (") for string with space(s)

- A parameter may be omitted if there is a default value or does not require a parameter

- If there are multiple parameters and you want to use the default for a preceding parameter, use "!" as a placeholder:
  - /dir/command ! second

Command submission

- Geant4 UI command can be issued by:

  - (G)UI interactive command submission

  - Macro file

  - Hard-coded Implementation

    - Slow but no need for the targeting class pointer

    - Should NOT be used inside an event loop

```
G4UImanager::GetUIpointer()->ApplyCommand("/run/verbose 1");
```

- The availability of individual commands, the ranges of the parameters, the available candidates on individual command parameter may vary according to the implementation of your application and may even vary dynamically during the execution of your job

- Some commands are only available for limited Geant4 application state(s)

  - *e.g.* "/run/beamOn [n]" is available only for *Idle* states

Command refusal

- Command will be refused in case of:

    - Wrong application state

    - Wrong type of parameter

    - Insufficient number of parameters

    - Parameter is out of its range (integer and double type parameter)

    - Parameter is out of its candidate list (string type parameter)

    - Command is not found

Application States

- Application State(s)

    - G4State_PreInit — Initial state

    - G4State_Init — while G4RunManager::Initialize() is occuring

    - G4State_Idle — Ready for starting run

    - G4State_GeomClosed — BeamOn() called. Geom, physics, cuts are locked

    - G4State_EventProc — Event is being processed

    - G4State_Quit — Application is done. Geant4 managers are deleted.

    - G4State_Abort — G4State_Quit + G4Exception occurred

- In certain interactive terminals (such as TCSH), current state is displayed in "PS1" field:

    - *PreInit* > /run/initialize

    - *Idle* > /run/beamOn 100

Macro File

- Macro file is an ASCII file containing UI commands
- All commands must be given their full-path directories
- Use "#" for comment line
  - First "#" to end of line will be ignored
  - Comment lines will be echoed if **/control/verbose 2** is set
- Macro file can be executed
  - Interactively or in another macro file
    - **/control/execute file_name**
  - Hard-coded

```cpp
G4UImanager* UI = G4UImanager::GetUIpointer();
UI->ApplyCommand("/control/execute file_name");
```

Available Commands

- You can get a list of the available commands (including your custom ones) by
    - Plain text
        - **/control/manual [directory]**
    - HTML
        - **/control/createHTML [directory]**
- List of built-in commands can also be found in Section 7.1 of the *User's Guide for Application Developers*
- Switching between UI sessions
    - Set the environment variable: **G4UI_USE_<interface>=1**
    - *e.g.* **export G4UI_USE_TCSH=1**
    - There are priorities within Geant4, *i.e.* if G4UI_USE_TCSH=1 and G4UI_USE_QT=1, then Geant4 will choose Qt for the UI interface

Alias

- Alias can be defined by
    - /control/alias [name] [value]
- Alias is dereferenced by curly brackets { and }
- Frequently used commands can be aliased to shorten them
    - **/control/alias tv /tracking/verbose**
    - {**tv**} **1**
- Aliases can be used recursively
    - **/control/alias file1 dirA/run.mac**
    - **/control/alias file2 dirB/run.mac**
    - **/control/alias run 1**
    - **/control/execute {file{run}}**

Loop

- /control/loop and /control/foreach commands execute a macro more than once
- Aliased variable name(s) can be used inside the macro file
- **/control/loop [macroFile] [counterName] [initialValue] [finalValue] [stepSize]**
  - *counterName* is aliased to the number as a loop counter
- **/control/foreach [macroFile] [counterName] [valueList]**
  - *counterName* is aliased to a value in *valueList*
  - *valueList* must be enclosed in double quotes

Loop Example

- in init.mac: **/control/loop run.mac Ekin 10. 20. 5.**

- in run.mac: **/control/foreach params.mac pname "p e- alpha"**

- in params.mac:
    - **/gun/energy {Ekin} MeV**
    - **/gun/particle {pname}**
    - **/run/beamOn 100**

- Result — 9 Simulations of 100 events:
    - proton at 10., 15., 20. MeV
    - electron at 10., 15., 20. MeV
    - alpha particle at 10., 15., 20. MeV

Command Syntax
oooo

Macro File
ooooo

G4UIExecutive
●

Interactive Terminal Commands
o

G4UIExecutive - Batch mode / Interactive mode

- In your main():

```cpp
int main(int argc, char** argv)
{
  ...
  if(argc > 1)
  { // batch mode
    G4String cmd = "/control/execute ";
    G4String fname = argv[1];
    UImanager->ApplyCommand(cmd + fname);
  } else
  { // interactive mode : define UI session
    G4UIExecutive* ui = new G4UIExecutive(argc, argv);
    ui->SessionStart();
    delete ui;
  }
}
```

Terminal Commands

- Interactive terminal supports some Unix-like commands

  - cd — Change current directory

  - pwd — Display current directory

  - ls — list available UI commands and sub-directories

  - history — show previous commands

  - !*historyID* — re-issue previous command

  - arrow keys and tab — TC-shell only

  - ?*UIcommand* — show current parameter values of command

  - help — help

  - exit — job termination

- All these commands are not valid in a non-interactive terminal, *i.e.* you cannot use them in a macro file