Workflow overview

Table of contents

1 Illustrative overview

An overview of a Coreform IGA for Abaqus workflow is shown in the illustration below. In this illustration, processes are shown as rounded rectangles while files are shown as icons representing databases or text-files.

- A process that is colored purple signifies that it is a standard Abaqus process or workflow. A process that is colored green signifies that it is part of the Coreform IGA custom application or solver workflow. A process that is bi-colored purple—green signifies that the process can utilize both standard Abaqus and Coreform IGA workflows.
- A file-icon that is colored purple signifies that it is a file that would exist in a standard Abaqus workflow and users should be able to interact with these files as they normally would in a standard Abaqus workflow. A file-icon that is colored green signifies that the file's existence is unique to the Coreform IGA workflow and users should exercise caution before interacting with these files.

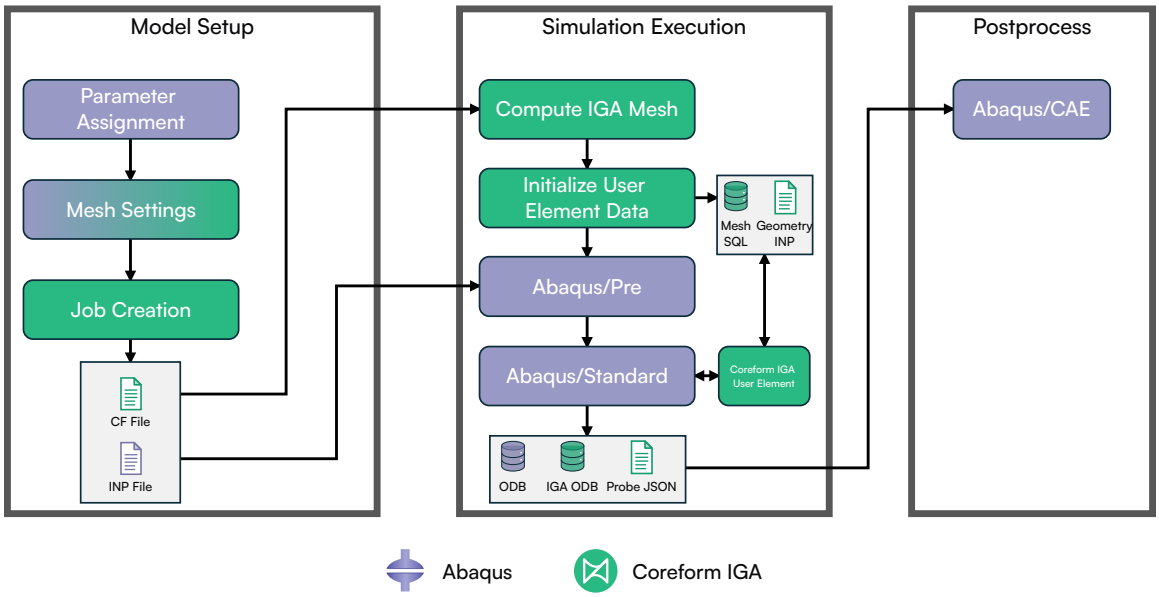

Figure 1: Overview of a Coreform IGA for Abaqus workflow.

2 Narrative overview

1. Model Setup *(Only Abaqus/CAE license required)*

   No Coreform licenses are required for the model setup phase, only an Abaqus/CAE license is required.

   a. Parameter assignment
      a. Traditional model setup in Abaqus/CAE, importing geometry, creating & assigning materials, defining steps & load conditions, etc.
   b. Mesh settings
      a. Coreform IGA for Abaqus supports simulations comprising both regular Abaqus elements as well as Coreform IGA meshes. Users can use Abaqus/CAE's built-in meshing tools for regular Abaqus meshes, or the Coreform IGA mesh tool to define IGA meshes.
   c. Job creation
      a. Simulations involving IGA meshes require the use of the Coreform IGA job tool to create and submit the job.
      b. This step produces two files:
         a. An Abaqus `INP` file whose name is `<job-name>.inp`
         b. A Coreform `CF` file, whose name is `<job-name>.cf` that provides input for the trimmed-mesh generation routine

