

### ■ A.3.1 Execution procedure

To perform a finite element analysis using Abaqus, one must prepare a text file and save it with an extension ".inp". Assuming a file named 'feajob.inp' exists, the program can be executed by typing the following command at the shell prompt.

```
abaqus job=feajob
```

The analysis proceeds entirely in the background without any user interaction. Abaqus writes analysis results in several files, all starting with the same job name but with different extensions. The file with '.dat' extension is a text file containing a summary of the execution steps and any other requested output. Any warnings or errors produced during execution are saved in this file as well. The file with extension '.odb' contains the entire analysis database in a binary form. This file is used for graphical viewing of results using the abaqus viewer as follows.

```
abaqus viewer database=feajob
```

The viewer has menus for standard operations of plotting contours of nodal and element solutions, deformed shape etc.

The input file can either be created manually by typing appropriate commands in a text file or by using graphical preprocessor for abaqus. To use the preprocessor issue the following command at the shell prompt.

```
abaqus cae
```

The main screen of the cae environment contains the usual file manipulation, visualization, and help menus. Near the top left hand side, under the buttons for file save, print, etc. is a drop down list for accessing different modules for creating a finite element model. The order in which these modules appear in the list reflects the intended sequence in which the modules are to be used.

The following sections define typical commands and procedures from abaqus cae to create input file. For trusses, structural frameworks, and other situations involving simple structured finite element meshes, the direct creation of the input file is generally the most convenient option. For modeling situations involving generation of complex unstructured finite element mesh the use of abaqus cae is recommended. The following sections show examples of both approaches.