2. Simulation Execution

   a. Compute trimmed mesh *(Coreform IGA for Abaqus license required)*
      a. Generates the trimmed spline mesh as defined by the user.
      b. Computes element & basis function data (e.g., quadrature, extraction operators, extension operators, function splitting).
      c. Writes out a `SQL` database file, whose name is `<job-name>.sql`, that contains the trimmed mesh and element/basis function data.
   b. Initialize user element data & translate Abaqus keywords *(Does not require a Coreform IGA for Abaqus license)*
      a. Copies the original Abaqus setup deck, `<job-name>.inp`, to the protected source deck `<job-name>_source.inp`.
      b. Reads `<job-name>_source.inp`, translates applicable Abaqus keywords into forms compatible with Coreform IGA, and writes the translated, runnable setup deck back to `<job-name>.inp`.
         - See the Abaqus keyword support guide for the translation behavior of each supported semantic case.
      c. Converts element and basis function data into efficient data structures for Abaqus user-elements and stores this data in the same `SQL` database file, whose name is `<job-name>.sql`.

   d. Generates the IGA geometry deck `<job-name>_iga_geom.inp`, which contains `*USER ELEMENT` and related definitions. The translated `<job-name>.inp` references this geometry deck with an `*INCLUDE` keyword.
3. c. Execute Abaqus/Pre *(Does not require a Coreform IGA for Abaqus license)*
      a. Regular Abaqus/Pre execution
   d. Execute Abaqus/Standard with Coreform IGA User Elements *(Coreform IGA for Abaqus license required)*
      a. Abaqus/Standard is executed with the `uniquelibs` option.
         - When submitted via Abaqus/CAE this option is automatically set.
         - If submitting a job via the command line, the user must specify `uniquelibs` manually in the command and make the Coreform user-element libraries in `<coreform-install-dir>/bin` discoverable to Abaqus/Standard. On Windows, add `<coreform-install-dir>/bin` to the `PATH` environment variable; on Linux/macOS, add it to `LD_LIBRARY_PATH` (equivalently, set `usub_lib_dir` in `abaqus_v6.env`). In `uniquelibs` mode either mechanism is sufficient to locate the `*_usub` libraries.
      b. Abaqus/Standard loads shared libraries, `*_usub.[so|dll]` packaged by Coreform that provide the user element functionality.
         - If no Coreform IGA for Abaqus license is available the simulation will pause until a license becomes available.
      c. Coreform IGA for Abaqus creates `<job-name>_iga_results.sql` at the start of the analysis. This results database is separate from the `<job-name>.sql` mesh and runtime-input database. It contains a copy of the IGA mesh together with the requested IGA result frames and fields written as the analysis proceeds.
      d. Coreform IGA for Abaqus creates an ODB file with the name `<job-name>_iga.odb`.
         - Coreform IGA for Abaqus queries Abaqus/Standard for output field variables and stores them on a linear hex-tet mesh computed during Coreform IGA Mesh within this ODB file.
      e. If the user specified a Coreform IGA for Abaqus *probe*, Coreform IGA computes the probe data and saves it to `probe_data.json`.
      f. During execution, Abaqus will query Coreform IGA for Abaqus to provide linear system information (e.g., stiffness matrix, mass matrix, force vector).
      g. After completion of the simulation, the Coreform IGA for Abaqus license is released.

3. Postprocessing
   a. Visualize and analyze results in Abaqus/CAE
      a. The `<job-name>.odb` file contains any regular Abaqus elements defined by the user, as well as nodes from the Coreform IGA for Abaqus user-elements, as well

as surface meshes generated by Coreform IGA Mesh that are used for boundary conditions, load conditions, constraints, contact, etc., and is compatible with Abaqus/CAE.

b. The `<job-name>_iga.odb` file contains the linear hex-tet mesh generated by Coreform IGA Mesh and the (currently hard-coded to a limited set) output field variables.
c. The `probe_data.json` file contains Coreform IGA for Abaqus probe variables.
    a. While not natively supported by Abaqus/CAE, skilled analysts can use Abaqus/CAE's integrated Python console to `import json` and use the `json` module to extract the data as Python objects, and then use the Abaqus Python interface to convert to `XYData` objects for plotting in Abaqus/CAE.
    b. Alternatively, the `json` file can be visualized in 3rd-party tools like Python (`matplotlib`), Julia, Matlab, etc.

b. Visualize and analyze results in ParaView

> **Important**
>
> Alpha feature
>
> Coreform IGA includes an alpha ParaView reader that can open `<job-name>_iga_results.sql` and visualize its IGA mesh and supported result fields. The results database itself is written by Coreform IGA during analysis; the alpha designation applies to the current ParaView reader and postprocessing workflow. Its interface, compatibility requirements, and supported result fields can change.